Online version available at: https://jurnal.buddhidharma.ac.id/index.php/te

JOURNAL OF TECH-E

| 2581-1916 (Online) | 2598-7585 (Printed) |

Articles

# Design and Build Customer Complain Applications for Mobile Based MVVM Architecture Method

*Alexius Hendra Gunawan*[1], *Andri Wijaya*[2], *Dellih Wijaya*[3],

[1, 2 ,3]*Buddhi Dharma University, Information System, Banten, Indonesia*

SUBMISSION TRACK

Received : Dec 1, 2019
Revised : Dec 15, 2019
Accepted: Feb 10, 2020
Available online : Feb 26, 2020

KEY WORD

*MVVM Architecture, Cross Platform Mobile app, Rapid Application Development (RAD)*

CORRESPONDENCE

E-mail : alexius.gunawan@ubd.ac.id

ABSTRACT

PT. DhanaJaya BogaIndo has more than twenty restaurants spread across Jakarta, Surabaya and Australia. At present complaints regarding services and food served are still being done by providing limited criticism and suggestions during operational hours, so employees in the service section always record complaints and suggestions from customers who come to restaurants are accommodated, then convey related to food, service and the environment dirty one. related to customer complaints. Information on customer complaints can be received directly by management, so actions can be taken quickly.

## INTRODUCTION

PT. DhanaJaya BogaIndo has more than twenty restaurants spread across Jakarta, Surabaya and Australia. At present complaints regarding services and food served are still being done by providing limited criticism and suggestions during operational hours, so employees in the service section always record complaints and suggestions from customers who come to restaurants are accommodated, then convey related to food, service and the dirty environment.

Seeing this problem the author intends to create a Customer Complain Application Design Method Integration of MVVM Architecture based on Mobile Multi Platform using *C#, Xamarin Cross Platform* and *SQL Server* database. It is expected that by being able to help companies to get information from restaurant employees related to customer complaints. Information on customer complaints can be received directly by management, so actions can be taken quickly.

## I. Literatures Review

### a. Information Systems

The information system is a collection of components - components that collect, process, store and provide output of any information needed in business processes and applications used through software, databases and even manual processes that are related [1].

### b. Android

Android is an operating system for cellular phones based on Linux. Android provides an open platform for developers to create their own applications for use by a variety of mobile devices [2].

### c. Windows Phone

Windows phone is an operating system from Microsoft for smartphones, as a replacement for its predecessor operating system, namely Windows Mobile. This Windows phone is integrated with the operating system that is often used on a computer or notebook and Windows Phone changes its name to Windows 10 [3].

### d. IOS

IOS (formerly Iphone OS) is a mobile operating system developed and distributed by Apple Inc. This operating system was first launched in 2007 for the iPhone and iPad Touch [3].

### e. Xamarin Development

Xamarin development is an open source framework for making cross-platform mobile application software which is an implementation of the Common Language Infrastructure (CLI) and Microsoft's .Net framework, also called Mono [4].

## II. RESEARCH METHODS

### a. General Method

In designing this mobile-based customer complaint application using the appropriate methodology so that the customer complaint application can accommodate user needs for application development, while the general methodology used is as follows:

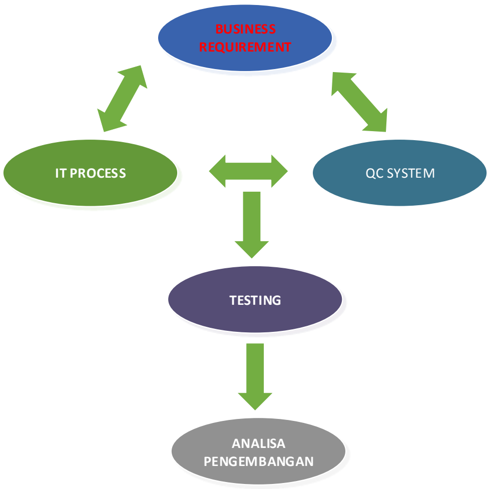


Figure 1. General Methodology

The results of Business Requirement are used for the analysis of the development incorporated in the evaluation that has been processed by IT and QC systems that are adjusted to the company's business demand. Module development testing and analysis is the final stage after the IT process and QC system are running well.
In the IT Process and QC module, this system plays a key role and determines the success rate of customer complaints application design and engineering activities of the MVVM Architecture method based on mobile multiplatform at DinTaiFung restaurant.

