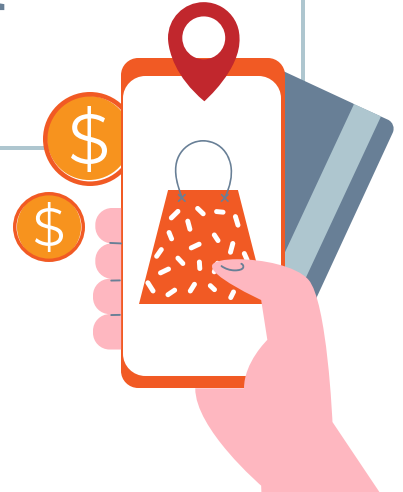




# Shop Sphere - Online Shopping System

Here is where your presentation begins





## Team Member :

1-Alaa ALI

2-Somaya Magdy

3-Asmaa Refaat


4-Abdulrahman Al-Maghraby





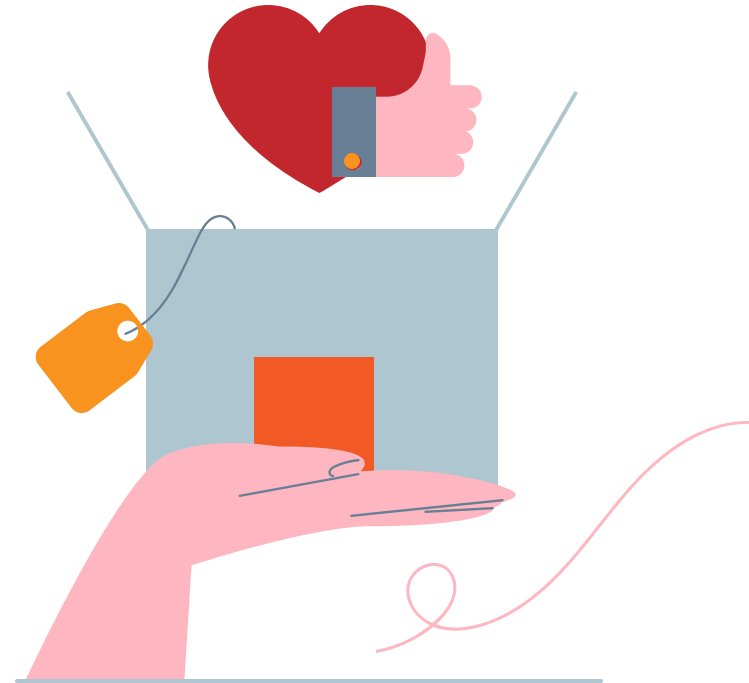
## The Role of Web Technologies in Modern Life:

The web plays a vital role in modern life by enabling instant access to information, global communication, online education, business transactions, and e-commerce. It connects people and services worldwide, making everyday tasks faster, easier, and more efficient. Web technologies are now essential for both personal use and business success.



# About us

***Shop Sphere*** is a project aimed at providing a simple and secure online shopping experience through a user-friendly interface and modern technologies. We focus on customer satisfaction by offering a wide range of products at competitive prices, with fast delivery and excellent support.



