NEO Learning Systems

design document – Project Raptor

teacher app

july 2017

VERSION 1.2

CONFIDENTIAL

# Table of Contents

[1 Table of Contents 2](#_Toc489268622)

[2 Overview 4](#_Toc489268623)

[2.1 Project Objectives 4](#_Toc489268624)

[3 Teacher Role 5](#_Toc489268625)

[4 Teacher Login 5](#_Toc489268626)

[4.1.1 Browser Session Storage 6](#_Toc489268627)

[4.1.2 Tables to Use 6](#_Toc489268628)

[5 Classes Tab 7](#_Toc489268629)

[5.1.1 Filter section 8](#_Toc489268630)

[5.1.2 Average assessment scores graph 9](#_Toc489268631)

[5.1.3 Strands Graph 11](#_Toc489268632)

[5.2 Class Score Details 12](#_Toc489268633)

[5.3 Student Assessment Details 14](#_Toc489268634)

[6 Curriculum Tab 16](#_Toc489268635)

[6.1.1 Filter Section 17](#_Toc489268636)

[6.1.2 Curriculum Scores by Strand Graph 17](#_Toc489268637)

[6.1.3 Curriculum Chart 19](#_Toc489268638)

[6.2 Student Strand Scores 20](#_Toc489268639)

[6.3 Student Assessment Details 22](#_Toc489268640)

[7 Student Tab 25](#_Toc489268641)

[7.1.1 Assessment Overview 25](#_Toc489268642)

[7.1.2 Strand Details 27](#_Toc489268643)

[7.2 Edit Student 31](#_Toc489268644)

[8 Scorecard 31](#_Toc489268645)

[9 Groups 35](#_Toc489268646)

[9.1.1 Adjust Grouping Constrains 35](#_Toc489268647)

[9.1.2 Grouping Results 36](#_Toc489268648)

[9.1.3 Buttons 36](#_Toc489268649)

[9.2 Create Group From List 36](#_Toc489268650)

[9.3 Create New Group 38](#_Toc489268651)

[9.4 Modify And Delete Group 39](#_Toc489268652)

[10 Question Rating 41](#_Toc489268653)

[11 WHiteboard 42](#_Toc489268654)

[11.1.1 Question Selection 42](#_Toc489268655)

[11.1.2 White board 43](#_Toc489268656)

[12 Tickets 44](#_Toc489268657)

[12.1 Setup 45](#_Toc489268658)

[12.2 Review 46](#_Toc489268659)

[13 Feedback 47](#_Toc489268660)

[14 Color Palette 49](#_Toc489268661)

[15 Figure Guide 50](#_Toc489268662)

[16 Non Disclosure 53](#_Toc489268663)

# Overview

Neo Learning Systems (NLS) is a software provider of classroom testing and analytics for the education market. Our software is directly tied to the Ontario standardized mathematics curriculum. The software is role based and provides user interfaces for students, teachers and administrators of varying levels.

## Project Objectives

The project must meet the following general objectives (specific objectives and project deliverables will be covered in other sections of this document).

* Change the UI to meet the guidelines set by the Ontario Curriculum
* Create the appropriate level of application security for the Students.
* Responsiveness of the Project for Students to be able to view in any device in Chrome browser.
* Deliver the changed functioning applications by July 2017.

# TeacherRole

The following sub-sections correspond to the various screens or tabs that have changed to the changing requirements of the customer.

# Teacher Login

The login screen for the teacher app is pictured below in Figure 1. A user should only be able to log in via the login page by entering the login details. If the user tries to login using the browser forward/ back button they are redirected to the login page.. If the user is inactive for more than 2 hours the application should also auto log out and be redirected to the login page.

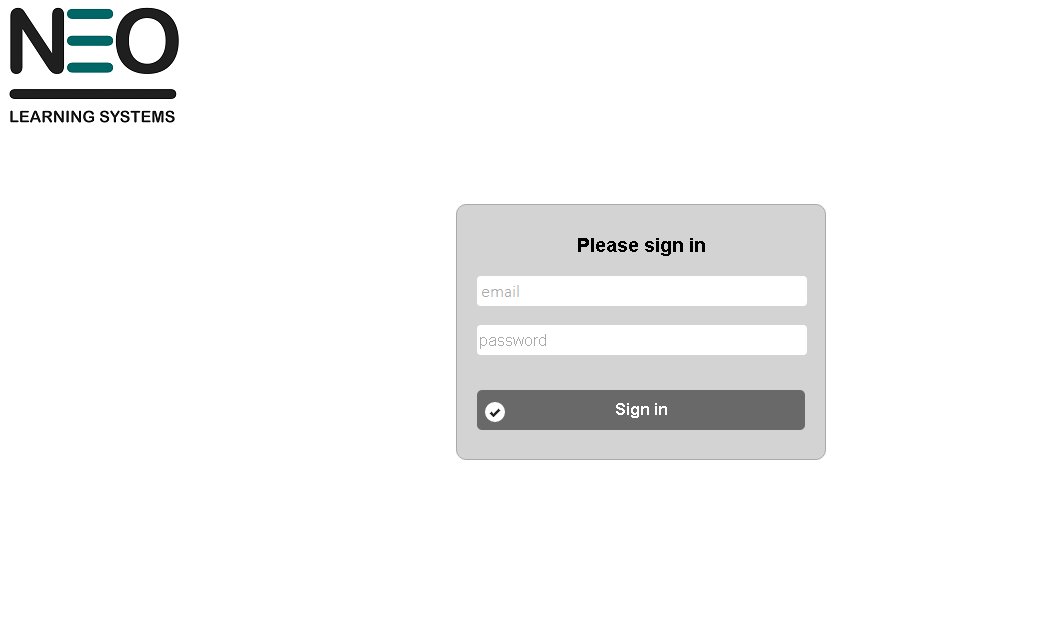


Figure 1: Login screen for the Teacher user role

They should not be allowed to use the forward/back button of the browser. In doing so the user should be redirected to the login page. The menu includes Classes, Curriculum, Student, Scorecard, Groups, Question Rating, Library, Assessment, Templates, Calendar, Pending, Whiteboard, Tickets, Feedback; details for this will be explained further in the document. The menu page is pictured as below in Figure 2. Once logged in the landing page should be the Classes module.

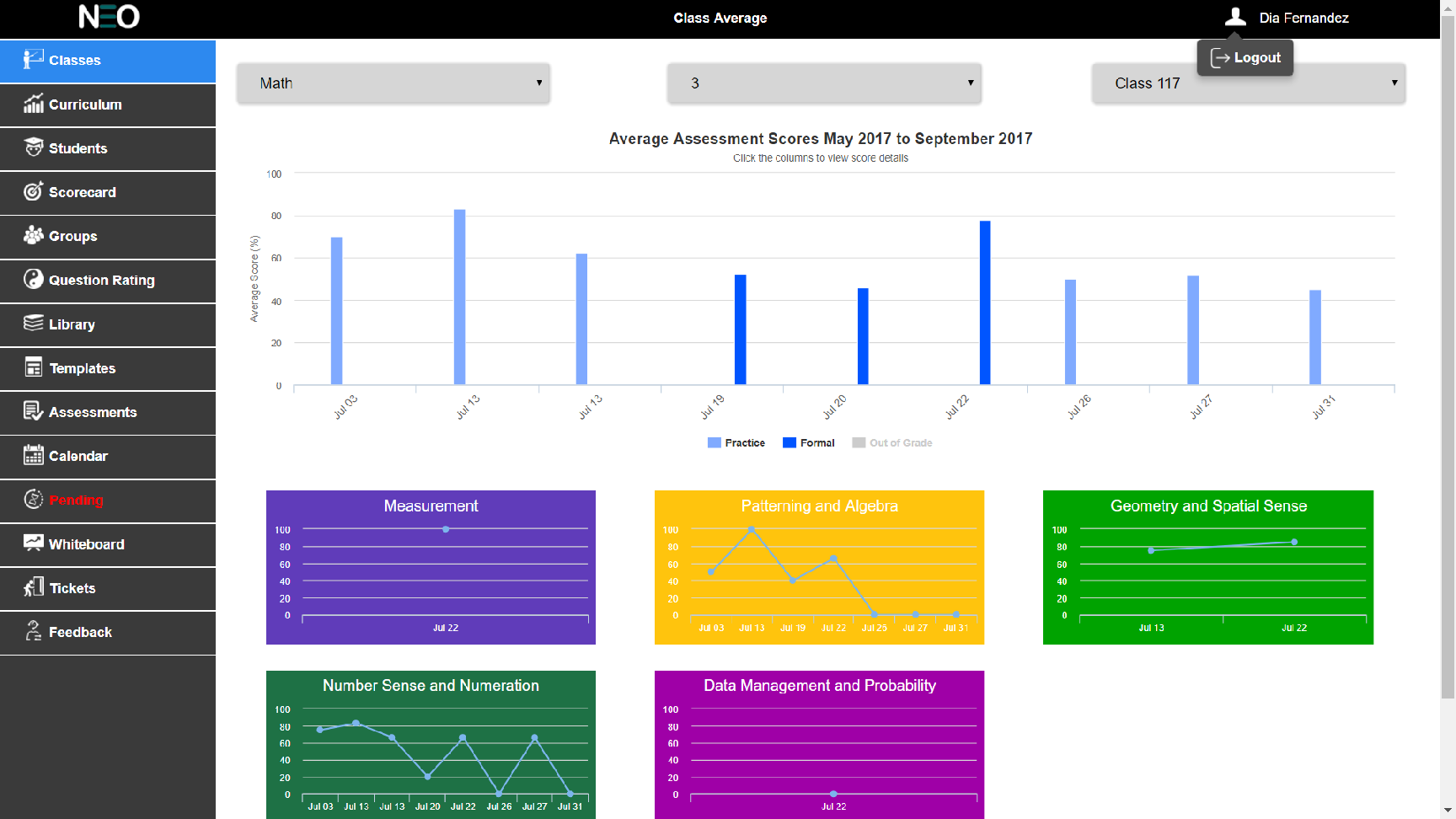


Figure 2: Menu screen for the Teacher user role

### LOGOUT

(Write about how the user logouts)Whenever a user logs out of the application, they should be brought back to this login screen and any sessional data stored within the browser should be deleted. If the user is inactive for more than 2 hours the application should also auto log out and be redirected to the login page.

### Browser Session Storage

The following values will need to be queried from the database after login and placed in the web browser’s session storage for later use on other pages:

| User ID | Role |
| --- | --- |
| First Name | Last Name |
| Location ID (for the school) | Location Name |
| Location ID (for the board) | Board |
| Grades (for the Teacher role) | Courses (for the Teacher role) |
| Province |  |
|  |  |

### Tables to Use

Query the following tables to retrieve the relevant data: User, ClassUser, Class, Location, Board and Ministry.

**Additional Notes about this screen:**

* All logos, graphics and style files can be found within the application repository on GitHub.

# ClassesTab

The classes tab is useful to view the student assessment results from the perspective of the entire class. All data in this tab is read-only, the data cannot be changed, modified or deleted. All of the top-level data analysis tabs have common UI elements. On the Classes tab, these common elements include the Grade, Course and Class drop lists to filter/select the data. The classes tab look like below Figure3**.**

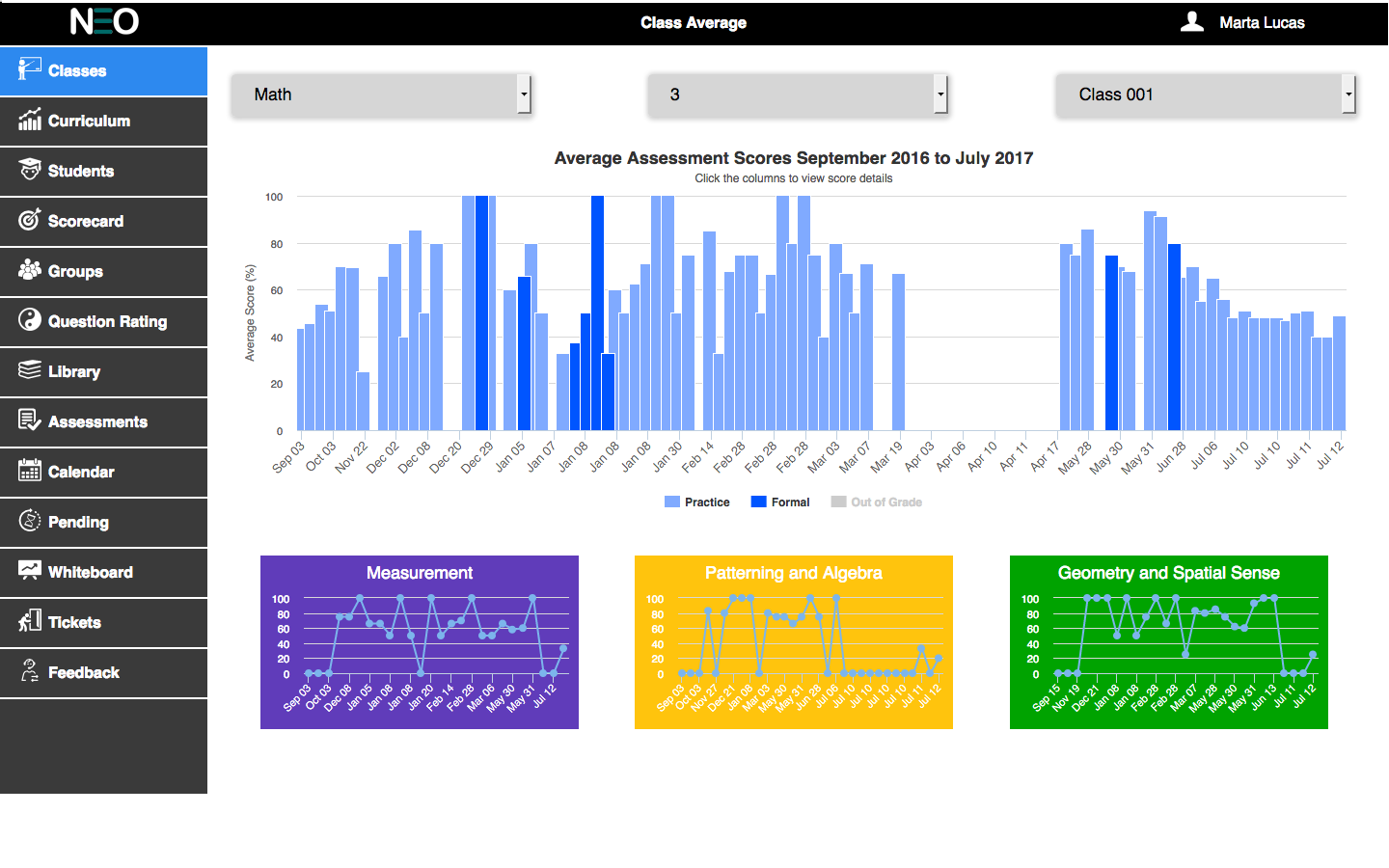


Figure 3: Classes Tab

All of the top-level data analysis tabs have common UI elements. On the Classes tab, these common elements include the Grade, Course and Class drop lists to filter/select the data.

The UI elements that are unique to this tab/page are the main graph that stretches over the width of the screen. There are also color coded graph that represent course strands. The graphs on this screen have varying levels of interactivity which include tooltips that appear as the pointer rolls over a data point or bar. The individual bars should be color coded by test type (formal, practice and out of grade). For the color codes for the different test type refer Section 3.2.2.

The main graph is drillable that is on clicking on a bar in the graph will take the user to another screen with a deeper analysis of the data.

All of the graphs developed for the software should make use High Charts. A licensed copy of the High Charts software has been provided by NLS.

### Filter section

The top level data analysis looks like below (Figure 4):



Figure 4: Top Level Filters

The order of the drop lists should be: Course, Grade and Class.

The Course drop list should be a filtered list of Courses that are being taught by the user currently logged in.

The Grade drop list should also be populated by a database query using the selected course above and the currently logged in User ID as filters.

The Class drop list should be populated by a query using the selected course, the selected grades, and the currently logged in User ID as filters.

**By default, the Course, Grade and Class should be set to the first available options for the logged in user. This way, some default data can be displayed to the user when they first log in or select this tab. This should be common for all tabs related to data analysis.**

### Average assessment scores graph

The Average Assessment Scores graph is the main graph on this tab (see Figure 5 below). This graph shows the average scores for the user selected class over time; as such, the number of bars on the chart will vary depending on the number assessments that currently exist for this class. The displayed assessments should be a filtered, date sensitive list. The valid dates for this list of assessments should be from Start Date to the End Date of the selected class; this data can be found in the database Class table.

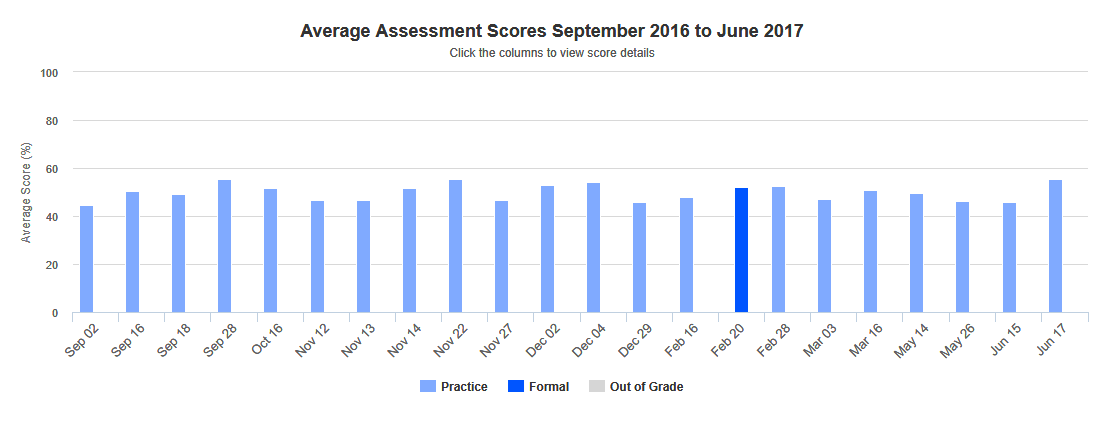


Figure 5: Average Assessment Scores Graph

The graph title should read “Average Assessment Scores: “followed by the start and end dates of the selected class. The subtitle should read “Click the columns to view score details”. The y-axis title should read “Average Score (%)”. There is no need for an x-axis title.

The y-axis should display data from 0 to 100 with intervals of 20. The x-axis data points should be the short date of the specific assessment. The date format used is as month and date.

The color of the bars on the graph will vary by assessment type (Formal or Practice; the Out of Grade is disabled by default), and if the mouse pointer rolls over a bar. Use the Hex color table below (Table 1) as aguide (see the Color Palette section for specific details).The out of grade legend will be disabled by default. If the user wants to see the scores of the student scored in out of grade then the user have to enable by clicking on it.

|  |  |  |
| --- | --- | --- |
| Practice Assessment | #80AAFF |  |
| Practice Assessment Rollover | #B3DBFF |  |
| Formal Assessment | #0055FF |  |
| Formal Assessment Rollover | #B3DBFF |  |
| Out of grade Assessment | #D6D6D6 |  |
| Out of grade Assessment Rollover | #EFEFEF |  |

**Table 1: Assessment Graph Color Palette**

Rolling the mouse pointer over a specific bar should reveal a tooltip with specific information about that data point (see Figure 6 below). The tooltip should have a title (Class Average), the specific test date, and the percent average for the class on that test.

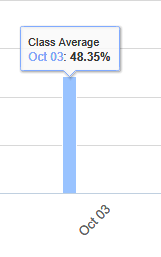


Figure 6: Tooltip for Class Average Graph

Each of the bars on the graph is user selectable. Clicking on a bar will drill down into the data for a deeper analysis of the assessment that the selected bar represents. Clicking on a bar takes the user to screen showing a breakdown of the individual student scores that make up selected class assessment.

Using the demo as an example, clicking on the October 3rd bar, drills down into that specific test and displays individual student scores for that test. See the Section 3.3 “Class Score Details” for more details on this.

### Strands Graph

At the bottom of Classes tab (see Figure 7 below), there will be a varying number of graphs showing data for the individual strands for the user selected course. For example, the math curriculum for grade 3 has 5 strands associated with it, so there should be 5 individual charts displayed on the Classes tab.

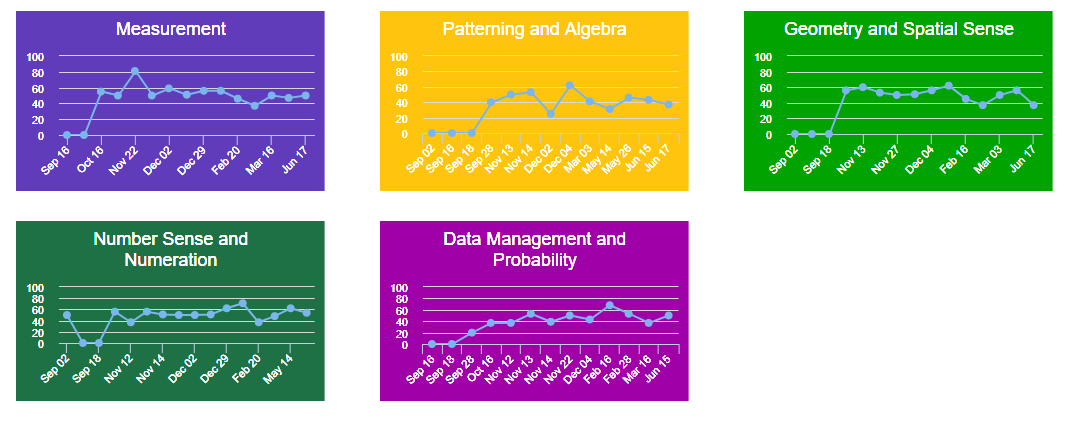


Figure 7: Curriculum Strands Graph

These graphs should be color coded by strand (following the guide in the Color Palette section of this document), and the strand colors should be consistent throughout the software.

