

Title of project

**Patrick Bondaruk**

**N00201161**

Software Project

CA 2 – Develop a PHP shopping cart website

Year 2 2021-22

DL836 BSc (Hons) in Creative Computing

Table of Contents

[1 Introduction 1](#_Toc94698843)

[2 Business Concept 2](#_Toc94698844)

[2.1 Business Idea 2](#_Toc94698845)

[2.2 Business model 2](#_Toc94698846)

[2.3 Market Research 2](#_Toc94698847)

[2.4 Marketing/Advertising 2](#_Toc94698848)

[2.5 Suppliers 2](#_Toc94698849)

[2.6 Competitors 2](#_Toc94698850)

[2.7 Employees 2](#_Toc94698851)

[2.8 Environmental Impact 2](#_Toc94698852)

[3 Requirements 3](#_Toc94698853)

[3.1 Introduction 3](#_Toc94698854)

[3.2 Requirements gathering 3](#_Toc94698855)

[3.2.1 Similar applications 3](#_Toc94698856)

[3.2.2 Interviews 3](#_Toc94698857)

[3.3 Requirements modelling 3](#_Toc94698858)

[3.3.1 Functional requirements 3](#_Toc94698859)

[3.3.2 Non-functional requirements 3](#_Toc94698860)

[3.3.3 Use Case Diagrams 4](#_Toc94698861)

[3.4 Feasibility 5](#_Toc94698862)

[4 Database Design 6](#_Toc94698863)

[4.1 Description 6](#_Toc94698864)

[4.2 Business Reporting Requirements 6](#_Toc94698865)

[4.3 Textual Representation of Data-Set 6](#_Toc94698866)

[4.4 Business Rules 7](#_Toc94698867)

[4.5 Entity Relationship Diagram 7](#_Toc94698868)

[4.6 Tables 8](#_Toc94698869)

[4.7 Database Dictionary 9](#_Toc94698870)

[5 System Design/ Architecture Overview 10](#_Toc94698871)

[5.1 Introduction 10](#_Toc94698872)

[5.2 Model View Controller 10](#_Toc94698873)

[5.3 User Authenticaion 10](#_Toc94698874)

[5.4 Routing 10](#_Toc94698875)

[5.5 Templating 10](#_Toc94698876)

[6 Testing 11](#_Toc94698877)

[6.1 Introduction 11](#_Toc94698878)

[6.2 Functional Testing 11](#_Toc94698879)

[6.2.1 Login/Registration 12](#_Toc94698880)

[6.2.2 Navigation 12](#_Toc94698881)

[6.2.3 Calculation 12](#_Toc94698882)

[6.2.4 CRUD 12](#_Toc94698883)

[6.3 Discussion of Functional Testing Results 13](#_Toc94698884)

[6.4 User Testing 13](#_Toc94698885)

[6.5 Conclusion 13](#_Toc94698886)

[7 Project Management 14](#_Toc94698887)

[7.1 Introduction 14](#_Toc94698888)

[7.2 Project Phases 14](#_Toc94698889)

[7.2.1 Requirements 14](#_Toc94698890)

[7.2.2 Design 14](#_Toc94698891)

[7.2.3 Implementation 14](#_Toc94698892)

[7.2.4 Testing 14](#_Toc94698893)

[7.3 SCRUM Methodology 14](#_Toc94698894)

[7.4 Project Management Tools 15](#_Toc94698895)

[7.4.1 Github Project 15](#_Toc94698896)

[7.4.2 GitHub 15](#_Toc94698897)

[8 Reflection 16](#_Toc94698898)

[8.1 Your views on the project 16](#_Toc94698899)

[8.2 How could the project could be developed further? 16](#_Toc94698900)

[8.3 Assessment of your learning. 16](#_Toc94698901)

[8.4 Completing a large software development project 16](#_Toc94698902)

[8.5 Technical skills 16](#_Toc94698903)

[8.6 Further competencies and skills 16](#_Toc94698904)

# Introduction

Overall aim

Application area

Technologies

PHP, MySQL, Bootstrap, CSS, Vanilla

Tools

IDE, phpMyAdmin, Miro

Project management

GitHub

Business Concept

Requirements

Design

Implementation

Testing

Reflection

# Business Concept

## Business Idea

Describe the business idea of your project.

