



SOAL

BIDANG LOMBA

IT-SOFTWARE APPLICATION FOR BUSINESS Tahun 2021



**DINAS PENDIDIKAN PROVINSI JAWA TIMUR
BIDANG PEMBINAAN PENDIDIKAN SMK**

Jalan Geteng kali Nomor 33 Tlp. (031) 5342706-08 Fax. 5341107Telepon (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 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.

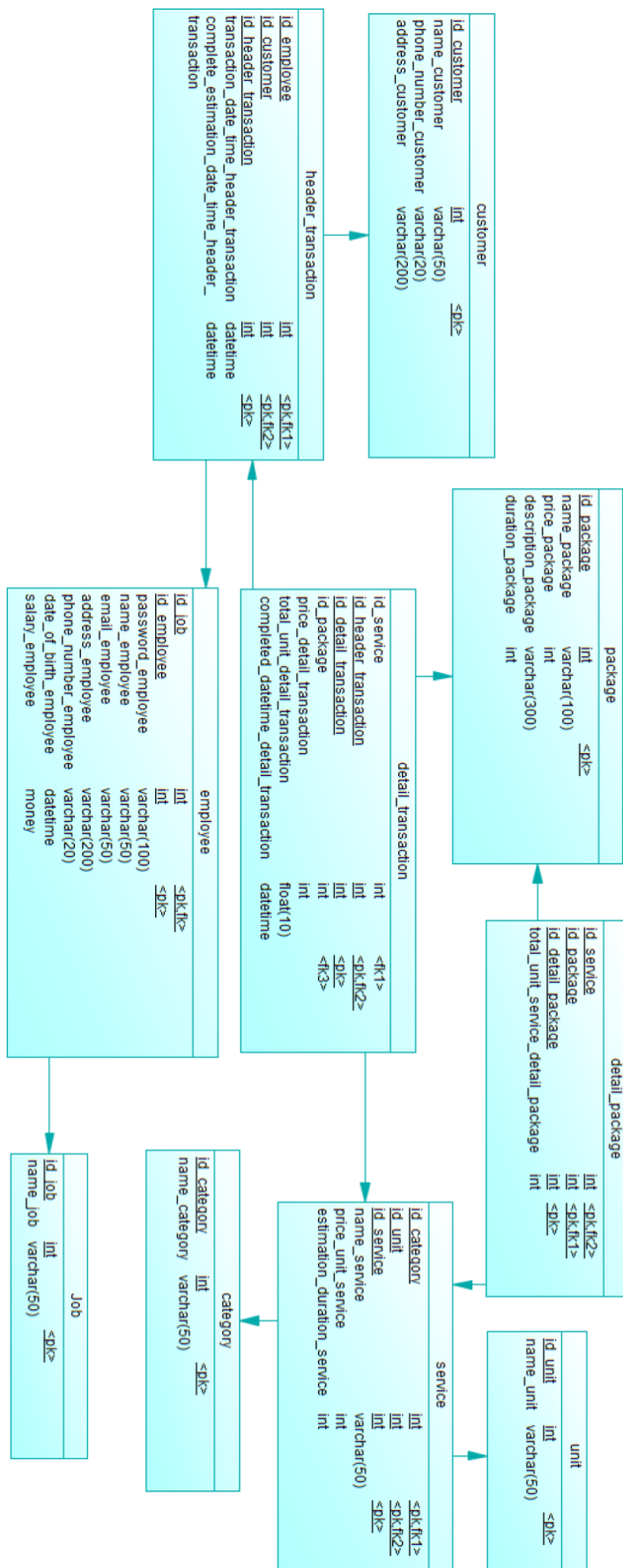
3. Example Design (Wireframe) of desired Information System

