Project Definition Document

Recipick

Group 8

Date 9/05/2019

Version 2.0

Authors Mohammad Khan
Nithesh Koneswaran
Max Krawiec
Clyde Leal
Sze Lee
Pavlos Lekkas
Oscar Levy

Document Revision History

Date	Revision	Description
18/03/2019	1.0	Initial version.
20/03/2019	1.1	Corrected an error in the project charter.
25/03/2019	1.2	Identified non-essential high-level requirements and moved them to extended goals following a change request.
09/05/2019	2.0	Updated risks and RoE following feedback from sponsor. HIgh-level requirements were further adapted to scope. Added the goal oriented produced roadmap and updated section 4.

Table of Contents

1 Introduction

This Project Definition Document (PDD) is an overview of the project plans for the Android application 'Recipick' developed by Group 8 for the University of Surrey's Year 2 Software Engineering Project module. It aims to assist the members of Group 8 in garnering a mutual awareness of the project charter, team charter, work plan, risk analysis and change management plan to facilitate the smooth development of 'Recipick'.

2 Project Charter

Project Name	Recipick		
Project Sponsor	François Dupressoir	**Project Approval Date**	26/02/2019
Project Manager	Oscar Levy	**Last Revision Date**	09/05/2019

2.1 Problem Statement

Although most university students and people who live alone are aware of the merits of healthy eating, the majority still opt to consume unhealthy, processed food for the convenience (1) and due to a lack of cooking expertise (2). Poor nutrition can lead to a myriad of health issues ranging from osteoporosis, heart disease to depression (3).

2.2 Project Overview

An Android application to solve poor dieting habits that simplifies the cooking process, finding recipes based on a user's currently owned ingredients and preferences. By home cooking meals, one can eat food that is more nutritious than ready meals towards a healthier life (4).

2.3 Goal Statement

To make cooking a simpler, less intimidating and more palatable experience, serving as a benefit to society and a catalyst for improving the world's health and wellbeing.

2.4 High-Level Requirements

The following is a list of requirements that Recipick must conform to meet project objectives:

1. To track the ingredients a user has and suggest recipes based on those ingredients.
2. To allow users to add missing ingredients for a recipe to a shopping list.
3. To inform users of the nearest supermarkets to purchase missing ingredients utilising the user's location (i.e. with the device's GPS sensor).
4. Shop planning, where users can generate a shopping list of ingredients from a recipe.
5. To add new recipes which can be stored privately or shared publicly.
6. To allow moderators to process and review user recipes before they are published.
7. To add other users' recipes to their favourites to view at a later date and while offline.
8. Ability to take and share pictures of their freshly cooked meals from following recipes.
9. To ensure that a subset of Recipick's functionality continues to operate in spite of an unstable Internet connection, i.e. when a user continuously reconnects and disconnects to the application due to a poor and unstable Internet signal.

2.5 Project Scope

2.5.1 In Scope

1. To create an Android mobile application that will make cooking more approachable.
2. To design a friendly brand image and an intuitive user interface for the application.
3. To promote a healthy eating lifestyle and improve the world's nutrition.
4. To have a candidate release version available by 22 April 2019.

2.5.2 Out of Scope

1. It is up to the discretion of users to post recipes that are nutritious and high-quality.
2. Providing and showing the fastest route to nearest supermarkets.

2.5.3 Extended Goals

These extended goals are to be tackled after the successful initial implementation of Recipick:

1. To be able to discover the trending and most popular recipes of Recipick.
2. The ability to scale ingredients for a specific number of servings and to accommodate a reduced ingredient if the user does not have the required amount of an ingredient.
3. Recipe filtering in terms of budget, cooking duration and diets. Users may exclude specific ingredients and any cooking appliances that the user may not have.
4. To have a cookbook catalogue, where professional chefs may sell recipe packs.
5. To have advertising and brand deals with ingredient companies.
6. To have video instructions, along with paid cooking shows by professional chefs.
7. To grow into a social network with fully-featured profiles and messaging capabilities.
8. To have a web browser version that is integrated with the Android application.
9. To provide support for languages other than English.

