Ashley Stallcop
Susie Kim

The Creation Challenges of Quiz Quest

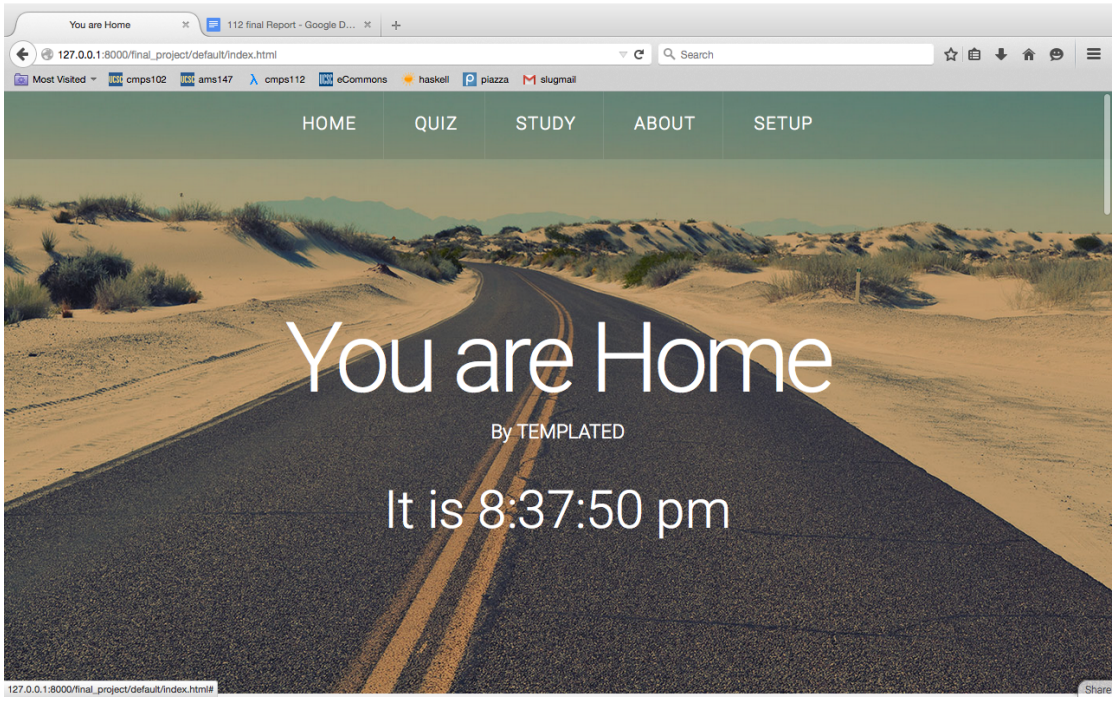
Introduction

Project Idea

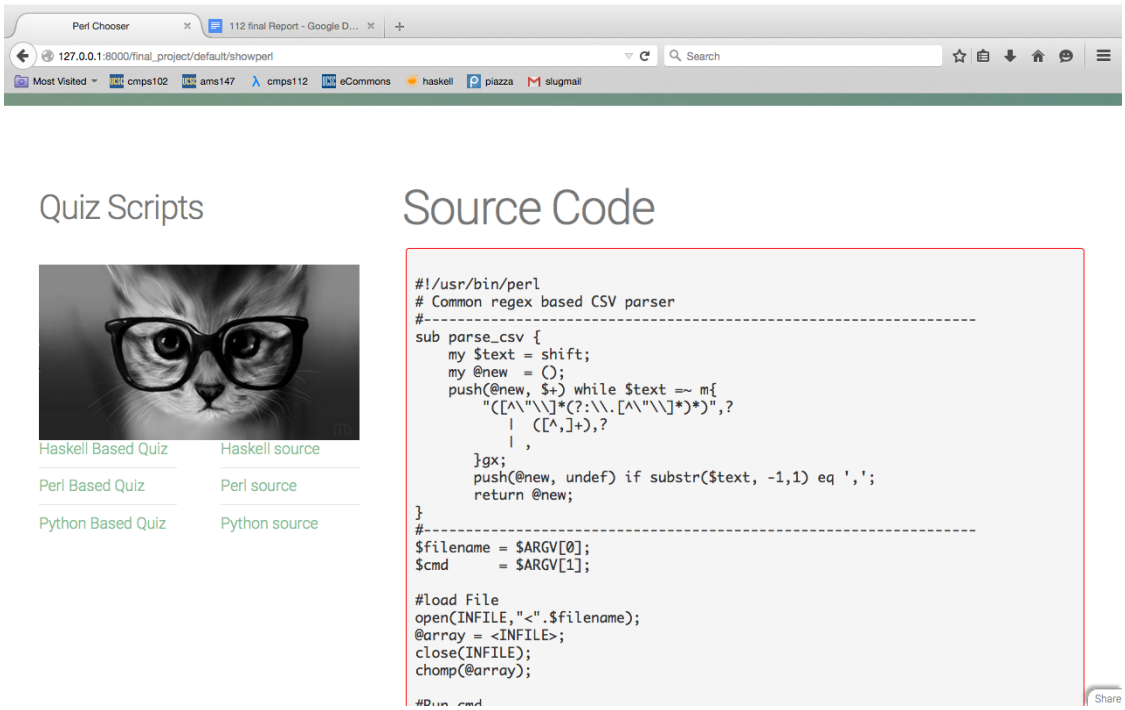
We initially intended to make a tutorial website on how to make the site we've made writing in Haskell using Snap web framework, as well as including a quiz at the end of the tutorial. However, due to our inexperience with Snap and not much code generation with Snap, we decided to make a website off of the quiz idea