

For my Capstone Project, I would like to make a modification for the video game Minecraft. I would like to introduce additional content and mechanics to enhance the experience of the game. This modification would be for entertainment purposes, with the goal of simplifying the modification process and reducing the stress that a machine might face when needing to download potentially hundreds of different modifications to achieve a single effect. My goal as a developer is to work for a large video game company, such as Bungie, Blizzard, or Mojang, and eventually create my own game. I believe that making a modification instead of an application only used to search an api would help boost my portfolio and jumpstart my career. If I can show video game companies that I already know my way around game code, then the interviewing and hiring process would be significantly easier.

This modification will be written in Java, which is the language that the original game is written in. I do not plan on using an API, although that may change. I will be getting all of my necessary data from the game's original code. This modification will be backend-focused, with small frontend tasks such as designing icons for added content and making 3-D models to put into the game. The backend content will be focused on adding these additional items and mechanics into the game, and getting the game to recognize the new additions.

The first goal of this project is to expand my portfolio and publish my first game development content. I plan on having a career in game development, and having a modification to an existing game is an excellent precursor to making a game of my own. It would show companies that I know how to adapt to different languages, read existing code, and help improve and add to it. The second goal of this project is to bring a simpler experience to others looking to use modifications. The majority of the time, people put together very large packs of hundreds of modifications to achieve a single effect. This often puts stress on the machine, as it has to load and store a lot of extra and unnecessary data. My modification will condense hundreds of other modifications into one simple program that will be easy to apply and use. It will be easy to navigate, and ideally be low-stress on the desired machine.

The demographic of this program will be very large. It will be for all ages, and all types of people. My program will be focused on expanding the botanical possibilities, so the demographic will be people interested in different types of plants and teas. People who enjoy the original Minecraft, Stardew Valley, Fields of Mistria, and other cozy games will enjoy this program. No matter who uses this modification, I want it to feel light, happy, and simple for everyone.

As I am not making a search based application, I do not plan on using or making an API. I have multiple search applications in my portfolio, so I do not feel that it will hurt my portfolio. If there is a point where I need an API, there are multiple available options for me. Instead of using an API, I will be pulling data directly from the game's code. I will be using the game's code blocks as templates to add in additional items. There are already different types of plants, flowers, and crops in the game, so I will use those code blocks as templates and examples of how my code should be structured for maximum efficiency and integration. There are currently many other modifications available that can help to answer questions that I might have. I can

look into other modifications to see how they solve certain problems, and I can learn from their mistakes. I am very experienced at keeping large quantities of data organized, so I do not see that becoming an issue.

I have started to look into the process of building this application, and the steps are simple. To start, I need to become fluent in Java, which I have already begun. Because I know a few other programming languages, it's easier to learn Java all the way through. After I am familiar with Java syntax, I can begin to outline the application. This is where I would find the relevant pieces of code from the original game, and use it as a template to fit my code into. Once this is done, I can add in the source images and 3-D models for the added content, and attempt to run the game from there. Once everything is complete, I will post the modification for free public use.

The ideal user flow for this program is seamless. I want the styles of my added content to match the original style of the game so as not to break immersion. I have played a few modifications that completely ruin the point of the original game, and are almost a separate game entirely. My goal is to seamlessly integrate the added content so that it isn't easy to distinguish from the original game.

I believe that all of these goals and ideas are highly achievable. There are countless resources that I have access to, and I am confident that this can be done. I am so excited to start working on this project, and even more excited to see the final product. This project will help me gain so many new skills, expand my portfolio, and begin my journey into the world of game development.