Final Report

● Ungraded

Group

Elliu Huang
Kevin Lu
Shih Ting Huang
...and 2 more
View or edit group

Total Points

- / 15 pts

Question 1

Report 15 pts

Question assigned to the following page: 1

StudySphere is a crowdsourced study guide platform where students contribute, answer, and refine practice questions to support collaborative learning and exam preparation.

Kevin Lu, Kristine Huang, Elliu Huang, Amish Sethi, Kerry Zhao

Introduction

StudySphere tackles the challenge of finding high-quality, targeted practice materials for exam preparation and learning. Many existing resources lack customization and refinement, which can make them less effective for addressing specific learning needs. By enabling collaborative question creation and iterative improvement, the platform fills this gap, helping students access reliable, diverse, and constantly updated study materials tailored to their educational goals.

StudySphere has similarities to platforms like Quizlet and Ed discussion but stands out due to its collaborative focus. Quizlet allows users to create their own set of flashcards and study the material using different games. You can also share sets of flashcards or make them public. On Ed Discussion, users can ask clarifying questions about topics related to their current courses, and the TAs or professor can answer them. StudySphere combines aspects of these platforms while introducing a unique layer of community-driven refinement to ensure high-quality and dynamic study guides.

Our project is a collaborative educational tool that leverages crowdsourcing to develop a repository of high-quality, subject-specific practice questions. It emphasizes user engagement and cooperation to create a constantly improving and evolving resource for

Question assigned to the following page: 1

learning and exam preparation. The project blends elements of social platforms, educational technology, and collaborative content development.

The main focus of our team's effort was a combination of engineering a complex system and conducting an in depth analysis of data in order to best support a platform that encourages users to learn specific topics from each other.

Participants will contribute by submitting new practice questions, reviewing existing ones, and providing answers and feedback. Basic subject knowledge is essential for contributors to create relevant and accurate questions or to evaluate and refine others' contributions effectively. We want to add the feature in the future where advanced contributors, professors, or TAs can also facilitate content for quality control.

We implemented basic aggregation and quality control modules. Quality control will rely on a multi-layered approach, including a voting system for users to upvote valuable content and flag inaccuracies. A peer-review mechanism will allow experienced and active users to validate contributions. Low quality submissions can be downvoted and if it reaches less than -3 votes, it will be hidden from the users.

We chose aggregation and quality control for in-depth analysis.

Final Presentation video

The Crowd

The members of our crowd are university students. Our data was from a real crowd: students within NETS 2130 and more generally, students of the University of Pennsylvania. In order to recruit participants, we asked students within the class as well as peers outside of the class. We had a total of 90 unique participants.

Incentives

The main motivating factors that the crowd had for participating in our project were learning more about different topics about various classes and getting questions answered from a wide range of people. These motivating factors are out of enjoyment rather than monetary incentives.

In order to incentivize the crowds to participate, we promoted the following as a means to help users expand their knowledge. One of the incentives stems from general

Question assigned to the following page: 1

intellectual curiosity, where different individuals asking different questions could spark interests for specific topics. Because of the wide variety of classes, students curious about a certain topic but have not had the opportunities to learn it in depth could discover different ideas and conversations in courses they are both taking and not taking. The second incentive is getting their questions answered by their peers, so they can deepen their understanding of a topic they are learning. More specifically, if a user is confused by a certain topic but is not able to ask the question until class time, or if the user does not know any peers to ask the question to, our project provides a space for users to have these questions answered.

We did not provide any specific analysis comparing different incentives, as the enjoyment incentives were the main feasible solutions. However, we did consider a couple other incentives in addition to general enjoyment such as a rewards system, where users could earn participation points towards a class if they answered and asked a set amount of questions. We found this to not be feasible at this stage, as this would require specific integration of our project in classes.

