CI/CD Proposal

Udapeople

What is CI/CD?

Continuous Integration:
It's the process of merging all developers work into a mainline branch multiple times a day.

Continuous Deployment:
It's the process of automatically releasing changes from a repo to production, where it is usable by customers.

Continuous Delivery:
A software engineering approach that employs continuous integration and continuous deployment in order to regularly produce software in short cycles and ensure that such software can be reliably released at any time without manual intervention

Why CI/CD?

- Superior Code Quality
- Reduced Change & Review Time
- Accelerated Release Cycles
- Instant Fault Detection & Isolation
- Enhanced Test Reliability
- Cost Deduction
- Streamlined Communication

Superior Code Quality

A CI/CD pipeline allows developers to share code into a centralised repository in small batches. This reduces the possibility of buggy code making it too far into production. It also means that all developers on the team can thoroughly detect and fix the most serious bugs, and deliver software free of any critical issues. This results in more satisfied user bases, high level of customer loyalty and higher probability of users recommending our products and services to others.

Reduced Change & Review Time

A CI environment can be configured to communicate with Version Control systems, which means any changes pushed to the mainline repo can trigger a CI run, automatically checking the new code and whether all tests are passed. Such an approach is very beneficial when direct communication is a roadblock or when teams are spread across remote locations, saving on the costs and delays related to manual code reviews and meetings.

Accelerated Release Cycles

Since CI/CD shortens the time taken to detect and correct newly introduced bugs, it allows for accelerated release rates. Any software development system can support recurring releases only if code is developed in a continuous automated testing pipeline, which is a job CI/CD fullfills by continuously merging new code and deploying it to production-like systems to keep it in a release-ready state allowing developers to constantly churn out improvements and feature updates which in turn maintains customer satisfaction and trust in the product.

Instant Fault Detection & Isolation

Fault isolation refers to systems designed to limit the negative results of an error by pinpointing its cause and location; Continuous testing in CI pipelines reduces the possibility of undetected problems, as well as deploying new code in clean, isolated environments which makes newly introduced errors easier to pinpoint making fault isolation simpler and quicker. Faster fault detection massively cuts back on downtime in production environments which lowers the losses resulting from bugs and keeps customer trust in our services high.