COSC345 Software Engineering

Assignment 1

AllRight Project Report

Josh Mullin
2067221

Benjamin McCarthy
2626998

Usman Shah
1006293

Sebastian Sole
1045948

22 April 2020

Contents

Overview

Trade Me currently promotes itself as "NZs biggest & most popular auction & classifieds site." [1] Few, if any, sites have managed to truly challenge Trade Me. One competitor, AllGoods, launched in 2018, has shown some promise announcing itself as “New Zealand's fastest growing marketplace.” [2] However, Trade Me still is the dominant player in its market. While AllGoods presents unique features, including no fees “for regular users to buy and sell,” [3] their Android application, last updated in August 2019, is incomplete and requires refinements. We propose a partial remake of the AllGoods Android mobile application. We will name our app “AllRight” to avoid copyright issues for the time being.

Project Description

AllGoods’ mobile development appears to have stagnated. This project aims to rewrite the AllGoods Android mobile application from the ground up. This redevelopment will not target feature parity. Instead, the central focus will be stability, modern design, and browsability. The outcome shall be an Android mobile application whose design encourages users to browse through categories. This will expose only relevant listings. All further interaction with listings will, at least initially, be performed on the users’ default web-browser. Once browsing and searching has been implemented, additional features, such as login, notifications, and messaging, may be added. The mobile application will integrate with AllGoods using their web API. This will be performed using HTTP requests in Kotlin using existing open-source libraries.

Resource Requirements

The main resources we will need for this project are:

- Access to computers to program with.
- An Android phone to test the application on (though this is not totally necessary as Android Studio provides a virtual device to test on).
- Time to work on the project. This will be managed both individually and as a team, with tasks being assigned to each individual as well as group work, and a project leader to keep track of deadlines and manage productivity.
- Android Programming Software. The software we use (such as Android Studio, libraries and APIs etc) will all be free to use so will not run us any costs.
- Version Control Software. Git and Github are free for use, so we will use this to manage our project’s progression.
- Access to the Internet. This is needed to communicate with each other whilst we are in a state of lockdown, and for any research needed for our programming and design.

[1] https://www.trademe.co.nz/
[2] https://allgoods.co.nz/
[3] https://www.scoop.co.nz/stories/BU1803/S00924/new-kiwi-site-ready-to-topple-trade-me.htm

User and System Requirements

- The app shall display a list of product categories on the startup screen.
- The user shall be able to select categories to view listings in that category.
- The user shall be able to switch between the “Mall” (Firsthand) and secondhand stores.
- The user shall be able to filter what products they view by selecting item specifics in a hideable popup menu.
- The user shall be able to browse products.
- The products shall be displayed in a list-like fashion.
- The products in the list-like view shall display a picture of the product or a placeholder, its title, and its start and buy price.
- Products seen while browsing shall act as buttons. When tapped the user shall be presented with detailed information on the item.
- The information displayed when viewing a specific product in its detailed view **shall** include:
    - A larger picture of the product than on the list-like view
    - The name of the product
    - The start price of the product
    - The buy now price of the product
    - A description of the product
    - Additional pictures the seller has added to the product
    - Seller information, which shall include name, profile name, location, feedback
    - Whether the item is new or secondhand
    - A clickable link, which opens the desktop version of the auction in the user’s default web browser
    - The user shall be able to zoom in on a picture of the product
- The information displayed when viewing a specific product in its detailed view **should** include:
    - Shipping Information
    - Item Specifics
    - Rating of the store
    - Location of the store
    - How many items the store is selling
- The user shall be able to press the back button on their device to see the previous screen they have seen, or their home screen if they are at the home page.
- The app shall not crash.
- The app should take no longer than 6 seconds to launch when tested on an entry level smartphone. We will test with a Samsung J3 Pro, which we consider to be an entry level smartphone.
- The user should be able to search for products.
- The user may be able to login to their account.
- The user may be able to receive notifications about messages sent through AllGoods.
- Buyers may be able to chat with Sellers.

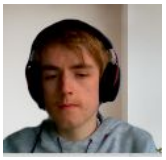
Organization

We are four Computer Science students in our third year at Otago University. We all have a base understanding of programming and design, with some experience developing computer programs and working with APIs. We also all have experience working with each other, as we have all taken the same classes together for the last two years, meaning we have a good idea of what each individual is capable of, and which areas each of us excel in. We hope to gain more experience in application programming and software engineering as a practice, with emphasis on code readability, working to a deadline, and general team-skills.

People and Roles

Individual roles are assigned in a talent oriented manner.

