NEO LEARNING SYSTEMS

DESIGN DOCUMENT
PROJECT RAPTOR – PHASE 2

SEPTEMBER 2016

VERSION 1.2

CONFIDENTIAL

1 TABLE OF CONTENTS

2 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.

This software is to be used as an in-class pilot project in September 2016 for the purpose of collecting user/institutional feedback, as well as a showcase for future product development prior to a production rollout no later than January 2017.

Web development will be executed by W5RTC (W5).

2.1 PROJECT OBJECTIVES

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

- Recreate the features and functions of the NLS demo for the user roles of Student, Teacher, Principal, Board, Ministry, and Admin and as detailed in this document.
- Recreate the visual styling of the NLS demo.
- Create the appropriate level of application security for all user roles.
- At a minimum, deliver functioning applications for both the student and teacher roles for the classroom pilot starting at the end of September 2016 and as described by the Project Order Milestones.
- Deliver functioning applications for all user roles by January 2017.

The following sub-sections correspond to the various screens or tabs showcased in the NLS demo. They are listed in order of development priority.

3 CLASSES TAB

The Classes tab (Figure 1 below) is used by the teacher role to view student assessment results from the perspective of the entire class. 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 from the Classes tab.

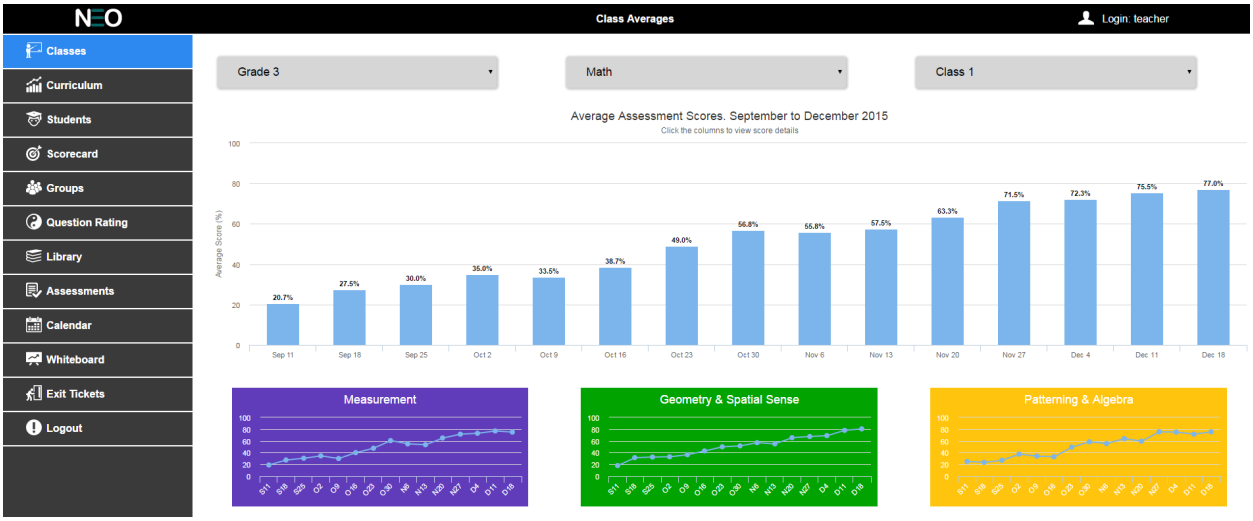


Figure 1: Classes Tab

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.

UI Elements that are unique to this screen include the main bar graph that stretches to fill the available screen width (the standard height should be hard coded – see the demo). There will also be colour coded sub-graphs 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 datapoint or bar. Also, the main graph is drillable so that clicking on a bar in the graph will take the user to another screen with a deeper analysis of the data (again, see following sections for more detail on interactivity).

If possible, the individual bars should be colour coded by test type (Formal or Practice). See the colour chart in the following section for details on color codes.

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.

3.1 FILTER SELECTIONS

The order of the drop lists should be: Course, Grade and Class (this is the order that is being used on the Library and Assessment tabs). ***Please note: this is different from how the demo currently works and should be implemented this way on all tabs being developed.***

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.

3.2 AVERAGE ASSESSMENT SCORES GRAPH

The Average Assessment Scores graph is the main graph on this tab (see Figure 2 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.

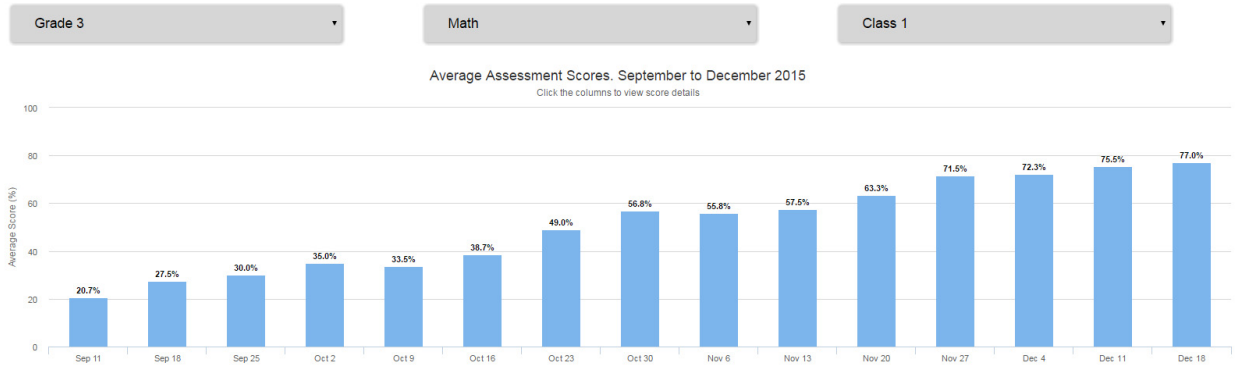


Figure 2: Average Assessment Score Graph

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 or a graph legend.

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 (select a date format that looks best given that there may be a lot of columns as the year comes to an end).

The colour of the bars on the graph will vary by assessment type (Formal or Practice; the Exit Ticket type should always be EXCLUDED from this graph), and if the mouse pointer rolls over a bar. ***Please note: this is different than how the demo currently works.*** Use the Hex colour table below (Table 1) as a guide (see the Colour Palette section for specific details).

Practice Assessment	80AAFF	
Practice Assessment Rollover	B3DBFF	
Formal Assessment	0055FF	
Formal Assessment Rollover	B3DBFF	

Table 1: Assessment Graph Colour Palette

Rolling the mouse pointer over a specific bar should reveal a tooltip with specific information about that data point (see Figure 3 below). The tooltip should have a title (Class Average), the specific test date, and the percent average for the class on that test.

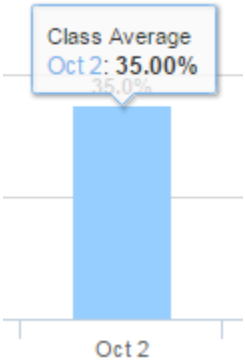


Figure 3: Tooltip for Class Average Graph

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 September 11th bar, drills down into that specific test and displays individual student scores for that test. See the Section 3.1.4 "Class Score Details" for more details on this.

3.3 STRAND GRAPHS

At the bottom of Classes tab (see Figure 4 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 (the demo only shows three), so there should be 5 individual charts displayed on the Classes tab.

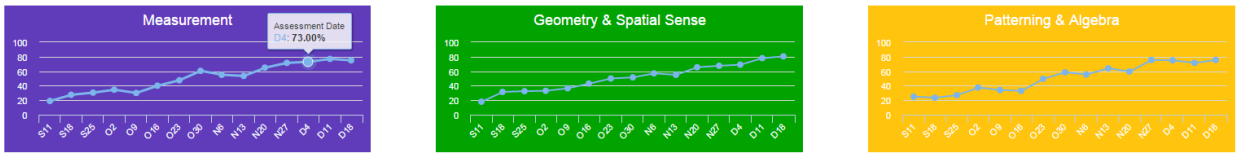


Figure 4: Curriculum Strand Graphs

These graphs should be colour coded by strand (following the guide in the Colour Palette section of this document), and the strand colours 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 (see the demo).

Given the tight spacing on these graphs, the date along the x-axis can be abbreviated. The month can be reduced to a single letter followed by the day number. See Figure 5 below.

