Data Sharing

Physics IQP A09

Stephen Mann
Advisors: Prof. Germano Iannacchione, Prof. Frederick Hutson

10/26/2009

Table of Contents

Acknowledgements

Abstract

WPI received a grant from the David Foundation for improving their Physics labs. This paper describes the formulation, implementation, testing and analysis of a data sharing system for use in the Physics labs. Its uses include real-time data graphing during labs, allowing faculty and students to identify unusual problems quickly; it automatically sends data to each student in a group, reducing time spent shutting down at the end of the lab; and it allows students to enter data online to be saved into a central database accessible by faculty. The system uses stock Microsoft SharePoint and Office functionality to perform its tasks, ensuring that the system is maintainable in the future. The final product may act as a base for future improvements, extensions, and as a system for other departments to use for their labs.

Introduction

Background

WPI received a grant from the Davis Education Foundation to improve the quality of its Physics labs. Part of this improvement went into implementing a system that allows students to submit experiment data electronically during a lab—this compilation of data is known as data sharing. This project covers the design, implementation and evaluation of data sharing in WPI Physics labs.

The Davis Education Foundation, a public charitable foundation established in 1985 (Davis Educational Foundation), gives grants to regionally accredited undergraduate colleges and universities in order to improve their effectiveness at teaching (Davis Educational Foundation: FAQ). Founded by Elisabeth Davis and Stanton Davis, previous chairman of Shaw's Supermarkets, the David Educational Foundation has given over $75.8 million to over 139 institutions (Davis Educational Foundation). WPI received their grant in May of 2008.

Before data sharing, Physics labs at WPI were done mostly on paper, with hand-written results and data sheets. Labs would start with students pairing off into groups of two, followed by a roughly ten minute introduction from the Teacher's Assistant (TA), and then the students would turn to their workstations and complete their lab. During this time, the TA would help students who were having problems with equipment, having difficulty understanding the concept behind the lab, or having any other problems. After another forty minutes, results from these labs would have to be copied down by each partner, who would then return to their rooms, type their hand-written data into an online worksheet, and then turn it in on myWPI.

Lab time is a problem that data sharing might help. Fifty minutes to listen to the TA and learn how to use lab equipment and solve problems and perform an experiment and write down experiment data and distribute it to partners was cutting it close. Every second taken off of listening and off of performing tedious tasks gives one more second of comprehension for the lab concepts. With so little time, every second counts.

With data sharing in place, much of the difficulty of labs can be eased. Instead of hand-writing data, students submit it to one central location online, where they can access it from any WPI computer and avoid the hassle of copying the data for partners. And the data is right there, immediately usable in their homework. In addition to looking for students to help, TA's can watch a graph of submitted data, which allows them to efficiently identify students who are having problems with the lab. And they can identify them quickly and accurately, instead of waiting for the student to realize he or she has a problem, and then raise his or her hand. Lab time should be reduced significantly by these benefits.

A significant advantage of data sharing is giving students the ability to view the data of their classmates during each step of the lab. This does more than just warn the TA of failures; it also gives confidence to the students, since they know they are doing the experiment correctly. Instead of wondering if they have erred, students can check if their data is close to the average, and then continue forward to the next step without worrying if they are performing the lab correctly. More time is spent looking forward, and less time is spent looking back.

Having all data in one location, locked from editing at the end of the lab, also gives TA's a way to verify lab data when student reports are submitted. Students cannot copy data from each other since a record of their results is saved and viewable by faculty, which forces them to perform

calculations and understand the lab on their own. And it is a simple task for a TA to sort data and look for duplicate data at the end of a lab; cheating becomes very difficult with data sharing in place.

These benefits stretch further than just the lab. Professors can display experiment data during lectures, allowing them to bridge the gap between lecture time and lab time. This helps reduce comprehension time of students, as they will be able to ask questions about the data and better understand the concepts which the data reveals. It may also help professors to gauge student understanding by allowing an exchange of thoughts on real, current lab data.

Data sharing helps tie together labs and lectures, helps students get through labs with understanding, and helps TA's in running labs and grading reports. A central, secure archive of experiment data combined with the ability to view and format this data helps everyone involved.

