High School Programming Contest Specifications

Daniel J. Bell | Kyle C. Fairfax | Joseph E. Webster

Summary

The High School Programming Contest (HSPC) Web Application should handle all aspects involved with registration, event management and administration. Our primary audience for this project is advisors and students from Kansas high schools, computer science faculty and staff, and volunteer judges. Our goal is to provide a streamlined, easy, and intuitive way for all users to act within their respective roles. Advisors should be able to register teams of students for events. Teams/Students will have access to questions of varying difficulty at said events. Judges will have the ability to see questions and teams. Administrators should be able to create events, add questions to said event with varying difficulty, and alter team information. In short, Administrators will have all access to every portal and behavior necessary within the application. Master users are having a special administrative role and have all previously mentioned access with the addition of granting administrative access to specific users.

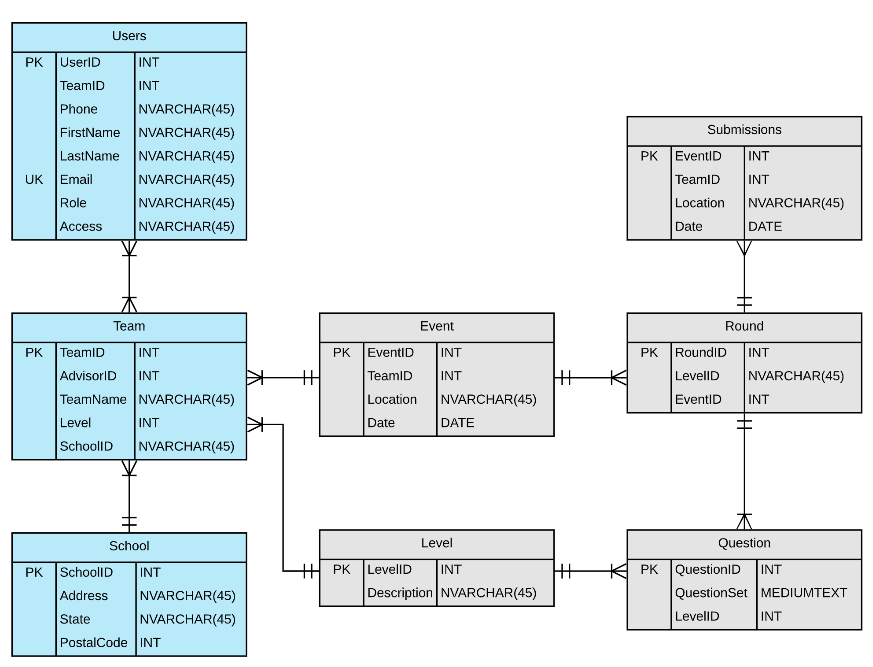
Design

The HSPC Web Application was designed based on contemporary web development practices and modern libraries. For our front-end, we chose to use the modern and fairly new libraries of ReactJS coupled with Bootstrap. This framework is referred to as React-Bootstrap and can be found [here](https://react-bootstrap.github.io/). ReactJS makes development quick and intuitive, with an easier learning curve compared to Angular or Express. This has allowed us to quickly update our front-end and create a project quickly. We have been able to create our front-end with the goal of creating a single-page application. In our scenario, we common traits and tools shared across multiple components to implement this, and we render our components based on events. **Figure 1** depictsa simple diagram of how our front-end design:

C:\Users\josep\Downloads\front-end-design.pngHere we show the varying level of access within the HSPC front-end. Our main component responsible for containing and displaying all our user portals / components. The level of access is shown by the diagram. As the amount of area is decreased, the less the access. For instance, an Advisor would only have limited access to the controls that a Master / Administrators would have access to. However, some roles will share access to some varying degree. Such is the case with Judges, Advisors, and Masters / Administrators.

Design (cont.)

Our API or back-end includes a variety of tools. It utilizes the HTTP client, [Axios](https://www.npmjs.com/package/axios) to handle API calls and HTTP requests. We have also implemented an MSSQL database within a container created by [Docker](https://opensource.com/resources/what-docker). Docker is a tool that was designed to allow developers to things like MSSQL (.NET based), on open-source operating systems. It does this using “containers” which can house any number of programs. Our database was meticulously designed with the help of John Keller. Our main intention with our database design was to create something that could be easily extendable, but vast enough as-is to house anything our clients may need to store in the future.

**Figure 2** is a depiction of our database schema: 

The schema above was developed by the cumulative efforts of John Keller, Dr. Josh Weese, and us. Our database can handle all, current necessary information required from all our clients. This may change in the future, but the core schema has been designed in such a way to allow for extension. This way if more data is needing to be stored, developers may add a table or relationship in an easy and effective manner.

Hosting & Environment

We are currently trying to get our project hosted on a local, computer science owned server. We have spoken with Mr. Seth Galitzer through email but have been unsuccessful in having our work hosted. We will need to host this in a 64-bit Linux environment as we have not tested it on a 32-bit OS. In particular, we’ve developed on the 64-bit Linux Distribution, [Ubuntu v.18.04](http://releases.ubuntu.com/18.04/). Our current environment with Ubuntu includes Node v.11.1.0, React v.16.6.1, React-Boostrap v.0.32.4, and Axios v.0.18.0.

Supported Browsers

The current iteration of the HSPC Web Application can run on Firefox & Google Chrome. We have not yet tested other browser environments, as we are focusing on core functionality at this time.

Milestones

A few milestones we have reached with this project include but are not limited to a fully functional authentication system, basic portal / component design, a fully functioning database, a fully functioning implementation of Axios. More specific milestones include a working registration system for users, the ability to create custom teams’ w/members, and a working implementation of a captcha. Milestones in the future will include a fully functional Event creation process and a timer for Events.