Evidenzbasierte, softwarebezogene Anforderungsanalyse für Patientenregister in der medizinischen Forschung

Doris Lindörfer

2022

Aus dem Institut für Medizinische Informationsverarbeitung, Biometrie und Epidemiologie (IBE)
Institut der Universität München

Vorstand/Direktor: Prof. Dr. Ulrich Mansmann

Evidenzbasierte, softwarebezogene Anforderungsanalyse für Patientenregister in der medizinischen Forschung

Dissertation
zum Erwerb des Doktorgrades der Humanbiologie
an der Medizinischen Fakultät der
Ludwig–Maximilians–Universität zu München

vorgelegt von

Doris Lindörfer

aus

Reichenbach

Jahr

2022

Mit Genehmigung der Medizinischen Fakultät
der Universität München

Berichterstatter:	Prof. Dr. Ulrich Mansmann
Mitberichterstatter:	Prof. Dr. Annette Peters
	Prof. Dr. Sebastian Baumeister
Dekan:	Prof. Dr. med. Thomas Gudermann
Tag der mündlichen Prüfung:	02.08.2022

"In order to write about life, first you must live it."

— Ernest Hemingway (1899-1961)

Eidesstattliche Versicherung

Lindörfer, Doris

Name, Vorname

Ich erkläre hiermit an Eides statt,

dass ich die vorliegende Dissertation mit dem Titel

Evidenzbasierte, softwarebezogene Anforderungsanalyse für Patientenregister in der medizinischen Forschung

selbständig verfasst, mich außer der angegebenen keiner weiteren Hilfsmittel bedient und alle Erkenntnisse, die aus dem Schrifttum ganz oder annähernd übernommen sind, als solche kenntlich gemacht und nach ihrer Herkunft unter Bezeichnung der Fundstelle einzeln nachgewiesen habe.

Ich erkläre des Weiteren, dass die hier vorgelegte Dissertation nicht in gleicher oder in ähnlicher Form bei einer anderen Stelle zur Erlangung eines akademischen Grades eingereicht wurde.

München, 30.09.2021 Doris Lindörfer

Ort, Datum Unterschrift Doktorandin

Inhaltsverzeichnis

Abkürzungsverzeichnis

AHRQ	Agency for Healthcare Research and Quality
API	Application Programming Interface
CDMS	Clinical Data Management System
CIPROS	Checklist with Items for Patient Registry SOftware Systems
CML	chronische myeloische Leukämie (Chronic Myeloid Leukemia)
CONSORT	CONsolidated Standards Of Reporting Trials
CTMS	Clinical Trial Management System
DBFORM	DataBase FORM generator
DGU	Deutsche Gesellschaft für Unfallchirurgie
DSGVO	Datenschutz-Grundverordnung
eCRF	electronic Case Report Form
EDC	Electronic Data Capture
EMA	European Medicines Agency
EPRD	Endoprothesenregister Deutschland
EU	Europäische Union (European Union)
EUTOS	European Treatment and Outcome Study
IBE	Institut für medizinische Informationsverarbeitung, Biometrie und Epidemiologie
IEC	International Electrotechnical Commission
IEEE	Institute of Electrical and Electronics Engineers
ISO	International Organization for Standardization
IT	Informationstechnik (Information Technology)
MI	Medizinische Informatik (Medical Informatics)
NIH	National Institute of Health
QCA	Qualitative Content Analysis
PRISMA	Preferred Reporting Items for Systematic Reviews and Meta-Analyses
RAD	Requirements Analysis Document

REDCap	Research Electronic Data Capture
SLR	Systematic Literature Review
SaaS	Software as a Service
SRS	Software Requirements Specification
StRS	Stakeholder Requirements Specification
SyRS	System Requirements Specification

Publikationen

Diese kumulative Dissertation basiert auf den beiden im Folgenden aufgeführten Publikationen und deren Supplements. Die Publikationen werden in dieser Arbeit entsprechend als Publikation I und Publikation II bezeichnet. Beide Publikationen wurden jeweils mit Zusatzmaterial veröffentlicht, das online abgerufen werden kann.

Publikation I:

Lindoerfer, D. and Mansmann, U. (2015). A Comprehensive Assessment Tool for Patient Registry Software Systems: The CIPROS Checklist. *Methods Inf Med,* 54(5):447-454.

`https://www.thieme-connect.com/products/ejournals/abstract/10.3414/ME14-02-0026`

Publikation II:

Lindoerfer, D. and Mansmann, U. (2017). Enhancing Requirements Engineering for Patient Registry Software Systems with Evidence-based Components.*J Biomed Inform,* 71:147-153.

`https://www.sciencedirect.com/science/article/pii/S1532046417301090?via%3Dihub`

Die Arbeit wird durch die im Anhang A aufgeführte Publikation und deren Supplement ergänzt. Diese wird als Publikation III bezeichnet.

Anhang A:

Publikation III:

Lindoerfer, D. and Mansmann, U. (2017). Data for the elaboration of the CIPROS checklist with items for a patient registry software system: Examples and explanations. *Data Brief*, 14:494-497.

`https://www.sciencedirect.com/science/article/pii/S2352340917303773?via%3Dihub`

Vorarbeiten zu Publikation I waren zwei Konferenzbeiträge die in Anhang B aufgeführt sind. Den ersten Beitrag präsentierte ich auf der *Medical Informatics Europe 2014 (MIE 2014)*, in Istanbul, als Vortrag. Den zweiten Beitrag präsentierte ich auf der Tagung der *Deutschen Gesellschaft für Medizinische Biometrie und Epidemiologie (GMDS) 2014 (GMDS 2014)*, in Göttingen, auch als Vortrag.

Der GMDS Beitrag wurde ausgewählt um eine Publikation in *Methods of Information in Medicine,* dem offiziellen Journal der GMDS, zu veröffentlichen. Diese Publikation ist Publikation I dieser kumulativen Dissertation.

Vorarbeit zu Publikation II war ein Konferenzbeitrag den ich auf der Tagung *Health Exploring Complexity (HEC) 2016 (HEC 2016)*, in München als Vortrag präsentierte. Dieser Artikel ist auch in Anhang B aufgeführt.

Anhang B:

GMDS 2014 Beitrag:

Lindoerfer, D. and Mansmann, U. (2014). A proposal of Checklist Items for evaluating a patient registry software system.

`https://www.egms.de/static/en/meetings/gmds2014/14gmds124.shtml`

MIE 2014 Beitrag:

Lindoerfer, D. and Mansmann, U. (2014). CIPROS - A checklist with items for a patient registry software system. *Stud Health Technol Inform.*, 205:161-5.

`http://ebooks.iospress.nl/publication/37468`

HEC 2016 Beitrag:

Lindoerfer, D. and Mansmann, U. (2016). Proposing an Evidence-Based Strategy for Software Requirements Engineering. *Stud Health Technol Inform.*, 228:648-52.

`http://ebooks.iospress.nl/publication/44693`

Zusammenfassung

Hintergrund: Patientenregister sind ein wichtiges Instrument in der medizinischen Forschung. Sie werden häufig zur Dokumentation von Patientengeschichten verwendet, einschließlich medizinischer, biologischer und Behandlungsdaten. Zur Realisierung dieser Patientenregister werden heute häufig elektronische Datenerfassungssysteme oder elektronische Fallberichtssysteme eingesetzt. In dieser Arbeit werden sie Patientenregister Softwaresysteme genannt.

Oft ist jedoch nicht klar, ob ein bestehendes System geeignet ist, die Anforderungen des spezifischen Patientenregister Projekts zu erfüllen. Um diese Systeme in der medizinischen Forschung erfolgreich einsetzen zu können, fehlen klare Definitionen und Standards. Es gibt einige Initiativen zur Unterstützung der medizinischen Forschung, indem bestehende Systeme klassifiziert, Open-Source-Code für biomedizinische Softwareanwendungen und semantische Infrastrukturen bereitgestellt und gemeinsame Anforderungen der Informationstechnologie für Kohorten und Register definiert werden. Eine übersichtliche Checkliste die Forscher unterstützt, ein geeignetes System für die Durchführung ihrer Projekte zu finden, gibt es nicht.

Zielsetzung: Mit dieser Arbeit soll eine Checkliste für Patientenregister Softwaresysteme in der medizinischen Forschung erstellt werden, mit deren Hilfe ein geeignetes System für ein bestimmtes Projekt gefunden werden kann. Es gibt einige Initiativen die Infrastrukturen für die medizinische Forschung bereitstellen. Ebenso gibt es Hilfen und Richtlinien, die allgemeinen Anforderungen an Patientenregister betreffend. Ziel dieser Arbeit ist die Erstellung einer evidenzbasierten Checkliste zur Unterstützung der softwarebezogenen Anforderungsanalyse für Patientenregister in der medizinischen Forschung.

Methoden: Es wurde eine systematische Literatursuche in *PubMed*, der *National Library of Medicine*, durchgeführt um Publikationen mit Systembeschreibungen von Patientenregister Softwaresystemen zu ermitteln. Relevante Publikationen wurden ausgewählt. Die ausgewählten Arbeiten wurden mit einer qualitativen Inhaltsanalyse analysiert. Wichtige Begriffe wurden identifiziert, notiert und in eine Liste sortiert. Schließlich wurde eine Checkliste erstellt.

Es wurden bereits bestehende Standards und Instrumente zur Anforderungserhebung sowie Vorgehensmodelle zum Softwareengineering ermittelt. Die entwickelte Checkliste wurde mit den bereits vorhandenen Standards und Templates zur Anforderungserhebung verglichen.

Ergebnisse: Es wurde die evidenzbasierte Checklist with Items for Patient Registry SOftware Systems (CIPROS) Checkliste entwickelt. Die CIPROS Checkliste besteht aus 72 Items, die in zwölf Hauptkategorien unterteilt sind. Jedes Item besteht aus einer Kurzbezeichnung mit dem Namen, der Aspekt/Thema genannt wird, einer Nummer sowie einer Beschreibung. Es sind auch die Referenzen der Publikationen hinzugefügt, aus denen das jeweilige Item abgeleitet wurde.

Weiterhin wurde eine ausführliche Liste aller Items mit Beispielen aus der Literatur zu jedem Item und eigenen Erklärungen erstellt. Die entwickelte Checkliste wurde mit bereits vorhandenen Instrumenten zur Anforderungserhebung verglichen. Mit der entwickelten CIPROS Checkliste können Anforderungen identifiziert werden, die dann in bereits bestehende Standards oder Templates zur Anforderungserhebung übernommen werden können. Die entwickelte Checkliste kann als zusätzliches Instrument zur Anforderungserhebung in den bestehenden Vorgehensmodellen zum Softwareengineering verwendet werden.

Diskussion: Die CIPROS Checkliste ist ein wichtiges Instrument, das Wissenschaftlern hilft, ein geeignetes Softwaresystem für ihr Patientenregisterprojekt zu finden. Mit Hilfe der CIPROS Checkliste können zusätzliche Anforderungen ermittelt werden, an die die Forscher vorher nicht gedacht haben. Die CIPROS Checkliste hat auch eine ökonomische Komponente, denn sie kann die Anforderungserhebung effizienter und kostengünstiger gestalten als traditionelle Methoden. Mit Hilfe der CIPROS Checkliste kann evaluiert werden, inwieweit ein System geeignet ist, die Anforderungen eines Patientenregister Projekts zu erfüllen. Die Methode mit der die CIPROS Checkliste erstellt wurde, kann als Vorlage dienen, um ähnliche Checklisten für andere Themen zu erstellen.

Fazit: Die CIPROS Checkliste beschreibt Themen die für Patientenregister Softwaresysteme wichtig sind, um Patientenregister Forschungsprojekte erfolgreich durchzuführen. Die evidenzbasierte Anforderungserhebung ist eine neue Methode, mit der die CIPROS Checkliste erstellt wurde. Diese kann als Vorlage dienen, um ähnliche Checklisten für andere Themen zu erstellen. Die evidenzbasierte Anforderungserhebung kann als neue Anforderungserhebungsmethode im Software Engineering eingesetzt werden. Mithilfe der CIPROS Checkliste können Softwareentwickler ihre Systeme bewerten. Projektkoordinatoren können Mithilfe der CIPROS Checkliste die Anforderungen ihrer Projekte ermitteln und so ein geeignetes System für ihre Projekte finden. Darüber hinaus kann die CIPROS Checkliste helfen, Standards für die Berichterstattung über Patientenregister zu erstellen.

Summary

Background: Patient registries play a significant role in medical research. They are frequently used to document patient histories, including medical, biological and treatment data. To realize these patient registries, today often Electronic Data Capture (EDC) systems or electronic Case Report Form (eCRF) systems are used. In this thesis they are called *patient registry software systems*.

Often it is unclear, if an existing system is able to fulfill the requirements of a specific patient registry project. In order to be able to use these systems in medical research successfully, clear definitions and standards are missing. There are a number of initiatives to support medical research by classifying existing systems, preparing open source code, which can be used for software applications in biomedical informatics, providing semantic infrastructures, and defining general requirements for cohorts and registry information technology. A clear checklist that helps researchers to find a suitable system to carry out their projects does not exist.

Objective: This work aims to formulate a checklist with criteria for software systems for patient registries in medical research, that can be used to find a suitable system for a particular project. There are some proposals which provide help, concerning the general requirements on patient registries. The aim of this work is to prepare an evidence-based checklist to assist researchers to find software-related requirements to fulfill their patient registry projects successfully with a suitable patient registry software system.

Methods: A Systematic Literature Review (SLR) was conducted in PubMed, to identify publications with system descriptions of patient registry software systems. Relevant publications with descriptions of patient registry software systems were selected. The selected papers were analyzed with a Qualitative Content Analysis (QCA) and important terms were identified, noted and sorted into a list. Finally, a checklist was created.

Existing standards and documents for requirements specification were identified and characterized, as well as some software life cycle models. The prepared checklist was compared with the existing standards and documents.

Results: The evidence-based Checklist with Items for Patient Registry SOftware Systems (CIPROS) checklist was created. The CIPROS checklist consists of 72 items, subdivided into

twelve main categories. Each item consists of a short name, which is called aspect/topic, a number, a description, and references to the publications from which the respective item was derived.

Furthermore, an extensive list with all CIPROS items, with examples from the literature and own explanations, was developed. The created CIPROS checklist was compared with existing requirements specification standards and templates. With the CIPROS checklist requirements can be identified and then transferred to the existing standards or templates. The developed CIPROS checklist can be used as an additional instrument for requirements elicitation in the existing software life cycle models.

Discussion: The CIPROS checklist is an important instrument that helps scientists, to find a suitable system for their patient registry project. The CIPROS checklist can help to find additional requirements, which were not in the mind of the researchers before. The CIPROS checklist has also an economic component, because it makes the requirements specification cheaper as traditional methods, or makes traditional methods more efficient. Using the CIPROS checklist as requirements engineering method is suggested as a new, innovative and evidence-based method, which can be applied for requirements engineering in software projects. It suggests, how this method can be integrated into various software life cycle models. The way the CIPROS checklist was created can serve as a template to create similar checklists for other topics.

Conclusion: The CIPROS checklist describes topics that are important for software systems for patient registries, that research projects for patient registries can be successfully carried out. The evidence-based requirements collection is a new approach that was used to create the CIPROS checklist. This can serve as a template to create similar checklists for other topics. Evidence-based requirements specification can be applied as a new and innovative method for requirements collection in software engineering. Software developers can use the CIPROS checklist to evaluate their systems, and project coordinators can apply the CIPROS checklist to determine the needs of their projects and find a suitable system for their projects. In addition, the CIPROS checklist can assist to generate standards for the reporting about patient registries.

Kapitel 1

Einleitung

1.1 Patientenregister

Patientenregister spielen in der medizinischen Forschung eine wichtige Rolle. Sie werden seit Jahrzehnten eingesetzt um Patientengeschichten und Krankheitsverläufe über einen längeren Zeitraum zu erfassen und so neue Erkenntnisse über spezifische Krankheiten, Krankheitsverläufe und Behandlungsmöglichkeiten zu gewinnen (Jäger et al., 2000; EUTOS, 2011). Dabei werden je nach Fragestellung verschiedene Daten, wie Anamnesedaten, medizinische Daten, Labordaten, und Molekulardaten erfasst. Es existieren vielfältige Einsatzmöglichkeiten für Patientenregister (Mathis-Edenhofer & Piso, 2011). So gibt es Register für bestimmte Krankheiten, Register für Medizinprodukte, Transplantationsregister, Krebsregister, Intensivregister, Traumaregister usw. um nur einige zu nennen.

Die Anzahl der Register steigt ständig und sie haben in den vergangenen Jahrzehnten in der medizinischen Forschung an Bedeutung gewonnen, (Anazawa et al., 2015; Zaletel & Kralj, 2015; NIH, List of Registries, 2021). Im Jahr 2015 zählten Zaletel & Kralj (2015, S. 34) 1028 Register in Europa, 41 davon in Deutschland. In den Vereinigten Staaten von Amerika startete im Jahr 2012 die *Agency for Healthcare Research and Quality (AHRQ)* ein Online Register von Patientenregistern um eine durchsuchbare Datenbank von Patientenregistern bereitzustellen Glicklich et al. (2012). Das *National Institute of Health (NIH)* erstellte eine Liste von Registern, die Informationen über Patientenregister in den USA bereitstellt (NIH, List of Registries, 2021).

Seit 2013 sind in Deutschland alle Bundesländer per Gesetz dazu verpflichtet alle Krebspatienten zu registrieren und klinische Krebsregister aufzubauen (kbv, 2021). In einigen Regionen Deutschlands wird das bereits seit Jahrzehnten gemacht, beispielsweise in Bayern, wo Krebsregister bereits seit den 1970er Jahren existieren (TRM, 2020). Andere Regionen bauen jetzt neue Krebsregister auf.

Neben diesen Krebsregistern gibt es verschiedene andere Register in Deutschland und weltweit. Zum Beispiel:

- Die European Treatment and Outcome Study (EUTOS) chronische myeloische Leukämie (Chronic Myeloid Leukemia) (CML) Register, die von 2008 bis 2014 europaweit, sowohl retrospektiv als auch prospektiv, Baseline, Behandlungs- und Outcome-Daten von Patienten mit CML sammelten, (EUTOS, 2011; Hasford et al., 2011; Hoffmann et al., 2013, 2015, 2017).

- Das Endoprothesenregister Deutschland (EPRD), das Daten zu den jährlich mehr als 450.000 Operationen zu künstlichen Hüft- und Kniegelenken in Deutschland registriert, (EPRD, 2021).

- Die RESIST-Studie, ein translationales Forschungsprojekt über Patienten mit fortgeschrittenem kolorektalem Karzinom, die bei sekundären Resistenzen, eine im Xenograft Modell gefundene experimentelle Zweitlinientherapie erhalten, (Lindoerfer & Mansmann, 2017c; Vangala et al., 2021).

- Das Traumaregister der Deutschen Gesellschaft für Unfallchirurgie (DGU), (TraumaRegister DGU®), (TraumaRegister DGU, 2021). Das TraumaRegister DGU® wurde 1993 gegründet, seitdem sind dem TraumaRegister DGU® über 800 Kliniken beigetreten und es wurden bisher mehr als 400.000 Patienten in das TraumaRegister DGU® eingeschlossen, (TraumaRegister DGU, 2021).

- Das italienische Register für seltene Erkrankungen (Taruscio et al., 2014; Kodra et al., 2018), das seit 2001 Daten von Patienten mit seltenen Erkrankungen in Italien sammelt.

Es gibt medizinische Register für unterschiedliche Zwecke: Public Health, Epidemiologie, Qualitätskontrolle, klinische Register, Register zu bestimmten Krankheiten, usw. (NIH, List of Registries, 2021).

In dieser Arbeit werden forschungsorientierte Patientenregister nach der Definition von Glicklich betrachtet, (Glicklich & Dreyer, 2010).

Diese Patientenregister werden als organisierte Systeme bezeichnet, die mit strukturierten Methoden klinische und andere Daten zu Patienten sammeln um neue Erkenntnisse über bestimmte Krankheiten und Behandlungsmöglichkeiten zu gewinnen, [Übersetzung d. Verfass.] (Glicklich & Dreyer, 2010, vgl. S. 9) [1]

[1] „A *patient registry* is an organized system that uses observational study methods to collect uniform data (clincal and other) to evaluate specified outcomes for a population defined by a particular disease, condition, or exposure, and that serves one or more predetermined scientific, clinical, or policy purposes."

Im Folgenden werden einige Beispiele für forschungsorientierte Patientenregister näher vorgestellt.

Diese Registerbeispiele zeigen, dass gute und effiziente Patientenregister Softwaresysteme nötig sind, um diese Patientenregister erfolgreich durchführen zu können.

1.1.1 Die EUTOS CML-Register

Die **EUTOS CML-Register** (EUTOS, 2011) wurden zwischen 2008 und 2014 aufgebaut und sammelten, sowohl retrospektiv als auch prospektiv, europaweit Baseline-, Therapie- und Outcome-Daten zu Patienten mit chronischer myeloischer Leukämie (CML), (Lindörfer & Müller, 2010; Lindörfer et al., 2011).

Im **EUTOS In-Study Register** wurden retrospektiv Daten zu CML Patienten gesammelt, die bereits an nationalen klinischen Studien teilgenommen hatten. Die Daten aus teilweise sehr heterogenen Strukturen wurden im EUTOS In-Study Register gesammelt und integriert. Mit den Daten aus diesem Register wurde der EUTOS Score entwickelt (Hasford et al., 2011; Hoffmann et al., 2012).

Im **EUTOS Out-Study Register** wurden retrospektiv Daten zu CML Patienten gesammelt, die bereits in nationalen Registern gespeichert wurden. Mit den Daten aus dem Out-Study Register wurde der EUTOS Score validiert, (Hoffmann et al., 2013).

Im **EUTOS Population-based Register** wurden prospektiv Daten zu neu diagnostizierten CML Patienten gesammelt. Die Daten im Population-based Register wurden web-basiert erhoben und zentral gespeichert, (Lindörfer & Müller, 2010). Ergebnisse des EUTOS Population-based Registers zu Inzidenz, klinischen Merkmalen, Therapie und Outcome wurden publiziert, (Hoffmann et al., 2015, 2017).

Die Datenbanken für alle drei EUTOS Register wurden von mir im Institut für medizinische Informationsverarbeitung, Biometrie und Epidemiologie (IBE) aufgebaut, die Daten wurden integriert und gespeichert. Für das Population-based Register kam ein electronic Case Report Form (eCRF) System zum Einsatz. Das im IBE entwickelte eCRF System DataBase FORM generator (DBFORM) wurde erweitert um die Anforderungen für das EUTOS Population-based Register zu erfüllen, (Lindörfer & Müller, 2010; Lindörfer et al., 2011).

1.1.2 Das Endoprothesenregister Deutschland

Das **Endoprothesenregister Deutschland (EPRD)** (EPRD, 2021) sammelt Daten zu Patienten mit Endoprothesen in Deutschland. Es werden jährlich mehr als 450.000 künstliche Hüft- und Kniegelenke in Deutschland eingesetzt, (EPRD, 2021). Bei vielen Patienten verläuft der Einsatz

gut, aber ein Teil davon hat Probleme und es sind Wechseloperationen nötig, (EPRD, 2021). Das EPRD untersucht die Hintergründe. Am EPRD sind verschiedene Organisationen, wie Krankenhäuser, Krankenkassen, Implantat-Hersteller, Registerstelle und Registerpartner beteiligt. Eine Grafik zum Aufbau und Datenfluss zum EPRD zeigt die Zusammenarbeit der verschiedenen Organisationen, (Aufbau des EPRD, 2021). Um alle Beteiligten und die nötigen Informationen zusammenzubringen, ist ein leistungsfähiges System zum Informationsmanagement nötig.

1.1.3 Die RESIST-Studie

Die **RESIST-Studie** ist ein translationales Forschungsprojekt über Patienten mit fortgeschrittenem kolorektalem Karzinom, in dem Xenografts verwendet werden um eine individuelle Zweitlinientherapie für die Patienten zu finden, (Lindoerfer & Mansmann, 2017c; Vangala et al., 2021). Die RESIST-Studie besteht aus drei Teilen:

1. Einem Register für ein AVATARMODELL zu dem die Patienten zustimmen müssen und das die Verwendung einer Tumorprobe dokumentiert.

2. Der zweite Teil besteht aus einer klinischen Studie, auch hier müssen die Patienten einwilligen. Wenn Metastasen diagnostiziert werden, wird der Patient zuerst mit Chemotherapie und Cetuximab behandelt, bis eine sekundäre Resistenz diagnostiziert wird. Für die Zweitlinientherapie wird dann die experimentelle Therapie verwendet, die in dem Xenograft Modell gefunden wurde.

3. Einer großen Sammlung von AVATAREN, mit eingebrachten Proben der individuellen Patienten, die mit alternativen Substanzen behandelt werden.

Die RESIST-Studie braucht eine Informationstechnik (Information Technology) (IT)-Infrastruktur, die eine Datenbank für die klinische Studie, ein eCRF System mit Datenmanagementfunktionen und eine Datenbank für das Biomaterial (OMICS) (scnat wissen, 2018) kombiniert, sowie Anonymisierung und Pseudonymisierung ermöglicht. Das im IBE entwickelte eCRF System DBFORM wurde erweitert um die Anforderungen der RESIST Studie zu erfüllen, (Lindoerfer & Mansmann, 2017c).

1.2 Softwaresysteme für Patientenregister

Diese Registerbeispiele zeigen, dass gute und effiziente Patientenregister Softwaresysteme nötig sind, um diese Patientenregister erfolgreich durchführen zu können.

Zur Erfassung und dem Management der Daten für solche Register werden heutzutage hauptsächlich web-basierte Online-Erfassungssysteme, sogenannte electronic Case Report Form (eCRF) Systeme, verwendet. Diese Systeme stellen Instrumente zur Datenerfassung und zum Datenmanagement bereit, sowie teilweise auch Instrumente zur Analyse, um erste Auswertungen erstellen zu können, mit deren Hilfe man sich einen Überblick über die bereits erfassten Daten verschaffen kann.

1.2.1 Kommerzielle Systeme

Es gibt **kommerzielle Systeme** wie secuTrial® (secuTrial, 2021) oder Marvin (XClinical, 2021). Diese Systeme werden von kommerziellen Firmen entwickelt und verkauft. Dabei gibt es verschiedene Möglichkeiten: Entweder es wird eine Distribution der Software dem Kunden zur Selbstinstallation verkauft, oder es werden dem Kunden die Dienstleistungen zur Datenspeicherung als *Software as a Service (SaaS)* verkauft, dann können die Kunden ihre Daten in einer Softwareinstallation der Firma speichern und müssen sich nicht um Serverbeschaffung und -verwaltung, Installation der Software, Sicherung und Updates kümmern.

1.2.2 Open-source Systeme

Daneben gibt es **open-source Systeme** wie OpenClinica (OpenClinica, 2021). OpenClinica bietet eine open-source Lösung und eine kommerzielle Lösung, die kommerzielle Lösung umfasst auch den Support.

1.2.3 Von einem Konsortium verwaltete Systeme

Außerdem gibt es **Konsortiums Lösungen**, wie das von der Vanderbilt University (Vanderbilt University, 2021) entwickelte Research Electronic Data Capture (REDCap)® Softwaresystem, (REDCap, 2021). Dabei liegen alle Rechte bei Vanderbilt. Akademische Institutionen können Mitglied des Konsortiums werden, dazu müssen sie einen Vertag mit der Vanderbilt Universität abschließen und bekommen dann die Erlaubnis die REDCap® Software zu installieren und zu nutzen.

1.2.4 Eigene Entwicklungen

Manche Institutionen entwickeln auch **eigene** Softwaresysteme um flexibel auf sich ändernde Anforderungen reagieren zu können. Am IBE wurde das System DataBase FORM generator (DBFORM) (Müller & Adelhard, 2002) entwickelt, das in mehreren Projekten des IBE eingesetzt wurde.

1.3 Zielsetzung der Arbeit

Die vorgestellten Registerbeispiele zeigen, dass gute und effiziente Patientenregister Softwaresysteme nötig sind, um solche Patientenregisterprojekte erfolgreich durchführen zu können.

1.3.1 Ausgangslage

Doch wie weiß man, ob ein bestimmtes Softwareregistersystem geeignet ist, die Anforderungen eines Patientenregisterprojekts zu erfüllen? Wie wichtig die Anforderungsermittlung für Softwareprojekte, gerade im medizinischen Bereich ist, hat Kossmann (Kossmann, 2014) betont. Genauso sollen auch die ethischen und rechtlichen Aspekte und Strategien zur Patientenrekrutierung berücksichtigt werden (Glicklich & Dreyer, 2010). Diese Punkte fasste Müller et al. (2010) in einer Checkliste für *Registries for Health Services Research*, (Müller et al., 2010) zusammen. Diese Checkliste berücksichtigt zwar einige Punkte mit Softwarebezug, aber die softwarebezogenen Aspekte wurden in diesen Beiträgen nie explizit diskutiert. Es fehlt eine Checkliste mit der man überprüfen kann, ob ein bestimmtes System geeignet ist, die Anforderungen eines Patientenregisterprojekts zu erfüllen.

Um für zukünftige Projekte eine Entscheidungshilfe bereitzustellen, wurde entschieden, eine Checkliste zu erstellen, mit der überprüft werden kann, ob ein bestimmtes System die Anforderungen eines Patientenregisterprojekts erfüllt.

1.3.2 Bisherige Initiativen

Die Arbeit an der Checkliste startete 2013. Zu diesem Zeitpunkt gab es nur wenige Initiativen die Infrastrukturen für die medizinische Forschung bereitstellten. Vor allem die folgenden drei Initiativen waren bekannt:

1. Das *caBIG® Program* des *US National Cancer Institute* (caBIG, 2015) zertifizierte existierende Softwaresysteme nach speziellen Kriterien mit Bronze, Silber und Gold. Dieses Programm wurde 2015 eingestellt.

2. Das *National Cancer Informatics Program* (NCI, 2015) (inzwischen wurde die Seite umgezogen und es gibt eine upgedatete Version (NCI, 2021)) unterstützt Interoperabilität in der medizinischen Forschung, indem es Open Source Code für Software-Anwendungen für biomedizinische Informatik und semantische Infrastrukturen bereitstellt.

3. Zwei Projekte an der *Technologie- und Methodenplattform für die vernetzte medizinische Forschung e.V. (TMF e.V.)* (TMF e.V., 2021): Ein nationales *Metadaten Repository für*

klinische Studien und Register, und ein Projekt an der TMF, *das Anforderungen an Kohorten und Register-IT* definiert. Dieses Projekt startete 2012, (Lindoerfer & Mansmann, 2014b, 2015, 2017b)

Eine Checkliste für Register für seltene Erkrankungen wurde 2013 publiziert und soll hier auch erwähnt werden, diese Checkliste hatte keinen Einfluss auf diese Arbeit.

- Eine Checkliste für Register für seltene Erkrankungen, die 2013 von (Bellgard et al., 2013) publiziert wurde. Diese Checkliste umfasst die wichtigsten Punkte, ohne weiter auf Details einzugehen.

1.3.3 Neuere Initiativen

In den letzten Jahren nahm die Diskussion über Patientenregister an Fahrt auf. Im Rahmen eines von der Europäischen Union (EU) finanzierten Projekts, publizierten Zaletel & Kralj (2015) einen umfassenden Beitrag, mit dem Titel: „Methodological guidelines and recommendations for efficient and rational governance of patient registries“, (Zaletel & Kralj, 2015). In einem weiteren Beitrag: „Patient Registries: An Underused Resource for Medicines Evaluation“, (McGettigan et al., 2019) werden Patientenregister als bisher zu wenig genutzte Ressource für die medizinische Forschung beschrieben.

Die European Medicines Agency (EMA), (European Medicines Agency, 2021) publizierte unter der Section *Patient registries*, 2015 auf ihrer Webpage eine „Initiative for patient registries“ (European Medicines Agency, 2015), es fanden mehrere Workshops statt, unter anderem 2018 ein Workshop (European Medicines Agency, 2018), und ein *Discussion Paper*, mit dem Titel: „Use of patient disease registries for regulatory purposes - methodological and operational considerations“ (European Medicines Agency, The Cross-Committee Task Force on Patient Registries, 2018). Im Jahr 2020 wurde eine „Guideline on registry-based studies“ (European Medicines Agency, 2020) veröffentlicht. Die Veröffentlichungen konzentrieren sich auf das Registerprotokoll, die Datensammlung, Datenanalyse und die Berichterstattung. Softwarebezogene Themen werden nur am Rande behandelt.

1.4 Erstellung der CIPROS Checkliste

Um Wissenschaftlern künftig eine Entscheidungshilfe zur Auswahl eines geeigneten Softwaresystems für ihr Registerprojekt zur Verfügung zu stellen, wurde entschieden eine Checkliste mit Items für Patientenregister Softwaresysteme zu erstellen, die **CIPROS Checkliste** (Lindoerfer & Mansmann, 2015). Der Name leitet sich von den englischen Begriffen wie folgt ab: **C**hecklist

with **I**tems for **P**atient **R**egistry S**O**ftware **S**ystems. Dies soll nicht nur eine Checkliste, basierend auf eigenen Erfahrungen sein, sondern eine evidenz-basierte Checkliste. Dazu wurde eine systematische Literatursuche in PubMed, der *National Library of Medicine*, (PubMed, 2021a) durchgeführt. Dadurch ist gewährleistet, dass die Begriffe aus realen Systemen und Projekten abgeleitet wurden, die in qualitativ hochwertigen Journals publiziert wurden.

Wie Altman et al. (Altman et al., 2001) die mit dem *CONsolidated Standards Of Reporting Trials (CONSORT) Statement* eine Checkliste zur Berichterstattung in klinischen Studien erstellt haben, mit dem Ziel, die Berichterstattung klinischer Studien zu verbessern, soll mit der CIPROS Checkliste eine Checkliste bereitgestellt werden, welche die Wissenschaftler unterstützt, für ein bestimmtes Patientenregisterprojekt ein geeignetes Softwaresystem zu finden, oder ein neues Softwaresystem zu entwickeln. Gleichzeitig kann eine solche Checkliste auch dazu beitragen, die Berichterstattung über den Einsatz von Patientenregister Softwaresystemen in realen Projekten zu verbessern, (Lindoerfer & Mansmann, 2014a,b, 2015, 2016, 2017b).

1.4.1 Methoden zur Erstellung der CIPROS Checkliste

Die CIPROS Checkliste wurde in drei Schritten erstellt, die im Folgenden kurz beschrieben werden:

1. Systematische Literatursuche
2. Qualitative Inhaltsanalyse
3. Erstellung der CIPROS Checkliste

1.4.1.1 Systematische Literatursuche

Um die CIPROS Checkliste zu erstellen wurde eine systematische Literatursuche in *PubMed*, der *National Library of Health*, (PubMed, 2021a) durchgeführt. Am 17. Januar 2014 wurde folgende Suche in PubMed durchgeführt:

"(registry or registries) AND (eCRF or EDC or CDMS or CTMS or web) AND (software or open-source or open source or Java)".

Am 1. Dezember 2014 und am 15. Februar 2015 wurden Updates mit der gleichen Suchanfrage in PubMed durchgeführt. Die Anfrage war motiviert durch die Begriffe: **electronic Case Report Form (eCRF), Electronic Data Capture (EDC), Clinical Data Management System (CDMS), Clinical Trial Management System (CTMS)** und **web**-basierten Systemen, ebenso wie **Software** Systeme oder **open-source** Systeme und **Java**, weil dies in einigen einschägigen Titeln von Publikationen über Software Systeme vorkam, (Lindoerfer & Mansmann, 2014a,b, 2015, 2016, 2017b,a).

1.4.1.2 Qualitative Inhaltsanalyse

Die gefundenen Publikationen wurden mit einer qualitativen Inhaltsanalyse nach Mayring (Mayring, Philipp, 2000) analysiert. Die Abstracts und Publikationen wurden gesichtet.

Wenn die Publikation eine **Softwaresystembeschreibung** enthielt und die **Sprache der Publikation Englisch oder Deutsch** war, wurde die ganze Publikation gelesen. Wichtige Begriffe, die softwarebezogene Features beschreiben, wurden in einer Liste notiert. Aus dem Material wurden Kategorien definiert. Dabei wurde in einem ersten Schritt das Ablaufmodell der induktiven Kategorienbildung nach Mayring, Philipp (2000) angewendet, siehe Abbildung 1.1. Die gefundenen Publikationen wurden gelesen, nach ca. 10 - 50 % des Materials wurden die Kategorien überarbeitet, dann wurde der Rest der Publikationen gelesen, (Lindoerfer & Mansmann, 2016, 2017b,a).

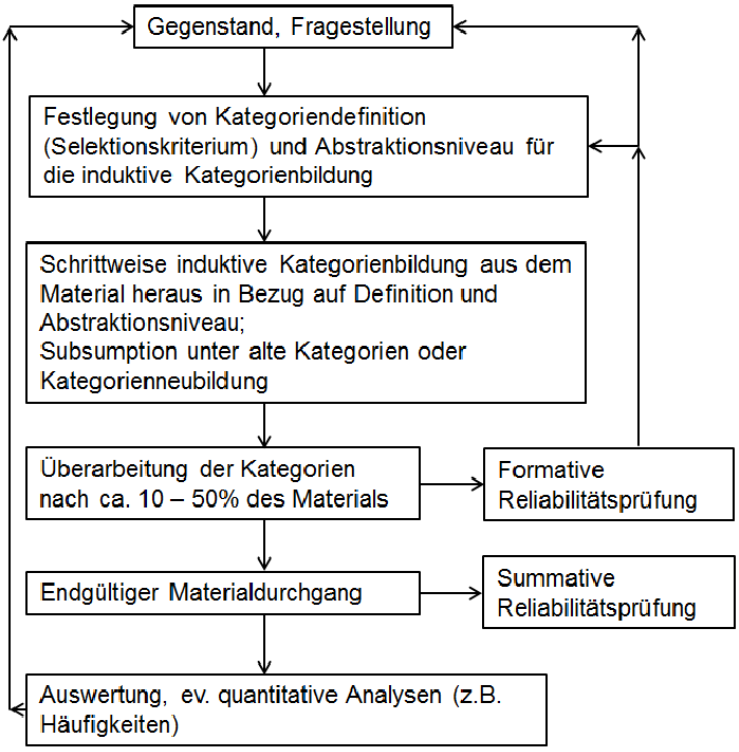


Abbildung 1.1: Ablaufmodell induktiver Kategorienbildung; adaptiert von Mayring, Philipp (2000) (CC-BY-NC), vgl. Lindoerfer & Mansmann (2016, S. 650), Lindoerfer & Mansmann (2017b, S. 149).

1.4.1.3 Erstellung der CIPROS Checkliste

In einem zweiten Schritt wurde die Kategorienliste mit dem Ablaufmodell der deduktiven Kategorienanwendung nach Mayring, Philipp (2000) überarbeitet, siehe Abbildung 1.2.

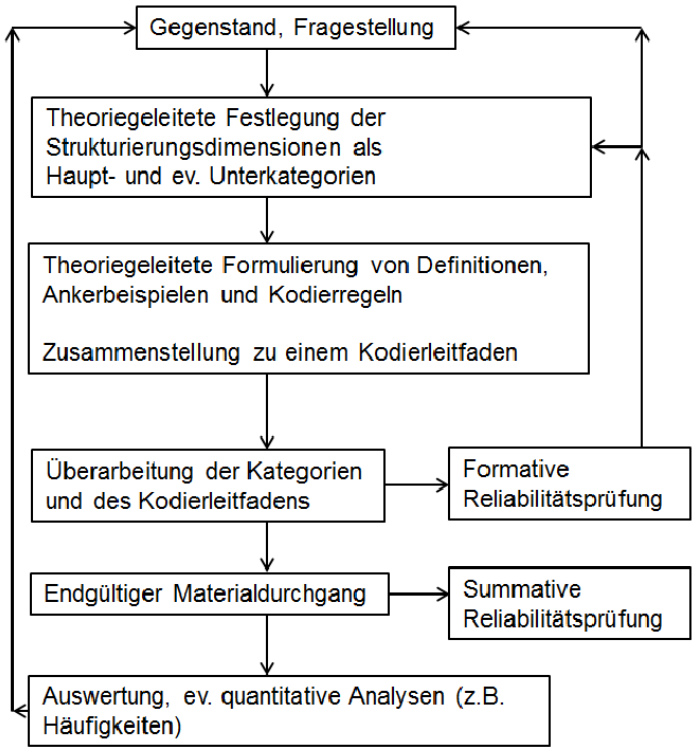


Abbildung 1.2: Ablaufmodell deduktiver Kategorienanwendung; adaptiert von Mayring, Philipp (2000) (CC-BY-NC), vgl. Lindoerfer & Mansmann (2016, S. 650), Lindoerfer & Mansmann (2017b, S. 149).

