INSTITUTE of TECHNOLOGY CARLOW

Visual Developer tool for Mycroft: Final Report

by

Michal Wojnowski

C00213963

Software Development

Year 4

Supervisor: Joseph Kehoe

Date of Submission: 20/04/2020

Abstract

The aim of this project is to build a visual development tool in the vain of Scratch which would allow the users to easily build skills(apps) for Mycroft which is an open source voice assistant for Linux. This application should allow the users to create a skill without having to write any code by just dropping visual blocks on a canvas with prewritten code in the. The only thing that a user will have to do is answer a short questionnaire to create a skill template and select the file

Table of Contents

Introduction

This is the final report of the development process for the Visual Development Tool for Mycroft. The objective of this document is to give an overview of the project and its development cycle as well as some issues and pitfalls that project got into. This document will be divided into 5 sections with each section giving an in-depth overview of its respective topic

Section 1: *Project Description* - The first section will give a general project overview of what it is and its functionality, the technologies used and a step by step walkthrough of using the app. Furthermore, the section will be finished off with the hurdles that the project faced and how they were solved.

Section 2: *Comparison to Original Design* - In the second section the final product will be compared to the original design outline in previous documents to see how the application changed and evloved

Section 3: *Learning Outcome* - Third section of the document will be a reflection on what I have learned during the development process of the project

Section 4: *Project Review* - Finally the last section of the document will cover the reflection of the project as a whole, how I believe it turned out, what are the success and failures of the project

1. Project Description

The Visual Development Tool for Mycroft is a Python based application designed for Linux which allows the user to create skills (apps) for the Mycroft AI voice assistant by combining fitting blocks together which allows users with no to little programming experience to easily create skills for Mycroft. The way this was achieved was that a Graphic User Interface (GUI) using Tkinter was created which gave the user access to all the tools they would need to create and edit skills without having to write a single line of code. Sublime Text was used as the base code editing tool due to its easy availability of Linux and support using it for Python.

Tkinter was selected as the GUI framework for the project due to its easy learning curve, robust toolkit and widespread use which allowed for swift problem solving reducing the gaps in feature development. Moreover its canvas feature allowed for a simple creation of robust and limitless shape creation to serve as the project's code blocks.

1.1 Project Walkthrough

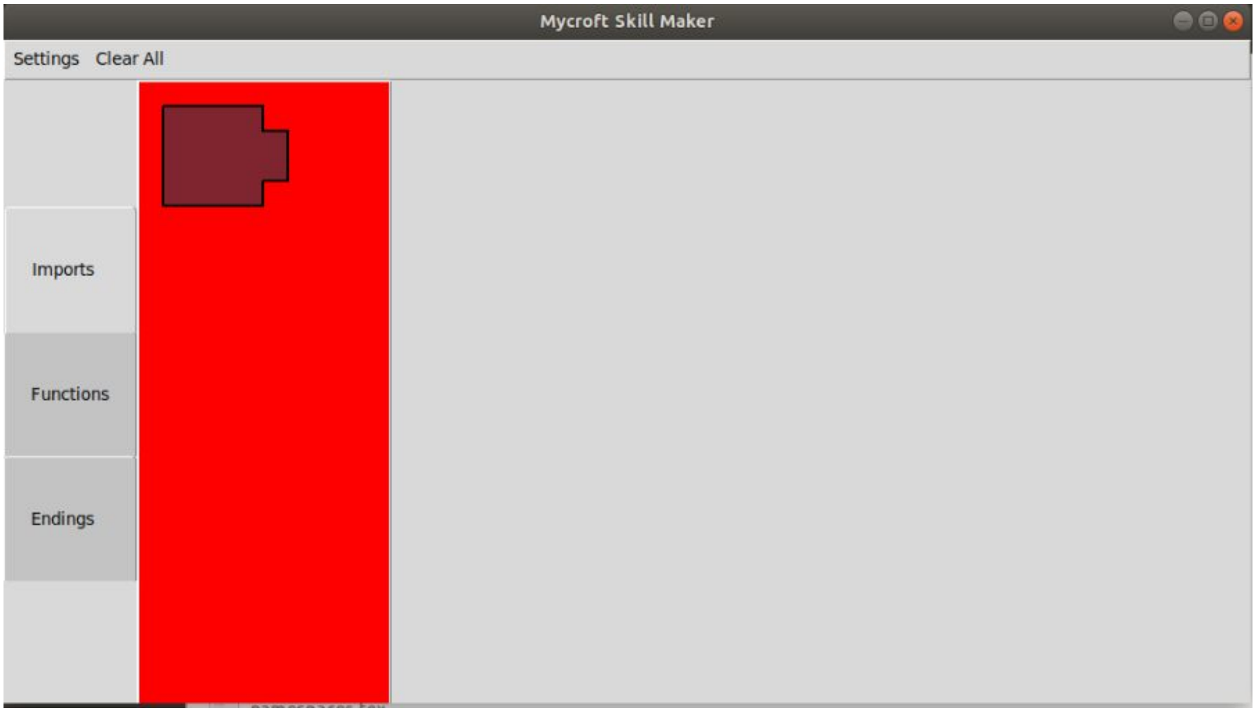


Fig 1. Opening screen.

