Design And Implement A Bus Ticketing And Payment System Through Any Open Cloud Platform.

NAME: G. Nagarjuna
Reg.no: 192211487
CSE

Introduction to the Bus Ticketing System

The bus ticketing system aims to simplify the process of purchasing tickets for travelers.

It serves as a digital platform that connects passengers with bus operators in real-time.

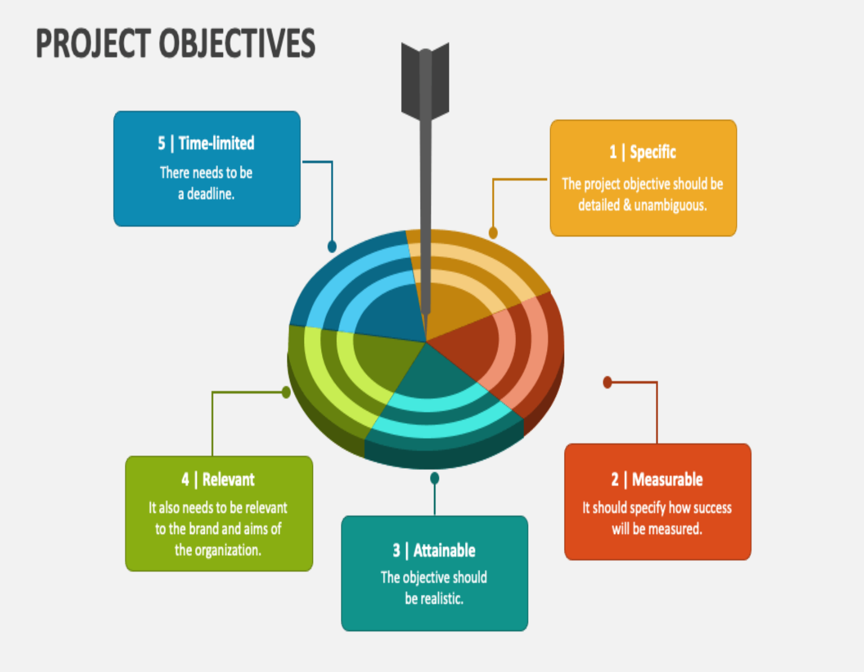
This system enhances customer experience while optimizing operational efficiency.

Objectives of the System

The primary objective is to provide an easy-to-use interface for ticket booking and payment.

It aims to reduce operational costs for bus operators by automating ticket sales.

The system seeks to improve customer satisfaction through better accessibility and convenience.

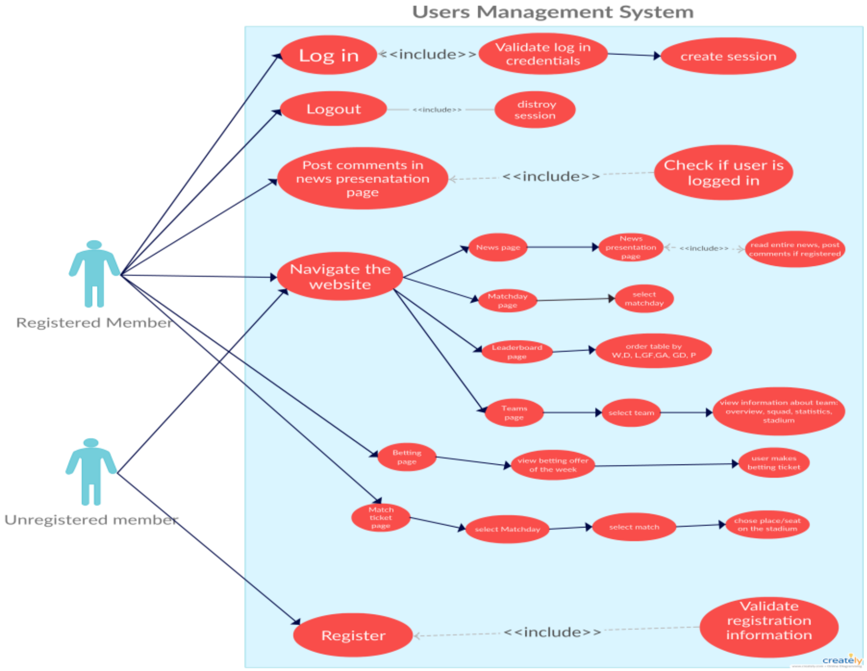

PROJECT OBJECTIVES
5 | Time-limited
There needs to be a deadline.
1 | Specific
The project objective should be detailed & unambiguous.
4 | Relevant
It also needs to be relevant to the brand and aims of the organization.
3 | Attainable
The objective should be realistic.
2 | Measurable
It should specify how success will be measured.

