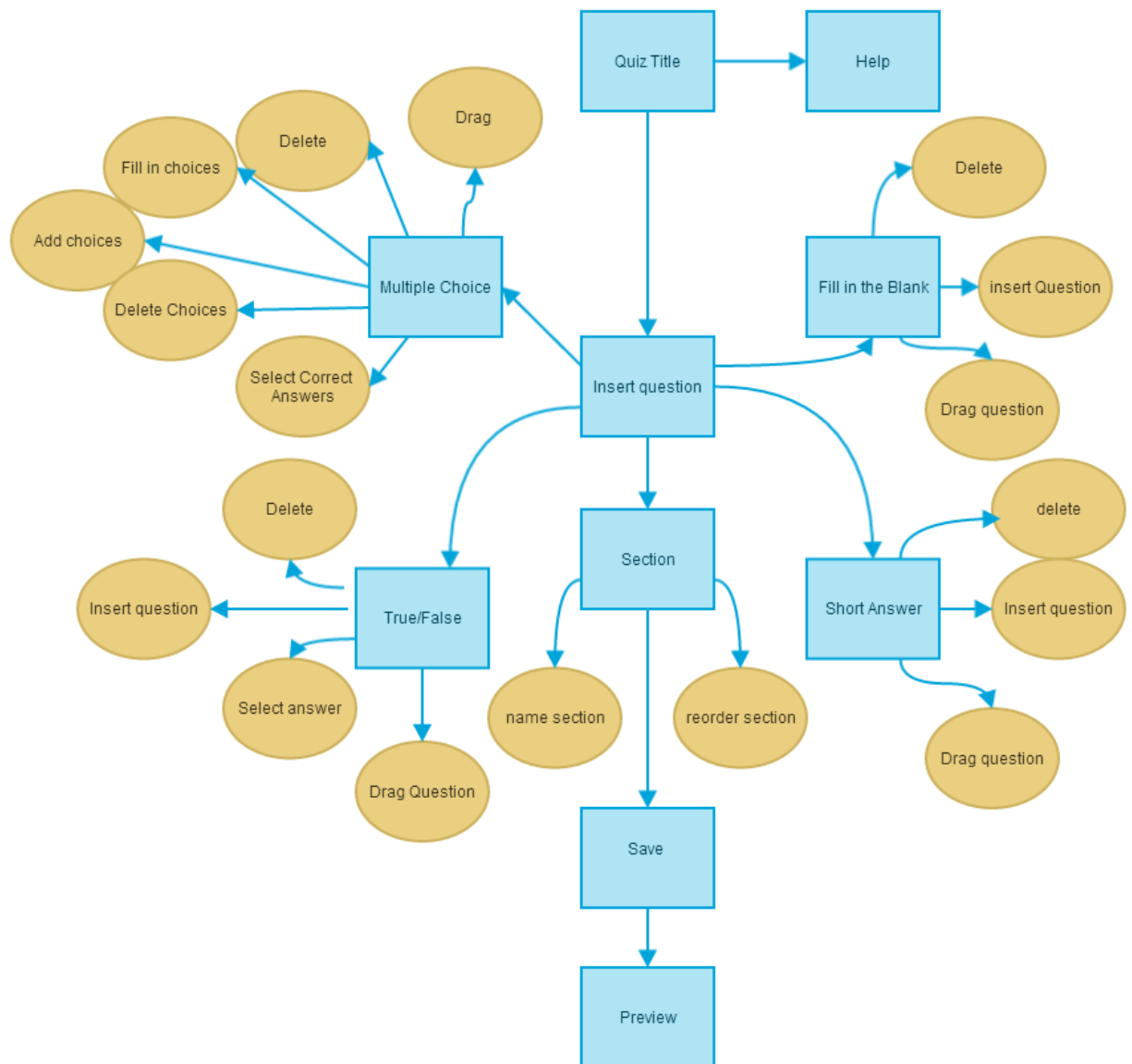


# Homework #4: Documentation

Team Pow: Christopher Speegle, Kyle So, Darren Syu, Tina Szutu, Jerry To

## Discussion:

Throughout our five-week long course, we were tasked with creating a **Quiz Maker** application. To create the page, we first created a wireframe using **HTML** to layout the features that were to be implemented. Next, we styled the application using **CSS** to improve its look and feel. After the layout and design was complete, we utilized **JavaScript** and **JQuery** to implement a variety of features. To store the questions, we used **JSON** to save the structure.



## Functionality:

For our HTML template, we chose to use the **Wire Frame 2 Electric Boogaloo** HTML template to implement the Quiz Maker functionality. The new features and functionality include:

- **Quiz Title:** User will be able to give their quiz a name(e.g. Philosophy Final Exam).
- **Preview:** Allows the user to preview their quiz before publishing it.
- **Multiple Choice:** Users will be able to add multiple choice questions to their exams.
  - a. **Add Another Option:** Users will be able to add additional answers to multiple choice questions.
  - b. **Delete Extra Answers:** Users will be able to delete all, but two answers from the multiple choice questions.
- **True / False:** Users will be able to add True/False questions to their exams.
- **Fill in the Blank:** Users will be able to add Fill in the Blank questions to their exam, which they will enclose the correct answers in braces.
- **Short Answer:** Users will be able to add Short Answer questions to their exam.
- **Delete Question:** Users will be able to delete any question by clicking the trash can icon in the upper right hand corner.
- **Point Value:** Users will be able to assign point values to different questions, which will update in real time to let the user know what the total points are for the exam.
- **Time:** Users can specify a time limit for the exam.
- **Help:** Short informational page that explain the functionality on the website.

## JQuery:

When designing the architecture for our Quiz Maker application, we wanted to allow the users to create quizzes quickly and efficiently. To accomplish this we used JQuery, which allows developers to create powerful features and to increase usability to the user.

