# *papaya*

Team 6 – Sprint 1 Planning Document

*Ben Maxfield, Christian Lock, Adam Johnston, Caleb Flynn, Scott Hanberg*

Scrum Master: Ben Maxfield
Scrum Meetings: Tuesdays @ 7pm - 8pm

## Sprint Overview:

**Overview:**

During this spring we hope to implement all the main functions of our app so we can get our platform up and running. User features and quality of the app will be taken care of during later sprints. The frontend team will be setting up authentication as well as the main UI for our core features (Scrollable map, Study session icons, Study session filters, etc..). The backend team will focus on setting up processes to implement features using AWS services while implementing the first couple core features. These services include AWS API Gateway, AWS Lambda, and AWS RDS, which cover the entire range of backend services we will have to work with for the rest of the project.

**Risks and Challenges:**

We have setup our sprint to accomplish a large amount of work so it will be challenging if we fall behind. This is emphasized during our first sprint because more research will need to be done since we are still relatively unfamiliar with the API calls as well as Android development. Another problem is possible merge conflicts that may come up during development. The front end team will also need to ensure that the UI will be fitted for all Android devices and resolutions.

## Sprint Detail:

**User Stories:**

**User Story #1:**

*As a Student/TA/Professor I would like to login to the app:*

| Description | Time | Person |
|---|---|---|
| Facebook and Google Button integration | 10 | Christian |
| Package necessary profile data into JSON for API call | 2 | Adam |
| Make API call and await response, perform logic depending on success/failure | 5 | Adam |
| Store successful pass keys in Keystore | 5 | Adam |
| Loading Screen UI | 5 | Christian |
| Android Login screen UI | 5 | Christian |
| Setup API gateway for user authentication | 1 | Ben |
| Create and tie Lambda function for authentication | 10 | Ben |
| Create RDS table for user data, SQL queries | 5 | Adam |

**Acceptance Criteria:**

Overall, the authentication system should be complete with robust testing, by the end of the sprint.

- Users are able to log in using either their Google login or their Facebook login.
- Given the user logged in successfully within 60 days, when users open the app, they should not be prompted to log in again and should be redirected to the main action page, unless their token was invalidated by the authorizer (password change, token timeout).
- Whenever an API call is made, we should have a segment of Lambda code in our backend that will validate the given key with our saved database key.
- When a user opens the app, it should personally validate the stored access key with its authorizing service and notify our backend if a new access key has been issued.

**User Story #2:**

*As a Student/TA/Professor I would like to be able to filter what study sessions appear on my map based on class.*

| Description | Time | Person |
|---|---|---|
| Create Join a class interface and model | 6 | Caleb |
| Create API Gateway endpoint to join class | 2 | Caleb |
| Create Lambda method to join class | 7 | Caleb |
| Create RDS table for user to class assignments, used SQL queries | 5 | Adam |
| Create UI to change map views based on class | 6 | Caleb |

**Acceptance Criteria:**

- Given that the filter design is correctly implemented, when a student/TA chooses a class to sort by through the UI, then the map will only display study sessions for the specified class.
- When a student joins a class, the backend will have tied his/her user to the class and set his/her role for the class to the role assumed by the key given (three different keys are given for a single class for professors, TAs, and students).

**User Story #3:**

*As a Student/TA/Professor I want to be able to view study sessions that are near me on a map.*

| Description | Time | Person |
|---|---|---|
| View and interact with a map on the main page. | 3 | Scott |
| Create timed logic to retrieve new study sessions from backend using API | 10 | Scott |
| Update map UI to display study session icon | 6 | Scott |
| Create API Gateway endpoint to retrieve study sessions by class | 2 | Ben |
| Create Lambda function to retrieve study sessions by class | 8 | Ben |

| | | |
|---|---|---|
| Create RDS table for class to study session assignments, used SQL queries | 5 | Ben |

**Acceptance Criteria:**

- Given a user successfully logs in, when that user opens the app he/she should be able interact with a map.
- Given a user is registered in at least one class, when that user opens the app he/she should be immediately able to see open study sessions (unfiltered).
- When a user remains with the app open for a set interval of time, the app will poll the backend and retrieve any new sessions that have been registered.

