



Hochschule für Telekommunikation Leipzig
University of Applied Sciences

ENTWICKLUNG EINER HFTL-APP DOKUMENTATION

Studienmodul *Software-Engineering*
der Hochschule für Telekommunikation
Leipzig

Projektarbeit - Softwareentwicklung

vorgelegt von

BKMI Matrikel 13

15. September 2015

Dozent: Profn. Dr.-Ing. Sabine Wieland

INHALTSVERZEICHNIS

1 Einführung	2
1.1 Zweck	2
1.2 Hintergründe und Ziele des Projekts	2
1.3 Produktumfang	2
1.4 Musskriterien	2
1.5 Abgrenzungskriterien	3
1.5.1 Kostenrahmen	3
1.6 Definitionen, Akronyme, Abkürzungen	3
1.7 Referenzen	3
2 Allgemeine Übersicht	4
2.1 Beschreibung der Ausgangssituation (Ist-Zustand)	4
2.2 Produkteinsatz	4
2.2.1 Anwendungsbereiche	4
2.2.2 Zielgruppen, Qualifikationsniveau	4
2.3 Produktfunktionalität	4
2.4 Randbedingungen	4
2.5 Annahmen und Abhängigkeiten	5
2.6 Verzögerungen	5
3 Anwendungsszenarien	5
3.1 Beschreibung aus der Nutzersicht	5
4 Anforderungen	5
4.1 Fachkonzept	5
4.1.1 Überblick über das Gesamtsystem	5
4.1.2 Verwendete Bibliotheken von Drittanbietern	6
4.2 Anforderungen an die Datenhaltung	6
4.2.1 allgemeine Beschreibung der Daten	7
4.3 Anforderungen an die Benutzeroberfläche	7
4.3.1 allgemeine Anforderungen an die Oberfläche	7
4.3.2 Berechtigungen	7
4.3.3 Bildschirmlayout	7
4.4 Leistungsanforderungen	7
4.5 Anforderungen für Inbetriebnahme und Einsatz	8
4.5.1 Sicherheitsziele	8
4.5.2 Installationsprozedur	8
4.5.3 Pilot- bzw. Probeflug	8
4.5.4 Fehlerreaktion, Garantie, Service, »Wiederanlauf«	8
4.5.5 Schulungen	8
4.6 Qualitätsanforderungen	8
4.6.1 Qualitätsmerkmale	8

4.6.2	Qualitätssicherung	9
4.6.3	Qualitätsnachweise	9
4.6.4	Offenlegung der Qualitätskontrollpläne	9
4.6.5	Berichte, Protokolle zum Nachweis des Vorgehens gemäß der Qualitätskontrollpläne	10
4.7	Anforderung an die Entwicklung	10
4.7.1	Entwicklungs-Umgebung	10
4.7.2	Projekt-Organisation	10
4.7.3	Projekt-Planung	13
4.7.4	Änderungsmanagement	13
4.7.5	Testanforderungen	13
5	Anhang	13
5.1	Benutzerhandbuch	15
5.1.1	Funktionsumfang	15
5.1.2	Installation	16
5.1.3	Startbildschirm	18
5.1.4	Newsansicht	19
5.1.5	Noten	19
5.1.6	Stundenplan	22
5.2	Entwicklerhandbuch	23
5.3	Gantt	73
5.4	App-Layout	78
5.5	Release-Historie	85
5.6	Testprotokollentwurf	94
5.7	Testprotokolle	106
5.8	Sitzungsprotokolle und Aufgabenlisten	216

1 EINFÜHRUNG

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.

1.2 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 Beteiligungsge-sellschaft der Deutschen Telekom AG. Die Hochschule 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 Stu-denpläne 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 im QIS hinterlegten Daten vereinfacht werden soll.

1.3 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-Webseite zu lesen
- auf QIS zuzugreifen
- Studenten sollen ihre(-n) Noten(-spiegel) aufrufen können
- Stundenpläne zu lesen

Zusätzlich werden folgende Anforderungen gestellt:

- Erweiterbarkeit für weitere Funktionen und Anwender
- spätere IOS-Version

1.4 MUSSKRITERIEN

Zunächst müssen zwingend folgende Punkte des Umfangs erfüllt werden:

- NEWS
- NOTEN
- STUNDENPLAN

1.5 ABGRENZUNGSKRITERIEN

Die APP soll später auch für zusätzliche Informationen, wie eine Raumplanung erweiterbar sein. Versionen für andere Betriebssysteme müssen in einem separaten Projekt bearbeitet werden und sind nicht Bestandteil dieses Projektes.

1.5.1 KOSTENRAHMEN

Für die Entwicklung der APP soll auf kostenfreie Opensource-Programme oder auf vordefinierte Klassen für die Programmierung zurückgegriffen werden. Außer den personellen Aufwand dürfen keine Zusätzlichen Kosten entstehen.

1.6 DEFINITIONEN, AKRONYME, ABKÜRZUNGEN

HfTL	Hochschule für Telekommunikation Leipzig
APP	Kurzform für Applikation
mbH	mit beschränkter Haftung
QIS	Qualitätssteigerung der Hochschulverwaltung im Internet durch Selbstbedienung
iCal	Datenformat zum Austausch von Kalenderinhalten
SoSe15	Sommersemester 2015
XML	Extensible Markup Language
HTTPS	HyperText Transfer Protocol Secure
AES	Advanced Encryption Standard
SQL	Structured Query Language
.apk	Android application package
MTBF	mean time between failure
CI/CD	Corperate Identity/Corperate Design
GUI	Graphical User Interface

1.7 REFERENZEN

- QIS-System: <https://qisweb.hispro.de/tel/rds?state=user&type=0>
- News der HfTL <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 können, 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 Stundenpläne abzurufen ist kein Login erforderlich, jedoch ist es notwendig sich umständlich zu dem entsprechenden Studiengang zu navigieren. Die Termine lassen sich einzeln oder als Block im [iCal](#)-Format herunterladen. Die Termine der Teletutorien sind ebenfalls im QIS zu finden. Die aktuellen Nachrichten der Hochschule sind auf der Webseite selbiger 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 mittels regelmäßiger Abfragen der [HfTL](#)-Homepage, sowie von [QIS](#), die News, Noten, Stunden- und ggf. Raumbelegungspläne aktualisieren 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 erhalten. 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 begrenzt sich auf das Sommersemester 2015 ([SoSe15](#)) und dem Studienmodul Software-Engineering. 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 während der Projektphase vom Projektteam übernommen, bis die APP der Hochschule kostenfrei zur Verfügung gestellt wird. Die Dokumentation wird ebenfalls vollständig an die Hochschule übergeben. Durch das Projektteam wird es nach

Ende des Projektes 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 gestellt. 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 SoSe15 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

Die HfTL-APP wird in Java programmiert, um durch Verwendung bestehender Klassen die Erweiterbarkeit und Realisierbarkeit zu vereinfachen. Für das Design werden XML-Stylesheets verwendet. Pull-und Push-Services werden zur Benachrichtigung und Abfrage verwendet.

4.1.1 ÜBERBLICK ÜBER DAS GESAMTSYSTEM

Noten

- die APP soll die Noten lokal auf dem Smartphone nach Semester geordnet anzeigen
- Login im QIS-System durch die APP

- Anmeldung über gesicherte, verschlüsselte Übertragung ([HTTPS](#))
- verschlüsselte Speicherung der Daten auf dem Smartphone ([AES](#))
- Pull-Nachrichten (Einstellbares Intervall und/oder manuell)
- Benachrichtigung bei Eingang von neuen Noten
- Nutzer wird mit Hinweismeldungen 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: <https://www.hft-leipzig.de/de/studierende/service/news.html>

4.1.2 VERWENDETE BIBLIOTHEKEN VON DRITTANBIETEREN

- 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 werden 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 die Stundenpläne 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

Das Layout richtet sich nach dem Corporate Identity/Corporate Design ([CI/CD](#)) der Deutschen Telekom AG, speziell dem der [HFTL](#) Trägergesellschaft, stand 14.12.12. Entsprechend sind primär die Farben, sowie Schriftarten vorgegeben. Das Layout zieht sich einheitlich (mit funktionsbedingten Abweichungen) durch alle APP-Teile und soll eine leichte Bedienung begünstigen. Die Größe der einzelnen Elemente (Buttons, Zeilen, Überschriften etc.) ist an den Designempfehlungen seitens Android angelehnt. Auch die Abstände zwischen den einzelnen Elementen wurden – sofern sie mit dem [CI/CD](#) in Einklang stehen – entsprechend der Designempfehlung festgelegt. Die Reaktionszeit der Benutzereingaben soll mit möglichst geringer Verzögerung verarbeitet und dargestellt werden.

4.3.2 BERECHTIGUNGEN

Die APP wird als Open Source bereitgestellt.

Nicht angemeldete Benutzer können sich die News und die Stundenpläne 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
- Kalendereinträge bearbeiten

4.3.3 BILDSCHIRMLAYOUT

siehe Anhang [5.4](#)

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 ANFORDEUNGEN FÜR INBETRIEBNAHME UND EINSATZ

4.5.1 SICHERHEITSZIELE

Der Benutzername und das Passwort werden in einem 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**-Webseite 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 Android application package (**.apk**) über die HfTL-Homepage verteilt.

4.5.3 PILOT- BZW. PROBE BETRIEB

siehe Anhang Versionsmanagementprotokolle **5.5**

4.5.4 FEHLERREAKTION, GARANTIE, SERVICE, »WIEDERANLAUF«

Die Zuverlässigkeit sollte sich durch eine große mean time between failure (**MTBF**) 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 QUALITÄTSNACHWEISE

Während des Projektzeitraumes werden sämtliche Meeting, Tests und Kontrollen protokolliert und zur Dokumentation in einem dafür vorgesehen Bereich abgelegt. Die gesetzten Standards und 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 QUALITÄSKONTROLLPLÄNE

Nr.	Anforderung	Verantwortliche
1	Projektrahmen festlegen	gesamtes Team
1.1	Projektrahmen festlegen	gesamtes Team
1.2	Architektur	gesamtes Team
1.3	Vorgehensmodell	gesamtes Team
2	Vollständige Dokumentation	gesamtes Team
2.1	UML Diagramme erstellen	AD, CM
2.2	Installationsanleitung und Benutzerhandbuch erstellen	GE, JS
2.3	Lasten- Pflichtenheft erstellen	SK, SC
2.4	Protokolle	ML, PK
2.5	Dokumentation zusammenstellen und gestalten	JS, SC
3	App erstellen	gesamtes Team
3.1	Design nach HfTL-Vorgaben erstellen	ML, SC
3.2	Programmierung der APP	GE
4	Qualitätskontrolle	gesamtes Team
4.1	Testkriterien festlegen	PK
4.2	Testprotokollelayout erstellen	ML
4.3	Softwaretests	ML, PK
4.4	Kontrolle der Dokumentation	PK
5	Anfertigen der Präsentation	gesamte Team
5.1	Präsentation halten	SC, ML, SK, JS

4.6.5 BERICHTE, PROTOKOLLE ZUM NACHWEIS DES VORGEHENS GEMÄSS DER QUALITÄTSKONTROLLPLÄNE

siehe Anhang "Testprotokolle" [5.7](#) und Anhang "Sitzungsprotokolle" [5.8](#)

4.7 ANFORDERUNG AN DIE ENTWICKLUNG

4.7.1 ENTWICKLUNGS-UMGEBUNG

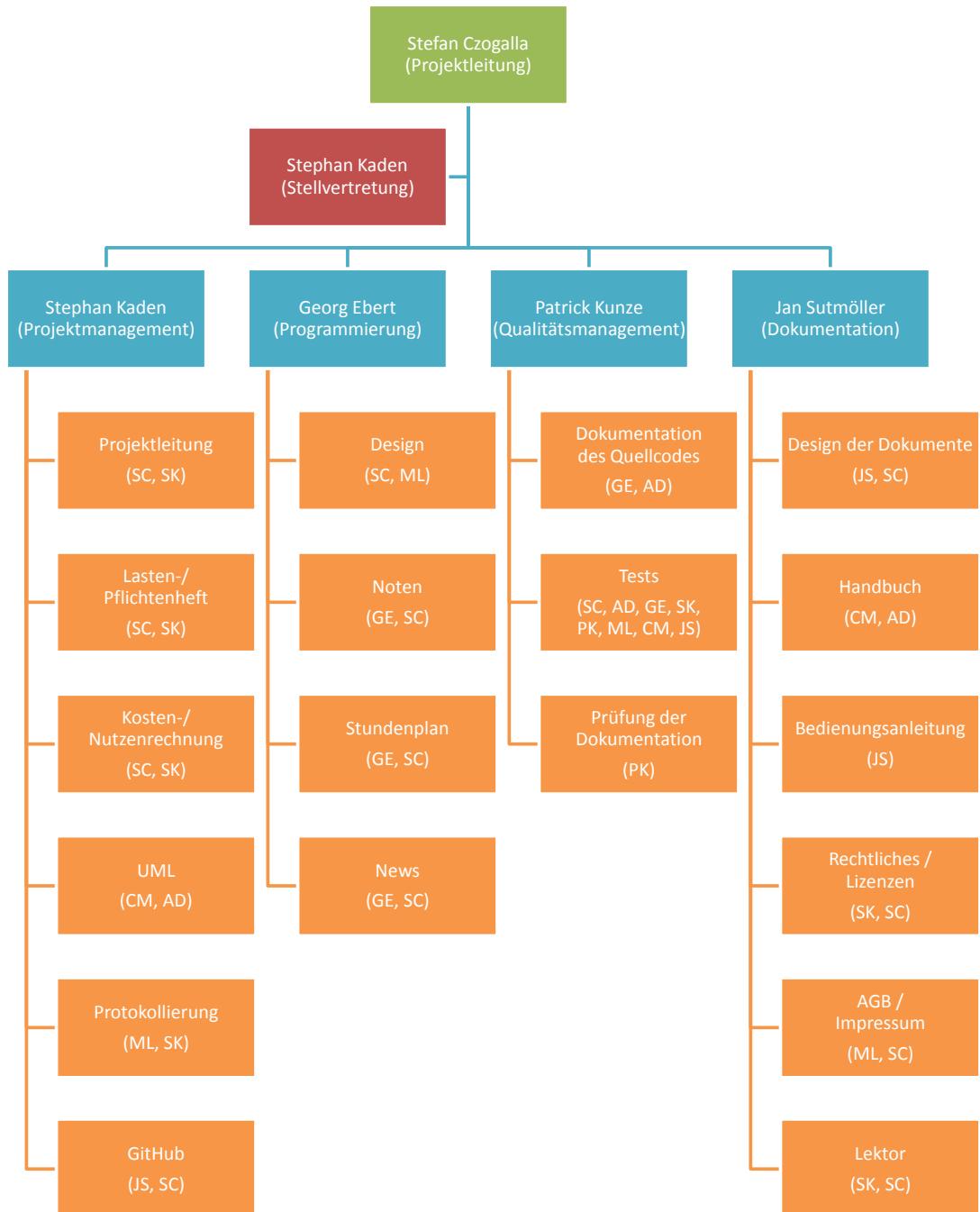
Für die Entwicklung wird Android Studio inkl. Gradle in der Version 1.x genutzt. Für die Dokumentation und Projektkoordination wird GitHub verwendet. Die Dokumentation wird mittels \LaTeX erstellt.

4.7.2 PROJEKT-ORGANISATION

Die Projektorganisation wird wie in folgender Abbildung strukturiert. Als Vorgehensmodell wird das Spiralmodell mit Prototyping gewählt. Es ist ein iteratives Modell, wobei jeder Zyklus in den einzelnen Quadranten folgende Aktivitäten enthält:

- Festlegung von Zielen, Identifikation von Alternativen und Beschreibung von Rahmenbedingungen
- Evaluierung der Alternativen und das Erkennen, Abschätzen und Reduzieren von Risiken, z. B. durch Analysen, Simulationen oder Prototyping
- Realisierung und Überprüfung des Zwischenprodukts
- Planung des nächsten Zyklus der Projektfortsetzung.

Die einzelnen Aufgaben werden Personen zugeordnet. In wöchentlichen Online-Meetings über Teamviewer stellt jeder seine Ergebnisse vor und es werden diese bewertet. Anhand dieser Ergebnisse werden für den neuen Zyklus Aufgaben verteilt. Der Protokollführer hält alle Ergebnisse und Aufgaben fest und legt die Protokolle im Projektordner ab.





4.7.3 PROJEKT-PLANUNG

siehe Anhang "Gantt" [5.3](#)

4.7.4 ÄNDERUNGSMANAGEMENT

Zur Versionsverwaltung wurde Git eingesetzt. Als Hosting-Anbieter wurde dabei auf GitHub gesetzt, welcher einen kostenfreien Zugang für nicht kommerzielle Projekte bereitstellt. Ein Graphical User Interface ([GUI](#)) oder ein Konsolenprogramm für Windows und Linux übernehmen dabei die Steuerung der Versionsverwaltung. Konflikte in den einzelnen Versionen können nur über die Konsole behoben werden. Auf der Webseite von GitHub können Milestones erstellt werden und an die jeweiligen Mitarbeiter zugeteilt werden. In den Milestones werden einzelne Aufgaben, sogenannte Issues angelegt und wiederum den Bearbeitern zugeordnet, somit ist der Bearbeitungsstand zu jeder Zeit des Projektes ersichtlich und es kann schnell auf sich ergebende Probleme reagiert werden.

4.7.5 TESTANFORDERUNGEN

siehe Anhang "Testprotokollentwurf" [5.6](#)

5 ANHANG



Benutzerhandbuch

- HfTL-APP -

5.1 BENUTZERHANDBUCH

5.1.1 FUNKTIONSUMFANG

In diesem Dokument werden die Benutzerfunktionen der HfTL-APP für Android-Geräte beschrieben. Es dient als Benutzerhandbuch für die unterschiedlichen Funktionen der Anwendung und soll Ihnen beim Ausführen von häufigen Aktionen innerhalb der Anwendung Hilfe bieten.

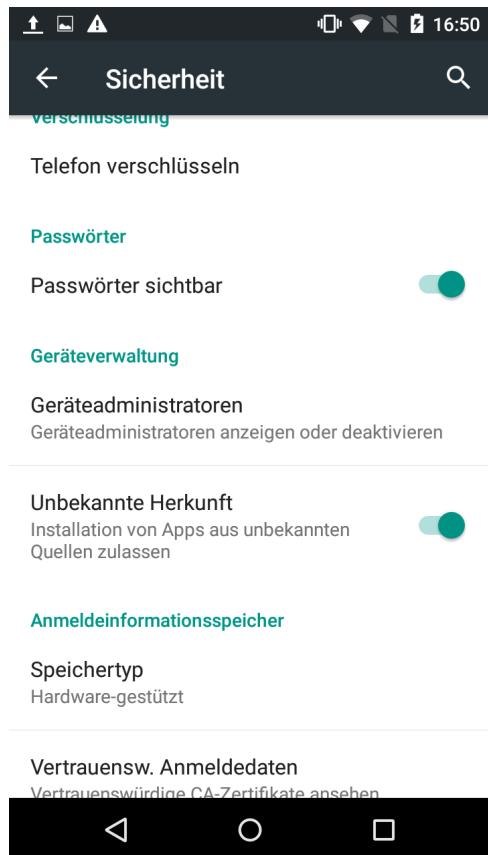
Die HfTL-APP ist eine mobile Informationslösung für Android Geräte. Die App kann kostenlos über das Rechenzentrum der Hochschule für Kommunikation-Leipzig bezogen werden.

Die HfTL-APP bietet folgende Funktionen:

- Abfrage der News von der HfTL-Homepage
- Abfrage der Noten aus QIS/HIS nach erfolgreicher Anmeldung an dem betreffenden System
- Abfrage des zu einem Studenten passenden Stundenplans

5.1.2 INSTALLATION

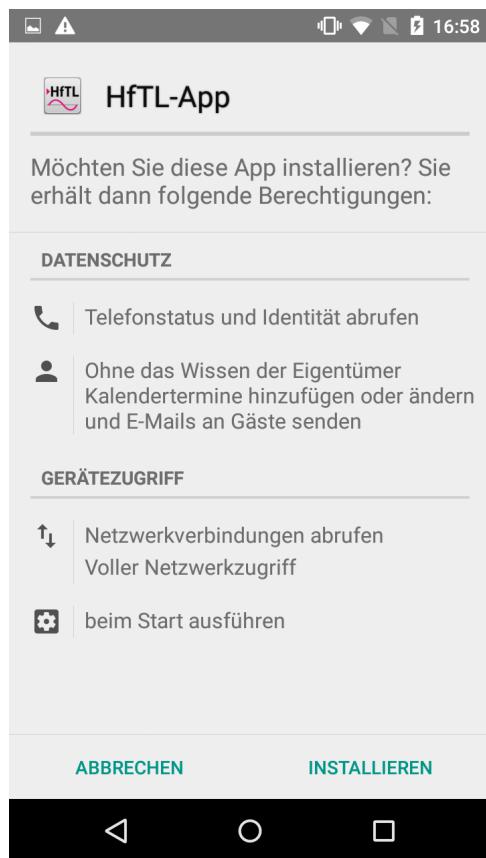
Vor der Installation müssen auf Ihrem Smartphone APP's mit unbekannter Herkunft freigegeben werden. Dies kann von Smartphone zu Smartphone unterschiedlich sein. In den meisten Fällen (hier unter Android V5.1) findet man den Punkt unter 'Einstellungen/Sicherheit/Unbekannte Herkunft'



Falls es bei Ihnen anders sein sollte, informieren Sie sich bitte bei Ihrem Handyhersteller dazu.

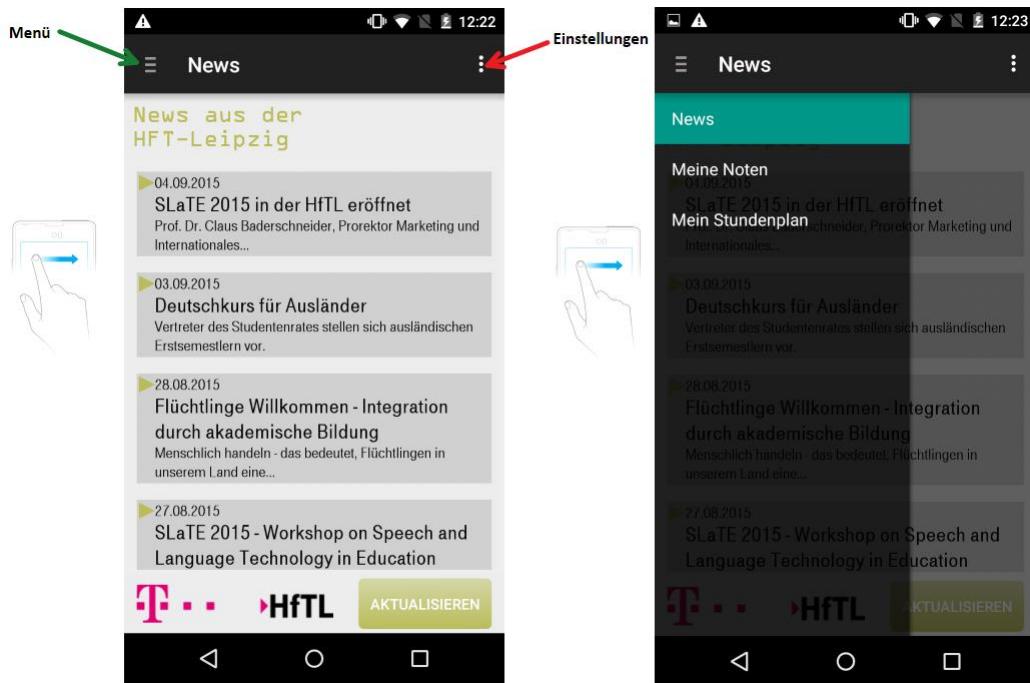
Um die Installation durchzuführen, laden sie sich bitte die APP von der HFTL-Webseite auf Ihr Android-Smartphone herunter. Nach Starten der Installation erscheint zunächst die Abfrage der Berechtigungen.

- Telefonstatus und Identität abrufen
- Ohne das Wissen der Eigentümer Kalendertermine hinzufügen oder ändern und E-Mails an Gäste senden
- Netzwerkverbindungen abrufen, Voller Netzwerkzugriff
- beim Start ausführen



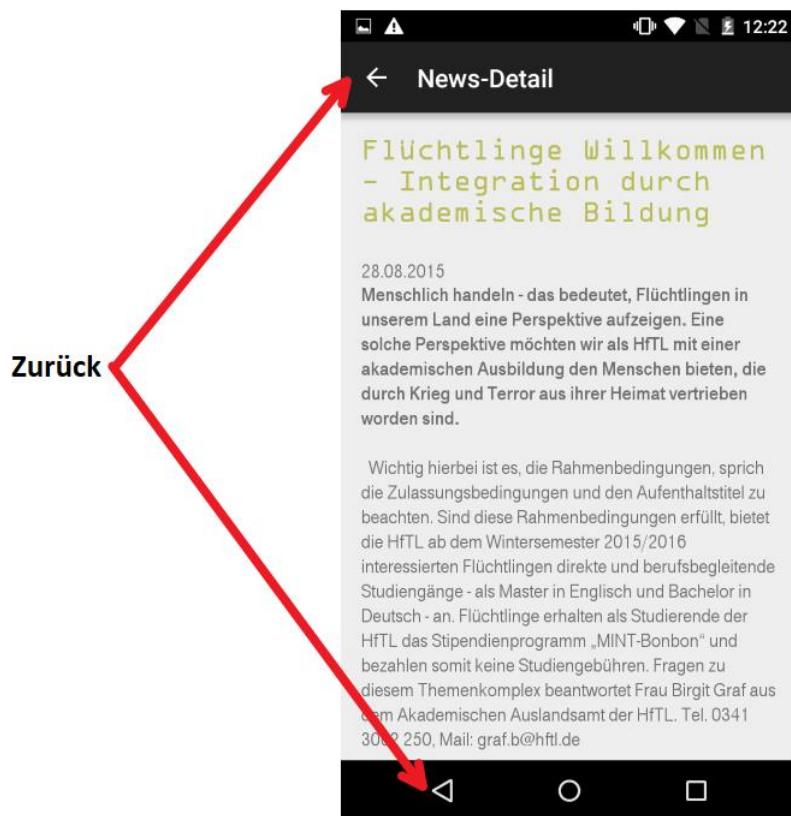
Wenn Sie damit einverstanden sind, bestätigen Sie dieses bitte mit dem Button 'Installieren'
Die Installation ist abgeschlossen.
Deinstalliert werden kann die APP wie jede Andere auf Ihrem Smartphone auch.

5.1.3 STARTBILDSCHIRM



Nach dem Starten der APP erscheint zunächst die News-Seite. Die News werden bei bestehender Internetverbindung automatisch aktualisiert. Mit klicken auf den Aktualisierungsbutton kann eine manuelle Aktualisierung durch den Nutzer angestoßen werden. Über den Menü-Button gelangt der Nutzer in das Programm-Menü. Ein Wischen vom linken Bildschirmrand in die Mitte öffnet ebenfalls das Menü. Über den Einstellungs-Button gelangt man in das Einstellungs-Menü. Der Einstellungsbbutton wird nicht bei jedem Gerät gesondert angezeigt. Wenn das Gerät einen Einstellungsbutton unter dem Bildschirm hat z.B. beim Galaxy S3 muss dieser genutzt werden. Mit Auswählen der einzelnen News gelangt man in deren Detailansicht.

5.1.4 NEWSANSICHT



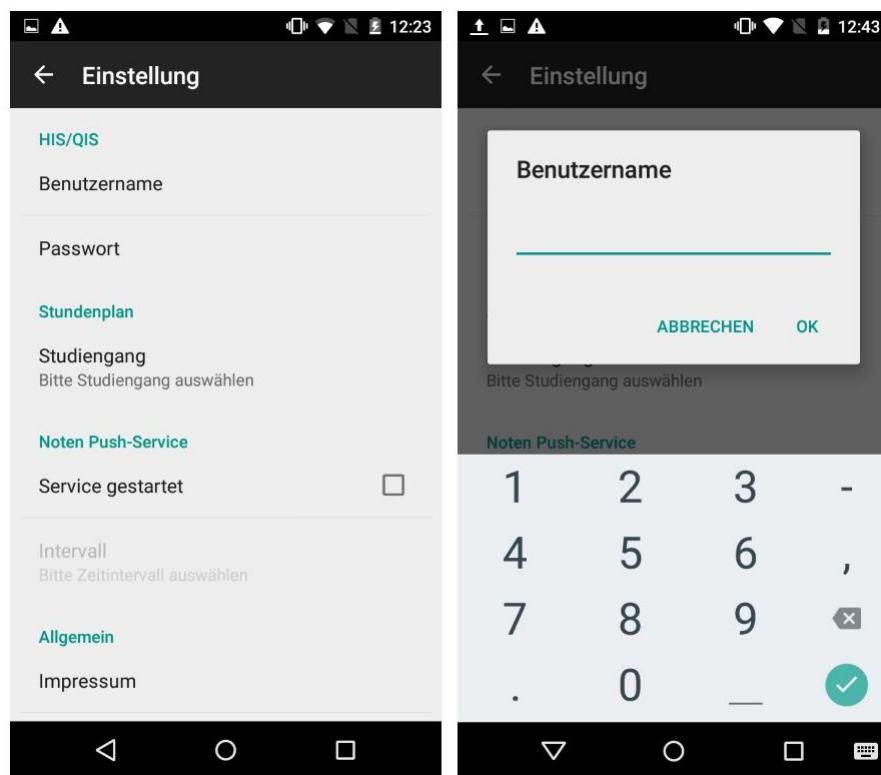
Mit den Zurück-Buttons gelangt man in die vorherige Ansicht.

5.1.5 NOTEN

Um die Noten abzurufen geht man zunächst in den Einstellungskontext. Dort kann der Nutzer und das jeweilige Passwort eingegeben werden.

Mit einem Klick auf Benutzername bzw. Passwort öffnet sich ein neuer Kontext, welcher zum eingeben des Benutzernamens bzw. Passwortes auffordert.

Die Eingabe wird mit "OK" gespeichert und mit "Abbrechen" verworfen. Bei beiden Aktionen schließt sich der Kontext. Mit Auswählen des Punktes "Service gestartet" weist man die APP an, den Service im Hintergrund zu starten. Das Intervall zum Abfragen auf neue Noten kann dann über den Punkt "Intervall" ausgewählt werden.



Nach erfolgreichem Eintragen des Benutzeraccounts und verlassen der Einstellungen kann über den Menü-Button der Punkt “Noten“ ausgewählt werden. Hier werden die aktuellen Noten aus QIS/HIS geladen und angezeigt. Sollte es beim Anmelden an QIS/HIS zu einem Fehler kommen, erscheint eine Fehlermeldung und die vorher vorgenommenen Einstellungen sollten noch einmal kontrolliert werden.



5.1.6 STUNDENPLAN

Um sich den jeweiligen Stundenplan anzeigen zu lassen, muss in den Einstellungen unter "Studiengang" das korrekte Matrikel ausgewählt werden. Den Stundenplan erreicht man über das Menü.



5.2 ENTWICKLERHANDBUCH



Entwicklerhandbuch

- HfTL-APP -

Stand: 15. September 2015

INHALTSVERZEICHNIS

1 Allgemeines	3
2 Verwendete Software	3
3 Aufbau des Projekts	5
3.1 Manifest.XML	5
3.2 Ordnerstruktur	6
4 Activities	10
4.1 NewsActivity.java	10
4.2 EinstellungsActivity.java	11
5 NewsClickedActivity.java	12
5.1 Allgemein	12
5.2 Klasse DetailHelper	12
6 Fragmente	14
6.1 NewsFragment	14
6.2 Klasse NewsHelper	16
6.3 NavigationDrawerFragment	17
6.4 Notenfragment	19
6.5 StundenplanFragment	23
7 CustomAdapter	25
7.1 Allgemein	25
7.2 CustomAdapterNews.java	27
7.3 CustomAdapterNoten.java	27
7.4 CustomAdapterStundenplan	28
8 XML-Dateien	29
8.1 fragment.xml	29
8.2 _list.xml	29
8.3 activity_news_clicked.xml	29
8.4 settings_toolbar	29
8.5 impressum.xml & activity_impressum	29
8.6 colors.xml	30
8.7 array.xml	30
8.8 einstellung.xml	30
8.9 strings.xml	30
9 Layout	32
9.1 Allgemeines	32
9.2 Schriftarten	32

9.3 Buttons	34
10 Erweiterungen und Verbesserungen für kommende Versionen	36
10.1 Pflege der array.xml	36
10.2 Raumplan	36
10.3 Portierung auf andere Systeme	37
10.4 Speiseplan	37
10.5 Login und Funktionen für Lehrkräfte	37
10.6 Unterstützung von Tabletts	37
10.7 Landscape-Modus	37
10.8 Bekannte Fehler	37
11 UML	38

1 ALLGEMEINES

Dieses Dokument dient lediglich als Hilfswerkzeug zur (Weiter-)Entwicklung und Wartung der HfTL-App. In diesem Dokument ist der grobe Aufbau, sowie die wichtigsten verwendeten Funktion mitsamt zugehörigen Quelltext erklärt.

Dieses Dokument ist keine Programmieranleitung.

Es empfiehlt sich, gewisse Grundkenntnisse in objektorientierter Programmierung im Allgemeinen und in JAVA, XML und AndroidStudio im Speziellen mitzubringen, um dieses Dokument effizient nutzen zu können.

Standard-Methoden und Klassen sind nicht im Detail erklärt, da das den Rahmen dieses Dokuments übersteigen würde. Für tiefer greifende Informationen wird die [Android-API](#) empfohlen.

2 VERWENDETE SOFTWARE

AndroidStudio

AndroidStudio ist die Standard-Entwicklungsumgebung für Android. Es bietet bereits ein fertiges Gerüst für eine funktionsfähige App an. Das Programm bietet Klassenbibliotheken, Debugger und selbst ein Emulator mit dessen Hilfe Android-Endgeräte auf dem PC virtuell dargestellt werden können.

Das Programm kann kostenlos unter developer.android.com heruntergeladen werden.

GitHub

GitHub ist ein onlinebasierter Dienst, der es einem ermöglicht, Dateien zu hosten und parallel im Team zu bearbeiten. Zudem bietet GitHub eine einfache und übersichtliche Form der Versionierung.

Durch das Erstellen unterschiedlicher Repositorys kann an mehreren Stellen eines Projekts gleichzeitig gearbeitet und getestet werden, ohne Datenverluste befürchten zu müssen.

Gimp

Gimp ist ein Open-Source Bildbearbeitungsprogramm. Es wurde in dem konkreten Fall genutzt, um Grafiken für die App zu erstellen.

Das Programm kann kostenlos auf der Seite des Herstellers (www.gimp.org) herunter geladen werden.

Microsoft Office

MS Office ist das Office Paket von Microsoft, welches bei der Entwicklung der App nur eine beiläufige Rolle spielte. Genutzt wurde insbesondere MS Word zur schnellen Erstellung von Fließtext bzw. der Dokumentation. Die Eigentliche Aufbereitung des Textes erfolgte dann in LaTex.

Die Erstellung der Organigramme bzw. Pläne erfolgte durch MS Excel.

LATEX

ist ein Softwarepaket, das die Benutzung des Textsatzsystems TeX mit Hilfe von Makros vereinfacht. Die Projektdokumentation wurde mit TeX entwickelt

3 AUFBAU DES PROJEKTS

3.1 MANIFEST.XML

Diese Datei ist für jede Android-App zwingend notwendig. Hier werden grundsätzliche Dinge definiert, z.B. Welche Berechtigung diese App benötigt und auf welche Hardware im laufenden Zustand zugegriffen werden muss.

Des Weiteren wird hier auch das package für den Javaquellcode definiert:

Listing 1: AndroidManifest.XML

```
<manifest xmlns:android="..." package="bkmi.de.hftl_app" >
```

Die Zugriffsberechtigungen sind wie folgt definiert:

Listing 2: AndroidManifest.XML

```
<uses-permission android:name="android.permission.INTERNET" />
<uses-permission android:name="android.permission.
    ACCESS_NETWORK_STATE" />
<uses-permission android:name="android.permission.READ_PHONE_STATE
    " />
```

Bei der Installation der App wird der Nutzer entsprechend informiert, dass die App auf die jeweiligen Funktionen des Endgerätes zugreift.

In der manifest.xml ist ebenfalls eine Übersicht über die verwendeten Verzeichnisse und Komponenten hinterlegt, z.B. für die Activities.

Bei der HfTL-App ist der Verweis für die MainActivity (die Activity, mit der die App startet) für die NewsActivity gesetzt.

Die folgenden Tags wurden bei der App nicht verwendet, sind aber theoretisch für Erweiterungen möglich:

Listing 3: Zugriffsbeispiel

```
<uses-feature android:name="android.hardware.camera" />
```

Das wäre ein Tag, um im laufenden Betrieb auf die Kamera des Telefons zuzugreifen.

3.2 ORDNERSTRUKTUR

Database

- beinhaltet die NotenDB.java – Inhalt ist der Connector und die Kernfunktionen um Inhalte der eigentlichen Datenbank zu aktualisieren und zu modifizieren
- NotenTabelle.java – das ist die eigentliche Datenbank, bzw. die eigentliche Definition vom Aufbau der Datenbank

Fragmente

- beinhaltet die Fragmente, die von den Activities verwendet werden.

Service

- beinhaltet die Datei NotenService.java, die zum Erzeugen von Push-Nachrichten dient, sobald es in der Notenübersicht neue Noten für den jeweiligen Studenten gibt.

Help

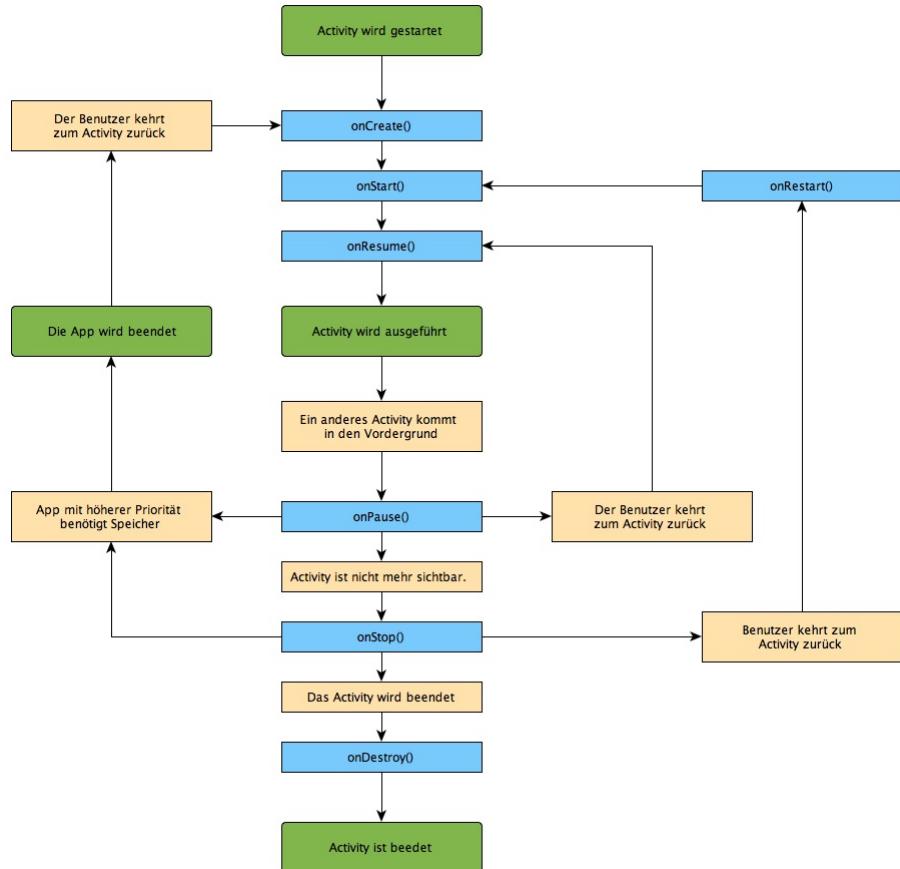
- beinhaltet diverse Hilfsfunktionen, die unter anderem zum Ausführen von Threads dienen.

Ressourcen

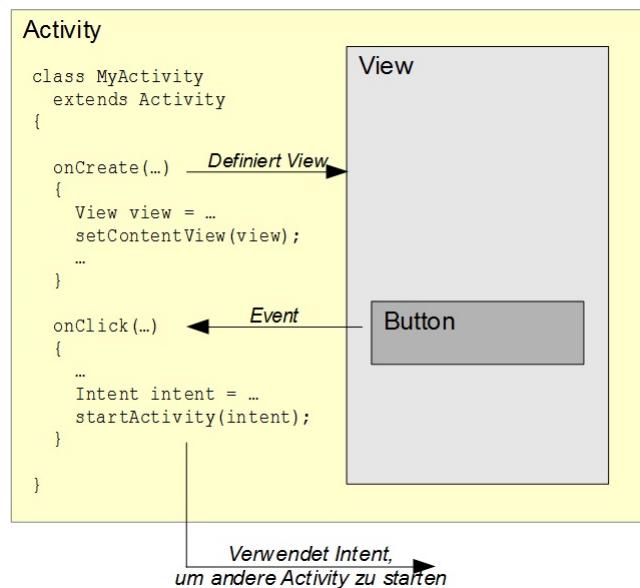
- unter /src/main/res befindet sich eine Ordnerstruktur, welche XML- und Bilddateien für verschiedenste Anwendungszwecke beinhaltet. Diese werden beim Kompilieren des Projekts in die Ressourcendatei R.java geschrieben. Über diese Datei, wird dann auf die Ressourcen zugegriffen.
Es wird im Weiteren nicht detailliert auf alle Dateien eingegangen. Es wird nur auf jene Dateien eingegangen, deren Inhalt zur Schilderung von wichtigen Kernfunktionen dienlich ist.

Activities

- Eine Activity stellt ein sichtbares Fenster dar, welches die eigentliche Interaktion mit dem Nutzer ermöglicht. Diese Interaktionen werden als Events bezeichnet bzw. behandelt. Durch sogenannte Intents ist es möglich andere Activities zu starten. Der sichtbare Teil der Activity wird durch eine View definiert. Dort befinden sich dann die Eventauslöser, wie zum Beispiel Buttons etc.
- Folgende Grafik dient zur Übersicht:

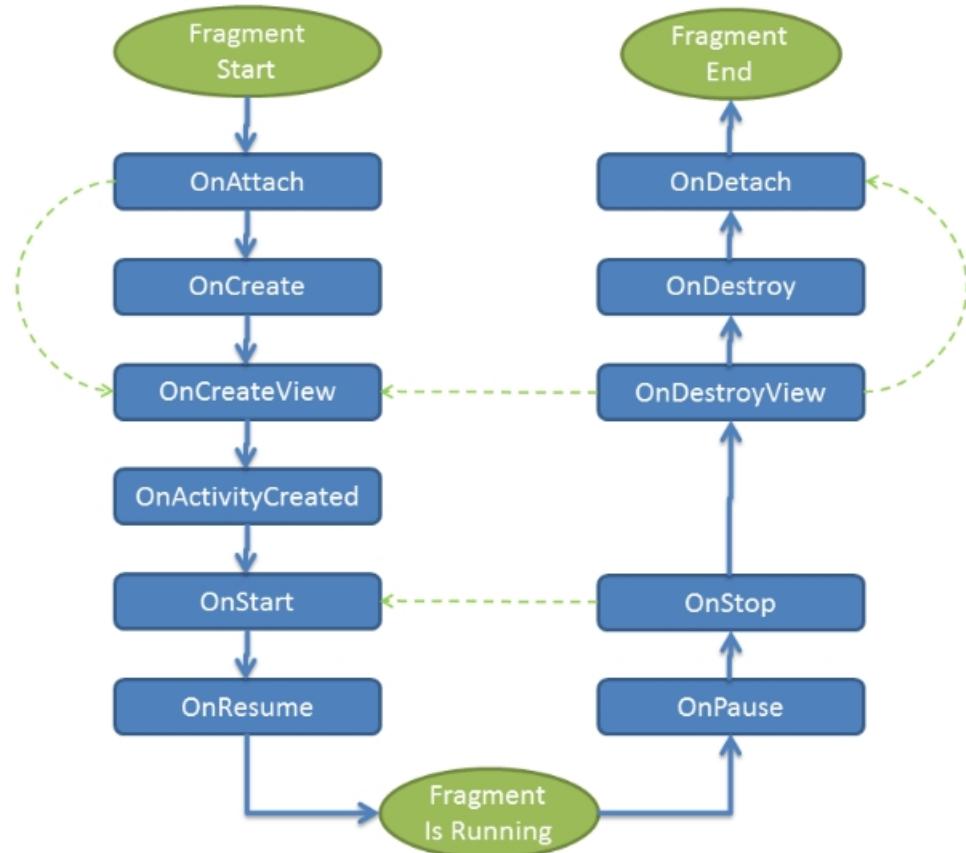


- Jede Activity hat einen sogenannten “Lifecycle“ (=Lebenszyklus), der von dem Betriebssystem gesteuert und verwaltet wird. Dem Entwickler ist es nun überlassen, welche Ressourcen er entsprechend sichert, sollten die Ressourcen auf dem Mobiltelefon knapp werden. Das passiert in den Events:
 - onPause(): die App ist noch sichtbar, aber nicht mehr im Vordergrund
 - onStop(): die App ist nicht mehr sichtbar
- Lifecycle einer Activity:



Fragmente

- In Android ist es nicht möglich, mehrere Activities gleichzeitig anzuzeigen, wenn es der Platz auf dem Display theoretisch erlauben würde. Um das dem Entwickler dennoch zu ermöglichen, bietet Android den Objekttypen "Fragment" an.
- Fragmente sind UI-Container, die über laufende Activities gelegt werden. Dies macht besonders dann Sinn, wenn es um die Fragmentierung der App geht (verschiedene Displaygrößen) bzw. wenn man zwischen den Ansichten wechselt (Haltung des Telefons)
- Ebenso wie eine Activity hat ein Fragment ein Lifecycle, welcher in direkter Kommunikation mit dem Lifecycle der Activity steht. So werden zum Beispiel alle Fragmente einer Activity pausiert, wenn die Activity selbst pausiert wird. Folgendes Diagramm soll dies veranschaulichen:



4 ACTIVITIES

4.1 NEWSACTIVITY.JAVA

- Klasse wird mit Fragment extended
- **NavigationDrawerFragment** wird benötigt, um die Interaktion mit Fragmenten der Anzeige zu ermöglichen
- Intent wird benötigt, um andere Activities neben der NewsActivity zu verwalten.
- mTitle ist der Titel des zuletzt geladenen Screens

onCreate()

- Hier wird das Layout geladen, welches in der zugehörigen xml definiert wurde
- Laden des mNavigationDrawerFragment, zum Abbilden des Menüs am linken Seitenrand
- Laden des Layouts für das Fragment

Durch das Laden des Layouts für das mNavigationDrawerFragment (durch setUp) wird die Funktion onNavigationDrawerItemSelected gerufen, weil in der Layoutdefintion Item 1 der Listview ausgewählt wird.

onNavigationDrawerItemSelected()

- Durch das Aufrufen der Funktion wird das eigentliche Fragment für den FragmentManager ausgewählt (durch die switch-case-Anweisung) und abschließend durch das commit aktiv geladen.

onSectionAttached()

- Je nach Auswahl wird hier mTitle aktualisiert.

restoreActionBar()

- Dient zum Aktualisieren der Titelleiste (durch setTitle und mTitle als Argument)

onCreateOptionsMenu()

- Erstellt das Menü oben rechts (drei Punkte) und befüllt es mit den Daten aus der /res/menu/news.xml

onOptionsItemSelected()

- Überprüft welches Element aus dem Menü ausgewählt wurde

4.2 EINSTELLUNGSACTIVITY.JAVA

onCreate()

- Einbinden der “einstellung.xml“ (beinhaltet Definitionen für Stringvariablen, Listen und Checkboxen)
- Setzen von Startwerten für shared, check und list
- Ausführen von registerPreferenceListener()

registerPreferenceListener()

- Es wird ein anonymer Listener erstellt, der auf Änderungen in der SharedPreference.xml reagiert
- In der Methode onSharedPreferenceChanged(SharedPreferences prefs, String key) werden die Änderungen abgefangen
- Listener wird am SharedPreferences Objekt registriert

testeBenutzerdaten()

- Funktion zum Überprüfen von Anmeldedaten
- via TextSecure ts wird Ver- und Entschlüsselung gewährleistet

keineBenutzerdaten()

- Funktion zum Ausgeben, dass die Anmeldeinformationen falsch eingegeben wurden

5 NEWSCLICKEDACTIVITY.JAVA

5.1 ALLGEMEIN

Diese Activity wird geladen, sobald in der News-Übersicht ([NewsActivity.java](#)) ein Eintrag geöffnet wird.

Der Inhalt wird durch NewsResolver und dessen Funktion `getDetailsStringArray` in das String-Array "s" geladen.

