Requirement Document

for Elevator System

Team 5 Project 1

Made by : Yang Runkang

* Table of Contents
* Introduction
* Project Overview
* Basic Requirement
* Domain Analysis
* UML Illustration
* Use Case Diagram
* UML Activity Diagram
* UML Sequence Diagram
* Detailed Requirement
* Overview
* Passengers’ Perspective
* Visual Components
* Elevator’s Perspective
* Control System

Introduction

Project Overview

The scope of this project includes the deployment of a dual-elevator system to service a commercial/office building which spans three floors. The building should satisfy the need for efficient vertical transport to facilitate the movement of people.

Basic Requirement

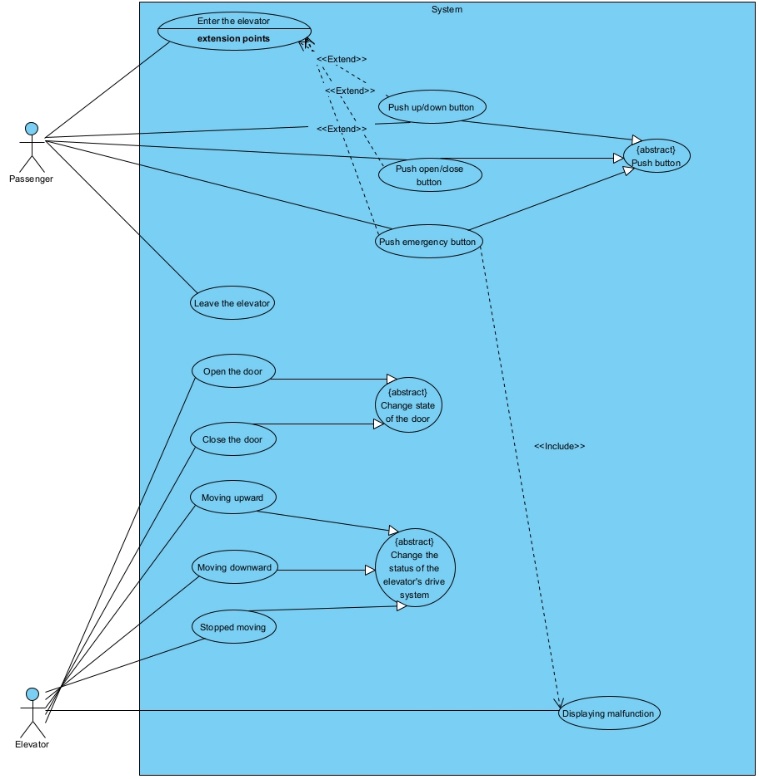
In each elevator car, there are floor buttons. On each floor, there is a set of up and down buttons. When passengers enter the elevator, they press the floor buttons, which illuminate when pressed, and then notify the elevator to travel to the desired floor. When the elevator arrives at the destination floor, the buttons stop flashing. If necessary, passengers can press the emergency button, which automatically sends a distress signal indicating that the elevator needs technical repair. Apart from that, a control system need the integration of a decision-making controller to allocate elevator resources dynamically. Installation of an interactive panel within each elevator to show the elevator's current position, direction.

Domain Analysis

The system primarily involves two actors, namely passengers and elevators. Passengers should be able to control elevator operations through actions such as pressing buttons, while elevators should be able to transport passengers to corresponding floors through actions such as opening and closing doors. When a passenger presses the emergency button, it triggers the elevator to stop operation. The UML use case diagram for this process is shown below, with passengers and elevators being the two main use cases in the system. The relevant behaviors of passengers and elevators are depicted in the following UML use case diagram.

