**1. Einleitung**

**1.1 Zweck**

Dieses Dokument dient als Grundlage zur Beauftragung des berufsbegleitenden Studienganges, Kommunikations- und Medieninformatik des Matrikel 13, mit der Programmierung einer HfTL-APP. Es setzt dabei die Rahmenbedingungen fest.

**1.1.1 Hintergründe und Ziele des Projekts**

Die Hochschule für Telekommunikation Leipzig (HfTL) ist eine private, staatlich anerkannte Fachhochschule. Träger der HfTL ist die HFTL Trägergesellschaft mbH, eine Beteiligungsgesellschaft der Deutschen Telekom AG. Die Schule befindet sich im Leipziger Stadtteil Connewitz. Es werden sowohl Direkt- als auch duale Studiengänge und berufsbegleitende Studiengänge angeboten.

Aufgrund des umständlichen Beschaffens der Noten und Stundenpläne sowie der Termine für Teletutorings, wird eine Smartphone-Applikation benötigt. Diese stellt ein Benutzerinterface für Studenten der HfTL dar, mit dem der Zugriff auf die in QIS hinterlegten Daten vereinfacht.

**1.2 Produktumfang**

Der Betrieb der HfTL-APP muss auf allen gängigen Android-Smartphones ab Version 4.0 möglich sein.

Durch die APP wird den Studenten der HFTL ermöglicht:

- die aktuellsten Nachrichten der HFTL-Website zu lesen

- auf QIS zu zugreifen

- Studenten sollen so ihre(-n) Noten(-spiegel) aufrufen können

- Vorlesungspläne zu lesen

- Raumbelegungspläne abzurufen

Zusätzlich werden folgende Anforderungen gestellt:

- Erweiterbarkeit für weitere Funktionen und Anwender

- spätere IOS-Version

**1.2.1 Musskriterien**

Zunächst müssen zwingend folgende Punkte des Umfangs erfüllt werden:

- NEWS

- NOTEN

- STUNDENPLAN

**1.2.3 Abgrenzungskriterien**

Die APP soll später auch für zusätzliche Informationen, wie ein Raumplanung erweiterbar sein. Eine spätere Version für IOS-Geräte ist ebenfalls geplant, ist hier aber nicht Bestandteil.

**1.2.4 Kostenrahmen**

Für die Entwicklung der APP soll auf kostenfreie Opensource-Programme oder auf vordefinierte Klassen der Programmierung zurück gegriffen werden.

Außer den personellen Aufwand dürfen keine Zusätzlichen Kosten entstehen.

**1.3 Definitionen, Akronyme, Abkürzungen**

HfTL - Hochschule für Telekommunikation Leipzig

BKMI\_13 - berufsbegleit. Bachelor-Studiengang (Kommunikations- & Medieninformatik)

APP - Applikation

QIS - HIS Online-Portal - Hochschulportal für Studierende, Lehrende und Mitarbeiter

**1.4 Referenzen**

- https://qisweb.hispro.de/tel/rds?state=user&type=0

- https://www.hft-leipzig.de/de/studierende/service/news.html

**2. Allgemeine Übersicht   
2.1 Beschreibung der Ausgangssituation (Ist-Zustand)**

Damit Studenten auf QIS zugreifen kann, müssen diese sich über einen Browser auf der QIS-Seite einloggen und über das unübersichtliche Menü ihre Daten suchen.

Studenten haben die Möglichkeit ihre Noten oder einen Notenspiegel einzusehen.

Für IPhone-Nutzer gibt es bereits eine kostenpflichtige APP , namens Grades, die Noten und Informationen über die angemeldete Prüfungen auslesen kann.

Um die auf QIS hinterlegten Vorlesungspläne abzurufen ist kein Login erforderlich, jedoch ist es notwendig sich umständlich zu dem entsprechenden Studiengang zu navigieren. Einzelne Vorlesungen oder Stundenpläne können als iCal heruntergeladen werden.

Ebenso sind die Teletutorien hier zu finden.

Die aktuellen Nachrichten der Hochschul-Homepage sind auf einer anderen Seite zu finden, welche öffentlich zugänglich ist.