Falls erforderlich wurden neue Haupt- oder Unterkategorien gebildet. Bestehende Kategorien wurden nach inhaltlicher Ähnlichkeit sortiert, identische Kategorien wurden zusammengefasst. Die Begriffe wurden in die neuen Haupt- und Unterkategorien sortiert. Wenn ein neues Feature gefunden wurde, wurde ein neues Item erstellt und in der entsprechenden Kategorie der Liste beigefügt. Für jedes neue Feature wurde ein neues Item erstellt, unabhängig davon, wie oft das Feature in der Literatur gefunden wurde.

Zum Schluss wurden noch wichtige Items der Checkliste angefügt die in keiner der Publikationen gefunden wurden, aber aufgrund eigener Erfahrungen als wichtig erachtet wurden.

Dieses Vorgehen unterstreicht den evidenz-basierten Ansatz. Die beiden Methoden wurden kombiniert: Der Erfahrungsschatz der publizierten Literatur, kombiniert mit den eigenen Erfahrungen, (Lindoerfer & Mansmann, 2016, 2017b).

1.4.2 Ergebnisse der Erstellung der CIPROS Checkliste

Die CIPROS Checkliste basiert auf einer upgedateten Literatursuche. Die erste Suche, die am 17. Januar 2014 durchgeführt wurde, lieferte 132 Ergebnisse. Basierend auf diesen Ergebnissen wurde eine erste Version der CIPROS Checkliste erstellt. Aus den 132 Ergebnissen wurden 42 Publikationen mit einer Systembeschreibung selektiert. Diese 42 Publikationen wurden mit einer qualitativen Inhaltsanalyse nach Mayring, (Mayring, Philipp, 2000) analysiert. Es wurden 67 Items extrahiert. Diese erste Version der CIPROS Checkliste enthielt 67 Items die in zehn Kapitel eingeteilt waren. Aufgrund dieser Version der CIPROS Checkliste wurde auf der MIE 2014 in Istanbul ein Beitrag auf Englisch als Vortrag präsentiert und in (Lindoerfer & Mansmann, 2014b) publiziert. Auf der GMDS 2014 in Göttingen wurde ein Beitrag auf Deutsch als Vortrag präsentiert und publiziert (Lindoerfer & Mansmann, 2014a).

Der GMDS 2014 Beitrag wurde ausgewählt um eine erweiterte Publikation in *Methods of Information in Medicine* zu publizieren, dem offiziellen Journal der GMDS.

Am 1. Dezember 2014 wurde ein Update der Literatursuche mit denselben Suchkriterien in PubMed durchgeführt. Diese Suche lieferte 156 Ergebnisse. Es wurden 24 zusätzliche Publikationen gefunden, die in der ersten Suche noch nicht vorhanden waren. Von diesen 156 Publikationen enthielten 64 eine Systembeschreibung. Diese 64 Publikationen wurden mit einer qualitativen Inhaltsanalyse nach Mayring, (Mayring, Philipp, 2000) analysiert. Verglichen mit der vorherigen Analyse wurde ein zusätzliches Item identifiziert. Vier weitere Items wurden, basierend auf eigenen Erfahrungen, zu der Liste hinzugefügt. Diese Items wurden nicht in der Literatur gefunden, wurden aber, aufgrund eigener Erfahrungen, als wichtig erachtet.

Die Liste wurde überarbeitet und teilweise neu sortiert. Im Folgenden wird die Entwicklung der Kategorien kurz skizziert.

Die Kategorie **Software Architecture** wurde kreiert um die Items *System Architecture*, *Platform Independence* und *Open Source* zu beinhalten. Diese Kategorie hat die Nummer eins, weil sie Items beinhaltet, über die ganz am Anfang eines Projekts entschieden werden muss.

Development und **Training** bildeten zuvor eine Kategorie, aber das stellte sich als unpraktisch heraus. Deshalb wurden sie getrennt und es wurden daraus zwei Kategorien. Die Kategorie **Development** hat die Nummer zwei, weil dieser Schritt auch am Anfang eines Projekts steht. Diese Kategorie enthält Items, die während der Entwicklung eines neuen Patientenregisterprojekts wichtig sind.

Die Hauptkategorie Nummer drei, **Interfaces and Interoperability** enthält technische Features und Systemschnittstellen, die für die Interoperabilität wichtig sind. In dieser Kategorie gibt es die Unterkategorien **End-User Interfaces**, **Programming Interface** und **Interfaces to other Systems.**

Kategorie Nummer vier, **Interoperability and Semantics & Standardization** fokussiert auf die Interoperabilität der Daten. Diese Kategorie beinhaltet Items welche standardisierte Datensammlung und Datenaustausch unterstützen.

Kategorie Nummer fünf **Internationality**, enthält nur ein Item, *Multilingualism*. Trotzdem ist es als Kategorie gesetzt, da dieses Feature als wichtig erachtet wurde.

Kategorie Nummer sechs, **Data Management, Data Quality and Usability** enthält alle Items die für die Datenerhebung, das Datenmanagement, die Datenvalidierung sowie das Querymanagement wichtig sind. Es sind auch Items enthalten, die dazu beitragen, die Datenqualität zu managen, wie *Data Query Flags* und *Plausibility Flags*. Das Item *Software Ergonomics* bezieht sich auf die Handhabbarkeit und ist auch in dieser Kategorie enthalten.

Kategorie Nummer sieben, **Data Analysis** enthält Items, die für die Datenanalyse wichtig sind, zum Beispiel um schnelle Analysen der Daten zu erstellen um einen Überblick zu erhalten, welche Daten aktuell im Register enthalten sind. Das Item *Risk Analysis* ist ebenfalls in dieser Kategorie enthalten. Es dient dazu, Risikopatienten frühzeitig zu identifizieren.

Kategorie Nummer acht, **Security Aspects**, enthält Items die für den Datenschutz wichtig sind, zum Beispiel, dass nur autorisierte Benutzer zugreifen dürfen und dass alle Zugriffe in einem *Audit trail* geloggt werden.

Kategorie Nummer neun, **Privacy**, enthält Items die garantieren sollen, dass die gesammelten Daten gegen unberechtigten Zugriff geschützt sind. Das Item *Data Protection Concept* war in der früheren Version in der Kategorie **Organizational**, dies stellte sich aber als unpassend heraus. Als das wichtige Item *Double Pseudonymization* hinzugefügt wurde, wurde die Kategorie **Privacy** erstellt und das Item *Data Protection Concept* wurde darunter eingeordnet.

Die Kategorie zehn, **General Features**, enthält allgemeine Items, die nicht in andere Kategorien passen, zum Beispiel das Item *Costs*, das hervorhebt, dass die Kosten eines Registers bei der Auswahl des Patientenregister Softwaresystems berücksichtigt werden müssen.

Kategorie elf, **Organizational**, enhält regulatorische Items.

Kategorie zwölf, **Training**, enthält Items die wichtig sind, um die Personen die mit dem Register arbeiten zu schulen. Es sind die Items *User Manuals*, *User Training*, *User Feedback* und *Online Help* enthalten.

Die CIPROS Checkliste enthält nun 72 Items die in zwölf Kapiteln angeordnet sind. Diese Version wurde in *Methods of Information in Medicine*, dem offiziellen Journal der GMDS publiziert, (Lindoerfer & Mansmann, 2015). Diese Publikation ist jetzt Publikation I dieser Dissertation.

Am 15. Februar 2015 wurde nochmal eine Suche mit denselben Suchkriterien durchgeführt. Diese Suche lieferte 157 Ergebnisse. Verglichen mit der Suche am 1. Dezember 2014 wurde eine weitere Publikation gefunden. Es wurden keine weiteren Items identifiziert. Diese Suche war

Grundlage für die Publikation in *Journal of Biomedical Infomatics*, (Lindoerfer & Mansmann, 2017b). Diese Publikation ist jetzt Publikation II dieser Dissertation.

Die Erstellung der CIPROS Checkliste wurde in drei Schritten durchgeführt.

1. Zuerst wurde eine systematische Literatursuche durchgeführt. Dies geschah in wiederholten Updates, wie oben beschrieben. Die Suche am 15. Februar 2015 lieferte 157 Ergebnisse. Die Titel und Abstracts der Publikationen wurden gelesen und, wenn vielversprechend, auch die kompletten Publikationen. Wenn die Publikation eine Systembeschreibung, oder zumindest eine kurze Systembeschreibung enthielten und die Sprache der Publikation Englisch oder Deutsch war, wurden sie als relevant betrachtet, (Lindoerfer & Mansmann, 2015, vgl. S. 448), (Lindoerfer & Mansmann, 2017b, vgl. S. 150), (Lindoerfer & Mansmann, 2017a, vgl. S. 496). In der Suche vom 15. Februar 2015 wurden 64 Publikationen mit einer Systembeschreibung gefunden und für die weitere Analyse ausgewählt. Von den restlichen 93 Publikationen die ausgeschlossen wurden, enthielten 26 eine Beschreibung biologischer Informationssysteme, 24 handelten von Infrastrukturen für klinische oder andere medizinische Systeme, 40 Publikationen behandelten andere Aspekte und bei drei Publikationen war die Sprache nicht Deutsch oder Englisch. Der Auswahlschritt ist in einem "Preferred Reporting Items for Systematic Reviews and Meta-Analyses (PRISMA) Statement", (Moher et al., 2009) in Figure 1.3 dargestellt, (Lindoerfer & Mansmann, 2015, vgl. S. 448), (Lindoerfer & Mansmann, 2017b, vgl. S. 150f.).

2. Die gefundenen 64 Publikationen wurden mit einer qualitativen Inhaltsanalyse nach Mayring, (Mayring, Philipp, 2000) analysiert. Relevante Begriffe für die Beschreibung von Softwaresystemen wurden notiert und in eine Liste sortiert. Durch die qualitative Inhaltsanalyse wurden 68 Items gefunden, 4 Items wurden hinzugefügt, obwohl sie in keiner der Publikationen gefunden wurden. Es handelt sich um die Items 6.10, 6.12, 6.15 und 9.2. Sie wurden aufgrund eigener Erfahrungen als wichtig erachtet. Insgesamt wurden 72 Items identifiziert, basierend auf dem systematischen Literaturreview und eigenen Erfahrungen mit realen Projekten, (Lindoerfer & Mansmann, 2015, vgl. S. 448), (Lindoerfer & Mansmann, 2017b, vgl. S. 150f.), (Lindoerfer & Mansmann, 2017a, vgl. S. 497).

3. Inspiriert durch die caBIG® Kriterien wurde eine Checkliste erstellt die 72 Items enthält und in zwölf Systemkomponenten, funktionale Aspekte oder Design Schritte unterteilt ist. Die CIPROS Checkliste ist in Publikation I und als Appendix von Publikation II und als Appendix von Publikation III veröffentlicht. Im Appendix zu Publikation III werden die CIPROS Items vorgestellt, mit den gefundenen Beispielen aus der Literatur belegt und mit eigenen Erklärungen begründet.

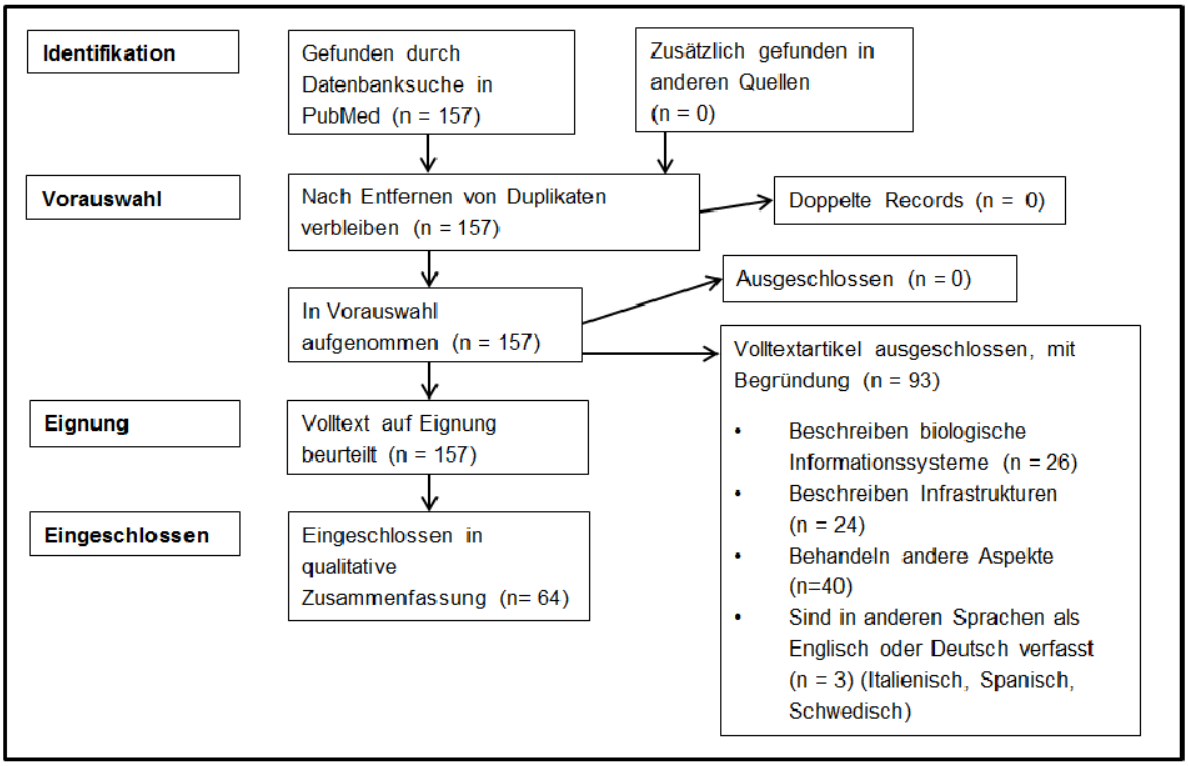


Abbildung 1.3: PRISMA Ablaufmodell, Beschreibung der Anzahl selektierter Publikationen; adaptiert von (Moher et al., 2009) (CC-BY), Lindoerfer & Mansmann (2016, vgl. S. 651), Lindoerfer & Mansmann (2017b, vgl. S. 151).

1.5 CIPROS im Software Engineering

1.5.1 Traditionelle Methoden zur Anforderungsermittlung

Traditionelle Methoden zur Anforderungsermittlung (Requirements Elicitation) im Software Engineering sind **Interviews** mit den betroffenen Personen (Stakeholdern) und Workshops wie **Focus Groups** oder **Expert Panels** mit den betroffenen Personen um so in Gesprächen die Anforderungen an ein neues System zu ermitteln (Lindoerfer & Mansmann, 2016, 2017b).

1.5.2 Templates zur Anforderungsermittlung in Softwareprojekten

Es gibt Templates zur Anforderungsermittlung in Softwareprojekten, dabei handelt es sich um **Software Requirements Specification (SRS) Templates**. Im Folgenden werden einige Templates kurz vorgestellt:

- **Der International Organization for Standardization (ISO)/International Electrotechnical Commission (IEC)/Institute of Electrical and Electronics Engineers (IEEE) 29148:2011 International Standard**, (IEEE STANDARD. 29148-2011, 2011): Der ISO/IEC/IEEE 29148:2011 International Standard (IEEE STANDARD. 29148-2011, 2011) beschreibt detailliert wie Anforderungen in einem SRS Dokument geschrieben werden sollen, welche Standards und Abkürzungen verwendet werden sollen, wie der Prozess der Anforderungsanalyse organisiert werden soll, etc. Es wird zwischen Stakeholder Requirements, die in der Stakeholder Requirements Specification (StRS) spezifiziert werden, System Requirements, die in der System Requirements Specification (SyRS) spezifiziert werden und Software Requirements, die in der SRS spezifiziert werden, unterschieden. Er enthält auch Requirements die allgemeines Management und Änderungsmanagement betreffen, (Lindoerfer & Mansmann, 2017b, vgl. S. 148f.)

- Das **SRS Template von Wiegers & Beatty, 2013**, (Wiegers & Beatty, 2013). Das kompakte SRS Template von Wiegers & Beatty (2013) strukturiert die Spezifikation der projektspezifischen Inhalte und Requirements, (Lindoerfer & Mansmann, 2017b, cf. S. 149).

- Das **SRS Template von McKinnon** (McKinnon, AD, 2005) wurde 2005 veröffentlicht. Das Dokument stellt eine Struktur für die Spezifikation von projektspezifischen Inhalten und Anforderungen, basierend auf dem IEEE830-1984 Standard, bereit. Das Dokument ist im Internet frei verfügbar, (McKinnon, AD, 2005), (Lindoerfer & Mansmann, 2017b, cf. p. 149).

- **Volere Requirements Specification Template** (Robertson, J. and Robertson S., 2015). Das Volere SRS Template wurde von James und Suzanne Robertson von 1995 bis 2015 entwickelt. Es handelt sich um ein umfangreiches Template zur Spezifikation von projektspezifischen Inhalten und Anforderungen, das auch Beispiele für die Spezifikation von projekt-spezifischen Requirements enthält. Es kann über die Volere Webpage gekauft werden, die Webpage wurde in der Zwischenzeit umgezogen (Robertson, S. and Robertson J., 2021). Bei dem Vergleich in Publikation II wurde die Version von 2015 verwendet, (Lindoerfer & Mansmann, 2017b, cf. p. 149).

- Das von Bruegge & Dutoit (2014) vorgestellte **Requirements Analysis Document (RAD)** ist eine kompakte Aufzählung von Aspekten die im Software Engineering Prozess wichtig sind. Dieses Dokument wird in Bruegge & Dutoit (2014) in einem Beispiel im Software Engineering verwendet.

1.5.3 Vergleich der vorhandenen Templates mit CIPROS

In Publikation II wurde ein Vergleich zwischen den Templates und der CIPROS Checkliste vorgenommen. Dabei wurde verglichen, ob die Items der CIPROS Checkliste in den SRS Templates vorkommen, ob es entsprechende Punkte in den SRS Templates gibt, die das entsprechende CIPROS Item abdecken, welche Punkte die Templates zusätzlich bereitstellen und welche Items nur die CIPROS Checkliste bereitstellt, (Lindoerfer & Mansmann, 2017b).

Appendix C von Publikation II gibt einen Überblick über die CIPROS Items und wie sie mit den Elementen der SRS Templates korrespondieren. Der ISO Standard und die Templates definieren allgemeine Requirements und wie die Requirements geschrieben werden sollen, während die CIPROS Checkliste eine Liste von Requirements bereitstellt, die in einem Patientenregister Softwaresystem implementiert werden können, deshalb kann dieses Matching nicht perfekt sein. Jedoch stellt die CIPROS Checkliste spezifische Items für Patientenregister Softwaresysteme bereit, die in den Templates nicht vorkommen, aber in der CIPROS Checkliste abgedeckt sind.

1.5.4 Software Engineering Vorgehensmodelle

Zur Planung und Realisierung großer Softwareprojekte werden für die Softwareentwicklung sogenannte Vorgehensmodelle eingesetzt.

Das erste Vorgehensmodell im Software Engineering war das **Wasserfallmodell**, das zuerst 1970 von Winston E. Royce publiziert wurde, (Royce, 1970). Das Wasserfallmodell definiert die nötigen Schritte: *System-Anforderungen, Software-Anforderungen, Analyse, Design, Implementierung, Test, Betrieb*, [Übersetzung d. Verfass.] (Royce, 1970) [2] in einer sequentiellen Reihenfolge. Das heißt, dass ein neuer Schritt erst dann starten kann, wenn der vorherige Schritt abgeschlossen ist. Dieses Model ist von den Ingenieursdisziplinen abgeleitet. Es stellte sich heraus, dass es für Software Projekte nicht so gut geeignet ist, da es in Software Projekten keinen Sinn macht, die Schritte so strikt voneinander zu trennen. Im Softwareengineering ist ein iterativer Ansatz passender. Das Wasserfallmodell wird wegen der strikten Trennung der Schritte heute als veraltet eingestuft. Die meisten Softwareprojekte verwenden einen iterativen Ansatz. Nichtsdestotrotz war das Wasserfallmodell das erste Modell und viele der späteren Modelle verwenden die gleichen Schritte nur ohne die strikte Trennung.

Das **V-Modell** ist eine Erweiterung des Wasserfallmodells, indem jeder Stufe aus der Planungsphase eine Testaktivität zugeordnet wird. Dadurch wird jeder Schritt aus der Planungsphase in der Testphase verifiziert und validiert. Genauso wie das Wasserfallmodell ist das V-Modell eine vereinfachte Abstraktion des Softwareentwicklungsprozesses. Diese Modelle suggerieren, dass, wenn ein Schritt beendet ist, die dazugehörigen Aktivitäten beendet sind und der nächste

[2] „System Requirements, Software Requirements, Analysis, Program Design, Coding, Testing, Operations“

Schritt beginnen kann. Doch diese strikte Trennung suggeriert, dass die Spezifikation der Anforderungen verlässlich und stabil ist. In der Praxis jedoch folgt die Softwareentwicklung nicht diesen idealisierten Modellen. Änderungen in einer späteren Phase erfordern häufig, dass man zu einer früheren Aktivität zurückkehrt, (Bruegge & Dutoit, 2014, vgl. 624-625).

Das **V-Modell XT**, (V-Modell XT, 2021) ist eine Erweiterung des V-Modells. Das V-Modell XT wird auch für das Projektmanagement verwendet und unterstützt Transparenz indem Rollen mit Ergebnissen verbunden werden. Das V-Modell XT ist besonders für sehr große Projekte geeignet. Das V-Modell XT wird vom Weit e.V. - Verein zur Weiterentwicklung des V-Modell® XT weiterentwickelt, (Weit e.V., 2021). Es gibt auch eine, auf die Bedürfnisse von Behörden, angepasste Weiterentwicklung, das V-Modell XT Bund, (VModellXTBund, 2020).

Eine risikobasierte Erweiterung des Wasserfallmodells stellt das **Spiralmodell** dar, das von BW Boehm 1986 (Boehm, 1986) publiziert wurde. Das Spiralmodel basiert auf den gleichen Aktivitäten wie das Wasserfallmodell, aber es fügt verschiedene Aktivitäten wie Risikomanagement, Wiederverwendbarkeit, und Prototyping zu jeder Aktivität hinzu. Dieses Model arrangiert die Schritte als Kreise. Jeder Kreis wird durch eine Risikoanalyse und einen Prototyp charakterisiert. Es fokussiert auch auf die Software Anforderungen und auf die Validierung der Anforderungen.

Es gibt noch weitere Modelle wie das **Sägezahnmodell (Sawtooth Model)** (Rowen, 1990) das versucht, das unterschiedliche Verständnis von Requirements von Entwicklern und Kunden zusammenzubringen.

Ein weiteres Model ist das **Haifischzahnmodell (Shark Tooth Model)** das ist eine Erweiterung des Sägezahnmodells. Es berücksichtigt auch Management Reviews und Demonstrationen, (Bruegge & Dutoit, 2000, vgl. S. 481-482).

Eine neuere Methode ist das **Agile Software Development**. Der große Unterschied zu den zuvor beschriebenen Modellen ist, dass im Agilen Software Development eine Funktion entwickelt wird, dann implementiert wird und dann getestet wird. Anschließend wird die nächste Funktion entwickelt, implementiert und getestet usw., bis alle Funktionen implementiert sind. Dann wird ein Prototyp entwickelt und dann können Änderungen gemacht werden. Diese Schritte werden als Kreise dargestellt. Der Kreis wird iterativ solange durchlaufen, bis alle Funktionen implementiert sind. Das *Agile Software Development* wurde zuerst nur für kleinere Projekte verwendet, heute existieren auch *Agile Modelle* für größere Projekte, (Sommerville, 2016, vgl. S. 72-100).

1.5.5 Integration von CIPROS in die vorhandenen Modelle

Die beschriebenen, generellen SRS Templates wie der ISO Standard, (IEEE STANDARD. 29148-2011, 2011) das SRS Template von (Wiegers & Beatty, 2013), das SRS Template von (McKinnon, AD, 2005), das Volere Template (Robertson, S. and Robertson J., 2021) und das RAD

(Bruegge & Dutoit, 2014) werden benutzt um die Software Requirements für ein neues Projekt zu spezifizieren.

Jedes der Softwareengineering Vorgehensmodelle enthält den Punkt der Anforderungsspezifikation (Requirements).

Die CIPROS Checkliste ist eine neue Methode um Anforderungen für ein geplantes Patientenregister Projekt zu spezifizieren. Jedes Item der CIPROS Checkliste wird überprüft, ob es für das neue Projekt nötig ist. Wenn ein Item benötigt wird, kann es in das Software Requirements Specification (SRS) Dokument übernommen werden und für das neue Projekt spezifiziert werden. So kann die evidenzbasierte CIPROS Checkliste den Requirements Engineering Prozess für ein neues Patientenregister Projekt beschleunigen und die Funktionen in das neue Patientenregister Projekt übernehmen, denn es gibt 72 Items in der CIPROS Checkliste, die in Betracht gezogen werden können. Da können einige Funktionen darunter sein, die ohne die CIPROS Checkliste nicht gefunden worden wären.

1.6 Praktischer Einsatz von CIPROS

Um für ein konkretes Projekt mit Hilfe der CIPROS Checkliste die Requirements zu identifizieren, geht man die CIPROS Checkliste durch und überprüft für jedes Item ob es für das neue Projekt nötig ist.

Wenn man ein neues System entwickeln will, liefert diese Überprüfung eine Liste von Features die für das neue System wichtig sind und implementiert werden müssen.

Wenn man bereits ein konkretes System hat, das man für das neue Projekt verwenden will, geht man die CIPROS Checkliste durch und überprüft jedes Item ob es in dem neuen System vorhanden ist.

Anschließend vergleicht man die beiden Überprüfungen. Dadurch erfährt man, wie geeignet das System ist, die Anforderungen des neuen Projekts zu erfüllen.

Durch die Verwendung der CIPROS Checkliste für das Requirements Engineering kann es auch sein, dass die Items dazu inspirieren neue Features zu finden, an die man vorher nicht gedacht hat, die aber für das neue Projekt wichtig sind.

So beschleunigt die CIPROS Checkliste das Requirements Engineering für Patientenregister Software Projekte.

Die CIPROS Checkliste beinhaltet auch eine ökonomische Komponente und kann helfen, die Entwicklungskosten eines neuen Patientenregister Software Projekts zu reduzieren. Die Verwendung der CIPROS Checkliste ist kostengünstiger als die Verwendung von *Interviews* oder *Workshops* wie *Expert Panels* und *Focus Groups*, bzw. sie macht diese Aktivitäten effizienter.

1.7 Zusammenfassung und Ausblick

1.7.1 Der Beitrag der CIPROS Checkliste

Hier wird kurz zusammengefasst was die Entwicklung der CIPROS Checkliste für den Einsatz von Patientenregister Softwaresystemen in der medizinischen Forschung beigetragen hat:

1. Die CIPROS Checkliste fasst wichtige Punkte der aktuellen Diskussion über die IT-Infrastruktur für Patientenregister Softwaresysteme der Medizinische Informatik (Medical Informatics) (MI) Literatur zusammen. Die Notwendigkeit der CIPROS Checkliste ist offensichtlich.
   (a) Die CIPROS Checkliste hilft Projektkoordinatoren die Requirements für ihre neuen Projekte zu identifizieren und geeignete Systeme dafür zu finden.
   (b) Die CIPROS Checkliste unterstützt Entwickler die Requirements für bestehende Systeme zu evaluieren.
   (c) Die CIPROS Checkliste unterstützt die Berichterstattung über Patientenregister Softwaresystembeschreibungen. Sie kann ein erster Schritt hin zu einem standardisierten Reporting über Patientenregister Softwaresysteme sein.
2. Das methodische Vorgehen bei der Erstellung der CIPROS Checkliste kann ein Vorbild sein, um ähnliche Listen für andere Zwecke zu kreieren.
   (a) Die Methoden die verwendet wurden um die CIPROS Checkliste zu erstellen, namentlich der Systematische Literaturreview, die Qualitative Inhaltsanalyse und die Erstellung der CIPROS Checkliste, können auf andere Gebiete transferiert werden um evidenzbasierte Checklisten zu erstellen.
   (b) Das evidenzbasierte Requirements Engineering kann als neue Requirements Engineering Methode in das Software Engineering integriert werden.
3. Die dreistufige Methode der Evaluation kann auch in andere Gebiete übernommen werden.
   (a) Die Projektevaluation mit der Checkliste um die Projekt Requirements zu identifizieren.
   (b) Die Systembewertung mit der Checkliste, um festzustellen, welche Funktionen das System bereitstellt.
   (c) Die Kombination der System- und Projektbewertung um die Übereinstimmung zu demonstrieren.

Diese dreistufige Evaluation kann verwendet werden, um andere Patientenregister Softwaresysteme zu evaluieren. Dieser Typ der Evaluation kann auch auf andere Gebiete übertragen werden. Wenn Checklisten für andere Gebiete erstellt werden, kann diese dreistufige Evaluation helfen, das geeignete System für ein Projekt zu finden.

1.7.2 Der Einfluss anderer Initiativen auf CIPROS

Es sollen noch zwei Initiativen genannt werden, die in der jüngeren Zeit größeren Einfluss auf die medizinische Forschung hatten:

1. Die FAIR Prinzipien (Wilkinson et al., 2016) beschreiben vier fundamentale Prinzipien welche die Produzenten und Autoren von Daten beachten sollen **F**indability, **A**ccessibility **I**nteroperability und **R**eusability, also, dass die Forschungsdaten *auffindbar, zugreifbar, interpretierbar* und *lesbar* sein sollen. Die CIPROS Checkliste enthält Items welche die FAIR Prinzipien unterstützen. So definieren „Item 4.3, Metadata“ und „Item 4.4, Vocabularies“ , dass ontologie-basierte, standardisierte Metadaten und Vokabeln verwendet werden sollen. Diese Items beziehen sich auf den geforderten Punkt *interpretierbar* der FAIR Prinzipien. Die CIPROS Checkliste bezieht sich auf die Auswahl und Evaluation von Softwaresystemen für Patientenregisterprojekte und Aspekte, die zu Beginn und während der Projektlaufzeit wichtig sind. Das Item 3.10, welches die Bereitstellung eines Application Programming Interface (API) beschreibt, und das Item 7.4, das die Möglichkeit, farbige Graphen in Realtime zu erstellen beschreibt, beziehen sich auf den Punkt *accessibility* der FAIR Prinzipien. Diese Items beschreiben, dass die Daten zugreifbar sein sollen. Diese Beispiele zeigen, dass die CIPROS Checkliste die FAIR Prinzipien unterstützt.

2. Im Jahr 2018 trat die Europäische Datenschutz-Grundverordnung (DSGVO) (DSGVO, 2018; GDPR, 2018) in Kraft, die einen enormen Einfluss auf die Planung und Implementierung von medizinischen Forschungsprojekten hat. Die CIPROS Items 11.1, das sich auf die Regeltreue zu bestehenden Gesetzen bezieht, und Item 11.4, welches auf die Datenschutzgesetze verweist, decken diese Punkte ab. Die CIPROS Checkliste beinhaltet diese Punkte bereits und muss deswegen nicht erweitert werden.

1.7.3 Der Einfluss von CIPROS auf andere Projekte

Die CIPROS Checkliste stößt auf reges Interesse in der Medizininformatik Community. Ich erhalte immer wieder Anfragen von Interessierten, die die CIPROS Checkliste für Ihre Projekte einsetzen wollen.

Die drei Publikationen die dieser Dissertation zugrunde liegen, wurden bereits mehrfach von anderen Autoren zitiert. Für die Ermittlung der Zitate habe ich auf PubMed (PubMed, 2021b) und Google Scholar, (GoogleScholar, 2021) zurückgegriffen.

Publikation I wurde bereits neun Mal zitiert, (Bellgard et al., 2017a,b; Bluhmki et al., 2017; Kodra et al., 2017; Napier et al., 2017a,b; Mojarrab et al., 2017; Osborne et al., 2018; Zakerabasali et al., 2020).

Publikation II wurde in fünf Publikationen (Osborne et al., 2018; Dar et al., 2018; Shanbehzadeh et al., 2020; Zakerabasali et al., 2020; Cristancho Triana et al., 2021) und in einer spanischen Master Thesis (Ordoñez Calero, H.D., 2018) zitiert.

Publikation III wurde von Osborne et al. (2018) zitiert, die die Ergebnisse dieser Publikation verwendet haben um ihr Register aufzubauen.

Der Konferenzbeitrag (Lindoerfer & Mansmann, 2014b) mit der ersten Version der CIPROS Checkliste wurde von Kodra et al. (2016) und Osborne et al. (2018) zitiert.

1.7.4 Ausblick

Die aktuelle Version der CIPROS Checkliste wurde 2015 entwickelt. Für die Zukunft ist ein Update der CIPROS Checkliste und eine Miteinbeziehung bereits bestehender Systeme und deren Funktionen geplant.

Kapitel 2

Beiträge zu den Publikationen

Diese kumulative Dissertation basiert auf den beiden im Folgenden aufgeführten Publikationen, die entsprechend als Publikation I und Publikation II bezeichnet werden. Die Arbeit wird ergänzt und abgerundet durch die Publikation in Anhang A die als Publikation III bezeichnet wird.

Patientenregister werden in der medizinischen Forschung häufig eingesetzt um Antworten auf bestimmte Fragestellungen zu finden. Zur Durchführung dieser forschungsorientierten Patientenregister werden heutzutage zunehmend webbasierte Softwaresysteme verwendet. Diese Softwaresysteme dienen zur Datenerhebung und stellen Werkzeuge zum Datenmanagement und zur Qualitätssicherung der Daten zur Verfügung. Teilweise sind auch einfache Analysewerkzeuge zu einer ersten Datenanalyse vorhanden. Doch wie kann man wissen ob ein bestimmtes Patientenregister Softwaresystem die Anforderungen des speziellen Patientenregisterprojekts erfüllt?

Ziel dieser Arbeit ist die Entwicklung einer evidenzbasierten Checkliste mit der überprüft werden kann, inwieweit ein Patientenregister Softwaresystem geeignet ist, die speziellen Anforderungen für ein bestimmtes Patientenregister Projekt zu erfüllen. Dazu wurde eine systematische Literaturrecherche in PubMed durchgeführt. Die gefundenen Publikationen wurden mit einer qualitativen Inhaltsanalyse nach Mayring, Philipp (2000) analysiert. Die gefundenen Items wurden notiert und in eine Liste sortiert. Es wurden Haupt-und Unterkategorien gebildet. Schließlich wurde die CIPROS Checkliste erstellt.

2.1 Beitrag zu Publikation I - Methods of Information in Medicine, 2015

Lindoerfer, D. and Mansmann, U. (2015). A Comprehensive Assessment Tool for Patient Registry Software Systems: The CIPROS Checklist. *Methods Inf Med,* 54(5):447-454.

Publikation I präsentiert und beschreibt die CIPROS Checkliste. Der Name leitet sich von den genannten englischen Wörtern wie folgt ab: **C**hecklist with **I**tems for **P**atient **R**egistry S**O**ftware **S**ystems (CIPROS). Die CIPROS Checkliste ist eine evidenzbasierte Checkliste mit deren Hilfe festgestellt werden kann, inwieweit ein Patientenregister Softwaresystem geeignet ist, die Anforderungen eines Patientenregister Projekts zu erfüllen. Zur Erstellung der CIPROS Checkliste wurde eine systematische Literaturrecherche in PubMed durchgeführt um Publikationen über Patientenregister Projekte mit Softwaresystembeschreibungen zu erhalten. Aus den gefundenen Papers wurden mit einer qualitativen Inhaltsanalyse nach Mayring (Mayring, Philipp, 2000) relevante Items extrahiert. Diese Items wurden nach den Methoden von Mayring, Philipp (2000) analysiert und in der CIPROS Checkliste angeordnet.

Die CIPROS Checkliste kann zur Bewertung von Patientenregister Softwaresystemen und zur Bewertung der Anforderungen von Patientenregister Projekten eingesetzt werden. Diese beiden Bewertungen müssen anschließend gegenübergestellt werden um zu überprüfen inwieweit ein Patientenregister Softwaresystem geeignet ist um die Anforderungen eines Patientenregister Projektes zu erfüllen.

Als Vorlage für die Erstellung der CIPROS Checkliste diente die Arbeit von Altman et al. (2001), in der die Autoren eine Checkliste für die Berichterstattung von klinischen Studien erstellten.

Publikation I beschreibt, nach einer Einführung in die Problematik, die Literatursuche, die Qualitative Inhaltsanalyse die Erstellung der Liste und die resultierende CIPROS Checkliste. Die Publikation I wird mit einer ausführlichen Diskussion der Problematik und einer kurzen Zusammenfassung beschlossen.

Ich bin Erstautorin dieser Publikation. Meine Arbeit bei dieser Publikation war die Formulierung der Suchstrategie, die Konzeption, Formulierung und Durchführung der Literaturrecherche, die Analyse der gefundenen Publikationen und die Erstellung der CIPROS Checkliste.

Bei der Erstellung der Publikation habe ich diese Schritte beschrieben. Ebenso habe ich die Recherchen zu bereits bestehendem Material durchgeführt und in der Einleitung beschrieben und auch die Einordnung der CIPROS Checkliste in der Diskussion dargestellt.

2.2 Beitrag zu Publikation II - Journal of Biomedical Informatics, 2017

Lindoerfer, D. and Mansmann, U. (2017). Enhancing Requirements Engineering for Patient Registry Software Systems with Evidence-based Components. *J Biomed Inform,* 71:147-153.

Publikation II beschreibt die Hintergründe die zur Erstellung der CIPROS Checkliste geführt haben. Zuerst wird beschrieben, wie sich die CIPROS Checkliste in Softwareerstellungsmodelle einordnen lässt. Danach wird ein Vergleich mit anderen Templates zur Anforderungsanalyse durchgeführt. Es wird ausführlich die Erstellung der CIPROS Checkliste beschrieben. Anschließend wird die resultierende CIPROS Checkliste beschrieben. Danach wird beschrieben, wie die CIPROS Checkliste im Softwareengineering eingesetzt werden kann um die erforderlichen Requirements zu identifizieren.

Ich bin ist Erstautorin dieser Publikation. Ich hatte die Idee zu dieser Publikation und habe auch sämtliche Recherchen zu Softwareerstellungsmodellen und anderen Anforderungsanalysetemplates selbständig durchgeführt. Die Vergleiche der CIPROS Checkliste mit den gefundenen anderen Anforderungsanalysetemplates habe ich selbständig durchgeführt und beschrieben. Weiterhin habe ich beschrieben, wie sich die CIPROS Checkliste in die Softwareerstellungsmodelle einordnen lässt, um den Softwareerstellungsprozess zu beschleunigen.

2.3 Beitrag zu Publikation III (Anhang A) - Data in Brief, 2017

Lindoerfer, D. and Mansmann, U. (2017). Data for the elaboration of the CIPROS checklist with items for a patient registry software system: Examples and explanations. *Data Brief*, 14:494-497.

Publikation III entstand aus Publikation II, auf Vorschlag der Editoren des Journals. Publikation III beschreibt die Textzitate aus den gefundenen Publikationen aus denen die Items der CIPROS Checkliste kreiert wurden und ergänzt diese durch eigene Erklärungen. Die Publikation ist sehr kurz, sie enthält nur eine Beschreibung der Daten. Die Resultate, die Zitate aus den Publikationen aus denen die Items entstanden sind, sowie die Erklärungen zu den Items sind im Appendix der Publikation enthalten, der auch Teil dieser Arbeit ist.

Die Publikation wurde von mir erstellt, unter der Supervision von Ulrich Mansmann.

Auch der Appendix mit den Textzitaten und den Erklärungen wurde von mir erstellt.