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

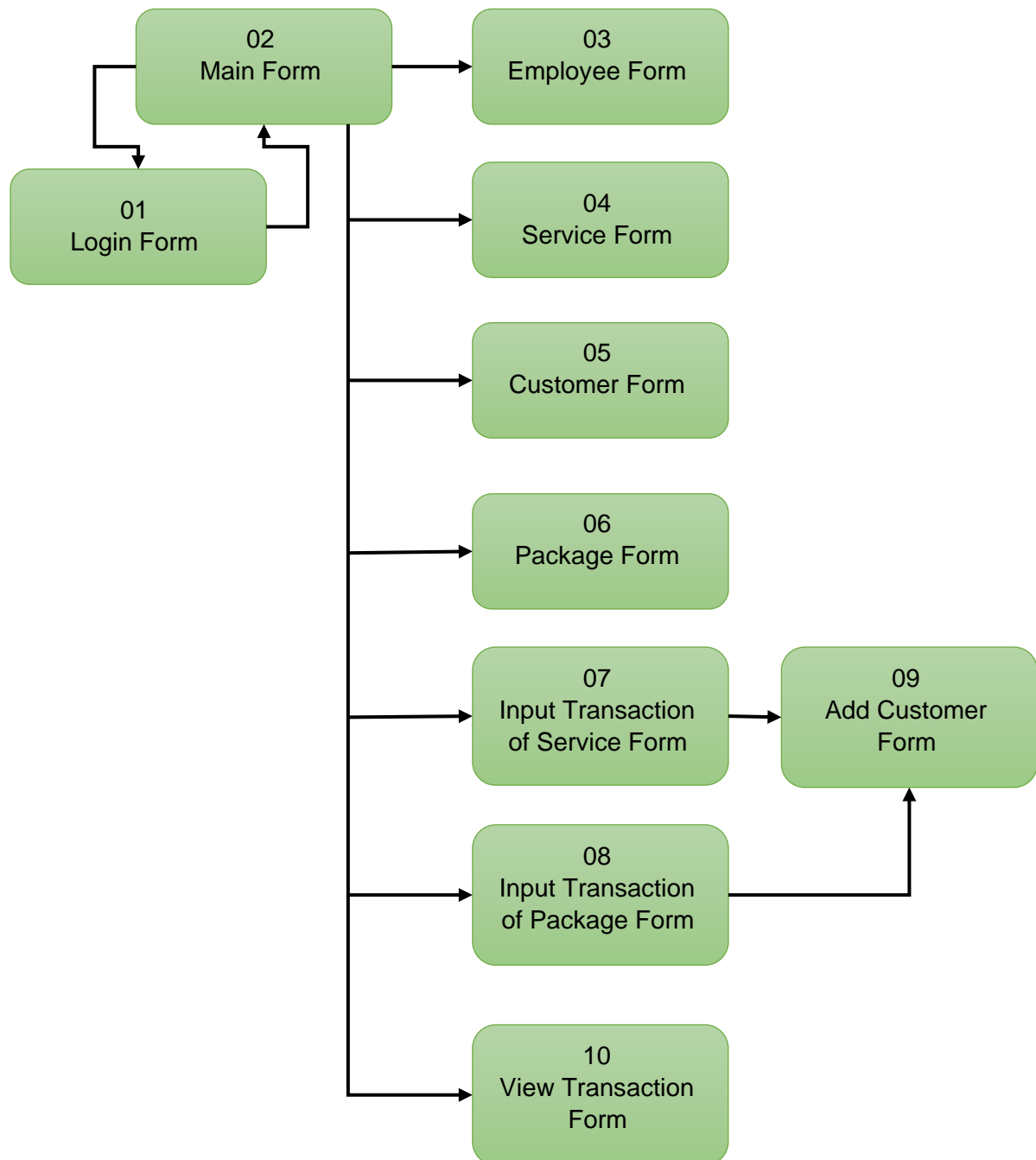
4. Data files (if any)

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 other resources outside the given data files only if instructed in the module.

