

UNIVERSITY OF MELBOURNE



THE UNIVERSITY OF
MELBOURNE

SOFTWARE PROCESSES AND MANAGEMENT

SWEN 90016

Project Management Plan

for

Shipping Management System

Prepared by Team CSR_04_5

Team Member:

Linrong CHEN	linrongc@student.unimelb.edu.au
Boheng LUAN	bluan@student.unimelb.edu.au
Ruihan ZHANG	ruihanz@student.unimelb.edu.au
Zeyu HUANG	zeyuh3@student.unimelb.edu.au

September 2018

Executive Summary

This project aims to develop a web-based shipping management system for our client Susanto.

Susanto has been engaged in cargo shipping services from Melbourne to Jakarta, Indonesia for many years. Based on his work experience, he found that the companies in the industry had paid little attention to improving efficiency and user experience in their services. After failing to persuade his employer into solving these problems, Susanto made up his mind to start a his own business which provides dynamic, efficient and user-friendly cargo shipping services. Susanto hopes this project can implement a software platform for his new company.

A small team named CSR_04.5 consisting of postgraduate students from the subject SWEN90016 at University of Melbourne will be responsible for evaluating, planning and implementing the project. The roles and members in the team are shown below.

Linrong Chen	Full-stack Developer & Scrum Master
Boheng Luan	Back-end Developer & Product Owner
Ruihan Zhang	Front-end Developer & Quality Assurance
Zeyu Huang	Front-end Developer & Risk Manager

Table 1: Members and Roles

As this is an experimental product in the market with constraints in time and budget, the Agile framework will be followed as the Software Development Life Cycle (SDLC) to ensure flexibility and productivity.

All members of the implementing team are not paid for the project. The financial cost of the project can thus be ignored.

The desired outcome of the project is a web-based system facilitating door-to-door shipment service. Users of the system, as Shipper, Collector or Customer, should be able to initiate, monitor, modify and/or be notified of shipping orders from Melbourne to Jakarta, Indonesia. Shipping booking information such as size, destination and cost can be stored and viewed on the system.

Benefits of the project are multifold. The desired product will help boost the efficiency and customer satisfaction of logistics service that ships cargo from Melbourne to Jakarta. It will also build a solid competitive edge for Susanto's start-up company against traditional shipping companies and bring potential profit to the company.

Through the project, members of team CSR_04.5 will gain experience in project management within Agile framework and develop skills in implementation of a web-based system.

The project schedule will be consistent with the curriculum plan of the subject. The team will implement the project through multiple Agile sprints in roughly 6-8 weeks. The project will be finalised by week 11 of the semester (14 Oct 2018).

Contents

1	Introduction	3
1.1	Purpose of Document	3
1.2	Audience of Document	3
1.3	Limitations of Document	3
1.4	Evolution of Document	3
2	Project Information	3
2.1	Key Stakeholders	3
2.2	Scope	4
2.2.1	What is in-scope?	4
2.2.2	What is out-of-scope?	6
2.3	Delivery Approach	6
2.4	Business Value	6
2.5	Constraints	7
3	Project Governance	7
3.1	Roles and Responsibilities	7
3.2	Communication Plan	9
3.2.1	Communication Channels	9
3.2.2	Communication Matrix	9
3.3	Risk Management	10
3.4	Technology	11
3.5	Project Planning	12
4	Project Execution, Monitoring and Control	13
4.1	Project Status: Non-teaching week	13
4.1.1	Process Related Artefacts	13
4.1.2	Product Related Artefacts	14
4.1.3	Risk Monitoring and Control	15

1 Introduction

There is a big Indonesian community in Melbourne, and they provide a lot of demand in sending gift boxes back to Indonesia. Project sponsor Susanto is committed to providing better logistics services to the Indonesian community in Melbourne via information technology. This project aims to develop a website-based system for Susanto's logistics companies to facilitate users to obtain information, including shipping schedule, price information and so on. Users can also fill in/modify/query their orders information on the website, and the system will notify the sender and recipient by automatic mail whenever the order information changes.

1.1 Purpose of Document

The purpose of this document is to provide a complete project management plan for this development project. This document will not only guide the development process, but also document progress of the project.

In the following chapters, we will discuss in detail the key Stakeholders of the project, the scope of the project, the reasons for choosing Agile as the SDLC, and the business value and constraints of the project. We also documented the roles and responsibilities of each member in the first release, as well as the communication plan, the initial technical framework, risk management and project plan.

1.2 Audience of Document

This document is oriented to stakeholders, which include the project managers and the developers of this project.

1.3 Limitations of Document

Since the project is still in the initial planning stage of development, the technical framework and system features described in the first version of the plan document is determined by preliminary discussions, which may be changed with the project progresses and further communications with stakeholders and mentors. Similarly, since it's a planning phase, this document doesn't cover specific Project Execution, Monitoring and Control. Because there are no historical data, we can't estimate the initial velocity of the project at burn down chart. This document is for internal developers and is responsible for documenting the progress and management of the entire development process, so this document does not contain an analysis of the project's marketing strategy. Moreover, quality benchmarks not being explicitly specified can also be considered as a limitation.

1.4 Evolution of Document

Please refer to the table 2 'Evolution of Document'.

