Dithi Saxena
Catherine Penquite
Abhishek Gunasekar
Christopher Yu
Vivek Nair
Team 6

Sprint 2 Retrospective

5th April 2021

Werk It

TABLE OF CONTENTS

What went well?

In general, we developed the basic functionality of our cross-platform application. The mobile app can now take user input for workouts/exercises and record it in our database. The web app can illustrate the recorded data into a simple histogram and display their weekly progress and streak counter. Lastly, we have implemented a backend route that communicates smoothly with both the web and mobile client simultaneously, thereby enforcing the cross-platform nature of our project.

User Story #2

As a user, I would like to have touch ID login once my credentials are saved.

Completed:

The mobile application had functional touch ID login for any phone that has hardware for touch ID functionality. Additionally, the UI for the login page included a touch ID icon that made it obvious for the user that the functionality is available.

User Story #3

As a user, I would like my device to remember me until the next time I log out on the mobile app.

Completed:

Once a user logged in at least once on the mobile application and selected the option to be remembered until the next time they logged out, the app successfully did not log out the user upon closing of the application. In the case the user did log out, then upon closing and reopening the application, the user was again prompted to log back in.

User Story #4

As a user, I would like to be able to input the number of sets I did for each type of lift.

Completed:

When creating a workout on the mobile application, the user was successfully able to input the number of sets completed for each type of lift, assuming they were creating a workout of the type “lifting”. Additionally, the application was expected to limit the number of possible sets to 99, and this functioned accordingly.

User Story #5

As a user, I would like to be able to input the weight I did per set of each type of lift.

Completed:

When creating a workout on the mobile application, the user was successfully able to input the weight lifted per set of each type of lift, assuming they were creating a workout of the type “lifting”. Additionally, the application was expected to limit the amount of possible weight to 999, and this functioned accordingly.

User Story #6

As a user, I would like to be able to set the duration of my run.

Completed:

When creating a workout on the mobile application, the user was successfully able to input the duration of a completed run, assuming they were creating a workout of the type “cardio”. Additionally, the application was expected to limit the maximum possible duration to 999, and this functioned accordingly.

User Story #7

As a user, I would like to be able to set the speed for my run.

Completed:

When creating a workout on the mobile application, the user was successfully able to input the speed of a completed run, assuming they were creating a workout of the type “cardio”. Additionally, the application was expected to limit the maximum possible speed to 99, and this functioned accordingly.

User Story #8

As a user, I would like to be able to set the number of laps for my swim.

Completed:

When creating a workout on the mobile application, the user was successfully able to input the number of laps completed for a swim, assuming they were creating a workout of the type “swimming”. Additionally, the application was expected to limit the maximum possible number of laps to 99, and this functioned accordingly.

User Story #9

As a user, I would like to be able to select individual types of exercises.

Completed:

The user is successfully able to select the type of exercises when on the workout editor screen. The user can add a new exercise when the “add exercise” button is tapped. In addition, if the user does not select an exercise type, the mobile client will give an error message to the user.

User Story #10

As a user, I would like to be able to create a new type of exercise if it does not already exist.

Completed:

The user is able to create new exercises when on the workout editor screen. When they go to add a new exercise, if the exercise they want to add does not exist in the available options, they can opt to create a new one. As long as their new exercise does not have the same name as a preexisting exercise, the new exercise will be created, saved to the database, and accessible for future use as well.

User Story #11

As a user, I would like to be able to select the type of workout I plan to do (lifting, running, swimming, etc.).

Completed:

The user is able to select the type of workout they wish to create when on the workout editor screen. When the user goes to create a new workout, they will see a dropdown menu that says “Select a workout type”. Upon selecting a workout type, then when the user tries to add an exercise to the workout, they are presented with a selection of exercises that are filtered to match that workout type.

User Story #12

As a user, I would like to be able to set a custom type of workout.

Completed:

The user is able to create new workout types when on the workout editor screen. When they go to select the workout type, if the workout type they want does not exist in the available options, they can opt to create a new one. As long as their new workout type does not have the same name as a preexisting workout type, the new workout type will be created, saved to the database, and accessible for future use as well.

User Story #13

As a user, I would like to be able to set the exercises for my custom workout.

Completed:

The user is able to select the exercises they want to include in their custom workout when on the workout editor screen. As long as the user has specified a workout type, they can choose from any of the exercises that appear in the filtered list to add to their workout. Once the user is done adding exercises, when they submit their workout, then the workout and the exercises selected will be saved to the database.

User Story #14

As a user, I would like to be able to create my own workout plans.

Completed:

The user is able to create their own workout plans when on the workout plan editor screen. So long as the user has previously created workouts and saved them, then when they go to create their workout plan, they will be able to choose from those workouts and assign them to days of the week. Once the user is done, when they submit their workout plan, then the workout plan will be saved to the database.

User Story #17

As a user, I would like to see my workout time per week on a histogram.

Completed:

The *Werk It* dashboard page has a histogram that aesthetically illustrates the weekly workout time for the given user. The histogram uses the Google Charts API that dynamically updates the chart when a user completes a given workout within a week from the mobile client. Additionally, just like the other components on the web dashboard page, the histogram also adapts to the dark mode view of the web app.

User Story #18

As a user, I would like to be able to stay motivated through a workout streak counter.

Completed:

The workout streak counter keeps track of the number of consecutive weeks for which the user's workout goals have been met. As such, when the weekly goal progress bar is completely filled within the WerkIt dashboard page, the workout streak counter number field increases by 1 for the given user. Towards the end of the week on Sunday, when the weekly progress bar is not filled, then the workout streak counter is aptly reset to 0, alerting the user to workout more frequently in upcoming weeks.

