e-ISSN: 2614-8404
p-ISSN:2776-3234

Development of a React Native-Based Mobile E-Commerce Application to Optimize Online Sales for MSMEs: A Case Study of New Delisio Bakery Cake

William Kessler Suryanto[1], Sulistyo Dwi Sancoko[2]
[1]Program Studi Informatika, Fakultas Sains dan Teknologi, Universitas Teknologi Yogyakarta
[2]Program Studi Sains Data, Fakultas Sains dan Teknologi, Universitas Teknologi Yogyakarta
Email: [1]williamksuryanto@gmail.com, [2]sulistyo.dwisancoko@staff.uty.ac.id

Abstract – The development of a mobile-based e-commerce application aims to assist MSMEs, particularly New Delisio Bakery Cake, in optimizing their online sales and expanding market reach. Currently, New Delisio operates using conventional methods that limit its customer base and overall sales potential. To address this challenge, the application was developed following the waterfall methodology, encompassing requirements analysis, design, implementation, testing, and deployment. The front-end was built using React Native, while the back-end utilized Express.js and MongoDB. A user testing phase was conducted with a sample group, providing valuable feedback on usability and functionality. The application offers a user-friendly platform for customers to browse and purchase products, while enabling the business to efficiently manage operations. The test results showed a high overall success rate of 92.52%, with 91.11% success for user interactions and 100% for admin tasks. This demonstrates the application's effectiveness in enhancing usability and functionality. By leveraging modern mobile technology, the solution enhances the store's visibility, engagement, and scalability, potentially increasing revenue. This research demonstrates the potential for MSMEs to transition into the digital market through a structured development process and iterative improvements.

Keywords – *Mobile E-Commerce, Online Sales Optimization, React Native, Waterfall Method, Market Expansion*

I. INTRODUCTION

The rapid advancement of information technology has integrated the internet into daily human life, transforming it into a dynamic medium for efficient information access and online transactions[1], [2]. Users can conveniently browse, request, and pay for products from the comfort of their homes, significantly saving time and effort. This shift has encouraged many small businesses to adopt online sales systems to increase revenue [3]. However, many micro, small, and medium enterprises (MSMEs), such as New Delisio Bakery in East Bekasi, continue to rely on conventional sales methods, which results in inefficiencies and long wait times for customers who must visit the store to place orders.

Previous studies have explored mobile applications as a solution for improving business operations. While some solutions have demonstrated success, they often lack scalability and user-friendliness, making them less effective for MSMEs [4]. For instance, many existing systems fail to cater to the specific operational challenges faced by small businesses, including inventory management and customer engagement [5]. Addressing these gaps, this research proposes the development of a mobile-based e-commerce application using React Native and Express.js with MongoDB, aimed specifically at enhancing New Delisio's operational efficiency and expanding its market reach.

The objective of this research is to create a user-friendly, scalable mobile e-commerce application that simplifies order management for New Delisio Bakery, enhances customer experience, and increases market visibility. By addressing the limitations of conventional methods and previous solutions, this study aims to provide a digital transformation model tailored to MSMEs, ultimately optimizing online sales and increasing revenue [6].

1.1 Literature Review

1. Bakery Shop

A bakery shop is a retail outlet focused on the creation and sale of a range of baked items, including bread, pastries, cakes, and cookies [7].

2. Online Sales System

The online sales system (e-commerce) designed not only aims to simplify the purchasing process for consumers, but also to provide more complete product information and increase customer satisfaction. In developing this system, needs analysis, system design, implementation, and testing are carried out to ensure that the website built can meet user needs and function optimally [8].

3. Mobile Application

A mobile application (app) is a software program designed to run on mobile devices, such as smartphones and tablets, providing users with specific functionalities or services, often enhancing user experience and accessibility[9].

4. User Interface (UI/UX)

User Interface (UI) is a communication mechanism between the user (user) and the system in a program, be it a website, mobile, or software application. User Experience (UX) refers to the overall interaction and satisfaction a user has with a program, focusing on enhancing the user's experience, unlike User Interface (UI), which centers on the design elements [10].

5. React Native

React Native is a JavaScript framework for writing real, natively rendering mobile applications for iOS and Android [11].

6. ExpressJS

Express is a Web Framework built upon Node.js. Express builds on top of its features to provide easy to use functionality that satisfies the needs of the Web Server use-case [12].

e-ISSN: 2614-8404
p-ISSN:2776-3234

7. MongoDB

MongoDB is a powerful, flexible, and scalable general-purpose database. It combines the ability to scale out with features such as secondary indexes, range queries, sorting, aggregations, and geospatial indexes [13].

8. Cloudinary

Cloudinary provides a secure and comprehensive API (Application Programing Interface) for users to easily upload images from the server side directly in the browser or from mobile applications [14].

