finappster

sigma Σ

Chris Stehlin

John Sangalang

Jose Santos

Pete Scandle

MID-PROJECT REVIEW

Version History

Version	Details	Author	Date
One	Initial Draft	*finappster Sigma*	16/10/2021
Two	Final	*finappster Sigma*	18/10/2021

Key Stakeholders

Name	Student ID	Email
Chris Stehlin	1120279	prq5006@autuni.ac.nz
John Isaiah Sangalang	18017056	jsj6212@autuni.ac.nz
Jose Santos	17993442	sff4900@autuni.ac.nz
Peter Scandle	15922445	vzp7444@autuni.ac.nz
Barry Dowdeswell	N/A	barry.dowdeswell@aut.ac.nz
Leeanna Kohn-Hardy	N/A	leeanna.kohn-hardy@finappster.co.nz

Executive Summary

The Mid-Project Review will serve to provide an update on the progress that *finappster Sigma* has made through the first semester of their project. The team first began by working with the client to upskill on the FinTech industry and create the fundamental design principles. The main issue that was faced was the misunderstanding that was faced with the Project Proposal, but this has been overcome and the team is now on track to meet all upcoming milestones. A comprehensive list of user stories has also been created, encompassing the entirety of *finappster Sigma*'s scope.

Project Description

The goal of *finappster Sigma*'s project is to design and prototype the next feature component for the *finappster* application. This will involve creating a quiz to determine users' ethical views, displaying the results of this quiz in an easily digestible format on the user's 'My Values' page and creating a system to suggest future investments that are the closest match to the user's values. Our next key objective is to implement the suggestion system along with dummy investments to test it.

Overview

Objectives

To design and begin prototyping a core missing piece of *finappster,* giving end user investors the ability to align their financial investments with their personal values. The design process will come first, documenting, and illustrating the user journey of investors, investment providers and admin support users. Prototypes for the investor user that were mapped out in the design phase will be implemented by *finappster* sigma. This will serve as a foundation for future teams to build upon.

Scope

To plan out the potential implementation of *finappster's* three major users and the functionality required to achieve this, with one of these users being the development focus of *finappster sigma,* and the other two designed for use by future *finappster teams*. These three users being the investor, investment provider, and the admin support user, with the investor's expected user journey being the development scope of *finappster sigma*, meaning the development of feature wherein the investor discover where their ethical and sustainable values stand, and using this information to compare how their existing investments stack up against these results, and to get suggestions on new investments based off these results.

Approach

For this project, a mixture of the Scrum and Kanban (Scrum-ban) methodologies will be followed, utilizing two to three-week sprints. Testing is a key part of our approach, and it will initially be performed by the development team. Once they are satisfied the client will test before approving any features.

Milestones / Deliverables

The key deliverables required for a successful completion of this project are:

1. The prototype functionality is completed on time
2. Users will be able to complete the long quiz to determine their ethical beliefs
3. Once a user submits their quiz, they will be able to see investment options that align with their ethical beliefs

Activity	Completion Date	Status
Project Kick-Off	12th July 2021	Done
Project Proposal Draft Compete	15th August 2021	Done
User Stories Defined	9th September 2021	Done
Planning Poker	10th September 2021	Done
Final Project Proposal Complete	17th September 2021	Done
Sprint 1 Development Complete	11th October 2021	Done
Mid Project Review	20th October 2021	On Track
Sprint 2 Development Complete	29th November 2021	On Track
Sprint 3 Development complete	10th December 2021	On Track
Sprint 4 Development complete	28th Jan 2022	On Track
Sprint 5 Development complete	11st Feb 2022	On Track
Sprint 6 Development & complete	25th Feb 2022	On Track
Final Project Poster Complete*	17th June 2022	On Track
Final Project Presentation*	17th June 2022	On Track

For a full breakdown of our Project Purpose, Objective, Scope and Approach please refer to our Project Proposal Document.

Changes to the Project Proposal

Sprint one deliverables added to existing finappster work

Originally, the "Long Quiz page" and "My Values page with existing investments" deliverables were to be implemented on a standalone *'create-react-app'* project for Sprint One. When it came time to start writing up user stories and looking into development however, the development team changed their mind on this concept and decided to implement these Sprint One deliverables in the existing *finappster* application. This meant spending more time learning how to use the front and backend of the existing *finappster* application but will save time down the track by not needing to migrate the work completed in Sprint One to the existing application.