Kapitel 3

Die Publikationen im Original

3.1 Publikation 1 - Methods of Information in Medicine, 2015

Lindoerfer, D. and Mansmann, U. (2015). A Comprehensive Assessment Tool for Patient Registry Software Systems: The CIPROS Checklist. *Methods Inf Med,* 54(5):447-454.

DOI: 10.3414/ME14-02-0026

https://www.thieme-connect.com/products/ejournals/abstract/10.3414/ME14-02-0026

A Comprehensive Assessment Tool for Patient Registry Software Systems: The CIPROS Checklist*

D. Lindoerfer; U. Mansmann

Ludwig-Maximilians-University Munich, Institute for Medical Information Processing, Biometry and Epidemiology (IBE), Munich, Germany

Keywords
Checklist, patient registry software system, technical requirements

Summary

Background: Patient registries are an important instrument in medical research. Often their structure is complex and their implementation uses composite software systems to meet the wide spectrum of challenges.

Objectives: For the implementation of a registry, there is a wide range of commercial, open source, and self-developed systems available and a minimal standard for the critical appraisal of their architecture is needed.

Methods: We performed a systematic review of the literature to define a catalogue of relevant criteria to construct a minimal appraisal standard.

Results: The CIPROS list is developed based on 64 papers which were found by our systematic review. The list covers twelve sections and contains 72 items.

Conclusions: The CIPROS list supports developers to assess requirements on existing systems and strengthens the reporting of patient registry software system descriptions. It can be a first step to create standards for patient registry software system assessments.

Correspondence to:
Doris Lindoerfer
Institute for Medical Information Processing, Biometry and Epidemiology (IBE)
Ludwig-Maximilians-University Munich
Marchioninistraße 15
81377 Munich
Germany
E-mail: lindoerf@ibe.med.uni-muenchen.de

Methods Inf Med 2015; 54: 447–454
http://dx.doi.org/10.3414/ME14-02-0026
received: December 15, 2014
accepted: July 24, 2015
epub ahead of print: September 30, 2015

* Supplementary online material published on our website www.methods-online.com

1. Introduction

A registry collects information about individuals, usually focused around a specific diagnosis or condition. Registries can provide health care professionals and researchers with first-hand information about persons with certain health conditions, both individually and as a group, and over time, to increase understanding of that condition. The number of registries is constantly increasing and the use of medical registries has gained importance over the last years (Mathis-Edenhofer et al. [1], Anazawa et al. [2]). There are medical registries for different purposes: public health, epidemiology, quality control, clinical registries, disease specific registries, etc. and they can be regarded for different aspects: interoperability, data quality, security, protection of data privacy, etc. (Register in der Medizin [3]).

In this work we focus at research oriented Patient Registries according the definition of Glicklich et al. [4]. Such registries document the clinical phenotypes which are essential for

modern biomedical research. Arts et al. [5] clearly state that the value of a registry strongly depends on the quality of the data contained in it. Solomon et al. [6] make clear – already more than 20 years ago – that the successful implementation and use of a registry depends on a thoroughly and accurate planning and construction of a suitable IT infrastructure. Therefore, the IT infrastructure depends on the definition of the registry because the definition determines how to plan and run the registry. Furthermore, the focus should also be the legal and ethical aspects as well as the strategies of patient recruitment (Glicklich et al. [4]). Müller et al. [7] summarized these points in a checklist for Registries for Health Services Research. These issues are linked to IT infrastructure, but IT infrastructure was never explicitly discussed in these contributions.

The IT infrastructure of a registry may be quite challenging. It consists beside the central database on a series of interfaces and procedures to guarantee the cooperation of partners involved on a high quality level. For example, the German Registry for endoprosthesis (http://www.eprd.de/) involves producers of endoprosthesis, health insurance companies, and a large number of hospitals. The creation of consistent patient histories from all these sources needs an IT system which supports the communication as well as a complex verification system in order to produce reliable information on the quality of endoprostheses used in the German health care setting.

In spite of the availability of commercial and open source software systems, many institutions develop own solutions which fit their needs and they are flexible to address changing requirements. Therefore, guidance is needed to support assessment and selection of an existing, or to plan the development of a new registry software system.

Important functionalities of a patient registry software system should be assorted and considered before making the decision for a system of a new registry project. Critical issues are minimal requirements to the representation of the system's complexity. How they should be defined in a harmonizing process? What is an objective and reliable basis for the assessment of corresponding applications? On what to base funding decisions for such complex systems?

2. Objectives

To this end, we developed a **c**hecklist with **i**tems for a **p**atient **r**egistry s**o**ftware **s**ystem (CIPROS) which is inspired by the idea of a reporting checklist as first introduced by Altman et al. [8]. The checklist is to be derived from a qualitative systematic review of the literature and own experiences in this field (Lindörfer et al. [9]).

Especially three initiatives with the aim to support clinical research and interoperability in providing adequate software systems and semantic infrastructure inspired our work: 1) The caBIG® program from the National Cancer Institute [10] certified existing software systems, according special criteria with bronze, silver, and gold. 2) The recent National Cancer Informatics Program [11] aims to support interoperability in Medical Research in providing open source code for biomedical informatics software applications and in providing semantic infrastructures. 3) A National Metadata Repository for Clinical Studies and Registries is provided by "Technologie- und Methodenplattform für die vernetzte medizinische Forschung e.V. " (TMF e.V.) [12]. There is also an ongoing project at TMF e.V. [12] which defines common requirements for cohorts and registry IT.

3. Methods

This paper is based on the update of an earlier qualitative systematic literature review. Parts of this earlier version were presented at MIE 2014 – XXV Conference of the European Federation for Medical Informatics, August 31–September 3, 2014 in Istanbul, Turkey (Lindoerfer et al. [13]) and at the 59th GMDS-Jahrestagung, September 7–10, 2014 in Göttingen, Germany (Lindoerfer et al. [14]).

Our Systematic Review identified papers with patient registry software system descriptions. In order to get exhaustive results on papers containing patient registry software system descriptions the following search in Pubmed was applied: *"(registry or registries) AND (eCRF or EDC or CDMS or CTMS or web) AND (software or open-source or open source or Java)".*

We performed an updated search with the same criteria in Pubmed for this paper at December 1, 2014. The former search and the update resulted in 156 findings. We identified additional 24 papers which have not been available during the first review. We classified the total of 156 papers according to their content by reading title and the abstract (using the criteria of [13] and [14]).

A total of 64 provided a system description or at least a short system description (the update added 22 new papers to the discussion). Out of the remaining 92 papers, a total of 26 were about biological information systems, 23 were about infrastructures for clinical or other medical systems, 40 deal with other aspects and had no system description. Three papers were written in languages other than English or German (Spanish, Italian, Swedish).

A qualitative analysis was performed on the 64 papers containing a system description by searching them for relevant software specific features. We tried to classify the identified software specific features in system components, functional aspects, or design steps. The features come mainly from the more actual papers (published after 2003). Older papers (published before 2003) with system descriptions did not provide essential aspects used for the item creation process.

4. Results

The result is presented in Table 1.

The qualitative analysis elicited 72 different items which are involved in describing registry software systems. Inspired by the caBIG® [5] criteria we created a checklist organized in twelve logical system components, functional aspects or design steps which is presented in Table 1.

Compared to our preliminary published work we now identified 22 more papers containing a patient registry software system description. Although we have approximately one-third more new papers, we added only one new item to section 12 of our list. This shows on the one hand that our approach is stable and on the other hand, that it is successful, since we could identify one more new item (item 12.4 – online help) in this evolving field.

In Table 1 we present the items, a short description of the item and relevant references from which we created the items. We cite at least one to four references per item, although some items we found in multiple references. The complete list with the 64 references providing a system description can be found in Appendix 1. For some items we do not provide a reference. These items were not discussed in the literature found by our review, but have to be considered as relevant by our own experience.

Table 1 A checklist of Items to consider when choosing or developing a software system for patient registries

Aspect/Topic	Item No.	Description	Relevant References
Software architecture	**1**	**This topic contains items related to the architecture of the patient registry software system.**	
System architecture	1.1	The system has a modular multi-tier architecture.	[15 –17]
Platform independence	1.2	The system runs on different platforms.	[18]
Open source	1.3	Open source components are used to create the software of the patient registry system and it is made open source.	[19]
Development	**2**	**This topic comprises aspects which are important during the development process of a new registry project.**	
Design model	2.1	The system itself is developed following a design model.	[20]
Framework-based design	2.2	The system provides a framework for the development process of a new registry project.	[16, 20]
Questionnaire builder	2.3	The system has a table-based or web-based questionnaire builder.	[20]
Usability testing	2.4	Usability of the system is tested. Users get involved in the development process early.	[20 – 22]
Performance testing	2.5	The performance of the system is tested.	[23]
Interfaces and interoperability	**3**	**This topic comprehends different kinds of interfaces of the registry software system and related aspects of these interfaces.**	
End-user interfaces		**This subtopic contains interfaces to end-users and related aspects.**	
Web interface	3.1	The patient registry software system has a web interface.	[15, 22]
Compatibility	3.2	The web interface is compatible with the major web browsers.	[15]
E-mail alert	3.3	E-mail alerts are possible, as reminders for follow-up, etc.	[24]
Messaging interface	3.4	There is a messaging interface to provide information for the end-users.	[25]
Online discussion forum	3.5	An online discussion forum for the end-users is established.	[25]
Mobile interface	3.6	An interface for mobile devices is available.	[15, 25 – 27]
Patient interface	3.7	The system provides also a patient interface, where quality of life (QOL) and other information can be collected.	[20]
Programming interface		**This subtopic describes programming interfaces to other patient registries and related aspects.**	
Third party access	3.8	Programmatic access to data from an external resource is possible.	[15]
API for inserting data	3.9	The system provides an application programming interface (API) for inserting data automatically.	[15, 28]

Table 1 Continued

Aspect/Topic	Item No.	Description	Relevant References
API for retrieving data	3.10	The system provides an API for retrieving data.	[15, 28, 29]
Data update mechanism	3.11	There is an update mechanism for automatically inserted data.	
Interfaces to other systems		**This subtopic contains interfaces to other systems.**	
Interface to HIS / CIS	3.12	An interface to HIS / CIS (HL7) is available to exchange data.	[20, 29, 30]
Integration of biological data	3.13	The system has an interface to integrate pseudonymized biological data (see item 6.1). See for example the concepts of the TMF e.V. [12] or an anonymizing tool like described in Prasser et al. [31].	[15]
Extensibility is possible	3.14	It is possible, if necessary, to create further interfaces to other systems.	[16]
Interoperability and semantics & standardization	**4**	**This topic contains issues of interoperability with focus on semantic and standardization aspects.**	**[10, 11]**
CRFs	4.1	Standardized CRFs are used whenever possible	[15, 25, 32]
Data	4.2	Standardized Data are used whenever possible.	[15]
Metadata	4.3	Ontology-based, standardized metadata are used.	[15, 19, 30]
Vocabularies	4.4	Ontology-based, standardized vocabularies are used.	[15, 19, 30]
XML schema	4.5	An XML schema definition (XSD) is available for structured data exchange.	[17, 22, 25]
Internationality	**5**	**This topic contains an item which a registry software system can provide to support international cooperation in a registry project.**	
Multilingualism	5.1	The whole questionnaire life-cycle can be displayed in different languages.	[20, 24]
Data management, data quality and usability	**6**	**This topic contains important items which should be considered by the data management to support data quality and usability of the registry.**	
Pseudonymous patient identifier	6.1	A pseudonymous patient identifier (PID) is created by the system.	[30, 33]
CRF is divided in parts	6.2	The CRF is divided in logical parts.	[19, 20, 25]
Customizable CRF parts	6.3	CRFs are customizable according to the user's selections.	[25]
Minimal and extended dataset	6.4	If requested, a minimal and extended dataset can be used.	[22]
All data types are supported	6.5	The system supports the use of all common data types.	[16]
Special data types are possible	6.6	Special data types, like images, X-rays, and links can also be stored.	[16]
Multiple choice is used	6.7	Multiple choice data collection is used whenever possible.	[24, 25]

Table 1 Continued

Aspect/Topic	Item No.	Description	Relevant References
No predefined selection	6.8	No predefined selection is used, to avoid unwanted entries.	[21]
Data validation components	6.9	The system has implemented data validation components (Hard- and Soft Checks).	[22, 28, 34]
Data query tool	6.10	The system has a query tool to perform automatic data queries.	
Interface for manual data check	6.11	The system has an interface for manual data check.	[35]
Manual data queries	6.12	It is possible to perform manual data queries within the system.	
Data query flags	6.13	Entries with unresolved queries are marked with flags at different levels (item, part, patient, etc.).	[33]
Plausibility flags	6.14	Implausible entries are marked with flags at different levels.	[34]
Insertion of unplanned visits	6.15	Unplanned visits can flexibly be integrated.	
Software ergonomics	6.16	The system should be designed following the standards of software ergonomics, defined in ISO 9241-110.	[20, 36]
Data analysis	**7**	**This topic contains issues with which the registry software system can support data analysis.**	
Query builder for researchers	7.1	The system has a query builder to assist researchers to select interesting patient cohorts.	[19]
Report generation	7.2	The system is able to generate reports for selected cohorts.	[19, 20]
Download of datasets	7.3	Datasets can be generated and downloaded for analysis in different formats, complete or for selected cohorts.	[19, 29]
Graphical presentation of results	7.4	Results can be presented as colored graphs in real time.	[20, 19]
Risk analysis	7.5	The system gives interactive feedback, classifying the patient through to implemented knowledge bases or scoring systems.	[20, 27, 33]
Security aspects	**8**	**This topic contains security aspects of the registry software system and important security aspects of the registry operation process.**	
Authorized users	8.1	Only authorized users have access to the data.	[15, 25]
Role-based access	8.2	The system provides role-based user access.	[15, 22]
Encrypted data transfer	8.3	The system utilizes secure web server communication through encrypted data transfer.	[15, 22, 25, 29]
Encrypted data storage	8.4	Sensitive data can be stored encrypted in the database.	[15]
Audit trail	8.5	All changes in the database are tracked and monitored through an audit trail.	[15, 16, 24, 34]
Master-slave replication	8.6	If necessary, master-slave replication should be established.	[25]
Backup management	8.7	Backups are stored separately regularly.	[25]

Table 1 Continued

Aspect/Topic	Item No.	Description	Relevant References
Firewall	8.8	The server is behind a firewall.	[35]
Server room	8.9	The server room is locked and temperature controlled.	[15, 32]
Privacy	**9**	**This topic describes privacy aspects of the registry and of the software system additional to items 6.1 and 11.4.**	
Data protection concept	9.1	A data protection concept should be established before starting a registry project.	[22]
Double pseudonymization	9.2	The registry software system should provide double pseudonymization for biological and genomic data.	
General features	**10**	**This topic contains general items which have no direct relation to the other topics.**	
Costs	10.1	The costs of a patient registry software system must be taken into consideration, when choosing a software system: Is there a realistic calculation of the costs considering the costs of the procurement / programming, operation, life-cycle, archive, system replacement? Are the financial resources sufficient regarding the anticipated costs of sustainability of the registry guaranteed?	[28, 37]
Multi-client capability	10.2	The system has a multi-client capability. Several projects can simultaneously be executed in one installation of the system, but should be strictly separated.	[19]
Update mechanism	10.3	An update mechanism for the system is in place.	[20]
Source documentation in pdf	10.4	Source documentation of CRFs in pdf format is possible.	[24]
Organizational	**11**	**This topic comprises software-related organizational items.**	
Compliance with regulations	11.1	The system is compliant with all known relevant regulations.	[22, 24]
Informed consent	11.2	The registry is compliant with Chapter 11, Title 21, Code of Federal Regulations and HIPPA.	[15, 22]
Rights on the data	11.3	There are clearly defined rules which describe the rights on the data for each participating institution.	[22]
Data protection guidelines	11.4	The system is compliant with all known appropriate data protection guidelines of the registry project.	[22]
Training	**12**	**This topic contains items which are important for the user training.**	
User manuals	12.1	There should be manuals for the registry end-users and for the operators.	[15, 26]
User training	12.2	At the beginning, and if necessary during the project time, a training is provided for the users.	[15]
User feedback	12.3	Regularly user feedback is collected for further improvements of the system.	[26, 38]
Online help	12.4	The system provides an online help for data entry.	[21]

5. Discussion

Patient registries have been defined as “an organized system that uses observational study methods to collect uniform data (clinical and other) to evaluate specified outcomes for a population defined by a particular disease, condition, or exposure, and that serves a predetermined scientific, clinical, or policy purpose(s)” (Glicklich et al. [4]). Currently, there is no consistent definition of the term “patient registry”used in the health research field. Terms such as clinical registries, clinical data registries, disease registries, and outcomes registries are also used to describe the same data collection method (Silveira et al. [39]).

Critical to the success of a patient registry is the digital technology used to enable patients and other users to join the network, report and store (and display) information, search for patients with similar experiences or conditions, and/or link to other resources. The design of a successful virtual platform requires technical expertise, patient-user involvement, and significant funding. Models of effective technical solutions and platforms for registries and networks vary widely, and have not yet been formally evaluated or compared. Patient/caregiver use of the platform, however, serves as a critical guidepost for the success or failure of the registry or network (Workman et al. [40]).

Patient registries have traditionally been researcher-generated. Research institutions, academic clinical institutions, or individual research teams establish a registry, using private or public funds, for the purpose of observational data collection that can be used for a specific research agenda. These registries may be organized and operated in a variety of forms and formats. They may be operated by a single institution or by a collaborative of multiple institutions or clinics. Researcher-generated patient registries currently exist for a wide range of conditions, including many forms of diseases.

The creation and use of researcher-generated patient registries has grown steadily for several decades, although the actual number of existing registries in Germany is unknown. As far as the United States are concerned, in 2012, the Agency for Healthcare Research and Quality launched an online registry of patient registries to provide a searchable database of patient registries in the United States (Glicklich et al. [41]).

The ideas to set up a well-designed technology for such registries are very individual and do not follow clearly published principles. Therefore, the aim of the CIPROS checklist is to provide study coordinators, developers and decision makers with an instrument to evaluate the technology and architecture of existing systems. CIPROS consists of 72 items which are organized in 12 sections.

Similar to Bellgard et al. [42], who published a checklist for rare disease registries, our approach is based on a systematic review of the literature on patient registries for medical research. Bellgard provides the big topics without further details. We found several additional unique points and our approach is more comprehensive and detailed. This paper just presents the CIPROS list. A forthcoming more extensive paper will also present an exploration and elaboration paper comparable to the presentation given in Altman et al. [8].

We performed a systematic review which allowed an objective assessment of the current literature on registry IT infrastructure for clinical registries and related systems. It also collected the very different views on this field. The planned exploration and elaboration paper will give more insight into the screened literature. Therefore, we do not present a detailed list of the different findings of the systematic review. The restricted space only allows a summary of relevant items and ideas which we ordered in 12 head categories. Therefore, CIPROS

contains subjective elements which are open for discussion. This may initialize a further development of the list.

We recommend to use the list in specific projects by checking the project plan by the proposed items. All presented features should be assessed for relevance when embarking in the development of a registry project. The items also are valuable in choosing an existing system or in developing an own solution. Another aim of our checklist is to provide persons reporting about patient registry software systems an instrument to check and present the features of their system. Like the CONSORT Statement (Altman et al. [8]), which improved the reporting of clinical trials by focusing on its relevant aspects, our checklist can help, making the description of a patient registry software system description more transparent. The CIPROS checklist can help to create standards in reporting registry-based medical research, for which there is also a need.

Since compared to the literature on clinical studies the literature on registry software systems is not well structured, it may be the case that our systematic review missed some relevant work. However, the list proposed is seen as a starting point to discuss the relevant features of such systems and to develop a more structured presentation of software systems for patient registries. For example we did not consider systems as described by Schendel et al. [43], who describe a web-based analysis tool that aggregates the data form previously harmonized, locally stored data from different sources. This system can be seen as an analysis meta-tool on registries and sections 4 and 5 of CIPROS are relevant for such technologies.

Our own use case is the EUTOS registry (Lindörfer et al. [9]), which collects baseline, treatment and outcome data of representative samples of Chronic Myeloid Leukemia (CML) patients in European countries (http://www.eutos.org). Specific results concerning the IT infrastructure behind the EUTOS registry will be presented elsewhere.

EUTOS and many other registries convincingly demonstrate that specific informatics strategies have the potential to improve care. Software to implement these strategies should be developed, and rigorously evaluated within the context of organizational efforts to improve care. The minimal requirements of these standards which have to be followed need to be explicitly formulated. CIPROS offers a first step into this direction. We believe that the given version of CIPROS is a good starting point. Since our update added 22 new papers to the existing version of 42 papers, we added only one new item (12.4) to our list. This shows on the one hand that our approach is robust and on the other hand, that it is successful, since we could identify one more new item (item 12.4 – online help) in this evolving field.

6. Conclusions

This checklist is a proposal which summarizes the main issues presented in the actual discussion on registry IT infrastructure in the Medical Informatics (MI) literature. The need for this instrument is paramount. The CIPROS list supports developers to assess requirements on existing systems and strengthens the reporting of patient registry software system descriptions. It can be a first step to create standards for patient registry software system assessments.

Acknowledgment

We would like to thank the reviewers for their constructive criticism and pointing us to four additional items which are now included into our checklist.

References

1. Mathis-Edenhofer S, Piso B. Formen medizinischer Register – Definitionen, ausgewählte methodische Aspekte und Qualität der Forschung mit Registern. Wiener Medizinische Wochenschrift 2011; 161/23–24: 580–590.
2. Anazawa T, Miyata H, Gotoh M. Cancer registries in Japan: National Clinical Database and site-specific cancer registries. Int J Clin Oncol 2014. (Epub ahead of print.)
3. Register in der Medizin. Available: http://www.arztbibliothek.de/themenschwerpunkt/themen von-a-z/medizinregister (Accessed: Dec 14, 2014).
4. Gliklich RE, Dreyer NA, eds. Registries for Evaluating Patient Outcomes: A User's Guide. Second Edition. Rockville, AHRQ Publication No. 10-EHC049, 2010.
5. Arts DG, De Keizer NF, Scheffer GJ. Defining and improving data quality in medical registries: a literature review, case study, and generic framework. J Am Med Inform Assoc 2002; 9: 600– 611, .
6. Solomon DJ, Henry RC, Hogan JG, et al. Evaluation and implementation of public health registries. Public Health Rep1991; 106: 142–150.
7. Müller D, Augustin M, Banik N, Baumann W, Bestehorn K, Kieschke J, Lefering R, Maier B, Mathis S, Rustenbach SJ, Sauerland S, Semler SC, Stausberg J, Sturm H, Unger C, Neugebauer EAM. Memorandum Register für die Versorgungsforschung. Das Gesundheitswesen 2010; 72: 824– 839.
8. Altman DG, Schulz KF, Moher D, Egger M, Davidoff F, Elbourne D, Gøtzsche PC, Lang T; CONSORT GROUP (Consolidated Standards of Reporting Trials). The revised CONSORT statement for reporting randomized trials: explanation and elaboration. Ann Intern Med 2001; 134: 663– 694.
9. Lindörfer D, Müller TH, Integration und Analyse von Patienten mit Chronischer Myeloischer Leukämie (CML) in 25 Europäischen Ländern mit webbasierter elektronischer Datenerfassung. In: Schmücker P, Elsässer K-H, Hayna S (Hrsg.): Tagungsband der 55. GMDS Jahrestagung 2010, Mannheim, 2010.
10. National Cancer Institute. The caBIG program has been retired. Available: https://cabig.nci.nih.gov/. (Accessed: Dec 14, 2014).
11. National Cancer Institute. National Cancer Informatics Program. Available: http://cbiit.nci.nih.gov/ncip. (Accessed Dec 14, 2014).
12. TMF e.V. Available: http://www.tmf-ev.de/ (Accessed: Dec 14, 2014).
13. Lindoerfer D, Mansmann U. CIPROS – A checklist with items for a patient registry software system. Stud Health Technol Inform 2014; 205: 161–165.
14. Lindoerfer D, Mansmann U. A proposal of Checklist Items for evaluating a Patient Registry Software System. In: Nußbeck SY, Wolff S, Rienhoff O (Hrsg.). Tagungsband der 59. GMDS Jahrestagung 2014, Göttingen, 2014.
15. Sherman S, Shats O, Fleissner E, Bascom G, Yiee K, Copur M, Crow K, Rooney J, Mateen Z, Ketcham MA, Feng J, Sherman A, Gleason M, Kinarsky L, Silva-Lopez E, Edney J, Reed E, Berger A, Cowan K. Multicenter breast cancer collaborative registry. Cancer Inform 2011; 10: 217–226.
16. Deserno TM, Haak D, Brandenburg V, Deserno V, Classen C, Specht P. Integrated Image Data and Medical Record Management for Rare Disease Registries. A General Framework and its Instantiation to the German Calciphylaxis Registry. J Digit Imaging 2014.
17. Messiaen C, Le Mignot L, Rath A, Richard JB, Dufour E, Ben Said M, Jais JP, Verloes A, Le Merrer M, Bodemer C, Baujat G, Gerard-Blanluet M, Bourdon-Lanoy E, Salomon R, Ayme S, Landais P. CEMARA: a Web dynamic application within a N-tier architecture for rare diseases. Stud Health Technol Inform 2008; 136: 51–56.
18. Singh SK, Malik A, Firoz A, Jha V. CDKD: a clinical database of kidney diseases. BMC Nephrol 2012; 13: 23.
19. Natter MD, Quan J, Ortiz DM, Bousvaros A, Ilowite NT, Inman CJ, Marsolo K, McMurry AJ, Sandborg CI, Schanberg LE, Wallace CA, Warren RW, Weber GM, Mandl KD. An i2b2-based, generalizable, open source, self-scaling chronic disease registry. J Am Med Inform Assoc 2013; 20 (1): 172–179.

20. Holzner B, Giesinger JM, Pinggera J, Zugal S, Schöpf F, Oberguggenberger AS, Gamper EM, Zabernigg A, Weber B, Rumpold G. The Computerbased Health Evaluation Software (CHES): a software for electronic patient-reported outcome monitoring. BMC Med Inform Decis Mak 2012; 12: 126.
21. Lanzola G, Parimbelli E, Micieli G, Cavallini A, Quaglini S. Data quality and completeness in web stroke registry as the basis for data and process mining. J Healthc Eng 2014; 5 (2): 163–184.
22. Plotnicki L, Kohl CD, Höcker B, Krupka K, Rahmel A, Pape L, Hoyer P, Marks SD, Webb NJ, Söylemezoglu O, Topaloglu R, Szabo AJ, Seeman T, Marlies Cornelissen EA, Knops N, Grenda R, Tönshoff B. The CERTAIN Registry: a novel, webbased registry and research platform for pediatric renal transplantation in Europe. Transplant Proc 2013; 45: 1414–1417.
23. Steurbaut K, De Backere F, Keymeulen A, De Leenheer M, Smets K, De Turck F. NEOREG: design and implementation of an online Neonatal Registration System to access, follow and analyse the data of newborns with congenital cytomegalovirus infection. Inform Health Soc Care 2013; 38 (3): 223–235.
24. Sanderson G, Ariyaratne TV, Wyss J, Looi V. A global patient outcomes registry: Cochlear paediatric implanted recipient observational study (Cochlear™ P-IROS). BMC Ear Nose Throat Disord 2014; 14: 10
25. Xie F, Zhang D, Wu J, Zhang Y, Yang Q, Sun X, Cheng J, Chen X. Design and implementation of the first nationwide, web-based Chinese Renal Data System (CNRDS). BMC Med Inform Decis Mak 2012; 12: 11.
26. France DJ, Miles P, Cartwright J, Patel N, Ford C, Edens C, Whitlock JA. A chemotherapy incident reporting and improvement system. Jt Comm J Qual Saf 2003; 29 (4): 171–180. Erratum in: Jt Comm J Qual Saf 2003; 29 (5): 209.
27. Zargaran E, Schuurman N, Nicol AJ, Matzopoulos R, Cinnamon J, Taulu T, Ricker B, Garbutt Brown DR, Navsaria P, Hameed SM. The electronic Trauma Health Record: design and usability of a novel tablet-based tool for trauma care and injury surveillance in low resource settings. J Am Coll Surg 2014; 218 (1): 41–50.
28. Teng JE, Thomson DR, Lascher JS, Raymond M, Ivers LC. Using Mobile Health (mHealth) and Geospatial Mapping Technology in a Mass Campaign for Reactive Oral Cholera Vaccination in Rural Haiti. PLoS Negl Trop Dis 2014; 8 (7): e3050.
29. Gillies MC, Walton R, Liong J, Arnold JJ, McAllister I, Morlet N, Hunyor A, Guymer R, Keeffe J, Essex R, Herrera-Bond A, Glastonbury B, Simpson JM, Barthelmes D. Efficient capture of highquality data on outcomes of treatment for macular diseases: the fight retinal blindness! Project. Retina 2014; 34 (1): 188 –195.
30. da Silva KR, Costa R, Crevelari ES, Lacerda MS, de Moraes Albertini CM, Filho MM, Santana JE, Vissoci JR, Pietrobon R, Barros JV. Glocal clinical registries: pacemaker registry design and implementation for global and local integration – methodology and case study. PLoS One 2013; 8 (7): e71090.
31. Prasser F, Kohlmayer F, Lautenschläger R, Kuhn KA. ARX – A Comprehensive Tool for Anonymizing Biomedical Data. AMIA Annual Symposium Proceedings 2014; 2014: 984 –993.
32. Sherman S, Shats O, Ketcham MA, Anderson MA, Whitcomb DC, Lynch HT, Ghiorzo P, Rubinstein WS, Sasson AR, Grizzle WE, Haynatzki G, Feng J, Sherman A, Kinarsky L, Brand RE. PCCR: Pancreatic Cancer Collaborative Registry. Cancer Inform 2011; 10: 83 –91.
33. Ko GT, So WY, Tong PC, Le Coguiec F, Kerr D, Lyubomirsky G, Tamesis B, Wolthers T, Nan J, Chan J. From design to implementation – the Joint Asia Diabetes Evaluation (JADE) program: a descriptive report of an electronic web-based diabetes management program. BMC Med Inform Decis Mak 2010; 10: 26.
34. Pang X, Kozlowski N, Wu S, Jiang M, Huang Y, Mao P, Liu X, He W, Huang C, Li Y, Zhang H. Construction and management of ARDS/sepsis registry with REDCap. J Thorac Dis 2014; 6 (9): 1293 –1299.

35. Wang Y, Tao Z, Cross PK, Le LH, Steen PM, Babcock GD, Druschel CM, Hwang SA. Development of a web-based integrated birth defects surveillance system in New York State. J Public Health Manag Pract 2008; 14 (6): E1-E10.
36. Ergonomic requirements for office work with visual display terminals (VDTs) – Part 11: Guidance on usability (ISO 9241-11:1998). Available at: http://www.it.uu.se/edu/course/homepage/acsd/vt09/ISO9241part11.pdf. (Accessed July 14, 2015)
37. Hira AY, Lopes TT, de Mello AN, Filho VO, Zuffo MK, de Deus Lopes R. Establishment of the Brazilian telehealth network for paediatric oncology. J Telemed Telecare 2005; 11 (Suppl 2): S51–52.
38. O'Reilly DJ, Bowen JM, Sebaldt RJ, Petrie A, Hopkins RB, Assasi N, MacDougald C, Nunes E, Goeree R. Evaluation of a chronic disease management system for the treatment and management of diabetes in primary health care practices in Ontario: an observational study. Ont Health Technol Assess Ser 2014; 14 (3): 1– 37.
39. Silveira P, Rodríguez C, Casati F, et al. On the Design of Compliance Governance Dashboards for Effective Compliance and Audit Management. In: Dan A, Gittler F, Toumani F, editors. Service-Oriented Computing. ICSOC/ServiceWave 2009 Workshops. Vol. 6275. Berlin, Heidelberg: Springer 2010. pp. 208 –217.
40. Workman TA. Rockville (MD): Agency for Healthcare Research and Quality (US); 2013 Sep. Engaging Patients in Information Sharing and Data Collection: The Role of Patient-Powered Registries and Research Networks [Internet]. Available at: http://www.ncbi.nlm.nih.gov/books/NBK164514/ (Accessed: Dec 14, 2014)
41. Gliklich R, Levy D, Karl J, et al. Registry of Patient Registries (RoPR): project overview. Rockville, MD: Agency for Healthcare Research and Quality;2012.
42. Bellgard M, Beroud C, Parkinson K, Harris T, Ayme S, Baynam G, Weeramanthri T, Dawkins H, Hunter A. Dispelling myths about rare disease registry system development. Source Code Biol Med 2013; 8 (1): 21.
43. Schendel DE, Bresnahan M, Carter KW, Francis RW, Gissler M, Grønborg TK, Gross R, Gunnes N, Hornig M, Hultman CM, Langridge A, Lauritsen MB, Leonard H, Parner ET, Reichenberg A, Sandin S, Sourander A, Stoltenberg C, Suominen A, Surén P, Susser E. The International Collaboration for Autism Registry Epidemiology (iCARE): multinational registry-based investigations of autism risk factors and trends. J Autism Dev Disord 2013; 43 (11): 2650 –2663.

3.2 Publikation 2 - Journal of Biomedical Informatics, 2017

Lindoerfer, D. and Mansmann, U. (2017). Enhancing Requirements Engineering for Patient Registry Software Systems with Evidence-based Components.*J Biomed Inform,* 71:147-153.

https://doi.org/10.1016/j.jbi.2017.05.013

https://www.sciencedirect.com/science/article/pii/S1532046417301090?via%3Dihub

Im Anschluss:

Ebenfalls online abrufbar:

Appendix C dieser Publikation - Vergleich von CIPROS mit den Standards und Templates zur Anforderungsanalyse.

Danach:

Graphical Abstract zu dieser Publikation: Erstellung und Zweck der CIPROS Checkliste.

Dieses Graphical Abstract ist zusammen mit der Publikation und den Supplements auch online verfügbar.

Journal of Biomedical Informatics 71 (2017) 147–153

Contents lists available at ScienceDirect

Journal of Biomedical Informatics

journal homepage: www.elsevier.com/locate/yjbin

Enhancing requirements engineering for patient registry software systems with evidence-based components

Doris Lindoerfer *, Ulrich Mansmann [1]

Institute for Medical Information Processing, Biometry and Epidemiology (IBE), Ludwig-Maximilians-Universität München, Munich, Germany

ARTICLE INFO

Article history:
Received 28 October 2016
Revised 12 May 2017
Accepted 16 May 2017
Available online 20 May 2017

Keywords:
Checklist
Patient registry software system
Technical requirements
Evidence-based requirements engineering

ABSTRACT

Introduction: Patient registries are instrumental for medical research. Often their structures are complex and their implementations use composite software systems to meet the wide spectrum of challenges. Commercial and open-source systems are available for registry implementation, but many research groups develop their own systems. Methodological approaches in the selection of software as well as the construction of proprietary systems are needed. We propose an evidence-based checklist, summarizing essential items for patient registry software systems (CIPROS), to accelerate the requirements engineering process.

Methods: Requirements engineering activities for software systems follow traditional software requirements elicitation methods, general software requirements specification (SRS) templates, and standards. We performed a multistep procedure to develop a specific evidence-based CIPROS checklist: (1) A systematic literature review to build a comprehensive collection of technical concepts, (2) a qualitative content analysis to define a catalogue of relevant criteria, and (3) a checklist to construct a minimal appraisal standard.

Results: CIPROS is based on 64 publications and covers twelve sections with a total of 72 items. CIPROS also defines software requirements. Comparing CIPROS with traditional software requirements elicitation methods, SRS templates and standards show a broad consensus but differences in issues regarding registry-specific aspects.

Discussion: Using an evidence-based approach to requirements engineering for registry software adds aspects to the traditional methods and accelerates the software engineering process for registry software. The method we used to construct CIPROS serves as a potential template for creating evidence-based checklists in other fields.

Conclusion: The CIPROS list supports developers in assessing requirements for existing systems and formulating requirements for their own systems, while strengthening the reporting of patient registry software system descriptions. It may be a first step to create standards for patient registry software system assessments.

1. Introduction

Patient registries are a widespread and important instrument in medical research, clinical epidemiology, and quality management [1]. The number of medical registries is constantly increasing, which demonstrates their cumulative importance over the last years [2]. Medical registries serve different purposes, among others public health, epidemiology, quality control, clinical registries, disease-specific registries, etc. They present different conceptual and technical challenges for example interoperability, data quality, security, protection of data privacy, etc.

Registries document clinical phenotypes, potentially also with molecular information, and are instrumental for deep phenotyping. Arts et al. [3] observed that the value of a registry strongly depends on the quality of the data contained within. More than 20 years ago Solomon et al. [4] stressed that the successful implementation and use of a registry is determined by the thorough and accurate planning and construction of a suitable IT infrastructure. Furthermore,

* Corresponding author at: Institut für medizinische Informationsverarbeitung, Biometrie und Epidemiologie (IBE), Ludwig-Maximilians-Universität München, Marchioninistraße 15, D-81377 München, Germany.

E-mail addresses: lindoerf@ibe.med.uni-muenchen.de (D. Lindoerfer), mansmann@ibe.med.uni-muenchen.de (U. Mansmann).

[1] Institut für medizinische Informationsverarbeitung, Biometrie und Epidemiologie (IBE), Ludwig-Maximilians-Universität München, Marchioninistraße 15, D-81377 München, Germany.

http://dx.doi.org/10.1016/j.jbi.2017.05.013
1532-0464/

it is also important to focus on legal, and ethical aspects, as well as strategies for patient recruitment [1]. Müller et al. [5] summarized these aspects and provided a checklist for registries in health services research. In this publication there is a clear link to IT infrastructure, but this aspect was never explicitly discussed by the authors ([1–5]).

The IT infrastructure of a registry is an integrated set of interactive elements such as services, people, processes, hardware, software, logic, and information. In addition to the central database, a series of interfaces and procedures have to be implemented to guarantee the high-level involvement and cooperation of partners. For example, the German Endoprosthesis Registry [6] involves manufacturers of endoprostheses, health insurance companies, and a large number of hospitals where endoprostheses are implanted. The creation of consistent patient and product histories from all these sources requires an IT system that supports communication, as well as a complex verification system, to produce reliable information on the quality of endoprostheses used in the German healthcare setting. Similar registries exist in Europe [7] and the United States [8].

In spite of the availability of commercial and open-source software systems many institutions develop their own solutions to fit their needs and are flexible in addressing changing requirements. For this reason guidance is needed to support the assessment and selection of an existing, or the design of a new registry software system.

There is clear evidence that requirement-related issues are the main cause of the majority of project failures and problems. There are two issues: (1) Instruments to manage projects and requirements, (2) strategies to formulate requirements. In order to support the first step big software projects are planned following a life-cycle model that defines the necessary steps to be performed during the project time to move forward [9], these are the waterfall model [10], its extension the V-model [11], the spiral model [12] as a risk-driven enhancement of the waterfall model, and the saw tooth model [13]. The shark tooth model [9] is a refinement of the saw tooth model by also taking management reviews and demonstrations into account. Today, agile software development models [14] provide processes in which requirements and solutions evolve through collaboration between self-organizing cross-functional teams. These instruments need more admission into the medical community [15].

A collection of templates exists to specify requirements for projects at hand: The Volere Requirements Specification Template [16] supports the formulation of software requirements by asking the right questions. The ISO/IEC/IEEE 29148:2011 International Standard [17] is a comprehensive document that also provides templates for the requirements specifications of software projects. Bruegge and Dutoit [9], as well as Wiegers and Beatty [18] describe software requirements elicitation methods and provide software requirements specification (SRS) templates. However, all these templates and standards are generic and not specific to the medical field.

To this end, we developed a **c**hecklist to support the requirements engineering process for **p**atient **r**egistry s**o**ftware **s**ystems (CIPROS) [19], which is inspired by the CONSORT[2] Statement [20] and based on a systematic literature review. Similar to the idea of Altman et al. [20] to improve the reporting of clinical trials, the CIPROS checklist [19] aims to support the construction or selection of patient registry software systems.

A separate elaboration paper [21] provides examples and explanations for each item, which can be tailored by specifying the software requirements for the researchers' own systems and by implementing these features in software systems for their own projects.

