**Software Engineering 14:332:452**

Group: 9

Project Title: Minerva

Project Site URL: <https://rutgerssoftwareengineering.github.io/Minerva/>

Date Submitted: February 24, 2019

Team Members: Jacob Battipaglia, Jonathan Hong, Skyler Lee, Brian Ma, Justin May, Salman Omer, Joshua Olazo, Phurushotham Shekar

# Individual Contributions Breakdown

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  | Jacob | Jon | Skyler | Brian | Justin | Salman | Joshua | Phuru |
| Project Management |  | 12.5% | 12.5% | 12.5% | 12.5% | 12.5% | 12.5% | 12.5% | 12.5% |
| Section 1 | Prob Stat. |  |  |  |  | 55% |  | 45% |  |
| Gloss | 15% | 15% | 15% | 15% | 5% | 15% | 5% | 15% |
| Section 2 | Funct req | 50% |  |  | 50% |  |  |  |  |
| Non funct | 30% |  |  | 30% |  | 40% |  |  |
| UI req | 25% |  |  | 25% |  | 15% |  | 35% |
| Section 3 | Stkholders |  |  |  |  | 100% |  |  |  |
| Actor/Goal | 50% |  |  | 50% |  |  |  |  |
| UC casual |  | 42.5% |  |  |  | 42.5% | 15% |  |
| UC diag | 50% | 50% |  |  |  |  |  |  |
| Trace Matrix |  |  |  | 100% |  |  |  |  |
| UC full |  |  |  | 15% |  | 45% | 40% |  |
| Section 4 | Prelim desig |  |  | 20% | 20% |  | 25% |  | 35% |
| Eff. Est. |  |  |  |  |  |  |  | 100% |
| Section 5 | Concepts |  |  |  |  |  | 60% | 40% |  |
| Assoc |  |  |  |  |  | 60% | 40% |  |
| Attrib |  |  |  |  |  | 60% | 40% |  |
| Contracts |  | 47.5% |  | 5% |  |  |  | 47.5% |
| Math Model |  |  |  |  |  |  |  | 100% |
| Section 6 | Size estim | 100% |  |  |  |  |  |  |  |
| Section 7 | Plan of Work | 12.5% | 12.5% | 12.5% | 12.5% | 12.5% | 12.5% | 12.5% | 12.5% |

* Jacob Battipaglia
  + Enumerated functional requirements/user stories
  + Defined actors and goals for numerous use cases
  + Use Case Diagram
  + Effort Estimation
* Jonathan Hong
  + Defined various terms to be used throughout the proposal
  + Defined casual descriptions of use cases
  + Use Case Diagram
  + Enumerated pre- and post- conditions per use case under System Operation Contracts
  + Formatting
* Skyler Lee
  + Contributed to problem statement
  + Created Basic UI
* Brian Ma
  + Enumerated functional requirements/user stories
  + Laid out project management and plan of work
  + Defined various terms to be used throughout the proposal
  + Listed use cases
  + Created and specified basic UIs for Forum
  + Created traceability matrix
* Justin May
  + Added all of the Stakeholders
  + Created the Github organization, sub-teams, git-pages branch, and temporary website. Created Slack channel
  + Contributed 50% to problem statement.
  + Created the References. Found over 80% of cited studies and cited them.
* Salman Omer
  + Contributed to non functional and appearance requirements
  + UC-1, UC-2, UC-12 and UIs for them (taking/making quizzes and managing resources)
  + Domain models and analysis
  + General proofreading, checking and feedback
* Joshua Olazo
  + Contributed 50% to problem statement
  + Created some references. The other cited studies and cited them.
  + UC-3, UC-4, UC-5, UC-6, UC-7, UC-8, UC-9, UC-10, UC-11
  + Added extra requirements
* Phurushotham Shekar
  + 2c. Contributed to on-screen appearance requirements
  + 2c. Designed the User Interface
  + Contributed 10% to the Plan of Work
  + User Effort Estimation
  + Mathematical Model
  + Contributed 20% to System Operation Contracts

**Table Of Contents**

|  |  |
| --- | --- |
| Section | Page # |
| 1. Customer Problem Statement | 5 |
| a. Problem Statement | 5 |
| b. Glossary of Terms | 10 |
| 2. System Requirements | 12 |
| a. Enumerated Functional Requirements | 12 |
| b. Enumerated Non-Functional Requirements | 13 |
| c. On-Screen Appearance Requirements | 14 |
| 3. Functional Requirements Specification | 15 |
| a. Stakeholders | 15 |
| b. Actors and Goals | 15 |
| c. Use Cases | 18 |
| 4. User Interface Specification | 33 |
| a. Problem Statement | 33 |
| b. User Effort Estimation | 41 |
| 5. Domain Analysis | 42 |
| a. Domain Model | 42 |
| b. System Operation Contracts | 50 |
| c. Mathematical Model | 51 |
| 6. Project Size Estimation | 52 |
| 7. Plan of Work | 54 |
| b. Resources | 55 |

**1. Customer Problem Statement**

This section will contain the problem as described by the customer and what type of system they want in order to solve their issues.

**a. Problem Statement**

With rising University education costs in the United States, the question of the efficacy of a bachelor's degree is put more and more into question. A Georgetown Public Policy Institute study puts the percentage of American jobs that require post-secondary education beyond high school at 65 *per cent* by the year 2020; however, as more and more Americans attend universities across the country, professor to student ratios will rise and university infrastructure will struggle.[[1]](#footnote-0) In addition, as class sizes continue to increase new challenges and barriers—such as decreasing student in-class interaction—arise that prevent the retention of information and the educational process.[[2]](#footnote-1) A report from the Cornell School of Industrial Labor Relations indicated “that both class size and student load negatively impact student assessments of courses and instructor”

Skyrocketing university tuition, in conjunction with ballooning classroom sizes, could positively benefit from the introduction of innovative software in the classroom.

One of the largest areas of concern is with classroom size. College attendance has increased by 28 *per cent* since 2000, and various studies have demonstrated a negative impact on student performance from classroom size.[[3]](#footnote-2),[[4]](#footnote-3) In fact, the National Center for Academic Transformation estimated that only 25 of the most attended college courses contributed to over 35 *percent* of four-year college attendance annually.[[5]](#footnote-4)

**Common complaints from large classroom (Students):**

1. **Hard to have a voice**
   1. **Difficult to ask questions to the professor without disrupting the lecture**
      1. **Sometimes class time is wasted on less important questions**
      2. **Speaking in front of the entire class is intimidating**
      3. **Especially hard to ask for clarifications when it is your fault for not paying attention or missing the last class**
      4. **Even “good questions” could lead to a back and forth which could waste class time**
      5. **Asking questions could lead to judgement from classmates**
   2. **Awkward to give feedback to professor during class**
      1. **Hard to ask for special accommodations such as professor speaking louder**
      2. **Individual problems may not be worth it to disrupt lecture**
      3. **May not want to give negative feedback in a non anonymous way**
2. **Seating can be important based on the professor’s style of teaching**
   1. **For lectures that depend on complex diagrams and equations that are presented on the blackboards, it is often difficult for students farther from the front to understand and copy what is being written down**
   2. **Some professors possess poor or no equipment to project their voice, making it much more difficult to listen to students not in the front**

**Common complaints from large classrooms (Professors):**

1. **Difficult to keep students engaged**
   1. **Cannot keep students attention for students in the back**
2. **Hard to assess students’ understanding**
   1. **Paper and Pen assessments take limited lecture time and need to be graded by hand**
3. **Lack of feedback from students**
   1. **Negative feedback usually only comes in when semester in anonymous survey**
4. **Discussion amongst students interrupts lecture**
   1. **Cannot tell meaningful discussion apart from idle chat**
5. **Often held back after class so that students can ask questions**
   1. **Long queues can take away too much time from professors**
   2. **Some professors have another class to teach, and must redirect students to email him/her, unable to give an immediate response**
6. **Misuse of emails and online forums**
   1. **Students may send emails with inquiries that can be solved through Sakai forums/resources, cluttering the professors inbox**
   2. **Some inquiries posted on online resources end up being unanswered due to the nature of the problem only being possibly solved by the professor him/herself**

A recent Kennesaw State University study using student gaze as a proxy for attention found specific in-class methodologies and factors that increased student attention. Some factors were out of the control of the professor; students that sat in the back of the classroom showed less overall attention that students that sat in the middle and front, which would be exacerbated with increased classroom size.[[6]](#footnote-5) The study found that students who sat on the sides and back of the classroom would inadvertently see the computer screens and phone screens of students in front of them, of which texts and Facebook seemed to be the largest contributor to distractions.[[7]](#footnote-6) In addition, student and professorial back and forths seemed to indicate a large increase in attention, yet was polarizing in that some students would quickly lose attention.[[8]](#footnote-7) Finally, a large predictor of student distraction was pre-printed or pre-released class notes, suggesting that students that could only access notes in class tended to pay more attention.[[9]](#footnote-8)

Where other industries, such as transportation and the hotel industry, have greatly innovated over the past few decades, education continues to lag behind. The college classroom has seldom changed over the past few decades and the pedagogical methods professors adopt with it. Technology has the potential to solve various problems plaguing the classroom. Most students already the technology needed, “83% own a laptop, and over 50% have a Smartphone”[[10]](#footnote-9). Additionally, classrooms are already equipped to interact with students through technology, “97% of classrooms had one or more computers, and 93% of classroom computers had Internet access.”[[11]](#footnote-10)

Currently some college lectures are using IClickers for in class polling and quizzes. This solution is bad for students because it does not leverage the smartphones and laptops that students already carry. Requiring students to buy specialized hardware and carry it to class everyday is less than ideal

1. **Problems with IClickers**
   1. **Cost money to buy and app needs to be paid per semester**
   2. **Have to carry the remote to class**
   3. **University has to buy receivers**
   4. **Can only answer simple multiple choice questions**
   5. **Cannot see questions and answers after class for review**

There exists demonstrable and specific actions that can greatly increase the attentiveness of students in class that can counteract the trend of rising classroom numbers, giving greater benefit from the rising costs of colleges. The aforementioned studies outline key moments in the classroom that can be eliminated or improved upon with technology. Bridging the generation gap between antiquated classrooms and 21st century technology can provide a panacea for the modern ailments of education.

