

M223: Multiuser Applikation

Probe IPA

1 Inhaltsverzeichnis

2	Vorwort	4
3	Organisation der Arbeitsergebnisse	4
4	Teil 1	4
4.1	Aufgabenstellung.....	4
4.2	Projektaufbauorganisation.....	4
4.3	Mittel und Methoden.....	5
4.4	Vorkenntnisse.....	5
4.5	Vorarbeit.....	6
4.6	Deklaration der benützten Firmenstandards	6
4.7	Zeitplan.....	6
4.8	Arbeitsprotokoll.....	7
5	Teil 2	10
5.1	Abstract	Fehler! Textmarke nicht definiert.
5.2	Informieren.....	10
5.3	Planen.....	10
5.3.1	Zeitplan/ Arbeitsaufteilung	11
5.3.2	Testing	12
5.3.3	UMLs.....	12
5.4	Entscheiden	14
5.4.1	Projekt / Technologie	14
5.4.2	Design	16
5.5	Realisieren	16
5.5.1	Persistenz.....	17
5.5.2	Sensor	Fehler! Textmarke nicht definiert.
5.5.3	Netzwerkfunktionalität	17
5.6	Kontrollieren.....	18
5.6.1	Testing	18
5.7	Auswerten	18
5.7.1	Zeitplan (Soll/Ist Vergleich)	18
5.7.2	Reflexion.....	18
5.8	Glossar	20
5.9	Quellenverzeichnis	20
5.10	Abbildungsverzeichnis.....	20
5.11	Tabellenverzeichnnis.....	20
5.12	Anhang.....	20

2 Vorwort

3 Organisation der Arbeitsergebnisse

4 Teil 1 – Umfeld und Ablauf

4.1 Aufgabenstellung

Im Rahmen des ÜK Moduls 223 soll eine Multiuser Applikation erstellt werden.

Folgende Rahmenbedingungen wurde durch den Modulleiter Remo Steinmann vorgegeben.

- Objektorientierte Multi-User-Applikation
- Umfang gemäss Planung (ca. 3 Tage Entwicklung, 2.5 Tage Dokumentation; 6h pro Tag)
- Zentrale Datenbank
- Mehrere Clients müssen gleichzeitig auf den gleichen Datenbestand zugreifen
- Zentrale Benutzer- und Rechte-Verwaltung

Für dies werde ich das bekannte Spiel «Faltpapier-Geschichte» digitalisieren. Dieses Spiel funktioniert wie folgt:

1. Spieler 1 schreibt zwei Sätze
2. Spieler 2 sieht den zweiten Satz des Spieler 1
3. Aufgrund dessen schreibt er ebenfalls zwei Sätze
4. Spieler 3 sieht nur den zweiten Satz von Spieler 2
5. Aufgrund dessen schreibt er ebenfalls zwei Sätze

Und so weiter.

Das Erstellen der zwei Sätze wird Zeitbegrenzt sein, sodass man nicht zu lange studiert und die Geschichte spontaner wird. Zudem kann jeder User nur 1-mal alle 2h die Geschichte um zwei Sätze erweitern. So schreibt der User nicht eine Geschichte allein und es schützt vor Spam-Angriffen. Nach dem Absenden der erstellten Sätze, gibt es die Möglichkeit den aktuellen Stand Geschichte herunterzuladen.

Es gibt folgende Rollen und ihre Berechtigungen.

- **Writer**
 - Standardrolle, wenn man sich neu registriert.
 - Einloggen und ausloggen
 - Schreiben von 2 Sätzen
 - Herunterladen der Geschichte
- **Game Master**
 - Wird in der DB manuell von einem Writer zu einem Game Master migriert
 - Einloggen und ausloggen
 - Sieht aktueller Stand der Geschichte
 - Sieht welcher User was Geschrieben hat

Alle User müssen sich beim Start der Applikation über ein Login Fenster authentifizieren. Die Authentifizierung wird mittels Mailadresse und Passwort durchgeführt. Es gibt keine weiteren Möglichkeiten der Authentifizierung. Falls der User noch kein Login besitzt, kann er sich über den Registrierungs-Button als neuen User anmelden. Es ist nicht möglich, als einen Gast mitzuschreiben.