`onCreate()`

- Laden des in der zugehörigen xml definierten Layouts (`activity_news_clicked.xml`)
- Einbinden der "Extras" (Übergebene Variablen)
- Zuweisung der URL aus dem rufenden NewsFragment
- Aufruf des DetailHelpers durch

```
new Detailhelper().execute();
```

5.2 KLASSE DETAILHELPER

`onPreExecute()`

- Anzeige eines ProgressDialogs (Ladebalken mit Hinweistext), um den Anwender zu informieren.

```
ladebalken = ProgressDialog.show(NewsClickedActivity.this, "Bitte warten", "Nachricht wird geladen", true, false);
```

`onPostExecute()`

- Ressourcen des ProgressDialogs werden freigegeben

```
ladebalken.dismiss();
```

- TextViews 1 bis 3 werden mit den Inhalten (Strings) befüllt.

```
public void run() {
    tv1=(TextView)findViewById(R.id.tv_news_date);
    ...
    tv1.setText(s[2]);
}
```

- Der User kann den Inhalt der TextViews kopieren und in die Zwischenablage des Endgeräts speichern.

```
tv1.setTextIsSelectable(true);
```

- Im TextView 4(*Inhalt der News*) werden einige besondere Anweisungen benötigt:
 - Darstellung und Verfolgung von (Hyper-)links.

```
tv4.setOnClickListener(true);
tv4.setMovementMethod(LinkMovementMethod.getInstance());
```

- Verarbeitung des vom NewsResolver übergebenen HTML-Strings

```
tv4.setText(Html.fromHtml(s[3]));
```

dolnBackground()

- Neuinstanzierung eines NewsResolvers mit übergebener Url, um durch die Funktion getDetailsStringArray das Array "s" mit Daten zu füllen

```
s[0]=elements.get(0).child(1).text()+"\n";      //Ueberschrift
s[1]=elements.get(0).child(2).text()+"\n";      //Subhead
s[2]=elements.get(0).child(0).text();           //Zeit
s[3]=elements.get(0).child(0).outerHtml().replaceAll("<p>&
nbsp;*</p>", "");                           //Text
return s;
```

6 FRAGMENTE

6.1 NEWSFRAGMENT

Initialisierung erfolgt durch „NewInstance“, die aus der NewsActivity heraus gerufen wird. (siehe Funktionsaufruf -> onNavigationDrawerItemSelected) Durch das .commit wird diese neue Instanz der Klasse dann geladen.

Überschriebene Funktionen

OnAttach()

- Verknüpfung des NewsFragments mit der MainActivity

OnCreateView()

- Hier wird lediglich das Layout der View des Fragments geladen und angewendet

onActivityCreated()

- es wird geprüft, ob es SavedInstances (bereitsgeladene Inhalte) gibt
- wenn JA:
 - die Informationen werden aus dem ARRAYSPEICHER geholt und angewandt
- wenn NEIN:
 - die Funktion „zeigeNews“ wird gerufen, welche die aktuellsten News vom Server lädt

onStart()

- in dieser Funktion wird nur noch der Listener für den Aktualisierbutton mit der Schaltfläche verknüpft. Als onClick-Event wird dann lediglich die Funktion „zeigeNews“ gerufen.

standalone-Funktionen

`isOnline()`

- Diese Methode prüft durch einen Connectivity Manager, ob eine Verbindung zum Internet besteht

`zeigeNews()`

- Zunächst wird durch „`isOnline`“ geprüft, ob eine aktive Netzverbindung besteht
- wenn NEIN:
 - Es wird ein Hinweis an den Nutzer ausgegeben
- wenn JA:
 - Es wird eine Instanz der Klasse `NewsHelper` erstellt, welche dann als Hintergrund-Task ausgeführt wird, um ein Einfrieren der App zu verhindern.

6.2 KLASSE NEWSHELPER

- Klasse mit asynchroner Task-Ausführung
- Nach dem Aufruf der Methode *zeigeneNews* wird durch den Unterfunktionsaufruf *.execute* zunächst die Funktion *doInBackground* ausgeführt, wo eine neue Instanz des NewsResolvers erstellt wird.

```
private void zeigeNews() {
    NewsHelper nh = new NewsHelper();
    nh.execute();
}
```

- Durch *getTermineStringArray* des NewsResolvers wird ein String-Array zurückgegeben.
- Danach wird die Methode *onPostExecute* gerufen.

onListItemClick()

- Es wird ein intent verwendet um die *NewsClickedActivity* zu starten, als “Übergabeparameter“ wird *putExtra* verwendet, im Falle der App die URL zu dem angeklickten Event.

```
intent = new Intent(getActivity(), NewsClickedActivity.class)
;
intent.putExtra(TERMINDETAIL, newsResolver.getURLAsString(
    position));
startActivity(intent);
```

onPostExecute()

- Überprüfung ob das Fragment noch aktiv ist

```
if(getActivity() == null) return;
```

- einbinden des *CustomAdapterNews* in die ListView

```
setListAdapter(new CustomAdapterNews(getActivity(), ...));
```

6.3 NAVIGATIONDRAWERFRAGMENT

- Dieses Fragment bildet das Menü auf dem linken Rand der App ab und aktiviert den Button mit dem man in die Einstellungen gelangt.

`onCreate()`

- In dieser Methode werden die Einstellungen der Activity übernommen und der Drawer ausgewählt.
- Außerdem wird geprüft ob eine gesicherte Instanz vorhanden ist, aus der dann das zuletzt ausgewählte Fragment ermittelt wird und über die Funktion `selectItem()` aufgerufen wird.
- Falls keine gespeicherte Instanz existiert, wird die “0” (*NewsFragment*) als Standardwert übergeben.

`onActivityCreated()`

- Hier wird das Menü für das aktuelle Fragment aktiviert indem an `setHasOptionsMenu()` “true“ übergeben wird.

`onCreateView()`

- In dieser Funktion wird das Design für die Actionbar (als Listview) und die dazu gehörigen Menüpunkte festgelegt.
- Zudem wird hier der ClickListener initialisiert, der dann den ausgewählten Eintrag an `selectItem()` übergibt.

`isDrawerOpen()`

- Diese Methode prüft ob der Drawer bereits offen ist und liefert einen Wahrheitswert zurück.

`setUp()`

- Es werden hier folgende Einstellungen vorgenommen:
 - Einstellungen zum Design
 - Aktivierung des HomeButtons und dessen Animation beim Draufklicken
 - Zusammenführung der *ActionBar* und des *NavigationDrawers*
 - weitere Einstellungen zum Drawer
- Dann wird die Konfiguration in `mDrawerToggle` abgespeichert

`selectItem()`

- Hier wird die Animation auf den angeklickten Menüpunkt ausgeführt.
- Zudem wird der Drawer geschlossen und die Position des angeklickten Punktes an `mCallbacks.onNavigationDrawerItemSelected()` übergeben.

onAttach()

- Diese Methode setzt den Zeiger *mCallbacks* auf die Activity.

onDetach()

- Hier wird der Zeiger *mCallbacks* auf “**NULL**“ gesetzt.

onSaveInstanceState()

- Diese Funktion sichert die aktuelle Instanz.

onConfigurationChanged()

- Bei einer Änderung in den Einstellungen konfiguriert diese Methode *mDrawerToggle* um.

onCreateOptionsMenu()

- Diese Funktion legt das Design, aus einer XML-Datei für das Menü Einstellungen, fest.

onOptionsItemSelected()

- Hier wird der Button, über den man zu den Einstellungen gelangt, aktiviert und mit dessen Klasse verknüpft.

NavigationDrawerCallbacks()

- Hier wird die ausgewählte Menüpunkt-ID an die Activity übergeben.

6.4 NOTENFRAGMENT

onCreateView()

- Laden des entsprechenden XML-Layouts *fragment_noten.xml*
- Laden und Zuweisung der Schriftart des TextViews für die Überschrift des Fragments

onAttach()

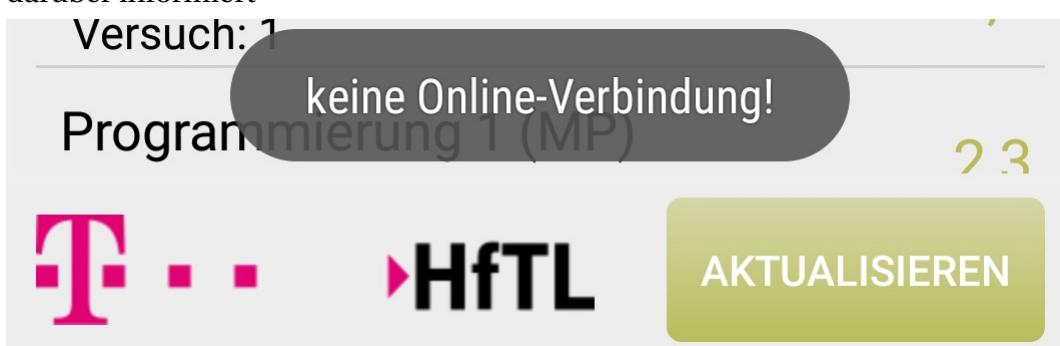
- Aufruf der Datenbank, in der die Noteneinträge abgelegt werden

onDetach()

- Schließen der Datenbank

onStart()

- Erstellen des Buttons zum Aktualisieren
- Mittels *OnClickListener* für den Button, wird über die Methode *testeBenutzerdaten()* überprüft, ob die Benutzerdaten für QiS eingetragen wurden. Andernfalls erfolgt die Ausgabe mittels der Methode *keineBenutzerdaten()*, dass diese nicht eingetragen wurden.
- Sofern die Benutzerdaten eingetragen wurden, und eine Onlineverbindung zu QiS besteht, werden die Noten erneut abgerufen.
- Ist keine Verbindung zu QiS vorhanden, wird der Nutzer mittels Toast-Benachrichtigung darüber informiert



getNoten()

- Liest in der Notendatenbank alle Werte in der Spalte **NOTENABFRAGE** aus und schreibt diese in das String-Array "s", welches auch übergeben wird.

getSemester()

- Liest in der Notendatenbank alle Werte in der Spalte **SEMESTERABFRAGE** aus und schreibt diese in ein String-Array "s", welches auch übergeben wird.

getFach()

- Liest in der Notendatenbank alle Werte in der Spalte **FACHABFRAGE** aus und schreibt diese in das String-Array “s“, welches auch übergeben wird.

getVersuche()

- Liest in der Notendatenbank alle Werte in der Spalte **VERSUCHABFRAGE** aus und schreibt diese in das String-Array “s“, welches auch übergeben wird.

Hinweis:

Die String-Arrays sind nur lokal in den jeweiligen Methoden definiert, weshalb diese - der Einfachheit halber - alle den Namen “s“ erhielten. Tatsächlich werden hier vier unterschiedliche String-Arrays befüllt.

setzeListview()

- Hier werden zunächst entsprechende String-Arrays durch die jeweiligen Methoden zur Abfrage in der Datenbank befüllt:

```
notenList = getNoten();
semesterList = getSemester();
fachList = getFach();
versuchList = getVersuche();
```

- Die Strings stehen entsprechend der alphabetischen Reihenfolge der Semesterbeschreibung in den Arrays.
- Bsp: SoSe 12, SoSe 13, WiSe 11/12, WiSe 12/13, WiSe 13/14
- Um die Übersichtlichkeit zu wahren, werden die Werte in den Arrays nun sortiert, wobei die Werte vom aktuellsten Semester am Anfang stehen und anschließend chronologisch geordnet werden.
- Um dies zu realisieren wird zunächst mit Hilfe eines regulären Ausdrucks ein Suchpattern festgelegt, nach dem die zu sortierenden Elemente gefunden werden.
- Entscheidend für die Sortierung sind die letzten beiden Ziffern für die Jahreszahl.

```
Pattern p = Pattern.compile("\\d{2}");
```

- Anschließend wird der Einfachheit halber ein Bubblesort verwendet (kann ggf. geändert werden), der durch das String-Array **semesterList** geht und bei benachbarten Einträgen den Suchpattern anwendet.

```
Matcher m = p.matcher(semesterList[j]);
Matcher n = p.matcher(semesterList[j+1]);
```

- Sofern die Pattern in den beiden Einträgen vorhanden sind (das ist eine zwingende Bedingung), werden diese jeweils in Hilfsvariablen geschrieben.

```

if((m.find())&&(n.find())){
    int x = Integer.parseInt(m.group());
    int y = Integer.parseInt(n.group());
    ...
}

```

- Danach werden diese Hilfsvariablen verglichen und ggf sortiert.
- Sofern die Einträge in (`semesterList`) sortiert werden, müssen auch die zugehörigen Werte in den anderen Arrays (`notenList`, `fachList`, (`versuchList`) entsprechend sortiert werden.

```

if(x<y){
    temps = semesterList[j];
    semesterList[j] = semesterList[j+1];
    semesterList[j+1] = temps;
    temps = notenList[j];
    notenList[j] = notenList[j+1];
    notenList[j+1] = temps;
    temps = fachList[j];
    ...
}

```

- Sobald die Sortierung erfolgt ist, müssen noch redundante Werte im Array (`semesterList`) gefunden und durch (`NULL`) ersetzt werden.
 - Dadurch wird eine Auflistung der Noten, Fächer und Versuche, aufgeschlüsselt nach Semester realisiert, ohne dass über jedem Eintrag die Semesterbeschreibung steht. (s. `CustomAdapterNoten.java`)

```

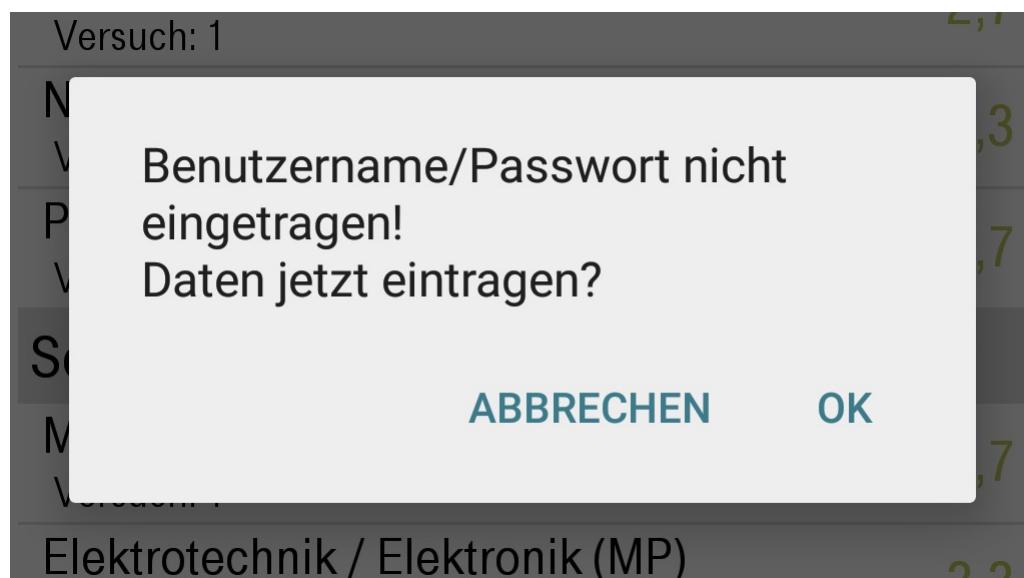
String[] vgl = new String[1];
vgl[0] = "";
for(int k=0; k<semesterList.length; k++){
    if (semesterList[k].equals(vgl[0])) {
        vgl[0] = semesterList[k];
        semesterList[k] = null;
    }
    else{
        vgl[0] = semesterList[k];
    }
}

```

- Anschließend werden diese String-Arrays an den `CustomAdapterNoten` übergeben. Damit wird eine individuelle Befüllung und Formatierung der Liste mit den ausgelesenen Werten aus der Datenbank realisiert.

keineBenutzerdaten()

- Diese Methode prüft, ob in den Einstellungen Login und Passwort für QiS einge tragen wurden. Andernfalls wird eine Fehlermeldung ausgegeben:



– NotenHelper (Class)

6.5 STUNDENPLANFRAGMENT

newInstance()

- Erstellt ein StundenplanFragment und “steckt“ die aktive Position (aus dem Navigation Drawer) in das “Bundle args“ welches als Argument im Fragment übergeben wird.

onCreateView()

- Laden des entsprechenden XML-Layouts fragment_noten.xml
- Laden und Zuweisung der Schriftart für das TextView für die Überschrift des Fragments.

onViewCreated()

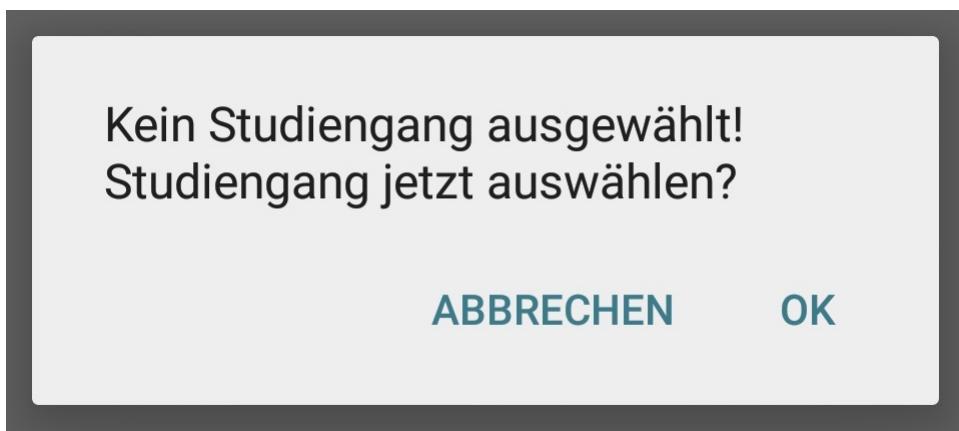
- Falls im “Stundenplanspeicher“ Daten vorhanden sind, werden diese geladen.
- Methode für das Dropdownmenü wird gerufen und Array für “events“ erstellt.
- Anonyme Listener für die Buttons werden erstellt.

erzeugeDropdown()

- Dropdown aus xml einem Objekt zuweisen.
- Listener für Dropdown (als anonymer Listener) wird erzeugt und registriert.
- Beim registrieren wird die Methode *onItemSelected* aufgerufen und ein StundenplanHelper ausgeführt.
- Mittels eines “Calendar“, “Date“ und zwei “SimpleDateFormat“ wird das Dropdownmenü befüllt, indem die Daten in eine String-List eingefügt werden (*list.add(temp)*).
- Dropdown wird mit Liste verknüpft.

keinStudiengang()

- Prüft ob in den Einstellungen der Studiengang eingetragen wurde, ansonsten wird eine Fehlermeldung ausgegeben:



erstelleStundenplan()

- Erzeugt die Ausgabe des Stundenplans und fügt sie in den ListView ein.
- Falls keine Daten vorhanden sind, wird “keine Daten“ ausgegeben.
 - StundenplanHelper (class)

onPreExecute()

- erzeugt einen Ladebalken.

onPostExecute()

- falls das Fragment noch aktiv ist wird die Methode erstelleStundenplan() gerufen.
- Ladebalken wird entfernt.

doInBackground

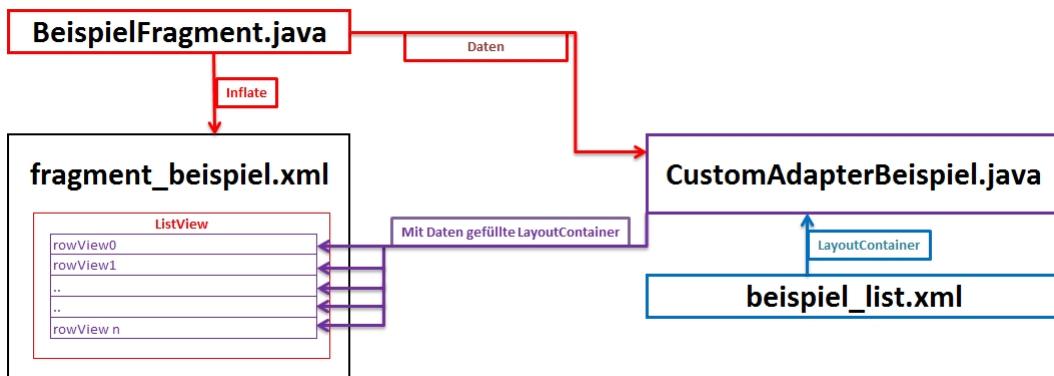
- Die Methode erzeugt einen *StundenplanResolver* und befüllt das “StundenplanEvents-Array *Events*“ mit Daten durch die Methode *erzeugeStundenplan(String Woche)* des *StundenplanResolvers*.
- Falls ein Fehler auftritt, wird ein Event erstellt, in dem keine Daten sind.

7 CUSTOMADAPTER

7.1 ALLGEMEIN

Die CustomAdapter kommen da zum Einsatz, wo eine ListView genutzt und individuell gefüllt werden muss.

- 1 Fragment.java wird mittels inflate der fragment.xml zugeordnet
- 2 Fragment.java über gibt die zuvor ermittelten Daten (mit Strings gefüllte Arrays) an den CustomAdapter
- 3 Entsprechend der Länge x der Arrays wird x-mal die zugehörige _list.xml als LayoutContainer aufgerufen und ihre TextViews mit den Inhalten der Arrays gefüllt.
- 4 Die gefüllten LayoutContainer werden zeilenweise an die ListView in der fragment.xml übergeben.



class Holder

Dies ist ein Container mit TextViews

```
TextView tv_name;
```

Dieser Container wird dann als jeweilige Zeile in der ListView der zugehörigen Activity dargestellt.

Die Grundlage der Ausgabe bilden spezielle XML-Dateien. Diese dienen definieren die Output jeder einzelnen Zeile der ListViews. Hier sind Name, Position, Dimensionen, Farben, **Schriftarten**, Schriftgröße etc festlegt.

Für jeden Customadapter gibt es eine entsprechende _list.xml:

- news_list.xml
- noten_list.xml
- stundenplan_list.xml

public View getView()

erstellt einen neuen Container:

```
Holder holder = new Holder;
```

erstellt eine View rowView:

```
View rowView;
```

Das Layout der View und damit jeder Zeile in der ListView wird durch das Laden der entsprechenden XML namelist formatiert:

```
rowView = inflater.inflate(R.layout.name_list, null);
```

Die TextViews des Containers werden den entsprechenden TextViews in der XML zugeordnet:

```
holder.tv_name=(TextView)
rowView.findViewById(R.id.namelist_name.xml);
```

Das TextView wird mit dem entsprechenden Wert des zugehörigen String-Arrays befüllt:

```
holder.tv_name.setText(name[position]);
```

Der nun befüllte Container wird nun als rowView übergeben:

```
return rowView;
```

7.2 CUSTOMADAPTERNEWS.JAVA

Dies ist der simpelste CustomAdapter der HfTL-App. Er wird verwendet um die News-Liste (also die Übersicht der News aus der HfTL-Homepage) zu formatieren.

Dabei werden drei String-Arrays (date, headline, content) in das zugehörige TextView der rowView übergeben. Der Inhalt dieser Arrays wird mittels der Klasse NewsHelper, die wiederum die Methoden der Klasse NewsResolver aufruft, befüllt.

7.3 CUSTOMADAPTERNOTEN.JAVA

Dieser CustomAdapter wird für die Befüllung der ListView des Notenfragments benutzt.

Hier werden vier String-Arrays (**subject**, **trys**, **mark**, **semester**) mit den Daten aus der Notendatenbank befüllt.

Entsprechend des Inhalts, werden die zugehörigen TextViews noch gesondert formatiert.

Ist der Inhalt an der Position des String-Arrays “**NULL**”, wird das zugehörige TextView auf “**GONE**” gesetzt. So wird realisiert, dass mehrere Fächer unter dem selben Semester gelistet sind, ohne dass ein leeres (dunkelgraues) TextView erscheint.

```
if (semester[position]==null){
    holder.tv_semester.setVisibility(TextView.GONE);
}
```

Auch für die jeweiligen Noten gibt es eine gesonderte Formatierung:

- Note schlechter als 5.0: **magenta**
- Note schlechter als 3,4: **gelb**
- Note besser als 3,5: **grün**

```
if (mark[position].equals("5,0"))
    holder.tv_mark.setTextColor
        (context.getResources().getColor(R.color.magenta));
else if (mark[position].equals("4,0" ) |
         mark[position].equals("3,9" ) |
         mark[position].equals("3,8" ) |
         mark[position].equals("3,7" ) |
         mark[position].equals("3,6" ) |
         mark[position].equals("3,5" ) )
    holder.tv_mark.setTextColor
        (context.getResources().getColor(R.color.gelb));
else
    holder.tv_mark.setTextColor
        (context.getResources().getColor(R.color.gruen));
```

7.4 CUSTOMADAPTERSTUNDENPLAN

Dieser CustomAdapter befüllt die ListView des StundenPlanfragments.

Hier werden fünf StringArrays(**datum,fach,zeit,raum,kategorie**) übergeben, die zuvor mittels der Methode *erstelleStundenplan()* des StundenplanFragments aus dem HTML-Code der QiS/HiS-Seite ausgelesen wurden.

Auch hier gibt es einige spezifische Formatierungen, abhängig vom übergebenen Inhalt.

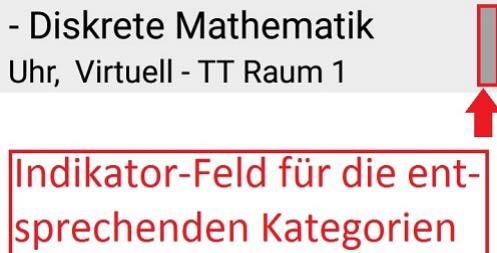
Ist der Inhalt des StringArrays **datum** an einer Stelle “**NULL**“, so wird die Sichtbarkeit des zugehörigen TextViews auf “**GONE**“ gesetzt.

Analog zum **CustomAdapterNoten** werden so die einzelnen Fächer unter dem selben Datum gelistet. Andernfalls würde über jedem Fach ein dunkelgraues TextView stehen.

```
if (datum[position] != null)
    holder.tv_date.setText(datum[position]);
else
    holder.tv_date.setVisibility(TextView.GONE);
```

Das StringArray **kategorie** wird, neben der reinen Ausgabe dazu verwendet, wichtige Ereignisse farblich kenntlich zu machen. Dabei wird der Inhalt an der jeweiligen Position des StringArrays geprüft und eine entsprechende (CI/CD-)Farbe für das Rechteck auf der rechten Seite gesetzt:

- Prüfung: **magenta**
- Praktikum: **dunkelblau**
- Rest: **grau01**



```
if (kategorie[position].equals("Pruefung"))
    holder.tv_category.setBackgroundColor
    (context.getResources().getColor(R.color.magenta));
else if ((kategorie[position].equals("Praktikum"))
    holder.tv_category.setBackgroundColor
    (context.getResources().getColor(R.color.dunkelblau));
else
    holder.tv_category.setBackgroundColor
    (context.getResources().getColor(R.color.grau01));
```

8 XML-DATEIEN

8.1 FRAGMENT.XML

Die fragment.xml bilden das visuelle Rückgrat der App. Hier wird das Layout festgelegt und damit wo welcher Inhalt steht bzw. zu stehen hat. Statische Inhalte und Formatierungen(Farben, Dimensionen, Elementposition, Schriftarten etc.) werden bereits hier festgelegt.

Variable Formatierungen und Inhalte werden entweder direkt durch die zugehörige *Fragment.java* oder - wenn sie die ListViews betreffen - durch den zugehörigen CustomAdapter befüllt.

Auf den detaillierten Aufbau wird nicht näher eingegangen, da dieser mit einigen XML-Kenntnissen selbsterklärend sein dürfte.

8.2 _LIST.XML

Hier gibt es derzeit drei XML-Dateien:

- news_list.xml
- noten_list.xml
- stundenplan_list.xml

Diese dienen als LayoutContainer für die ListViews. Sie werden durch den entsprechenden CustomAdapter mit den Daten aus der Fragment.java befüllt und zeilenweise an das zugehörige ListView der Fragment.xml übergeben.

8.3 ACTIVITY_NEWS_CLICKED.XML

Dient als Designvorlage für eine aufgerufene News aus der Newsübersicht. Der Inhalt wird durch die Klasse [NewsClickedActivity.java](#) befüllt.

8.4 SETTINGS_TOOLBAR

Definiert die Kopfzeile der einstellung.xml.

8.5 IMPRESSUM.XML & ACTIVITY_IMPRESSUM

Hier steht das Impressum. Während das Layout in der activity_impresum.xml festgelegt wird, wird der Inhalt (*Ähnlich wie bei strings.xml*)zentral abgelegt.

8.6 COLORS.XML

Um den Aufruf der Farben nach CI/CD zu vereinfachen, wurden die RGB-Werte (in hexadezimal) zentral in der Datei main\res\values\colors.xml gespeichert.

```
<item name="magenta" type="color">#E20074</item>
<item name="grau01" type="color">#A4A4A4</item>
...
<integer-array name="telekomcolors">
    <item>@color/magenta</item>
    <item>@color/grau01</item>
    ...
</integer-array>
```

Die Farben können dann sowohl mittels JAVA als auch mittels XML aufgerufen werden:

- XML:

```
    android:color="@color/magenta"
```

- JAVA:

```
getResources().getColor(R.color.magenta)
```

8.7 ARRAY.XML

Hier stehen alle aktuellen Studiengänge mit Matrikel und ihre zugehörigen Links zu QiS.

Ein Studiengang muss in den Einstellungen ausgewählt werden, damit der Stundenplan angezeigt werden kann.

Ferner sind hier die Werte für das Intervall der Notenabfrage abgelegt.

Die Pflege der Datei erfolgt derzeit manuell. (s. [Pflege der array.xml](#))

8.8 EINSTELLUNG.XML

Layout-Definition des Einstellungsmenüs.

8.9 STRINGS.XML

Statische Strings, wie sie beispielsweise für die Fragmentüberschriften verwendet werden, werden nicht in den XML-Tag geschrieben.

```
<TextView
    ...
    android:text="News aus der HFT-Leipzig"
    ...>
```

Vielmehr werden diese zentral in der *strings.xml* abgelegt.

Listing 4: Ablage in strings.xml

```
<string name="News_headline">News aus der HFT-Leipzig</string>
```

Anschließend werden diese Strings dort abgerufen, wo sie benötigt werden.

Listing 5: Aufruf im fragment.xml

```
<TextView  
...  
android:text="@string/News_headline"  
.../>
```

9 LAYOUT

9.1 ALLGEMEINES

Das Layout wird zum Großteil über die **XML-Dateien** realisiert. Um die Handhabung zu realisieren, wurden einige Elemente bzw. Bezeichnungen standardisiert.

Im Folgenden werden die wichtigsten Details rund um die Darstellung des Layouts erläutert. Mit einigen XML-Kenntnissen sollte der Großteil selbsterklärend sein.

9.2 SCHRIFTARTEN

Da die Standardbibliothek von Android nur sehr wenige Schriftarten liefert, mussten die Schriftarten nach CI/CD nachträglich eingefügt werden. Diese befinden sich unter main\assets\fonts. Der Aufruf der Schriftarten erfolgt im Allgemeinen über folgende JAVA-Anweisung:

Listing 6: Beispiel: Einbinden der Schriftart Ocra im Notenfragment

```
View notenView = inflater.inflate(R.layout.fragment_noten,
    container, false);

TextView hl = (TextView) notenView.findViewById(R.id.hl_Noten);
Typeface headline = Typeface.createFromAsset(getActivity().
    getAssets(), "fonts/OCRA.TTF");
hl.setTypeface(headline);

return notenView;
```

In der Praxis hat sich jedoch herausgestellt, dass diese Art und Weise, Schriftarten einzubinden auf zwei Probleme trifft:

- Über das Fragment können die Schriftarten nicht den ListViews zugewiesen werden, da bei Erstellung der Fragmente die ListViews bzw. deren Inhalt (die durch die jeweiligen CustomAdapter befüllt werden) noch nicht existieren.
- Es wäre möglich die Schriftarten im jeweiligen CustomAdapter zuzuweisen. Je- doch hat die Praxis gezeigt, dass dadurch der Speicherverbrauch extrem ansteigt (200 bis 300 Prozent) und damit die Performance der App merkbar sinkt. Der Grund hierfür liegt daran, dass bei jedem TextView, das angezeigt wird, die Schrift-arten aus den Assets geladen und im Cache abgelegt werden.

Das Problem wurde durch einige Anpassungen umgangen:

- Erstellen der Klasse FontCache
 - Diese Klasse prüft ob die geforderten Schriftarten bereits im Cache hinterlegt sind.

- Sollten die Schriftarten bereits im Cache hinterlegt sein, dann werden diese genommen. Andernfalls werden sie aus den Assets geladen und im Cache abgelegt.
- Einbinden der Schriftarten in jeweilige Klassen
 - Die Klassen werden direkt durch die TextViews in den XML-Dateien aufgerufen.
 - Beim Aufruf wird die o.g. Klasse *FontCache* aufgerufen.
- Setzen der Schriftarten in den XML-Dateien
 - Um die benötigten Schriftarten zu laden, werden die XML-Tags geändert.
 - * Ursprüngliche (native) XML-Tags:

Listing 7: Normales TextView

```
<TextView  
    android:id="@+id/newslist_date"  
    android:layout_width="wrap_content"  
    android:layout_height="wrap_content"  
    ... />
```

* Pfadangabe zur jeweiligen Schriftart als neues XML-Tag:

Listing 8: Einbinden von TeleGrotNorm

```
<bkmi.de.hftl_app.help.Typefaces.TeleGrotNorm  
    android:id="@+id/newslist_date"  
    android:layout_width="wrap_content"  
    android:layout_height="wrap_content"  
    ... />
```

9.3 BUTTONS

Grundlegend werden vier Standardbuttons in der App verwendet. Sie alle wurden über eine eigene XML-Datei definiert und durch die XML-Dateien der jeweiligen Fragmente aufgerufen. Die Methoden, die bei Betätigung der Buttons aufgerufen werden, befinden sich in der JAVA-Datei des jeweiligen Fragments.

Durch die Auslagerung der Definitionen für die Buttons ist es u.a. möglich, Farbverläufe, abgerundete Ecken und interaktives Verhalten (bspw. Farbänderungen bei Betätigung) festzulegen. Im Grunde genommen wird nur ein *Item* generiert, das als Vorlage für einen Button herhalten muss.

Exemplarischer Button

- Es gibt drei Zustände:

- gedrückt

```
<item android:state_pressed="true"> ... </item>
```

- fokussiert

```
<item android:state_focused="true"> ... </item>
```

- standard

```
<item> ... </item>
```

- Der Status *focused* wird derzeit nur als Dummy angesehen, da er derzeit nicht verwendet wird.
- Die Definitionen innerhalb der Items findet jeweils zwischen diesen Tags statt:

```
<item>
    <shape> ... </shape>
</item>
```

- Farbverlauf mit Start- und Endfarbe, sowie dem Winkel des Farbverlaufs.

```
<gradient
    android:startColor="#77E20074"
    android:endColor="#E20074"
    android:angle="270"/>
```

- Umrandung

```
<stroke
    android:width="3dp"
    android:color="@color/hellblau"/>
```

- Rundung der Ecken

```
<corners  
    android:radius="5dp" />
```

Folgende Standardbuttons werden benutzt:

custom_button

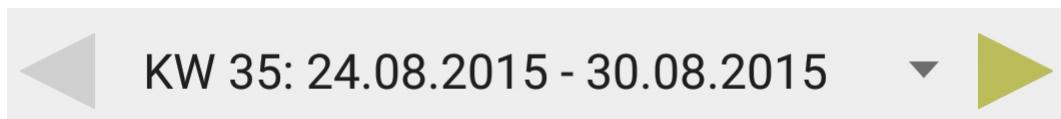
- Dies ist der Standard-Button.

custom_newsclick

- Hier werden die News-Einträge der News-Übersicht als Button verwendet, um ein optisches Feedback zu erhalten, wenn eine News ausgewählt wurde.

custom_vorbutton & custom_zurbutton

- Diese beiden Buttons werden beim Stundenplan verwendet. (Grüne Dreiecke um zwischen den Wochen vor- und zurückzublättern.)



- als Vorlage dienen jpg-Images
- Je nach Status werden die entsprechenden Images geladen:
 - inaktiv: grau
 - aktiv: grün
 - gedrückt (optisches Feedback): magenta

10 ERWEITERUNGEN UND VERBESSERUNGEN FÜR KOMMENDE VERSIONEN

Da es aufgrund der Projektrahmenbedingungen nicht möglich gewesen ist, alle gewünschten Features zu implementieren, folgt hier eine Auflistung weiterer Funktionen oder Verbesserungen, die für die App denkbar wären.

10.1 PFLEGE DER ARRAY.XML

Die Datei array.xml dient als Grundlage für das Einstellungsmenü.

Hier werden alle Studiengänge, mit den entsprechenden Matrikeln aufgelistet. Das Kernproblem ist, dass diese Liste manuell auf dem aktuellen Stand gehalten werden muss.

Denkbar wäre hier eine Methode, die zu gewissen Zeitpunkten (*Ende der Semesterferien*) die neuen Matrikel hinzufügt. Problematisch dabei sind jedoch mehrere Aspekte, die beachtet werden müssen:

- Festlegung der Zeitpunkte zur Aktualisierung der Daten
- Wegfall der alten Matrikel (Vermeidung von Datenmüll)
- Umbenennung von Studiengängen
- Hinzufügen neuer Studiengänge
- Wegfall alter Studiengänge

Die Problemstellung ist damit für das Projekt zu komplex und kann wahrscheinlich besser in Absprache und Zusammenarbeit mit dem Hochschul- und Prüfungsamt der HfTL gelöst werden.

10.2 RAUMPLAN

Ein Kernfeature, das aber von der Komplexität her wohl ein eigenes Projektteam beschäftigen kann, ist die Raumplanung.

Hierbei soll es dem Nutzer bspw. möglich sein, freie Räume für selbst organisiertes Lernen zu finden.

Wichtig dabei ist auch zu wissen, wann welcher Raum durch wen belegt ist. Insofern müsste es abseits von der zentralen Raumplanung durch die HfTL möglich sein, dass der Nutzer sich einen freien Raum buchen kann und dieser Buchungswunsch dann auch für andere Anwesende zu sehen ist.

Idealerweise gäbe es auch einen Lageplan, wo welcher Raum zu finden ist.

10.3 PORTIERUNG AUF ANDERE SYSTEME

Die hier beschriebene App ist eine reine Android-App. Sofern es eine Nachfrage gibt, wäre eine Portierung auf andere Betriebssysteme (iOS, Windows Mobile, Ubuntu Touch und weitere) sicherlich sinnvoll.

10.4 SPEISEPLAN

Eine Funktion zur Einsicht aktueller Kantinenangebote der HfTL und HTWK.

10.5 LOGIN UND FUNKTIONEN FÜR LEHRKRÄFTE

Eine Erweiterung des Funktionsumfangs der App für die Lehrkräfte der HfTL wäre denkbar, müsste aber noch näher spezifiziert werden.

10.6 UNTERSTÜTZUNG VON TABLETTS

Die App ist gegenwärtig nur für Displaygrößen von Smartphones konzipiert. Eine Darstellung auf Android-Tablets ist möglich, wurde jedoch nicht getestet und müsste gegebenenfalls in späteren Versionen mit bedacht werden.

10.7 LANDSCAPE-MODUS

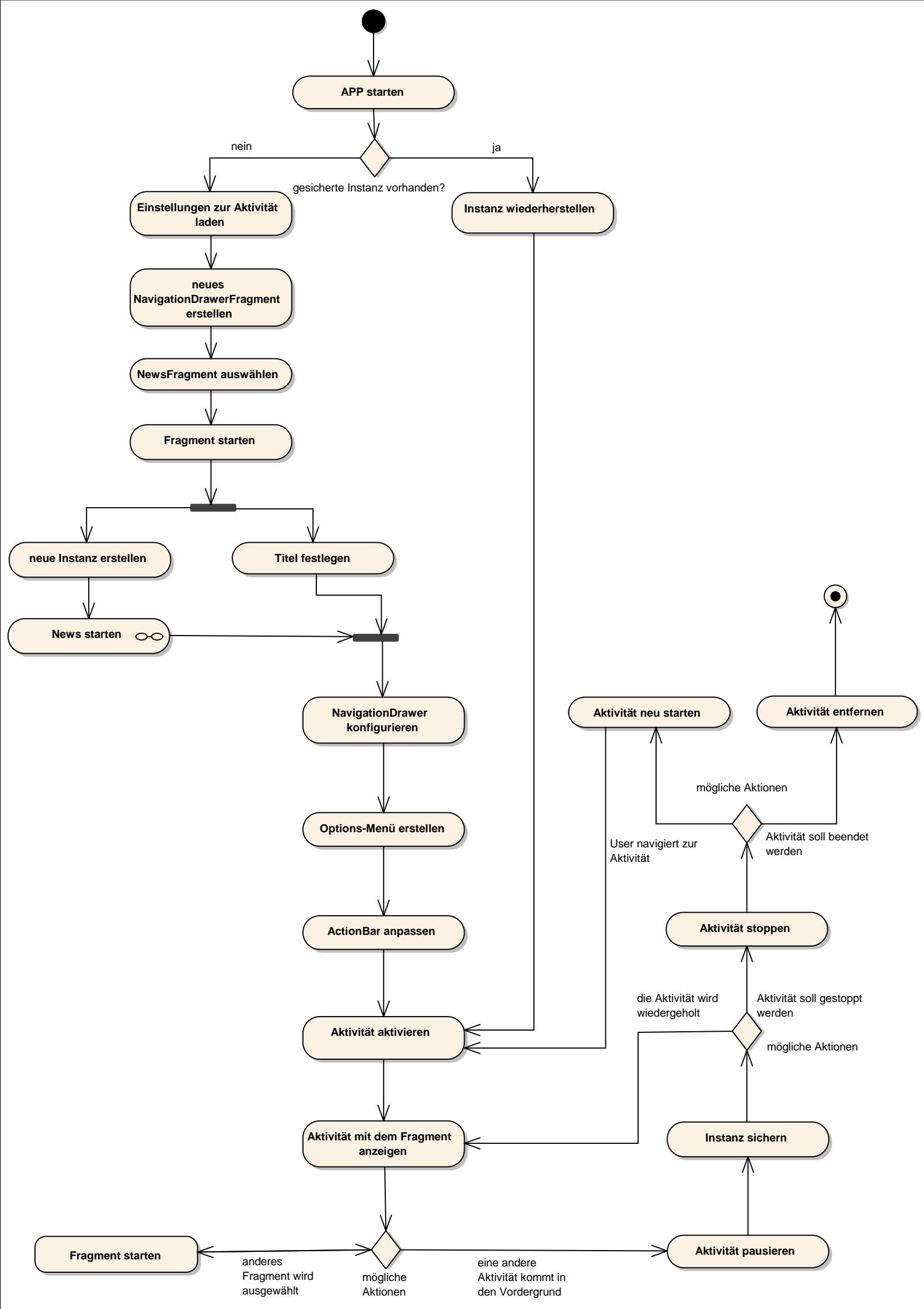
Der Landscape-Modus wird derzeit nicht unterstützt.

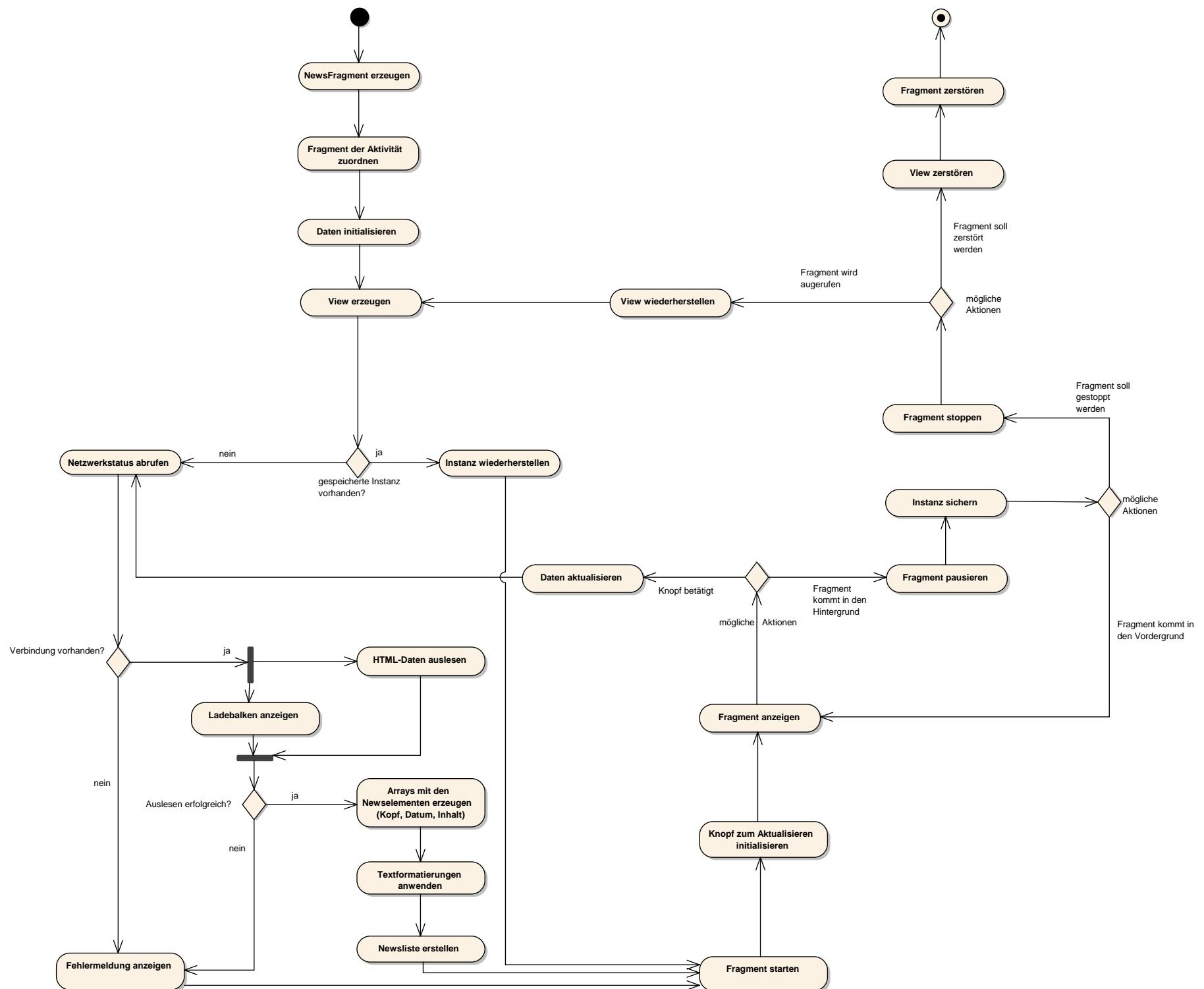
10.8 BEKANNTES FEHLER

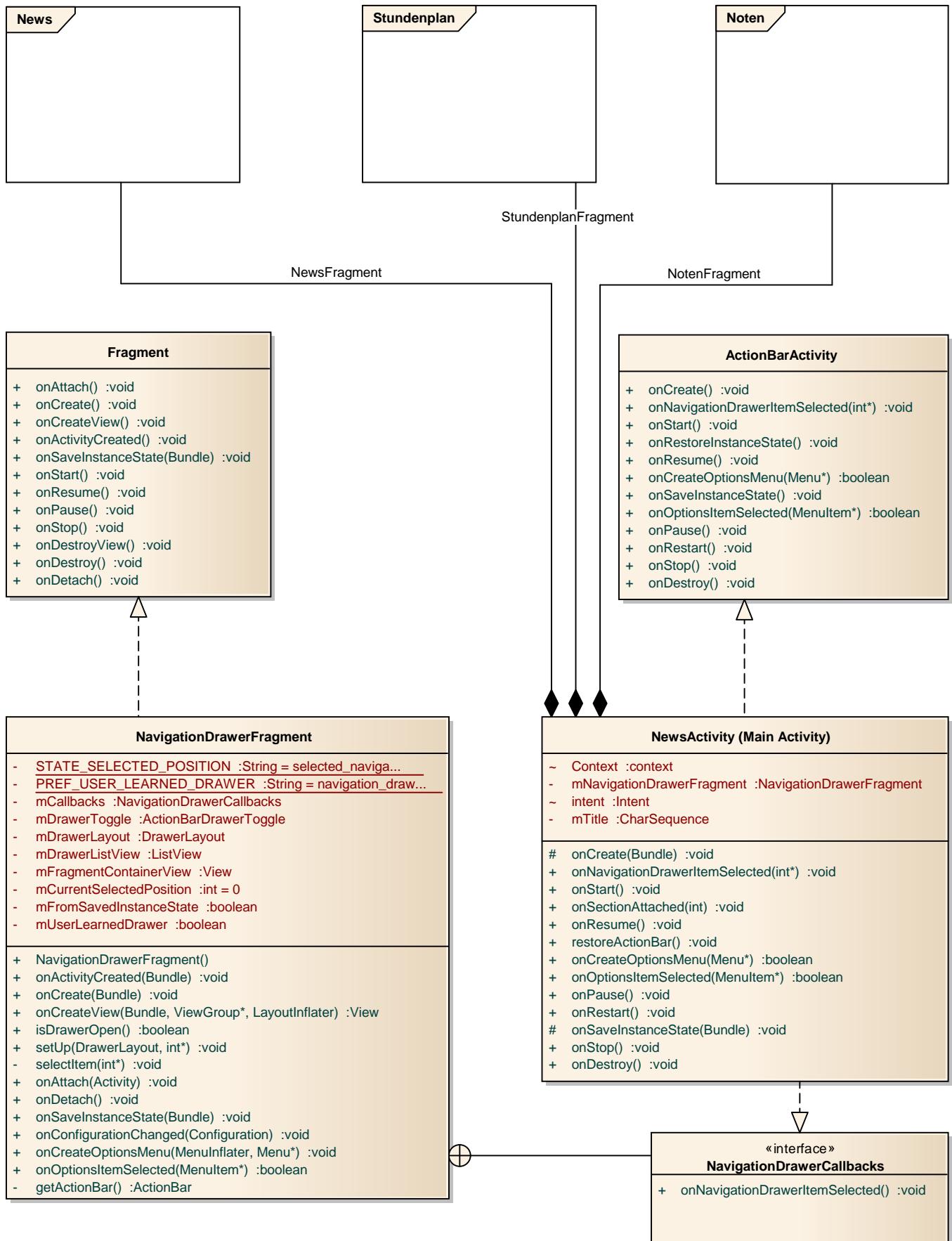
Es gibt einige Bugs bzw. Fehler die in zukünftigen Versionen behoben werden müssen:

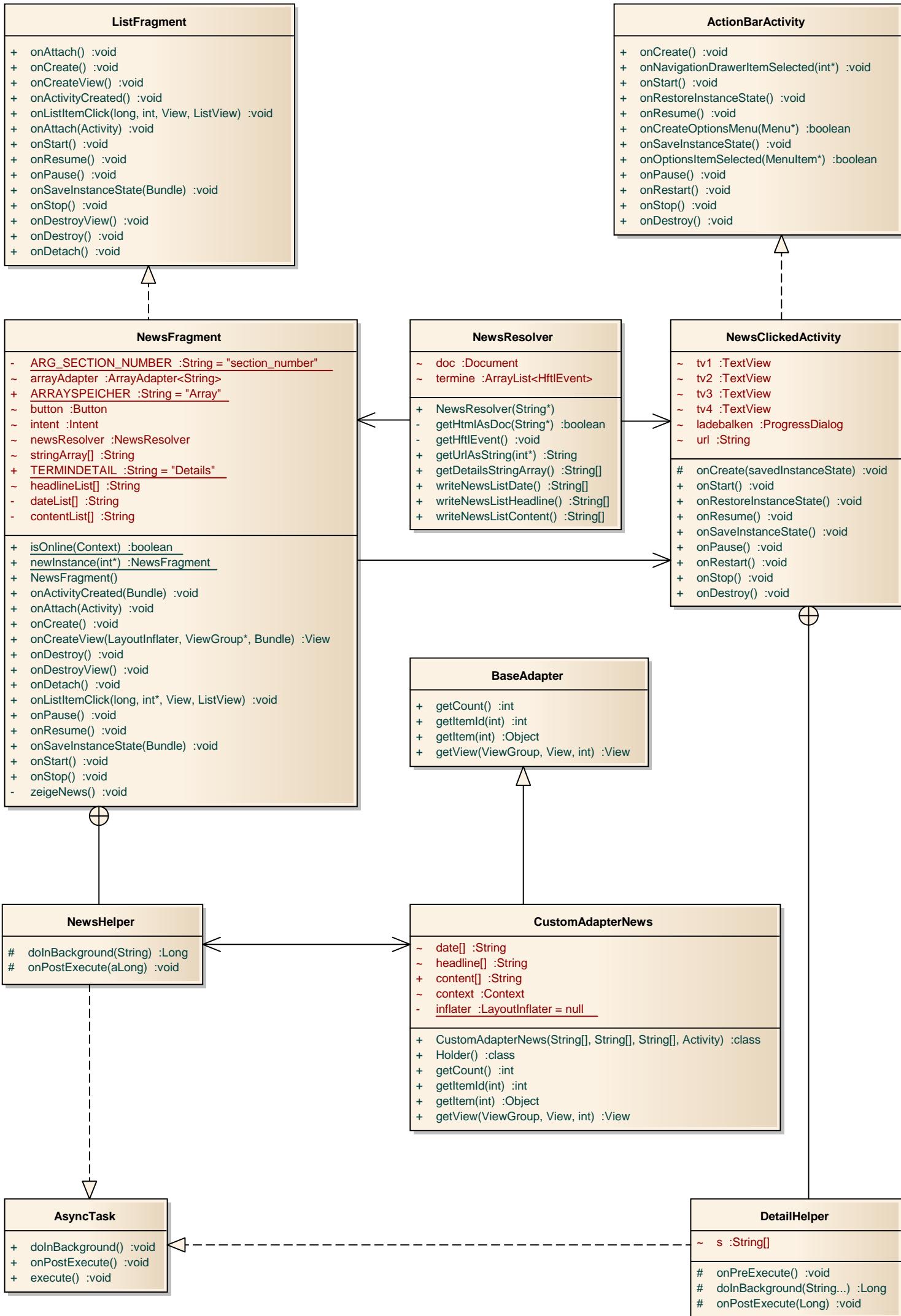
- Ist kein Email-Cient auf dem Smartphone installiert, kann womöglich die App abstürzen, wenn in den News ein Mail-Link aufgerufen wird.
- Sind in den News auf der Homepage der HfTL Links nicht absolut angegeben (*bspw. *), dann stürzt die App ab.

