**ISEN1000 Introduction to Software Engineering Trimester 3, 2023**

**1.PLANNING :**

* **WORK BREAKDOWN STRUCTURE:**

1. Sign up to access the SKYFLY airline reservation system.
2. User management

2.1 User profile

2.2 User authentication and authorization

3.Functionality for Customer details

3.1 Input Passenger info (passport no, name, DOB, Address)

4. Flight information

4.1 Searching flight and availability

4.2. Select flight type (international/domestic)

4.3 View destination / arrival airport, Flight number

4.4 Departure/ Arrival Timing

4.5 View Flight fare

5. Seat allocation and management

5.1 Selecting seat class (Economy, Economy first, Business, FirstClass)

5.2 Seat Type (Regular, Emergency Exit)

6. Ticket information

6.1 Ticket cancellation

7. Payment and Billing

7.1 confirm flight booking

7.2 Payment processing (online/on-desk) for booking

7.3 Must have confirmation email / receipts and invoice system

7.4 Payment status

8 Updating Custom schedules

8.1 Maintaining all the records of flights and additional analysis 8.2 Notifications for flight reservations, flight status, update, etc.

9.Reservation enquiry

9.1 Reservation number, passenger list and their selected seats,

9.2 Reservation status (Checked in, request, pending, cancel, confirm)

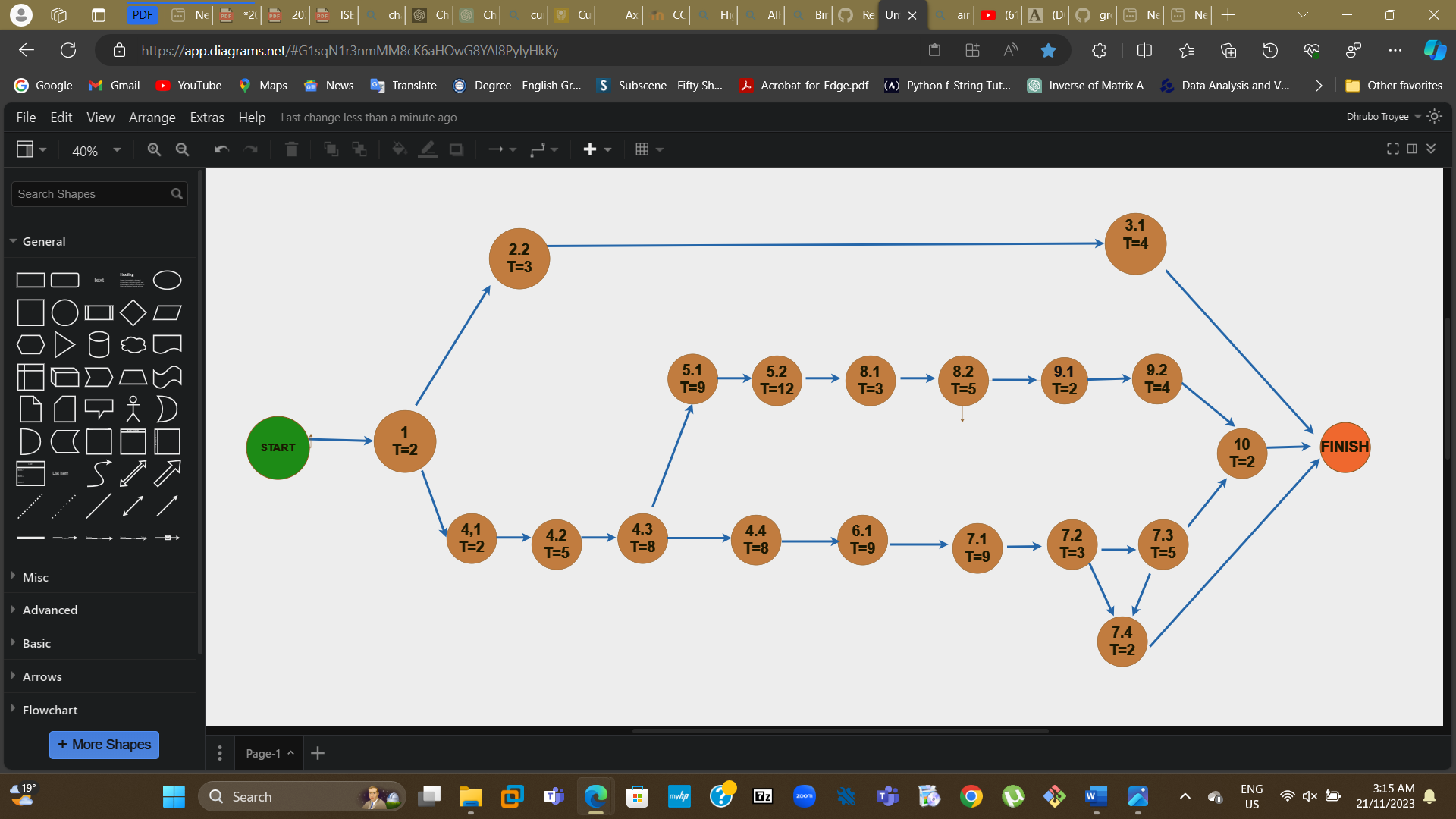
10.Technical support for any unexpected occurrence

