|  |  |  |
| --- | --- | --- |
| **SCIT**  **School of Computing and Information Technology**  **Faculty of Engineering & Information Sciences** | | |
| **CSIT321 - Project** | |  |
| **Project Topic:** | Carpool World | |
| **Topic Code:** | FYP20-S1-10 | |
|  | |  |
| Individual Work Diary          Name: Mohamed Shafiq Bin Roslan  UOW Student Number: 6098253  E-mail: [iam.shafiq@live.com](mailto:iam.shafiq@live.com) | | |

# Week 1

Timeline (11/01/2020 - 17/01/2020)

## Tasks done:

#### Research into taxi, carpool and car-sharing companies & applications

We were each tasked to research into companies that were doing things similar to our projects. While researching we found GrabHitch, SGHitch, Ryde, Share Transport and Sameride Commute. I was assigned to research on Sameride Commute and found that Sameride is doing something very closely related to our project, in that they are also a carpool app.

## Personal Reflection:

Honestly, it is not much different from other car-hailing applications like Grab or Uber, the only difference being that the rides are usually free, but drivers are allowed to set their own prices. I feel this could be abused by the drivers and I plan to set a fixed price for the rides in our application since it is not supposed to be profit driven for the drivers, but instead a cost-effective way to get to SIM.

# Week 2

Timeline (18/01/2020 - 24/01/2020)

## Tasks done:

#### App name and team structure

For this week, we were to come up a name for our project. We eventually chose to use ‘SIMRide’ because from the name itself it is obvious enough to users that it is an application that has to do with transport, catered to staff and students of SIM. We were to also break into roles and I was assigned to be the Chief Programmer of the team.

#### Project scope

We have also been researching more into which OS platform, database, hosting server, language and software architecture we are to use. For myself, I researched into which database would be best suited for our project. Vinny had recommended looking into Firebase as it had a database that can be used. While researching into it, I found that they also had Firebase hosting, Google Cloud functions and Google Authentication tied together. Since we are doing a car-sharing project and we are going to use a map, and that Google has one of the best maps available, I decided to stick with Firebase as I feel integrating each Google API with each other would be fairly easy.

On top of that, our group also discussed the programming language that everyone is comfortable with and which OS we were going to use. Vinny again recommended using React Native for Expo, which will be written in JavaScript. She recommended it as she has been using it in her internship and that she has experience with it. She also mentioned that it can easily be used for all platforms, whether web apps, iOS or Android and you do not have to write specifically for each.

We have also decided to use Agile Methods as our project development methodology as it is the one we are most familiar with, having used it in a previous module, and it is also the one that makes the most sense to us as it is flexible and safer, as each function is fully tested before moving on to the next one.

#### Features for SIMRide

We also discussed features for the project and the functions seem to be straight-forward – the toughest one right now would be the automatic route planning – but even then, this would be handled by the Google Maps API, so it should not be that big of a problem.

#### Risk assessment

We also discussed the risks that comes with the project and I feel the risks are accurate in that it might really happen in the coming weeks; especially the delays that comes with the learning curve due to us having to learn to use React Native and the lack of communication between us as for most of us, this is the first time we are seeing each other.

## Personal Reflection:

What I hope will happen is that even though I am the Chief Programmer, the rest of the team will still help to with the coding, testing and debugging of the application as to be fair, this is still a group project.

We have also decided to use GitHub as our version control for our project. It will be my first time using it and I am actually very happy to learn to use it as I have seen many companies using this as their main version control.

# Week 3

Timeline (25/01/2020 - 31/01/2020)

## Tasks done:

#### First foray into React Native & Firebase

Being the chief programmer, I decided to check out React Native as we are going to use that for our project. It is basically built on JavaScript and is similar to web development but as I am not familiar with JavaScript, having only used it in the web development class a year ago, I have to relearn it.

With the little knowledge that I have learned, I moved on to creating a log in and log out, sign up and the first parts of the messaging functions. As Firebase has Google Authentication tied in, it was easy for me to create the sign up, logging in and out as Google will check for the email, and store it in their own database. All I have to do is store the user details to allow them to update their details later on.

I have also tried getting firebase to work with our project with success. The Firebase database is also not the usual database that we have learned in school as it is a NoSQL database and the way items are stored and accessed are much like JSON objects. I also tried to host the website, but to no avail. The website is hosted, but does not show the dummy pages that I have created. I will probably look into this in the coming weeks.

