



SOAL

BIDANG LOMBA

IT-SOFTWARE APPLICATION FOR BUSINESS



**KEMENTERIAN PENDIDIKAN DAN KEBUDAYAAN
DIREKTORAT JENDERAL PENDIDIKAN MENENGAH
DIREKTORAT PEMBINAAN SEKOLAH MENENGAH KEJURUAN**

**Kompleks Kemdiknas Gedung E Lantai 13
Jalan Jenderal Sudirman Senayan Jakarta 10270
Telepon (021) 5725477 (hunting), 5725466-69, 5725471-75
Fax. 5725467, 5725469, 5725049**

Site: www.ditpsmk.net

PROJECT OVERVIEW

In this Test Project, you are required to develop an information system following the requirement given on the Test Project. In general, there will be 4 project resources given to you, which is:

1. ERD and Data Dictionary for Test Project

This will be used to guide you to create the database files. Ensure that all entities created on the database are related with the given ERD, following the relationship and also the criteria in Data Dictionary for each entity!

2. Navigation Diagram for the Application

Use the Navigation Diagram to develop the form interaction inside the application. You are free to add new interaction between forms, without neglecting the defined interaction in this diagram.

3. Example Design (Wireframe) of desired Information System

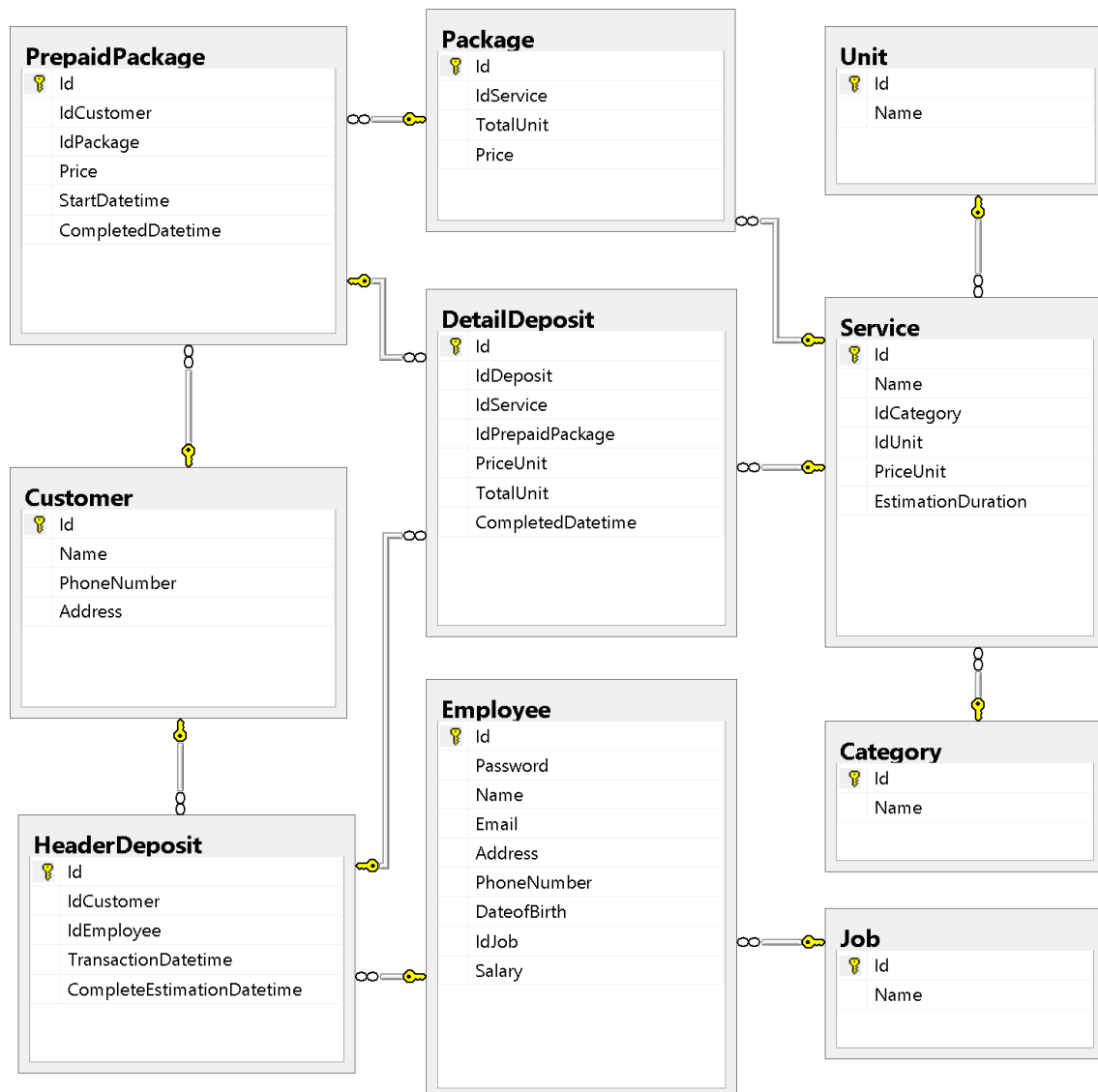
This file is used to give you design guidelines of all required forms. Please note that your form designs are not limited to these examples!

