
Software Requirements Specification

for

Airport Management System

Version 1.0

Prepared by Armaan Gupta, Suvan Kumar, Gokul Nair

Manipal Institute of Technology

29/3/2025

Table of Contents

Table of Contents	ii
Revision History	ii
1. Introduction.....	1
1.1 Purpose.....	1
1.2 Document Conventions.....	1
1.3 Intended Audience and Reading Suggestions	1
1.4 Product Scope	1
1.5 References.....	1
2. Overall Description	2
2.1 Product Perspective.....	2
2.2 Product Functions	2
2.3 User Classes and Characteristics	2
2.4 Operating Environment.....	2
2.5 Design and Implementation Constraints	2
2.6 User Documentation	2
2.7 Assumptions and Dependencies	3
3. External Interface Requirements	3
3.1 User Interfaces	3
3.2 Hardware Interfaces	3
3.3 Software Interfaces	3
3.4 Communications Interfaces	3
4. System Features	4
4.1 Login.....	5
4.2 Schedule Management	5
4.3 Passenger Information System.....	6
5. Other Nonfunctional Requirements	7
5.1 Performance Requirements	7
5.2 Safety Requirements	7
5.3 Security Requirements	7
5.4 Software Quality Attributes	7
5.5 Business Rules	7
6. Other Requirements	7
Appendix A: Glossary.....	8
Appendix B: Analysis Models	8
Appendix C: To Be Determined List.....	8

Revision History

Name	Date	Reason For Changes	Version

1. Introduction

1.1 Purpose

Product: Airport Management System

Version: 1.0

The AMS covers most of the operational aspects of an airport which includes, but is not limited to, baggage handling, staff management, boarding gate announcements, flight accommodation etc. The scope of this project is focused on a single airport.

1.2 Document Conventions

IEEE Standard 830: The SRS follows the IEEE 830 standard for software documentation.

Headings and Content: Headings use 14px Times New Roman font. Respective contents use 11px Times New Roman font.

Hierarchy: Subheadings, if any, are marked hierarchically (e.g.: 1.1, 1.1.1, 2.1 etc.).

1.3 Intended Audience and Reading Suggestions

This document is intended for developers and testers involved in the implementation of the Airport Management System. The document starts with an introduction, followed by an overall description, detailed requirements, and appendices for reference. For most readers, it is recommended to start with the overview sections to understand the system's purpose and scope. The readers can then proceed to the sections relevant to their roles.

1.4 Product Scope

The AMS is designed to automate the management of operations throughout the airport. It aims to coordinate flight schedules with the passenger boarding schedule, synchronize baggage arrivals with arrival schedules, ensure efficient cooperation and coordination among staff members as well as between the staff and the customer. This document alone tries to consolidate and provide a holistic view of the features and specifications of the product.

1.5 References

Due to the projects' scope being local and limited, it does refer to other online sources at present.

2. Overall Description

2.1 Product Perspective

The AMS is a self-contained product designed to enhance the operational efficiency of airports. It is not part of a product family or a follow-on member but rather a standalone system designed for the sole purpose of the demonstration of the working of already established systems in the industry. AMS provides integrated solutions for flight scheduling, passenger management, baggage handling, and resource allocation.

2.2 Product Functions

The AMS aims to provide the following interactional functionalities for the user(s):

- *The passengers must be able to log in and make a booking*
- *The staff must be able to log in and change the database for flight delays*
- *The passengers must have an option to view their flight status, baggage status and flight schedule details.*
- *The passengers must have a brief of the information forwarded from the airport to their respective airlines.*
- *The staff must have an option to manage the gate numbers for a particular airline, which should reflect for the passengers associated with that airline*

2.3 User Classes and Characteristics

The following are the expected classes which should be a part of AMS:

- *Passenger (Main end-user)*
- *Staff (Airport staff having administrator privileges)*
- *Baggage (Baggage management controlled directly by the staff)*
- *Gate (Gate management controlled directly by the staff)*
- *Flight (Flight details acquired by the passenger which would be forwarded to the respective airline)*
- *Lost and Found (Lost baggage database controlled directly by the staff)*

2.4 Operating Environment

The software is designed to be compatible with Microsoft Windows operating system, Windows 10 and 11 should be comfortably able to run the software without performance constraints. The project prototype is expected to run smoothly, without performance drops on an intel i5 8-core CPU and a 16 GB RAM module.

2.5 Design and Implementation Constraints

The current state of the product is presented only as a minor project without industrial deployment; therefore, it is not limited by corporate or regulatory policies nor any specific hardware limitations. The AMS is, however, incompatible with Debian based operating Systems at its current developmental state. The product is also only available in English, adding a language constraint towards the final product.