### b. Special Method

Specific methodology that we use in designing customer complain application Architeture and Rapid Application Development (RAD) based MVVM mobile multiplatform methods.

### 1. Definition

Rapid Application Development (RAD) or Rapid Prototype is a software development process model that belongs to *incremental* techniques. RAD emphasizes short, short and fast development cycles [5]. RAD is an "high-speed" adaptation of the waterfall model, where rapid development is achieved using a component-based construction approach [5].

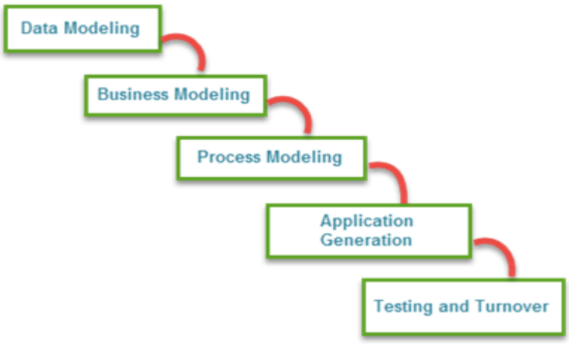


Figure 2. Rapid Application Development *(*RAD)

### 2. Phases and Stages of Application Development

According to Kendal [6], there are three phases in RAD that involve the analyst and the user in the assessment, control and implementation stages. The three phases include:

1. Requirements Planning
2. Workshop design RAD
3. Implementation

## III. MVVM ARCHITECTURE

MVVM is the most common architectural pattern used in windows applications. This pattern was formulated by Ken cooper and Ted PeSters [7]. The purpose of the MVVM architecture is almost the same as the MVC architecture, but the control is replaced by the view model. This view model presents from the view module in terms of actions and properties. Components contained in bind can adjust from view model and view model updates if there are changes every time.

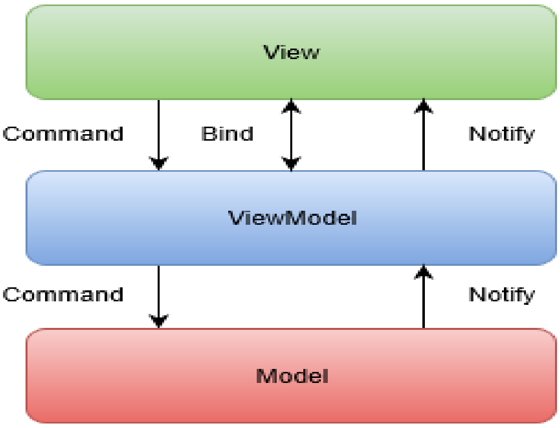


Figure 3. Pola MVVM Architecture [8]

### 3.1 VIEW

View is responsible for determining the structure, layout and appearance of what users see on the application screen. Every screen display in xamarin uses XAML with limited *code-behind* that does not contain code and algorithms.

In Xamarin Form, views with type of class derived (*Page-derived*) or content view. Names can be used templates that can beautify a screen display in Xamarin Form.

There are several options for running the algorithm code on the display model in response to interactions on the display such as clicking a button or selecting a menu item. If control passes a command in the *Command Control* property it can be retrieved permanently with data in the property.

### 3.2 VIEW MODEL

View Model implements properties and algorithms that can coexist in executing commands, and notifies information if there are changes. The algorithm properties and commands provided by the display model determine the functionality created by the user interface, but the display determines how the functionality will be displayed.

View Model is responsible for coordinating display interactions with whatever model classes are needed. Usually there is a one-to-many relationship between display models and model classes. The display model might choose to expose the model class directly to the view and the model class directly to the view so that the controls in the view can be explained directly by the view property, in this case the model class needs to be designed to be run along with the data.

Each display model provides data from a shape model that can be easily used. To achieve this, the display model sometimes changes data. Changing this data can provide properties that are simultaneously carried out together by the display model for example, the view model will combine the values of two properties to be displayed by the display.

## 3.3 MODEL

The model class is a non-visual class that encapsulates application data. Therefore, the model can be considered to represent the application domain model, which usually includes a data model along with business logic and validation. Examples of object models include *Data Transfer Objects (DTOs), Plain Old CLR Objects (POCOs)* and *generated entities* and *proxy objects*.