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.

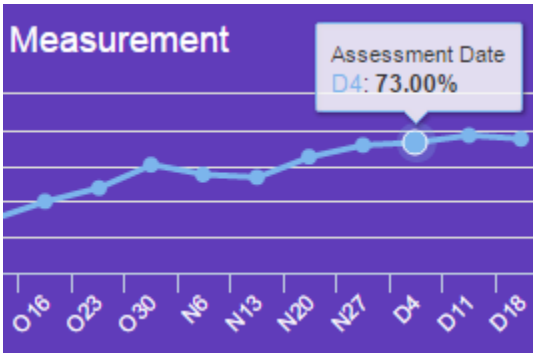


Figure 5: Abbreviated Date & Tooltip for Strand Graphs

3.4 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 6 below).

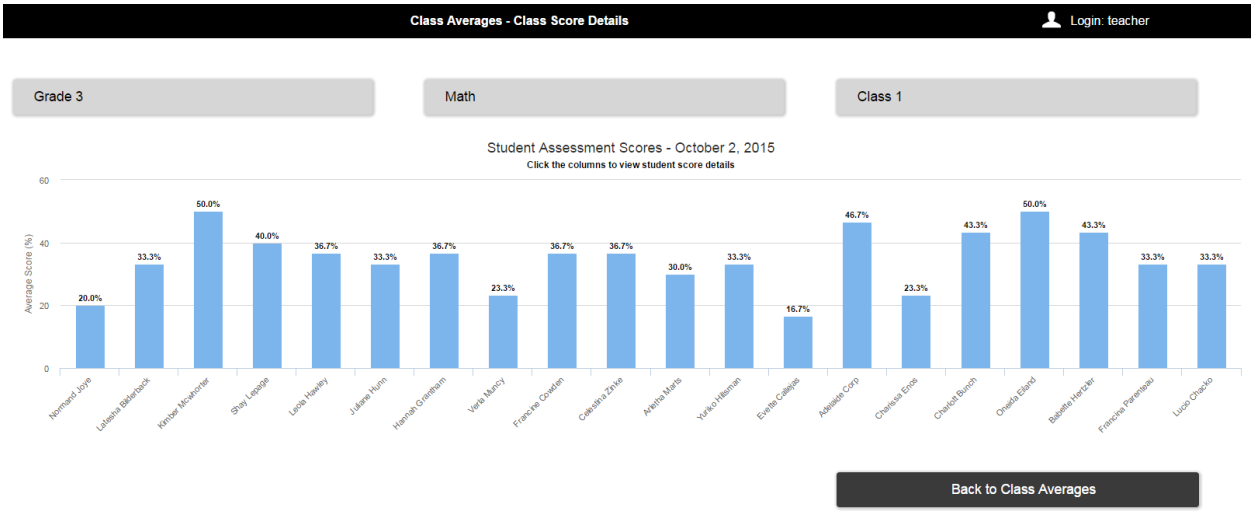


Figure 6: Class Score Details Screen

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. ***Please note: this is different than in the demo.***

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

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 7). 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 colouring and highlighting should function and be the same as in the prior screen. Bar 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).

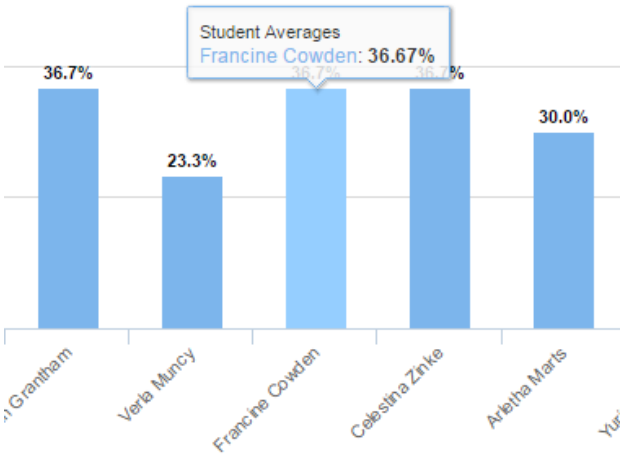


Figure 7: Graph Labels and Tooltips

3.5 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.

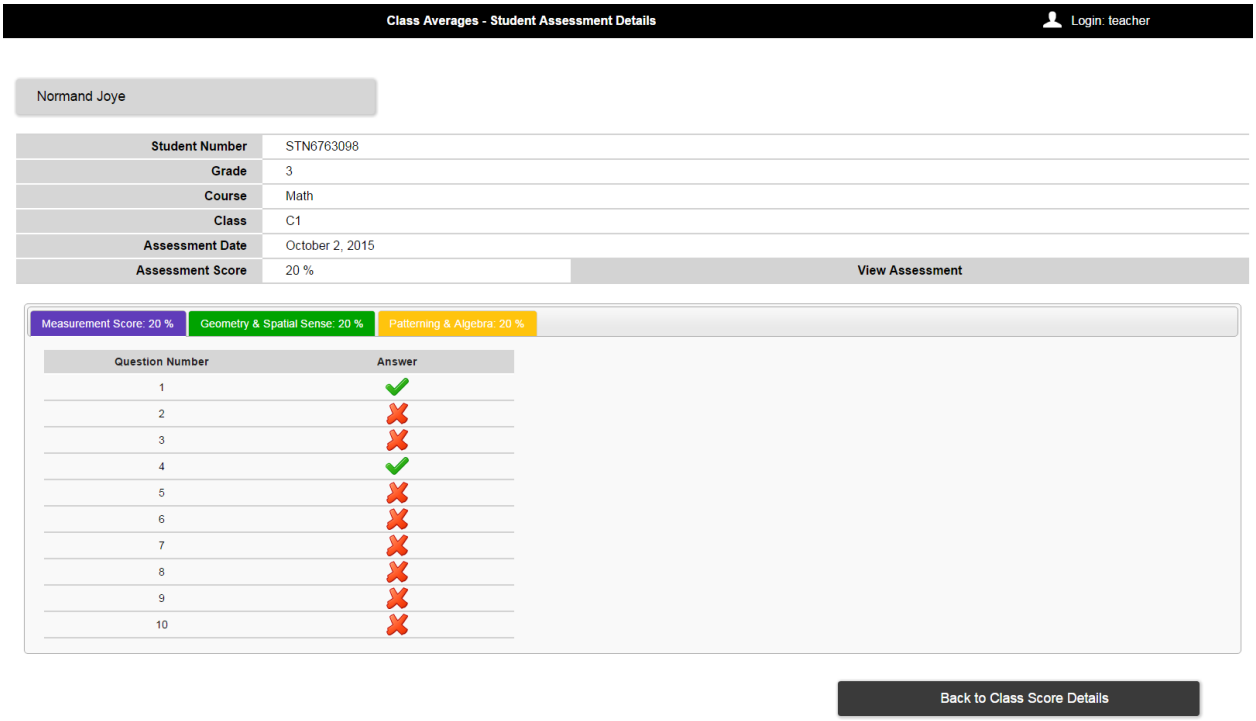


Figure 8: Student Assessment Details Screen

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 test date, and the student's score for that selected test.

The "View Assessment" button can be ignored for now; this can be a stretch goal if there is time after October milestone.

The lower portion of the screen should display colour 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 9).

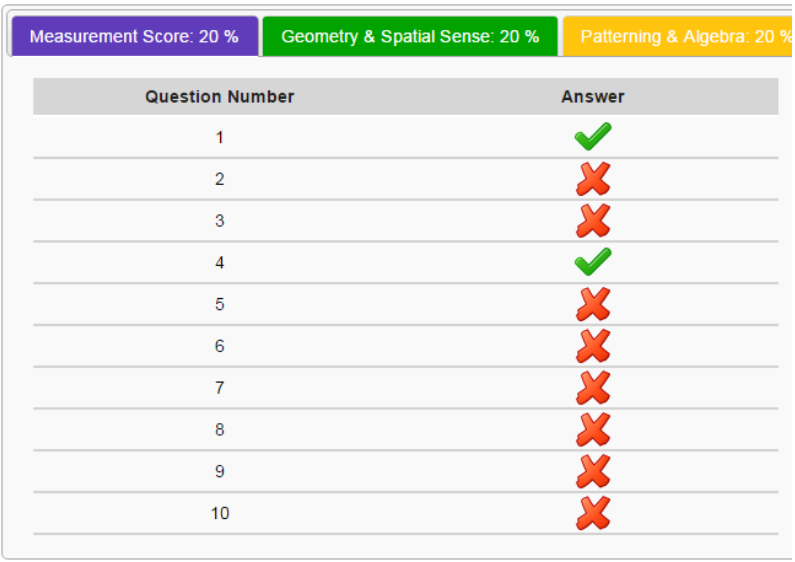


Figure 9: Assessment Question Results

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 10 below).