9. Stripe

Stripe is a payment platform, app and services platform based on APIs that can be implemented directly in code or no-code [15].

10. Unified Modeling Language

The Unified Modeling Language (UML) is the standard modeling language for software and systems development [16].

11. Use Case Diagram

A use case is all the ways of using a system to achieve a particular goal for a particular user. Taken together the set of all the use cases gives you all the useful ways to use the system and illustrates the value that it will provide [17].

12. Entity Relationship Diagram

The ER diagram is a semantic data modeling tool that is used to accomplish the goal of abstractly describing or portraying data [18].

13. Waterfall Method

The Waterfall Model is a linear, sequential software development process where each phase must be completed before the next begins, with no overlap. It ensures project success by following a step-by-step, phase-based approach [19].

II. RESEARCH METHODOLOGY

2.1 Research Framework

The following is a phased research framework.

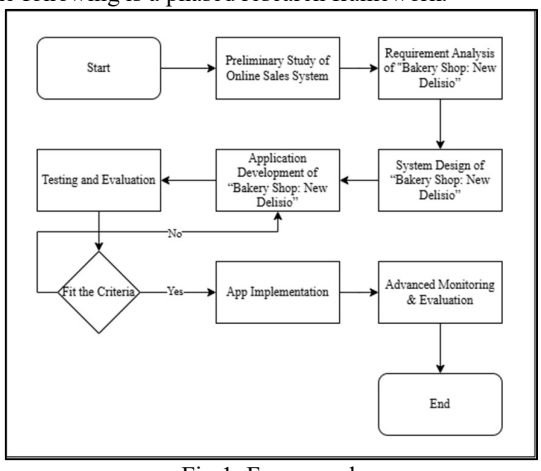

Fig 1. Framework

This research follows a phased framework illustrated in Fig 1, comprising several critical stages before deploying the application. These include a preliminary study of existing online sales systems, a needs analysis for New Delisio Bakery, system design, application development using React Native, testing and evaluation, and implementation. Post-launch, the application undergoes continuous monitoring and evaluation based on user feedback to ensure it meets the bakery's evolving needs.

2.2 Development Model

This research uses the Waterfall model. This model was chosen because it has the advantage of structuring the software development process in a linear and structured manner, allowing detailed identification of old system problems. Each stage in the Waterfall model, such as requirements analysis, system design, implementation, deployment, and maintenance, is carried out sequentially. This makes it easier to determine and identify the needs of the new system to be built more precisely and systematically. In addition, this approach minimizes the risk of errors by ensuring that each phase must be completed before moving on to the next phase, which is especially important in the development of complex e-commerce applications.

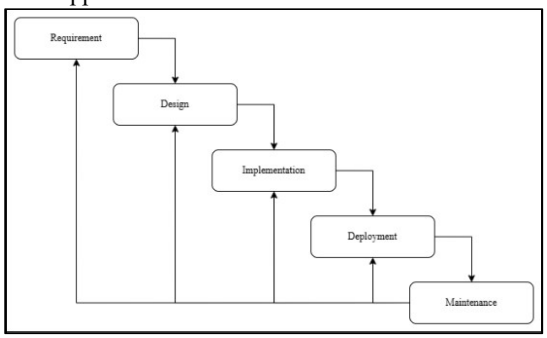

Fig 2. Waterfall Model

Each stage is described below:

2.2.1 Requirement

The system requirement analysis stage will carry out the following activities:

2.2.1.1 Analysis User Needs

The activities conducted in this research began with gathering initial information regarding the existing system to identify the problems encountered. The process of collecting primary data was carried out through surveys and direct field observations, as well as interviews with respondents, specifically the owners and managers of New Delisio Bakery. Additionally, a literature review was performed to obtain relevant information. To explore specific data, particularly concerning the challenges faced, focus group discussions were employed involving customers and bakery staff. The data collected was then analyzed to identify the needs and potential solutions for the development of this mobile e-commerce application.

2.2.1.2 Identification System

The New Delisio Bakery mobile e-commerce application is designed to enhance the efficiency of order management and improve customer engagement. The system comprises three main components: the front end, back end, and database. The front end, developed using React Native, provides an interactive user interface for customers, allowing them to browse products, place orders, and manage their accounts, as well as an admin panel for business owners. The back end, built on Express.js, handles API requests, manages user authentication, and processes data operations, ensuring secure and efficient communication between the front end and the database. For

e-ISSN: 2614-8404
p-ISSN:2776-3234

data management, the application utilizes MongoDB, which stores essential data such as product information, user profiles, and transaction records, providing a scalable and flexible solution [13], [20]. Additionally, the system integrates external services like Cloudinary for image management and Stripe for secure payment processing, ensuring a smooth and user-friendly experience for all users.

