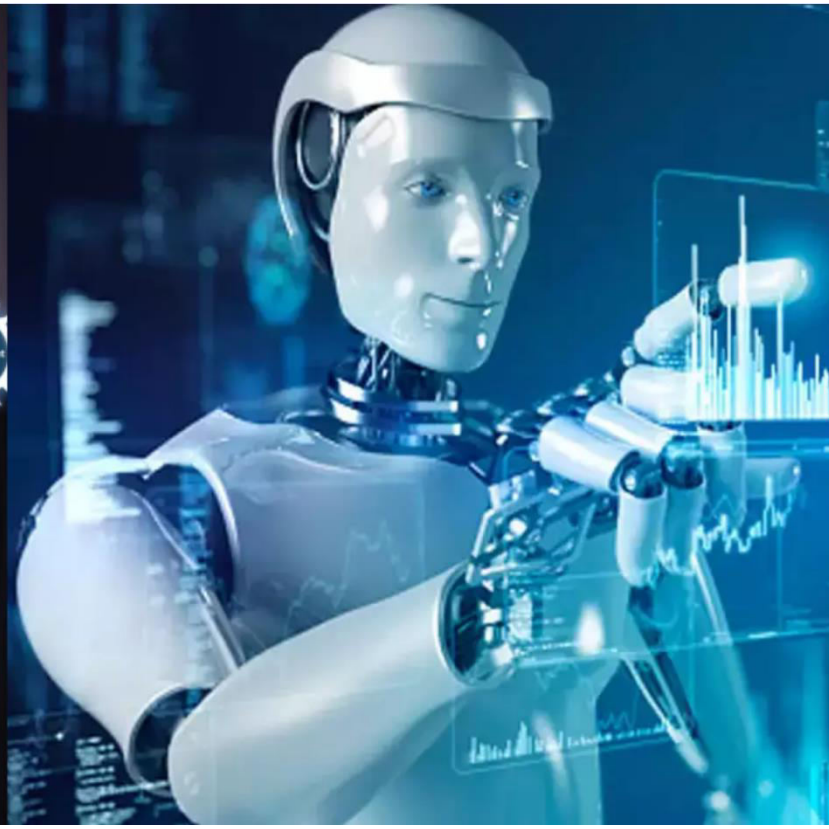
IJIRSET
International Journal of Innovative Research in Science | Engineering | Technology

ISSN(O): 2319-8753
ISSN(P): 2347-6710

International Journal of Innovative Research in Science Engineering and Technology (IJIRSET)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)



Impact Factor: 8.699

Volume 14, Issue 3, March 2025

International Journal of Innovative Research of Science, Engineering and Technology (IJIRSET)

Impact Factor 8.699

|www.ijirset.com |A Monthly, Peer Reviewed & Refereed Journal| e-ISSN: 2319-8753| p-ISSN: 2347-6710|

Volume 14, Issue 3, March 2025

|DOI: 10.15680/IJIRSET.2025.1403161|

E-Commerce App Development Using Flutter

Bhavya Fulwala, Kirtan Barot, Prof. Dinesh Swami

Department of Computer Science and Engineering, Parul Institute of Engineering and Technology, Vadodara, Gujarat, India

ABSTRACT: This paper describes the creation of a cutting-edge Flutter e-commerce mobile appli- cation tailored for nearby companies. By offering a smooth shopping experience with real-time product listings, secure payment gateways, and effective order tracking, the app seeks to close the gap between small businesses and consumers. The applica- tion saves development time and money by utilizing Flutter's cross-platform features to guarantee consistent performance on both the iOS and Android platforms. The backend solution for managing dynamic data, cloud storage, and user authentication is Firebase. Personalized recommendations based on user behavior, store recommen- dations based on geography, and an easy-to-use user interface that is responsive to mobile devices are some of the app's most notable features. By digitizing their store- fronts and improving client engagement, the solution empowers small business owners and eventually promotes the growth of local commerce.

Mobile applications are having a progressively more significant role in our day-to-day lives. Ever since November 2016, there is more network traffic made by mobile devices compared to desktops or laptops. To dispense it to most of the users, a mobile appli- cation needs to familiarize itself with two independent platforms which are Android and iOS.

