Sprint 2 Plan

Product name: SmartShift

Team name: None / Same as Product Name

Sprint Completion date: Tuesday, May 09, 2023

Revision number: 1.0 (Release Plan 1.1)

Revision date: 4/25

Scrum Master: Christie

Goal: Learn React Native + Firebase (Prof recommended), set up the development environment, and develop the basic functionality of the app.

Spikes & Resources

- Learn React Native + Firebase
  - Prof Recommended Resources
  - O'Reilly library (also prof recommended)
    - https://learning.oreilly.com/videos/react-native/9781789139747/ (React Native course)
  - https://reactnative.dev/ (React Native)
- Collapsible Lists
  - https://blog.logrocket.com/building-react-native-collapsible-accordions/
- React Native Push Notifications
  - https://reactnative.dev/docs/pushnotificationios

    - ➢ https://blog.logrocket.com/react-native-push-notifications-firebase-cloud-messaging/ (React Native push notifications with Firebase Cloud Messaging)
    - ➢ https://duckduckgo.com/?q=firebase+react+native+push+notifications&t=ffab&atb=v344-1&ia=web (search on “firebase react native push notifications”)
- ❖ Account Set Up
    - ➢ https://blog.logrocket.com/email-authentication-react-native-react-navigation-firebase/ (Setting up email authentication with React Native, react-navigation, and Firebase)

Task Listing, organized by user story

User Story 3.0: As a student, I want to access my account via different devices. (5 story points)

1. Task 3.1: Set up firebase for external login with email and phone number (2 hours)
2. Task 3.2: Set up firebase for login with username and password (2 hours)
3. Task 3.3: Set up registration & login page (2 hours)
4. Task 3.4: Set up firestore to store user preferences (2 hours)
5. Task 3.5: Allow user to stay logged in after closing app, and allow user to log out (2 hours)

Done Conditions:

- ➔ User Story 3.0: Users can login (externally or internally) with email and phone number or username and password with a good UI registration and login page which stores everything in firebase. User’s stay logged in after closing the app and can sign out if desired. On a local machine/simulation without errors.

Total for user story 3.0: 10 hours

User Story 4.0: As a forgetful student, I want to receive notifications before an upcoming deadline to remind me to finish an assignment and when a timer goes off. (1 story points)

1. Task 4.1: Make notifications based on the deadlines on the list(?) (3 hours)
    a. Ask for permissions from user
    b. 1 day before the deadline (by default)
    c. Option to change alarm noise (1 h)
        i. Put it into a little tab thing under the timer so user can change noise as timer is existing (going down) (like timer page on phone visually)

Done Conditions:

- ➔ User Story 4.0: User can get notification a day before the deadline **when the app is closed**. User can change the alarm noise.

Total for user story 4.0: 3 hours

User Story 5.0: As someone with specific needs or preferences, I want to be able to customize my personal settings for notifications (2 story points)

1. Task 5.1: Make a page allowing users to customize the settings (7 hours total)
    a. Settings page (2 h)
        i. Navigation + UI
    b. Option to toggle notifications for alarm (1 h)
    c. Option to toggle notifications for deadlines (2 h)
    d. Option to change work/break timer (how long it is - default settings) (2 h)

        i. The flow timer page changes temporarily - it’s changes may not stick through an app reload and change back to default
        ii. While the settings will permanently change the default settings that the app pulls timer from. (changes the hardcoded default to a user set new default set in the settings page - ask Liza if need further explanation)

2. Task 5.2: Set user name / icon / profile (2 h)
    a. Preset defaults
    b. Preset set of images to choose icon from for users own icon

Done Conditions:

- User Story 5.0:

Total for user story 5.0: 10 hours

User Story 6.0: As a student with multiple tasks to do, I want a way to keep track of all these tasks and add them to a list as well as mark them complete as I finish them. (5 story points)

Filtering FlatList (current ideas):

https://stackoverflow.com/questions/70095887/how-to-filter-items-in-flatlist-in-react-native

1. Task 6.1: Make a list (2 hours) (to do list - that is automatically showing - collapsible upon tap)
    a. User can input tasks (1 hrs)
    b. User can input the tasks’ deadlines (3 hrs)
    c. Users can delete a task. (1 hrs)
    d. Users can edit/update a task. (1 hrs)
2. Task 6.2: Make a list of completed tasks

a. Create a collapsible list of completed tasks (which starts off collapsed and can be expanded upon a tap) (2 hrs)

b. Can delete a completed task. (1 h)

c. Unmark a task as completed so it returns to to-do list (1 hr)

Done Conditions:

- ➔ Users can add/edit/delete a task. Task includes description, due date, and if it is complete or not.

Total for user story 6.0: 12 hours

User Story 7.0: As a student who gets overwhelmed by large assignments, I want a way to break my tasks down into smaller tasks so that I can manage them more easily (8 story points)

https://stackoverflow.com/questions/67113767/how-to-display-different-content-in-a-flatlist

1. Task 7.1: Allow the user to create sub-tasks to a larger task (4 h)
    a. Stand alone tasks vs. creating one large task with one sub layer of sub-tasks.
    b. Collapsible large task to tap show all sub tasks under it
    c. If some sub tasks are complete they still stay under large task if the large task is still incomplete
    d. Large task can be either marked complete (even if some sub tasks not yet marked complete) or be automatically marked complete once all sub tasks have been marked complete.