Deciding on an Implementation

Deciding on an implementation uncovered a few important decisions. Any project implemented must be maintained easily. This ruled out the option of creating a custom system, which would provide an exact solution to every requirement. Having a custom solution would incur future headaches once the technology on which it was built became obsolete, or once the designers left and the requirements changed. Ease of use required that the system be familiar to both students and professors, which ties in with the requirement for security. There had to be as little setup as possible, and as much redundancy as possible. Less to teach meant less to learn.

Two solutions were seriously considered: a custom made system, and Microsoft SharePoint. The custom system would have performed all of the necessary tasks, but it failed in maintainability

and security. It would have been too much of a problem to keep up with changing technology and keep faculty trained on how to fix and upgrade. These limitations ended up Microsoft SharePoint.

Microsoft SharePoint is an enterprise content management system which is already widely in use at WPI. It is tightly coupled with Windows permissions, which WPI uses on all computers, easing the problem of integrating with WPI security; it is already set up with a sub-site for the Physics department, relieving much of the pain of setup; it allows an internal folder structure with specific permissions, making complex permissions possible; and it allows submission of form data into a file in SharePoint's folder structure, which allows student submissions of data. These capabilities, with some setup, provide a stable system on which to build data sharing for the Physics department.

Implementation

SharePoint is a Document Management System which links in closely with the Microsoft Office product line. This has a few significant advantages for this project, being that SharePoint has the innate ability to store spreadsheets, and Microsoft Excel has the ability to read, manipulate, and chart spreadsheets. This functionality is out of the box, and is exactly what the project requires.

SharePoint must be made do three tasks to meet the requirements for this project. It must collect data and store it real-time; it must format and that data real-time and be able to display it on a projector; and it must give students access to their data, and faculty access to all data. These three capabilities fulfill the most basic requirements for data sharing.

Collecting data in SharePoint uses SharePoint's most fundamental concept: lists. Lists are building blocks in SharePoint—data lists, document lists, page lists, task lists, etc. This project makes extensive use of SharePoint's custom list, which provides many useful and relevant features out of the box.

Custom lists are custom in that they store data according to a set of columns. Each of these columns is created by the user with a set of restrictions on the type of data that can be stored in it. For instance, a custom list could have a column titled "First Name" that can only contain a sequence of characters or a sentence; it could also have a column titled "Age [Years]" that can only contain numbers greater than 0. It might also have a column titled "Birthday" that can only contain dates and times. Once the columns are set up, the list can store rows of data with an entry for each column in each row.

Upon list creation, SharePoint automatically makes a few default templates for interacting with it. It creates a form for adding a row of data, a form for editing a row of data, a grid form for editing bulk data, a table for viewing data, and a properties view for inspecting and changing list properties, and for deleting the list. These default views provide basic functionality for interacting with custom lists: create, read, update and delete.

This seems to satisfy the requirement for collecting data, but it still needs to tie into the lab write-ups so that students can find the data submission forms. Lab write-ups are currently in a Wiki, so a natural method of accessing the data forms would be to simply put links in the write-up that navigate the student to them. This works, but it also has the downside of leading the student to a different page, which requires that the student reorient to understand what the new page is, and then find his or her way back to the lab write-up and find his or her place on the page. This takes time and adds confusion. Better would be to embed the data submission forms directly into the page, which can be accomplished using SharePoint Web Part pages.

Web Part pages are simply web pages with columns that can contain blocks of custom templates, called Web Parts. There are a number of Web Parts included in SharePoint, including default form views, list views, permission views, filters, editable content blocks, and so on. In order to embed a form inside of a lab write-up, a combination of two default Web Parts can be used: the editable content area, and the custom list form parts. The editable content area is simply a block of text that can be edited. Instructions are placed inside of these content areas, and list forms are placed in-between content areas in order to embed the data submission forms within lab instructions.

Custom list forms display one of the views for a given custom list. One of the default views could be used, but the default data submission view contains a few extraneous fields, such as Attachments and an extra toolbar. Removing those fields requires creating a custom view based on the submission view, and then selecting that view in the form Web Part. Once that procedure is finished, the lab write-ups will have embedded data submission forms on a single page, which meets the first requirement for data submission.

