Main Links

GitHub

https://github.com/meghna-dash/store-management-system

Testing Videos

https://www.loom.com/share/4006b2d145a845829c68952602ca28b8

https://www.loom.com/share/6590eae61c7644859430876143037376

Project 2

Store Management System

User Stories

As a user, I want to add a new product or update a current product in the system.

As a user, I want to add a new customer or update a current customer in the system.

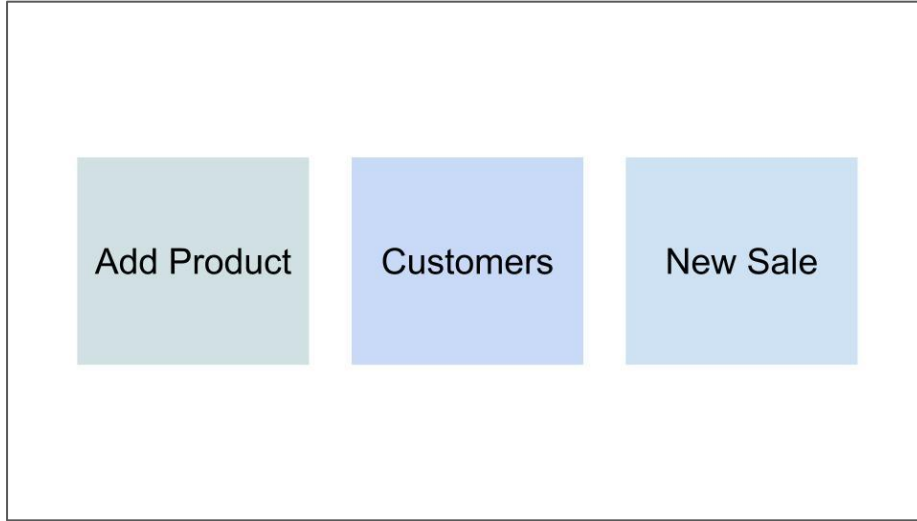As a user, I want to add a new purchase or update a current purchase in the system.

miro

Use Cases

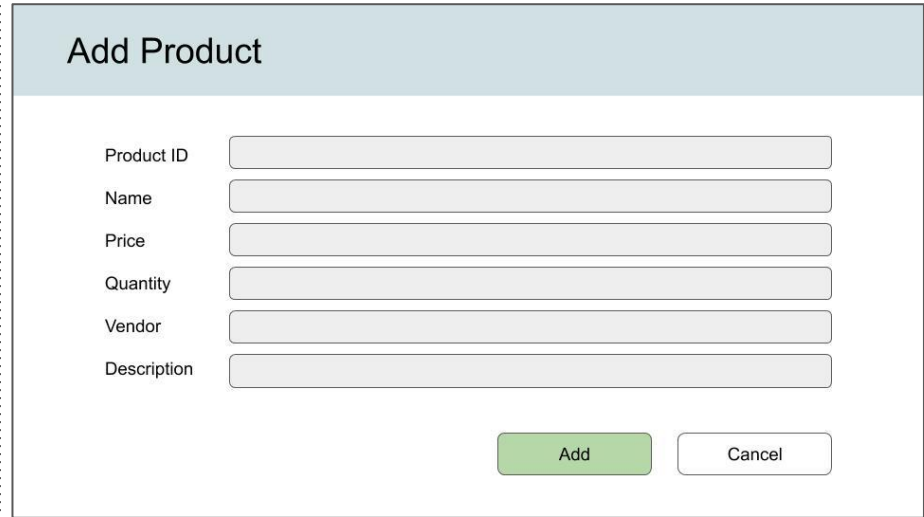| Action	Regular Cases	Irregular Case
Add/Update a Product	All required fields are filled correctly and of correct type.	Values are of the incorrect type or required field (id) is not filled.
Add/Update a Customer	All required fields are filled correctly and of correct type.	Values are of the incorrect type or required field (id) is not filled.
Add/Update a Purchase	All required fields are filled correctly and of correct type.	Values are of the incorrect type or required field (id) is not filled.

UI: Add/Update Product

Design UI and write use case for "add or update a product in the system".

1. **Actor** selects Add Product

Add Product   Customers   New Sale

2. **System** displays Add Product Screen

Add Product

Product ID

Name

Price

Quantity

Vendor

Description

Add   Cancel

3. **Actor** adds information & clicks Add button.