2.6 Deliverables and Milestones Schedule

Item	Type	Start Date	Target Date	Status
Project proposal	Deliverable	18/02/2019	22/02/2019	Completed
Work plan (Agile)	Milestone	27/02/2019	02/03/2019	Completed
Project pitch	Milestone	12/03/2019	15/03/2019	Completed
Project Definition Document Draft	Deliverable	22/02/2019	18/03/2019	Completed
Back-end infrastructure with testing	Milestone	07/03/2019	12/04/2019	Completed
Front-end infrastructure with testing	Milestone	07/03/2019	12/04/2019	Completed
Integration with testing	Milestone	12/04/2019	17/04/2019	Completed
Recipick beta version with user acceptance testing	Deliverable	17/04/2019	22/04/2019	Completed
Project demo	Milestone	05/05/2019	10/05/2019	Completed
Recipick release candidate with testing	Deliverable	22/04/2019	13/05/2019	Completed
Final audit report	Deliverable	18/03/2019	13/05/2019	Completed

2.7 Constraints

1. Recipick must function fully on the Asus ZenPad 8.0.
2. Recipick must use two sensors from the device.
3. Recipick must have partial functionality if there is an intermittent Internet connection.
4. Recipick must have a server-side data gathering component.
5. Recipick must be GDPR compliant, securely processing and handling user data.

These extended constraints follow the fulfilment of the project's extended goals:

1. To have rules against plagiarism, discrimination, abuse, harassment and illegal content established in the Terms of Service as Recipick grows as a social network.
2. Recipick's website version must be cross-browser compatible.
3. Anti-piracy measures (e.g. no recording, no copying and pasting) must be implemented for the paid contents of the application.

2.8 Budget and Estimated Cost

Total budget of £200, funded equally by all team members, almost all of which is invested into Internet access and electrical bills due to the utilisation of free tools to develop this project:

Item	Estimated Cost
Internet access	£50
Electrical bills	£100
Unforeseen expenses	£50
Total cost	£200

3 Team Charter

3.1 Project Description, Scope and Constraints

The team will build, test and deploy Recipick, an Android application which aims to overcome malnutrition by simplifying cooking, finding recipes based on a user's owned ingredients and preferences. Recipick will comply with stakeholder expectations by using two device sensors: the camera will be used for sharing pictures of freshly cooked meals and the location sensor will be used to locate the nearest shops from where they can purchase missing ingredients. Recipick will allow users to save favourite recipes, creating a subset of functionality which works with an intermittent Internet connection. Available ingredients will be processed server-side in the required data-gathering module. Detailed high-level requirements can be viewed in section 2.4 of this document. This project began on 18 February 2019 and is projected to end by 13 May 2019, with a candidate release of Recipick available by then.

3.2 Team and Roles

The team is composed of the following members alongside the specific areas of the project they were assigned to focus on:

- Mohammad Khan: Quality assurance, testing and DevOps.
- Nithesh Koneswaran: Front-end design.
- Max Krawiec: Back-end design.
- Clyde Leal: Front-end functionality.
- Sze Lee: Documentation and planning.
- Pavlos Lekkas: Back-end functionality.
- Oscar Levy: Project manager.

3.3 Evaluation Criteria

- Evaluation of High Level Requirements
  - Each requirement and constraint listed in Section 2.4 and 2.7 must be fulfilled, tested for code and user accepted.
  - 100% of the code must pass unit tests.
  - 70% of users should be able to use the application with minimal explanation (early adopters and early majority in market penetration diagram[1]).

[1] The 5 Stages of Technology Adoption | OnDigitalMarketing.com [Internet]. On Digital Marketing. 2019 [cited 17 March 2019]. Available from: https://ondigitalmarketing.com/learn/odm/foundations/5-customer-segments-technology-adoption/

    - Security and privacy (GDPR) concerns must be addressed and reported on throughout the project.
- Evaluation of team
    - Team cohesion will be measured and assessed by end of project with evaluation questionnaires for others to self improve on.
- Evaluation of deliverables and timing
    - The product must follow the milestones and deliverables set in Section 2.6, aiming to hit 100% of target dates throughout.

