Software development plan

We have been contracted by GibJohn tutoring to provide them with a digital solution that will provide interactive teaching and learning within a range of subjects, provide access to digital content to encourage eider learning and support assessment and monitoring of learner progress. The client has also suggested implementing potential features of collaborative teaching and learning tools, accessibility features, a learning rewards scheme and gamified learning. Within this report, I will discuss the plan we have developed to create this system and meet deadlines.

## Project Organisation

We will organise this project by allocating all team member with separate roles so that work can be completed simultaneously. The team leader creates tickets on linear for each task, and through meetings, a team member is assigned to each task. This helps to organise tasks and ensures all team members understand what their responsibilities are. To meet the time scale, we will ensure to work at a fast past and prioritise an efficient working environment.

## Project Scope

The project scope is to meet all client and user requirements in a timely manner and to an excellent standard. We will also have to meet other standards as well, such as security and performance. To meet security requirements, we will implement a user authentication method using a sign-up system and potentially 2FA. To ensure the performance of the website, we will use a responsive framework for the user interface (React). To ensure the efficiency of the development, we will use tailwind to prevent writing lengthy CSS code. On the back end, we will use a SQL database for structured data storage and data accuracy.

## Development approach

The software development life cycle we will use is agile, to ensure flexibility and fast-paced development within the project. To do this, we will split the development up into small sprints, and review after each sprint. Every Thursday, we have a team meeting to discuss where each team member is at and what steps need to take place moving forward. This helps to ensure that the team is staying on track and prioritises productivity.

## Work breakdown structure

We use the computational thinking method of decomposition to break down each big task such as lengthy reports into smaller tasks. This helps to prevent confusion and pressure from large tasks and helps to ensure workflow. Additionally, each team member can work on a different task simultaneously which enhances efficiency. Tasks are each logged on linear with a different level of priority to ensure high-priority tasks are completed first.

## Timeline and Milestones

In order to meet deadlines, we have set out a timeline including milestones to meet. The first milestone involves planning the solution, where the team defines the scope, target audience, and core features such as online booking, tutor profiles, and access to learning resources.

Next, during design and prototyping this is where a design of the solution will be displayed. A prototype will then be produced and refined based on user feedback to guide development.

After the design stage, core development will begin, and this is where we will begin to add features based on client needs and user experience. This will result in a functional Minimum Viable Product (MVP) for the client to test and review.

We will then reach the testing and refinement phase, usability and performance testing will be carried out, with feedback used to fix bugs and improve responsiveness and accessibility for users.

Finally, the deployment and evaluation milestone will involve launching the website, monitoring performance, and reviewing outcomes against project objectives during this time as well the websites traffic will indicate if the solutions has worked and has been a success for the client.

## Quality Assurance Plan

During development of the solution, it will need to be regularly evaluated by the client and end users, this is to gain information based on how well the solution looks, functions and works with both the users and client, for example when these regular checks/tests are carried out we will require the user to fill out a form what will have questions such as how satisfied they were with the solution and what we could improve on or make different, with this information we are able to see how satisfied the client and users are with the solution. Once the satisfactory percentage has reached over a certain point for example if 8 out of 10 users are satisfied then that will reassure that the solution is working as intended and meets the needs of the users, this shows that there may be no more improvements to be made, and the solution can be deployed.

## Risk Management

Using the agile methodology the risks will be reviewed at the end of each sprint and then evaluated to see if they will cause a problem.

Key risks and mitigations include:

* Time Constraints: the project may take longer than initially intended Mitigation: effectively plan out sprints in order to set a realistic timeline for the solution
* Team Communication: Miscommunications will lead to confusion and uncompleted tasks

Mitigation: hold regular meetings where the team can asses where they are in the project and make any changes if necessary

* Unexpected Work: The client may request a feature that was not initially planned in the project but needs to be implemented

Mitigation: Regularly communicate with the client to make sure that the solution is meeting their needs and that they are satisfied with the solution

## Recourse Requirements

Effective resource management ensures that team members, tools, and time are used efficiently throughout the project. Roles and responsibilities are assigned based on individual strengths, with regular communication to balance workloads and maintain steady progress.

## Communication And Reporting

During development and planning the team will use multiple resources for collaboration on projects:

* GitHub: Where the team will upload files and review documents/code for the solution
* Linear: where the team will be assigned to different tasks during the development and research stages
* Teams: the main communication tool used by the team, it will also be used for collaboration on documents that need to be completed quickly or need a lot of information e.g. PowerPoint presentations

## Deployment and Maintenance Plan

The product that we develop will only be deployed to a test system as it is only a prototype. However, with further development, it could be deployed to a live system for the public to use, which would require maintenance. To maintain the site, we could perform regular software updates to enhance security and functionality. We would also store backups of the site to prevent data loss and improve digital resilience. We would also frequently monitor the site to ensure performance and robustness.

## Training and handover

When handing the prototype to the client, we would have to provide sufficient training to ensure that their staff understand how to use the site. We could run training sessions to provide the staff with a hands-on learning experience, to ensure that they develop a good understanding of how to use the product. This is important because it is likely that the staff are not skilled with technology, and if they can’t understand the prototype, then it is no use to them and will not positively contribute towards the business.

## Benefits of the chosen approach

By using an agile software development lifecycle, we can ensure efficient communication between our team and the client. We will break our tasks into smaller tickets using Linear which provides a clear understanding of each task to each team member. An agile approach also allows us to be adaptable to ensure the product meets the requirements as closely as possible.

## Software Configuration Management

The Software Configuration Management tool that we will be using is GitHub. This allows us to upload software components into a central repository which will store the whole project. This allows us to review each other’s code before it is merged into the main branch, enhancing collaboration within the team and ensuring that everybody is satisfied with the development of the project, as we go on.

## Approval & sign off

After each feature has been developed, the whole team will review it and provide feedback. We will do this by leaving comments on pull requests in GitHub. If the team is satisfied, then it will be merged into main. After key phases have been completed, the end work will be reviewed by a tutor, who will provide us more feedback and allow us to make any key changes or upload the end project.