**NCFE Level 3 Coding Practices**

CSS Pizzas {Pizzas with style}

**Shopping Cart Website Development**

[Introduction 2](#_Toc2099663124)

[Design Documents 3](#_Toc1564534455)

[Use Case 4](#_Toc2014578911)

[Data Dictionary 4](#_Toc1421452682)

[Entity Relationship Document (ERD) 5](#_Toc1246380876)

[Database Creation 6](#_Toc804143101)

[Wireframes 7](#_Toc443302571)

[Website 7](#_Toc1938461094)

[Technologies used 8](#_Toc266692000)

[Summary and conclusions 9](#_Toc1172927763)

[Appendices 9](#_Toc1965213186)

[1.1 - Shopping Cart Data Dictionary 10](#_Toc850869509)

[1.2 - Table creation SQL statements 12](#_Toc541091805)

[1.4 - Payment and Delivery Use Case diagram 17](#_Toc1603133483)

[1.5 - Wireframe (V1.0) 17](#_Toc1265425282)

Introduction

CSS Pizzas commissioned a proof-of-concept web site to provide its customers with the ability to place online orders for pizza delivery from any of its network of national stores.

As this a proof of concept only, the payment and delivery of orders is beyond the scope of this project.

This document details the stages of the software development lifecycle from high level concept to implementation of the working prototype.

The solution should adhere to the following requirements/constraints;

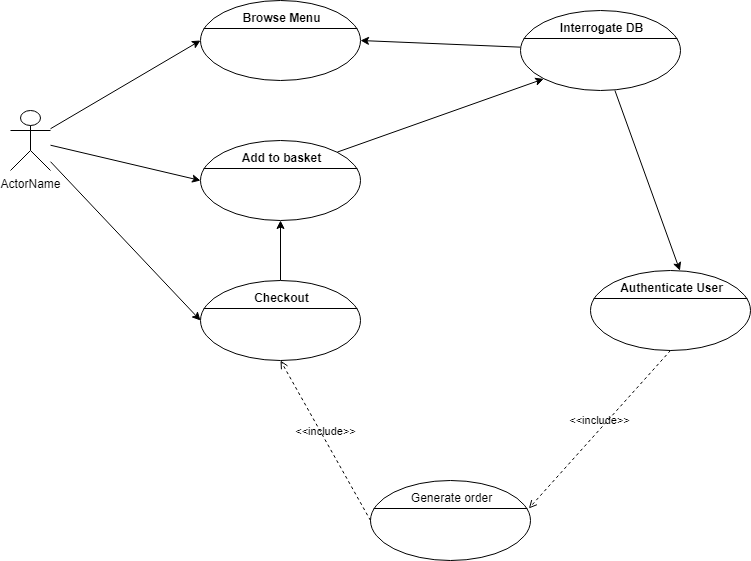
* The same products are available at every store.
* Each order may contain one or more products.
* Each order is associated with only one store.
* The status of a live order should be recorded (order received/order being prepared/order out for delivery).
* The system should retain details of previous customer orders.

# Design Documents

## Use Case

Below is a use case diagram which describes the high-level overview of the CSS Pizza ordering system. This diagram was developed in collaboration with the client to define the required interactions between the actor (customer) and the system.

As seen in the diagram, the customer should have the ability to browse the menu, add menu items to the basket or shopping cart and finally place their order using the checkout facility. The customer-facing functions interact with the back-end functions (interrogate database, authenticate user and generate order).



The ‘Generate order’ function includes the functionality to implement Payment and Delivery activities, which as previously mentioned, is outside the scope of this project. However, a Use Case diagram for this has been included for completeness and can be seen in Appendix item 1.x.

## Data Dictionary

Using the output of the previous stage (Use Case) and following further analysis of the requirements, a data dictionary was created to further detail the relevant entities and their properties.

The following core entities were identified;

* Store
* Customer
* Product

The transactional entities which form the concept of the customer order are;

* Shopping Cart
* Shopping Cart Details

\* Please see appendix item 1.1 for the full data dictionary.