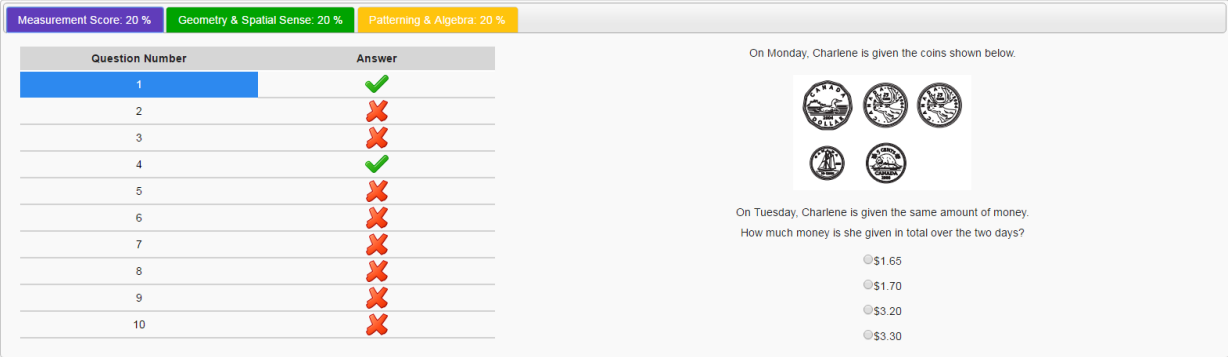


Figure 10: Question Preview

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

4 GROUPS TAB

The Groups tab (Figure 11 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.

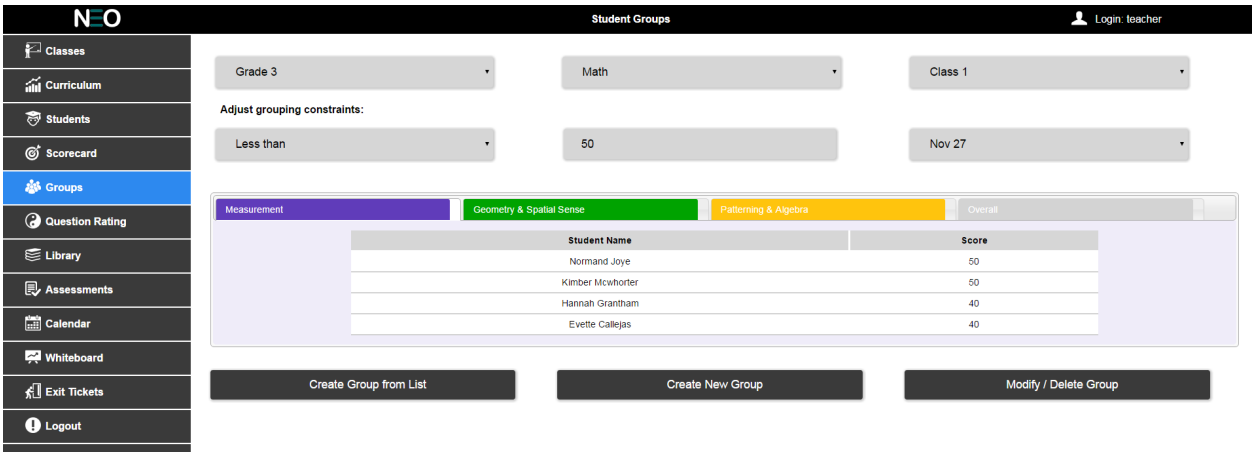


Figure 11: Groups Tab

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.

4.1 FILTER SELECTIONS

The order of the drop lists should be: Course, Grade and Class (this is the order that is being used on the Library and Assessment tabs). ***Please note: this is different from how the demo currently works and should be implemented this way on all tabs being developed.***

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.

4.2 GROUPING CONSTRAINTS

There will be three more user configurable constraints on this tab under the heading “Adjust grouping constraints:”. See Figure 12 below.

Figure 12: 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.

4.3 GROUPING RESULTS

The lists of students generated by the above filters and constraints should be organized by curriculum strands and overall test scores (colour coded as in other areas of the software for consistency). The HTML field set backgrounds should also be lightly coloured to reflect the tab the user is currently viewing (see the figure below or the demo for an example of this).

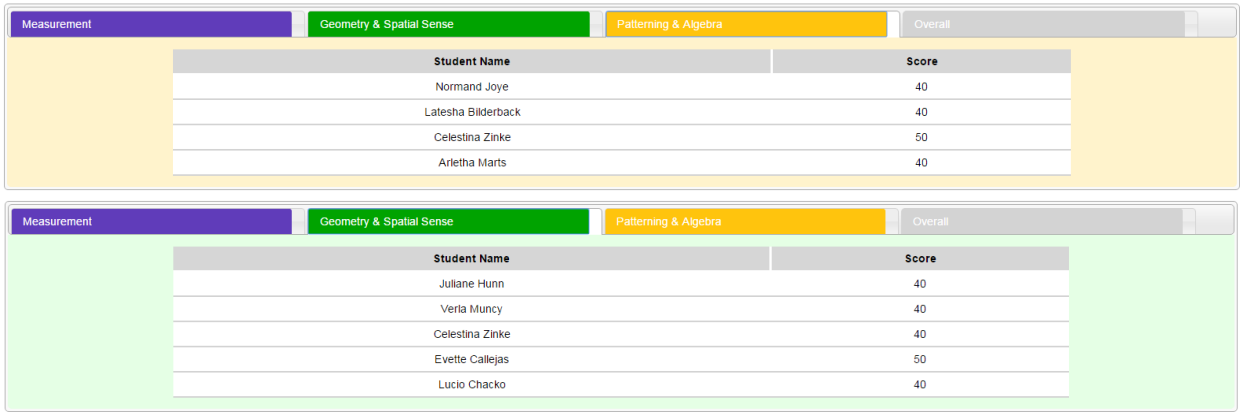


Figure 13: Colour Coding by Tab

4.4 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.

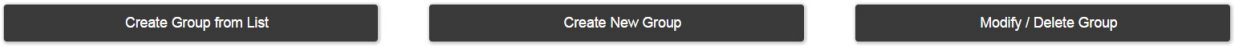


Figure 14: Groups Tab Buttons

4.5 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 15 below). Unlike the demo, the user should be able 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 (for example, if the current month is October 2016, then the end date should default to June 30, 2017). Figure 15 does not show the end date option. ***Please note, this is different than how the demo functions.***

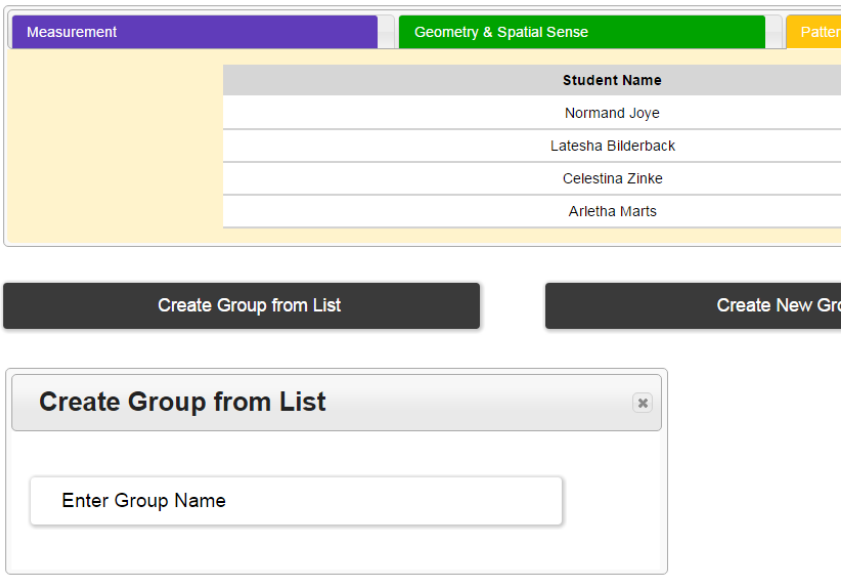


Figure 15: Naming a Group

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.

