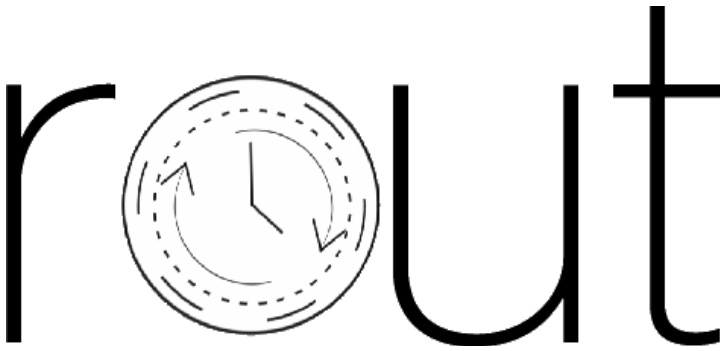

CS 30700 Sprint 3 Planning Document
04.06.2021

Team 17 :
Steven Bass
Luke Irons
Logan Sweeney
William White
Austin Wilson

Sprint 3 Overview:

This goal for this sprint is to add quality of life features and connect more UI features to backend and algorithm features implemented in previous sprints. We also plan on finishing up and polishing user stories that we did not entirely complete last sprint. Lastly, we plan on doing more testing this sprint.

Scrum Master: William White

Meeting Plan: Monday/Wednesday/Friday @ 6:30pm

Risks and Challenges:

Now that we have a partially complete and entirely integrated website, the user stories for this sprint are more for polishing rather than large feature implementation. That being said, we do have a couple user stories that may pose a challenge this sprint. First, we would like to be able to load saved routes from the database so that they are displayable on the user statistics page. Most of the groundwork is already laid for this feature so the work ahead is mostly integrating. Another big challenge ahead is retrieving information from Google Maps Street View and displaying it so that it is useful to the user. This user story is much different than the other Google Maps API user stories so it will require more substantial research.

Sprint 3 User Stories:

User Story #1:

As a user, I would like to be able to export saved routes as a file for sharing purposes.

#	Description	Estimated Time	Owner
1	File will be exported as a text file.	4 Hours	Logan
2	File will be downloadable from the website.	3 Hours	Logan
3	File will be copy/pastable to the website for future use.	4 Hours	Logan

Acceptance Criteria:

- File should be small enough to share over email, compression may be necessary but not required.
- File download should not crash the server or cause significant slowdowns for other users.
- File upload should be copy/pasting of the text within the file.
- Manual testing will be conducted to make sure exported file is reusable.

User Story #2:

As a user, I would like to be able to share routes with other users through a unique code.

#	Description	Estimated Time	Owner
1	Route will be stored in DB upon the user requesting unique code.	4 Hours	Logan
2	Unique code will be generated and stored in the DB alongside the Rout.	2 Hours	Logan
3	Users can input the route code on the main screen to re-generate the previously made route.	4 Hours	Logan

Acceptance Criteria:

- Code generated should be a random 6 character string.
- Generated code should be unique (not already in the DB).
- If no route is inputted, code will not be generated and an alert message will appear signaling the user to input a route first.
- Manual testing to make sure if duplicate codes are generated, a new code is generated until one is generated that does not exist in the DB.

User Story #3:

As a user, I would like to be able to share my routes through popular social media services (Facebook, Twitter, Instagram).

#	Description	Estimated Time	Owner
1	Social media login will pop up and allow users to share their routes.	6 Hours	Logan
2	Route will be shared similar to generated code and will have an image with the code generated for social media sharing.	3 Hours	Logan

Acceptance Criteria:

- Social media sharing works for Facebook, Twitter, and Instagram.
- Generated code should follow same acceptance criteria of User Story #2.
- Route is stored regardless of successful social media sharing.
- Manual testing of social media sharing (does it appear on the user's social media?).

User Story #4:

As a user, I would like to save a route for future use.

#	Description	Estimated Time	Owner
1	Use the Google Maps Directions API to export routes as JSON objects.	3 Hours	Steven
2	Post the exported JSON object to the database	2 Hours	Steven

Acceptance Criteria:

- Sufficient information should be saved to the database, such that the route can be recreated without additional input from the user.
- The saved objects should be accessible for use in User Stories #5 and #6
- Manual testing with database view to ensure objects are being saved correctly

User Story #5:

As a user, I would like to view and use saved routes.

