**Project Deliverable 2 – Documentation and Coding**

**Faculty Name:** Information Technology

**Module Code:** ITECA3-B12

**Module Name:** Web Development and e-Commerce

**Content Writer:** Sandile Dlamini

**Copy Editor:** Mr Kyle Keens

**Submission Date:** Block 2 Week 4

|  |  |
| --- | --- |
| **Student Name** | Daniel de Waal |
| **Student Number** | DV.2022.Y6W2J1 |
| **Project Title** | Dungeons Dragons & Deals |
| **Submission Date** | 10/06/2024 |

**Table of Contents**

[2.1 Introduction 3](#_Toc97544592)

[2.2 Prototyping 3](#_Toc97544593)

[2.3 Designing 3](#_Toc97544594)

[2.4 Coding 3](#_Toc97544595)

[2.5 Conclusion 4](#_Toc97544596)

[2.6 Sign-off 4](#_Toc97544597)

# Introduction

This project has been the bane of my existence the last few weeks. I sincerely hope that whoever wrote this project specification reconsiders their career path, or at the very least comes to the realization that their module is not the only module in this degree, as the project specification was ridiculously broad, and lacking specificity. The requirement for an introduction in the second deliverable when I already crafted God-tier lore in my first deliverable is a needless attempt to flesh out an already bloated project. I lost months, maybe even years of my lifespan over this project. I now have less hair than I did at the start. I am more than willing to throw away 2 marks over this introduction, and I’m sure it’s not even close to the only marks I’ll lose for this nonsense project.

# Prototyping

Literally why do we need two prototypes? Does Eduvos encourage, nay, require their students to get it wrong the first time? What am I saying, I’ve come to expect mediocrity from Eduvos at this point. Their website is so broken I’m convinced one of their graduates wrote it.

1. **Prototype 1.**

**A black and white picture of a couple of black lines

Description automatically generated with medium confidence**

**Front page literally doesn’t need anything but a navigation bar. Products are on the product page.**

**A screenshot of a computer

Description automatically generatedProduct Page.**

**A screenshot of a login form

Description automatically generatedLogin Page**

**A screen shot of a login form

Description automatically generatedAccount Information page after logging in.**

**A screenshot of a computer

Description automatically generatedCart**

1. **Prototype 2.**

Your client must select the prototype you should use for the website development.

# Designing

1. **Class Responsibility Collaborator (CRC) cards**

A screenshot of a computer

Description automatically generated

1. **Enhanced Entity Relationship Diagram (EERD)**

A computer screen shot of a computer

Description automatically generated

1. **Context Diagram**

A diagram of a software company

Description automatically generated

1. **Data Flow Diagram (DFD)**

A diagram of a computer

Description automatically generated

1. **Use Case Diagram**

A diagram of a customer account

Description automatically generated

1. **Database Design**

**Have no idea what you meant by this, so here’s the schema:**

CREATE TABLE TB\_Profiles (

 vUUID      BINARY(16) NOT NULL,

 sName      TEXT       NOT NULL,

 sEmail     TEXT       NOT NULL,

 sCellphone TEXT           NULL,

 sAddress   TEXT       NOT NULL,

 sUsername  TEXT       NOT NULL,

 sPassword  TEXT       NOT NULL,

 CONSTRAINT TB\_Profiles\_PK PRIMARY KEY (vUUID)

);

CREATE TABLE TB\_Manufacturers (

 vUUID BINARY(16) NOT NULL,

 sName TEXT       NOT NULL,

 CONSTRAINT TB\_Manufacturers\_PK PRIMARY KEY (vUUID)

);

CREATE TABLE TB\_Products (

 vUUID             BINARY(16) NOT NULL,

 vManufacturerUUID BINARY(16) NOT NULL,

 sName             TEXT       NOT NULL,

 iStore            INTEGER    NOT NULL,

 sDescription      TEXT       NOT NULL,

 CONSTRAINT TB\_Products\_PK PRIMARY KEY (vUUID),

 CONSTRAINT TB\_Products\_Manufacturer\_FK FOREIGN KEY (vManufacturerUUID) REFERENCES TB\_Manufacturers(vUUID)

