Airport Management System

**High Level Design**

Version 1.0



|  |  |  |  |
| --- | --- | --- | --- |
|  | **Prepared By/Last Updated by** | **Reviewed By** | **Approved By** |
| **Name** | **(ADM21DF006)**   1. **VISHAL KUMAR(922322)** 2. **SUMIT KUMAR LABH(922114)** 3. **SUMIT KUMAR(922500)** 4. **ABHISHEK KUMAR(921619)** 5. **HARSHIT ANAND(922269)** |  |  |
| **Role** | 1. **VISHAL KUMAR (Pod Leader)** 2. **SUMIT KUMAR LABH(Member)** 3. **SUMIT KUMAR(Member)** 4. **ABHISHEK KUMAR(Member)**   **5.HARSHIT ANAND (Member)** |  |  |
| **Signature** |  |  |  |
| **Date** |  |  |  |



# Table of Content

* 1. Introduction 3
  2. [Purpose of this document 3](#_TOC_250009)
  3. [Project overview 3](#_TOC_250008)
  4. Solution Summary 3
  5. [Scope 3](#_TOC_250007)
  6. [Assumption 3](#_TOC_250006)
  7. [Dependencies 3](#_TOC_250005)
  8. [Risks 3](#_TOC_250004)
  9. Schematic Diagram 4
  10. [Dataflow Diagram 4](#_TOC_250003)
  11. [System Architecture 5](#_TOC_250002)
  12. System Design 6
  13. Proposed Design 11
  14. Database Design 12
  15. Data Model 15
  16. Appendices 16
  17. Glossary 16

7.0 Terms & Conditions 16



**1.0 Introduction**

## Purpose of this document

The purpose of this document is to document the technical design, component details and Database design. This will also capture the scope, assumptions, risk, dependencies of the Airport Management System project.

## Project overview

Our project is the design of a Airport Management System with the help of which a Employee can manage the flight and pilot schedule. The key features of this project are:

* + - Admin have right to register the employee as well as admin can add and schedule flight.
    - Employee can see his time table in the dashboard.
    - It is fast, reliable and easy to use.

**2.0 Solution Summary**

# 

# Scope

To provide an overview of the design document and creation of an application to all the work related to airport management like flight scheduling, time management, allocate and deallocate hanger.

# Assumption

1. Admin details are hard-coded for Admin-Login Verification.

# Dependencies

This application does not depend on any external software integrations for any microservice.