I. INTRODUCTION

In today's fast-paced digital environment, ecommerce has become a cornerstone for business growth, allowing entities to reach a wider demographic and streamline their operations. However, small local firms sometimes struggle to build an online pres- ence due to hefty development costs and technical difficulties. This project tries to overcome these difficulties by constructing an ecommerce mobile application using Flutter, a versatile and durable framework for cross-platform app development. The key features of the application are user behavior-based personalized product sugges- tions, location-based store recommendations, and a reward system for loyalty. These features are designed to enhance customer engagement and retention while helping small businesses grow their online presence. The responsive and intuitive user inter- face guarantees accessibility and usability across a broad audience.

By merging the ease of Flutter with the powerful backend features of Firebase, this project seeks to build a groundbreaking ecommerce platform that narrows the dis- tance between local stores and customers, promoting community-oriented commerce and tech development.

II. LITERATURE REVIEW

1. Research by **Wu et al. (2018)** :
This study compares the performance of two popular cross platform mobile applica- tion development frameworks, Flutter and React Native. As the number of mobile users continues to grow, the ability to target multiple platforms using a single code- base is increasingly important for developers and compa nies. We conducted three manual UI tests; scrolling through a list, testing the camera, and filtering a large dataset to measure the performance of the frameworks in terms of CPU usage, mem- ory usage, and janky frames on an Android device. The results indicate that Flutter may provide better performance in specific situations when compared to React Na- tive. The study contributes to the existing research by providing additional in sights into the performance of these frameworks under specific test scenarios. **[1]**.

2. Research by **Aakanksha Tashildar, Nisha Shah, Rushabh Gala, Trishul Giri, Pranali Chavhan**:
In today's fast-paced world, cross-platform mobile app development is a top priority. Developers often face a

International Journal of Innovative Research of Science, Engineering and Technology (IJIRSET)

Impact Factor 8.699

|www.ijirset.com |A Monthly, Peer Reviewed & Refereed Journal| e-ISSN: 2319-8753| p-ISSN: 2347-6710|

Volume 14, Issue 3, March 2025

|DOI: 10.15680/IJIRSET.2025.1403161|

tough choice: build the same app multiple times for different operating systems (like iOS and Android) or settle for a lower-quality solution that sacrifices performance for portability. Enter Flutter—an open-source SDK that lets you create high-performance, reliable apps for both iOS and Android with ease.

Flutter's standout features include Just-in-Time (JIT) compilation, which allows code to be compiled during runtime, making development faster and more flexible. It also supports Ahead-of-Time (AOT) compilation, turning code into native machine lan- guage for lightning-fast performance.

One of Flutter's most loved features is hot reload, which lets you instantly see changes as you tweak your code. Fix bugs, experiment with designs, or add features without restarting the app—it's a game-changer for productivity.**[2]**

3. Conference paper by **Boukhary et al. (2019)**:

Discussed clean architecture practices in Flutter, demonstrating how its layered ap- proach allows for better scalability and maintainability of mobile applications [3].

Additionally, studies have shown that Dart's strong typing and async-await sup- port enable more reliable and responsive applications, crucial for modern e-commerce platforms.

The integration of Firebase as a backend solution has also been extensively re- searched, highlighting its seamless real-time database management, user authentica- tion, and cloud storage, making it an ideal companion for Flutter apps.

This literature review lays the foundation for understanding the strengths and limitations of existing technologies, providing the necessary insights to develop a robust and innovative e-commerce application using Flutter and Dart.**[3]**

III. SYSTEM DESIGN AND ARCHITECTURE

A. System Overview

The ecommerce application developed using Flutter is designed to provide a seam- less shopping experience for users while enabling local businesses to showcase their products effectively. The system follows a client-server architecture, consisting of a frontend built with Flutter and a backend powered by Firebase. The frontend serves as the user interface, allowing customers to browse products, add items to their cart, and complete purchases. It features an intuitive UI/UX design, real-time product search, secure user authentication, and order tracking.

The backend utilizes Firebase services, including Firestore for database manage- ment, Firebase Authentication for secure login and registration, and Cloud Functions for handling business logic. Firestore, a NoSQL cloud database, stores structured data in collections and documents, ensuring real-time synchronization between users and the server. The stored data includes product information such as names, prices, descriptions, and stock levels, along with user profiles, order history, and business details.

User authentication is implemented through Firebase Authentication, supporting various methods like email/password, Google sign-in, and phone number authentica- tion. Additionally, the app offers an admin panel, developed using Flutter web or managed through the Firebase Console, allowing business owners to add or update product listings, view customer orders, and manage store data efficiently.

The app's workflow begins with users registering or logging in, after which they can browse the product catalog retrieved from Firestore. Selected items are added to the cart, and payment processing is handled through integrated gateways. Once an order is confirmed, the data is stored in the database and simultaneously reflected in the admin panel, where business owners can track and manage incoming orders in real-time.

B. System Architecture

The system architecture of the ecommerce application developed using Flutter fol- lows a client-server model, ensuring seamless interaction between users, businesses, and backend services. It is divided into three main layers: the Presentation Layer, Business Logic Layer, and Data Layer. The Presentation Layer is built using Flutter, providing a cross-platform user interface compatible with both Android and iOS. This layer handles user interactions, such as browsing products, searching, adding items to the cart, and processing authentication and payments. Flutter's reactive framework ensures dynamic UI updates, smooth animations, and an engaging user experience.

The Business Logic Layer serves as the intermediary between the user interface and the backend, processing user requests and managing state. It employs state management techniques like BLoC (Business Logic Component) or Provider to handle essential functions, including product filtering, cart updates, and order

International Journal of Innovative Research of Science, Engineering and Technology (IJIRSET)

Impact Factor 8.699

|www.ijirset.com |A Monthly, Peer Reviewed & Refereed Journal| e-ISSN: 2319-8753| p-ISSN: 2347-6710|

Volume 14, Issue 3, March 2025

|DOI: 10.15680/IJIRSET.2025.1403161|

confirmation. By keeping the business logic separate from the UI, this layer ensures clean code practices, making the app more scalable and maintainable.

The Data Layer is powered by Firebase, which provides real-time data manage- ment and secure backend operations. It includes Firestore, a NoSQL cloud database for storing structured data such as product details, user profiles, and order history. Firebase Authentication is integrated to facilitate secure user sign-ups, logins, and session management using methods like email/password, Google sign-in, and phone authentication. Cloud Functions handle server-side operations, such as processing payments, updating order statuses, and sending notifications. Firebase Storage is used to store product images and other media, ensuring fast and reliable content delivery.

Additionally, the system features an Admin Panel, developed using Flutter web or accessed through the Firebase Console, enabling business owners to manage their product catalogs, view customer orders, and update store information. Any changes made through the admin panel are instantly synchronized with the app's user interface using Firestore's real-time database capabilities.

The app also integrates various third-party APIs to enhance functionality. A secure payment gateway processes transactions, while Google Maps API supports location-based services, such as displaying store locations or calculating delivery routes. Firebase Cloud Messaging (FCM) is employed to send push notifications, keeping users informed about order updates and promotions.

The workflow of the app begins with users logging in and browsing the product catalog. When a user selects items and proceeds to checkout, the business logic layer processes the order, communicates with the backend to authenticate the user, and finalizes the purchase through the payment gateway. Once an order is confirmed, the data is updated in Firestore and reflected in both the user's app and the admin panel in real-time, ensuring smooth coordination between customers and business owners.

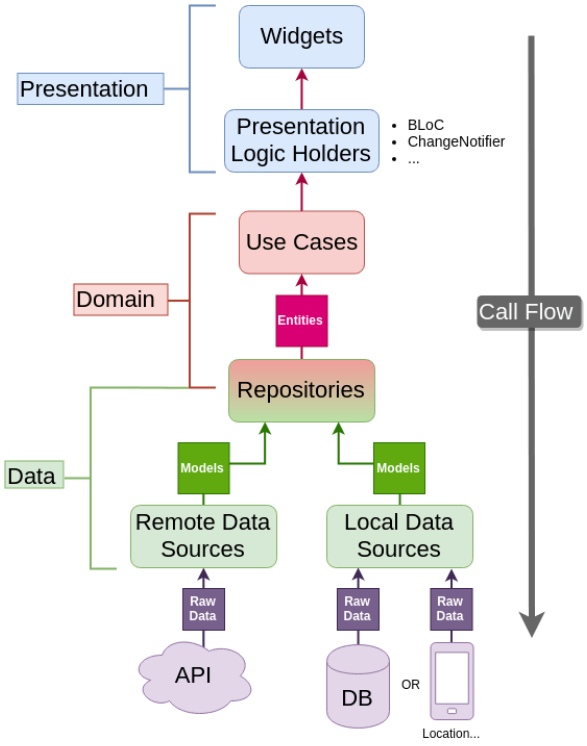


Figure 1: Architect

C. System Design

1. Client-side (Frontend)

The client-side is coded in Flutter to create a cross-platform mobile app that is com- patible with both Android and iOS devices. The user interface (UI) is created with a