);

CREATE TABLE TB\_ProductListings (

 vUUID        BINARY(16) NOT NULL,

 vProductUUID BINARY(16) NOT NULL,

 iStock       INTEGER    NOT NULL,

 iPrice       INTEGER    NOT NULL,

 dtStart      TIMESTAMP  NOT NULL,

 dtEnd        TIMESTAMP      NULL,

 CONSTRAINT TB\_ProductListings\_PK PRIMARY KEY (vUUID),

 CONSTRAINT TB\_ProductListings\_Product\_FK FOREIGN KEY (vProductUUID) REFERENCES TB\_Products(vUUID)

);

CREATE TABLE TB\_CartItems (

 vUUID               BINARY(16) NOT NULL,

 vProfileUUID        BINARY(16) NOT NULL,

 vProductListingUUID BINARY(16) NOT NULL,

 iQuantity           INTEGER    NOT NULL,

 CONSTRAINT TB\_CartItems\_PK PRIMARY KEY (vUUID),

 CONSTRAINT TB\_CartItems\_Profile\_FK FOREIGN KEY (vProfileUUID) REFERENCES TB\_Profiles(vUUID),

 CONSTRAINT TB\_CartItems\_ProductListing\_FK FOREIGN KEY (vProductListingUUID) REFERENCES TB\_ProductListings(vUUID)

);

CREATE TABLE TB\_ProductListingImages (

 vUUID               BINARY(16) NOT NULL,

 vProductListingUUID BINARY(16) NOT NULL,

 iIndex              INTEGER    NOT NULL,

 sPhotoURL           TEXT       NOT NULL,

 sCaption            TEXT           NULL,

 CONSTRAINT TB\_ProductListingImages\_PK PRIMARY KEY (vUUID),

 CONSTRAINT TB\_ProductListingImages\_ProductListing\_FK FOREIGN KEY (vProductListingUUID) REFERENCES TB\_ProductListings(vUUID)

);

CREATE TABLE TB\_Reviews (

 vUUID               BINARY(16) NOT NULL,

 vProfileUUID        BINARY(16) NOT NULL,

 vProductListingUUID BINARY(16) NOT NULL,

 iRating             INTEGER    NOT NULL,

 sComment            TEXT           NULL,

 dtTime              TIMESTAMP  NOT NULL,

 CONSTRAINT TB\_Reviews\_PK PRIMARY KEY (vUUID),

 CONSTRAINT TB\_Reviews\_Profile\_FK FOREIGN KEY (vProfileUUID) REFERENCES TB\_Profiles(vUUID),

 CONSTRAINT TB\_Reviews\_ProductListing\_FK FOREIGN KEY (vProductListingUUID) REFERENCES TB\_ProductListings(vUUID)

);

CREATE TABLE TB\_Orders (

 vUUID        BINARY(16) NOT NULL,

 vProfileUUID BINARY(16) NOT NULL,

 sAddress     TEXT       NOT NULL,

 iStatus      INTEGER    NOT NULL,

 dtOrdered    TIMESTAMP  NOT NULL,

 dtPaid       TIMESTAMP      NULL,

 dtConcluded  TIMESTAMP      NULL,

 CONSTRAINT TB\_Orders\_PK PRIMARY KEY (vUUID),

 CONSTRAINT TB\_Orders\_Profile\_FK FOREIGN KEY (vProfileUUID) REFERENCES TB\_Profiles(vUUID)

);

CREATE TABLE TB\_OrderProductListings (

 vOrderUUID          BINARY(16) NOT NULL,

 vProductListingUUID BINARY(16) NOT NULL,

 iQuantity           INTEGER    NOT NULL,

 CONSTRAINT TB\_OrderProductListings\_PK PRIMARY KEY (vOrderUUID, vProductListingUUID),

 CONSTRAINT TB\_OrderProductListings\_Order\_FK FOREIGN KEY (vOrderUUID) REFERENCES TB\_Orders(vUUID),