# Comparison Between Online and Offline Shopping:

01

Convenience

02

Product Variety

05

Delivery

03

User Experience

04

Prices & Offers

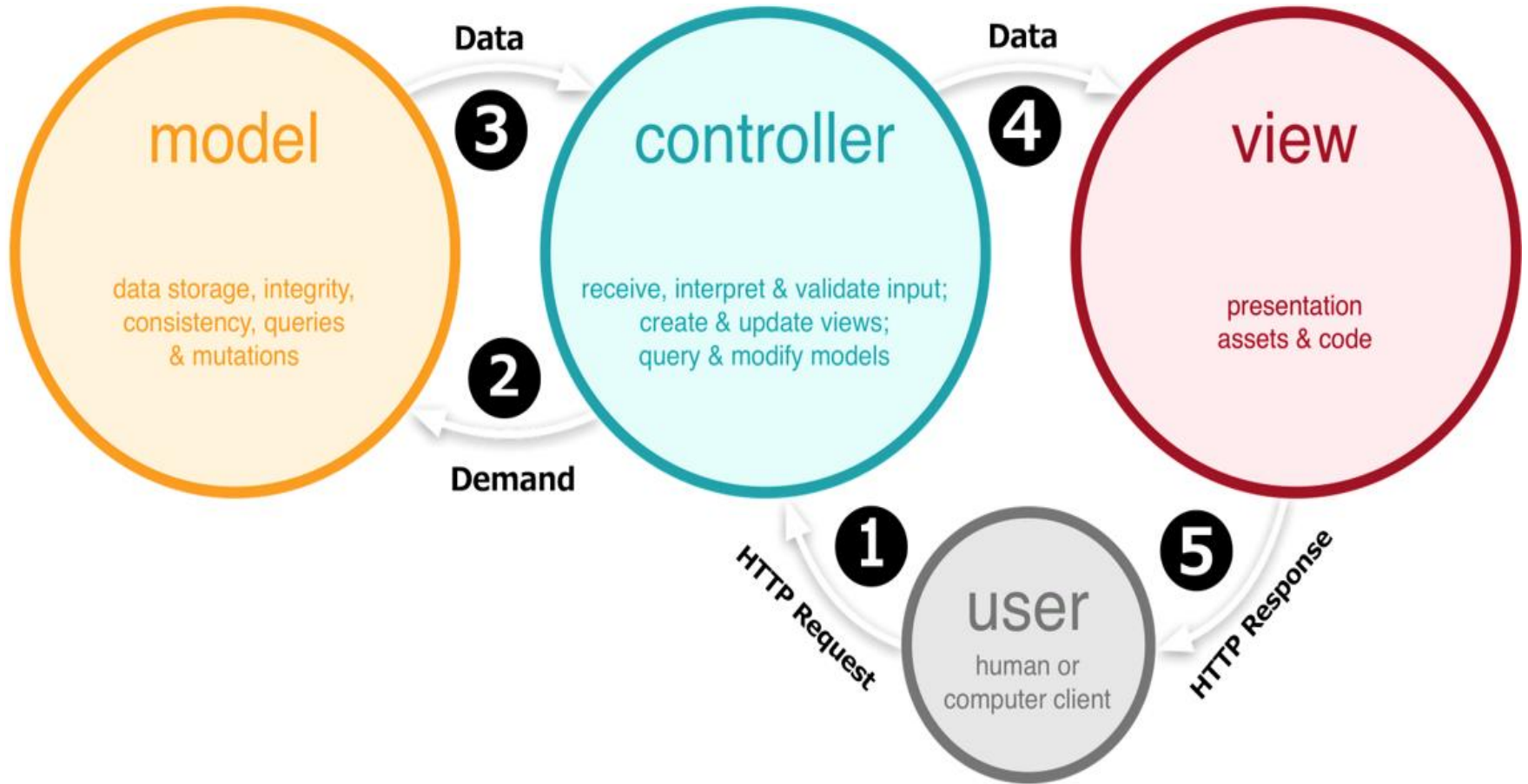


# Separation of concerns

Model (Data)

View (HTML)

Controller (Processing)

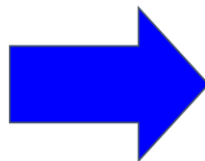


# Data Access Layer

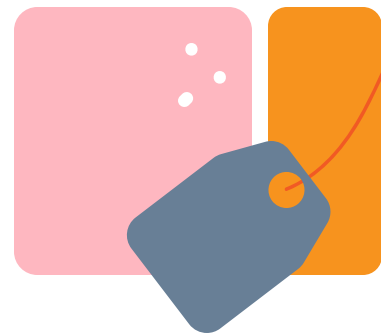
Model

DbContext

Migrations



DataBase





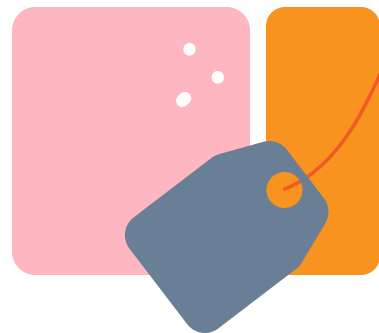
# Presentation Layer

MVC

WEb API

Desktop

Mobile



# Product Module

Core

## Product

Int Id

string Name

string Description

string PictureUrl

decimal Price

## ProductBrand

Int Id

string Name

## ProductType

Int Id

string Name

# Product Module

## Presentation

0 references

```
public class ProductsController : APIBaseController
{
    // Get All Products

    // Get Product By Id
}
```

**1. View**

**2. Partial View**

**3. \_Layout View**

**4. \_ViewImports**

# Security Module

## Authentication



Confirms users  
are who they say they are.

## Authorization



Gives users permission  
to access a resource.