**2.2 Produkteinsatz   
2.2.1 Anwendungsbereiche**

Aktuell soll die APP nur für die Studenten der HfTL zugänglich sein, welche ein Android-Smartphone besitzen.

**2.2.2 Zielgruppen, Qualifikationsniveau**

Da bei der Nutzergruppe von Studenten mit Erfahrung im Umgang mit solchen APP's ausgegangen werden kann, wird auch die Oberfläche dementsprechend gestaltet.

**2.3 Produktfunktionalität**

Die APP soll sich mittels regelmäßiger Abfragen der HfTL-Homepage, sowie von QIS, die News, Noten und Stunden- und ggf. Raumbelegungspläne ziehen und die Noten für den Nutzer lokal auf dem Smartphone speichern und entsprechend darstellen.

Es soll sichergestellt werden das auf sensible Daten wie z.B. Noten auch nur autorisierte Nutzer Zugang bekommen. Die APP soll in deutscher Sprache dargestellt werden. Evtl. wird sie im Nachgang in andere Sprachen übersetzt.

**2.4 Randbedingungen**

Der zeitliche Rahmen für die Entwicklung und Programmierung dieser APP begrenz sich auf das Sommersemester 2015 (SoSe15) und dem Studienmodul Software-Engeneering.

Für die Entwicklung dieser APP, sowie Vertrieb, Programmierung usw. dürfen keine Kosten entstehen.

Supportleistungen werden in der Projektphase über das Projektteam geleistet. Die Wartung wird ebenfalls vom Projektteam übernommen.

Bis mit der Veröffentlichung die APP der Hochschule kostenfreie zur Verfügung gestellt wird. Die Dokumentation wird ebenfalls vollständig an die Hochschule übergeben.

Durch das Projektteam wird es keine weitere Softwarebetreuung, Wartung oder der gleichen geben. Es finden ebenfalls keine Schulungen oder Einweisungen statt.

**2.5 Annahmen und Abhängigkeiten**

Die APP wird für Android-Geräte ab Version 4.0.3 zur Verfügung stehen.

Entsprechend der Vorgaben der Deutschen Telekom AG muss bei der Programmierung der APP, explizit beim Design, auf die Konzernrichtlinien geachtet werden. Es soll zusätzlich auf die Designempfehlungen für Androidgeräte geachtet werden.

**2.6 Verzögerungen**

Durch die strikte Abtrennung des zeitlichen Rahmens auf das Sommersemester 2015 darf es über diesen Zeitraum hinaus nicht zu Verzögerungen kommen.  **3. Anwendungsszenarien   
3.1 Beschreibung aus der Nutzersicht**

Die Benutzeroberfläche muss intuitiv bedienbar sein. Der strukturierte Aufbau durch Kategorien (News, Noten, Stundenplan) soll die Übersichtlichkeit erhöhen.

Die Logindaten werden verschlüsselt auf dem Smartphone gespeichert und auch verschlüsselt übertragen.

Durch eine durchgehende und vollständige Dokumentation soll eine Wartung auch durch spätere Matrikel oder Administratoren der Hochschule möglich sein.

Eine Implementierung weiterer Funktionen soll auch im Nachhinein möglich sein.

**4. Anforderungen**

**4.1 Fachkonzept**

APP wird in Java programmiert um durch Verwendung bestehender Klassen die Erweiterbarkeit und Realisierbarkeit zu vereinfachen. Für das Design werden XML-Stylesheets verwendet.

Pull-& Push-Service werden zur Benachrichtigung und Abfrage verwendet.

**4.1.1 Überblick über das Gesamtsystem**

* Noten
* APP soll die Noten lokal auf dem Smartphone nach Semester aufgeschlüsselt anzeigen
* Login über die APP
* Anmeldung über gesicherte, verschlüsselte Übertragung (HTTPS)
* verschlüsselte Speicherung der Daten auf dem Smartphone (AES)
* Pull-Nachrichten (Einstellbares Intervall und/oder manuell)
* Push-Benachrichtigung
* Nutzer wird mit Hinweismeldung informiert, wenn Noten aktualisiert wurden
* Optionale Angaben im Menüpunkt Noten:
* Klassenspiegel
* Notenverteilung
* Anzahl der Teilnehmer
* Notenschnitt
* Anzeige der Creditpoints
* zu erreichende Creditpoints
* erreichte Creditpoints
* Stundenpläne
* Stundenplan nur für zum Nutzer passenden Studiengang
* Pull-Nachrichten (Einstellbares Intervall und/oder manuelle Abfrage)
* Synchronisierung mit dem Kalender auf dem Smartphone
* News
* Pull-Nachrichten (manuelle Aktualisierung)
* News von: \url{https://www.hft-leipzig.de/de/studierende/service/news.html}

