**PROJECT ARTIFACTS**

**FOR**

**CODEWORD GENERATOR**

**(ANGULAR)**

**Table of Contents**

Introduction 3

Methodology 3

Requirement 4

User Stories 5

Design 10

ER-Diagram 15

Tools and Technologies 16

Test Cases 18

Result and Analysis of Test Cases 18

Deployment 18

**Introduction:**

In this project Administrator anonymous surveys the students in there course at the beginning and end of the semester. To improve the quality of the data collected, administrator would also like to be able to associate the initial and final surveys of a student (while maintaining anonymity.) Administrator’s low-tech solution is to provide each student before the survey with a slip of paper that has a code word on it. Students enter the code word into the initial survey and mail it to themselves. They find the email and retrieve their code word before the end survey. They enter the code word into the end survey. The anonymity of responses is maintained, and administrator can associate the initial and end surveys. Here each student gets a code word, which is random, anonymous distribution of code words. Further, other professors also administer surveys and this application may also be of use to them.

**Methodology:**

Any MEAN stack project needs attentive consideration and result evaluation in the context it is used because the extracted knowledge is significant to assist the decision-making process in an application. Below is the web application methodology, which we as a team have followed, to develop application to randomly assign Codeword to students for taking surveys.

Starting for the bottom up, we followed each step-in order to find the sentiment of text based on a tweet.



We attained this methodology through:

1. **Daily standup meetings**

The standup meetings are meetings in which all the team members discuss about “what they have done yesterday, what they are going to do today and what impediments did they go through”. By doing this, the team gets to know till what extent the project has been accomplished and what are the remaining aspects to be completed in the project. We have done standup meeting every day in the class in order to discuss the complete and incomplete aspects of the project. For better and clear understanding, we have also used google docs to maintain the details of standup meeting.

1. **Weekly client meetings**

The client meetings are the meetings in which team discusses their progress regarding the project with the client. By doing this, the client gets to know till what extent the project has been accomplished and provides suggestions regarding any required necessary changes. We have been meeting the client every Wednesday in order to discuss our progress and took suggestions and feedback from the client so that we can further work on the project and make necessary required improvements.

**Requirements:**

There will be a web app with roles for: (1) administrators, (2) professors, (3) students. Students will be able to retrieve their code word for a specific course survey. A Student will be able get repeated access to the code word for a course over an extended period. Professors will be able to set an expiration date (generally the last day of the term). This can default to 4 months from the date of creation. Professors will be able to create a list of code words to be assigned to students in a course. Professors will be able to anonymously assign code words to students. Professors will be able to monitor that students in a class have received a code word, but not be able to determine what the code word is. Professors will be able to use the application to assign code words for concurrent surveys in different courses, and different sections active at the same time. Therefore, students may have multiple code words during a term. Administrators will be able to manage all information, including professor surveys, student lists, sets of code words. Security must be addressed. The app must work on an iPhone SE.

**User Stories:**

MVP 1:

1.01 As a user, I want to register by providing my email ID and password so that I can access my account.

1. User has entered the username, password and confirmed the password in the registration page.
2. If user is an instructor, the Instructor box is checked
3. User clicks on ‘Register Now’ button in the registration screen
4. User should be able to redirect to the Login page

1.02 As a user, I want to log in with my Email Id and password.

1. User has entered correct username and password in the login screen
2. User click on ‘Sign in’ button in the login screen
3. User should be able to see the home screen with codewords

1.03 - As a user, if I forget the password, I want to retrieve my account by entering the registered Email ID and submit it. Later I can access my account with the temporary password provided.

1. Invalid login results will display the error message as “Password wrong”
2. User clicks on ‘Forget password’, is asked for email address to send password reset link

MVP 2:

2.01 As an instructor, I want to check an "Instructor?" box when I register so that I can distribute code words.

1. Should be able to View the Login page
2. Should be able to Click on Sign Up/Register and should be able to view the Registration page
3. Register himself/herself by filling information like Name, Email id, Password, Confirm password and checking the Instructor checkbox
4. After clicking on “Register Now” button, he/she should be able to view a message “Successfully Registered and Redirecting to Sign In page”

2.02 As an instructor, I want to see my list of active courses

1. After filling information in Sign In page and clicking on Sign In button, he/she should be able to view a message “Sign In successful. Redirecting to Instructor’s Dashboard”
2. After successful Sign In, he/she should be able to view Instructor’s dashboard where they can view all the courses he/she are added to along with list of active courses.
3. After clicking on courses, instructor should be able to view course details like Course Name and Code-words assigned for respective course

2.03 As an instructor, I want to create a new set of code words by entering a unique name for the codeword set and uploading a plain text file that contains a new list of codewords.