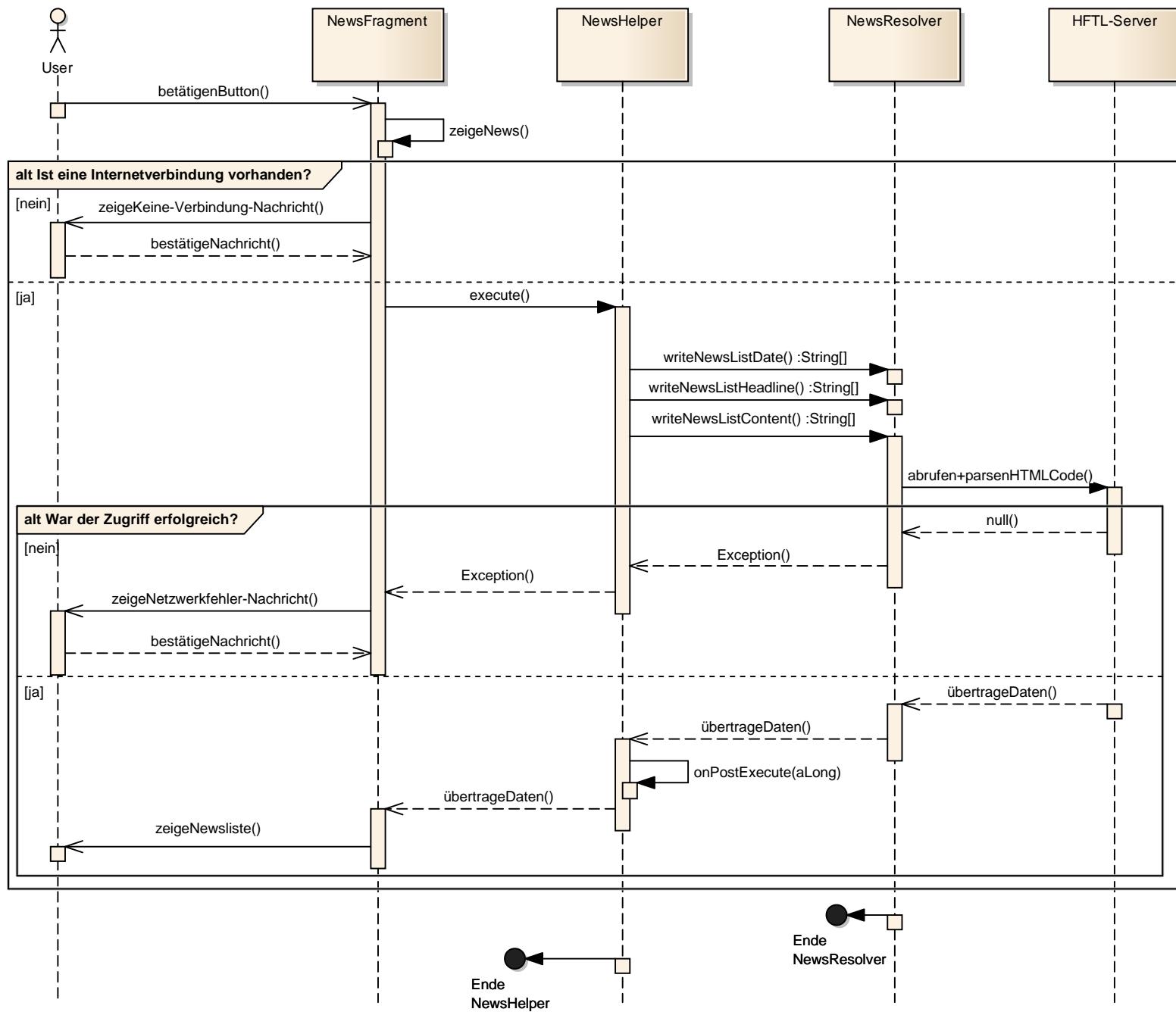
11 UML

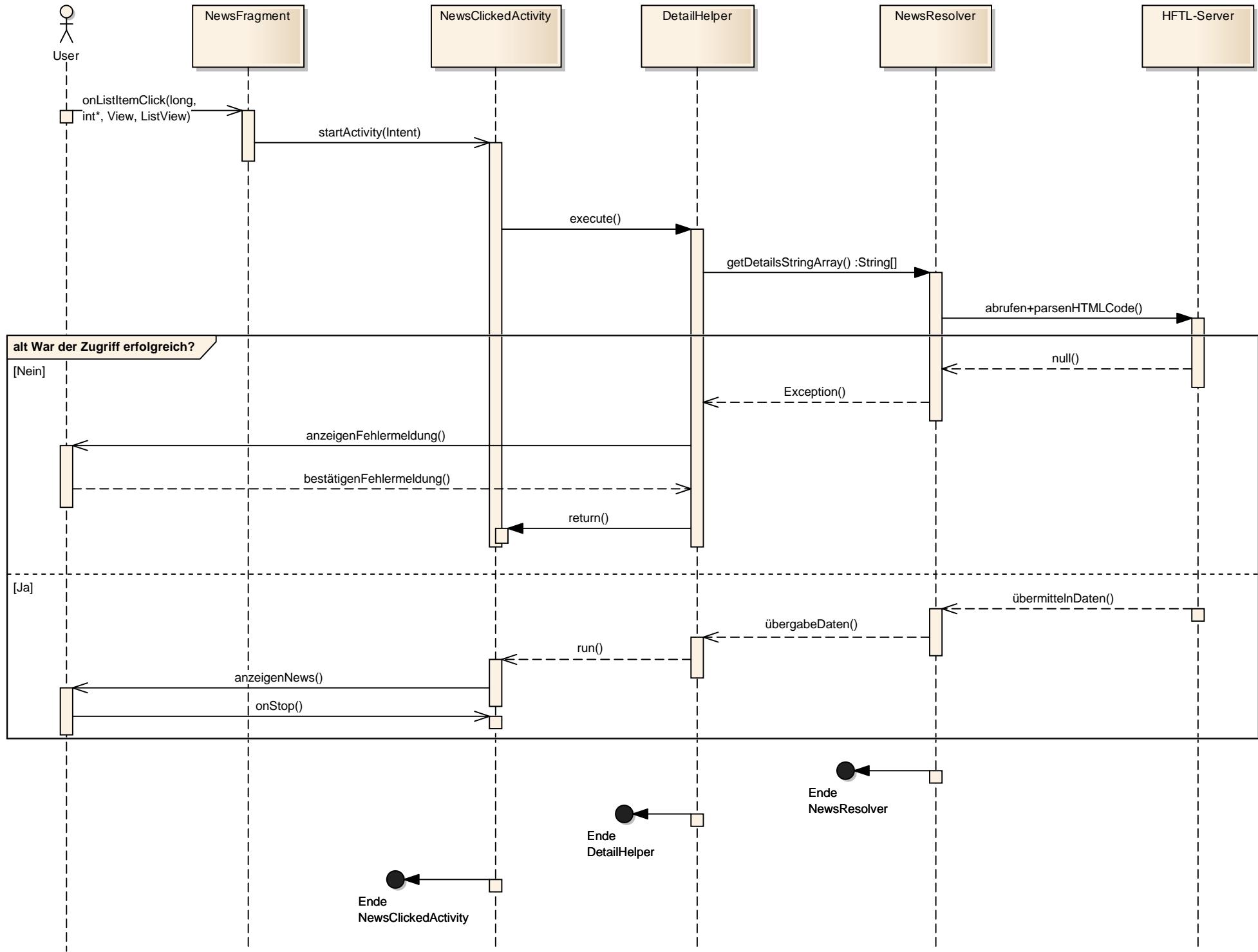


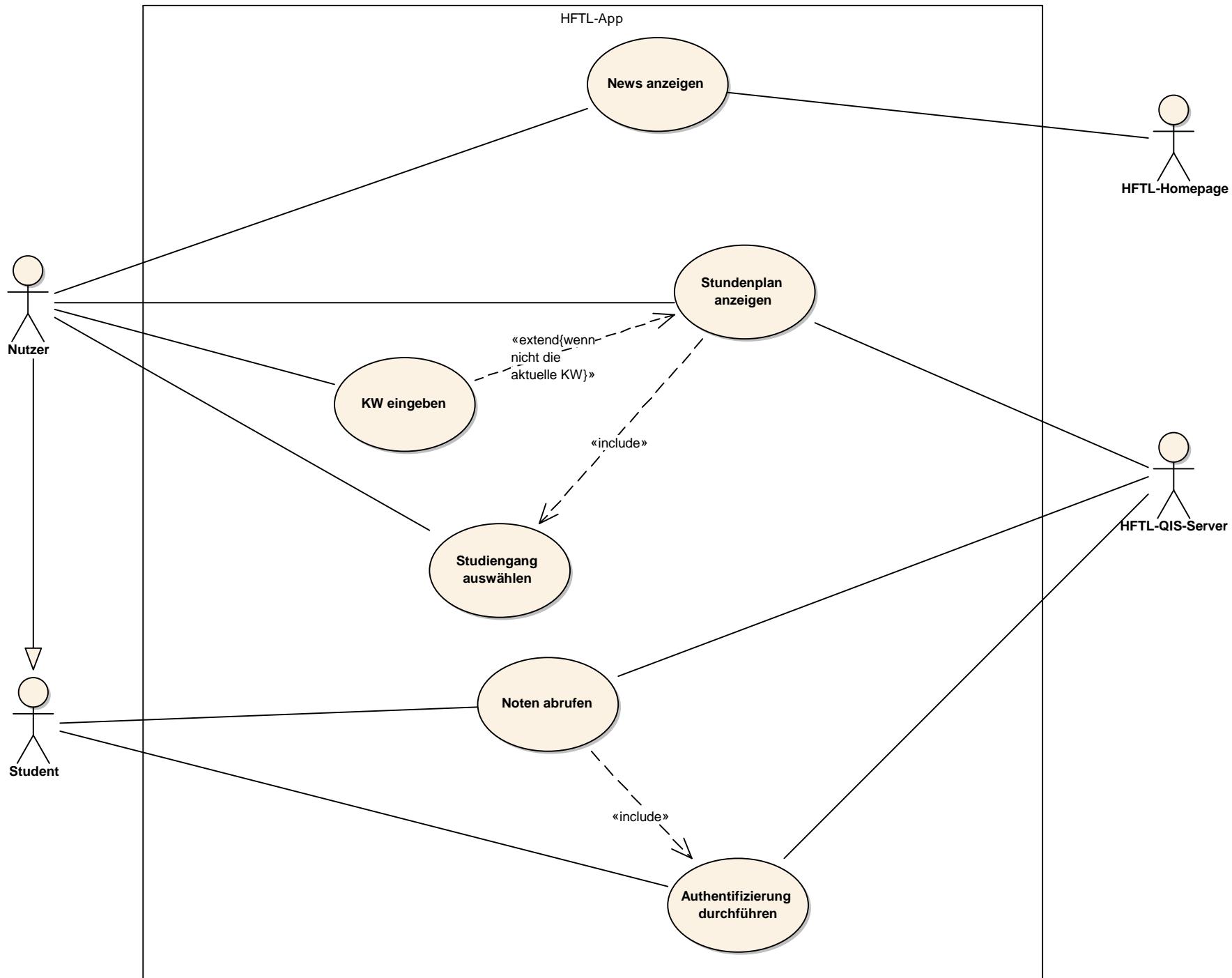


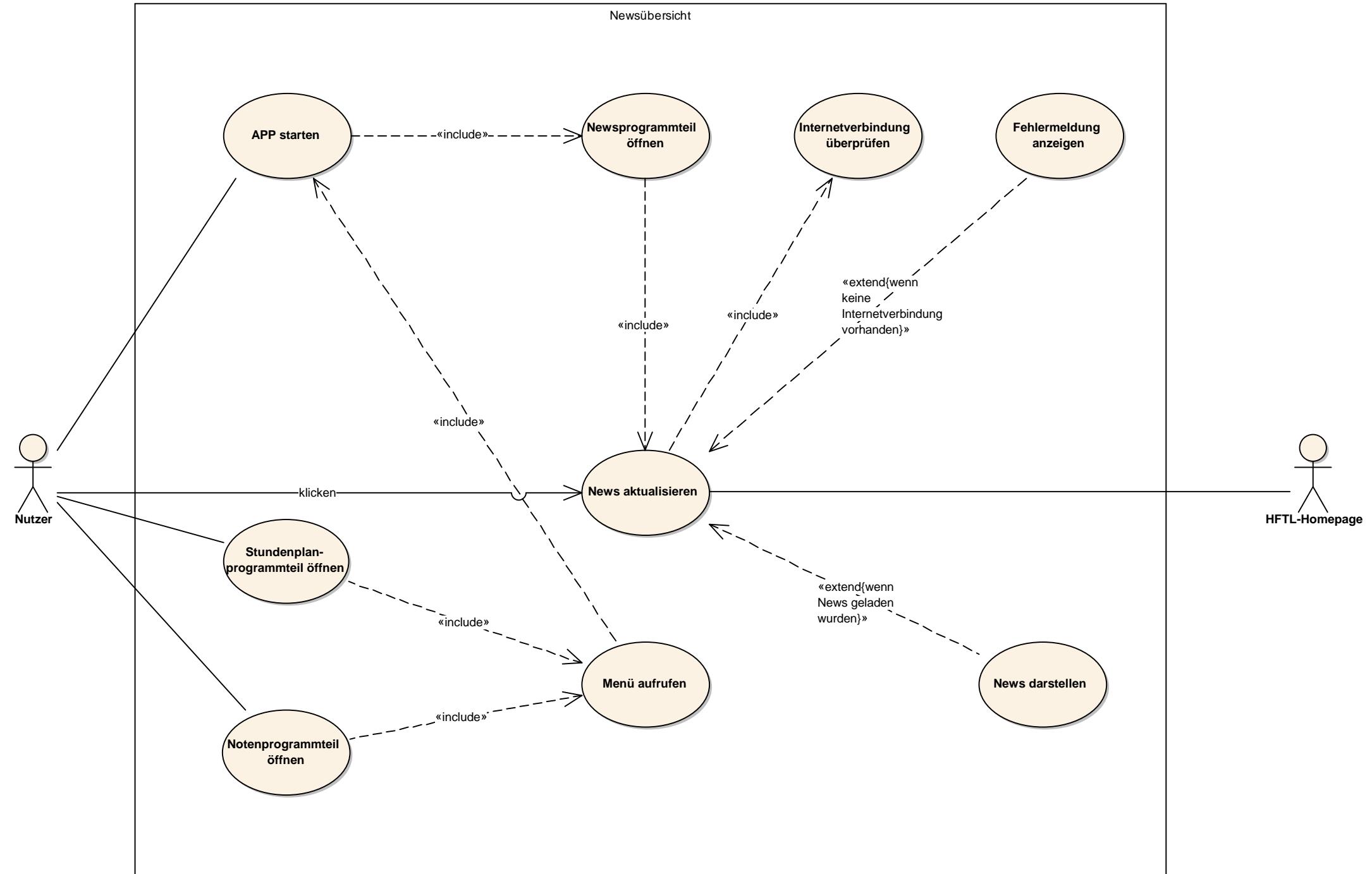


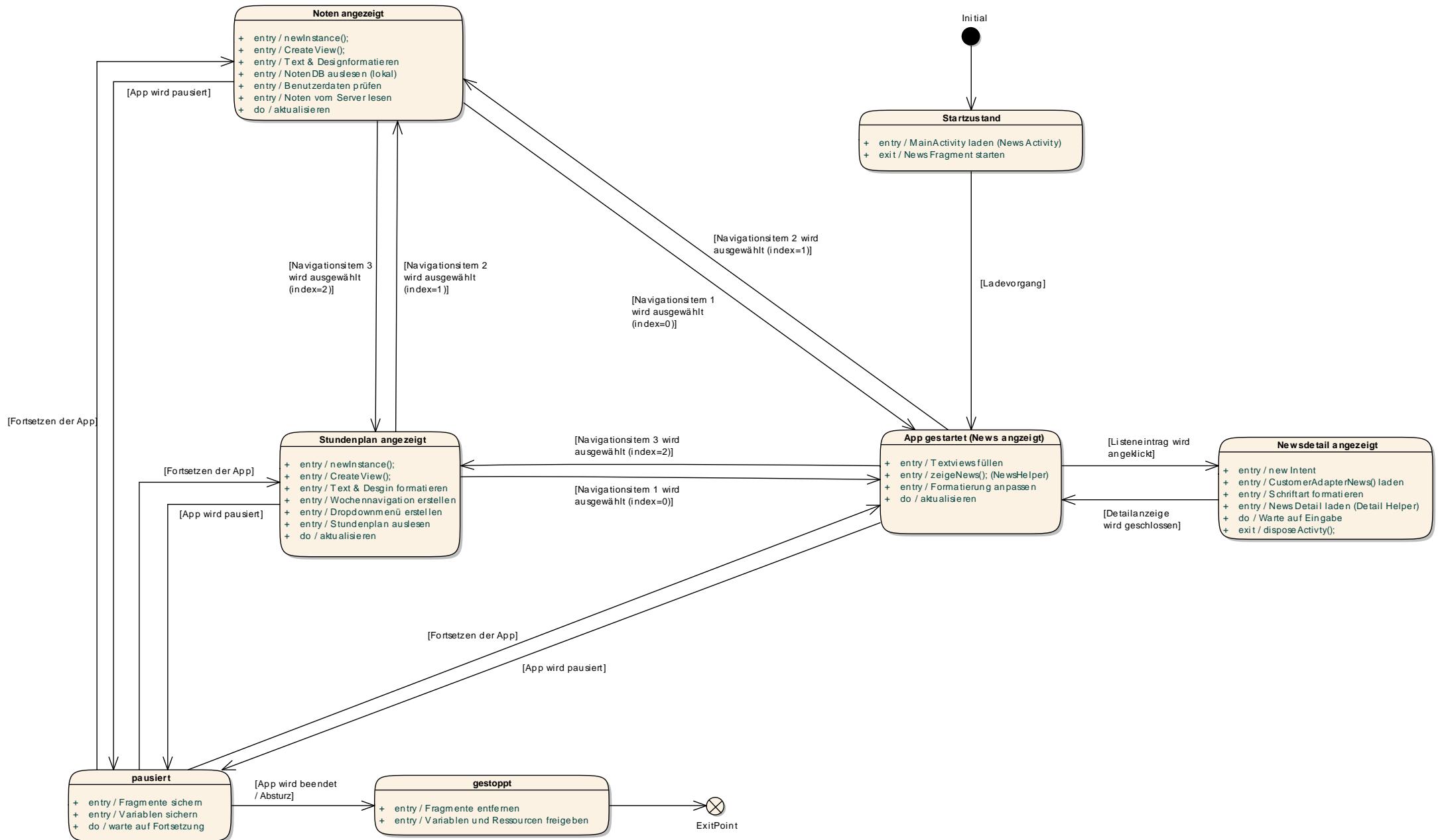


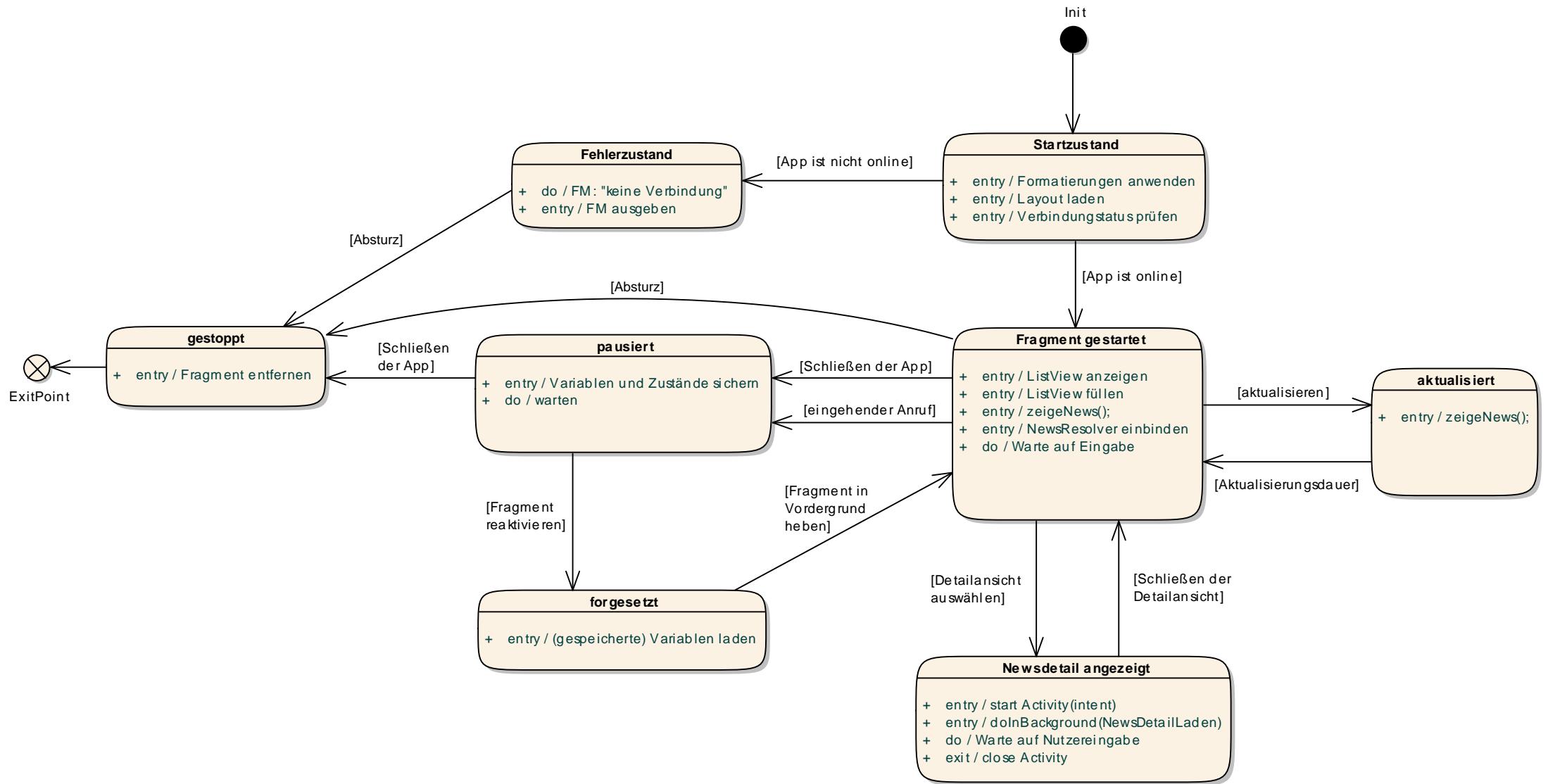






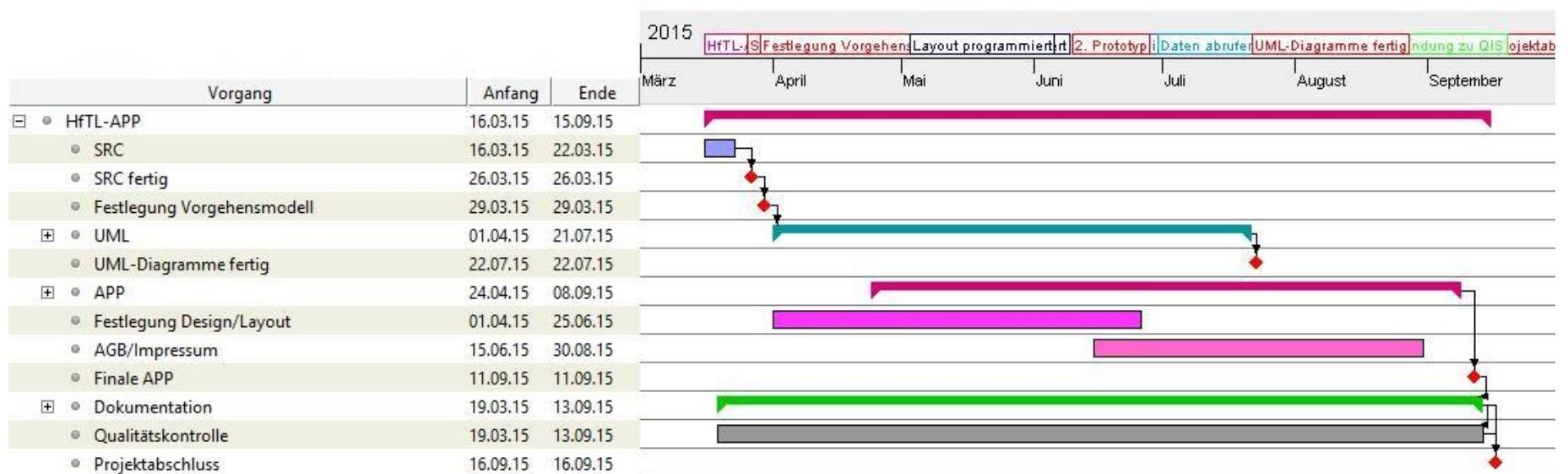




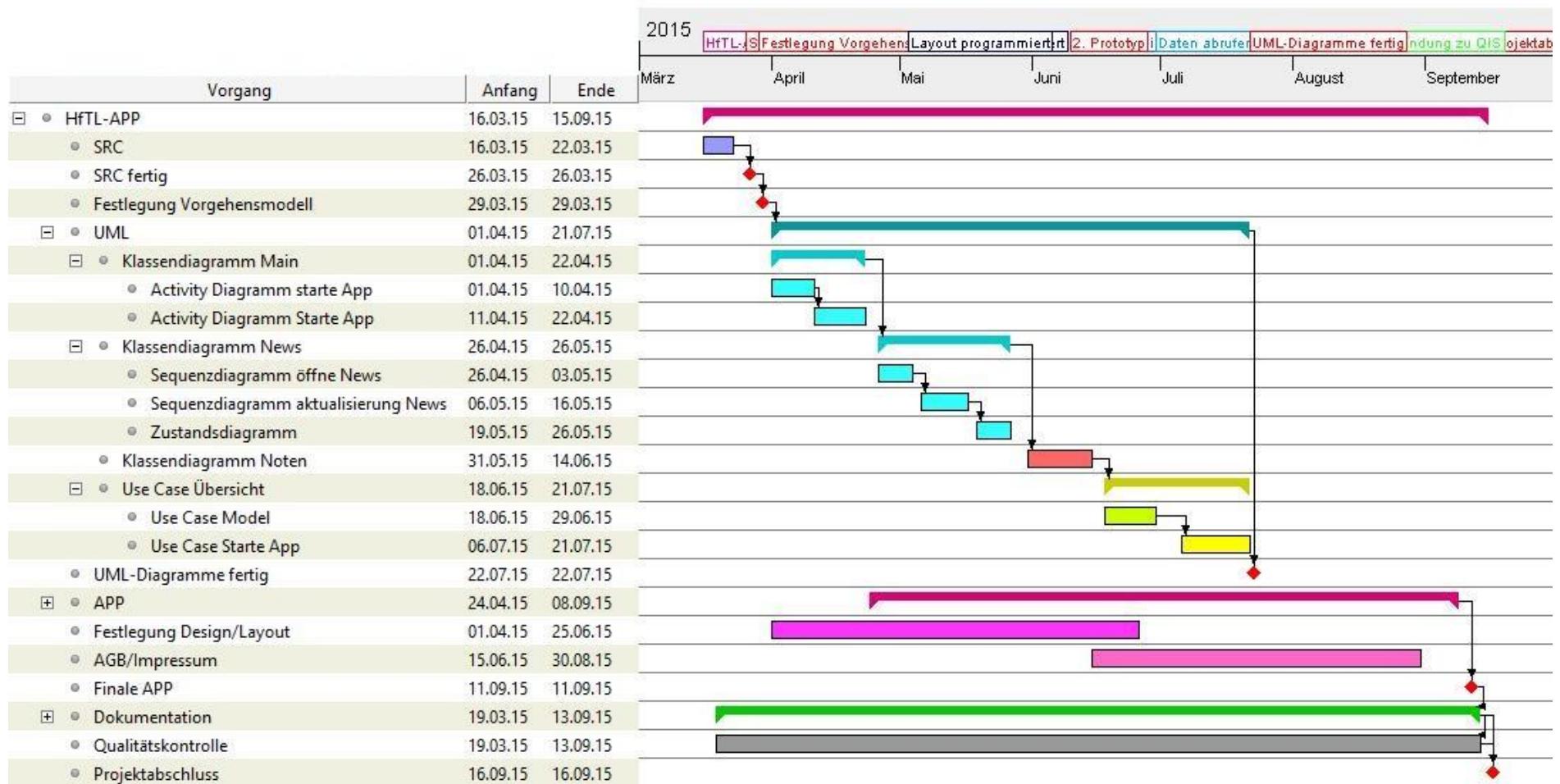


5.3 GANTT

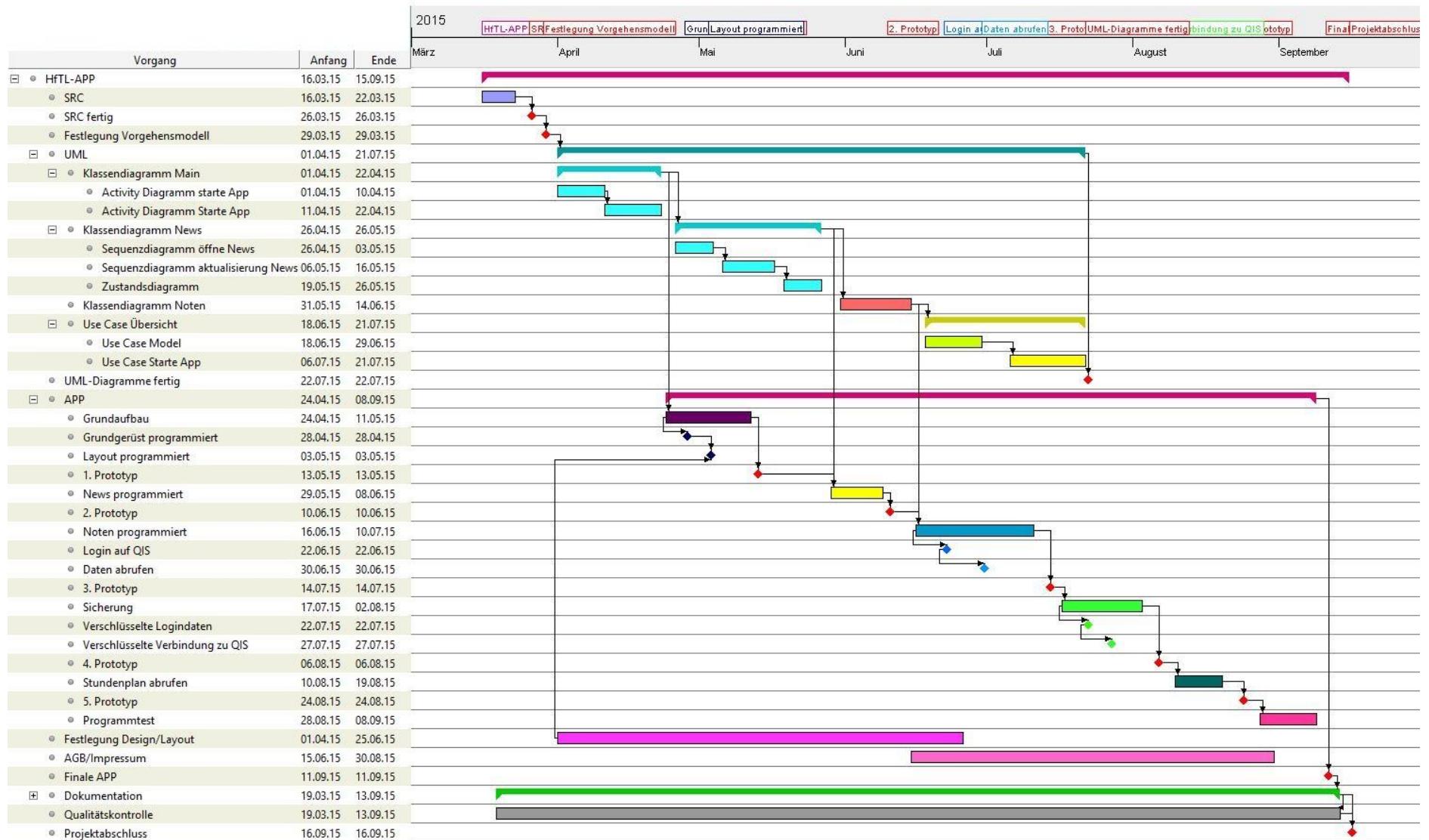
Allgemeine Übersicht



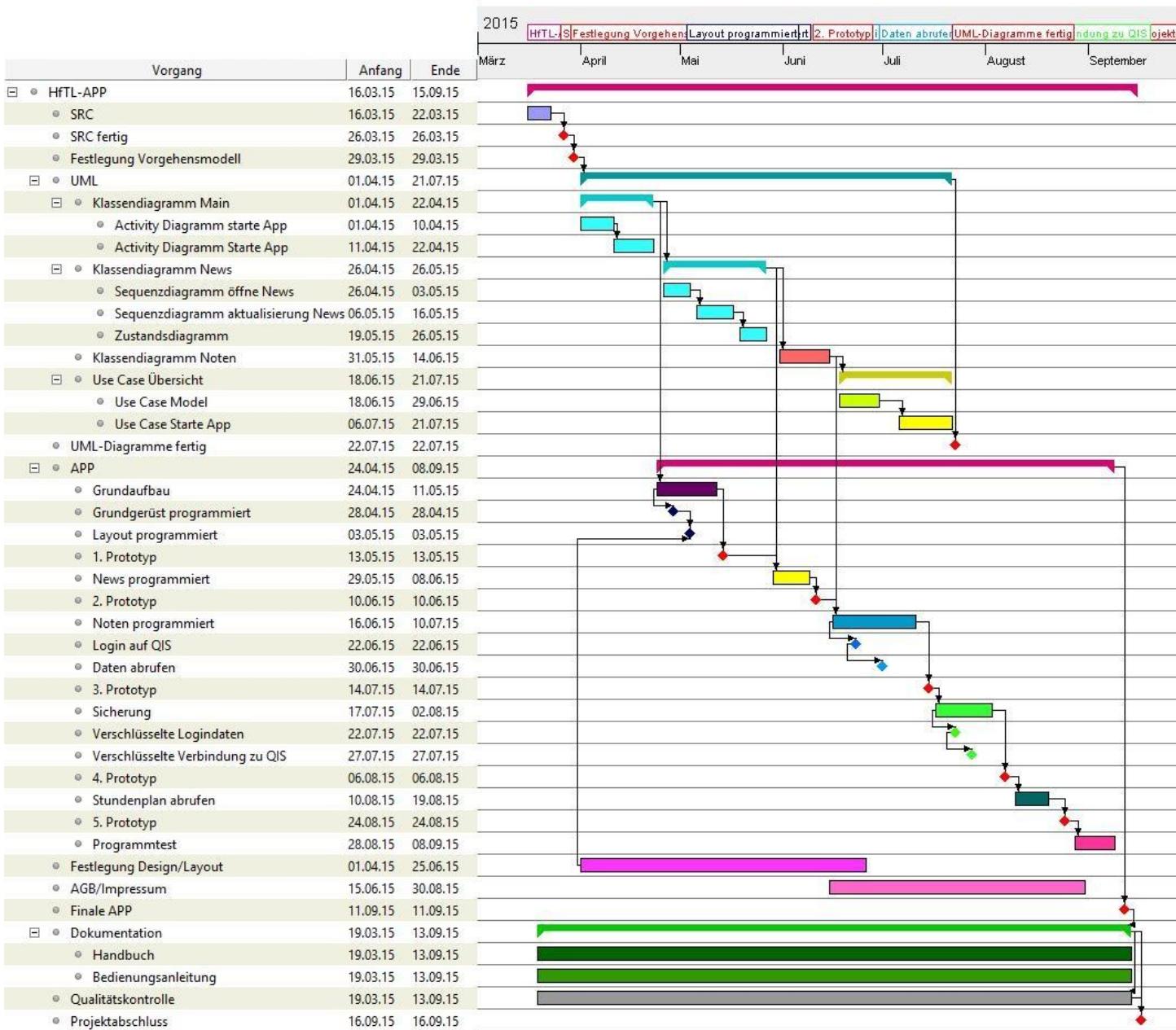
Detailsicht UML



Detailsicht UML & APP



Gesamtübersicht



5.4 APP-LAYOUT



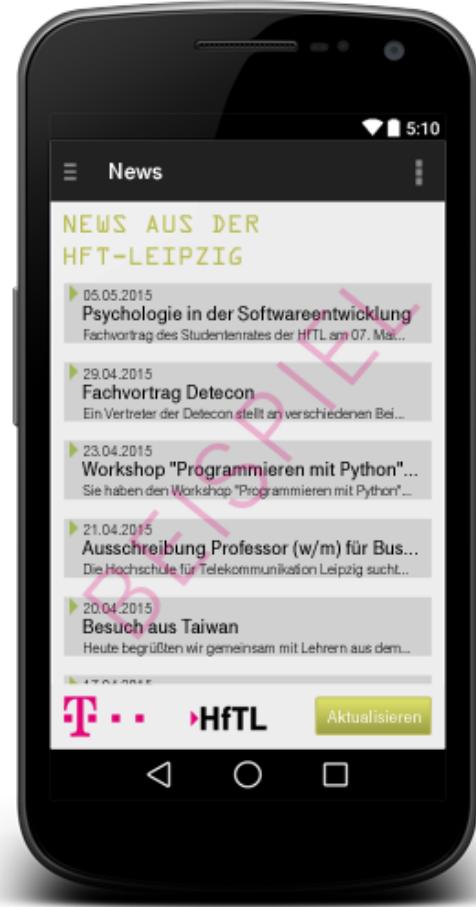
Seite: App-Start
Hintergrund: Grafik_Stift.eps
Logo: HfTL_4C_P.eps
Auflösung: 720 x 1280 px

Content: Padding (global): 8dp;
Hintergrundfarbe (global): Grau04;

Allgemeine Angaben:

- Die angegebenen Werte beziehen sich auf eine Punktdichte von 320dpi
- sollte die verwendete Schriftart nicht verfügbar sein gilt 'Arial' als Rückfalloption
- Der Seitenabstand von 8dp stellt einen Kompromiss zwischen dem CD der Hochschule und den Vorgaben von Google-Developers dar





Seite: News

A - Header: Menü-Button-Grafik: ic_drawer.png
Headline-Schrift: wird vom Smartphone gesetzt

B - Überschrift: Schrift: OCR A Extended; 24sp; Grün;

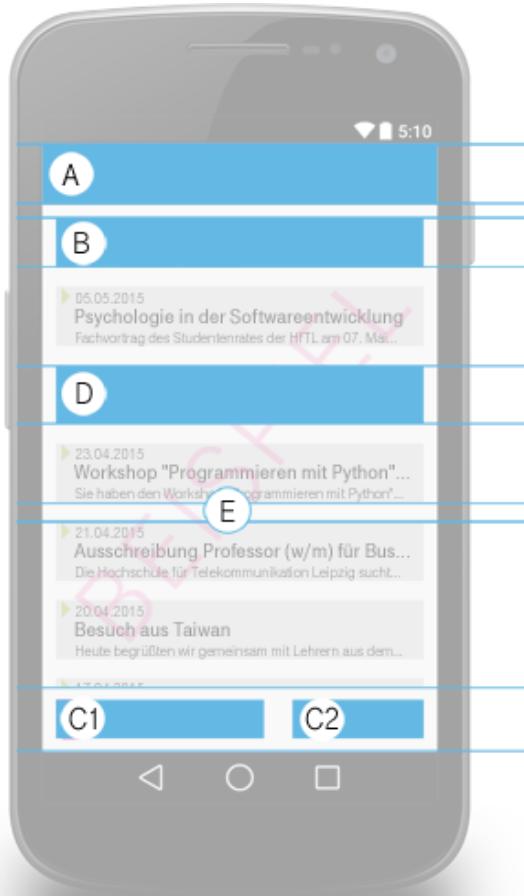
C - Footer

C1 - Logo: HfTL_OB_4C_P.eps

C2 - Button: Schrift: Tele-GroteskHal; 18dp; Weiß;
Verlauf: #E4E96E, #9A9C44

D - Content: Hintergrund: Grau06;
Grafik: Bulletpoint.png; Größe: 15dp;
Linear Layout Orientation: Vertical;
Padding: 2dp;
Schrift: Datum - Tele-GroteskNor; 15sp;
Überschrift - Tele-GroteskHal; 20sp;
Introtext - Tele-GroteskNor; 15sp;
Margin Left: Überschrift - 15dp;
Introtext - 15dp;

E - Abstand: Padding Top + Bottom: 8dp;
Divider: Grau06;





Seite: Artikel
F - Content: Linear Layout Orientation: Vertical;
Schrift: Datum - Tele-GroteskNor; 16sp;
Einleser - Tele-GroteskFet; 16sp;
Lesetext - Tele-GroteskNor; 16sp;
URL - Tele-GroteskHal; 16sp;





Seite: Noten (mit Benutzerdaten)

G - Semester: Hintergrund: Grau06;
Schrift: Tele-GroteskHal; 20sp; Schwarz;
Padding: 4dp;

H - Modulnote: Linear Layout Orientation: Vertical;

H1 - Modul: Schrift: Tele-GroteskHal; 18sp; Schwarz;
Margin Left: 8dp;

H2 - Versuchsanzahl:

Schrift: Tele-GroteskNor; 15sp; Schwarz;
Margin Left: 12dp;

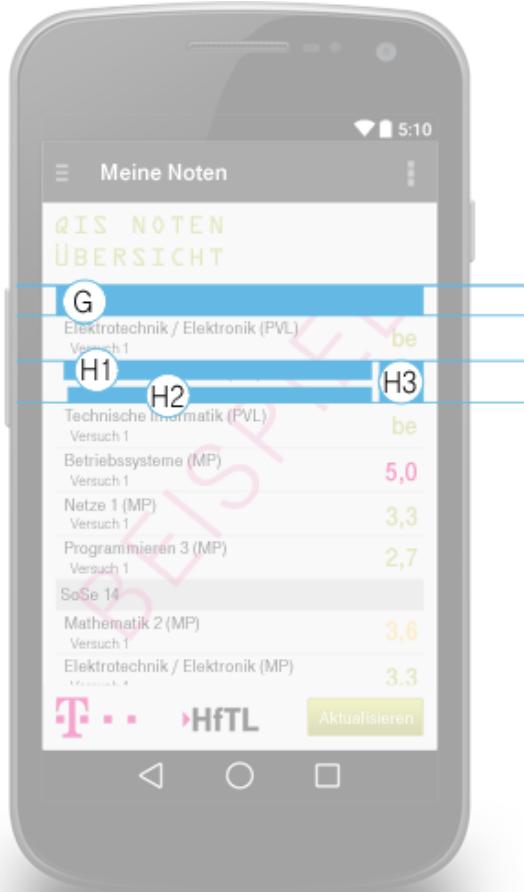
H3 - Note: Schrift: Tele-GroteskHal; 20sp;

Schriftfarbe: Grün - bei Noten <= 3,4
Gelb - bei Noten <= 4,0
Magenta - bei Note 5,0

Layout Gravity: Center;

Margin Left: 4dp;

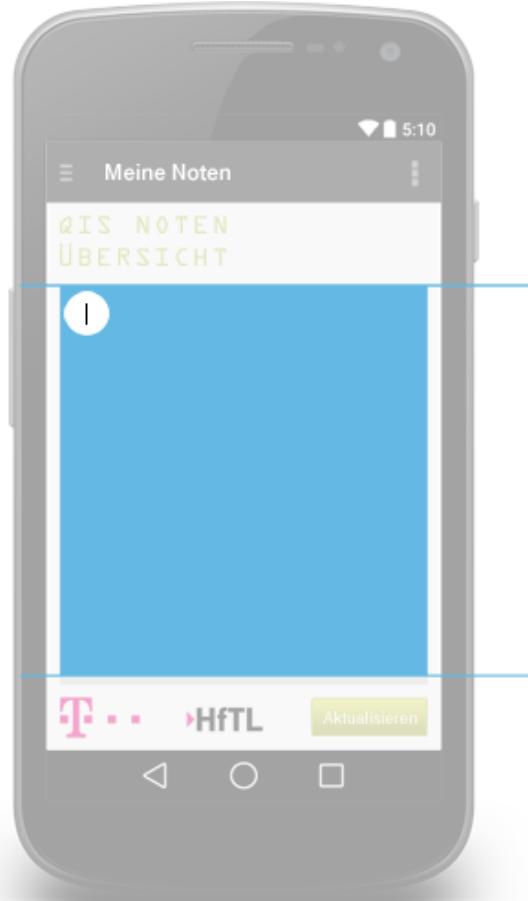
Padding: 4dp;





Seite: Noten (ohne Benutzerdaten)

I-Content: Hintergrund: Grau06;
Schrift: Tele-GroteskHal; 20sp; Weiß;





Seite: Stundenplan

J - Datumwahl: Linear Layout Orientation: Horizontal;

J1 - Button-Zurück:

Grafik: Bulletpoint_rev.png

Grafik onClick: Bulletpoint_rev_magenta.png

Grafik inaktiv: Bulletpoint_rev_grau.png

Größe: 25dp;

Margin: 8dp;

J2 - Spinner: Schrift: wird vom Smartphone gesetzt

Layout Gravity: Center;

Schriftfarbe: Grün;

J3 - Button-Vor:

Grafik: Bulletpoint.png

Grafik onClick: Bulletpoint_magenta.png

Grafik inaktiv: Bulletpoint_grau.png

Größe: 25dp;

Margin: 8dp;

K - Wochentag: Hintergrund: Grau06;

Schrift: Tele-GroteskHal; 20sp; Schwarz;

Padding: 4dp;

L - Modul: Linear Layout Orientation: Vertical;

L1 - Kategorie:

Schrift: Tele-GroteskHal; 18sp; Schwarz;

Margin Left: 8dp;

L2 - Modul: Schrift: Tele-GroteskHal; 18sp; Schwarz;

Margin Left: 8dp;

L3 - Zeit: Schrift: Tele-GroteskNor; 15sp; Schwarz;

Margin Left: 10dp;

L4 - Kategoriefarbe:

Prüfung - Magenta;

Praktikum - Dukelblau;

Vorlesung, Seminar - Grau06;

Margin Left: 4dp;

Padding: 4dp;

Layout Gravity: Center;



5.5 RELEASE-HISTORIE

Pre-release

- v0.1

News und Notenabfrage funktionstüchtig

Ein erster Testrelease. News- und Notenabfrage sind möglich. Design noch nicht komplett implementiert. Stundenplanabfrage noch nicht implementiert.

