# Solace

20.02.2022


## Team Take-off

Aliza Saleem Lakhani (al05435)
Hafsa Irfan (hi05946)
Mohammad Bilal Fayaz (mf05942)
Sudais Baig (sb05588)

# Table of Contents


I. Introduction
    1. Purpose
    2. Scope of the Product
II. System Requirements
    1. Functional Requirements
    2. Non- Functional Requirements
III. Use Cases
IV. Class Diagram
V. Sequence Diagrams

# I. INTRODUCTION

## 1. Purpose

This document provides a comprehensive architectural overview of the system, using several different architectural views to depict the working and different aspects of the system.

## 2. Scope of the Product

According to a report published by Times about 146,000 people were killed in road accidents in India/Pakistan in the year 2016. Unfortunately, about 30% of deaths are caused due to delayed ambulances. More government statistics show that more than 50% of heart attack cases reach the hospital late, which can constitute the unavailability of ambulances too but the majority of it is due to patients stuck in traffic. This great percentage of the deaths occur due to the delayed arrival of an ambulance in emergencies. Typically, this occurs when patients are unable to quickly contact local hospitals or provide appropriate information about their whereabouts in these circumstances. Also, most of the ambulance services in the local region have very little infrastructure to track the whereabouts of their ambulances and respond quickly to any situation. This is one of the major reasons, along with the deteriorating conditions of roads and bad traffic situations, that leads to a majority of deaths caused in ambulances. Our goal is to solve this problem of quick service ambulance booking by our application, Solace.

# II. Functional Requirements

1. User Signup Page: To allow users to create and maintain their personal profiles
2. Booking an ambulance
3. The ability to login to your profile and access saved location
4. To view different kinds of ambulances available
5. Drivers should be able to view all booking requests

# III. Non-Functional Requirements

## 1. Reliability

The system will be subjected to the following testing operations before being deployed.:

- QA Testing: Testing for responsiveness, Unit testing.
- Software Testing: Testing for bugs and vulnerabilities.

## 2. Performance

The system performance may depend on the hardware, network, and internet connection. However, the system responds to requests within 30 seconds i.e under standard database and web server script timeouts.

