School of Computing and Information Sciences

Saint Louis University

1st Semester, A.Y. 2017 – 2018

FINAL COURSE

PROJECT DELIVERABLES

9346A Web Systems and Technologies

2:00 – 3:00 TF D515

De Vera, Melody Riza C.

Del Rosario, Miguel Paolo B.

Dolor, Rey Christian D.

Patricio, Jan Ronald M.

Pis-o, Henrhod Y.

Introduction

The website contains topics mainly about Java Web Servlets and Java Server Pages (JSP), PHP, Server-Side JavaScript (with Node.JS) and Web Application Security that contains the web application security risks as well as the references used to identify each topic available used as content for the website. It contains user interactive features such as a crossword puzzle encoded in javascript and dynamic quiz generated from the websites' database. The website was implemented to have user interactions that is made by applying web technologies to function completely.

About the website

The website covers topics from the finals of the subject Web System and Technologies that has been discussed in the course. The users can use the website as a short review on each of the topics under the different Server-side web scripting. In the website the user can also, interact within the website through the Quiz and activity module which contains the crossword puzzle that aims to review the user on the different topics by identifying the word being described for each question. The quiz is an objective type wherein it is presented as a multiple choice type of quiz that dynamically shows the choices in different form when the quiz is loaded.

Website Architecture

The website architecture shown in the diagram, shows the basic structure of the website and how the user can navigate through the different parts in the website. Using the search bar, the user can easily navigate to other topics without manually searching each module. In each topic, the user can select a subject as an accordion about the topic to read further. Users can also select two types of activity in the navigation bar under activity module containing two drop down options and continue to change topics using the navigation bar, or can read more about a topic in the official websites linked as references for the contents.

