



# Software design specification document

## 2023 - 2024

### Project Team

ID	Name	Email
20211042	Reem Waleed	reemw971@gmail.com
20210165	Saif El-Din Hazem	saifehazemb@gmail.com
20210331	Muhammed Ehab	mflashcr@gmail.com
20210208	Abdelrahman Ezzat	abdelrhmanakeel@gmail.com



# SDS Document

## Contents

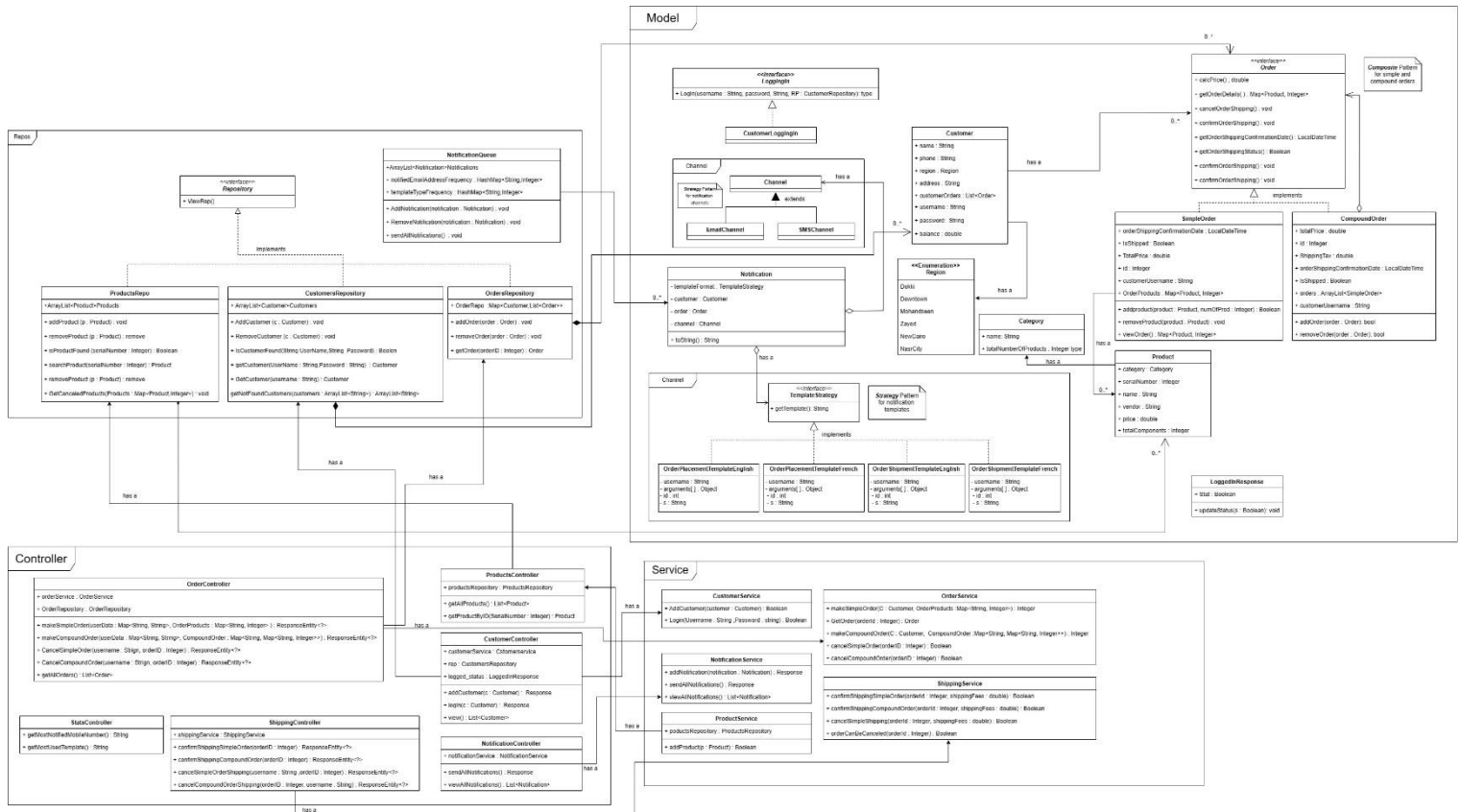
Class diagram design .....	3
Class diagram Explanation .....	4
Requirements Exposure as Web Service API .....	5
Requirement .....	6
Exposed API .....	6



## CS352: Sprint SDS – Dream Team – Order & Notification Module

# SDS Document

## Class diagram design



Class Diagram [Link](#)



## SDS Document

### Class diagram Explanation

- **Strategy pattern** is used twice in the following:

1. Choosing the **Template Type** that the customer wants to use and the
2. Choosing the **channel** which will be used to send the notification later.

We used this pattern to adhere to the Open Closed Principle which states that the code should be open for extensions and closed for modifications, accordingly we can easily add new templates without modifying the existing code.

- **Composite Pattern** is used to handle the compound orders as compound orders could consist of either a **list of simple orders** or **list of Compound orders**.

So, this pattern is used to allow dealing with both **Simple** orders and **Compound** orders through a common interface, forming a tree structure.



## SDS Document

### Requirements Exposure as Web Service API

#### Part 1: Exposed Postman Collection

Here's the [link](#) of the collection to be used in testing, First run the code then the given collection.