2 Project Information

2.1 Key Stakeholders

The key stakeholders of this project are identified as in Table 3.

version	created by	date created	location	comments
1.0	Boheng Luan	7 Sept 2018	unimelb ERC	created executive summary & section 1.1 - 1.4
1.0	Zeyu Huang	7 Sept 2018	unimelb ERC	created section 2.1 - 2.5
1.0	Ruihan Zhang	7 Sept 2018	unimelb ERC	created section 3.1 - 3.3
1.0	Linrong Chen	7 Sept 2018	unimelb ERC	created section 3.4 - 3.5
1.1	Boheng Luan	27 Sept 2018	unimelb physics	updated executive summary, section 1.3 - 1.4; co-created section 4.1.2
1.1	Ruihan Zhang	27 Sept 2018	unimelb physics	updated section 3.1 - 3.3; co-created section 4.1.1
1.1	Linrong Chen	27 Sept 2018	unimelb physics	updated section 3.4, 3.5; co-created section 4.1.1, 4.1.2
1.1	Zeyu Huang	27 Sept 2018	unimelb physics	updated section 2.1 - 2.3, 3.3; created section 4.1.3

Table 2: Evolution of Document

As the initiator of the company and client of the project, Susanto is responsible for clarifying the requirements for the product and give feedback to the scrum team to help them better understand use cases and key demands.

As the collector of the company, Susanto’s brother-in-law is responsible for providing feedback as a Shipper. Together with Susanto who is also the Shipper, they will have access to specific features unavailable to other customers and help the scrum team test the part of the system.

As the management consultant of the project, Professor Karunasekera will provide valuable advice on project management and help the team develop skills in communication, scheduling, risk management, etc..

For roles and responsibilities of internal stakeholders, please refer to Section 3.1.

2.2 Scope

2.2.1 What is in-scope?

The project is to develop a web-based platform for door-to-door shipment management which should be dynamic, efficient and user-friendly. The scope for this project can be classified as below.

User registration and login

- **Shipper**: A Shipper should have a pre-defined email username with a default password for login. A Shipper should be permitted to perform Shipper specific functions.
- **Collector**: A Collector should have a pre-defined email username with a default password for login.

Name	Position	Project Role	In/External	Contact Information
Susanto	Initiator	Client, Sponsor, Shipper	External	
Susanto's brother-in-law	Truck Fleet Owner	Collector	External	
Linrong Chen	Postgraduate student	Full-stack Developer & Scrum Master	Internal	linrongc@student.unimelb.edu.au
Boheng Luan	Postgraduate student	Back-end Developer & Product Owner	Internal	bluan@student.unimelb.edu.au
Ruihan Zhang	Postgraduate student	Front-end Developer & Quality Assurance	Internal	ruihanz@student.unimelb.edu.au
Zeyu Huang	Postgraduate student	Front-end Developer & Risk Manager	Internal	z.huang56@student.unimelb.edu.au
Professor Shanika Karunasekera	Subject Co-ordinator	Management Consultant	External	karus@unimelb.edu.au

Table 3: Key Stakeholder Declaration

- **Customer:** A customer should be able to register with their personal information including name, home address, contact phone number, email address and initial password. A customer can login with their email address and password.

Personal Information The personal information of customers should be maintained as profiles of users and can be updated by customers.

Shipping_Booking system

- **Requests:** Logged in customers should be able to make **Shipping_Booking: Requests**, which specify size, destination, pick-up address, preferred departure and arrival dates and (optional) message to the shipper.
- **Acks:** A shipper can respond to a **Shipping_Booking:Request** with **Acks**, which must specify status, pick-up date and time, cost, HBL number and (optional) message to the customer.
- **Viewing:** Logged in customers can *view* the status and all associated data of a **Shipping_Booking** that they created earlier.
- **Modifying:** The Shipper can *modify* a **Shipping_Booking:Ack** with permitted status but cannot change the cost.

Notification The customer and the collector should get email notifications of modification to the Shipping_Bookings with all the information.

Shipment Information The shipment information should be selected from a list of dates with at least three date set options customizable to the users.

Persistent Storage The customer information and Shipping_Booking details should be persistent in the system.

UX and Web Design The product should have a web-based UI. The UX design must be responsive.

2.2.2 What is out-of-scope?

The following features are out of the scope of the project.

- Integrated payment system
- Multi-Collector or Multi-Shipper support
- Explicit data validation

2.3 Delivery Approach

We prefer Agile SDLC for this project based on the following considerations.

- **Flexibility** The desired product is innovative and adapts to a boosting business which can be volatile in style and paradigm. Thus, the Agile framework is suitable for this project where user requirements are not fully defined and functionalities are up to change. For example, since this the first company that offers door-to-door cargo-shipping service from Melbourne to Jakarta, both business logic and software platform requirements are unclear as we start the project. It is important that the chosen SDLC can adapt to changing requirements and finalise details of implementation on the go, which is the advantage of the Agile framework.
- **Reduced Cost** As we will discuss in Section 5.5, this project has very limited budget and available staff (labour). Using Agile frameworks can reduce the cost of the project by having cross-functional teams and improved productivity.
- **User Experience** One of the competitive edges of the desired product is enhanced user experience and efficiency. Our client Susanto aims to provide more dynamic and user-friendly services, which requires user-oriented UI design and implementation of (possibly) streamlined business logic. All these features can be achieved in an Agile framework by a client-facing and collaborating team and optimized through continuous releases.

