Friendstagram:
Personal Friends Recommender

Project Proposal

Group Members: Royce Ang Jia Jie (Leader)
Manav Arora (Vice-Leader)
Jovan Huang Tian Chun
Clarence Hong Shi Man
Zhu Weiji
Tan Hui Zhan

Team name: Inspiration (Lab: SSP4)

Date of submission: 7 February 2022

Submitted to—

Prof Shen Zhiqi (Dr)
School of Computer Science & Engineering

Table of Content

Table of Content 1

Executive Summary 2

Statement of Problem 3

Objectives 4

Technical Approach 6

- Customer Needs 8
- Target Specifications 8
- Technology Stack 9
- Table 1: List of frameworks and libraries used and the reasons 9
- System Architecture/Platform 10

Project Management 11

Deliverables 13

Budget 14

Communication and Coordination with Sponsor 15

Team Qualifications 16

References 18

Appendix A: 19

- Résumés of Team Members 19

Executive Summary

This is a project proposal for ‘Friendstagram’, a friend recommendation system for Nanyang Technological University students in the form of a web application. The objective of ‘Friendstagram’ is to provide opportunities for users to find new potential friends of similar interest and locations. The web application will be optimised for mobile devices as students are regularly on their mobile phones. Various requirements have been identified for the web application, along with the proposed solution to fulfil these requirements.

In the beginning of our design phase, we will be using Figma. Our Front-end developers will be using React component library and our back-end developers will be using Django. For front-end unit testing and functional testing, we will be using React-Testing Library while for back-end testing, we will be using Pytest. The architecture we are using for our web application is a client-server architecture. As for hosting, we are using Netlify for the client and PythonAnywhere for the server. We will also be using Firebase for authentication and image storing purposes.

Since the goal of the project is to create a web application that recommends new potential friends to students without having any unessential features, we decided to use a Minimum Viable Product (MVP) approach. This MVP approach is taken after considering various factors such as duration of the project, team size, and the technical skills of each individual member. The following sections will help to elaborate further on the points mentioned before.

Statement of Problem

The COVID-19 pandemic has disrupted university students' daily lives. Many social restrictions have been put in place to prevent the spread of the virus amongst student cohorts. Co-curricular activities have been cancelled and lessons have been largely moved online, thus, reducing social interactions between students [1, 2].

The inability to interact with others has caused students to achieve a less fulfilling university life. In a late 2020 survey conducted on 400 university students found that the shifting of in-person classes to online-learning had caused a significant drop in students' satisfaction with the university courses. Students also felt like they do not belong to a community. Furthermore, when compared to the survey results conducted in the previous year, there was a 23% decline in "feeling engaged in coursework" and 28% decline in "interacting with peers in the classroom". Being together with friends and in-person classes were also selected as the top 2 reasons why students valued being on campus [3]. In Singapore, a survey conducted on the impact of pandemic with 1,066 youths aged between 18 and 35 found that 58% responded they had become more fearful and 54% had responded they had become less sociable [4].

The pandemic has undoubtedly affected many students' academic and social experiences in their university journies. Therefore, we concluded that there is a need for a platform for Nanyang Technological University students to help them in finding new potential friends. We propose the development of 'Friendstagram', a personal friend recommender webapp which is catered specifically for *Nanyang Technological University* students, to help them connect with new potential like-minded friends based on their interests and hall accommodation, so as to enhance their university lives.

Objectives

This document proposes the design of ‘Friendstagram’, a web application that helps users to find new potential friends with the following functionalities:

1. Register and login with personal account

   Each user will have their own individual account that can be accessed only by the user himself/herself.

   The personal account allows the system to save users’ information which will be used by the friends matching algorithm, which is further elaborated in part 3.

2. View and edit user’s information

   The user can view their existing information in the web application and edit if necessary. Password, interests and locations can be edited to update the user's current liking.

3. Recommend new potential friends with similar interests to users through our questionnaire and algorithm

   The designed algorithm will access the saved information of the user and search for new potential friends with similar interests and hall accommodation. The algorithm will then provide the contact details of these new potential friends for the user to contact.

4. Display contact information and interests of a new potential friend from the list of recommended new potential friends

   The user can view the contact information and interests of a new potential friend by clicking on his/her card from a list where each card represents a new potential friend.

Technical Approach

In order to plan how we can achieve our objectives and approach the problem statement, we have to first identify the requirements we need to fulfil when building our web application. After that, we will brainstorm for potential solutions that can meet these requirements.

As mentioned in our problem statement previously, we found out that students find it difficult to make friends due to COVID-19 social distancing rules and restrictions. From there, we started brainstorming for technical solutions that can resolve this issue. We then decided on building a web application that recommends new potential friends to users based on his/her information which we gathered from our questionnaire. In order to make sure users have a unique and personalised experience, our web application will allow them to register for an account which will store their personal information and a list of new recommended potential friends. All in all, our goal is to create a friend recommendation system for Nanyang Technological University students in the form of a web application.