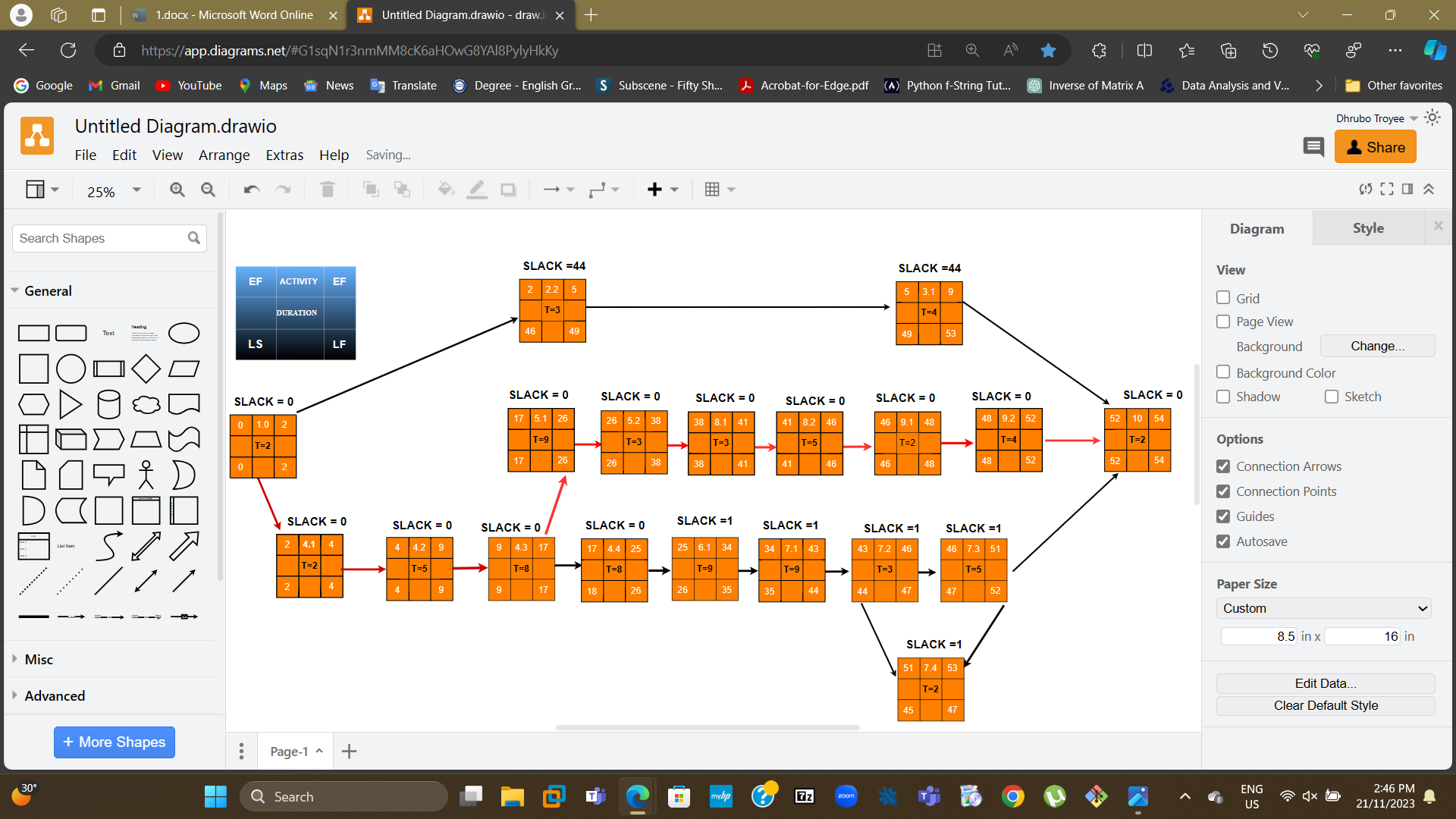
10.1 Customer support

* **.TASK TABLE:**

|  |  |  |  |
| --- | --- | --- | --- |
| **Sr. no** | **Activity** | **Time /Duration**  **WEEK** | **Dependencies** |
| **1.** | Sign up to access the SKYFLY airline reservation system. | 2 | **--** |
| **2.2** | User authentication and authorization | 3 | **1.** |
| **3.1** | Input Passenger info (passport no, name, DOB, Address) | 4 | **2.2** |
| **4.1** | Searching flight and availability | 2 | **1** |
| **4.2** | Select flight type (international/domestic) | 5 | **4.1** |
| **4.3** | Destination / arrival airport, Flight number | 8 | **4.2** |
| **4.4** | Departure/ Arrival Timing | 8 | **4.3** |
| **5.1** | Selecting seat class (Economy, Economy first, Business, FirstClass) | 9 | **4.3** |
| **5.2** | Seat Type (Regular, Emergency Exit) | 12 | **4.3 and 5.1** |
| **6.1** | Ticket cancellation | 9 | **4.2 , 4.3 , 4.4** |
| **7.1** | Confirmation of flight booking | 9 | **4.4 and 6.1** |
| **7.2** | Payment processing (online/on-desk) for booking | 3 | **7.1** |
| **7.3** | Must have confirmation email / receipts and invoice system | 5 | **7.1 and 7.2** |
| **7.4** | Payment status | 2 | **7.2 and 7.3** |
| **8.1** | Maintaining all the records of flights and additional analysis | 3 | **5.1,5.2** |
| **8.2** | Notifications for flight reservations, flight status, update, etc | 5 | **8.1** |
| **9.1** | Reservation number, passenger list and their selected seats | 2 | **8.1 and8.2** |
| **9.2** | Reservation status (Checked in, request, pending, cancel, confirm) | 4 | **9.1** |
| **10** | Customer support | 2 | **7.3 and 9.2** |

* **Activity on Node (AON graph):**





1 4.1 4.2 4.3 5.1 5.2 8.1 8.2 9.1 9.2 10 is our critical path