One the user turns on the application they will be greeted by the default screen

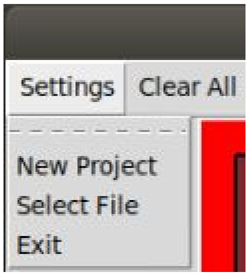
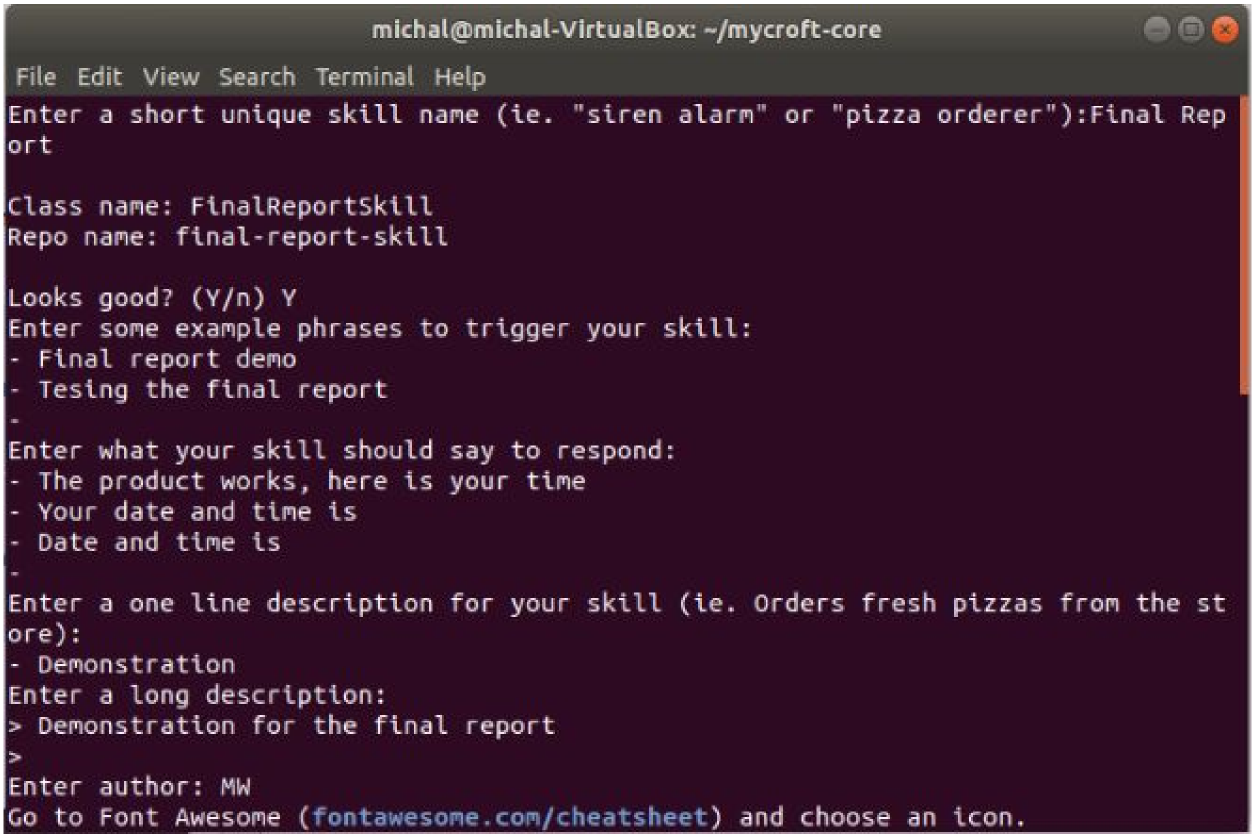


Fig 2. Settings bar

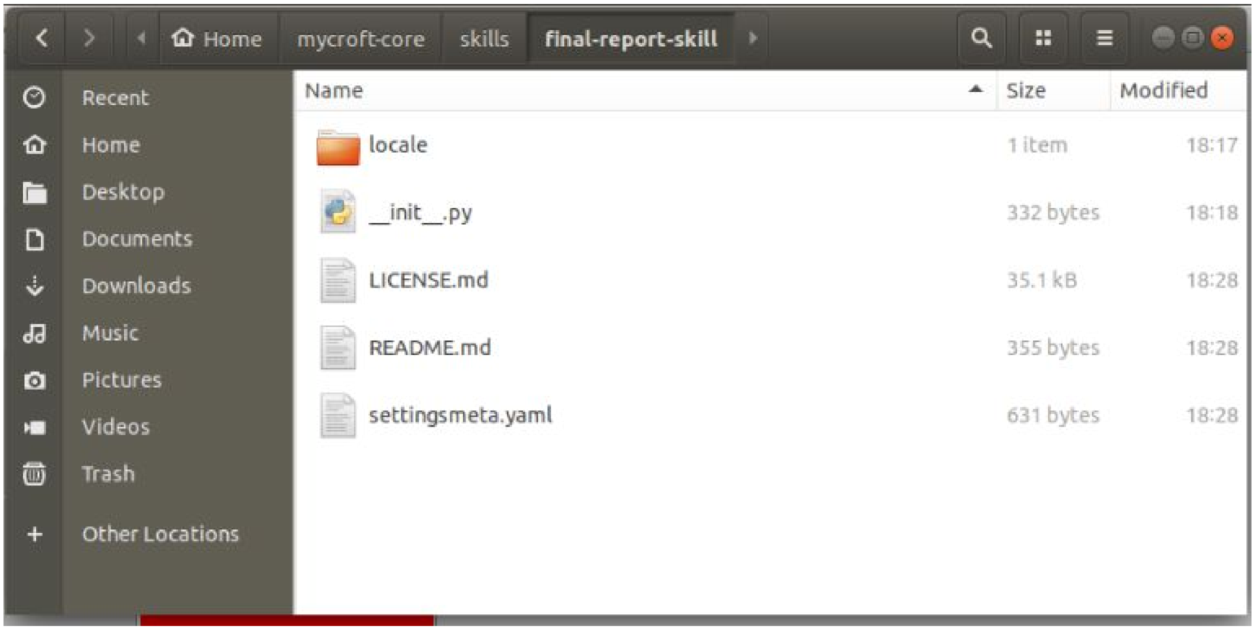
Once the “ Settings “ button is clicked the app will present the user with three options. First will allow the user to create a new project, second allows the user to select the project they wish to work on and finally to exit the app.


michal@michal-VirtualBox: ~/mycroft-core
File Edit View Search Terminal Help

```
Enter a short unique skill name (ie. "siren alarm" or "pizza orderer"):Final Rep
ort

Class name: FinalReportSkill
Repo name: final-report-skill

Looks good? (Y/n) Y
Enter some example phrases to trigger your skill:
- Final report demo
- Tesing the final report
-
Enter what your skill should say to respond:
- The product works, here is your time
- Your date and time is
- Date and time is
-
Enter a one line description for your skill (ie. Orders fresh pizzas from the st
ore):
- Demonstration
Enter a long description:
> Demonstration for the final report
>
Enter author: MW
Go to Font Awesome (fontawesome.com/cheatsheet) and choose an icon.
```