In order to address this issue, we need to devise the requirements and functionalities which the web application should have. After brainstorming, we decided to have features such as a ‘login’ function, a ‘register account’ function that lets users store their personal account details, interests and hall accommodation details and a ‘find potential friends’ function that recommends people of similar interests and hall accommodation to the user. Our goal is to create a MVP with the least possible features that can address this issue. Thereafter, if the MVP is successful in addressing this issue, we can discuss further about future improvements. Hence, we collectively decided to prioritise the quality of these few functionalities instead of extending it into a huge project. Our decision was finalised upon careful considerations of the project duration, the amount of manpower and the technical capabilities of every member.

We decided on the roles we need and assigned them accordingly after careful considerations of our individual technical skill sets and experiences:

1. Project Manager - Manav Arora
2. QA Manager - Jovan Huang Tian Chun
3. QA Engineer - Tan Hui Zhan
4. Lead Developer - Royce Ang Jia Jie
5. Front-end Developer - Royce Ang Jia Jie
6. Back-end Developer - Zhu Weiji
7. Release Engineer/Manager - Clarence Hong Shi Man

Next, we mapped out how the functionalities and requirements would transform into a graphic representation through the web application. We then designed the UI (User Interface) / UX (User Experience) of the web application using Figma. Figma is a great tool as it is platform-independent and works on multiple platforms such as Linux, MacOS and Windows. Being able to view and interact with Figma files in real-time helps ease communication within the group regarding any changes in designs.

As for our designs, we used wireframing to help us ensure that all features, user interactions and transitions were reviewed and consented to by every member. After wireframing, we experimented with different looks for our web-application by using different images to create a moodboard. After we finalised the UI / UX aspect of the web application, we searched for different methods to create an MVP that supports our design. Our team prioritised effective communication throughout the entire design planning phase, which allows us to come to a quick conclusion on the optimal design and way to build the prototype.

While working on our UI / UX, we also agreed on the technology frameworks we need to use for our MVP. For our front-end technology framework, we decided to use React and Redux as they are component-based, efficient and have a smooth learning curve. React allows us to reuse components easily due to its component-based property, which helps us in splitting work. As for Redux, it provides greater ease in state management when we scale the web-application. For our back-end technology framework, we decided to use Django as it is simple, flexible, reliable and scalable. It is also powerful and efficient in terms of API calls and it supports relational databases.

As quality management is extremely important in software engineering, we also planned heavily for testing. We agreed to use functional testing, which essentially means checking the web application to ensure that it is doing exactly what it is meant to. For our front-end, we are using React-Testing-Library. This can help to ensure that our codes are consistently working when we are refactoring them. It also lets us test the components from the end-user's point of view and experience. As for our back-end, we are using Pytest because it is free of charge, quick and easy to learn, and can execute multiple test cases simultaneously, thereby reducing the execution duration. It is also easy to write small tests with pytest and it supports complex functional testing for applications. With its scalability, we can easily expand our project in the future.

Finally, we agreed on using Github for our source code’s version control management system. It is free to use and allows us to collaborate effectively. We can easily review new code, see visual code changes, and confidently merge code changes with automated status checks on Github. As we are in a team of six members, it is important that we manage the project well and allocate all the tasks clearly with clear documentation. Github helps us in allocating the coding tasks well with clear documentation and ensures that the project is well-managed in a unified system. With Github, we can also easily revert any changes made when errors are made.

Customer Needs

Our customer needs are identified as follows:

1. Customers need to be able to access the web application on mobile
2. Customers need to be able to fill in different types of information like interests and hall accommodation details
3. Customers need to be able to receive recommendations on the new potential friends whom they can connect with
4. Customers need to be able to see contact information of the recommended new friends

Target Specifications

1. Customers need to be able to access the web application on mobile - the web application needs to be optimised for mobile browsers since students are regularly on their mobile phones
2. Customers need to be able to fill in different types of information like interests and hall accommodation details - the web application needs to prompts the customer to input their information during the registration of account
3. Customers need to be able to receive recommendations on the new potential friends whom they can connect with - the web application needs to provide suggestions of similar users via the algorithm based on user information
4. Customers need to be able to see contact information of the recommended new friends - the web application needs to display the contact information of each new potential friend that was suggested

Technology Stack

Table 1 below lists the frameworks and libraries used, how they are used, and why they are used:

