



# VERIFIABLE PAPER CERTIFICATES

Software-Entwicklungspraktikum (SEP)  
Sommersemester 2015

## Angebot

Auftraggeber  
Technische Universität Braunschweig  
Institut für Betriebssysteme und Rechnerverbund  
Prof. Dr.-Ing. Lars Wolf  
Mühlenpfordtstraße 23  
38106 Braunschweig

Betreuer: Björn Gernert, Dominik Schürmann

Auftragnehmer:

Name	E-Mail-Adresse
Eileen Seeliger	e.seeliger@tu-braunschweig.de
Fabian Kötz	f.koetz@tu-braunschweig.de
Fabian Ochmann	f.ochmann@tu-braunschweig.de
Leif Scheppelmann	l.scheppelmann@tu-braunschweig.de
Marisa Krummrich	m.krummrich@tu-braunschweig.de
Macrel Wiertelorz	m.wiertelorz@tu-braunschweig.de
Stephan Merker	s.merker@tu-braunschweig.de
Tobias Volze	t.volze@tu-braunschweig.de

Braunschweig, 24. April 2015

## Inhaltsverzeichnis

<b>1</b>	<b>Einleitung</b>	<b>4</b>
1.1	Ziel . . . . .	4
1.2	Motivation . . . . .	4
<b>2</b>	<b>Formale Grundlagen</b>	<b>5</b>
<b>3</b>	<b>Projektablauf</b>	<b>6</b>
3.1	Meilensteine . . . . .	6
3.2	Geplanter Ablauf . . . . .	7
<b>4</b>	<b>Projektumfang</b>	<b>8</b>
4.1	Lieferumfang . . . . .	8
4.2	Kostenplan . . . . .	8
4.3	Funktionaler Umfang . . . . .	8
<b>5</b>	<b>Entwicklungsrichtlinien</b>	<b>10</b>
5.1	Konfigurationsmanagement . . . . .	10
5.2	Design- und Programmierrichtlinien . . . . .	10
5.3	Verwendete Software . . . . .	10
<b>6</b>	<b>Projektorganisation</b>	<b>12</b>
6.1	Schnittstelle zum Auftraggeber . . . . .	12
6.2	Schnittstelle zu anderen Projekten . . . . .	12
6.3	Interne Kommunikation . . . . .	12
<b>7</b>	<b>Glossar</b>	<b>13</b>

## Abbildungsverzeichnis

3.1	Zeitplan des Projekts als Gantt-Diagramm . . . . .	7
4.1	Use-Case des funktionalen Umfangs . . . . .	9

# 1 Einleitung

## 1.1 Ziel

Anwender sollen durch die Nutzung einer Android-**App**<sup>1</sup> schnell, einfach und sicher Zeugnisse auf ihre Echtheit überprüfen können.

Konkret soll der Anwender mit einem Smartphone ein Zeugnis einscannen können und eine Meldung über dessen Authentizität bekommen. Eine **Server**applikation übernimmt im Hintergrund die Prüfung der Daten und die Verwaltung der Zeugnisse. Ebenso ist es über die Serverapplikation möglich, anhand einer Vorlage neue Zeugnisse zu erstellen.

## 1.2 Motivation

Bisher ist es für Unternehmen sehr aufwändig, von Bewerbern eingereichte Dokumente zu überprüfen. Kopien der Dokumente müssen an die zuständigen Abteilungen des Zeugnis-Ausstellers gesendet werden, was mehrere Wochen in Anspruch nehmen kann. Daher existiert ein Lösungsbedarf zur schnellen Authentizitätsprüfung von Zeugnissen. Die Aufgabenstellung im hier genannten Projekt soll die Überprüfung vereinfachen.

Die Erstellung erfolgt im Rahmen des Softwareentwicklungspraktikums 2015 an der Technischen Universität Braunschweig.

---

<sup>1</sup>Alle fettgedruckten Wörter sind im Glossar erklärt

## 2 Formale Grundlagen

Als Programmiersprache für das Projekt wird *Java* für die Android-App und *PHP*, sowie auch *Java*, für den Server verwendet.

