PROJECT DOCUMENTATION

# Airlines Management System

Revision History

|  |  |  |  |
| --- | --- | --- | --- |
| Version No. | Date | Prepared by / Modified by | Significant Changes |
| 1 | 8 July , 2022 | Group 3 | Draft version |
|  |  |  |  |

Group Members -

* + - * Salve Vivek Gajanan
      * Naveeenraj M
      * Mohd Yasin Malik
      * Sakeena Banu
      * Sakeena Banu
      * G. Madhavan
      * Manish

Glossary

|  |  |
| --- | --- |
| Abbreviation | Description |
| DB | Database |
| DDB | Distributed Database |
| ER | Entity Relationship |

Table of Contents

[School Management 1](#_Toc106354638)

[1 INTRODUCTION 3](#_Toc106354639)

[1.1 Significations 3](#_Toc106354640)

[1.2 Scope 4](#_Toc106354641)

[1.3 Definitions, Acronyms and Abbreviations 4](#_Toc106354642)

[2 SCOPE OF CHANGE 4](#_Toc106354643)

[3 List of impacted module 4](#_Toc106354644)

[4 Design and Detailed technical updates 5](#_Toc106354645)

[4.1 Process model 5](#_Toc106354646)

[4.1.1 Use case Model 5](#_Toc106354647)

[6](#_Toc106354648)

[4.1.2 Overview Description 6](#_Toc106354649)

[4.2 Database design changes 8](#_Toc106354650)

[4.3 Architecture Diagram 8](#_Toc106354651)

[5 DETAILS OF ALTERNATIVE DESIGN APPROACH 9](#_Toc106354652)

[6 Additional details 10](#_Toc106354653)

[6.1 Software Requirements 10](#_Toc106354654)

[6.2 Non Functional Requirements 11](#_Toc106354655)

# INTRODUCTION

Airline Reservation System aims to automate the flight operations and ticketing / seat booking and confirmation system of an Airline company. The software is providing options for viewing different flights available within a different timings for a specific day.

That provide customers within facility to able to book ticket smoothly. The customers can modify and able to cancel the ticket for any reason.

That prepare within a role and policies. The software should provide option for checking availability of the tickets. That is important for the customers to get message if the ticket unavailable. That will be displayed into customers. The customers should be noted when the change has been made or any further changes.

## Significations

* Easy to book a flight ticket.
* Work becomes speedy.
* Access of any information individually.
* Decrease the load of the person involve in existing manual system.
* Well-designed reports.
* Easy & fast retrieval of information.
* Accuracy in work.
* It contains better storage capacity.
* Robust database back-end.
* Creating and changing data at ease.

## Scope

Our project aims to safe and easy book and cancel a flight ticket i.e. we have made a computerized process to store data and distribution.

* It satisfies the admin(Owner).
* It is easy and safe to store data.
* It is easy to operator.
* Have a good user interface.
* It saves time and function faster.
* It helps the owner to manage the restaurant.
* We have tried to develop safe and secure software with above mentioned specifications.

## Definitions, Acronyms and Abbreviations

**Airline management systems** are systems that allow an airline to sell their inventory (seats). It contains information on schedules and fares and contains a database of reservations (or passenger name records) and of tickets issued (if applicable).

**Airline management systems**  are part of [passenger service systems](https://en.wikipedia.org/wiki/Passenger_service_system) , which are applications supporting the direct contact with the passenger..

# SCOPE OF CHANGE

Change will not be entertained and whatever specifications mentioned in this document is final.

# List of impacted modules

All the functional modules will be created from scratch.

# Design and Detailed technical updates

## 

will

Login

ER-DIAGRAM FOR AIRLINES MANAGEMENT SYSTEM

will

Passenger

Pay

Register

add

booked

Admin

Ticket

Flight

### Overview Description

|  |  |
| --- | --- |
| Brief Description | Admin Login |
| Basic Flow | This use case describes how a client register in to the system   1. The admin has to login himself into the system. 2. After the successful login, admin will get a success message, and admin can add a flight details. 3. The following information is required during login.  * Name * Password |
| Alternate Flow | 1. The system will validate the information provided. If any invalid data is found, the input form will be redirected with error message. |
| Validation | 1. Name is required and can’t be empty. 2. Password should be valid. |
| Pre-Conditions | User should have network access and Browser with latest updates. |
| Post-Conditions | Success message should be shown. |

|  |  |
| --- | --- |
| Brief Description | User Registration |
| Basic Flow | This use case describes how a supplier register in to the system   1. The user are being registered by the admin into the system. 2. After the successful registration, user will get a success message. 3. The following information is required during registration.  * User Name * Email id * Password * Mobile no. |
| Alternate Flow | 1. The system will validate the information provided. If any invalid data is found, the input form will be redirected with error message. |
| Validation | 1. User Name is required. 2. Valid email id is required. 3. Password is required. 4. Valid mobile no. is required. |
| Pre-Conditions | User should have network access and Browser with latest updates. |
| Post-Conditions | Success message should be shown. |

|  |  |
| --- | --- |
| Brief Description | User Login |
| Basic Flow | This use case describes how a user log-in in to the system   1. After the successful login, user will be taken to the appropriate landing page. 2. The following information is required to login.  * Username * Password |
| Alternate Flow | 1. The system will validate the credentials provided. If credentials are invalid, login form will be redirected again with error message. |
| Validation | 1. Valid username 2. Valid password |
| Pre-Conditions | User should have network access and Browser with latest updates. |
| Post-Conditions | Landing page has to be displayed. |