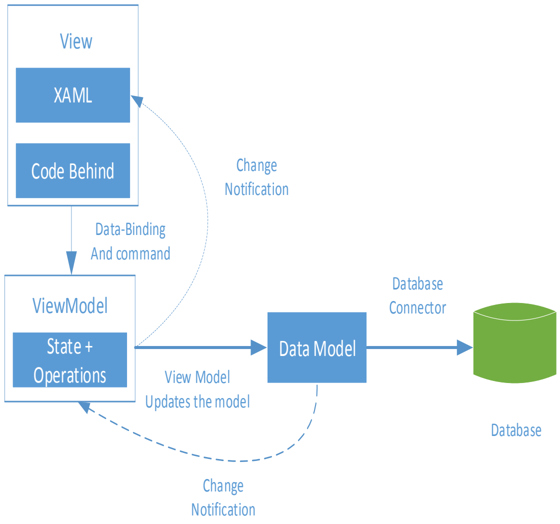


Figure 4. The Detailed Process of MVVM Architecture Patterns.

# IV. RESULTS

## 4.1 Requirement Analysis

### 4.1.1 Activity Diagram

The Activity diagram illustrates the various streams in the system to be designed, how each flow starts, the decisions that might occur and how they end. The following is a diagram of the activities in the customer's application system that is running, The Activity diagram shown below

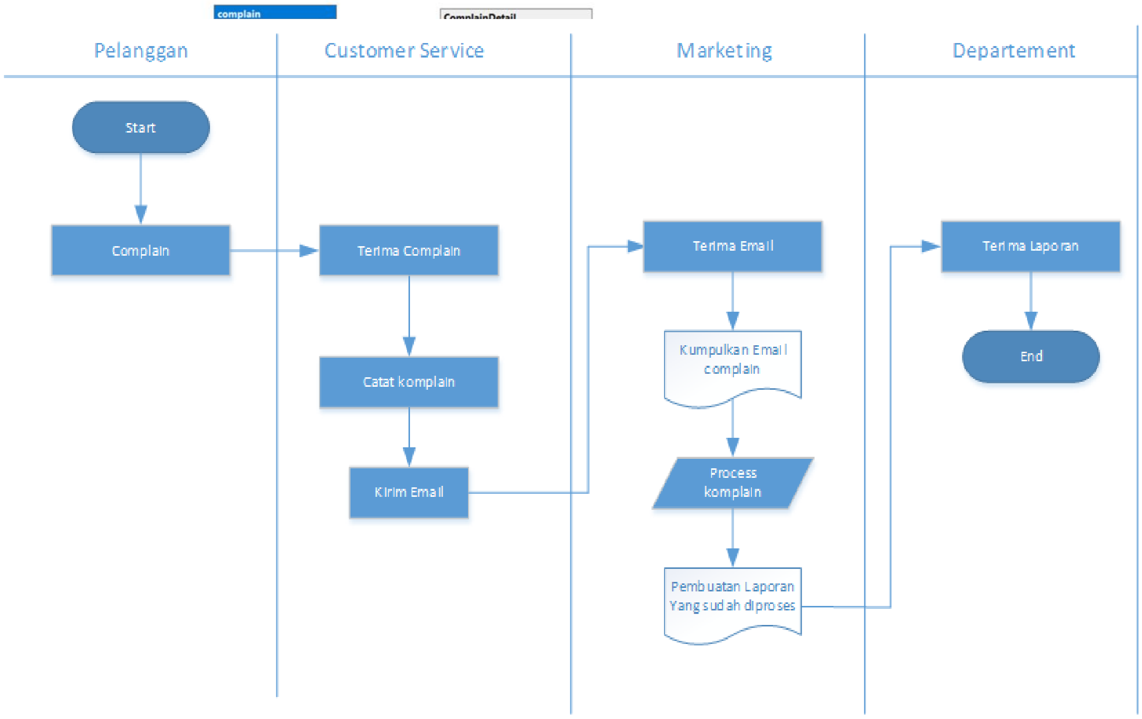
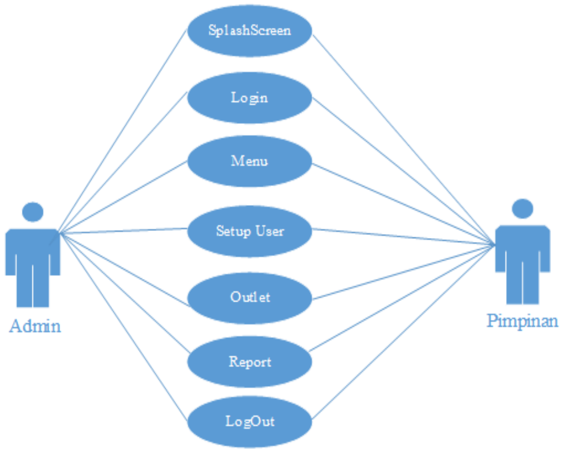