2.4 Business Value

The table below shows the business value distribution among stakeholders.

The project improves efficiency and experience of both clients and services. By providing a platform for request passing, order management and message sharing, it enables customers and shippers to communicate and negotiate over the order in a smooth and timely fashion. It can also be a workplace for the shipper and the collector to organise and schedule requests they have to fulfil.

Stakeholder	Financial Benefits	Non-Financial Benefits
Client/Sponsor	Profit	Delivering a dream product
Shipper	Profit	Expanded business types and improved communication & scheduling efficiency
Collector	Profit	Expanded business types
Product Owner	None	Product management skills
Scrum Master	None	Agile development skills
Developers	None	Front-end / Back-end development skills
Risk Manager	None	Risk management skills
Quality Assurance	None	Quality assurance experience
Customers	Reduced cost	Improved service

Table 4: Business value distribution

Because this IT project is such a fundamental part for building a dynamic and user-friendly shipment management system, it will also influence the box shipping market in Melbourne by introducing a new end-to-end service model and benefit a broader range of users.

2.5 Constraints

The main constraints of the project is time, budget, labour, and management skills. There are also additional constraints like technology stack.

- **Time** The project has to be delivered in roughly eight weeks, which is about 4-8 sprints in an Agile SDLC. Within such a short time, it is critical to schedule the project properly so that user stories are released in time.
- **Budget** The initiator and sponsor of this project, Susanto has very limited capital to start his business. Additional cost in developing is not allowed.
- **Labour** The Collector (Susanto's brother-in-law) in this project works part-time because he runs a Truck Fleet with regular business in furniture delivery. The developing team are master students. These labour can only dedicate limited time to this project compared to full-time staff.
- **Management skills** Although Susanto is familiar with box shipping business, he has no experience with software development management. Neither are the scrum team members knowledgeable in managing SDLCs.
- **Technology stack** The development team have some experience with Java and Python programming. But none of them is familiar with front-end frameworks and techniques.

3 Project Governance

3.1 Roles and Responsibilities

We use Agile as our project management framework, which enables us to have a small cross-functional scrum team with members taking on multiple roles.

There will have a Scrum Master, a Product Owner, one Full-stack Developer, one Back-end Developer, two Front-end developers, one Quality Assurance and one Risk Manager. Since our team consists of only four members. Each member will take on management roles in addition to their development role.

The roles and responsibilities in this project are listed as follows:

- **Product Owner**

One of our team members will take this role. He will in charge of communications with Susanto to define features of the software and decide detailed schedule of the team as a product owner.

In details, he will communicate with Susanto, define the features of the product as product backlog and estimate them. For every sprint planning meeting, he will choose some user stories as the goal of the sprint. These user stories forms some tasks as the sprint backlog. At the last of each sprint, he will check and analysis the result.

- **Scrum Master**

For the team member who take the role of scrum master, he will in charge of all management decision in the team, to make team fully functional and productive.

In details, every time a sprint backlog is defined, he will separate the workload to each team members. During the sprint, he will supervise the progress of each team members. At last, he will in charge of maintaining the sprint showcase and sprint retrospective.

- **Front end programmer**

They take charge of all front end including web design and request design. Since all of the team members are not familiar with this part, three of us will take this role. They will use wix to create a web app.

- **Back end programmer**

They take charge of all back end including building server environment and inner logic of the service. 1-2 team members will take this role. They will create a backend for web app.

- **Risk manager**

There are many possible risks for this project. The team member who take the role of risk manager should try to avoid the occurrence of these risks.

In details, he will manage the risk during the whole process. The management including estimate the probability and cost of risks, avoid risks during each sprint, and take some solution when some risks appear to reduce the loss.

- **Quality assurance**

When some pieces of codes have been done, team member with the role of quality assurance will check the program to achieve higher quality.

At the end of each sprint, he will estimate the quality of the result including quality and performance of code. At the very end, he will estimate the quality of the whole system (e.g. through some use cases).

3.2 Communication Plan

Since we are in an Agile framework, communication is critical to the productivity of development and quality of the product. We will be utilising both formal and informal communication channels to maximise message sharing and collaboration within the team.

Each sprint lasts for a week. For every sprint, we will hold three face to face meetings. One of these meetings will be used for sprint showcase, sprint retrospective and sprint planning. The rest of the meetings are used for developments.

3.2.1 Communication Channels

The formal and informal communication channels for the project are described as below.

- **Formal: face to face meetings**

We will hold the first meeting at Monday evening(6pm-8pm). In this meeting, we will first review the last sprint(sprint showcase and sprint retrospective). Then, we will talk about the planning for next sprint(sprint planning for next sprint).

The other two meetings will be held on Tuesday and Thursday evening (6pm to 9pm). These meetings are used for development. This will help the efficiency of the whole process.

- **Formal: emails**

Communication between clients and development team will be through emails. It is official and convenient. When needed, we can access the email box and check from records easily.

- **Informal: online communication**

Once some work hasn't been done in the meeting, team members may work at their home. At this time, team members may communicate online.