GroupID	ClassID	EndDate	GroupName
GRP000001	CL00201	30/06/2017	Group 1
GRP000002	CL00089	01/04/2017	Group 2
GRP000003	CL00227	16/06/2017	Group 3
GRP000004	CL00210	10/04/2017	Group 4
GRP000005	CL00197	30/06/2017	Group 5
GRP000006	CL00009	26/03/2017	Group 6
GRP000007	CL00158	29/03/2017	Group 7
GRP000008	CL00205	03/05/2017	Group 8
GRP000009	CL00001	08/02/2017	Group 9
GRP000010	CL00101	19/05/2017	Group 10
GRP000011	CL00126	27/04/2017	Group 11
GRP000012	CL00181	05/06/2017	Group 12

Figure 16: 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.

GroupID	UserID
GRP000001	000001070
GRP000001	000001145
GRP000001	000001245
GRP000001	000001545
GRP000001	000001870
GRP000001	000003045
GRP000001	000003545
GRP000001	000003695
GRP000001	000004270
GRP000001	000004595

Figure 17: Sample GroupUser Table

4.6 CREATE NEW GROUP

Clicking on the “Create New Group” button will move the user to another screen for just that purpose (See Figure 18 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 that the user can use to select the students in the new group.

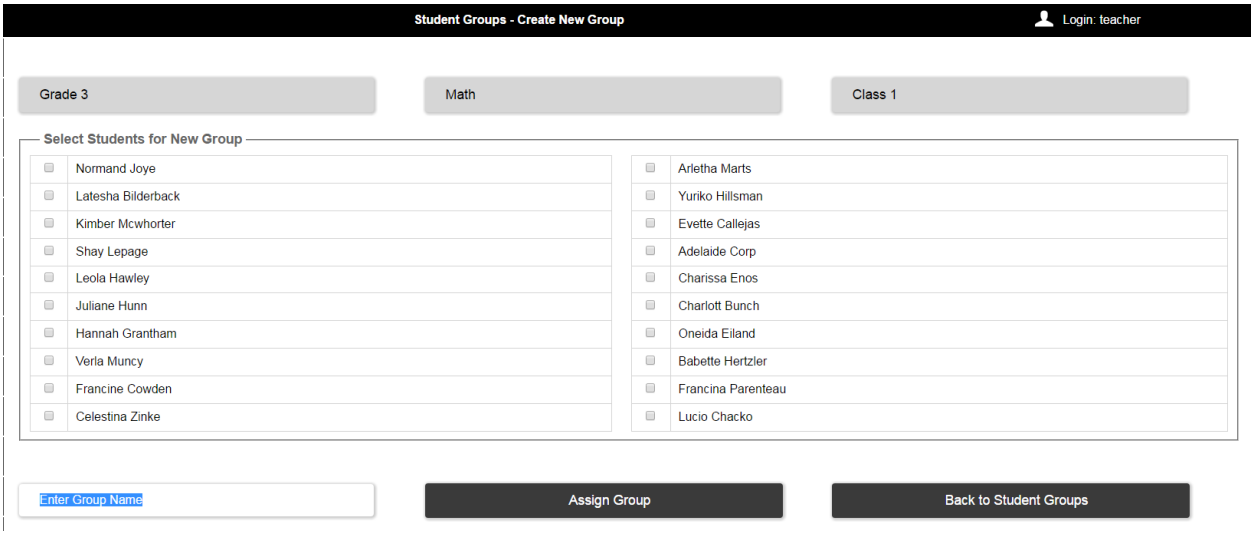


Figure 18: Create New Group Screen

Below the grid there should be an input box for the user to name the group, and there should also be a date input box for the user to enter an end date for this group (date input not shown in Figure 18 above).

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.

4.7 MODIFY / DELETE GROUP

Clicking on the "Modify/Delete Group" button will move the user to another screen for just that purpose (See Figure 19 below). This screen does not exist in the demo, so a potential mockup has been made of what it should approximately look like. If you have any thoughts on how to better execute on this screen, feel free to share them.

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. **Error checking 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. ***This is different than how the demo currently functions.***

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 boxes 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 (the order is wrong in Figure 19) and logged in User ID. 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.

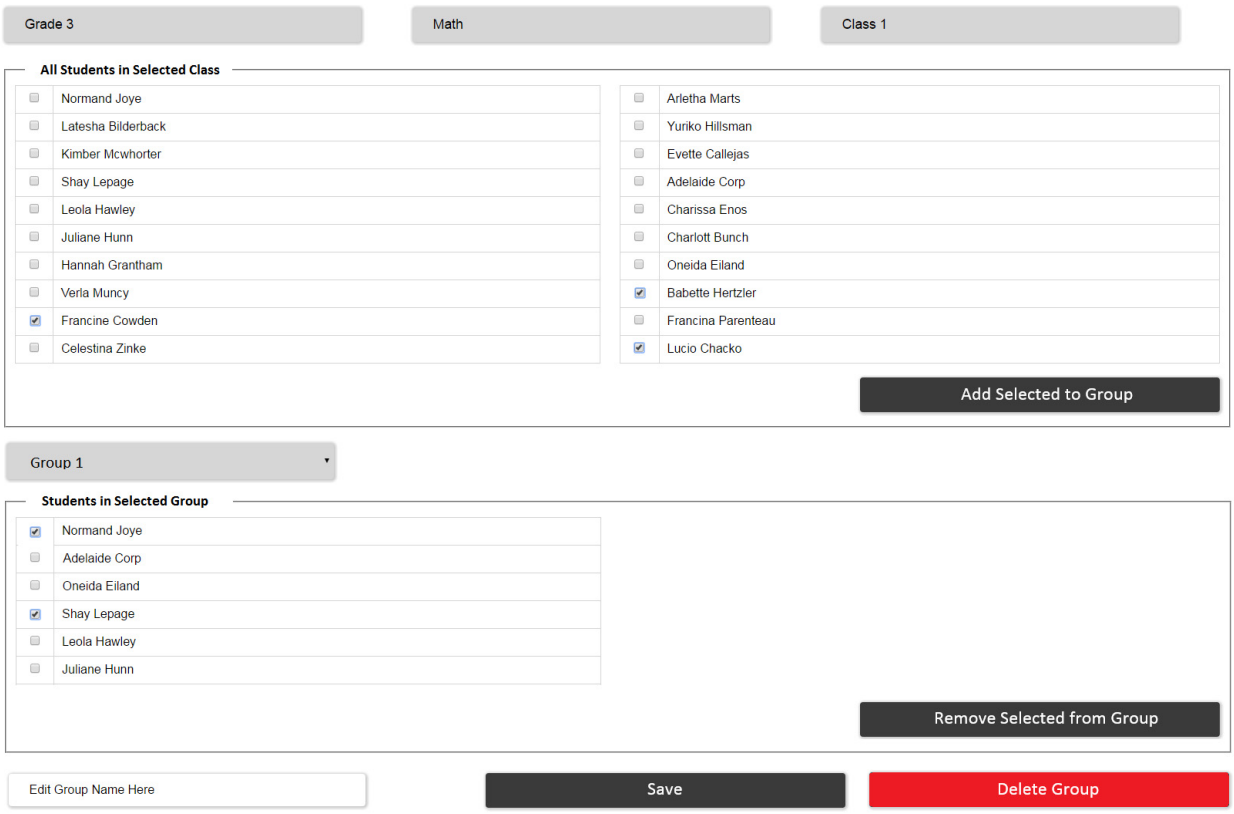


Figure 19: Modify / Delete Group Screen

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. This button is not shown in Figure 19.

- "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 coloured red (Hex #E60000).

5 WHITEBOARD TAB

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.

5.1.1 QUESTION SCREEN

The question selection screen (picture below in Figure 20) functions as a simplified assessments setup screen (few selections and nothing is ever written to the database).