Total for user story 7.0: 4 hours

- ➔

User Story 8.0: As a student I want to keep **track of things that I've completed** already.

- Difference b/w task being marked complete (still stored) and it being deleted from memory entirely

1. Task 8.1: (4 h)
    a. Show complete tasks only vs. in progress tasks (filter tasks list)
    b. Allow user to mark a task complete (or back to incomplete task)
    c. Allow user to delete a task entirely (different from marking it complete)

Total for user story 8.0: 4 hours

Done Condition:

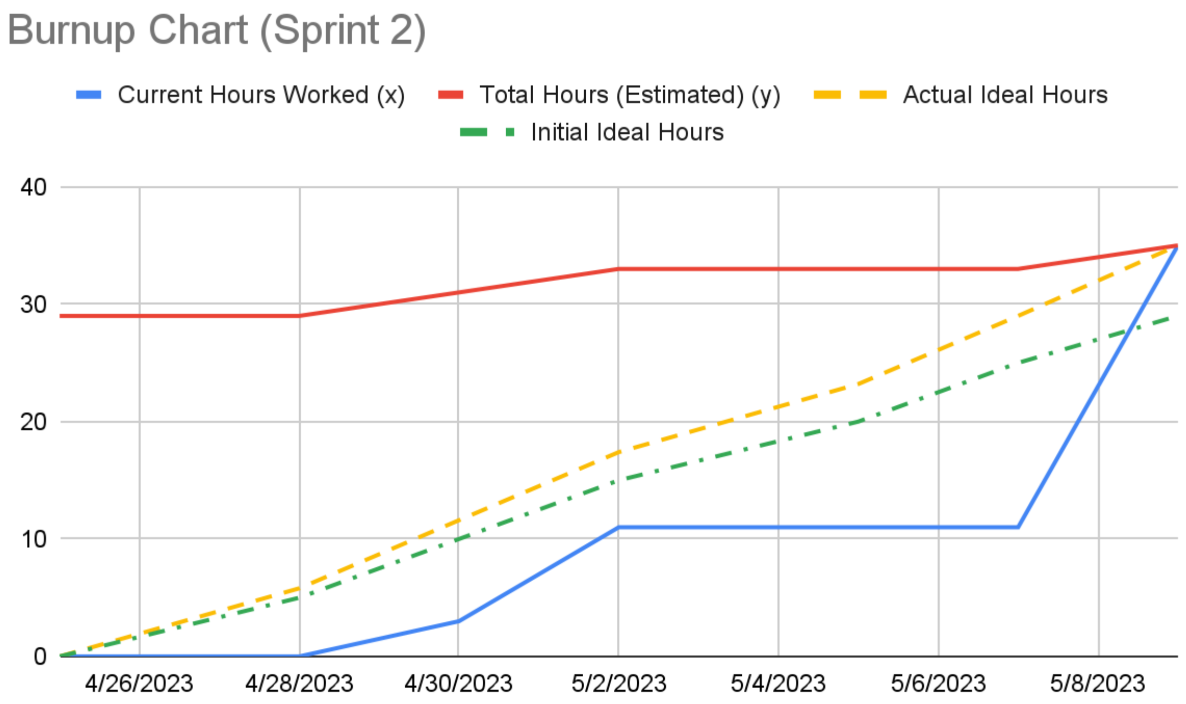
➔ User can delete task. User can mark a task complete/incomplete. User can filter tasks based on if it is complete/incomplete.

Team Roles

- Clarence: Product Owner
- Liza: Team Member
- Annie: Team Member
- Skyler: Team Member
- Christie: Scrum Master

Initial Burnup Chart

- Graph: (total hours (y-axis) vs. hours actually done so far (as of _ scrum meeting) (x-axis)) https://docs.google.com/spreadsheets/d/1au_oWLCHtTuuMn3sqmIrYYQ-7x1d1KbjpKSQwVR8qOw/edit#gid=0



Initial Scrum Board

Titled user stories	Tasks not started	Tasks in progress	Tasks completed
3.0 [1]		Task 3.1 (Clarence)	Task 3.2 (Clarence) (3h) Task 3.3 (Clarence) (2h) Task 3.4 (Clarence) (4h) Task 3.5 (Clarence) (2h)

4.0		Task 4.1 (Christie)	
5.0	Task 5.2	Task 5.1 (Skyler)	
6.0			Task 6.1 (Liza) Task 6.2 (Liza) (~20h total)
7.0	Task 7.1 (Annie)		
8.0			Task 8.1 (Liza) (~4h total)

Infrastructure Tasks:

- **Make the Timer UI better looking (2 h) (Annie) (COMPLETED)**
- **Merging sprint 1 stuff (2 h) (Skyler) (COMPLETED)**

Scrum Times

Even if online you need to have cameras on.

- Tuesday 7:30 pm (In person)
- TA: Thursday 1:50 pm - 2:35 pm (zoom link currently)
- Friday 7:30 pm - Online - Discord
- Sunday 3:00 pm - Online - Discord