Technology	**Function**	**Reason**
Figma	UI / UX Design	Online design tool with a real-time collaboration feature similar to *Google Docs*; Inexpensive; Used for designing and prototyping the User Interface.
React	Frontend Development	A *JavaScript* component library that utilises virtual DOM results in faster performance. Supports the component-based system architecture (CBSA) we have designed.
React-testing-library	Quality Control (Frontend)	A testing library built for *React* component library. For writing unit tests and integration tests.
Django	Backend Development	A *Python* Web API library with powerful support with Relational Database Management Systems (RDMS). Also supports Content Management System (CMS)
SQLite	Persistent Data Storage	*SQLite* is a lightweight, fast and full-featured RDMS that is embedded in the backend server.
Firebase Authentication	Backend Authentication	*Firebase Authentication* is a cloud-based third-party identity management services provider. This allows us to manage newly registered and current users.
PyTest	Quality Control (Backend)	A testing library built for *Python* language. Supports unit tests and integration tests to be written and executed; Easy-to-use;
Visual Paradigm	Documentation	A modelling tool with many ready-to-use templates and shapes. Used for creating diagrams to document our implemented system
Netlify	Frontend deployment with CI/CD	A cloud-based hosting platform; Used for deploying the proposed frontend server.
Python Anywhere	Backend deployment with CI/CD	A cloud-based hosting platform for Python applications; Used for deploying the proposed backend server.

Table 1: List of frameworks and libraries used and the reasons

System Architecture/Platform

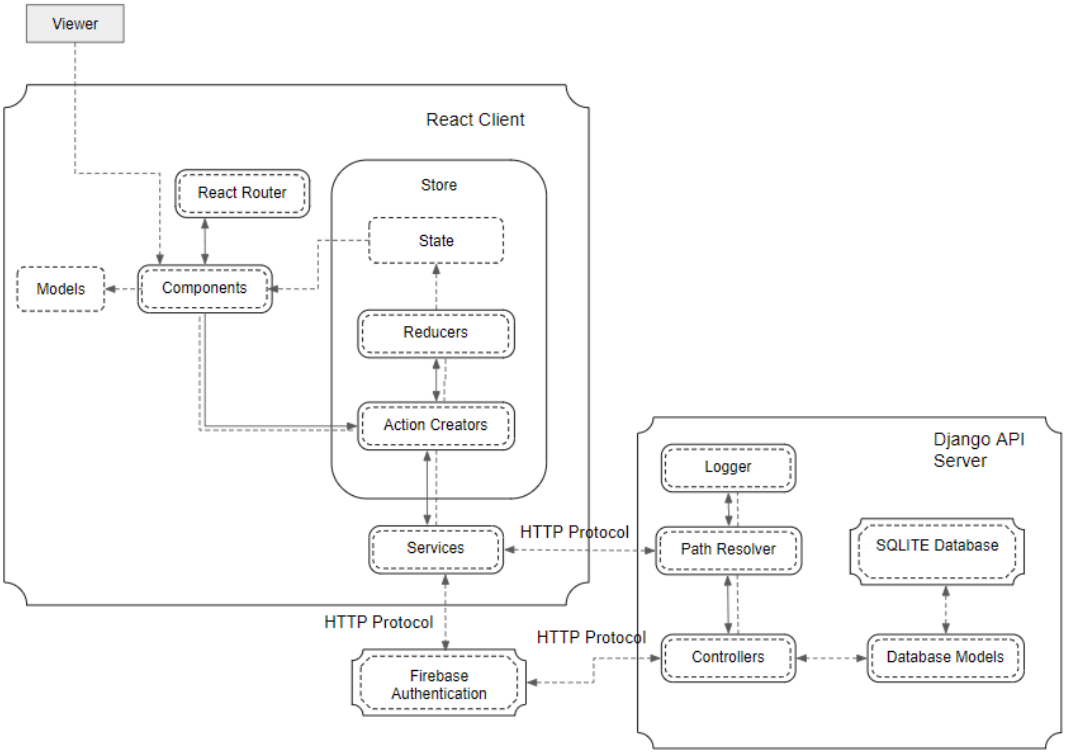

Figure 1: System Architecture of the Web Application.

Figure 1 shows our proposed system architecture. The proposed system architecture has mainly three actively running components: Client built with *React* component library, API Resource Server built with *Django* and *Firebase Authentication*. The *React* Client and *Django* API Resource server will be designed and developed by the project team. To save development time, we propose to use *Firebase Authentication*, a third-party secured identity management service provider.

The *React* Client is composed of the following components:

- **React Router**: resolves the urls which users enter in the user agent (browser)
- **Components**: contain the HTML/CSS/JavaScript and display text or interactable controls to users
- **Services**: interfaces externally with the Django API resource server and Firebase Authentication via HTTP protocol
- **Models**: specify how data objects would look like
- **Store**: a global application storage which stores data used by the components. Store is composed of the following components: State, Reducers and Action Creators. Components call Action Creators and pass data to modify state in the store. Action Creators may call one or more Reducers to change one or more States in the store.