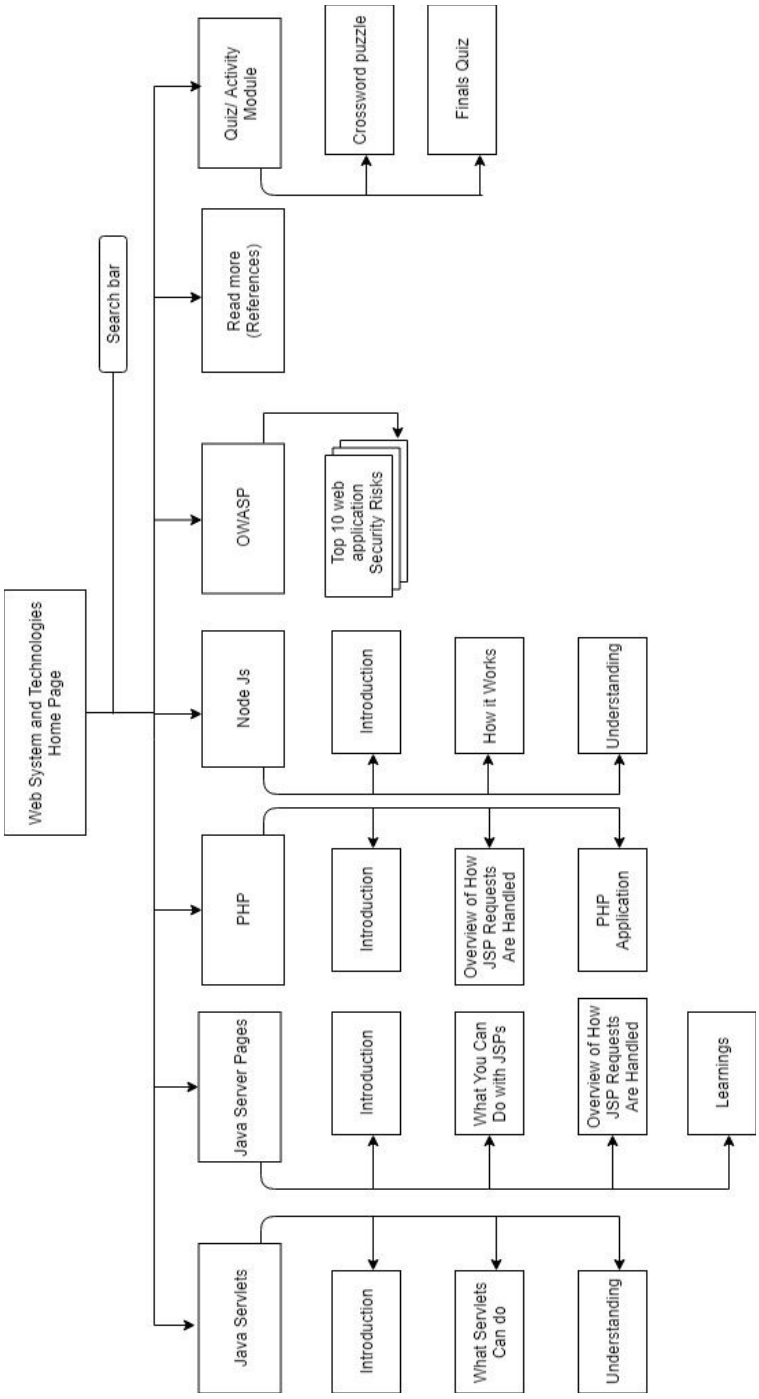

Figure 1: Website Architecture

- **Website Components**
  1. HTML (Hypertext Markup Language)
     a. The html file in the website is where the main content of the website is located and where other functionalities of the website can be accessed.

  2. CSS (Cascading Style sheet)
     a. The CSS file contains the main design of the web pages together with the added multimedia used for the website.
  3. JavaScript
     a. The JavaScript file contains the different functionalities for the website that was applies such as the accordion functionality, search functions, crosswords and other functions present in the website.
  4. PHP (Hypertext Preprocessors)
     a. The php was used for the quiz feature of the website as well as in accessing the database where the quiz information was listed to be able to generate a quiz with dynamic choices for each user.
  5. MySQL Database
     a. The database contains the quiz information the questions and the answers. It also contains the accounts for the website.

- **Website Features Implemented**

1. Mobile Responsiveness
    a. The website contains a responsive design which can be viewed using a mobile version and can be applied whenever the browser is adjusted in size. The responsiveness of the website made use of CSS and Media Query.

