Virginia Polytechnic Institute and
State University
Blacksburg, VA 24061
CS 4624 – Multimedia, Hypertext, and Information Access
Spring 2020

Final Report

Music Survey

Yash Makkena, Mrinal Save, Kelly Tran, Mehmet Yuksel

Client: Brett Jones
Professor: Edward A. Fox
May 12, 2020

Table of Contents

1 List of Figures **4**

2 Executive Summary **5**

3 User Manual **6**

3.1 Contract Overview **6**

3.2 Purpose of the Project **7**

3.3 Scope and Users **8**

3.4 Contents of Website **9**

3.4.1 Webpages Overview **9**

3.4.2 Dashboard - Teacher View **9**

3.4.3 Surveys - Student View **10**

3.5 Website Requirements **10**

3.6 Maintenance Requirements **10**

4 Developer Manual **11**

4.1 Methodology Overview **11**

4.1.1 Methodology Goals **11**

4.1.2 Goal 1 Workflow **12**

4.1.3 Goal 2 Workflow **14**

4.1.4 Goal 3 Workflow **16**

4.1.4 Goal 4 Workflow **18**

4.2 Design **19**

4.2.1 Technologies/Tools Used **19**

4.2.2 Project Organization **19**

4.2.3 Web Page Design **20**

4.2.4 MUSIC Score Visualization **22**

4.2.5 Survey Program and Integration **22**

4.2.6 Database **22**

4.3 Implementation **23**

4.3.1 Technology Used **23**

4.3.2 Database Setup **23**

4.3.3 Access Code Generation **24**

4.3.4 Access Code Web Page **25**

4.3.5 Permission Web Page **26**

4.3.5 Base Survey Form Building **27**

4.3.7 PHP Queries to Database 28
4.3.8 Data Visualization 28
4.3.9 Alternate Surveys 30
4.3.10 Languages 31

5 Testing 32
5.1 Database Testing 32
5.2 Web Page Testing 32
5.3 Visualization Testing 32

6 Lessons Learned 34
6.1 Timeline 34
6.2 Challenges 35
6.3 Solutions 35

7 Future Work 37

8 Acknowledgements 38

9 References 39

1 List of Figures

Figure 1. Model of Original Process of Survey Generation
Figure 2. Model of Current Process of Survey Generation
Figure 3. Research Options to Share MUSIC Survey Data
Figure 4. Overall Goal 1 Workflow Diagram
Figure 5. Goal 1, Task 1 Flowchart Diagram
Figure 6. Goal 1, Task 2 Flowchart Diagram
Figure 7. Overall Goal 2 Workflow Diagram
Figure 8. Goal 2, Task 1 Flowchart Diagram
Figure 9. Goal 2, Task 2 Flowchart Diagram
Figure 10. Overall Goal 3 Workflow Diagram
Figure 11. Goal 3, Task 1 Flowchart Diagram
Figure 12. Goal 3, Task 2 Flowchart Diagram
Figure 13. Overall Goal 4 Workflow Diagram
Figure 14. Goal 4, Task 1 Flowchart Diagram
Figure 15. Layout of Web Page and Database Connection
Figure 16. Concept Wireframe of Log-in Access Code Page
Figure 17. Concept Wireframe of Access Code Generation Page
Figure 18. Concept Wireframe of Instructor Dashboard Page
Figure 19. Concept Wireframe of Create Survey Page
Figure 20. Concept Model for Visualization of Data
Figure 21. Database Schema
Figure 22. Quadratic Probing Flow Diagram
Figure 23. Access Code Generation Web Page
Figure 24. Instructor Options to Share MUSIC Survey Data
Figure 25. Layout of Survey Questions
Figure 26. Current Model for Visualization of Data
Figure 27. Sample Present Tense Questions in the College Survey
Figure 28. Sample Past Tense Questions in the College Survey
Figure 29. Initial D3 Visualization of MUSIC Scores

2 Executive Summary

The goal of this project is to create an interactive website that teachers can use to obtain feedback about students' perceptions of their class. Professor Brett Jones has developed a survey (called the MUSIC Model of Motivation Inventory) that teachers use to assess their students' perceptions of their class. Teachers around the world are using this survey. However, to use it, they either need to use paper copies and enter the data, or use an online program (Qualtrics, Google Forms) to collect the data. Then, the teachers need to compute the averages for each of the five scales in the survey. As an alternative, Dr. Jones is developing an online, interactive tool that teachers can access to collect these data and provide the results automatically. Using Google apps, Dr. Jones and his colleague, Dr. Cruz, have started working on this website which will consist of a page linked to www.theMUSICmodel.com.

Teachers will enter their information and select which parts of the survey they want to give students. The website will then send a link to the teacher that s/he can give to his/her students to complete. After students have completed the online survey, the teacher can go back into the website and access the results, which will include numbers on at least five different scales that were created by averaging individual items. For example, for the "Interest" score, the website will automatically average the scores from 6 of the 26 items and will provide the average and a graph showing the results. In addition, the teacher will be provided with suggestions for improving their instruction based on the results.

The main goal of this project is to create an interactive website to survey students about their course perceptions as related to the MUSIC model of learning. The interactive website allows instructors to focus on certain parts of the survey they would like to give their students (e.g. an instructor would like to see how empowered their students feel). After completing the survey, instructors can visit the website to view individual component results, as well as averages and visualizations.

The core technologies used for this project are primarily related to HTML, databases, and JavaScript. The surveys are created using HTML and a database is used to store the answers. Since the website is hosted on Bluehost, phpMyAdmin and MySQL were used for the database due to Bluehost support and compatibility. For the visualization of the results, D3.js, a JavaScript library, was used to provide dynamic and interactive visualization.

3 User Manual

3.1 Contract Overview

The Music Model questionnaire was created for Dr. Brett Jones' and teachers who utilize his model of learning in the classroom. Dr. Jones provides surveys to teachers who in turn provide the survey to their students. This project aims to assist the teachers by making it easier to store and visualize survey results, as well as making it simpler to provide the surveys to students. The deliverable for this project includes a functional website with a connected database to store the teachers' scores.

The expectation for the outcome of this project is that teachers using our survey tool will be better equipped to understand the needs of their students and provide an improved learning experience. The skills required for this project include both back-end and front-end web development, specifically with HTML, Javascript, PHP, and some CSS. The survey questions were formatted and listed according to the specifications which can be found on Dr. Jones' website [3].

The roles for the project are divided as below:

Mrinal Save:	Front end, Design, JavaScript
Yash Makkena:	Back-end, JavaScript, PHP
Kelly Tran:	Quality Control, Reports/Presentations
Mehmet Yuksel:	Database, PHP

Date that the client has signed off on this:	01/28/20
Date that the instructor has signed off on this:	02/02/20