2.2.1.3 Identification System Needs

The New Delisio Bakery mobile e-commerce application requires specific needs to effectively serve both customers and business owners. First, it needs to support user registration and authentication, allowing customers to create accounts and securely log in. The application must also provide a robust product browsing feature, enabling users to search, filter, and view detailed product information, including images and descriptions.

Additionally, the system requires a shopping cart functionality that allows customers to add, edit, and remove items while displaying the total cost. A seamless checkout process is essential, integrating secure payment processing through Stripe and providing various payment options. The application should also include order tracking capabilities so customers can monitor the status of their purchases.

2.2.2 Design System

The system design stage aims to model the mobile e-commerce application that will be implemented for New Delisio Bakery. This stage involves two key activities: first, creating a general design of the system using UML diagrams to illustrate the various components and their interactions. This includes class diagrams that describe the processes occurring within each class, along with their attributes and relationships with other classes, specifically tailored to meet the functional and non-functional requirements identified earlier.

Second, the newly created class diagrams are analyzed and refined to define attributes that represent the identity of each entity, leading to the development of a Data Dictionary. This dictionary provides detailed descriptions of each entity's unique attributes, ensuring clarity in data management. Following this, an Entity-Relationship Diagram (E-R Diagram) is constructed to visualize the relationships between different entities in the system, such as products, users, and transactions. The E-R Diagram serves as a foundation for designing the database schema, which is then translated into structured tables within the MongoDB database, ensuring that all necessary data structures are in place for the effective functioning of the application.

2.2.3 Implementation

The implementation phase aims to build the mobile e-commerce application and its associated database according to the results of the system design. This stage involves two crucial steps following the design phase:

2.2.3.1 Developing the Application

This includes writing the code for the user interface (UI) using React Native, ensuring it provides an intuitive and engaging experience for customers and an effective admin panel for business owners. Additionally, the backend is developed using Express.js, creating the necessary API endpoints to facilitate communication between the mobile application and the MongoDB database.

2.2.3.2 Testing the Application

After the development of both the user interface and the backend, comprehensive testing is conducted. This includes functional testing of the user interface to ensure that all features work as intended, as well as testing the backend for API functionality, database integration, and payment processing with Stripe. This phase is essential to identify and rectify any issues before the application goes live, ensuring a smooth user experience and operational efficiency.

2.2.4 Deployment

Comprehensive testing was conducted to ensure functionality and accuracy using real operational data before the official launch.

2.2.5 Maintenance

Ongoing maintenance and updates were established based on user feedback to guarantee optimal performance and user satisfaction.

III. Results And Discussion

3.1 Analyze Current System Problems

The technology applied in the New Delisio Bakery mobile e-commerce application includes Stripe API for payment processing and Cloudinary for image storage. By utilizing the Stripe payment gateway, users can securely complete their transactions online, ensuring a smooth and efficient checkout experience. The integration of Stripe allows for various payment options, enhancing customer convenience and fostering trust in the payment process.

In addition to payment processing, the application leverages Cloudinary for efficient image management. This technology enables the bakery to store, optimize, and serve product images seamlessly. With Cloudinary, the application can deliver high-quality images while ensuring fast loading times, which is crucial for maintaining user engagement.

For customers browsing through the mobile application, the combination of Stripe and Cloudinary creates a cohesive and user-friendly experience. Customers can view appealing images of bakery products and complete their purchases with confidence, knowing that their payment information is securely handled. This integration not only simplifies the purchasing process but also enhances the overall functionality of the application, making it an essential aspect of the digital transformation for New Delisio Bakery.

3.2 Non-functional Requirement Analysis

The application must load quickly, targeting a response time of less than two seconds for all interactions to provide a smooth user experience. Additionally, the system should accommodate an increasing number of users and transactions without performance degradation,

e-ISSN: 2614-8404
p-ISSN:2776-3234

particularly during peak times. Robust security measures are crucial, including encryption for data transmission and compliance with PCI DSS standards for payment processing via the Stripe API. The user interface must be intuitive and accessible to a diverse demographic, ensuring clear navigation and responsive design. Furthermore, the system should maintain a target uptime of 99.9% to ensure high availability and minimal interruptions for users.

3.3 Functional Requirements Analysis

Functional requirements define the specific behaviors and functionalities that the New Delisio Bakery mobile e-commerce application must support.