Pre-release

- v0.2

Newsmodul fertiggestellt

- News optisch angepasst
- Notenabfrage funktionstüchtig, aber optisch noch nicht abgeschlossen.
- Stundenplan noch nicht implementiert

Pre-release

- v0.3

Stundenplan Funktion implementiert

Änderungen:

- Stundenplan Funktion implementiert
- in den Einstellungen kann der Studiengang und das Matrikel gewählt werden
- Design wird noch angepasst

FEHLER:

- wenn der Studiengang nicht vor öffnen des Stundenplans gewählt wird, stürzt die APP ab

Angehängt:

- aktuelle .apk
- aktueller Quelltext

Noten-Design implementiert

JAVA:

NotenTabelle.java:

- TYPO von "SEMESTER" auf "SEMESTER" korrigiert
- Die Spalte "VERSUCH" hinzugefügt

CustomAdapter.java:

- Umbenannt in "CustomAdapterNews"

NewsFragment.java:

- Einbinden des umbenannten CustomAdapters

CustomAdapterNoten

- neuer Adapter für die Darstellung der Noten
- Befüllung von 4 TextViews in der noten_list.xml
- wenn redundante Semesterbeschreibung (Wert: null), dann wird das entsprechende TextView auf "GONE" gesetzt
- farbliche Darstellung der Noten (5 | 4,0 -3,5 | 3,4-1,0)

NotenFragment.java:

- Einbinden des CustomAdaptersNoten.java
- Methode getVersuche() hinzugefügt
- Methode getSemester() angepasst sodass redundante Semesterbezeichnungen als "null" in das StringArray eingetragen werden

NewsResolver.java

- Löschen von auskommentiertem Code, der nicht mehr benötigt wird

NotenResolver.java

- Auslesen des Felds "Versuch" aus QiS
-

XML:

noten_list.xml

- neu erstellt
 - gibt die die 3 bzw 4 TextViews als Reihe im ListViews der NotenView zurück
-

BUGS:

Design Stundenplan und PUSH-Service implementiert

funktionelle Änderungen

- Notenservice eingefügt,
- kleine Änderung am Exceptionhandling
- Anpassung Optionsmenü
- Push-Nachrichten-Funktion implementiert

Probleme:

- Push-Nachrichten starten die Newsaktivität, aber das Fragment wechselt nicht auf NotenFragment.

Todo:

- Layout: Icon für Push einfügen (hat bei mir nicht geklappt -- siehe Todo im Quelltext).
- Intervalleinstellung: "einmal pro Stunde" anstatt "einmal Stunde"

Änderungen am Design:

JAVA:

CustomAdapterNoten:

- aufgrund von Leistungsproblemen wurden die Telekom-Schriftarten für das Notenfragment entfernt (auskommentiert).
- Überschrift des Noten-Fragments bleibt davon unbetroffen.

StundenplanFragment.java:

- Typeface und Schriftart OCRA für die Überschrift des Layouts eingefügt
- Bei der Auswahl der Kalenderwoche im Spinner wird diese nun direkt geladen. Damit ist der Go-Button obsolet. (Wird in einer kommenden Version entfernt)
- Anzeige im Spinner der Woche von "Woche: " auf "KW: " geändert (*)
- Anzeige des Wochentags, neben dem Datum in der ListView

StundenplanResolver.java:

- (*) Änderung des Vergleichspatterns, damit Einträge mit "KW: " verglichen werden

NewsResolver.java:

- Änderung der Newsabfragen (Überschrift, Datum, Link und Text [s. FIXES])
-

XML:

fragment_stundenplan.xml:

- Design des Fragments eingefügt

strings.xml:

- Überschrift für Fragment Stundenplan hinzugefügt
-

SONSTIGES:

Grafik:

- bulletpoint_rev eingefügt, als "zurück-Button" für den Spinner im Stundenplanfragment
-

FIXES:

Da aufgrund der Umstellung am 10.07.2015 der Homepage der HfTL die Parser für die News teilw. ins Leere gelaufen sind, wurde die Newsliste teilw. falsch befüllt oder gar nicht befüllt. Aufgrund dessen kam es zu Crashes der App.

-> Fehler wurden durch Anpassung der Parser behoben.

Kalenderübertragung implementiert

Optimierungen & Einbinden der Schriftarten nach CI/CD

ALLGEMEIN

Neues Package "Typefaces" unter Package "help" erstellt.

Die Schriftarten nach CI/CD konnten nun ohne Probleme eingebunden werden. Durch die Speicheroptimierung konnte der allgemeine Cache-Verbrauch reduziert werden. (ca. 10%)

JAVA:

Neue Klassen erstellt, die via Aufruf in XML die Schriftart für das jeweilige TextView festlegen:

- Ocra.java.java: -TeleGrotFett.java: -TeleGrotHalb.java: -TeleGrotNorm.java:

FontCache.java:

Neue Klasse, die beim Laden der Schriftarten zunächst prüft, ob diese bereits in den Cache eingelagert wurden. Wenn ja, werden diese aus dem Cache verwendet.

Die Individuellen Formatierungen in den Java-Files der einzelnen Fragmente wurden entfernt

- NewsFragment.java
- Notenfragement.java
- StundenplanFragment.java
- NewsClickedActivity.java

CustomAdapterStundenplna.java:

- Variable Zeilenanzahl in Abhängigkeit der Länge der Fachbezeichnung
 - Entfernen von Kommentare
 - Standardtext, falls keine Raumnummern übergeben werden.
-

XML:

Einbinden der Schriftarten über die Java-Klassen:

- fragment_nes.xml
 - fragment_noten.xml
 - fragment_stundenplan.xml
 - activity_news_clicked.xml
 - stundenplan_list.xml
 - noten_list.xml
 - news_list.xml
-

weiteres

Sortierung der Noten nach Semester (Absteigend nach Aktualität)

- Sortierung (BubbleSort) in Notenfragment.java eingefügt
- Dass "null"-Setzen der Semester in eine andere Methode verschoben.
- kleinere Designanpassungen

Pre-release

- v0.7rc
- 74032d2

Release Candidate

- Entwicklerhandbuch fertiggestellt
- Neue Abschnitte (von CM) aus google.docs übernommen
- Neue Bilder eingefügt
- .png durch .jpg ersetzt
- Querverweise eingefügt

Optimierung der Newsdarstellung

NewsResolver.java:

- Auskommentierung der Methode textFormatieren()
- Übergabe der outerHtml-Elemente an den String s[3]
- Entfernung von Leerzeilen (mittels regex)

NewsClickedActivity:

- Alle Bei allen Textviews kann der Inhalt ausgewählt und in die Zwischenablage kopiert werden
- Darstellung von Links in Grün (Nach Designvorlage)
- Verarbeitung des HTML-Strings im TextView
- Links sind klickbar

Bugfix aus Testprotokollen, Einfügen eines Aktualisieren-Button im Optionsmenü vom Stundenplan, Ladebalken für Noten und News, in den Einstellung wird jetzt der ausgewählte Studiengang angezeigt, Timeout für News von 3 sek auf 10 sek geändert

5.6 TESTPROTOKOLLENTWURF

Testprotokoll

Projekt: HFTL-APP

Testobjekt: Dateiname/Pfad (inkl. Versionsnummer)

Prototyp-Version:

Protokoll-ID:

Testumgebung:

Datum:

Tester:

Testfall 1:	<i>Kurze Beschreibung was getestet wird</i>
Erwartetes Ergebnis:	<i>Beschreibung welches Ergebnis erwartet wird</i>
Tatsächliches Ergebnis:	<i>Beschreibung welches Ergebnis tatsächlich erhalten wurde</i>
Testergebnis:	<i>Stimmt das tatsächliche Ergebnis mit dem erwarteten Ergebnis überein?</i>
Aufgetretene Probleme:	<i>Welche Komplikationen sind aufgetreten?</i>
Bewertung:	<i>Aussage, ob der Test als erfolgreich gewertet werden kann. Wenn nicht, dann muss dies detailliert dargelegt werden. Insbesondere die signifikanten Abweichungen sind hervorzuheben.</i>

Testfall 2:	<i>Kurze Beschreibung was getestet wird</i>
Erwartetes Ergebnis:	<i>Beschreibung welches Ergebnis erwartet wird</i>
Tatsächliches Ergebnis:	<i>Beschreibung welches Ergebnis tatsächlich erhalten wurde</i>

Testergebnis:	<i>Stimmt das tatsächliche Ergebnis mit dem erwarteten Ergebnis überein?</i>
Aufgetretene Probleme:	<i>Welche Komplikationen sind aufgetreten?</i>
Bewertung:	<i>Aussage, ob der Test als erfolgreich gewertet werden kann. Wenn nicht, dann muss dies detailliert dargelegt werden. Insbesondere die signifikanten Abweichungen sind hervorzuheben.</i>

Protokollempfänger: Alle Projektteilnehmer

Datum: xx.xx.xxxx

Testprotokoll News Modul

Projekt: HFTL-APP

Testobjekt: Dateiname/Pfad (inkl. Versionsnummer)

Prototyp-Version:

Protokoll-ID:

Testumgebung:

Datum:

Tester:

Funktionstest:

Funktion des News Modul testen

Vorschau News:

Wird eine News-Vorschau angezeigt und ist die Formatierung usw. OK?

ja nein nicht bewertet

Anzeige News:

Wird die News vollständig angezeigt und ist die Formatierung usw. OK?

ja nein nicht bewertet

Aktualisierung:

Lassen sich die News aktualisieren?

ja nein nicht bewertet

Anhänge und Links:

Werden anhänge und Links innerhalb der News angezeigt und funktionieren diese auch?

ja nein nicht bewertet

Sonstiges:

Anmerkungen zum Modul

Hinweis:

z.B. langsames Gerät beim nutzen des Moduls oder dergleichen

Fehlermeldungen:	<i>Fehlermeldungen protokollieren</i>
Fehler:	<i>Beschreibung des Fehlers</i>
Fehlerursache:	<i>Bei welchen eingaben /Bedienung ist der Fehler aufgetreten</i>
Fehler reproduzierbar:	<i>Lässt sich der Fehler zuverlässig reproduzieren?</i>
Kritikalität:	<i>Wie schwer ist der Fehler auf einer Skala von 1-10</i>

Fehlermeldungen:	<i>Fehlermeldungen protokollieren</i>
Fehler:	<i>Beschreibung des Fehlers</i>
Fehlerursache:	<i>Bei welchen eingaben /Bedienung ist der Fehler aufgetreten</i>
Fehler reproduzierbar:	<i>Lässt sich der Fehler zuverlässig reproduzieren?</i>
Kritikalität:	<i>Wie schwer ist der Fehler auf einer Skala von 1-10</i>

Protokollempfänger: Alle Projektteilnehmer

Datum: xx.xx.xxxx

Testprotokoll Noten Modul

Projekt: HFTL-APP

Testobjekt: Dateiname/Pfad (inkl. Versionsnummer)

Prototyp-Version:

Protokoll-ID:

Testumgebung:

Datum:

Tester:

Funktionstest:

Funktion des Noten Modul testen

Abfrage:

Funktioniert die Abfrage der Noten fehlerfrei?

ja nein nicht bewertet

Intervallabfrage:

Funktioniert das Abfragen der Noten mit einem voreigestellten Intervall?

ja nein nicht bewertet

Speicherung der Daten:

Werden die Noten auf dem Gerät gespeichert?

ja nein nicht bewertet

Benachrichtigung:

Funktionieren die Benachrichtigungen bei Änderungen?

ja nein nicht bewertet

Datenübertragung:

Werden die Benutzerdaten verschlüsselt übertragen oder im Klartext?

ja nein nicht bewertet

Formatierung:

Ist das Layout in Ordnung? (wird alles Lesbar dargestellt)

ja nein nicht bewertet

Einstellungen:	<i>Test der Einstellungen des Moduls</i>		
Eingaben:	<i>Können Benutzername und Passwort eingegeben werden ohne Fehler zu erzeugen?</i>		
	<input type="checkbox"/> ja	<input type="checkbox"/> nein	<input type="checkbox"/> nicht bewertet
Intervallabfrage:	<i>Lässt sich ein Abfrageintervall einstellen?</i>		
	<input type="checkbox"/> ja	<input type="checkbox"/> nein	<input type="checkbox"/> nicht bewertet
Benachrichtigungen:	<i>Lässt sich einstellen ob man bei Änderungen benachrichtigt werden möchte?</i>		
	<input type="checkbox"/> ja	<input type="checkbox"/> nein	<input type="checkbox"/> nicht bewertet
Länge Eingabefelder:	<i>Ist die Länge der Eingabefelder ausreichend?</i>		
	<input type="checkbox"/> ja	<input type="checkbox"/> nein	<input type="checkbox"/> nicht bewertet
Erlaubter Zeichensatz:	<i>Sind die erlaubten Zeichen ausreichend um z.B. ein sicheres Passwort zu garantieren?</i>		
	<input type="checkbox"/> ja	<input type="checkbox"/> nein	<input type="checkbox"/> nicht bewertet
Sonderzeichen:	<i>Sind Sonderzeichen erlaubt? Wenn ja welche und werden diese fehlerfrei akzeptiert?</i>		
	<input type="checkbox"/> ja	<input type="checkbox"/> nein	<input type="checkbox"/> nicht bewertet
	<i>Wenn ja:</i>		
Zu kurze Eingaben:	<i>Was ist bei kurzen bis sehr kurzen Eingaben? (z.B. nur ein Zeichen oder keines)</i>		
	<i>Bemerkung:</i>		
Zu lange Eingaben:	<i>Was ist bei sehr langen Eingaben? (größer als der Variablenbereich)</i>		
	<i>Bemerkung:</i>		
Nicht unterstützte Zeichen / Sonderzeichen:	<i>Was passiert bei der Verwendung von nicht unterstützten Zeichen und Sonderzeichen?</i>		
	<i>Bemerkung:</i>		
Speicherung:	<i>Funktioniert das Speichern der Benutzerdaten fehlerfrei?</i>		
	<input type="checkbox"/> ja	<input type="checkbox"/> nein	<input type="checkbox"/> nicht bewertet

Verschlüsselung:	<p><i>Werden Benutzername und Passwort verschlüsselt gespeichert?</i></p> <p><input type="checkbox"/> ja <input type="checkbox"/> nein <input type="checkbox"/> nicht bewertet</p>
	<p><i>Funktioniert die Verschlüsselung?</i></p> <p><input type="checkbox"/> ja <input type="checkbox"/> nein <input type="checkbox"/> nicht bewertet</p>
Anzeige Benutzerdaten:	<p><i>Wird das Passworts in den Einstellungen im Klartext angezeigt?</i></p> <p><input type="checkbox"/> ja <input type="checkbox"/> nein <input type="checkbox"/> nicht bewertet</p>
	<p><i>Wird der Benutzername in den Einstellungen im Klartext angezeigt?</i></p> <p><input type="checkbox"/> ja <input type="checkbox"/> nein <input type="checkbox"/> nicht bewertet</p>

Sonstiges:	<p><i>Anmerkungen zum Modul</i></p>
Hinweis:	<p><i>z.B. langsames Gerät beim nutzen des Moduls oder dergleichen</i></p>

Fehlermeldungen:	<p><i>Fehlermeldungen protokollieren</i></p>
Fehler:	<p><i>Beschreibung des Fehlers</i></p>
Fehlerursache:	<p><i>Bei welchen eingaben /Bedienung ist der Fehler aufgetreten</i></p>
Fehler reproduzierbar:	<p><i>Lässt sich der Fehler zuverlässig reproduzieren?</i></p>
Kritikalität:	<p><i>Wie schwer ist der Fehler auf einer Skala von 1-10</i></p>

Fehlermeldungen:	<p><i>Fehlermeldungen protokollieren</i></p>
Fehler:	<p><i>Beschreibung des Fehlers</i></p>
Fehlerursache:	<p><i>Bei welchen eingaben /Bedienung ist der Fehler aufgetreten</i></p>

Fehler reproduzierbar:

Lässt sich der Fehler zuverlässig reproduzieren?

Kritikalität:

Wie schwer ist der Fehler auf einer Skala von 1-10

Protokollempfänger: Alle Projektteilnehmer

Datum: xx.xx.xxxx

Testprotokoll Stundenplan Modul

Projekt: HFTL-APP

Testobjekt: Dateiname/Pfad (inkl. Versionsnummer)

Prototyp-Version:

Protokoll-ID:

Testumgebung:

Datum:

Tester:

Funktionstest:

Funktion des Stundenplan Modul testen

Abfrage:

Funktioniert die Abfrage des Stundenplans fehlerfrei?

ja nein nicht bewertet

Intervallabfrage:

Funktioniert das Abfragen des Stundenplans mit einem voreigestellten Intervall?

ja nein nicht bewertet

Synchronisierung:

Funktioniert die Synchronisierung mit dem Kalender des Smartphones?

ja nein nicht bewertet

Studiengang:

Werden nur zum Nutzer passende Stundenpläne angezeigt?

ja nein nicht bewertet

Formatierung:

Ist das Layout in Ordnung? (wird alles lesbar dargestellt)

ja nein nicht bewertet

Einstellungen:	<i>Test der Einstellungen des Moduls</i>
Wochenwahl:	<i>Funktioniert die Anzeige verschiedener Wochen?</i> <input type="checkbox"/> ja <input type="checkbox"/> nein <input type="checkbox"/> nicht bewertet
Intervallabfrage:	<i>Lässt sich ein Abfrageintervall einstellen?</i> <input type="checkbox"/> ja <input type="checkbox"/> nein <input type="checkbox"/> nicht bewertet
Benachrichtigungen:	<i>Funktionieren die Benachrichtigungen bei sich ändernden Stundenplänen usw.?</i> <input type="checkbox"/> ja <input type="checkbox"/> nein <input type="checkbox"/> nicht bewertet
Sonstiges:	<i>Anmerkungen zum Modul</i>
Hinweis:	<i>z.B. langsames Gerät beim nutzen des Moduls oder dergleichen</i>
Fehlermeldungen:	<i>Fehlermeldungen protokollieren</i>
Fehler:	<i>Beschreibung des Fehlers</i>
Fehlerursache:	<i>Bei welchen eingaben /Bedienung ist der Fehler aufgetreten</i>
Fehler reproduzierbar:	<i>Lässt sich der Fehler zuverlässig reproduzieren?</i>
Kritikalität:	<i>Wie schwer ist der Fehler auf einer Skala von 1-10</i>
Fehlermeldungen:	<i>Fehlermeldungen protokollieren</i>
Fehler:	<i>Beschreibung des Fehlers</i>
Fehlerursache:	<i>Bei welchen eingaben /Bedienung ist der Fehler aufgetreten</i>
Fehler reproduzierbar:	<i>Lässt sich der Fehler zuverlässig reproduzieren?</i>
Kritikalität:	<i>Wie schwer ist der Fehler auf einer Skala von 1-10</i>

Protokollempfänger: Alle Projektteilnehmer

Datum: xx.xx.xxxx

5.7 TESTPROTOKOLLE

Gesamtprotokolle

ID.	ERSTELLT	TESTER	TESTTYP	BESCHREIBUNG	TESTERGEBNIS	ERFOLGSQUOTE	STATUS	GESAMT
1	27.05.2015	ML	Systemtest	Test der App auf einem Smartphone im Echtzeitbetrieb	nicht bestanden	0 - 29 %	ungenügend	nicht bestanden
2	31.05.2015	ML	Modultest	Funktionstest News-Fragment und Formatierungsprüfung	bestanden	81 - 91 %	gut	bestanden
3	31.05.2015	ML	Modultest	Funktionstest Noten-Fragment ohne Benutzerdaten	bestanden	92 - 100 %	sehr gut	bestanden
4	31.05.2015	ML	Modultest	Funktionstest Noten-Fragment mit Benutzerdaten	bestanden	81 - 91 %	gut	bestanden
5	07.06.2015	PK	UML	Überprüfung UML Diagramm	bestanden	81 - 91 %	gut	bestanden
6	07.06.2015	PK	UML	Überprüfung UML Diagramm	bestanden	81 - 91 %	gut	bestanden
7	08.06.2015	PK	UML	Überprüfung UML Diagramm	bestanden	67 - 80 %	befriedigend	bestanden
8	08.06.2015	PK	UML	Überprüfung UML Diagramm	bestanden	67 - 80 %	befriedigend	nicht bestanden
9	08.06.2015	PK	UML	Überprüfung UML Diagramm	bestanden	81 - 91 %	gut	bestanden
10	08.06.2015	PK	UML	Überprüfung UML Diagramm	nicht bestanden	50 - 66 %	ausreichend	nicht bestanden
11	10.06.2015	PK	Modultest	News	bestanden	50 - 66 %	ausreichend	nicht bestanden
12	10.06.2015	PK	Modultest	Noten	bestanden	30 - 49 %	mangelhaft	nicht bestanden
13	10.06.2015	PK	Modultest	Stundenplan (nicht implementiert)	nicht bestanden	0 - 29 %	ungenügend	nicht bestanden
14	20.06.2015	ML	Modultest	Stundenplan	bestanden	67 - 80 %	befriedigend	bestanden
15	23.06.2015	PK	Modultest	News	bestanden	50 - 66 %	ausreichend	nicht bestanden
16	23.06.2015	PK	Modultest	Noten	bestanden	67 - 80 %	befriedigend	nicht bestanden
17	23.06.2015	PK	Modultest	Stundenplan	bestanden	30 - 49 %	mangelhaft	nicht bestanden
18	23.06.2015	ML	Systemtest	Prüfung der App auf Formatierungen	bestanden	81 - 91 %	gut	bestanden
19	28.06.2015	ML	Modultest	Noten	bestanden	92 - 100 %	sehr gut	bestanden
20	05.07.2015	PK	Modultest	News	nicht bestanden	30 - 49 %	mangelhaft	nicht bestanden
21	05.07.2015	PK	Modultest	Noten	bestanden	81 - 91 %	gut	bestanden
22	05.07.2015	PK	Modultest	Stundenplan	bestanden	81 - 91 %	gut	nicht bestanden
23	14.07.2015	PK	Modultest	News	nicht bestanden	0 - 29 %	ungenügend	nicht bestanden
24	14.07.2015	PK	Modultest	Noten	bestanden	81 - 91 %	gut	bestanden
25	14.07.2015	PK	Modultest	Stundenplan	bestanden	81 - 91 %	gut	nicht bestanden
26	15.07.2015	ML	Modultest	Stundenplan	nicht bestanden	67 - 80 %	befriedigend	nicht bestanden
27	09.08.2015	ML	Modultest	Stundenplan	nicht bestanden	67 - 80 %	befriedigend	nicht bestanden
28	09.08.2015	ML	Modultest	News	bestanden	81 - 91 %	gut	bestanden
29	10.08.2015	PK	UML	Aktivitätsdiagramm App starten	bestanden	92 - 100 %	sehr gut	bestanden

30	10.08.2015	PK	UML	Aktivitätsdiagramm News starten	bestanden	92 - 100 %	sehr gut	bestanden
31	11.08.2015	PK	UML	Klassendiagramm Main	bestanden	92 - 100 %	sehr gut	bestanden
32	12.08.2015	PK	UML	Klassendiagramm News	bestanden	92 - 100 %	sehr gut	bestanden
33	13.08.2015	PK	UML	Sequenzdiagramm aktualisiere Newsliste	bestanden	81 - 91 %	gut	bestanden
34	14.08.2015	PK	UML	Sequenzdiagramm oeffne News	bestanden	81 - 91 %	gut	bestanden
35	15.08.2015	PK	UML	Use-Case-diagramm grobe Übersicht	bestanden	67 - 80 %	befriedigend	nicht bestanden
36	16.08.2015	PK	UML	Use-Case-diagramm starteApp	bestanden	67 - 80 %	befriedigend	nicht bestanden
37	17.08.2015	PK	UML	Zustandsdiagramm App	nicht bestanden	81 - 91 %	gut	nicht bestanden
38	18.08.2015	PK	UML	Zustandsdiagramm NewsFragment	nicht bestanden	81 - 91 %	gut	nicht bestanden
39	10.08.2015	JS	Modultest	Stundenplan	bestanden	92 - 100 %	sehr gut	bestanden
40	10.08.2015	JS	Modultest	News	bestanden	92 - 100 %	sehr gut	bestanden
41	10.08.2015	JS	Modultest	Noten	bestanden	92 - 100 %	sehr gut	bestanden
42	19.08.2015	PK	Modultest	News	bestanden	81 - 91 %	gut	bestanden
43	19.08.2015	PK	Modultest	Noten	bestanden	92 - 100 %	sehr gut	bestanden
44	19.08.2015	PK	Modultest	Stundenplan	bestanden	81 - 91 %	gut	nicht bestanden
45	20.08.2015	PK	UML	Aktivitätsdiagramm App starten	bestanden	92 - 100 %	sehr gut	bestanden
46	20.08.2015	PK	UML	Aktivitätsdiagramm News starten	bestanden	92 - 100 %	sehr gut	bestanden
47	20.08.2015	PK	UML	Sequenzdiagramm aktualisiere Newsliste	bestanden	92 - 100 %	sehr gut	bestanden
48	20.08.2015	PK	UML	Sequenzdiagramm oeffne News	bestanden	92 - 100 %	sehr gut	bestanden
49	20.08.2015	PK	UML	Use-Case-diagramm grobe Übersicht	bestanden	92 - 100 %	sehr gut	bestanden
50	20.08.2015	PK	UML	Use-Case-diagramm starteApp	bestanden	92 - 100 %	sehr gut	bestanden
51	20.08.2015	PK	UML	Zustandsdiagramm App	bestanden	92 - 100 %	sehr gut	bestanden
52	20.08.2015	PK	UML	Zustandsdiagramm NewsFragment	bestanden	92 - 100 %	sehr gut	bestanden
53	21.08.2015	PK	UML	Klassendiagramm Main	bestanden	92 - 100 %	sehr gut	bestanden
54	21.08.2015	PK	UML	Klassendiagramm News	bestanden	92 - 100 %	sehr gut	bestanden
55	22.08.2015	PK	Unterlagen	Projektdokumentation	nicht bestanden	67 - 80 %	befriedigend	nicht bestanden
56	23.08.2015	AD	Modultest	News	nicht bestanden	67 - 80 %	befriedigend	nicht bestanden
57	23.08.2015	AD	Modultest	Noten	bestanden	92 - 100 %	sehr gut	bestanden
58	06.09.2015	SC	Modultest	News	bestanden	92 - 100 %	sehr gut	bestanden
59	06.09.2015	SC	Modultest	Noten	bestanden	92 - 100 %	sehr gut	bestanden
60	06.09.2015	SC	Modultest	Stundenplan	bestanden	92 - 100 %	sehr gut	bestanden
61	06.09.2015	CM	Modultest	Gesamttest: Noten / News / St.plan	bestanden	92 - 100 %	sehr gut	bestanden

62	07.09.2015	PK	Unterlagen	Projektdokumentation	nicht bestanden	81 - 91 %	gut	nicht bestanden
63	09.09.2015	SK	Modultest	News	bestanden	92 - 100 %	sehr gut	bestanden
64	09.09.2015	SK	Modultest	Noten	bestanden	92 - 100 %	sehr gut	bestanden
65	09.09.2015	SK	Modultest	Stundenplan	bestanden	92 - 100 %	sehr gut	bestanden

Testprotokoll

Projekt: HFTL-APP
Testobjekt: HfTL-App (Stand: 27.05.2015, 20:00Uhr)
Prototyp-Version: V02
Protokoll-ID: 01
Testumgebung: HTC One Android 5.0.2
Datum: 27.05.15
Tester: ML

Testfall 1:	<i>Test der App auf einem Smartphone im Echtzeitbetrieb</i>
Erwartetes Ergebnis:	<i>App startet und die Funktionen können getestet werden</i>
Tatsächliches Ergebnis:	<i>App konnte nicht gestartet werden, es war kein Funktionstest möglich</i>
Testergebnis:	<i>Es fand kein erfolgreicher Test statt</i>
Aufgetretene Probleme:	<i>Execution failed for task ':app:mergeDebugResources'. > C:\..\App\HfTL-App\app\src\main\res\values\strings.xml:15:1: Error: Der Content von Elementen muss aus ordnungsgemäß formatierten Zeichendaten oder Markups bestehen.</i> <i>BUILD FAILED</i>
Bewertung:	<i>Der Test der App im Echtzeitbetrieb konnte nicht durchgeführt werden. Fehler in der Programmierung erzeugen Fehler in der Erstellung einer lauffähigen App.</i>

Protokollempfänger: Alle Projektteilnehmer
Datum: 31.05.2015

Testprotokoll

Projekt: HFTL-APP
Testobjekt: HfTL-App (Stand: 23.06.2015, 11:00Uhr)
Prototyp-Version: V02
Protokoll-ID: 18
Testumgebung: HTC One Android 5.0.2
Datum: 23.06.2015
Tester: ML

Testfall 1:	<i>Test der aktuellen Version auf Formatierungen</i>
Erwartetes Ergebnis:	<i>Grundlegende Formatierungen an die Designvorlage mit kleineren Änderungspunkten</i>
Tatsächliches Ergebnis:	<i>Formatierungen sind weitestgehend an die Designvorlage angepasst, es sind Änderungen in den Abmessungen erforderlich.</i>
Testergebnis:	<i>Der Test lief wie erwartet.</i>
Aufgetretene Probleme:	<i>App-Abstürze durch Fehlbedienung, fehlender Daten.</i>
Bewertung:	<i>Die einzelnen Module sind an die Designvorlage angepasst. Innen- und Außenabstände einzelner Elemente passen nicht und unterscheidet sich zur Vorschau über Android-Studio. Schriftarten sind nicht eingebunden oder funktionieren nicht. In den News werden nicht alle Formatierungen des Textes übernommen, es fehlen Links und Anhänge. Im Modul Noten werden die Farben der Noten falsch ausgegeben. Das Modul Stundenplan ist noch im funktionalen Aufbau.</i>

Protokollempfänger: Alle Projektteilnehmer
Datum: 23.06.2015

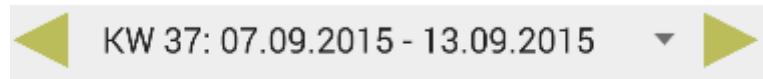
Testprotokoll

Projekt: HFTL-APP
Testobjekt: Dateiname/Pfad (inkl. Versionsnummer)
Prototyp-Version: V6
Protokoll-ID: 61
Testumgebung: Genymotion Android Emulator mit Samsung Galaxy S5
Datum: 5.9.2015
Tester: Christoph Mathies

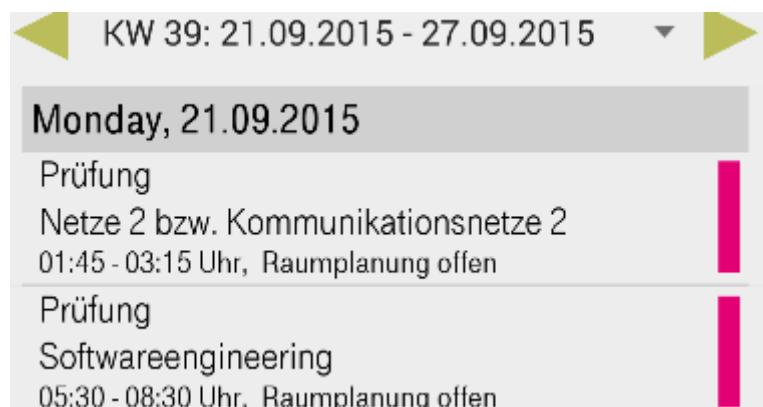
Testfall 1:	<i>Gesamte App, News Fragment: Funktionalität / Design</i>
Erwartetes Ergebnis:	<i>News werden korrekt angezeigt / Design ist durchgehend einheitlich</i>
Tatsächliches Ergebnis:	<i>News werden entsprechend korrekt und formatiert geladen unabhängig von eingegebenen oder nicht eingegebenen Nutzerdaten (user + PW)</i>
Testergebnis:	<i>Erwartetes Ergebnis und tatsächliches Ergebnis stimmen überein, alles OK</i>
Aufgetretene Probleme:	Welche Komplikationen sind aufgetreten? Keine
Bewertung:	<i>Die Erwartungen an die Newsübersicht, Detailansicht und Formatierung sind zu 100% erfüllt worden.</i>

Testfall 2:	<p><i>Gesamte App: Testen des Stundenplanfragments / Einstellung</i></p>
Erwartetes Ergebnis:	<p><i>Zukünftige Termine werden entsprechend formatiert und korrekt angezeigt.</i></p>
Tatsächliches Ergebnis:	<ul style="list-style-type: none"><i>-Bei der Eingabe der Nutzerinformation (user, pw und Studiengang) nimmt die App die eingegebenen Informationen nicht an und stürzte sogar ab.</i><i>-Nach Neustart sind die gespeicherten Daten vorhanden und das Stundenplan-Fragment arbeitet wie erhofft.</i><i>-Anzeigefehler innerhalb des Stundenplans</i>
Testergebnis:	<p><i>Stimmt das tatsächliche Ergebnis mit dem erwarteten Ergebnis überein?</i></p> <ul style="list-style-type: none"><i>-teilweise (die aktuelle Kalenderwoche wird falsch angezeigt) Testdatum: 6.9.2015 (KW 36.)</i><i>- in der App wird in der aktuellen Woche bereits die KW 37 angezeigt, incl. der nächstwöchigen Termine außerdem mit falschem Datum (7.9 – 6.9.2015):</i> 

Die Folgewochen sind korrekt formatiert – (ein Klick auf den Pfeil nach rechts):



Außerdem werden die Uhrzeiten der Termine falsch angezeigt, was aber am Emulator liegen könnte, Zeit habe ich von 7.00 a.m. auf 1 p.m. angepasst (Diff=6h)



Aufgetretene Probleme:

Welche Komplikationen sind aufgetreten?

- Fehler beim erstmaligen Einlesen von Anmeldeinformationen, Rekonstruktion leider nicht möglich. Im zweiten Versuch war alles OK.

Fehlerrekonstruktion:

- App Cache gelöscht
- App neuinstalliert
- Vom NewsFragment auf Stundenplan gewechselt
- Aufforderung zur Dateneingabe -> Ja
- Daten eingetragen
Fehler tritt nicht erneut auf

Bewertung:

Der Test wurde im Großen und Ganzen bestanden. Die Anzeigefehler innerhalb des Stundenplans sind vermutlich auf den, in US-Einstellungen laufenden Emulator zurückzuführen

Testfall 3:	<i>Gesamte App, Noten-Fragment: Funktionalität / Design</i>
Erwartetes Ergebnis:	<i>Anzeige der Noten in entsprechender Form</i>
Tatsächliches Ergebnis:	<i>Noten werden (nach Aufforderung von der Eingabe der Nutzerdaten) angezeigt. Form ist sehr gut und übersichtlich</i>
Testergebnis:	<i>Erwartetes Ergebnis und tatsächliches Ergebnis stimmen überein, alles OK</i>
Aufgetretene Probleme:	Welche Komplikationen sind aufgetreten? Keine
Bewertung:	<i>Die Erwartungen an die Notenübersicht und deren Formatierung sind zu 100% erfüllt worden.</i>

Protokollempfänger: Alle Projektteilnehmer

Datum: 05.09.2015 aktualisiert am 06.09.2015

Testprotokolle zum News Modul

Testprotokoll News Modul

Projekt: HFTL-APP
Testobjekt: HfTL-App (Stand: 31.05.2015, 14:30Uhr)
Prototyp-Version: V02
Protokoll-ID: 02
Testumgebung: HTC One Android 5.0.2
Datum: 31.05.2015
Tester: ML

Funktionstest:	<i>Funktion des News Modul testen</i>		
Vorschau News:	<i>Wird eine News-Vorschau angezeigt und ist die Formatierung usw. OK?</i>		
	<input checked="" type="checkbox"/> ja	<input type="checkbox"/> nein	<input type="checkbox"/> nicht bewertet
Anzeige News:	<i>Wird die News vollständig angezeigt und ist die Formatierung usw. OK?</i>		
	<input type="checkbox"/> ja	<input checked="" type="checkbox"/> nein	<input type="checkbox"/> nicht bewertet
Aktualisierung:	<i>Lassen sich die News aktualisieren?</i>		
	<input checked="" type="checkbox"/> ja	<input type="checkbox"/> nein	<input type="checkbox"/> nicht bewertet
Anhänge und Links:	<i>Werden anhänge und Links innerhalb der News angezeigt und funktionieren diese auch?</i>		
	<input type="checkbox"/> ja	<input checked="" type="checkbox"/> nein	<input type="checkbox"/> nicht bewertet
Sonstiges:	<i>Anmerkungen zum Modul</i>		
Hinweis:	<i>Der Test verlief weitestgehend erfolgreich. Formatierungsvorgaben wurden umgesetzt. Es gibt nur noch kleine Auffälligkeiten, welche die grundlegende Funktion jedoch nicht beeinflussen. Die Abfrage nach neuen News kann nicht bewertet werden, da die Funktion des</i>		

Aktualisierungsbuttons nicht „sichtbar“ ist.

Aufgefallene Punkte:

- *Es werden nicht alle News geladen*
- Links in einer News nicht anklickbar
- Bilder und weiterführende Links einer News werden nicht mit ausgegeben
- „Überschrift“ in der News ist „← Termindetails“

Aktualisierungsbutton könnte Hinweistext ausgeben

Protokollempfänger: Alle Projektteilnehmer

Datum: 31.05.2015

Testprotokoll News Modul

Projekt: HFTL-APP
Testobjekt: HfTL-App\app\Versionen\HFTL-APP_V0_2.apk
Prototyp-Version: 2
Protokoll-ID: 11
Testumgebung: Emuliertes Gerät Nexus 5 API 22 mit Android Studio
Datum: 10.06.2015
Tester: PK

Funktionstest:	<i>Funktion des News Modul testen</i>		
Vorschau News:	<i>Wird eine News-Vorschau angezeigt und ist die Formatierung usw. OK?</i>		
	<input checked="" type="checkbox"/> ja	<input type="checkbox"/> nein	<input type="checkbox"/> nicht bewertet
Anzeige News:	<i>Wird die News vollständig angezeigt und ist die Formatierung usw. OK?</i>		
	<input checked="" type="checkbox"/> ja	<input type="checkbox"/> nein	<input type="checkbox"/> nicht bewertet
Aktualisierung:	<i>Lassen sich die News aktualisieren?</i>		
	<input checked="" type="checkbox"/> ja	<input type="checkbox"/> nein	<input type="checkbox"/> nicht bewertet
Anhänge und Links:	<i>Werden anhänge und Links innerhalb der News angezeigt und funktionieren diese auch?</i>		
	<input type="checkbox"/> ja	<input checked="" type="checkbox"/> nein	<input type="checkbox"/> nicht bewertet
Sonstiges:	<i>Anmerkungen zum Modul</i>		
Hinweis:	<i>Kein Feedback bei aktualisieren der News.</i>		
	<i>Überschriften der News werden in der Vorschau nicht angezeigt, nur das Datum ist groß als Überschrift zu sehen.</i>		

Fehlermeldungen:	<i>Fehlermeldungen protokollieren</i>
Fehler:	<i>App stürzt ab</i>
Fehlerursache:	<i>Wenn die Datenverbindung getrennt ist sürzt die App ab.</i>
Fehler reproduzierbar:	<i>Lässt sich zu 100% reproduzieren</i>
Kritikalität:	<i>10/10 schwerer Fehler</i>

Protokollempfänger: Alle Projektteilnehmer

Datum: 10.06.2015

Testprotokoll News Modul

Projekt: HFTL-APP
Testobjekt: HfTL-App\app\Versionen\HFTL-APP_V0_3.apk
Prototyp-Version: 3
Protokoll-ID: 15
Testumgebung: Emuliertes Gerät Nexus 5 API 22 mit Android Studio
Datum: 23.06.2015
Tester: PK

Funktionstest:	<i>Funktion des News Modul testen</i>		
Vorschau News:	<i>Wird eine News-Vorschau angezeigt und ist die Formatierung usw. OK?</i>		
	<input checked="" type="checkbox"/> ja	<input type="checkbox"/> nein	<input type="checkbox"/> nicht bewertet
Anzeige News:	<i>Wird die News vollständig angezeigt und ist die Formatierung usw. OK?</i>		
	<input checked="" type="checkbox"/> ja	<input type="checkbox"/> nein	<input type="checkbox"/> nicht bewertet
Aktualisierung:	<i>Lassen sich die News aktualisieren?</i>		
	<input checked="" type="checkbox"/> ja	<input type="checkbox"/> nein	<input type="checkbox"/> nicht bewertet
Anhänge und Links:	<i>Werden anhänge und Links innerhalb der News angezeigt und funktionieren diese auch?</i>		
	<input type="checkbox"/> ja	<input checked="" type="checkbox"/> nein	<input type="checkbox"/> nicht bewertet
Sonstiges:	<i>Anmerkungen zum Modul</i>		
Hinweis:	<i>Kein Feedback bei aktualisieren der News.</i>		
	<i>Überschriften der News werden in der Vorschau nicht angezeigt, nur das Datum ist groß als Überschrift zu sehen</i>		

Keine änderungen zum letzten Test.

Fehlermeldungen:	<i>Fehlermeldungen protokollieren</i>
Fehler:	<i>App stürzt ab</i>
Fehlerursache:	<i>Wenn die Datenverbindung getrennt ist sürzt die App ab.</i>
Fehler reproduzierbar:	<i>Lässt sich zu 100% reproduzieren</i>
Kritikalität:	<i>10/10 schwerer Fehler</i>

Protokollempfänger: Alle Projektteilnehmer

Datum: 23.06.2015

Testprotokoll News Modul

Projekt: HFTL-APP
Testobjekt: HfTL-App\app\Versionen\HFTL-APP_V0_4.apk
Prototyp-Version: 4
Protokoll-ID: 20
Testumgebung: Emuliertes Gerät Nexus 5 API 22 mit Android Studio
Datum: 05.07.2015
Tester: PK

Funktionstest:	<i>Funktion des News Modul testen</i>		
Vorschau News:	<i>Wird eine News-Vorschau angezeigt und ist die Formatierung usw. OK?</i>		
	<input checked="" type="checkbox"/> ja	<input type="checkbox"/> nein	<input type="checkbox"/> nicht bewertet
Anzeige News:	<i>Wird die News vollständig angezeigt und ist die Formatierung usw. OK?</i>		
	<input type="checkbox"/> ja	<input checked="" type="checkbox"/> nein	<input type="checkbox"/> nicht bewertet
Aktualisierung:	<i>Lassen sich die News aktualisieren?</i>		
	<input type="checkbox"/> ja	<input checked="" type="checkbox"/> nein	<input type="checkbox"/> nicht bewertet
Anhänge und Links:	<i>Werden anhänge und Links innerhalb der News angezeigt und funktionieren diese auch?</i>		
	<input type="checkbox"/> ja	<input checked="" type="checkbox"/> nein	<input type="checkbox"/> nicht bewertet
Sonstiges:	<i>Anmerkungen zum Modul</i>		
Hinweis:	<i>Überschriften werden noch immer nicht angezeigt. Im Fenster in dem die Komplette News angezeigt wird steht oben Termindetails, dass muss geändert werden.</i>		

Fehlermeldungen:	<i>Fehlermeldungen protokollieren</i>
Fehler:	<i>App stürzt ab</i>
Fehlerursache:	<i>Die App stürzt beim laden der vollständigen News ohne Fehlermeldung ab. Die News Vorschau wird angezeigt. Gründe sind möglicherweise Änderungen an der HFTL Homepage</i>
Fehler reproduzierbar:	<i>Lässt sich zu 100% reproduzieren</i>
Kritikalität:	<i>10/10 schwerer Fehler</i>

Protokollempfänger: Alle Projektteilnehmer

Datum: 05.07.2015

Testprotokoll News Modul

Projekt: HFTL-APP
Testobjekt: HfTL-App\app\Versionen\HFTL-APP_V0_5.apk
Prototyp-Version: 5
Protokoll-ID: 23
Testumgebung: Emuliertes Gerät Nexus 5 API 22 mit Android Studio
Datum: 14.07.2015
Tester: PK

Funktionstest:	<i>Funktion des News Modul testen</i>		
Vorschau News:	<i>Wird eine News-Vorschau angezeigt und ist die Formatierung usw. OK?</i>		
	<input type="checkbox"/> ja	<input checked="" type="checkbox"/> nein	<input type="checkbox"/> nicht bewertet
Anzeige News:	<i>Wird die News vollständig angezeigt und ist die Formatierung usw. OK?</i>		
	<input type="checkbox"/> ja	<input checked="" type="checkbox"/> nein	<input type="checkbox"/> nicht bewertet
Aktualisierung:	<i>Lassen sich die News aktualisieren?</i>		
	<input type="checkbox"/> ja	<input checked="" type="checkbox"/> nein	<input type="checkbox"/> nicht bewertet
Anhänge und Links:	<i>Werden anhänge und Links innerhalb der News angezeigt und funktionieren diese auch?</i>		
	<input type="checkbox"/> ja	<input checked="" type="checkbox"/> nein	<input type="checkbox"/> nicht bewertet
Sonstiges:	<i>Anmerkungen zum Modul</i>		
Hinweis:			

Fehlermeldungen:	<i>Fehlermeldungen protokollieren</i>
Fehler:	<i>News werden nicht geladen oder nicht dargestellt</i>
Fehlerursache:	<i>unbekannt</i>
Fehler reproduzierbar:	<i>Lässt sich zu 100% reproduzieren</i>
Kritikalität:	<i>10/10 schwerer Fehler</i>

Protokollempfänger: Alle Projektteilnehmer

Datum: 14.07.2015

Testprotokoll News Modul

Projekt: HFTL-APP

Testobjekt: HfTL-App (Stand: 09.08.2015, 11:00Uhr)

Prototyp-Version: V05

Protokoll-ID: 28

Testumgebung: HTC One Android 5.0.2

Datum: 09.08.2015

Tester: ML

Funktionstest:

Funktion des News Modul testen

Vorschau News:

Wird eine News-Vorschau angezeigt und ist die Formatierung usw. OK?

ja nein nicht bewertet

Anzeige News:

Wird die News vollständig angezeigt und ist die Formatierung usw. OK?

ja nein nicht bewertet

Aktualisierung:

Lassen sich die News aktualisieren?

ja nein nicht bewertet

Anhänge und Links:

Werden anhänge und Links innerhalb der News angezeigt und funktionieren diese auch?

ja nein nicht bewertet

Sonstiges:

Anmerkungen zum Modul

Hinweis:

Das Layout entspricht der Designvorlage, Funktionell ist die Abfrage der News in Ordnung. Anhänge werden nicht angezeigt und die App stürzt ab, wenn man eine schwankende Internetverbindung hat.

