

# Cloud Computing Assignment 3

Leonard McDonald  
S3879586

## Contents

Links:.....	3
Summary.....	4
Introduction .....	5
Related Work.....	6
System Architecture .....	7
System Architecture Description .....	8
Elastic Beanstalk .....	8
API Gateway .....	8
Lambda .....	8
DynamoDB .....	8
S3 .....	8
Mojang API.....	8
Server List Ping API .....	9
mctthings.json (DynamoDB data).....	9
Important Libraries.....	9
References.....	10

## Links:

Live Application: <http://a3mdev.us-east-1.elasticbeanstalk.com/>

GitHub: <https://github.com/lenover12/MineDev>

External API's

Mojang: [https://wiki.vg/Mojang\\_API](https://wiki.vg/Mojang_API)

SLP: [https://wiki.vg/Server\\_List\\_Ping#Status\\_Request](https://wiki.vg/Server_List_Ping#Status_Request)

## Summary

MineDev is a web app that hosts some tools for Minecraft Datapack & Resource pack developers to aid in their development and creations used in Minecraft, as well as share their creations with the public. Utilising AWS (Amazon Web Services) cloud platform and infrastructure, a host of AWS technologies, and various API's and libraries.

Minecraft Pack Developers who visit the web app will find these features to interact with:

- Player search
- Player information grabber (UUID and texture)
- Block/Item search and random search
- Block/Item Information summary
- Block/Item media showcase
- Language Translator
- Book text formatter for creating formatted Minecraft Book items.
- Share page.

Services used in the development of MineDev were:

AWS	External Services
<ul style="list-style-type: none"><li>• Elastic Beanstalk</li><li>• API Gateway</li><li>• Lambda</li><li>• DynamoDB</li><li>• S3</li></ul>	<ul style="list-style-type: none"><li>• Server List Ping (SLP) API</li><li>• Mojang API</li></ul>

## Introduction

Minecraft is currently the most popular game created, released in mid-2009 it has seen ongoing updates by Mojang the parent company. It is a sandbox survival game that allows for a lot of creativity and experimentation, players use blocks to build any voxel structure or mechanical circuits to manipulate these blocks, and survive off animals and the land to defend against monsters and starvation.

Players of Minecraft have managed to find their way into game development hobbies and paid work due to the vast popularity of the game, sub communities of developers who wish to add or change the game, and Minecraft's in-built support of technical play and developer tools.

MineDev was created as a web app designed for a niche community focused around supporting those interested in developing anything for Minecraft, including the Author of this paper. With easily accessible player and server information and some implemented formatting and localisation tools to help developers create datapacks and resource packs in a community, while creating a sharing space or `show-and-tell` for their creations that is accessible and useable by anyone.

Utilising Python and Flask in the backend with various packages for handling, processing, and requesting data, showcasing the data through HTML via Jinja, CSS, JavaScript and hosted on and utilising AWS.

There are websites that offer similar tools for developers, MineDev is alike these other websites but focuses on simpler implementation of developer tools and out-of-the-box choices that create a suite of tools I have not seen combined before all-in-one place.

## Related Work

<https://mcstacker.net/>

A website that covers “commands” used in Minecraft in every intricate detail, one of the most useful websites for a Minecraft pack developer that has a lot of experience, aimed at those who are trying to achieve complicated feats with commands and functions.

[https://minecraft.fandom.com/wiki/Minecraft Wiki](https://minecraft.fandom.com/wiki/Minecraft_Wiki)

The Unofficial Wikipedia information HUB of Minecraft, serving all information about the core game publicly known.

<https://www.digminecraft.com/index.php>

A website that has many beginner friendly tutorials and guides, including command generators not unlike the book formatter found on MineDev.

<https://advancements.thedestruc7i0n.ca/>

A website for creating .json files used in datapacks in the creation of new custom advancements or achievements, developers can use this website to create advancements and assign image textures to those advancements with resource packs.

