Software Requirements Specification

for

Project MoveInLo

Version 1.0 approved

Prepared by CodeCrafters:

Eugene Wee Jun Lin (Leader) [U2120698A]

Iain Roderick Tay Rong Yu [U221382K]

Joseph Teo Chee Hoe [U2223041J]

Jiang Yifei [U2201092F]

Kim Hyun Bin [U2020316D]

SCSE Lab SCSX Group 3

14 Sep 2023

Table of Contents

1. Introduction

1.1. Purpose

This documents the software requirements and specifications for MoveInLo. MoveInLo is a mobile application that seeks to smoothen the process of moving in and out of university halls/ hostels for new and existing students in Singapore. MoveInLo achieves this by providing a seamless experience for moving in/ out as well as a new avenue for students to seek temporary jobs.

This app can be installed from the Android or IOS App store. This document will explain the purpose and features of the system, the interfaces of the system, what the system will do, the constraints under which it must operate and how the system will react to external stimuli. This document is intended for both the stakeholders and the developers of the system.

1.2. Document Conventions

For documentation purposes, we will abide by the following conventions:

a. **Font**: Times New Roman
b. **Font Size**s:
    i. **Heading 1 (18px Bold)**
    ii. **Heading 2 (14px, Bold)**
    iii. **Heading 3 (12px, Bold)**
    iv. Body (12 px, Normal)
c. **Line Height**: 1.5 spacing
    i. **Spacing:** Before and after each heading

1.3. Intended Audience and Reading Suggestions

The document is intended for all the stakeholder customers and developers — designers, coders, testers and maintainers. This document need not be read sequentially; users are encouraged to jump to any section they find relevant. Below is a brief overview for each part of the document.

Part 1 (Introduction)

This section offers a summary of the MoveInLo project, including goals, objectives, project scope, general system details and key stakeholders.

Part 2 (Product Design)

This section describes MoveInLo system class by class, including interface details, class hierarchies, performance/ design constraints and process details.

Part 3 (User Interface Design)

This section covers all of the details related to the structure of the graphical user interface (GUI), including some preliminary mockups of our MoveInLo mobile application. Readers can view this section for a tentative glimpse of what the final product will look like.

Part 4 (System Features & Functional Requirements)

This section offers a comprehensive overview on the functional design and system features of MoveInLo mobile application.

Part 5 (Non-functional Requirements)

This section covers the non-functional aspects of MoveInLo mobile application, pertaining to performance, safety, security and business requirements.

Part 6 (Appendices)

This section includes any additional information which may be helpful to the readers.

1.4. Product Scope

MoveInLo is a mobile application which serves to ease the process of moving in/ out of hall residents for students in Singapore. It aims to achieve this through this key features:

a. Seamless moving services/ temporary storage facility of hall-related items.
b. Temporary job opportunities as logistics deliverers for students.

1.5. References

Bruegge, & Dutoit, A. H. (2010). Object-oriented software engineering : using UML, patterns, and Java (3rd ed.). Prentice Hall.

Sommerville, I. (2016). Software Engineering. 10th Edition, Pearson Education Limited, Boston.

2. Overall Description

2.1. Product Perspective

MoveInLo was inspired by one of the biggest woes of university students during the hall life. The hassle of moving in/ out of halls during the start/ end of each academic term.

The inspiration for the name “MoveInLo” was drawn from the commonly-used term in Singapore national service, “ORDLo” which signifies the end of a significant milestone, yet start of a new journey.

2.2. Product Functions

2.2.1. Moving In/ Out Services

Users can schedule for moving services during any of the given periods indicated below:

a. **Moving In**: At the start of the academic term, items will be collected and delivered to the indicated locations within the same day.
b. **Moving Out**: At the end of the academic term. Users can opt for these options:
    i. **Temporary Storage**: Items will be stored in our storage facility, and delivered back to their hall room, at the start of the academic term.
    ii. **Home Delivery**: Items will be delivered back to their preferred address.

Storage boxes will be provided to users to store their items for collection & delivery.

2.2.2. Part-Time Job Opportunities

This mobile application will serve as a platform for university students to find temporary job opportunities as logistics movers (i.e. delivery, driver) to earn additional cash, to aid them ease the financial burdens of university studies.

2.3. User Classes and Characteristics

This section summarises the key user classes and their respective characteristics that will use our mobile application. User classes indicated with red asterisk (*) represents our important manpower personnels, which need to be satisfied in order for the operation to commence.