- **Hot-key:** We implemented hot-keys that allow the user to create quizzes quickly.
  - a. **<CTRL> + <ALT> + <M>:** Inserts a multiple choice question at the bottom of the page.
  - b. **<CTRL> + <ALT> + <T>:** Insert a true/false question at the bottom of the page.
  - c. **<CTRL> + <ALT> + <F>:** Inserts a Fill in the Blank question at the bottom of the page.
  - d. **<CTRL> + <ALT> + <W>:** Inserts a Short Answer question at the bottom of the page.
  - e. **<CTRL> + <ALT> + <P>:** Allows a user to preview the quiz that they are working on.
  - f. **<CTRL> + <ALT> + <S>:** Allows a user to save the quiz that they are working on.
- **Sectioning:** We implemented a way for users to add sections to their quizzes, in order to make much more organized quizzes. The user will also be able to rename the sections to fit their needs.
- **Adding Questions:** When a user clicks to add a question, we used JQuery to generate an HTML5 template for the question that the user clicked.
- **Real-Time Point Updates:** When a user enters points into the points text fields located in the upper right hand corner, our application updates the points in real time to allow the user to see the total amount of points their quiz is worth.

## JavaScript

In order to achieve the goals of the Quiz Maker, we utilized JavaScript and JQuery, creating various methods to implement functionality.

- **addMultipleChoice()**
  - Adds a new multiple choice question at the end of the page. Increments the question counter after.
- **addTrueFalse()**
  - Adds a new true/false question at the end of the page. Increments the question counter after.
- **addFillIn()**
  - Adds a new fill in the blank question at the end of the page. Increments the question counter after.
- **addShortAnswer()**
  - Adds a new short answer question at the end of the page. Increments the question counter after.
- **addExtraAnswer()**
  - Adds an extra multiple choice answer to the question.
- **addSection()**
  - Creates a new section.
- **removeQuestion()**
  - Permanently deletes a question from the quiz.
- **deleteExtraAnswer()**
  - Removes a choice from a multiple choice question.
- **incrementMultipleChoiceCounter()**
  - Increases the multiple choice counter. Used to prevent conflicts.
- **incrementQuestionCounter()**
  - Increases the question counter. Used to prevent conflicts.
- **incrementAnswerCounter()**
  - Increases the answer counter. Used to prevent conflicts.
- **incrementSectionCounter()**
  - Increases the section counter. Used to prevent conflicts.
- **pointCounter()**
  - Sums up the total number of points and lists it under “Allocated Points”.
- **recall()**
  - Allows the drag and drop re-ordering of questions and answers.
- **allowNumbersOnly()**
  - Limits the point value to numeric characters.
- **storeQuestions()**
  - Creates a JSON object, populating it with data from the Quiz Maker.

## General Notes:

- **Sectioning Prototype:** Due to time constraints, we were unable to fully implement the previously stated sectioning functionality within our Quiz Maker webpage. In an attempt to provide ample resources for any developer after this course to expand upon the functionality, we have prepared a prototype (titled: sectioning\_prototype.html) to showcase how our sectioning functionality would have worked had we been able to fully implement the code into our Quiz Maker. Features of the sectioning prototype are as follows:
  - Sections are created using the “Add Section” button located in the upper-left corner.
  - Pressing on the “Add Section” button will bring up a dialog box, where the user is able to input the desired name for the section (or in the event that the user does not enter a name for the section, there is a counter incrementing in the background that will default the name of the section to be “Section [value of the counter].” After completing the naming of the section, the user will then click the “Add” button, which will close the dialog box and add the newly created section to the section bar, or click the “Cancel” button, which will simply close the dialog box and not add the section to the section bar.
  - Once sections have been introduced to the section bar, the user is then able to rearrange the sections into whatever order is desired.
  - The user may also delete any undesired sections by simply pressing the “x” button located in the upper-right corner of each section tab.
  - Upon clicking a section tab, that section block will open up to display any questions that are associated with that section. These questions may be rearranged within the section itself, or if desired, these questions can be dragged to the section bar and dropped into any given existing section and the question will disappear from the currently viewed section and reappear in the dropped section, thus showcasing the ability to move questions to various sections.
- **Sectioning Implementation:** The sectioning prototype offers the foundation for a future developer to properly implement the sectioning capability into the Quiz Maker. My advice for the path to accomplish this task, which is supported by multiple attempts at it myself, is that the developer include the dialog html code within the quizmaker html code as well as replace the javascript function “recall()” within the “add.js” file. Then, adjustments to the quizmaker.html code is needed in order to account for the introduction of adding sections and the ability to move any added questions within a pre-designated default added question zone, where those questions can be dropped into desired sections. With the proper rearrangement of the prototype’s code, the developer should be able to introduce sectioning and pagination in one feature, for the questions themselves are both logically placed into each section, which can be seen when viewing the source code of the webpage, and pagination comes into play in the sense that only the questions of the selected section are displayed on the screen and the other questions are hidden from view.
- **JSON Implementation:** The functionality of the JSON feature is partially completed due to time constraints. Currently, the JSON file is able to be created and is able to be saved to local storage; however, the saved quizzes are not accessible when clicked even when the JSON files contain all the necessary information associated with the saved quiz. The JSON file itself contains all of the following features: the title of the quiz, the time allotted for the quiz, the number of overall questions, number of sections, and the types of each question. Included in this updated version

of our quizmaker is the “Create New” button that fully resets the quiz making platform in order for the user to have a clean start for any given quiz.

- **Known Bug:** There is a bug in the clearing of the Section titles when clicking “Create New,” in where the Section titles are not cleared from the quiz making platform, while the questions themselves are cleared.