Depending on the number of strands for a selected course, it will be acceptable to split the charts across two rows. For example, the grade 3 curriculum has 5 strands, so the first row would have 3 charts followed by a second row of 2 charts. All the charts should be a consistent width.

Rolling the mouse pointer over a data point on the graph should display a tooltip with the following: a title (Assessment Date), the test date, and the class average for that strand on that specific test date (see below Figure 8).

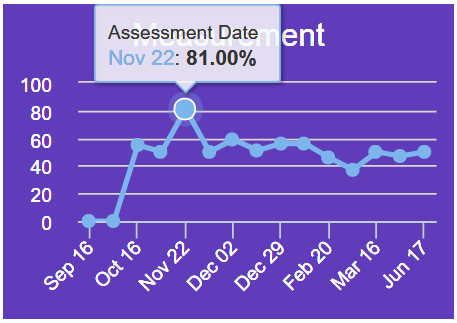


Figure 8: Abbreviated Date & Tooltip for Strand Graphs

## Class Score Details

The Class Scores Details screen displays a breakdown of the individual student scores for the user selected assessment from the prior screen (Figure 9 below).

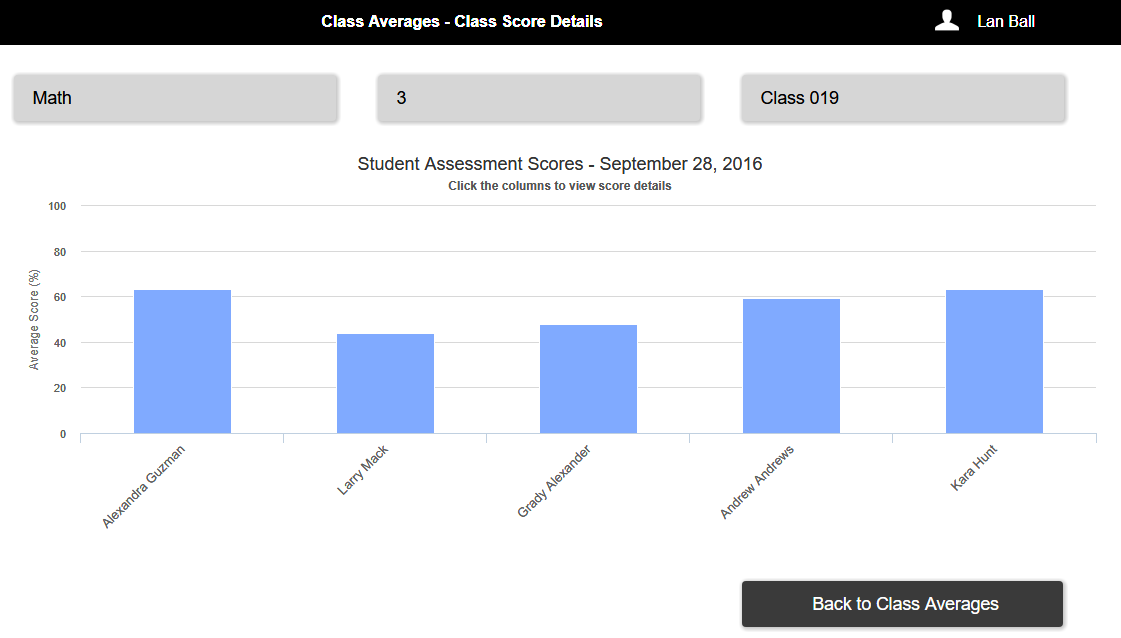
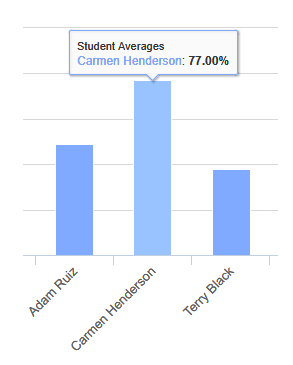


Figure 9: Class Score Details Screen

Across the top of this screen, the user can see the selected Course, Grade and Class. While these look like the filter drop lists for UI consistency, these are read-only. These selections can only be changed by returning to the prior screen (Class Averages) and changing the filter there. This type of read-only UI element is common across all secondary UI screens in the software. The order of the displayed filters should be Course, Grade and Class.

Rolling over the filter selections should change the mouse pointer to indicate that the filters are read only (see below Figure 10).



**Figure 10: Graph Labels and Tooltips**

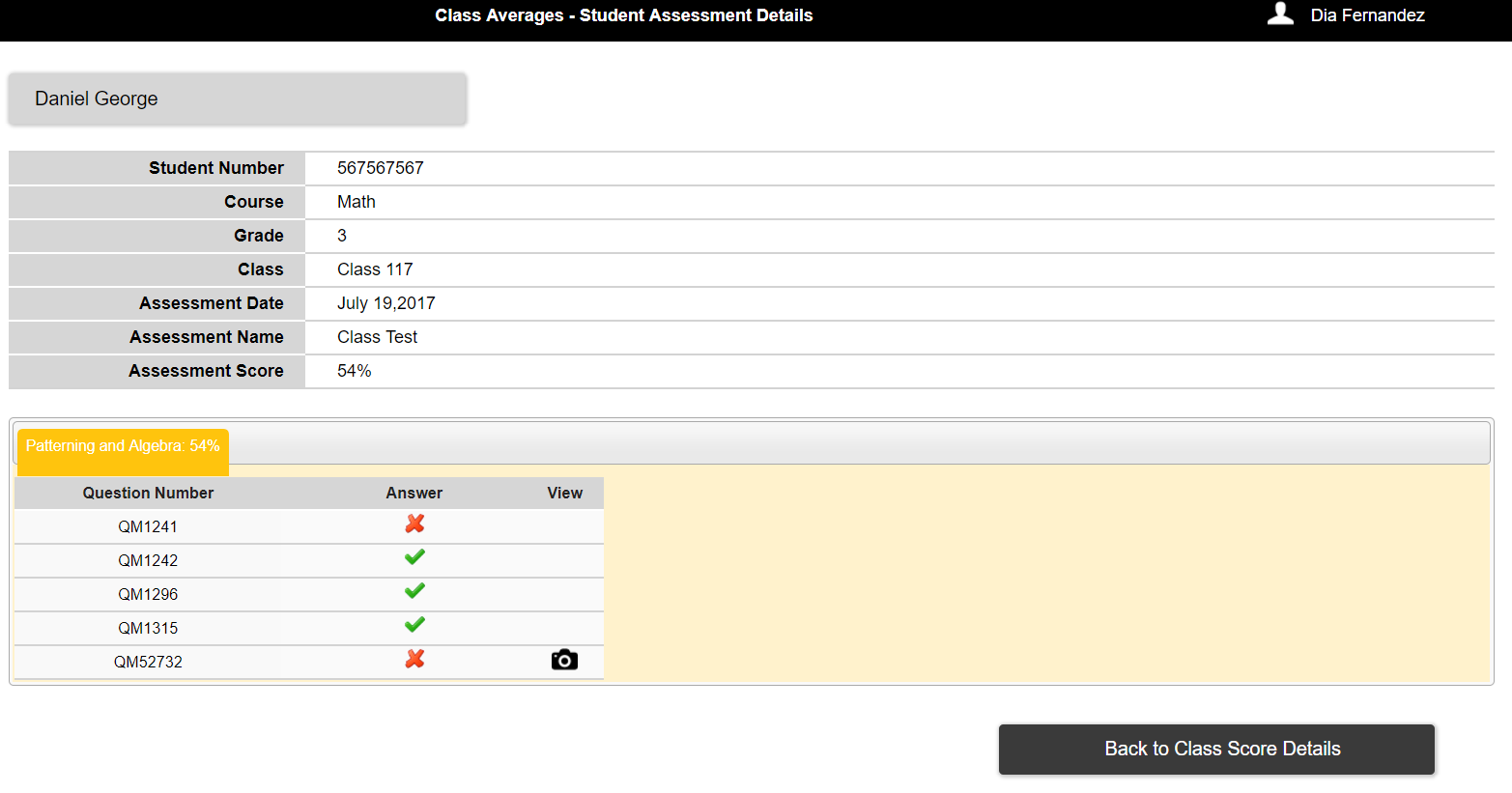
The central graph should have a title of “Student Assessment Scores – “, followed by the date of the selected test. The subtitle should read “Click the columns to view student score details”. The y-axis title should read “Average Score (%)”. There is no need for an x-axis title or graph legend. Each x-axis data point should be labeled with the students name (at an angle for space saving, see Figure 9). Each bar should also show the numeric value of the student’s test score.

Rolling over a bar on the graph should reveal a tooltip with the following information: title (Student Average), the student’s first and last name, and the student’s test score. Bar coloring and highlighting should function and is the same as in the prior screen. Bars are user selectable and will take the user to another analysis screen; drilling down deeper into the data to the Student Assessment Details screen.

The “Back to Class Averages” button at the bottom the screen simply returns the user to the prior screen (all user selections should still be in force).

## Student Assessment Details

The Student Assessment Details screen is the lowest level that the teacher can drill down to. It shows information about the specific student as selected from the prior screen, and for a specific test.

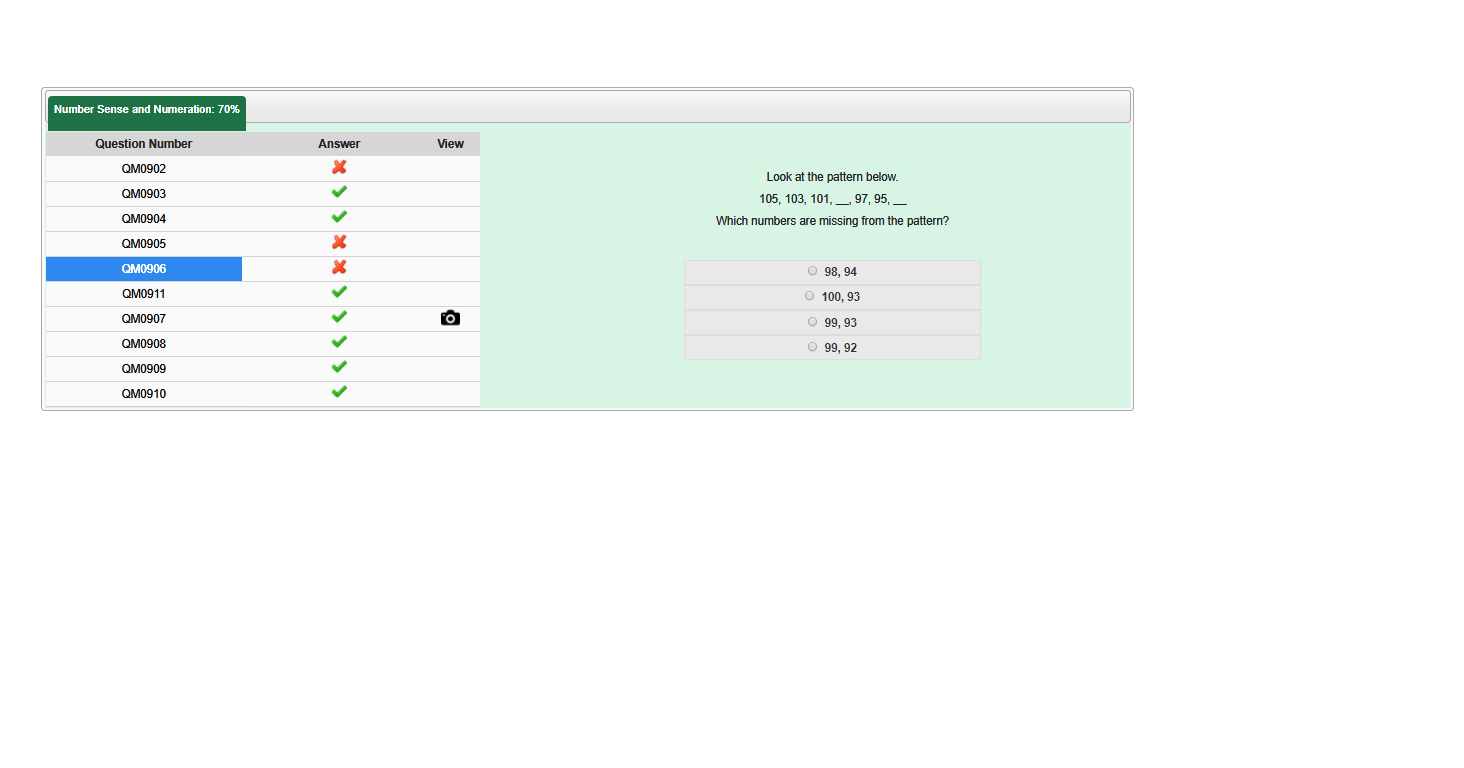


**Figure 11: Student Assessment Details Screen**

The top portion of this screen should display a combination of information about the selected student as well as the selected data filters that got the user to this screen. This information includes: the student’s first and last name, student ID, the selected course, grade and class (in that order), the selected assessment date, assessment name and the student’s assessment score for the selected test.

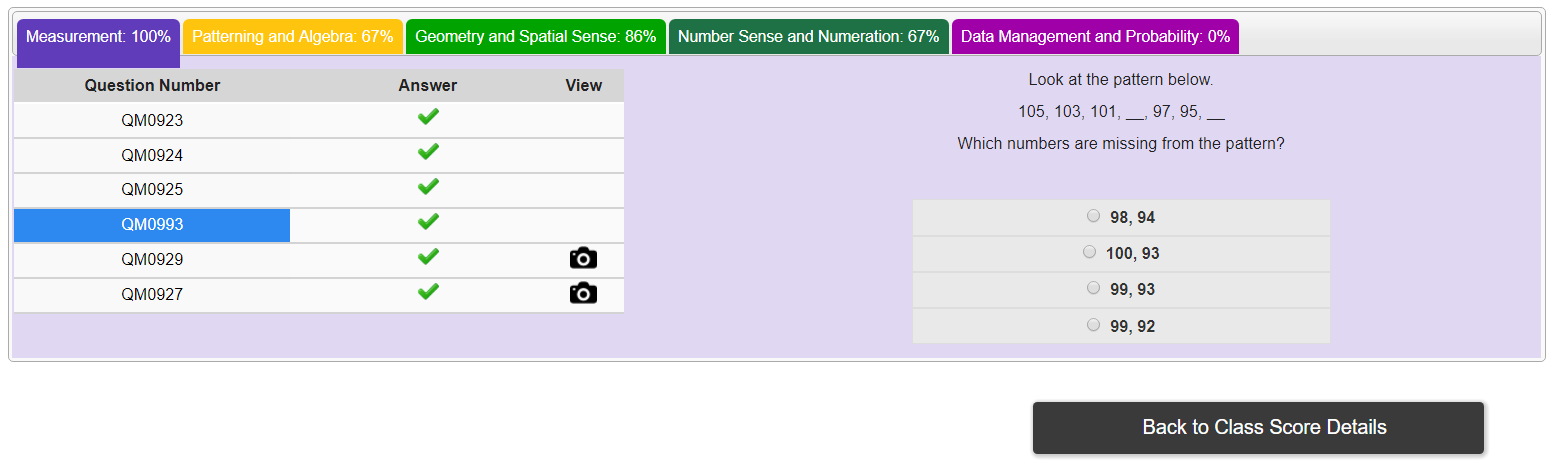
The lower portion of the screen should display color coded tabs only for the strands, for which the student has answered questions in the selected test. In turn, each tab will display all of the test questions that relate to the specific strands for that tab. For example, in grade 3 math, the Measurement tab refers to the Measurement strand and only displays the test questions related to Measurement that were on that specific assessment.

The student’s result for that question (right or wrong) is displayed along with each question ID number (Figure 12).



**Figure 12: Assessment Question Results**

Each question ID is user selectable. Rolling over the question ID with the mouse pointer will highlight it. Clicking on the highlighted question will reveal a preview of that question beside the chart (see Figure 13 below). On click of the camera icon, the user can see the answer written by the student for text input and open response question types.

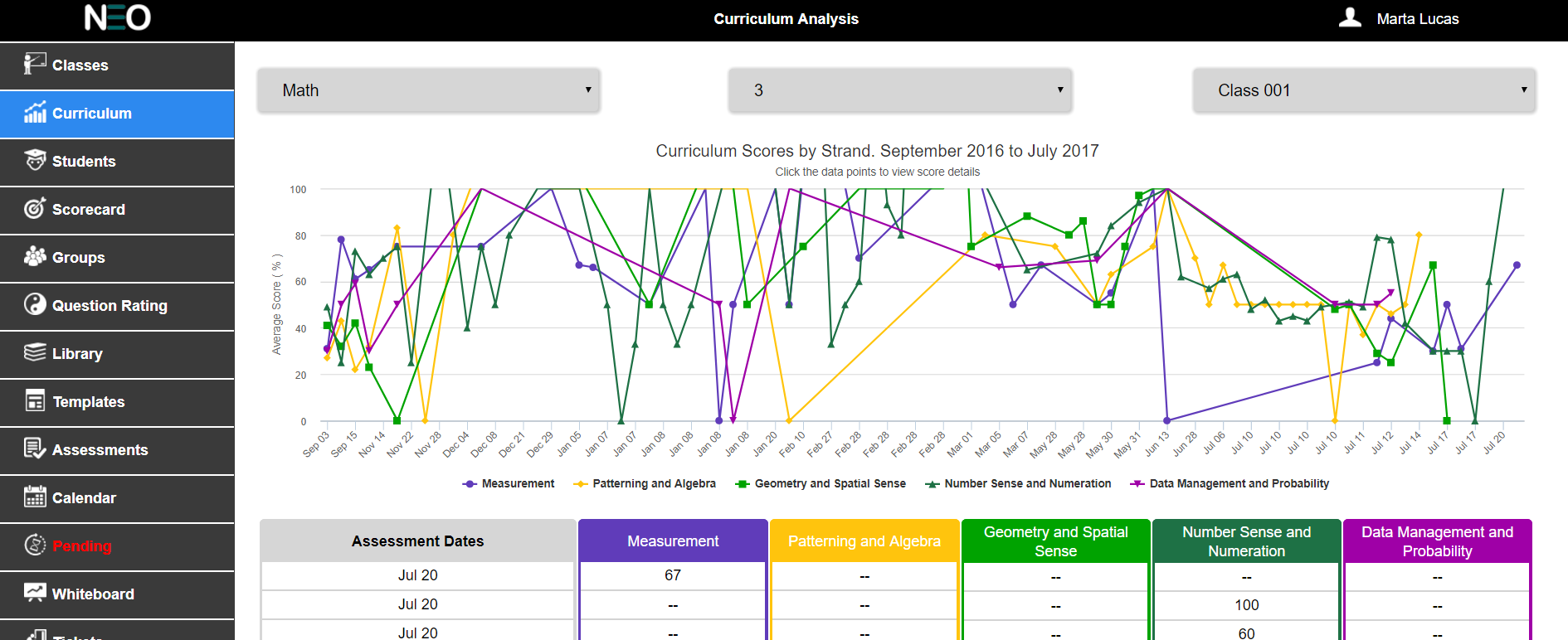


**Figure 13: Question Preview**

The “Back to Class Score Details” button will return the user to the prior screen with all their previous filters and selections.

# Curriculum Tab

The Curriculum tab (Figure 14 below) is used by the teacher role to view student assessment results from the perspective of the standardized curriculum (broken down by Strand). All the data on this screen (and the related drill-down screens) is read-only; the user cannot change, modify or delete any student data in this tab same as in classes tab.



**Figure 14: Curriculum First Page**

Common UI elements on this tab include the Course, Grade and Class drop lists to filter/select the data.

UI Elements that are unique to this screen include the main line graph that stretches to fill the available screen width. There is also a chart that occupies the lower portion of this screen.The graph on this screen is interactive with tooltips that appear as the mouse pointer rolls over a data point. Also, the main graph is drillable so that clicking on a data point on the graph will take the user to another screen with a deeper analysis of the data (see following sections for more detail on interactivity).

The chart is a time ordered list of all the tests taken by the selected class. The chart simply represents numerically the data that the graph shows visually. The recent assessments that were given by the students will be shown on the top of the chart.

### Filter Section

The top level data analysis looks like below (Figure 15):



Figure 15: Top Level filters

The order of the drop lists should be: Course, Grade and Class.

The Course drop list should be a filtered list of Courses that are being taught by the user currently logged in.

The Grade drop list should also be populated by a database query using the selected course above and the currently logged in User ID as filters.

The Class drop list should be populated by a query using the selected course, the selected grades, and the currently logged in User ID as filters.

**By default, the Course, Grade and Class should be set to the first available options for the logged in user. This way, some default data can be displayed to the user when they first log in or select this tab. This should be common for all tabs related to data analysis.**

### Curriculum Scores by Strand Graph

The Curriculum Scores by Strand graph is the main graph on this tab (see Figure 16 below). This graph shows the average scores for the selected class over time and by strand for the selected course; as such, the number of lines on the chart will vary depending on the number strands that currently exist for this course. And the number of data points on the chart will vary depending on the number of tests that have been written by the selected class.

The displayed data should be a filtered, date sensitive set. The valid dates for this list of assessments should be from Start Date to the End Date of the selected class; this data can be found in the database Class table.

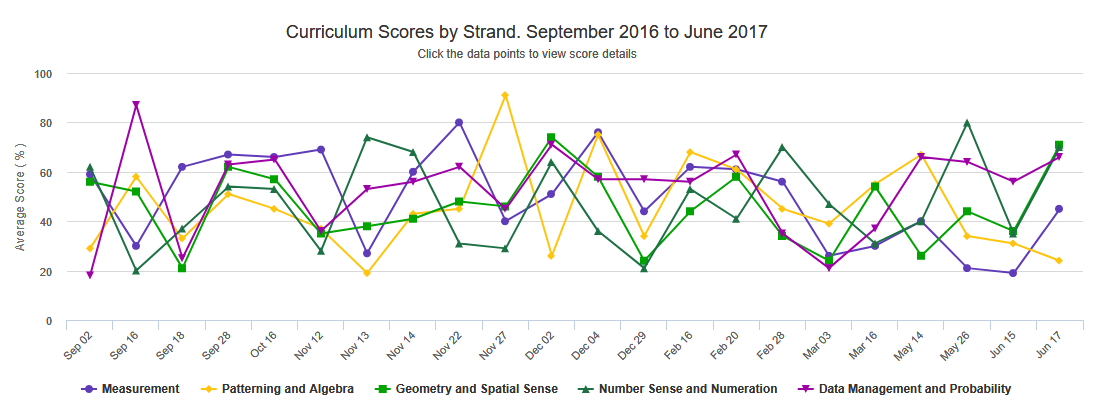


Figure 16: Curriculum Scores by Strand Graph

The graph title should read “Curriculum Scores by Strand: “followed by the start and end dates of the selected class. The subtitle should read “Click the data points to view score details”. The y-axis title should read “Average Score (%)”. There is no need for an x-axis title or a graph legend.