in our previous project idea. We created a quiz implemented and written in three different programming languages, Haskell, Python, and Perl, that test the user on the following coding languages: Haskell, C, C++, Java, Python, CSS, HTML, JavaScript, and JQuery. Our site also include a study material for users to refresh their minds on the material.

Screenshots

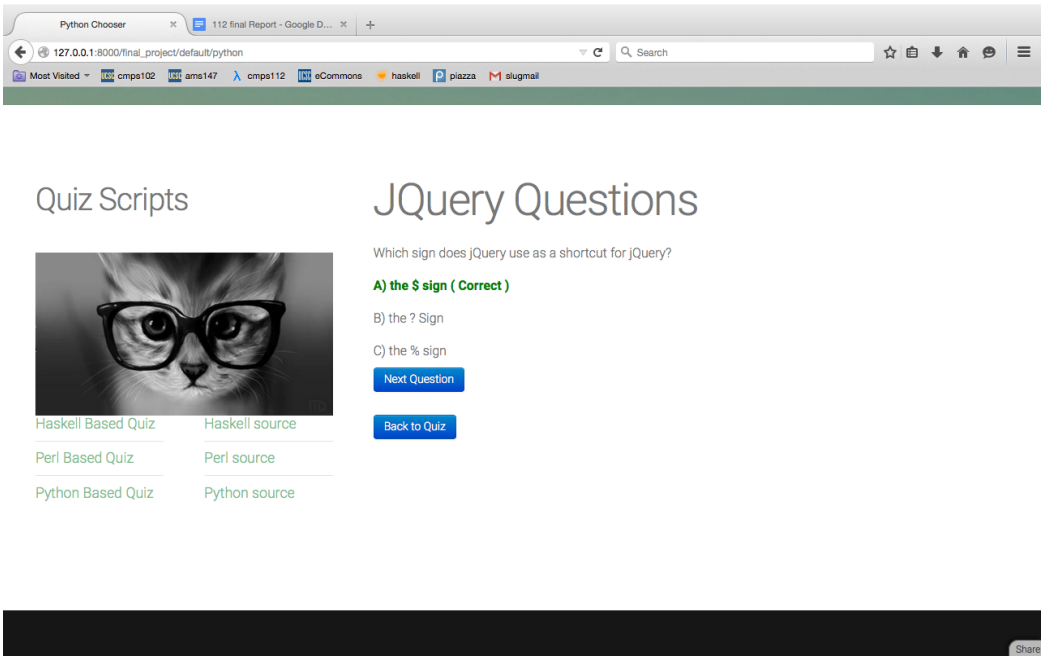
Below is the screenshot of our homepage that lets the user navigate through the website.



