Computer Science and Engineering Department, University of Nevada, Reno

Lecture Goggles

Team #22, Logan Long, Nathan Yocum, Zachary Johnson
Dr. Sergiu Dascalu, Devrin Lee
Dr. Shamik Sengupta
November 2, 2018

Table of Contents

Introduction

The project proposed by Team 22 is a free, open-source, educational resource repository to help students gain a better understanding of school subjects. Currently, students use Chegg, Wolfram Alpha, Khan Academy, and Google searches to scour the internet for any information that helps them retrieve the answers to tough questions. Using these methods can take time, money, result in misinformation, or if unable to find reasonable explanations, a way to cheat. By providing a centralized location for students and professors to gain access to quality resources, we could help eliminate the cheating culture that has plagued institutes of higher learning and encourage better learning practices.

For our project Lecture Goggles, we will be creating a web application for students and teachers to collaborate to build a school subject resource hub. After the users register for a student or teacher account, they will have the option to provide they found throughout the internet. Breaking the website into a continuously expanding tree of topics will allow every subject covered in K-12 and higher education to be addressed. If there are missing subjects, then users can add them to the list of available subjects. Each main subject will be broken down into subtopics to cover all areas of learning.

The website should provide a clean, easy to navigate layout, and a well-organized structure. If any of these aspects are lost in project development, the users will find the site unusable and leave for other options. To keep the website well-organized and easy to navigate it will limit each site page to minimal selection. By breaking the complexity down, the website should guide each user to the topic they need.

Our intended audience for the first implementation of the web application is college students and professors. Using verified .edu emails, the webpage can limit direct access to a less anonymous user base. Since each student and professor have a verified college email, we can hold each user accountable. This will keep inappropriate links and malicious attacks to a minimum. As the website grows, we will be able to implement more access. Since the end goal is all students and teachers, implementing K-12 access would come later.

Summary of Stakeholder Interviews

Interview Questions
1. Do you currently use or know of Chegg homework assistance?   2. What resources do you use when studying new material?   3. When using these resources, do you have trouble finding answers that fully answer your questions?   4. What type of learner are you (visual learner, auditory learner, reading/writing learner, hands-on kinesthetic learner, or a combination)?   5. Do you have trouble finding quality resources to school subjects (math, science, biology, etc.)?   6. Have you ever found resources you wished to share with others?   7. How do you currently share helpful website links with colleagues and students?   8. On average, how much money do you spend on learning material (tutors, books, Chegg, etc.)?   9. Do you normally need more than one resource to learn a difficult subject topic?

Fig. 1: List of interview questions that are being asked of each interviewee. Each question relates to Senior Project's Lecture Goggles group project.

Dr. Shamik Sengupta's Responses
1. He has never used or any knowledge of Chegg. Since he is a professor there is no need for these services.
2. The resources he uses are IEEExplore, ACM, Elsevier, Wikipedia, and different forums he can find online.
3. There are some occasions where these resources do not fully answer his questions.
4. He is a visual and reading/writing learner. Reading books and forums are his go-to method of choice.
5. Sometimes, yes. Particularly when he is depending on google searches, there might be problems as Google provides a vast amount of information, so it's easy to get lost.
6. He has found sources in the past that would be helpful to share with colleagues and students.
7. Typically the easiest way for Dr. Shamik is using emails to share the links.
8. Because he is a professor, he can get free resources through University.
9. He wishes there was a better way to share resources he has found other than email.

Fig. 2: Answers to the list of interview Fig 1. questions. Each question is summarized answers from each of Dr. Shamik Sengupta's answers.

Dr. Shamik Sengupta is Lecture Goggles' project advisor. He is a networks professor who also specializes in education. Our project will require modern network security to provide a safety to our user base. In Fig. 2 provides answers to the list of interview Fig 1 questions. Each question is summarized answers from each of Dr. Shamik Sengupta's answers.

Sophia Harrigan's Responses
1. Sophia currently has an account with Chegg. She pays monthly for this service.
2. The resources Sophia typically uses are Chegg, textbooks from her classes, using google searches and different websites she may remember from the past.
3. Finding good resources can be easy for Sophia but there are times when she has issues. The real problem is locating trusted sources that do not have misinformation.
4. She is a kinesthetic learner, using examples that she can work through is the best way to learn.
5. Yes, it is difficult for Sophia to find quality resources. Some resources are unreliable and hard to trust in more advanced subjects.
6. She has found sources she would like to share with classmates. She just struggles with knowing the best way to do that.
7. Normally she emails or texts the source link, or just gives the answers to them willingly.
8. Sophia spends nearly $500 a year on school resources. These resources are books and Chegg.
9. Sophia almost always needs multiple resources to learn a subject.

Fig. 3: Answers to the list of interview Fig 1. questions. Each question is summarized answers from each of Sophia Harrigan's answers.

Sophia Harrigan is a student at the University of Nevada, Reno. We chose Sophia because she is a student and does research for school. In Fig. 3 provides answers to the list of interview Fig 1. questions. Each question is summarized answers from each of Sophia Harrigan's answers.

