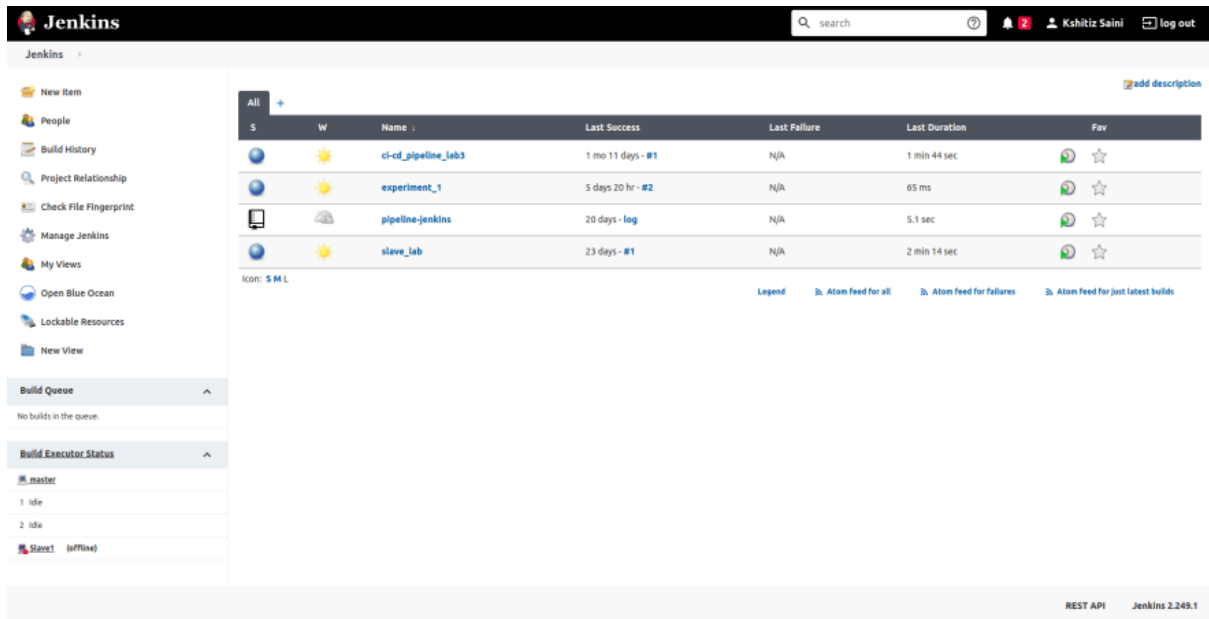


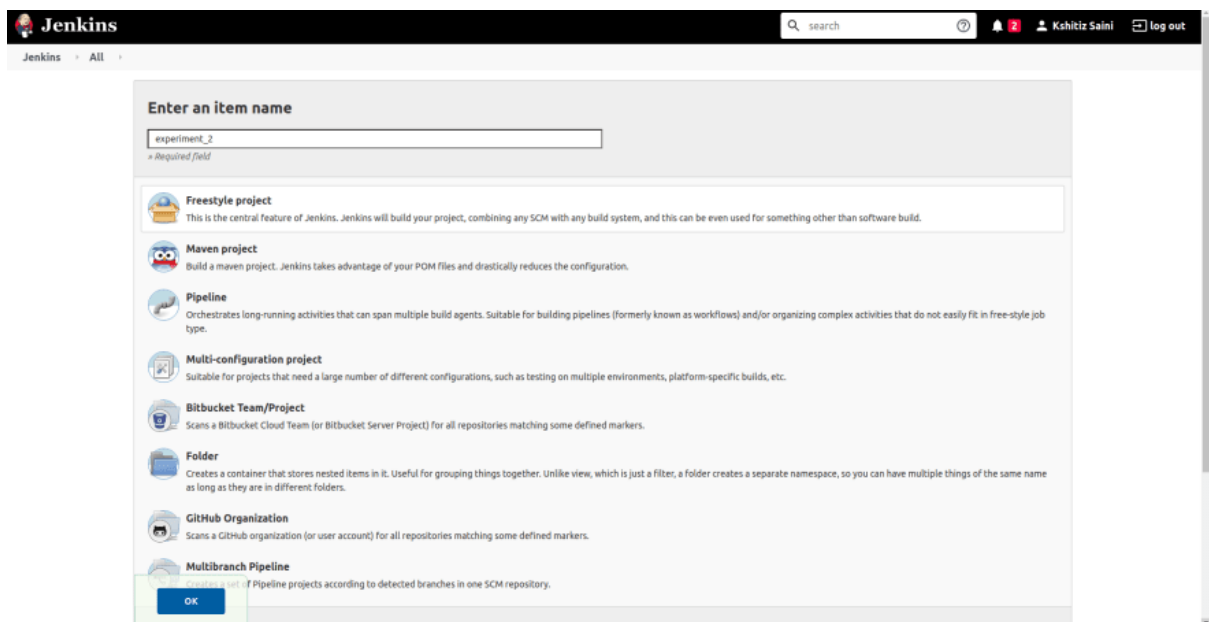
Experiment 2

Integrating Jenkins + GitHub

- 1) First, we need to launch Jenkins in our device using the specified IP and login to the Jenkins server using the credentials.



- 2) Create a new item and select the freestyle project and give it a name.



3) In the settings, we need to go to the Source Control Management option and select Git as our version control repository.

4) Enter the credentials of GitHub (if required) and the URL of the repo to work with.

The screenshot shows the Jenkins configuration page for project 'experiment_2', specifically the 'Source Code Management' tab. The 'Restrict where this project can be run' checkbox is unchecked. Under 'Source Code Management', 'Git' is selected. The 'Repositories' section contains a table with one entry: 'Repository URL' is 'https://github.com/kshitzsaini113/TCP-IP-Stack-Functionality.git' and 'Credentials' is set to '- none -'. Below this, the 'Branches to build' section has a 'Branch Specifier (blank for \'any\')' set to '*/master'. The 'Repository browser' is set to '(Auto)'. At the bottom, there are 'Save' and 'Apply' buttons.

5) Click on Save and Apply.

The screenshot shows the Jenkins main dashboard for project 'experiment_2'. The top navigation bar includes the Jenkins logo, a search bar, and user information 'Kshitz Saini' with a 'log out' link. The left sidebar contains a list of actions: 'Back to Dashboard', 'Status', 'Changes', 'Workspace', 'Build Now', 'Configure', 'Delete Project', 'Favorite', 'Open Blue Ocean', 'Rename', 'Build History' (with a 'trend' link), and a 'find' input field. The main content area displays 'Project experiment_2' with links to 'workspace' and 'Recent Changes', and a 'Permalinks' section. On the right, there are links to 'add description' and 'Disable Project'. The footer shows 'REST API' and 'Jenkins 2.249.1'.