UNIVERSITY OF WATERLOO
Faculty of Mathematics

Web Development Framework - Front and Back Ends

Caribou Contests
St. Catharines, Ontario, Canada

Lechuan Peng
2B Computer Science
March 2019

MEMORANDUM

To: Thomas Wolf

From: Lechuan Peng

Date: March 19$^{\text{th}}$, 2019

Re: Work Report: Web Development Framework - Front and Back Ends

---

I have prepared the enclosed report entitled, "Web Development Framework - Front and Back Ends", for my 2B work report and for Caribou Contests. This report, the first of four work reports that the Co-operative Education Program requires that I successfully complete as part of my BMath Co-op degree requirements, has not received academic credit.

Caribou Contests team that you lead provides an online platform for students to participate in the contest from all over the world. My job as Computer Programmer and Interactive Media Developer required that I fix the buggy interface of web-pages and develop some tools for any users to be more convenient. I responded to the queries from students, teachers and parents. This report is an in-depth study of the development of Caribou Contests website.

The Faculty of Mathematics requests that you evaluate this report for command of topic and technical content/analysis. Following your assessment, the report, together with your evaluation, will be submitted to the Math Undergrad Office for evaluation on campus by qualified work report markers. The combined marks determine whether the report will receive credit and whether it will be considered for an award.

Thank you for your assistance in preparing this report.

Lechuan Peng

TABLE OF CONTENTS

List of Figures

List of Tables

Executive Summary

This report provides basic ideas and frameworks of web development on Caribou Contests web pages. Method of analysis include vertical and horizontal analyses as well as contrast analyses on disadvantages. Results of analysis show that proper models should be chosen to provide appropriate interfaces. In particular, to improve programming experience, alternatives of current languages (PHP, JavaScript) should be taken into consideration. Moreover, all future programmers are advised to make the code more readable.

1.0   Introduction

Nowadays, in order to save the time from repetitive works, web frameworks have more applications in website. Websites have more complex tasks that store users' information, process users' input and connect the world. Originally when Web page was created, its purpose was to display stationary contents (Schwarz, 2013, p.3) [1]. Due to growing demand for internet, static contents cannot satisfy users' needs. Therefore, more powerful, robust systems are created to perform difficult tasks and meet people's needs. Web design, as well as other popular software designs, has adopted a basic architecture: front and back ends. This framework enables the page to react dynamically and responsively. Schwarz (2013) argues that "With simple scripts that generated content, programmers became able to write applications. These applications allowed interactions between clients and servers without requiring the client to install additional software" (p.3) [1]. Therefore, non-static web design saves us from downloading which might have extra cost and makes our lives much more convenient.

Caribou Contests is an online mathematics contest which is purely based on web pages. The company adopts this basic idea: front and back ends in its design. Front end is to provide an interface for admins, teachers, parents and students. Back end is for programmers to process the data from interface and react to the database. In this way, web page can respond to the changes in the interface dynamically.

This report discusses the existing front-and-back-end model and gives a brief over view of the implementation. It then analyzes other frameworks related to front-and-back-end, discusses the

drawbacks of current languages and their alternatives, and examines the target audience of web

pages.

2.0   Analysis

There are many models using on front and back ends. One of the most significant models is MVC, which stands for model view controller. 'View' is the interface, which user sees it. Then he uses 'controller'. The backbone of MVC, 'model', manipulates the controller then updates the view accordingly. (see Figure 2.1).

2.1   MVC on the Front and Back Ends

"Model–view–controller is an architectural pattern commonly used for developing user interfaces that divides an application into three interconnected parts" ("Model–view–controller," n.d, para.1) [2]. This pattern is also adopted in front and back ends of web design.