Graphing the data can be done in a variety of ways, but in order to stay within the bounds of stock SharePoint functionality this project will use Microsoft Office and its built-in connection with SharePoint. Microsoft Office products are designed to be able to navigate lists within SharePoint, and to process list data. This is especially useful with Access and Excel, which are products built for tabular data manipulation, both of which are equipped with extensive graphing capabilities.

The difference, with respect to this project, is that Excel provides lower level control of data, which allows complicated data processing before graphing. Access, on the other hand, has a default project type which allows immediate simple graphs of data; it is simpler to use, but does not handle complexity with as much ease as Excel. Both of these graphing methods have potential use, depending on the lab data.

Filtering data before graphing is a necessity. Being able to limit data to a few sections, to a specific lab time, or to a specific student allows TA's to more quickly identify problems in a lab, students to more quickly identify their data points with respects to others', and any research being done on the data significantly easier.

Both applications have the ability to filter data before graphing. Access makes it especially simple by allowing a user to drag columns into a "Filter" box. Excel requires the creation of a PivotTable in order to filter data; this PivotTable maps the entire data down to a smaller table that can be linked to a graph. The latter works, and only needs to be set up once, but it requires a timely setup.

Both applications, however, can synchronize the data for graphs with a SharePoint server. Clicking a single button refreshes the data, which in turn refreshes the graph. This graph can be displayed on a projector for a class to see, which completes the requirement for live graphing of streamed data.

Lastly, security of data is solved by sending the data to students through their WPI e-mail accounts. This removes the need for any custom permissions giving students access to their data and their partner's data, which makes permissions setup simple: deny all students access to any data. And all of this can be done using stock SharePoint workflows.

Workflows are a set of custom rules and actions that are executed when an event occurs in SharePoint. In order to use a workflow for e-mailing upon data submission, the workflow is attached to data sharing list modification events - add item and edit item. When one of these actions occur, the workflow stores information about the student who submitted it and saves it into a compiled string. Once all data has been collected, a final form at the bottom of the page allows both students to enter their e-mails and submit the data. Once this is completed, the compiled string is stored in the body of the e-mail, using a workflow attached to the e-mail form, and sent off to the students.

The purpose of workflows in SharePoint is to allow common users to create complex

functionality without requiring custom code to be written. In this case, the condition for the

workflow actions to start is that the list must have an item added to it. Once the item is added,

then the e-mail action starts. This completes the last step required for data sharing submission.

Discussion

The current implementation of Data Sharing performs the intended tasks of the system: real time data entry, real time data processing, and secure data storage. This system supplies the needed function, but also requires that a few decisions be made during the creation of labs. These decisions change the complexity of the initial setup, of faculty use and maintenance, and of student interaction.

One major decision is whether or not to create the labs in the Wiki or in a Web Part page. The Wiki provides a few significant advantages: it's easy to use, easy to train to use, and has built in shortcuts to create new pages and link to other pages. This makes the setup simple, and the maintenance simple. Web Part pages have a slightly steeper learning curve, requiring the use of Microsoft SharePoint Designer, but provide the ability to embed data sharing forms within the page. Depending on the complexity and count of data submissions, one may be preferable over the other.

Embedding forms into the lab allows students to see the entire lab on one page. This lets them quickly assess what data will be submitted, and when it will be submitted. Having separate pages for data submission, which would result from using the Wiki, requires that students constantly navigate to and from different pages, which may be disorienting. At the very least, navigating between pages will slow down the lab, as reorienting after switching pages adds time. With only one or two submissions of simple data, the Wiki might be preferable. With any more than that, or with complex data, the Web Part pages will reduce comprehension time and speed up the lab.

The second major decision is whether to use Excel or Access to view data. This decision greatly affects the complexity and time required for lab setup and maintenance. Access has a simple interface that provides quick graphing capabilities. Excel requires that all data be calculated, arranged, sorted into PivotTables, and then graphed. The latter has a steep learning curve, requiring much time spent in help files.

The only situations in which Excel seems to be a requirement are those in which data must pass through some calculations before being graphed. For the purposes of comparing raw data measurements, this is not necessary. Raw data comparison provides the teaching related benefits of data sharing. While it may be useful to perform calculations on the data for analysis purposes, it does not seem to be as much of an issue for real-time display in the laboratory.

