Final Requirements Specification

SOFTWARE REQUIREMENTS SPECIFICATION

Team Bilby

Task

Semester 2, 2017
Revision 2.0

Abstract

This Software Requirements Specification (SRS) specifies the complete, up-to-date and final requirements for the system produced by this project. It is an updated version of the Baseline Requirements

This document contains background information on task management and resource scheduling. The SRS also outlines the proposed system to be developed. This is detailed in terms of the functional and on functional requirements as well as examples of system behaviour and prototypes for the graphical user interface.

See Traceability Notes for a glossary of Terms.

Contents

1 Introduction

1.1 Overview of Project

The aim of this project is to create a software system which enables individuals to find skilled workers to complete a particular task. To create a Task environment, the system displays a list of Task posts that a user can view and apply to. Similarly, the system allows users to create a Task posting which other users can then apply for.

The system can be divided into 6 sections:

1. Account and profile creation: to use the platform, a user must first create an account with a user profile. This profile contains information about the user such as their name, location and skill set.
2. Creation of a Task posting: using data provided by the user, a Task listing is created and displayed in the Task feed.
3. Viewing of Task listings: users are able to view a list of Tasks currently on offer, and filter these listings according to relevancy, location and so forth.
4. Shortlisting, and Applying for a Task: a user is able to apply for a particular Task listing. Shortlisting a Task places it into an intermediate filtered list.
5. Shortlisting, and Accepting applicants: the Task poster is able to view all Task applications and select an appropriate candidate. Shortlisting an Application places it into an intermediate filtered list.
6. Rating users: following the completion of a Task, the Task owner is able to provider feedback and a rating for the candidate. Similarly, the candidate is able to rate the Task poster according to a set of criteria.

1.2 Scope of Document

The purpose of this Software Requirements Specification is to provide information regarding the Tasks posting system to be created. In accordance with the IEEE Software Engineering Standards, this document outlines the functional requirements of the features of the system.

The SRS also details the non-functional requirements of the system, such as the nature of users, implementation requirements, software constraints and software systems attributes. This is extended to include prototypes of the graphical user interface as well as example behaviour of the system.

1.3 Purpose of Document

The aim of the SRS is to list all of the requirements that the final product must satisfy. Additionally, it is intended to be used by the Team Bilby, the project's clients, the team's supervisor and subjecy coordinator, and any software developers who may work on this project in the future.

The SRS is to help facilitate shared understanding of the software system among the project's various stakeholders. This document will be used by the team when planning the system's architecture and detailed designs, and will also be referred to when undergoing sprint planning sessions.

Furthermore, given the disjoint nature of these project subjects, this document has been designed to serve as a basis for future improvements or development of the software system. It aims to provide a comprehensive overview of the system to assist any potential future developers who may be charged with working on the project.

1.4 Client Details

The project’s client is Paul Ashkar from Cap Gemini. Paul can be contacted via email at paul.ashkar@capgemini.com

1.5 References

- “IEEE Standards Collections, Software Engineering”, 1997, The Institute of Electrical and Electronic Engineers.

2 Background on Resource Scheduling

Resource scheduling refers to the practice of assigning resources to Tasks, tasks or projects that need to be completed. This can be conducted by project and resource managers, or by individuals who simply need assistance in completing a task. The aim of resource scheduling is to ensure an efficient allocation of non-finite resources so that a goal can be achieved within a given time frame.

2.1 Creating a Task

Task creation occurs whenever a project, Task or activity needs to be completed by some individual or team. A task usually includes features such as:

1. Deadline for task completion;
2. Remuneration for task;
3. Skills required to complete task;
4. Poster of task;
5. Location of task; and
6. Overview/description of task.

The creator of a task is referred to as a 'Poster'.

2.2 Searching and Applying for Tasks

Tasks can be searched for based on any of the fields of a Task, for example, its Title, Description, Location, etc.

A 'Helper' can Apply to a Task.

Task applications can range from being informal, such as through a verbal expression, or via an application. Typically this application includes the following information about the interested candidate:

1. Name and basic details;
2. Possible brief overview of experience;
3. Accompanying bid/quote for the Task.

2.3 Assigning Tasks

When a Poster has decided upon their preferred applicant, they can Assign the Task to the Helper. The selected candidate is informed that they are successful in receiving the Task, and have the Poster's email address revealed to them. Only one applicant may be accepted for a task.

3 Existing System

3.1 Overview

This section provides an overviewing of the existing systems currently available to the client. At present, online marketplaces and communities such as Airtasker and Facebook groups serve to aid individuals in finding people willing to complete small tasks. Although both of these platforms cover a range of task categories, a platform that focuses on the IT and developer industries does not exist. With the IT sector growing exponentially, there is high demand for people and organisations to connect with quality software engineers, developers, web and graphics designers and IT professionals. The client therefore wants to capitalise on this gap in the market by introducing a platform that targets these skills.

3.2 Main Features

The main features of these systems include:

1. Creating a “bare bones” account within minimal user information.
2. Searching for Tasks based on location and remuneration only.
3. Applying for a task by making a bid.
4. Creating and publishing a task.

3.3 Limitations

Given a vast majority of IT and developer projects require a number of people working together, it is imperative that task resourcing platforms are capable of facilitating collaboration and teamwork. The aforementioned platforms such as Airtasker and Facebook groups however do not have this capacity, thereby severely reducing their effectiveness at finding people for larger projects. In addition to this, a high volume of Task spam is encountered on these apps as there are no restrictions on who can apply to a task. This means that individuals without the required skills set are not screened during the application process, creating overheads for the Task poster when selecting a suitable candidate. Similarly, these systems have no means by which candidates can filter tasks based on skills sets, meaning they must wade through a large number of inappropriate Tasks. This may discourage users from engaging with current platforms, thereby diminishing the pool of talent from which Task posters are able to select a candidate.