## Entity Relationship Document (ERD)

Below is the ERD for the shopping cart application which shows the relationships between the various entities defined in the previous Data Dictionary creation phase.



This diagram was created using the VS Code extension ‘ERD Editor’ as it allows the ability to enter the metadata about each table. This extension also allows the ERD to be exported to SQL DDL in the appropriate database format.

\*Appendix item 1.2 - ‘Table creation SQL statements’ shows the exported SQL DDL.

## Database Creation

A MariaDB implementation of MySQL was used as the datastore for this application, which was provided by a local installation of XAMPP.

The database was created manually via the PHPMyAdmin console.

The database tables were created by executing the SQL DDL generated previously (see appendix 1.x) in the SQL QUERY pane of PHPMyAdmin.

\* Appendix item 1.3 shows a screenshot of the database once created in the Designer pane of PHPMyAdmin console.

## Wireframes

The design of the website interface went through several iterations following feedback from the client.

The initial proposal consists of 4 pages as follows;

* Home page
* Sign Up page
* Sign In page
* View basket/check out page

See ‘Appendix 1.5 - Wireframe (V1.0)’ for images of the layout and functionality.

# Website

## Technologies used

HTML

PHP

CSS

SQL

# Summary and conclusions

Some text

# Appendices

## 1.1 - Shopping Cart Data Dictionary

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Table Name:** | **Store** |  |  |  |  |  |
| ***Key*** | ***Name*** | ***Type*** | ***Length*** | ***Null*** | ***Auto Increment*** | ***Default Value*** |
| Primary | storeID | int | 3 | No | Yes |  |
|  | storeName | varchar | 20 | No |  |  |
|  | storeAddress | varchar | 50 | No |  |  |
|  | storeEmail | varchar | 20 | No |  |  |
|  | status | int | 1 | No |  | 1 |
| Notes: |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
| **Table Name:** | **Customer** |  |  |  |  |  |
| ***Key*** | ***Name*** | ***Type*** | ***Length*** | ***Null*** | ***Auto Increment*** | ***Default Value*** |
| Primary | customerID | int | 3 | No | Yes |  |
|  | firstName | varchar | 20 | No |  |  |
|  | secondName | varchar | 20 | Yes |  |  |
|  | surname | varchar | 20 | No |  |  |
|  | houseNumName | varchar | 20 | No |  |  |
|  | addressTwo | varchar | 20 | No |  |  |
|  | townCity | varchar | 20 | No |  |  |
|  | postCode | varchar | 20 | No |  |  |
|  | emailAddress | tinytext |  | No |  |  |
|  | pwd | tinytext |  | No |  |  |
|  | status | int | 1 | No |  | 1 |
| Notes: |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
| **Table Name:** | **Product** |  |  |  |  |  |
| ***Key*** | ***Name*** | ***Type*** | ***Length*** | ***Null*** | ***Auto Increment*** | ***Default Value*** |
| Primary | productID | int | 3 | No | Yes |  |
|  | productName | varchar | 20 | No |  |  |
|  | productDesc | varchar | 20 | No |  |  |
|  | productPrice | int | 5 | No |  |  |
|  | productQty | int | 3 | No |  |  |
|  | status | int | 1 | No |  | 1 |
| Notes: |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
| **Table Name:** | **Shopping\_Cart** |  |  |  |  |  |
| ***Key*** | ***Name*** | ***Type*** | ***Length*** | ***Null*** | ***Auto Increment*** | ***Default Value*** |
| Primary | shopCartID | int | 5 | No | Yes |  |
| Foreign | storeID | int | 3 | No |  |  |
| Foreign | customerID | int | 3 | No |  |  |
|  | date | date |  | No |  |  |
|  | orderStatus | int | 1 | No |  |  |
|  | status | int | 1 | No |  | 1 |
| Notes: | Foreign key storeID reference storeID in the Store Table |  |  |  |  |  |
|  | Foreign key customerID reference customerID in the CustomerTable |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
| **Table Name:** | **Shopping\_Cart\_Details** |  |  |  |  |  |
| ***Key*** | ***Name*** | ***Type*** | ***Length*** | ***Null*** | ***Auto Increment*** | ***Default Value*** |
| Primary | shopCartDetID | int | 5 | No | Yes |  |
| Foreign | shopCartID | int | 5 | No |  |  |
| Foreign | productID | int | 3 | No |  |  |
|  | prodAddedQty | int | 3 | No |  |  |
| Notes: | Foreign key shopCartID reference shopCartID in the Shopping\_Cart Table |  |  |  |  |  |
|  | Foreign key productID reference productID in the Product Table |  |  |  |  |  |