We will use messaging app for online communications. All of us are in a chatting group and able to chat with each other conveniently. Also, history of our communication will be saved.

3.2.2 Communication Matrix

To enhance communication within the project so that features are delivered in a timely manner, we have drafted the following Communication Matrix in table 5.

Sprint planning and Sprint retrospective are held in the same meeting. We will have two types of daily stand-up: off-line (face to face) when we have a development meeting(Tuesday and Thursday) and online stand-up for rest of days.

Communication Type	Purpose	Frequency	Owner	Distribution	Outcome
Ice-breaking meeting	Introducing team members; Introducing the project	One Time	Product Owner	Product Owner, Scrum Master, Developers, Quality Assurance, Risk Manager	Agenda, Initial product backlog
Sprint planning	Discussing how to achieve sprint goal	Before each sprint	Product Owner	Product Owner, Scrum Master, Scrum team	Sprint backlog
Daily stand-up	Daily updates of each member to avoid other unnecessary meetings	Daily	Scrum Master	Product Owner, Scrum Master, Scrum team	Updates to sprint backlog
Sprint retrospective	Reviewing what is and what is not working	After each sprint	Scrum Master	Scrum Master, Scrum team	Updates to product backlog

Table 5: Communication Matrix

3.3 Risk Management

Risks are listed in table 6.

Risk ID	Risk Type	Description	Prob.	Impact	Justification
1	Project	Schedule influenced	high	high	Team members are part time and have other assignment and tasks.
2	Project	Unfamiliar technologies	medium	high	Some technologies are new for developers.
3	Product	Budget	medium	medium	No funding for platform developing and testing.
4	Product	Difficulty in system migration	medium	medium	Free cloud services are usually limited, migration to other cloud services may required.
5	Business	Competition, similar services	low	high	We are part time developers and business rivals tend to be competitive.
6	Product	Security & Privacy	low	high	User credentials and sensitive information may be exposed via network

Table 6: Risk Impact Analysis Table

For risk 3, lack of funding may result in some problems. For instance, the system may only be tested in free cloud service. This may cause the system not been well tested for real information stream before release.

Risk ID	Trigger	Owner	Response	Resources Re-quired
1	team member absent from meeting	scrum master	Mitigate: adjust workload.	others' extra work time
2	fail to finish the work in scheduled time	scrum master	Mitigate: hold a brainstorming for the technology.	team members' extra time
3	error occurs when released	product owner	Avoid: try to consider all circumstances	careful work
4	system needs migration after release	product owner	Avoid: try to handle this when developing	careful work
5	news about other projects	product owner	Ignore: out of scope of the project.	-
6	suspicious user behaviour	back-end developer, front-end developer	Mitigate: use secure protocols	knowledge, developing time

Table 7: Risk Register Table

3.4 Technology

- **Front-end**

The front-end website would be built with Wix. Since every team member does not have much knowledge of front-end developing, we choose to use tools like Wix that is able to do layout without using HTML/CSS. Other possible choices include libraries like JQuery, Bootstrap or frameworks like Vue, Angular or React. All of these choices are not suitable given the experience of our team members.

- **Back-end**

The server is going to be implemented in Java and python. Java is the most popular server side language and each team member is familiar with Java programming. Python is a dynamic scripting language useful for setting up the back-end. There are also useful open-source libraries in python that help the team with agile delivery. Other popular back-end programming languages like PHP or NodeJS are not chosen because of unfamiliarity.

- **Database**

We will use MongoDB to store persistent data. We can store data in JSON format

with MongoDB which is flexible for potential changes comparing to relational DBMS like MySQL. In addition, MongoDB is free to use and can be easily scaled at need to ensure data security or availability if business grows fast.

- **Others**

We are going to use cloud computing resources to host server. Specifically, we will use virtual machines on Nectar Research Cloud. Since funding is limited, we could use Nectar while developing and move to other platforms like Amazon AWS or Microsoft Azure later in production. We will use Docker to deploy database services since it is easy and reliable across platforms.

3.5 Project Planning

User Stories:

1. As a *shipper*, I can acknowledge and modify shipping booking orders so that I can pick up orders from customers and ship them. - 3 story points
2. As a *collector*, I need to view shipping status and get notified so that I can arrange delivery. - 2 story points
3. As a *customer*, I can sign up so that I can use the service. - 1 story point
4. As a *customer*, I can log in so that I can place orders and view order status. - 2 story point
5. As a *customer*, I need to provide the shipping details so that the order can get to intended destination. - 4 story points
6. As a *collector* or *customer*, I want to be informed whenever shipping status changes so that I can get updated with orders. - 1 story point

We plan to finish the project in three sprints. Each sprint we aim to finish 2 user stories. Story points are estimated by searching time spent of similar project online and discussed among all developers. More sprints can be added if more request is received from product owner. We will use Trello board to track progress(<https://github.com/orgs/ATasteOfWix/projects/1>).

In the first sprint, we intend to complete user story 3 and 4. Since we need to configure environment and set up server first, the work load is high than story points indicates. Following table shows the time break down and number indicates hours.