Following the completion of a task, there are either no, or very trivial metrics by which a user’s performance can be measured. In addition to receiving payment for task completion, many users want recognition for and feedback on their skills. This would not only help Task Helpers demonstrate their competency, but also assist Task Posters when considering different applications for a project.

4 Proposed System

The proposed system creates a platform target at resource allocation within the IT and software industry. The main attributes of this will be the ability to create, search and apply for Tasks that relate to the users’ specific skills set. It is intended that the platform will be used by both individuals search for work and those needing a task to be completed. Posters are uses which create and publish tasks. Helpers apply for tasks. Posters choose an applicant to complete the task. All user profiles simulatenously have the ability to act as Posters and Helpers simultaneously (creating, applying for tasks, etc).

The proposed system is a proof of concept, and is not intended to be immediately usable by a larger user base.

The functionality of the system has been divided into the categories core and non-core. Given the relative inexperience of the development team and the project’s tight deadline, features classified as “core” must be implemented in order for the project to be accepted, whilst “non-core” features are considered to be “nice to have”.

4.1 Core Requirements

The below features are the essential requirements of the system. These must be implemented in order for the product to be accepted by the client. These features are considered from the perspective of the end user.

1. Create a user account and login.
2. Create and publish a Task.
3. Display Tasks in a task feed.
4. Apply for a Task.
5. Select a candidate to complete a Task.
6. Complete a Task, Rate Helper
7. Shortlisting and Discarding Tasks
8. Shortlisting and Discarding Task Applicants
9. Gamification elements
    a. Swiping of tasks and applicants
    b. Display average user rating on profile page

4.2 Non-Core Requirements

These requirements reflect features that the client considers to be highly desirable, but not indicative of the product's acceptance. Following implementation of the previously aforementioned core-requirements, the development team will then focus on building out these features.

It is anticipated that at least some of these features will be implemented, however there is the possibility that the remainder will be de-scoped due to time constraints or lack of knowledge. It is plausible that this functionality may be implemented by a different team of developers in the future.

1. Rank Task Feed based on relevance
2. Filter Task Feed based on search criteria
3. Constraints on number of Applications per day
4. Listing of Task Renumeration
5. Logo and product name.
6. Update Profile
7. Delete Task
8. Gamification elements
    a. Display number of times an applicant has been shortlisted on profile page

4.3 Out of Scope Requirements

The following requirements were decided to be Out of Scope during meetings following the establishment of the Baseline Requirements

1. An in-app payment system
2. An in-app points system which allocates points to Helpers who complete tasks
3. An in-app task management system within Tasks (ie sub-tasks)
4. Tasks being completed by multiple users in collaboration
5. An in-app messaging sytem
6. Task Posters can rate candidates on specific skills.
7. Helpers and Posters can comment on a task posting.
8. A leaderboard displays Helpers ranked by number of shortlists and average rating
9. Users can upload a maximum of 4 pieces of work onto their profile for Task posters to view.
10. Helpers can rate the Poster of Tasks they have completed

4.4 Use Cases

Due to a lack of relevant provided Use Cases and changing requirements, the team has written our own after requirements gathering. It was decided that Casual Use Cases provide, along with the Architecture views, sufficient provide system-user interaction descriptions.

The use cases are as follows:

4.4.1 Find new tasks

Main success scenario

After logging in at the login screen, the Helper scrolls through the Task Feed (accessible from the navigation pane under "Search"). They then shortlist a task by swiping it to the right. After shortlisting tasks, the helper navigates to the Helper tab of the My Jobs Screen, located in the navigation panel. They then press on listed task and are taken to the Task Detail screen, where they read the details of the Task. They then choose to apply to the Task, and press 'Apply' after filling in the three Application Questions. After a period of time, the Poster accepts the Helper's task application. At this point, the helper can navigate to the Assigned tab of the My Jobs Screen and press on this task to bring up the Task Detail screen, which now has the Poster's email. The Helper begins work on the task, emailing the Poster with questions.

Alternative Scenarios

- After logging in at the Task feed, a Helper can discard a task they are not interested in by swiping left. It is then removed from the task feed permenantly for that Helper.
- After logging in at the Task feed, a Helper can press on a task to be taken straight to the Task Detail screen. Here they may Apply directly, skipping the Shortlisting step.
- After applying for a Task, the Helper's application might not be accepted by the Poster. In this case, the Task remains in the Applied Tasks tab of the My Jobs screen.

4.4.2 Create Profile

Main success scenario

At the Login screen, the User (Helper or Poster) selects 'Create a new Account'. They are taken to the Create Account screen, where they can add information about themselves, including a name, description, and an uploaded photo. Upon pressing Save, the account is saved and may be logged into.

Alternative Scenarios

- After pressing Edit profile, the User may also Add Skills from a pop-up menu of predefined skills.
- At any time in the Account Creation Screen, the user may discard the account by pressing Cancel.

4.4.3 Edit Profile

Main success scenario

After logging in at the login screen, the User (Helper or Poster) selects the My Profile Screen from the navigation panel. After pressing Edit Profile, they may modify any of the editable fields (does not included derived statistics such as Rating, Number of tasks completed, etc). Upon pressing Save, the changes are saved.

Alternative Scenarios

- After pressing Edit profile, the User may also Add Skills from a pop-up menu of predefined skills.
- After making changes, the user may discard changes by pressing Cancel.

4.4.4 View Profile

Main success scenario

After logging in at the login screen, the User (Helper or Poster) selects the My Profile Screen from the navigation panel. They are shown their provided information, as well as statistics such as average rating, number of tasks completed, etc.

4.4.5 Find Applicant for New Task

Main success scenario

After logging in at the login screen, the Poster presses the Plus button at the Task Feed and are taken to the Create Task screen. The Poster enters all relevant Task information, such as location and remuneration, and presses Create. After waiting for Helpers to Apply, the Poster can browse Applications at the Poster Open Tab of the My Jobs screen, accessible from the Navigation pane. Of the listed Open tasks created by the Poster, they click the Task in question and are taken to All Tab of the Applicants Screen. The Poster swipes right on an application card to Shortlist the application, and it is removed from the list. They then navigate to the Shortlist Tab where shortlisted applications are displayed. They select the application and are taken to the Application Detail Screen, containing details about the applicant along with answers to their Application Questions and their email address. The Poster presses 'Accept' to accept the applicant and Start the task, which is moved from Open to Assigned.