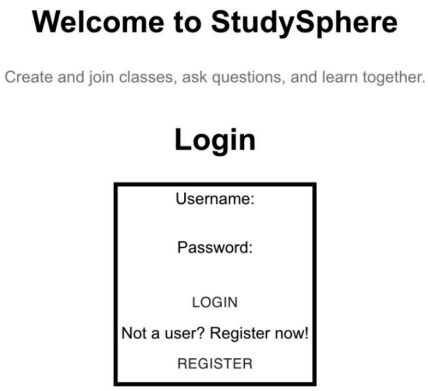
Crowd Contributions

The crowd provides a variety of data: the name of a course, questions within courses, answers to specific questions, difficulty of a certain question, and quality of a question. While general questions regarding course content and generic questions about a certain topic can be generated using LLMs like GPT, Claude, or Gemini, the purpose of StudySphere is to have questions and discussions specific to a midterm or final exam. This could be as specific as one of the steps in the proof of an algorithm or a technique for wedge products that is only taught by the professor at Penn. The other crowd contributions, difficult and quality of questions, shouldn't be automated because this is specific feedback from the students.

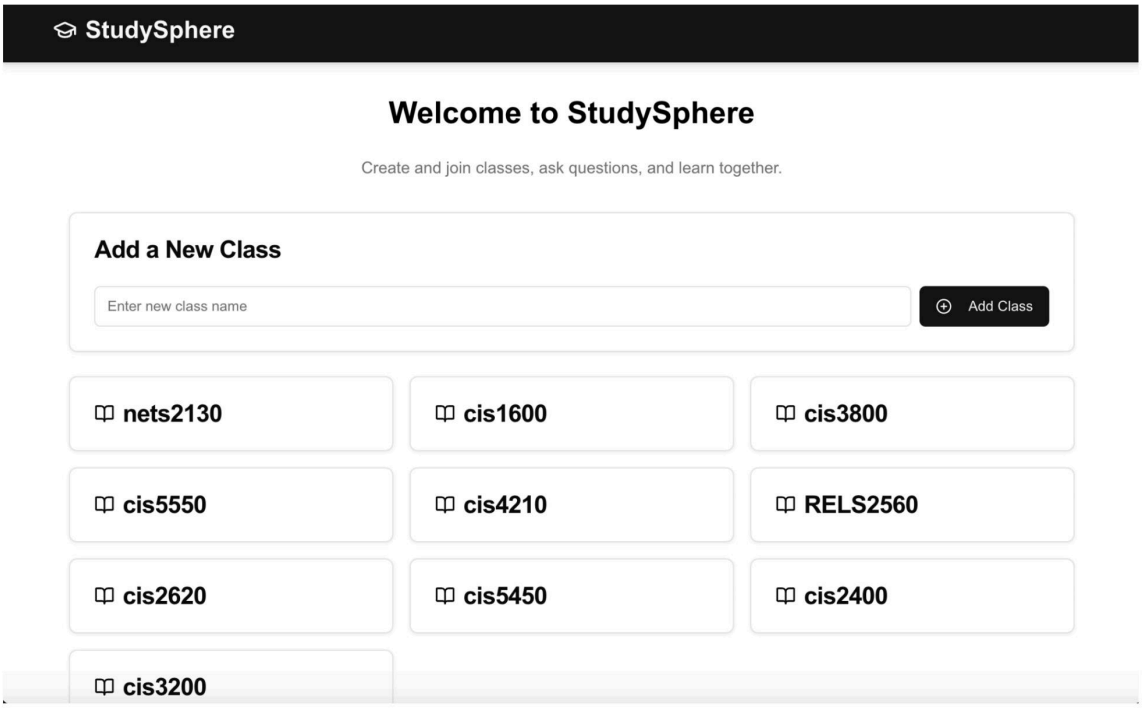
Our user interface is a Next.js platform, where users can input their class, specific questions and answers, and their opinions on questions through upvoting and downvoting.

The following shows the register and login pages, where users can either create an account with a University of Pennsylvania email account and existing users can login with their credentials.