2.1.1   MVC on the front end

Front end is the outermost layer of whole web page, which is also regarded as the user interface. However, this leads to a question: front end is now the outermost layer, how can we perform the model and controller in it? While, this can be implemented in JavaScript. Since front end directly communicates with users, the program must interact with users as a 'real-time programming'. Without an MVC framework, codes are hard to maintain and improve.
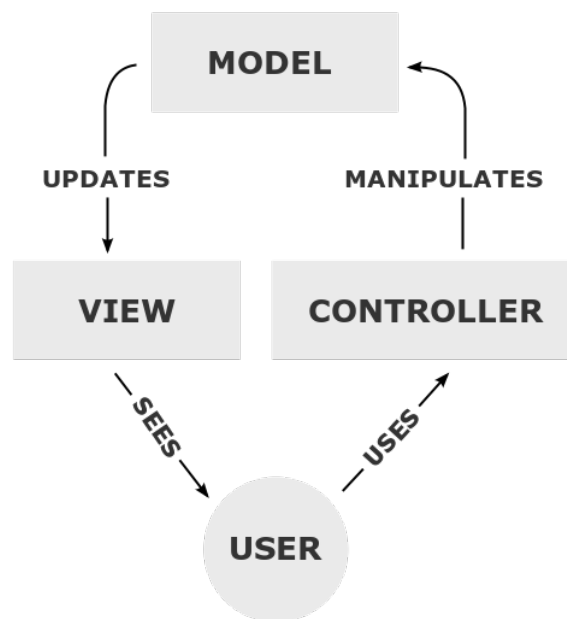
Figure 2.1 Diagram of interactions within the MVC pattern.

Figure 2.2 Reidemeister Moves
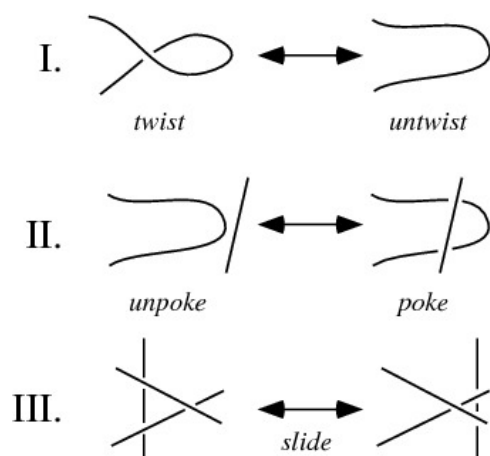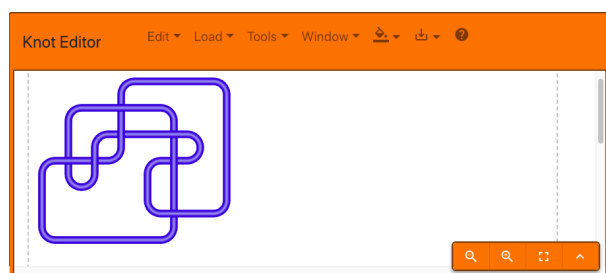
Figure 2.3 Knot Editor

Caribou Contests developed a tool, 'Knot Editor' (see Figure 2.3), which allows users to draw any knots and perform any knots deformations as they want. In this editor, users can perform 3 type of 'Reidemeister Moves' which do not change the invariant of this knot (see Figure 2.2). After several 'moves', users can untangle this knot to the form which cannot proceed any further.

Knot Editor adopts the idea of MVC. It is implemented purely by JavaScript (front end). It uses a 'model' to store the information of lines and corners, a 'controller' to enable the user to edit the knot, and a 'view' to display the whole knot. One of the advantages is that new features can be easily added without changing other part, i.e. view and controller, which minimizes the cost of programming.

2.1.2   MVC on the back end

On the back end, there are a number of choices to realize MVC: Java, C#, Ruby, JavaScript (Node.js), PHP. For Caribou Contests, it uses the most popular language, PHP. Compared with front-end MVC, there is something different. A generic model which for MVC involves 5 parts: browser, controller, view, model, database. Four steps are involved (see Table 2.1 and Figure 2.4) (Meng, 2015) [3].

For Caribou Contests, it uses MVC to create one 'template model' for all programs using similar approaches. For example, 'Email' model is generic for sending emails. It processes the necessary information to fill the missing parts of email body, then it records in the database and changes the web content accordingly. In the future, MVC allows programmers to add more
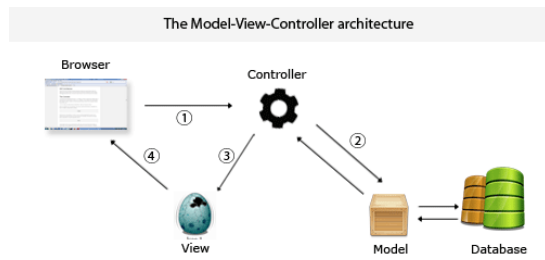
5

Figure 2.4 Four steps in MVC

(Galley Meng 2015)

Step 1	Controller gets the request from users
Step 2	Controller calls the model to record
Step 3	Controller pass the data to 'view'
Step 4	'View' renders the result to the web page

Table 2.1 Four steps in MVC

features easily without changing other sessions.