2.3.1. Hall Residents (Customer)

This includes university students who have been allocated housing within their campus, and are currently staying in their assigned housing.

2.3.2. Student Job Seekers *

It includes university students who want to earn additional cash, by assisting with the following roles:

a. **Drivers**. Requires a valid Class 3/ 3A driving licence.
b. **Logistics Carriers**. Able to handle and carry heavy loads (at least 15kg).

Individuals in this section will mainly be operating with 2.2.1 Moving In/Out Services.

2.4. Operating Environment

Our mobile application will operate on the IOS and Android platforms, and will require location services.

a. For Android platforms, the OS version must be at least Android 6 (Marshmallow).
b. For IOS platforms, the OS version must be at least IOS 9.
c. For location services, the device must have built-in support of Global Positioning System.

2.5. Design and Implementation Constraints

This section specifies the standards/ constraints which we abide to when designing and implementing our mobile application:

a. **Frontend Interface (User Interface, UI)**: React Native version 0.72
b. **Backend**: Golang/ ExpressJS
c. **API**: GoogleMapsAPI
d. **Android**: at least Android 6 (Marshmallow)
e. **IOS**: at least IOS 9

2.6. User Documentation

Our mobile application comes with the following user documentation guide to help new users get started:

a. A YouTube video showcasing the core functionalities of the mobile application.
b. A Quick Start guide documenting the steps to get started using the application.

2.7. Assumptions and Dependencies

This section goes in-depth on the assumed factors and dependencies from third-party components which the mobile application has used.

a. GoogleMaps API requests is limited to
    i. Geolocation: 6000 queries per minute
    ii. Static Maps: 30,000 queries per min
b. The user's device has in-built support for the Global Positioning System.

3. External Interface Requirements

3.1. User Interfaces (UI)

3.1.1. UI Design

The design of the user interface must have a noticeable difference in terms of colour after a user selection.

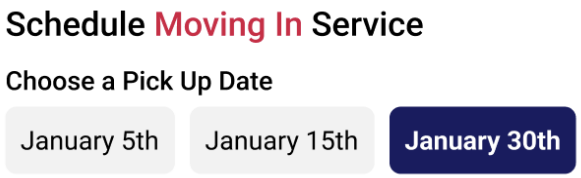


Figure 1: User Selection UI

3.1.2. Top Navigation Bar

The top navigation bar should have a back button to allow the user to go to the previous screen upon clicking on the button.

Figure 2: Top navigation bar

3.1.3. Bottom Navigation Bar

The application must have a navigation bar at the bottom of the screen once the user is logged in to allow them to switch between different services in the mobile application.

Figure 3: Bottom navigation bar

3.1.4. Invalid input with error message

The application must display an error message when there is an invalid input provided by the user.

