Trevor McEwen

MBA 614 Final Project

Executive Summary

My project was not done for a business, but to help me in my calling as a Sunday school teacher in my ward. I have noticed that when the students read the lesson materials before coming to class then there is much better discussion and learning that takes place. I think that in general everyone wants to read, but with our busy lives we sometimes forget. I have tried to send out reading reminders at the beginning of each week, but I also tend to forget to do that and when I do remember it takes time to actually find the links and write the emails.

My project seeks to use user forms to make sending a digital reading reminder quick and easy. In addition to sending reminders for Sunday school I built in the ability to also send a reminder about the Priesthood or Relief Society lesson that will be taught that week. The system requires a one-time setup to input student information (name, email address, phone number, classes attending, and contact method preference) and information about the ward and teaching material (annual Sunday school topic, when the ward has stake conference, etc.). After that information is entered into the system then the user only needs to open the spreadsheet, take a quick look to make sure the information looks accurate (it will update automatically based on the date) and press the send button. There is also an ability to adjust the message that is sent out each week (so you can wish everyone a Merry Christmas in the same email).

Implementation Documentation

There are three main parts of my project that I built:

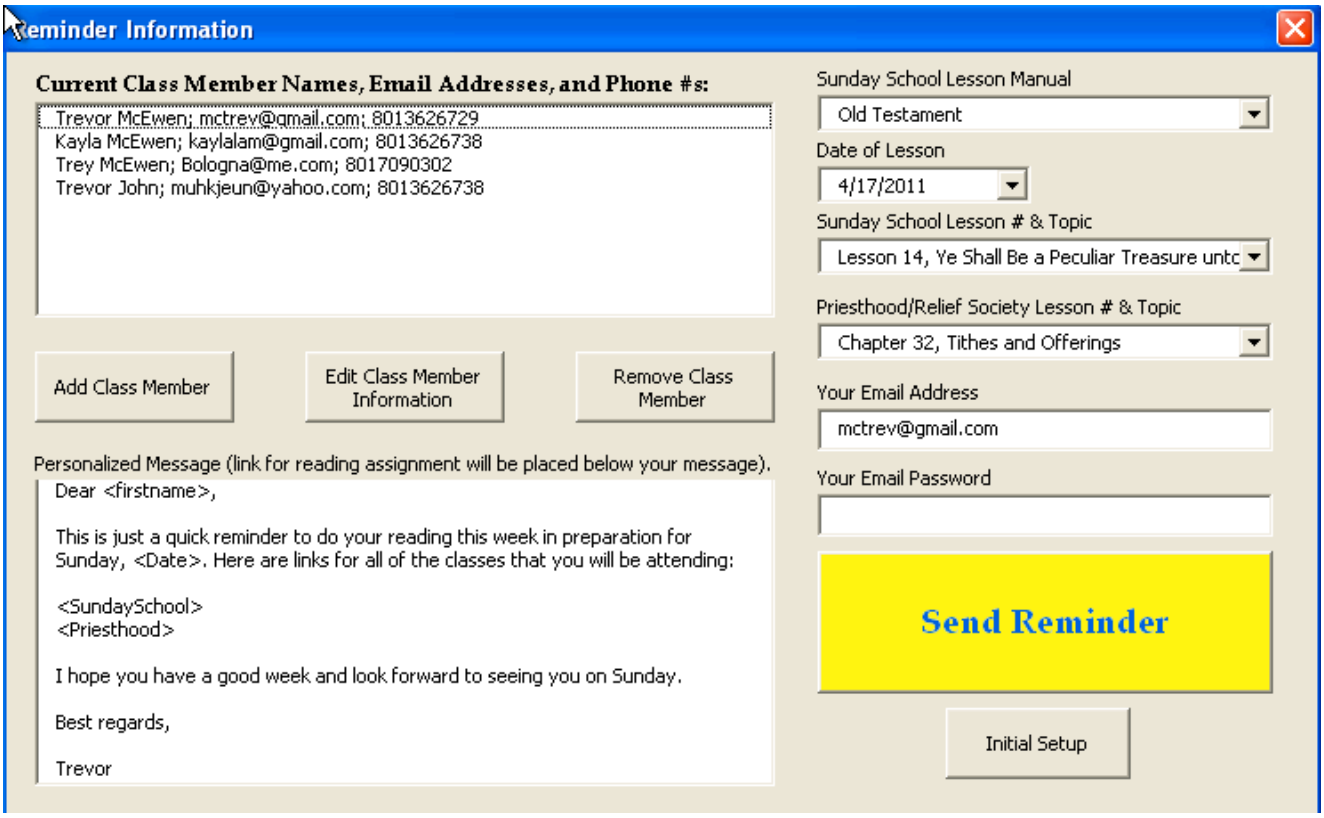
1. Forms – The forms were used to collect meaningful data to determine class schedules. Here is a list of information that I gathered through the forms and why it was meaningful:
    a. Conference Dates – Since every ward will be unique as to when they will not have the regular block schedule it was important to request information from the user about when their stake conferences will be held. Some wards also choose to not have regular meetings on their ward conference day, so this form allows users to add as many “skip dates” as they need. The “skip dates” can also be removed if needed. Once entered the “skip dates” are recorded to the workbook so that the user doesn’t need to enter these each time.
    b. Manuals – There is a combo box on the form that allows the user to select the Sunday school manual from a list of the regular classes (Book of Mormon, New Testament, Doctrine and Covenants, and Old Testament). This information is also stored to the workbook to be used later when the specific lesson information is pulled from the internet.

