README

Let’s Shop

Ver 1.0

Created by	Trac Duc Anh Luong	**Created date**	10/03/2024
Updated by		**Updated date**	

Table of contents

Overview

An e-commerce website to sell electronic products for an existing store in Hanoi.

Important Docs

1. README: This document
2. Entity Design: Entity Design
3. Detail Design: 🗀 Detail Design
4. Testing Checklist: View testing tasks on [ClickUp]()
5. Source code: 🗀 Code

Architecture

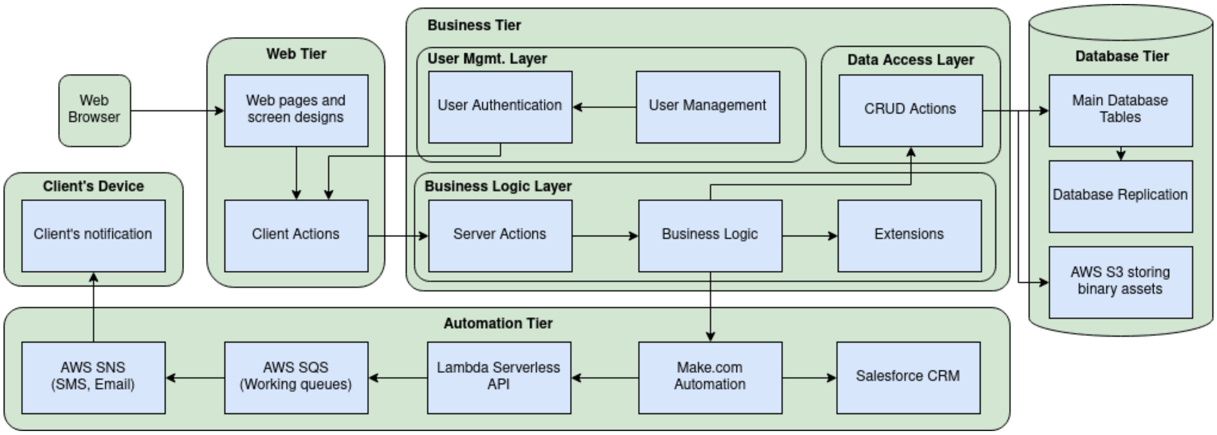


Figure 1: System architecture

Strong cohesion and weak coupling: The block diagram shows a layered architecture that can achieve strong and weak coupling. In a layered architecture, each layer has a specific responsibility and interacts with other layers through well-defined interfaces. This makes the code easier to understand, maintain, and modify.

- **Strong cohesion:** The blocks in the diagram have well-defined responsibilities. For instance, the "Web Tier User Mgmt. Layer" block likely handles user management tasks, while the "Business Logic Layer" block might handle complex application logic.
- **Weak coupling:** The layers seem loosely coupled. This means that a change in one layer, like the "Web Tier," should not require widespread changes in other layers, like the "Business Logic Layer" or the "Data Access Layer."

MVC design pattern: The architecture uses the MVC (Model-View-Controller) design pattern. The Model is the Database Tier, the View is the Web Tier, and the Controller is the Business Logic Tier.