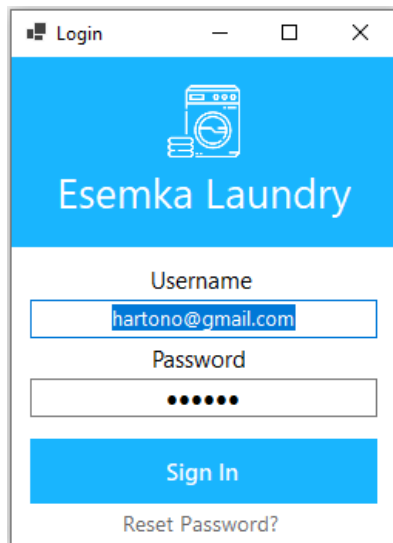
ESEMKA LAUNDRY INFORMATION SYSTEM ENTITY RELATIONSHIP DIAGRAM



ESEMKA LAUNDRY INFORMATION SYSTEM NAVIGATION DIAGRAM

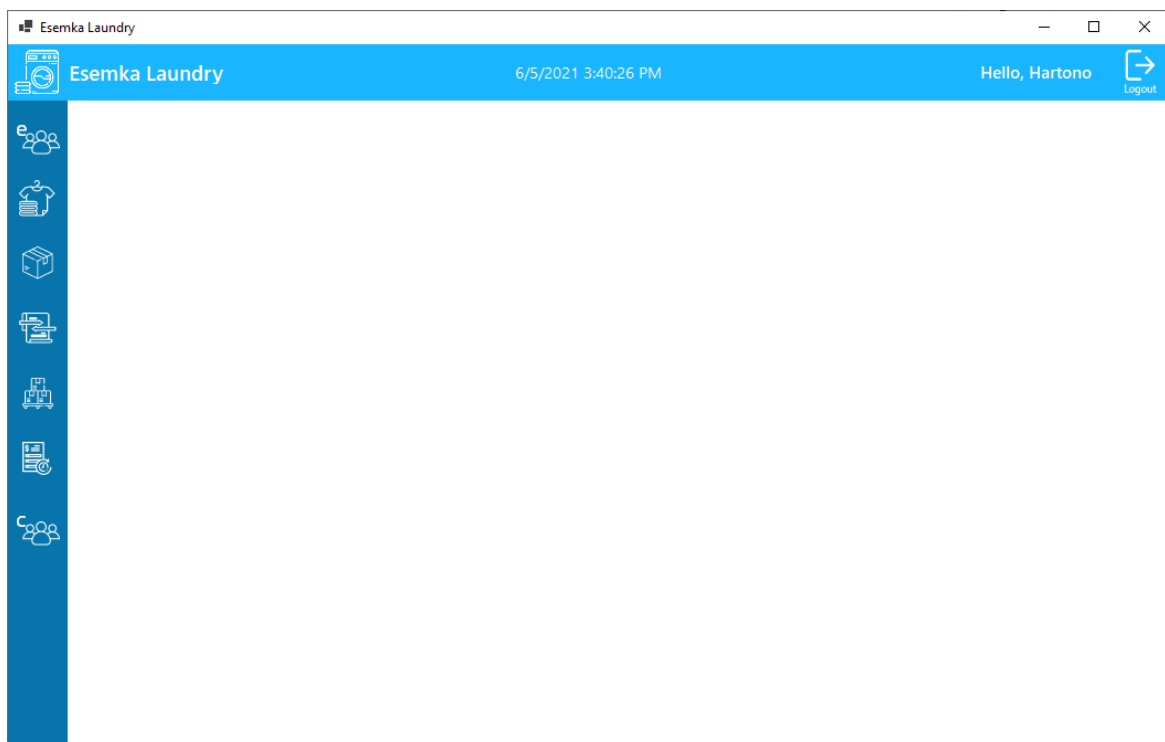


01 Login Form



The screenshot shows a web browser window titled "Login". The page has a blue header with a washing machine icon and the text "Esemka Laundry". Below the header, there are two input fields: "Username" with the value "hartono@gmail.com" and "Password" with masked characters "•••••". A blue "Sign In" button is positioned below the password field, and a "Reset Password?" link is at the bottom.

02 Main Form



The screenshot shows the main interface of the Esemka Laundry application. It features a blue header bar with the application name "Esemka Laundry", the date and time "6/5/2021 3:40:26 PM", and a user greeting "Hello, Hartono" with a "Logout" button. A vertical blue sidebar on the left contains several icons representing different functions: a group of people, a document, a box, a printer, a laundry machine, a calendar, and a group of people. The main content area is a large, empty white space.

03 Employee Form

Esemka Laundry

6/5/2021 3:49:32 PM

Hello, Hartono

Logout

Manage Employee

Insert Data

Search

	Id Employee	Name	Email	Phone Number	Address	Date of Birth	Job Title	Salary
▶	1	Admin	admin@admin.com	+6200000000	admin	1/1/2021	Pencuci	Rp2.500.000
	2	Hartono	hartono@gmail.co...	+6281278122245	keputih no. 2	4/22/1990	Pencuci	Rp2.500.000
	3	Azaon	azaon@gmail.com	+6285190763411	keputih Gebang n...	3/29/1992	Pencuci	Rp2.500.000
	4	Tini	tini@gmail.com	+6281900765418	keputih Lor no. 98	2/15/1991	Penyetrika	Rp2.500.000
	5	Ubed	ubed@gmail.com	+6287258412568	Gebang Kejawab n...	1/30/1993	Penyetrika	Rp2.500.000
	6	Komar	komar@gmail.com	+6281254542687	Keputih Marina no...	8/25/1993	Kurir	Rp2.500.000
*								

Update

Delete

Esemka Laundry

6/6/2021 9:39:25 AM

Hello, Hartono

Logout

Manage Employee

	Id Employee	Name	Email	Phone Number	Address	Date of Birth	Job Title	Salary
▶	1	Admin	admin@admin.com	+6200000000	admin	1/1/2021	Pencuci	Rp2.500.000
	2	Hartono	hartono@gmail.co...	+6281278122245	keputih no. 2	4/22/1990	Pencuci	Rp2.500.000
	3	Azaon	azaon@gmail.com	+6285190763411	keputih Gebang n...	3/29/1992	Pencuci	Rp2.500.000
	4	Tini	tini@gmail.com	+6281900765418	keputih Lor no. 98	2/15/1991	Penyetrika	Rp2.500.000
	5	Ubed	ubed@gmail.com	+6287258412568	Gebang Kejawab n...	1/30/1993	Penyetrika	Rp2.500.000
	6	Komar	komar@gmail.com	+6281254542687	Keputih Marina no...	8/25/1993	Kurir	Rp2.500.000
*								

