Abstract :

Functional Reactive Programming Using Elm

Leondra Morse*
Department of Computer and Information Science
University of Pennsylvania, Philadelphia, PA 19104
DREU Computing Research Association Program
Project Faculty Advisor: Dr. Stephanie Weirich
Summer 2015

Functional Programming is a programming paradigm. A style of building elements and structure of computer programs. This will treat computation as the evaluation of mathematical functions and will avoid changing-state and mutable data. Functional Reactive Programming is a programming paradigm for asynchronous data-flow using the building blocks of functional programming such as map, filter, reduce, etc. In the field of computer science, we are especially interested in forming uncomplicated and simple approaches in order to simplify a series of actions or steps taken to achieve a finish product. These procedures provide a collection of processes in the development and construction of algorithms. This will enable clarity, understanding, simplicity, and elegance to the knowledge in comprehending codes of various languages from different authors and editors. Functional style is different to current promoted languages such as Java, C++, C, and Visual Basic. Software engineers are always looking for the next best step to a concise, abstracted manner of coding. The goal in this research project is to learn, understand, and comprehend functional reactive programming by using Elm language to design and develop a program.

How many weeks have you worked so far, and how many weeks do you have left?

Scheduled Work Dates: 06/01/2015 - 08/07/2015

Week Worked: 06/01/2015 - 07/05/2015 (5 weeks)

Week Left: 07/06/2015 - 08/10/2015 (5 weeks)

What is the status of your project? (Include what you have accomplished so far and what you expect to accomplish in the rest of the summer)

I accomplished so far for the first month, was understanding the functional reactive language style. I learned the features of Elm such as the programming syntax. These syntax include creating functions, calling functions, pattern matching, automatic currying, creating list, elm-html, elm conversion into JavaScript.

In order to test run Elm, my mentor helped me set up the Elm platform that's needed to be performed in order to run Elm on your computer system. As well as installing the necessary packages to perform the various HTML, and JavaScript conversions.

What is expected to be accomplished by the rest of the summer is for me to be able to not only read and interpret Elm, but also create my own projects. This will enable me to translate programs that I have written in JavaScript into Elm. I would like to have a clear understanding of all aspects of the Elm language so that I will be able to use Elm in my senior design project for my computer science bachelor's degree. This will not only help me become a more diverse programmer, but it will also open up professors', and my fellow students' minds to other programming languages that are out there in the world.

What has been your biggest challenge so far?

Elm is a Functional Reactive Programming Language. Learning this programming style has been the biggest challenge for me. The hardest part of learning this programming style is letting go old habits, such as ending a line with (;), or changing the values of variables (x= 45) , and writing every step out step-by-step. Also, the abstract content of the coding language has made it very difficult to get used too.

Example:

Creating Function

Functions: mappings that take one or more arguments and produces a single result. Defined by using equations that's gives a name for the function, a name for each of its arguments, and a body that specifies how the result can be calculated in terms of the arguments. Shown in figure 1.

Figure 1:

```
[function name] [arg (s)] = [body (action (s) )]

          |          |           |

          V          V           V

       add       x  y   =     x + y
```

Note: There is not ending (;) line of code, no mandatory parentheses for functions calls. Everything in Elm is an expression providing more of an concise code.

Call Function

When the function is applied to actual arguments, the result is obtained by substituting these arguments into the body of the function in place of the arguments names when called. Must provide the exact number of arguments placed in the function. As provided below in figure 2, the function takes two arguments (1 , 1) such as it is expressed in the function (Figure 1).

Figure 2:

```
show (  [fucnction name] [ (arg (s) ) ]  )
  |             |              |
  V             V              V
show (      add               1  1        )
```

Note: The show function allows the function to be printed to the screen, or to be displayed with one word command known as show.

What makes it even harder is the scarce amount of material on the subject. This is a relatively new programming language, so there are no books on the subject. As I tried to review the tutorials located on the website, I became highly confused with the information provided. The information provided skimmed the surface of the programming language. Following the program packages (Library Documentation), did not help me understand the functions provided in the examples. Instead it only gave me the type of the function, and not an example of how to use it.

What is the most exciting thing that has happened so far?

The most exciting thing that has happened so far is meeting new people and seeing that I am not the only one that is doing an internship this summer. It was exciting going out with people I had met over the past weeks through the internship, and going to see the historical buildings in the old city of Philadelphia. Also, it is exciting to get material that you have been struggling on for so long, and to have it click in your brain like you already know the material.

What is your working environment? For example, how frequently do you meet your mentor, the grad students in your group, others in the department, etc? Are you working on your project alone, with another undergrad, or with a graduate student. Have you attended any research talks?

My work environment is in a computer lab located in the basement of Levine, the same building that my mentor's office is located. I am provided with my personal area, computer, draw locker, and a key to access the computer lab area. There are other undergraduate students also working in the space.

Working on this research project alone, I met with my mentor frequently during the first month of working. At least 4-5 times during the week for about 2-3 hours a day because in the beginning of learning the new language Elm I had a hard time understanding the logic of functional reactive programming. I needed more assistance on learning the material, so I tried to meet my mentor as much as possible to review and analyze examples in Elm.

As I began to become a little more comfortable with the functional programming in Elm, I try to meet my mentor at least 2-3 times throughout the week in order to discuss what I have been reviewing on my own and the understanding that I have acquired.

I attended a few research project presentations recommended by my mentor Stephanie Weirich because she is a part of the Programming Language Club. These sessions consisted of students presenting their research, and getting feedback from other students and faculty members. Also, I attended a proposal defense given by Richard Eisenberg, Stephanie Weirich's doctorate student, on the subject of Dependent Types in Haskell: Theory and Practice.

What is your housing and roommate situation?

I live on the campus of the University of Pennsylvania in Philadelphia. I stay in the Rodin Housing building in apartment with three bedrooms. I have a private bedroom with 2 other suitemates, one bathroom, and one kitchen.

Is there anything else that you would like us to know about your experience?

This is my first summer internship, and I was very nervous coming alone to a new place that I am not familiar with. Being that I can be quite shy, this internship helped make me become more sociable. I have been impacted positively by attending this internship. I am thankful for being selected out of the few that were selected to learn something new.

Is there anything you would like to tell your mentor (these comments, if any, will be forwarded to your mentor)?

I would like to thank my mentor for having the patience to analyze and break down the Elm examples for me, and being able to explain every line of code clearly for my understanding. I know it can get quite frustrating working with a new student who does not get the programming style the second or third time around, but she has been very patient with me, and has given me the confidence to learn Elm. Even when I was having a hard time understanding the information, she always had time to meet me and explain what is going on in the codes. I can truly say, that my mentor, Stephanie Weirich, is the very reason I enjoy trying to learn functional reactive programming in Elm. She has made it fun and interesting.