INFS3208 - Cloud Computing
Thi Thuy Tien Nguyen - 48338433
Individual Project - Project Proposal

1. Introduction

Background

This project is a store management system that leverages cloud technology for scalability and availability, enabling administrators to effectively manage inventory and orders while providing users with an intuitive platform for browsing products and placing orders.

Motivation

The motivation behind this project comes from the real challenges that retailers face in today's fast-paced market. For administrators, the ability to manage inventory and orders efficiently helps reduce operational overhead, minimize errors, and improve fulfillment speed. For users, a seamless browsing and ordering experience fosters customer satisfaction and loyalty, ultimately driving sales. This project utilizes cloud computing to enhance both administrative operations and the customer shopping experience in a competitive retail environment.

Objectives and Features

The project aims to develop a comprehensive store management system that enhances retail operations for users and administrators. Key features include user account creation, secure login, item browsing, order placement, order history access, and order cancellation. For administrators, it provides inventory management, enabling them to create, update, delete items, and manage customer orders. Finally, the project aims to use cloud computing to deploy the services, ensuring scalability, reliability, and availability that can handle varying traffic levels while delivering a seamless experience for users and administrators alike.

Limitations of traditional computing solutions

Traditional computing solutions have several limitations for modern retail needs. They involve high upfront costs for hardware and infrastructure, making them expensive to implement. Their reliance on fixed hardware hinders scalability during peak demand, leading to performance bottlenecks and underutilization of resources. Additionally, resources can be underutilized during off-peak times, resulting in wasted investments. Managing these systems is also complex and time-consuming, requiring specialized expertise for installation and maintenance. Finally, keeping software and hardware up to date can also disrupt operations, causing downtime and negatively impacting the customer experience.

Benefits of cloud computing

Cloud computing technologies greatly improve the effectiveness of the store management system. They offer the flexibility to scale resources as needed, especially during peak shopping periods, ensuring a smooth experience for users. With built-in redundancy and failover mechanisms, these solutions enhance reliability and minimize downtime. Moreover, the pay-as-you-go model eliminates costly upfront investments, making it ideal for small businesses. Additionally, cloud solutions simplify management by abstracting infrastructure complexities, allowing administrators to focus on core functions like inventory and order processing while enabling quicker updates and feature integrations.

2. Technical Solutions

In this project, I plan to leverage several cloud computing technologies to build a robust store management system. Google Kubernetes Engine (GKE) will be used for orchestration and deployment of all containerized applications at scale. The Application Load Balancers will be used to expose the services and handle routing. For data storage, PostgreSQL is used as a reliable relational database to manage inventory and order data. The backend services will be developed using Flask and Python, providing a flexible framework for RESTful APIs. On the frontend, ReactJS and TypeScript will be used for creating frontend UI. Finally, Docker will be used for containerization.

The number of nodes will be scaled up and down depending on demand. For the purpose of cost estimation, I will assume the cluster requires an average of 2 nodes. The estimated monthly cost for the cloud resources used in this project includes approximately $80 for a GKE cluster with 2 n1-standard-1 nodes, $36 for two load balancers (one for the backend and one for the frontend), and $3 for a static IP address for the backend. This brings the total estimated monthly cost to around $119.

3. Architecture Design

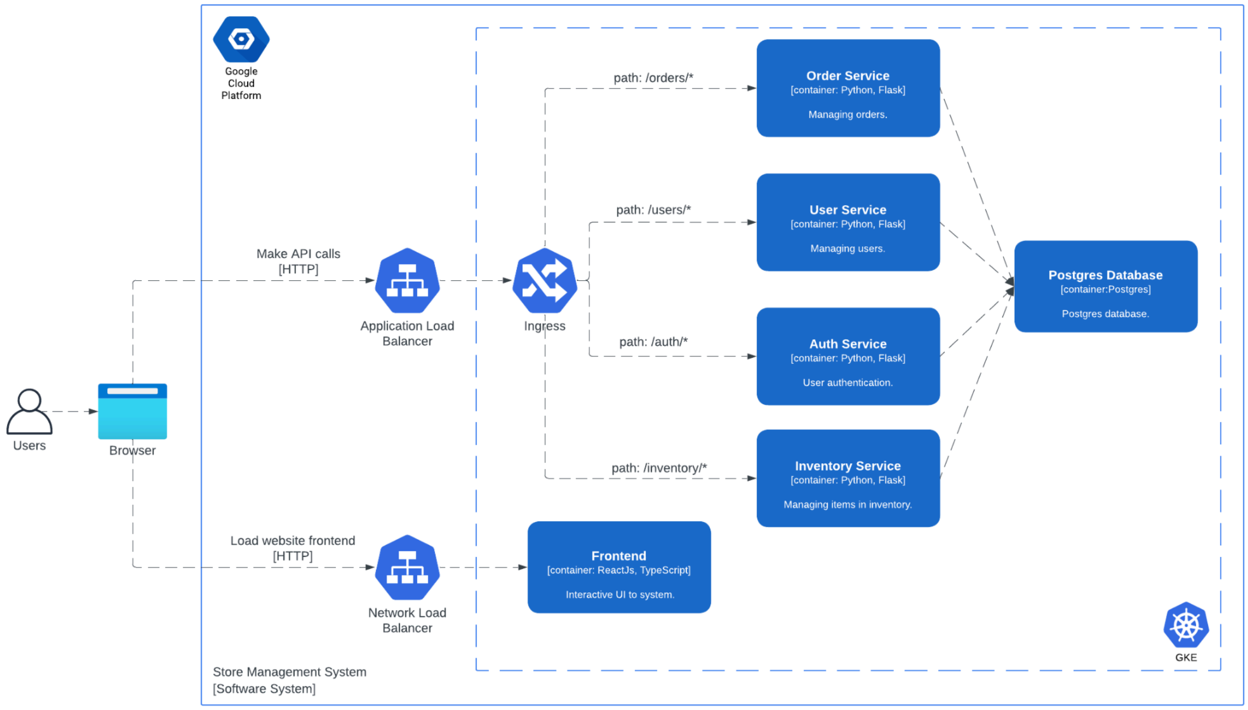

Figure 1. Architecture Design

The architecture design utilizes Google Cloud to host a robust application entirely on Google Kubernetes Engine (GKE). It includes six services: four backend services (order, auth, user, and inventory) built with Flask and Python, one frontend service using ReactJS and TypeScript, and a PostgreSQL database, all containerized for scalability and efficiency. Traffic management is handled by two load balancers: one directs requests to the backend via an ingress controller, while the other routes requests to the frontend. Users access the web frontend and make API calls using the IP addresses associated with the load balancers