Bei den Userdaten werden nur die Mailadresse sowie das Passwort in der Datenbank gespeichert.

Es werden für die gesamte Projektdokumentation die Standards des Betriebs bzw. der Abteilung verwendet. So wie die Standards Programmiersprachen der Abteilung (Perl und Java). Alle Daten werden in einer zentralen relationalen MySQL-Datenbank abgespeichert.

Folgendes wird in diesem Projekt nicht berücksichtigt.

- Keine Profiländerungen des Benutzers
- Änderungen am Datenbestand durch einen anderen Client wird nicht automatisiert im eigenen Programm aktualisiert
- Die Applikation muss nur auf dem Gerät, welches während der Arbeit benutzt wird, laufen.
- Andere Betriebssysteme oder Geräte werden nicht berücksichtigt.
- Die Versionierung wird nur mittels Gitlab gestattet.

4.2 Projektaufbauorganisation

Mehr infos zu personen

Auftraggeber: Remo Steinmann

Auftragnehmer: Lara Felix

Experte: Remo Steinmann

Nebenexperte: Ruwen Wiederkehr und Merjem Hamza

Versionierung?

4.3 Mittel und Methoden

Folgende Mittel werden in diesem Projekt gebraucht:

- Programmiersprache: Java, Perl, SQL
- Datenbank: MySQL
- Betriebssystem: Windows 10
- Versionierungssystem: GitHub Repository und Git
- Diagramme (UML's): diagrams.net
- DIE: Visual Studio Code

Als Projektmanagementmethode wird IPERKA benutzt.

4.4 Vorkenntnisse

Ich arbeite in meiner Abteilung hauptsächlich mit den Programmiersprachen Java und Perl. Das verbinden von Frontend (Java) und Backend (Perl) ist mir ebenfalls durch die Arbeit in meiner Abteilung bekannt. Jedoch stehen dort diverse Template für das Frontend mittels Java zur Verfügung. Alles vom Grunde auf neu aufzubauen wird für mich Neuland sein. Das Einbinden einer Datenbank ist mir ebenfalls schon bekannt, dies jedoch auch wieder nur mittels Librarys.

Eine Multiuserapplikation an sich habe ich noch nie erstellt.

Die oben aufgeführten Mittel habe ich alle schon mehrmals eingesetzt.

4.5 Vorarbeit

Für das Erstellen des Projektantrags, machte ich mir schon ein paar Gedanken bezüglich des groben Aufbaus. Dies wird im Schritt Planen erwähnt.

4.6 Deklaration der benützten Firmenstandards

Es wird die aktuelle Version der Firmenstandards Siemens Mobility benutzt.

4.7 Zeitplan

Fertiger Einfügen

4.8 Arbeitsprotokoll

29.11.2021 – Tag 1

Geplant

- Projektdokumentation Aufbau erstellen
- Gantt - Planung erstellen
- Projektjournal Tag 1
- Vorkonzept / Projektbeschreibung
- Gespräch mit HE (Remo Steinmann)

Gemacht

- Gantt – Planung
- Projektdokumentation Aufbau
- Projektjournal Tag 1
- Einlesen in FArbeit_2021 und Kriterienkatalog
- Aufgabenstellung
- Projektaufbauorganisation
- Erstes Gespräch mit Hauptexperten (Remo Steinmann)

Probleme

Bei der eigentlichen Umsetzung von den oben genannten Punkten traten keine Probleme auf. Jedoch kam ich nicht so gut voran wie gedacht.