Alternative Scenarios

- After logging in at the login screen, the Create Task screen is alternatively accessible from the Poster Open Tab of the My Jobs screen.
- At any time before the Task is marked as In Progress, the Poster may navigate to the Poster Open Tab of the My Jobs screen and select "Delete Task" for a Task. The Task is then deleted and is no longer visible to any Users.
- At the Open Tab of the My Jobs screen, the Poster may press on the application to be taken directly to the Application Detail page. After reviewing the Applicant's details, they may press Accept to accept the application and start the job. This skips the Shortlisting stage.
- At the Applicant Detail screen, the Poster can press Reject to Decline a Helper's application, removing it from the displayed list of applications.
- At the Open Tab of the My Jobs Screen, the Poster may swipe left on an application card in order to Decline the Helper's application, removing it from the displayed list of applications.
- The Poster can contact the Applicant via email before accepting their application, in order to clarify or gain further information.

4.4.6 Manage In-Progress Task

Main success scenario

After logging in at the login screen, given that the Poster has an assigned task, the Poster navigates to the Poster   In Progress Tab of the My Jobs Screen. The Helper's email is displayed here, and the Poster keeps updated on the progress of the task by emailing the Helper. Once they deem the task to be completed, they navigate to the Poster In Progress Tab of the My Jobs Screen, and press Close Job to be taken to the Rate Helper screen. After enterring a rating for the Helper's performance, they press Submit in order to rate that Helper and mark the Task as Complete.

Alternative Scenarios

- None

4.4.7 Addiditonal Scenarios

- All use cases contain a 'Go Back' scenario, available at any point in the use case. This can be accessed by pressing the Go Back arrow, and takes the user to their previous page, discarding any enterred form data.
- All use cases involving user-submitted forms contain a large number of potential Failure scenarios, where any form is not correctly filled out. If any data inputted fails integrity constraints, the user is warned and allowed to resubmit the correct form.

5 Functional Requirements

5.1 Core Requirements

These features are the essential requirements of the system which must be implemented prior to the product being accepted by the client. The features are considered from the perspective of the end user.

5.1.1 Create a User Account and Login

In order to create a valid user account, the system must implement the following functions:

1. Read user data - a new user will provide information such as their name, location and skills which the system must read in from a text form.
2. Create a new user - using the data provided, a new user account must be created and stored on the User database.
    a. Includes a user photo
    b. Includes a list of skills chosen from a predefined list
3. Read in login details - the system must be able to read and verify that the entered username and password match what is stored on the system prior to granting the user access to the platform.

5.1.2 Create and Publish a Task

This feature is critical for the project as it is considered to be part of the Minimum Viable Product (MVP). The creation of a task requires the implementation of the following features:

1. User input - a user can provide information via a text form such as Task description, date to be completed and location of task.
    a. Also includes 3 questions to be answered by applicants (For example, "what is your experience working with the Agile methodology?")
    b. Includes a list of required skills, chosen from a predefined list
2. Creation of task - the task is creating according to this information and saved on the Task database.
3. Publication - the newly created task is displayed in the task Task feed.

5.1.3 Display Tasks in a Task Feed

To enable a user to view a Task, the system implements the following feature:

1. View - the task is displayed as a posting within the Task task feed.
    a. The task feed is ranked by relevance to the user, which is calculated based on the similarity of a user's skills and location to the task

5.1.4 Apply for a Task

This requirement makes up part of the MVP, thus is critical to the project’s acceptance. The system therefore must implement the following features:

1. Apply - on a Task listing, a user must be able to apply for a Task by selecting on the post.
2. Basic details - information about the interested user from their profile (such as name and location) must be automatically included in the application.
3. Provide additional information - the user answers three questions (selected by the Poster) as part of their application
4. Save - the application must be saved and the Task stored within the user’s “applied tasks” tab for easy access.
5. Send to Task poster - the application must then be sent to the Task poster for viewing. The poster also receives the Applicant's email addres.

5.1.5 Select a Task Applicant

To select a candidate for a Task, the system must implement the following features:

1. View - the Task Poster must be able to view each candidate's’ application.
2. Select - the Task Poster is able to select a candidate for a task from the list.
3. Notify - upon selection of a candidate, a notification is sent to that user that they have been successful. The Applicant also receives the email of the Poster.

5.1.6 Complete a Task and Rate a Task Applicant's Performance

To facilitate this requirement, the system must implement the following features:

1. Update - Set the task as complete
2. View - stars must be displayed that the user can interact with.
3. Select - the Task poster is able to assign an appropriate rating (0-5 stars).
4. Notify - the user who completed the task is notified of the rating and can view it.
5. Update - the aggregate rating of the user who completed the task must be updated to reflect the recent feedback.

5.1.7 Shortlisting and Discarding Tasks

To facilitate this requirement, the system must implement the following features:

1. Shortlist tasks for later application
   a. Select - a user can select a task that they wish to shortlist
   b. Save - the selected task is added to the user’s Shortlisted Tasks List for future reference.
   c. View - when a user accesses their Shortlisted Tasks List, the saved Task is listed.
2. Discard Task
   a. Select - a user can select a task that they are not interested in to discard it
   b. Save - the selected task is added to the user’s discarded list
   c. View - when a user views the task feed, the discarded task is no longer displayed

5.1.8 Shortlisting and Discarding Task Applicants

To facilitate this requirement, the system must implement the following features:

1. Reject a candidate outright
   a. Reject - the candidate is rejected by the Task Poster.
   b. View - the list of candidates is refreshed to remove the rejected candidate.
2. Shortlist candidates
   a. Select - a candidate is selected as shortlisted.
   b. View - the candidate’s application is moved to “Maybe” bucket for future reference.
3. Select a candidate as successful
   a. Select - a candidate is selected to complete the task.
   b. View - the candidate is saved into a “Selected” bucket for future reference.

