

## **GIT & GITHUB Essentials**

*Date of the Event:* October 30, 2024

**1. Introduction** The GitHub Essentials workshop was conducted to provide participants with a comprehensive understanding of GitHub's features, tools, and practical applications in software development. The resource person explained the importance of GitHub as a collaborative platform for developers, allowing for version control, project management, and teamwork. The session began with an introduction to GitHub and its role in managing code repositories. The participants were also introduced to the concept of repositories, branches, commits, and pull requests.

**2. Setting Up GitHub** The resource person guided the participants through the process of setting up GitHub accounts. Detailed instructions were provided on how to create an account, configure user profiles, and link the local system to GitHub using SSH keys. The participants learned how to install Git on Windows, macOS, and Linux operating systems. Emphasis was placed on configuring Git with a username and email to track code changes effectively. The instructor also demonstrated how to create local and remote repositories.

**3. Repository Management** This segment of the session focused on managing repositories effectively. The resource person explained how to create new repositories and the difference between public and private repositories. Participants were taught how to clone existing repositories, create branches, and switch between them. The concepts of commits and commit messages were emphasized as essential practices for maintaining clear version history. Practical demonstrations were given on pushing changes from a local repository to a remote GitHub repository.

**4. Branching and Merging** The workshop introduced the concept of branching and its role in collaborative development. The resource person demonstrated how to create branches, switch between branches, and make changes without affecting the main branch. The importance of merging branches into the main branch was highlighted, and participants were shown how to resolve merge conflicts. Hands-on exercises were provided to help participants understand branching, merging, and pull requests.

**5. Collaborating on GitHub** Collaboration was a key focus of the workshop. Participants were introduced to the process of forking repositories and creating pull requests. The resource person explained how to review and merge pull requests to ensure code quality. Strategies for team collaboration, such as assigning reviewers, using issues, and creating project boards, were discussed. Participants also learned about using GitHub Discussions to communicate and collaborate effectively with team members.

**6. Working with Git Commands** The resource person provided a hands-on demonstration of essential Git commands used in version control. Key commands such as `git init`, `git clone`, `git add`, `git commit`, `git push`, and `git pull` were explained in detail. Participants practiced using these commands to create and manage repositories. The session emphasized the use of `git status` and `git log` to track repository status and changes made.

**7. Managing Issues and Projects** Participants were introduced to the concept of issues and project boards as tools for project management. The resource person explained how to create and manage issues, assign them to team members, and track their progress. Project boards were introduced as a visual tool to organize tasks using Kanban-style boards. Participants practiced creating issues, linking them to pull requests, and tracking project progress.

**8. Advanced Concepts** In the final segment, the resource person introduced advanced concepts like GitHub Actions, automation workflows, and security best practices. Participants were exposed to Continuous Integration/Continuous Deployment (CI/CD) pipelines, enabling automatic testing and deployment. The importance of protecting branches, using code owners, and managing repository access rights was also highlighted. The instructor shared insights on best practices for maintaining security and privacy while working on open-source and private repositories.

**9. Conclusion** The workshop concluded with an interactive Q&A session where participants clarified their doubts about GitHub usage. The resource person reiterated the importance of GitHub in modern software development and encouraged participants to practice regularly. Participants were advised to explore GitHub documentation and engage in open-source contributions to gain hands-on experience. The session was well-received, with participants gaining confidence in using GitHub for version control, collaboration, and project management.



**Fig: Brochure of GIT & GITHUB Essentials**