**Module Code (FC723)**

**Class/Group: (Group C)**

**Module Title (Programming Theory)**

**Assessment Title (Project 1)**

**Tutor Name: (****Sophie Norman)**

**Student GUID Number: (3075614)**

**Date of Submission: (6th April 2025 at 23:59)**

**I confirm that this assignment is my own work.**

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

**Part A, Q1**

For the development of the seat-booking system for Apache Airlines, I have chosen the Waterfall Software Development Process. The Waterfall model is a linear and sequential approach to software development, where each phase must be completed before moving on to the next. It consists of the following main stages:

1- Requirements Gathering and Analysis.

2- System Design.

3- Implementation (Coding).

4- Testing.

5- Deployment.

6- Maintenance.

Reasons:

**Clear and Well-Defined Requirements**

The Apache Airlines project has already provided a detailed system description, including a seat layout diagram and a list of required functionalities. Since the requirements are unlikely to change during development, the Waterfall model is suitable because it works best when everything is known from the start.

**Strong Focus on Documentation**

Waterfall places a strong emphasis on documentation at every phase, which aligns with the academic requirement of this project. Each stage (e.g., functional specifications, activity diagrams, UML, and version control steps) can be clearly documented and presented.

**Simple and Easy to Manage**

Waterfall is straightforward to plan and track. It suits small- to medium-sized projects like this one, where all team members (or a single student developer) can follow a step-by-step approach without the need for iterative feedback loops or changing scope.

**Part A, Q3**

**Part A, Q5**

The group\_booking function allows a user to book 2 to 4 adjacent seats in the same row. It scans each row while skipping over the aisle column, looking for a continuous block of free seats ("F"). If such a block is found, the seats are reserved ("R"), and their seat codes are displayed. If no suitable seats are available, the user is notified.

**Part A, Q6**

The first step I took to create a Git repository and commit my code was to set up a public repository on GitHub. I then opened Git Bash and configured my Git identity by entering my name and email address using the git config command. After that, I navigated to my local project directory and used the git clone command to link it to the remote repository. To initialize version control locally, I ran git init, which created an empty Git repository in my project folder. I then added my project files to the staging area using git add . and committed the changes with a message using git commit -m "Initial commit". Next, I set the remote repository using the git remote add origin <URL> command, and finally, I used git push --force to push my committed changes to the main branch of the remote GitHub repository.

Link: <https://github.com/Hydro-XD10/Project-1-FC723.git>

Branch: “master”