

Universität Leipzig  
Softwaretechnik-Praktikum

# Recherchebericht

SLR17 - Tool zur Durchführung und Dokumentation strukturierter  
Literaturrecherchen

18.12.2016

## Inhaltsverzeichnis

<b>1.</b>	<b>Begriffe</b>	<b>3</b>
1.1.	Systematic Literature Review – SLR	3
1.2.	Datenbank	3
1.3.	Rückwärtssuche	3
1.4.	Vorwärtssuche	3
1.5.	Mashup	3
1.6.	Taxonomie	3
1.7.	Java	3
1.8.	GUI	3
1.9.	Google Scholar	4
1.10.	BibTeX	4
<b>2.</b>	<b>Konzepte</b>	<b>4</b>
2.1.	Graphische Benutzeroberfläche	4
2.2.	Modularer Aufbau der Anwendung	4
	2.2.1. Definitionsmodul	4
	2.2.2. Konzeptmodul	4
	2.2.3. Literatursuche	5
	2.2.4. Literaturanalyse und -synthese	5
2.3.	Agile Softwareentwicklung	5
2.4.	Verarbeitung externer Daten	5
<b>3.</b>	<b>Aspekte</b>	<b>5</b>
3.1.	Bestehende Tools	5
	3.1.1. ReviewER	5
	3.1.2. RevMan	5
	3.1.3. PrepLit und LitAssist	5
	3.1.4. Docear	5
3.2.	Ziele	6
	3.2.1. Benutzerfreundlichkeit	6
	3.2.2. Kompatibilität	6
	3.2.3. Modularität	6
	3.2.4. Integration	6
	3.2.5. Prozessunterstützung	7
<b>4.</b>	<b>Quellen</b>	<b>7</b>

## 1. Begriffe

### 1.1. Systematic Literature Review - SLR

Systematic Literature Review oder kurz SLR bezeichnet eine Art Literaturübersicht zu einem bestimmten Thema. Dabei wird mit geeigneten Methoden eine objektive und durchdringende Suche nach relevanten Artikeln zu diesem Thema durchgeführt. Das gesammelte Wissen wird dabei aggregiert und kritisch bewertet. Oft wird auch eine Alternative zu Fachbüchern geboten, da Reviews zum einen häufiger erscheinen als Fachbücher und zum anderen auch der Aufwand für die Erstellung einer solchen Übersichtsarbeit wesentlich geringer ist.

### 1.2. Datenbank

Eine Datenbank oder auch Datenbanksystem (DBS) ist ein System zur dauerhaften Speicherung und Verwaltung großer Datenmengen. Dabei besteht das DBS aus zwei Hauptkomponenten: Aus der Datenbank im engeren Sinn, also den gespeicherten Daten (DB) und aus einem Programm zur Verwaltung der Daten (Datenbankverwaltungsprogramm).

### 1.3. Rückwärtssuche

Bei der Rückwärtssuche werden die Quellen der gefundenen Texte analysiert. Wenn diese für die Literaturrecherche von Bedeutung sein könnten, sollten sie eingebunden werden. Daraufhin kann eine erneute Durchführung weitere Quellen aufdecken. Allerdings geht dadurch die Aktualität verloren, da stets ältere Quellen gefunden werden.

### 1.4. Vorwärtssuche

Hierbei werden nicht die Quellen der Texte sondern Werke, welche Auszüge der Texte zitieren oder vermerken, untersucht. Diese Assoziationen müssen in den zugrundeliegenden Datenbanken protokolliert sein. Auch hier können, wie bei der Rückwärtssuche, die gefundenen Materialien eingebunden und erneut untersucht werden. Ein bekannter Vertreter der Vorwärtssuche ist Google Scholar.

### 1.5. Mashup

Als Mashup werden Webseiten beziehungsweise Webanwendungen, welche mindestens zwei bereits existierende Programmstrukturen neu kombinieren, sodass eigenständige Inhalte entstehen, beschrieben. Über offene Schnittstellen können Medien wie Fotos oder Videos, aber auch Ressourcen und Daten eingebunden sowie verwendet werden.

### 1.6. Taxonomie

Eine Taxonomie ist ein einheitliches Verfahren oder Modell (Klassifikationsschema), mit dem Objekte nach bestimmten Kriterien klassifiziert werden. Naturwissenschaftliche Disziplinen verwenden den Begriff der Taxonomie für eine in der Regel hierarchische Klassifikation (Klassen, Unterklassen usw.).

### 1.7. Java

Java ist eine objektorientierte und plattformunabhängige Programmiersprache, welche um 1995 von Sun Microsystems entwickelt worden ist.

