

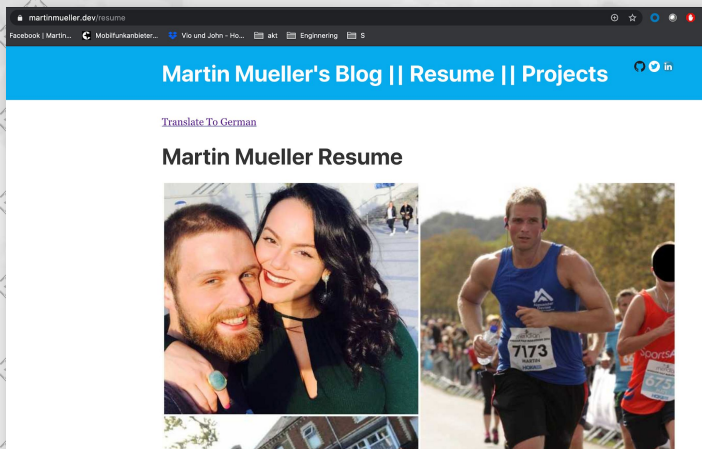


# **Run your AI LangChain on AWS**

Martin Müller

# Who am I?

MartinMueller.dev/Resume



## Summary

office@martinmueller.dev

Passionate Full Stack Developer. I like to have a broad collection of skills. From developing the backend and frontend applications written in Java, JavaScript, or any other language, to developing a cost-optimized cloud infrastructure solution. An enthusiastic agile player with strong communication skills. Looking for taking on new challenges and learning new technologies.

"What I don't know, I will learn"

## Highlights

- AWS Professional Solution Architect certified 
- AWS Community Builder
- TypeScript, Java, JavaScript, Angular, React
- AWS Lambda, EC2, CDK, DynamoDB and many more
- Maven, Gradle, SQL, ElasticSearch
- Docker, Kubernetes, Helm
- Spring, GitHub (Actions), Travis, GitLab, BitBucket

## Experience

AWS Community Builder  
Mar 2022 - current

AWS Consultant  
Nov 2020 - current

- Helping startups into the AWS Cloud
- Leveraging AWS CDK to manage AWS resources

Senior Cloud Architect - www.kreuzwerker.de, GER  
Jan 2022 - May 2022

- Together with the customer building awesome AWS cloud solutions

# Agenda

**01**

**What is  
LangChain?**

**03**

**LangChain  
Demo**



**02**

**Why run it  
on AWS?**

**04**

**Reflecting & QA**



# Disclaimer!

- I'm not a Data Engineering or ML expert! I'm an AWS DevOps Full-stack engineer!
- LangChain is super new to me.



# **01 What is LangChain?**

# What is LangChain?

- Cohesive framework for interacting with multiple LLM models
  - I love the abstraction on top of like OpenAI
- Python library <https://github.com/hwchase17/langchain>
- JavaScript library <https://github.com/hwchase17/langchainjs>
- Has cool features like:
  - Chains - Sequence of LLM calls. Even across multiple LLMs.
  - Agents - Involve an LLM making decisions about which Actions to take.
  - Memory - Persisting state (like conversations) between calls of a chain/agent.
  - ...



## **02 Why run it on AWS?**

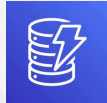


# Why running it on AWS?

- Your laptop isn't production ready! You will need an external service like AWS to run your AI app.
- AWS offers an amazing set of Serverless services you easy can integrate like:



S3



DynamoDB



Cloudfront

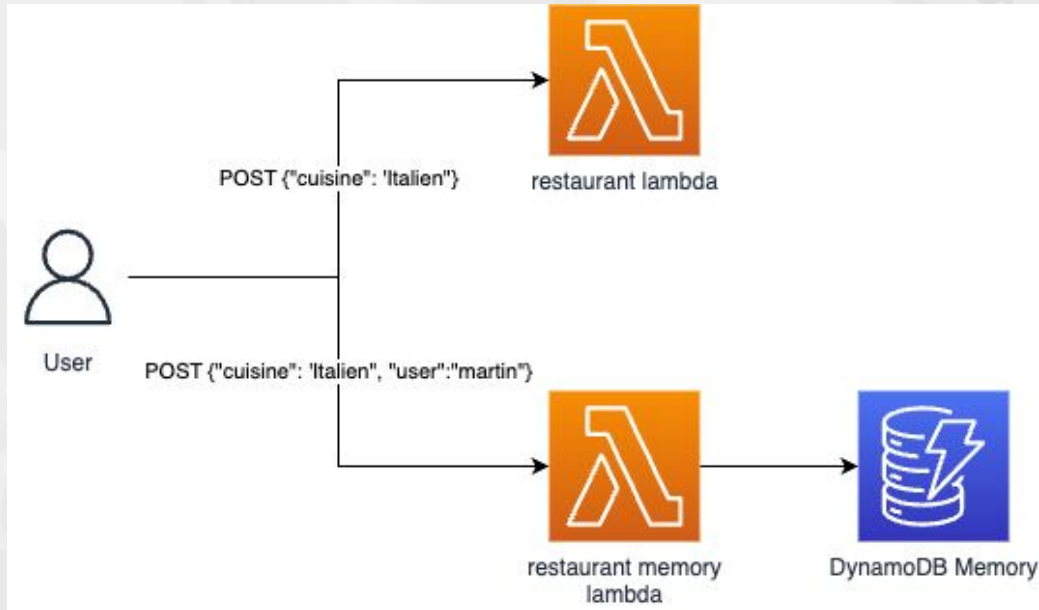




## 03 LangChain Demo

# LangChain Demo

<https://github.com/mmuller88/aws-ts-langchain>



# 04 Reflecting & QA

- LangChain is super cool and fun.
- The library is early so you will encounter limits/problems.
  - Reach out to the LangChain Discord Community! <https://discord.gg/nTWCmrtWdp>
  - Perfect opportunity to contribute!
  - DynamoDB sortkey doesn't seem to be properly implemented. Perhaps will write a PR.



MartinMueller\_



mmuller88



[martinmueller.dev](https://martinmueller.dev)



[office@martinmueller.dev](mailto:office@martinmueller.dev)



04 QA



MartinMueller\_



mmuller88



[martinmueller.dev](https://martinmueller.dev)



[office@martinmueller.dev](mailto:office@martinmueller.dev)

# What is AWS?



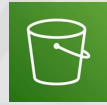
- Amazon **Web Services**
- Part of Amazon
- Offers services for Storage, Compute, Content Delivery and many more



EC2



Lambda

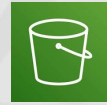


S3



Cloudfront

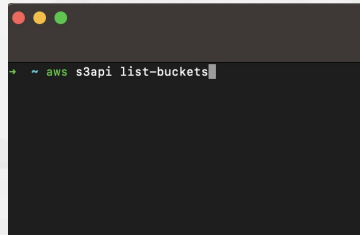
# What is AWS?



EC2   Lambda   S3   Cloudfront

- Amazon Web Services
- Part of Amazon
- Offers services for Storage, Compute, Content Delivery and many more

- Access with AWS Console, CLI, SDK (Python, TypeScript, ...)



```
You, 18 hours ago | 1 author (You)
1 import AppsyncClient from 'appsync-client';
2 import * as lambda from 'aws-lambda';
3
4 const { TABLE_NAME } = process.env;
5
6 const USER_ID = 'd0a5517a-9088-487e-aec6-fbd7b3696363';
7
8 import * as AWS from 'aws-sdk';
9 import { CreateDeviceDocument, DeviceType } from './lib/api';
10 const docClient = new AWS.DynamoDB.DocumentClient();
11 const { APPSYNC_URL } = process.env;
12
13 const client = new AppsyncClient({ apiUrl: APPSYNC_URL });
14
15 /**
16  * @param event
17  * @returns
18  */
19
20 export async function handler(event: lambda.UserMigrationTriggerEvent) {
21   console.debug('event: ${JSON.stringify(event)}');
22
23   const params: AWS.DynamoDB.DocumentClient.QueryInput = {
24     TableName: TABLE_NAME,
25     KeyConditionExpression: 'id = :id',
26     ExpressionAttributeValues: {
27       ':id': 'USER_${USER_ID}',
28     },
29   };
30 }
```

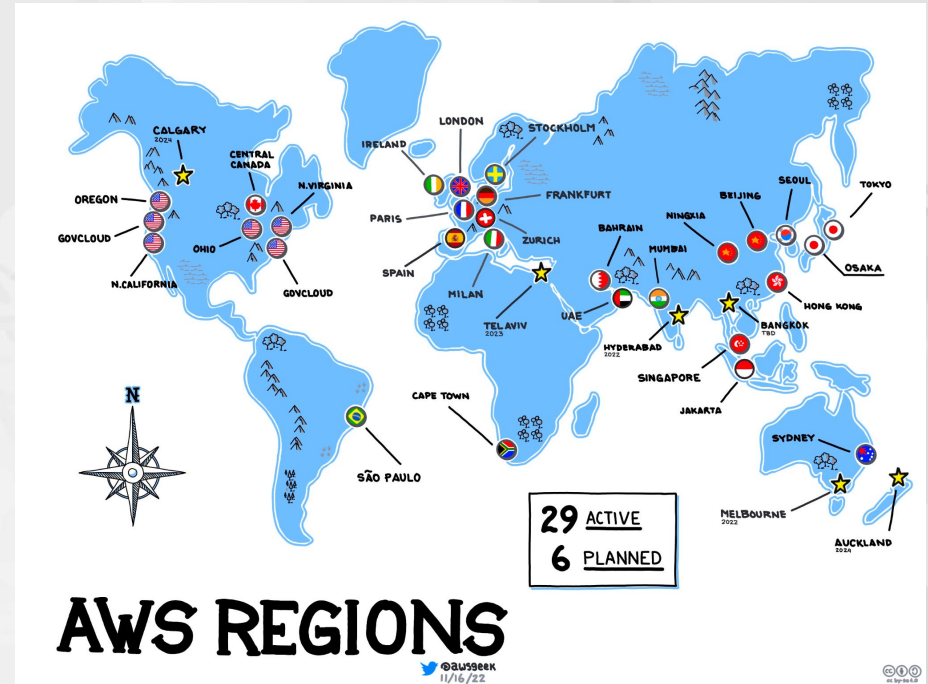


# What is AWS?

- Big server buildings around the world



Frankfurt (Main) eu-central-1





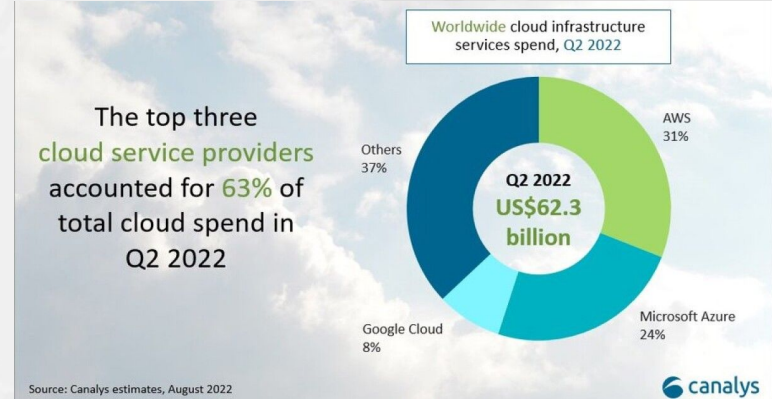
# What is AWS?

- Redundant buildings per region for better availability



# What is AWS?

- Alternative Cloud Provider are:



- BTW you are probably using AWS already indirectly  
The tool/service you are using is probably build in the cloud!



# Why use it? (Company)

Outsourcing:

- Almost every company has an IT department
- Outsource IT hardware for reducing cost
- Migrate to cloud native / serverless offers done

correctly improves aspects of:

- Performance
- Security
- Operational Excellence
- Costs
- Reliability
- Sustainability



# Why use it? (Startup)

## Greenfield:

- Having a cool SaaS business idea?
- AWS Cloud is the perfect place to implement it!
- Don't worry about IT hardware
- Pay as you go. If you use the cloud less you pay less

## Startup friendly:

- Apply for AWS Active and get \$100k credits





# Why (not) use it?

- Lock-in
- Inexperience can cost you
  - Use AWS Courses to learn!
  - AWS certificates are super helpful

## Available AWS Certifications

### FOUNDATIONAL

Six months of fundamental AWS Cloud and industry knowledge



### PROFESSIONAL

Two years of experience designing, operating, and troubleshooting solutions using the AWS Cloud



### ASSOCIATE

One year of experience solving problems and implementing solutions using the AWS Cloud



### SPECIALTY

Technical AWS Cloud experience in the Specialty domain as specified in the exam guide



aws  certified  
Updated February 2022



**03 Why I love it!**

# Why I love it!

- I can offer my expertise to earn money as a freelancer to buy food
- Helps me to evaluate and implement my startup ideas
- The developer experience is amazing and super satisfying
- Having amazing AWS communities to help and learn from each other
- If you need help with the cloud look on:
  - hire in those communities!
  - Upwork.com
  - Ask me :)

