DEMO 5 - NGA NGUYEN

1. Here is the link to the web page created on during the exercise: https://demo-cicd-mnnguy-b4ba1c2f8ddc4cd14f6e07cbcdc3e3393417d7066d04ab.utugit.fi/

2. Analyze how CI/CD pipeline can be utilized in

a) Traditional Software Development Project:

In a traditional software development project, teams often work on large codebases using a step-by-step process.

Multiple developers push their changes regularly to the version control system, triggering the CI/CD pipeline to automatically build and test the application. CI/CD pipelines help ensure that changes made by different developers integrate smoothly and do not break the existing functionality (CI).

This process helps catch bugs and issues early in the development cycle, reducing the likelihood of major problems during later stages.

Once changes pass all tests and quality checks, they can be deployed to staging and production environments, ensuring that new features and updates are delivered to users quickly and reliably. (CD)

b) Maintenance of an Already Published Software Product:

In the maintenance phase of a software product, developers focus on fixing bugs, addressing performance issues, and making incremental improvements. CI/CD pipelines play a crucial role in this phase by automating the process of testing and deploying bug fixes and updates. Developers can quickly identify and fix issues by pushing changes to the version control system, triggering the CI/CD pipeline to run tests and deploy updates. Automated testing ensures that fixes do not introduce new bugs or regressions, maintaining the stability and reliability of the software product. This helps keep the software up-to-date and working smoothly for users without causing disruptions.

c) Development Project of Modern Cloud-Based Software:

In modern cloud-based software development projects, teams often work on different components of the cloud-based software. CI/CD pipelines are essential for automating the deployment of microservices and managing the complex infrastructure required for cloud-based applications. Developers can push changes to individual microservices, triggering automated builds, tests, and deployments for each component. This allows developers to make changes and improvements to the

software regularly and confidently, without worrying about breaking things or causing problems for users.

In all these scenarios, CI/CD pipelines help teams build and maintain software that is reliable, fast, and always up-to-date with the latest improvements.