3.2 Purpose of the Project

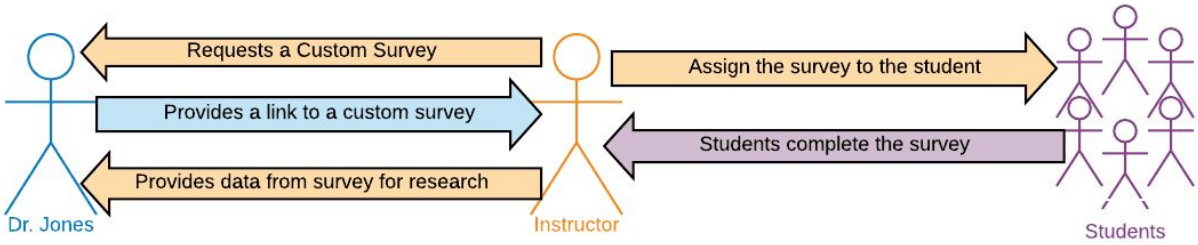

Figure 1. Model of Original Process of Survey Generation

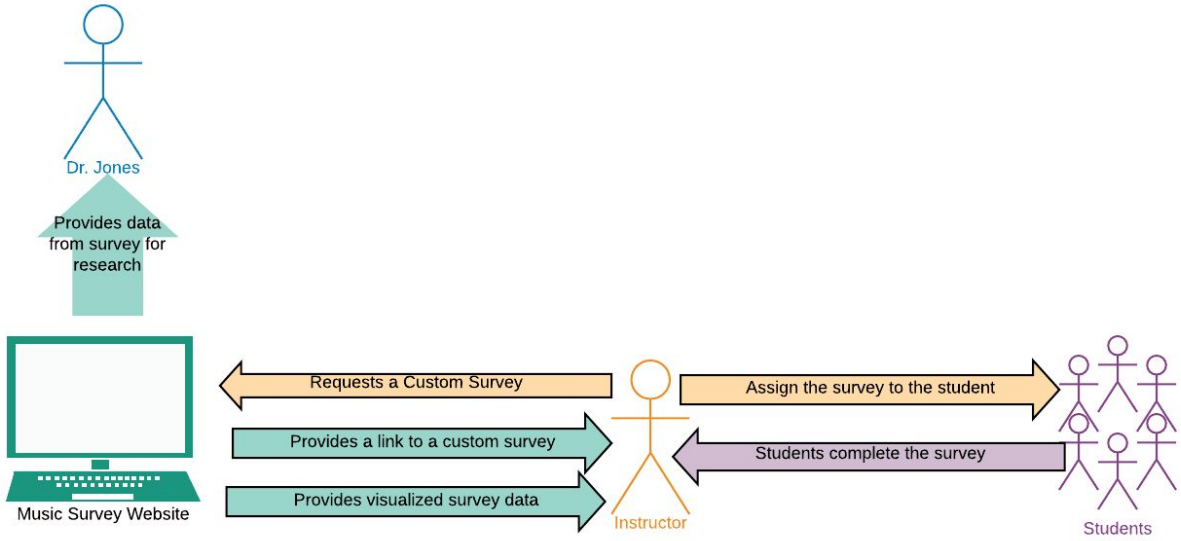

Figure 2. Model of Current Process of Survey Generation

The MUSIC Model of Motivation Inventory is a survey developed by Dr. Brett Jones that teachers use to assess their students' perceptions of the class. The MUSIC model represents five key principles: empowerment (M), usefulness (U), success (S), interest (I), and caring (C). His model aims to help teachers understand their strengths and weaknesses in each principle of the MUSIC model by providing teachers with a survey to give to their students. Students are asked to score the teachers on a scale from 1-6 in response to questions that pertain to one of the five principles. The scores for each question are averaged and then are provided to the teacher. The surveys have different questions based on the grade-level of the students and the subjects in question.

Due to the volume of interest in Dr. Jones' learning model, he is seeking a more streamlined process in which he is able to provide the customized surveys to instructors. Many instructors have contacted Dr. Jones, requesting that they be provided with a certain survey pertaining to a specific portion of the MUSIC model. This project aims to reduce the workload of Dr. Jones in this process by automating the process of survey generation. Currently, Dr. Jones has to manually provide each survey to the instructors, as seen in Figure 1. By creating a website that directly generates the surveys for instructors using the question subsets that Dr. Jones had created, we streamline the process, as shown in Figure 2. It also provides immediate results from the surveys so that instructors can more efficiently evaluate their method of teaching.

The main purpose of the website is for teachers to gather feedback. Instructors should be able to use the website to view and sort their results in a straightforward manner. They should also be able to receive feedback on how to improve instruction based on their results. For instance, a low score for U (usefulness) would indicate that instructors may need to better explain why the material is useful for students' short-term or long-term goals.

3.3 Scope and Users

There are two main audiences for this interactive website - the instructors collecting data, and the students taking the surveys. Since instructors and students of many different fields and ages will be using this website, the website needs to be easy to navigate so that it can be a useful resource. The website aims to let instructors easily set up surveys and view results, as well as for students to be able to easily access and take the surveys.

The website is intended to replace the survey system that is currently in place, so it needs to be fully functional with all the features implemented. It should be easy to use for the students, accessible to younger students as well, and should be free of any bias that might skew a student towards a certain answer.

3.4 Contents of Website

3.4.1 Webpages Overview

In order to create the website's functionality, there are a few key webpages that we needed to develop:

- Access Page
- Permissions Page
- Teacher Dashboard
- Survey Creation Page
- Student Survey Page

3.4.2 Dashboard - Teacher View

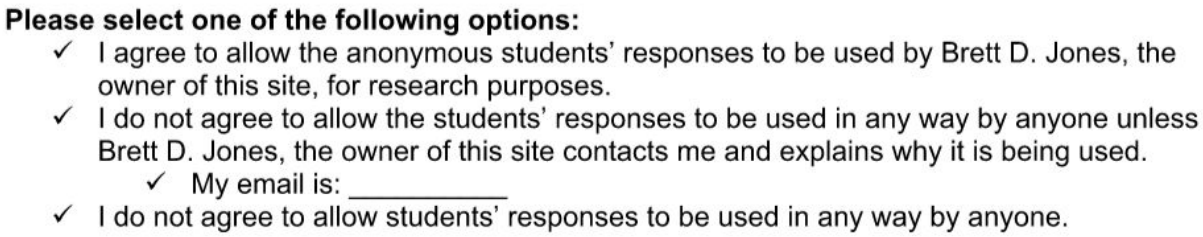

Figure 3. Research Options to Share MUSIC Survey Data

The teacher's view is a dashboard that contains links to all the required functionalities. The dashboard appears after a teacher enters their access code and is logged in. The initial view is a tabbed list for options to view completed surveys and ongoing surveys. Selecting a completed survey will generate a barchart of the averaged MUSIC [3] values; selecting an ongoing survey will show a button allowing the instructor to mark the survey as completed. There is also a link to change the permissions of data access. Since the MUSIC Model is part of a research program run by Dr. Jones, he hopes to be able to utilize the data in some of his own work. Instructors can select one of three options as seen in Figure 3.

There are also options to create a new survey which provides the instructor with a dropdown menu to select the type of survey they want to create (based on grade-level and/or subject).

3.4.3 Surveys - Student View

The students will be provided with a permalink to a survey page with the questions that their instructor has selected. This page is a simple view of just the survey questions and the 6 options to rate the instructor based on the question. The submit button at the bottom posts the results to the back-end and displays a message saying the student has submitted the survey.

3.5 Website Requirements

The website needs to be compatible with basic web browsers. We cannot be sure which browser or even the version of browser that instructors or students are using, so we need to ensure that the website is able to load and be functional across the platforms. To guarantee this, we are using the most basic of HTML elements, with no images or packages that would cause the website to become bulky. We want the survey to be accessible to all students and instructors.

3.6 Maintenance Requirements

In addition to our website deliverables, we also set up the back-end database on Dr. Jones' Bluehost account. Bluehost uses phpMyAdmin, which Dr. Jones and his colleague, Dr. Juan Cruz, are already familiar with. By creating a database to store the survey questions, we create a simple interface for Dr. Jones to be able to update or remove any questions as his research progresses. We have also created a simple query to be able to collect all the data from the surveys (according to the permissions that the instructor has provided) into a CSV file, so the data can easily be observed and manipulated as desired.

4 Developer Manual

4.1 Methodology Overview

4.1.1 Methodology Goals

Our project needs to support both teachers and students.

Goal 1: Teachers should have access to their student's results broken up by classes and options to visualize and compare different datasets of results. This will be done by generating a unique access code for a teacher when they first submit their information. Teachers will then be able to use this access code to view the submitted survey information and visualize any sets of data. They can also use the access code to create a new survey, whose data will be added to their "account". See Figure 4.

Goal 2: Students should only have the ability to take the survey once with their information which will be cross-checked in the database with any form of student ID. The student ID will be encrypted when stored in the database so that no one will be able to access the data - it will only be used to check if a student has already submitted a survey for that specific course. See Figure 7.

Goal 3: We want to take the submitted data and create visualizations of the different sections using the D3.js library. This will involve separating the data into the MUSIC sections and calculating numbers. See Figure 10.

Goal 4: Ensure privacy of student and teacher data - student IDs should be accessible by no one, only teachers should have access to their own data - through encryption and hashing. See Figure 13.