5.1.9 Gamification Elements

The following additional gamification elements are included:

1. Swiping to shortlist tasks and applicants
   a. Tabs may be swiped with a finger left (Discard) or right (Shortlist)
   b. Discarding a task removes it from view permanently for that user
2. Display average user rating on profile page

5.2 Non-Core Requirements

These requirements reflect features that the client considers to be highly desirable, but not indicative of the product’s acceptance.

5.2.1 Rank Task Feed based on Relevance

To facilitate this non-core requirement, the system must implement the following features:

1. Rank tasks by the following scoring system against the user's profile:
   a. Matching location +3 points
   b. Matching skill +1 point
   c. Missing skill (the task requires a skill not posessed by the user) -1 point

5.2.2 Filter Task Feed based on Search Criteria

To facilitate this non-core requirement, the system must implement the following features:

1. Displayed tasks must match the search terms enterred in the search bar. The following attributes of Tasks are compared to the query:
    a. Title
    b. Location
    c. Description
    d. Poster name

5.2.3 Constraints on Number of Task Applications Per Day

To facilitate this non-core requirement, the system must implement the following features:

1. User has a limit on the number of Tasks they can apply to, dependent on their rating
    a. Count - the user has a ceiling on the number of Tasks they can apply to.
    b. Update - as a user applies to a Task, their daily limit is reduced to reflect this application.
        i. A Helper with an average above four stars may apply to 20 Tasks per day.
        ii. A Helper with an average above three stars may apply to 15 Tasks per day.
        iii. A Helper with an average above two stars may apply to 10 Tasks per day.
        iv. A Helper with an average above one star may apply to 5 Tasks per day.
        v. A Helper with an average above zero stars may apply to 2 Tasks per day.
        vi. A helper with no average rating may apply to 10 Tasks per day
    c. Block - if a user has reached their application ceiling, the system prevents them from being able to apply to any more Tasks.

5.2.4 Listing of Task Remuneration

To facilitate this non-core requirement, the system must implement the following features:

1. Specify remuneration
    a. Specify value - when creating a Task task, the Poster is able to specify a monetary value.
    b. Create - when Task task is published, remuneration is included in Task listing.
    c. View - users looking at Task feed are able to see the task’s remuneration value.

5.2.5 Logo and Product name

To facilitate this non-core requirement, the system must implement the following features:

1. Product contains an acceptable and appealing Product name
    a. The name cannot end with the letter 'R' or contain 'Illy' (specifically according to the client)
2. Product contains an acceptable and appealing logo at the login screen

5.1.6 Update Profile

To facilitate this non-core requirement, the system must implement the following features:

1. At any time, a user should be able to update any fields on their profile, excluding derived fields (for example, date joined, rating, number of shortlists, etc).

5.1.7 Delete Task

To facilitate this non-core requirement, the system must implement the following features:

1. At any time prior to a Helper being assigned to the Task, a Task Poster may delete the task permanently.

5.1.8 Gamification Elements

The following additional gamification elements are included:

1. Display number of times an applicant has been shortlisted on profile page.
2. Display number of tasks an applicant has completed on profile page.

6 Non-Functional Requirements

6.1 Nature of Users

This section outlines the nature of the users of the software, including their level of technical expertise and purpose for engaging with the system.

6.1.1 Users of the System

Task Posters

a. To find and allocate individuals or teams to a task.
b. To widen pool of talent that otherwise would not be available to them.
c. To achieve some goal that they are unable to do themselves.

Task Helpers

a. To earn extra cash on the side of a Task.
b. To improve their skills in a particular area.
c. To gain experience necessary when applying for future Tasks.
d. To provide an income through freelancing.
e. To work on projects otherwise not available to them.
f. To work with people outside of their professional network.

6.1.2 Hardware Competence

Task Posters will have some basic knowledge about mobile devices, such as using apps and typing in input. Given this platform is targeted at the IT and software engineering industry, Task Helpers will have extensive knowledge in one or more fields such as graphic design, front-end development, coding and so forth.

6.1.3 Software Competence

As this system runs on smartphones, all users will be familiar with either Android or iOS devices.

6.2 User Documentation

As the client has not made any requests for user documentation, this shall be kept relatively lightweight and basic. A Setup Guide shall be provided to assist the client and additional users in downloading and accessing the system via their smartphone.

6.3 Hardware Constraints

This system has been designed to run on Andriod and iOS smartphones. Although there are no hardware constraints per se, the mobile device must be capable of running one of these two operating systems.

6.4 Performance Requirements

The performance of the system should meet the following requirements:

The software should be launched in 5 seconds.

The response time of the system should be less than or equal to 5 seconds.

The system’s Task feed should be updated to include a recent Task posting within 2 minutes of creation.

The processing time of Task applications, candidate selections, commenting and ratings should be less than 1.5 seconds.

These performance requirements are in accordance with those defined by Teams Emu, Wallaby and Koala in SWEN90009, Semester 1, 2017

6.5 Design Constraints

Given the ambiguity surrounding gamification within the system, it is necessary that the design is flexible so as to accommodate any future functionality. As aforementioned, there is a small chance that future developers will work on this project, thus the design needs to be extendable so that improvements can be made with minimal impact. The design should therefore be as modular as possible, and well documented so as to assist in understanding.

The client is impartial to the programming language chosen, and has no preference for whether the system is in the form of a native app or mobile-first web application. The client has also outlined that it is sufficient for the system to run on either iOS or Andriod. The system does not have to run in a web browser.

6.6 Software System Attributes

6.6.1 Reliability

Reliability of the system is important. The system should automatically recover within 10-15 minutes from any faults, errors or crashes. Additionally, the system should be able to recover from events such as power supply being cut from the mobile device, switching between applications or an abrupt termination of the program, as encountered by other apps.

