

Title of project

**Your name Liam Bedford**

**Your student number N00202070**

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 3](#_Toc94698852)

[3 Requirements 4](#_Toc94698853)

[3.1 Introduction 4](#_Toc94698854)

[3.2 Requirements gathering 5](#_Toc94698855)

[3.2.1 Similar applications 5](#_Toc94698856)

[3.2.2 Interviews 9](#_Toc94698857)

[3.3 Requirements modelling 9](#_Toc94698858)

[3.3.1 Functional requirements 9](#_Toc94698859)

[3.3.2 Non-functional requirements 9](#_Toc94698860)

[3.3.3 Use Case Diagrams 9](#_Toc94698861)

[3.4 Feasibility 10](#_Toc94698862)

[4 Database Design 14](#_Toc94698863)

[4.1 Description 14](#_Toc94698864)

[4.2 Business Reporting Requirements 14](#_Toc94698865)

[4.3 Textual Representation of Data-Set 14](#_Toc94698866)

[4.4 Business Rules 15](#_Toc94698867)

[4.5 Entity Relationship Diagram 15](#_Toc94698868)

[4.6 Tables 16](#_Toc94698869)

[4.7 Database Dictionary 17](#_Toc94698870)

[5 System Design/ Architecture Overview 18](#_Toc94698871)

[5.1 Introduction 18](#_Toc94698872)

[5.2 Model View Controller 18](#_Toc94698873)

[5.3 User Authenticaion 18](#_Toc94698874)

[5.4 Routing 18](#_Toc94698875)

[5.5 Templating 19](#_Toc94698876)

[6 Testing 19](#_Toc94698877)

[6.1 Introduction 19](#_Toc94698878)

[6.2 Functional Testing 20](#_Toc94698879)

[6.2.1 Login/Registration 20](#_Toc94698880)

[6.2.2 Navigation 20](#_Toc94698881)

[6.2.3 Calculation 21](#_Toc94698882)

[6.2.4 CRUD 21](#_Toc94698883)

[6.3 Discussion of Functional Testing Results 21](#_Toc94698884)

[6.4 User Testing 21](#_Toc94698885)

[6.5 Conclusion 21](#_Toc94698886)

[7 Project Management 22](#_Toc94698887)

[7.1 Introduction 22](#_Toc94698888)

[7.2 Project Phases 22](#_Toc94698889)

[7.2.1 Requirements 22](#_Toc94698890)

[7.2.2 Design 22](#_Toc94698891)

[7.2.3 Implementation 22](#_Toc94698892)

[7.2.4 Testing 22](#_Toc94698893)

[7.3 SCRUM Methodology 22](#_Toc94698894)

[7.4 Project Management Tools 23](#_Toc94698895)

[7.4.1 Github Project 23](#_Toc94698896)

[7.4.2 GitHub 23](#_Toc94698897)

[8 Reflection 24](#_Toc94698898)

[8.1 Your views on the project 24](#_Toc94698899)

[8.2 How could the project could be developed further? 24](#_Toc94698900)

[8.3 Assessment of your learning. 24](#_Toc94698901)

[8.4 Completing a large software development project 24](#_Toc94698902)

[8.5 Technical skills 24](#_Toc94698903)

[8.6 Further competencies and skills 24](#_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

My business idea is a shopping website that allows users to put up kickboxing boxing gear for sale and allows other users to view and purchase these products.

## Business model

The business will make money by advertising on the website while also taking 10% of the cost of an item when an item is sold on the website.

## Market Research

**Market for Product/Service**

The target market consists of people of all ages who kickbox. They will buy products as they are cheaper than buying brand new products from other sources. I think there is definitely a big enough market to support and sustain this product for a long time.

**Customers - Demographics, Profile**

The service focuses on people who train at kickboxing and who need new training gear, whether they are beginners, amateurs, or fighters.

## Marketing/Advertising

## Suppliers

## Competitors

Competitors in the same space are sites like ebay, amazon or craigslist. Weaknesses of these competitors is that they are not specific to used Muay Thai Training gear. They sell a more larger base of products at higher costs. These sites all allow users to put products up for sale and allows them to purchase products.

## Employees

## Environmental Impact

# Requirements

## Introduction

My software project is a web application/website which allows users to sell and purchase 2nd hand kickboxing gear. This application may be used when for someone wants to purchase 2nd hand training gear or when someone would like to sell on old training gear they no longer use.

For example, a customer may be an individual who has just started training at kickboxing and would like to have their own gear to use in training, rather than using the gyms gear. With this application, the customer can purchase high quality second hand gear for a low price, as oppose to buying premium gear first hand, when they are still a beginner to the sport. Applications similar to this are websites like ebay, amazon or size.ie. All of these websites allow users to buy second items and allows users to list items up for sale on the site. With this application, those who kickbox will now have an alternative place to buy their gear for lower prices and a place to sell on their unused kickboxing gear.