The last decision is whether or not to remove the extraneous fields from the data submission forms. There are two methods for doing this: graphically removing the fields from within SharePoint Designer, and manually removing the fields from the form's XSLT (Extensible Stylesheet Language Transformation). Removing fields from within the SharePoint designer is easy to learn, but occasionally resulted in corrupt lab files during the setup of a test lab. Most of the errors resulted from removing the "Title" field, which is a required field in SharePoint custom lists. Modifying the XSLT is a complicated process requiring significant study. It allows all fields to be completely removed or customized, but has the steepest learning curve of any skill required for this project. The effort is likely not worth the benefits, but still remains as a viable option.

One of the labs in PH1121 was redone - the RC discharge lab. The forms were formatted using XSL and the graphs were created in Microsoft Access. This process took about an hour, and no major holes in methodology were discovered during this process.

This lab was tested later on by faculty, who acted as students and submitted data during a mock-lab. During this test, all of the data was graphed on the projector in real-time. At the end, every faculty member received an e-mail with their data. The system performed its intended tasks without err, and the instruction given to faculty on how to use it was minimal.

The only issues found during the lab were esoteric or cosmetic. In one case, two faculty members were logged into the same account, and therefore ended up overwriting their own data. This is a rare case, and should never happen in a real lab. Second, the data submission forms had submit buttons on the bottom and top of the form. This was confusing and might result in submitting the wrong data, or at least slow down comprehension of the form.

Overall, the first test was a success. The system performed exactly as intended, and no major flaws were uncovered. It is ready for a real world test.

Conclusion

WPI received a grant from the David Foundation for improving their Physics labs. This project's goal was to formulate, implement, test and analyze a data sharing system for use in the Physics labs.

Both students and faculty benefit from data sharing. Students can view their data with respects to their peers' data in real-time, allowing them to quickly identify measurement errors or misunderstanding of the concept. It also gives teacher's assistants the same benefits, giving them a way to find students who need help or are who are having difficulties. Professors also benefit from being able to present laboratory data to students during class time, which in turn may help the students better understand the concepts being taught. Lab data from consecutive terms may also help the lab manager identify broken or inaccurate equipment, which helps reduce in-lab problems. Overall it saves everyone time which them focus more on Physics.

The final data sharing system used a combination of Microsoft SharePoint and either Microsoft Excel or Microsoft Access. Avoiding custom code and extensions and sticking to out-of-the-box functionality helps assure that maintenance will be easy, supported by Microsoft, and requires no changes to the existing network from outside of the Physics department. This helps ensure that the system continues to run smoothly.

Internally, the system stores data in SharePoint custom lists. This is a fundamental concept in SharePoint systems, and should be fully understood by anyone modifying the system. Custom lists store the data, provide views for the data, and allow the structure of the data to be modified. This generic custom list is the reason data sharing is possible in SharePoint.

Office, Access and SharePoint designer have built in capability to read custom lists. This is the fundamental concept behind graphing the stored data. Because SharePoint and Office are so closely linked, all of the graphing and calculation features in Office and Access are available for SharePoint data. Graphing the remotely stored SharePoint data works without modification and provides the graphing capability for this project.

Students receive all of this data through e-mail, which is sent from a SharePoint workflow. E-mailing data to students, rather than giving them access to it on SharePoint, guarantees that students will only have access to their own data. This reduces the complexity in setting up permissions on submitted data, making setup and interaction easier.

This system is still untested during school laboratories, but has been through a simulated test with faculty acting as students. Everything went smoothly, from data submission to graphing to retrieving data after the lab. While this does not guarantee that no problems will arise, it does signify that a major milestone has been reached; the system has passed the smoke test and is ready to be tested in a real-world environment.

The system functions as is, but the complexity of setting up a lab is unnecessarily difficult. This is mainly due to restrictions from SharePoint; and worse, it is limited by the inability to install third party extensions. Using a custom made data sharing system would still provide a significant improvement in ease-of-use, as would third party SharePoint extensions. For future projects based on this system, third party extensions would likely have the greatest benefit to cost ratio.