**According to the graph above, the project’s completion time is 54 WEEKS.**

* **CRITICAL PATH METHODS IN PROJECT MANAGEMENT:**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **ACTIVITY** | **TIME(t) WEEK** | **EARLY START**  **(ES)** | **EARLY FINISH(EF) EF = ES +t** | **LATE START(LS)LS=LF-t** | **LATE FINISH(LF)** | **SLACK TIME**  **=LS − ES = LF-EF** | **CRITICAL PATH** |
| **1.** | 2 | **0** | **2** | **0** | **2** | **0** | **yes** |
| **2.2** | 3 | **2** | **5** | **46** | **49** | **44** | **no** |
| **3.1** | 4 | **5** | **9** | **49** | **53** | **44** | **no** |
| **4.1** | 2 | **2** | **4** | **2** | **4** | **0** | **yes** |
| **4.2** | 5 | **4** | **9** | **4** | **9** | **0** | **yes** |
| **4.3** | 8 | **9** | **17** | **9** | **17** | **0** | **yes** |
| **4.4** | 8 | **17** | **25** | **18** | **26** | **0** | **yes** |
| **5.1** | 9 | **17** | **26** | **17** | **26** | **0** | **yes** |
| **5.2** | 12 | **26** | **38** | **26** | **38** | **0** | **yes** |
| **6.1** | 9 | **25** | **34** | **26** | **44** | **1** | **no** |
| **7.1** | 9 | **34** | **43** | **35** | **44** | **1** | **no** |
| **7.2** | 3 | **43** | **46** | **44** | **47** | **1** | **no** |
| **7.3** | 5 | **46** | **51** | **47** | **52** | **1** | **no** |
| **7.4** | 2 | **51** | **53** | **45** | **47** | **1** | **no** |
| **8.1** | 3 | **38** | **41** | **38** | **41** | **0** | **yes** |
| **8.2** | 5 | **41** | **46** | **41** | **46** | **0** | **yes** |
| **9.1** | 2 | **46** | **48** | **48** | **52** | **0** | **yes** |
| **9.2** | 4 | **48** | **52** | **48** | **52** | **0** | **yes** |
| **10** | 2 | **52** | **54** | **52** | **54** | **0** | **yes** |