Fazit

Folgendes wurde im Gespräch besprochen

- Gantt Planung
 - Schritte mehr aufteilen (z.B. Login planen, Login erstellen, Login testen)
 - Legende im Excel – File und nicht nur in Textform in der Dokumentation
 - Meilensteine Soll/Ist Vergleich zum Beispiel Meilenstein als eigenen Task
 - Jeden Iperka – Schritt aufteilen und nicht zusammenfassen
 - Schriftgrösse von Tasks grösser → Soll/Ist – Boxen im Vergleich zu gross
- Aufgabenstellung
 - Gut, kommt von abgegebener Aufgabenstellung umgeändert in die Dokumentation
- Projektdokumentation Aufbau
 - Glossar, Quellenverzeichnis, Abbildungsverzeichnis und Tabellenverzeichnis kommen in den Punkt Anhang und wird als Punkt 6 gekennzeichnet

Die oben genannten Verbesserungspunkte werden bis morgen Abend 30.11.2021 17.00 Uhr angepasst.

30.11.2021 – Tag 2Geplant

- UML's Mockup
- Testkonzept
- Projektjournal Tag 2
- Varianten evaluieren
- Dokumentation nachführen
 - Teil 1
 - Informieren und Planen Teil 2

Gemacht

- Teil 1 nachgeführt
 - Mittel und Methoden
 - Vorkenntnisse
 - Vorarbeit
- Wireframes
- Ddl
- UseCase (muss noch überarbeitet werden)
- Projektjournal Tag 2

Probleme

Ich konnte mich, wie gestern, nicht so gut konzentrieren. Da wir uns in dieser Konstellation schon lange nicht mehr gesehen haben. Deshalb wurde viel geredet anstatt gearbeitet. Ich hoffe, dass ich dies morgen im Home Office aufholen kann. Falls dies nächste Woche immer noch der Fall sein sollte, werde ich mehr Musik hören, um konzentrierter zu arbeiten und nicht abgelenkt zu werden. Bei dem Use Case war ich mir wegen den Notationen nicht mehr sicher. Dies klärte ich jedoch mit Remo.

Gelerntes

Input Remo

- Check-Liste für Gespräch mit HEX
 - Aufgeräumt
 - Infrastruktur geprüft (vor ort bzw. zu Hause)
 - HEX als Besucher anmelden
 - Gantt - Planung parat
 - Geht ca. 20 min.
 - Kriterienkatalog ausdrucken
 - Tipps umsetzen
- Use Case
 - Überblick über das Programm geben
 - Muss noch in Text erläutert werden
 - Akteur, Voraussetzung, Kurz Beschreib

Es ist wichtig, dass ich fokussierter arbeite. Dies ist mein Hauptproblem. Ich konnte viel aus diesem Tag mitnehmen insbesondere die Inputs von Remo haben mir auf Hinsicht dieses Projektes sowie auf die IPA sehr geholfen.

Konnte nicht umgesetzt werden und muss morgen nachgeholt werden

- Use Case überarbeiten mit Inputs von Remo
- Testkonzept (Können von Use Case abgeleitet werden)
- Evtl. weitere UML's einfügen bzw. bestehende überarbeiten.

5 Teil 2

5.1 Kurzzusammenfassung

Ausgangslage

Umsetzung

Ergebnis

5.2 Informieren

Folgende Rahmenbedingungen wurde durch den Modulleiter Remo Steinmann vorgegeben

- Objektorientierte Multi-User-Applikation.
- Umfang gemäss Planung (ca. 3 Tage Entwicklung, 2.5 Tage Dokumentation; 6h pro Tag)
- Zentrale Datenbank
- Mehrere Clients müssen gleichzeitig auf den gleichen Datenbestand zugreifen
- Zentrale Benutzer- und Rechte-Verwaltung