2.2 Drawbacks of the Current Languages

2.2.1 PHP

Welling and Thomson (2017) show that "it was adopted by other talented people and has gone through several major rewrites to bring us the broad, mature product we see today" (p.3) [4]. The main reason for the popularity is easiness to master as a beginner. However, this feature leads to many problems for experienced programmers.

Since PHP is easy to get started with, there is a large percentage of new PHP programmers making mistakes. Unfortunately for them, PHP does not have a complete model to check errors. Veekun (2012) found that PHP is undebuggable since there does not exist any stack traces to track where the real bug is [5].

Moreover, there is another historical issue due to its lack of maintenance. PHP has inconsistency since the built-in functions are named improperly (Veekun, 2012) [5]. PHP borrowed

many features from other languages, therefore PHP does not have an uniform standard for some functions. When less experienced programmer misuses one with another, fatal errors might occur.

7.5 million sites are using PHP as a main language in Web design today (SimilarTech, 2019) [6]. Although PHP is widely used all over the world, it has alternatives which have better architecture. The most popular one is Node.js which processes the data in JavaScript on server side. Capan (2017) argued the following:

> In one sentence: Node.js shines in real-time web applications employing push technology over websockets. What is so revolutionary about that? Well, after over 20 years of stateless-web based on the stateless request-response paradigm, we finally have web applications with real-time, two-way connections, where both the client and server can initiate communication, allowing them to exchange data freely. [7]

Node.js performs the exact same task as PHP, but it adopts better designed language, JavaScript, which solves the issues mentioned above. Therefore, it is not a bad choice to use Node.js instead of PHP.

2.2.2 JavaScript

Unlike the back-end JavaScript, front-end JavaScript has some disadvantages.

First, it lacks for security. Everyone can easily right click and select 'inspect' to see what are
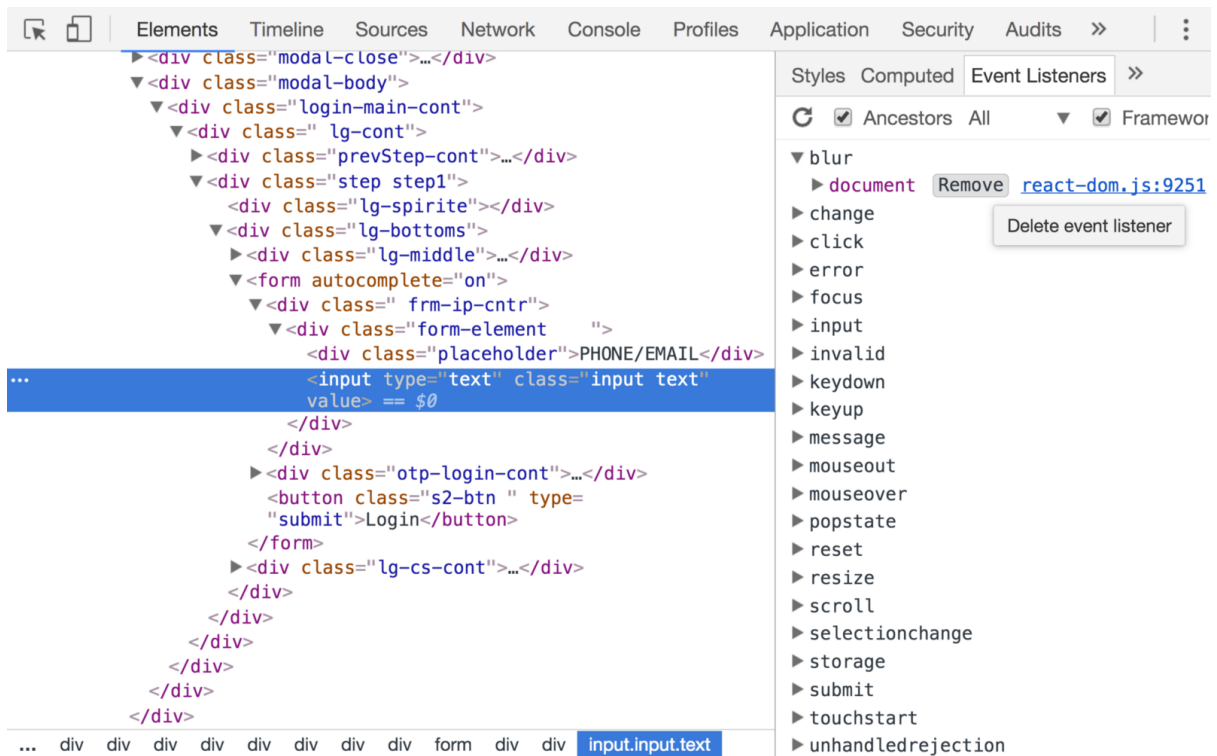
Figure 2.5 Malicious tampering JavaScript via console

