EECS 448 Group 2 - Project 3

Design Paradigm

We will be using an object-oriented approach because we want a simple framework that allows a user to quickly create a document in a specific format that the framework will use to build the functionality of the questionnaire. Since there will be multiple questionnaires and we want them to be easy to add, we can't focus on designing functionality for questionnaires about specific topics. So makings a questionnaire an constructible class repeatedly makes it simple to add new questionnaires. An object-oriented design also makes it easy for us to define a question object which contains templates for each type of question and can handle its input, which means a questionnaire and its questions can be defined separately if, for example, there is a picture multiple choice question that needs access jpegs of the image options for the question.

A framework consisting of a few files for each object allows questionnaires to be passed in a folder with the instruction script to build the questionnaire and any supplementary files. Having a folder referenced for each questionnaire makes it simple to store questionnaire information on a server and pass it to the executive file that creates the questionnaire.

Object-oriented design also has the advantage of being easy to understand. Our functionality is based on the simple concept of a questionnaire that can logically be broken down into parts. Having a questionnaire and question object with all the instructions to build any type of questionnaire makes sense. This avoids the hassle and confusion of thinking about the questionnaires being processed as a stream of information rather than a type of questionnaire with attributes that have varieties of questions with attributes. Unlike object-oriented design, functional design would not allow the questionnaires and questions to be affected by user input as you can't define scope when using a series of functions.

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, rating scale (1-5).