2.6 User Documentation

The software will be delivered along with an Entity-Relationship Diagram, a Progress Report and a virtual prototype.

2.7 Assumptions and Dependencies

The AMS assumes and depends on the following factors:

- *Stable and reliable internet connection*
- *Internet proficiency of end-users*
- *Pre-configured hardware (e.g., scanners, baggage conveyor systems, airline APIs)*
- *Accurate and synchronized data by third-party systems such as airline reservation platforms and air traffic management databases*

3. External Interface Requirements

3.1 User Interfaces

- *Staff login page: dedicated login for staff*
- *Baggage check-in page: check in portal for staff access only*
- *Lost and Found: for the staff to keep record of lost items*
- *Main dashboard: Staff/Passenger main navigation dashboard*
- *Gate: Departure gate information for the passengers*
- *Delay: Flight delay changes control for the staff*

3.2 Hardware Interfaces

The hardware and software interaction boils down to the interaction of staff towards making changes in the system. The staff have a login portal which they interact through the hardware provided by the airport.

3.3 Software Interfaces

The software uses the node.js JavaScript framework as its backend for fetching data from the database built using MySQL. The system communicates using HTTP requests from the frontend to fetch related data and display it on screen.

3.4 Communications Interfaces

The main communication function for general communication among employees and users would be e-mail. The communication protocol used by the software for all the tasks is HTTP.

4. System Features

4.1 Login

4.1.1 Description and Priority

Enables the passenger or the staff to login to view/customize data.

Priority: 8

4.1.2 Stimulus/Response Sequence

The login sequence is as follows:

- *The user selects their designation (passenger/staff)*
- *The user is shown a login form for entering relevant identification details*
- *The user enters their Adhaar ID for the parameter, while the staff enters their assigned ID*
- *Upon successful login, the user can view/modify the data relevant to them.*

4.1.3 Functional Requirements

REQ-1: User Authentication

REQ-2: Form Validation

REQ-3: User-role identification

4.2 Schedule Management

4.1.1 Description and Priority

Enables the staff to change flight times according to the data received from the airline/ATC.

Priority: 9

4.1.2 Stimulus/Response Sequence

The schedule management sequence is as follows:

- *The staff logs in from their staff ID*
- *The staff selects the available flights arriving/departing from the airport*
- *The staff is met with the option to change the flight arrival/departure time, which would be directly updated for all passengers.*
- *The staff confirms updating, which is reflected in the flight timings for the passengers taking the same flight.*

4.1.3 Functional Requirements

REQ-1: Staff Authentication

REQ-2: Form Validation

REQ-3: Automated updates for all relevant passengers

4.3 Passenger information display

4.1.1 Description and Priority

Enables the passengers to log in view their flight/baggage details

Priority: 8

4.1.2 Stimulus/Response Sequence

The information display sequence is as follows:

- *The passenger logs in from their authentication ID*
- *The passenger is met with a dashboard, containing multiple information selection modules/options*
- *The passenger selects the information relevant to them and is able to view and log out from the portal.*

4.1.3 Functional Requirements

REQ-1: Passenger Authentication

REQ-2: Form Validation

5. Other Nonfunctional Requirements

5.1 Performance Requirements

There is currently no performance requirements decided upon by the team (TBD).

5.2 Safety Requirements

Due to the limited scope of the project, the safety requirements are not clear at this stage of development (TBD).

5.3 Security Requirements

Due to the limited scope of the project, the safety requirements mainly include correct authentication with respect to the login modules for the staff and passengers.

5.4 Software Quality Attributes

The AMS is expected to provide the following software quality attributes:

- *Adaptability*
- *Flexibility*
- *Maintainability*
- *usability.*

5.5 Business Rules

The bare minimum business rule under efficient operational condition of the product include the staff access of the flight timing database and passenger access to the passenger and flight related information.

6. Other Requirements

Other Requirements include a normalized and efficiently defined database to reduce redundancy to the minimum and ensure faster load times. The software also requires a well-documented, indented and readable codebase, following modern coding practices and ensuring modularity for easy scaling.

Appendix A: Glossary

Due to the simple nature and scope of the project, there are no abbreviations/glossary requirements specified.

Appendix B: Analysis Models

Entity-relationship diagrams will be submitted during lab sessions.

Appendix C: To Be Determined List

The TBD list currently includes:

- *Performance requirements*
- *Safety requirements*
- *Glossary (if needed)*