**User Story #4:**

*As a Student/TA/Professor I want to create a new study session.*

| Description | Time | Person |
|---|---|---|
| Create the UI button on top of the map and integrate option to start a new study session | 2 | Christian |
| Create the UI window to fill out details of the new session, including the form analysis. | 8 | Christian |
| Create study session form analysis for the fields of class, location, time limit, location description, and study material description and submit form upon successful analysis. | 8 | Ben |
| Handle/testing logic for study session form analysis (ensuring proper format, required fields are filled, and prevent SQL injection). | 10 | Scott |
| Create new API Gateway endpoint to create a new study session in backend. | 2 | Caleb |
| Create Lambda function to create new session record in database from API call. | 8 | Adam |
| Create RDS table for class to session assignments, and also a table for session data. | 8 | Caleb |

**Acceptance Criteria:**

- Given a user successfully logs in, he/she should be able to create a new study session from the map.
- When a user creates a new study session, he/she should be able to list study sessions details (class, location, time, description, etc..)
- Given a user is a TA or Professor for the class the session is being submitted for, when that user submits a session, he/she can choose a location not at the current phone's GPS location.

## Remaining Backlog:

(Working Progress: 14/48)

**Functional:**

Students:

1. ~~I want to create a new study session.~~
2. ~~I want to display to other students what topics will be studied at the study session and other session specific data (a session description).~~
3. ~~I want to set a time frame that the study session will be active.~~
4. ~~I want to have my study session's location displayed to my other classmates.~~
5. I want to automatically end my study session when I leave the nearby area.
6. I want to transfer ownership (transfer host) of a study session.
7. ~~I want to filter what study sessions appear on my map based on class.~~
8. ~~I want to be able to view study sessions that are near me on a map.~~
9. I want to be able to join a study session created by a classmate.
10. I want to see the other students that are attending an active study session.
11. I want to show I joined a study session to my classmates in the session.
12. I want to invite friends to a study session.
13. I want to post to any active study session to communicate with those who are a part of it.
14. I want to remove my post from public view at an active study session.
15. ~~I want to sign up and login to the app.~~
16. I want to add friends to a friends list and have them appear differently on maps.

Professors:

17. I want to see who has been to past study sessions.
18. I want to see the dates of past study sessions.
19. I want to view the topics discussed from past study sessions.
20. I want to see posts from active study sessions.
21. ~~I want to view active study sessions on a map.~~
22. I want to post and view comments on an active study session.

23. I want to create professor sponsored study sessions without having to be located nearby.
24. I want to be able to remove any post on any active study session.

    TA's:

~~25. I want to filter what study sessions appear on my map based on class.~~
~~26. I want to be able to view study sessions that are near me on a map.~~
27. I want to be able to join a study session created by a classmate.
28. I want to see the other students that are attending an active study session.
29. I want to show I joined a study session to my classmates in the session.
30. I want to invite friends to a study session.
31. I want to post to any active study session to communicate with those who are a part of it.
32. I want to remove my post from public view at an active study session.
~~33. I want to create TA sponsored study sessions without having to be located nearby.~~
~~34. I want to display to other students what topics will be studied at the study session and other session specific data (a session description).~~
~~35. I want to set a time frame that the study session will be active.~~
~~36. I want to have my study session's location displayed to my other classmates.~~
37. I want to automatically end my study session when I leave the nearby area.
38. I want to transfer ownership (transfer host) of a study session.
39. I want to remove any student post on any active study session.

**Non-Functional:**

40. Users must be able to use the app on an android device.
41. The interface will be simple and user friendly.
42. The app will be accessible on all screen sizes and resolutions but will be built primarily for late generation Android, iPhone and large browser screens.
43. The app must implement location (and GPS) services.
44. The app will integrate a back-end web service using AWS, to manage all user data.
45. The app will have fast response times, with no back-end call delay above a second.
46. The app and backend service should support at least 100 concurrent users during this development phase.
47. Our app will prevent security and privacy breaches. All data will be encrypted during transit between our service and our client (HTTPS). And all necessary client encryption keys and passwords will be stored as securely as possible on the client devices (Android Keystore System on Android devices).
48. Our app will enforce user roles (student, professor, teaching assistant) and their associated permitted actions.