(Aziz Khambati from Hackernoon 2017)

exactly implemented. Then if he wants to maliciously tampering any parts of the codes, he can

easily change the code via console (see Fig 2.5).

Second, some APIs are not supported by some browsers. For example, an API,

'`Intl.DateTimeFormat().resolvedOptions().timeZone`', is used to determine the sys-

tem time zone of the user in Caribou Contests Website. However, `Intl` might not be supported

in some browsers, then Caribou programmers have to come up with other options to make up.

Therefore, there are several languages improving readability of JavaScript and fixing some

historical issues. Currently, TypeScript and Dart are much better than JavaScript, though there

are some limitations for them.

2.3   Target Audience

In terms of Caribou Contests, there are two types of target audience. First type is users in general, i.e. parents, students and teachers. Second type is admins who are superusers.

2.3.1   Users

Caribou Contests is a world wide online contest. Contest participation is not free for some grades, so it is important to find potential customers in order to sell services.

Website needs to advertise and appeal to more users. In order to achieve this, Caribou Contests creates free 5 minutes to some interactive games for students to play. Moreover, one of six contests participation is free in a year. Therefore, many students can benefit from free services, then decide whether to pay for the following. Laszlo (2016) argued the following:

What's good about TypeScript?

- Every piece of JavaScript code is valid TypeScript and therefore it integrates much better than CofeeScript or Dart languages.

- Aligns with future JavaScript standards. [8]

Website tools also needs to be comprehensive and robust. In terms of teachers, they need to manage many students under their contact accounts to handle any requests they might have.
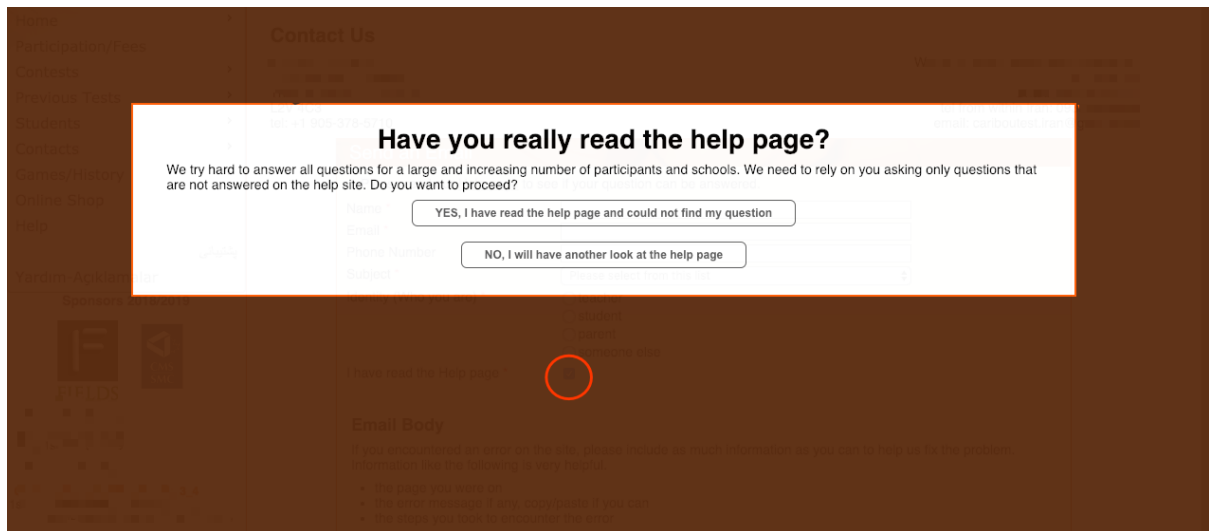
Figure 2.6 Extra confirmation

