CSC309 Individual Final Project Proposal

Tian Yue Liu

1004743753

skyjoy.liu@mail.utoronto.ca

Project Name: VeryGraphical JS

My proposed JavaScript library aims to provide a tool that can easily generate various visual maps using given data. Some of the examples include: knock out brackets, single/double elimination brackets, round robins, and group stage information for varying team sizes in tournaments and matches, various templates for family trees, and as a bonus, search trees with nodes to represent data structures etc. In the library, it will include both functions to generate skeletons for these structures as well as functions to alternate their appearances, including size, color, and position in the viewport. For a front-end user, he/she will have the ability to input his/her desired data and expect their respective fully rendered, straightforward, and illustrative graphical representation. As a developer, there sometimes exist the need to showcase the input data in an intriguing, comprehendible way. With this library, it saves the need to manually plan out and physically code the layout of the graphs, which in some cases are very troublesome. This allows developers to be more time-efficient and productive in their projects, creating apps with apprehensible visual representations that also are uniformed in style. The idea is novel and unique; I have needed this feature while making personal projects and could not find this resource on famous libraries such as material UI. Yet, such need is urgent as visuals can contribute immensely to the attractiveness of an application while also playing an imperative role in assuring that the ideas conveyed by the developer can be properly grasped by the users.

There is a myriad of web apps that can benefit from such a library. For instance, I participated in a non-competitive table tennis team and the league uses IMLeague to host their tournament information. Last time I used that site, it was a massive data dump and extremely difficult to navigate. Using the team brackets feature in my library, not only does the developer save time on entering all relevant statuses, but the users also benefit from having tournament information more comprehendible. Furthermore, the Wikipedia page on the queen Elizabeth II could use the family tree feature. A user interested in seeing members of the royal family current has to navigate from pages to pages if only using Wikipedia. At last, storyline, tasks involved gaming web apps, as well as career planning sites can all use the node tree feature where each node is a necessary step. User on the sites can follow the task paths to beat the game, or to plan out an agenda towards their career goals. For instance, the UofT computer science courseography site can use a tree with nodes representing courses. This way, the developers can present all the offered courses easily without spending time on creating layouts, at the same time as offering a way through which students can easily view specific requirements.