

**The Superior University**

**Operating Systems Lab – Project Documentation Template**

1. **Project Title**

**Flight Scheduling for Airplanes By using Shortest Job First**

1. **Group Members**

**Team members:**

* Abdullah (SU92-BSSEM-F23-087)
* Abdur Rehman (SU92-BSSEM-F23-057)
* Arslan Jutt (SU92-BSSEM-F23-225)
* Ahsanullah (SU92-BSSEM-F23-217)

1. **GitHub Repository**

✅ Both the Python code file and this documentation must be uploaded to a public GitHub repository.

GitHub Repository Link:

<https://github.com/Abdur-rehman1to2/Flight-Scheduling>

Your repository should include:

✔️ Python .py file containing the scheduling algorithm

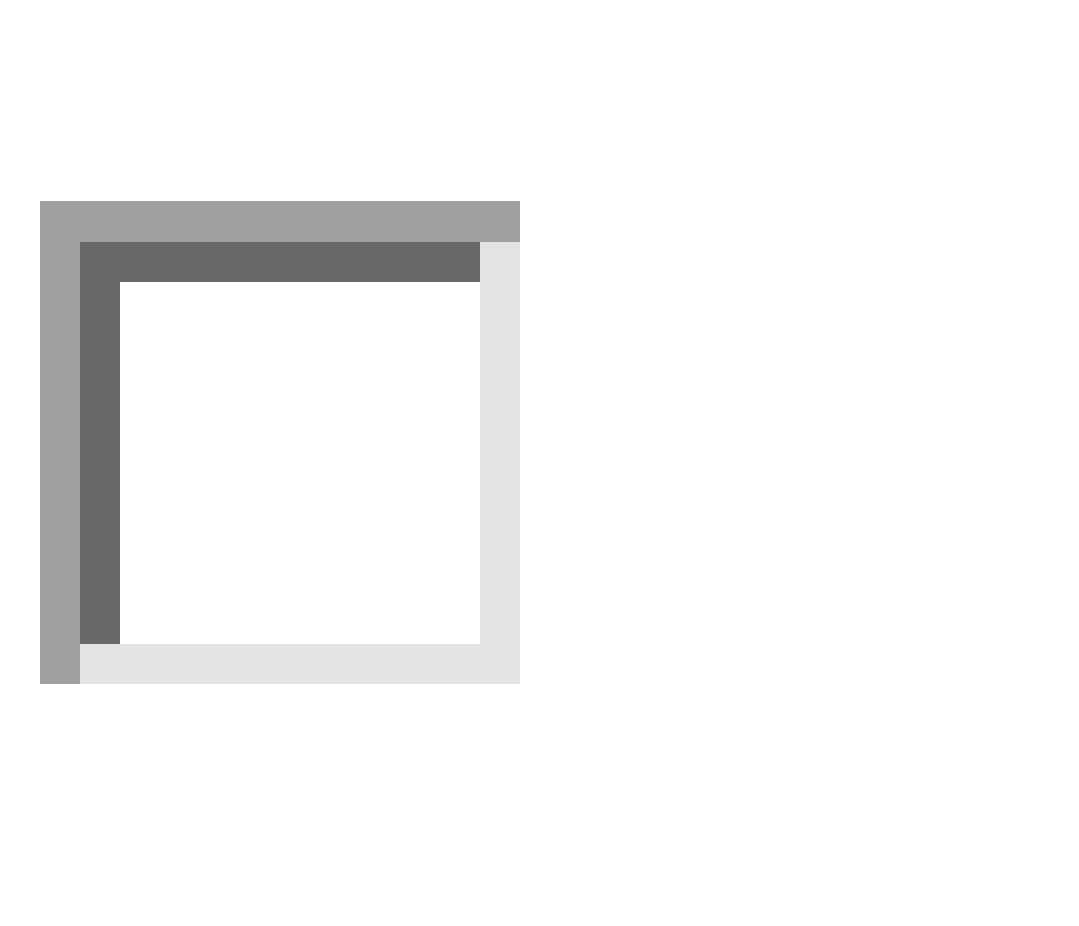
✔️ README.md with basic project info and instructions