1. After successfully login, he/she should be able to view his/her Email Id (used for signing in) on top right corner of navigation bar along with dropdown arrow symbol for change password and logout options
2. Once clicking on Code-word Set in Navigation bar, he/she should be able to view “Create Code-word Set” page
3. Instructor should be able to view 2 default Codewordset (Small & Large set) along with count of codewords in the sets
4. Once clicking on Code-word Set button at top left corner of the page, he/she should be able to view "Create Code-word Set” modal
5. Post filling Codeword set Name and uploading excel file containing codewords in it, Instructor should be able to view number of codewords present in the uploaded set.
6. After filling all the required fields and clicking on Create Codewordset button, Instructor should be able to view the new created codeword set along with count of codewords and Show Details button in the codewordset list
7. Once clicking on Show details button, Instructor should be redirected to Codeword page.

2.04 As an instructor, I want the app to always have my Basic small codeword set (with 90 unique codewords) and a second Large Codeword Set (with 200 unique codewords) and instructors are not allowed to delete them.

1. After logging In as Instructor, he/she should be able to have all the Instructor's privileges and view all the options related to Instructor's dashboard.
2. After successful Sign In of Instructor, he/she should be able to select the codeword set from the codeword set dropdown menu.
3. At the time of codeword selection, Instructor should be able to view Basic Small and Basic Large code-word sets in the dropdown list but they should not have any access to delete those code-word sets

2.05 As an instructor, I want to create a new course by adding a list of students (containing one \*.edu email & one name field for each student), selecting a codeword set from a drop-down that shows the name of the set with the calculated count in parenthesis, and setting the start date (default to today) and end date (defaulted to 4 Months after then start date) for the course, and a pre-survey URL, and a post-survey URL. Team: implement either cut & paste a tab or comma separated list of students OR offer to import by selecting a csv file or Excel file in the proper format).

1. User should be able to add student’s lists by uploading names and Email Id.
2. User should be able to select codeword set from dropdown list.
3. User should be able to add new course
4. And also able to set start date and end date for the course
5. User should able to enter Survey Start URL
6. User should able to enter Survey End URL

2.06 As an instructor, if I click on “View Students” I want to see a list of students for the selected course (emails and names) and want to see the calculated count of students in that course.

1. If user clicks on “View Students”, list of students for the selected course should be displayed.
2. Email address and names should be included while displaying the list of students.
3. User can see the calculated count of the students in the selected course.

2.07 As an instructor, when I view my list of active courses, for each active course, I want to see:

* 1. The unique name,
  2. The codeword set name with the calculated count of words,
  3. The start date and the end date,
  4. The pre-survey URL and the post-survey URL
  5. An option to “Edit”,
  6. An option to “Delete” an active course (expired courses will be not appear)

1. User should be able to see a unique name for the codeword set.
2. User should be able to see the codeword set name which is selected from dropdown list.
3. User should also able to see the calculated count of words in the set by selecting a code word set from dropdown box.
4. User should be able to see the Start date and End date for courses
5. User must be able to see pre-survey URL and the post-survey URL
6. User must be able to see the edit option to edit the student names and Email ID
7. User must be able to see delete option to delete an active course.

2.08 As an instructor, I want one codeword from the specified set to be randomly assigned to only one student and each student must have a unique codeword in that course and each codeword can only be issued once in each course (or not used at all) when a course is created.

1. User should be able to see the unique codeword for each student in every course
2. when course is created user should be able to see the unique codeword in each course.

MVP 3:

3.01 As a student, I want to see my courses and see/acknowledge/get my random codeword for the respective course on my home page.

1. User should be able to see the list of courses assigned to him/her.
2. User should be able to see the Random codeword which is displayed on the home page.

3.02 As an instructor, I want to see my list of courses, those are “active” today between the start and end date inclusively.

1. User should be able to see the list of his/her courses.
2. User should be able to see the courses which are active today.

3.03 As an instructor, I want to be notified if the codeword set is too small. If the calculated count of codewords is less than the calculated count of students then I want to see a warning that says “You have n students, but the codeword set has only m words.

1. When the codeword doesn’t not meet the requirements, user should be notified.
2. User gets a warning "You have n students, but the codeword set has only m words" when count of codewords are less that the count of students.

3.04 As an instructor, I want the codewords confidential and must not know or be able to see the code word each student gets.

1. User should not know the codeword when each student gets.
2. Codewords should be confidential.

3.05 As a user, I want to navigate between multiple pages (Home (list of active courses), + New Course, View a Set of Codewords and Confirm Delete Course).

1. User should be able to navigate easily between the pages.
2. When clicked on "+ Add course", it should redirect to the page where we can add the course details.
3. When clicked on "View Students", it should redirect to the page where it has the course details and students assigned to that particular course.
4. When clicked on "Delete course", user should get a pop up asking for a confirmation.