# Risks

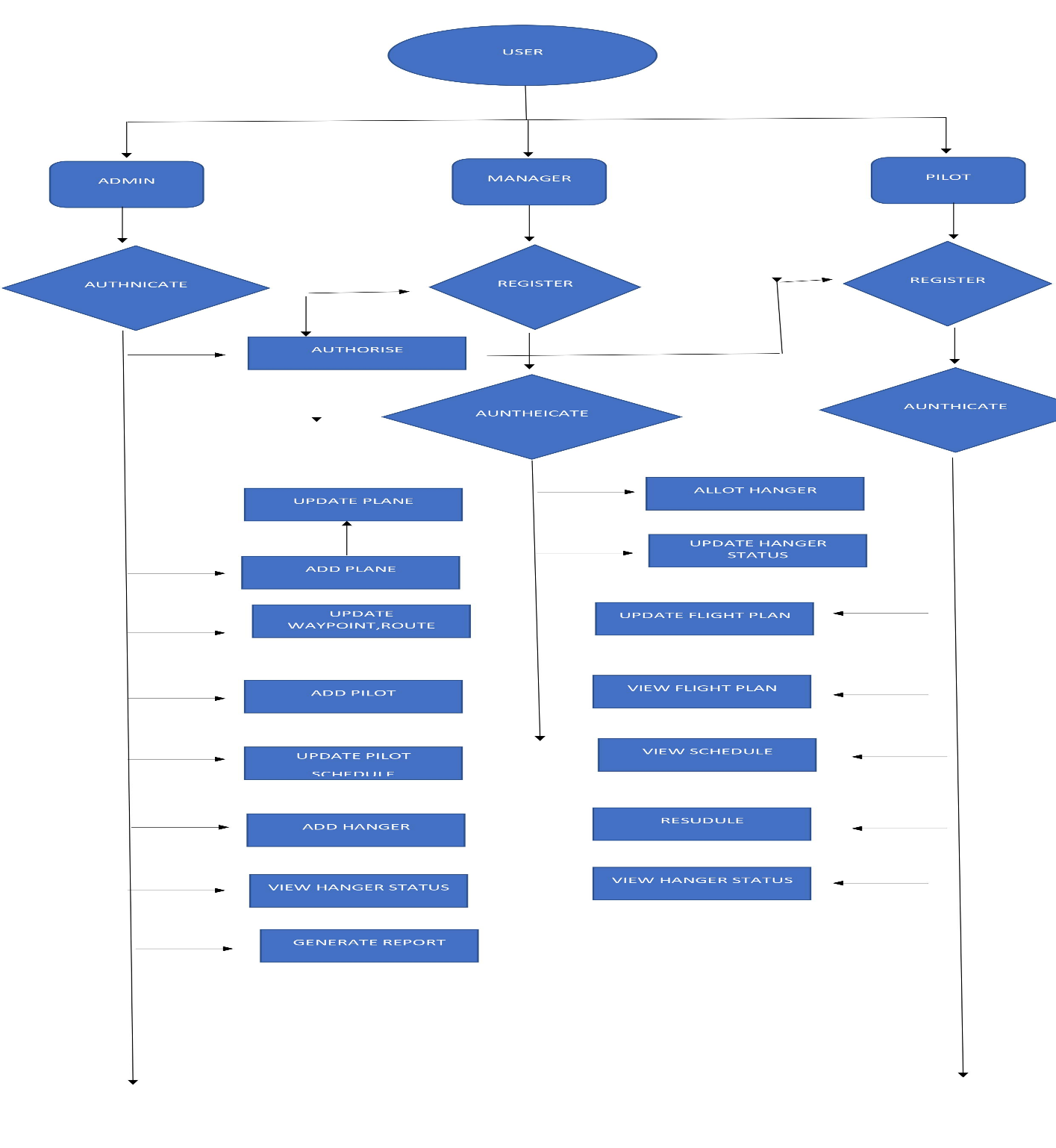
1. Working behind schedule.
2. Efficiency in coding.



**3.0 Schematic Diagram**

A schematic, or schematic diagram, is a representation of the elements of a system using abstract, graphic symbols rather than realistic pictures. It gives an overview of overall system.

## Data Flow Diagram

****



## System Architecture

**Application Layer**

It holds the UI of the application. Input is taken from the end user in this layer. It is a design layer where whole UI is designed through which end user can interact.

**Business Logic Layer**

It performs all the business operations and rules

**Microservice**

It takes data from database and connects the business layer with database. It performs all data manipulation operations (CURD Operation).



**Database**



**4.0 System Diagram**

# Proposed design

This section discusses about the detailed design for each user story. It displays information about the ID, name, role, description, precondition, post-condition, and the basic path that the user story follows in the project.

## Role Selection

|  |  |
| --- | --- |
| **User Story Id** | US\_01 |
| **User Story Name** | Initial Selection |
| **Role** | All Users |
| **Description** | the user to choose Admin / Manager / pilot Login/Registration |
| **Basic Path** | 1)Screen should display the option for Admin login / Registration and Manager login / Registration and |

* + 1. **Employee Registration**

|  |  |
| --- | --- |
| **User Story Id** | US\_02 |
| **User Story Name** | Employee Registration |
| **Role** | Employee |
| **Description** | procure the fundamental details of the Manager |
| **Basic Path** | 1)When the Employee clicks on the registration link, it should re-direct to registration form.  2)Employee needs to fill some of the basic attributes/fields as mentioned below in requirement: First Name, Last Name, DOB, Gender, Contact Number, Employee Id, Password etc.  3)Before entering the next field, each field should be validated and appropriate error message should be displayed near the corresponding text box.  4) Admin should be able to view the New Managers registration  5) Admin should approve / reject the Manager Request.  6) 10-Manager should be notified upon approvalrejection while trying to enter the application. |



