
DEFENSE UNICORNS



Website Architectural Design Review

About

This document outlines the tools and frameworks that will be used in bringing this project to completion. As the team learns and decides what the outcome of this site will be, there will be updates and additional content added until a final draft is agreed upon. The final draft will reflect the complete toolset and all additional details required to get an overview of the inner workings of the website.

Featuring:

Front-end and Back-end Frameworks
AWS CodePipeline and Elastic Beanstalk

Prepared by: Wedefegerson Desir
June 10, 2021

DEFENSE UNICORNS

BACK-END

Node JS

Node JS is an Asynchronous Javascript runtime that comes with a built in package manager, NPM or Node Package Manager. This allows users to have access to an extensive set of tools that they can utilize when creating their web applications.

Node JS should be implemented to build out the Defense Unicorns website in order to take advantage of its large library of modules and event loop driven behavior.

Benefits

- Allows usage of several modules which will be the backbone of our web application
- High performance due to single threaded and event-driven nature linked with Chrome's V8 engine
- Using Node JS allows Javascript to be used on both the front end and backend and won't require an additional language to be used.
- Easy scalability
- Various dev tools available as modules
- Free and open source

Disadvantages

- NPM packages may be of poor quality and compromise security
 - Can slow down with long running calculations
 - Unstable API - updates may reduce compatibility with old codebases
-

DEFENSE UNICORNS

FRONT-END

TBD

DEFENSE UNICORNS

DELIVERY

AWS CodePipeline

The Code Pipeline service gives the ability to incrementally grab our files from our GitHub repository after each update and then deploy to Elastic Beanstalk. AWS charges \$1 per active pipeline per month and any additional charges are based on utilization of s3 or other connected AWS services.

Benefits

- Customizable steps in the pipeline allow us to specify where code is built
- Testing can be added as a step in the pipeline to assure delivery
- Ability to add manual approval or automate all steps in the deployment process
- Issues or test failures will suspend automated deployment
- Able to deploy to staging environments as part of the Pipeline
- Charged only based on active Pipelines.
- Ability to integrate infrastructure as code solutions such as Terraform.
- Easy integration with Elastic Beanstalk
- Wide range of available source providers: Github, AWS s3, BitBucket

Disadvantages

- Costs: Tools such as Jenkins exist that work just as well at no cost.
-

DEFENSE UNICORNS

AWS Elastic Beanstalk

With AWS Elastic Beanstalk we are able to seamlessly integrate with the code pipeline to deploy our web application and easily configure provisioning or resource control. Elastic Beanstalk as a service is completely free and all charges will be based on utilized EC2 instances.

Benefits

- Ability to control which AWS Resources are used to run our web application
- Features such as auto-scaling, load balancing and health monitoring are built in.
- Support for a wide range of web application backend frameworks including Node JS
- Only pay for the resources that you use.
- Set up alarms based on monitoring metrics
- Majority of LeapFrog team has experience with AWS

Disadvantages

- A little bit more complex to setup than some similar tools
 - Overhead orchestration of several other AWS Services
 - Not extensively documented
-