### 1.8. GUI

Als Graphical User Interface oder kurz GUI bezeichnet man diejenige grafische Benutzeroberfläche, die das Bedienen von Computern und Anwendungen auf Computern

erleichtert. Mit Vorrang ist eine GUI übersichtlich und intuitiv verständlich aufgebaut, um die erleichterte Bedienbarkeit zu gewährleisten.

### 1.9. Google Scholar

Google Scholar wurde 2004 von Google Inc. zur Literaturrecherche wissenschaftlicher Werke erstellt. Eingebunden sind neben Universitätsdokumenten und akademischen Verbänden auch Gerichtsurteile, wodurch eine riesige Datenbank geschaffen wurde.

### 1.10. BibTeX

Bibtex ist ein Literaturverwaltungsprogramm zur Entwicklung von Bibliografien. Die plattformunabhängige Anwendung wurde 1985 in der Programmiersprache Web verfasst und hat im Bereich der digitalen Erfassung und Verarbeitung von Literaturverzeichnissen erhebliche Bedeutung. Da selbst große Datenmengen mühelos untersucht werden können, findet Bibtex sogar in den Datenbanken namhafter Institutionen wie der NASA Verwendung.

## 2. Konzepte

### 2.1. Graphische Benutzeroberfläche

Aufgrund der im Studium bereits erlangten Kenntnisse aller Teilnehmer, haben wir uns dazu entschieden unser Projekt in Java umzusetzen. Java ist aufgrund seiner Plattformunabhängigkeit gut geeignet um möglichst vielen Nutzern unsere Software bereitzustellen. Des Weiteren beinhaltet Java AWT und Java Swing als Bestandteile der Java Foundation Class. Mithilfe dieser Bestandteile lässt sich eine graphische Oberfläche erzeugen, die ein essentieller Bestandteil unserer Software darstellt. Wir beabsichtigen unsere GUI simpel und intuitiv zu gestalten, um den Prozess der strukturierten Literaturrecherche bestmöglich zu unterstützen und zu vereinfachen.

### 2.2. Modularer Aufbau der Anwendung

Um das schrittweise Vorgehen bei der strukturierten Literaturrecherche widerzuspiegeln, haben wir beschlossen, den Prozess, den die Benutzer in unserer Anwendung durchlaufen, auf vier zentrale Module herunter zu brechen. Diese Module werden nacheinander in festgelegter Reihenfolge bearbeitet, wobei das nächste Modul erst begonnen werden kann, wenn das vorherige abgeschlossen ist. Im Folgenden werden wir die vier Module näher erläutern:

#### 2.2.1. Definitionsmodul

In diesem Modul werden die Ziele der Literaturrecherche vom Benutzer definiert, wodurch der Rahmen für die restlichen Schritte festgelegt wird. Dies geschieht beispielsweise anhand der von Cooper vorgeschlagenen Taxonomie von Literaturreviews, die eine nähere Klassifikation des Vorgangs zulassen.

#### 2.2.2. Konzeptmodul

Hilfe beim Herausfiltern von Schlüsselwörtern, die für das angestrebte Thema der Literaturrecherche relevant sind. Außerdem werden Methoden bereitgestellt um zentrale Konzepte bzw. Terme zu organisieren. Der Benutzer legt fest, welche Quellen für die Suche geeignet sind.

#### 2.2.3. Literatursuche

Es werden Suchbegriffe und Randbedingungen festgelegt, mit denen eine Suche in den angegebenen Quellen durchgeführt wird. Dem Benutzer wird die Möglichkeit gegeben, die gefundene Literatur nach Relevanz zu bewerten und gegebenenfalls eine Vorwärts-/Rückwärtssuche durchzuführen.

#### 2.2.4. Literaturanalyse und -synthese

In diesem Schritt wird die ausgewählte Literatur vom Nutzer nach eigenen Maßstäben analysiert und die Anwendung bietet ihm die Möglichkeit bereits definierte Konzepte zu vereinigen, neue Konzepte zu bilden und die Literatur mit Anmerkungen zu versehen. Weiterhin können die Quellen der Literaturrecherche näher untersucht werden z.B. könnten Referenzierungen und Zitierungen unter den Dokumenten als Graph dargestellt werden

Weiterhin soll der zeitliche Verlauf der Recherche sowie der Ergebnisse dokumentiert werden können und der Zugriff auf andere Anwendungen und Datenformate, sowie den Export in verschiedene Formate möglich sein. Weitere mögliche Funktionalitäten wären Module zur Unterstützung von kollaborativer Arbeit, der Bewertung und Wiederholung von Reviews und der Auswertung inhaltlicher Aspekte der Dokumente, z.B. statistischer Methoden.

