

DEVOPS PROJECTS HANDBOOK

SKILL-UP / SKILL-BRIDGE EXERCISES

CONTENTS

S No.	Project Name	Page No.
1.	Introduction to DevOps	2
	a) Development Team	
	b) Operations Team	
	c) SecOps Team	
2.	Java Online Book Store Project with DevOps	10
3.	Online Shopping Cart (E-Commerce Web Site) with DevOps	17
4.	Tender Management System with DevOps	24
5.	Train Ticket Reservation System with DevOps	29
6.	Electricity Bill Payment System with DevOps	41
7.	Paytm Mall with DevOps	46
8.	Student Management System with DevOps	51
9.	Automated Web Development Environment	56

Introduction to DevOps

DevOps is a Software engineering methodology that combines software development and operations. It aims to integrate the work of development and operations teams by promoting collaboration and shared responsibility. DevOps is a culture of collaboration between Development and Operations teams that is supported by a variety of practices and tools.

DevOps => Development team and Operations team.

Development team consists of **Developers/Programmers/Software Engineer, QA (QA Manager/QA Team Lead/QA Architect/Tester/Test Automation Engineer/QA Analyst)**

Developer: Someone who creates or builds software or applications. It can refer to who writes code, design user interfaces, or works on various aspects of software development.

Programmer: Someone who writes code and test code to create software programs. This team consists of strong focus on coding and programming languages.

Software Engineer: Have a broader skill set than programmers. They are involved in the entire software development process including design, coding, testing and maintenance. They often have a strong understanding of software architecture and engineering principle.

Computer Scientist: Computer Scientist are more focused on the theoretical and algorithm aspects of computing. They study algorithm data structure and the theoretical foundations of computation. While they may also write code, their primary focus is on the science of computing.

Tester: This role refers those people who test the code for any error at different phases of software lifecycle. They do this work either manually or using some available tools.

QA Manager: Creates and implements company Quality Standards, Investigating Customer Complaints, Determining Training needs, Regulation and Compliance, Problem Solver and Project Manager, Lead Quality Teams, Quality Management Functions, Leadership Experience, Measure Performance and Recommend improvements, Reduce Waste, Perform Statistical analysis of test results, Supervising technical and laboratory staff, Testing and sampling products

QA Team Lead: Test planning, Kick starting projects, Overseeing and leading projects, Reviewing test documentations, Test execution, Reporting, Providing feedback and suggesting improvements, Test closure, Collaborating with other stakeholders, Creating tracking metrics

QA Architect: Provides, functional, technical, process leadership, Designing automation frameworks, Covering scenarios, Reviewing platform architecture and designs, Leading initiatives, Managing QA processes

Tester: Study and analyse the software application, Develop test plans, Design test cases, Setup test environments and data, Run test, Log and report defects, Restes bug fixes, Regression test entire system, Provide test status updates, Recommend improvements, Maintain documentation

Different types of Software Testing:

Functional testing:

- Validates software system requirements and specifications
- Checks that functions, inputs, integrations, and outputs work as expected
- Covers workflows, data manipulations, calculations, security and access levels etc

Helps identify gaps and inconsistencies in behaviour and functionality

Non-functional testing:

Aspects unrelated to specific system functions:

Performance testing: Load levels, peak capacity and system behaviour under high stress

Security testing: Identify vulnerabilities to various cyber-attacks and penetration risks

Usability testing: Human interaction and GUI design elements

Scalability testing: System stability and hardware needs for future growth

Automation testing:

Employs test execution tools and reduces manual work

Automate repetitive tasks like regression test without human intervention

Efficiently cover a large number of test scenarios, configurations, and data sets

Integrate with DevOps processes and continuous testing

Exploratory testing

Unstructured approach to find bugs without written test cases

Explore without predefined steps and find failure through intuition and experience

Particularly effective in exposing hard and find UI, usability, and integration bugs

Complementary to scripted testing techniques

Essential skills for software tester:

Technical skills

Software testing tools and frameworks

Programming

SQL and databases

Operating systems

DevOps methodology

Cloud platforms

Virtualization

Bug tracking tools

Design documentation

Software skills

Communication

Teamwork

Creativity

Analytical thinking

Reporting

Multitasking

Time management

Continuous learning

Test Automation Engineer:

Identifying and selecting the automation test cases

Applying various designs and documenting the automation test strategies

Creating an automation test plan and getting approval

Configuring Selenium Test Environment (STE) to set it up

Participating in Selenium Environment Setup with an IDE

Automating the design of a framework

Implementing it as per the structure of the project

Creating, enhancing, debugging, and running the test cases

Collating and monitoring the defect management process

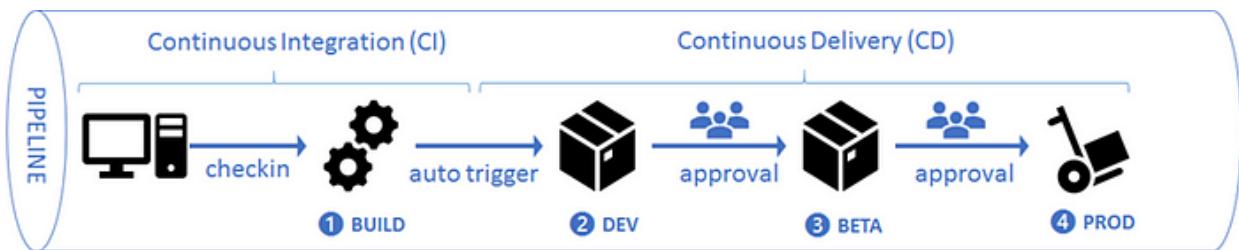
Managing the changes and executing regression tests

Coming up with exact solutions for problem related to object identity and error handing

Interacting with customers/clients to solve the various issues they face and updating on the situation

QA Analyst:

Create and maintain test documentation, Reviewing the adherence to product demands, Attention to detail, Good communication skills, Planning and implementing product testing, Test planning and strategy, Analysing the test results, Compile and analyse statistical data, Draft quality assurance policy and procedures



Common roles in IT Operations (ITOps)

- **Operations Engineer** – manages operations for environment implementation, application development and optimization of ITSM framework best practices.
- **Operations Manager** – oversees the design and implementation of ITOps from a business and technical aspect.
- **Cloud Architect** – designs the cloud environment with respect to the organizations business requirements. Additional responsibilities include Infrastructure deployment, maintenance, monitoring, management and security.
- **Business Architect** – interprets organizational policies, risks and goals and bridges the gap between strategy and execution. Synthesizes the strategy development process across all departments to ensure that the collective vision of all teams are aligned with the organization.
- **System Integrator** – designs, builds, and synthesizes all IT systems subsets used collectively to deliver an IT service or solution.
- **Incident & Security Manager** – develops strategies and system that help reduce risk and maximize dependability of an IT service.
- **Applications Analyst** – is responsible for monitoring, administration, and maintenance of software applications, interacting data and the underlying IT infrastructure.
- **Technical/Systems Analyst** – analyses how IIT systems, solutions and applications fit with the business needs of the organization . Defines and analyses system performance, standards and issues.
- **Networking Administrator** – Configures systems, identifying, and solving network issues; budgeting, maintenance.
- **IT Operators** – deploys, install, and setups devices for users. Responding to service requests by fixing issues, replacing or upgrading equipment.

Operations team

consists of
Compliance, Assist with budgeting and planning, Boosts employee productivity, Build relationships with global teams, Financial auditing and reporting, Improves customer satisfaction, Inventory management, Quality control, Train and guide employees toward success, Budget administration, Ensures product quality, Formulate strategic and operational objectives, Monitoring performance, Strategy, Supply chain management, Team Leadership, Applying risk assessment, Assist with company objectives, Collaborate with Environment, Health & Safety, Data Analysis, Design, Development management, Development of new products, Ensuring safety.

Job/Role Title	Responsibilities	Qualifications
DevOps Engineer	<ul style="list-style-type: none"> • Automating code deployment to various environments. 	<ul style="list-style-type: none"> • Experience with one or more programming languages (e.g.

	<ul style="list-style-type: none"> • Managing the configuration of servers and network devices. • Building and maintaining CI/CD pipelines. • Implementing and maintaining monitoring and logging systems. • Managing and scaling cloud-based infrastructure. • Troubleshooting and resolving technical issues that arise. 	<ul style="list-style-type: none"> Python, Java, JavaScript, Go • Experience with configuration management tools (e.g. Ansible, Puppet, Chef) • Experience with containerization technology (e.g. Docker, Kubernetes) • Experience with cloud platforms (e.g. AWS, Azure, GCP) • Experience with monitoring and logging tools (e.g. Prometheus, ELK Stack)
Test Automation Engineer	<ul style="list-style-type: none"> • Designing and implementing automated testing strategies for software applications, ensuring high-quality software releases • Collaborate with development and QA teams to identify and prioritize areas for automation testing • Perform regression testing and validate bug fixes. • Analyze and report test results and defects. • Continuously improve and maintain the test automation infrastructure. • Work closely with development teams to ensure the timely delivery of high-quality software. • Stay up to date with the latest test automation tools and techniques. 	<ul style="list-style-type: none"> • Experience with test automation tools (Selenium, Appium, Cucumber) and testing methodologies (Agile, Scrum, TDD), • Strong programming skills in at least one language (Java, Python, JavaScript) • Strong analytical and problem-solving skills. • Excellent verbal and written communication skills. • Experience with CI/CD pipelines. • Experience with cloud-based testing and test automation in cloud environments such as AWS or Azure.
Site Reliability Engineer	<ul style="list-style-type: none"> • Ensuring the reliability and availability of a software system. • Identifying and addressing potential performance bottlenecks. • Creating and maintaining 	<ul style="list-style-type: none"> • Strong understanding of Linux/Unix administration. • Experience with automation and configuration management tools. • Experience with monitoring and logging tools.

	<p>processes to prevent and resolve incidents.</p> <ul style="list-style-type: none"> Improving the scalability and performance of the system. Collaborating with development teams to design and implement new features. 	<ul style="list-style-type: none"> Experience with performance tuning and optimization. Experience with one or more programming languages.
Security Engineer	<ul style="list-style-type: none"> Implementing security best practices and tools to protect the company's software and systems from potential threats 	<ul style="list-style-type: none"> Strong knowledge of security protocols and technologies, experience with security testing tools (Nessus, Metasploit, Burp Suite) and compliance standards (PCI-DSS, HIPAA, SOC 2)
Cloud Engineer	<ul style="list-style-type: none"> Designing, building, and maintaining the infrastructure and architecture of cloud-based systems. Implementing security and compliance controls. Managing and scaling cloud-based resources. Automating the deployment of code to the cloud. Troubleshooting and resolving technical issues that arise. 	<ul style="list-style-type: none"> Experience with one or more cloud platforms (e.g. AWS, Azure, GCP) Experience with automation and configuration management tools. Experience with containerization technology. Experience with network and security technologies. Knowledge of programming languages and frameworks is a plus.
Containerization Engineer	<ul style="list-style-type: none"> Building and maintaining containerized environments. Managing and scaling containerized infrastructure. Configuring and troubleshoot container orchestration software. Collaborating with development teams to package and deploy software consistently. 	<ul style="list-style-type: none"> Experience with containerization technology (e.g. Docker, Kubernetes) Experience with container orchestration software (e.g. Kubernetes, ECS, AKS) Experience with cloud-based platforms and automation tools. Knowledge of programming languages and frameworks is a plus.

SecOps the fusion of both the security team and operations team. SecOps into their Security Operations Centres (SOCs).

- Chief InfoSec Officer (CISO)
- Security Manager

- Security Engineer
- Security Analyst
- IT Operations Manager
- System Admin
- System Analyst

Chief InfoSec Officer (CISO)

- Develop the entire security strategy
- Communicate interests and activities to the C-Suite
- Oversee any compliance needs
- Ensure security strategy covers prevention along with detection
- Deeply know and understand the threat landscape

Security Manager

- Oversee the SOC
- Create the vision for hiring strategies, technology, and security processes
- Establish the incident response plan
- Establish a vulnerability management program
- Hire necessary security personnel
- Communicate security and technology needs to the CISO
- Analyse and optimize orchestration and automation

Security Engineer

- **Titles under Security Engineer**
 - Security Architect
 - Security Device Manager
 - SIEM Engineer
- Those who specialize in endpoint security
- Create, implement and monitor all security systems
- Develop orchestration and automation between security tools
- Troubleshoot infrastructure
- Develop solutions to mitigate security vulnerabilities
- Communicate any security incidents with the team and necessary staff
- Report on evaluations and recommendations for improvement

Security Analyst

- Plan preventive measures and procedures
- Create a plan for how to respond to threats
- Establish and implement security measures
- Monitor alerts
- Investigate and respond to any security incidents
- Provide training on information security and network security procedures

IT Operations Manager

- Manager and guide IT Technicians
- Monitor IT systems and servers
- Develop department procedures and policies
- Oversee installations and upgrades
- Negotiate vendor contracts
- Resolve help desk escalations

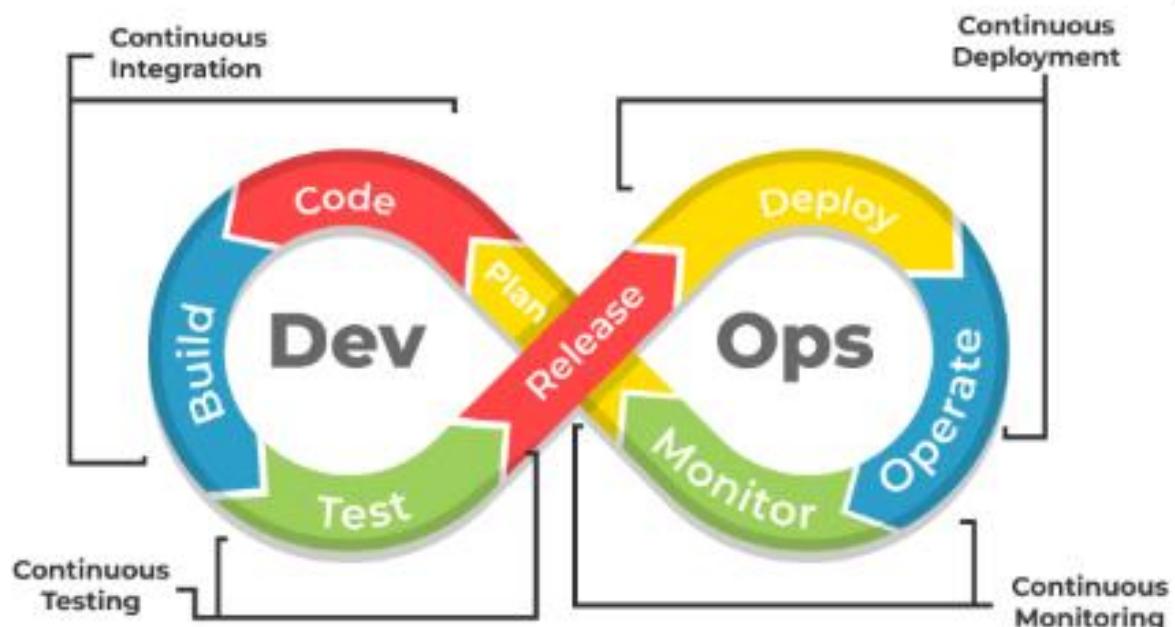
System Admin

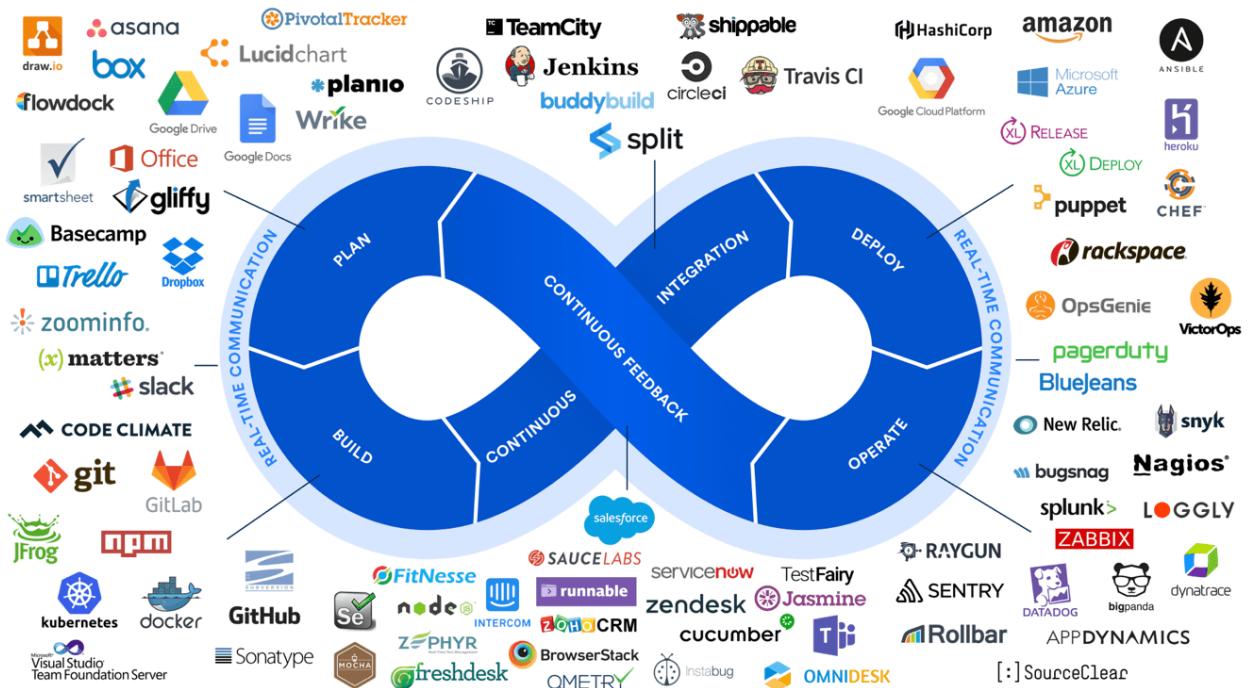
- Install and update hardware and software
- Maintain and configure network servers and computer systems
- Integrate automation procedures and processes

- Run diagnostics and troubleshoot errors
- Lead help desk efforts
- Provide training and documentation to staff regarding new IT infrastructure

System Analyst

- Install, maintain and troubleshoot information and computer systems
- Research innovative technologies and make recommendations for the organization
- Monitor current systems and analyse automation
- Review and backup systems regularly





1. Java Online Book Store Project with DevOps

1.1 Description

A user-friendly Online Bookstore project in which users can log in or register, view the available books, select books along with their quantity, and buy them. Users can also get payment receipts after successful payment. The project can also be used by the administrator, who can add new books, remove books, increase, and decrease the quantity of books, change the price of the books as well as maintain the selling history of books.

The objective of this project is to create a online book store that allows users to search and purchase a book online based on title, author, and subject. The selected books are displayed in tabular format and the user can order their books online through credit card payment. Using this application users can purchase a book online instead of going out to a book store and wasting time.

There are many online books stores like Books Wagon, indiabookstore.net, 99booksstore.com, 99booksstore.com, therightbookstoreindia.com, flipkart.com, bookbazaar.com, bookshop.org, oxfordbookstore.com, snapdeal.com, booksliced.com, Abebooks, alibris.com, bookoutlet.com, kidsbook.com, Walmart.com, booktopia.com.au, thriftbooks.com, booksamillion.com, barnesandnobel.com, betterworldbooks.com, biblio.com, bookmooch.com, target.com, Costco.com, Daedalusbooks.com, ebay.com, strandbooks.com, hudsonbooksellers.com, Powell's, Amazon etc which were designed using web technologies. We are developing this web application using Web Technologies, Java, Testing and DevOps tools.

