

## Executive Summary

I currently work on campus as a research assistant for the BYU Writing Department. As an extension of the English department, our focus is to help teachers across all curriculums learn how to improve their students' writing, and how to be a better writing teacher. Our department hosts several luncheons and clinics that various professors across campus choose to participate in. Over the years, University Writing has composed a matrix of teacher names, department information, and email addresses of those teachers who have previously expressed interested in being a part of the University Writing curriculum. As a department, we send out multiple emails every week to different departments about different events that we are hosting. There are often errors produced in these emails with incorrect dates and times of events, or they are sent to the wrong people. I created a system that takes the list and cleans up the data, and then I also created a form that the department can use to send mass emails to people in different departments to ensure that all of the event information is entered correctly.

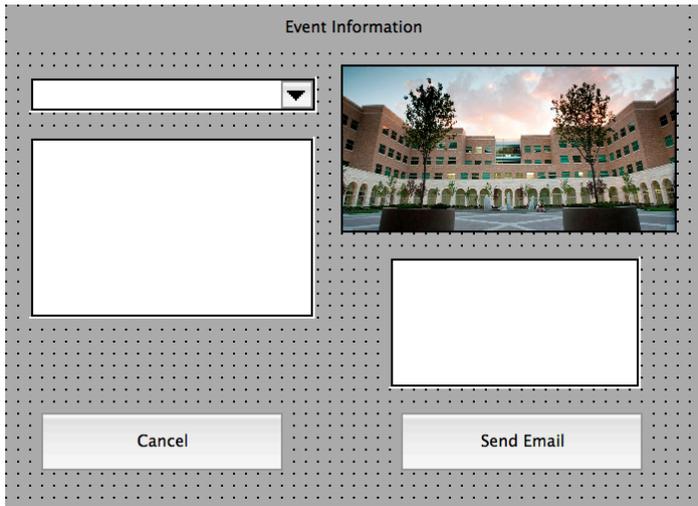
## Implementation Documentation

The first part of this assignment I wanted to clean up the data sheet that hosted the

```
Sub lastFirstName()  
|  
Range("A2").Select  
  
Do Until IsEmpty(ActiveCell)  
    ActiveCell = Trim(ActiveCell)  
    ActiveCell.Offset(1, 0).Select  
Loop  
  
Range("B2").Select  
Do Until IsEmpty(ActiveCell)  
    ActiveCell = Trim(ActiveCell)  
    ActiveCell.Offset(1, 0).Select  
Loop
```

information. I trimmed up the first and last names of the participants in case I wanted to directly input their names in the emails that I would be generating from the editor. I also trimmed up the emails so that they would be readable when I would automatically generate the code for the email.

After cleaning up the matrix information, I decided to create a user form that could be used by



a secretary or research assistant to accurately input the information, and it would then generate and send an email out to the people that were selected. We often target specific departments when we host activities, and need to send emails to those that participate, but sometimes the information entered is incorrect and can get confusing. The form that is generated, pulls information for the combobox and the listbox from

other worksheets that house all of the correct information and only need to be updated annually. The form has a combobox that allows someone to pick which event information they would like to submit, and a listbox which determines which departments to send the information to, and then also an optional additional message to be included in the body of the message. The VBA formula composes these drop down option menus, and then upon clicking "send email" will loop through the entire data set and match it with the department chosen, and if it matches, it will

send the email address to a string of emails that are attached by a semicolon. I wanted every email to also be send to

```
department = ListBox1.Value
Set s = Sheets("Master")
For x = 0 To Sheets("Master").UsedRange.Rows.Count
    If Cells(x + 1, 3).Value = department Then
        list = "writing@byu.edu" & "; " & Cells(x + 1, 4).Value
    End If
Next
```

**On Error GoTo cleanup**

```
Set OutMail = OutApp.CreateItem(0)
On Error Resume Next
With OutMail
    .To = list
    .Subject = CboBox.Value
    .body = body
    'You can add files also like this
    '.Attachments.Add ("C:\test.txt")
    .send
End With
On Error GoTo 0
Set OutMail = Nothing
```

the standard University Writing Email as well just to be sure that every email was sent correctly.

The next part was figuring out how to generate the email from the push button. I didn't have outlook on my laptop, so I did the

remainder of the programming on my work computer. To access Outlook, it did not require any password or protection, I just needed to launch the application. This code also allowed me to input my own body and subject how I would like it to appear on an email. For the sake of this project, I just kept everything extremely simple in my code, but it can all be easily manipulated for whatever purpose.

There were several other procedures that were used in this program, but these were the key procedures. The job to be done was to easily produce an accurate email that could be sent out to a mass list of people. This form makes that process easy when specific departments of people need to be emailed.

## Implementation Documentation

This project taught me a lot about the endless possibilities that are available within VBA. I thought VBA was fairly limited to data and numbers, but this project has shown me that there are so many different facets to use VBA programming in. I was actually extremely impressed while I was watching other student's presentations, and made me wish I had been more creative in my own project!

There are a few elements that I would have included if I had the patience and ability to do so. This code alone took me quite a while, as I have never coded anything before. It's like learning an entire new language! I would have liked to change the Send Mail option to make each email individual to the person's name. And then also create a response form for the respondents to be able to go in and RSVP their own information. I tried to add that today after talking to a classmate about how to do it, but after an hour of frustration, it just didn't seem worth it. I think that with more practice, I could have created a form that would house the information, but this much coding alone was a lot of work for me!

I also learned the important of debugging as you are going along. I would get so frustrated because the code wasn't working, and then all I had to do was sit down and go through the debug.print process to better understand where my errors were. It was so helpful! By simplifying the code and starting at a basic level, and then manipulating it to eventually get to the point that I wanted it to, finally resulted in the answers I wanted. The immediate windows and local windows saved the day.

The other thing I learned is that programming requires creativity. Problems need to be solved and there is usually more than one way to do it! I found that talking with someone about the process and brainstorming ways to get to a solution was the best way to figure out how to do

something. I visited the TA's many times, and they were extremely helpful in teaching me how to train my brain to think methodically. It was hard for me to put my thoughts into formulas to construct something that would work!

And lastly I learned that VBA programming on a mac is a nightmare. Don't do it.

## Assistance

I was able to get through most of the coding on my own, but because my laptop doesn't have Microsoft Outlook, Taylor Maughan helped me with the "send mail" portion because we sent my file over to him to debug and walk through the email process.