## Business model

## Market Research

Market for Product/Service

Customers - Demographics, Profile

## Marketing/Advertising

## Suppliers

## Competitors

## Employees

## Environmental Impact

# Requirements

## Introduction

The application will be an appliance shop where people will be able to list, purchase, checkout in a smooth process. The customers will be able to add the items that they desire to purchase, edit the items that they’re purchasing, and checkout for delivery or pickup and add the details for the payment in a single page so it will be faster and efficient.

## Requirements gathering

### Similar applications

Harvey Norman

The similar application is a Harvey Norman appliance shop. It is a shop where they sell appliances and all sorts of other products that they provide.

Two people sitting at a desk with computers

Description automatically generated with low confidence

A collage of a living room

Description automatically generated with low confidence

The application provides different sections of appliance equipment that can lead to a more specific product of choice.

Like going from the ‘Cooking Appliances’ section, leading to going in ‘Ovens’ section, and then leads to a list of ovens while still providing more detailed sections above.

Graphical user interface, application

Description automatically generated

A picture containing text, indoor, screenshot, kitchen appliance

Description automatically generated

Graphical user interface, application, PowerPoint

Description automatically generated

The checkout page provides the details for delivery, and a payment option section in one.

Graphical user interface, application, email, website

Description automatically generated

Graphical user interface, text, application

Description automatically generated

Advantages

* The products can be searched by categories and brands
* Provides a filtering section for narrowing down the options of products
* Specific products provide the product’s specifications like weight, battery life, energy consumption grade etc.

Disadvantages

* Does not show the ratings and reviews of the products, and so the products have to be viewed individually to see if there are any ratings or reviews of the product

Argos

Another similar application for an appliance shop is Argos. It is another alternative application where they provide appliances with a range of furniture, electronics, kitchen equipment, gadgets, outdoor equipment and so forth.

Graphical user interface, website

Description automatically generated

Text

Description automatically generated

In a similar way the products can be selected and lead to a more specific selection of the products. In this example is ‘mobile phones and accessories’, and can be selected further by type, brands, price range and so forth.

Graphical user interface, website

Description automatically generated

The list of products shows the name, price, availability, ratings along with the amount of reviews, and also the option to compare the products.

Graphical user interface

Description automatically generated with medium confidence

A picture containing graphical user interface

Description automatically generated

Advantages

* Shows the amount of user reviews and the average rating
* Displays if it provides a delivery and/or pick-up
* Similar products can be compared to see the differences.

Disadvantages

* The application requires a few steps to complete a purchase transaction of a product

### Interviews

Conduct interviews with 2 or 3 users to find out what the important features for them for the app are. There may be various issues that arise in multiple interviews. These can be grouped together into a number of themes.

## Requirements modelling

### Functional requirements

The application should provide the users to be able to log in/sign up to their accounts. The users should receive a message or an email of their order confirmation, and additional function the application should have is for the users to be able to reset their passwords of their accounts in case if they forget it.

### Non-functional requirements

The application should have a solid security, so when users sign up/log in the data and personal information should be kept secured in the database system. Another requirement should be on usability, so when users will be using the application they should be able to use it with ease and to not have any misleading with the products. Another additional requirement the application should have is the localization, so the application should be matching the local specifics.

### Use Case Diagrams

Diagram

Description automatically generated

## Feasibility

The software that are planned to be used is mainly HTML, PHP and Laravel. These software programs should be compatible and it wouldn’t cause technical issues.

# Web application Design

## Layout

The layout for the application will be mostly minimalist with some of detailed layouts in specific sections of the application. The application will be responsive with different layout designs depending on the size of the device. The application won’t really depend on the bootstrap but it can.

## Interaction

The user will interact with the application by having a search bar above, the profile for sign up and log in, the cart for products, and settings menu. Those will be the main navigation elements for the user.

## Colour schemes

The color palette that I’m considering to use consistently for the application, is that for the main colors it will be a range of red, blue, orange, and possibly purple. The background will be an off-white color.

A picture containing square

Description automatically generated

## Font choices

The main font for headings that I’ll be using is Lato, and the main font that I’ll be using for paragraph text is IBM Plex Mono.

Text

Description automatically generated with medium confidence

## Wireframes

Table

Description automatically generated

# Database Design

## Description

A company has a website that sells appliances and other products for different areas like kitchen, garden, electronics. They would need a database for all their appliances and order places. For each order place, they would need products bought, total price, date of the order, and how long it will take to deliver. The database needs to keep track of all appliances that are being sold. Customers will have to input their information when registering an account. Customers will also have to input their card details when making a payment for their order.