#### User manual

Together with Seng Yang, I also made the first iteration of the user manual with the functions that I have already made.

## Personal Reflection:

As 4 out of 5 of us are not familiar with React Native, NoSQL and not comfortable with JavaScript, I hope this does not deter the team to learn it as we need as many hands as possible to get everything running smooth.

# Week 4

Timeline (01/02/2020 - 07/02/2020)

## Task done:

#### More Firebase

This time I tried to create a log in with the authentication from Google and store the data into Firebase’s Real Time Database. It was very easy as Google had all the documentation up online. I then set up the log out functions as well since it was just an extra line of code to safely log a user out.

## Personal Reflection:

With the documentation being online, I feel it would be easy for me and my team mates to follow and we can probably do all the functionalities fast and manage to go through iterations of testing and debugging and eventually come out with a product that we are all happy with.

# Week 5

Timeline (08/02/2020 - 14/02/2020)

## Tasks done:

#### Programming – update account, view other profiles

I continued on with programming and fine-tuned the log in and log out functions. I also added a function that allows users to update their account details and update their passwords. Passwords are not stored in the Firebase Database but instead it is kept with Google Authentication. I figured this was a better idea as Google would have better security and encryption than the ones that we are able to do. I also added a function for users to search for other users and view their profile.

#### Set up GitHub for the team

As Vinny was taking a while to set up the GitHub page, I decided to set the page up myself and did a simple ReadMe file so that the rest of the team would be able to access the project and I also listed the functions that we needed to work on in the ReadMe so that we have an idea of what needs to be done.

## Personal Reflection:

As the functions are not big yet, it is still quite easy to do the project and get everything running. But with the inexperience of using React Native, I am unsure if what I am doing is efficient and correct and if it will help with the front-end that Vinny is going to do.

After Vinny finally updated the GitHub ReadMe with proper instructions on how to access the files and I am hoping the rest would take the chance to try using and learning to use React Native.

# Week 6

Timeline (15/02/2020 - 21/02/2020)

## Tasks done:

#### Design application architecture

This week we are to complete the documentation for the first iteration and I have been tasked to do the design application architecture and database design diagram. I had to research on the application architecture as I was not sure how to do it and found that it was describing the parts of the application much like describing how a house is built. As nothing was set in stone, I gave a few different ways of how we might create the application and which APIs we might use and described them in detail.

#### Design Database Diagram

I designed the database diagram of how we currently see each item is stored and how is it used as I understand throughout the project there will be changes made as ideas change and designs improve.

#### Physical architecture documentation

For the physical architecture, it was initially tasked to Seng Yang, but as he was unsure of the components in the architecture, I had to step in and help with the research and writing in the documentation to be able to hand it in, in time to our supervisor.

#### Programming – update password, navigation across pages

Again, with Google’s authentication, changing passwords is easy due to Google updating and storing the password on their end. All I have to do is set up the validation for the password. I also set up dummy pages for me to test the navigation in React Native. The way pages are navigated is very different from using HTML or .NET pages in that it doesn’t use Response.Redirect(“xxx.aspx”); but instead it uses <Routes> and other form of components to navigate.

## Personal Reflection:

I feel that Seng Yang could have done the research himself if he put in the required effort to, but as this is a team project, I did not mind helping him out, as looking at the bigger picture, it is just me helping myself for the project.

The differences in React Native and HTML pages are beginning to slowly show as I progress through the project, and with every function I feel like I have to learn something new regarding to React Native to solve a problem that I can easily do in HTML.

# Week 7 - Exam Study Week

Timeline (22/02/2020 - 28/02/2020)

## Task done:

Studied for exams

## Personal Reflection:

Will continue FYP after the exams are over.

# Week 8 - Exam Week

Timeline (29/02/2020 - 06/03/2020)

## Task done:

Studied for exam.

## Personal Reflection:

Will continue FYP after the exams are over.

# Week 9

Timeline (07/03/2020 - 13/03/2020)

## Tasks done:

#### Programming – Storing of chats, retrieving chats

Firebase has 2 databases, the other being Firestore. I decided to use Firestore for the messages features as Google mentioned that Firestore has the capabilities to expand easily and there will be more messages than there will be registered users.