Ziele der Aufgabenstellung 19

Vorgaben 19

Fragen 19

5.3 Planen

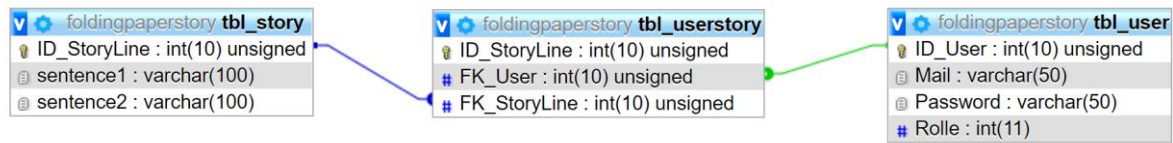
5.3.1 Zeitplan

		Montag - 29.11.2021		Dienstag - 30.11.2021		Mittwoch - 01.12.2021		Montag - 06.12.2021		Dienstag - 07.12.2021		Mittwoch - 08.12.2021			
		2h	2h	2h	2h	2h	2h	2h	2h	2h	2h	2h	2h		
Informieren/Planen/Entscheiden	Vorkonzept/Projektbeschreibung	soil												Meilensteine	
	ist														
	Projektokumentation aufbau	soil													
	ist														
	Storyboard/Mockup	soil													
	ist														
Realisieren	Planung	soil												Informieren/Planen und Entscheiden abgeschlossen	
	ist														
	Testkonzept	soil													
	ist														
	Grundstruktur Datenbank erstellt	soil													
	ist														
Realisieren	Login fertig	soil												Grundaufbau der Applikation fertig (Datenbank und Layout)	
	ist														
	Layout (Grundstruktur) Hauptseite fertig	soil													
	ist														
	Funktionalität "Geschichte erweitern"	soil													
	ist														
Realisieren	Funktionalität "Geschichte Export"	soil												Grundfunktionen der Applikation fertig (Geschichte erweitern und Login)	
	ist														
	Funktionalität "Zeit Counter"	soil													
	ist														
	Testing	soil													
	ist														
Reflexion	Reflexion	soil												Dokumentation und Applikation abgeschlossen	
	ist														
	Dokumentation nachführen	soil													
	ist														
	Journal	soil													
	ist														
Algorithmen	Experten Gespräch	soil													
	ist														

5.3.2 UMLs

Folgendes wurde als Vorarbeit für die Definition der Aufgabenstellung

5.3.2.1 Datenbank



5.3.2.2 Wireframe

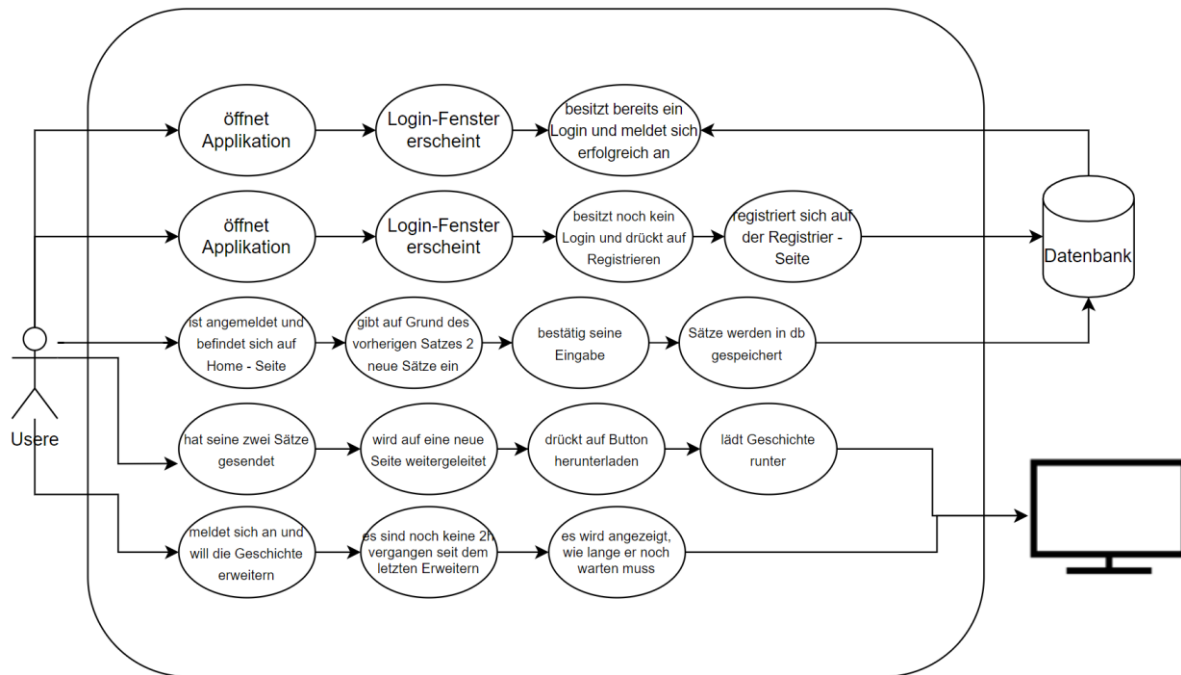
Die unten dargestellten Wireframes sind nach ihrer chronologischen Reihenfolge angeordnet und nummeriert. Das Design wird schlicht gehalten.