Heroku deployment moved to Sprint two

The development pipeline decided upon for this project was a mixture of Scrum and Kanban; Scrumban. This is still the plan, but for Sprint One, the user stories and tasks completed were done locally rather than deploying to a development server on Heroku. This meant that the client was not able to test user stories that passed the development team's acceptance tests, (which would be pushed back into earlier columns if the client did not approve) eliminating the Kanban element of the project at this stage.

This has been discussed with the client and they understand and approve of this approach. Upskilling, understanding the existing *finappster* codebase and prioritizing delivery of features formed the development team's focus instead. Rather than deploy to Heroku for Sprint One, the development team showcased the features developed this sprint through screen-sharing the localhost environment of the project over Microsoft Teams. The development team has also declared that implementation of the Heroku development server for the project will be a priority feature for Sprint Two. The client is satisfied with this approach.

Current Status

Though this is the first time many of *finappster Sigma's* team members have interacted with a real-world client, the team is performing very well and has adapted rapidly to a new environment. The team did have some misunderstandings around the original project proposal expectations (completion was expected rather than a draft in week six), but through feedback from moderator, mentor, and paper leader input, as well as some hard work in scope definition, design, and development, *finappster Sigma* is making good progress on delivery of their R&D project.

At the current point, *finappster Sigma* has completed their first sprint where implementation of the base structure of the quiz and quiz results was completed, straying away from developing features which rely on data that may not be currently available, as development from another *finappster* team is (or now, was) pending.

The mentor of *finappster Sigma* has also proved to be an asset to the development team, providing crucial feedback after client meetings and to any documentation the team needed to submit.

The client of *finappster Sigma* has also developed a strong working relationship with the team, actively involving herself in design and testing.

Work Completed

Project Proposal

For the first six weeks of the semester, *finappster Sigma* focused on getting familiar with the current state of *finappster*, the financial industry in general, and brainstorming how to add the next feature that the client had in mind (to a general sense). This process went on until after the original project proposal presentation in week 6, at which point the team received feedback from their moderator and paper leader that completion of the project proposal was expected at this point rather than a draft. This led to the team pushing out a completed project proposal in week 8.

Designs & Upskilling

After the proposal, *finappster Sigma's* main priority was to collaborate with the client and amongst each other to develop a clear design on what features both *finappster Sigma* and the teams that come after would focus on. This led to Miro designs on the user journeys of the three main users of *finappster,* the Investor, Investment Provider and Admin Support users. After getting approval from the client on these designs, work on wireframes in Figma were created, used to have the client and the development team reach a consensus on the expected portrayal of the designs outlined earlier. Upskilling in Figma and Miro, general understanding of the unrelated user journeys and how to translate them across to *finappster* were important steps that were followed throughout the design process.

Upskilling in the existing *finappster* codebase and in the Django Rest Framework/Python and React were intended at this point, but due to time constraints it was moved to Sprint One.

Sprint One & Upskilling

During Sprint One *finappster Sigma's* main priority was to be able to implement the basic structure for the quiz and "My Values" page. As most of the remainder of the project will rely on both the data that *finappster Tau* will produce and SDG data that the client has retrieved, early user stories were designed make do without *Tau's* data. Work was also split up on the user stories into separate

development tasks to effectively organise implementation and track the dependencies of each task while developing.

Throughout Sprint One, time was dedicated to understanding the existing *finappster* codebase. A meeting with *finappster Delta* was also arranged during the Sprint to get further clarification on the existing codebase that they had worked upon, as *Delta's* work was the basis for *finappster Sigma's.* Following this, work was done on existing codebase on our machines in preparation for sprint 1. Further upskilling throughout the Sprint also included online tutorials and official documentation on the Django Rest Framework, Postgres and React.

At the end of Sprint One, development was complete for the quiz and quiz results related user stories and tasks, but Heroku deployment and development of the existing investments related user stories/tasks were not finished and have been pushed back to Sprint Two.

Mid-Project Review

Upon the completion of Sprint One, *finappster Sigma* scheduled a Sprint Review meeting with the client and showcased their progress. The team have now completed the Mid-Project Review and all required deliverables.

Recommendations for improving team performance

While *finappster Sigma* is happy with their progress overall, there remains some room for improvement. Aspects we have identified are listed below:

More thorough user story development

Spending some more time on development of user stories and tasks in future sprints. Something that happened a couple of times in Sprint one was the development of a user story or task was rushed through, and then when it came time to implement it, time was wasted developing it in a way that did not match what we wanted to develop or lead to confusion by the assigned developer until they realized they needed to redesign the user story. One example of this was the task that involved creating dummy data for quiz results. This concept made no sense when the quiz design we had in mind meant we would be adding data to the database when submitting the quiz results, so real data was available. That task ended up being deleted later in the sprint.