The *Django API resource* Server is composed of the following components:

- **Path resolver** - determines which controller to handle the incoming HTTP request.
- **Controllers** - processes incoming requests and updates the necessary Database Model.
- **Database Model** - updates the persistent storage, SQLite Database

Project Management

ID	Task Mode	Task Name	Duration	Start	Finish	Predecessors
1		**Project Friendstagram**	97 days	Mon 24/1/22	Mon 28/3/22	
2		**Project Planning (Proposal)**	**21 days**	**Mon 24/1/22**	**Sun 6/2/22**	
3		Assignment of Team Roles	1 day	Mon 24/1/22	Mon 24/1/22	
4		Identify Problem	0.5 days	Tue 25/1/22	Tue 25/1/22	3
5		Elicit Requirements	3 days	Tue 25/1/22	Fri 28/1/22	4
6		Identify Technologies to use	2 days	Tue 25/1/22	Thu 27/1/22	4
7		Use Case Modelling	7 days	Thu 27/1/22	Sun 30/1/22	6
8		Design System Architecture	7 days	Sun 30/1/22	Sat 5/2/22	7
9		Design User Interface	7 days	Thu 27/1/22	Sun 30/1/22	6
10						
11		Setup Development Environment	2 days	Sun 6/2/22	Tue 8/2/22	2
12		**Backend Development**	**30 days**	**Wed 9/2/22**	**Sun 27/2/22**	**11**
13		Design Database Structure and Sche	7 days	Wed 9/2/22	Sun 13/2/22	
14		Deploy server to cloud	1 day	Sun 13/2/22	Sun 13/2/22	13
15		**Develop API Endpoints**	**22 days**	**Sun 13/2/22**	**Sun 27/2/22**	**14**
16		Integration with Firebase	7 days	Sun 13/2/22	Sat 19/2/22	
17		Find Friends Endpoints	7 days	Sat 19/2/22	Tue 22/2/22	16
18		Profile Management Endpoints	7 days	Tue 22/2/22	Sun 27/2/22	17
19						
20		**Frontend Development**	**30 days**	**Tue 8/2/22**	**Sun 27/2/22**	**11**
21		Authentication Pages	10 days	Wed 9/2/22	Tue 15/2/22	
22		Find Friends Pages	10 days	Tue 15/2/22	Mon 21/2/22	21
23		Profile Management Pages	10 days	Mon 21/2/22	Sun 27/2/22	22
24						
25		**System Integration**	**21 days**	**Sun 27/2/22**	**Sun 13/3/22**	**15,20**
26		Integration of Authentication Pages	7 days	Sun 27/2/22	Sat 5/3/22	
27		Integration of Find Friends Pages	7 days	Sat 5/3/22	Wed 9/3/22	26
28		Integration of Profile Mgmt Pages	7 days	Thu 10/3/22	Sun 13/3/22	27
29						
30		**Quality Planning and Control**	**50 days**	**Sun 6/2/22**	**Sat 12/3/22**	**2**
31		Write testing plan (what to test)	14 days	Sun 6/2/22	Wed 16/2/22	
32		Implement testing for Authenticatic	10 days	Sat 19/2/22	Fri 25/2/22	31,16
33		Implement testing for Find Friends	10 days	Tue 22/2/22	Mon 28/2/22	17,22
34		Implement testing for Profile Mgmt	10 days	Sun 27/2/22	Sun 6/3/22	18,23
35		Implement Integration testing	10 days	Sat 5/3/22	Sat 12/3/22	26
36						
37		**Other Documentation**	**57 days**	**Tue 8/2/22**	**Fri 18/3/22**	
38		System Requirements Specification	21 days	Tue 8/2/22	Mon 21/2/22	
39		Risk Management Plan	10 days	Mon 21/2/22	Sun 27/2/22	38
40		Software Maintenance Report	14 days	Sun 27/2/22	Wed 9/3/22	39
41		Configuration Management Plan	12 days	Thu 10/3/22	Fri 18/3/22	40
42						
43		Prepare for Presentation	14 days	Fri 18/3/22	Sat 26/3/22	

