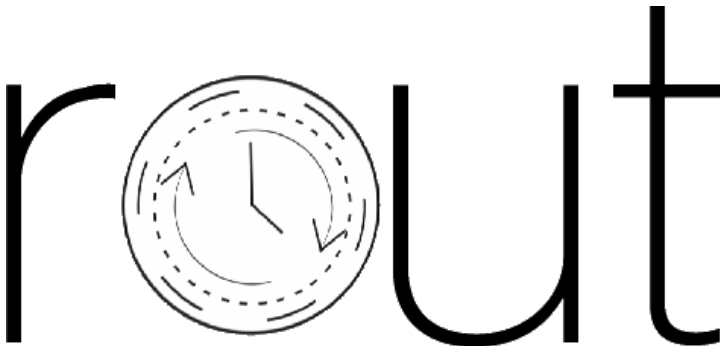

CS 30700 Sprint 2 Retrospective
04.05.2021

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

What went well?

Most of the user stories went relatively well. The two separate parts at the end of Sprint 1 (the route-finding algorithm and the frontend webpage) were fully integrated with one another, and none of the existing functionality in either was lost. Additionally, although we had occasional merge conflicts, in general we used git/github more cleanly than in Sprint 1, with feature branches and a stable main.

Sprint 2 Completed User Stories

User Story #1:

As a user, I would like to be able to reset my password through SMS.

#	Description	Estimated Time	Owner
1	Generate a unique 6 digit string to be sent through SMS for confirmation.	2 Hour	Logan
2	Send the 6 digit code to the user through SMS	3 Hours	Logan
3	Change user's password if correct 6 digit code is entered.	2 Hours	Logan

Completed:

Currently the service used for SMS password resetting and verification using Twilio. This service has limited use for free users so SMS's can only be sent to phone numbers that are physically linked to the service and the account. Because we do not plan to deploy this website as a long term service, we think not paying for the actual intended use of SMS pass work testing and verification is fine. After clicking the forgot password on the Login page, the user can then click the send code button, which if the phone number kept in the database is held within Twilio's verified phone numbers, the user will get an SMS to reset their password.

User Story #2:

As a user, I would like to be able to secure my account with 2 factor email authentication.

#	Description	Estimated Time	Owner
1	User account page allows enabling of 2FA	2 Hours	Logan
2	On login with correct login information, send user generated 6 character code.	3 Hours	Logan
3	Login the user if the code entered is the same as the code sent to the email.	2 Hour	Logan

Completed:

Email verification works by sending a code to a user's email address that is stored within the database. If the user selected the 2FA option when creating an account they will have to enter a 2FA code for every attempted login. On the login page they can click the Send 2FA Code button to send an email (or SMS) to the user, then after entering the 6 character code and clicking the login button the user will be logged in.

User Story #3:

As a user, I would like to be able to secure my account with 2 factor SMS authentication.

#	Description	Estimated Time	Owner
1	User account page allows enabling of 2FA SMS	2 Hours	Logan
2	On login with correct login information, send user generated 6 character code.	2 Hours	Logan
3	Login the user if the code entered is the same as the code sent through SMS.	2 Hour	Logan

Completed:

SMS verification works by sending a code to a user's SMS and uses Twilio as described above in User Story #1. The SMS is sent to the phone number that is stored within the database. If the user selected the 2FA option when creating an account they will have to enter a 2FA code for every attempted login. On the login page they can click the Send 2FA Code button to send an SMS (or email) to the user, then after entering the 6 character code and clicking the login button the user will be logged in.

User Story #4:

As a developer, I would like to be able to edit user information within the database.

#	Description	Estimated Time	Owner
1	Allow tables to be changed in the database to edit user information.	3 Hour	Steven
2	User login should reflect new changes within the DB.	2 Hours	Steven
3	Non-developer users should be blocked from editing user information	2 Hours	Steven

Completed:

A button was added to the navbar which allows developers to access the phpmyadmin backend for the database, which provides full functionality to updating the database. The button does not appear for non-developer users, but even if accessed by a non-developer, login credentials are needed to modify the MySQL server.

User Story #5:

As a developer, I would like administrator accounts which give access to backend data.

#	Description	Estimated Time	Owner
1	Dev accounts are flagged in the DB with a dev boolean.	3 Hour	Logan, Steven
2	Developer UI screen can be accessed from the landing page only if the logged in user is flagged as dev.	3 Hours	Logan, Steven
3	Developer UI screen should display site statistics.	6 Hour	Logan, Steven

Completed:

The users SQL table now features a column that stores a boolean flagging dev status, and users with developer access are able to see additional navbar buttons which direct to pages that normal users should not have access to (for example, the database modification page)

User Story #7:

As a user, I would like to be able to adjust default measurements between Miles/Kilometers.