Figure 2a. Responsiveness

Figure 2b. Search bar

Figure 2c. Topic responsiveness

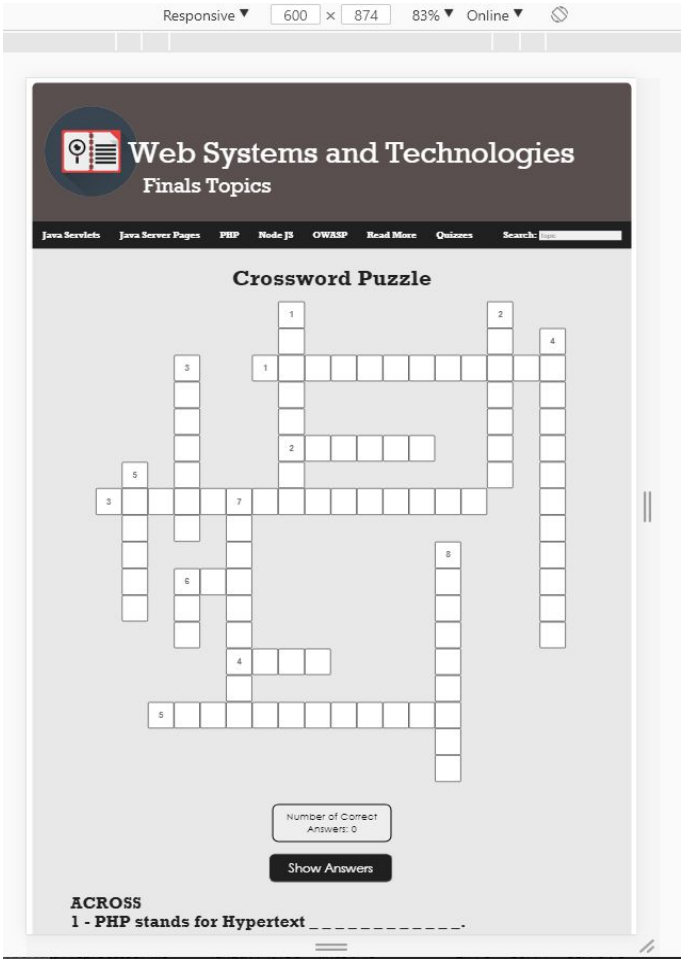

Figure 2d. Crossword

2. Search Topic feature
    a. The search topic feature was implemented by creating an XML file (links.xml) where the topic title and link was saved using <link> <title><url> tags to specify the title that could relate to the search and the corresponding url linked to the title. Using the javascript (script.js) where in a function is stored

there as showResult wherein it is used whenever the onkeyup is active. When the taken length of the function is empty it would be static, and when the function is given a value from the search bar, it will create a XMLHttpRequest that would take the value from the js and access the livesearch.php file. The php file will access the links.xml and compare the input from the user to the available links and returns the corresponding title (title tags) having only one or more than one result as the return value shown when the user searches a certain topic and would return ‘no related topics’ if no title was found that matches the links and title in the xml file .

Figure 3a: Search Topic Feature

When a topic is being searched, corresponding links will be shown after the search bar. The corresponding searches will be similar to the contents placed in the xml file as the key words for each topic. After the search, the results can be selected and you will be directed to the url of the topic you selected.

Figure 4b: No related topic

When the search topic did not contain any related information, set initially in the xml file, the result would return 'no related topics' as its value shown in

3. Crossword Activity
   a. The crossword activity has fourteen (14) items to be answered correctly. Item clues are provided below the puzzle (across and down sections). This was implemented using HTML, CSS, and Javascript codes. The content was generated using HTML input boxes, the interface was designed using CSS. The JavaScript contains two functions: (1) *enterLetter* and *(2) showAnswer.*

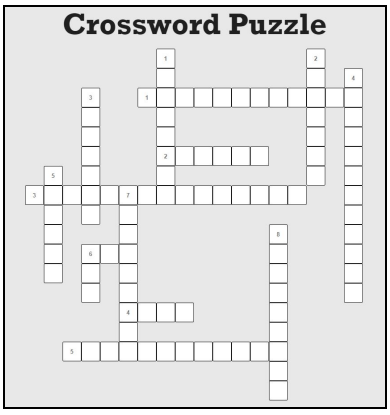

Figure 4a. Crossword Puzzle

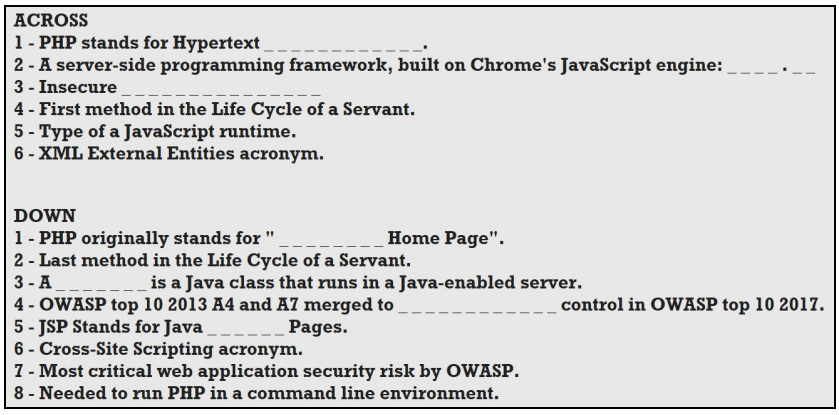

Figure4b. Crossword Puzzle Clues

b. The *enterLetter* function is called whenever the user clicks an input box. The *maxlength* of each input box is one therefore the user can only input one letter. Within the *enterLetter* function are conditional statements that check whether the answer (whole word) is correct. If the answer is correct, the border of the input box will become green, the text will become bold and the input box will be disabled and cannot be modified. The number of correct answers will increment when the user enters a correct word. The user finishes the activity when all the items were answered correctly. An alert window will appear and notifies the user that the puzzle is completed.