## 1.2 - Table creation SQL statements

CREATE TABLE Customer

(

customerID INT(3) NOT NULL AUTO\_INCREMENT,

firstName VARCHAR(20) NOT NULL,

secondName VARCHAR(20) NULL ,

surname VARCHAR(20) NOT NULL,

houseNumName VARCHAR(20) NOT NULL,

addressTwo VARCHAR(20) NOT NULL,

townCity VARCHAR(20) NOT NULL,

postCode VARCHAR(20) NOT NULL,

emailAddress TINYTEXT NOT NULL,

pwd TINYTEXT NOT NULL,

status int(1) NOT NULL DEFAULT 1,

PRIMARY KEY (customerID)

);

CREATE TABLE Product

(

productID INT(3) NOT NULL AUTO\_INCREMENT,

productName VARCHAR(20) NOT NULL,

productDesc VARCHAR(20) NOT NULL,

productPrice INT(5) NOT NULL,

productQty INT(3) NOT NULL,

status INT(1) NOT NULL DEFAULT 1,

PRIMARY KEY (productID)

);

CREATE TABLE Shopping\_Cart

(

shopCartID INT(5) NOT NULL AUTO\_INCREMENT,

storeID INT(3) NOT NULL,

customerID INT(3) NOT NULL,

date DATE NOT NULL,

orderStatus INT(1) NOT NULL,

status INT(1) NOT NULL DEFAULT 1,

PRIMARY KEY (shopCartID)

);

CREATE TABLE Shopping\_Cart\_Details

(

shopCartDetID INT(5) NOT NULL AUTO\_INCREMENT,

shopCartID INT(5) NOT NULL,

productID INT(3) NOT NULL,

prodAddedQty INT(3) NULL ,

PRIMARY KEY (shopCartDetID)

);

CREATE TABLE Store

(

storeID INT(3) NOT NULL AUTO\_INCREMENT,

storeName VARCHAR(20) NOT NULL,

storeAddress VARCHAR(20) NOT NULL,

storeEmail VARCHAR(20) NOT NULL,

status int(1) NOT NULL DEFAULT 1,

PRIMARY KEY (storeID)

);

ALTER TABLE Shopping\_Cart

ADD CONSTRAINT FK\_Customer\_TO\_Shopping\_Cart

FOREIGN KEY (customerID)

REFERENCES Customer (customerID);

ALTER TABLE Shopping\_Cart

ADD CONSTRAINT FK\_Store\_TO\_Shopping\_Cart

FOREIGN KEY (storeID)

REFERENCES Store (storeID);

ALTER TABLE Shopping\_Cart\_Details

ADD CONSTRAINT FK\_Shopping\_Cart\_TO\_Shopping\_Cart\_Details

FOREIGN KEY (shopCartID)

REFERENCES Shopping\_Cart (shopCartID);