**Here, activities with zero slack make up the critical path and are highlighted in yellow. In project management, the critical path consists of the tasks, having zero slacks, that must be completed to finish the project within the shortest possible time frame. Any delay in any of the tasks on the critical path will cause a delay in the overall project completion. Besides, tasks that are not on the critical path might start and conclude at any time, therefore delays in these tasks may not necessarily affect the project's final finishing schedule.**

* **Calculation of expected time: Program Evaluation and Review Technique (PERT):**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **ACTIVITY** | **Estimated Duration (WEEK)** | | | **Optimistic + (4 ×Most Likely) +Pessimistic**  **Expected = \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**  **Time 6** |
|  | **Optimistic**  **Time Estimate** | **Most likely Time Estimates** | **Pessimistic**  **Time Estimate** |  |
| **1.** | **2** | **4** | **6** | **4** |
| **2.2** | **5** | **1** | **5** | **2..3333** |
| **3.1** | **3** | **7** | **9** | **6.66667** |
| **4.1** | **2** | **7** | **4** | **5.6666667** |
| **4.2** | **8** | **4** | **1** | **4.1666667** |
| **4.3** | **3** | **5** | **7** | **5** |
| **4.4** | **2** | **5** | **10** | **5.333333333** |
| **5.1** | **4** | **3** | **12** | **4.666667** |
| **5.2** | **5** | **7** | **8** | **6.833333** |
| **6.1** | **6** | **4** | **7** | **4.833333** |
| **7.1** | **7** | **8** | **10** | **8.166667** |
| **7.2** | **9** | **10** | **11** | **10** |
| **7.3** | **6** | **12** | **5** | **9.833333** |
| **7.4** | **6** | **4** | **5** | **4.5** |
| **8.1** | **2** | **7** | **3** | **5.5** |
| **8.2** | **5** | **8** | **10** | **7.83333333** |
| **9.1** | **8** | **9** | **12** | **9.3333** |
| **9.2** | **6** | **2** | **6** | **3.33333** |
| **10** | **9** | **7** | **7** | **7.333333** |

**Expected project duration**

**=4+ 5.66667 +4.16666667 +5 +4.66667+6.8333333 +5.5 + 7.8333333 +9.333333 + 3.33333333**

**=52.13 Weeks (about 1 year)**

**The expected project duration is 52.13 Weeks (about 1 year).**

**The project completion time is 54 weeks.**

**2. Functional Requirement Analysis**

**Human Actor:**

1. **Passengers: T**he primary actor who interacts with the system by checking flight schedules, reservation, cancelling tickets, making payment receiving reminders or update notifications.
2. **Pilots/ Crew:** They can view their flights that they assigned and schedules.
3. **Front desk officers**: They interact with passengers to help with reservation, cancelling or booking tickets.
4. **Customer:** They can view flights schedules, cancel the tickets and book reservation according to their needs.
5. **Admin / Management Staff:** Basically, they are responsible for adding new flights and their schedules, managing flight updates, cancelling flights, maintaining staff etc.

**NON-HUMAN ACTOR:**

1. **Database:** Restore all the details of flight schedule, flights status update, sending notification regarding any changes.
2. **Payment gateway Service:** This one is responsible for handlining online payment, billing, processing transactions, generating receipts for passengers.