4.1.2 Goal 1 Workflow

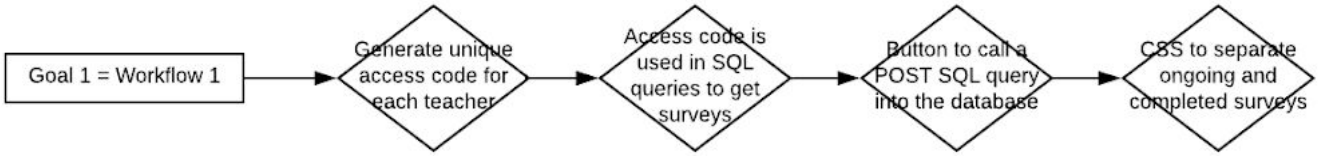

Figure 4. Overall Goal 1 Workflow Diagram

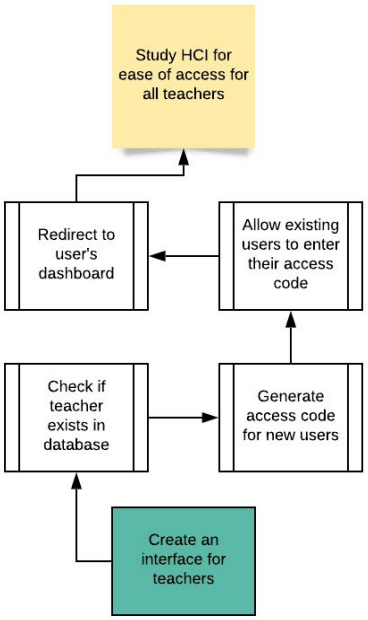

Figure 5. Goal 1, Task 1 Flowchart Diagram

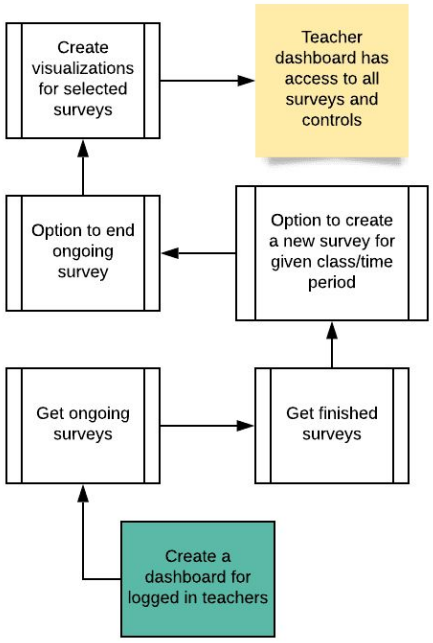

Figure 6. Goal 1, Task 2 Flowchart Diagram

Task 1:

In order to create unique access codes for each teacher, we will have a database storing the basic information of each teacher - name and school. Using a basic hashing function, we will generate an access code based on the teacher's information. A teacher can input their name and access code in order to gain access to a pre-existing account in order to view information about any surveys that they have created. Once logged in, the teacher will be redirected to their dashboard with their information as described in Task 2. See Figure 5.

Task 2:

Teacher's dashboard will contain past surveys and ongoing surveys. Teachers will have a button in order to be able to create a new survey, which will then get added to their list of ongoing surveys. All of this information will be stored in the database with the teacher's access code as a column so that only that teacher will have access to them. A SQL query will allow the back-end to get all surveys that are connected to that access code. See Figure 6.

4.1.3 Goal 2 Workflow

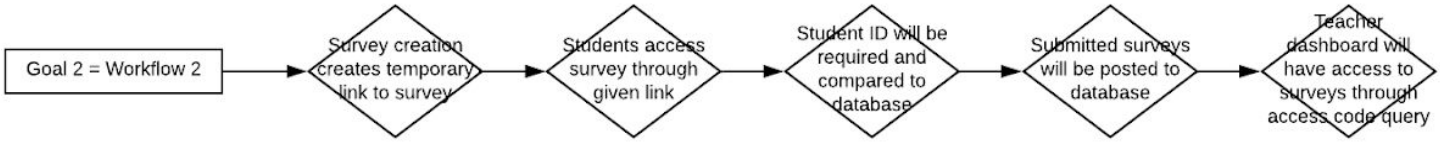

Figure 7. Overall Goal 2 Workflow Diagram

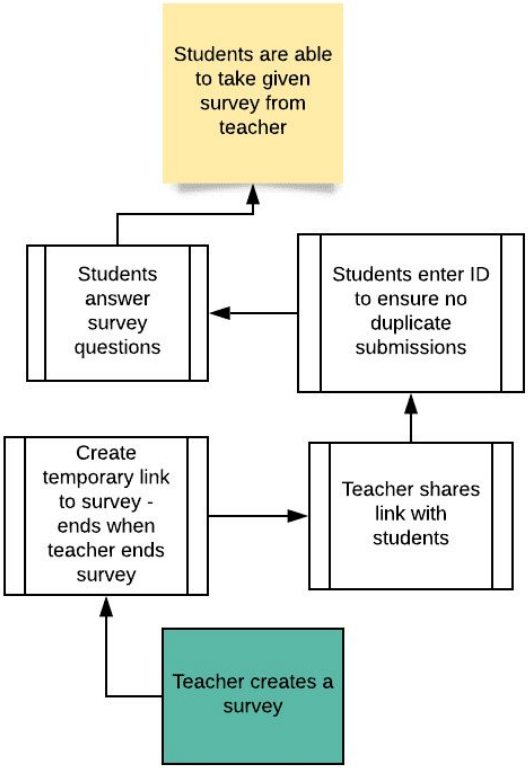

Figure 8. Goal 2, Task 1 Flowchart Diagram

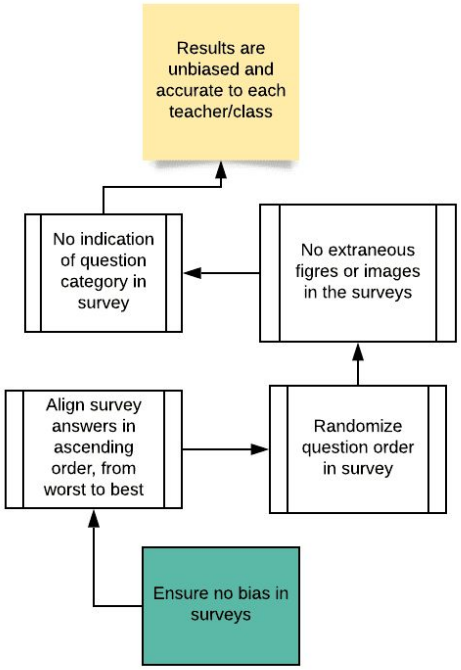

Figure 9. Goal 2, Task 2 Flowchart Diagram

Task 1:

When the teacher creates a survey, the back-end JS will automatically create a semi-permanent link to the survey using randomization. The teachers will be able to share this link with their students and the students will now be able to take the survey until the teacher closes the survey. The students will be required to enter their student ID, which will be hashed before storing in the database. This is only to ensure students do not accidentally submit a survey more than once. Once a teacher ends a survey, the student IDs will all be removed from the database, since we only need to store the MUSIC category average values and no further survey submissions can be made. See Figure 8.

Task 2:

Since this project is based on an existing research project and the results of the survey can be used by Dr. Jones, based on the teachers' permission settings, we need to ensure that the survey layout is as unbiased as possible. The main thing is to make sure that the survey answers are laid out so that there is no leaning towards one answer. The best way to do this is to make sure that all answers are laid out from 1-6 in vertical layout. Another important bias-prevention tactic is to keep the survey form empty of clutter. Only have the basic instructions, no figures or images. There also should not be an indication of the category of the question on the survey, as this may skew student answers. See Figure 9.