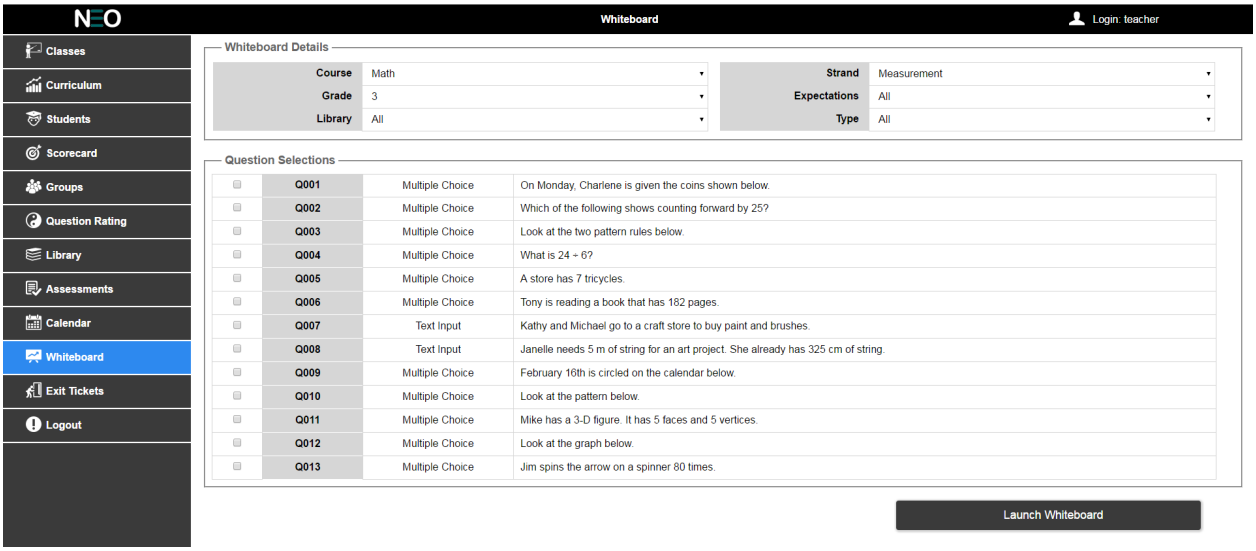


Figure 20: Whiteboard Selection Screen

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 (not shown in Figure 20), and the question type; the default selection for all of these should be "All".

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; please review how the demo functions.

Rolling the mouse pointer over a question ID should highlight this field. Clicking on the question ID should show a preview of this question (just like the Assessments tab does).

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.

5.1.2 WHITEBOARD SCREEN

The Whiteboard is very much like the question preview except that it is full screen without any of the surrounding menus (see Figure 21).

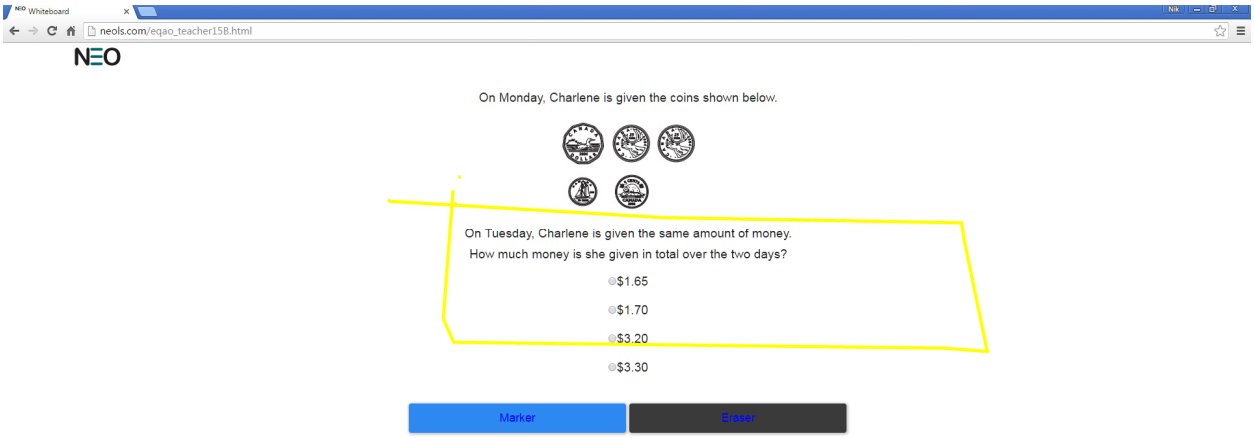


Figure 21: Whiteboard Screen

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. To accomplish this, the selected question should either be the background image for the Literally Canvas element (not shown in Figure 21), or HTML layers should be used so that the question is the bottom layer while the Literally Canvas element is the top layer.

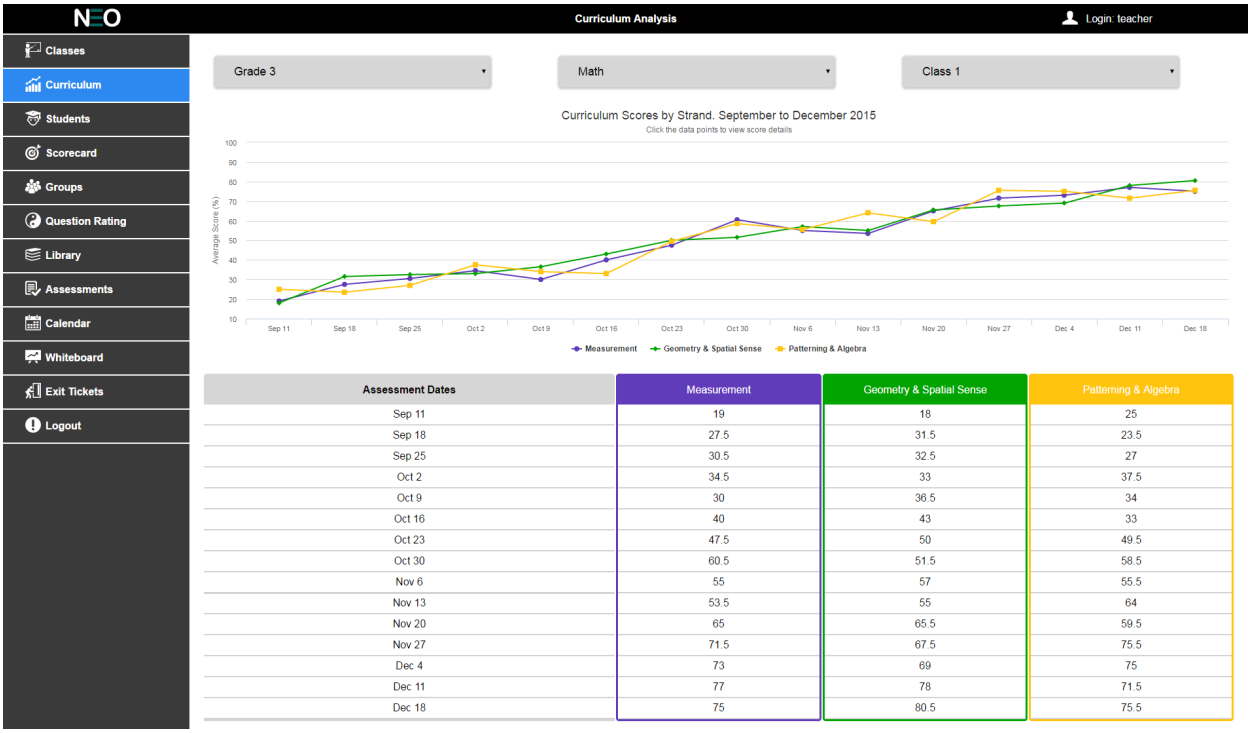
The Literally Canvas tool bar should be set along the bottom of the window or along the right side of the window; we’ll determine the permanent location of the tool bar during the build of this screen.

Please note: this is different than how the demo works. The demo makes use of a different software canvas element; Literally Canvas is the preferred option for this screen and in other areas of the software where drawing is required.

Clicking on the Neo logo in the top left corner of the screen should return the user to the prior screen.

6 CURRICULUM TAB

The Curriculum tab (Figure 22 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 from the Classes tab.



