SOFTWARE DESIGN

This document outlines a high-level software design

Brian Mvukwe

Chris Langeveldt

Elizna Horn

Garryk Walker

Jana Van Niekerk

Contents

Contents....1
Software flow....2
Software Design....3
Frontend....3
JavaScript....3
React....3
Figma....3
Backend....4
Python....4
Flask....4
Database....4
MySQL....4

Software flow

The implementation of our software shall begin with a basic “sign in page” where a unique code is inserted to identify as an admin or user, accompanied by relevant error messages. This will couple with a database of users and a connection between frontend and database will be established. These can be considered the furthest reaches of the project and we will build inwards. Once our database is established, more tables will be added to hold the most amount of information, using the minimum space required. We plan on using BCNF or 4th Normal form.

We will then expand on our frontend as well, implementing the question pages, and devise a way to display multiple questions, one at a time, without having to load a new web page for every question. Relevant tables will be added to store both questions and answers from each question.

At this point we need to implement our algorithm to convert this data into information. Perhaps this information could also be stored in a new table. This will increase the size of the database but provide faster results in the future as the algorithm will not have to be run twice (discuss with clients).

Now that we have our information, we need to devise methods of displaying the information. Some suggestions are graphs, charts, matrices, tables, and diagrams. Graphs would most likely be the most accurate and easy to interpret. The point of this aspect is not to stand out, but rather to make sure the user understands the information being provided to them, simple displays are most likely going to be the most effective, however this does not mean they cannot be colourful!

The final stage will be downloading the information as a pdf.

Once the project and the most important features are completed, we can work through the project once more adding branding and user-friendly techniques such as prompts and hints. Once the minor features are added, the project can move into its “maintenance phase”.

Software Design

As requested, our app will run as a web-based application. For the user, this will be accessed through a link on the main website. Our software will have a JavaScript (React framework) frontend, a Python (Flask framework) backend, and a MySQL database.

Frontend

JavaScript

We have decided to code frontend functionality using JavaScript. JavaScript not only lets you build a highly interactive web-based application, but it also helps in enhancing the speed, performance, functionality, usability, and features of the application without any hassle. JavaScript is also cross browser compatible, and it comes with a variety of libraries and frameworks. It also has the added benefit that the development team are already familiar with it.

React

For our client-side system, we are opting to utilize the high-level framework for the JavaScript programming language named React. React is a JavaScript built-in library, that allows the ease of visual manipulation. React houses a large and versatile library of pre-designed visuals, along with their open-source software. Allowing us to customize these visuals to suit our visual needs. React stood out as it has one of the best libraries for creating encapsulated components that update and render in real time making merging straight forward. React makes it simple to implement the cascading style sheets, fondly referred to as CSS. Also, the development team are already familiar with it.

Figma

Figma is a design tool that can help developers create anything from websites and applications to logos. For this project our team will use it to design the User Interface and design a unique user experience. Figma can be used by joining with a free account. Figma can also be used in conjunction with React. All that is required is to install the plugin, is FigAct. Depending on the preference of the team and the client and the availability of time, we will consider using Figma for developing the user interface.

Backend

Python

Python is a simple but powerful language that runs smoothly on the most popular operating systems (Windows, Linux, and Mac). Python is an adaptable, versatile, and highly efficient programming language that offers dynamic typing capabilities. Python also has great functionality when it comes to data handling/manipulation. This will be extremely helpful when we work with the data from user responses.

Flask

Flask is a web framework written in Python. It is very explicit and easy to read. It is highly scalable, flexible, modular, and lightweight. The development team also have experience using Flask. All these reasons made the choice of API language quite simple.

Database

MySQL

MySQL is a relational database management system (RDBMS). It stores records in multiple, separate, and highly codified tables rather than a single repository. After thorough research we decided to use MySQL because it is simple, easy to set up and use, fast, trustworthy, and well-understood. MySQL is scalable, flexible, has high performance and strong data protection. We also have prior experience using MySQL.