CNG 495 FALL -2024

TERM PROJECT PROGRESS REPORT II

SAFEBOOK

MEMBERS

Wendy Susan Ondola – 2413490

Aaron Bandado - 2476984

1. **INTRODUCTION**

1.1. CLOUD BASED SERVICE BOOKING MANAGEMENT SYSTEM :SAFEBOOK

SafeBook is an all-inclusive cloud-based booking platform that enables small businesses to easily manage their appointments and schedules. Its main purpose is to enable small businesses to allow their customers to create appointments based on their availability on a publicly shared schedule and equally manage them seamlessly and efficiently. For example, a barber has many clients and would like to manage all their appointments. This software would allow the barber to specify the time slots when they are available and the services they offer along with any other vital information and their customers would be able to book an appointment with them.

Key features will include user authentication, role-based access (either the businesses or the clients), calendar integration (for efficient booking functionality), and automated reminders to reduce no-shows.

There are three different cloud delivery models in cloud computing namely Platform As A Service(PAAS), Software As A Service(SAAS), and Infrastructure As A Service (IAAS). In this project, we will implement Software As A Service (SAAS) as the cloud delivery model and later deploy our web application on Vercel.

1.2. BENEFITS OF SAFEBOOK.

Customers

Convenience: Customers can easily book appointments at their convenience, 24/7, without the need to call or visit the business in person.

Time-saving: Customers don't need to wait for confirmation over the phone; they get immediate confirmation for available time slots.

Automated reminders: Customers will receive automated reminders, reducing the chances of missed appointments.

Access to business services: Customers can view a business's services and available time slots before making an appointment therefore making the booking process more transparent and efficient.

Business Owners

Efficient scheduling: Business owners can easily manage their appointments and avoid double-booking or missed appointments.

Automated reminders: The system sends reminders to customers, reducing no-shows and cancellations.

Simplified client management: Businesses can keep track of their clients and services offered without any manual work.

Time-saving: Businesses can save time by not needing to handle appointment scheduling manually or answer calls for bookings.

Growth opportunities: By offering an easy online booking system, businesses can attract more customers who prefer digital booking.

1.3. PROJECT NOVELTIES

Our novel idea

SafeBook is a simple cloud-based platform designed for small businesses to manage appointments. Unlike general scheduling tools, it focuses on ease of use, works well with existing calendar tools, and is customized for businesses like hair salons, barbershops, and small consultants. It also includes helpful features like automatic availability management and reminders, which help reduce no-shows and improve customer satisfaction.

Safebook contributes to the business management field

SafeBook helps minimize administrative work, boost booking efficiency, and enhance customer service by providing small businesses with an accessible and reasonably priced online appointment management solution. It helps small businesses undergo a digital transformation that enables them to reach a wider audience and function more professionally.

1.4. WIDELY USED SIMILAR PROJECTS

There are various online scheduling platforms that serve similar functions to SafeBook, such as:

- **Acuity Scheduling**: A popular scheduling platform for small businesses that allows customers to book appointments online.

- **Calendly**: A widely-used scheduling tool that allows users to set their availability and allows others to book meetings during those times.
- **Square Appointments**: A booking software integrated with Square's payment system, popular with small businesses.

1.5 GITHUB REPOSITORY

https://github.com/aaronaminubandado/SafeBook

2. PROJECT STRUCTURE

2.1. PARTS OF PROJECT AND FUNCTIONS

Frontend

The frontend is the user interface that allows customers and business owners to interact with the application. It handles the display of information and the user's experience, such as booking appointments, viewing available time slots, and managing schedules.

