Introduction

Neogrowth is a pioneer in lending based on the underwriting of digital payment data. It is a NBFC registered with RBI organization and started commercial operations in 2013. The company's proprietary technology platform offers unsecured loans to merchants who accept cards and other digital payments from customers. By far, Neogrowth has partnered with 13000+ business merchants to help them grow their businesses. Their mission is to have a positive social impact on the financial lives of small & medium business merchants across India. These merchants are more than 50% of whom are creditworthy but until now have been excluded from accessing loans based on traditional underwriting methods.

Business Needs

As an organization that provides loans to SMEs, maintaining a proper base of customer relation and interaction is a huge requirement. Hence, the client was developing an internal CRM application for which they were performing manual deployments in their environment. This made the development process highly prone to human erros, reduced efficiency and affected the code quality.

Developers at Neogrowth were logging into the production servers which was a major security risk fo the organization. With these pain points, the overall

- Overall development time increased with backlogs of non-critical errors

- The process did not allow optimal utilization of resources

- Coding process was prone to human error and rectifying changes was causing unintended consequences.challenges the client was facing were:

Solution Approach

- The client was using GitHub as their version control system for which AWS Code Pipelines were implemented for each environment that would get triggered whenever any commit was made to the corresponding branch of GitHub repository.

- As soon as the code was committed, Code Pipeline would get triggered and pull the code from repository and trigger the Jenkins build.

- Jenkins was used to validate the config variables. Hence, in any scenario of any variable missing, the entire build would fail.

- For any deployment failures, Code Deploy Rollbacks were implemented to deploy the previously stable running version of the application.

- Upon successful build, AWS Code Deploy would get triggered and it would Deploy the artifacts to the target servers.

Reaping Rewards

Cloud experts at Rapyder provided the whole solution in just two weeks in which the right solution and configuration that could support the seamless functioning of the application was designed to achieve the desired results.

- With fully automated CI/CD solution, any scope of human prone errors was removed which increased fastened the entire development cycle.

- With automation in the development cycle, the time to market of this CRM application was reduced to 40%.

- The client to witness a ROI of 30%.

- As the development lifecycle was automated, developers now spend more time development rather than operations.