Online Book Store is an online web application where the customer can purchase books online. Through the web browser the customer can search for a book by title or author, later can add to the shopping cart and finally purchase using credit card transaction. The user can login using his account details or new customers can setup an account very quickly. They should give the details of their name, contact number and shipping address. The user can also give feedback to a book by giving ratings on a score of 5. The books are divided into many categories base on subject like Software, Database, English, Architecture etc.

1.2 The project has following functionalities

I. A Home page with product catalogue

This is the page where the user will be navigated after a successful login. It will display all the book categories and will have a search keyword option to search for the required book. It also includes some special sections like recommended titles, weekly special books.

II. Search

A search by keyword option is provided to the user using a textbox. The keyword to be entered should be the book title.

III. Advanced Search

Advanced search helps the user to search for a book based on Title, Author, Category and Price range. All the books which match the particular search criteria and their total count will be displayed. From here the user can select a book and add to the shopping cart.

IV. Book Description

If the user would like to know details about a book he can click on the title from where he will be directed to a Book description page. It includes the notes on the book content and also a link to Amazon.com to get the book review.

V. User Voting

The user can give rating to a book based on his interest. He can rate it by giving a score of five as Excellent, four for very good, three for good, two for regular and one for deficient. The final rating of book will depend on all the individual user rating.

VI. Shopping Cart

The user can manage a shopping cart which will include all the books he selected. The user can edit, delete and update his shopping cart. A final shopping cart summary is displayed which includes all the items the user selected and the final total cost.

VII. Managing User Accounts

Each user should have an account to access all the functionalities of website. User can login using login page and logout using logout page. All the user sessions will be saved in the database.

VIII. Administration

The Administrator will be provided with the special functionalities like:

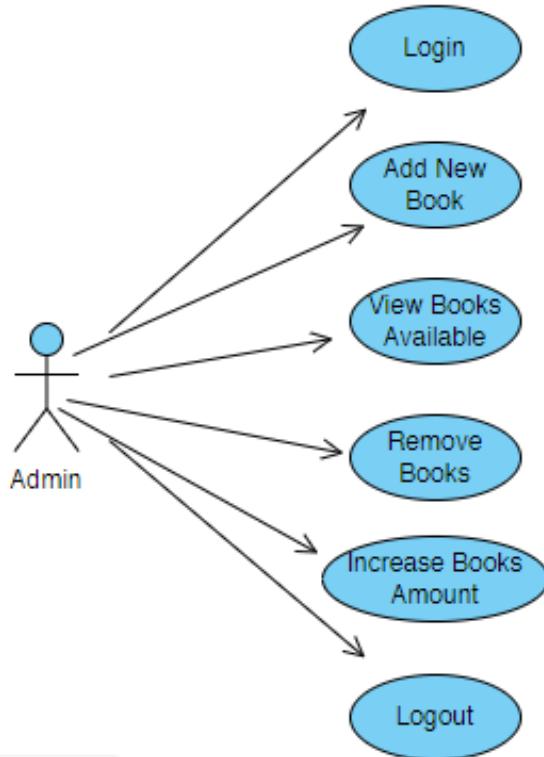
- Add or delete a book category
- Add or delete a member
- Manage member order
- Add or delete a Credit Card Type

1.3 This Web Project is built for following purpose

- For Selling books online.
- Maintaining books selling history.
- Adding and managing books.
- User Friendly.
- For Implementation of Http Servlets in Java.
- This is a Project developed using Java, JDBC, Servlets and JSP.

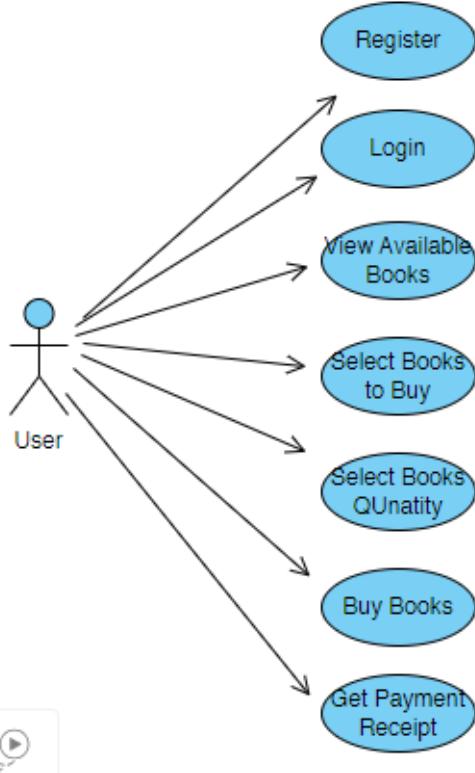
1.4 Admin Have Following Access for this online store site

- Add New Books.
- View Books Available.
- Remove Books.
- Increase Books Amount.



1.5 User Have Following Access for this online store site

- Create New Account or Register.
- Login.
- View Available Books.
- Select Books to Buy.
- Select Books Quantity.
- Buy Books.
- Get Payment Receipt.



1.6 Softwares & Technologies used in the Project

Front End	
HTML	Layout & Form Designing
CSS	Page Styling
Bootstrap	Responsive Web Designing
jQuery	AJAX Calling & DOM Manipulation
Angular/ReactJS	Single Page Application calling web application
Middleware	
JDBC	Java to Database Connectivity
Servlets	Server-side form processing, Session management & Client state persistence, Security management, Page Includes, Page Routing, Page-Page communication
JSP	Dynamic web page generation
Java Mail API	E-Mail integration
Java Tools	Log4J, SLF4J, ANT, Maven, JUnit, HttpUnit, Selenium
Database	
Oracle/MySQL/SQLite /PostgreSQL/HSQldb	Rational Database Management system to perform manipulation & query operations on tables
DevOps Tools	
Git	Version Control System
ANT/Maven	Build Tool
Jenkins	Continuous Integration
Junit/HttpUnit	Unit Testing
Selenium	Web Page Testing
SonarQube	Code Analysis
JIRA	Issue Tracking
Jfrog Artifactory	Artifact Repository

Docker	Containerization
Ansible	Deployment & Configuration Management
Kubernetes	Container Management
Prometheus	Monitoring
Grafana	Alerting
Nagios	Application Monitoring & Solution
Containers	Tomcat/Apache HTTP Server XAAMP/LAMP/WAMP
Slack	Team Collaboration
Twilio	SMS/WhatsAPI/Chat/MMS Integration

1.7 Database SQL

create database if not exists onlinebookstore;

use onlinebookstore;

create table if not exists books (barcode varchar(100) primary key, name varchar(100), author varchar(100), price int, quantity int);

create table if not exists users(username varchar(100) primary key,password varchar(100), firstname varchar(100), lastname varchar(100), address text, phone varchar(100), mailid varchar(100), usertype int);

insert into books values ('9780134190563','The Go Programming Language','Alan A. A. Donovan and Brian W. Kernighan',400,8);

insert into books values ('9780133053036','C++ Primer','Stanley Lippman and Josée Lajoie and Barbara Moo',976,13);

insert into books values ('9781718500457','The Rust Programming Language','Steve Klabnik and Carol Nichols',560,12);

insert into books values ('9781491910740','Head First Java','Kathy Sierra and Bert Bates and Trisha Gee',754,23);

insert into books values ('9781492056300','Fluent Python','Luciano Ramalho',1014,5);

insert into books values ('9781720043997','The Road to Learn React','Robin Wieruch',239,18);

insert into books values ('9780132350884','Clean Code: A Handbook of Agile Software Craftsmanship','Robert C Martin',288,3);

insert into books values ('9780132181273','Domain-Driven Design','Eric Evans',560,28);

insert into books values ('9781951204006','A Programmers Guide to Computer Science','William Springer',188,4);

insert into books values ('9780316204552','The Soul of a New Machine','Tracy Kidder',293,30);

insert into books values ('9780132778046','Effective Java','Joshua Bloch',368,21);

insert into books values ('9781484255995','Practical Rust Projects','Shing Lyu',257,15);

insert into users values ('demo','demo','Demo','User','Demo Home','42502216225','demo@gmail.com',2);

insert into users values ('Admin','Admin','Mr.','Admin','Haldia WB','9584552224521','admin@gmail.com',1);

insert into users values ('shashi','shashi','Shashi','Raj','Bihar','1236547089','shashi@gmail.com',2);

commit;

1.8 Screenshots

Welcome to Online Book Store

[Home](#)
[Login](#)
[Books Available](#)
[Logout](#)
[Login or Signup Below](#)
[Login As Admin](#)
[Login As User](#)
[New user ! Register Here](#)

Home Available Books Cart About Us Logout

Available Books

 Test_Book Author: Shashi Raj Id: 1123 Trending Price: ₹ 678.0 - 1 + Id: 9780133053036 Only 7 items left	 C++ Primer Author: Stanley Lippman and Josée Lajoie and Barbara Moo Id: 9780133053036 Only 7 items left Price: ₹ 976.0 - 3 + Id: 9781491910740 Only 17 items left	 Head First Java Author: Kathy Sierra and Bert Bates and Trisha Gee Id: 9781491910740 Only 17 items left Price: ₹ 754.0 Add To Cart
 Domain-Driven Design Author: Eric Evans Id: 9780132181273 Only 8 items left Price: ₹ 560.0 Add To Cart	 The Go Programming Language Author: Alan A. A. Donovan and Brian W. Kernighan Id: 9780134190563 Only 0 items left Price: ₹ 400.0 Out Of Stock	 Fluent Python Author: Luciano Ramalho Id: 9781492056300 Only 2 items left Price: ₹ 1014.0 Add To Cart

Home Available Books Cart About Us Logout

Shopping Cart

Bookid	Name	Author	Price/Item	Quantity	Amount
1123	Test_Book	Shashi Raj	₹ 678.0	- 1 +	₹ 678.0
9780133053036	C++ Primer	Stanley Lippman and Josée Lajoie and Barbara Moo	₹ 976.0	- 3 +	₹ 2928.0
9780316204552	The Soul of a New Machine	Tracy Kidder	₹ 293.0	- 1 +	₹ 293.0

Total Amount To Pay ₹ 3899.0

Proceed to Pay ₹ 3899.0

Cart Checkout

Billing Address

Full Name
John M. Doe

Email
john@example.com

Address
542 W. 15th Street

City
New York

Payment

Accepted Cards

Name on Card
John More Doe

Credit card number
1111-2222-3333-4444

Exp Month
September

Exp Year
2018

CVV
352

Email
john@example.com

Address
542 W. 15th Street

City
New York

State
NY

Zip
10001

Shipping address same as billing

Name on Card
John More Doe

Credit card number
1111-2222-3333-4444

Exp Month
September

Exp Year
2018

CVV
352

Total Amount **₹ 3899.0**

Pay & Place Order

Your Orders



Test_Book
Author: Shashi Raj

Order Id: ORD1123TM
Item Yet to be Delivered

Amout Paid: ₹ 678.0

Order Placed



C++ Primer
Author: Stanley Lippman and Josée Lajoie and Barbara Moo

Order Id: ORD9780133053036TM
Item Yet to be Delivered

Amout Paid: ₹ 976.0

Order Placed



The Soul of a New Machine
Author: Tracy Kidder

Order Id: ORD9780316204552TM
Item Yet to be Delivered

Amout Paid: ₹ 293.0

Order Placed

2. Online Shopping Cart (E-Commerce Web Site) with DevOps

2.1 Description

This E-Commerce Website build for selling of any electronics products online.

In this project a user can visit the websites, registers, and login to the website. They can check all the products available for shopping, filter and search item based on different categories, and then add to cart. They can add multiple items to the cart and also plus or minus the quantity in the cart. Once the cart is updated, the user can proceed to checkout and click the credit card payment details to proceed. Once the payment is success the orders will be placed and users will be able to see the orders details in the orders section along with the shipping status of the product.

The admin also plays an important role for this project as the admin is the one responsible for adding any product to the store, updating the items, removing the item from the store as well as managing the inventory. The admin can see all the product orders placed and also can mark them as shipped or delivered based on the conditions.

One of the best functionalities that the projects include is mailing the customers, so once a user registers to the website, they will receive a mail for the successful registration to the website, and along with that whenever a user orders any product or the product got shipped from the store, then the user will also receive the email for its confirmation. Sometimes, if the user tried to add any item which is out of stock, them they will get an email one the item is available again the stock.

Note: The payment page is created only for demo purpose and its not fully integrated with any payment gateway. So for now any credit card details will be accepted and the demo orders will be placed.

We are developing this web application using Web Technologies, Java, Testing and DevOps tools.

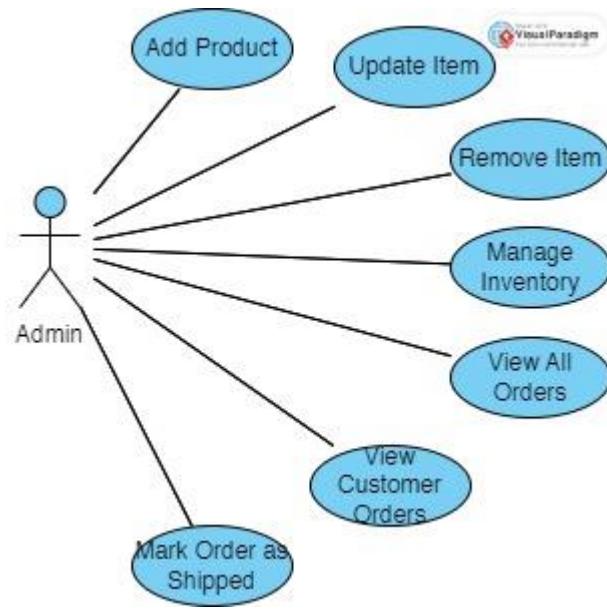
2.2 Highlights

The users will get a mail to their registered Mail ID during:

- New User Registration
- Order Successfully Placed
- The Item was out of stock while exploring but now it got available in the store
- Successful shipment and delivery of the Item

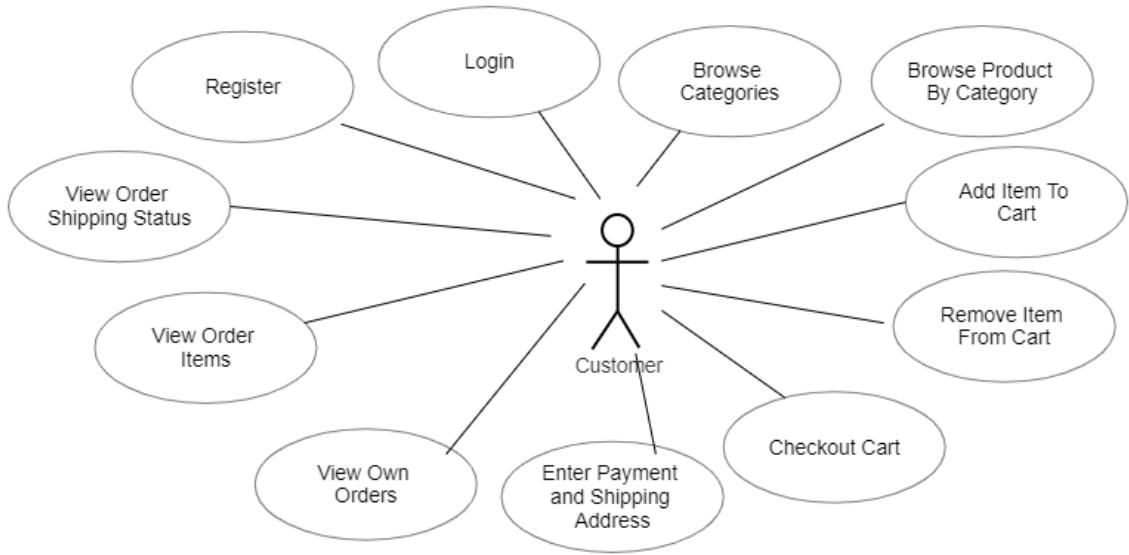
Admin Have Following Access for this online store site:

- Adding product to store
- Updating the items
- Removing the items
- Managing Inventory
- View All the orders
- View All Customer Orders
- Mark Order as Shipped/Delivered



Users Have Following Access for this online store site:

- Create New Account or Register.
- Login.
- Browse Categories
- Browse Product by Category
- Add Item to Cart
- Remove Item from Cart
- Checkout Cart
- Enter Payment and Shopping Address
- View Own Orders
- View Order Items
- View Order Shipping Status



2.3 Softwares & Technologies used in the Project

Front End	
HTML	Layout & Form Designing
CSS	Page Styling
Bootstrap	Responsive Web Designing
jQuery	AJAX Calling & DOM Manipulation
Angular/ReactJS	Single Page Application calling web application
Middleware	
JDBC	Java to Database Connectivity
Servlets	Server-side form processing, Session management & Client state persistence, Security management, Page Includes, Page Routing, Page-Page communication
JSP	Dynamic web page generation
Java Mail API	E-Mail integration
Java Tools	Log4J, SLF4J, ANT, Maven, JUnit, HttpUnit, Selenium
Database	
Oracle/MySQL/SQLite /PostgreSQL/HSQLDB	Rational Database Management system to perform manipulation & query operations on tables
DevOps Tools	
Git	Version Control System
ANT/Maven	Build Tool
Jenkins	Continuous Integration
Junit/HttpUnit	Unit Testing
Selenium	Web Page Testing
SonarQube	Code Analysis
JIRA	Issue Tracking
Jfrog Artifactory	Artifact Repository
Docker	Containerization
Ansible	Deployment & Configuration Management
Kubernetes	Container Management
Prometheus	Monitoring

Grafana	Alerting
Nagios	Application Monitoring & Solution
Containers	Tomcat/Apache HTTP Server XAAMP/LAMP/WAMP
Slack	Team Collaboration
Twilio	SMS/WhatsAPI/Chat/MMS Integration

2.4 Screenshots of the project

The screenshot shows the homepage of the Ellison Electronics website. The header features the site's name, a search bar, and navigation links for Login, Register, Products, and Category. Below the header, a green banner displays the slogan "We specialize in Electronics". The main content area shows a grid of three product cards:

- APPLE iPhone 13 Pro (Graphite, 512 GB)**: Price Rs 125999.0. Buttons: Add to Cart, Buy Now.
- HP Intel Core i5 11th Gen**: Price Rs 40990.0. Buttons: Add to Cart, Buy Now.
- LED Smart Google TV**: Price Rs 41999.0. Buttons: Add to Cart, Buy Now.

Below the grid, there are smaller thumbnail images of the same products.

The screenshot shows a registration form titled "Registration Form" on the Ellison Electronics website. The form includes fields for Name, Email, Address, Mobile, Pin Code, Password, and Confirm Password. At the bottom are "Reset" and "Register" buttons.