The y-axis should display data from 0 to 100 with intervals of 10. The x-axis data points should be the short date of the specific assessment (select a date format that looks best given that there may be a lot of columns as the year comes to an end).

The data points on this graph should reflect only the Formal and Practice type tests. Individual lines should be color coded by Strand (see Table 2 below this section).

Rolling the mouse pointer over a specific data point should reveal a tooltip with specific information about that data point (see Figure 17 below). The tooltip should have a title (Strand), the specific test date, and the percent average for the class on that test strand.

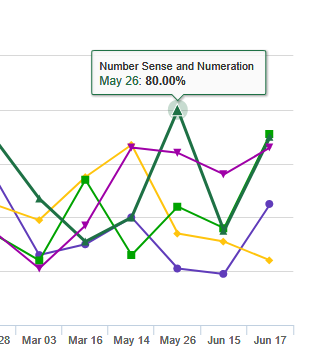


Figure 17: Data Point Tooltip

Each of the data points on the line graph is user selectable. Clicking on a data point will drill down into the data for a deeper analysis of the assessment that the selected data point represents. Clicking on a data point takes the user to screen showing a breakdown of the individual student scores by strand that make up selected class assessment.

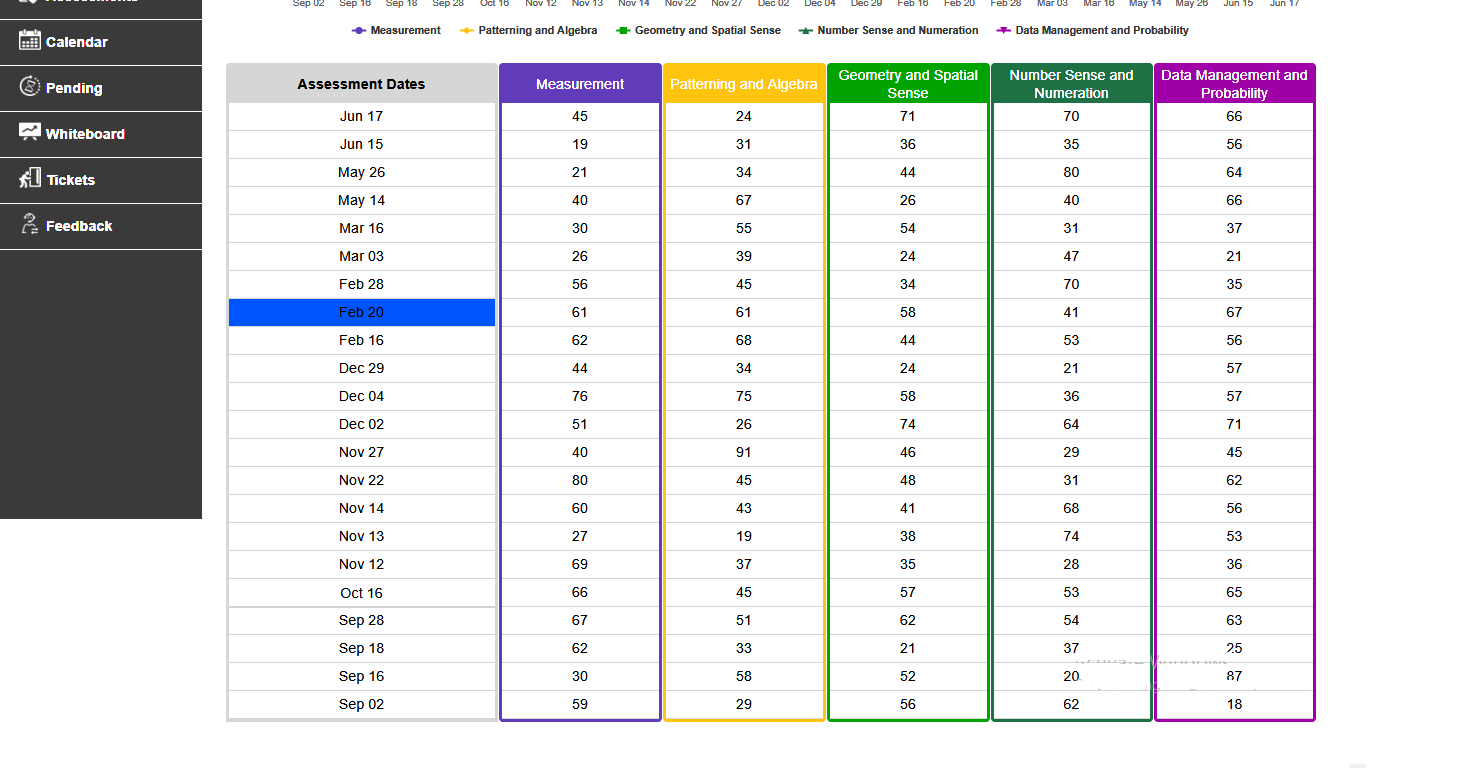
Using the demo as an example, clicking on the May 26th data point, drills down into that specific test and displays individual student scores for that test by strand. See the Section 4.2 “Student Strand Scores” for more details on this.

### Curriculum Chart

The Curriculum chart is simply a numerical representation of the data displayed on the chart above it.The example in Figure 17 shows all the five strands. Columns on the chart include the test date (the start date where an assessment is valid over a range of dates). And the included test should only be of the Formal and Practice type. Rows on the chart that represent Formal tests should have a background color to indicate that they are different than the practice tests.

Only the Formal Tests are highlighted with background color as HEX #0055FF and Practice Tests are not highlighted. Rolling the mouse pointer over a test date should highlight the date cell.

The strand columns in the chart should be color coded to match the strand colors. The data in the Strand columns is the overall average score for the selected class on that particular strand.Clicking the highlighted cell will take the user to another screen; the Strand Scores Detail screen.



**Figure 18: Curriculum Chart**

## Student Strand Scores

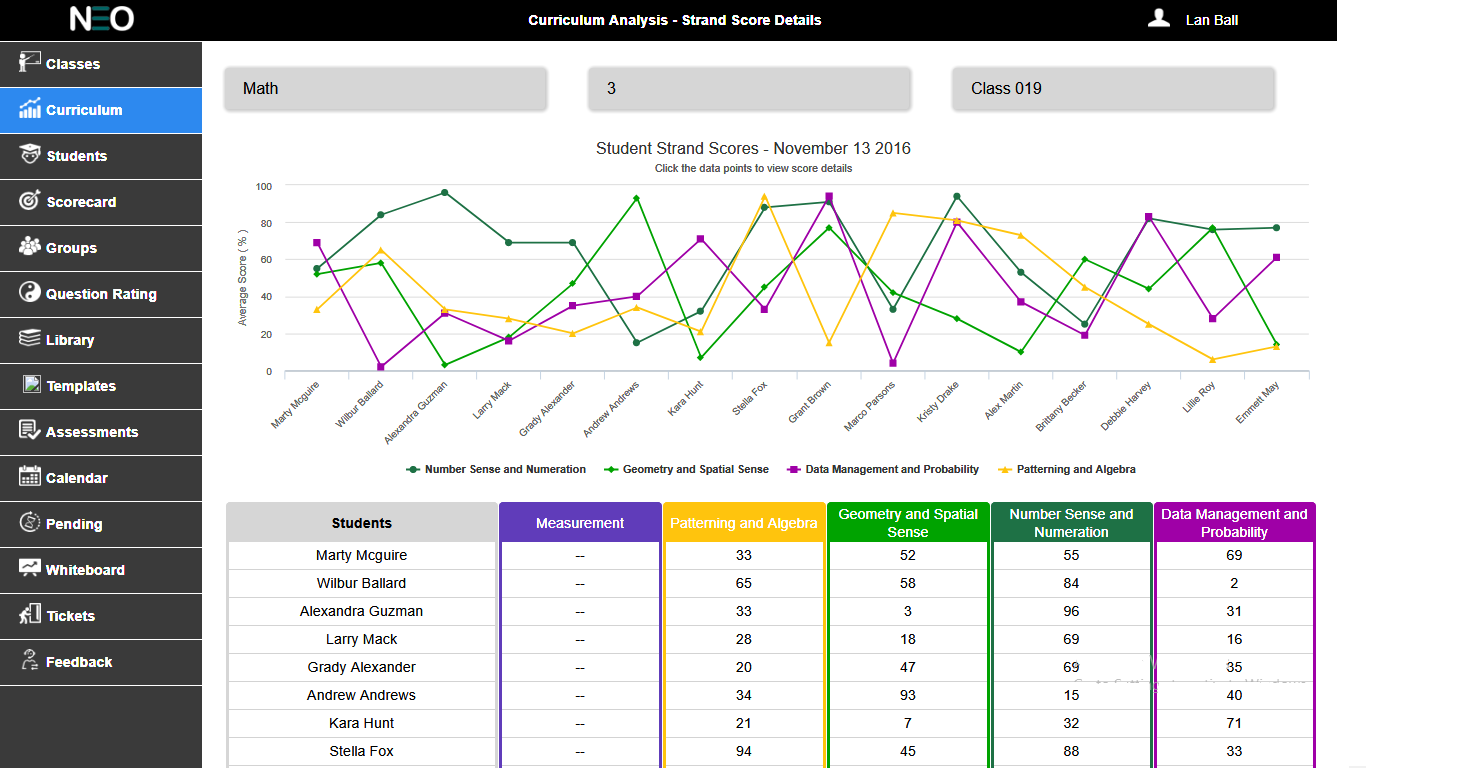
The Student Strand Scores screen displays a breakdown of the individual student scores by strand for the user selected assessment from the prior screen (Figure 19 below).

Across the top of this screen, the user can see the selected Course, Grade and Class. While these look like the filter drop lists for UI consistency, these are read-only. These selections can only be changed by returning to the prior screen and changing the filter there. The order of the displayed filters should be Course, Grade and Class.

Rolling over the filter selections should change the mouse pointer to indicate that the filters are read only.

The central graph should have a title of “Student Strand Scores – “, followed by the date of the selected test. The subtitle should read “Click the data points to view student details”. The y-axis title should read

“Average Score (%)”. There is no need for an x-axis title. The graph legend should be placed at the bottom of the graph and detail the strands and their associated colors. Each x-axis data points should be labeled with the student’s name

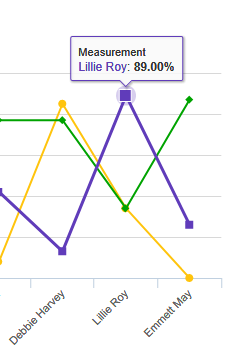


**Figure 19: Curriculum Strand Score Page**

Rolling over a data point on the line graph should reveal a tooltip with the following information: title

(Strand Name), the student’s first and last name, and the student’s strand score (Figure 20 below).

Line coloring should be the same as in all the prior screens for the various curriculum strands. Data points are user selectable and will take the user to another analysis screen; drilling down deeper intothe data to the Student Assessment Details screen.



**Figure 20: Tool Tip**

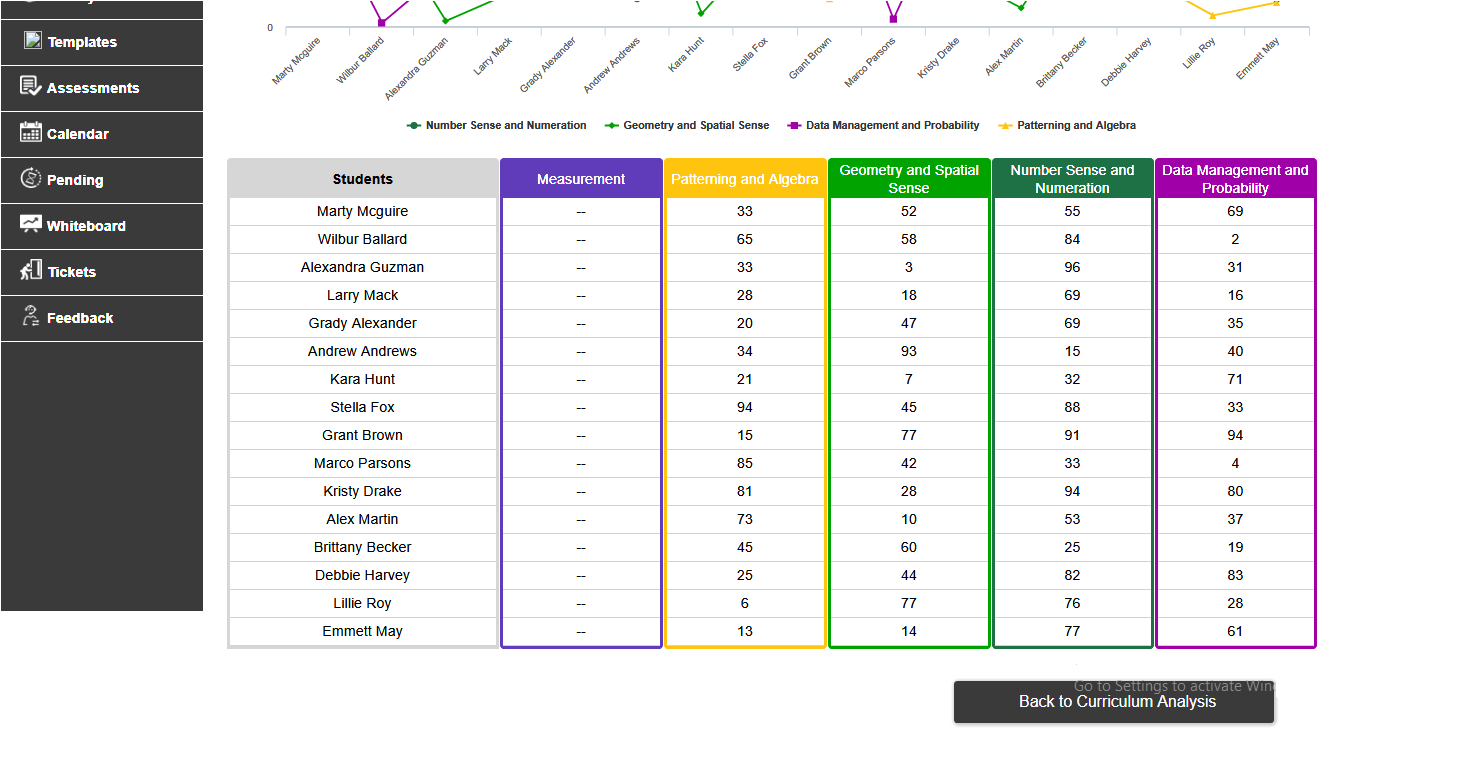
The chart below the graph shows a numerical representation of the data displayed on the chart above it.

The strand columns in the chart should be color coded the same as the lines on the graph. See the

Color Palette section for more details (see figure 21).

The data in the Strand columns is the score of a particular student in a particular strand on the selected date. Clicking the highlighted cell will take the user to another screen; the Student Assessment Detail screen. If the assessment does not contain any question for a particular strand, then that particular strand field should be shown with a hyphen (-).

The “Back to Curriculum Analysis” button at the bottom the screen simply returns the user to the prior screen (all user selections should still be in force).

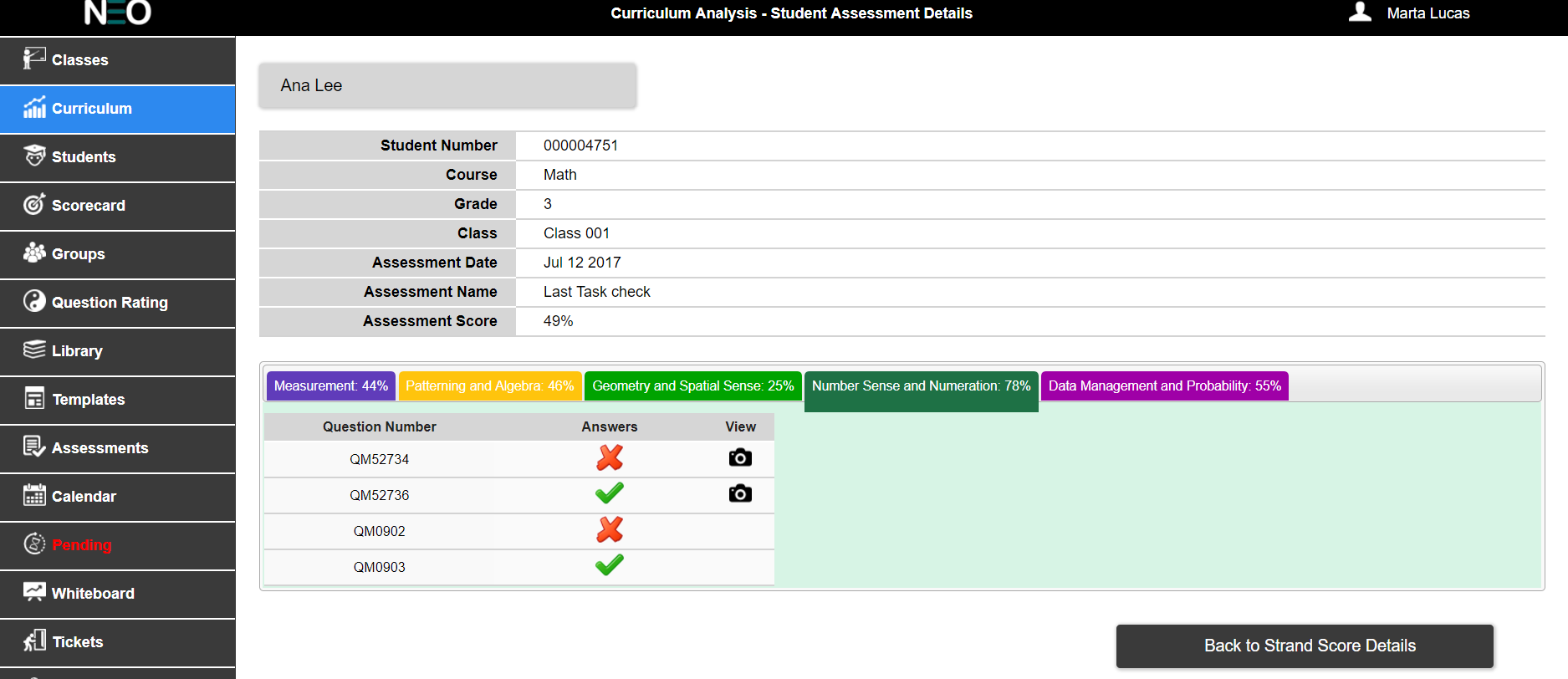


**Figure 21: Curriculum Chart with Student Scores**

## Student Assessment Details

The Student Assessment Details screen is the lowest level that the teacher can drill down to. It shows information about the specific student as selected from the prior screen, and for a specific test.

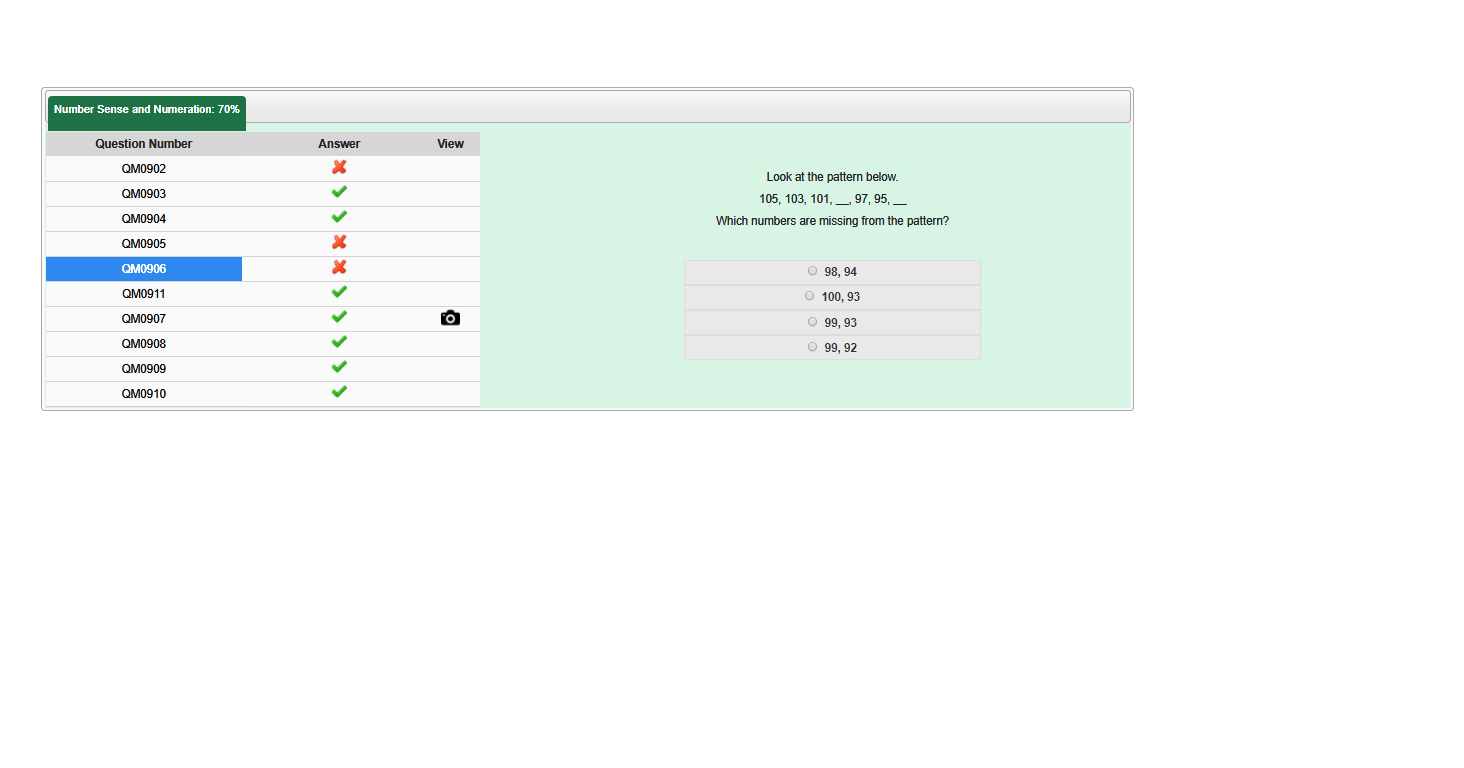
The top portion of this screen should display a combination of information about the selected student as well as the selected data filters that got the user to this screen. This information includes: the student’s first and last name, student ID, the selected course, grade and class (in that order), the selected assessment date, assessment name, and the student’s score for that selected test.