Die App wird für die Android Version 4.0.3 (Ice Cream Sandwich) programmiert und erreicht damit 90,4% der im Google Play Store registrierten Geräte. Um die Funktionalität der PDF-Erstellung zu gewährleisten, wird die **Open-Source** Software *iTextpdf* verwendet.

Das Einlesen der Zeugnisse wird mittels der *tess-two* Bibliothek realisiert, während die *PDFlib* Bibliothek zum generieren verwendet wird.

Die Benutzeroberfläche der App wird in deutscher Sprache ausgeliefert, dennoch ist die Übersetzung in andere Sprachen möglich. Aufgrund der Unsicherheit von *SHA1* wird die Echtheitsüberprüfung mittels der **SHA256-Hash Funktion** durchgeführt.

Generell ist eine Erweiterung auf andere mobile Betriebssysteme möglich. Da PHP und Java für den Server verwendet werden, läuft dieser unabhängig von dem verwendeten Betriebssystem.

## 3 Projektablauf

### 3.1 Meilensteine

Nummer	Meilenstein	Dokumente	Abgabetermin
1	Projektstart	-	13.04.15
2	Angebot	Angebot	24.04.15
3	Server eingerichtet	-	29.04.15
4	App scannt Dokumente	-	08.05.15
5	Server stellt Zeugnisse aus	-	08.05.15
6	Pflichtenheft	Pflichtenheft	13.05.15
7	Zwischenpräsentation	<b>Prototyp</b>	15.05.15
8	Kommunikation zwischen App und Server	-	22.05.15
9	App generiert Hash	-	26.05.15
10	Hash Überprüfung mit Rückmeldung	-	31.05.15
11	Fachentwurf	Fachentwurf	03.06.15
12	Abschluss der Android-App	Quellcode	10.06.15
13	Abschluss der Serverapplikation	Quellcode	16.06.15
14	Technischer Entwurf	Technischer Entwurf	01.07.15
15	Testdokumentation	Testdokumentation	15.07.15
16	Tag der jungen Software Entwickler	Fertige Applikationen	23.07.15