3.06 As an instructor, I want to delete an existing course once it is completed.

1. User should be able see the delete button for each course.
2. When clicked on delete button, existing course should be deleted.

3.07 As an instructor, I want the percentage of students who have accessed their codewords in that course.

1. When student access their codewords for the course, it should calculate the percentage.
2. When user clicks on "View students", it should show the percentage of students.

3.08 Each course is limited to no more than 1000 students.

3.09 As an Instructor, while creating codeword set, If there is more than 1200 codewords in the file which is getting uploaded to create codeword set, Instructor should be notified with a pop up message that "Each codeword set has a limit of up to 1200 codewords".

1. After successfully login, Instructor should be able to view Codeword Set list where he/she should be able to view all the codeword set created by him/her along with the two default codeword set
2. After clicking on Create Codeword Set button at the top left corner of the page and post filling the required fields for codewordset creation and post uploading the file containing codewordset, Instructor should be able to view actual count of codewords in the uploaded file
3. After filling all the required fields and clicking on Create codeword set button, If there is more than 1200 codewords in the file which was uploaded, Instructor should be notified with a pop up message that "Each codeword set has a limit of upto 1200 codewords".

3.10 - Registration is limited to 25 teachers.

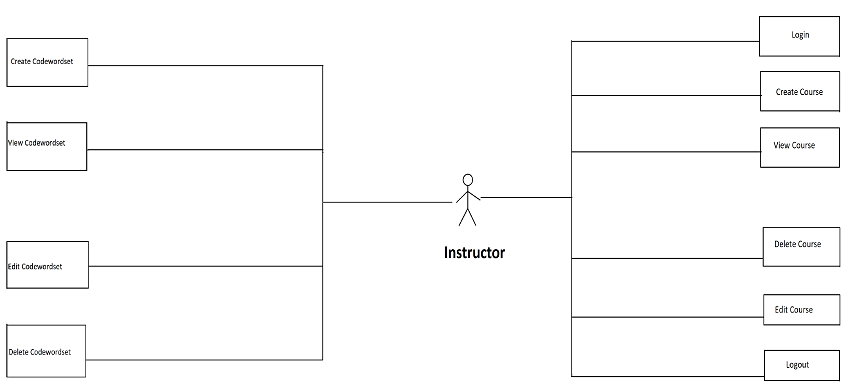
MVP 4

Administrator Functionality

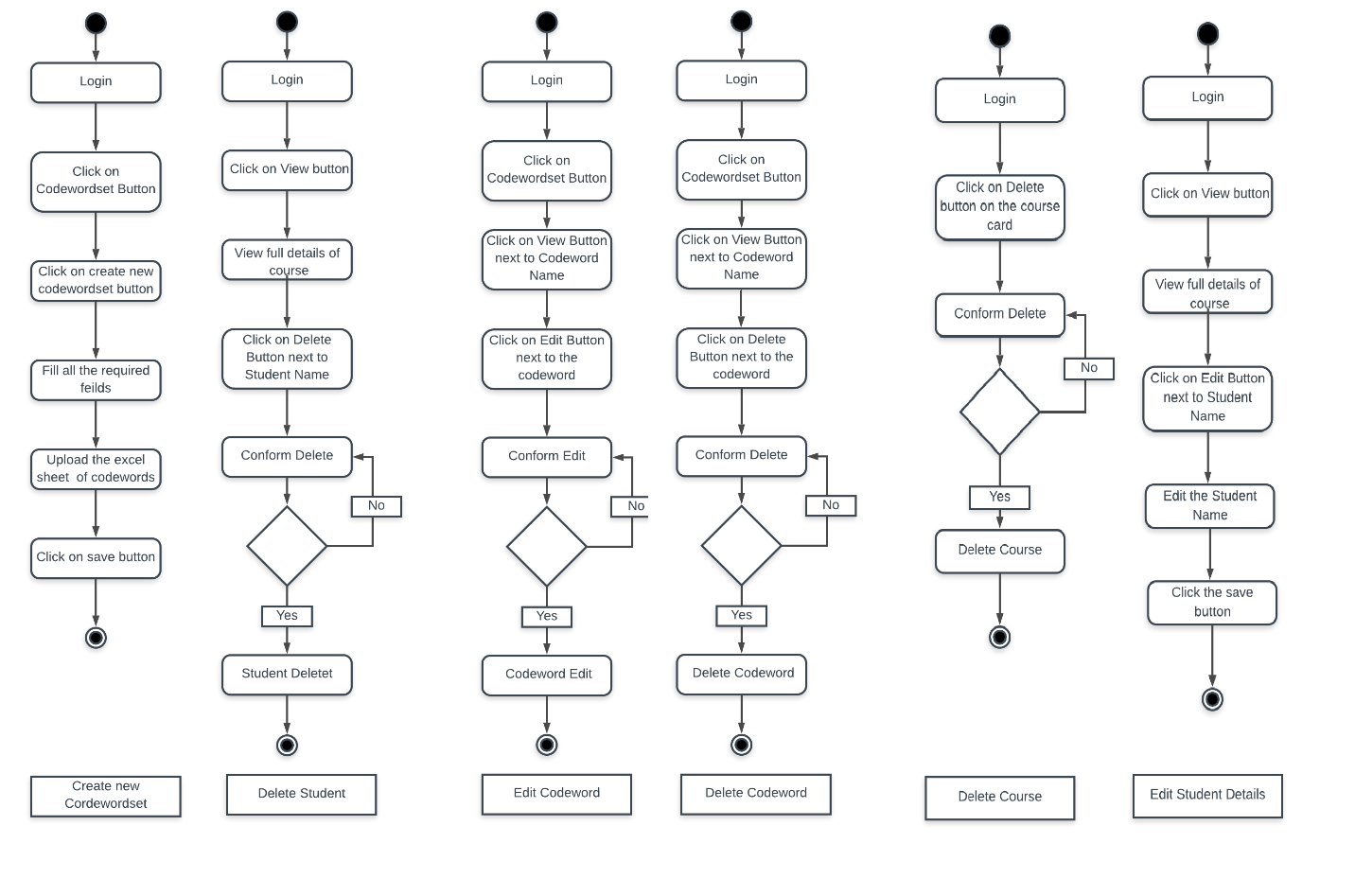
1. As an administrator, I want to add new instructors, see the list of instructors and instructor surveys details.
2. As an administrator, I want to be able to add new sets of codewords.
3. As an administrator, I want the option to hide the Instructor checkbox when new users are registering.
4. As an administrator, I want a set of instructions on how to deploy a new instance of the application (e.g. an administrator at another institution OR if each instructor has their own deployed version.)
5. As an instructor, I want to see the calculated count of students for codewords when I view a course.
6. As an instructor, I want to see less than 10 active courses.
7. As a User, I want to search the name of particular courses in a search bar.
8. As a User, I want Email id to be auto-filled.