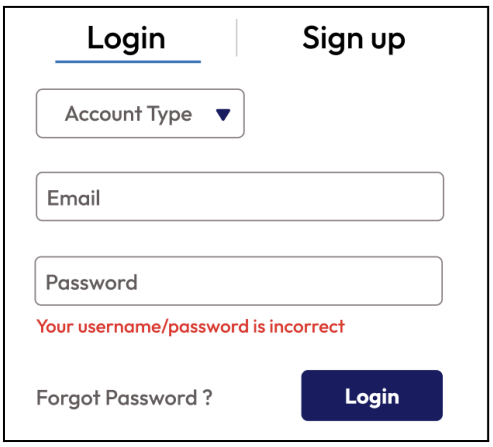
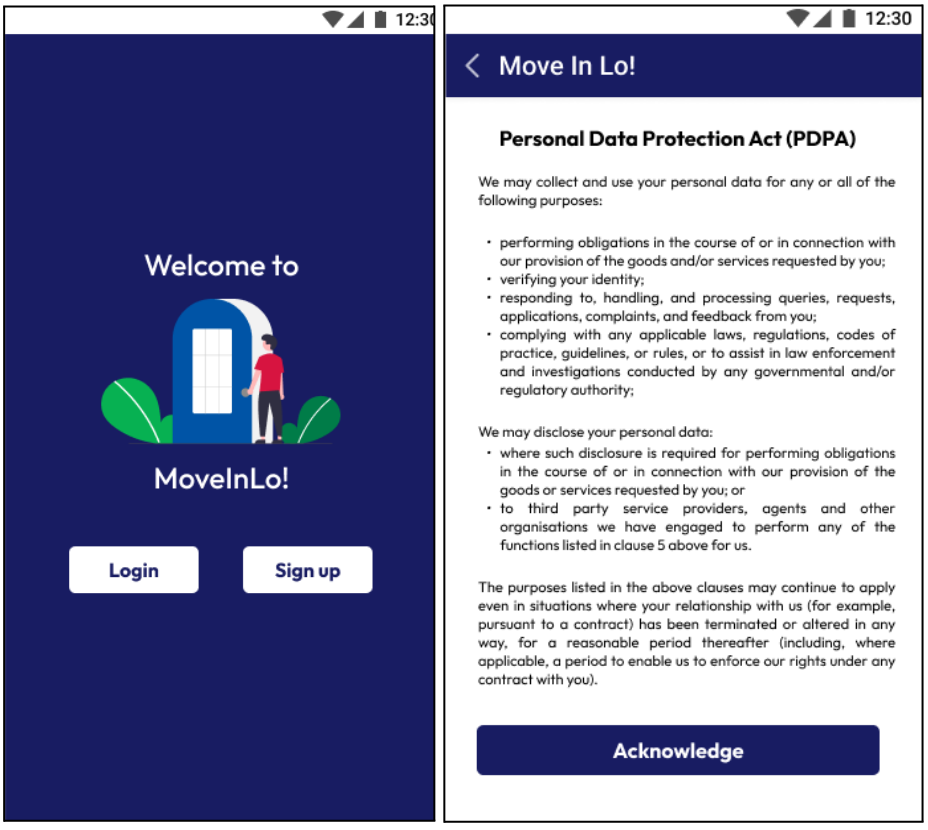