<https://sourceblock.net/beta/en-US/tools/data-packs>

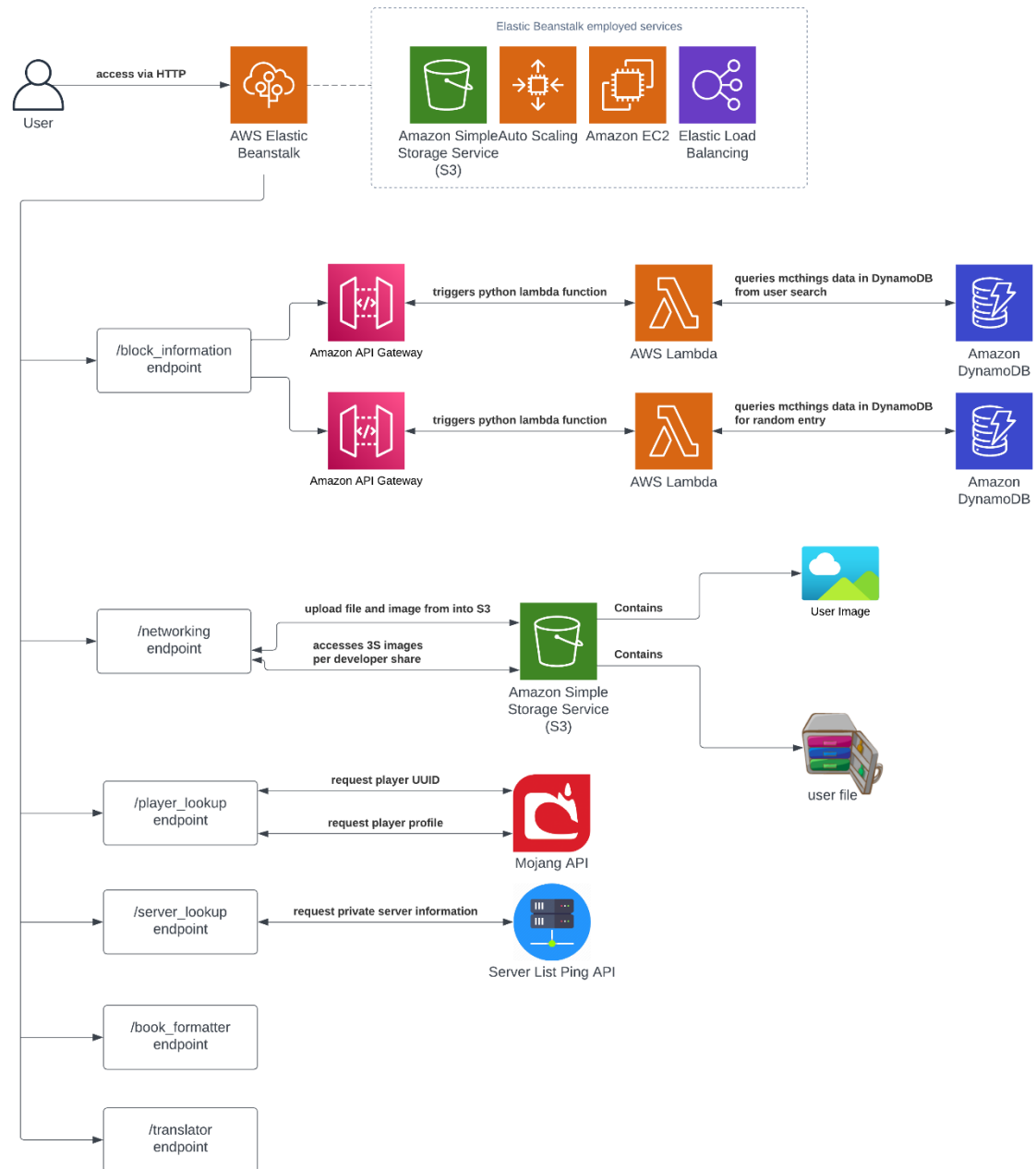
A website with a variety of generators for very specific tasks shared by a single developer, including necessary but difficult to implement datapacks like Raymarching functionality (difficult due to the way Minecraft allows developers to edit their game)

<https://minecraft.novaskin.me/>

Minecraft default game .json files that are able to be manipulated through datapack and resource packs are easily accessible on this website for developers who don’t know how to extract them from the core game files

This website also allows the searching and editing of player skins live on the website.

# System Architecture



# System Architecture Description

## Elastic Beanstalk

AWS Elastic Beanstalk is utilized as the hosting platform as it provides a scalable and reliable infrastructure for the MineDev app. The application is developed using Python with Flask as the web framework. The source code for the application is stored in an S3 bucket, ensuring reliable storage.

Elastic Beanstalk is configured to handle the HTTP traffic for the MineDev web app. The application load balancer, along with the associated security groups, is set up to allow and route HTTP requests to the EC2 instances running the web application.

