TRAINING TR-102 DAY 16 REPORT

Agile methodology is an iterative approach to project management and software development that emphasizes collaboration, flexibility, continuous improvement, and customer feedback.

ADDITIONAL TECHNOLOGIES EXPLORED:

1. **Scrum Framework:**
   - **Scrum Roles:** Product Owner, Scrum Master, Development Team.
   - **Scrum Artifacts:** Product Backlog, Sprint Backlog, Increment.
   - **Scrum Events:** Sprint Planning, Daily Stand-ups, Sprint Review, Sprint Retrospective.
2. **Kanban Methodology:**
   - **Visual Board:** Workflow visualization using columns (e.g., To Do, In Progress, Done).
   - **Work-in-Progress (WIP) Limits:** Managing and optimizing flow.
   - **Continuous Delivery:** Incremental improvements and frequent releases.
3. **Agile Tools and Techniques:**
   - **User Stories:** Describing requirements from an end-user perspective.
   - **Burndown Charts:** Tracking progress towards completing work in a sprint.
   - **Pair Programming:** Collaborative coding technique for improved code quality.

INTRODUCTION TO GITHUB

OVERVIEW OF GITHUB

GitHub is a web-based platform and version control system (VCS) used for code hosting, collaboration, and project management. It utilizes Git, a distributed version control system, to track changes in code repositories.

1. **Cloning a Repository:**
   - Creating a local copy of a remote repository on your computer using the `git clone` command.
2. **Pushing Changes:**
   - Uploading local changes to a remote repository on GitHub using `git push`.
3. **Merging Changes:**
   - Combining changes from one branch into another branch (e.g., merging a feature branch into the main branch) using pull requests (`git merge`).
4. **Forking a Repository:**
   - Creating a personal copy (fork) of someone else's repository to make changes without affecting the original repository. This is often used for contributing to open-source projects.
5. **Pulling Changes:**
   - Fetching and integrating changes from a remote repository into your local repository using `git pull`.

GITHUB WORKFLOW:

- **Branching Strategy:**
  - Creating and managing branches for different features or fixes (`git branch`, `git checkout`).
  - Using feature branches to isolate changes and collaborate on specific tasks.
- **Collaboration and Code Review:**
  - Initiating and participating in pull requests (PRs) for proposing and reviewing changes.
  - Discussing and iterating on code improvements through comments and feedback.

CONCLUSION

Day 16 of training covered essential technologies related to Agile methodology, focusing on Scrum, Kanban, and Agile tools. Additionally, participants gained hands-on experience with GitHub, mastering key operations such as cloning, pushing, merging, forking, and pulling in a collaborative development environment