

A dark blue vertical bar on the left side of the page. A blue arrow points to the right from the bar, containing the date.

12/3/2023

GitFlow Report

Several thin, curved lines in dark blue and light grey originate from the bottom left corner and sweep upwards and to the right.

Mohammad Khasbah
ATYPON TRAINING

❖Project Overview:

The project focused on the collaborative development of a calculator application using a text-based format (.txt files) as a representation of various functionalities. Two developers, Mohammad and Ali, worked on different features, and the workflow involved the implementation of GitFlow for version control and collaboration.

❖Implementation of GitFlow:

1. Branching Strategy:

Developers cloned the repository and followed a features

Each developer worked in their respective branches for adding features, such as the equation solver and significance graph.

The develop branch served as the main development branch, with feature branches created from it.

2. Release Management:

Release branches were created for versioning and preparing the application for deployment.

Challenges: Managing versioning and preparing the release required careful coordination. Conflicts in version numbers and potential merge conflicts needed attention.

Solutions: Regular communication and following a structured release process helped in mitigating conflicts and ensuring a smooth release.

3. Hotfix Handling:

A hotfix branch was used to address critical issues, such as the division by zero problem.

Challenges: Hotfixes needed to be deployed quickly without disrupting ongoing development.

Solutions: The use of a separate hotfix branch allowed for immediate resolution without impacting the ongoing work in the develop branch.

❖ Learning Experience and Insights:

1. Structured Workflow:

GitFlow provided a structured workflow that facilitated collaboration and organization.

Insights: The clear branching strategy and defined roles in GitFlow made it easier to manage concurrent development tasks.

2. Versioning Challenges:

Coordinating version numbers and managing release branches required attention.

Insights: Versioning challenges highlighted the importance of communication and adherence to a versioning strategy to avoid conflicts.

3. Quick Hotfix Deployment:

The ability to quickly create a hotfix branch and address critical issues was valuable.

Insights: GitFlow's flexibility in handling urgent fixes demonstrated its effectiveness in real-world scenarios.

4. Collaborative Development:

Multiple developers working on separate features simultaneously showcased the collaborative strength of GitFlow.

Insights: GitFlow promotes a collaborative environment, but effective communication and coordination are crucial to managing concurrent changes.

❖ Conclusion:

The GitFlow workflow proved to be an effective strategy for managing collaboration, versioning, and hotfixes in a simulated team environment. Challenges were addressed through communication and adherence to the GitFlow model. The experience highlighted the importance of structured workflows and effective collaboration in software development.