User Story #19

As a user, I would like to see a progress bar at the start of the week indicating how much of the weekly goal is accomplished.

Completed:

The weekly goal progress bar present in the *Werk It* dashboard page acts as a visual tool to help the user measure his or her workout progress for any given week. Consequently, when the user completes a workout, the progress bar is increased by 1 unit and does not increase when the user fails to complete a workout. More importantly, when the progress bar is full, it is determined that the user has fulfilled the weekly goal and the web app congratulates the user with a warm message.

User Story #20

As a user, I would like to be able to view the web app in dark mode.

Completed:

Just like many other modern apps, the *Werk It* web app also provides an optional dark mode functionality. When a user visits the profile page, they can change their defaults preferences for the dark mode, and this change will be reflected in the rest of the pages of the web app. So, the next time the user has logged in to the *Werk It* web app, the change of dark mode will persist within the app.

What did not go well?

In general, we were unable to implement features such as the profile picture and loading symbol/network connectivity for the mobile and web app. While this did not create an obstacle for basic functionality of the apps, it is important we incorporate these in the next sprint as they are essential features most exercise applications include and features that users expect.

User Story #15

As a developer, I need to display a loading symbol if requests take longer than a second so that the user does not think the app has frozen.

1	Design loading symbol UI for web app if requests to the server take longer than one second.	3 hrs	Vivek

2	Design loading symbol UI for mobile app if requests to the server take longer than one second.	3 hrs	Vivek

3	Debug and test	1 hr	Vivek

Not Completed:

While the web application had a functional loading symbol, the mobile app failed to display the loading symbol. In addition to this, the loading symbol on the web was displayed on the bottom left corner rather than in the center of the screen. Lastly, because the user story was not fully completed in time for the sprint review preparation (the night before the sprint review), thorough debugging and testing was also not able to happen.

User Story #16

As a user, I would like to be able to set my profile picture.

1	Add profile pic feature on the web app.	6 hrs	Vivek

2	Add profile pic feature on the mobile app.	6 hrs	Vivek

3	Debug and test	1 hr	Vivek

Not completed:

The web and mobile app were not able to display the profile picture feature. Both apps did not allow the user to set their profile picture and both apps did not display the profile picture on the top of the profile page/screen. Lastly, because the user story was not fully completed in time for the sprint review preparation (the night before the sprint review), thorough debugging and testing was also not able to happen.

How should you improve?

There are many lessons our team learned from this second sprint. First, because this sprint was heavy on testing with user data, we should have planned to prepare for the sprint more than one day in advance of the sprint review. When we were preparing the night before our sprint review, we ran into many errors while testing the acceptance criteria because we did not have enough data prepared for testing in the database. Because of this, we spent an hour trying to prepare and debug our code when we should have been reviewing for our sprint presentation. This could have been prevented if we moved our hard deadline two days before the sprint review and debug one night and test the acceptance criteria the following night.

Additionally, our team still needs to improve the level of communication among all the members within the group. Although we seem to be communicating well and promptly address issues at the start of the sprint, this does not seem to be the case during the rest of the sprint. The tasks for sprint 2 required more collaboration than the previous sprint, and sometimes it was difficult to reach members when someone needed help. As mentioned previously above, pushing new code onto the main branch on the night before the sprint review was not a wise thing to do, and could have been easily avoided if there was proper communication throughout the entire sprint.

Moreover, our team needs to put priority on implementing features that involve minimum to no interfacing with hardware. Use of mobile features such as the camera and file browser involve permissions, which can vary in terms of requirements due to device, operating system (e.g. Android/iOS), and OS version. These often cannot be implemented in “pure javascript” and often need modifications to AndroidManifest.xml or Pod installation (iOS). In some cases, the MainActivity.java file would have to be modified, adding to the complexity of implementation. Therefore, such “native” features can take significant amounts of time to implement, and should be done after the other “non-native” features are implemented. In relation to the previous paragraph, any extra packages/dependencies installed should be told to the rest of the team so that changes can be checked for possible regression in other functionality early on and get resolved, allowing for faster and easier development.

Furthermore, just as it is necessary to communicate with each other, it is also equally important to respect each others’ time. As mentioned earlier, there were times during this sprint when we had to spend time during our meetings debugging code instead of completing the actual objective for that meeting. With proper communication, situations like these can be avoided so that meetings can accomplish their intended purpose in a timely manner. In addition, a certain element of individualism should be enforced. While the team exists to help one another, one should always spend time trying to figure out their own bugs before turning towards others for help. Tying this back to respecting each others' time, every team member has their own bugs in their code that they have to deal with, and it is unfair to become reliant on others to fix your own bugs. Going forward into Sprint 3, we should aim to communicate with each other when we run into issues in a timely manner, but try to resolve them on our own first.

Lastly, something that will help our team achieve the earlier hard deadline, as mentioned earlier, is starting to work on our sprint tasks with more urgency during the first week. The first week of Sprint 2, our team in general took it easy on the amount of work getting done, due to having just finished the Sprint 1 review, planning for Sprint 2, and putting more priority on other classes. However, this led to having a lot more stress to finish during weeks 2 and 3. Our third week can be much smoother with an earlier hard deadline if we prioritize a strong start on our work during week 1. This should include early communication to connect different parts of the project, instead of only doing the individual parts during week 1.