* + 1. **Employee Authentication**

|  |  |
| --- | --- |
| **User Story Id** | US\_03 |
| **User Story Name** | Employee Authentication |
| **Role** | Employee |
| **Description** | Authenticate the Manager credentials of the registered Manager. |
| **Basic Path** | 1) A registered Employee – is able click ‘Login’ link, after keying in ‘Employee ID’ & ‘Password’ field and get his credentials authenticated with the existing database entry. |

* + 1. **Admin Authentication**

|  |  |
| --- | --- |
| **User Story Id** | US\_04 |
| **User Story Name** | Admin Authentication |
| **Role** | Admin |
| **Description** | Authenticate the Manager credentials of the registered Manager. |
| **Basic Path** | 1. Admin User id and password can be preloaded in the database and use it to login into the system. 2. On keying in ‘Admin ID’ & ‘Password’ field and get his credentials authenticated with the existing database entry. |

* + 1. **View registration request**

|  |  |
| --- | --- |
| **User Story Id** | US\_05 |
| **User Story Name** | View registration request |
| **Role** | Admin |
| **Description** | Admin to view the registration request |
| **Basic Path** | 1. Admin should be able to view the registration request awaiting approval |

* + 1. **Approve / Deny manager registration**

|  |  |
| --- | --- |
| **User Story Id** | US\_06 |
| **User Story Name** | Approve / Deny manager registration |
| **Role** | Admin |
| **Description** | Admin to approve or deny the registration request. |
| **Basic Path** | 1. Admin should be able to approve or deny the registration request. 2. Upon approval, Manager should be able to log in successfully. 3. Upon denial, message - "Your Registration is denied by Admin" should be display when they tried to log in. |

* + 1. **Add planes / update planes**

|  |  |
| --- | --- |
| **User Story Id** | US\_07 |
| **User Story Name** | Add planes / update planes |
| **Role** | Admin |
| **Description** | Admin to approve or deny the registration request. |
| **Basic Path** | 1. On clicking “Add Planes”, it should redirect to the page to add Plane. 2. Admin should be able to update the mandatory fields of the Planes. Plane id, Model number, Purchase date, Last serviced date etc can be fields. 3. Clicking ‘Submit’ should validate the datatype constraints for each field 4. Admin failing to provide information on the mandatory fields be provided with an alert message – ‘Please update the highlighted mandatory field(s).’ Also, highlight the missed out field in red 5. Post-successful field level validation, save the information in the database 6. Upon saving the information in the database, display the message ‘Your details are submitted successfully’. 7. Admin should be able to view the list of planes by clicking “View Planes” link. 8. On clicking the Plane ID, the system should display the details of the planes. Plane Id should be used as primary key the DB and should not be editable. 9. Admin should be able to update the details of the plane on clicking “Update details” button 10. On clicking “Save” button, the updated details should be saved to the database. Upon saving the information in the database, display the message ‘Your details are submitted successfully’. |

* + 1. **Update Flight Plan**

|  |  |
| --- | --- |
| **User Story Id** | US\_08 |
| **User Story Name** | Update Flight Plan |
| **Role** | Pilot |
| **Description** | Pilot to update the flight plan. |
| **Basic Path** | 1. Pilot on clicking Update Flight Plan should be redirected to the Flight Plan update page. 2. Pilot should update mandatory fields - departure location, waypoints, estimated time of arrival, destination location, alternate airports, aircraft information, and the pilot’s information. 3. Clicking ‘Submit’ should validate the datatype constraints for each field 4. Pilot failing to provide information on the mandatory fields be provided with an alert message – ‘Please update the highlighted mandatory field(s).’ Also, highlight the missed out field in red 5. Post-successful field level validation, save the information in the database 6. Upon saving the information in the database, display the message ‘Your details are submitted successfully’. |

* + 1. **Update Post-flight mode**

|  |  |
| --- | --- |
| **User Story Id** | US\_09 |
| **User Story Name** | Update Post-flight mode |
| **Role** | Pilot |
| **Description** | gather information during flight mode. It should display the information regarding future flights. |
| **Basic Path** | 1. Pilot should be able to update the pre flight mode checklist. 2. On clicking the Pre flight mode tab, all the flight that requires pre flight check should be displayed. 3. On clicking the flight the pilot want to update, it should open the checklist form. 4. Pilot should update the checklist 5. Manager/Pilot should be able to view it |