4. Data files (*if any*)

The data files will be used to support you test and develop the information system on each module. Please use this data files in the development, you are allowed to use another resources outside the given data files only if instructed in the module.

ESEMKA LAUNDRY INFORMATION SYSTEM

ENTITY RELATIONSHIP DIAGRAM



ESEMKA LAUNDRY INFORMATION SYSTEM

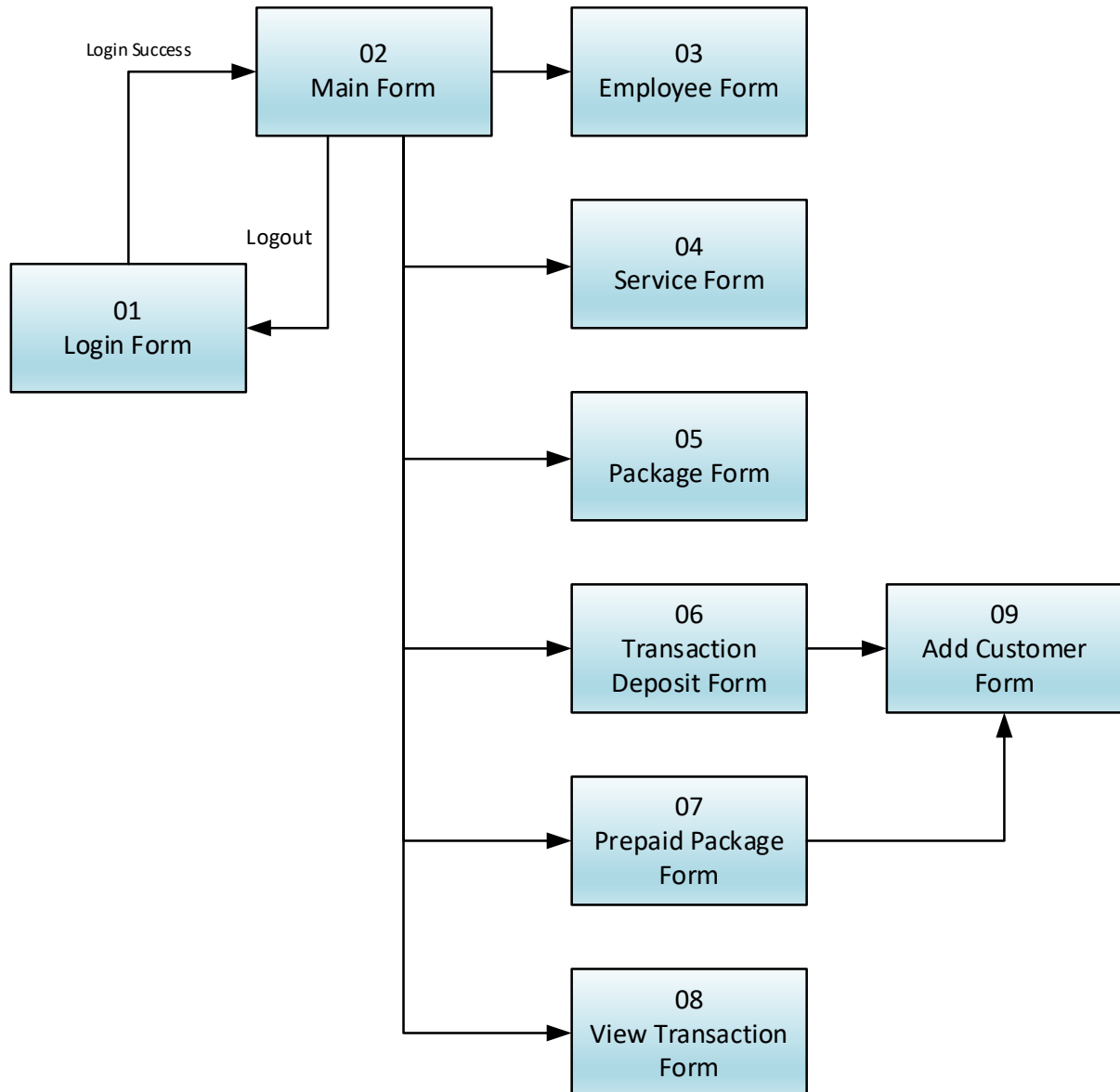
DATA DICTIONARY

Table	PK	FK	Column	Required	Type	Length	Notes
Category	PK		Id	Y	int		Auto Increment/Identity
			Name	Y	varchar	50	
Customer	PK		Id	Y	int		Auto Increment/Identity
			Name	Y	varchar	50	
			PhoneNumber	Y	varchar	20	
			Address	Y	varchar	200	
DetailDeposit	PK		Id	Y	int		Auto Increment/Identity
		FK	IdDeposit	Y	int		
		FK	IdService	Y	int		
		FK	IdPrepaidPackage	N	int		
			PriceUnit	Y	int		
			TotalUnit	Y	float		
			CompletedDatetime	N	datetime		
Employee	PK		Id	Y	int		Auto Increment/Identity
			Password	Y	varchar	50	
			Name	Y	varchar	50	
			Email	N	varchar	50	
			Address	Y	varchar	200	
			PhoneNumber	Y	varchar	20	
			DateOfBirth	Y	date		
		FK	IdJob	Y	int		
			Salary	Y	money		
HeaderDeposit	PK		Id	Y	int		Auto Increment/Identity
		FK	IdCustomer	Y	int		
