Internship Report – Day 5

Today was my fifth day of internship at Surfboard Payments, and I learned many important concepts related to teamwork and GitHub version control. These skills are essential for working in a team environment

In the morning, my supervisor explained the different types of team players and how their qualities impact team player. He divided team members into three main categories : Humble, Hungry , Smart

Accidental Mess Makers (Humble + Hungry): Hardworking but may cause issues due to lack of strategic thinking.

Stackers (Humble + Smart): Trustworthy and reliable but may hesitate to take the lead.

Skillful Politicians (Hungry + Smart): Strategic and results-driven but may prioritize personal goals over team success.

Understanding these roles helped me reflect on my own strengths and weaknesses in a team. It made me realize that balancing humility, ambition, and intelligence is crucial for being an effective team player.

In the afternoon, we had a session on GitHub and version control, which completely changed my understanding of GitHub. Before today, I thought GitHub was only for storing code, but now I know that it plays a crucial role in collaborative software development.

One of the most important things I learned was about branches. Branches allow developers to work on different features separately without affecting the main

project. Once the changes are tested and reviewed, they can be merged into the main branch, ensuring that the main project remains stable.

I also learned about pull requests, which are used to review and approve changes before merging them into the main project. This ensures that the code is error-free and follows best practices.

During the session, I practiced using essential Git commands, including:

git init – Initializes a new Git repository.

git clone – Copies an existing project from GitHub.

git add – Stages changes before committing them.

git commit -m "message" – Saves changes with a message.

git push – Uploads local changes to GitHub.

git pull – Fetches the latest updates from the remote repository.

git merge – Combines changes from different branches.

I also learned about merge conflicts, which occur when two developers edit the same part of a file. GitHub provides tools to compare, resolve, and merge these changes correctly. Understanding how to handle merge conflicts is important for ensuring a smooth workflow in a team project.

Another key takeaway was the importance of writing clear commit messages. A well-written commit message helps team members understand what changes were made, making it easier to track updates and debug issues if needed. Our

instructor recommended using Conventional Commits, which provide a structured format for writing commit messages.

I also explored pull requests and code reviews, which are commonly used in real-world projects. Instead of making direct changes to the main branch, developers submit a pull request, allowing teammates to review the code, suggest improvements, and approve it before merging. This process improves code quality and prevents mistakes.

Today's session helped me realize that GitHub is not just about writing code—it is about teamwork and collaboration. It ensures that multiple developers can work on the same project without errors or overwriting each other’s work.