## Database design changes

|  |  |  |  |
| --- | --- | --- | --- |
| NAME OF FIELD | TYPE | SIZE | Description |
| Id | number | 12 | Primary key |
| Username | Varchar2 | 50 | Not null, unique |
| Password | Varchar2 | 20 | Not null |
| Email id | Varchar2 | 200 |  |
| Mobile no | number | 10 |  |

## Architecture Diagram

**Logical View**

**Technology/ Framework**

**Layer**

UI Components

Angular

Presentation Layer

Add

delete

Search

Update

Web-Server Classes

Java 1.8/ Spring

Boot

Application

Layer

Controllers

Configuration

REST Controllers

Service

Entity/ Model Classes

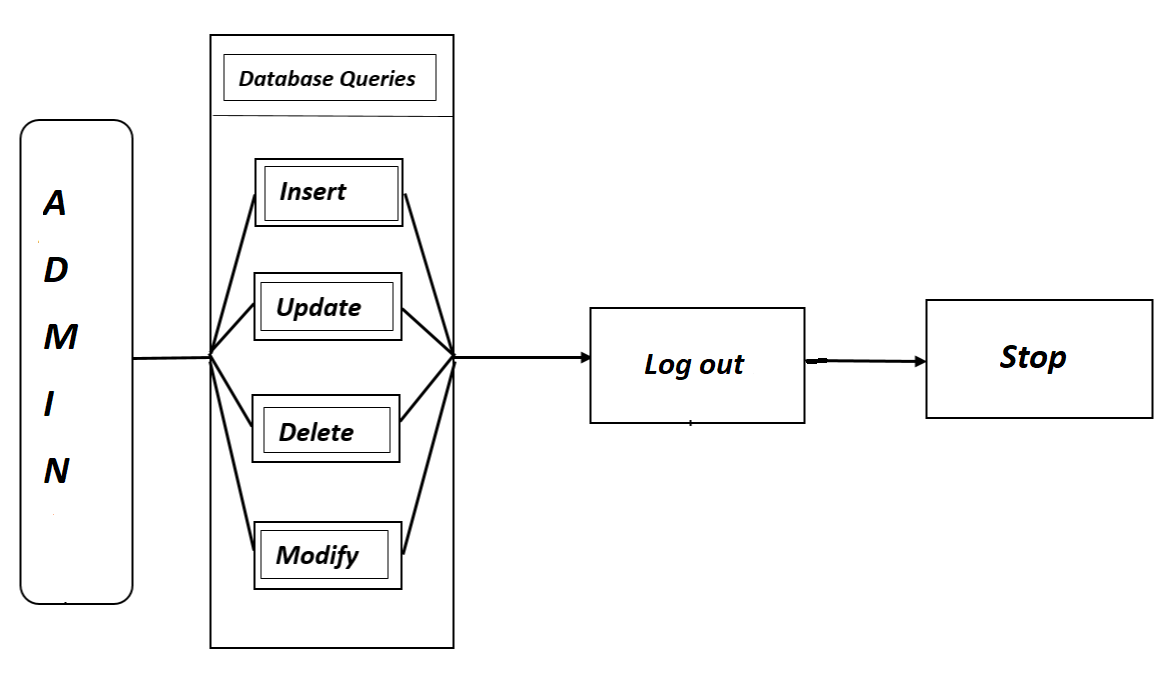
Data Access Layer

Spring JPA

Repository classes

Oracle 11g Database

# DETAILS OF ALTERNATIVE DESIGN APPROACH



# Additional details

## 6.1 Software Requirements

* Windows version 7+.
* A browser which supports CGI, HTML & JavaScript.
* Front-end software:

**HTML** - HTML (Hypertext Markup Language) is the code that is used to structure a webpage and its content. For example, content could be structured within a set of paragraphs, a list of bulleted points, or using images and data tables.

**CSS** - Stands for "Cascading Style Sheet." CSS sheets are used to format the layout of Web pages. They can be used to define text styles, table sizes, and other aspects of Web pages that previously could only be defined in a page's HTML.

**Angular** - Angular is a platform and framework for building single-page client applications using HTML and Type Script. Angular is written in TypeScript. It implements core and optional functionality as a set of Type Script libraries that you import into your applications. The architecture of an Angular application relies on certain fundamental concepts.

* Back-end software:

**JAVA** - Java is a general-purpose, class-based, object-oriented programming language designed for having lesser implementation dependencies. It is a computing platform for application development. Java is fast, secure, and reliable, therefore. It is widely used for developing Java applications in laptops, data centers, game consoles, scientific supercomputers, cell phones, etc.

**Spring Boot** - Spring Boot is an open-source micro framework maintained by a company called Pivotal. It provides Java developers with a platform to get started with an auto configurable production-grade Spring application. With it, developers can get started quickly without losing time on preparing and configuring their Spring application.

**Spring Data JPA** - JPA is a Java specification that is used to access, manage, and persist data between Java object and relational database. It is a standard approach for ORM.

**Spring Security** - Spring Security is a powerful and highly customizable authentication and access-control framework. It is the de-facto standard for securing Spring- based applications. Spring Security is a framework that focuses on providing both authentication and authorization to Java applications.

* Database:

**Oracle SQL** - SQL (pronounced sequel) is the set-based, high-level declarative computer language with which all programs and users access data in an Oracle database. Although some Oracle tools and applications mask SQL use, all database tasks are performed using SQL.

## 6.2 Non Functional Requirements

* **AVAILABILITY:** The flight should be available on the specified date and specified time as Admin may need it any time.
* **CORRECTNESS:** Correct data should be displayed.
* **MAINTAINABILITY:** The administrators should maintain correct data.
* **USABILITY:** Can be used by Admin.