* + 1. **Update Post-flight mode**

|  |  |
| --- | --- |
| **User Story Id** | US\_10 |
| **User Story Name** | Update Post-flight mode |
| **Role** | Pilot |
| **Description** | gather information during flight mode. It should display the information regarding future flights. |
| **Basic Path** | 1. Pilot should be able to update the post flight mode checklist. 2. On clicking the post flight mode tab, all the flight that requires post flight check should be displayed. 3. On clicking the flight the pilot want to update, it should open the checklist form. 4. Pilot should update the checklist 5. Manager/Pilot should be able to view it |

* + 1. **Update Resources**

|  |  |
| --- | --- |
| **User Story Id** | US\_11 |
| **User Story Name** | Update Resources |
| **Role** | Admin |
| **Description** | update the Waypoints, maps, and routes. |
| **Basic Path** | 1. Admin should be able to update the Resource details which is required for the pilot. 2. Admin should be able to update the Waypoints, maps, and routes. |

* + 1. **Target Generation Facility**

|  |  |
| --- | --- |
| **User Story Id** | US\_12 |
| **User Story Name** | Target Generation Facility |
| **Role** | Pilot |
| **Description** | know information about current flights or registered flight plans. |
| **Basic Path** | 1. Pilot should be able to get information about current flights or registered flight plans. |

* + 1. **Add pilots work schedule**

|  |  |
| --- | --- |
| **User Story Id** | US\_13 |
| **User Story Name** | Add pilots work schedule |
| **Role** | Admin |
| **Description** | add pilot’s work schedule details. |
| **Basic Path** | 1. Admin should be able to add pilot’s work schedule details. |

* + 1. **View schedule**

|  |  |
| --- | --- |
| **User Story Id** | US\_14 |
| **User Story Name** | View schedule |
| **Role** | Pilot |
| **Description** | I want to view the schedule |
| **Basic Path** | 1. Pilot on clicking view the schedule should redirect to the Pilot schedule page 2. The page should display the Pilots schedule. Pilot should have the option to view that days schedule, Weekly schedule and the schedule for the month. |

* + 1. **Request rescheduling**

|  |  |
| --- | --- |
| **User Story Id** | US\_15 |
| **User Story Name** | View schedule |
| **Role** | Pilot |
| **Description** | reschedule my work time. |
| **Basic Path** | 1. Pilot should be able to raise a request for rescheduling the work schedule if required.. |

* + 1. **View / approve rescheduling request**

|  |  |
| --- | --- |
| **User Story Id** | US\_16 |
| **User Story Name** | View / approve rescheduling request |
| **Role** | Admin |
| **Description** | reschedule my work time. |
| **Basic Path** | 1. Admin should be able to view the reschedule request from the pilot 2. On clicking each request ID, it should display the details. 3. Admin should be able to approve / deny the request. |

* + 1. **Update schedule**

|  |  |
| --- | --- |
| **User Story Id** | US\_17 |
| **User Story Name** | Update schedule |
| **Role** | Admin |
| **Description** | update pilot’s work schedule details. |
| **Basic Path** | 1. Upon approval, Admin should be able to update pilot’s work schedule details.. |

* + 1. **Add Hangars**

|  |  |
| --- | --- |
| **User Story Id** | US\_18 |
| **User Story Name** | Add Hangars |
| **Role** | Admin |
| **Description** | add Hangar details. |
| **Basic Path** | 1. On clicking “Add Hangars”, it should redirect to the page to add Plane. 2. Admin should be able to update the mandatory fields of the Hangars 3. Clicking ‘Submit’ should validate the datatype constraints for each field 4. Admin failing to provide information on the mandatory fields be provided with an alert message – ‘Please update the highlighted mandatory field(s).’ Also, highlight the missed out field in red 5. Post-successful field level validation, save the information in the database 6. Upon saving the information in the database, display the message ‘Your details are submitted successfully’. 7. Admin should be able to view the list of Hangars by clicking “View Hangars” link. 8. On clicking the Plane ID, the system should display the details of the Hangars. 9. Admin should be able to update the details of the plan on clicking “Update details” button 10. On clicking “Save” button, the updated details should be saved to the database. Upon saving the information in the database, display the message ‘Your details are submitted successfully’. |

