**CS691 – Jenkins Progress Report**

**Team 1**

**Jenkins Admin:** Bhavin Himatkumar Goswami

**Application:** EzCars

**Sprint 1:** Planned and Completed Tasks (10/08/2024 to 10/29/2024)

| **Planned Tasks** | **Completed Tasks** |
| --- | --- |
| 1. **Integrating and Automating QA**     * Connect QA pipeline to the QA instance  * Set up GitHub webhook for auto code pulling and deployment from the QA branch  1. **Plugin Evaluation and Installation**  * Install Blue Ocean Plugin for enhanced pipeline visualization. * Install Slack Plugin for build notifications.  1. **Plugin Installation and Configuration**  * Install and configure selected plugins. * Integrate with Slack and test build notifications.  1. **Jenkins Security**  * Update Jenkins plugins to the latest versions * Identify and fix pipeline scripts vulnerabilities  1. **Pipeline Optimization**  * Implement deployment stages in Dev and QA pipelines * Configure conditional triggers  1. **Documentation**  * Document Jenkins setup and plugin usage * Share updated CI/CD process with the team | 1. **Integrating and Automating QA**  * Connected QA pipeline to instance and configured GitHub webhook for implementing automatic code pulling and deployment for the QA branch upon new commits.  1. **Plugin Evaluation and Installation**  * Installed Blue Ocean for enhanced visualization and Slack for real-time build notifications.  1. **Plugin Installation and Configuration**  * Configured plugins to meet project needs and integrated Slack with Jenkins, successfully tested for timely build alerts.  1. **Jenkins Security**  * Updated all plugins to the latest versions and resolved vulnerabilities in pipeline scripts, using SSH Agent for secure access, enhancing security.  1. **Pipeline Optimization**  * Implemented deployment stages in both Dev and QA pipelines and added conditional triggers for more controlled and efficient builds.  1. **Documentation**  * Documented Jenkins usage, providing an easy reference for team members on slack. |

**Sprint 2:** Planned and Completed Tasks (10/30/2024 to 11/19/2024)

| **Planned Tasks** | **Completed Tasks** |
| --- | --- |
| 1. **Automated Workspace Cleanup**  * Set up automatic cleanup for workspaces after builds to maintain storage efficiency without manual intervention.  1. **Backup and Recovery**  * Set up scheduled backups for Jenkins configuration, plugins, and jobs to ensure recoverability in case of server failure. * Create a simple guide on restoring Jenkins from a backup, so anyone on the team can follow it if necessary.  1. **Testing Integration**  * Add dedicated stages for unit and integration testing in the Jenkins pipeline to verify code quality and functionality on each code push. * Create automated unit tests for core components using Jest and React Testing Library to cover key functions like UI updates and calculations. * Configure Jenkins to display test results using junit or direct Jest output which will allow quick review of pass/fail statuses and error logs.  1. **Disk Usage Monitoring**  * Install a disk usage plugin to monitor storage and set up alerts when usage reaches critical levels.  1. **Documentation of New Processes**  * Update documentation to include new processes (workspace cleanup, Backup and restore, and testing), providing a quick reference for team members. | 1. **Automated Workspace Cleanup**  * Implemented automatic workspace cleanup post-builds to manage storage efficiently.  1. **Backup and Recovery**  * Configured regular backups using Thin Backup for Jenkins configurations, plugins, and jobs. * Documented a simple recovery process for easy restoration.  1. **Testing Integration**  * Developed unit and integration tests for core functionality, including the Navbar component and essential interactions. * Integrated these tests into the Jenkins pipeline to automatically validate code quality on each push, with test results viewable directly in Jenkins.  1. **Disk Monitoring**  * Installed and configured the Disk Usage plugin to monitor storage and set alerts for critical levels.  1. **Documentation of New Processes**  * Updated and organized documentation for recent Jenkins configurations, providing team members with a clear reference guide. * Documented key processes, including automated workspace cleanup, backup and recovery procedures with Thin Backup, and testing integration with Jest. |

**Sprint 3:** Planned and Completed Tasks (11/20/2024 to 12/17/2024)

| **Planned Tasks** | **Completed Tasks** |
| --- | --- |
| 1. **Slack Notifications Enhancement**    * Add detailed error messages or log excerpts in Slack notifications for build failures to help the team debug issues faster without checking Jenkins manually.    * Customize the notifications to include relevant links (logs or console output). 2. **Automatic Schema Validation**    * Implement automatic schema validation to ensure the database schema matches the expected state during deployments. 3. **Custom Metrics Dashboard**    * Implement a metrics dashboard using tools like Prometheus and Grafana to monitor Jenkins pipeline metrics, including build duration, success rates, and failure trends.    * Visualize the metrics to gain insights into pipeline efficiency. 4. **Pipeline Documentation Update**  * Document newly added processes, such as schema validation, Slack enhancements, and post-deployment validation. * Provide clear instructions, code snippets, and troubleshooting steps for team members to follow. | 1. **Slack Notifications Enhancement**    * Added detailed error messages and log snippets to Slack notifications for build failures, reducing debugging time.    * Customized notifications to include links to logs and console output for easier access. 2. **Automatic Schema Validation**    * Configured “spring.jpa.hibernate.ddl-auto=update” in Spring Boot for automatic schema updates during deployments.    * Ensured schema consistency across environments by verifying updates during application startup. 3. **Custom Metrics Dashboard**    * Developed a Prometheus-Grafana dashboard to monitor pipeline metrics, such as build success rates and durations.    * Integrated real-time metrics for actionable insights into Jenkins pipeline performance. 4. **Pipeline Documentation Update**  * Documented processes for Slack notification enhancements, schema validation, and metrics dashboard setup. * Provided clear, user-friendly instructions with code snippets and troubleshooting tips for team members. |