

Sprint 4 Plan

Product name: Rootify

Team name: Rootify

Sprint Completion: 12/03/17

Revision number: 2

Revision date: 11/30/17

Goal: Add the track name and artists under the tree node for a better experience. Create a dialog popup / box that allow user to name their new playlist. Create About and Help page and clean up the code.

Tasks:

User story 1: “As a Spotify user, I want to be able to see a node’s track name and artist if it’s a track and a node’s artist name if the node represents an artist so that I can easily distinguish between nodes with the same album art or have similar images”

- Add track name & artists below the track’s tree node
 - (Track should be white, and artist should be green) **(0.5 hours)**
- Add artist’s name under the tree node
 - Artist name should be green **(0.5 hours)**

Total Hour: 1 hour

User acceptance criteria:

- the text must be easy to read under the tree
- The text should not interfere the tree

User story 2: “As a Spotify user, I want the ability to name the new recommended playlist based on my selected artists, tracks, and/or genres and filtering options so I distinguish my name playlist from my other playlists.”

- Create a pop up dialog box after the user press “Generate Playlist” **(1 hour)**
- In the dialog, it should contain “New Playlist Name” and below it’s a textbox where the user can type the name of the playlist. **(1 hour)**
- Add “Save” button under the text box **(.2 hour)**
- Add X button on the right corner (if the user change their mind) **(.1 hour)**
- After the user click, save there a follow up message that “it has been saved on their playlist account” **(.3 hour).**

Total Hour: 3 hours

User acceptance criteria:

- The new playlist name is pass to the user's spotify account with the new generated playlist.
 - Dialog pop up should have a nice transition
 - (x) button on the upper right corner, if the user change their mind.
-

User story 3: "As a Spotify user, I want a help and about page so I have a better understanding how to use the program if I am confused."

- Create a help page with FAQ and details about the program (**.5 hour**)
- Create an about page (**.5 hour**)

Total Hour: 1 Hour

User acceptance criteria:

- Must have really thought out descriptive content for the help page
 - The help page must include pictures
 - It should be neat and easy to read
 - The about page should be short and simple description, which include all the developers and link through linkedin / github.
-

User story 4: "As a Spotify user, I want to be able to search for a track and add it to my selected tracks so that I can improve my recommended playlist."

- Create search box (using Search2) (**0.2 hours**)
- Load relevant track results from search in the form of rows (using Search2) (**1 hour**)
- Add the chosen track to selected tracks when a row is clicked (**0.5 hours**)

Total Hours: 1.7 hours

User acceptance criteria:

- Must be able to search for a track
 - For each row, must show the track album art and track name, along with the primary artist.
-

User story 5: "As a Spotify user, I want to be able to search for an artist and add it to my selected artists so that I can improve my recommended playlist."

- Create search box (using Search2) (**0.1 hours**)
- Load relevant artist results from search in the form of rows (using Search2) (**0.5 hours**)
- Add the chosen artist to selected artists when a row is clicked (**0.2 hours**)

Total Hours: 0.8 hours

User acceptance criteria:

- Must be able to search for an artist
- For each row, must show the artist image and name

User story 6: “As a Spotify user, I want to be able to reset the tree so that I can rediscover related tracks and artists.”

- Create reset tree button at the top (**0.2 hours**)
- Empty the root’s children arrays and then update and center the root node (**0.2 hours**)

Total hours: 0.4 hours

User acceptance criteria:

- Must be able to click on the reset tree button
- Should collapse the tree (root’s children’s children)

User story 7: “As a Spotify user, I want to be able to minimize a node’s children so that I can make the tree cleaner and easier to understand and read.”

- Create up-triangle for an opened node (**0.2 hours**)
- Store the node’s children in a temp. array in the corresponding node (**0.2 hours**).
- Call update() function on that node (**0.3 hours**).

Total hours: 0.7 hours

User acceptance criteria:

- When the up-triangle is selected for a node, it should minimize children
- Up-triangle should disappear after doing this.

User story 8: “As a Spotify user, I want to be able to zoom in and out of the tree and click and drag so that I can have control over the tree.”

- Add zoom listener function to the group SVG, define default zoom (**0.2 hours**)
- Handle scaling in centerNode() properly (**0.2 hours**)
- Add panning (similar to dragging) to ensure that nodes that go off the screen are brought back into view (**3 hours**)

Total hours: 3.4 hours

User acceptance criteria:

- Should be able to zoom in and out with scrolling over the main tree layout (SVG)
- Should be able to hold mouse click and drag around the tree.