4.1.4 Goal 3 Workflow

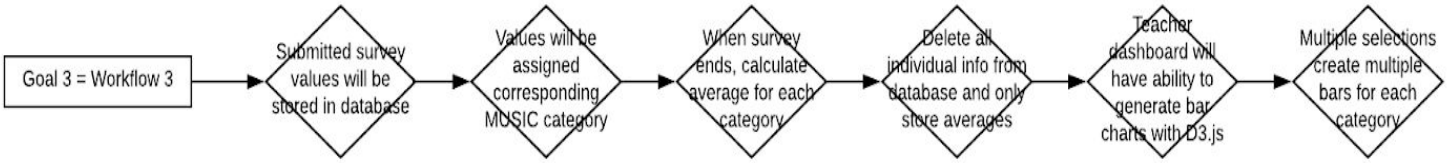

Figure 10. Goal 3 Overall Workflow Diagram

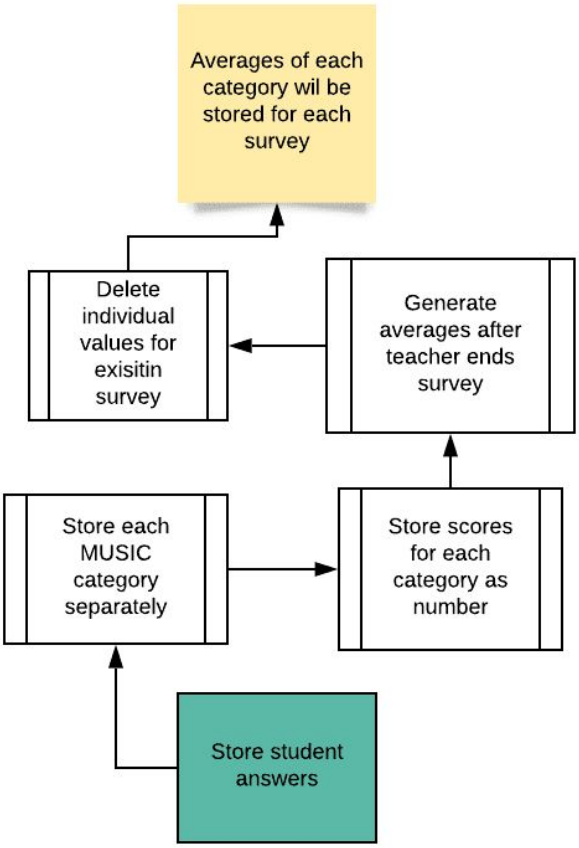

Figure 11. Goal 3, Task 1 Flowchart Diagram

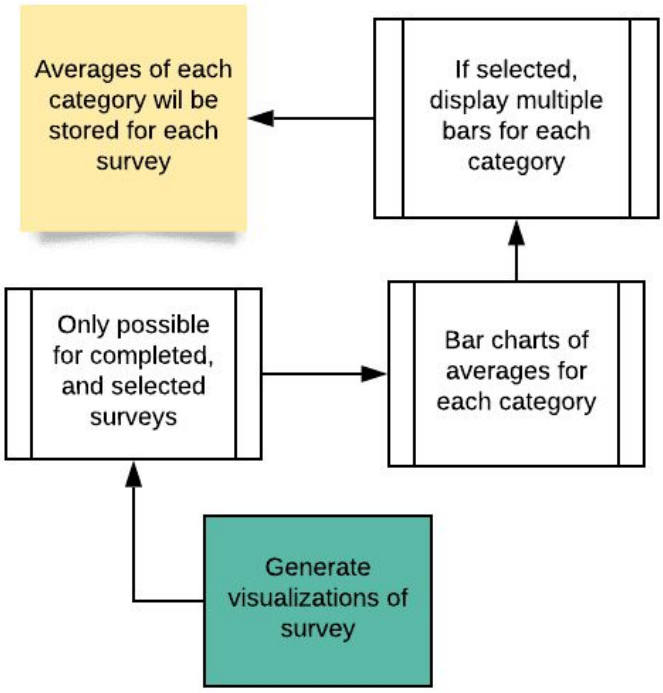

Figure 12. Goal 3, Task 2 Flowchart Diagram

Task 1:

When a student submits a survey, we want to use the back-end JavaScript and post the information to the corresponding survey table using a SQL statement to store the values submitted by the student. Each value will have a corresponding column to indicate what category of MUSIC the question belongs to. Once a teacher chooses to end a survey, we will change the type of table that corresponds to the access code and course. Then, only the averages of each category will be stored and no other information about individual values or information. See Figure 11.

Task 2:

After a survey is submitted, the teacher's dashboard will have an option to select the survey to generate a bar chart of the average values. Otherwise, an ongoing survey will not be able to be selected for visualization. To create the bar charts, we will use our back-end JavaScript and the D3.js library to get the average-values data from the database and feed them into D3. Multiple surveys are able to be created, and D3 can handle placing multiple bars on each category. See Figure 12.

4.1.4 Goal 4 Workflow

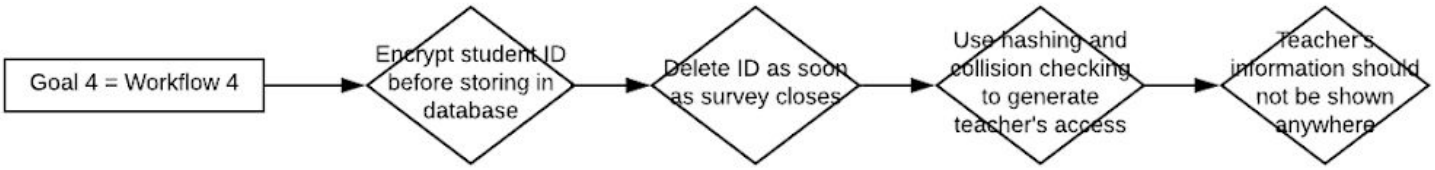

Figure 13. Goal 4 Overall Workflow Diagram

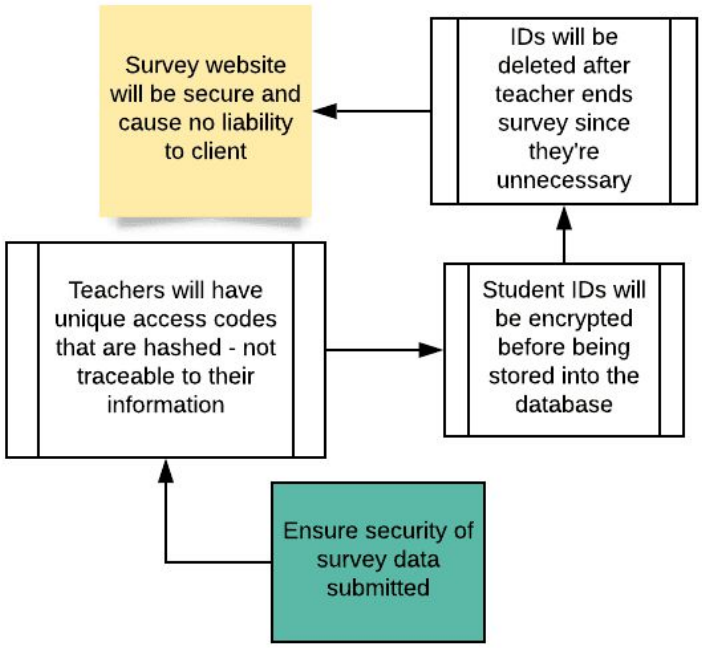

Figure 14. Goal 4, Task 1 Flowchart Diagram

Task 1:

Because this website will handle some slightly sensitive data, we want to ensure that the necessary information is being secured. That is why the student ID will be encrypted before it's stored in the database. They will also be removed as soon as the teacher closes the survey, since then it would be unnecessary data being stored in the database. We will also ensure that each teacher's access code is unique by hashing and collision checking. This access code will be tied to a teacher and their school, but that data will not be used anywhere. See Figure 14.

4.2 Design

4.2.1 Technologies/Tools Used

To develop this project, the main programming languages we used were HTML, PHP, JavaScript, and CSS. In order to create the database, we used the MySQL widget provided on BlueHost [2] [4].

4.2.2 Project Organization

As was shown earlier in Figure 2, the goal of the project is to minimize Dr. Jones' work in the MUSIC survey creation project. To do this, we have created a series of webpages that connect to a database with several tables.