		FK	IdEmployee	Y	int		
			TransactionDatetime	Y	datetime		
			CompleteEstimationDatetime	N	datetime		
Job	PK		Id	Y	int		Auto Increment/Identity
			Name	Y	varchar	50	
Package	PK		Id	Y	int		Auto Increment/Identity
		FK	IdService	Y	int		
			TotalUnit	Y	int		
			Price	Y	int		

Table	PK	FK	Column	Required	Type	Length	Notes
PrepaidPackage	PK		Id	Y	int		Auto Increment/Identity
		FK	IdCustomer	Y	int		
		FK	IdPackage	Y	int		
			Price	Y	int		
			StartDatetime	Y	datetime		
			CompletedDatetime	N	datetime		
Service	PK		Id	Y	int		Auto Increment/Identity
			Name	Y	varchar	50	
		FK	IdCategory	Y	int		
		FK	IdUnit	Y	int		
			PriceUnit	Y	int		
			EstimationDuration	Y	int		
Unit	PK		Id	Y	int		Auto Increment/Identity
			Name	Y	varchar	50	

ESEMKA LAUNDRY INFORMATION SYSTEM

NAVIGATION DIAGRAM



ESEMKA LAUNDRY INFORMATION SYSTEM

WIREFRAME


01 Login Form

The login form is titled "Esemka Laundry - Login". It features a light blue header with the text "Esemka Laundry". Below the header, there are two input fields: "Username" and "Password". The "Username" field contains the text "username". The "Password" field contains seven dots. Below the input fields, there are two buttons: "Reset" and "Login".