User story 3 - customer sign up					
tasks	Mon	Tue	Wed	Thur	Fri
configure virtual machine	1				
set up database	1				
set up back-end server		1	3	1	
build front-end interface		2		2	1
define communication protocol			1		
define user account storage schema			1		
testing		1		1	1

User story 4 - user log in					
tasks	Mon	Tue	Wed	Thur	Fri
set up account for shipper and collector		1			
build front-end interface	1		2	1	
define back-end api	1		2		
testing				1	1

4 Project Execution, Monitoring and Control

4.1 Project Status: Non-teaching week

Up to the Non-teaching week, we have finished 1st sprint of the project and are in the developing stage of the 2nd sprint. In particular, user story 3 (customer sign up) and user story 4 (customer log in) are available and fully functional.

According to the Kanban board (Appendix A), Sprint 1 is finished on schedule. The burndown chart (Figure 2) also shows the project is being conducted at steady velocity within estimation.

As a scrum team, we have held two sprint planning meetings (Appendix C - F) ahead of each sprint to work out the user stories to implement, the decomposed tasks, the estimated time and task assignment. During the developing stage, a Gantt chart (Figure 1) has been used to monitor progress of each member and identify risks of delay.

At the end of Sprint 1, a sprint retrospective was held (Appendix E, F) to help the team review the design decisions and difficulties in the last sprint. We have identified a new risk (Section 4.1.3) to the project during the meeting and updated the Risk Management document (Section 3.3).

4.1.1 Process Related Artefacts

The first sprint went smoothly and all work has been done according to schedule. The Gantt chart of first sprint is shown in figure 1.

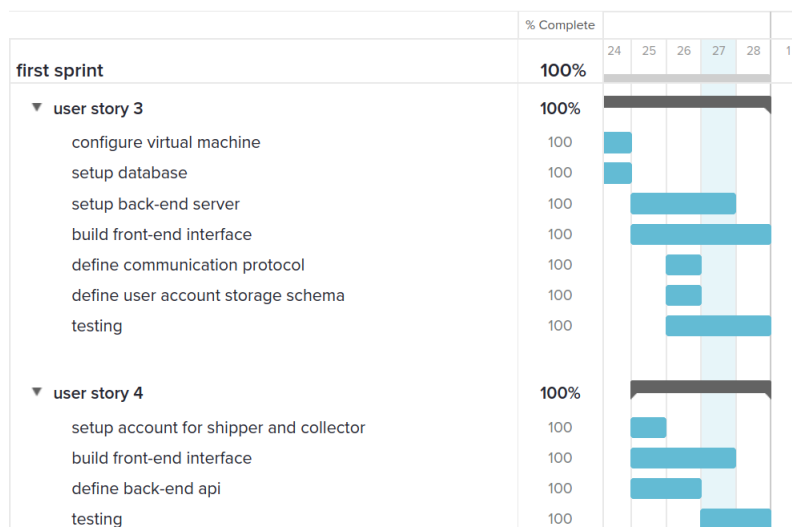


Figure 1: First sprint Gantt chart

The meeting agenda and details of first and second major spring meeting are included

in Appendix C to F. Personal time sheet of each member is included in Appendix G. Sprint review is conducted at the beginning of second sprint meeting. Usability of implemented features are tested and the scrum team is ready to conduct second sprint. The updated schedule is shown in Kanban board in Appendix A.

In the second sprint, we move on to complete user story 2 and 5. Following tables shows the low level task and time break down. Each number indicates hours.

User story 2 - collector view shipping status					
tasks	Mon	Tue	Wed	Thur	Fri
build front-end interface	2	1			
define back-end api		2			
connect to database		1	1		
testing		1		1	1
User story 5 - user place shipping order					
tasks	Mon	Tue	Wed	Thur	Fri
build front-end interface	2	2	1		
define back-end api	1	2	2		
connect to database			1	1	
define user order storage schema			1		
testing		1		2	2

The burndown chart after first sprint is shown in figure 2. All planned tasks are for first sprint are completed.

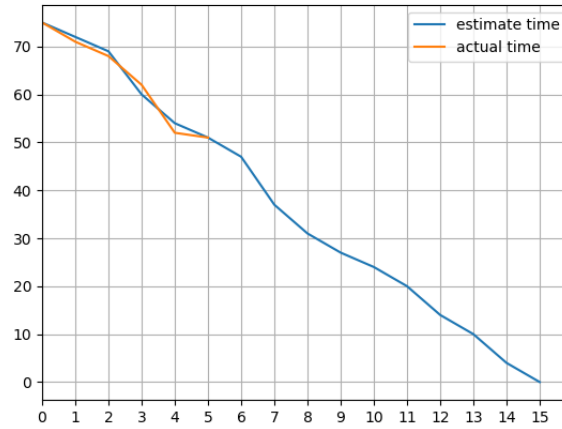


Figure 2: Burndown chart after first sprint

4.1.2 Product Related Artefacts

After first sprint, the log-in and sign-up functionality has been completed. A screen shot of front-end interface is shown in Appendix B. Users can use left side to sign up or use the right side to sign in. In sign up phase, user would get a warning if two passwords do not match or the email has already been used. User would also get a warning if trying to sign in with wrong email or password.

Our front-end implementation is wix-based, but in order to ensure data privacy, we decided to run the server's back-end and database on NECTAR, which IAAS service under our control. As shown in the figure 3, our back-end design follows the restful approach.

