

Information System for Providing Food Services Based on Mobile Application Using Flutter Framework

Asroni^{1*}, Slamet Riyadi¹, Taufiq Cahyono¹

¹*Universitas Muhammadiyah Yogyakarta, Indonesia*

^{*}*Corresponding author. Email: asroni@umy.ac.id*

ABSTRACT

Foodservice is a food management that is handled by individuals and companies that provide food somewhere to meet the various needs of its providers based on orders. The progress of information technology is growing rapidly. One of them is a smartphone or better known as a smartphone, which uses the operating system like a computer. Al Iman Boga has a website-based foodservice booking information system that is required to open a browser and is less efficient. For the public to use food services, this research requires the development of a mobile-based food service information system. In the development of mobile applications is widely facilitated with frameworks, one of which is flutter. Flutter has advantages over other frameworks, one of which is super smooth animation. If there is no excessive element, then mobile applications developed using flutter can constantly run 60-120fps for using its own rendering system. In flutter, it can download android APK and iOS IPA files. This application is built with the Waterfall method with several stages: analysis, design, Coding, and Testing. The results of this study show that the application is following the purpose and design. With this application, it can help the provision of food service in building mobile applications. The system helps in the transaction process of food catering sales in Al Iman Boga.

Keywords: *Boga Services, Information Systems, Flutter, Waterfall*

1. INTRODUCTION

Foodservice is a food management that is handled by individuals and companies that provide food somewhere to meet the various needs of its providers based on orders. Foodservice products or catering food is a benchmark of consumer satisfaction that is adjusted to the habits and experiences of those consumers who enjoy the product [1].

Advances in information technology are increasingly rapid, one of which is smartphones or better known as smartphones that have used the operating system like a computer. Smartphones can be installed in a wide range of applications easily and quickly. Android is one of the Linux based operating systems used on mobile devices. Android systems and applications are constantly evolving, making developers to continue to advance the technology in the world[2].

Al-Iman Boga has a website-based foodservice booking information system. This application still has its drawbacks, namely the display that is still simple and less dynamic. Most people nowadays are used to using smartphones. Ordering food in Boga Al-Iman itself still requires opening a browser on a smartphone or computer first. Therefore, in this research is the development of a mobile-based food service information system[3].

A mobile application is a system that can run on both android and android iOS devices[4]. Mobile applications

are widely developed using The Java, React, and Dart languages for now. It has now used a framework in a language to make it easier for develop pers to develop a system. Frameworks for mobile application development are Java, React Native, and Flutter[5]. Each framework has its disadvantages and advantages[6]. But flutter has advantages over other frameworks, one of which is super smooth animation. If there is no excessive element, then mobile applications developed using framework flutter can constantly run 60-120fps because it uses its own rendering system. In flutter, it can download android APK and iOS IPA files[7] [8].

2. METHOD

This research applies the SDLC (*Software Development Life Cycle*) method for system development. This method is the cycle of system development[9] [10]. For SDLC model that was included in this study is the *waterfall model*[9]. This method is a system development cycle in software. There are several stages, as follows:

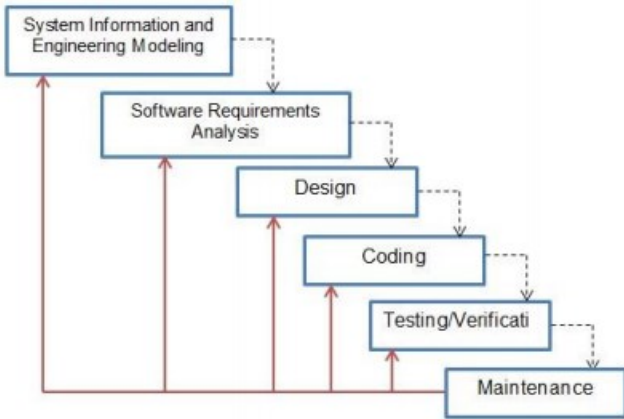


Figure 2.1 Waterfall Method

2.1. Planning Stage

The planning or modeling stage is a study of application module development needs. This stage is done planning about the application module to be created and determine the strategy for application development.