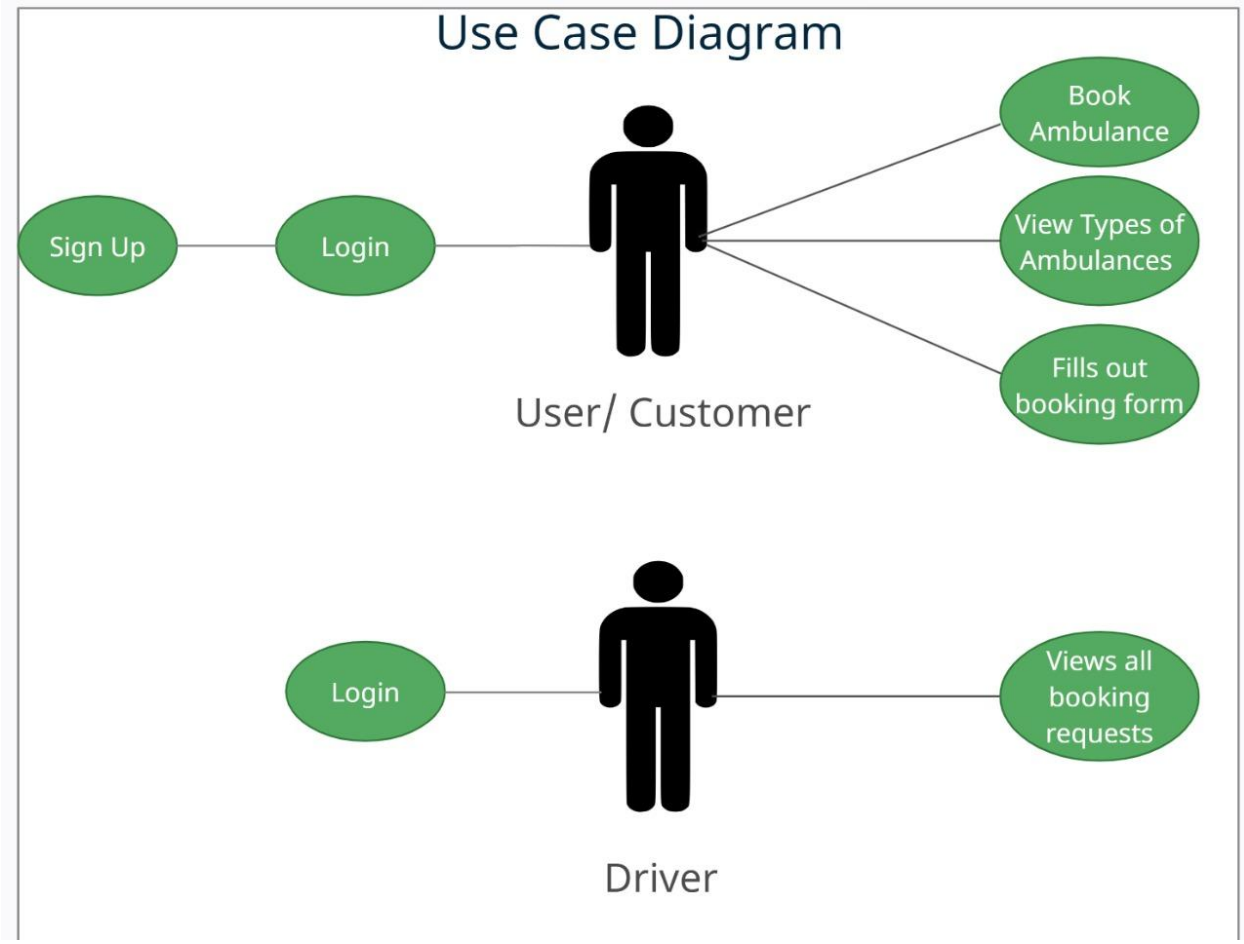
## 3. Development Tools

1. Framework & Programming Language: The .NET framework 4.7.2 - C# - Client/Server Architecture
2. Visual Design: Html, CSS
3. Software Development Tool: Microsoft Visual Studio 2022
4. Database: Microsoft SQL Server

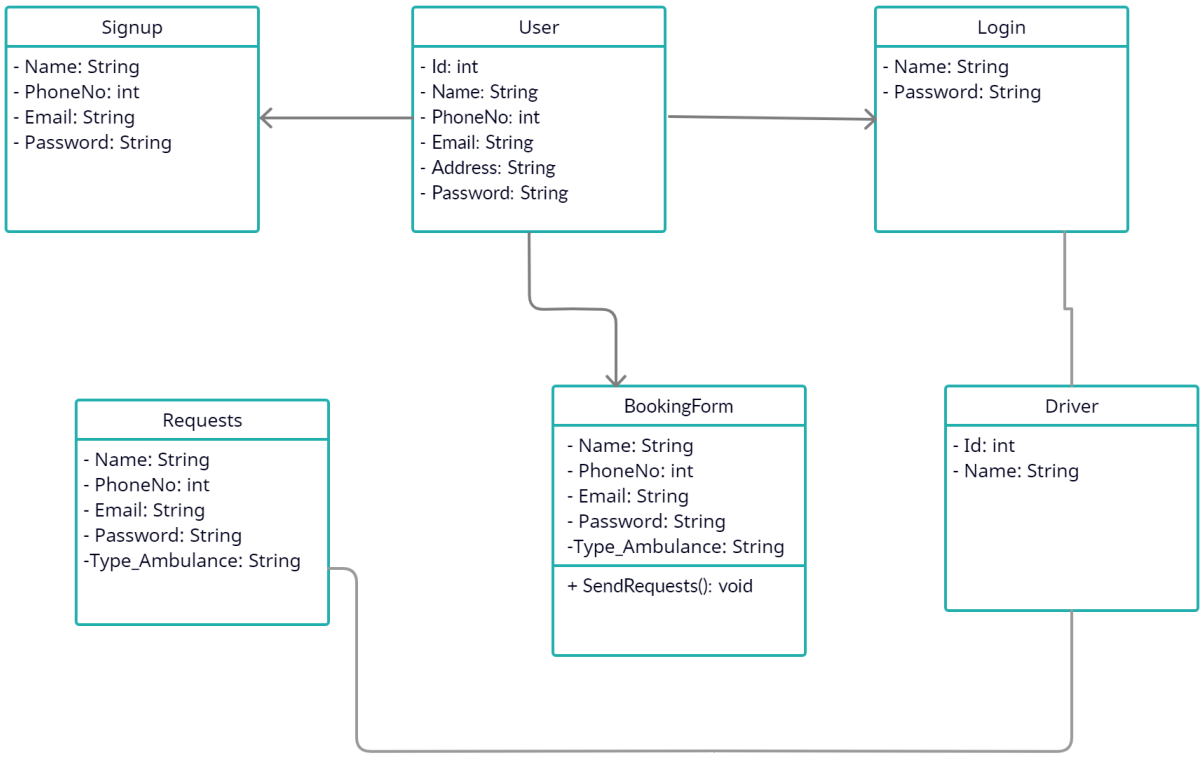
# IV. Use Cases:

1. User-specific use cases:
    - User signs up and creates an account
    - User logs into the system
    - User views the types of ambulance
    - The user fills out the booking form
2. Driver specific use cases:
    - Driver logs in
    - Driver views all the requests made by the users

Below is the use case diagram:

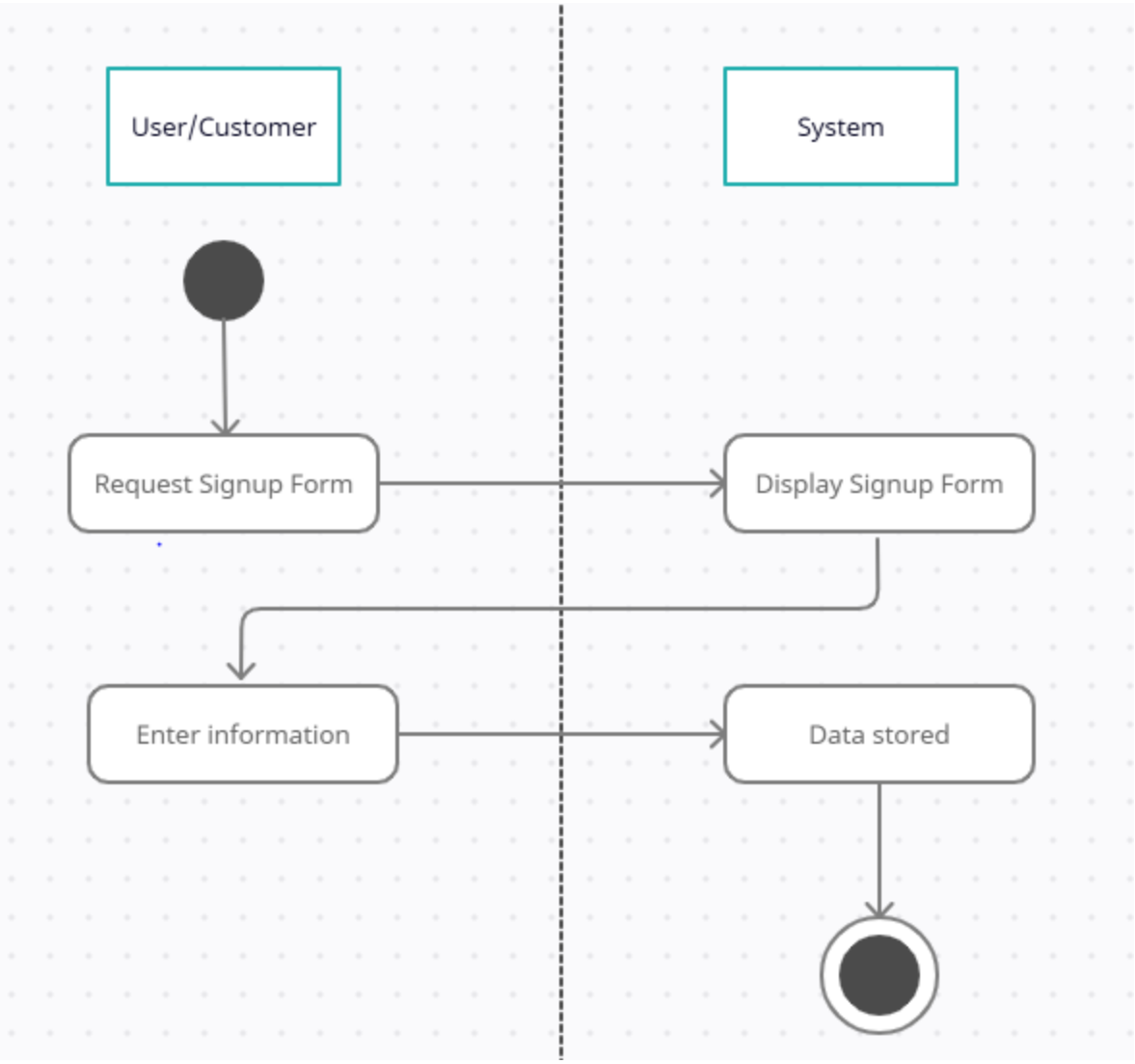


# V. Class Diagram

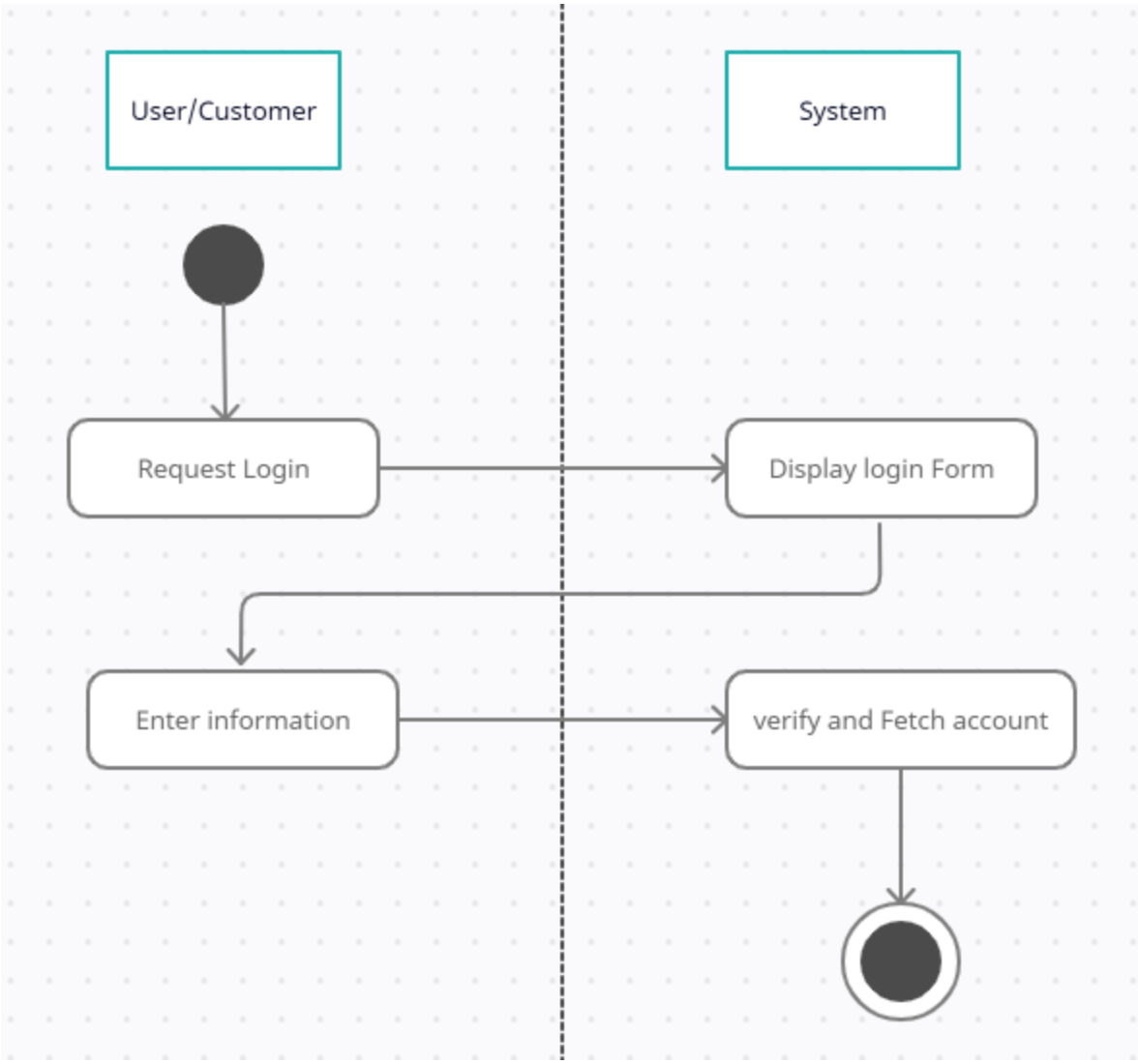


# VI. Sequence Diagrams

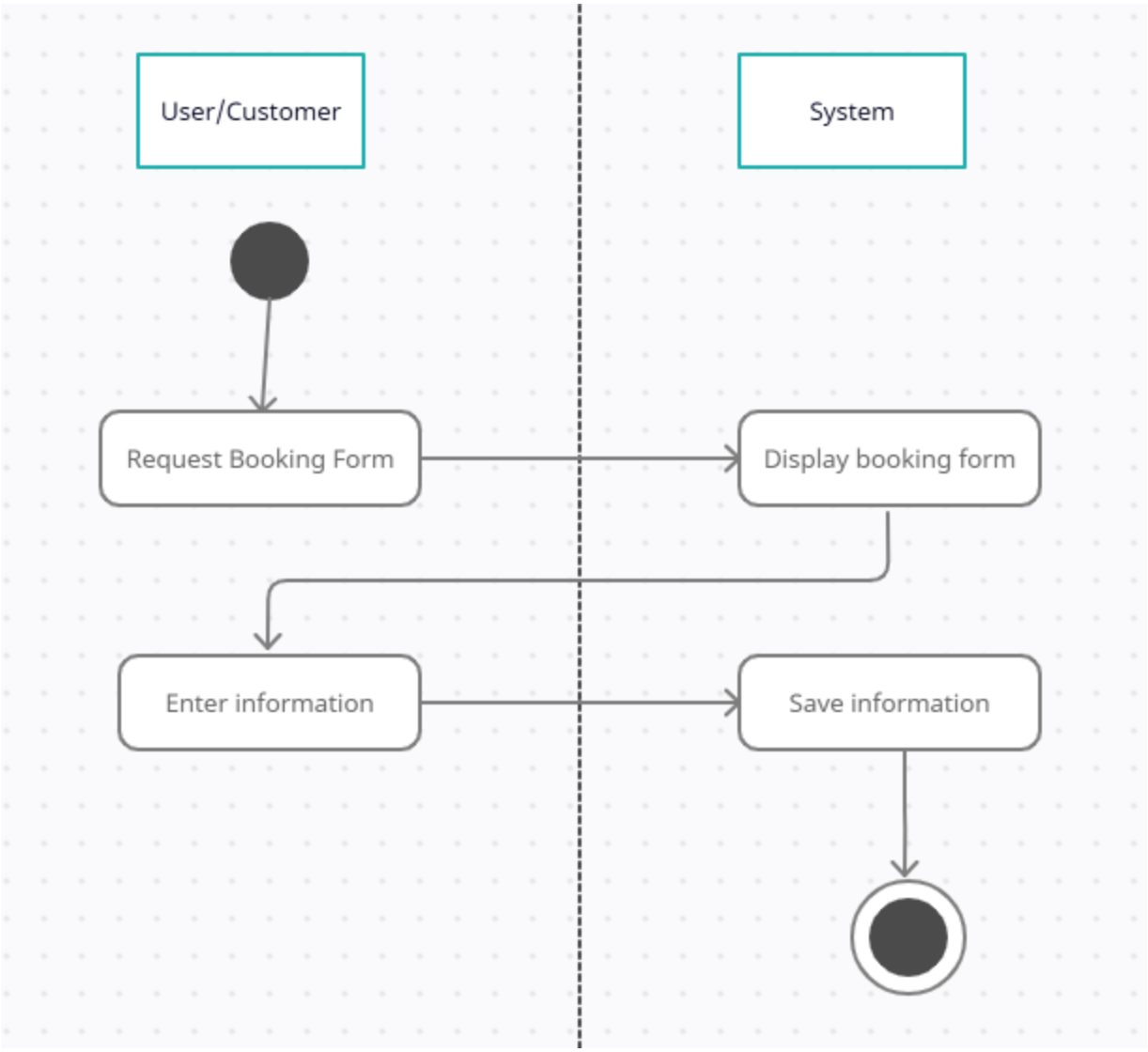
1. User Signs up