Sitemap

Customer Flow

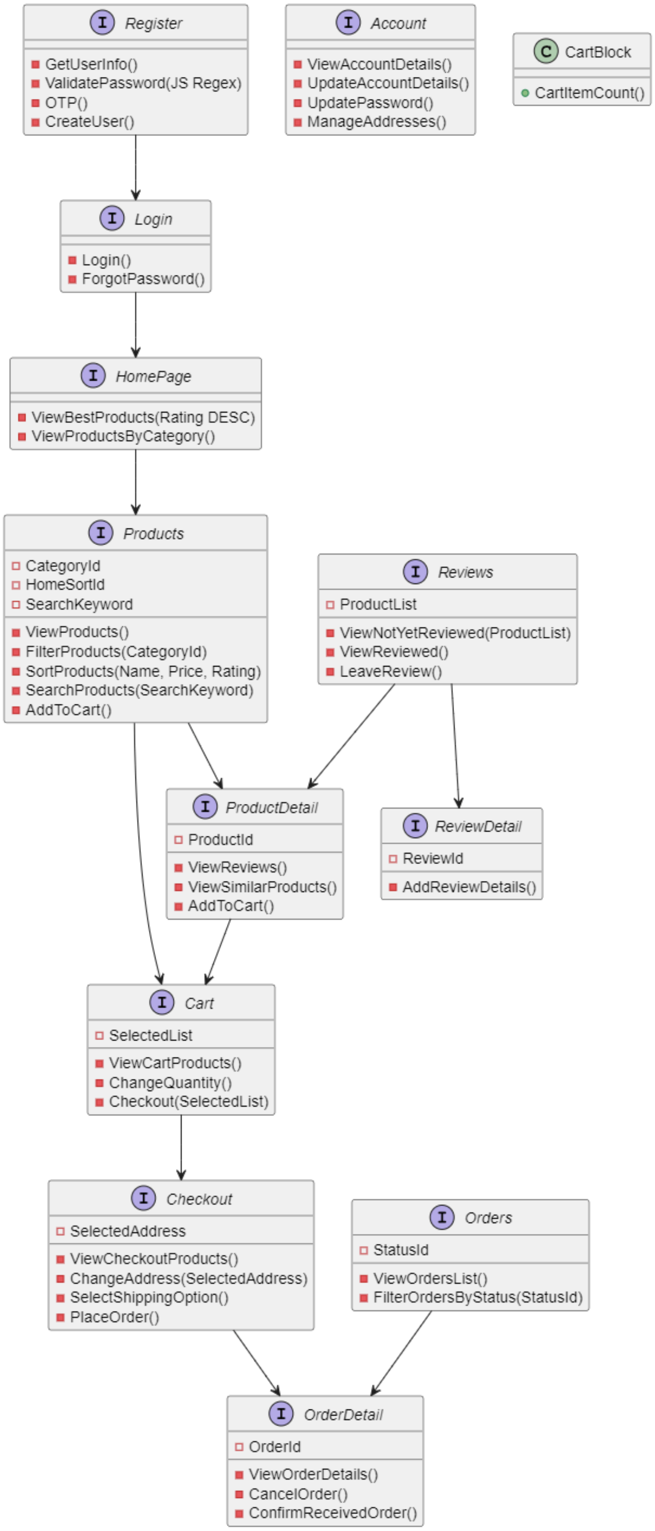


Figure 2: Customer Flow

Admin Flow

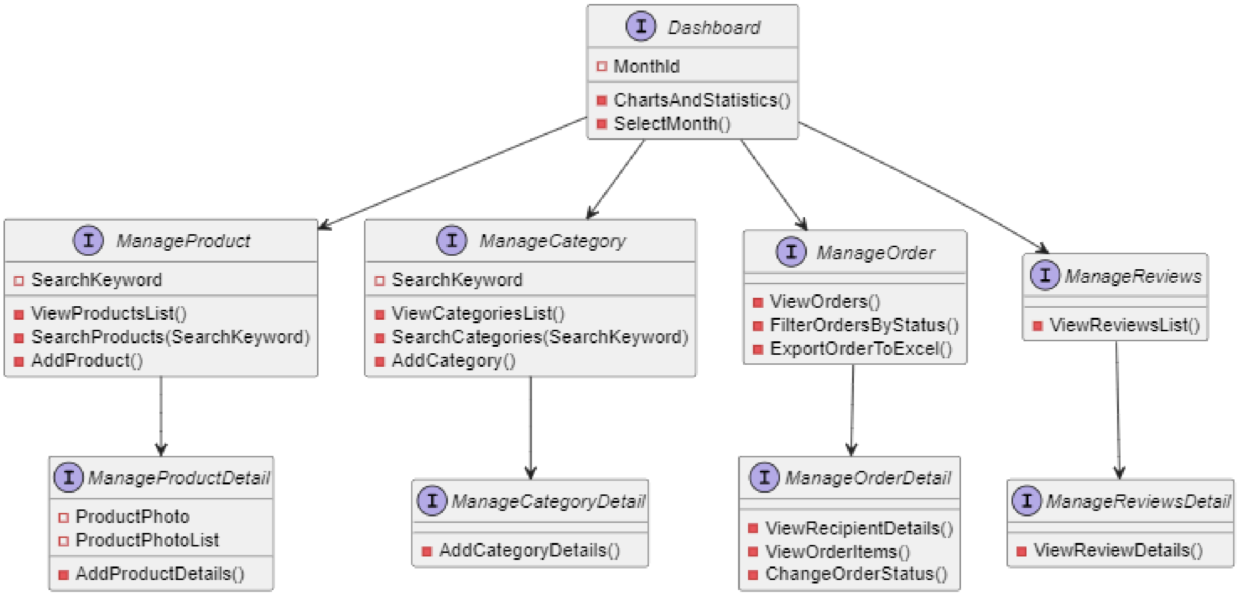
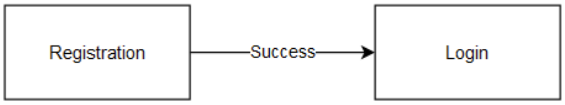


