

Planungsphase

Probleme:

11.5.2021

-Zeitaufwand

-App Priorisierung

-Aufgabenverteilung

-Maxi & Kayra müssen andere Aufgaben annehmen (z.B. Design)

-Datenschutz (Art des Logins, Speicherung der Daten, Server Standort etc.)

-Stand Alone System (Ohne Microsoft's Active Directory/API oder andere Third-Party)

Anforderungen:

-API

-App/Programm/Website

-Server

-Vielleicht Database

-GUIs (z.B. NetBeans)

Aufgaben der Software:

-Liveverfolgung der Stunden- & Vertretungspläne

-Anzeigen des aktuellen Mensaplan für die aktuelle & nächste Woche

-Spezielles Interface für das Sekretariat

-Verschiedene Funktionen für verschiedene Rollen (Schüler, Lehrer etc.)

Testen erfolgt während jedem Schritt

Grundsätzliche Aufgabenverteilung:

13.5.2021

-Leo: Server, Server API (Mithilfe), Website, App

-Simon: Server API, Server, Programm (Mithilfe)

-Visar: Server API (Mithilfe), App (Mithilfe), Programm, Website (Mithilfe)

-Kayra: Dokumentation, Tests & Feedback, GUI Design

-Maxi: GUI Design, Tests & Feedback, Backup für Dokumentation

Implementier Reihenfolge

16.5.2021

Bei jeder Implementierung (Siehe Reihenfolge der Implementierungen: INF1 11.05.2021):

Zunächst bilden wir die konkrete Softwarestruktur (Klassen, Interfaces usw.) Dann implementieren wir die Softwarestruktur und testen sie letztendlich. Falls es Verbesserungsvorschläge gibt, fangen wir mit dem Prozess wieder an, bis wir mit dem Ergebnis zufrieden sind.

1. Server & API
2. App & Design/GUI
3. Lehrerprogramm
4. Website App
5. Programm

11.5.2021

Probleme aufgetreten während der Implementierung

- IOS App nicht möglich
 - o Zeitmangel
 - o Implementierung nur über Apple Gerät möglich
 - o Wird höchst wahrscheinlich nicht im App Store verfügbar zu machen sein

18.5.2021

- Microsoft Azure Active Directory funktioniert nicht

15.6.2021

- Distribution
 - o Play Store verlangt 25 \$ für eine Developer-Lizenz um die App online zu stellen
 - o App Store verlangt \$100 pro Jahr für eine Developer-Lizenz um die App online zu stellen

8.7.2021

Implementierung Fortschritt

Anmeldung

Dadurch dass die Implementierung der Anmeldung über Azure nicht klappt, benutzen wir stattdessen einen E-Mail Bot, um die angemeldeten Personen zu verifizieren
Funktionalität:

15.6.2021

-19.6.2021

1. Zunächst melden sich die Benutzer mit ihrer Schulemail an

2. Dann erhalten sie von unserem Bot eine Nachricht, welche einen 6-stelligen Code beinhaltet
3. Dieser Code muss in der App innerhalb von 15 Minuten angegeben werden, um sich in der App zu verifizieren

Dazu benötigt:

- No Reply E-Mail Adresse : noreply.maristenkolleg@gmail.com
- Implementierung mit Mail.java (generiert automatische E-Mail):

Hallo Visar Lumi,

Dein Code lautet 201022.

Mit freundlichen Grüßen,
Ihr Maristenplaner-Team

- Umwandlung Mail.java zu Kotlin
 - o Bessere Performance
 - o Bessere Codeintegration

6.7.2021

Android App Entwicklung

Implementierung über:

26.6.2021
-XX.7.2021

Flutter (Framework) und Dart (Programmiersprache)

Funktionen:

- Vertretungsplan ansehen
- Stundenplan ansehen
- Mensa Menüplan

Schritte:

- Stundenplan API Request
 - o Die App stellt eine Request an den Server, der der App dann den Stundenplan übermittelt
- Navigation Bar
 - o Design und Funktion der Navigation Bar hinzugefügt
- About „App“ Bereich mit Lizenzen
 - o Beinhaltet alle Lizenzen
- Server Fix wegen API request
 - o Bugfix
- Dark Mode
 - o Design; Kompatibilität zum Dark Mode
- Stundenplan schöner machen
 - o Design fix
- Einbau erster Settings
 - o Hinzufügen eines Tabs „Settings“ mit noch nicht vielen Funktionen
- Login und Authentication Screen
 - o Implementierung des Login- und Authenticationscreens
 - o System siehe „Anmeldung“

26.6.2021

29.6.2021

30.6.2021

2.7.2021

4.7.2021

- Refresh Funktion
 - o Neu laden des, bis jetzt, Stundenplans
- Sharred preferences
 - o Hinzufügen, dass man sich nicht bei jedem öffnen der App neu einloggen muss (App resettet die Nutzerdaten nicht)

6.7.2021

Sekretariatsapp

Funktionen:

- Vertretungen eintragen
- Lehrerstundenpläne einsehen/ändern
- Lehrerliste einsehen/ändern
- Klassenlisten einsehen/ändern
- Raumplan einsehen/ändern

Implementierung über:

IntelliJ in Java

Schritte:

- Erstellung eines Designs
 - o Über Figma.com
- To be continued

29.6.2021