Name

Email

Phone Number

Password

Confirm Password

Alamat

Job Title

Pencuci

Date Of Birth

Sunday , June 6, 2021

Salary

500,000

Save

Cancel

04 Service Form

Esemka Laundry

6/6/2021 9:40:12 AM

Hello, Hartono

Logout

Manage Service

Insert Data

Search

Service ID	Service Name	Category	Unit	Price	Estimation Duration
1	Cuci Kiloan	Kiloan	Kg	Rp20.000	4
2	Cuci Seterika	Kiloan	Kg	Rp30.000	6
4	Cuci Kilat	Kiloan	Kg	Rp30.000	1
5	Seterika Kilat	Kiloan	Kg	Rp20.000	1
3	Cuci Korden	Meteran	m	Rp10.000	2
*					

Update

Delete

Esemka Laundry

6/6/2021 9:40:39 AM

Hello, Hartono

Logout

Manage Service

Search

Service ID	Service Name	Category	Unit	Price	Estimation Duration
1	Cuci Kiloan	Kiloan	Kg	Rp20.000	4
2	Cuci Seterika	Kiloan	Kg	Rp30.000	6
4	Cuci Kilat	Kiloan	Kg	Rp30.000	1
5	Seterika Kilat	Kiloan	Kg	Rp20.000	1
3	Cuci Korden	Meteran	m	Rp10.000	2
*					

Service Name

Price

0

Category

Kiloan

Est Duration

0

Hour(s)

Unit

Kg

Save

Cancel

05 Customer Form

Esemka Laundry

6/7/2021 2:42:34 AM

Hello, Hartono

Logout

Manage Customer

Insert Data

Search

Customer Name	Phone Number	Address
Dodo	Surabaya	+08111222333
Didi	Malang	+0822223333444
Koko	Sidoarjo	+0833334444555
Kiki	Mojokerto	+08666677778
Lala	Jombang	+0877766655
Lulu	Gresik	+0844443322

Update

Delete

Esemka Laundry

6/7/2021 2:42:50 AM

Hello, Hartono

Logout

Manage Customer

Search

Customer Name	Phone Number	Address
Dodo	Surabaya	+08111222333
Didi	Malang	+0822223333444
Koko	Sidoarjo	+0833334444555
Kiki	Mojokerto	+08666677778
Lala	Jombang	+0877766655
Lulu	Gresik	+0844443322

Name: Dodo Address: Surabaya

Phone Number: +08111222333

Cancel Save

06 Package Form

Esemka Laundry

6/6/2021 9:41:32 AM

Hello, Hartono

Logout

Manage Package

Insert Data

Search

Package Name	Description	Price	Estimation Duration
Paket Hari Raya	paket untuk hari raya	Rp120.000	14
Paket Kilat	Cuci Setrika secepat kilat	Rp200.000	6
Paket Rombongan	Untuk Orang banyak	Rp160.000	5

Detail Package

Service Name	Price	Estimation Duration	Total Unit	Action
Cuci Kiloan	Rp20.000	4	1	Delete
Cuci Seterika	Rp30.000	6	2	Delete
Cuci Korden	Rp10.000	2	2	Delete

Service

Total Unit

0

Add

Update

Delete

Esemka Laundry

6/6/2021 8:17:44 PM

Hello, Hartono

Logout

Manage Package

Search

Package Name	Description	Price	Estimation Duration
Paket Hari Raya	paket untuk hari raya	Rp120.000	14
Paket Kilat	Cuci Setrika secepat kilat	Rp200.000	6
Paket Rombongan	Untuk Orang banyak	Rp160.000	5

Detail Package

Service Name	Price	Estimation Duration	Total Unit	Action
Cuci Kiloan	Rp20.000	4	1	Delete
Cuci Seterika	Rp30.000	6	2	Delete
Cuci Korden	Rp10.000	2	2	Delete

Package Name

Paket Hari Raya

Description

paket untuk hari raya

Save

Cancel

07 Input Transaction of Service Form

Esemka Laundry

6/6/2021 10:01:35 AM

Hello, Hartono [Logout](#)

Input Service Transaction

Service Name	Service Price	Total Unit	Estimation Time per Service	Sub Total
Cuci Kiloan	Rp20.000	1	4	Rp20.000
Seterika Kilat	Rp20.000	1	1	Rp20.000

Service

Cuci Kiloan

Total Unit

0.0

Add

Customer

Phone Number

0833334444555

[Not Found? Add New +](#)

Name

Koko

Address

Sidoarjo

Remove

Total Pay

Rp40.000

Estimation Time

5 Hours

Submit

Succes

i

Transaction Added!