2.2. Needs Analysis

At this stage, researchers collect data related to the need to build applications by observing existing research objects. The collected data is used to analyze the system's needs and functions in the application to be built.

2.3. System Design

This stage results from processed needs analysis than presented in a system design that will be used for the application development process's needs. In this stage also carried out planning related to programming languages and databases used.

2.4. Implementation

This stage is the result of the planning stage will be implemented at this stage. Each function is built based on the needs of the system design. The system is built using programming language and *database as* planned at an earlier stage.

2.5. Testing

In this stage, the results of system development are tested based on the design and needs that existed in the previous stages to ensure that the system is under the needs and there are no *errors*.

2.6. Maintenance Stage

This stage is necessary for the maintenance of the system, because for maintenance in fixing errors or weaknesses of the system that are not detected during testing and improve system performance so that all functions run properly.

3. RESULTS AND DISCUSSIONS

3.1. DATABASE CREATION

In this step, it is ensured that Apache and MySQL are connected[11]. To create a database open a browser with an address <http://localhost:8282/phpmyadmin/index.php>. Databases that have been built are user, product, and basket, as for the database as in figure 3.1.

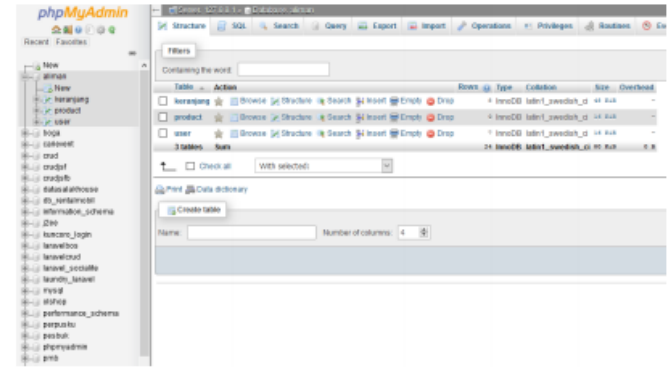


Figure 3.1 PhpMyAdmin Database

3.2. Login Page

The *login* page listed in figure 3.2 shows there are two options. The *user* can go directly to the menu if it has been registered, and the *user can* register if there is no account. After filling out the form, the *user* can enter the menu[12].

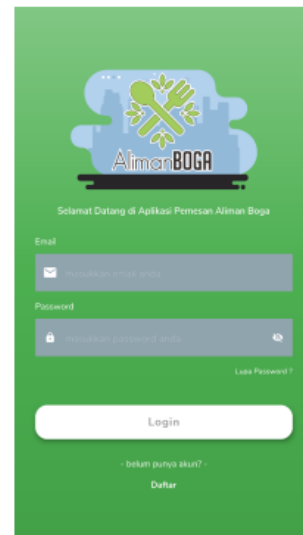


Figure 3.2 Login Page

3.3. Register Page

In the 3.3-page register image, the *user must* complete by filling in the data on the form. If it is not complete, then the user cannot register a new account[2].