# Identity Module

Core

## Address

Int Id

String FirstName

String LastName

String Street

String City

String Country

## AppUser

String DisplayName

# Identity Module

controller

## Register EndPoint

Email

DisplayName

PhoneNumber

Password



Email

DisplayName

Token

# Identity Module

Controller

Login EndPoint

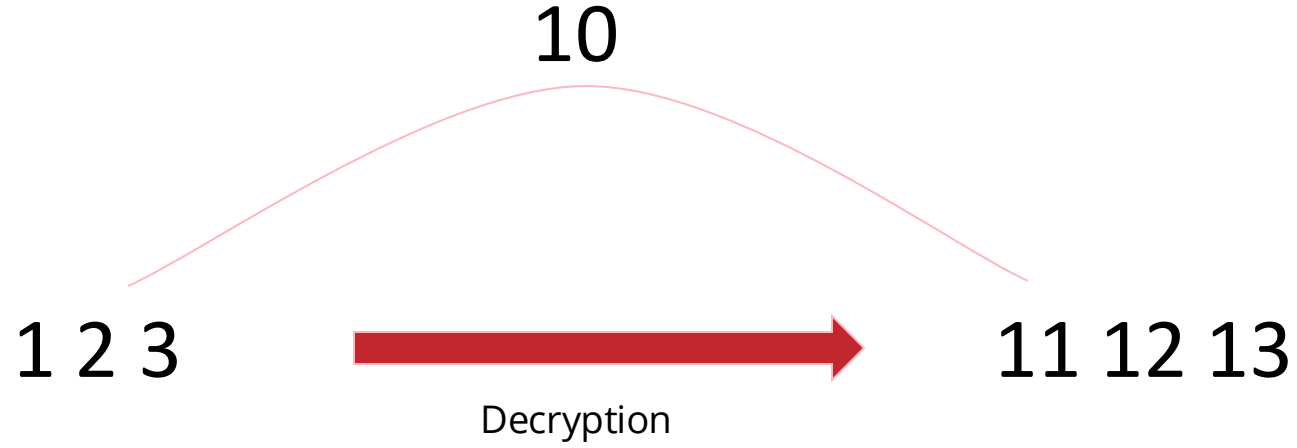
Email  
Password



Email  
DisplayName  
Token

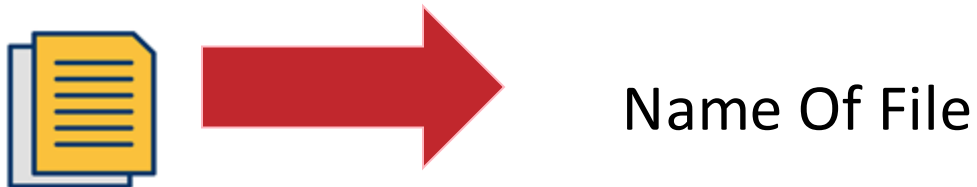
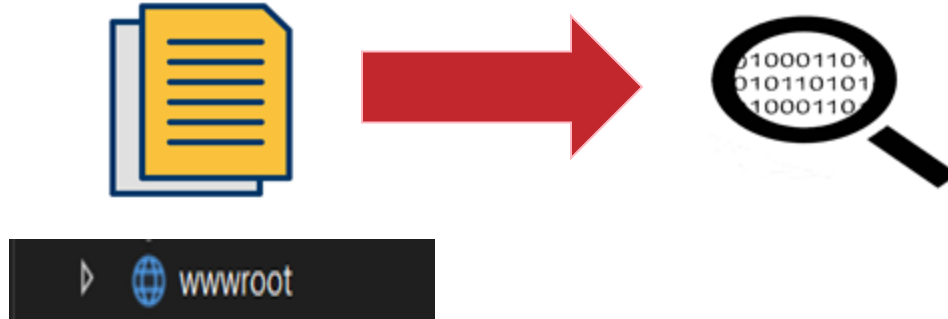


# Token



# Document Settings

## 1. File To Binary





## Shopping Cart



Sceptre 24-inch Professional Thin 1080p LED Monitor 99% sRGB 2x HDMI VGA Build-in Speakers, Machine Black (E248W-19203R Series)

**\$89.97**

**#1 Best Seller** in Computer Monitors

In Stock

☐ This is a gift [Learn more](#)

Style: 24" 75Hz

Qty: 1 ▼

[Delete](#)

[Save for later](#)

[Compare with similar items](#)

[Share](#)

Subtotal (1 item): **\$89.97**

# Redis

The open source, in-memory data store used by millions of developers as a database, cache, streaming engine, and message broker.