6.6.2 Security

There are a number of security requirements that must be met by the system. Only the Task poster has permission to make changes to, delete and assign candidates to a Task task. Similarly, personal details of a Task Helper (such as phone number or email) can only be accessed by the Task Poster once a task has been assigned to them. Similar information about a Task Poster is only accessible to a Task Helper following assignment of a Task. Users are also only able to create or apply for Tasks with a registered account, and account passwords and information is stored securely.

6.6.3 Accuracy

The information recorded and displayed by the system shall be accurate. Once a task has been assigned to a candidate, it is no longer open to further applications. When a user is browsing the Tasks task feed, only those relevant to them shall be displayed, and the system will block a user from applying to any Task whereby they do not meet the criteria, or they have reached their maximum number of applications. A user’s ratings will accurately reflect the aggregate of what they have received from all Tasks completed. Additionally, only candidates that have applied to a specific Task will be viewable by the Task Poster.

6.6.5 Interface

This system will be user friendly and operable with only a basic understanding of smartphone devices.

6.7 Gamification

The client has placed a strong emphasis on the need for a sound user experience, with a focus on including multiple gamification elements. The client envisions this system to be “a combination of LinkedIn meets Airtasker, meets Uber meets Tinder”, and thus wants an interactive and playful experience. The client has indicated a desire for swiping features (similar to Tinder) to be included in the Task Feed. The aforementioned points system and leaderboard is another example of gamification.

7 Graphical User Interface

This section outlines the proposed graphical user interface (GUI) of the system based on the functional requirements outlined in Section 5. The GUI is to be used as a guide for design purposes only, as the capabilities of the chosen framework may limit what can be achieved. For this reason, this section is kept quite brief.

A note on images:

The included images are from early prototypes and concept mockups in order to more easily outline the required screens. **They are accurate only as to the workflow and basic layout of the system**. **They are not accurate with respect to all fields, details, and appearance**. In particular, visual look/polish, colours, images, and text fields may all differ in the final product. The Leaderboard Tab, which is visible in some images, is also no longer a requirement. The team has already spent a significant amount of time gathering requirements (due to irrelevance of provided requirements), and it was deemed that providing 100% accurate prototypes and mockups would detract from the available time in compeleting the project. Although notes are provided under each image outlining any outdated elements, in any instances of ambiguity t**he text in this requirements document should be instead taken as final and override any ambiguity in the images.**

The interface is separated into the following sections. Each section is required in the Core Requirements (non-core requirements add no extra screens).

1. User Account and Login
2. Create and Publish a Task
3. Task Feed
4. Apply for a Task
5. Select a Candidate to Complete a Task
6. Complete a Task and rate Helper
7. Shortlist/Discard Tasks
8. Shortlist/Discard Task Applicants

For brevity, every page will contain a navigation bar at the top of the screen. If the page is not a root page, this navigation bar will also include a back arrow.

7.1 User Account and Login

Text Forms - these forms are provided to allow the user to provide input.

Login

Username: user provides their username.

Password: user provides their password.

Create Account

a. Name: user provides their name.
b. Email: user provides their email.
c. Location: user provides their location.
d. Password: user has to specify a password, and then re-type it.
5. Skills - select a number of predefined skills required for the task.
6. Photo - upload a photo for the user
7. Description - upload a description of the user

Buttons - this then submits the provided information to the system, to be checked against, or added to, the database.

Login

Login: submits username and password to be authorised.

Create Account

Creates a new account.

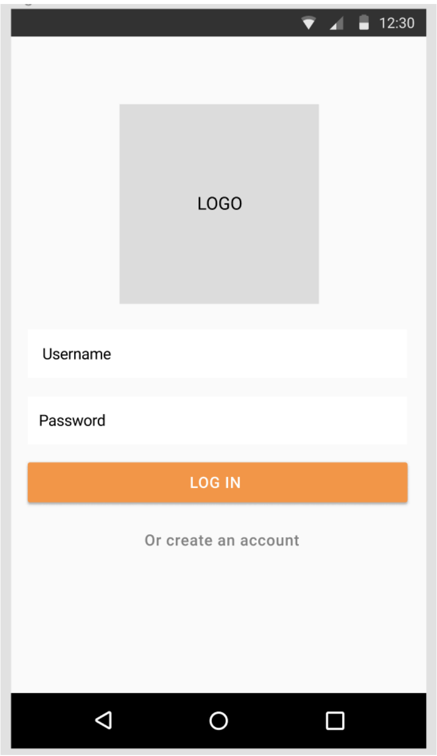
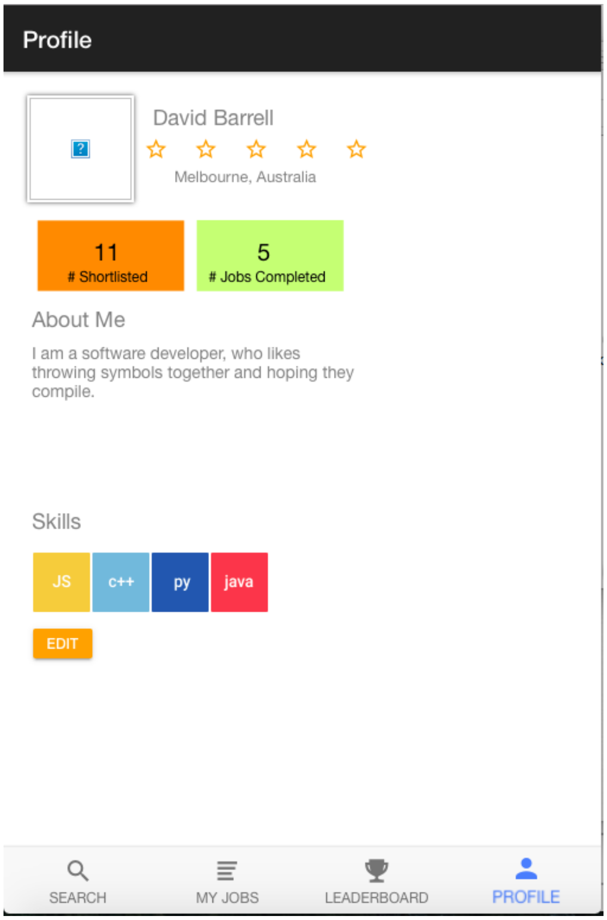


Figure 1: Login Page



I am a software developer, who likes throwing symbols together and hoping they compile.