Figure 5. Activity Diagram of Customer Complain Application Process

### 4.1.2 Use Case

Use Case diagram explains an interaction between one or more actors with the system to be created. This diagram is used to find out what is in a system and who has the right to use these functions.



Gambar 6. *Use Case* Diagram Admin

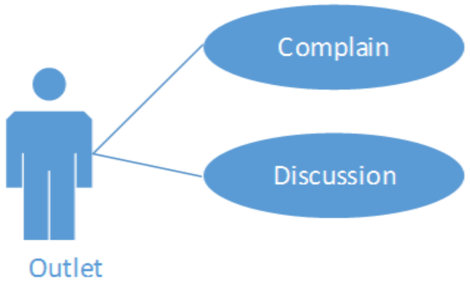


Figure 7. Use Case Outlet Diagram

### 4.1.3 Class Diagram

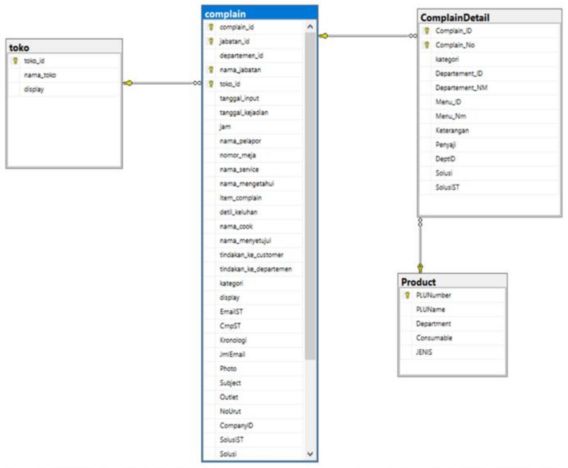


Figure 8. Customer Complain Diagram Class

### 4.1.4 Sequence

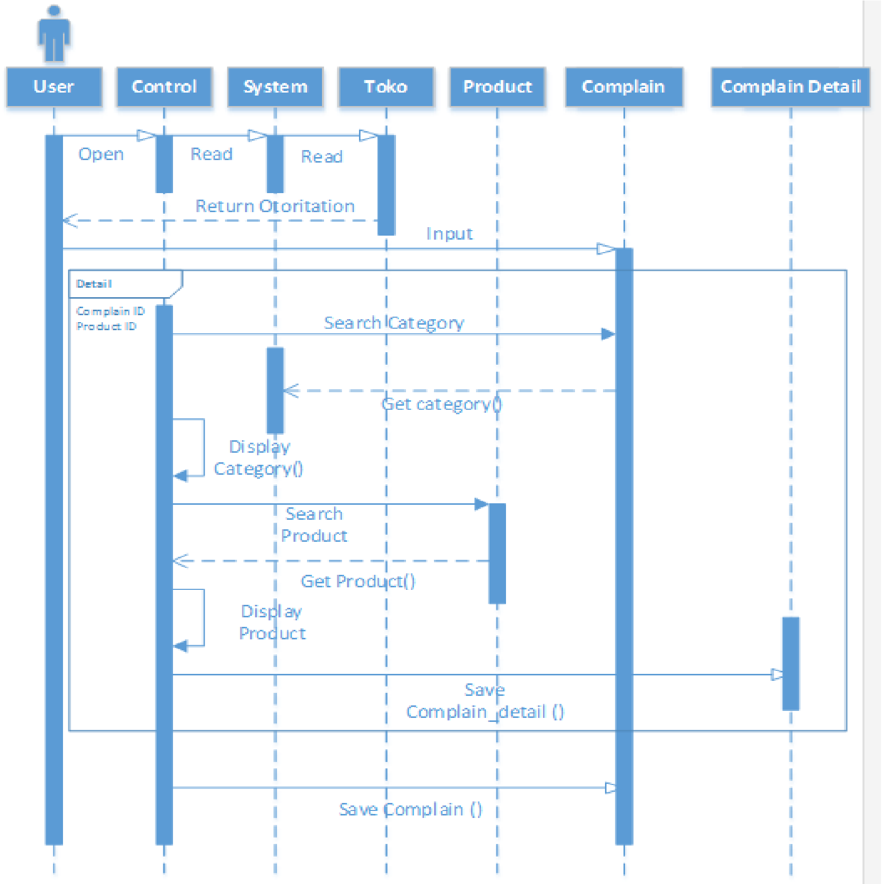


Figure 9. Customer complain Sequence Diagram

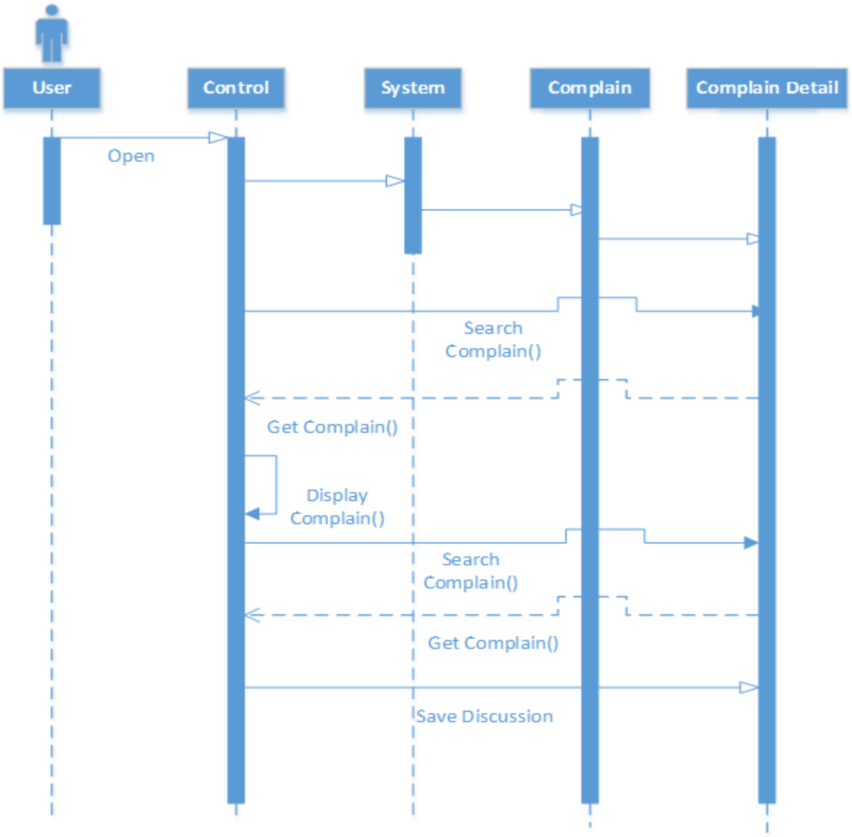


Figure 10. Sequence Diagram Discussion

## V. USER INTERFACE

### 5.1. Splash Screen

Figure 11. Splash Screen Display

### 5.2. Login

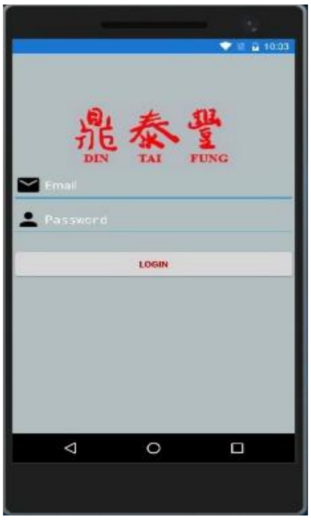


Figure 12. Display Login

### 5.3. Menu

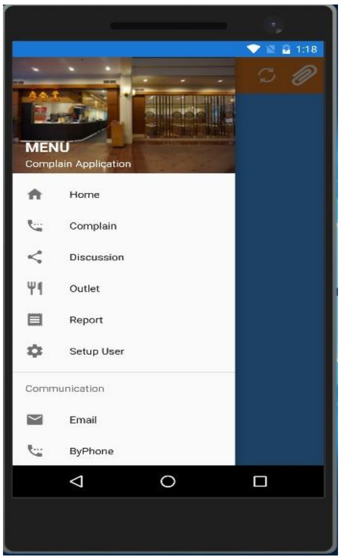


Figure 13. Display Menu