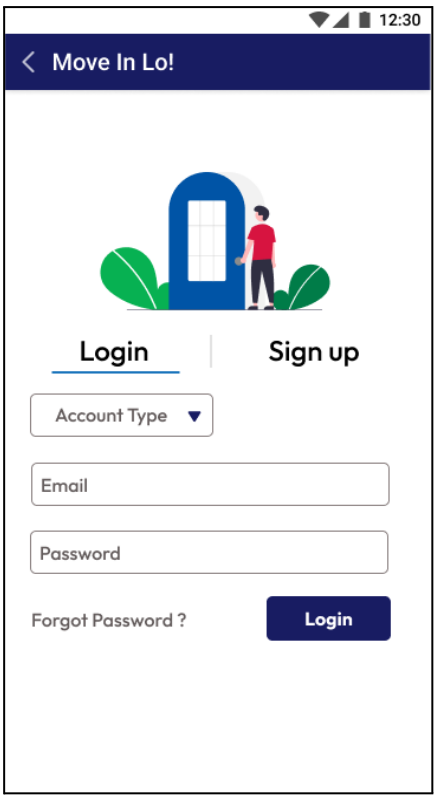
Figure 3: Error message

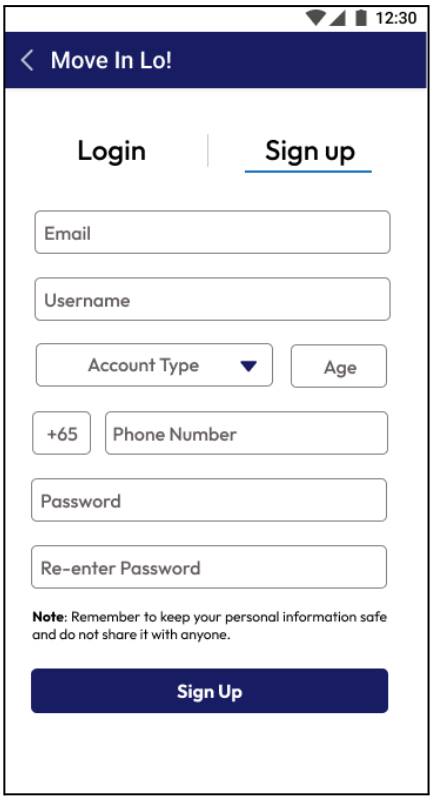
3.2. UI Mockups

3.2.1. Landing Page & PDPA Notice

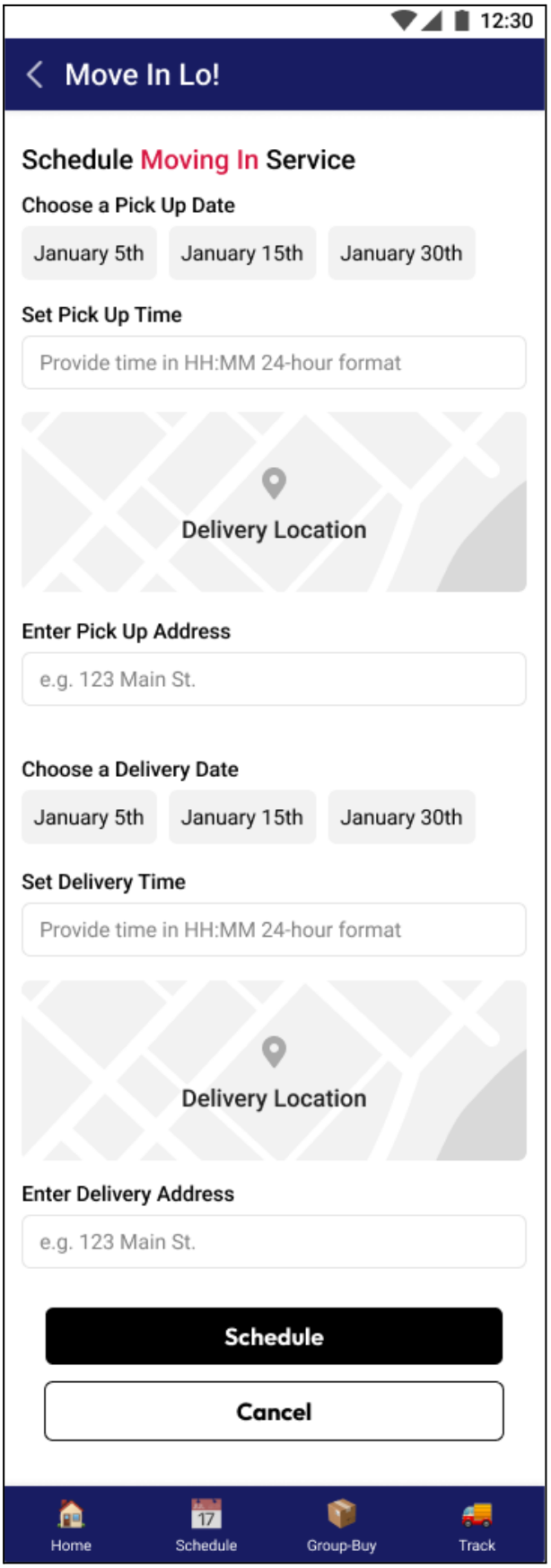


3.2.2. Login Page & Sign up Page