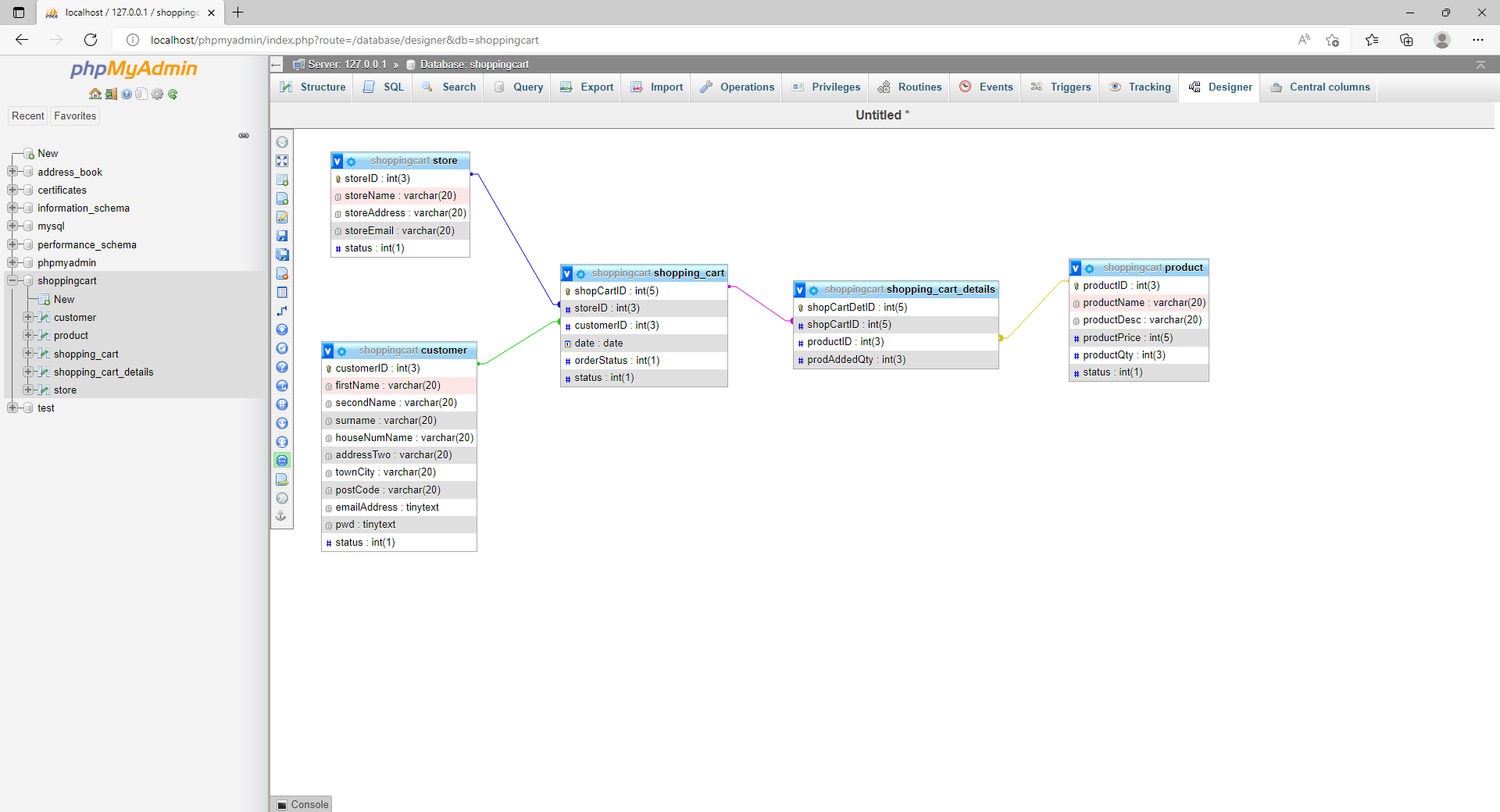
ALTER TABLE Shopping\_Cart\_Details

ADD CONSTRAINT FK\_Product\_TO\_Shopping\_Cart\_Details

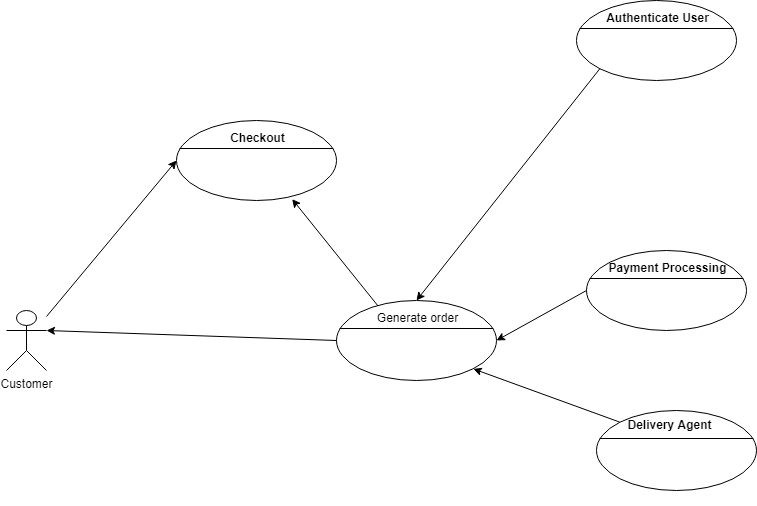
FOREIGN KEY (productID)

