**Beschreibung der Arbeitspakete**

Google API

Google bietet eine Schnittstelle zur Darstellung und Erstellen von Kalender-Ereignisse. Zusätzlich kann man mit andere Kalendern synchronisieren und Ereignisse bearbeiten.  
Code zur Implementierung der Schnittstelle wird entwickelt und die Methoden der Schnittstelle getestet. Eine funktionierende Implementierung der Google API, bietet das Basis für die Weiterentwicklung der Applikation. Zusätzlich wird möglich sein, Termine mit CIS-Kalender zu synchronisieren.

Hier werden die Methoden: request(), synchronise(), edit() und syncCis() implementiert.

Kalender

Grundfunktionalität der Applikation wird entwickelt. Die Klasse Kalender enthält die Methoden: dynamischeVerteilung(), filter() und suche().

dynamischeVerteilung() - siehe Ziele (dynamische Terminverteilung).

filter() – Filteroptionen werden integriert, z.B. will man nur Ereignisse, die mit der FH zu tun haben, darstellen lassen.

suche() – Suche nach einem bestimmten Ereignis.

Eintrag

Code zur Einträge (Ereignisse) hinzufügen, bearbeiten bzw. löschen wird entwickelt. Hierbei wird zwischen SingleDate und MultiDate Ereignisse. Hier folgt die Implementierung der Klassen Eintrag, SingleDate und MultiDate implementiert.

SingleDate: Sind Ereignisse die das gleiche Anfangs- und Enddatum haben. Hier werden die Methoden showEvent(), createEvent(), editEvent().

MultiDate: Sind Ereignisse das verschiedene Anfangs- und Enddatum haben. Technisch gesehen ist ein Multidate Eintrag ein Array von SingleDate Klasse.

File Manager

Serialisierung von Objekten implementieren. Dafür wir eine externe Bibliothek gewählt. Kalender-Ereignisse werden in eine lokale Datei gespeichert und bei Starten der App wieder von der Datei abgerufen und in App zur Verfügung gestellt.

Hier werden die Methoden: save(), load() und synchronize() implementiert.

Userpräferenzen

Die Klasse Userpräferenzen ermöglicht die Änderung von Benutzereinstellungen. Für die dynamische Terminverteilung werden die Lernzeiten bzw. Lerntyp bestimmt, z.B. ich lerne von 08:00 bis 16:00 jeden Tag, oder ich habe in progressiver Lerntyp und lerne viel mehr eine Woche vor der Prüfung usw.

Graphische User Interface

GUI für die App entwickeln. Als Basis dient das Layout von Google Kalender API. Es wird eine Navigationsleiste implementiert, wo die verschiedenen Buttons zur Funktionalität der App zu finden sind.

Projektmanagement

Dokumente die zur Projektmanagement zählen vorbereiten. Projektbeschreibung, Projektstrukturplan, Projektablaufplan, Statusberichte und Stundenaufzeichnungen entwickeln. Der Projektablauf wird dokumentiert und am Ende, basiert auf die Dokumentation, wir ein Entwicklungshandbuch / Benutzerhandbuch zur Verfügung gestellt.

Java einarbeiten

Java als Programmiersprache erlernen. Tutorials für Einsteiger sehen bzw. lesen und dann spezifische Recherche machen, je nach dem was für unsere Applikation notwendig ist, z.B. Serialisierung, Implementierung von Schnittstellen usw.

GIT einarbeiten

Das Konzept hinten GIT erlernen. GIT installieren, GIT Repository erstellen und GIT GUI erlernen.

Logo Design

Ideen für das Logo sammeln. Logo zuerst entwickeln und dann mithilfe von Design Apps das Logo digitalisieren.

Debuggen

Zeitpuffer vorher planen um Fehler zu finden und Fehler zu beheben.