Fehlermeldungen:	<i>Fehlermeldungen protokollieren</i>
Fehler:	<i>App stürzt ab</i>
Fehlerursache:	<i>Deaktivieren der Internetverbindung/ Abbruch der WLAN-Verbindung am Smartphone</i>
Fehler reproduzierbar:	<i>Wenn die Vorschauliste der News geladen ist, die Internetverbindung trennen und dann eine News versuchen zu öffnen.</i>
Kritikalität:	<i>05/10</i>

Protokollempfänger: Alle Projektteilnehmer

Datum: 09.08.2015

Testprotokoll News Modul

Projekt: HFTL-APP

Testobjekt: HfTL-App\app\Versionen\HFTL-APP_V0_6.apk

Prototyp-Version: 0.6

Protokoll-ID: 40

Testumgebung: Sony Xperia Z3 Android 5.1.1

Datum: 24.08.2015

Tester: JS

Funktionstest:

Funktion des News Modul testen

Vorschau News:

Wird eine News-Vorschau angezeigt und ist die Formatierung usw. OK?

ja nein nicht bewertet

Anzeige News:

Wird die News vollständig angezeigt und ist die Formatierung usw. OK?

ja nein nicht bewertet

Aktualisierung:

Lassen sich die News aktualisieren?

ja nein nicht bewertet

Anhänge und Links:

Werden anhänge und Links innerhalb der News angezeigt und funktionieren diese auch?

ja nein nicht bewertet

Sonstiges:

Links können nicht direkt aufgerufen werden

Hinweis:

Ist meiner Meinung nach aber zu vernachlässigen

Fehlermeldungen:	
Fehler:	<i>Links können nicht direkt aufgerufen werden</i>
Fehlerursache:	<i>Links werden beim Parsen wahrscheinlich nicht richtig deklariert</i>
Fehler reproduzierbar:	<i>ja</i>
Kritikalität:	<i>2</i>
Fehlermeldungen:	<i>Fehlermeldungen protokollieren</i>
Fehler:	<i>Beschreibung des Fehlers</i>
Fehlerursache:	<i>Bei welchen Eingaben /Bedienung ist der Fehler aufgetreten</i>
Fehler reproduzierbar:	<i>Lässt sich der Fehler zuverlässig reproduzieren?</i>
Kritikalität:	<i>Wie schwer ist der Fehler auf einer Skala von 1-10</i>

Protokollempfänger: Alle Projektteilnehmer

Datum: 24.08.2015

Testprotokoll News Modul

Projekt: HFTL-APP
Testobjekt: HfTL-App\app\Versionen\HFTL-APP_V0_6.apk
Prototyp-Version: 6
Protokoll-ID: 42
Testumgebung: Emuliertes Gerät Nexus 5 API 22 mit Android Studio.
Samsung Galaxie S3 mit Android 4.3
Datum: 19.08.2015
Tester: PK

Funktionstest:	<i>Funktion des News Modul testen</i>		
Vorschau News:	<i>Wird eine News-Vorschau angezeigt und ist die Formatierung usw. OK?</i>		
	<input checked="" type="checkbox"/> ja	<input type="checkbox"/> nein	<input type="checkbox"/> nicht bewertet
Anzeige News:	<i>Wird die News vollständig angezeigt und ist die Formatierung usw. OK?</i>		
	<input checked="" type="checkbox"/> ja	<input type="checkbox"/> nein	<input type="checkbox"/> nicht bewertet
Aktualisierung:	<i>Lassen sich die News aktualisieren?</i>		
	<input checked="" type="checkbox"/> ja	<input type="checkbox"/> nein	<input type="checkbox"/> nicht bewertet
Anhänge und Links:	<i>Werden anhänge und Links innerhalb der News angezeigt und funktionieren diese auch?</i>		
	<input type="checkbox"/> ja	<input checked="" type="checkbox"/> nein	<input type="checkbox"/> nicht bewertet
Sonstiges:	<i>Anmerkungen zum Modul</i>		
Hinweis:	<i>Werden z.B. gerade andere Aktionen der App ausgeführt wie z.B. das Aktualisieren der Noten so dauert das laden der News mitunter recht lange. Es gibt dabei kein Feedback und der Nutzer könnte denken die App sei eingefroren. Eventuell sollte man wie beim Stundenplan einen</i>		

Ladebildschirm einbauen, damit keine ungewollten eingaben oder Bildschirm Wechsel erfolgen, die die App zum Absturz bringen können. Das laden ist nach ein paar Sekunden erledigt und somit sollte dies keine Einschränkung auf die Benutzerfreundlichkeit haben.

Fehlermeldungen:

Fehlermeldungen protokollieren

Fehler:

App stürzt ab

Fehlerursache:

Bei anstoßen des Ladens der Noten und dem sofortigen Wechsel in die News. Siehe Logcat Auszug

Fehler reproduzierbar:

Lässt sich sporadisch reproduzieren.

Kritikalität:

5/10 mittlerer Fehler

Fehlermeldungen:

Fehlermeldungen protokollieren

Fehler:

App stürzt ab

Fehlerursache:

Beim Laden der vollständigen News. Ursache ist nicht erkennbar

Fehler reproduzierbar:

Fehler tritt selten auf.

Kritikalität:

5/10 mittlerer Fehler

Protokollempfänger: Alle Projektteilnehmer

Datum: 19.08.2015

Testprotokoll News Modul

Projekt: HFTL-APP
Testobjekt: HfTL-App\app\Versionen\HFTL-APP_V0_6.apk
Prototyp-Version: 6
Protokoll-ID: 56
Testumgebung: Samsung Galaxy S5 (Android 5.0.5)
Datum: 23.08.15
Tester: AD

Funktionstest:	<i>Funktion des News Modul testen</i>		
Vorschau News:	<i>Wird eine News-Vorschau angezeigt und ist die Formatierung usw. OK?</i>		
	<input checked="" type="checkbox"/> ja	<input type="checkbox"/> nein	<input type="checkbox"/> nicht bewertet
Anzeige News:	<i>Wird die News vollständig angezeigt und ist die Formatierung usw. OK?</i>		
	<input checked="" type="checkbox"/> ja	<input type="checkbox"/> nein	<input type="checkbox"/> nicht bewertet
Aktualisierung:	<i>Lassen sich die News aktualisieren?</i>		
	<input checked="" type="checkbox"/> ja	<input type="checkbox"/> nein	<input type="checkbox"/> nicht bewertet
Anhänge und Links:	<i>Werden anhänge und Links innerhalb der News angezeigt und funktionieren diese auch?</i>		
	<input type="checkbox"/> ja	<input checked="" type="checkbox"/> nein	<input type="checkbox"/> nicht bewertet
Sonstiges:	<i>Anmerkungen zum Modul</i>		
Hinweis:	<i>Ggf. eine Meldung, einen Ladebalken oder die News kurz Ausblenden nach dem Aktualisieren, da man ggf. nicht weiß ob er nun aktualisiert hat oder nicht.</i>		

Fehlermeldungen:	<i>Fehlermeldungen protokollieren</i>
Fehler:	<i>Absturz</i>
Fehlerursache:	<i>Wenn beim anklicken einer News die Internetverbindung nicht da ist stürzt die APP ab. -> Auffangen wie im Hauptteil.</i>
Fehler reproduzierbar:	<i>JA</i>
Kritikalität:	<i>5</i>
Fehlermeldungen:	<i>Fehlermeldungen protokollieren</i>
Fehler:	<i>Beschreibung des Fehlers</i>
Fehlerursache:	<i>Bei welchen eingaben /Bedienung ist der Fehler aufgetreten</i>
Fehler reproduzierbar:	<i>Lässt sich der Fehler zuverlässig reproduzieren?</i>
Kritikalität:	<i>Wie schwer ist der Fehler auf einer Skala von 1-10</i>

Protokollempfänger: Alle Projektteilnehmer

Datum: 23.08.15

Testprotokoll News Modul

Projekt: HFTL-APP
Testobjekt: HfTL-App\app\Versionen\ HFTL-APP_V0_6.apk
Prototyp-Version: 6
Protokoll-ID: 58
Testumgebung: Samsung Galaxy S5, Android 5.0
Datum: 06.09.2015
Tester: SC

Funktionstest:	<i>Funktion des News Modul testen</i>		
Vorschau News:	<i>Wird eine News-Vorschau angezeigt und ist die Formatierung usw. OK?</i>		
	<input checked="" type="checkbox"/> ja	<input type="checkbox"/> nein	<input type="checkbox"/> nicht bewertet
Anzeige News:	<i>Wird die News vollständig angezeigt und ist die Formatierung usw. OK?</i>		
	<input checked="" type="checkbox"/> ja	<input checked="" type="checkbox"/> nein	<input type="checkbox"/> nicht bewertet
Aktualisierung:	<i>Lassen sich die News aktualisieren?</i>		
	<input checked="" type="checkbox"/> ja	<input type="checkbox"/> nein	<input type="checkbox"/> nicht bewertet
Anhänge und Links:	<i>Werden anhänge und Links innerhalb der News angezeigt und funktionieren diese auch?</i>		
	<input type="checkbox"/> ja	<input checked="" type="checkbox"/> nein	<input type="checkbox"/> nicht bewertet
Sonstiges:	<i>Anmerkungen zum Modul</i>		
Hinweis:	<i>Es fehlt die Darstellung der Hyperlinks und E-Mail Links</i>		
	<i>[Fix ist bereits umgesetzt und wird mit kommender Version live gehen.]</i>		

Fehlermeldungen:	<i>HfTL-App angehalten</i>
Fehler:	<i>Absturz der App</i>
Fehlerursache:	<i>Ist beim Aufruf einer News keine Netzwerkverbindung aktiv, dann stürzt die App ab.</i>
Fehler reproduzierbar:	<i>Ja</i>
Kritikalität:	<i>6</i>
Fehlermeldungen:	<i>Fehlermeldungen protokollieren</i>
Fehler:	<i>Beschreibung des Fehlers</i>
Fehlerursache:	<i>Bei welchen eingaben /Bedienung ist der Fehler aufgetreten</i>
Fehler reproduzierbar:	<i>Lässt sich der Fehler zuverlässig reproduzieren?</i>
Kritikalität:	<i>Wie schwer ist der Fehler auf einer Skala von 1-10</i>

Protokollempfänger: Alle Projektteilnehmer

Datum: 06.09.2015

Testprotokoll News Modul

Projekt: HFTL-APP

Testobjekt: ...GitHub\HFTL-APP\HfTL-App\app\Versionen

Prototyp-Version: Version 7

Protokoll-ID: 63

Testumgebung: Android 4.1.2

Datum: 09.09.2015

Tester: SK

Funktionstest:

Funktion des News Modul testen

Vorschau News:

Wird eine News-Vorschau angezeigt und ist die Formatierung usw. OK?

ja nein nicht bewertet

Anzeige News:

Wird die News vollständig angezeigt und ist die Formatierung usw. OK?

ja nein nicht bewertet

Aktualisierung:

Lassen sich die News aktualisieren?

ja nein nicht bewertet

Anhänge und Links:

Werden anhänge und Links innerhalb der News angezeigt und funktionieren diese auch?

ja nein nicht bewertet

Sonstiges:

--KEINE--

Hinweis:

Lange Ladezeiten

Fehlermeldungen: --KEINE--

Fehler: --KEINE--

Fehlerursache: --KEINE--

Fehler reproduzierbar: --KEINE--

Kritikalität: --KEINE--

Fehlermeldungen: --KEINE--

Fehler: --KEINE--

Fehlerursache: --KEINE--

Fehler reproduzierbar: --KEINE--

Kritikalität: --KEINE--

Protokollempfänger: Alle Projektteilnehmer

Datum: 09.09.2015

Testprotokolle zum Noten Modul

Testprotokoll Noten Modul

Projekt: HFTL-APP
Testobjekt: HfTL-App (Stand: 31.05.2015, 14:30Uhr)
Prototyp-Version: V02
Protokoll-ID: 03
Testumgebung: HTC One Android 5.0.2
Datum: 31.05.2015
Tester: ML

Funktionstest:	<i>Funktion des Noten Modul testen (ohne Benutzerdaten)</i>		
Abfrage:	<i>Funktioniert die Abfrage der Noten fehlerfrei?</i>		
	<input checked="" type="checkbox"/> ja	<input type="checkbox"/> nein	<input type="checkbox"/> nicht bewertet
Intervallabfrage:	<i>Funktioniert das Abfragen der Noten mit einem voreigestellten Intervall?</i>		
	<input type="checkbox"/> ja	<input checked="" type="checkbox"/> nein	<input type="checkbox"/> nicht bewertet
Speicherung der Daten:	<i>Werden die Noten auf dem Gerät gespeichert?</i>		
	<input type="checkbox"/> ja	<input type="checkbox"/> nein	<input checked="" type="checkbox"/> nicht bewertet
Benachrichtigung:	<i>Funktionieren die Benachrichtigungen bei Änderungen?</i>		
	<input type="checkbox"/> ja	<input type="checkbox"/> nein	<input checked="" type="checkbox"/> nicht bewertet
Datenübertragung:	<i>Werden die Benutzerdaten verschlüsselt übertragen oder im Klartext?</i>		
	<input type="checkbox"/> ja	<input type="checkbox"/> nein	<input checked="" type="checkbox"/> nicht bewertet
Formatierung:	<i>Ist das Layout in Ordnung? (wird alles Lesbar dargestellt)</i>		
	<input type="checkbox"/> ja	<input checked="" type="checkbox"/> nein	<input type="checkbox"/> nicht bewertet

Einstellungen:	<i>Test der Einstellungen des Moduls</i>		
Eingaben:	<i>Können Benutzername und Passwort eingegeben werden ohne Fehler zu erzeugen?</i>		
	<input checked="" type="checkbox"/> ja	<input type="checkbox"/> nein	<input type="checkbox"/> nicht bewertet
Intervallabfrage:	<i>Lässt sich ein Abfrageintervall einstellen?</i>		
	<input type="checkbox"/> ja	<input checked="" type="checkbox"/> nein	<input type="checkbox"/> nicht bewertet
Benachrichtigungen:	<i>Lässt sich einstellen ob man bei Änderungen benachrichtigt werden möchte?</i>		
	<input type="checkbox"/> ja	<input checked="" type="checkbox"/> nein	<input type="checkbox"/> nicht bewertet
Länge Eingabefelder:	<i>Ist die Länge der Eingabefelder ausreichend?</i>		
	<input checked="" type="checkbox"/> ja	<input type="checkbox"/> nein	<input type="checkbox"/> nicht bewertet
Erlaubter Zeichensatz:	<i>Sind die erlaubten Zeichen ausreichend um z.B. ein sicheres Passwort zu garantieren?</i>		
	<input type="checkbox"/> ja	<input type="checkbox"/> nein	<input checked="" type="checkbox"/> nicht bewertet
Sonderzeichen:	<i>Sind Sonderzeichen erlaubt? Wenn ja welche und werden diese fehlerfrei akzeptiert?</i>		
	<input type="checkbox"/> ja	<input type="checkbox"/> nein	<input checked="" type="checkbox"/> nicht bewertet
	<i>Wenn ja:</i>		
Zu kurze Eingaben:	<i>Was ist bei kurzen bis sehr kurzen Eingaben? (z.B. nur ein Zeichen oder keines)</i>		
	<i>Bemerkung: -</i>		
Zu lange Eingaben:	<i>Was ist bei sehr langen Eingaben? (größer als der Variablenbereich)</i>		
	<i>Bemerkung: -</i>		
Nicht unterstützte Zeichen / Sonderzeichen:	<i>Was passiert bei der Verwendung von nicht unterstützten Zeichen und Sonderzeichen?</i>		
	<i>Bemerkung: -</i>		
Speicherung:	<i>Funktioniert das Speichern der Benutzerdaten fehlerfrei?</i>		
	<input type="checkbox"/> ja	<input checked="" type="checkbox"/> nein	<input type="checkbox"/> nicht bewertet

Verschlüsselung:

Werden Benutzername und Passwort verschlüsselt gespeichert?

ja nein nicht bewertet

Funktioniert die Verschlüsselung?

ja nein nicht bewertet

Anzeige Benutzerdaten:

Wird das Passwort in den Einstellungen im Klartext angezeigt?

ja nein nicht bewertet

Wird der Benutzername in den Einstellungen im Klartext angezeigt?

ja nein nicht bewertet

Sonstiges:

Anmerkungen zum Modul

Hinweis:

Erwartet wurde eine einfache Fehlermeldung, dass keine Daten eingetragen sind. Der Nutzer erhält eine Auswahloption die Daten sofort oder später einzutragen. Beide Optionen wurden getestet und funktionieren problemlos. Bricht man die Abfrage ab, kommt eine leere Seite, doch mit einem Klick auf „Aktualisieren“ kann die Abfrage neu „gestartet“ werden.

Verbesserungsvorschläge:

Nach abbrechen der Abfrage erscheint eine leere Seite. Zusätzlich zum Aktualisierungsbutton könnte noch ein Hinweistext oder ein Button ähnlich der Formatierungsvorlage erscheinen.

Fehlermeldungen:

Fehlermeldungen protokollieren

Fehler:

keine

Fehlerursache:

-

Fehler reproduzierbar:

-

Kritikalität: -

Protokollempfänger: Alle Projektteilnehmer

Datum: 31.05.2015

Testprotokoll Noten Modul

Projekt: HFTL-APP
Testobjekt: HfTL-App (Stand: 31.05.2015, 14:30Uhr)
Prototyp-Version: V02
Protokoll-ID: 04
Testumgebung: HTC One Android 5.0.2
Datum: 31.05.2015
Tester: ML

Funktionstest:	<i>Funktion des Noten Modul testen (mit Benutzerdaten)</i>		
Abfrage:	<i>Funktioniert die Abfrage der Noten fehlerfrei?</i>		
	<input checked="" type="checkbox"/> ja	<input type="checkbox"/> nein	<input type="checkbox"/> nicht bewertet
Intervallabfrage:	<i>Funktioniert das Abfragen der Noten mit einem voreigestellten Intervall?</i>		
	<input type="checkbox"/> ja	<input checked="" type="checkbox"/> nein	<input type="checkbox"/> nicht bewertet
Speicherung der Daten:	<i>Werden die Noten auf dem Gerät gespeichert?</i>		
	<input type="checkbox"/> ja	<input checked="" type="checkbox"/> nein	<input type="checkbox"/> nicht bewertet
Benachrichtigung:	<i>Funktionieren die Benachrichtigungen bei Änderungen?</i>		
	<input type="checkbox"/> ja	<input type="checkbox"/> nein	<input checked="" type="checkbox"/> nicht bewertet
Datenübertragung:	<i>Werden die Benutzerdaten verschlüsselt übertragen oder im Klartext?</i>		
	<input type="checkbox"/> ja	<input type="checkbox"/> nein	<input checked="" type="checkbox"/> nicht bewertet
Formatierung:	<i>Ist das Layout in Ordnung? (wird alles Lesbar dargestellt)</i>		
	<input checked="" type="checkbox"/> ja	<input type="checkbox"/> nein	<input type="checkbox"/> nicht bewertet

Einstellungen:	<i>Test der Einstellungen des Moduls</i>		
Eingaben:	<i>Können Benutzername und Passwort eingegeben werden ohne Fehler zu erzeugen?</i>		
	<input checked="" type="checkbox"/> ja	<input type="checkbox"/> nein	<input type="checkbox"/> nicht bewertet
Intervallabfrage:	<i>Lässt sich ein Abfrageintervall einstellen?</i>		
	<input type="checkbox"/> ja	<input checked="" type="checkbox"/> nein	<input type="checkbox"/> nicht bewertet
Benachrichtigungen:	<i>Lässt sich einstellen ob man bei Änderungen benachrichtigt werden möchte?</i>		
	<input type="checkbox"/> ja	<input checked="" type="checkbox"/> nein	<input type="checkbox"/> nicht bewertet
Länge Eingabefelder:	<i>Ist die Länge der Eingabefelder ausreichend?</i>		
	<input checked="" type="checkbox"/> ja	<input type="checkbox"/> nein	<input type="checkbox"/> nicht bewertet
Erlaubter Zeichensatz:	<i>Sind die erlaubten Zeichen ausreichend um z.B. ein sicheres Passwort zu garantieren?</i>		
	<input checked="" type="checkbox"/> ja	<input type="checkbox"/> nein	<input type="checkbox"/> nicht bewertet
Sonderzeichen:	<i>Sind Sonderzeichen erlaubt? Wenn ja welche und werden diese fehlerfrei akzeptiert?</i>		
	<input type="checkbox"/> ja	<input type="checkbox"/> nein	<input checked="" type="checkbox"/> nicht bewertet
	<i>Wenn ja:</i>		
Zu kurze Eingaben:	<i>Was ist bei kurzen bis sehr kurzen Eingaben? (z.B. nur ein Zeichen oder keines)</i>		
	<i>Bemerkung: -</i>		
Zu lange Eingaben:	<i>Was ist bei sehr langen Eingaben? (größer als der Variablenbereich)</i>		
	<i>Bemerkung: -</i>		
Nicht unterstützte Zeichen / Sonderzeichen:	<i>Was passiert bei der Verwendung von nicht unterstützten Zeichen und Sonderzeichen?</i>		
	<i>Bemerkung: -</i>		
Speicherung:	<i>Funktioniert das Speichern der Benutzerdaten fehlerfrei?</i>		
	<input checked="" type="checkbox"/> ja	<input type="checkbox"/> nein	<input type="checkbox"/> nicht bewertet

Verschlüsselung:

Werden Benutzername und Passwort verschlüsselt gespeichert?

ja nein nicht bewertet

Funktioniert die Verschlüsselung?

ja nein nicht bewertet

Anzeige Benutzerdaten:

Wird das Passwort in den Einstellungen im Klartext angezeigt?

ja nein nicht bewertet

Wird der Benutzername in den Einstellungen im Klartext angezeigt?

ja nein nicht bewertet

Sonstiges:

Anmerkungen zum Modul

Hinweis:

Nach Eingabe der Nutzerdaten lädt die App die Notenübersicht aus QIS problemlos. Formatierungen müssen noch angepasst werden, die Funktion ist jedoch erkennbar.

Fehlermeldungen:

Fehlermeldungen protokollieren

Fehler:

keine

Fehlerursache:

-

Fehler reproduzierbar:

-

Kritikalität:

-

Protokollempfänger: Alle Projektteilnehmer

Datum: 31.05.2015

Testprotokoll Noten Modul

Projekt: HFTL-APP
Testobjekt: HfTL-App\app\Versionen\HFTL-APP_V0_2.apk
Prototyp-Version: 2
Protokoll-ID: 12
Testumgebung: Emuliertes Gerät Nexus 5 API 22 mit Android Studio
Datum: 10.06.2015
Tester: PK

Funktionstest:	<i>Funktion des Noten Modul testen</i>		
Abfrage:	<i>Funktioniert die Abfrage der Noten fehlerfrei?</i>		
	<input checked="" type="checkbox"/> ja	<input type="checkbox"/> nein	<input type="checkbox"/> nicht bewertet
Intervallabfrage:	<i>Funktioniert das Abfragen der Noten mit einem voreigestellten Intervall?</i>		
	<input type="checkbox"/> ja	<input checked="" type="checkbox"/> nein	<input type="checkbox"/> nicht bewertet
Speicherung der Daten:	<i>Werden die Noten auf dem Gerät gespeichert?</i>		
	<input type="checkbox"/> ja	<input checked="" type="checkbox"/> nein	<input type="checkbox"/> nicht bewertet
Benachrichtigung:	<i>Funktionieren die Benachrichtigungen bei Änderungen?</i>		
	<input type="checkbox"/> ja	<input checked="" type="checkbox"/> nein	<input type="checkbox"/> nicht bewertet
Datenübertragung:	<i>Werden die Benutzerdaten verschlüsselt übertragen oder im Klartext?</i>		
	<input type="checkbox"/> ja	<input checked="" type="checkbox"/> nein	<input type="checkbox"/> nicht bewertet
Formatierung:	<i>Ist das Layout in Ordnung? (wird alles Lesbar dargestellt)</i>		
	<input type="checkbox"/> ja	<input type="checkbox"/> nein	<input checked="" type="checkbox"/> nicht bewertet

Einstellungen:	<i>Test der Einstellungen des Moduls</i>		
Eingaben:	<i>Können Benutzername und Passwort eingegeben werden ohne Fehler zu erzeugen?</i>		
	<input checked="" type="checkbox"/> ja	<input type="checkbox"/> nein	<input type="checkbox"/> nicht bewertet
Intervallabfrage:	<i>Lässt sich ein Abfrageintervall einstellen?</i>		
	<input type="checkbox"/> ja	<input checked="" type="checkbox"/> nein	<input type="checkbox"/> nicht bewertet
Benachrichtigungen:	<i>Lässt sich einstellen ob man bei Änderungen benachrichtigt werden möchte?</i>		
	<input type="checkbox"/> ja	<input checked="" type="checkbox"/> nein	<input type="checkbox"/> nicht bewertet
Länge Eingabefelder:	<i>Ist die Länge der Eingabefelder ausreichend?</i>		
	<input checked="" type="checkbox"/> ja	<input type="checkbox"/> nein	<input type="checkbox"/> nicht bewertet
Erlaubter Zeichensatz:	<i>Sind die erlaubten Zeichen ausreichend um z.B. ein sicheres Passwort zu garantieren?</i>		
	<input checked="" type="checkbox"/> ja	<input type="checkbox"/> nein	<input type="checkbox"/> nicht bewertet
Sonderzeichen:	<i>Sind Sonderzeichen sind erlaubt? Wenn ja welche und werden diese fehlerfrei akzeptiert?</i>		
	<input checked="" type="checkbox"/> ja	<input type="checkbox"/> nein	<input type="checkbox"/> nicht bewertet
Zu kurze Eingaben:	<i>Wenn ja: Es werden alle Sonderzeichen akzeptiert, unabhängig davon ob diese den Passwortrichtlinien von QIS entsprechen</i>		
	<i>Was ist bei kurzen bis sehr kurzen Eingaben? (z.B. nur ein Zeichen oder keines)</i>		
	Bemerkung: Kein Zeichen ist nicht möglich. Eingaben mit nur einem Zeichen werden akzeptiert.		
Zu lange Eingaben:	<i>Was ist bei sehr langen Eingaben? (größer als der Variablenbereich)</i>		
	Bemerkung: Es gibt keine Begrenzung der Eingabefelder		
Nicht unterstützte Zeichen / Sonderzeichen:	<i>Was passiert bei der Verwendung von nicht unterstützten Zeichen und Sonderzeichen?</i>		
	Bemerkung: soweit getestet werden alle Zeichen und Sonderzeichen akzeptiert		
Speicherung:	<i>Funktioniert das Speichern der Benutzerdaten fehlerfrei?</i>		

ja nein nicht bewertet

Verschlüsselung:

Werden Benutzername und Passwort verschlüsselt gespeichert?

ja nein nicht bewertet

Funktioniert die Verschlüsselung?

ja nein nicht bewertet

Anzeige Benutzerdaten:

Wird das Passwort in den Einstellungen im Klartext angezeigt?

ja nein nicht bewertet

Wird der Benutzername in den Einstellungen im Klartext angezeigt?

ja nein nicht bewertet

Sonstiges:

Anmerkungen zum Modul

Hinweis:

Das Layout ist noch nicht final, deswegen auch nicht bewertet. Formatierung stimmt noch nicht und auch die Reihenfolger der Noten ist weder geordnet noch lässt sie sich ordnen

Fehlermeldungen:

Fehlermeldungen protokollieren

Fehler:

App stürzt ab.

Fehlerursache:

Wenn keine datenverbindung vorhanden ist stürzt die app ab

Fehler reproduzierbar:

Zu 100% reproduzierbar

Kritikalität:

10/10 schwerer Fehler

Fehlermeldungen:	<i>Fehlermeldungen protokollieren</i>
Fehler:	<i>App stürzt ab</i>
Fehlerursache:	<i>Nicht zu erkennen. Ausgelöst vermutlich beim Wechsel zwischen den Seiten.</i>
Fehler reproduzierbar:	<i>Fehler tritt unregelmäßig beim aktualisieren oder anzeigen der Noten auf.</i>
Kritikalität:	<i>10/10 schwerer Fehler</i>

Protokollempfänger: Alle Projektteilnehmer

Datum: 10.06.2015

Testprotokoll Noten Modul

Projekt: HFTL-APP
Testobjekt: HfTL-App\app\Versionen\HFTL-APP_V0_3.apk
Prototyp-Version: 3
Protokoll-ID: 16
Testumgebung: Emuliertes Gerät Nexus 5 API 22 mit Android Studio
Datum: 23.06.2015
Tester: PK

Funktionstest:	<i>Funktion des Noten Modul testen</i>		
Abfrage:	<i>Funktioniert die Abfrage der Noten fehlerfrei?</i>		
	<input checked="" type="checkbox"/> ja	<input type="checkbox"/> nein	<input type="checkbox"/> nicht bewertet
Intervallabfrage:	<i>Funktioniert das Abfragen der Noten mit einem voreigestellten Intervall?</i>		
	<input type="checkbox"/> ja	<input checked="" type="checkbox"/> nein	<input type="checkbox"/> nicht bewertet
Speicherung der Daten:	<i>Werden die Noten auf dem Gerät gespeichert?</i>		
	<input checked="" type="checkbox"/> ja	<input type="checkbox"/> nein	<input type="checkbox"/> nicht bewertet
Benachrichtigung:	<i>Funktionieren die Benachrichtigungen bei Änderungen?</i>		
	<input type="checkbox"/> ja	<input checked="" type="checkbox"/> nein	<input type="checkbox"/> nicht bewertet
Datenübertragung:	<i>Werden die Benutzerdaten verschlüsselt übertragen?</i>		
	<input checked="" type="checkbox"/> ja	<input type="checkbox"/> nein	<input type="checkbox"/> nicht bewertet
Formatierung:	<i>Ist das Layout in Ordnung? (wird alles Lesbar dargestellt)</i>		
	<input type="checkbox"/> ja	<input checked="" type="checkbox"/> nein	<input type="checkbox"/> nicht bewertet

Einstellungen:	<i>Test der Einstellungen des Moduls</i>		
Eingaben:	<i>Können Benutzername und Passwort eingegeben werden ohne Fehler zu erzeugen?</i>		
	<input checked="" type="checkbox"/> ja	<input type="checkbox"/> nein	<input type="checkbox"/> nicht bewertet
Intervallabfrage:	<i>Lässt sich ein Abfrageintervall einstellen?</i>		
	<input type="checkbox"/> ja	<input checked="" type="checkbox"/> nein	<input type="checkbox"/> nicht bewertet
Benachrichtigungen:	<i>Lässt sich einstellen ob man bei Änderungen benachrichtigt werden möchte?</i>		
	<input type="checkbox"/> ja	<input checked="" type="checkbox"/> nein	<input type="checkbox"/> nicht bewertet
Länge Eingabefelder:	<i>Ist die Länge der Eingabefelder ausreichend?</i>		
	<input checked="" type="checkbox"/> ja	<input type="checkbox"/> nein	<input type="checkbox"/> nicht bewertet
Erlaubter Zeichensatz:	<i>Sind die erlaubten Zeichen ausreichend um z.B. ein sicheres Passwort zu garantieren?</i>		
	<input checked="" type="checkbox"/> ja	<input type="checkbox"/> nein	<input type="checkbox"/> nicht bewertet
Sonderzeichen:	<i>Sind Sonderzeichen sind erlaubt? Wenn ja welche und werden diese fehlerfrei akzeptiert?</i>		
	<input checked="" type="checkbox"/> ja	<input type="checkbox"/> nein	<input type="checkbox"/> nicht bewertet
Zu kurze Eingaben:	<i>Was ist bei kurzen bis sehr kurzen Eingaben? (z.B. nur ein Zeichen oder keines)</i>		
	Bemerkung: Kein Zeichen ist nicht möglich. Eingaben mit nur einem Zeichen werden akzeptiert.		
Zu lange Eingaben:	<i>Was ist bei sehr langen Eingaben? (größer als der Variablenbereich)</i>		
	Bemerkung: Es gibt keine Begrenzung der Eingabefelder		
Nicht unterstützte Zeichen / Sonderzeichen:	<i>Was passiert bei der Verwendung von nicht unterstützten Zeichen und Sonderzeichen?</i>		
	Bemerkung: soweit getestet werden alle Zeichen und Sonderzeichen akzeptiert		
Speicherung:	<i>Funktioniert das Speichern der Benutzerdaten fehlerfrei?</i>		

ja nein nicht bewertet

Verschlüsselung:

Werden Benutzername und Passwort verschlüsselt gespeichert?

ja nein nicht bewertet

Funktioniert die Verschlüsselung?

ja nein nicht bewertet

Anzeige Benutzerdaten:

Wird das Passwort in den Einstellungen im Klartext angezeigt?

ja nein nicht bewertet

Wird der Benutzername in den Einstellungen im Klartext angezeigt?

ja nein nicht bewertet

Sonstiges:

Anmerkungen zum Modul

Hinweis:

*Ausgabe der Noten unsortiert als Liste.
Sortierung der Noten nicht möglich.*

Fehlermeldungen:

Fehlermeldungen protokollieren

Fehler:

App stürzt ab

Fehlerursache:

Nicht zu erkennen. Ausgelöst vermutlich beim Wechsel zwischen den Seiten.

Fehler reproduzierbar:

Fehler tritt unregelmäßig beim aktualisieren oder anzeigen der Noten auf.

Kritikalität:

10/10 schwerer Fehler

Fehlermeldungen:	Fehlermeldungen protokollieren
Fehler:	App stürzt ab oder gibt falsche Fehlermeldung aus
Fehlerursache:	Bricht die Datenverbindung beim Anzeigen oder aktualisieren der Noten ab stürzt die App entweder ab oder es kommt die Fehlermeldung „Benutzername und Passwort falsch“
Fehler reproduzierbar:	Fehler zu 100% reproduzierbar
Kritikalität:	10/10 schwerer Fehler

Protokollempfänger: Alle Projektteilnehmer

Datum: 23.06.2015

Testprotokoll Noten Modul

Projekt: HFTL-APP
Testobjekt: HfTL-App (Stand: 28.06.2015, 15:00Uhr)
Prototyp-Version: V03
Protokoll-ID: 19
Testumgebung: HTC One Android 5.0.2
Datum: 28.06.2015
Tester: ML

Funktionstest:	<i>Funktion des Noten Modul testen</i>		
Abfrage:	<i>Funktioniert die Abfrage der Noten fehlerfrei?</i>		
	<input checked="" type="checkbox"/> ja	<input type="checkbox"/> nein	<input type="checkbox"/> nicht bewertet
Intervallabfrage:	<i>Funktioniert das Abfragen der Noten mit einem voreigestellten Intervall?</i>		
	<input type="checkbox"/> ja	<input type="checkbox"/> nein	<input checked="" type="checkbox"/> nicht bewertet
Speicherung der Daten:	<i>Werden die Noten auf dem Gerät gespeichert?</i>		
	<input type="checkbox"/> ja	<input type="checkbox"/> nein	<input checked="" type="checkbox"/> nicht bewertet
Benachrichtigung:	<i>Funktionieren die Benachrichtigungen bei Änderungen?</i>		
	<input type="checkbox"/> ja	<input type="checkbox"/> nein	<input checked="" type="checkbox"/> nicht bewertet
Datenübertragung:	<i>Werden die Benutzerdaten verschlüsselt übertragen oder im Klartext?</i>		
	<input type="checkbox"/> ja	<input type="checkbox"/> nein	<input checked="" type="checkbox"/> nicht bewertet
Formatierung:	<i>Ist das Layout in Ordnung? (wird alles Lesbar dargestellt)</i>		
	<input type="checkbox"/> ja	<input checked="" type="checkbox"/> nein	<input type="checkbox"/> nicht bewertet

Einstellungen:	<i>Test der Einstellungen des Moduls</i>		
Eingaben:	<i>Können Benutzername und Passwort eingegeben werden ohne Fehler zu erzeugen?</i>		
	<input type="checkbox"/> ja	<input type="checkbox"/> nein	<input checked="" type="checkbox"/> nicht bewertet
Intervallabfrage:	<i>Lässt sich ein Abfrageintervall einstellen?</i>		
	<input type="checkbox"/> ja	<input type="checkbox"/> nein	<input checked="" type="checkbox"/> nicht bewertet
Benachrichtigungen:	<i>Lässt sich einstellen ob man bei Änderungen benachrichtigt werden möchte?</i>		
	<input type="checkbox"/> ja	<input type="checkbox"/> nein	<input checked="" type="checkbox"/> nicht bewertet
Länge Eingabefelder:	<i>Ist die Länge der Eingabefelder ausreichend?</i>		
	<input type="checkbox"/> ja	<input type="checkbox"/> nein	<input checked="" type="checkbox"/> nicht bewertet
Erlaubter Zeichensatz:	<i>Sind die erlaubten Zeichen ausreichend um z.B. ein sicheres Passwort zu garantieren?</i>		
	<input type="checkbox"/> ja	<input type="checkbox"/> nein	<input checked="" type="checkbox"/> nicht bewertet
Sonderzeichen:	<i>Sind Sonderzeichen erlaubt? Wenn ja welche und werden diese fehlerfrei akzeptiert?</i>		
	<input type="checkbox"/> ja	<input type="checkbox"/> nein	<input checked="" type="checkbox"/> nicht bewertet
	<i>Wenn ja:</i>		
Zu kurze Eingaben:	<i>Was ist bei kurzen bis sehr kurzen Eingaben? (z.B. nur ein Zeichen oder keines)</i>		
	<i>Bemerkung:</i>		
Zu lange Eingaben:	<i>Was ist bei sehr langen Eingaben? (größer als der Variablenbereich)</i>		
	<i>Bemerkung:</i>		
Nicht unterstützte Zeichen / Sonderzeichen:	<i>Was passiert bei der Verwendung von nicht unterstützten Zeichen und Sonderzeichen?</i>		
	<i>Bemerkung:</i>		
Speicherung:	<i>Funktioniert das Speichern der Benutzerdaten fehlerfrei?</i>		
	<input type="checkbox"/> ja	<input type="checkbox"/> nein	<input checked="" type="checkbox"/> nicht bewertet

Verschlüsselung:

Werden Benutzername und Passwort verschlüsselt gespeichert?

ja nein nicht bewertet

Funktioniert die Verschlüsselung?

ja nein nicht bewertet

Anzeige Benutzerdaten:

Wird das Passwort in den Einstellungen im Klartext angezeigt?

ja nein nicht bewertet

Wird der Benutzername in den Einstellungen im Klartext angezeigt?

ja nein nicht bewertet

Sonstiges:

Anmerkungen zum Modul

Hinweis:

Das Modul wurde auf Korrektheit der Formatierungen geprüft. Änderungen in Schrift, Farben und Größen wurden angepasst. Das Layout ist nun an die Designvorlage angepasst und bietet soweit keine weiteren Fehler.

Protokollempfänger: Alle Projektteilnehmer

Datum: 29.06.2015

Testprotokoll Noten Modul

Projekt: HFTL-APP
Testobjekt: HfTL-App\app\Versionen\HFTL-APP_V0_4.apk
Prototyp-Version: 4
Protokoll-ID: 21
Testumgebung: Emuliertes Gerät Nexus 5 API 22 mit Android Studio
Datum: 05.07.2015
Tester: PK

Funktionstest:	<i>Funktion des Noten Modul testen</i>		
Abfrage:	<i>Funktioniert die Abfrage der Noten fehlerfrei?</i>		
	<input checked="" type="checkbox"/> ja	<input type="checkbox"/> nein	<input type="checkbox"/> nicht bewertet
Intervallabfrage:	<i>Funktioniert das Abfragen der Noten mit einem voreigestellten Intervall?</i>		
	<input type="checkbox"/> ja	<input checked="" type="checkbox"/> nein	<input type="checkbox"/> nicht bewertet
Speicherung der Daten:	<i>Werden die Noten auf dem Gerät gespeichert?</i>		
	<input checked="" type="checkbox"/> ja	<input type="checkbox"/> nein	<input type="checkbox"/> nicht bewertet
Benachrichtigung:	<i>Funktionieren die Benachrichtigungen bei Änderungen?</i>		
	<input type="checkbox"/> ja	<input checked="" type="checkbox"/> nein	<input type="checkbox"/> nicht bewertet
Datenübertragung:	<i>Werden die Benutzerdaten verschlüsselt übertragen oder im Klartext?</i>		
	<input checked="" type="checkbox"/> ja	<input type="checkbox"/> nein	<input type="checkbox"/> nicht bewertet
Formatierung:	<i>Ist das Layout in Ordnung? (wird alles Lesbar dargestellt)</i>		
	<input checked="" type="checkbox"/> ja	<input type="checkbox"/> nein	<input type="checkbox"/> nicht bewertet

Einstellungen:	<i>Test der Einstellungen des Moduls</i>		
Eingaben:	<i>Können Benutzername und Passwort eingegeben werden ohne Fehler zu erzeugen?</i>		
	<input checked="" type="checkbox"/> ja	<input type="checkbox"/> nein	<input type="checkbox"/> nicht bewertet
Intervallabfrage:	<i>Lässt sich ein Abfrageintervall einstellen?</i>		
	<input type="checkbox"/> ja	<input checked="" type="checkbox"/> nein	<input type="checkbox"/> nicht bewertet
Benachrichtigungen:	<i>Lässt sich einstellen ob man bei Änderungen benachrichtigt werden möchte?</i>		
	<input type="checkbox"/> ja	<input checked="" type="checkbox"/> nein	<input type="checkbox"/> nicht bewertet
Länge Eingabefelder:	<i>Ist die Länge der Eingabefelder ausreichend?</i>		
	<input checked="" type="checkbox"/> ja	<input type="checkbox"/> nein	<input type="checkbox"/> nicht bewertet
Erlaubter Zeichensatz:	<i>Sind die erlaubten Zeichen ausreichend um z.B. ein sicheres Passwort zu garantieren?</i>		
	<input checked="" type="checkbox"/> ja	<input type="checkbox"/> nein	<input type="checkbox"/> nicht bewertet
Sonderzeichen:	<i>Sind Sonderzeichen sind erlaubt? Wenn ja welche und werden diese fehlerfrei akzeptiert?</i>		
	<input checked="" type="checkbox"/> ja	<input type="checkbox"/> nein	<input type="checkbox"/> nicht bewertet
Zu kurze Eingaben:	<i>Was ist bei kurzen bis sehr kurzen Eingaben? (z.B. nur ein Zeichen oder keines)</i>		
	Bemerkung: Kein Zeichen ist nicht möglich. Eingaben mit nur einem Zeichen werden akzeptiert.		
Zu lange Eingaben:	<i>Was ist bei sehr langen Eingaben? (größer als der Variablenbereich)</i>		
	Bemerkung: Es gibt keine Begrenzung der Eingabefelder		
Nicht unterstützte Zeichen / Sonderzeichen:	<i>Was passiert bei der Verwendung von nicht unterstützten Zeichen und Sonderzeichen?</i>		
	Bemerkung: soweit getestet werden alle Zeichen und Sonderzeichen akzeptiert		
Speicherung:	<i>Funktioniert das Speichern der Benutzerdaten fehlerfrei?</i>		

ja nein nicht bewertet

Verschlüsselung:

Werden Benutzername und Passwort verschlüsselt gespeichert?

ja nein nicht bewertet

Funktioniert die Verschlüsselung?

ja nein nicht bewertet

Anzeige Benutzerdaten:

Wird das Passwort in den Einstellungen im Klartext angezeigt?

ja nein nicht bewertet

Wird der Benutzername in den Einstellungen im Klartext angezeigt?

ja nein nicht bewertet

Sonstiges:

Anmerkungen zum Modul

Hinweis:

Die Reihenfolge der Semester ist ungeordnet, was die Übersichtlichkeit stört. Formatierungen sind noch nicht Final aber werden korrekt dargestellt.

Fehlermeldungen:

Fehlermeldungen protokollieren

Fehler:

Die App gibt eine falsche Fehlermeldung aus

Fehlerursache:

Wenn die Datenverbindung beim laden der Noten abbricht wird die Fehlermeldung „Benutzername/Passwort falsch!“ angezeigt.

Fehler reproduzierbar:

Zu 100% reproduzierbar

Kritikalität:

5/10 mittlerer Fehler

Fehlermeldungen:	<i>Fehlermeldungen protokollieren</i>
Fehler:	<i>App stürzt ab</i>
Fehlerursache:	<i>Nicht zu erkennen. Ausgelöst vermutlich beim Wechsel zwischen den Seiten.</i>
Fehler reproduzierbar:	<i>Fehler tritt unregelmäßig beim aktualisieren oder anzeigen der Noten auf.</i>
Kritikalität:	<i>10/10 schwerer Fehler</i>

Protokollempfänger: Alle Projektteilnehmer

Datum: 05.07.2015

Testprotokoll Noten Modul

Projekt: HFTL-APP
Testobjekt: HfTL-App\app\Versionen\HFTL-APP_V0_5.apk
Prototyp-Version: 5
Protokoll-ID: 24
Testumgebung: Emuliertes Gerät Nexus 5 API 22 mit Android Studio
Datum: 14.07.2015
Tester: PK

Funktionstest:	<i>Funktion des Noten Modul testen</i>		
Abfrage:	<i>Funktioniert die Abfrage der Noten fehlerfrei?</i>		
	<input checked="" type="checkbox"/> ja	<input type="checkbox"/> nein	<input type="checkbox"/> nicht bewertet
Intervallabfrage:	<i>Funktioniert das Abfragen der Noten mit einem voreigestellten Intervall?</i>		
	<input type="checkbox"/> ja	<input type="checkbox"/> nein	<input checked="" type="checkbox"/> nicht bewertet
Speicherung der Daten:	<i>Werden die Noten auf dem Gerät gespeichert?</i>		
	<input checked="" type="checkbox"/> ja	<input type="checkbox"/> nein	<input type="checkbox"/> nicht bewertet
Benachrichtigung:	<i>Funktionieren die Benachrichtigungen bei Änderungen?</i>		
	<input type="checkbox"/> ja	<input type="checkbox"/> nein	<input checked="" type="checkbox"/> nicht bewertet
Datenübertragung:	<i>Werden die Benutzerdaten verschlüsselt übertragen oder im Klartext?</i>		
	<input checked="" type="checkbox"/> ja	<input type="checkbox"/> nein	<input type="checkbox"/> nicht bewertet
Formatierung:	<i>Ist das Layout in Ordnung? (wird alles Lesbar dargestellt)</i>		
	<input checked="" type="checkbox"/> ja	<input type="checkbox"/> nein	<input type="checkbox"/> nicht bewertet

Einstellungen:	<i>Test der Einstellungen des Moduls</i>		
Eingaben:	<i>Können Benutzername und Passwort eingegeben werden ohne Fehler zu erzeugen?</i>		
	<input checked="" type="checkbox"/> ja	<input type="checkbox"/> nein	<input type="checkbox"/> nicht bewertet
Intervallabfrage:	<i>Lässt sich ein Abfrageintervall einstellen?</i>		
	<input checked="" type="checkbox"/> ja	<input type="checkbox"/> nein	<input type="checkbox"/> nicht bewertet
Benachrichtigungen:	<i>Lässt sich einstellen ob man bei Änderungen benachrichtigt werden möchte?</i>		
	<input type="checkbox"/> ja	<input checked="" type="checkbox"/> nein	<input type="checkbox"/> nicht bewertet
Länge Eingabefelder:	<i>Ist die Länge der Eingabefelder ausreichend?</i>		
	<input checked="" type="checkbox"/> ja	<input type="checkbox"/> nein	<input type="checkbox"/> nicht bewertet
Erlaubter Zeichensatz:	<i>Sind die erlaubten Zeichen ausreichend um z.B. ein sicheres Passwort zu garantieren?</i>		
	<input checked="" type="checkbox"/> ja	<input type="checkbox"/> nein	<input type="checkbox"/> nicht bewertet
Sonderzeichen:	<i>Sind Sonderzeichen sind erlaubt? Wenn ja welche und werden diese fehlerfrei akzeptiert?</i>		
	<input checked="" type="checkbox"/> ja	<input type="checkbox"/> nein	<input type="checkbox"/> nicht bewertet
Zu kurze Eingaben:	<i>Wenn ja: Es werden alle Sonderzeichen akzeptiert, unabhängig davon ob diese den Passwortrichtlinien von QIS entsprechen</i>		
	<i>Was ist bei kurzen bis sehr kurzen Eingaben? (z.B. nur ein Zeichen oder keines)</i>		
	Bemerkung: Kein Zeichen ist nicht möglich. Eingaben mit nur einem Zeichen werden akzeptiert.		
Zu lange Eingaben:	<i>Was ist bei sehr langen Eingaben? (größer als der Variablenbereich)</i>		
	Bemerkung: Es gibt keine Begrenzung der Eingabefelder		
Nicht unterstützte Zeichen / Sonderzeichen:	<i>Was passiert bei der Verwendung von nicht unterstützten Zeichen und Sonderzeichen?</i>		
	Bemerkung: soweit getestet werden alle Zeichen und Sonderzeichen akzeptiert		
	<i>Funktioniert das Speichern der Benutzerdaten fehlerfrei?</i>		

Speicherung: ja nein nicht bewertet

Verschlüsselung: *Werden Benutzername und Passwort verschlüsselt gespeichert?*

ja nein nicht bewertet

Funktioniert die Verschlüsselung?

ja nein nicht bewertet

Anzeige Benutzerdaten: *Wird das Passwort in den Einstellungen im Klartext angezeigt?*

ja nein nicht bewertet

Wird der Benutzername in den Einstellungen im Klartext angezeigt?

ja nein nicht bewertet

Sonstiges: *Anmerkungen zum Modul*

Hinweis: *Die Benachrichtigung bei Änderungen der Noten konnte nicht getestet werden da keine Möglichkeit besteht diese in QIS anzupassen.
Die Noten könnten noch nach Semester sortierbar gemacht werden*

Fehlermeldungen: *Fehlermeldungen protokollieren*

Fehler: *App zeigt falsche Fehlermeldung an*

Fehlerursache: *Wenn die Datenverbindung beim laden der Noten abbricht wird als Fehlermeldung „Benutzername/Passwort falsch“ angezeigt.*

Fehler reproduzierbar: *Zu 100% reproduzierbar*

Kritikalität: *5/10 mittlerer Fehler*

Fehlermeldungen:	<i>Fehlermeldungen protokollieren</i>
Fehler:	<i>App stürzt ab</i>
Fehlerursache:	<i>Nicht zu erkennen. Ausgelöst vermutlich beim Wechsel zwischen den Seiten.</i>
Fehler reproduzierbar:	<i>Fehler tritt unregelmäßig beim aktualisieren oder anzeigen der Noten auf.</i>
Kritikalität:	<i>10/10 schwerer Fehler</i>

Protokollempfänger: Alle Projektteilnehmer

Datum: 10.06.2015

Testprotokoll Noten Modul

Projekt: HFTL-APP
Testobjekt: HfTL-App\app\Versionen\HFTL-APP_V0_6.apk
Prototyp-Version: 0.6
Protokoll-ID: 41
Testumgebung: Sony Xperia Z3 Android 5.1.1
Datum: 24.08.2015
Tester: JS

Funktionstest:	<i>Funktion des Noten Modul testen</i>		
Abfrage:	<i>Funktioniert die Abfrage der Noten fehlerfrei?</i>		
	<input checked="" type="checkbox"/> ja	<input type="checkbox"/> nein	<input type="checkbox"/> nicht bewertet
Intervallabfrage:	<i>Funktioniert das Abfragen der Noten mit einem voreigestellten Intervall?</i>		
	<input type="checkbox"/> ja	<input type="checkbox"/> nein	<input checked="" type="checkbox"/> nicht bewertet
Speicherung der Daten:	<i>Werden die Noten auf dem Gerät gespeichert?</i>		
	<input type="checkbox"/> ja	<input type="checkbox"/> nein	<input checked="" type="checkbox"/> nicht bewertet
Benachrichtigung:	<i>Funktionieren die Benachrichtigungen bei Änderungen?</i>		
	<input type="checkbox"/> ja	<input checked="" type="checkbox"/> nein	<input type="checkbox"/> nicht bewertet
Datenübertragung:	<i>Werden die Benutzerdaten verschlüsselt übertragen oder im Klartext?</i>		
	<input type="checkbox"/> ja	<input type="checkbox"/> nein	<input checked="" type="checkbox"/> nicht bewertet
Formatierung:	<i>Ist das Layout in Ordnung? (wird alles Lesbar dargestellt)</i>		
	<input checked="" type="checkbox"/> ja	<input type="checkbox"/> nein	<input type="checkbox"/> nicht bewertet

Einstellungen:	<i>Test der Einstellungen des Moduls</i>		
Eingaben:	<i>Können Benutzername und Passwort eingegeben werden ohne Fehler zu erzeugen?</i>		
	<input checked="" type="checkbox"/> ja	<input type="checkbox"/> nein	<input type="checkbox"/> nicht bewertet
Intervallabfrage:	<i>Lässt sich ein Abfrageintervall einstellen?</i>		
	<input checked="" type="checkbox"/> ja	<input type="checkbox"/> nein	<input type="checkbox"/> nicht bewertet
Benachrichtigungen:	<i>Lässt sich einstellen ob man bei Änderungen benachrichtigt werden möchte?</i>		
	<input checked="" type="checkbox"/> ja	<input type="checkbox"/> nein	<input type="checkbox"/> nicht bewertet
Länge Eingabefelder:	<i>Ist die Länge der Eingabefelder ausreichend?</i>		
	<input checked="" type="checkbox"/> ja	<input type="checkbox"/> nein	<input type="checkbox"/> nicht bewertet
Erlaubter Zeichensatz:	<i>Sind die erlaubten Zeichen ausreichend um z.B. ein sicheres Passwort zu garantieren?</i>		
	<input type="checkbox"/> ja	<input type="checkbox"/> nein	<input checked="" type="checkbox"/> nicht bewertet
Sonderzeichen:	<i>Sind Sonderzeichen sind erlaubt? Wenn ja welche und werden diese fehlerfrei akzeptiert?</i>		
	<input checked="" type="checkbox"/> ja	<input type="checkbox"/> nein	<input type="checkbox"/> nicht bewertet
	<i>Wenn ja: #</i>		
Zu kurze Eingaben:	<i>Was ist bei kurzen bis sehr kurzen Eingaben? (z.B. nur ein Zeichen oder keines)</i>		
	<i>Bemerkung:</i>		
Zu lange Eingaben:	<i>Was ist bei sehr langen Eingaben? (größer als der Variablenbereich)</i>		
	<i>Bemerkung:</i>		
Nicht unterstützte Zeichen / Sonderzeichen:	<i>Was passiert bei der Verwendung von nicht unterstützten Zeichen und Sonderzeichen?</i>		
	<i>Bemerkung:</i>		
Speicherung:	<i>Funktioniert das Speichern der Benutzerdaten fehlerfrei?</i>		
	<input checked="" type="checkbox"/> ja	<input type="checkbox"/> nein	<input type="checkbox"/> nicht bewertet

Verschlüsselung:

Werden Benutzername und Passwort verschlüsselt gespeichert?

ja nein nicht bewertet

Funktioniert die Verschlüsselung?

ja nein nicht bewertet

Anzeige Benutzerdaten:

Wird das Passworts in den Einstellungen im Klartext angezeigt?

ja nein nicht bewertet

Wird der Benutzername in den Einstellungen im Klartext angezeigt?

ja nein nicht bewertet

Sonstiges:

%

Hinweis:

%

Fehlermeldungen:

%

Fehler:

%

Fehlerursache:

%

Fehler reproduzierbar:

%

Kritikalität:

%

Fehlermeldungen:

Fehlermeldungen protokollieren

Fehler:

Beschreibung des Fehlers

Fehlerursache:

Bei welchen eingaben /Bedienung ist der Fehler aufgetreten

Fehler reproduzierbar:

Lässt sich der Fehler zuverlässig reproduzieren?

