

Continuous Integration and Continuous Deployment Jenkins Job Description

Since this is a prototype and there are only a handful of jobs, a manual chaining of jobs was used to automate the CI process.

Following are the Jenkins Jobs Used

1. EZRxLookup

This is the primary job that polls the GitHub every 10 minutes to check for changes to the master branch. If any changes are checked in, the job will do the following Gradle tasks

1. Clean
2. Build
3. Run the Unit tests
4. Run the Integration tests
5. Run the code coverage report – coberturaReport
6. Run the code analysis report – SonarRunner
7. If all these are successful archive the WAR artifact and archive the workspace to be cloned by dependent jobs
8. Deploy the build to Stage to enable testing from different devices
9. If the deploy is successful, kick off EZRxLookupUITests Job
10. Send email notifications if any of the above steps fail to the dev group and the initiator(s) of change.

2. EZRxLookupUITests

The main purpose of this Job is to automate the UI testing. The following things are accomplished by this Job.

1. The Job clones the workspace from the previous job
2. Launches a headless browser - Xvfb
3. Since this is a prototype application, the application is tested using a headless firefox browser only. The UI test cover the main business functionality. This is accomplished by executing the webTests target of the gradle
4. Send email notifications if any of the above steps fail to the dev group and the initiator(s) of change

3. EZRxLookup-Perf-Test

The main purpose of this job is to check the performance of the application. This Job monitors EZRxLookupUITests and if it was successful, it will launch itself and accomplish the following tasks

1. The Job clones the workspace from EZRxLookup
2. Run the performance testing using jmeter
3. Publish the reports

4. Send email notifications if any of the above steps fail to the dev group and the initiator(s) of change

4. EZRxLookupDeployToCert

The main purpose of this job is to deploy the last stable build to cert environment on-demand. This would allow the testing team and users to test the app in a stable environment.

1. The Job clones the workspace from EZRxLookup
2. Deploys the WAR to the cert env
3. Archive the workspace to be cloned by dependent jobs
4. Send email notifications if any of the above steps fail to the dev group and the initiator(s) of change

5. EZRxLookupDeploy

This is a job to manually build and deploy to stage environment