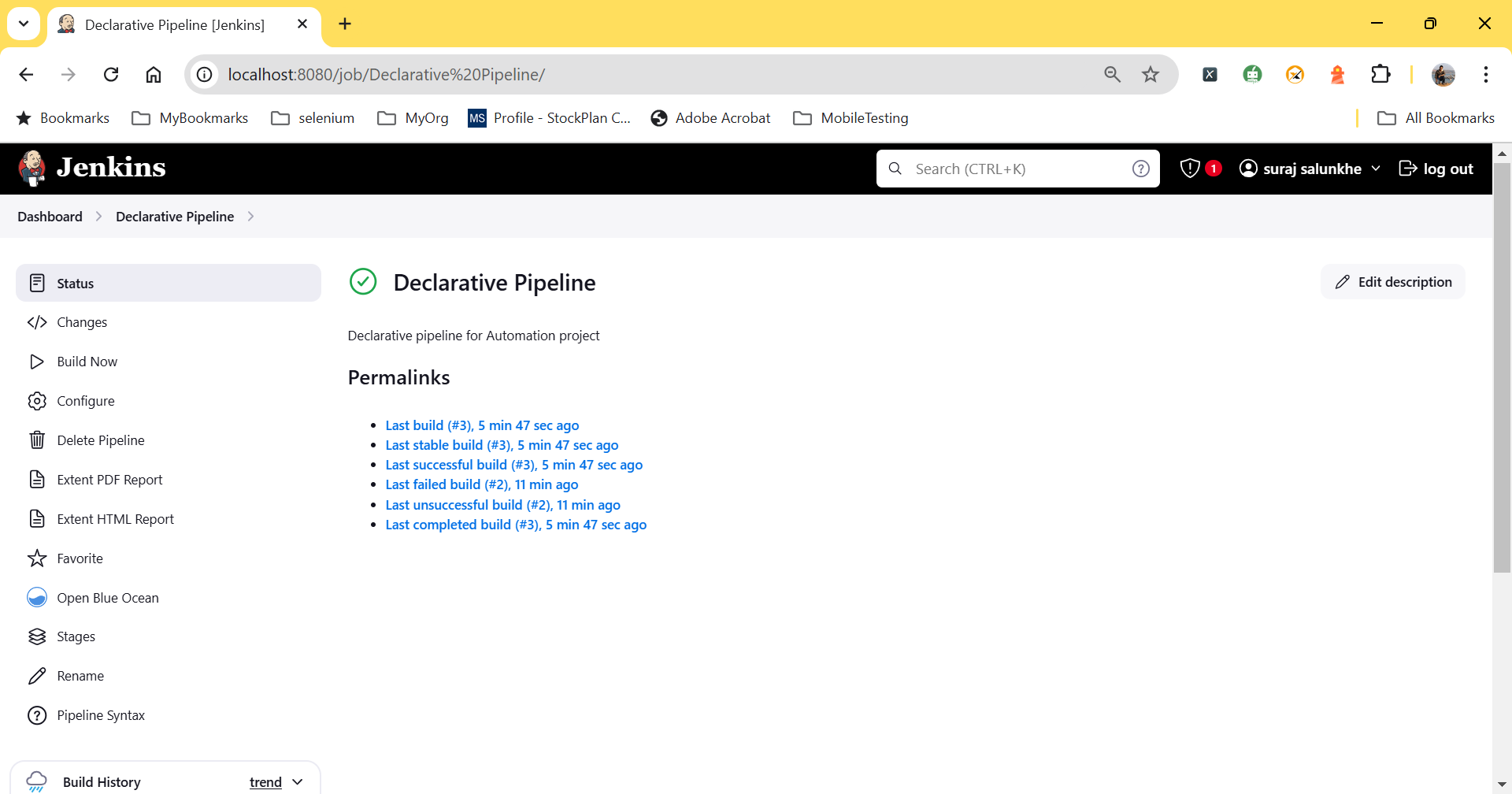
PipeLines:

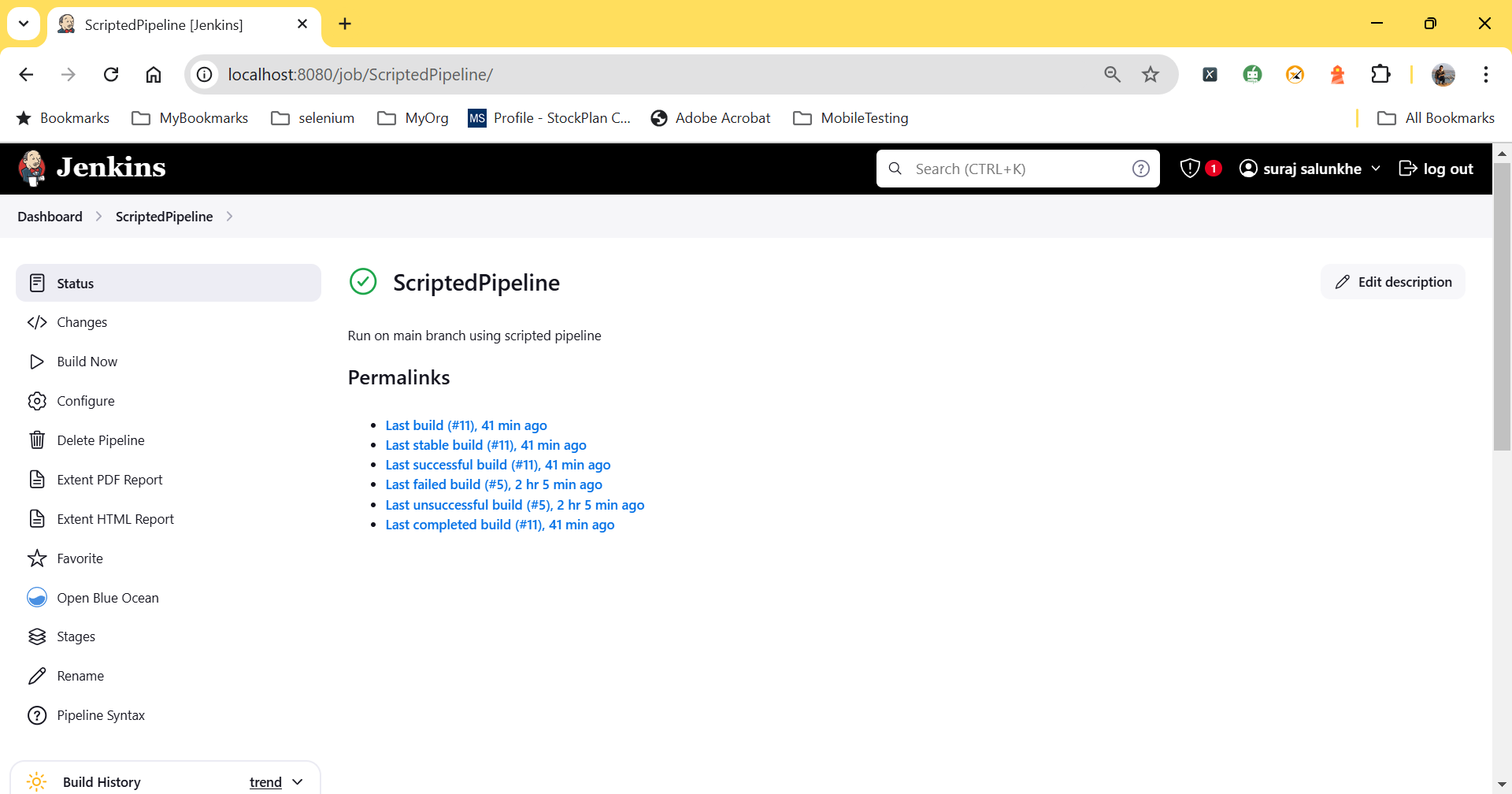
Declarative Pipeline:

https://git.epam.com/suraj\_shivajisalunkhe/testautomationproject/-/blob/develop/Jenkinsfile?ref\_type=heads