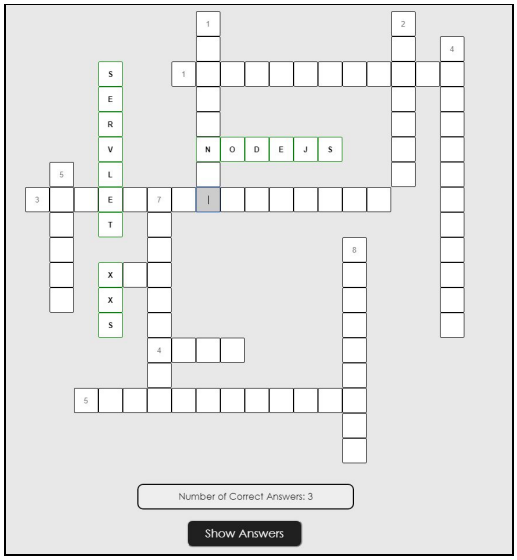

Figure 4c. Puzzle With Answers

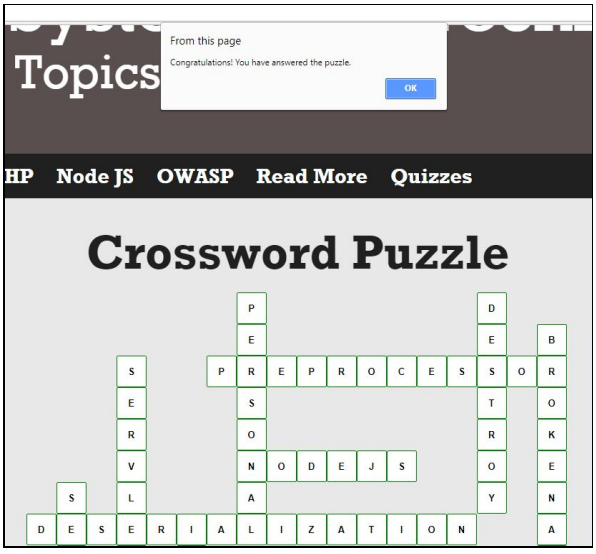

Figure 4d. Answered Crossword Puzzle

c. The *showAnswer* function is called when the *showAnswer* button is clicked. This function displays the correct answers for the crossword puzzle. The border of the input box will become red. The items that were answered correctly can be distinguished by having a bold font weight while unanswered items have regular ones.

are shuffled every time you refresh the page. A summary of the

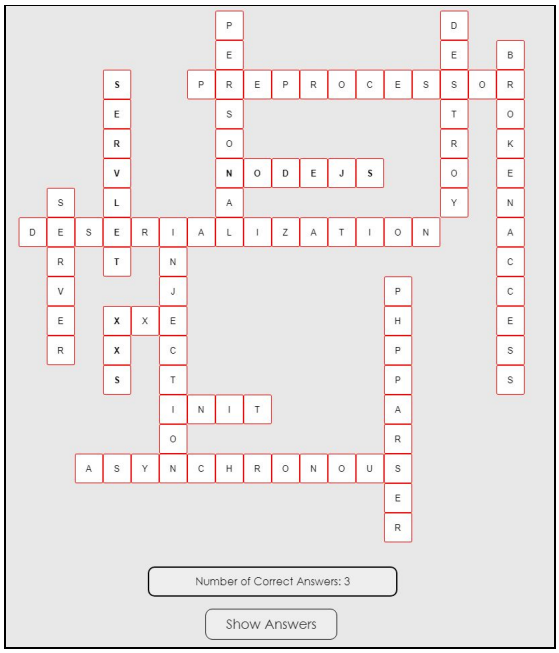

Figure 4e. Crossword Puzzle When Show Answers Button Is Clicked

4. Dynamic Quiz implementation

   a. The quiz items which is composed of the quiz number, question and 3 choices, one of which is the answer, are stored in a database (test.sql) and are being retrieved via the PHP code, by accessing the localhost where the database is stored. The choices quiz is then displayed in the results.php page. The summary includes the players answer, the correct answer and the score.

   b. The database is named as test.sql and contains the following information; quizID, question, choice1, choice2, answer as the columns listed, that will be used to generate the contents of the quiz.

