

Class.ly: Helping Students Collaborate
Requirement Documents
CS 3892 Human-Computer Interaction
Vanderbilt University Fall 2015

By: Parker Klein & Mitchell Masia

I pledge my honor that I have neither given nor received aid on this work.

Mitchell J. Masia & Parker A. Klein

I. Class.ly Mission Statement

Class.ly decreases time and effort necessary to gather contact information while increasing communication and collaboration among Vanderbilt undergraduate students. Class.ly automatically creates chatrooms and helps schedule working sessions without the need to even create an account. Future versions of Class.ly will provide interfaces for university officials and professors to encourage open communication and statistics tracking about collaboration.

II. User Profiles

A. Stakeholders

1. Vanderbilt University Undergraduate Students (Primary User)

The key stakeholder for Class.ly is Vanderbilt University undergraduate students. The application is designed and built primarily to help facilitate group work among these users, and their adoption is absolutely critical to the success of the system. The features of the app, chat and scheduling, rely on a high adoption rate among potential users. A chatroom with a class full of students is exponentially more valuable than one with just a small group. These students are generally highly adept with mobile technology- the primary platform for Class.ly. A very small percentage of the population is physically or mentally impaired. Additionally, the application is designed with “Americanized” design standards in mind: clear, blocked text, tabbed navigation, and bright, colorful displays that encourage specific user interactions and tasks most pertinent for students.

a) Context Of Use

Motivation:	Discretionary	Mandatory		
Use Frequency:	Non-User	Infrequent	Frequent	
User Category:	Beginner	Novice	Expert	
Task Nature:	Mission Critical	Calm		
Interaction:	Continuous	Intermittent		
Social Env.:	Public	Private		
	Work	Entertainment		
	Synchronous	Asynchronous	Both	
Physical Env.:	Indoor	Outdoor	Both	
Auditory Env.:	Quiet	Medium	Loud	All
Visual Env.:	Dark	Ambient	Bright	All
Tech. Env.:	Networked	Isolated		
	Wired	Wireless		
	Intranet	Extranet	Internet	
	Fixed	Mobile		

b) Cognitive Ability

Education:	High School	Undergraduate	Graduate
Computer Lit.:	Low	Medium	High
Mobile Lit.:	Low	Medium	High
Typing Skill:	Low	Medium	High
Learning Style:	Auditory	Visual	Linguistic

2. Vanderbilt University Registrar Personnel (Secondary User)

The Vanderbilt University Registrar personnel are the next largest key stakeholder in the Class.ly application. This is for two major reasons: data security concerns and usage statistics tracking. Legislation under the Family Educational Rights and Privacy Act (FERPA) and the Protection of Pupil Rights Rights Amendment (PPRA) make it extremely difficult for educational institutions to create and release non-anonymized student information to online educational services, mobile applications, and other web-based software platforms (1). This means that incorporating Class.ly with enrollemnt information from YES must be an extremely secure process, Class.ly must not have full database priveliges to the back-end of YES, and individual students must be able to opt out of having their information shared with the 3rd-party application. This makes the registrar's office a key stakeholder because they are the system administrators in this case, and must ensure their actions comply with both federal and state privacy regulations.

Despite the pains of dealing with security regulations, the Vanderbilt University Registrar and Vanderbilt as a whole has much to gain by incorporating Class.ly into the enrollement system. A feature, not included in the Class.ly MVP, will be meta-statistics

tracking for school administrators. This means that registrar personnel can acquire visual insights into how students communicate about their classes and view meetup and collaboration statistics about their students. This will greatly help in compiling reports about student activities across all schools and departments at Vanderbilt. Additionally, highlighting this information in info sessions can be a potential huge draw for prospective students who are prone to collaboration and enjoy working with others. Class.ly will allow administrators to track the actions of students like never before and facilitate them to gain insights about collaboration that was never possible before.