* + 1. **Allotting planes to the Hangar**

|  |  |
| --- | --- |
| **User Story Id** | US\_19 |
| **User Story Name** | Allotting planes to the Hangar |
| **Role** | Manager |
| **Description** | allot planes to Hangar. |
| **Basic Path** | 1. When Manager clicks Allot Hangar, the system should display the list of Hangars available and the list of planes waiting for hanger 2. Manager should have the option to select the available hangar and allot it for a plane |

* + 1. **Update Hangar status**

|  |  |
| --- | --- |
| **User Story Id** | US\_20 |
| **User Story Name** | Update Hangar status |
| **Role** | Manager |
| **Description** | update the Hangar details. |
| **Basic Path** | 1. After allotting Manager should update the status as Available / Allotted 2. Manager on clicking “View Hangar Status”, the page should display the list of all Hangars with status |

* + 1. **View Hangar Status**

|  |  |
| --- | --- |
| **User Story Id** | US\_21 |
| **User Story Name** | View Hangar Status |
| **Role** | Manager/Admin |
| **Description** | view the Hangar details. |
| **Basic Path** | 1. Admin on clicking Reports should redirect to the reporting page. 2. It should display drop down Pilot schedule, Hanger allotment, Plane details. 3. On selecting Pilot schedule, it should display the Pilots work schedule. 4. On clicking Hanger allotment, should display all the available Hangers and the allotment details 5. -On clicking Plane details., should display all the available Planes and the details |

* + 1. **Generate Report**

|  |  |
| --- | --- |
| **User Story Id** | US\_22 |
| **User Story Name** | Generate Report |
| **Role** | Admin |
| **Description** | generate report on Pilot schedule, Hanger allotment, Plane details. |
| **Basic Path** | 1. During registration, System should pop up three secret questions for Password recovery. 2. When the user clicks Forgot User ID, system should ask for the secret questions. On answering the questions correctly, the User ID should be displayed. |

* + 1. **Other validations**

|  |  |
| --- | --- |
| **User Story Id** | US\_23 |
| **User Story Name** | Other validations |
| **Role** | Admin |
| **Description** | System to have validations like Forget User ID, Forget Password |
| **Basic Path** | 1. When the user clicks Forgot Password, system should ask for the User ID and secret questions. 2. On answering the questions correctly, the password reset page should be displayed. 3. On entering the details in the password reset page, password should be validated 4. On clicking Submit, the details should be saved to the Database |

* + 1. **Help**

|  |  |
| --- | --- |
| **User Story Id** | US\_24 |
| **User Story Name** | Help |
| **Role** | Admin |
| **Description** | Page to assist the users to report technical issues |
| **Basic Path** | 1. Page to display a form that allows user to report the technical issues. |

* + 1. **Manager / Admin logoff**

|  |  |
| --- | --- |
| **User Story Id** | US\_25 |
| **User Story Name** | Help |
| **Role** | All Users |
| **Description** | Ability of the system to enable Manager / Admin to logoff |
| **Basic Path** | 1. Option to log off from the system. |



**5.0 Database Design**



# Data Model

# Employee

# Table Description automatically generated

# Flight Plan

# Table Description automatically generated

# Hangar Details

# Table Description automatically generated

# Issue Table

# Table Description automatically generated

# Pilot Schedule

# Table Description automatically generated

# Plane

# Table Description automatically generated

# Post Flight Checklist

# Table Description automatically generated

# Pre Flight Schedule

# Table Description automatically generated

# Security Question

# Table Description automatically generated

# Request Rescheduling

# Table Description automatically generated



**6.0 Appendices**

**6.1 Glossary**

|  |  |
| --- | --- |
| **Acronyms** | **Definitions** |
| **Admin** | Owner of the system (Developer) |
| **End User** | Customers |

**7.0 Terms & Condition**

**Any employee cannot share their id, password and private data with anyone.**