You can write a bit about your project area. Each paragraph has a blank line between it and the previous paragraph

## Requirements gathering

### Similar applications

Look at and document 2 similar applications. Be sure to include the following for each:

* Screen shots
* Descriptions
* Advantages
* Disadvantages

Application 1: Depop

Timeline

Description automatically generated with low confidence

Depop Description: Depop is a website that allows users to sell and purchase clothing items online. Users can put items up for sale, other users can then buy said item.

Advantages:

* Intuitive design and layout
* Aesthetically pleasing website
* Easy to use
* Navbar scrolls
* Fully functional

Disadvantages:

* Design is a little plain
* Scrolling footer makes the website feel cluttered

Application 2: Vinted

A collage of different outfits

Description automatically generated with low confidence

Vinted Description: Vinted is a another website that allows users to sell and purchase clothing items online, similar to depop. Vinted focuses more on high-street brands and luxury clothing items. However, the sites layout and design works well for a buying/selling website.

Advantages:

* Intuitive design and layout
* Easy to navigate
* Navbar scrolls
* Shows items for sale at first glance and shows lots of information about them
* Fully functional

Disadvantages:

* Site layout is a little busy/cluttered
* Layout of items is the same the whole way through the home page
* Navbar elements are very small and not pronounced
* Banner on homepage is not aesthetically pleasing

### 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

* Log In/Out functionality
* Create Account functionality
* Allow users to purchase items
* Security

### Non-functional requirements

* Basket function
* Wishlist function
* Favorites function
* Maintainability

### Use Case Diagrams

Consists of actors and use cases. You should document each individual use case.

Delete the following diagram and insert your diagram. Use draw.io



## Feasibility

This section describes which technologies are planned to be used in the development of the application. It then explains if there are any issues in terms of the technical feasibility of the project, for example, if there are two different types of software which may have compatibility issues.

Software I used in the development of this application includes Laravel, phpMyAdmin

# Web application Design

## Layout

Describe the layout of your web application. Does this depend on a framework like bootstrap? Is it repsonsive?

My web application depends on Laravel as a framework.

My web application contains a number of pages, a homepage, an item browse page, a log-in page, a sell page and a basket page.

At the very top of each page of my web application is a navbar, as well as a footer found at the bottom of the page.

Underneath the navbar is a banner which tells users what the website is and contains a link to allow users to sell items/log in

Items for sale are then listed in rows, under headings such as popular, new and trending.

## Interaction

What are the navigation elements, form elements. How does the user interact with the application?

Navigation elements include links found in the navbar, footer, throughout the pages of the site and on images of products.

## Color schemes

The color palette I decided to use throughout my web application is a simple palette containing white, black and a shade of red (#f91723). I chose these colors as they are not too distracting or loud in the context of a shopping website, however the red I chose helps make the website pop a little more and fits with the theme of the site.

## Font choices

Specify the fonts that you will use for different types of text. Include samples for paragraph text, headings and bold and italicised text.

The font family that I decided to use throughout my web application is the Do Hyeon font family. I chose this font as it visually stood out to me and in my opinion, would give my website a modern feel.

## Wireframes

Describe how to navigate from one page to the next by adding an diagram of the different screens and what the main functionality is.

On the main home page users can navigate to various other pages of the website. The login button in the top left of the page takes users to the Sign In/Register Page. The Sell button takes users to the Sell Item page. Clicking on one of the displayed items for sale brings users to the corresponding item’s item view page. User’s can navigate to the home page by clicking the logo on every page. Clicking New items or shop brands or any of the item specific tabs brings users to the item browse page.

Chart, treemap chart

Description automatically generated

# Database Design

## Description

The company has a website that sells kickboxing gear from different brands. A database is needed for all the training items and orders. For each order placed, information such as items bought, total price, date/time of order , and more must be tracked. The database needs to keep track of which items are selling. Users will need to create an account by registering with their information, such as username, email and password.

## Business Reporting Requirements

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

1. Users need to be able to create, read, update, and delete: items, brands, conditions, prices, and type.
2. Users will need to be able to find all items ordered by their listing date.
3. Users may want to find a festival by a specific brand.
4. Users need to find all festivals using a list of conditions.
5. Users need to find the price for a specific show.
6. Users need to find the shows using a brand name.
7. Performers may need to find the list of festival types.
8. Users need to find item by location and the location needs to be displayed on a Google Map
9. User may need to find items by location.
10. Users need to find prices of an item by the items condition
11. Users need to display a list of customers that are assigned to a specific item.