Key Features of the System

Users can search for routes, view schedules, and book tickets seamlessly.

The system includes multiple payment options for user convenience, including credit cards and e-wallets.

It provides real-time notifications about ticket confirmation and bus schedules.



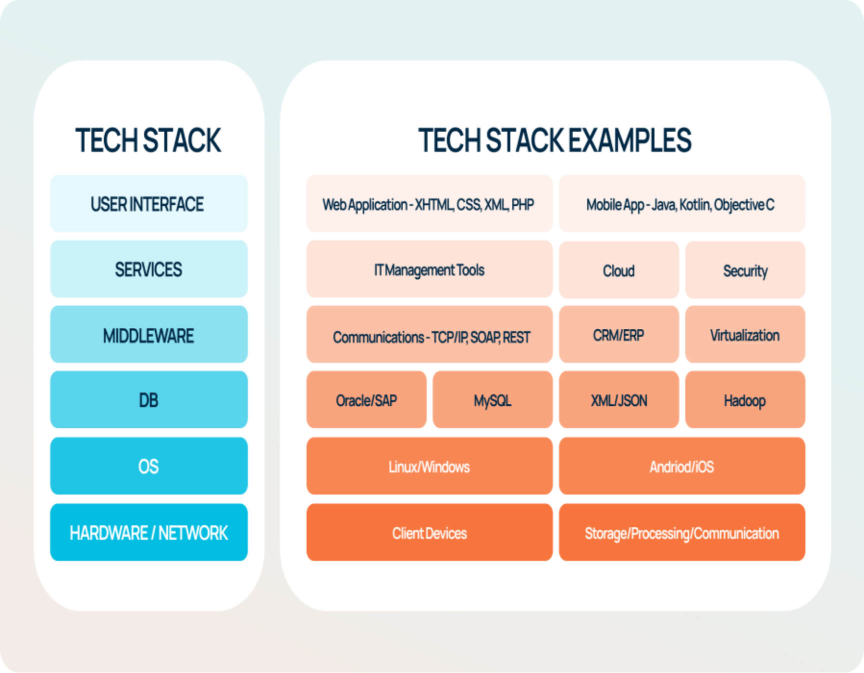
Users Management System
Log in
<<include>>
Validate log in credentials
create session
Logout
<<include>>
distroy session
Post comments in news presenatation page
<<include>>
Check if user is logged in
Navigate the website
News page
News presentation page
<<include>>
read entire news, post comments if registered
Matchday page
select matchday
Leaderboard page
order table by W,D, L,GF,GA, GD, P
Teams page
select team
view information about team: overview, squad, statistics, stadium
Betting page
view betting offer of the week
user makes betting ticket
Match ticket page
select Matchday
select match
chose place/seat on the stadium
Registered Member
Unregistered member
Register
<<include>>
Validate registration information
creately

Technology Stack Overview

The system will be implemented using an open cloud platform such as AWS or Google Cloud.

Frontend technologies may include React or Angular for a responsive user interface.

The backend will utilize Node.js or Python with a database like MongoDB or PostgreSQL.



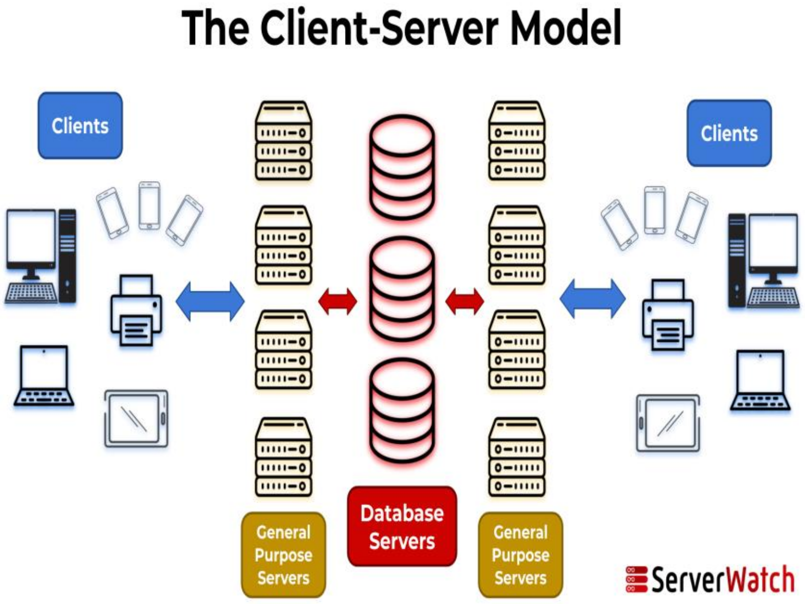
TECH STACK	TECH STACK EXAMPLES			
USER INTERFACE	Web Application - XHTML, CSS, XML, PHP		Mobile App - Java, Kotlin, Objective C	
SERVICES	IT Management Tools		Cloud	Security
MIDDLEWARE	Communications - TCP/IP, SOAP, REST		CRM/ERP	Virtualization
DB	Oracle/SAP	MySQL	XML/JSON	Hadoop
OS	Linux/Windows		Andriod/iOS	
HARDWARE / NETWORK	Client Devices		Storage/Processing/Communication	