3.4 Rules of Engagement

- Meetings and discussions
    - Meetings occur on a weekly basis according to Agile Scrum methods and are structured in the following way in order to increase efficiency:
        - Meetings are planned in advance, generally by the project manager, assisted by the team member(s) responsible for the matter of focus for a certain week.
        - The meeting starts with a debrief from the past week's Sprint, including task completion and general morale.
        - The next Sprint is then planned and objectives for each member are updated.
        - The meeting is ended by closing remarks.
        - Repeated failure to attend meetings will be notified to the project sponsor by the project manager as this could jeopardise the development of Recipick.
- Decision making and protocols
    - Decision making is done on a consensus basis, generally revolving around the opinion of the team member most responsible for the direction to be decided.
    - Repeated unilateral decisions and non-respect of protocol will be regarded as a threat to team harmony and Recipick's development. The project manager may involve the project sponsor in such a case.
- Self-assessment and improvement
    - Each team member holds a self-performance evaluation at the end of each project week, highlighting their contribution in terms of strengths and weaknesses.
    - Team members are expected to complete their weekly tasks to the best of their ability while being respectful (but critical) of each other's work in order to achieve the best possible results on the project.
    - Team members are expected to respect deadlines and communicate effectively through tools which have been agreed should be used.
    - Repeated failure to meet deadlines will create a need for the project manager to try and mitigate this situation. Several remedies may be employed such as redelegating tasks, with the most severe remedy being the direct involvement of the project sponsor in the issue.

3.5 Project Management

- Procedures and change management
    - The team uses forms of Agile methodology, more precisely Scrum. Objectives are set on a weekly "Sprint" basis and assessed and discussed during weekly meetings.
    - Outstanding tasks are stored in a backlog and used to populate Sprints each week.
    - If tasks are not completed during a Sprint, they are pushed back to the next one and marked as urgent.
    - Efficiency is emphasized by not requiring all members to attend all meetings.

- Records of every meeting are recorded in a “minutes” style for every meeting.
- Procedures relating to change management are exhaustively documented in Section 6.
- Documents are held on Google Drive for easy access and collaboration.
- Communications between members are made through Slack channels.
- Communication with supervisors and sponsors is usually done through email.
- Scrum and effective task planning management are done through Trello boards.
- Documentation of code including inputs, outputs and expected behaviour should be written in a separate document.
- Commenting must follow the coding conventions recommended by Oracle[2].
- Version control is done through Gitlab in order to have effective peer review and secure code storage along with practical version control.

- Record Keeping procedures
  - Records of the project are placed in Google Drive and include (but are not limited to):
    - Historical versions of documentation, planning and definition charters used to monitor change in tools and decisions.
    - “Minutes” records of every meeting in order to support future decisions and vision by being able to refer to past discussions and meetings easily.
    - Presentations and pitches.
    - Sheets relating to planning and deliverables.
- Group Development procedures
  - Code is to be peer reviewed before each release amongst teams, meaning that back-end coders will review each other’s code while front-end coders will do the same. Code will finally be reviewed by team members in charge of Quality Assurance and deployed to GitLab.
  - Testing is done on a unit basis, therefore making sure that every piece of functionality is valid and behaves as expected.

---

4 Work Plan

Recipick is developed using the Agile approach of software development with the Scrum framework. This allows for flexible scheduling suited to fit the group’s workflow.

A work breakdown structure (WBS) was created to determine the initial breakdown of the necessary implement Recipick and for an concise, clear overview of the project and to set expectations for the final product. This WBS may be viewed in Appendix C and its intended audience are the project manager and the project sponsor.

The group primarily utilises a board on Trello to keep track of tasks. The board is populated with user stories derived from the project’s high-level requirements as defined in section 2.4, aligning with Scrum practices. This backlog is shown in Appendix D and its intended audience

[2] Oracle. How to Write Doc Comments for the Javadoc Tool. [Internet] Oracle. [cited 17 March 2019]. Available from: https://www.oracle.com/technetwork/java/javase/documentation/index-137868.html

are the members of Group 8 to aid in the planning and development of the project. Following every start of a sprint, user stories are moved from the backlog to the current sprint's 'progress' section, and when complete, will move to the sprint's 'done' section. The user stories to be moved are determined during meetings and thus do not have deadlines. However, it can be expected that the team aims to implement every summary task of the WBS per project week and expect to complete the beta version of Recipick by 22 April 2019.

A formalised project schedule based on the Trello board was created using a spreadsheet and can be viewed in Appendix E. This spreadsheet contains more detailed information about the tasks to be completed with dependencies listed as well as milestones to be achieved.

As of 14 April 2019, the WBS and formalised project schedule were discovered to not be as effective as hoped and thus were not kept up to date. The team preferred to use the Trello board due to its ease of use and effectiveness in planning out scrum sprints. As a result, a goal oriented product roadmap was created which clearly and concisely outlines all of the tasks that have to be completed by each quarter of the project duration, inclusive of all the documentation and software engineering aspects of developing Recipick. This allows the project manager and members of the team to have an overall view of all the tasks that this project entails so that the team ensures that they are on track, and also for the project sponsor to track the progress of the development of Recipick. This roadmap may be viewed in Appendix F and its intended audience are the project manager and project sponsor.

