CCAS 4.3: – Software Engineering
Milestone 1: Requirements Specification

Project: Travel Agency Management System

Team: Jana Farghal (202201414), Ali Osman (202202499), Abdulrahman Moustafa (202200373)

1. System Overview

The Travel Agency Management System is a web-based platform designed to help a travel agency manage trips, customers, bookings, and payments efficiently. The system provides travel-related services such as trip packages, hotel and airline reservations, and booking management for three main users: Customers, Travel Agency Staff, and System Administrators.

2. Functional Requirements

Functional requirements describe what the system must do, categorized using the MoSCoW method.

2.1 Must Have Requirements

REQ_001 – The system must allow users to register and log in.

REQ_002 – The system must allow customers to view available trips.

REQ_003 – The system must allow customers to book trips.

REQ_004 – The system must store customer information and booking details.

REQ_005 – The system must allow administrators to add new trips.

REQ_006 – The system must allow administrators to update or delete trips.

REQ_007 – The system must allow customers to make payments for bookings.

2.2 Should Have Requirements

REQ_008 – The system should allow customers to cancel bookings.

REQ_009 – The system should send booking confirmation messages.

REQ_010 – The system should allow administrators to view booking reports.

2.3 Could Have Requirements

REQ_011 – The system could support discount offers.

REQ_012 – The system could allow customers to rate trips.

2.4 Won't Have Requirements

REQ_013 – The system will not support real-time flight tracking in this version.

REQ_014 – The system will not support multiple languages in this version.

3. Non-Functional Requirements

REQ_015 – The system shall be available through modern web browsers.

REQ_016 – The system shall protect user data using secure login methods.

REQ_017 – The system shall store passwords in encrypted form.

REQ_018 – The system shall support up to 1,000 users at the same time.

REQ_019 – The system shall have a simple and easy-to-use interface.

REQ_020 – The system shall be scalable for future upgrades.

4. Assumptions and Constraints

The system will be web-based.

A relational database (PostgreSQL via Supabase) will be used.

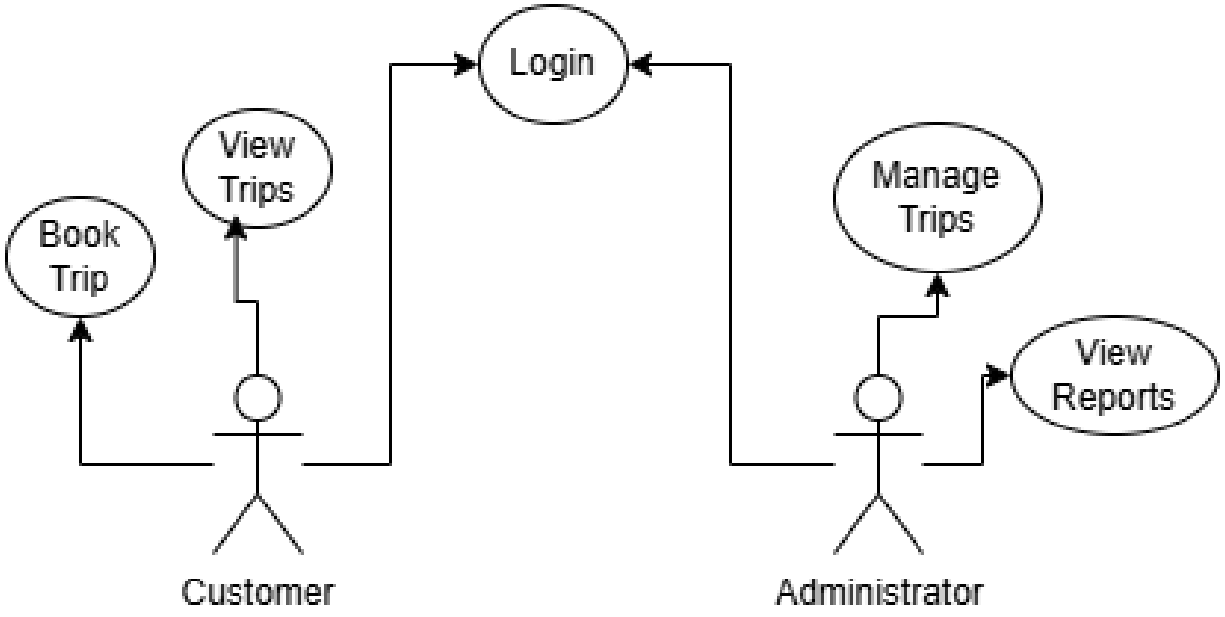
Online payment will be simulated or handled by a third-party service.

5. Use Case Modeling

This section defines the interactions between the users (Actors) and the system to fulfill the requirements listed above.

5.1 Use Case Diagram

The diagram below was created using Draw.io and illustrates the relationship between the Customer and Administrator and the core system functions.



5.2 Use Case Descriptions

Use Case ID: UC_01 – Customer Trip Booking

Actor: Customer

Related Requirement: REQ_002, REQ_003, REQ_007

Description: The process by which a customer selects and pays for a travel package.

Flow of Events:

1. Customer logs into the system.
2. Customer browses the "Trips" dashboard (REQ_002).
3. Customer selects a trip and clicks "Book Now."
4. Customer enters traveler details.
5. Customer submits payment information for simulation (REQ_007).
6. System validates seats and saves the booking (REQ_004).

Post-condition: The database reflects a new booking and reduced seat availability.

Use Case ID: UC_02 – Administrative Trip Management

Actor: System Administrator

Related Requirement: REQ_005, REQ_006

Description: The process of maintaining the inventory of available travel services.

Flow of Events:

1. Admin logs into the secure portal.
2. Admin selects "Add Trip."
3. Admin enters destination, price, and available seats.
4. Admin clicks "Save."

Alternative Flow: Admin selects an existing trip to "Update" or "Delete" (REQ_006).

Post-condition: The changes are immediately visible to customers on the frontend.

4. Assumptions and Constraints

- The system will be web-based
- A relational database will be used
- Online payment will be simulated or handled by a third-party service

5. Conclusion

This document defines the functional and non-functional requirements for the Travel Agency Management System. These requirements will be used as the foundation for database design, system development, and testing in the next milestones.