1. The application must allow users to create accounts and securely log in using email and password. Users should have the option to reset their passwords and verify their accounts through email confirmation.
2. The system must enable administrators to add, update, and delete product listings, including descriptions, prices, and images. Products should be categorized for easy navigation.
3. Users should be able to add products to a shopping cart, view their selected items, modify quantities, and remove items before proceeding to checkout.
4. The application must facilitate the checkout process, allowing users to review their orders, and select payment methods. The system should handle order confirmations and send notifications to users upon successful purchases.
5. The application must integrate with the Stripe payment gateway to process secure online payments, including support for various payment methods (credit/debit cards, digital wallets, etc.).
6. Users should have access to their order history, including details of past purchases and current order statuses. This functionality should allow users to track their orders until delivery.
7. The application must provide a search feature that allows users to find products quickly. Filtering options should enable users to sort products by categories, price range, and popularity.
8. The application must include an admin panel for managing products, change user order status, and monitoring sales performance

3.4 Application Architecture Analysis

The physical architecture of the system is based on a two-tier application architecture model. This architecture comprises four primary components: the Client, the Application Server, the Database Server, and the Payment Gateway Server. The structure of the New Delisio application is illustrated in Fig 3 below.

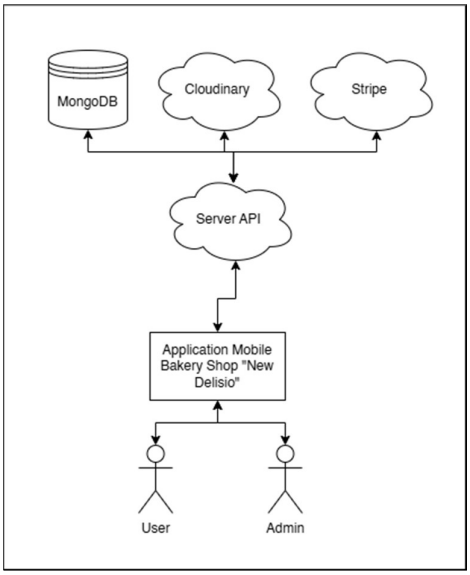

Fig 3. System Architecture

3.5 Software Features

This application includes features tailored to the roles of both the admin and the user.

3.5.1 Admin Authority

The admin has the following capabilities:

1) Manage overall product inventory.
2) Manage product categories.
3) Update order statuses.

3.5.2 User Authority

Users are granted the following functionalities:

1) Register for a new account.
2) Sign in to an existing account.
3) Change or reset their password.
4) View order history.
5) Check order status.
6) Update their profile information.
7) Change their shipping address.
8) Browse available products.
9) Search for specific products.
10) Filter products by category.
11) Add products to the shopping cart.
12) Modify the quantity of items to purchase.
13) Complete online payments.
14) Perform sign out.

3.6 Application Model

3.6.1 Use Case Diagram

The use case diagram depicts the interactions between the actors and the system. The first actor is the user of the mobile frontend application, who interacts with the app. The second actor is the admin, who engages with the admin panel via mobile application.

e-ISSN: 2614-8404
p-ISSN:2776-3234

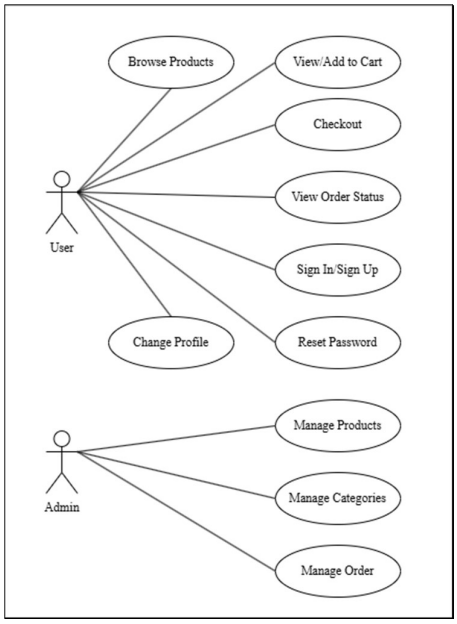

Fig 4. Use Case Diagram

3.6.2 Entity Relational Diagram

While this research uses a non-relational database (MongoDB), an ER-Diagram still provides valuable visualization of entities and their relationships within the system [18]. It outlines key collections, such as Users, Products, Transaction, Admin, and their connections, offering essential guidance for structuring data and ensuring coherent application design. The structure of the New Delisio database is illustrated in Fig 5 below.

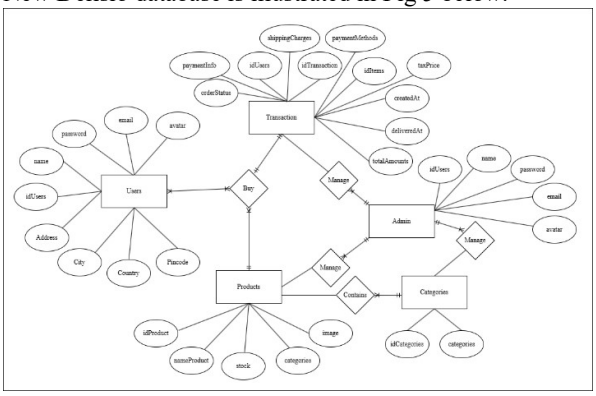

Fig 5. Entity Relational Diagram

3.6.3 Dictionary Data

This dictionary provides an outline of main attributes, data types, and descriptions for each field.

