CO3261 Cloud Computing and Applications

Mini Project

**InvoiceEase**

InvoiceEase is a cutting-edge mobile application designed to streamline the invoicing process and enhance sales efficiency for sales representatives and businesses of all sizes. The app leverages the latest technology to provide a comprehensive invoicing solution that simplifies the creation, customization, and distribution of invoices.

**Key features of InvoiceEase**

* Effortless Invoicing:

With InvoiceEase, sales representatives can create and customize professional invoices in a matter of minutes.

* User Login**:**

Secure user login functionality, ensuring that only authorized users can access the app and its features.

* Digital Invoicing:

InvoiceEase eliminates the need for manual, paper-based invoicing processes by enabling sales representatives to *generate invoices digitally*. This not only saves time but also reduces the risk of errors associated with manual data entry.

* Shop Tracking:

For sales representatives on the go, InvoiceEase includes a shop tracking feature that uses GPS technology to ***suggest the next shop near the user's current location***. This helps sales reps *optimize their routes* and cover their target areas more efficiently.

* Sales Support:

InvoiceEase provides valuable support for sales representatives by *highlighting important shops based on previous order values*. This allows sales reps to *prioritize their visits* and focus on high-potential clients.

* Cloud-Based:

InvoiceEase stores all data securely in the cloud, ensuring that sales representatives can access their invoices and sales information from anywhere, at any time.

* Performance Tracking:

The app includes a home tab that displays,

* + - The daily target value to be covered,
    - The covered value till a given time period, and
    - The required amount.

This gives sales representatives a clear *overview of their progress* towards their sales targets.

**Technology**

* Flutter frame work: For the mobile app development
* Geolocator Package: for location-based services
* Firebase Firestore: For storing and managing data related to invoices, sales, and user information.
* Firebase Authentication: For user authentication..
* Google Maps API (Optional): for displaying maps and integrating location services.
* Cloud Hosting Service: For hosting the backend services, Eg:Amazon Web Services (AWS).
* Design Tools (Optional): Figma for designing the app's user interface.

**Milestones**

* UI Design:
  + Design app interfaces using Flutter
* User Authentication and Data Management:
  + Implement authentication (user registration and login screens).
  + Create Firestore collections for users, invoices, and shops.
* Implement Invoicing Features:
  + Add functionality to save and retrieve invoices from Firestore.
  + Add functionality to calculate invoice total value and related parameters.
* Shop Tracking and GPS Integration:
  + Configure the Geolocator package to access the device's location.
  + Implement **algorithms to suggest the next shop based on GPS**.
    - Only show nearest shops (with in particular range).
    - Periodically update the suggestions based on the user's movement.
    - Eg: Nearest Neighbor algorithm, K-Nearest Neighbors (k-NN) algorithm
* Sales Support Features: Home Screen and Highlight important shops.
  + Develop **algorithms to highlight important shops based on order values**.
    - Calculate the total order value for each shop by summing the values of all previous orders then take average.
    - Develop algorithm to filter nearest shops based on highest average total value to highlight important shops to be covered in that day.
  + Implement the sales performance tracking screen.
  + Implement features to track daily sales targets and progress.
* Cloud Service Configuration
  + Set up a Cloud Platform and implement backend functionality.

**Basic Interfaces**

* User Login and Signup Interface:
  + Parameters: Username(Email), password
* Invoice Interface:
  + Parameters: Invoice number, customer information (name, address, contact), items (description, quantity, price), total amount, due date.
* Shop Interface:
  + Parameters: Shop name, location (latitude, longitude), order history, order values.
* Dashboard Interface:
  + Parameters: Daily target value, covered value till a given time period, required amount to be covered, Summary of invoices, sales targets, and performance metrics.

**Need to find**

* Algorithm for Finding Nearest Shop and to highlight important shop.
* How data storing and retrieve.
* How implement Backend in Cloud Server.
* How configure Cloud Service.
* How connect App with Backend Implemented in Cloud Server.