Shopping Center

Products Category Orders Profile Logout

Ellison Electronics
We specialize in Electronics

Search Items

Showing Results for 'laptop'

	HP Intel Core i5 11th Gen Hp Laptop (8 GB/512 GB SSD/Windows 11 Home) 15s-fr4000TU Thin and Light Laptop (15.6 inch, Natural.. Rs 40990.0		ASUS Vivobook 15 Core i3 11th Gen ASUS Vivobook 15 Core i3 11th Gen - (8 GB/512 GB SSD/Windows 11 Home) X515EA-EJ322WS X515EA-EJ328W.. Rs 50999.0		Acer Extensa Core i5 11th Gen 8 GB/512 GB SSD/Windows 11 Home EX 215-54-583M Thin and Light Laptop (15.6 inch, Charcoal Black,.. Rs 57999.0
Remove From Cart	Checkout	Add to Cart	Buy Now	Add to Cart	Buy Now

localhost:8080/shopping-cart/userHome.jsp?type=laptop

- Cart Items

Cart Details

localhost:8080/shopping-cart/cartDetails.jsp

Shopping Center

Products Category Orders Profile Logout

Ellison Electronics
We specialize in Electronics

Search Items

Cart Items

Picture	Products	Price	Quantity	Add	Remove	Amount
	realme NEO 80 cm (32 inch) HD Ready LED Smart Linux TV (RMV2101)	11999.0	2	Update	-	23998.0
	APPLE iPhone 13 Pro (Graphite, 512 GB)	125999.0	1	Update	-	125999.0
	HP Intel Core i5 11th Gen	40990.0	1	Update	-	40990.0
	ASUS Vivobook 15 Core i3 11th Gen	50999.0	1	Update	-	50999.0
Total Amount to Pay (in Rupees)						241986.0
Cancel						Pay Now

- Credit Card Payment

Credit Card Payment

Name of Card Holder
Enter Card Holder Name.

Enter Credit Card Number
4242-4242-4242-4242

Expiry Month
MM

Expiry Year
YYYY

Enter CVV
123

Pay : Rs 241986.0

- Order Details & Status

Order Placed Successfully!

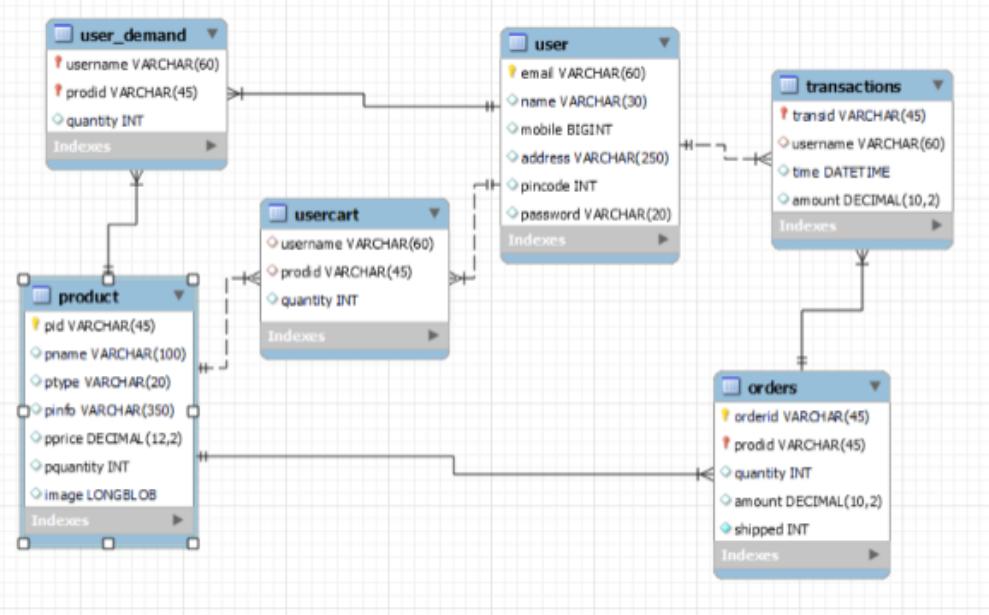
Picture	ProductName	OrderId	Quantity	Price	Time	Status
	REDMI Pad 6	T20230501055718	1	18999.00	2023-05-01 17:57:19.0	ORDER_PLACED
	APPLE iPhone 13 Pro (Graphite, 512 GB)	T20230520083929	1	125999.00	2023-05-20 20:39:30.0	ORDER_PLACED
	HP Intel Core i5 11th Gen	T20230520083929	1	40990.00	2023-05-20 20:39:30.0	ORDER_PLACED
	realme NEO 80 cm (32 inch) HD Ready LED Smart Linux TV (RMV2101)	T20230520083929	2	23998.00	2023-05-20 20:39:30.0	ORDER_PLACED
	ASUS Vivobook 15 Core i3 11th Gen	T20230520083929	1	50999.00	2023-05-20 20:39:30.0	ORDER_PLACED
	APPLE iPhone 13 Pro (Graphite, 512 GB)	TR10001	1	125999.00	2023-04-23 20:39:30.0	ORDER_SHIPPED

2.5 Class Diagram

☰ README.md

Come Shop Again!
Ellison Electronics.

- Class Diagram



3. Tender Management System with DevOps

3.1 Description

Whenever a company requires a service / merchandise, a tender is floated. This project developed to enable the vendors to get all the tender details online and provide facility to submit the tender. Company maintains an empanelled list of Vendors. An empanelled vendor can only bid for a tender. Every vendor can bid only once against each tender. Against each tender there may be bids from several vendors. The company will then select the most suitable bid and places the order to that vendor. We are developing this web application using Web Technologies, Java, Testing and DevOps tools.

3.2 Features

This provides a system which will reduce the extra cost which involves for this entire process like:

- Advertising
- Document processing
- Getting confirmation results.

Vendors can save a lot of time. This application makes the evaluation process to complete within the assigned date.

Different vendors with provides different Bids among which some selected bids can be approved

3.3 Existing System

- This Tender Management System works need to be completed manually.
- Contractors need to submit their documents by visiting the office or centers on time.
- Contractors has to bid for particular tender on time by applying forms and needs to wait for approval reply.
- The Admin Needs to post the Projects or contracts as a poster or advertisements regularly.
- It takes long time to access tenders or vendors data accessing.
- Not much secure.

3.4 Proposed System

- User Needs to Have a Network Connection.
- New User needs to register as new vendor in the website.
- Contractors can login and bid for some existing tenders.
- They need to login using their login-id and password which they used while registration
- To reduce paper work, data which are available online, using this final decision can be taken by the administration and can contact directly to such suppliers.
- More data security.
- Easy to access the tenders and vendors data.

3.5 There are 2 users in the system

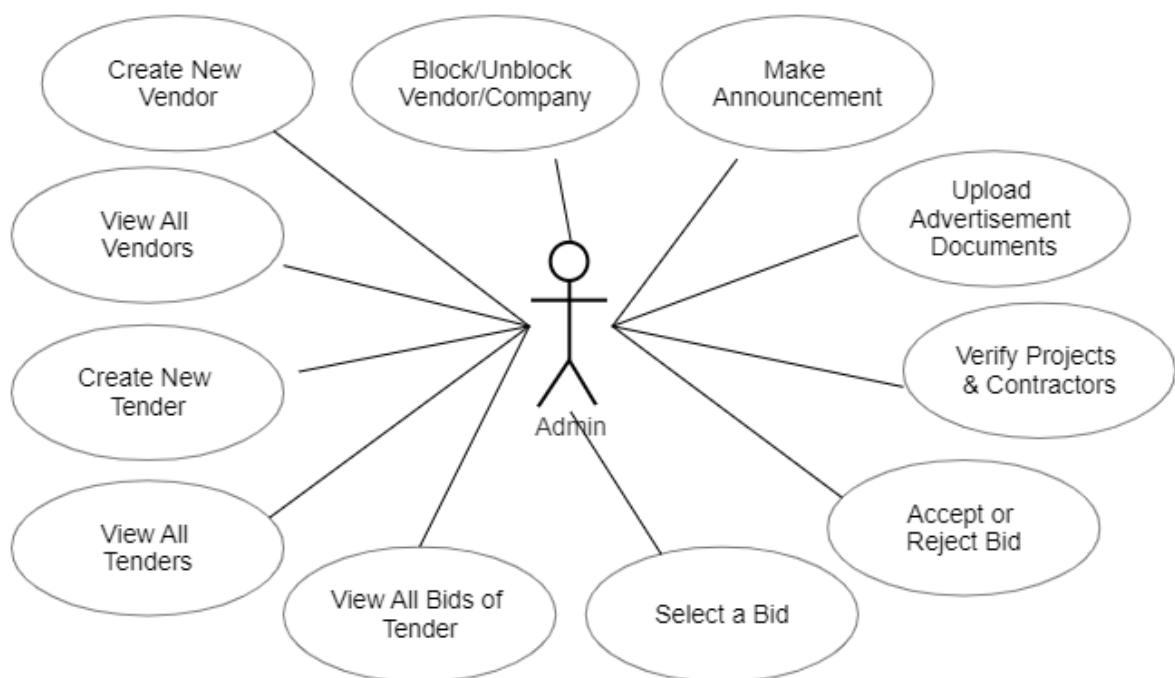
1. Administrator and
2. Vendor

3.6 The Role of Administrator

1. Create new Vendor.
2. View all the vendors.
3. Create new tenders.
4. View All the Tenders.
5. View All the Bids of a tender.
6. Select a Bid.

3.7 Responsible of Admin Includes

1. Uploading advertisement documents
2. Verifying projects and contractors by using their information
3. Accepting and Rejecting Bids
4. Maintaining data and confirmation of final report.
5. Update Notice Board
6. Make some announcement
7. Block or unblock some vendors or company



3.8 Company

1. The Company needs to get registered to proceed.
2. Company can log in with a valid username and password.
3. Apply or releases new tenders and view the status of the tender.
4. Make Some announcement.
5. Change the tender details
6. Documentations

3.9 The Role of a Vendor

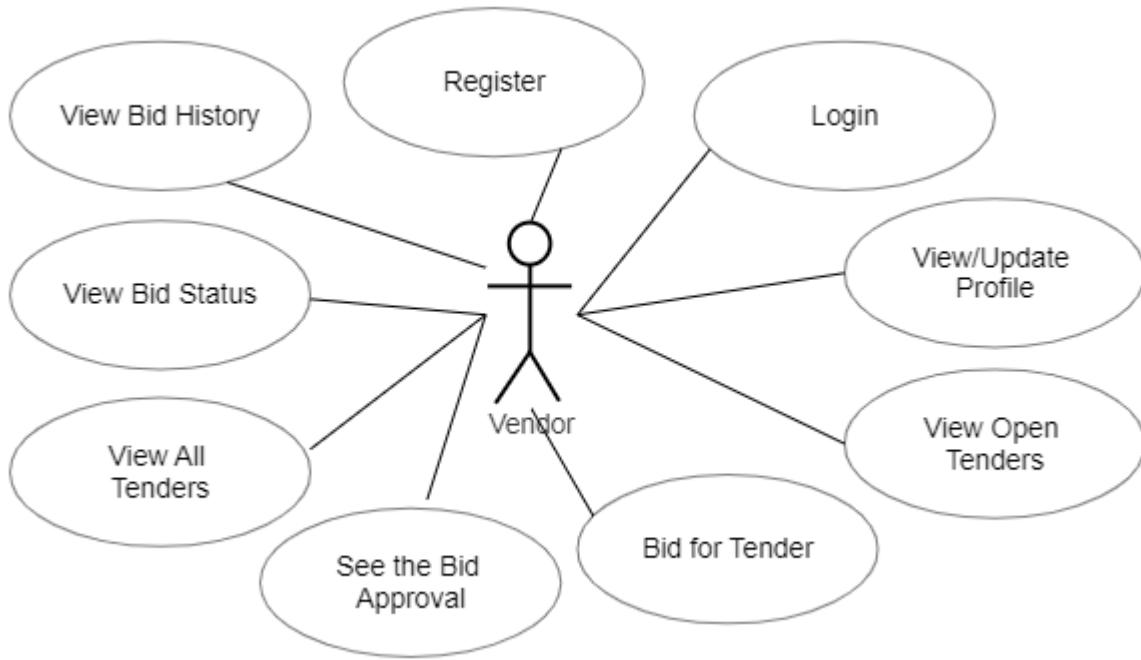
1. View all the current Tenders.
2. Place a Bid against a Tender.
3. View status of a Bid(Whether Selected or Not)
4. View his own Bid History.

3.10 Responsibility of Vendor includes

Vendors are the key persons or contractors who looks for new upcoming tenders on the website

Vendors Functionality include:

- New Vendors can register
- Login using vendor-id and password
- View and update their profile
- See the new open tenders and their details
- Bid for a tender with an amount greater than its base price
- See the bid approval or denial status



After the tender status approval they can view and submit the documentation

Vendors can look for the important notice on the notice section

3.11 Softwares & Technologies used in the Project

Front End	
HTML	Layout & Form Designing
CSS	Page Styling
Bootstrap	Responsive Web Designing
jQuery	AJAX Calling & DOM Manipulation
Angular/ReactJS	Single Page Application calling web application
Middleware	
JDBC	Java to Database Connectivity
Servlets	Server-side form processing, Session management & Client state persistence, Security management, Page Includes, Page Routing, Page-Page communication
JSP	Dynamic web page generation
Java Mail API	E-Mail integration
Java Tools	Log4J, SLF4J, ANT, Maven, JUnit, HttpUnit, Selenium
Database	
Oracle/MySQL/SQLite /PostgreSQL/HSQLDB	Rational Database Management system to perform manipulation & query operations on tables
DevOps Tools	
Git	Version Control System
ANT/Maven	Build Tool
Jenkins	Continuous Integration
Junit/HttpUnit	Unit Testing
Selenium	Web Page Testing
SonarQube	Code Analysis
JIRA	Issue Tracking
Jfrog Artifactory	Artifact Repository

Docker	Containerization
Ansible	Deployment & Configuration Management
Kubernetes	Container Management
Prometheus	Monitoring
Grafana	Alerting
Nagios	Application Monitoring & Solution
Containers	Tomcat/Apache HTTP Server XAAMP/LAMP/WAMP
Slack	Team Collaboration
Twilio	SMS/WhatsAPI/Chat/MMS Integration

3.12 Tables

Table: Notice				
Field Name	Data Type	Size	Is Null ?	Key
id	Int	3	No	Primary Key
title	Varchar	35	Yes	
Info	varchar	500	Yes	

Table: TenderStatus				
Field Name	Data Type	Size	Is Null ?	Key
Tender-id	Varchar	15	No	Primary Key
Bidder-id	Varchar	15	Yes	
Status	Varchar	15	No	
Vendor-id	Varchar	15	Yes	

Table: Tender				
Field Name	Data Type	Size	Is Null ?	Key
Tid	Varchar	15	No	Primary Key
tname	Varchar	40	Yes	
Ttype	Varchar	20	Yes	
Tprice	Int	11	Yes	
Tdesc	Varchar	300	Yes	
Tdeadline	Date		Yes	
Tloc	Varchar	70	Yes	

Table: Vendor				
Field Name	Data Type	Size	Is Null ?	Key
Vid	Varchar	15	No	Primary Key
Password	Varchar	20	Yes	
Vname	Varchar	30	Yes	
Vmob	Int	12	Yes	
Vemail	Varchar	40	Yes	
Company	Varchar	20	Yes	
Address	Varchar	100	Yes	

4. Train Ticket Reservation System with DevOps

4.1 Description

This project is about the Train-Ticket-Reservation-System which is used to view Train Schedule, search trains, Seat availability, Train timings. We can also enquire about fare of different trains. We can get information about train between two stations. We can book seats online. This provides a safe and secure seat reservation system.

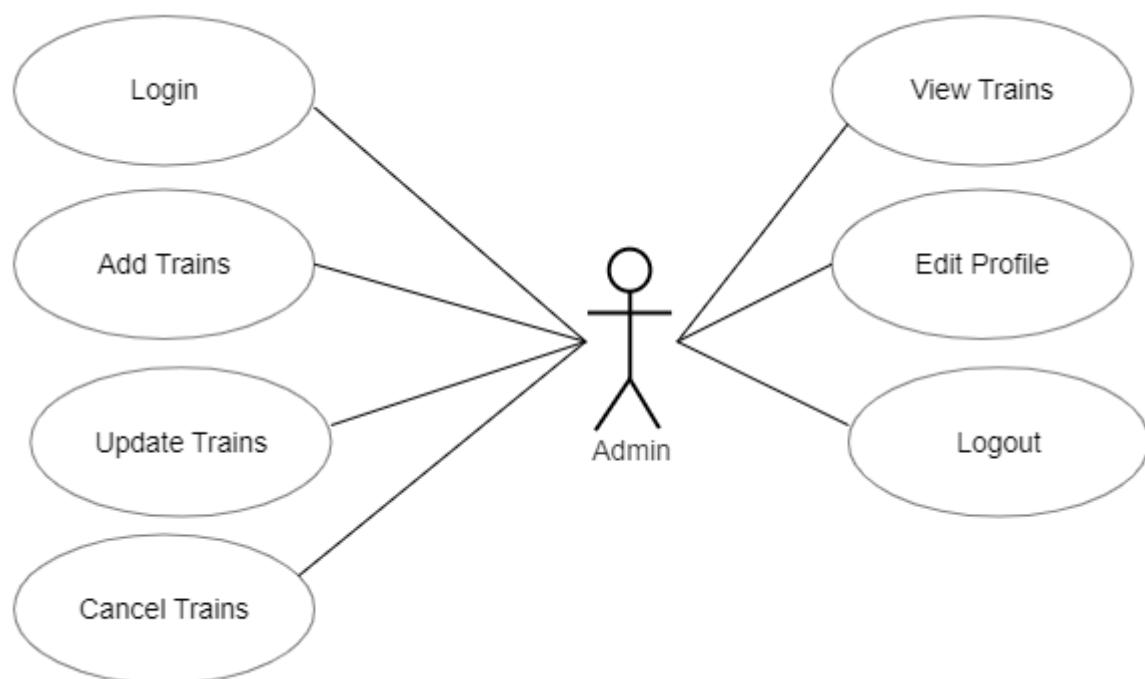
4.2 Online Train Information and Reservation

This Website is built for following purpose: -

- View Trains Schedule
- Search Trains
- Seat Availability
- Train Timings
- Fare Enquiry
- Trains Between Stations
- Booking seats online.
- Login and Logout Security
- Password Changes
- Payment Gateway
- Ticket Booking History