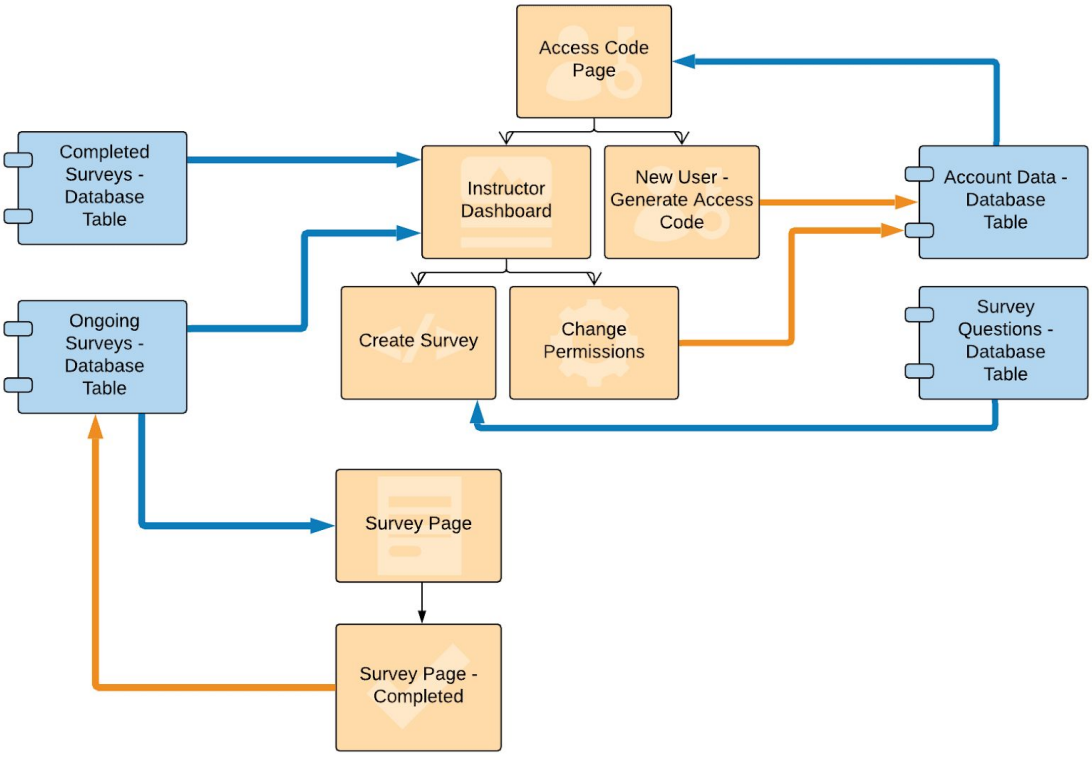

Figure 15. Layout of Web Page and Database Connection

4.2.3 Web Page Design

Before creating the website, we made mockups of each page to guide us on our CSS and HTML design. This helped us visualize the connection between the different webpages and the links that should be placed in each page. See Figures 16-19.

Figure 16. Concept Wireframe of Log-in Access Code Page

Figure 17. Concept Wireframe of Access Code Generation Page

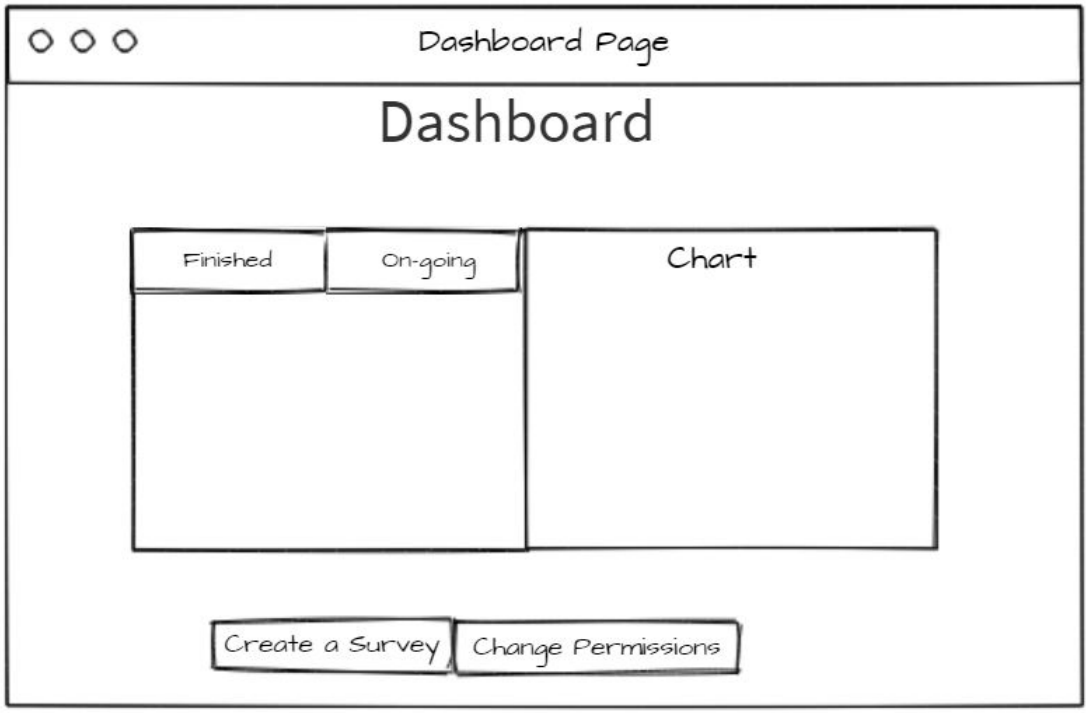

Figure 18. Concept Wireframe of Instructor Dashboard Page

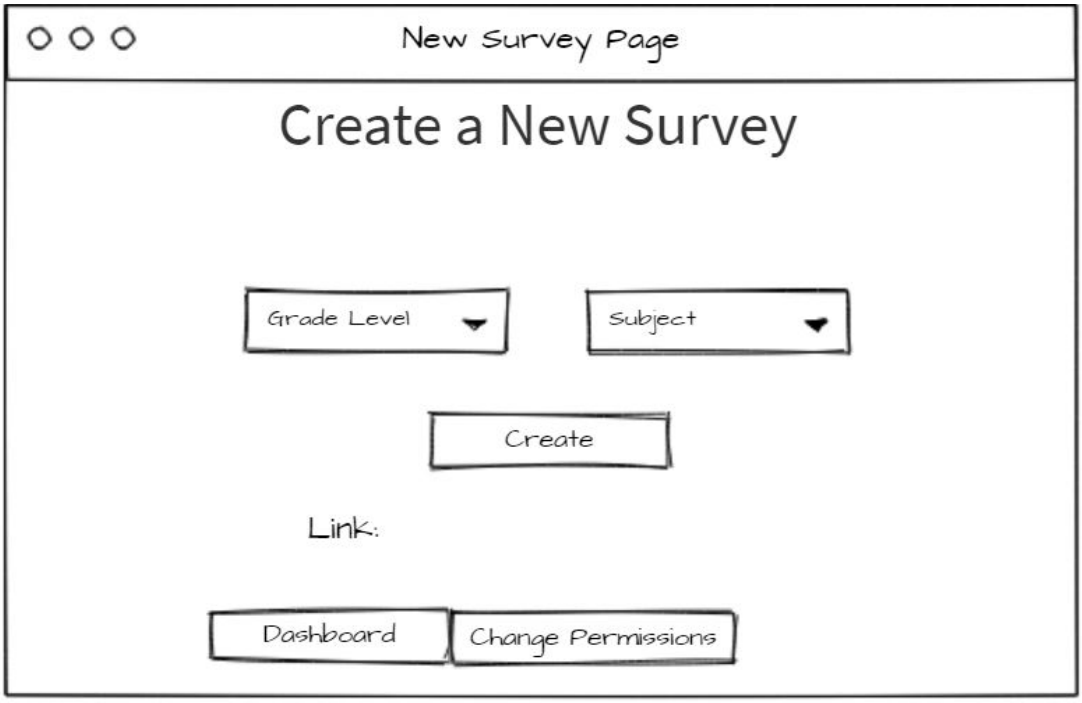

Figure 19. Concept Wireframe of Create Survey Page

4.2.4 MUSIC Score Visualization

The client proposed an idea for data visualization he had in mind (see Figure 20). This was used as our template when working with different ways to model the data. Each bar in a chart represents the average score of the questions in each category of the MUSIC model.

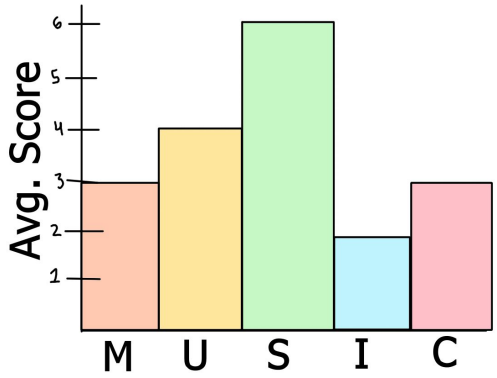

Figure 20. Concept Model for Visualization of Data

4.2.5 Survey Program and Integration

The surveys generated must be easy to navigate, much like a Qualtrics survey, and match the current theme of the existing website. However, when working on the project, due to slight communication issues during the transition to virtual classes, we did not get access to the client's website when we had hoped to, and therefore could not work with the existing framework - this included styling and design. As a result, the survey interface and program were created separately from the existing website with different templates. Our surveys and visualizations were completed locally and were integrated after obtaining the client's credentials for the website.

