**Exercise Submission 1**

**Name: Shayan Ali Xain**

**Student Code: 90021567**

**Requirements and Planning:**

**Actors:**

1. Passenger (human)
2. System (Database) - (non-human)
3. Administrator (human)
4. Management Staff (human)
5. Developer (human)

**User Stories for the Actors:**

1. **Passenger:**

“As a passenger I want to add and update my details so that I can check the plane's arrival time, departure time, scheduled dates, maximum seating capacity and important communication details.”

“As a passenger I want to check if there are multiple planes leaving/arriving for the same destination so that I can choose my preferred one accordingly."

“As a passenger, I want good map location coordinates while in the flight that I can exactly know where the flight is at the moment.”

“As a passenger, I want to compare the prices of different flights to see which one fits my budget."

1. **System (Database).**
2. **Administrator:**

“As an administrator I want to enter data in the database, so that I can add new reservations and keep records of them.”

1. **Management Staff:**

"As the management staff I want to check the updated schedule for the flights available at all times so that I can keep track of all the activity of flights and functional level."

“As the management staff, we want a money collection program so that we can collect and check all transactions of the fee/money.”

1. **Developer:**

“As a developer, I want to design a flexible and suitable airline flight software so that the user can make use of it easily.”

“As a developer, I want to develop accurate and pinpoint map location coordinates while so that the passenger can exactly know where the plane is at the moment.”

"As a developer I want to develop the system in such a way that it should send reminder email(s) to each passenger travelling in next 12 hours, clearly providing, terminal and gate details, also provide instructions on special arrangements if any."

**Use Cases:**

**Use Case 1:**

Use Case: Passenger

Goal: Adding and Updating Passenger Details

Primary Actor: Passenger, Management Staff, and System

Secondary actor: Administration and Developer

Preconditions: Passenger must have a valid passenger ID

Trigger: Selecting the option "passenger details"

**Flow of Events:**

1. Ask the passenger to enter their details

2. Check passenger ID

3. Check phone number

4. Check email ID

5. Store this data in the system (database)

6. Let the data in successfully with a "Successful" pop-up on screen.

**Use Case 2:**

**Use Case: Passenger**

**Goal:** Check if there are multiple planes leaving/arriving for the same destination

**Primary Actor:** Passenger and the System

**Secondary actor:** Management Staff

**Preconditions:** The user must be a registered passenger in the system.

**Trigger:** Choosing the option "Flight Arrivals & Departures"

**Flow of Events:**

1. Ask the customer to enter their details

2. Check if the user is already registered

4. Verify through email

4. Verify phone number

5. Successfully show a page showing all the arrival and departure timings.

**Use Case 3:**

**Use Case:** **Management Staff**

**Goal:** To check the updated schedule for the flights

**Primary Actor**: Management Staff and System

**Secondary Actor:** Passenger and Developer

**Preconditions:** Management Staff must have authorized access.

**Trigger:** Clicking the “Updated Schedule” option and then the page for the updated schedule flights successfully shows up.

**Flow of Events:**

1. Ask the management staff to check the updated schedule for the flights.
2. Check if the user is an authorized management staff member
3. Verify through email
4. Successfully show a screen showing the updated schedule.

