Step 1: db design

Name: survey

Tables and values:

Id: object (default by mongo db)

**Name**: *String* (the name of the user who is creating the survey)

**Question**: *String* (the title of the survey and the question of what is the survey is about)

**OptionOne**: *Dictionary* (It needs two values)

**Name**: *String* (the name of what you can vote on)

**Vote**: *Number* (It is set to 0 first creation, increment each time a vote was selected for this object.

**OptionTwo**: *Dictionary* (It needs two values)

**Name**: *String* (the name of what you can vote on)

**Vote**: *Number* (It is set to 0 first creation, increment each time a vote was selected for this object.

**OptionThree**: *Dictionary* (It needs two values)

**Name**: *String* (the name of what you can vote on)

**Vote**: *Number* (It is set to 0 first creation, increment each time a vote was selected for this object.

**OptionFour**: *Dictionary* (It needs two values)

**Name**: *String* (the name of what you can vote on)

**Vote**: *Number* (It is set to 0 first creation, increment each time a vote was selected for this object.

**Created\_at**: *Date* (time of creation)

Step 2: Create pages

Index.ejs – home page used for login

Dashboard.ejs – show all the surveys

Poll.ejs – to show the individual survey

Create.ejs – where you can create a new survey

Step 3: To do

**Queries to db:**

Add new survey

Show all survey

Show one survey

Update a vote for one survey

Show new result

Delete a survey

Search – There are no criteria given, assumption is made:

* Search by user who created it the survey
* Search by full or partial title
* Search by date

Login and out – It is needed to track who created the survey and not allow that person to vote on its own survey.

* Phase one I will add filed as placeholder to enter name. One the login and out working than that filed is not needed and can be replaced by grabbing the user account.
* I need if statement with an error if the user matches survey creation than that person can’t view the survey.

Step 4: video

Need to video tape it or screen capture it, quick time free version does not work on Windows. The option to record is missing.

Step 5: deploy

This may or may not going to work as I am not using a full MEAN technologies (I have not finished learning them all.)

Exam instructions:

Required Features:

* User login/logout
* Display of all survey questions, author of the survey, and date created on the Polls dashboard page.
* Search feature.
* Individual survey page which displays questions, options and vote counts per option.
* Ability to Vote and the vote count should increase.
* Ability to add/create new survey question. Validation applies.
* Delete survey feature for the user who created the survey question.
* You must be able to deploy your work to Amazon EC2 and provide the IP address or subdomain/domain name to where your work has been deployed. Please note that Heroku deployment is not honored.

**Submission Requirement:** Your code, a text file with the IP address or subdomain/domain name for the deployed work and short video demo link.

**Note**:

* The wireframe is available for download [here](http://s3.amazonaws.com/General_V88/boomyeah/company_209/chapter_2779/handouts/chapter2779_3427_survey-3.png).
* Create a 2-3 video showing what you have built. On the video, show the functionality of the page you have built. You may download a trial version of Jing or Snag-it or Quicktime for video creation. You may save it on Dropbox public folder and  copy the public link.
* To prevent other students from copying your work, please make sure your video demo is set to unlisted or with a password, and code is not saved on GitHub.
* To minimize score deduction, please make sure you upload the complete requirements (database file, video demo).
* If you are having trouble uploading files on the Exam App, please DO NOT unlock another exam. Please email the file to remote-instructors@codingdojo.com and CC your instructor.