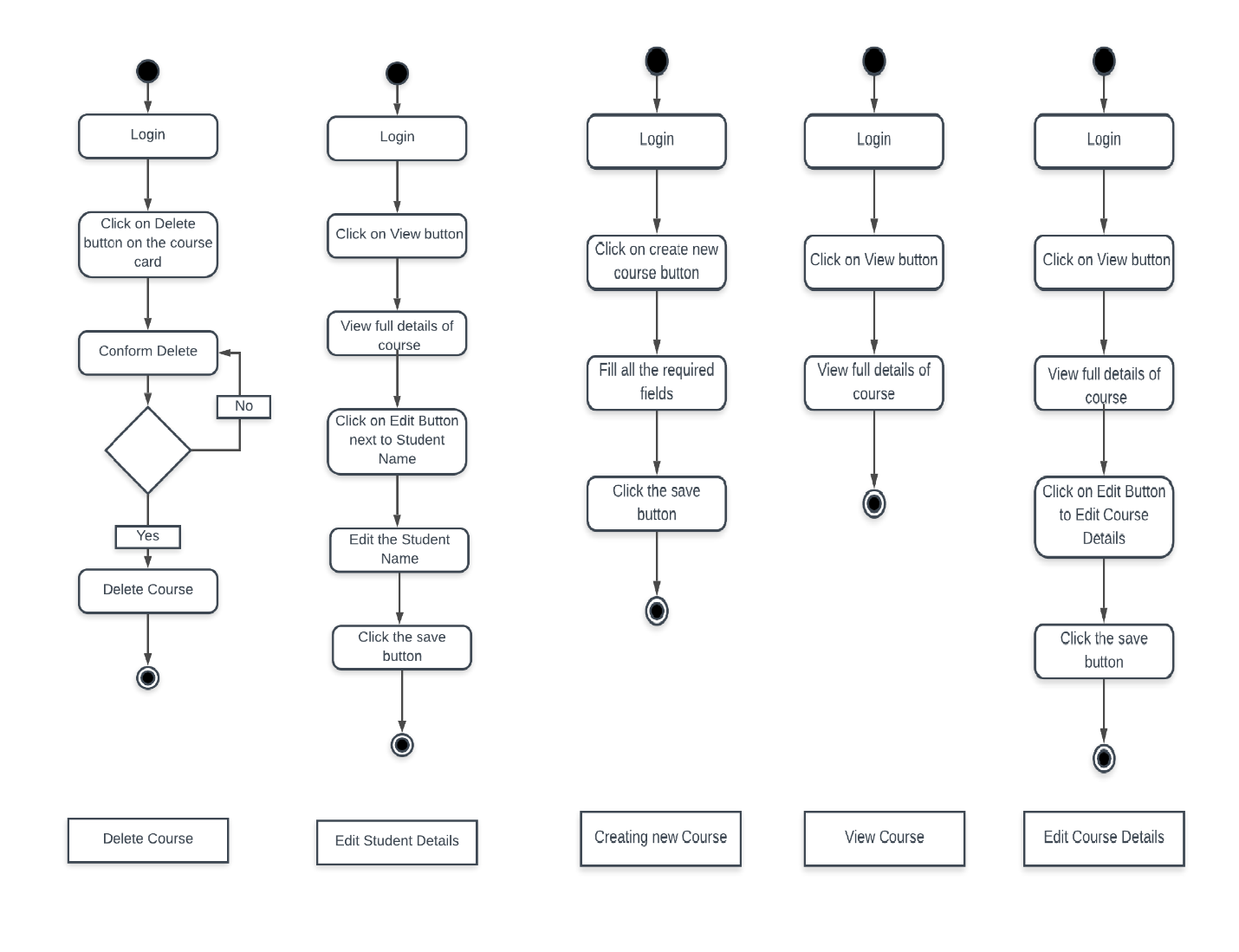
**Design:**

**User Case Diagram**

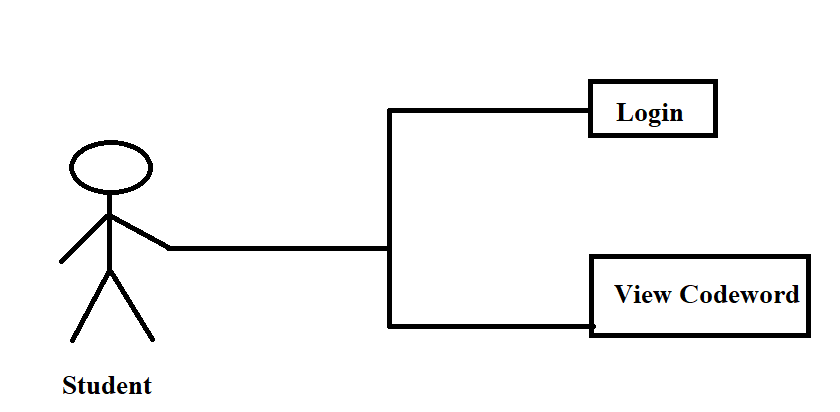


**Activity Diagram for Admin**

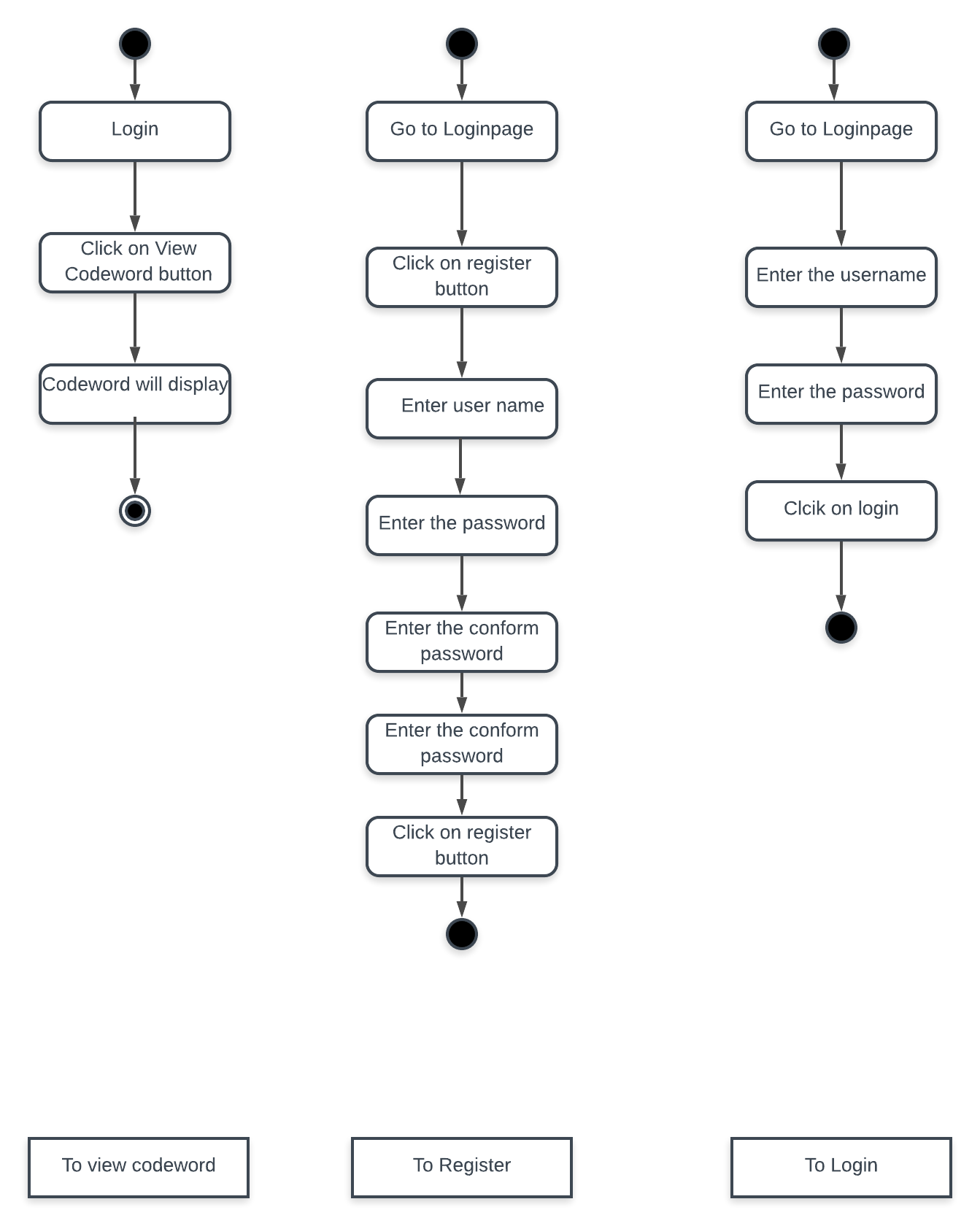




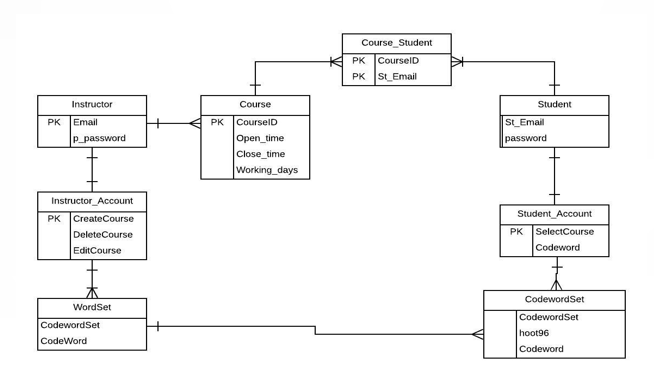
**User Case Diagram**



**Activity Diagram for Student**



**Entity Relation Diagram:**



**Tools and Technologies:**



**Angular**

Angular is a platform that makes it easy to build applications with the web. Angular combines declarative templates, dependency injection, end to end tooling, and integrated best practices to solve development challenges. Angular empowers developers to build applications that live on the web, mobile, or the desktop.

**Node JS**

Node.js is an open source, cross-platform runtime environment for developing server-side and networking applications. Node.js applications are written in JavaScript, and can be run within the Node.js runtime on OS X, Microsoft Windows, and Linux.

Node.js also provides a rich library of various JavaScript modules which simplifies the development of web applications using Node.js to a great extent.