To address this issue, *finappster Sigma* will dedicate more time to the user story development process, rather than rushing it through like what was done in Sprint One. This may not completely fix this issue, as part of the problem is the lack of knowledge around how to implement the task or user story before diving into development. What these changes will do however, is increase user story and task quality through increased design and discussion during this process. This should reduce the likelihood of future occurrences of this problem.

Assigning developers to work on preferred user stories

This did not end up being too much of a problem, but it could have been. Allocation of user stories to developer was random, with the idea of having developers work on the tasks they want to work on not being used. Certain team members have preferences for certain tasks or certain development environments (backend vs frontend), so in future sprints, having developers work to their strengths and preferences should improve performance.

Keeping options open

One hiccup that led to a task being stuck in limbo for a week was a front-end library that was almost working to complete a certain task. The library, according to the documentation, should have been able to accomplish what the developer set out to do, but it was discovered much later after much trouble that it was not capable of properly accomplishing the task being worked on. The main developer at this point had received help from another developer, but progress was still slow. At this point, the developers decided to look elsewhere and found an alternative library that managed to complete the task in short order. This situation served as a lesson to the team; if the method at hand is not working, remember to consider alternative methods (something that React has plenty of, in most scenarios).

Individual contributions / learning

We discussed our individual contributions as a team and agreed on the following:

Chris Stehlin

Chris contributed to all facets of the *finappster Sigma* project, whether that be communication, documentation, design, or programming.

Communication-wise, Chris served as the primary communication medium between client and development team, speaking and often helping articulate ideas on behalf of the development team in client meetings and emails.

Documentation-wise, Chris contributed to many sections of the project proposal and helped proof-check all project documentation. He also helped construct many of the user stories and acceptance tests on Trello.

Design-wise, Chris encouraged discussion on design concepts between client, development team, and amongst the development team themselves. He also contributed to whiteboard, paper, figma and Miro designs, additionally fleshing these out as needed.

Programming-wise, Chris added to both front-end and back-end user stories/tasks, but with a focus on the back end. In the Python/Django Rest Framework pre-existing backend of *finappster,* he spent time learning how to use this language/framework that his team was not familiar with, working with John to understand it. Following that, he created the Sustainable Development Goals (SDG) table, added the 17 SDG fields to it, and linked it to the existing user table. Thereafter, he set up the POST, GET, PUT and DELETE requests for the new table, testing it using Postman. Finally, he created and formatted the bars of the 5Ps on the My Values page, the bars reading in the data from the quiz developed by others of his team.

John Isaiah Sangalang

John contributed to the initial design in Miro and Figma particularly in creating user journeys, wireframes, and designs for the three major users (investor, investment provider and admin support). At the same time working with our client and the development team to understand the user journey and visualize the missing piece of *finappster*, along the way he was able to grasp better understanding on how the software works and the benefit it brings in terms of designing and prototyping.

In terms of development, John was active on both the backend and frontend, assisting the team in setting up their development environment and helping them understand if they are not on track with their tasks. John introduced a template for Pull Requests in the GitHub repository, which allowed the team to have a detailed description on the changes and makes it easier to review the code. The tech stack for the backend was relatively new for him, however, by setting up the environment and assisting his team members, he was able to gain a better understanding of it. Allowing him to complete tasks that he was involved with such as setting up the base code for both the long-quiz and my-values page so that the team may start developing, setting up the route for API requests (POST, GET, UPDATE and DELETE), implementation of the SDG model data with the assistance of a team member, implementation of long-quiz page with a team member utilising a React library to help achieve this goal and fixing bugs that occur while at the same time consult his team members about his task and whether they are up to standard.

Jose Santos

During the early stages of the project, Jose contributed to the brainstorming of the initial idea of our project through the creation of diagrams on paper and whiteboard. During the initial discussion on how we are going to refine the idea of finding your KiwiSaver into smaller steps, Jose mocked up example designs for how it could look.

Along with contributing to the creation of the project proposal document, Jose also helped with the creating user stories, acceptance tests and suggested to the team a new method of creating and assigning tasks on the Trello board that proved to be more efficient than what most of the team was used to.