## SDS Document

### Part 2:

Requirement	Exposed API
Create Account	<p><b>Method:</b> POST  <b>URL:</b> <a href="http://localhost:8080/api/customer/create">http://localhost:8080/api/customer/create</a></p> <p>{ "name": "Name Example", "phonenumber": "0150000085", "region": 1, "address": "1 Example, State, Country", "username": "username", "password": "password", "balance": 1000000 }</p> <p><b>Description:</b> This is a POST request for creating a customer account. A successful request typically returns a 200 OK or 201 Created response code.</p>
Login	<p><b>Method:</b> POST  <b>URL:</b> <a href="http://localhost:8080/api/customer/login">http://localhost:8080/api/customer/login</a></p> <p>{ "username": "username", "password": "password" }</p> <p><b>Description:</b> This is a POST request for customer login. A successful request typically returns a 200 OK or 201 Created response code.</p>
View all customers	<p><b>Method:</b> GET  <b>URL:</b> <a href="http://localhost:8080/api/customer/view">http://localhost:8080/api/customer/view</a></p>
View all products	<p><b>Method:</b> GET  <b>URL:</b> <a href="http://localhost:8080/api/products/all">http://localhost:8080/api/products/all</a>  <b>Description:</b> This is a GET request to view all system products</p>



## SDS Document

View single product by ID	<b>Method:</b> GET <b>URL:</b> <a href="http://localhost:8080/api/products/{orderId}">http://localhost:8080/api/products/{orderId}</a> <b>Description:</b> This is a GET request to view a specific system product by ID
View all orders	<b>Method:</b> GET <b>URL:</b> <a href="http://localhost:8080/api/order/view">http://localhost:8080/api/order/view</a> <b>Description:</b> This is a GET request to view all system orders
Make simple order	<b>Method:</b> POST <b>URL:</b> <a href="http://localhost:8080/api/order/make/simple">http://localhost:8080/api/order/make/simple</a>  { "Banana": 1, "Watermelon": 2, "Apple": 1 }  <b>Description:</b> This is a POST request for making a simple order. A successful request typically returns a 200 OK or 201 Created response code.
Cancel simple order	<b>Method:</b> POST <b>URL:</b> <a href="http://localhost:8080/api/order/cancel/simple/{username}/{orderId}">http://localhost:8080/api/order/cancel/simple/{username}/{orderId}</a> <b>Description:</b> This is a POST request for canceling a simple order. A successful request typically returns a 200 OK or 201 Created response code.
Make compound order	<b>Method:</b> POST <b>URL:</b> <a href="http://localhost:8080/api/order/make/compound">http://localhost:8080/api/order/make/compound</a>  { "username": { "Product Name": Quantity } }  <b>Description:</b> This is a POST request for making a compound order. A successful request typically returns a 200 OK or 201 Created response code.
Cancel compound order	<b>Method:</b> POST <b>URL:</b> <a href="http://localhost:8080/api/order/cancel/compound/{username}/{orderId}">http://localhost:8080/api/order/cancel/compound/{username}/{orderId}</a> <b>Description:</b> This is a POST request for canceling a compound order. A successful request typically returns a 200 OK or 201 Created response code.
Confirm simple shipping	<b>Method:</b> POST <b>URL:</b> <a href="http://localhost:8080/api/shipping/confirmSimple/{orderId}">http://localhost:8080/api/shipping/confirmSimple/{orderId}</a>



## SDS Document

Cancel simple shipping	<b>Method:</b> POST <b>URL:</b> <a href="http://localhost:8080/api/shipping/cancel/simple/{username}/{orderId}">http://localhost:8080/api/shipping/cancel/simple/{username}/{orderId}</a> <b>Description:</b> This is a POST request for canceling simple shipping. A successful request typically returns a 200 OK or 201 Created response code
Confirm compound shipping	<b>Method:</b> POST <b>URL:</b> <a href="http://localhost:8080/api/shipping/confirmCompound/{orderId}">http://localhost:8080/api/shipping/confirmCompound/{orderId}</a> <b>Description:</b> This is a POST request for confirming compound shipping. A successful request typically returns a 200 OK or 201 Created response code
Cancel compound shipping	<b>Method:</b> POST <b>URL:</b> <a href="http://localhost:8080/api/shipping/cancel/compound/{username}/{orderId}">http://localhost:8080/api/shipping/cancel/compound/{username}/{orderId}</a> <b>Description:</b> This is a POST request for canceling compound shipping. A successful request typically returns a 200 OK or 201 Created response code.
Send Notification	<b>Method:</b> GET <b>URL:</b> <a href="http://localhost:8080/api/notification/send">http://localhost:8080/api/notification/send</a> <b>Description:</b> This is a GET request to send notification to customers and empty the queue.
View notification	<b>URL:</b> <a href="http://localhost:8080/api/notification/view">http://localhost:8080/api/notification/view</a> <b>Method:</b> GET <b>Description:</b> This is a GET request to view system notifications.
Phone statistics	<b>Method:</b> GET <b>URL:</b> <a href="http://localhost:8080/api/stats/phone">http://localhost:8080/api/stats/phone</a> <b>Description:</b> This is a GET request to get the most notified phone-number.
Template statistics	<b>Method:</b> GET <b>URL:</b> <a href="http://localhost:8080/api/stats/template">http://localhost:8080/api/stats/template</a> <b>Description:</b> This is a GET request to get the most used template.

You can also see these URLs mapping in swagger using this link while code is running.

→ <http://localhost:8080/swagger-ui/index.html#/>