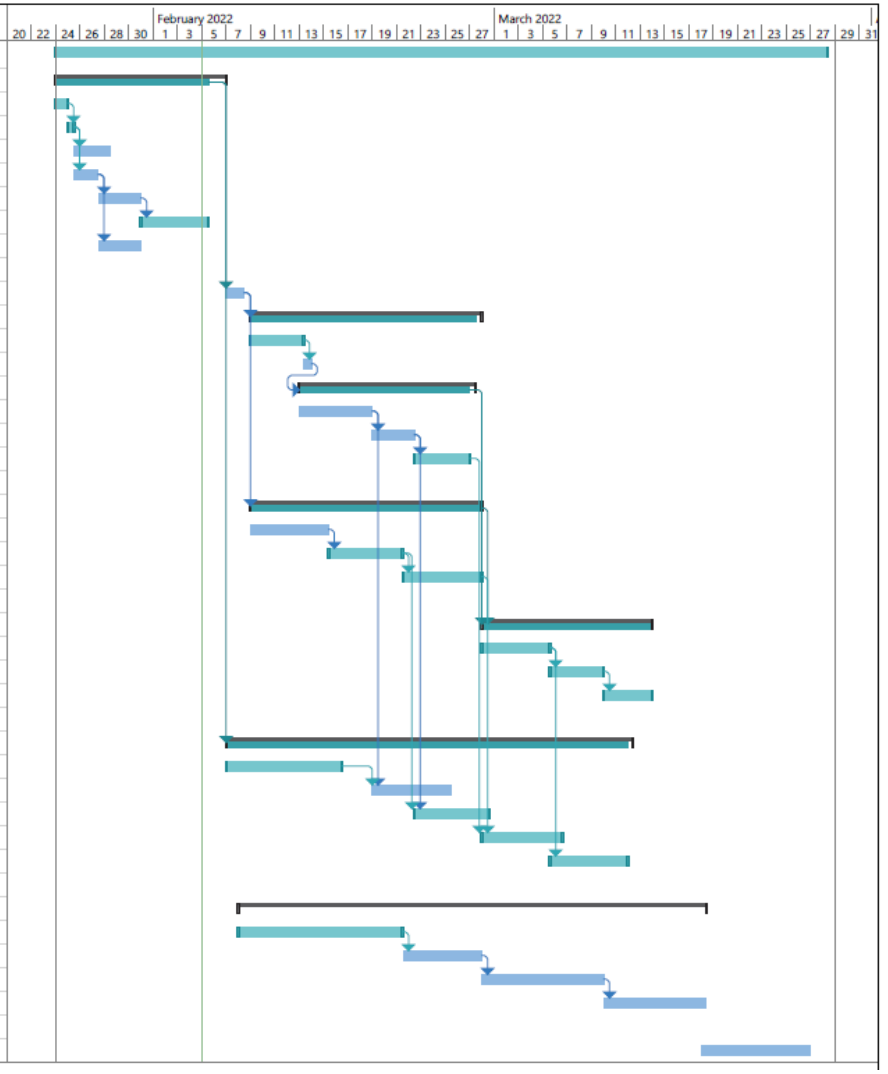

Figure 2: Gantt chart for the project.

The Gantt Chart is split into 6 main groups:

1. Project Planning - The project team plans out the project to be developed.
2. Backend Development - The Engineering (backend) subgroup will commence development of the backend system.
3. Frontend Development - The Engineering (frontend) subgroup will commence development of the frontend system.
4. System Integration - The Engineering subgroup will commence integration of frontend system and backend system together.
5. Quality Planning and Control - This occurs iteratively as development work progresses. The QA subgroup will first plan how the system will be tested, and subsequently implement the automated tests after development for the specific module is completed.
6. Other Documentation - The project team cooperates together to document the developed system.

As the product is being developed, unforeseen circumstances may arise e.g., departure of a team member, or unforeseen bug(s) that take too long to resolve. As such, the specified dates periods are not hard deadlines to be followed but act as a reference for the project team to track if we are on schedule and whether resources should be moved (i.e. moving team members to another subgroup) to ensure the successful delivery of the product by 28 March 2022. As each milestone is reached, we will seek feedback from our target users through demonstration and surveys to ensure the product meets their expectations and correctly resolve their pain points.

Deliverables

For our first deliverable, we will be providing a comprehensive **System Requirement Specification (SRS)**. The SRS will be compiled based on all the meetings we had with our customer and internal team. All the stakeholders of this project must approve the SRS so as to reduce the potential for conflict or other project crippling issues nearing the project deadline.

Apart from providing a SRS, we will also be providing a **lo-fi prototype** to help convey our idea of the entire web design to the customer and internal team. These prototypes will be finalised and provided after we considered all the feedback from the stakeholders. This helps everyone to be aligned on the web design.

We will also provide **technical documentation** for our customers. We aim to make these documentation as beginner-friendly as possible so that customers without any technical experience can understand them well. For customers who are able to understand technical terms well and requests for documentation that are more technical in nature, we can provide our system architecture, use case models, class diagrams and many more. This helps them to understand better on how we build our product.

Next, once our actual product is completed, our product manager will demonstrate our **live physical prototype** and go through every functionalities with our customer. Then, we will share a copy of our **computer program source code** and a **detailed description of our test procedures** which contains all the test cases that were used to ensure that the system satisfies all customer requirements.

Also, if there are any amendments made during the whole Software Development Life Cycle, we will provide our **latest version of the SRS** for our customer. They can then refer to the SRS when checking our **live physical prototype** against their requirements.

Last but not least, we will also provide a **user-friendly instruction manual** that trains the users on how to use our product and modify the code whenever they need to.