Zachary Johnson's Responses
1. He doesn't have a subscription currently but uses his friend's log-in occasionally, it helps him study difficult assignments. Class assignments can get so hard he needs a bit more assistance to solve the questions.
2. Typically he uses google searches, asking professors, and if class books are good, then using those too.
3. Some books are written for other professionals who already have a grasp of the topic. When looking on the internet, most websites only include small portions of the question. Also with new material, it's difficult to ask the right questions.
4. He is a Visual learner who needs to see someone work through an example for better understanding but also a hands-on kinesthetic learner. if he's not physically doing the homework or process it can be difficult to really grasp the topic fully.
5. When topics get more advanced it gets harder to find resources. While he is taking senior level courses in school, the available resources for good information decline dramatically.
6. Yes, he goes on about how time-consuming it can be to find quality sources and misinformation online.
7. He uses email and peer-to-peer interaction. Otherwise, he cannot come up with other easy ways to share.
8. After paying for Chegg, school books, and tutors it's probably around $700 a year or more for him. If he was no longer a student it would be much cheaper.
9. Not only does he need more resources, but Zachary also discusses how it can take more than he can find. Some topics get too difficult and can have limited easy to find options.

Fig. 4: Answers to the list of interview Fig 1. questions. Each question is summarized answers from each of Zachary Johnson's answers.

Zachary Johnson is a student at the University of Nevada, Reno. He is a member of Team 22 and is a Computer Engineering student. We interviewed Zachary to determine In Fig. 4 provides answers to the list of interview Fig 1. questions. Each question is summarized answers from each of Zachary Johnson's answers.

High-Level Business Requirements

1. Implement a well-organized website that all users can navigate and use
2. Mitigate the addition of malicious and inappropriate links
3. Keep the project's cost less than $100
4. Create a product that reduces the need for cheating on homework and tests preparation
5. Only allow students and teachers the option to sign up for an account
6. Give all users access to resource links
7. Allow registered users the option to upvote or downvote links
8. Allow registered teachers the option to petition inappropriate links
9. Keep Lecture Goggles free to all users

Technical Requirements Specification

Functional Requirements

High Priority

1. Lecture Goggles shall allow users to register for an account
2. Lecture Goggles shall allow users to log into their account
3. Lecture Goggles shall allow users to view posted content
4. Lecture Goggles shall allow users to vote on posted content
5. Lecture Goggles shall allow users to search for resources by topic
6. Lecture Goggles shall permit users to upload resources
7. Lecture Goggles shall permit all users to submit error reports
8. Lecture Goggles shall permit users to submit hyperlinks to resources
9. Lecture Goggles shall allow administrators to remove resources
10. Lecture Goggles shall allow all users to report inappropriate content

11. Lecture Goggles shall allow users to subscribe to notifications for selected topics and uploaded resources.
12. Lecture Goggles shall support two-factor authentication
13. Lecture Goggles shall be able to accept donations
14. Lecture Goggles shall permit educators to have a verified user account
15. Lecture Goggles shall allow educator accounts to request removal of content

Low Priority

16. Lecture Goggles shall have a public facing API
17. Lecture Goggles shall have a mobile app

Non-Functional Requirements

1. Lecture Goggles shall be implemented using the Django web framework
2. Lecture Goggles shall be available for use 99% of the time during any 24 hour period
3. Lecture Goggles shall be resistant to all attacks listed in the OWASP Top 10
4. Lecture Goggles shall perform a search in no more than three seconds
5. Lecture Goggles shall allow users with a valid .edu email address to register for an account
6. Lecture Goggles shall have a valid SSL certificate for all domains and subdomains
7. Lecture Goggles shall be GDPR compliant
8. Lecture Goggles shall be designed to be a responsive, mobile-first website
9. Lecture Goggles shall be PCI-DSS compliant
10. Lecture Goggles shall use MongoDB to store locations of resources
11. Lecture Goggles shall not require users to transfer over 2MB of data on a fresh page load.
12. Lecture Goggles shall be a progressive web application

Use Case Modeling

In Fig. 5 an image of the Use Case Diagram displays the connections between actors and use cases. Each colored line represents an actor to keep track of interaction