#### 2.3. Agile Softwareentwicklung

Ziel der agilen Softwareentwicklung ist es einen flexiblen Softwareprozess zu erzeugen, indem das Ergebnis des Fortschrittes höher geschätzt wird, als die dazu gehörige Dokumentation. Die wichtigste Eigenschaft der Agilen Softwareentwicklung ist, dass Benutzer/Kunden an Entwicklung/Evaluation aller Versionen beteiligt sind, damit wird erreicht, dass auf kurzfristige Änderungen Rücksicht genommen werden kann. Somit liegt das fertige Produkt möglichst nah an den Vorstellungen beider Parteien.

#### 2.4. Verarbeitung externer Daten

Während des Prozesses der Literaturrecherche möchten wir dem User ermöglichen seine Suche auf externe Datenbanken zu erweitern. Der Nutzer kann unter mehreren Möglichkeiten frei wählen. Gefundene Dokumente können heruntergeladen werden und innerhalb der Anwendung geöffnet und weiter bearbeitet werden. Außerdem möchten wir unseren Nutzern ermöglichen bereits angefertigte Dateien zu importieren, sowie bei einem abgeschlossenen Review die gesammelten Daten in ein geeignetes Format zu exportieren.

### 3. Aspekte

#### 3.1. Bestehende Tools

Im Folgenden wird eine Auswahl an Programmen betrachtet, die bereits zur Erstellung von Literatur-Reviews genutzt werden.

3.1.1. ReviewER ist ein sehr grundsätzliches Tool zur Selektion von Quellen, allerdings zur Verwaltung von Literatur im größeren Stil nicht geeignet.

3.1.2. RevMan bietet einen großen Funktionsumfang, ist jedoch auf Literaturrecherchen in der Medizin (im Speziellen Cochrane-Reviews) ausgelegt. Zudem erfordert seine GUI einige Eingewöhnung.

3.1.3. PrepLit und LitAssist sind kommerzielle Tools, die zwar eine akzeptable Funktionalität bieten, jedoch wenig freie Anbindungsmöglichkeiten an andere Programme ermöglichen.

3.1.4. Docear ist die wohl ausgereifteste Alternative, mit einer großen Auswahl an Export und Import zu anderen Programmen, einem offenen Entwicklungsmodell und einer ansprechenden GUI. Spezifische Funktionalität zum Verwalten von Literaturrecherchen (wie zum Beispiel Extrahieren und Aufbereiten von Konzepten) fehlt jedoch, da Docear eher auf die Verwaltung von wissenschaftliche Arbeiten im Allgemeinen konzipiert ist.

Eine verbreitete Methode zur Organisation von Quellen ist eine Tabelle in Microsoft Excel oder ähnlichen Tabellenkalkulationsprogrammen. Dies erfordert jedoch viel manuelle Wartung vom Benutzer und wird zunehmend unübersichtlich mit größeren Datenmengen.

### 3.2. Ziele

Literatur-Reviews sind aus dem wissenschaftlichen Arbeiten in allen Fachgebieten nicht mehr wegzudenken. Die existierenden technischen Hilfsmittel sind jedoch für viele Autoren unzureichend. Das Ziel dieses Projektes ist daher, ein Programm zum Sammeln und Organisieren von Literatur für Literatur-Reviews zu erstellen. Folgende Aspekte sollen dabei als Schwerpunkte dienen:

#### 3.2.1. Benutzerfreundlichkeit

Ein entscheidender Aspekt für unsere Software ist die Benutzerfreundlichkeit. Um diese zu gewährleisten ist zum einen eine klare, intuitiv verständliche GUI von entscheidender Wichtigkeit, zum anderen eine gute Prozessunterstützung (siehe Prozessunterstützung). Auch soll das Programm eine Möglichkeit zum Speichern und Exportieren von den Einstellungen bereithalten, damit ein Team/Unternehmen einheitlich und nachvollziehbar arbeiten kann. Dies unterstützt auch den Mehrbenutzerbetrieb, welcher in Teamarbeit von hoher Wichtigkeit ist. Um diesen zu verbessern möchten wir Optionen zur Synchronisation oder mindestens zum Vergleich der bewerteten Literatur und den extrahierten Konzepten zwischen verschiedenen Benutzern einbringen.

