International Journal of Scientific Research in Engineering and Management (IJSREM)
Volume: 09 Issue: 05 | May - 2025 SJIF Rating: 8.586 ISSN: 2582-3930

The Pipeline Maestro – Flight Reservation System

[1]Dr.Mrs.R.A Pimpale, [2]Prathamesh Ghule, [2]Sameer Kute, [2]Himanshu Khapre

[1]Assistant Professor, Department of Information Technology, Priyadarshini Bhagwati College of Engineering, Nagpur, Maharashtra, India

[2]UG Student, Department of Information Technology, Priyadarshini Bhagwati College of Engineering, Nagpur, Maharashtra, India

radhapimpale@gmail.com, prathameshghule777@gmail.com, sameerkute9@gmail.com, Nikhilkhapre8@gmail.com

Abstract: **The aviation industry has undergone significant transformation due to advancements in cloud computing. Traditional flight reservation systems often face challenges related to scalability, security, and availability. The increasing demand for reliable and efficient booking platforms has necessitated the adoption of cloud- based solutions that offer high availability, cost efficiency, and enhanced user experience. This paper presents the development and implementation of a Cloud-Based Flight Reservation System using Amazon Web Services (AWS). The system leverages AWS services such as EC2, S3, RDS, Lambda, and API Gateway to provide a robust, scalable, and secure booking platform. By using a microservices architecture and serverless computing, the system ensures reduced latency, improved performance, and seamless handling of concurrent user requests. Furthermore, AWS Identity and Access Management (IAM) is employed to enhance security, preventing unauthorized access to sensitive flight and user data.It allows users to search for flights, book tickets, and make secure payments, while enabling administrators to manage flight schedules, reservations, and user data. Built with scalability and security in mind, the system integrates modern technologies for real-time booking and payment processing, ensuring a seamless and reliable experience for all users.**

Keywords: **Amazon Web Services (AWS), EC2, S3, Identity and Access Management (IAM)**

I. INTRODUCTION

Pipeline Maestro is a comprehensive software solution designed to optimize and automate the software development lifecycle, specifically focusing on the orchestration of build, test, and deployment processes. Key features include automated build and test capabilities, seamless integration with version control systems, deployment automation across multiple environments, monitoring and reporting functionalities, and customization options to accommodate diverse development needs.

In today's fast-paced world, air travel plays a crucial role in global connectivity, facilitating business, tourism, and personal travel. Millions of passengers rely on airline services every day, making the efficiency and reliability of flight reservation systems critical. The process of booking a flight should be seamless, secure, and accessible from anywhere in the world. However, traditional flight reservation systems, often based on legacy on-premise infrastructure, face numerous challenges related to performance, security, scalability, and operational costs.

As air travel demand grows, airlines must adopt modern technological solutions to manage increasing passenger numbers while ensuring high service availability. The introduction of cloud computing has provided an innovative approach to overcoming these challenges. Cloud-based systems

offer scalability, cost savings, high availability, and enhanced security, making them ideal for managing critical airline operations.

AWS (Amazon Web Services) has emerged as a leading cloud service provider, offering various services that enable businesses to build robust, scalable, and cost-efficient applications. The implementation of a Cloud- Based Flight Reservation System using AWS leverages the advantages of cloud computing to enhance airline booking processes, optimize resource allocation, and ensure secure transactions. By adopting AWS technologies such as EC2 for computing, RDS for database management, S3 for storage, and Lambda for serverless computing, this study aims to develop a highly efficient and resilient flight reservation system. This thesis explores the significance of cloud-based reservation systems, their benefits over traditional booking platforms, and the impact of AWS services in transforming airline operations. It provides insights into the architecture, development, and deployment of a cloud-based flight reservation system while evaluating its performance, security, and scalability.