#	Description	Estimated Time	Owner
1	Create user personalization field in database for setting default distance unit	2 Hours	Steven
2	Parameterize fields to display value corresponding to user's default	3 Hours	Steven
3	Handle unit conversions between possible user input and known units of algorithm input	3 Hours	Steven

Completed:

On the main page there are inputs for distance, pace, and time. The displayed preview value of distance is "Distance (Kilometers)" and it is tied to the state of the main page. On click of the Change Units button the preview is changed to "Distance (miles)" also changing the state. This state can then be passed to our algorithm, allowing the algorithm to run the correct distance numbers based on the units displayed as a preview, or the units chosen.

User Story #8

As a user, I would like to be able to visualize the amount of calories I've burned.

#	Description	Estimated Time	Owner
1	Design UI of user statistics page	2 Hours	Austin
2	Implement UI of user statistics page	10 Hours	Austin
3	Create calories section of user statistics page	4 Hours	Austin

Completed:

There is a calories burned section of the statistics page that displays a line graph. On entering the statistics page the database is queried to find and display the date and calories burned of all previous routes for a given user. Currently there is no default for the line graph when there are no or only 1 routes found, as both of these values are not displayed in an appealing way on a line graph.

User Story #9

As a user, I would like to compare my pace with distances similar to those ran by famous or olympic runners.

#	Description	Estimated Time	Owner
1	Create comparison section of the user statistics UI	3 Hours	Austin
2	Store distance and time of popular runners	2 Hours	Austin
3	Display comparison between popular runners and the user	3 Hours	Austin

Completed:

There is a comparison section of the statistics page that displays a table. Directly above this table is a drop down menu of runners with world records. Choosing a runner and clicking the enter button will query the database to find previous routes run by the user within 10% of the distance run for the world record. If a route is found, it will be displayed next to the runners information for comparison. If not, N/A will be displayed on the table.

User Story #10

As a user, I would like to find a new route without needing an account.

#	Description	Estimated Time	Owner
1	Research and testing to get Typescript working in React project	6 Hours (split)	William, Austin
2	Integrate Google Maps Javascript API code with algorithm into the React Webpage	6 Hours (split)	William, Austin
3	Test and debug	3 Hours	William

Completed:

The Google Maps object and algorithm are fully integrated into the React website. Because integration with Typescript was more challenging, vanilla Javascript in addition to the Google Maps API was used instead in order to display the map. All features from the example are now implemented into the new webpage.

User Story #11

As a user, I would like to find multiple routes from the same starting point.

#	Description	Estimated Time	Owner
1	Research and create method to allow for the route generation algorithm to generate multiple routes	2 Hours	Luke
2	Implement changes to current route generation algorithm	10 Hours	Luke
3	Test and debug functionality	2 Hours	Luke

Mostly Completed:

Although we developed several possible methodologies for the generation of multiple routes at once, we were not able to get a fully working version implemented before the end of Sprint 2. However, this is a feature that will be quickly implemented during the course of Sprint 3.

User Story #12

As a user, I would like to use an address as a start point.

#	Description	Estimated Time	Owner
1	Allow user input of an address as a starting point for the route.	2 Hours	Luke
2	Implement the address as a starting point within the routing algorithm	6 Hours	Luke
3	Test and debug functionality	2 Hours	Luke

Completed:

Users are now able to input addresses, and the algorithm will take that address and convert it into a latitude/longitude object using the Google Maps Geocoder. That object can then be used to designate the starting and ending point of a route. Through this, a route can now be generated with a user-inputted address used as a starting point.

User Story #13

As a user, I would like to have autocomplete when typing an address.

#	Description	Estimated Time	Owner
1	Use Google Maps APIs to develop method to add autocomplete to address input	2 Hours	Luke
2	Implement method within the algorithm	4 Hours	Luke
3	Test and debug using unit tests	2 Hours	Luke

Completed:

When typing in the address text input area, the Google Maps Autocomplete will now display several addresses below the text field. Users can select one of these autocompleted addresses, and that address will then fill out the text field. Upon clicking “enter”, the algorithm will use that address as the starting point for the new route.

User Story #14

As a user, I would like to find a route using distance.

#	Description	Estimated Time	Owner
1	Link UI element for distance to Route Finding Algorithm with options for kilometers or miles	1 Hour	William
2	Modify current algorithm to be more accurate from inputted distance to route	4 Hours	William
3	Test and Debug	2 Hours	William

Completed:

UI element to take user inputted distance is linked to algorithm with a conversion function for both miles and kilometers so when the “Enter” button is pressed a route is displayed. Also the route finding algorithm is updated to be more accurate with user inputted distance.

