

Open Source Software: Proposal and Specifications

Description of Project

For this project, I have decided to go with Option 2: I will implement a computational solution to a problem I perceive in daily life. It is no secret that millions of people play video games every single day. Naturally, these gamers wish to test their skill against other gamers, and so they engage in video game tournaments held all over the world. However, many of these tournaments are poorly organized, making them difficult and stressful to manage. My goal is to create software that will allow people to quickly and easily manage tournaments.

My program will focus on two tournament types, single elimination and double elimination, since these are the most popular. In single elimination tournaments, the players all start out at the leaves of a binary tree. As they win matches, the players advance up towards the root of the tree, until only one player remains and is declared winner. Double elimination tournaments use two trees: one for the winner's bracket, one for the loser's bracket. Because of this, I will have to implement algorithms to create, modify, and traverse binary trees in my program.

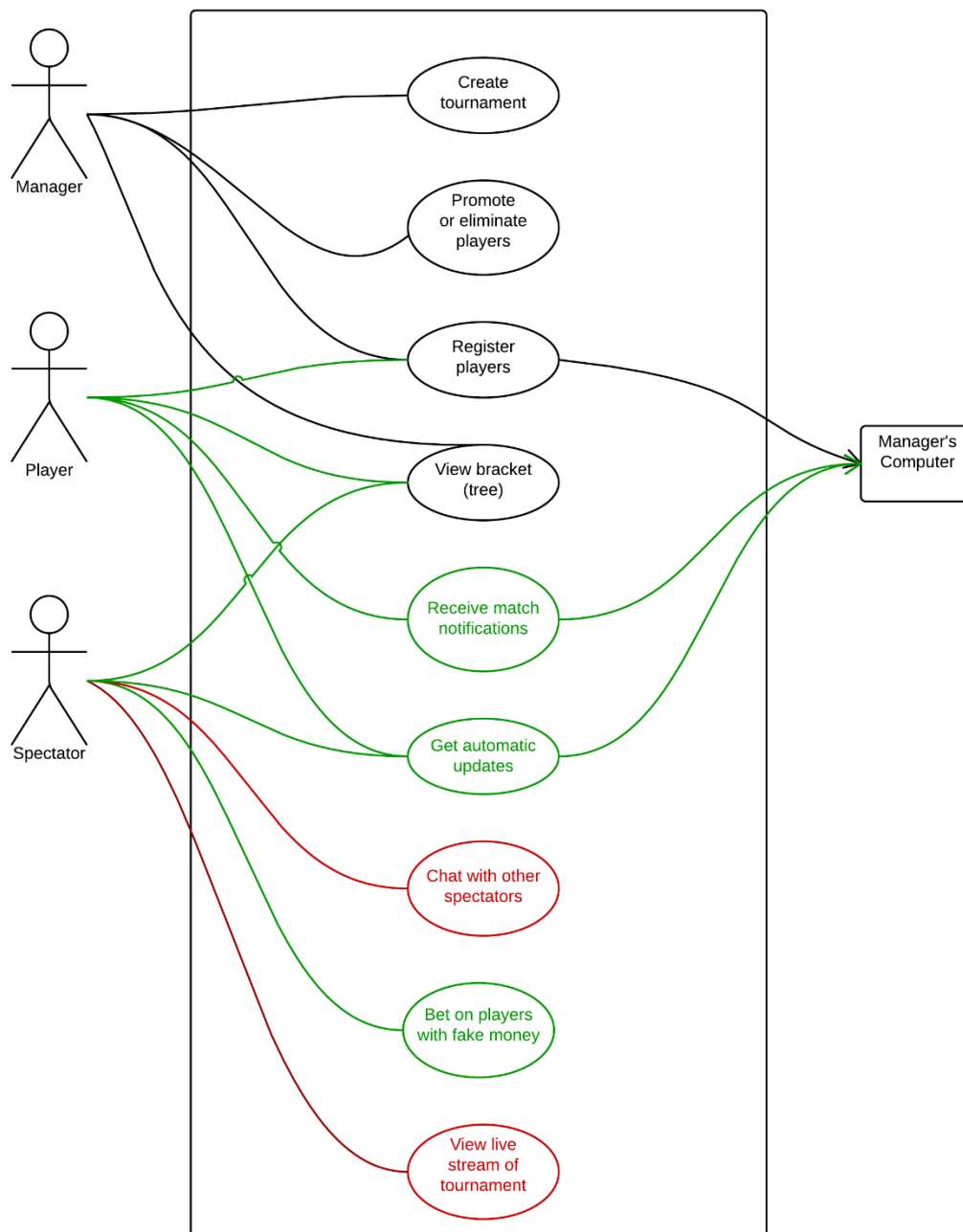
My choice of language is C++. Now, you might be thinking that this project would be easier to implement as a WebApp because it would be much easier to create the user interface. But I am not looking for an easy project; I have been wanting to learn how to program GUIs outside of web browsers for quite a while and I think this is a good opportunity. I would like to use an open source cross-platform GUI library (such as GTK+ or wxWidgets) to create a GUI for the user to easily visualize the data. One benefit of creating the software this way is that it will not depend on a web server to be up and running in order to use it.

Currently, there are a few web-based tournament managers which provide very basic features: user registration, visualizing the bracket as a tree, and player promotion/elimination. In order to make my software different from these, I will add some other features not seen in any other tournament management system. I would like to implement network connectivity between computers running this software. Players can register themselves from their own computer by connecting to the tournament manager. Players and spectators can connect to the tournament manager's computer to stay automatically updated. Players can get notifications of when it is their turn to play and where to go to play their match. Spectators will be able to bet (with fake currency) on who they think is going to win, and they will win or lose their fake money accordingly. There could also be a chat where people can talk about things like who they think is going to win. And there could be a live video stream integrated into the program for people spectating from home. However, these features require knowledge of networking, which I may not have time to learn. If I have enough time to learn basic network programming, I

may be able to implement some of these features. But for now I must defer the completion of these network-related features until a later date.

Use Case Diagram

- Black = This feature will be implemented by project completion date
- Green = This feature will not be implemented by completion date unless I have enough time to learn about basic networking
- Red = This feature almost definitely will not be implemented by the project completion date



Open Source Licenses

- **Apache** — The Apache license requires you to include the license and copyright information with the code, as well as state changes you made to the code. It allows both commercial and private use, distribution and modification of the code, and patent rights. The author of the software cannot be held liable for damage caused by the software.
- **GPL (GNU Public License)** — The GPL is similar to the Apache license, with one main difference: it is required to make the source code available when distributing the software.
- **MIT** — This is a simple license that basically lets anyone do anything with your code as long as they credit you and understand that no warranty is provided.

For my project, I would like to use the GPL. I do not care how other people will modify and use my project, and I do not plan on making any money from it. But I do want to encourage the creation and use of open source software, and allow other people to create derivative works if they have good ideas to add on.

Github repository

I will hold off on creating the private repository until the Github issues have been resolved.