3.2 Geplanter Ablauf

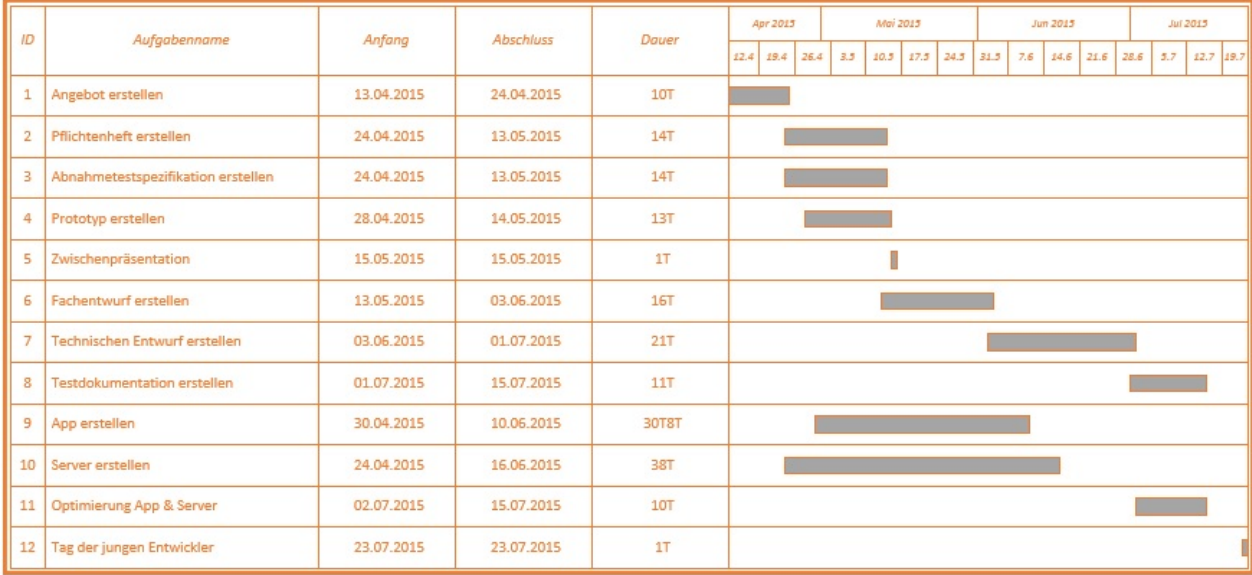


Abbildung 3.1: Zeitplan des Projekts als Gantt-Diagramm

## 4 Projektumfang

### 4.1 Lieferumfang

Im Lieferumfang enthalten sind funktionierende und lauffähige Quellcodes der Serverapplikation sowie die lauffähige Android App, eine Designvorlage zum Erstellen von Zeugnissen und ein beispielhaft erstelltes Zeugnis. Außerdem sind jegliche geforderte Dokumente wie Fachentwurf, technischer Entwurf, Pflichtenheft, etc. im Lieferumfang enthalten.

### 4.2 Kostenplan

Der Arbeitsaufwand für jeden Entwickler wird aufgrund der vorgesehenen 8 Credit Points auf 240 Stunden geschätzt. An dem Projekt sind acht Entwickler beteiligt, daraus ergibt sich ein Gesamt-Zeitaufwand von 1.920 Stunden. Somit werden die Kosten bei einem Stundensatz von 100€ voraussichtlich etwa 192.000€ betragen.

### 4.3 Funktionaler Umfang

Der **User** kann mithilfe der App ein Zeugnis scannen. Anschließend läuft das OCR über das eingescannte Zeugnis, bildet ein Hash aus den persönlichen Daten und sendet damit eine Anfrage an den Server. Sobald dieser die Anfrage empfangen hat, wird anhand der **Secure ID** (Darstellung im Klartext oder per **QR-Code** möglich) das Zeugnis überprüft. Sollte das Zeugnis in der Datenbank nicht existieren, wird eine negative Rückmeldung an die App gesendet. Ist die Secure ID bekannt, wird der dazugehörige Hash überprüft. Stimmen diese überein, wird eine positive Rückmeldung an die App gesendet, anderenfalls eine negative Rückmeldung.

Empfängt die App eine negative Rückmeldung, wird dem Benutzer auf dem Display eine Meldung gezeigt, dass das gescannte Zeugnis gefälscht ist.

Ist die empfangene Rückmeldung positiv, wird durch eine Meldung auf dem Display die Echtheit des Zeugnisses bestätigt.

Ein **Administrator** kann zudem über die Server-Applikation Daten wie persönliche Daten und Noten in eine Zeugnisvorlage eingeben sowie die Datenbank mit allen ausgestellten Zeugnissen verwalten.