Add Product

Product ID	100
Name	Cap
Price	$10.99
Quantity	30
Vendor	Sporting Goods
Description	Dark blue cap; for men and women.

[Add] [Cancel]

4. **System** hides Add Product screen and shows confirmation.

X

New product added successfully!

5. **Actor** clicks on Okay button

New product added successfully!

Okay

6. **System** closes pop-up and returns to main screen
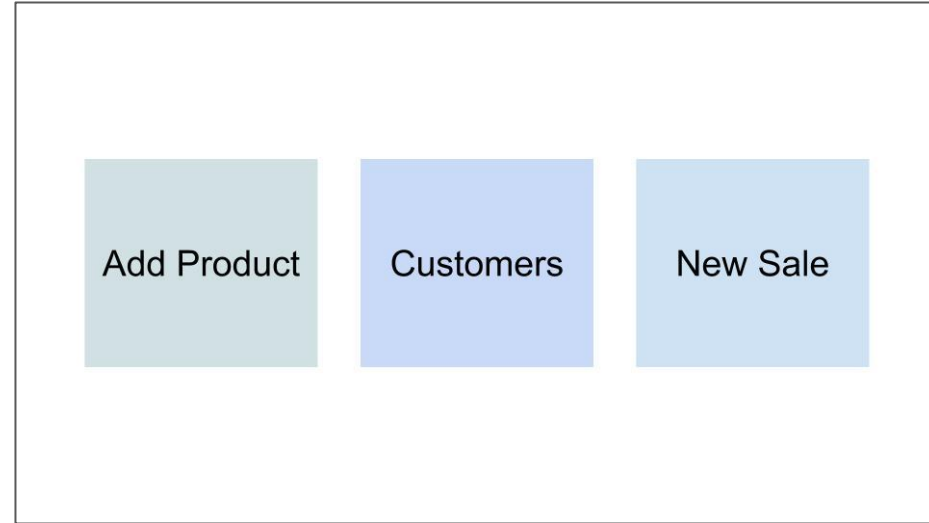
Add Product

Customers

New Sale

Add Product

Customers

New Sale

Add Product

Product ID

Name

Price

Quantity

Vendor

Description

Add

Cancel

Add Product

Product ID   100

Name   Cap

Price $10.99

Quantity   30

Vendor   Sporting Goods

Description   Dark blue cap; for men and women.