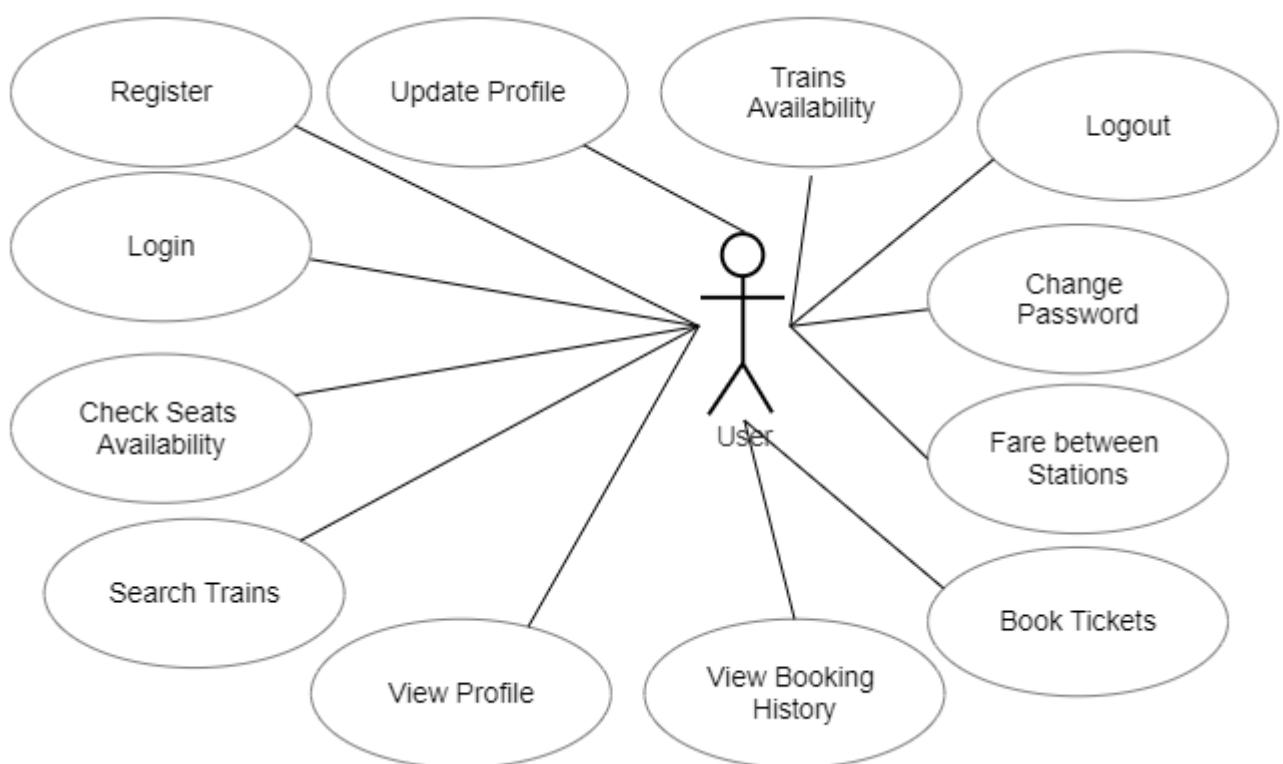
The Admin have the following access to this website: -

- Login
- Add Trains
- Update Trains
- Remove or cancel Trains
- View Trains
- Profile Edit
- Logout



The Users have the following Access :-

- Register
- Login
- View Trains
- Check Seat Availability
- Search Trains
- Train Availability and Fare Between Stations
- Books Tickets
- View Booking History
- View Profile
- Update Profile
- Change Password
- Logout



4.3 Database Initialization

STEP 1: Open SQL Plus OR SQL Developer

STEP 2: Login and connect to database using administrator username and password

STEP 3 :Execute the below command first to create a new user:

```
ALTER SESSION SET "_ORACLE_SCRIPT"=TRUE;
```

```
CREATE USER RESERVATION IDENTIFIED BY MANAGER;
```

```
GRANT DBA TO RESERVATION;
```

```
COMMIT;
```

NOTE: If the above command fails for alter session issues, try to remove first line and then execute it.

STEP 4: Now execute the below sql query in same terminal

```
CONNECT RESERVATION/MANAGER;
```

```
CREATE TABLE "RESERVATION"."CUSTOMER" (
    "MAILID" VARCHAR2(40) PRIMARY KEY,
    "PWORD" VARCHAR2(20) NOT NULL,
    "FNAME" VARCHAR2(20) NOT NULL,
    "LNAME" VARCHAR2(20),
    "ADDR" VARCHAR2(100),
    "PHNO" NUMBER(12) NOT NULL
);
```

```
CREATE TABLE "RESERVATION"."ADMIN" (
    "MAILID" VARCHAR2(40) PRIMARY KEY,
    "PWORD" VARCHAR2(20) NOT NULL,
    "FNAME" VARCHAR2(20) NOT NULL,
    "LNAME" VARCHAR2(20),
    "ADDR" VARCHAR2(100),
    "PHNO" NUMBER(12) NOT NULL
);
```

```
CREATE TABLE "RESERVATION"."TRAIN" (
    "TR_NO" NUMBER(10) PRIMARY KEY,
    "TR_NAME" VARCHAR2(70) NOT NULL,
    "FROM_STN" VARCHAR2(20) NOT NULL,
    "TO_STN" VARCHAR2(20) NOT NULL,
    "SEATS" NUMBER(4) NOT NULL,
    "FARE" NUMBER(6,2) NOT NULL
);
```

```
CREATE TABLE "RESERVATION"."HISTORY" (
    "TRANSID" VARCHAR2(36) PRIMARY KEY,
    "MAILID" VARCHAR2(40) REFERENCES "RESERVATION"."CUSTOMER"(MAILID),
    "TR_NO" NUMBER(10),
    "DATE" DATE,
    "FROM_STN" VARCHAR2(20) NOT NULL,
    "TO_STN" VARCHAR2(20) NOT NULL,
    "SEATS" NUMBER(3) NOT NULL,
    "AMOUNT" NUMBER(8,2) NOT NULL
);
```

```
COMMIT;
```

```
INSERT INTO RESERVATION.ADMIN VALUES ('admin@demo.com','admin','System','Admin','Demo Address  
123 colony','9874561230');
```

```
INSERT INTO RESERVATION.CUSTOMER VALUES ('shashi@demo.com','shashi','Shashi','Raj','Kolkata, West Bengal',954745222);
```

```
INSERT INTO RESERVATION.TRAIN VALUES (10001,'JODHPUR EXP','HOWRAH','JODHPUR', 152, 490.50);
```

```
INSERT INTO RESERVATION.TRAIN VALUES (10002,'YAMUNA EXP','GAYA','DELHI', 52, 550.50);
```

```
INSERT INTO RESERVATION.TRAIN VALUES (10003,'NILANCHAL EXP','GAYA','HOWRAH', 92, 451);
```

```
INSERT INTO RESERVATION.TRAIN VALUES (10004,'JAN SATABDI EXP','RANCHI','PATNA', 182, 550);
```

```
INSERT INTO RESERVATION.TRAIN VALUES (10005,'GANGE EXP','MUMBAI','KERALA', 12, 945);
```

```
INSERT INTO RESERVATION.TRAIN VALUES (10006,'GARIB RATH EXP','PATNA','DELHI', 1, 1450.75);
```

```
INSERT INTO RESERVATION.HISTORY VALUES ('BBC374-NSDF-4673','shashi@demo.com',10001,TO_DATE('02-FEB-2024'), 'HOWRAH', 'JODHPUR', 2, 981);
```

```
INSERT INTO RESERVATION.HISTORY VALUES ('BBC375-NSDF-4675','shashi@demo.com',10004,TO_DATE('12-JAN-2024'), 'RANCHI', 'PATNA', 1, 550);
```

```
INSERT INTO RESERVATION.HISTORY VALUES ('BBC373-NSDF-4674','shashi@demo.com',10006,TO_DATE('22-JULY-2024'), 'PATNA', 'DELHI', 3, 4352.25);
```

```
COMMIT;
```

STEP 5: Now Execute the below query one by one to check if the tables are created successfully

```
SELECT * FROM ADMIN;
```

```
SELECT * FROM CUSTOMER;
```

```
SELECT * FROM TRAIN;
```

```
SELECT * FROM HISTORY;
```

4.4 Importing and Running the Project Through Eclipse EE

Step 0: Open Eclipse Enterprise Edition. Install if not available

Step 1: Click On File > Import > Git > Projects From Git > Clone Uri > Paste The Repository

Url: <https://github.com/shashirajraja/Train-Ticket-Reservation-System.git> > Next > Select Master Branch > Next > Finish

Step 2.A: Right Click on Project > Run as > Maven Build > In the goals field enter "clean install" > apply > run

Step 2.B: Right Click On Project > Build Path > Configure Build Path > Libraries > Remove And Update Any Libraries With Red Mark > Finish

Step 3: [Only if Tomcat v8.0 is not Configured in Eclipse]: Right Click On Project > Run As > Run On Server > Select Tomcat v8.0 > (Select Tomcat V8.0 Installation Location If Asked) Next > Add > Finish

Step 4: In The Server Tab > Double Click On Tomcat Server > Ports > Change The Port Number For Http/1.1 To 8083 > Close And Save

Step 5: Right Click On Project > Run As > Run On Server > Select Tomcat V8.0 > Next > Add All > Done

Step 6: Check Running The Site At <http://localhost:8083/trainbook/>

Step 7: Default Username And Password For Admin Is "admin@demo.com" And "admin"

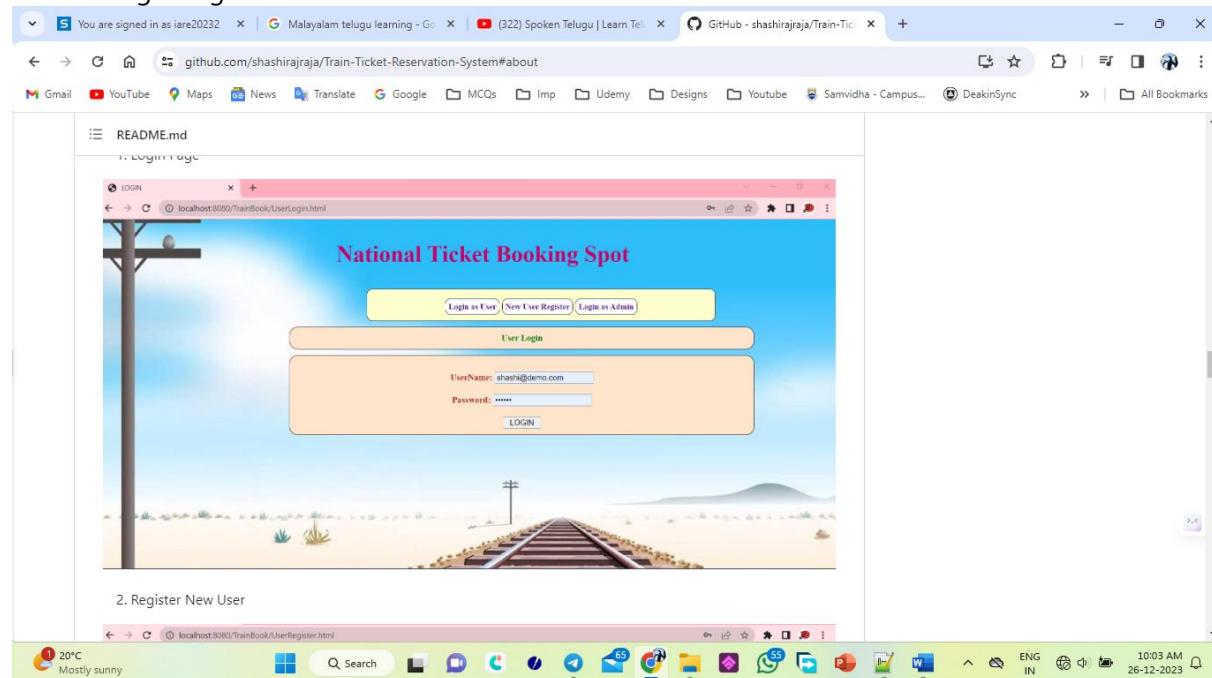
Step 8: Default Username And Password For User Is "shashi@demo.com" And "shashi"

4.5 Importing and Running the Project Through Eclipse EE

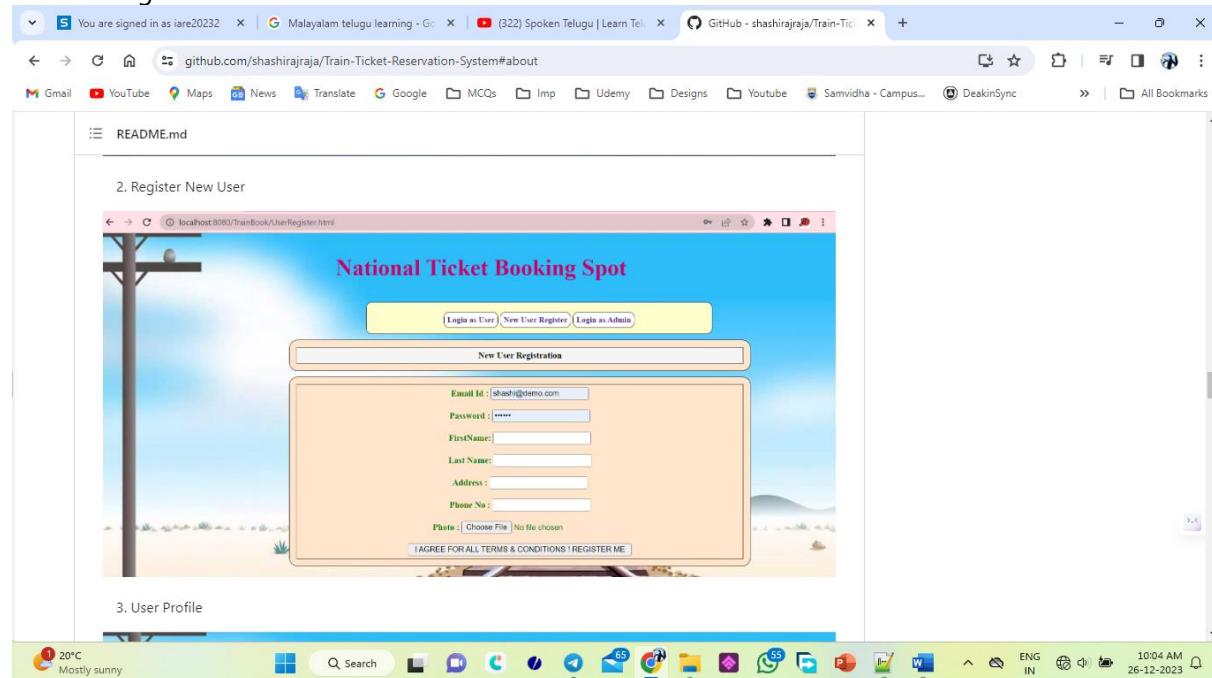
Front End	
HTML	Layout & Form Designing
CSS	Page Styling
Bootstrap	Responsive Web Designing
jQuery	AJAX Calling & DOM Manipulation
Angular/ReactJS	Single Page Application calling web application
Middleware	
JDBC	Java to Database Connectivity
Servlets	Server-side form processing, Session management & Client state persistence, Security management, Page Includes, Page Routing, Page-Page communication
JSP	Dynamic web page generation
Java Mail API	E-Mail integration
Java Tools	Log4J, SLF4J, ANT, Maven, JUnit, HttpUnit, Selenium
Database	
Oracle/MySQL/SQLite /PostgreSQL/HSQLDB	Rational Database Management system to perform manipulation & query operations on tables
DevOps Tools	
Git	Version Control System
ANT/Maven	Build Tool
Jenkins	Continuous Integration
Junit/HttpUnit	Unit Testing
Selenium	Web Page Testing
SonarQube	Code Analysis
JIRA	Issue Tracking
Jfrog Artifactory	Artifact Repository
Docker	Containerization
Ansible	Deployment & Configuration Management
Kubernetes	Container Management
Prometheus	Monitoring
Grafana	Alerting
Nagios	Application Monitoring & Solution
Containers	Tomcat/Apache HTTP Server XAAMP/LAMP/WAMP
Slack	Team Collaboration
Twilio	SMS/WhatsAPI/Chat/MMS Integration

4.6 Screenshots of some of the Web Pages

1. Login Page



2. Register New User



3. User Profile

The screenshot shows a web browser window with the URL github.com/shashirajraja/Train-Ticket-Reservation-System#about. The main content area displays the 'National Ticket Booking Spot' website. On the left, there is a vertical sidebar with yellow buttons labeled: Home, View Trains, Trains Between Stations, Ticket Booking History, Fare Enquiry, Seat Availability, Search By TrainNo, Profile, and Logout. The main content area has a blue header bar with the text 'Hello Shashi! Welcome to our new NITRIC Website'. Below this is a yellow navigation bar with buttons for View Profile, Edit Profile, and Change Password. The central part of the page is titled 'Profile Update' and contains a form with fields for User Name, First Name, Last Name, Address, and Phone No. The phone number field contains '954745224'. At the bottom of the form is a blue 'Update Profile' button. The background of the page features a landscape with railway tracks.

4. Search Trains Between Stations

The screenshot shows a web browser window with the URL github.com/shashirajraja/Train-Ticket-Reservation-System#about. The main content area displays the 'National Ticket Booking Spot' website. On the left, there is a vertical sidebar with yellow buttons labeled: Home, View Trains, Trains Between Stations, Ticket Booking History, Fare Enquiry, Seat Availability, Search By TrainNo, Profile, and Logout. The main content area has a blue header bar with the text 'Trains Between Stations'. Below this is a search form with input fields for 'From Station' and 'To Station', and a blue 'SEARCH TRAIN' button. The background of the page features a landscape with railway tracks.