**4.1.2 Verwendet Biblotheken von Drittanbietern**

* jsoup 1.8.1 – MIT-Lizenz
* iCal4j 1.0.6 – BSD-Lizenz

**4.2 Anforderungen an die Datenhaltung**

Alle gespeicherten Daten müssen vor unbefugtem Zugriff geschützt werden, dafür werden Benutzerdaten in einer SQL-Datenbank gespeichert und durch das Passwort und den Benutzernamen freigegeben. Passwort und Benutzername wurden mit AES verschlüsselt.

**4.2.1 allgemeine Beschreibung der Daten**

Sicherheitsrelevante Daten sind in dem Notenteil der App zu finden. Diese sind der Benutzername, das Passwort, die Prüfungs- und Prüfungsvorleistungsergebnisse, der Klassenspiegel und die Creditpoints.

Die öffentlichen Daten sind in den anderen beiden Teilen der App enthalten.

Dazu gehören die News mit ihren Terminen und Inhalt und der Stundenplan mit den Veranstaltungen und den dazugehörenden Informationen wie Dozent und Veranstaltungsort.

**4.3 Anforderungen an die Benutzeroberfläche   
4.3.1 allgemeine Anforderungen an die Oberfläche**

**Maik/Stefan**

**4.3.2 Berechtigungen**

Die APP wird als Open Source bereitgestellt.

Nicht angemeldete Benutzer können sich die News und den Stundenplan anzeigen lassen. Angemeldete Benutzer können zusätzlich ihre Noten abrufen.

Die App braucht folgende Berechtigung auf dem Smartphone:

- Netzwerkstatus abrufen: zur Kontrolle ob das Handy online ist

- Internet: App greift auf das Internet zu

- Telefonstatus: wird zur Verschlüsselung benötigt

**4.3.3 Bildschirmlayout, Dialogstruktur, Dialogabläufe**

**- siehe Anhang 5.7 App-Layout**

**4.4 Leistungsanforderungen**Es wird von einer maximalen Nutzerzahl von 1000 Studenten ausgegangen.

Die Reaktionszeit/Programmstart der APP soll möglichst gering gehalten werden.

Das Datenaufkommen soll möglichst gering ausfallen.

**4.5 Anforderung für Inbetriebnahme und Einsatz   
4.5.1 Sicherheitsziele**

Der Benutzername und das Passwort werden in einen gesicherten Speicherbereich (nur root), mit AES verschlüsselt, gespeichert.

Die Datenbank in der die Noten gespeichert werden, liegt ebenfalls in diesem Bereich und darf nur von der HfTL-App geöffnet werden.

Die Kommunikation mit der HfTL-Website und QIS erfolgt gesichert per HTTPS.

**4.5.2 Installationsprozedur**

Der Wirkbetrieb soll nach Möglichkeit über den Android-Playstore realisiert werden. Falls es dabei zu Lizenz- und Kostenproblemen kommt, wird die APP als .apk über die HfTL-Homepage verteilt.

**4.5.3 Pilot- bzw. Probebetrieb**

**- siehe Anhang 5.8**

**4.5.4 Fehlerreaktion, Garantie, Service, »Wiederanlauf«**

Die Zuverlässigkeit sollte sich durch eine große MTBF (mean time between failure) darstellen.

**4.5.5 Schulungen**

Nach Beendigung des Projektes werden keine Schulungen usw. durchgeführt.

