Website Development

Individual Report

Hugh Culling

20206905

Level I

2021 – 2022

# Table of contents

# Introduction

This website was created to help Ace Training keep track of the students that sign up to their courses, it does this by using an authorisation system so that not anyone can become apart of a course, they have to first be authorised by the tutor of that course, who, in turn, will have to be authorised by an admin, who, in turn, will have to be authorised by the controller of the database. It also adds other features which allows tutors to create any number of courses, view the courses that they’ve created and also allow students to view the courses that they’ve signed up for, this would act as an effective tool to prove that a student was part of a course as they’d need their user ID and password to log in to show that they’d been authorised by a tutor.

# Website Design Principles

## Legality Issues

### Copyright

This website doesn’t infringe on any copyright laws as it doesn’t mention any other companies or use their branding, slogans or products in any way. As well as this, the programming languages used: HTML, PHP, SQL and CSS are all open-source languages which means that they can be used freely.

### Data Protection

The data collected in this website is purely for the use of this website, it won’t be shared or held unnecessarily in the context of the website. Cookies also aren’t used so there’s no need to ask for the user’s permission. As of yet, however, there is no option for the user to view the data held on them or delete this data, which is necessary in order to comply with data protection laws.

### Disability and the Equality Act 2010

This website was tested on Microsoft Edge which offers a ‘Read-aloud’ tool to enable blind users to use the pages of the website, it has been tested and works effectively.

### Privacy Policy

The data collected by this website include a user’s: forename, surname, email, password and which course(s) they are taking or teaching, the purpose of this collection of data is purely to aid in the database management of the website as these data values offer a unique identification of an individual, particularly an email address which is impossible to replicate (have two emails which are identical). There are no functions in place for any student, tutor or admin of the website to view anyone’s sensitive data (i.e., email or password).

## Professional Issues

The choice of language for this website is English. The website is secure using an ID and password system, in which both of these are needed in order to view and amend your information. It also establishes a hierarchy in which a student must be authorised by a tutor, who in turn, must be authorised by an admin, who in turn, must be authorised by the database controller.

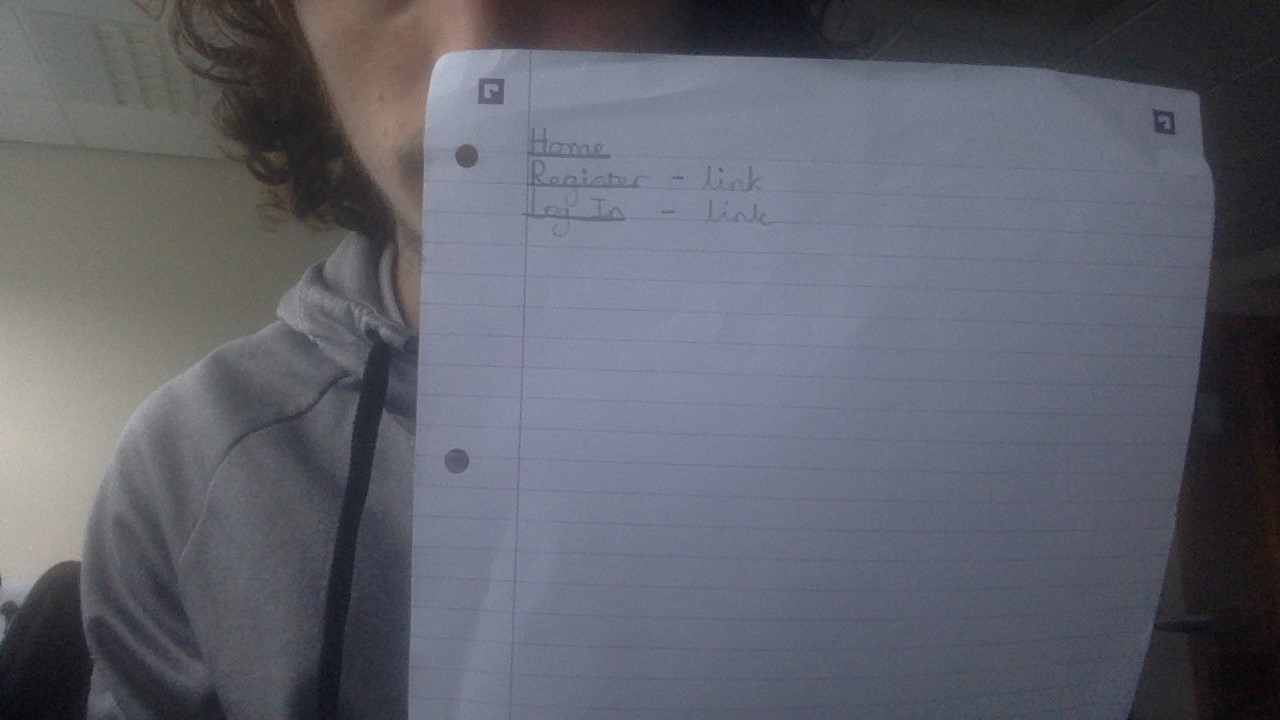
### Usability Issues

The website could have been subject to criticism for not catering for blind users, however, because of Microsoft Edge’s ‘Read-Aloud’ system, blind users are able know what information fields they are required to fill in.