Scripted pipeline:

https://git.epam.com/suraj\_shivajisalunkhe/testautomationproject/-/blob/main/Jenkinsfile?ref\_type=heads



Jenkins Question and answers:

**1. What is Jenkins and how is it used?**

Jenkins is an open-source automation server that enables CI/CD in software development.

• Usage: Automates software development, testing, and deployment.

• Works with multiple version control systems and software development tools.

• Assists with automating processes with scripts and commands.

**2. What are the prerequisites for installing Jenkins on your machine?**

• Jenkins requires Java to run. You must have either the JDK or the JRE installed.

• A web browser is required to access the Jenkins web interface after installation.

**3. Why use Jenkins LTS over the most recent Jenkins release? When should I use the most recent release?**

• Favoring LTS:

• LTS versions are reliable and thoroughly tested, making them ideal for production environments.

• Long-term support includes updates and security patches over a longer period.

• When to use the most recent release:

• If you require new features or upgrades not currently available in the LTS version.

• Uses cutting-edge features for testing and development.

**4. The distinction between a job/project and a build.**

• A job/project is a setup for a specific Jenkins task, such as building a project.

• Define parameters, triggers, and execution steps.

• Build refers to the precise execution of a job/project.

• The output of a job can include artifacts or logs.

**5. What is a workspace?**

• A workspace is a directory on the Jenkins server that stores the job's files.

• The purpose of this file is to store the necessary source code and files for the job to run successfully. To avoid conflicts, each job is assigned its own workspace.

6. What is a build trigger, and how would you apply it in a job?

A build trigger is an event that initiates a job automatically.

• Usage: • Enables automated builds based on code changes in a version control system.

• Scheduled times (cron jobs).

• Users manually activate events.

**7. What is a parameterized job, and when would you use it?**

• A parameterized task enables users to pass parameters to the job at runtime.

• Usage: • Customize job execution based on input values, including environments (e.g., staging vs. production).

• Specific construction configurations.

**8. What are the definitions of Pipeline as Code and Infrastructure as Code?**

• Pipeline as Code: • Define Jenkins pipelines with code (e.g. Jenkinsfile).

• Enables versioning and simplified management of CI/CD procedures.

• Infrastructure as Code involves managing and provisioning infrastructure using code.

• Automate infrastructure deployment with tools like Terraform or Ansible.

**9. Advantages of using pipelines over standard build jobs in Jenkins**

• Store pipelines in version control systems for better management.

• Supports sophisticated workflows with parallel execution and conditional phases.

• Reusability: Code can be reused for multiple projects.

• Visualization: Enhances understanding of the building process.

**10. What is a Jenkins file?**

A Jenkinsfile is a text file that defines a Jenkins pipeline.

• The purpose of this document is to outline the stages, steps, and environment for the CI/CD process, which allows for version control and easy upgrades.

**11. Differences between declarative and scripted pipelines**

• The Declarative Pipeline uses a simpler, more structured syntax.

• Easier to read and write, emphasizing "what" rather than "how."

• Scripted Pipeline: • Provides greater flexibility and control through standard programming constructs. • Can be complex and difficult to maintain.

**12. Benefits of using Blue Ocean.**

• Jenkins features a user-friendly interface that is both modern and intuitive.

• Pipeline Visualization: Enhances understanding of pipeline stages.

• Simplified Configuration: Allows for easy pipeline creation and management without prior Jenkins expertise.

**13. What is a multibranched pipeline?**

• A multibranch pipeline automatically generates pipelines for each branch in a repository.

• Jenkins can manage several branches and pipelines for a project, making it more efficient.