<p>1</p> <p>Registrieren</p> <h2>Login</h2> <p>E - Mail Adresse <input type="text"/></p> <p>Passwort <input type="password"/></p> <p>Anmelden</p>	<p>2</p> <p>abmelden</p> <p>Vorheriger Satz</p> <h2>Schreibe weiter....</h2> <p>Satz 1 <input type="text"/></p> <p>Satz 2 <input type="text"/></p> <p>Zeit Counter</p> <p>senden</p>
<p>3</p> <p>abmelden</p> <h2>Danke für deine Ergänzung :)</h2> <p>Geschichte downloaden</p> <p>weiter</p>	<p>4</p> <h2>Schreibe in 24 h weiter</h2> <p>abmelden</p>

Falls man sich anmeldet, obwohl die 2h 24h noch nicht vorbei sind, kommt folgendes Fenster, welches einem die verbleibende Wartezeit anzeigt. X steht in der Darstellung unten für die verbleibende Zeit. Das Registrierungsfenster ist gleich aufgebaut wie das Login Fenster.

<h2>Schreibe in x weiter</h2> <p>abmelden</p>	<p>Login</p> <h2>Registrierung</h2> <p>E - Mail Adresse <input type="text"/></p> <p>Passwort <input type="password"/></p> <p>Registrieren</p>
---	---

5.3.2.3 UseCase



UC1 --> Login	
Bedingung	Applikation wurde betriebsbereit auf dem System installiert.
Beschreibung	<p>User startet App. App öffnet sich und es ist die Home-Seite zu sehen. Es werden die hinzugefügten Lebensmittel angezeigt. Pro Lebensmittel werden auf den ersten Blick folgenden Informationen gezeigt</p> <ul style="list-style-type: none"> • Name des Lebensmittels • Vorhandene Menge zu Hause • Mittels Dreiecks in grün, orange/gelb oder rot wird angezeigt, ob das Lebensmittel noch lange haltbar ist • Foto (falls vorhanden, ansonsten ist dieses Feld leer)
Nachbedingung	keine

5.3.3 Testing

5.4 Entscheiden

5.4.1 Projekt / Technologie

Zielgruppe

Zielgruppe unserer App sind Menschen jedes Alters und Geschlechts. Voraussetzung ist das sie Zugang zum Internet haben und ein Mobiltelefon besitzen. Zudem hat unsere Zielgruppe Freude an

Filmen. Nicht zu unserer Zielgruppe sind Menschen mit Beeinträchtigungen, da wir keine Zeit haben diese zusätzlichen Features einzubauen.

5.4.2 Design

Unsere APP berücksichtigt wegen mangelnder Zeit Barrierefreiheit nicht.

Navigation

Die Icons in der Navigation sind so gewählt, dass der User genau weiss, wo was ist und er sich schnell zurechtfindet.

Die Icons werden unabhängig der Bildschirmgröße horizontal gleichmässig positioniert.

Zudem wird der aktuelle Tab durch dunklere Farbe gekennzeichnet.