OK

08 Input Transaction of Package Form

Esemka Laundry

6/6/2021 10:13:09 AM

Hello, Hartono [Logout](#)

Input Package Transaction

Search

Package Name	Package Price	Quantity	Estimation Time per Package	Sub Total
Paket Hari Raya	120000	1	14	Rp120.000

Package: Paket Hari Raya

Quantity: 0

Add

Customer

Phone Number: 0877766655 [Net Found? Add New +](#)

Name: Lala

Address: Jombang

Total Pay

Rp120.000


Estimation Time

14 Hours

Submit

Remove

Succes

 Transaction Added!

OK

09 Add Customer Form

AddUserForm

Add Customer

Customer Name

Phone Number

Address

Submit

10 View Transaction Form

Esemka Laundry
Hello, Hartono

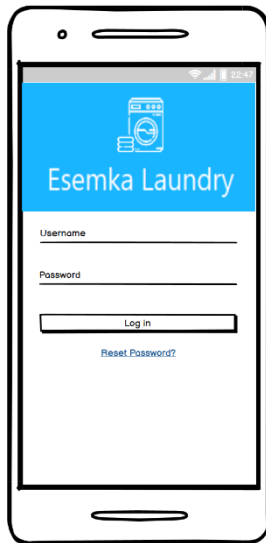
View Transaction

6/7/2021 12:29:00 AM

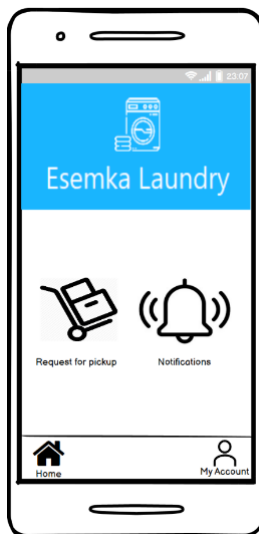
Header Transaction						
	ID	Customer ID	Customer Name	Employee Name	Transaction Time	Complete Estimation Time
	1	1	Dodo	Hartono	6/6/2021 9:52 AM	6/6/2021 7:52 PM
▶	5	1	Dodo	Hartono	6/6/2021 11:24 PM	6/7/2021 5:24 AM
	2	2	Didi	Hartono	6/6/2021 9:54 AM	6/7/2021 5:54 AM
	3	3	Koko	Hartono	6/6/2021 10:01 AM	6/6/2021 3:01 PM
	4	5	Lala	Hartono	6/6/2021 10:14 AM	6/7/2021 12:14 AM
*						

Detail Transaction						
	Name	Type	Price per Unit	Total Unit	Complete Time	Action
▶	Cuci Kiloan	Service	Rp20.000	1		Complete
	Seterika Kilat	Service	Rp20.000	1		Complete
	Cuci Kilat	Service	Rp60.000	2		Complete
*						

11 Mobile Application Login



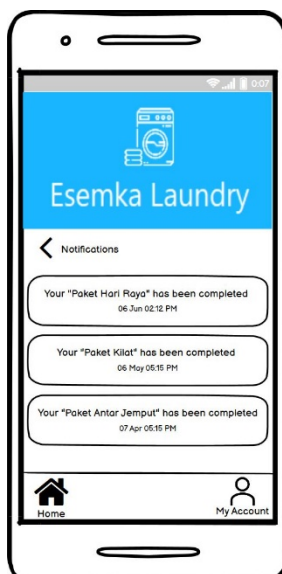
12 Mobile Application Main Menu



13 Mobile Application Request for Pickup



14 Mobile Application Notifications



THE DEVELOPMENT OF ESEMKA LAUNDRY INFORMATION SYSTEM

Esemka Laundry is a laundry located in Surabaya. During the rainy season, this laundry has a hard time managing the transaction. Several points make customers come to this laundry: the clean, the good service, low price, and located in a very strategic location. As time passed, more customers are coming to this laundry, and the owner thinks that now is the right time to find out the solution to the problem they got. The description below will specify the problems below.

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

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

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

Module 1: Database Creation and Main Form Development

A. 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 **DetailTransaction**
- d. Create Table **Employee**
- e. Create Table **HeaderTransaction**
- f. Create Table **Job**
- g. Create Table **Package**
- h. Create Table **DetailPackage**
- i. Create Table **Service**
- j. Create Table **Unit**

B. 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 **DetailTransaction** from Master Data
- d. Insert Table **Employee** from Master Data
- e. Insert Table **HeaderTransaction** from Master Data
- f. Insert Table **Job** from Master Data
- g. Insert Table **Package** from Master Data
- h. Insert Table **DetailPackage** from Master Data
- i. Insert Table **Service** from Master Data
- j. Insert Table **Unit** from Master Data

C. 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 Form** Interface
- g. Design **Input Service Transaction Form** Interface
- h. Design **Input Package Transaction Form** Interface
- i. Design **View Transaction Form** Interface
- j. Design **Customer Form** Interface
- k. Design **Add Customer Form** Interface

Modul 2: Master Form Development

A. 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 successful, the user will be directed into **Main Form**, and the system will **hide the Login Form**.