02 Main Form

The main form is titled "Esemka Laundry - Main". It features a light blue header with the text "Esemka Laundry". To the right of the header, there is a date and time "19 October 2020 12:43:52" and a greeting "Hello, username" followed by a "Logout" button. On the left side, there is a sidebar with a menu icon and a list of buttons: "Manage Employee", "Manage Service", "Manage Package", "Transaction Deposit", "Prepaid Package", and "View Transaction". The main content area is a large light blue rectangle with the text "Welcome" centered in the middle.

03 Employee Form


Esemka Laundry

19 October 2020 12:44:42
Hello, username
Logout

Manage Employee
Manage Service
Manage Package
Transaction Deposit
Prepaid Package
View Transaction

Manage Employee
Search


Employee Id	Name	Email	Phone Number	Address	Date of Birth	Job Title	Salary
1	Tono	Tono@gma...	081712345678	Jalan Kebu...	12/30/2000	Cashier	3600000

Employee ID
Name
Email
Phone Number
Address
Date Of Birth

Job Title
Salary
Password
Confirm Password

Insert
Update
Delete
Save
Cancel

04 Service Form


Esemka Laundry

19 October 2020 12:45:20
Hello, username
Logout

Manage Employee
Manage Service
Manage Package
Transaction Deposit
Prepaid Package
View Transaction

Manage Service
Search

Service Id	Service Name	Category	Unit	Price	Estimation Duration
1	Cuci Kiloan	Kiloan	KG	8000	48
1	Cuci Kiloan Kilat	Kiloan	KG	15000	24

Service ID
Service Name
Category
Unit

Price
Est Duration

Insert
Update
Delete
Save
Cancel

05 Package Form

Esemka Laundry

19 October 2020 12:45:53
Hello, username
Logout

Manage Employee
Manage Service
Manage Package
Transaction Deposit
Prepaid Package
View Transaction

Manage Package

Search

Package Id	Service Name	Total Unit	Price
1	Cuci Kiloan	20	150000

Package ID

Service

Total Unit

Price

Insert
Update
Delete
Save
Cancel

06 Transaction Deposit Form

Esemka Laundry

19 October 2020 01:09:54
Hello, username
Logout

Manage Employee
Manage Service
Manage Package
Transaction Deposit
Prepaid Package
View Transaction

View Transaction

Search

Header Deposit

Id	Customer Id	Customer Name	Employee Name	TransactionTime	Complete Estimation Time
1	1	Budi	Tono	10/19/2020 1:09:...	10/21/2020 1:09:...

Detail Deposit

ServiceName	PrepaidPackageld	Price per Unit	Total Unit	Complete Time	Action
Cuci Kiloan	1	8000	2.7		Complete
Cuci Kiloan Kilat		15000	1		Complete

07 Prepaid Package Form

Esemka Laundry - Prepaid Package
19 October 2020 01:07:56
Hello, username
Logout

Manage Employee
Manage Service
Manage Package
Transaction Deposit
Prepaid Package
View Transaction

Transaction Prepaid Package

Search

Prepaid Package Id	Customer	Package Name	Price
1	Budi	Cuci Kiloan 20KG	150000

Prepaid Package Id
Package
Price

Customer

Phone Number

[Not Found? Add New +](#)

Name

Address

08 View Transaction Form

Esemka Laundry - View Transaction
19 October 2020 01:09:54
Hello, username
Logout

Manage Employee
Manage Service
Manage Package
Transaction Deposit
Prepaid Package
View Transaction

View Transaction

Search

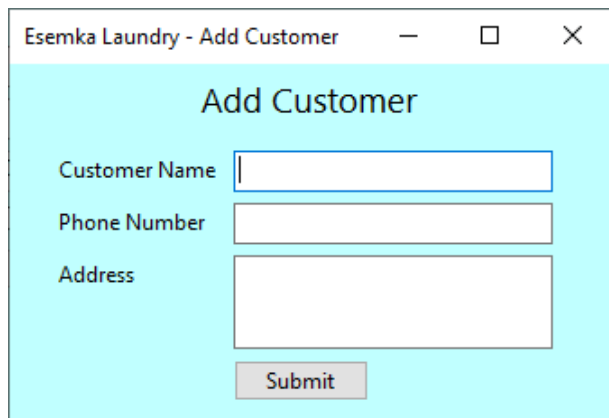
Header Deposit

Id	Customer Id	Customer Name	Employee Name	TransactionTime	Complete Estimation Time
1	1	Budi	Tono	10/19/2020 1:09:...	10/21/2020 1:09:...