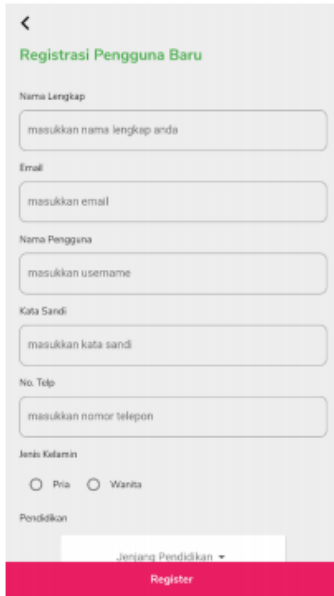


Figure 3.3 Register Page

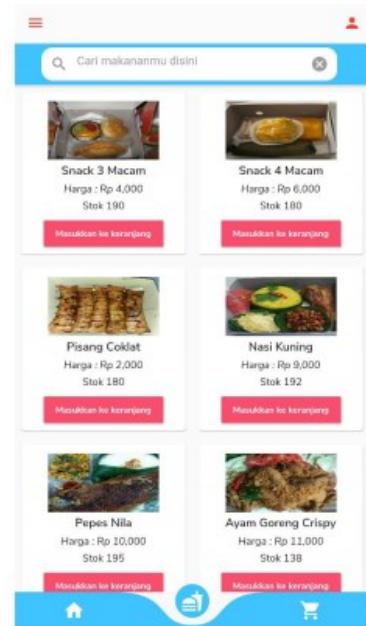


Figure 3.4 Catalog Page

3.4. Home Page

In figure 3.4 shows the home menu. On the home or main menu, users can see promotional slide banners, select buttons to go to catalog navigation, my cart and about, as well as best-selling products. To move to another page the user can press the bottom bar button[13].

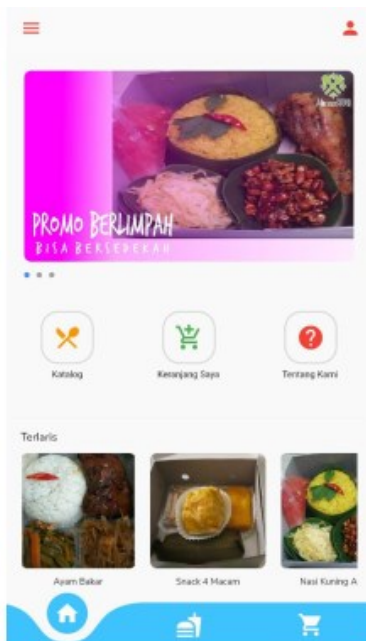


Figure 3.4 Home Page

3.5. Catalog page

In figure 3.4 shows the catalogue pages. Users can order a variety of food, more than 1 product. Users can also search for food by searching for the search field.

3.6. Order Details page

In figure 3.5 shows the order detail page. Users can order with the amount as desired and the date for the booking. If you have pressed the *order*, the data automatically enters the basket and sent via *user email*.

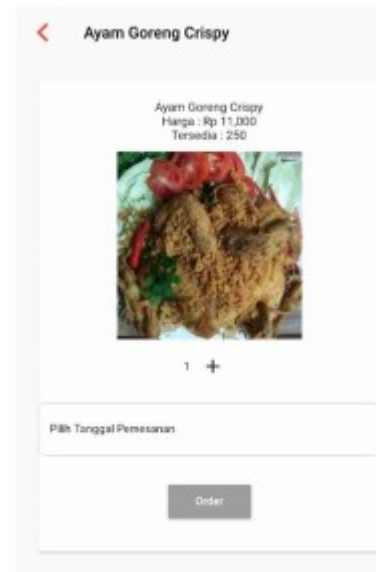


Figure 3.5 Order Detail Page

3.7. Successful Order Page

In figure 3.5 shows the successful order page. Users can see order cart details after pressing the “my order” check button.

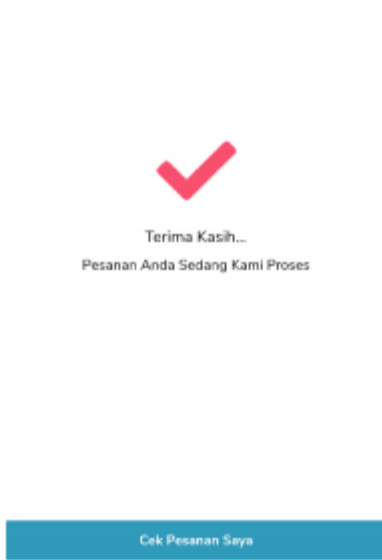


Figure 3.5 Success Order Page

3.8. Cart Page

In figure 3.6 shows the order basket page. The user can cancel the order if there is an error when ordering. Payment will be sent to email by the system. The booking status is also stated on the label. If you have not made a payment, then the status is still waiting. If you have already made a payment, then the status changed to have been paid. The admin will confirm the status itself.

