ULTIMATE CI/CD PIPELINE | JENKINS END TO END PROJECT ||#devopsinterviewquestions |
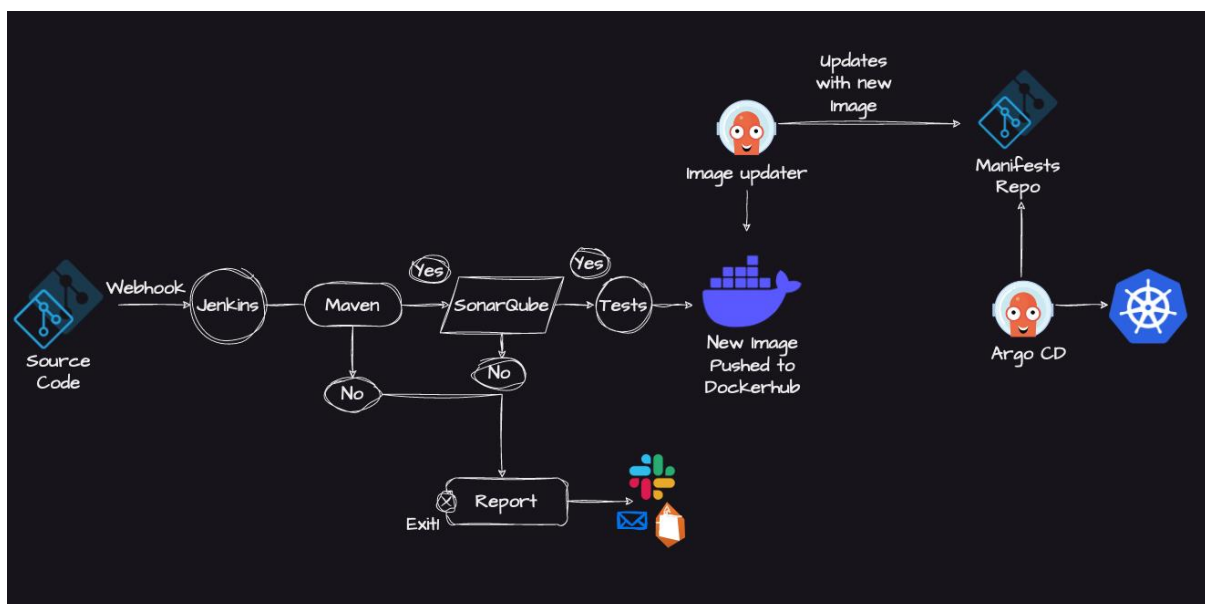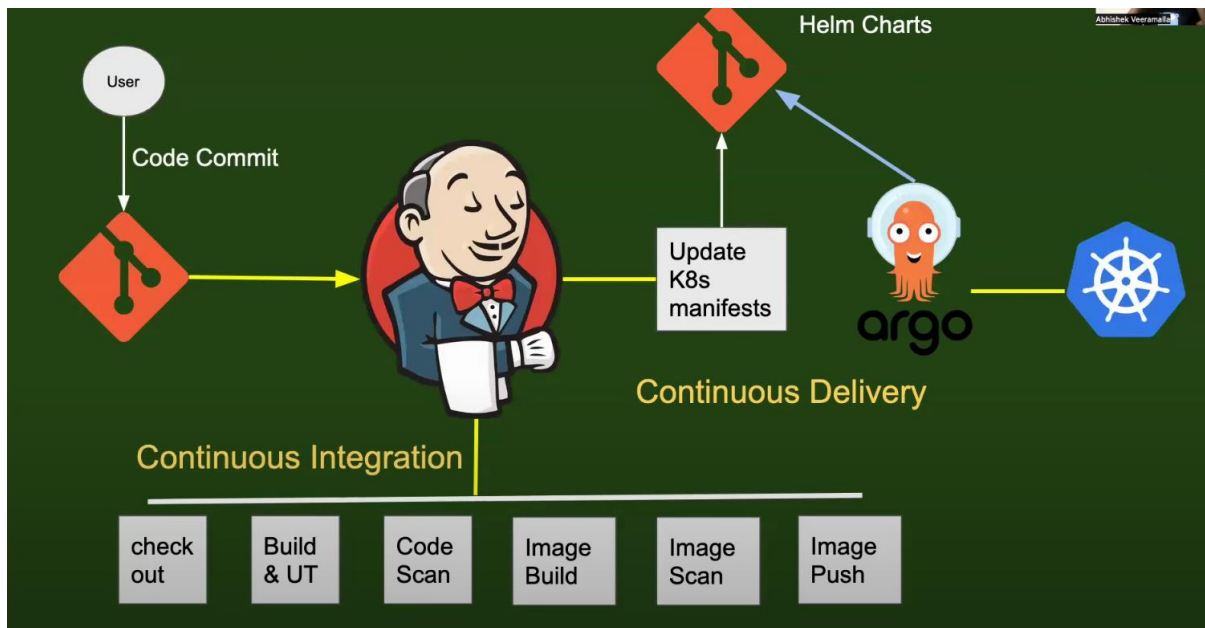

CI/CD Architecture:

- DevOps Engineer who installs and maintains Jenkins will create a Jenkins file, using plugins installs maven to build the code.
- At each of the stages we can use Docker Agent we can skip the installation due to the images we can automate.
- At build stage unit test and code quality tests are done, if it passes it goes forward otherwise won't.
- We can also create an alerts that if we get an errors we can get an email.
- SonarQube is used to check security vulnerabilities which can lead to hacking.
- Next docker we can use docker agent/plugin we make it build the image and push it to docker registry.
- When we use docker agent we don't to install everything individually.