Kritikalität:

Wie schwer ist der Fehler auf einer Skala von 1-10

Protokollempfänger: Alle Projektteilnehmer

Datum: 24.08.2015

Testprotokoll Noten Modul

Projekt: HFTL-APP
Testobjekt: HfTL-App\app\Versionen\HFTL-APP_V0_6.apk
Prototyp-Version: 6
Protokoll-ID: 43
Testumgebung: Emuliertes Gerät Nexus 5 API 22 mit Android Studio.
Samsung Galaxie S3 mit Android 4.3
Datum: 19.08.2015
Tester: PK

Funktionstest:	<i>Funktion des Noten Modul testen</i>		
Abfrage:	<i>Funktioniert die Abfrage der Noten fehlerfrei?</i>		
	<input checked="" type="checkbox"/> ja	<input type="checkbox"/> nein	<input type="checkbox"/> nicht bewertet
Intervallabfrage:	<i>Funktioniert das Abfragen der Noten mit einem voreigestellten Intervall?</i>		
	<input checked="" type="checkbox"/> ja	<input type="checkbox"/> nein	<input type="checkbox"/> nicht bewertet
Speicherung der Daten:	<i>Werden die Noten auf dem Gerät gespeichert?</i>		
	<input checked="" type="checkbox"/> ja	<input type="checkbox"/> nein	<input type="checkbox"/> nicht bewertet
Benachrichtigung:	<i>Funktionieren die Benachrichtigungen bei Änderungen?</i>		
	<input type="checkbox"/> ja	<input type="checkbox"/> nein	<input checked="" type="checkbox"/> nicht bewertet
Datenübertragung:	<i>Werden die Benutzerdaten verschlüsselt übertragen oder im Klartext?</i>		
	<input checked="" type="checkbox"/> ja	<input type="checkbox"/> nein	<input type="checkbox"/> nicht bewertet
Formatierung:	<i>Ist das Layout in Ordnung? (wird alles Lesbar dargestellt)</i>		
	<input checked="" type="checkbox"/> ja	<input type="checkbox"/> nein	<input type="checkbox"/> nicht bewertet

Einstellungen:	<i>Test der Einstellungen des Moduls</i>		
Eingaben:	<i>Können Benutzername und Passwort eingegeben werden ohne Fehler zu erzeugen?</i>		
	<input checked="" type="checkbox"/> ja	<input type="checkbox"/> nein	<input type="checkbox"/> nicht bewertet
Intervallabfrage:	<i>Lässt sich ein Abfrageintervall einstellen?</i>		
	<input checked="" type="checkbox"/> ja	<input type="checkbox"/> nein	<input type="checkbox"/> nicht bewertet
Benachrichtigungen:	<i>Lässt sich einstellen ob man bei Änderungen benachrichtigt werden möchte?</i>		
	<input checked="" type="checkbox"/> ja	<input type="checkbox"/> nein	<input type="checkbox"/> nicht bewertet
Länge Eingabefelder:	<i>Ist die Länge der Eingabefelder ausreichend?</i>		
	<input checked="" type="checkbox"/> ja	<input type="checkbox"/> nein	<input type="checkbox"/> nicht bewertet
Erlaubter Zeichensatz:	<i>Sind die erlaubten Zeichen ausreichend um z.B. ein sicheres Passwort zu garantieren?</i>		
	<input checked="" type="checkbox"/> ja	<input type="checkbox"/> nein	<input type="checkbox"/> nicht bewertet
Sonderzeichen:	<i>Sind Sonderzeichen sind erlaubt? Wenn ja welche und werden diese fehlerfrei akzeptiert?</i>		
	<input checked="" type="checkbox"/> ja	<input type="checkbox"/> nein	<input type="checkbox"/> nicht bewertet
Zu kurze Eingaben:	<i>Wenn ja: Es werden alle Sonderzeichen akzeptiert, unabhängig davon ob diese den Passwortrichtlinien von QIS entsprechen</i>		
	<i>Was ist bei kurzen bis sehr kurzen Eingaben? (z.B. nur ein Zeichen oder keines)</i>		
	Bemerkung: Kein Zeichen ist nicht möglich. Eingaben mit nur einem Zeichen werden akzeptiert.		
Zu lange Eingaben:	<i>Was ist bei sehr langen Eingaben? (größer als der Variablenbereich)</i>		
	Bemerkung: Es gibt keine Begrenzung der Eingabefelder		
Nicht unterstützte Zeichen / Sonderzeichen:	<i>Was passiert bei der Verwendung von nicht unterstützten Zeichen und Sonderzeichen?</i>		
	Bemerkung: soweit getestet werden alle Zeichen und Sonderzeichen akzeptiert		
	<i>Funktioniert das Speichern der Benutzerdaten fehlerfrei?</i>		

Speicherung: ja nein nicht bewertet

Verschlüsselung: *Werden Benutzername und Passwort verschlüsselt gespeichert?*

ja nein nicht bewertet

Funktioniert die Verschlüsselung?

ja nein nicht bewertet

Anzeige Benutzerdaten: *Wird das Passwort in den Einstellungen im Klartext angezeigt?*

ja nein nicht bewertet

Wird der Benutzername in den Einstellungen im Klartext angezeigt?

ja nein nicht bewertet

Sonstiges: *Anmerkungen zum Modul*

Hinweis: *Die Benachrichtigung bei Änderungen der Noten konnte nicht getestet werden da keine Möglichkeit besteht diese in QIS anzupassen.*

Fehlermeldungen: *Fehlermeldungen protokollieren*

Fehler: *App zeigt falsche Fehlermeldung an*

Fehlerursache: *Wenn die Datenverbindung beim laden der Noten abbricht wird als Fehlermeldung „Benutzername/Passwort falsch“ angezeigt.*

Fehler reproduzierbar: *Zu 100% reproduzierbar wenn die Datenverbindung beim übertragen der Benutzerdaten zu QIS abbricht.*

Kritikalität: *3/10 leichter Fehler*

Protokollempfänger: Alle Projektteilnehmer

Datum: 19.08.2015

Testprotokoll Noten Modul

Projekt: HFTL-APP
Testobjekt: HfTL-App\app\Versionen\HFTL-APP_V0_6.apk
Prototyp-Version: 6
Protokoll-ID: 57
Testumgebung: Samsung Galaxy S5 (Android 5.0.5)
Datum: 23.08.2015
Tester: AD

Funktionstest:	<i>Funktion des Noten Modul testen</i>		
Abfrage:	<i>Funktioniert die Abfrage der Noten fehlerfrei?</i>		
	<input checked="" type="checkbox"/> ja	<input type="checkbox"/> nein	<input type="checkbox"/> nicht bewertet
Intervallabfrage:	<i>Funktioniert das Abfragen der Noten mit einem voreigestellten Intervall?</i>		
	<input checked="" type="checkbox"/> ja	<input type="checkbox"/> nein	<input type="checkbox"/> nicht bewertet
Speicherung der Daten:	<i>Werden die Noten auf dem Gerät gespeichert?</i>		
	<input checked="" type="checkbox"/> ja	<input type="checkbox"/> nein	<input type="checkbox"/> nicht bewertet
Benachrichtigung:	<i>Funktionieren die Benachrichtigungen bei Änderungen?</i>		
	<input checked="" type="checkbox"/> ja	<input type="checkbox"/> nein	<input type="checkbox"/> nicht bewertet
Datenübertragung:	<i>Werden die Benutzerdaten verschlüsselt übertragen oder im Klartext?</i>		
	<input checked="" type="checkbox"/> ja	<input type="checkbox"/> nein	<input type="checkbox"/> nicht bewertet
Formatierung:	<i>Ist das Layout in Ordnung? (wird alles Lesbar dargestellt)</i>		
	<input checked="" type="checkbox"/> ja	<input type="checkbox"/> nein	<input type="checkbox"/> nicht bewertet

Einstellungen:	<i>Test der Einstellungen des Moduls</i>		
Eingaben:	<i>Können Benutzername und Passwort eingegeben werden ohne Fehler zu erzeugen?</i>		
	<input checked="" type="checkbox"/> ja	<input type="checkbox"/> nein	<input type="checkbox"/> nicht bewertet
Intervallabfrage:	<i>Lässt sich ein Abfrageintervall einstellen?</i>		
	<input checked="" type="checkbox"/> ja	<input type="checkbox"/> nein	<input type="checkbox"/> nicht bewertet
Benachrichtigungen:	<i>Lässt sich einstellen ob man bei Änderungen benachrichtigt werden möchte?</i>		
	<input checked="" type="checkbox"/> ja	<input type="checkbox"/> nein	<input type="checkbox"/> nicht bewertet
Länge Eingabefelder:	<i>Ist die Länge der Eingabefelder ausreichend?</i>		
	<input checked="" type="checkbox"/> ja	<input type="checkbox"/> nein	<input type="checkbox"/> nicht bewertet
Erlaubter Zeichensatz:	<i>Sind die erlaubten Zeichen ausreichend um z.B. ein sicheres Passwort zu garantieren?</i>		
	<input checked="" type="checkbox"/> ja	<input type="checkbox"/> nein	<input type="checkbox"/> nicht bewertet
Sonderzeichen:	<i>Sind Sonderzeichen erlaubt? Wenn ja welche und werden diese fehlerfrei akzeptiert?</i>		
	<input checked="" type="checkbox"/> ja	<input type="checkbox"/> nein	<input type="checkbox"/> nicht bewertet
	<i>Wenn ja: alle</i>		
Zu kurze Eingaben:	<i>Was ist bei kurzen bis sehr kurzen Eingaben? (z.B. nur ein Zeichen oder keines)</i>		
	Bemerkung: ab einem Zeichen wird das Passwort akzeptiert		
Zu lange Eingaben:	<i>Was ist bei sehr langen Eingaben? (größer als der Variablenbereich)</i>		
	Bemerkung: keine Begrenzung vorhanden		
Nicht unterstützte Zeichen / Sonderzeichen:	<i>Was passiert bei der Verwendung von nicht unterstützten Zeichen und Sonderzeichen?</i>		
	Bemerkung: alle Sonderzeichen werden akzeptiert		
Speicherung:	<i>Funktioniert das Speichern der Benutzerdaten fehlerfrei?</i>		
	<input checked="" type="checkbox"/> ja	<input type="checkbox"/> nein	<input type="checkbox"/> nicht bewertet

Verschlüsselung:	<p><i>Werden Benutzername und Passwort verschlüsselt gespeichert?</i></p> <p><input checked="" type="checkbox"/> ja <input type="checkbox"/> nein <input type="checkbox"/> nicht bewertet</p>
	<p><i>Funktioniert die Verschlüsselung?</i></p> <p><input checked="" type="checkbox"/> ja <input type="checkbox"/> nein <input type="checkbox"/> nicht bewertet</p>
Anzeige Benutzerdaten:	<p><i>Wird das Passwort in den Einstellungen im Klartext angezeigt?</i></p> <p><input type="checkbox"/> ja <input checked="" type="checkbox"/> nein <input type="checkbox"/> nicht bewertet</p>
	<p><i>Wird der Benutzername in den Einstellungen im Klartext angezeigt?</i></p> <p><input checked="" type="checkbox"/> ja <input type="checkbox"/> nein <input type="checkbox"/> nicht bewertet</p>

Sonstiges:	<p><i>Anmerkungen zum Modul</i></p>
Hinweis:	<p><i>z.B. langsames Gerät beim nutzen des Moduls oder dergleichen</i></p>

Fehlermeldungen:	<p><i>Fehlermeldungen protokollieren</i></p>
Fehler:	<p><i>Beschreibung des Fehlers</i></p>
Fehlerursache:	<p><i>Bei welchen eingaben /Bedienung ist der Fehler aufgetreten</i></p>
Fehler reproduzierbar:	<p><i>Lässt sich der Fehler zuverlässig reproduzieren?</i></p>
Kritikalität:	<p><i>Wie schwer ist der Fehler auf einer Skala von 1-10</i></p>

Fehlermeldungen:	<p><i>Fehlermeldungen protokollieren</i></p>
Fehler:	<p><i>Beschreibung des Fehlers</i></p>
Fehlerursache:	<p><i>Bei welchen eingaben /Bedienung ist der Fehler aufgetreten</i></p>

Fehler reproduzierbar:

Lässt sich der Fehler zuverlässig reproduzieren?

Kritikalität:

Wie schwer ist der Fehler auf einer Skala von 1-10

Protokollempfänger: Alle Projektteilnehmer

Datum: 23.08.2015

Testprotokoll Noten Modul

Projekt: HFTL-APP
Testobjekt: HfTL-App\app\Versionen\HFTL-APP_V0_6.apk
Prototyp-Version: 6
Protokoll-ID: 59
Testumgebung: Samsung Galaxy S5, Android 5.0
Datum: 06.09.2015
Tester: SC

Funktionstest:	<i>Funktion des Noten Modul testen</i>		
Abfrage:	<i>Funktioniert die Abfrage der Noten fehlerfrei?</i>		
	<input checked="" type="checkbox"/> ja	<input type="checkbox"/> nein	<input type="checkbox"/> nicht bewertet
Intervallabfrage:	<i>Funktioniert das Abfragen der Noten mit einem voreigestellten Intervall?</i>		
	<input type="checkbox"/> ja	<input type="checkbox"/> nein	<input checked="" type="checkbox"/> nicht bewertet
Speicherung der Daten:	<i>Werden die Noten auf dem Gerät gespeichert?</i>		
	<input type="checkbox"/> ja	<input type="checkbox"/> nein	<input type="checkbox"/> nicht bewertet
Benachrichtigung:	<i>Funktionieren die Benachrichtigungen bei Änderungen?</i>		
	<input type="checkbox"/> ja	<input type="checkbox"/> nein	<input checked="" type="checkbox"/> nicht bewertet
Datenübertragung:	<i>Werden die Benutzerdaten verschlüsselt übertragen oder im Klartext?</i>		
	<input type="checkbox"/> ja	<input type="checkbox"/> nein	<input checked="" type="checkbox"/> nicht bewertet
Formatierung:	<i>Ist das Layout in Ordnung? (wird alles Lesbar dargestellt)</i>		
	<input checked="" type="checkbox"/> ja	<input type="checkbox"/> nein	<input type="checkbox"/> nicht bewertet

Einstellungen:	<i>Test der Einstellungen des Moduls</i>		
Eingaben:	<i>Können Benutzername und Passwort eingegeben werden ohne Fehler zu erzeugen?</i>		
	<input checked="" type="checkbox"/> ja	<input type="checkbox"/> nein	<input type="checkbox"/> nicht bewertet
Intervallabfrage:	<i>Lässt sich ein Abfrageintervall einstellen?</i>		
	<input checked="" type="checkbox"/> ja	<input type="checkbox"/> nein	<input type="checkbox"/> nicht bewertet
Benachrichtigungen:	<i>Lässt sich einstellen ob man bei Änderungen benachrichtigt werden möchte?</i>		
	<input checked="" type="checkbox"/> ja	<input type="checkbox"/> nein	<input type="checkbox"/> nicht bewertet
Länge Eingabefelder:	<i>Ist die Länge der Eingabefelder ausreichend?</i>		
	<input checked="" type="checkbox"/> ja	<input type="checkbox"/> nein	<input type="checkbox"/> nicht bewertet
Erlaubter Zeichensatz:	<i>Sind die erlaubten Zeichen ausreichend um z.B. ein sicheres Passwort zu garantieren?</i>		
	<input checked="" type="checkbox"/> ja	<input type="checkbox"/> nein	<input type="checkbox"/> nicht bewertet
Sonderzeichen:	<i>Sind Sonderzeichen erlaubt? Wenn ja welche und werden diese fehlerfrei akzeptiert?</i>		
	<input checked="" type="checkbox"/> ja	<input type="checkbox"/> nein	<input type="checkbox"/> nicht bewertet
	<i>Wenn ja: Keine Einschränkung festgestellt</i>		
Zu kurze Eingaben:	<i>Was ist bei kurzen bis sehr kurzen Eingaben? (z.B. nur ein Zeichen oder keines)</i>		
	<i>Bemerkung: Zeichenketten beliebiger Länge werden akzeptiert</i>		
Zu lange Eingaben:	<i>Was ist bei sehr langen Eingaben? (größer als der Variablenbereich)</i>		
	<i>Bemerkung: s. oben</i>		
Nicht unterstützte Zeichen / Sonderzeichen:	<i>Was passiert bei der Verwendung von nicht unterstützten Zeichen und Sonderzeichen?</i>		
	<i>Bemerkung: keine nicht unterstützte Zeichen festgestellt</i>		
Speicherung:	<i>Funktioniert das Speichern der Benutzerdaten fehlerfrei?</i>		
	<input checked="" type="checkbox"/> ja	<input type="checkbox"/> nein	<input type="checkbox"/> nicht bewertet

Verschlüsselung:

Werden Benutzername und Passwort verschlüsselt gespeichert?

ja nein nicht bewertet

Funktioniert die Verschlüsselung?

ja nein nicht bewertet

Anzeige Benutzerdaten:

Wird das Passworts in den Einstellungen im Klartext angezeigt?

ja nein nicht bewertet

Wird der Benutzername in den Einstellungen im Klartext angezeigt?

ja nein nicht bewertet

Sonstiges:

keine

Hinweis:

Fehlermeldungen:

Keine Fehler festgestellt

Fehler:

Fehlerursache:

Fehler reproduzierbar:

Kritikalität:

Fehlermeldungen:

Fehler:

Fehlerursache:

Fehler reproduzierbar:

Lässt sich der Fehler zuverlässig reproduzieren?

Kritikalität:

Wie schwer ist der Fehler auf einer Skala von 1-10

Protokollempfänger: Alle Projektteilnehmer

Datum: 06.09.2015

Testprotokoll Noten Modul

Projekt: HFTL-APP

Testobjekt:

Prototyp-Version: 7

Protokoll-ID: 64

Testumgebung: Android 4.1.2

Datum: 09.09.2015

Tester: SK

Funktionstest:

Funktion des Noten Modul testen

Abfrage:

Funktioniert die Abfrage der Noten fehlerfrei?

ja nein nicht bewertet

Intervallabfrage:

Funktioniert das Abfragen der Noten mit einem voreigestellten Intervall?

ja nein nicht bewertet

Speicherung der Daten:

Werden die Noten auf dem Gerät gespeichert?

ja nein nicht bewertet

Benachrichtigung:

Funktionieren die Benachrichtigungen bei Änderungen?

ja nein nicht bewertet

Datenübertragung:

Werden die Benutzerdaten verschlüsselt übertragen oder im Klartext?

ja nein nicht bewertet

Formatierung:

Ist das Layout in Ordnung? (wird alles Lesbar dargestellt)

ja nein nicht bewertet

Einstellungen:	<i>Test der Einstellungen des Moduls</i>		
Eingaben:	<i>Können Benutzername und Passwort eingegeben werden ohne Fehler zu erzeugen?</i>		
	<input checked="" type="checkbox"/> ja	<input type="checkbox"/> nein	<input type="checkbox"/> nicht bewertet
Intervallabfrage:	<i>Lässt sich ein Abfrageintervall einstellen?</i>		
	<input checked="" type="checkbox"/> ja	<input type="checkbox"/> nein	<input type="checkbox"/> nicht bewertet
Benachrichtigungen:	<i>Lässt sich einstellen ob man bei Änderungen benachrichtigt werden möchte?</i>		
	<input checked="" type="checkbox"/> ja	<input type="checkbox"/> nein	<input type="checkbox"/> nicht bewertet
Länge Eingabefelder:	<i>Ist die Länge der Eingabefelder ausreichend?</i>		
	<input checked="" type="checkbox"/> ja	<input type="checkbox"/> nein	<input type="checkbox"/> nicht bewertet
Erlaubter Zeichensatz:	<i>Sind die erlaubten Zeichen ausreichend um z.B. ein sicheres Passwort zu garantieren?</i>		
	<input checked="" type="checkbox"/> ja	<input type="checkbox"/> nein	<input type="checkbox"/> nicht bewertet
Sonderzeichen:	<i>Sind Sonderzeichen erlaubt? Wenn ja welche und werden diese fehlerfrei akzeptiert?</i>		
	<input type="checkbox"/> ja	<input type="checkbox"/> nein	<input checked="" type="checkbox"/> nicht bewertet
	<i>Wenn ja:</i>		
Zu kurze Eingaben:	<i>Was ist bei kurzen bis sehr kurzen Eingaben? (z.B. nur ein Zeichen oder keines)</i>		
	Bemerkung: --KEINE--		
Zu lange Eingaben:	<i>Was ist bei sehr langen Eingaben? (größer als der Variablenbereich)</i>		
	Bemerkung: --KEINE--		
Nicht unterstützte Zeichen / Sonderzeichen:	<i>Was passiert bei der Verwendung von nicht unterstützten Zeichen und Sonderzeichen?</i>		
	Bemerkung: --KEINE--		
Speicherung:	<i>Funktioniert das Speichern der Benutzerdaten fehlerfrei?</i>		
	<input checked="" type="checkbox"/> ja	<input type="checkbox"/> nein	<input type="checkbox"/> nicht bewertet

Verschlüsselung:

Werden Benutzername und Passwort verschlüsselt gespeichert?

ja nein nicht bewertet

Funktioniert die Verschlüsselung?

ja nein nicht bewertet

Anzeige Benutzerdaten:

Wird das Passworts in den Einstellungen im Klartext angezeigt?

ja nein nicht bewertet

Wird der Benutzername in den Einstellungen im Klartext angezeigt?

ja nein nicht bewertet

Sonstiges:

--KEINE--

Hinweis:

Lange Ladezeiten

Fehlermeldungen:

--KEINE--

Fehler:

--KEINE--

Fehlerursache:

--KEINE--

Fehler reproduzierbar:

--KEINE--

Kritikalität:

--KEINE--

Fehlermeldungen:

--KEINE--

Fehler:

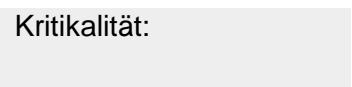
--KEINE--

Fehlerursache:

--KEINE--

Fehler reproduzierbar:

--KEINE--

Kritikalität:  *Wie schwer ist der Fehler auf einer Skala von 1-10*

Protokollempfänger: Alle Projektteilnehmer

Datum: xx.xx.xxxx

Testprotokolle zum Stundenplan Modul

Testprotokoll Stundenplan Modul

Projekt: HFTL-APP
Testobjekt: HfTL-App\app\Versionen\HFTL-APP_V0_2.apk
Prototyp-Version: 2
Protokoll-ID: 13
Testumgebung: Emuliertes Gerät Nexus 5 API 22 mit Android Studio
Datum: 10.06.2015
Tester: PK

Funktionstest:	<i>Funktion des Stundenplan Modul testen</i>		
Abfrage:	<i>Funktioniert die Abfrage des Stundenplans fehlerfrei?</i>		
	<input type="checkbox"/> ja	<input type="checkbox"/> nein	<input checked="" type="checkbox"/> nicht bewertet
Intervallabfrage:	<i>Funktioniert das Abfragen des Stundenplans mit einem voreigestellten Intervall?</i>		
	<input type="checkbox"/> ja	<input type="checkbox"/> nein	<input checked="" type="checkbox"/> nicht bewertet
Synchronisierung:	<i>Funktioniert die Synchronisierung mit dem Kalender des Smartphones?</i>		
	<input type="checkbox"/> ja	<input type="checkbox"/> nein	<input checked="" type="checkbox"/> nicht bewertet
Studiengang:	<i>Werden nur zum Nutzer passende Stundenpläne angezeigt?</i>		
	<input type="checkbox"/> ja	<input type="checkbox"/> nein	<input checked="" type="checkbox"/> nicht bewertet
Formatierung:	<i>Ist das Layout in Ordnung? (wird alles lesbar dargestellt)</i>		
	<input type="checkbox"/> ja	<input type="checkbox"/> nein	<input checked="" type="checkbox"/> nicht bewertet

Einstellungen:	<i>Test der Einstellungen des Moduls</i>
Wochenwahl:	<i>Funktioniert die Anzeige verschiedener Wochen?</i> <input type="checkbox"/> ja <input type="checkbox"/> nein <input checked="" type="checkbox"/> nicht bewertet
Intervallabfrage:	<i>Lässt sich ein Abfrageintervall einstellen?</i> <input type="checkbox"/> ja <input type="checkbox"/> nein <input checked="" type="checkbox"/> nicht bewertet
Benachrichtigungen:	<i>Funktionieren die Benachrichtigungen bei sich ändernden Stundenplänen usw.?</i> <input type="checkbox"/> ja <input type="checkbox"/> nein <input checked="" type="checkbox"/> nicht bewertet
Sonstiges:	<i>Anmerkungen zum Modul</i>
Hinweis:	<i>Modul noch nicht implementiert. Test deswegen nicht möglich</i>
Fehlermeldungen:	<i>Fehlermeldungen protokollieren</i>
Fehler:	<i>Beschreibung des Fehlers</i>
Fehlerursache:	<i>Bei welchen eingaben /Bedienung ist der Fehler aufgetreten</i>
Fehler reproduzierbar:	<i>Lässt sich der Fehler zuverlässig reproduzieren?</i>
Kritikalität:	<i>Wie schwer ist der Fehler auf einer Skala von 1-10</i>
Fehlermeldungen:	<i>Fehlermeldungen protokollieren</i>
Fehler:	<i>Beschreibung des Fehlers</i>
Fehlerursache:	<i>Bei welchen eingaben /Bedienung ist der Fehler aufgetreten</i>
Fehler reproduzierbar:	<i>Lässt sich der Fehler zuverlässig reproduzieren?</i>
Kritikalität:	<i>Wie schwer ist der Fehler auf einer Skala von 1-10</i>

Protokollempfänger: Alle Projektteilnehmer

Datum: 10.06.2015

Testprotokoll Stundenplan Modul

Projekt: HFTL-APP

Testobjekt: HfTL-App (*Stand: 20.06.2015, 12:00Uhr*)

Prototyp-Version: V02

Protokoll-ID: 14

Testumgebung: HTC One Android 5.0.2

Datum: 20.06.2015

Tester: ML

Funktionstest: *Funktion des Stundenplan Modul testen*

Abfrage: *Funktioniert die Abfrage des Stundenplans fehlerfrei?*

ja nein nicht bewertet

Intervallabfrage: *Funktioniert das Abfragen des Stundenplans mit einem voreigestellten Intervall?*

ja nein nicht bewertet

Synchronisierung: *Funktioniert die Synchronisierung mit dem Kalender des Smartphones?*

ja nein nicht bewertet

Studiengang: *Werden nur zum Nutzer passende Stundenpläne angezeigt?*

ja nein nicht bewertet

Formatierung: *Ist das Layout in Ordnung? (wird alles lesbar dargestellt)*

ja nein nicht bewertet

Einstellungen:	<i>Test der Einstellungen des Moduls</i>
Wochenwahl:	<i>Funktioniert die Anzeige verschiedener Wochen?</i> <input checked="" type="checkbox"/> ja <input type="checkbox"/> nein <input type="checkbox"/> nicht bewertet
Intervallabfrage:	<i>Lässt sich ein Abfrageintervall einstellen?</i> <input type="checkbox"/> ja <input checked="" type="checkbox"/> nein <input type="checkbox"/> nicht bewertet
Benachrichtigungen:	<i>Funktionieren die Benachrichtigungen bei sich ändernden Stundenplänen usw.?</i> <input type="checkbox"/> ja <input checked="" type="checkbox"/> nein <input type="checkbox"/> nicht bewertet
Sonstiges:	<i>Anmerkungen zum Modul</i>
Hinweis:	<i>Die grundlegende Funktion ist erkennbar. Die Abfrage der Stundenpläne ist erst mit ausgewähltem Studiengang möglich. Ohne diesen stürzt die App ab. Layout und Formatierungen müssen noch erfolgen.</i>
Fehlermeldungen:	<i>Fehlermeldungen protokollieren</i>
Fehler:	<i>App stürzt ab</i>
Fehlerursache:	<i>Öffnen des Reiters „Stundenplan“ ohne ausgewählten Studiengang</i>
Fehler reproduzierbar:	<i>Keinen Studiengang auswählen</i>
Kritikalität:	<i>7/10</i>

Protokollempfänger: Alle Projektteilnehmer

Datum: 21.06.2015

Testprotokoll Stundenplan Modul

Projekt: HFTL-APP
Testobjekt: HfTL-App\app\Versionen\HFTL-APP_V0_3.apk
Prototyp-Version: 3
Protokoll-ID: 17
Testumgebung: Emuliertes Gerät Nexus 5 API 22 mit Android Studio
Datum: 23.06.2015
Tester: PK

Funktionstest:	<i>Funktion des Stundenplan Modul testen</i>		
Abfrage:	<i>Funktioniert die Abfrage des Stundenplans fehlerfrei?</i>		
	<input checked="" type="checkbox"/> ja	<input type="checkbox"/> nein	<input type="checkbox"/> nicht bewertet
Intervallabfrage:	<i>Funktioniert das Abfragen des Stundenplans mit einem voreigestellten Intervall?</i>		
	<input type="checkbox"/> ja	<input checked="" type="checkbox"/> nein	<input type="checkbox"/> nicht bewertet
Synchronisierung:	<i>Funktioniert die Synchronisierung mit dem Kalender des Smartphones?</i>		
	<input type="checkbox"/> ja	<input checked="" type="checkbox"/> nein	<input type="checkbox"/> nicht bewertet
Studiengang:	<i>Werden nur zum Nutzer passende Stundenpläne angezeigt?</i>		
	<input checked="" type="checkbox"/> ja	<input type="checkbox"/> nein	<input type="checkbox"/> nicht bewertet
Formatierung:	<i>Ist das Layout in Ordnung? (wird alles lesbar dargestellt)</i>		
	<input type="checkbox"/> ja	<input checked="" type="checkbox"/> nein	<input type="checkbox"/> nicht bewertet

Einstellungen:	<i>Test der Einstellungen des Moduls</i>
Wochenwahl:	<i>Funktioniert die Anzeige verschiedener Wochen?</i> <input type="checkbox"/> ja <input checked="" type="checkbox"/> nein <input type="checkbox"/> nicht bewertet
Intervallabfrage:	<i>Lässt sich ein Abfrageintervall einstellen?</i> <input type="checkbox"/> ja <input checked="" type="checkbox"/> nein <input type="checkbox"/> nicht bewertet
Benachrichtigungen:	<i>Funktionieren die Benachrichtigungen bei sich ändernden Stundenplänen usw.?</i> <input type="checkbox"/> ja <input checked="" type="checkbox"/> nein <input type="checkbox"/> nicht bewertet
Sonstiges:	<i>Anmerkungen zum Modul</i>
Hinweis:	<i>Stundenplan wird als unsortierte Liste ausgegeben.</i>
Fehlermeldungen:	<i>Fehlermeldungen protokollieren</i>
Fehler:	<i>Beschreibung des Fehlers</i>
Fehlerursache:	<i>Bei welchen eingaben /Bedienung ist der Fehler aufgetreten</i>
Fehler reproduzierbar:	<i>Lässt sich der Fehler zuverlässig reproduzieren?</i>
Kritikalität:	<i>Wie schwer ist der Fehler auf einer Skala von 1-10</i>
Fehlermeldungen:	<i>Fehlermeldungen protokollieren</i>
Fehler:	<i>Beschreibung des Fehlers</i>
Fehlerursache:	<i>Bei welchen eingaben /Bedienung ist der Fehler aufgetreten</i>
Fehler reproduzierbar:	<i>Lässt sich der Fehler zuverlässig reproduzieren?</i>
Kritikalität:	<i>Wie schwer ist der Fehler auf einer Skala von 1-10</i>

Protokollempfänger: Alle Projektteilnehmer

Datum: 23.06.2015

Testprotokoll Stundenplan Modul

Projekt: HFTL-APP
Testobjekt: HfTL-App\app\Versionen\HFTL-APP_V0_4.apk
Prototyp-Version: 4
Protokoll-ID: 22
Testumgebung: Emuliertes Gerät Nexus 5 API 22 mit Android Studio
Datum: 05.07.2015
Tester: PK

Funktionstest:	<i>Funktion des Stundenplan Modul testen</i>		
Abfrage:	<i>Funktioniert die Abfrage des Stundenplans fehlerfrei?</i>		
	<input checked="" type="checkbox"/> ja	<input type="checkbox"/> nein	<input type="checkbox"/> nicht bewertet
Intervallabfrage:	<i>Funktioniert das Abfragen des Stundenplans mit einem voreigestellten Intervall?</i>		
	<input type="checkbox"/> ja	<input checked="" type="checkbox"/> nein	<input type="checkbox"/> nicht bewertet
Synchronisierung:	<i>Funktioniert die Synchronisierung mit dem Kalender des Smartphones?</i>		
	<input type="checkbox"/> ja	<input checked="" type="checkbox"/> nein	<input type="checkbox"/> nicht bewertet
Studiengang:	<i>Werden nur zum Nutzer passende Stundenpläne angezeigt?</i>		
	<input checked="" type="checkbox"/> ja	<input type="checkbox"/> nein	<input type="checkbox"/> nicht bewertet
Formatierung:	<i>Ist das Layout in Ordnung? (wird alles lesbar dargestellt)</i>		
	<input checked="" type="checkbox"/> ja	<input type="checkbox"/> nein	<input type="checkbox"/> nicht bewertet

Einstellungen:	<i>Test der Einstellungen des Moduls</i>
Wochenwahl:	<i>Funktioniert die Anzeige verschiedener Wochen?</i> <input checked="" type="checkbox"/> ja <input type="checkbox"/> nein <input type="checkbox"/> nicht bewertet
Intervallabfrage:	<i>Lässt sich ein Abfrageintervall einstellen?</i> <input type="checkbox"/> ja <input checked="" type="checkbox"/> nein <input type="checkbox"/> nicht bewertet
Benachrichtigungen:	<i>Funktionieren die Benachrichtigungen bei sich ändernden Stundenplänen usw.?</i> <input type="checkbox"/> ja <input checked="" type="checkbox"/> nein <input type="checkbox"/> nicht bewertet
Sonstiges:	<i>Anmerkungen zum Modul</i>
Hinweis:	<i>Das Layout ist noch nicht final aber die Formatierung ist ok Wird eine bestimmte Woche ausgewählt wird der entsprechende Stundenplan nicht automatisch geladen, sondern muss noch mit dem GO Knopf gestartet werden. Das automatische laden wäre hier sinnvoll</i>
Fehlermeldungen:	<i>Fehlermeldungen protokollieren</i>
Fehler:	<i>App stürzt ab</i>
Fehlerursache:	<i>Besteht keine Datenverbindung stürzt die App beim laden der Stundenpläne sofort ohne Fehlermeldung ab.</i>
Fehler reproduzierbar:	<i>Lässt sich zu 100% reproduzieren</i>
Kritikalität:	<i>10/10 schwerer Fehler</i>

Fehlermeldungen:	<i>Fehlermeldungen protokollieren</i>
Fehler:	<i>App stürzt ab</i>
Fehlerursache:	<i>Bricht die Datenverbindung während des ladens von Stundenplänen ab so stürzt die App sofort ohne Fehlermeldung ab</i>
Fehler reproduzierbar:	<i>Lässt sich zu 100% reproduzieren</i>
Kritikalität:	<i>10/10 schwerer Fehler</i>

Protokollempfänger: Alle Projektteilnehmer

Datum: 05.07.2015

Testprotokoll Stundenplan Modul

Projekt: HFTL-APP
Testobjekt: HfTL-App\app\Versionen\HFTL-APP_V0_5.apk
Prototyp-Version: 5
Protokoll-ID: 25
Testumgebung: Emuliertes Gerät Nexus 5 API 22 mit Android Studio
Datum: 14.07.2015
Tester: PK

Funktionstest:	<i>Funktion des Stundenplan Modul testen</i>		
Abfrage:	<i>Funktioniert die Abfrage des Stundenplans fehlerfrei?</i>		
	<input checked="" type="checkbox"/> ja	<input type="checkbox"/> nein	<input type="checkbox"/> nicht bewertet
Intervallabfrage:	<i>Funktioniert das Abfragen des Stundenplans mit einem voreigestellten Intervall?</i>		
	<input type="checkbox"/> ja	<input checked="" type="checkbox"/> nein	<input type="checkbox"/> nicht bewertet
Synchronisierung:	<i>Funktioniert die Synchronisierung mit dem Kalender des Smartphones?</i>		
	<input type="checkbox"/> ja	<input checked="" type="checkbox"/> nein	<input type="checkbox"/> nicht bewertet
Studiengang:	<i>Werden nur zum Nutzer passende Stundenpläne angezeigt?</i>		
	<input checked="" type="checkbox"/> ja	<input type="checkbox"/> nein	<input type="checkbox"/> nicht bewertet
Formatierung:	<i>Ist das Layout in Ordnung? (wird alles lesbar dargestellt)</i>		
	<input checked="" type="checkbox"/> ja	<input type="checkbox"/> nein	<input type="checkbox"/> nicht bewertet

Einstellungen:	<i>Test der Einstellungen des Moduls</i>
Wochenwahl:	<i>Funktioniert die Anzeige verschiedener Wochen?</i> <input checked="" type="checkbox"/> ja <input type="checkbox"/> nein <input type="checkbox"/> nicht bewertet
Intervallabfrage:	<i>Lässt sich ein Abfrageintervall einstellen?</i> <input type="checkbox"/> ja <input checked="" type="checkbox"/> nein <input type="checkbox"/> nicht bewertet
Benachrichtigungen:	<i>Funktionieren die Benachrichtigungen bei sich ändernden Stundenplänen usw.?</i> <input type="checkbox"/> ja <input checked="" type="checkbox"/> nein <input type="checkbox"/> nicht bewertet
Sonstiges:	<i>Anmerkungen zum Modul</i>
Hinweis:	
Fehlermeldungen:	<i>Fehlermeldungen protokollieren</i>
Fehler:	<i>App stürzt ab</i>
Fehlerursache:	<i>Bricht die Datenverbindung beim laden des Stundenplans ab stürzt die App ohne Fehlermeldung ab</i>
Fehler reproduzierbar:	<i>Zu 100% reproduzierbar</i>
Kritikalität:	<i>10/10 schwerer Fehler</i>
Fehlermeldungen:	<i>Fehlermeldungen protokollieren</i>
Fehler:	<i>Absturz der App</i>
Fehlerursache:	<i>Die App stürzt ab wenn in den Einstellungen kein Studiengang ausgewählt ist</i>
Fehler reproduzierbar:	<i>Fehler lässt sich zu 100% reproduzieren</i>
Kritikalität:	<i>Sehr schwerer Fehler 10/10</i>

Protokollempfänger: Alle Projektteilnehmer

Datum: 14.07.2015

Testprotokoll Stundenplan Modul

Projekt: HFTL-APP

Testobjekt: HfTL-App (Stand: 14.07.2015, 12:30Uhr)

Prototyp-Version: V03

Protokoll-ID: 26

Testumgebung: HTC One Android 5.0.2

Datum: 14.07.2015

Tester: ML

Funktionstest: *Funktion des Stundenplan Modul testen*

Abfrage: *Funktioniert die Abfrage des Stundenplans fehlerfrei?*

ja nein nicht bewertet

Intervallabfrage: *Funktioniert das Abfragen des Stundenplans mit einem voreigestellten Intervall?*

ja nein nicht bewertet

Synchronisierung: *Funktioniert die Synchronisierung mit dem Kalender des Smartphones?*

ja nein nicht bewertet

Studiengang: *Werden nur zum Nutzer passende Stundenpläne angezeigt?*

ja nein nicht bewertet

Formatierung: *Ist das Layout in Ordnung? (wird alles lesbar dargestellt)*

ja nein nicht bewertet

Einstellungen:	<i>Test der Einstellungen des Moduls</i>
Wochenwahl:	<i>Funktioniert die Anzeige verschiedener Wochen?</i> <input checked="" type="checkbox"/> ja <input type="checkbox"/> nein <input type="checkbox"/> nicht bewertet
Intervallabfrage:	<i>Lässt sich ein Abfrageintervall einstellen?</i> <input type="checkbox"/> ja <input checked="" type="checkbox"/> nein <input type="checkbox"/> nicht bewertet
Benachrichtigungen:	<i>Funktionieren die Benachrichtigungen bei sich ändernden Stundenplänen usw.?</i> <input type="checkbox"/> ja <input checked="" type="checkbox"/> nein <input type="checkbox"/> nicht bewertet
Sonstiges:	<i>Anmerkungen zum Modul</i>
Hinweis:	<i>Die Funktion des Stundenplans ist erkennbar. Das Layout ist noch nicht an die Designvorlage angepasst. Außerdem erfolgt noch ein Absturz, wenn man keinen Studiengang ausgewählt hat.</i>
Fehlermeldungen:	<i>Fehlermeldungen protokollieren</i>
Fehler:	<i>Absturz der App</i>
Fehlerursache:	<i>Es ist kein Studiengang ausgewählt</i>
Fehler reproduzierbar:	<i>Keinen Studiengang wählen und dann den Stundenplan öffnen</i>
Kritikalität:	<i>8/10</i>

Protokollempfänger: Alle Projektteilnehmer

Datum: 14.07.2015

Testprotokoll Stundenplan Modul

Projekt: HFTL-APP

Testobjekt: HfTL-App (*Stand: 09.08.2015, 11:00Uhr*)

Prototyp-Version: V05

Protokoll-ID: 27

Testumgebung: HTC One Android 5.0.2

Datum: 09.08.2015

Tester: ML

Funktionstest: *Funktion des Stundenplan Modul testen*

Abfrage: *Funktioniert die Abfrage des Stundenplans fehlerfrei?*

ja nein nicht bewertet

Intervallabfrage: *Funktioniert das Abfragen des Stundenplans mit einem voreigestellten Intervall?*

ja nein nicht bewertet

Synchronisierung: *Funktioniert die Synchronisierung mit dem Kalender des Smartphones?*

ja nein nicht bewertet

Studiengang: *Werden nur zum Nutzer passende Stundenpläne angezeigt?*

ja nein nicht bewertet

Formatierung: *Ist das Layout in Ordnung? (wird alles lesbar dargestellt)*

ja nein nicht bewertet

Einstellungen:	<i>Test der Einstellungen des Moduls</i>
Wochenwahl:	<i>Funktioniert die Anzeige verschiedener Wochen?</i> <input checked="" type="checkbox"/> ja <input type="checkbox"/> nein <input type="checkbox"/> nicht bewertet
Intervallabfrage:	<i>Lässt sich ein Abfrageintervall einstellen?</i> <input type="checkbox"/> ja <input checked="" type="checkbox"/> nein <input type="checkbox"/> nicht bewertet
Benachrichtigungen:	<i>Funktionieren die Benachrichtigungen bei sich ändernden Stundenplänen usw.?</i> <input type="checkbox"/> ja <input type="checkbox"/> nein <input checked="" type="checkbox"/> nicht bewertet
Sonstiges:	<i>Anmerkungen zum Modul</i>
Hinweis:	<i>Das Layout wurde weiter an die Designvorlage angepasst. Kleine Änderungen in Schrift und Absätzen notwendig. Weiterhin Abbrüche, wenn kein Studiengang ausgewählt wurde.</i>
Fehlermeldungen:	<i>Fehlermeldungen protokollieren</i>
Fehler:	<i>App stürzt ab</i>
Fehlerursache:	<i>Keinen Studiengang ausgewählen und dann den Stundenplan öffnen</i>
Fehler reproduzierbar:	<i>Keinen Studiengang auswählen, dann Stundenplan öffnen</i>
Kritikalität:	<i>8/10</i>

Protokollempfänger: Alle Projektteilnehmer

Datum: 10.08.2015

Testprotokoll Stundenplan Modul

Projekt: HFTL-APP
Testobjekt: HfTL-App\app\Versionen\HFTL-APP_V0_6.apk
Prototyp-Version: 0.6
Protokoll-ID: 39
Testumgebung: Sony Xperia Z3 Android 5.1.1
Datum: 24.08.15
Tester: JS