Unlike undergraduate students, these administrative users will use the application far less frequently, mainly for report compilation. Their experience in computing environments is likely limited to basic tasks, using guides and protocols. They generally work in stationary (desk), indoor, private conditions, as opposed to students who will likely be using Class.ly to communicate even on the short walk from class to class. To deal with the stationary environment, and lack of mobile computing literacy, the creators of Class.ly utilize the write-once, run-everywhere development paradigm. Using the “Ionic Framework” for development, the creators are able to write one application that can be run on a desktop computer, iOS device, or Android device. On all three mediums, the application is highly usable, with a clear visual interface, and colors that indicate suggested actions in the context of each user.

As administrators do not need the same capabilities as student users of Class.ly, and they are more likely to have visual impairments, the actions they can take in the app are severely limited. Instead of having access to a chatroom and scheduler, they are prompted instead to see compiled views of statistics on a per-class, per-student, and chronological basis. The buttons are generally larger, and with far fewer action options, they are less likely to encounter “consumer’s paralysis” and be confused or nervous about what actions to take. In this way, the creators of Class.ly have constructed the app such that it behaves differently, and responds in the context of the user.

a) Context Of Use

Motivation:	Discretionary	Mandatory		
Use Frequency:	Non-User	Infrequent	Frequent	
User Category:	Beginner	Novice	Expert	
Task Nature:	Mission Critical	Calm		
Interaction:	Continuous	Intermittent		
Social Env.:	Public	Private		
	Work	Entertainment		
	Synchronous	Asynchronous	Both	
Physical Env.:	Indoor	Outdoor	Both	
Auditory Env.:	Quiet	Medium	Loud	All
Visual Env.:	Dark	Ambient	Bright	All
Tech. Env.:	Networked	Isolated		
	Wired	Wireless		
	Intranet	Extranet	Internet	
	Fixed	Mobile		

b) Cognitive Ability

Education:	High School	Undergraduate	Graduate
Computer Lit.:	Low	Medium	High
Mobile Lit.:	Low	Medium	High
Desktop Lit.:	Low	Medium	High
Typing Skill:	Low	Medium	High
Learning Style:	Auditory	Visual	Linguistic

3. Vanderbilt University Professors & Faculty (Secondary User)

Part of Class.ly's objective is also to increase communication between students and their professors. As opposed to normal email communication, Class.ly creators plan to encourage this by providing professors with an administrative interface to the chatroom and scheduling features, as well as a limited feature set of statistic tracking tools. Because the professors are likely to more technically inclined than registrar officials, but still behind students, they are offered a very intuitive interface, with just a few advanced capabilities. In this admin interface, they are able to send instant messages to the whole classroom via the chatroom, as well as delete messages in the chatroom that they deem inappropriate. Contrarily, they do not have access to the private group chatrooms created by students for their individual groups projects by default. They can be given access if the groups elects to do so. Additionally, in the meetup scheduler, professors can post their own office hours, and comment on notes taken by students about a past meetup. They will not, however, be able to delete or reschedule meetups that they

did not personally create. This interface will be limited, and not implemented in the Class.ly MVP prototypes presented in the HCI class, but would be part of a true system implementation.

Similar to registrar personnel, professors are likely to be much more sporadic users of the system, with less computing knowledge than the primary target users (students). Because of this, they too will likely use the desktop version of the application on their personal computer, in a more private setting, likely not on a short walk across campus. Additionally, professors will be allowed a small suite of statistics tools to track collaboration in their own classes. They will not have the student-based, and meta-view available to registrar users, but will only have access to statistics about each class they are personally responsible for. Their usage of Class.ly is entirely discretionary, and is rivaled by other communication systems like Blackboard or Piazza, but with the intuitive chatroom interface, professors will have a much more immediate, “unofficial” open communication line with their students.