Figure 3: Admin Flow

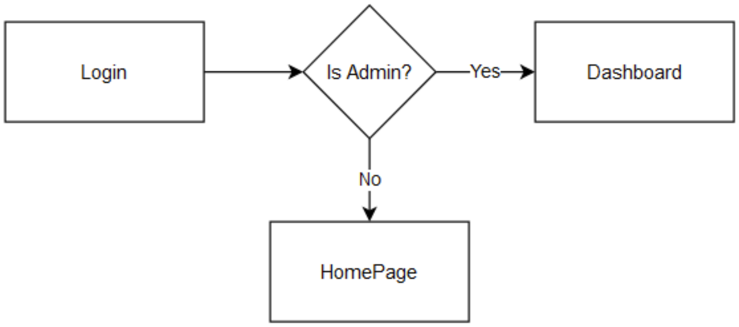
Screens Overview

1. Registration



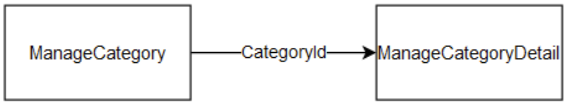
Main Functions		1. Get and validate new customers' information 2. Send OTP to customers' emails to authenticate 3. Redirects to Login screen
Evaluation	Complexity	Hard
	Reasoning	1. The validations must catch all edge cases 2. The displayed timer to count down before the OTP expires can only be implemented with custom JavaScript

2. Login



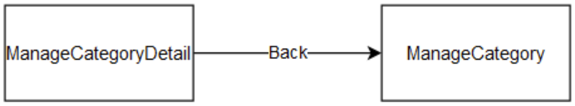
Main Functions		1. Login to the website if users enter the correct credentials 2. Redirects to the HomePage or Dashboard, depending on the users' roles 3. Send a new password to the users' email if they forgot their password
Evaluation	Complexity	Medium
	Reasoning	1. Important entry point of the system 2. Need to set cookies for the browser's session

3. ManageCategory



Main Functions		1. Display a list of categories 2. Open the details of a category when users click on a record
Evaluation	Complexity	Easy
	Reasoning	Can be created using OutSystems' scaffolding functionality

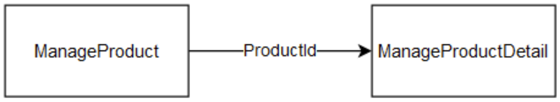
4. ManageCategoryDetail



Main Functions	1. Display the details of a category on a form 2. Users can edit and save new details of the selected category

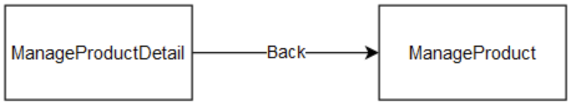
Evaluation	Complexity	Easy
	Reasoning	1. Can be created using OutSystems' scaffolding functionality 2. Add duplicate checking

5. ManageProduct



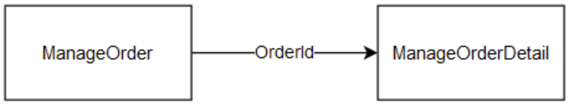
Main Functions		1. Display a list of products 2. Open the details of a product when users click on a record
Evaluation	Complexity	Easy
	Reasoning	Can be created using OutSystems' scaffolding functionality

6. ManageProductDetail



Main Functions		1. Display the details of a category on a form 2. Users can edit and save new details of the selected category 3. Add a feature for multiple images and thumbnail selection
Evaluation	Complexity	Medium
	Reasoning	1. Can be created using OutSystems' scaffolding functionality 2. Add duplicate checking 3. List of binary data for images

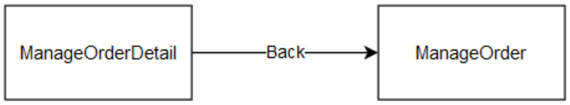
7. ManageOrder



Main Functions	1. Display the list of orders 2. Display count of orders by status 3. Filter orders by status 4. Export order to Excel by month 5. Open the details of an order when users click on a record

Evaluation	Complexity	Hard
	Reasoning	1. Exporting Excel must be precise with all cells 2. Important feature that is difficult to implement

8. ManageOrderDetail



Main Functions		1. Display the details of an order on a form and the customer’s details on another form 2. Update order’s status
Evaluation	Complexity	Medium
	Reasoning	1. Get user details accordingly to order id on after fetch 2. Implement the Wizard/Navigation for order status progress display