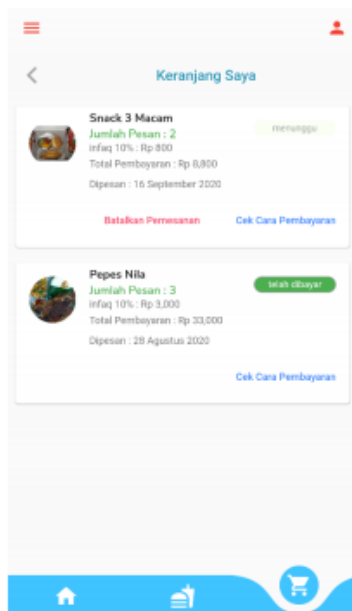


Figure 3.6 Cart Pages

3.9. Check Payment Page

In figure 3.7 shows the payment check page. To find out how to pay, users can view this page. On this page, there is payment details information.



Figure 3.7 Payment Check Page

3.10. About Page

In figure 3.8 shows the about page. Users can find out about app and version developer information.



Figure 3.8 Page About

3.11. Black Box Testing

This test is carried out to determine the results of the system's execution by observing the test data by examining the overall functional and evaluating the appearance of the application [14] [15]. The following tests are in table 3.1.

Table 3.1 Black Box Test Results

No	Test Case	Expected Results	Conclusion
1	Clear the <i>input field</i> in the login <i>form</i> and press the login button	The system will reject, and an error message <i>appears</i>	Successful
2	Just fill in the <i>email form</i> and empty the password <i>form</i> and press the login <i>button</i>	The system will reject and an error message <i>appears</i>	Successful
3	Just fill in the <i>password form</i> and empty the <i>email form</i> and press the login <i>button</i>	The system will reject and an error message <i>appears</i>	Successful
4	Fill in the login <i>form</i> with your registered <i>email password</i> and password	The system receives <i>login access</i> and enters the menu	Successful
5	Press list text	The system will go to the <i>register page</i>	Successful
6	Clear the <i>input field</i> on the register <i>form</i> and press the <i>register button</i>	The system will reject, and an error message <i>appears</i>	Successful
7	Just fill in the <i>full name form</i> and empty <i>form password</i> then press the register <i>button</i>	The system will reject, and an error message <i>appears</i>	Successful
8	Just fill in the <i>email form</i> and empty the other <i>form</i> and press the <i>register button</i>	The system will reject and an error message <i>appears</i>	Successful
9	Just fill in the <i>username form</i> and clear the other <i>form</i> and press the register <i>button</i>	The system will reject and an error message <i>appears</i>	Successful
10	Just fill in the <i>password form</i> and clear the other <i>form</i> and press the register <i>button</i>	The system will reject and an error message <i>appears</i>	Successful
11	Just fill out the <i>phone number</i>	The system will reject and an	Successful

No	Test Case	Expected Results	Conclusion
	<i>form</i> and empty the other <i>form</i> and press the <i>register button</i>	error message <i>appears</i>	
12	Fill out <i>all the forms</i> in the register <i>form</i> and press the register <i>button</i>	The system receives <i>login access</i> and goes to the login <i>page</i>	Successful
13	Fill out the <i>email form</i> with the one that has been registered then press the <i>register button</i>	The system will reject and appear message <i>box "email already in use"</i>	Successful
14	When pressing the <i>bottom bar home button</i>	<i>The system will move to the home page</i>	Successful
15	When pressing the <i>bottom bar order button</i>	The system will move to the <i>order page</i>	Successful
16	When pressing the <i>bottom bar button of a basket</i>	The system will move to the cart <i>page</i>	Successful
17	When pressing the <i>menu icon button</i> on the upper left side then pressing the <i>current message</i>	System switches to <i>order page</i>	Successful
18	When pressing the <i>menu icon button</i> on the upper right side then pressing <i>log out</i>	The system moves to the <i>login page</i>	Successful
19	Pressing the <i>enter basket button</i>	The system moves to the <i>order details page</i>	Successful
20	Fill out the <i>search form</i>	The system will display based on the product <i>name</i>	Successful
21	Press the (+) and (-) keys to <i>jumlah order</i>	The system will display the number of <i>products</i>	Successful
22	Pressing the <i>order button</i>	The system will display <i>page successfully</i> and the product <i>stock</i> will be reduced according to the	Successful