a) Context Of Use

Motivation:	Discretionary	Mandatory		
Use Frequency:	Non-User	Infrequent	Frequent	
User Category:	Beginner	Novice	Expert	
Task Nature:	Mission Critical	Calm		
Interaction:	Continuous	Intermittent		
Social Env.:	Public	Private		
	Work	Entertainment		
	Synchronous	Asynchronous	Both	
Physical Env.:	Indoor	Outdoor	Both	
Auditory Env.:	Quiet	Medium	Loud	All
Visual Env.:	Dark	Ambient	Bright	All
Tech. Env.:	Networked	Isolated		
	Wired	Wireless		
	Intranet	Extranet	Internet	
	Fixed	Mobile		

b) Cognitive Ability

Education:	High School	Undergraduate	Graduate
Computer Lit.:	Low	Medium	High
Mobile Lit.:	Low	Medium	High
Desktop Lit.:	Low	Medium	High
Typing Skill:	Low	Medium	High
Learning Style:	Auditory	Visual	Linguistic

III. System Input/Output

A. Inputs

1. Enrollment Data

Enrollment data from the Vanderbilt University YES system is the driving input which will facilitate the functionality of Class.ly. By pulling data on individual students, classes, and faculty, Class.ly is able to associate this information in order to create chatrooms, help schedule meetups, and provide statistics. The data will be consumed by an externally-facing back-end data ingestion interface capable of receiving and parsing data in a format-agnostic way for extensibility.

2. User Actions

The user and their actions is the mechanism by which Class.ly capabilities are used. The user creates messages, adding value to chatrooms, creates custom chatrooms, schedules meetups, and tracks collaboration statistics. Not all of these functionalities are available to all of the user profiles; however, none of these functions, nor the application provide value without user input to the system. Class.ly is solely meant to create a collaborative environment, but it is the users that must act in order for the environment to be useful.

3. Computing Environment

The last major input to the Class.ly application is a valid computing environment in which the application can run. Whether the environment is fixed (desktop) or mobile (phone), the environment itself must have a power supply, high-speed internet

connectivity, an operating system to schedule computing tasks, and a processor to execute instructions. Because of Class.ly's architecture (using the write-once, run everywhere paradigm), the computing environment does not have to provide specialized software such as a Java Virtual Machine to run the application; however, it only must be able to create a web browsing context.

B. Outputs

1. Chat

Upon logging into Class.ly with appropriate Vanderbilt credentials, the system (automatically seeded with YES data), will identify the user. Based on the context of the user (student, faculty, registrar personnel), the application will automatically perform a set of tasks. In the case of a student, Class.ly will automatically create chatrooms on a per-class basis, based on the enrolled classes of the student user. All students in their individual classes, using Class.ly, will be automatically added to the chat. In the case of the professor, that user will be provided an administrative interface to the class chatroom.

2. Scheduler

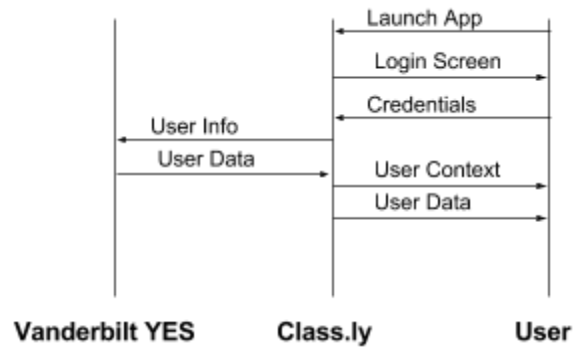
Mimicking the process of creating a chatroom, if Class.ly identifies the user as being a student or professor for a given class, both of these users will be provided an interface to the meetup scheduler. Each student will be able to schedule, edit, and take notes about a given meetup. The professor will be able to create a special meetup, often referred to as office hours, which they will attend, and students can come visit them for assistance.