During the initial creation of the design and discussion for the specifics of the new features to be implemented, Jose contributed by creating the initial wireframes for the quiz and my values page and creating multiple mock-up designs for how each page could look to give our client choices on how data could be efficiently communicated to the user.

During the development however, Jose was focused on the front-end and the implementation of the quiz page and its sortable grid feature for the 17 SDG's. Jose also noted that because we are working with code from previous teams, we may be working with the previous team's technical debt. Because of this, he brought it up to the team and asked for our mentor's suggestion as it is not clear if refactoring their old code, having two coding standards as the current one seems untidy or if it's more efficient to follow the old coding standards.

Peter Scandle

During the initial design and industry upskilling phase Pete worked with the rest of the team, helping them create user journeys, wireframes and Miro designs as required. He worked with the client and the rest of the team, shaping discussions and creating designs to both upskill in the industry and further define the direction of the project. Pete also worked with Chris to help lead team meetings and keep them on track.

He also met with the client to discuss the implementation of the databases currently in *finappster* and liaised between her and the rest of the team to gain an understanding of what data was already in the system and what data would need to be added to meet *finappster Sigma*'s goal. This is still unclear, as we have not had a chance to investigate the exact data feeds that has been added by the current team, *finappster Tau.*

Pete contributed to many sections of the Project Proposal and did a lot of the formatting of team documents. He performed proof-reads throughout development of the document, including assisting Chris with the final proof-read. He proof-read and formatted this Mid-Project Review document as well.

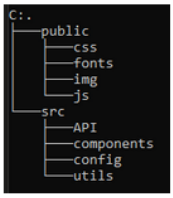
Programming wise Pete needed to spend much of the first sprint upskilling, as he was largely unfamiliar with both the React frontend and the Django / Postgres backend. Once he had completed this, he began working on the *add dummy data for existing investments* task as well as the *switch deployment from localhost to Heroku* task, but both tasks were pushed back to the next sprint for a multitude of reasons.

Issues Around Inherited Technical Debt

From: Jose Santos <sff4900@autuni.ac.nz>
Sent: Friday, 24 September 2021 11:45 PM
To: Barry Dowdeswell <barry.dowdeswell@aut.ac.nz>
Subject: Finappster - front-end code suggestion

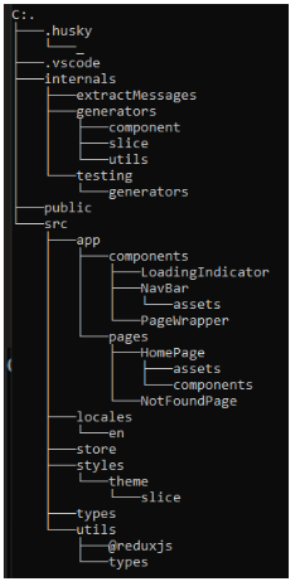
Hi Barry,

I just have one question about the front-end. I have been playing around a lot with the code in front-end and I can't help but notice that the folder structure is a bit unorganized and that the code could be written in a better way. The current folder structure looks like this:



All the pages and components are all put inside the components folder and nearly all of the CSS is put inside one file. Files like dashboard file and CSS files are way over a thousand line long. I was wondering if it may be a good idea to fit in within our project or as I was thinking, during the semester break, to restructure it to make it easier for future teams to add on to the current design in the front-end and potentially improve in speed when a user has a lot of investments to load on each page.

I'm a bit stuck currently as I'm caught in between if we should follow the old teams coding standards or improve upon it and by organizing the file structure. I was thinking of a structure like this which is what I normally use in my projects.



This structure will not only organize it, but I was also thinking of new technologies as opposed to the ones being used currently.