## 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)

**TYPE** (title, description)

**CONDITION** (title, description)

**IMAGE** (id, filename)

**BRAND** (title, popularity, description)

**PRICE** (number)

**BRAND**\_**TYPE** (id, brand\_id, type\_id)

**SELLER** (name, phone, email)

**ITEM\_SELLER** (item\_id, seller\_id, role)

## Business Rules

Substitute in here the business rules for your database

 A SELLER has many **ITEMS**.

 A **ITEM** belongs to one **SELLER**.

 A **ITEM** is one **BRAND**.

 A **ITEM** is one **TYPE**.

 A **BRAND** can have many **ITEMS**.

 A **ITEM** has one **CONDITION**.

 A **Performer** can have many **Genres**.

 A **Genre** can belong to many **Performers**.

 A **Performer** can have a single **Image**.

 A **Festival** can have a single **Image**.

 A **Stage** can have a single **Image**.

 An **Image** can be associated with a **Performer**, **Festival**, or **Stage**

 A **Festival** can have many **Employees** associated with it

 An **Employee** can be assigned to one **Festival** at a time

## Entity Relationship Diagram

Substitute in here your ERD from draw.io

Diagram

Description automatically generated

## Tables

Substitute in here your tables and the relationships between tables from draw.io in the format you used in DBMS with Mohammed.

A picture containing graphical user interface

Description automatically generated

## Database Dictionary

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Table | Attribute | Datatype | Range | Required | PK/FK | FK Ref Table |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |

# System Design/ Architecture Overview

* 1. Introduction

This section will describe the internal functionality of the web framework that you have chosed for the implementation. Add further sections if required by the specification of your web application

Laravel is the web framework I used for this project. Laravel is a php framework that allows users to develop their own web applications. Laravel is a server-side frame work and uses a Model-View-Controller Design(MVC)The version of Laravel I used for this project is Laravel 5. There are many benefits to using Laravel when developing a web app, mainly the fact that it handles many aspects of web development that are trivial or may take a long time to implement, such as authentication, templating HTML pages and developing routes.

* 1. Model View Controller

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

Laravel uses a Model-View-Controller Design(MVC). The model part of MVC refers to the shape of data your application uses and operates with. For example, if you have a table in your database containing user types, this is a model.

The controller aspect of MVC interacts with the model part. When requesting to view a page, the controller interacts with the model and retrieves the relevant information to be displayed. If a user wants to add a new item to a table, the controller updates the model.

Once the controller retrieves the info, it constructs a view for the info. This is where the view component of MVC comes in. The view is a template that the model can be linked to and displayed and it can also be manipulated by the controller. The view consists of a web application’s HTML components.

* 1. User Authenticaion

Explain how user authentication is implemented in the web application framework.

Laravel uses controller in order to implement user authentication in a web application.

* 1. Routing

Describe the routes that were defined in the web application

* 1. Templating

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

Laravel uses the Blade templating engine. Blade breaks HTML code down into pieces to be managed by the controller. Using blade, users of Laravel can quicky style their web application to their liking in an intuitive manner.

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. Navigation

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

* + 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 |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |

## Discussion of Functional Testing Results

Describe the results from the tests. Address any functionality where unexpected behavior could not be debugged.

* 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.

This project was managed by breaking it down into sections, based on different aspects of designing and developing my laravel web application. The very first step of the project was deciding on an idea for the web application, from then on I set out to develop the web application and detail how the project was managed throughout the course of the development. I decided to design and develop a website which allows users to buy/sell Muay Thai kickboxing gear. The next stage of the project was system design. I describe the internal workings of Laravel and started developing some of the back-end of the site.

## Project Phases

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

### Requirements

Next was requirements gathering, in this section I looked at 2 similar applications, vinted and depop. This research helped in designing/developing my website as it inspired me and gave me an idea of what the website needed.

### Design

Next was the design aspect of the website, in this section I specify all the components which make up the design of the web application and started developing the skeleton of my design.

### Implementation

### Testing

## SCRUM Methodology

Sprints

## Project Management Tools

### Github Project

Github Project was an essential tool in managing my project. This was very helpful for me as it gave me clear objectives to focus on and it made completing the project easier as it broke the project down into manageable parts.

Include screen shots

Githubs Project feature allowed me to organize and priotitize all the different components of my project. I used a Kanban board in order to manage my project. This board is split into 3 parts. To do, In progress and Done. I then added and moved around cards in each of these sections as I made progress on my project. This was extremely useful as checklists could be made, breaking down big tasks into smaller tasks.

### 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.