#### 3.2.2. Kompatibilität

Um die Arbeit in Gruppen zu verbessern soll unser Programm plattformunabhängig einsetzbar sein, um ein Arbeiten in vertrauter Umgebung zu ermöglichen, sowie für eine möglichst breite Masse zur Verfügung zu stehen. Die Plattformunabhängigkeit wird durch das Programmieren in Java ermöglicht. Java ermöglicht uns Unabhängigkeit vom Betriebssystem, als ersten Schritt. Der zweite Schritt wäre das entwickeln von Applications für Smartphones und E-Reader, damit direkt während des Lesens z.B. Konzepte extrahiert werden können ohne einen PC gleichzeitig benutzen zu müssen. Das Programm wird auch unabhängig vom Gebiet der Literaturrecherche einsetzbar sein, somit darf kein technisches Wissen zur Einrichtung der Software nötig sein. Für weitere Kompatibilität muss eine Integration in bestehende Strukturen gewährleistet sein(siehe Integration).

#### 3.2.3. Modularität

Literatur-Recherche ist ein komplexer und vielseitiger Prozess. Dieser lässt sich in mehrere Phasen einteilen, welche alle unterschiedliche Anforderungen an Autor und Software stellen. Zudem muss berücksichtigt werden, dass sich Literatur-Reviews und die damit verbundenen Recherchen je nach Intention, Herangehensweise und auch der Fachrichtung des Autors unterscheiden können. Die Modularität der Software soll diese Vielfältigkeit abbilden und eine leichte Anpassungsfähigkeit und Erweiterbarkeit an verschiedenste Bedürfnisse gewährleisten (siehe Oben).

#### 3.2.4. Integration

Das Tool sollte verschiedene Datenformate wie Excel und LaTeX importieren und exportieren können und Schnittstellen zu anderer Software – besonders zu Literatur-Datenbanken – bieten, um eine Integration in bestehende Arbeitsabläufe, Softwaresysteme und Anwendungsgebiete zu ermöglichen.

### 3.2.5. Prozessunterstützung

Literatur-Reviews haben verhältnismäßig strikte Vorgaben zur Gliederung, und der Ablauf der Erstellung ist linear. Diesen Ablauf sollte das Programm mitverfolgen und neuen Benutzern eine Stütze zur Gliederung ihres Reviews bilden. Erfahrenen Autoren sollte diese Hilfestellung jedoch nicht zur Behinderung werden. Der Prozess sollte vom Programm genau dokumentiert werden, damit die Erstellung des Reviews nachvollziehbar bleibt - auch für Andere.

## 4. Quellen

Internetquellen abgerufen am 18.12.2016

[https://de.wikipedia.org/wiki/Systematische\\_%C3%9Cbersichtsarbeit](https://de.wikipedia.org/wiki/Systematische_%C3%9Cbersichtsarbeit)

Vorlesung Datenbanksysteme 1, Rahm, 2016

<http://www.marketing-lehre.org/content.php?p=29>

<http://www.marketing-lehre.org/content.php?p=30>

[https://de.wikipedia.org/wiki/Mashup\\_\(Internet\)](https://de.wikipedia.org/wiki/Mashup_(Internet))

[https://en.wikipedia.org/wiki/Mashup\\_\(web\\_application\\_hybrid\)](https://en.wikipedia.org/wiki/Mashup_(web_application_hybrid))

<https://de.wikipedia.org/wiki/Taxonomie>

<http://www.bibtex.org/de/>

<https://de.wikipedia.org/wiki/BibTeX>

<https://en.wikipedia.org/wiki/BibTeX>

<http://www.itwissen.info/definition/lexikon/Java.html>

<http://www.itwissen.info/definition/lexikon/graphical-user-interface-GUI-Grafische-Benutzeroberflaeche.html>

[https://de.wikipedia.org/wiki/Google\\_Scholar](https://de.wikipedia.org/wiki/Google_Scholar)

<https://scholar.google.de/intl/de/scholar/about.html>

<https://sites.google.com/site/eseportal/tools/reviewer>

<http://agilemanifesto.org/>

[https://de.wikipedia.org/wiki/Abstract\\_Window\\_Toolkit](https://de.wikipedia.org/wiki/Abstract_Window_Toolkit)

[https://de.wikipedia.org/wiki/Swing\\_\(Java\)](https://de.wikipedia.org/wiki/Swing_(Java))

<http://www.caterdev.de/2012/05/30/hinweise-zur-anfertigung-eines-literatur-reviews/>

Harris Cooper. Organizing knowledge syntheses: A taxonomy of literature reviews. Knowledge, Technology & Policy, 1:104-126, 1988