II. LITERATURE SURVEY

Title of paper	**Method/ Techniques used**	**Analysis and Observation**
"PIPELINE MAESTRO – ORCHESTRATING BUILD, TEST, AND DEPLOY", International Journal of Emerging Technologies and Innovative Research, June-2024	Use of CI/CD tools, Terraform for IaC, Ansible, Puppet etc for configuration management, and orchestration tools like Docker Swarm is used	Various tools are used for automation and deployement process with which orchestration is achieved
"FLIGHT RESERVATION SYSTEM", International Journal of Creative Research Thoughts (IJCRT), June 2022	MySQL is used for database management along with .NET framework.	Users can book flights, view rates and book according to his/her needs. Inventory control method is used
"Airline Reservation System", International Journal of Innovative Science and Research Technology, June 2021	Java is used for frontend and MySQL is used for backend and DBMS	Easily one can see,book,and analysis the timing and rates of flights. Report any issue with it.
"A Next-Generation Approach to Airline Reservations: Integrating Cloud Microservices with AI and Blockchain for Enhanced Operational Performance", Institute of Information Technology, Jahangirnagar University, November 2024	Microservices Architecture, AI, Blockchain Technology and Cloud Computing	Airline services can be developed and made very reliable to use with the help of technologies like AI, Blockchain technology and cloud services.
"AIR RESERVE -An Airline Reservation System", *International Journal of Advanced Research in Science, Communication and Technology (IJARSCT), Dec* 2022	JavaScript, TypeScript and Node.js, languages and runtimes (such as C++, C#, Java, Python, Go, .NET).	customer can easily book Flight ticket. Air reserve is fully integrated airline reservation with advance customer management tools from reservation product merchandising, inventory and ticketing

III. PROPOSED METHODOLOGY

- **Frontend (Client-Side) -** The frontend is responsible for the user interface (UI) and user experience (UX) of the system. For Making a frontend we use S3 Bucket. S3 Bucket is a service of aws where we can store our frontend.

➢ Core Functionalities:
- User authentication (Login/Signup).
- Flight search with filters (date, destination, price, etc.).
- Booking system with seat selection and fare details.
- Payment gateway integration for transactions.
- Dashboard for users to view booking history.

- **Backend-** The backend handles server requests, authentication, and business logic. For configuration of Backend end we use EC2 service from aws. Firstly, we go to the EC2 and launch instance of image linux and configure other credentials and launch instance. Now we have our Backend in EC2 instance.

➢ Core Functionalities:
- User authentication & authorization (JWT/OAuth).
- Flight search logic (querying database efficiently).
- Booking system with availability checks.
- Payment processing & order management.
- API integration for real-time flight status.

➢ Security Measures:
- Encryption (SSL/TLS) – To protect user data.
- Role-Based Access Control (RBAC) – To manage user privileges.
- Rate Limiting – To prevent DDoS attacks.

- **Database (Storage & Management) -** The database stores user details, flight schedules, bookings, and transactions. For the database we use RDS (Relational Database Service) To story the data of user. RDS is basically use for the storing only so it is best for the storage. That's why we used RDS service as our database.

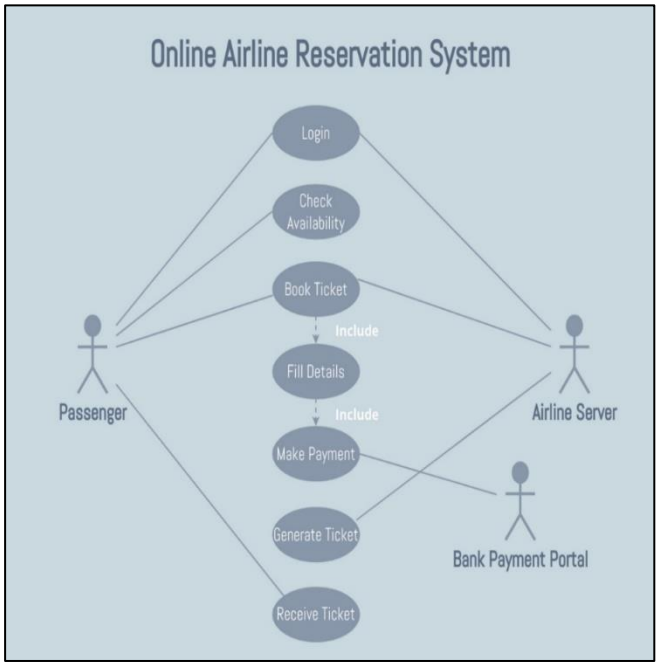
➢ Tech Stack Options:
- Relational (SQL-Based) – MySQL for structured and relational data.
- NoSQL (Document-Based) – MongoDB for flexibility and scalability.