5 Risk Analysis

Risk management is essential to ensure project management success. For the purposes of this project, several risks were identified as outlined in Table 5.1 and analysed according to their probability of occurrence, impact (Low, Moderate, High) and severity. Risks were then evaluated according to their rank and risk mitigation strategies (avoid, accept/ignore, contain, contingency) were determined to prevent, control the risk or minimise the impact.

Table 5.1: Risk Assessment Matrix

Event	Prob	Impact	Severity	Rank	Risk Mitigation Plan
Unsafe sensitive data storage	High	High	Very High - Unacceptable	1	Check how APIs handle features such as keyboard press caching, application backgrounding, Intermediate data and analytics sent to 3rd parties (Contingency)
Team member pulled	Low	High	High	2	Discuss with sponsor regarding replacement
Unsafe sensitive data transmission	Med	High	High	3	Using HTTPS for sensitive data transmission (contain)
Choosing the wrong business model	Med	High	High	4	Research on what users need and want (contain)
Platform Fragmentation	Med	High	High	5	Testing software across different devices to make sure the user experience is consistent across devices and versions of OS (Contingency)

Insecure Data Storage	**Med**	**High**	**High**	6	Not to store data unless absolutely necessary. Never store credentials on the phone file system. Storage or caching of information using encryption. Protection against jailbreak or root exploit (contingency)
Taking too long on key tasks	**Med**	**Med**	**Moderate**	7	Time management and prioritization (Contain)
Improper Session Handling	**Med**	**Med**	**Moderate**	8	Session Management be done in secure way and session destruction at server side after logout (contingency)
Not serving anything unique	**Med**	**Med**	**Moderate**	9	Research on food and app market (contain)
Lack of Binary Protections	**Med**	**Med**	**Moderate**	10	Use of binary hardening techniques and protect against common exploits to avoid confidential data theft (contain)
Poor Authorization and Authentication	**Low**	**High**	**Moderate**	11	All authentication requests are performed server-side (contain)
No Guaranteed Google Involvement in Android for the Long Term	**Low**	**High**	**Moderate**	12	Use of other OS (avoid)
Damage to market reputation	**Low**	**High**	**Moderate**	13	Predict market trends (accept)
~~Proposal rejection~~	~~**Low**~~	~~**Med**~~	~~**Low**~~	~~14~~	~~Change proposal (avoid)~~
~~Software late (> 1 week)~~	~~**Med**~~	~~**Low**~~	~~**Low**~~	~~15~~	~~Ask for extra time (avoid)~~
Going over budget	**Low**	**Med**	**Low**	16	Reduce spending (accept)
Bad UI/UX	**Low**	**Med**	**Low**	17	Use of responsive design and no plagiarism data (contain)
Storage cost higher than expected	**Low**	**Med**	**Low**	18	Backup solutions planned (contain)

6 Change Management Plan

This section provides guidance for members on how approaching project change requests.

6.1 Change Management Procedure

The following steps illustrate the process of a general change request:

Step 1: Fill up the change request form and submit it prior to or during the next nearest meeting, either physically or as an email attachment.

Step 2: Form contents will be discussed and evaluated by the project manager and/or sponsor.

Step 3: Update the change management log to reflect this newly requested change.

Step 4: If implementing the changes is feasible and deemed necessary for the success of the project, update log for approval and implement change as soon as possible.

Change requests are assessed against the project charter and risk assessment matrix to determine whether implementing the change will potentially impact the development of the project such as in terms of costs and scheduling.

If the benefits of implementing the change outweigh its impacts and have been determined to increase the success rate of the project, the change request shall be approved and implemented.

The change request form can be found in Appendix A of this PDD.

6.2 Change Management Log

#	Submitter	Date Raised	Date Required	Change Request Description	Priority	Decision
1	Sze Ying Lee	20/03/2019	25/05/2019	Move non-essential goals from high-level-requirements to extended goals so that the team may focus on other areas of the project.	Medium	Approved by project manager.
2						

