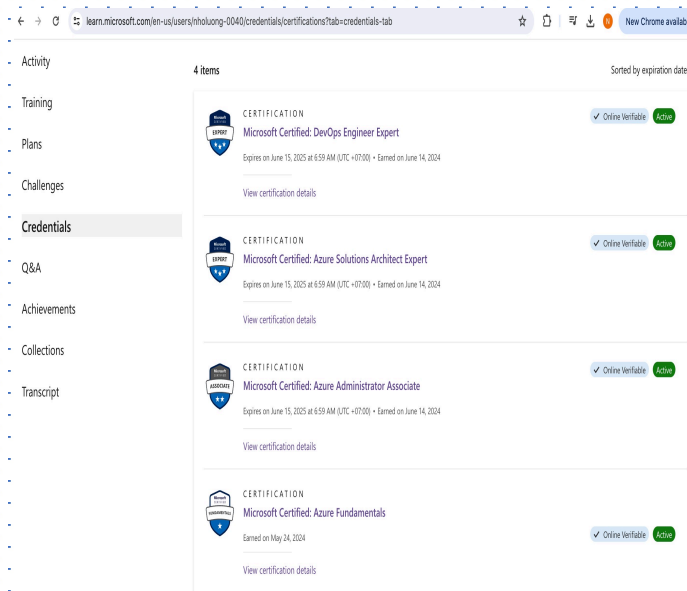
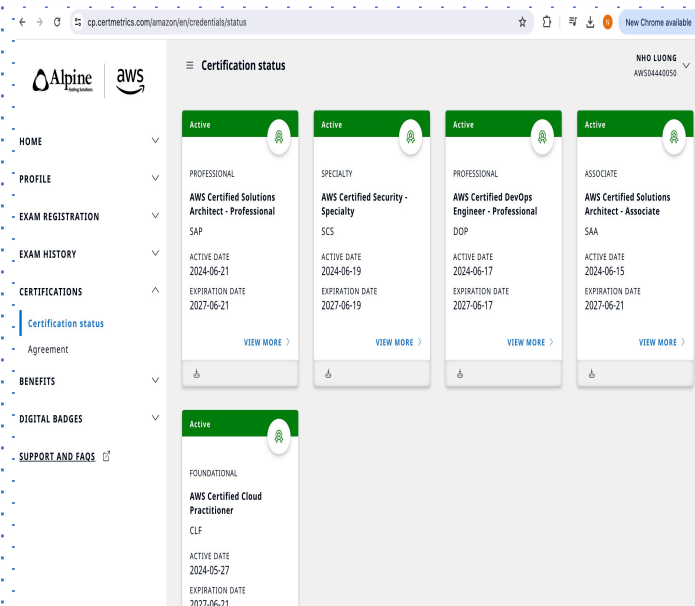