 CONSTRAINT TB\_OrderProductListings\_ProductListing\_FK FOREIGN KEY (vProductListingUUID) REFERENCES TB\_ProductListings(vUUID)

);

# Coding

1. **Screenshots**

**A screenshot of a computer

Description automatically generated**

**A screenshot of a login form

Description automatically generated**

**A screen shot of a login form

Description automatically generated**

1. **Sample PHP Code**

maybe\_create\_mysql\_connection.php

**A screen shot of a computer code

Description automatically generated**

**add\_item\_to\_cart.php**

**A screen shot of a computer program

Description automatically generated**

**deactivate\_account.php**

**A screen shot of a computer code

Description automatically generated**

1. **Sample HTML Code**

**register/index.php**

<!DOCTYPE html>

<html>

 <head>

  <title>DD&D</title>

  <link rel="stylesheet" href="style.css">

  <link rel="stylesheet" href="/navigation\_bar.css">

  <script src="/navigation\_bar.js"></script>

  <script src="script.js"></script>

 </head>

 <body>

  <?php include("client\_generators/navigation\_bar.php"); ?>

  <div id="register-form">

   <h1 id="register-form-head">Register:</h1>

   <b><p1 class="register-form-label">Name:</p1></b><br>

   <div class="register-input"><input id="name-input" class="register-form-input" placeholder="Name" onkeypress="name\_input\_onkeypress(event)"></div>

   <br>

   <b><p1 class="register-form-label">Email:</p1></b><br>

   <div class="register-input"><input id="email-input" class="register-form-input" placeholder="email@provider.com" onkeypress="email\_input\_onkeypress(event)"></div>

   <br>

   <b><p1 class="register-form-label">Cellphone:</p1></b><br>

   <div class="register-input"><input id="cellphone-input" class="register-form-input" placeholder="" onkeypress="cellphone\_input\_onkeypress(event)"></div>

   <br>

   <b><p1 class="register-form-label">Address:</p1></b><br>

   <div class="register-input"><input id="address-input" class="register-form-input" placeholder="42 Somewhereington Ave, Someplacia" onkeypress="address\_input\_onkeypress(event)"></div>

   <br>

   <b><p1 class="register-form-label">Username:<p1></b><br>

   <div class="register-input"><input id="username-input" class="register-form-input" placeholder="Username" onkeypress="username\_input\_onkeypress(event)"></div>

   <br>

   <b><p1 class="register-form-label">Password:</p1></b><br>

   <div class="register-input"><input id="password-input" class="register-form-input" type="password" placeholder="Password" onkeypress="password\_input\_onkeypress(event)"></div>

   <br>

   <b><p1 class="register-form-label">Retype Password:</p1></b><br>

   <div class="register-input"><input id="retyped-password-input" class="register-form-input" type="password" placeholder="Password" onkeypress="retyped\_password\_input\_onkeypress(event)"></div>

   <br>

   <br><br>

   <div id="register-button-container"><button id="register-button" onclick="register\_button\_onclick(event)">Register</button></div>

   <label id="remember-me-label"><input id="remember-me-checkbox" type="checkbox" checked="checked">Remember me</label><br>

   <br><br>

   <label id="register-label">

    Already have an Account?&nbsp;<a href="/login">Login</a>

   </label>

  </div>

 </body>

</html>

**login/index.php**

<!DOCTYPE html>

<html>

 <head>

  <title>DD&D</title>

  <link rel="stylesheet" href="style.css">

  <link rel="stylesheet" href="/navigation\_bar.css">

  <script src="/navigation\_bar.js"></script>

  <script src="script.js"></script>

 </head>

 <body>

  <?php include("client\_generators/navigation\_bar.php"); ?>

  <div id="login-form">

   <h1 id="login-form-head">Login:</h1>

   <b><p1 class="login-form-label">Username:<p1></b><br>

   <div class="login-input"><input id="username-input" class="login-form-input" placeholder="Username" onkeypress="username\_input\_onkeypress(event)"></div>

   <br>

   <b><p1 class="login-form-label">Password:</p1></b><br>

   <div class="login-input"><input id="password-input" class="login-form-input" type="password" placeholder="Password" onkeypress="password\_input\_onkeypress(event)"></div>