3. Statistics

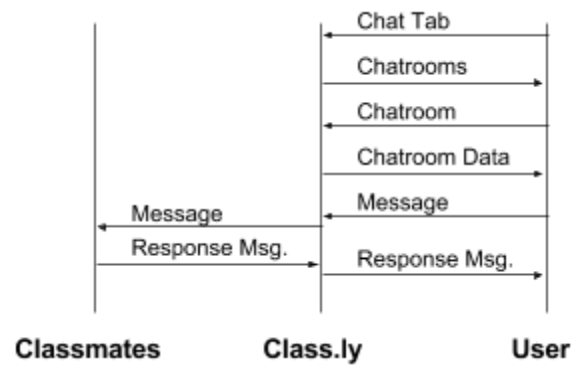
If a professor or registrar personnel chooses to use Class.ly, they will be provided a "statistics" interface. In the case of the professor, they will be able to view statistics about Class.ly usage among students in the classes they are directly responsible for. Initially, they will be able to gauge individual student participation and meetup attendance. Contrarily, the registrar personnel will have the option to view these same statistics, as well as those for the entire student population. The output here will be a nicely formatted visualization displaying how student and faculty users of Class.ly are interacting and collaborating.

IV. Usage Scenarios (I/O Traces)

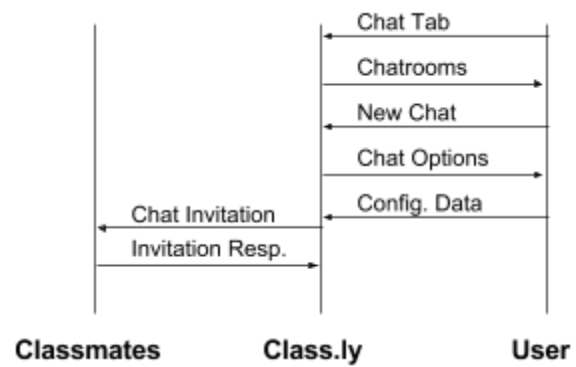
A. Registration



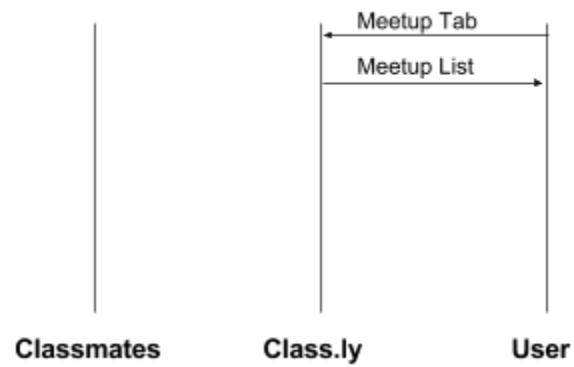
B. Chat



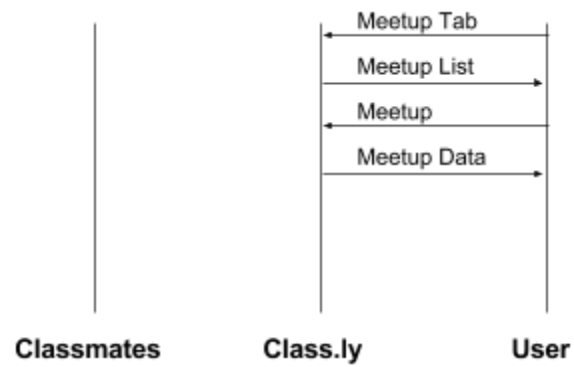
C. Create A Custom Chat



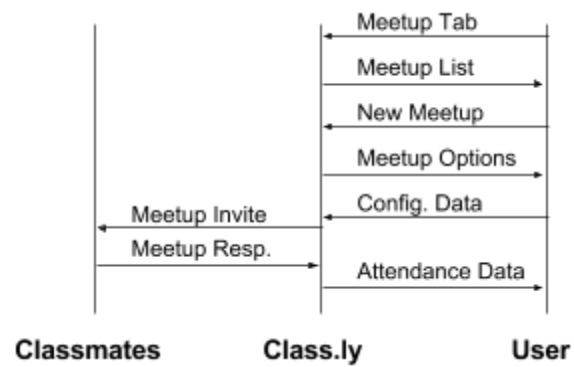
D. View Meetup List



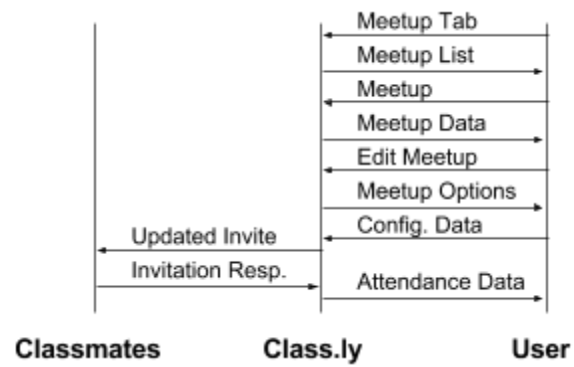
E. View Individual Meetup



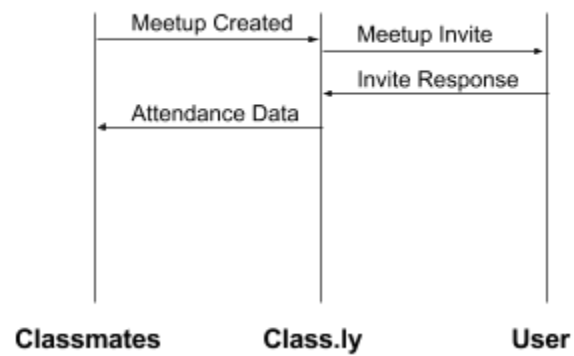
F. Create A Meetup



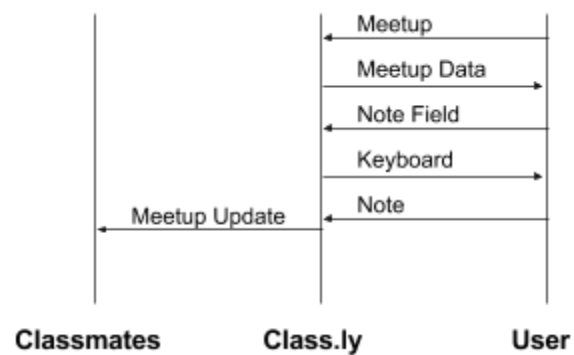
G. Edit A Meetup



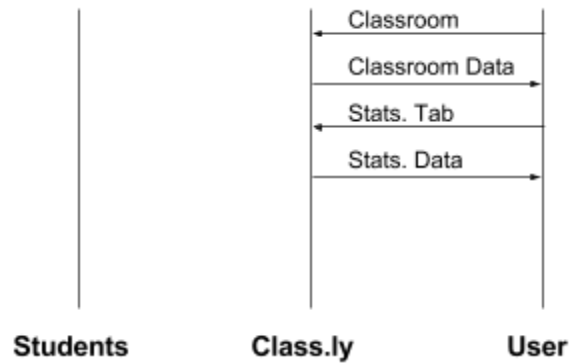
H. Attend A Meetup



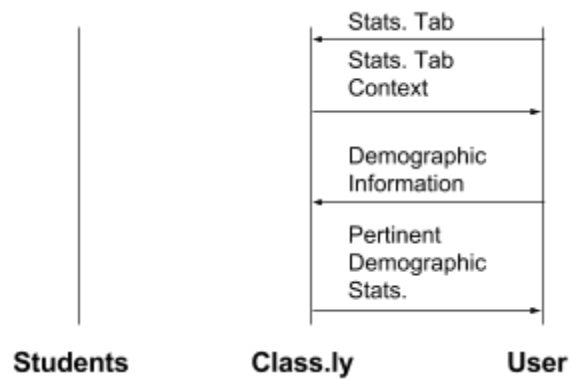
I. Note A Meetup



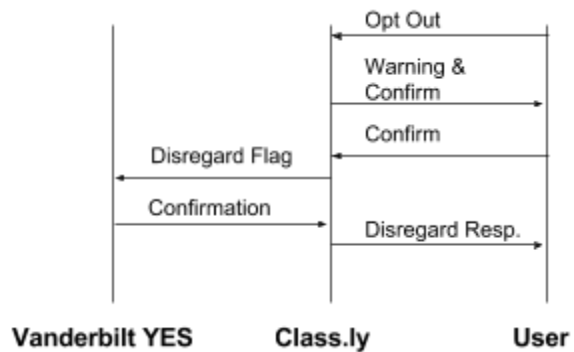
J. View Classroom Statistics (Professors & Registrar Personnel)



K. View Meta Statistics (Registrar Personnel)



L. Opt Out



V. System Requirements

NOTE: Not all requirements will be met in functional prototypes. The prototypes are primarily to demonstrate the chat and meetup interfaces. Additional features meeting requirements can be implemented in later versions of Class.ly.