Backend

Backend is the interface between the frontend and the backend where processing of data takes place. Once

Database

The database stores all necessary data, such as customer information, appointment details, and business schedules. It is used to ensure that the data is readily available and up-to-date for both customers and business owners.
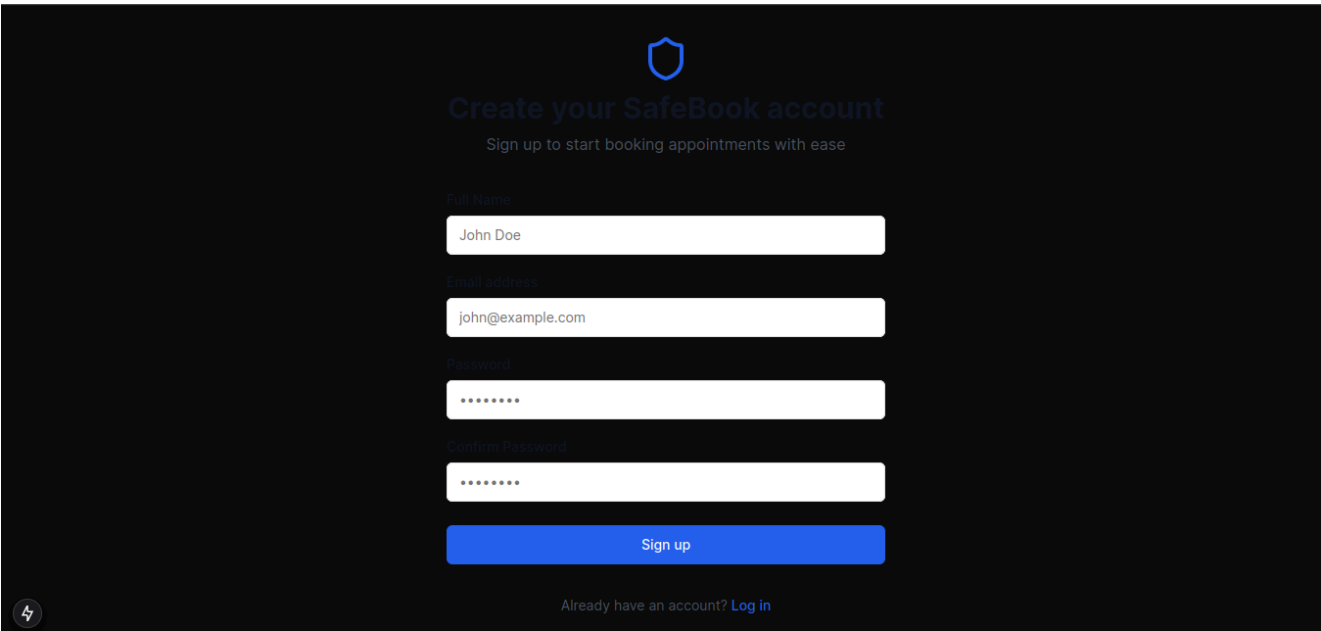
2.2. CLOUD SERVICES UTILIZED FOR EACH PART

Project Part	Cloud Service
Frontend	Firebase Authentication
Backend	Firebase
Database	Firestore