between the actors and use cases. To get a better description of each use case Fig 6. has a 2 to 4 sentence long description of each use case.
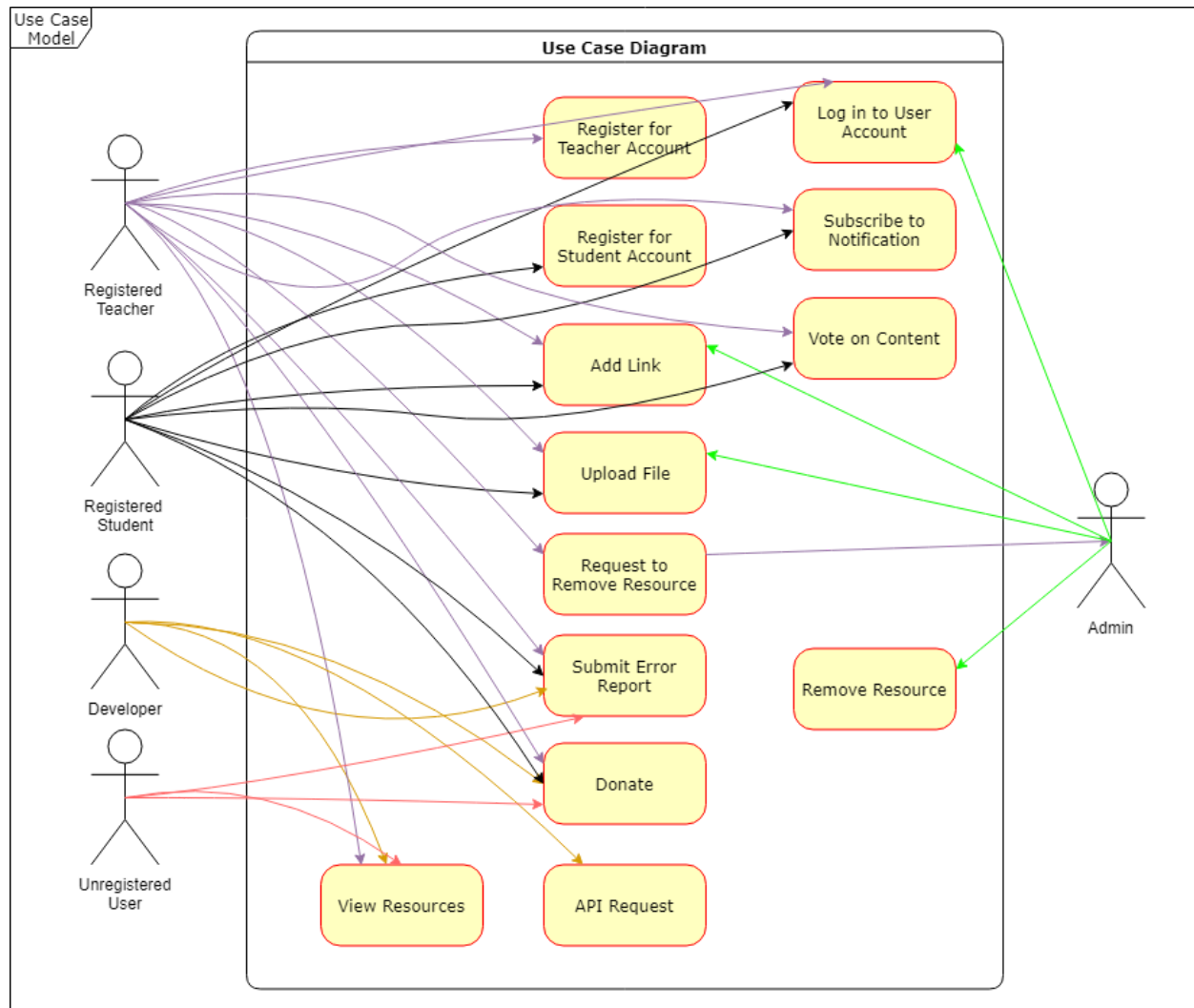

Fig. 5: Image of Lecture Goggles' Use Case Diagram. Each user points to Lecture Goggles' software features.

Detailed Use Cases
[UC1] *Register for Teacher Account*: Using a verified teacher email, each teacher can register for an account. These accounts are given more privilege to report and request the removal of inappropriate material. These accounts will be free for all teachers.
[UC2] *Register for Student Account*: Using a verified student email, students are given the ability to add links, upload files, donate, and submit error reports. Only verified students can register and if banned will remove access from uploading

resources. These accounts will be free for all students.

[UC3] *Add Link*: Using a popup box, all teacher and student accounts will have the access rights to add links to outside sources. These links will be compared to already available links, if the link is already posted, then it will inform the poster. If the link has not already been added, then a new post will be created.

[UC4] *Upload File*: All teacher and student accounts will have the access rights to upload files. Each uploaded file will require an accepted file format and description of the content of the file. Each file will be checked for malicious content to keep users safe.

[UC5] *Request to Remove Resource*: Each verified teacher account has the option to request removal of inappropriate links. Initial implementation will require the admin to review each request. After review, the admin has the choice to either remove or keep them available to users. Reported links will be frozen until review.

[UC6] *Submit Error Report*: All users will have access to submitting error reports. Since admins cannot know of all issues, having a good reporting system is key to the web pages success. These reports will go directly to admins for further review.

[UC7] *Donate*:There will be expenses to keep the web page up and running. Since Team 22 wants to keep the website free to every user, we will implement an optional donation section. Student accounts, teacher accounts, and anonymous users will be given the option to donate.

[UC8] *API Request*: Programmers will be given access to the website API. The API responds to programmers requests with values asked for. These values range from available topics to sub-content of the topics.

[UC9] *View Resources*: All users, whether registered for an account or not, have the ability to access all resources. This includes links and uploaded files. Although the platform requires an account to upload data, having access to resources will remain open to all users.

[UC10] *Remove Resources*: Only admin will have access to remove links and files. Typically after receiving a submitted report from a verified teacher account, the admin will review the complaint and remove the resource if needed. At any time admin can remove a link or file they deem inappropriate.

[UC11] *Log in to user account:* After teachers and students register for an account they will have access to log in. Once logged in they will have access as a teacher or student account.

[UC12] *Subscribe to notifications:* Registered users will have the ability to turn on notifications. These notifications, even if the webpage is not open, will be pushed to

their desktop.
[UC13] *Vote on posted content:* Using a registered account, users can upvote or downvote webpage content. This will allow users to find good content faster.

Fig. 6: Detailed use case descriptors. Contains each use case and the detailed description of each use case.

Use case: Add Link
ID: 1
Brief description: A registered user posts a useful link for a school subject. This link will be posted on the webpage.
Primary actors: Registered Teacher and Registered Student.
Secondary actors: None.
Preconditions: The user has registered for an authorized account.