Figure 2: Profile Page

7.2 Create and Publish Task

Text Forms - these forms are provided to allow the user to provide input.

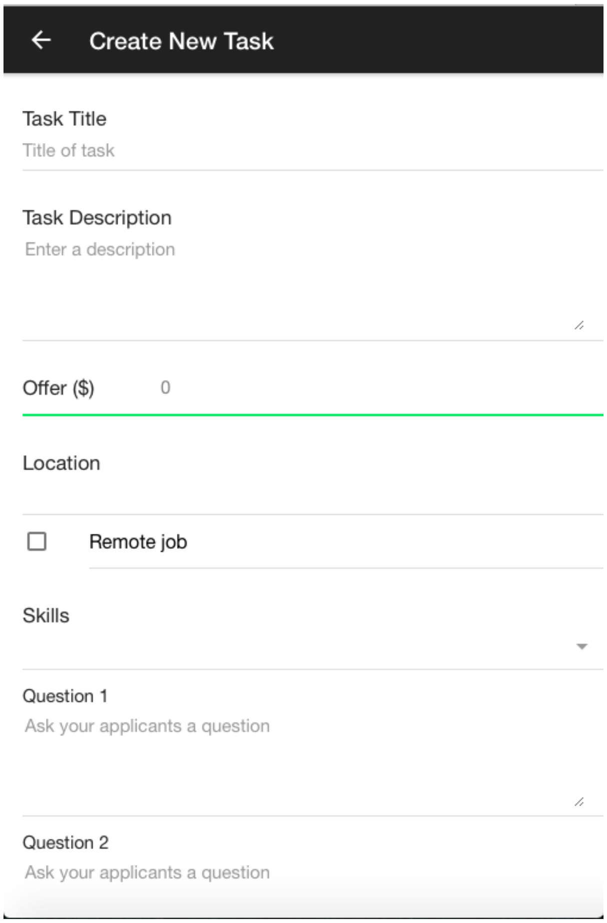
a. Task name: assign a name to the task.
b. Task description: provide a brief overview of the task.
c. Remuneration: provide a monetary value.

Date selector - select the date upon which the project is due.

Check boxes - select which skills are relevant to a Task from a predefined list.

Button - submit information so it can be saved and published.

Skills - select a number of predefined skills required for the task.



← Create New Task

Task Title
Title of task

Task Description
Enter a description

Offer ($) 0

Location

□ Remote job

Skills

Question 1
Ask your applicants a question

Question 2
Ask your applicants a question

Figure 3: Task Creation Page

7.3 Display Tasks in a Task Feed

Posting - displays all information relevant to the Task.

a. Task Title
b. Skills

c. Due Date
d. Image
e. Remuneration

Grid - arranges each post into a grid-like structure.

Search bar - allows the user to search based on various fields of the Tasks

a. Skills: user can select skills to include in search.
b. Remuneration: user can specify a range of values.

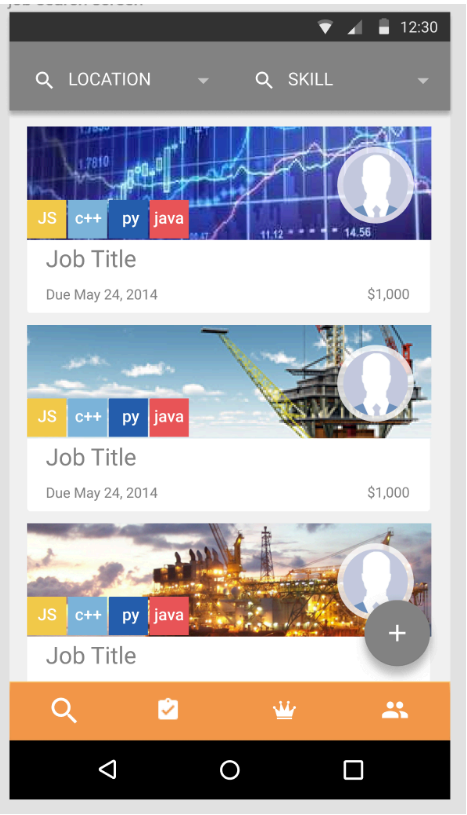


Figure 4: Task Feed Screen. The Location and Skill filters are to be replaced with a single search bar. Tasks no longer should have photos, and Task Cards are to be redesigned.

7.4 Apply for a Task

Task title - displays the name of the Task being applied to.

Posted by - displays the name of the Task poster.

Questions - displays a question that must be answered by the user.

Textform - a text field whereby the user can type their answer to the question.

Button - a submit button that sends the user's application to the Task poster.

7.5 Select Candidates

Candidate view - displays a subset of information about a candidate.

a. Name
b. Skills
c. Location
d. Profile image

Select button - selects a candidate and brings up their full profile.

Reject button - button removes candidate from view.

7.6 Complete a Rask, Rate Helper

Candidate profile - provides a subset of information about the candidate.

a. Candidate name
b. Candidate’s profile picture

Task profile - provides a subset of information about the task completed.

a. Task name
b. Task completion data
c. Task remuneration value

Rating scale - provides a mechanism to rate the user who completed the task. Below are some rating ideas:

a. Stars: select the appropriate number of stars, out of 5.
b. Slider: move the slider bar, out of 10.
c. Smiley faces: select one of five “smiley faces” reflecting quality of service.