5. View Trains

5. View Trains

The screenshot shows a web browser window with multiple tabs open. The active tab displays the 'View Trains' section of the system. The page has a header 'National Ticket Booking Spot' and a sidebar with yellow buttons for various functions like Home, View Train, Train Between Station, etc. The main content area shows a table of running trains:

Train Name	Train Number	From Station	To Station	Time	Seats Available	Fare (INR)	Booking
JODHPUR EXP	10007	GAYA	GAYA	19:05	75	559.5 RS	Book Now
JODHPUR EXP	10001	HWH	JODHPUR	17:01	171	491.0 RS	Book Now
NILANCHAL EXP	10003	GAYA	HOWRAH	22:22	91	451.0 RS	Book Now
JAN SATADHI EXP	10004	RANCHI	PATNA	00:54	180	550.0 RS	Book Now
GANGE EXP	10005	MUMBAI	KERALA	05:09	12	945.0 RS	Book Now
GARIB RATH EXP	10006	PATNA	DELHI	08:01	1	1459.75 RS	Book Now
YAMUNA EXP	10002	GAYA	DELHI	12:45	51	550.0 RS	Book Now

7. Book Trains

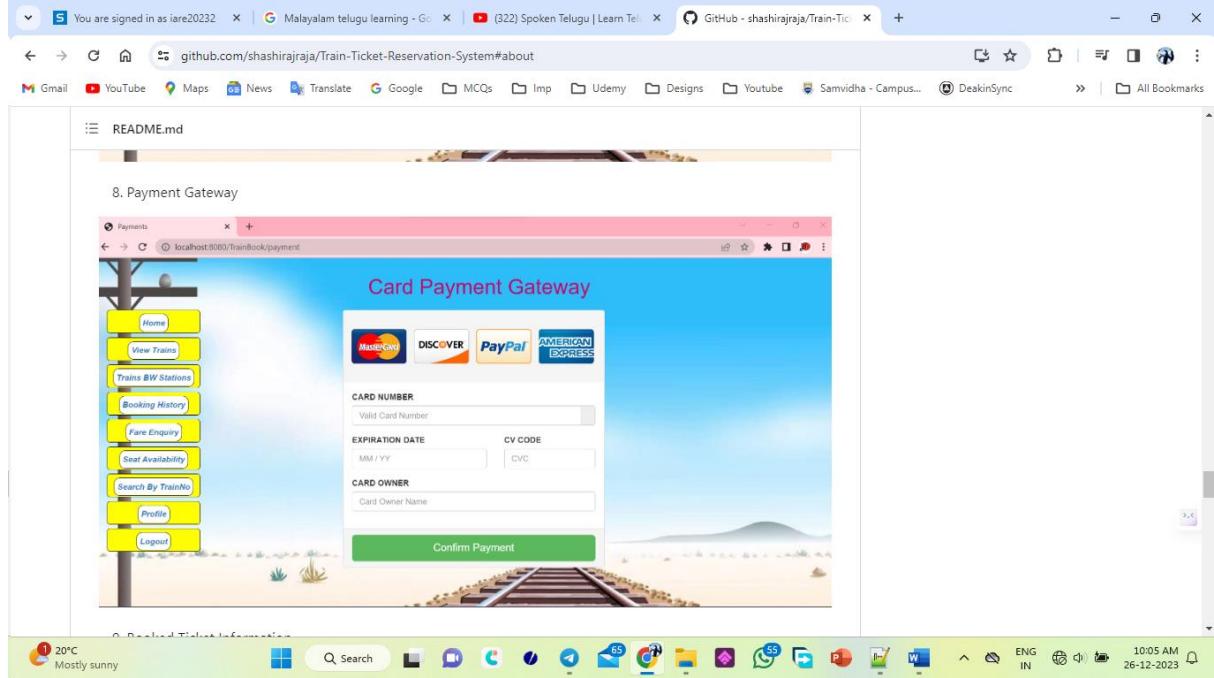
The screenshot shows the 'Book Trains' section of the system. The page has a header 'National Ticket Booking Spot' and a sidebar with yellow buttons. The main content area shows a form for booking a ticket:

Your Ticket Booking Information			
USER ID:	shashi@demo.com	Train NO:	10001
From Station:	HWH	To Station :	JODHPUR
Journey Date:	15-04-2023	No of Seats:	1
Select Class:	Shopper(SL)	Berth Preference:	No Preference

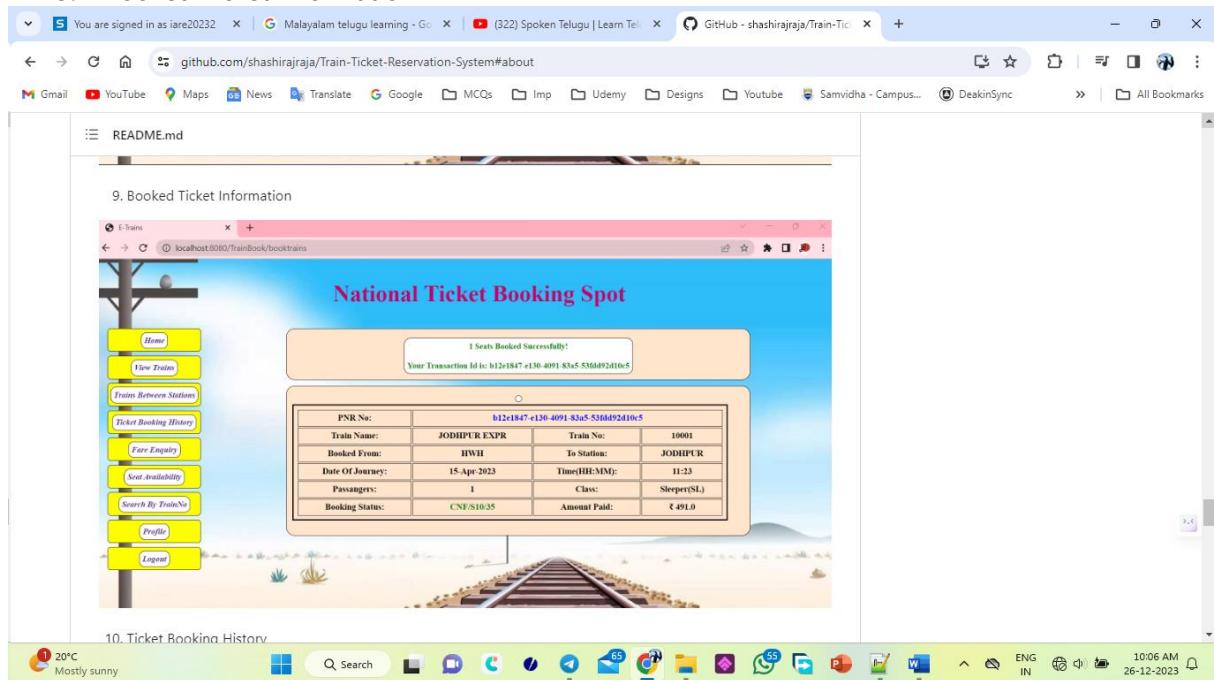
Below the form is a large image of a railway track receding into the distance under a blue sky.

8. Payment Gateway

8. Payment Gateway



9. Booked Ticket Information



10. Ticket Booking History

The screenshot shows a web browser window with the URL github.com/shashirajraja/Train-Ticket-Reservation-System#about. The main content area displays a table titled "Booked Ticket History" with the following data:

Transaction ID	Train Number	From Station	To Station	Journey Date	Seat	Amount Paid
0a153de-16ba-4508-94cc-c2f0a12c41	10001	BHOWRAH	JODHPUR	2023-04-11 00:00:00	3	1470.0
6d21885ca-1961-410c-806b-3e125d55c5b	10001	BWH	JODHPUR	2023-04-08 00:00:00	1	491.0
45D55ea-b3f4-4341-b730-4899593e3016	10001	BWH	JODHPUR	2023-04-08 00:00:00	1	491.0
BBC374-NSDF-4673	10001	BHOWRAH	JODHPUR	2024-02-02 00:00:00	2	981.0
BBC375-NSDF-4675	10004	RANCHI	PATNA	2024-01-12 00:00:00	1	981.0
BBC373-NSDF-4674	10006	PATNA	DELHI	2024-07-22 00:00:00	3	981.0
67e854c-e745-4150-a80d-724c14b25245	10001	BHOWRAH	JODHPUR	2023-04-10 00:00:00	1	490.0

11. Fare Enquiry

The screenshot shows a web browser window with the URL github.com/shashirajraja/Train-Ticket-Reservation-System#about. The main content area displays a form titled "Fare Enquiry for Trains Between Stations" with the following fields:

From Station: Gaya
To Station: Dant
Get Fare

12. Change Password

The screenshot shows a browser window with the URL `localhost:8083/trainbook/changeuserpassword`. The page title is "National Information Railway Tourism Control Hyderabad | NITRTC". A sidebar on the left contains links like Home, View Trains, Search Trains, Book Trains, Availability, Train Route Suggestion, Fare Enquiry, Profile, and Logout. The main content area has a "Hello shashi! Welcome to our new NITRTC Website" message and three buttons: view Profile, Edit Profile, and Change Password. Below these are fields for User Name (shashi), Old Password (redacted), New Password (redacted), and a "Change Password" button.

13. Add Trains By Admin

The screenshot shows a browser window with the URL `localhost:8083/trainbook/admintrainview`. The page title is "National Information Railway Tourism Control Hyderabad | NITRTC". The sidebar on the left includes Admin Home, Search Trains, View Trains, Add Train, Cancel Train, Update Schedule, and Logout. The main content area features a "Add Train" button and a form with fields: Train Number (10104), Train Name (Garni Rath Express), From Station (Pune), To Station (Mumbai), Available (06), and Fare (INR) (999). A "ADD TRAIN" button is at the bottom right of the form.

5. Electricity Bill Payment System with DevOps

5.1 Description

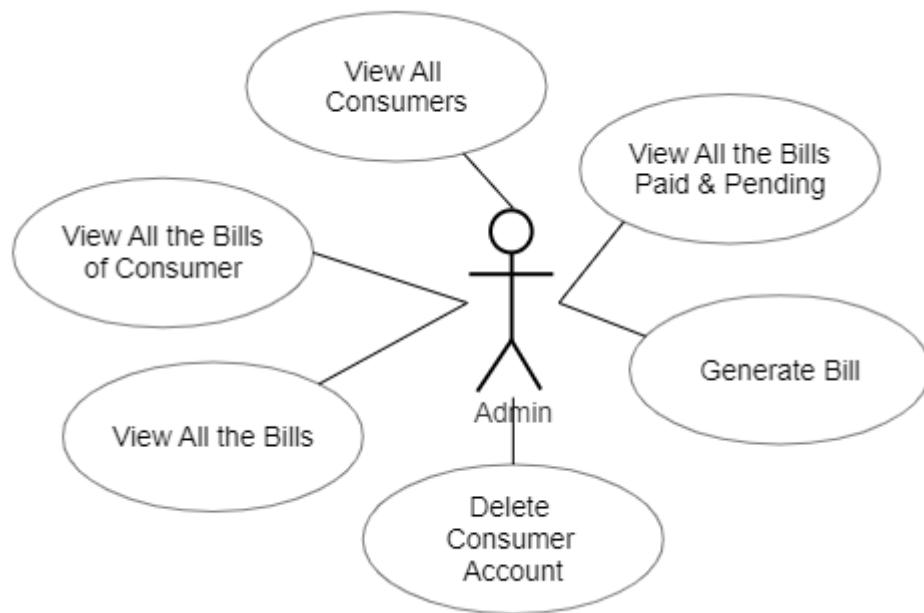
It's a console based App in which user can apply for Electricity connections and Admin can add consumers that have been applied for new connection.

5.2 There are 2 users in this system

- Admin
- User

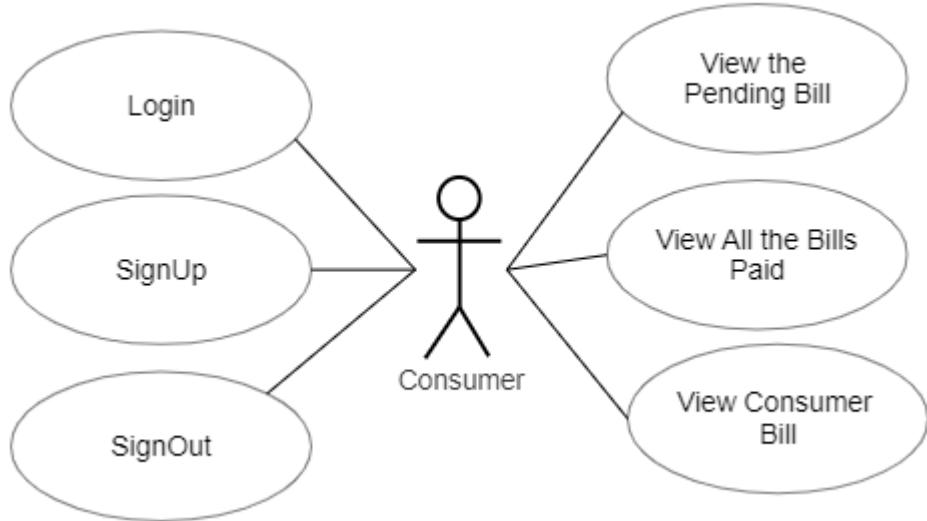
The Admin have the following access to this website: -

- View All Consumers
- View All the Bill of Consumer
- View All the Bills
- Delete Consumer Account
- Generate Bill
- View All the Bills Paid & Pending



The Consumer have the following access to this website: -

- User Login
- User SignUp
- User SignOut
- View Consumer Bill
- View All the Bills Paid
- View the Pending Bill

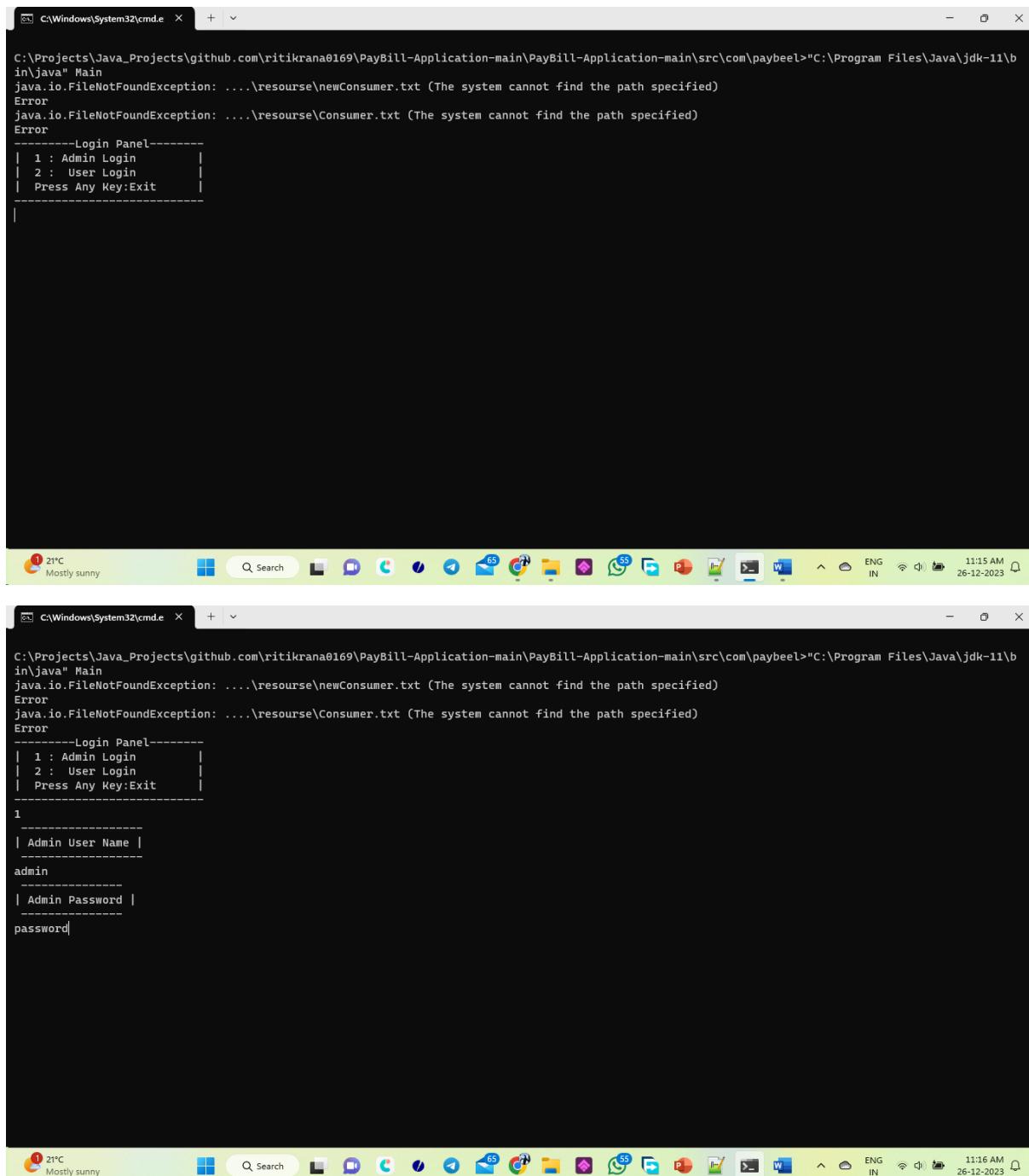


5.2 Softwares & Technologies used in the Project

Front End	
HTML	Layout & Form Designing
CSS	Page Styling
Bootstrap	Responsive Web Designing
jQuery	AJAX Calling & DOM Manipulation
Angular/ReactJS	Single Page Application calling web application
Middleware	
JDBC	Java to Database Connectivity
Servlets	Server-side form processing, Session management & Client state persistence, Security management, Page Includes, Page Routing, Page-Page communication
JSP	Dynamic web page generation
Java Mail API	E-Mail integration
Java Tools	Log4J, SLF4J, ANT, Maven, JUnit, HttpUnit, Selenium
Database	
Oracle/MySQL/SQLite /PostgreSQL/HSQLDB	Rational Database Management system to perform manipulation & query operations on tables
DevOps Tools	
Git	Version Control System
ANT/Maven	Build Tool
Jenkins	Continuous Integration
Junit/HttpUnit	Unit Testing
Selenium	Web Page Testing
SonarQube	Code Analysis
JIRA	Issue Tracking
Jfrog Artifactory	Artifact Repository
Docker	Containerization
Ansible	Deployment & Configuration Management
Kubernetes	Container Management
Prometheus	Monitoring
Grafana	Alerting

Nagios	Application Monitoring & Solution
Containers	Tomcat/Apache HTTP Server XAAMP/LAMP/WAMP
Slack	Team Collaboration
Twilio	SMS/WhatsAPI/Chat/MMS Integration

5.3 Screenshots



```
C:\Projects\Java_Projects\github.com\ritikrana0169\PayBill-Application-main\PayBill-Application-main>C:\Program Files\Java\jdk-11\bin\java" Main
java.io.FileNotFoundException: ....\resource\newConsumer.txt (The system cannot find the path specified)
Error
java.io.FileNotFoundException: ....\resource\Consumer.txt (The system cannot find the path specified)
Error
-----Login Panel-----
| 1 : Admin Login      |
| 2 : User Login       |
| Press Any Key:Exit   |
|
| 1
-----|
| Admin User Name | |
| admin            |
-----|
| Admin Password | |
-----|
password|
```

```
C:\Projects\Java_Projects\github.com\ritikrana0169\PayBill-Application-main\PayBill-Application-main\src\com\paybeel>"C:\Program Files\Java\jdk-11\b  
in\java" Main  
java.io.FileNotFoundException: ....\resource\newConsumer.txt (The system cannot find the path specified)  
Error  
java.io.FileNotFoundException: ....\resource\Consumer.txt (The system cannot find the path specified)  
Error  
-----Login Panel-----  
| 1 : Admin Login |  
| 2 : User Login |  
| Press Any Key:Exit |  
  
1  
-----  
| Admin User Name |  
-----  
admin  
-----  
| Admin Password |  
-----  
password  
-----Admin Panel-----  
| 1 : View All Consumers |  
| 2 : View All The Bill Of Consumer |  
| 3 : View All The Bills |  
| 4 : Delete Consumer Account |  
| 6 : Back |  
| 7 : Generate Bill |  
| 8 : View All The Bills Paid & pending |  
| 0 : Exit |
```



```
C:\Projects\Java_Projects\github.com\ritikrana0169\PayBill-Application-main\PayBill-Application-main\src\com\paybeel>"C:\Program Files\Java\jdk-11\b  
in\java" Main  
java.io.FileNotFoundException: ....\resource\newConsumer.txt (The system cannot find the path specified)  
Error  
java.io.FileNotFoundException: ....\resource\Consumer.txt (The system cannot find the path specified)  
Error  
-----Login Panel-----  
| 1 : Admin Login |  
| 2 : User Login |  
| Press Any Key:Exit |  
  
2  
-----  
1: User Login  
2: User SignUp  
9: Back/SignOut  
0: Exit  
-----  
1  
-----  
| Consumer User Name |  
-----
```



```
C:\Projects\Java_Projects\github.com\ritikrana0169\PayBill-Application-main\PayBill-Application-main\src\com\paybeel>"C:\Program Files\Java\jdk-11\b  
in\java" Main  
java.io.FileNotFoundException: ....\resource\newConsumer.txt (The system cannot find the path specified)  
Error  
java.io.FileNotFoundException: ....\resource\Consumer.txt (The system cannot find the path specified)  
Error  
-----Login Panel-----  
| 1 : Admin Login |  
| 2 : User Login |  
| Press Any Key:Exit |  
2  
-----  
1: User Login  
2: User SignUp  
9: Back/SignOut  
0: Exit  
-----  
1  
-----  
| Consumer User Name |  
shyaam  
-----  
| User Password |  
sundar  
-----  
1: User Login  
2: User SignUp  
9: Back/SignOut  
0: Exit  
-----
```



6. Paytm Mall with DevOps

6.1 Description

Paytm mall is a platform provided by Paytm for user to shop, it just works like any other E-Commerce website or Application. Paytm Mall is an Indian e-commerce website. Paytm Mall strives hard to make your online shopping a more pleasant and cost-effective experience by providing the top quality products at the best price rates in India. Our collection of products consists of electronic appliances, clothes, grocery, home and kitchen products, backpacks, automobiles books and stationary to name a few.

