

**Software Engineering and Testing. BSC Year 2, 2020/2021**

**(Assignment 3 - 20%)**

**Assessment 3: Design and Draft Implementation**

**Submitted by: Names, Student numbers**

**Submission date**

**Declaration**

I herby certify that this material, which I now submit for assessment on the programme of study leading to the award of Ordinary Degree in Computing in the Institute of Technology Blanchardstown, is entirely my own work except where otherwise stated.

Author: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Dated: \_\_\_\_\_\_\_\_\_\_\_\_\_

Author: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Dated: \_\_\_\_\_\_\_\_\_\_\_\_\_

**Table of Contents**

# Title: Stressfree Pets

# *Abstract / Executive Summary (200 Words max)*

The below document will outline the methodology, structure, and development plan of the pet service website we are building. Using PHP, CSS, HTML, and MySQL, we aimed to build a system that provides a user-friendly website that will allow them to book our services or allow businesses to offer the services. The website contains a booking system, a login/ sign up system, a review page and a contact us page. This document will break down the website’s structure, ERD design, UML design, user interfaces and use cases.

# Project Definitions (Steven)

* The purpose of this document is to give and overall outlook and design structure of our website. Below you will see the design and implementation of the website including UML design, ERD design, database implementation and general outline of functionality of the website.
* The project we are designing is a front-end website for a local pet walking, pet sitting and daycare business. Users will be able to log in, book the service they like or leave reviews. Business operators will be able to manage times and bookings. The administrator will have full privileges of the entire website.
* The website has a few elements to get it to work, but our main priorities are ease of use and simplicity for the front end. We believe this is important as the user needs to have a clear understanding of the services provided and the layout of the site. Also, for ease of use we have databases connected to the system to manage user and other site data. We are using MySQL to write the database, PHP, CSS, and HTML are handling the scripting and front end of the website.
* The software system has 2 main components, front-end and back-end. The front-end is a simple design. It has three squares on the homepage which will bring the user to the area they would like (those being pet sitting, daycare and walking). When the user picks the service they would like they will be brought to a booking page of that service. From the business end, they will be able to view their bookings and edit their available slots which site users with an account are able to see.

# Document Revision

Rev. 1.0 date – initial version

# Methodology

System models – UML (Habiba)

Use of, and necessity of OOAD (Piotr)

Purpose of using classes / What is a class diagram? (Rochelle)

Static Versus Dynamic Case Diagrams? (Rochelle)

What is an ERD? (Piotr)

Purpose of using classes? (Habiba)

Volatile versus Persistent storage – Object Instances / Database? (Steven)

* Volatile storage refers to storage that retains data if the system is turned on. Persistent storage will retain data when the system is turned off. Most of the data on the site will be stored using persistent storage. We have a database built for the site which will contain information on users, administrators, the services provided etc. This is essential to the functionality of the system as it provides scalability and persistence to the website. A way to store data always is essential to a system like this as users need to be able to access and view their data whenever they like. If this data goes down if it was in volatile storage the functionality of the website would come into question. The data constraints and foreign key relationships ensure data will be updated correctly in the appropriate tables in our database.
* As stated in the project definitions, the main goal of the system was to create a simple but functional layout and I believe this has been achieved. A screenshot of a computer

  Description automatically generated

Above is the home page of the website. This is very effective in functionality as it gives users a clear and defined idea of the sites purpose and provided services.

A person walking dogs on leashes

Description automatically generated

A screenshot of a computer screen

Description automatically generatedA screenshot of a computer screen

Description automatically generatedAbove is what a user will see when they select one of the services (the other service pages are nearly identical). It is a very simple page just giving the user a small description of the service and a button to a page that allows them to book the service.

Above is two images, one of the user signup page and the other is the business signup page. The left image is the user signup form. This form authenticates the users data and allows them to create an account which will give them the access to book services, leave reviews and more.

The right image is the business signup form. While similar to the user signup the additional information required is the service they will be offering and proof of certifications/ qualifications. These will then be stored in the table for the business providers.

The simplicity of the UI is important to functionality and user use. This also assists us in data management as it allows our databases to have a relatively simple structure and make it easier for management.

User Interface template chosen and how it can aid in executing the functional specification of the project. (Steven)

1. **Requirements (Habiba)**

4.1 Use Cases

4.2 Use Case Specifications

(Specifically – how use case specifications have been used as a means to develop the ***classes/attributes/methods*** and database ***tables***)

1. **Case Diagrams (Rochelle)**

**Class Diagram** – Show all relationships, multiplicities, associations, generalisations (inheritance), aggregations (compositions) - See lecture 4.

Paragraph to explain ALL design decisions.

**Entity Relationship Diagram** – Show all relationships, multiplicities,

**Conceptual Diagram**:A diagram of a company

Description automatically generated

**Physical ERD of DATABASE**:

A diagram of a computer

Description automatically generated

Paragraph to explain ALL design decisions.

# Conclusions (Piotr)

Your conclusions and recommendations as to how far the project has progressed.

Your changes to the original proposal that the design has revealed and necessitated.

Additional sections: Table of Contents, Table of Figures, References, Index

Checklist: Is your document complete and correct?

*Content:*

* Does the design include all requirements from the customers’ needs
* Are you satisfied with all parts of the document?
* Do you believe all parts have been implemented?
* Have you explained your methodology and design choices?
* Have you clearly articulated your understanding of the purpose of all diagrams created ?
* What are these diagrams? Why you need them? How were they developed?
* Is each part of the document in agreement with all other parts?
* Does the design create a solution for the initial proposal?

*Completeness*:

* Are all the necessary components specified?
* Are the design specifications precise enough?
* Are all sections from the document template included – if changed, why?

*Clarity*:

* Is the design reasonable?
* Is the level of details for each design section appropriate?
* Is the design written in a language appropriate to the intended audience of software engineering teams?
* Are all items clear and unambiguous?