### 5.4. Outlet

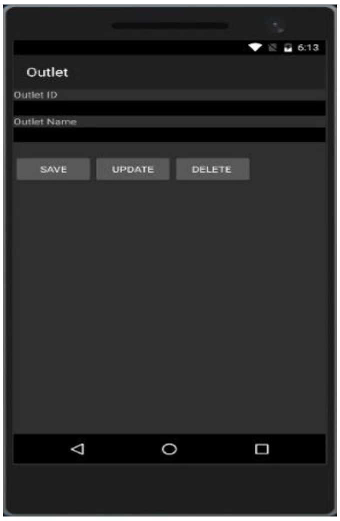
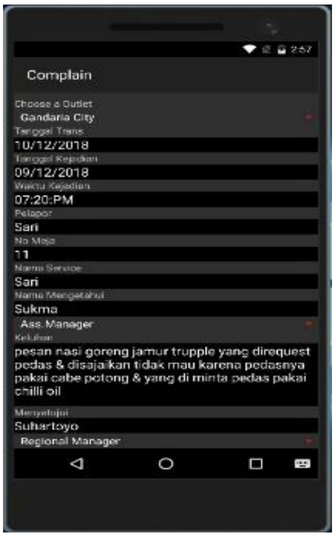


Figure 14. Display Outlet / Store

### 5.5. Complain Transaction



Complain

Choose a Outlet
Gandaria City

Tanggal Trans
10/12/2018

Tanggal Kejadian
09/12/2018

Waktu Kejadian
07:20:PM

Pelapor
Sari

No Meja
11

Nama Service
Sari

Nama Mengetahui
Sukma
Ass. Manager

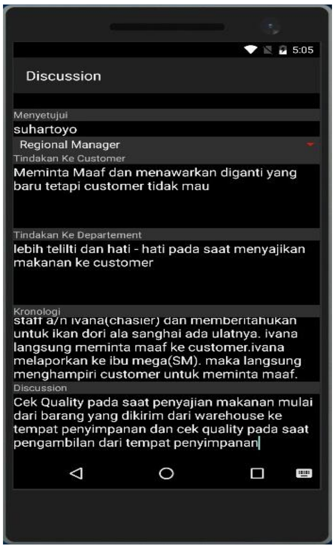
Keluhan
pesan nasi goreng jamur trupple yang direquest pedas & disajaikan tidak mau karena pedasnya pakai cabe potong & yang di minta pedas pakai chilli oil

Menyetujui
Suhartoyo
Regional Manager

Figure 15. Display Of ComplainTransactions

In this view, the Restaurant staff inputs customer complaints data into the database. This data that will be inputted must be known by superiors and approved by the Regional Manager at the restaurant. The data will be processed by the related management.

### 5.6. Discussion Transactions



5:05

Discussion

Menyetujui
suhartoyo
Regional Manager

Tindakan Ke Customer
Meminta Maaf dan menawarkan diganti yang baru tetapi customer tidak mau

Tindakan Ke Departement
lebih telilti dan hati - hati pada saat menyajikan makanan ke customer

Kronologi
staff a/n ivana(chasier) dan memberitahukan untuk ikan dori ala sanghai ada ulatnya. ivana langsung meminta maaf ke customer.ivana melaporkan ke ibu mega(SM). maka langsung menghampiri customer untuk meminta maaf.

Discussion
Cek Quality pada saat penyajian makanan mulai dari barang yang dikirim dari warehouse ke tempat penyimpanan dan cek quality pada saat pengambilan dari tempat penyimpanan

Figure 16. Display Discussion Transactions

In this view the discussion is used to carry out a discussion of customer complaints that have been inputted by the restaurant staff. From the discussion, it must be able to get the solutions that occur and not be repeated. The results of the discussion are inputted and stored in the database.

