CSG3101 – Applied Project

Project Proposal

Cargo Shipping System

Table of Contents

Project Name
Cargo Shipping System

Project Goal
The goal of this project is to build a mobile application that directly connects small scale ship owners directly to customers at click of fingertip.

Supervisor
Mr. Buddhika Gunasekara

Team

Student Name	ECU Number
Sanojan Thirunarukkarasu	10527055
Thabitha Sylvester	10508998
Shenuki Perera	10527054
Nazhath Sulthana	10527096

Background
Cargos are known as the best form of international transport in regard to goods that are of heavy weight which require a more reliable and secure source of transportation. Life could be made much easier if individuals, shipping agents and ship owners from all around the world could connect one platform(application) that could enable shipping much easier leading to certain advantages such as reduced paper work and avoid third parties costs such as commissions and handling fees.

Currently applications like FedEx, UPS, DHL provide these services, but customers have to pay excess fees like commissions, excess charges service charges, and some applications do not provide the facility to select the date frame the customer requires and customers also do not have the option to compare prices.

Scope
This application connects the registered customer (the person which requires the transportation of the good) to shipping agents, where the connected user can establish communication and request shipment of their goods to the intended country/port. The customers can find a shipment more convenient to them by entering the port they plan on shipping from and the destination port, the app will display a list of available shipments and price per kg. according to the criteria. The customer will fill out a form providing necessary information which will be sent to the ship owner. Once the relevant payments have been made the shipping will agent will change customer order status from “processing’’ to “approved”, then the customer will drop-off the package at the relevant location, the package will then be loaded and shipped to the intended country where the shipment will be completed through this cargo system once the package in unloaded.

The ship agents will use application to keep the customers informed about their shipment schedules. They will maintain up-to-date schedules for each ship they own. They can view a list of customers for each shipment and update their status(processing/approved) once the payments have been made. The ship owners can request to add a new shipment from the admin by submitting a request form.

The Admin uses this application the maintain customer and ship owner profiles. They can add new customers, ship owners and ships.

Functional Requirements

- The system will manage 3 types of users: Customer, Ship Agent, Admin.
- The system will allow a potential customer to browse through the system as a guest.
- The system will allow the guest to track a shipment and to view shipping schedules.
- The system will allow a customer to book a shipment.
- The system will allow a shipping agent manage ship details, update/manage schedules, approve customers for a shipment and view details of a single shipment.
- The system will allow the Admin to add new Shipping agents and ships.
- The system will allow Admin the manage customer and Ship Agent details.

Non-Functional Requirements.

- The system will be accessible anywhere around the globe granted there is stable internet.
- The system should be real-time system with latency period of 30 seconds
- Response time should be minimal, and it should be less than two minutes.
- The system shall operate properly 24/7
- Expected overall system reliability is 99.0%

Out of Scope

- Only supports mobile applications with medium to large screen resolutions (does not support smart watches, Smart TV's).
- The application will not handle the pickup and drop-off of packages.
- The application is **only** tested on android devices.
- The application will not handle payments.

Schedule

We have chosen to utilize an Incremental approach where at the conclusion of each week a prototype will be presented to the supervisor for assessment and feedbacks. Since majority of the technologies need to be researched and learnt therefore learning curve is high during first few weeks of the project which will be used to learn and map out the application structure. A set of weekly documents such as weekly status report, weekly action plan, meeting minutes, agenda and report on testing results will be produced for each week as an evidence for the progression of work being done.

Week	**Date**	**Appointed to**	**Task**
Week 01		Everyone	Group Formation
		Everyone	Discussing and finalizing project idea.

