

Streamlining IT Infrastructure for an AI-based Startup

About the Client

This customer is a startup company, established in 2016 at the San Francisco Bay Area, California with an aim to improve business presentations in an organisation. Their primary product is an advanced presentation building platform which uses Artificial Intelligence to suggest the most relevant and useful slides from prior decks which leads to faster and better quality presentations. The product also provides content insights and customer engagements in the form of an interactive dashboard.

Business Challenge

- As a startup, the company's primary challenge was to manage the IT Infrastructure effectively. The already existing architecture needed a complete restructuring.
- The application was initially built to withstand only minimal code changes and enhancements at a given time.
- The application functionalities were hosted in independent servers which were prone to application failures.
- Application downtime was mandatory during any code release.
- Lack of automation that resulted in frequent manual intervention.
- The customer required a long-term solution to restructure their entire IT Infrastructure.

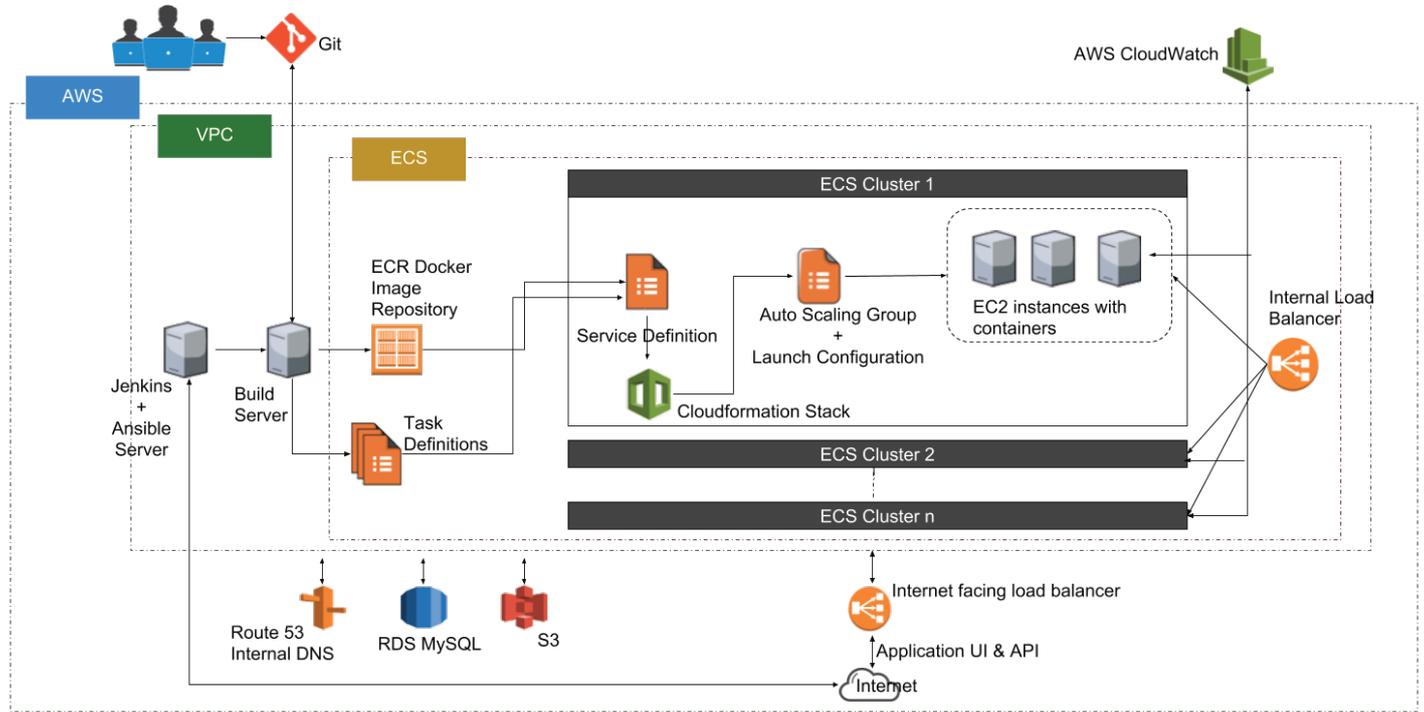
InApp's Solution

InApp proposed a DevOps driven approach which led to dividing the entire architecture into microservices to make the application more scalable. We built each functionality as Docker containers that can communicate with one other through well-established channels. A centralized repository is used for storage purpose which is available on-demand and can be accessed through APIs.

The whole idea behind the DevOps practice is to make the application more scalable and ready to use for the customer 24x7 while allowing the developers to continually make necessary enhancements without any downtime. We used a load balancer for the incoming requests to different instances based upon the load and availability.

InApp employed a centralized git repository which the developers could use to pool their code enhancements or fixes. We used tools to automate the build process and to enable continuous integration and deployment of frequent code changes while maintaining the stability of the application.

Architecture



Highlights

Code Automation

Integrated Jenkins and Ansible scripts to automate the build process and to enable continuous integration and deployment. This allows the developers to make various enhancements and upgrades to the application without compromising its stability.

Central Storage

We used secure and centralized storage containers such as Amazon S3 and RDS to store application data files which could be accessed using custom APIs.

Technology

The infrastructure is made scalable by using Docker containers and AWS Cloud services. This allows the developers to implement code changes without any application downtime.

Monitoring & Alerting

AWS services such as CloudWatch and ELK are used to gather insights regarding various transactions and present it in the form of a user-friendly dashboard.

Business Benefit

- Microservices based architecture which enables the application to be readily available on demand.
- Docker containers are used to host different instances which makes the application more scalable and stable.
- An automated platform which promotes continuous integration and deployment to implement new changes frequently.
- Efficient monitoring and alerting mechanism to provide user-friendly insights and to capture production issues.