✔️ This completed documentation file (.docx or .pdf)

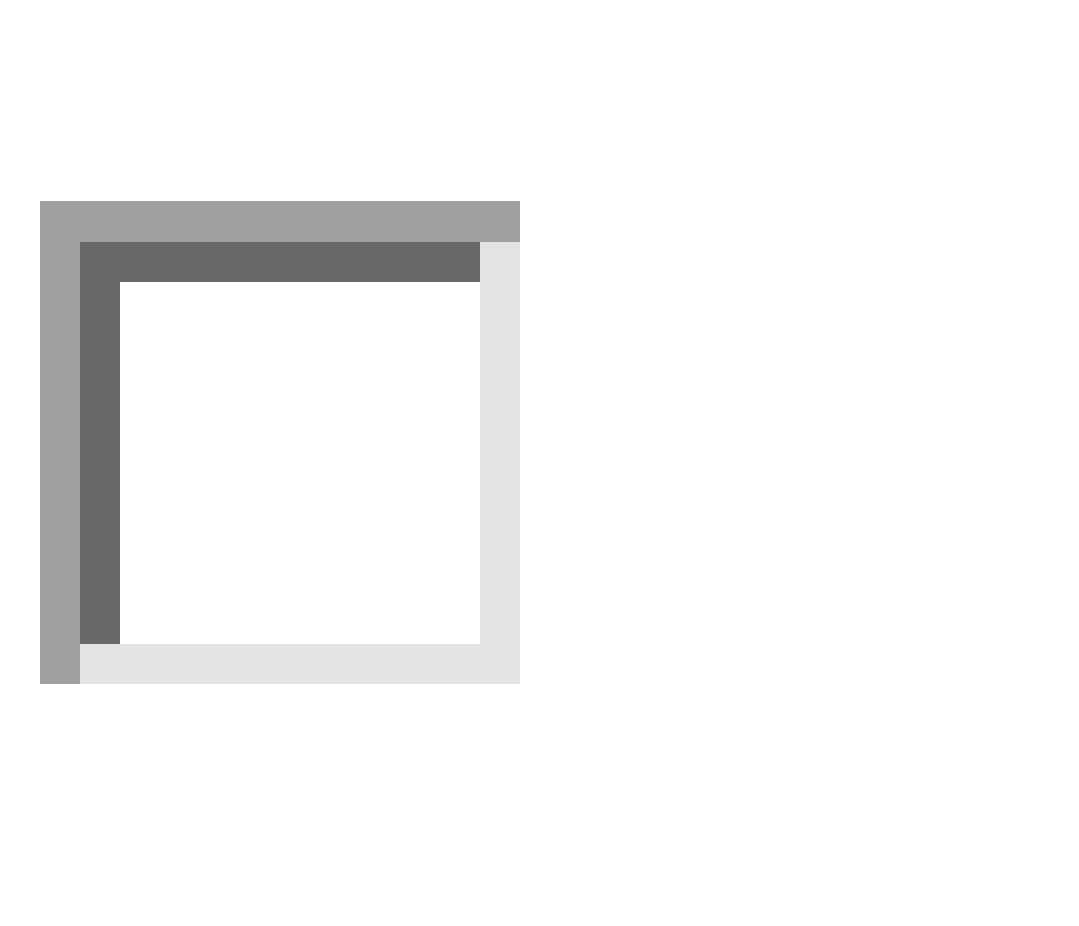
✔️ At least 2–3 output screenshots

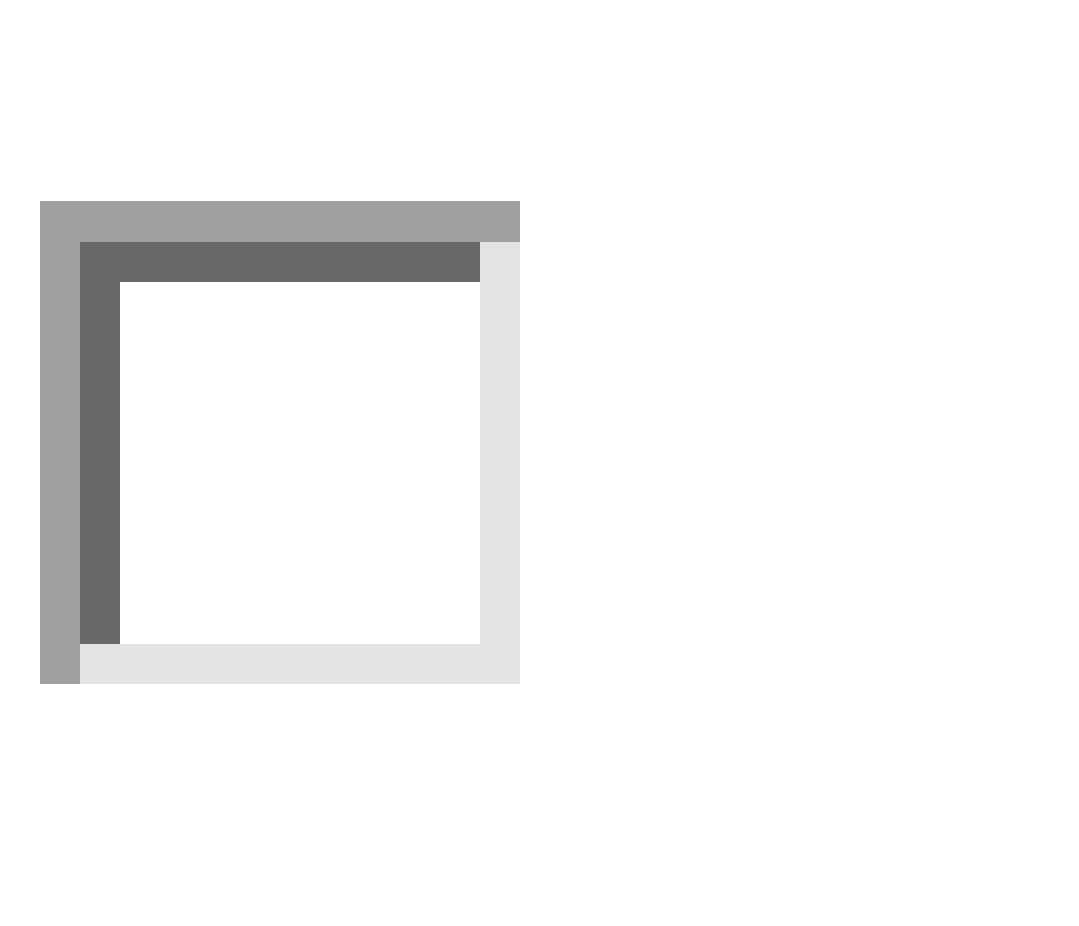
1. **Scheduling Algorithm Implemented**

✅ Tick the scheduling algorithm your group implemented:

FCFS (First Come First Serve)

☑ **SJF (Shortest Job First – Non-Preemptive)**

SJF (Preemptive)

Round Robin

1. **Project Description**

Our project implements a flight scheduling system using the non-preemptive Shortest Job First algorithm. The system:

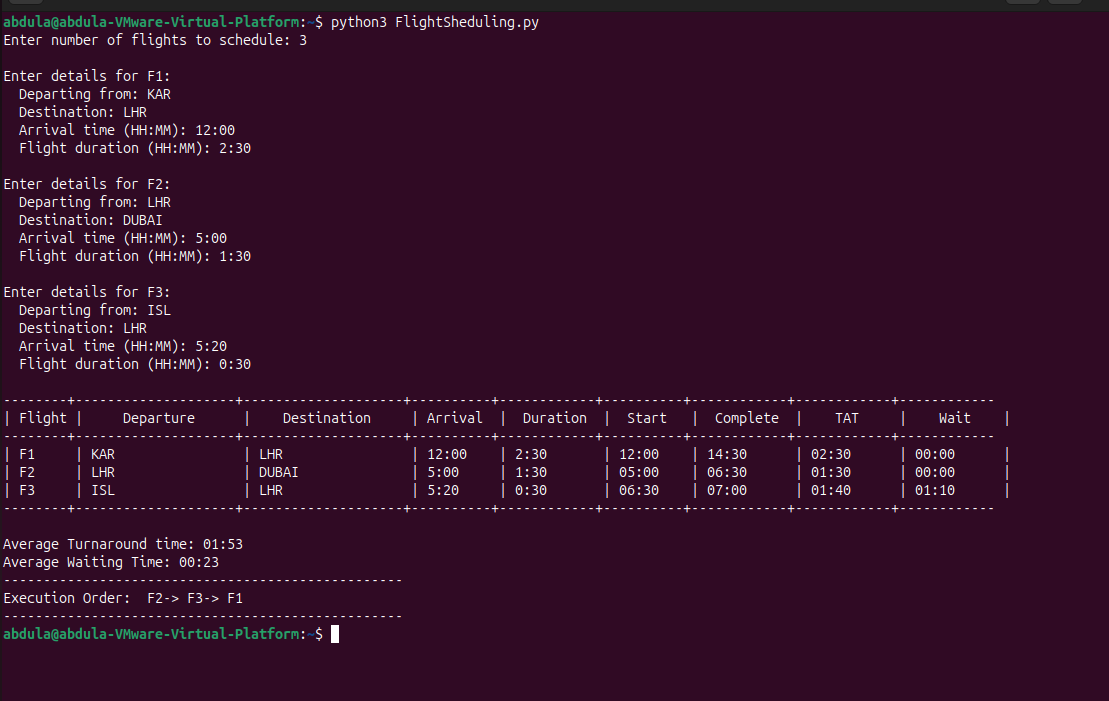
* Accepts flight details including departure city, destination, arrival time, and duration
* Schedules flights based on shortest duration first
* Calculates key performance metrics:
* Start and completion times
* Turnaround time (total time from arrival to completion)
* Waiting time (time spent waiting before execution)
* Generates a clear execution order and comprehensive statistics

**Inputs Required:**

Number of flights to schedule

* For each flight:
* Departure city
* Destination city
* Arrival time (HH:MM format)
* Flight duration (HH:MM format)

1. **Output Screenshots**



1. **Code Structure & Explanation**

The code is organized into several key components:

**1. Time Handling Functions:**

- validate\_time(): Validates and converts HH:MM format to minutes

- minutes\_to\_time(): Converts minutes back to HH:MM format

**2. Input Collection:**

- Gets number of flights with validation

- Collects flight details with robust input validation

**3. SJF Scheduling Algorithm:**

- Sorts flights by arrival time

- Processes flights using non-preemptive SJF:

* Selects shortest duration flight from ready queue
* Starts immediately if no competing flights
* Calculates all scheduling metrics

**4. Results Display:**

- Well-formatted table output

- Calculated averages

- Execution sequence

1. **Performance Metrics**

| Metric | Value |
| --- | --- |
| Average Waiting Time | Calculated |
| Average Turnaround Time | Calculated |
| Time Quantum (if RR) | N/A (SJF) |

1. **Challenges Faced**

**1. Time Validation:**

Initially struggled with validating HH:MM format inputs. Solved by creating dedicated validation functions that check both format and value ranges.

**2. Unnecessary Waiting Times:**

Early versions had artificial delays even when no flights were competing. Fixed by implementing immediate start when no conflicts exist.

**3. Output Formatting:**

Creating clean, aligned tables was challenging. Developed a dynamic formatting system that adjusts based on column widths.