Total overall hours: 12 hours

Team roles:

Tristan Iverson:	Product Owner , Developer
Jonathan Ortiz:	Developer
Kristine Nguyen:	Scrum Master , Developer
Chloe Jiang:	Developer

Brian Tran: Developer

Initial task assignment:

Tristan Iverson: User Story 8, User Story 7, User story 6

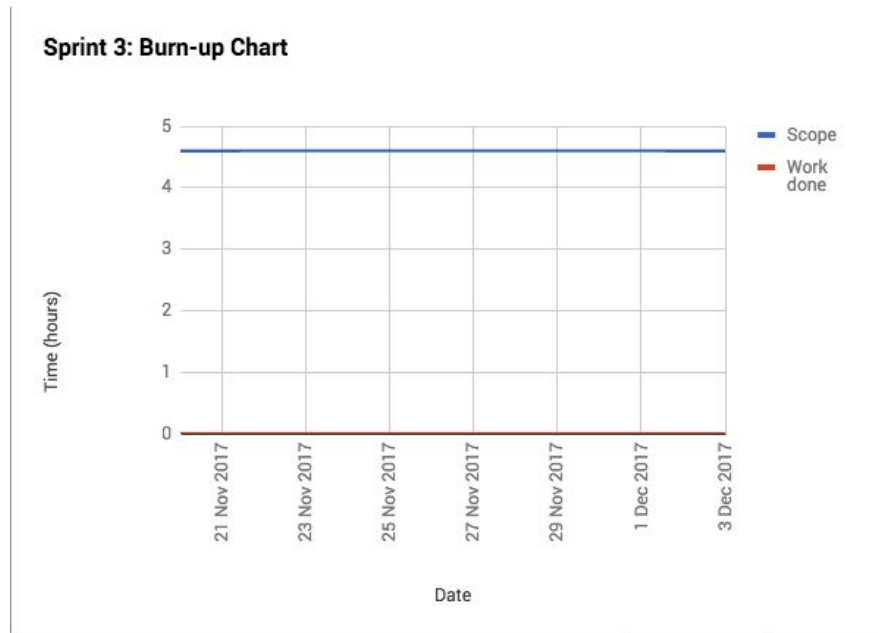
Jonathan Ortiz: User Story 1.

Kristine Nguyen: User Story 1.

