CLOUD COMPUTING

PROJECT REPORT

Team: Homeuca Radu, Dumitrascu Ilinca

Introduction

This case study presents a groundbreaking application developed to revolutionize the way shoes are sold. Leveraging cutting-edge technologies including Google Maps for location services, Google Cloud Storage for data management, and Google Cloud Functions for seamless integration, the application provides users with an intuitive platform to discover and purchase shoes from local sellers.

The project aims to enhance the user experience, empower local businesses, and promote convenient and efficient shopping practices. This case study highlights the key features, implementation details, and potential impact of the application in the international/national retail industry.

The shoe-selling application project introduces a novel approach to connect shoe buyers with local sellers, offering a seamless shopping experience. By utilizing the power of location-based services and cloud technologies, the application bridges the gap between customers and nearby footwear vendors, fostering economic growth and convenience in the retail sector.

Technical details

The shoe-selling application is developed using a combination of frontend and backend technologies. The frontend is built with modern web development frameworks to provide an intuitive and responsive user interface. The backend utilizes Google Cloud Functions to handle various functionalities, including data retrieval, seller verification, and order processing. Google Maps API integration ensures accurate geolocation services, allowing users to discover nearby shoe vendors effortlessly.

The application offers a range of features that enhance the user experience and streamline the purchasing process:

- Seamless user interface with intuitive navigation and search functionality.
- Geolocation services powered by Google Maps API to identify nearby shoe sellers.
- Detailed seller profiles providing information about their products, ratings, and reviews.
- Integration with Google Cloud Storage and DataStore for efficient management of product images and descriptions.
- Customer review system to foster trust and transparency.

Technologies overview

The application was developed using a combination of technologies and services. Here's an overview of the key components:

Google Maps Integration: By integrating Google Maps API, users can easily find shoe sellers in their vicinity. The application provides real-time directions, enabling customers to navigate effortlessly to the desired store location.

Seller Information: The application fetches comprehensive information about each seller, including store details, contact information, operating hours, and customer reviews. This data is stored and retrieved from Google Cloud Storage, ensuring scalability, reliability, and efficient data management.

Product Catalog: The application showcases an extensive catalog of available shoes, including detailed descriptions, images, sizes, and prices. This information is retrieved from the sellers' databases using secure APIs, ensuring accurate and up-to-date product listings.

Ratings: The application provides a platform for customers to leave feedback and ratings for both the sellers and the purchased products. This feedback system fosters transparency, helps build trust, and encourages sellers to maintain high-quality services.

Business Canvas

Key Partners:

1. Shoe Sellers: Establish partnerships with various shoe sellers, including retail stores, boutiques, and online platforms, to list their products on the application.

2. Google: Leverage the capabilities of Google Maps, Google Cloud Storage, DataStore and Google Cloud Functions for location tracking, data storage, and seamless integration.

Key Activities:

1. Application Development: Continuously enhance and update the application to provide a user-friendly interface, seamless navigation, and secure transaction capabilities.

2. Seller Onboarding: Onboard new shoe sellers onto the platform, ensuring their products and information are accurately represented.

3. Data Management: Manage and maintain the data of sellers, products and user feedback securely.

Key Resources:

1. Technology Infrastructure: Maintain a robust and scalable technological infrastructure to handle user traffic, data storage, and application functionality.

2. Development Team: Employ a skilled team of developers, designers, and testers to ensure continuous improvement and bug-free operation.

3. Marketing and Sales: Allocate resources for marketing efforts to attract both customers and shoe sellers to the platform.

Value Propositions:

1. Convenient Shoe Shopping: Enable customers to easily locate shoe sellers, explore product catalogs, and make informed purchasing decisions.

2. Extensive Product Range: Offer a wide range of shoe options from multiple sellers, providing customers with a diverse selection.

3. Enhanced Seller Visibility: Increase the visibility and reach of shoe sellers by connecting them with a larger customer base.