**GitHub**

GitHub is a web-based [hosting service](https://en.wikipedia.org/wiki/Internet_hosting_service) for [version control](https://en.wikipedia.org/wiki/Version_control) using [Git](https://en.wikipedia.org/wiki/Git" \o "Git). It is mostly used for [computer code](https://en.wikipedia.org/wiki/Source_code). It offers all of the [distributed version control](https://en.wikipedia.org/wiki/Distributed_version_control) and [source code management](https://en.wikipedia.org/wiki/Source_code_management) (SCM) functionality of Git as well as adding its own features.

It provides [access control](https://en.wikipedia.org/wiki/Access_control) and several collaboration features such as [bug tracking](https://en.wikipedia.org/wiki/Bug_tracking_system), [feature requests](https://en.wikipedia.org/wiki/Software_feature), [task management](https://en.wikipedia.org/wiki/Task_management), and [wikis](https://en.wikipedia.org/wiki/Wiki) for every project.

GitHub offers plans for enterprise, team, pro and free accounts which are commonly used to host [open-source](https://en.wikipedia.org/wiki/Open-source_software) software projects

**Mongo DB**

MongoDB is an [open source](https://whatis.techtarget.com/definition/open-source) database management system (DBMS) that uses a document-oriented database model which supports various forms of data. It is one of numerous nonrelational [database](https://searchsqlserver.techtarget.com/definition/database) technologies which arose in the mid-2000s under the [NoSQL](https://searchdatamanagement.techtarget.com/definition/NoSQL-Not-Only-SQL) banner for use in big data applications and other processing jobs involving data that doesn't fit well in a rigid relational model. Instead of using [tables](https://whatis.techtarget.com/definition/table) and [rows](https://searchoracle.techtarget.com/definition/row) as in [relational databases](https://searchdatamanagement.techtarget.com/definition/relational-database), the MongoDB architecture is made up of collections and documents.