* **User Stories:**

1. **As a customer service representative, I want to make the refund process smoother**, s**o that passengers do not have to wait for a long time.**
2. **As a pilot I want to access detailed flight pants and routes before departure so that I can ensure a safe journey for all passengers.**
3. **As an Airline admin, I want to manage user access levels, so that passenger details keep safe.**
4. **As a passenger, I want to see real-time flight status updates so that I can plan my travel accordingly.**
5. **As a passenger, I want to receive reminders of travel-related information, so that I don’t miss any flights and their updates.**
6. **As a passenger, I want to view all available flight details, so that I can plan my future travels without committing a reservation**.
7. **As a passenger, I want to download the boarding passes so that I access the terminal and get onto the flight.**
8. **As an airline agent, I want to distribute seats so that I can manage seating arrangements for passengers.**
9. **As a Payment service gateway, I would like to have a system so that I can be able to deal with online/on-desk payments for booking, providing confirmation emails/receipts and tax invoices.**
10. **As a Database system, I would like to ensure an updated 24/7 schedule for passengers, pilots, and management staff so that They can access recent information about flight details**
11. **As a front desk representative, I want to access passenger's details, so that I can provide correct assistance and support.**
12. **As an Admin, I want to manage seat capacity so that the flight does not allow overbooking.**
13. **As a passenger, I want to receive a reminder email with terminal and gate details 12 hours before my flights so that I can prepare myself.**
14. **As a staff member, I want to update gate and terminal information so that passengers can easily find their way to the airport.**
15. **As a database system I would like to have the capability to assign/remove new flights to/from a destination if required** **so that customers/ passengers find it easy to use according to their needs**
16. **As a database system, I want to add new flights, destinations, and routes into the current reservation system so that passengers, staff, and customers can find the flight operator details according to their destination.**

