LKSMart Desktop Application

LKSMart is a proposed self-service desktop application to help customer in order and buy process. The application has **four** main **key** **features** such as: **Manage Profile, Shopping, Transaction History and Point History**. As a well-known **C#** developer, you are hired to develop this application. The analyst team already prepared the database design, and you can find it in **LKSMart.sql** file. You are not allowed to change the structure of the database. However, you may add additional records to the database.

Analyst team also has two additional notes for you. First, all **delete** process is a **soft**-**delete** one. It means for **delete** **process**, instead of deleting the record, you only need to fill in **deleted\_at** field. Second, every time a record is **updated**, make sure to record the time in **last\_updated\_at** field.

The designer also prepared the user interface design for you to use. Please try to stick into the given design. Some changes are allowed under the condition that you maintain all input output process. Some image files also given for you to be used in the application.

Below, you will find detail of the application workflows and functions.

# Login Form

This form will be used by the customer to log in to the application. Customers need to fill in **combination** of their **phone** **number** **or** **email** **and** **PIN** **number**. Make sure their **credentials** **exist** in the **database**. **Please provide all necessary validation**.

Graphical user interface, application

Description automatically generated

Figure 1. Login Form

# Main Form

* After successful login, customer will be redirected to this form.
* Show **customer** **profile** **picture** and **name**. If the customer does not have a profile picture set, show ***default\_profile\_picture*** image instead.
* Show **live** **datetime** information.
* There will be **four** **main** **menus** available as shown in Figure 2. Use images provided for each corresponding menu.
* **Cart** **Image** **Button** will redirect customer to **Cart** **Form**.
* **Logout** **Button** will log out customer from the application.

Shape, polygon

Description automatically generated

Figure 2. Main Form

# Profile Form

* This form shows current **customer** **profile**.
* Customer can click **Edit** **Button** once to **enable** **changing** the corresponding data. Provide **proper** **validation** and **error** **message** for each field if any. Clicking **Edit** **Button** once again will **save** **the** **changes** (if any) to the database.
* **PIN** must be **six** **digit** **numeric**.
* Customer can use **Back** **Button** to return to **Main** **Form**.
* Uploading a new image will immediately change the **profile** **picture**.

Graphical user interface, table

Description automatically generated

Figure 3. Profile Form

# Shop Form

* This form shows **products** sold by LKS Mart.
* Customer can use **search** **bar** to search **product** **name** that **contains** **the** **keyword** they typed in.
* Customer can also fill in **price** **range** **filter** to narrow their search result by price.
* There is a **breadcrumb** feature that display **subsequent** **category** of the **product**. Clicking the **sub-category** will display all product belong to that sub-category.
* Display the product data (**image**, **name**, **price**, **stock**). If the product does not have image, use **not\_available** image instead.
* There is an **Add to Cart Button** beside of each product. Clicking this button will show **Add** **to** **Cart** **Form**.

Graphical user interface, text, application

Description automatically generated

Figure 4. Shop Form

# Add to Cart Form

* This form will be shown if customer:
  + Click **Add** **to** **Cart** **Button** in one of the **product** from **Shop** **Form**.
  + Click **Edit** **Button** in one of the product in the **Cart** **Form**.
* Shows detail of the product (**name**, **image**, **price**, and **description**)
* Customer can use **(+)** and **(-) Button** to increase or decrease the number of product to buy. Make sure to validate that the product is available for purchase.
* When **Button** **Submit** is clicked, update the **cart** **data**.

Shape

Description automatically generated with low confidence

Figure 5. Add to Cart Form

# Cart Form

* This form shows all product data (**image**, **name**, **price**, and **quantity**) added by customer from **Shop** **Form**. If the product does not have image, use **not\_available** image instead. There is also **Edit** **Button** and **Delete** **Button** for each item in the cart.
* When **Edit** **Button** is clicked, **Add** **to** **Cart** Form will be shown.
* When **Delete** **Button** is clicked, a **confirmation** **dialog** will be shown. If the customer confirms yes, then the product will be **removed** from the **cart**.
* **Payment** **Type** data must be taken from database.
* If **Pay Using Available Point** is **chosen**, then **all available point** will be used to **deduct** **amount to pay** **after the tax addition**. Else, **disable** **available** **point** field.
* If customer **chose to not use the point for payment**, he/she will earn **20% of sub-total value as points earned** **rounded** to **nearest** **integer**.
* If the customer clicks **Submit Button, Payment Notification Form** will be shown.
* Customer can use **Back** **Button** to return to **Main** **Page.**

Graphical user interface, text

Description automatically generated with medium confidence

Figure 6. Cart Form

# Payment Notification Form

* This form is shown when customer click **Submit** **Button** in **Cart** **Form**.
* **Payment** **Type** is based on the chosen one in **Cart** **Form.**
* **Payment Code** is a string of **12 randomly generated alphanumeric characters**.
* After customer click **Ok** **Button**, **cart** **data** will be **removed,** and the application will be **redirected** to **Main** **Form**.

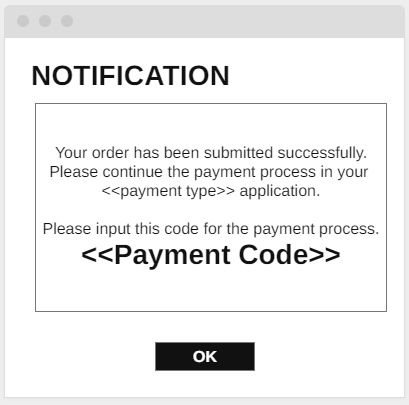


Figure 7. Payment Notification Form

# Transaction History

* This form shows **history** **transaction** **data** (**date**, **total** **payment**, **point** **gained or deducted**, and **payment** **code**) ever made by the customer.
* Selecting one specific transaction will shows all **product** **data** (**image**, **price**, **quantity**) purchased on those transaction. If the product does not have image, use **not\_available** image instead.
* Customer can use **Back** **Button** to return to **Main** **Page.**

Graphical user interface, text, application

Description automatically generated

Figure 8. Transaction History Form

# Point History

* This form shows **history** of **earning** and **deduction** of the customer points (**date**, **payment** **code** if any, **point** **gain (positive or negative)**, **point** **before**, and **point** **after**).
* Give **different** **row** **color** to differentiate between **earning** and **deduction**.
* Customer can use **Back** **Button** to return to **Main** **Page.**

Table

Description automatically generated with medium confidence

Figure 9. Point History Form

-- Good Luck and Do Your Best! --