Detail Deposit

ServiceName	PrepaidPackageld	Price per Unit	Total Unit	Complete Time	Action
Cuci Kiloan	1	8000	2.7		Complete
Cuci Kiloan Kilat		15000	1		Complete

09 Add Customer Form



The image shows a web browser window with the title "Esemka Laundry - Add Customer". The page has a light blue background and a white title bar. The main heading "Add Customer" is centered at the top. Below it, there are three input fields: "Customer Name" (a single-line text box), "Phone Number" (a single-line text box), and "Address" (a multi-line text box). A "Submit" button is located at the bottom right of the form area.

Esemka Laundry - Add Customer

Add Customer

Customer Name

Phone Number

Address

THE DEVELOPMENT OF ESEMKA LAUNDRY INFORMATION SYSTEM

Esemka Laundry is a laundry which located in Jakarta. During the rainy season this laundry is having a hard time to manage the transaction. There are several points that makes customer comes to this laundry which are the clean, the good service, cheap price, and located in very strategic location. As the time passed, the more customer is coming to this laundry and the owner thinks that now is the right time to find out the solution of the problem they got. The problems will be specified down below.

1. The customer wants the easier way to deposit they laundry. The system just needs input the customer data one time and able to search data.
2. The customer and the owner want more accurate estimation time to laundry.

The directors by then decided to hire famous consultant to analyze and design the system, it is the time to develop the system.

Based on the system design given to you, you're required to develop Information system to support the business process of Esemka Laundry. Ensure your developed solutions can accommodate Esemka Laundry business process properly, following the instruction on the project!

DATABASE DEVELOPMENT, DATA INSERTION, AND USER INTERFACE DESIGN

1. Create the Database

Create a database using your MS SQL Server on the local database server following the ERD and Data Dictionary given to you. Please be aware of the data type and the constraint of each table given.

- a. Create Table **Category**
- b. Create Table **Customer**
- c. Create Table **DetailDeposit**
- d. Create Table **Employee**
- e. Create Table **HeaderDeposit**
- f. Create Table **Job**
- g. Create Table **Package**
- h. Create Table **PrepaidPackage**
- i. Create Table **Service**
- j. Create Table **Unit**

2. Insert the Data

Insert the master data given to you on file “*master-data.xlsx*” to the related table on your newly created database.

- a. Insert Table **Category** from Master Data
- b. Insert Table **Customer** from Master Data

- c. Insert Table **DetailDeposit** from Master Data
- d. Insert Table **Employee** from Master Data
- e. Insert Table **HeaderDeposit** from Master Data
- f. Insert Table **Job** from Master Data
- g. Insert Table **Package** from Master Data
- h. Insert Table **PrepaidPackage** from Master Data
- i. Insert Table **Service** from Master Data
- j. Insert Table **Unit** from Master Data

3. Design User Interface

Design user interface for Esemka Laundry according to the **wireframe** given above using your preferred platform (**C# or Visual Basic.NET**). Please be aware of using the right GUI Component used on the wireframe above.

- a. Design **Login Form** Interface
- b. Design **Main Form** Interface
- c. Design **Employee Form** Interface
- d. Design **Service Form** Interface
- e. Design **Package Form** Interface
- f. Design **Transaction Deposit Form** Interface
- g. Design **Prepaid Package Form** Interface
- h. Design **View Transaction Form** Interface

- i. Design **Add Customer Form** Interface

MASTER FORM DEVELOPMENT

1. Create “01 Login Form”

Create the main menu of the application as outlined in “**01 Login Form**” in the wireframe. Email and password of the employee will be used to do login process. During login process, there are several processes that needs to be done:

- a. If the email or password does not match the data on database then the program will show a message “**Please Try Again, Your Data is not Valid!**”
- b. If the login process is success then user will be direct into **Main Form** and **Login Form** will be hidden.