References

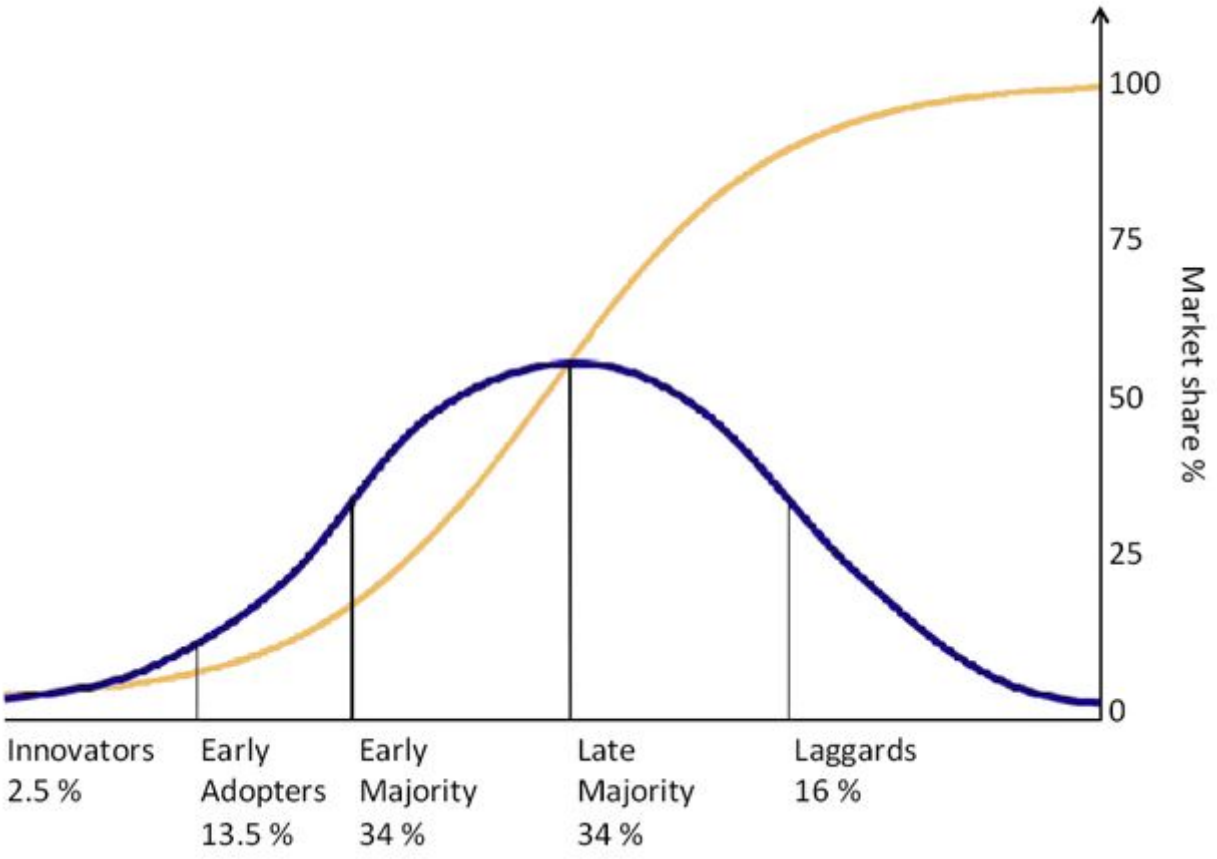
1. Abraham S. College students eating habits and knowledge of nutritional requirements. [Internet]. Allied Academies. 2018 [cited 13 March 2019] Available from: http://www.alliedacademies.org/articles/college-students-eating-habits-and-knowledge-of-nutritional-requirements-9188.html
2. Derla K. Living Alone Linked To Poor Diet: Single People Eat Fewer Fruits, Veg And Opt For Ready-Made Meals. [Internet]. Tech Times. 2015 [cited 13 March 2019] Available from: https://www.techtimes.com/articles/102847/20151104/living-alone-linked-to-poor-diet-single-people-eat-fewer-fruits-veg-and-opt-for-ready-made-meals.htm
3. SA Health. The risks of poor nutrition. [Internet] SA Health. 2012 [cited 13 March 2019] Available from: https://www.sahealth.sa.gov.au/wps/wcm/connect/public+content/sa+health+internet/healthy+living/is+your+health+at+risk/the+risks+of+poor+nutrition
4. Clay X. Time to rethink ready meals? Posh TV dinners put to the test. [Internet] The Telegraph. 2016 [cited 13 March 2019] Available from: https://www.telegraph.co.uk/food-and-drink/features/time-to-rethink-ready-meals-posh-tv-dinners-put-to-the-test/
5. On Digital Marketing. The 5 Stages of Technology Adoption. [Internet] On Digital Marketing. 2019 [cited 17 March 2019]. Available from: https://ondigitalmarketing.com/learn/odm/foundations/5-customer-segments-technology-adoption/
6. Oracle. How to Write Doc Comments for the Javadoc Tool. [Internet] Oracle. [cited 17 March 2019]. Available from: https://www.oracle.com/technetwork/java/javase/documentation/index-137868.html

