

DevOps Adoption - Building A CI/CD Pipeline With Azure DevOps

ABOUT CLIENT



The client caters to the digital and business analyses needs of large Oil & Gas companies and other large enterprises.

BUSINESS REQUIREMENT

The client's large-scale web apps and business solutions are extensively used and require frequent updates to keep up with their customers' rising business needs. The client planned to launch a new SaaS product hosted on Azure and wanted to opt for seamless deployments with a dedicated CI/CD pipeline.

By implementing a DevOps culture, the client wanted to improve its software delivery methods by implementing Continuous Delivery principles to enable faster releases, better software quality, and more efficient engineering workflows. They also wanted to implement Agile practice and enhanced quality control reports and dashboards.



CCS SOLUTION

- Based on the business priorities and product engineering requirements, the customer chose Azure DevOps for their current needs and CCS Cloud and Infrastructure Team helped the client to implement DevOps for the SaaS product development to improve the software delivery methods, software quality, and engineering process.
- Under Azure DevOps several services were used to deliver the DevOps framework and follow the CI/CD approach for the SaaS product.
- Following the agile approach, Azure Boards were provided for work items, tasks, and backlogs.
- Configured cloud-hosted private Git Repositories and collaborated with managed Azure DevOps Services to build better code with pull requests, version control and advanced file management.
- Configured Azure Pipelines to build and deploy resources like function apps, logic apps and SPFx solutions using the CICD process.
- Relied on Azure Test Plans to improve code quality using planned and exploratory testing services for apps.
- Made use of project wiki to record all the documentation and share information with all users.



BUSINESS BENEFITS

- Azure DevOps automated the build and source code management which enhanced the speed of deployment.
- Version-controlled Git Repositories helped the client maintain efficient and well-organized code.
- The client could deliver the SaaS product through the Azure DevOps package with relevant checkpoints at each stage of the product lifecycle.
- Zero downtime during build deployment.
- Streamlined the testing phase and improved test coverage.
- Provided an improved code standard with the support of testing services availed from Azure DevOps.
- Reports and dashboards using managed Azure DevOps Services that provide business intelligence.
- Adopting Azure DevOps helped the client control costs, drive performance and conduct fault-tolerant optimizations.