**Figure 22: Student Assessment Details Screen**

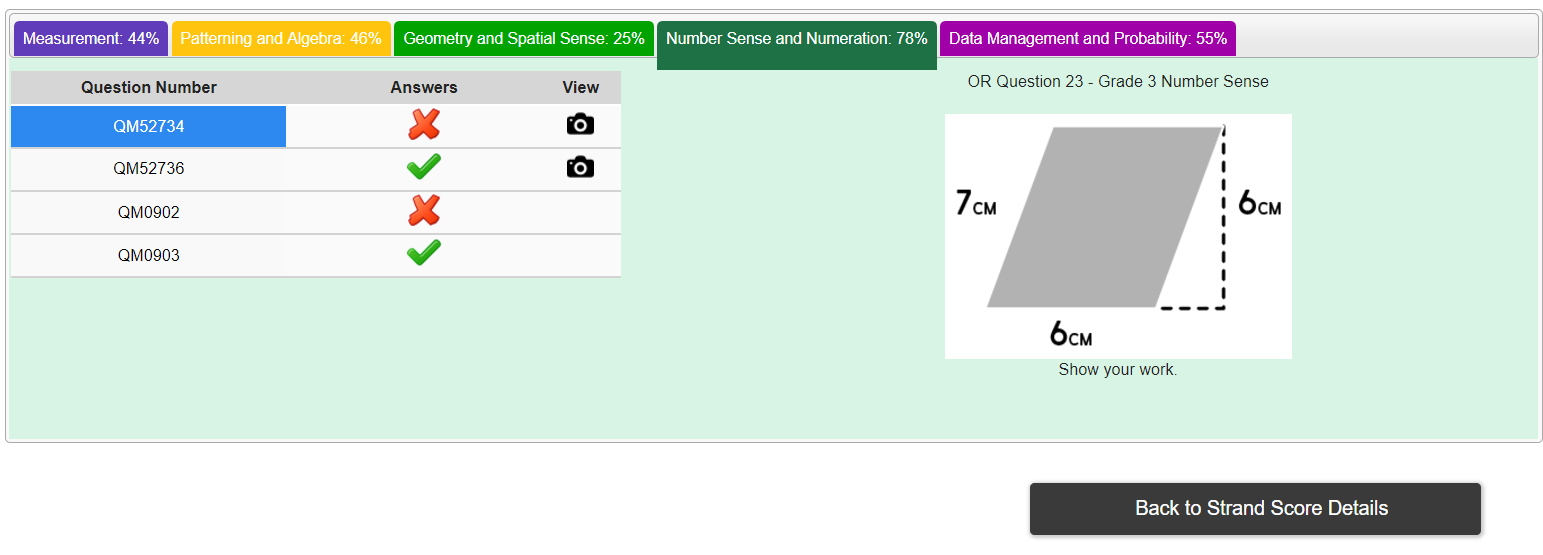
The lower portion of the screen should display color coded tabs for all the strands in the selected course. In turn, each tab will displayed all of the test questions that relate to the specific strands for that tab. For example, in grade 3 math, the Measurement tab refers to the Measurement strand and only displays the test questions related to Measurement that were on that specific assessment.

The student’s result for that question (right or wrong) is displayed along with each question ID number (Figure 23).



**Figure 23: Assessment Question Results**

Each question ID is user selectable. Rolling over the question ID with the mouse pointer will highlight it. Clicking on the highlighted question will reveal a preview of that question beside the chart (see Figure 24 below). On click of the camera icon, the user can see the answer written by the student for an open response or text input questions.

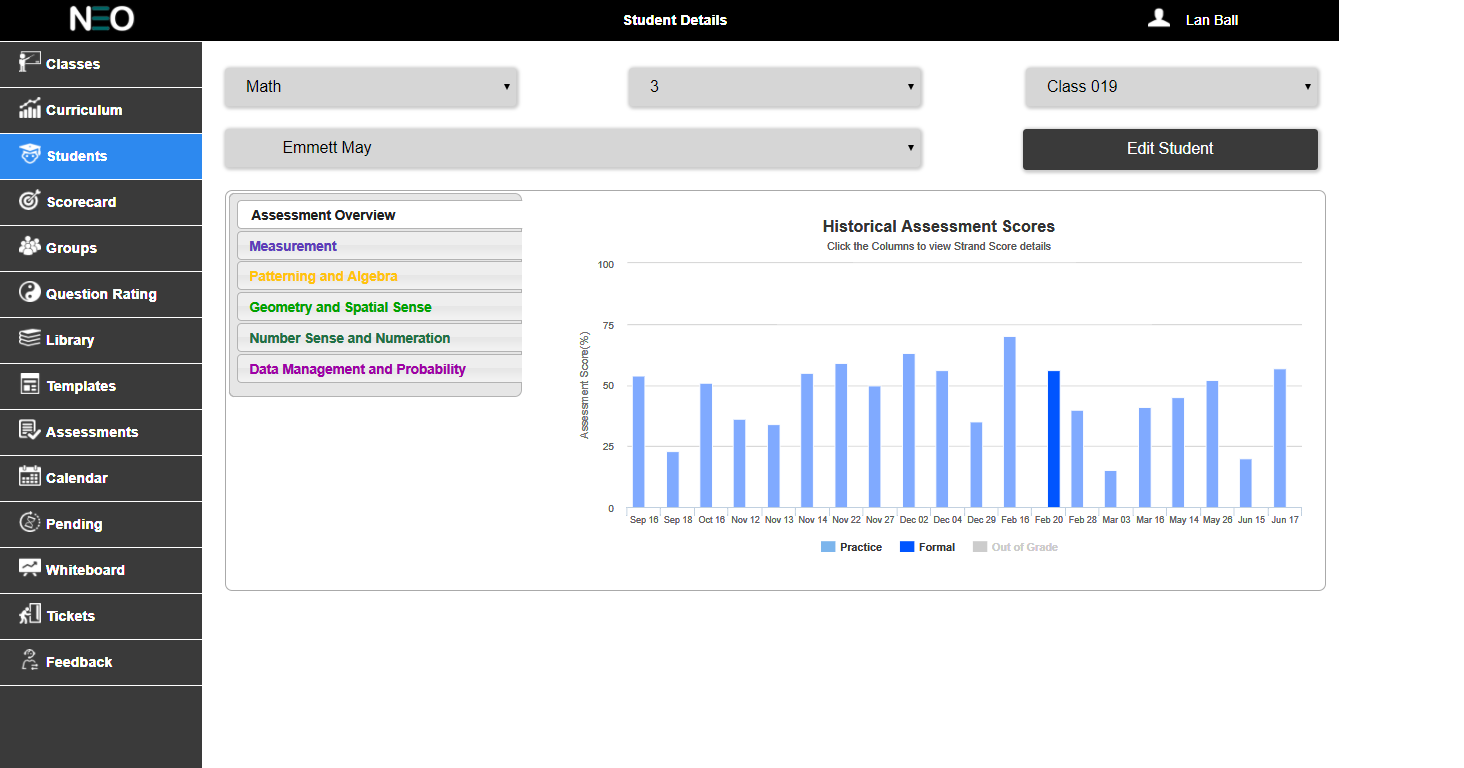


**Figure 24: Question Preview**

The “Back to Strand Score Details” button will return the user to the prior screen with all their previous filters and selections still in force.

# Student Tab

The Students tab (Figure 25 below) is used by the teacher role to view assessment results from the perspective of the individual student and their test history (broken down by Strand).



**Figure 25: Student Details Tab**

Common UI elements on this screen include the Course, Grade and Class drop lists to filter/select the data. Additionally, there is a drop list to be populated by all the student’s names from the selected class.

The main UI element on this screen is a tab group that allows the teacher to switch between an overview of the selected student’s test achievements and their question results by curriculum strand.

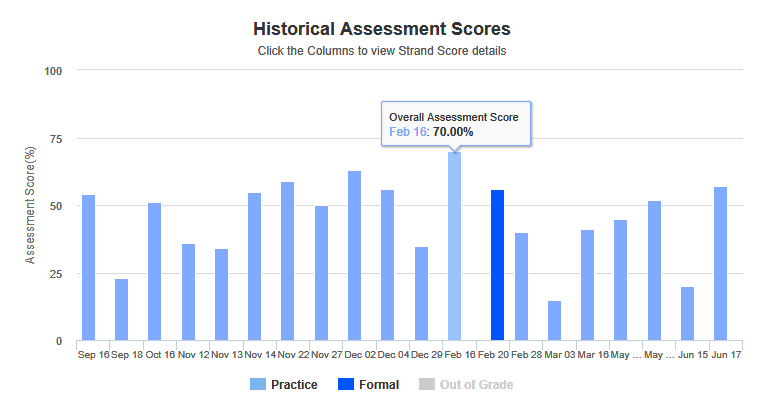
Initially, this screen defaults to the “Assessment Overview” tab with a bar graph of the selected student’s test scores over time; see the following sections of this for more details.

### Assessment Overview

The Assessment Overview tab allows the teacher to see selected students historical test scores. The date range for this bar graph is constrained by the start and end dates of the selected class. This graph will show test scores for Formal and Practice test types and Out Of Grade, and the bars should be colored coded for these test types as they were for the Classes tab (section 3.3.2 of this document).

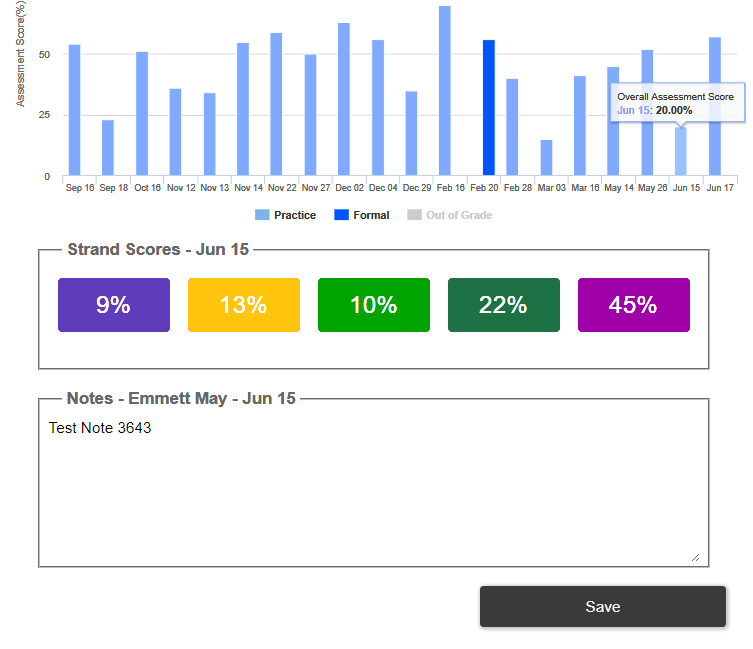
The bar graph should have the title “Historical Assessment Scores”, and a subtitle “Click the columns to view strand score details”. The y-axis should have the title “Assessment Score (%)” and a range of 0 to100 with interval line every 25 points. The x-axis will have no title and the data points should be in short date format.

Rolling over a bar on the graph will highlight that bar and display a tool tip. The tool tip will have the title “Overall Assessment Score” followed by the date and percentage score (see Figure 26).



**Figure 26: Historical Assessment Score Graph**

Clicking on a bar will display a color coded breakdown of the strand scores for the selected test, as well as a space for the teacher to make notes (Figure 27). Notes should be stored in the Result table of the database in a field of the same name. Notes are the only element on this screen that is editable; everything else is read-only.Notes are saved to the database table on clicking of the Save button.

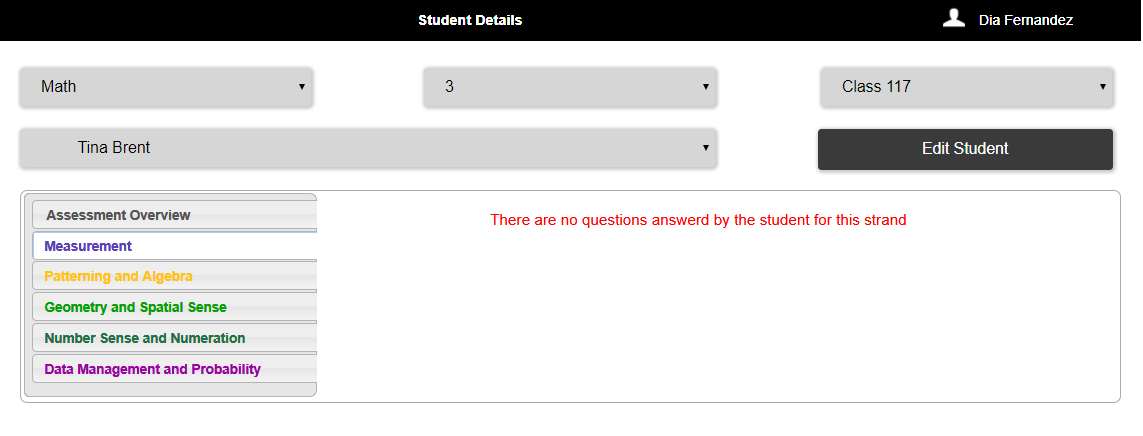


**Figure 27: Strand Scores & Notes Revealed**

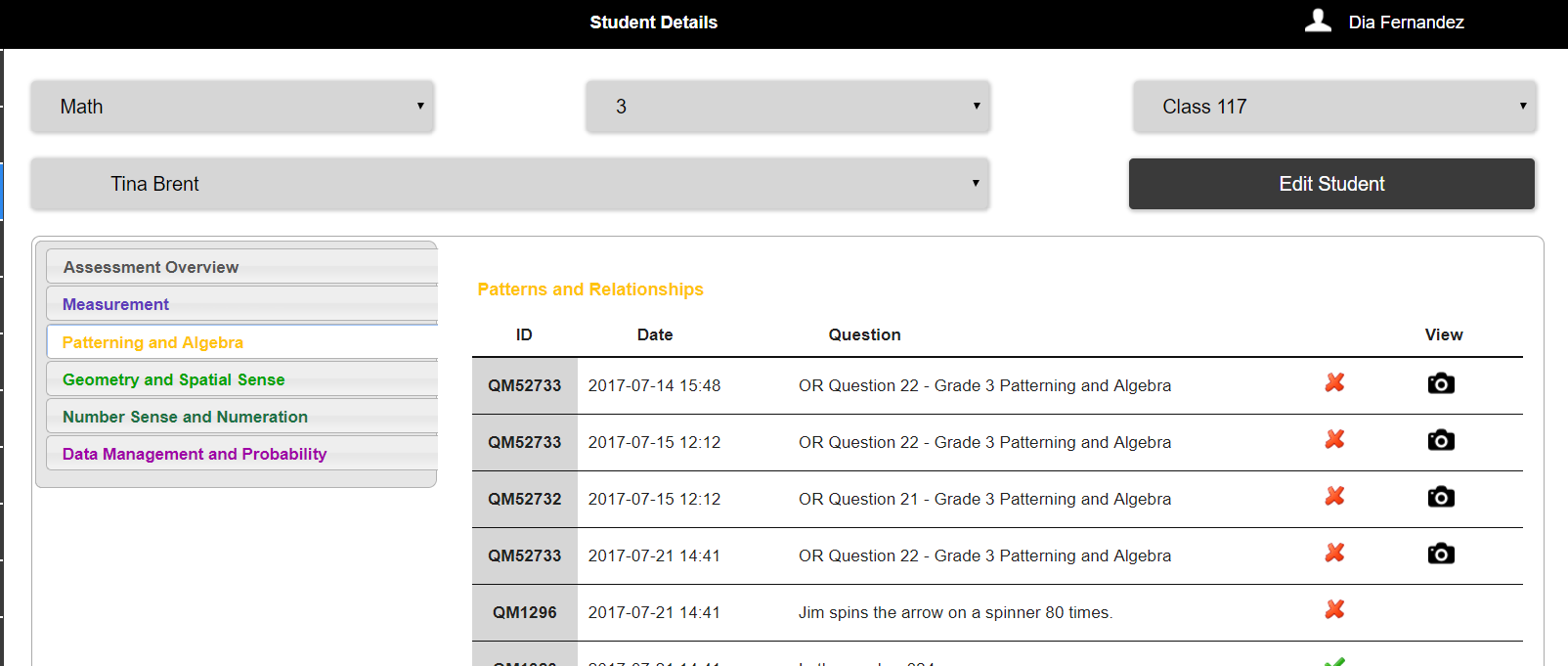
### Strand Details

The strand details tabs should display a list of every question that the selected student has answered broken down by specific strand and then organized by General Expectation. Each row of detail should have the following columns: Question ID, Date and time that the question was answered by the selected student, the first line of the question, an indicator of the student’s response (right or wrong)and View column to see the responses of the student for input type/open response questions.

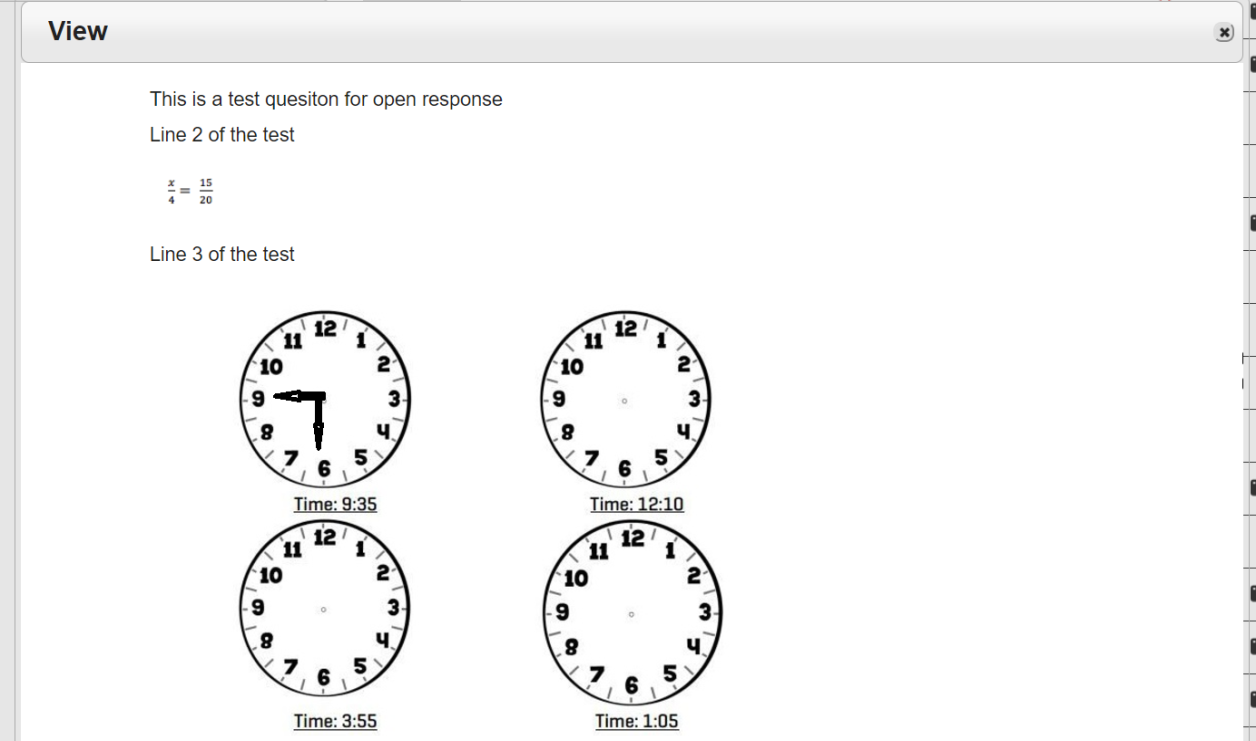
In case there are no questions that are answered by the selected student, then there should be a message saying “There are no questions that are answered by the student for this strand” (Figure 28).The color of General Expectation of different strands should be same as the color as the strand color (Figure 29, Figure 30, Figure 31, Figure 32and Figure 33).