The aim of this paper is to (1) provide an overview of traditional requirements elicitation methods and standards, (2) introduce a new evidence-based method for requirements elicitation, (3) demonstrate this method through the evidence-based checklist creation in reporting on CIPROS development, and (4) compare the CIPROS checklist with traditional SRS templates and standards.

The Section 2 (Methods) introduces the traditional requirements elicitation methods, along with the software requirements specification templates and standards. It also presents the specific aims of CIPROS development: (1) a systematic literature review (SLR) to identify papers with patient registry software system descriptions, (2) a qualitative content analysis (QCA) according to Mayring [22] of identified publications to extract information describing software-specific features of patient registry software systems, and (3) development of a checklist using the information of the qualitative content analysis. The Section 3 (Results) offers a description of CIPROS and a comparison with the specific SRS Templates.

Our work was inspired by three initiatives, which supported clinical research and interoperability by providing adequate software systems and semantic infrastructure: (1) Within the medical community and in spite of its practical failure, the caBIG® Program from the National Cancer Institute [23] was a starting point for a general movement to create and share structured clinical knowledge. The achievement of caBIG was to introduce system requirements and to certify existing software systems based on these requirements. (2) The recent National Cancer Informatics Program [24] aims to support interoperability in medical research by providing open-source code for biomedical informatics software applications and semantic infrastructures. (3) Two German projects are provided by the "Technologie- und Methodenplattform für die vernetzte medizinische Forschung e.V." (TMF e.V.) [25]: A National Metadata Repository for Clinical Studies and Registries and a second project that defines common requirements for cohorts and registry IT.

2. Methods

2.1. Traditional software requirements engineering

2.1.1. Traditional software requirements engineering methods

Interviews with stakeholders are the most popular method to identify and gather the requirements of a new software product. They are used to derive scenarios and use cases. In a more advanced project phase, these use cases are transferred into class diagrams and are subsequently implemented. **Workshops** and **Expert panels** are also important and widespread methods to define the requirements of a new software product. This process guarantees that experiences from different persons and projects will be considered. Of course, the **personal experiences** of those involved also play an important role.

2.1.2. A selection of software requirements specification templates

Several software requirements specification (SRS) templates exist. Some are freely available on the Internet, and others can be purchased or are published as book chapters. Here is a short list of prominent examples:

- The ISO/IEC/IEEE 29148:2011 International Standard [17] describes in detail how requirements should be written in an SRS document, which standards and abbreviations should be used, how the requirements analysis process should be

[2] Consolidated Standards of Reporting Trials (CONSORT), http://www.consort-statement.org/

organized, etc. It distinguishes between stakeholder requirements specification (StRS), system requirements specification (SyRS), and software requirements specification (SRS). It also includes requirements regarding overall and change management.
- The compact SRS template of Wiegers [18] structures the specification of the project-specific content and requirements. The first version was published in 1999 [26] and is now freely available on the internet. We used the updated version published in 2013 [18].
- The SRS template by McKinnon [27] was published in 2005. The document offers a structure for the specification of project-specific content and requirements based on the IEEE830-1984 Standard. It is freely available on the internet.
- The Volere SRS template [16] was developed by James and Suzanne Robertson from 1995 to 2015. It is a comprehensive template for specifying project-specific content and requirements while providing examples that indicate how to write requirements. It may be purchased via the Volere webpage [16].

2.2. Evidence-based checklist creation by qualitative content analysis

A qualitative content analysis (QCA) method, developed by Mayring [22], was adopted to create a checklist for SRS in patient registry software systems by performing three steps:

(1) A systematic literature review,
(2) A qualitative content analysis, and
(3) Creation of an early version of the checklist.

2.2.1. Systematic literature review

A systematic literature review (SLR) was performed to collect, in an objective way, papers with patient registry software system descriptions. In order to obtain high-quality publications, we performed the search in PubMed to guarantee that the software systems are used in real-life medical research projects. The first search was performed on 17th of January 2014 using the query "*(registry or registries) AND (eCRF or EDC or CDMS or CTMS or web) AND (software or open-source or open source or Java)*". An update using the same search strategy followed on 15st of February 2015.

The search terms were motivated by our interests in registries with electronic case report forms (eCRFs), electronic data capture (EDC) technology, clinical data management systems (CDMS) or clinical trial management systems (CTMS), as well as in software systems or open-source systems. We also decided to include the term "Java" because it was included in the title of some publications on software systems.

2.2.2. Qualitative content analysis

The qualitative content analysis followed the method of Mayring [22]. This method follows the step model of inductive category development, according to the steps shown in Fig. 1.

We formulated the research question: "Is there a description of software-related features?" The titles and abstracts of all papers found in the SLR were screened with regard to the following inclusion criteria: (1) patient registry software systems are described, and (2) the language of the paper is English or German. For all included papers the full texts were analyzed. A publication was considered for further analysis if a system description, or at least a short system description, was provided. We started with a large set of publications and searched these for descriptions of relevant software-specific features. We collected significant phrases describing these features in a list, built inductive categories out of the material, and categorized the items or formulated new categories. After reading approximately 50% of the papers we revised the categories. Then we read the remaining papers.

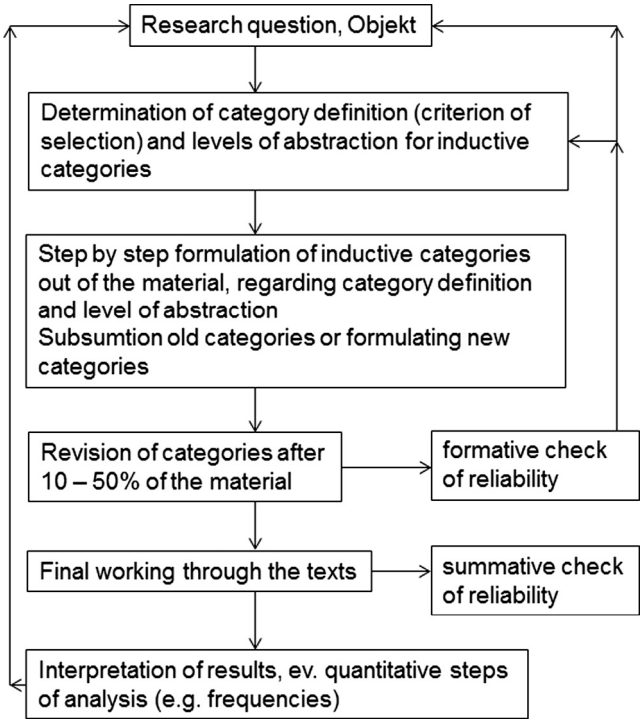


Fig. 1. Step model of inductive category development [22].

2.2.3. Creation of an early version of the item list for the CIPROS checklist

We created an early version of the checklist using the step model of deductive category application, which is described by Mayring [22] and shown in Fig. 2.

We compiled the phrases retrieved from the QCA into an early version of our item list. We defined main categories and subcategories and sorted the items into these categories. Within a feedback loop, we revised these categories, eventually reduced

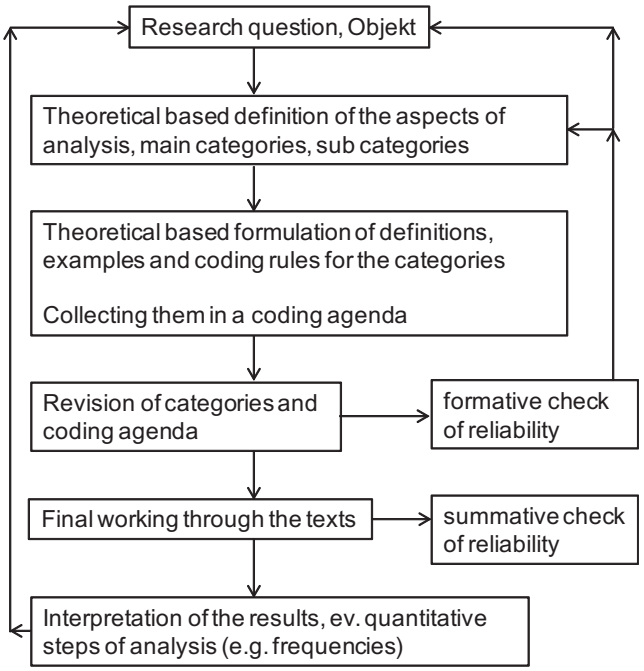


Fig. 2. Step model of deductive category application [22].

them to main categories, and sorted the items into the new categories. If we found a new feature we added a new item within the corresponding category. A new item was added for each new feature, regardless of how often we found the feature. Finally, we enhanced the checklist with items that we did not find in any of the papers but considered important based on our own experiences (the respective items are listed in Section 3 (Results), subsection 3.1.2).

We classified the identified software-specific features into system components (S), functional aspects (F), and design steps (D).

3. Results

The development of the CIPROS checklist for patient registry software systems is described, and a concise summary of the checklist is presented. This is followed by a comparison of CIPROS with other SRS templates and standards.

3.1. The evidence-based checklist

3.1.1. Development of the CIPROS checklist

The CIPROS checklist is based on an updated search.

(1) The first search, performed on the 17th of January 2014, delivered 132 papers. The first version of the CIPROS checklist was created. We performed an updated search with the same criteria. Our search (15st of February 2015) delivered 157 papers. A PRISMA[3] flowchart [28,29] is given in Fig. 3. We identified 25 additional papers that were not available during the first search. We classified the total of 157 papers according to their content by reading the titles and abstracts, and then applying the criteria described in the methods section. No duplicate records were removed. No records were excluded. All 157 records were assessed for eligibility.

A total of 64 papers provided a patient registry software system description or at least a short system description (the update added 22 new papers to the discussion) and were selected for further analysis. Of the remaining 93 papers, a total of 26 were about biological information systems, 24 were about infrastructures for clinical or other medical systems, and 40 dealt with other aspects and contained no system description. Three papers were written in languages other than English or German (Spanish, Italian, and Swedish).

(2) A qualitative content analysis, according to Mayring [22], was performed and enhanced with our own experiences.
(3) We created a checklist that was organized into twelve logical system components, functional aspects and design steps, which can be found in Appendix A.

Our approach is evidence-based because the items in the CIPROS checklist were derived from previously published papers. It is based on systems that are used in real-life medical research projects published in PubMed-indexed journals. For this reason our approach differs from all the other SRS templates. All of the cited SRS templates are general, not content specific, and do not utilize items from published literature. The development process also used theoretical background but was enhanced by empirical findings in the literature.

3.1.2. The CIPROS checklist

CIPROS consists of 72 items, organized into 12 topics presented in Appendix A. In an additional Data-in-Brief article [21] we illustrate each item by citing examples from the literature and giving explanations. The complete list, with the 64 references that provide system descriptions, can be found in Appendix B. For some items, we do not provide a reference, as these items were not discussed in the literature that we found in our SLR. However, they are considered relevant based on our own experience and theoretical reasoning (items no. 3.11, 6.10, 6.12, 6.15, 9.2).

In the following section we will provide a short summary of the CIPROS checklist by describing each topic and summarizing the items it contains.

3.1.3. A summary of CIPROS

3.1.3.1. Software architecture

The system should have a *modular multi-tier architecture*, it should *run on different platforms*, and it should be *open-source*.

3.1.3.2. Development

The system itself should be developed following a *design-model*, it should provide a *framework* for the development process of a new registry project, and it should provide a *table-based or web-based questionnaire builder*. The *usability* and *performance* of the system should be tested.

3.1.3.3. Interfaces and interoperability

End-user interfaces: The system has a *web interface* that is *compatible with the major web-browsers*. *Email alerts* are possible, a *messaging interface* and an *online discussion* forum are available. There is a *mobile interface* to the system and a *patient interface*.

Programming interfaces: *Third-party access* to the data from an external resource is possible, and there are *APIs for inserting data* and *retrieving data*. There is also a *data-update mechanism*.

Interfaces to other systems: An *interface to HIS/CIS*[4] *(HL7)*[5] should be available, as well as *integration of pseudonymized biological data*. *Extensibility* should be possible.

3.1.3.4. Interoperability and semantics & standardization

The system should use *standardized Case Report Forms (CRFs), data, metadata* and *vocabularies*. An *XML*[6] *schema* for structured data exchange should be available.

3.1.3.5. Internationality

To support *multilingualism* the whole questionnaire is available in different languages.

3.1.3.6. Data management, data quality, and usability

The system is able to create a *pseudonymous patient identifier (PID)*. The CRF *is divided into logical parts*. CRF parts are *customizable* according to the user's selections. A *minimal or extended dataset* can be used, *all common data types are supported*, and *special data types are also possible*. *Multiple-choice data collection is used whenever possible*. To avoid unwanted entries *no predefined selection is used*. The system has *implemented data validation components*. A *data query tool* is available, as well as an *interface for manual data verification or validation*. It is possible to perform *manual data queries* within the system. Entries with *unresolved queries* or *implausible entries* are flagged at different levels. *Unplanned visits can be easily integrated*, and the system is designed according to the standards of *software ergonomics* defined in ISO 9241-110.

3.1.3.7. Data analysis

The system provides a *query builder for researchers*, and it is able to provide *reports* for selected cohorts. Complete *datasets* or

[3] Preferred Reporting Items for Systematic Reviews and Meta-Analyses (PRISMA)

[4] Hospital Information System (HIS) / Clinical Information System (CIS)

[5] Health Level Seven (HL7), http://www.hl7.org/

[6] Extensible Markup Language (XML), https://www.w3.org/XML/

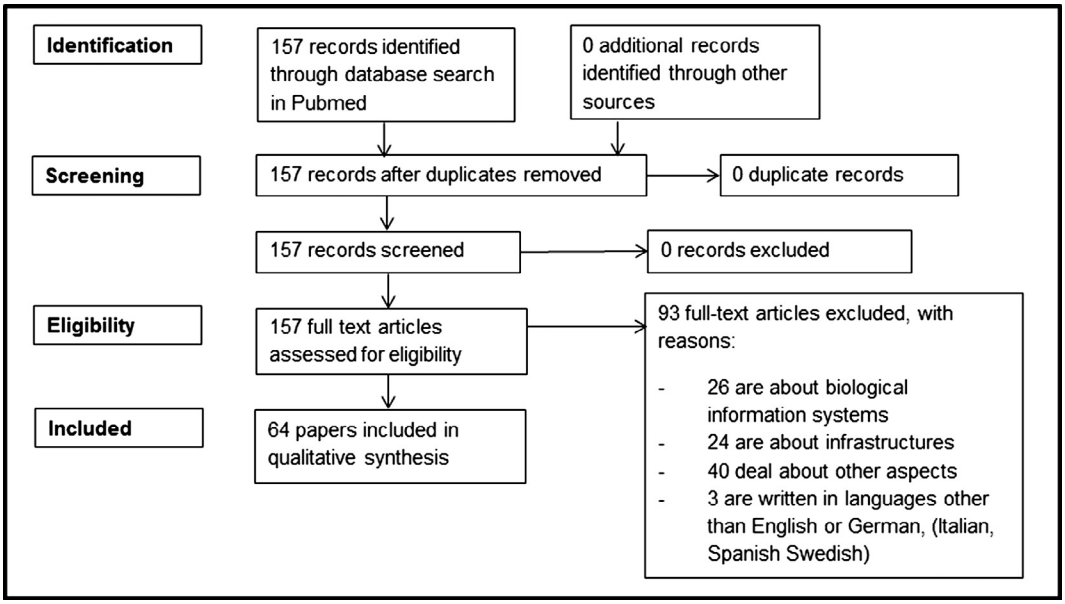


Fig. 3. PRISMA Flow diagram, explaining the number of selected papers.

selected cohorts can be *generated* and *downloaded* for analysis in different formats. Results can be presented as *colored graphs* in real time. The system gives interactive feedback on risk analysis.

3.1.3.8. Security aspects

The system provides *role-based user access* for *authorized users* only. *Encrypted data transfer* is standard and *encrypted data storage is possible*. All changes are stored in an *audit trail*. If necessary, *master-slave replication* should be established. *Backups* are stored regularly. The server is behind a *firewall* and located in a *locked* and *temperature-controlled server room*.

3.1.3.9. Privacy

A *data-protection concept* should be established before starting the project, and the software system should provide *double pseudonymization* for biological and genomic data.

3.1.3.10. General features

The *costs* of the project must be taken into consideration. The system provides a *multi-client capability*, so several projects can simultaneously be executed in one installation of the system. An *update mechanism* for the system is in place and *source documentation of CRFs in pdf format* is possible.

3.1.3.11. Organizational

The system is *compliant with all known relevant regulations*. *Informed consent* is collected according to Chapter 11, Title 21, Code of Federal Regulations and HIPPA. The *rights to the data* are documented, and the system is compliant with all known appropriate *data protection guidelines*.

3.1.3.12. Training

There should be *manuals* for the *registry end users* and for the *operators*. A *user training session* should be held if necessary. *User feedback* is collected regularly, and if suitable, *online help* is implemented.

3.2. Comparison

SRS templates specify the description of a project and the corresponding requirements. CIPROS is an SRS tool that suggests a list of potentially relevant requirements for the project. CIPROS is also useful to check if an already existing software system provides all the relevant features.

Appendix C provides an overview of CIPROS items and how they are matched with the elements of SRS templates and the ISO/IEC/IEEE 29148 standard. This matching is not perfect because the ISO/IEC/IEEE 29148 standard and the SRS templates define how the requirements should be described, while CIPROS provides a list of requirements that can be implemented in a patient registry software system. The standard and the SRS templates differentiate between functional requirements, non-functional requirements, and constraints.

Some of the CIPROS items can be matched to non-functional requirements (for example, performance and security requirements). Other items relate to constraints (for example, costs). The standard and the SRS templates provide space for, with no further specification, functional requirements [16,17,27], product functions [16], and system features [18], which are fully defined in the CIPROS checklist.

4. Discussion

The success of a patient registry project depends on the quality of the data contained in it. The data quality relies to a great extent on the planning and construction of its software system [3,4]. The construction of patient registries is a highly interdisciplinary field encompassing data bases, data management, knowledge structure, interfaces, etc. Normally, medical staff provides the data and computer scientists construct the software system. Both are experts in their respective fields. Medical staff has a subject-matter understanding of what the system should do but usually have little experience in constructing a fitting IT infrastructure. Computer scientists are experts in constructing computer systems but normally have little understanding of patient registries, which reflect in an objective way complex clinical processes and individual medical histories. They probably have a less detailed understanding of the specific medical project for which the system is constructed. Requirements will come from both sides, and they must be thoroughly elicited and specified. The clinician will contribute the requirements dictated by the clinical environment, and the computer scientist will stress the conceptual and technical points of

a successful IT infrastructure. To deliver a successful project both sides have to meet. SRS templates may be helpful to this end.

There exist a broad range of requirements specification templates [16,18,27] and standards [17], but all of them are general and not specific to any field. As a contribution to bridge the gap between both disjunctive groups of experts we developed CIPROS [19]. CIPROS is based on an SLR of already published papers with patient registry software system descriptions which are available via PubMed.

Drolet and Johnson [30] discuss important points regarding registry architecture from the perspective of databases. Our approach is more comprehensive and explores the overarching infrastructure needed to install and to run registries, from development and training, web-based data collection and security aspects to data storage, checking and analysis.

A systematic review is a type of literature review based on a protocol that defines a reproducible search and information extraction. It involves collecting and critically analyzing multiple research studies or publications. A review of existing studies is often quicker and cheaper, providing comprehensive information about what to do before embarking on a new study. Systematic reviews are not limited to medicine and are quite common in all other sciences where data are collected and published in the literature, and for which an assessment of methodological quality for a precisely defined subject would be helpful [31]. Improving methodological quality was the motivation to start the SLR. For example, it was possible to change the search strategy by adding more programming languages (Add PHP or Python or Perl to the search item JAVA). Within the time frame of the SLR this search did not bring new information.

Following the information collection of the SLR a strategy is needed to process this information in an objective way. We investigated the qualitative research strategies for processing information. Qualitative methods are often part of survey methodology. Our SLR is a survey in a population regarding a specific question. We used qualitative content analysis, which works out an inductive development of categories and deductive application of categories.

We call our approach empirical and evidence-based because the items in the CIPROS checklist are derived from already published papers. These items are based on systems that are used in real-life medical research projects that are published in PubMed-indexed journals. This differentiates our approach from all the other SRS templates. All of the cited SRS templates are general, are not content specific, and do not provide items from the literature.

Some of the more general items in our checklist are also covered by the SRS templates, but all the patient-registry specific items are only provided by our checklist.

5. Conclusion

The CIPROS checklist [19] provides an instrument for requirements engineering in software projects for patient registry software systems. This checklist provides 72 items, which are organized into twelve logical system components, functional aspects, and design steps. The items can be used for the creation of a new patient registry software system or they can be checked when a new system is selected. The checklist is not a SRS template and it will not replace existing SRS templates, but it can help to create new systems with a specific focus by accelerating the requirements engineering process. It brings the already published treasure trove of experiences into the development process of new software systems for patient registries. The approach we used to construct CIPROS can be utilized as a template for creating similar checklists in other fields.

A literature review is helpful for a systematic overview of the opinions and experiences in a specific field. This provides important knowledge when planning new projects. A limitation of our approach is the restricted scope of our literature research. Our focus was on publications available in PubMed. This excludes books and specific conference proceedings.

We wrote this paper to provide researchers and developers with a tool to assist in describing software requirements for their own projects and in selecting existing systems for their own patient registry software projects. The examples and explanations in [21] can help to specify requirements for similar projects, and may be an inspiration to implement these features in other projects.

Another aim of our checklist is to provide persons reporting on patient registry software systems with an instrument to check and present the features of their system. Like the CONSORT Statement [20] which improved the reporting of clinical trials by focusing on their relevant aspects by providing examples and explanations, our checklist can help make the description of a patient registry software system more transparent. The CIPROS checklist can help to create standards in reporting registry-based medical research, for which there is also a need.

Compared to papers on clinical studies, papers on patient registry software systems appear quite disparate with respect to structure and content. This makes it very difficult to compare the architectures and features of different systems. Therefore, we see the need to unify the presentation of patient registry software systems. This motivated us to develop CIPROS.

Authors' contributions

DL and UM developed the conception and design, drafted the manuscript and revised it critically for important intellectual content. They gave final approval of the version to be submitted. DL performed the systematic literature review, the qualitative content analysis, the creation of the checklist and the comparison of the CIPROS checklist with the SRS templates and the ISO/IEC/IEEE 29148 standard. UM supervised the complete work and took part in the qualitative analysis part.

Conflicts of interest

none

Ethical approval

For this study, no patient or animal data were used, and therefore, no ethical approval is needed.

Funding

This research did not receive any specific grant from funding agencies in the public, commercial, or not-for-profit sectors.

Summary points

What was already known on this topic?

- There are general methods and templates to support the requirements engineering process, for example, [16–18,27].
- The National Cancer Informatics Program [24] provides open-source code and semantic infrastructures for Medical Research.
- The CIPROS Checklist for evaluating patient registry software systems is published in [19].

What this study has added to our knowledge:

- This paper provides insights into the structure and intentions of the CIPROS checklist.
- The techniques of systematic review and evidence synthesis were used to collect and process subject-specific information in an objective way.
- The collected evidence can be reviewed and used for other projects.
- Qualitative content analysis was used as an evidence-based strategy for software requirements specification.

Acknowledgments

We would like to thank PD Dr. Eva Rehfuess for discussions and her intellectual contribution in performing the systematic literature review, Dr. Michaela Coenen for discussions and her intellectual contribution in performing and describing the qualitative content analysis of the literature, and James Robertson for providing us free access to the Volere Requirements Specification Template. Furthermore we would like to thank Kathrin Halfter for her contribution in English polishing of this manuscript.

Appendix A. Supplementary material

Supplementary data associated with this article can be found, in the online version, at http://dx.doi.org/10.1016/j.jbi.2017.05.013.

References

[1] R.E. Gliklich, N.A. Dreyer (Eds.), Registries for Evaluating Patient Outcomes: A User's Guide, Second Edition. Rockville, AHRQ Publication No. 10-EHC049, 2010.
[2] T. Anazawa, H. Miyata, M. Gotoh, Cancer registries in Japan: national clinical database and site-specific cancer registries, Int. J. Clin. Oncol. 20 (1) (2015) 5–10, http://dx.doi.org/10.1007/s10147-014-0757-4.
[3] D.G. Arts, N.F. De Keizer, G.J. Scheffer, Defining and improving data quality in medical registries: a literature review, case study, and generic framework, J. Am. Med. Inform. Assoc. 9 (2002) 600–611.
[4] D.J. Solomon, R.C. Henry, J.G. Hogan, G.H. van Amburg, J. Taylor, Evaluation and implementation of public health registries, Public Health Rep. 106 (1991) 142–150.
[5] D. Müller, M. Augustin, N. Banik, W. Baumann, K. Bestehorn, J. Kieschke, R. Lefering, B. Maier, S. Mathis, S.J. Rustenbach, S. Sauerland, S.C. Semler, J. Stausberg, H. Sturm, C. Unger, E.A.M. Neugebauer, Memorandum register für die Versorgungsforschung, Das Gesundheitswesen. 72 (2010) 824–839.
[6] EPRD: Endoprothesenregister Deutschland, 2015. <http://www.eprd.de> (accessed 11 May 2017).
[7] Patient registries of Europe, 2015. <http://www.parent-ror.eu/#/registries> (accessed 11 May 2017).
[8] List of Registries, 2016. <https://www.nih.gov/health-information/nih-clinical-research-trials-you> (accessed 11 May 2017).
[9] B. Bruegge, A. Dutoit, Object-Oriented Software Engineering Using UML, Patterns, and Java, third ed., Pearson, Harlow, Essex, 2014.
[10] W.W. Royce, "Managing the development of large software systems: Concepts and techniques." In IEEE WESTCON, 1-9. Los Angeles, CA, 1970.
[11] R.W. Jensen, C.C. Tonies, Software Engineering, Prentice Hall, Englewood Cliffs, NJ, 1979.
[12] B. Boehm, A spiral model of software development and enhancement, ACM SIGSOFT Softw. Eng. Notes (ACM) 11 (4) (1986) 14–24.
[13] R.B. Rowen. Software project management under incomplete and ambiguous specifications. IEEE Transactions on Engineering Management, vol. 37, no. 1, 1990.
[14] I. Sommerville, Software Engineering, 10th ed., Pearson, Harlow, Essex, 2016.
[15] M. Kossmann, The significance of requirements engineering for the medical domain, J. Med. Eng. Technol. 38 (2014) 238–243, http://dx.doi.org/10.3109/03091902.2014.918199.
[16] Volere Requirements Specification Template. <http://www.volere.co.uk/template.htm>. (accessed 11 May 2017).
[17] IEEE STANDARD. 29148-2011 – Systems and Software Engineering – Life Cycle Processes –Requirements Engineering. <https://standards.ieee.org/findstds/standard/29148-2011.html>. (accessed 11 May 2017).
[18] K. Wiegers, J. Beatty, Software Requirements, third ed., Microsoft Press, Redmont, Washington, 2013.
[19] D. Lindoerfer, U. Mansmann, A comprehensive assessment tool for patient registry software systems: the CIPROS checklist, Methods Inf. Med. 54 (5) (2015) 447–454, http://dx.doi.org/10.3414/ME14-02-0026 (Epub 2015 Sep 30).
[20] D.G. Altman, K.F. Schulz, D. Moher, M. Egger, F. Davidoff, D. Elbourne, P.C. Gøtzsche, Lang T; CONSORT GROUP (Consolidated Standards of Reporting Trials). The revised CONSORT statement for reporting randomized trials: explanation and elaboration, Ann. Intern. Med. 134 (2001) 663–694.
[21] D. Lindoerfer, U. Mansmann, Elaboration of the CIPROS Checklist with Items for a Patient Registry Software System: Examples and Explanations. J Biomed Inform. Data in Brief, submitted for publication.
[22] P. Mayring, Qualitative Content Analysis. Forum Qualitative Sozialforschung / Forum: Qualitative Social Research, vol. 1, no. 2, Art. 20, June 2000. ISSN 1438-5627. <http://www.qualitative-research.net/index.php/fqs/article/view/1089/2386>. (accessed 11 May 2017).
[23] National Cancer Institute. The caBIG program has been retired. <https://cabig.nci.nih.gov/>. (accessed 12 Feb 2015).
[24] National Cancer Institute. National Cancer Informatics Program, 2015. <http://cbiit.nci.nih.gov/ncip> (accessed 11 May 2017).
[25] TMF e.V., 2015. <http://www.tmf-ev.de/> (accessed 11 May 2017).
[26] K.E. Wiegers. Software Requirements Specification. <https://www.google.de/url?sa=t&rct=j&q=&esrc=s&source=web&cd=2&ved=0ahUKEwiTzOqX_tvKAhUBshQKHZuhCvoQFggnMAE&url=https%3A%2F%2Fweb.cs.dal.ca%2F~hawkey%2F3130%2Fsrs_template-ieee.doc&usg=AFQjCNEfUo1OVJWXtibKolXDi5mOQ6q1Gg&bvm=bv.113034660,d.bGQ&cad=rja> (accessed 11 May 2017).
[27] A.D. McKinnon. Software Requirements Specification Template. <http://users.tricity.wsu.edu/~mckinnon/cpts322/cpts322-srs-v1.doc> (accessed 11 May 2017).
[28] PRISMA, Transparent Reporting of Systematic Reviews and Meta-Analyses, 2013. <http://www.prisma-statement.org/> (accessed 11 May 2017).
[29] D. Moher, A. Liberati, J. Tetzlaff, D.G. Altman, The PRISMA group. Preferred reporting items for systematic reviews and meta-analyses: the PRISMA statement, PLOS Med. 6 (2009) 7.
[30] B.C. Drolet, K.B. Johnson, Categorizing the world of registries, J. Biomed. Inform. 41 (6) (2008) 1009–1020, http://dx.doi.org/10.1016/j.jbi.2008.01.009 (Epub 2008 Feb 5).
[31] Herman J. Ader, Gideon J. Mellenbergh, David J. Hand, "Methodological quality". Advising on Research Methods: A consultant's companion, Johannes van Kessel Publishing, 2008, ISBN 978-90-79418-02-2.

Appendix C: A Comparison of the CIPROS Checklist Items with related Aspects or Topics mentioned in the ISO/IEC/IEEE 29148 Systems and Software engineering – Life cycle processes - Requirements engineering - International Standard and some Software Requirement Specification (SRS) Templates.

CIPROS Checklist (2015)		**(IEEE-830-1984) Newest version: ISO/IEC/IEEE 29148 (2011)**	**Wiegers & Beatty Microsoft Press 2013**	**Mc Kinnon SRS Template (2005) (Based on IEEE Std. 830-1984)**	**Robertson Volere RS Template (1995-2015)**
Aspect / Topic	**Item No.**	**Related Aspects or Topics mentioned in paragraphs in the appropriate lists or Templates**			
Software architecture	**1**		**2.3 Operating Environment**		
System Architecture	1.1		2.3 Operating Environment		
Platform independence	1.2		2.3 Operating Environment	3.5.6 Portability	
Open Source	1.3				
Development	**2**				
Design model	2.1				
Framework-based design	2.2				
Questionnaire builder	2.3				
Usability testing	2.4	9.4.5 Usability requirements 9.5.12 Usability requirements	6.1 Usability		11 Usability and Humanity Requirements
Performance testing	2.5	9.4.6 Performance requirements 9.5.13 Performance requirements	5.4 Communications interfaces data transfer rates, handshaking, and synchronization mechanisms, 6.2 Performance	3.5.1 Performance	12 Performance Requirements
Interfaces and Interoperability	**3**	**9.4.7 System interfaces 9.5.3.1 System interfaces 9.5.10 External interfaces**	**5 External Interface Requirements.**	**3.1 External Interface Requirements**	**13c Req. for Interfacing with Adjacent Systems**

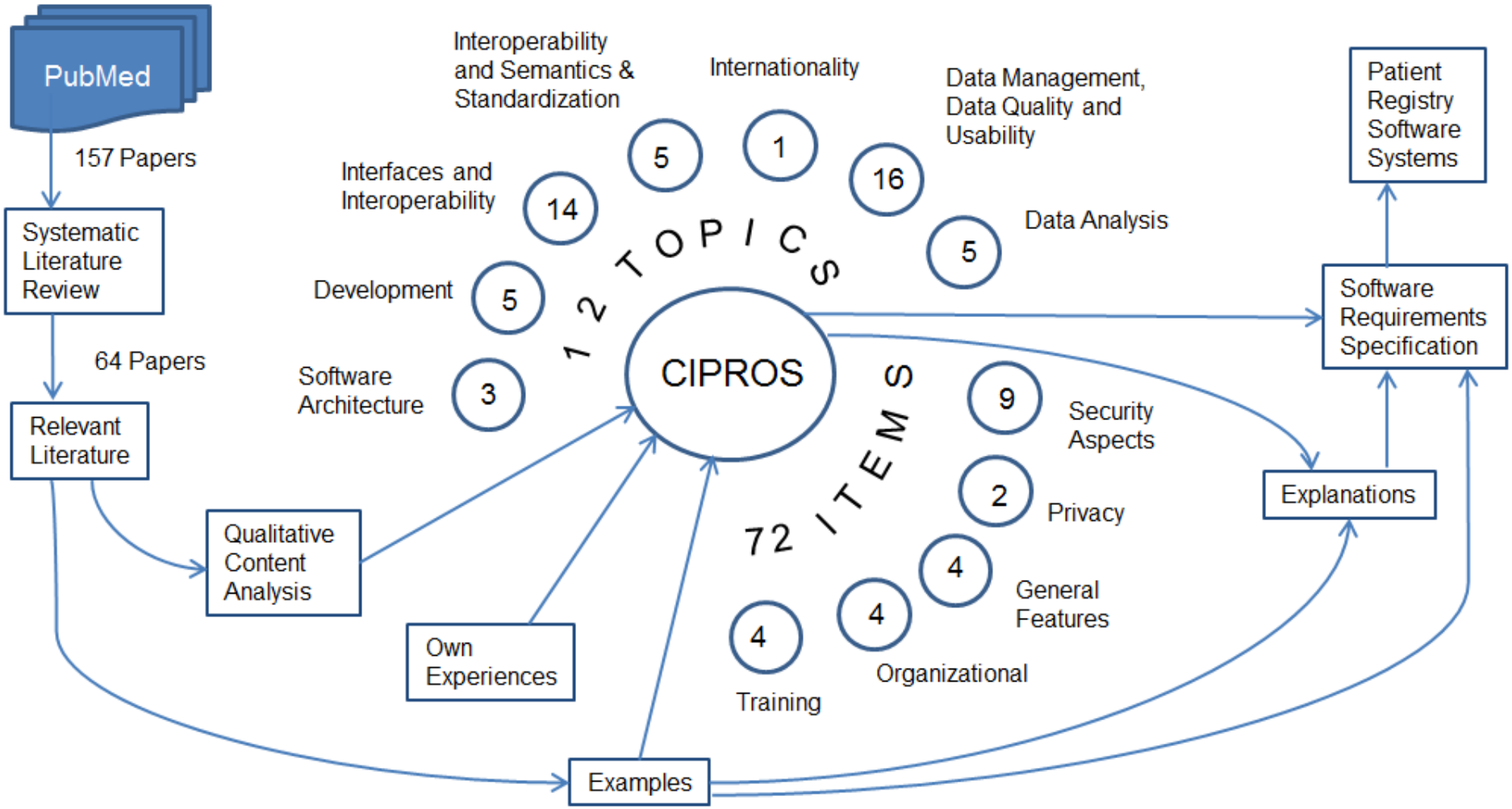
CIPROS Checklist (2015)		**(IEEE-830-1984) Newest version: ISO/IEC/IEEE 29148 (2011)**	**Wiegers & Beatty Microsoft Press 2013**	**Mc Kinnon SRS Template (2005) (Based on IEEE Std. 830-1984)**	**Robertson Volere RS Template (1995-2015)**
Aspect / Topic	**Item No.**	**Related Aspects or Topics mentioned in paragraphs in the appropriate lists or Templates**			
End-User Interfaces		**9.5.3.2 User interfaces**	**5.1 User interfaces 5.4 Communi-cations interfaces**	**3.1.1 User Interfaces**	
Web Interface	3.1		5.4 Communi-cations interfaces		
Compatibility	3.2				
Email-alert	3.3		5.4 Communi-cations interfaces		
Messaging interface	3.4				
Online discussion forum	3.5				
Mobile interface	3.6		5.3 Hardware interfaces	3.1.2 Hardware Interfaces	
Patient interface	3.7				
Programming interface		**9.5.3.4 Software interfaces**	**5.2 Software interfaces**	**3.1.3 Software Interfaces**	
Third party access	3.8				
API for inserting data	3.9				
API for retrieving data	3.10				
Data update mechanism	3.11				
Interfaces to other systems		**9.4.7 System interfaces 9.5.3.4 Software interfaces 9.5.10 External interfaces**	**5.2 Software interfaces**	**3.1 External Interface Requirements**	
Interface to HIS / CIS	3.12				
Integration of	3.13				

CIPROS Checklist (2015)		**(IEEE-830-1984) Newest version: ISO/IEC/IEEE 29148 (2011)**	**Wiegers & Beatty Microsoft Press 2013**	**Mc Kinnon SRS Template (2005) (Based on IEEE Std. 830-1984)**	**Robertson Volere RS Template (1995-2015)**
Aspect / Topic	**Item No.**	**Related Aspects or Topics mentioned in paragraphs in the appropriate lists or Templates**			
biological data					
Extensibility is possible	3.14		2.5 Assumptions and dependencies		
Interoperability and Semantics & Standardization	**4**	**9.5.16 Standards compliance**			**17b Standards Compliance Req.**
CRFs	4.1				
Data	4.2				
Metadata	4.3		4.2 Data dictionary		
Vocabularies	4.4				
XML Schema	4.5				
Internationality	**5**		7. Internationali-zation and localization requirements		11b. Personalization and Internationali-zation Requirements
Multilingualism	5.1	9.4.14 Policies and regulations, multilingual support			
Data management, data quality and usability	**6**		**4. Data requirements**		
Pseudonymous patient identifier	6.1				
CRF is divided in parts	6.2				
Customizable CRF parts	6.3				
Minimal and extended dataset	6.4				
All data types are supported	6.5				

CIPROS Checklist (2015)		**(IEEE-830-1984) Newest version: ISO/IEC/IEEE 29148 (2011)**	**Wiegers & Beatty Microsoft Press 2013**	**Mc Kinnon SRS Template (2005) (Based on IEEE Std. 830-1984)**	**Robertson Volere RS Template (1995-2015)**
Aspect / Topic	**Item No.**	**Related Aspects or Topics mentioned in paragraphs in the appropriate lists or Templates**			
Special data types are possible	6.6				
Multiple choice is used	6.7				
No predefined selection	6.8				
Data validation components	6.9				
Data query tool	6.10				
Interface for manual data check	6.11				
Manual data queries	6.12				
Data Query Flags	6.13				
Plausibility Flags	6.14				
Insertion of unplanned visits	6.15				
Software ergonomics	6.16	9.4.8.1 Human system integration requirements, Ergonomics			
Data Analysis	**7**			**4. Analysis Models**	
Query builder for researchers	7.1				
Report generation	7.2		4.3 Reports		
Download of datasets	7.3				
Graphical presentation of results	7.4			4.1 Sequence Diagrams 4.2 Data Flow Diagrams (DFD) 4.2 State-Transition Diagrams (STD)	

CIPROS Checklist (2015)		**(IEEE-830-1984) Newest version: ISO/IEC/IEEE 29148 (2011)**	**Wiegers & Beatty Microsoft Press 2013**	**Mc Kinnon SRS Template (2005) (Based on IEEE Std. 830-1984)**	**Robertson Volere RS Template (1995-2015)**
Aspect / Topic	**Item No.**	**Related Aspects or Topics mentioned in paragraphs in the appropriate lists or Templates**			
Risk Analysis	7.5				
Security aspects	**8**	**9.4.12 System security**	**6.3 Security** **6.4 Safety**	**3.5.4 Security**	**15 Security Requirements**
Authorized users	8.1				15a Access Req.
Role-based access	8.2				
Encrypted data transfer	8.3		5.4 Communi-cations Interfaces, communication security or encryption issues		
Encrypted data storage	8.4		5.4 Communi-cations Interfaces, communication security or encryption issues		
Audit trail	8.5	9.5.16 Standards compliance, audit tracing	8. Other, audit trail		15d Audit requirements
Master-Slave replication	8.6			3.5.2 Reliability	
Backup management	8.7	9.4.13 Information management, Backup management	4.4 Data acquisition, integrity, retention, and disposal	3.5.2 Availability	
Firewall	8.8				
Server room	8.9				
Privacy	**9**	**9.4.12 System security, privacy requirements**	**6.3 Security, privacy issues**		**15c Privacy Requirements**
Data Protection concept	9.1				
Double pseudonymi-zation	9.2				
General Features	**10**				

CIPROS Checklist (2015)		(IEEE-830-1984) Newest version: ISO/IEC/IEEE 29148 (2011)	Wiegers & Beatty Microsoft Press 2013	Mc Kinnon SRS Template (2005) (Based on IEEE Std. 830-1984)	Robertson Volere RS Template (1995-2015)
Aspect / Topic	**Item No.**	**Related Aspects or Topics mentioned in paragraphs in the appropriate lists or Templates**			
Costs	10.1				24 Costs
Multi-client capability	10.2				
Update mechanism	10.3				13e Release Req. 13f Backwards Compatibility Req.
Source documentation in pdf	10.4				
Organizational	**11**				
Compliance with regulations	11.1	9.4.14 Policies and regulations 9.5.6 Limitations, a) Regulatory policies	8. Other requirements, Legal, regulatory		17a Legal Compliance Requirements
Informed Consent	11.2				15c Privacy Requirements
Rights on the data	11.3				
Data protection guidelines	11.4		6.3 Security, privacy policies and regulations		
Training	**12**				**25 User Documentation and Training**
User manuals	12.1				25a User Documentation Requirements
User training	12.2				25b Training Requirements
User feedback	12.3				
Online help	12.4				


PubMed
157 Papers
Systematic Literature Review
64 Papers
Relevant Literature
Qualitative Content Analysis
Own Experiences
Examples
CIPROS
12 TOPICS
72 ITEMS
Interoperability and Semantics & Standardization
5
Internationality
1
Data Management, Data Quality and Usability
16
Interfaces and Interoperability
14
Data Analysis
5
Development
5
Software Architecture
3
Security Aspects
9
Privacy
2
General Features
4
Organizational
4
Training
4
Patient Registry Software Systems
Software Requirements Specification
Explanations

Anhang A

Publikation III

Lindoerfer, D. and Mansmann, U. (2017). Data for the elaboration of the CIPROS checklist with items for a patient registry software system: Examples and explanations. *Data Brief*, 14:494-497. (CC-BY 4.0)

https://doi.org/10.1016/j.dib.2017.07.075

https://www.sciencedirect.com/science/article/pii/S2352340917303773?via%3Dihub

Im Anschluss:

Ebenfalls online:

Appendix von Publikation III:

Die CIPROS Checkliste gefolgt von den Literaturbeispielen und Erklärungen zu den Items.