Question assigned to the following page: 1

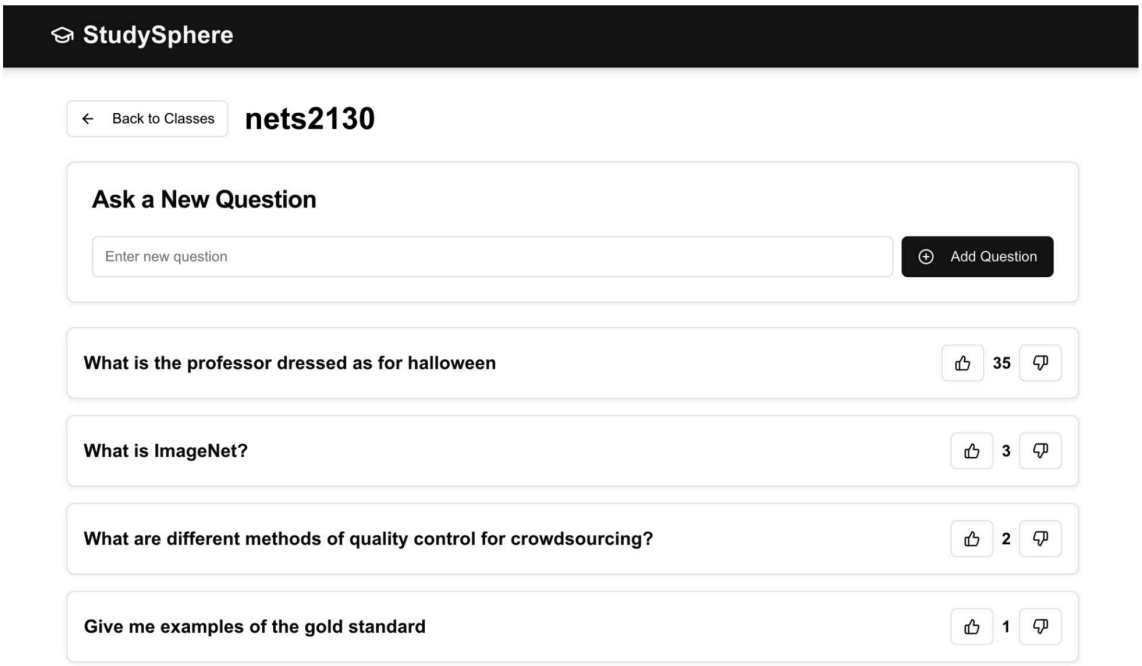


This is our home page, where users can create a new class or go to a specific class page.

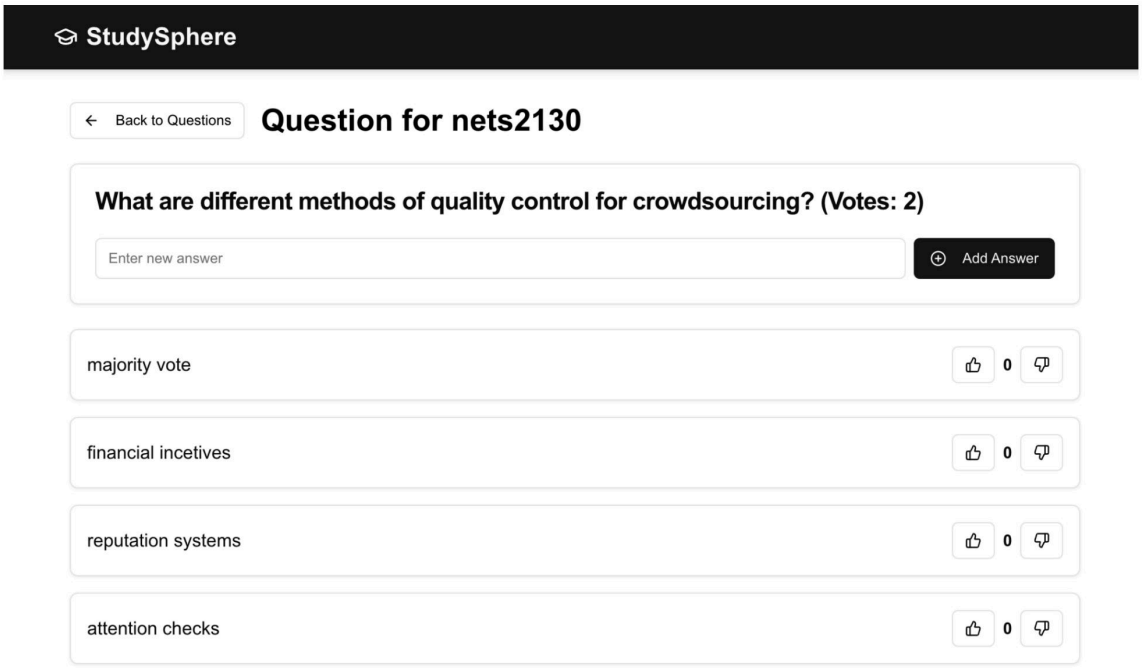


This page is for a class, where users can ask new questions, upvote/downvote existing questions, or click into a question that has been asked in order to answer or read existing answers.