4.2.6 Database

There are two main reasons why a database is necessary for this project. Since the MUSIC model and associated surveys are used internationally by a large number of instructors and students, all of this information needs to be stored somewhere, so that the client can properly access and utilize this data. Secondly, a database also allows instructors to have easy access to the information gathered from the surveys. Using simple queries, the data associated with each instructor can be accessed without delay.

Originally, due to the sensitive nature of information used (i.e., student IDs), encryption was going to be used when storing these items. However, upon further discussion with the client, personal information was eliminated from the database and instead, access codes for each instructor were created and nothing pertaining to the students was stored.

4.3 Implementation

4.3.1 Technology Used

The main concept of this project is to allow instructors to visualize the results of their surveys, so that they can better understand how to educate. Some of the core technologies used pertain to HTML, databases, and JavaScript. HTML is used to create the necessary survey forms. To make the front-end design easier and more reliable, we also used Bootstrap, jQuery, and CSS.

4.3.2 Database Setup

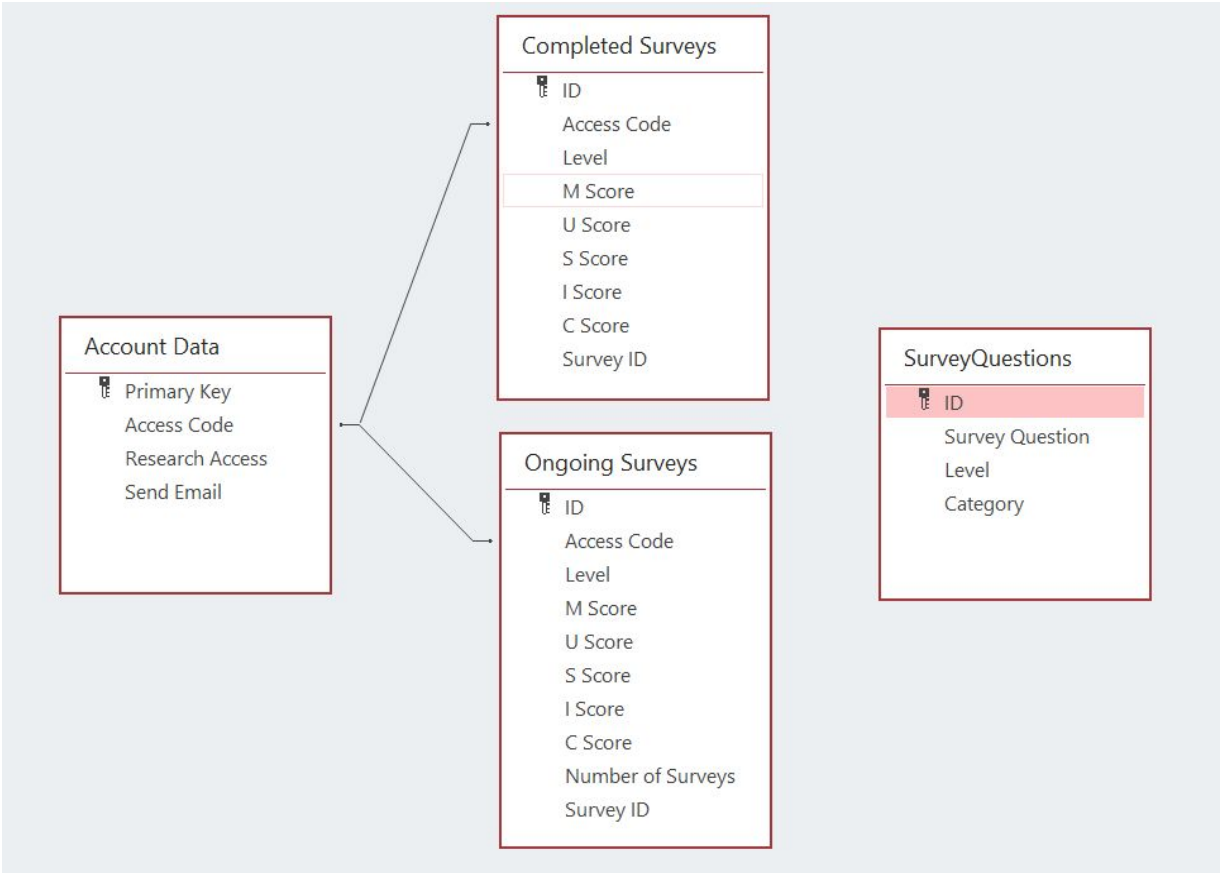

Figure 21. Database Schema

The client's website is on a shared web server hosted by Bluehost. Our local website was integrated into the existing website that our client has. As mentioned earlier, a database was necessary, so that the survey results and teacher login information could be stored and maintained. To make the implementation easier, we chose to use the existing MySQL database support that Bluehost has. With this, we did not need to look for another database support software and could also ensure that our formatting and setup was similar to the client's.

As seen in Figure 21, the database schema comprises four main tables: account data, completed surveys, ongoing surveys, and survey questions. The account data table stores data for each of the instructors. Since the client did not want any personal information stored, we created an access code that corresponds to each instructor. The other data elements in this table are "research access" and "send email". "Research access" is a flag (yes/no) that indicates whether or not the instructor gives the client permission to use their results in his study. "Send email" is another flag (yes/no) that indicates whether or not the instructor would like to receive an email from the client detailing what the data would be used for.

"Ongoing Surveys" and "Completed Surveys" are two similar tables and contain almost the same data elements. Each table contains an access code, which is linked to the instructor's access code (and hence, the "Account Data" table), as seen in the schema. In the "Ongoing Surveys" table, the number of surveys represents the number of students who have submitted answers for that particular survey. For both ongoing surveys and completed surveys, the MUSIC scores from the surveys are stored in the table, however the numerical value has a different meaning for each one.

In "Ongoing Surveys," the individual MUSIC scores are cumulative - that is, each is a summation of all of the MUSIC scores generated from each student submission. In "Completed Surveys," the individual MUSIC scores are an average of all of the submissions. The MUSIC scores from the "Ongoing Surveys" table are averaged, using the number of surveys as the basis. As a result, the numerical values of the MUSIC scores in "Completed Surveys" are standardized to a scale of 1 to 6, whereas the ones in "Outgoing Surveys" are far greater.

4.3.3 Access Code Generation

To generate the access code, we used a simple hashing function on a string - in this case, the instructor's last name. For security/privacy purposes, this method is disclosed to the client only. As mentioned earlier, the client did not want any personal information to be stored, so there

is a disclaimer on this page to let the user know that it would not be used. This access code is then associated with the instructor and can be used to access their results.

We handle collisions of pre-existing hash codes using a simple quadratic probing method similar to what is shown in Figure 22 [5].

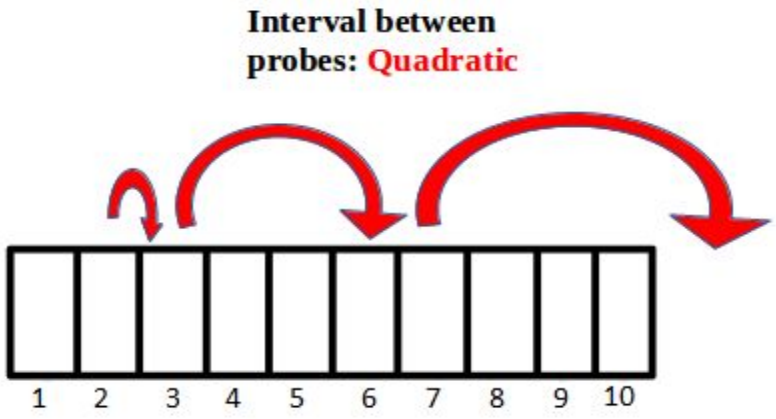

Figure 22. Quadratic Probing Flow Diagram

4.3.4 Access Code Web Page

The instructor is able to generate a new access code based on their last name if they do not already have one. Once they click the "generate access code" button, the access code will appear under the button, as shown in Figure 23. The instructor will have to copy down the access code in order to keep track of it.