- All this process we have came across is known as the Continuous Integration.
- The last step in CI involves building and pushing the image to docker container registry. Now to connect to CD process we are using GitOps tool, ArgoCD Tools.
- GitOps can be another source code repository it helps in application manifest by including pod.yml, service.yml, deployment.yml or Helm Charts. If we make a volume mount to the existing one Without using this DevOps engineer need to make change separately and main thing is there is no verification.
- This can be known as application manifest git repository.
- We have Argo Image Updater which will monitors and checks the Docker container registry of CI, whenever the image used in CI process is update in the registry it will come to git so that the change can be updated in Helm charts, pod.yml, deployment.yml, service.yml.
- When a change is pushed to GitOps repository it will update the application manifest. It is updated by Argo Image Updater.
- ArgoCD will be update and push the change into the k8's Cluster.
- ArgoCD which will be sitting inside the k8's cluster, which is a k8's controller which it does is it always try to make a state b/w your git repo and k8's cluster.
- ArgoCD pick the changes from repo and deploy them into k8's cluster.

Interview Questions:

1. If there is a pull request how Jenkins will notify?

- Basically, Jenkins will not look at GitHub, we have a webhook in GitHub**, From Jenkins take the webhook URL and paste in GitHub settings and mention the triggering points such as commit, push, pull request.**

2. Explain the CI process used in the project step by step?

- **Jenkins is the CI tools used.**
- As I have pushed the code to GitHub repository. And created a webhook to maintain the connection between Jenkins and webhook. By this any change made in the code by pushing, committing, pulling will trigger the Jenkins pipeline.
- I have used a declarative pipeline for better and easy way of understanding.
- Next we perform a few steps that include static code analysis, code quality, unit tests. By using maven, SonarQube plugins and SAST/SDAST.
- Here we configure an email notification if we get any errors/failures. For an alert notification we can use slack.
- Now we can docker registry like docker hub to push the image that we built.
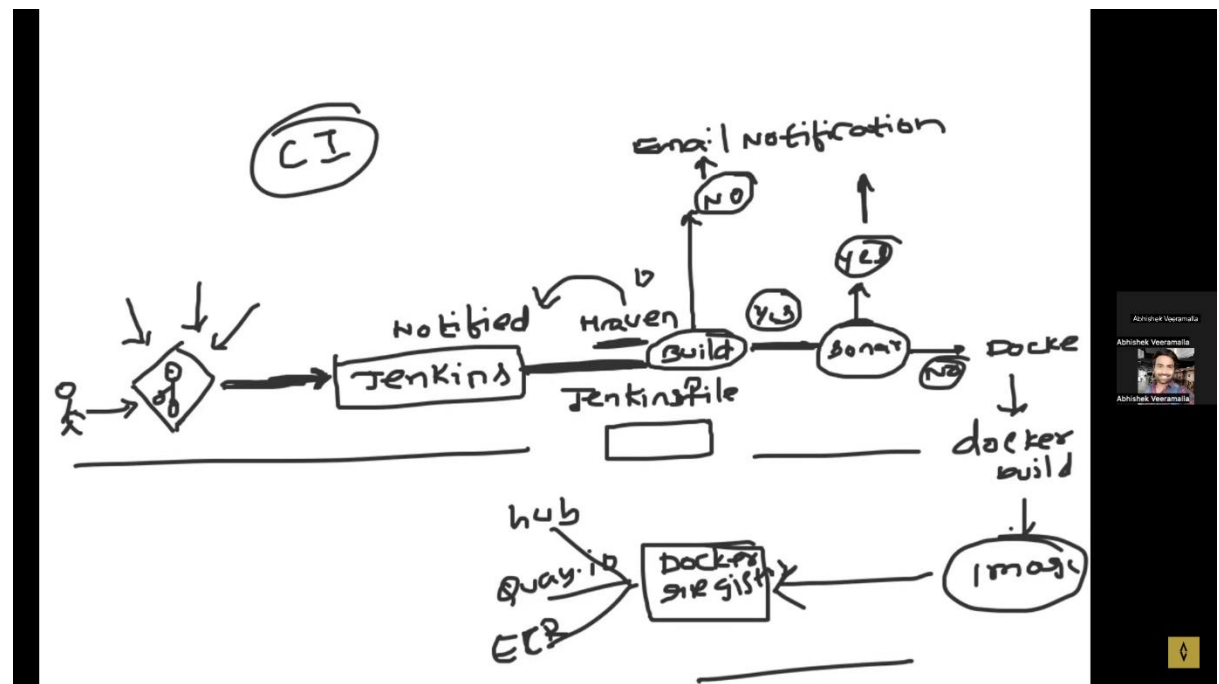
3. Explain the CD process used in the project step by step?

- **All this process we have came across is known as the Continuous Integration.**
- **The last step in CI involves building and pushing the image to docker container registry. Now to connect to CD process we are using GitOps tool, ArgoCD Tools.**
- GitOps can be another source code repository it helps in application manifest by including pod.yml, service.yml, deployment.yml or Helm Charts.
- If we make a volume mount to the existing one Without using this DevOps engineer need to make change separately and main thing is there is no verification.
- This can be known as application manifest git repository.
- **We have Argo Image Updater which will monitors and checks the Docker container registry of CI, whenever the image used in CI process is update in the registry it will come to git so that the change can be updated in Helm charts, pod.yml, deployment.yml, service.yml.**
- When a change is pushed to GitOps repository it will update the application manifest. It is updated by Argo Image Updater.
- ArgoCD will be update and push the change into the k8's Cluster.
- **ArgoCD which will be sitting inside the k8's cluster, which is a k8's controller which it does is it always try to make a state b/w your git repo and k8's cluster.**
- **ArgoCD pick the changes from repo and deploy them into k8's cluster.**