Main flow:
    1. The use case starts when the Registered Account User selects "Add Resources"
    2. The system displays a selection of types of submission.
    3. The User selects the appropriate option for upload.

Postconditions:
    1. System checked for malicious content.
    2. If the content was not malicious it posts it under the section it was added to.

Alternative flow: None.

Fig. 7: A single use case description from Fig 5. Includes ID, Brief description, Primary actors, Secondary actors, Preconditions, Main flow, Postconditions, Alternative flow.

Use case: Log in to User Account
ID: 2
Brief description:

| Each registered user can log into their created account. |
|---|
| Primary actors:
Registered Teacher and Registered Student. |
| Secondary actors:
None. |
| Preconditions:
Users must have registered for an account already. |
| Main flow:
    1. The user with a registered account selects log-in.
    2. A system display shows a Username and Password input box.
    3. The user inputs their credentials and selects the log-in button.
        3.1 The system displays a loading cursor while waiting. |
| Postconditions:
    1. The user is given registered account privilege. |
| Alternative flow:
    1. The user with a registered account selects log-in.
    2. A system display shows a Username and Password input box.
    3. The user inputs their credentials and selects the log-in button.
        3.1 The system displays a loading cursor while waiting.
        3.2 Display red text "Username or Password was incorrect, please register for an account or try again" |

Fig. 8: A single use case description from Fig 5. Includes ID, Brief description, Primary actors, Secondary actors, Preconditions, Main flow, Postconditions, Alternative flow.

Requirement Traceability Matrix

	UC1	UC2	UC3	UC4	UC5	UC6	UC7	UC8	UC9	UC10	UC11	UC12	UC13
FR1	X	X											
FR2											X		
FR3									X				
FR4													X
FR5									X				
FR6				X									
FR7						X							
FR8			X										
FR9										X			
FR10					X								
FR11												X	
FR12											X		
FR13							X						
FR14	X												
FR15					X								
FR16								X					
FR17	X	X	X	X	X	X	X		X		X	X	X

Fig. 9: Table demonstrating the relationship between use cases and functional requirements.