Budget

Item	Supplier	Quantity	Unit Price($)	Total($)
Authentication / Storage	Google Firebase	1	0	0
Front-end server Deployment	Netlify	1	0	0
Back-end server Deployment	PythonAnywhere	1	0	0
			Total:	**0**

Table 2: Online hosting services

Item	Supplier	Quantity	Unit Price($)	Total($)
React Library	Open Source	6	0	0
Redux Library	Open Source	6	0	0
Figma	Figma, Inc.	6	0	0
Django	Open Source	6	0	0
Github	Microsoft	6	0	0
Visual Paradigm	Visual Paradigm International Ltd.	6	0	0
Visual Studio Code	Microsoft	6	0	0
React Testing Library	Open Source	6	0	0
Pytest	Open Source	6	0	0
Trello	Atlassian	6	0	0
			Total:	**0**

Table 3: Technological tools

Communication and Coordination with Sponsor

We place great emphasis on our communication and coordination with the sponsor as it plays a pivotal role in ensuring that we are successful in our project. By doing so, we believe it helps us to manage the sponsor's expectations and maximise their satisfaction throughout the whole Software Development Life Cycle of our project.

In order to provide assurance for the sponsor on the team's commitment and abilities and help the team stay accountable, we propose to have **weekly meetings**. These weekly meetings will be conducted online and the platform used will be chosen by the sponsor. The project manager will direct the team's meetings and also acts as the single point of contact between the team and the sponsor.

In our weekly meetings, updates on the technological and non-technological aspects of the product will be provided. Such updates can include the technological blockers which the team is currently facing, the features which are completed, the last minute requests for design or specifications changes by the customer and the set of administrative procedures which the team will follow when carrying out their tasks. These will help to manage everyone's expectations and provide clear visibility for all the stakeholders on the progress of the project every week.

For the **first meeting** which involves the commencing of this project, we will require all the stakeholders, which include the sponsor and the team, to attend. This ensures that every stakeholder is aligned on the problem statement and has a common agreement when coming up with a solution. We will also discuss and finalise all the administrative matters so that all the stakeholders are aware about what has been agreed on in the signed proposal such as the budget and a rough idea of sponsor's expectations.

For **future meetings**, it will progress from discussions on technical requirements to design specifications to code reviews to user acceptance testing. Should there be any last minute and urgent changes before the **weekly meeting**, the project manager will inform everyone through **email**. By using email, it facilitates fast, professional, well-documented and back-and-forth communication which can help to expedite the project.

We hold the sponsor's needs in high regards and aim to exceed the expectations of the sponsor from the beginning to the end of this entire project. Therefore, should there be any areas which fall short of your expectations, do inform us and we will make the appropriate adjustments as soon as possible.

Team Qualifications

Royce has experience in full stack web application development, DevOps and cloud infrastructure engineering. He has also led various teams in multiple projects to completion. His recent internship experience as an infrastructure intern in the United States has also contributed to his ability to build, scale and deploy complex projects in a time-limited setting.

Manav has experience in backend and frontend development using technologies like Python, Android Studio, HTML, SQL, etc. In addition he has experience in scaled project development using agile processes.

Clarence has experience in frontend development using technologies such as Android Studio, etc. In addition, he also has some experience in writing test cases with Junit tests.

Weiji has experience in full stack web application development using both Backend technologies like Django, Flask, Node.js, etc. as well as Frontend technologies like Bootstrap, React, Tailwind etc. Additionally, he provides robust testing of code through a full suite of comprehensive unit tests, full scale integration tests as well as load/stress testing through religious implementation of Test-Driven Development (TDD) to ensure high reliability in all areas of code. Lastly, by bringing experience from leading projects in large MNCs like OCBC, he can help the team in managing the complexities of this project.

Jovan has experience in full stack web application development using BootStrap, Flask/Django, MSSQL, HTML & CSS. He also has some experience in writing back-end logic codes in JavaScript.

Hui Zhan has experience in development using React and Redux and backend development using NodeJS and others. Additionally, he has some experience in writing test cases with Jest and deployment with Docker.

References

[1]

N. Elangovan, “Mix of online and in-person classes for undergrads, as new academic year begins under Covid-19 spectre,” *TODAY*, Aug. 11, 2020. https://www.todayonline.com/singapore/mix-online-and-person-classes-undergrads-new-academic-year-begins-under-covid-19-spectre (accessed Jan. 28, 2022).

[2]

W. K. Ng, “S’pore unis move more classes online to reduce spread of Covid-19 as cases continue to rise | The Straits Times,” *www.straitstimes.com*, Sep. 30, 2021. https://www.straitstimes.com/singapore/parenting-education/spore-unis-move-classes-online-to-reduce-spread-of-covid-19-as-cases (accessed Jan. 28, 2022).