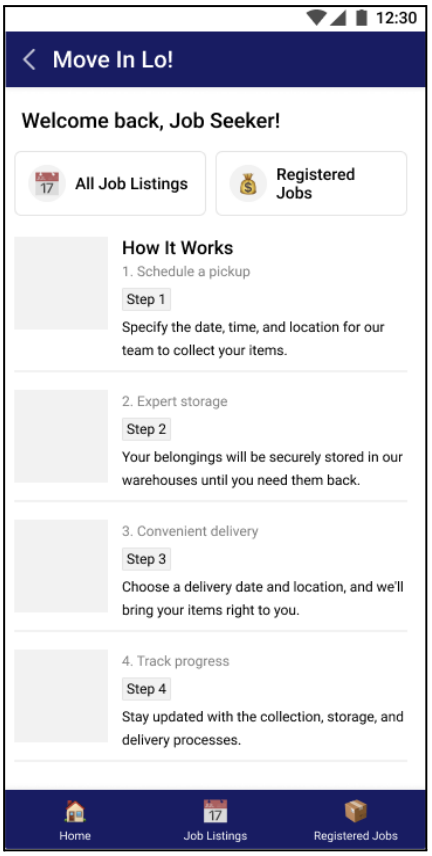


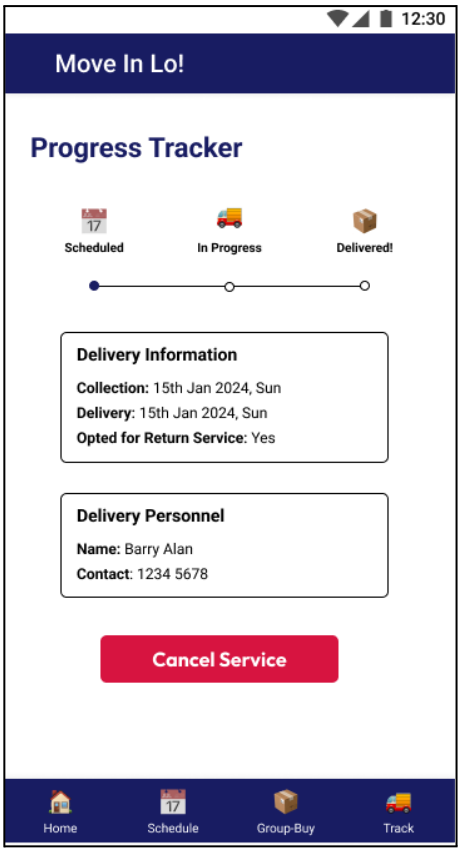


3.2.3. Schedule Moving Service

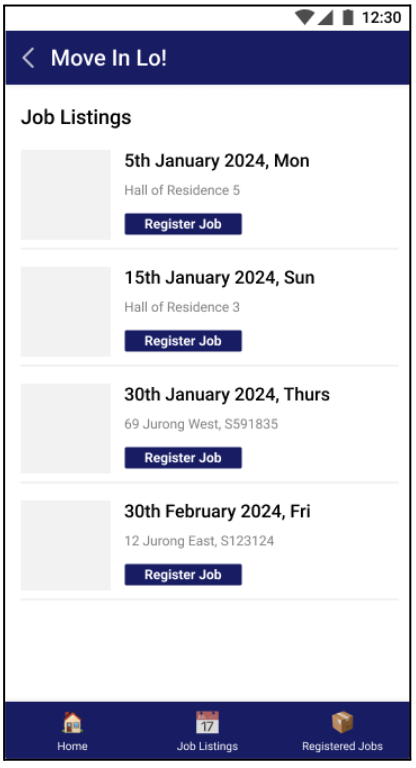


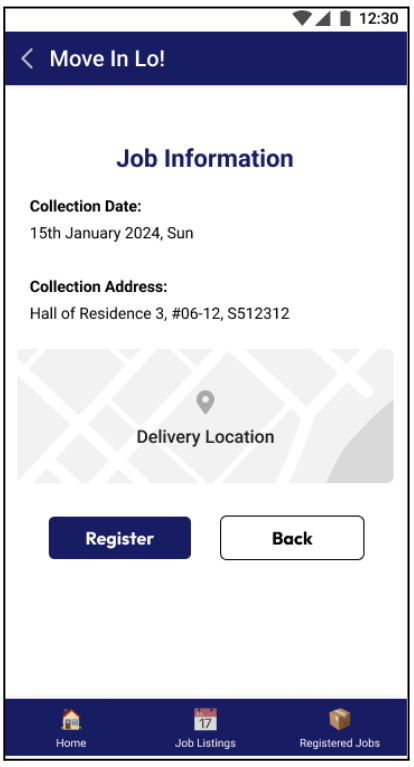
3.2.4. Job Seeker Home Page & Progress Tracker



