Project Proposal Outline

Lev Villanueva Guzman

“**M**anagement and **A**utomation of **G**raceful **I**nteractive **C**oding”

Background

Bachelor of Science in Computer Science third semester student currently working solo in the

Previously worked as an English teacher, took three full semesters of Software Engineering at an accredited Peruvian university and courses from the fourth semester. I have created linked Autodesk AutoCAD and Excel plugins for reading and creating AutoCAD elements programmatically.

I’m interested in the understanding and implementation of optimization algorithms, the automation of repetitive tasks through scripts and artificial intelligence in games.

The idea to create a manageable tool that autocompletes or replaces text came from my experience with different tools that allow you to create scripts to automate almost any task in Windows. It’s with this interest that, while taking a summer class in databases, that I created an automation tool for answering calls and making annotations for those calls at the call center I was working at two years ago. Since then, and having worked with a few different text editors, I have noticed there was no easy way to create my own snippets of code or, more generally, auto-replace short text with longer text. Before this project, the most straight-forward approach is to write a script. This solution seems relevant and useful for users of text editors of all sorts using the Windows OS.

Current literature and state-of-the-art

For the Windows environment there are programs such as AutoHotkey and AutoIT that parse scripts and turn them into automation modules or executables. AutoIT is BASIC-like while AutoHotkey has C-esque syntax.

For the problem space proposed, there are commercial tools such as PhraseExpander (<https://www.phraseexpander.com/>), aimed at doctors, and Phrase Express (<https://www.phraseexpress.com/>) for more general purpose. Both of these tools are paid solutions and too complex for the problem this project tries to solve. It should be noted that Phare Express is multi-platform (Windows, iOS, MacOS) and supports database architectures. Another multi-platform application called TextExpander (<https://textexpander.com/>) is available as a paid tool with a much larger scope than this project. Plugins or extensions with similar solutions for the proposed problem space in this document were not found when searched for VS Code, Brackets or Notepad++.

The AutoHotkey (‘AHK’ from now on) programming environment has active and involved members. Commercial courses such as those offered by Joe Glines (<https://www.udemy.com/user/joe-glines/>) are available for asynchronous online learning and free tools such as Maciej Słojewski’s *Hotstrings* (<https://github.com/mslonik/Hotstrings>, <https://www.autohotkey.com/boards/viewtopic.php?t=82352>) uses AHK to solve a similar problem but it suffers from a few shortcomings: it doesn’t support multi-line text expansion (code snippets), it displays terms such as ‘output function’, ‘trigger’ and ‘SendInput’ which will be unknown to users not familiar with AHK, it doesn’t support multiple programming languages and doesn’t have differentiated behavior when interacting with distinct text editor applications.

AutoIT, with an equally vibrant community as AHK, has a similar user-generated text expander project called QuickPhrase (<https://www.autoitscript.com/forum/files/file/448-quickphrase-11f/>), but its scope is smaller than that proposed here.

1. Deliverables

-a-

1) A clearly defined set of deliverable components of the software and the job of work required to bring these components to completion.

2. Project timeline

-a-

2) The defined timescale of work, including any dependencies, milestones or contingencies.

3. System Specification

-a-

3) A formal specification of the desired system (e.g. UML, technical and functional specification.)

4. Project Scope

-a-

4) A clearly defined scope for the project.

5. Requirement solicitation

-a-

5) Some evidence of requirements elicitation involving some/all of your project stakeholders.

6. Research summary

-a-

6) A research summary that highlights the challenges of working within your chosen domain

7. Market Analysis

-a-

7) Evidence that compares your project to similar software tools (e.g. market analysis.)

8. Design approach

-a-

8) A description of your approach that discusses the motivations and reasoning for working in a particular manner (e.g. Agile, User-Centred Design, TestDriven Development.)

9. Prototypes

-a-

9) Some early prototypes showing how the project will work and highlighting the strengths and weaknesses of your proposition.

10. Testing and Design Validation

-a-

10) Some early evidence of assumption testing and validation of your designs to date (e.g. user tests or automated feedback such as W3C validation/accessibility testing, heuristic tests etc.)

11. Critical evaluation

-a-

11) A critical evaluation of your concept, your project in its current state and the proposed software project.