Names: Michael Lam (mqtlam@cs)
        James Vaughan (vaughanj@cs)

Subject: (Assignment #1) Proposal for Funding the Homework Aggregator Project

PURPOSE

The project we wish to propose can be simply described as a Homework Aggregator.

VISION

Every quarter at the University of Washington, professors usually publish course websites to post information about homework, syllabus, calendar, etc. For some students, digging for information from every website for every class can be time consuming and inconvenient. For students who like to organize, gathering information from every website and compiling them into one centralized location (e.g. Google Calendar) can also be time consuming if done manually.

The Homework Aggregator solves this problem by automatically aggregating useful and relevant information from every course website into one convenient page. Essentially this page would be a one-stop location that has information about all the current courses, the currently assigned homework from every class, a comprehensive calendar, etc. A student may only need to visit this page for homework assignments, course announcements, etc. Additionally, the Homework Aggregator would provide services to organize, personalize and even export the aggregation.

Currently, *MyUW* provides convenient aggregation of links for university services like registration, transcripts, finances, and much more. However, while it knows what classes a student is taking, it does not know how to extract useful links and information from each course to provide additional information on the *MyUW* home page, such as currently assigned homework and class announcements.

The target customers are University of Washington students. This project is worthwhile to develop since such a service can make the lives of students a little more convenient.

SOFTWARE ARCHITECTURE

In its simplest form, the Homework Aggregator is a website that provides aggregation services. However, such service may be extended to widgets like *MyUW* or Facebook, or even to a mobile application in the future. In order to keep the scope of this vision reasonable, we focus on developing the product as a web application.

The service works by crawling course websites and finding links. Using some data mining, links to specific content may be recognized and learned. Optionally, if professors choose to provide metadata on the website, recognizing useful links is much easier.

There are five modules that will need to be developed.

1. **Networking**. This is a given, since the program needs to connect to CSE websites to find the appropriate data required to display what it needs to. There are already existing frameworks for web navigation that can easily be integrated with the program.
2. **Data gathering**. This module will extract the data from the web page for the next module.
3. **Data interpretation**. This module will transform the data into a more usable form.
4. **Data storage**. The program would likely not grab the data every single time the user wants to view it as this could cause bandwidth issues, so we need a server of some kind that can store the results for later.
5. **Interface**. The user needs a clean UI in order to read the data gathered of their classes.

CHALLENGES AND RISKS

The single most challenging aspect of this product will be to develop a data gathering and interpreting system that can actually sort through what it manages to gather and give information relevant to the user. This will probably be the most time consuming part of the project, and may become the most infuriating for our users if we require too much input from them in order to successfully gather data to present to the user. After all, the Aggregator is meant to collect the data for them, not act as simply a collective display page put together by the user.

To mitigate this risk, the team that develops this product will probably want to set aside a specific stage where most, if not all, of the team members will want to be researching and developing this aspect together. This will allow the free flow of ideas to go through and to create a temporary environment more similar to ad-hoc programming so that a prototype of this service can be made. Once a working version is put out, the team can restructure it in a secondary stage of development so that it is a significantly more polished aspect of the product. This may interrupt the overall flow of the development cycle and require some specification changes but in the end will allow the product to be shipped much faster.

Another large challenge is the issue with the way our intended users will be utilizing this service. Provided that this project is aimed at being a web app, we will have to find some way to integrate the service into common websites that students already use (e.g., Google Calendar or *MyUW*) without intruding on the way the users utilize these preexisting services so that the users receive the most amount of exposure to the app without going too much out of their way to do so.

One potential solution to such a problem is to contact the content manager of *MyUW* and to work with them in developing the app as a custom tool for use inside *MyUW* so that users can use the preexisting *MyUW* interface in order to place the app where the users want it.