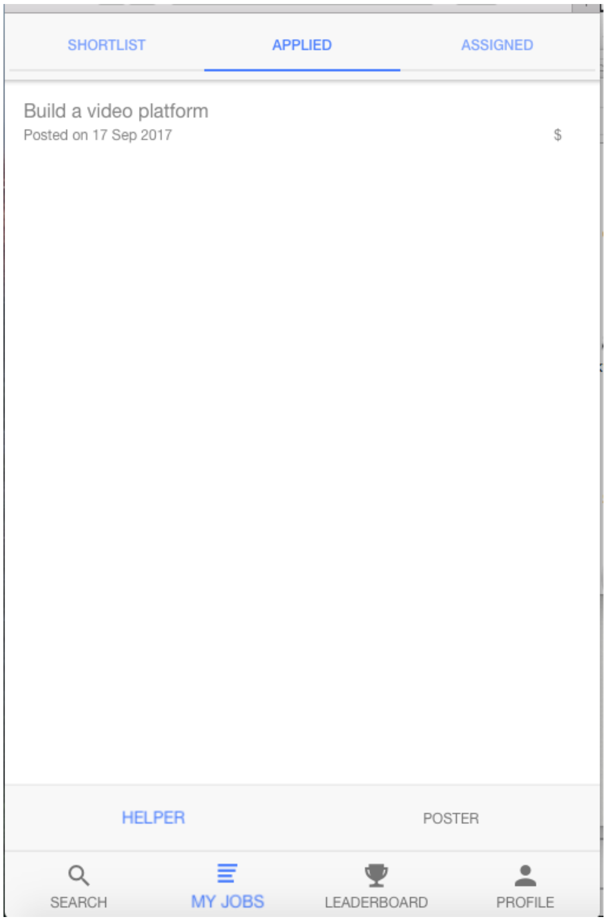
Submit button - submits the rating to the user who completed the task.

7.7 Shortlist/Discard Tasks

Button - in the Task task feed, provide a button (e.g. star) which saves the task.

Navigation bar at bottom of page - navigates to saved tasks.

View - lists all tasks that has been saved.



Build a video platform
Posted on 17 Sep 2017 $

Figure 5: View Shortlisted/Applied/Assigned Tasks as a Helper

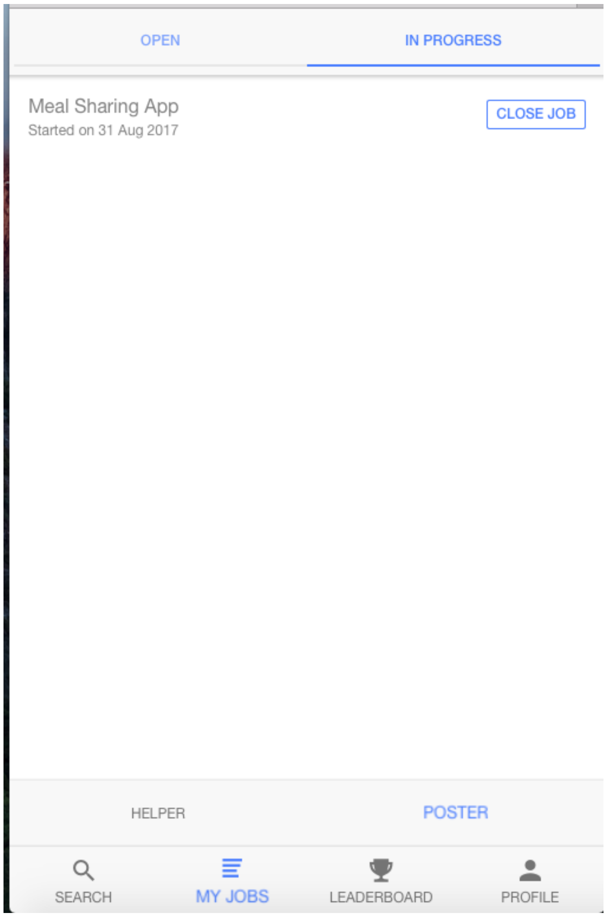
7.8 Shortlist/Discard Task Applicants

Buttons - Task poster can assign a candidate into one of three categories.

a. Reject button: removes a candidate from the list.
b. Maybe button: marks a candidate as a possibility.
c. Yes button: marks a candidate as successful and notifies them.

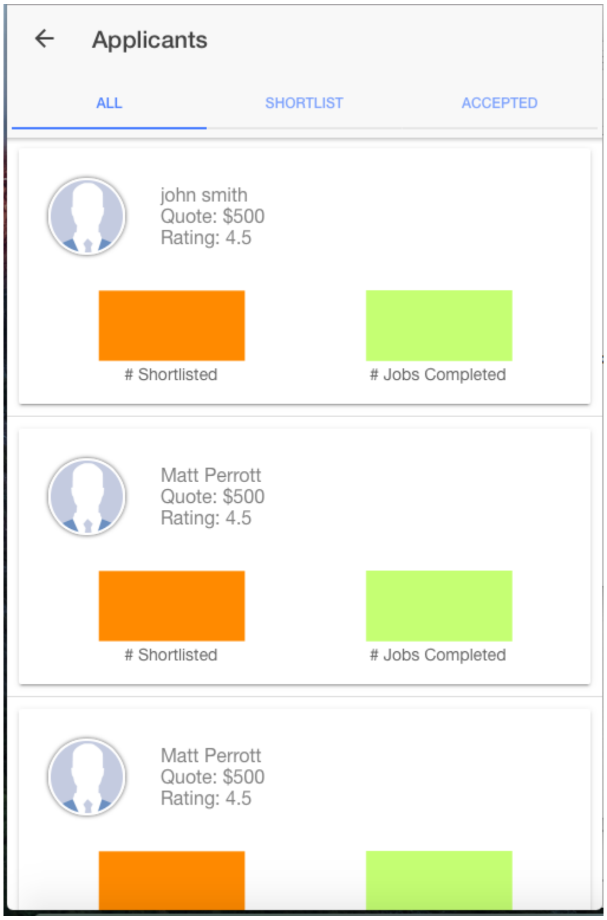
Navigation bar at bottom of screen - allows the Task poster to view the candidates by category.

a. Maybe: displays list of all possible candidates.
b. Selected: displays the list of all selected candidates.