#	Description	Estimated Time	Owner
1	Implement parsing of the saved route database objects	3 Hours	Steven
2	Using the existing Google Maps integration, re-draw the route.	5 Hours	Steven
3	Implement UI element for users to see their saved routes	2 Hours	Steven

Acceptance Criteria:

- Routes should be imported to the client from the database (as stored in User Story #4).
- Saved routes should be accessible through browsing or from a unique code.
- Manual testing to ensure routes are successfully recreated.

User Story #6:

As a user, I would like to be able to export saved routes as a file for sharing purposes.

#	Description	Estimated Time	Owner
1	Implement a system for downloading files from the server through the web client	5 Hours	Steven
2	Save the file to be downloaded in a temporary environment.	1 Hour	Steven
3	Design UI elements for downloading a route which has been saved	2 Hours	Steven

Acceptance Criteria:

- Routes should be imported to the client from the database (as stored in User Story #4).
- Routes should be accessible through same means as in User Story #5 (browser/unique codes)
- Manual testing to ensure downloading is functional and UI elements function

User Story #7:

As a user, I would like to see how many calories I've burned after taking a route.

#	Description	Estimated Time	Owner
1	Design the layout of the details component.	1 Hour	Austin
2	Implement basic route information into the details component along with the calculated calories burned.	10 Hours	Austin
3	Manual testing of details component and information.	2 Hours	Austin

Acceptance Criteria:

- Details component is shown and hidden on click of the details button.
- Details component is designed to show statistics and information of the route in an appealing format.
- Calories are calculated by the route algorithm and stored as a variable.
- Calories burned variable is displayed in the details component.
- Testing will be done manually through the viewing of the UI to insure UI looks and works as intended.

User Story #8

As a user, I would like to view my total calories burned over all routes I've taken.

#	Description	Estimated Time	Owner
1	Design addition of total calories to calories section of the statistics page.	1 Hour	Austin
2	Implement database queries to calculate total queries and it being displayed.	3 Hours	Austin
3	Manual testing for total calories burned.	1 Hour	Austin

Acceptance Criteria:

- Total calories burned component is designed to show total calories burned of all routes in an appealing format.
- Total calories should be displayed on the viewing of the statistics page.
- Total calories will display "N/A" if no routes have been run/stored by a logged in user.
- Testing will be done manually through the viewing of the UI and database to insure UI and database look and work as intended.

User Story #9

As a user, I would like the information and statistics of previous routes to be stored.

#	Description	Estimated Time	Owner
1	Research how to best store routes in the database (compare directly saving routes or information used to create them).	2 Hours	Austin, Will
2	Create a method to save the current route as an object and calculate additional statistics.	4 Hours	Austin, Will
3	Implement the conversion of the object created and its insertion into the database.	4 Hours	Austin, Will
4	Manual testing for the storage of route information and statistics.	2 Hours	Austin, Will

Acceptance Criteria:

- The map object created with the route algorithm is stored in a variable alongside with additional needed information.
- The variable is correctly parsed and stored in the database for use with other parts of the project.
- Testing will be done manually through the viewing of the UI and database to insure UI and database look and work as intended.

User Story #10

As a user, I would like to be able to rate a route following its completion.

#	Description	Estimated Time	Owner
1	Add route rating UI element to home page.	2 Hours	Austin
2	Connect route rating to database information.	3 Hours	Austin
3	Manual testing of route rating front end and database.	1 hour	Austin

Acceptance Criteria:

- Route rating component is designed to show possible rating of a route in an appealing format.
- Rating can be submitted and stored in the database on the click of a submission button.
- Testing will be done manually through the viewing of the UI and database to insure UI and database look and work as intended.

User Story #11

As a user, I would like a “difficulty score” based on the intensity of my route.

#	Description	Estimated Time	Owner
1	Create a method to convert route characteristics into a quantifiable score.	3 Hours	Luke
2	Implement method in the route algorithm.	4 Hours	Luke
3	Test and debug the score creation.	2 Hours	Luke

Acceptance Criteria:

- Given that the score is generated correctly, when the user generates a route to use, the difficulty score should be viewable by the user.
- Given that the route is generated correctly, when the difficulty score is created, routes of greater length should have a higher score.
- Given that the route is generated correctly, when the difficulty score is created, routes with more hilliness should have a higher score.
- Manual testing will be done by inputting routes of varying difficulties and comparing the generated difficulty scores.

