

Heading: Sprint 1 Plan, TP, TimeLords, 12 July 2016, #1, 14 July 2016

Sprint Goal: Create functional android application that can add tasks that have the name, project, due date, notes, and boolean completed attributes and are displayed under its associated project on the apps home page.

User stories for release:

Sprint 1 User stories

1. (3) As a user I want an android app user interface to open and use my app.
2. (5) As a user I want to be able to quickly make new tasks so that I can engage in “brain dump” quickly when I think of a task.
3. (5) As a user I want to be able to create a new project so that I can organize my tasks into groups.
4. (2) As a user I want to be able to organize my tasks into projects so that I can separate tasks and view them more easily.
5. (8) As a user I want to add Task attributes, and easily edit them to distinguish them.
6. (2) As a user I want to be able to mark tasks as done so that I know what tasks I have completed
7. (1) As a user I want to be able to schedule a due date for my task so I know when to get it done.
8. (2) As a user I want to be able to add notes to each task so I can have a place to put other relevant information not provided by the pre-existing list of attributes
9. (2) As a team member I want to have a super class that holds all instances of projects so that later it will be easier to create a collapsible project list in our menu.
10. (3) As a user I want an user interface that allows me to swipe from the menu page to other pages *flawlessly*, so I can navigate through the application
11. (13) As a user I want the app to REMEMBER my tasks when I exit the app because otherwise it would be completely useless to even use this app.

Sprint 1 Task Listing

Task 1 Set up developer’s environment (1)

Task 2 Download a proper android emulator (1)

Task 3 Use and tweak a template to produce a starting UI based on our sprint 1 sketches (menu) (1)

Task 4 create Task object and fields (0.5)

Task 5 create Task addition (“new-task”) GUI button (1.5)

Task 6 create Project object (0.5)

Task 7 decide and implement Project’s underlying data structure (1.5)

- Task 8 create main homepage for app listing projects and associated tasks (2)
- Task 9 add new attributes to the Task object(0.5)
- Task 10 create a debug-like method to read Task's attribute values (1)
- Task 11 create an active prototype to *edit* Task obj's attributes - make a new page(2.5)
- Task 12 create popup page in app for viewing/editing task attributes (3)
- Task 13 allow task attributes to be changed via the UI (2)
- Task 14 create UI element that shows task as complete (mark box or cross out) (1)
- Task 15 add due date field to task class (0.5)
- Task 16 add attribute to task page (0.5)
- Task 17 make the notes attribute editable via the UI (1)
- Task 18 create super class of projects (3)
- Task 19 implement swiping function (4)
- Task 20 research to decide efficient memory saving implementation (4)
- Task 21 implement file to save user data (10)
- Task 22 write code to reload user data when reopening app 10)

Burn up Chart:







