Software Design Document

Joseph Cao, Jacob Wyner, Kyle Wright, Anthony Chan

1. Introduction

Classroom Connect is an intuitive and reliable software solution which serves to bridge the current gap between students and faculty in the same space through real time polling. This software, designed to bridge the gap between students and classroom faculty, allows for direct communication between participants.

In its current form, the Classroom Connect software aims to be designed as a web service, rather than a native mobile or desktop application. This design choice provides an easy way to ensure accessibility across most platforms. Classroom connect does not intend to connect users outside of the physical classroom setting. Direct interactions are only possible when users are connected to the same virtual classroom instance.

2. Design Considerations

One important consideration facing the Classroom Connect design team is the method that should be used to authenticate and differentiate users. Due to the nature of the software, high security should not be necessary, however it is important to determine the proper user group (what class) each user is in, as well as make sure that user input is only accounted for once.

Another important consideration in the design of Classroom Connect is determining how much information will be available to the student role in the software. While it may serve useful to allow other students to view poll results the same way the faculty role can, this may not always be the preferred method of classroom faculty.

2.1 Assumptions

Due to the live polling nature of the Classroom connect software, use of the software must adhere to relatively strict environmental restrictions. An active internet connection must be maintained at all times throughout the use of the software. The user must be enrolled and attending the specific classroom they are providing poll results for.

Assumptions are made that the student body and classroom faculty alike share a common good will about our softwares presence in the classroom. Classroom Connect software is not designed to, nor will it, work in an environment where one or both parties do not want it to be used.

2.2 Constraints

One major constraint of the Classroom Connect software is the necessity to verify a student's authenticity in the classroom. In order to make sure that students who are not enrolled in the class do not participate in the class's poll results, a classroom faculty member will most likely have to provide Classroom Connect software with a list of student user's university email addresses. While this is a necessary step in user authentication, it

is a constraint as it does not align with our goal of streamlining the service so that it is not a burden on either party.

2.3 System Environment

The user must maintain an active internet connection throughout the use of the Classroom Connect software. Any system capable of obtaining an internet connection and supporting a modern web browser and common development packages will have the ability to run Classroom Connect.

3. Architectural Design

Classroom connect provides users with real time polling analytics, bridging the student to lecturer communication gap more and more with every application. In order to provide such a service, several initial user functionalities and components are necessary:

- User Dashboard
- Mobile Friendly Interface
- Poll Instantiation
- Poll Search
- Poll Visualization
- Poll User Feedback

3.1 Overview

ClassRoom Connect relies on the general basis of requests and responses, as many other services provide such as Facebook, Twitter, and Moodle. In short, the service is held within a large data bay - the server. Within the server, there are two main compartments - the Database, and the Web Framework / Front End. The role of the Database is to store, organize, normalize, protect, and provide all data which is received. The role of the Web Framework is to act as an interface for users to interact with the server in a simple & intuitive fashion. The Front End Interface automates and simplifies the transactions that the user needs to perform in order to receive, create, and modify data. Below is an illustration of ClassRoom Connect's decomposition of the high-level system:
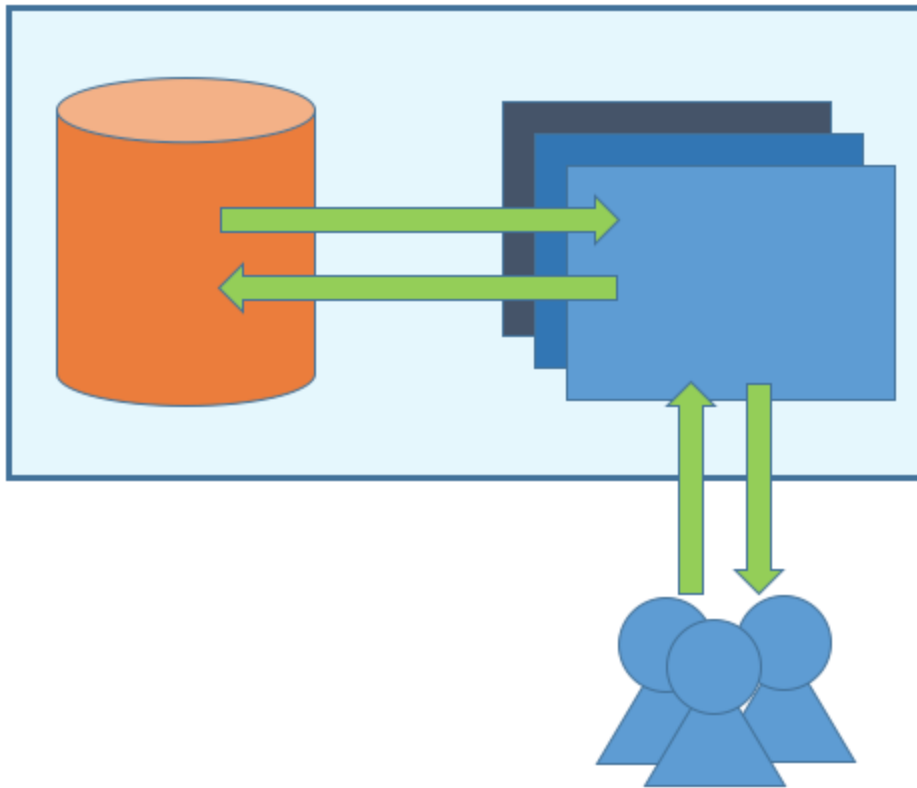
Figure 3.1.0: High Level Structure, *ClassRoom Connect*

Structure / Component	Function
Server (RECTANGLE)	- Store Database, Front End - Handle Client / DB Transactions - Domain Hosting
Database (ORANGE CYLINDER)	- Store normalized tables - Receive requests, push responses - Protect from unauthorized requests
Client Side / Front End Framework (PEOPLE)	- Provide UI/UX Interface for users - Handle user requests, DB responses'

Table 3.1.0T: Service Structure & Function

3.2 Rationale

The design of the high overview structure was primarily guided by the past experience from previous internships dealing with & working alongside Back End Developers. It was understood that modularizing and

normalizing data into parsable, sortable tables and objects will provide the best possibility for customization, scalability, and portability of the service. In short, this structure is both well defined and clearly organized, and should prove to be durable architecture. Although extremely broad, the essence of splitting Model, View, and Control are there. In addition, further research suggests that separating the storage and data from the front end enables for overall simpler and more general requests and responses, allowing for extremely customizable data analytics and visualizations. Overall, the design is general enough for customization for future vanities of the service, however complete enough for comparison to other working services following the model.

3.3 Conceptual (or Logical) View

To elaborate on the previous section, this service is provided over the internet via 'cloud' storage and hosting. Users interact to the website through the internet, receiving mobile-friendly interfaces which receive and send data to and from the database. NOTE: both the website and the database are essentially part of the internet component, however hold so much value in the project that isolation and description of each are necessary and beneficial to the understanding of the connections. Below is a more thorough diagram which helps show the connections made at this deeper level of understanding:
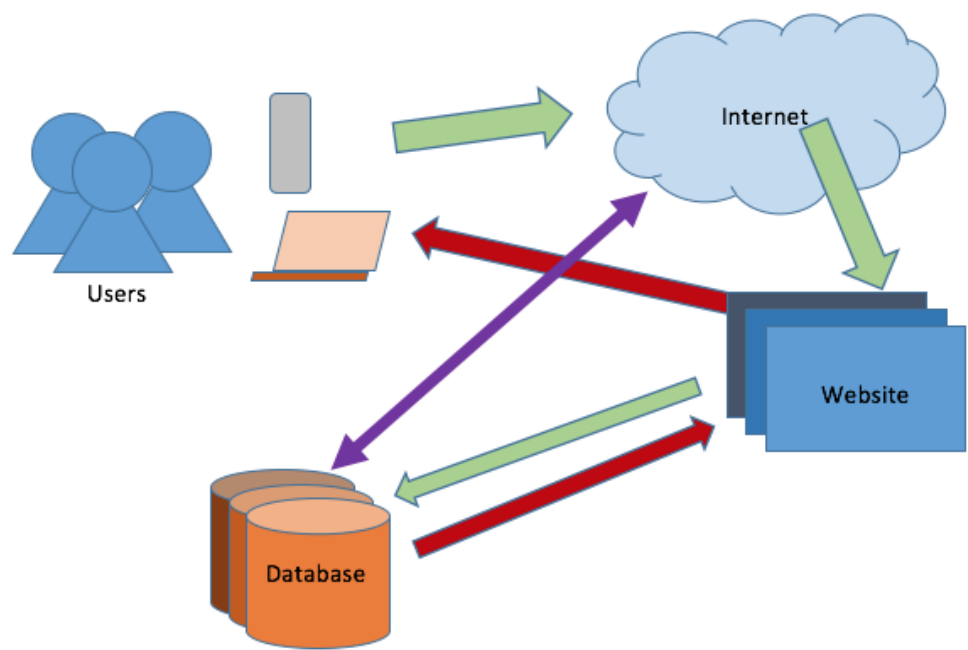
Figure 3.3.0: Component & Deployment Diagram, *ClassRoom Connect*

	Users	Internet	Website	Database
Users	Poll initiator, Poll member	Access Website	Access, Create, Modify Poll Data	X

	communication via Website			
Internet	-	-	Host Domain, Enable access to users	Sign In Authentication (GOOGLE)
Website	-	-	Website traversal (unique pages & analytics)	Pass data from form fields, requests/response
Database	X	-	-	Sorting, Normalizing, Parsing Data Received and Transmitted

Table 3.3.0T: Connection Table Plot

*Note: All levels of interaction are via Requests and Response from X to Y, and are a Bi-Directional relationship; X or Y can do both the request and the response.

4. Low Level Design

Section 4 of the Software Design Document focuses on the elaboration of the functionality, structure, and connections of the Server, and its Databases and Website Architecture. The database is comprised of several tables of normalized data: USERS, POLLS, MARKERS, GROUPS (potentially). Each table will hold unique fields of information, referencing each other table as need be. The goal is for a system of limited redundancy in data, to prevent from duplication and pre-modification update errors (updates occurring in some systems for a specific table, but not being updated in other areas).

4.1 Class Diagram

ClassRoom Connect relies on OO design, and with each, there are helper classes which assist in the creation, modification, and storage of the object classes. Below is a UML class diagram of the database & necessary data sets. Please note that this is an intermediate design - the flow and structure shown below may change after further research and implementations of this diagram.
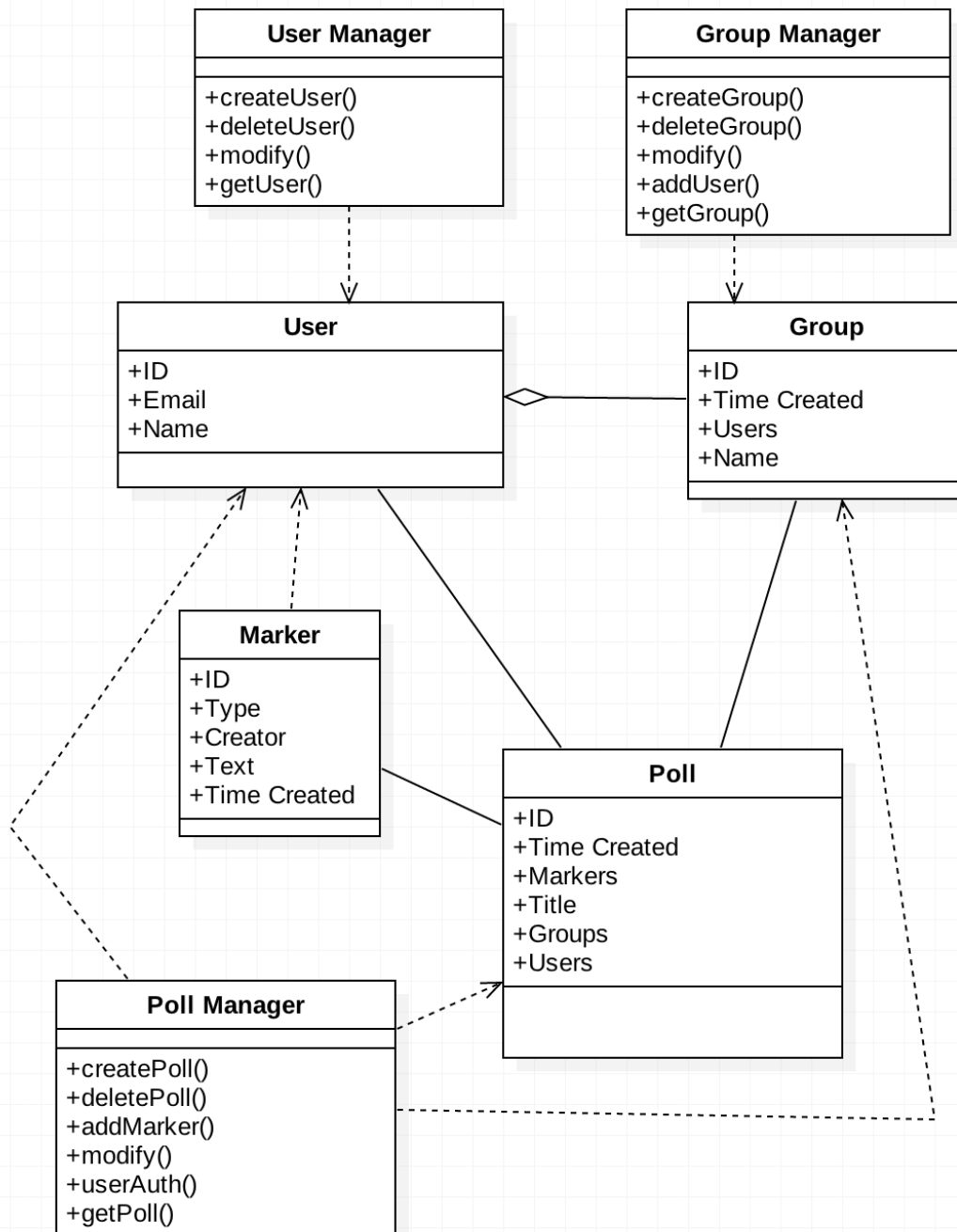
Figure 4.1.0: UML Class Diagram, *ClassRoom Connect*

4.2 Sequence Diagram

This diagram shows details of how and in which order objects within *ClassRoom Connect* interact with one another.
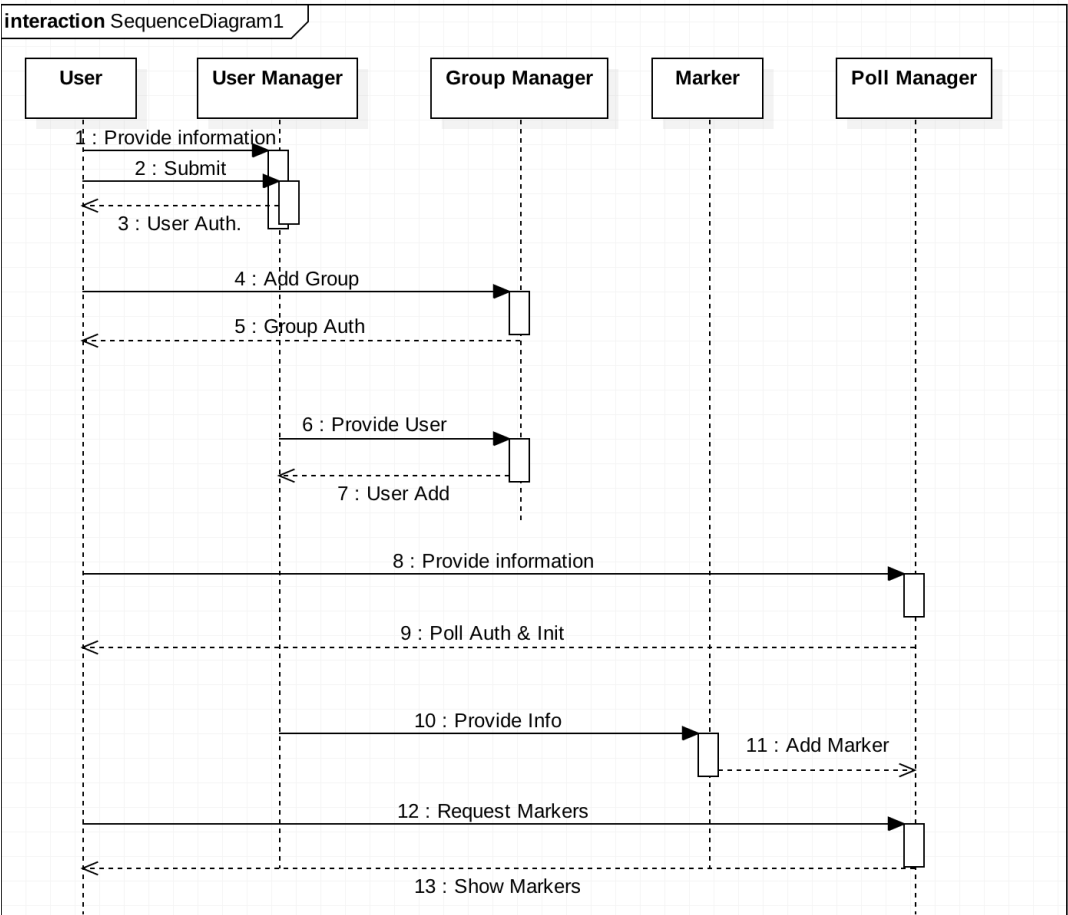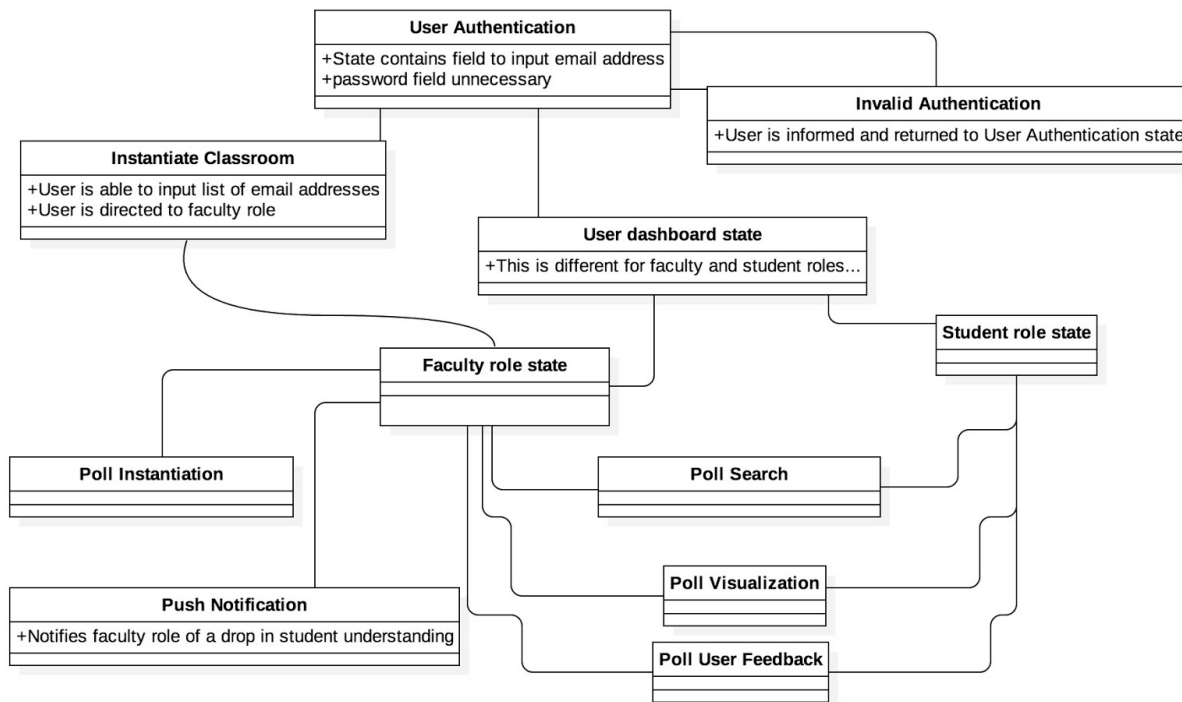

Figure 4.2.0: Sequence Diagram, *ClassRoom Connect*

4.3 State Diagram

5. User Interface Design

This section should provide the specific format/layout of the user interface of the system.