User Story #12

As a user, I would like to know the pace of the previous route.

#	Description	Estimated Time	Owner
1	Create a method to convert time (either user given or not) into a pace.	3 Hours	Luke
2	Implement a method to convert time into a pace + return the value to the user.	4 Hours	Luke
3	Test and debug functionality.	2 Hours	Luke

Acceptance Criteria:

- Given that the user has completed a route, when the time spent on the route is received (either through user input or another suitable method), the time should be correctly converted into a pace.
- Given that the user has completed a route, when the pace is being generated, the route distance + time should be utilized to ensure a correct pace (in units of minutes per mile).
- Given that the user's time has been collected, when the correct pace has been calculated, the pace should be displayed to the user.
- Manual testing will be done through generating routes of multiple lengths/times and ensuring that the calculated pace is correct (and correctly displayed).

User Story #13

As a developer, I would like a Javascript framework for making calls to the Google Maps API.

#	Description	Estimated Time	Owner
1	Finalize integration of Typescript into Javascript/React.js.	2	Luke
2	Ensure Javascript framework works correctly with Google Maps API calls.	2	Luke
3	Test and debug algorithm and javascript framework.	2	Luke

Acceptance Criteria:

- Given that the algorithm has been implemented correctly, when a route is generated, the algorithm should make calls to the correct Google Maps APIs.
- Given that the correct Google Maps API calls have been made, the calls should return the information necessary to create a usable route.
- Given that the integration of the algorithm and UI is complete, when a route generation is requested, the framework should correctly use the algorithm and API calls to return usable route(s) to the user.
- Manual testing should be done through different route generations to ensure that the integrated React.js framework does not break under differing lengths and conditions.

User story #14

As a user, I would like to have a UI element to customize what type of route I am looking for.

#	Description	Estimated Time	Owner
1	Research placing UI elements either on top of the map or next to it	2 hrs.	Will
2	Create two UI elements that can be checked, allow for expansion too	3 hrs.	Will
3	Connect UI elements to the algorithm so that the customizations are taken into account.	2 hrs.	Will

Acceptance Criteria:

- The UI elements are simple, easy to use, and look good.
- The UI elements are correctly connect to the algorithm so when one is checked, appropriate adjustments are made when running the algorithm.
- Additional selection boxes are easy to add within the code for future expansion.
- Manual testing will be done to ensure that the UI elements make the correct adjustments.

User story #15

I would like to be able to see important turning points on my route through Google 360° Street View.

#	Description	Estimated Time	Owner
1	Research Google 360 Street View and how to pull images from specific places	4 hrs.	Will
2	Implement basic method to show Google Street View at beginning of route.	2 hrs.	Will
3	Create space under Details tab to show images from Google Street View	2 hrs.	Will
4	Implement method to show Google Street View images at turning points in the Route	4 hrs.	Will

Acceptance Criteria:

- A Google Street View image is visible under the details tab that shows the beginning of the route.
- A Google Street View image is visible for each change in street or turn in the route.
- Manual testing will be done in order to compare the generated images from the route with equivalent images from google.com/maps.

Sprint 2 - Incomplete User Stories, Tasks, & Acceptance Criteria

User Story #6:

As a developer, I would like to be able to view usage metrics of Rout.

#	Description	Estimated Time	Owner
1	Usage metrics should be viewable in the statistics page.	3 Hour	Steven
2	Usage metrics should be stored server side and updated as users interact with Rout.	2 Hours	Steven
3	Non-developers should be prohibited access from the statistics page	2 Hours	Steven

Acceptance Criteria:

- Metrics tracked should include number of accounts, number of users currently logged in, and number of routes saved across all users.
- Site statistics page should only be accessible by dev users.
- Site statistics should be accurate to the database at the time of fetching.
- Manual testing by comparing displayed data with stored database info.

User Story #11

I would like to find multiple routes from the same starting point.

#	Description	Estimated Time	Owner
2	Implement changes to current route generation algorithm	5 Hours	Luke
3	Test and debug functionality	2 Hours	Luke

Acceptance Criteria:

- Given that the algorithm is functional, when the user chooses to generate routes, the algorithm should generate multiple routes.
- Given that the user has chosen to generate routes, when the algorithm completes its API calls, multiple routes should display on the map UI.
- Given that multiple routes are being displayed, when the user wants to choose a specific route, they should be able to choose any of those shown.
- Given that the algorithm was successful, when the different routes are created, they should all be similar lengths.
- Manual testing will be done through running the algorithm with various inputs to ensure that multiple runnable routes are generated

User story #15

As a user, I would like to find a route by hilliness.