The front end and the back end interact via HTTP or HTTPS. Currently the finished service include:

- **sign up:** allow user create new account according to their name, password and email
- **sign in:** allow user login to the system by their email and password

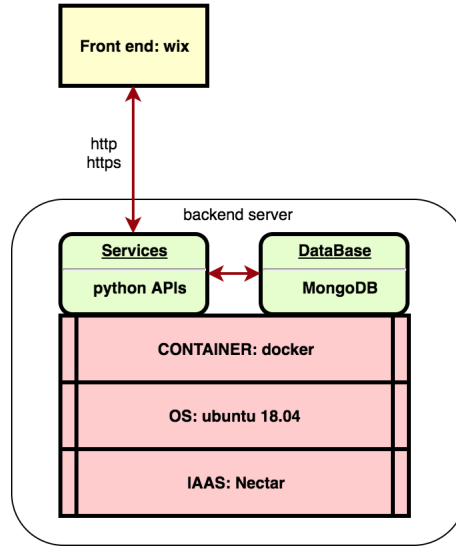


Figure 3: backend server

4.1.3 Risk Monitoring and Control

Of the risks we originally identified in section 3.3, we encountered Risk 1 (Schedule Impact), Risk 2 (Technology Unfamiliarity). We have also identified a new risk (Security) to the project.

Schedule Impact

The nature of Risk 1 is mainly that the scrum team consists of part-time students who are currently burdened with other subject assignments, especially near the end of semester. We try to avoid or mitigate most impacts on project schedule by planning meeting time in advance and decomposing the tasks into less correlated parts.

- **Planning Meeting Schedule in Advance**

Since all team members of the project have different subjects, we have tried to coordinate each member's timetable via instant chat tools to make sure we have a consistent time block committed to working on the project together. We have also posted room booking information to group chat as a memo for all members.

- **Decomposing the Task**

Although Agile framework emphasises teamwork and communication, where tasks can be undertaken in an asynchronous manner, we have tried to decompose them into individual parts so that team members can finish them in their individual time.

For example, during sprint meetings, we will try to brain-storm front-end and back-end frameworks and APIs most suitable for the project. But in the implementation stage, tasks such as database modelling, UI design can be conducted by individual developers until we implement interactive functionality. This way, we can minimize the length of sprint meetings to fit most members' timetable while not impacting the productivity of collaboration.

Technology Unfamiliarity

During the first two sprints, unfamiliarity with front-end frameworks and technologies has caused sudden delays of task delivery.

We have tried to avoid such risks by taking advantage of Wix. We have also mitigated the risk of delays by having three members on the front-end developing.

Security and User Privacy (New Risk)

In the 1st sprint (customer sign up and log in), we used explicit user name and password representations in client-server communication, which brings up a security risk to the project as user credentials might be exposed to unauthorized parties.

To mitigate this risk, we have planned to switch to more secure protocols, e.g. HTTPS in stead of HTTP, in later sprints.

Appendices

- Appendix A - Kanban
- Appendix B - Screenshots
- Appendix C - First meeting agenda
- Appendix D - First meeting minutes
- Appendix E - Second meeting agenda
- Appendix F - Second meeting minutes
- Appendix G1 - Personal timesheet
- Appendix G2 - Personal timesheet
- Appendix G3 - Personal timesheet
- Appendix G4 - Personal timesheet

