Personal Information

Lachlan O’Neill

Student Number: s3802024
Nationality: Australian
Education:

- Queensland Certificate of Education

Profile:

Hi, my name is Lachlan O'Neill, I'm 17 and I live in Queensland, more specifically, Brisbane, I work as an administrator for a disability support company on the Gold Coast. I graduated Year 12 last year. I was originally going to study a diploma of Software Development at TAFE and then continue with a Bachelor of IT with QUT afterwards, but I thought I might as well try and apply for the qualification through OUA anyway. I have Cerebral Palsy so I'm dependent on a wheelchair, it limits my capabilities somewhat, but I'm not too restricted in everyday activities. This also explains my choice in hobbies and my gamertag, “Wheelz”! Some of my hobbies include video games, Dungeons and Dragons, and reading.

Tools

My website can be found here, the website is still very much a work in progress and may behave weirdly on monitors not outputting at 1080p resolution. The website details facts about myself as well as the IT project I have been working on. The repository for the website can be found here and the repository for the actual project can be found here.

You’ll notice on the project repository that there are commits from people other than me. This is because I was originally working on this project with a group however the people working on the assignment with me withdrew from the course. The others in the group were non-communicative so it was decided that I would continue this project as a solo endeavour.

Project Description

As a competitive Pokémon fan, I am frustrated with the lack of support the developers have given the community regarding producing competition ready teams. I'm consistently disappointed in the minimal progress Game Freak has made to fast track the team building process. Things like breeding for IVs and natures take hours for each Pokémon not to mention the extra effort required to get those stats plus any egg moves needed.

To be fair to Game Freak, it has made steps to make this process easier, the streamlining of EV training and mint items for specific natures come to mind. But in my opinion, this isn't enough. If there was a way to build teams in a matter of minutes instead of a few days, I feel that it would be a huge benefit to the community.

That's why for my IT project I have decided to develop a tool that streamlines the Pokémon team building process, allowing users to generate teams to use in their games in a manner of minutes.

Overview

Topic

With this project I am aiming to produce a program that allows users to generate a Pokémon team as a plain text file which can then be imported into external tools like the save editor PKHex or the battle simulator Pokémon Showdown. The program will take input from the user, letting them dictate how the Pokémon is built, down to the stats, held item, moves, etc.

With this tool I believe the barrier for entry into the competitive Pokémon scene will be lowered drastically allowing new players to quickly generate a team to use online. Afterwards if they need to tweak something, they can edit the file themselves to change whatever is needed. This leads to much faster fine-tuning of strategies and combinations.

Motivation

My motivation for this project is very personal. As stated earlier I am very frustrated with the developers of Pokémon for making the barrier for entry into the scene so high. This annoys me because it is a big deterrent for people interested in the community. If they see what is required to even start playing competitively as a basic level than they may be discouraged and be disinterested in joining the scene. I am always very happy to see new faces in the community and I think that with this tool more people will be able to explore that interest in competitive battling.

Landscape

The landscape for this type of tool is very new. From my research there hasn't ever been a tool like the one I am proposing. As such it would benefit the Pokémon community greatly, introducing new faces to the competitive battling scene and with those new faces, new and unorthodox strategies, thus revitalizing the game and making the community more active.

Detailed Description

Aims

My aim for this project is to make getting into the competitive Pokémon battling scene as easy for newcomers as possible as well as make it easier for veterans of the scene to come up with and test new strategies and concepts. This would bring new life into the scene and make it more active, fun and thus more enticing for new players repeating the cycle and growing the scene.

To do this I'll need to remedy the awfully long amount of time it takes to get a team up and running. People that just hop into battles with their story team will be in for a rude awakening and be discouraged from joining the scene when they learn how long it takes to make a team. Breeding for the right stats is awful and time consuming let alone getting those IVs and the right nature and moves. It's the worst.

Plans and Progress

The aim of this project is to create a tool that generates Pokémon teams as a plain text file in the format described below:

```
Species @ Item
IVs (If specified): # Stat / # Stat / # Stat
EVs: # Stat / # Stat / # Stat / # Stat
Ability: Ability Name
Shiny: (Yes/No)
(Nature Name) Nature
- Move 1
- Move 2
- Move 3
- Move 4
```

This format has been adopted by the Pokémon community as a quick and efficient way to share and edit teams, allowing for other people to test the team and quickly provide feedback as to what could be improved or changed to better suit the person's playstyle or desired strategy.

In order to produce a program that accomplishes these goals I need to write a few key features:

1. I need a few classes that stores the aspects of the Pokémon (i.e. one for EVs, one for IVs and one for move sets etc.) and returns those values when needed
2. I need a class checks if the EVs and IVs are legal (IVs are limited to a minimum of 0 and a max of 31 per stat. And EVs can have a minimum of 0 and a max of 255 per stat but the total number of EVs cannot exceed 510).
3. I need a class that stores all the Pokémon's data in one place so that it can be easily accessed.
4. I need a class that accesses the Pokémon's data and outputs it in a text file in the correct format.