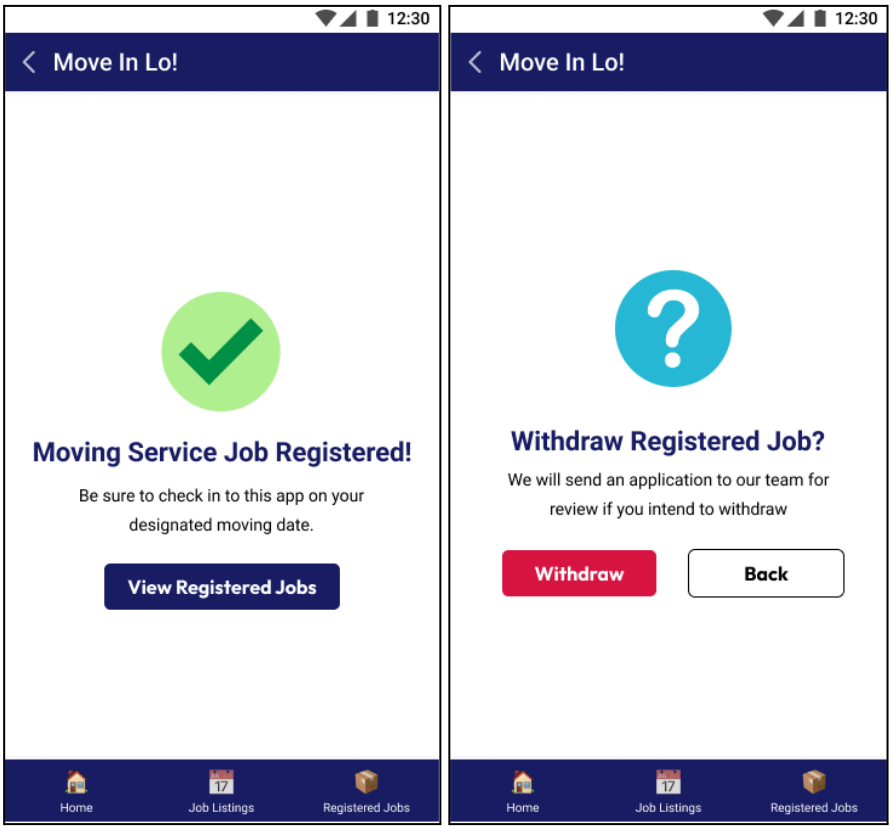


3.2.5. Job Listings & View Job

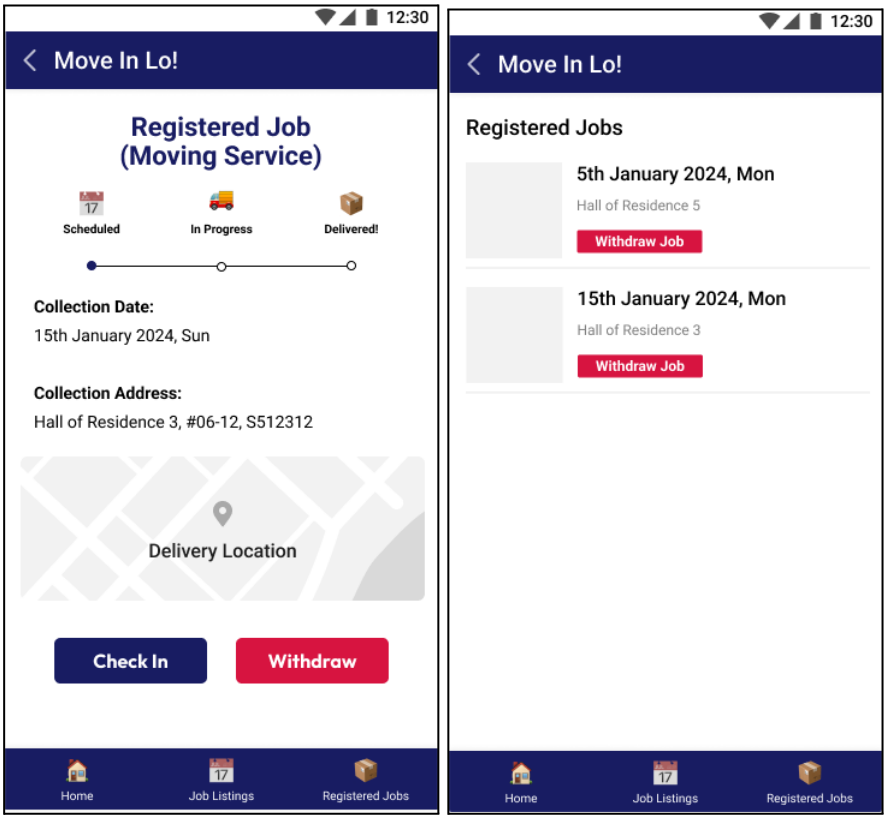




3.2.6. Register Job & Withdraw Job



3.2.7. Registered Job Information & Registered Job Lists



3.3. Hardware Interfaces

a. The device used to open this application must support Global Positioning System for the tracking of current user's location.

3.4. Software Interfaces

a. For Android devices, the OS version must be at least Android 6 (Marshmallow).
b. For IOS devices, the OS version must be at least IOS 9.

3.5. Communications Interfaces

a. The application requires internet access.
b. The application requires access to Global Positioning System (GPS) for tracking the current user's location.