Graphing is the main area in which the system might be improved. With third party extensions, graphs could be done entirely within the browser, even inline in the lab write-up. This would

completely cut out Access and Excel, reducing system complexity and allowing students to save graphs for use in their reports. There are plenty of free graphing libraries for SharePoint that provide this functionality. The difficult work would be in getting the permission to add plugins to the SharePoint server.

The final system provides a base future data-sharing. It is flexible enough for sharing data in both simple and complex labs, and provides various means of graphing and processing the shared data. After testing is completed, this system can serve as a base for other projects related to data sharing - for improvements, for extensions, and for use in other departments.

Works Cited

"Davis Educational Foundation: FAQ." *The Davis Foundations*. Web. 26 Oct. 2009.

        <http://www.davisfoundations.org/site/faq_educational.asp>.

"Davis Educational Foundation." *The Davis Foundations*. Web. 26 Oct. 2009.

        <http://www.davisfoundations.org/site/educational.asp>.

Appendix A – How to create a data-sharing lab

Creating a lab involves completing five major steps: creating lists, then graphs, then the page for lab instructions, then adding forms to the lab instructions, and then setting up the automatic e-mail.

1 – Create Data Submission Lists

1. Navigate to the Data Collection tab in the Physics department's SharePoint site
    (https://student.SharePoint.wpi.edu/courses/physicslabs/datacollection/default.aspx)

2. Click the "Site Actions" menu in the top right corner of the page.
    A menu will drop down.
    Click "Create."

3. A page will open describing the different types of content to create.
    Select "Custom List" at the top of the fourth column to the right.

4. This new page contains the form for entering the initial custom list information.
    Fill out the information:

    a. In the "Title" field, enter the course number, followed by the lab title, followed by the submission title.

        Example
        PH1121 – Determining Resistance – Part 1

    b. Select "No" in the field titled "Display this list on the Quick Launch?"

    c. Click the "Create" button at the bottom of the form

5. The default view for the custom list will open. At the top of the list are three button menus: New, Actions, and Settings.
    Click the "Settings" menu, and then, in the menu that drops, click "List Settings."

6. The list settings view opens. From this page, two properties need to be changed: the permissions, so that the students cannot see each other's data, and the required columns, so that when the forms are combined into one page they can be submitted sequentially.

    a. Setting the permissions
        i. Click "Advanced Settings" under the "General Settings" column.

ii. A page opens containing advanced settings.
Under "Read access:" select "Only their own."
Under "Edit access:" select "None."
Under "Attachments" select "Disabled."
Click "OK."

iii. A message box opens warning that you will delete all attachments on the current list. You do not care.
Click "OK."

b. Disable required columns
    i. Under the "Columns" section, click the "Title" row.
    ii. A page opens containing the column settings for the "Title" column.
    Under "Require that this column contains information:" click "No."
    iii. Click "OK."

7. At the top of the page, click the title of the custom list. This will navigate back to the default view of the list. All that is list is to add all the necessary columns to the list.

    a. Click the "Settings" menu again, but now click "Create Column" in the menu that drops down.

    b. A new page opens containing the initial column settings for the new column.
    Under "Title" type "Station Number."
    Under "The type of information in this column is:" select "Lookup (information already on this site)"
    Under "Get information from:" select "Station Numbers – OH###." (### is the room number of the lab)
    Click "Create."

8. Repeat step seven for every column of data required for the first part.

9. Repeat up to step 7 for the final e-mail list. This list will perform the task of collecting lab-specific submission information and will provide an event to which the automated e-mail Workflow can attach.
Create the columns: Term, Section, and E-Mail To.

2 – Create Data Graphs

1. At the top right corner of the list is a menu titled "View: All Items."
Click this menu, and then select "Create View."

2. A page opens describing the types of views to create.
   Click "Access View."

3. Microsoft Office Access opens with a save dialog.
   Under the "File name:" field, delete the text "Views for" at the beginning of the file name

   Example
   PH1121 – Determining Resistance – Part 1.accdb

   Click "Save."
   SharePoint will ask you to Login. You don't care to login, as it will default to the currently logged in Windows user, so click "Cancel."