Table 1. Dictionary Data

Name Field	Type	Description
idUser	String (UUID)	Unique identifier for each user
name	String	Full name of the user
email	String	User's email address for login and contact
password	String (hashed)	Hashed password for user authentication
role	String	Role of the user
address	String	Primary address for the user
city	String	City of the user's address
country	String	Country of the user's address
pincode	Number	Postal code of the user's address
avatar	String (URL)	URL to the user's avatar image
idProduct	String (UUID)	Unique identifier for each product
nameProduct	String	Name of the product
stock	String	Available quantity of the product
imageProduct	String (URL)	URL to the product image
idCategories	String (UUID)	Unique identifier for each product category
category	String	Name of the product category
idTransaction	String (UUID)	Unique identifier for each transaction
shippingInfo	String	Shipping details and address
paymentMethod	String	Payment method used
paymentInfo	String	Additional payment information or reference
itemsPrice	Decimal	Total price of the items before tax and shipping
taxPrice	Decimal	Total tax on the items
shippingCharges	Decimal	Cost of shipping
totalAmount	Decimal	Final total amount (items + tax + shipping)
orderStatus	String	Status of the order (e.g., Pending, Completed)

3.6.4 Flowchart

The flowchart illustrates the sequential steps in the application's operation, mapping out user interactions, admin functions, and system processes. It provides a visual overview of the workflow, from user login and product selection to order processing and payment, ensuring clear and efficient functionality throughout the system.

e-ISSN: 2614-8404
p-ISSN:2776-3234

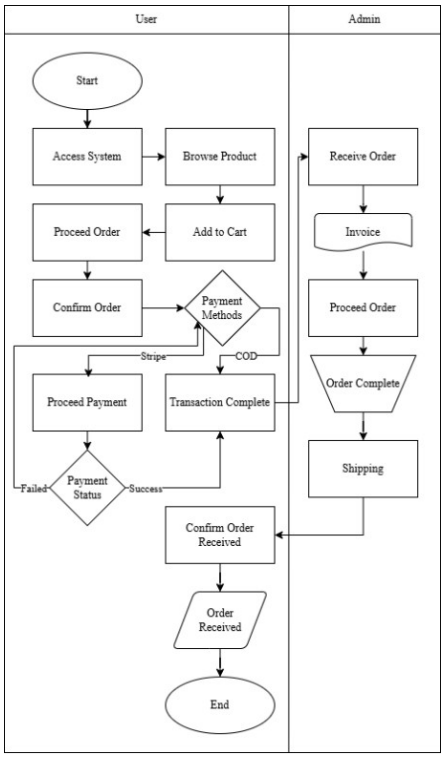

Fig 6. Flowchart

3.7 Design And Implementation User Interface

3.7.1 User Panel

3.7.1.1 Home Page

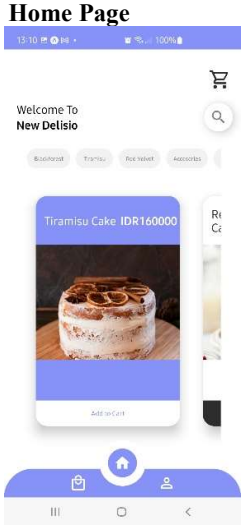

Fig 7. Home Page

The "New Delisio" app displays product categories, latest items, and a search feature for easy navigation and detailed product access.

3.7.1.2 Profile Page

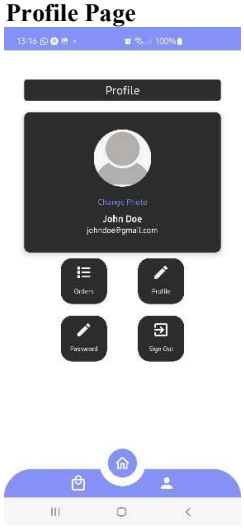

Fig 8. Profile Page

The profile screen allows users to manage their account, update their profile, view orders, change passwords, and log out, all in one place for easy access.

