

Anthony V. Perozich IV

I chose the “statusbars” plugin to look at more indepth. This plugin shows the status effects that are currently inflicted on a player to show what kind of damage they are taken. I chose this because I know status bars are very important in gaming to help players visualize exactly what effects are on their character and how they should make their next decisions. The information from the status bar can also help with what kind of armor to wear and what kind of status effect preventers that should be used. I also wanted to see how much code exactly goes into something that seems as simple as showing a player what status effects are inflicted on them at the current moment.

Overall, I think the code styling looks very clean and is understandable for the most part. One suggestion I would have would be variable names. I noticed that there are some variables that are named with just letters and that does not give me any information on what that variable is supposed to represent. One specific example in the code is the “renderHelaingBar” method. There are two integer variables ‘x’ and ‘y’ that I am not exactly sure what they are for or what they represent right off the bat. These variables should probably be renamed so that it is much easier to visualize what these numbers represent without the need to crawl through the entirety of the code. While the integers are probably not significant, show that those variables are not important by naming it “unsigNum” or something like that. Another suggestion I would give for the code styling is to make sure that each method created is used. One method in the plugin “provideConfig” is never used and there are no comments for what this should be used for. I would suggest either taking this method out of commenting to describe what this method could be used for in the future.

To update this program, I think this plugin would be pretty simple since it is just small things that need to be fixed. The only thing that would be a worry is making sure that variable name changes are accounted for all plugins and methods the variable is mentioned in. The code speaks for itself and documentation is not necessarily needed but documentation is always a help for people who do not want to search through code just to find where a method is referenced. Documentation could help show where different methods and variable are referenced so they are easier to find, but overall the code does a good job showing how the plugin works and what goes into it. The code is very well organized and broken up into different sections. Each part of the code accomplishes one thing and an update would not hinder another part of the code to the point where it has to be changed completely for it to function again.

I am not going to lie, I had a really difficult time trying to think of an idea for a plugin I would want to see in runelite that is not already there. I struggled so much that I decided to go to reddit to figure out what people who actually play the game would want to see and I was not disappointed. I thought this was a good idea since the players are what truly dictates if a game is good or not and listening to the audience is a good way to get different ideas. One idea that I thought would be interesting would be having a display that shows the actual damage dealt to an enemy. This seems extremely useful just because this would show the exact amount of damage to help the player visualize how strong an opponent is and how much more training they need to do before they are able to defeat the foe. I do think that this plugin would be able to use another plugin to help structure and make it functional. More specifically the “dpscounter” plugin would be helpful since it is already doing to math of the damage per second. This would be helpful since the code for the equations are already done and can be used in another plugin. This would just leave programming the plugin to display the numbers in the correct location.