**Use Case 4:**

**Use Case: Passenger**

**Goal:** Compare the prices of different flights

**Primary Actor:** Passenger and the System

**Secondary actor:** Management Staff

**Preconditions:** The user must be a registered passenger in the system.

**Trigger:** Choosing the option "Compare Flights"

**Flow of Events:**

1. Ask the customer to enter their details

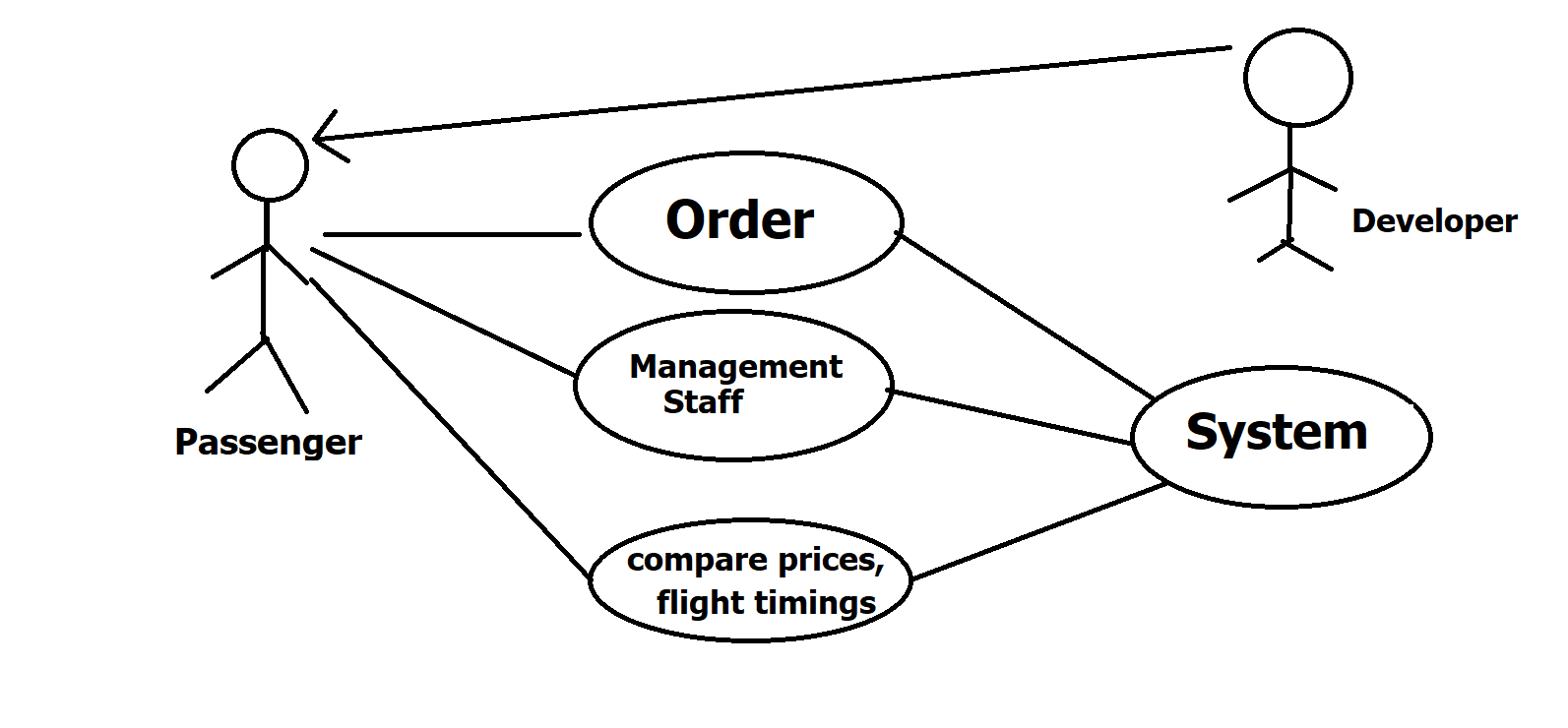
2. Check if the user is already registered

4. Verify through email

4. Verify phone number

5. Successfully shows a page, comparing the prices of different flights.

**UML Diagram:**



**Usability Requirements:**

a) The flight system must be flexible and simple to use.

Example: There must be an option on the system that quick books a flight based on the passenger’s preferences with a single click.

b) It must show all the relevant information on the very same page.

Example: In the flight system, the relevant information like plane's arrival time, departure time, scheduled dates, maximum seating capacity and important communication details should show on the same page.

c) It must make use of all the features effectively.

Example: the verification of the user and management staff's IDs must be quick and done in a single "verify now" click.