Assessment Dates	Measurement	Geometry & Spatial Sense	Patterning & Algebra
Sep 11	19	18	25
Sep 18	27.5	31.5	23.5
Sep 25	30.5	32.5	27
Oct 2	34.5	33	37.5
Oct 9	30	36.5	34
Oct 16	40	43	33
Oct 23	47.5	50	49.5
Oct 30	60.5	51.5	58.5
Nov 6	55	57	55.5
Nov 13	53.5	55	64
Nov 20	65	65.5	59.5
Nov 27	71.5	67.5	75.5
Dec 4	73	69	75
Dec 11	77	78	71.5
Dec 18	75	80.5	75.5

Figure 22: Curriculum Tab

Figure 22 only shows three strands, however the software should produce a line for all strands in a selected course. ***Please note, this is different than how the demo works.***

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 (the standard height should be hard coded – see the demo). 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 datapoint. 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.

6.1 FILTER SELECTIONS

The order of the drop lists should be: Course, Grade and Class (this is the order that is being used on the Library and Assessment tabs

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 select this tab. This should be common for all tabs related to data analysis.

6.2 CURRICULUM SCORES BY STRAND GRAPH

The Curriculum Scores by Strand graph is the main graph on this tab (see Figure 23 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.

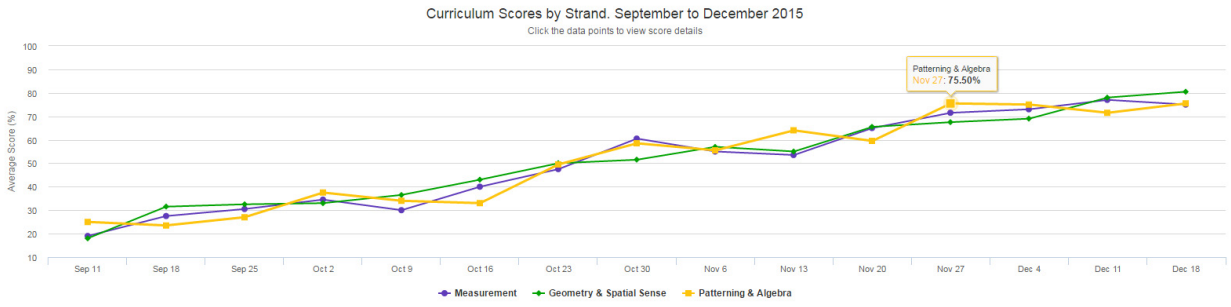


Figure 23: Curriculum Scores by Strand Graph

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 colour coded by Strand (using the same colour scheme for Strands used on the Classes tab).

Rolling the mouse pointer over a specific data point should reveal a tooltip with specific information about that data point (see Figure 24 below). The tooltip should have a title (Strand), the specific test date, and the percent average for the class on that test strand.

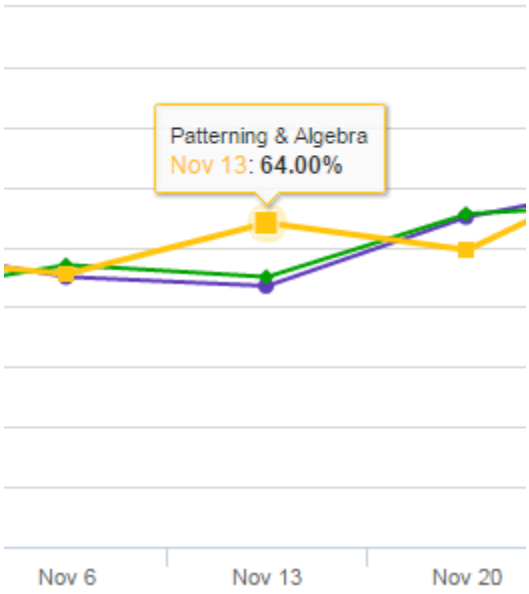


Figure 24: Data Point Tool Tip

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 September 11th data point, drills down into that specific test and displays individual student scores for that test by strand. See the Section 6.4 “Student Strand Scores” for more details on this.

6.3 CURRICULUM CHART

The Curriculum chart is simply a numerical representation of the data displayed on the chart above it. The example from the demo in Figure 25 only shows three strands, however the software should show all strands for the selected course. ***Please note, this is different than how the demo functions.***

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 colour to indicate that they are different than the practice tests. Use Hex # 0055FF to highlight these rows. ***Please note, this is different than how the demo functions.***

Assessment Dates	Measurement	Geometry & Spatial Sense	Patterning & Algebra
Sep 11	19	18	25
Sep 18	27.5	31.5	23.5
Sep 25	30.5	32.5	27
Oct 2	34.5	33	37.5
Oct 9	30	36.5	34
Oct 16	40	43	33
Oct 23	47.5	50	49.5
Oct 30	60.5	51.5	58.5
Nov 6	55	57	55.5
Nov 13	53.5	55	64
Nov 20	65	65.5	59.5
Nov 27	71.5	67.5	75.5
Dec 4	73	69	75
Dec 11	77	78	71.5
Dec 18	75	80.5	75.5

Figure 25: Curriculum Chart

The Strand columns in the chart should be colour coded the same as the lines on the graph. See the Colour Palette section for more details.

The data in the Strand columns is the overall score for the selected class on that particular strand.

Rolling the mouse pointer over a test date should highlight the date cell (use Hex # B3DBFF). Clicking the highlighted cell will take the user to another screen; the Strand Scores Detail screen.

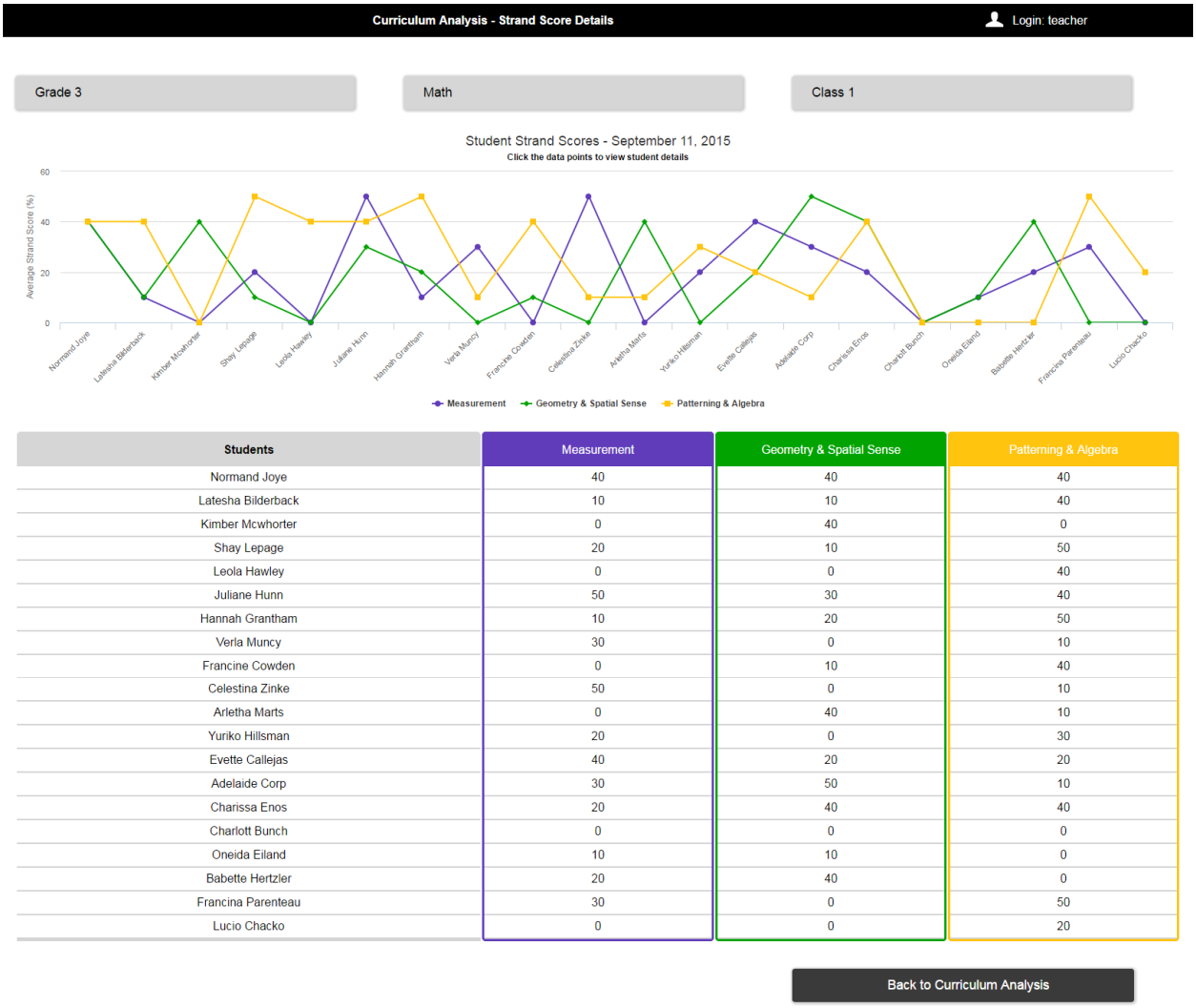
6.4 STUDENT STRAND SCORES

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 26 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. ***Please note: this is different than in the demo.***

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

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 colours. Each x-axis data points should be labeled with the students name (at an angle for space saving, see Figures 26 and 27).



Students	Measurement	Geometry & Spatial Sense	Patterning & Algebra
Normand Joye	40	40	40
Latesha Bilderback	10	10	40
Kimber Mcwhorter	0	40	0
Shay Lepage	20	10	50
Leola Hawley	0	0	40
Juliane Hunn	50	30	40
Hannah Grantham	10	20	50
Verla Muncy	30	0	10
Francine Cowden	0	10	40
Celestina Zinke	50	0	10
Arletha Marts	0	40	10
Yuriko Hillsman	20	0	30
Evette Callejas	40	20	20
Adelaide Corp	30	50	10
Charissa Enos	20	40	40
Charlott Bunch	0	0	0
Oneida Eiland	10	10	0
Babette Hertzler	20	40	0
Francina Parenteau	30	0	50
Lucio Chacko	0	0	20

Figure 26: Strand Score Details Screen

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 27 below).

Line colouring 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 into the data to the Student Assessment Details screen.

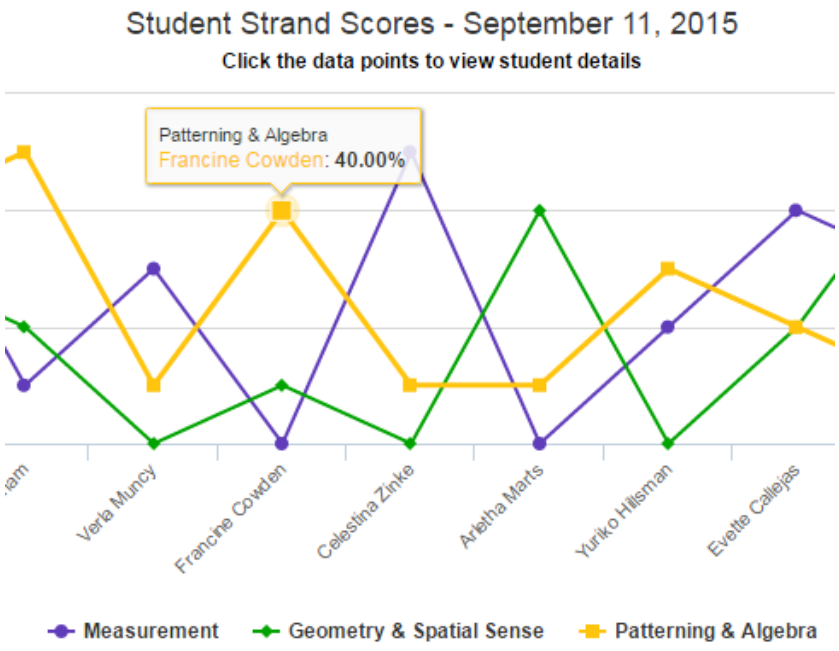


Figure 27: Tool Tip

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).

6.5 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 test date, and the student's score for that selected test.

The "View Assessment" button can be ignored for now; this can be a stretch goal if there is time after October milestone.

The lower portion of the screen should display colour 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.

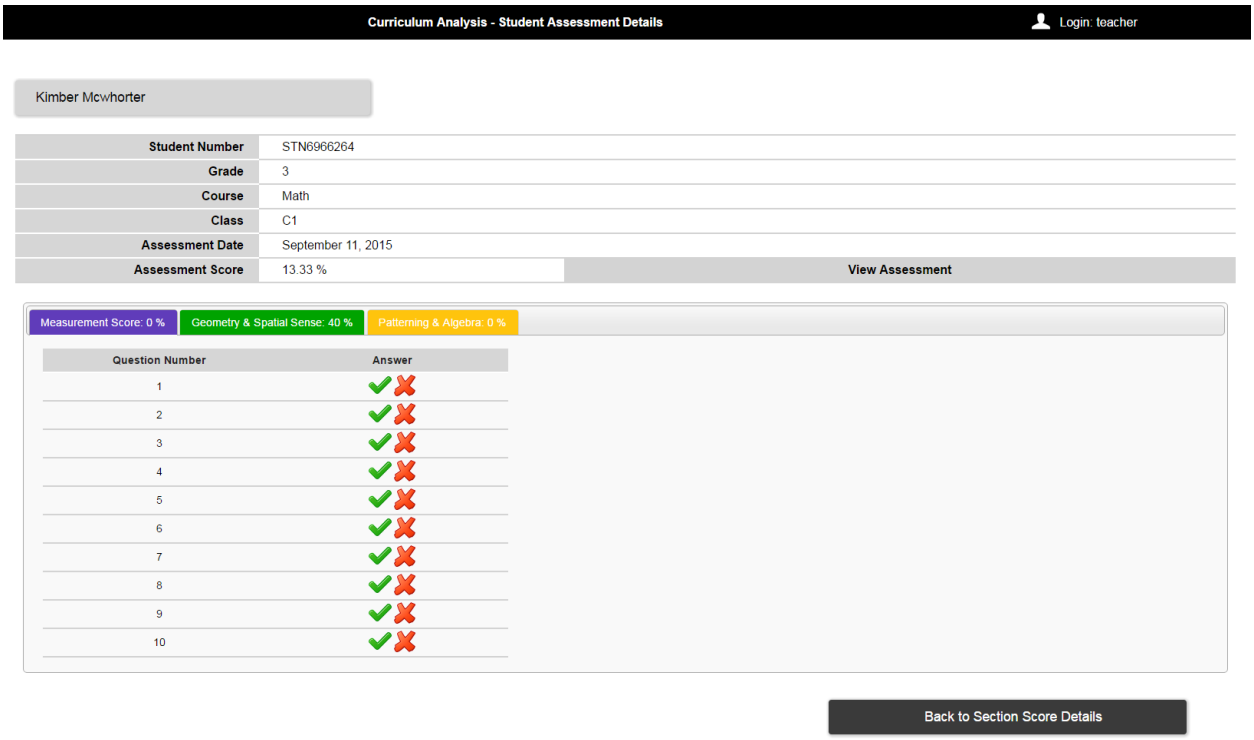


Figure 28: Student Assessment Details Screen

The student's result for that question (right or wrong) is displayed along with each question ID number (Figure 29).

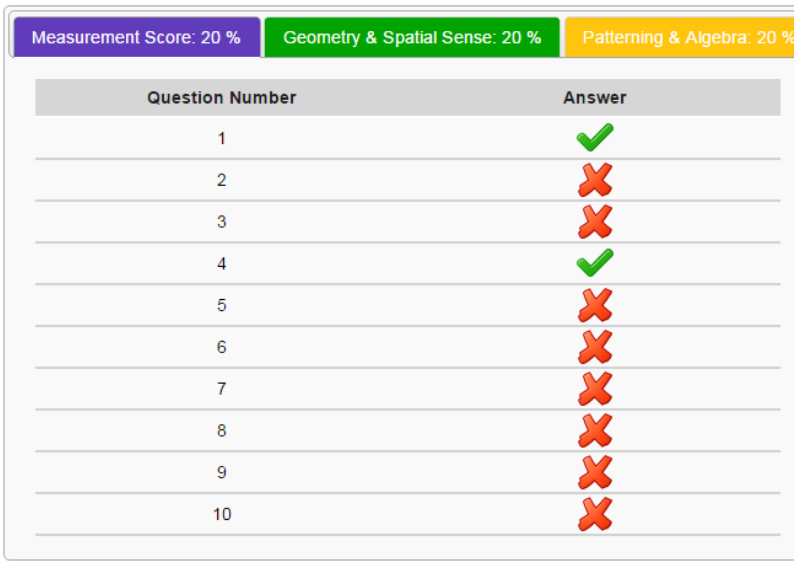


Figure 29: Assessment Question Results

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 30 below).