Initial Snapshots
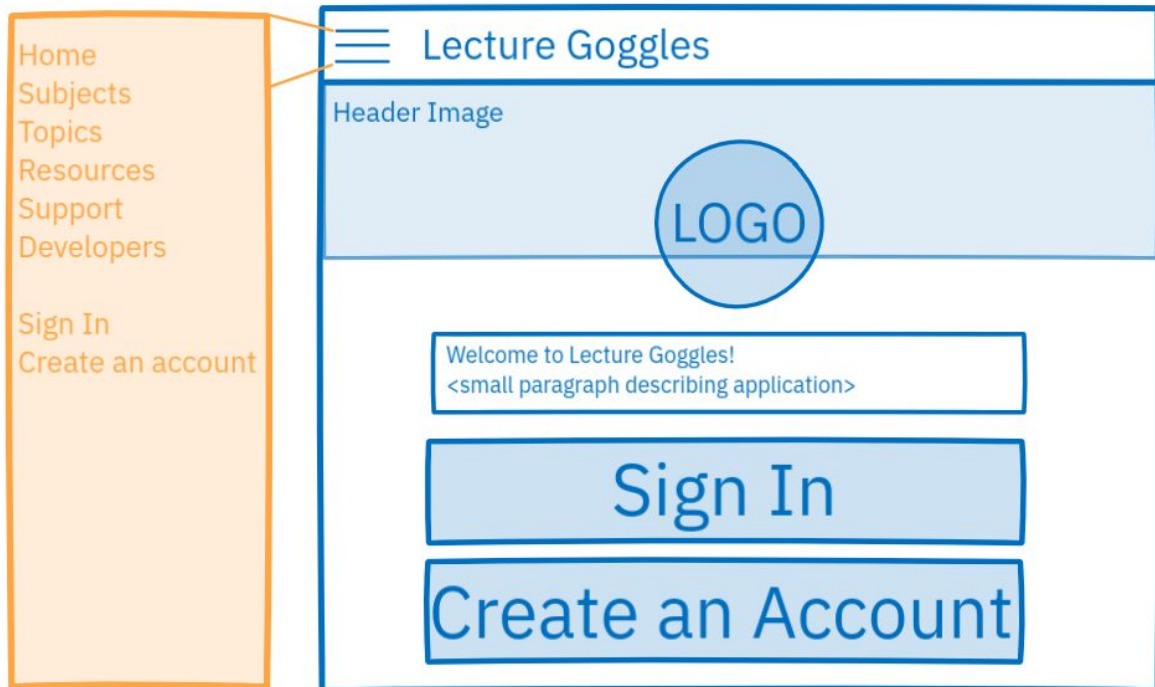
Fig 10: Wireframe of home page when not signed in
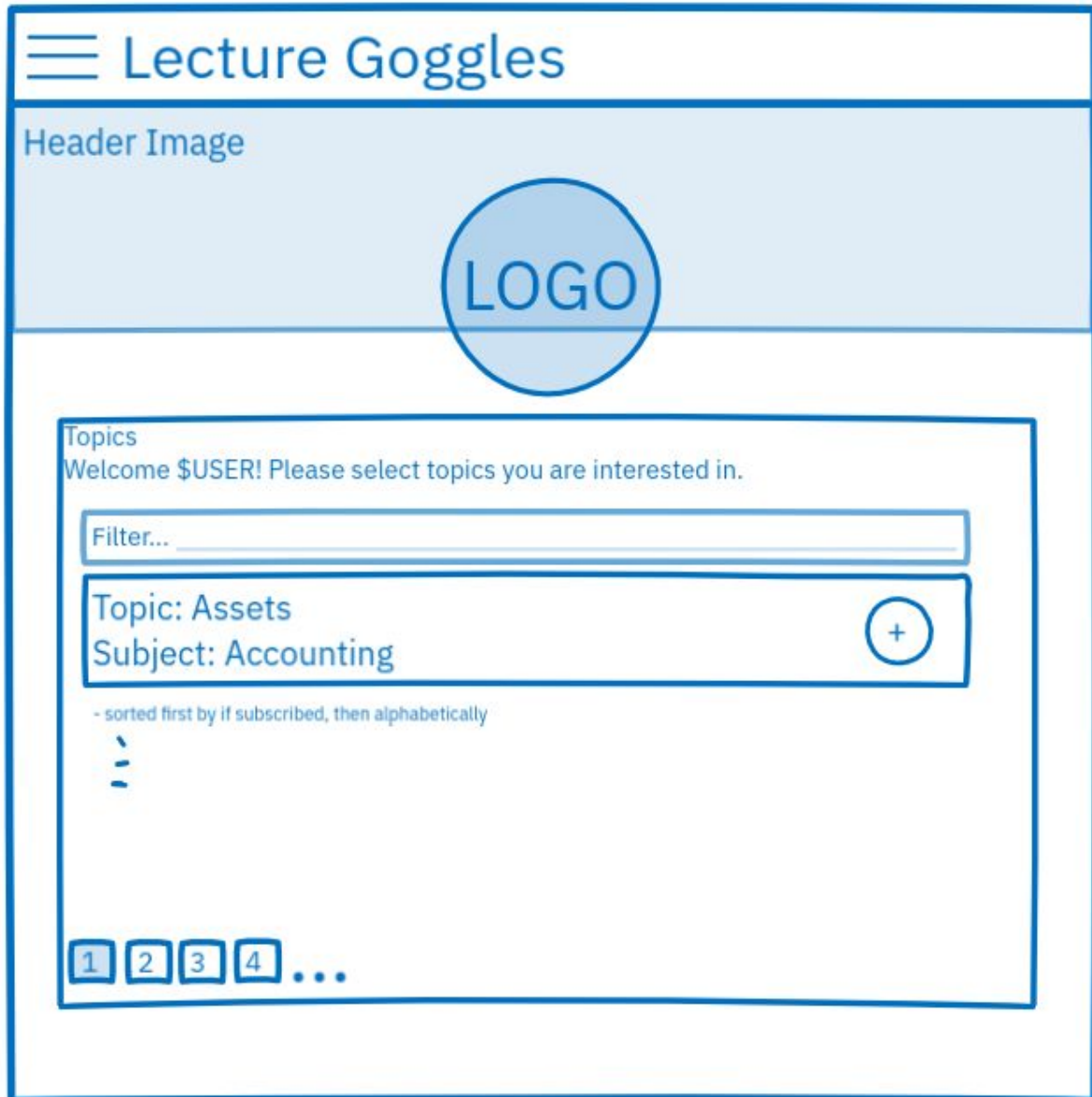