4. A new dialog will open allowing you to select a type of view.
   Select the "PivotChart" type.
   Click "OK."

5. On the right side of the new PivotChart is a list of fields.
   Drag the "Section Number" field into the box labeled "Drop Category Fields Here" at the bottom of the page.
   Drag relevant data fields into the box labeled "Drop Data Fields Here" above the graph.
   Save.

   i. Click the text labeled "Axis Title" on the left side of the graph.
   ii. Click the "Design" tab at the top of Access.
   iii. Click the "Property Sheet" button on the top right side of the toolbar. A dialog opens.
   iv. Click the "Format" tab in this dialog.
   v. Enter the vertical axis title in the "Caption" field.
      Press "Enter."
   vi. Click the horizontal "Axis Title" text on the graph.
   vii. Enter the horizontal axis title in the "Caption" field.
      Press "Enter."
   viii. Click "Publish to SharePoint" above the top of the graph.
      Select a relevant folder for the view to be saved.
      Click "Publish."
   ix. Accept all resulting dialogs.

   Close Microsoft Access.

3 – Create the Web Part Page

1. Click the "Site Actions" menu in the top right corner of the page.
   A menu will drop down.
   Click "Create."

2. A page will open describing the different types of content to create.
   Select "Web Part Page" in the farthest column to the right.

3. A new page opens containing the initial Web Part Page information.
   Type the lab name in the "Name:" field.
   Select "Full Page, Vertical" under the "Choose a Layout Template:" field.
   Click "Create."

4. Copy the entire URL from the navigation bar at the top of Internet Explorer.

5. In Windows, click the "Start" button.

6. Type "SharePoint" in the "Start Search" box.
   Select "Microsoft Office SharePoint Designer 2007" under the "Programs" section of the "Start" menu.

7. SharePoint Designer will open.

   a. It will ask you to login.
      Click "Cancel."

   b. Open the "File" menu and click "Open Site…"

   c. A dialog opens.
      Paste copied URL from step 3.4 into the "Site name:" field.
      Remove the filename from the URL.

      Example
      https://student.SharePoint.wpi.edu/courses/physicslabs/datacollection/Data_Documents/

      Click "Open."

   d. The site opens. On the left is a file browser toolbox. In the middle is the content view.

   e. In the file browser, navigate to the Web Part Page created in step 3.2 and double click it.

The Web Part Page will open in the content view.

Do not close SharePoint Designer.

4 – Add Data Submission forms to Web Part Page

This section assumes that you are working in SharePoint Designer and are viewing the Web Part Page for the current lab. If you are adding forms to an already existing Web Part Page rather than performing this step immediately after Part 3, then navigate to the Web Part Page in a web browser, and then complete steps 3.4 – 3.7.

1. Click the "Click to insert a Web Part" text in the middle of the page.

2. Open the "Insert" menu, then the "SharePoint Controls" menu, and then click the "Custom List Form…" option.

3. A dialog opens.
   Under "List or document library to use for this form:" select the data submission list created for the first part of the lab.
   Uncheck "Show standard toolbar."
   Click "OK."

4. The custom list form for the first data submission opens.
   Remove the unnecessary buttons.

   a. Click the "OK" button at the top of the form.
      Press the "Delete" key.

   b. Repeat for both "Cancel" buttons on the form.

5. Click the title of the data submission form. It should select the form (a gray box covers the form and a small tab appears at the top).
   Press the right arrow key on the keyboard.

6. Repeat steps 2 through 5 for each part of the lab, and lastly for the e-mail form.

7. Save the document.

8. Return to the browser.
   Press "F5" to refresh the page.

9. The data forms should appear on the page. At the top of them a wide orange button titled "Add a Web Part" appears.
   Click "Add a Web Part."

10. The "Add a Web Part" dialog appears.
    Scroll down to the "Miscellaneous" section and check the "Content Editor Web Part" item.
    Click "Add."

11. The "Content Editor Web Part" is created at the top of the page. Inside of it is text explaining how to change its content.
    Click the highlighted text: "open the tool pane."

12. The tool pane opens on the right side of the document.
    Click the "Rich Text Editor…" button.

13. A content editor dialog appears.
    Type the lab instructions into the content of the dialog.
    Click "OK."
    At the bottom of the tool pane, click "OK."