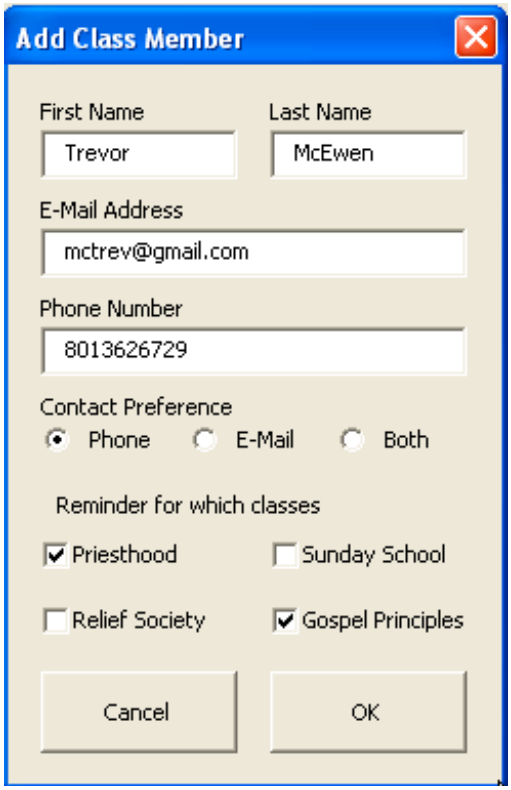
c. Class Members – A separate form was created to gather information about individual members of the ward. The form includes fields for their first and last name, email address, phone number, preferred contact method, and which classes they are members of. The cell phone was included for possible functionality of being able to choose whether to send them an email reminder or a text message reminder. For now the functionality is only for email. There are check boxes that will indicate whether or not they are a member of 4 classes (Priesthood, Relief Society, Sunday School, and Gospel Principles). The idea is that each member can receive one email that has links to all the reading they need to do for that week. All of this information will be recorded to the workbook so that the user only needs to enter it once.
d. Personalized Message – This is the message that will be sent out via email. There are markers for member's first names and the classes that the member will be attending so that each email sent out can be personalized to an individual member. This message is also stored to the workbook, but can easily be edited within the user form if needed.
e. User Information – There are fields where the user can input their email address and password for the email that they would like the messages to be sent from. The email address is stored to the workbook for convenience sake, but the password needs to be entered each time. In future iterations it would be easy to make a check box so the user could decide whether or not to save their password in the form.