Another domain of this problem is out of class information retention. For example, there exists an asymmetry between the homework knowledge and student knowledge and thus students will often revert to cheating or copying homework answers when it gets too challenging. If the purpose of homework and quizzes is to test student knowledge and encourage out of class exploration, it fails because the friction is too high (student frustrations often trump the desire to learn). Thus, a system that is knowledgeable of student flaws and weaknesses, possibly attaining this knowledge from homework and quiz answers, would suppose the place of a tutor. This type of personalized learning would increase a students willingness to learn. For example, specific Laplace transforms presuppose knowledge about integration by parts. A good tutor would have identified a student weakness with integration by parts, perhaps by a low grade in a prior quiz, and provide a integration by parts problem before attempting the Laplace transform problem.

**Common complaints of existing classroom technology:**

1. **Sakai/Canvas/Piazza**
   1. **Hassle to log on and switch between different sites for different classes**
   2. **Lack of connectivity and integration between sites**

# 

# **b. Glossary of Terms**

The following is a list of terms which may appear throughout our report and are necessary for complete understanding of our design and development process. These terms may describe technical features as well as functional capabilities of our software.

* Polls - This feature allows professors to ask short, multiple choice questions during class. Students then have the ability to answer the questions from their phones. After the professor closes the question, the application will show the distribution of answers from the students in class.
* Mobile App - a separate downloadable app that links with the website to provide extra functionality not available on the web page, but does not contain the same features as the web page
* Public Forum - a collection of messages that students can reply to or ask questions; a place of discussion for both the instructor and his/her students
* Private Forum - The student-only forum that professors and TAs do not have access to
* Upvote - a quantifiable value that represents the approval or agreement from fellow classmates
* Downvote - an expression of disapproval or disagreement about a post or comment
* Large Classroom - a learning environment where the student to professor ratio exceeds at least 50:1, making teaching almost impractical
* Active Feedback - Feedback given to the professor during the class, through the mobile app, generally about more specific actions. E.g. “talk louder”
* Passive Feedback - Feedback given to the professor at any time, through the web site, generally about more broad actions. E.g. “Plan more lectures around the blackboard”
* Platforms - the devices that users will be able to access our features through (smartphones or desktops/laptops)
* I-Clicker - small handheld device used to answer multiple choice questions the professor posts during class. Sometimes mandatory for classes
* Account - a collection of user-related information (username, class-enrollment, professor or student, etc.) that is only accessible to the owner of that information through username and password authentication
* Online Office Hours - live video recordings for professors to provide additional academic help or clarifications

# 

# 

# 

# 

# 

# 

# 

# 

# 

# 

# 

# 

# 

# 

# 

# 

# 

# **2. System Requirements**

**a. Enumerated Functional Requirements**

1. Out of Class Requirements (Web Page)

|  |  |  |
| --- | --- | --- |
| **Label** | **Priority Weight** | **User Story** |
| REQ-1 | 4 | As a student or professor, I should have the ability to post and answer questions, and communicate with the other members of my class via an online forum |
| REQ-30 | 4 | As a student or professor, I should have the ability to view announcements |
| REQ-31 | 3 | As a professor, I should have the ability to create announcements |
| REQ-2 | 3 | As a student, I would also like to have access to a private, student-only forum for discussing class topics with my peers and arranging things like tutoring sessions or study group meetings |
| REQ-3 | 2 | As a user, I will have the ability to access and view grades for assignments, and update grades if I have permission |
| REQ-4 | 2 | As a professor, I will be able to upload files such as slides, additional reading, and other class resources for my students |
| REQ-5 | 4 | As a student, I will be able to access class resources uploaded by my professor, as well as view my current performance in the class and on individual assignments |
| REQ-6 | 1 | As a professor, I will have the ability to post quizzes and gather specific information on what topics my students have fully mastered or identify weak areas. |
| REQ-7 | 1 | As a professor, I will have the ability to host online lectures to provide additional out-of-class help for my students. |
| REQ-8 | 5 | As any user, I do not want to go to multiple sources in order to access class information. (Want to combine Piazza + iClickers + Sakai) |

2. In class Requirements (Mobile App for Students, and Web Page for Professors)

|  |  |  |
| --- | --- | --- |
| **Label** | **Priority Weight** | **User Story** |
| REQ-9 | 5 | As a student, I can use the polling feature of Minerva to participate in in-class quizzes and questionnaires |
| REQ-10 | 5 | As a student, I can use the app to give live feedback and anonymously ask questions to my professors during lecture |
| REQ-11 | 4 | As a student, I can view questions that my peers have asked and use a voting system to vote on questions that I agree with |
| REQ-12 | 5 | As a professor, I can view questions that are submitted by my students and see which have been “upvoted” by other students for visibility. |

**b. Enumerated Non-Functional Requirements**

1. Out of Class Requirements (Web Page)

|  |  |  |
| --- | --- | --- |
| **Label** | **PW** | **User Story** |
| REQ-13 | 5 | As a professor, I should not be able to view the student-only forum |
| REQ-14 | 4 | As a professor, I shall have the ability to control what students are registered for my particular class. |
| REQ-15 | 3 | As a user, I should be able to change my password if I forget it. |
| REQ-16 | 3 | As a user, I should be able to access past chat and forum information. |
| REQ-17 | 2 | As a user, my personal data, such as what classes I am enrolled in, shall persist across all platforms. |
| REQ-18 | 1 | As a user, I will have the option to have the website and app remember my password and log me in automatically |

2. In class Requirements (Mobile App for Students, and Web Page for Professors)

|  |  |  |
| --- | --- | --- |
| **Label** | **PW** | **User Story** |
| REQ-19 | 5 | As a user, I should be able to view questions asked in class in real time |
| REQ-20 | 1 | As a user, I will have the option to have the website and app remember my password and log me in automatically |

**c. On-Screen Appearance Requirements**

1. Out of Class Requirements (Web Page)

|  |  |  |
| --- | --- | --- |
| **Label** | **Priority Weight** | **User Story** |
| REQ - 21 | 5 | There needs to be a log in screen that logs me into the relevant type of account, professor or student. |
| REQ - 22 | 1 | As a user, I must not be bogged down in features that are cluttered and poorly organized |

2. In class Requirements (Mobile App for Students, and Web Page for Professors)

|  |  |  |
| --- | --- | --- |
| **Label** | **Priority Weight** | **User Story** |
| REQ - 23 | 5 | There needs to be a log in screen that logs me into the relevant type of account, professor or student. |
| REQ - 24 | 5 | As a student in class, I need to be able to provide feedback to the professor at the touch of a button. |
| REQ - 25 | 3 | As a student in class, I need to be able to ask questions easily and anonymously without too much navigation. |
| REQ - 26 | 4 | As a student in class, I need a screen where I can view other students questions and upvote/downvote them. |
| REQ - 27 | 2 | As a student in class, I want to be able to quickly answer the in-class quizzes at the touch of a button. |
| REQ - 28 | 3 | As a professor in lecture, I need highly upvoted questions to pop up on my screen without me having to navigate anywhere else during lecture |
| REQ - 29 | 1 | As a student in class, I need to be able to easily read poll questions and answers and select them intuitively |

# 

# **3. Functional Requirements Specification**

**a. Stakeholders:**

Professors, Teaching Assistants, Learning Assistants, and University Faculty: this group of people are interested in seeing the benefits of this technology in the classroom to improve student educational retention.

University Governance: this group of people are interested in seeing better student outcomes over time, improved student performance would boost University statistics.

Students: this group of people are interested in benefiting by being able to interact with the learning material in novel ways, as well as give immediate feedback in class.

Prospective Parents and Students, Current Parents: would be attracted to novel uses of technology in the classroom, especially if there is a demonstrable boon in student education.

Academic Researchers: The current academia surrounding student focus is mostly centered around studying the effects in primary and secondary education. Controlled studies could be conducted on post-secondary education, to study and reproduce the existing findings in post-secondary education metrics.