Funktionstest:	<i>Funktion des Stundenplan Modul testen</i>		
Abfrage:	<i>Funktioniert die Abfrage des Stundenplans fehlerfrei?</i>		
	<input checked="" type="checkbox"/> ja	<input type="checkbox"/> nein	<input type="checkbox"/> nicht bewertet
Intervallabfrage:	<i>Funktioniert das Abfragen des Stundenplans mit einem voreingestellten Intervall?</i>		
	<input type="checkbox"/> ja	<input type="checkbox"/> nein	<input checked="" type="checkbox"/> nicht bewertet
Synchronisierung:	<i>Funktioniert die Synchronisierung mit dem Kalender des Smartphones?</i>		
	<input checked="" type="checkbox"/> ja	<input type="checkbox"/> nein	<input type="checkbox"/> nicht bewertet
Studiengang:	<i>Werden nur zum Nutzer passende Stundenpläne angezeigt?</i>		
	<input checked="" type="checkbox"/> ja	<input type="checkbox"/> nein	<input type="checkbox"/> nicht bewertet
Formatierung:	<i>Ist das Layout in Ordnung? (wird alles lesbar dargestellt)</i>		
	<input checked="" type="checkbox"/> ja	<input type="checkbox"/> nein	<input type="checkbox"/> nicht bewertet

Einstellungen:	<i>Test der Einstellungen des Moduls</i>
Wochenwahl:	<i>Funktioniert die Anzeige verschiedener Wochen?</i> <input checked="" type="checkbox"/> ja <input type="checkbox"/> nein <input type="checkbox"/> nicht bewertet
Intervallabfrage:	<i>Lässt sich ein Abfrageintervall einstellen?</i> <input type="checkbox"/> ja <input type="checkbox"/> nein <input checked="" type="checkbox"/> nicht bewertet
Benachrichtigungen:	<i>Funktionieren die Benachrichtigungen bei sich ändernden Stundenplänen usw.?</i> <input type="checkbox"/> ja <input type="checkbox"/> nein <input checked="" type="checkbox"/> nicht bewertet
Sonstiges:	<i>Anmerkungen zum Modul</i>
Hinweis:	<i>z.B. langsames Gerät beim nutzen des Moduls oder dergleichen</i>
Fehlermeldungen:	<i>Fehlermeldungen protokollieren</i>
Fehler:	<i>Beschreibung des Fehlers</i>
Fehlerursache:	<i>Bei welchen eingaben /Bedienung ist der Fehler aufgetreten</i>
Fehler reproduzierbar:	<i>Lässt sich der Fehler zuverlässig reproduzieren?</i>
Kritikalität:	<i>Wie schwer ist der Fehler auf einer Skala von 1-10</i>
Fehlermeldungen:	<i>Fehlermeldungen protokollieren</i>
Fehler:	<i>Beschreibung des Fehlers</i>
Fehlerursache:	<i>Bei welchen eingaben /Bedienung ist der Fehler aufgetreten</i>
Fehler reproduzierbar:	<i>Lässt sich der Fehler zuverlässig reproduzieren?</i>
Kritikalität:	<i>Wie schwer ist der Fehler auf einer Skala von 1-10</i>

Protokollempfänger: Alle Projektteilnehmer

Datum: 24.08.2015

Testprotokoll Stundenplan Modul

Projekt: HFTL-APP
Testobjekt: HfTL-App\app\Versionen\HFTL-APP_V0_6.apk
Prototyp-Version: 6
Protokoll-ID: 44
Testumgebung: Emuliertes Gerät Nexus 5 API 22 mit Android Studio.
Samsung Galaxie S3 mit Android 4.3
Datum: 19.08.2015
Tester: PK

Funktionstest:	<i>Funktion des Stundenplan Modul testen</i>		
Abfrage:	<i>Funktioniert die Abfrage des Stundenplans fehlerfrei?</i>		
	<input checked="" type="checkbox"/> ja	<input type="checkbox"/> nein	<input type="checkbox"/> nicht bewertet
Intervallabfrage:	<i>Funktioniert das Abfragen des Stundenplans mit einem voreigestellten Intervall?</i>		
	<input type="checkbox"/> ja	<input checked="" type="checkbox"/> nein	<input type="checkbox"/> nicht bewertet
Synchronisierung:	<i>Funktioniert die Synchronisierung mit dem Kalender des Smartphones?</i>		
	<input checked="" type="checkbox"/> ja	<input type="checkbox"/> nein	<input type="checkbox"/> nicht bewertet
Studiengang:	<i>Werden nur zum Nutzer passende Stundenpläne angezeigt?</i>		
	<input checked="" type="checkbox"/> ja	<input type="checkbox"/> nein	<input type="checkbox"/> nicht bewertet
Formatierung:	<i>Ist das Layout in Ordnung? (wird alles lesbar dargestellt)</i>		
	<input checked="" type="checkbox"/> ja	<input type="checkbox"/> nein	<input type="checkbox"/> nicht bewertet

Einstellungen:	<i>Test der Einstellungen des Moduls</i>
Wochenwahl:	<i>Funktioniert die Anzeige verschiedener Wochen?</i> <input checked="" type="checkbox"/> ja <input type="checkbox"/> nein <input type="checkbox"/> nicht bewertet
Intervallabfrage:	<i>Lässt sich ein Abfrageintervall einstellen?</i> <input type="checkbox"/> ja <input checked="" type="checkbox"/> nein <input type="checkbox"/> nicht bewertet
Benachrichtigungen:	<i>Funktionieren die Benachrichtigungen bei sich ändernden Stundenplänen usw.?</i> <input type="checkbox"/> ja <input checked="" type="checkbox"/> nein <input type="checkbox"/> nicht bewertet
Sonstiges:	<i>Anmerkungen zum Modul</i>
Hinweis:	...
Fehlermeldungen:	<i>Fehlermeldungen protokollieren</i>
Fehler:	<i>App stürzt ab</i>
Fehlerursache:	<i>Besteht bereits am Anfang keine oder bricht die Datenverbindung beim laden des Stundenplans ab stürzt die App ohne Fehlermeldung ab</i>
Fehler reproduzierbar:	<i>Zu 100% reproduzierbar</i>
Kritikalität:	<i>10/10 schwerer Fehler</i>

Protokollempfänger: Alle Projektteilnehmer

Datum: 19.08.2015

Testprotokoll Stundenplan Modul

Projekt: HFTL-APP
Testobjekt: HfTL-App\app\Versionen\HFTL-APP_V0_6.apk
Prototyp-Version: 6
Protokoll-ID: 60
Testumgebung: Samsung Galaxy S5, Android 5.0
Datum: 06.09.2015
Tester: SC

Funktionstest:	<i>Funktion des Stundenplan Modul testen</i>		
Abfrage:	<i>Funktioniert die Abfrage des Stundenplans fehlerfrei?</i>		
	<input checked="" type="checkbox"/> ja	<input checked="" type="checkbox"/> nein	<input type="checkbox"/> nicht bewertet
Intervallabfrage:	<i>Funktioniert das Abfragen des Stundenplans mit einem voreigestellten Intervall?</i>		
	<input type="checkbox"/> ja	<input checked="" type="checkbox"/> nein	<input checked="" type="checkbox"/> nicht bewertet
Synchronisierung:	<i>Funktioniert die Synchronisierung mit dem Kalender des Smartphones?</i>		
	<input checked="" type="checkbox"/> ja	<input type="checkbox"/> nein	<input type="checkbox"/> nicht bewertet
Studiengang:	<i>Werden nur zum Nutzer passende Stundenpläne angezeigt?</i>		
	<input checked="" type="checkbox"/> ja	<input type="checkbox"/> nein	<input type="checkbox"/> nicht bewertet
Formatierung:	<i>Ist das Layout in Ordnung? (wird alles lesbar dargestellt)</i>		
	<input checked="" type="checkbox"/> ja	<input type="checkbox"/> nein	<input type="checkbox"/> nicht bewertet

Einstellungen:	<i>Test der Einstellungen des Moduls</i>
Wochenwahl:	<i>Funktioniert die Anzeige verschiedener Wochen?</i> <input checked="" type="checkbox"/> ja <input type="checkbox"/> nein <input type="checkbox"/> nicht bewertet
Intervallabfrage:	<i>Lässt sich ein Abfrageintervall einstellen?</i> <input type="checkbox"/> ja <input checked="" type="checkbox"/> nein <input type="checkbox"/> nicht bewertet
Benachrichtigungen:	<i>Funktionieren die Benachrichtigungen bei sich ändernden Stundenplänen usw.?</i> <input type="checkbox"/> ja <input type="checkbox"/> nein <input checked="" type="checkbox"/> nicht bewertet
Sonstiges:	<i>Anmerkungen zum Modul</i>
Hinweis:	<i>z.B. langsames Gerät beim nutzen des Moduls oder dergleichen</i>
Fehlermeldungen:	<i>Bug bei Raumplanung</i>
Fehler:	<i>Bei allen Modulen steht „Raumplanung offen“</i>
Fehlerursache:	<i>Falsche If-Anweisung in der Befüllung der TextViews [Fix geliefert, geht kommende Version live]</i>
Fehler reproduzierbar:	<i>Ja</i>
Kritikalität:	<i>2</i>
Fehlermeldungen:	<i>Fehlermeldungen protokollieren</i>
Fehler:	<i>Beschreibung des Fehlers</i>
Fehlerursache:	<i>Bei welchen eingaben /Bedienung ist der Fehler aufgetreten</i>
Fehler reproduzierbar:	<i>Lässt sich der Fehler zuverlässig reproduzieren?</i>
Kritikalität:	<i>Wie schwer ist der Fehler auf einer Skala von 1-10</i>

Protokollempfänger: Alle Projektteilnehmer

Datum: 06.09.2015

Testprotokoll Stundenplan Modul

Projekt: HFTL-APP

Testobjekt:

Prototyp-Version: 7

Protokoll-ID: 65

Testumgebung: Android 4.1.2

Datum: 09.09.2015

Tester: SK

Funktionstest:

Funktion des Stundenplan Modul testen

Abfrage:

Funktioniert die Abfrage des Stundenplans fehlerfrei?

ja nein nicht bewertet

Intervallabfrage:

Funktioniert das Abfragen des Stundenplans mit einem voreigestellten Intervall?

ja nein nicht bewertet

Synchronisierung:

Funktioniert die Synchronisierung mit dem Kalender des Smartphones?

ja nein nicht bewertet

Studiengang:

Werden nur zum Nutzer passende Stundenpläne angezeigt?

ja nein nicht bewertet

Formatierung:

Ist das Layout in Ordnung? (wird alles lesbar dargestellt)

ja nein nicht bewertet

Einstellungen:	<i>Test der Einstellungen des Moduls</i>
Wochenwahl:	<i>Funktioniert die Anzeige verschiedener Wochen?</i> <input checked="" type="checkbox"/> ja <input type="checkbox"/> nein <input type="checkbox"/> nicht bewertet
Intervallabfrage:	<i>Lässt sich ein Abfrageintervall einstellen?</i> <input checked="" type="checkbox"/> ja <input type="checkbox"/> nein <input type="checkbox"/> nicht bewertet
Benachrichtigungen:	<i>Funktionieren die Benachrichtigungen bei sich ändernden Stundenplänen usw.?</i> <input checked="" type="checkbox"/> ja <input type="checkbox"/> nein <input type="checkbox"/> nicht bewertet
Sonstiges:	--KEINE--
Hinweis:	--KEINE--
Fehlermeldungen:	--KEINE--
Fehler:	--KEINE--
Fehlerursache:	--KEINE--
Fehler reproduzierbar:	--KEINE--
Kritikalität:	--KEINE--
Fehlermeldungen:	--KEINE--
Fehler:	--KEINE--
Fehlerursache:	--KEINE--
Fehler reproduzierbar:	--KEINE--
Kritikalität:	--KEINE--

Protokollempfänger: Alle Projektteilnehmer

Datum: 09.09.2015

5.8 SITZUNGSPROTOKOLLE UND AUFGABENLISTEN

Aufgabenlisten

NR.	ERSTELLT	VERANTWORTLICHE	BESCHREIBUNG	SOLL ZIEL	FINAL	VERZUG1	VERZUG2	STATUS	LÖSUNG
A1	19.04.2015	JS, SC	Vorgehen des Projektteams bzgl. Ordnerstruktur und Änderungen der Daten (Wer Merged, Wann?) festlegen.	26.04.2015	03.05.2015	keine Zeit, neu 03.05.15		erl	Verzeichnisse: "Ablage" für Protokollanhänge, "Shared" für Dateien die jeder brauchen könnte; Oberordner für "Dokumentation", "App" und "UML"; zum Vorgehen: TN erstellen Branches, Verantwortlicher gibt "ok" und merged, danach den alten Branch löschen
A2	19.04.2015	ML	Was passiert mit Klarnamen in Protokollen oder wird anonymisiert?	26.04.2015	20.04.2015			erl	Ilias oder GitHub, Festlegung dann im Meeting
A3	19.04.2015	GE, SC, SK	Umgang mit Datenschutz innerhalb der App, da Nutzerdaten in Klartext auf Smartphone gespeichert werden	offen	03.05.2015			erl	Verschlüsselung von Nutzerdaten und anschließender Zugriff erfolgreich
A4	19.04.2015	ML, SK	Ausarbeitung eines Protokolllayouts für Besprechungen und Testphase	26.04.2015	25.05.2015	Testprotokoll noch offen, neu 10.05.15	keine Zeit (privat), neu 17.05.15	erl	
A5	26.04.2015	JS	Klärung ob Lasten-/Pflichtenheft als Einzel- oder Gesamtdokumentation erstellt werden soll, Rücksprache mit Prof.	03.05.2015	04.05.2015	Antwort von Prof. noch offen, neu 10.05.15		erl	Dokumentation als Ganzes abgeben, Teilabsprache im Vorfeld mit Prof. möglich
A6	26.04.2015	ML, SC	Ergebnisse vom Treffen „Design“	03.05.2015	03.05.2015			erl	Klärung zur App-Programmierung und Umgang mit Android-Studio verschafft; Abschluss AD1 + AD2
A7	26.04.2015	AD, CM	Ergebnisse vom Treffen „UML“	03.05.2015	03.05.2015			erl	Klärung zum Ablauf der App-Programmierung, nachreichen offener UML Klassen-, und Ablaufdiagramme und Vorarbeit
A8	26.04.2015	SC, SK	Terminierung der Meilensteine	03.05.2015	29.04.2015			erl	Zeiten vergeben

NR.	ERSTELLT	VERANTWORTLICHE	BESCHREIBUNG	SOLL ZIEL	FINAL	VERZUG1	VERZUG2	STATUS	LÖSUNG
A9	29.04.2015	SK	Meilensteine grafisch	03.05.2015	17.06.2015	keine Zeit, neu 10.05.15	Upload vergessen, falscher PC, per Mail zum 13.05.15	erl	Stand: 07.06.15 - ML klärt mit SK weiteren Verlauf // Stand: 17.06.15 - Diagramm erstellt
A10	03.05.2015	SK	Vorlage Lasten-/ Pflichtenheft nach IEEE 830	10.05.2015	12.06.2015	kein Upload erfolgt bis 15.05.15	Stand: 07.06.15 - SC & PK klären weiteren Verlauf mit SK	erl	Vorlage erstellt, Inhalte werden nun mit JS abgesprochen und zusammengeschrieben; Ablage in GitHub;
A11	10.05.2015	SK	Rechtliches / Lizzenzen: Klären ob fertige App in AppStore oder nur auf Website der HfT; Rücksprache mit Prof	offen	02.08.2015	Klärunung mit GF der HfTL läuft		erl	Rücksprache mit Rechenzentrum und Bereich für Öffentlichkeitsarbeit der HfTL. App bekommt einen Bereich im Webauftritt der Hochschule.
A12	10.05.2015	ML	Klären ob CI-Grafiken für Flash-Grafik verändert werden können/dürfen; Rücksprache mit Brand-Support (intern)	offen	21.06.2015			erl	Weiterer Nutzen der animierten Startseite wurde vom Projektteam abgelehnt.
A13	10.05.2015	CM, AD	UML Diagramme in GitHub einstellen	17.05.2015	04.06.2015			erl	Diagramme hochgeladen in Ordner "UML"
A14	21.06.2015	GE, SC	Überschriften der Artikel anpassen	28.06.2015	12.07.2015	keine Zeit, neu 19.07.15		erl	geändert von "Termindetails" in "News-Details"
A15	21.06.2015	SK	Zeiten der Meilensteine anpassen	28.06.2015	28.06.2015			erl	Zeiten an Projektablauf angepasst
A16	21.06.2015	SK	Gantt-Diagramm an Vorgehensmodell anpassen	28.06.2015	28.06.2015			erl	Ablauf an Prototypen orientiert
A17	21.06.2015	ML, SC	Name für Formatierungsdatei finden	28.06.2015	28.06.2015			erl	HFTL-App_Designvorlage.pdf

NR.	ERSTELLT	VERANTWORTLICHE	BESCHREIBUNG	SOLL ZIEL	FINAL	VERZUG1	VERZUG2	STATUS	LÖSUNG
A18	21.06.2015	ML, PK	Klären der Testprotokoll-ID	28.06.2015	25.06.2015			erl	Testprotokolle je nach App-Modul und für Gesamtsystem erstellt, Ablage auf GitHub > Dokumentation > Testprotokolle; Test-ID wird in im Protokollkopf vermerkt, eine Übersicht der Tests liegt ebenfalls auf GitHub (Testübersicht), daraus kann jeder sehen, welche ID als nächstes genutzt werden soll
A19	02.08.2015	GE	Push-Service vom Modul „Noten“ optimieren	09.08.2015	03.08.2015			erl	Einstellungen angepasst, neue Berechtigung für die App "Start bei Systemstart", kleinere Testläufe
A20	02.08.2015	ML	Formatvorlage abschließen und versenden	09.08.2015	09.08.2015			erl	Designvorlage abgeschlossen und aktualisiert, Ablage im Ordner "Share" auf GitHub
A21	02.08.2015	alle	GitHub Issues bearbeiten/abschließen	09.08.2015	09.08.2015			erl	bereits abgeschlossene aber noch offene Issues wurden geschlossen
A22	02.08.2015	PK	Klärtung zum Thema Versionsmanagementprotokolle	09.08.2015	30.08.2015	Inhalte und erforderliche Dateien waren nicht bekannt; neues Zeitziel: 16.08.15	PK klärt: Thematische Überschneidung, Neues Zeitziel: 30.08.2015	erl	Ergebnis auf GitHub
A23	02.08.2015	PK	UML Diagramme prüfen, Fehler melden	09.08.2015	16.08.2015	offen, neues Zeitziel: 16.08.15		erl	UML wurden geprüft und aktualisiert.
A24	02.08.2015	GE	beschriebene Methoden im Handbuch prüfen und Fehler melden	09.08.2015	09.08.2015			erl	Absprache mit UML-Team erfolgt und Änderungen durchgeführt
A25	02.08.2015	alle	Mindestens ein Test pro Modul durchführen	13.09.2015	06.09.2015			erl	Tests für APK6 erledigt, weitere Tests mit APK7

NR.	ERSTELLT	VERANTWORTLICHE	BESCHREIBUNG	SOLL ZIEL	FINAL	VERZUG1	VERZUG2	STATUS	LÖSUNG
A26	02.08.2015	SK, GE, SC	Inhalt des Impressums zusammenschreiben und Menüpunkt in der App einbauen.	09.08.2015	16.08.2015	Inhalte zusammengetragen, Ablage unter "Share", Einbinden in App noch offen, neues Zeitziel: 16.08.15		erl	Das Impressum wurde erstellt, in die App eingefügt und mit dem aktuellen Layout versehen.
A27	02.08.2015	alle	Abschlusspräsentation	09.08.2015	09.08.2015			erl	Planung der Präsentatoren, Ausarbeitung der Inhalte durch PK, neues Zeitziel 16.08.15
A28	09.08.2015	alle	Offene Stellen im Entwicklerhandbuch mit Inhalt füllen (allgemeiner Text zu Activities und Fragmenten)	16.08.2015	30.08.2015	Inhalt wird an SC zur Einarbeitung übergeben. Neues Zeitziel: 30.08.2015		erl	Änderungen in LaTeX übernommen und als PDF auf GitHub
A29	09.08.2015	GE	Kalendersynchronisierung des Stundenplans	16.08.2015	16.08.2015			erl	Die Funktionalität ist bereits mit APK-Version 6 implementiert. Zu finden im Stundenplan -> Auswahl der gewünschten KW -> rechter Menüpunkt -> „Mit Kalender synchronisieren“
A30	09.08.2015	PK	Inhalte der Präsentation zusammentragen	offen	03.09.2015	grober Entwurf vorhanden, alle sollen Feedback geben, Präsentationsdatei unter „Präsentation > Dokumentation“, Neues Zeitziel: 30.08.15	offen durch andere Aufgaben, neues Zeitziel: 06.09.15	erl	Präsentation v1.0.pptx auf GitHub

NR.	ERSTELLT	VERANTWORTLICHE	BESCHREIBUNG	SOLL ZIEL	FINAL	VERZUG1	VERZUG2	STATUS	LÖSUNG
A31	16.08.2015	GE	Erinnerungsfunktion für anstehende Ereignisse	30.08.2015	06.09.2015	offen, war optional geplant, neu: offen		erl	Option bleibt optional und wird eine Funktion für die spätere Entwicklung
A32	16.08.2015	CM, AD	Doku – 4.7.3 Projektstrukturplan	30.08.2015	06.09.2015	Skizze wurde angefertigt, aktuelles Organigramm als Basis, ML erstellt bis 06.09.15		erl	Dokumentation\shared\swe_projektstukturplan.pdf'
A33	16.08.2015	JS	Doku – Einfügen der Testprotokolle in den Anhang	30.08.2015	30.08.2015			erl	auf GitHub
A34	16.08.2015	SC	Doku – Entwicklerhandbuch wird final auf Basis APK-Version 6 fertiggestellt	30.08.2015	30.08.2015			erl	Angepasst an APK6 und aktuelle Version auf GitHub
A35	16.08.2015	CM, AD	Doku – Offene Punkte (verwendete Software)	30.08.2015	30.08.2015			erl	auf GitHub
A36	16.08.2015	PK	QM – Prüfung der bereits erledigten Punkte in der Projektdokumentation	30.08.2015	30.08.2015			erl	Rechtschreibung und andere kleine Fehler ausgebessert
A37	16.08.2015	PK	QM – Prüfung ob Anforderungen in Dokumentation vorhanden	30.08.2015	06.09.2015	rote stellen konnten noch nicht geprüft werden, Info an Verantwortliche muss noch mal raus (u.a. 4.5.3., 4.6.5.)		erl	Änderungen erfolgt und eingepflegt
A38	16.08.2015	CM, AD	UML – Prüfung ob UML auf Stand APK-Version 6 sind.	30.08.2015	30.08.2015			erl	auf GitHub

NR.	ERSTELLT	VERANTWORTLICHE	BESCHREIBUNG	SOLL ZIEL	FINAL	VERZUG1	VERZUG2	STATUS	LÖSUNG
AD1	27.04.2015	SC	benötigte Farben und Schriften erarbeiten	03.05.2015	29.04.2015			erl	in GitHub "layout.xml"
AD2	27.04.2015	ML	Liste im farblichen Wechsel darstellen	03.05.2015	10.05.2015	Probleme mit AndroidStudio, Lösung per ListView-Adapter		erl	ListView an Layoutvorlage erstellt und angepasst (SC); Stand im GitHub ersichtlich
AD3	27.04.2015	SC	Symbole und Icons erarbeiten	offen	03.05.2015			erl	Icon erstellt und eingebunden
AD4	27.04.2015	SC, ML	Einbinden von Layoutdateien in App erarbeiten	offen	24.05.2015			erl	Layout wird über xml-file formatiert
AD5	27.04.2015	SC, ML	Quellcode der QIS-Notendaten analysieren, um Verwendung in App zu ermöglichen	offen	06.05.2015			erl	Abfrage der Noten mit QIS-Abfrage möglich; Vorläufige Ausgabe per unformatiertem String
AD6	07.05.2015	ML	Formatierungsdaten aus Preview-Layout erfassen	10.05.2015	20.06.2015	Verwendete Bilder, Schriften und Farben erfasst; Abmessungen noch offen; neu 17.05.15	Formatierung erfolgt, Umrechnung von px in dp erforderlich: neu 21.06.15	erl	Formatierungen wurden abgeschlossen, Layoutdaten wurden zusammengefasst
AD7	10.05.2015	SC, ML, GE	Menüfarben anpassen	offen	06.09.2015			erl	Option bleibt optional und wird ein Punkt für die spätere Entwicklung
AD8	07.06.2015	SC, ML	Anpassen des Notenlayouts	21.06.2015	27.06.2015	in Arbeit, neu 28.06.15		erl	Formatierungen des Notenmoduls an Vorlagen angepasst
AD9	07.06.2015	SC, GE	Übernahme der Textformatierung einer News in die App	21.06.2015	25.08.2015	in Arbeit, neu offen		erl	News werden nun als html übernommen und formatiert
AD10	07.06.2015	ML	Layout für Stundenplan erstellen	21.06.2015	19.06.2015			erl	Layout erstellt, Umsetzung in App läuft
AD11	28.06.2015	ML	Gestaltungsvorlage für Stundenplan ListView erstellen	12.07.2015	28.06.2015			erl	stundenplan_list.xml

NR.	ERSTELLT	VERANTWORTLICHE	BESCHREIBUNG	SOLL ZIEL	FINAL	VERZUG1	VERZUG2	STATUS	LÖSUNG
AD12	28.06.2015	SC	Anordnung der Semester zeitlich und nicht alphabetisch anpassen	19.07.2015	16.08.2015	in Arbeit, neu offen		erl	in APK6 integriert
AD13	02.08.2015	SC, ML, GE	Klärung zum Thema offene Icons	09.08.2015	25.08.2015	in Arbeit, neu 16.08.15		erl	Icon für Statusleiste eingebunden
AD14	09.08.2015	SC	Rückfalloption auf Schriftart ‚Arial‘ prüfen	16.08.2015	16.08.2015			erl	Schriftarten nach CI/CD konnten erfolgreich implementiert. Das Speicherproblem konnte durch Erstellen einer neuen Klasse umgangen werden.

NR.	ERSTELLT	VERANTWORTLICHE	BESCHREIBUNG	SOLL ZIEL	FINAL	VERZUG1	VERZUG2	STATUS	LÖSUNG
AU1	12.05.2015	AD	Aktivitätsdiagramm für den Unterpunkt News erstellen.	17.05.2015	26.05.2015			erl	Activity_StarteNews.pdf
AU2	12.05.2015	CM	(gezeichneter) Entwurf vom Klassendiagramm für die gesamte APP digitalisieren	17.05.2015	29.05.2015			erl	Klassendiagramm_Main.pdf
AU3	12.05.2015	CM	Beginnen mit der Beschreibung des Programmablaufs	17.05.2015	01.06.2015			erl	begonnen, Upload erfolgt nach Absprache mit SK
AU4	21.06.2015	CM, AD	Programmieränderungen in UML-Diagramme übernehmen	28.06.2015	02.08.2015	noch nicht alle Änderungen übernommen, neu nach 28. KW		erl	Alle UML-Diagramme auf GitHub

Sitzungsprotokolle

Sitzungsprotokoll

Thema der Besprechung: Entwicklung/ Status im Projekt HFTL-APP

Datum: 19.04.2015

Zeit: 19:30

Ort: TeamViewer Meeting

Eingeladene Stefan Czogalla; Andrej Dederer; Georg Ebert; Stephan
Teilnehmer: Kaden; Patrick Kunze; Maik Lorenz; Christoph Mathies; Jan Sutmöller

Leitung: Stefan Czogalla

Protokollführer: Maik Lorenz

Anwesende: Stefan Czogalla; Andrej Dederer; Georg Ebert; Stephan
Kaden; Patrick Kunze; Maik Lorenz; Christoph Mathies; Jan
Sutmöller

TOP 1:	<i>Aufgabenliste</i>	<i>Status erledigt</i>
Diskussion:	Im Vorfeld des Kick-Offs sollten sich die Teilnehmer zu verschiedenen Aufgaben belesen um die Ergebnisse vor der Gruppe zu präsentieren:	
Ergebnis:	SC: App-Design allgemein <ul style="list-style-type: none">- Layout auf Basis des CI/CD der Hochschule: https://www.hft-leipzig.de/de/hochschule/service/medien.html Festlegung: Unterscheidung nach Studiengängen direkt, dual und berufsbegleitend kann umgesetzt werden, da dies auch für den Stundenplan benötigt wird	

Ergänzung PK,
01.05.2015

Änderungswunsch PK,
01.05.2015

SK: Projektplanung

- Vorstellung einzelner Projektphasen (Initialisierung, Projektplanung, Durchführung, Projektsteuerung, Projektabschluss) und deren Unterpunkte
- Vorstellung der Prozessmodelle „Wasserfall-“ und „V-Modell“, sowie Gegenüberstellung/Bewertung der Modelle zur Projektanforderung
- Vorstellung Rollenverteilung und einzelne Unteraufgaben
→ top_projektmanagement.pdf

JS + GE: Umgang mit GitHub

- Vorstellung des Projekteordners, wie werden Dateien angelegt, wie reagiert das System auf Änderungen, wann wird aktualisiert, wie werden Rollen verteilt
- Klärung des Unterschiedes zwischen Branches- und Master-Bereichen
- Nutzung von GitHub zum Versionsmanagement
Festlegung: Testdateien werden aus Projektordner entfernt und neue Unterordner für Teilbereiche erstellt.

ML: Protokollierung

- Vorstellung zu Protokollinhalten, Protokollänge, Aufbau und Struktur
- Aufgaben der Protokollteilnehmer: Protokoll lesen und absegnen, ggf. Änderungen mit Zielzeit X bekannt zu geben, nach Verstreichen der Frist gilt das Protokoll als **Final festgelegt**
- Protokollorganisation: zentral und für alle Teilnehmer zugänglich, als Dateiformat wird PDF gewählt
→ top_protokollierung.pdf

GE: Aktueller Stand App-Programmierung

- [Verwendung der Entwicklungsumgebung Android Studio](#)
- Zur Klärung der App-Möglichkeiten wurden die Funktionen HfTL-News-Abruf, Nutzung mit Benutzerdaten und Fehlerbehandlungen getestet
- Einsatzmöglichkeit für Zielgruppe „Androidnutzer“ soll bei 90% liegen, App-Programmierung in der Aktuellen Form ist für den Einsatz ab Android 4.0 gedacht

	<p>AD: UML</p> <ul style="list-style-type: none"> - Vorstellung bisheriger Ergebnisse durch Einsatz von <i>Enterprise Architect Academic</i>, zur Nutzung wird eine VPN Verbindung zur HfTL benötigt damit Lizenz genutzt werden kann - Präsentation eines Use-Case-Diagramms und eines Klassendiagramms <p>Klärung in der Gruppe über Notwendigkeit im Projekt -> dient als Vorarbeit für die Programmierung, zeigt den Projektzustand, dient allgemein zur besseren Übersicht</p>
Aufgaben:	<p>A1: Vorgehen des Projektteams bzgl. Ordnerstruktur und Änderungen der Daten (Wer Merged, Wann?) festlegen. Verantwortliche: JS, SC Termin: 26.04.15</p>
	<p>A2: Was passiert mit Klarnamen in Protokollen oder wird anonymisiert? Verantwortliche: ML Termin: 26.04.15</p>
	<p>A3: Umgang mit Datenschutz innerhalb der App, da Nutzerdaten in Klartext auf Smartphone gespeichert werden Verantwortliche: GE, SC, SK Termin: offen</p>

TOP 2:	<i>Aufgabenverteilung</i>	<i>Status erledigt</i>
Diskussion:	Nach der Vorstellung wurden 4 Teilgebiete (Projektleitung, Programmierung, Qualitätsmanagement, Dokumentation) abgesprochen und Unteraufgaben notiert. Im Anschluss wurden die Aufgaben auf die Teilnehmer verteilt und im Organigramm zusammengefasst:	
Ergebnis:	➔ swe_organigramm.pdf	

TOP 3:	<i>Sonstiges</i>
Diskussion:	Besprechung offener Themen.
Ergebnis:	<ul style="list-style-type: none"> - Sitzungsprotokolle spätestens nach einer Woche final ablegen - Verwendung von LaTeX zum Schreiben der Dokumente <i>Handbuch</i> und <i>Bedienungsanleitung</i> - Klärung der Testphasen und Umgang mit Ergebnissen für die Dokumente - Testprotokolle gehen an Programmierteam, welches die Erkenntnisse in die Bedienungsanleitung einbringen - Nächste Treffen wöchentlich, sonntags. 19:30Uhr
Aufgaben:	<p>A4: Ausarbeitung eines Protokolllayouts für Besprechungen und Testphase</p> <p>Verantwortliche: ML, SK Termin: 26.04.15</p>
TOP 4:	<i>Nächster Termin</i>
Ergebnis:	Sonntag, 26.04.2015

Protokollempfänger: Alle Projektteilnehmer

Datum: 29.04.2015

Festlegung: 03.05.2015

Sitzungsprotokoll

Thema der Besprechung: Entwicklung/ Status im Projekt HFTL-APP

Datum: 26.04.2015

Zeit: 20:00

Ort: TeamViewer Meeting

Eingeladene Stefan Czogalla; Andrej Dederer; Georg Ebert; Stephan
Teilnehmer: Kaden; Patrick Kunze; Maik Lorenz; Christoph Mathies; Jan Sutmöller

Leitung: Stefan Czogalla

Protokollführer: Maik Lorenz

Anwesende: Stefan Czogalla; Andrej Dederer; Georg Ebert; Stephan
Kaden; Patrick Kunze; Maik Lorenz; Jan Sutmöller

TOP 1:	<i>Aufgabenliste</i>	<i>Status erledigt</i>
Diskussion:	Alle TN geben den Stand Ihrer Aufgaben wieder. Themenverantwortliche berichten über neue Ergebnisse und Fortschritte.	
Ergebnis:	A1: Vorgehen des Projektteams bzgl. Ordnerstruktur und Änderungen der Daten (Wer Merged, Wann?) festlegen. Verantwortliche: JS, SC <i>Status: offen</i> Neues Zeitziel 03.05.15	
	A2: Was passiert mit Klarnamen in Protokollen? Verantwortliche: ML <i>Status: erledigt</i> Es besteht die Möglichkeit den internen Bereich von Ilias zu nutzen, wenn Anonymisierung gewünscht Abstimmung: Protokolle können mit Namen auf GitHub	

A3: Umgang mit Datenschutz innerhalb der App, da Nutzerdaten in Klartext auf Smartphone gespeichert werden

Verantwortliche: GE, SC, SK *Status: offen*

Thema „Rechtliches/ Lizenzen“ noch später im Zeitplan vorgesehen

A4: Ausarbeitung eines Protokolllayouts

Verantwortliche: ML, SK *Status: offen*

Besprechungsprotokolllayout fertig, Testprotokolllayout in Arbeit; Neues Ziel, wenn Testphase beginnt, Planung baut auf Phasendiagramm auf.

GE: Programmierung:

- Funktionalität von News vorhanden,
- TOP Stundenplan:
 - Test Abruf der Daten aus QIS mit „iCal“;
 - Test mittels Java-Classes mit NetBeans

ML, GE, SC: App-Design

- Absprache zum Treffen: 27.04.15; 21Uhr

SK: Protokollierung

- Vorlage nun auf GitHub zugänglich

JS: Dokumentation

- LaTeX-Vorlage zum Lastenheft erstellt und in GitHub hinterlegt (nach Absprache mit SC und SK vom 22.04.15)

JS: GitHub

- Vorgehen zum Erstellen **und Aktualisieren** einer Datei und aktualisieren, sowie Fehlerbehandlung über GitHub-Shell erläutert

Ergänzung PK;
01.05.2015

TOP 2:	Sonstiges
Diskussion:	Besprechung offener Themen und Fragen.
Ergebnis:	Lastenheft Festlegung: alle TN sind aufgefordert den Stand des Lastenheftes zu lesen und ggf. Änderungen an SC und SK zu melden
	Protokolle Festlegung: Ablage von Protokollen auf GitHub im Ordner „Sitzungsprotokolle“

Festlegung: Ablage zusätzliche Unterlagen zum Protokoll im Ordner „Protokolle“ > „Ablage“

Projektmanagement

Festlegung: als Vorgehensmodell wurde Prototyping kombiniert mit Spiralmodell gewählt

TOP 3:	<i>Nächster Termin</i>
Ergebnis:	Sonntag, 03.05.2015

Protokollempfänger: Alle Projektteilnehmer

Datum: 29.04.2015

Festlegung: 03.05.2015

Sitzungsprotokoll

Thema der Besprechung: Entwicklung/ Status im Projekt HFTL-APP

Datum: 27.04.2015

Zeit: 21:00

Ort: TeamViewer Meeting

Eingeladene Teilnehmer: Stefan Czogalla; Georg Ebert; Maik Lorenz

Leitung: Stefan Czogalla

Protokollführer: Maik Lorenz

Anwesende: Stefan Czogalla; Georg Ebert; Maik Lorenz

TOP 1:	<i>Aufgabenliste</i>	<i>Status erledigt</i>
Diskussion:	Alle TN geben den Stand Ihrer Aufgaben wieder. Themenverantwortliche berichten über neue Ergebnisse und Fortschritte.	
Ergebnis:	GE: Programmierung <ul style="list-style-type: none">- Zeigt aktuellen Stand und gibt erste Hinweise im Umgang mit <i>Android Studio 1.1.0</i>	
TOP 2:	<i>Aufgabenverteilung</i>	<i>Status offen</i>
Diskussion:	TN sammeln Ideen zur möglichen grafischen Gestaltung der App, erörtern eventuellen Aufbau des Menüs und planen das weitere Vorgehen.	
Ergebnis:	Starten der App <ul style="list-style-type: none">- Soll mit Ladebildschirm beginnen,- Keine Anmeldemaske, da die Daten nicht für alle Funktionen benötigt werden	

AD4: Einbinden von Layoutdateien in App erarbeiten

Verantwortliche: SC, ML Termin: 03.05.15

AD5: Quellcode der QIS-Notendaten analysieren, um Verwendung in App zu ermöglichen

Verantwortliche: SC, ML Termin: offen

TOP 2: *Sonstiges*

Diskussion: Besprechung offener Themen und Fragen.

Ergebnis: Farbliche Anpassung nach Studienart

- Eine farbliche Gestaltung nach Direkt, Dual oder Beruf, bzw. eine Option zur Layoutwahl erfolgt vorerst nicht.

Anzahl der angezeigten News

- Die Anzahl der aktuellen News wird vorerst nicht durch die Funktion „Anzeigen älterer Nachrichten“ ergänzt.

Protokollempfänger: Alle Teilnehmer

Datum: 29.04.2015

Festlegung: 03.05.2015

Sitzungsprotokoll

Thema der Besprechung: Entwicklung/ Status im Projekt HFTL-APP

Datum: 29.04.2015

Zeit: 19:30

Ort: TeamViewer Meeting

Eingeladene Stefan Czogalla; Stephan Kaden; Patrick Kunze; Maik
Teilnehmer: Lorenz; Jan Sutmöller

Leitung: Stefan Czogalla

Protokollführer: Maik Lorenz

Anwesende: Stefan Czogalla; Stephan Kaden; Maik Lorenz

TOP 1:	Aufgabenliste	Status erledigt
Diskussion:	Alle TN geben den Stand Ihrer Aufgaben wieder. Themenverantwortliche berichten über neue Ergebnisse und Fortschritte.	
Ergebnis:	<p>SC: Design</p> <ul style="list-style-type: none"> - Kurze Absprache der Projektleitung zum Meeting der Programmiergruppe vom 27.04.15 - Einbinden von Farben und Schriften in die App ist geklärt <p>ML: Organigramm</p> <ul style="list-style-type: none"> - Letzte Änderungen übernommen 	
A8: Terminierung der Meilensteine	Verantwortliche: SC, SK	<i>Status: erledigt</i>
	<ul style="list-style-type: none"> - Meilensteine aus dem Meeting 26.04.15 wurden terminiert - Zur Darstellung wird ein Gantt-Ablaufplan erstellt 	

Aufgaben:

A9: Meilensteine grafisch darstellen

Verantwortliche: SC, SK

Termin: 03.05.15

TOP 2:

Sonstiges

Diskussion:

Besprechung offener Themen und Fragen.

Ergebnis:

Lastenheft

- Rücksprache zu JS
- Darstellung des Lastenhefts soll auf Basis der *SRS – Software Requirements Specification IEEE 830* erfolgen

Gantt-Diagramm

- Verwendung einer Bearbeitungssoftware zur Zeiteinsparung und Fehlervermeidung in der Zeitplanung

Protokollempfänger: Alle Teilnehmer

Datum: 29.04.2015

Festlegung: 03.05.2015

Sitzungsprotokoll

Thema der Besprechung: Entwicklung/ Status im Projekt HFTL-APP

Datum: 03.05.2015

Zeit: 19:30

Ort: TeamViewer Meeting

Eingeladene Stefan Czogalla; Andrej Dederer; Georg Ebert; Stephan
Teilnehmer: Kaden; Patrick Kunze; Maik Lorenz; Christoph Mathies; Jan Sutmöller

Leitung: Stefan Czogalla

Protokollführer: Maik Lorenz

Anwesende: Stefan Czogalla; Georg Ebert; Stephan Kaden; Patrick Kunze; Maik Lorenz; Christoph Mathies; Jan Sutmöller

TOP 1:	<i>Aufgabenliste</i>	<i>Status erledigt</i>
Diskussion:	Alle TN geben den Stand Ihrer Aufgaben wieder. Themenverantwortliche berichten über neue Ergebnisse und Fortschritte.	
Ergebnis:	A1: Vorgehen des Projektteams bzgl. Ordnerstruktur und Änderungen der Daten (Wer Merged, Wann?) festlegen. Verantwortliche: JS, SC Verzeichnisse: "Ablage" für Protokollanhänge, "Shared" für Dateien die jeder brauchen könnte; Oberordner für "Dokumentation", "App" und "UML"; Vorgehen: TN erstellen Branches, Verantwortlicher gibt "ok" und merged, danach wird der alte Branch gelöscht	<i>Status: erledigt</i>

A3: Umgang mit Datenschutz innerhalb der App

Verantwortliche: GE, SC, SK Status: *erledigt*

Verschlüsselung von Nutzerdaten und anschließender Zugriff erfolgreich

A4: Ausarbeitung eines Protokolllayouts

Verantwortliche: ML, SK Status: *offen*

Besprechungsprotokolllayout fertig, Testprotokolllayout in Arbeit; Rücksprache mit PK zu Inhalten

A5: Klärung ob Lasten-/Pflichtenheft als Einzel- oder Gesamtdokumentation erstellt werden soll, Rücksprache mit Prof.

Verantwortliche: JS Status: *offen*

Rückmeldung von Prof. noch offen, neues Zeitziel: 10.05.15

A6: Ergebnisse vom Treffen „Design“

Verantwortliche: ML, SC Status: *erledigt*

Klärung zur App-Programmierung und Umgang mit Android-Studio verschafft; Abschluss AD1 + AD2; nächster Termin in 19.KW

A7: Ergebnisse vom Treffen „UML“

Verantwortliche: AD, CM Status: *erledigt*

Klärung zum Ablauf der App-Programmierung, nachreichen offener UML Klassen-, und Ablaufdiagramme und Vorarbeit; nächster Termin in 19.KW

A8: Terminierung der Meilensteine

Verantwortliche: SK Status: *erledigt*

→ **top_meilensteine.pdf**

A9: Meilensteine grafisch darstellen

Verantwortliche: SK Status: *offen*

Neues Zeitziel: 10.05.2015

Änderungswunsch ML;
05.05.2015

ML: Protokollierung

- Probleme mit eingebetteten Dateien im Protokoll soll durch anfügen als Anhang geklärt werden

Abstimmung: Protokoll wird auf GitHub zentral abgelegt; Änderungswünsche/Ergänzungen werden an **Protokollführer** gemeldet und in der Datei auf GitHub mit Vermerk umgesetzt; dies soll das Versionsmanagement der Protokolle transparenter machen

CM: UML

- Treff in 19.KW soll klären, wie der Aufbau der App bzw. das Abbilden des gesamten Projekts in UML-Diagrammen umgesetzt werden kann

PK: Qualitätsmanagement

- Erinnerung an alle die UML-Ergebnisse
gegenzulesen

Festlegung: Zwischenergebnisse auf GitHub ablegen

PK: Qualitätsmanagement

- Es wurden Probleme beim Testen der App-Dateien direkt aus GitHub entdeckt. GitHub aktualisiert die Dateien und möchte synchronisieren. Es wird daher empfohlen beim Test einen separaten Branch anzulegen oder die Daten anderweitig zu kopieren.(siehe Lösung aus A1)

Aufgaben:

A10: Vorlage Lasten-/ Pflichtenheft nach IEEE 830

Verantwortliche: SK

Termin: 10.05.2015

TOP 2:

Sonstiges

Diskussion:

Besprechung offener Themen und Fragen.

Ergebnis:

Ausblick

- Nach grafischer Abbildung der Meilensteine, kann die Funktion „Milestone“ in GitHub eingesetzt werden. Verantwortliche bekommen dann entsprechende „Issues“ zugewiesen. Dies schafft einen besseren Überblick über

- den Stand eines Meilensteins und bietet die Möglichkeit schneller auf veränderte Anforderungen zu reagieren.
- Einen kurzen Einblick in das Vorgehen erfolgte durch JS

TOP 3:	<i>Nächster Termin</i>
Ergebnis:	Sonntag, 10.05.2015

Protokollempfänger: Alle Projektteilnehmer

Datum: 03.05.2015

Festlegung: 08.05.2015

Sitzungsprotokoll

Thema der Besprechung: Entwicklung/ Status im Projekt HFTL-APP

Datum: 10.05.2015

Zeit: 19:30

Ort: TeamViewer Meeting

Eingeladene Stefan Czogalla; Andrej Dederer; Georg Ebert; Stephan
Teilnehmer: Kaden; Patrick Kunze; Maik Lorenz; Christoph Mathies; Jan Sutmöller

Leitung: Stephan Kaden

Protokollführer: Maik Lorenz

Anwesende: Andrej Dederer; Georg Ebert; Stephan Kaden; Patrick Kunze; Maik Lorenz; Christoph Mathies; Jan Sutmöller

TOP 1:	Aufgabenliste	Status erledigt
Diskussion:	Alle TN geben den Stand Ihrer Aufgaben wieder. Themenverantwortliche berichten über neue Ergebnisse und Fortschritte.	
Ergebnis:	A4: Ausarbeitung eines Protokolllayouts Verantwortliche: ML Testprotokolllayout in Arbeit	Status: offen
	A5: Klärung ob Lasten-/Pflichtenheft als Einzel- oder Gesamt-dokumentation erstellt werden soll, Rücksprache mit Prof. Verantwortliche: JS Dokumentation als Ganzes abgeben, Teilabsprache im Vorfeld mit Prof. möglich → klaerung_A5.pdf	Status: erledigt

<p>Änderungswunsch ML; 15.05.2015</p>	<p>A9: Meilensteine grafisch darstellen Verantwortliche: SK <i>Status: offen</i> Wird per Mail nachgereicht, neues Zeitziel: 13.05.2015</p> <p>A10: Vorlage Lasten-/ Pflichtenheft nach IEEE 830 Verantwortliche: SK <i>Status: offen</i> Vorlage erstellt, Inhalte werden nun mit JS abgesprochen und zusammengeschrieben; Teilgebiete werden an Verantwortliche zugesandt; Ablage in GitHub → top_dokumentvorlage.pdf (folgt)</p> <p>GE: Programmieren</p> <ul style="list-style-type: none"> - Teilbereich „News“ und „Noten“ funktionieren - Zeigt aktuellen Stand der Programmierung direkt an App - Diskussion: Soll Matrikelnummer verschlüsselt werden? Abstimmung: In App kann Matrikelnummer sichtbar sein (usability), gegen Zugriffe durch Dritte wird verschlüsselt. <p>ML: Design</p> <ul style="list-style-type: none"> - Layoutvorlage wurde präsentiert - Einbinden von Bildern, Schriften und Farben wurde geklärt, Fragment „News“ wurde farblich an Layout angepasst, - Aufgabenliste der Designgruppe wurde besprochen - Diskussion: Position von Logo und Button Abstimmung: Logo und Button am Bildschirmende <p>JS: GitHub</p> <ul style="list-style-type: none"> - Meilensteine wurden erstellt und Verantwortlichen zugeordnet; Vorstellung zum Vorgehen am direkten Erstellen von Aufgabe A11 - Noch nicht abgeschlossen, Ergebnis von A9 fehlt noch <p>AD, CM: UML</p> <ul style="list-style-type: none"> - AD: Vorstellen der aktuellen Diagramme - Klassendiagramm „News“ muss noch durch CM und GE zugestimmt werden, dann auf GitHub - Ablaufdiagramm der App begonnen
---	--

TOP 2:	<i>Sonstiges</i>
Diskussion:	Besprechung offener Themen und Fragen.
Ergebnis:	<p>Qualitätsmanagement</p> <ul style="list-style-type: none">- Alle Projektteilnehmer werden aufgefordert, die eingestellten Ergebnisse auf GitHub einzusehen und Fehler an PK zu melden.
	<p>App-Design</p> <ul style="list-style-type: none">- Anpassen der Menüpunkte an das Layout gewünscht (AD7)
	<p>Nächster Termin</p> <ul style="list-style-type: none">- Durch Präsenzphase keine Treffen am 17. bzw. 24.05. nächster fester Termin wird 07.06.2015. Sollten dazwischen Treffen gewünscht sein, wird man sich kurzfristig absprechen.

