Tomasz Parzadka, Hans Allendorfer, Weiyi Zhou, Puja Mittal, Ashwin Prasad

## Problem Statement:

A university student's academic success is dependent on their ability to synthesize and process information learned in lecture and lab. Tutors, professor office hours, study groups, and other academic resources can greatly assist a student in achieving this goal. However, students are often forced to comb through numerous websites and scattered posters in order to determine the location and times of these resources. Even then, it's not always easy to identify how much help they can be with the students' specific problems. Therefore, we aim to centralize this information on a single platform with an emphasis on convenience.

## Background Information:

**Audience:**
Our target audience are students at a university that are seeking resources outside of class, and tutors that want to offer their services. We are going to be targeting this service at students at Purdue to give us a reasonable scope.

**Similar Platforms:**
Most tutors advertise by putting up flyers, which is time consuming and has limited reach. Other services like UniversityTutor and Wyzant exist, but they do not incorporate their platform-specific tutors with resources already available through the university. Our platform aims to create a single location where students can access all helpful resources available to them, whether they be university provided or posted via our platform.

**Limitations:**
The key limitation of other solutions is that they do not provide a centralized location from which all information and resources for a course may be accessed. For example, the course website may not have a list of tutor names, or supplemental instruction (SI) session dates and times, or even links to related supplemental material. The tutor list might not have course information or SI times, the SI session times/locations might not have tutor information and so on. Thus, oftentimes students can be discouraged from utilizing these resources because of the tedium of visiting multiple sites to figure out the necessary information. Our goal is to collect the information from all of these resources and display it in a single web page for each course, bringing students a convenient and accessible option to view all course related information.

## Functional Requirements:

**As a user, I would like to:**

1. Register for an account
2. Log into my account
3. Edit my profile (e.g. add a picture, contact info, personal info, etc.)
4. Reset my password

5. See a list of available tutors
6. Sort the tutor list by price, class, rating, etc.
7. See the profile for the tutor that posted a listing
8. Leave reviews for tutors including ratings, pricing, and other comments
9. Create a posting looking for a tutor with specified subjects and location
10. Opt to be a tutor and have a public profile with reviews and information
11. View all current requests for a tutor
12. Filter tutor postings by subject, time, price, rating
13. Add/remove courses I am taking
14. Have course information auto-filled from Purdue databases
15. View course related information (e.g. professor names)
16. View all preexisting tutors, office hours, or SI sessions for each course, scraped from Purdue-based sources
17. View tutor listings for each course
18. View profiles of professors
19. View professor reviews and ratings from RateMyProfessor on their profile
20. Comment on courses/professors, so future students enrolled in that class/with that professor can see a review
21. View previous course exams and lecture notes if publicly available
22. Allow students to see required materials for the course.

**As an admin, I would like to:**

1. Update course pages manually
2. Edit stored information about tutors and other resources
3. Send reset password emails in case of forgotten passwords

## Non-Functional Requirements:

1. The application will run on the web browser
2. Must be able to store all information on a SQL-variant database
3. Must have a user friendly interface
4. Must only allow Purdue affiliates to use the app in case of sensitive/copyrighted information
5. Must only allow users to access information pertaining to themselves (i.e. by sanitizing inputs)
6. Must store all sensitive data safely (i.e. salting+hashing)
7. Must have fast response times; the connection between the UI, server, and database must be fast.
8. Must be portable (will render similarly for all common windows sizes)