Add   Cancel

New product added successfully!
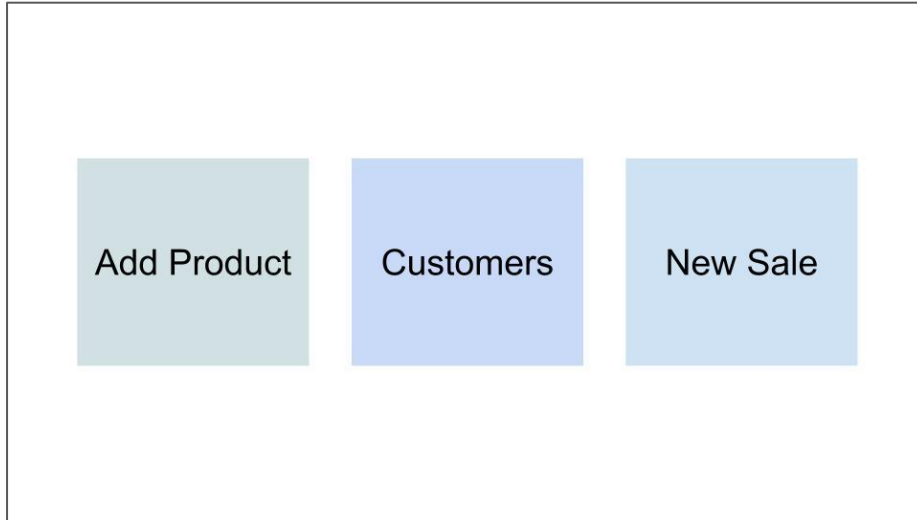
Okay

Add Product

Customers

New Sale

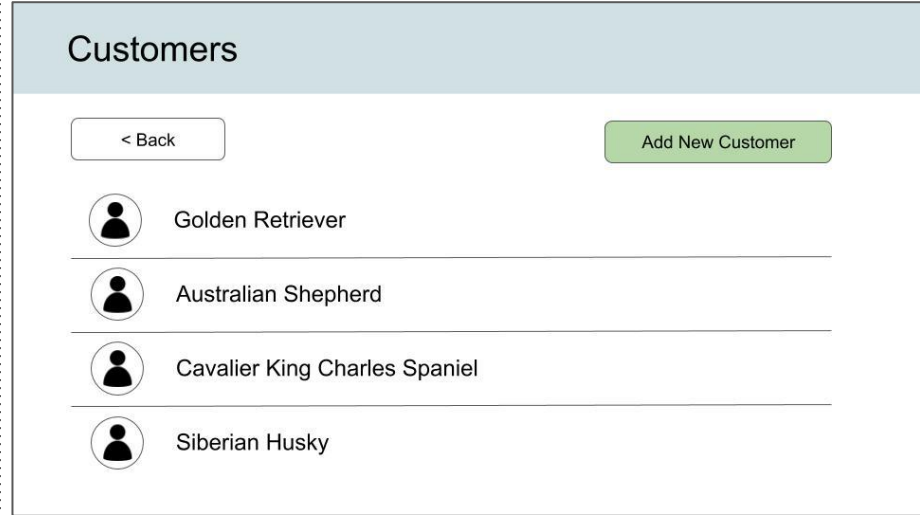# UI: Add/Update a Customer

Design UI and write use case for "add or update a customer in the system".

1. **Actor** selects Customers

Add Product	Customers	New Sale

2. **System** displays list of all customers

Customers

< Back   Add New Customer

👤 Golden Retriever

👤 Australian Shepherd

👤 Cavalier King Charles Spaniel

👤 Siberian Husky

3. **Actor** clicks Add New Customer

Customers
< Back   Add New Customer
👤 Golden Retriever
👤 Australian Shepherd
👤 Cavalier King Charles Spaniel
👤 Siberian Husky

4. **System** shows Add New Customer screen

Add New Customer
Customer ID
Name
Add   Cancel

5. **Actor** adds information & clicks Add button.

Add New Customer

Customer ID	101
Name	Bichon Frise

Add   Cancel

6. **System** hides Add Customer screen and shows confirmation.

New customer added successfully!

Okay

7. **Actor** clicks on Okay button

New customer added successfully!

Okay

8. **System** closes pop-up and shows updated list of customers
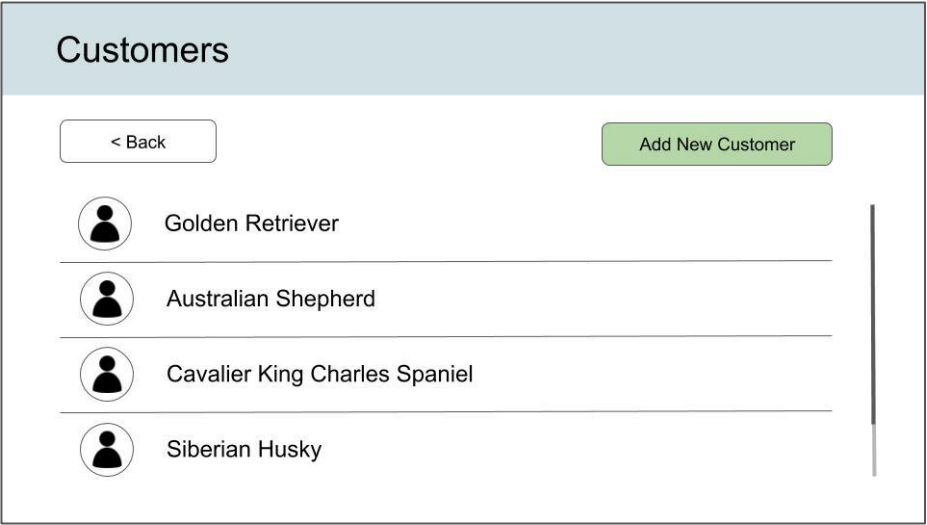
Customers

< Back

Add New Customer

Golden Retriever

Australian Shepherd

Cavalier King Charles Spaniel

Siberian Husky

Add Product

Customers

New Sale

Customers

Add New Customer

👤 Golden Retriever

👤 Australian Shepherd

👤 Cavalier King Charles Spaniel

👤 Siberian Husky

Add New Customer

Customer ID

Name

Add

Cancel

Add New Customer

Customer ID

101

Name

Bichon Frise

Add

Cancel

New customer added successfully!

