

Software Project

**Aaron Saula**

**Your student number n00201123**

Software Project

CA 2 – Develop a PHP shopping cart website

Year 2 2020-21

DL836 BSc (Hons) in Creative Computing

Table of Contents

[1 Introduction 1](#_Toc95917146)

[2 Business Concept 2](#_Toc95917147)

[2.1 Business Idea 2](#_Toc95917148)

[2.2 Business model 2](#_Toc95917149)

[2.3 Market Research 2](#_Toc95917150)

[2.4 Marketing/Advertising 2](#_Toc95917151)

[2.5 Suppliers 2](#_Toc95917152)

[2.6 Competitors 2](#_Toc95917153)

[2.7 Employees 2](#_Toc95917154)

[2.8 Environmental Impact 2](#_Toc95917155)

[3 Requirements 4](#_Toc95917156)

[3.1 Introduction 4](#_Toc95917157)

[3.2 Requirements gathering 4](#_Toc95917158)

[3.2.1 Similar applications 4](#_Toc95917159)

[3.3 Requirements modelling 5](#_Toc95917160)

[3.3.1 Functional requirements 5](#_Toc95917161)

[3.3.2 Non-functional requirements 5](#_Toc95917162)

[3.3.3 Use Case Diagrams 5](#_Toc95917163)

[4 Web application Design 6](#_Toc95917164)

[4.1 Layout 6](#_Toc95917165)

[4.2 Interaction 6](#_Toc95917166)

[4.3 Colour schemes 6](#_Toc95917167)

[4.4 Font choices 6](#_Toc95917168)

[4.5 Wireframes 6](#_Toc95917169)

[5 Database Design 7](#_Toc95917170)

[5.1 Description 7](#_Toc95917171)

[5.2 Business Reporting Requirements 7](#_Toc95917172)

[5.3 Textual Representation of Data-Set 7](#_Toc95917173)

[5.4 Business Rules 8](#_Toc95917174)

[5.5 Entity Relationship Diagram 8](#_Toc95917175)

[5.6 Tables 9](#_Toc95917176)

[5.7 Database Dictionary 10](#_Toc95917177)

[6 System Design/ Architecture Overview 11](#_Toc95917178)

[6.1 Introduction 11](#_Toc95917179)

[6.2 Model View Controller 11](#_Toc95917180)

[6.3 User Authenticaion 11](#_Toc95917181)

[6.4 Routing 11](#_Toc95917182)

[6.5 Templating 11](#_Toc95917183)

[7 Testing 12](#_Toc95917184)

[7.1 Introduction 12](#_Toc95917185)

[7.2 Functional Testing 12](#_Toc95917186)

[7.2.1 Login/Registration 13](#_Toc95917187)

[7.2.2 Navigation 13](#_Toc95917188)

[7.2.3 Calculation 13](#_Toc95917189)

[7.2.4 CRUD 13](#_Toc95917190)

[7.3 Discussion of Functional Testing Results 14](#_Toc95917191)

[7.4 User Testing 14](#_Toc95917192)

[7.5 Conclusion 14](#_Toc95917193)

[8 Project Management 15](#_Toc95917194)

[8.1 Introduction 15](#_Toc95917195)

[8.2 Project Phases 15](#_Toc95917196)

[8.2.1 Requirements 15](#_Toc95917197)

[8.2.2 Design 15](#_Toc95917198)

[8.2.3 Implementation 15](#_Toc95917199)

[8.2.4 Testing 15](#_Toc95917200)

[8.3 SCRUM Methodology 15](#_Toc95917201)

[8.4 Project Management Tools 16](#_Toc95917202)

[8.4.1 Github Project 16](#_Toc95917203)

[8.4.2 GitHub 16](#_Toc95917204)

[9 Reflection 17](#_Toc95917205)

[9.1 Your views on the project 17](#_Toc95917206)

[9.2 How could the project could be developed further? 17](#_Toc95917207)

[9.3 Assessment of your learning. 17](#_Toc95917208)

[9.4 Completing a large software development project 17](#_Toc95917209)

[9.5 Technical skills 17](#_Toc95917210)

[9.6 Further competencies and skills 17](#_Toc95917211)

[10 References 18](#_Toc95917212)

# 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

The business idea will be a social media/clothes website where you can be able to buy fresh

and new clothes online.