#	Description	Estimated Time	Owner
2	Create method that find routes with larger altitude changes and routes with less altitude changes	3 Hours	William
4	Test and Debug	3 Hours	William

Acceptance Criteria:

- Users are able to check UI element to create a route that has a relatively large number of altitude changes.
- Users are able to check UI element to create a route that has a relatively small number of altitude changes.
- Testing will be done manually using the UI element to ensure hilly route has more altitude changes than non-hilly route.

Sprint 3 Total Hours:
Steven: 30 Hours
Luke: 31 Hours
Logan: 30 Hours
William: 31 Hours
Austin: 30 Hours

Backlog:

Functional

User Account:

As a user,

- ~~I would like to find a new route without needing an account.~~
- ~~I would like to register a new Rout account.~~
- ~~I would like to login to my Rout account.~~
- ~~I would like to logout of my Rout account.~~
- ~~I would like to be able to reset my password through email.~~
- ~~I would like to be able to reset my password through SMS.~~
- ~~I would like to be able to secure my account with 2 factor email authentication.~~
- ~~I would like to be able to secure my account with 2 factor SMS authentication.~~
- ~~I would like to be able to customize my nickname on my profile.~~
- ~~I would like to be able to customize my profile with a profile picture.~~

User Interface:

As a user,

- ~~I would like the UI of the website to be easily understood and used.~~
- I would like a brief tutorial on the UI the first time I use the application.
- ~~I would like to adjust the color of the UI to dark mode.~~
- ~~I would like the website to have a visually appealing color palette.~~
- ~~I would like to be able to adjust default measurements between Miles/Kilometers.~~
- ~~I would like to have a UI element to customize what type of route I am looking for.~~

Route Planning:

As a user,

- ~~I would like for outside temperature to be displayed in the route planning screen.~~
- ~~I would like for outside weather conditions (rain, snow, etc) to be displayed in the route planning screen.~~
- ~~I would like for date/time information to be displayed in the route planning screen.~~
- ~~I would like to drop a pin to use as a start point.~~

- ~~I would like to use an address as a start point.~~
- ~~I would like to have autocomplete when typing an address.~~
- ~~I would like to be able to see important turning points on my route through Google 360° Street View.~~
- ~~I would like to find multiple routes from the same starting point.~~
- ~~I would like to find a route using distance.~~
- ~~I would like to find a route using time and a pace.~~
- ~~I would like to find a route by hilliness.~~
- I would like to find a route only on sidewalks.
- ~~I would like a "difficulty score" based on the intensity of my route.~~
- ~~I would like to plot routes with specific waypoints to pass through.~~
- I would like to track my progress along a route in real time via GPS.

Route Completion:

As a user,

- ~~I would like to save a route for future use.~~
- ~~I would like to view and use saved routes.~~
- ~~I would like to be able to export saved routes as a file for sharing purposes.~~
- ~~I would like to be able to share routes with other users through a URL.~~
- ~~I would like to be able to share my routes through popular social media services (Facebook, Twitter, Instagram).~~
- ~~I would like to be able to rate a route following its completion.~~
- ~~I would like to know the pace of the previous route.~~

User Statistics:

As a user,

- ~~I would like to see how many calories I've burned after taking a route.~~
- ~~I would like to be able to visualize the amount of calories I've burned.~~
- ~~I would like to view my total calories burned over all routes I've taken.~~
- ~~I would like to compare my pace with distances similar to those ran by famous or olympic runners.~~
- ~~I would like the information and statistics of previous routes to be stored.~~

Non-Functional

As a developer,

- I would like the pathfinding algorithm to run in polynomial time.
- ~~I would like administrator accounts which give access to backend data.~~

- I would like to be able to push changes to the UI without taking down the site for management.
- ~~I would like to implement a pathfinding algorithm which uses the Google Maps Directions API.~~
- ~~I would like to use the Google Maps API to create possible routes.~~
- ~~I would like a Javaseript framework for making calls to the Google Maps API.~~
- ~~I would like to be able to edit user information within the database.~~
- ~~I would like a MySQL database for storing account information.~~
- ~~I would like to be able to view usage metrics of Rout.~~