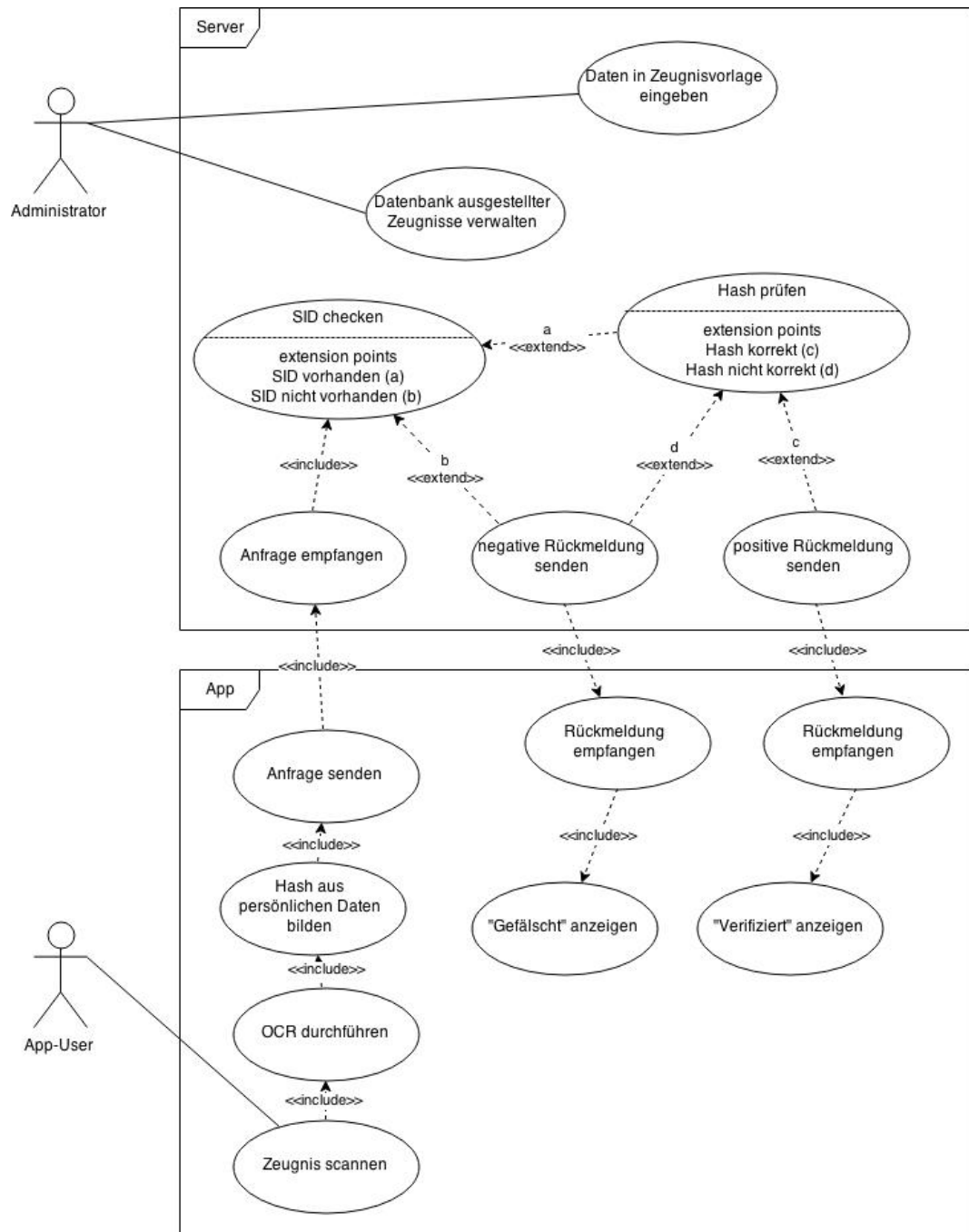


Abbildung 4.1: Use-Case des funktionalen Umfangs

## 5 Entwicklungsrichtlinien

### 5.1 Konfigurationsmanagement

Es existiert ein zentrales **Repository** mit festgelegten **Commit**-Regeln und Folgender Ordnerstruktur:

- SVN Gruppe 2
  - Server
  - App
  - Vorlagen
  - Dokumente

Die Commit-Regeln besagen, dass nur sinnvolle bzw. aussagekräftige Commit-Nachrichten geschrieben werden dürfen. Außerdem soll jede Commit-Nachricht den Vornamen der Person enthalten die den Commit erstellt. Sollte es mehrere Personen mit dem selben Vornamen in der Gruppe geben wird zur Identifikation zusätzlich der erste Buchstabe des Nachnamens angefügt. Des Weiteren dürfen nur ausführbare Dateien hochgeladen werden.

Folgendes Format soll für die Commit-Nachrichten verwendet werden: Vorname - Nachricht.

### 5.2 Design- und Programmierrichtlinien

Zur Orientierung bezüglich der Design- und Programmierrichtlinien werden die Java Code Conventions und der PHP Coding Standard genutzt.

