ADOPTING CI/CD PIPELINE

CI/CD can be pictured as a pipeline, where new code is submitted on one end, tested over a series of stages (source, build, test, staging, and production), and then published as production-ready code. Each stage of the CI/CD pipeline is structured as a logical unit in the delivery process.

Automating CI means developers can catch errors easily and spend less time on issues, hence saving costs because of reduced man-hours

Automating Unit Tests in CI means developers can catch production-breaking bugs in development and have less production bugs. Costs can be saved with this strategy

Automating infrastructure deployment in CD means there will be less human error and faster deployment

Automating Smoke Tests helps to reduce downtime from deployment crashes, and ensures production servers are always online, reducing the risk of money loss due to deployment downtimes.

Automating rollbacks in CD makes it possible to quickly rollback to working state of service, and thus helps prevent production downtimes.

This enables the rapid and reliable delivery of features and updates.