By leveraging Elastic Beanstalk, the MineDev web app benefits from automatic scaling and load balancing. As the traffic to the web app fluctuates, Elastic Beanstalk automatically adjusts the number of EC2 instances to handle the load efficiently. This ensures optimal performance and availability for users accessing the MineDev web app.

## API Gateway

API Gateway leverages a RESTful API approach. This API Gateway generates routes that serve as triggers for the associated Lambda functions, enabling seamless execution of serverless code.

The API Gateway plays a crucial role in the MineDev web app by acting as a centralized entry point for all incoming API requests.

## Lambda

Dedicated Lambda functions are designed to handle specific tasks and are seamlessly connected to RESTful API routes. These functions enable the sending and retrieval of user comments and facilitate the retrieval of block information from DynamoDB tables.

## DynamoDB

DynamoDB serves as the non-relational database for MineDev, offering two tables for data storage. The first table stores and provides access to user comments based on their creation time. The second table contains Minecraft block/item information, which was provisioned from a JSON file. DynamoDB efficiently handles the storage and retrieval of these data sets.

## S3

The web app utilizes two S3 buckets provided by the S3 service. The first bucket hosts the application backend, while the second bucket stores files and images submitted through user comments.

## Mojang API

Integrating with the Mojang API, the web app retrieves user UUIDs and accesses user profile data and texture image references. It first connects to the Mojang API endpoint to obtain the user UUID based on the provided username. This UUID is then utilized to make another API request to the Mojang API, specifically to the endpoint responsible for fetching user profile data and texture image.



references. These details are subsequently used in the `player_lookup` page of the web app to display comprehensive user information and texture images associated with the player.

### Server List Ping API

Server List Ping API is leveraged to retrieve information about private servers that are hosted online through IP addresses. By utilizing this API, MineDev can send a server ping request to the specified IP address and port of the server. The API then returns valuable information about the server, such as its status, player count, server version, and other relevant details.

### `mcthings.json` (DynamoDB data)

A json file containing summary information and media links scrapped via a python web scraper program from Minecraft Wiki.

### Important Libraries

MineDev imports libraries of Translate to utilise as a localisation tool.

## References

"Amazon API Gateway: What Is It? | Knowledge Base | Dashbird." *Dashbird*, <https://dashbird.io/knowledge-base/api-gateway/what-is-aws-api-gateway/>. Accessed 10 May 2023.

Hava, Team. "What Is AWS Elastic Beanstalk?" *Hava Pty Ltd*, <https://www.hava.io/blog/what-is-aws-elastic-beanstalk>. Accessed 17 May 2023.

"More than 'Hello World in Lambda': Build and Deploy Python Flask APIs in AWS Lambda via CDK." *DEV Community*, 23 Mar. 2020, [https://dev.to/raphael\\_jambalos/more-than-hello-world-in-lambda-build-and-deploy-flask-apis-in-aws-lambda-via-cdk-1m04](https://dev.to/raphael_jambalos/more-than-hello-world-in-lambda-build-and-deploy-flask-apis-in-aws-lambda-via-cdk-1m04).

says:, kprasannaPK. "Agile Project Management in 2017A Remote Retrospective." *Joshua Hoover About Me Back Joshua Hoover About Me Back*, 16 June 2017, <https://joshuahover.com/deploying-a-python-flask-web-app-on-aws-lambda/>.

Taylor, David. "What Is AWS Lambda? Lambda Function with Examples." *Guru99*, 20 Jan. 2020, <https://www.guru99.com/aws-lambda-function.html>.

"What Is an API Gateway? A Quick Learn Guide | NGINX Learning." *NGINX*, <https://www.nginx.com/learn/api-gateway/>. Accessed 02 May 2023.

*What Is AWS Elastic Beanstalk?* <https://docs.aws.amazon.com/elasticbeanstalk/latest/dg/Welcome.html>. Accessed 13 May 2023.

*What Is AWS Lambda?* <https://docs.aws.amazon.com/lambda/latest/dg/welcome.html>. Accessed 29 April 2023.

"What Is AWS Lambda? | How Does It Work? | When to Use Lambda?" *Besant Technologies*, 22 Nov. 2018, <https://www.besanttechnologies.com/aws-lambda>.

Wigmore, Ivy. "What Is AWS Elastic Beanstalk ? | Definition from TechTarget." *TechTarget*, 1 June 2014, <https://www.techtarget.com/searchaws/definition/Elastic-Beanstalk>.