Question assigned to the following page: 1



This page is for a specific question, where users can add new answers, rank the difficulty of the question, or upvote/downvote existing answers.



Question assigned to the following page: 1

Ethics

We asked a couple of questions in consideration of ethics.

Q: Should my application exist at all? What effect can it have on them?

A: Yes. The effects are because this helps users build on one another's various domains of knowledge. Users do not have to contribute a set amount of time in order to use the platform, allowing users to utilize the platform to best fit their needs.

Q: Does this task potentially expose workers to harm (for example, content moderation)? What effect can it have on them?

A: There is a lower chance of exposing workers to harmful content, because the content is that of the content covered in courses. However, if there were to be harmful content, it would get removed, as the questions that meet a certain threshold of downvotes get removed. A possible effect if workers would be exposed to harmful content would be being uncomfortable, but our platform aims to reduce the possibility.

Q: Are you fairly compensating the workers for their time?

A: Since there is no set task we are providing the workers with and their contributions are based on their own intellectual curiosity, we are not providing any monetary incentives.

Q: Is your evaluation sound? Do the conclusions you reach stand up to scientific scrutiny?

A: Yes, our evaluation is sound. The conclusions we found stand up to scientific scrutiny, where our conclusions we found stem from a wide range of data and users.

Skills

Our crowd workers do not need specialized skills. However, if they were answering questions on behalf of a class, they should have the domain knowledge in order to ensure the accuracy of their answer. The skills depend more on the class itself (ie. a worker contributing an answer to the CIS 1200 course would need to have knowledge of OCaml and Java topics). The skills of individual workers can vary widely. A factor that can cause one person to be better than another is the thoroughness and clarity of their question or answer. Skills of the crowd were not explicitly analyzed, but the use of

Question assigned to the following page: 1

downvoting prevents any user from submitting an incorrect or irrelevant answer. We had a couple graphs illustrating the number of questions and answers compared to the class, the frequency of the number of votes for questions and answers, the difficulty of the questions compared to number votes and answers, and the rating compared to upvotes (refer to Aggregation).

Quality Control

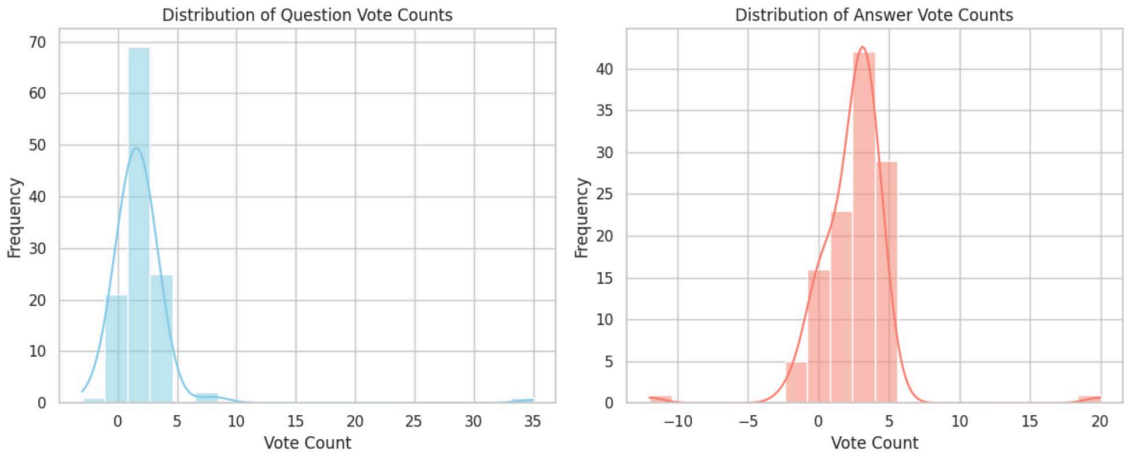
Quality is the single most important factor for our platform. The quality of the questions and answers to the questions is what makes students use the site, so it's important to have a robust quality control system.

StudySphere implements a majority vote mechanism to ensure the prioritization of high-value contributions. Questions and answers submitted by users are ranked based on community votes, with the most highly upvoted responses prominently displayed. This approach helps maintain relevance and encourages the creation of high-quality content. Users collectively determine which submissions are most effective, fostering a democratic and inclusive system for refining the study material.

