**CS691 – Jenkins Progress Report**

**Team 4**

**Application: Hearts & Paws**

**Sprint 1: Planned and Completed Tasks**

|  |  |
| --- | --- |
| Planned Tasks | Completed Tasks |
| 1. Jenkins Configuration:  * Updated & Install new plugins  1. AWS Setup – Dev & QA VMs 2. Create two CI/CD pipelines (Deployment to DEV & Deployment to QA) 3. Write and implement Jenkins pipeline script for automated deployment 4. Integrate Slack notifications for real-time build and deployment alerts 5. Automate tasks and stages within the Jenkins pipeline to streamline code deployment and testing workflows 6. Configure environment variables to optimize pipeline execution and ensure consistency across deployments 7. Integrate Jenkins with Git for seamless CI/CD workflow, connecting the pipeline to a specific branch and maintaining the Jenkins pipeline script within the Git repository | 1. Completed with all Jenkins Configurations. 2. Completed with AWS setup 3. Completed creating two CI/CD pipelines. 4. Implemented Jenkins scripts for automated deployment 5. Completed slack notifications Integration for real-time built & deployment alerts 6. Completed writing stages in scripts for deployment and testing workflow 7. Completed creating all required environment variables to optimize pipeline execution and ensure consistency across deployments 8. Completed integrating Jenkins with Git for CI/CD workflow and successfully connected with specific branch and maintaining the Jenkins pipeline script within the Git repository. |

**Sprint 2: Planned and Completed Tasks**

|  |  |
| --- | --- |
| Planned Tasks | Completed Tasks |
| 1. Install the BlueOcean plugin for an improved visual interface in Jenkins, providing a streamlined view of pipeline stages and enhancing team collaboration with a more intuitive layout. 2. Manage and Update Jenkins Plugins:  * Regularly update plugins for security, new features, and stability.  1. Document Initial Challenges  * Record and address initial challenges in automating deployments and sending Jenkins alerts.  1. Testing Integration  * Configure Jenkins to display test results using junit or direct Jest output, allowing quick review of pass/fail statuses and error logs.  1. Git Push Notification on slack  * Setup webhook where We can see all notification in our slack channel when PR, PUSH and commit is done in Main branch. * Integrate JIRA with Jenkins where it will notify with task name, status and branch name when PUSH code to git. | 1. Completed installing the BlueOcean plugin for enhanced visual interface in Jenkins, enabling intuitive pipeline visualization and improved team collaboration. 2. Completed regular updates and management of Jenkins plugins to ensure security, stability, and availability of the latest features. 3. Documented initial challenges faced in automating deployment and sending Jenkins alerts; created internal documentation to address and resolve these issues. 4. Testing Integration Setup Done on Branch-QA. Implementation of all test cases remaining will be done in sprint 3. 5. Completed Git Push Notification on slack:  * Completed webhook setup where We can see all notification in our slack channel when PR, PUSH and commit is done in Main branch * Integrated JIRA with Jenkins where it will notify with task name, status and branch name when PUSH code to git. |

**Sprint 3: Planned and Completed Tasks**

|  |  |
| --- | --- |
| Planned Tasks | Completed Tasks |
| 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  * 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. 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. | 14. Backup and Recovery   * Completed setting up automated scheduled backups for Jenkins configurations, plugins, and job data, ensuring full server recovery readiness. * Created and shared a step-by-step internal guide for restoring Jenkins from backups, enabling the team to independently perform recovery when needed.   15. Testing Integration   * Completed writing and integrating automated unit tests for core React components using Jest and React Testing Library. * Configured Jenkins pipelines to display detailed test results using JUnit reports, streamlining pass/fail visibility for faster troubleshooting.   16. Pipeline Documentation Update   * Completed documentation of all newly added Jenkins processes, including schema validation setups, Slack integration enhancements, and post-deployment validation workflows. * Provided comprehensive step-by-step instructions, sample code snippets, and troubleshooting tips for team members, ensuring smooth adoption and knowledge sharing. |