4. System Features

4.1. Functional Requirements

4.1.1. User Authentication

[REQ-1] Registration as Customer

The application must allow the current user to sign up for a customer account if there is no account associated with their email address.

[REQ-2] Registration as Job Seeker

The application must allow the current user to sign up for a job seeker account if there is no account associated with their email address.

[REQ-3] Login as Customer

The application must allow the current user to login to their account as a customer using their login credentials consisting of email address and password.

[REQ-4] Login as Job Seeker

The application must allow the current user to login to their account as a job seeker using their login credentials consisting of email address and password.

[REQ-5] Forget Password

The application must allow the current user to reset their password when they provide their email address.

[REQ-6] Successful Login

The application must allow the authenticated user to access their designated home page (i.e. Customer access Customer Home Page).

4.1.2. Student Moving Services

[REQ-7] Selecting type of moving services

The application must allow the customer to select between 2 options for the moving services, moving in or moving out.

[REQ-8] Scheduling Move In services

If the customer selects the “Move In” option, the application must allow the customer to schedule a service for moving in.

[REQ-9] Scheduling Move Out services

If the customer selects the “Move Out” option, the application must allow the customer to schedule a service for moving out.

[REQ-10] Providing location for item delivery

When the customer is scheduling a moving service, the application must request the customer to provide a valid 6-digit postal code, SXXXXX within Singapore for the item delivery.

[REQ-11] Providing date for item delivery

When the customer is scheduling a moving service, the application must request the customer to select an available date provided by our system.

[REQ-12] Providing time of item delivery

When the customer is scheduling a moving service, the application must request the customer to provide a date in HH:MM 24-hour format.

[REQ-13] Display Progress Tracker

When the customer has successfully scheduled a moving service, the application must display a progress tracker to the customer in the progress tracking page.

[REQ-14] Live Location Tracking

When the user is at the progress tracker, the application must display live location tracking using GoogleMapsAPI.

[REQ-15] Update Progress Tracker for Scheduled

When the customer has successfully scheduled the moving service, the application must update the progress tracker to display “Scheduled” in the progress tracking page.

[REQ-16] Update Progress Tracker for Delivering

When the job seeker has successfully collected the items for the moving service, the application must update the progress tracker to display “In Progress” in the progress tracking page.

[REQ-17] Update Progress Tracker for Delivered

When the job seeker has successfully delivered the items for the moving service, the application must update the progress tracker to display “Delivered” in the progress tracking page.

[REQ-18] Cancellation of Moving Service

If the current user has scheduled a moving service, the application must allow the user to cancel the service.

4.1.3. Job Listings

[REQ-19] Job Seeker Guide

If the current user logged in is a job seeker, the application must display a guide to the Job Seeker, showing step-by-step instructions.

[REQ-20] Available Job Listings

The application must display a list of available jobs to the current job seeker in the Job Listings page.

[REQ-21] Moving Service Job Listing Information

When a job seeker clicks on a moving service job listing in the Job Listings page, the application must display information on the date/ time/ location of pickup and delivery.

[REQ-22] Job Registration

The application must allow the current job seeker to register for an available job listed in the Job Listings page.

[REQ-23] Registered Jobs Listings

If the current job seeker has successfully registered for jobs, the application must display a list of registered jobs to the job seeker in their Registered Jobs page.

[REQ-24] Job Completion

If the current job seeker has successfully completed a job, the application must allow the job seeker to indicate its completion.

[REQ-25] Job Payment

When a job has been completed, the application must allow the user to indicate whether payment has been completed.

[REQ-26] Job Withdrawal

If the current job seeker has successfully registered for a job, the application must allow the job seeker to withdraw from the registered job.

4.2. Use Case Descriptions & Diagrams

Refer to separate PDF document on Use Case Description.

5. Other Non-functional Requirements

5.1. Performance Requirements

5.1.1. System Responsiveness

[REQ-1] Registration and Login Responsiveness

All registration and login processes for customers and job seekers must have a response time of less than 2 seconds under normal system load to ensure a seamless user experience.

[REQ-3] Schedule Tracking Responsiveness

The schedule tracking page should load within 3 seconds to provide real-time updates to users without delays.

[REQ-4] Job Listing Retrieval Responsiveness

Retrieving and displaying job listings should take less than 2 seconds to ensure a smooth experience for job seekers.