Fig 3. Skill Creation

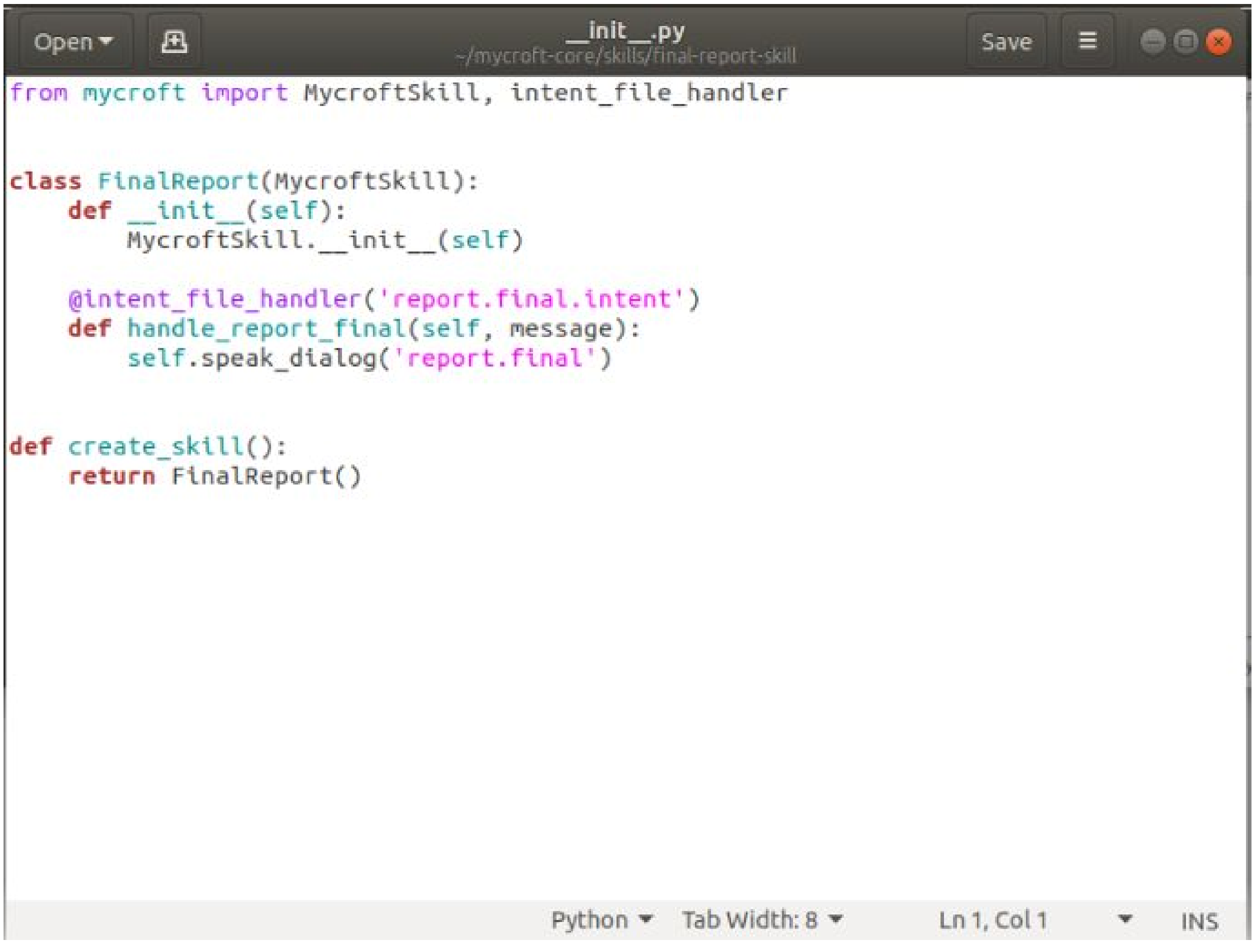
If the user chooses to start a new project a command prompt will open up on their screen with a short questionnaire allowing the user to outline the basics of what they wish their skills to be by choosing its name, what phrases it will be activated by and respond with as well as some more technical questions like the licence they wish to apply to their product and if they wish to create a github repository .



Name	Size	Modified
locale	1 item	18:17
__init__.py	332 bytes	18:18
LICENSE.md	35.1 kB	18:28
README.md	355 bytes	18:28
settingsmeta.yaml	631 bytes	18:28

Fig 4. Created skill folder

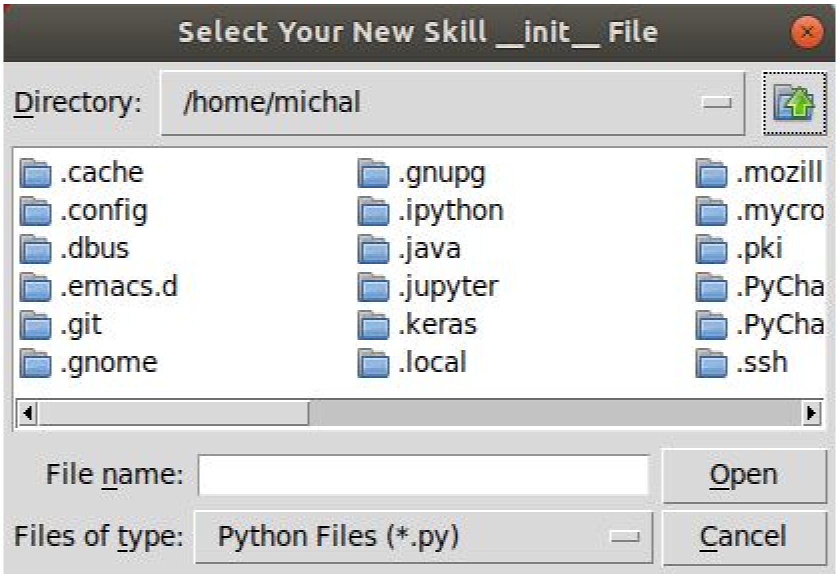
Once the questionnaire is complete it creates a folder in the main Mycroft skills directory. It consists of the auto generated skill template, alongside this licence of choice will be created as well the user input trigger and response phrases.



```python
from mycroft import MycroftSkill, intent_file_handler


class FinalReport(MycroftSkill):
    def __init__(self):
        MycroftSkill.__init__(self)

    @intent_file_handler('report.final.intent')
    def handle_report_final(self, message):
        self.speak_dialog('report.final')


def create_skill():
    return FinalReport()
```

Fig 5. Template code

The questionnaire will generate a standard template code for all skills. At its base the code responds to a trigger with a response. While it's action is not advance, the purpose of this code is to give newer users a guideline on skill creation.



Directory: /home/michal

Files of type: Python Files (*.py)

Fig 6. Select File

If a user selects “ Select File “ a popup window will open allowing them to search through the files on their computer until they find the one they wish to edit. The ability to only view Python files is set as default in order to help new users find the file they’re looking for however if the user wishes they can change the view type to see all files.

