* + - Module Code : FC723
    - Class/Group: Group C
    - Module Title : Programming Theory
    - Assessment Title : Portfolio Project 5
    - Tutor Name: Sophine Norman
    - Student GUID Number: 3075353
    - Date of Submission: 06/04/2025

“I confirm that this assignment is my own work.

Where I/we have referred to academic sources, I have provided in-text citations and included the sources in the final reference list. “

**Functional Requirements Specification**

**git repository link:** <https://github.com/HAORAN599/Apache-system.git>

**Introduction**

The following is a detailed description of the seating reservation system capabilities developed in python, using tables to more clearly demonstrate user requirements, traceability, and scoring

**Functional Requirement**

|  |  |
| --- | --- |
| Requirement ID | R-01 |
| Name | Check Seat Availability |
| User Requirement | User want to check if a seat is available before booking. |
| Description | The system allows the user to enter a seat number and tell if the seat is available. |
| Expected System Behavior | Returns 'Available' or 'Booked' |
| Example Scenario | Input: '1A' → Output: 'Available' |
| Priority | High |

|  |  |
| --- | --- |
| Requirement ID | R-02 |
| Name | Book a Seat |
| User Requirement | When a user books a seat, display a code to confirm the booking success. |
| Description | System collects seat number, name, passport and generates booking code. |
| Expected System Behavior | Marks seat as booked, stores booking in memory and DB. |
| Example Scenario | User enters '2B' + details → Output: 'Booking Successful - Code: NXD2DYKA' |
| Priority | High |

|  |  |
| --- | --- |
| Requirement ID | R-03 |
| Name | Cancel Booking |
| User Requirement | User can cancel his booking and release the seat. |
| Description | System cancels booking via seat number or code and updates status. |
| Expected System Behavior | Removes booking info and frees seat. |
| Example Scenario | Input: Booking Code 'X8G9PQ7Y' → Output: 'Cancelled' |
| Priority | High |

|  |  |
| --- | --- |
| Requirement ID | R-04 |
| Name | Display All Seats |
| User Requirement | User can see seat availability to choose easily. |
| Description | System displays a top seat view of seats with status. |
| Expected System Behavior | According to GUI interface to show seat layout with status (A - Available, B - Booked) |
| Example Scenario | Output: 1A - Booked, 1B - Available, ... |
| Priority | High |

|  |  |
| --- | --- |
| Requirement ID | R-05 |
| Name | Filter by Seat Type |
| User Requirement | As a user, he can view available window or aisle seats. |
| Description | System filters available seats by seat type input by user. |
| Expected System Behavior | Lists seats based on selected type. |
| Example Scenario | Input: 'window' → Output: ['1A', '2A', '3D'] |
| Priority | Medium |

|  |  |
| --- | --- |
| Requirement ID | R-06 |
| Name | Automatic conversion of uppercase and lowercase Spaces |
| User Requirement | As a user, he can Any uppercase and lowercase characters can be entered and recognized normally. |
| Description | Whether the user enters uppercase or lowercase, whether there is a space, it can be recognized normally |
| Expected System Behavior | Automatic recognition of conversion case |
| Example Scenario | Input: '1a' → Output: [available] |
| Priority | Medium |