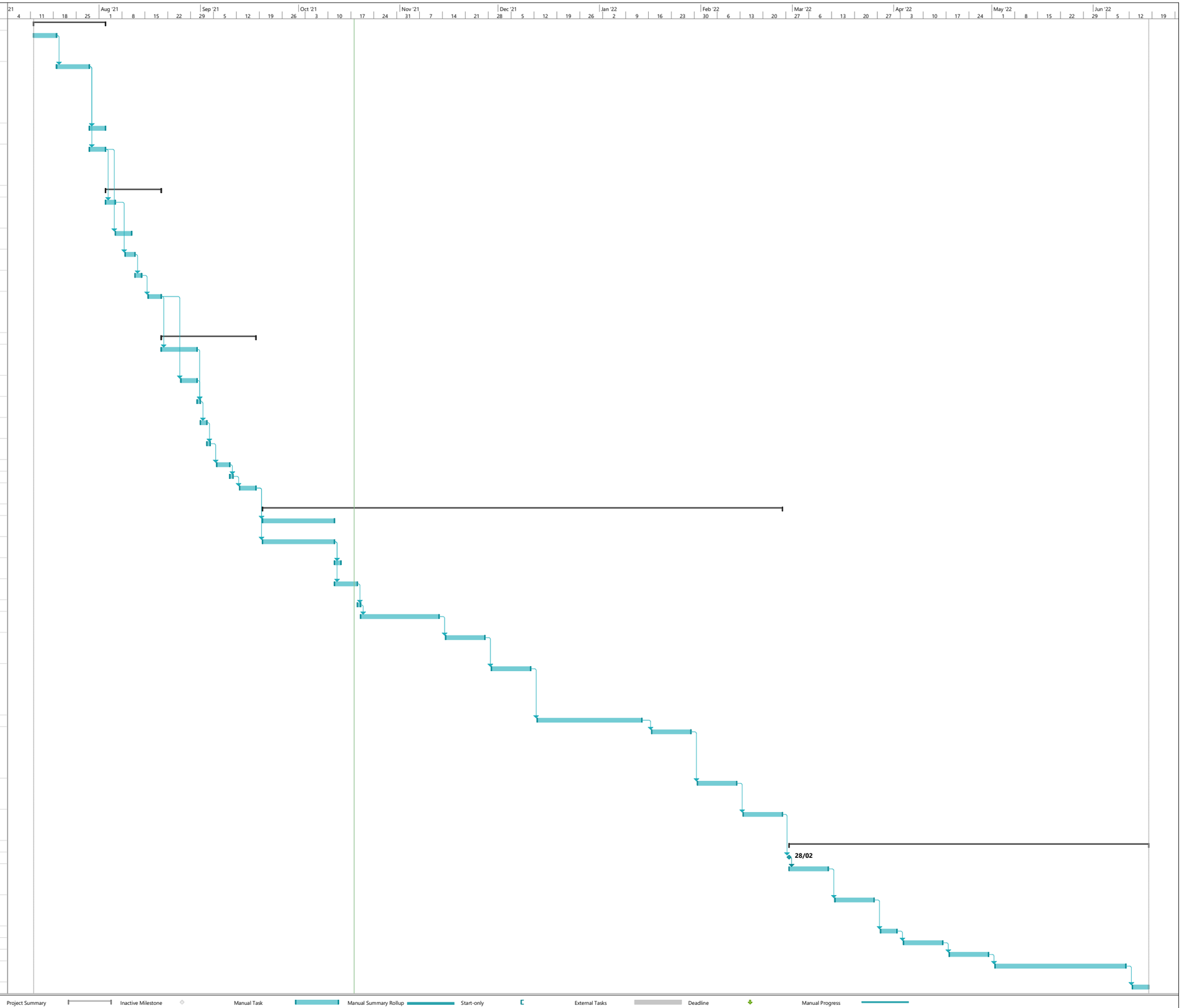
I would love to hear your input on this as we've started on developing and I feel the front-end could be improved a lot for future teams and for us. The backend, though have a lot of deprecated technologies is fine however and only requires us to learn more about python and the Django framework.

Also, I understand if refactoring the code base is not ideal, but should we follow the old teams coding standards, or would it be a good idea if we use our own coding standards and use new technologies as this most likely won't affect the code at all. It will affect future teams as two different coding standards could be confusing to debug. But if you have any questions, I would like to discuss this with you further when you have the time.