➢ Core Database Schema:
- Users Table – Stores user details (ID, Name, Email, Password, Role).
- Flights Table – Contains flight details (Flight ID, Airline, Departure, Arrival, Price).
- Bookings Table – Stores booking info (User ID, Flight ID, Payment Status, Seats).
- Payments Table – Tracks transactions (Booking ID, Payment ID, Status).

➢ Database Management & Optimization:
- Indexes – To speed up searches and queries.
- Caching (Redis) – To store frequent queries and improve performance.
- Backup & Replication – To ensure data availability and recovery.

1. Block Diagram:

Figure 1. Block Diagram of Proposed System



2. Project Requirements:

2.1 Application Software Requirements

- Operating System - Windows 10 or Higher / Ubuntu 20.04+
- Languages & Frameworks - HTML, CSS, JavaScript, React, Node.js, Express.js, Python (Flask/Django)
- Database - MongoDB, PostgreSQL, AWS RDS
- Cloud Services - AWS EC2, AWS S3, AWS Lambda, AWS CloudFront
- Development Tools - VS Code, Postman, Docker, GitHub, AWS CLI

2.2 Backend Software Requirements

- Server-Side Frameworks - Express.js (Node.js), Flask/Django (Python)
- APIs - RESTful APIs, GraphQL
- Authentication & Security - JWT, OAuth 2.0, AWS IAM, AWS WAF

2.3 Hardware Requirements

- Processor - Intel Core i5 (8th Gen) / AMD Ryzen 5 or Higher
- RAM - 8GB or Higher
- Storage - 256GB SSD or Higher
- Network Connectivity - (10Mbps – 100Mbps)