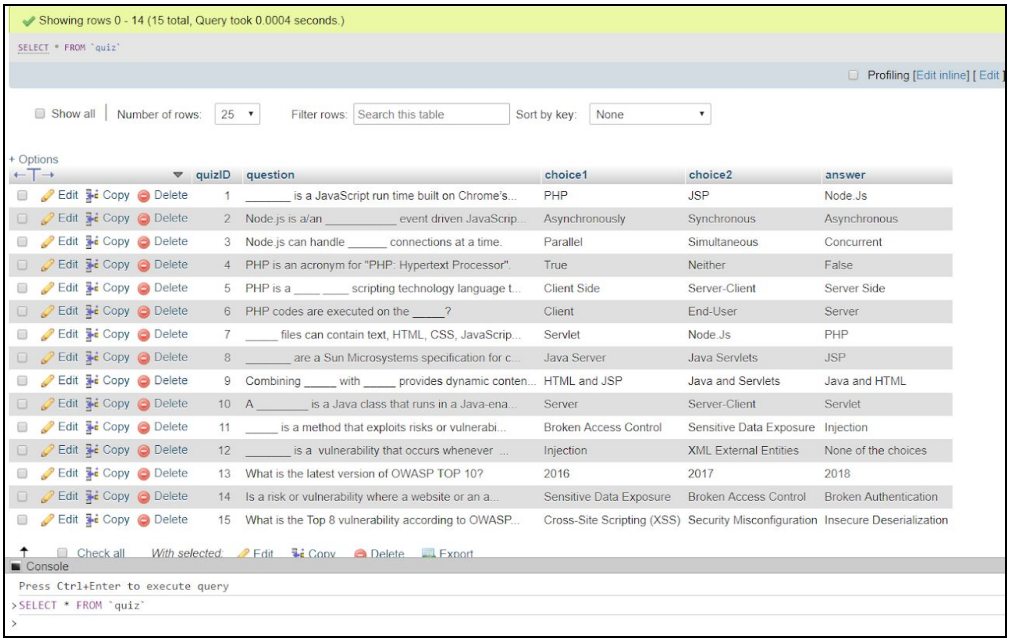

Figure 4a: test database

c. Users can select an answer by selecting the radio button with corresponding choices, and submit the quiz after they are finished answering each question. If the user attempts to submit the quiz without answering all of the questions asked, the user will be prompted to answer the unanswered question see Figure 4c.