No	Test Case	Expected Results	Conclusion
		number of messages and send messages via <i>user email</i> . pengguna.	
23	Pressing undo button on <i>cart list</i>	The system will remove the basket and the stock of the product increases according to the number of pre-ordered orders	Successful

4. CONCLUSION

The results of the discussion and testing described and described by the author in "Information System *for Providing Mobile-Based Food Services Using Flutter Framework*," it can be concluded that:

1. Information System for Providing Food Services Based on Mobile Using Flutter Framework can help provide food service services in building mobile applications.

2. The system helps in the process of selling food in Al-Iman Boga.

REFERENCES

- [1] S. Kardigantara, "Diklat: Operasional Katering," Bdg. STPB, 2006.
- [2] A. D. Latture, "Backdrop: An Exploration of Flutter," p. 9.
- [3] A. E. Fentaw, "Cross platform mobile application development: a comparison study of React Native Vs Flutter," p. 98.
- [4] A. Praveen, K. Nanda, N. Rajith, N. Giriraj, R. Radhika, and N. Mahesh, "Conference Room Booking Application using Flutter," p. 3.
- [5] C. Rieger and T. A. Majchrzak, "Towards the definitive evaluation framework for cross-platform app development approaches," *J. Syst. Softw.*, vol. 153, pp. 175–199, 2019.
- [6] B. Milner and X. Shao, "Clean speech reconstruction from MFCC vectors and fundamental frequency using an integrated front-end," *Speech Commun.*, vol. 48, no. 6, pp. 697–715, Jun. 2006, doi: 10.1016/j.specom.2005.10.004.
- [7] G. Idan Arb and K. Al-Majdi, "A Freights Status Management System Based on Dart and Flutter Programming Language," *J. Phys. Conf. Ser.*, vol. 1530, p. 012020, May 2020, doi: 10.1088/1742-6596/1530/1/012020.
- [8] O. Dahl, "Exploring End User's Perception of Flutter Mobile Apps," p. 46.
- [9] A. B. Kamaludin and D. Dharmayanti, "APPLICATION MONITORING THE COURIER ON PESHOP UTILIZING WEBSOCKET AND FLUTTER."
- [10] Y. R. Handayanto and K. E. Dewi, "DEVELOPMENT OF BLOOD DONOR APPLICATION USING GEOFENCING AND FIREBASE TECHNOLOGY ON ANDROID PLATFORM."
- [11] A. M. Qadir and P. Cooper, "GPS-based Mobile Cross-platform Cargo Tracking System with Web-based Application," in 2020 8th International Symposium on Digital Forensics and Security (ISDFS), Beirut, Lebanon, Jun. 2020, pp. 1–7, doi: 10.1109/ISDFS49300.2020.9116336.
- [12] Y. Cheon and C. Chavez, "Creating Flutter Apps from Native Android Apps," p. 10.
- [13] S. Aminah, S. B. Bhaskoro, and A. S. Sunarya, "Desain dan Implementasi Aplikasi Inventaris Alat Praktikum Pada Laboratorium Berbasis Android dan QR Code," 2020, vol. 1, no. 1, pp. 91–95.
- [14] R. Groz, N. Bremond, A. Simao, and C. Oriat, "hW-inference: A heuristic approach to retrieve models through black box testing," *J. Syst. Softw.*, vol. 159, p. 110426, Jan. 2020, doi: 10.1016/j.jss.2019.110426.
- [15] T. Hidayat and M. Muttaqin, "Penguujian Sistem Informasi Pendaftaran dan Pembayaran Wisuda Online menggunakan Black Box Testing dengan Metode Equivalence Partitioning dan Boundary Value Analysis," vol. 6, p. 5, 2018.