Data in Brief 14 (2017) 494–497

Contents lists available at ScienceDirect

Data in Brief

journal homepage: www.elsevier.com/locate/dib

Data Article

Data for the elaboration of the CIPROS checklist with items for a patient registry software system: Examples and explanations

Doris Lindoerfer *, Ulrich Mansmann

Institut für medizinische Informationsverarbeitung, Biometrie und Epidemiologie (IBE), Ludwig-Maximilians-Universität München, Marchioninistraße 15, D-81377 München, Germany

ARTICLE INFO

Article history:
Received 19 May 2017
Received in revised form
20 July 2017
Accepted 31 July 2017
Available online 3 August 2017

Keywords:
Checklist
Patient registry software systems
Examples and explanations

ABSTRACT

The data presented relates to the publication "Enhancing Requirements Engineering for Patient Registry Software Systems with Evidence-based Components" (Lindoerfer and Mansmann, 2017) [1], which describes the strategy behind the development of the CIPROS checklist. This manuscript also compares CIPROS with general requirements specification templates, and standards. The data is shortly described in Section 2.4 and presented in Appendix A. The *examples* represent the material extracted from the literature used in qualitative analysis. The *explanations* summarize the example contents from which the CIPROS checklist was created.

Patient registries are a crucial part of medical research. High quality registries use efficient information systems software selected from a wide variety of existing software solutions.

An efficient selection process requires focused selection criteria. The evidence-based CIPROS checklist [2] accelerates this requirements engineering process.

CIPROS was developed in a multistep procedure: (1) A systematic literature review provided an exhaustive collection of relevant publications (64 articles), (2) a catalogue of relevant criteria was derived by a qualitative content analysis, and (3) the checklist containing 72 items was composed which provides a minimal appraisal standard.

The data presented per checklist item provide the relevant textual information (examples) and a first qualitative summary (explanation).

DOI of original article: http://dx.doi.org/10.1016/j.jbi.2017.05.013
* Corresponding author.
E-mail addresses: lindoerf@ibe.med.uni-muenchen.de (D. Lindoerfer), mansmann@ibe.med.uni-muenchen.de (U. Mansmann).

http://dx.doi.org/10.1016/j.dib.2017.07.075

The examples and explanations provide the background information on CIPROS. They elucidate how to implement the checklist items in other projects. The literature list and the selected texts serve as a reference for scientists and system developers.

Specifications Table

Subject area	*Medical informatics*
More specific subject area	*Patient Registry Software Systems*
Type of data	*Examples and explanations from which the CIPROS Checklist items are derived and which elaborate the CIPROS Checklist items demonstrative and detailed.*
How data was acquired	*Examples are chosen from the literature, explanations are created by the authors.*
Data format	*Text*
Experimental factors	*Examples are cited from reference papers, explanations are created by the authors.*
Experimental features	*Examples and explanations may inspire scientists and system developers how to implement the CIPROS checklist items in own projects. Examples and explanations of each Item of the CIPROS checklist can serve as reference book for scientists and system developers how to implement the items in own projects and systems.*
Data source location	*Examples are cited from the reference papers. Explanations are created by the authors. Reference papers are cited.*
Data accessibility	*The data are part of this article they are presented in Appendix A. We linked the respective CIPROS checklist Items directly to the examples and explanations in the elaboration part. This is a perfect way to make the elaboration part of the CIPROS checklist items directly accessible to the readers in a comfortable way.*

Value of the data

- A collection of references and examples how patient registries are implemented and used in the medical research community.
- Examples and explanations represent a wide range of practices, and provide the practical background how to implement CIPROS items in projects.
- The examples provide the raw data used to derive the CIPROS checklist.
- The structured collection of examples supports decision makers and developers to formulate project requirements and explain how to implement CIPROS.
- The data presented are a reference book for scientists and system developers on how to implement the features in their own projects.

1. Data

CIPROS is the result of a qualitative text analysis. This paper presents the qualitative data used for this process. It was created via a systematic review (see Section 2.1) which identified 64 relevant publications. These were read and qualitative information on relevant aspects was extracted. Therefore the data consists of citations from the analyzed papers (referred to as examples) and represent information describing individual aspects of systems and software structures for patient

registries. From the text analysis we identified 72 relevant items and the corresponding data is shown. The data also contains qualitative summaries (called explanations) based on the citations per item. The text summaries were used to formulate the item content.

2. Experimental design, materials and methods

For the development of the CIPROS checklist we used qualitative content analysis (QCA) methods developed by Mayring [3] for social research, and adapted them to create a checklist which supports requirements extraction for patient registry software systems. To create the CIPROS checklist we performed three steps which are described below:

1) A systematic literature review
2) A qualitative content analysis
3) Creation of an early version of the checklist

2.1. Systematic literature review

A systematic literature review was performed in PubMed to find papers on patient registry software systems. This guarantees that the software systems are used in real-life medical research projects. The following search in PubMed was performed at 17th of January 2014: *"(registry or registries) AND (eCRF or EDC or CDMS or CTMS or web) AND (software or open-source or open source or Java)"*. It was updated at 15st of February 2015.

The search terms were chosen in an iterative process: We are interested in registries with electronically Case Report Forms (eCRFs) or Electronic Data Capture (EDC) technology or Clinical Data Management Systems (CDMS) or Clinical Trial Management Systems (CTMS). We were mainly interested in software systems or open-source systems. We also included Java in our search terms.

2.2. Qualitative content-analysis

A qualitative content analysis according to Mayring [3] was performed. In a first step a process of inductive category development [3] was performed. We looked at titles and abstracts if there was a description of software related features. We looked at papers published in English or German. For respective papers, the full-texts were analyzed. Papers were considered for further analysis if a system description or at least a short system description was provided. We searched the papers for passages describing relevant software-specific features. We extracted these phrases describing software systems and built inductive categories out of the material, categorized the items or formulated new categories. After reading about 10–50% of the papers we revised the categories. Then we read the remaining papers to find additional categories for system's features.

2.3. Creation of an early version of the item list for the CIPROS checklist

To create the checklist we performed the step model of deductive category application described by Mayring [3]. Compiling the phrases retrieved from the qualitative content analysis we created a summary of content and defined the items. Within a feedback loop we revised these categories and eventually reduced them to main categories. If we found a new feature, we added a new item within the respective category. A new item was added for each new feature, regardless how often we found the feature. We enhanced the checklist with items which we found in none of the papers, but we considered as important based on our own experiences.

The CIPROS checklist consists of 72 items organized within 12 topics which relate to features in system components (S), functional aspects (F), or design steps (D).

2.4. Explanation and elaboration of the CIPROS checklist items

The data are presented in Appendix A. The CIPROS Checklist items are linked to the respective elaboration data, the examples and explanations.

For some items, we do not provide a reference and cannot provide an example, as these items were not discussed in the literature that we found in our SLR. However, they are considered relevant based on our own experience and theoretical reasoning (items no. 6.10, 6.12, 6.15, 9.2).

Acknowledgements

We would like to thank PD Dr. Eva Rehfuess for their discussion and intellectual contribution in performing the systematic literature review and Dr. Michaela Coenen for their discussion and intellectual contribution in performing and describing the qualitative content analysis of the literature. Furthermore we would like to thank Kathrin Halfter for her contribution in English polishing of this manuscript.

Transparency document. Supplementary material

Transparency data associated with this article can be found in the online version at http://dx.doi.org/10.1016/j.dib.2017.07.075.

Appendix A. Supporting information

Supplementary data associated with this article can be found in the online version at http://dx.doi.org/10.1016/j.dib.2017.07.075.

References

[1] D. Lindoerfer, U. Mansmann, Enhancing requirements engineering for patient registry software systems with evidence-based components, J. Biomed. Inform. 71 (2017) 147–153. http://dx.doi.org/10.1016/j.jbi.2017.05.013 (Epub 2017 May 20).
[2] D. Lindoerfer, U. Mansmann, A comprehensive assessment tool for patient registry software systems: the CIPROS checklist, Methods Inf. Med. 54 (5) (2015 12) 447–454. http://dx.doi.org/10.3414/ME14-02-0026 (Epub 2015 Sep 30).
[3] P. Mayring, Qualitative content analysis, Forum Qual. Sozialforschung/Forum: Qual. Soc. Res. 1 (2) (2000) (ISSN 1438-5627) ⟨http://www.qualitative-research.net/index.php/fqs/article/view/1089/2386⟩.

Appendix A: The Checklist with Items for Patient Registry Software Systems (CIPROS) linked with the related data, examples and explanations for the elaboration of the CIPROS Checklist items.

Table 1. CIPROS - A Checklist of Items to consider when choosing or developing a software system for patient registries. [Strg+Clicking on the Item No. will jump to the respective elaboration part of the Checklist Item.]

Aspect / Topic	Item No.	Description	Relevant References
Software architecture	**1**	**This topic contains items related to the architecture of the patient registry software system.**	
System Architecture	**1.1**	The system has a modular multi-tier architecture.	[1], [2], [3], [4], [5]
Platform independence	**1.2**	The system runs on different platforms.	[6], [7]
Open Source	**1.3**	Open source components are used to create the software of the patient registry system and it is made open source.	[8]
Development	**2**	**This topic comprises aspects which are important during the development process of a new registry project.**	
Design model	**2.1**	The system itself is developed following a design model.	[9]
Framework-based design	**2.2**	The system provides a framework for the development process of a new registry project.	[1], [5]
Questionnaire builder	**2.3**	The system has a table-based or web-based questionnaire builder.	[1], [9]
Usability testing	**2.4**	Usability of the system is tested. Users get involved in the development process early.	[9], [10], [11]
Performance testing	**2.5**	The performance of the system is tested.	[12]
Interfaces and Interoperability	**3**	**This topic comprehends different kinds of interfaces of the registry software system and related aspects of these interfaces.**	
End-User Interfaces		**This subtopic contains interfaces to end-users and related aspects.**	
Web Interface	**3.1**	The patient registry software system has a web interface.	[4], [6], [11]
Compatibility	**3.2**	The web interface is compatible with the major web browsers.	[4]
Email-alert	**3.3**	Email alerts are possible, as reminders for follow-up, etc.	[15]

Aspect / Topic	Item No.	Description	Relevant References
Messaging interface	**3.4**	There is a messaging interface to provide information for the end-users.	[1]
Online discussion forum	**3.5**	An online discussion forum for the end-users is established.	[1]
Mobile interface	**3.6**	An interface for mobile devices is available.	[1], [4], [16]
Patient interface	**3.7**	The system provides also a patient interface, where quality of life (QOL) and other information can be collected.	[9]
Programming interface		**This subtopic describes programming interfaces to other patient registries and related aspects.**	
Third party access	**3.8**	Programmatic access to data from an external resource is possible.	[1], [4]
API for inserting data	**3.9**	The system provides an application programming interface (API) for inserting data automatically.	[1], [4], [17], [18]
API for retrieving data	**3.10**	The system provides an API for retrieving data.	[4], [9], [18]
Data update mechanism	**3.11**	There is an update mechanism for automatically inserted data.	[5]
Interfaces to other systems		**This subtopic contains Interfaces to other systems.**	
Interface to HIS / CIS	**3.12**	An interface to HIS / CIS[1] (HL7)[2] is available to exchange data.	[9], [17]
Integration of biological data	**3.13**	The system has an interface to integrate pseudonymized biological data (see item 6.1). See for example the concepts of the TMF e.V. [19] or an anonymizing tool like described in Prasser et al. [20].	[4]
Extensibility is possible	**3.14**	It is possible, if necessary, to create further interfaces to other systems.	[5]
Interoperability and Semantics & Standardization	**4**	**This topic contains issues of interoperability with focus on semantic and standardization aspects.**	**[21], [22]**
CRFs	**4.1**	Standardized CRFs are used whenever possible	[1], [4]
Data	**4.2**	Standardized Data are used whenever possible.	[4]
Metadata	**4.3**	Ontology-based, standardized metadata are used.	[4], [17]
Vocabularies	**4.4**	Ontology-based, standardized vocabularies are used.	[4], [8]

[1] Hospital Information System (HIS) / Clinical Information System (CIS)
[2] Health Level Seven (HL7), http://www.hl7.org/

Aspect / Topic	Item No.	Description	Relevant References
XML Schema	**4.5**	An XML[3] schema definition (XSD) is available for structured data exchange.	[1], [2]
Internationality	**5**	**This topic contains an item which a registry software system can provide to support international cooperation in a registry project.**	
Multilingualism	**5.1**	The whole questionnaire life-cycle can be displayed in different languages.	[9], [15]
Data management, data quality and usability	**6**	**This topic contains important items which should be considered by the data management to support data quality and usability of the registry.**	
Pseudonymous patient identifier	**6.1**	A pseudonymous patient identifier (PID) is created by the system.	[4], [23]
CRF is divided in parts	**6.2**	The CRF is divided in logical parts.	[4]
Customizable CRF parts	**6.3**	CRFs are customizable according to the user's selections.	[1]
Minimal and extended dataset	**6.4**	If requested, a minimal and extended dataset can be used.	[11]
All data types are supported	**6.5**	The system supports the use of all common data types.	[5]
Special data types are possible	**6.6**	Special data types, like images, X-rays, and links can also be stored.	[5]
Multiple choice is used	**6.7**	Multiple choice data collection is used whenever possible.	[15], [24]
No predefined selection	**6.8**	No predefined selection is used, to avoid unwanted entries.	[10]
Data validation components	**6.9**	The system has implemented data validation components (Hard- and Soft Checks).	[1], [4], [11], [18], [25]
Data query tool	**6.10**	The system has a query tool to perform automatic data queries.	
Interface for manual data check	**6.11**	The system has an interface for manual data check.	[26]
Manual data	**6.12**	It is possible to perform manual data queries within the system.	

[3] Extensible Markup Language (XML), https://www.w3.org/XML/

Aspect / Topic	Item No.	Description	Relevant References
queries			
Data Query Flags	**6.13**	Entries with unresolved queries are marked with flags at different levels (item, part, patient, etc.).	[23]
Plausibility Flags	**6.14**	Implausible entries are marked with flags at different levels.	[25]
Insertion of unplanned visits	**6.15**	Unplanned visits can flexibly be integrated.	
Software ergonomics	**6.16**	The system should be designed following the standards of software ergonomics, defined in ISO 9241-110 [27].	[9]
Data Analysis	**7**	**This topic contains issues with which the registry software system can support data analysis.**	
Query builder for researchers	**7.1**	The system has a query builder to assist researchers to select interesting patient cohorts.	[8]
Report generation	**7.2**	The system is able to generate reports for selected cohorts.	[8]
Download of datasets	**7.3**	Datasets can be generated and downloaded for analysis in different formats, complete or for selected cohorts.	[6], [8], [10]
Graphical presentation of results	**7.4**	Results can be presented as colored graphs in real time.	[8], [9]
Risk Analysis	**7.5**	The system gives interactive feedback, classifying the patient through to implemented knowledge bases or scoring systems.	[9], [16], [23]
Security aspects	**8**	**This topic contains security aspects of the registry software system and important security aspects of the registry operation process.**	
Authorized users	**8.1**	Only authorized users have access to the data.	[1], [4]
Role-based access	**8.2**	The system provides role-based user access.	[4], [11]
Encrypted data transfer	**8.3**	The system utilizes secure web server communication through encrypted data transfer.	[1], [4], [6], [11]
Encrypted data storage	**8.4**	Sensitive data can be stored encrypted in the database.	[1], [4]
Audit trail	**8.5**	All changes in the database are tracked and monitored through an audit trail.	[4], [5], [15], [25]
Master-Slave replication	**8.6**	If necessary, master-slave replication should be established.	[1]
Backup management	**8.7**	Backups are stored separately regularly.	[1]
Firewall	**8.8**	The server is behind a firewall.	[26]

Aspect / Topic	Item No.	Description	Relevant References
Server room	**8.9**	The server room is locked and temperature controlled.	[3], [4]
Privacy	**9**	**This topic describes privacy aspects of the registry and of the software system additional to items 6.1 and 11.4.**	
Data Protection concept	**9.1**	A data protection concept should be established before starting a registry project.	[11]
Double pseudonymization	**9.2**	The registry software system should provide double pseudonymization for biological and genomic data.	
General Features	**10**	**This topic contains general items which have no direct relation to the other topics.**	
Costs	**10.1**	The costs of a patient registry software system must be taken into consideration, when choosing a software system: Is there a realistic calculation of the costs considering the costs of the procurement / programming, operation, life-cycle, archive, system replacement? Are the financial resources sufficient regarding the anticipated costs of sustainability of the registry guaranteed?	[28], [29]
Multi-client capability	**10.2**	The system has a multi-client capability. Several projects can simultaneously be executed in one installation of the system, but should be strictly separated.	[8]
Update mechanism	**10.3**	An update mechanism for the system is in place.	[9], [13]
Source documentation in pdf	**10.4**	Source documentation of CRFs in pdf format is possible.	[12]
Organizational	**11**	**This topic comprises software-related organizational items.**	
Compliance with regulations	**11.1**	The system is compliant with all known relevant regulations.	[11]
Informed Consent	**11.2**	The registry is compliant with Chapter 11, Title 21, Code of Federal Regulations and HIPPA.	[4], [28]
Rights on the data	**11.3**	There are clearly defined rules which describe the rights on the data for each participating institution.	[4], [6], [11]
Data protection guidelines	**11.4**	The system is compliant with all known appropriate data protection guidelines of the registry project.	[11]
Training	**12**	**This topic contains items which are important for the user training.**	
User manuals	**12.1**	There should be manuals for the registry end-users and for the operators.	[4], [12]

Aspect / Topic	Item No.	Description	Relevant References
User training	**12.2**	At the beginning, and if necessary during the project time, a training is provided for the users.	[4], [17]
User feedback	**12.3**	Regularly user feedback is collected for further improvements of the system.	[4], [30], [31]
Online help	**12.4**	The system provides an online help for data entry.	[10]

Explanation and Elaboration of the CIPROS Checklist Items

1. Software Architecture (S)

This topic contains items related to the architecture of the patient registry software system.

Item 1.1 The system has a modular multi-tier architecture.

Examples

> "The CNRDS is implemented using a JavaEE three-tier, web-based architecture: 1) a web interface that interacts with the user, 2) a middle tier that contains the application's business logic, and 3) an integration tier that consists of the enterprise resources. The CNRDS runs on an Apache Tomcat web server and has a MySQL cluster as the back-end database." [1]
>
> "The architecture of CEMARA is based on a n-tier architecture. Via a web browser, the client tier connects to the middle tier, which is connected to several databases: the production database, the geographical dictionary database, and the thesaurus database. A data warehouse and a geographical information system allowing queries and representation are in progress. Their framework is close to an already available application for end-stage renal disease. The middle tier supports client services through Web containers and business logic services through component containers. Business logic in the middleware is interfaced with a SGBD dependant handler which supports the transactions toward the database. At the client side, CEMARA relies on existing local Internet networking facilities and on a widely spread computer configuration in medical settings. Access via a personal digital assistant is also available." [2]
>
> "The PCCR has been implemented using a three-tier, Web-based architecture: i) a client that interacts with the user; ii) an application server that contains the business logic of the application; and iii) a resource manager that stores the data. The PCCR utilizes Java Servlet/JSP technology and has Oracle 10g database as a back-end." [3]
>
> "The BCCR is a multi-tier web application that utilizes Java Servlet/JSP technology and has an Oracle 11g database as a back-end." [4]
>
> "So far, we have defined six optional modules, but the modular structure is easily extensible to fit on other needs. Out of those, five have already been implemented." [5]

Explanation

The system should be implemented as a multi-tier architecture or at least three-tier architecture to separate the business-logic from the database at the backend and the web server at the frontend [1], [2], [3], [4]. The data should be stored in one or more relational databases. The system should be

modular and extendable as the system presented by Deserno et al. [5] so that additional components and further developments can easily be integrated.

Item 1.2 The system runs on different platforms.

Examples

> "The FRB! software is a web-based application, which was developed using freely available software such as Apache, MySQL, PHP, and RubyonRails. This approach allows the application to be run on different server operating systems. Any device with Internet access and a recent browser can be used to interact with the application." [6]

> "**Database architecture**
>
> CDKD was designed as client–server architecture with PHP5 and MySQL, running on an Apache server. PHP, MySQL and Apache technology were preferred as they are open-source components and platform independent. CDKD is hosted on both windows and Linux platform." [7]

Explanation

In some cases it might be an advantage if the software runs on different platforms. If it is planned to run the system on different platforms or if it is not clear if this feature will be needed in the future, this item should be considered.

Item 1.3 Open source components are used to create the software of the patient registry system and it is also made available as open source.

Example

> "**Codebase**
>
> The i2b2-SSR code is available as open source software, licensed under LGPL version 3 for i2b2-SSR components; constituent components and dependencies are available under their respective open source licenses (repository and links at https://open.med.harvard.edu/display/CARRANET). The Webserver package is available from Cincinnati Children's Hospital Medical Center at https://bmi.cchmc.org/svn/i2b2/i2b2/public/." [8]

Explanation

To enable full interoperability beside semantics and standardization of metadata and vocabularies, open source components should be used to create the system and the code of the software system can be made open source as stated in the above example from Natter et al. [8]. The National Cancer Institute supports with the National Cancer Informatics Program [NCI NCIP] interoperability through providing source code, semantics and standardization of CRFs, vocabularies and metadata.

2. Development

This topic comprises aspects which are important during the development process of a new registry project.

Item 2.1 The system itself is developed following a design model.

Example

> "***Development process***

For this project Extreme Programming (XP) [28] has been adopted as software development approach. XP is considered a flexible approach to planning, making it well suited to react on changed requirements during the course of the project. As suggested in XP, the CHES development team adopted an incremental software architecture to match the demands of the various user groups. More specifically, the software architecture was not fixed upfront, but rather evolved over time. For this purpose, periodic releases and resulting feedback allowed for driving the development of CHES in a direction, which ensured the acceptance in daily clinical routine." [9]

Explanation

The software of the patient registry system should be developed following a design model to avoid ad hoc implementation and unstructured extensions.

Item 2.2 The system provides a framework for the developmental process of a new registry project.

Examples

"Nephrologists from the Registry Board of Advisors participated in the development of the CRFs in an effort to delineate the current standard of care of the patients with kidney disease and to facilitate consistent, thorough, and precise patient evaluations throughout the nephrology community. Nephrologists were given a Microsoft Excel spreadsheet for entering the required metadata information (e.g., field name, data type, value range, and presentation style) about each measurement in each case report form. An IT support team staff collected worksheets via electronic mail and then built web-based electronic forms for data collection. A hyperlink to the prototype application was given to the nephrologists along with instructions for testing and further iteration of the metadata spreadsheet. When feedback returned, the IT support team edited the electronic forms and republished them. The CNRDS utilizes CapitalBio electronic medical records (EMR) system's (CapitalBio Corporation, Beijing, China) dynamic case report form (dCRF) technology to enable modification and publishing of the CRFs on-the-fly. The case report forms created from the CNRDS are presented in Table 1. Additionally, the following administrative data elements are stored along with the CRFs: date when the case was submitted, registering institution code, and the person who submitted the data." [1]

"As parts of the core functionality of any RDR, we suggest five modules: Data Core, Access Control, Audit Trail, EDC, and Term. All data is stored in a relational database, but—according to the special needs of any instantiation of such a registry—the data tables, their fields, and the respective web rendering may vary." [5]

Explanation

The system should provide a framework for the development of a new registry project, for example as described by Deserno et al. [5] through specific modules. There should also be a framework for building up the project-specific content. This minimizes the need for programming skills for persons developing for example the medical content such as questionnaires, variables, metadata, range values for variables and variable coding.

Item 2.3 The system has a table-based or web-based questionnaire builder.

Examples

"Nephrologists were given a Microsoft Excel spreadsheet for entering the required metadata information (e.g., field name, data type, value range, and presentation style) about each measurement in each case report form. An IT support team staff collected worksheets via electronic mail and then built web-based electronic forms for data collection. A hyperlink to the prototype application was given to the nephrologists along with instructions for testing and further iteration of the metadata spreadsheet. When feedback returned, the IT support team edited the electronic forms and republished them. The CNRDS utilizes CapitalBio electronic medical records (EMR) system's (CapitalBio Corporation, Beijing, China) dynamic case report form (dCRF) technology to enable modification and publishing of the CRFs on-the-fly." [1]

"Questionnaire Builder: CHES Questionnaire Builder was developed for defining the structural properties (i.e., question and answer texts, or psychometric item characteristics) of questionnaires. In addition, CHES Questionnaire Builder enables researchers to define the visual appearance of questionnaires in order to adapt them to different devices, e.g., tablets, smart phones, or specific patient groups (e.g., elderly people, visually impaired)." [9]

Explanation

For the developmental process a questionnaire builder should be available, then this part can be done by persons who are more familiar with the content of the planned project and must not be done by the system developer. This reduces communication time and misunderstandings.

Item 2.4 Usability of the system is tested. Users are involved in the developmental process early on.

Examples

"Usability prototyping and usability testing ensure that users are included at an early stage (standards for user oriented software are specified in ISO 13407). It is crucial to tailor the software to the specific needs and abilities of potential users. We identified several user groups for CHES leading to specific requirements on the availability of certain features for each of them: …" [9]

"The first precondition to acquire good quality data is to pay a special attention to the GUI design. Unfortunately, physicians are not much interested in the design phase, and quite often, they let computer scientists and technicians decide for them. Being end users, physicians simply request easy and fast data entry procedures, and it is also not uncommon that they ask to replicate the same layout of their previous paper-based clinical charts. Nevertheless, a careful and shared preliminary analysis is mandatory to anticipate as much as possible any type of future exploitation of the acquired data. In fact, the impossibility to accomplish the necessary statistics is usually realized at a later time when it's too late for revising the model." [10]

"To ensure intuitive usage of the registry application, data entry staff members such as study nurses have been involved in its development at an early stage." [11]

Explanation

The usability of the system should be tested and therefore end-users must be involved in the developmental process at an early stage. This is crucial to avoid insertion errors and adding missing questions. It is also necessary to sensitize users for the content, and that all data which are required for the analysis will be captured.

Item 2.5 The performance of the system is tested.

Example

"*Performance results*

The performance was evaluated by emulating 500 simultaneous users interacting with the web application. We evaluated the performance by measuring the load times of the web frontend (as shown in Figure 6). Each user calls the website through an http request. The evaluation was performed on a client that has an Intel Core 2 Duo CPU with 2.4 GHz and 4 GB RAM, running the 64-bit Windows 7 OS, Java Runtime 1.6.21 and Apache JMeter. The web application was deployed on a Windows XP Pro OS server on an AMD Athlon 64 X2 Dual Core Processor 5200+ with 2.7 GHz and 3 GB RAM. The server has the Glassfish 3.0 Application Server. It is expected that initially only 10–50 users will access the application. This corresponds to a load time between 28 and 48 ms, which results in an excellent quality of experience of the users. Figure 7 shows the throughput and number of kilobytes per second for 10–500 users. Table I shows the detailed results of a typical user scenario, measured by FireBug, a plug-in for web browser Firefox. The user logs in on the website with his/her user credentials, downloads an empty PDF questionnaire, fills in this template and uploads the form. Later, the user reviews and analyses these data on the website. The display and download times are within the acceptable range,

but the PDF takes more time due to the generation process. In addition, the execution times for the retrieval of data analysis cases were also measured. Figure 8 shows the execution times for the audiological and neurological data for several data analysis cases. One case had a longer execution time and took 61.12 s. All other cases' execution time ranges from 0.23 to 2.54 s and are shown in Figure 8. The queries for audiological data take more time due to a higher number of variables in the queries." [12]

Table I. Website performance measurements.

Webpage	Size(kb	Time (s)	Std dev. (s)
Login page	613.7	4.036	0.957
Physician overview	623.4	3.319	1.031
Download page	624.9	3.998	1.277
PDF selection page (registr)	291.2	12.950	1.321
PDF selection page (follow)	283.2	11.924	1.421
Patient list page	670.8	5.534	1.853
Data analysis page	807.7	6.564	1.515

Explanation

The complete system architecture of a new registry project should be planned, taking into account the number of end-users who will work simultaneously with it. Before starting a new project, the performance of the patient registry software system should be tested to prevent system failures caused by system overload.

3. Interfaces and Interoperability (S)

This topic comprehends different kinds of interfaces of the registry software system and related aspects of these interfaces.

End-User Interfaces

This subtopic contains interfaces to end-users and related aspects.

Item 3.1 The patient registry software system has a web interface.

Examples

"The BCCR public website (Fig. 2) can be accessed at http://bccr.unmc.edu/." [4]

"There are two ways of entering data into the FRB! system: via a web interface or via a third-party software interface. If the User has no electronic patient management system, the independent web-based application, which can be accessed from a wide range of devices and operating systems (e.g., Windows-PC, Macintosh, tablet computer, mobile phone), can be used with a regular browser (e.g., Safari, Internet Explorer, Firefox, Opera). Additional software on the user's terminal is not required." [6]

"The CERTAIN web application, accessible via http://www.certain-registry.eu/RegApp, supports, therefore, not only the data entry, presentation, visualization, and export, but also the automatic and manual data validation. These functionalities are active at any time and location, requiring only a common web browser and internet access." [11]

Explanation

We recommend a web interface for patient registry software systems. Web-based data entry is a very comfortable and state of the art way to collect patient data for multi-center registries. All data is

immediately available in the database. The data from all centers are collected in the same way, no data-integration with the data from different centers is necessary. No additional software must be installed on the client side. When a software update is necessary it must only be installed on the server, the clients get a new version when they log in next time.

We recommend a real web-based system which requires no additional software-installation at the client-side. Compared to Lycett et al. [13] and Wake et al. [14] in their registry project software installation turned out as complicated, time consuming and prevented some groups from participating in the registry project since they failed to install the required software.

Item 3.2 The web interface is compatible with the major web browsers.

Example

> "The BCCR user interface is compatible with all major web browsers, such as Microsoft Internet Explorer 6+, Mozilla Firefox 3+, Opera, and Safari." [4]

Explanation

The system should be compatible with all common web browsers, such as Microsoft Internet Explorer, Mozilla Firefox, Sea Monkey and Safari, that most of the clients can use their familiar web browser.

Item 3.3 Email alerts are possible as reminders for follow-up, etc.

Example

> "For investigators, automated email reminders notifying follow-up times for individual patients will be sent two months before the evaluation is due. If participating families wish to opt-out of the study, completion of an End of Study form will be required.
>
> For parents/caregivers who have elected and agreed on the patient informed consent form to respond directly on-line, an automated email reminder will be sent two months prior to the next follow-up evaluation for their child, and on the day of the scheduled follow-up. If a parent/caregiver do not respond and complete the recommended evaluations for a particular time-point, the clinician will be subsequently notified by email. A notification will be sent to the clinician, once the parent has completed their data entry for that time-point." [15]

Explanation

It should be possible to send emails from the system to the end-users, mainly to send them administrative information about their usernames, passwords, etc. but also for patient management for example as reminder for follow-ups.

Item 3.4 There is a messaging interface to provide information for the end-users.

Example

> "**Online discussion forum**
>
> … The CSN uses this forum to make announcements and publish SOP documents and online surveys. …
>
> **Social networking**
>
> … We introduced two social networking features: online discussion and messaging." [1]

Explanation

The system should provide a messaging interface to provide information like SOPs, user manuals, etc., and to make announcements to the end-users.

Item 3.5 An online discussion forum for the end-users is established.

Example

> "**Online discussion forum**
>
> The web forum embedded within the CNRDS promotes active discussion among the renal community. People in China are familiar with online discussion forums. The professional online forum for kidney disease can facilitate interactions between nephrologists, including experts in the field. They use the platform to ask questions and share their medical experience and expertise. Feedback and suggestions for the CNRDS are also posted on the forum. The CSN uses this forum to make announcements and publish SOP documents and online surveys. From April 22nd, 2010 to Nov 15th, 2011, 2,441 threads composed of 9500 messages were exchanged on the forum. Table 4 shows a summary of topics and threads. At the first couple of months when the CNRDS just launched, physicians are not familiar with it, most of the posts are about how to use the system. After that, topics are more and more concentrated in the discussion of dialysis. Weekly digest of the forum posts are reported to the Board of the CSN. The online forum has been an important communication channel for nephrologists and had a positive influence on patient care." [1]

Explanation

The patient registry software system should provide an online discussion forum, to enable active discussion and interaction for the participants. As described by Xie et al. [1] in the above example, this can have a positive influence on patient care.

Item 3.6 An interface for mobile devices is available.

Examples

> "The latest generation of smartphones, such as Apple's iPhone and various Google's Android touch screen devices, are increasingly viewed as handheld computers rather than phones, due to their powerful on-board computing capability, capacious memories, large screens and open operating systems that encourage application development [30]. It is clear that the potential for mobile communication to transform healthcare and clinical intervention in the community is tremendous. Several previous studies have evaluated the use of mobile phones to support healthcare and public health interventions, notably in the collection and collation of data for healthcare research and education purposes [31,32]. Rather than developing native apps for different platforms, we built the CNRDS mobile interface using HTML5 and the jQuery mobile framework [33]. The mobile version works well on the iPhone (see Figure 3), iPad, and Android phones." [1]
>
> "To improve portability, convenience and ease of use, a separate interface for Apple's iPads has been created." [4]
>
> "The eTHR was designed as a downloadable, multiplatform, web-based application, initially for use on the iPad (Apple Computers). The user interface was built using the jQuery Mobile 1.0 Framework and was designed to save and update data in a MySQL database through asynchronous javascript requests sent over Secure Socket Layer (SSL) encryption. Data transfer, with 128-bit encryption, was designed to enable synchronous or intermittent upload of data to dedicated, secured servers within a host site." [16]

Explanation

If necessary and useful for the project an interface for mobile devices should be available, to enable data entry from distributed locations.

Especially in rural, under-resourced environments the use of mobile devices becomes a necessary tool to manage information for patient care effectively Zargaran et al. [16]. An interface to mobile

devices can help to collect distributed data and integrate them in a patient registry. Therefore a patient registry software system should provide this interface.

Item 3.7 The system also provides a patient interface where quality of life and other information can be collected.

Example

> "***Implementation***
>
> By 2012 CHES has been implemented in a number of hospital settings in various fields of medicine (e.g., urology, nephrology, orthopedics, radiation therapy, neurology, oncology, gynecology, health psychology) in Austria, Germany, Switzerland, and the UK. So far, about 5000 patients completed the computerized questionnaires with a total of approximately 15000 assessments. Its web-interface for completing questionnaires and case report forms online is currently used in an international EORTC questionnaire validation study (EORTC QLQ-TC26) in five European countries and Australia, and in a CAT validation study in Denmark.
>
> At these centers data assessment is done via tablet- PCs with screen sizes of 10" or 12". Usually one to four questions depending on screen size and patient's eyesight are shown at once. In most settings, a client–server solution for data storage is in place based on either Wi-Fi or LAN. This is necessary in order to provide results without time loss at all working stations required." [9]

Explanation

If appropriate within the project a patient interface should be provided. Some data, such as quality of life (QOL) data, can be inserted by the patients themselves. If a patient interface is provided, it is useful to provide an interface for mobile devices (Item 3.6) to collect these data as well.

Programming Interface

This subtopic describes programming interfaces to other patient registries and related aspects.

Item 3.8 Programmatic access to data from an external resource is possible.

Examples

> "**The CNRDS application programming interface (API)**
>
> Prior to the nationwide CNRDS, Shanghai, Beijing, and Zhejiang provinces established their own local renal registries. Indeed, these three registries have been running for several years and contain valuable data. In order to maximize the value of the previous investment, a RESTful web service API was developed for third party software providers to exchange their data with CNRDS. The RESTful web service was chosen due to its streaming capability and on-the-fly compression [35]. The data exchange format is a customized XML schema, which is downloadable as an additional file of this manuscript (Additional file 1). Each regional registry was assigned a unique API key for discrimination. Because the API is an interface between machines, third party software vendors should ensure that the data they submit is validated and complete. Data validation rules are also performed on the CNRDS server; an error code will return if the submission contains abnormal data." [1]
>
> "**Programming interface**
>
> BCCR end-users utilize a web interface for data entry and management. To allow third-party applications to access the BCCR's data directly and to satisfy the caBIG® bronze compatibility requirements, a set of application programming interfaces (API) has been developed." [4]

Explanation

If necessary within the project, the patient registry software system should provide an Application Programming Interface (API) to enable participants to insert and retrieve data automatically in the system. This may be an important tool for data interchange and cooperation within different institutions.

Item 3.9 The system provides an application programming interface (API) for inserting data automatically.

Examples

> "Because several local dialysis registries have been established [5,19,20] prior to the national one, the RESTful web service can be used as a data exchange interface for automatically importing data from third party registries." [1]
>
> "This set of APIs consists of methods for both retrieving data from and inserting data into the BCCR." [4]
>
> "In order to avoid duplicate data entry, the EHR from the Clinics Hospital of the University of Saõ Paulo Medical School (HCFMUSP) was integrated to the EDC through the REDCap API (Application Program Interface). The REDCap API is an interface that allows external applications to connect to REDCap remotely, and it is used for programmatically retrieving or modifying data or settings within REDCap. As the API is a built-in feature of REDCap, no installation is required and this tool implements the use of tokens as a means of authenticating and validating all API requests that are received. In addition, the API also implements data validation when the API is used for data import purposes in order to ensure that only valid data will be stored. By using the REDCap API, it was possible to retrieve useful demographic information directly from the sources of hospital systems." [17]
>
> "Data were collected using handheld Samsung (Seoul, South Korea) Galaxy Tabs with a 7-inch display running Android operating system; 50 tablets were used during the census, and 40 during the vaccination campaign. The software platform was built by a contracted partner (Majella Global Technologies, Portland, ME, USA) on Open Data Kit. External battery packs with dual USB charging ports provided a portable, backup power supply. Data records were first stored locally on devices in the field, and uploaded nightly via office Wi-Fi to a secure, web-hosted database. …
>
> Each night, data records collected in the field were uploaded from each tablet and merged into a web-hosted database." [18]

Explanation

Inserting data automatically from other registries is useful for registry projects which deal with retrospective data, or when data is captured in systems other than the registry. It enables cooperation between different institutions and prevents errors which occur if the data were inserted twice by hand. It is an innovative way to enable cooperation within different registries and it supersedes the cumbersome exchange of data, for example with excel sheets, and then imported using individual programs.