System Architecture

The architecture consists of a client-side application, server-side API, and a database.

Each component communicates through RESTful APIs to ensure modularity and scalability.

Cloud services like AWS Lambda can be utilized for serverless functions to enhance performance.

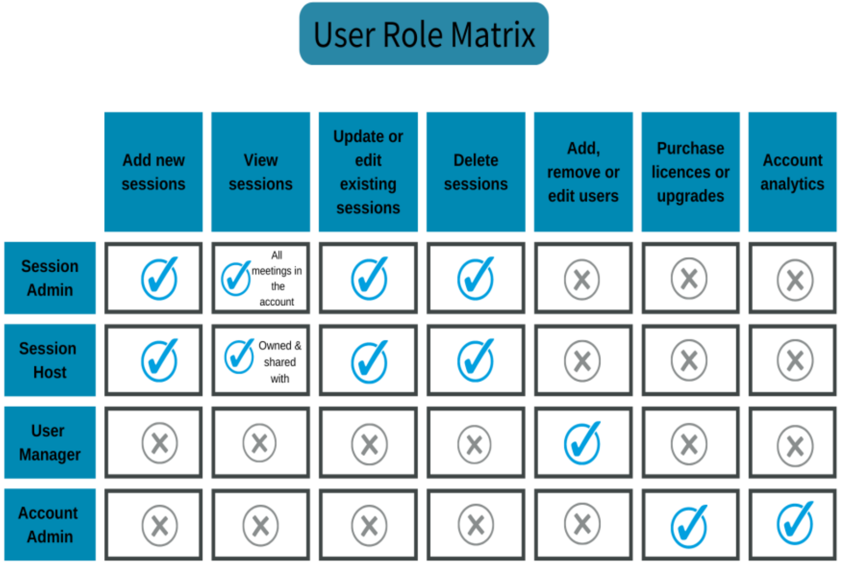


User Roles and Permissions

The system defines several user roles, including passengers, bus operators, and administrators.

Each role has specific permissions that govern access to various functionalities.

This role-based access control helps maintain security and manage user interactions effectively.



User Role Matrix

	Add new sessions	View sessions	Update or edit existing sessions	Delete sessions	Add, remove or edit users	Purchase licences or upgrades	Account analytics
Session Admin	✓	✓ All meetings in the account	✓	✓	✗	✗	✗
Session Host	✓	✓ Owned & shared with	✓	✓	✗	✗	✗
User Manager	✗	✗	✗	✗	✓	✗	✗
Account Admin	✗	✗	✗	✗	✗	✓	✓

Payment Integration

The payment module will integrate with payment gateways like Stripe or PayPal for transactions.

Security protocols including SSL encryption will be implemented to protect user data.

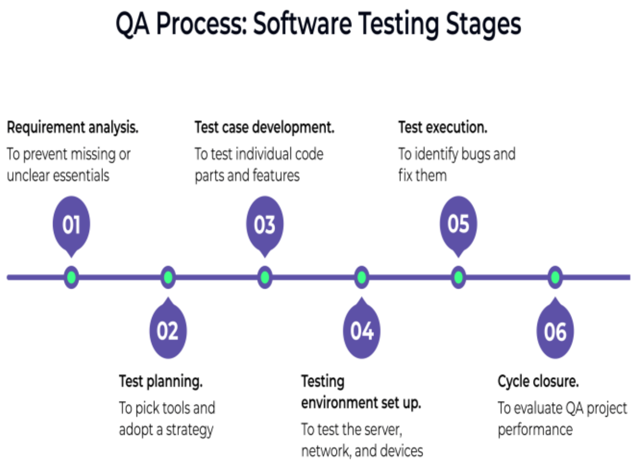
A simple and intuitive checkout process will enhance user experience while minimizing cart abandonment.