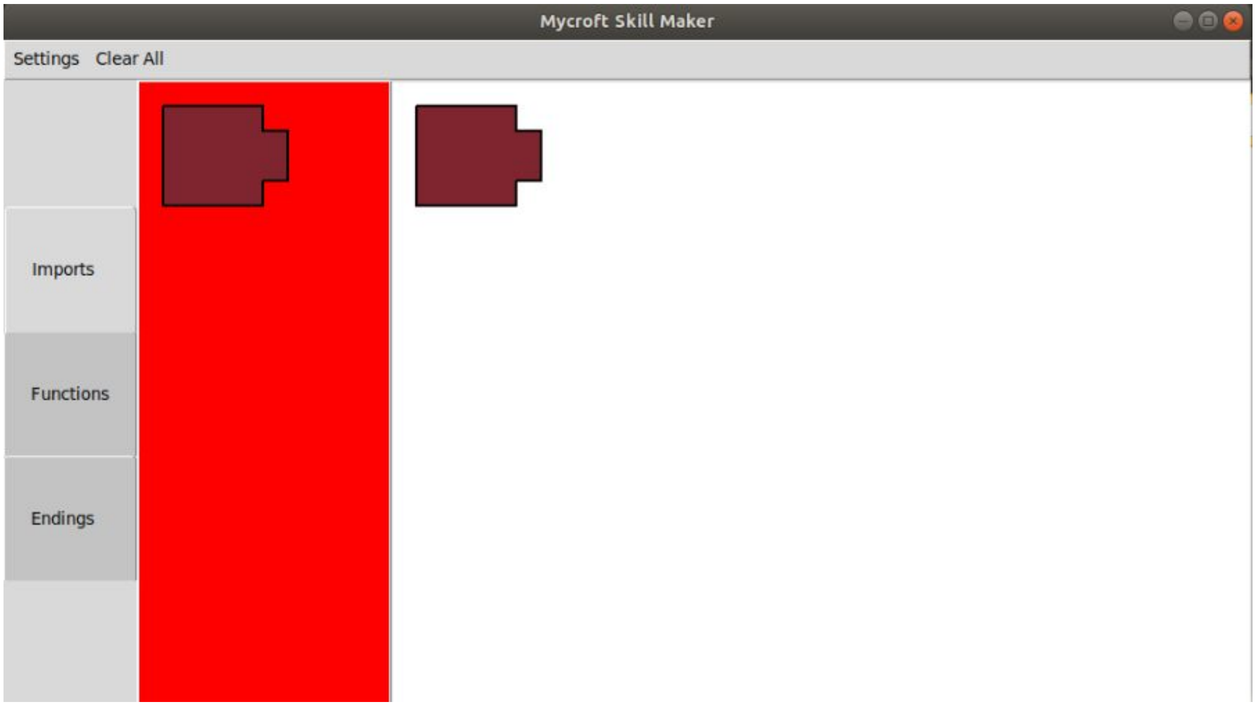
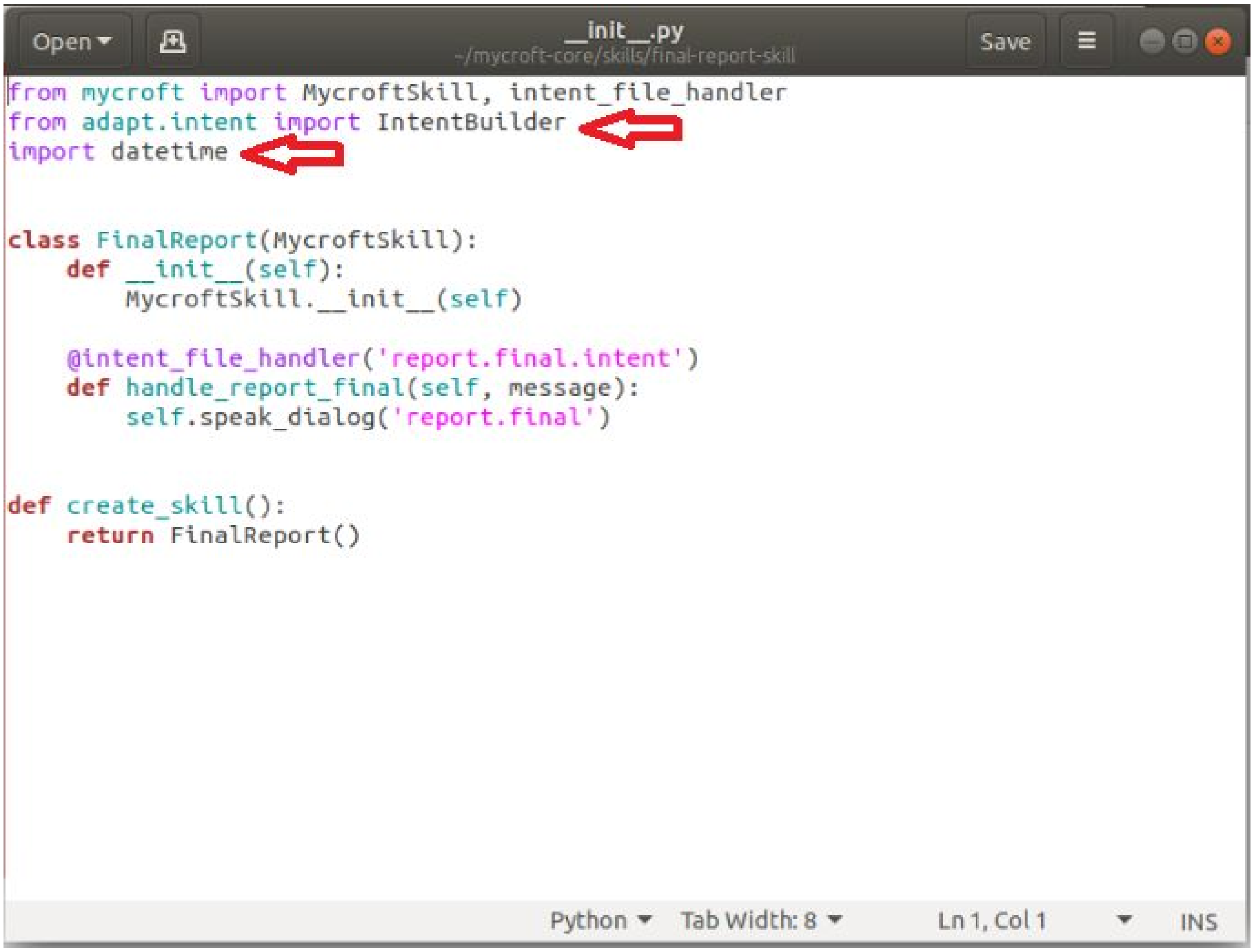


Fig 7.Placing a block

When a file has been selected the user will click on a block corresponding to the type of skill they wish to create, the code will write its code in its predetermined location.



```python
from mycroft import MycroftSkill, intent_file_handler
from adapt.intent import IntentBuilder
import datetime


class FinalReport(MycroftSkill):
    def __init__(self):
        MycroftSkill.__init__(self)

    @intent_file_handler('report.final.intent')
    def handle_report_final(self, message):
        self.speak_dialog('report.final')


def create_skill():
    return FinalReport()
```

Fig 9. Added code

The highlighted code has been inserted by the insertion of a block shown in Fig 8. This process is repeated for the remaining blocks in figures 10 to 13