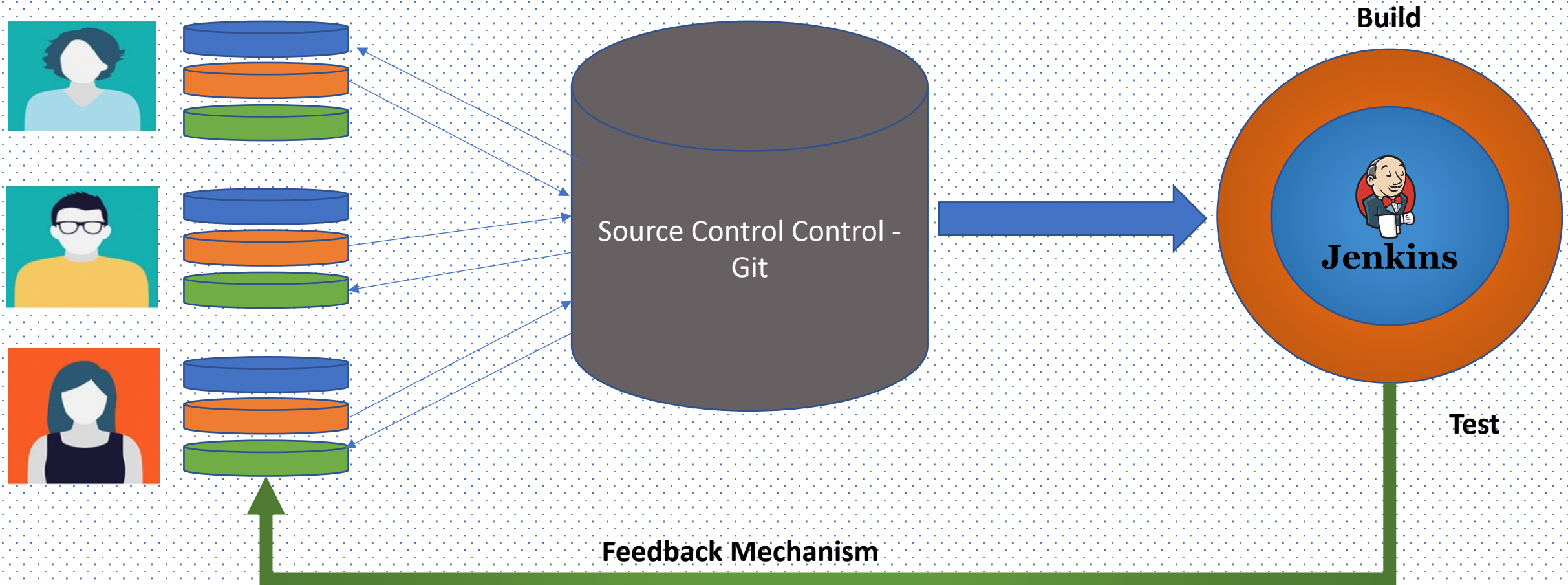
Jenkins Overview

Author: Nho Luong

Skill: DevOps Engineer Lead



Continuous Integration



Continuous Integration

- Continuous Integration (CI) is a development practice that requires developers to integrate code into a shared repository several times a day. Each check-in is then verified by an automated build, allowing teams to detect problems early.
- By integrating regularly, you can detect errors quickly, and locate them more easily.