IV. OUTPUT

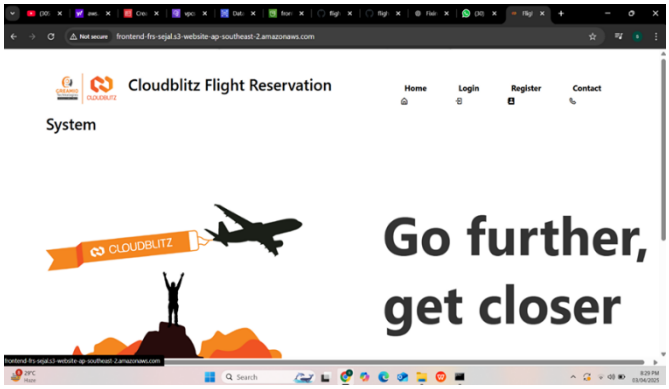


Figure 2. Home Page

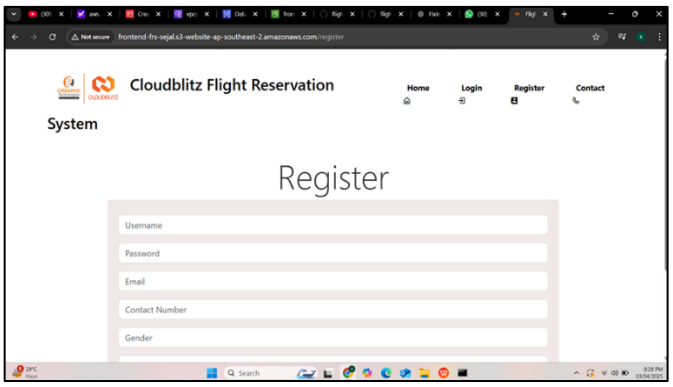


Figure 3. Register Page

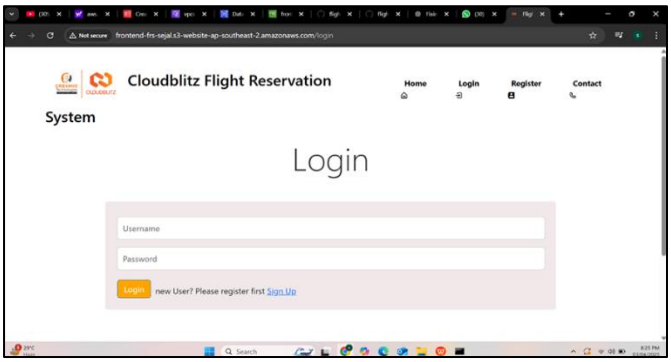


Figure 4. Login Page

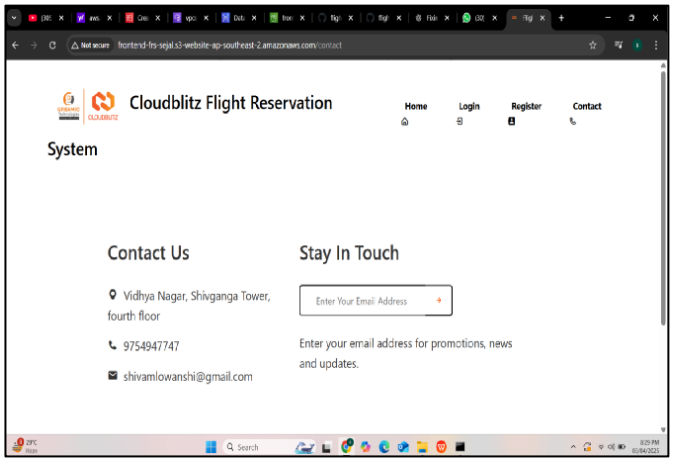


Figure 5. Dashboard

V. CONCLUSION:

In conclusion, the **Pipeline Maestro – Flight Reservation System** is an advanced, user-centric application designed to simplify and enhance the process of flight booking and management. Its features, such as real-time flight data, user-friendly interface, booking flexibility, and payment integration, contribute to a smooth and efficient travel experience for both customers and airlines.

The system's robust back-end architecture ensures scalability, security, and seamless integration with other travel-related services. Additionally, the inclusion of user management, personalized options, and booking history further enhances the overall functionality.

From an operational perspective, the system streamlines airline operations, allowing for better resource management and reduced administrative overhead. Overall, **Pipeline Maestro** stands as a comprehensive, efficient solution for modern-day flight reservations, supporting both user satisfaction and airline operational efficiency.

VI. REFERENCES:

[1] Phillips, M. Sens, A. de Jonge and M. van Holsteijn, The IT Manager's Guide to Continuous Delivery, XebiaLabs, Hilversum, The Netherlands, 2015.

[2] J. Humble and D. Farley, Continuous Delivery: Reliable Software Releases Through Build Test and Deployment Automation, Reading, MA, USA: Addison-Wesley, 2010.

[3] Crosby, T. (5th August, 2015). How Airline E-Tickets Work. from An airline reservation system is a part of so-called Passenger Services Systems (PSS), which are applications supporting the direct contract with the passenger (Crosby, 2015).IEEE

[4] Arik Air, n.d. Viewed on August 4, 2014 from http://www.arikair.com. Airline Reservation Systems. In Wikipedia (2014, August). Retrieved from http://en.wikipedia.org/wiki/Airline_reservations_system

[5] P. Rodríguez et al., "Continuous deployment of software intensive products and services: A systematic mapping study", J. Syst. Softw., vol. 123, pp. 263-291, Jan. 2017.

[6] Shao, Q., Chen, Y., Tao, S., Yan, X. andAnerousis, N. (2008). a ticket routing recommendation engine for enterprise problem resolution. Proceedings of the VLDB Endowment, Vol 1(2). IEEE.

[7] Smith, M. J. (2002). The Airline Encyclopedia. Scarecrow Press, New York.In 1946, American Airlines installed the first automated booking system, the experimental electromechanical Reservation. A newer machine with temporary storage based on a magnetic drum, the Magnetron Reservoirs, soon followed (Smith, 2002). IEEE.