For the chats, I had to do a research on how the one-to-one chat functionalities were designed by other users using Firebase. I came across Firechat and used it as a reference on designing my own chat interface. The chat can be used live as I have tested with 2 different browsers and 2 different accounts, and they will be updated whenever a user enters a message.

## Personal Reflection:

With each function being implemented, I feel more confident with using React Native but I do wish that there was somebody that could tell me if I was using it wrong, or could have done what I did in another, better way.

# Week 10

Timeline (14/03/2020 - 20/03/2020)

## Tasks done:

#### Programming – create a booking

For this week I have been tackling creating a booking. Creating it is easy enough; all I have to do is store the information entered by the user. While creating the functions, I found that it wasn’t the case as I had to plan for when users has a booking on the same date and time, or when the user has a booking 2 hours after, or 2 hours before the current booking. I also have to check for the amount of money they have in their wallet, if they choose to pay by E-Wallet, which I will implement after I start the E-Wallet functionalities.

#### Physical architecture & application architecture elaboration

As told by our supervisor, our documentation on physical architecture and application architecture is not enough and he requires that we explain in detail for each part. As I had been doing the documentation for both of them, I had to elaborate further for the two of them.

## Personal Reflection:

I find I should take a step back and plan for each function before starting to code because I could be missing out on a lot of things if I just go headfirst without planning. For instance, I could be missing a very obvious validation and now I have to redesign the function due to the missing validation.

Three of my team mates have mentioned that they are not used to the way the project is coded and they still do not understand how to access the files on GitHub. Vinny has updated the ReadMe file to allow them to understand it better and I offered to help them in any way I can, and that they can message me personally if they have any questions on how to use React Native or any questions regarding the project.

# Week 11 - Submission & First Presentation on 28/03/2020 on Moodle

Timeline (21/03/2020 - 27/03/2020)

## Tasks done:

#### Programming – apply to be driver, upload license image, join booking

This week I worked on the functions to submit the application to be a driver on the app. Luckily for me, Google has a storage for images, and I was able to push the picture of the licenses there. So all I had to do was store the data that the users input, do a validation check, and check if they are entering the correct data, that is, they have at least 2 years’ worth of driving experience, entered a valid NRIC number and the pictures they uploaded are in the correct format. I also had to make sure that they do not press cancel while the images are being uploaded and that the database updates checks if they have successfully completed the registration or if they cancelled the registration process halfway. This is to ensure that there are no duplicated submission forms by each user in the database.

To join a booking, there is also a validation check to see if the user has any bookings on the same date and time, or on the same date but within 2 hours of the desired booking. This is to ensure that they do not accidentally overbook themselves.

## Personal Reflection:

As the programming phase has started, Vinny is currently doing the front-end and Seng Yang has volunteered to do the Google Maps API. If Seng Yang is able to do his parts fast, then we will be able to complete the back end and do testing, debugging and improve on our design. Once Vinny has finished her parts, the new are able to combine the front and back end together to see the final results.

# Week 12

Timeline (28/03/2020 - 03/04/2020)

## Tasks done:

#### Programming – cancel confirmed ride

Passengers are able to cancel their booking if they need to. No penalty for cancelling at the last minute, but I am planning to allow reporting of such behaviour. The messaging system was created to allow users to talk to each other, so I am hoping if they were to cancel at the last minute, they would inform the driver. In the case that they do not, the driver is able to report them for no show. Admin is then notified of the report and will be able to investigate the matter and even ban the user if necessary.

## Personal Reflection:

After the presentation to Mr Tian, he has told us that our current progression is very slow, and that he hopes that we are able to finish our product by the 23rd of May. I will continue to do my part of the project and will hurry my functions along and I hope the rest of my team mates hears that and also do their part to hurry the project along.

Currently I am waiting for Seng Yang to finish his Google Maps API and for Vinny to finish her front-end.

# Week 13

Timeline (04/04/2020 - 10/04/2020)

## Tasks done:

#### Programming – view created rides, e-wallet page, reset password email, review driver application

For viewing created rides, I had to store the booking ID in the button as its ID, and then call the events function and get the booking ID to pass on to a function to show the requested details of said booking.

For the E-Wallet page, I made a blank canvas to be used later when creating the functions for the e-wallet page.

