

# UIUC Course Hub

*Get all the info on classes you are interested in, including statistics and testimonials*

## 1. Summary

Come  $\frac{2}{3}$  of the way through each semester, and we are constantly searching through various sources to find courses and specific professors to take courses with. We are relatively lucky to have the [grade disparity](#) website from the CS department, Rate My Professor, the [HKN Wiki](#), and the [MCS website](#) for specific CS courses. However, the issue is that a lot of the great functionality of the MCS website is not expanded to all courses across campus and lacks some of the beneficial information from websites like linking professors to courses, seeing the average course GPA, and helpful general course info (what really happens in the course, not what the course description says it is). With a lot of the data publicly available or crowdsourced, we believe building a new app similar to the MCS app - essentially, a place for students to rate and find courses that best suit them - with extra functionality and for all of the University courses would be especially helpful to all students.

## 2. Detailed Description

Our summary describes the project reasonably well, but to reiterate, our project is a web application that mimics the [MCS course website](#) from scratch for all university courses, not just MCS ones, and with some extra functionality. We are considering recent professors to teach a course and their ratings, summarizing course content from all the user reviews with GPT and GPA data for each course from recent semesters. The summarization would be an interesting (and a bit of a challenge) feature to use GPT for and especially helpful when the official description of a course is very poor. The GPT feature is our creative challenge, and we are planning to achieve this by assigning 1-2 team members to dive deep into this task later in the development process.

Our **Data Sources** and **UI Mockup** sections explain the data sources and show what users might interact with, but it is essential to talk about the “why” and the problem. We often find ourselves (and others) searching Reddit for a lot of this information, and while Reddit does an ok job, using a platform like the MCS course website is 20x better because we save much time and find exactly what we are looking for. We often look for information related to what the course is actually about since many of the descriptions on Course Explorer are pretty poor, whether the class is useful, whether the professor is good, and whether the average GPA is not abysmally low. Our data sources will allow us to aggregate a lot of the info students care about and solve








the problem: finding useful class information takes cross-referencing multiple platforms and a lot more time than it should if the information is aggregated in one place. With much of this publicly available or crowdsourced data, we believe we can solve this problem university-wide.

Overall, the application aims to provide a centralized platform for students seeking comprehensive information about various courses. It aims to address students' challenges in making informed decisions about course selection by offering detailed insights into factors such as difficulty level, time commitment, average GPA, professor nuances, and covered topics. The primary problem is the lack of a consolidated resource where students can find and contribute valuable course information.

### 3. Usefulness

An application like this would be extremely useful because we find many existing websites handy, especially the MCS website. However, the MCS website lacks most courses and some functionality that these other sites offer, which is why we think making an application for all courses with the publicly available GPA statistics and saving professor ratings (similar to how the MCS site saves course ratings) would increase the quality of information students have access to when deciding which courses to sign up for. To reiterate, our application will be similar to the MCS website, but we will differentiate our app by adding professor ratings and GPA info and making all UIUC courses available on it.

On our web application, users can expect to see overall course statistics (see **Figure 1** below; average GPA would be included in the final view) in a table view, grouped likely by department. Clicking on a course will go to that course's page, where users will be able to see reviews of the course from other users, the professors who have recently taught the course and their average rating from users, and (stretch goal) a GPT generated description of what students can expect to learn and do in the course based on all of the reviews.

Course Number	Course Name ↑	Reviews	Difficulty (1-5)	Workload (hrs/wk)	Rating (1-5)	Semesters
CS 598	Advanced Bayesian Modeling	6	4.3	19.5	4.5	
CS 525	Advanced Distributed Systems	1	5	16	5	
CS 441	Applied Machine Learning	32	2.3	8.3	2.7	 
CS 498	Cloud Computing Applications	27	3.7	13.4	3.9	
CS 598	Cloud Computing Capstone	2	3	10	4	
CS 435	Cloud Networking	10	4.5	17	4.6	

**Figure 1. Example table view with course information**

## 4. Data Sources / Realness

This project will be using a variety of Data Sources. The primary data source will be Course Explorer, and we plan to use the Course Explorer API to pull in course names, departments, descriptions, professors, credit hours, semesters offered, etc. This data can be saved as a .csv to be loaded into our database later. While we are still determining the total number of courses offered at this time, the College of LAS claims they have [over 1400 courses alone](#). We should have a few thousand courses from across the University.