TOP 3:	<i>Nächster Termin</i>
Ergebnis:	Sonntag, 07.06.2015

Protokollempfänger: Alle Projektteilnehmer

Datum: 10.05.2015

Festlegung: 15.05.2015

Sitzungsprotokoll

Thema der Besprechung: Quellcodeanalyse, Erstellung UML

Datum: 12.5.2015

Zeit: 16:30

Ort: TeamViewer Meeting

Eingeladene Teilnehmer: Andrej Dederer; Christoph Mathies

Leitung: Christoph Mathies

Protokollführer: Andrej Dederer

Anwesende: Andrej Dederer; Christoph Mathies

TOP 1:	Quellcodeanalyse	<i>Status:[erledigt]</i>
Diskussion:	-Quellcode durchgegangen für: NewsActivity, NavigationDrawerFragment und NewsFragment -Xml-Ressourcen besprochen	
Ergebnis:	Ablauf der App beim Starten geklärt Zusammenhänge zwischen Activities, Views und Fragmenten geklärt Verständnis zwischen Ressourcen, Quellcode und Programmeinbindung verstanden	

TOP 2:	<i>Erstellung UML</i>
Diskussion:	Das bereits angefertigte Klassen und Aktivitätsdiagramm angeschaut und weitere Schritte festgelegt.
Ergebnis:	Für das Aktivitätsdiagramm soll der Fragment Lifecycle als Grundlage dienen geänderte Methoden werden dort mit eingefügt. Klassendiagramm wird aufgesplittet werden. Es wird EIN Klassendiagramm geben, welches die APP in der groben Funktionsweise zwischen Activities und Fragmente beschreibt. Dann wird es noch für jedes Fragment/Activity ein separates Klassendiagramm geben, was die Funktion im inneren genauer erklärt.
Aufgaben:	Aktivitätsdiagramm für den Unterpunkt News erstellen. Das Ergebnis dann auch als PDF auf Github hochladen. Verantwortliche: Andrej Termin: 17.05.2015
	(gezeichneter) Entwurf vom Klassendiagramm für die gesamte APP digitalisieren Verantwortliche: Chris Termin: 17.05.2015
TOP 3:	<i>Vorstellung des Handbuchentwurfs</i>
Diskussion:	Christoph stellte die grobe Struktur/Inhaltsverzeichnis des Handbuchs vor.
Ergebnis:	Die Android-Studio-Beschreibung(Installation usw.) soll nicht in dem Handbuch erläutert werden.
Aufgaben:	Beginnen mit der Beschreibung des Programmablaufs: Welche Methoden lösen was aus usw. - Erstellung des Inhaltsverzeichnisses - Erstellung der jeweiligen Abschnitte im Dokument mit Beschreibung, was jeweils beschrieben werden soll
	Verantwortliche: Christoph Termin: 17.5.2015

Protokollempfänger: Alle Projektteilnehmer

Datum: 13.05.2015

Festlegung: Offen

Sitzungsprotokoll

Thema der Vorlesung/Workshop - UML
Besprechung:

Datum: 20.05.2015

Zeit: 07:45

Ort: Präsenzveranstaltung

Eingeladene BKMI13,BWI13

Teilnehmer:

Leitung: Profn. Dr.-Ing. Sabine Wieland

Protokollführer: Jan Sutmöller

Anwesende: BKMI13, BWI13

TOP 1:	<i>Einführung UML</i>	<i>Status: [erledigt]/[offen]</i>
Diskussion:	Frau Wieland gibt eine Einführung zum Umgang mit dem Enterprise Architect.	
Ergebnis:	Setup-Dateien liegen im Ilias. Softwarelizenzierung läuft über einen Lizenzserver in der HfTL. Im Netz der FH ist dieser direkt zu erreichen. Ausserhalb des Netzes ist eine Einwahl über ein VPN notwendig. Für die Dokumentation der Projektarbeit werden folgende Diagramme gefordert: Klassendiagramm, Aktivitätsdiagramm, Use-Case-Diagramm, Sequenzdiagramm, Zustandsdiagramm	
Aufgaben:	Axx:	

	Verantwortliche:	Termin:
TOP 2:	<i>Workshop UML</i>	
Diskussion:	Die Studenten erstellen erste eigene UML Diagramme	
Ergebnis:	Anhand von Beispielen (Mitfahrglegenheit, Vegan-APP,HFTL-APP) werden passende Use-Case-Diagramme erstellt. Frau Wieland stellt zu jeder Diagrammart Beispiele vor.	
Aufgaben:	Von jedem der fünf Diagramme mindestens 2 erstellen. Ein Grobes und ein Detailiertes.	Verantwortliche: Projektteams Termin: Abgabe der Projektarbeit
TOP 3:	21.05.2015, 11:30	
Ergebnis:	Sonntag, xx.xx.xxxx	

Protokollempfänger: Alle Projektteilnehmer

Datum: 20.05.15

Festlegung: abgeschlossen

Sitzungsprotokoll

Thema der Besprechung: Entwicklung/ Status im Projekt HFTL-APP

Datum: 07.06.2015

Zeit: 19:30

Ort: TeamViewer Meeting

Eingeladene Stefan Czogalla; Andrej Dederer; Georg Ebert; Stephan
Teilnehmer: Kaden; Patrick Kunze; Maik Lorenz; Christoph Mathies; Jan Sutmöller

Leitung: Stefan Czogalla

Protokollführer: Maik Lorenz

Anwesende: Stefan Czogalla; Andrej Dederer; Georg Ebert; Patrick Kunze; Maik Lorenz; Christoph Mathies

TOP 1:	Aufgabenliste	Status erledigt
Diskussion:	Alle TN geben den Stand Ihrer Aufgaben wieder. Themenverantwortliche berichten über neue Ergebnisse und Fortschritte.	
Ergebnis:	A4: Ausarbeitung eines Protokolllayouts Verantwortliche: ML	Status: erledigt
	Protokoll für Testphase auf GitHub geladen, erste Test durchgeführt Anmerkung GE: Bei Fehlern im Test Fehlerlog mit einfügen, um Fehler reproduzierbar zu machen.	
	A9: Meilensteine grafisch darstellen Verantwortliche: SK	Status: offen
	Keine Zusendung erfolgt, ML übernimmt nach Rücksprache mit SK	

A10: Vorlage Lasten-/ Pflichtenheft nach IEEE 830

Verantwortliche: SK *Status: offen*

Keine weitere Info seit letztem Meeting, SC und/oder PK
übernehmen nach Klärung mit SK

A11: Verbleib der App im AppStore oder auf Website

Verantwortliche: SK *Status: offen*

Klärung durch Prof. mit Geschäftsleitung noch offen

A12: Klärung der CI-Grafiken als Flash

Verantwortliche: ML *Status: offen*

Durch Krankheit offen, neues Zeitziel: 21.06.15

A13: UML Diagramme in GitHub einstellen

Verantwortliche: CM, AD *Status: erledigt*

Diagramme wurden hochgeladen und können nun
gegengelesen werden.

AD4: Einbinden von Layoutdaten für App

Verantwortliche: SC, ML *Status: erledigt*

Layoutvorstellungen können in App umgesetzt werden

AD6: Formatierungen des Layouts zusammenfassen

Verantwortliche: ML *Status: offen*

Umrechnung der Pixelwerte noch offen, neues Zeitziel:
21.06.15

AD7: Menüfarben an CI anpassen

Verantwortliche: SC, ML, GE *Status: offen*

Niedrige Priorität, Zeitziel offen

AU1: Aktivitätsdiagramm für News erstellen

Verantwortliche: AD *Status: erledigt*
mit A13 geklärt, in GitHub: **Activity_StarteNews.pdf**

AU2: Klassendiagramm der gesamten App

Verantwortliche: CM *Status: erledigt*
Mit A13 geklärt, Teilung in einzelne Klassendiagramme
nach Fragmenten

AU3: Beschreibung des Programmablaufs

Verantwortliche: CM *Status: offen*

-

GE: Programmierung:

- SQL-Datenbank für Newsabfrage wurde erfolgreich eingebunden, News laden damit nun schneller
- Frage SC:** Werden alle Daten der App durch die Deinstallation gelöscht? → Ja.
- Frage SC:** Ist es gewollt, dass durch ein App-Update die Benutzerdaten gelöscht werden? → Nein, wird geklärt.
- Frage ML:** Ist es gewollt, dass nicht alle News geladen werden? → Nein, wird geklärt.

CM, AD: UML

- Erstellung der Diagramme läuft soweit problemlos, Zusammenhänge im Quellcode werden weitestgehend verstanden, sonst wird bei GE nachgefragt
- Rücksprache bei Prof. ergibt nur 10 Diagramme erforderlich, davon sind 6 bereits erstellt bis zum aktuellen Stand der App
- hauptsächlich wird das News-Fragment bearbeitet

PK: Qualitätsmanagement

- Diagramme und Protokolle wurden geprüft, Änderungen wurden direkt an Verantwortlichen gesendet
- GE wird gebeten die Klassendiagramme auf Aktualität zu prüfen (sind Namen noch aktuellen, etc.)

SC, ML: Design

- Wiedergabe des aktuellen Standes und Besprechung der Layoutvorstellungen
- Abstimmung:** Button mit abgerundeten Ecken

ML: Protokollierung

- Testprotokoll wurde erstellt und kann genutzt werden
- Nach jedem Merge wird ein Test durchgeführt, Ergebnisse werden in GitHub unter Dokumentation → Testprotokolle abgelegt
- **Anmerkung GE:** Aufgetretene Fehler mit Logfile belegen!

TOP 2:

Sonstiges

Diskussion:

Besprechung offener Themen und Fragen.

Ergebnis:

Qualitätsmanagement

- Alle Projektteilnehmer sollen weiter Fehler an PK melden

App-Tests

- Testprotokoll steht zur Verfügung, Installations-APK wurde von JS versendet, so dass jeder mit einem Android-Smartphone die App testen kann.
- Fehler während des Tests sollen mit Fehlermeldung bzw. Fehlerlog aus AndroidStudio belegt werden

Zeitmanagement

- Meetingsintervall vorerst 14 Tage, Grund sind weitere Aufgaben und PVL bis 28.KW

Aufgaben:

AD8: anpassen des Notenlayouts in App

Verantwortliche: SC, ML Termin: 21.06.15

AD9: Übernahme von Textformatierungen der News

Verantwortliche: SC, GE Termin: 21.06.15

AD10: Layout für Stundenplan erstellen
Verantwortliche: ML Termin: 21.06.15

TOP 3: *Nächster Termin*
Ergebnis: Sonntag, 21.06.2015

Protokollempfänger: Alle Projektteilnehmer

Datum: 08.06.2015

Festlegung: 12.06.2015

Sitzungsprotokoll

Thema der Besprechung: Entwicklung/ Status im Projekt HFTL-APP

Datum: 20.06.2015

Zeit: 16:00

Ort: TeamViewer Meeting

Eingeladene Teilnehmer: Stefan Czogalla; Maik Lorenz

Leitung: Stefan Czogalla

Protokollführer: Maik Lorenz

Anwesende: Stefan Czogalla; Maik Lorenz

TOP 1:	Aufgabenliste	Status erledigt
Diskussion:	Alle TN geben den Stand Ihrer Aufgaben wieder. Themenverantwortliche berichten über neue Ergebnisse und Fortschritte.	
Ergebnis:	Lasten-/ Pflichtenheft <ul style="list-style-type: none">- Besprechung der Inhalte, die an SK geleitet werden- Klärung zum Thema 4.3.3 und 4.3.4; Möglichkeit die Themen zusammen zufassen muss geklärt werden Bau der News <ul style="list-style-type: none">- SC klärt den Aufbau des News-Fragment, Zusammenhang der Dateien und was wichtig zur Anpassung an die Layoutvorlage war- Absprache zu Problemen und deren Behebung Absprache: ML übernimmt Anpassung der Noten	

Formatierungen

- ML präsentiert den Stand der aktuellen Layoutvorlage, die Anpassungen an das HfTL-CI/CD und die Android-Vorgaben
 - Aufbau und Abmessungen der Buttons werden an Vorgaben besprochen und in Android Studio angepasst
 - Umrechnungstabelle von px zu dp wird besprochen, weitere Anpassungen der Layoutvorlage mit diesen Daten wird geklärt

Stundenplan

- Möglichkeiten zum Aufbau des Stundenplans werden besprochen
 - Klärung zum Wochenwechsel in der App müssen mit GE besprochen werden, Aktuelle Idee ist Umsetzung mit DropDown-Auswahl

Aufgaben:

AD8: Anpassen des Notenlayouts

Verantwortliche: ML

Termin: 28.06.15

TOP 2:

Sonstiges

Diskussion:

Besprechung offener Themen und Fragen.

Ergebnis:

Startseite

Weitere Verwendung des Startbildschirms muss mit Projektgruppe geklart werden. Bisher startet News-Seite der App, ohne vorher die Startseite zu zeigen.

Anpassung der Startseite als Flash-Animation wird mittlerweile überbewertet. Die Klärung findet im Projektmeeting statt.

Stundenplan

Kennzeichnen der Fächer ist denkbar. Die Auswertung der ical-Datei ergab bis zu vier Kategorien: Praktikum, Prüfung, Seminar, Vorlesung
Möglichkeiten und Bedarf dieser Funktion wird mit Projektgruppe besprochen.

Protokollempfänger: Alle Projektteilnehmer

Datum: 22.06.2015

Festlegung: 27.06.2015

Sitzungsprotokoll

Thema der Besprechung: Entwicklung/ Status im Projekt HFTL-APP

Datum: 21.06.2015

Zeit: 19:30

Ort: TeamViewer Meeting

Eingeladene Stefan Czogalla; Andrej Dederer; Georg Ebert; Stephan

Teilnehmer: Kaden; Patrick Kunze; Maik Lorenz; Christoph Mathies; Jan Sutmöller

Leitung: Stefan Czogalla

Protokollführer: Maik Lorenz

Anwesende: Stefan Czogalla; Georg Ebert; Stephan Kaden; Patrick Kunze; Maik Lorenz; Jan Sutmöller

TOP 1:	Aufgabenliste	Status erledigt
Diskussion:	Alle TN geben den Stand Ihrer Aufgaben wieder. Themenverantwortliche berichten über neue Ergebnisse und Fortschritte.	
Ergebnis:	A9: Meilensteine grafisch darstellen Verantwortliche: SK	Status: erledigt
	Meilensteine als Gantt-Diagramm aufbereitet und verteilt. Änderungen werden aktuell umgesetzt.	
	A10: Vorlage Lasten-/ Pflichtenheft nach IEEE 830 Verantwortliche: SK	Status: erledigt
	Vorlage erstellt und an Projektteilnehmer verteilt; offene Abschnitte werden vom Zuständigen Bearbeiter ergänzt.	

A11: Verbleib der App im AppStore oder auf Website

Verantwortliche: SK *Status: offen*

Klärung durch Prof. mit Geschäftsleitung noch offen

A12: Klärung der CI-Grafiken als Flash

Verantwortliche: ML *Status: erledigt*

Abstimmung: Flash-Darstellung nicht mehr erwünscht.
Siehe Abschnitt „Design“

AD6: Formatierungen des Layouts zusammenfassen

Verantwortliche: ML *Status: erledigt*

Formatierungen wurden erstellt und werden in Lasten-/
Pflichtenheft eingebunden.

AD7: Menüfarben an CI anpassen

Verantwortliche: SC, ML, GE *Status: offen*

Niedrige Priorität, Zeitziel offen

AD8: anpassen des Notenlayout in der App

Verantwortliche: SC, ML *Status: offen*

Keine Zeit, neues Zeitziel: 28.06.15

AD9: Übernahme der Textformatierungen einer News

Verantwortliche: SC, GE *Status: offen*

Klärung läuft noch, neues Zeitziel 28.06.15

AD10: Layout für Stundenplan erstellen

Verantwortliche: ML *Status: erledigt*

Layout erstellt und in Formatierung bereits aufgenommen.

GE: Programmierung

- Bugfixes, App-Abstürze werden weiter unterbunden
 - KW-Auswahl im Stundenplan über Dropdown bereits in Testphase, evtl. Ladebildschirm einpflegen beim Wechsel
- Anmerkung ML:** Überschrift der Artikel ist "Termindetail"

SK: Meilensteine

- Gantt-Diagramm wurde verteilt. Bereits gesendete Änderungsvorschläge wurden angepasst, weitere Änderungen sind in Bearbeitung

	<p>SK, JS: Lastenheft</p> <ul style="list-style-type: none">- Ansprechpartner für einzelne Themen wurden nochmals informiert die fehlenden Textstellen zu ergänzen- Präsentation der aktuellen Version in GitHub durch JS <p>Anmerkung SC, ML: Punkt 4.3.3 und 4.3.4 der Vorlage können zusammengelegt werden, da sie durch die Formatierungsdatei beantwortet werden.</p> <p>SC, ML: Design</p> <ul style="list-style-type: none">- Präsentation zum Stand „Stundenplan“, Klärung des Aufbaus und Vorstellung der Formatierungen- Nutzen des Startbildschirms wurde überdacht, Alternative zum Ladebildschirm von Android ist der Startbildschirm, da der Nutzer nicht durch zu schnelle Bedienung einen Absturz provozieren kann. Abstimmung: Ein Startbildschirm bleibt gewünscht, die Form wird dem Designbereich überlassen. Abstimmung: Der Startbild muss nicht animiert sein. <p>PK: Qualitätsmanagement</p> <ul style="list-style-type: none">- Diagramme und Protokolle wurden weiter gesichtet und ASP entsprechend zu Änderungen informiert- Die Dokumentation wird geprüft, sobald die Zuständigen einer Prüfung zustimmen, nicht jedoch während der Entstehung <p>Anmerkung ML: Testprotokolle benötigen Test-ID, Absprache mit PK erfolgt</p> <p>Aufgaben:</p> <p>A14: Überschrift im Artikel anpassen Verantwortliche: GE, SC Termin: 28.06.15</p> <p>A15: Zeiten der Meilensteine anpassen Verantwortliche: SK Termin: 28.06.15</p> <p>A16: Diagramm an Vorgehensmodell anpassen Verantwortliche: SK Termin: 28.06.15</p> <p>A17: Name für Formatierungsdatei finden Verantwortliche: ML, SC Termin: 28.06.15</p> <p>A18: Klärung der Protokoll-ID Verantwortliche: ML, PK Termin: 28.06.15</p>
--	---

TOP 2:	<i>Sonstiges</i>
Diskussion:	Besprechung offener Themen und Fragen.
Ergebnis:	<p>Stundenplan Wochenwechsel</p> <ul style="list-style-type: none"> - Ein Wechsel der Kalenderwochen (KW) in die Zukunft und die Vergangenheit ist schwieriger in der Umsetzung, stattdessen wird vom aktuellen Tag die KW plus die nächsten 3 oder 6 Wochen als Dropdown-Auswahl angezeigt. <p>Stundenplan exportieren</p> <ul style="list-style-type: none"> - Das Exportieren des Stundenplans aus der App in den Kalender ist nicht Bestandteil der Anforderung und wird daher noch nicht umgesetzt. Es kann zu einem späteren Zeitpunkt ergänzt werden. <p>UML Anpassungen</p> <ul style="list-style-type: none"> - Programmieränderungen müssen noch in UML-Diagrammen umgesetzt werden → Custom Adapter fehlt - SK benötigt weitere Angaben zu den offenen UML-Diagrammen
Aufgaben:	<p>AU4: UML-Diagramme überarbeiten</p> <p>Verantwortliche: CM, AD Termin: 28.06.15</p>

TOP 3:	<i>Nächster Termin</i>
Ergebnis:	Sonntag, 28.06.2015

Protokollempfänger: Alle Projektteilnehmer

Datum: 22.06.2015

Festlegung: 27.06.2015

Sitzungsprotokoll

Thema der Besprechung: Entwicklung/ Status im Projekt HFTL-APP

Datum: 28.06.2015

Zeit: 19:30

Ort: TeamViewer Meeting

Eingeladene Stefan Czogalla; Andrej Dederer; Georg Ebert; Stephan
Teilnehmer: Kaden; Patrick Kunze; Maik Lorenz; Christoph Mathies; Jan Sutmöller

Leitung: Stefan Czogalla

Protokollführer: Maik Lorenz

Anwesende: Stefan Czogalla; Andrej Dederer; Georg Ebert; Stephan
Kaden; Patrick Kunze; Maik Lorenz; Christoph Mathies; Jan
Sutmöller

TOP 1:	Aufgabenliste	Status erledigt
Diskussion:	Alle TN geben den Stand Ihrer Aufgaben wieder. Themenverantwortliche berichten über neue Ergebnisse und Fortschritte.	
Ergebnis:	A11: Verbleib der App im AppStore oder auf Website Verantwortliche: SK Klärung durch Prof. mit Geschäftsleitung noch offen Alternative: App-apk wird auf GitHub abgelegt	<i>Status: offen</i>
	A14: Überschriften der Artikel anpassen Verantwortliche: GE, SC Andere Aufgaben priorisiert	<i>Status: offen</i>

A15: Zeiten der Meilensteine anpassen

Verantwortliche: SK *Status: erledigt*

Zeiten an Projektablauf angepasst

A16: Gantt-Diagramm an Vorgehensmodell anpassen

Verantwortliche: SK *Status: erledigt*

Ablauf an Prototypen orientiert

A17: Name für Formatierungsdatei finden

Verantwortliche: ML, SC *Status: erledigt*

HFTL-App_Designvorlage.pdf

A18: Klären der Testprotokoll-ID

Verantwortliche: ML, PK *Status: erledigt*

Testprotokolle je nach App-Modul und für Gesamtsystem erstellt, Ablage auf GitHub > Dokumentation > Testprotokolle; Die Test-ID wird im Protokollkopf vermerkt, eine Übersicht der Tests liegt ebenfalls auf GitHub, daraus kann jeder sehen, welche ID als nächstes genutzt werden soll

GE: Programmierung

- Kleine Bugfixes nach Änderungen am Notenlayout, die dazu führten, dass App immer abstürzte
- Funktion des Stundenplans wurde weiter bearbeitet, Daten werden zwischengespeichert und ermöglichen schnelles laden

SC, ML: Design

- Aktuellen Stand präsentiert: Notenlayout wurde an Vorlage angepasst, kleine Formatierungen müssen noch erledigt werden, Aufbau des Stundenplans wurde besprochen
Festlegung: Stundenplan soll nach anklicken im Dropdown-Menü sofort laden und nicht durch Klick auf Go

CM, AD: UML + Handbuch

- Aktuellen Stand besprochen: noch ein Zustandsdiagramm in Arbeit, Klärung bis 19.07.
 - Anpassungsaufgaben der UML-Daten an Programmieränderungen sind im Gange
 - Handbuch in Arbeit, wird mit SK abgesprochen

SK, JS: Lastenheft

- Aktuellen Stand präsentiert: offene Teilbereiche wurden ergänzt, der Stand ist auf GitHub abrufbar
 - Anpassungen der LaTeX-Version

JS: GitHub

- App-Release 4 erstellt mit Stand 28.06.15
 - Allgemeine Anpassungen auf GitHub
 - Dokumentation in LaTeX weiter ergänzt

Anmerkung: Verantwortliche sollen auf GitHub die Issues abschließen, wenn die Themen erledigt sind. Ebenfalls sollen erledigte Branches geschlossen werden.

Aufgaben:

AD12: Anordnung der Semester anpassen

Verantwortliche: SK

Termin: 19.07.15

TOP 2:

Sonstiges

Diskussion:

Besprechung offener Themen und Fragen.

Ergebnis:

Nächstes Projektmeeting

- Präsenzphase in der 28.KW wird zum Meeting genutzt und dort der nächste Termin besprochen

Protokollempfänger: Alle Projektteilnehmer

Datum: 30.06.2015

Festlegung: 04.07.2015

Sitzungsprotokoll

Thema der Besprechung: Entwicklung/ Status im Projekt HFTL-APP

Datum: 02.08.2015

Zeit: 20:00

Ort: TeamViewer Meeting

Eingeladene Stefan Czogalla; Andrej Dederer; Georg Ebert; Stephan
Teilnehmer: Kaden; Patrick Kunze; Maik Lorenz; Christoph Mathies; Jan Sutmöller

Leitung: Stefan Czogalla

Protokollführer: Maik Lorenz

Anwesende: Stefan Czogalla; Andrej Dederer; Georg Ebert; Stephan
Kaden; Patrick Kunze; Maik Lorenz; Christoph Mathies; Jan
Sutmöller

TOP 1:	<i>Aufgabenliste</i>	<i>Status erledigt</i>
Diskussion:	Alle TN geben den Stand Ihrer Aufgaben wieder. Themenverantwortliche berichten über neue Ergebnisse und Fortschritte.	
Ergebnis:	<p>A11: Verbleib der App im AppStore oder auf Website Verantwortliche: SK <i>Status: erledigt</i></p> <p>Nach Rücksprache mit dem RZ, sowie der Abteilung für Öffentlichkeitsarbeit der HFTL wurden Möglichkeiten zur Ablage der App auf der Homepage besprochen. Es wird vorerst nicht mehr versucht, die App in den Playstore zu laden.</p> <p>A14: Überschriften der Artikel anpassen Verantwortliche: GE, SC <i>Status: offen</i></p> <p>Andere Aufgaben priorisiert</p>	

GE: Programmieren

- Fehlerbehebungen, bspw. wenn keine Studiengruppe ausgewählt oder falsche Nutzerdaten gespeichert wurden
- Push-Service bei Notenabfrage noch nicht optimal, es wird eine Möglichkeit gesucht, die Funktion besser zu implementieren

SC, ML: Design

- Design der App fertig an Formatvorlage angepasst
- Kleine Änderungen in bestehenden Formatierungen und Texten; onClick-Funktion für Aktualisieren Button implementiert
- Design- und Programmiergruppe werden sich wegen fehlender Icons zusammensetzen und Wichtigkeit klären
- Formatvorlage muss an Dokumentationsgruppe

AD12: Anordnung der Semester

Verantwortliche: SC Status: offen

Andere Aufgaben priorisiert

JS: GitHub

- Alle Teilnehmer sollen aktiv nach ihren offenen Issues schauen, diese bearbeiten und wenn möglich schließen
- Für das Versionsmanagement werden die Daten der erstellten Prototypen aus GitHub verwendet

SK, JS: Lastenheft

- Anpassungen der LaTeX-Version an Software Requirements Specification_Stand_28062015, so wie Anpassung der Autoren
- Offene Punkte: Design wird mit A20 geklärt, Versionsmanagementprotokolle in A22
- Gantt-Diagramm zum Zeitmanagement eingepflegt, Vorgehensmodell wird daraus ersichtlich und später in Projektpräsentation genannt

CM, AD: UML + Handbuch

- Alle Diagramme sind nun auf GitHub, Qualitätscheck wird durchgeführt und Fehler gemeldet
- Schriftlicher Teil vom Handbuch abgeschlossen, Prüfung auf inhaltliche Korrektheit noch offen

PK, ML: Qualitätsmanagement + Protokolle

- Projektdokumente weiter gesichtet und geprüft, Fehler wurden entsprechend gemeldet
 - Testprotokolle wurden abschließend bearbeitet und liegen nun auf GitHub
 - Durchgeführte Tests werden in der **Testübersicht.xlsx** auf GitHub dokumentiert
 - **Festlegung:** Durch Änderungen an der HFTL-Website und QIS sind alte Prototypen nicht mehr prüfbar (Die App stürzt ab). Tests ab dem 02.08.15 finden mit dem letzten Prototyp statt. Die Projektteilnehmer sollen jedes Modul mindestens einmal testen.

Aufgaben:

A19: Push-Service vom Modul „Noten“ optimieren

Verantwortliche: GE Termin: 09.08.15

A20: Formatvorlage abschließen und versenden

Verantwortliche: ML Termin: 09.08.15

A21: GitHub Issues bearbeiten/ abschließen

Verantwortliche: alle Termin: 09.08.15

A22: Klärung zum Thema

Versionsmanagementprotokolle

Verantwortliche: PK Termin: 09.08.15

A23: UML Diagramme prüfen, Fehler melden

Verantwortliche: PK Termin: 09.08.15

A24: beschriebene Methoden im Handbuch prüfen und Fehler melden

Verantwortliche: GE Termin: 09.08.15

Ergänzung ML;

09 08 2015

A25: Mindestens ein Test pro Modul auf Basis der vorhandenen apk durchführen.

Verantwortliche: alle Termin: 13.09.15

AD13: Klärung zum Thema offene Icons

Verantwortliche: GE SC MI Termin: offen

Protokollempfänger: Alle Projektteilnehmer

Datum: 02.08.2015

Festlegung: 09.08.2015

Sitzungsprotokoll

Thema der Besprechung: Entwicklung/ Status im Projekt HFTL-APP

Datum: 09.08.2015

Zeit: 20:00

Ort: TeamViewer Meeting

Eingeladene Stefan Czogalla; Andrej Dederer; Georg Ebert; Stephan
Teilnehmer: Kaden; Patrick Kunze; Maik Lorenz; Christoph Mathies; Jan Sutmöller

Leitung: Stefan Czogalla

Protokollführer: Maik Lorenz

Anwesende: Stefan Czogalla; Andrej Dederer; Georg Ebert; Stephan
Kaden; Patrick Kunze; Maik Lorenz; Jan Sutmöller

TOP 1:	<i>Aufgabenliste</i>	<i>Status erledigt</i>
Diskussion:	Alle TN geben den Stand Ihrer Aufgaben wieder. Themenverantwortliche berichten über neue Ergebnisse und Fortschritte.	
Ergebnis:	A19: Push-Service vom Modul „Noten“ optimieren Verantwortliche: GE	<i>Status: erledigt</i>
	Es wird nun geprüft, ob der Dienst schon mal gestartet war, sollte die App oder das Smartphone neu gestartet werden. Dafür verlangt die App nun eine weitere Berechtigung zum Start beim Systemstart. Der Dienst wurde unter verschiedenen Bedingungen erfolgreich getestet.	
	A20: Formatvorlage abschließen und versenden	
	Verantwortliche: ML	<i>Status: erledigt</i>

Die Designvorlage wurde abschließend besprochen.
Fragen zum Thema „Schriften“ wurden geklärt. Die Datei und Bilder werden im „Share-Ordner“ bereitgestellt.

A21: GitHub Issues bearbeiten/ abschließen

Verantwortliche: alle *Status: erledigt*
Offene Issues die bereits beendet waren, wurden geschlossen. Es befinden sich nur noch Issues auf GitHub an denen noch gearbeitet wird.

A22: Klärung zum Thema Versionsmanagementprotokolle

Verantwortliche: PK *Status: offen*
Inhalt und erforderliche Dateien wurden nach Rücksprache mit SK geklärt. Neues Zeitziel: 16.08.15

A23: UML Diagramme prüfen, Fehler melden

Verantwortliche: PK *Status: offen*
Neues Zeitziel: 16.08.15

A24: Beschriebene Methoden im Handbuch prüfen und Fehler melden

Verantwortliche: GE *Status: erledigt*
Rücksprache mit AD erfolgt und entsprechende Änderungen durchgeführt.

A25: Mindestens ein Test pro Modul durchführen

Verantwortliche: alle *Status: offen*
Erneuter Aufruf an alle, Tests mit den vorhandenen APK's durchzuführen.

A26: Inhalt des Impressums zusammenschreiben und Menüpunkt in der App einbauen.

Verantwortliche: SK, GE, SC *Status: offen*
Inhalte wurden besprochen und verfasster Text von SK wurde besprochen. Das Ergebnis wird im „Share-Ordner“ bereitgestellt. Neues Zeitziel: 16.08.15

A27: Abschlusspräsentation

Verantwortliche: alle

Status: offen

PK erstellt entsprechende Folien zum Inhalt der Präsentation, als Präsentatoren melden sich SC, SK, JS und ML. Die weitere Ausarbeitung erfolgt zwischen den Präsentatoren.

GE: Programmieren

- Einbinden und Test des Push-Services der Noten
- Langzeittest über 1 Woche in Planung

ML: Design

- Fertigstellung der Designvorlage und entsprechender Bilder

JS: GitHub

- Keine Änderungen erforderlich gewesen, Verantwortliche sind weiter angehalten ihre Issues abzuschließen

JS, SC, SK, AD, CM: Dokumentation

- Lasten- & Pflichtenheft, Benutzer- & Entwicklerhandbuch weiter ergänzt und Fehler ausgebessert. Offene Stellen wurden markiert und werden noch bearbeitet.

PK: Qualitätsmanagement

- Testprotokolle an neue Vorlage angepasst und Testübersicht.xlsx aktualisiert

Aufgaben:

AD14: Rückfalloption auf Schriftart ‚Arial‘ prüfen

Verantwortliche: SC

Termin: 16.08.15

A28: Offene Stellen im Entwicklerhandbuch mit Inhalt füllen (allgemeiner Text zu Activities und Fragmenten)

Verantwortliche: alle

Termin: 16.08.15

A29: Kalendersynchronisierung des Stundenplans

Verantwortliche: GE

Termin: 16.08.15

A30: Inhalte der Präsentation zusammentragen

Verantwortliche: PK

Termin: 16.08.15

TOP 3:	<i>Nächster Termin</i>
Ergebnis:	Sonntag, 16.08.2015

Protokollempfänger: Alle Projektteilnehmer

Datum: 10.08.2015

Festlegung: Offen

Sitzungsprotokoll

Thema der Besprechung: Entwicklung/ Status im Projekt HFTL-APP

Datum: 16.08.2015

Zeit: 20:00

Ort: TeamViewer Meeting

Eingeladene Stefan Czogalla; Andrej Dederer; Georg Ebert; Stephan
Teilnehmer: Kaden; Patrick Kunze; Maik Lorenz; Christoph Mathies; Jan Sutmöller

Leitung: Stefan Czogalla

Protokollführer: Stefan Czogalla

Anwesende: Stefan Czogalla; Andrej Dederer; Georg Ebert; Patrick Kunze;
 Christoph Mathies; Jan Sutmöller

TOP 1:	<i>Aufgabenliste</i>	<i>Status: [erledigt]/[offen]</i>	
Diskussion:	Alle TN geben den Stand Ihrer Aufgaben wieder. Themenverantwortliche berichten über neue Ergebnisse und Fortschritte.		
Ergebnis:	<p>A22: Klärung zum Thema Versionsmanagementprotokolle</p> <p>Verantwortlicher: PK</p> <p>Überschrift von 4.5.3 passt nicht zum angeforderten Inhalt „Versionsmanagementprotokolle“. Diese sind durch JS in Punkt 4.7.4 „Release-Historie“ eingefügt. PK klärt: Thematische Überschneidung, kann Punkt 4.5.3 entfallen?</p>	<i>Status: offen</i>	
		<p>Neues Ziel: 30.08.2015</p> <p>A23: UML Diagramme prüfen, Fehler melden</p> <p>Verantwortliche: PK</p> <p>UML wurden geprüft und aktualisiert.</p>	<i>Status: erledigt</i>

A25: Mindestens ein Test pro Modul durchführen

Verantwortliche: alle

Status: offen

Die neue APK-Version 6.0 gilt ab sofort als Textgrundlage. Nur die „00_Testübersicht.xlsx“ im Hauptordner nutzen, die anderen in den Unterordnern sind nicht aktuell.

A26: Inhalt des Impressums zusammenschreiben und Menüpunkt in der App einbauen

Verantwortliche: SK, SC, GE

Status: erledigt

Das Impressum wurde erstellt, in die App eingefügt und mit dem aktuellen Layout versehen. Kritik am Inhalt des Impressums (Standard-Impressum für Webseiten), da dies nicht wirklich auf die Funktionalitäten der App anwendbar ist.

A27: Abschlusspräsentation

Verantwortliche: alle

Status: offen

PK hat ersten groben Entwurf getätigt und wird dies bis zum nächsten Termin ausbauen. Alle sind dazu angehalten, den Entwurf zu überprüfen und ggf. Feedback zu geben. Die Präsentationsdatei liegt im neu erstellten Ordner „Präsentation“ im Projektordner „Dokumentation“
Neues Ziel: 30.08.2015

A28: Offene Stellen im Entwicklerhandbuch mit Inhalt füllen (allgemeiner Text zu Activities und Fragmenten)

Verantwortliche: CM

Status: offen

Inhalt wird an SC zur Einarbeitung übergeben.
Neues Ziel: 30.08.2015

A29: Kalendersynchronisierung des Stundenplans

Verantwortlicher: GE

Status: erledigt

Die Funktionalität ist bereits mit APK-Version 6 implementiert. Zu finden im Stundenplan -> Auswahl der gewünschten KW -> rechter Menüpunkt -> „Mit Kalender synchronisieren“

A30: Inhalte der Präsentation zusammentragen

Verantwortliche: PK , GE *Status: offen*

Punkt „Umsetzung der Architektur“ sollte durch GE befüllt werden.

Weiteres s. A27!

Neuer Zieltermin: 30.08.2015

GE: Programmieren

Implementierung einer Erinnerungsfunktion, 15 Minuten vor Start des Termins, für den Kalender. (s. A31)

SC: Design

Schriftarten nach CI/CD konnten erfolgreich implementiert werden. Das Speicherproblem konnte durch Erstellen einer neuen Klasse umgangen werden. Funktionalität in APK-Version 6 bereits integriert.

AD14 (Rückfall auf „Arial“) entfällt dadurch.

AD12 Anordnung der Semester abgeschlossen, bereits in APK-Version 6 integriert.

JS: Github

APK-Version 6 wurde freigegeben

JS, SC, SK, CM, AD: Dokumentation

Punkt 4.3.4 entfällt, da durch 4.3.3 abgedeckt.

4.7.3 Projektstrukturplan: noch offen (s. A32)

Einfügen der Testprotokolle in den Anhang (s. A33)

Entwicklerhandbuch wird final auf Basis APK-Version 6 fertiggestellt. (s. A34)

Offene Punkte (verwendete Software) wird nachgereicht (s. A35)

PK: Qualitätsmanagement
Prüfung der bereits erledigten Punkte in der Projektdokumentation. (s. A36)
Prüfung ob Anforderungen nach alternativePL_Aufgabenbeschreibung.pdf in Dokumentation vorhanden sind. (s. A37)

CM, AD: UML

Aufgaben:

A31: Erinnerungsfunktion für anstehende Ereignisse

Verantwortliche: GE Termin: 30.08.2015

Einstellungsmöglichkeit, ob 15 Minuten vor Termin eine Erinnerung (Push-Nachricht?) erscheinen soll.

A32: Doku – 4.7.3 Projektstrukturplan

Verantwortliche: CM, AD Termin: 30.08.2015

Prüfen auf Unterschiede zur Projektorganisation und ggf. Erstellung eines Projektstrukturplans. ggf kann Hilfe durch ML eingeholt werden.

A33: Doku – Einfügen der Testprotokolle in den Anhang

Verantwortliche: JS Termin: 30.08.2015

A34: Doku – Entwicklerhandbuch wird final auf Basis APK-Version 6 fertiggestellt

Verantwortliche: SC Termin: 30.08.2015

A35: Doku – Offene Punkte (verwendete Software)

Verantwortliche: CM, AD Termin: 30.08.2015

A36: QM – Prüfung der bereits erledigten Punkte in der Projektdokumentation

Verantwortliche: PK Termin: 30.08.2015

A37: QM – Prüfung ob Anforderungen in Dokumentation vorhanden

Verantwortliche: PK Termin: 30.08.2015

Pflichtangaben gemäß:
alternativePL_Aufgabenbeschreibung.pdf (s. Ordner Informaterial)

A38: UML – Prüfung ob UML auf Stand APK-Version 6 sind.

Verantwortliche: CM, AD Termin: 30.08.2015

TOP 2:	<i>Sonstiges</i>
Diskussion:	Besprechung offener Themen und Fragen.
Ergebnis:	Die Bewertungskriterien wurden gemeinsam angeschaut und diskutiert. Die App ist weitgehend fertiggestellt. Es muss nur noch die Nacharbeit (Doku, etc) fertiggestellt werden.
Aufgaben:	Keine

	Verantwortliche:	Termin:
TOP 3:	<i>Nächster Termin</i>	
Ergebnis:	Sonntag, 30.08.2015	

Protokollempfänger: Alle Projektteilnehmer

Datum: 16.08.2015

Festlegung: 30.08.2015

Sitzungsprotokoll

Thema der Besprechung: Entwicklung/ Status im Projekt HFTL-APP

Datum: 30.08.2015

Zeit: 20:00

Ort: TeamViewer Meeting

Eingeladene Stefan Czogalla; Andrej Dederer; Georg Ebert; Stephan

Teilnehmer: Kaden; Patrick Kunze; Maik Lorenz; Christoph Mathies; Jan Sutmöller

Leitung: Stefan Czogalla

Protokollführer: Maik Lorenz

Anwesende: Stefan Czogalla; Andrej Dederer; Georg Ebert; Stephan
Kaden; Patrick Kunze; Maik Lorenz; Christoph Mathies

TOP 1:	<i>Aufgabenliste</i>	<i>Status erledigt</i>
Diskussion:	Alle TN geben den Stand Ihrer Aufgaben wieder. Themenverantwortliche berichten über neue Ergebnisse und Fortschritte.	
Ergebnis:	A22: Klärung zum Thema Versionsmanagementprotokolle Verantwortliche: PK	<i>Status: erledigt</i>
	Inhalte durch PK in Dokumentation eingebracht und auf GitHub geladen; JS klärt ob richtige Dateien auf GitHub liegen	
	A25: Mindestens ein Test pro Modul durchführen	
	Verantwortliche: alle	<i>Status: offen</i>
	Neue Grundlage bleibt APK6 bis 06.09.15 nun alle	

A28: Offene Stellen im Entwicklerhandbuch

Verantwortliche: alle *Status: erledigt*

Änderungen liegen auf GitHub

A30: Inhalte der Präsentation zusammentragen

Verantwortliche: PK *Status: offen*

Andere Aufgaben, neues Zeitziel: 06.09.15

A31: Erinnerungsfunktion für anstehende Ereignisse

Verantwortliche: GE *Status: offen*

Aufgabe war optional, neues Zeitziel: offen

A32: Projektstrukturplan

Verantwortliche: CM, AD *Status: offen*

Skizze wurde angefertigt, aktuelles Organigramm als Basis, ML erstellt bis 06.09.15

A33: Einfügen der Testprotokolle

Verantwortliche: JS *Status: erledigt*

Ergebnis auf GitHub in Dokumentation

A34: Entwicklerhandbuch final auf Basis APK-Version 6 fertig

Verantwortliche: SC *Status: erledigt*

Angepasst auf APK6 und aktuelle GitHub-Version

A35: Doku – Offene Punkte

Verantwortliche: CM, AD *Status: erledigt*

Angepasst und auf GitHub gespeichert

A36: Prüfung der bereits erledigten Punkte in der Projektdokumentation

Verantwortliche: PK *Status: erledigt*

Rechtschreibung und andere kleine Fehler ausgebessert

A37: Prüfung ob Anforderungen in Dokumentation vorhanden

Verantwortliche: PK

Status: offen

rotestellen konnten noch nicht geprüft werden, Info an
Verantwortliche muss noch mal raus (u.a. 4.5.3., 4.6.5.)

A38: Prüfung ob UML auf Stand APK-Version 6 sind

Verantwortliche: CM, AD

Status: erledigt

Änderungen liegen auf GitHub

GE, SC: Programmieren

- Icons eingebunden; Formatierungen angepasst
- Ordnerstruktur im Projekt angepasst (neue Packages zu Übersichtlichkeit); alten, auskommentierten Quellcode entfernt; alte Bilder gelöscht

SC, ML: Design

- Icons für Statusleiste erstellt
- Einbinden der Formatierungen aus den News
- [Newstexte](#) können ausgewählt werden (Share von Textpassagen)
- Branch Design nun geschlossen

CM, AD, SC: Dokumentationen

- siehe A28, A32, A33, A34, A35
- Text-Formatierungen im Entwicklerhandbuch müssen noch angepasst werden
- Querverweise von Schlagworten in entsprechende Kapitel

PK: Qualitätsmanagement

- siehe A30, A36, A37
- PK bekommt Entwicklerhandbuch bis 06.09.15 zu Prüfung

ML: Protokolle

- Wiederholung zum Einbinden der Testprotokolle in die Übersicht, im Zweifel schon mal „Platz reservieren“ und Protokoll nachreichen

Änderungswunsch SC,
06.09.2015

Aufgaben:	Keine
TOP 2:	<i>Sonstiges</i>
Diskussion:	Besprechung offener Themen und Fragen.
Ergebnis:	Qualitätsmanagement Alle Teilnehmer sind nun angehalten vorhandene Dokumente zu prüfen und Fehler an das Qualitätsmanagement zu melden.
Aufgaben:	Keine
TOP 3:	<i>Nächster Termin</i>
Ergebnis:	Sonntag, 06.09.2015

Protokollempfänger: Alle Projektteilnehmer

Datum: 30.08.2015

Festlegung: 06.09.2015

Sitzungsprotokoll

Thema der Besprechung: Entwicklung/ Status im Projekt HFTL-APP

Datum: 06.09.2015

Zeit: 20:00

Ort: TeamViewer Meeting

Eingeladene Stefan Czogalla; Andrej Dederer; Georg Ebert; Stephan
Teilnehmer: Kaden; Patrick Kunze; Maik Lorenz; Christoph Mathies; Jan Sutmöller

Leitung: Stefan Czogalla

Protokollführer: Maik Lorenz

Anwesende: Stefan Czogalla; Andrej Dederer; Georg Ebert; Stephan
Kaden; Patrick Kunze; Maik Lorenz; Christoph Mathies; Jan
Sutmöller

TOP 1:	<i>Aufgabenliste</i>	<i>Status erledigt</i>
Diskussion:	Alle TN geben den Stand Ihrer Aufgaben wieder. Themenverantwortliche berichten über neue Ergebnisse und Fortschritte.	
Ergebnis:	A25: Mindestens ein Test pro Modul durchführen Verantwortliche: alle Tests für APK6 sind eingestellt.	<i>Status: erledigt</i>
	A30: Inhalte der Präsentation zusammentragen Verantwortliche: PK Präsentation als Version 1.0 auf GitHub, muss noch an Layout angepasst werden.	<i>Status: erledigt</i>

A31: Erinnerungsfunktion für anstehende Ereignisse

Verantwortliche: GE

Status: erledigt

Option bleibt optional und wird eine Funktion für die spätere Entwicklung

A32: Projektstrukturplan erstellen

Verantwortliche: AD, CM, ML *Status: erledigt*

Auf GitHub unter
,Dokumentation\shared\swe_projektstukturplan.pdf'

A37: Prüfung ob Anforderungen in Dokumentation vorhanden

Verantwortliche: PK

Status: erledigt

Änderungen erfolgt und eingepflegt

GE, SC: Programmierung

- Bugfixes aus den Testprotokollen eingespielt (auf Git)
- Ladebalken sind jetzt überall
- Timeout für Newsabfrage wurde programmiert, da die HFT-Seite zuweilen langsam ist
- Geplante Bugfixes konnten mangels Reproduzierbarkeit nicht realisiert werden:
 - o Absturz der App, wenn ein Maillink geöffnet wird, aber kein Mailclient auf dem Smartphone installiert ist
 - o Absturz der App, wenn als Link ein unbekannter Pfad verwendet wird (bsp.: href="/de/service/formular.html")
 - o Beide Bugs wurden im Entwicklerhandbuch aufgenommen

JS, SC, CM: Dokumentation

- Offene Punkte werden noch abgearbeitet
- Bedienungsanleitung wurde aktualisiert
- Entwicklerhandbuch weitgehend fertig
- Einbinden der neuen Testprotokolle noch offen
- Einbinden der Projektstrukturpläne noch offen
- Letzte Punkte werden direkt mit Personen geklärt:
 - o UML in Entwicklerhandbuch (JS & SC)
 - o Referenz (4.7.3??) auf Gantt-Diagramm (JS)
 - o Korrektur der durch QM beanstandeten Punkte

	<p>AD, CM: UML</p> <ul style="list-style-type: none"> - UML an Bugfixes angepasst - QM hat abgesegnet <p>SC, ML: Design</p> <ul style="list-style-type: none"> - Keine ToDos mehr (ferdsch) <p>PK: Qualitätsmanagement</p> <ul style="list-style-type: none"> - Fehler in Dokumentation wurden kenntlich gemacht - UML wurden geprüft und freigegeben - Finale Prüfung Entwicklerhandbuch offen
Aufgaben:	Keine
TOP 2:	<i>Sonstiges</i>
Diskussion:	Besprechung offener Themen und Fragen.
Ergebnis:	<p>Abschlusspräsentation</p> <ul style="list-style-type: none"> - Finale Version von PK liegt auf Git, Layout übernimmt ML <p>Allgemeines:</p> <ul style="list-style-type: none"> - Große Projektmeetings gibt es nicht mehr <ul style="list-style-type: none"> o Offene Punkte werden nun bilateral zwischen den Verantwortlichen geklärt - Präsentationsteam trifft sich am Dienstag (Einladung wurde bereits versandt) - Neuer Prototyp Version 7 <ul style="list-style-type: none"> o Nach Möglichkeit sollten noch Tests durchgeführt werden, damit evtl. Bugs behoben werden können.

Protokollempfänger: Alle Projektteilnehmer

Datum: 08.09.2015

Festlegung: Offen