To manage low-quality contributions, the platform utilizes database triggers to automatically eliminate questions or answers that receive a score of less than -3 votes and have more than 60% of their votes as downvotes. Additionally, users who have three or more of their submissions removed due to poor quality are flagged, with their profiles stored in a separate database table for potential review. This system ensures accountability and encourages users to contribute thoughtfully while maintaining the integrity of the content within the platform.

We analyzed the quality of the questions and answers by aggregating it into charts and graphs.

Question assigned to the following page: 1



In the chart above, we have 2 distributions of the vote counts and answer counts. This is where users upvoted or downvoted questions and answers depending on the quality. Bad or off-topic questions typically have lower votes or even negative votes. Better questions have higher votes. This pattern holds for the quality of the questions. We can see that the median votes per question was around 2-3 votes. This is due to the fact that there weren't many users on the platform in the first place, so not many people saw the questions and rated them accordingly. For the answers, the distribution is more spread out with a median of around 4-5 votes per answer. Many users were probably interested in reading the answers and voting accordingly.

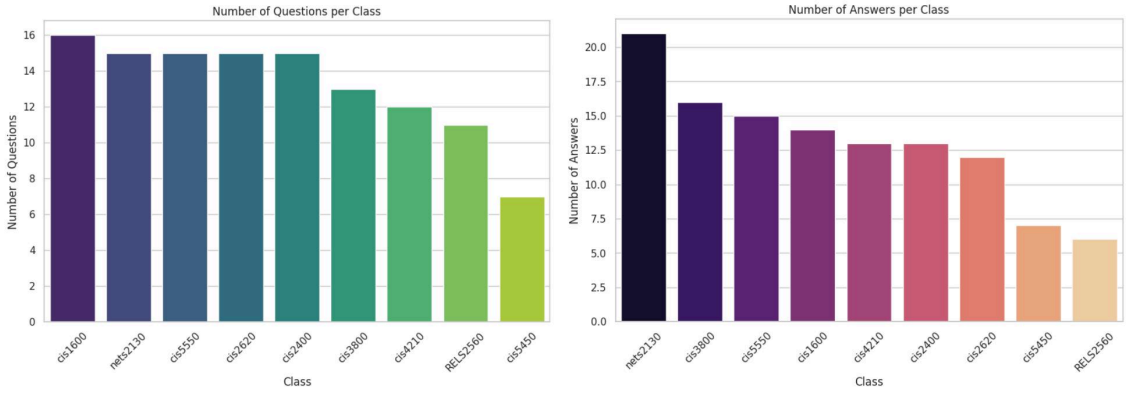
Aggregation

StudySphere has gathered a robust dataset of 90 data points from a diverse group of contributors. Approximately 40 participants are students enrolled in NETS2130, while the remaining 50 are students from other Penn CIS courses. Together, they have aggregated study questions for nine different classes, creating a valuable resource tailored to a range of academic subjects. The platform also allows users to sort questions by the number of upvotes or their difficulty level, making it easier for students to find the most relevant and appropriately challenging material.

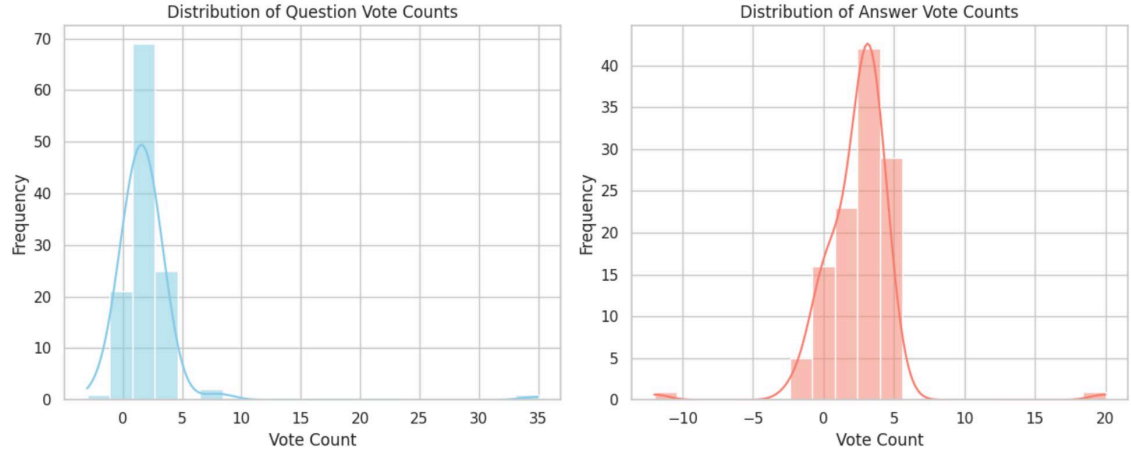
The dataset includes 119 total questions, 117 answers, and 122 difficulty ratings. On average, each question has received 1.91 votes, while answers have garnered a slightly higher average of 2.42 votes. Difficulty ratings, which provide insight into how challenging users perceive the questions to be, average at 3.66. These metrics indicate active engagement from the community and highlight the platform's effectiveness in

