

# Week 4 Reflection

Last week, I began getting interested in object oriented programming in ROBLOX, after I learned about module scripts and how they're used. After watching a few tutorials, I imported the open-source module that contains the skeleton code for the "class" object, which has methods such as:

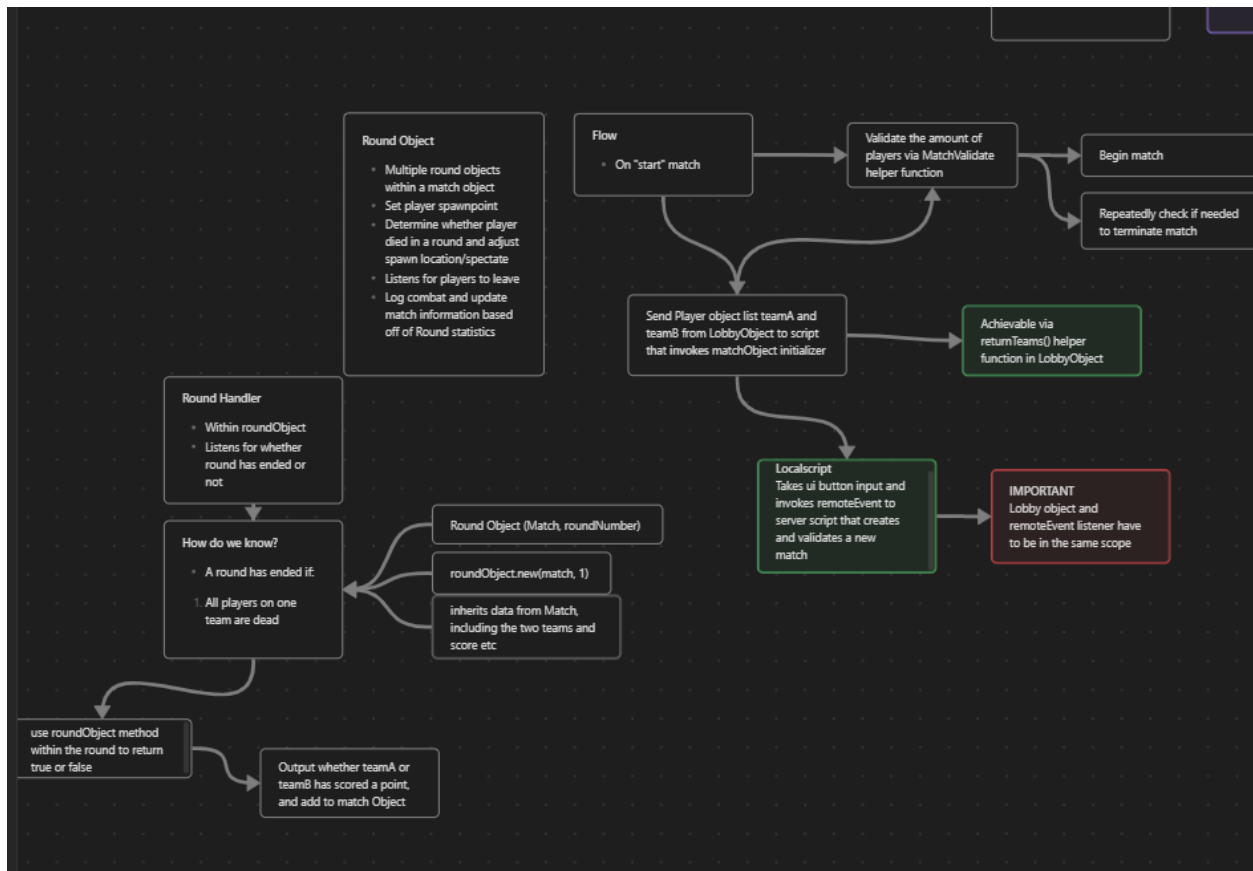
- `":new()"`, which is self-explanatory. As with any language, you can create the newly imported objects with multiple arguments.

However, something that I still have questions about is how garbage collection works in ROBLOX. If a module or class isn't used anymore, yet it still has been created, is that module going to be collected and deleted?

Irregardless, I am excited to continue learning about OOP in ROBLOX and finally being able to organize my code in an understandable manner that makes me less upset haha

## Object-Oriented-Programming Mini-Project

As part of this object-oriented theme for this week, I decided to plan out the queue and match system for a game. It will work as described below:



The text may be small on the document that's being read, so I'll explain my thoughts below:

Objects:

- Match
- Round

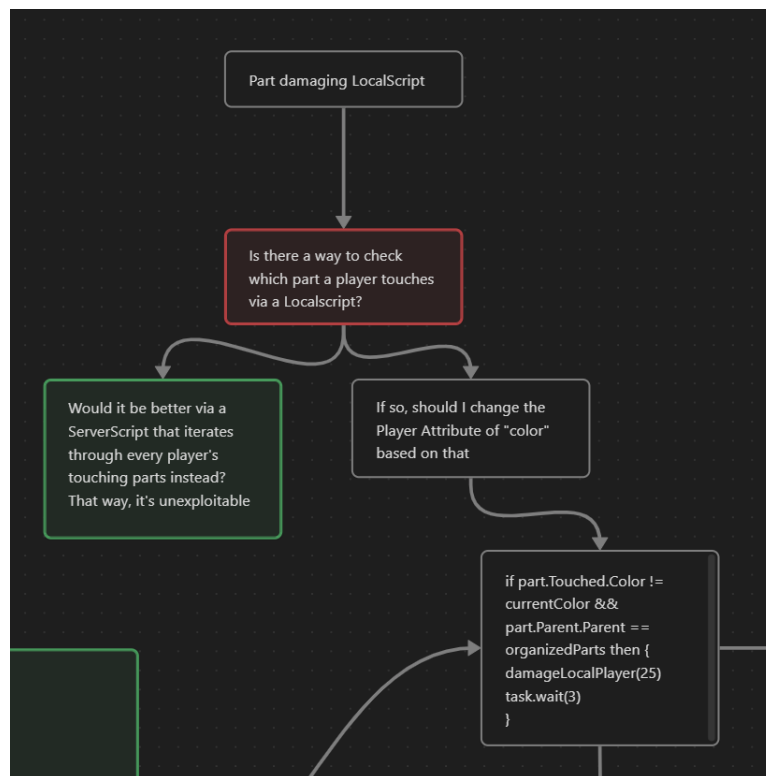
There are multiple rounds within a match, and each round, there's a winning team. Team A or B. Whoever wins the most rounds within five rounds wins the entire match.

I also need to design a way for the player to click the "start match" button once a match is able to be started (if there are equal players on each team). So, I'll need to design a way to check that as well.

One more thing that I'd like to touch on is the importance of having logically sound code & ideas before implementing or writing it out. Most of my time was spent brainstorming

how this could work and how to organize it. Since it's in an object-oriented style, I think that this has improved my general code organization as well.

**Most importantly**, however, is my discovery of the desktop app Obsidian. It allows me to make easy flowcharts and documents, and make a mind map of what I think should work versus what is actually feasible. For the final project that I've been working on, I've been able to hypothesize and then evaluate my thoughts, prior to writing code.



As someone who has many thoughts and ideas racing constantly when coding, writing and organizing my programs has made it a lot easier to get the results I want.

This tool will carry on much further than ROBLOX as well. I'm going to use this for the rest of my programming related projects because it's just that helpful.