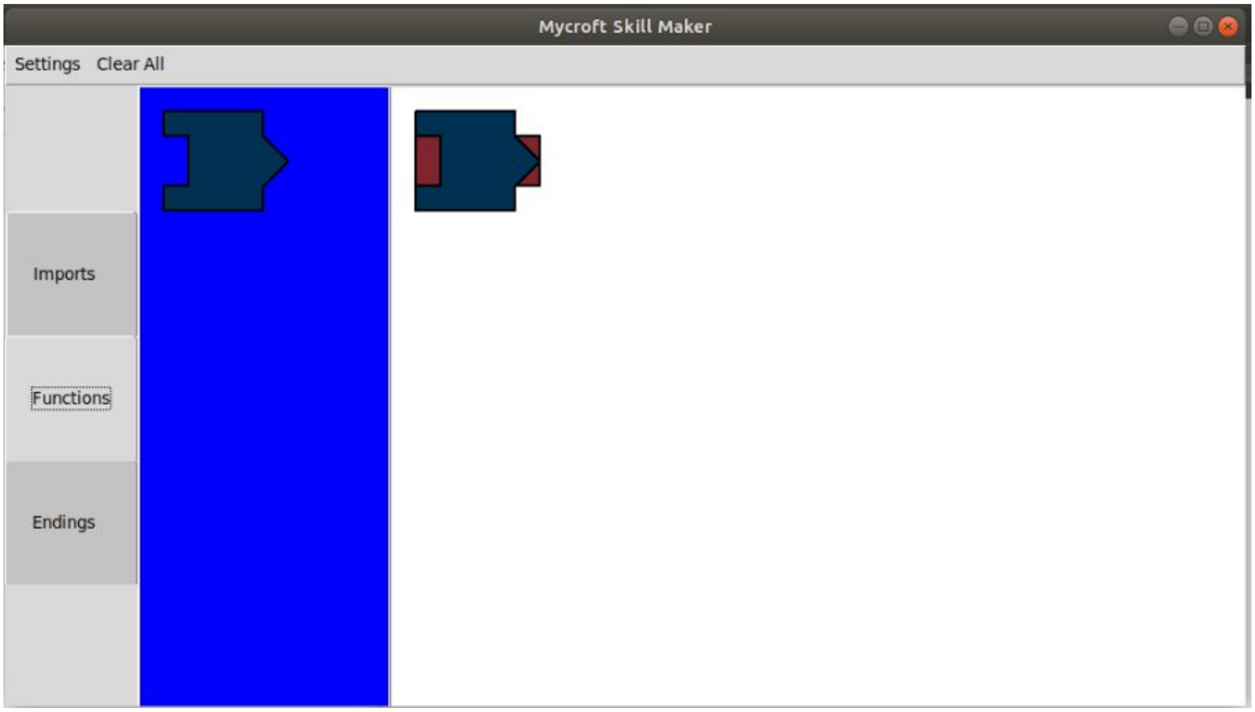
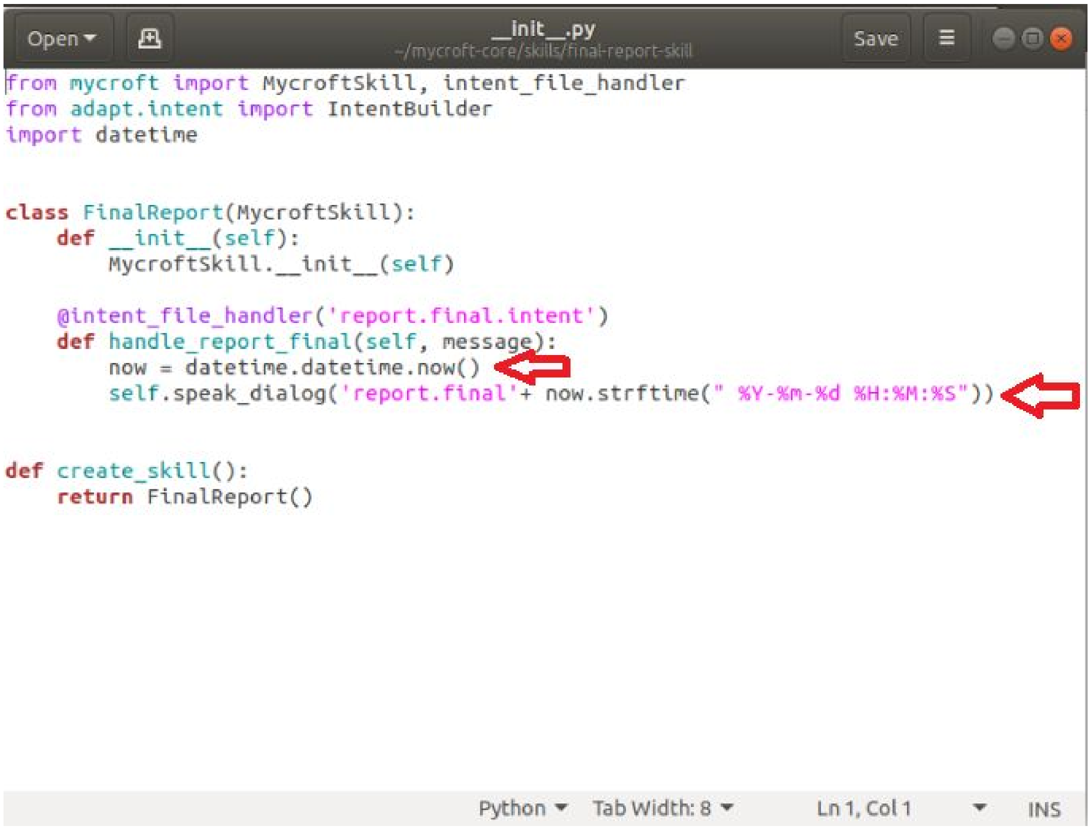


Fig 10



```
from mycroft import MycroftSkill, intent_file_handler
from adapt.intent import IntentBuilder
import datetime


class FinalReport(MycroftSkill):
    def __init__(self):
        MycroftSkill.__init__(self)

    @intent_file_handler('report.final.intent')
    def handle_report_final(self, message):
        now = datetime.datetime.now()
        self.speak_dialog('report.final'+ now.strftime(" %Y-%m-%d %H:%M:%S"))


def create_skill():
    return FinalReport()
```