When resetting the password, I was able to use Google Authentication and it will help send the email to the user who requested it, along with a specified body of text of my choosing.

When logged in as admin, the admin is able to view the applicant who applied to be a driver and the details he has entered. The admin then is able to approve or reject the application and the database will update accordingly.

## Personal Reflection:

The app is functioning better now, as users are able to create and join a booking/ride. All that is left is the payment method and for the maps to be fully functioning to allow the drivers to seek direction. It is currently the third week waiting for Seng Yang to finish his **ONE** function.

Also waiting for Vinny to finish her front-end.

Update 10th April 2020: Seng Yang has finally finished his Google Maps API after 3 weeks. But the Map doesn’t do anything but show the current location of the user. He is currently moving on to setting the directions for the user. I hope this doesn’t take another 3 weeks.

# Week 14

Timeline (11/04/2020 - 17/04/2020)

## Tasks done:

#### Programming – split users into roles and define what each role can see, cancel created booking, ban user, report user, dashboard, choose to pay by cash or wallet when joining booking, set up wallet for all users

Drivers are allowed to delete their own created ride if they deem necessary. As users have not paid for the ride yet, they do not have to be refunded. Users are also notified of the cancellation through their dashboard (which I plan to add later on). Same for the drivers, the riders are able to report them and the admin will be able to investigate the matter.

As there are currently three roles in the app, admin, driver and passenger, each role can view different things on the website. This is to stop the overload of information and also to prevent users from changing things that they are not supposed to.

The dashboard page is also updated to show notifications, reported users and upcoming bookings.

Now, when joining a booking, users are able to choose which payment mode they are paying by. Drivers will be informed of the payment method chosen by the passenger.

E-Wallet is also set up for all users when they sign up as customers.

## Personal Reflection:

Things are getting messier as I scale up the project, and I now see the reason why commenting the code and letting people know how each function works is very important for other users, and even for yourself as you may have forgotten over time how it works.

As the group has voiced out that they aren’t very familiar with React Native, I have sent them a free online course from Harvard that teaches people to use React Native. With COVID-19 forcing people to stay home, they have no reason not to try the online course, especially since it is free. I am trying to help my team mates in every way I can, to again, help myself, but they are not doing their part to make this work. I really hope things improve.

Waiting for Seng Yang to finish getting the directions for each user. He says that he cannot get the postal code, from each passenger. I have shown him the function to get the postal code from the stored passengers in the booking. I am currently in doubt if he even tried the database functions.

Also waiting for Vinny to show something from her front-end. It’s been a month.

# Week 15

Timeline (18/04/2020 - 24/04/2020)

## Task done:

#### Programming – find a datetime picker, filter by area

To make it easier for users to pick a date and time, I searched for an easy to use date time picker. I also created a function where users are able to filter the bookings by a specified area

#### Programming – passengers enter pick-up point, checks if passenger has a booking around the same time, and checks for wallet balance when joining a booking

When joining a booking now, passengers are to enter a pick-up/drop-off point by entering the address in the specified textbox. It is a Google auto-complete textbox, as such it will search the database for an actual address matching the description. The app will then store the address and longitude and latitude of the entered address.

The app will also check if the passenger has a booking around the same time and will check for the wallet balance of the user before allowing them to join the booking, if they choose to pay by e-wallet.

#### Programming – when creating a ride add information to say going to school or going home, drivers able to remove passenger from booking, when creating a booking, app now checks if driver has a booking around the same time

When drivers create a booking, drivers are to specify if they are going towards or away from school, as it will be easier for the app to manage the directions. Drivers are also allowed to remove passengers if they want to, but they have to specify a reason. The passenger will then be informed of the reason why they are removed. Again, if passengers feel they have been wronged, they are allowed to report the driver.

#### Programming – app notifies user if balance is below $5, transaction history page, cash out for users

The app will also notify the user if they have a balance below $5.00, as that is the minimum amount for each ride.

A transaction history page is also made to be used later to view all transactions made. Users are also allowed to make a cash out if they wish to, and they amount will be deducted from the user’s wallet and the admin will be notified of how much they request to be returned. Admin will have to transfer the amount manually on their own.

## Personal Reflection:

I have been trying to polish up all the functions and add on any logic that I may have missed out when designing them the first time, and I am also making sure to make checks so that the users will not key in any wrong data that will be detrimental to the functioning of the app.