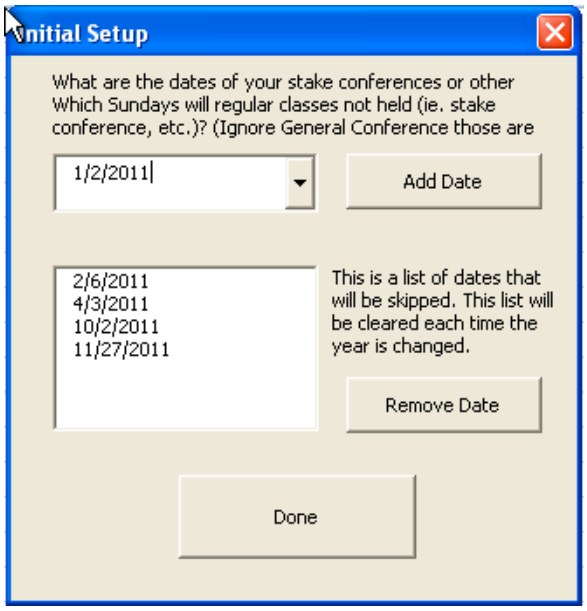
Once all the information is entered into the form there are other fields that provide feedback to the user. Here is a list of these fields and their functionality:

a. Date of Lesson – This is a combo box that shows the date of the upcoming Sunday. This is automatically computed in the VBA code so the user does not need to adjust it, but does have the capability of changing the date in case the user would like to send a reminder 2 or more weeks before class.
b. Sunday School Lesson and Topic – This is a combo box that shows the Sunday School lesson for this coming week. This is also automatically calculated based on the user's prior inputs. Just as the lesson date, the user has the ability to choose another lesson if they desire.
c. Priesthood/Relief Society Lesson & Topic – This is the same as the Sunday School Lesson and Topic except it is for the Priesthood and Relief Society Manual.

On the next page there is a screenshot of the forms.


Reminder Information
Current Class Member Names, Email Addresses, and Phone #s:
Trevor McEwen; mctrev@gmail.com; 8013626729
Kayla McEwen; kaylalam@gmail.com; 8013626738
Trey McEwen; Bologna@me.com; 8017090302
Trevor John; muhkjeun@yahoo.com; 8013626738
Add Class Member
Edit Class Member Information
Remove Class Member
Personalized Message (link for reading assignment will be placed below your message).
Dear <firstname>,
This is just a quick reminder to do your reading this week in preparation for Sunday, <Date>. Here are links for all of the classes that you will be attending:
<SundaySchool>
<Priesthood>
I hope you have a good week and look forward to seeing you on Sunday.
Best regards,
Trevor
Sunday School Lesson Manual
Old Testament
Date of Lesson
4/17/2011
Sunday School Lesson # & Topic
Lesson 14, Ye Shall Be a Peculiar Treasure unto
Priesthood/Relief Society Lesson # & Topic
Chapter 32, Tithes and Offerings
Your Email Address
mctrev@gmail.com
Your Email Password
Send Reminder
Initial Setup


Add Class Member
First Name
Trevor
Last Name
McEwen
E-Mail Address
mctrev@gmail.com
Phone Number
8013626729
Contact Preference
Phone
E-Mail
Both
Reminder for which classes
Priesthood
Sunday School
Relief Society
Gospel Principles
Cancel
OK


Initial Setup
What are the dates of your stake conferences or other Which Sundays will regular classes not held (ie. stake conference, etc.)? (Ignore General Conference those are
1/2/2011
Add Date
2/6/2011
4/3/2011
10/2/2011
11/27/2011
This is a list of dates that will be skipped. This list will be cleared each time the year is changed.
Remove Date
Done

2. Web Scraping – Originally I was going to use this tool to pull data (like member contact information, stake conference dates, and lesson schedule information) directly from the lds.org website. The more I talked to people in my ward, the more I found out that the information they had listed online was usually outdated or incorrect. So in the end I used the web scraping tool to capture the list of lessons for each Sunday school manual and the Priesthood/Relief Society manual. The way I have it set currently it will pull this information down from the web each time the manual combo box is changed. Although connecting to the internet and downloading the list of lessons and titles causes quite a long delay, I felt that this method was the most dynamic and showcased the web scraping ability. After pulling this information down from the web and with the information entered into the user forms the calculations can be made to show what lesson is supposed to be taught on which date.
3. Sending Emails – After compiling and combining information from the user forms and lds.org then the user just has to push the big yellow button to send out individual emails to each member in the list. Since the Priesthood and Relief Society manuals are only used on the second and third Sundays there is likely to be empty spots in the lesson combo box. If the send email button is pushed while the combo box is empty then there will be a prompt asking the user to enter a URL for the lesson that will be taught that week (sometimes it is a conference talk and other times it may be from the scriptures).