#01 -- Non-Functional

Description: The system will be able to pull enrollment data from Vanderbilt YES.

Rationale: Data from YES links students to classrooms and is vital in automatically creating communication lines with the correct groups of people.

Acceptance Criteria: Class.ly can make requests to YES and accept data in JSON or XML format.

Dependencies:

Externally accessible YES API to request data.
Data security is up to FERPA standards.

#02 -- Non-Functional

Description: The system will link between individual students and their classes.

Rationale: A trace between students and their classes is essential in forming chatrooms and notifying the correct students.

Acceptance Criteria:

Given a student, Class.ly can find all of their classes.

Given a class, Class.ly can find all enrolled students.

Dependencies:

Student contact data has primary/foreign key combination to provide linkage with class enrollment.

#03 -- Non-Functional

Description: The system implements a communication protocol.

Rationale: The chat feature necessitates that messages can be sent in an “instant message” type fashion.

Acceptance Criteria:

Given an email, Class.ly can abstract communication details to appear as an instant message.

Dependencies:

Fully reliable instant message wrapper.

#04 -- Non-Functional

Description: The system can identify the context of a user as student, professor, or registrar personnel.

Rationale: To provide different users different capabilities and interfaces in order to perform designated operations.

Acceptance Criteria:

Given user credentials, the system can identify the type of user.