Figure 23. Access Code Generation Web Page

4.3.5 Permission Web Page

After the instructor generates an access code, they will be redirected to a permissions page, the first time they log in with their access code, as seen in Figure 24. This allows them to set the permissions to share their data with Dr. Brett Jones, since this entire project is part of his research. The instructor is able to change the permissions at any time through a button on the dashboard.

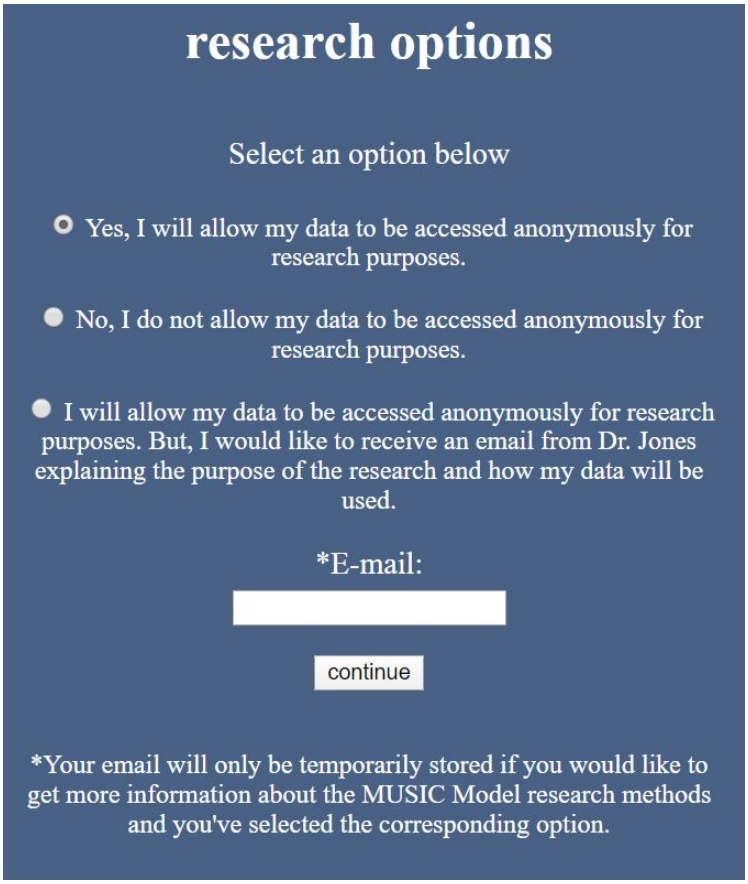

Figure 24. Instructor Options to Share MUSIC Survey Data

4.3.5 Base Survey Form Building

The basic website form originally consisted of fields corresponding to personal information, course information, and survey questions. Personal information included student ID, name, email, and grade level. Similarly, course information covered course ID, name, school ID (where the course was offered), and professor information.

Anything pertaining to personal information was later dismissed. The client did not want any personal information to be stored, regardless of encryption in the database itself. As a result, these fields in our forms were removed. Since we didn't need to store this information, the form became fairly simple - all it needed to contain was the survey questions itself.

As stated by the client, the manner in which the questions are given affects the accuracy of the survey. Most forms tend to group questions of the same type together, and in doing so, users eventually start to get some understanding of what the survey is asking for. We want to avoid this situation, and so the questions will be taken from a question bank and displayed in a randomized order.

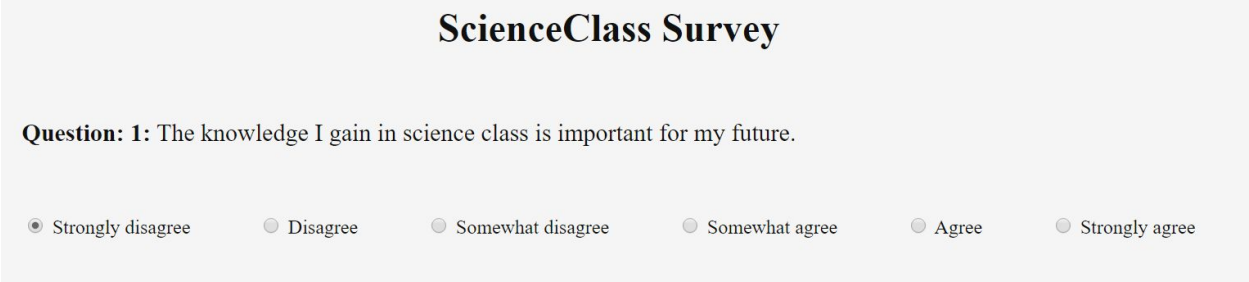

Figure 25. Layout of Survey Questions

We also intend to word the questions in such a way that not only are they not leading questions, but are more holistic. Questions that ask if the instructor did a "good job" or a "bad job" are not necessarily helpful. It does not focus on the specifics of the class or instructor, which the MUSIC model hopes to resolve. We followed our client's guidelines to implement an unbiased survey by using the questions provided in his user manual.

4.3.7 PHP Queries to Database

phpMyAdmin, an administration tool for MySQL, allowed us to access and alter the MySQL database using a web browser. This was particularly useful for the website, since we could use SQL queries within our code to send and store the answers in the database. It also let us use the existing results in the database for the data visualization. Additionally, phpMyAdmin facilitates maintenance of survey results for the client; little technological understanding is required for managing the tables and data, so the client need not be bothered with the specifics.

Due to the reasons described prior, we used PHP for integration between the front-end and the database. The existing website that our client has is based on WordPress, which uses PHP. Using PHP as our back-end development language eased the development and integration.

4.3.8 Data Visualization

The data gathered from survey results was eventually represented in dynamic and interactive tables and charts. For instructors to easily understand their results, the data visualization feature lets them view their results in different categories corresponding to each survey separately. The chart updates to show the visualization of the survey selection (see Figure 26).

In order to implement this, we use the D3.js library. It produces dynamic and interactive visualization (graphs, charts, etc.). Instructors will be able to clearly see the results of their surveys and have the option to view data by course, year, or any other useful categories.

When a survey is selected in the dashboard, the back-end code calls a JavaScript function to create a D3 visualization function in the chart div. The function clears any pre-existing chart, and then performs a PHP query to get the data based on the access code from the selected survey.

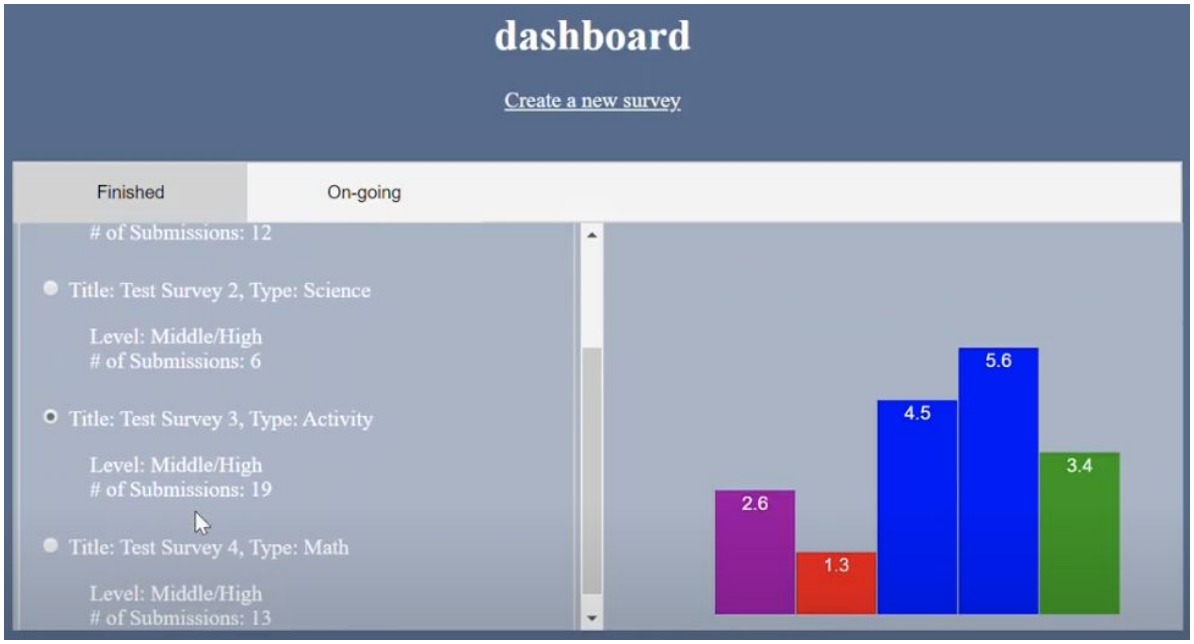

Figure 26. Current Model for Visualization of Data

4.3.9 Alternate Surveys

It's improbable that one survey will be able to be used for all courses, instructors, and grade levels. Many instructors will likely want to target certain focuses of MUSIC for each of their courses and instructors often teach different grade levels. The same survey for college students would not apply to elementary schoolers, nor would it have the same complexity. As a result, we created different types of general forms for various ages/grades and changed some of the parameters. For instance, a survey targeted towards high schoolers will be more targeted towards usefulness of the coursework, as well as the specific course type (e.g. mathematics, science, English, etc.).