REFERENCES Product (productID);

1.3 - Database Tables

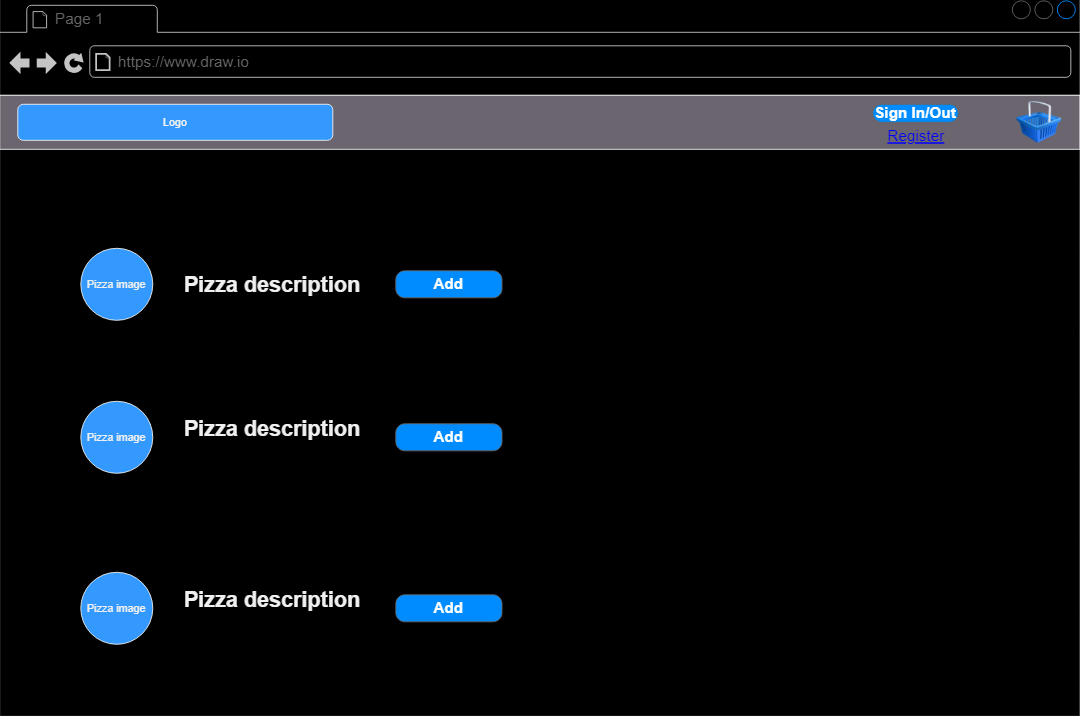


## 1.4 - Payment and Delivery Use Case diagram