## Business Reporting Requirements

Substitute in here the information the users of your application will want to be able to view.

1. Organisers need to be able to create, read, update, and delete: products, brands, and sales.
2. Users will need to be able to find all products ordered by their date.
3. Users may want to find a product by a specific brand.
4. Users need to find all products using a list of brands.
5. Users need to find the product for a specific price.
6. Users need to find the products using a brands name.
7. Customers may need to find the list of staff contacts.
8. Organisers need to display a list of products that are assigned to a specific brand.

## Textual Representation of Data-Set

Substitute in here the tables for your database

**FESTIVAL** (title, description, latitude, longitude, city, start\_date, end\_date, image\_id)

**PERFORMER** (title, description, contact\_email, contact\_phone, image\_id)

**GENRE** (title, description)

**IMAGE** (id, filename)

**SHOW** (date, start\_time, end\_time, performer\_id, stage\_id)

**STAGE** (title, description, location, festival\_id, image\_id)

**GENRE**\_**PERFORMER** (id, genre\_id, performer\_id)

**EMPLOYEE** (name, phone, email)

**FESTIVAL\_EMPLOYEE** (employee\_id, festival\_id, role)

## Business Rules

Substitute in here the business rules for your database

 A **Brand** has many **Products**.

 A **Product** belongs to one **Brand**.

 An **Order** has many **Products**.

 A **Product** is placed in one **Order**.

 A **Customer** can place many **Orders**.

 An **Order** is ordered by one **Customer**.

 A **Product** can have many **Images**.

 An **Image** can be associated with a **Product**.

## Entity Relationship Diagram

Substitute in here your ERD from draw.io

A diagram of a flowchart

Description automatically generated with low confidence

## Tables

A picture containing text, indoor

Description automatically generated

## Database Dictionary

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Table | Attribute | Datatype | Range | Required | PK/FK | FK Ref Table |
| users | id | Int |  | Yes | PK |  |
|  | first\_name | Varchar(20) |  | Yes |  |  |
|  | last\_name | Varchar(20) |  | Yes |  |  |
|  | email | Varchar(20) |  | Yes |  |  |
|  | phone\_number | Varchar(20) |  | Yes |  |  |
|  | address | Varchar(40) |  | Yes |  |  |
| user\_role | id | Int |  | Yes | PK |  |
|  | user\_id | Varchar(20) |  | Yes | FK | users |
|  | role\_id | Varchar(20) |  | Yes | FK | roles |
| roles | id | Int |  | Yes | PK |  |
|  | name | Varchar(20) |  | Yes |  |  |
|  | description | Varchar(100) |  | Yes |  |  |
| orders | id | Varchar(20) |  | Yes | PK |  |
|  | date | Date |  | Yes |  |  |
|  | status | Varchar(20) |  | Yes |  |  |
|  | amount | Decimal |  | Yes |  |  |
|  | user\_id | Varchar(20) |  | Yes | FK | users |
| products | id | Int |  | Yes | PK |  |
|  | name | Varchar(20) |  | Yes |  |  |
|  | description | Varchar(100) |  | Yes |  |  |
|  | weight | Decimal |  | Yes |  |  |
|  | price | Decimal |  | Yes |  |  |
|  | image\_location |  |  | Yes |  |  |
| order\_products | id | Int |  | Yes | PK |  |
|  | quantity | Decimal |  | Yes |  |  |
|  | total\_price | Decimal |  | Yes |  |  |
|  | order\_id | Varchar(20) |  | Yes | FK | orders |
|  | product\_id | Varchar(20) |  | Yes | FK | products |
| brands | id | Int |  | Yes | PK |  |
|  | name | Varchar(20) |  | Yes |  |  |
| product\_brand | id | Int |  | Yes | PK |  |
|  | product\_id | Varchar(20) |  | Yes | FK | products |
|  | brand\_id | Varchar(20) |  | Yes | FK | brands |

# System Design/ Architecture Overview

* 1. Introduction

This section will describe the internal functionality of the web framework that you have chosen for the implementation.

* 1. Model View Controller

Explain the follows a model-view-controller design pattern and how it is implemented in your web application.

* 1. User Authentication

There is an authentication controller in the controllers folder that verifies and confirms the user that is logging in or registering.