Item 3.10 The system provides an API for retrieving data.

Examples

> "This set of APIs consists of methods for both retrieving data from and inserting data into the BCCR." [4]
>
> "Data Export/Import: Sociodemographic, clinical and questionnaire data can be exported to different file formats and imported from files, e.g., SPSS or MSExcel." [9]
>
> "Online registries were subsequently downloaded nightly for analysis. …
>
> At the end of census, population data were downloaded from the webhosted electronic database and formatted in Microsoft Excel to become a dataset, or a ''lookup table''. The lookup table was loaded

back on to all tablets and embedded within the electronic forms, and served as a locally stored database from which previously collected population data could be retrieved.“ [18]

Explanation

A tool fulfilling this item should be provided by all systems, it is necessary to provide data for analysis tools such as SAS or R, also to return data to the participating institutions.

Item 3.11 There is an update mechanism for automatically inserting data.

Example

“Basically, the BLOB module ensures that large data files are safely transferred via the internet and attached to the subject’s identifier providing the date of filing. The date of transfer and the transferring person’s identifier are logged in the audit trail module. Versioning of data is possible, since the module defines a document identifier (DID) that is not unique, and a Boolean flag “latest” indicating the latest version of the respective DID.” [5]

Explanation

If the system provides automatic data import from external resources it may be useful to update already inserted data regularly. This feature should also be available to a patient registry software system. Like in the above mentioned example from Deserno et al. [5] the update of binary large objects (BLOBs) is possible, so it should also be possible to update common data.

Interfaces to other systems

This subtopic contains interfaces to other systems.

Item 3.12 An Interface to HIS / CIS (HL7) is available to exchange data.

Examples

“Interface to clinical information systems (HL7): In order to exchange medical and sociodemographic data between CHES and clinical information systems (CIS) a HL7-interface is available.” [9]

“In order to avoid duplicate data entry, the EHR from the Clinics Hospital of the University of Saõ Paulo Medical School (HCFMUSP) was integrated to the EDC through the REDCap API (Application Program Interface). The REDCap API is an interface that allows external applications to connect to REDCap remotely, and it is used for programmatically retrieving or modifying data or settings within REDCap. As the API is a built-in feature of REDCap, no installation is required and this tool implements the use of tokens as a means of authenticating and validating all API requests that are received. In addition, the API also implements data validation when the API is used for data import purposes in order to ensure that only valid data will be stored. By using the REDCap API, it was possible to retrieve useful demographic information directly from the sources of hospital systems.” [17]

Explanation

The system should provide an interface to Clinical Information Systems (CIS) to enable the automatic import of clinical data and reduce the error rate by entering data twice or transferring data manually from one to another system.

Item 3.13 The system has an interface to integrate pseudonymized biological data (see item 6.1). See for example the concepts of the TMF e.V. [19] or an anonymizing tool described by Prasser et al. [20].

Example

"**Integration with caTissue**

The caTissue Suite,31 which is the tissue bank repository tool developed under the caBIG® umbrella, has been adopted and integrated with the BCCR to collect and manage the biospecimen data in a standard and efficient way. It is used to track the collection, storage and distribution of specimens and provides quality assurance for all of these activities. The participating centers are able to either submit biospecimen data into the central repository or maintain their own installation of caTissue and store biospecimen data locally." [4]

Explanation

If it is planned to analyze biological data within the registry project the software system should provide a module for integrating biological data. Since personalized medicine is becoming more and more popular, we suggest that a patient registry software system should provide this feature. When integrating biological data they must be pseudonymized, see for example the concepts of the TMF e.V. [19] or an anonymizing tool described by Prasser et al. [20].

Item 3.14 Extensibility is possible

Example

"So far, we have defined six optional modules, but the modular structure is easily extensible to fit on other needs. Out of those, five have already been implemented." [5]

Explanation

As already mentioned in item 1.1, the architecture of the system should be extensible so new modules can easily be integrated if they turn out to be necessary during the project time. This is also valid for new interfaces. If it turns out during the project time that a patient interface or an interface for mobile devices would be helpful, it should be possible to integrate them in the system.

4. Interoperability and Semantics & Standardization (F)

This topic contains issues of interoperability with focus on semantic and standardization aspects.

Item 4.1 *Standardized CRFs are used whenever possible.*

Examples

"Data collection forms

To promote the acquisition of high quality, clinically meaningful data, standard data collection forms were defined according to the blood purification standard operation procedures (SOPs) developed by the CSN [21,22], which outline CKD-related signs, symptoms, laboratory tests, and treatments that are internationally accepted to monitor onset, progression, and outcomes over the lifelong course of the disease. Nephrologists from the Registry Board of Advisors participated in the development of the CRFs in an effort to delineate the current standard of care of the patients with kidney disease and to facilitate consistent, thorough, and precise patient evaluations throughout the nephrology community." [1]

"The BCCR questionnaires have been designed and developed to collect comprehensive data related to the diagnosis, treatment and follow-up of BC patients, as well as information pertaining to demographics and survivorship. Existing, well established and recognized in the cancer research community questionnaires, such as the SF-36v2 Health Survey to measure QOL[26] and the NCI Quick Food Scan questionnaire[27] for the dietary habits have been implemented in the BCCR registry. The American Cancer Society's (ACS) examples of moderate versus vigorous physical activity guidelines for cancer prevention[28] were used to create the physical activity form. Sleep habits are assessed by using the Pittsburgh Sleep Quality Index.[29]" [4]

Explanation

We suggest the use of standardized case report forms (CRFs), because the use of standardized CRFs supports a unified collection of research data and is a prerequisite for multi-center analyses at a later stage. In addition it simplifies data integration from multiple centers at a later stage.

Item 4.2 *Standardized Data are used whenever possible*

Examples

> "The data elements of the BCCR vocabulary have been defined based on the caDSR convention and, when possible, were mapped to the caDSR." [4]

Explanation

We propose to collect standardized data whenever possible, because they are comparable and it is much easier to analyze them.

Item 4.3 Ontology-based, standardized metadata are used.

Examples

> "BCCR data element descriptors (metadata) have been constructed from the aforementioned vocabularies using the NCI CBIIT Data Standards Registry and Repository (caDSR) common data elements convention[25] and are available in an electronic format." [4]

> "Our registry adopted all applicable data elements and definitions in accordance with ACC/AHA available published data standards, including those developed for Electrophysiology, Atrial Fibrillation, Acute Coronary Syndromes, Heart Failure, and Cardiac Imaging [29–33]. Other data sources included data elements from large device clinical trials and registries, such as CTOPP (Canadian Trial of Physiologic Pacing) [34], MOST (Mode Selection Trial in Sinus Node Dysfunction) [35], COMPANION (Comparison of Medical Therapy, Pacing, and Defibrillation in Heart Failure) [36], REVERSE (Resynchronization reVErses Remodeling in Systolic Left vEntricular Dysfunction) [37]. We also reviewed case report forms, data elements, and definitions from international data collection efforts. Examples of these data sources include the ACC National Cardiovascular Data Registry (NCDR) [38,39], Health Level Seven International (HL7) [40], Clinical Data Interchange Standards Consortium (CDISC) [41] and Cancer Data Standards Registry and Repository (caDSR) [42,43]. Finally, we also included standardized definitions for clinical endpoints and adverse events in cardiovascular trials from the US Food and Drug Administration (FDA) [44]." [17]

Explanation

We propose to use standardized metadata to support later cooperation with other centers and to support integration and multi-center analysis of the research data.

Item 4.4 Ontology-based, standardized vocabularies are used.

Examples

> "The following controlled terminologies have been implemented both in the BCCR front-end and metadata: NCI Thesaurus (NCIt)[23] and Systematized Nomenclature of Medicine-Clinical Terms (SNOMED-CT).[24] These publicly accessible controlled vocabularies meet all caBIG® Bronze requirements." [4]

> "Shared Ontology service (F)
>
> The Shared Ontology service provides web service access to hierarchical vocabularies that describe i2b2 data elements and provide term mappings for i2b2 query panels. This component functions identically to the standard i2b2 Ontology Cell for query panels, with the additional external policy requirement that at least one common vocabulary exists and is mapped at all data contributor nodes. In practice, as part

of the Shared Ontology model, we additionally provide the requisite i2b2 concept dimension rows as a Shared Ontology service public table; however, these additional term mappings may be equivalently implemented at the SHRINE adapter translation layer.25 We also utilize a new, streamlined Shared Ontology module that incorporates Apache Lucene search capabilities." [8]

Explanation

We propose to use standardized vocabularies to support cooperation with other centers and to support faster integration and analysis of multi-center research data.

Item 4.5 An Extensible Markup Language (XML) schema definition (XSD) is available for structured data exchange.

Examples

"**The CNRDS application programming interface (API)**

Prior to the nationwide CNRDS, Shanghai, Beijing, and Zhejiang provinces established their own local renal registries. Indeed, these three registries have been running for several years and contain valuable data. In order to maximize the value of the previous investment, a RESTful web service API was developed for third party software providers to exchange their data with CNRDS. The RESTful web service was chosen due to its streaming capability and on-the-fly compression [35]. The data exchange format is a customized XML schema, which is downloadable as an additional file of this manuscript (Additional file 1). Each regional registry was assigned a unique API key for discrimination. Because the API is an interface between machines, third party software vendors should ensure that the data they submit is validated and complete. Data validation rules are also performed on the CNRDS server; an error code will return if the submission contains abnormal data." [1]

"*Interoperability*

CEMARA was conceived in order to communicate with other sources of information: the use of XML, as an exchange format, permits a greater flexibility and better capacities to exchange data with other information systems such as French Medical Insurance system or Hospital Information systems. It also allows importation of former databases (Figure 2)." [2]

Explanation

An XML schema definition for structured data exchange should be available to support interoperability, to foster active data exchange and communication between different centers, and to enable cooperative research projects.

5. Internationality (F)

This topic contains an item which a registry software system can provide to support international cooperation in a registry project.

Item 5.1 The whole questionnaire life-cycle can be displayed in different languages.

Examples

"Multilingualism: The whole life-cycle — designing questionnaires, administering questionnaires and displaying the results to physicians — was designed to accommodate for the need of supporting different languages. Therefore, nurses are able to select the language for questionnaire administration for each patient individually. The software is currently available in English, German and Italian. Further translations are currently ongoing." [9]

"The Cochlear P-IROS electronic, web-based platform is currently available in five languages; English, Mandarin, Korean, Japanese and Russian." [15]

Explanation

A patient registry software system should support internationality by providing multilingualism for the whole life-cycle – if necessary. For most international projects it may be sufficient to provide the whole life-cycle in English, but in some cases it may be necessary to provide the whole life-cycle in other languages as well.

6. Data management, data quality and usability (F)

This topic contains important items which should be considered by data management to support data quality and usability of the registry.

Item 6.1 A pseudonymous patient identifier (PID) is created by the system.

Examples

"Administrative data include: (i) date when questionnaire is submitted; (ii) current status of the questionnaire; (iii) registering institution code; (iv) clinician's ID; and (v) subject's identification code—an automatically generated number that can be used to re-identify the subject when data are deidentified (as permitted by HIPAA regulations)." [4]

"To limit the impact of potential exposure or leakage of patient information, the JADE Program does not store any identifying patient information electronically. No name or national identity number is captured and a case specific code (ADF code) is generated for each enrolled patient. Only the physician has the information to identify the patient given a specific ADF code. These case sensitive codes are known to the patients and all report forms generated electronically or in paper form, are kept by the physicians or in the case records as appropriate." [23]

Explanation

It is not necessary to store the personal information of the patients such as names, addresses or social security numbers in the registry. For patient identification a pseudonymous patient identifier (PID) should be created by the system when a new patient is allocated for the first time. This PID is used in the tables of the relational database to identify the corresponding patient data. The personal patient information should be stored separately and can be identified through the PID, which must be marked there. (See also item 8.4 Encrypted data storage).

Item 6.2 CRF is divided in logical parts.

Example

"According to the BCCR rules, information on personal, demographic, lifestyle, physical activity, dietary habits, family history, women's health, genetics data, symptoms, QOL, and medical history may be provided by a subject; whereas medical information on diagnostic studies, pathology/staging, treatment, surgeries, biospecimens, and survival can be provided only by clinical personnel. The Core Data Set categories included in this registry are described below." [4]

Explanation

The collected data in the eCRF should be divided in logical parts, as described by Sherman et al. [4] in the above example. Registries often collect a big number of variables they can be structured for different aspects, which data belong together, who will insert them, who will elevate them.

Associated variables should be collected together. Not all variables are available at one moment. Data can be structured, for example in demographic data, laboratory data, quality of life (QOL) data, treatment data, etc.

Item 6.3 CRFs are customizable according to the user's selections.

Example

> "**Figure 2 Data collection form fragment.** The initial erythropoietin (EPO) data collection form is shown in (a). As shown in (b) and (c), the visible data elements change according to the user's selections" [1]

Explanation

To avoid ambiguity, CRFs should be customizable according to the user's selections if necessary. In a customizable CRF, for example gender-specific questions can then only be shown if applicable, and the not applicable questions can be faded-out. This is timesaving and prevents people to insert their own abbreviations, because they want to show that the questions are not applicable.

Item 6.4 If requested a minimal and extended dataset can be used.

Example

> "The dataset is divided into 2 sections: minimal and extended. The minimal dataset is mandatory for all participating centers, and only data fulfilling these minimal requirements will be incorporated into the research database. The extended dataset facilitates deeper insight into patients' treatment and supports the documentation of additional items that are partly predefined and that can partly be defined by the participating center itself." [11]

Explanation

If a minimal and extended dataset can be used, this allows the definition of the core variables mandatory for all cases or most wanted in the registry and more specific documentation in other cases. Especially in multi-center studies this feature is often asked for because different centers capture different variables and so each center can keep their specific variables in the registry.

Item 6.5 The system supports the use of all common data types.

Example

> "Electronic data capture (EDC): the core purpose of any registry is to collect medical data on subjects electronically. All of such data must be given a data type, which can be either numerical, date/time, or one or more items selected from a predefined list (terminology, cf. next item). Regarding an appropriate statistical assessment, unstructured text is disadvantageous and shall be avoided. Numerical items must have a unit and a reference interval for instantaneous plausibility checks. Of course, EDC is an inherent component of any registry." [5]

Explanation

A patient registry software system should support the use of all common data types which will be selected in CRFs, like integers, booleans, characters, floating-point numbers, alphanumerical strings, and also advanced types such as dates and multiple choice selection. This enables the collected data which are relevant for the research to be stored directly in the database.

Item 6.6 Special data types, like images, X-rays, and links can also be stored.

Example

> "Basically, the BLOB module ensures that large data files are safely transferred via the internet and attached to the subject's identifier providing the date of filing. The date of transfer and the transferring person's identifier are logged in the audit trail module. Versioning of data is possible, since the module defines a document identifier (DID) that is not unique, and a Boolean flag "latest" indicating the latest version of the respective DID. Terminology is used to classify the types of data (e.g., photograph, ECG recording, scanned diagnostic letter) as well as the according file endings such as the portable document format (PDF), portable network graphics (PNG), or DCM for digital imaging and communication in medicine (DICOM) files. Technically, the BLOB module parses, extracts, and handles BLOB data received from a hypertext transfer protocol (HTTP) request object. The request object is built by the hypertext markup language (HTML) file upload object, according to the specification of the Internet Engineering Task Force (IETF) [25]. … In the German Calciphylaxis Registry, the RDR BLOB module allows web-based image integration. Photographs that have been taken on patient's bed site are uploaded and linked to the subject ID in the registry. All images are described by their recording date and the body region that is visualized in the image. For precise localization, the body part terminology has been defined according to the image retrieval in medical applications (IRMA) code for medical images, i.e., a mono-hierarchical multiaxial classification scheme [28]. … Figure 5 (left) visualizes the BLOB module integrated into the German Calciphylaxis Registry. An overview of images is displayed, which can be magnified on the user's selection. The list can be accessed by patient ID, recording date, body region, detailed position, left or right hand side, and any combination of those (filter bars on top)." [5]

Explanation

It should be possible to store special data types like images, x-rays, links, etc. in the registry. For some diseases these data contain important information which should be available beside other patient data so the information in these data is available for analysis and further treatment.

Item 6.7 Multiple choice data collection is used whenever possible.

Examples

> "The vast majority of responses on all forms are multiple choice, check box, radio buttons or pull-down response options to facilitate entry and reduce entry error. Minimal free-text response fields are included." [15]

> "To minimize manual entry by the end user, drop down boxes and check boxes were used for data entry." [24]

Explanation

Whenever possible the data in a patient registry software system should be captured systematically by using multiple choice selections, drop down menus or radio buttons, etc. Free text fields should be avoided or only used if the data cannot be captured systematically. If the data are captured systematically they can be analyzed easily, while it is extremely time-consuming to evaluate free text.

Item 6.8 No predefined selection is used to avoid unwanted entries.

Example

> "Another frequent user complaint is that data entry is too time-consuming. A common solution to streamline data entry consists of pre-instantiating all items with informative default values (e.g., choosing the most frequent value as the default). However, in that case, users could pay less attention to the input fields and stay with the default values even when they are not the correct ones. In order to compensate for that bias, we decided to use the "missing value" as default." [10]

Explanation

We recommend utilizing no predefined selection in the data entry fields. As stated by Lanzola et al. [10] in the above example, if predefined selection is used, users may pay less attention to the input

fields and use the default values even when they are not correct. This leads to false values in the database and when analyzed may falsify the results.

Item 6.9 The system has implemented data validation components (Hard-and Soft Checks).

Examples

"Data validation

CNRDS is dedicated to ensuring that the data entered into the registry is accurate. Formatting and acceptable range information for each data element was provided in the spreadsheet after the experts defined the CRFs. The IT support team used this information to build logical evaluations for the data. The user's input is validated both server-side and client-side. The validation components of the web interface prevent the users from entering erroneous information into the system. A JavaScript validator checks the user's input against the validation rules before submitting the form. If a user does not enter data in a required field, the system displays error messages requesting this information prior to accepting the entry for submission to the registry. The system will not accept the data until the required fields are filled in. The same applies if the data entered is out of the expected range. For example, CNRDS recognizes the expected range for an individual's total bilirubin as between 0 and 1000 µmol/L. A warning message displays if the data entered by the user is below or above the acceptable range. This gives the user a chance to review the information he/she entered and apply any necessary corrections before submitting the data. Similar algorithms are performed server-side before the data stored in the database. Authorized batch submission from third party registries through the RESTful Web service is only validated server-side. Third party registry software providers are informed to guarantee the integrity and validity of their data." [1]

"The BCCR system includes validation components that prevent entering erroneous information by the users." [4]

"The data is automatically validated during the data entry process." [11]

"Data Quality

The software included automatic check features, such as logic branching and requiring a response, to ensure accuracy and completeness before being saved. During data collection, team supervisors accompanied enumerators and did spot checks to ensure that enumerators were filling in forms correctly. If any errors were found, they were corrected on the tablet if possible, or else recorded in an error log and reported to the Data Manager for resolution in the electronic database daily or at the end of each phase. During vaccination, data collected on vaccine recipients for the vaccine registries were linked directly with census data in the tablet records, allowing for accurate data linking at the point of vaccination. The registries were reviewed nightly or every two nights, and cleaned at the end of each vaccination phase." [18]

"With REDCap, we restricted data format/type, set ranges for date and numeric fields, and allowed data validation. Data consistency problems such as incorrect data type, values out of range, and outliers for numerical fields can be reported using the data quality module." [25]

Explanation

The system should have implemented checks to validate inserted data automatically. In this phase mainly range checks and missing data will be rejected. There should be hard and soft checks. While hard checks prevent the insertion of implausible values and don't let the user continue until the implausible values are corrected, soft checks display a warning only and enable the user to continue, also if complained values will not be changed.

Item 6.10 The system has a query tool to perform automatic data queries.

Explanation

The system should have an integrated query tool that checks the data and performs automatic queries. These queries can be performed by the study groups and the answers will be inserted automatically in the database. With such a system it is possible to perform more sophisticated checks than during the input phase.

Item 6.11 The system has an interface for manual data check.

Example

> "CMR data quality assurance and quality control
>
> Web-based applications were developed to allow CMR staff to routinely perform data quality assurance and quality control (QA/QC) activities. For instance, the link on the main menu, CheckCMRSybase tables (right section in Figure 1), leads to applications that enable CMR staff to check invalid data and track referential integrity of links among CMR's relational data tables to make sure that no "orphan" records exist in the data tables." [26]

Explanation

Beside the automatic checks during data insertion and an automatic query tool, a patient registry software system should also provide an interface for manual data check. Not all errors can be detected by program and in some cases it is essential to check the data manually and perform individual queries.

Item 6.12 It is possible to perform manual data queries within the system.

Explanation

Since the data is checked manually it should also be possible to perform manual data queries within the system. These questions can be answered by the study centers and the correct data can be inserted directly in the system. Otherwise it may be too cumbersome to resolve manually detected errors if this must be done outside the system and results must be integrated later.

Item 6.13 Entries with unresolved queries are marked with flags at different levels (item, part, patient, etc.).

Example

> "While the monitor cannot change any data, he/she can flag reminders to the local supporting team when suspicious data is encountered such as large discrepancy in body mass index and waist circumference." [23]

Explanation

All entries with unresolved queries should be marked with flags at different levels (item, part, patient, etc.) to indicate the user that there are data to check. If the queries are answered these query flags should disappear.

Item 6.14 Implausible entries are marked with flags at different levels.

Example

> "The red, grey, yellow, and green icons signify incomplete, blank, unverified, and complete records, respectively. Clicking any of the colored buttons in the table automatically activates the associated data collection instrument." [25]

Explanation

All entries with implausible values should be marked at different levels with flags to indicate the user that these data must be checked and possibly corrected. This would improve data quality and completeness, so we recommend that a patient registry software system should provide this feature. As described by Pang et al. [25] in the above example the query flags should be connected with the associated data collection instrument, so the corrected values will be inserted directly in the database.

Item 6.15 Unplanned visits can be integrated flexibly.

Explanation

It should be possible to easily integrate unplanned visits in the system, so the data of additional visits can also be captured and are available in the system.

Item 6.16 The system should be designed following the standards of software ergonomics, defined in ISO 9241-110 [27].

Example

> "As pointed out in the introduction, CHES was designed to foster the integration of PROs in daily clinical routine. To develop an appropriate IT solution for this scenario, it is of utmost importance to constantly take the application's end users' feedback into account [26]. Usability prototyping and usability testing ensure that users are included at an early stage (standards for user oriented software are specified in ISO 13407)." [9]

Explanation

The standards of software ergonomics, as defined in ISO 9241-110 [27] should be adhered to when developing a new patient registry software system. As described by Holzner et al. [9], the end users should get involved in the development process early to evaluate the system, see also item 2.4.

7. Data Analysis (F)

This topic contains issues with which the registry software system can support data analysis.

Item 7.1 The system has a query builder to assist researchers to select interesting patient cohorts.

Example

> "The end user, typically a research investigator, accesses the registry through a web-based query interface. Following secure log in, the user encounters a graphical query builder interface in which registry-specific ontologies may be browsed; search terms may be dragged and dropped to construct queries to define subject cohorts of potential interest." [8]

Explanation

The system should provide a web-based interface with a query builder for researchers. As described by Natter et al. [8], in this query builder registry-specific ontologies may be browsed, search terms may be dragged and dropped to construct queries and select subject cohorts of potential interest for further analysis.

Item 7.2 The system is able to generate reports for selected cohorts.

Example

"Selected cohorts are returned as patient sets, for which choices of pre-defined summary reports and visualizations may be generated in real time (figure 2)." [8]

Explanation

The system should be able to generate reports for selected cohorts. With such a component researchers can get easily an impression of the data in the registry, this is an important instrument to generate status reports during the project time and reports can be created to support decisions for further analysis of the data.

Item 7.3 Datasets can be generated and downloaded for analysis in different formats, complete or for selected cohorts.

Examples

"Since the FRB! Project software is designed to be a research tool, data export and analysis features are very important. Individual Users can download their own data at any time as a text file in comma separated variable format. The software also offers statistical tools for simple analyses. Users can export their own data and analyze it as they see fit for more sophisticated analyses." [6]

"In this way, end users may iteratively refine their searches and are able, with appropriate authorizations, to download the resulting datasets for further analyses." [8]

"The database is kept and managed at the coordinating center, but data export and statistics are available at any time to every user within the SUN." [10]

Explanation

It is important that datasets can be selected and downloaded for further analysis. This is necessary for analysis of the complete data and also for downloading specific cohorts. For example, in cooperative research projects it should be possible for each participating group to have access to their own data at any time.

Item 7.4 Results can be presented as colored graphs in real time.

Examples

"The end user, typically a research investigator, accesses the registry through a web-based query interface. Following secure log in, the user encounters a graphical query builder interface in which registry-specific ontologies may be browsed; search terms may be dragged and dropped to construct queries to define subject cohorts of potential interest. Selected cohorts are returned as patient sets, for which choices of pre-defined summary reports and visualizations may be generated in real time (figure 2). In this way, end users may iteratively refine their searches and are able, with appropriate authorizations, to download the resulting datasets for further analyses." [8]

"Graphical presentation of results: results are presented as colored graphs in real time. The graphical output (see Figure 2) links PRO to the course of disease and treatment and in addition specific medical interventions can be easily incorporated and displayed. Results can be displayed optionally in a longitudinal or cross-sectional setup." [9]

Explanation

It should also be possible to show results as colored graphs in real time within the system. This helps researchers get a quick overview and obtain an impression of the captured data in the system and make informed decisions for further analysis or continuation of the registry.

Item 7.5 The system gives interactive feedback, classifying the patient through based on implemented knowledge or scoring systems.

Examples

> "Flag System: Based on reference values from literature or previously collected data, the Flag System allows for the quick identification of patients with clinically relevant problems, using cut-off scores or score distributions." [9]

> "Clinicians were impressed by the automatic Revised Trauma Score calculator, which provided an instant estimate of survival probability once admission vital signs were entered. Early availability of prognostic scores for patients who are actively being resuscitated would represent a significant advance over North American injury scoring, in which data collection and entry by analysts are often required before these scores are available. Even more importantly, injury scoring can create the unprecedented possibility of comparing adjusted outcomes to national or international norms." [16]

> "Based on results estimated by the JADE Risk Engine, the e-portal displays the 5-year probability of major clinical events which can be adjusted by changing values of modifiable risk factors to promote discussions between patients and care providers (Figure 3). Data collected at each review visit are displayed to show the trends of control of modifiable risk factors including BP, HbA1c, LDLC and body weight. General recommendations can be triggered by predefined levels of risk factors to prompt care providers and patients to take appropriate actions. Printable reports showing risk predictions, trends of risk factor control and practice tips can be generated for care providers (in English) and patients (in 5 different Asian languages i.e. English, Thai, Korean, Malay and Chinese [both traditional and simplified Chinese]) for record purpose (Figure 4). Furthermore, the portal provides matrixes to help doctors monitor patients' levels of adherence to care processes (e.g. annual assessment, review visits, education sessions, laboratory tests) and self management as well as their status of attainment of treatment targets. These targets can be modified depending on the evolution of international healthcare standards." [23]

Explanation

The system should give interactive feedback according to the entered patient data to implemented knowledge bases and scoring systems. This is a great advantage of web-based information systems that patients can get immediate feedback when their data is entered into the system. They can take advantage of already available information at the beginning of their treatment. This is also a great advantage for physicians, they do not have to check each patient data by hand but they get the information from the system immediately after entering the patient data. Therefore the patient registry software system can help in knowledge-based decision making.

8. Security aspects (F)

This topic contains security aspects of the registry software system and important security aspects of the registry operation process.

The system must be secure according to the actual regulations.

Item 8.1 Only authorized users have access to the data.

Examples

> "The CNRDS system is only accessible to registered users at participating sites. To ensure that they have the authority to proceed with data entry, authorized users are issued their own unique electronic signature, i.e., a username/password combination." [1]

> "The BCCR web application that supports data collection is accessible only for authorized users, … . The authorized users must have their own unique electronic signature—a combination of a user name and a password." [4]

Explanation

Only authorized users may have access to the system. This must be guaranteed at least via an access control consisting of a username and password. In some cases it can be necessary to provide an additional secure component for example an identity card, but in most cases username and password should be sufficient.

Item 8.2 The system provides role-based user access.

Examples

> "Each user has an appropriate level of access to data. The user roles and types of authority are described in Table 1." [4]

Table 1. BCCR user roles and their authority.

Role	Authority
Subject/patient	Can enter/update personal, demographic, lifestyle, symptoms, QOL, family and medical history data
Lab technician	Can enter/update biospecimen data only
Clinician	All of the above + Enter medical data, retrieve and edit existing cases of his/her patients
Coordinator	All of the above for the assigned clinicians
Center manager	All of the above + Retrieve and edit cases of the patients of the center/institution
System coordinator	All of the above + Retrieve and edit all cases, activate/suspend users, assign user authorities

> "The spectrum of functionality offered to the logged-in user varies dependent on his/her role in the system; the predefined roles include study nurse, clinician, supervising clinician, data quality manager, and registry administrator." [11]

Explanation

Role-based user access is an important method to give several users different rights to the system and the data. A patient registry system should provide this feature because not all users need all rights. This is also an important feature to protect the data because some users require only the right to read the data and not to insert or change data. So unintended data changes can be prevented. The user roles can be created and allocated according to the specific needs of the project.

Item 8.3 The system utilizes secure web server communication through encrypted data transfer.

Examples

> "The system utilizes Hypertext Transfer Protocol Secure (HTTPS) for web server communication." [1]

> "The system utilizes secure web server communication and supports Secure Socket Layer (SSL) (an Internet encryption method that provides two-way encryption along the entire route that data travels to and from a user's computer) and Hypertext Transfer Protocol Secure (HTTPS) authentication (the communications standard used to securely transfer pages on the Web)." [4]

> "All data transmissions between the user and the server are encrypted using 128-bit encryption (Secure Sockets Layer)." [6]

> "Complete data transfer is encrypted using an industry standard (SSL/TLS: Secure Sockets Layer/Transport Layer Security)." [11]

Explanation

The complete data transfer must be encrypted according to actual transcription protocols and standards (SSL/TLS: Secure Sockets Layer/Transport Layer Security, with a secure transcription, for example Triple-DES or AES and the hash total SHA with high key length, 256 or 512), so if data are intercepted they are useless for the thief because they are encrypted and not readable.

Item 8.4 Sensitive Data can be stored encrypted in the database.

Examples

> "All subjects' identical information collected in the CNRDS is encrypted when entered into the database, which minimizes the risk of unauthorized access to this data." [1]
>
> "All Protected Health Information (PHI) is stored encrypted in the database and requires encryption/decryption functions with a pass phrase in order to insert or select data." [4]

Explanation

It should be possible to store sensitive data encrypted in the database. However, personal identification data should not be stored together with the medical data in the registry. Therefore we created item 6.1 pseudonymous patient identifier (PID). If it is necessary to store the identifying information, it should be stored in a separate database and it can be connected through the PID. Then it should be possible to store the identifying information encrypted in a separate database. However, beside the PID it might be possible that there is also medical information which requires encrypted storage in the database.

Item 8.5 All changes in the database are tracked and monitored through an audit trail.

Examples

> "The BCCR maintains an audit trail of all data entries to protect the authenticity, integrity and confidentiality of all data entries." [4]
>
> "*Audit Trails* implement a complete logging of any database transaction. Disregarding the data that is changed, all changes are logged in the same table build from the columns (i) User, (ii) Timestamp, (iii) Action, (iv) Revision, (v) Entity, (iv) Property, (v) DataType, (vi) OldValue and (vii) NewValue. The Action identifies whether new data has been created or existing has been modified (i.e., insert or update) and Revision is a counter that is incremented with the transaction. Hence, modifications in the database resulting from the same user action are labeled with the same revision number and can be easily joined. Entity and Property refer to the database table that has been modified and the according field, respectively." [5]
>
> "The electronic registry platform operating the Cochlear P-IROS contains an inherent audit trail to trace all amendments made with each form, the investigator making the change and when changes are made." [15]
>
> "Every interaction with the data is logged, creating an audit trail." [25]

Explanation

All data entries and changes should be maintained in an audit trail so that they can be tracked, and if necessary be reversed.

Item 8.6 If necessary, master-slave replication should be established.

Example

> "We established MySQL master-slave replication for load-balance and, more importantly, for backup. Replication enables data from one MySQL database server (the master) to be replicated to one or more additional servers (the slaves). Because data is replicated to the slave(s), and the slave(s) can pause the replication process, it is possible to run backup services on the slave(s) without corrupting the corresponding master data [34]." [1]

Explanation

In some cases it might be useful to have a system which provides master-slave replication. If it is not sufficient to store backups regularly (item 8.7) it should be considered, master-slave replication may be a useful additional security component.

Item 8.7 Backups are stored separately regularly.

Example

> "Further, a dumped database is packaged and transferred to a backup facility located in a different network zone every night." [1]

Explanation

Backups should be stored regularly, at least every night, so that if a system failure occurs it is possible to restore the system to the data status from the previous night and only the changes from the current day are lost.

Item 8.8 The server is behind a firewall.

Example

> "The security system consists of both the infrastructure, policies, and protocols in place today, as well as novel functionality and approaches to support privacy and security protections including firewalls at strategic points that enforce access control policies between networks, and authentication and identity proofing procedures ensuring that only authorized users can access the application and data for which they are authorized." [26]

Explanation

The server must be located behind a firewall to minimize the risk for attacks.

Item 8.9 The server room is locked and temperature controlled.

Examples

> "According to the HIMSS, a complete security solution that maximizes the benefits of networked data communications must contain the following elements: User Authentication, Access Control, Encryption, Physical Protection, and Management.[19]" [3]

> "Every effort is made to ensure confidentiality by means of data encryption, password authentication of users, electronic firewalls and locked storage facilities, de-identification of PHI, audit trails, a disaster prevention and recovery plan, and security measures for back-up." [4]

Explanation

The server must be located in an access and temperature controlled room to minimize the risk for unwanted access and system failure through improper temperature.

9. Privacy (F)

This topic describes privacy aspects of the registry and of the software system additional to items 6.1 and 11.4.

Item 9.1 A data protection concept should be established before starting a registry project.

Example

> "Because the registry focuses also on the long-term outcome, patient identity is also obtained, which imposes high requirements on data privacy and protection. In this regard, the system's architecture and the data privacy and policy concept follow the generic framework of the German Technology, Methods and Infrastructure for Networked Medical Research (TMF) organization.6 The TMF and the state commissioner for data protection of Baden-Württemberg were both consulted regarding the concept. In addition, the data privacy and policy concept of CERTAIN was approved by the Ethics Committee of the University of Heidelberg. To reduce the amount of regulatory work for participating centers, the registry headquarters offers the service of providing the required documents to the respective local Ethics Committees." [11]

Explanation

Before starting a new registry project there should be a data protection concept established which takes into consideration all known use cases that may occur during the registry project.

Item 9.2 The registry software system should provide double pseudonymization for biological and genomic data.

Explanation

If biological data are integrated in the registry they should be double pseudonymized according to the concepts of the TMF e.V. for example [19] or using an algorithm such as specified in Prasser et al. [20], see also item 3.13.

10. General Features (F)

This topic contains general items which have no direct relation to the other topics.

Item 10.1 The costs of a patient registry software system must be taken into consideration, when choosing a software system: Is there a realistic calculation of the costs considering the costs of the procurement / programming, operation, life-cycle, archive, system replacement? Are the financial resources sufficient regarding the anticipated costs of sustainability of the registry guaranteed?

Examples

> "DADOS Prospective is a Web-based application developed by the research on research group (RoR) [37] to support data collection activities among researchers, research groups, and research networks [7]. It enables users to replicate any case report form into an eCRF, collect data in single/multisite studies, and extract data in an interoperable format. It is compliant with Chapter 11, Title 21, Code of Federal Regulations [4] and Health Privacy and Accountability Act (HIPAA) guidelines for EDC. It can be used to streamline and support individual/departmental/institutional databases, registries, and single/multisite clinical/nonclinical studies and clinical at a low cost [28]. … Cost is inarguably an important factor to be considered while choosing an EDC system. Although the nature of licensing and support required to maintain an EDC system are the main predictors of cost, the purpose of data capture and workflow at the site of implementation also have a major influence on the cost. Implementation at an institutional, departmental or individual level, number and type of users, single site or multi site data collection and workflow complexity are some examples of the latter. Commercial EDC systems like Oracle Clinical,

InForm and Rave are expensive in comparison to open source EDC systems like DADOS prospective, OpenClinica and TrialDB. While, the former have a higher presence in industry sponsored clinical trials, the latter are more common in academic settings." [28]

"The technological platform was notable for its low production cost." [29]

Explanation

As it is important to calculate and check the costs of a patient registry project, this is also true for the used system. Especially in universities research projects have to cope with limited financial resources and the costs for commercial software or outsourced software systems are not affordable. Besides this disadvantage, commercial software systems are often restrictive and in case of outsourced systems the data must be given elsewhere. So in university settings open source systems and own developments are predominant.

Item 10.2 The system has a multi-client capability. Several projects can simultaneously be executed in one installation of the system, but should be strictly separated.

Example

"The i2b2-based self-scaling registry platform (i2b2-SSR) developed for this purpose allows individual investigators and institutions to join a secure research data network by contributing a unique dataset and working with others to create larger, collaborative datasets that may be shared with the network as a whole or within specific subsets of sites and investigators." [8]

Explanation

If several projects can be handled in one installation of a registry software system this would reduce installation time and costs, and it would foster the cooperation within different registry projects. In the above example Natter et al. [8] describe the cooperation of different registries in one installation of a registry software system. Especially the different user-groups can benefit from such a solution, because they can share their data and increase the number of patients in each individual project.

Item 10.3 An update mechanism for the system is in place.

Examples

"Update Mechanism: In order to efficiently install updates for local CHES installations, an update mechanism is in place, providing an one-click solution for installing, new features and available bug fixes." [9]

"An automatic update function is also highly recommended for future software upgrades." [13]

Explanation

There should be an update function for the software which allows easy installation of future software updates. So the system can be updated when a new version is available and new features or security aspects can easily be installed.

Item 10.4 Source documentation of CRFs in pdf format is possible.

Example

"The system must be integrated in the existing physician's workflow. Therefore, this electronic registration is almost identical as paper-based registration. Instead of the paper forms, data are entered in a PDF, which can be processed by a computer. If the physician still wants to keep a paper-based print-out, an identical paper will be produced, while the system stores the data entries for long-term analysis." [12]

Explanation

Sometimes the study team requires that the CRFs with the source values can also be extracted as PDF for offline use. If this feature is needed it should be considered when the software for the system is chosen.

11. Organizational (F)

This topic comprises software-related organizational items.

Item 11.1 The system is compliant with all known relevant regulations.

Example

> "In this regard, the system's architecture and the data privacy and policy concept follow the generic framework of the German Technology, Methods and Infrastructure for Networked Medical Research (TMF) organization.[6] The TMF and the state commissioner for data protection of Baden-Württemberg were both consulted regarding the concept. In addition, the data privacy and policy concept of CERTAIN was approved by the Ethics Committee of the University of Heidelberg. To reduce the amount of regulatory work for participating centers, the registry headquarters offers the service of providing the required documents to the respective local Ethics Committees." [11]

Explanation

The patient registry and also the used software system must be compliant with all known relevant regulations which are applicable in the specific governing area. The regulations may vary from area to area, so it has to be checked which regulations are applicable and it must be ensured, that the patient registry and the used software system are compliant with all these regulations.

Item 11.2 The registry is compliant with Chapter 11, Title 21, Code of Federal Regulations and HIPPA.