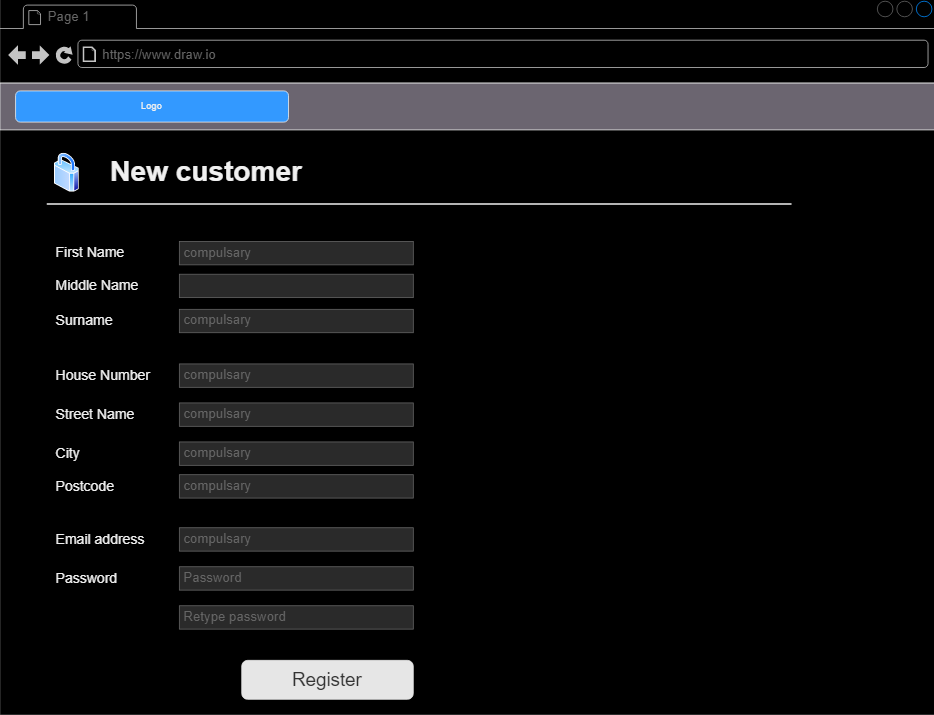


## 1.5 - Wireframe (V1.0)

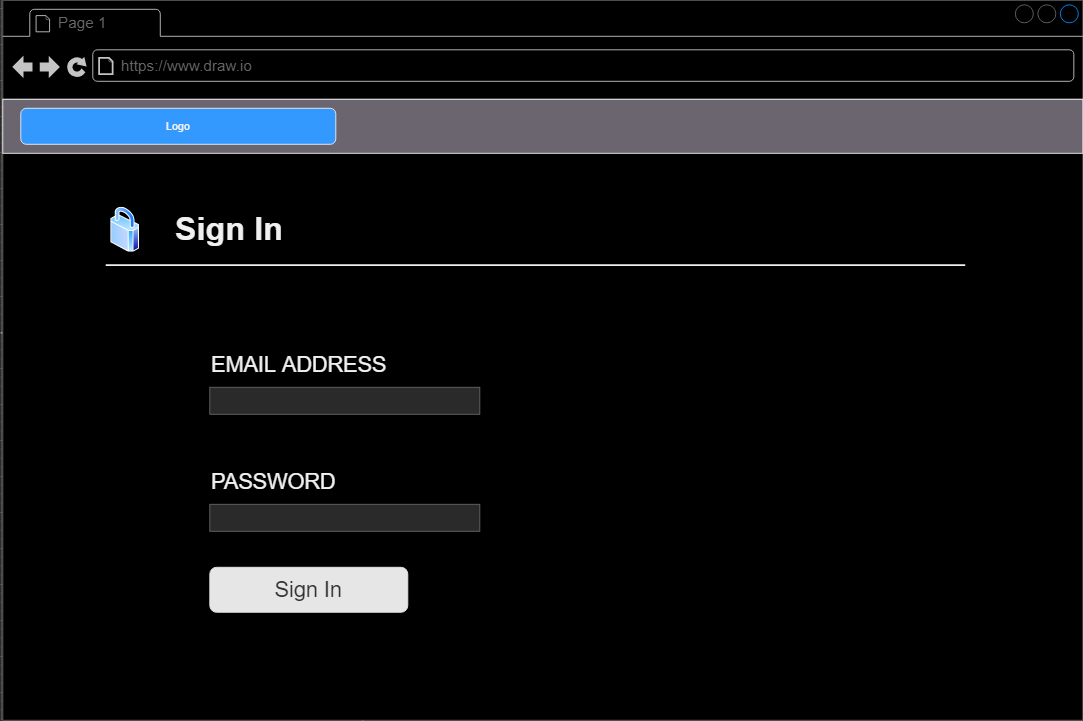
Home Page



Sign Up



Login



View Basket/Checkout