User story #15

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

#	Description	Estimated Time	Owner
1	Research how to get altitude changes on a route	3 Hours	William
3	Create UI element that displays altitude change under details box	4 Hours	William

Partially Completed:

Preliminary research into how to get altitude changes on a route as well as the UI element to visualize the altitude changes on the current route were completed.

User story #16

I would like to find a route using time and a pace.

#	Description	Estimated Time	Owner
1	Link UI Element to Route Finding Algorithm	1 Hour	William
2	Create method to convert time and pace to distance before sending it to route algorithm	1 Hour	William
3	Test and Debug	1 Hour	William

Completed:

UI element for time and pace inputs are linked to the algorithm with a function that converts the given metrics into meters. Manual testing was successful in determining that the inputted time and pace produce a route within 400 meters of the calculated distance.

Sprint 1 Incomplete User Stories Completed in Sprint 2

User Story #6

As a user, I would like to be able to customize my nickname on my profile.

#	Description	Estimated Time	Owner
1	Create UI element for allowing users to change nickname	2 Hour	Logan, Austin
3	Create UI element for allowing users to change profile pictures	2 Hours	Logan, Austin

Completed:

A profile page was created to hold fields and buttons for users to view their current information as well as change it. Currently there is nickname and profile picture integration for viewing and changing on the profile picture. Change the value in the fields and clicking the enter button will directly change the values stored in the database.

Note: User stories 9 and 10 were combined because both used the same HTTP API that was not generating HTTPS requests viewable on the website.

User Stories #9 and #10

As a user, I would like for outside temperature to be displayed in the route planning screen.

As a user, I would like for outside weather conditions (rain, snow, etc) to be displayed in the route planning screen.

#	Description	Estimated Time	Owner
2	Implement the outside temperature into the panel	2 Hours	Austin

#	Description	Estimated Time	Owner
2	Implement the outside weather into the panel	2 Hours	Austin

Completed:

The requests using axios and a weather api were changed so that they used https requests instead of http requests. This change allowed our hosted website to use and view the weather api information returned and now is correctly displayed.

What did not go well?

We had problems with workload around week 5 because of midterms and the shell project in CS252. While more user stories were completed, less overall change was visible in the project due to the user stories being smaller. We also had too many user stories and problems unfinished by the Thursday before the sprint review, causing us to work for much too long right before the deadline to fully implement, merge, and deploy everything.

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

Not Completed:

Due to an injury in the group during Sprint 2, no progress was able to be made towards this task; however, because of the systems put in place by other tasks (i.e: the developer checking for the database page), this task should not be difficult to make up in Sprint 3.

User Story #11

As a user, I would like to find multiple routes from the same starting point.

#	Description	Estimated Time	Owner
2	Implement changes to current route generation algorithm	10 Hours	Luke

Not Completed:

Due to time constraints as the end of Sprint 2 approached, we were not able to implement a working version of the multi-route feature. We have researched and developed several possible methods to approach this problem, including using recursive calls to the Google Maps API and running the API through a short loop, and this should (hopefully) be implemented smoothly during Sprint 3.

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	4 Hours	William
4	Test and Debug	3 Hours	William

Not Completed:

Due to the amount of time spent finding an appropriate solution to integration and reworking part of the route finding algorithm, this user story was not able to be entirely implemented. Adding a method to differentiate large and small altitude changes proved to be more challenging due to the limited number of requests that can be sent to the Google Directions API. However the altitude graph and the foundation for implementing this feature are complete for next sprint.

How should you improve?

Although we completed less work this sprint than the previous, we still completed most of the user stories and are fully on track to create a finished project by the end of sprint 3. We will work to fully complete and flesh out the user stories implemented so far, as well as the new user stories for sprint 3, so that our finished product can be as robust as possible.

The first significant problem we faced was a failure to make the full number of meetings we agreed upon. Although our meetings generally run relatively short, failing to meet at agreed-upon times contributed to the tight time constraints we faced near the end of the sprint, due to delays from lack of consistent communication.

The second major problem that we faced was leaving too many problems unfinished too close to the deadline. While we may have gotten most of the work completed before the sprint review, our final product for sprint 2 could have been better if we managed the time better. Not having to cram work into a tight schedule on the last day would be much more beneficial for our schedules and final product.. Going forward we will set a hard deadline of completing all user stories and integration needed for the sprint review for Wednesday night, so that we can solely work on merging between the different branches on the Thursday before a sprint review.

Overall, we felt that there were no critical problems, other than that we struggled with completing the content and deadlines for Sprint 2. Going forward we will make sure to follow more rigorous deadlines in order to hold ourselves accountable and create a better, more balanced workload. Through this, we will ensure our final product is as polished as possible.