**Figure 28: No questions answered**

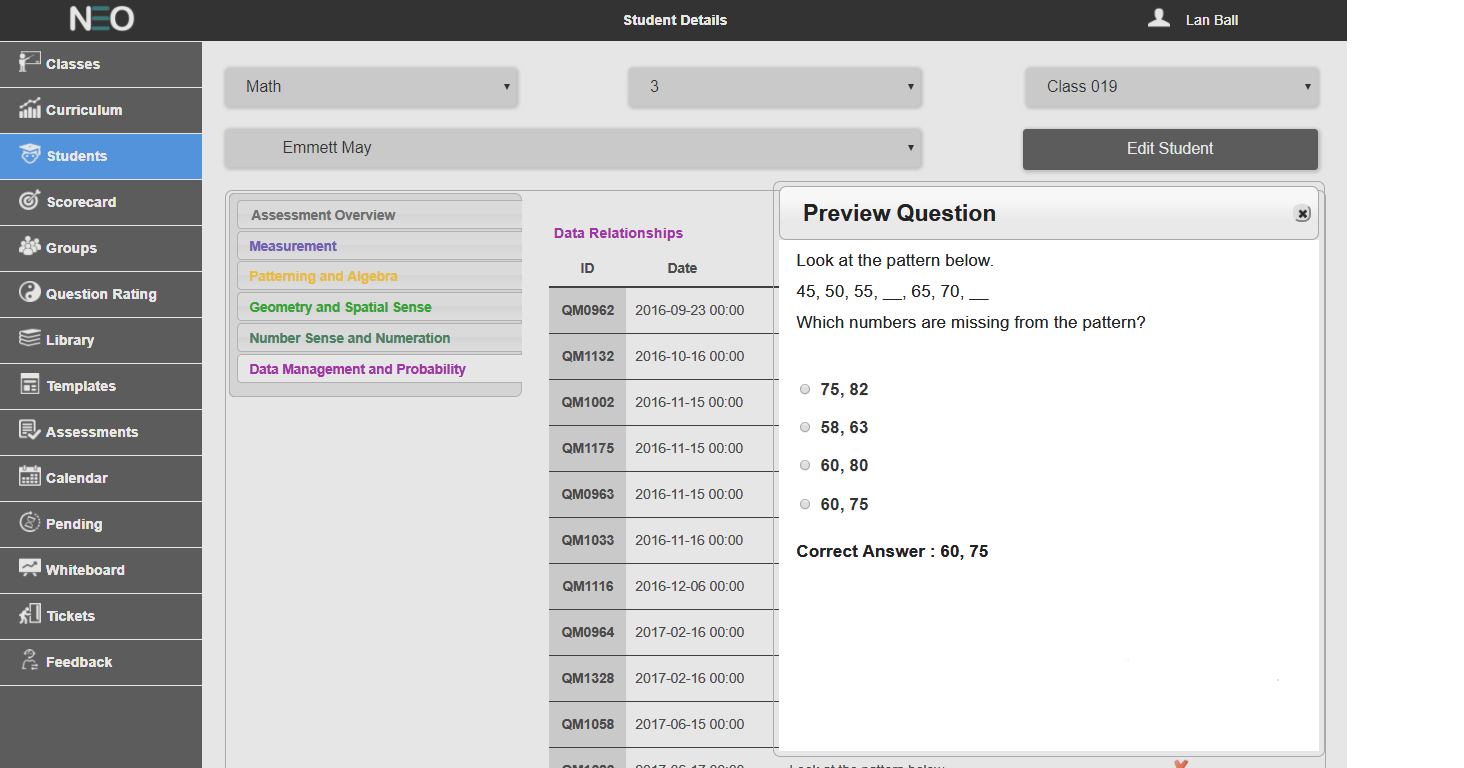


**Figure 29: Questions listed under General Expectation**

****

**Figure 33: Camera Icon Preview**

Rolling the mouse of the Question ID should highlight the field (use HEX #0055FF). Clicking on the Question ID should produce a preview of the entire question, as seen in Figure 34 below.

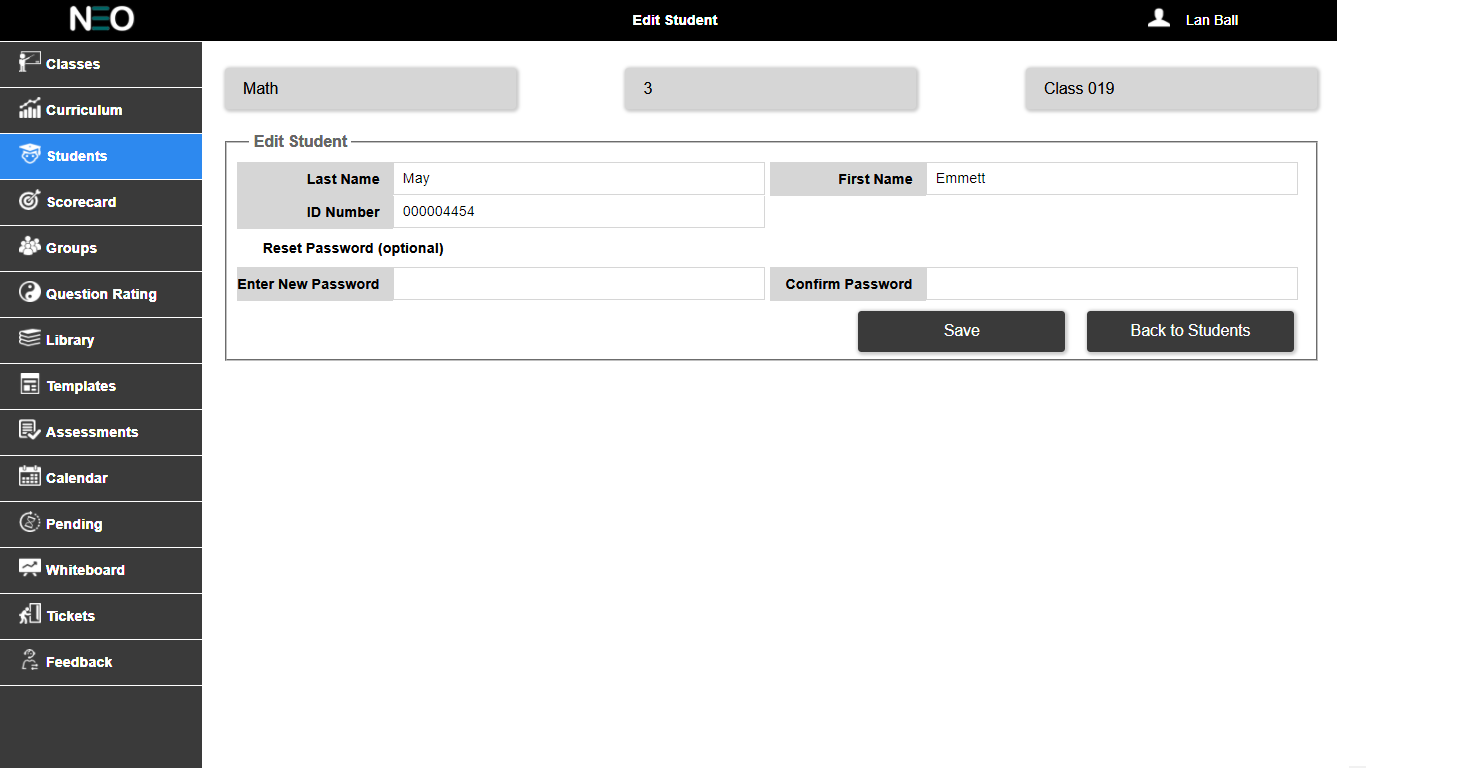


**Figure 34: Question Preview Popup**

## Edit Student

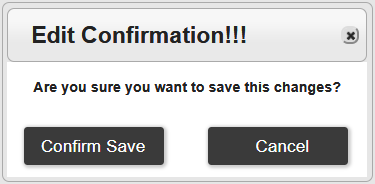
The Edit Student page is used by the teacher to edit the student details like First name, Las t name, ID number, etc. After making the required changes and clicking the save button, the changes will be updated to the database (Figure 35).

There are more details to this section, refer below issue numbers in closed issues of Github. Take Ramya help to get it and check the app too. Student: Edit Student- Reset Password #177, Edit Student Details - #165



**Figure 35: Edit Student page**

The user can also change the password for the student, which is optional. In case the user changes the password for a particular student and clicking on the save button, a pop-up message comes up saying whether the user want to save the changed details (see Figure 36).

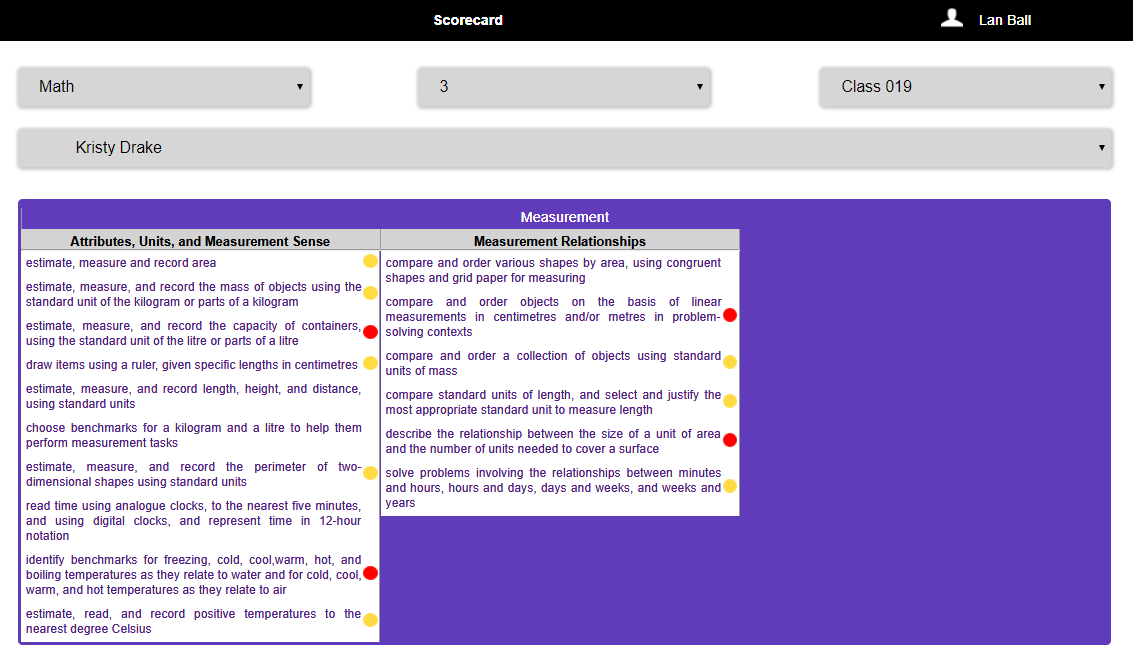
****

**Figure 36: Confirm Save Changes**

On clicking of the “Back to Students” button, the user will be taken back to the prior page i.e. students tab.

# Scorecard

The Scorecard tab provides the teacher with an overview of how the selected student is doing as measured against the course curriculum (see Figure 37).



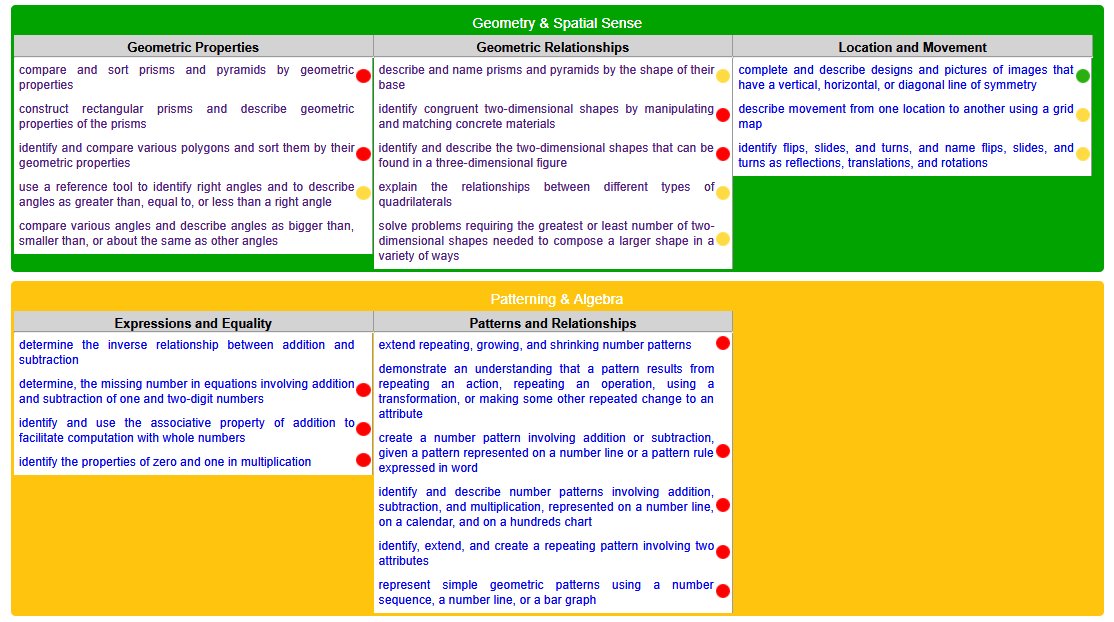
**Figure 37: Scorecard Tab**

The Scorecard screen uses the same common UI elements as the Students Tab; Course, Grade, Class and

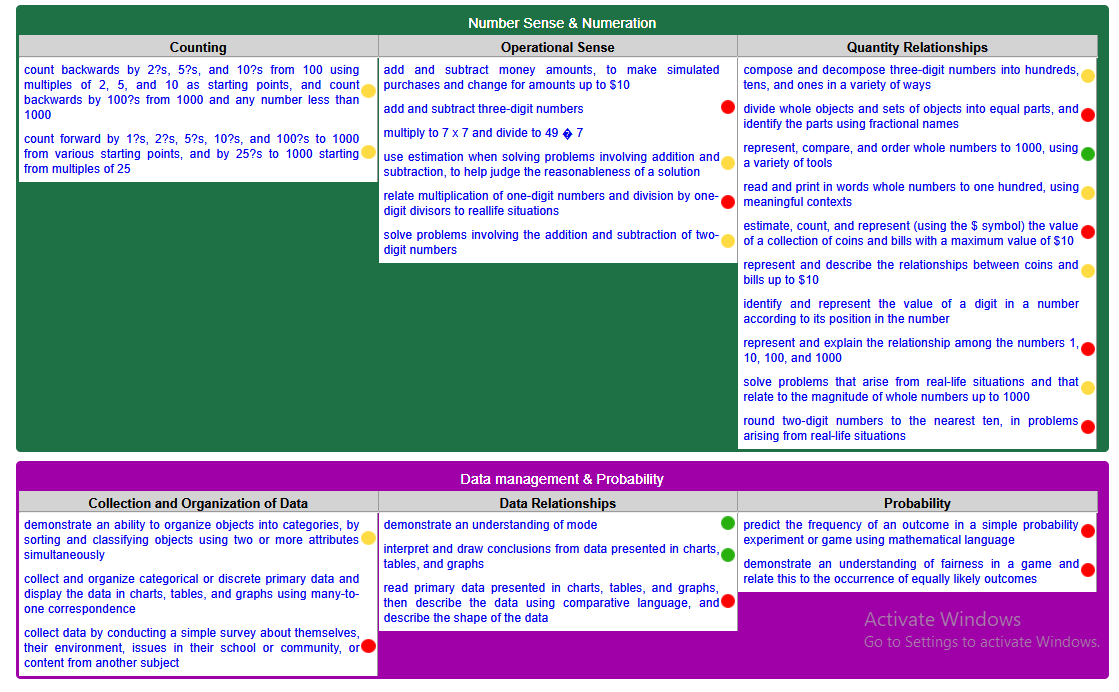
Student drop lists.

The curriculum strands should be colored coded (just as on other screens using the Color Palette as a guide).

To accommodate the extra space required for displaying the strands, the colored strands should be horizontally. Each strand should be organized by General Expectation followed by a list of all the associated Specific Expectations. The different General Expectations with its Specific Expectations should be displayed in different column for every strand (Figure 38 and Figure 39). Each Specific Expectation is accompanied by a colored dot. The color of the dot is determined by the student’s average score on questions related to that Specific Expectation.



**Figure 38: Scorecard Tab - Geometry & Spatial Sense and Patterning & Number Sense**

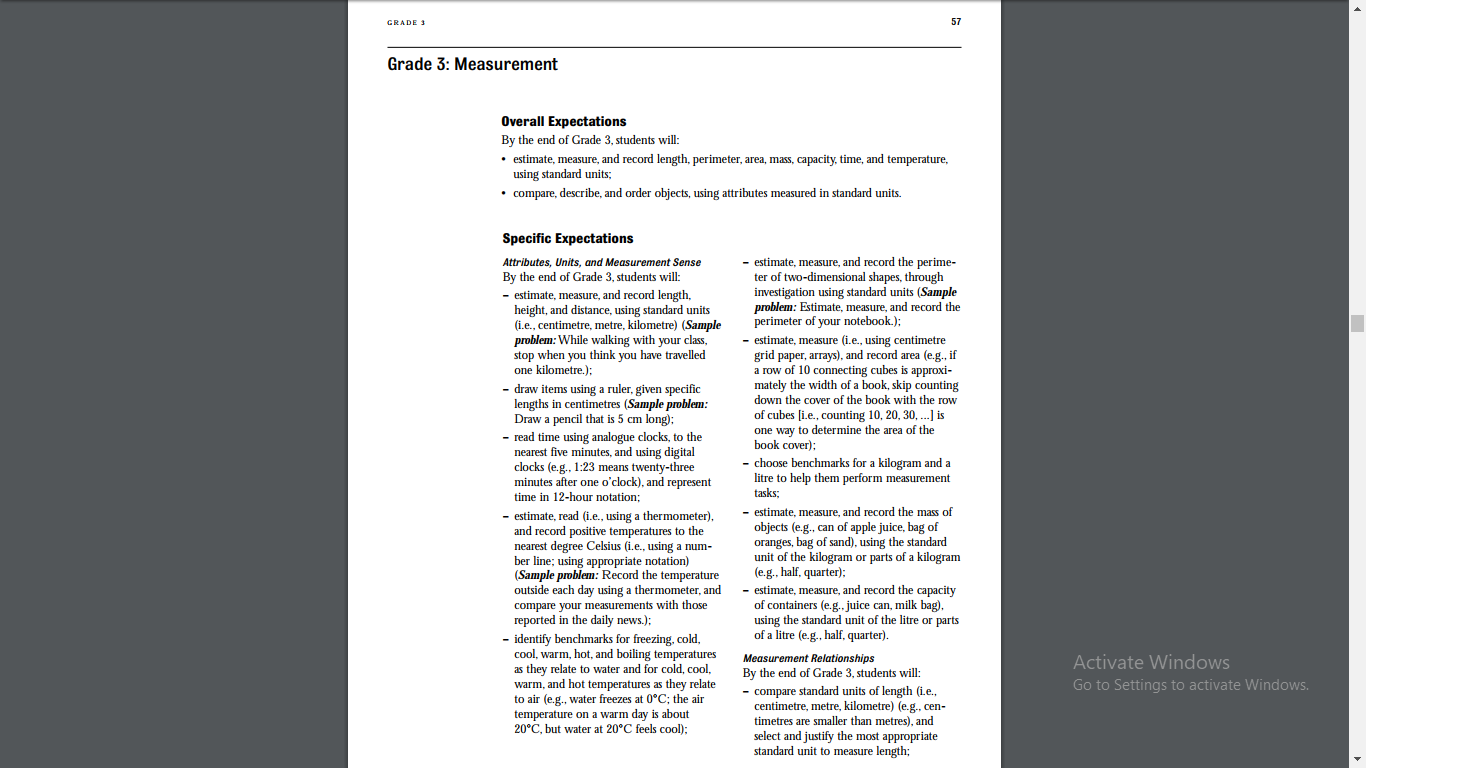
****

**Figure 39: Scorecard Tab – Number Sense & Numeration and Data Management & Probability**

The logic for determining the dot color is:

|  |  |  |
| --- | --- | --- |
| Dot Color | Average for Specific Expectation | Notes |
| White or no dot | Nothing to calculate | If the student has NOT answered any questions on this Specific Expectation, they cannot be assigned a colored dot. |
| Green | >75% | Students must answer 2 or more questions with an average of greater than 75%. Students that 2 or less questions, but still average greater than 75%, should be assigned a yellow dot. |
| Yellow | >50% to <75% | Students that average between 50% and 75% will be assigned a yellow dot. Also, students that average greater than 75% but have answered 2 or fewer questions should be assigned a yellow dot. |
| Red | 0 to <50% | Students that average less than 50% should be assigned a red dot. |

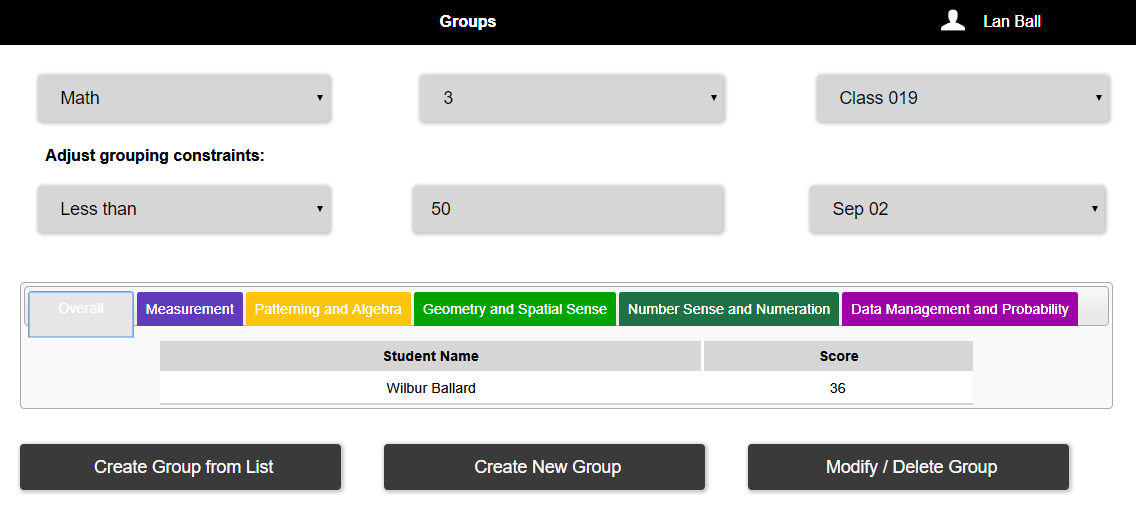
On click of any Specific Expectation, the user will be redirected to the Specific Expectation in the curriculum document of the class and grade which were selected by the user (see Figure 40).



**Figure 40: Scorecard - Curriculum**

# Groups

The Groups tab (Figure 41 below) is used by the teacher role to create groups of students, within specific classes, organized by overall test scores and by curriculum strand results.