2. Create “02 Main Form”

Create the main menu of the application as outlined in “**02 Main Form**” in the wireframe. This form will be containing all menus which already described on the wireframe above. This form will be acted as a MDI Form or Parent Form where every other form will be comes inside it. Below are the menus which need to be defined on this form:

- a. Logout, Employee, Service, Package, Transaction Deposit, Prepaid Package, and View Transaction.
- b. Each menu is direct to the right specific form (Such as Logout will close the Main Form and hide it then show the Login Form, for detail process please look at the wireframe given)
- c. Show the username based on the login form.
- d. Show the real current time.

3. Create “03 Employee Form”

Create the main menu of the application as outlined in “**03 Employee Form**” in the wireframe. Below is the specification process needed of this form:

- a. When the **form started**, then:
 - i. **show Employee data** from database on DataGridView.
 - ii. **load Job data** on Job combobox
 - iii. **disable** all fields, save button and cancel button
 - iv. **enable** insert button, update button, and delete button
- b. **Textfield Search** can be done the **searching process** where based on **employee name, email, or phone**.
- c. When user **select the row** on datagridview, the data will be show on the field.
- d. When **Insert Button** is **clicked**, then:
 - **clear** all fields, and **enable** all fields
 - **disable** insert button, update button, and delete button
 - **enable** save button and cancel button
- e. **Update Button** is only can **be clicked** after specific row in data grid is **selected**. After the **Edit button** is **clicked**:
 - **enable** all fields, save button and cancel button
 - **disable** insert button, update button, and delete button
- f. **Delete Button** is only can **be clicked** after specific row in data grid is **selected**. After the delete button is clicked, the confirmation message will be show. If the user clicks the **Yes Button**, the data will be **deleted** from database.

g. When **Save Button** is clicked, then:

i. Validate the data, base on this rule:

- All field must be filled
- **Email** must be valid format.
- **Phone number** must be start with "+" and another with digit
- **Password** must be consist a Alphabet, a Numeric and an Symbol.
- **Password and Confirm Password** must be same

ii. When the any data **is not valid**, then **show** the error message. When the all data **is valid**, then:

- **insert or update** the data to **database**
- **disable** all fields, save button and cancel button
- **enable** insert button, update button, and delete button

h. When **Cancel Button** is clicked, then:

- i. **disable** all fields, save button and cancel button
- ii. **enable** insert button, update button, and delete button

4. Create "04 Service Form"

Create the main menu of the application as outlined in "**04 Service Form**" in the wireframe. Below is the specification process needed of this form:

a. When the **form started**, then:

- i. **show Service data** from database on DataGridView.

- ii. **Load Category data** on category combobox
 - iii. **Load Unit data** on category combobox
 - iv. **disable** all fields, save button and cancel button
 - v. **enable** insert button, update button, and delete button
- b. **Textfield Search** can be done the **searching process** where based on **Service name, category, unit or price**.
- c. When user **select the row** on datagridview, the data will be show on the field.
- d. When **Insert Button** is **clicked**, then:
- **clear** all fields, and **enable** all fields
 - **disable** insert button, update button, and delete button
 - **enable** save button and cancel button
- e. **Update Button** is only can **be clicked** after specific row in data grid is **selected**. After the **Edit button** is **clicked**:
- **enable** all fields, save button and cancel button
 - **disable** insert button, update button, and delete button
- f. **Delete Button** is only can **be clicked** after specific row in data grid is **selected**. After the delete button is clicked, the confirmation message will be show. If the user clicks the **Yes Button**, the data will be **deleted** from database.
- g. When **Save Button** is clicked, then:
- i. Validate the all field must be filled.