### Design and Implementation

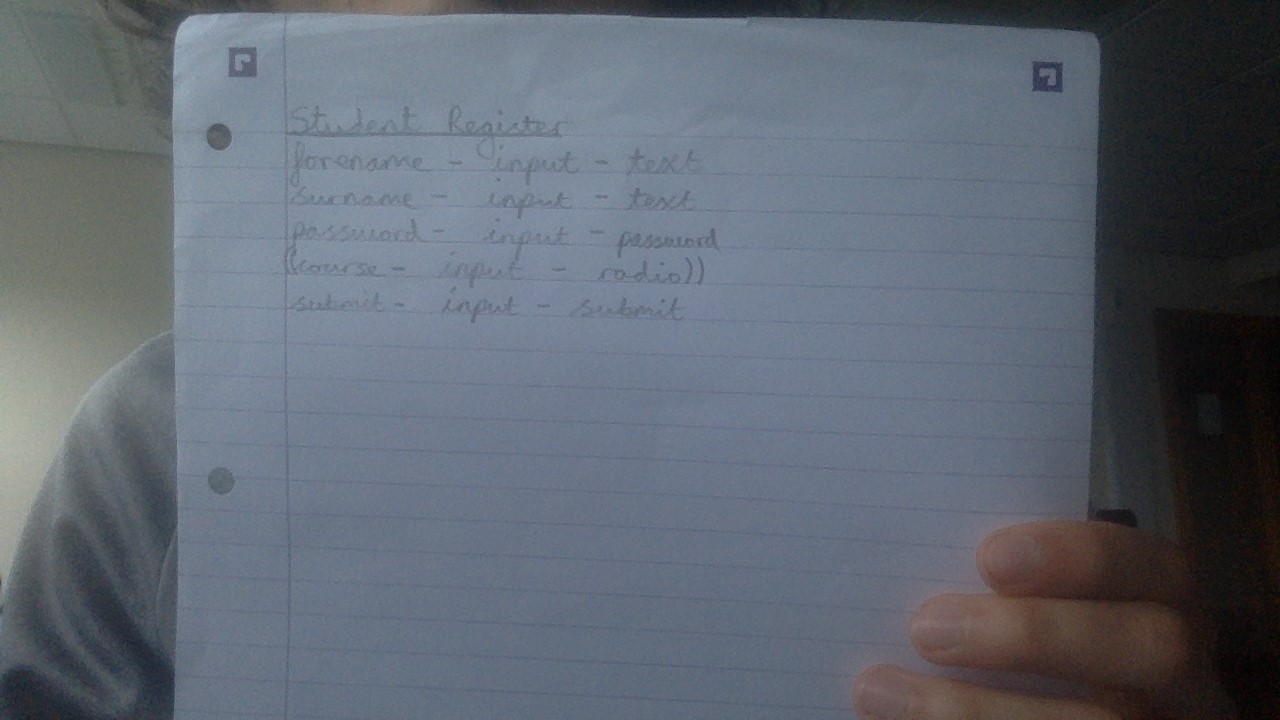
This website was designed following the specification, aiming to first complete the ‘Satisfactory Website’ specification and then moving up. The webpages were designed on pieces of paper and then implemented one-by-one, this decision was made after various attempts to create the website without any planning which kept on leading to difficulties and failure, this approach meant that I could focus on getting one function to work before focusing on other functions. It also meant that when going through the pages one-by-one, that if a page required another function to already be in place, for example, the page that comes up when a student who has been authorised logs in required a function for the tutor to first be able to authorise the student. Below are some examples of these pages, which can be seen as an alternative to storyboards.

#### Home page

This home page shows how links are represented:

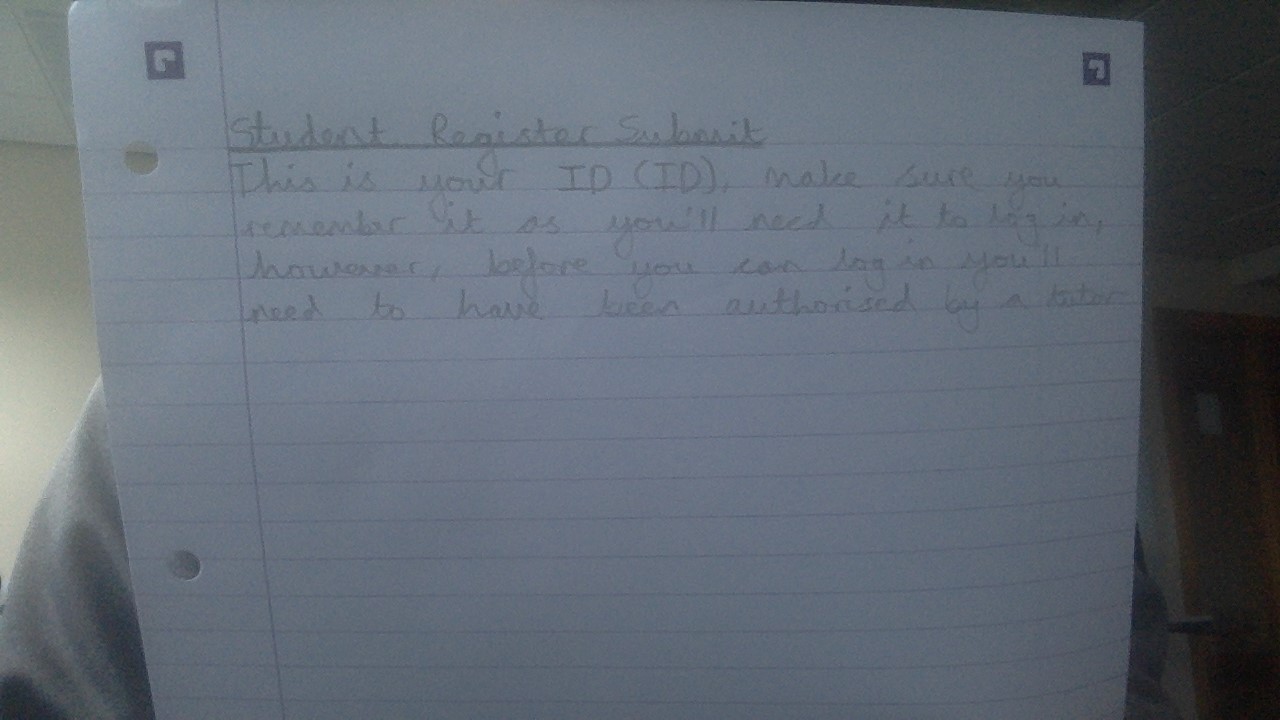
#### Student register

This student register page shows how inputs and how a list requiring a database query are represented, the latter being surrounded by two pairs of parentheses:



#### Student Register submit

This student register submit page shows how a single variable requiring a database query is represented, by a single pair of parentheses:



In total there are twenty-four pages which each show the design blueprint for the webpage and use the grammar shown above to explain the desired functionality.

The implementation varies in a particular way, for example, when the student logs in, are authorised, they are greeted with a link where they can view their courses, in the design this leads straight to the page with the name of the course which they have signed up for, however, in the implementation it will lead to another log in page as they need to post their details again so that the webpage can retrieve their information.

The implementation transfers these on-paper designs onto webpages using HTML to provide content such as titles for the webpages tab, links to take the user to a new webpage and a way to display text. CSS was used to add borders to the tables, PHP was used to connect to the database to add information from registration forms, retrieve information of students or tutors who hadn’t been authorised so it was possible to authorise them and also to see whether a user is allowed to log in and have access to certain privileges. For example, a student can’t create a course, only an authorised tutor can.

The implementation of the databases was so that if they sign up as an admin or a tutor they are required to enter their forename, surname, email and password, this information is then added into the users table of the database which has fields: ID, forename, surname, email, password, type and authorised. The tutor, once authorised can then create courses, when they’ve created a course, the course name gets added to the course table which has an ID which is a primary key and set to auto-increment. On top of this, the tutors ID and the courses ID get added into a table called ‘tutorteaching’, this is necessary as it allows a tutor to create multiple courses without adding null fields to the user table and having to accommodate to however many courses a tutor creates, they could create fifty, and so there would be fifty extra fields, named ‘tutorcourse1’, ‘tutorcourse2, etc. Furthermore, the Admin and students wouldn’t be using these fields, so this is a suitable alternative, furthermore if a tutor wanted to create fifty-one courses then it would require a whole restart of the database. By using foreign keys in this table we are able to link information from other tables such as the course and user table to retrieve information through the use of an inner-join. A similar approach is used when a student signs up for a course, they are added to a table called ‘studenttaking’.

### Promoting the website

The code uses the meta tags provided Sublime text when you enter the ‘<html>’ tag, these are ‘<meta charset=”utf-8”>’ and ‘<meta name=”viewport” content=”width=device-width, initial-scale=1”>’.

Search engine optimisation is achieved through using a simple website name: Ace Training.

### Conclusion

Upon reflection the website does achieve certain specified requirements such as students being able to sign up for a course, allowing the tutor to create and view their courses and implementing an authorisation system, it however, lacks in that it doesn’t use session variables which would improve the user experience as it would provide a secure way of ensuring that a certain user is using the system without having to log in every time that they want to use some functionality. It also doesn’t allow a tutor to upload resources for a student to download, meaning that the website has little use beyond an identification system.