Question assigned to the following page: 1

fostering collaboration and content evaluation.

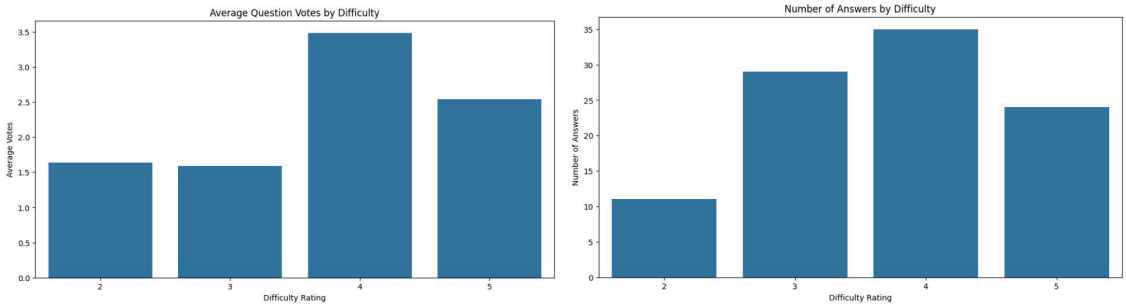


In this graph, we noticed that there was a skew for the number of questions per class towards the CIS/NETS classes. There was also a large amount of answers submitted for NETS 2130 in comparison to the other classes, which could be due to the fact that a significant amount of our data was collected from NETS 2130.

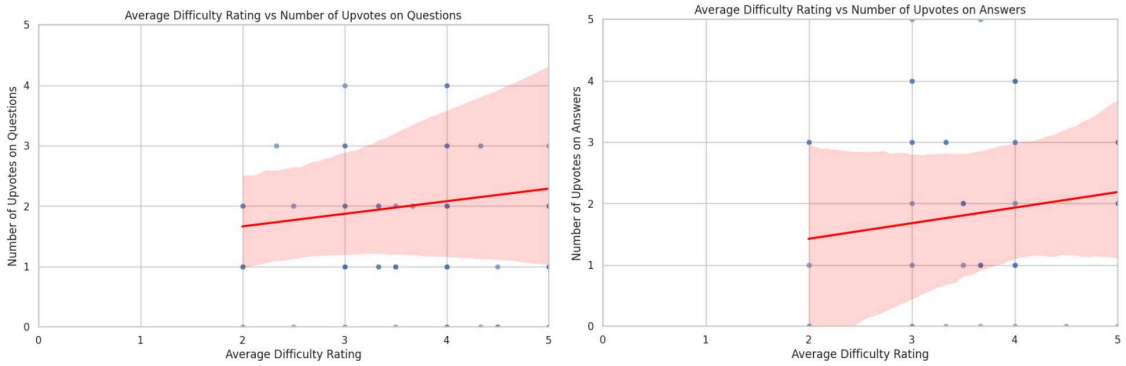


We also concluded that the number of votes for each question and each answer typically was less than 5 votes, where the majority of the questions had 1-2 votes and the majority of the answers had 4 votes.

Question assigned to the following page: 1



Also, users tended to vote more for the moderately difficult questions and answer more for the moderately difficult questions, where both peaked for a question's difficulty rating of 4.



Lastly, we noted that there was not a significant correlation between question difficulty and the number of upvotes on questions and answers, but was slight as the question difficulty increased, the number of upvotes increased.