Fig 11

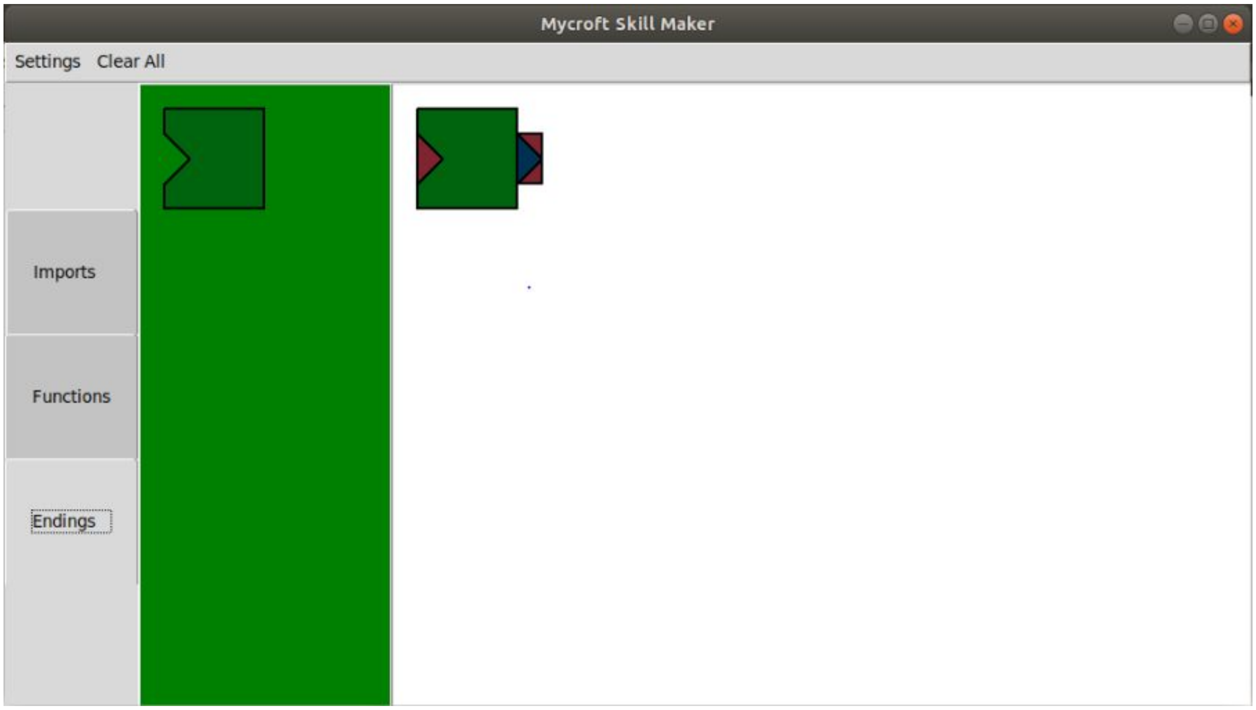
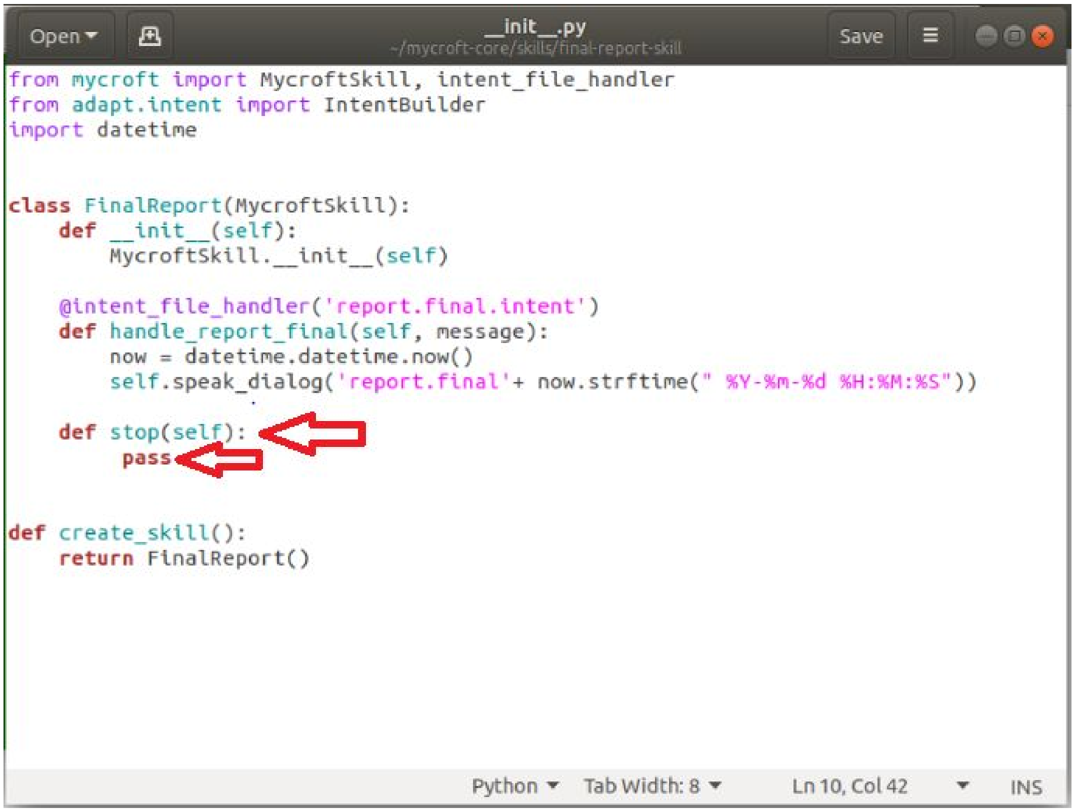
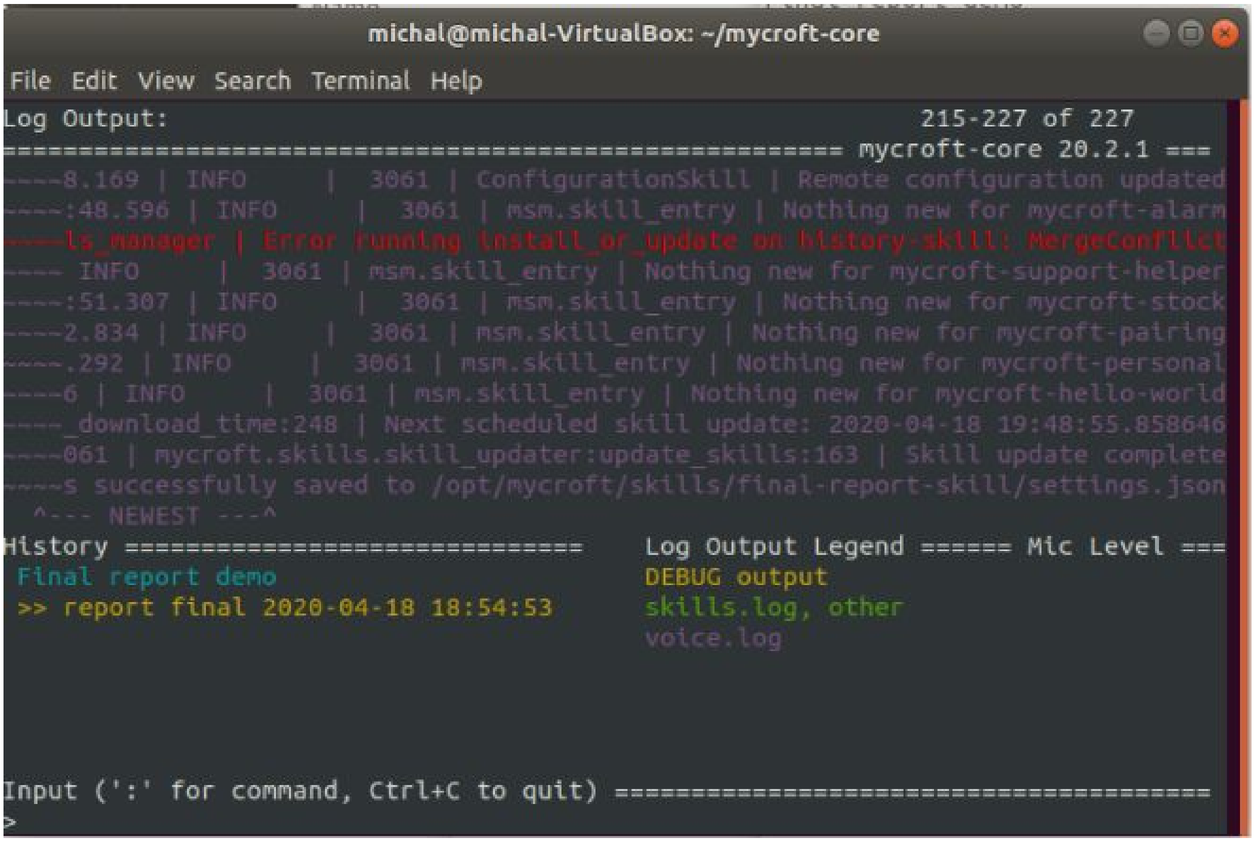