**Figure 41: Groups Tab**

The Overall score is calculated by simply averaging all of the individual student’s test scores (within the selected class); including all Formal and Practice test types but excluding the Exit Ticket type. The strand scores for each student are calculated the same way.

On this tab, the common UI elements used include the Course, Grade and Class drop lists to filter/select the data.

UI Elements that are unique to this screen include a second row of filters under the heading of “Adjust grouping constraints”. The lower portion of this screen displays tabs for all of the strands in the selected course as well as a tab for overall test scores. Each of the tabs will show a list of students that complies with the user selected filters and constraints from above.

### Adjust Grouping Constrains

There will be three more user configurable constraints on this tab under the heading “Adjust grouping constraints:” See Figure 42 below.



**Figure 42: Grouping Constraints**

The first drop list gives the user a choice of “Less than” or “Greater than” the mark specified in the second constraint. For filter logic, the “Greater than” option is equivalent to “greater than or equal to”.

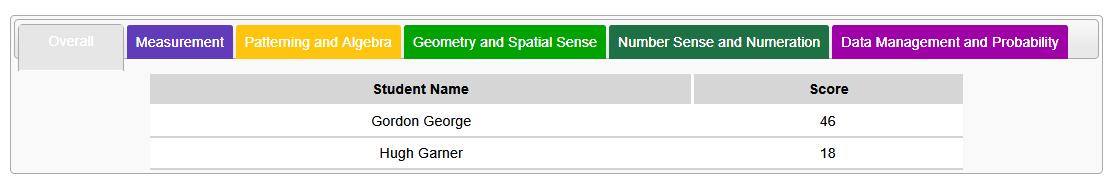
The second constraint is test score that the teacher if going to filter on. This constraint should be a spinner UI element and default to 50 percent.

The third constraint is a list of all of the Formal and Practice test dates for a selected class. This list should be populated by a database query using the logged in teacher ID, selected course, grade and class as filters.

### Grouping Results

The lists of students generated by the above filters and constraints should be organized by curriculum strands and overall test scores (color coded as in other areas of the software for consistency). The

HTML field set backgrounds should also be lightly colored to reflect the tab the user is currently viewing.

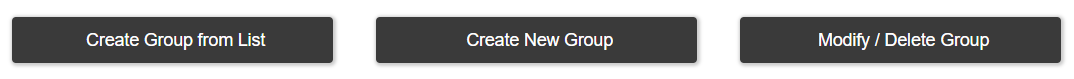


**Figure 43: Grouping Results**

### Buttons

The buttons along the bottom of the Groups tab allow the user to:

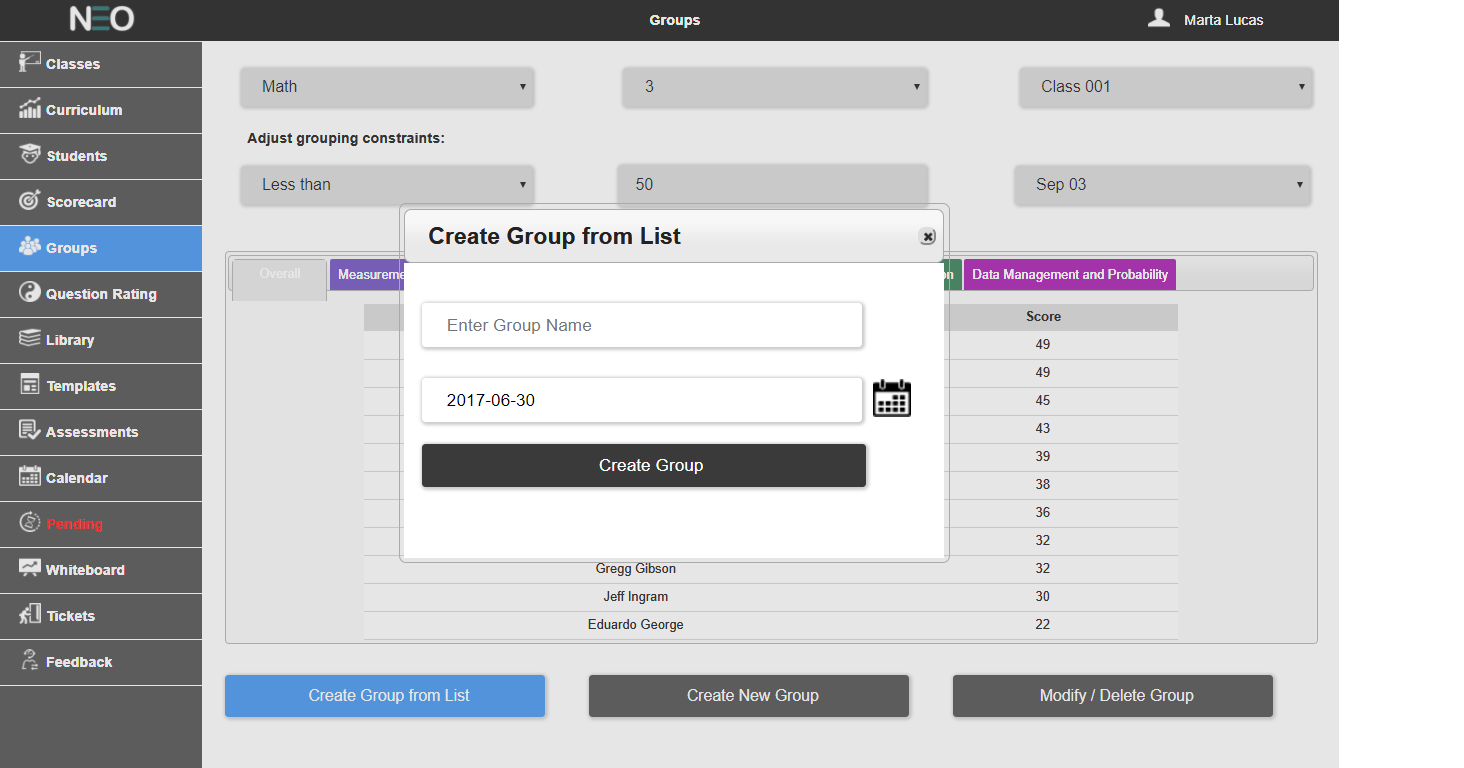
* Create groups, within the selected class, from the selected filters and constraints.
* Create groups from scratch, within the selected class, from a full list of students in specific class.
* Modify or delete existing groups but only if no test results exist for that specific group in the database Test table.



**Figure 44: Groups Tab Buttons**

## Create Group From List

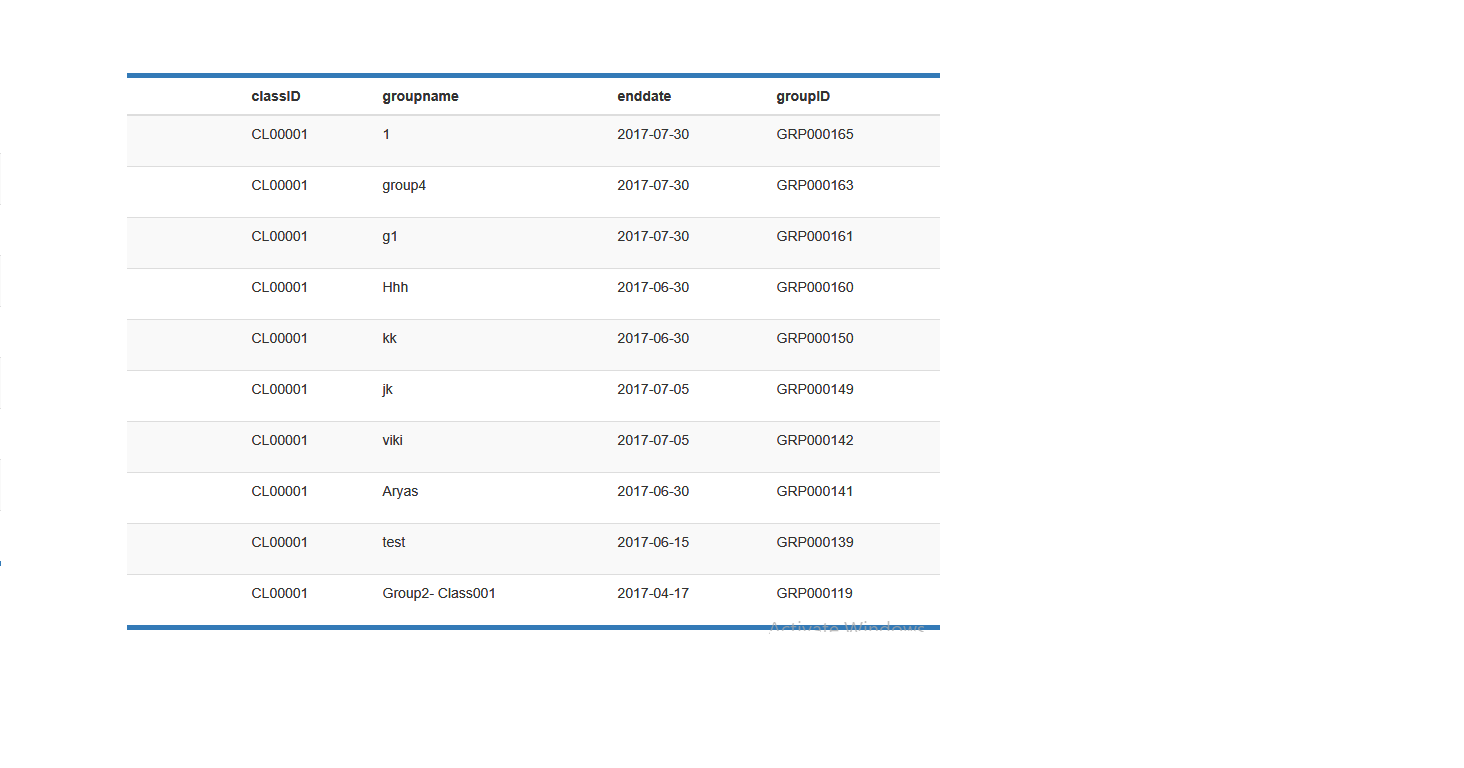
Selecting the “Create Group from List” button will give the user the option to name the group before saving it to the database (see Figure 45 below). The user should to select the End Date for this new group. The End Date being the last day that this group is considered to be active. This End Date should default to the last day of the current school year. On clicking of the “Create Group”, the entered group name by the user with the date should be saved to the Group table.



**Figure 45: Naming a Group**

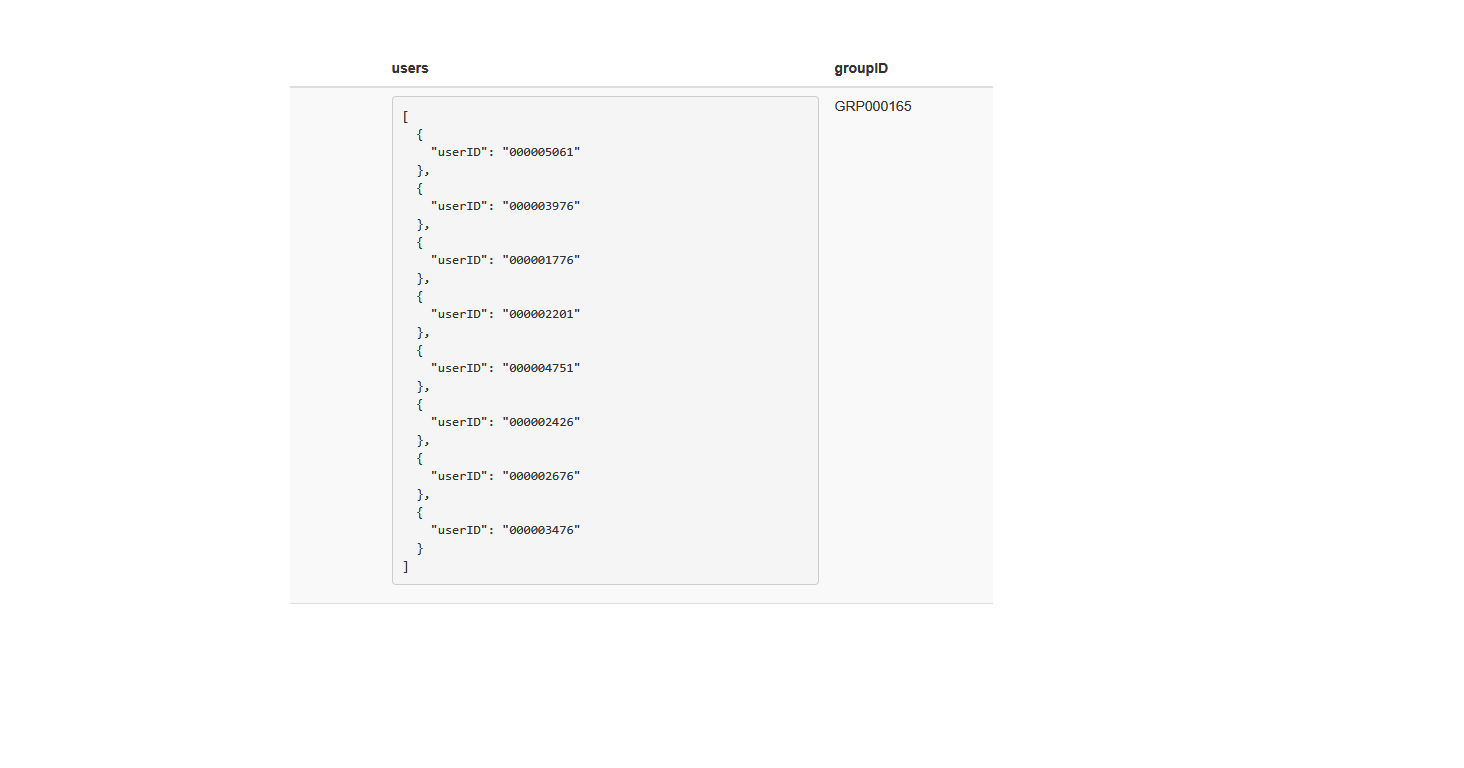
Data about this new group should be stored in the database Group and GroupUser tables. This group data should consist of a Group ID (created by the software like the other system IDs are created), Class

ID (user selected class), End Date, and Group Name; stored within the Group table (see Figure 46 below).



**Figure 46: Sample Group Table**

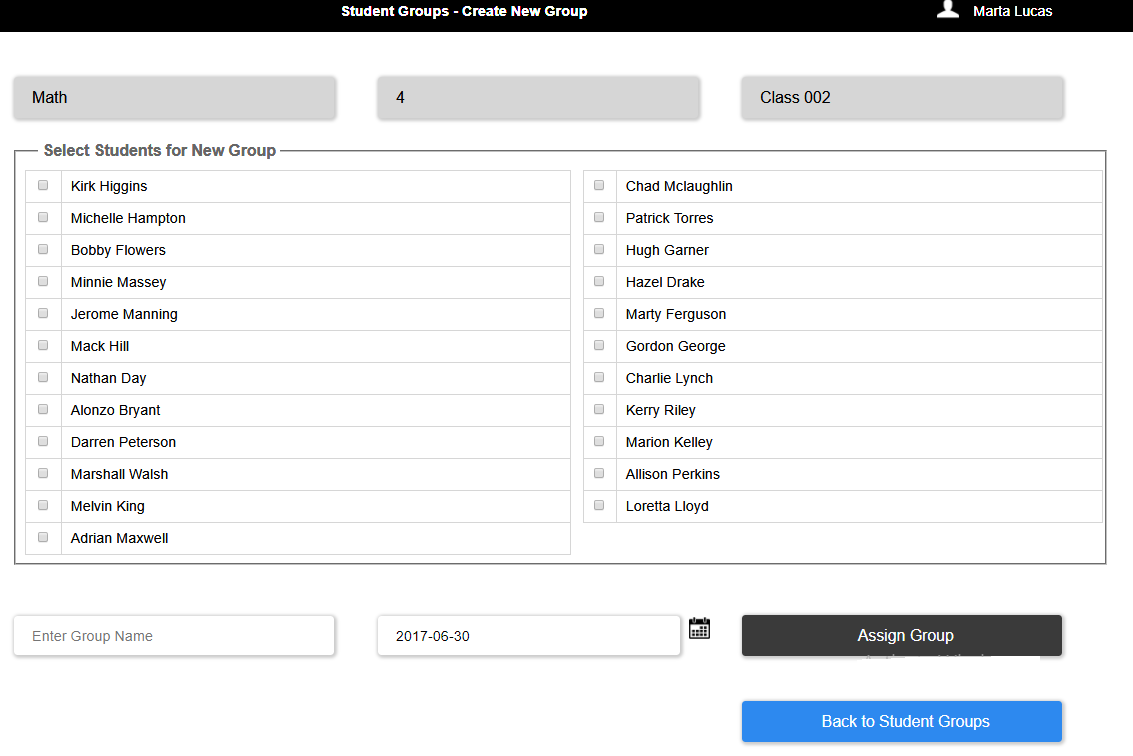
The GroupUser table will append a list of all the filtered students from the currently displayed tab along with the new Group ID (see Figure 47 below).



**Figure 47: Sample GroupUser Table**

## Create New Group

Clicking on the “Create New Group” button will move the user to another screen (See Figure 48 below). At the top of the Create New Group screen, there should be the common read only UI element that shows which filters are currently in force. Below that, will be a grid showing the first and last name of every student (in alphabetical order) with a check box, the user can use to select the students for the new group.

****

**Figure 48: Create New Group Screen**

Below the grid there should be an input box for the user to name the group, and there is also be a date input box for the user to enter an end date for this group. Here also the default should be the last day of the current school year.

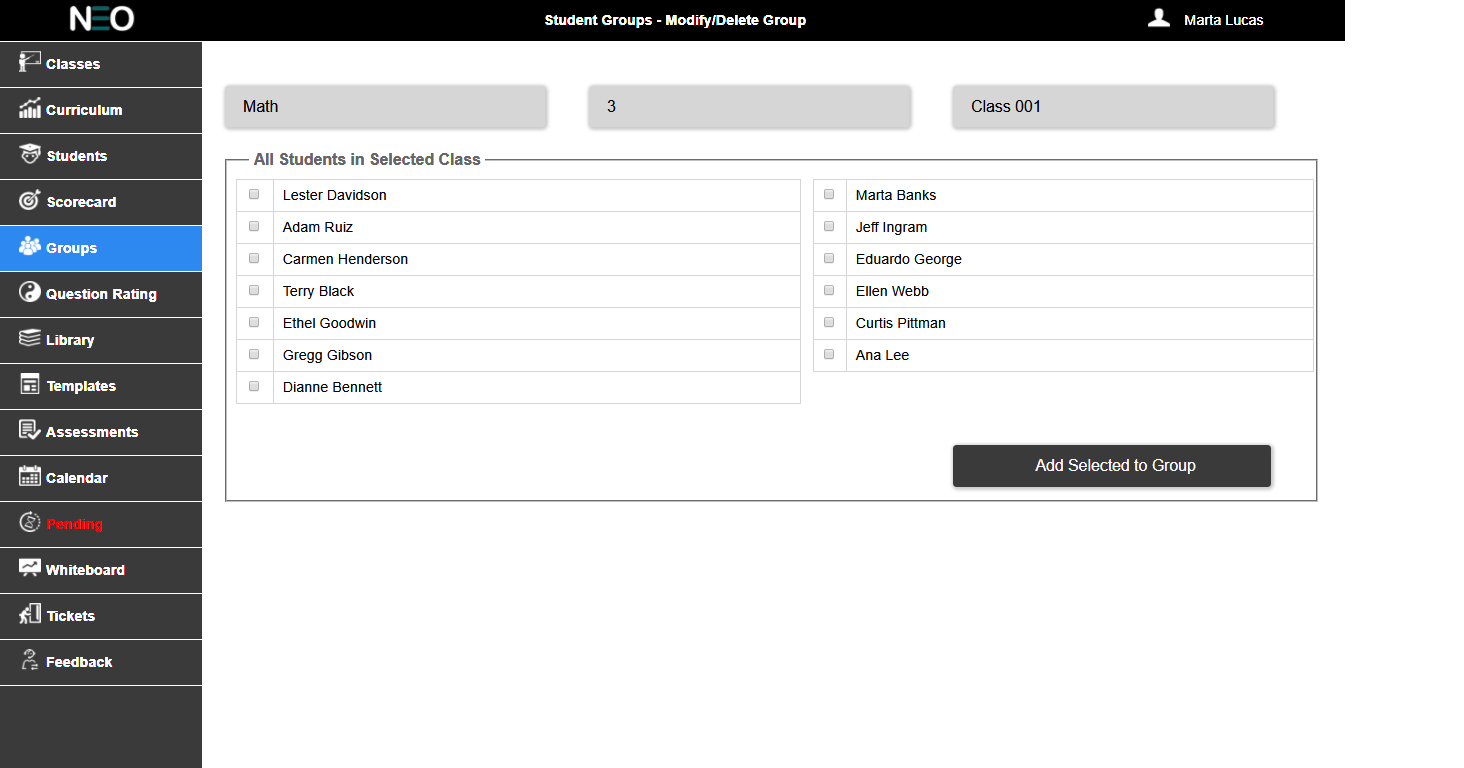
There should also be two buttons: one to create the group (Assign Group), and another to take the user back to the prior screen (Back to Student Groups); these button are essentially save and cancel buttons respectively. Data for new groups should be stored in the Group and GroupUser tables.