Josh *Back-end developer, tester*



Josh is comfortable in Python and Java, and has some experience in C and C++. He enjoys playing with software for novelty, so will be helpful in stress-testing and bug-finding. He hopes to come out of the project having built an attractive app that people may or may not download. Josh wants to improve his teamwork, communication skills, and problem solving skills by working with Usman to develop the algorithms and data structures needed for the project on the Back-End.

Benjamin *Front-end developer, maintainer*

Benjamin feels the most comfortable in Java, C and C++, as most of the work he has had to do has mainly involved those languages, but is keen to be exposed to as many languages as possible. He enjoys optimizing code and problem solving. He will mostly deal with User Experience and Front-End design as he has not had much to do with the Front-End before, but will also be involved in reviewing code written by the others and assisting in the other team members' tasks.

Usman *Planning, back-end developer*

Usman has experience in Java, Python and C languages, and is up for a challenge of learning a new language. He is also studying Information Science papers which will be helpful during the project. He is a Software Engineering Major, so will bring to the table knowledge in the best practice of Software Engineering concepts, and will also be involved in building the Back-End.

Sebastian *Project lead, maintainer*

Sebastian enjoys finding exploits and bugs in existing systems. He is also comfortable programming in C, C++, Java, and Python, but is always ready to learn a new language. He is an avid user of GitHub and GitLab. With a strong attention to detail, he will ensure the project continues to run smoothly while development is completed remotely. He will be managing and maintaining the project when not helping with programming across the stack.

Project Breakdown

Identifiable Activities

- Group members learning the Kotlin language
- Group members gaining an understanding of Android Programing
- Group members becoming competent with Version Control (Git)
- Group members learning to work as a team
- Group members learning to work remotely

Deliverables and Milestones

Report
Version: 0.0.1
Release Date: 22/04/2020
Milestones: GitHub repository created; Written report finalised

Alpha
Version: 0.1.0 TBC
Release Date: 29/05/2020
Milestones: Retrieve categories listings from the AllGoods API; Browse AllGoods listings in GUI; Filter listings by AllGoods categories; Open listings in web browser; Switch between the 'Mall' and 'Second Hand' marketplace

Beta
Version: 0.2.0 TBC
Release Date: 31/07/2020 TBC
Milestones: Retrieve item specifics from AllGoods API; Filter listings using AllGoods 'Item Specifics'; Display detailed listing information in GUI; Enlarge images for closer inspection

Final Release
Version: 1.0.0 TBC
Release Date: 25/09/2020 TBC
Milestones: Sort items by price or name; Searching listings with keywords; No crashes

Risk Analysis

During this project, there are a few setbacks we may encounter.

AllGoods demand we cease and desist all development:
If this were to happen, we would do our best to work with AllGoods to come to an agreement. Failing that, we would look to pivot our app to either a new idea, a different website, or simply change ideas completely.

Reverse engineering APIs may prove challenging:
We can monitor API requests made when browsing the AllGoods website using a HTTPS proxy. If this does not reveal enough information, intercepting traffic from AllGoods' existing app may fill in the gaps. However, this attack is less trivial. If we cannot retrieve enough information, we may contact AllGoods and request support for official integration, should they approve of the app. The alternative is to pivot to a new idea.

Learning Kotlin could take more time than anticipated:
While Kotlin is an elegant language, certain tasks may prove too much for a challenge. If we begin running over time due to our language inexperience, we will begin using Java for some components. This may speed up development as the team are all familiar with Java.

Team member issues:
There are a couple of situations which could prevent a team member from completing their assigned tasks. If a team member fell ill, the team would work with the ill member and Dr. Andrew Trotman to work out a solution. Any solution will take into account how far along we are in the project and the severity of the condition. The other possibility is that a team member falls behind on their duties or becomes increasingly difficult to work with. We would exhaust all efforts to reconcile or come to an understanding before considering excommunicating the member from the team.