Fig. 11: Wireframe demonstrating Account creation
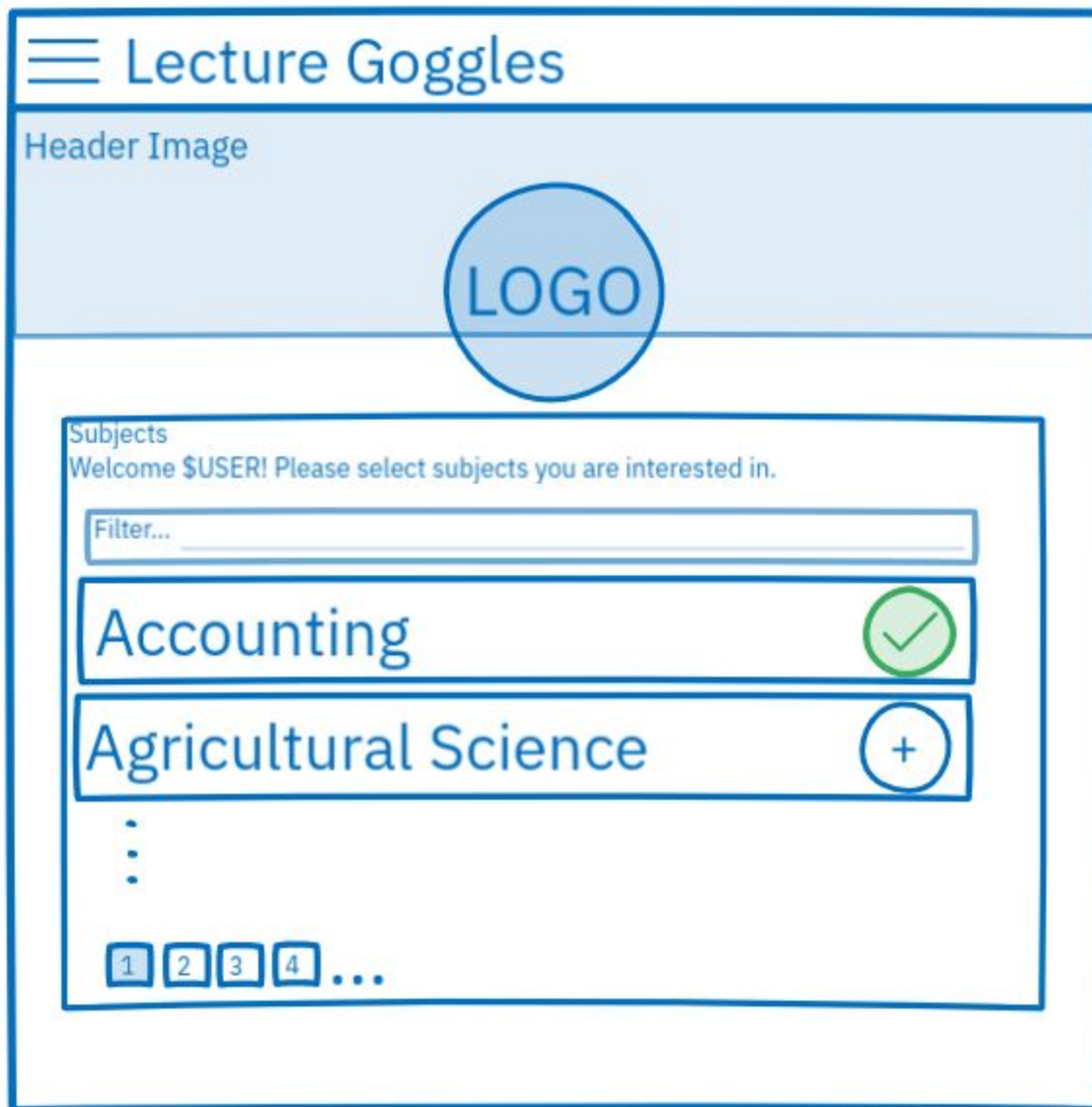
Fig. 12: Wireframe demonstrating subscribing to subjects upon account creation and in user preferences