## Modify And Delete Group

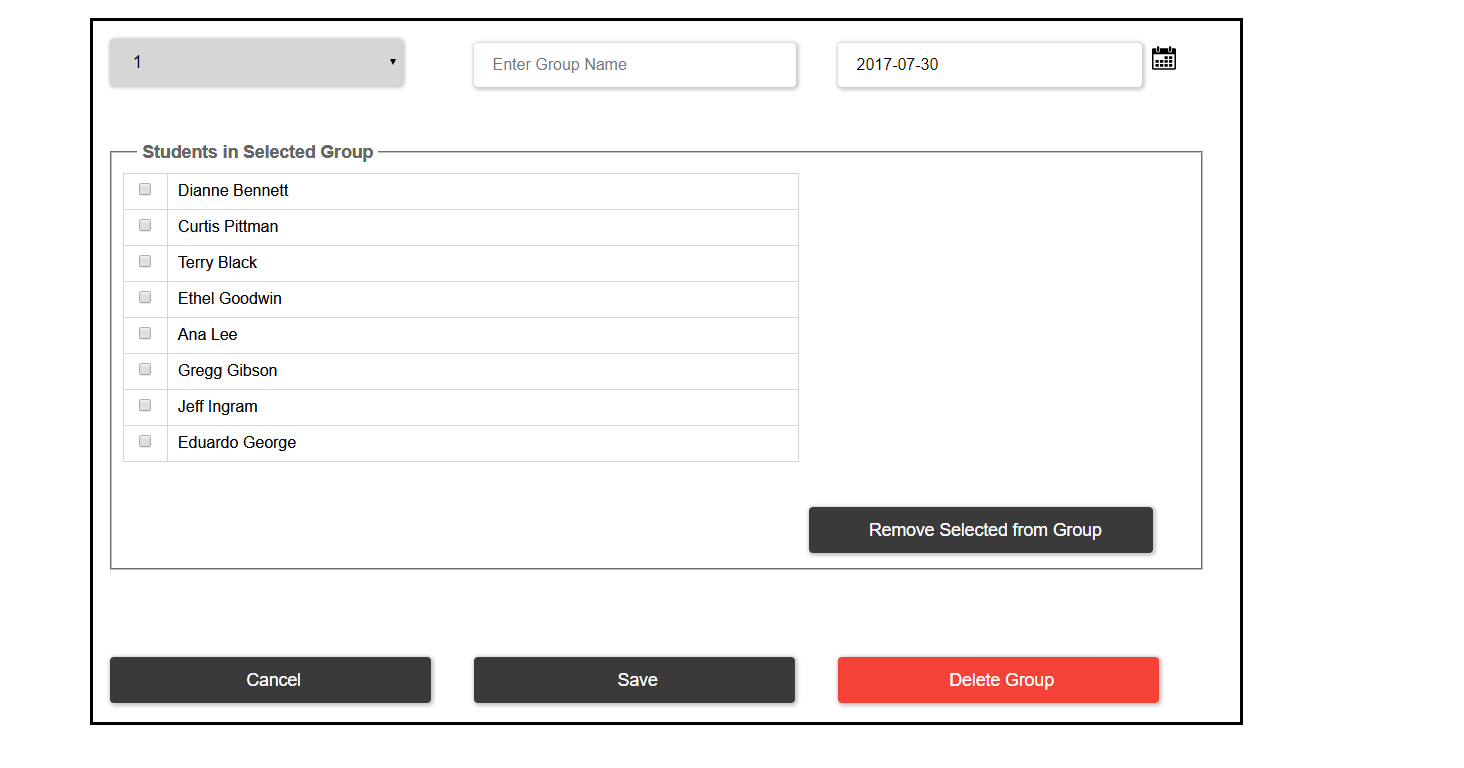
Clicking on the “Modify/Delete Group” button will move the user to another screen (See Figure 49 and Figure 50).

***Please note:*** while editing the name of a group should always be allowed, deleting or modifying the user list should only be allowed if the selected group has no history in the database Test table. Deleting or modifying a group that has history could potentially create orphaned data in the database. **Errorchecking for this condition will need to be implemented on this screen.** For example, if a user tries to delete a group with testing history, an error message should be displayed that this action is NOT allowed.

At the top of this screen, there should be the common read-only UI element that shows which filters are currently in force. Below that, will be a grid showing the first and last name of every student in the selected class (in alphabetical order), with a check box that the user can use to select students. At the bottom of this grid is a button that, when clicked, will add the selected students (checked in the upper grid) to the list of students in the grid below it (which are the students in the current group displayed in the drop list). Below the class grid, will be a drop list for all the groups that currently exist for the selected Course, Grade and Class and logged in User ID. Beside the dropdown, a text field should be there to enter a new group name and a date picker to select an End Date for the group. The default date provided should be the last day of the current school year. Selecting a different group from the drop lists should update the lower grid (the group grid) with the associated students in that selected group. These name should also be in alphabetical order and selectable via a checkbox. Within the group grid is a button that will allow the user to remove names from the selected group.



**Figure 49: Modify Group Screen**

****

**Figure 50: Delete Group Screen**

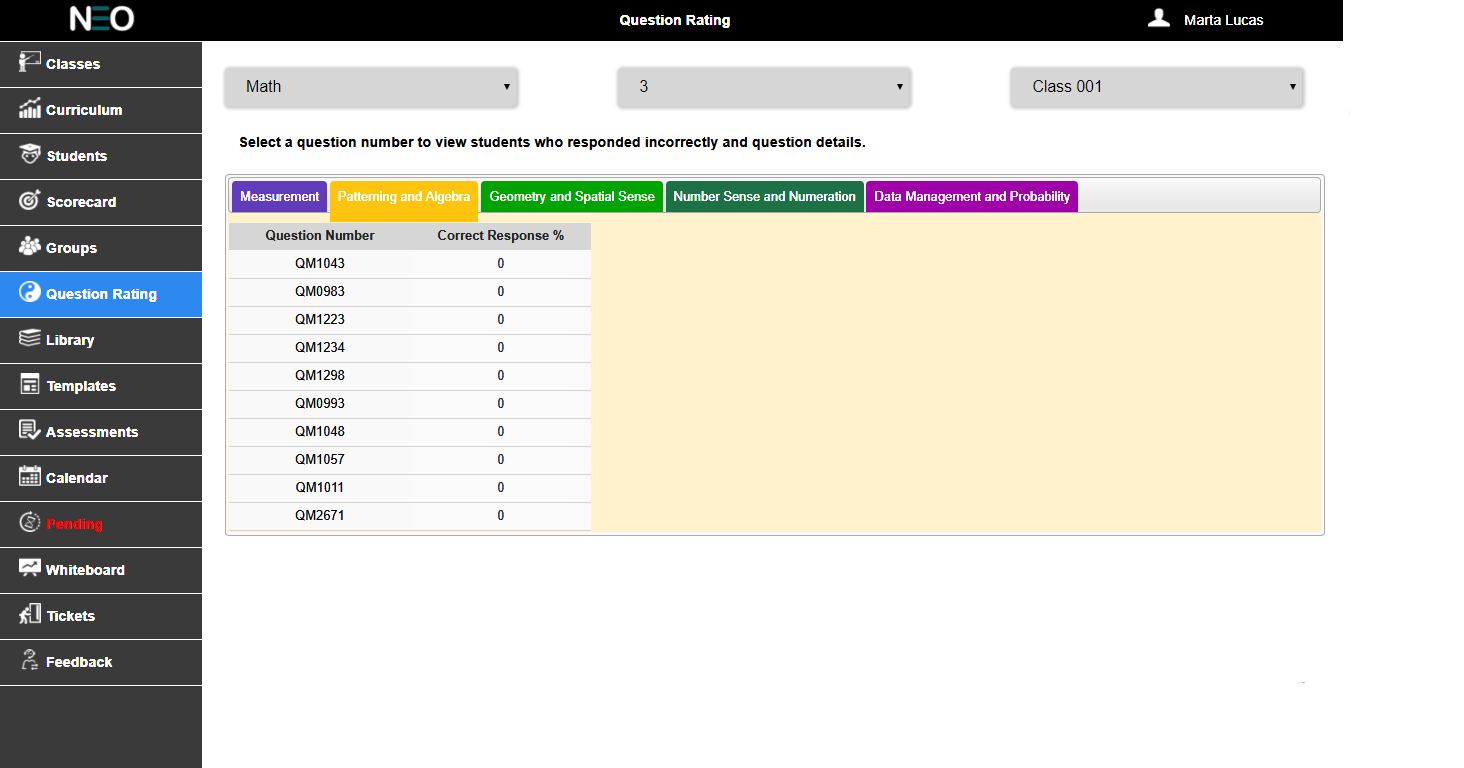
At the bottom of this screen there should be an input box where the user can modify the group name; the current group name should display by default.

There should also be three buttons:

* “Save” button to save all changes back to the database and return the user to the prior screen.
* “Cancel” button that will return the user to the prior screen without saving their changes.
* “Delete Group” button to delete the entire group from the database. Records related to this group should be deleted from both the Group and GroupUser tables. This button should be colored red (Hex #E60000).

# Question Rating

The Question Ratings tab allows the teacher to see the top 10 most difficult questions for the selected class (Figure 51).



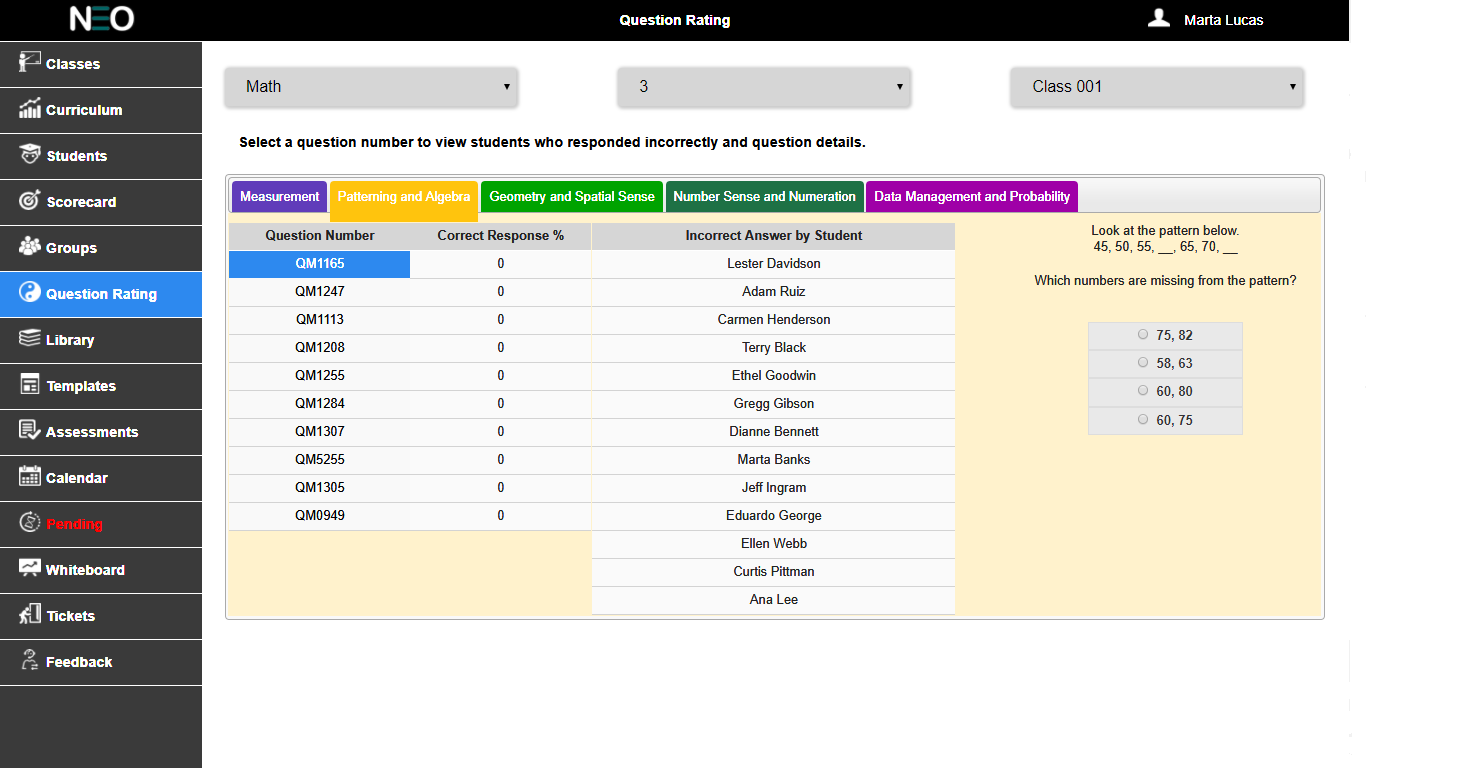
**Figure 51: Question Rating Tab**

The common UI elements on this screen will be the drop lists for Course, Grade and Class (in that order).

Below the common drop lists should be a line of text as follows: Select a question number to view students who responded incorrectly and question details.

The lower portion of the screen will be a series of tabs. Each tab will represent a strand of the selected course curriculum; so that the total number of tabs should be equal to the total number of strands in the curriculum. Each tab should be color coded to the curriculum as in other screens; see the Color Palette section for more details about Hex colors to use.

Initially, the tabs should display only two columns (as in Figure 51 above), one for Question ID and another for the calculated percent of correct responses. The database Transaction table will need to be filtered by Course, Grade and Class and then grouped by Question ID. The score result for each question should be calculated and sorted in ascending order. The first 10 will then be displayed. Rolling the mouse pointer over a Question ID will highlight that field. Clicking on a Question ID will reveal the names of all of the students in that class that responded incorrectly to the selected question. It will also reveal a preview of the selected question. See Figure 52 below.



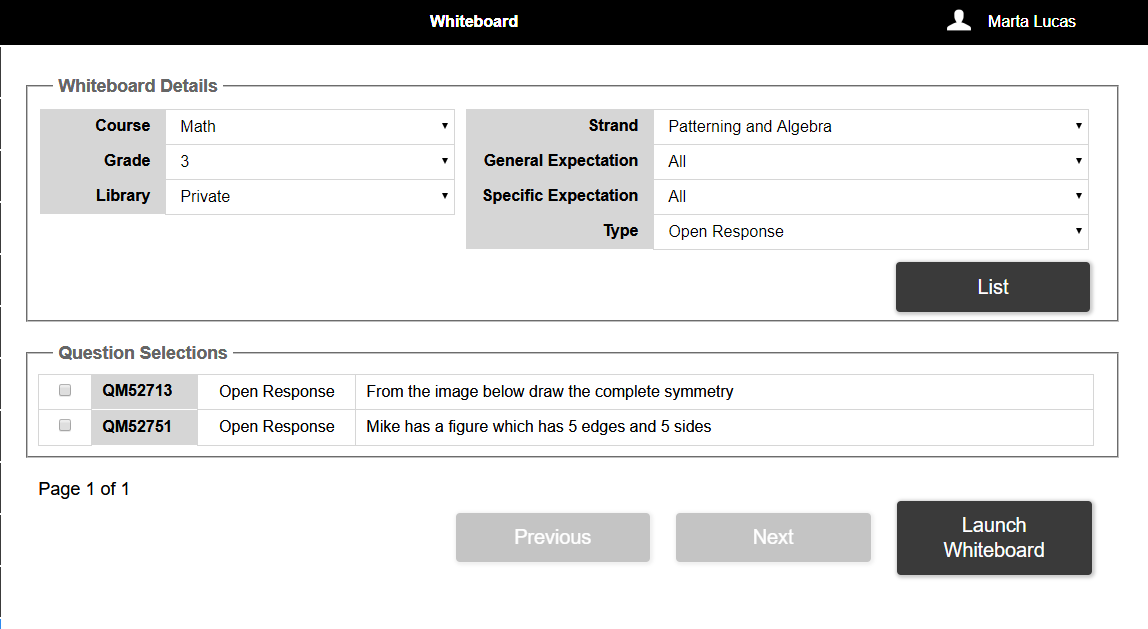
**Figure 52: Click on Question ID**

# WHiteboard

The Whiteboard tab should allow the teacher to select a question and then project that question onto an external screen. In this way they will be able to directly interact with their class when teaching lessons. There are two screens required to make this happen: the question selection screen and the whiteboard screen.

### Question Selection

The question selection screen (picture below in Figure 53) functions as a simplified assessments setup screen (few selections and nothing are ever written to the database).



**Figure 53: Whiteboard Selection Screen**

At the top of this screen, the Whiteboard Details field set should allow the user to select the Course, Grade (both filtered by the User ID) and Library (Public, Private or All); as usual, the course and grade are filtered by logged in User ID. They should also be able to choose the Strand, General Expectation, Specific Expectation, and the question type; the default selection for all of these should be “All”. On clicking of the List button, the user should be able to see the list of questions based on the user selected filters. The Question Selections field set should display a filtered list of questions (in blocks of 10) using the above selections from the Whiteboard Details. This displayed list should show Line 1 of the question, the question ID, the question type and a check box.

User can only select a single question for display on the whiteboard. Rolling the mouse pointer over a question ID should highlight this field. Clicking on the question ID should show a preview of this question.

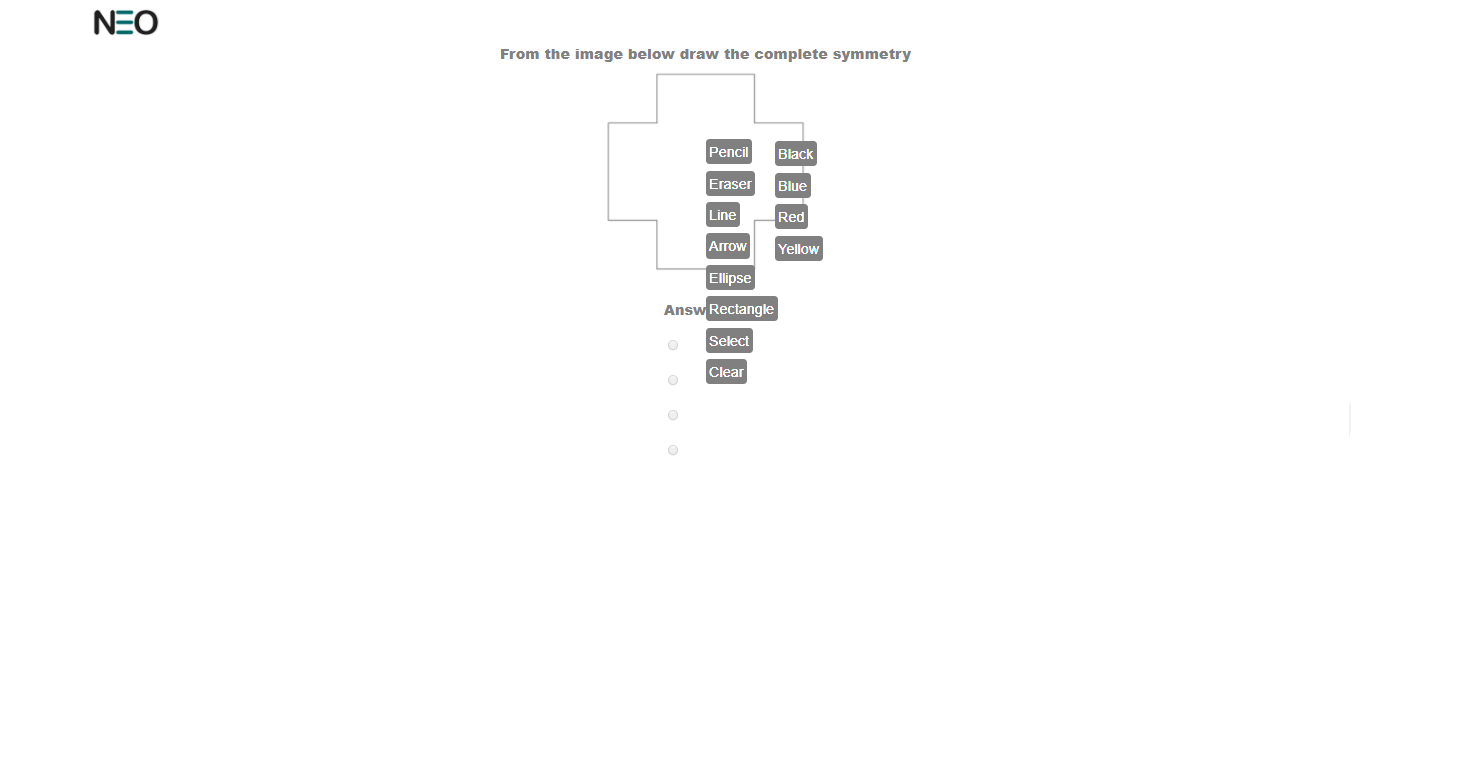
At the bottom of this screen is a button to “Launch Whiteboard” which will take the user to another screen which displays the selected question in literary canvas.

### White board

The Whiteboard is very much like the question preview except that it is full screen without any of the surrounding menus (see Figure 54) Image has to be changed.

There are issues in the literary canvas of the white board:

1. Tool set
2. For open response questions also the radio button is coming, which should not come.



**Figure 54: Whiteboard Screen**

The whiteboard will also make use of the Literally Canvas software so that the teacher can draw all over the question while teaching the lesson. The selected question is displayed as a background image and the literary canvas is laid on top of the question. The Literally Canvas tool bar should be set along the right side of the window. Clicking on the Neo logo in the top left corner of the screen should return the user to the prior screen.

# Tickets

An Exit Ticket (see Figure 55) is a test that only has a single question. It’s meant for the teacher to very quickly assign a single question to gauge the level of the class’ understanding.

