**Preface:**

This is version 1.0.0 of Pupil. This is our initial launch version of the Pupil App.

**Introduction:**

The need for this system is for a stand alone student assistant. The system will be a stand alone application for iPhones that will have a calendar that will store assignments and dates, a grade calculator, a syllabi storing system, and a study timer. This application is the primary goal for iTeam.

**Glossary:**

**iOS:** Apple's Operating System that is used on their Mobile Devices

**UI**: User Interface

**Swift**: Apple's Proprietary Programming Languages

**User Requirements Definition:**

**User Requirements:**

1. The Pupil App shall have a calendar that will store assignments and due dates
2. The Pupil App shall have a grade calculator that will allow users to calculate their grade for each course.
3. The Pupil App shall have a way to store syllabi for courses as image files.
4. The Pupil App shall have a study timer feature that will help students zone in on their studies.

**Goals:**

* The user should be able to quickly add assignments with all necessary details.
* The Pupil App should have easily accessible features with smooth transitions.
* The Pupil App should clearly list the assignments for ease of use.
* The Pupil app should have a minimal UI that matches Apple’s design language.

**Non-Functional Requirements:**

* Pupil shall adhere to iOS user requirements.

**System Architecture:**

The Pupil application is a student assistant with four main features: A calendar to store assignments and due dates, a way to store syllabi for each class as images, a grade-point average calculator, and a study timer. The users will primarily be college students.

There will be 5 main interfaces:

1. The calendar interface
   1. This interface will be the largest component of the application. This is where all of the user’s assignments will be stored and visible either as a daily assignment calendar view, a weekly assignment calendar view, or a weekly calendar view with the student’s classes and assignments.
2. The class interface
   1. This interface will control and store all aspects of a student’s academic class. Information such as class name, dates, professor, syllabus, assignments, and grades will be visible from here. The GPA interface and the photo-taking interface will be accessed from here.
3. The GPA interface
   1. This interface will be accessible from each class’ view. Here the student’s current grade in the class will be visible, calculated from the class’ grades information.
4. The study timer interface
   1. This interface will contain the application’s study timer feature.
5. The photo-taking interface
   1. This interface will be used by the class interface as a way of taking and storing photographs of relevant material pertaining to each class, such as a photo of the class’ syllabus.

**System Requirements Specification:**

**System Feature Requirements**

**1. Calendar that will store assignments and due dates**

1.1. Calendar shall be always accessible from the top bar in the home screen.

1.2. Assignment added shall need to have a name, a course, due date, and an optional importance flag.

1.3. Main Screen should have access to calendar and display due dates on assignments from that information.

1.4. The Calendar should show a day view, and a week view of the assignments.

**2.Way to store syllabus for each course as image**

2.1. Syllabi shall be stored as image files they should be in the camera roll to be referenced by The Pupil App.

2.2. Syllabi shall be associated with a course and should be accessible from that course page.

2.3.There should be an in app photo prompt to take a picture of the Syllabi, stored in Camera Roll and accessible by Pupil.

2.4. There should be an editable form where important information from the Syllabi can be entered by the user.

**3.Grade in course feature**

3.1. Pupil shall have the option to add a grade to a completed assignment or change a grade after it is submitted if needed

3.2. Pupil should use grading weights to find the grade for each course

3.3. Pupil should show GPA earned with the current grades for the semester

3.4. Pupil should show your current grade in a course based off of graded assignments

**4.Study timer feature**

4.1. The study timer shall display a running clock displaying user study duration, it can be started and stopped.

4.2. The study timer shall implement a Pomodoro study timer using a 25 on, 5 off system

4.3. The study timer shall visibly or audibly alert the user at the appropriate intervals

4.4. The study timer should keep track of the user’s time spent studying and keep a history of their study sessions

4.5. The study timer should sync with the assignments so the user can mark a current assignment completed

**System Models:**

We are using an object model that will describe the relationship between the different objects in the system from the Calendar Object, to the Courses and Assignments objects. These objects exchange data between each other, and may be represented using a data-flow model showcasing the flow of data between the different objects in the system.

**System Evolution:**

This system is built upon the assumption that it will be usable through the end of life of Swift 3.0 at a minimum. In this life span it is anticipated that new updates of Swift and new hardware updates of iPhones will be introduced. No major changes should need to be made to accommodate these changes according to Apple. Further functionality maybe added with respect to developer interests and customer response.

**Appendices:**

Our hardware will be iPhone's supporting applications built in Swift 3.0, all re-generatable data will be stored within the application, whilst user-generated data should be stored on the device with possible backing up to iCloud. The application will referenced local data such as syllabi stored in the camera roll.

**Index:**

**Preface:** Page 1

**Introduction**: Page 1

**Glossary**: Page 1

**User Requirements Definition:** Page 1

**System Architecture:**  Page 1

**System Requirements Specification:** Page 2

**System Models:** Page 3

**System Evolution:** Page 3

**Appendices:** Page 3