Additionally, we will use the [WadeFagan GPA Datasets](#) to pull in course GPA information and provide that to users. This dataset contains data on most of the courses at the University and historical information from 2010-present, stored in .csv format. We will generate user data to fill in course reviews and other user input stored in the database.

If necessary, we can expand to other Universities and list their courses, but we need to investigate the best way to get similar data from other Universities. This project is similar to the real world and could continue after this class.

## 5. Functionality Description

The website is designed to be a place where students can go to figure out a variety of information about courses that they may want to take - difficulty level, time commitment, average GPA, professor differences, topics covered, etc. Our overall idea is that a user can come to the site and search for a course they are interested in. Once they find the course, they can view reviews and aggregated information, potentially sorting it by semester offered, professor, etc.

Additionally, users can contribute by logging in with their NetIDs and submitting reviews containing the above information if they wish to contribute to the site. These reviews (combined with information from the university) will aggregate the data into a summary that anyone can see at the top of the course page.

### 5.1 Course Exploration:

- Users can search for specific courses of interest
- Detailed information is accessible, covering difficulty level, time commitment, average GPA, Professor nuances and topics covered
- Facilitates informed decision-making based on individual preferences and academic goals

### 5.2 Sorting and Filtering:

- Users can refine search results by sorting and filtering
- Parameters will include semester offered, professor, and other relevant criteria

- We believe this will help enhance the user experience by allowing optimization of search results

### 5.3 Contribution Mechanism:

- Secure login with NetIds for users wishing to contribute
- Users can submit reviews containing course details and insights
- User-generated content supplements official university data, enriching the platform's information pool

### 5.4 Data Aggregation and Summary Generation:

- Systematically integrates user-generated reviews with official university data
- Dynamically generating summaries visible at the top of each course page.
- This summary provides a quick and consolidated overview of key course attributes.

### 5.5 Individual User Profiles

- Users can create personalized profiles tied to their NetIds
- Track and manage their contributed reviews and engagement history

### 5.6 CRUD Operations for Users:

- Users can perform Create, Read, Update, and Delete operations for their submitted reviews.
- Allows users to manage and edit their contributions, providing control over their shared experiences.

## 6. UI Mockup

**I**

Home

Add Review

Settings

Help & Getting Started

Log Out

Course Investigator

Course Number(CRN)

Course Title

Professor Name

SUBMIT

**I**

Home

Add Review

Settings

Help & Getting Started

Log Out

Course Investigator

Course Number(CRN)

Course Title

Professor Name

Rating

Avg GPA


Difficulty


Workload


SUBMIT


**I**


**Course Investigator**

 Home

 Add Review

 Settings

 Help & Getting Started

 Log Out

Course Number(CRN)

Course Title

Professor Name

**SUBMIT**

CRN	Course Title	Professor Name	Rating	Avg GPA	Difficulty	Workload
CS 411	Database Systems	Abdu da legend	5.0	3.75	2	moderate

## 7. Project Work Distribution

Our project's roles are organized into critical categories: Database Design and Implementation, API Integration, GPA Dataset Integration, User Data Generation, Backend Logic, Frontend Development, Testing, and Documentation.

For Database Design and Implementation, Team Member 1(Shreya) is responsible for defining tables and establishing relationships based on our discussed requirements. Concurrently, API Integration is handled by Team Member 2(Avery), who is focused on seamlessly incorporating the Course Explorer API into our system. Simultaneously, Team Member 3 (Shreyas) is tasked with the GPA Dataset Integration and is responsible for extracting GPA information and linking it to our database.

In the development phase for User Data Generation, Team Member 4 (Alec) will generate dummy user data to simulate course reviews, playing a pivotal role in populating the database. Backend logic is a collaborative effort between Team Members 1 and 2, encompassing functionalities such as calculating average ratings, managing user reviews, and generating GPT-based course summaries.

Frontend Development is divided into UI Design and Mockup Implementation, managed collaboratively by Team Members 3 and 4. Integration with the backend is a joint effort between Team Members 1 and 2. In the testing phase, Team Member 4 takes the lead in thoroughly evaluating the application, ensuring all functionalities work as expected, with collaboration from others for integration testing.

Lastly, the documentation phase is a collective responsibility, with everyone contributing to creating and maintaining project documentation, including database schema documentation, API documentation, and any other relevant information.