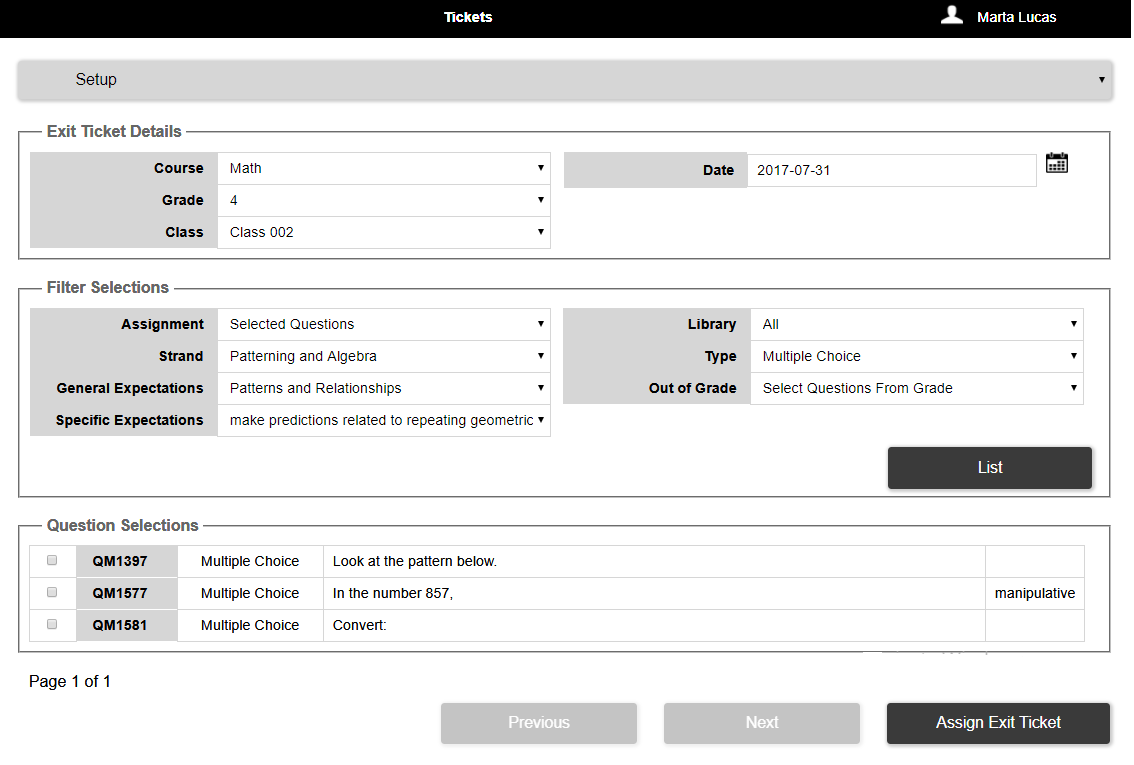
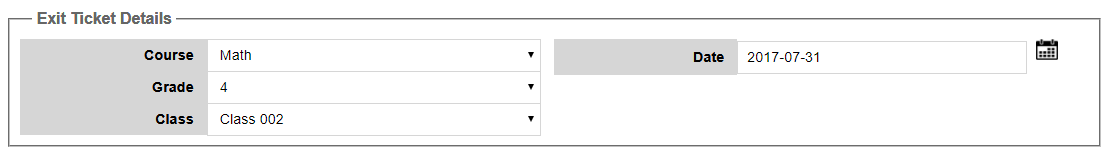


Figure 55: Exit Tickets Tab

At the top of this screen, there is a drop list with two options: Setup and Review. The Setup option looks like a simplified version of the Assessments tab. While the Review option allows the teacher to quickly see the result of the Exit Ticket.

## Setup

The Setup screen has three sections: Exit Ticket Details, Filter Selections, and Question Selection.

****

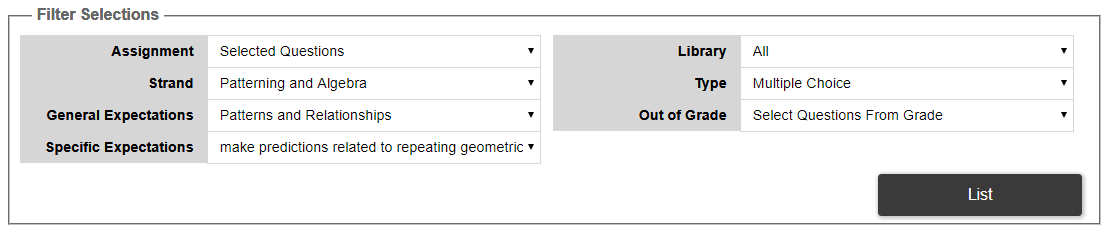
**Figure 56: Details Selections for the Exit Ticket**

The Exit Ticket Details field set contains the common drop lists for Course, Grade and Class; these are all filtered by the logged in User ID as well (for example, if Teacher1 is logged in, only the courses taught byTeacher1 are listed in the Course drop list).The only Target for an Exit Ticket is always “Class”; therefore there is no need for a drop list to select the target type.

The Class drop list should be populated by querying the database for a list of classes using the selected Course, the selected Grade, and the currently logged in User ID as filters.

The date field should default to the current date. As a rule each specific class can only have one Exit

Ticket assigned per day. The test type for Exit Ticket is always “ExitTicket” (no space and this is how it should be saved in the database).



**Figure 57: Filter Selections for the Exit Ticket**

The Filter Selection field set only has several filters but the option within those filters is restricted. The

Assignment filter should only have one option: Selected Question. In a future version of the software, there will be an option to randomly select a question, but that is not needed for this phase of the project.

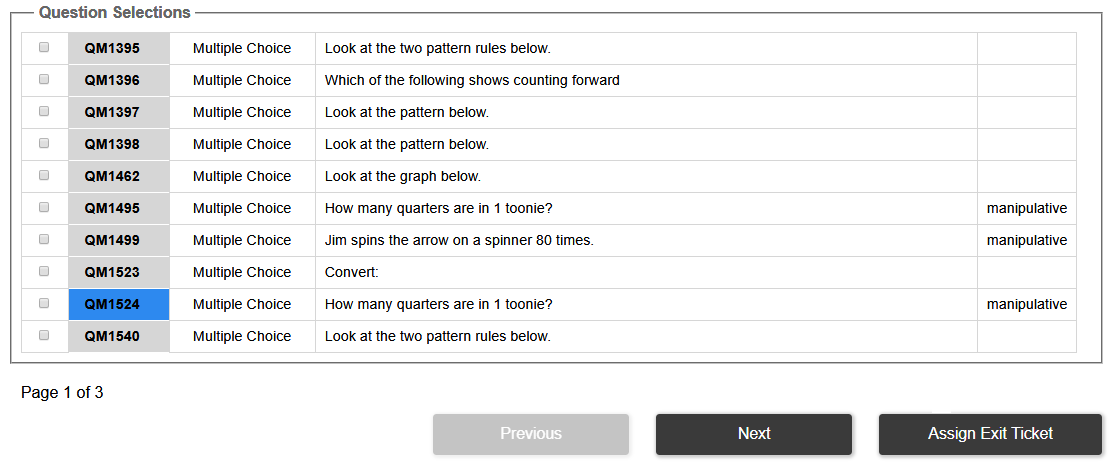
The Strand drop list should be populated with all of the available strands in the selected course.

However, “All” should NOT be an option; therefore the list should default to the first available course strand.

The Expectations drop list should be populated based on the user’s selection of Strand. For the general expectations, “All” is an acceptable option.

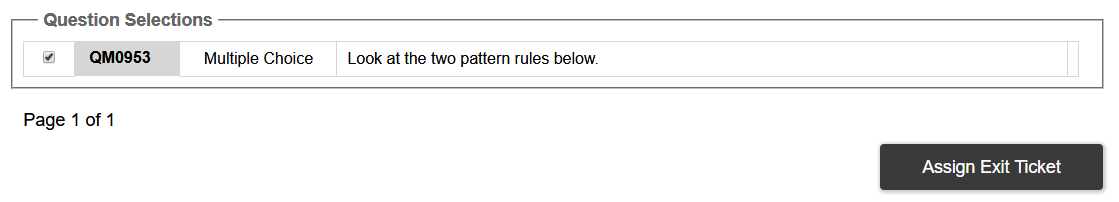
The Library drop list should have the choice of All, Public, or Private. “All” should be the default selection. The Type drop list should default to Multiple Choice.

Also, like the Assessments screen, a “List” button should be added within this field set and used to fetch the appropriate list of questions.



**Figure 58: Question Selection for the Exit Ticket**

The Question Selection field set should be a filtered list of available questions. They should be displayed in blocks of 10 questions, and the fields used are: Question ID, Question Type, and the first line of the question. Each line should be preceded by a checkbox (to be used to select the question).Rolling over the Question ID should highlight that field, while clicking on the Question ID should bring a preview of that question. When a question is selected, all other choices should be hidden (see Figure 59 below).



**Figure 59: Selected Question**

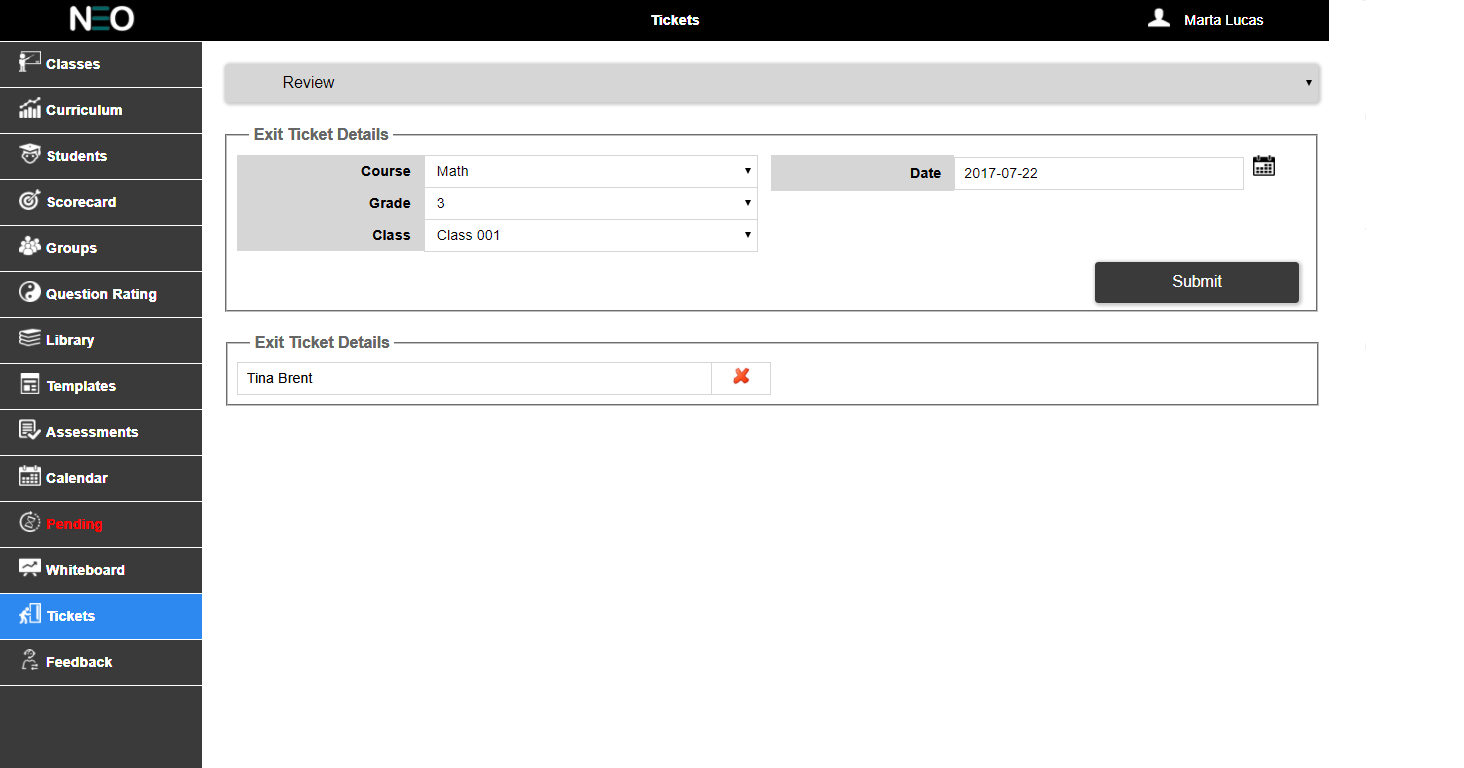
Clicking on the “Assign Exit Ticket” button, commits the test to the database. The Exit Ticket should appear on the Calendar tab just like Formal and Practice test types. Exit Tickets can only be deleted from the Calendar screen.

Doubt: without selecting a question, the user should not be able to assign an exit ticket? Till the user selects a question the Assign Exit ticket button should be disabled?

## Review

Switching to the Review option allows the teacher to see the results of a selected Exit Ticket (Figure 60).

The Exit Ticket Details field set persists to this screen and should also default to the current date.

****

**Figure 60: Exit Ticket Review**

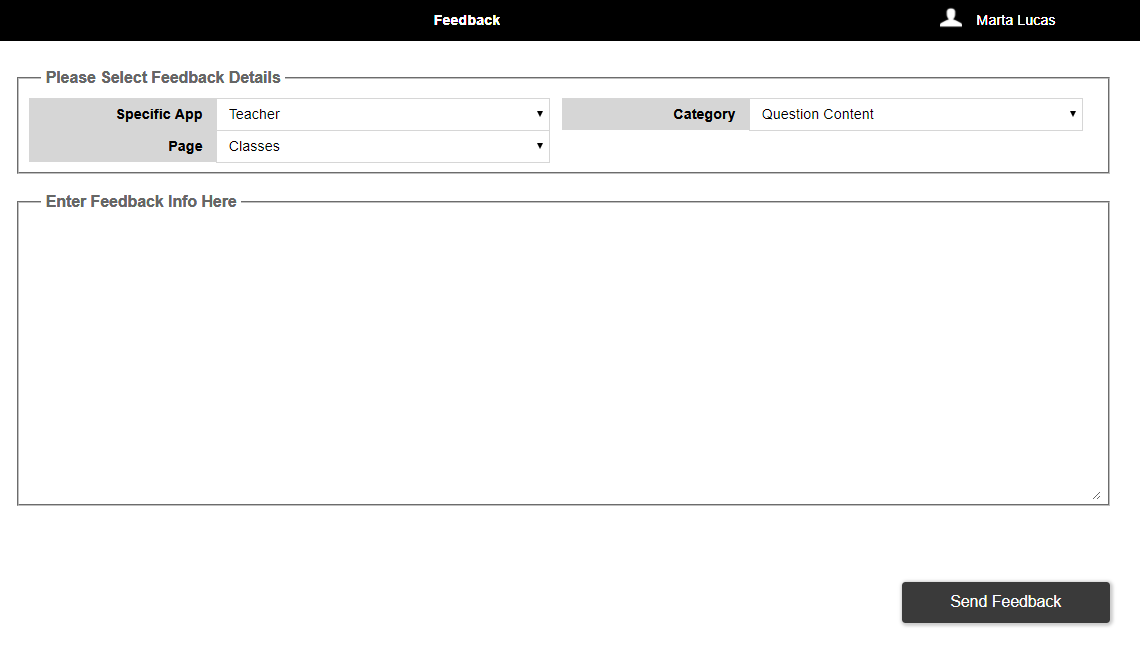
The Exit Ticket Results field set is simply a filtered list of student names (first and last) that answered the assigned Exit Ticket question incorrectly. This list is filtered by the selected Course, Grade and Class. As well as by Valid field of the database Transaction table (the value should be “False”). **Only the students with incorrect responses are ever listed here.**

There should be a visual indication after every students name that they answered incorrectly (red X), as pictured in Figure 60 above.

Here if there are no students whose ticket has to be checked and the user clicks on the Submit button then there should be an error message. As far as now there is no error message.

# Feedback

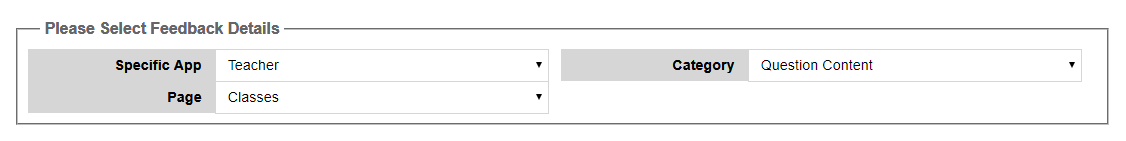
The feedback is used to get the user comments and enhancements for the teacher and student app (see Figure 61).



**Figure 61: Feedback Tab**

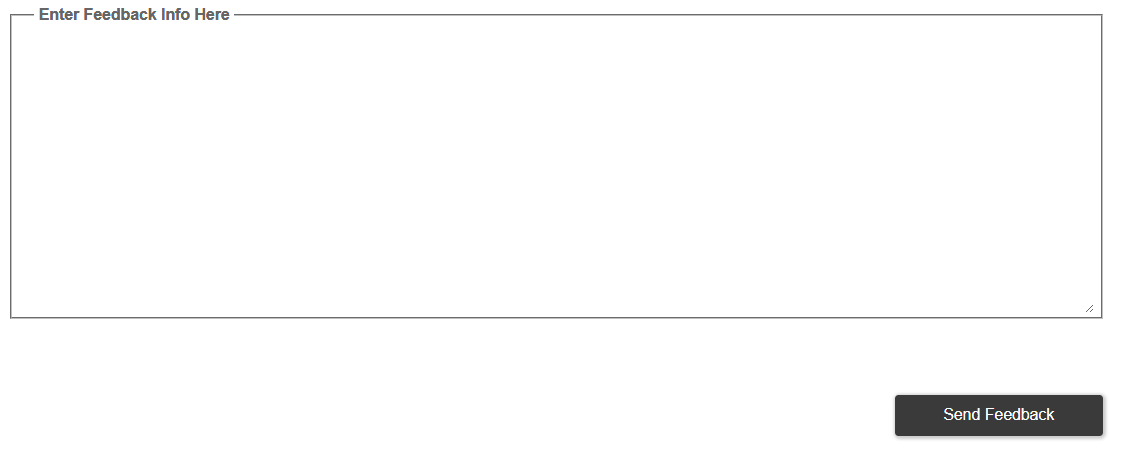
When the user first selects the Feedback tab, the dropdowns will show Feedback details that a user is interested to give (See Figure 62 below). The user has to select for which app the user wants to give feedback, the category of the feedback like general comment or question content. Based on the app the

Page dropdown should be populated. If the user chooses teacher in the Specific App, the Page dropdown should also populate with teacher app pages like classes, students, curriculum etc.



**Figure 62: Selection of Feedback Details Fieldset**

Below Selection of Feedback Details fieldset, there is another fieldset with text area where the user can enter their feedback (see Figure 63). Clicking on the Send Feedback button should send an email message to [nt@neols.com](mailto:nt@neols.com). The Message Title should be: Pilot Feedback. And the text message should automatically contain the following information: User ID, School Name, Board Name, (from the drop lists) Specific App, Page, Category, User's Name, User’s Email Address and the User's feedback.



**Figure 63: Feedback Text Area**

# Color Palette

Use the following chart as a guide to the color of UI elements. Individual curriculum strand color should follow the colors listed in the chart below and in the order displayed below.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| HEX | R | G | B | SAMPLE |
| 603CBA | 96 | 60 | 186 |  |
| 00A300 | 0 | 163 | 0 |  |
| FFC40D | 255 | 196 | 13 |  |
| 1E7145 | 30 | 113 | 69 |  |
| 9F00A7 | 159 | 0 | 167 |  |
| E3A21A | 227 | 162 | 26 |  |
| 2D89EF | 45 | 137 | 239 |  |

UI colors:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| HEX | R | G | B | SAMPLE |
| E9E9E9 | 233 | 233 | 233 |  |
| CCCCCC | 204 | 204 | 204 |  |
| D6D6D6 | 214 | 214 | 214 |  |
| 3A3A3A | 58 | 58 | 58 |  |
| 666666 | 102 | 102 | 102 |  |
| F9F9F9 | 249 | 249 | 249 |  |
| 000000 | 0 | 0 | 0 |  |
| 80AAFF | 128 | 170 | 255 |  |
| B3DBFF | 179 | 219 | 255 |  |
| 0055FF | 0 | 85 | 255 |  |
| E60000 | 230 | 0 | 0 |  |

# Figure Guide

|  |  |
| --- | --- |
| **Figure Number** | **Image Description** |
|  | Login |
|  | Home Page |
|  | Classes Tab |
|  | Top Filters |
|  | Classes Tab - Main Graph |
|  | Data Point Tool Tip - Main Graph |
|  | Strands Graph - Classes Tab |
|  | Strands Graph with Tool Tip |
|  | Class Score Details |
|  | Class Score – Tool Tip |
|  | Student Assessment Details |
|  | Assessment Question Results |
|  | Question Preview |
|  | Curriculum Tab |
|  | Top Filters |
|  | Curriculum Scores by Strand Graph |
|  | Curriculum Scores by Strand – Tool Tip |
|  | Curriculum Chart |
|  | Strand Score Details |
|  | Strand Score Details – Tool Tip |
|  | Strand Score Chart |
|  | Student Assessment Details |
|  | Assessment Question Results |
|  | Question Preview |
|  | student details tab |
|  | Historical Assessment Score Graph |
|  | Strand Scores and Notes |
|  | No Question Answered |
|  | General Expectation – Patterning and Algebra |
|  | General Expectation – Geometry and Spatial Sense |
|  | General Expectation – Number Sense and Numeration |
|  | General Expectation – Data Management and Probability |
|  | Camera Icon Preview |
|  | Question Preview |
|  | Edit Student |
|  | Confirm Save Changes |
|  | Scorecard tab |
|  | Scorecard Tab - Geometry & Spatial Sense and Patterning & Number Sense |
|  | Scorecard Tab – Number Sense & Numeration and Data Management & Probability |
|  | Scorecard - Curriculum |
|  | Groups Tab |
|  | Grouping Constraints |
|  | Grouping Results |
|  | Groups Tab Buttons |
|  | Naming a Group |
|  | Sample Group Table |
|  | Sample GroupUser Table |
|  | Create New Group Screen |
|  | Create New Group Screen |
|  | Delete Group Screen |
|  | Question Rating Tab |
|  | Click on Question ID |
|  | Whiteboard Selection Screen |
|  | Whiteboard Screen |
|  | Exit Tickets Tab |
|  | Details Selections for the Exit Ticket |
|  | Filter Selections for the Exit Ticket |
|  | Question Selection for the Exit Ticket |
|  | Selected Question |
|  | Exit Ticket Review |
|  | Feedback Tab |
|  | Selection of Feedback Details Fieldset |
|  | Feedback Text Area |

# Non Disclosure

All of the information within this document is considered to be confidential and the property of NeoLearning Systems. The software developer will treat this document, and any other information or materials disclosed or provided, as private and confidential and will not disclose, reproduce or transmit said information and materials to any other party without the written consent of Neo Learning Systems.

It is the responsibility of the software developer to ensure that all materials provided are returned to Neo Learning Systems.