- ii. When the any data **is not valid**, then **show** the error message. When the all data **is valid**, then:

- **insert or update** the data to **database**
- **disable** all fields, save button and cancel button
- **enable** insert button, update button, and delete button

- h. When **Cancel Button** is clicked, then:

- i. **disable** all fields, save button and cancel button
- ii. **enable** insert button, update button, and delete button

5. Create “05 Package Form”

Create the main menu of the application as outlined in “**10 Package Form**” in the wireframe. Below is the specification process needed of this form:

- a. When the **form started**, then:
 - i. **show Package data** from database on DataGridView.
 - ii. **load Service data** on service combobox
 - iii. **disable** all fields, save button and cancel button
 - iv. **enable** insert button, update button, and delete button
- b. **Textfield Search** can be done the **searching process** where based on **Package name, total unit, or price**.
- c. When user **select the row** on datagridview, the data will be show on the field.
- d. When **Insert Button** is **clicked**, then:
 - **clear** all fields, and **enable** all fields

- **disable** insert button, update button, and delete button
 - **enable** save button and cancel button
- e. **Update Button** is only can **be clicked** after specific row in data grid is **selected**. After the **Edit button** is **clicked**:
- **enable** all fields, save button and cancel button
 - **disable** insert button, update button, and delete button
- f. **Delete Button** is only can **be clicked** after specific row in data grid is **selected**. After the delete button is clicked, the confirmation message will be show. If the user clicks the **Yes Button**, the data will be **deleted** from database.
- g. When **Save Button** is clicked, then:
- i. Validate the **all field** must be filled
 - iii. When the any data **is not valid**, then **show** the error message. When the all data **is valid**, then:
 - **insert or update** the data to **database**
 - **disable** all fields, save button and cancel button
 - **enable** insert button, update button, and delete button
- h. When **Cancel Button** is clicked, then:
- i. **disable** all fields, save button and cancel button
 - ii. **enable** insert button, update button, and delete button

TRANSACTION FORM DEVELOPMENT

1. Create “06 Transaction Deposit Form”

Create the main menu of the application as outlined in “06 Transaction Deposit Form” in the wireframe. Below is the specification process needed of this form:

- a. When the **form started**, then:
 - i. **load Service data from database** on service combobox
 - ii. **show the current time**
- b. The user can **search customer** by **phone number**.
- c. When the “**Add New +**” link is clicked, then show add customer form as dialog.
- d. When **Add button** is clicked, then **add selected data** to datagridview. And **calculate the subtotal, total pay, and estimation time**.
- e. When **Remove button** is clicked, the **remove selected row** from datagridview. And **calculate the subtotal, total pay, and estimation time**.
- f. To **calculate the subtotal**, the **subtotal** is **PricePerUnit** times **TotalUnit**. Calculate the subtotal, if prepaid package is empty.
- g. To calculate the estimation time, if the category’s service is “Kiloan”, just using the **estimation time per service**. Otherwise, using total unit times **estimation time per service**. And total all estimation time.
- h. When **Submit Button** is clicked, then validate the customer must be selected and the count data on datagridview more than or equals one. If the data is not valid, show the error message. Otherwise, insert to database.

2. Create “07 Prepaid Package Form”

Create the main menu of the application as outlined in “**07 Prepaid Package Form**” in the wireframe. Below is the specification process needed of this form:

- a. When the **form started**, then:
 - i. **show Prepaid Package data** from database on DataGridView. For the package name, combine the service name, total unit, and unit name
 - ii. **load Package data** on package combobox
- b. **Textfield Search** can be done the **searching process** where based on **Prepaid Package customer, or package name.**
- c. When the **Submit button** is clicked, then validate all field must be field and the customer must be selected. If the data is not valid, then show the error message. Otherwise, insert the data to database.
- d. When the “**Add New +**” link is clicked, then show add customer form as dialog.

3. Create “**08 View Transaction Form**”

Create the main menu of the application as outlined in “**08 View Transaction Form**” in the wireframe. Below is the specification process needed of this form:

- a. When the **form started**, then **show Deposit data** from database on DataGridView.
- b. **Textfield Search** can be done the **searching process** where based on **Customer name, Employee name, or TransactionDateTime.**
- c. When user **select the row** on first datagridview, the detail data will be show on the second datagridview .
- d. When the **complete datetime** is null, then show the **complete button.**

- e. When the **Complete button** is clicked, then add the current time to CompleteDatetime Column on selected row.

4. Create “09 Add Customer Form”

Create the main menu of the application as outlined in “09 Add Customer Form” in the wireframe. Below is the specification process needed of this form:

- a. When the **Add button** is clicked, then validate the all field **must filled** and the phone number must be **unique in database**. If the data is not valid, then show the error message. Otherwise, insert the data to database and close.