### Functional Requirements Specification Document for Apache Airlines Seat Booking Software

#### 1. Introduction

1.1 Purpose

The purpose of this document is to specify the functional and non-functional requirements for the development of a seat booking software for Apache Airlines. This software is intended to support the expansion of Apache Airlines' operations by enabling efficient booking of seats on their newly acquired Burak757 passenger jets.

#### 2. Functional Requirements

2.1 Requirement 1: Seat Availability Check

**Description:** The system must allow customers to check the availability of seats on a specified flight. The seat availability must be displayed in a format that distinguishes between booked ("R"), free ("F"), isles ("X"), and storage areas ("S").

2.2 Requirement 2: Book a Seat

**Description:** The system must enable customers to book a seat if it is available. The booking process should update the seat status to "R" and ensure no bookings are made on "X" (isles) or "S" (storage areas). A unique booking reference must be generated and associated with the booking.

2.3 Requirement 3: Free a Seat

**Description:** The system must allow customers or administrators to free a previously booked seat. This action should update the seat status back to "F" and remove any associated booking reference and customer details from the system.

2.4 Requirement 4: Show Booking State

**Description:** The system must provide functionality to display the current booking state of the entire aircraft, showing all seats and their statuses.

2.5 Requirement 5: Customer Details Storage

**Description:** When a booking is made, the system must store the customer's details (passport number, first name, last name, seat row, and seat column) in a database. The system must ensure data integrity and confidentiality.

#### 3. Non-functional Requirements

3.1 Performance

**Description:** The system must handle requests efficiently, with response times for checking seat availability, booking, and freeing seats not exceeding 2 seconds under normal operating conditions.

3.2 Usability

**Description:** The user interface must be intuitive and accessible, allowing users to easily navigate through the booking process. The system should support common browsers and mobile devices.

3.3 Security

**Description:** The system must implement data encryption for sensitive information, such as customer details and booking references. Access control mechanisms should be in place to restrict administrative functionalities to authorized personnel only.

#### 4. Assumptions and Constraints

4.1 Assumptions

* The system will be used primarily by Apache Airlines customers and administrators.
* The system will be integrated with existing Apache Airlines IT infrastructure.
* The system will be developed using Python and hosted on Apache Airlines' servers.

4.2 Constraints

* The system must comply with relevant data protection regulations.
* The system must be scalable to handle peak booking periods, such as holidays.

GitHub:[Zewen666/final-project (github.com)](https://github.com/Zewen666/final-project)