Appendix A - Kanban board after first sprint

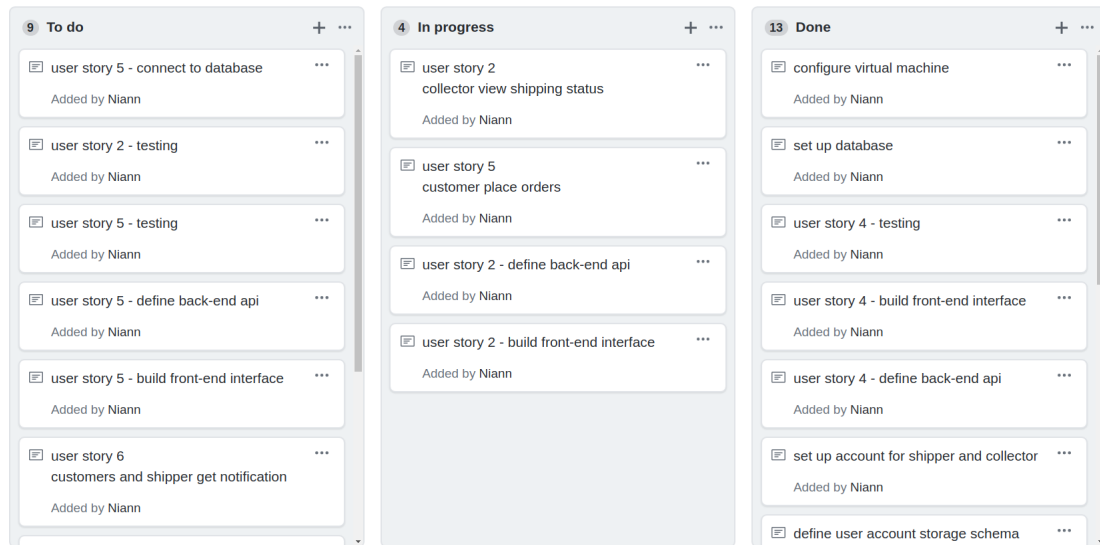


Figure 4: Kanban board after first sprint

Appendix B - Screen shots of login interface

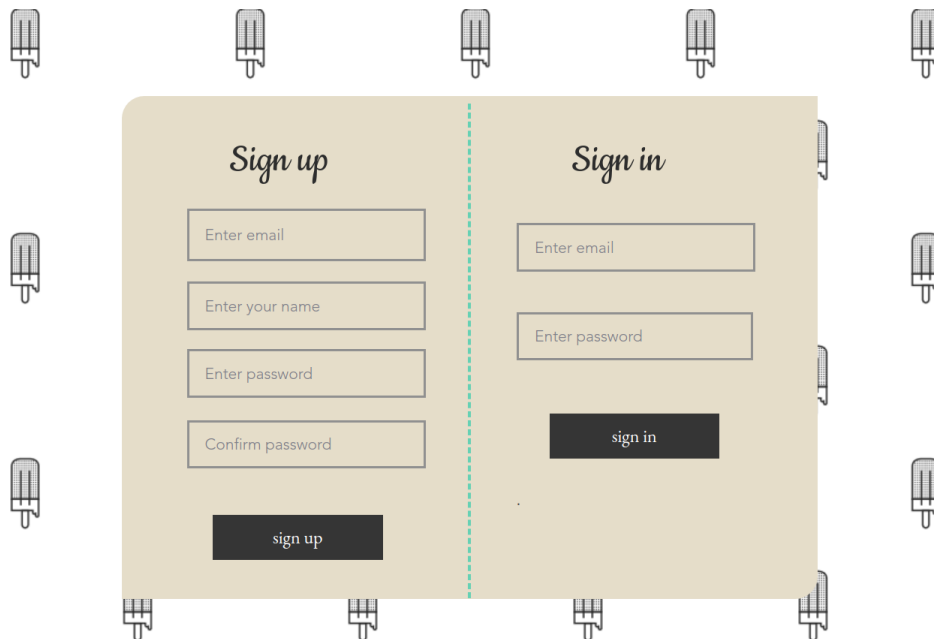


Figure 5: Screen shot of login page

Appendix C
Team CSR_04_5 Agenda
Date: *Wed 19 sept*

1. Apologies

None

2. Minutes of Previous Meeting

Ice break meeting. We decide the agile as our SDLC. Roles and responsibilities are determined for each team member. Other things like risks, product backlog, features and user stories (and their user points).

3. Reports

a. Scrum Master's Report

Low level task decomposition and time estimation, work load distribution.

4. Next activity for the assignment

First sprint and its related meetings

5. Other tasks

1. Choose technologies for each task
2. Generally, determine some details for web app
3. Adjust PMP based on feedback

Appendix D
Team CSR_04_5 Agenda
Date: Thu 27 sept

1. Apologies

None

2. Minutes of Previous Meeting

Sprint backlog for first sprint. This may include a general design for web app, choose for technologies we use and select a cloud service and framework for system.

3. Reports

a. Scrum Master's Report

Conclusion for sprint 1.

b. Quality assurance's Report

Quality of first sprint's result.

c. Front end's Report

Sharing some useful developing experience

4. Next activity for the assignment

Sprint showcase and retrospective sessions for first sprint. Sprint planning for second sprint

5. Other tasks

1. Design backend logic
2. Design of web app
3. Review PMP for submission
4. Ordering meeting room

Appendix E – First Meeting Minutes

Meeting of: CSR_05_4
Held at: ERC
Date: Wed 19 Sept
From: 6.30 pm

Opening:

The regular meeting of the CSR_05_4 was opened at 6.30 pm on wed 10 Sept in ERC.

Present:

Boheng Luan
Linrong Chen
Ruihan Zhang
Zeyu Huang

Approval of agenda:

The agenda was unanimously approved as distributed.

Approval of minutes

The minutes of the previous meeting were unanimously approved as distributed.

Project Plan: Week 9, Date 19/09/18					
Task	Resources/ Attendees	Estimated Time	Actual Time	Completed	Comment
Team meeting to prepare first sprint	All	1 hours	0.5 hours	yes	Make sure everyone understands his role.
Sprint planning	All	1 hour	1 hour	yes	User story 3 and 4
Technologies chosen Design for web app	All	1 hour	1.5 hours	no	Still some details not determined
Time for next meeting	All	0.5 hours	0.5 hours	yes	New place is found

Conclusions

Roles and responsibilities are applied to all team members. For first sprint, we choose user story 3 and 4 as they are the easiest. In this meeting, wix, mongoDB and python are chosen to be the main technologies used for the project as they are quite simple. Most of the time are used to discuss the structure and design for the web app. Also, we agree that most of the time in sprint one will be used for learning new skills.

Next meeting

The next general meeting will be at 4 pm on Thu 27 Sept at unimelb Phys.

Minutes submitted by: Ruihan Zhang
Approved by: Linrong Chen

Appendix F – Second Meeting Minutes

Meeting of: CSR_05_4
Held at: ERC
Date: Thu 27 sept
From: 4 pm

Opening:

The regular meeting of the CSR_05_4 was opened at 4 pm on Thu 27 Sept in unimelb phys.