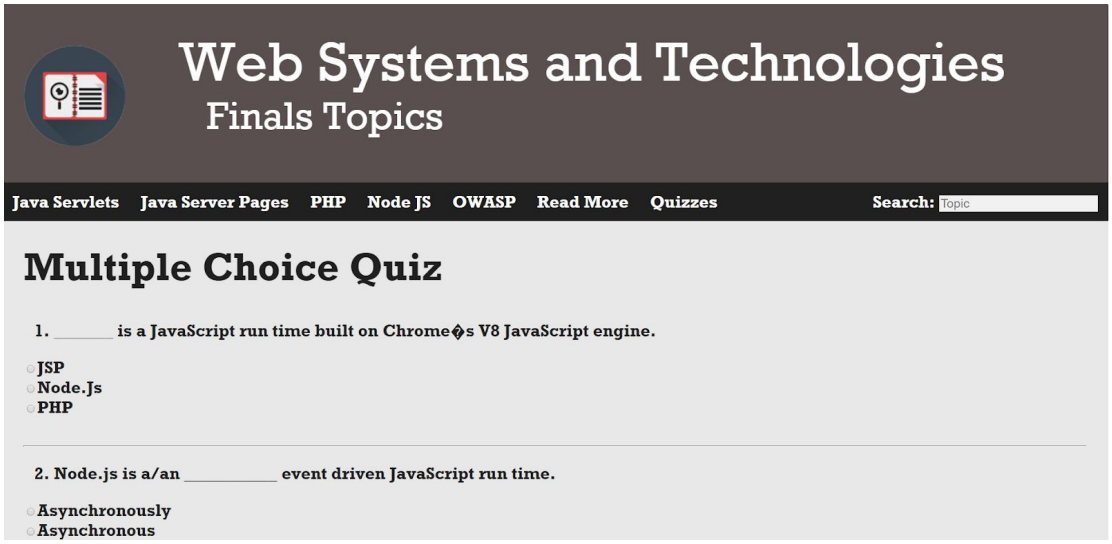

Figure 4b: Quiz module

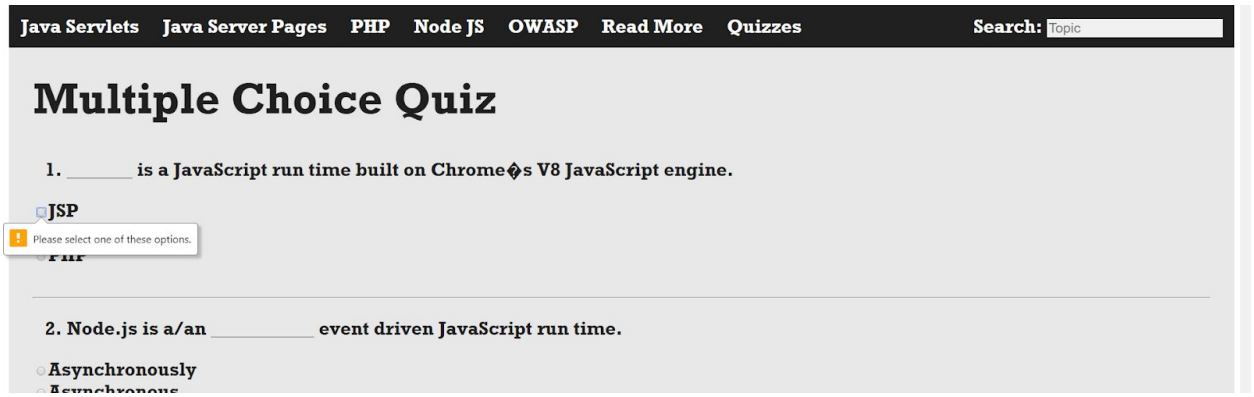

Figure 4c: Unanswered questions

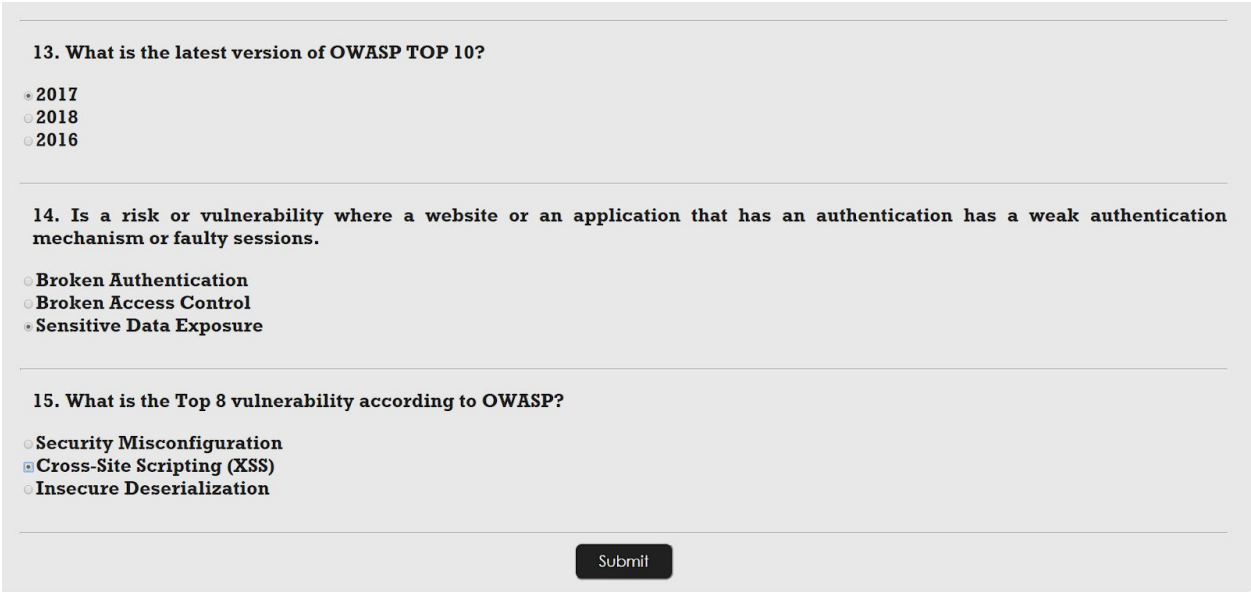

Figure 4d: Quiz Submission

d. After answering all the questions, the user can click the 'Submit' (Figure 4d) button