Dependencies:

Contextual information in personnel records.

#05 -- Context Of Use

Description: The system implements and serves different interfaces depending on the identified user role.

Rationale: To provide different interfaces to people in different user roles.

Acceptance Criteria:

Different interface served based on the user type.

Dependencies:

Multiple interface implementations.

Flag indicating the user type and appropriate context.

#06 -- Functional

Description: The system automatically creates a chatroom for the current user.

Rationale: To avoid collecting contact information and facilitate communication.

Acceptance Criteria:

Upon login, the chatroom is automatically created.

Dependencies:

Linking user and class.

Instant message implementation

#07 -- Functional

Description: The system automatically creates a meetup scheduler interface for the current user.

Rationale: To facilitate meetup creation.

Acceptance Criteria:

Upon login, the meetup scheduling interface is available.

Dependencies:

Linking user and class.

#08 -- Non-Functional

Description: Class.ly repulls class information on a fresh restart during the add/drop period.

Rationale: To account for students who have dropped or enrolled in the class late.

Acceptance Criteria:

Class.ly pulls class information when opened during the add/drop period.

Dependencies:

Recognize the add/drop period.

Limited bandwidth, high load on YES.

#09 -- Functional

Description: Class.ly has a side drawer.

Rationale: To navigate to high-level options such as log-out and opt-out.

Acceptance Criteria:

Class.ly has a working side navigation drawer with pertinent options.