2.3. USER MANUAL FOR IMPLEMENTED FUNCTIONS AND SCREENSHOTS
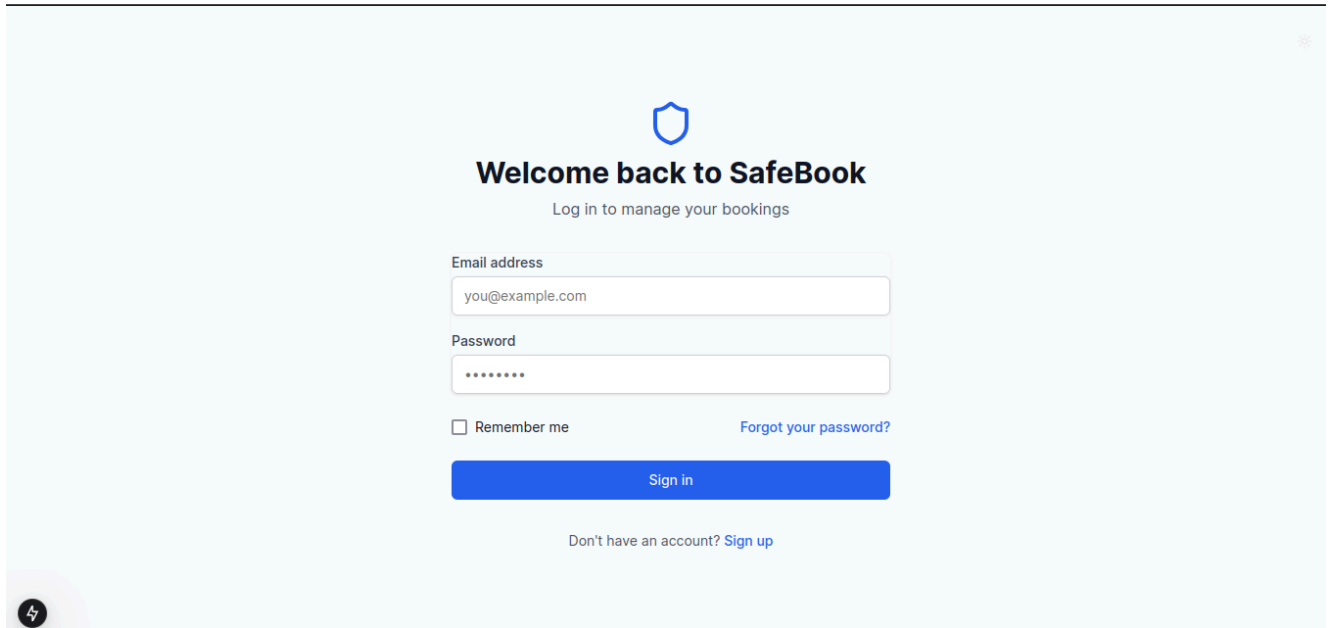
Customer Tutorial

1. Create an account

2. Login to your account with your email and password

3. Search for business you would like to book

4. Enter the appointment details and book the appointment

Business Tutorial:

1. Create an account and login similar to customer.
2. Click on manage schedule to add you business schedule.

3. Click appointment to see all appointments

Business Dashboard

Total Bookings	
254	
+20% from last month	

Appointments	
12	

Schedule   Business Info   Appointments

Appointment Bookings
View and manage your upcoming appointments.

📅 Pick a date

Customer	Service	Date & Time
Unknown Customer	Retwist	December 28th, 2024 12:43 PM
Unknown Customer	Haircut	December 26th, 2024 9:59 PM

4. Update business information in business info tab.

Business Information
Update your business details and services.

