Research Assessment #12

Date: 21 January, 2020

Subject: Designign a Language

APA citation:

Turbak , F., Gifford , D., & Sheldon, M. S. A. (n.d.). Design Concepts in Programming Languages. Retrieved February 21, 2020, from https://www.e-reading.club/bookreader.php/134602/Gifford,_Turbak_-_Design_Concepts_in_Programming_Languages.pdf

Assessment:

This week, I further researched the development of programming languages. Specifically, the designing of a language. Last week, in my first mentor visit with Mr Copps and Mr Rohde, I was advised to find open source projects about Natural Language Programming as a foundation for my program. I spent this week trying to find such resources but came up empty-handed. The closest result I got was an article about TurkeyCode, an application which had the same basis as mine, conversion of natural language into code. Unfortunately, the article turned out to be an April Fool's Day post, thus leaving me with no leads. This failed attempt led me to continue my research regarding creating my own language. My previous assessment dealt more with the technical aspect of the steps and processes involved in the making of a language. Steps such as converting text to another language and then to machine code to run. These steps were important, but the next part is to learn about the design of the language and the elements involved in designing a language. The three most important elements are Syntax, Semantics and Pragmatics.

Syntax refers to the form of the programming language, I believe this is the most important for me to consider at this moment. The form of the language is determinate of the user's ease of interaction with the language. By designing my syntax to have a flow like normal languages, but have enhanced

readability due to its use of words rather than symbols, I can make the language easy to access for beginners, and also make the language a learning tool for transition to other programming languages.

The semantics of a language is the meaning of the programming language. This refers to how the text is interpreted by the compiler or interpreter. What symbols and keywords symbolise is also important at this level. To think about the semantics of the language, I will have to think of how I can reduce the number of symbols from normal syntax to be replaced by more verbose keywords for readability. The interpretation of keywords is also going to vary depending on the form of syntax the language has.

Finally, there is the pragmatics, the implementation of the language. This part of designing a language is not important to me. Pragmatics deals with how the interpretation of the code is dealt with by the computer and how it is computed. This aspect is important when the design of the language is towards particular goals of efficiency, but since my language is going to be a husk over another language I don't need to worry about such implementations.

Based on my research from this week I have decided to further research the syntax and semantic design of a language in detail, using the same resource. I believe that by learning more about such elements of design, I will be more successful in my design for my own language. I could also use this efficiency to finish the language design more quickly.