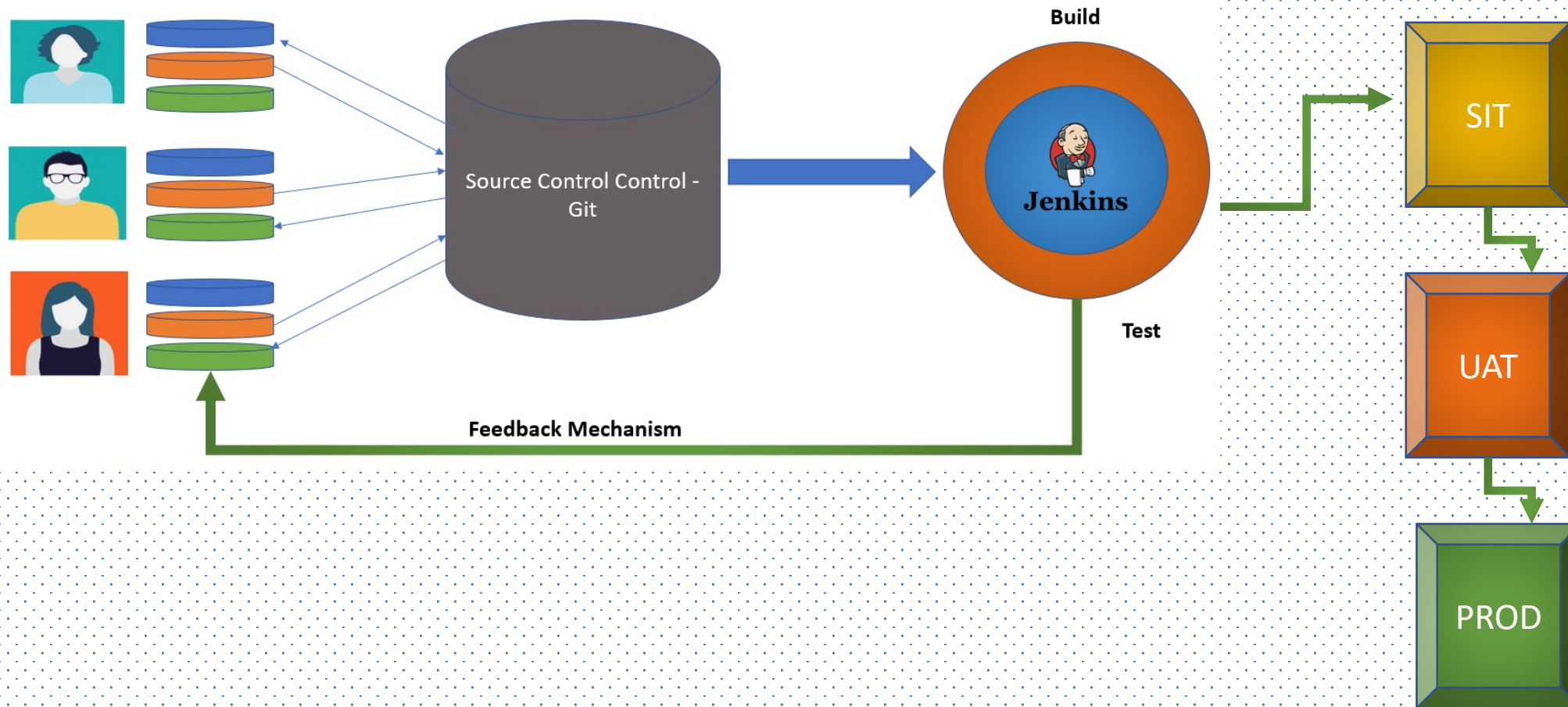
Why Continuous Integration

- If developers do not commit the changes on regular basis then code changes has to be integrated and tested manually.
- If developers commit the changes on regular basis like daily or weekly and Build script compiles the code and runs the test script automatically then on regular basis the integration happens but still it is not completely continuous integration because the changes get compiled on regular basis not immediately and incase of Build failure the notification sent back to developer.
- So that is why we need a CI process means whenever developer commits the code into source control, it should be immediately build and incase of failure it sends the notification to developer.

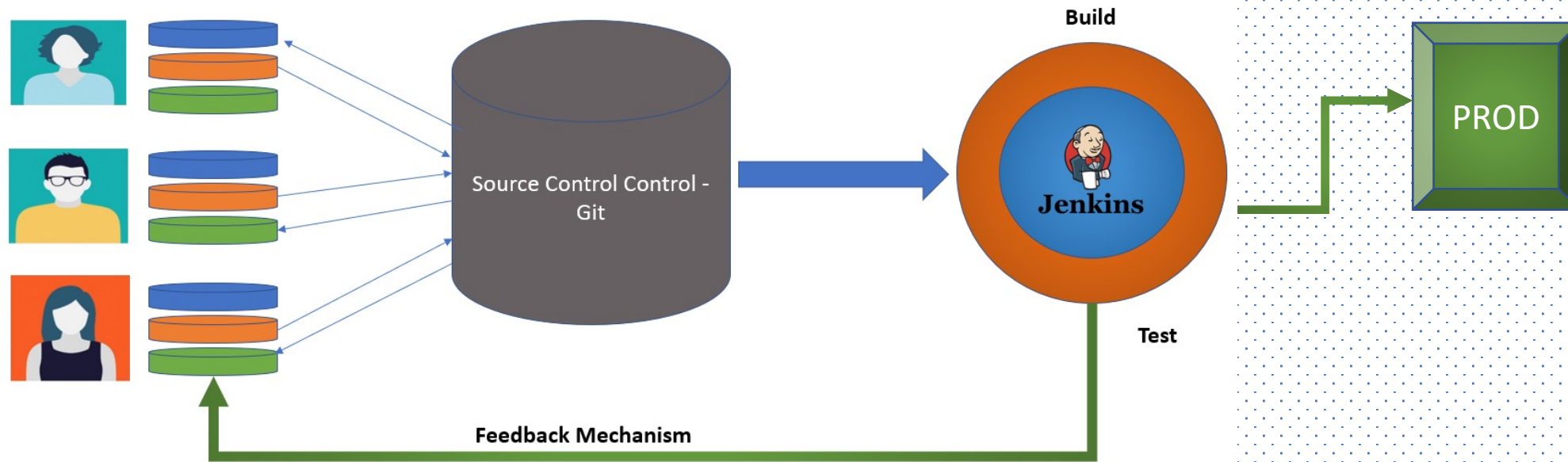
CI-Advantages

- Goodbye to long and tense integration.
- Increase visibility enabling greater communication.
- Catch the issues early.
- Spend less time for debugging and more time on adding more features.

Continuous Delivery



Continuous Deployment



Installing Jenkins

--->Minimum hardware requirements:

- 256 MB of RAM
- 1 GB of drive space (although 10 GB is a recommended minimum if running Jenkins as a Docker container)

---->Recommended hardware configuration for a small team:

- 1 GB+ of RAM
- 50 GB+ of drive space

Software requirements:

- Java 8 - either a Java Runtime Environment (JRE) or a Java Development Kit (JDK) is fine



Thank You