Okay

Customers

Add New Customer

Golden Retriever

Australian Shepherd

Cavalier King Charles Spaniel
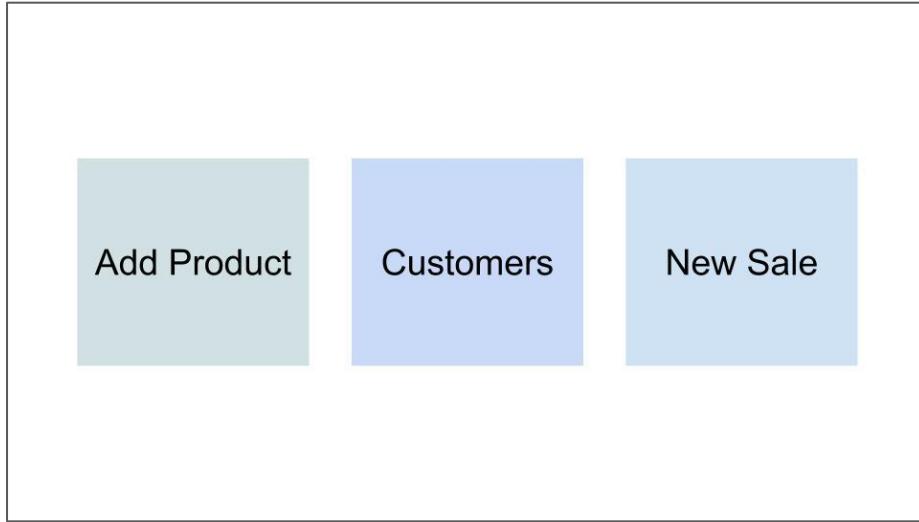
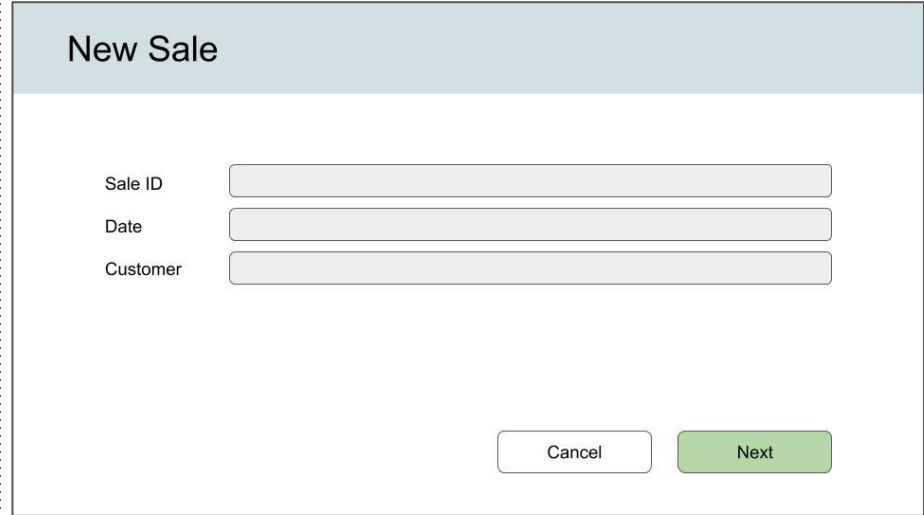Siberian Husky

Add Product

Customers

New Sale

UI: Add/Update a Purchase

Design UI and write use case for "record or edit a purchase from a customer into the system".

1. **Actor** selects New Sale

Add Product	Customers	New Sale

2. **System** displays Add New Sale page

New Sale

Sale ID

Date

Customer

Cancel   Next

3. **Actor** adds basic information & clicks Next button.

New Sale

Sale ID	102
Date	09/14/2019
Customer	Bichon Frise

Cancel   Next

4. **System** shows fields to add items.

New Sale

Items

Item ID		Quantity	1

⊕ Add Another Item

Cancel   Next

5. **Actor** add items in the sale and clicks add.

New Sale

Items

Item ID   112   Quantity   1

Item ID   143   Quantity   2

⊕ Add Another Item

Cancel   Add

6. **System** shows confirmation and hides the New Sale screens

New sale added
successfully!

Okay

7. **Actor** clicks okay

New sale added successfully!

Okay

6. **System** takes the user back to the main screen

Add Product

Customers

New Sale

Add Product

Customers

New Sale

New Sale

Sale ID

Date

Customer

Cancel

Next

New Sale

Sale ID

102

Date

09/14/2019

Customer

Bichon Frise

Cancel

Next

New Sale

Items

Item ID [   ]   Quantity [1]

⊕ Add Another Item

[Cancel] [Next]

New Sale

Items

Item ID | 112 | Quantity | 1

Item ID | 143 | Quantity | 2

⊕ Add Another Item

Cancel   Add

New sale added successfully!