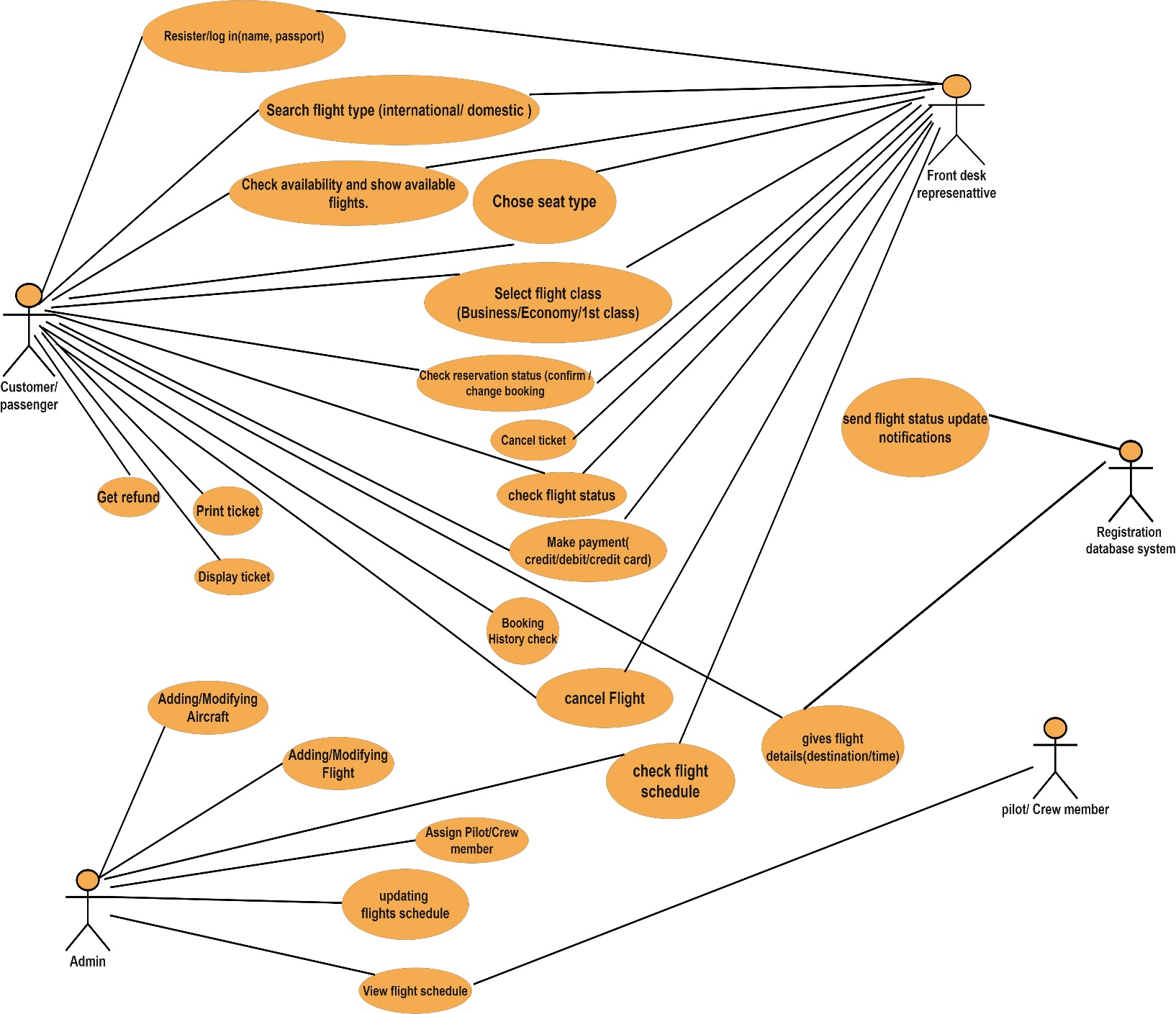
* **Use case diagrams**

|  |  |
| --- | --- |
| **Use Case 1:** | **Managing seating arrangements** |
| **Goal**: | Arranging seats for passengers according to their selections and availability |
| **Primary**: | Airline agent |
| **Secondary Actor(s)**: | Passenger |
| **Pre-conditions**: | * Passengers must have to choose the flights category (International/domestic) * Seat rearrangement is a must * Flights must be booked once. |
| **Trigger**: | For a fully booked airline, an agent must reserve seats. |
| **Flow of Events:** |  |
|  | **1.** At first, **t**he airline agent needs to access the flight (International/Domestic) details using the website/system |
|  | **2**. From the website/ system the agent can verify the seats based on class (business/ economy) |
|  | **3**. System displays available seats for (Economy, Economy first, Business, and First-class) and show seat type (Regular/ Emergency exit) |
|  | **4. from** the display agent selects the seats and starts arranging the seats. |
|  | **5**. The agent mainly rearranges the seats based on the passenger's request or recommendations |
|  | **6.** Getting confirmation from the agent system to update seat reservations successfully |
|  |  |
| **Extensions** |  |
|  | **3**. IF suitable seats are not available |
|  | **3.1** Agent asks to help from the administrator for help or otherwise offers additional seats |
|  | **3.2** will resume at step 2 |
|  | **4.** If the customer wants to modify their seats after it has been placed, |
|  | **4.1.** The agent will look after that; they will give them options based on the current available seat plan |
|  | **4.2** Continue from step3 |
|  | **5.** if a passenger wants a particular seat |
|  | **5.1** The agent will see the passenger's requirement and allow her/ him a suitable seat according to their need |
|  | **5.2** Customers continue from Step 4 |
|  | **6**. During order placement, if the customer runs into technical issues |
|  | **6.1 The** system provides support options or assistance, or the agent will report for that for further investigating |
|  | **6.2** Continue from step3 |

|  |  |
| --- | --- |
| **Use Case 2:** | **Process refunds for cancellations** |
| **Goal**: | Process the refunds and update the passenger about the refund status |
| **Primary**: | Customer Service Representative |
| **Secondary Actor(s)**: | Passenger, Database |
| **Pre-conditions**: | * Passenger must cancel their flights. * They must have a receipt of cancellation by email or invoice |
| **Trigger**: | Customer service representatives need to access the Airline’s System according to passenger details. |
| **Flow of Events:** |  |
|  | **1.** After accessing the passenger booking details, they will want to know the reason for the cancellation |
|  | **2.** They will select the option of cancelling the flight from the system |
|  | **3.** The system will check the refund/cancellation policy of the airline system |
|  | **4.** Then the system will again verify if the passenger is eligible or not for the refunds and check refund methods such as credit cards etc. |
|  | **5** The representative will initially check again all the ongoing process |
|  | **6.** Now, the system will generate a refund confirmation after refunds are successful, including details such as refund amount, methods of funds, and time for the refund processed. |
|  | **7**. The system will update the passenger booking status and marked it “Canceled” or “Refunded “for tracking. |
|  | **8.** The representative also informs about the confirmation process to the passenger via (email, phone call, SMS or face-to-face) and refund details more briefly |
| **Extensions** |  |
|  | **1**. IF the customer representative finds the details of the passenger is not right |
|  | **1.1** They will ask the passenger to provide the right details to process further inquiries. |
|  | **1.2** will resume at step 1 |
|  | **4.** If a passenger is not eligible for the refund |
|  | **4.1** The representative will briefly go through the policy, explain it to the passenger and provide alternative Solutions for it |
|  | **4.2** Continue from step 3 |
|  | **7.** At the time of updating confirmation, if it runs into a technical issue, then it will not be done |
|  | **7.1** The system will notify the representative and provide them with alternate options for confirmation |
|  | **7.2** will continue from step 6 |