**b. Actors and Goals**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Related Req’s | Actor | Goal | Use Case Name | Priority |
| REQ-6 REQ-9 REQ-27 REQ-29 | Student | View and answer quiz questions on the web site | UC-1 Take Quiz | 5 |
| REQ-6, REQ-9, REQ-27, REQ-29 | Professor | Create and post a quiz on the website for the class to see | UC-2 Create Quiz | 5 |
| REQ-30, | Student  Professor | See any past announcements the professor posted about the class | UC-3 View Announcement | 3 |
| REQ-30, REQ-31 | Professor or TA | Create and post an announcement on the website for everyone to see | UC-4 Create Announcement | 3 |
| REQ-9, REQ-10, REQ-11, REQ-12, REQ-19, REQ-24, REQ-25, REQ-26, REQ-28 | Student | Create a question for other students to see and vote on in the mobile app | UC-5 Create Question (in-class) | 5 |
| REQ-9, REQ-10, REQ-11, REQ-12, REQ-19, REQ-24, REQ-25, REQ-26, REQ-28 | Student | Allows student to move questions they see useful higher on the professor’s list | UC-6 Vote on Question (in-class) | 2 |
| REQ-9, REQ-10, REQ-11, REQ-12, REQ-19, REQ-24, REQ-25, REQ-26, REQ-28 | Professor | Address and delete questions or feedback from the professor’s list | UC-7 Dismiss Question (in-class) | 2 |
| REQ-16 | Student  Professor | View all public forum posts | UC-8 Read Public Forum | 4 |
| REQ-17  REQ-3  REQ-14 | Student  Professor  TA | Look at profile and data related to user | UC-9 Profile View | 3 |
| REQ-18  REQ-20  REQ-23 | Student  Professor  TA | Log into web or mobile app without entering password | UC-10 Remembered Log-in | 1 |
| REQ-15 | Student  Professor  TA | Change password used to authenticate and access account | UC-11  Change Password | 1 |
| REQ-1  REQ-8 | Student  Professor  TA | Post a reply and new thread for everyone else to see and respond to | UC-12 Post in Public Forum | 5 |
| REQ-1  REQ-2  REQ-13 | Student | View all past private forum posts | UC-13 Read Private Forum | 1 |
| REQ-4 | Professor  TA | Post class materials such as lecture slides and videos in one location that students can view and download from | UC-14 Upload Class Materials | 3 |
| REQ-5 | Student  Professor  TA | View and download class materials uploaded by the professor | UC-15 View Class Materials | 3 |
| REQ-21, | Student  Professor  TA | Authenticate account details to allow the user to access their relevant features | UC-16 Log-In | 5 |
| REQ-3 | Professor  TA | Access and update grades for specific assignment or student | UC-17 Modify Grades | 3 |
| REQ-3 | Student  Professor | View grade summary for all assignments in a class | UC-18 View Gradebook | 2 |
| REQ-7 | Professor  TA | Host live stream to teach or review material for students | UC-19 Host Lecture | 1 |
| REQ-14 | Professor  Student | Manage the members registered for a class and the classes that the user is a member of | UC-20 Modify Classes | 2 |
| REQ-22 | Professor  Student | See all the different sections of the website easily | UC-21 View Hub | 2 |

# 

# 

# 

# **c. Use Cases**

## **i. Casual Descriptions**

*UC-1:* Take quiz - to solve timed quizzes posted by professor

*UC-2:* Create quiz - professors can create multiple choice or open-ended questions

*UC-3:* View Announcement - students can see the announcements that professors post

*UC-4:* Create Announcement - professors post important announcements on the website

*UC-5:* Create Question (in-class) - students can propose a question to be asked in class to the professor

*UC-6:* Vote on Question (in-class) - other students can vote on whether a question is important

enough to be asked

*UC-7:* Dismiss Question (in-class) - professors can choose to ignore a question

*UC-8:* Read Public Forum - students, professors, and TA’s can discuss with each other on forum posts

*UC-9:* Profile View - everyone can view information about a particular user

*UC-10:* Remembered Log-in - users can stay logged in without having to enter a password all the time

*UC-11:* Change Password - users can change the password they use to log in with

*UC-12:* Post in Public Forum - anyone can post in the public forum to ask a question, or bring up a

topic of interest for further discussion

*UC-13:* Read Private Forum - only students can read any posts in this forum

*UC-14:* Upload Class Materials - professors and TAs can post any additional resources students may

need

*UC-15:* View Class Materials - students can access and download any files hosted on the class’s

resources page

*UC-16:* LogIn - authentication for every single user that every other feature uses to identify the

students

*UC-17:* Modify Grades - professors can modify the grades of their students

*UC-18:* View Gradebook - professors and students can view grades

*UC-19:* Host Lecture - professors/TAs can host online lectures through the use of video calling and

virtual whiteboard that students can see in real time.

*UC-20:* Modify Classes - professors can control the students registered for their classes and students can

manage all the classes they are registered for

UC-21: View Hub - Users can clearly see exactly where to go for whatever they want to do on the website

## **ii. Use Case Diagram**

## 

## 

## 

## 

## 

## 

## 

## 

## 

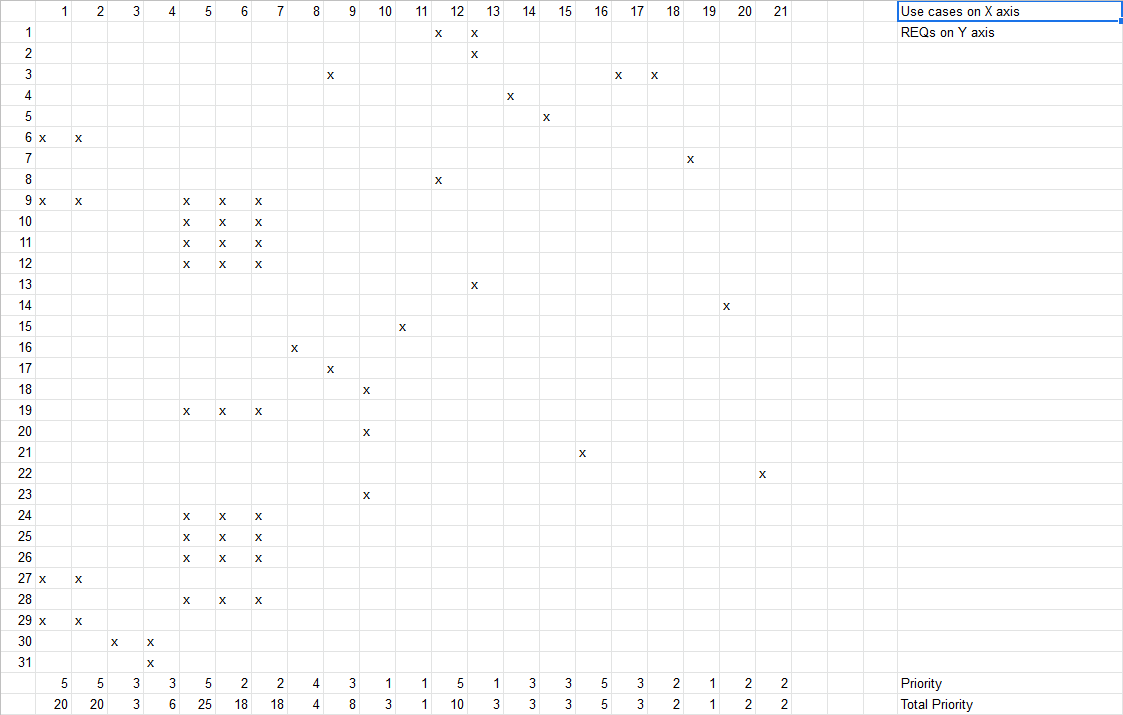
## 

## 

## 

## 

## **iii. Traceability Matrix**



## 

## 

## 

## 

## 

## 

## 

## 

## 

## 

## 

## 

## 

## 

## **iv. Fully Dressed Descriptions**

|  |  |
| --- | --- |
| **UC-1** | **Take Quiz and Analyze Results** |
| Related Requirement | REQ-6, REQ-9, REQ-28, REQ-30 |
| Initiating Actor | Student |
| Actor’s Goal | To take quizzes through the app as well as online quizzes and be able to view and analyze results |
| Participating Actors | Professor, Web App, Phone App, Database |
| Pre Conditions | The student is logged in and authenticated. The particular class is specified |
| Post Conditions | None |

Flow of Events for Main Success Scenario

|  |  |  |
| --- | --- | --- |
| -> | 1. | Student either answers a poll question in class or through an online quiz |
| <- | 2. | System checks the student response(s), and marks them as correct or incorrect. System associates each question with a certain topic. |
| -> | 3. | Through the webapp, student can view specific question to see results at any time in the future. The student can also search a particular topic in the database to see all quiz questions related to that topic. |

|  |  |
| --- | --- |
| **UC-2** | **Create and Administer Quiz/In Class Questions** |
| Related Requirement | REQ-6, REQ-9, REQ-28, REQ-30 |
| Initiating Actor | Professor or TA |
| Actor’s Goal | To create a quiz questions that the students can take in class or a full quiz that the students can take out of class. |
| Participating Actors | Students, Database, Web App |
| Pre Conditions | Professor/TA is authenticated and the particular class is specified |
| Post Conditions | If the quiz question is an in-class question, the question can be triggered at any time by the professor. If the quiz is to be taken outside of class, the quiz form persists for the specified amount of time. |