Examples

> "IRB and subject recruitment
>
> The BCCR participating centers are required to obtain approval from its Institutional Review Board (IRB). The BCCR provides standard protocol templates and privacy assurances in procedures of informed consent that have been formulated to detail the use of web-based tools. A template of common protocol statements includes: (i) methods and procedures applied to human subjects; (ii) data storage and confidentiality; (iii) potential risk assessment for human subjects; (iv) risk classification; (v) protection against the potential risks for human subjects; (vi) potential benefit assessment for human subjects; (vii) potential benefits to society; and (viii) alternatives to participation. A template for a common informed consent form includes the following HIPAA-mandated information: (i) a specific description of the information to be used or disclosed; (ii) the person or entity to whom disclosure will be made; (iii) the purpose of the use or disclosure; (iv) an expiration date or event for use of the information; (v) an explanation of how authorization may be revoked; and (vi) any restrictions placed on the subject's access to the information with access granted upon completion of the research. All participating investigators are able to use these standardized statements to assist them with their IRB applications.
>
> All BCCR participating researchers and clinicians are required to complete the computer-based training course on the Protection of Human Research Subjects. All information gathered in the BCCR should be compliant with IRB approvals at participating sites that are monitored by each center's IRB. The BCCR coordinator opens new accounts and enables data entry into the BCCR only after receiving the documented proof of IRB protocol approval. In order to enter data into the BCCR, a copy of the consent form for each subject must be submitted to the BCCR coordinator. Under the informed consent process,

study participants have been asked to voluntarily participate in the BCCR. The potential participants are asked about their willingness to share the information they provided in the BCCR with research collaborators. The information the participants provide is collected for research purposes only. The subjects are informed in the consent that their PHI will be encrypted and that the web-based registry is accessible to authorized users only. Identifiers will never be released in order to protect participant confidentiality. Every effort is made to ensure confidentiality by means of data encryption, password authentication of users, electronic firewalls and locked storage facilities, de-identification of PHI, audit trails, a disaster prevention and recovery plan, and security measures for back-up.

Participants have also been informed that they may revoke the authorization to use and share their PHI at any time by contacting the principal investigator in writing. If they revoke the authorization, they may no longer participate in the research studies and the use or sharing of future PHI will be stopped, but the PHI which has already been collected may still be used." [4]

"DADOS Prospective, OpenClinica1 and Redcap are examples of open source EDC systems. DADOS Prospective is a Web-based application developed by the research on research group (RoR) [37] to support data collection activities among researchers, research groups, and research networks [7]. It enables users to replicate any case report form into an eCRF, collect data in single/multisite studies, and extract data in an interoperable format. It is compliant with Chapter 11, Title 21, Code of Federal Regulations [4] and Health Privacy and Accountability Act (HIPAA) guidelines for EDC. It can be used to streamline and support individual/departmental/ institutional databases, registries, and single/multisite clinical/nonclinical studies and clinical at a low cost [28]." [28]

Explanation

The patient registry software system and the complete data workflow and data storage should be compliant with Chapter 11, Title 21, Code of Federal Regulations and HIPPA to ensure high quality research. If necessary, informed consent must be obtained from the patients.

Item 11.3 There are clearly defined rules which describe the rights on the data for each participating institution.

Examples

"Organizational model

The BCCR utilizes the confederation model assuring that each institution voluntarily participates in the registry, retains all rights to its own data, and has equal representation in the registry's steering committee. A confederation encourages any interested center regardless of its size or location to participate in database development and utilization. The data collected at any location can be used by other participants only after obtaining required permissions and by providing corresponding references and acknowledgements." [4]

"Since the FRB! Project software is designed to be a research tool, data export and analysis features are very important. Individual Users can download their own data at any time as a text file in comma separated variable format. The software also offers statistical tools for simple analyses. Users can export their own data and analyze it as they see fit for more sophisticated analyses." [6]

"The steering committee created bylaws and rules of procedure for the registry, which clearly describe the ownership of the stored data and the rights for analyses and publications. The data provided by each participating center can be exported directly from the CERTAIN web application by this particular center without involvement of the registry headquarters or the steering committee. Similar rights are granted on a national level for the participating countries. Country-specific analyses are facilitated by a national coordinator, who primarily interacts with the registry's headquarters. Only analyses of the multinational dataset in the CERTAIN Registry require approval by the steering committee." [11]

Explanation

The registry should have a steering committee as described by Plotnicki et al. [11], which defines among other things the rights on the data in the registry. It should be regulated that each participating institution receives a copy of their inserted data at any time. It would be preferable that they can download a copy of their own data at any time without contacting the registry headquarters.

Item 11.4 The system is compliant with all known appropriate data protection guidelines of the registry project.

Example

> "In this regard, the system's architecture and the data privacy and policy concept follow the generic framework of the German Technology, Methods and Infrastructure for Networked Medical Research (TMF) organization.[6] The TMF and the state commissioner for data protection of Baden-Württemberg were both consulted regarding the concept. In addition, the data privacy and policy concept of CERTAIN was approved by the Ethics Committee of the University of Heidelberg. To reduce the amount of regulatory work for participating centers, the registry headquarters offers the service of providing the required documents to the respective local Ethics Committees." [11]

Explanation

A data protection concept should be formulated and the patient registry software system, the complete data workflow, and data storage should be compliant with all known according data protection guidelines.

12. Training (D)

This topic contains items which are important for the user training.

Item 12.1 There should be manuals for the registry end-users and for the operators.

Examples

> "The comprehensive training materials, manuals defining vocabulary used in the BCCR and user manuals with a defined set of procedures and lines of responsibilities for each level of participants were distributed to the centers." [4]

> "A quick tutorial on how to use CIRIS was created by the SIT and placed on the laptops and in a project binder next to these computers." [12]

Explanation

A user manual how to use the system should be available for the operators, and there should be also one with project specific content for all end-users involved in the registry project.

Item 12.2 At the beginning, and if necessary during the project time, a training is provided for the users.

Examples

> "To guarantee the consistency and reliability of data collection across the participating centers, the BCCR coordinator continuously provides educational training sessions and audits submitted data, whereas individual center managers review the data submitted from their respective centers." [4]

> "Personnel Training for Data Collection

We performed a semi-structured training with the clinical research coordinators. Our goal was to provide a general overview of the registry database, while concurrently identifying specific factors which could compromise the integrity of the data collection. To ensure a standardized and consistent data collection we developed a standard operating procedure (SOP) specifically related to the primary data collectors tasks. This SOP provides a description of all data elements collected as well as the sources used to obtain the data. After the training process, the data entry activities of clinical research coordinators were closely monitored for three months by the principal investigators (RC and KRS) to assess whether data collection was conducted according to the study protocol. We used the REDCap report tool for monitoring and querying patient records. Corrective actions were taken to address problems related to data inconsistency and missing information, involving retraining and immediate feedback on issues such as missing, out-of-range values and logical inconsistencies." [17]

Explanation

At the beginning, and if necessary during the project time, a user training should be provided to instruct the end-users how to use the system. The training should also include project specific content to minimize data errors caused by wrong insertion. Also for the operators, appropriate training sessions should be provided.

Item 12.3 Regularly user feedback is collected for further improvements of the system.

Examples

"The BCCR developers regularly collect feedback from the end users and evaluate the system's interface for further improvements." [4]

"The incident reporting screen is the cornerstone of the safety improvement initiative and will evolve over time on the basis of user feedback." [30]

"P-PROMPT Implementation, Training, and Impact

The primary health care team's appraisals of P-PROMPT were examined via questionnaire with respect to the following domains: Learning, Training, Using, Usefulness, Daily Practice, Practice Planning, CDMS, Support from the Service Provider, and Satisfaction. Each domain was evaluated using several questions. All questions were phrased using a 5-point Likert scale in a positive direction, where *completely agree* was a positive response and *completely disagree* was a negative response (Appendix 1). All physicians enrolled in the study were asked to complete periodic questionnaires, at 2 months, 6 months, and 12 months." [31]

Explanation

We recommend collecting regularly user feedback for further improvements of the patient registry software system. Like in O'Reilly et al. [31], the collected data can be analyzed and it can be shown if the used system had an influence on the disease management. These data can also be used to improve further projects for example like O'Reilly et al. concluded for user trainings how to use a system at the beginning of a new project, [31].

Item 12.4 The system provides an online help for data entry.

Example

"An on-line guide has been designed to help the users in better understanding the meaning of the information to be entered. Its contents have been structured at three different levels, namely context, input form, and single item.

Context-related material explains the rationale for collecting a set of data which can be entered through one or more forms. For example, SUN registry data belong to the "stroke" context, that can be further

refined into narrower contexts such as *emergency, SU admission, discharge and follow-up*. Form-specific help instructs the user about the intents and the features of each input form, relating those to the specific context they belong to. Finally, item-related help addresses any possible ambiguity about the meaning of a specific form item. …

We often added explanatory labels for date and time data, which is very important as some data may be associated to different timestamps. …

In addition to plain explanations used by the on-line help, we also explicitly represented any relationship, based on medical knowledge, useful for detecting treacherous data entry errors …

These relationships have been organized into a semantic network, and are exploited to generate warnings whenever the actual data violate them. …" [10]

Explanation

An online help is an innovative instrument to support end-users in the data entry process. It is immediately available when questions during the data entry process appear and it is more often used than a written manual because it is available on-site. When contexts are organized in a semantic network to provide online help, as described by Lanzola et al. [10], this can help to prevent serious errors which otherwise will not be detected. For example, if a medication is applied which is not related to the disease this can be detected by an online help when contexts are organized in a semantic network. Such errors would otherwise not be detected by simple range checks. We recommend implementing such an online help in patient registry software systems.

References

[1] Xie F, Zhang D, Wu J, Zhang Y, Yang Q, Sun X, Cheng J, Chen X. Design and implementation of the first nationwide, web-based Chinese Renal Data System (CNRDS). BMC Med Inform Decis Mak. 2012;12:11. doi: 10.1186/1472-6947-12-11.

[2] Messiaen C, Le Mignot L, Rath A, Richard JB, Dufour E, Ben Said M, Jais JP, Verloes A, Le Merrer M, Bodemer C, Baujat G, Gerard-Blanluet M, Bourdon-Lanoy E, Salomon R, Ayme S, Landais P. CEMARA: a Web dynamic application within a N-tier architecture for rare diseases. Stud Health Technol Inform. 2008;136:51-6.

[3] Sherman S, Shats O, Ketcham MA, Anderson MA, Whitcomb DC, Lynch HT, Ghiorzo P, Rubinstein WS, Sasson AR, Grizzle WE, Haynatzki G, Feng J, Sherman A, Kinarsky L, Brand RE. PCCR: Pancreatic Cancer Collaborative Registry. Cancer Inform. 2011;10:83-91.

[4] Sherman S, Shats O, Fleissner E, Bascom G, Yiee K, Copur M, Crow K, Rooney J, Mateen Z, Ketcham MA, Feng J, Sherman A, Gleason M, Kinarsky L, Silva-Lopez E, Edney J, Reed E, Berger A, Cowan K. Multicenter breast cancer collaborative registry. Cancer Inform. 2011;10:217-26.

[5] Deserno TM, Haak D, Brandenburg V, Deserno V, Classen C, Specht P. Integrated Image Data and Medical Record Management for Rare Disease Registries. A General Framework and its Instantiation to the German Calciphylaxis Registry. J Digit Imaging. 2014. doi: 10.1007/s10278-014-9698-8.

[6] Gillies MC, Walton R, Liong J, Arnold JJ, McAllister I, Morlet N, Hunyor A, Guymer R, Keeffe J, Essex R, Herrera-Bond A, Glastonbury B, Simpson JM, Barthelmes D. Efficient capture of high-quality data on

outcomes of treatment for macular diseases: the fight retinal blindness! Project. Retina. 2014;34(1):188-95.

[7] Singh SK, Malik A, Firoz A, Jha V. CDKD: a clinical database of kidney diseases. BMC Nephrol. 2012;13:23 doi: 10.1186/1471-2369-13-23.

[8] Natter MD, Quan J, Ortiz DM, Bousvaros A, Ilowite NT, Inman CJ, Marsolo K, McMurry AJ, Sandborg CI, Schanberg LE, Wallace CA, Warren RW, Weber GM, Mandl KD. An i2b2-based, generalizable, open source, self-scaling chronic disease registry. J Am Med Inform Assoc. 2013;20(1):172-9. doi: 10.1136/amiajnl-2012-001042.

[9] Holzner B, Giesinger JM, Pinggera J, Zugal S, Schöpf F, Oberguggenberger AS, Gamper EM, Zabernigg A, Weber B, Rumpold G. The Computer-based Health Evaluation Software (CHES): a software for electronic patient-reported outcome monitoring. BMC Med Inform Decis Mak. 2012;12:126. doi: 10.1186/1472-6947-12-126.

[10] Lanzola G, Parimbelli E, Micieli G, Cavallini A, Quaglini S. Data quality and completeness in a web stroke registry as the basis for data and process mining. J Healthc Eng. 2014;5(2):163-84.

[11] Plotnicki L, Kohl CD, Höcker B, Krupka K, Rahmel A, Pape L, Hoyer P, Marks SD, Webb NJ, Söylemezoglu O, Topaloglu R, Szabo AJ, Seeman T, Marlies Cornelissen EA, Knops N, Grenda R, Tönshoff B. The CERTAIN Registry: a novel, web-based registry and research platform for pediatric renal transplantation in Europe. Transplant Proc. 2013;45:1414-7.

[12] Steurbaut K, De Backere F, Keymeulen A, De Leenheer M, Smets K, De Turck F. NEOREG: design and implementation of an online Neonatal Registration System to access, follow and analyse the data of newborns with congenital cytomegalovirus infection. Inform Health Soc Care. 2013;38(3):223-35.

[13] Lycett K, Wittert G, Gunn J, Hutton C, Clifford SA, Wake M. The challenges of real-world implementation of web-based shared care software: the HopSCOTCH Shared-Care Obesity Trial in Children. BMC Med Inform Decis Mak. 2014;14:61.

[14] Wake M, Lycett K, Clifford SA, Sabin MA, Gunn J, Gibbons K, Hutton C, McCallum Z, Arnup SJ, Wittert G. Shared care obesity management in 3-10 year old children: 12 month outcomes of HopSCOTCH randomised trial. BMJ. 2013 Jun 10;346:f3092.

[15] Sanderson G, Ariyaratne TV, Wyss J, Looi V. A global patient outcomes registry: Cochlear paediatric implanted recipient observational study (Cochlear(™) P-IROS). BMC Ear Nose Throat Disord. 2014;14:10. doi: 10.1186/1472-6815-14-10.

[16] Zargaran E, Schuurman N, Nicol AJ, Matzopoulos R, Cinnamon J, Taulu T, Ricker B, Garbutt Brown DR, Navsaria P, Hameed SM. The electronic Trauma Health Record: design and usability of a novel tablet-based tool for trauma care and injury surveillance in low resource settings. J Am Coll Surg. 2014;218(1):41-50.

[17] da Silva KR, Costa R, Crevelari ES, Lacerda MS, de Moraes Albertini CM, Filho MM, Santana JE, Vissoci JR, Pietrobon R, Barros JV. Glocal clinical registries: pacemaker registry design and implementation for global and local integration--methodology and case study. PLoS One. 2013;8(7):e71090. doi: 10.1371/journal.pone.0071090.

[18]Teng JE, Thomson DR, Lascher JS, Raymond M, Ivers LC. Using Mobile Health (mHealth) and Geospatial Mapping Technology in a Mass Campaign for Reactive Oral Cholera Vaccination in Rural Haiti. PLoS Negl Trop Dis. 2014;8(7):e3050. doi: 10.1371/journal.pntd.0003050.

[19]TMF e.V. 2015. http://www.tmf-ev.de/ (accessed: 15 Apr 2016).

[20]Prasser F, Kohlmayer F, Lautenschläger R, Kuhn KA. ARX - A Comprehensive Tool for Anonymizing Biomedical Data. AMIA Annual Symposium Proceedings. 2014;2014:984-993.

[21]National Cancer Institute. The caBIG program has been retired. https://cabig.nci.nih.gov/. (accessed 12 Feb 2015).

[22]National Cancer Institute. National Cancer Informatics Program. 2015. http://cbiit.nci.nih.gov/ncip. (accessed 15 Apr 2016).

[23]Ko GT, So WY, Tong PC, Le Coguiec F, Kerr D, Lyubomirsky G, Tamesis B, Wolthers T, Nan J, Chan J. From design to implementation--the Joint Asia Diabetes Evaluation (JADE) program: a descriptive report of an electronic web-based diabetes management program. BMC Med Inform Decis Mak. 2010;10:26. doi: 10.1186/1472-6947-10-26.

[24]Zaman B, Khandekar R, Al Shahwan S, Song J, Al Jadaan I, Al Jiasim L, Owaydha O, Asghar N, Hijazi A, Edward DP. Development of a web-based glaucoma registry at King Khaled Eye Specialist Hospital, Saudi Arabia: a cost-effective methodology. Middle East Afr J Ophthalmol. 2014;21(2):182-5.

[25]Pang X, Kozlowski N, Wu S, Jiang M, Huang Y, Mao P, Liu X, He W, Huang C, Li Y, Zhang H. Construction and management of ARDS/sepsis registry with REDCap. J Thorac Dis. 2014;6(9):1293-9. doi: 10.3978/j.issn.2072-1439.2014.09.07.

[26]Wang Y, Tao Z, Cross PK, Le LH, Steen PM, Babcock GD, Druschel CM, Hwang SA. Development of a web-based integrated birth defects surveillance system in New York State. J Public Health Manag Pract. 2008;14(6):E1-E10. doi: 10.1097/01.PHH.0000338377.83736.9c.

[27]Ergonomic requirements for office work with visual display terminals (VDTs) - Part 11: Guidance on usability (ISO 9241–11:1998), 1998. http://www.it.uu.se/edu/course/homepage/acsd/vt09/ISO9241part11.pdf. (accessed 15 Apr 2016).

[28]Shah J, Rajgor D, Pradhan S, McCready M, Zaveri A, Pietrobon R. Electronic data capture for registries and clinical trials in orthopaedic surgery: open source versus commercial systems. Clin Orthop Relat Res. 2010;468(10):2664-71.

[29]Hira AY, Lopes TT, de Mello AN, Filho VO, Zuffo MK, de Deus Lopes R. Establishment of the Brazilian telehealth network for paediatric oncology. J Telemed Telecare. 2005;11 Suppl 2:51-2.

[30]France DJ, Miles P, Cartwright J, Patel N, Ford C, Edens C, Whitlock JA. A chemotherapy incident reporting and improvement system. Jt Comm J Qual Saf. 2003;29(4):171-80. Erratum in: Jt Comm J Qual Saf. 2003;29(5):209.

[31]O'Reilly DJ, Bowen JM, Sebaldt RJ, Petrie A, Hopkins RB, Assasi N, MacDougald C, Nunes E, Goeree R. Evaluation of a chronic disease management system for the treatment and management of diabetes in primary health care practices in Ontario: an observational study. Ont Health Technol Assess Ser. 2014;14(3):1-37.

Anhang B

Konferenzbeiträge

B.1 Vorarbeiten zu Publikation I:

GMDS 2014 Publikation: - nur online -

Lindoerfer, D. and Mansmann, U. (2014). A proposal of Checklist Items for evaluating a Patient Registry Software System. (CC-BY-NC) - Online article.

https://dx.doi.org/10.3205/14gmds124

https://www.egms.de/static/en/meetings/gmds2014/14gmds124.shtml

MIE 2014 Publikation:

Lindoerfer, D. and Mansmann, U. (2014). CIPROS - A checklist with items for a patient registry software system. *Stud Health Technol Inform.*, 205:161-5. (CC-BY-NC)

DOI: 10.3233/978-1-61499-432-9-161

http://ebooks.iospress.nl/publication/37468

e-Health – For Continuity of Care
C. Lovis et al. (Eds.)

doi:10.3233/978-1-61499-432-9-161

CIPROS - A checklist with items for a patient registry software system

Doris LINDOERFER[a,1] and Ulrich MANSMANN[a]
[a] *Institut für medizinische Informationsverarbeitung, Biometrie und Epidemiologie, Ludwig-Maximilians-Universität München*

Abstract. Patient registries are an important instrument in medical research. Their implementation uses complex software systems to meet the wide spectrum of challenges. There is a wide range of systems available and a critical appraisal of their architecture is needed. Therefore, we developed, based on a systematic review of the literature, a checklist for patient registry software systems (CIPROS) which is organised in 67 items and 10 sections. CIPROS supports developers to assess requirements of an existing system. It also supports the reporting of patient registry software system descriptions in papers and it can be a first step to create standards for patient registry software systems.

Keywords. Checklist, patient registry software system, technical requirements

Introduction

Patient Registries are essential for medical research, clinical epidemiology, and quality management. The successful implementation and use of a registry depends on a thoroughly and accurate planning and construction of a suitable IT infrastructure. A definition of the registry is needed as well as descriptions how to plan and run the registry. Focus should also be the legal and ethical aspects as well as the strategies of patient recruitment [1]. Müller et al. [2] summarized these points in a checklist for Registries for Health Services Research.

In spite the availability of commercial and open-source software systems, many institutions develop own solutions which fit their needs and they are flexible to address changing requirements. Therefore, guidance is needed to support assessment and selection of an existing, or to plan the development of a new registry software system.

To this end, we developed CIPROS (a **c**hecklist with **i**tems for a **p**atient **r**egistry s**o**ftware **s**ystem) and present it in the style of an explanation and elaboration paper which is inspired by [3]. The checklist is derived from a systematic review of the literature and own experiences in this field [4].

Especially three initiatives with the aim to support clinical research and interoperability in providing adequate software systems and semantic infrastructure

[1] Corresponding Author: Doris Lindörfer, Dipl.-Inf.; Institut für medizinische Informationsverarbeitung, Biometrie und Epidemiologie, Ludwig-Maximilians Universität München, Marchioninistrasse 15, D-81377 München, Germany; Email: lindoerf@ibe.med.uni-muenchen.de

inspired our work: (1) The caBIG® program from the National Cancer Institute [5] certified existing software systems, according special criteria with bronze, silver, and gold; (2) The recent National Cancer Informatics Program [6] aims to support interoperability in Medical Research in providing open-source code for biomedical informatics software applications and in providing semantic infrastructures; (3) A National Metadata Repository for Clinical Studies and Registries is provided by TMF e.V. [7]. There is also an ongoing project at [7] which defines common requirements for cohorts and registry-IT.

1. Methods

Our Systematic Review identified papers with patient registry software system descriptions. In order to get exhaustive results on papers containing patient registry software system descriptions the following search in Pubmed was applied: *"(registry or registries) AND (eCRF or EDC or CDMS or CTMS or web) AND (software or open-source or open source or Java)"*. This search resulted in 132 findings when performed at 17th of January 2014.

We classified the 132 papers according to their content by reading title and the abstract. A total of 42 (32%) provided a system description or at least a short system description. A total of 23 (17%) were about biological information systems, 23 (17%) were about infrastructures for clinical or other medical systems, 30 (22%) deal with other aspects and had no system description. Eleven papers (9%) were only published as abstract; three papers (3%) were written in languages other than English or German.

A qualitative analysis was performed on the 42 papers containing a system description by searching them for relevant software specific features. We tried to classify the identified software specific features in system components, functional aspects, or design steps. The features come mainly from the more actual papers (published after 2005). Older papers (published before 2005) with system descriptions did not provide essential aspects used for the item creation process.

2. Results

The qualitative analysis elicited 67 different items which are involved in describing registry software systems. Inspired by the caBIG® [5] criteria we created a checklist organized in ten logical system components, functional aspects or design steps which is presented in subsection 2.1.

2.1. Checklist of Items - without Description

1. *Architecture*: System architecture, Platform independence
2. *Development and Training*: Design model, Framework-based design, Questionnaire builder, Usability testing, Performance testing, User manual, User training, User feedback
3. *Interfaces*: *End-user Interfaces*: Web-Interface, Compatibility, Email-alerts, Messaging interface, Online discussion forum, Mobile interface, Patient interface; *Programming interface*: Third-party access, API for inserting data,

API for retrieving data, Data update mechanism; *Interfaces to other systems*: Interface to CIS, Integration of biological data, Extensibility is possible

4. *Interoperability*: Open Source; *Semantics and Standardization*: CRFs, Metadata, Vocabularies, XML Schema
5. *Internationality*: Multilingualism
6. *Data management and data quality*: Pseudonymous patient identifier, CRF is divided in parts, Customizable CRF parts, Minimal and extended dataset, All data types are supported, Special data types are possible, Multiple choice is used, No predefined selection, Data validation components, Data query tool, Interface for manual data check, Manual data queries, Data query flags, Plausibility flags, Insertion of unplanned visits.
7. *Data Analysis*: Query builder for researches, Report generation, Download of datasets, Graphical presentation of results, Risk analysis
8. *Security aspects*: Authorized users, Role-based access, Encrypted data transfer, Encrypted data storage, Audit trail, Master-slave replication, Backup-management, Firewall, Server room.
9. *General Features*: Costs are an argument, several projects simultaneously, Update mechanism, Source documentation in pdf.
10. *Organizational*: Compliance with regulations, Informed Consent, Rights on the data, data protection guidelines.

The following subsection 2.2 represents for the items 4.2 - 4.5 a short Description and the relevant references.

2.2. A Part of the Checklist of Items - with Description

Table 1. A Part of the Checklist of Items to consider when choosing or developing a software system for patient registries.

Aspect / Topic	Item No.	Description	Relevant References
Interoperability	**4**		
…			
Semantics & Standardization			
CRFs	4.2	Standardized CRFs are used whenever possible.	[9], [10]
Metadata	4.3	Ontology-based, standardized metadata are used.	[8], [9]
Vocabularies	4.4	Ontology-based, standardized vocabularies are used.	[8], [9]
XML Schema	4.5	An XML schema definition (XSD) is available for structured data exchange.	[11], [12], [13]

The list of the 42 reference papers used to build the list and the Complete Table with Checklist Items and Description are available from the corresponding author.

For the items 4.2 - 4.5 in Table 1, we describe in the following subsection the item creation process.

2.3. Item Creation Process of some Items

The literature sees standardization as an essential aspect of a registry and points to different aspects of standardization which are summarized in items 4.2 to 4.5 The

seriation of the four items is open to discussion. We chose the proposed order following a time-perspective of data capture, data storage, and exchange of data.
One Requirement of the caBIG® [5] criteria is the use of standardized vocabularies / terminologies and ontologies (as a prerequisite to facilitate cooperation by better data transfer within different registries). Natter et al. [8] describe a shared ontology service in their project which enables communication between different components. In [9] and [10] Sherman et al. describe the use of controlled terminologies and metadata in their registry projects to get the caBIG® Bronze certification. Therefore, we created Items 4.3 and 4.4 for standardized vocabularies and metadata. The standardization of questionnaires is also described and proposed in [9] and [10] we created Item 4.2 for this issue.

Xie et al. describes in [11] the data integration from different previous registries in their new registry using structured data exchange with XML. Messiaen et al. describe in [12] structured data exchange between different Registries with XML. Plotnicki et al. describe this feature also in [13] to allow the data transfer within different organizations and to avoid repeated data entry. We created Item 4.5 to support this important feature.

In the following subsection we present one Item out of the full Example and the Explanation selection.

2.4. Example and Explanation of Item 3.1

Item 3.1 The patient registry software system has a web-interface.
Example:

The CERTAIN web application … supports therefore not only the data entry, presentation, visualization, and export, but also the automatic and manual data validation. These functionalities are active at any time and location, requiring only a common web browser and internet access. [13]
Explanation:

The literature identifies a web-interface for patient registry software systems as the proper way to communicate with distributed users. Web-based data entry is a comfortable and a state of the art way to collect patient data for multi-center registries. All data are immediately available in the database. First quality checks can automatically be performed. The data from all centers are collected in the same way, no specific data-integration with the data from different centers is necessary. No additional software must be installed at the client side. When a software update is necessary, it must only be installed at the server, the clients get a new version when they log in next time.

3. Discussion

The aim of the CIPROS checklist is to provide study coordinators, developers and decision makers with an instrument to evaluate existing systems while checking it by the proposed items. All presented features should be checked for relevance when embarking in the development of a registry project. They also are valuable in choosing an existing system or in developing an own solution. Another aim of our checklist is to provide persons reporting about patient registry software systems an instrument to check and present the features of their system. Like the CONSORT Statement [3],

which improved the reporting of clinical trials by focusing on its relevant aspects, our checklist can help, making the description of a patient registry software system description more transparent. The CIPROS checklist can help to create standards in reporting registry-based medical research, for which is also a need.

Since compared to the literature on clinical studies the literature on registry software systems is not well structured, it may be the case that our systematic review missed some relevant work. However, the list proposed is seen as a starting point to discuss the relevant features of such systems and to develop a more structured presentation of software systems for patient registries.

This checklist is a proposal which considers the actual situation. Maybe in the future other techniques become popular which recommend an update of this checklist.

References

[1] Gliklich RE, Dreyer NA, eds. *Registries for Evaluating Patient Outcomes: A User's Guide, Second Edition.* Rockville, AHRQ Publication No. 10-EHC049, 2010.

[2] Müller D, Augustin M, Banik N, Baumann W, Bestehorn K, Kieschke J, Lefering R, Maier B, Mathis S, Rustenbach SJ, Sauerland S, Semler SC, Stausberg J, Sturm H, Unger C, Neugebauer EAM. Memorandum Register für die Versorgungsforschung. *Das Gesundheitswesen* **72** (2010), 824-839.

[3] Altman DG, Schulz KF, Moher D, Egger M, Davidoff F, Elbourne D, Gøtzsche PC, Lang T; CONSORT GROUP (Consolidated Standards of Reporting Trials). The revised CONSORT statement for reporting randomized trials: explanation and elaboration. *Ann Intern Med.* **134** (2001), 663-694.

[4] Lindörfer D, Müller TH, Integration und Analyse von Patienten mit Chronischer Myeloischer Leukämie (CML) in 25 Europäischen Ländern mit webbasierter elektronischer Datenerfassung. In: Schmücker P, Elsässer K-H, Hayna S (Hrsg.): *Tagungsband der 55. GMDS Jahrestagung 2010,* Mannheim, 2010.

[5] National Cancer Institute. The caBIG program has been retired. Available: https://cabig.nci.nih.gov/. (Accessed, January 13, 2014).

[6] National Cancer Institute. National Cancer Informatics Program. Available: http://cbiit.nci.nih.gov/ncip. (Accessed January 13, 2014).

[7] TMF e.V. Available: http://www.tmf-ev.de/ (Accessed Jan, 23 2014).

[8] Natter MD, Quan J, Ortiz DM, Bousvaros A, Ilowite NT, Inman CJ, Marsolo K, McMurry AJ, Sandborg CI, Schanberg LE, Wallace CA, Warren RW, Weber GM, Mandl KD. An i2b2-based, generalizable, open source, self-scaling chronic disease registry. *J Am Med Inform Assoc.* **20(1)** (2013) 172-179.

[9] Sherman S, Shats O, Fleissner E, Bascom G, Yiee K, Copur M, Crow K, Rooney J, Mateen Z, Ketcham MA, Feng J, Sherman A, Gleason M, Kinarsky L, Silva-Lopez E, Edney J, Reed E, Berger A, Cowan K. Multicenter breast cancer collaborative registry. *Cancer Inform.* **10** (2011), 217-226.

[10] Sherman S, Shats O, Ketcham MA, Anderson MA, Whitcomb DC, Lynch HT, Ghiorzo P, Rubinstein WS, Sasson AR, Grizzle WE, Haynatzki G, Feng J, Sherman A, Kinarsky L, Brand RE. PCCR: Pancreatic Cancer Collaborative Registry. *Cancer Inform.* **10** (2011) 83-91.

[11] Xie F, Zhang D, Wu J, Zhang Y, Yang Q, Sun X, Cheng J, Chen X. Design and implementation of the first nationwide, web-based Chinese Renal Data System (CNRDS). *BMC Med Inform Decis Mak.* 2012, **12**:11.

[12] Messiaen C, Le Mignot L, Rath A, Richard JB, Dufour E, Ben Said M, Jais JP, Verloes A, Le Merrer M, Bodemer C, Baujat G, Gerard-Blanluet M, Bourdon-Lanoy E, Salomon R, Ayme S, Landais P. CEMARA: a Web dynamic application within a N-tier architecture for rare diseases. *Stud Health Technol Inform.* **136 (**2008) 51-56.

[13] Plotnicki L, Kohl CD, Höcker B, Krupka K, Rahmel A, Pape L, Hoyer P, Marks SD, Webb NJ, Söylemezoglu O, Topaloglu R, Szabo AJ, Seeman T, Marlies Cornelissen EA, Knops N, Grenda R, Tönshoff B. The CERTAIN Registry: a novel, web-based registry and research platform for pediatric renal transplantation in Europe. *Transplant Proc.* **45** (2013), 1414-1417.

B.2 Vorarbeiten zu Publikation II:

HEC 2016 Publikation:

Lindoerfer, D. and Mansmann, U. (2016). Proposing an Evidence-Based Strategy for Software Requirements Engineering. *Stud Health Technol Inform.*, 228:648-52. (CC-BY-NC 4.0)

DOI: 10.3233/978-1-61499-678-1-648

`http://ebooks.iospress.nl/publication/44693`

Exploring Complexity in Health: An Interdisciplinary Systems Approach
A. Hoerbst et al. (Eds.)

doi:10.3233/978-1-61499-678-1-648

Proposing an Evidence-Based Strategy for Software Requirements Engineering

Doris LINDOERFER[1] and Ulrich MANSMANN
Institute for Medical Information Processing, Biometry and Epidemiology (IBE), Ludwig-Maximilians-University Munich, Munich, Germany

Abstract. This paper discusses an evidence-based approach to software requirements engineering. The approach is called evidence-based, since it uses publications on the specific problem as a surrogate for stakeholder interests, to formulate risks and testing experiences. This complements the idea that agile software development models are more relevant, in which requirements and solutions evolve through collaboration between self-organizing cross-functional teams. The strategy is exemplified and applied to the development of a Software Requirements list used to develop software systems for patient registries.

Keywords. Checklist, patient registry software system, software requirements engineering

1. Introduction

Software projects need to be planned following a life-cycle model that defines the necessary steps which must be done during the project time and move forward [1]. The first published software life-cycle model was the Waterfall model [2], which defined the necessary steps in a sequential order. An extension of the Waterfall model is the V-Model [3], which introduced test phases in the process to verify and validate the planned steps. The Spiral model [4] is a risk-driven enhancement of the Waterfall model which defines the iterative production of prototypes to verify the planned goals. The Sawtooth model [5] tries to bring developers and clients together by prototype demonstrations. The Shark Tooth Model [1] is a refinement of the Sawtooth model by taking management reviews and demonstrations also into account. Today agile software development models [6] are more relevant, in which requirements and solutions evolve through collaboration between self-organizing cross-functional teams.

In order to make the interaction between stakeholder and software developer more efficient, we follow the idea that the literature offers a large series of reports on experiences, failures, solutions, and concepts. This corresponds to concentrated expert knowledge of cross-functional teams. We propose to condense this reported experience in an instrument which helps to organize software requirement engineering for a specific field.

[1] Corresponding Author: Doris Lindoerfer, Dipl.-Inf.; Institute for Medical Information Processing, Biometry and Epidemiology (IBE), Ludwig-Maximilians University Munich, Marchioninistrasse 15, D-81377 Munich, Germany; Email: lindoerf@ibe.med.uni-muenchen.de

Our use-case is the field of clinical patient registries, in general. We mainly focus at research oriented patient registries according the definition of Glicklich et al. [7]. We intend to perform a systematic review of the corresponding literature available. We use the literature to formulate requirements elicitation, specification and analysis. To this end we use qualitative content analysis (QCA) as evidence-based approach to the process of requirements elicitation. The qualitative content analysis follows ideas of Mayring [8].

This paper outlies our strategy and presents an application, the CIPROS [9] checklist, which is constructed to assist in evolving software systems for patient registries. We compare this approach with the more conventional software development strategies. We call our approach "evidence-based" because it is based on systematic literature review (SLR) and uses qualitative methods for the corresponding meta-analysis.

2. Methods

2.1. Traditional Software Requirements Engineering Methods

The most popular method to identify and gather the requirements of a new software product, are **interviews** with the involved people, also named as stakeholders. From these interviews, scenarios and use-cases can be derived. In a more advanced project phase these use-cases are transferred into class diagrams and will then be implemented. **Expert panels** and **workshops** are also important and widespread methods to elicit and define the requirements of a new software product. This guarantees that experiences from different persons and projects will be considered. Of course **own experiences** play also an important role.

2.2. A selection of Software Requirements Specification Templates

There exist a lot of Software Requirements Specification (SRS) Templates. Some are freely available on the Internet, others can be purchased. The following list gives some examples of Software Specification Templates which can be found on the Internet.

- The ISO/IEC/IEEE 29148:2011 International Standard [10].
- Wiegers KE. Software Requirements Specification [11].
- McKinnon AD. Software Requirements Specification Template [12].
- Volere Requirements Specification Template [13].

The ISO/IEC/IEEE 29148:2011 International Standard [10] is a comprehensive document which first describes detailed how the requirements should be written in a SRS document, which standards are applicable, which abbreviations are used how the requirements analysis process should be applied, etc. It distinguishes between stakeholder requirements (StRS), system requirements (SyRS) and software requirements (SRS). It includes also requirements management and change management. This standard can be purchased.

The Software Requirements Specification Template published by KE. Wiegers [11], in 1999, is a document which provides a structure for the specification of the project specific content and requirements and it is freely available on the Internet.

The Software Requirements Specification Template published by AD. McKinnon [12], published 2005, is also a document with a structure for the specification of project specific content and requirements. It is freely available on the Internet.

The Volere Requirements Specification Template [13], published by James and Suzanne Robertson and developed from 1995-2015, is a template for specifying project specific content and requirements. It can be purchased from the Volere Webpage [13].

2.3. Creation of an evidence-based Checklist by Qualitative Content Analysis

To create the CIPROS checklist we performed a multistep procedure:

- A systematic literature review
- A qualitative content analysis
- Creation of the checklist

First we performed a systematic literature review (SLR). Because we want to get high qualitative results we decided to perform the search on already published papers which appeared in PubMed. This guarantees that the software systems are used on real-life medical research projects. To get exhaustive results we performed the following search in PubMed: *"(registry or registries) AND (eCRF or EDC or CDMS or CTMS or web) AND (software or open-source or open source or Java)"*.

The search terms were chosen in an iterative process: We are interested in registries with eCRFs or EDC technology or CDMS or CTMS systems and we are mainly interested in software systems or open-source systems, we decided to include also Java in our search question, because it is used in some paper titles on software systems.

The second step was a qualitative content analysis (QCA) according to Mayring [8] which consists first of the step model of inductive category development, which is shown in Figure 1. According to the steps in Figure 1, we first formulated the research question: "Is there a description of software related features?" We read the title and the abstract, if the paper contained a software system description we read the complete paper. We collected all features as items in a list, build inductive categories out of the material, categorized the items or formulated new categories. After 10-50% of the material we revised the categories. Than we read the remaining papers.

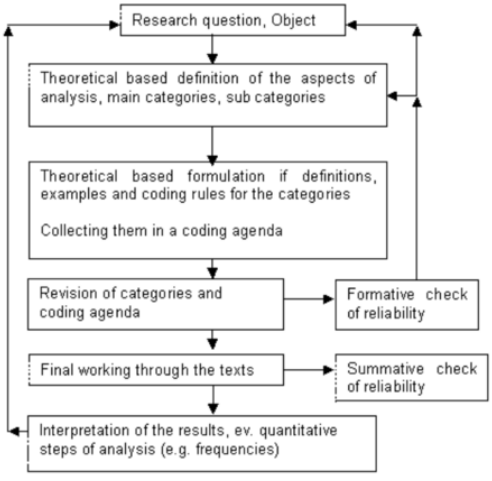


Figure 1. Inductive category development [8].

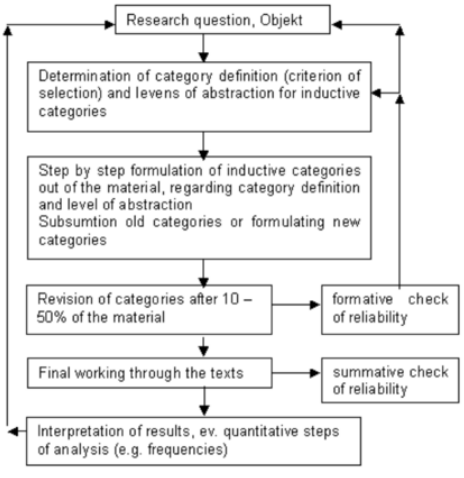


Figure 2. Deductive category application [8].

The third step was the creation of an early version of the checklist. Therefore we performed the step model of deductive category application, which is described by Mayring [8] and is shown in Figure 2. According to the steps in Figure 2 we build main categories and sub-categories and assorted the items to these categories. If necessary we revised the categories and assorted the items to the new categories. We enhanced the checklist with items which we found in none of the papers, but we counted as important by our own experiences.

3. Results

Our search delivered 156 findings when performed at 1st December 2014. A flowchart according to the PRISMA statement [14] is given in Figure 3. 64 of the papers were included in the analysis, 92 papers were excluded the reasons are given in Figure 3. Since the review is of qualitative nature and studies multiple aspects of software systems, the PRISMA checklist is not fully applicable. A qualitative content analysis according to Mayring [8] was performed and enhanced with own experiences. Finally we established the CIPROS checklist. Due to the limited space in this abstract we cannot present the CIPROS checklist here. The CIPROS checklist which is published by Lindoerfer and Mansmann [9] consists of 72 items which are organized in twelve logical system components, functional aspects or design steps.

Because the items in the CIPROS checklist are derived from already published papers we call our approach evidence-based. Evidence-based, because it is based on systems which are used in real-life medical research projects which are published in PubMed-indexed journals. This differentiates our approach from all the other SRS templates. All of the cited SRS templates are general, they are not content specific and they don't provide items from the literature.

Some of the more general items in our checklist are also covered by the SRS templates, but all the patient-registry specific items are only provided by our checklist.

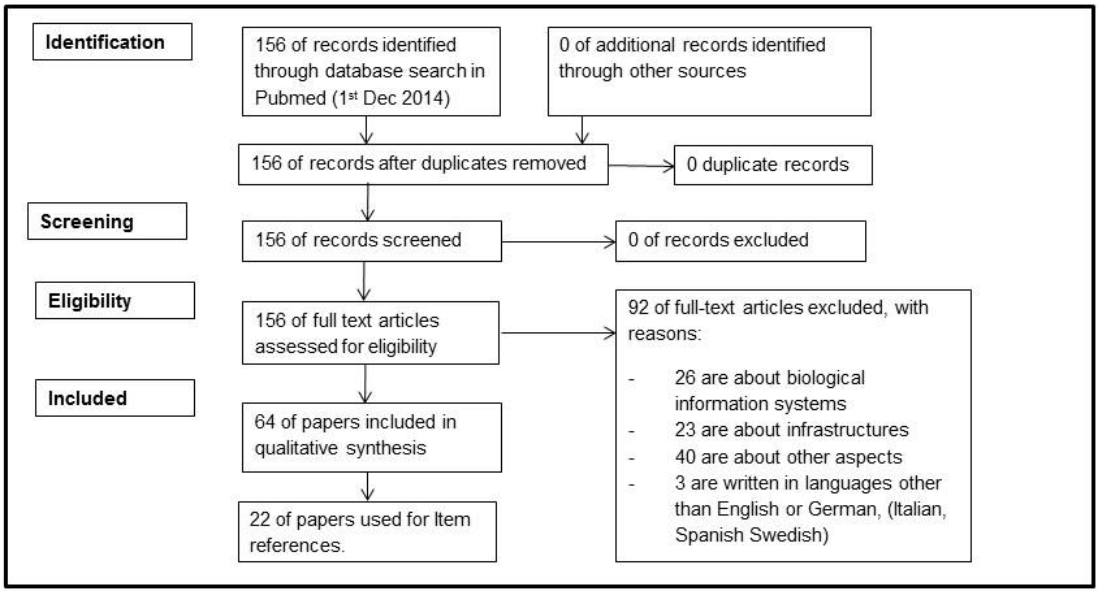


Figure 3. PRISMA Flowchart, explaining the number of selected papers.

4. Conclusion

With our CIPROS checklist [9] we provide an instrument for requirements engineering in software projects for patient registry software systems. It provides 72 items which are organized in twelve logical system components, functional aspects or design steps. The items can be used for the creation of a new patient registry software system or they can be checked when a new system shall be selected. It is not a SRS Template, it will not replace existing SRS Templates, but it can help to create new systems, by accelerating the requirements engineering process, because it brings the already published treasure trove of experiences into the development process of new software systems for patient registries. Our approach can be a template for creating similar checklists in other fields.

A literature review is helpful for a systematic overview on opinions and experiences created for a specific field. This provides an important knowledge when planning new projects. A limitation of our approach is the restricted scope of our literature research. Our focus was on papers available in PubMed. This excludes books and specific conference proceedings.

References

[1] Bruegge B, Dutoit AH, Object-Oriented Software Engineering: Conquering Complex and Changing Systems, ISBN 0-13-489725-0 Prentice Hall, 2000.
[2] Royce WW, 1970, "Managing the Development of Large Software Systems: Concepts and Techniques." In *IEEE WESTCON*, 1-9.
[3] V-Model. Available: https://en.wikipedia.org/wiki/V-Model_%28software_development%29. Accessed 10 Feb 2016.
[4] Boehm B, A Spiral Model of Software Development and Enhancement. In: *ACM SIGSOFT Software Engineering Notes (ACM)* **11**(4):14-24, Aug 1986.
[5] Rowen RB, Software project management under incomplete and ambiguous specifications. *IEEE Transactions on Engineering Management*, Vol. **37**, No. 1, 1990.
[6] Agile software development, Available: https://en.wikipedia.org/wiki/Agile_software_development. Accessed 10 Feb 2016.
[7] Gliklich RE, Dreyer NA, eds. Registries for Evaluating Patient Outcomes: A User's Guide, Second Edition. Rockville, *AHRQ Publication No. 10-EHC049*, 2010.
[8] Mayring P, Qualitative Content Analysis. *Forum Qualitative Sozialforschung / Forum: Qualitative Social Research*, Volume **1**, No. 2, Art. 20, June 2000. ISSN 1438-5627. http://www.qualitative-research.net/index.php/fqs/article/view/1089/2386. Accessed 10 Feb 2016.
[9] Lindoerfer D, Mansmann U, A Comprehensive Assessment Tool for Patient Registry Software Systems: The CIPROS Checklist, *Methods Inf Med*. 2015 Oct 12;**54**(5):447-54. doi: 10.3414/ME14-02-0026.
[10] IEEE STANDARD. 29148-2011 - Systems and software engineering -- Life cycle processes -- Requirements engineering. Available at: https://standards.ieee.org/findstds/standard/29148-2011.html. Accessed 4 Mar 2016
[11] Wiegers KE, Software Requirements Specification, Available: http://tinyurl.com/zrwbz9g Accessed 10 Feb 2016
[12] McKinnon AD, Software Requirements Specification Template., Available: http://users.tricity.wsu.edu/~mckinnon/cpts322/cpts322-srs-v1.doc. Accessed 10 Feb 2016
[13] Robertson J., Robertson S, Volere Requirements Specification Template, Available: http://www.volere.co.uk/template.htm. (Accessed 10 Feb 2016)
[14] Moher D, Liberati A, Tetzlaff J, Altman DG, The PRISMA Group. Preferred Reporting Items for Systematic Reviews and Meta-Analyses: The PRISMA Statement, *PLOS Medicine* (2009);6:7.

Abbildungsverzeichnis

Literaturverzeichnis

Altman, D. G., Schulz, K. F., Moher, D., Egger, M., Davidoff, F., Elbourne, D., Gøtzsche, P. C., & Lang, T. (2001). The revised CONSORT statement for reporting randomized trials: explanation and elaboration. *Ann. Intern. Med.*, 134(8), 663–694.

Anazawa, T., Miyata, H., & Gotoh, M. (2015). Cancer registries in Japan: National Clinical Database and site-specific cancer registries. *Int. J. Clin. Oncol.*, 20(1), 5–10.

Aufbau des EPRD (2021). Aufbau und Arbeit des EPRD. `https://www.eprd.de/de/ueber-uns/organisation/aufbau-des-eprd/`. [Online; Accessed 29-Jun-2021].

Bellgard, M., Beroud, C., Parkinson, K., Harris, T., Ayme, S., Baynam, G., Weeramanthri, T., Dawkins, H., & Hunter, A. (2013). Dispelling myths about rare disease registry system development. *Source Code Biol Med*, 8(1), 21.

Bellgard, M. I., Napier, K. R., Bittles, A. H., Szer, J., Fletcher, S., Zeps, N., Hunter, A. A., & Goldblatt, J. (2017a). Design of a framework for the deployment of collaborative independent rare disease-centric registries: Gaucher disease registry model. *Blood Cells Mol. Dis.*

Bellgard, M. I., Walker, C. E., Napier, K. R., Lamont, L., Hunter, A. A., Render, L., Radochonski, M., Pang, J., Pedrotti, A., Sullivan, D. R., Kostner, K., Bishop, W., George, P. M., O'Brien, R. C., Clifton, P. M., Bockxmeer, F. M. V., Nicholls, S. J., Hamilton-Craig, I., Dawkins, H. J., & Watts, G. F. (2017b). Design of the Familial Hypercholesterolaemia Australasia Network Registry: Creating Opportunities for Greater International Collaboration. *J. Atheroscler. Thromb.*, 24(10), 1075–1084.

Bluhmki, T., Bramlage, P., Volk, M., Kaltheuner, M., Danne, T., Rathmann, W., & Beyersmann, J. (2017). Time-to-event methodology improved statistical evaluation in register-based health services research. *J Clin Epidemiol*, 82, 103–111.

Boehm, B. W. (1986). A spiral model of software development and enhancement. *ACM SIGSOFT SOFTWARE ENGINEERING NOTES*, 11(4), 14–24.

Bruegge, B. & Dutoit, A. H. (2000). *Object-Oriented Software Engineering. Conquering complex and changing systems*. Upper Saddle River, NJ: Prentice Hall.

Bruegge, B. & Dutoit, A. H. (2014). *Object-Oriented Software Engineering Using UML, Patterns and Java*. Harlow, UK: Pearson Education Limited, third edition.

caBIG (2015). National Cancer Institute. The caBIG program has been retired. `https://cabig.nci.nih.gov/`. [Online; Accessed 12-Feb-2015].

Cristancho Triana, G. J., Ninco Hernández, F. A., Cancino Gómez, Y. A., Alfonso Orjuela, L. C., & Ochoa Daza, P. E. (2021). Aspectos clave del plan de negocios para emprender en el contexto colombiano. *Suma de Negocios*, 12, 41 – 51.

Dar, H., Lali, M., Ashraf, H., Ramzan, M., Amjad, T., & Shahzad, B. (2018). A Systematic Study on Software Requirements Elicitation Techniques and Its Challenges in Mobile Application Development. *IEEE Access*, 6, 63859–63867.

DSGVO (2018). Datenschutz-Grundverordnung DSGVO. https://dsgvo-gesetz.de/. [Online; Accessed 14-Jul-2021].

EPRD (2021). EPRD - Endoprothesenregister Deutschland. http://www.eprd.de/. [Online; Accessed 29-Jun-2021].

European Medicines Agency (2015). Initiative for patient registries. strategy and pilot phase. https://www.ema.europa.eu/en/documents/other/initiative-patient-registries-strategy-pilot-phase_en.pdf. [Online; Accessed 05-Jul-2021].

European Medicines Agency (2018). Use of disease registries for benefit risk evaluation of medicines: A regulatory perspective. https://www.ema.europa.eu/en/documents/presentation/presentation-registry-initiative-april-2018_en.pdf. [Online; Accessed 05-Jul-2021].

European Medicines Agency (2020). Guideline on registry-based studies - draft. https://www.ema.europa.eu/en/documents/scientific-guideline/guideline-registry-based-studies_en.pdf. [Online; Accessed 07-Jul-2021].

European Medicines Agency (2021). European medicines agency. https://www.ema.europa.eu/en. [Online; Accessed 17-May-2021].

European Medicines Agency, The Cross-Committee Task Force on Patient Registries (2018). Discussion paper: Use of patient disease registries for regulatory purposes - methodological and operational considerations. https://www.segcib.org/ema-consultation-discussion-paper-use-of-patient-disease-registries-for-regulatory-purposes-methodological-and-operational-considerations/. [Online; Accessed 05-Jul-2021].

EUTOS (2011). EUTOS for CML. http://www.eutos.org/content/home/index_eng.html. [Online; Accessed 23-Jun-2021].

GDPR (2018). General Data Protection Regulation GDPR. https://gdpr-info.eu/. [Online; Accessed 14-Jul-2021].

Glicklich, R., Levy, D., Karl, J., Leavy, M. B., Taylor, T., & Campion, D. M. (2012). Registry of Patient Registries (RoPR): Project Overview. https://effectivehealthcare.ahrq.gov/topics/registry-of-patient-registries/research-2012-3. [Online; Accessed 07-Jul-2021].

Glicklich, R. E. & Dreyer, N. A., Eds. (2010). *Registries for Evaluating Patient Outcome: A Users Guide*. Rockville: AHRQ Publication No. 10-EHC049, second edition.

GoogleScholar (2021). Google Scholar - Doris Lindoerfer. https://scholar.google.com/scholar?hl=de&as_sdt=0%2C5&q=Doris+Lindoerfer&oq=. [Online; Accessed 09-Aug-2021].

Hasford, J., Baccarani, M., Hoffmann, V., Guilhot, J., Saussele, S., Rosti, G., Guilhot, F., Porkka, K., Ossenkoppele, G., Lindoerfer, D., Simonsson, B., Pfirrmann, M., & Hehlmann, R. (2011). Predicting complete cytogenetic response and subsequent progression-free survival in 2060 patients with CML on imatinib treatment: the EUTOS score. *Blood*, 118(3), 686–692.

Hoffmann, V., Baccarani, M., Hasford, J., Guilhot, J., Saussele, S., Rosti, G., Guilhot, F., Porkka, K., Ossenkoppele, G., Lindoerfer, D., Simonsson, B., Pfirrmann, M., & Hehlmann, R. (2012). The EUTOS CML score aims to support clinical decision-making. *Blood*, 119(12), 2966–2967.

Hoffmann, V. S., Baccarani, M., Hasford, J., Castagnetti, F., Di Raimondo, F., Casado, L. F., Turkina, A., Zackova, D., Ossenkoppele, G., Zaritskey, A., Hoglund, M., Simonsson, B., Indrak, K., Sninska, Z., Sacha, T., Clark, R., Bogdanovic, A., Hellmann, A., Griskevicius, L., Schubert-Fritschle, G., Sertic, D., Guilhot, J., Lejniece, S., Zupan, I., Burgstaller, S., Koskenvesa, P., Everaus, H., Costeas, P., Lindoerfer, D., Rosti, G., Saussele, S., Hochhaus, A., & Hehlmann, R. (2017). Treatment and outcome of 2904 CML patients from the EUTOS population-based registry. *Leukemia*, 31(3), 593–601.

Hoffmann, V. S., Baccarani, M., Hasford, J., Lindoerfer, D., Burgstaller, S., Sertic, D., Costeas, P., Mayer, J., Indrak, K., Everaus, H., Koskenvesa, P., Guilhot, J., Schubert-Fritschle, G., Castagnetti, F., Di Raimondo, F., Lejniece, S., Griskevicius, L., Thielen, N., Sacha, T., Hellmann, A., Turkina, A. G., Zaritskey, A., Bogdanovic, A., Sninska, Z., Zupan, I., Steegmann, J. L., Simonsson, B., Clark, R. E., Covelli, A., Guidi, G., & Hehlmann, R. (2015). The EUTOS population-based registry: incidence and clinical characteristics of 2904 CML patients in 20 European Countries. *Leukemia*, 29(6), 1336–1343.

Hoffmann, V. S., Baccarani, M., Lindoerfer, D., Castagnetti, F., Turkina, A., Zaritsky, A., Hellmann, A., Prejzner, W., Steegmann, J. L., Mayer, J., Indrak, K., Colita, A., Rosti, G., & Pfirrmann, M. (2013). The EUTOS prognostic score: review and validation in 1288 patients with CML treated frontline with imatinib. *Leukemia*, 27(10), 2016–2022.

IEEE STANDARD. 29148-2011 (2011). IEEE STANDARD. 29148-2011 - Systems and software engineering – Life cycle processes –Requirements engineering. `https://standards.ieee.org/findstds/standard/29148-2011.html`. [Online; Accessed 23-Jun-2021].

Jäger, H. R., Mansmann, U., Hausmann, O., Partzsch, U., Moseley, I. F., & Taylor, W. J. (2000). MRA versus digital subtraction angiography in acute subarachnoid haemorrhage: a blinded multireader study of prospectively recruited patients. *Neuroradiology*, 42(5), 313–326.

kbv (2021). Klinische Krebsregister. `http://www.kbv.de/html/krebsregister.php`. [Online; Accessed 30-Jun-2021].

Kodra, Y., Posada de la Paz, M., Coi, A., Santoro, M., Bianchi, F., Ahmed, F., Rubinstein, Y. R., Weinbach, J., & Taruscio, D. (2017). Data Quality in Rare Diseases Registries. *Adv. Exp. Med. Biol.*, 1031, 149–164.

Kodra, Y., Posada de la Paz, M., Coi, A., Santoro, M., Bianchi, F., Rubinstein, Y. R., WEINBACH, J., & Taruscio, D. (2016). Data Quality in Rare Diseases Registries. `http://www.radico.fr/images/Article_Data_Quality_Sept_2016.pdf`. [Online; Accessed 23-Jun-2020].

Kodra, Y., Weinbach, J., Posada-de-la Paz, M., Coi, A., Lemonnier, S. L., van Enckevort, D., Roos, M., Jacobsen, A., Cornet, R., Ahmed, S. F., Bros-Facer, V., Popa, V., Van Meel, M., Renault, D., von Gizycki, R., Santoro, M., Landais, P., Torreri, P., Carta, C., Mascalzoni, D., Gainotti, S., Lopez, E., Ambrosini, A., Muller, H., Reis, R., Bianchi, F., Rubinstein, Y. R., Lochmuller, H., & Taruscio, D. (2018). Recommendations for Improving the Quality of Rare Disease Registries. *Int J Environ Res Public Health*, 15(8).

Kossmann, M. (2014). The significance of requirements engineering for the medical domain. *J Med Eng Technol*, 38(5), 238–243.

Lindoerfer, D. & Mansmann, U. (2014a). A proposal of Checklist Items for evaluating a Patient Registry Software System. GMDS 2014. 59. Jahrestagung der Deutschen Gesellschaft für Medizinische Informatik, Biometrie und Epidemiologie e.V. (GMDS). Göttingen, 07.-10.09.2014. Düsseldorf: German Medical Science GMS Publishing House; 2014. DocAbstr. 149 (Abstr. 149). `http://www.egms.de/static/en/meetings/gmds2014/14gmds124.shtml`. [Online; Accessed 07-Jul-2021].

Lindoerfer, D. & Mansmann, U. (2014b). CIPROS–a checklist with items for a patient registry software system. *Stud Health Technol Inform*, 205, 161–165.

Lindoerfer, D. & Mansmann, U. (2015). A Comprehensive Assessment Tool for Patient Registry Software Systems: The CIPROS Checklist. *Methods Inf Med*, 54(5), 447–454.

Lindoerfer, D. & Mansmann, U. (2016). Proposing an Evidence-Based Strategy for Software Requirements Engineering. *Stud Health Technol Inform*, 228, 648–652.

Lindoerfer, D. & Mansmann, U. (2017a). Data for the elaboration of the CIPROS checklist with items for a patient registry software system: Examples and explanations. *Data Brief*, 14, 494–497.

Lindoerfer, D. & Mansmann, U. (2017b). Enhancing Requirements Engineering for Patient Registry Software Systems with Evidence-based Components. *J Biomed Inform*, 71, 147–153.

Lindoerfer, D. & Mansmann, U. (2017c). IT Infrastructure of an Oncological Trial Where Xenografts Inform Individual Second Line Treatment Decision. *Stud Health Technol Inform*, 235, 226–230.

Lindörfer, D., Hoffmann, V., & Müller, T. H. (2011). Europaweite Integration und Analyse von Patienten mit chronischer myeloischer Leukämie (CML). `https://www.egms.de/static/en/meetings/gmds2011/11gmds529.shtm`. [Online; Accessed 22-Apr-2020].

Lindörfer, D. & Müller, T. H. (2010). Integration und Analyse von Patienten mit Chronischer Myeloischer Leukämie (CML) in 25 Europäischen Ländern mit webbasierter elektronischer Datenerfassung. In Schmücker P, Elsässer K-H, Hayna S (Ed.), *Tagungsband der 55. GMDS Jahrestagung* (pp. 161–165).

Mathis-Edenhofer, S. & Piso, B. (2011). [Types of medical registries - definitions, methodological aspects and quality of the scientific work with registries]. *Wien Med Wochenschr*, 161(23-24), 580–590.

Mayring, Philipp (2000). Qualitative Inhaltsanalyse. *Forum Qualitative Sozialforschung / Forum: Qualitative Social Research*, 1(2). [http://www.qualitative-research.net/index.php/fqs/article/view/1089/2383].

McGettigan, P., Alonso Olmo, C., Plueschke, K., Castillon, M., Nogueras Zondag, D., Bahri, P., Kurz, X., & Mol, P. G. M. (2019). Drug SafPatient Registries: An Underused Resource for Medicines Evaluation : Operational proposals for increasing the use of patient registries in regulatory assessments. *Drug Saf*, 42(11), 1343–1351.

McKinnon, AD (2005). Software Requirements Specification Template. `http://users.tricity.wsu.edu/~mckinnon/cpts322/cpts322-srs-v1.doc`. [Online; Accessed 23-Jun-2021].

Moher, D., Liberati, A., Tetzlaff, J., Altman, D. G., & Group, T. P. (2009). Preferred reporting items for systematic reviews and meta-analyses: the PRISMA statement. *PLoS Med.*, 6(7), e1000097.

Mojarrab, S., Rafei, A., Akhondzadeh, S., Jeddian, A., Jafarpour, M., & Zendehdel, K. (2017). Diseases and Health Outcomes Registry Systems in I.R. Iran: Successful Initiative to Improve Public Health Programs, Quality of Care, and Biomedical Research. *Arch Iran Med.*, 20, 696–703.

Müller, D., Augustin, M., Banik, N., Baumann, W., Bestehorn, K., Kieschke, J., Lefering, R., Maier, B., Mathis, S., Rustenbach, S. J., Sauerland, S., Semler, S. C., Stausberg, J., Sturm, H., Unger, C., & Neugebauer, E. A. (2010). [Memorandum registry for health services research]. *Gesundheitswesen*, 72(11), 824–839.

Müller, T. H. & Adelhard, K. (2002). A web-based central diagnostic data repository. *Stud Health Technol Inform*, 90, 246–250.

Napier, K. R., Pang, J., Lamont, L., Walker, C. E., Dawkins, H. J., Hunter, A. A., Bockxmeer, F. M. V., Watts, G. F., & Bellgard, M. I. (2017a). A Web-Based Registry for Familial Hypercholesterolaemia. *Heart Lung Circ*, 26(6), 635–639.

Napier, K. R., Tones, M., Simons, C., Heussler, H., Hunter, A. A., Cross, M., & Bellgard, M. I. (2017b). A web-based, patient driven registry for Angelman syndrome: the global Angelman syndrome registry. *Orphanet J Rare Dis*, 12(1), 134.

NCI (2015). National Cancer Institute. National Cancer Informatics Program. `http://cbiit.nci.nih.gov/ncip`. [Online; Accessed 11-May-2017].

NCI (2021). National Cancer Institute. Center for Biomedical Informatics & Information Technology. `https://datascience.cancer.gov/`. [Online; Accessed 06-Jul-2021].

NIH, List of Registries (2021). National Institutes of Health. List of Registries. `https://www.nih.gov/health-information/nih-clinical-research-trials-you/list-registries`. [Online; Accessed 29-Jun-2021].

OpenClinica (2021). OpenClinica. `https://www.openclinica.com/`. [Online; Accessed 29-Jun-2021].

Ordoñez Calero, H.D. (2018). EVALUACIÓN DE LA CALIDAD EN USO DE PRODUCTOS DE SOFTWARE PARA GESTIÓN DE HISTORIAS CLÍNICAS. https://bibdigital.epn.edu.ec/bitstream/15000/19731/1/CD-9132.pdf. [Online; Accessed 23-Jun-2020].

Osborne, J. D., Khare, A., Dempsey, D. M., Wells, J. M., Wyatt, M., Gordon, G., Liang, W. H., & Cimino, J. (2018). Phenotype Detection Registry System (PheDRS) - Implementation of a Generalizable Single Institution Clinical Registry Architecture. *AMIA Annu Symp Proc*, 2018, 847–856.

PubMed (2021a). PubMed, National Library of Medicine. https://www.ncbi.nlm.nih.gov/pubmed/. [Online; Accessed 23-Jun-2021].

PubMed (2021b). PubMed, National Library of Medicine - Lindoerfer D. https://pubmed.ncbi.nlm.nih.gov/?term=Lindoerfer+D&sort=date. [Online; Accessed 09-Aug-2021].

REDCap (2021). REDCap. https://www.project-redcap.org/. [Online; Accessed 29-Jun-2021].

Robertson, J. and Robertson S. (2015). Volere Requirements Specification Template. http://www.volere.co.uk/template.htm. [Online; Accessed 10-Feb-2016].

Robertson, S. and Robertson J. (2021). Volere Requirements Resources. http://www.volere.org. [Online; Accessed 23-Jun-2021].

Rowen, R. B. (1990). Software project management under incomplete and ambiguous specifications. *IEEE Transactions on Engineering Management*, 37(1), 10–21.

Royce, W. W. (1970). Managing the development of large software systems. *Proceedings, IEEE WESTCON*, (pp. 1–9).

scnat wissen (2018). Personalisierte Gesundheit erklärt - Omics. https://naturwissenschaften.ch/personalized-health-explained/welche_methoden_werden_eingesetzt_/omcis. [Online; Accessed 22-Jul-2021].

secuTrial (2021). secuTrial. https://www.secutrial.com/. [Online; Accessed 29-Jun-2021].

Shanbehzadeh, M., Kazemi-Arpanahi, H., Arzani-Birgani, A., Karimyan, A., & Mobasheri, F. (2020). Improving hypertension surveillance from a data management prospective: Data requirements for implementation of population-based registry. *J Educ Health Promot*, 9, 134.

Sommerville, I. (2016). *Software Engineering*. Harlow, Essex, UK: Pearson Education Limited, tenth edition.

Taruscio, D., Agresta, L., Amato, A., Bernardo, G., Bernardo, L., Braguti, F., Carbone, P., Carta, C., Ceccarini, M., Censi, F., Coppola, S., Crialese, P., De Santis, M., Diemoz, S., Donati, C., Gainotti, S., Ferrari, G., Floridia, G., Frank, C., Frazzica, R. G., Gentile, A. E., Granata, O., Kodra, Y., Latrofa, M., Laricchiuta, P., Magrelli, A., Morciano, C., Polizzi, A., Razeto, S., Salvatore, M., Sanseverino, A., Savini, D., Torreri, P., Tosto, F., Villani, F., Vincenti, G., & Vittozzi, L. (2014). The Italian National Centre for Rare Diseases: where research and public health translate into action. *Blood Transfus*, 12 Suppl 3, 591–605.

TMF e.V. (2021). TMF e.V. `http://www.tmf-ev.de`. [Online; Accessed 29-Jun-2021].

TraumaRegister DGU (2021). TraumaRegister DGU. `http://www.traumaregister-dgu.de`. [Online; Accessed 23-Jun-2021].

TRM (2020). Tumorregister München (TRM) - Munich Cancer Registry (MCR). `http://www.tumorregister-muenchen.de`. [Online; Accessed 29-Jun-2021].

V-Modell XT (2021). Das V-Modell XT. `http://www.cio.bund.de/Web/DE/Architekturen-und-Standards/V-Modell-XT/vmodell_xt_node.html`. [Online; Accessed 23-Jun-2021].

Vanderbilt University (2021). Vanderbilt University. `https://www.vanderbilt.edu/`. [Online; Accessed 29-Jun-2021].

Vangala, D., Ladigan, S., Liffers, S. T., Noseir, S., Maghnouj, A., Götze, T. M., Verdoodt, B., Klein-Scory, S., Godfrey, L., Zowada, M. K., Huerta, M., Edelstein, D. L., de Villarreal, J. M., Marqués, M., Kumbrink, J., Jung, A., Schiergens, T., Werner, J., Heinemann, V., Stintzing, S., Lindoerfer, D., Mansmann, U., Pohl, M., Teschendorf, C., Bernhardt, C., Wolters, H., Stern, J., Usta, S., Viebahn, R., Admard, J., Casadei, N., Fröhling, S., Ball, C. R., Siveke, J. T., Glimm, H., Tannapfel, A., Schmiegel, W., & Hahn, S. A. (2021). Secondary resistance to anti-EGFR therapy by transcriptional reprogramming in patient-derived colorectal cancer models. *Genome Med*, 13(1), 116.

VModellXTBund (2020). Das V-Modell XT Bund. `https://www.cio.bund.de/Web/DE/Architekturen-und-Standards/V-Modell-XT-Bund/vmodellxt_bund_node.html`. [Online; Accessed 01-Jul-2021].

Weit e.V. (2021). WEIT e.V. - Verein zur Weiterentwicklung des V-Modell XT. `http://weit-verein.de/`. [Online; Accessed 23-Jun-2021].

Wiegers, K. & Beatty, J. (2013). *Software Requirements*. Redmont: Microsoft Press, third edition.

Wilkinson, M. D., Dumontier, M., Aalbersberg, I. J., Appleton, G., Axton, M., Baak, A., Blomberg, N., Boiten, J. W., da Silva Santos, L. B., Bourne, P. E., Bouwman, J., Brookes, A. J., Clark, T., Crosas, M., Dillo, I., Dumon, O., Edmunds, S., Evelo, C. T., Finkers, R., Gonzalez-Beltran, A., Gray, A. J., Groth, P., Goble, C., Grethe, J. S., Heringa, J., 't Hoen, P. A., Hooft, R., Kuhn, T., Kok, R., Kok, J., Lusher, S. J., Martone, M. E., Mons, A., Packer, A. L., Persson, B., Rocca-Serra, P., Roos, M., van Schaik, R., Sansone, S. A., Schultes, E., Sengstag, T., Slater, T., Strawn, G., Swertz, M. A., Thompson, M., van der Lei, J., van Mulligen, E., Velterop, J., Waagmeester, A., Wittenburg, P., Wolstencroft, K., Zhao, J., & Mons, B. (2016). The FAIR Guiding Principles for scientific data management and stewardship. *Sci Data*, 3, 160018.

XClinical (2021). XClinical. `https://xclinical.com/de/`. [Online; Accessed 29-Jun-2021].

Zakerabasali, S., Kadivar, M., Safdari, R., Niakan Kalhori, S. R., Mokhtaran, M., Karbasi, Z., & Sayarifard, A. (2020). Development and validation of the Neonatal Abstinence Syndrome Minimum Data Set (NAS-MDS): a systematic review, focus group discussion, and Delphi technique. *J Matern Fetal Neonatal Med*, (pp. 1–8).

Zaletel, M. & Kralj, M., Eds. (2015). *Methodological guidelines and recommendations for efficient and rational governance of patient registries*. National Institute of Public Health. [https://ec.europa.eu/health/sites/health/files/ehealth/docs/patient_registries_guidelines_en.pdf Online; Accessed 29-Jun-2021].

Danksagung

An dieser Stelle bedanke ich mich bei allen, die mich während der Zeit meiner Promotion unterstützt und ermutigt haben.

Zuerst und vor allem gilt mein besonderer Dank meinem Betreuer Herrn Professor Dr. Ulrich Mansmann. Zuallererst, dass er mir die Möglichkeit gab, an diesem Thema als Doktorandin zu arbeiten und zweitens für seine große Unterstützung über all die Jahre, für die Zeit, die er sich für Diskussionen und die Beantwortung meiner Fragen nahm und für seine Hilfe und Unterstützung bei den Publikationen und Konferenzen.

Zusätzlich möchte ich mich bei meinem ehemaligen Kollegen Dr. Thomas Müller, für seine Leistungen bei der Entwicklung und Programmierung des DBFORM Systems, sowie für seinen Beitrag bei der Evaluation dieses Systems mit der CIPROS Checkliste bedanken.

Weiterhin bedanke ich mich ganz herzlich bei Professor Dr. Michele Baccarani von der Universität Bologna, dem Leiter des *Scientific Head Quarter* der *EUTOS CML Registries*, für seine große Unterstützung, für Diskussionen und die Beantwortung meiner Fragen, CML und die EUTOS CML Registries betreffend. Es war eine große Freude mit ihm zu arbeiten.

Desweiteren bedanke ich mich bei meinen Freunden und meiner Familie für das Verständnis, die Ermutigung und Unterstützung die sie mir während der Zeit meiner Promotion entgegengebracht haben.

Ganz herzlich bedanke ich mich bei dem Verlag Schattauer / Thieme Gruppe für die Erlaubnis zum Abdruck meines Papers in dieser Arbeit:

Lindoerfer D, Mansmann U. A Comprehensive Assessment Tool for Patient Registry Software Systems: The CIPROS Checklist. Methods Inf Med. 2015;54(5):447-54. doi: 10.3414/ME14-02-0026. Epub 2015 Sep 30.

Ebenso bedanke ich mich ganz herzlich bei dem Verlag Elsevier für die Erlaubnis zum Abdruck meines Papers in dieser Arbeit:

Lindoerfer D, Mansmann U. Enhancing requirements engineering for patient registry software systems with evidence-based components. J Biomed Inform. 2017 Jul;71:147-153. doi: 10.1016/j.jbi.2017.05.013. Epub 2017 May 20.

Publikationsliste

Vangala, D., Ladigan, S., Liffers, S. T., Noseir, S., Maghnouj, A., Götze, T.M., Verdoodt, B., Klein-Scory, S., Godfrey, L., Zowada, M. K., Huerta, M., Edelstein, D. L., de Villarreal, J. M., Marqués, M., Kumbrink, J., Jung, A., Schiergens, T., Werner, J., Heinemann, V., Stintzing, S., **Lindoerfer, D.**, Mansmann, U., Pohl, M., Teschendorf, C., Bernhardt, C., Wolters, H., Stern, J., Usta, S., Viebahn, R., Admard, J., Casadei, N., Fröhling, S., Ball, C. R., Siveke, J. T., Glimm, H., Tannapfel, A., Schmiegel, W., Hahn, S.A. (2021). Secondary resistance to anti-EGFR therapy by transcriptional reprogramming in patient-derived colorectal cancer models. Genome Medicine, (2021)13:116.

Lindoerfer, D., Mansmann, U., Reinhardt, I. (2020). Incorporation of Multiple Sources into IT - and Data Protection Concepts: Lessons Learned from the FARKOR Project. Stud Health Technol Inform, 270:262-266.

Hoffmann, S., Crispin, A., **Lindoerfer, D.**, Sroczynski, G., Siebert, U., Mansmann, U., FARKOR Consortium. (2020). Evaluating the effects of a risk-adapted screening program for familial colorectal cancer in individuals between 25 and 50 years of age: study protocol for the prospective population-based intervention study FARKOR. BMC Gastroenterology, 20(1):131.

Lindoerfer, D. and Mansmann, U. (2017). Enhancing Requirements Engineering for Patient Registry Software Systems with Evidence-based Components. J Biomed Inform, 71:147-153.

Lindoerfer, D. and Mansmann, U. (2017). Data for the elaboration of the CIPROS checklist with items for a patient registry software system: Examples and explanations. Data Brief, 14:494-497.

Lindoerfer, D. and Mansmann, U. (2017). Evaluation of the IT Infrastructure of the RESIST Study with the Evidence-Based CIPROS Checklist. Stud Health Technol Inform, 243:47-51. 62. GMDS Jahrestagung, 17.-21. September 2017, Oldenburg. Paper und Vortrag

Lindoerfer, D. and Mansmann, U. (2017). IT Infrastructure of an Oncological Trial Where Xenografts Inform Individual Second Line Treatment Decision. Stud Health Technol Inform, 235:226-230. Informatics for Health 2017 24 - 26 April 2017 Manchester, UK. Paper und Vortrag

Lindoerfer, D. and Mansmann, U. (2016). Proposing an Evidence-Based Strategy for Software Requirements Engineering. Stud Health Technol Inform, 228:648-652. HEC 2016 - Health - Exploring Complexity: An Interdisciplinary Systems Approach. 28. August - 2. September 2016. Munich, Germany. Paper und Vortrag

Hoffmann, V. S., Baccarani, M., Hasford, J., Castagnetti, F., Di Raimondo, F., Casado, L. F., Turkina, A., Zackova, D., Ossenkoppele, G., Zaritskey, A., Hoglund, M., Simonsson, B., Indrak, K., Sninska, Z., Sacha, T., Clark, R., Bogdanovic, A., Hellmann, A., Griskevicius, L., Schubert-Fritschle, G., Sertic, D., Guilhot, J., Lejniece, S., Zupan, I., Burgstaller, S., Koskenvesa, P., Everaus, H., Costeas, P., **Lindoerfer, D.**, Rosti, G., Saussele, S., Hochhaus, A., and Hehlmann, R. (2017). Treatment and outcome of 2904 CML patients from the EUTOS population-based registry. Leukemia, 31(3):593-601.

Lindoerfer, D. and Mansmann, U. (2015). A Comprehensive Assessment Tool for Patient Registry Software Systems: The CIPROS Checklist. Methods Inf Med, 54(5):447-454.

Hoffmann, V. S., Baccarani, M., Hasford, J., **Lindoerfer, D.**, Burgstaller, S., Sertic, D., Costeas, P., Mayer, J., Indrak, K., Everaus, H., Koskenvesa, P., Guilhot, J., Schubert-Fritschle, G., Castagnetti, F., Di Raimondo, F., Lejniece, S., Griskevicius, L., Thielen, N., Sacha, T., Hellmann, A., Turkina, A. G., Zaritskey, A., Bogdanovic, A., Sninska, Z., Zupan, I., Steegmann, J. L., Simonsson, B., Clark, R. E., Covelli, A., Guidi, G., and Hehlmann, R. (2015). The EUTOS population-based registry: incidence and clinical characteristics of 2904 CML patients in 20 European Countries. Leukemia, 29(6):1336-1343.

Lindoerfer, D. and Mansmann, U. (2014). CIPROS - a checklist with items for a patient registry software system. Stud Health Technol Inform, 205:161-165. MIE 2014 - XXV Conference of the European Federation for Medical Informatics, 31 August - 3 September 2014, Istanbul, Turkey. Paper und Vortrag

Hoffmann, V. S., Baccarani, M., **Lindoerfer, D.**, Castagnetti, F., Turkina, A., Zaritsky, A., Hellmann, A., Prejzner, W., Steegmann, J. L., Mayer, J., Indrak, K., Colita, A., Rosti, G., and Pfirrmann, M. (2013). The EUTOS prognostic score: review and validation in 1288 patients with CML treated frontline with imatinib. Leukemia, 27(10):2016-2022.

Hoffmann, V., Baccarani, M., Hasford, J., Guilhot, J., Saussele, S., Rosti, G., Guilhot, F., Porkka, K., Ossenkoppele, G., **Lindoerfer, D.**, Simonsson, B., Pfirrmann, M., Hehlmann, R. (2012). The EUTOS CML score aims to support clinical decision-making. Blood. 2012 Mar 22; 119(12):2966-7.

Hasford, J., Baccarani, M., Hoffmann, V., Guilhot, J., Saussele, S., Rosti, G., Guilhot, F., Porkka, K., Ossenkoppele, G., **Lindoerfer, D.**, Simonsson, B., Pfirrmann, M., and Hehlmann, R. (2011). Predicting complete cytogenetic response and subsequent progression-free survival in 2060 patients with CML on imatinib treatment: the EUTOS score. Blood, 118(3):686-692.

München, 30.09.2021 Doris Lindörfer