Business Name
Loctician's Credence

Description
Hairstylist

Address
12 Jakob Sinen Way, Juriam

Phone
+905488577499

Email
bandiaiki1@gmail.com

Website
https://locscredence.com

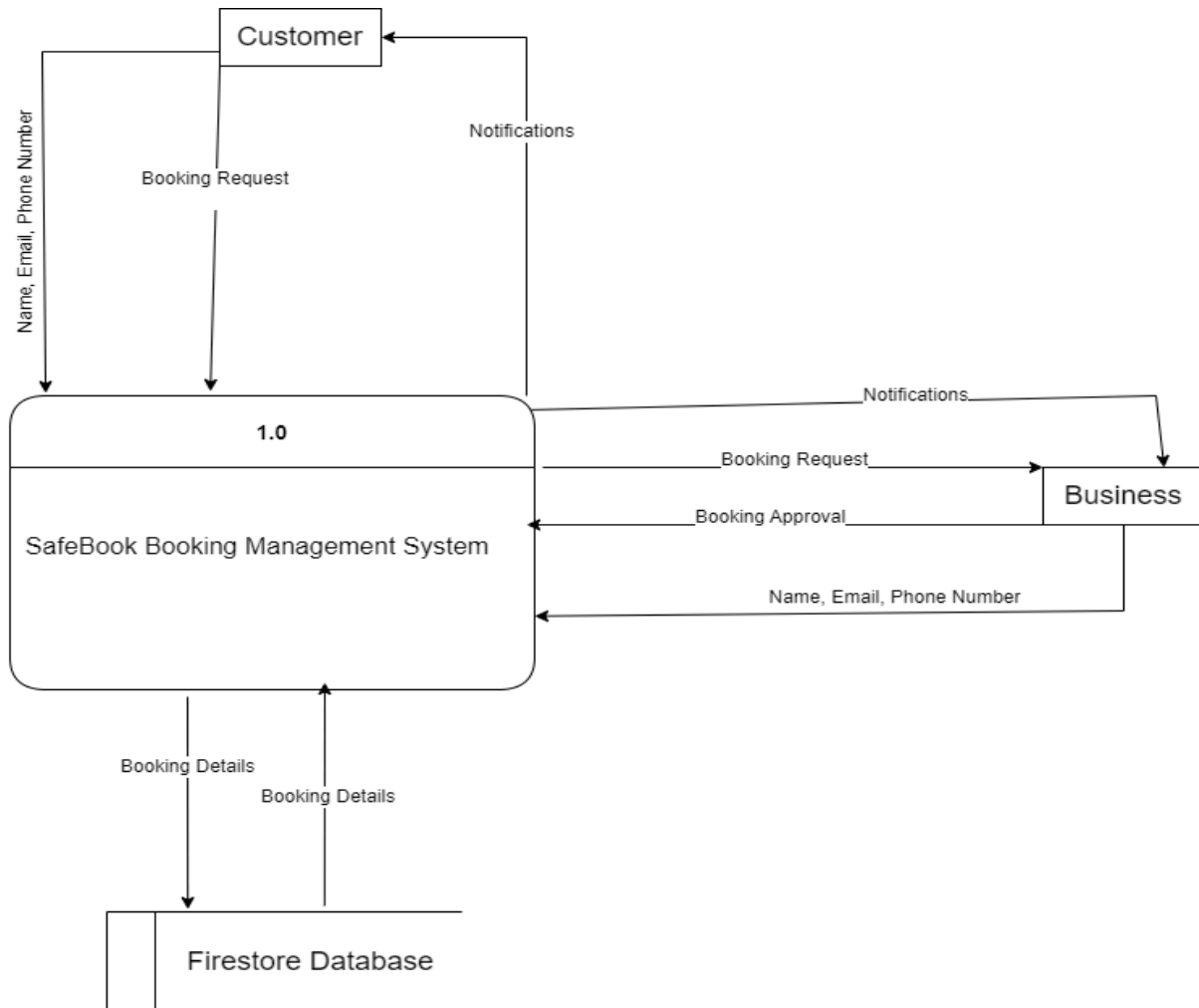## 2.4. INTERACTION AND INFORMATION FLOW BETWEEN PARTS

Client Service Interaction Diagram
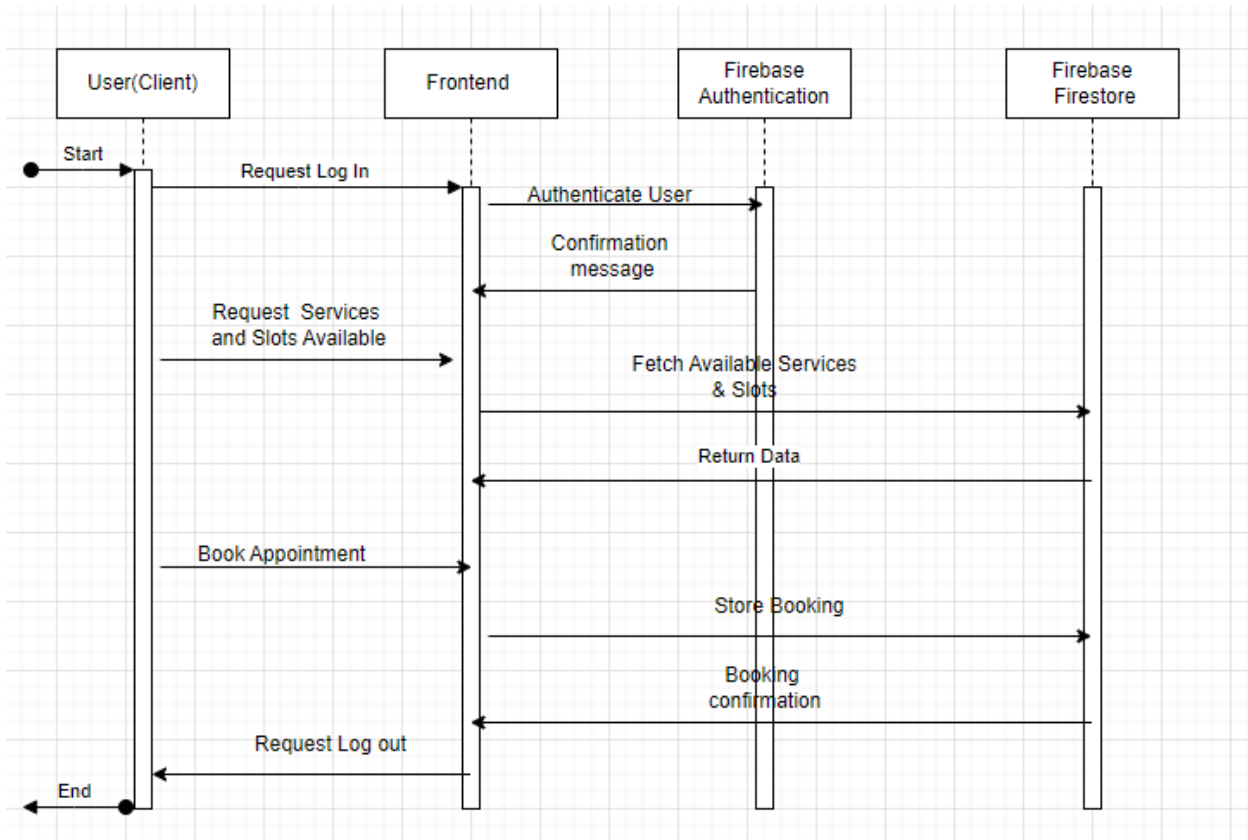
Data flow diagrams