6.2 Benefits of shopping on Paytm mall

User friendly

Leading online shopping experience in india which is simple and faster to load. Shop on the go and get the products delivered in just a few easy clicks.

Safe and Secure shopping

You need not worry about your account credentials or personal details.

Each and every transaction at Paytm Mall is securely processed and you can be rest assured that your money is in safe hands

Check Out closely

View products of your choice in detail with clear images and through description before buying

Multiple payment options

Choose a payment option of your choice i.e. Credit/Debit Cards, Net Banking and UPI

Use Paytm wallet for quick checkouts & hassle-free payments

Paytm postpaid services

Now you can buy today and pay next month with Paytm-ICICI Bank Postpaid

Order History

All your orders show up at one place, you can track them or repeat them with ease

Trouble free returns, track request & replacement of your orders

24X7 Customer Care

6.3 Pages & Features

Landing Page: Navbars, Categories, and footer section.

Sign-In Page: User confirmation to check whether user email and password match with details existing in the database.

Product Page: When user sign in with their respective email ids and page it will redirect to products page where user can select or add to cart for further process.

Cart Page: User can add products to this page or remove products added in the cart.

Checkout Page: User can add the delivery address

Payment Page: Users can add their payment details and also avail for discounts using promo code.

Order Processing Page: setTimeout function is used to emulate original payment flow.

Additional Features: Kept the branding, fonts, colors, styling and favicon and titles consistent throughout the whole website to tried building atleast mvp of site. Not exact but tried building same features as in original site.

6.4 How to use

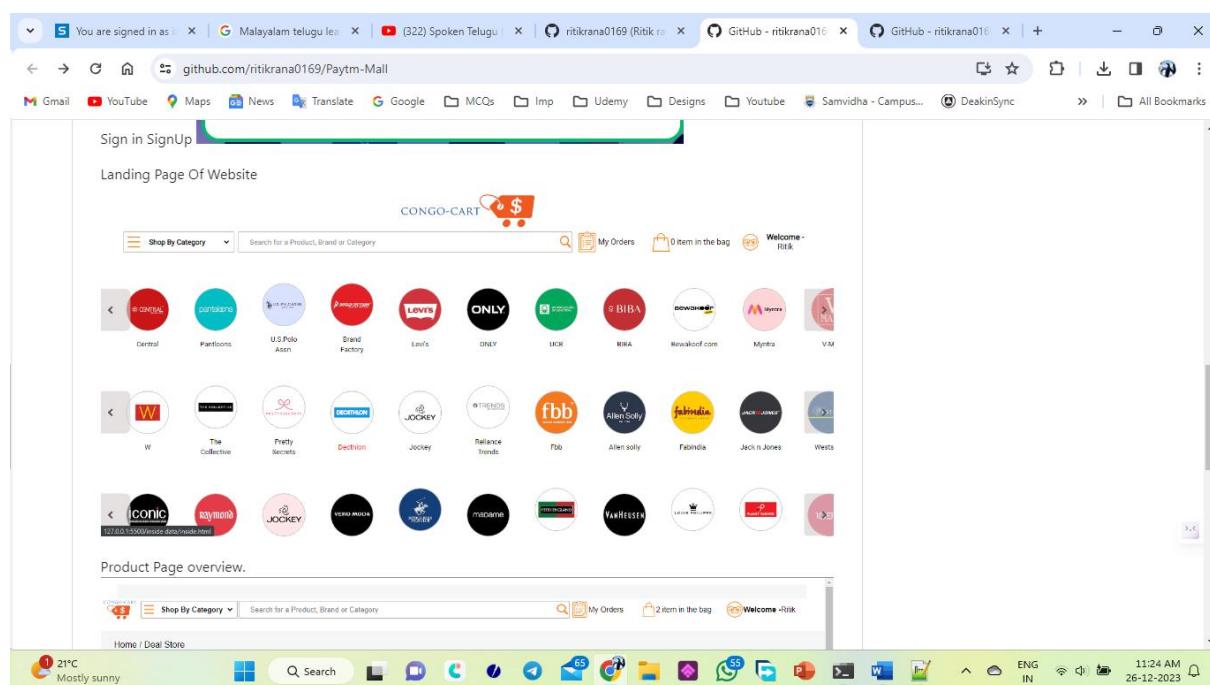
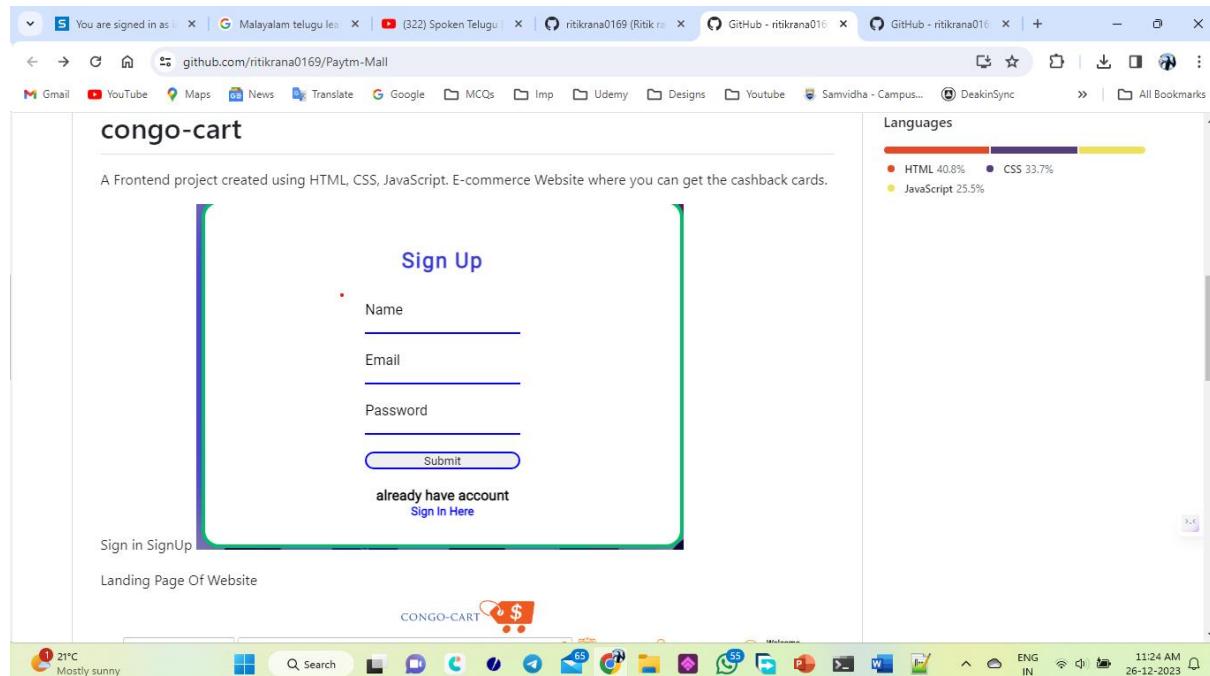
- At first user has to click on login on landing page. User have to signup if he does not have an account. In signup page user has to fill all the fields in the form.
- After signup user has to login with same details, otherwise it will show invalid credentials. After login user will be redirected to home page.
- User can see shop by categories and he / she can click on any category as per his / her requirement and it will be redirected to particular category page.
- User can see the products and clicking on particular product which will be redirected to order page.
- In order page user can see the details of that particular product and also the add to cart button. After clicking the add to cart button the cart value gets updated which is on top nav bar.
- If he clicks on cart then he will be redirected to cart page where he can see products which he has added to cart. If he wants to remove any product he can click on remove button which will remove the product.
- Then he will proceed to payment page, where he has to fill all his personal details for delivering the product.
- After filling all the details including card details then click on pay button. It will take some time for payment.
- After payment has completed successfully then he will be redirected to home page.

6.5 Softwares & Technologies used in the Project

Front End	
HTML	Layout & Form Designing
CSS	Page Styling
Bootstrap	Responsive Web Designing
jQuery	AJAX Calling & DOM Manipulation
Angular/ReactJS	Single Page Application calling web application
Middleware	
Mongoose	Object Data Model (ODM) library for MongoDB and Node.js
MongoDB	NoSQL database management program alternative to traditional relational databases. MongoDB is a document oriented database program that uses JSON-like documents with optional schemas (relationship between tables).
Node.js	Open source, cross platform JavaScript runtime environment that executes on command prompt to run server side programs. It is built on Chrome's V8 JavaScript engine and uses a event-driven, non-blocking I/O model that makes it light weight and efficient. It is used in web application development and also used in desktop and mobile applications also.
Express.js	Free and open source Backend web application framework for building RESTful APIs with Node.js. It is designed for building APIs. Routing, Middleware, Templating, View Helpers, Static file serving
Database	
Oracle/MySQL/SQLite /PostgreSQL/HSQLDB	Rational Database Management system to perform manipulation & query operations on tables

DevOps Tools	
Git	Version Control System
ANT/Maven	Build Tool
Jenkins	Continuous Integration
Junit/HttpUnit	Unit Testing
Selenium	Web Page Testing
SonarQube	Code Analysis
JIRA	Issue Tracking
Jfrog Artifactory	Artifact Repository
Docker	Containerization
Ansible	Deployment & Configuration Management
Kubernetes	Container Management
Prometheus	Monitoring
Grafana	Alerting
Nagios	Application Monitoring & Solution
Containers	Tomcat/Apache HTTP Server XAAMP/LAMP/WAMP
Slack	Team Collaboration
Twilio	SMS/WhatsAPI/Chat/MMS Integration

6.6 Screenshots



The screenshot shows a web browser window with multiple tabs open. The main content area displays two pages from the Paytm-Mall website.

Product Page Overview:

- The URL is github.com/ritikrana0169/Paytm-Mall.
- The page shows a search bar and navigation links like "Shop By Category" and "Welcome - Riti".
- A sidebar on the left lists categories: "Deals For You", "Fashion", and "Apparels".
- The main content area shows a grid of products under the heading "ONLY - 6Products".
- Each product card displays the brand name "ONLY", a price of ₹1000, a discount of ₹100 off, and a "Add To Cart" button.

Cart Page Overview:

- The URL is github.com/ritikrana0169/Paytm-Mall.
- The page shows a search bar and navigation links like "Shop By Category" and "Welcome - Riti".
- The top of the page indicates a total of "Total: 4490".
- A "CHECKOUT" button is prominently displayed.
- The main content area shows a grid of items in the cart.
- Items include a red item labeled "BIBA" with a price of ₹1500 and a green item labeled "UCB" with a price of ₹2990.
- Quantity dropdowns and "Remove" buttons are shown for each item.

The browser interface includes a toolbar at the top with various icons and a taskbar at the bottom showing system status and date/time (11:24 AM, 26-12-2023).

7. Student Management System with DevOps

7.1 Description

Schools and universities serve as the basis of knowledge as well as an educational body on which students rely. As a result, they must maintain a robust student database in order to store all current records and readily exchange information with students.

Most schools and universities rely on a sophisticated software program known as a 'Student Management System (SMS)' to manage all of their student data and administrative procedures such as tests, attendance, and other activities.

The use of the Student Management System has improved the performance and efficiency of the education industry in recent years. With its well-organized, simple, and dependable online school management software, this application has effectively taken over the admin department's task.

7.2 Softwares & Technologies used in the Project

Front End	
HTML	Layout & Form Designing
CSS	Page Styling
Bootstrap	Responsive Web Designing
jQuery	AJAX Calling & DOM Manipulation
Angular/ReactJS	Single Page Application calling web application
Middleware	
JDBC	Java to Database Connectivity
Servlets	Server-side form processing, Session management & Client state persistence, Security management, Page Includes, Page Routing, Page-Page communication
JSP	Dynamic web page generation
Java Mail API	E-Mail integration
Java Tools	Log4J, SLF4J, ANT, Maven, JUnit, HttpUnit, Selenium
Database	
Oracle/MySQL/SQLite /PostgreSQL/HSQLDB	Rational Database Management system to perform manipulation & query operations on tables
DevOps Tools	
Git	Version Control System
ANT/Maven	Build Tool
Jenkins	Continuous Integration
Junit/HttpUnit	Unit Testing
Selenium	Web Page Testing
SonarQube	Code Analysis
JIRA	Issue Tracking
Jfrog Artifactory	Artifact Repository
Docker	Containerization
Ansible	Deployment & Configuration Management
Kubernetes	Container Management
Prometheus	Monitoring
Grafana	Alerting
Nagios	Application Monitoring & Solution
Containers	Tomcat/Apache HTTP Server XAAMP/LAMP/WAMP
Slack	Team Collaboration

Twilio	SMS/WhatsAPI/Chat/MMS Integration
--------	-----------------------------------

7.3 Project Overview

Project Name:	Student Management System Project in Java
Abstract	It's a GUI-based project used with the Swing module to organize all the elements that work under library management.
Language/s Used:	Java
IDE	IntelliJ Idea Professional (Recommended)
Java version (Recommended):	Java SE 18.0. 2.1
Database:	MySQL
Type:	Desktop Application
Recommended for	Final Year Students

7.4 System Design Requirements

The Input Design:

The input design consists of exact input instructions that are simple, logical, and error-free. The data input and format, including allotted space and field sequence, are already included in the source document. Online data entering into the input form should be error-free. It employs a processor that receives commands and data from the user in order to analyse and then proceed.

Depending on whether it is correct or not, the procedure proceeds and it is either approved or denied.

The input stage is a collection of various backgrounds:

- Recording and transcribing data
- Conversion of Data
- Verification of Data
- Correction of Data
- Transmission of Data

The Output Design:

The output design aims to produce a precise and efficient result to the query entered by the user. Because they are the provider's direct source of information, they must produce satisfactory answers that answer the inquiry. The specific outputs for the questions are set along with their forms during the program's logical design.

The Database Design:

The database is where users' interrelated data is saved in order to present them with an effective solution. As one of the process's inputs and outputs, database design is critical to making information accessible and adaptable for users to retrieve.

- `id` – An unique id for each record which will be AUTO_INCREMENT
- `name` – Name of the student
- `dob` – Date of birth of the student

- `fname` – First name of the student
- `lname` – Last name of the student
- `sex` – Gender of the student
- `blood` – The blood group of the student
- `subject` – List of subjects belonging to a particular student
- `address` – Address of the student
- `studentid` – An unique ID that will help to track a particular student.

7.5 Modules, Functions, and Requirements

This application is a **JFrame** and **MySQL** project that runs on the Tomcat server; it may also be run in Eclipse and Netbeans. We created this JFrame and My SQL Project on Student Management System to automate the procedure of the Student Management.

The primary functions of this project are to manage Logins, Fees, Profiles, Students, Courses, and Exams.

This is a Major Project in Java that is suited for students searching for final-year java projects. The project's primary modules are the Logins module, Fees module, Profiles module, Student module, and Courses module, which conduct all operations in their respective domains. This is a secure online application that runs within the JVM.

Let's see some important modules:

Login module

The name makes it obvious what this function is for. It allows both the user and the administrator to log in. The initial login would always be the admin login, with the same password, admin.

After logging in as an administrator, you may do administrative tasks such as adding, updating, generating and deleting student information.

GUI module

This function, as the name implies, links the database to the GUI. After connecting the database, the function must be started by entering the database's login and password. The database, tables, and data may then be added to the table using the create function. This is accomplished through the use of SQL queries, which aid in connecting to the GUI and enabling login.

Reports of Student Management System

Reports of Student Management system

Logins report: We may produce all Logins reports in this report.

Fees report: Generate Fees reports.

Profiles report: All Profiles reports will be available in this part

Student report: You can produce filtered Student reports

You can generate Student, Logins, and Courses reports

Dynamic search is available for Fees, Exams, and Profiles.

PDF reporting has been included for exporting reports for Logins, Fees, and Profiles modules.

CSV reports for Logins, Students, and Fees may be created.

Modules of Student Management System

Login module: From this module, we can handle all Login operations.

Logins module: From this module, we can handle all Login-related activities.

Fees module: This module is often used to manage Fee operations.

Profiles module: Profiles will handle all actions related to Profiles.

Student module: This module is in charge of Student functions.

Course module: Courses modules perform all the create, read, update and delete operations of Courses

Exams module: It manages all Exams functionality.

Software Requirements

JDK 1.8: You need JDK 1.8 to execute this project.

Tomcat 8: The Tomcat 8 server is used for this project.

MySQL: This project requires the use of a MySQL database.

MySQL JConnector: This is required for connecting MySQL and Java.

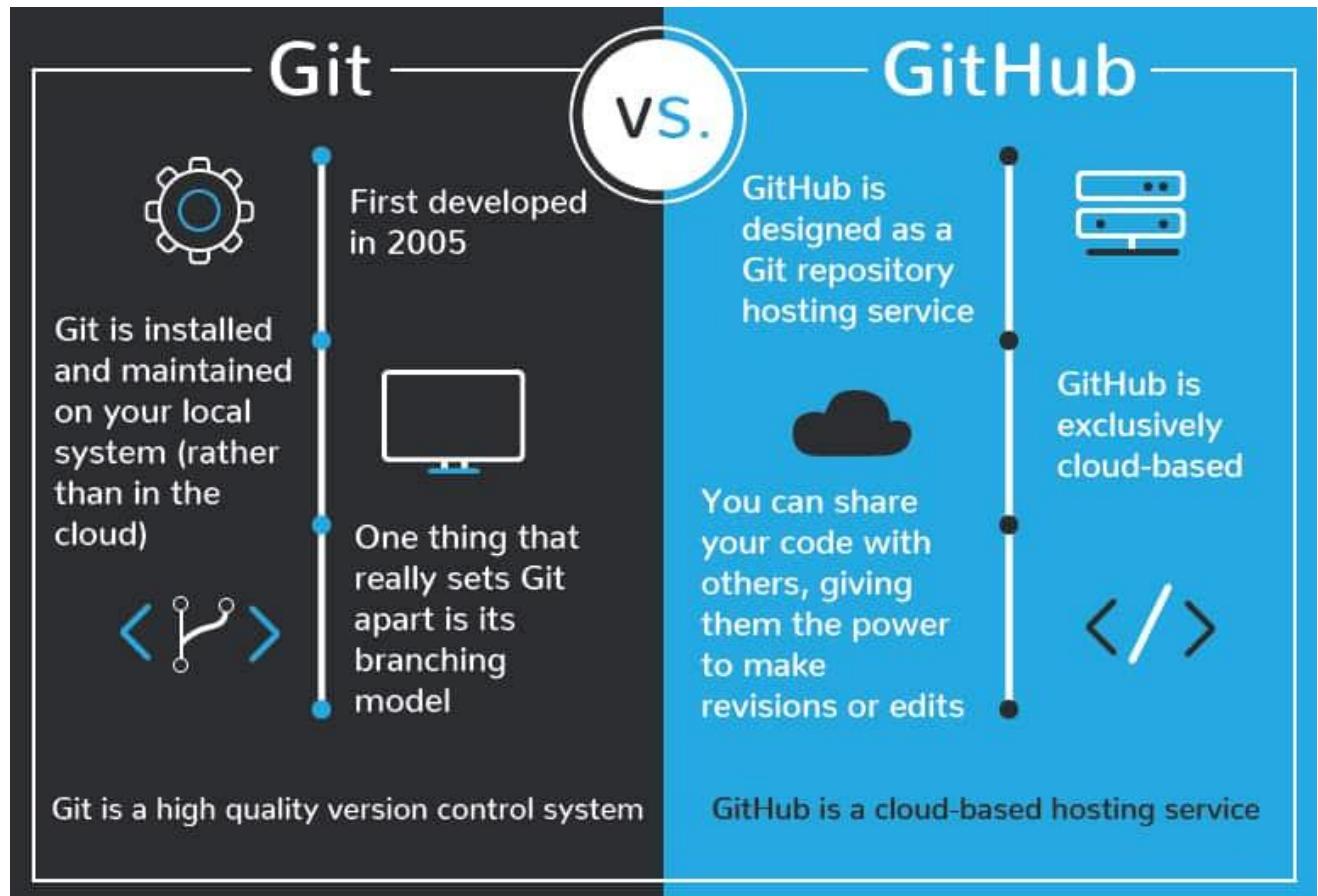
8. Automated Web Development Environment

8.1 Description

In this project I have created an automated web development environment integration Git/GitHub, Jenkins, and Tomcat Web Server in Windows.