5 – Create Workflow to Send E-Mail

1. In SharePoint Designer, open the "File" menu and click "New."

2. The "New" dialog opens.
   Select the "SharePoint Content" tab.
   Select the "Workflow" item in the left column.
   Select "Blank Workflow" in the center column.
   Click "OK."

3. The "Workflow Designer" dialog opens.
   Type an appropriate name for the Workflow Designer, such as the course number and lab title.

   Example
   PH1121 – Determining Resistance

   Uncheck "Allow this workflow to be manually started from an item."
   Check "Automatically start this workflow when a new item is created."
   Click the "Next" button at the bottom of the dialog.

4. The Workflow setup step appears.
   In the "Specify details for 'Step 1'" section, click the "Actions ↓" button.

In the dropdown menu, click "More Actions…"

5. The Workflow Actions dialog appears.
   Select "Build Dynamic String" from the select box in the middle of the dialog.
   Click "Add."

6. Some text appears next to the "Actions ↓" button.
   Click the highlighted text, "dynamic string."

7. The "String Builder" dialog opens. Inside of this dialog you will create the body of the e-mail sent to students.
   Type the title of the document.

8. Add the lab-specific fields:

   a. Press the "enter" key and then type "Term: ".

   b. Press the "Add Lookup" button in the bottom left corner of the "String Builder" dialog.

   c. In the "Field:" dropdown, select the "Term" item.

   d. Click "OK."

   e. Repeat steps 8.a – 8.d for the "Section" and "E-Mail To" fields.

9. Add the data fields:

   a. Type the first data column title followed by a ":".

   b. Click the "Add Lookup" button in the bottom left corner of the "String Builder" dialog.

   c. In the "Source:" dropdown, select the list from which to retrieve the first data column value.

   d. In the "Field:" dropdown in the "Lookup Details" section, select the column from which to pull the value.

   e. In the "Field:" dropdown in the "Find the List Item" section, select the "… - Created By" item.

   f. Next to the "Value:" field, click the "…" button.

    g.   The "Select Users" dialog appears.

In the "Or select from existing Users and Groups:" field, select "User who created the current item."

Click the "Add >>" button.

Click "OK."

Click "OK."

Click "Yes," because you understand that it will pull data from the first matched item in the selected list.

    h.   Repeat steps 9.a – 9.g for each data value measured by the student.

    i.   Click "OK" to exit the "String Builder" dialog.

10. Create the "body" variable:

    a.   Click the text "Variable: variable."

    b.   Click "create a new variable…" in the menu that appears.

    c.   Type "body" in the "Name:" field.

    d.   Click "OK."

11. Create the e-mail to list:

    a.   Click the "Actions ↓" button.

    b.   Click "Build Dynamic String" in the menu that appears.

    c.   Click "dynamic string" in the text that appears.

    d.   Click the "Add Lookup" button in the "String Builder" dialog that appears.

    e.   Under the "Field:" field, select the "E-Mail To" item.

    f.   Click "OK" twice.

12. Create the "emails" variable:

    a.   Click the text "Variable: variable1."

    b.   Click "create a new variable…" in the menu that appears.

    c.   Type "emails" in the "Name:" field.

    d.   Click "OK."

13. Create the "Send E-Mail" action:

    a.   Click the "Actions ↓" button.

    b.   Click "Send an Email" in the menu that appears.

    c.   Click the "this message" text that appears.

    d.   The "Define E-mail message" dialog appears.

         i.   Next to the "To:" field, click the button with a book icon.
        ii.   The "Select Users" dialog appears.
            Select "Workflow Lookup…"
            Click the "Add >>" button.
        iii.   The "Define Workflow Lookup" dialog appears.
            Select "Workflow Data" from the "Source:" dropdown.
            Select "Variable: emails" from the "Field:" dropdown.
        iv.   Click "OK" twice.
        v.   Type an appropriate e-mail subject.
        vi.   Click the "Add Lookup to Body" in the bottom left of the "Define E-mail message" dialog.
        vii.   The "Define Workflow Lookup" dialog appears.
            Select "Workflow Data" from the "Source:" dropdown.
            Select "Variable: body" from the "Field:" dropdown.
        viii.   Click "OK" twice.

14. Click "Finish."