Flow of Events for Main Success Scenario

|  |  |  |
| --- | --- | --- |
| -> | 1. | Professor selects to make an out-of- class quiz administered through the web app. The professor specifies the time frame for which this quiz should be available. |
| <- | 2. | The system creates an in-progress data store for this quiz. System displays a page that allows for user inputs in quiz questions and quiz answers |
| -> | 3. | Professor can write questions and associated answers in a multiple choice format. The professor can mark questions with a topic. The professor can add or remove questions as desired. Professor saves quiz. |
| <- | 4. | System saves the quiz questions and answers. The system indicates to the professor that the quiz is in-progress. |
| -> | 5. | Professor adds more questions or removes questions from the quiz. The Professor publishes the quiz. |
| <- | 6. | System marks the quiz as published. The system sends an announcement to all students that the quiz will be available at the originally specified time. At the specified time, the quiz becomes available to take by students, leading into UC-1. |

Flow of Events for Alternate Scenario

|  |  |  |
| --- | --- | --- |
| -> | 1. | Professor selects to make an in-class quiz administered through the phone app. The professor specifies the date of the lecture for which this quiz question corresponds to. |
| <- | 2. | The system creates an in-progress data store for this quiz. System displays a page that allows for user inputs in quiz questions and quiz answers |
| -> | 3. | Professor can write questions and associated answers in a multiple choice format. The professor can mark questions with a topic. The professor can add or remove questions as desired. Professor saves quiz. |
| <- | 4. | System saves the quiz questions and answers. The system indicates to the professor that the quiz is in-progress. |
| -> | 5. | Professor adds more questions or removes questions from the quiz. The Professor publishes the quiz. |
| <- | 6. | System marks the quiz as published. On the specified date, the quiz becomes viewable on the professor’s in-class dashboard. |
| -> | 7. | Professors triggers individual questions on the day of the lecture. |
| <- | 8. | System recognizes that question was triggered by professor. The system now accepts student responses for that particular question, leading into UC-1. |

|  |  |
| --- | --- |
| **UC-3** | **View Announcement** |
| Related Requirement | REQ-31, REQ-32 |
| Initiating Actor | Student, Professor, and TAs |
| Actor’s Goal | View announcement published by professor or TA |
| Participating Actors | Student, Professor, TAs, Database, Web App, Mobile App |
| Pre Conditions | Professor or TA previously published an announcement. Actor is logged in. |
| Post Conditions | Announcement is marked as read for actor. |

Flow of Events for Main Success Scenario

|  |  |  |
| --- | --- | --- |
| -> | 1. | Student, Professor, or TA goes to the announcement tab |
| <- | 2. | System displays all recent announcements and flags unread ones |
| -> | 3. | Student, Professor, or TA open an announcement |
| <- | 4. | System flags announcement as read |

|  |  |
| --- | --- |
| **UC-4** | **Create Announcement** |
| Related Requirement | REQ-31, REQ-32 |
| Initiating Actor | Professor or TA |
| Actor’s Goal | Create announcement viewable to everyone in the class and send notifications via mobile app and or email. |
| Participating Actors | Students, Database, Web App, Mobile App |
| Pre Conditions | Professor/TA is authenticated and the particular class is specified |
| Post Conditions | Send announcements via email and display on mobile and web app. |

Flow of Events for Main Success Scenario

|  |  |  |
| --- | --- | --- |
| -> | 1. | Professor selects to make an announcement though the web app. |
| <- | 2. | The system creates an in-progress data store for the announcement. System displays a HTML text editor. |
| -> | 3. | Professor can write and format the announcement. Professor saves announcement. |
| <- | 4. | System saves the in-progress announcement. The system indicates to the professor that the announcement is being edited and saved. |
| -> | 5. | Professor continues to edit until it is ready to publish. Professor publishes announcement immediately or at a certain time. |
| <- | 6. | System marks the announcement as published. The system sends a notification to all students, TAs, and the professor about the announcement at the specified time. The announcement becomes viewable, leading into UC-3 |

Flow of Events for Alternate Scenario

|  |  |  |
| --- | --- | --- |
| -> | 1. | Professor selects to make an announcement though the web app. |
| <- | 2. | The system creates an in-progress data store for the announcement. System displays a HTML text editor. |
| -> | 3. | Professor can write and format the announcement. Professor saves announcement and professor stops editing. |
| <- | 4. | System saves the in-progress announcement. The system indicates to the professor that the announcement is being edited and saved. Announcement can be edited in the future. |

|  |  |
| --- | --- |
| **UC-5** | **Create Question/Feedback (In Class)** |
| Related Requirement | REQ-9, REQ-10, REQ-11, REQ-12, REQ - 19, REQ-25, REQ - 26, REQ - 27, REQ-29 |
| Initiating Actor | Student |
| Actor’s Goal | Create a question for the professor to answer in class |
| Participating Actors | Professor, System, Database |
| Pre Conditions | Class is in session. Student is logged in |
| Post Conditions | Students may vote on question and professor may read and answer question |

Flow of Events for Main Success Scenario

|  |  |  |
| --- | --- | --- |
| -> | 1. | Student initiates question creation. |
| <- | 2. | System opens question creation workspace with plain text box and option to include name with question. |
| -> | 3. | Student writes and edits question until satisfied. Student may choose to remain anonymous when publishing question |
| <- | 4. | System saves question on database, and publishes question to view by other students and the instructor. |

|  |  |
| --- | --- |
| **UC-6** | **Vote on Question/Feedback (In Class)** |
| Related Requirement | REQ-9, REQ-10, REQ-11, REQ-12, REQ - 19, REQ-25, REQ - 26, REQ - 27, REQ-29 |
| Initiating Actor | Student |
| Actor’s Goal | Move better questions higher on professor’s list |
| Participating Actors | Student, Professor, System, Database |
| Pre Conditions | Class is in session. Question is previously asked. Student is logged in and authenticated. |
| Post Conditions | Question priority is updated. |

Flow of Events for Main Success Scenario

|  |  |  |
| --- | --- | --- |
| -> | 1. | Student opens live question and feedback tab and votes on a question/feedback already made. |
| <- | 2. | System updates the question/feedback score and displays update for all users. |
| -> | 3. | Student can see the vote by highlight and can change or take away the vote. |

|  |  |
| --- | --- |
| **UC-7** | **Dismiss Question/Feedback (In Class)** |
| Related Requirement | REQ-9, REQ-10, REQ-11, REQ-12, REQ - 19, REQ-25, REQ - 26, REQ - 27, REQ-29 |
| Initiating Actor | Professor |
| Actor’s Goal | Address and delete question or feedback from list |
| Participating Actors | Professor, System, Database |
| Pre Conditions | Class is in session. Question is previously asked. Professor is logged in and authenticated. |
| Post Conditions | Question priority is updated. |

Flow of Events for Main Success Scenario

|  |  |  |
| --- | --- | --- |
| -> | 1. | Professor opens live question and feedback tab on web app and may address a question on the list. Professor dismisses a question. |
| <- | 2. | System deletes question/feedback, but keeps a System updates the question/feedback score and displays update for all users. |
| -> | 3. | Student can see the vote by highlight and can change or take away the vote. |

|  |  |
| --- | --- |
| **UC-8** | **Read Forum** |
| Related Requirement | REQ-1, REQ-16, REQ-8 |
| Initiating Actor | Student, Professor |
| Actor’s Goal | View past forum posts from other students and the professor |
| Participating Actors | Student, Professor, System, Database |
| Pre Conditions | Student or Professor are logged into the web page |
| Post Conditions | Forum page is updated with the new post |

Flow of Events for Main Success Scenario

|  |  |  |
| --- | --- | --- |
| -> | 1. | Professor or student opens the window to submit a new forum post and correctly submits it |
| <- | 2. | System updates the new post on the public forum and displays this new post for everyone visiting the page |
| -> | 3. | Students and the professor can view the new post and respond to it |

|  |  |
| --- | --- |
| **UC-14** | **Upload and Manage Resources** |
| Related Requirement | REQ-4, REQ-5, REQ-8, |
| Initiating Actor | Professor or TA |
| Actor’s Goal | To upload and manage class resources such as lecture slides, notes, homework, etc. |
| Participating Actors | Students, Database, Web App |
| Pre Conditions | Professor/TA is authenticated and the particular class is specified |
| Post Conditions | Uploaded data persists until deleted by the initiating actor |

Flow of Events for Main Success Scenario

|  |  |  |
| --- | --- | --- |
| -> | 1. | Professor selects documents from local file system to upload to resources application. |
| <- | 2. | The system copies the uploaded documents and populates the resources with them. |
| -> | 3. | The professor may move the resources around within the application by creating subdirectories organized in a traditional file system as seen on regular computers. |
| <- | 4. | The system keeps track of file locations to update its UI without making multiple copies of the data. |

# 

# 

# 

# 

# **4. User Interface Specification**

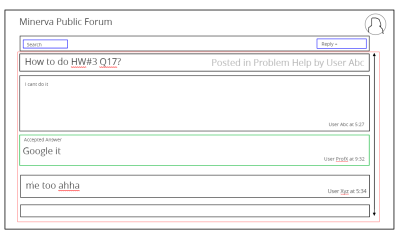
## **A. Forum**

### Main Forum Page

# 

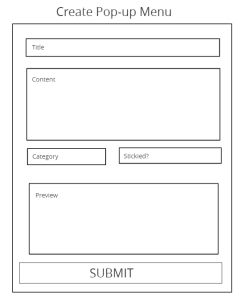
Scrollable window with different categories, with each category giving a preview of the most recent posts in that category. Bars at top to search and filter post for easier access. Button to submit a new post in top right corner

### Single Forum Thread



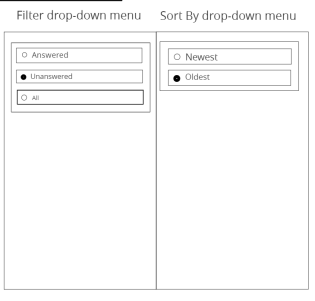
Initial post and title at top with professor selected best answer immediately after and outlined in green. All other replies are listed after. Button to create reply to thread at top right corner and that bar does not scroll with the rest of the page.