Filme

Die Filme werden in jedem Tab in Kacheln angezeigt. Diese werden abhängig der Bildschirmgröße zentriert.

Es wird der Titel und eine kurze Beschreibung des Filmes angezeigt.

Durch ein Dropdown Menü, kann man den Status auswählen (Plan to watch, watched) und die Bewertung abgeben.

Suchleiste/ Titel / Untertitel

Die Suchleiste, der Titel sowie der Untertitel sind auf jedem Tab vorhanden und werden ebenfalls zentriert und passen sich so der Bildschirmgröße an.

Farb-/Schriftlegende

Schrift:

- Nunito (Google Fonts)

Farbe:

- #2F2B36
- #3C3746
- #6B6478
- #7840FF
- #C5ACFF
- #ECE4FF
- #7840FF14

Namenskonvention

- Variablen und Funktionen werden klein geschrieben (Camel Case)
- Files werden klein geschrieben

5.5 Realisieren

Das Projekt inkl. Dokumentation ist unter dem folgenden Link erreichbar: [GIT - REPO](#)

Eine Installationsanleitung ist im Readme.md ersichtlich.

5.5.1 Persistenz

Wir speichern alle **Sachen** in eine JSON File. So sind die Änderungen bei einem unerwarteten Enden der App nicht verloren.

5.5.2 Netzwerkfunktionalität

Um auf die Daten des APIs zuzugreifen, wird eine Internetverbindung gebraucht. Falls diese nicht gegeben ist, kommt eine Meldung, dass man nicht mit dem Internet verbunden ist.

Eine Verbindung zum Internet ist für unsere App zwingen notwendig.

5.6 Kontrollieren

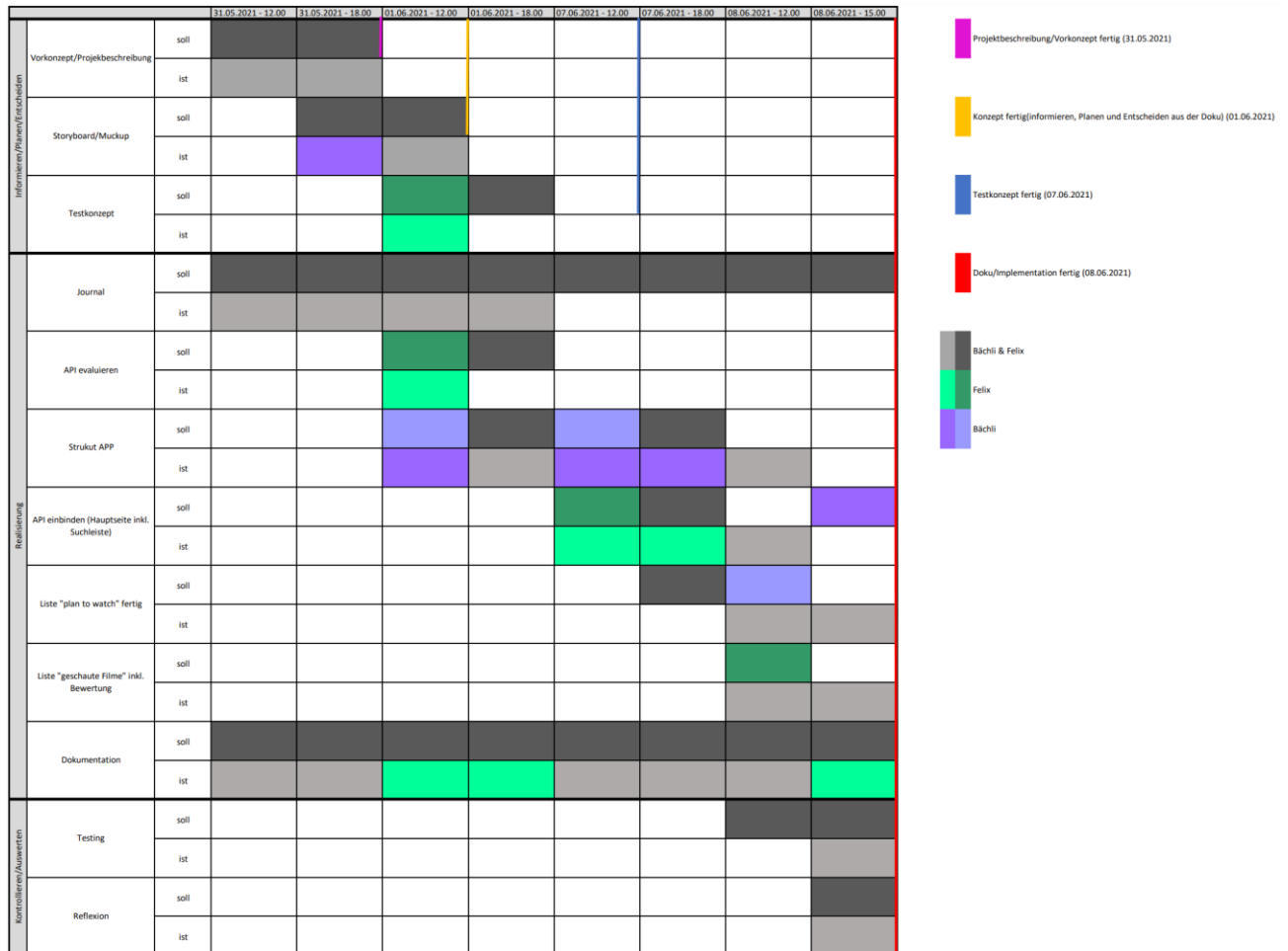
Da wir einiges nicht wie geplant umsetzen konnten. Konnten wir gewisse Punkte im Testing nicht erfolgreich erfüllen.

