Food Cloud - Food Ordering Application

Team Members:

1. Abhay Tamilselvan (atamilselvan@scu.edu)
2. Arthi Kundadka (akundadka@scu.edu)
3. Jayasri Ramakrishnan (jramakrishnan@scu.edu)
4. Lakshmi Narayanan Vaigai Shrinivasan (lvaigaishrinivasan@scu.edu)

Main Goal:
This project aims to build an online food ordering platform to bring the best restaurants and customers closer and make ordering hassle free. This aims to satisfy the demands of restaurants and customers by using Amazon Web Services (AWS) as the cloud platform. This project will be implementing a serverless architecture and will rely on Amazon Web Services for all the core business concepts from hosting to registering restaurants to storage of restaurant information, menus, customer information, and scaling on the number of order requests.

Motivation:
The motivation behind this project was to choose a real time application that we can connect with easily while building. This field of ordering is of high interest because it requires the service to be up at all times (no downtime accepted), ability to seamlessly scale with increase in the number of customers and orders and the flexibility to expand on compute and storage with respect to surges. This topic seemed to us to be the best use of cloud computing resources where once setup, there would be no need to worry about maintenance of servers and developers would simply have to improve the business model with analytics and new features.

Stretch Goals:
If time permits, we plan to extend this application to include

1. Payment service and more features for delivery riders (we plan to call them clouders).
2. Personalization and Recommendation using Amazon SageMaker/ AWS Personalize

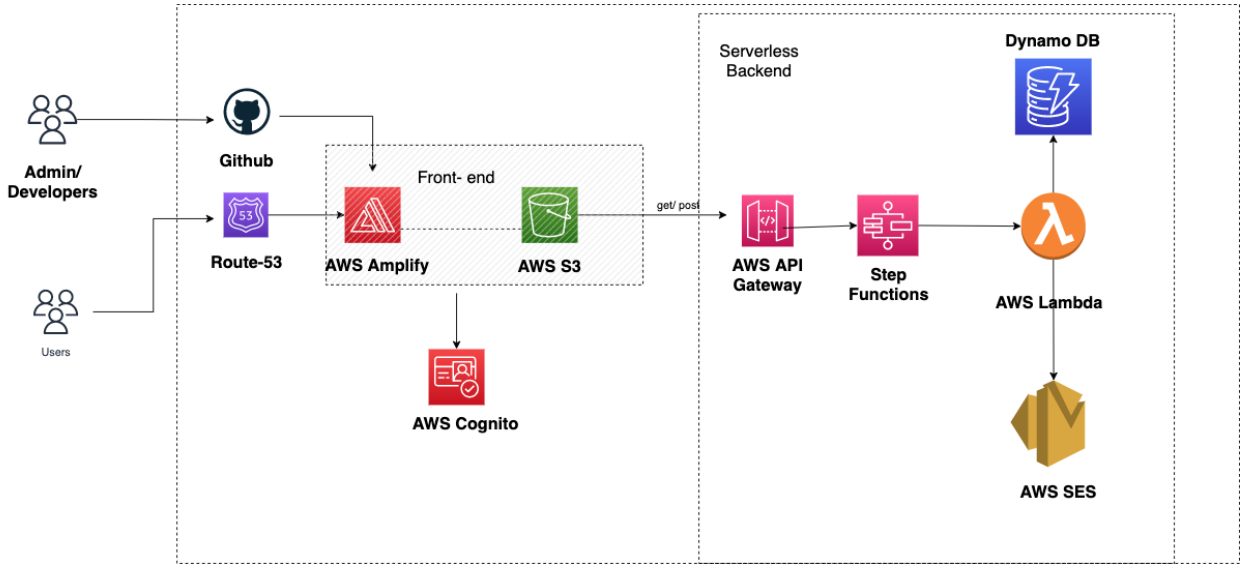
Cloud Technologies to be used:

For this project we plan to use Amazon Web Services as our IaaS. Under this we plan to use