International Journal of Innovative Research of Science, Engineering and Technology (IJIRSET)

Impact Factor 8.699

|www.ijirset.com |A Monthly, Peer Reviewed & Refereed Journal| e-ISSN: 2319-8753| p-ISSN: 2347-6710|

Volume 14, Issue 3, March 2025

|DOI: 10.15680/IJIRSET.2025.1403161|

responsive design to offer an easy-to-use shopping experience, such as product naviga- tion, search, cart control, and safe checkout. State management is provided through Provider or BLoC (Business Logic Component) to ensure smooth data transfer and live updates without sacrificing app performance.

2. Server-side (Backend)

The backend is fueled by Firebase services, which execute the core functions: Authentication: Firebase Authentication processes user sign-up, logins, and secure sessions, including email/password, Google sign-in, and phone authentication.

Cloud Functions: Serverless functions execute custom backend logic, including pay- ment processing, sending push messages, and updating order status. Real-time Up- dates: Firestore provides real-time data synchronization, meaning product listings, cart items, and order information update in real-time on user devices as well as the admin panel.

3. Database

The application employs Cloud Firestore, a NoSQL database, to hold structured data, including:

Product Information: Product names, descriptions, prices, and image URLs. User Profiles: Names, contact information, and order history. Order Details: Order sta- tus tracking, payment confirmation, and delivery status. Moreover, Firebase Storage hosts product images and other multimedia material to provide quick and secure ac- cess.

4. Admin Panel

A separate Admin Panel enables store owners to control the content of their store. Constructed with Flutter Web or accessed via Firebase Console, the panel offers the facility to:

Add, modify, or delete product listings. Track customer orders and change statuses. See user analytics and sales figures.

5. Workflow

The system works as follows:

User Interaction: Users sign up or log in using Firebase Authentication. Product Browsing: Product information is retrieved from Firestore and presented via Flutter UI. Cart and Checkout: Chosen products are placed into the cart, and order infor- mation is saved to Firestore. Payment Processing: Cloud Functions process payment gateways and update order statuses for successful payments. Real-time Sync: Fire- store makes sure updates — such as product stock or order status — are immediately displayed in both user app and admin dashboard.

D. Important Diagrams

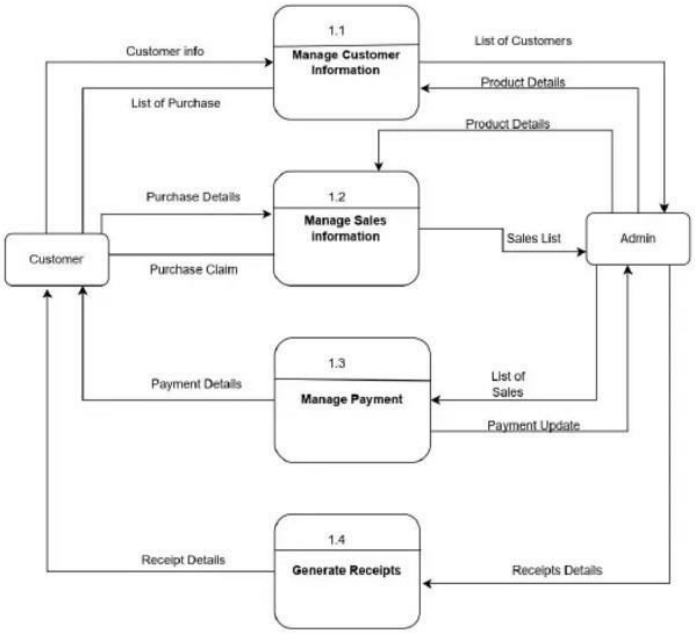


Figure 2: Data Flow Diagram

International Journal of Innovative Research of Science, Engineering and Technology (IJIRSET)