[3]

WeWork and brightspot strategy, “The impact of COVID-19 on the university student experience,” *Ideas*, Feb. 03, 2021. https://www.wework.com/ideas/research-insights/research-studies/the-impact-of-covid-19-on-the-university-student-experience (accessed Jan. 28, 2022).

[4]

J. Ong, “TODAY Youth Survey: Majority more fearful, less sociable due to Covid-19 but hopeful of ‘better’ life a year from now,” *TODAY*, Nov. 08, 2021. https://www.todayonline.com/singapore/today-youth-survey-majority-more-fearful-less-sociable-due-covid-19-hopeful-better-life (accessed Jan. 28, 2022).

Appendix A:

Résumés of Team Members

The following pages present one-page résumés of the team members for this project.

Appendix A1: Royce Ang Jia Jie's Resume

EDUCATION

Nanyang Technological University, Singapore **Aug 2018 – Jul 2022**

- Bachelor of Engineering (Computer Science)
- Expected Honours (Distinction)

ACADEMIC PROJECTS & WORK EXPERIENCE

Workstream **Jun 2021 – Dec 2021**

- Pioneered the architecture design and developed an internal development tool to deploy an on-demand end-to-end isolated development environment in 4 mins with Terraform, Docker, AWS, Jenkins, Bash, GitHub Action for engineers to develop remotely and share application changes without impacting real users.
- Restored service disruption with a Senior DevOps engineer caused by wide-scale September 30th Let's Encrypt Root Certification Expiration that affected more than 250,000 of Workstream's customers.

SAP **Jan 2021 – May 2021**

- Designed and implemented the front-end software capable of rendering high-resolution 3D models for buildings, with features such as automatic defect detection and tagging using AI, report generation and image manipulation. This is done through ReactJS, Redux, ElectronJS and Python frameworks. Front-end libraries like VTK.js, KonvaJS and OpenCV were heavily used in the software.
- Introduced and implemented testing frameworks such as Jest and React Testing Library. Testing tools like selenium were also explored and introduced.

Nanyang Technological University, Singapore

Software Engineering Project **Feb 2020 – Apr 2020**

Title: 1Pair

- Documented the project's use cases, system architectures, UML class diagram, such that the structure of the program will be understood without referring to the code.

SKILLS

Programming Skills: JavasScript, Typescript, Python, Java, HTML, CSS, C, C++
Frameworks and Software Applications: React, Redux, ElectronJS

Appendix A2: Manav Arora's Resume

EDUCATION

Nanyang Technological University, Singapore **Aug 2018 – Jul 2022**

- Bachelor of Engineering (Computer Science)
- Expected Honours (Distinction)

ACADEMIC PROJECTS & WORK EXPERIENCE

Envision Digital Pvt. Ltd. **May 2021 –Jan 2022**

Data Science Intern

- Developed cutting edge deep learning models to bring intelligent insights to local users and to unlock the true performance of solar assets.
- Deployed deep learning pipelines for issues like IV-curve classification, PV cell shading detection and Irradiance series anomaly detection using techniques like LSTM, CNN, SVM, XGB, TadGAN, etc.
- Submitted the paper titled 'Automatic IV Curve Diagnosis with Deep Learning' to the 48th IEEE Photovoltaic Specialists Conference as a co-author.

Seagate Technologies Pvt. Ltd. **Aug 2020 - Dec 2020**

AI Engineer Intern

- Researched and analysed the characteristics, suitability and scalability of AI workloads of high entropy tasks for Computational Storage use cases.
- Incorporated scalable analytics framework and distributed edge computing into the Video Multistream pipeline.
- Applied technologies including Dask, Docker, Opencv, Mxnet etc, to put together a cohesive, Parallel Processing, Horizontally Scalable, Distributed pipeline and it showed a 262% increase in performance w.r.t. the pipeline I started with.

SKILLS

Programming Skills: JavasScript, Python, Java, HTML, CSS, C, C++
Frameworks and Software Applications: Tensorflow, Keras, Sklearn

Appendix A3: Jovan Huang Tian Chun's Resume

EDUCATION

Nanyang Technological University, Singapore **Aug 2019 – Jul 2023**

- Bachelor of Engineering (Computer Science)

ACADEMIC PROJECTS & WORK EXPERIENCE

Micron Technology **Aug 2021 – Oct 2021**

Software Engineer Intern

- Deployed an internal tool, a Full-Stack Web Application using BootStrap, Django, MSSQL, uWSGI & Supervisor in a Linux Server to aid Micron's machine operators in segregating SSD.

Nanyang Technological University, Singapore

Object-Oriented Programming Project **Aug 2020 – Dec 2020**

- Developed a Java application that allows the creation of courses and adding of student records as well as registration of courses and students using Object-Oriented (OO) concepts.