In addition to this, each survey also has different variations. These include present tense, past tense, and course/major-specific questions. For instance, the present tense surveys may be administered towards the beginning and middle of the course (see Figure 6), whereas the past tense surveys would likely be administered towards the end of a course (see Figure 7).

1. The coursework holds my attention.
2. I have the opportunity to decide for myself how to meet the course goals.
3. In general, the coursework is useful to me.
4. The instructor is available to answer my questions about the coursework.
5. The coursework is beneficial to me.

Figure 27. Sample Present Tense Questions in the College Survey

1. The coursework held my attention.
2. I had the opportunity to decide for myself how to meet the course goals.
3. In general, the coursework was useful to me.
4. The instructor was available to answer my questions about the coursework.
5. The coursework was beneficial to me.

Figure 28. Sample Past Tense Questions in the College Survey

4.3.10 Languages

As per our client's request, we looked into posing the survey to students in various languages, so that they could choose whichever language best suited them. There are several reasons for this: 1) international students and students who speak a different language at home may feel more comfortable viewing/understanding the survey in their native language and 2) since this project is an international one, it would be helpful to distribute the survey in the language corresponding to their region. It's possible that most students will view the survey in English, but we wanted this option to be there regardless.

As for the implementation specifically, HTML contains tags directly corresponding to language and can display the survey in a different language accordingly. Based on geolocation, Google also automatically enables a pop-up allowing the user to translate the page to the language of the region.

5 Testing

5.1 Database Testing

We did two stages of database testing: pre-integration and post-integration. The initial testing was done to ensure that the fields and table layout were appropriate for the project. The bulk of this testing was done on Microsoft Access. As was earlier shown in Figure 10, the tables were set up. Then, we used various Query Wizard setups in access to test the different POST and GET methods we would be using the website.

The next stage of testing for the database was done through DigitalOcean. Prior to having access to the client's website backend, we hosted our own web pages and database on DigitalOcean. This testing was done in conjunction with our web page testing.

5.2 Web Page Testing

Testing the web pages was a simple matter of getting the sites running on our host and connecting them to the database. We created a few test access codes in order to be able to create test surveys and submit them to the database. The details of the example survey can be seen in Figure 15. We were able to test and port the web pages without too many issues.

We tested the web pages on different browsers, including Chrome, Firefox, and Edge. The functionality remained the same over the browsers, though the design varied inconsequently.To fix the design issue, we modified the CSS to be portable across platforms by enclosing the different sections in divs.

5.3 Visualization Testing

A key feature of this project was creating the visualization of the music scores. Although the process of creating the bar chart was just a matter of using a PHP query to get data from the database, the difficult part of the visualization was ensuring that the chart was clear to read and understand. In the initial creation, we tried to match the color scheme of the website with the bars, but found that it was difficult to distinguish as seen in Figure 18. In the end, we went with a color scheme based on the score, so that the different scores could be distinctly recognized and emphasized the size difference in the bar length.

Figure 29. Initial D3 Visualization of MUSIC Scores

6 Lessons Learned

6.1 Timeline

Task	Date To Complete	Completion
Meeting with client	1/28/20	100%
Project Description Approved	2/2/20	100%
Mockups for survey pages	2/12/20	100%
Presentation 1	2/18/20	100%
Basic front end forms/ conditionals and access	2/24/20	100%
Database structures and relational models	3/9/20	100%
Data visualization models	3/15/20	100%
Interim Report	3/28/20	100%
User Interface and Database Connection finished	3/29/20	100%
Presentation 2	4/2/20	100%
Server Testing	4/12/20	100%
Integration into Website	4/20/20	100%
Website Testing	4/25/20	100%
Final Presentation	4/30/20	100%
Integration and Server Testing (BlueHost)	5/4/20	100%

6.2 Challenges

Some of the main challenges we faced throughout our project were working with HTML formatting, calculations, and communication. Since the surveys were generated for specific levels (e.g., high school, college, etc.) and types (e.g., past, present tense), we could not simply hardcode the survey questions. Regardless of this fact, hardcoding wouldn't have been ideal, given that not all of these surveys had the same number of questions. Further, there was also the possibility that the client may wish to change the questions posed in the survey.

This meant that we needed to dynamically generate the survey questions in the form. This proved to be difficult for one main reason: each question needed to be associated with an ID, so that the MUSIC score for the survey could be calculated and then sent to the database. Accessing the survey questions from the database was easy enough, but actually displaying them on the HTML form required more work.

As briefly mentioned earlier, there were slight communication issues during the transition to virtual classes. Maintaining contact with the client was a bit difficult, so we did not get the client's website credentials until nearly the time of our project deadline. Final integration and testing with the client's website was done in far less time than originally intended.

6.3 Solutions

Using simple PHP queries, we were able to access all of the correct survey questions, but the issue lay in displaying it in the form. In approaching this, we primarily utilized JavaScript. We first created an empty div element in the body of our HTML page. This is where all of the questions would be displayed, once populated.

JavaScript is usually used in these kinds of situations, especially when you want to make a website more dynamic. We used it for form validation and dynamic changes of content on our page. Using the innerHTML element, we were able to iterate through the questions we accessed and subsequently display them on the page. However, we also needed to make sure that the questions could be linked to the corresponding MUSIC attribute. That is, we needed to ensure that a question that represented empowerment (M) would be used to calculate the empowerment (M) score, not the usefulness (U) score.

To do so, when displaying the questions on the page, as described before, we created an ID for each of these elements. We used the unique ID that corresponded to each of the questions,

as seen in Figure 10. The database schema in this figure also shows a category for each of the questions. This category represents one of the MUSIC attributes. So, when calculating the MUSIC score for the submitted survey, we could cross-reference the ID with the category and accordingly perform summations before sending it to the database.

7 Future Work

Groups working on this project in the future might want to consider the following: cleaner design and multiple submissions.

From the figures provided throughout the paper, it is noticeable that the style and formatting of our website did not entirely match up with the client's website. Though functional, it could stand to benefit from stylesheets that are similar to the client's. It is also important to note that although the website could match the style of the original website, it should not include any images or clutter that might cause bias towards any answers in questions.

Lastly, and perhaps the largest focus, would be multiple submissions. Since the client did not want any personal information to be stored, we had to scrap our original implementation of storing (an encryption of) the students' IDs. As our database schema stands, there is no storing of individual survey results; the only remotely related value that changes is the number of surveys, which increments with each submission. Time constraints prohibited us from completing the multiple submissions feature, but with a little work, this could be implemented.

8 Acknowledgements

The team would like to acknowledge our clients Dr. Brett Jones and Dr. Juan Cruz for their work on the survey and initial website. We would also like to thank the students of our class as well as our professor, Dr. Fox, for providing constructive criticism and support on our project.

9 References

[1] B. Jones, “MUSIC Model of Motivation,” MUSIC Model of Motivation. [Online]. Available: https://www.themusicmodel.com/. [Accessed: 06-May-2020].

[2] “Bluehost,” Bluehost. [Online]. Available: https://www.bluehost.com/. [Accessed: 06-May-2020].

[3] B. Jones, “User Guide for Assessing the Components of the MUSIC ® Model of Motivation”. [Accessed: 06-May-2020].

[4] M. Bostock, “Data-Driven Documents,” D3.js. [Online]. Available: https://d3js.org/. [Accessed: 06-May-2020].

[5] C. Leung, “Quadratic Probing and Double Hashing,” Quadratic Probing and Double Hashing · Data Structures and Algorithms. [Online]. Available: https://cathyatseneca.gitbooks.io/data-structures-and-algorithms/content/tables/quadratic_probing_and_double_hashing.html. [Accessed: 07-May-2020].