|  |  |
| --- | --- |
| **Use Case 3:** | **Manage and Access passenger details** |
| **Goal**: | Access passenger details to provide accurate support for the inquire |
| **Primary**: | Front desk representative |
| **Secondary Actor(s)**: | Passengers, database system |
| **Pre-conditions**: | * Front dusk representative must log into the Airline system with internet * The system must have details of all passenger information regarding the inquiry. * Passenger must show their credentials as verification |
| **Trigger**: | In the system interface area, the Front desk offer will access to named “Customer Information” / “passengers Details “ |
| **Flow of Events:** |  |
|  | **1.** After the login procedure is done, the Front Desk representative will search for some keywords for passenger details. Such as passenger name, DOB, flight details of travel, reservation number |
|  | **2**. The Front Desk representative will input any of those keywords for the requested information |
|  | **3**. According to the given data the system starts searching and tries to pick up the passenger details from the database |
|  | **4.** After that, the system will display passengers' details in a format such as passenger Name, DOB, Passport, booking history, current flight details and so on |
|  | **5**. The Front desk representative will look at the details very carefully from the system and check again if the data is correct or not. |
|  | **6.** Then the Front desk representative after evaluating provides the precise and specific guidance based on the data that was collected from the passenger |
|  | **7**. After providing the necessary support, the Front Desk representative confirms the contract is over and makes sure the passengers' questions have been answered properly. |
| **Extensions** |  |
|  | **1**. IF passenger details are not found |
|  | **1.1.** The front desk representative will reassess the entered criteria based on the information given by a passenger or may ask for more data |
|  | **1.2** They can also provide alternative solutions such as verifying the speaking of the passenger’s name and DOB of the passenger. |
|  | **3.** If additional information is required |
|  | **3.1.** The Front desk representative communicates with the passenger to gather more specific details based on the criteria |
|  | **3.2** Continue from step 2. |
|  | **4**. If the system access permission is limited/denied displaying details of the passenger |
|  | **4.1** The front desk representative asks for help from a senior or administrator to access specific restricted information. |
|  | **4.2** then will continue from step 3 |
|  | **5**. If the front desk representative runs into technical issues, during the search. |
|  | **5.1** The front desk will immediately inform the It support team to solve this tech issue |
|  | **5.2**The front desk representative will try alternative methods to access another system if available to gather passenger details |
|  | **5.3** then will continue from step 4 |
|  | |
| **Use Case 4:** | **Access and download boarding pass** |
| **Goal**: | Passengers want to access their boarding passes and download them. |
| **Primary**: | Passenger |
| **Secondary Actor(s)**: | Database, Customer Reservation system |
| **Pre-conditions**: | * Passengers have booked their flight * They must have to do it before the arrival time. |
| **Trigger**: | Initially, customers/Passengers logged into the AIRLINE system using their details. |
| **Flow of Events:** |  |
|  | **1.** After accessing the online airline system customers browse the desired items |
|  | **2**. The website searches for the selected list and shows the list of upcoming flights for the passengers. |
|  | **3**. The passengers select their chosen flights and add credential details for them. |
|  | **4.** Then, after adding their details, they can access their boarding pass. |
|  | **5**. The system also generates the digital boarding pass in a downloadable format such as a PDF |
|  | **6.** Getting the digital format, passengers download their pass to their device or print the hard copy. |
|  | **7**. The system then updates the boarding pass status of the passenger. |
| **Extensions** |  |
|  | **3**. IF passengers fail to log in using their credentials |
|  | **3.1** The system notifies the customers and offers similar substitutes |
|  | **3.2** will resume at step 2 |
|  | **4.** If the passengers are not able to access the boarding pass due to technical issues: |
|  | **4.1.** The web page will show options for updating, including the ability to access it again, modify some details |
|  | **4.2** Passengers must immediately contact customer service and try to collect the boarding pass from the airport service counter if possible. |
|  | **5.** If a passenger wants to modify their boarding pass information |
|  | **5.1 The** system will ask the passenger again to modify it before converting it into a downloadable format. |
|  | **5.2. If** the above one does not work, then the system provides support options or assistance. |
|  | **5.3** Customers continue from step 4 |