Discussion of Learning and Conceptual Difficulties

Of the three main parts, working with the user forms was by far the most complex part of the project. This section of the report will go into detail about the difficulties encountered while programming the project.

1. The most complex problem to solve was how to account for the days that there would either not be any lessons taught (because of General or Stake Conference) or when there will be a special presidency or bishopric lesson taught (as is the case for the 1st, 4th, and 5th Sundays each month). The difficulty stems from making the program dynamic enough to work for any year and still be able to know which months have 5 Sundays. The following code was used to recognize on what day a particular Sunday fell on:

```
If Weekday(skipWeek) > 1 Then
   skipWeek = skipWeek + Application.WorksheetFunction.Choose(Application.WorksheetFunction. _
      Weekday(DateSerial(Lessons.Cells(1, 1).Value, theMonth, 1)), 7, 6, 5, 4, 3, 2, 1) + (weekNo - 1) * 7
Else
   skipWeek = skipWeek + (weekNo - 1) * 7
End If
```

The code will check to see if the first day of any month is a Sunday. If the first day of the month is a Sunday then it will be recorded, but if it isn't a Sunday then the correct number of days will be added to the date to calculate the first Sunday. In this way it is possible to pinpoint any Sunday (1st, 2nd, 3rd, etc.) in any month.

2. After determining the dates of each Sunday of each month the next problem was figuring out how to allocate the lessons for each class accordingly. The method that I ended up using was first populating a worksheet with the dates of each Sunday in one column and the lesson for Sunday School and Priesthood/Relief Society in the next two columns. After populating without skipping any dates the program will go through and insert an empty cell on the weeks that the normal lesson manuals will not be used. The difficulty with this method is that the empty cells need to be inserted in order from the top down otherwise the resulting lesson schedule will be incorrect. This also means that the list containing the conference dates needs to be sorted before inserting those dates as well. This also required some ingenuity because the conference dates were stored in a worksheet in a row rather than a column (I used a recorded macro to figure out how to sort a row).
3. In addition to the two points above it was generally difficult to work with multiple lists especially when dates are involved. If I was to try this again I probably would try to use arrays rather than storing everything in a worksheet and recalling it later. The way I overcame the list difficulty was by liberally using "Do Until" loops.
4. The final and probably most important thing that I learned is how important it is to plan out a structure before actually beginning to code. When I started the project I had a general idea of what I wanted to do, but I hadn't mapped out on paper how I was going to accomplish it. Because I didn't plan well there were many small problems that I encountered that I had to "patch up" rather than solve completely. As a result my code is not very eloquent and probably takes much longer to run then it would have had I carefully mapped out the program on paper beforehand. It is basically like using duct tape to fix your car (no matter what the problem); you can get by, but it looks horrible and will probably lead to future problems where you need to use more duct tape.

Additional Elements for Future Iterations

Especially as I was coding there were multiple thoughts that came to mind of how I could further improve the program. Many of these additions would have been great, but would have needed a significant amount of time to complete. Here is a short list of some of these:

1. Ability to send text messages. I was hoping to be able to add functionality to send the reminder in an email, a text message, or both. This wouldn't have been too difficult to implement, but the problem is collecting information about which carrier each person uses would have started to get tedious. Also, it would have made for a long text message.
2. I wanted to add the functionality of being able to enter the unique lesson topics (1^{st}, 4^{th}, and 5^{th} Sundays) in advance rather than only adding the URL right as the message was sent. The reason I couldn't do this was how I had structured the lesson list to repopulate each time the workbook was opened. The user could then input the future unique lessons, but they would be written over the next time the workbook was opened.

3. I would have liked to have more interaction with the web. For example it would have been nice to download all of the names and email addresses from the online ward directory and automatically add them into my workbook. It also would have been nice to have more interaction with the online calendar function on lds.org to possibly make a unique calendar that would also be updated when the reminders were sent out. I think I would have been able to do this, but I just did not have enough time (I had already spent 30+ hours on the rest of the assignment).