## Business model

Best deals on the main home page. We will try to make categories trend able.

## Market Research

Market for Product/Service

Customers - Demographics, Profile

## Marketing/Advertising

The customers will be the main target marketers. It depends how much they rate the

website and if so then twice a week. To make online shopping easier and quicker to buy.

## Suppliers

## Competitors

JD, IKEA, Penneys, Sports Direct are our main competitors for the website and also similar

have very similar websites to us. For each websites they have way too many categories and

to shipping options takes longer to get.

## Employees

## Environmental Impact

# Requirements

## Introduction

The purpose of the requirements phase is to allow for developers to work out what the application should be able to do. It is important to understand what the users would like the application to do rather than the developer deciding what is required.

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

## Requirements gathering

### Graphical user interface, text, application, email Description automatically generatedGraphical user interface, application Description automatically generatedSimilar applications

This is a home page (left side) and checkout page(right side) for eBay. It’s a similar website to my design website.

**Advantages:**

* Has the search bar which makes it much easier for the customer to find what they want on the website?
* Deals and offers for the top bit of the page so customers will be intrigued to see the best offers.
* The shipping price for the checkout page is free. Most sites its extra bit of money for shipping.
* For customers you can receive for your item when trying to sell something

**Disadvantages**

* Too many categories which will get customers fed up eventually
* Website gets updated so often

## Requirements modelling

### Functional requirements

* Log in and register your account
* Whenever you’re in a page for a website there should be a back button
* Admin/User Authorization

### Non-functional requirements

These are requirements which if not met do not stop the application from working, but which mean that the application is not working as well as it should. They are usually based on issues such as:

* Usability
* Performance
* Security

### Use Case Diagrams

Graphical user interface, application

Description automatically generated

# Web application Design

## Layout

The layout for the web application Bootstrap, CSS and Laravel

## Interaction

For the homepage when you click on an item it’ll go on to the next page (product list page). Each product you click on the first page will go on to the product list page. There will be a back button if you don’t want to be on that page anymore. On the product list when you click on a product item it’ll lead to the detail page of it.

## Colour schemes

Have a blue colour background for the web page.

## 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.

## Wireframes

For the homepage it’ll have the name of the website on the top center of the homepage. After that it’ll have the navbar for the list of categories. It’ll also have the search bar to look something up for the item you want to search

The layouts will have the main relevant items for the homepage.

There’ll be another page after the search result or clicking on one the navbars. On that it’ll have the list of items.

When you click on one the items it’ll view the detail of the item. When you want to click on the pay item it’ll go on the checkout page to complete the purchase of the item

Chart

Description automatically generated Table

Description automatically generated Chart, bar chart

Description automatically generated Graphical user interface, application

Description automatically generated

# Database Design

## Description

A company has a website that sells video games for different consoles. They would need a database for all their games and order places. For each order place, they would need games bought, total price, date of the order, and how long it will take to deliver. The database needs to keep track of all games 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: festivals, stages, shows, performers, and genres.
2. Users will need to be able to find all festivals ordered by their start date.
3. Users may want to find a festival by a specific start date.
4. Users need to find all festivals using a list of genres.
5. Users need to find the stage for a specific show.
6. Users need to find the shows using a performers name.
7. Performers may need to find the list of festival contacts.
8. Users need to find festivals by location and the location needs to be displayed on a Google Map
9. User may need to find festivals by city
10. Users need to find stages within a festival by the stage’s location
11. Organisers need to display a list of employees that are assigned to a specific festival

## 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 **Festival** has many **Stages**.

 A **Stage** belongs to one **Festival**.

 A **Stage** hosts many **Shows**.

 A **Show** is performed on one **Stage**.

 A **Performer** can perform in many **Shows**.

 A **Show** is performed by one **Performer**.

 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



## Tables

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



## 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

* 1. Model View Controller

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

* 1. User Authenticaion

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

* 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.

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.

## 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.