Saber Modpack Manager

MtgSaber

# System Synopsis

This system will facilitate version control, documentation, distribution, and client installation of my modpacks. It will consist of a client JavaFX program, one REST API server for the primary services of the system, a second API for account services and authentication, and a small website for account services and the distribution of the client program for this system.

# Component Synopses

## Client Program

The client program will be built on JavaFX and will attempt to update itself by querying the API at startup. It will allow the end user to log in to their account and will maintain a refresh token for it. Once logged in, the program will allow the user to view and install the various modpacks that they have access to. For a given modpack installation, the user will be able to adjust client mods and RAM settings, and the program will automatically retrieve active, public servers for the pack. The user will also be able to view and install different versions of the modpack. By default, the latest version will be selected, the installation will automatically update, and the user will be notified of major version updates. The UX and application logic requirements will be documented in a separate document.

The client program will contain a LuaMAP installation, which will be used for admin-assisted debugging and executing the installation process by requesting an installation agent from the primary API. This allows for complex and dynamic installation processes, which will involve many HTTP requests to the Curseforge site and local directory manipulations.

## API servers

The primary API will be a RESTful API server running on JVM, and it will facilitate access and manipulation of modpack mod lists, modpack version control, modpack access permissions, public server registration, modpack documentation, mod documentation, and mod author accreditation. The endpoints will be documented in a separate document.

The primary server will also contain a LuaMAP installation, which will be used for remote debugging and for API administration and maintenance.

The account services and authentication API will be another RESTful API server running on JVM, and it will facilitate login authentication, token validation, account management, and account services administration. The endpoints will be documented in a separate document.

Notably, the account services and authentication API will *not* feature a LuaMAP installation in the interest of the security of authentication and users’ accounts.

## Website

The landing page of the website will feature several modpacks and their documentation and will allow for users to log into and manage their accounts. Once logged in, the main page will have an option to download the client program. Mod lists, mod documentation, mod author accreditation, and mod Curseforge links will be available on the site.