**4.6 Qualitätsanforderungen   
4.6.1 Qualitätsmerkmale**

**Folgende Qualitätsansprüche werden gestellt:**

* Hohe Zuverlässigkeit der Software
* schnelle und zuverlässige Verarbeitung der gewünschten Daten
* sichere Datenspeicherung u. -übertragung mittels entsprechender Verschlüsselung
* Fehler werden mit einer entsprechenden Fehlermeldung beantwortet
* Intuitiv benutzbar
* Leicht zu warten und zu erweitern
* Vollständige Dokumentation des Projektes

**4.6.2 Qualitätssicherung**

Zur Qualitätssicherung werden einheitliche Testprotokolle und die entsprechenden Testkriterien erstellt. Es finden regelmäßige Kontrollen durch die Qualitätssicherung statt, um die Einhaltung der gegebenen Standards zu überprüfen. Es findet ebenfalls eine regelmäßige Kontrolle der Dokumentation statt. Die jeweiligen Software-Prototypen werden nach der Prototypisierung zur Fehlererkennung und zum Funktionstest an das Projektteam geschickt. Zur Verbesserung der Software werden die Tests anhand vorgegebener Testkriterien durchgeführt, um standardisierte Testergebnisse zu erhalten. Die Testergebnisse werden zur Auswertung an die entsprechende Projektgruppen zurück gespiegelt.

**4.6.3 Qualitätsnachweis**

Während des Projektzeitraumes werden sämtliche Meeting, Tests und Kontrollen protokolliert und zur Dokumentation in einem dafür vorgesehen Bereich abgelegt. Die gesetzten Standards, sowie Vorgaben werden eingehalten. Zum Projektabschluss ist der geplante Umfang erreicht und alle Muss-Kriterien sind enthalten. Die APP ist funktionstüchtig.

**4.6.4 Offenlegung der Qualitätskontrollpläne**

|  |  |  |
| --- | --- | --- |
| Nr | Anforderung | Verantwortliche |
| 1 | Projektrahmen festlegen | gesamtes Team |
| 1.1 | Architektur | gesamtes Team |
| 1.2 | Versionsverwaltung | gesamtes Team |
| 1.3 | Vorgehensmodell | gesamtes Team |
|  |  |  |
| 1 | Vollständige Dokumentation | gesamtes Team |
| 1.1 | UML Diagramme erstellen | AD, CM |
| 1.2 | Installationsanleitung und Benutzerhandbuch erstellen | GE, JS, SC |
| 1.3 | Lasten- Pflichtenheft erstellen | SK, SC |
| 1.4 | Protokolle | ML, PK |
|  |  |  |
| 2 | App erstellen | gesamtes Team |
| 2.1 | Design nach HFTl vorgaben erstellen | ML, SC |
| 2.2 | Programmierung der App | GE, SC |
|  |  |  |
| 3 | Qualitätskontrolle | gesamtes Team |
| 3.1 | Testkriterien festlegen | PK |
| 3.2 | Testprotokollelayout erstellen | ML |
| 3.3 | Softwaretests | ML, PK |
| 3.4 | Kontrolle der Dokumentation | PK |
|  |  |  |
| 4 | Anfertigen der Präsentation | gesamtes Team |
| 4.1 | Halten der Präsentation | SC, SK, ML, JS |

**4.6.5 Berichte, Protokolle zum Nachweis des Vorgehens gemäß der Qualitätskontrollpläne**

Es liegen bei:

- siehe 5.9 Sitzungsprotokolle

- siehe 5.10 Testprotokolle

**4.7 Anforderung an die Entwicklung   
4.7.1 Entwicklungs-Umgebung**

Für die Entwicklung wird Android Studio inkl. Gradle in der Version 1.x genutzt. Für die Dokumentation und Projektkoordination wird GitHub verwendet.

**4.7.2 Projekt-Organisation**

- siehe 5.4 Projektorganigramm im Anhang

**4.7.3 Projektplanung**

- siehe 5.5 Projektstrukturplan im Anhang

**4.7.4 Änderungsmanagement**

- siehe 5.3 Prototypenplan (GitHubplan)

**4.7.5 Testanforderungen**

**( einfügen Testprotokollentwurf)  
  
5. Anhang   
5.1 Dokumentation  
5.2 Benutzerhandbuch   
5.3 Prototypenplan  
5.4 Projektorganigramm   
5.5 Projektstrukturplan   
5.6 Gant**

**5.7 Layout**

**5.8 Releases**

**5.9 Sitzungsprotokolle**

**5.10 Testprotokolle**