

What is the exploratory testing you would perform in 30 minutes:

- Check that file uploads should check the extensions that are allowed.
 - For example, "Check File Fingerprint" should only allow to upload jar files. Any other file extension should be avoided.
- Check that same context is consistent between Jenkins and Blue Ocean views:
 - For example you are running the "ATH-Jenkinsfile" job in Jenkins. So switching to Blue Ocean dashboard should take you to the same job.
 - Taking you to the TOP view would be a bad user experience.
- Check that any kind of "SQL injection" is harmless for any input fields.
 - Inputs like "Search Box" or "Project relationship" should filter this kind of injections.
 - Some examples of injections are:
 - "" or 1==1;--"
 - "" or select version();--"
 - "" or select * from INFORMATION_SCHEMA.TABLES;--"
- Check that results returned by "Search Box" are consistent between Jenkins and Blue Ocean interfaces.
 - When searching for a pipeline/job in Jenkin, also should appear in then Blue Ocean.
 - Not the case now, given that Blue Ocean returns only jobs/pipelines and Jenkins search returns in addition folder and users.
- Check that logged-in user is required to manage some job/user settings.
 - Given limited access of the read-only account. It is needed to be logged-in to flag a job/pipeline as favourite.
Also settings should be stored across successive login sessions.
 - User settings like case sensitive on search should be met.
- Login session should be consistent with both Jenkins and Blue Ocean interfaces:
 - For example if you are logged-in in Jenkins and you logged-out would remain in Jenkins view.
 - But if you're being logged-in in Blue Ocean and when you logout you get redirected to Jenkins. That would cause a bad user experience.

If you had more time, what are the things that you would test for?

- Check exhaustively that all jobs/pipelines existing in jenkins are also available via Blue Ocean view.
- Check more extensively file upload functionality:
 - "Check File Fingerprint" should check for maximum size permitted to upload.
Also check that any compressed file upload contains a valid file, otherwise would be refused.
- Check that pagination is used to retrieve long amount of records.
 - For example "Build History" in top level should paginate the results to minimize the impact on the backend when querying for build results.