### Thread Create Pop-Up Menu



Small popup window, not redirect. Category is a drop down menu and stickied is a check box

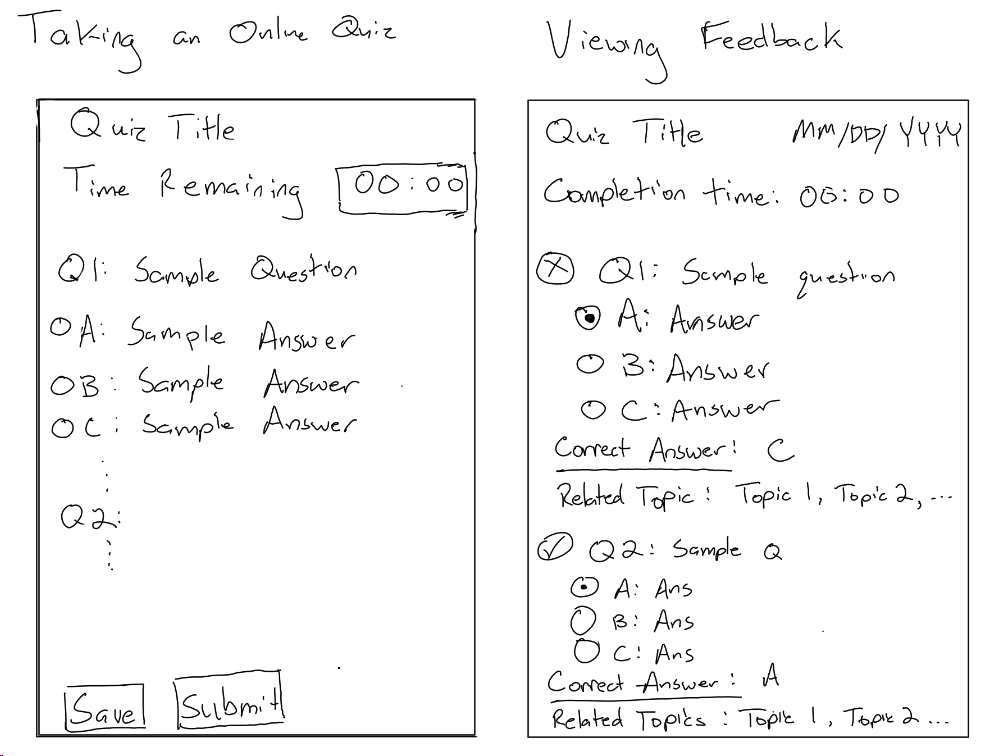
#### Filter Drop Down Menu



Small drop down windows to select what posts to view and how to sort them. The two drop downs are separate.

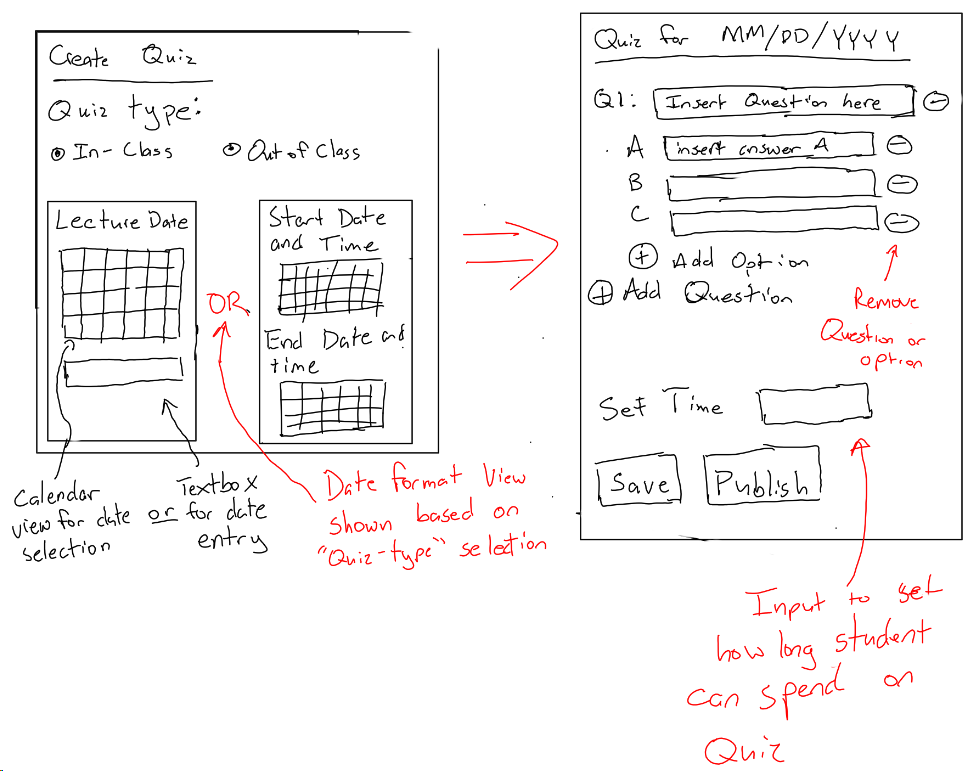
## B. Quiz

### Taking an online quiz and viewing feedback.



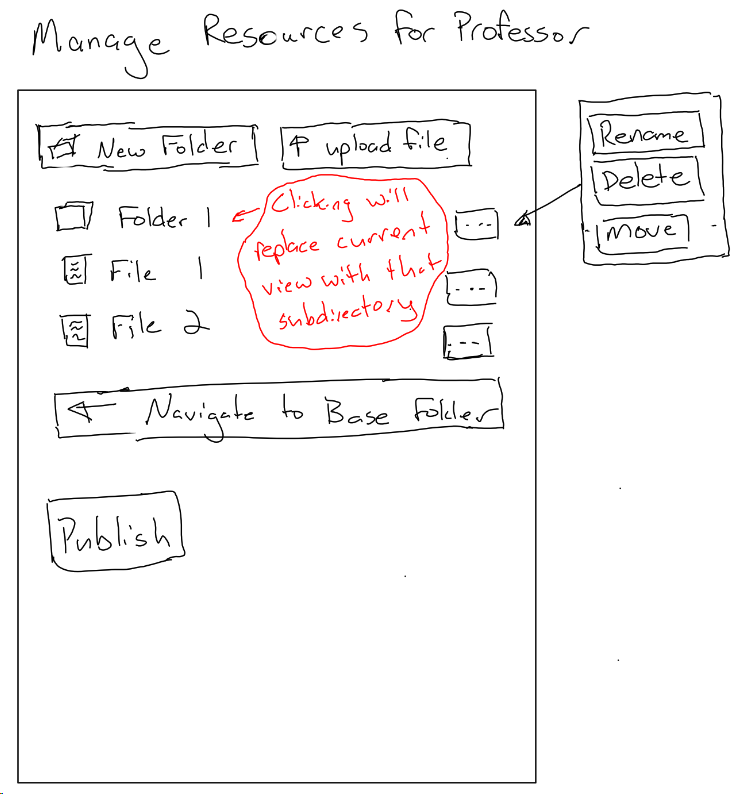
This is the student’s screen for taking online (not in class) quizzes. The student should be able to select options and save/submit the quiz during the time remaining. Selecting options will be done through the use of radio buttons as shown on the sketch above.

### Creating a quiz



This is the UI for creating quizzes as a professor/TA. For the create quiz UI, the user will select what kind of quiz they want to make using the radio buttons. Depending on what they choose, a date-option display will show that allows the user to specify the date for the quiz. Then, the user is taken to the second screen which allows them to make a quiz as described in the sketch above. This UI is designed so as to minimize the number of button presses by the user by having all appropriate options shown on the screen from the outset (such as adding/removing questions).