	27/07/2020 to 30/07/2020	Everyone	Research on project – Functional Requirements
		Everyone	Research on suitable tools and technologies.
		Everyone	Research on React native, Firebase and other technologies
Week 02	01/08/2020 to 07/08/2020	Everyone	Functional Requirements
		Everyone	Use case diagram draft
		Thabitha	Initial Mock app screens
		Sanoj	Finalizing technology (React native and Firebase) and researching learning tutorials
		Everyone	Finalize and sign Team Contract
		Shenuki	Draft of Project Proposal
		Sulthana	Project Schedule draft
Week 03	08/08/2020 to 14/08/2020	Everyone	Finalize project scope, functional requirements and UCD.
		Everyone	Continue working on the project proposal
		Everyone	Learn technologies.
Week 04	15/08/2020 to 21/08/2020	Everyone	**Milestone: Deliverable – Project Proposal**
		Everyone	Setup project environment – GitHub, OneDrive
		Everyone	Learn technologies.
Week 05	22/08/2020 to 28/08/2020	Shenuki, Sanoj	Guest User Interface – home page, top menu, sign-in, sign-up.
		Thabitha, Sulthana	Testing prototype
		Thabitha, Sulthana	Create Weekly Documentation
Week 06	29/08/2020 to 04/09/2020	Shenuki, Sanoj	Creating Customer User Interfaces - Schedule, Book, form, top menu.
		Thabitha, Sulthana	Testing prototype
		Thabitha, Sulthana	Create Weekly Documentation
Week 07	5/09/2020 to 11/09/2020	Shenuki, Sanoj	Creating Ship Owner Interfaces – homepage, Ships, Schedule, top menu
		Thabitha, Sulthana	Testing prototype

		Thabitha, Sulthana	Create Weekly Documentation
Week 08	12/09/2020 to 18/09/2020	Shenuki, Sanoj	Creating Admin Interfaces – homepage, top menu.
		Thabitha, Sulthana	Testing prototype
		Thabitha, Sulthana	Create Weekly Documentation
Week 09	19/09/2020 to 25/09/2020	Shenuki, Sanoj	Mapping out Database configuration and connection for Admin functions
		Thabitha, Sulthana	Testing prototype
		Thabitha, Sulthana	Create Weekly Documentation
Week 10	26/09/2020 to 2/10/2020	Shenuki, Sanoj	Database configuration and connection for Ship owner functions
		Thabitha, Sulthana	Testing prototype
		Thabitha, Sulthana	Create Weekly Documentation
Week 11	03/10/2020 to 09/10/2020	Shenuki, Sanoj	Database configuration and connection for Customer functions
		Thabitha, Sulthana	Testing prototype
		Thabitha, Sulthana	Create Weekly Documentation
		Everyone	Improving UI
		Everyone	User Acceptance Testing
Week 12	10/10/2020 to 16/10/2020	Everyone	**Milestone: Deliverable – Project Presentation**
		Everyone	Improving UI
		Everyone	User Acceptance Testing
Week 13	17/10/2020 to 23/10/2020	Everyone	Making last minute changes, bug fixes and testing
Week 14	24/10/2020 to 30/10/2020	Everyone	**Milestone: Deliverable – Final Project Output**
		Everyone	**Milestone: Deliverable – Contribution Reflection**

Team Capability

Team Members	Roles	Capabilities, Skills
Sanojan Thirunavukkarasu	Developer, Tester	• Strong communication skills. • Programming skills in Python, C++, java, JavaScript, basics of React. • Experience in developing black box test cases using Junit.
Thabitha Sylvester	Document handler, Tester	• Programming skills in Python, C++, PHP, ASP.NET. • Developing web pages using HTML, ASP.NET and PHP. • Excellent document handling skills with an excellent command of English. • Experience in developing black box test cases using Junit.
Shenuki Perera	Team Leader, Developer, Tester	• Quick decision making and strong communication skills. • Programming skills in Python, C++, java, HTML, basics of React. • Experience in developing black box test cases using Junit. • Developing Desktop applications using Java and Swing.
Nazhath Sulthana	Document handler, Tester	• Strong communication skills with an excellent document creating skills. • Programming skills in Python, C++, java. • Experience in developing black box test cases using Junit.

Tools and Technical Requirements

Hardware – laptops with Windows OS for building the application and Android mobile devices for testing. (Owned by team members)

Software - Android Studio IDE version 4, Node Js version 12.18.3, Firebase version 7 Web SDK, database facilities. Expo CLI (virtual device testing) etc. will be obtained freely from internet sources.

Version Control – OneDrive to create a centralized document repository and GitHub for source control.