Git is a version control system that lets you manage and keep track of your source code history.

GitHub is a Cloud based hosting services that lets you manage Git Repositories.

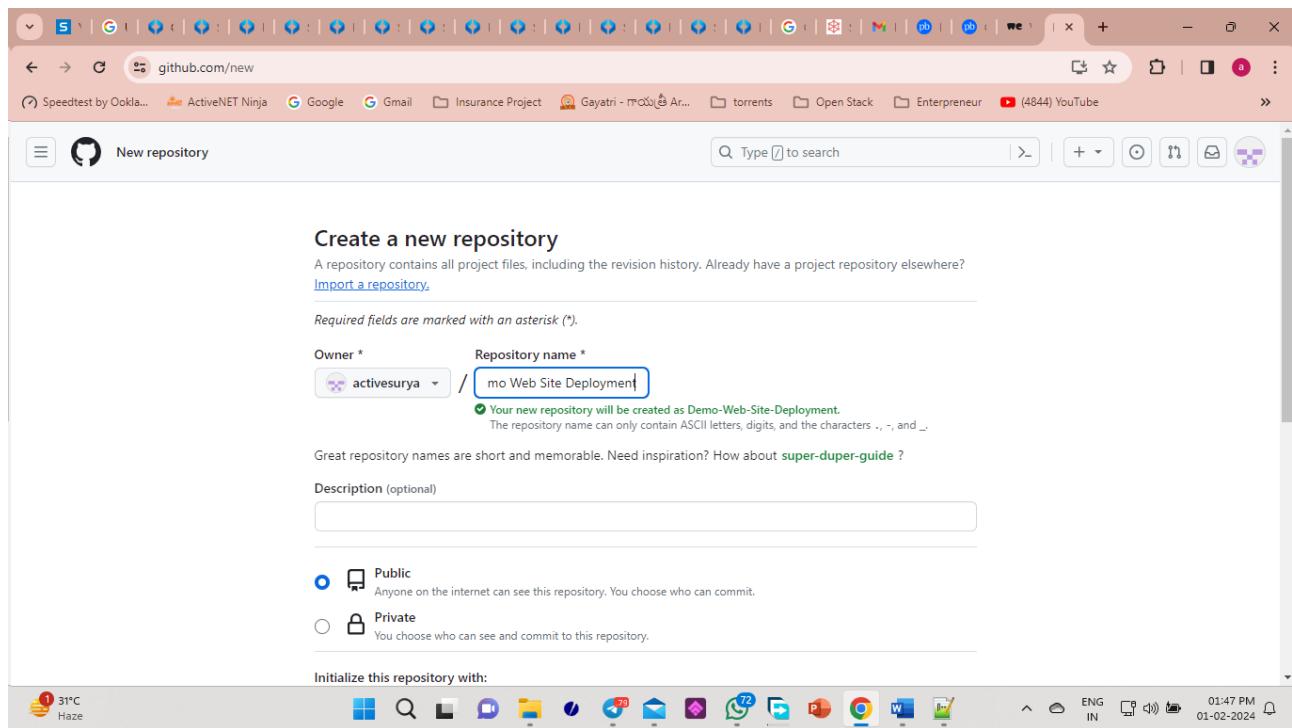


S.No.	Git	GitHub
1.	Git is a software.	GitHub is a service.
2.	Git is a command-line tool	GitHub is a graphical user interface
3.	Git is installed locally on the system	GitHub is hosted on the web
4.	Git is maintained by Linux.	GitHub is maintained by Microsoft.
5.	Git is focused on version control and code sharing.	GitHub is focused on centralized source code hosting.

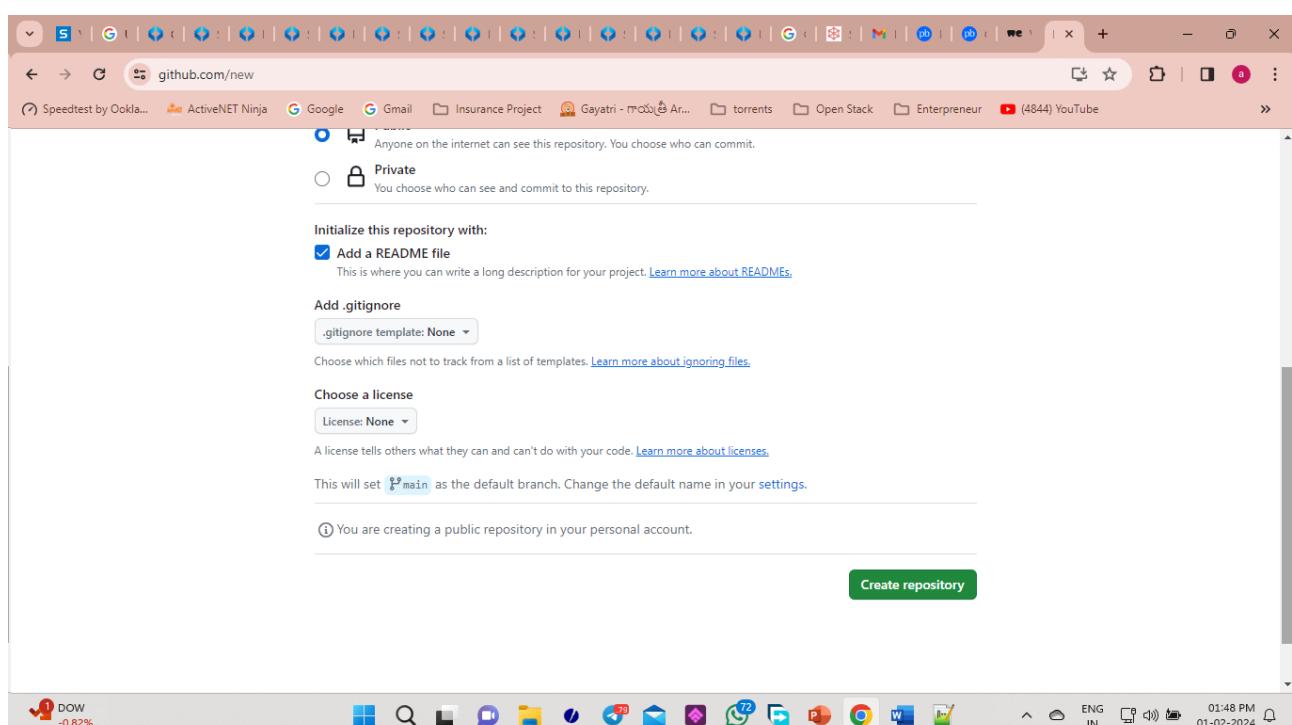
S.No.	Git	GitHub
6.	Git is a version control system to manage source code history.	GitHub is a hosting service for Git repositories.
7.	Git was first released in 2005.	GitHub was launched in 2008.
8.	Git has no user management feature.	GitHub has a built-in user management feature.
9.	Git is open-source licensed.	GitHub includes a free-tier and pay-for-use tier.
10.	Git has minimal external tool configuration.	GitHub has an active marketplace for tool integration.
11.	Git provides a Desktop interface named Git Gui.	GitHub provides a Desktop interface named GitHub Desktop.
12.	Git competes with CVS, Azure DevOps Server, Subversion, Mercurial, etc.	GitHub competes with GitLab, Bit Bucket, AWS Code Commit, etc.

Now the first thing is setting up GitHub is to create a GitHub account and I assume everyone can create it. Now we need to setup git in our Operating System. Download Git from <https://git-scm.com/download/win>.

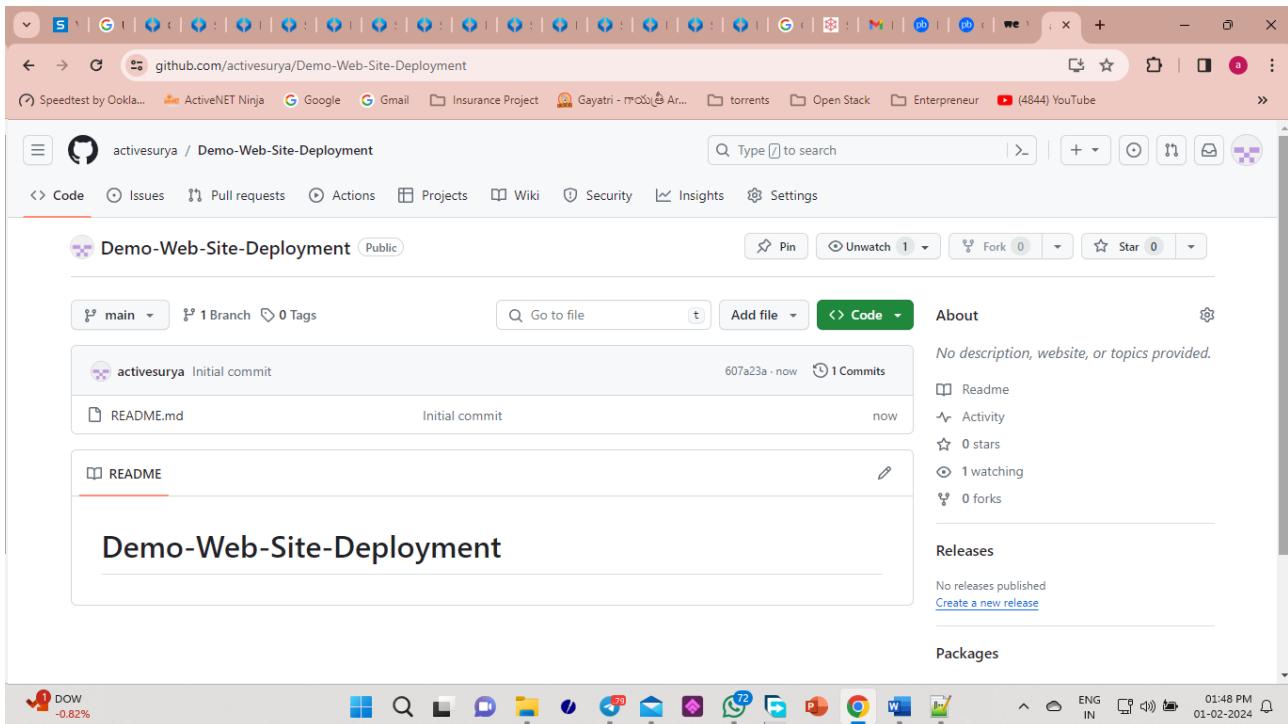
After downloading Git in Windows/Linux/MacOS we can use terminal to run Git commands. Now after setting up Git environment goto GitHub and create a new repository say "Demo Web Site Deployment". You can give any name to repository. Now by using terminal/git move to the directory in which you have your website files which need to be uploaded and there are some commands of GitHub by using which you can upload those file to GitHub.



The screenshot shows the GitHub 'Create a new repository' page. The URL in the address bar is `github.com/new`. The repository name is set to `Demo Web Site Deployment`. The repository is marked as public. A note says, "Your new repository will be created as Demo-Web-Site-Deployment." Below the repository name, there's a link to "Import a repository". The description field is empty. Under "Initialize this repository with:", the "Public" option is selected. The status bar at the bottom shows system information like weather (31°C Haze), battery level, and system date/time (01-02-2024 01:47 PM).



This screenshot shows the continuation of the GitHub repository creation process. It includes options for adding a README file (which is checked) and choosing a license (None). A note states, "Choose which files not to track from a list of templates." Below that, it says, "A license tells others what they can and can't do with your code." and "This will set `main` as the default branch." A note at the bottom indicates, "You are creating a public repository in your personal account." A large green "Create repository" button is prominently displayed. The status bar at the bottom remains the same as the previous screenshot.



<https://github.com/activesurya/Demo-Web-Site-Deployment.git>

open command prompt

create one folder named git_temp

git init

git commit -m "First commit"

git branch -M master

git remote add origin <https://github.com/activesurya/Demo-Web-Site-Deployment.git>

git push -u origin master

git push origin main

Basic commands required of GitHub

git config – used to configure username and user email of GitHub account.

git init – Is used to initialize any directory as GitHub repo in system.

git add file_name / git add * - It will add your file or if you give * then it will add all files present in the directory.

git commit -m "Initial commit" – It will commit all the files added to git.

git remote add origin link_to_github_repo – It will add a remote location of repo.

Ex: git remote add origin <https://github.com/OWNER/REPOSITORY.git>

git remote add origin <https://github.com/octocat/Spoon-Knife.git>

git push -u origin master – It is used to upload your files to repository.

git status – It is used to check the status of the repo whether file is committed or not, is there any change in metadata of files or not.

git fetch – It is used to fetch data from the GitHub about the latest commit.

git pull – it is used to fetch and merge the changes in GitHub repo to local repo.

git log – It will show the version history of the files.

git clone – It is used to clone the complete repository from the github to your local system.

One of the important concept is that whenever we run git init command it will initialize an empty GitHub repo name .git and it will be hidden. It will contain all the details of your codes and commits like logs, hooks etc.

Detailed Git commands are available in <https://git-scm.com/docs>

Jenkins

Jenkins is a software that allows continuous integration. Jenkins will be installed on a server where the central build will take place. Jenkins is an open source automation tool written in Java programming language that allows continuous integration.

Jenkins builds and tests our software projects which continuously making it easier for developer to integrate changes to the project and making it easier for users to obtain a fresh build.

It also allows us to continuously deliver our software by integrating with a large number of testing and deployment technologies.

Now let's setup Jenkins and webserver in Windows. Before configuring Jenkins we ned to install Java JDK 11 in our system. After installing java follow these steps to install Jenkins.

Visit official [Jenkins](#) website and select your OS from LTS version. <https://www.jenkins.io/download/>

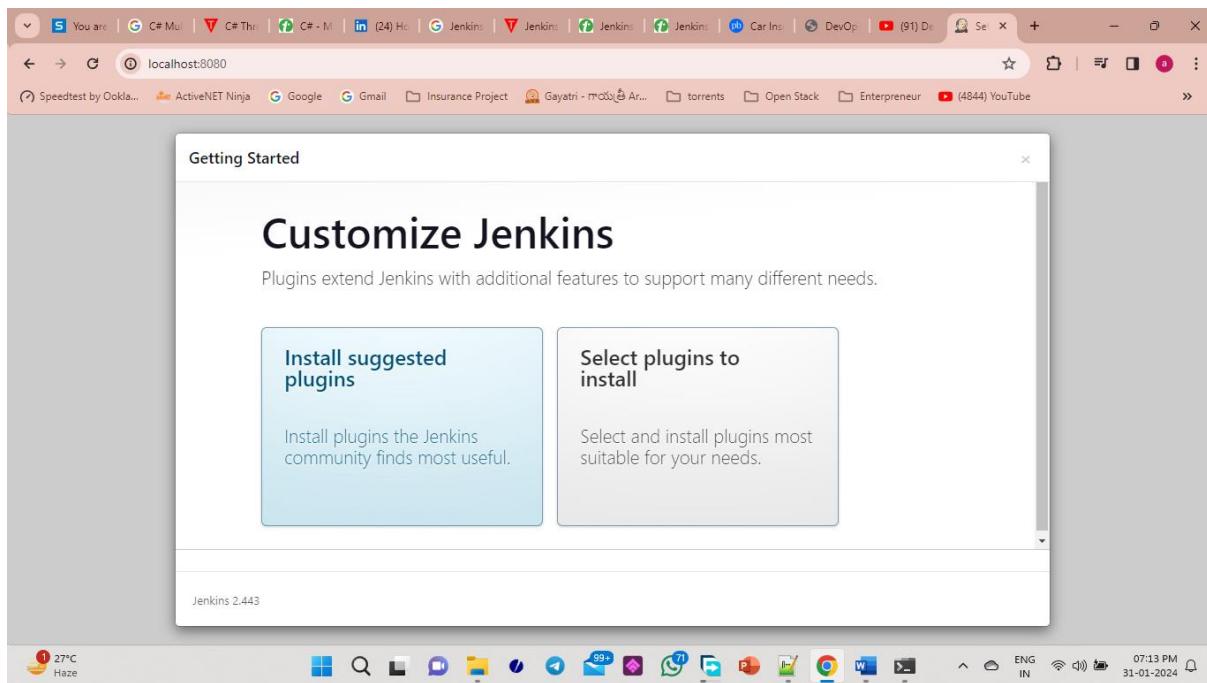
Download Jenkins 2.426.3 LTS: <https://get.jenkins.io/war-stable/2.426.3/jenkins.war>

Goto folder where jenkins is installed and type

```
java -jar jenkins.war
```

Jenkins by default runs on port number 8080.

Open browser and type <http://localhost:8080>



This screenshot shows the Jenkins 'Organization and Administration' section of the plugin selection interface. On the left, there's a sidebar with categories like 'Organization and Administration', 'Build Features', etc. On the right, a list of selected plugins is shown, with a note that the full list is not visible here. The selected plugins include 'Dashboard View', 'Folders', 'Configuration as Code', and 'OWASP Markup Formatter'. The Jenkins version is listed as 2.443 at the bottom left.