Seng Yang has somewhat finished the directions portion of Google Maps. Currently it only can set directions to points that he has hard coded. On top of that, the directions shown will not be on the map, instead it will launch the Google Maps website (when running the web app on a browser) and show the directions there instead. This was not such an issue to me, as long as it still worked. Next, I have told him to try getting the information from the database, and he has said he’s been trying to. As the submission date is nearing, I will finish up the remaining functions that I have and then take over Seng Yang’s work if he still doesn’t solve it by then.

Vinny has shown the first few features of the app, and it is only the login and dashboard page after 5 weeks. As mentioned earlier, the submission date is nearing and we still seem to be so far away from completion.

# Week 16

Timeline (25/04/2020 - 01/05/2020)

## Tasks done:

#### Refolder files, deploy to firebase hosting, solve express.js issues, get Stripe API working

For this week I was trying to get the Stripe API functionalities working. After getting the API sorted in the app, when trying to make a test payment, the API returns a message that it cannot connect to the server. To me, this boils down to two things, that there is an issue with the way I implemented the express.js in the codes or it needs a proper server for the test payment to go through. I asked Vinny about this and she recommended that I refolder all the files and sort them out, and then try deploying to Firebase hosting again.

After refoldering, there were a lot of bugs that I have to solve due to the fact that now each function is on a different file, and I have to amend them and think of a different method to call them if they have a state function that they require. Over time I realised that refoldering the functions into files will not solve the Firebase hosting issue but when on to do it anyway because I was already halfway done.

I researched on why I could not host the app and it turns out it was because I did not build the web app before hosting. After doing so, the web app hosted without a problem, but the API call was still not going through, which means the problems was with how I implemented the express.js code

## Personal Reflection:

Another lesson in researching before implementing as this was a long, painful process that could have been avoided if I made the proper research.

Vinny has promised to finish the front-end within the next week.

Seng Yang has not finished his part on Google Maps Directions and thus I have to take over his part next week when I have finished with the Stripe API.

Currently it feels like I am doing the programming alone with my other three members doing the documentation and front end. I really don’t know what Seng Yang has been doing for the past 2 months as we could have really moved forward with the project if he did his parts.

# Week 17

Timeline (02/05/2020 - 08/05/2020)

## Tasks done:

#### Get Stripe API working

After further research and more debugging and testing, I found out that I have to use Firebase’s Cloud Functions to get the API to connect as Firebase is actually a serverless host. Redesigning the API call again, I then was able to get Stripe working and finally, got the payment system to work.

#### Programming – create recurring rides, top-up e-wallet, show transaction history

As requested by our supervisor, he wants to be able to create a weekly ride, as such I have created a function where the driver is able to create weekly rides for an *x* number of weeks that he chooses, and it will create the same booking, for the same time, and area but one week apart from each other. After getting the Stripe API to work, I can now add the amount paid to Stripe to be added to the user’s wallet. The transaction history page will now show the past transactions made through Stripe.

#### Programming – get all passengers pick-up/drop-off points, get directions to and from school, notify passengers that driver has arrived at pick-up point, remove payment amount from passenger wallet, add to driver wallet, driver able to update if passenger has boarded the car or did not show up, driver able to mark booking as completed once he reaches the last destination

As Seng Yang took too long with this function, almost a month with no proper progress, I decided to take over and get it over and done with. Over a period of 3 days, I finished up his functions.

Drivers are now able to get all the passengers pick-up/drop-off points when starting a ride, the app will check if the ride is currently towards, or away from school and will show each information accordingly. When driver has reached the meeting point, he is able to a notification to the passenger by clicking on the *I’m Here* button. When passenger has boarded the car, the transaction is made and the passenger’s e-wallet will be deducted and the amount will be added to the driver’s e-wallet. If passenger did not show up, a report will be sent to the admin. Driver will be able to mark the booking as completed once he reaches the last destination.

## Personal Reflection:

Maybe we should have picked a language and a framework that everyone was comfortable building in. Due to more than half of the group not able to code in the current environment made it harder for the other half to accomplish what they set to do. But it could also be said that they did not put in the same effort required as I knew as much as them about React Native and JavaScript when the project started.

Vinny has mentioned problems in the routing of the front-end and has promised to finish it up soon.