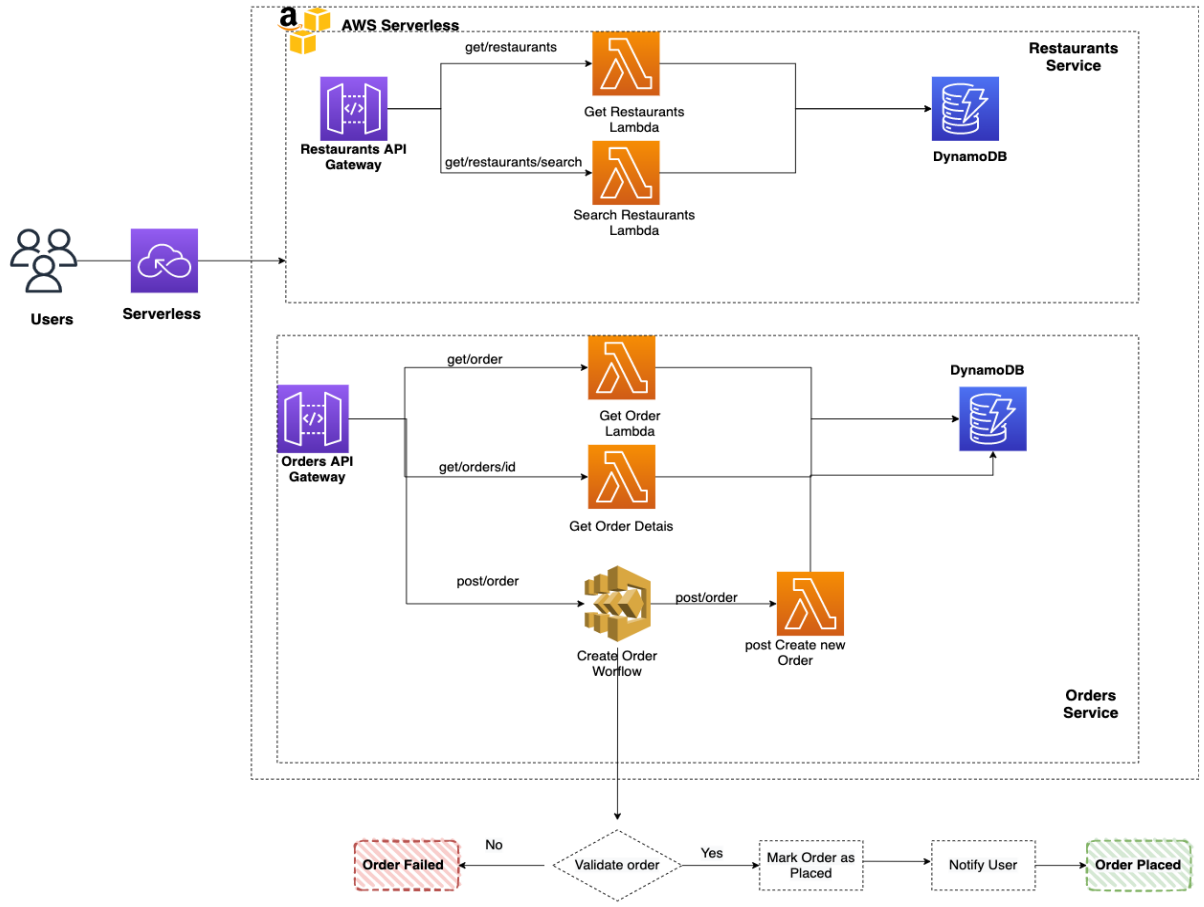
1. Compute:
    a. AWS Lambda - Event driven computing service
2. Storage and Database:
    a. AWS S3
    b. AWS DynamoDB - NoSQL Database
3. Front End Web:
    a. AWS API Gateway - To build,deploy and manage APIs
    b. AWS Amplify - To build and deploy web applications
4. Authentication and Authorization:
    a. Cognito
    b. IAM
5. Orchestration:
    a. Step Functions
6. Management - Iac:
    a. Cloud Formation / SAM

Architecture:

High-level Architecture overview:



Serverless Backend - Architecture:



Division of Responsibility:

1. Proposal, Design and Architecture - All
2. Restaurant Lambda Services - Abhay Tamilselvan
3. Order Lambda Services - Arthi Kundadka
4. Database design and DynamoDB setup/Integration - Jayasri Ramakrishnan
5. Cloud Formation configuration/ IAC - Lakshmi Narayanan Vaigai Shrinivasan
6. IAM Users and Roles Management - Abhay Tamilselvan
7. Cognito Authorization setup - Arthi Kundadka
8. Step function set-up - Lakshmi Narayanan Vaigai Shrinivasan
9. FrontEnd Sign Up and Login - Jayasri Ramakrishnan
10. FrontEnd Restaurant views - Lakshmi Narayanan Vaigai Shrinivasan
11. Customer views - Abhay Tamilselvan
12. Order View - Arthi Kundadka
13. S3 Web hosting - Jayasri Ramakrishnan
14. Route 53 configuration - Lakshmi Narayanan Vaigai Shrinivasan