Present:

Boheng Luan
Linrong Chen
Ruihan Zhang
Zeyu Huang

Approval of agenda:

The agenda was unanimously approved as distributed.

Approval of minutes

The minutes of the previous meeting were unanimously approved as distributed.

Project Plan: Week 9, Date 27/09/18					
Task	Resources/ Attendees	Estimated Time	Actual Time	Completed	Comment
Sprint showcase and retrospective sessions for sprint 1	All	1 hours	1.5 hours	yes	Basic framework has been built, user story 3 and 4 released
Sprint planning for sprint 2	All	1 hour	1 hour	yes	User story 2 and 5
Time for next meeting	All	0.5 hours	0.5 hours	yes	-

Conclusions

Main part for this meeting is to share the experience for the skills learned in the first sprint. Besides, user story 3 and 4 are released at cloud service. This means that a basic structure for the project has been built. For next sprint, we determined the inner logic for background and design the basic structure for web app. For this sprint, we need to consider what an order like for this system. At last, a new place for meeting room has been booked.

Next meeting

The next general meeting will be at 4 pm on Thu 4 Oct at unimelb Giblin.

Minutes submitted by: Ruihan Zhang
Approved by: Linrong Chen

Appendix G1: Personal Timesheet

Member Name: *Linrong Chen*
 Team name: *CSR_04_5*
 Tutor: *Rajesh Chittor Sundaram*
 Date: *27th, Sept*

Date	Activity	Planned	Actual
Sun 9 Sept	Reading Assignment spec	1 hour	1 hour
Tue 11 Sept	Ice-breaking	1 hour	1 hour
Fri 14 Sept	Sprint planning	3 hours	3 hours
Sat 15 Sept	Learning Front-end	2 hours	2 hours
Mon 17 Sept	Writing PMP 3.4, 3.5	2 hours	3 hours
Wed 19 Sept	Sprint meeting	2 hours	2 hours
Sat 22 Sept	Building website	3 hours	3 hours
Sun 23 Sept	Updating PMP 3.5	1 hour	1 hour
Wed 26 Sept	Online sprint meeting	1 hour	1 hour
Thu 27 Sept	Sprint planning	3 hours	5 hours
Fri 28 Sept	Building website and testing, updating PMP	3 hours	6 hours

Appendix G2: Personal Timesheet

Member Name: Boheng Luan
Team name: CSR_04_5
Tutor: Rajesh Chittor Sundaram
Date: 29th, Sept

Date	Activity	Planned	Actual
Wed 10 Sept	Reading Assignment spec	1 hour	2 hour
Wed 11 Sept	Ice-breaking	1 hour	1 hour
Wed 14 Sept	Sprint planning	3 hours	3 hours
Wed 19 Sept	Sprint meeting	2 hour	2 hour
Wed 26 Sept	Online sprint meeting	1 hour	1 hour
Thu 27 Sept	Sprint planning & development	3 hours	5 hours
Thu 27 Sept	Backend development	4 hours	5 hours
Fir 28 Sept	Sprint meeting & development	3 hours	6 hours

Appendix G3: Personal Timesheet

Member Name: *Ruihan Zhang*
Team name: *CSR_04_5*
Tutor: *Rajesh Chittor Sundaram*
Date: *29th, Sept*

Date	Activity	Planned	Actual
Mon 10 Sept	Reading Assignment spec	1 hour	2 hour
Tue 11 Sept	Ice-breaking	1 hour	1 hour
Wed 12 Sept	Writing PMP section 3.1 3.2 3.3	4 hours	4 hours
Fri 14 Sept	Sprint planning	3 hours	3 hours
Wed 19 Sept	Sprint meeting	2 hour	2 hour
Sun 23 Sept	Update PMP section 3.1 3.2 3.3	1 hour	3 hours
Wed 26 Sept	Online meeting	1 hour	1 hour
Thu 27 Sept	Development	3 hours	5 hours
Thu 27 Sept	Sprint showcase, retrospective and planning	2 hours	2 hours
Fri 28 Sept	Ckeck agenda and minutes for meetings	1 hour	1 hour

Appendix G4: Personal Timesheet

Member Name: *Zeyu HUANG*
Team name: *CSR_04_5*
Tutor: *Rajesh Chittor Sundaram*
Date: *29th Sept*

Date	Activity	Planned	Actual
Sun 9 Sept	Reading Assignment spec	1 hour	1 hour
Tue 11 Sept	Ice-breaking	1 hour	1 hour
Wed 12 Sept	Write PMP Section 2.1, 2.3, 2.4, 2.5	3 hours	5 hours
Thu 13 Sept	Write PMP Section 2.2	2 hours	2 hours
Fri 14 Sept	Sprint planning	3 hours	3 hours
Wed 19 Sept	Sprint meeting	1 hour	2 hours
Thu 20 Sept	Update PMP Section 2.1, 2.3	1 hour	1 hour
Tue 25 Sept	Update PMP Section 2.2	0.5 hour	1 hour
Wed 26 Sept	Sprint meeting (Online)	1 hour	1 hour
Thu 27 Sept	Sprint planning	3 hours	5 hours
	Create PMP Section 4.1.3, Update PMP Section 3.3	2 hours	2 hours
Fri 28 Sept	Sprint meeting	3 hours	6 hours