   <br><br><br>

   <div id="login-button-container"><button id="login-button" onclick="login\_button\_onclick(event)">Login</button></div>

   <label id="remember-me-label"><input id="remember-me-checkbox" type="checkbox" checked="checked">Remember me</label><br>

   <br><br>

   <label id="register-label">

    New to DD&D?&nbsp;<a href="/register">Register</a>

   </label>

  </div>

 </body>

</html>

1. **Sample JavaScript Code**

**login/script.js**

function username\_input\_onkeypress(event) {

}

function password\_input\_onkeypress(event) {

}

function login\_button\_onclick(event) {

 const xhr = new XMLHttpRequest();

 xhr.open("POST", "/client\_scripts/verify\_credentials.php");

 xhr.setRequestHeader("content-type", "application/json; charset=UTF-8");

 const body = JSON.stringify({

  sUsername: document.getElementById("username-input").value,

  sPassword: document.getElementById("password-input").value

 });

 xhr.onload = () => {

  console.log(xhr.responseText);

  const response = JSON.parse(xhr.responseText);

  if (response['status'] == 0) {

   window.location.replace("/");

  }

 }

 xhr.send(body);

}

**cart/script.js**

function checkout\_button\_onclick() {

}

function cart\_item\_oninput(event) {

 if (event.srcElement.value == "") return;

 const quantity = parseInt(event.srcElement.value);

 if (quantity == 0) return;

 set\_cart\_item\_quantity(event.srcElement.id.split("-").at(-1), quantity);

}

function set\_cart\_item\_quantity(product\_listing\_uuid, quantity) {

 const xhr = new XMLHttpRequest();

 xhr.open("POST", "/client\_scripts/set\_cart\_item\_quantity.php");

 xhr.setRequestHeader("content-type", "application/json; charset=UTF-8");

 const body = JSON.stringify({

  product\_listing\_uuid: product\_listing\_uuid,

  quantity: (quantity == NaN) ? 1 : quantity

  });

 xhr.onload = () => {

  console.log(xhr.responseText);

  const response = JSON.parse(xhr.responseText);

  if (response['status'] == 2) {

   window.location.replace("/login");

  } else {

   let quantity\_element = document.getElementById("cart-item-quantity-input-" + product\_listing\_uuid);

   quantity\_element.value = quantity;

   let grandparent = quantity\_element.parentNode.parentNode;

   const price\_element = grandparent.getElementsByClassName("cart-item-price")[0].getElementsByTagName("label")[0];

   const price = parseFloat(price\_element.textContent.split("$")[1]);

   let total\_price\_element = grandparent.getElementsByClassName("cart-item-total-price")[0].getElementsByTagName("label")[0].textContent = "Total Cost: $" + (price \* quantity).toFixed(2);

  }

 }

 xhr.send(body);

}

document.addEventListener("DOMContentLoaded", function() {

 for (let cart\_item\_element of document.getElementsByClassName("cart-item-quantity-input")) {

  cart\_item\_element.addEventListener("input", cart\_item\_oninput);

 }

});

1. **Sample CSS Code**

**store/product\_listing.css**

.product-listing {

 display: flex;

 height: 90%;

 margin: 0.5%;

 background-color: #dddddd;

}

.product-listing-header {

 flex-grow: 0;

 background-color: #cccccc;

 margin: 0.5%;

}

.product-listing-header .product-listing-image {

 height: 100%;

}

.product-listing-body {

 flex-grow: 1;

 background-color: #cccccc;

 margin: 1%;

}

.product-listing-title {

 display: flex;

 justify-content: center;

 margin: 0.25%;

 font-size: 16pt;

 background-color: #bbbbbb;

}

.product-listing-description {

 display: flex;

 justify-content: center;

 font-size: 12pt;

}

.product-listing-footer {

 display: grid;

 justify-content: center;

 grid-template-columns: repeat(2, 50%);

 grid-template-rows: repeat(2, 50%);

 padding: 1%;