We have the source code for implementing the quiz displayed on our webpage. On the left hand side is a menu of which language the user would like to view.



This is an example of how our questions are displayed in the quiz.



Implementation

We used Javascript, HTML, Bootstrap and CSS to create an attractive and pleasant web site that is easy to understand and use. Our web page is educational based where users can learn simple syntax implementation in JQuery, Javascript, and many more. Using Web2py, we used CSV files in conjunction with Python, Perl and Haskell were used to implement the three different types of quizzes.

Design Decisions

Web2py was selected to provide a web server and application framework. It also provides a convenient and easy package system and fairly decent development environment. Its database and user account system are ignored for this project.

Programming Languages

We select Haskell, Python, and Perl to write the quizzes because these languages all differ in what they are and how to work. Comparing between these languages will tell us the which languages will prevail victoriously and which ones are weak and aren't as efficient.

Perl, Python and Haskell are used to create the three programs that perform the same identical function so that the languages could be compared. The three programs each have to perform the following tasks

1) Must open a GIVEN file and load its lines into a list or array.
2) Create a random number based on the size for the list or array.
3) Use the random number to select one of the elements/lines in the list or array
4) Parse the selected elements/lines based on the rules for a Coma Separated Values (CSV) formatted file.
5) Print each field on a separate line.

The CSV files contain the questions and answer in the following format:

Field 1 is the question to be asked
Field 2 is the correct answer to the question
Field 3 and 4 are incorrect answers to the question

Program Architecture

The website layout is as follows:

Home: the main welcome page
Setup: Pages that install/uninstall the Perl and Haskell scripts and describe other setup requirements
About: a page describing the project team
Study: Pages with information about various topics, is mostly mock data and intended to show where and how this kind of data would be store

Quiz: Pages that quiz the user on the various topics

The Quiz page allows the user to select a top to be quizzed on and then select one of three programming languages to be run to generate the question for the selected topic. The quiz process works as follows:

1) Web2py uses the python subprocess to make a system call run the select programming language. It passes the program filename and the CSV filename contain the topic question.
2) The program returns one of the question as 4 separate lines.
3) Web2py generates a random number from 1 to 3 to deterring where to put the correct answer in the list of choices. If 1 then the answer will be the first choice. If 2 the answer will be the second choice and if 3 the answer will be the last choice.
4) Javascript is used to tell the user if the choice they have clicked on is the correct choice be marking it green for correct and red of incorrect.
5) The user clicks next question to repeat this process and gets a new question.

Evaluation

Risks and Problems

Starting out with a framework we weren't familiar with was a big risk because it would take great amounts of time to learn how to use it, which is why we fell back onto Web2py since we were familiar with this framework from previous experience in a previous class we took.

Pros and Cons for each language

Haskell is a functional language, whereas Python is a high level programming language, and lastly, Perl is a scripting language. The greatest difference between the programs is the way they parsed the CSV files. Haskell a functional based language, which required the most lines of code, and using its functional parsing was how we achieved the parsing of the CSV format. Perl required fewer lines of code and was able to achieve the parsing by use of complex regex, which is a native part of perl. And lastly, Python, being a high level general purpose language, was able to parse the CSV format in a single line of code using a common module. Python's design is to provide many complex methods so that the programs can be written with very few lines. Of the three, Python was the fastest because of its precompiled module for parsing the CSV format, Perl took the second fastest because it had more lines that had to be interpreted first before running, and lastly, Haskell took the longest because it had more lines to interpret and run. This is not meant to say that any of these languages are faster than another. The code was written to demonstrate the differences of each language rather than to produce the optimal method.

Conclusion

Overall, we we learned a lot about each and every language we worked with, mainly Haskell, Perl, and Python. Python was implemented with just 9 lines whereas Haskell and Perl took five or six times that. It is clear to say that Haskell and Perl were extremely difficult to implement. As creating questions for the quiz, we also learned few facts about each language we created questions for such as HTML and JQuery. Although we switched gears in the middle of our previous project, we managed to complete our final goal in time.