B. 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 already described on the wireframe above. This form will be acted as an MDI Form or Parent Form where every other form will come inside it. Below are the menus which need to be defined on this form:

- a. Logout, Employee, Service, Package, Input Service Transaction, Input Package Transaction, View Transaction, Customer.
- 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.

C. 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 show **Employee data** from the database on DataGridView.
- b. **Textfield Search** can be done the **searching process** were based on **employee name, email, or phone**.
- c. When user **select the row** on datagridview, then click on the **update button**, some field to update will appear at the bottom of the form containing all data from the selected row.
- d. When **Insert Data Button** is **clicked**, then:
 1. Some field to update will appear at the bottom of the form with empty fields.
 2. Hide the **Insert Data Button**.
- e. When the **update button** is clicked while the row in the datagridview is not selected, it will display an error, the program will show a message “**Please, select One Row!**”
- f. When the **delete button** is clicked while the row in the datagridview is not selected, it will display an error, the program will show a message “**Please, select One Row!**”
- g. When the **delete button** is clicked while the row in the datagridview is selected, the program will show a confirmation message “**Are you Sure Delete [name_of_employee]?**” (Example: “Are you Sure Delete Hartono?”). If the user clicks the **Yes** Button, the data will be deleted from database. If the user clicks the **No** button, the confirmation window will be closed.
- h. When **Save Button** is clicked, then:
 1. Validate the data, based 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 consisting of an Alphabet, a Numeric and a Symbol.
 - ✓ Password and Confirm Password must be same
2. When any data **is not valid**, then **show** the error message. When all data **is valid**, then:
 - ✓ insert or update the data to database.
 - ✓ Hide all fields at the bottom of the form.
 3. Show the **Insert Data Button**.
- i. When **Cancel Button** is clicked, then:
 1. Hide all fields at the bottom of the form.
 2. Show the **Insert Data Button**.

D. 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 **show Service data** from database on DataGridView.
- b. **Textfield Search** can be done by the **searching process** based on Service name, category, unit, or price.
- c. When user **select the row** on datagridview, then click on the **update button**, some field to update will appear at the bottom of the form containing all data from the selected row, the **Insert Data Button** must be hidden.
- d. When **Insert Data Button** is **clicked**, then:
 - a. Some field to update will appear at the bottom of the form with empty fields.
 - b. Hide the **Insert Data Button**.
- e. When the **update button** is clicked while the row in the datagridview is not selected, it will display an error, the program will show a message “**Please, select One Row!**”

- f. When the delete button is clicked while the row in the datagridview is not selected, it will display an error, the program will show a message **"Please, select One Row!"**
- g. When the **delete button** is clicked while the row in the datagridview is selected, the program will show a confirmation message **"Are you Sure Delete [name_of_service]?"** (Example: "Are you Sure Delete Cuci Kiloan ?"). If the user clicks the **Yes Button**, the data will be deleted from database. If the user clicks the **No** button, the confirmation window will be closed.
- h. When **Save Button** is clicked, then:
 - 1. Validate all fields must be filled.
 - 2. When any data **is not valid**, then **show** the error message.
 - 3. When all data **is valid**, then:
 - ✓ insert or update the data to **database**.
 - ✓ Hide all fields at the bottom of the form.
 - 4. Show the **Insert Data Button**.
- i. When **Cancel Button** is clicked, then:
 - 1. Hide all fields at the bottom of the form.
 - 2. Show the **Insert Data Button**.

E. Create "05 Customer Form"

Create the main menu of the application as outlined in **"05 Customer Form"** in the wireframe. Below is the specification process needed of this form:

- a. When the **form started**, then **show Customer data** from database on DataGridView.
- b. **Text field Search** can be done the **searching process** were based on **Customer name, Address, or phone number**.

- c. When user **select the row** on data grid view, then click on the **update button**, some field to update will appear at the bottom of the form containing all customer data from the selected row, the **Insert Data Button** must be hidden.
- d. When **Insert Data Button** is **clicked**, then:
 - a. Some field to update customer data will appear at the bottom of the form with empty fields.
 - b. Hide the **Insert Data Button**.
- e. When the **update button** is clicked while the row in the data grid view is not selected, it will display an error, the program will show a message "**Please, select One Row!**"
- f. When the delete button is clicked while the row in the data grid view is not selected, it will display an error, the program will show a message "**Please, select One Row!**"
- g. When the **delete button** is clicked while the row in the data grid view is selected, the program will show a confirmation message "**Are you Sure Delete [name_of_customer]?**" (Example: "Are you sure to delete Didi?"). If the user clicks the **Yes** Button, the data will be deleted from database. If the user clicks the **No** button, the confirmation window will be closed.
- h. When **Save Button** is clicked, then:
 - 1. Validate all fields must be filled.
 - 2. When any data **is not valid**, then **show** the error message.
 - 3. When all data **is valid**, then:
 - ✓ insert or update the data to **database**.
 - ✓ Hide all fields at the bottom of the form.
 - 4. Show the **Insert Data Button**.
- i. When **Cancel Button** is clicked, then:
 - 1. Hide all fields at the bottom of the form.
 - 2. Show the **Insert Data Button**.