}

.product-listing-price {

 display: flex;

 grid-column: 1 / span 1;

 grid-row: 1 / span 1;

 justify-content: center;

 align-items: center;

 white-space: nowrap;

 padding-left: 5pt;

 padding-right: 5pt;

 background-color: #aaaaaa;

}

.product-listing-stock {

 display: flex;

 grid-column: 0 / span 1;

 grid-row: 1 / span 1;

 justify-content: center;

 align-items: center;

 background-color: #bbbbbb;

 padding-left: 5pt;

 padding-right: 5pt;

}

.product-listing-add-cart {

 grid-column: 1 / span 2;

 grid-row: 0 / span 1;

 background-color: #cccccc;

}

**navigation\_bar.css**

#navigation-bar-container {

 display: flex;

 position: relative;

 align-items: center;

 /\* justify-content: space-between; \*/

 left: 4%;

 width: 92%;

}

#navigation-bar-container::before {

 content: '';

 position: absolute;

 width: 100%;

 height: 50%;

 left: 0%;

 top: 25%;

 z-index: -1;

 border-radius: 16px;

 background-color: #111111;

}

#logo {

 flex-grow: 0;

 width: 128px;

 height: 128px;

 margin-left: 25px;

 margin-top: 10px;

 margin-right: 25px;

}

#search-container {

 display: flex;

 position: relative;

 align-items: center;

 flex-grow: 1;

 height: 24pt;

 background-color: #444444;

 border-radius: 12px;

}

#search-input {

 flex-grow: 1;

 height: 16pt;

 border-radius: 6px;

 border: none;

 margin-left: 4pt;

 margin-right: 1pt;

}

#store-selection {

 flex-grow: 0;

 height: 16pt;

 border-radius: 6px;

 border: none;

 margin-left: 1pt;

 margin-right: 4pt;

}

#search-button {

 display: flex;

 flex-grow: 0;

 position: relative;

 justify-content: center;

 align-items: center;

 border-radius: 8px;

 margin-right: 5px;

 background-color: #444444;

 cursor: pointer;

}

#search-button-magnifier {

 width: 24px;

 height: 24px;

 align-items: center;

 margin: 1px;

}

#navigation-bar{

 display: flex;

 flex-grow: 0;

 position: relative;

 align-items: center;

 justify-content: space-between;

 margin-left: 25px;

 margin-right: 25px;

}

#account {

 display: flex;

 flex-grow: 0;

 position: relative;

 align-items: center;

 justify-content: center;

 border-radius: 8px;

 background-color: #444444;

 margin-right: 1px;

}

#pfp {

 width: 24px;

 height: 24px;

 align-items: center;

 margin: 1px;

 border-radius: 50%;

}

#cart-container {

 flex-grow: 0;

 display: flex;

 position: relative;

 align-items: center;

 border-radius: 8px;

 background-color: #444444;

 cursor: pointer;

 margin-left: 1px;

}

#cart {

 width: 24px;

 height: 24px;

 margin: 1px;

 margin-left: 4px;

}

#cart-count {

 margin-left: 4px;

 margin-right: 6px;

}

1. **Sample MySQL Table Screenshots**

**Zoom in if you can’t see.**

**TB\_Profiles:**

A black screen with white text

Description automatically generated

**TB\_Products** A screenshot of a computer

Description automatically generated

TB\_ProductListingsA screenshot of a computer

Description automatically generated

TB\_ProductListingImagesA screenshot of a computer screen

Description automatically generated

TB\_Manufacturers

A screenshot of a computer program

Description automatically generated

**TB\_CartItems**A black and white screen with numbers

Description automatically generated

# Conclusion

For better or for worse, this has been a… transformative experience in my life, as I’ve come to realize that academia is a dying industry.

BTW, I’m colour blind so I hope my website being grey isn’t a problem. Wouldn’t want Eduvos discriminating against students with disabilities, would we?

# Sign-off

|  |  |  |  |
| --- | --- | --- | --- |
| rc |  |  | jm |
| Richard Clarkson  jh |  |  | Jeremy May  ddw |
| James Hammond  Date: 10/06/2024 |  |  | Daniel de Waal |