UML Illustration

Use Case Diagram

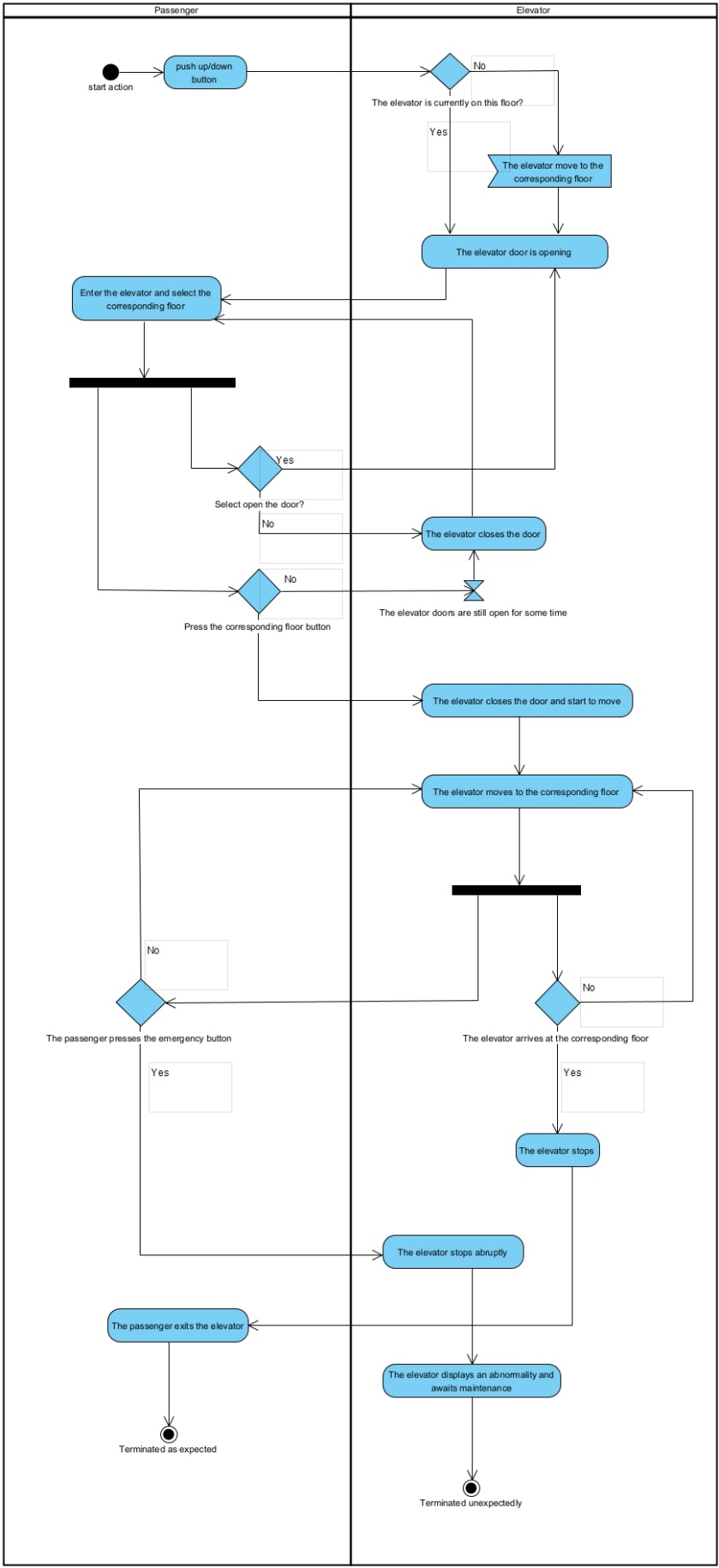


**The use case diagram includes several fundamental functions**

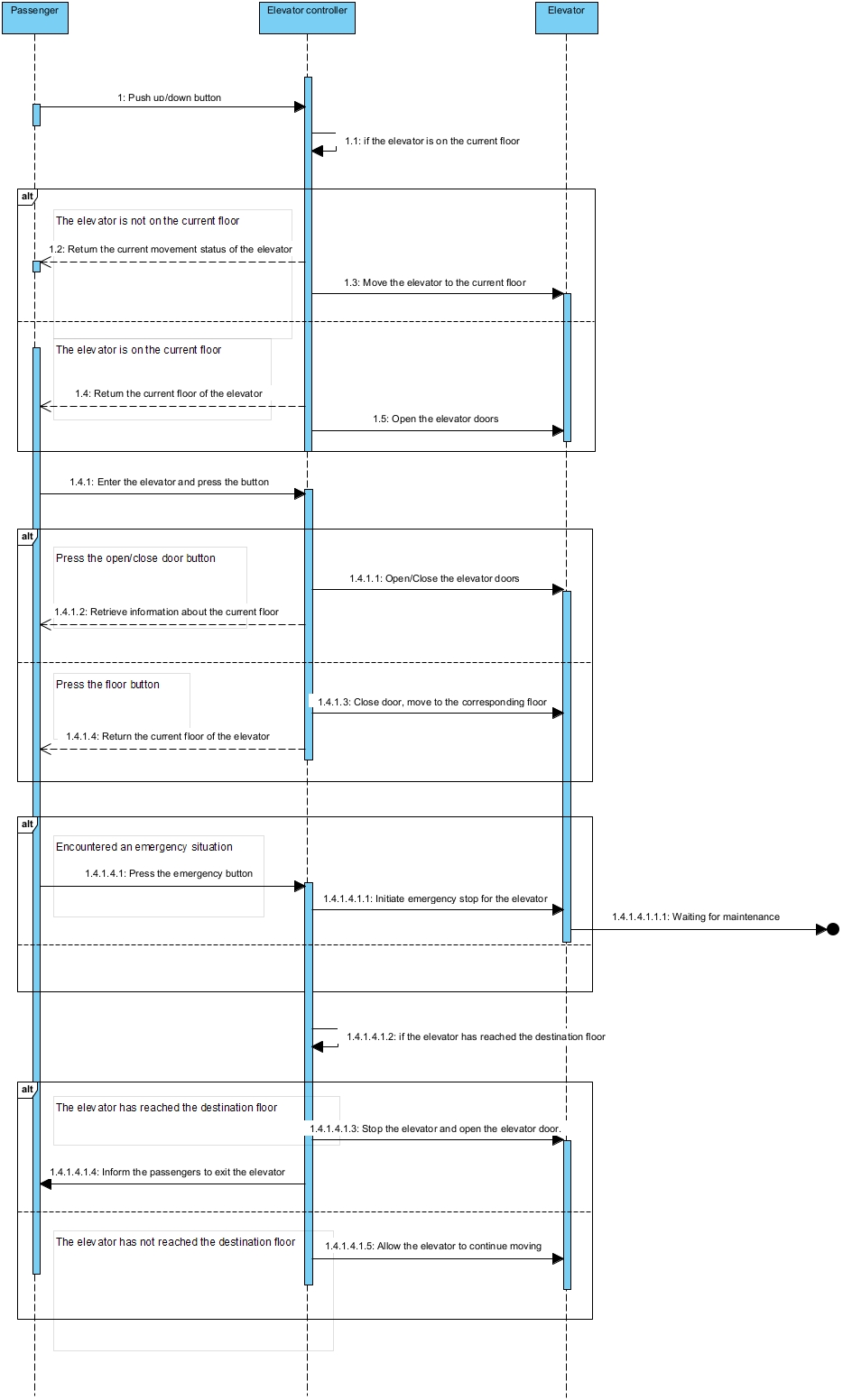
* **Enter the elevator**: This use case begins when the passenger approaches and enters the elevator, initiating interaction with the system.
* **Leave the elevator**: This occurs when the passenger exits the elevator at their desired floor.
* **Push up/down button**: Represents the action of the passenger pressing the button to select the floor they wish to travel to.
* **Push open/close button**: Involves the passenger pressing the button to open or close the elevator doors manually.
* **Push emergency button**: This use case is activated when the passenger uses the emergency button, typically to alert for help or report an issue.
* **Open the door**: The system enables the doors to open, either automatically upon reaching a floor or in response to a button press.
* **Close the door**: Corresponds to the elevator doors closing, either automatically after a delay or through a button press.
* **Moving upward**: The elevator ascends to a higher floor as requested by the passenger.
* **Moving downward**: The elevator descends to a lower floor as requested by the passenger.
* **Stopped moving**: The elevator ceases movement, typically upon reaching the selected floor or due to an emergency or malfunction.