(Caribou Contests 2019)

Therefore programmer should preview what requests will be made then develop the tools accordingly. Also, people tend to tamper or misuse the interface of program, so tools have to be robust against any bad attempts. In Caribou Contests, there are many cases triggering alert which notifies the user and prevent any further attempts. Also, this idea is useful when user tries to fake some data. For example, if one has not read the FAQ pages then he wants to contact Caribou Contests via emails, further attempt should not be allowed unless he has really read the FAQ pages. To solve this, Caribou Contests adopts a framework which blocks everywhere else on the page unless he acknowledges terms or rules (see Fig 2.6). After he checks the checkbox in red circle, he has to acknowledge that he has really read the help page. This extra confirmation can reduce the pressure of customer services.

2.3.2 Admins

Besides users, admins also need to have a good interface to manage all the data. It is time wasting to walk through all data in database when superusers want to find or update the information.

Moreover, proper interfaces are needed when admins want to bulk edit the users and send mass

emails. Ideally, some actions should be done automatically to reduce the work of admins.

3.0   Conclusions

In web development, MVC is important and useful both in front and back ends.

There are several disadvantages of the current languages: PHP, JavaScript, but now they both have alternatives which improve the programming experience.

It is crucial to find the target audience, users and admins, of web pages. Programmers should provide them with comprehensive, robust and handy tools to meet their requirements.

4.0   Recommendations

Because the tools for admins or users are added constantly, or the previous tools are maintained continuously, company should enforce the all programmers to make the code readable, i.e. proper indentation, sufficient comments, reasonable variable and function names, etc. This will save much time for future programmers.

References

[1] Schwarz, M. (2013). *Design and analysis of Web application frameworks* (Master's thesis). Retrieved from https://pure.au.dk/ws/files/54599671/PH.D_dissertation_Mathias_Schwarz.pdf.

[2] Wikipedia contributors. (2019, January). *Model–view–controller*. In Wikipedia, The Free Encyclopedia. Retrieved from

https://en.wikipedia.org/w/index.php?title=Model%E2%80%93view%E2%80%93controller&oldid=879482602

[3] Meng, G. (2015, September). *PHP MVC framework example tutorial*. Retrieved from

https://www.awaimai.com/128.html

[4] Welling, L., & Thomson, L. (2017). *PHP and MySQL web development*. Hoboken, NJ: Addison-Wesley.

[5] Veekun, E. (2012, April) *PHP: a fractal of bad design*. Retrieved from

https://eev.ee/blog/2012/04/09/php-a-fractal-of-bad-design/

[6] SimilarTech. (2019 January). *Market share & Web usage statistics - PHP*. Retrieved from

https://www.similartech.com/technologies/php

[7] Capan, T. (2017). *Why the hell would you use Node.js*. Retrieved from

https://medium.com/the-node-js-collection/why-the-hell-would-you-use-node-js-4b053b94ab8e

[8] Laszlo (2017). *The best JavaScript alternatives - how to choose?*. Retrieved from

http://blog.debugme.eu/best-alternatives-to-javascript/

Acknowledgements

Thanks to the members of all Caribou Contests for insightful discussions and inspiring friendships.

I truly appreciate to Thomas Wolf for his patient mentorship. He has been the best supervisor one can imagine.

I am thankful to Kindeep Singh Kargil for his constructive help on front end programming.

I would like to thank Tyler Crane for introduction on all web frameworks in Caribou Contests.

I would also show my gratitude to Sarah L. Hasselman for providing suggestion on English and French texts.

Last but not least, I would like to thank all my friends, especially my three lovely roommates, for their encouragement and support.