Customer Segments:

1. Shoe Enthusiasts: Target individuals who are passionate about shoes and seek convenience and a wide range of options while making their purchase decisions.

2. Local Shoppers: Cater to customers who prefer to shop for shoes from nearby sellers, enabling them to easily find and visit physical stores.

3. Online Shoppers: Serve customers who prefer the convenience of online shopping, providing a platform to explore and purchase shoes from various sellers.

Customer Relationships:

1. User Support: Offer customer support through multiple channels, such as chat, email, and phone, to address inquiries, resolve issues, and provide assistance.

2. Feedback and Engagement: Encourage customers to provide feedback, ratings, and reviews to foster engagement and improve the overall user experience.

Channels:

Online Presence: Establish a website showcasing the application's features, benefits, and seller information, attracting customers and sellers through online channels.

Revenue Streams:

1. Premium Listings: Offer premium listing options to shoe sellers for increased visibility and promotional features.

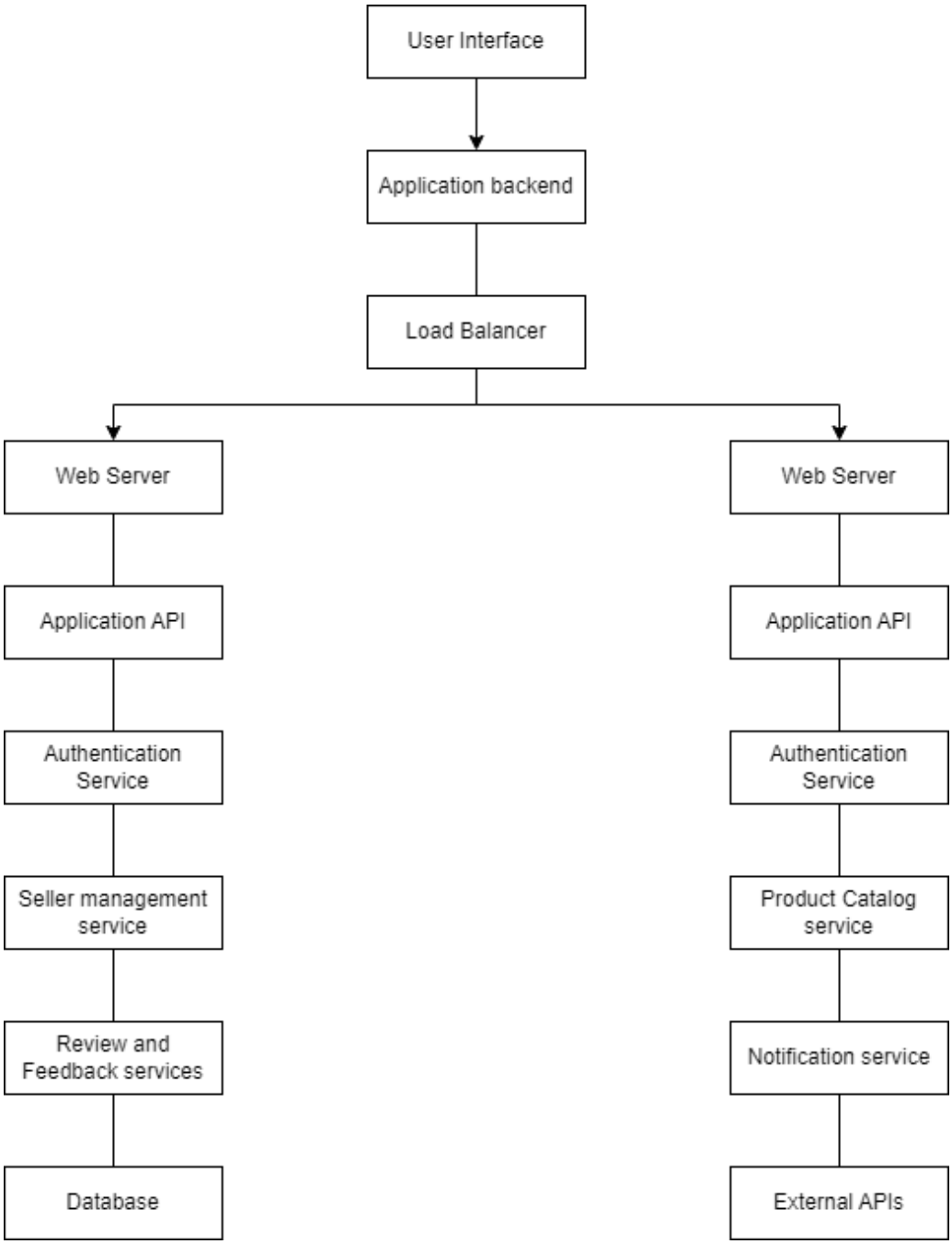
Cost Structure:

1. Development and Maintenance: Allocate resources for ongoing development, updates, and maintenance of the application.

2. Infrastructure and Hosting: Invest in reliable and scalable infrastructure to support the platform's operations.

3. Marketing and Promotion: Allocate funds for marketing campaigns, digital advertising, and promotional activities to attract customers and shoe sellers.

Architectural diagram

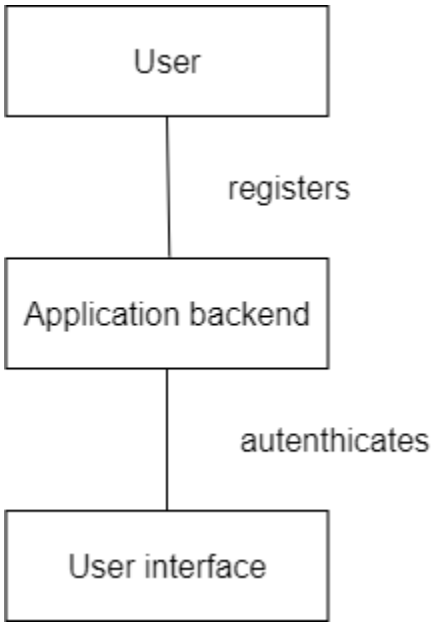


Components:

1. *User Interface:* The front-end interface where users interact with the application, including browsing shoes, searching for sellers, and making purchases.
2. *Application Backend:* The core of the application that handles business logic, user authentication, data processing, and communication with external services.
3. *Load Balancer:* Distributes incoming requests across multiple web servers to ensure scalability and high availability.
4. *Web Servers:* Host the application and serve web pages to users, handling HTTP requests and responses.
5. *Application API*: Exposes APIs for various functionalities such as user authentication, seller management, product catalog, order processing, and reviews/feedback.
6. *Authentication Service:* Handles user authentication and authorization, ensuring secure access to the application
7. *Seller Management Service:* Manages seller-related functionalities such as registration, verification, and updating seller information.
8. *Product Catalog Service:* Manages product information, including shoe details, pricing, availability, and synchronization with sellers' databases.
9. *Review Service:* Manages customer reviews and ratings, for sellers and products.
10. *Database:* Stores application data, including user profiles, seller information, product catalog, orders, and reviews.
11. *External APIs:* Integrates with external APIs, such as Google Maps API, for location services and other third-party services to enrich application functionality.

Use-case diagrams

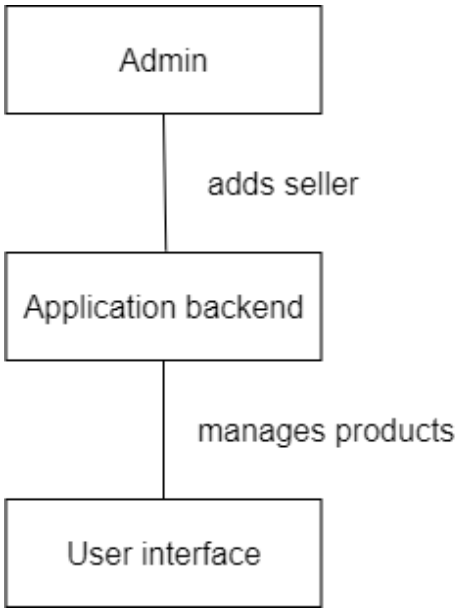
User Registration and Authentication use case diagram:



User registers: The user interacts with the application interface to create a new account.