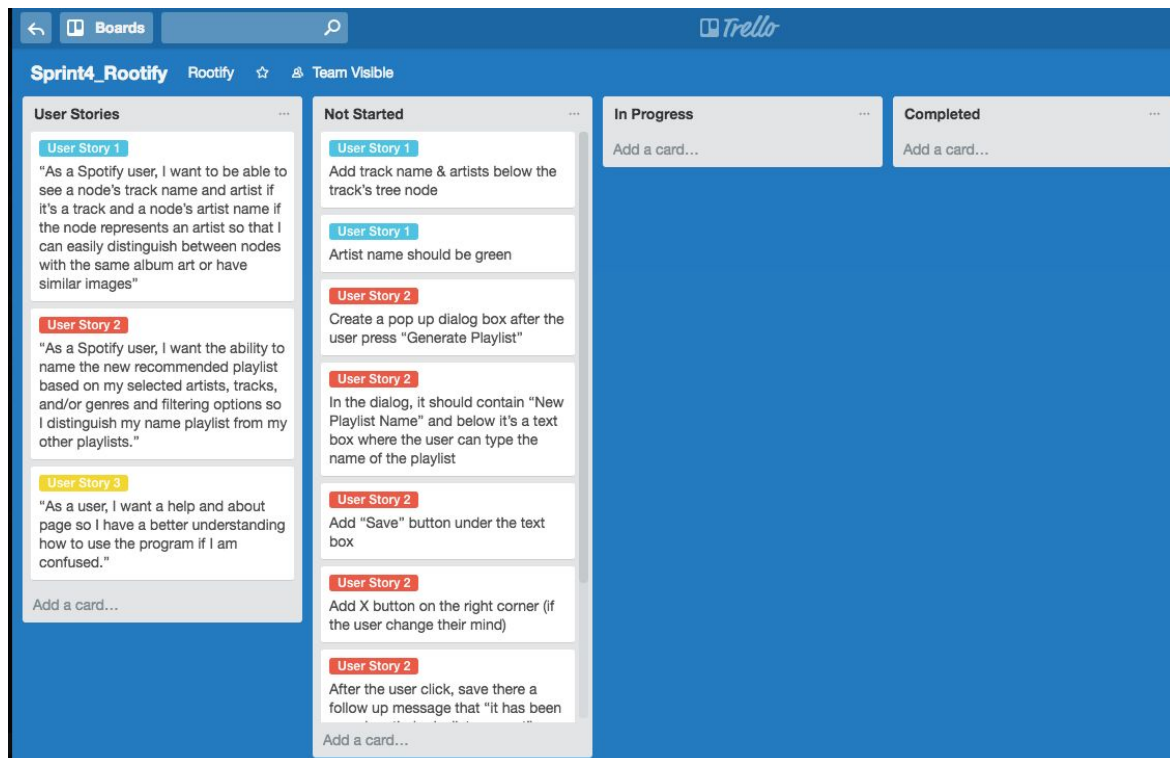
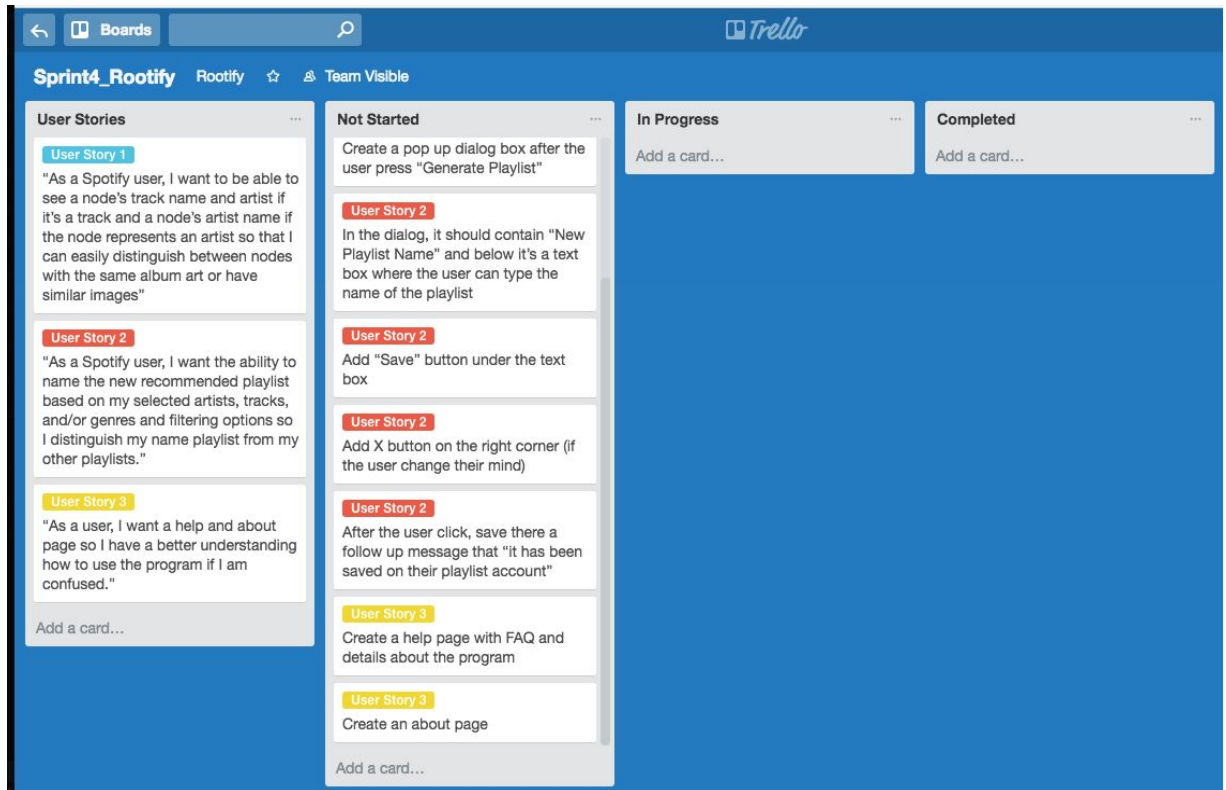
Chloe Jiang: User Story 3.

Brian Tran: User Story 2.

Initial BurnUp:



Initial Scrum Board:



Scrum Time Meetings:

Monday, 7:00 PM, BE 302

Wednesday, 10:00 AM, BE 302 (with Reihaneh)

Tuesday/Thursday 4:00 PM, BE 302