Meal Sharing App
Started on 31 Aug 2017

Figure 6: view Posted Tasks. A "Completed" tab should also be included at the top.

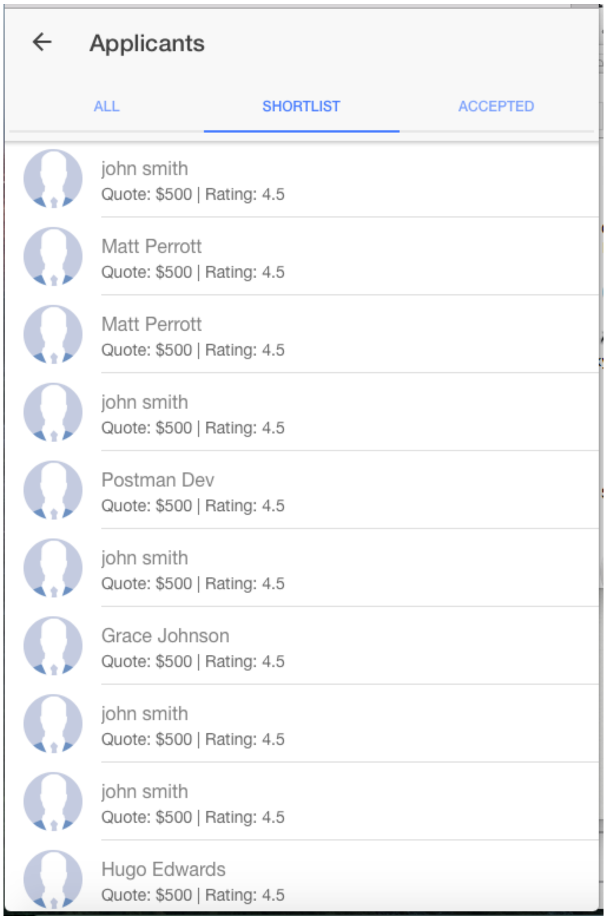


john smith
Quote: $500
Rating: 4.5

Matt Perrott
Quote: $500
Rating: 4.5

Matt Perrott
Quote: $500
Rating: 4.5

Figure 7: View unshortlisted Applicants for a job



Applicants

john smith
Quote: $500 | Rating: 4.5

Matt Perrott
Quote: $500 | Rating: 4.5

Matt Perrott
Quote: $500 | Rating: 4.5

john smith
Quote: $500 | Rating: 4.5

Postman Dev
Quote: $500 | Rating: 4.5

john smith
Quote: $500 | Rating: 4.5

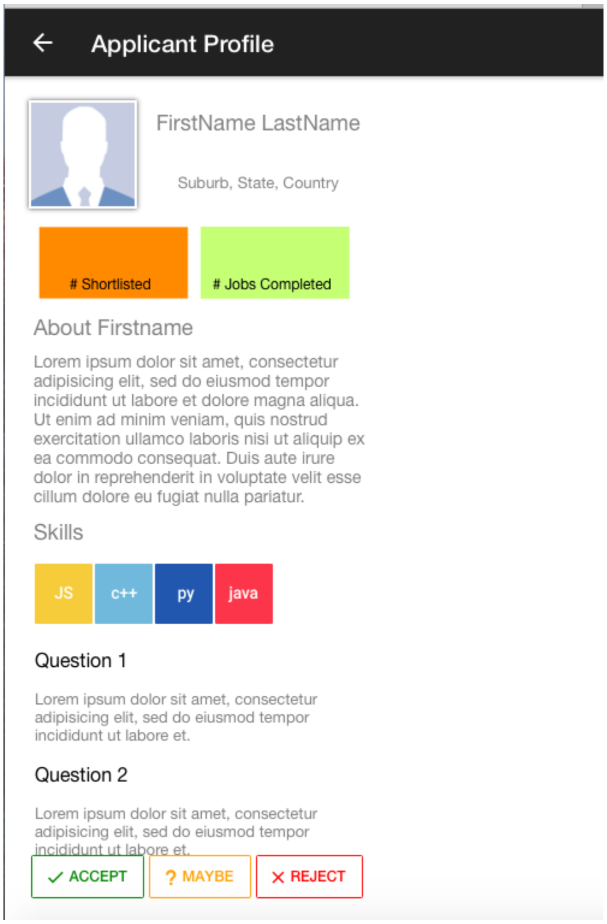
Grace Johnson
Quote: $500 | Rating: 4.5

john smith
Quote: $500 | Rating: 4.5

john smith
Quote: $500 | Rating: 4.5

Hugo Edwards
Quote: $500 | Rating: 4.5

Figure 8: View Shortlised Applicants for a job



FirstName LastName

Suburb, State, Country

About Firstname

Lorem ipsum dolor sit amet, consectetur adipisicing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur.

Skills

Question 1

Lorem ipsum dolor sit amet, consectetur adipisicing elit, sed do eiusmod tempor incididunt ut labore et.

Question 2

Lorem ipsum dolor sit amet, consectetur adipisicing elit, sed do eiusmod tempor incididunt ut labore et.

Figure 9: View Application details

8 Product Acceptance Criteria

This product will be accepted once the requirements identified in Section 5.1 Functional Requirements (Core) are completed, and the requirements defined within Section 6 Non-Functional Requirements are met.

9 Delivery Plan

The final product will be delivered at the conclusion of the project. A subset of the final product will however be delivered at the conclusion of each sprint. The delivery will include the following:

Software System

a. All core functional requirements as outlined in Section 5.1; and
b. Any non-core functional requirements detailed in Section 6 that were implemented.

User Documentation

a. The Setup Guide detailed in Section 6.2

The project is scheduled to be delivered on October 15th 2017.