**BACHELOR OF SCIENCE IN INFORMATION TECHNOLOGY**

**ON-THE-JOB TRAINING**

**DAILY ACTIVITIES**

|  |  |  |  |
| --- | --- | --- | --- |
| **NAME: DOMINGO, JOHN KENNETH NICKO M.** | | **INTERSHIP SITE: WEAVERSOFT** | |
| **DATE** | **ACTIVITIES PERFORMED** | **PROBLEMS ENCOUNTERED** | **ACTION TAKEN** |
| March 3, 2025 | * Created a new pull request for the Create New Task feature after it was declined by Sir Artem, despite prior approval from the lead programmer. * Identified that the pull request was rejected due to large logic inside setState and modifying working code without a valid reason. * Rewrote the code to align with company standards. * Replaced the switch with a button for copying reference data. * Used a button instead of a switch because it was the initial requirement and using a switch would require a major update due to its complex logic. * Removed the bug fixes since they were not allowed in the same pull request. * Added a new feature preventing the accordion from opening if there was no reference data. * Reduced the pull request size from 150 insertions and 85 deletions to 38 insertions and 4 deletions. | * The previous pull request was declined due to improper coding practices. * Bug fixes were mistakenly included in the pull request, which was against company guidelines. | * Rewrote the code to follow the company standards and ensured compliance with best practices. * Used a button instead of a switch to meet the initial requirements without introducing complex logic. * Removed unrelated bug fixes from the pull request. * Implemented a new feature preventing the accordion from opening without reference data. * Optimized the code, reducing the number of insertions and deletions significantly. |
| March 4, 2025 | * Received a new task but found that it lacked a clear description and context. * Asked for clarification, but there was no reply as the senior developers seemed busy. * Studied Next.js to prepare for upcoming tasks and improve development skills. * Created a CRUD application using Next.js and Supabase with Server-Side Rendering (SSR). * Implemented optimistic UI updates with rollback to minimize delays. * Built a user management system with ShadCN tabs and a settings dialog for editing and deleting users. * Integrated real-time updates to reflect changes instantly. | * The task assigned lacked sufficient details and context. * Could not proceed with the task due to no response from senior developers. | * Used the time to study Next.js and gain hands-on experience. * Developed a CRUD system with Supabase to practice key concepts. * Applied SSR, optimistic UI updates, rollback mechanisms, and real-time updates to enhance performance and responsiveness. |
| March 5, 2025 | * Added Output Policy in the task result. * Edited Reference and Validation to remove fallback messages as per the boss’s request. * Ensured that removing fallback messages would not cause crashes or errors due to undefined values. * Continued studying Next.js to prepare for upcoming tasks and improve development skills. * Applied best practices to improve code structure, optimize performance, and enhance maintainability. * Focused on refactoring existing code to ensure better readability, efficiency, and scalability. * Followed and applied the company's deployment pattern in my own project by creating feature branches, testing changes in the production/preview environment, and ensuring stability before merging them into the main branch as a practice and application of my learning. | * The payload for the task was not specified. * The GET request only returned the ID, requiring additional investigation. | * Traced the data flow by checking the POST request in task creation, then Django’s database insertion, and finally the GET request to understand how Output Policy was handled. * Adjusted the implementation to properly work with the ID-only response. * Ensured that hiding empty fields in Reference and Validation did not cause crashes or errors. |
| March 6, 2025 | * Had no assigned tasks as the senior developers were busy. * Continued reviewing Next.js to prepare for future tasks. * Discovered that the database was not properly indexed, leading to performance issues. * Focused on understanding latency issues and optimizing response times. * Learned about GitHub Actions to prevent Supabase and Vercel from sleeping. * Conducted client-side testing using Lighthouse and identified potential improvements. * Ran multiple latency tests on Supabase requests, local rendering, and Vercel deployment, analyzing response times. * Studied rate limiting to enhance performance and security. |  |  |
| March 7, 2025 | * Reviewed Next.js due to the absence of available tasks in Trello. * Learned about avoiding unnecessary API calls, Server-Side Rendering (SSR), and Server Actions. * Rewatched Clean Code as originally assigned by Sir Artem to ensure adherence to coding standards in future tasks. | * No tasks were assigned since the company does not operate on Fridays, and no pending tasks were left in Trello. | * Used the time to enhance knowledge of Next.js and best practices. * Reviewed Clean Code principles to improve coding standards for upcoming tasks. |