The screenshot shows the Jenkins 2.443 'Getting Started' page. A tooltip is displayed over the 'Folders' link in the sidebar, listing various Jenkins modules:

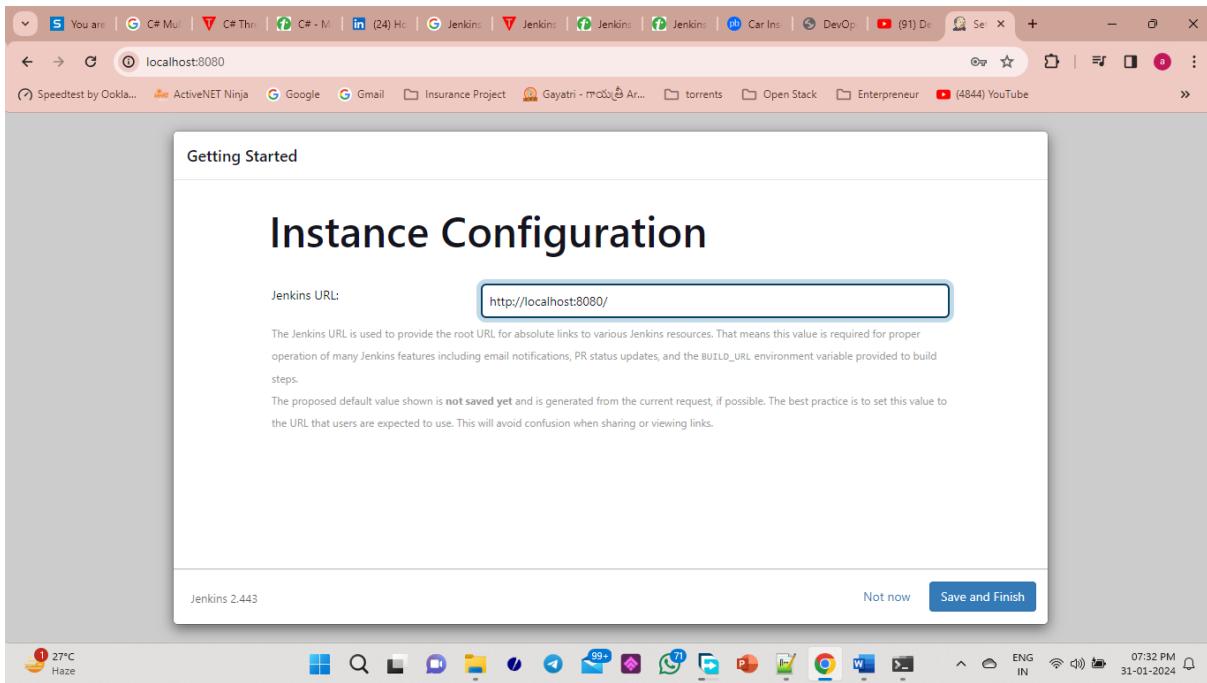
- QAASP Markup Formatter
- Build Timeout
- Credentials Binding

The tooltip also includes a note at the bottom: '** - required dependency'.

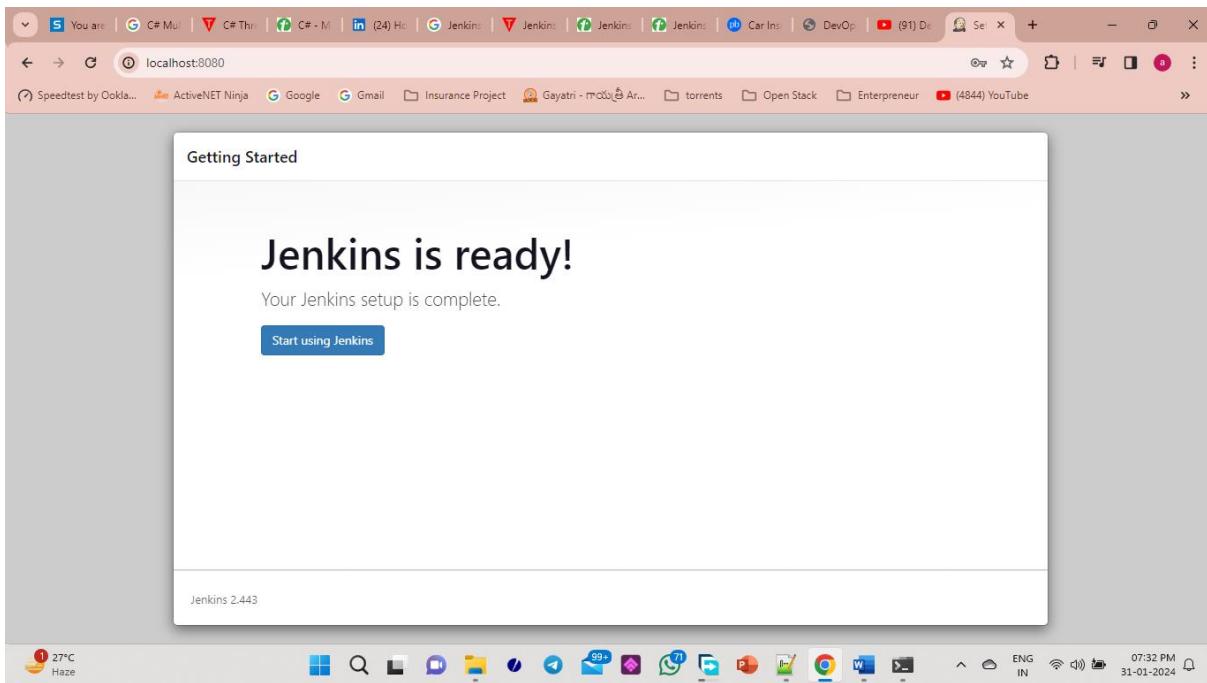
The screenshot shows the Jenkins 2.443 'Getting Started' page. A tooltip is displayed over the 'QAASP Markup Formatter' link in the sidebar, listing its dependencies:

- JSON Path API
- Structure API
- Step API
- Token Macro

The tooltip also includes a note at the bottom: '** - required dependency'.



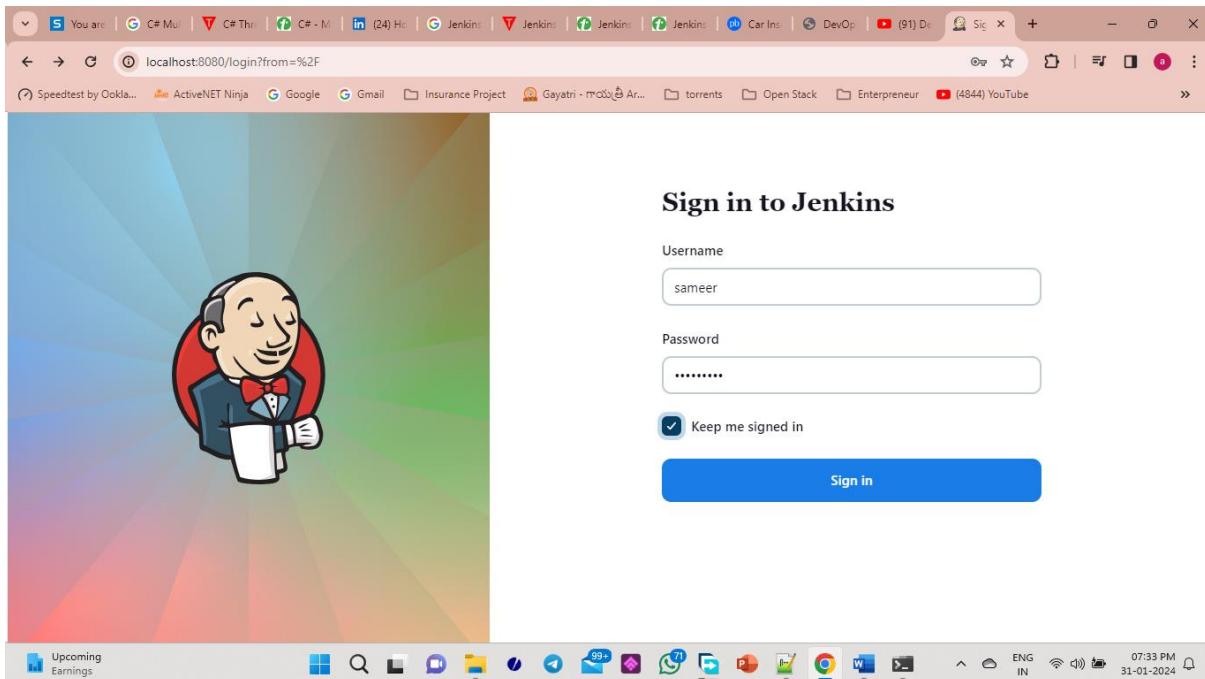
The screenshot shows the Jenkins 'Instance Configuration' page. At the top, it says 'Getting Started'. Below that is a large section titled 'Instance Configuration'. A 'Jenkins URL' input field contains the value 'http://localhost:8080/'. A note below the field states: 'The Jenkins URL is used to provide the root URL for absolute links to various Jenkins resources. That means this value is required for proper operation of many Jenkins features including email notifications, PR status updates, and the BUILD_URL environment variable provided to build steps.' Another note below it says: 'The proposed default value shown is **not saved yet** and is generated from the current request; if possible. The best practice is to set this value to the URL that users are expected to use. This will avoid confusion when sharing or viewing links.' At the bottom of the configuration section, there are two buttons: 'Not now' and 'Save and Finish'. The status bar at the bottom of the browser window shows 'Jenkins 2.443'.



The screenshot shows the Jenkins 'Getting Started' page after configuration. It features a large heading 'Jenkins is ready!' and the subtext 'Your Jenkins setup is complete.' Below this is a blue 'Start using Jenkins' button. The status bar at the bottom of the browser window shows 'Jenkins 2.443'.

The screenshot shows the Jenkins dashboard at localhost:8080. The left sidebar includes links for 'New Item', 'People', 'Build History', 'Manage Jenkins', and 'My Views'. A 'Build Queue' section indicates 'No builds in the queue.' Below it, a 'Build Executor Status' section shows '1 Idle' and '2 Idle'. The main content area features a 'Welcome to Jenkins!' message and a 'Start building your software project' button. It also includes sections for 'Set up a distributed build', 'Create a job', and 'Configure a cloud'. A bottom banner provides links to 'Learn more about distributed builds' and other resources.

A screenshot of a web browser showing the Jenkins sign-in page. The URL in the address bar is "localhost:8080/login?from=%2F". The page features a cartoon illustration of a bald man with a bow tie holding a coffee cup on the left side. The main area contains the text "Sign in to Jenkins" and fields for "Username" and "Password". There is also a "Keep me signed in" checkbox and a blue "Sign in" button.



```

2024-01-31 13:50:27.860+0000 [id=77] INFO h.m.UpdateCenter$UpdateCenterConfiguration#download: Downloading ldap
2024-01-31 13:50:34.939+0000 [id=77] INFO h.model.UpdateCenter$DownloadJob#run: Installation successful: ldap
2024-01-31 13:50:34.940+0000 [id=77] INFO h.model.UpdateCenter$DownloadJob#run: Starting the installation of email-ext on behalf of admin
2024-01-31 13:50:34.940+0000 [id=77] INFO h.m.UpdateCenter$UpdateCenterConfiguration#download: Downloading email-ext
2024-01-31 13:50:37.885+0000 [id=77] INFO h.model.UpdateCenter$DownloadJob#run: Installation successful: email-ext
2024-01-31 13:50:37.886+0000 [id=77] INFO h.model.UpdateCenter$DownloadJob#run: Starting the installation of mailer on behalf of admin
2024-01-31 13:50:37.887+0000 [id=77] INFO h.m.UpdateCenter$InstallationJob#run: Skipping duplicate install of: Mailer@463.vedf8358e006b_
2024-01-31 13:50:37.888+0000 [id=77] INFO h.model.UpdateCenter$DownloadJob#run: Installation successful: mailer
2024-01-31 13:50:37.888+0000 [id=77] INFO h.model.UpdateCenter$DownloadJob#run: Starting the installation of dashboard-view on behalf of admin
2024-01-31 13:50:37.888+0000 [id=77] INFO h.m.UpdateCenter$UpdateCenterConfiguration#download: Downloading dashboard-view
2024-01-31 13:50:39.522+0000 [id=77] INFO h.model.UpdateCenter$DownloadJob#run: Installation successful: dashboard-view
2024-01-31 13:50:39.523+0000 [id=77] INFO h.m.UpdateCenter$UpdateCenterConfiguration#download: Downloading theme-manager
2024-01-31 13:50:39.524+0000 [id=77] INFO h.m.UpdateCenter$UpdateCenterConfiguration#download: Starting the installation of theme-manager on behalf of admin
2024-01-31 13:50:41.064+0000 [id=77] INFO h.model.UpdateCenter$DownloadJob#run: Installation successful: theme-manager
2024-01-31 13:50:41.064+0000 [id=77] INFO h.model.UpdateCenter$DownloadJob#run: Starting the installation of dark-theme on behalf of admin
2024-01-31 13:50:41.065+0000 [id=77] INFO h.m.UpdateCenter$UpdateCenterConfiguration#download: Downloading dark-theme
2024-01-31 13:50:42.700+0000 [id=77] INFO h.model.UpdateCenter$DownloadJob#run: Installation successful: dark-theme
2024-01-31 13:50:42.700+0000 [id=77] INFO h.m.UpdateCenter$CompleteBatchJob#run: Completing installing of plugin batch
2024-01-31 13:50:46.658+0000 [id=77] INFO h.p.b.g.GlobalTimeOutConfiguration#load: global timeout not set
2024-01-31 13:50:48.694+0000 [id=77] INFO h.m.DownloadService$Downloadable#load: Obtained the updated data file for hudson.tasks.Ant.AntInstaller
2024-01-31 13:50:49.970+0000 [id=77] INFO h.m.DownloadService$Downloadable#load: Obtained the updated data file for hudson.plugins.gradle.GradleInstaller
2024-01-31 13:50:50.612+0000 [id=293] INFO jenkins.InitReactorRunner$1#onAttained: Started initialization
2024-01-31 13:50:50.623+0000 [id=298] INFO jenkins.InitReactorRunner$1#onAttained: Listed all plugins
2024-01-31 13:50:50.628+0000 [id=298] INFO jenkins.InitReactorRunner$1#onAttained: Prepared all plugins
2024-01-31 13:50:50.669+0000 [id=293] INFO jenkins.InitReactorRunner$1#onAttained: Started all plugins
2024-01-31 13:50:50.675+0000 [id=294] INFO jenkins.InitReactorRunner$1#onAttained: Augmented all extensions
2024-01-31 13:50:50.681+0000 [id=293] INFO jenkins.InitReactorRunner$1#onAttained: System config loaded
2024-01-31 13:50:50.684+0000 [id=293] INFO jenkins.InitReactorRunner$1#onAttained: System config adapted
2024-01-31 13:50:50.727+0000 [id=293] INFO jenkins.InitReactorRunner$1#onAttained: Loaded all jobs
2024-01-31 13:50:50.760+0000 [id=295] INFO jenkins.InitReactorRunner$1#onAttained: Configuration for all jobs updated
2024-01-31 13:50:51.437+0000 [id=299] INFO jenkins.InitReactorRunner$1#onAttained: Completed initialization
2024-01-31 13:50:51.439+0000 [id=77] INFO h.m.UpdateCenter$CompleteBatchJob#run: Completed installation of 87 plugins in 4 min 4 sec

```

After this you will redirected to Jenkins dashboard portal. Now we can lot of tasks like creating Jobs, downloading plugins etc.

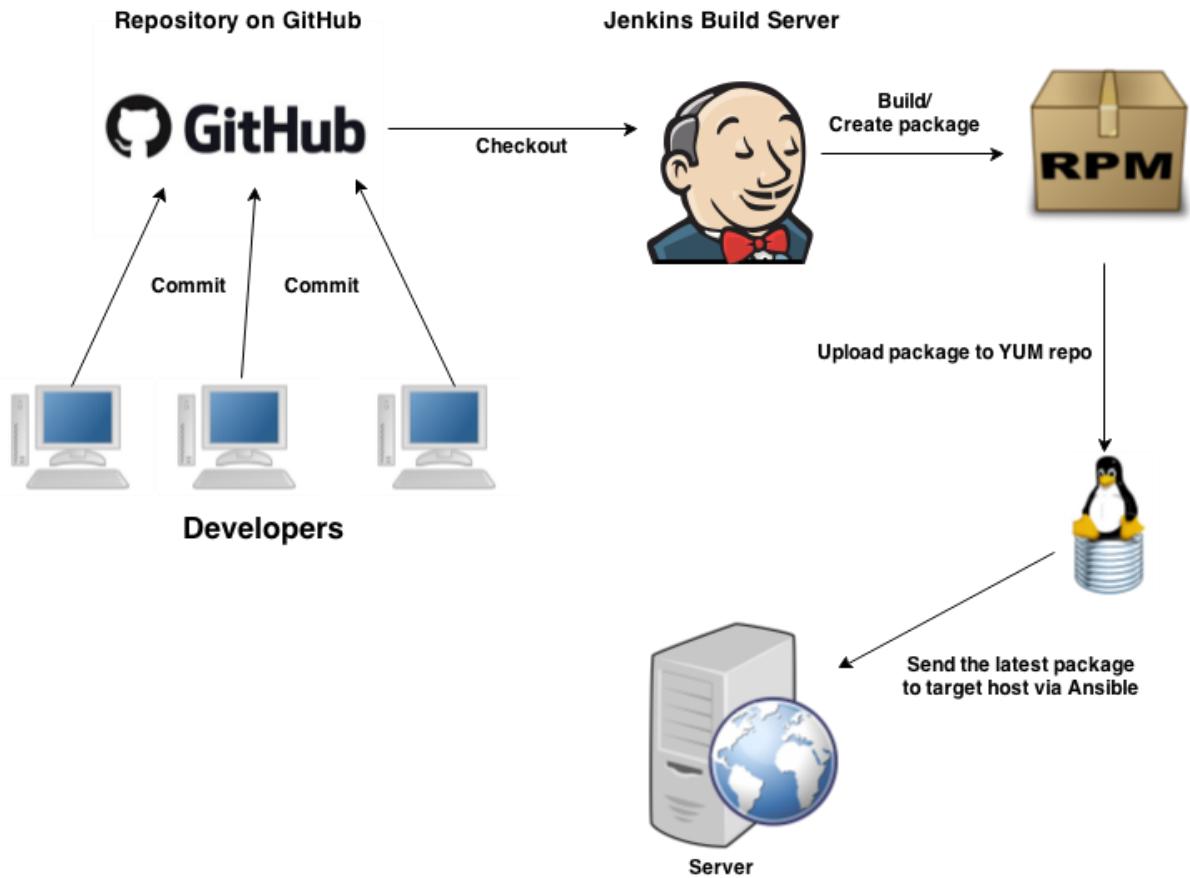
To install any plugin goto: Manage Jenkins → Manage Plugins → Available → Search for GitHub and select it and then click on Download and install.

Now we have done with setup of Jenkins and installed GitHub plugin in it.

Download and install Tomcat Web Server.

Now we can go for the concept of Automation. Now let's see how we interlink all these different applications and achieve complete automation.

How it works all together



Before implementing let's see the concept behind the automation using Jenkins. In the figure we have seen different stages. We can see Developers, GitHub, Jenkins build server, and Tomcat Web Server hosted on Windows. The concept behind this automation is at first, we have group of developers working on a project, let say we are building a portfolio website. As a developer, developer makes some changes in Home Page, About Us page etc. He will keep committing and uploading these pages to his GitHub account. Then we have GitHub which will be triggered to the Jenkins which is being installed in Windows OS.

As the GitHub receives any new commit it will send it to the Jenkins server and Jenkins will then send it to Windows and execute a shell command which is given for that particular job in Jenkins shell.

A company would use a CI/CD pipeline integrated with Jenkins, Maven, GitHub, and Apache Tomcat to streamline and automate the software development and deployment processes. Jenkins serves as the automation server, orchestrating the pipeline, while Maven manages project dependencies and builds. GitHub acts as a version control system, enabling collaboration, and Apache Tomcat facilitates the smooth deployment of applications, ensuring efficient and consistent delivery of software updates. This integrated toolchain enhances development speed, ensures code quality, and simplifies the continuous delivery of software applications.