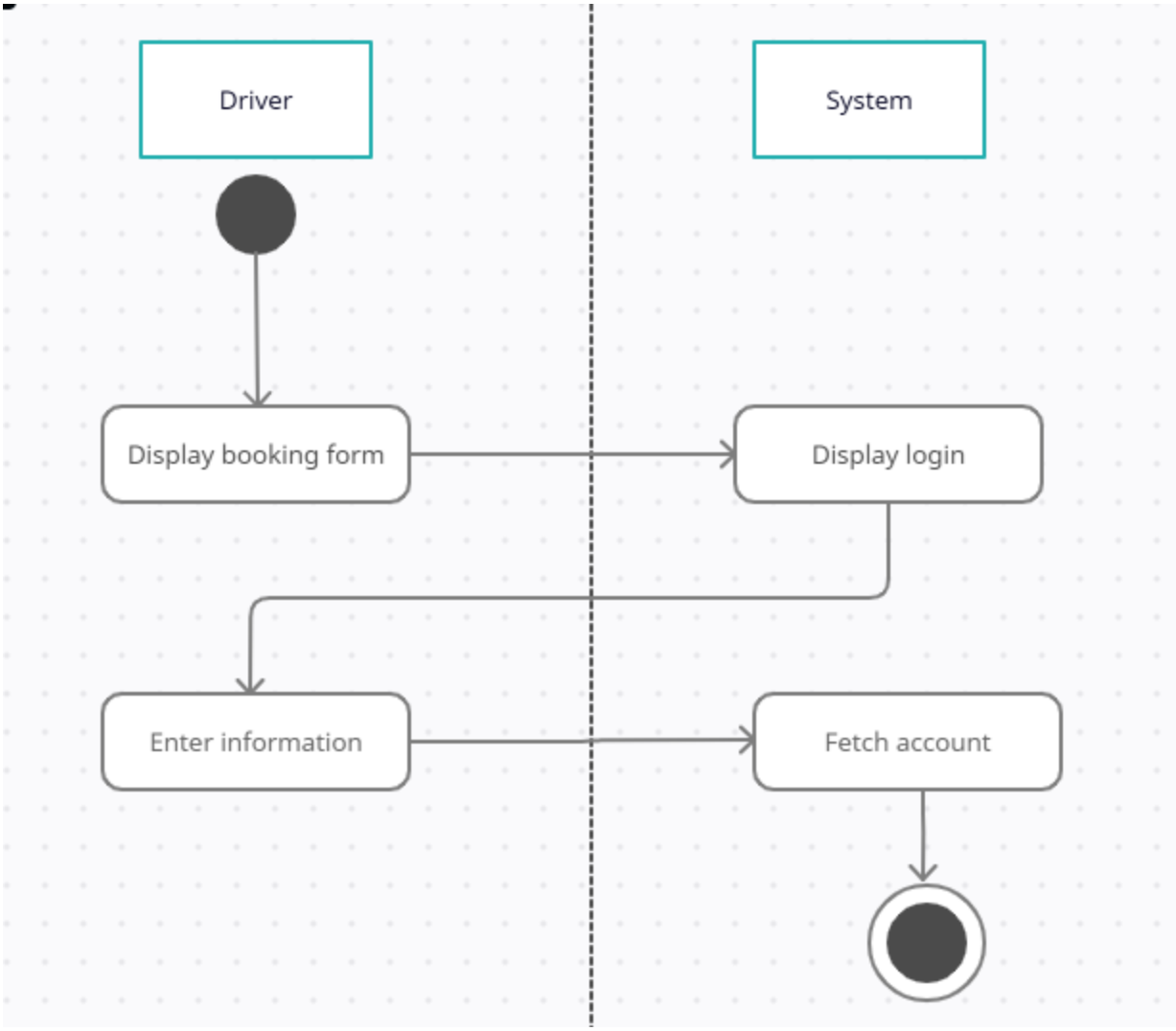
2. User logs in



3. The User fills out the booking form



4. Driver logs in



5. Driver views all the bookings made