**GRST Project Proposal**

**Problem Diagnosis:**

There are lots of websites that offer programming challenges. These sites vary in quality, usability, and focus. Currently, the only way of looking at challenges from a variety of sites is to visit those sites one at a time. To do this users have to search for a new site, learn to navigate its unique interface, and find challenges of the appropriate difficulty level. If the first site does not contain content that is a good fit, then users must repeat the process from the beginning. This is a process most users will quickly find tedious.

**Proposed Treatment:**

Our team proposes to build a project that will collect challenges from all of these disparate sources into one easy to use location. Users will no longer have to find challenge sites on their own. They can also view challenges from other sites without having to learn a new interface. Finally, by pulling from so many different sources, we will offer users the broadest possible selection of challenges.

Our primary users will be people wishing to practice programming skills. Some examples would be job-hunters practicing whiteboard problems, students studying for a test, or even people simply wanting to practice programming for their own benefit. It could also be used by giving programming problems to others. Employers need problems to give their prospective employees and programming clubs to their members.

The project will use a web scraper that will collect publicly available information and store it in a database. In addition to the text of the problem itself, we will collect other useful information including the original host site, the URL of individual challenge, the challenge’s name or number where applicable, and difficulty where applicable. New challenges can be collected either automatically at regular intervals or at an administrator's request.

End users will be able to use an interface to view collected challenges made up of three primary parts. First is a search menu. Showing every problem from all of the included sites would overwhelm most users. They will be able to use the menu to search through the database to limit the quantity of results to a manageable amount. They may only be interested in problems originating from one or more sites, or only looking for easy difficulty problems. Our search feature will allow them to do that or choose from a select list of other traits that are important to them. Next is the result list. This just displays all of the challenges that meet the requirements of the search. The user will be able to select a given challenge and view its text and other challenge-specific information. This comprises the third and final part of the user interface.

There are several other features that our program would ideally have. First and foremost is the ability to create and upload their own challenges. We will have to look at the pros and cons of adding them to the central database available to all, gated behind some kind of a log-in system or keeping them locally on the user's computer. Users should also be able to favorite challenges they like. They should be able to change the status of a challenge; statuses might include solved, viewed but not solved, not viewed, skipped, and newly downloaded. Favorite challenges and statuses will also be searchable.

We may want to associate each challenge with a folder and display the text of any files in that folder below the text of the challenge. This would allow users to easily see how they implemented solutions in the past. It could be useful both for people studying, and for employers seeing if their prospective hires are close to the right idea.

**Plan of Work:**

Our team of four is the ideal size for this project. One person will focus on interacting with the external challenge sites to collect the data. Two people will focus on the user interface and search functionality. The final person will focus on the database and ensuring overall cohesion between the other parts of the project.

The above proposal represents the ideal project. Implementing all of the features is certain to be too ambitious for a one-semester class to be completed in its entirety. However, we are following Dr. Marsic’s advice and starting ambitiously, understanding that we may have to scale down. Fortunately, our project is easily scalable. Collecting challenges from as little as a single site should be both doable and serve as sufficient proof-of-concept. The user interface needs a way of showing both a list of available challenges and the text of any specific challenge. Everything else can be scaled down or even removed completely. When the deadline for our project arrives, it will likely be somewhere between the two extremes presented. However, we will add as many features as time allows.