* 1. Routing

Describe the routes that were defined in the web application

The routes are implemented for ordinary users, admin users, and general. For the general is the welcome page and the about page which don’t require a user to be logged in. For the ordinary user there are routes for the home page, show, and index. For the admin user there are also same routes like the ordinary user and also a create, store, edit, update, and destroy routes.

Text

Description automatically generated

* 1. Templating

Describe the templating engine and how it was used to configure/ style the web application.

Add a sequence diagram in this section and other diagrams that illustrate the architecture clearly.

Diagram

Description automatically generated

# Testing

* 1. Introduction

This chapter describes the testing that has been undertaken for the application. This chapter is presented in two sections:

1. Functional Testing
2. User Testing

Functional testing is a type of software testing whereby the system is tested against the functional requirements. The app is tested by looking to see if the actual output for a given input corresponds with the expected output. The tests should be based on the requirements for the app. The results of functional testing can indicate if a piece of software is functional and working, but not if the software is easy to use.

User testing looks to see if a piece of software is easy and intuitive for the user.

* 1. Functional Testing

This section describes the functional tests which were carried out on the app. These functional tests can be categorised as: (whatever is relevant to your app)

Login/Registration

Navigation

Calculation

CRUD

Functional testing generally uses a Black Box Testing technique which means that the internal logic of the system being tested is not of interest to the tester. The tester is only interested in whether the actual output agrees with the expected output.

* + 1. Login/Registration

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Test No | Description of test case | Input | Expected Output | Actual Output | Comment |
| 1 | A user logging in to their account | A user enters the credentials to log in | A user is logged in to their account | A user is logged in to their account | No issues occurred during the process |
|  |  |  |  |  |  |
|  |  |  |  |  |  |

* + 1. Navigation

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Test No | Description of test case | Input | Expected Output | Actual Output | Comment |
| 1 | The user viewing the products through the navigation bar | The user selects the ‘products’ section | The user is brought to the products page | The user is brought to the products page | No issues occurred in the process |
|  |  |  |  |  |  |
|  |  |  |  |  |  |

* + 1. Calculation

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Test No | Description of test case | Input | Expected Output | Actual Output | Comment |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |

* + 1. CRUD

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Test No | Description of test case | Input | Expected Output | Actual Output | Comment |
| 1 | The admin is creating the product | The admin inputs the information for the product | The product has been created on the list | The product was created without the image | The asset was not added for the images to work |
| 2 | The admin is creating the product | The admin inputs the information for the product | The product has been created on the list | The product has been created on the list | The image on the product works and is being stored |
|  |  |  |  |  |  |

## Discussion of Functional Testing Results

The majority of the functionality testing went with little to no issues. The minor issues like for the images assets was easy the implement and to have it working.

* 1. User Testing
  2. Conclusion

Discussion of test results.

# Project Management

## Introduction

This chapter describes how the project was managed. It shows the phases of the project, going from the project idea through the requirements gathering, the specification for the project, the design, implementation and testing phases for the project. It also discusses GitHub as a tool which assist in project management.

## Project Phases

In this section, describe each of the following project phases. Explain any issues which arose for each of the phases.

### Requirements

### Design

### Implementation

### Testing

## SCRUM Methodology

Sprints

## Project Management Tools

### Github Project

Description

Include screen shots

How it worked in practice

### GitHub

Description

How it is used

How it worked in practice

# Reflection

## Your views on the project

Describe how you feel the project went from your perspective.

## How could the project could be developed further?

## Assessment of your learning.

Critically assess your learning. List what skills and competencies you have learned developed in this Continuous Assessment.

List which part of the project would need further development and itemize where you feel you have not satisfactorily completed the continuous assessment.

## Completing a large software development project

Describe what you have learnt from the project, from the point of view of completing a large software development project.

## Technical skills

Describe what you have learnt from the project, from a technical skills viewpoint.

## Further competencies and skills

Describe any extra competencies and skills that would help you with your development in the work place.

# References

Add a list of references that you used to complete the project.

The Department of Technology and Psychology in IADT uses APA 7th referencing style.

Use alphabetical order for your references.

This site gives details about how to cite websites using APA:

https://www.wikihow.com/Cite-a-Website-in-APA

The following is a useful site for creating citations for APA for websites.

<http://www.citationmachine.net/apa/cite-a-website>

You can also use the Referencing tab within Microsoft Word to enter reference information manually. Word then creates an APA style reference.