5.6.1 Testing

5.7 Auswerten

5.7.1 Zeitplan (Soll/Ist Vergleich)

Wir waren bis zum letzten Tag immer sehr gut in der Zeitplanung. Durch die Probleme am letzten Tag wurden wir sehr zurück geworfen und konnten so nicht alles erreichen wie gewünscht.



5.7.2 Reflexion

Wir sind sehr zufrieden mit unserem Projekt. Wenn wir unserem Projekt eine Note geben würde, werde es mindestens eine 5. Obwohl wir nicht alles so erreicht haben wie geplant haben wir viel Zeit und Mühe in dieses Projekt gegeben und für 2 Tage effektives Programmieren inklusive Doku sind wir zufrieden mit unserem Projekt. Wir waren immer sehr gut im Zeitplan und kamen gut und ohne grosse Probleme voran. Doch am letzten Tag hat uns das Kombinieren von Struktur und API einen Strich durch Rechnung gemacht und das Rendern der Filme uns mehr Zeit gekostet als gedacht. Wir vermuten, dass uns das API durch zu viele Anfragen immer wieder für eine kurze Zeit gesperrt hat. Jedoch steht auf der API Seite, dass das Rate limit disabled ist.

So konnten wir nicht alles erreichen wie wir wollten. Trotzdem versuchten wir das beste daraus zumachen. Durch die gewohnte zusammen Arbeit konnten wir eine gewisse Sicherheit und Kontrolle in den abschliessenden Stress bringen.

Wir können viel von diesem ÜK mitnehmen. Wir konnten unser Wissen bezüglich Projektarbeit und Aufbau einer Doku auffrischen und festigen. Da wir in der TBZ auch oft solche Projekte haben, war es für uns nicht so schwer. Wir können jedoch viel für die IPA mitnehmen und konnten in diesem ÜK nochmals genauer auf für die IPA wichtigen Punkte achten.

5.8 Glossar

5.9 Quellenverzeichnis

<https://reactnative.dev/>

[https://www.themoviedb.org/\\$](https://www.themoviedb.org/$)

5.10 Abbildungsverzeichnis

5.11 Tabellenverzeichnis

5.12 Anhang

Sind auf dem Git-Repo ersichtlich.

Testing

Gantt-Diagram