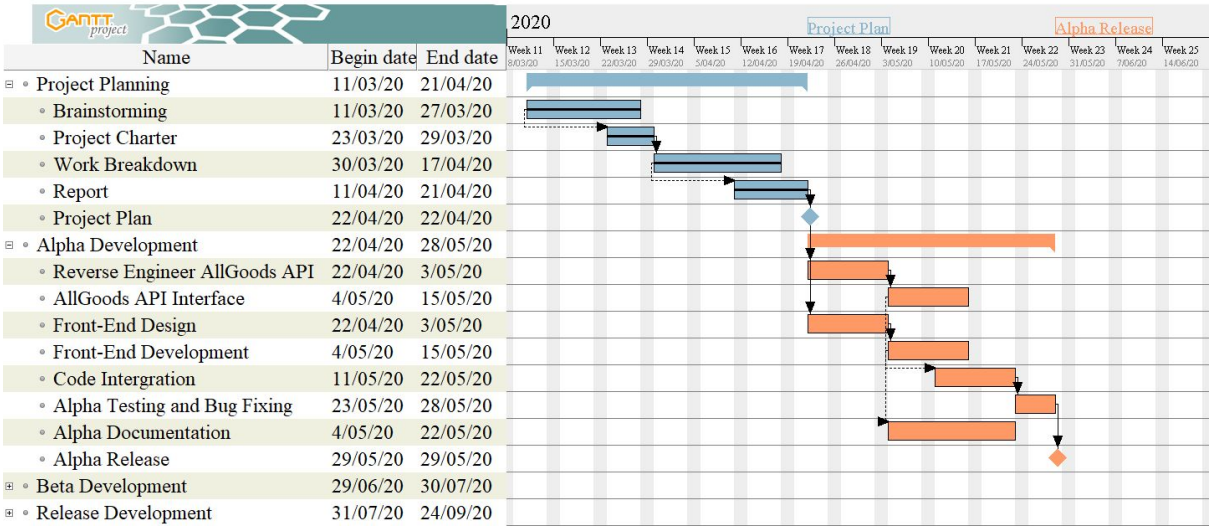
Open-source code repository:
We will have a public repository on Github. Our repository will have restricted write access to ensure no external party can modify our code without forking the project and creating a pull request. The master branch will be protected. All pull requests must also be reviewed before merging.

Remote teamwork problems:
We will hold bi-weekly meetings to ensure that all team members are updated on progress. This will minimize any confusion from not being able to work together physically for the first parts of the project due to COVID-19 restrictions. We will also use a kanban-style project board on GitHub.

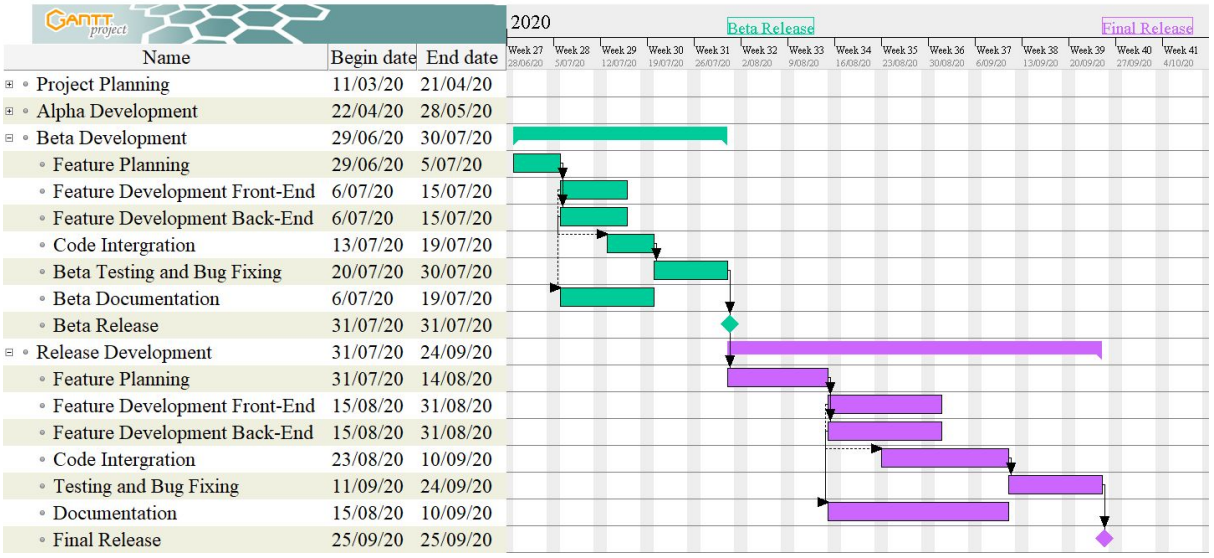
Another team has proposed the same idea as us, or our idea is unsuitable for the project:
In the unlikely event of having the same idea as another team, we would discuss potential modifications with Dr. Andrew Trotman and the other team. If our project is unsuitable, we would discuss our options with Dr. Andrew Trotman.

Project Schedule

We will use a Gantt Chart to keep track of the project schedule. This Gantt Chart will be revisited after each deliverable and is accessible by each team member online.



