Report on ASP.NET Project: Airports Management System

Introduction

The Airports Management System is an ASP.NET project designed to streamline the operations of an airport. This application encompasses five key tables: Airport, Facility, Flight, Passenger, and Staff. By leveraging the capabilities of ASP.NET, this project aims to provide a comprehensive solution for managing various aspects of airport operations, enhancing both efficiency and user experience.

Body

Overview of Tables

1. \*Airport Table\*

- \*Purpose\*: This table stores essential information about each airport, including its name, location, and code.

- \*Key Attributes\*:

- AirportID: Unique identifier for each airport.

- Name: Name of the airport.

- Location: Geographical location of the airport.

- Code: Short code used to identify the airport.

2. \*Facility Table\*

- \*Purpose\*: This table catalogs the facilities available at each airport, such as lounges, parking, and restaurants.

- \*Key Attributes\*:

- FacilityID: Unique identifier for each facility.

- AirportID: Foreign key linking to the Airport table.

- FacilityType: Type of facility (e.g., lounge, parking).

- Description: Brief description of the facility.

3. \*Flight Table\*

- \*Purpose\*: This table manages flight-related information, including schedules and destinations.

- \*Key Attributes\*:

- FlightID: Unique identifier for each flight.

- FlightNumber: Designation for the flight.

- DepartureAirportID: Foreign key referencing the departure airport.

- ArrivalAirportID: Foreign key referencing the arrival airport.

- DepartureTime: Scheduled departure time.

- ArrivalTime: Scheduled arrival time.

4. \*Passenger Table\*

- \*Purpose\*: This table holds information about passengers traveling through the airport.

- \*Key Attributes\*:

- PassengerID: Unique identifier for each passenger.

- Name: Passenger's full name.

- PassportNumber: Identification number for international travel.

- FlightID: Foreign key linking to the Flight table.

5. \*Staff Table\*

- \*Purpose\*: This table records details about airport staff, including roles and contact information.

- \*Key Attributes\*:

- StaffID: Unique identifier for each staff member.

- Name: Staff member's full name.

- Role: Position held at the airport (e.g., security, management).

- ContactNumber: Phone number for co

Functionality

The Airports Management System provides several functionalities, including:

- \*Flight Management\*: Users can add, modify, or delete flight information, allowing real-time updates to schedules.

- \*Passenger Management\*: The system allows for easy tracking of passenger details, ensuring quick access to information when needed.

- \*Facility Management\*: Users can manage airport facilities, including adding new facilities or updating existing ones.

- \*Staff Management\*: The system enables the management of airport staff, ensuring that roles and responsibilities are clearly defined.

### Technology Stack

The project utilizes the following technologies:

- \*ASP.NET\*: For building the web application.

- \*SQL Server\*: For database management and storage.

- \*HTML/CSS/JavaScript\*: For front-end development, ensuring a responsive and user-friendly interface.

- \*Entity Framework\*: For data access and manipulation.

Conclusions

The Airports Management System is a robust solution for managing airport operations efficiently. By utilizing a well-structured database with five key tables, the application facilitates the seamless management of flights, passengers, facilities, and staff. The integration of ASP.NET technologies ensures a responsive and interactive user experience. Future enhancements could include additional features such as online booking, real-time flight tracking, and advanced reporting capabilities. Overall, this project represents a significant step forward in airport management technology.