**Reliability Requirements:**

a) The flight system must be available on-demand and it should not fail less than 99.99%

b) The flight system should do false verification more than 0.1%

c) The rate of accuracy of the user data retrieved must not be any lower than 99.99%

**Performance Requirements:**

a) Response Time:

Example: the system should in 10 ms after a user enters it.

b) Data processing rate:

Example: 100 fields of data must be saved in a minute in the system.

c) Accuracy and effectiveness of the results:

Example: The data saved must be accurate when retrieved from the system.

**Work Breakdown:**

1. Hiring Staff

1.1 Developing Staff (7 weeks)

1.2 Management Staff (6 weeks)

1.3 Administration Staff (4 weeks)

2. Arranging Equipment for the entire flight System

2.1 Equipment requirement analysis for the System (8 weeks)

2.2 Ordering and gathering Equipment for the System (4 weeks)

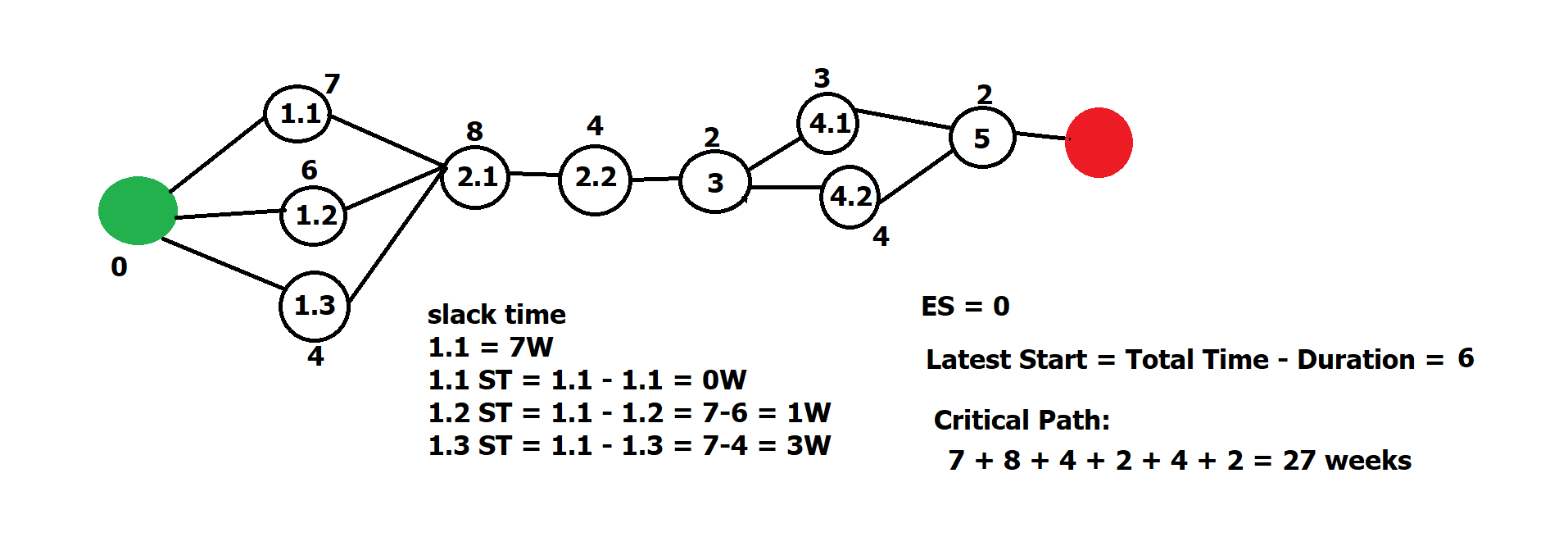
3. Documentations and Official work (2 weeks)

4. Preparing for the launch of the System

4.1 Setting up equipment (3 weeks)

4.2 Developing the System (4 weeks)

1. Using the System and testing it (2 weeks)



The development team should have branches like Hiring Staff, Arranging Equipment for the entire flight System, Documentations and Official work and Preparing for the launch of the System. The Total time for the project should be about 27 weeks, the estimated start time should be zero weeks and the slack time should be 7 weeks.