

VIP-ITS-Chatbot Fall 2022 Project Proposal

Group Membership:

Member		
Rish Desai	Skills	Java, Python, HTML, CSS
	Credits	2
	Responsibility	Working with frequency and pattern matching algorithm to determine which questions should there be for the time range.
Nirjhar Deb	Skills	Java, Python, HTML, SQL, CSS, JavaScript
	Credits	2
	Responsibility	Build an algorithm to rank questions by relevance when displaying them at end of autocompletion (e.g., prioritize questions with the most words in the index and/or textbook practice questions → helpful for students).
Yueqiao Chen (Christina)	Skills	Java, Python, C
	Credits	2
	Responsibility	Improve the chatBot on the recentness and frequency of the piazza questions and add some priorities choices for users and contribute to the algorithm part.
Nicholas Jaimes	Skills	Java, Python, HTML, CSS, JavaScript
	Credits	1
	Responsibility	Research the optimal algorithm to find similarities between data. Further develop a system to parse through the questions through Piazza within a given timeframe. Contribute to the algorithm that groups different questions by topic/subject.
Max Everest	Skills	Java, Python, Javascript, SQL, HTML, CSS
	Credits	2
	Responsibility	Further develop SQL storage, perhaps by adding timestamps and other useful columns that could help others experiment with the autocomplete rankings.

Project Timeline:

Week 1-3: Team formation, proposing project ideas, planning the project.

Week 4-5: Research how the ChatBot works, make a plan for project implementation.

Week 6-7: Research appropriate algorithms to be used within the project.

Week 8: Gather Piazza and textbook questions to be fed into the algorithm.

Week 9-10: Create the algorithm that will group questions by subject and track current progress in the semester.

Week 11-12: Create the algorithm that will track the frequency of certain questions.

Week 13: Make final contributions to the program so it will gather the three most-likely asked questions.

Week 14: Implement the project into the ChatBot.

Week 15: Testing, adjust and optimize accordingly.

Week 16-17: Final Presentation.

Project Goals:

We are focused on improving the chatbot user experience by developing a more robust algorithm for suggesting questions and supplementing it with a larger variety of data. We are generally interested in ways to contribute to the question suggestion side rather than the answering side. We envision ideas such as the chatbot generating sample questions before the user knows what to ask, perhaps to spark some curiosity and guide students who may want to utilize the chatbot more deeply

Project Description:

Nirjhar's responsibility:

To improve the autocompletion experience, I plan to build an algorithm to rank the questions displayed to the user after autocompleting their input. For example, questions will be ranked based on factors such as how many words in the question are present in the index and whether or not the question is related to a practice problem. Just to clarify, I **will not use machine learning algorithms** (due to a lack of data and expertise) but **will use a custom weighting system** for ranking. I will research similar ranking systems online (e.g., Clash of Clans war matchmaking, which uses factors such as win-loss ratio and weights of defense buildings to find evenly-matched opponents) and use that to inspire my factors.

Since I do not have much experience with building ranking systems, I will likely struggle a lot to determine which factors will provide the most relevant questions that will help the students succeed in class. So, I will need to consult my team and my professor throughout the semester to make sure that I am building a fair ranking system. I will also consult my friend Sam, who is a senior software engineer at Google who understands big data and analytics really well.

Christina's responsibility:

The autocomplete function currently didn't have suggested question when no typing presents. I think it would be helpful if we add perspective questions in an ranking order for user to look up when they didn't type things. Therefore, this semester, I will take an effort to improve the chatBot on its functions related to the relevance of questions presented to the user. Prioritizing could be useful for searching relevant questions. How frequently are the questions being asked? Is the question recently related? What is the question about? These questions are often being considered for a question. To make the user much more convenient to found related questions, I can add frequency and recentness to the chatBot to help searching relevant questions. I foresee that because we didn't have enough data to work on, and we don't have a recentness related to the dataset, so we should probably work on that.

Max's responsibility:

I want to continue working on the autocomplete from Spring 2022, and I am planning on contributing by making a more usable and organized database. The data from last semester is rather unorganized. All the data is in one table, and data that are fundamentally different, such as machine generated questions and user asked questions, are sharing unnecessary columns. Although I foresee that for testing purposes, a local database is enough, I also want to look for an easy way to store the data on the cloud so that when the time comes so that everybody that might need it could access a unified source of data on all the questions asked since we reach full integration once we get there. Since the data and its dependent algorithms are closely tied, I am also interested in making general contributions to the autocomplete module where needed. I am not very comfortable with the answering module or other modules, so I might not be able to make general contributions in those areas. If time permits, I would also like to make a quick graphical view of the autocomplete working in real time so that everybody could quickly test it out live before full integration.

Rish's responsibility:

I will be working on developing the pattern matching algorithm that will be looking to see the questions and how similar they are with one another. Some of the potential challenges associated with this are not just designing an algorithm that can find exact matches but trying to identify which questions are very closely similar. These strings will be coming from the Piazza question database. Based on the most frequently asked questions, students will be guided or recommended certain questions to do. This may involve looking into multiple algorithms to identify which is the best one to use for our project purpose.

Nicholas's responsibility:

I will be working on going through the ChatBot and learning its intricacies so that I will be able to contribute to its improvement. Then, I will research the best way to go through questions on Piazza and the textbook so that it can be used in later parts of the project. Furthermore, I intend on contributing to the algorithm that will go through the Piazza questions and find the most common questions so that we will be able to find the questions that users would most likely ask.

Some potential challenges I expect are going to be adjusting to working on this project, as I am the only person in the group that has not previously worked on the project. I will need to go through the ChatBot to find out how it works. Furthermore, I will have to do a large amount of research on different algorithms so that I can contribute to the project, as I do not have much experience working with these kinds of projects.

Implementation and Collaboration

This project is mainly developed in Python and we plan to continue using it. Most likely, we will fork the Spring 2022 Chatbot Autocomplete repository to work on the question side; however, we may use other repositories if needed. We are planning for each member to create their own branches and commit their individual contributions to their respective branches. We will communicate on Microsoft Teams and will organize a weekly meeting on Friday 3:30 PM to ask questions, solve problems, and discuss contributions. These meetings should last around 30 minutes.