### 5.7. Display Chart Report

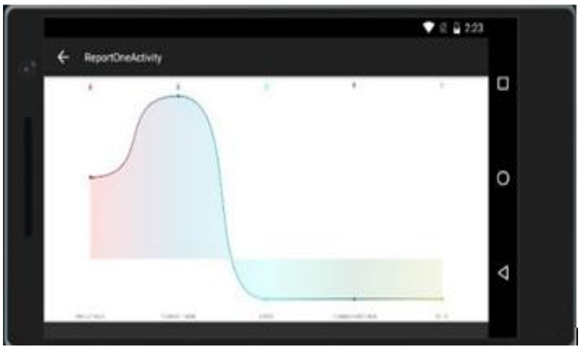
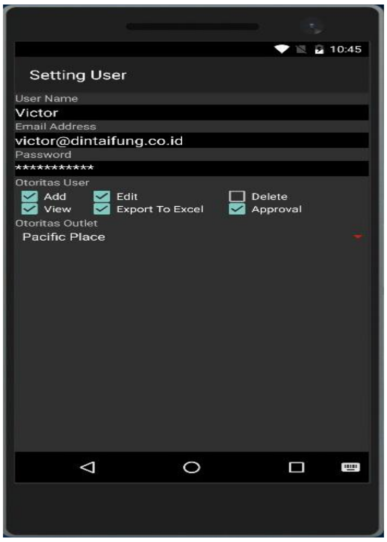


Figure 17. Display Chart Report

In this view, describing customer complaints in the form of charts, this report is used for management to conduct discussions in the restaurant.

### 5.8. Login



10:45

Setting User

User Name
Victor

Email Address
victor@dintaifung.co.id

Password
***********

Otoritas User
☑ Add ☑ Edit □ Delete
☑ View ☑ Export To Excel ☑ Approval

Otoritas Outlet
Pacific Place

Figure 18. Display User Setup

## VI. CONCLUSION

Based on the design, implementation and discussion that has been carried out in this study, the following conclusions can be drawn:

1. The design of this mobile-based customer complaint application is indeed very much needed by PT. DhanaJaya Bogaindo. This system makes it easier for departments related to customer complaints to improve service quality and food menus.
2. In addition, the process of data collection, calculation, report generation and presentation of information needed in decision making activities can be processed quickly and can reduce the level of error *(human error).*

## VII. ACKNOWLEDGEMENT

The authors would like to express our gratitude for the support provided by Faculty of Science and technology. The financial assistance from Research, Publication and Community Service Department, Buddhi Dharma University is also greatly acknowledged.

## REFERENCES

[1] Alfatta, Hanif 2007. Analisa dan perancangan Sistem Informasi Yogyakarta:Andi

[2] Ipung Purnomo,2018. Visual Studio Xamarin & CodeIgniter Koko media

[3] Xamarin, 2018, Xamarin Developer Doc.
Available at : https://developer.xamarin.com/api/root/MonoAndroid-lib/
[Accessed Nov 22nd 2018]

[4] Dennis Anchugov, 2016 CrossPlatform Software Development

[5] Roger S Pressman, 2005 Software Engineering.

[6] Rapid Application Development
Available at :https://piyaneo.wordpress.com/2014/05/10/rapid-application-development-rad/
[Accessed Dec 08th 2018]

[7] Valentin Rock,2015. Using MVVM for enhanced cross platform development ofmobile and desktop application, Master's Thesis.

[8] Microsoft Team, 2017, The Model-View-ViewModel Pattern,
Available at : https://docs.microsoft.com/en-us/xamarin/xamarin-forms/enterprise-application-patterns/mvvm
[Accessed Nov 01st 2018]

## BIOGRAPHY

**Alexius Hendra Gunawan,** Graduated in the Information Technology Study Program (S1) in 2005, continued his Masters in Information Systems in 2009 and graduated in 2011. He is currently a Lecturer Information Systems Study Program in Buddhi Dharma University

**Andri Wijaya,** Graduated in the Information Technology Study Program (S1) in 2011, continued his Masters in Information Systems in 2012 and graduated in 2016. He is currently a Lecturer Information Systems Study Program in Buddhi Dharma University

**Dellih Wijaya,** Graduated from the Information System Study Program (S1) in 2003, continued his Masters in Management In 2005 and graduated in 2009. She currently serves as a Lecturer in Information Systems Study Program, Buddhi Dharma University.