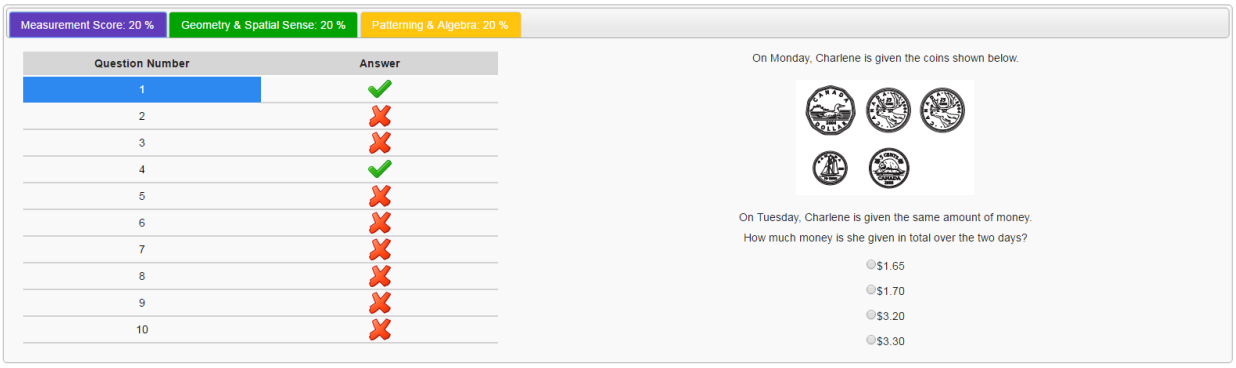


Figure 30: Question Preview

The "Back to Strand Scores" button will return the user to the prior screen with all their previous filters and selections still in force.

7 STUDENTS TAB

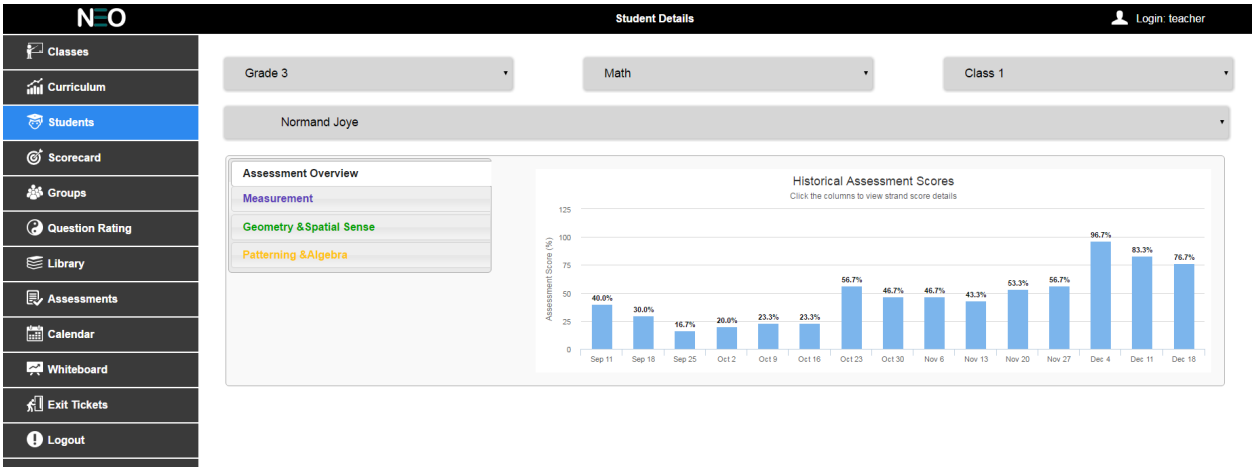


Figure 31: Student Details Tab

8 COLOUR PALETTE

Use the following chart as a guide to UI colouring. Curriculum strand colours should follow the chart colours 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 colours:

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	

9 PROJECT DELIVERABLES

The project deliverables section of this document does NOT represent the whole of the work required for this project, rather it is meant to facilitate project discussions, meetings and updates. This document should be considered in its entirety to successfully deliver this project.

This list of deliverables has been broken down into sections that follow those outlined in the preceding portion of this document. Therefore, detailed information on specific list items can be found by simply referring to prior document sections of the same name (and their associated sub-sections).

9.1 CLASSES TAB

- ☐ Build common UI elements for Course, Grade and Class drop lists
- ☐ Filter selections should be in order of Course, Grade and Class
- ☐ Build the Average Assessment Scores Graph (the Main graph on this tab)
- ☐ Main graph should be sensitive to valid class dates
- ☐ Main graph should have all of the correct titles, intervals and data point labels
- ☐ Bar colours should be sensitive to test type and rollovers
- ☐ Implement tool tips for the main graph
- ☐ Implement drill down to Class Score Details screen when clicking on a specific bar on the main graph
- ☐ Implement colour coded strand graphs
- ☐ Strand graphs follow the same dates and filtered data as the main graph
- ☐ Strand graphs should have correct titles, intervals and data point labels
- ☐ Implement tool tips for the strand graphs
- ☐ Build the Class Scores Detail (CSD) screen
- ☐ Build the common read-only UI elements for Course, Grade and Class
- ☐ CSD graph should be sensitive to selected filters and the user selected date
- ☐ CSD graph should have all the correct titles, intervals and data labels

- ☐ Build the CSD graph functionality for rollovers, tool tips, and drill downs to the Student Assessment Details screen
- ☐ Implement the correct colour coding for the CSD graph (sensitive to test type and rollovers)
- ☐ Implement the return to prior screen button on the CSD screen
- ☐ Build the Student Assessment Details (SAD) screen
- ☐ Build the strand tabs for the SAD screen and display the questions and selected student responses on the appropriate tab
- ☐ Build the question rollover and preview functionality
- ☐ Implement the return to prior screen button on the SAD screen

9.2 GROUPS TAB

- ☐ Build the Groups screen
- ☐ Add the common UI elements for filtering Course, Grade and Class
- ☐ Implement the logic for calculating the individual strand and overall student scores
- ☐ Build the unique UI elements for adjusting constraints; Less than/Greater than selector, Average Score spinner, and Test Date drop list
- ☐ Organize query results using tabs for individual strands and overall score
- ☐ Colour code the tabs and backgrounds
- ☐ Implement buttons for “Create New Group from List” , “Create New Group”, and “Modify/Delete Group”
- ☐ Build functionality for “Create New Group from List”; group naming and end date
- ☐ Build screen and functionality for “Create New Group”
- ☐ Build screen and functionality for “Modify/Delete Group”
- ☐ Implement the error checking needed for the modify/delete group screen

9.3 WHITEBOARD TAB

- ☐ Build the Question Selection screen
- ☐ Implement the appropriate queries and filters for the question screen

- ☐ Only allow a single question to be selected at a time; review the demo functionality
- ☐ Question ID highlight on rollover along with question preview when clicked on
- ☐ Launch whiteboard button
- ☐ Build the Whiteboard screen
- ☐ Implement Literally Canvas for drawing on top of the selected question
- ☐ Configure Literally Canvas toolbar either at the bottom or right side of the screen
- ☐ Implement return to prior screen by clicking on the Neo logo

9.4 CURRICULUM TAB

- ☐

9.5 STUDENTS TAB

- ☐

10 SOFTWARE

After consulting with NC, all third party software required for this project will be provided by NLS. However, every consideration should be made for using open source software for this project; for example, Cassandra database, charts from Highcharts.com, etc.

11 CONTACT INFORMATION

If you have any questions, please contact either:

Nick Tomljenovic
Email: nt@neols.com
Telephone: 647-808-1160

Ryan Solomon
Email: ryan@neols.com

12 NON-DISCLOSURE

All of the information within this document is considered to be confidential and the property of Neo Learning 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.