Lecture Goggles

Header Image

LOGO

Topics
Welcome $USER! Please select topics you are interested in.

Filter...

Topic: Assets
Subject: Accounting (+)

- sorted first by if subscribed, then alphabetically

[1] [2] [3] [4] . . .
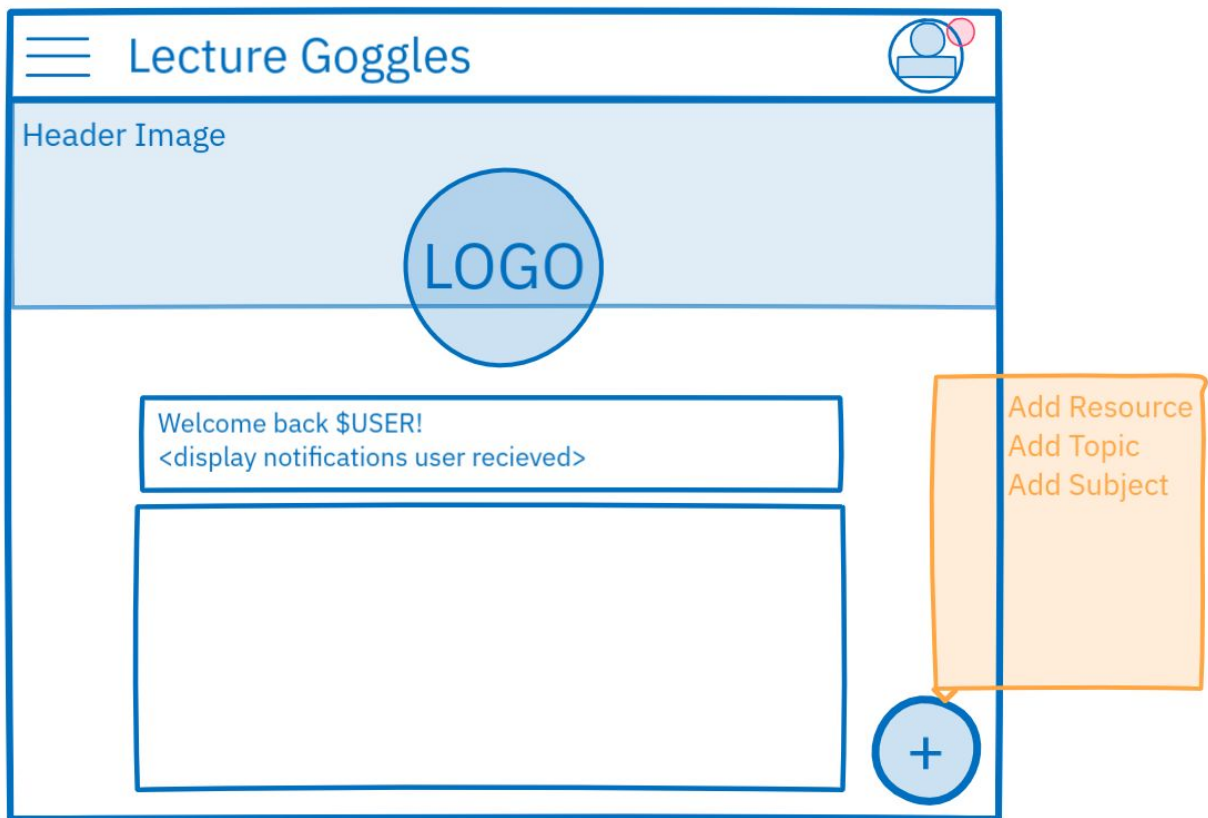
Topic Subscrption

Fig 13: Wireframe demonstrating topic subscription upon account creation and in user preferences
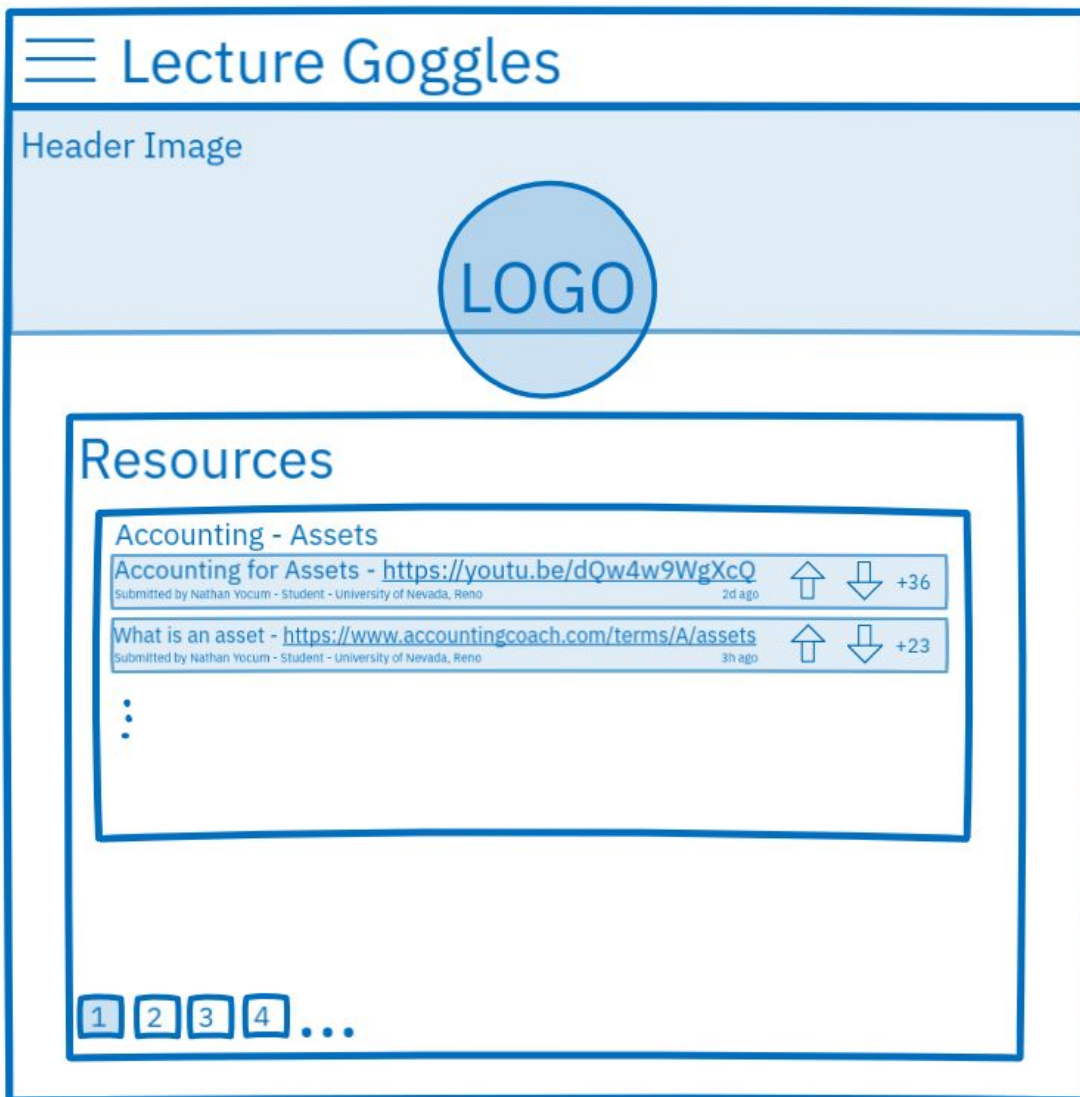
Fig. 14: Overview of browsing resources, upvoting, and downvoting

Fig. 15: Wireframe of homepage when signed in

Link to the project on InVision
(https://freehand.invisionapp.com/freehand/document/q48jpRvIX)

Glossary

1. **Application Program Interface (API)**
   An application program interface (API) allows developers to interface directly with a web application for the purpose of developing another application. APIs allow developers to interact directly with a website, such as retrieving information or uploading content.

2. **Backend Web Development**
   Backend web development refers to working on the parts of a website that users do not directly interact with. This can include tools, servers, and databases that a developer creates to manage a website.

3. **Database**

A database is a set of data stored on a server. Databases hold important information, such as user accounts, hashed passwords, and other important information for websites to function.