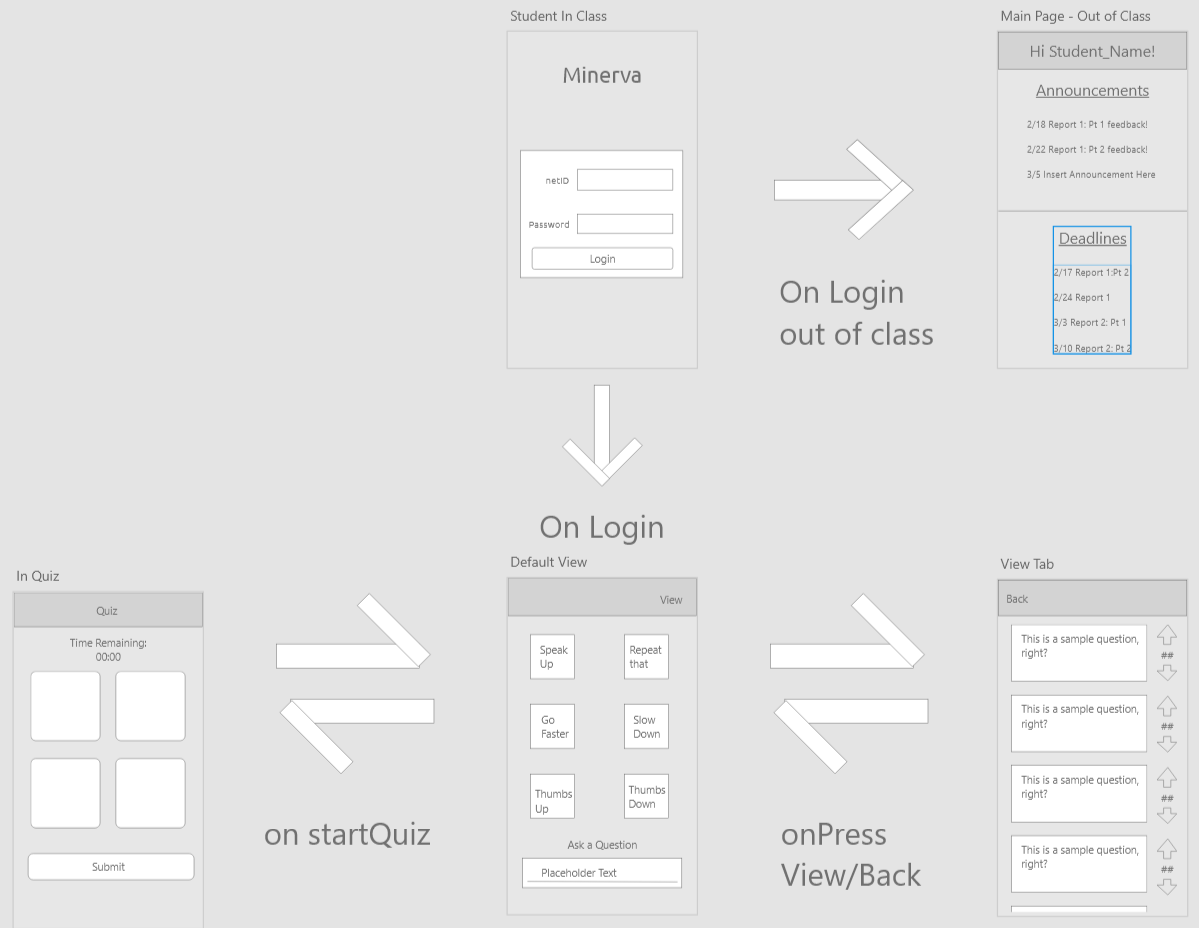
Managing resources as a professor/TA



The student resources UI is very similar to this UI, but much of the functionality is removed. The student should only be able to click on files and navigate to base folder. Clicking on a file should download the file to the user’s system. Clicking on a folder should open that subdirectory in the UI.

Mobile UI

|  |  |
| --- | --- |
|  | This screen is the login screen for the mobile student application.  The user can login with the minimal amount of touches in this setup. |
|  | This is the main screen that students have during their lecture. They can easily give the professor feedback during lecture and ask questions with very few touches.  The view button on the top right allows them to access the in-class questions asked by other students in only 1 tap. |
|  | This is the quiz page that pops up automatically when the professor starts a quiz. After choosing one of the four options and hitting submit, the student will be automatically taken back to the main screen.  The results will be shown on the screen by the professor, not on the mobile device. |
|  | This is the forum where students can ask in-class questions and upvote/downvote with only 1 tap. The back button takes the student back to the main screen. |
|  | When not in class, the students main screen will look like this. As the in-class features won’t be available, they can view announcements and deadlines. |
|  | This is a sample of what the Push Notifications would look like if the professor added an announcement or a deadline. It would contain course name + the announcement. |



This is the flowmap within the mobile application. You can see the multiple screens and how the students would easily navigate between them.

# 

# C. User Effort Estimation

The user must not require more than 3 taps to traverse between any two screens. Within each screen, the UI must be intuitive and no more than 1 or 2 taps should be used per action (excluding typing actions).

The website contains more features and more complex actions are available. To navigate between pages, there should be a max of 3 taps also. But since the actions (such as uploading files) are more complex, there should be a max 3 clicks within the site to perform the more complex actions such as opening files, uploading files etc.

From logging on to performing any 1 action (regardless of what page its on), should be a max of 5 clicks. 2 to navigate to another page and then 3 more to perform the most complicated possible task (ex: making a post, uploading a file or adding a comment to a post).

**5. Domain Analysis**

## **Domain Model**

### UC-1: Take Quiz and Analyze Results

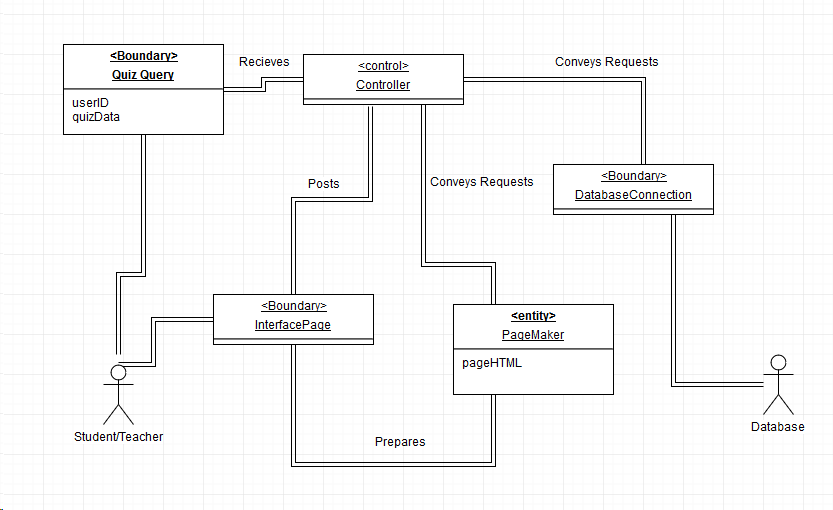
### UC-2: Create and Administer Quiz/In-Class Questions

The following concepts, responsibility descriptions, and attribute descriptions can be applied to both use cases. In practice, both use cases will have a different implementation of how the concepts will interact in regards to the specific data they will store and exchange. However, the basic concept to concept structure is kept the same between both use cases, making this design model versatile for many problems in this project.

|  |  |  |
| --- | --- | --- |
| **Responsibility Description** | **Type D/K** | **Concept Name** |
| Coordinates the actions of all concepts associated with this use case and delegate work to other concepts | D | Controller |
| HTML Document that shows current actions that can be done | K | Interface Page |
| Render the retrieved records into an HTML for the web browser to display | D | Page Maker |
| Prepare a query that matches the actor’s intended action, either for storage or retrieval | D | Database Connection |
| Storage of current information regarding the status of the quiz | K | Quiz Request |

|  |  |  |
| --- | --- | --- |
| **Concept Pair** | **Association Description** | **Association Name** |
| Controller <-> Page Maker | Controller passes request to Page Maker and receives the HTML to display | Conveys requests |
| Controller <-> Interface Page | Conveys info from “conveys requests” to the interface page to display | Control interface |
| Controller <-> Database Connection | Controller makes queries to database connection either storing or retrieving information and database sends confirmation back | Make query |
| Page Maker <-> Interface Page | Page maker makes page for interface page display | Page display |
| Interface Page <-> Quiz Request | Quiz request is updated based on user interaction with the interface page | Take quiz |
| Quiz Request <-> Controller | Controller handles quiz request and conveys to database connection | Quiz query |

|  |  |  |
| --- | --- | --- |
| **Concept** | **Attributes** | **Attribute Description** |
| Quiz Query | User Identity | Used to determine the actor’s credentials which determines what data the actor can access as well as what data they can submit |
|  | Quiz data | Quiz answers data submitted or viewed by the actor for communication with data storage. Includes any other relevant quiz data |
| Page maker | Page html | Actual HTML to display on the page |



### UC-3: View Announcement

### UC-4: Create Announcement

The Domain of creating and viewing announcement is similar to that of creating a quiz. In both cases, the user will be verified based on ID if the user can create an announcement. Announcements and quizzes will be edited on an HTML textbox, but the announcements will have less functionality as it does not require input.