3.7.1.3 Search Page

Fig 9. Search Page

The "New Delisio" product list displays items with images, names, and prices. Users can search for specific products or tap an item to view detailed information.

e-ISSN: 2614-8404
p-ISSN:2776-3234

3.7.1.4 Product Detail Page

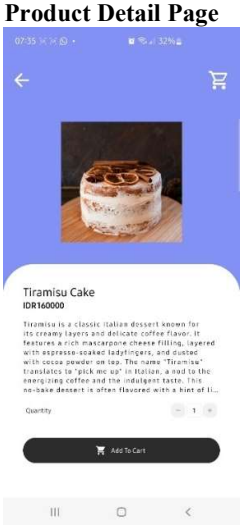

Fig 10. Product Detail Page

The product detail screen allows users to view specific product images, select cake sizes, add custom notes, and directly add items to the cart or proceed to the cart page without returning to the home screen.

3.7.1.5 Cart Page

Fig 11. Cart Page

The cart screen displays selected products, allowing users to view and adjust product quantities directly.

3.7.1.6 Confirm Order Page

Fig 12. Confirm Order Page

The order summary screen shows selected products, total cost, tax, shipping, and the final amount. After confirming, users proceed to choose a payment method.

3.7.1.7 Payment Method Page

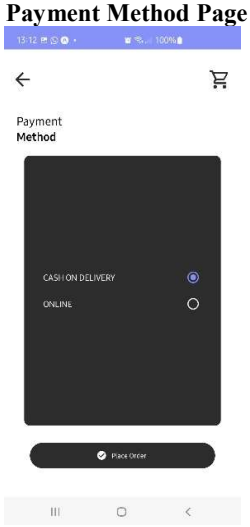

Fig 13. Payment Method Page

The payment method screen allows users to choose options like Cash on Delivery or Online by selecting a radio button. Users can finalize the order by tapping "Place Order."

e-ISSN: 2614-8404
p-ISSN:2776-3234

3.7.1.8 Stripe Payment Method Page

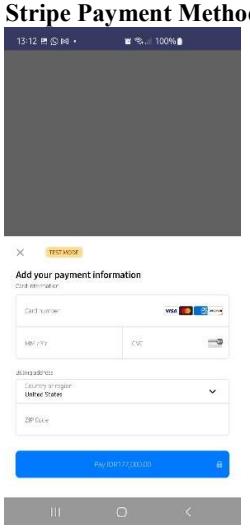

Fig 14. Stripe Payment Page

The payment information screen lets users enter credit or debit card details. After completing the form, users can proceed by tapping "Pay (Total Amount)."

3.7.1.9 Sign In Page

Fig 15. Sign In Page

The sign-in screen allows users to log in using their email and password.

3.7.1.10 Sign Up Page

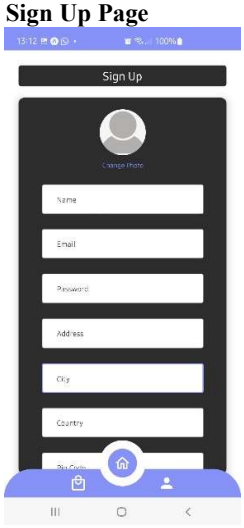

Fig 16. Sign Up Page

The sign-up screen lets users register with their name, email, username, and password. It includes authentication to alert users if the email is already registered or if the password is less than 6 characters.

3.7.1.11 Forgot Password Page

Fig 17. Forgot Password Page

The reset password screen lets users enter their email to receive a 6-digit OTP. After entering the OTP, users can set a new password.

e-ISSN: 2614-8404
p-ISSN:2776-3234

3.7.1.12 Order History Page

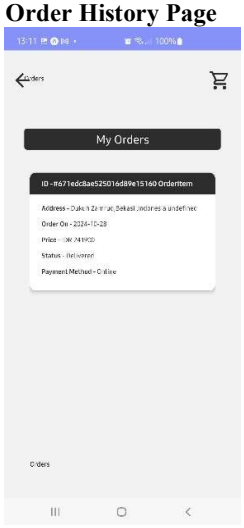

Fig 18. Order History Page

The transaction screen displays a list of user transactions, including details like invoice, date, and order status.

3.7.1.13 Edit Profile Page

Fig 19. Edit Profile Page

The edit profile screen allows users to update their name, email, address, city, country, and postal code. Authentication alerts users if the new email is already registered.

3.7.2 Admin Panel

3.7.2.1 Admin Dashboard

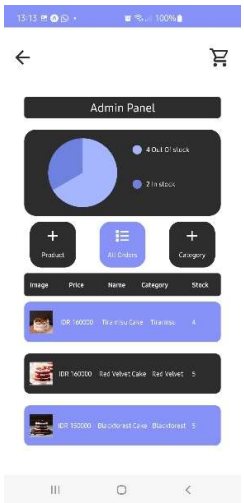

Fig 20. Admin Dashboard

The Admin Panel screen in the "New Delisio" app helps admins manage product data. It displays a pie chart showing product availability (in stock vs. out of stock). Key navigation buttons include Product (for adding new products), Category (for managing categories), and All Orders (to view all listed products). A table at the bottom provides product details like image, price, name, category, and stock quantity for easy management.