Impact Factor 8.699

|www.ijirset.com |A Monthly, Peer Reviewed & Refereed Journal| e-ISSN: 2319-8753| p-ISSN: 2347-6710|

Volume 14, Issue 3, March 2025

|DOI: 10.15680/IJIRSET.2025.1403161|

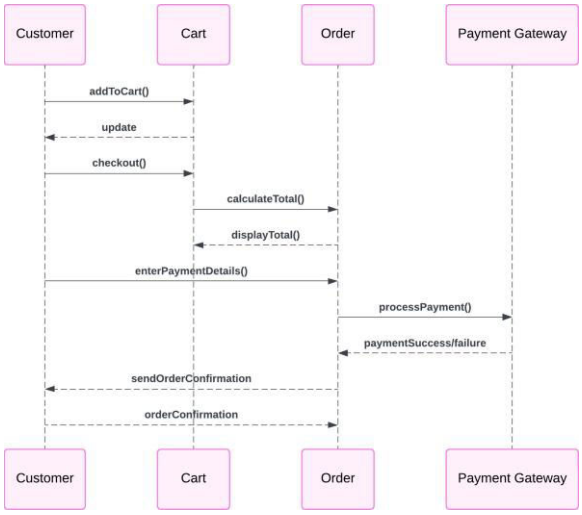


Figure 3: Sequence Diagram

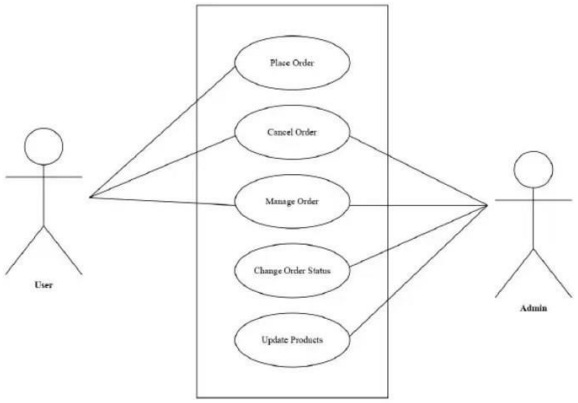


Figure 4: Use Case Diagram

IV. METHODOLOGY

The methodology defines the step-by-step procedure applied within the design, devel- opment, and deployment of the ecommerce mobile application. The approach adopts an Agile software development lifecycle (SDLC) to maintain iterative development, user feedback inclusion, and elasticity in feature add-ons. The development process consists of the below stated key stages:

1. Requirement Analysis The first step entailed the requirements gathering and anal- ysis of both administrator (business owners) and end-user (customer) needs. The main functions identified were:

   User features: Browsing through products, managing carts, safe checkout, or- der tracking, and user authentication. Admin features: Product management, order status updating, and analytics for sales. Technical requirements: Cross-platform op- erability (iOS and Android), real-time sync of data, and secure payments integration.
2. System Design System architecture was developed from the requirements into three layers:

   Frontend (Client-side): Developed with Flutter to offer a responsive and seam- less user experience. Backend (Server-side): Governed by Firebase, managing user authentication, cloud functions, and push notifications. Database: Real-time stor- age using Cloud Firestore for products, user data, and order details. A well-defined entity-relationship (ER) model was developed to represent associations between users, products, and orders, to maintain data integrity.
3. Implementation The application was implemented with Flutter and Dart, in a modular way:

International Journal of Innovative Research of Science, Engineering and Technology (IJIRSET)

Impact Factor 8.699

|www.ijirset.com |A Monthly, Peer Reviewed & Refereed Journal| e-ISSN: 2319-8753| p-ISSN: 2347-6710|

Volume 14, Issue 3, March 2025

|DOI: 10.15680/IJIRSET.2025.1403161|

User Interface (UI): Created with Flutter widgets to provide a dynamic and re- sponsive UI. State Management: Controlled through Provider to maintain an efficient flow of data between UI and backend. Firebase Integration: Used for authentication, database management, and cloud functions. Payment Gateway: Integrated through third-party APIs such as Razorpay or Stripe for secure and seamless transactions. The admin panel was created with Flutter Web to enable business owners to handle product listings and monitor orders.