Level 0

Customer

Name, Email, Phone Number

Booking Request

Notifications

1.0

SafeBook Booking Management System

Notifications

Booking Request

Business

Booking Approval

Name, Email, Phone Number

Booking Details

Booking Details

Firestore Database
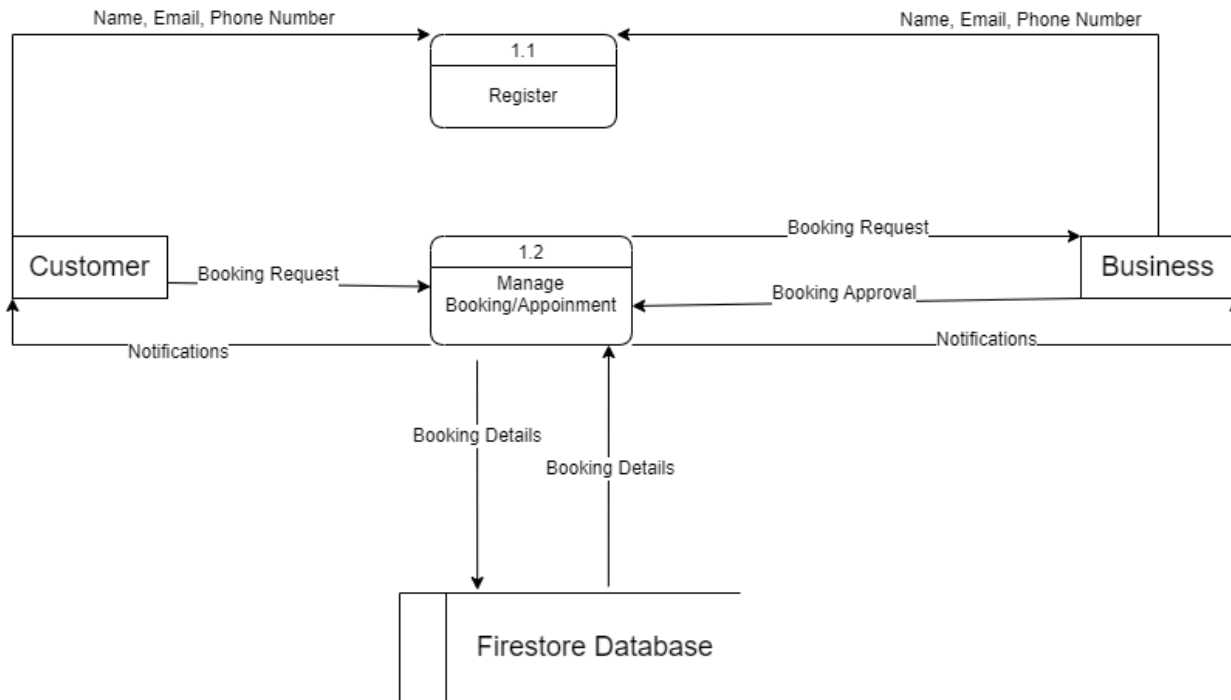
Level 1

2.5 TECHNOLOGIES USED FOR EACH PART

Frontend

For the implementation of this part, we used React within the Next.js framework. This provides us with a wide range of advantages including but not limited to faster page load, file-based routing, the ability to create APIs within the same project, easy deployment, etc.

Backend

Using the NextJS react framework we were able to have routing for our web application using NextJS page routing. Also, for the backend, we used Firebase (BAAS-Backend As a Service) to enable us to handle our requests. The code for the development, Javascript/Typescript was used .

Firebase is a comprehensive app development platform owned by Google, designed to help developers build, improve, and grow applications across various platforms.

Firebase is deeply integrated with Google Cloud Platform (GCP), enabling it to leverage Google's scalable infrastructure and powerful cloud services.

Database

Firestore, a NoSQL database in the Firebase ecosystem was used to implement the database. Communication with Firestore was managed through Firebase's SDK, allowing for seamless real-time data handling. We also used Firebase Cloud Functions for additional server-side logic as needed.

2.6 TUTORIALS/EXPLANATIONS /DIFFICULTIES

 During the development of SafeBook, we encountered a few challenges, such as:

- **Authentication Setup:** Initially, we had trouble integrating Firebase Authentication to handle user logins and role-based access control (business owners vs. customers). We followed tutorials to integrate Firebase Authentication into our Next.js application, but understanding how Firebase works with Next.js took some time.
- **Real-time Database Integration:** Firestore offers real-time database features, which meant we had to ensure that appointment data updated instantly across different clients. This was tricky to implement initially, but with Firebase SDK documentation, we figured out how to handle real-time data effectively.
- **Deployment Issues:** Deploying on Vercel had some challenges, such as setting up environment variables and ensuring proper deployment from the GitHub repository. After following the deployment guides and fixing minor configuration issues, the app worked perfectly

3. **PROJECT STATISTICS**

Database Type and Cloud Service Limits:

- Database Type: NoSQL (Firestore)

- Cloud Service: Firebase
- Maximum Storage Limit (Firestore):
  - Free tier: 1 GB storage for data and 50,000 reads/day, 20,000 writes/day.
  - Paid tier: Offers up to 1 TB storage and higher read/write limits, depending on the plan chosen.