* **UML Use case diagrams.**



* **3. Non-Functional Requirement Analysis**

**– Usability requirements**

* **User-Friendly Interface:** The system must have a user-friendly interface to ensure ease of use.

So that they can search for flight details, and complete their booking without any assistance or guides

* The system must allow the user to query their purchase history with at most 2 mouse clicks.
* **Finding efficient Data**: The system must allow customers to access the Airline system to find relevant information (seating availability, flight schedule, arrival time, departure time) within 30 seconds, on average.
* Thus, the passenger and front desk representative will be able to collect relevant information easily
* **Accessibility:** The software must facilitate customers choosing the correct payment method at least 99% of the time.
* The system must have features for disabled people like screen reader support, alternative text for graphic data, and multi translation Language system for everyone.
* The system must provide quick response times to user actions to avoid frustration.
* The system must show all flight schedules of different routes and options on one screen.

**– Performance requirements**

* The system must be able to handle multiple payment gateway options and should not be exceeding 4 seconds.
* **T**he system must perform identical recognition on up to 100 faces per second.
* The system must respond to user activity (searching flights, reservation booking, flight schedule) within 30 seconds and also, page loading time should not cross 4 seconds during the action
* Online payment processes and confirming the payment receipt transaction must be done in less than 20 seconds.
* The music player must use no more than 500 MB of RAM in total, and no more than 0.5 MB for each track in the playlist.
* The system must be able to handle a high volume of users and transactions simultaneously.
* The error rate of passengers submitting identical details on the user details page must not exceed 10 %

**– Reliability requirements**

* The system’s user registration component should be available 99.99% of the time.
* The system’s ticket print feature should fail on no more than 0.1% of attempts.
* The system should have an error-handling mechanism In order to avoid unexpected errors or system failure.
* The Airline reservation system must not make more than 1 incorrect payment every 1000 hours.
* The system must perform without failure in 94% of use cases during a month
* The system must have regular backups to prevent airline reservation data details in case of a system failure.
* **4. Version Contro**l

***VERSION CONTROL:***

***Branching Strategy: There will be 3 branches:***

*1****. Main Branch:*** *The primary branch that contains the stable version of the working, where other branches will be merged and quality assurance checks. It should only be updated with thoroughly tested and reviewed code.*

|  |
| --- |
| mkdir ToDoProject  cd ToDoProject  git init |

*2.* ***Development Branch****: The branch that contains the latest version of the work. Developers will create this branch to work on user stories, add features or fix issues. To maintain the branch with the most recent stable code, we need to update this branch weekly depending on the final work states,*

|  |
| --- |
| git checkout –b development  git checkout –b bugfix |

*3.* ***Feature Branches****: These branches are used to develop new features, major changes or update existing ones in the WBS. For user stories given, each feature branch will be created from the development branch. They will work on the feature branch, change and execute them. All new features and updates are merged into it. Then it should be tested here and merged back into the development branch*

|  |
| --- |
| git checkout –b features |

***When should they be created and when should they be merged and pushed.***

At the beginning of the project, the **main branch** and the **development branch** will be created. (Day1)

After this stage, **the Development Branch**  will be tested and the change should be merged into the **main branch**, which will represent a new version of the work that has been done. (Ongoing development weekly)

Then, a feature branch will be created for user stories and will be used. Then it *should be merged into the* ***Development branch*** *and pushed* (Weekly)

When the developer or programmer finishes a use story and tests it on the feature branch it will be merged into the development branch and pushed. (Every 2-3week)

Overall, after all the user stories have been completed and tested the development branch will be merged with the main branch and pushed.

**Pushing TO a remote repo:**

► Upload “mynewbranch” to “origin” (and make “origin” the current remote):

|  |
| --- |
| git push -u origin mynewbranch |

► Upload the current branch to the current remote:

|  |
| --- |
| git push |