Software Implementation and Testing Report

April 26th, 2016

Daniel Baggot - Lead Designer/User Interface Design
Matthew Canton - Database Management
Darryl Papke - Testing/Support
Daniel Silveri - Quality Assurance

University of North Carolina Wilmington

Abstract. The contents of this document show the final design process that went into our group project. The paper contains updated diagrams that have changed throughout the project's lifecycle. Also contained is a breakdown of the testing process that occurred to make sure the website is working according to ours and the client's needs. Along with this, included is a user manual that details everything needed for a new user to successfully navigate the website without any unnecessary hassle in having to learn through experimentation.

Keywords: Teacher User, Student User, Native Language, Login, Logout, MCU Server, Dashboard, Video Prompt, Response Video, Text Message, Video Chat, Video Practice Set, Edit User Info, Section

1 Introduction

Japanese professor Yoko Kano needs a web based software that allows Japanese English students and English Japanese students to communicate with each other. The system must support user logins through which the opposite native speakers can leave video audio and text messages for each other and respond. Ms. Kano also wants a quizzing/assessment component which records student responses to pre recorded video prompts by Ms. Kano. Student recordings and assessment responses as well as statistics about student usage, such as number of attempts, login dates and total talk time must be visible to teacher users. The site must be visually attractive. Other possible future features that may be added to take into account during design of the site include the addition of optional Japanese interface text, the addition of student to student video chat (must be recordable for teacher).

1.1 Ethical Issues

The most important ethical issue that will be handled is keeping client information secure and not sharing or using any information about clients for personal gain. Clients will be informed about the software's information practices and information will not be shared without their consent first. Along the lines of our software affecting a user's personal life is unlikely as recordings will be reviewed by the teacher for any misconduct by students and features will be added to keep chat rooms secure and only available for up to two students.

1.2 Legal Issues

Our project doesn't have many legal worries. The one area that needs to be looked out for is a user misusing the video chat system and displaying inappropriate images/objects to other students in the streams. As of now there is no protective measure in place to guard against such behavior. One safeguard that could be implemented, is the ability for study to flag other users as being inappropriate. When this occurs a video of the session could be sent to an admin, who then could ban the user and remove their account from the website.

1.3 Security Issues

Security issues will be handled in multiple ways on the site through various teachers viewing student's usage of the software and other students reporting misconduct. The software will also prevent the insertion of unwanted information or submissions, which can be monitored by teachers as they review submissions by students for grading. User information is safely secured to avoid any possibility of an outside threat easily obtaining it.

1.4 Societal Impact

The only societal impact that our software should have, would be affecting the learning ability of the students that decide to use the software. Our software should provide a positive impact on a student's ability to learn and master the Japanese language. The website will make the accessibility of lessons easier providing a convenient atmosphere for learning a second language while preventing harmful ethical and security issues.

1.5 Paper Outline

Section 2 will contain all of the updated models from the design phase through the implementation phase. This will contain the updated cost analysis, along with the original and updated risk analysis, scheduling, and requirements analysis.

Section 3 will contain the updated design models from the design phase, along with supporting text detailing the information contained in the diagrams.

Section 4 will be a breakdown of the implementation process. This will include a breakdown of the program languages used and their implementation. There will a breakdown of the functions included in the final implementation of the project, along with any functions that were not included. Also included will be a list of test cases and their results. Finally included will be a breakdown of the project, such as; problems encountered throughout the project, non-functional requirements, and lessons learned by the group.

2 Project Plan and Requirements Models

This section will contain an updated version of the LOC and cost estimations for the project. Along with updated risk analysis, scheduling, and requirements/analysis models. Each section will display the old models in the beginning of the section followed by the newly updated models.

2.1 Cost Estimation

Cost estimation from the requirements engineering.

Software Development (Elaboration and Construction)

Effort = 6.7 Person-months

Schedule = 6.9 Months

Cost = $50548

Total Equivalent Size = 1200 SLOC

Acquisition Phase Distribution

Phase	Effort (Person-months)	Schedule (Months)	Average Staff	Cost (Dollars)
Inception	0.4	0.9	0.5	$3033

Elaboration	1.6	2.6	0.6	$12132
Construction	5.1	4.3	1.2	$38417
Transition	0.8	0.9	0.9	$6066

Table 1 [3]*: first section of the requirements engineering phase COCOMO II Model*

Software Effort Distribution for RUP/MBASE (Person-Months)

Phase/Activity	Inception	laboration	Construction	Transition
Management	0.1	0.2	0.5	0.1
Environment/CM	0.0	0.1	0.3	0.0
Requirements	0.2	0.3	0.4	0.0
Design	0.1	0.6	0.8	0.0
Implementation	0.0	0.2	1.7	0.2
Assessment	0.0	0.2	1.2	0.2
Deployment	0.0	0.0	0.2	0.2

Table 2 [3]*:second section of the requirements engineering phase COCOMO II Model*

Table 1 breaks down the cost in dollars, required to complete the project. The cost is then further broken down to shown the cost of each stage of development. These are the best assumptions our group can make at this time as to how long the project will take, along with how many lines of code will be needed.

Table 2 breaks down all the stages of development into the required person months in order to complete the project.

Cost estimation from the design phase.

Software Development (Elaboration and Construction)
Effort = 3.3 Person-months
Schedule = 5.5 Months
Cost = $10625

Acquisition Phase Distribution

Phase	Effort (Person-months)	Schedule (Months)	Average Staff	Cost (Dollars)
Inception	0.2	0.7	0.3	$638

Elaboration	0.8	2.0	0.4	$2550
Construction	2.5	3.4	0.7	$8075
Transition	0.4	0.7	0.6	$1275

Table 3 [3]: *first section of the design phase COCOMO II Model*

Software Effort Distribution for RUP/MBASE (Person-Months)

Phase/Activity	Inception	Elaboration	Construction	Transition
Management	0.0	0.1	0.3	0.1
Environment/CM	0.0	0.1	0.1	0.0
Requirements	0.1	0.1	0.2	0.0
Design	0.0	0.3	0.4	0.0
Implementation	0.0	0.1	0.9	0.1
Assessment	0.0	0.1	0.6	0.1
Deployment	0.0	0.0	0.1	0.1

Table 4 [3]:*second section of the design phase COCOMO II Model*

Table 3, gives the breakdown of cost based off of the revised Lines of Code (LOC), along with revised person months required to complete the project. The cost of development is expected to decrease after further refining the number of LOC required.

Table 4 gives an updated listing of the person months required to complete the project. The expected times changed due to a better understanding of the project requirements after having started development.

Cost estimation from the implementation phase.

Effort = 8.1 Person-months
Schedule = 7.3 Months
Cost = $40669

Total Equivalent Size = 2500 SLOC

Acquisition Phase Distribution

Phase	Effort (Person-months)	Schedule (Months)	Average Staff	Cost (Dollars)
Inception	0.5	0.9	0.5	$2440
Elaboration	2.0	2.7	0.7	$9761

Construction	6.2	4.6	1.3	$30909
Transition	1.0	0.9	1.1	$4880

Table 5 [3]:*first section of the design phase COCOMO II Model*

Software Effort Distribution for RUP/MBASE (Person-Months)

Phase/Activity	Inception	Elaboration	Construction	Transition
Management	0.1	0.2	0.6	0.1
Environment/CM	0.0	0.2	0.3	0.0
Requirements	0.2	0.4	0.5	0.0
Design	0.1	0.7	1.0	0.0
Implementation	0.0	0.3	2.1	0.2
Assessment	0.0	0.2	1.5	0.2
Deployment	0.0	0.1	0.2	0.3

Table 6 [3]:*second section of the design phase COCOMO II Model*

Table 5 consists of the final estimations that were gathered for the implementation phase. After running into some technical hurdles in trying to figure out solutions for the project, the LOC have increased to meet the new goals.

Table 6 is the final estimation in person hours calculated during the implementation phase of the project. After working on the project for a certain time now, our group believes this to be the more accurate assumption of how long all the stages of the development will take.

2.2 Risk Analysis

Original Risk Analysis.

Risks	**Category**	**Probability**	**Impact**	**RMMM**
Group Member Loss	ST	10%	1	Mitigation: Keep all information related to the project open among team members to prevent loss of important details. Management: Redistribute responsibilities.
Client Not Satisfied	CL	40%	2	Mitigation: Give client updates on the project often. Management: Fix any issues that have been brought up by the client.

Computer Breaks	TE	15%	2	Mitigation: Keep important files on multiple machines. Management: Team member will buy new computer or use a school computer.
Webcams Will Not Be Available for Testing	TE	20%	2	Mitigation: Find a reliable source that has webcams available or computers already equipped with webcams.
Project Size	PS	40%	3	Mitigation: Discuss features that may be out of reach given our time constraint. Management: Cut features that would not be delivered to lessen upcoming work.
Unable to Find Video Solution/ Save Solution	TE	25%	2	Mitigation: Find solution early for saving/ video needs. Management: Meet with client for further instruction on needs

PS - Project Size Risk   BU - Business Risk   CL - Client Risk   TE - Technology risk

ST - Staff Risk

Impact Values: 1 – Catastrophic   2 – Critical   3 – Marginal   4 – Negligible

Table 7: *Risk Plan*

Table 7 is the original risk analysis that our group came up with when starting the project. The biggest concerns going into the project was starting a project that was to large and ambitious for our group to finish in the given amount of time. We also calculated that there could be a high chance of the client not being satisfied with our design, as well as, the project size being too big.

Updated Risk Analysis

Risks	**Category**	**Probability**	**Impact**	**RMMM**
Group Member Loss	ST	1%	4	Mitigation: Keep all information related to the project open among team members to prevent loss of important details. Management: Redistribute responsibilities.
Client Not Satisfied	CL	10%	2	Mitigation: Give client updates on the project often. Management: Fix any issues that have been brought up by the client.
Computer Breaks	TE	15%	2	Mitigation: Keep important files on multiple machines.

				Management: Team member will buy new computer or use a school computer.
Webcams Will Not Be Available for Testing	TE	0%	4	Mitigation: Find a reliable source that has webcams available or computers already equipped with webcams.
Project Size	PS	40%	3	Mitigation: Discuss features that may be out of reach given our time constraint. Management: Cut features that would not be delivered to lessen upcoming work.
Unable to Find Video Solution/ Save Solution	TE	25%	2	Mitigation: Find solution early for saving/video needs. Management: Meet with client for further instruction on needs.
Learning New Programming Languages	TE	25%	3	Mitigation: Rely on team members expertise for certain parts of the project.
Software Has Low Usability	BU	20%	2	Mitigation: Incorporate user centric testing to find what possible users dislike.

PS = Project Size Risk BU = Business Risk CL = Client Risk TE = Technology risk ST = Staff Risk

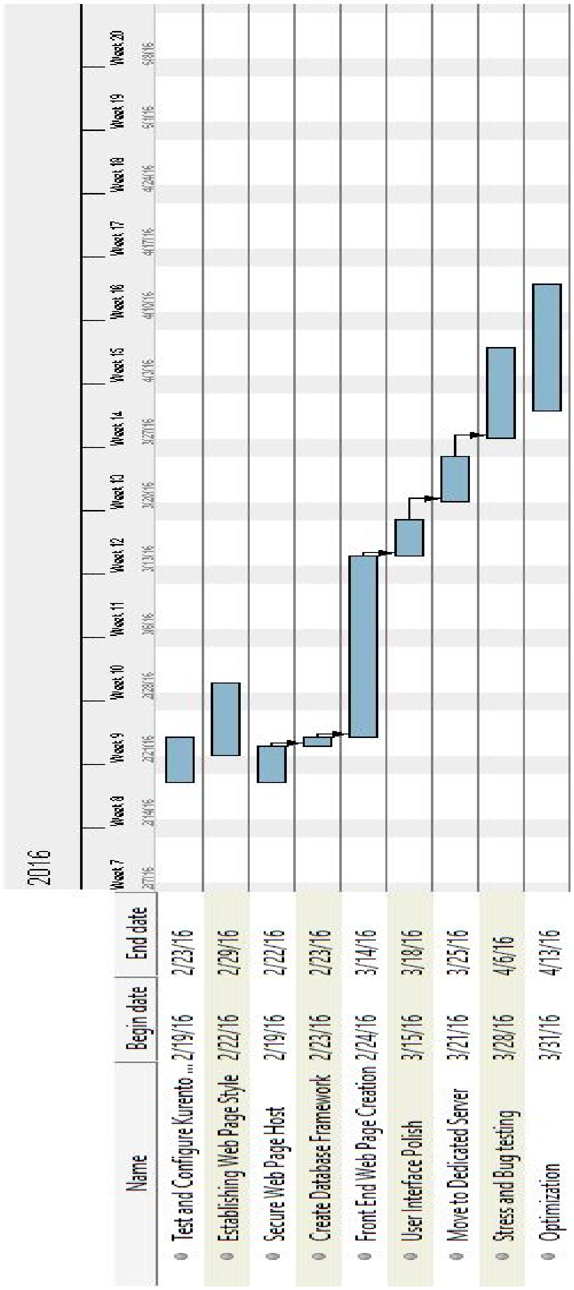
Impact Values: 1 – Catastrophic 2 – Critical 3 – Marginal 4 – Negligible

Table 8: *Risk Plan*

Table 8 is the updated risk analysis that our group came up with when finishing the project. All of the risk remained possible throughout the development of the software. Although some risk possibilities were reduced greatly as the development moved forward, including, team member loss and webcams not being available for testing. Also a couple risks were added due to later testing of usability of the software and understanding the necessity to learn new languages to complete the project.

2.3 Updated Schedule

Original GANTT Chart Schedule

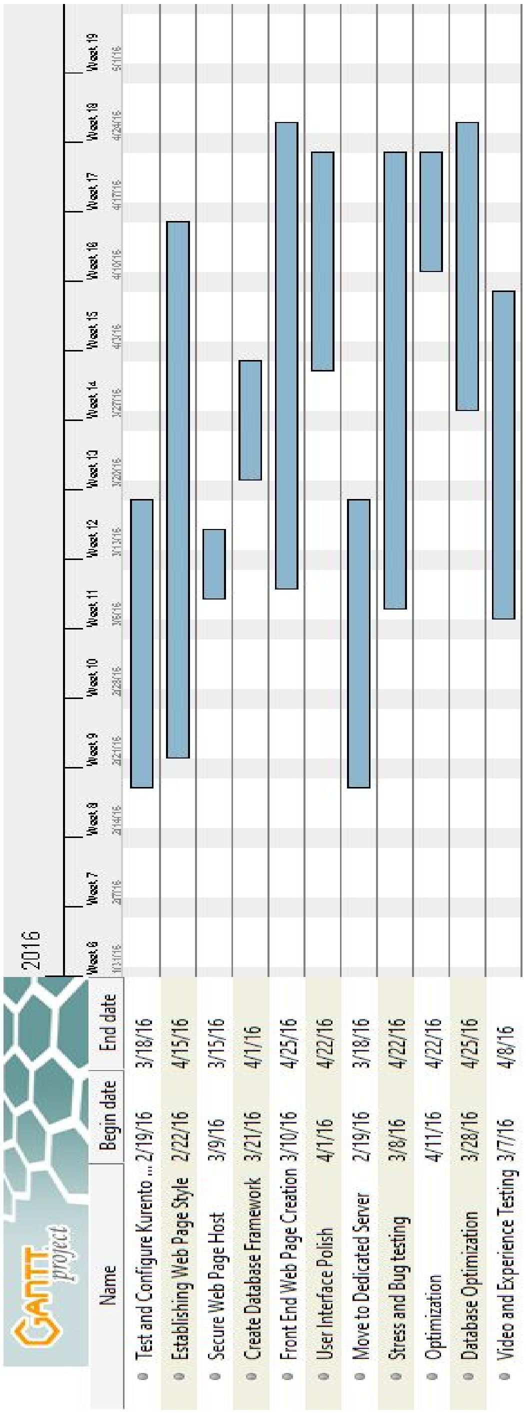


Name	Begin date	End date
Test and Configure Kurento ...	2/19/16	2/23/16
Establishing Web Page Style	2/22/16	2/29/16
Secure Web Page Host	2/19/16	2/22/16
Create Database Framework	2/23/16	2/23/16
Front End Web Page Creation	2/24/16	3/14/16
User Interface Polish	3/15/16	3/18/16
Move to Dedicated Server	3/21/16	3/25/16
Stress and Bug testing	3/28/16	4/6/16
Optimization	3/31/16	4/13/16

Figure 1: *Project Schedule*

Figure 1 is a detailed breakdown of the work that needs to be done along with a timeline of when each part needs to be completed. Most of the parts of the project can't be completed until the previous project piece has been finished. This project flow requires that each piece stay on schedule so as to not push the project deadline back, as well as, to keep the project moving forward smoothly.

Updated GANNT Chart Schedule



Name	Begin date	End date
Test and Configure Kurento ...	2/19/16	3/18/16
Establishing Web Page Style	2/22/16	4/15/16
Secure Web Page Host	3/9/16	3/15/16
Create Database Framework	3/21/16	4/1/16
Front End Web Page Creation	3/10/16	4/25/16
User Interface Polish	4/1/16	4/22/16
Move to Dedicated Server	2/19/16	3/18/16
Stress and Bug testing	3/8/16	4/22/16
Optimization	4/11/16	4/22/16
Database Optimization	3/28/16	4/25/16
Video and Experience Testing	3/7/16	4/8/16

Figure 2: *Updated GANNT Chart*

Figure 2 consists of the updated work schedule to include the setbacks that were incurred while to learn new programming languages and research that went into figuring how to build the features of the website. Some time was made up by paralleling work flow so as to not have to wait on previous sections to be developed.

2.4 Updated Requirements/Analysis Models

Use Case Name: Create New Assignment

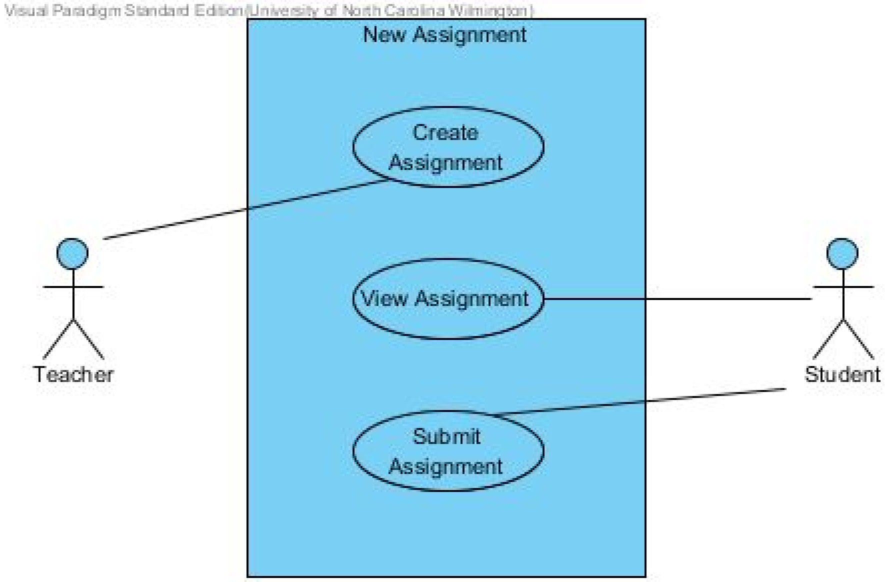


Figure 3: *Use case diagram for creating an assignment*

Stakeholder Interest:

- Teachers: Need to upload new assignments for students to complete.
- Students: Receive new assignment instructions for completion.

Main Scenario:

1. Teacher logs on to account
2. Teacher chooses the upload assignment option
3. Teacher enters specifications for the new assignment
4. Teacher makes the assignment available to students
5. Teacher Logs out
6. Student views the new assignment
7. Student completes and submits assignment for grading.

Exceptions:

1. Teacher Removes Assignment
    a. Teacher decides the assignment is unnecessary
    b. Students are notified of the cancellation
2. Teacher Attempts to Upload an Unsupported file
    a. Teacher will be notified the file was not accepted
    b. Teacher can recreate the assignment

Non-Functional Requirements

1. Student should be able to view the assignment
2. Teacher should be able to remove assignment

Use Case Name: Grade Assignment

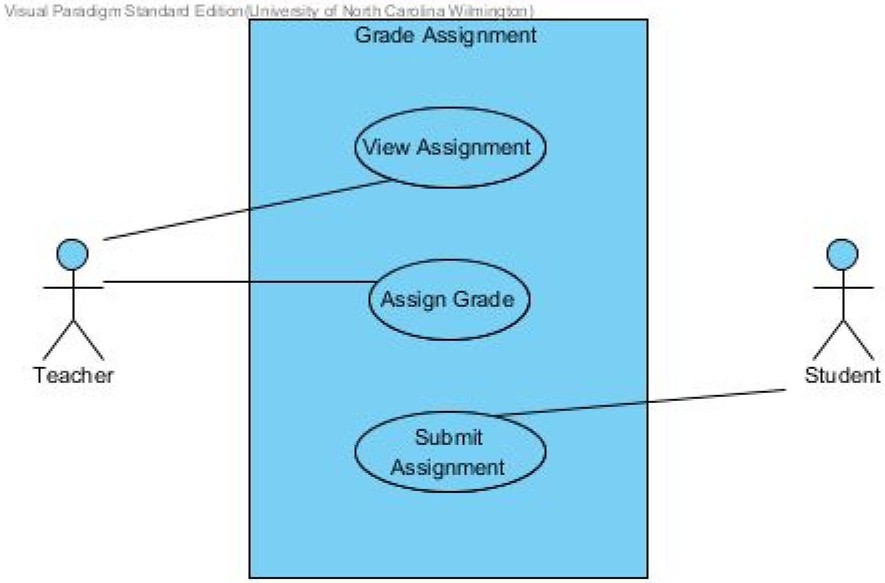


Figure 4: *Use case diagram for grading an assignment*

Stakeholder Interest

- Teacher: Need to view and grade assignment submissions by students
- Student: Need to submit assignment for grading by the teacher

Main Scenario

1. Students completes and submits an assignment
2. Teacher opens the submitted assignment
3. Teacher reviews the body of work by the student

4. Teacher assigns grade
5. Teacher makes the grade viewable to the student

Exceptions

1. The student's submission is empty/wrong assignment
    a. The teacher can inform the student of the mistake
    b. Grade assignment after resubmission

Non-Functional Requirements

1. Teacher should be able to view student submissions
2. Teacher should be able to assign grade to student
3. Student should be able to submit assignment

Use Case Name: Chat Room

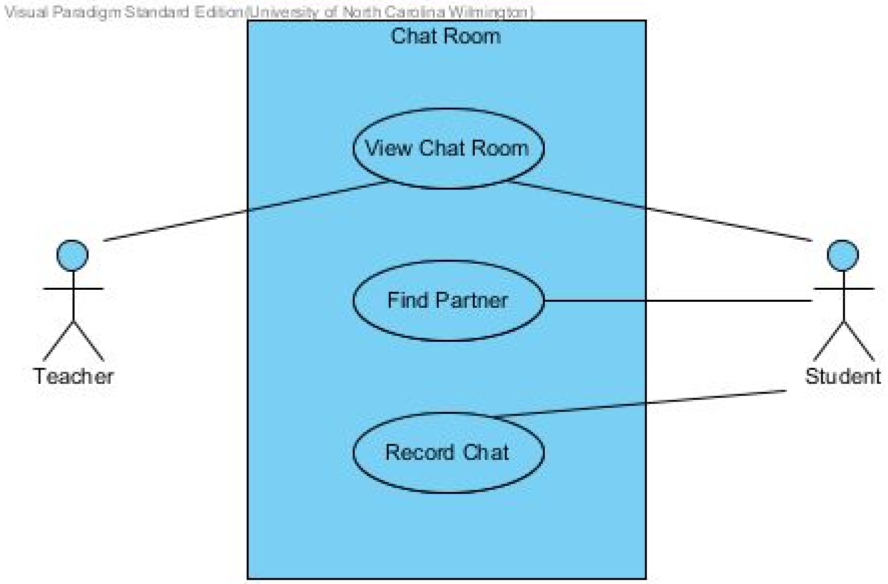

Visual Paradigm Standard Edition(University of North Carolina Wilmington)
Chat Room
View Chat Room
Find Partner
Record Chat
Teacher
Student

Figure 5: *Use case diagram for use of chat room*

Stakeholder Interest

- Student: Need to use chat room to find partner and record chat
- Teacher: Need to be able to view student behavior in chat room

Main Scenario

1. Student logs into/signs up for account

2. Student chooses the chat room option on dashboard
3. Student opens new chat room
4. Student finds a chat partner
5. Students practice speaking second language
6. Student makes recording of conversations

Exceptions

1. The student can’t find a chat partner
    a. Return to dashboard
    b. Create new chat room by clicking on the chat room option

Non-Functional Requirements

1. The student should be able to find chat partner in a few minutes
2. The student should be able to record conversations

Use Case Name: Students Upload a Conversation

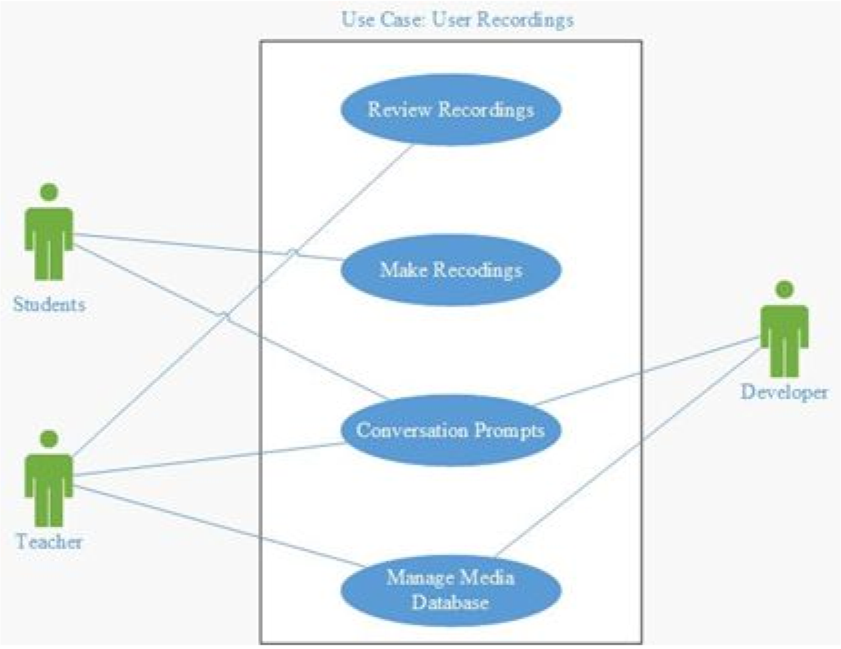


Figure 6: *Use case diagram for user recordings*

Stakeholder Interests:

- Students: Need to upload conversations in response to the prompts

- Teacher: Wants to review student’s uploaded conversations and add conversation prompts
- Developer: Needs to manage all conversation storage on media server.

Main Scenario

1. Student logs on to account
2. Student chooses a conversation prompt that they would like to respond to
3. Student records their response via webcam and microphone
4. Recording is uploaded to media server
5. Student Logs out
6. Teacher chooses a recording to grade based on its accuracy to the prompt
7. Grade is posted to the student’s account

Exceptions

1. Media Server is Full

a. student cannot upload recording

b. developer is notified to free space

2. Student Cannot Log into Account

a. student is prompted to reset password

b. student logs into account with new password

Non-Functional Requirements

1. Student should be able to upload video within 2 minutes
2. System should be able to handle 200 users
3. Student should be able to find a prompt within five seconds

Updated student activity diagram.

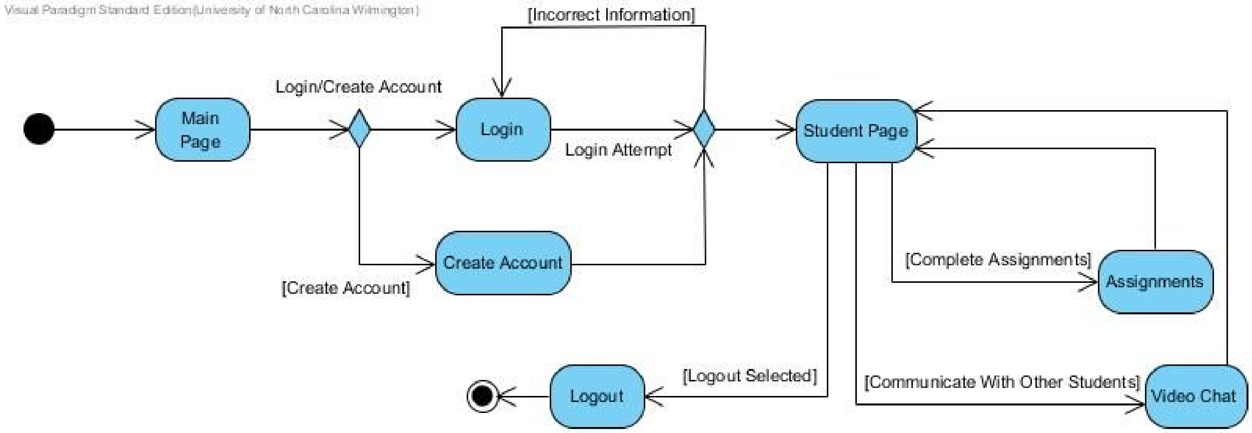


Figure 7: *Student Activity Diagram*

Figure 7 is the student activity diagram for the website. The students start on the main page of the website. If they are new students then they can register an account, if they are an existing user then they can choose to login. From the student dashboard the student is able to complete assignments that are posted by the teacher, they are also able get into a live video chat room and talk with other students. When the student is finished they can choose to log out of their account. The model has been altered to reflect that the students are not able to send messages to other students.

Updated teacher activity diagram.

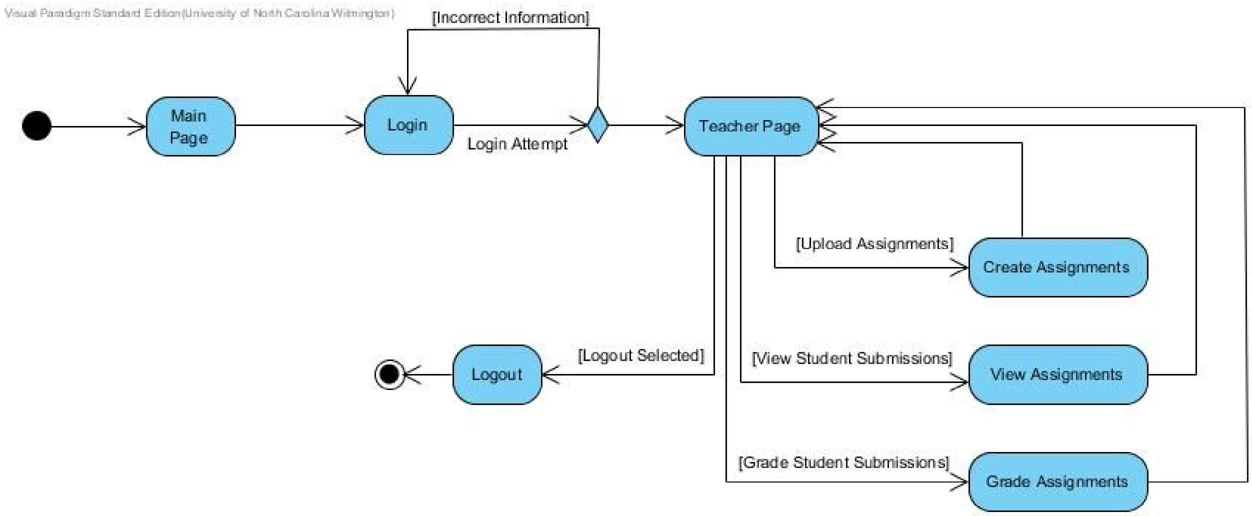


Figure 8: *Teacher Activity Diagram*

Figure 8 is the teacher activity diagram for the website. From the main page the teacher is able to login to their account. From the dashboard they are able create assignments for their students, view submissions from the students, and grade their assignments. After they are finished with the website, they can choose to log out. The diagram has been altered to reflect that the teachers are not able to send messages to the students directly at this time.

Updated Class Diagram

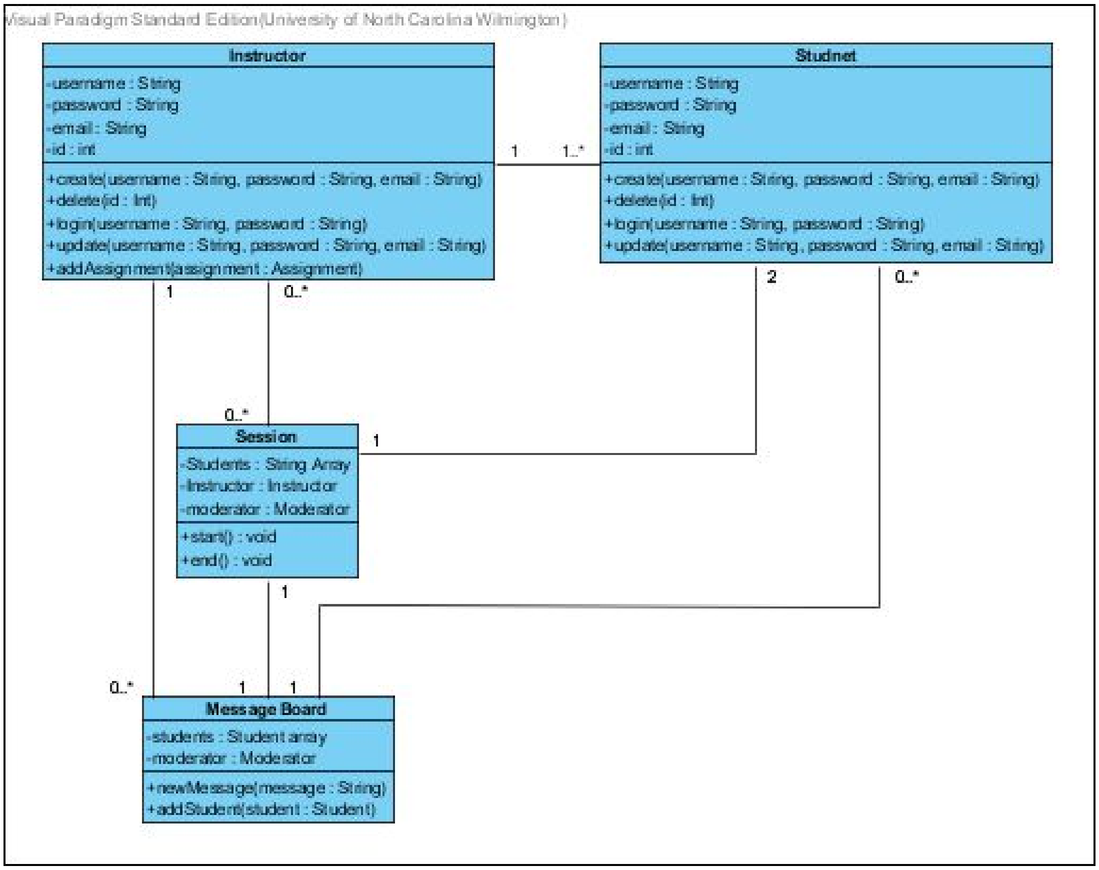

Visual Paradigm Standard Edition(University of North Carolina Wilmington)
Instructor
-username : String
-password : String
-email : String
-id : int
+create(username : String, password : String, email : String)
+delete(id : Int)
+login(username : String, password : String)
+update(username : String, password : String, email : String)
+addAssignment(assignment : Assignment)
Studnet
-username : String
-password : String
-email : String
-id : int
+create(username : String, password : String, email : String)
+delete(id : Int)
+login(username : String, password : String)
+update(username : String, password : String, email : String)
1
1..*
1
0..*
2
0..*
0..*
Session
-Students : String Array
-Instructor : Instructor
-moderator : Moderator
+start() : void
+end() : void
1
1
0..*
1
1
Message Board
-students : Student array
-moderator : Moderator
+newMessage(message : String)
+addStudent(student : Student)

Figure 9 : *Class Diagram*

Figure 9 is the updated class diagram for our software. The only major change from the original class diagram is the deletion of the moderator class. There is currently no moderator features implemented in the software, but would most likely be a necessary update later on.

2.5 Non Functional Requirements

During the design phase the non-functional requirements that were established were; ease of use for the website, security for the users, as well as performance/efficiency of the website. For the first hurdle of ease of use, we were able to accomplish this by getting other users, outside of our group, to test the website and get feedback from them. With this we were able to build the website so that new users could begin doing their required school work without first having to learn to use the website.

The second concern we had was with security with the website. We had to make sure that student's information is secure along with the need to restrict new teachers from using the website. With the student's password, we made sure to encrypt the information on the servers. This would ensure that if anyone unauthorized accessed the database then people's information would be safe since the account passwords would be unusable. The next part was making sure that not anyone could create a teacher account. In order to register as a teacher, the admin of the site would have to be contacted to authenticate the status of an individual as a teacher. The user, if made into a teacher account, could then create new classes and assignments for their students.

The last non-functional requirement is that the website needs to run efficiently and not have any performance issues. This was a big concern for our group, due to the nature of the website. Since video streaming depends on the connections of it's users, we needed to make sure that it could handle all forms of connections. We did this by limiting the quality of the videos that is recorded for student submissions. This lowers the data that is required to be passed through a network to the database. Another way was to buffer the video so as to only upload the submission when the student has completed their session and there isn't excess stress on the network.

3 Design Models

This section of the document contains all the updated design models from the previous design phase. Each diagram has been updated to reflect the changes in the process during the implementation phase.

3.1 Architectural Design

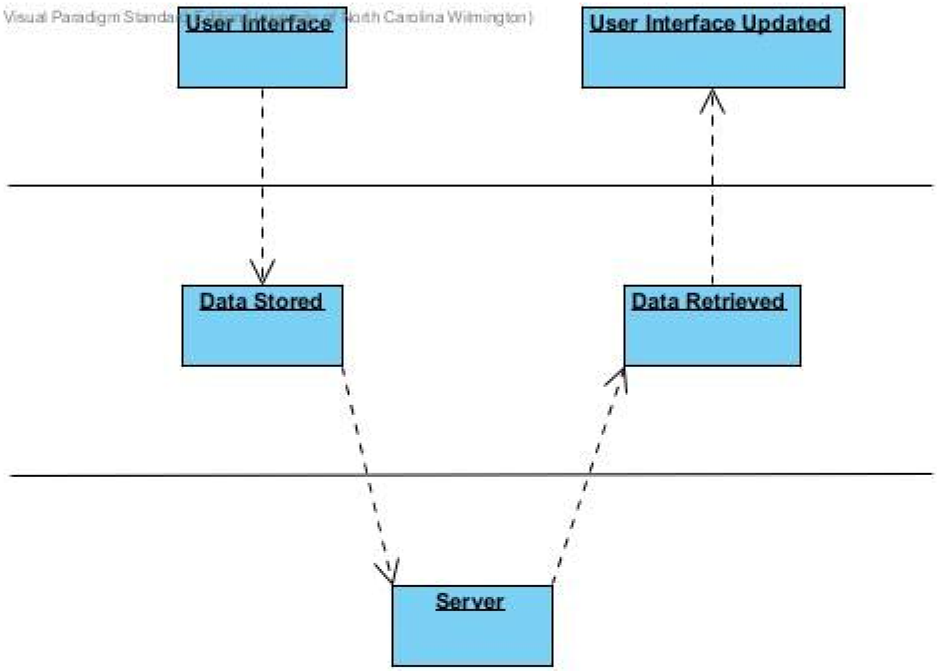


Figure 10: *Updated Architectural Design*

Figure 10 uses a different architecture style than was previously used in the design phase. Originally our project was using a real-time architecture style, however with further development our group decided that a three tiered system would work better for the project. Part of this was due to the shift in goals for the project. Originally a peer-to-peer video chat system was being developed, but with technical problems the idea was limited to an open video chat where multiple people could join. This change is what caused the switch to a user based system, with just the individual user in mind.

3.2 Solution Focused Activity Diagram

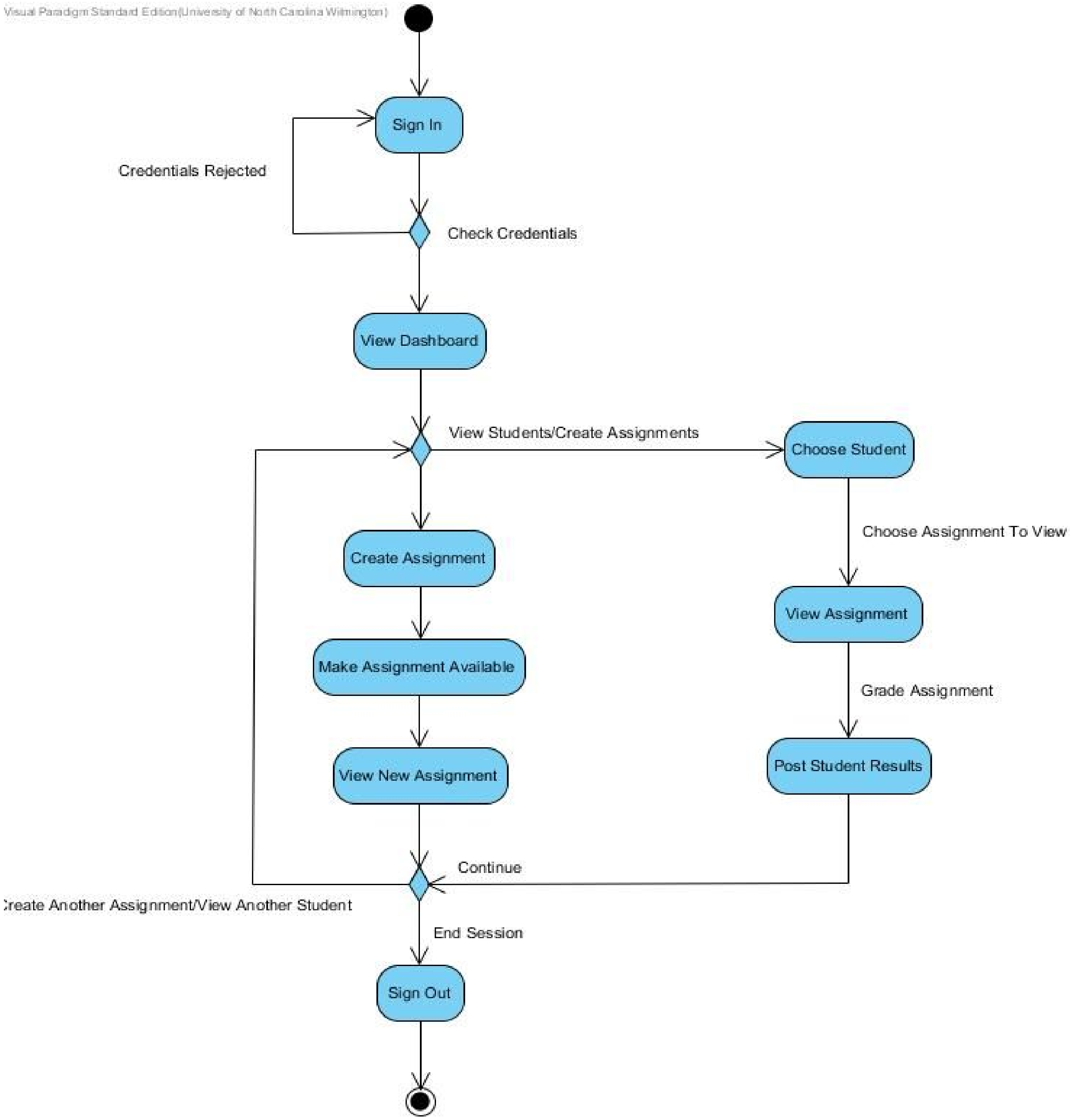

Visual Paradigm Standard Edition(University of North Carolina Wilmington)
Sign In
Credentials Rejected
Check Credentials
View Dashboard
View Students/Create Assignments
Choose Student
Create Assignment
Choose Assignment To View
View Assignment
Make Assignment Available
Grade Assignment
View New Assignment
Post Student Results
Continue
:reate Another Assignment/View Another Student
End Session
Sign Out

Figure 11: *Updated Teacher Activities Diagram*

Figure 11 is the updated teacher solution focused activity diagram. The parts that were changed involved the teacher submissions. Our project so far has condensed the submissions so that there are only picture prompts for the students assignments. Previously planned was the ability for the teacher to upload video prompts, so as to help teach the proper pronunciation of the Japanese language. However, due to time constraints this feature was left out in favor of the picture prompt.

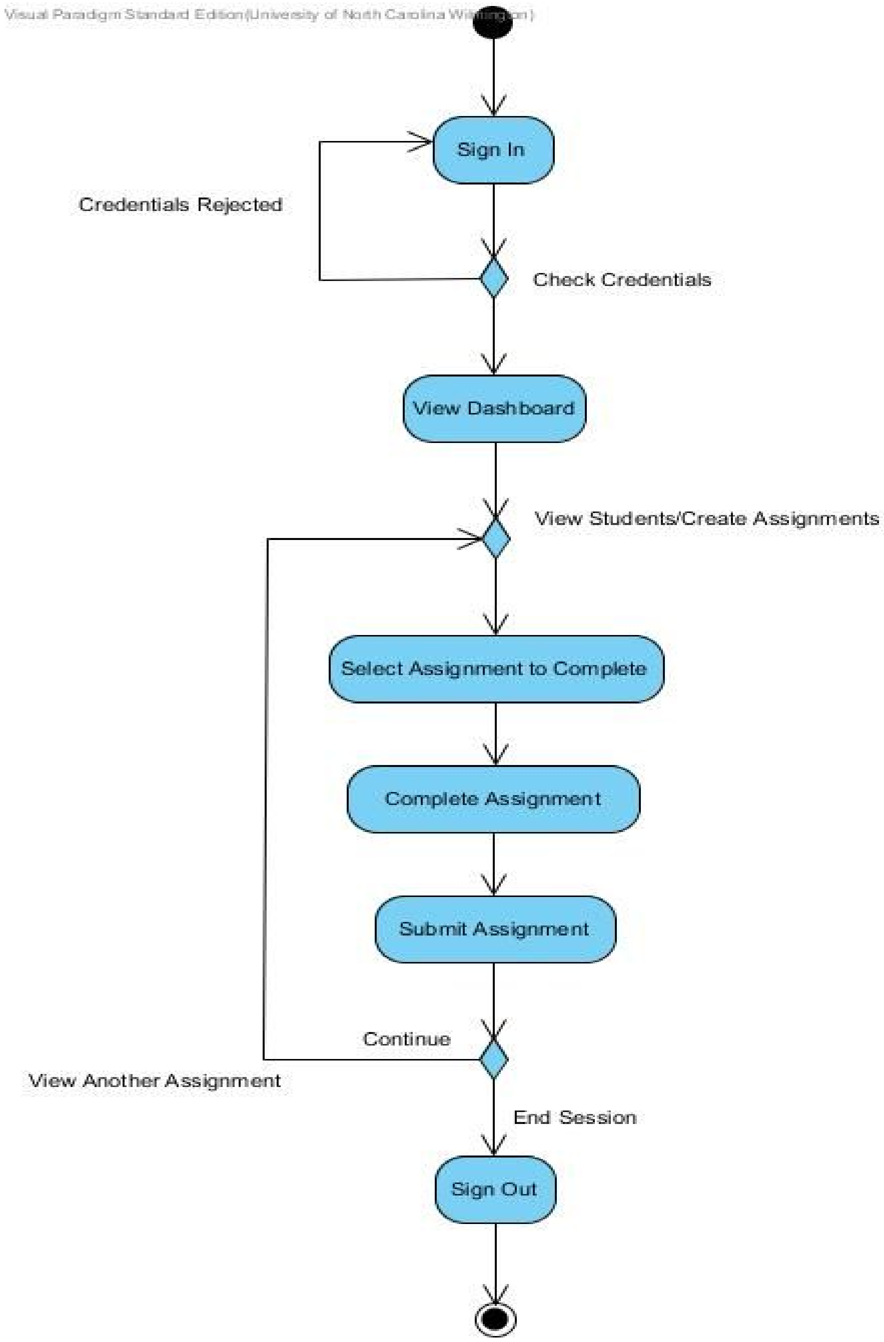


Figure 12: *Updated Student Activities Diagram*

Figure 12 is an updated version of the student solution focused diagram. The parts that were changed involved removing some of the assignment types that the student can complete. The video assignments, as well as the quiz assignments have been removed for just a picture prompt that the students will take. This has been adjusted to reflect the state of the project that has currently been developed.

3.3 Sequence Diagram

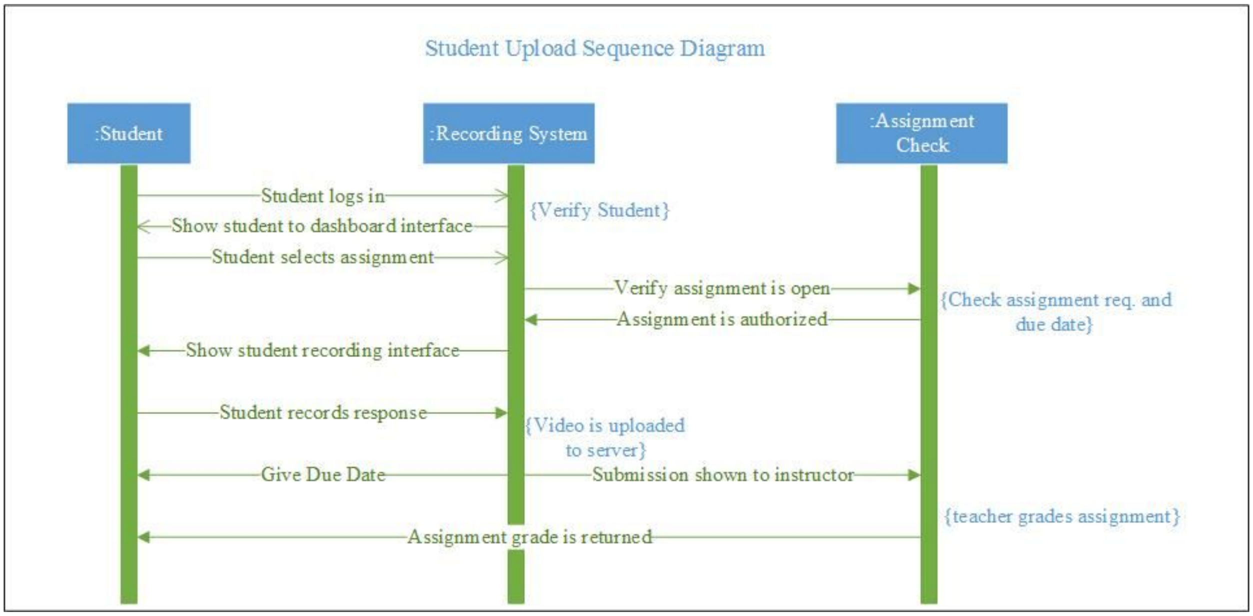

Student Upload Sequence Diagram
:Student
:Recording System
:Assignment Check
Student logs in
{Verify Student}
Show student to dashboard interface
Student selects assignment
Verify assignment is open
Assignment is authorized
{Check assignment req. and due date}
Show student recording interface
Student records response
{Video is uploaded to server}
Give Due Date
Submission shown to instructor
{teacher grades assignment}
Assignment grade is returned

Figure 13: *Sequence Diagram for Student Upload*

Figure 13 is a sequence diagram for the steps involved in the student uploading a video submission. The student begins by logging in and is directed to the student dashboard, then the student will select the assignment that needs to be completed. From here the student will record up to three submissions, where the student will select one to submit. The file is then uploaded to the server and the teacher is able to grade the submission.

3.4 State Diagram

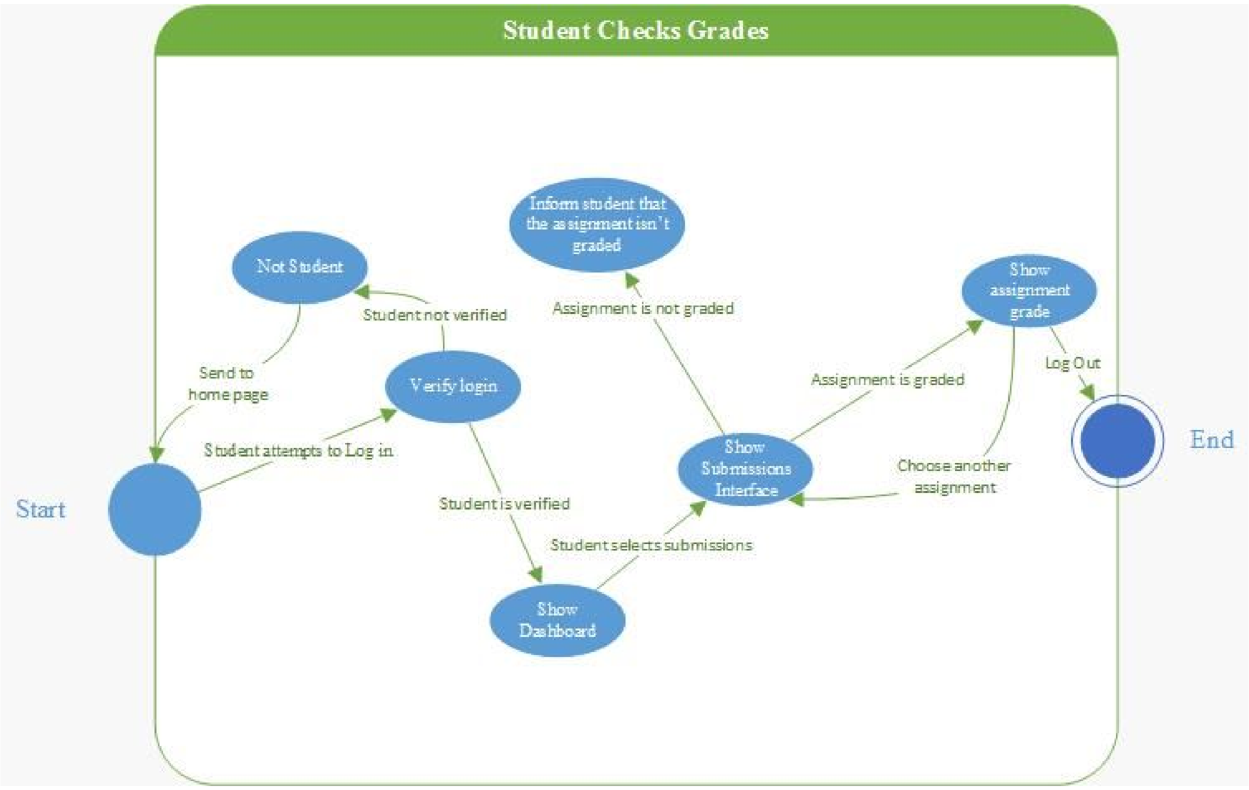


Figure 14: *State Diagram for Student Viewing Grades*

Figure 14 show the state diagram for how students access their accounts and view their grades. The student begins by logging into their account, after doing so they are directed to their dashboard. From there, the student goes to their submission tab and can view the status of their submitted assignments to check if they have been graded or not. From there the student can continue browsing the website , or they can log out if they are finished.

4 Implementation

This section of the report will contain the programming language breakdown of what languages were used and what feature of the project they were used to develop. Along with this, all of the testing that was carried out for the project will be detailed, such as; what testing method was used, the test cases that were developed, along with who did the testing. Also included will be a user manual that gives detailed instructions on how to properly use the website.

4.1 Programming Languages

PHP - This was used to handle the database interface. IT's tasks were; checking the input into the database, along with processing the data. PHP was also used to connect to the database and handle the server logic.

Cascading Style Sheet (CSS) - CSS is the building block of the website. This is where the code is run to make the website look and run the way it does.

Javascript - All of the video components were developed using javascript. In this code, the logic for running the videos from a webcam are placed, as well as, connecting to the PHP code to upload videos to the server.

Ajax - This code is used to package the data from the website so that it can be transferred to the server.

SQL - SQL was used to submit queries to withdraw information from the database.

HTML - This was used to help build the web pages for the site.

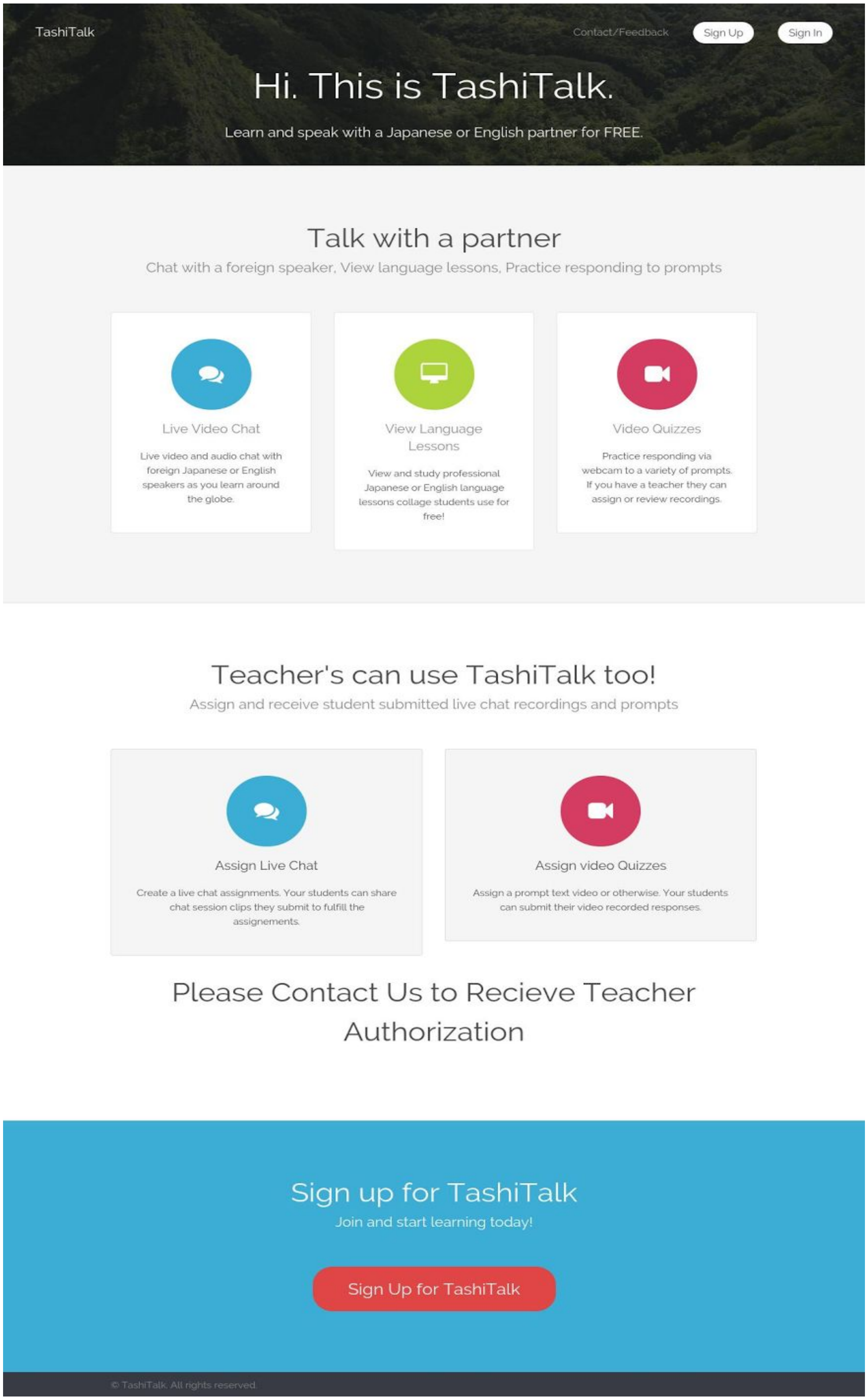
4.2 Implemented Software Functions

For the project we were able to implement all but one of the required features. What we did get implemented is; the ability for students to create new accounts and the ability for existing teachers and students to logon. The features on the teacher side that we implemented were the ability for the teacher to create classes that they are teaching, and the ability to create assignments for those classes. After the students complete the assignments the teacher is then able to view and grade the submissions. On the student side, they are able to view assignments that are created by their teacher, they can then complete the assignments, and upload their submission. Then the students are able to view assignments that they have submitted along with the grade given by the teacher.

4.3 Software Functions Not Completed

Through time constraints and technical hurdles that our group had to overcome, there was one feature that was not implemented to the full extent that was originally designed. This feature allowed students to create video chat sessions where they are able to talk one-on-one with another student of a different language. A basic version of this feature was implemented in an open chat room style session. This however wouldn't limit the number of users, or set the students up with foreign language students.

4.4 User Interface Design



Hi. This is TashiTalk.

Learn and speak with a Japanese or English partner for FREE.

Talk with a partner

Chat with a foreign speaker, View language lessons, Practice responding to prompts

Live Video Chat

Live video and audio chat with foreign Japanese or English speakers as you learn around the globe.

View Language Lessons

View and study professional Japanese or English language lessons collage students use for free!

Video Quizzes

Practice responding via webcam to a variety of prompts. If you have a teacher they can assign or review recordings.

Teacher's can use TashiTalk too!

Assign and receive student submitted live chat recordings and prompts

Assign Live Chat

Create a live chat assignments. Your students can share chat session clips they submit to fulfill the assignements.

Assign video Quizzes

Assign a prompt text video or otherwise. Your students can submit their video recorded responses.

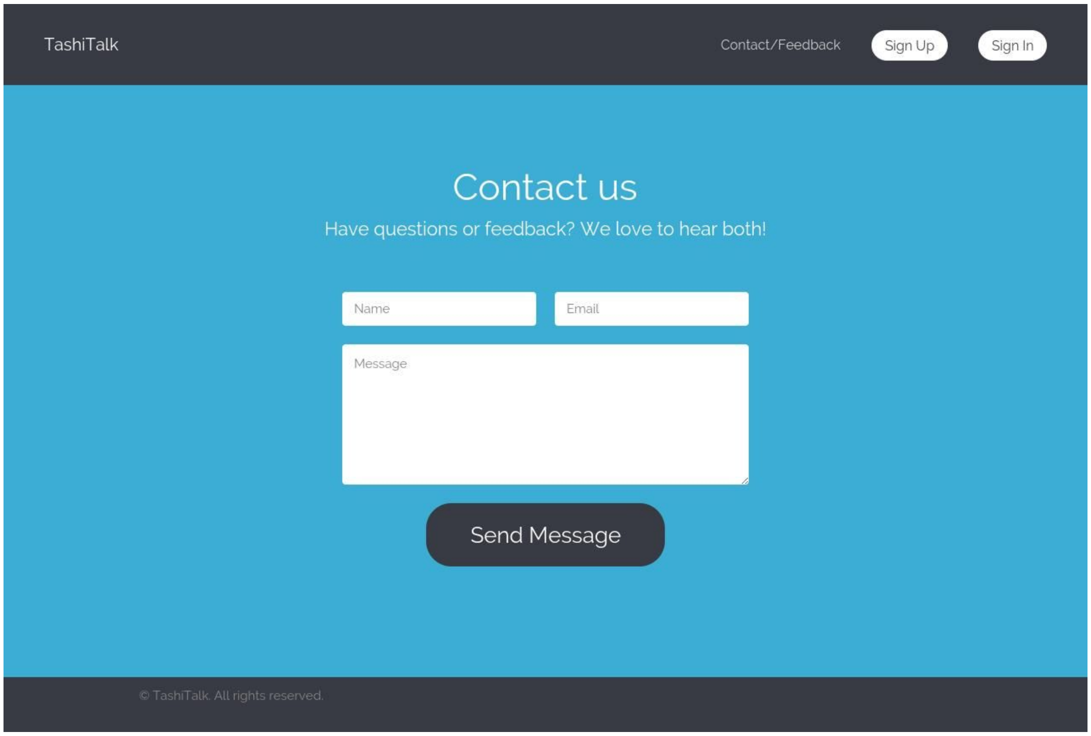
Please Contact Us to Recieve Teacher Authorization

Sign up for TashiTalk

Join and start learning today!

© TashiTalk. All rights reserved.

Figure 15: *The Homepage*



Contact us

Have questions or feedback? We love to hear both!

© TashiTalk. All rights reserved.

Figure 16: *Feedback Webpage*

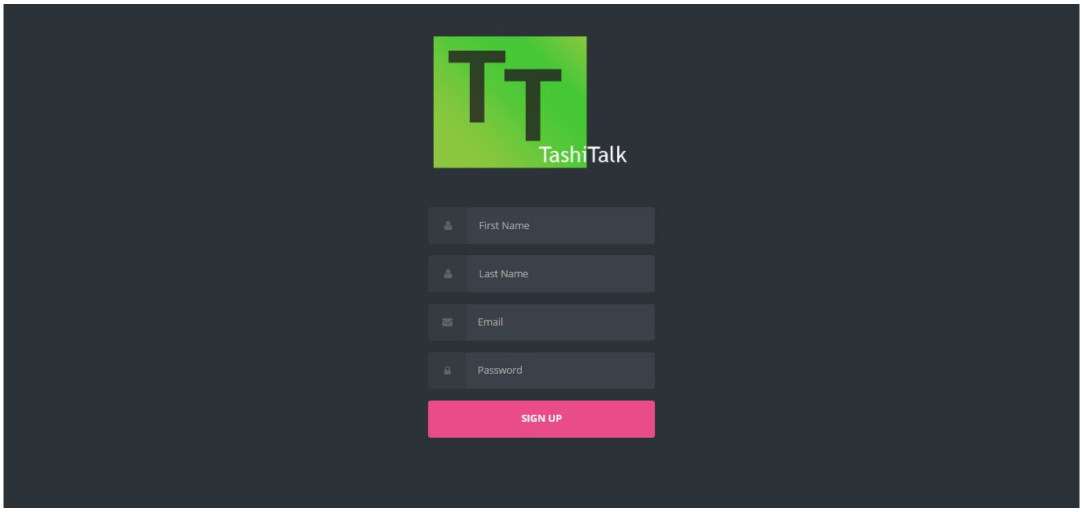


Figure 17: *The Sign Up Webpage*

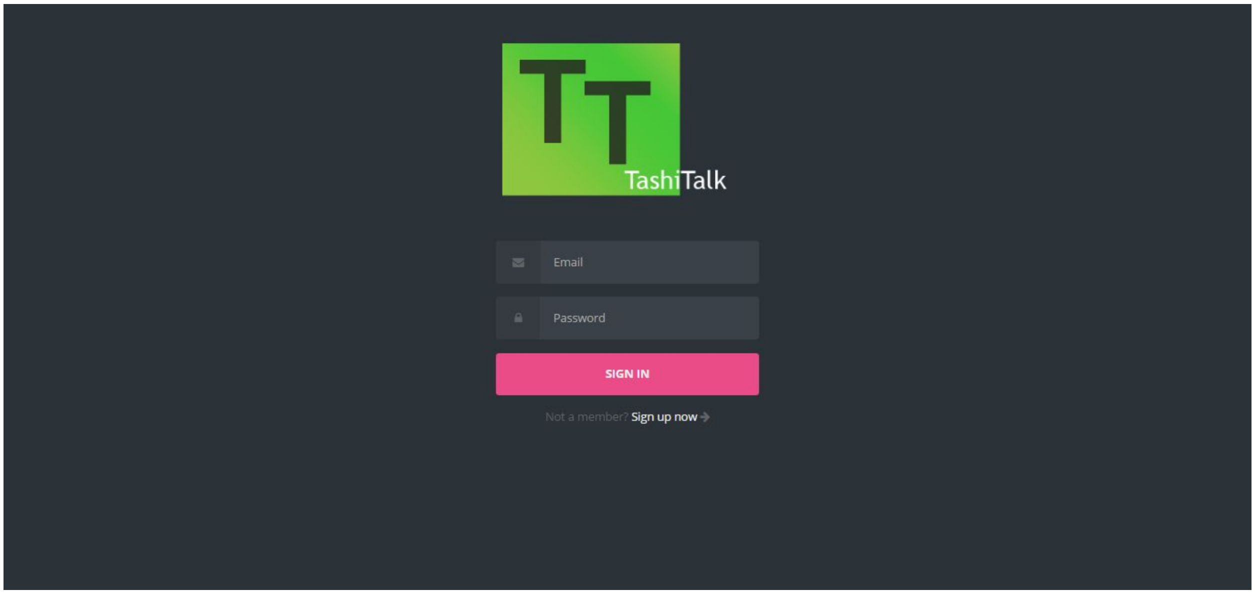


Figure 18: *The Sign In Webpage*

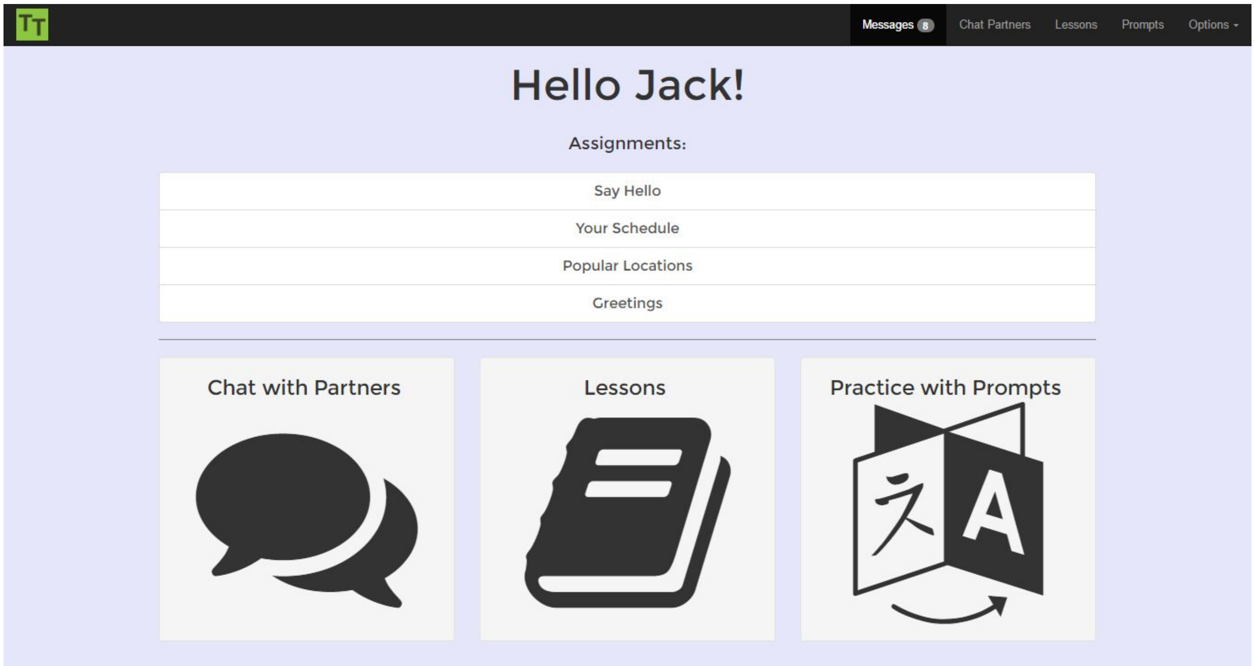
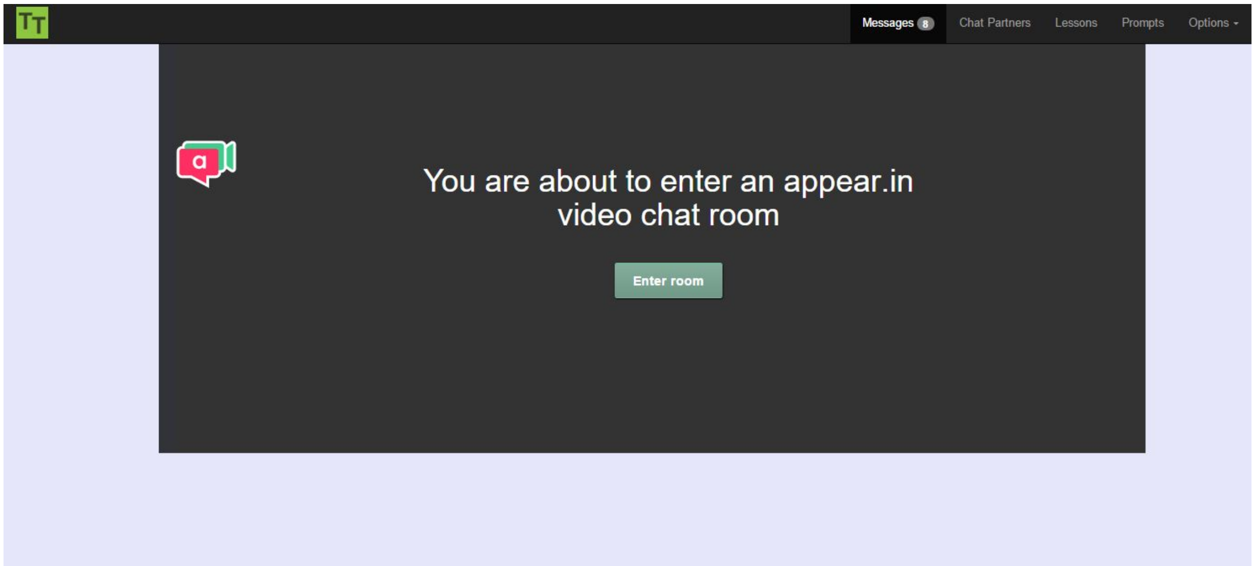
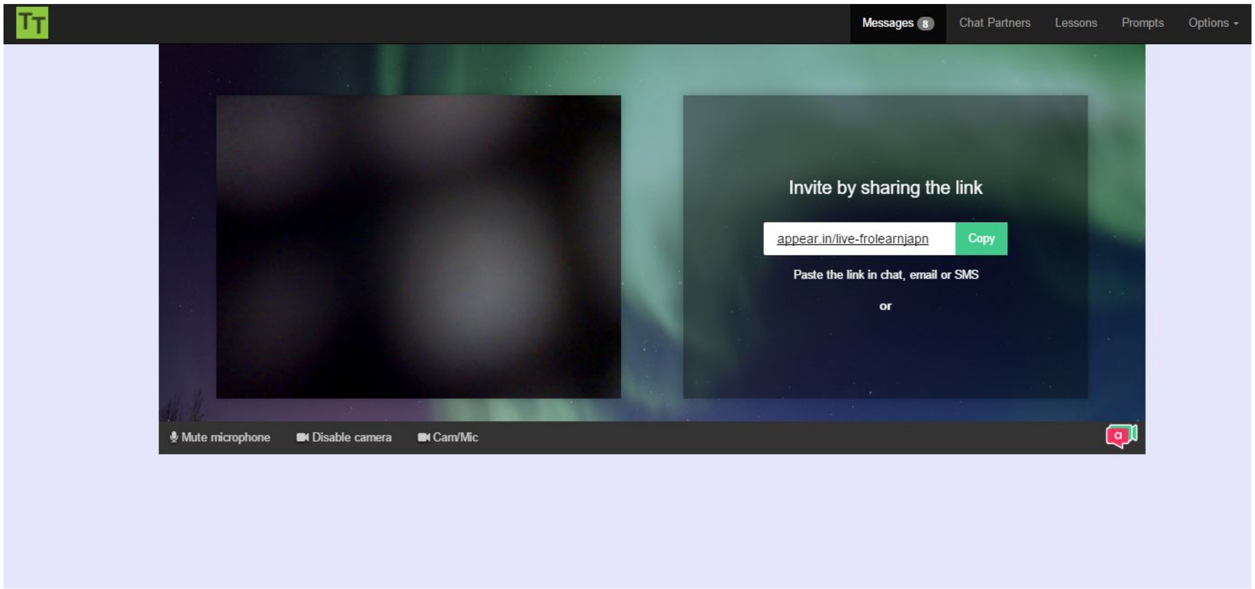


Figure 19: *The User Dashboard*



You are about to enter an appear.in video chat room

Figure 20: *Video Chat Page*



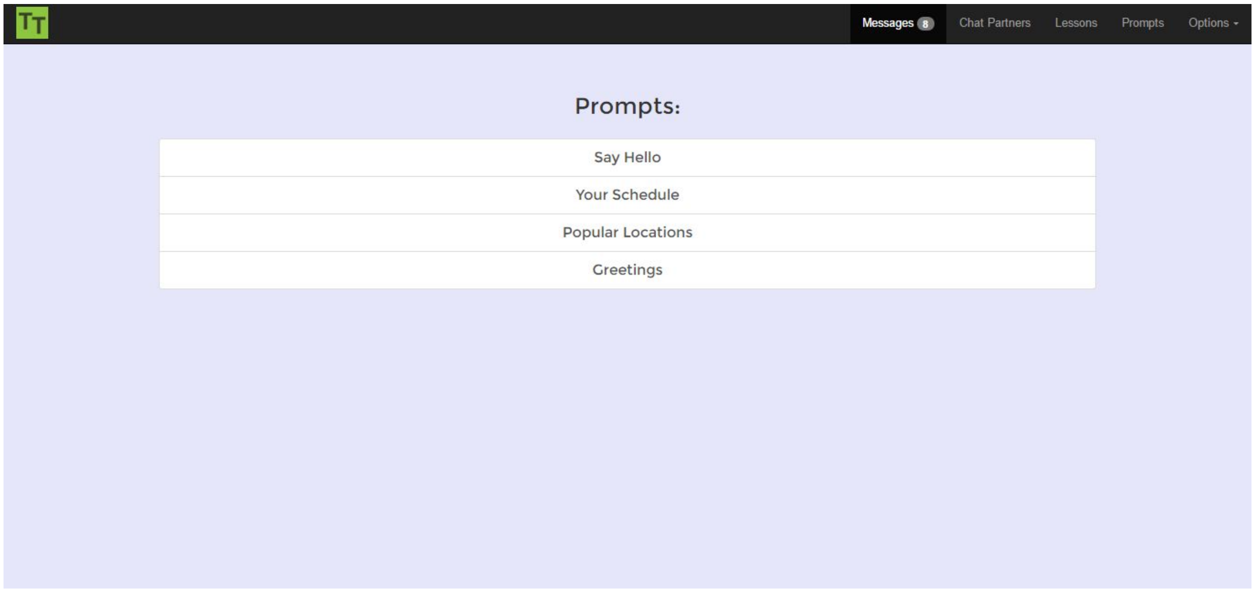
Invite by sharing the link

appear.in/live-frolearnjapn

Paste the link in chat, email or SMS

or

Figure 21: *Video Page After Entering Room*



Prompts:

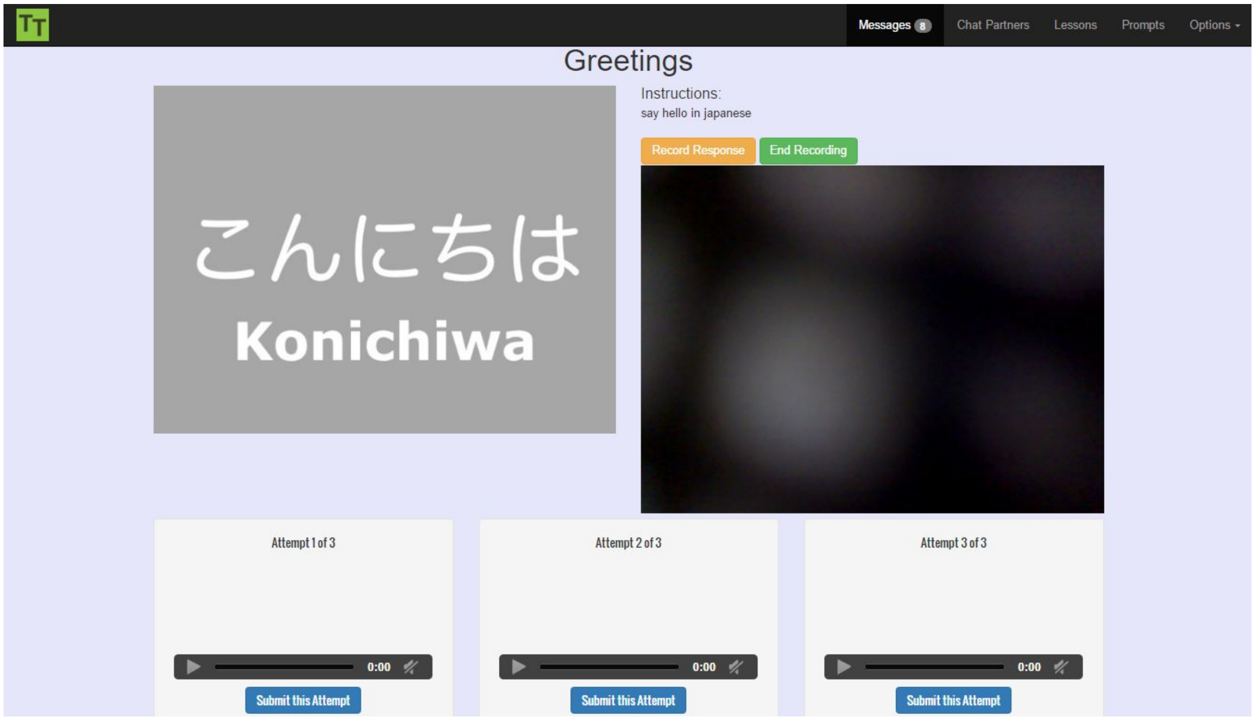
Say Hello

Your Schedule

Popular Locations

Greetings

Figure 22: *Assignments Webpage*



Greetings

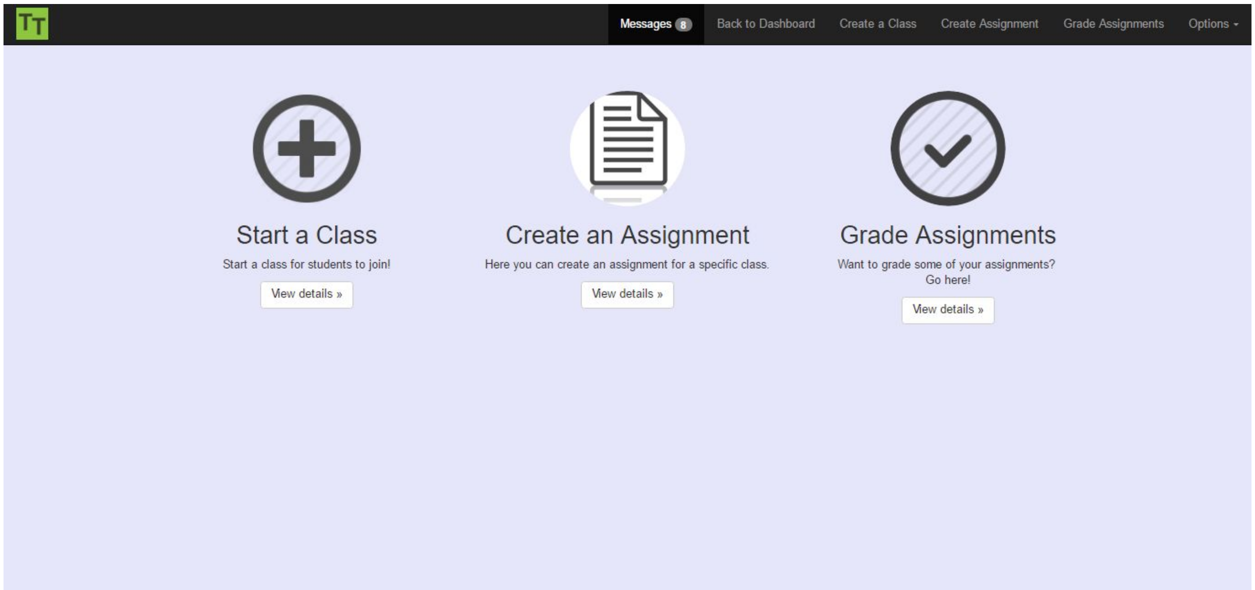
Instructions:
say hello in japanese

Attempt 1 of 3

Attempt 2 of 3

Attempt 3 of 3

Figure 23: *Respond to Prompt Webpage*



Start a Class

Start a class for students to join!

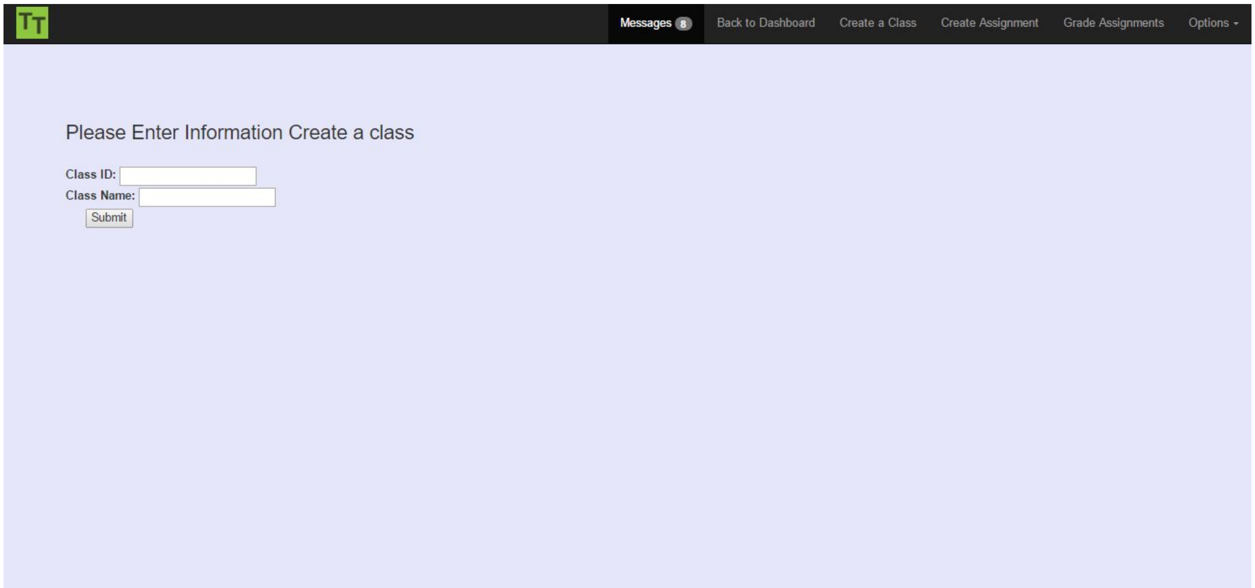
Create an Assignment

Here you can create an assignment for a specific class.

Grade Assignments

Want to grade some of your assignments? Go here!

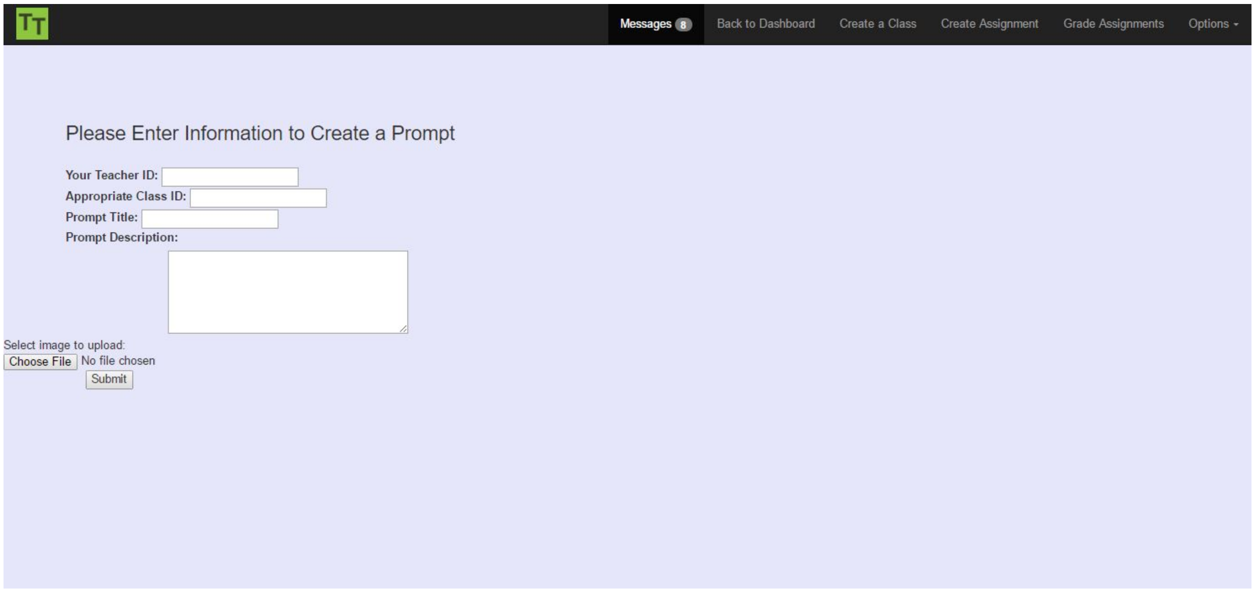
Figure 24: *Teacher Dashboard Webpage*



Please Enter Information Create a class

Class ID:
Class Name:

Figure 25: *Sequence Diagram for Student Upload*



Please Enter Information to Create a Prompt

Your Teacher ID:

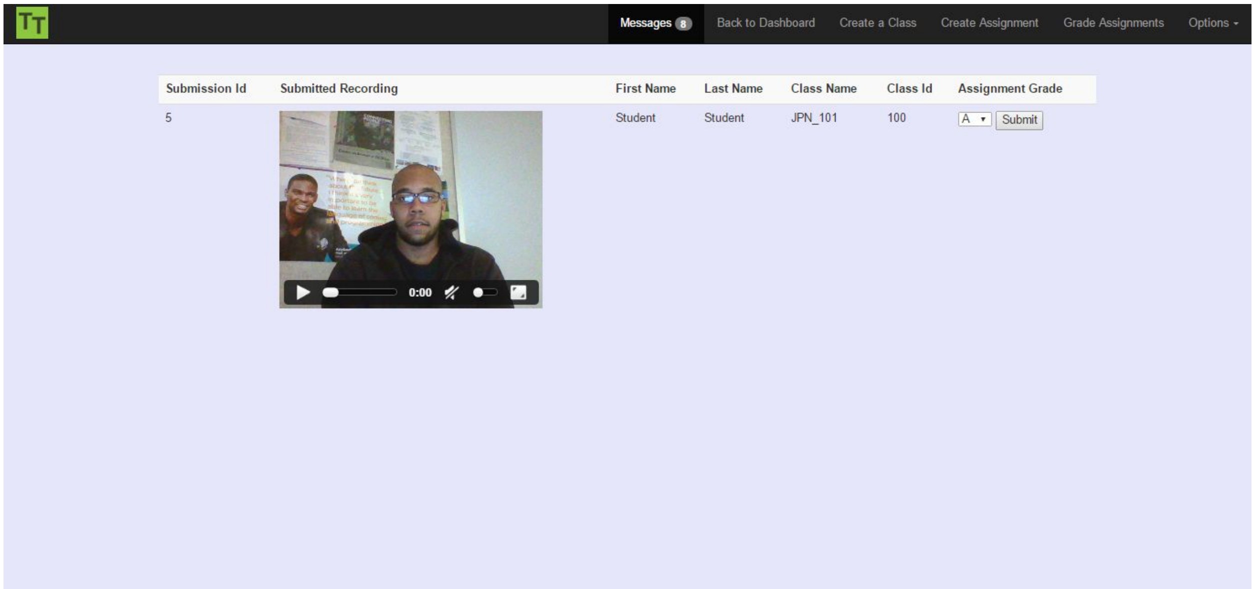
Appropriate Class ID:

Prompt Title:

Prompt Description:

Select image to upload:

Figure 26: *Create a Prompt Webpage*



Submission Id	Submitted Recording	First Name	Last Name	Class Name	Class Id	Assignment Grade
5		Student	Student	JPN_101	100	A ▾

Figure 27: *Grade Webpage*

4.4i Options implemented

- User toolbar supports visiting the live via chat page
- User toolbar supports visiting the Prompts list page
- User toolbar supports logging out via the Options drop down
- User toolbar supports visiting the feedback page via the options drop down
- User toolbar supports visiting the user dashboard via the logo to the left
- User dashboard page supports visiting the assignments and prompts, the video chat page, and visiting active assignments directly.
- Prompt page supports recording and stop recording via buttons.
- Prompt page supports submitting a recorded assignment via a submit button.
- Prompt submited page support a continue button that takes you to the user dashboard.
- Teacher tool bar supports a going back to the dashboard via a back to dashboard button
- Teacher tool bar supports going to the create a class page via a create a class button
- Teacher tool bar supports going to the create assignment page via the create assignment button
- Teacher tool bar supports going to the grade assignment page via the grade assignment button
- Teacher grade select page supports choosing and submitting a grade for an assignment via a drop down and a submit button
- Teacher prompt page supports creating a prompt via input boxes for, teacher ID, appropriate class ID, Prompt Title, Prompt Description, choosing file dialog and submit button
- Create class page supports creating a class via the class ID, class name and submit buttons.

4.4ii Options not implemented

- User to user chat menu not implemented. User to user chat was an additional feature that could be added but the team decided to not attempt to tackle those features.
- User options button not implemented in the drop down list. Advanced user options such as resetting or recovering an email were extra functionality that were beyond the scope of the project.
- The lessons link from the user dashboard does not link to anything because the transfer of the teaching content was beyond the scope of this project however the client wanted it as a feature that could be built out so a section was included.

4.5 Test Plan

For our testing process we used black-box testing. The reason we chose this method is because of the functionality of the website is better tested through use, rather than breaking down the code into it’s individual methods.

List of test cases, see attached document.

4.6 Testing Process

The main implementation of the project was handled by Daniel Baggott and Matthew Canton. As for testing, all group members tested the functionality of the website, to see if everything was working as it should. Most of the testing was carried out as the parts of the project were added to the website.

Responsibilities for testing are attached to test case document.

4.7 Problems Encountered

Major problems encountered included discovering what technology was required to accomplish the goals. During which we learned about media servers, software media routers, video conferencing, working with live media streams, webRTC, browser beta functions and more. Some members did not start the project familiar with web languages and as such had to learn as they went. Keeping a relational database synced and working proved somewhat challenging. Hosting the project ourselves on a cloud linux server required learning linux server command line tools.

4.8 Non-functional Requirements

The non-functional requirements that were looked into for our project are; ease-of-use, security, and performance of the website.

The first one of ease-of-use is important in that multiple people will be using the webpage, so it is impossible to guide every user through the site. Our goal was to make a website that an individual could go to without any instructions and easily navigate. In order to test our design, we had multiple people outside of the project test the website. The users found the website simple to use and they were able to quickly navigate around after only a couple minutes of use.

The next feature we wanted to take care of was the security of the user's information. Since personal email addresses and other information is being stored on our servers, we wanted to ensure the security of this information. There were several steps that we took. The first is that all passwords stored on the server are hashed. This makes all the text in the database unlegable, this would make sure that even if someone unauthorized accessed the server, that no useful information would be taken. The second feature that was implemented was logout security. This means that when a user is done with the website and they logout, then another user can't come in behind them and try to hit the "back arrow" on the browser to reload into the previous session.

The final area we wanted to cover is the performance of the website. This area is important due to the type of website that is being built. One of the features to be implemented is the ability for students to communicate via live chat. Since it is intended for students on the opposite side of the world from each other to communicate network problems are a concern. One of the areas we looked at improving is with bandwidth consumption. This was handled by decreasing the number of calls to the database. Variables were reused locally instead of constantly calling back into the database for repeated information.

4.9 Lessons Learned

One of the areas that our group found to be difficult was time management. Due to some of the group members not knowing specific programming languages that were used in the development of the project, time had to be taken away from development to learn these new languages. This would cause tasks that needed to be done by a certain time to be pushed back, causing a ripple effect through the rest of the development process.

Again learning a new series of programming languages caused a slowdown in production. For future projects like this it would be better to stick with more familiar languages for development, and if a specific problem needs to be addressed then the group can branch out to needed languages.

Our group at the beginning also had a problem in properly delegating the work out to be completed. When setting up tasks to be completed we had to hand out the parts based off of everyone's skill level and what languages they knew. This caused problems as we didn't want to overload a person with a task they couldn't complete quickly, due to the having to research the material first.

If we did this kind of project again, then we probably try to limit the scope of the project a little more. In the beginning we thought it would be easy to implement all of the features that we wanted, however, after getting into the code and doing more research we quickly realized that the project was more than we could handle in the given time period. Another thing we would change is the development process that we implemented. Our development went in with the idea of making a fully functioning website from scratch. Realistically this was not something we could accomplish in the limited time span and the scope of the project. If we did it again we would go with a prototype process, where we build a simplistic version of the website with all of the core features, then expand from there. Adding new features as needed and refining the functions already in place.

Our group took away from this project the concept of better time management. When doing the research for our project, we would keep pushing back the start of the actual building of the website. This caused delays further along in the development due to the necessity of finishing certain parts of the project before others. We also took away the importance of delegating work better. There were problems making sure everyone working on the project knew the piece they were working on, this also caused delays as some parts that were needed weren't being worked on.

4.10 User Manual

There are two user types that can use Tashi Talk, students and teachers. The following manual will start with students and then teachers.

1. 1.Students
    a. Registering - Before using Tashi Talk students are required to register. This can be done by clicking the “Sign Up” button located on the homepage of the website. Once on the sign-up page enter a first name longer than two characters. A last name longer than two characters. A valid email. And a password longer than 8 characters that must contain numbers and letters. Aftering entering the data the student can click the button following the fields to create their login.
    b. Logging in- Logging into Tashi Talk can be done via the login page. A student can reach the login page by clicking the “Sign In” button on the home page. Once on the login page then enter the email and the password you signed up with when creating your user. If entered correctly they will log you into the student dashboard (main page). If entered incorrectly an error message will be displayed and you may attempt to log in again.
    c. Contact/send feedback to Tashi Talk - Through the feedback page feedback and messages can be sent to Tashi Talk. The feedback page can be accessed via the front page of Tashi Talk by clicking the “Contact/Feedback” button located at the top of the page. When logged in the feedback page can be accessed through the feedback button available through the drop down menu in the toolbar at the top of the page. Sending feedback requires a message though a name and Email are optional. Clicking send message will send your message to Tashi Talk and a notification will be displayed.
    d. Log out - When logged in the top toolbar will contain a drop down menu. Click the drop down menu to expose more options. The option at the bottom of the list is to Log Out. Logging out will require that you log in again before you can use features exposed in Tashi Talk.
    e. Chat with Partners - You can chat with others on the site with live video chat by accessing the video chat page. The video chat page can be accessed via the “Chat Partners” button in the top toolbar. If on the home logged in page choosing the option “Chat with Partner” will send you to the chat page. When on the chat page click the enter room button to enter the live video chat.
    f. Respond to video prompts or assignments - Currently any default and assigned video prompts can be accessed via the home logged in page. Clicking the “Prompts” button in the toolbar or clicking the “Practice with Prompts” button in the student’s home page will take them to the prompts page from which they can see available default and assigned prompts in a list. Assigned prompts are available on the prompts page and on the student’s main page as a list.
2. Teachers
    a. Teacher’s have a special dashboard removed from the user dashboard. Obtaining the required id numbers to be a teacher user requires contacting Tashi Talk and becoming approved and being issued the appropriate id numbers.
    b. Teachers can create a class - Teachers can create a class by clicking “Start a Class” icon on the Teacher’s dashboard or by clicking the “Create a Class” button in the toolbar. Enter the class ID and class name of the new class then submit the information.
    c. Create assignment - Teachers can create an assignment on the create an assignment page. Teachers can access the create an assignment page by clicking on “Create an Assignment” page from the main page or the “Create an Assignment” button in the toolbar. On the create an

assignment page the teacher should enter their teacher ID, the class ID, the prompt title, the prompt description, and choose a prompt image to upload via the “Choose File button”. After entering the information clicking “Submit” will create the assignment.

d. Grade Assignment - Assignment grades can be chosen via the grade assignment page. The grade assignment page can be accessed from the teacher dashboard by clicking the “Grade Assignments” icon or by clicking the “Grade Assignments” button in the toolbar. On the grade assignments page a teacher can choose a grade for the drop down list and submit it via the “Submit” button.

e. Go back to dashboard - The teacher can go back to the teacher dashboard by clicking the “Back to Dashboard” button on the top toolbar.

4.11 Bibliography

[1] W. Rankin, “Increasing the Communicative Competence of Foreign Language Students Through the FL Chatroom,” *Foreign Language Annals*, vol. 30, no. 4, pp. 542–546, 1997.

[2] W. S. Lam, “LLT Vol8Num3: SECOND LANGUAGE SOCIALIZATION IN A BILINGUAL CHAT ROOM: GLOBAL AND LOCAL CONSIDERATIONS,” *LLT Vol8Num3: SECOND LANGUAGE SOCIALIZATION IN A BILINGUAL CHAT ROOM: GLOBAL AND LOCAL CONSIDERATIONS*, Sep-2004. [Online]. Available at: http://llt.msu.edu/vol8num3/lam/. [Accessed: 22-Mar-2016].

[3]R. Madachy, "COCOMO II - Constructive Cost Model", *Csse.usc.edu*, 2016. [Online]. Available: http://csse.usc.edu/tools/COCOMOII.php. [Accessed: 17- Feb- 2016].

[4] Ganttproject.biz, "GanttProject: free desktop project management app", 2016. [Online]. Available: https://www.ganttproject.biz/. [Accessed: 17- Feb- 2016].

[5] I. Cakir, “The Use of Video as an Audio-Visual Material in Foreign Language Teaching Classroom,” *The Turkish Online Journal of Educational Technology*, vol. 5, no. 4, Oct. 2006.

[6] C. Canning-Wilson, “Practical Aspects of Using Video in the Foreign Language Classroom,” *The Internet TESL Journal*, vol. 6, no. 11, Nov. 2000.