Fig 12



```
from mycroft import MycroftSkill, intent_file_handler
from adapt.intent import IntentBuilder
import datetime


class FinalReport(MycroftSkill):
    def __init__(self):
        MycroftSkill.__init__(self)

    @intent_file_handler('report.final.intent')
    def handle_report_final(self, message):
        now = datetime.datetime.now()
        self.speak_dialog('report.final'+ now.strftime(" %Y-%m-%d %H:%M:%S"))

    def stop(self):
         pass


def create_skill():
    return FinalReport()
```

Fig 13


michal@michal-VirtualBox: ~/mycroft-core
File Edit View Search Terminal Help

```
Log Output:                                             215-227 of 227
======================================================= mycroft-core 20.2.1 ===
~~~~8.169 | INFO     |  3061 | ConfigurationSkill | Remote configuration updated
~~~~:48.596 | INFO     |  3061 | msm.skill_entry | Nothing new for mycroft-alarm
~~~~ls_manager | Error running install_or_update on history-skill: MergeConflict
~~~~ INFO     |  3061 | msm.skill_entry | Nothing new for mycroft-support-helper
~~~~:51.307 | INFO     |  3061 | msm.skill_entry | Nothing new for mycroft-stock
~~~~2.834 | INFO     |  3061 | msm.skill_entry | Nothing new for mycroft-pairing
~~~~.292 | INFO     |  3061 | msm.skill_entry | Nothing new for mycroft-personal
~~~~6 | INFO     |  3061 | msm.skill_entry | Nothing new for mycroft-hello-world
~~~~_download_time:248 | Next scheduled skill update: 2020-04-18 19:48:55.858646
~~~~061 | mycroft.skills.skill_updater:update_skills:163 | Skill update complete
~~~~s successfully saved to /opt/mycroft/skills/final-report-skill/settings.json
  ^--- NEWEST ---^
History ===============================    Log Output Legend ====== Mic Level ===
 Final report demo                         DEBUG output
 >> report final 2020-04-18 18:54:53       skills.log, other
                                           voice.log




Input (':' for command, Ctrl+C to quit) =======================================
>
```

Fig 14. Working skill

After all of the blocks have been put in place, the skill is tested out. In this case the skill built was skill that informed the user of the current time and date. After the trigger set in Fig 3 was called, the skill gave back the pre-set response with the time and date

1.2 Development Challenges

1.2.1 Mycroft’s Obscurity

The main obstacle that kept appearing during the development process was the lack of resources when it came to development for Mycroft. Mycroft is still a fairly unknown product with only around 35,000 users (Mycroft AI is fundraising on Crowdfunder, n.d.). As with any niche technology, the support for Mycroft is quite limited which increases the learning curve of the technology and the time it takes to solve problems. .

There is a significant lack of tutorials and problem solving forms for Mycroft, and those that exist are often around 3 years old which is when the technology first launched. This results in the information from the tutorials no longer being applicable to the current technology and a lack of solving threads on forums. Combination of both of those problems resulted in a significantly higher proportion of learning and problem solving compared to development

1.2.2 Code inserting for loop

While it is a very specific problem, it brought development to a halt for about a week and a half. This issue arose during the skill development cycle where a method for code insertion was being implemented. The implementation of choice was a for loop using the FileInput Python library which read in the chosen file and would insert the code following a specific line in the template code.

```
for import_line in fileinput.FileInput(folder_path, inplace=1):
        if 'from mycroft import MycroftSkill, intent_file_handler' in import_line:
```

The issue was that the for loop was being completely ignored for no apparent reason. After a week and a half of problem solving it was discovered that due to Python’s dynamic writing the “ folder_path “ which was being set by users.

```
folder_path = filedialog.askopenfile(mode = "a+",title="Select Your New Skill __init__ File",
filetypes=(("Python Files", "*.py"), ("All Files", "*.*")))
```

Was not being set as a string, this issue was solved by adding “.name “ to the end of “folder_path” which changes the object’s name into a string. The end result was.

