# Task 3B – Evaluation

To achieve a higher grade, you need to:

- Provide more specific examples from user testing feedback
- Include a more detailed analysis of the ethical implications beyond data protection
- Strengthen the evaluation of content selection against the original client requirements

In this document, I will be going over the following content:

- Assessing
    - Assets and content
    - Sources of information
    - Legal and ethical
- How well the solution met
    - Functional and non-functional requirements
    - KPIs (Key performance indicators)
- How to improve

## Assets and content

Using illustrations is very important when creating a website, so I made sure to include many images and graphics throughout. I used a website called “Unsplash” for high-resolution, professional photos. I used many of these, at least one per page. Additionally, I used an additional website called “FontAwesome” for icons and graphics, which I used for buttons and links, etc.

I used these two websites for the following reasons:

- Professional
- High-quality & customisable
- Free of charge
- Non-copyright
- Lots to choose from

Unsplash images are high-quality but were compressed to reduce page loading times. FontAwesome uses SVGs, which means they are essentially HTML code, leading to minimal performance impact (they can also scale infinitely without loss of quality).

When it comes to content, I didn’t do as much as I initially had intended. Consequently, pages felt empty, unfinished, and uninviting.

If there was more information and writing on the pages, the website would feel much more complete and engaging. However, I believe I made up for this loss with functionality features. The content can be added in the future very easily, whereas functionality is more difficult to implement.

For the content that was included in each page, I gathered inspiration from websites which I had described in the research portion of the application. Websites such as Whipsnade Zoo.

## Sources of information

Gathering information is vital to the success of the project, so I made sure to do lots of research both before, during, and after the development of the application. However, I was also aware that there is lots of false information on the internet, so I had to be cautious about where I got my information from and which sources to trust.

Here's a list of sources I decided to trust:

- Stack branded sites
    - Stack Overflow
    - Stack Exchange
    - Etc.
- GeeksforGeeks
- LeetCode
- Medium (certain topics)
- YouTube
    - Making sure to only trust verified channels or channels linked directly within an organisation.
- Whipsnade Zoo
    - Example of a successful zoo website.

Some of these sites allow the posting of answers by any user. Despite this, answers are still regularly checked by reputable users and site representatives. This constant moderation means that answers are almost always factual.

## Legal and ethical

Whilst designing and developing the site, there were many ethical and legal attributes that had to be considered. Here are some of the legalities that I had to keep in mind throughout:

- GDPR
- Data Protection Act
- Copyright
- More

When storing user data, it is extremely important to keep data secure not only to keep in line with these guidelines but also to keep customers happy and trusting of our platform. There are multiple ways to make sure that user data is kept securely within the system.

Firstly, I made sure that I was hashing and salting my users' passwords. This means that if someone was somehow able to access the database, they wouldn't be able to view passwords, only the hashed version of it. I did this using a library called "bcrypt", which could automatically hash, salt, and check passwords.

I didn't, however, manage to implement other things to help protect user data. For example, I could have made an account locking system, in which if someone attempted to login to an account more than x number of times, it would lock that account and only be unlocked when the email associated with the account clicked a link. This would protect against brute-force attacks (which is where an attacker keeps entering different passwords until it is correct).

As for Copyright, I made sure not to use any libraries or assets that were protected under licenses stating that I was not able to use them (this is why I opted for Unsplash, which provided images completely free to use).

## Key performance indicators (KPIs)

I have made tables for technical and business key performance indicators, along with their meanings and why I have/haven't met their criteria or completed it.

### Technical

| KPI name | KPI meaning | Why/why not? |
|---|---|---|
| Website traffic | Measures the number of visitors and how often they visit. Higher traffic indicates the site is discoverable, visible, and being effectively marketed. | I think this KPI still needs to be tested as the application is yet to be publicized and marketed. |
| Page load speed | Measures the time it takes for the website to load. Faster load times improves the user's experience and keeps the user on the site for longer. Slow times push users away very quickly. | This KPI has been very successful. When testing, the load times are very quick, and there have been no direct complaints relating to the speed of the application in user feedback. |
| Uptime | The amount of time that the site has spent being online and operational. High uptime allows users to access the website consistently, allowing users to rely on the system. | This KPI has also been very successful. The application has been stress tested with thousands of fake users and requests, and it has stayed up regardless. |

### Business

| KPI name | KPI meaning | Why/why not? |
|---|---|---|
| Registration rate | The number of new registrations that have been made on the site. Gives a good indication of how good the site is at convincing users to sign up and generate leads. | From user testing, this KPI has been successful. Users are easily able to make their way to registration and create an account.<br>Further testing after the applications release would produce better results. |
| Activity engagement | The number of people that are booking into events or activities | From user testing, this KPI has been successful. Users are |

| | | |
|---|---|---|
| | in the zoo via the website. Tracks which events are most popular, and which drive engagement and revenue. | easily able to make their way to bookings and create a booking. Further testing after the applications release would produce better results. |

## Functional and non-functional requirements

I have made tables for functional and non-functional requirements with a description for each as to why I did or did not meet the requirement.

| **Functional requirement** | **How I did/didn't meet this requirement** |
|---|---|
| Account System | Users can create and login to accounts. They are also able to manage their account and view their own details. Additionally, sessions worked perfectly allowing users to stay logged in across pages. |
| Booking System | Users can make bookings using a form easily. Additionally, they can view all the bookings they've made from a singular page. |
| Hotel System | I didn't have enough time to implement this feature, unfortunately. |
| Help & Information | I didn't have enough time to implement this feature, unfortunately. |
| Educational Resources | I didn't have enough time to implement this feature, unfortunately. |
| Accessibility System | I added a simple widget to the website which added a button for users to click on in the bottom left-hand corner. Through this widget, users could change font-size, font-weight, family, and way more. |
| Loyalty & Reward (Membership) System | I didn't have enough time to implement this feature, unfortunately. |

| **Non-functional requirement** | **How I did/didn't meet this requirement** |
|---|---|
| Accessibility | I added the accessibility system which provided a widget for users to use. This made the website very accessible for all users.<br>Additionally, I added extra CSS which altered how the website looked for differently sized devices, such as phones vs desktops. |
| Security | I made sure to encrypt data that flowed throughout the application. For example, when authenticating users, I hashed & salted passwords to ensure they were unreadable to the human eye. Additionally, I used certain HTTP methods (POST) so the data being transferred was hidden. |

| | |
|---|---|
| Scalability/Maintainability | I used Object Oriented Programming (OOP) foundations and practices to ensure my work could be easily expanded and developed on further by future developers. |
| Ability to meet KPIs | I think most of the KPIs have been met sufficiently and this non-functional requirement has been met. |

## User testing feedback

Based on the feedback received from user testing, the functionality of the application is fine (besides the missed sections). They also mentioned adding more content and improving the overall accessibility of the website. To do so, I may consider adding more options to customize the site.

## How to improve

Overall, I think there are a couple of areas which need improvement.

Firstly, maintainability is something that should be improved for future iterations. I didn't include enough comments in my code, in most areas. Some bits of code are well documented and explained. If a new developer were to take over the project, they may struggle to understand what certain bits of code does. I should consider adding much more comments to explain particularly confusing functions etc.

Additionally, I need to be trying to include more functionality. I did do a lot of it but also missed quite a lot too. For example, the user accounts system and booking system are both finished, but I missed the L&R Scheme (Loyalty and Reward Scheme) system, hotel reservations, help and education systems. If I had managed to finish these off, I think the result would have been a lot better. The code is well organised though, so adding further functionality (like aforementioned) shouldn't be too difficult for a talented coder.