Testing and Quality Assurance

The system will undergo multiple testing phases including unit, integration, and user acceptance testing.

Automated testing tools will be used to ensure consistent performance and reliability.

Feedback from beta users will be collected to make necessary adjustments before launch.



QA Process: Software Testing Stages

01 **Requirement analysis.** To prevent missing or unclear essentials

02 **Test planning.** To pick tools and adopt a strategy

03 **Test case development.** To test individual code parts and features

04 **Testing environment set up.** To test the server, network, and devices

05 **Test execution.** To identify bugs and fix them

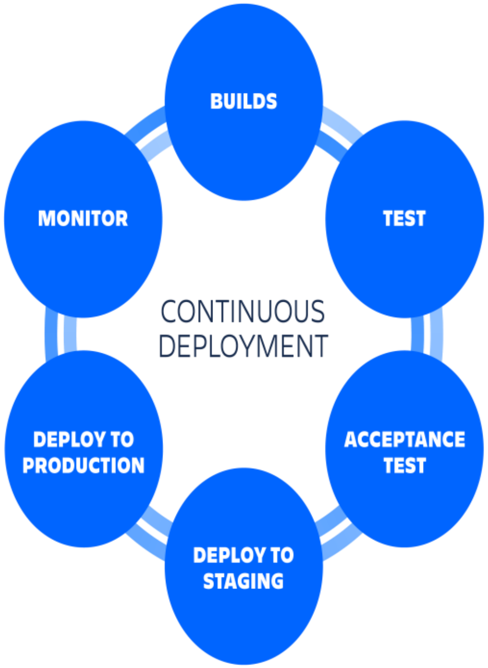
06 **Cycle closure.** To evaluate QA project performance

Deployment Strategy

Continuous integration and continuous deployment (CI/CD) practices will be employed for smooth updates.

The system will be deployed in the cloud for scalability and high availability.

Regular monitoring will be set in place to ensure system health and performance.



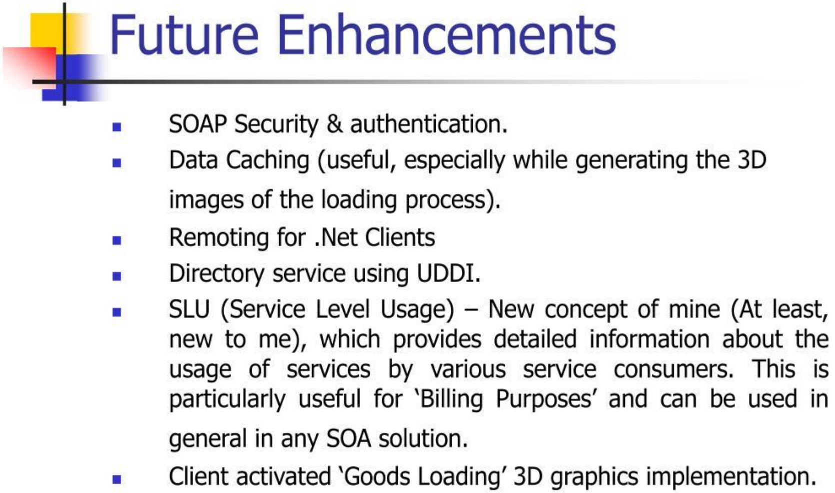
Future Enhancements and Conclusion

Future upgrades may include the addition of a mobile app to expand user accessibility.

Integration with other travel services like hotels and car rentals can enhance the overall travel experience.

The system aims to evolve continuously based on user feedback and technological advancements.

Feel free to customize any of the content or adjust the focus to better suit your specific needs!



Future Enhancements

- SOAP Security & authentication.
- Data Caching (useful, especially while generating the 3D images of the loading process).
- Remoting for .Net Clients
- Directory service using UDDI.
- SLU (Service Level Usage) – New concept of mine (At least, new to me), which provides detailed information about the usage of services by various service consumers. This is particularly useful for 'Billing Purposes' and can be used in general in any SOA solution.
- Client activated 'Goods Loading' 3D graphics implementation.

THANK
Y U!