INTUITIVE QUESTIONNAIRE HUB

Have you ever needed to make an important decision and spent hours, weeks, or even years researching and contemplating due to indecisiveness or fear of making the wrong choice? I experience it all the time. I'm confident that I will be moving out of the United States when I graduate; the question is where I want to move. It's a big decision, and so many factors are involved; it was difficult to pinpoint the places that are the best match for me and required months of research. When I thought about it, I realized I come across this problem often on a smaller scale. For example, when buying a new laptop or new headphones, or something expensive, I usually spend hours researching my options and their pros and cons before deciding on the best match for me.

So what is the solution to minimize the time spent on difficult decisions? Automating is certainly one option. Imagine an app with simple algorithms and a framework that makes it easy to create a questionnaire that determines your best option. This framework would make it easy to push as an open-source project that anyone could utilize. The framework also makes it easier for a team to work on because every questionnaire will use the same general graphical interface and user experience. So each person could research and work on questionnaires for subjects they are particularly interested in. I am passionate about travel, so I can make travel questionnaires. Aaron is passionate about music, so that he can complete music questionnaires. This means the same is true for others. Someone passionate about something could make questionnaires to help other interested people.

The framework for the app will be based on a few algorithms and types of questionnaires that can be combined or used by themselves to determine the best option. The types of questionnaires and their function would be predetermined and defined by unique classes. The types of the questionnaire we want to start with include: true/false; multiple-choice with a scoring system based on a scale of attributes and user input (1-5); multiple-choice where each possible result is scaled for each corresponding answer: every time user answers option_D, possible_result4 increments, tree of questions: the answers to earlier questions determines later questions.

Each questionnaire consists of types or types of questionnaires, but there are different types of questions we need to be able to ask. The kinds of questions we want to start with are text-based multiple-choice, picture-based multiple-choice, text-based true-false, picture-based true-false, and rating scale (1-5).

The functionality of the questionnaires and questions will be completed using javascript and possibly r to support javascript for analysis. We will need files to define each questionnaire type and question type that takes in files in a specific but straightforward format to create each questionnaire and question.

To make adding new questionnaires relatively simple, we need the framework that builds questionnaires to take the info the program needs to complete a questionnaire from some source that can be passed to it, such as a text file containing information about the questionnaire.

To display each questionnaire on the home page, we need to make an interface that displays the names and descriptions of the questionnaires within a clickable button that takes the user to the given questionnaire. To show each question within a questionnaire, the interface will need to display the prompt for each question alongside the options the user has for the question. To display the results, we will need a page that can display the questionnaire results for the user.

To make each of these interfaces will use HTML and CSS that calls our javascript and display and handle user input.

The project is incredibly scalable and relies on a simple framework to prototype for project 3. For project 4, we can expand on the question and questionnaire functionality and add our unique questionnaires.