A close up of a logo

Description automatically generated

****

**Requirement Review Document**

Flight Management System

# Date: 29/12/2023

# Reviewer: Eslam Fawzy

# Version: 1.1

**Purpose**

To evaluate the clarity, completeness, consistency, testability, and feasibility of the requirements specified in SRS of our **Flight Management System** In order to ensure that the system matches the customer expectations and meet the passenger expectations and experience.

## Completeness

### Functional Requirements

The document provides a food overview of the system’s purpose, scope, and user classes. However, it lacks detailed functional requirements like the followings:

* specific steps and actions to perform a business action like reserving a flight.
* inputs and outputs for each function
* limitations and constraints for each function
* Expected system behavior for to each functional requirements for both customers and employees.
* Can the user or admin edit the flight details ?

### Integration Requirements

The document didn’t mention any of the integration requirements with external database server that supporting hundreds of major cities around the world as well as thousands of flights by various airline companies or the airport database. The document also didn’t provide any information about the available payment gateways, Visa, Cash, Master Card, Bitcoin … etc. It also didn’t mention about the APIs that will communicate with the airports and the different reservation systems.

### Error Handling and Exception Scenarios

The document does not address how the system should handle error conditions or exceptions, such as:

* Invalid Inputs
* System failures or conflicts between two reservations
* Validations and the appropriate error messages in case of errors

### Data Validation and Constraints

There is no info about the validations and the constraints rules in the system. It is important to specify the acceptable formats, ranges, and any business rules that should be enforced to ensure data integrity and consistency.

### Non-Functional Requirements

The document didn’t mention any non-functional requirements such as:

* Performance of the system and the expected response time
* Security of the system and how to access this system and the privacy of data.
* Reliability and data integrity within the system.
* Usability aspects and the user interface design guidelines.

### System Performance and Scalability

The document didn’t provide any notes about the system performance, or the future scalability and I suggest including the followings:

* Expected Response Time.
* Performance benchmarks.
* Future extension and capacity
* Capacity planning
* Database volume
* Concurrent user support
* The expected time where the system could be facing some crowds like Christmas.

All these aspects’ considerations will help guide the development and infrastructure planning.

### Accessibility and Localization

The document does not address requirements related to accessibility for users with disabilities or localization needs for supporting multiple languages.

### Reporting and Analytics

The project objective is to help the passenger to book the most proper ticket for him based on the available airline tickets, the price, and the class. To do this you have to make a new utility in the system to make some analysis and reporting based on this info. There is no mention of this utility. It must be mentioned and also has to include the followings:

* Sales Reports (e.g. on each flight)
* Occupancy Rates for each plan/fligh/airport
* Customer Feedback
* Price change for each flight over the season per person

## Consistency

There are no evident conflicts or contradictions within the document. The requirements appear to be internally consistent and aligned with the purpose and scope of the system. However, after completion of all missing and mentioned requirements, we must revisit the consistency again to verify the provided info is consistent.

## Clarity

All requirements are clear but there is one expression that needs to be clarified “flight leg instances”. I didn’t understand what is meant by this expression and I need a business clarification about it.

## Testability

### Flight Details

* **Testability**: Testable
* **Testing Types**: Database testing, integration testing, data validation testing

### Customer Description

* **Testability**: Testable
* **Testing Types**: Database testing, data validation testing.

### Reservation Description

* **Testability**: Testable
* **Testing Types**: Database testing, data validation testing, reservation workflow and test steps testing

### Customer Functions

* **Make a new reservation.**
  + **Testability**: Testable
  + **Testing Type**: Functional testing, integration testing.
* **Cancel an existing reservation.**
  + **Testability**: Testable
  + **Testing Type:** Functional testing, integration testing
* **View their itinerary.**
  + **Testability**: Testable
  + **Testing Type**: Functional testing, integration testing.

### User Class and Characteristics

* **Testability**: Testable
* **Testing Types:** Functional testing, integration testing.

### Employee Functions

* **Customer Function.**
  + **Testability**: Testable
  + **Testing Type**: Functional testing, integration testing.
* **Administrative.**
  + **Testability**: Testable
  + **Testing Type:** Functional testing, integration testing

### Limited Seats

* **Testability**: Testable
* **Testing Types**: Load testing, stress testing, database testing

### Non-Testable Requirements:

The things that are lacking in the document and weren’t mentioned are not testable until it is mentioned. The document briefly mentions supporting hundreds of major cities and thousands of flights. However, specific performance and scalability requirements are not provided.

## Feasibility

The requirements appear to be feasible within the stated project scope. However, the document does not provide information on the technology stack, infrastructure, or constraints that could impact the feasibility of implementing the system. There are some suggestions from my side to increase feasibility to be more feasible.

### Purpose

Seems feasible for me and I have no suggestions.

### Project Scope

Define the scalability requirements for the system to handle increasing numbers of cities, flights, and users. Consider conducting performance testing to ensure the system can handle the expected load.

### Product Perspective

Define the database structure and relationships between tables and data clearly. Consider implementing appropriate data validation and integrity checks to ensure the accuracy and reliability of the stored information.

### User Class and Characteristics

Define the search criteria and algorithms for retrieving flight information efficiently. Consider implementing caching mechanisms to improve performance.

### Customer Functions

Implement a user-friendly interface and provide clear instructions for each function.

### Employee Functions

Implement access control mechanisms to ensure that employees can only perform authorized actions.

### Limited Seats

Define seat allocation algorithms and implement checks to prevent overbooking. Consider implementing real-time seat availability updates to provide accurate information to customers and employees.