**Heroku**

Heroku is a container-based cloud Platform as a Service (PaaS). Developers use Heroku to deploy, manage, and scale modern apps. Our platform is elegant, flexible, and easy to use, offering developers the simplest path to getting their apps to market.

Heroku is fully managed, giving developers the freedom to focus on their core product without the distraction of maintaining servers, hardware, or infrastructure. The Heroku experience provides services, tools, workflows, and polyglot support—all designed to enhance developer productivity.

**Postman**

Postman is a Google Chrome app for interacting with HTTP APIs. It presents you with a friendly GUI for constructing requests and reading responses. The people behind Postman also offer an add-on package called Jetpacks, which includes some automation tools and, most crucially, a Javascript testing library.

**VS Code**

Visual Studio Code is a source-code editor developed by Microsoft for Windows, Linux and macOS. It includes support for debugging, embedded Git control, syntax highlighting, intelligent codecompletion, snippets, and code refactoring.

**Google Docs**

Google Docs is a part of a comprehensive package of online applications offered by and associated withGoogle. Users of Google Docs can import, create, edit and update documents and spreadsheets in various fonts and file formats, combining text with formulas, lists, tables and images.

**Selenium**

Selenium is a portable [framework](https://en.wikipedia.org/wiki/Software_framework) for [testing](https://en.wikipedia.org/wiki/Software_testing) [web applications](https://en.wikipedia.org/wiki/Web_application). Selenium provides a playback (formerly also recording) tool for authoring [functional tests](https://en.wikipedia.org/wiki/Functional_testing) without the need to learn a test [scripting language](https://en.wikipedia.org/wiki/Scripting_language) (Selenium IDE). It also provides a test [domain-specific language](https://en.wikipedia.org/wiki/Domain-specific_language) (Selenese) to write tests in a number of popular programming languages, including [C#](https://en.wikipedia.org/wiki/C_Sharp_(programming_language)), [Groovy](https://en.wikipedia.org/wiki/Groovy_(programming_language)), [Java](https://en.wikipedia.org/wiki/Java_(software_platform)), [Perl](https://en.wikipedia.org/wiki/Perl), [PHP](https://en.wikipedia.org/wiki/PHP), [Python](https://en.wikipedia.org/wiki/Python_(programming_language)), [Ruby](https://en.wikipedia.org/wiki/Ruby_(programming_language)) and [Scala](https://en.wikipedia.org/wiki/Scala_(programming_language)). The tests can then run against most modern [web browsers](https://en.wikipedia.org/wiki/Web_browser). Selenium deploys on [Windows](https://en.wikipedia.org/wiki/Microsoft_Windows), [Linux](https://en.wikipedia.org/wiki/Linux), and [macOS](https://en.wikipedia.org/wiki/MacOS" \o "MacOS)platforms. It is [open-source software](https://en.wikipedia.org/wiki/Open-source_software), released under the [Apache 2.0 license](https://en.wikipedia.org/wiki/Apache_License): web developers can download and use it without charge.

# **Test Cases:**

Click the below link to open the Test Cases:

<https://github.com/ujjwalkumar1212/Codeword-Generator-Scrutinizer/blob/master/Project%20Artifacts/CodewordsTestCases.xlsx>

# **Result and Analysis of Test Cases:**

After performing the rigorous testing on the Codeword MEAN stack application the following are results of analysis.

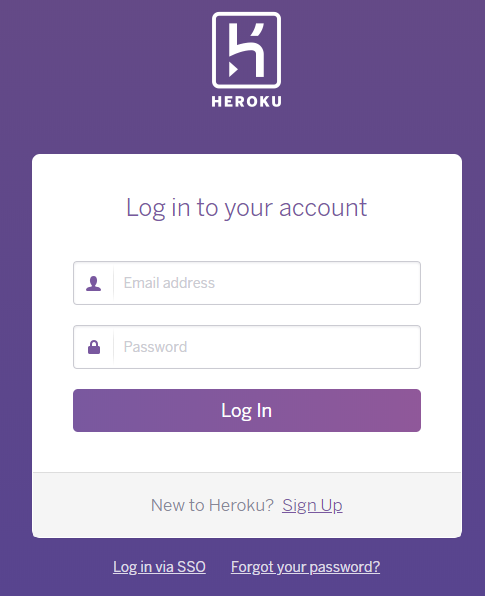
1. The user who has registered as Instructor has no chance of changing his role as student.
2. Course is getting created without uploading students list.
3. There is no function to add a single student for a course. It must be added only through the excel sheet.
4. There are no feedback messages to notify the user about the error.
5. Few of the Codeword set creation constraint are not working as expected.
6. No hints were provided to the Instructor to upload either the student details/ creating a code word set.