Okay

Add Product

Customers

New Sale

Client Side

The client side needs to load and save data to a remote server component.

I am running the GUI locally. The local GUI makes requests to a local SQL server. I am constantly making requests to the database, starting with the login process. Once I am logged in, the user can only view information specific to their account. As the user navigates the application, the client makes requests to the server for the data it needs to show the user.
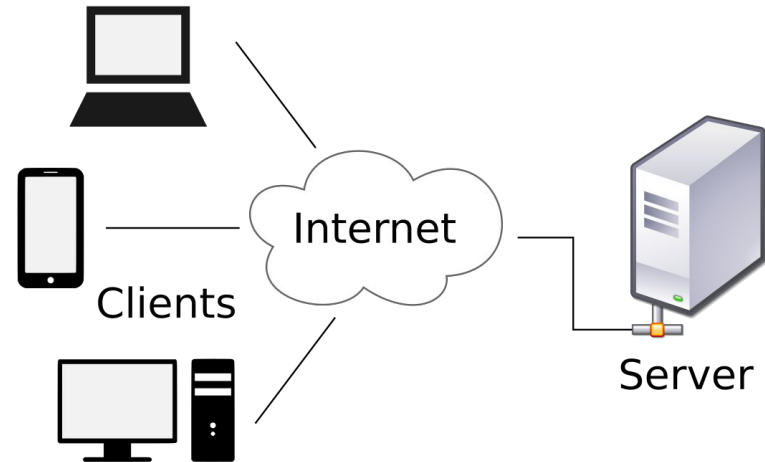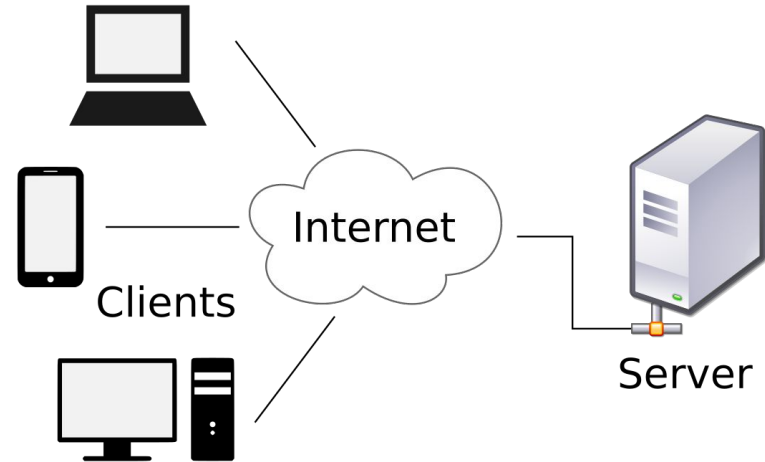
Clients

Internet

Server

Server Side

The client side needs to load and save data to a remote server component.

I am running an instance of a SQLite server locally on my machine. The client makes requests to the server, and the server responds in a way that I can define. For example, if I request to view the purchase history of a specific customer by passing a customerID to the server, the server will return results only for that customer.

Internet

Clients

Server

Implementation

GitHub

https://github.com/meghna-dash

Website

https://meghnadash.design

Testing

<u>User Stories 1 + 2</u>

Manager: Add+Update Products & Customers

https://www.loom.com/share/4006b2d145a845829c68952602ca28b8

<u>User Story 3</u>

Customer: Add+Update Purchase

https://www.loom.com/share/6590eae61c7644859430876143037376

* Video is also submitted on canvas.

Design: User Views

Admins and customers will have different user views.

Admin View vs Customer View

Design the home main screens for users of Admin and Customer classes.

An Admin user can:

- setup the system configuration
- add a new user
- delete an existing user

A Customer user can:

- make a new purchase
- view the list of his/her purchases
- cancel an existing purchase

Use Cases

Actor	Action
Admin	setup the system configuration
Admin	add a new user
Admin	delete an existing user
Customer	make a new purchase
Customer	view the list of his/her purchases
Customer	cancel an existing purchase

Admin Home Page

Set System
Configuration

Add a
New User

Delete an
Existing User

Customer Home Page

Make a New Purchase

View Past Purchases

Cancel a Purchase

Thank you!