UML Activity Diagram

The interaction between passengers and the elevator can be depicted through the following two UML diagrams: an Activity Diagram and a Sequence Diagram.



UML Sequence Diagram



Detailed Requirement

Overview

Overall, the main participants in this elevator system are the passengers and the elevator. The specific interaction process between them can be divided into interactions with the panel and the elevator control system.

R1. Passengers’ Perspective

1. **For passengers, they should be able to:**
   1. Know the floor information where the elevator is located.
   2. Understand the current operating status of the elevator (ascending/descending/stationary).
   3. Outside the elevator, press the corresponding floor button according to their destination floor, thereby controlling the elevator to reach the current floor of the passenger, and convey this information to the panel.
   4. When the elevator reaches the passenger's floor, they can control the opening and closing of the elevator doors at that floor, with the highest priority given to the operations performed by passengers on that floor.
   5. Inside the elevator, they can press the button for their destination floor to control the elevator to travel to the corresponding floor, and convey this information to the panel.
   6. Press the emergency help button at any time, and convey this information to the panel.

R2. Visual Components

1. **For the panel:**
   1. The buttons for controlling the opening and closing of the elevator doors are ineffective during the operation of the elevator.
   2. Receive all relevant information conveyed by passengers through the panel, and pass this information to the elevator control system.
   3. Instantly display the current floor information of the elevator, the current operating status of the elevator (ascending/descending/stationary), and whether the elevator is malfunctioning, etc., and pass this information to the elevator control system.

R3. Elevator’s Perspective

1. **For the elevator itself, it should be able to:**
   1. Receive signals from the elevator control system to ascend/descend/stay stationary.
   2. Receive signals from the elevator control system to open/close doors.
   3. Upload all current status of the elevator to the elevator control system.
   4. Stop operating and close the doors when receiving an emergency stop signal.

R4. Control System

1. **For the elevator control system, it should be able to:**
   1. Receive and process all information about the elevator's floor position and current operating status from the panel. When multiple users control the elevator concurrently, it should select the optimal algorithm to schedule the elevators, including:
   2. When there are multiple passenger requests, the elevator should first go to the floor where the nearest passenger is located and take them to their destination, minimizing the waiting time for passengers.
   3. When responding to multiple passenger requests, the elevator can use an intelligent route planning algorithm to choose the appropriate stopping sequence and route, transporting passengers to their destinations in the fastest time possible, thus minimizing the total journey time.
   4. Adjust the elevator's door opening and closing status, and the direction of travel.
   5. Adjust the speed of the elevator based on the position of the target floor and the distance from the current floor to the target floor (reflected in the time required for elevator travel).
   6. Immediately stop the elevator operation upon receiving an emergency signal from the panel and report any malfunction through the panel.