Modul 3: Transaction Form Development

A. Create “06 Package Form”

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

- a. When the **form started**, then **show Package data** from database on DataGridView Package.
- b. **Textfield Search** can be done the **searching process** were based on **Package name**.
- c. When a Row is selected in the Package DataGridView, then:
 1. Service Package details associated with that Package are displayed on the Package DataGridView.
 2. Some field to add **the service** associated with the selected package will appear at the bottom of the form containing the **service name**, which is filled from the **service table**, **total unit** and **Add** button.
- d. When user **select the row** on Package datagridview, then click on the **update button**, some field to update **the package** will appear at the bottom of the form containing all data from the selected row, the **Insert Data Button** must be hidden.
- e. When **Insert Data Button** is **clicked**, then:
 - a. Some field to update will appear at the bottom of the form with empty fields, which is used to update the package.
 - b. Hide the **Insert Data Button**.
- f. When the **update button** is clicked while the row in the **Package datagridview** is not selected, it will display an error, the program will show a message “**Please, select One Row!**”
- g. When the delete button is clicked while the row in the **Package datagridview** is not selected, it will display an error, the program will show a message “**Please, select One Row!**”

- h. When the **delete button** is clicked while the row in the package datagridview is selected, the program will show a confirmation message "**Are you Sure Delete [name_of_package]?**" (Example: "Are you Sure Delete Paket Hari Raya ?"). If the user clicks the **Yes** Button, the data will be deleted from database. If the user clicks the **No** button, the confirmation window will be closed.
- i. When **Save Button** is clicked, then:
1. Validate all field must be filled.
 2. When any data **is not valid**, then **show** the error message.
 3. When all data **is valid**, then:
 - ✓ insert or update the package data to **database**.
 - ✓ Hide all fields at the bottom of the form.
 4. Show the **Insert Data Button**.
- j. When **Cancel Button** is clicked, then:
1. Hide all fields at the bottom of the form.
 2. Show the **Insert Data Button**.
- k. Users can **add services** to the **related package** with the following steps:
1. Choose a service from the combo box which is filled from the **service table**.
 2. fill in the desired total service unit.
 3. Pressing the add button, the program must update the database and then display it on the **service datagridview**.
- l. User can **delete** the service from the selected package by pressing the **delete button** in the column **action** in the **service datagridview**.

B. Create “07 Input Transaction of Service Form”

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

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

1. **Load service data from database** on service combobox.
2. Clear the transaction data in **datagridview**.

b. User can add the transaction of services with the following steps:

1. Choose the **service** from combobox.
2. Enter the desired **Total Unit**.
3. Click the **Add Button**, then the program should update the data to the database.
4. The system must check whether the total unit is more than 1 unit or not, if the total unit is less than 1 unit, an error message will appear "**The number of service units is not valid**".
5. The program automatically calculates the **estimation time per service** and the **Sub Total** that is placed on the DataGridView
6. The program automatically calculates **Total Pay** and **Total Estimated Time** which is placed at the bottom of the form.

c. The user can **delete** the service transaction that has been entered by the following steps:

1. User selects a **row** in the DataGridView, then pressing the **Remove button**.
2. The program must display a **confirmation window**, if the user presses the **Yes button**, the program will delete the selected service transaction data from database.

3. If the process of deletion process is **successful**, the program must automatically **recalculate Total Pay** and **Total Estimated Time** at the bottom of the form.
- d. When the **Remove button** is clicked while the row in the **service transaction datagridview** is not selected, it will display an error, the program will show a message "**Please, select One Row!**"
- e. Users can enter a **customer data** who makes the transaction by entering a cellphone number in the **Phone Number field**, the **name** and **address fields** will be **automatically filled** in by the data from database. User can **create a new customer** with the steps as in the creation form in "**09 Add Customer Form**" which will be explained below.
- f. User can save transaction of service by pressing the **Submit button**, with the following steps:
 1. The program will check whether the **customer data** has been filled in or not, if it **has not been filled in**, an error window will **appear** with the message "**Please, Fill Customer Section!**"
 2. If all the fields have been filled in, the program will automatically save to the database. A successful window will appear with the message "**Transaction Added!**".