located at the end of the quiz and automatically see their corresponding score and the correct answers shaded in green and the wrong answers shaded in red . The user will be given the option to "Play again " after answering the quiz (Figure 4e).

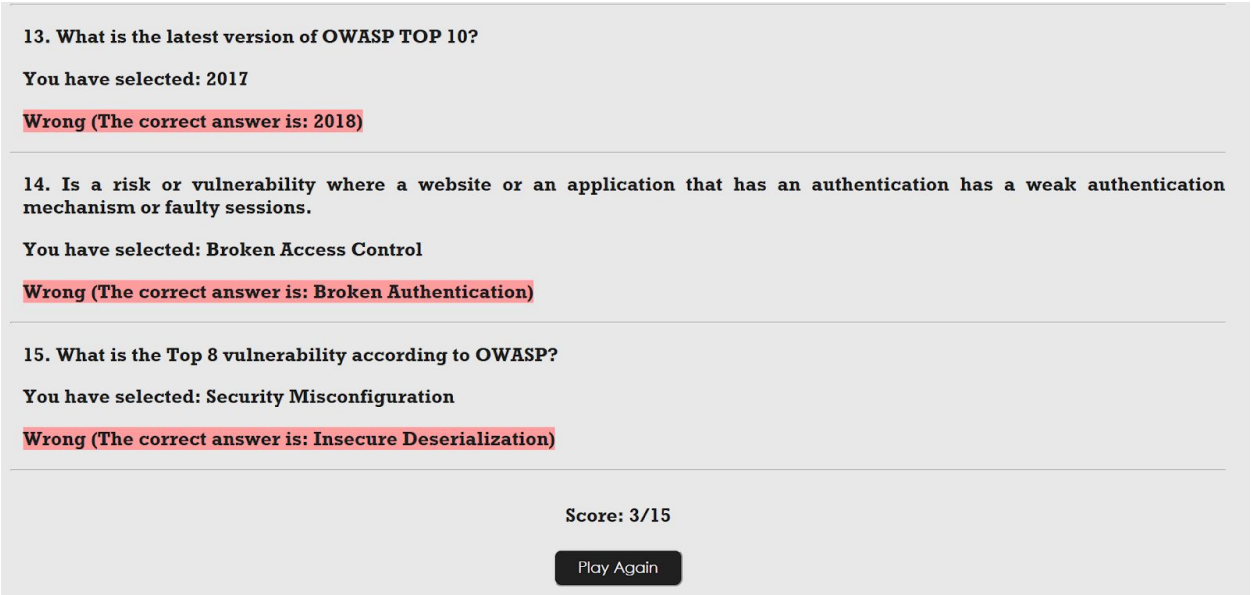

Figure 4e: Score