4. Testing Both unit testing and integration testing were conducted:

Unit tests: Tested isolated functions such as adding products to the cart or com- puting order amounts. Integration tests: Verified smooth communication between the app and Firebase services. User testing: With a limited group of beta users to obtain feedback on UI/UX and responsiveness. Testing utilities such as Flutter Test and Firebase Emulator Suite were utilized to test backend processes without impacting live data.

5. Deployment After testing, the app was deployed with the following:

Firebase Hosting: The admin panel. Google Play Store and Apple App Store: Flutter's cross-platform feature was utilized to package the mobile app, which was made available on both platforms.

6. Maintenance and Future Enhancements After deployment, the app is on a model of continuous improvement. Feedback is gathered in real-time using Firebase Ana- lytics, enabling the team to pinpoint areas for improvement. Future improvements are AI-driven product recommendations and personalized push notifications based on machine learning algorithms.

V. RESULTS

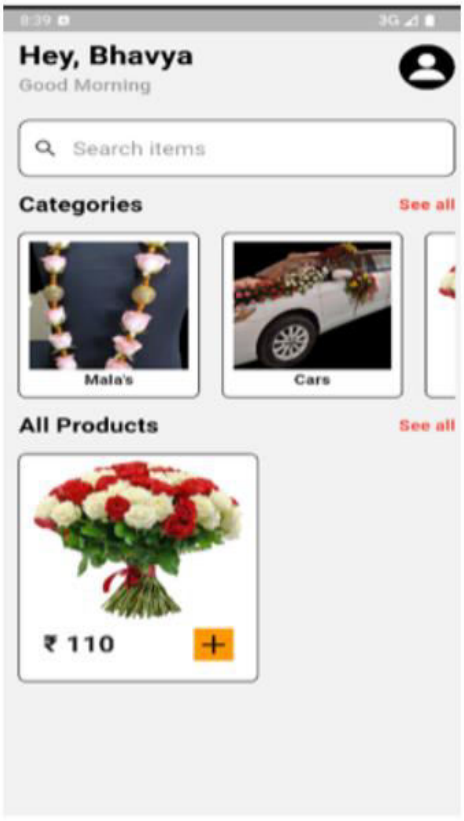


Figure 5: Pic. 1

Figure 6: Pic. 2

International Journal of Innovative Research of Science, Engineering and Technology (IJIRSET)

Impact Factor 8.699

|www.ijirset.com |A Monthly, Peer Reviewed & Refereed Journal| e-ISSN: 2319-8753| p-ISSN: 2347-6710|

Volume 14, Issue 3, March 2025

|DOI: 10.15680/IJIRSET.2025.1403161|

Figure 7: Pic. 3

VI. CONCLUSION

The creation of the ecommerce mobile app with Flutter illustrates the effectiveness of cross-platform technologies in developing friendly, high-performance applications. With integration of Flutter and Firebase, the application offers real-time data syn- chronization, secure authentication, and smooth payment processing.
The admin dashboard gives business owners the power to control products and mon- itor orders, closing the gap between local businesses and online consumers. The project showcases the strength of Flutter in ecommerce app development and pro- vides a foundation for future upgrades, including AI-powered product suggestions and personalized alerts.

REFERENCES

[1] J. Wu and J. Smith, “React native vs. flutter: A performance comparison between cross-platform mobile application frameworks,” International Journal of Mobile App Development, vol. 10, no. 3, pp. 123–134, 2018.
[2] A. Tashildar, N. Shah, R. Gala, T. Giri, and P. Chavhan, “Application de- velopment using flutter,” International Research Journal of Modernization in Engineering, Technology and Science, vol. 2, no. 8, pp. 1262–1266, 2020.
[3] S. Boukhary and E. Colmenares, “A clean approach to flutter development through the flutter clean architecture package,” in 2019 International Conference on Computational Science and Computational Intelligence (CSCI), pp. 1115–1120, IEEE, 2019.

INTERNATIONAL STANDARD SERIAL NUMBER INDIA

INNO SPACE SJIF Scientific Journal Impact Factor

doi cross ref

निस्केयर NISCAIR

INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH

IN SCIENCE | ENGINEERING | TECHNOLOGY

9940 572 462 6381 907 438 ijirset@gmail.com

www.ijirset.com

Scan to save the contact details