```
for import_line in fileinput.FileInput(folder_path.name, inplace=1):
```

```
if 'from mycroft import MycroftSkill, intent_file_handler' in import_line:
```

1.2.3 Hardware Limitations

The laptop that this project was done on is a seven year old HP Pavilion g6 with a i3 with 8GB of ram. In addition due to the laptop operating on the Windows OS and the project being designed for Linux, the laptop had to run a VirtualBox in order to be able develop the project. This caused several issues ranging from the laptop freezing and having several seconds of input delay to the battery being drained in about an hour, this caused increasing issues after the Covid-19 lockdown was put in place due to the limited amount of power outlets.

1.2.4 Covid-19

The Covid-19 pandemic caused numerous problems for this academic year with the main being the closure of all non-essential businesses. While on the technical level the project was not affected too heavily, the development process slowed down due to personal reasons such as the inability to call someone over to take a look at your code or discuss it with them. This is a very useful aspect of work in an office/shared area as a fresh pair of eyes can easily find the issue that you’ve missed after staring at the code for hours, much like explaining your code to someone, hence the infamous rubber duck.

2. Comparison to original design

The project stuck fairly closely to the original feature set and design with no major alterations since the Project Research document. The main decision that was taken during the development process was to drop database integration. This decision was made as Mycroft itself already has GitHub integration, allowing the users to have quick and easy access to cloud storage and version control of their skills.

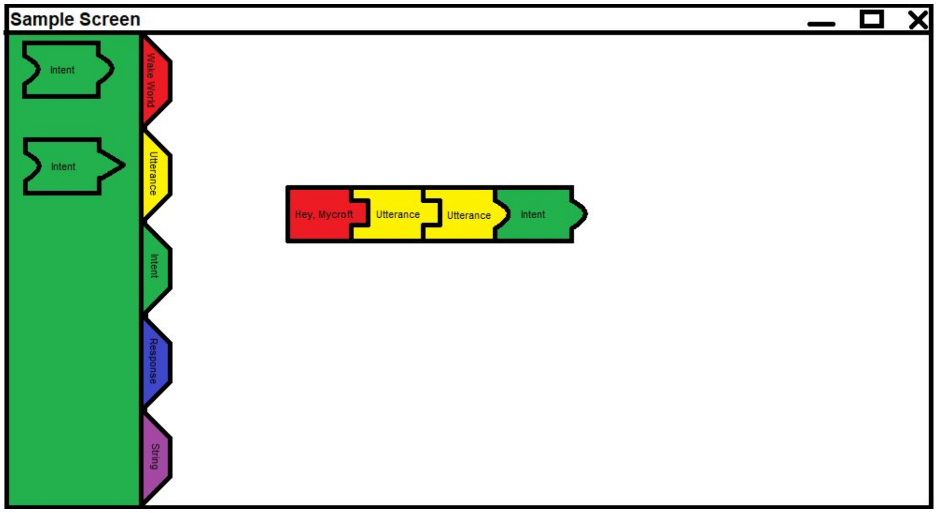


Fig 15. Proto project design before start of development

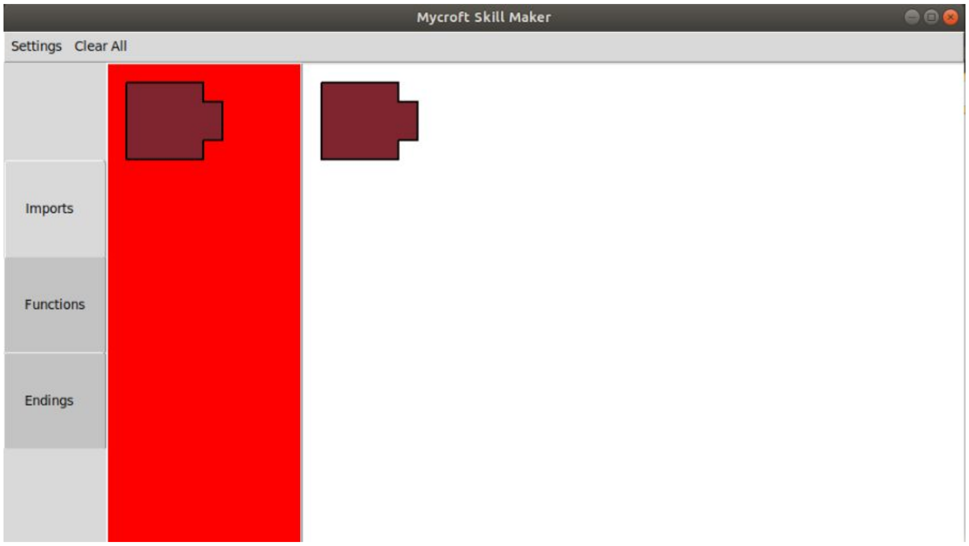


Fig 6 . End result project design for comparison

3. Learning Outcome

3.1 Technical Learning Outcome

The project required the use of some technologies I was unfamiliar with or were a weak link in my abilities, which is why this project was chosen as using and learning these technologies would increase the value of the learning outcome.

3.1.1 Python

Python was not only the programming language that Mycroft was written in but it was also one of my personal weakest skills which made it the obvious choice for this project.

Thanks to Python's popularity it was simple to find guides, tutorials and other information resources needed for this project's development. While the beginning stages of the project allowed for rapid development due to the wealth of information about Python, it slowed down or even grinded to a halt towards the later stages. This was due to a almost complete lack of information about Mycroft and its skill building process. As mentioned before Mycroft had one video tutorial and documentation to help the user understand the process, however the tutorial was three years old and not all the information was valid at this point meanwhile the documentation didn't answer all of the questions. At this point the learning process shifted away from finding information about a subject online to figuring out information on my own. While this process consumed a large amount of time it did give me a better understanding of how Python, and Mycroft skills work.

3.1.2 Tkinter

Tkinter is the default GUI toolkit for Python as it is the only one that is included in the base install of Python. As a result of its popularity, information on it was relatively easy to find and allowed for a quick development of the user interface with the exception of one visual bug and missing text on the blocks. However this is due to issues specific to the project which makes the solving of these issues time consuming and with the limited time on the project it was not worth time wise to fix.

3.2 Personal Learning Outcome

3.2.1 Time Management and Prioritisation Skill

While time management and prioritization could often be confused as one and the same thing, they are quite distinct and both are needed equally. This project required the performance of a challenging balancing act between it, all of the other tests and assignments we got during the year, my part time job and personal life. Like many other students I had to prioritize the most urgent thing at the time, be it a test or an assignment. However, when the Covid-19 lockdown started, all other tasks were put on hold and I was able to focus on the project.

In order to try to maximise the development of the project I adopted the Kanban strategy which I learned during my internship. This method only allows the developer to take up two tickets to work on at the same time, or features in my case. This allowed me to prioritize certain elements of the development process while still having flexibility to change between taks if I got stuck. Thanks to this all of the features that had its development started were implemented and it prevented time being wasted on side features that were started as a distraction from the main issue. This proved of great help towards the end of the project when the development was halted by the for loop and skill creation difficulty as development remained focused on those features.

However the Kanban process also had some downfalls, mainly time management. This was an issue as following the process prevented the user from moving onto another ticket/feature if the current one ran over its allocated time. While a ticket could be place on hold or another ticket taken up this would defeat the entire point of the Kanban process,

Overall I believe that the adoption of the process helped the project keep on track and get all of its main features in place.

4. Project Review

The project was a limited success when compared to the original specifications. It did achieve its core features set of skill creation for Mycroft without the end user typing in any codem however it is limited in its feature set. Currently the application allows for the creation of one skill and is missing some desired features like an undo button, a help window to help new users understand the program and learn how to use it and clear labeling on each block on what it does.

The current version of the project is a solid framework that allows for the development of new skill blocks by copying pre-existing code leaving only the variables like inputting the desired parameters for the shape and code that will be written as well as its location in the file to be done.

If the project had to be started all over, the first objective would be to create several skills of similar nature to give the direction of how the blocks should look like and how they should be divided. This way the core feature set would set the design of the GUI and would allow the neat instant implementation of the core feature as soon as the GUI framework was in place.

Declaration

- I declare that all material in this submission e.g. thesis/essay/project/assignment is entirely my/our own work except where duly acknowledged.
- I have cited the sources of all quotations, paraphrases, summaries of information, tables, diagrams or other material; including software and other electronic media in which intellectual property rights may reside.
- I have provided a complete bibliography of all works and sources used in the preparation of this submission.
- I understand that failure to comply with the Institute's regulations governing plagiarism constitutes a serious offense.

Student Name: Michal Wojnowski

Student Number: C00213963

Signature: M.Wojnowski

Date: 20/04/2020

Reference

Crowdfunder.com. n.d. *Mycroft AI Is Fundraising On Crowdfunder*. [online] Available at: <https://www.crowdfunder.com/mycroft/invest#kpis> [Accessed 19 April 2020].