Thanks,

Jose Santos
0277229099
Student ID : 17993442

ID	Task Mode	Task Name	Duration	Start	Finish
1		**Planning Phase**	**16 days**	**Mon 12/07/21**	**Mon 2/08/21**
2		Introduction to client & surface learning about project	6 days	Mon 12/07/21	Sun 18/07/21
3		Completion of planning documents (Team Contract, Project Justification, Project Charter, Scope Statement etc.)	8 days	Mon 19/07/21	Wed 28/07/21
4		Financial industry upskilling	3 days	Thu 29/07/21	Mon 2/08/21
5		Creation of KiwiSaver, Selling a Product and Admin Support user journeys	3 days	Thu 29/07/21	Mon 2/08/21
6		**Analysis Phase**	**13 days**	**Tue 3/08/21**	**Thu 19/08/21**
7		Refine user journeys and translate to finappster	3 days	Tue 3/08/21	Thu 5/08/21
8		Refine Project Purpose and Objective	3 days	Fri 6/08/21	Tue 10/08/21
9		Merge user journeys & discuss with Client	3 days	Mon 9/08/21	Wed 11/08/21
10		Update Risk & Issue Registers	2 days	Thu 12/08/21	Fri 13/08/21
11		Research, understand and meet with previous team to discuss scope	4 days	Mon 16/08/21	Thu 19/08/21
12		**Design Phase**	**21 days**	**Fri 20/08/21**	**Fri 17/09/21**
13		Build first general and next in-depth sequence diagrams to cover user journey steps	7 days	Fri 20/08/21	Mon 30/08/21
14		Create Figma wireframes of current scope	3 days	Thu 26/08/21	Mon 30/08/21
15		Discuss and gain feedback from client on designs & wireframes	1 day	Tue 31/08/21	Tue 31/08/21
16		Refine designs and diagrams to match client and user feedback	2 days	Wed 1/09/21	Thu 2/09/21
17		Consensus reached on scope and design with client	1 day	Fri 3/09/21	Fri 3/09/21
18		Develop user stories & acceptance te	4 days	Mon 6/09/21	Thu 9/09/21
19		Perform planning poker for Sprint 1	1 day	Fri 10/09/21	Fri 10/09/21
20		Final refinement of Project Proposal, signoff from Client / Mentor	5 days	Mon 13/09/21	Fri 17/09/21
21		**Implementation Phase Part 1**	**115 days**	**Mon 20/09/21**	**Fri 25/02/22**
22		Development of Sprint 1 features (Long Quiz, SDG / 5P fetching)	16 days	Mon 20/09/21	Mon 11/10/21
23		Internal testing of Sprint 1 features	16 days	Mon 20/09/21	Mon 11/10/21
24		Sprint 1 retrospective & client showcase	2 days	Tue 12/10/21	Wed 13/10/21
25		Mid-Project Review Documentation Completed	5 days	Tue 12/10/21	Mon 18/10/21
26		Mid-Project Review Presentation	1 day	Tue 19/10/21	Tue 19/10/21
27		Short break before development resumes over the summer break	18 days	Wed 20/10/21	Fri 12/11/21
28		Sprint 2 development & testing (Heroku deployment, polish Sprint 1 features, create Short Quiz)	10 days	Mon 15/11/21	Fri 26/11/21
29		Sprint 3 Development & Testing (Ensure all available data incl. from Tau is easily accessible to prepare for recommendation system, begin upskilling)	10 days	Mon 29/11/21	Fri 10/12/21
30		Christmas Break	24 days	Mon 13/12/21	Thu 13/01/22
31		Sprint 4 Development & Testing (Begin prototype Recommendation system work, implementation will vary depending on available data)	10 days	Mon 17/01/22	Fri 28/01/22
32		Sprint 5 Development & Testing (Complete Reccomendation system)	10 days	Mon 31/01/22	Fri 11/02/22
33		Sprint 6 Development & Testing (Polish existing features in preparation for handover)	10 days	Mon 14/02/22	Fri 25/02/22
34		**Implementation Phase Part 2**	**80 days**	**Mon 28/02/22**	**Fri 17/06/22**
35		R&D Part 2 Kick-off	0 days	Mon 28/02/22	Mon 28/02/22
36		Sprint 7 Development & Testing (Extra sprint in case further development is required)	10 days	Mon 28/02/22	Fri 11/03/22
37		Sprint 8 Development & Testing (Extra sprint in case further development is required)	10 days	Mon 14/03/22	Fri 25/03/22
38		Preparation for project handover	5 days	Mon 28/03/22	Fri 1/04/22
39		Begin work on Project Portfolio	10 days	Mon 4/04/22	Fri 15/04/22
40		Mid-Semester Break	10 days	Mon 18/04/22	Fri 29/04/22
41		Produce Portfolio, Poster and Reflective Reports	30 days	Mon 2/05/22	Fri 10/06/22
42		Poster Presentation during this week	5 days	Mon 13/06/22	Fri 17/06/22



Project: project_schedule_v3
Date: Mon 18/10/21

Task | Milestone | Project Summary | Inactive Milestone | Manual Task | Manual Summary Rollup | Start-only | External Tasks | Deadline | Manual Progress
Split | Summary | Inactive Task | Inactive Summary | Duration-only | Manual Summary | Finish-only | External Milestone | Progress