Dependencies:

#10 -- Functional

Description: Class.ly implements a tabbed navigational panel.

Rationale: To navigate between the main functions in the app given a specific user context.

Acceptance Criteria:

Working tabbed navigation

Intuitive tab icons

Indication to show the current tabbed state.

Dependencies:

Tab controller.

Proper icons.

Specific user context.

#11 -- Functional (User Requirement)

Description: Class.ly chatroom acts responsively.

Rationale: To help users realize their actions are effective.

Acceptance Criteria:

Quick load times.

Send message progress bar.

Dependencies:

Instant message implementation.

#12 -- Functional (User Requirement)

Description: Class.ly meetup scheduler is responsive.

Rationale: To help users realize their actions are effective.

Acceptance Criteria:

Quick load times.

Modals for config. information.

Dependencies:

#13 -- Functional (Usability Requirement)

Description: Class.ly classes, chatrooms, and meetups are easily selectable.

Rationale: To help users perform their desired actions.

Acceptance Criteria:

Easily selectable, separated buttons.

Clean, crisp lines.

Colors indicate possible desired actions.

Responsiveness prevails.

Dependencies:

Ionic Framework Styling Guides

iOS & Google Play Style Guides

#14 -- Functional

Description: Class.ly allows users to opt out of the service and “forget” their data.

Rationale: To aid users in acquiring control over their personal data and abiding by FERPA regulations regarding privacy.

Acceptance Criteria:

Easy to locate “Opt Out” options.

App will delete data if option selected.

Dependencies:

App does not have full CRUD access to YES data.

App can clear all traces of data from database.

#15 -- Functional

Description: Class.ly allows users to create custom chat rooms.

Rationale: To aid users in working on custom group projects while continuing to use Class.ly.

Acceptance Criteria:

Easy to locate create new chatrooms.

Easy to delete and edit chatrooms.

Creator of chatroom has admin privileges.

Dependencies:

Custom group projects.

#16 -- Functional

Description: Class.ly allows professors and registrar personnel access to collaboration statistics.

Rationale: Huge value add for administrators to adopt usage of the application.

Acceptance Criteria:

Easy to generate statistics reports.

Easy to export statistics reports.

Specific statistics available based on the user role.

Dependencies:

Proper activity logging on student user activities.

#17 -- Environmental

Description: Class.ly allows users to work on desktop and mobile environments.

Rationale: To cater to the skills and preferences of student and administrative users.

Acceptance Criteria:

Clean, intuitive interfaces everywhere.

Accessible by domain name or runnable application.

Dependencies:

Ionic Framework (Write once, run everywhere)

Apache Cordova (Wrapper to run on Android X)

VI. Updated Class.ly Project Schedule (No substantial changes)

COMPLETE

ONGOING

REMOVED

Date	Class Requirements	Class.ly Requirements
9/10	Initial Project Description Documents	Initial Project Description Documents
9/14		Wireframe Prototypes (Draw.io Wireframes)
9/24	Requirements Documents	Requirement Documents App Development Begins (Ionic Framework)
10/1	Data Analysis Results, Prototyping Plan	Data Analysis Results, Prototyping Plan
10/6		Mock Data Complete (In Non-Relational Database) Back-end Prototype Complete (Database, API)
10/8		Front-end Prototype Complete (Ionic Framework, Bootstrap Style)
10/13	Prototype Demonstration	Prototype Demonstration
10/22	User Test Plan	User Test Plan
11/30	Project and Supporting Documentation	Project and Supporting Documentation
12/10	Partner Evaluation	Partner Evaluation

The documentation for CS3892 Human-Computer Interaction, as well as the source code for Class.ly is publicly available online in a Git repository. The repository can be found here: <https://github.com/masiamj/CS3892-HCI-Class.ly>.

Bibliography

1. Department Releases New Guidance on Protecting Student Privacy While Using Online Educational Services. (2014, February 25). Retrieved September 20, 2015.