User authenticates: The user interacts with the application interface to log in using their registered credentials.

Add seller and Manage products use case diagram:



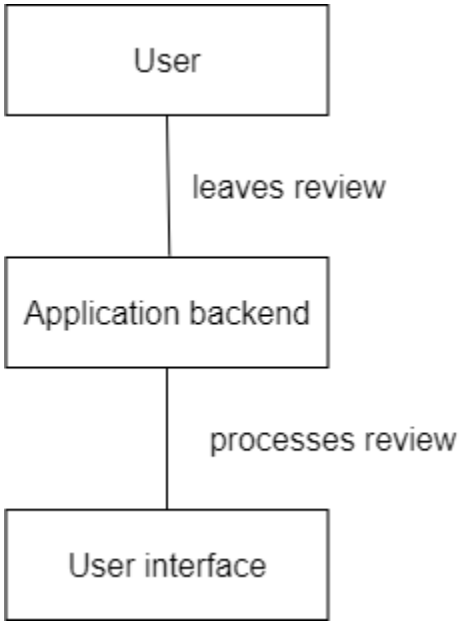
Admin adds seller: The admin interacts with the application interface to add a new seller to the platform.

Application backend handles seller addition: The application backend processes the admin's request to add the seller and updates the seller database accordingly.

Admin manages products: The admin interacts with the application interface to manage the products associated with a particular seller, including adding new products, updating product details, and removing products from the catalog.

User interface displays seller and product management: The user interface presents the admin with the necessary screens and options to add sellers and manage their products.

Review use case diagram:



User leaves review: The user interacts with the application interface to leave a review and feedback for a specific seller or product.

Application backend processes review: The application backend processes the user's review, including validating the review and storing it in the database.

User interface displays review and feedback: The user interface shows the user's review and feedback to other users, providing them with valuable insights about the seller or product.

APIs

Google Maps API: This API allows you to integrate location-based services into your application. You can use it to display store locations, calculate distances, generate directions, and provide map-based visualizations to help users find sellers near their location.

Google Cloud Storage API: With this API, you can integrate Google Cloud Storage into your application for securely storing various types of data, such as seller information, product images, and user-generated content like reviews and feedback.

Google Cloud Functions API: This API enables you to create and manage serverless functions that can be triggered by specific events in your application. You can use it to handle tasks like sending notifications, processing background tasks, and performing server-side operations.

Google Cloud Datastore API: The Google Cloud Datastore API allows you to store and retrieve structured data on Google Cloud Platform. You can use Datastore to store various types of information related to your application, such as user profiles, shoe listings, seller information, and order details.

Google Identity API (Google Sign-In): The Google Identity API, specifically the Google Sign-In feature, allows users to authenticate with your application using their Google accounts. By integrating this API, users can sign in to your shoe-selling application using their existing Google credentials, providing a convenient and secure authentication method.

Functionality Flow

Functionality: Browsing Shoes

Description: This functionality allows users to search for specific shoes or browse through different categories of shoes available on the platform.

Actors:

- User
- Application Backend

Pre-conditions:

- User is logged in.
- User has access to the internet.
- Application backend is operational.

Basic Flow:

1. User opens the application.
2. User is presented with the home screen displaying featured shoes and categories.
3. Application backend sends the retrieved shoe listings to the user interface.
4. User interface displays the shoe listings to the user.
5. User can scroll through the shoe listings.
6. User can click on a shoe listing to view more details.
7. User can add desired shoes to the favourites.

Exceptional Flows:

1. If there is an error in retrieving shoe listings
2. Application backend logs the error and displays an error message on the user interface.
3. User can try again later or contact customer support.