Appendices

Appendix A: Change Request Form

PLEASE FILL IN ALL FIELDS:	
Change Request #	
Submitter Name	
Date Submitted	
Date Required	
Change Request Description	
Justification	
Priority (tick one)	❏ Low ❏ Medium ❏ High ❏ Compulsory
FOR PROJECT MANAGER AND/OR PROJECT SPONSOR USE ONLY:	
Impacts	
Decision (tick one)	❏ Approved ❏ Approved with amendments ❏ Rejected
Amendments	
Decision Date	

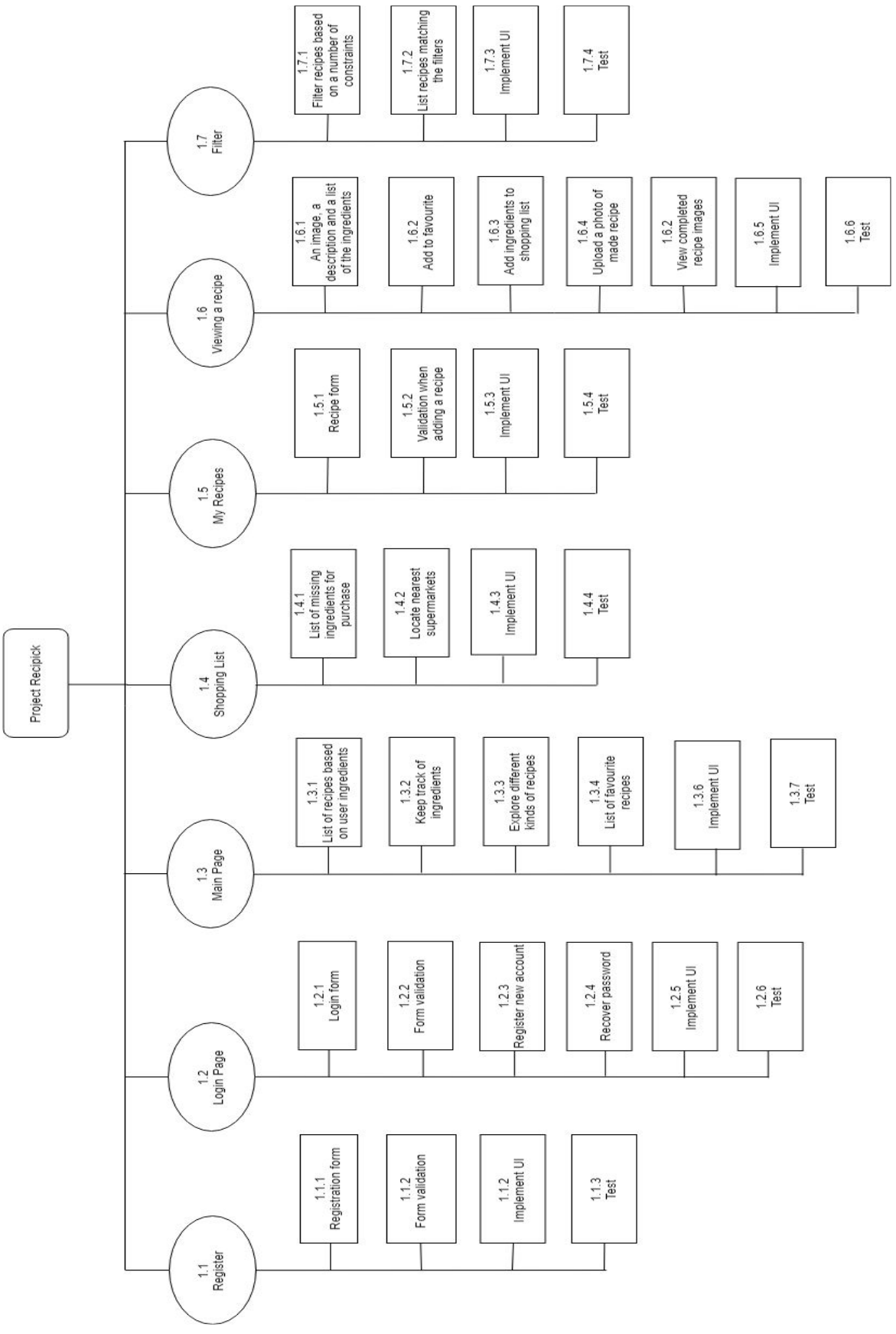
Appendix B : Distribution of Market Penetration