Once I get all of these done the essentials of the program will be complete and I can focus on extra things if I want. Some things that I may want to include are:

- Checking if the user spelt the Pokémon’s name correctly.
- Checking if the user spelt a move name correctly and whether the Pokémon can learn that move in game.
- Checking if the user spelt the ability name correctly and whether the Pokémon can have that ability in game.
- Checking if the user spelt the item name correctly.
- Checking if the user spelt the nature name correctly.

Basically, I would like to implement a whole lot of error checking, but I feel that this would be far outside the scope of this assignment as there are currently 890 Pokémon, 796 moves, 258 abilities and too many held items to count. As you could probably tell, implementing all of this in the time frame that I have is impossible. If I were able to get this all implemented and working properly in my own time then I would enhance UI elements like adding the appropriate Pokémon and held item sprite, the move’s typing, whether the move is physical or special, etc.

Scope and Limits

The scope of this project is quite large. Implementing error handling for every step of the generation process is impossible in the given time frame. As such, it is best to just focus on the essentials in order to get the program working on a basic level. The four classes required for the key features of the program are the ones that I need to focus on. Checking EVs/IVs, getting the data storage working and making sure the input/output system works correctly is the minimum required work for making the program function.

After the deadline is met and the project is handed in, then I can work on implementing error handling for different situations. This’ll be a huge undertaking and will probably take months to get working properly but once that is done, I can focus on freshening up the UI.

Tools and Technologies

For this project I’ll be using Visual Studio Code version 1.42 as an IDE to write the program. GitHub to keep a backup of the repository online and Red Hat OpenJDK and -JVM as the version of Java used. As for the video portion of the project I’ll be using Lightworks 14.5 to create and edit the video presentation. All these pieces of software are free to download and use, and all the software used for the creation of the program is open source.

Testing

Testing and debugging is a vital part of the development process. It's important to test a program every time a major feature is implemented, some would even argue it's vital to test the program every time a line of code is added that's dependant on an existing variable or a previous line of code. For smaller projects, I think it's adequate to test each time a feature is implemented, however if you are working on a large-scale project like a game then testing each line is warranted.

For this project, it's important that I test the stat checks for EVs and IVs as well as the text output portion of the program. I will create a test class that generates a hard coded Pokémon and outputs it to a file in the format needed, it'll also tell me whether it's EVs and IVs are illegal, I can just change values to something illegal and it should pick up on it.

Once testing is done with the hard-coded Pokémon, I can move on to implementing user input. Of course, I'd like to implement error checking for each step of the generation process here, but that's impossible given the one-man team and time constraints. If I had a larger team and more time than I could get the checks implemented and working properly with some more debugging.

Timeframe

With six weeks work on this project I will be able to write-up, test and upload a proof of concept program to the GitHub repository. It will have user input, text file output and some error checks for the stats. Given a further 10 weeks I could make a start on implementing error checks on Pokémon species spelling, item spelling, move sets etc. Making sure the Pokémon's name is spelt right is probably the next most important step in the error checking process after the stat checks. But considering that there are 890 Pokémon at the time of writing it'll take a while to get that written-up and working properly.

To be honest, it'd probably take me about a day and a half or non-stop coding to get the proof of concept done. Considering that the report is the most important part of this project I'll have plenty of time to work on it as well as the presentation if I get the program done as soon as possible.

Risk

The main risk with this project, and most programs, is the end user. You must try and account for anything and everything the end user will do that will screw with the program to make sure you don't get any unintended crashes, CPU hangs, etc. Anything you can think of that the user *might* do, implement an exception or check for. Always assume the end user is a moron and try to code around it. If the program asks for an integer, put a check in place that checks for characters or strings and throw an exception, so you don't get a crash. Implementing even the simplest of checks can go a long way to improve the programs stability and make it less volatile.

As for my project specifically the main risk I can identify is that the text file is created in the wrong place. I've coded it so that the text file is created in the project's root directory to avoid this however, so that wherever the program is on the user's computer it'll always create the file in that location. If the text file is somehow generated in the wrong place though, the program will still work as intended and will just create a new file in the intended directory, this is a known bug that has been fixed.

Skills and Jobs

With extra funding I'll be able to hire a team of four that I will direct to develop this project with a deadline six months from now. My team of four will likely consist of two programmers, and two UIX designers. The two programmers will be able to help me implement the additional error check along the generation process, as well as any miscellaneous checks required. The UIX designers will be able to freshen up the interface of the program and make it nicer to use, adding the visual elements I listed at the end of the *Plans and Progress* section of the report.

Job Descriptions

Programmer

Looking for two programmers/software developers to help write a tool to complement the *Pokémon* series of video games, the tool will allow users to generate Pokémon to be used in competitive battles. Must have a year or more experience with Java, must be comfortable working in a team environment and have good time management skills.

UIX Designer

Looking for two UIX designers to help work on a tool to complement the *Pokémon* series of video games. The tool will allow users to generate Pokémon to be used in competitive battles. You will help design a user interface that will make the program easier to use and easy on the eyes. Must have at least a year’s experience with Java and experience and/or qualifications in user interface design. Must also be comfortable working in a team environment and have good time management skills.