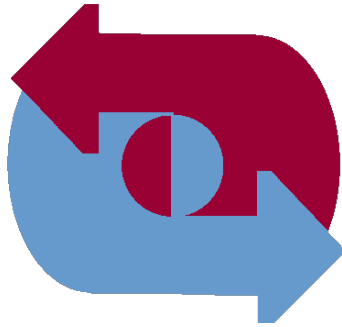


CourseSwap



Derek Tseng, Adam Kalman, Meg Li

Vision

Students at the University of Washington are strongly encouraged to take full schedules every quarter they attend. However, course space is limited so students often can't register for all the classes they want. The University is a big place, and we find it hard to believe that all these students in classes they don't want are stuck there. We want CourseSwap to allow these students to find other students that are in the same position, and hopefully have everyone get into the classes they want.

When students are registered for courses they don't want, they can come to CourseSwap to list the classes they have, and the classes they want. By looking through these matches, we can find them other students who they can switch courses with. Both get out of a course they don't want, and into the one they do.

While CourseSwap at its most basic level is just a networking site based around registration, it has the potential to grow into a full system. In addition to having students perform class switching en masse, and an intrinsic scheduling system to arrange them, CourseSwap could eventually incorporate all kinds of data around schedules. Such as class reviews, and even give foresight into the popularity of a course before registration opens.

Proposed Minimum Viable Product

Students using this website must register for an account using their UW email. A confirmation email is sent to the UW account that they register with in order to ensure that they are really a UW student. After they have registered and logged in, students may add or remove items from two separate lists: a list of classes that they are registered for and a list of classes that they want. Students can make a search on the site that finds other students that have a class that they want and want a class that they have. The website notifies students via email whether a match has been found.

Proposed Technical Details

We decided that CourseSwap could be best utilized as a mobile web application. The interface is simple enough that it would be easy to design a minimalistic UI, and accessibility via mobile phone can greatly improve response time if a swap is discovered and initiated. The front end will be written using Twitter Bootstrap, as it is easy to learn, and effective for our purposes. This will be backed by a Ruby on Rails service stack, as we feel that Ruby is a flexible language that will promote the kind of services we want to build. The database will be MySQL as it is free, simple and we have experience

with it.

Risks

CourseSwap has two main risks that we foresee. The first is that the interaction is driven by the students themselves. CourseSwap gets better and more efficient (in terms of swaps successfully completed) with a higher userbase. If only one person is in the system, there is nobody to swap with. So for the system to be as successful as possible, it would have to be advertised.

The other risk is that since CourseSwap isn't a part of the University of Washington registration process directly, we can't offer any guarantees about the mechanics of the swap. So the potential for abuse is very high. While we can't necessarily prevent this abuse, by forcing users to use a UW email that has been verified, we can offer effective and swift account bans when abuse is reported.

There is one other risk that could arise if CourseSwap does become popular. If the practice of using CourseSwap to get into the courses you want becomes widespread, then it creates a meta-economy around the courses based on their popularity and availability. So it is possible we would see a large spike in students registering for classes they have already taken in hopes that it would be more popular when trying to find a swap.

Stretch Goals

- Automatically update user profiles when swaps are completed
- Students can provide additional contact information (cell number, etc)
- Allow students to set up a "swap appointment"
- Simple messaging service for users before they swap contact info
- Password recovery
- 3+ man swapping
- Search caching
- Check if user enters in faulty classes / Editing field for existing class

Basic Application Flow

