# Case Study

## Cloud Challenge

## nubeGo Engagement

www.nubego.io

---

## ISKY · RESEARCH

# iSky Research - DevOps

iSky Research provide an innovative financial services CX (Customer Experience) digital intelligence platform. Through their online portal, iSky Research surface unique data insights incorporating hundreds of banking channels, as well as innovation news, competitor insights and a library of visual content including screenshots, video and banking correspondence.

---

### Build a Foundation for Best Practices

In order to retain high quality levels for their end users and to accelerate the development of their new data platform, iSky wanted to take advantage of software development best practices for automated testing and deployment of their web APIs. iSky wanted to ensure rapid feedback on the quality of code changes made to their new data API platform, and to protect against regression of existing services when new and third-party developers contribute changes to their application code or data platforms.

With limited available in-house experience of CI/CD or automated testing frameworks, iSky required a continuous integration pipeline that was straightforward to use, easy to understand, and required little ongoing maintenance with minimal hosting costs.

iSky approached nubeGo to define and implement a continuous integration pipeline and automated test framework for their GraphQL API and database.

---

### Prefer Managed Services

To ensure that little or no ongoing maintenance of the CI/CD platform was required, nubeGo selected AWS Managed Services for the DevOps Toolchain.

nubeGo had a good understanding of the iSky data APIs so were able to define an AWS CodeBuild project to build and collate software artefacts for deployment. Source code was moved into an AWS CodeCommit repository to locate it close to the software and take advantage of AWS IAM permissions structures for restricting access to source code.

AWS CodePipeline was selected for its simplicity as the CI/CD orchestrator to fetch source code, run tests and deploy software artefacts. The visual workflow view of the AWS CodePipeline provides clear feedback on build health for non-technical stakeholders.

Deployment of changes to the API and database were administered through

CloudFormation Change Sets to ensure that only those elements of the software that have changed are redeployed, reducing the risk of unnecessary changes causing unexpected behaviours.

The pipeline itself was defined and deployed using Infrastructure as Code (IaC) and stored in CodeCommit to provide version control. Using Cloudformation parameters, the pipeline can be replicated quickly and easily for new source code branches, for example when a new developer wishes to create a feature branch.

Automated unit tests and functional integration tests were written for existing source code using the Mocha test framework for Node JS. Tests were automatically executed as part of both the build and test phases of the continuous integration pipeline. With use of Chai for test assertions and Sinon for method stubbing/mocking, tests could be rapidly built and run against all new software changes.

> What nubeGo have done for us is to build the foundations of best practices that everyone who contributes to the iSky platform can build upon. By putting in place robust testing routines and clear visibility of code quality and build health, we can get rapid feedback on the changes people make, and increase our confidence in the quality of our evolving platform as we continue to innovate.
>
> Matt Webb - Technical Lead, iSky Research

# Case Study

## ISKY · RESEARCH

## Realised Benefits

## Tools Used

iSky now use a four-stage fully-automated continuous integration process for testing and deploying software changes to their web APIs. The platform is entirely serverless and managed within the AWS environment, secured using IAM roles.

Over 89% of new API application code is covered with automated unit tests that are run upon every change to the source code. Scenario-based functional tests are performed before deployment upon an entirely separate test deployment; this simulates the upgrade of the production system, to reduce the chance of errors at the point when new changes are deployed. All
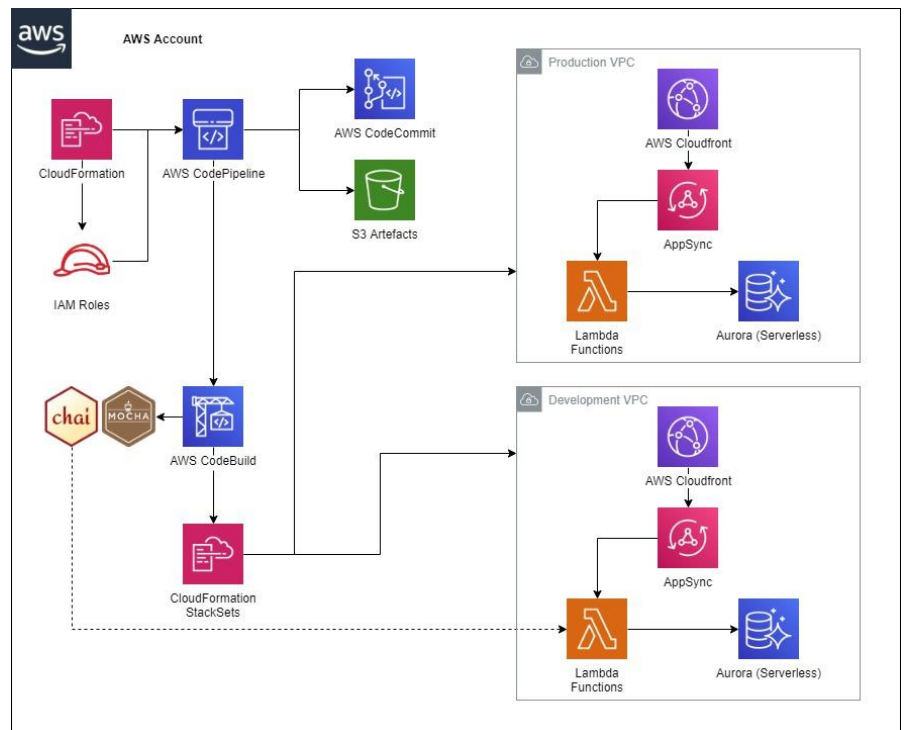
build and test activities are completed, hands-free, in under 7 minutes.

As the pipeline runs on serverless AWS Managed services, iSky only pay for what they use - pipeline executions, build time, source code and software artefact storage. The iSky CI/CD practices currently take advantage of the AWS Free Tier limits.

The CI/CD Pipeline and automated test framework will be used as a template for future projects and components..

AWS CloudFormation, AWS CodePipeline, AWS CodeBuild, AWS CodeCommit, AppSync, Amazon RDS, Aurora Serverless, IAM, S3, AWS Secrets Manager

Node JS, Chai, Mocha, Flyway