### UC-5: Create Question/Feedback

### UC-6: Vote on Question

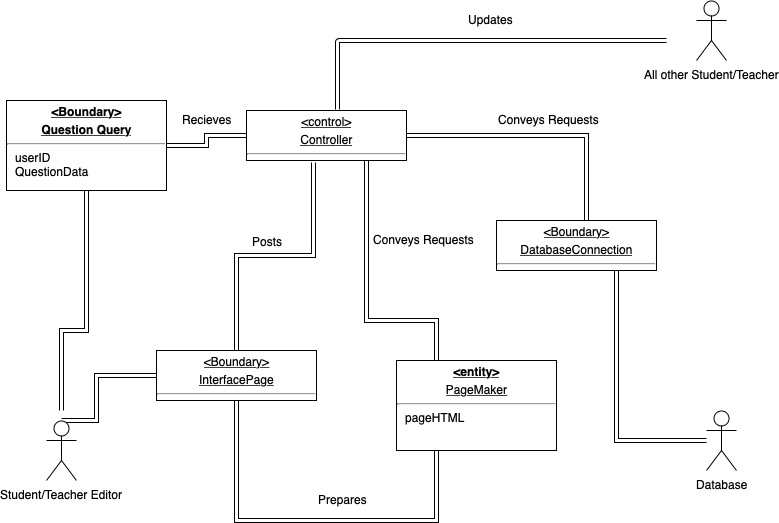
### UC-7: Dismiss Question/Feedback

The process of creating, voting, and dismissing a question happens with a similar structure to quizzes and announcements. Questions and Feedback are updated live for all users on the page whenever a vote, creation, or dismissal happens from a user. The capabilities of the user are determined by the userID. If this is not feasible, a simple prompt will appear on the bottom of the poll indicating the user to refresh the page to see updates. \*The Domain will vary between the webapp and the mobile app.

|  |  |  |
| --- | --- | --- |
| **Responsibility Description** | **Type D/K** | **Concept Name** |
| Coordinates the actions of all concepts associated with this use case and delegate work to other concepts | D | Controller |
| HTML Document that shows the current Question/Feedback list. Includes buttons to create and vote. | K | Interface Page |
| Render the page into an HTML for the web browser to display for the users. | D | Page Maker |
| Prepare a query that matches the actor’s intended action, either for storage, update, or retrieval. | D | Database Connection |
| Storage of current information regarding the status of the question. Can be votes, dismissal, or creation. | K | Question Request |

|  |  |  |
| --- | --- | --- |
| **Concept Pair** | **Association Description** | **Association Name** |
| Controller <-> Page Maker | Controller passes request to Page Maker and receives the HTML to display | Conveys requests |
| Controller <-> Interface Page | Conveys info from “conveys requests” to the interface page to display. | Control interface |
| Controller <-> Database Connection | Controller makes queries to database connection either storing, retrieving, or updating information. Database sends data on question back. | Make query |
| Page Maker <-> Interface Page | Page maker makes page for interface page display. Updates whenever a command is sent that the database has been updated (Live updates). | Page display |
| Interface Page <-> Announcement Request | Editing user interaction with the interface page. Can be creation of question/feedback or vote. In the case of the teacher, a dismissal. | Publish Announcement |
| Question Request <-> Controller | Controller handles question request and conveys to database connection. Request does not go directly to the database. | Announcement query |
| Database Connection <-> Controller | Send command to update question list for all users when update is made for live feedback. | Update request |

|  |  |  |
| --- | --- | --- |
| **Concept** | **Attributes** | **Attribute Description** |
| Question Query | User Identity | Used to determine the actor’s credentials which determines what data the actor can access as well as what data they can submit |
|  | Question data | Announcement answers data submitted or viewed by the actor for communication with data storage.  Includes Question, Votes, and Creator ( can be anonymous) |
| Page maker | Page html | Actual HTML to display on the page, includes HTML textbox |



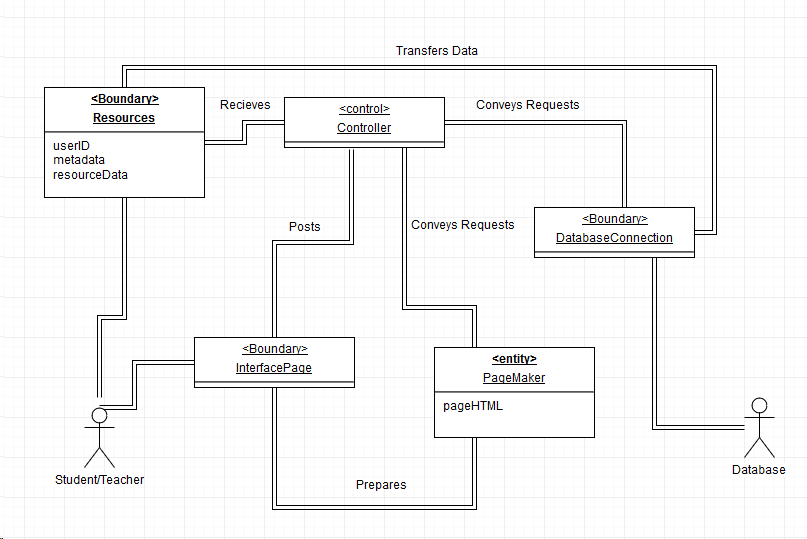
### UC-14: Upload Class Resources

These descriptions are similar to the descriptions for other use cases. They are modified to accommodate the use of “resources” which refers to files that an actor may want to access or manage within the software. This is a type of data that will be stored and transferred through the use of other concepts.

|  |  |  |
| --- | --- | --- |
| **Responsibility Description** | **Type D/K** | **Concept Name** |
| Coordinates the actions of all concepts associated with this use case and delegate work to other concepts | D | Controller |
| HTML Document that shows current actions that can be done | K | Interface Page |
| Render the retrieved records into an HTML for the web browser to display | D | Page Maker |
| Prepare a query that matches the actor’s intended action, either for storage or retrieval | D | Database Connection |
| Resources that can be accessed, uploaded, and managed | K | Resources |

|  |  |  |
| --- | --- | --- |
| **Concept Pair** | **Association Description** | **Association Name** |
| Controller <-> Page Maker | Controller passes request to Page Maker and receives the HTML to display | Conveys requests |
| Controller <-> Interface Page | Conveys info from “conveys requests” to the interface page to display | Control interface |
| Controller <-> Database Connection | Controller makes queries to database connection either storing or retrieving information and database sends confirmation back | Make query |
| Page Maker <-> Interface Page | Page maker makes page for interface page display | Page display |
| Interface Page <-> Database Connection | Interface interactions trigger database requests via the controller. | Interact Resource |
| Resources <-> Database Connection | Resources are stored, conveyed, and managed through the database connection | Access Resource |

|  |  |  |
| --- | --- | --- |
| **Concept** | **Attributes** | **Attribute Description** |
| Resources | User Identity | Used to determine the actor’s credentials which determines what data the actor can access as well as what data they can submit |
|  | metadata | Metadata information of the resource such as upload date, file type, etc |
|  | data | The resource data itself |
| Page maker | Page html | Actual HTML to display on the page |



## **b. System Operation Contracts**

### UC-1: Take Quiz and Analyze Results

* Precondition: The student is logged in and authenticated. The particular class is specified
* Postcondition: Results are stored and accessible by relevant actors.

### UC-2: Create and Administer Quiz/In-Class Questions

* Precondition: Professor/TA is authenticated and the particular class is specified
* Postcondition: If the quiz question is an in-class question, the question can be triggered at any time by the professor. If the quiz is to be taken outside of class, the quiz form persists for the specified amount of time.

### UC-3: View Announcement

* Precondition: Professor or TA previously published an announcement. Actor is logged in
* Postcondition: Announcement is marked as read for actor.

### UC-4: Create Announcement

* Precondition: Professor/TA is authenticated and the particular class is specified
* Postcondition: Send announcements via email and display on mobile and web app.

### UC-5: Create Question/Feedback

* Precondition: Class is in session. Student is logged in
* Postcondition: Students may vote on question and professor may read and answer question

### UC-6: Vote on Question

* Precondition: Class is in session. Question is previously asked. Student is logged in and authenticated.
* Postcondition: Question priority is updated.

### UC-7: Dismiss Question/Feedback

* Precondition: Class is in session. Question is previously asked. Professor is logged in and authenticated.
* Postcondition: Question priority is updated and is dismissed from screen.

### UC-8: Read Forum

* Precondition: Student or Professor are logged into the web page.
* Postcondition: Forum page is updated with the new post and comments.

### UC-14: Upload Class Resources

* Precondition: Professor is logged in and authenticated, and has prepared material to share with class
* Postcondition: Class material is made available for everyone to view and download.

## **c. Mathematical Model**

For the in-class questions, the algorithm to determine if the question will be displayed on the professors screen will be determined by Number of upvotes within a certain time frame.

Considerations:

* It is understood that it will take a few moments for the student to come up with a question after the material is discussed and to type it.
* There will also be some time for students to view the said question and decide whether or not they want to vote on it.
* However, having a question pop-up about a topic that was discussed a while back in lecture would inhibit progress in the lecture and slow it down.

Due to this, there will be a time limit of 150 seconds after a question is asked. If the question reaches a threshold of 10% upvotes within this time, then the Professor will get a notification on their screen in order to answer the question. So for example, if the class size is 200, and a Student asks a question through the app, there is a 150 second internal timer within which the question must get 20 more upvotes than downvotes for it to be sent to the professor in-class as a notification.