SKILLS

Programming Skills: Python, SQL, C Programming, Java, HTML/CSS, JavaScript, Bootstrap
Frameworks and Software Applications: Flask, Django

Appendix A4: Clarence Hong Shi Man's Resume

EDUCATION

Nanyang Technological University, Singapore **Aug 2019 – Jul 2023**

- Bachelor of Engineering (Computer Science)
- Expected Honours (Distinction)

Nanyang Polytechnic, Singapore **Apr 2015 – Apr 2018**

- Diploma with Merit in Mechatronics Engineering (with specialisation in Automation & Robotics)
- Director's List for all Semesters
- NYP-TDS Scholarship for all Semesters

ACADEMIC PROJECTS & WORK EXPERIENCE

Nanyang Technological University, Singapore

Software Engineering Project **Aug 2021 – Dec 2021**

Title: YOUtility

- Implemented front end with UI design using Android Studio.
- Documentation of system architecture and use cases using UML Models.

Nanyang Polytechnic, Singapore **Apr 2017 – Oct 2017**

Final Year Project – Design a LIDAR system for indoor mapping

- Developed an inexpensive system that promptly maps the interior of any room
- Enhanced the design to reduce parts used and achieved a decrease of 10% project's cost

Tamkang University, Taiwan **May 2017 – Jul 2017**

Overseas Internship Program – Design a prototype robotic arm for object picking

- Developed an autonomous robotic arm that efficiently pick up specified objects to a designated area
- Optimised the program used and obtained real time object searching and tracking

SKILLS

Programming Skills: SQL, Python, Java, HTML, C, C++, PLC
Frameworks and Software Applications: LabVIEW, OpenCV, Autodesk Inventor, Android Studio, Figma

Appendix A5: Zhu Weiji's Resume

EDUCATION

Nanyang Technological University, Singapore **Aug 2019 – Jul 2023**

- Bachelor of Engineering (Computer Science)
- Expected Honours (Distinction)

ACADEMIC PROJECTS & WORK EXPERIENCE

Oversea-Chinese Banking Corporation

Group Operations and Technology **May 2020 – Dec 2021**

- Lead in-house development and upgrading of File Tracking System because of a lack of in-house software development expertise
- Managed to deliver all initial user-specified functionality within 3 weeks of taking over the project, allowing for users to revise the upgrading project to take on a much larger scope, despite lack of pre-existing documentation or any project owners to communicate with.
- Going beyond what was initially tasked and providing value to the project by implementing a test-suite covering 75% of the 15000LOC codebase, fixing security vulnerabilities and adding documentation to the bare codebase.
- Performed data analysis and visualisation on department workflow and sales data

Ideas Ink. School (Buona Vista) **Aug 2020 – Mar 2021**

Web Development, Full Stack Developer (Contracted)

- Prepared functional, non-functional requirements and tech stack after discussions and training with owner and stakeholders to understand their needs.
- Developed and deployed a website for internal use to ramp-up tutors with Django, PythonAnywhere, and Google Cloud API
- Created evaluation metrics and back-end analysis software to improve productivity of tutors at the school by improving their teaching and reduce the need of senior tutors to mentor new tutors, reducing headcount by 2.
- Preparing extension of the project to include take-home assignments for the students at the school, and easily accessible digitised notes for the school.

SKILLS

Programming Skills: Python, JavaScript, Kotlin, Web development, Data Science and Visualisation
Frameworks and Software Applications: Django/Django-REST, Flask, React, D3.js, Antv G6, Node.js, Firebase,

Appendix A6: Tan Hui Zhan's Resume

EDUCATION

Nanyang Technological University, Singapore **Aug 2019 – Now**

- Bachelor of Engineering (Computer Science)

ACADEMIC PROJECTS & WORK EXPERIENCE

Omega Sloth Pte Ltd **Apr 2021 – June 2021**

Backend Intern

- Created several data management APIs for a backend built with NodeJS and MongoDB.
- Attained 95% test coverage for unit and integration testing using Jest testing framework to ensure correctness in implementations.
- Deployed application in Amazon Web Services (AWS) with Docker with CI/CD pipeline to automate testing and ensure reliable release of application.

Alpha Meerkat Pte Ltd **Apr 2020 – June 2020**

- Designed wireframes for a management dashboard on Figma for frontend development.
- Developed the frontend system with the designed wireframes using React component library.
- Collaborated with the backend team to ensure successful delivery of product.
- Achieved 90% coverage for unit tests on components using React Testing Library to ensure correctness of flow of user interactions.

Academic Projects **Aug 2020 - Dec 2021**

Title: LL Fruits Picker

- Full stack development of fruits picker website using ReactJS for frontend and NodeJS for backend.
- Designed a Hi-Fi Prototype on Figma.

SKILLS

Programming Skills: JavaScript, HTML, CSS, PHP, Python, Java, C
Frameworks and Software Applications: React, VueJS, .NET, Figma, Illustrator