1. **What are the names and NetIDs of all your team members?**
   a. Patrick Gruber – pgruber2
   b. Jesse Henn – jjhenn2 (Team Captain)

2. **What is your group name?**

PiazzaEd

3. **What system have you chosen? Which subtopic(s) under the system?**
4. **Briefly describe the datasets, algorithms or techniques you plan to use**

We have chosen to improve the usability and reach of the existing EducationalWeb system by creating an integration with Piazza. To build the dataset, we will create a scraper that downloads all the class relevant Piazza posts. The system already comes with a wide variety of toolkits. Metapy is currently used to rank relevant slides, based on the one that the student is currently viewing. To be consistent we will try to re-use the already existing ranker and build upon it. If time permits, we might also try to use Neural Networks to find relevant Piazza posts.

5. **If you are adding a function, how will you demonstrate that it works as expected? If you are improving a function, how will you show your implementation actually works better?**

The user interface will be enhanced to show piazza posts along the slides that are currently returned. Some Piazza posts contain questions about very specific slides of a lecture. To demonstrate the ranker works as expected, we will spot check some of those Piazza posts with what the ranker returns for a given slide.

6. **How will your code communicate with or utilize the system? It is also fine to build your own systems, just please state your plan clearly**
7. **Which programming language do you plan to use?**

EducationalWeb has an already existing Flask Python codebase that serves files, requested by the student. To have the frontend communicate with the backend, we will add another route to said app and serve recommendations based on queried search terms and viewed slides. Search terms for slides are already sent to the backend via JavaScript. To make this consistent, our addition will do the same. In conclusion, we will use both Python and JavaScript to complete this project.

8. Please justify that the workload of your topic is at least 20*N hours, N being the total number of students in your team. You may list the main tasks to be completed, and the estimated time cost for each task.

Main tasks to be completed:

- Get current system running
- Update system to use Docker and needed libraries
- Create mechanism to capture Piazza posts
- Integrate indexing and retrieval of Piazza posts
- Prototype user interface to display Piazza results, collect and analyze feedback
- Update user interface

9. What is the function of the tool?

EducationalWeb is a one-stop-shop for students. While it displays lecture slides at its core, it also integrates with external services that students often use. For this project, we plan on integrating Piazza into the tool. While students are browsing different lecture slides, a recommender tool will automatically suggest Piazza posts that might be of interest to the student, based on the current content of the slide. Additionally, the recommender system will also suggest Piazza posts that are related to the currently entered search term.

10. Who will benefit from such a tool?

As distance learning becomes more popular, it is especially important for students to stay in contact with teachers and each other. While they are going through lecture slides, they might have questions that were already discussed on Piazza. A well implemented recommender system would display those discussions, without the student having to navigate to a different service and search for related content manually.

11. What existing resources can you use?

A github project exists for accessing Piazza's internal api (https://github.com/hfaran/piazza-api). As piazza does not document there api we will need to investigate how well the project will work for our purposes. The last commit to this project is now over a year ago.

Hans Bas detailed his experience training a neural network in this post: https://www.hansbas.com/blog/personal/review/2019/11/09/i-trained-a-nn-intro-cs.html

12. Related material

- Web of Slides: Automatic Linking of Lecture Slides to Facilitate Navigation - https://bhaavya.github.io/files/wos_wip.pdf
- WOSView Demo: A Tool to Explore the Web of Slides - https://bhaavya.github.io/files/wos_demo.pdf
- Current Source Code - https://github.com/CS410Fall2020/EducationalWeb