3.7.2.2 Add Product Page

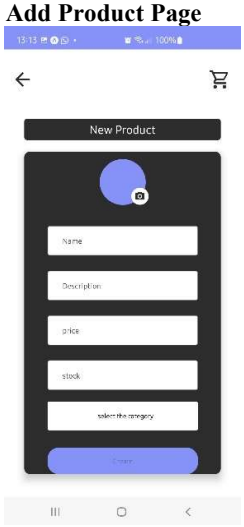

Fig 21. Add Product Page

The New Product page in the "New Delisio" app allows admins to add new products. Admins can enter product information such as name, description, price, stock quantity, and select a category. A camera icon at the top enables uploading product images. After filling out the details, the admin can add the product by pressing the "Create" button at the bottom.

e-ISSN: 2614-8404
p-ISSN:2776-3234

3.7.2.3 All Orders Page

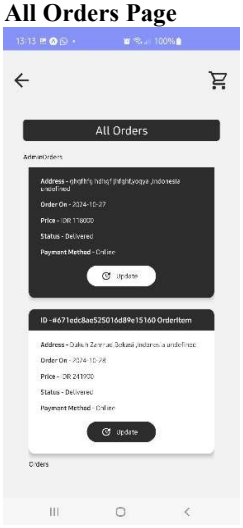

Fig 22. All Order Page

The All Orders page allows admins to manage customer orders. It displays order details such as delivery address, order number, date, total price, shipping status, and payment method in card format. Each card includes an "Update" button to modify order status or information.

3.7.2.4 Add Category Page

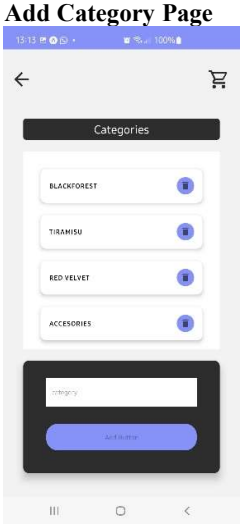

Fig 23. Add Category Page

The Categories page allows admins to manage product categories. It displays existing categories like Blackforest, Tiramisu, Red Velvet, and Accessories in card format, each with a trash icon to delete unnecessary categories. At the bottom, an input form lets admins add new categories by typing a name and pressing the "Add Category" button.

3.8 Black Box Testing

The *New Delisio* application was tested by 11 respondents, including 10 end-users and 1 admin. End-users focused on features like registration, ordering, and payments, while admins tested product, category, and order management. The testing followed a black-box approach to evaluate functionality and user experience. The table below summarizes the results, highlighting successes, issues, and improvements made.

Table 2. User Testing Result

Testing Scenario	Actual Outcome	Remarks/Issues Found	Success Rate (%)
Log in with valid credentials.	9 users succeeded, 1 experienced a delay.	Delay resolved by optimizing server response.	90%
Register with valid details.	All users succeeded.	No issues found.	100%
Enter valid email to receive OTP.	8 users succeeded, 2 experienced delays in email.	Improved email server configuration.	80%
Search for a product by entering a keyword in the search bar.	3 users succeeded, 3 reported no results.	Fixed bug in search filtering logic.	70%
View details of a product by clicking on it.	All users succeeded.	No issues found.	100%
Update product quantity in the cart.	9 users succeeded, 1 reported slow UI response.	Optimized cart UI responsiveness.	90%
Review and confirm the order.	All users succeeded.	No issues found.	100%
Select a payment method (e.g., Cash on Delivery).	All users succeeded.	No issues found.	100%
Enter credit card details for payment.	9 users succeeded, 1 reported incorrect error message.	Fixed error validation for card details.	90%
Admin adds a new product.	Success	No issues found.	100%
Admin updates the status of a pending order.	Success	No issues found.	100%
Admin deletes an existing category.	Success	No issues found.	100%

The user testing results indicate a high success rate of 91.11% for user interactions and a perfect success rate of 100% for admin tasks. The overall success rate stands at 92.52%. While most features performed well, minor issues such as delays in email verification and search functionality were identified. These have been addressed to enhance the user experience, ensuring smoother operations and greater reliability moving forward.

IV. CONCLUSION

In conclusion, this research successfully developed a mobile-based e-commerce application tailored for New Delisio Bakery, enabling MSMEs to transition from conventional to digital sales operations. By leveraging React Native, Express.js, and MongoDB, the application achieves both user-friendliness and scalability, effectively enhancing customer experience and operational efficiency for the bakery. This project advances the field by providing a digital transformation model adaptable for other MSMEs

e-ISSN: 2614-8404
p-ISSN:2776-3234

with similar needs, especially those aiming to expand their market reach through technology.