C. Create "08 Input Transaction of Package Form"

Create the main menu of the application as outlined in "**08 Input Transaction of Package Form**" in the wireframe. Below is the specification process needed of this form:

- a. When the **form started**, then:
 1. **Load package data from database** on **package** combobox.
 2. Clear the package transaction data in **datagridview**.
- b. User can add the transaction of packages with the following step:

1. Choose the **package** from combobox.
 2. Enter the desired **Quantity of Package**.
 3. Click the **Add Button**, then the program should update the data to the database.
 4. The system must check whether the **quantity** is more than 1 unit or not, if the total unit is less than 1 unit, an error message will appear "**The quantity of package is not valid**".
 5. The program automatically calculates the **estimation time per service** and the **Sub Total** that is placed on the DataGridView
 6. The program automatically calculates **Total Pay** and **Total Estimated Time** which is placed at the bottom of the form.
- c. The user can **delete** the **package** transaction that has been entered by the following steps:
1. User selects a **row** in the DataGridView, then pressing the **Remove button**.
 2. The program must display a **confirmation window**. if the user presses the **Yes button**, the program will delete the selected service transaction data from database.
 3. If the process of deletion process is **successful**, the program must automatically **recalculate Total Pay** and **Total Estimated Time** at the bottom of the form.
- d. When the **Remove button** is clicked while the row in the **package transaction datagridview** is not selected, it will display an error, the program will show a message "**Please, select One Row!**"
- e. Users can enter a **customer data** who makes the transaction by entering a cellphone number in the **Phone Number field**, the **name** and **address fields** will be **automatically filled** in by the data from database. User can **create a new**

customer with the steps as in the creation form in "**09 Add Customer Form**" which will be explained below.

- f. User can save transaction of package by pressing the **Submit button**, with the following steps:
 1. The program will check whether the **customer data** has been filled in or not, if it **has not been filled in**, an error window will **appear** with the message "**Please, Fill Customer Section!**"
 2. If all the fields have been filled in, the program will automatically save to the database. A successful window will appear with the message "**Transaction Added!**".

D. Create "09 Add Customer Form"

Add customer Form is used to quickly create a **New Customer** without having to open "**05 Customer Form**". "**09 Add Customer Form**" is called when the user wants to enter a new Customer when saving a transaction, (i.e. **step in module 3 , section B, step e and section C, step e**).

Users can save new customer data by following these steps:

- a. User fills in Customer Name, Phone Number and Address data.
- b. The user presses the **submit button**, with the following conditions:
 1. if any field is not filled, the program will automatically give an error message.
 2. If all fields are filled correctly, the program will add customer data to the database, then automatically fill in the **phone number, name, and address** on **Form 07 Input Transaction of Service Form** and **08 Input Transaction of Package Form**.

E. Create “10 View Transaction Form”

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

- a. When user **select the row** on **Header Deposit** datagridview, the Detail Deposit DataGridView will be filled from table Detail Transaction from database.
- b. Users can mark services and packages that have been completed by pressing the **Complete button** on the Detail Deposit DataGridView. Program should update the **detail transaction data** as **completed Service / Package**.

Modul 4: Database Modification

ESEMKA Laundry will continue developing its Information System to support another business process of modern laundry management. To reduce queues, especially regarding health protocols during the new normal. ESEMKA Laundry wants to add a laundry pick-up feature at the customer's house. ESEMKA Laundry also wants to attract customers by giving discounts.

Make modifications to the database so as to allow the laundry to apply Discount.

Regarding the laundry pick-up feature, this feature allows customers to order laundry pick-up services and get notifications when the laundry is finished by installing and using the application on their smartphone.

The scenario to be carried out is as follows:

1. The pickup service is a paid service with a fixed price.
2. The user uses the Mobile application (Module 5) to order a pickup service.
3. A pick-up service employee will get a notification from the desktop application that a customer has requested a laundry pick-up service. Then pick up the laundry according to the address entered by the customer.
4. ESEMKA LAUNDRY process the laundry. When it is finished, there will be a notification in the customer's application.
5. A pick-up service employee will deliver the finished laundry to the customer's address, after it is finished, the user gets a notification.

As an analyst, you are asked for create table and give example data minimum 5 data per table in SQL Server 2019 to support all system that needed in ESEMKA LAUNDRY.

Modul 5: Mobile Application Development

Make an application like the wireframe example above (Form 11 and Form 14)

Detail of this module:

1. Login page that contains user and password
2. Menu page that contains two submenus:
 - a. Request for pick-up, and
 - b. Notifications (Information about the laundry completion status)
3. The request for the pick-up page will display a selection of buttons to set up:
 - a. Customer's name (an auto-complete text field as per the user account information)
 - b. Customer's address
 - c. Laundry package type,
4. The notifications page displays all notifications about the laundry progress.