# Basket Module

Core

## Basket Item

Int Id

String Name

String PictureUrl

String Brand

String Type

Decimal Price

Int Quantity

## Customer Basket

String Id => GUID

List <BasketItem> Items

# Basket Module

## Functions :

1.Get Basket

1.Update Basket [Create Or Update]

1.Delete Basket

# Order Module

Core

Order

Components Of Object

Complex Object

Address

Delivery Method

Order Status

OrderItem

Enum

- String : FirstName
- String : LastName
- String : Street
- String : City
- String : Country

- Int : Id
- String : ShortName
- String : Description
- String : DeliveryTime
- Decimal :Cost

- Pending
- Payment Received
- Payment Failed

- Int: Id
- Int : ProductId
- String : ProductName
- String : PictureUrl
- Decimal : Price
- Int : Quantity

# Order Service

Interface

Implementation

Core

Service

## Functions :

1. Create Order [BuyerEmail , BasketId , Delivery Method Id , Shipping Address]
2. Get Orders For Specific User [BuyerEmail]
3. Get Order By Id For Specific User [BuyerEmail , Order Id]



# Unit Of Work

Core

Interface

Repository

Implementation

## MVC Unit Of Work

```
public interface IUnitOfWork : IDisposable
{
    3 references
    public IGenericRepository<Product> ProductRepo { get; set; }
    2 references
    public IGenericRepository<ProductBrand> ProductBrandRepo { get; set; }
    2 references
    public IGenericRepository<ProductType> ProductTypeRepo { get; set; }
    2 references
    public IGenericRepository<Order> OrderRepo { get; set; }
    2 references
    public IGenericRepository<OrderItem> OrderItemRepo { get; set; }
    1 reference
    public IGenericRepository<DeliveryMethod> DeliveryMethodRepo { get; set; }
    1 reference
    Task<int> CompleteAsync();
}
```

# Payment Module

**We Have 2 Approaches To Implement Payment Module**

1. Create Payment Gateway
2. Integrate With Existing Payment Gateway

## 1. Create Payment Gateway

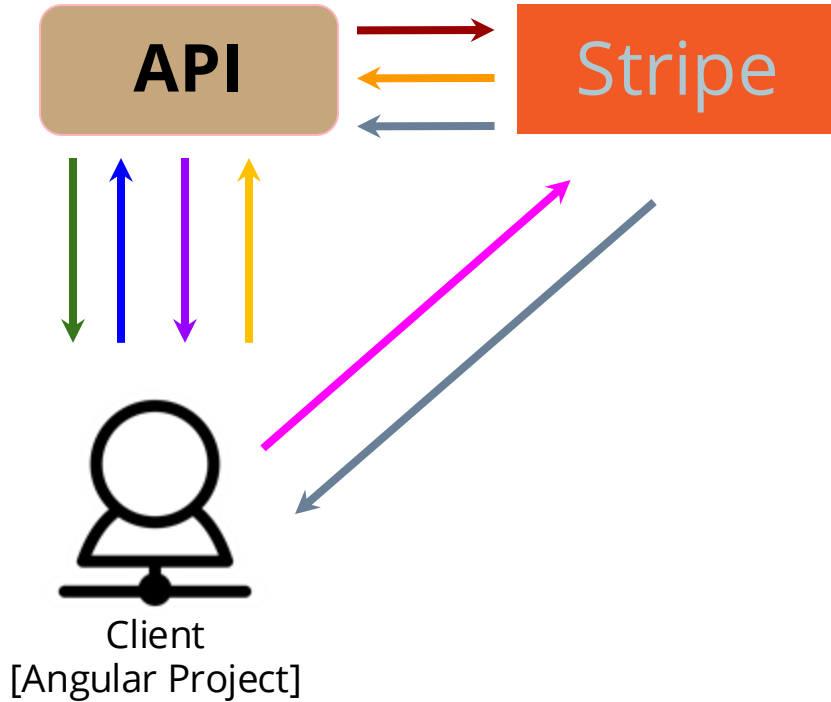
You Will Need a Network Infrastructure with the appropriate firewalls and high security to protect all of the infrastructure itself.

## 2. Integrate With Existing Payment Gateway

1. Stripe
2. PayMob
3. PayFort
4. PayPal
5. E-Finance

# PaymentModule

Stripe



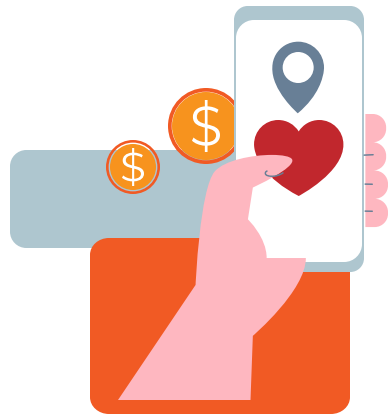
# Mapping

**Model**   =>   Class Represent Table in Database

**ViewModel**   =>   Class That Rendered In View

A picture is worth a  
thousands words





# Thanks!