The user testing results show a high success rate of 91.11% for user interactions and a perfect success rate of 100% for admin tasks, with an overall success rate of 92.52%. Despite this, some areas require improvement, including the accuracy of the product search feature, which experienced issues for a few users, and the email verification process. These issues have been identified and will be addressed to further enhance the application's reliability and user experience.

This application's functionality—such as product management, secure payment integration, and real-time order tracking—addresses existing limitations in MSME sales systems, positioning it as a valuable tool for small businesses seeking growth in a competitive digital landscape. Future development could expand this application with advanced analytics for customer insights and inventory management features, adding further value for users. Additionally, ongoing research could explore AI-driven recommendation engines to personalize the shopping experience, offering a broader application of the system and opening new research directions.

REFERENCES

[1] S. Batubara, D. S. Harahap, N. H. Hrp, and F. Maharani. (2021). Analysis Of The Impact Of Using E-Commerce In Increasing Sales Turnover Of Micro, Small And Medium Enterprises In Padangsidimpuan City. International Journal of Educational Research & Social Sciences. 2 (5), 1008–1014. Available: https://ijersc.org/.

[2] M. Saputra, S. Nurhaliza Sofyan, A. Aulia, A. Ernawati, A. Oftasari, and R. Farta Wijaya. (2024). Implementation of E-Commerce System as SME Development Strategy in the Digital Era. Bulletin of Information Technology (BIT). 5 (3), 195–202. DOI: 10.47065/bit.v5i2.1545.

[3] W. Lestari and S. Sartika. (2021). Aplikasi Penjualan Berbasis Web Pada Toko Era Bangunan. *SIMKOM*. 6 (2), 104–112. DOI: 10.51717/simkom.v6i2.94.

[4] M. Kleppmann. (2017). *Designing Data-Intensive Applications*. First Edition. Sebastopol: O'Reilly Media, Inc..

[5] H. Longenecker, J. W. Petty, E. L. Palich, and F. Hoy. (2017). *Small Business Management*, 17th ed. Cengage Learning.

[6] M. Muttaqin, M. Iqbal, and F. Wadly. (2017). Implementation of an E-Commerce Based UMKM Marketplace Application to Increase People's Income in Kota Pari Village. *Prosiding Universitas Dharmawangsa*. 3, 508–516.

[7] W. Gisslen. (2012). *Professional Baking*. Sixth Edition. New Jersey: John Wiley & Sons.

[8] L. Hanum, A. Hadi Nasyuha, N. Irawati, Y. H. Siregar, and S. Abbas. (2024). Designing an Online Sales System Using a Website. *Journal of Information System and Technology Research,* 3 (3), 101–106. DOI: https://doi.org/10.55537/jistr.v3i3.897.

[9] J. Cuello and J. Vittone. (2013). *Designing Mobile Apps*.

[10] H. Himawan and M. Yanu F. (2020). Interface User Experience. *Yogyakarta: Lembaga Penelitian dan Pengabdian kepada Masyarakat*.

[11] B. Eisenman. (2015). *Learning React Native*. Sebastopol: O'Reilly Media, Inc..

[12] F. Copes, *Express Handbook*.

[13] K. Chodorow. (2013). *MongoDB: The Definitive Guide*. Second Edition. Sebastopol: O'Reilly Media, Inc..

[14] A. Ayuningtias and A. Restu Pramudi. (2017). Undangan Paperless Berbasis Cloud Computing Dengan Memanfaatkan Cloudinary. *Compiler*, 6, (1), 1–7. DOI: 10.28989/compiler.v6i1.195.

[15] Ni'matul Ma and W. Wijaya. (2024). Pengembangan System Berlangganan Aplikasi di Usertip Menggunakan Stripe. *Journal of Information System Management (JOISM)*, 5 (2), 2715–3088.

[16] Russ. Miles and Kim. Hamilton. (2006). *Learning UML 2.0*. O'Reilly.

[17] I. Jacobson, I. Spence, and K. Bitnner. (2011). *Use-Case 2.0 The Definitive Guide*.

[18] Sikha. Bagui and Richard. Earp. (2004). *Database Design Using Entity-Relationship Diagrams*. CRC Press.

[19] U. S. Senarath. (2021). Waterfall Methodology, Prototyping and Agile Development. DOI: 10.13140/RG.2.2.17918.72001.

[20] S. Yange, I. Gambo, R. Ikono, O. Onyekwere, and H. Soriyan. (2020). An Implementation Of A Repository For Healthcare Insurance Using MongoDB. *Journal of Computer Science and Its Application*, 27 (1). DOI: 10.4314/jcsia.v27i1.3.