

Sluglet: Sprint 3 Review

Priority	User Story	Tasks	Story Points
10	As a user, I want to see all the classes I selected appear on the map so that I can easily find individual classrooms	<ul style="list-style-type: none"> <input checked="" type="checkbox"/> Clean location data so that the passed in string can be found by map API (either in web scraper, or in the app itself) - 4 SP <input checked="" type="checkbox"/> Create a "Map Screen" composable and viewmodel with its own navigation icon - 2 SP <input checked="" type="checkbox"/> Implement Map button on course tiles so that, when clicked, navigates to the map screen and triggers a function call - 2 SP <input checked="" type="checkbox"/> Display an integrated map on the map screen - 13 SP <input checked="" type="checkbox"/> Write a function to place a marker on the map at a given location - 8 SP <input checked="" type="checkbox"/> Populate map with markers for user's registered courses based on what is present in firestore - 4 SP 	33
6	As a user, I want to be notified when it's time for class so I don't forget	<ul style="list-style-type: none"> <input checked="" type="checkbox"/> Create a basic service interface and impl file for push notifications - 1 SP <input checked="" type="checkbox"/> Code a function that sends a push notification to the device when triggered - 7 SP <input type="checkbox"/> Set up Alarm Manager to wake up the app and perform a function at a designated time - 7 SP <ul style="list-style-type: none"> a Handle dateTime events. (Probably need to parse the dateTime field of the CourseData object) 	15
3	As a user, I want the app to have multiple settings to customize my experience and make it my own.	<ul style="list-style-type: none"> <input checked="" type="checkbox"/> Create an empty Settings Page with navigation and scrolling capabilities - 2 SP <input type="checkbox"/> Create a Calendar theme composable that can change the color of the calendar - 1 SP <input type="checkbox"/> Create a composable settings option to sign out the current user (AccountService) - 1 SP <input type="checkbox"/> Create a composable to choose between light/dark mode or use the system default - 1 SP <input type="checkbox"/> Create a composable to enable/disable push notifications - 2 SP <input type="checkbox"/> Combine composables into a final settings screen - 2 SP 	9

4	As a user, I want to use the app before signing in so I can start scheduling immediately	<input checked="" type="checkbox"/> Create an anonymous user by default when starting the app - 2 SP <input checked="" type="checkbox"/> Add support for anonymous users in functions that call AccountService - 5 SP <input checked="" type="checkbox"/> Allow anonymous users to store data in firestore - 4 SP <input checked="" type="checkbox"/> Link anonymous to a new or existing user when signing up - 3 SP	14
5	As a user, I want to have a home page where I can perform multiple functions at once so I can have everything in one place.	<input checked="" type="checkbox"/> Display user's currently active courses on home screen - 3 SP <input checked="" type="checkbox"/> Settings gear that opens side menu for settings - 4 SP <input checked="" type="checkbox"/> Back button to navigate from settings to home screen - 2 SP <input checked="" type="checkbox"/> Use back button on Android to navigate back to home - 2 SP	16
6	As a user, I want to see a calendar with all of my classes scheduled so I can visualize when my classes are	<input checked="" type="checkbox"/> Display a calendar on screen with the current day highlighted [5] <input checked="" type="checkbox"/> Display registered classes on selected calendar date (see sample project) [5] <input checked="" type="checkbox"/> Display registered classes below calendar with course information for selected date [3]	13
3	As a user, I want my classes to be loaded automatically so I don't have to manually input them all	<input checked="" type="checkbox"/> Create and propagate home screen with user's classes [4] <ol style="list-style-type: none"> a. Implement retrieval of user specific info from firestore 	14

Continue

- Reaching out for help and communicating regularly with teammates working on similar tasks
 - This was by far our most communicative Sprint, and as a result it was also our most productive so far
- 8 Hour work Minimum
 - Putting in a minimum of 8 hours of solid work each week allows us to keep on target with story point objectives
 - Creates a standard that we can hold ourselves accountable with
- Logging hours in Asana
 - Logging our work-hours in Asana helped us keep track of hours for both the Work Minimum and Burnup Chart
- Committing to working branches early and often
 - Creating a changelog by continuously updating branches helps keep track of progress.

Start Doing

- Refine [Definition of Done](#) from start of Sprint 3
 - Our previous Definition of Done did not cover integration thoroughly, and was not sufficient for user stories
 - New definition should be more explicit
- Refine loose rules into a [Coding Standards](#) document to better define project formatting
 - Having a single source for Coding Standards dispels uncertainty and confusion around formatting
 - Will help keep our code clean
- Create a Known Bug List to keep track of and fix errors
 - As we add additional features, unexpected interactions between new and existing functionality require fixing
 - Creating a list to track these issues will help everyone identify and solve them.
- Add a [Systems under Test](#) document to standardize and clarify functionality tests
 - Testing features has been largely informal thus far, this document will help standardize the process.
- Add basic [documentation](#) to function headers to understand them at a glance

Stop Doing

- Merging with the expectation that the process is unidirectional
 - Merging creates a bidirectional link between branches, and makes additional work in that branch inconvenient
 - Preparations should be made if a branch is expected to be worked on *after* merging with another branch
- Using a demo branch for presentations
 - Future demonstrations should be on the Main Branch to demonstrate that all code is both functional and fully integrated.

Burnup Chart