Name	Begin date	End date
Project Planning	11/03/20	21/04/20
Brainstorming	11/03/20	27/03/20
Project Charter	23/03/20	29/03/20
Work Breakdown	30/03/20	17/04/20
Report	11/04/20	21/04/20
Project Plan	22/04/20	22/04/20
Alpha Development	22/04/20	28/05/20
Reverse Engineer AllGoods API	22/04/20	3/05/20
AllGoods API Interface	4/05/20	15/05/20
Front-End Design	22/04/20	3/05/20
Front-End Development	4/05/20	15/05/20
Code Intergration	11/05/20	22/05/20
Alpha Testing and Bug Fixing	23/05/20	28/05/20
Alpha Documentation	4/05/20	22/05/20
Alpha Release	29/05/20	29/05/20
Beta Development	29/06/20	30/07/20
Release Development	31/07/20	24/09/20



Name	Begin date	End date
Project Planning	11/03/20	21/04/20
Alpha Development	22/04/20	28/05/20
Beta Development	29/06/20	30/07/20
Feature Planning	29/06/20	5/07/20
Feature Development Front-End	6/07/20	15/07/20
Feature Development Back-End	6/07/20	15/07/20
Code Intergration	13/07/20	19/07/20
Beta Testing and Bug Fixing	20/07/20	30/07/20
Beta Documentation	6/07/20	19/07/20
Beta Release	31/07/20	31/07/20
Release Development	31/07/20	24/09/20
Feature Planning	31/07/20	14/08/20
Feature Development Front-End	15/08/20	31/08/20
Feature Development Back-End	15/08/20	31/08/20
Code Intergration	23/08/20	10/09/20
Testing and Bug Fixing	11/09/20	24/09/20
Documentation	15/08/20	10/09/20
Final Release	25/09/20	25/09/20

Allocation of people to tasks

- One or two team members will be allocated per development task. Documentation, and planning will still involve all group members.
- Tasks planned for completion during the second semester will be revised after first semester exams finish.

Monitoring and Reporting

The project will be monitored through GitHub. The GitHub repository has an automated kanban-style project associated with it. This project board will display all tasks and their current status in a central place. This allows us to identify tasks that have potential issues or require review. To keep work organised we will use the following practices:

- Issues will be created for each task, enhancement, and bug. These will be labeled appropriately and assigned to team members.
- A branch will be created for each issue being developed.
- Once an issue is resolved or feedback is required, a Pull Request will be created.
- Each Pull Request requires two reviews and must pass CI checks before being merged into a develop branch.
- Once the develop branch reaches a milestone it will be tagged and merged into the master branch.
- The master branch will be protected and will contain the latest official version of the project.

Existing Applications

Several mobile applications exist for buying and selling goods online. We will restrict our comparison to apps for websites purely targeting the New Zealand market. The features listed for each existing app that follows are not exhaustive.

Trade Me

- Buy and sell goods online
- Strong listing search tools
- Detailed listing information
- Create account and login
- Chat with sellers about listings
- Regularly updated

AllGoods

- Buy and sell goods online
- Filter listings using geolocation
- Detailed listing information
- Create account and login
- Chat directly with sellers
- Last updated 7 August 2019

BigMarket

- Buy and sell goods and services online
- Create account and login
- Chat directly with sellers
- Single Page Application (SPA) running on Angular 5

How our application differs

The AllGoods mobile application currently provides a “quick” way to register, browse listings, and sell goods on their market place. Their Android application in particular, has numerous bugs and performance issues. Rather than replacing their application with an identical version, our app will prioritise browsing their marketplace using the latest Android design recommendations. While our application may not allow users to register or login to the AllGoods marketplace, we would like to make our app easily extendable for future development. Our app aims to improve the user experience when searching for goods on the AllGoods marketplace, and will redirect users to the official website if they would like to interact with the seller or make a purchase. The existing AllGoods app provides purchasing and communication features. Since browsing is our priority, we will present all listings in the mobile app including ones that do not have a “buy now” price, a feature that is currently missing from the official application.

Our application will obviously integrate with a different marketplace to both Trade Me and BigMarket. Our application will also be native to our platform. This is quite different to BigMarket, a SPA. We should thus have an application that looks and behaves exactly as our target users expect.

Anarchy Information

Performance pending, we may irritate AllGoods if we manage to replace their official application. However, our purpose is not to directly irritate AllGoods, but rather to revive their mobile presence and create stronger competition to Trade Me. As a profit maximising firm, Trade Me will be irritated if a small team manages to create an elegant mobile application for one of their competitors.

Disclaimer

We are not in any way affiliated with or endorsed by AllGoods.

“We don’t know what we want, but we know how to get it. We want to destroy the passerby, because we want to be anarchy.”