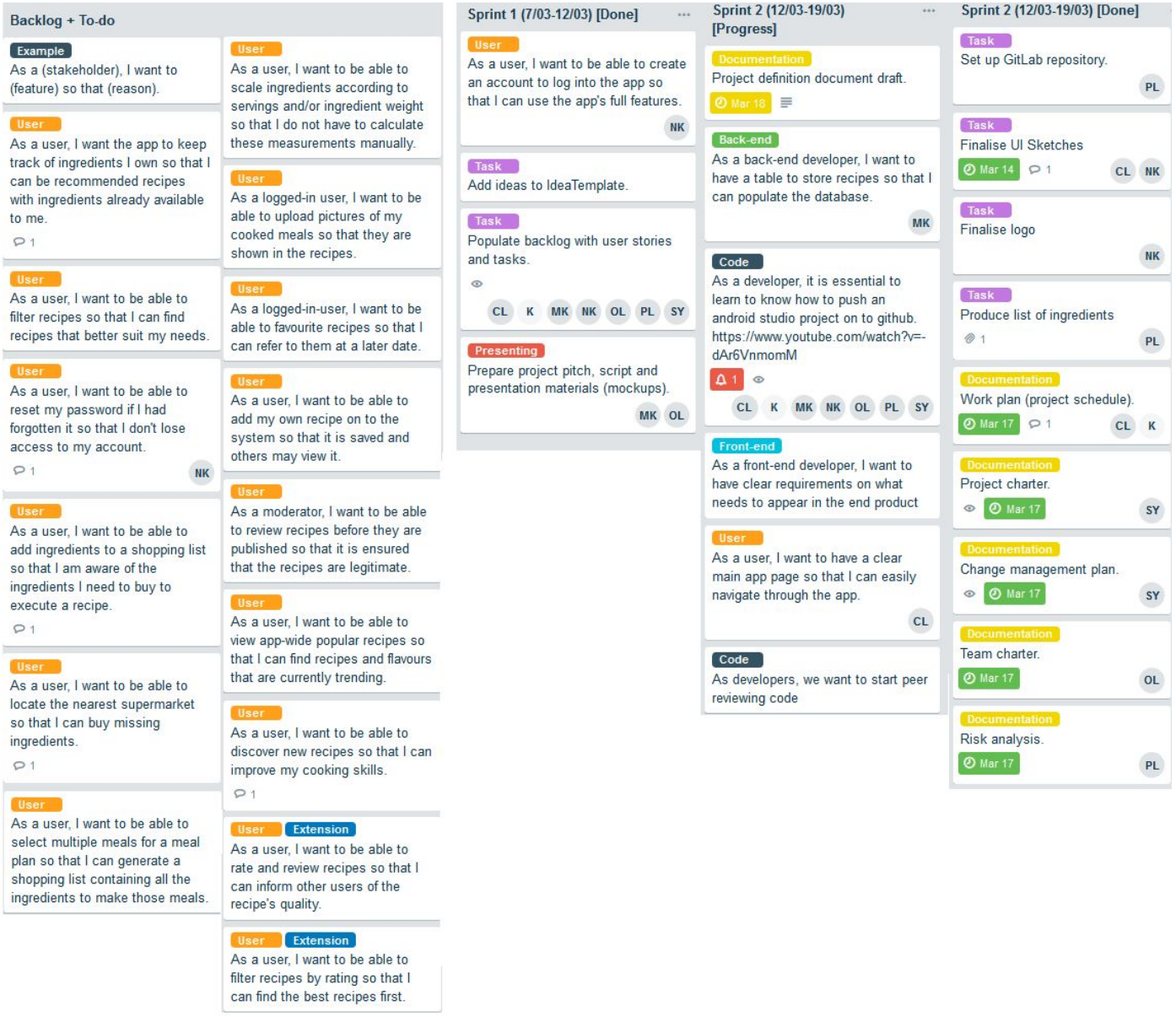
Source :

- The 5 Stages of Technology Adoption | OnDigitalMarketing.com [Internet]. On Digital Marketing. 2019 [cited 17 March 2019]. Available from: https://ondigitalmarketing.com/learn/odm/foundations/5-customer-segments-technology-adoption/