[REQ-5] Job Registration and Withdrawal Responsiveness

Processes related to job registration and job withdrawal should have a response time of less than 3 seconds.

5.1.2. System Capacity

[REQ-6] Users Capacity

The system must be able to support 1000 simultaneous users.

[REQ-7] Database Capacity

The database should be able to store up to 50,000 user activities, each mapped to a unique user.

5.1.3. System Applicable Platforms

[REQ-8] Available Download Platforms

This application can be installed from the Android or IOS App store.

[REQ-9] Available Running Operating Systems

This application can run on Android and IOS platforms.

5.2. Safety Requirements

5.2.1. Age Limit

[REQ-10] Age restriction

Users aged under 16 will not be able to register for an account.

5.3. Security Requirements

5.3.1. Data Privacy and Security

[REQ-11] Data Encryption

All user data must be encrypted during transmission using industry standard encryption protocols to prevent unauthorised access.

[REQ-12] Data privacy

The system must comply with data privacy regulations (e.g. PDPA) to protect user data, including personal information and transaction history. User consent for data handling must be obtained and documented.

[REQ-13] User Reminder for Security

The system must remind the users not to leak their personal information, username, and password.

5.3.2. User Account Policies

[REQ-14] Password Policies

The system should enforce the following password policies, including complexity requirements. The password must include the use of both upper-case and lower-case letters, one or more numerical digits and special characters, such as @, #, $.

[REQ-15] User Credentials

Users are required to login with their email and password.

5.4. Software Quality Attributes

5.4.1. Adaptability

[REQ-16] Adaptability Across Platforms

The system will be able to operate on both IOS and Android platforms.

5.4.2. Reliability

[REQ-17] System Reliability

The system must be reliable, with a target mean time between failures of at least 10,000 hours.

5.4.3. Maintainability

[REQ-18] Code maintainability

The system code must be provided with necessary comments by developers to make it easy to understand.

5.5. Business Rules

5.5.1. Developer

[REQ-19] User Removal

Developers can remove users not following the community guidelines.

[REQ-20] Activity Removal

Developers can remove activities not following the community guidelines.

6. Appendix A: Data Dictionary

a. Customer
    i. A user which engages in moving-related services, and is associated with an account, granting access to the Moving Scheduler feature of the application.
b. Job Seeker
    i. A user which takes up paid moving-related jobs, and is associated with an account, granting access to the Job Schedule feature of the application.
c. Database
    i. A database is a structured collection of data organised into tables, where data can be efficiently created, retrieved, updated, and deleted. It serves as a central repository for storing and managing information.
d. API
    i. An API (Application Programming Interface) is a set of rules, protocols, and tools that allows different software applications to communicate and interact with each other. It defines the methods and data formats that applications can use to request and exchange information.
e. Moving Service
    i. A Moving Service is a service performed by a Job Seeker to transport items from the designated collection point to the designated delivery point. It includes both “Move In” and “Move Out” services.
f. Payment
    i. Payment is the process when the Customer pay money to the Job Seekers. The activities that need Payment service include Moving Service and Group Buying.
g. Job Scheduler
    i. A Job Schedule is a system in the application designed to manage and display available jobs to Job Seekers. It serves to help match job opportunities to potential Job Seekers.
h. Progress Tracker
    i. A Progress Tracker is a feature within the application that monitors and records the progress of the moving-related task. It provides real-time updates and visual representations to help users track the progress of the movement. It contains “Scheduled”, “In Progress” and “Delivered”

i. Available Job
    i. A job is a job listing in the database that has yet been assigned to a job seeker. It represents a job opportunity for the Job Seeker to register for.
j. Registered Job
    i. A Registered Job is a job listing in a database which has been allocated to a job seeker and is actively stored in the job listing database.
k. Withdrawn Job
    i. A Withdrawn Job refers to a job listing in a database that was previously available or registered but has been removed by the previously assigned job seeker.
l. User Interface (UI)
    i. A User Interface is the visual aspect of the software application that allows users to interact and control it. It features components that accept user inputs and which accesses the functionality of the application.
m. Landing Page
    i. The initial page which is rendered to the user when they open up the application.

7. Appendix B: References

a. Google Maps API Platforms: https://developers.google.com/maps/faq#setup_billing
b. NTU Color Scheme: https://www3.ntu.edu.sg/CorpComms2/NTU%20Quick%20Brand%20Guide%202018.pdf