We did not create a user interface to see the user to see the aggregated results, but they are able to see the number of upvotes and downvotes as well as the difficulty rating.

Scaling Up

The problem of accessing targeted and high-quality study materials affects thousands of students on Penn's campus. Existing platforms are limited in their ability to address specific classes at Penn. By enabling crowdsourced questions and answers for high-quality study guides, StudySphere seeks to tackle this large-scale issue by creating a versatile, up-to-date, and inclusive repository of practice materials.

StudySphere would only work with a large crowd. A broad user base would enable the

Question assigned to the following page: 1

platform to cover a wider range of subjects and question types while improving the quality of content. Additionally, a larger crowd would accelerate the feedback and refinement cycle, ensuring that questions are quickly improved or replaced. There would also be stronger community engagement, fostering a self-sustaining platform.

Scaling to a large crowd could introduce several challenges:

- **Content Moderation**: Ensuring the quality of contributions while managing potentially harmful or inappropriate submissions.
- **Platform Performance**: Maintaining system stability and responsiveness as user activity increases.
- **User Management**: Preventing spam or abuse and maintaining engagement with meaningful contributors.

A basic cost analysis of the project mostly comes from the computational and server costs with hosting such a large discussion platform. Otherwise, there is no cost for getting users to contribute or moderate the content.

Project Analysis

Yes, our project worked. This is because we collected a variety of questions and answers from a variety of classes (refer to Aggregation).

Some of the positive outcomes of our project on the user side was being able to ask course content specific questions and get potential questions answered, with the quality of the questions and answers held. On the development end, we were able to understand crowdsourcing as an application to having users input questions and answers as well as voting. We were also able to strengthen our analytical abilities by looking at the different data and thinking deeper about potential reasons for the results we got.

There were a couple changes between what we had originally proposed. Initially, we wanted to have a rewards system for contributions such as badges, but we decided that badges were not a significant enough incentive compared to that of only intellectual curiosity. We also intended the platform to span different institutions, but we found that if we focused on one school, we could cater the questions and answers to more course specific content.

There are not a lot of limitations for our product, except for being specific to one school. In order to overcome this, we could incorporate another page for institutional login and

Question assigned to the following page: 1

have courses still be specific to a school in order to maintain the course specific content goal. If our project was a scientific study, a possible source of error would be answers with many upvotes that are incorrect. For instance, if there was a question that was a known misconception, there could be many users that input the known wrong answers unknowingly, which could mislead individuals on this topic.

Our results aligned with what we expected because we received data from mostly those who we expected, with an incentive to help their peers. In terms of data, we expected that there would be a majority of questions asked and answers in classes reflecting that of the users' backgrounds, as the majority of the workers were from CIS/NETS backgrounds. We also expected that there would be more user interaction to more difficult questions, as those were the questions of interest to more users.

Technical Challenges

The technical stack of our project consisted of using FastAPI/Python for the backend and Next.js/JavaScript for the frontend. For our database, we used an open source Postgres database, Supabase. Thus, there was definitely a somewhat significant amount of software engineering that went into getting the entire application up and running, from building out the API routes in the backend, setting up the database, and building out all of the frontend pages and components. The portion of our team that built this out have experience building full stack applications and so there was an immense amount of new learning that needed to be done in regards to new languages or APIs.

Across the entire project, the largest technical component would probably just have to be the chunk of frontend comprising all the different class list pages, question list pages, and answer list pages. All of these pages were interconnected and had substantial tooling from upvoting individual questions and answers, rating question difficulty, and of course the actual submission and creation of classes, questions, and answers. Naturally, this was also the largest technical challenge we faced as a team as getting this portion of the frontend interconnected with the backend to pull and post data was somewhat tricky and getting everything to visually look and feel good from a UI/UX perspective was also a hurdle we had to overcome.

Our team has luckily had the experience to have worked across the stack (frontend and backend) and so a tackling this challenge was primarily done by having clarity on what the overall architecture of everything would look like from components on the frontend to clearly outlined routes that could be easily used in the backend. By doing this well, the integration of the two became a lot easier and, specifically on the frontend, building out

Question assigned to the following page: 1

more complex UI/UX experiences were easier to put together with organized components and logical flow.

Here are some screenshots of the application itself and a link to our [github repository]: