

Design Studio 3

(Math with Friends)

Yu Fan
Karen Nguyen
Patrick Tran
Tyler Valdivieso
Linxuan Xin

Table of Contents

- 1. Overview
 - 1.1 Introduction
 - 1.2 Description
 - 1.3 Audience
 - 1.4 Other Stakeholders
 - 1.5 Goals
 - 1.6 Assumptions
 - 1.7 Constraints
- 2. Application Design
 - 2.1 Functionality
 - 2.2 Mind Mapping
 - 2.2.1 Rough Draft
 - 2.2.2 Final Draft
- 3. Interaction Design
 - 3.1 Personas
 - 3.1.1 Created Personas
 - 3.2 Mockups and descriptions
 - 3.2.1 Signing In
 - 3.2.2 Signing Up
 - 3.2.3 Main Menu
 - 3.2.4 Create/Join a Lobby
 - 3.2.5 Lobby
 - 3.2.6 Whiteboard
 - 3.2.7 Classes
 - 3.2.8 Settings
- 4. Architecture Design
- 5. Implementation Design
- 6. Details
 - 6.1 What Each Member Did
 - 6.2 Meeting Log
 - 6.3 Pictures of Our Meetings

1. Overview

1.1 Introduction:

This document is created to detail the system for a collaborative math learning application, Math with Friends, for an unspecified educational software company. Contained in this document are details on application, interaction, architecture, and key elements of implementation design for our application that we designed.

1.2 Description:

Our application is designed to promote collaboration among students to allow them to solve math problems together online at home so long as there is a computer and an internet connection.

The main function of our application is a split-screen whiteboard, divided into 2-4 sections depending on how many students are collaborating. Each student gets their own whiteboard which they can manipulate. They are also able to see the whiteboards of all the other students that they are working with, but students can only manipulate their own respective whiteboard.

Students work together on the same displayed math problem and come to a consensus on a single final answer. To communicate, students will be able to chat with voice and text when they are connected together in a lobby for a whiteboard session.

1.3 Audience:

- The target audience are students between the ages of 8 and 12.

1.4 Other Stakeholders:

- Instructors
- Parents
- Client educational software company

1.5 Goals:

- Enable students to learn by collaborating on math problems
- Provide a simple and easy-to-use interface for young students

1.6 Assumptions:

- Users have access to a computer with internet.
- Users or their instructors will coordinate when to work together
- Instructors will handle conflicts and deal with problematic students
- Access codes will be given to students by instructors to sign up for accounts
- Algebra is the highest math done by students ages 8-12

1.7 Constraints:

- Platform: Windows and MacOS
- Cannot be web-based
- Interface must be easy to understand and use for young students
- A whiteboard session shall have 2-4 students

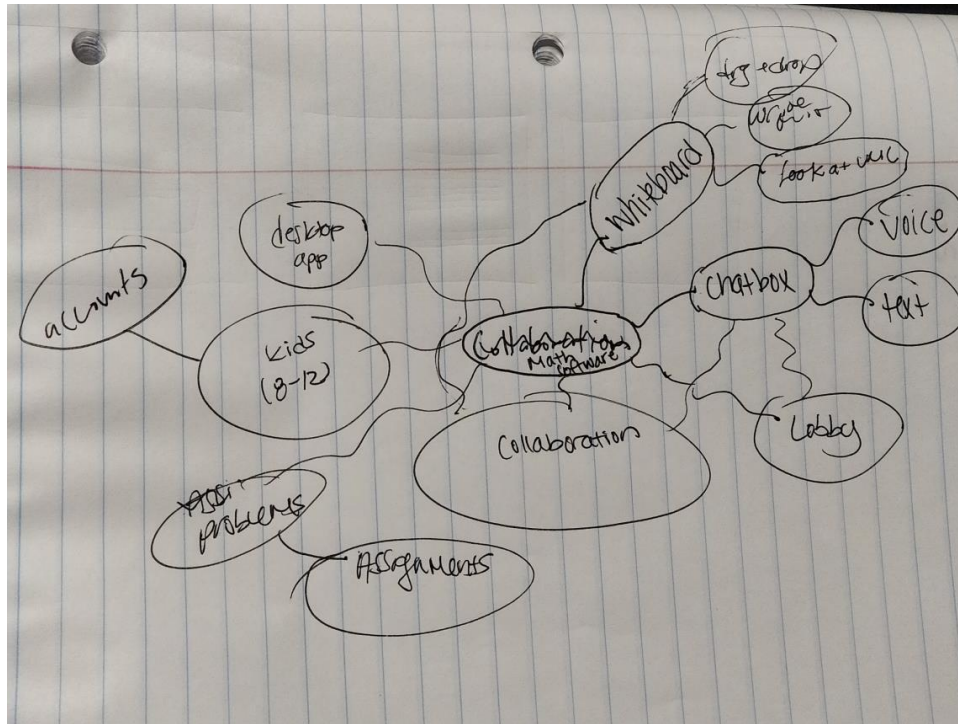
2. Application Design

2.1 Functionality:

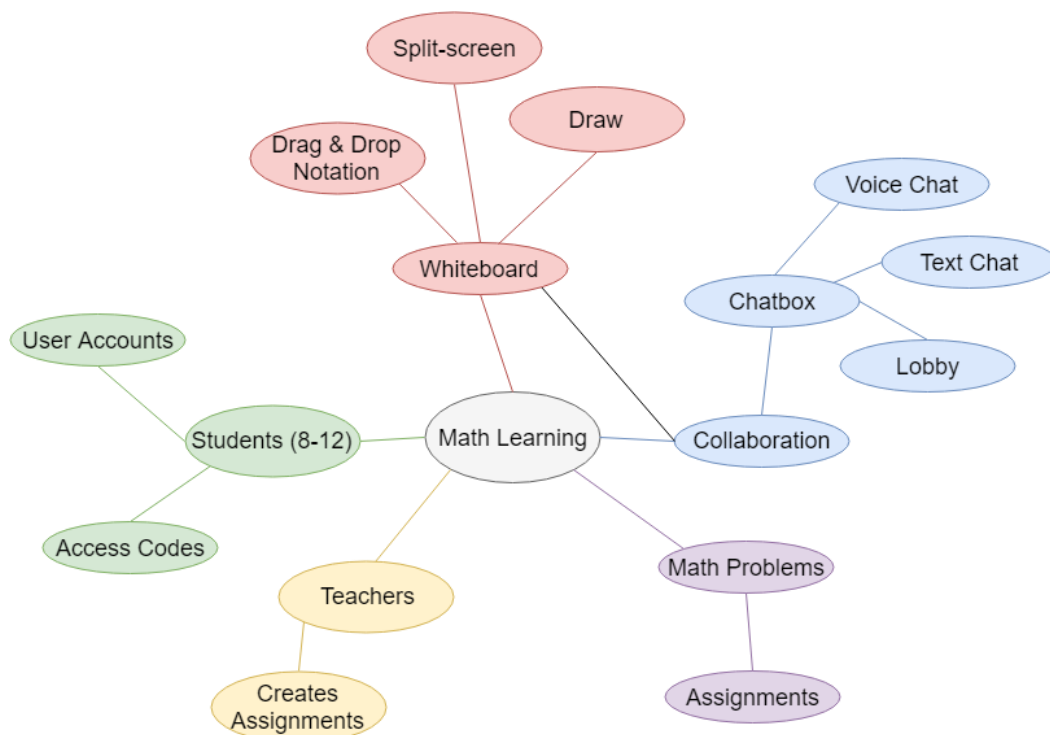
The main focus and functionality of our design for the system revolves around promoting collaborative math learning amongst kids aged 8-12. The functionality of our design caters to the process of collaboration through a few key features. Our key collaborative features are the whiteboard and lobby system. The users can create and/or join a lobby which consists of two to four users and each user gets their own whiteboard displayed in a split-screen fashion. The whiteboard has tools such as markers and symbols for the kids to use and serves as an area on the screen where students can jot down their ideas and scratch work for all in the lobby to view. Also, the teacher provides a set of problems that the students can work on which are selected in the lobby. After selecting a problem set to work on users are able to collaborate on the problems together through functionalities such as voice chat, text chat, and being able to view each other's whiteboards. In addition, users take turns submitting answers for each problem in order to prevent one user from doing all of the work for each problem and to make sure that everyone in the lobby is participating.

2.2 Mind mapping:

2.2.1 Rough Draft



2.2.2 Final Draft



These
are two
mind

maps our team created during one of our meetings on March 6th. The first was done very roughly and was meant to just get ideas down. The second one was redone to be

more organized and neat. It also contains a few more branches. The process of creating the mindmap allowed us to brainstorm and it helped us focus our efforts on what we found as the essence of the problem. Mind mapping allowed our team to easily visualize necessary features for our software and helped us structure our information, which in turn helped our comprehension of the problem.

3. Interaction Design

3.1 Personas:

The goals of personas are to:

1. Determine core value for a persona
2. Group similar users together
3. Explore goals, roles, needs, context, and values
4. Make summary research more concrete
5. Communicate user information to your team

Personas helped our design team better understand the needs and interests of our customers (8-12 years old kids). Typically, personas would be based on in-depth user research but since we were told this was not necessary for our assignment we based our personas off of our knowledge of kids in the age range mentioned above. We have four personas created by different group members to avoid stereotypes and improve diversity.

User personas are useful during the application development process. We use personas as a tool for communicating and discussing our primary audiences. It also helps to clarify the goals of our software and keep various members on point.

By creating good user personas, the whole process of designing the application is made easier because it allows us to easily list goals and assumptions. All of these are based on the typical customers in our audience, allowing us to target satisfying these personas and stay on track.

3.1.1 Created Personas



Name: Grace Johnson

Occupation: 4th grade student in elementary school

Age: 9

Goals:

- Practice more math problems
- Get help understanding math problems she doesn't understand
- Interact with her classmates to see how they solve the problems
- Improve overall math skills

Needs:

- Something that is easy to use and understand
- Something that will keep her focused
- Something that she won't get bored of after a couple of minutes

Bio:

Grace is a fourth grader who's nine years old. She's currently learning basic math such as addition, multiplication, subtraction, and division. All of these topics are also integrated into word problems. Math isn't her strongest subject; she tends to get a lot of the problems wrong on her homework and is often lost on concepts being taught by her teacher during class.



Name: Jeremy Lee
Occupation: student in 6th grade
Age: 11

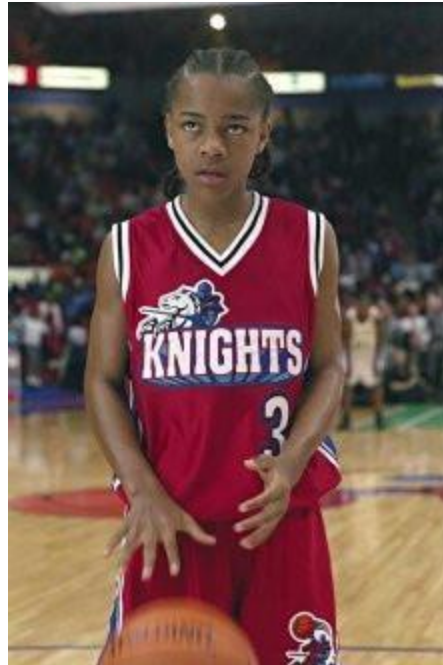
Goals:

- Do his homework quickly
- Make friends by collaboration
- Get familiar with the educational style of American

Needs:

- Easy to use interface
- Opportunity to involve the team-work and present himself to others

Bio: Jeremy is a 11 year old transfer student from China, he is not familiar with the new environment and new classmates. Math is one of the subjects Jeremy good at, and he is also smart on explaining math problems to others. However, Jeremy is not an outgoing kid, he feels lonely with the unfamiliar environment because of no friends, he expects to have a platform to let him make friends by working together.



Name: Calvin Cambridge

Occupation: student in 5th grade

Age: 10

Goals:

- Do the bare minimum to get at least a B- in his math class so he doesn't get grounded by his parents
- Have fun with his classmates and friends while doing math homework
- Do his homework quickly so he has time for video games and sports

Needs:

- Features that will let him and his parents track his grades more closely
- To be more involved in the collaborative work instead of just letting his classmates do all of the work so he can better understand the material
- A simple interface that won't frustrate him

Bio:

Calvin is a 10 year old 5th grade student who is much more interested in sports, video games, and other forms of entertainment than he is in school. This causes him to not put as much effort into school as he should and also causes his grades to slip severely from time to time. However, Calvin has proven to be capable of succeeding at school when he puts in the effort and is very good at working together with others since he is one of the most socially adept students in his class.



Name: Leo Gang

Age: 11

Occupation: a 5th grade student in an elementary school in Orange County

Gender: male

Home: Orange County, Santa Ana

Race: Asian American

Goals:

- Improve the academic behavior on Math
- Improve the interests in Math
- Improve the efficiency of Math assignments so that have more time on other courses

Needs:

- Can view the previous assignments so that have deeper impressions on problems that did wrongly before
- Easy operation and interface.

Bio:

Leo is a little boy born in north California, with a solid internet background. His parents are both software engineers who work for Microsoft in Silicon Valley. They usually have less time to do some actual help on Leo's academy. However, with the influence of his parents, Leo started to use computer and internet when he was only seven years old. The first product he used was Facebook, and then he started a lot of different social media. He often uses Facebook to chat with his mates. Leo is very good at physical science, especially Math. However, he found it hard to solve some problems when doing the homework because his parents were always not home. It usually takes a long time after he asks for a solution from his classmates on social media.

3.2 Mockups:

3.2.1 Signing In:

Math With Friends

Login

username

password

[forgot password?](#)

☒ remember username

[Sign up](#) [Sign in](#)

This is the first screen users see when launching Math With Friends. From this screen, users can sign in, sign up, or recover their password.

Main flow:

1. The user inputs their username
2. The user inputs their password **[A1: User has forgotten their password]**
3. The user hits the sign-in button
4. The system validates user information **[E1: Username & password do not match]**

Alternative flow:

Math With Friends

Forgot Password

email

confirmation code

new password

A1: User has forgotten their password

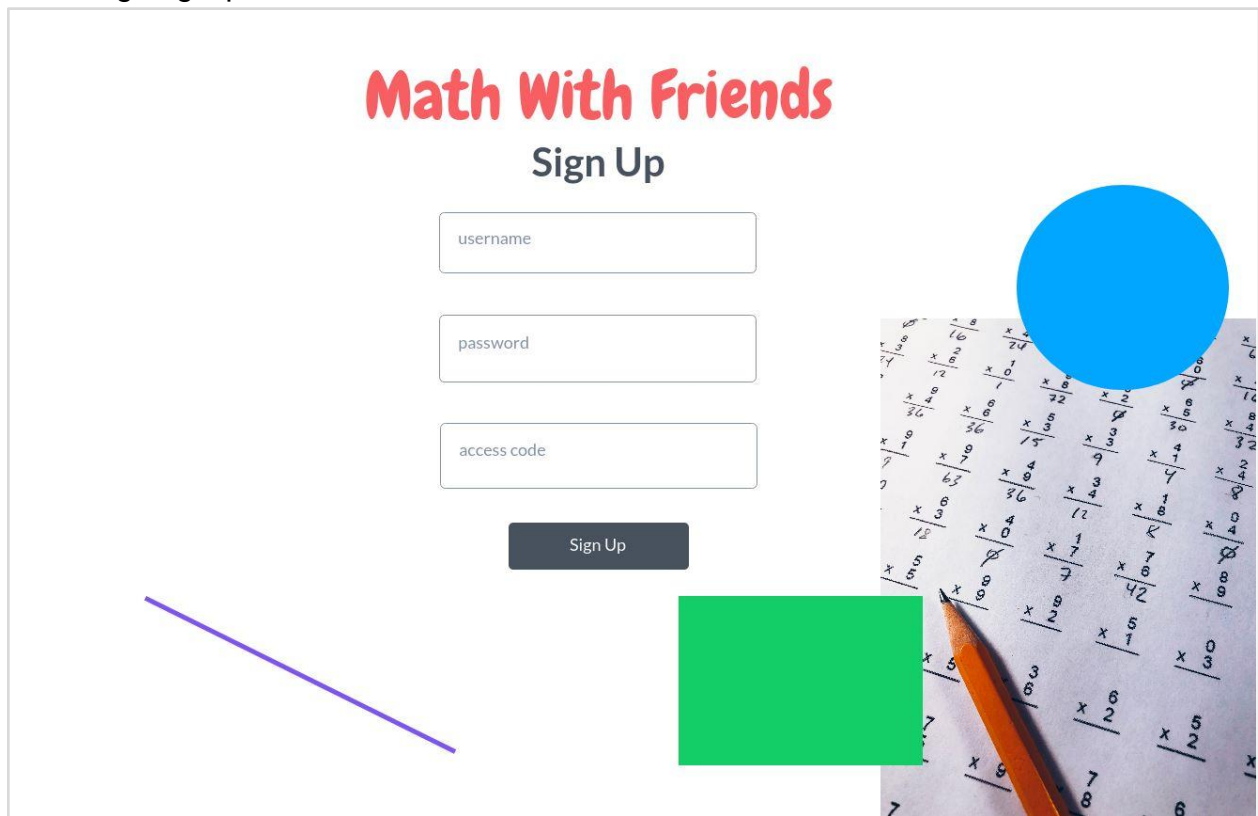
1. User hits forgot password
2. User inputs the email address they signed up with
3. System sends an email to the user with confirmation code
4. User inputs confirmation code
5. User inputs new password
6. System changes user's password

Exception flow:

E1: Username & password do not match

1. System cannot validate user information
2. System notifies user that the username or password do not match

3.2.2 Signing Up:



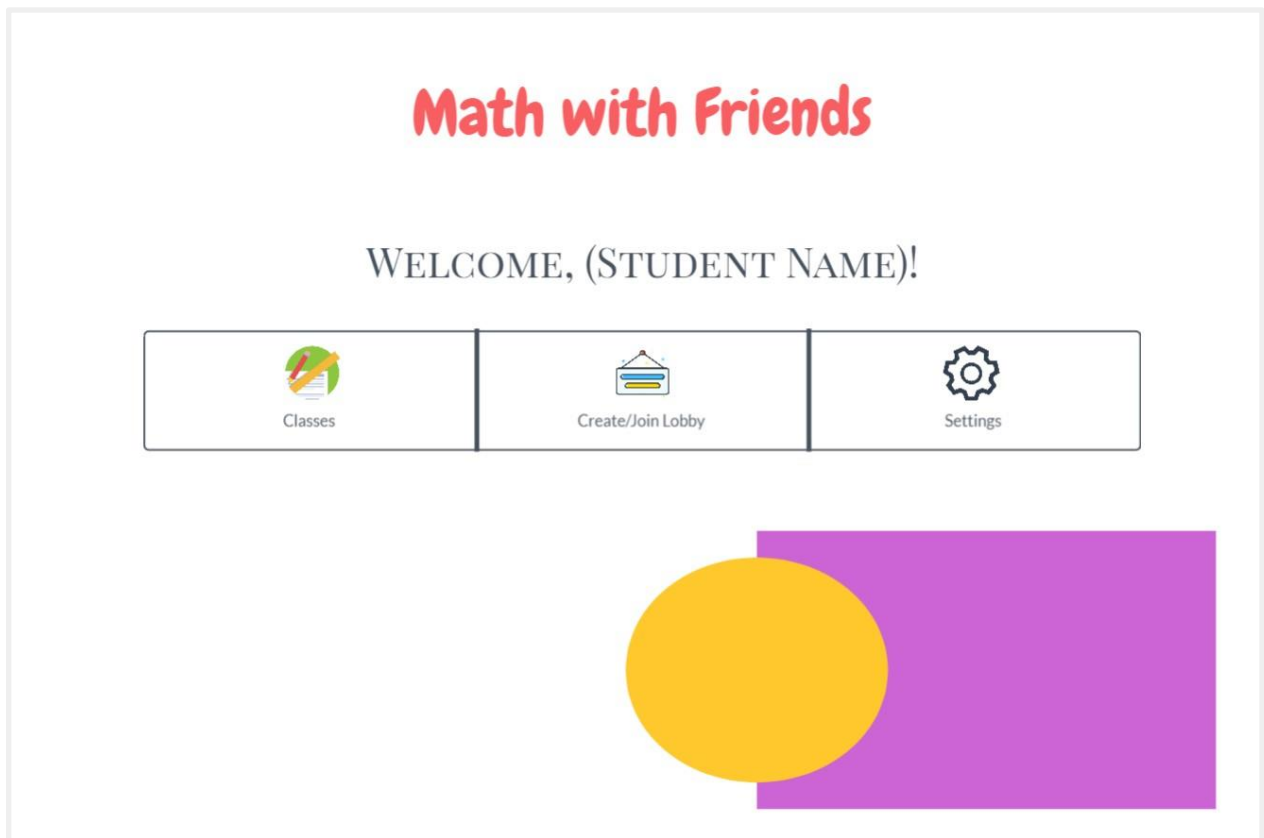
The image shows a 'Sign Up' screen for 'Math With Friends'. The title 'Math With Friends' is in red, and 'Sign Up' is in black. There are three input fields: 'username', 'password', and 'access code'. A dark grey 'Sign Up' button is below the fields. A blue circle is in the top right, a green square is in the bottom right, and a purple line is in the bottom left. The background features a math worksheet with multiplication problems and a pencil.

This screen is reached from the sign in screen when the green sign up button is pressed; this is for users who haven't created an account yet. On this screen, the user can input information for their account.

Main flow:

1. User inputs their desired username
2. User inputs their desired password
3. User inputs access code given by instructor
4. User hits sign up

3.3.3 Main Menu:



This is the main menu of Math with Friends. On this screen, there are three buttons the user can click on (“Classes”, “Create/Join Lobby”, and “Settings”) to navigate to the different pages respectively.

3.3.4: Create/Join a Lobby

Create/Join a Lobby

Select Class

Problem Sets

Invite

Create Lobby

Join:

Homework 5

This screen is reached from the main menu. Here the user can create a lobby or join an existing lobby which they have been invited to.

Main flow (creating a lobby):

1. User selects which class's assignment(s) the lobby will work on
2. User selects which problem set the lobby will work on
3. User selects which classmates to invite
4. User clicks on create lobby
5. System creates a lobby

Main flow (joining a lobby):

1. User selects which invited lobby they would like to join
2. System puts user into desired lobby

3.3.5: Lobby

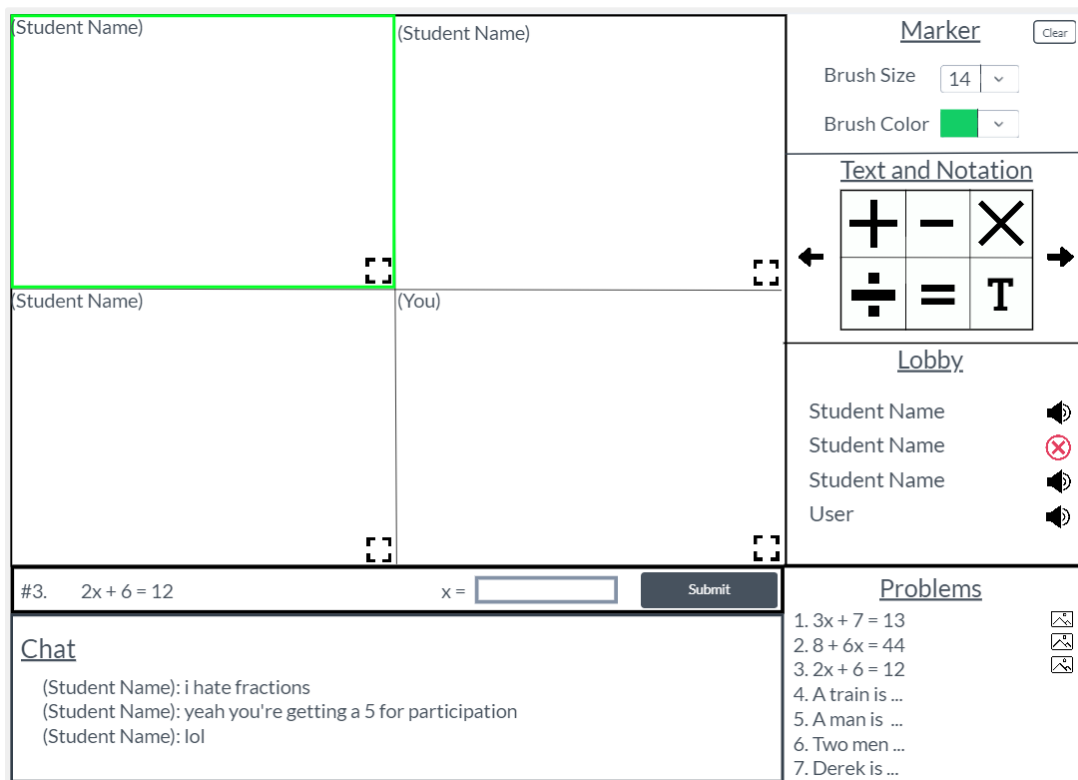
Lobby

You	Ready? <input checked="" type="checkbox"/>
Student #2 (Host)	Ready?
Student #3	Ready?

Start

This is the lobby screen. Here you can see all the Users in the lobby. Users in the lobby can press ready to indicate that they are ready to join and start working on the math problems. The host of the lobby can press the start button once everyone has indicated that they are ready.

3.3.6: Whiteboard



This screen is reached after the host of a lobby presses the start button.

Drawing:

On the user's respective Whiteboard, the user can use their mouse as a Marker by clicking on their Whiteboard. The user can fullscreen their whiteboard or any other lobby member's whiteboard by clicking the icon on the bottom right of the whiteboard. The user can manipulate their whiteboard when it's fullscreened. The marker's size and color can be adjusted in the top right under "Marker" by using the provided drop down menus labelled "Brush Size" and "Brush Color". The user can also drag and drop common mathematical symbols like addition, subtraction, multiplication, etc. from the right side under what's labelled "Text and Notation" onto their Whiteboard. The user can scroll through different symbols using the arrows provided.

Voice and Text Chat:

Users can type into the chatbox at the bottom left of the screen. Users can see messages that others in the lobby in the chatbox. Users can also communicate via voice by speaking into their connected microphone. Other users can be muted by clicking on the speaker icon next to the student's name in the lobby menu on the right of the screen.

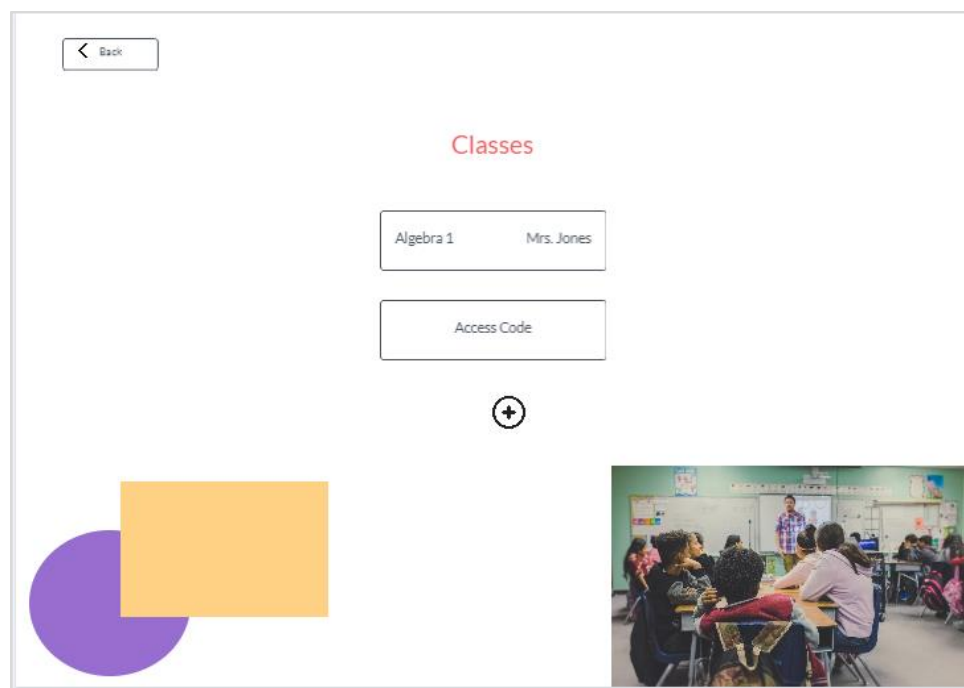
Viewing Problems:

The user can see the current problem below the whiteboards. Problems can be stepped through by clicking on the number in the “Problems” section at the bottom right of the screen. The whiteboard state can also be seen by clicking on the image icon next to the problem number.

Submitting Answer:

The user who’s in control will have their whiteboard highlighted green. The in control user can submit an answer by inputting it into the text box and hitting the submit button under the four whiteboards.

3.3.7: Classes



This screen is reached from the main menu. The classes page displays which classes the User has access to. The user can input an access code to enroll in a new class by entering the code and pressing the (+) button at the bottom.

3.3.8: Settings

The settings page is where users can go to adjust certain information as needed. It is divided into two categories: personal and communication.

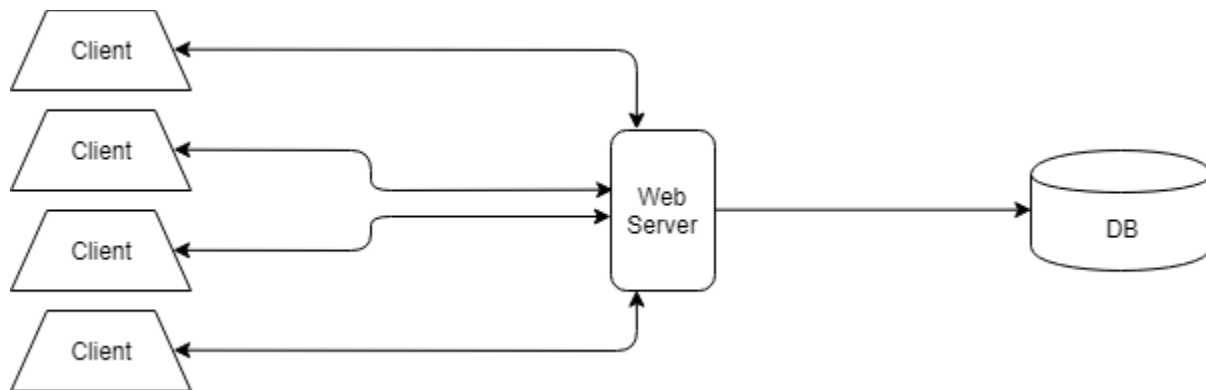
Adjusting personal settings:

1. User enters in their first and last name in the “Name” field
2. User enters desired password
3. User enters their email address

Adjusting communication settings:

1. User can use their mouse to drag and slide to their desired microphone output volume (how loud they want their microphone to be)
2. User can use their mouse to drag and slide to their desired volume (how loud they want the other users to be)
3. User can check “yes” or “no” for push to talk. If they check “yes” that means they have to press a button to be able to use their microphone, otherwise their microphone will always automatically be connected

4. Architecture Design



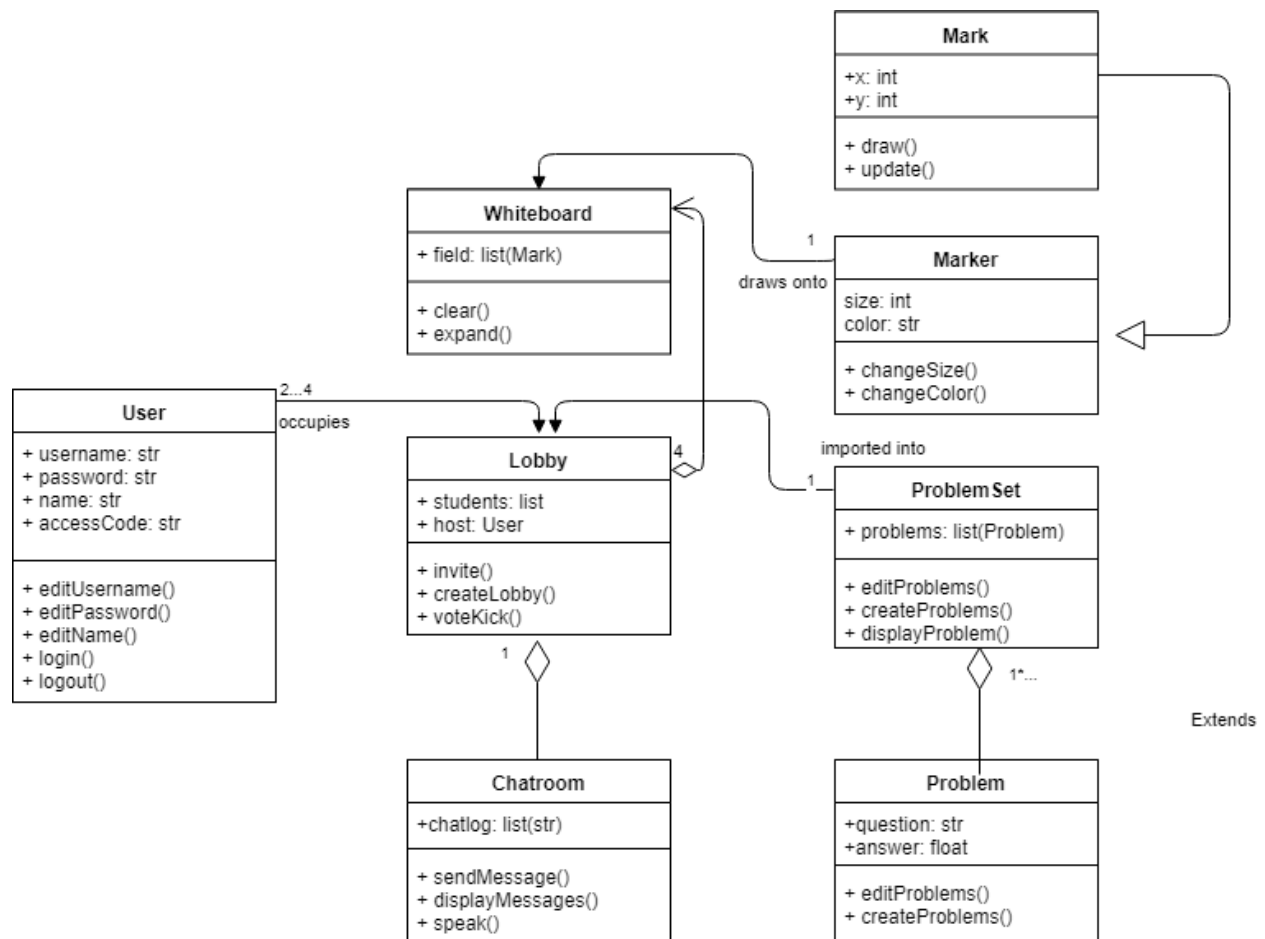
The architecture style used in Math with Friends is the client-server model.

The clients send out information such as whiteboard states from, any text messages that have been sent, and any voice input from users.

The web server relays data from one client to all other clients in a lobby so that they all see and hear the same things. The server also holds answers submitted for problem sets. The web server also handles login validation.

The database contains the students' information such as their name, username, password, and which class they are enrolled in. The database also contains all existing problem sets and previously submitted answers.

5. Implementation Design



*User = class, user = user of the system

UML Descriptions:

User:

- A User has a “username” and “password” so users can login and be differentiated by the system if there are users with the same name. User’s name is the user’s real life “name” intended to allow students to easily identify one another. User’s “accessCode” is required to be inputted when signing up for an account. An access code helps identify information such as which school the user goes to, what grade they’re in, what classes they are enrolled in, etc. The User class also has the methods `login()` and `logout()` to sign into their account and log out, respectively.

Whiteboard:

- The Whiteboard class represents the whiteboard that the users will use once they are inside a lobby. Whiteboard has one attribute, "field", which contains a list of Marks. The GUI handles creating a grid and displaying all Marks created by the Marker. The clear method turns field into an empty list, clearing all Marks from the Whiteboard. The expand method tells the GUI to fullscreen the selected Whiteboard

Marker:

- The marker class has a size and color attribute where users can change the size and color of the marker respectively. The changeSize() and changeColor() methods allow the marker size and color to be changed, respectively.

Mark:

- The mark class has an x and y attribute where the mark is placed in the size and color that the mark inherits from the marker class. It also has a draw method which places the marks on the whiteboard at the mark's x and y coordinates.

Lobby:

- The Lobby class represents a room which is occupied by "students". One of the methods is createLobby() which is done by the "host", the creator of the Lobby. Methods include invite(), which allows Users in a lobby to invite other Users.

Chatroom:

- The Chatroom class serves as the main communication medium between Users in a given Lobby. Chatrooms save "chatlog", which is a list of all the text messages sent by the Users. Methods include sendMessage() which is where the Users are able to send text messages back and forth, displayMessages() which allow them to see the conversation, and speak() which enables them to use voice chat.

ProblemSet:

- The ProblemSet class has the attribute "problem" which is a list of problems that are imported into the Lobby. The Lobby iterates through the ProblemSet, displaying the Problems.

Problem:

- The Problem class is an aggregation of the ProblemSet class. This class represents math problems to be solved. Each Problem has a "question" and an "answer". Methods include editProblems() which allow changes to be made to the current problems being displayed and createProblems() which allow the Users to create their own problems to practice with.

6. Details

This section contains information such as when our meetings were, what was accomplished during each meeting, who worked on which aspects of this design studio, and pictures/notes from our meetings.

6.1 What Each Member Did

Member Name	Tasks Worked On
Yu Fan	Helped brainstorm classes, mockups, and architecture Created one persona Part of the mockups Wrote persona introduction
Karen Nguyen	Part of the UML diagram and description Part of the interaction design (mockups) Finalizing the design document One persona The description for the architecture design Part of the application design
Patrick Tran	Helped brainstorm classes, mockups, and architecture Wrote introduction, description Part of table of contents Part of the UML class diagram and description Part of the mockups Created one (unused) persona Created architecture diagram
Tyler Valdivieso	Created the Calvin Cambridge persona Part of the final whiteboard mockup Drew the initial mockups drafts on a whiteboard Wrote part of the application design Wrote part of the UML class descriptions
Linxuan Xin	Created one persona UML class analysis Brainstormed the user interface design, mockups and architecture

	Part of whiteboards design, settings, lobby, main menu and login system design
--	--

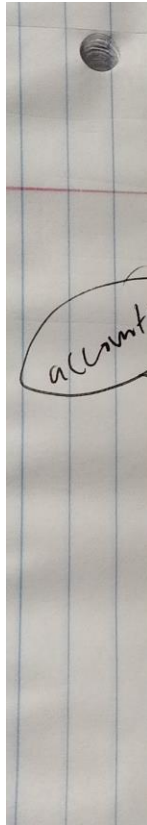
6.2 Meeting Log:

Date	Who Showed Up	What We Did Summary
March 2nd	Yu, Karen, Patrick, Tyler	<ul style="list-style-type: none"> - Brainstormed about general overview of application - Thought of goals, assumptions, constraints - Came up with a lot of ideas for features of the application
March 6th	Everyone	<ul style="list-style-type: none"> - Talked about more ideas - Created UML diagrams and discussed each feature of each class in detail - Mindmap
March 8th	Everyone	<ul style="list-style-type: none"> - Came up with more ideas for the UML diagram - Worked on the interaction design portion by drawing mockups on the whiteboard, discussing what the features are
March 12th (Online)	Karen, Patrick	<ul style="list-style-type: none"> - Redid UML Diagram - Added Introduction, Table of Contents,

		Class Descriptions - Finished most mockups
March 13th	Everyone	- Finished designing mockups
March 15th	Everyone	- Worked on architecture design and more final touches to the design document
March 16th	Patrick, Yu, Karen, Tyler	- Worked on application design, made changes to our UML diagram

6.3 Pictures of Our Meeting

Meeting 1

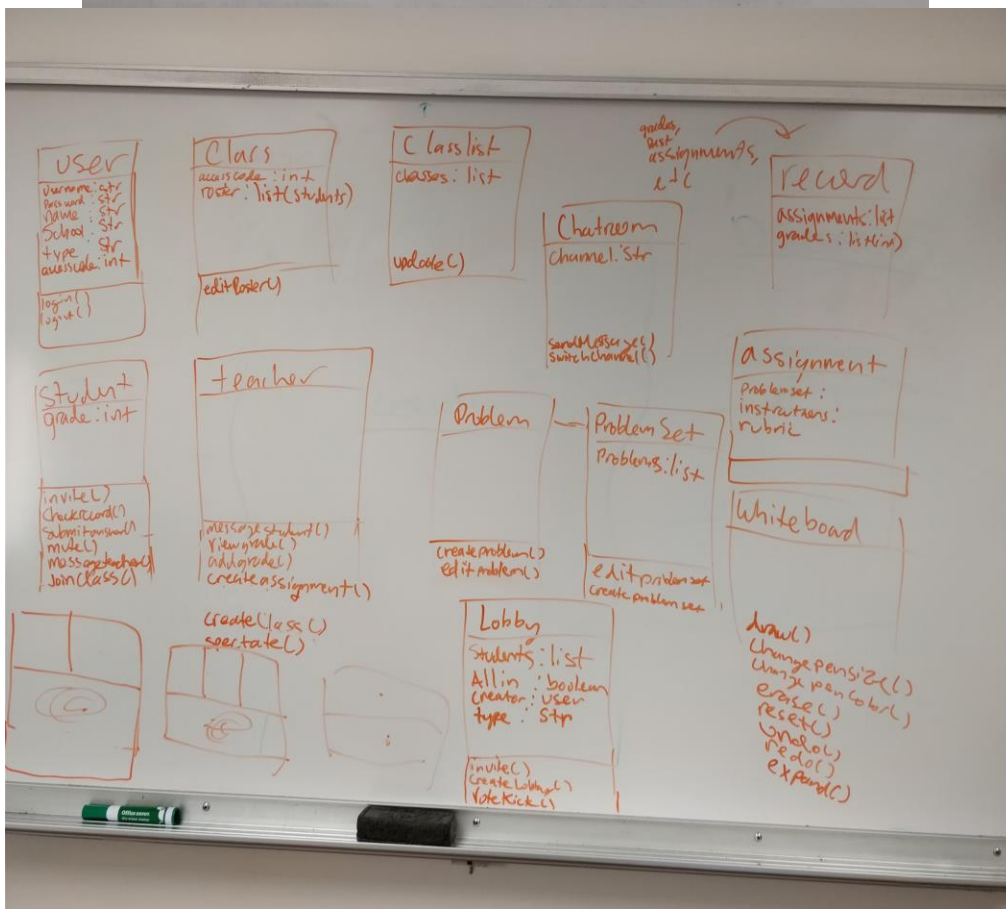


Me

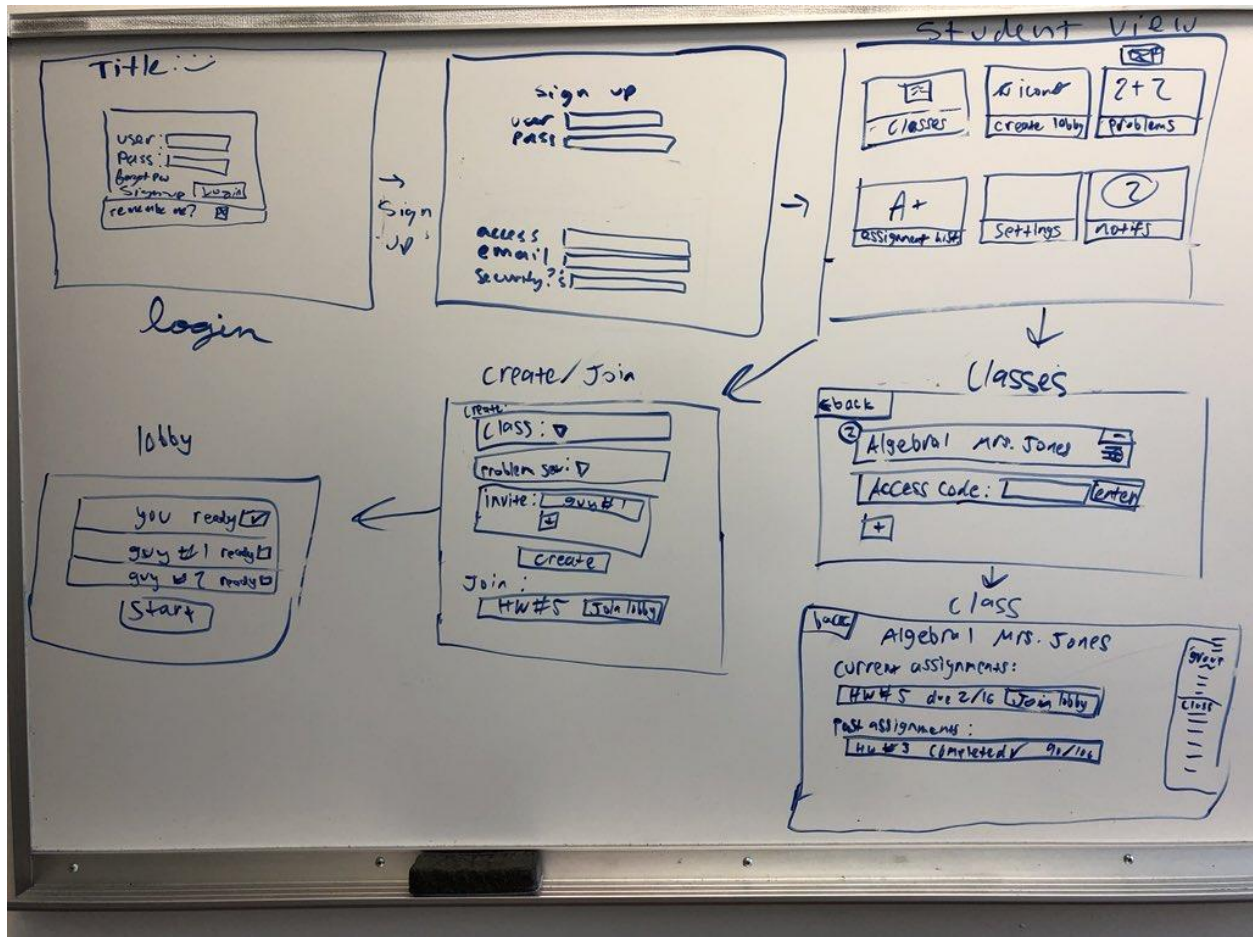
- A final report containing the design and your motivations for it.
- The company has heard that 'design methods' can help, and would like you to apply those that seem applicable to this design problem.

Whiteboard)

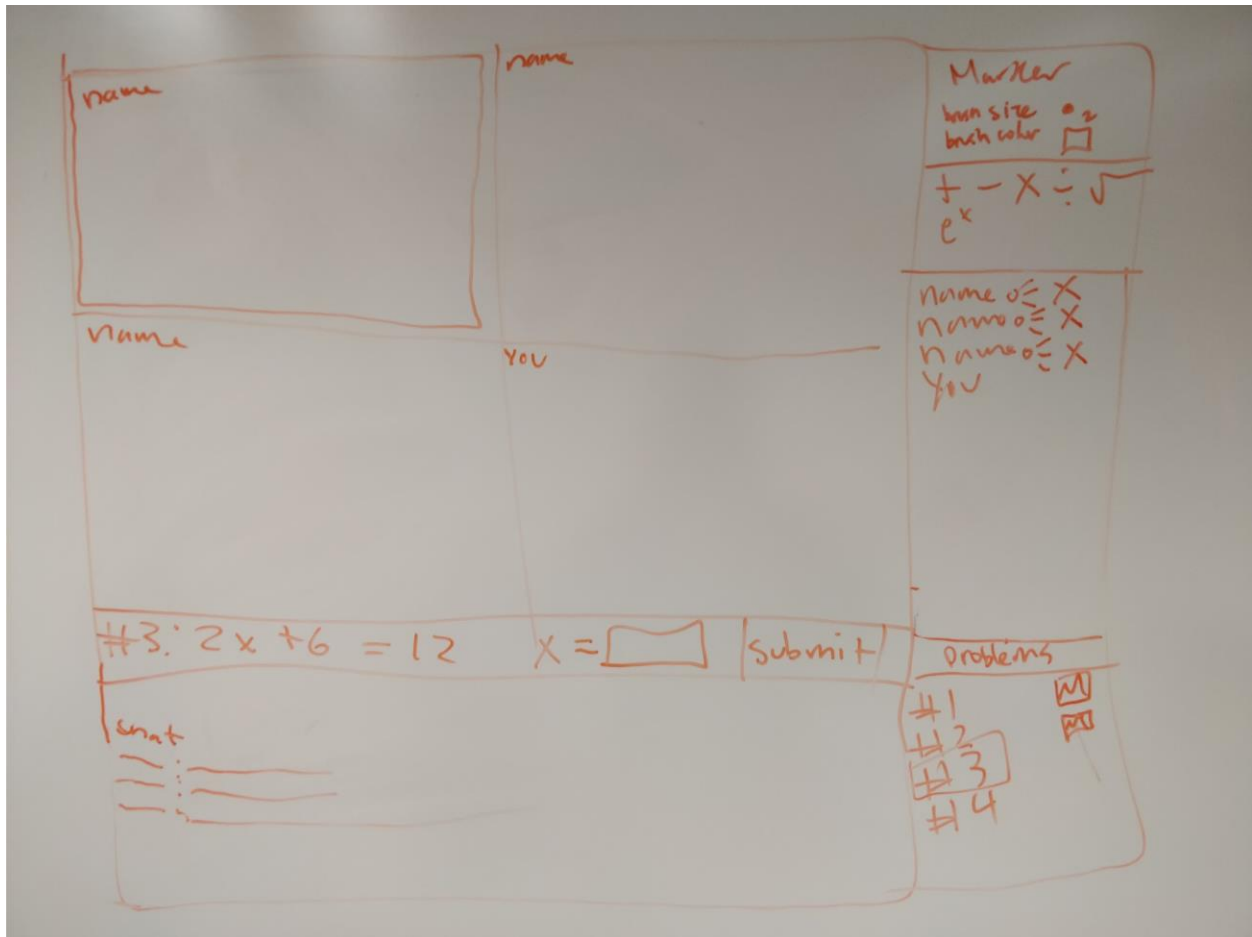
ap)



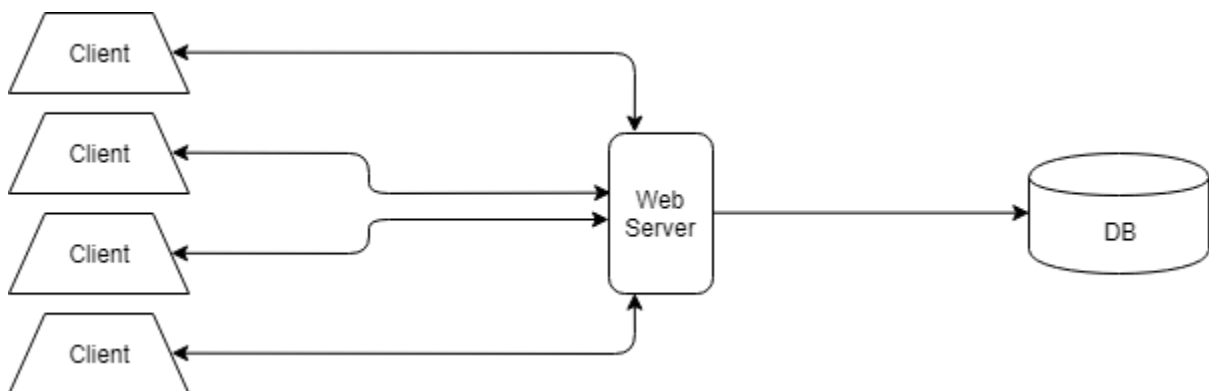
Meeting 2 (UML Class Diagram)



Meeting 3 (Mockups)



Meeting 4 (Finalized Mockup of Whiteboard)



Meeting 5 (Architecture Design)