Sprint #2 Plan

Product Name: E-tendance, **Team Name**: Midnight Cobra
Sprint Completion Date: 2/21/18 , **Revision Number**: 1.0.0, **Revision Date**: 2/4/18

- **Goal:** Set up the app and database so that a professor/student is able to create an account and their account information is saved in the database. User should be able to sign in to their account once it has been created, and functionality should be different for professors and students.

- **Task listing, organized by user story:**

  User story 1: As a professor I want to be able to create and save classes where students are able to join. **Story point: 8**

  Task 1.1: **Create Class UI (Professor)** (1 hour): Create "Create Class" page UI. Assignee: Joey

  Task 1.2: **My Class UI (Professor)** (1 hour): Create a class page for a particular class after you "Create" it.

  Task 1.3: **Save Class in Firebase** (5 hours): When class is created by a professor, the class and corresponding info are saved into a child node ("group") called "classes." Assignee: Katy

  Task 1.4: **Create Class Button** (2 hours): Professor-side of the app has a "Create Class" button. Professor fills out fields such as class code and pin, and these are saved with the class in the database. Assignee: Katy/Anisha

  Task 1.5: **Class Code Generation** (1 hour): Require a passcode/pin for students to join a class successfully. The passcode/pin will be inputted by professor and saved in the Class child node. Assignee: Katy/Anisha

  Task 1.6: **Navigating Firebase Database** (10 hours): Be able to navigate and retrieve data from a specific child inside the database explicitly. Assignee: Matt

  Total for user story 1: 20 hours

  User story 2: As a student, I want to be able to join the classes my professor created.
  Story point: 8
  Task 2.1: **My Classes UI (Student)** (1 hour): Create "My Classes" page UI. Assignee: Joey

Task 2.2: **Add Class by Code/Pin** (8 hours): Student should enter a class code and pin number to join a class. It should be visible in the database in the class' child that the student is in that class. Assignee: Anisha

Task 2.3: **List View of Classes** (3 hours): Populate a list view of the current user's classes. Assignee: Joey

Task 2.4: **Navigating to Classes** (10 hours): When a student selects a class from their classes, look into class' child node in the database for corresponding class and be able to navigate toward a new page populated with class information. Assignee: Darren

Total for user story 2: 22 hours

User story 3: As a professor, I want users to be able to check in to class, and see who checked in. I want to ensure only students that are physically in class can check in. **Story point: 13**

Task 3.1: **Check-in Button** (2 hours): Implement check-in button for students. When a student selects a class from "My Classes," there is a check-in button that they can select. This will set an isPresent boolean from false to true in the database child for that user in that class for that day.

Task 3.2: **Student Check-in** (8 hours): When a student checks-in, there should be a child node for that certain day under the parent class where all enrolled students would have the field "isPresent" set to false, and checking in will set such field to true.

Task 3.3: **Learn Location Services API** (6 hours): Study location services API and figure out useful implementation methods. Assignee: Everyone

Task 3.4: **Geolocation Check** (8 hours): Check the user's location while the app is open.

Task 3.5: **Implement GeoCheck with Attendance button** (11 hours): Figure out how to restrict location parameters so that attendance status is only updated when within the specified parameters.

Total for user story 3: 35 hours

- **Team roles:**

  Matthew Dunn: Product Owner, Q/A, Developer
  Katelyn Suhr: Developer
  Joseph Aronson: UX Designer, Developer
  Anisha Tavva: Scrum Master, Developer
  Darren He: Developer

- **Initial task assignment:** A listing of each team member, with their first user story and task assignment. This should look like:

  Darren He: 1.6, 2.4
  Matthew Dunn: 1.6, 2.4
  Katelyn Suhr: 1.3, 1.4, 1.5
  Joseph Aronson: 1.2, 2.1
  Anisha Tavva: 1.3, 1.4, 1.5

- **Initial burnup chart:** A graph giving the initial burnup chart for this sprint and is labeled as such with sprint number and project name and is located in the lab.

- **Initial scrum board:** Also known as a task board, the scrum board is a physical board and labeled as such with sprint number and project name and located in the lab. This board has four columns, user stories, tasks not started, tasks in progress, and tasks completed. Index cards or post-it notes representing the user stories and the tasks for this sprint should be placed in the user stories, tasks not started, and tasks in progress columns. Tasks associated with a user story should be placed in the same row as the user story.

- **Scrum times:** List at least the three days and times during the week when your team will meet and conduct Scrum meetings. Also, indicate which of these meetings will have the TA visit as arranged with the TA. It is expected the TA will visit during the Scrum meeting during your lab time.

1. Tuesday @ 3:30 PM
2. **TA Meeting: Thursday @ 11:00 AM**
3. Sunday @ 3:00 PM