### 5.3 Verwendete Software

Für die Erstellung der Dokumentation wird MikTex in Kombination mit dem TexMaker verwendet. Die benötigten Diagramme werden mit Hilfe des Redmine, [www.draw.io](http://www.draw.io) und Microsoft Visio erstellt.

Zur zentralen Lagerung und Versionsverwaltung der Dateien wird ein SVN Repository verwendet. Die Wahl des jeweiligen SVN-Clients ist dabei den einzelnen Gruppenmitgliedern selbst

überlassen.

Zur Texterkennung auf Android wird die tesseract-Bibliothek tess-two verwendet. Als Entwicklungsumgebung werden Eclipse, JavaEditor und Android Studio verwendet. Auf dem Server wird Apache und als Datenbank MySQL verwendet.

## 6 Projektorganisation

### 6.1 Schnittstelle zum Auftraggeber

Die Schnittstelle zum Auftraggeber besteht aus den wöchentlichen Übungsterminen sowie den verpflichtenden SEP-Terminen. Außerdem ist die Gruppe per E-Mail über den Phasenverantwortlichen erreichbar.

Zusätzlich dazu kann die Gruppe über die IBR-SEP Mailingliste erreicht werden.

### 6.2 Schnittstelle zu anderen Projekten

Das Projekt ist in sich geschlossen und läuft insofern eigenständig. Es gibt keine Abhängigkeiten zu anderen Projekten. Demnach gehören keine Schnittstellen zu diesem Projekt.

### 6.3 Interne Kommunikation

Pro Phase gibt es planmäßig ein wöchentliches Gruppentreffen, außerplanmäßige Treffen können spontan organisiert werden. Zwischen den Treffen läuft die Kommunikation in der Gruppe über **VoIP**, **Instant-Messenger** oder persönlich.

Außerdem hält der Phasenverantwortliche Kontakt zu allen Gruppenmitgliedern.

## 7 Glossar

**Administrator:** Ein Benutzer mit speziellen Sonderrechten. Diese Rechte erlauben die Erstellung von Zeugnissen und die Verwaltung der Datenbank.

**App:** Kurzform für Applikation. Der Begriff wird vorrangig im Kontext von mobilen Endgeräten verwendet und bezeichnet ein Anwendungsprogramm.

**Commit:** Das Hinzufügen von vorgenommenen Änderungen in das Versionskontrollsystem.

**Hash Funktion:** Berechnet aus einem beliebig großen Datensatz eine Kennung fester Länge. Unterschiedliche Datensätze sollten immer unterschiedliche Kennung ergeben. Dieser Wert dient zur Überprüfung der Echtheit der eingelesenen Zeugnisse.

**Instant-Messenger:** Zuständig für die direkte Nachrichtenübertragung zum Beispiel bei mobilen Endgeräten wie Smartphones.

**OCR:** Wandelt ein Bild in einen maschinenlesbaren Text um. Voraussetzung für die weitere Verarbeitung des eingelesenen Zeugnisses.

**Open Source:** Software, deren Quelltext öffentlich zugänglich ist und die damit von jeder Person weiterentwickelt werden kann.

**Prototyp:** Eine funktionierende Vorabversion des Produktes.

**QR-Code:** Zweidimensionales Pixelmuster zur Speicherung kleinerer Datenmengen in maschinenlesbarer Form.

**Repository:** Dient zur Verwaltung und Ablage von Dokumenten.

**Secure ID:** Die Secure ID dient zur Identifikation eines bestimmten Zeugnis.

**Server:** Ein Rechner, der anderen Rechner bestimmte Dienste zur Verfügung stellt.

**SVN:** Eine Versionsverwaltung für Dokumente und Verzeichnissen

**Use-Case Diagramm:** Stellt Anwendungsfälle, Use Cases und deren Akteure mit Abhängigkeiten und Beziehungen dar.

**User:** Das englische Wort für Benutzer. Im Gegensatz zu einem Administrator ein Benutzer ohne spezielle Sonderrechte.

**VoIP:** Telefonieren über das Internet. Die Verbindung erfolgt paketorientiert.