# **6. Project Size Estimation**

We use the following formula to calculate the Use Case Points for our project:

i. The Unadjusted Use Case Weight (UUCW) is calculated from the sum of total priorities of

each of our use cases, which can be found in the traceability matrix from section 3iii above.

ii. The Unadjusted Actor Weight is based on the complexity of all the relevant actors for our

use cases.

|  |  |  |
| --- | --- | --- |
| Actor | Complexity | Weight |
| Student | Complex | 3 |
| Professor | Complex | 3 |
| Web Application | Average | 2 |
| Mobile Application | Average | 2 |

iii. The Technical Complexity Factor (TCF) is calculated based on the weights and perceived

complexity of the following factors:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Technical Factor | Description | Weight | Perceived Complexity | Calculated Factor |
| T1 | In-class features must update in real time with a reasonable response time | 1 | 2 | 2 |
| T2 | Application must be an efficient replacement to already existent software | 1 | 3 | 3 |
| T3 | Processing complexity is fairly minimal | 1 | 1 | 1 |
| T4 | There is very little code reusability, as this is not a continuation of an existing/previous project | 1 | 4 | 4 |
| T5 | Our project will exist on multiple platforms (mobile and web applications) | 2 | 2 | 4 |
| T6 | Ease of use is very important | 0.5 | 5 | 2.5 |
| T7 | Security is not a major concern | 1 | 1 | 1 |
| T8 | We expect minimal system maintenance | 1 | 1 | 1 |
|  | | | Total: | 18.5 |

We use the formula to calculate our TCF value of 0.785.

iv. Environment Complexity Factor (ECF) is based on the relevant skills of our team members

and how they cater to the needs of our project.

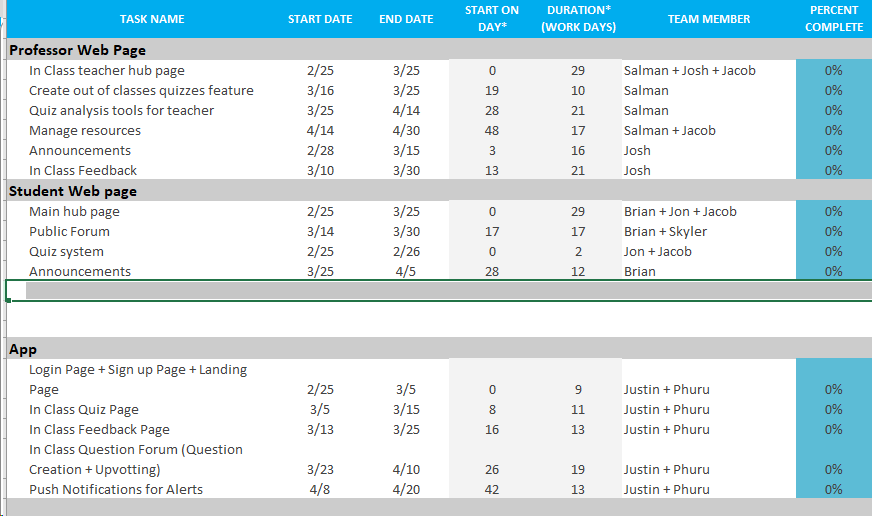
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Environmental Factor | Description | Weight | Perceived Impact | Calculated Factor |
| E1 | Not all team members have familiarity with development processes used in our project | 1.5 | 3 | 4.5 |
| E2 | Not all team members have familiarity with applications used in our project | 0.5 | 3 | 1.5 |
| E3 | Our team generally has a solid foundation with object-oriented programming, but not at an expert/professional level | 1 | 4 | 4 |
| E4 | Our team is generally motivated about the problem our program plans to address | 1 | 2 | 2 |
| E5 | Almost all team members are familiar with the programming languages they will be using | 1 | 4 | 4 |
|  | | | Total: | 16 |

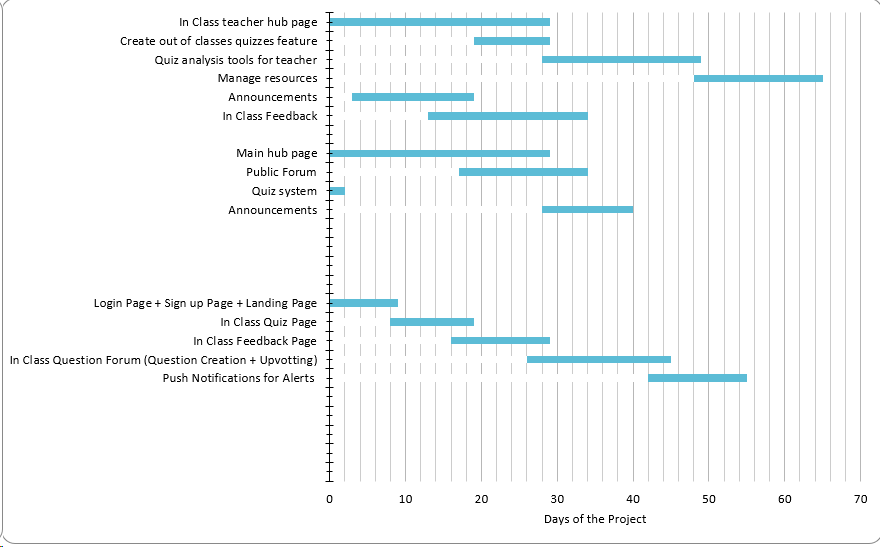
We us the formula to calculate our ECF value of 0.92

Finally, we calculate the final UCP value 123 Use Case Points.

# **7. Plan of Work**

The Gantt chart for the plan of work is shown in the graphics below.

****

****

**8.Resources**

Anthony P Carnevale, Nicole Smith, and Jeff Strohl, "Recovery: Job Growth and Education

Requirements Through 2020," *Georgetown Public Policy Institute*, last accessed 10 February

2019,https://cew.georgetown.edu/wp-content/uploads/2014/11/Recovery2020.ES\_.Web\_.pdf.

Monks, J. & Schmidt, R. (2010). The impact of class size and number of students on outcomes in

higher education[Electronic version]. Retrieved 10 February 2019, from Cornell University,

School of Industrial and Labor Relations site: http://digitalcommons.ilr.cornell.edu/workingpapers/114/

Undergraduate Enrollment,” *National Center for Education Statistics*, las accessed 10 February 2019,

https://nces.ed.gov/programs/coe/indicator\_cha.asp

“Monstrous class sizes unavoidable at colleges,” *NBC news,* last modified 24 November 2007,

<http://www.nbcnews.com/id/21951104/ns/us_news-education/t/monstrous-class-sizes-unav>

idable-colleges/

David Rosengrant, Doug Hearrington, Kerriann Alvarado, and Danielle Keeble, “Following Student

Gaze Patterns in Physical Science Lectures,” *University of Kennesaw*, last accessed 10 February

2019, https://www.usnews.com/pubfiles/PERC2011\_Rosengrant.pdf.

“The Evolution of Technology in the Classroom.” *Purdue University Online*, 1 Aug. 2017,

online.purdue.edu/ldt/learning-design-technology/resources/evolution-technology-classroom.

1. Anthony P Carnevale, Nicole Smith, and Jeff Strohl, "Recovery: Job Growth and Education Requirements Through 2020," *Georgetown Public Policy Institute*, last accessed 10 February 2019, https://cew.georgetown.edu/wp-content/uploads/2014/11/Recovery2020.ES\_.Web\_.pdf. [↑](#footnote-ref-0)
2. Monks, J. & Schmidt, R. (2010). The impact of class size and number of students on outcomes in higher education[Electronic version]. Retrieved 10 February 2019, from Cornell University, School of Industrial and Labor Relations site: http://digitalcommons.ilr.cornell.edu/workingpapers/114/ [↑](#footnote-ref-1)
3. Ibid. [↑](#footnote-ref-2)
4. “Undergraduate Enrollment,” *National Center for Education Statistics*, las accessed 10 February 2019, https://nces.ed.gov/programs/coe/indicator\_cha.asp [↑](#footnote-ref-3)
5. “Monstrous class sizes unavoidable at colleges,” *NBC news,* last modified 24 November 2007, http://www.nbcnews.com/id/21951104/ns/us\_news-education/t/monstrous-class-sizes-unavoidable-colleges/ [↑](#footnote-ref-4)
6. David Rosengrant, Doug Hearrington, Kerriann Alvarado, and Danielle Keeble, “Following Student Gaze Patterns in Physical Science Lectures,” *University of Kennesaw*, last accessed 10 February 2019, https://www.usnews.com/pubfiles/PERC2011\_Rosengrant.pdf. [↑](#footnote-ref-5)
7. Ibid. [↑](#footnote-ref-6)
8. Ibid. [↑](#footnote-ref-7)
9. Ibid. [↑](#footnote-ref-8)
10. “The Evolution of Technology in the Classroom.” *Purdue University Online*, 1 Aug. 2017, online.purdue.edu/ldt/learning-design-technology/resources/evolution-technology-classroom. [↑](#footnote-ref-9)
11. Ibid. [↑](#footnote-ref-10)