# Week 18

Timeline (09/05/2020 - 15/05/2020)

## Tasks done:

#### Debugging and Polishing Up

As I have to wait Vinny to finish her front end or at least build the app before I can try the push notifications, I have started to debug and polish up any code that I feel could be improved on or if any it is not logically sound. The testers seem to have found everything in working condition, which is pretty suspicious because as much as I can assure anyone that I have tried my best to get everything done correctly, I still was expecting bugs and errors to appear during tests. Nonetheless, I made the effort to test the code I have written to see if there were any problems.

Two full days of testing and I have found some bugs and logical errors to fix. Firstly, was the arrival and departure buttons, they did not work properly for when the car is being driven towards school, secondly there were logical errors when deducting payment after the ride for all passengers.

While debugging I found better ideas to implement for the notifications in case Vinny doesn’t finish her front end in time for me to implement the push notification. To cover up for the lack of outside-the-app updates, I made sure that the notification table was “live” in a way that if a user is on the dashboard page, the table will always update with latest items. For example, if the driver has arrived and clicked on the arrive button, the notification table will pop-up to notify that the driver has arrived as soon as the driver hits the button. Since I found a way to implement this, I decided to follow up with the bookings table as well and update them as soon as somebody joins a booking so that users will always be updated of the current number of passengers in a booking.

#### Documentation

As I have more time to do the documentation while waiting for Vinny, I helped out the documentation team with the Technical Report, the final presentation PowerPoint Slides and the Sprint documents. I wish they were done better because there were so many things lacking from each and I have to restructure and/or supplement more things into each document.

## Personal Reflection:

I feel that communication is key here. As leader of the group, I expected more from Vinny in terms of communication and updating the group of the projects progress, things we need to do and also getting everybody on the same page. Currently, she disappears for days without any updates before I have to check in with her and ask if she needs help, which she will say no to, and then take days on end to finish her parts. We have one week left, and I am starting to feel a bit panicky because we don’t have an end product as of yet and neither do we have the time to fix anything else if there’s a mishap.

# Week 19 - Final Submission on 23/05/2020 on Moodle

Timeline (16/05/2020 - 22/05/2020)

## Tasks done:

#### Documentation

I have updated the technical reports once more, also did the burn down charts for each sprint and helped out with the design and content of the marketing website done by Rachel.

#### Final checking

**22nd May 1pm (32 hours before submission)**

Project pushed into Vinny’s branch only has the front end, with only the bottom navigation working, I have checked with her as to what time she can finish the whole thing, she said it would be by 3pm. I checked through what she has done, none of the functions are implemented, worse still I don’t see any way to display the notification through the app which was a way to negate doing the push notification as we really don’t have time for it now. I will check to see if she will be implementing it later. I also gave her my opinions on the design and mentioned that it was good, but we could remove 2 of the buttons (messaging and settings) because we already have it as the bottom navigation, and it would remove some clutter. She did not seem to take it well and I suspect she doesn’t want to remove it, giving reasons that it was “okay, because it’s just a stack”, which doesn’t solve the problem of clutter at all.

22nd may 11pm (22 Hours before submission)

Turns out Vinny did not do as much as I hope for. She didn’t incorporate any of the functions while designing the front end, thus the functions are totally abandoned and we have to slowly figure out what to put where. This is very frustrating as we could’ve avoided this if we were all on the same page.

23rd May 7am (14 hours before submission)

Things are still looking very bleak. We are still so far away from the finish point and it kind of looks like we might not be able to pass the module at all. I have helped Vinny put the functions together with her front end, which isn’t completed either.

23rd May 4pm (5 hours before submission)

I have enough of this, we aren’t going to finish the project if we continue like this. I proposed to use my branch, with the complete functions that are working, and instead do the front end from there. I have managed to ask the whole group for help with the front-end as I believe they are capable enough to do it even though they have been complaining that they are not sure of how to code.

## Personal Reflection:

We finally came through as a group in the last 5 hours of the project, working in tandem and communicating as we should have from that start. The project did not look like what we wanted it to when we started the project, but nonetheless we each learned many important points that we can bring into our working lives pretty soon. For myself, I realised that you don’t have to be appointed the leader to take action and lead the group. I learned that the hard way as I thought enough was enough, and I wanted to bring myself and everyone out of the hole that we’ve been digging ourselves in.