Appendix C: Work Breakdown Structure



Appendix D: Backlog and Scrum Sprint (Serves as documentation WBS)



Appendix E: Project Schedule

Project Name	Project Manager	Start Date	End Date
Recipick	Oscar L	18/02/2019	13/05/2019

Task ID	Feature/Task	Responsible	Start	Finish	Duration (days)	Dependencies	Status	Comments
Sprint 1			07/03/2019	12/03/2019	5		Complete	
1	Design UIs	Nithesh K, Clyde L				6	Complete	
2	Register	Nithesh K				1, 6	Complete	
3	Login	Nithesh K				1, 6	Complete	
4	Main Page	Clyde L				1, 6	Extended	Extended to Sprint 2 - after discussion in the meeting, changes need to be made.
5	Project pitch	Oscar L, Nithesh K				1, 6	Complete	
6	Populate Backlog	Everyone					Ongoing	
Sprint 2			12/03/2019	19/03/2019	7		In progress	
7	Finalise UI designs	Nithesh K, Clyde L				1	Complete	
8	Finalise logo	Nithesh K					Complete	
9	Main Page	Clyde L				1, 7	In progress	
10	Set up GitLab repository	Pavlos L					Complete	
11	Learn how to use GitLab	Everyone					Ongoing	
12	Produce a list of ingredients	Pavlos L					Complete	
13	A way to store recipes in the back-end	Max K					In progess	
14	Project Charter	Sze L					Complete	
15	Change Management Plan	Sze L					Complete	
16	Risk Analysis	Pavlos L					Complete	
17	Team Charter	Oscar L					In progress	
18	Work Plan	Clyde L, Mohammed K				6	In progress	
19	Project Definition Document Draft	Sze L				14, 15, 16, 17, 18	In progress	Produced by everyone but put together by Sze L.
20	Start peer reviewing code	Everyone					Ongoing	
Sprint 3			19/03/2019	26/03/2019	7		Not started	
Sprint 4			TBA	TBA	TBA		Not started	
Sprint # (Other features/tasks with some due dates not yet confirmed)			TBA	TBA	TBA		Not started	
	Shopping List						Not started	
	My recipes						Not started	
	Viewing a recipe						Not started	
	Filter						Not started	
	Integration with Testing		-	17/04/2019			Not started	
	Recipick beta version with user acceptance testing		-	22/04/2019			Not started	
	Project demo		-	10/05/2019			Not started	
	Recipick release candidate testing		-	13/05/2019			Not started	
	Final Audit Report		-	13/05/2019			Not started	

Appendix F: Goal Oriented Product Roadmap

Project Start Date	18 February 2019					
Project End Date	13 May 2019					
Latest Revision Date	14 April 2019					
Date	**1st Quarter (18/02 - 11/03)**	**2nd Quarter (11/03 - 01/04)**	**3rd Quarter (01/04 - 22/04)**	**4th Quarter (22/04 - 13/05)**		
Name	Planning	Design	Beta version	Candidate release version		
Goal	- Outline the features of the product to aid in planning - Members to be aware of their roles and to establish a conducive team working environment	- Emphasis on finalising the look of the app and its DB structure	- Application ready for beta test with most of its planned functionality ready - Conduct tests to reduce flaws	- App ready for public release with extra features and minimal bugs - Extended features should time permit		
Features	- Basic login functionality	- Password recovery - All user interfaces - Main page navigation - Database functional	- Add recipes to the DB - Locate nearest supermarkets - Find recipes based on ingredients - Owned ingredient tracking - Ingredient shopping list - Upload pictures - Recipe favouriting	- Generate shopping list from recipe - Moderator recipe reviewing Extension: - View popular recipes - Meal planning with shopping list generation - Recipe rating and review system - Sort recipes by rating - Ingredient measurement scaling		
Tasks	- Project selection - Define member roles - Select methodology - Select communucation methods - Populate backlog	- Learn how to use Git - UI mockups - App logo - List of ingredients - Work plan - Team charter - Project charter - Risk analysis - Change management plan	- Revise PDD - Plan final audit report - Write code tests - Document code	- Write final audit report - Submit final audit report - Submit application		
Deliverables	- Project proposal (22/02) - Project pitch (15/03)	- Project definition document draft (18/03)	- Beta version of the app (22/04)	- Tech demo (10/05) - Final audit report (13/05) - App submission (13/05)		
Metrics	Feedback from project sponsor	Feedback from project sponsor	Feedback from other groups Unit tests	Feedback from project sponsor and external examiner User acceptance reports		