# **Deployment**

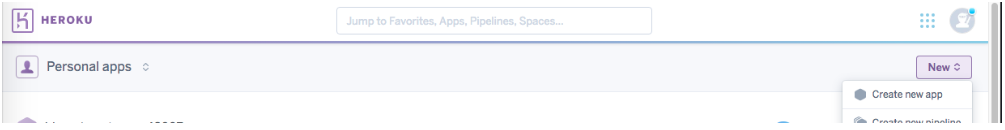
The project involves two different applications: a web application, accessed by EOC and a mobile application, accessed by CERT members. The deployment of web application is done on heroku. Heroku serves as a “Cloud Platfrom as a Service”, generally used to deploy any application so that the developers can access the application from anywhere and run it.

**Procedure to deploy and run the application:**

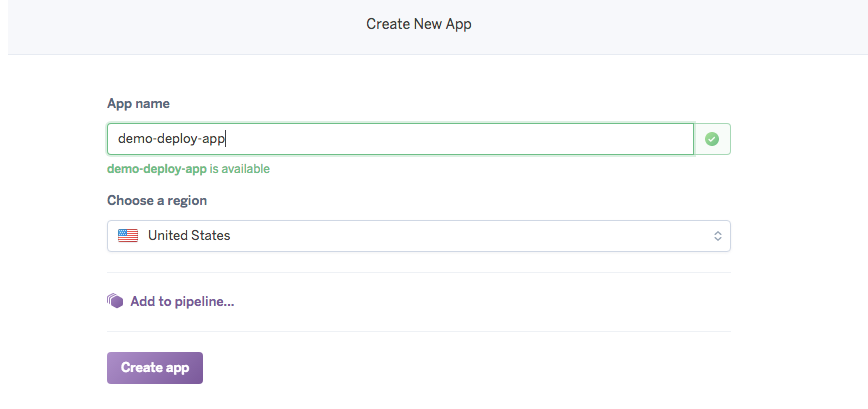
1. GitHub Repo link: <https://github.com/ujjwalkumar1212/Codeword-Generator-Scrutinizer>
2. Login to your Heroku account with your respective credentials, if not create an account and then login.

****

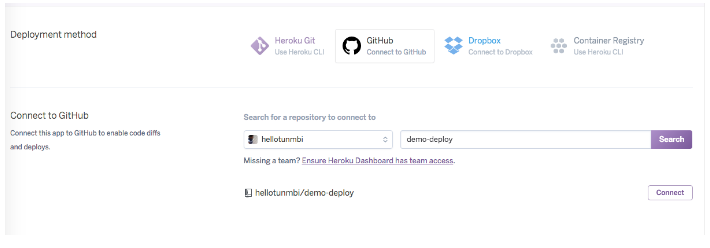
1. Click on “New” present on the top right corner.
2. Select “Create new app”.

****

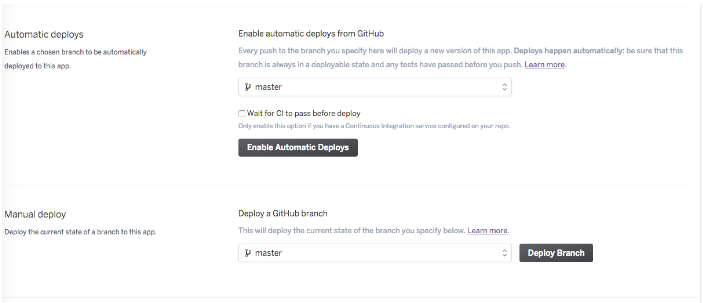
1. Enter the name of the application (Criteria: should contain lower letters, numbers and dashes).

****

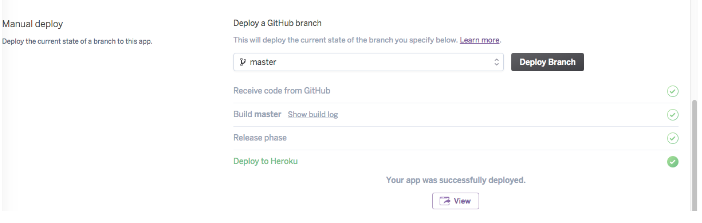
1. Click on “Create app”.
2. Select “GitHub” option and click on the search button in order to connect to the specific repository.

****

1. Under “Automatic Deploys”, select the master branch and click on “Enable Automatic Deploys”.
2. Under “Manual Deploys”, click on “Deploy Branch” in order to push in the fresh code to heroku.

****

1. It takes a little time to deploy the application. Once done a feedback message will be displayed at the bottom.

****

1. In order to run the deployed application, run the “Heroku Git URL”, present in settings, in any browser.