4. **Framework**

A framework is a tool, or set of tools, developed to make development easier. For web development, common frameworks include React, Angular, and Vue for frontend and Rails, Express, and Django for backend. Frameworks automate parts of development and design and allow new design patterns to form.

5. **Frontend Web Development**

Frontend web development refers to the development of the user facing application. The front end of a website interacts directly with the web browser.

6. **Non-relational Database**

A a non-relational database (often called a NoSQL database) is a database that does not follow the design paradigms of relational databases. They are better suited to handle unstructured data, scaling, and flexible models.

7. **Open Source Software**

Open source software is software constructed in a way that the source code is readable and modifiable. Software can be constructed to be different degrees of open source (fully open source and partially open source), but all open source software allows users to read the source and modify/change features based upon the license.

8. **Permissions**

User permissions refer to the features different users have access to. For example, a developer can access the API, but cannot deny resources from approval. Meanwhile, an admin may not have API access, but will be able to deny or approve resources.

9. **Progressive Web Application**

A progressive web application (PWA) is an application that uses browser features to act similar to a native app on a platform. Using tools such as service workers, manifests, and caching, PWAs allow fast, reliable, and engaging experiences in web applications

10. **Resource**

A resource, in context of the application, is a link or reference a user can follow to learn about a selected topic. Resources are a source of information which users can learn from.

11. **Server**

A server manages the website, delivering the front end code to the browser and handling back end code and operations. Servers provide many services to users and developers such as sharing and modifying data.

12. **Subject**

A subject is a collection of topics.

13. **Topic**

A topic is the subject to which a resource discusses.

14. **User Interface**

User interface refers to the parts of the website the user interacts with. The user interface directly affects the user experience, functionality of the software, and the usability of the software.

15. **Web Client**

The web client refers to the software users use to interface with websites. This is often a web browser (such as Chrome) but can also sometimes be an operating system or a web crawler. Clients display the application's front end.

16. **Web Site**

A web site (or application) refers to a site or application hosted on the world wide web.

List of References

Book:
- Orr, Dominic, et al. *Open Educational Resources: a Catalyst for Innovation*. OECD, 2015.
  - Orr et al. advocates for the usage and expansion of open education resources (OERs). This work explains how the availability of OERs can help expand the way students learn, allow teachers and students to interact in new ways, and facilitate learning at all levels. This work also explains how OER distributions models can be combined to become sustainable and useful to students.

Articles:
- McShane, Michael Q. "Open Educational Resources." *Education Next* 17.1 (2017) *ProQuest.* Web. 2 Nov. 2018.
  - This work details the OER initiative and what efforts have been made to make it more available to students and educators. It discusses the efforts of the US Department of Education, and how ineffective its OER repository is. It also discusses the efforts put forward by the Department of Education to work with the many States to develop more OERs available to students and teachers alike.
- Christine L. Ferguson (2017) Open Educational Resources and Institutional Repositories, Serials Review, 43:1, 34-38, DOI: 10.1080/00987913.2016.1274219

- This resource discusses how institutions catalog and retain educational information. It discusses the issues regarding the storage and usage of copyrighted materials. This work also discusses issues regarding versioning in educational repositories. This will be useful in identifying how to deal with potential duplicates or multiple versions of a submitted resource.
- Mouriño-García, Marcos, et al. "Cross-Repository Aggregation of Educational Resources." *Computers & Education*, Pergamon, 5 Oct. 2017, www.sciencedirect.com/science/article/pii/S036013151730218X.
  - This work discusses the number of educational resources, their advantages and their drawbacks. These authors propose a system of classifying educational resources using machine learning algorithms and metadata. This method allows for efficient categorization with only minimal human input or correction. Their system was able to categorize resources hosted on other systems.

Websites:
- "200 Free Kids Educational Resources: Video Lessons, Apps, Books, Websites & More." *Open Culture*, www.openculture.com/free_k-12_educational_resources.
  - This website has an enormous list of hyperlinks to free educational resources all around the internet. Though they are listed as K-12 resources, many remedial or introductory level college courses cover similar or identical content, making them excellent resources to use with Lecture Goggles
- Khan, Salman. "Khan Academy." *Khan Academy*, Khan Academy, www.khanacademy.org/.
  - Khan Academy is an excellent resource for many topics, but especially math and science based topics. Khan Academy does not require an account to use their resources, so it is easy to locate them using hyperlinks. These videos, worksheets, and interactive problems will be excellent initial resources to add to Lecture Goggles.
- Fox, Carolyn. "A Guide to Free and Open Source Education." *Opensource.com*, 2013, opensource.com/education/13/4/guide-open-source-education.
  - This page points to multiple large databases containing thousands of educational videos and courses. Some of the resources mentioned are designed to work in a similar way to Lecture Goggles, which will help us narrow down the specific functionalities that are well received, and which ones are not. The videos will be excellent to add to our initial release of Lecture Goggles.

Contribution of Team Members

Zachary Johnson did the Introduction, Interview section, High-Level Business Requirements, and Use Case Model. Zachary worked on the project for approximately 10 hours.

Logan Long did the Technical Requirements Specification, Requirement Traceability Matrix, and List of References. He also assisted in the Use Cases. Logan worked on the project for approximately 9 hours.

Nathan Yocum did the Initial Snapshot, Glossary, and added to the Technical Requirements. Nathan worked on the project for approximately 10 hours.

All three members were involved with checking each other's work and removing errors.