### **AWS Project Abstract**

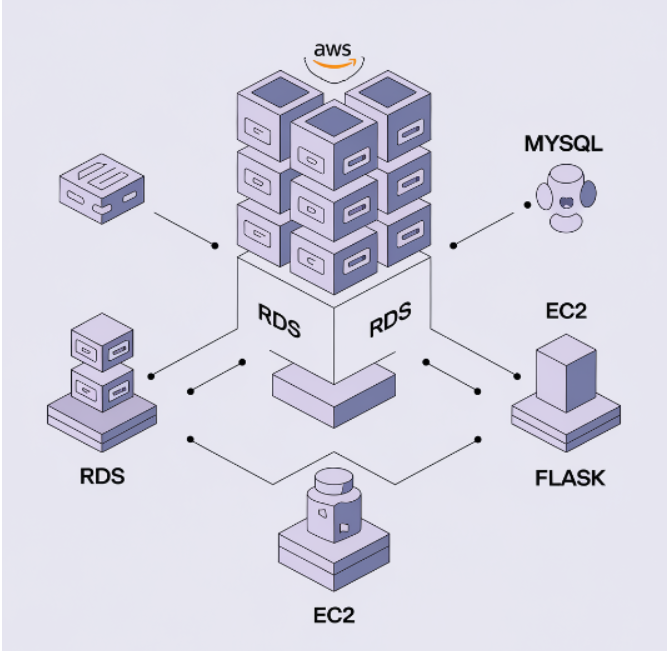
**Title: Harnessing AWS for Next-Gen Online Food Delivery**

In the rapidly evolving world of online food delivery, the need for scalable, secure, and efficient systems is critical. This project, Harnessing AWS for Next-Gen Online Food Delivery, focuses on developing a cloud-native platform that optimizes food ordering, delivery tracking, and customer service. Built using modern technologies like Flask (a Python web framework) and Amazon RDS for database management, the system allows restaurants to seamlessly manage orders and logistics while providing customers with real-time updates on their deliveries. Hosted on AWS, the platform leverages multiple AWS services to ensure enhanced scalability, security, and reliability, enabling a superior online food delivery experience.

**Objective:**

The core objective of this project is to build a fully functional and interactive online food delivery platform, where restaurants can efficiently manage orders, and customers can place and track their orders in real time. The platform aims to streamline the entire food ordering and delivery process, ensuring seamless communication between restaurants, delivery personnel, and customers through Amazon SNS for real-time notifications. By automating and optimizing the order management process, the system enhances both operational efficiency for restaurants and the overall customer experience.

**Architecture Overview**



**1. Cloud-Based Application Architecture**

* **Flask Application (Hosted on AWS EC2):**The core food delivery platform is built using Flask and deployed on AWS EC2 instances. It handles key functionalities like user registration, login, order placement, delivery tracking, and restaurant management. Both customers and restaurants have dedicated dashboards, ensuring a smooth user experience. The architecture is fully cloud-based, leveraging AWS services for scalability, reliability, and ease of maintenance.
* **Amazon RDS (MySQL Database):**Amazon RDS is used to manage the MySQL database, storing crucial information such as user details, restaurant menus, order histories, and delivery statuses. This setup provides a scalable solution with high availability, automated backups, and simplified database management.
* **User Authentication and Security:**The application uses Flask-WTF for secure form handling, Flask-Login for session management, and bcrypt for password hashing to ensure secure user authentication, safeguarding customer and restaurant data.

### **2. AWS Service Integration**

* **Amazon RDS (Relational Database Service):**Amazon RDS hosts the MySQL database that stores all application data, including customer orders, user details, restaurant menus, and order statuses. This cloud-based service ensures high reliability, security, and scalability, eliminating the need for on-premise database management.

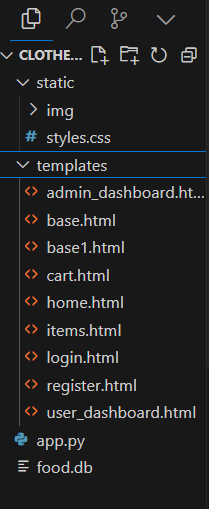
### **3. Workflow and User Interactions**

* **Customer Workflow:**Customers log in to their dashboard to browse restaurant menus, place orders, and track their deliveries in real time. Each order is managed through the system, enabling users to view order statuses from confirmation to delivery.
* **Restaurant Workflow:**Restaurants log in to their dashboard to manage new orders, update menu items, and track completed deliveries. Upon receiving an order, restaurants can view all necessary details and update the status as it progresses from preparation to delivery.

### **4. Deployment and Configuration**

* **AWS EC2:**The application is deployed on AWS EC2 instances, with Flask managing backend operations. Key libraries like boto3 are installed to interact with AWS services. The application is hosted behind an Nginx server for efficient request handling and load balancing.
* **AWS RDS Configuration:**Amazon RDS is configured with MySQL for storing critical application data, with security groups and automated backups ensuring high availability and data protection. This scalable database architecture supports increasing demand as the platform grows.

**File Architecture**

****

**Conclusion:**

In conclusion, *Harnessing AWS for Next-Gen Online Food Delivery* delivers a robust and scalable solution tailored for the modern food delivery industry. By leveraging AWS services, the platform ensures a seamless experience from order placement to delivery tracking, enhancing both restaurant operations and customer satisfaction. The use of Amazon SNS for real-time notifications and Amazon RDS for reliable data management demonstrates the system’s commitment to security, scalability, and performance. This project underscores the transformative role of cloud technology in revolutionizing the online food delivery landscape, setting the stage for future growth and innovation in the sector.