Executive summary

Lawn-O-Matic is a family owned and operated lawn care company that services the southern part of Utah County. Lawn-O-Matic has been operating for 10 years and has grown significantly in those 10 years. As one of the original founders of Lawn-O-Matic, I have been involved with every step of the process from bidding out the work to collection. Early on, we developed a spreadsheet to help us keep track of the work we do. However, this has always been a fairly manual process as we had to manually enter in every lawn we mow and record the details with that. To make things worse, we had no automated billing system so it was always very time consuming to produce bills on a monthly basis to our clients.

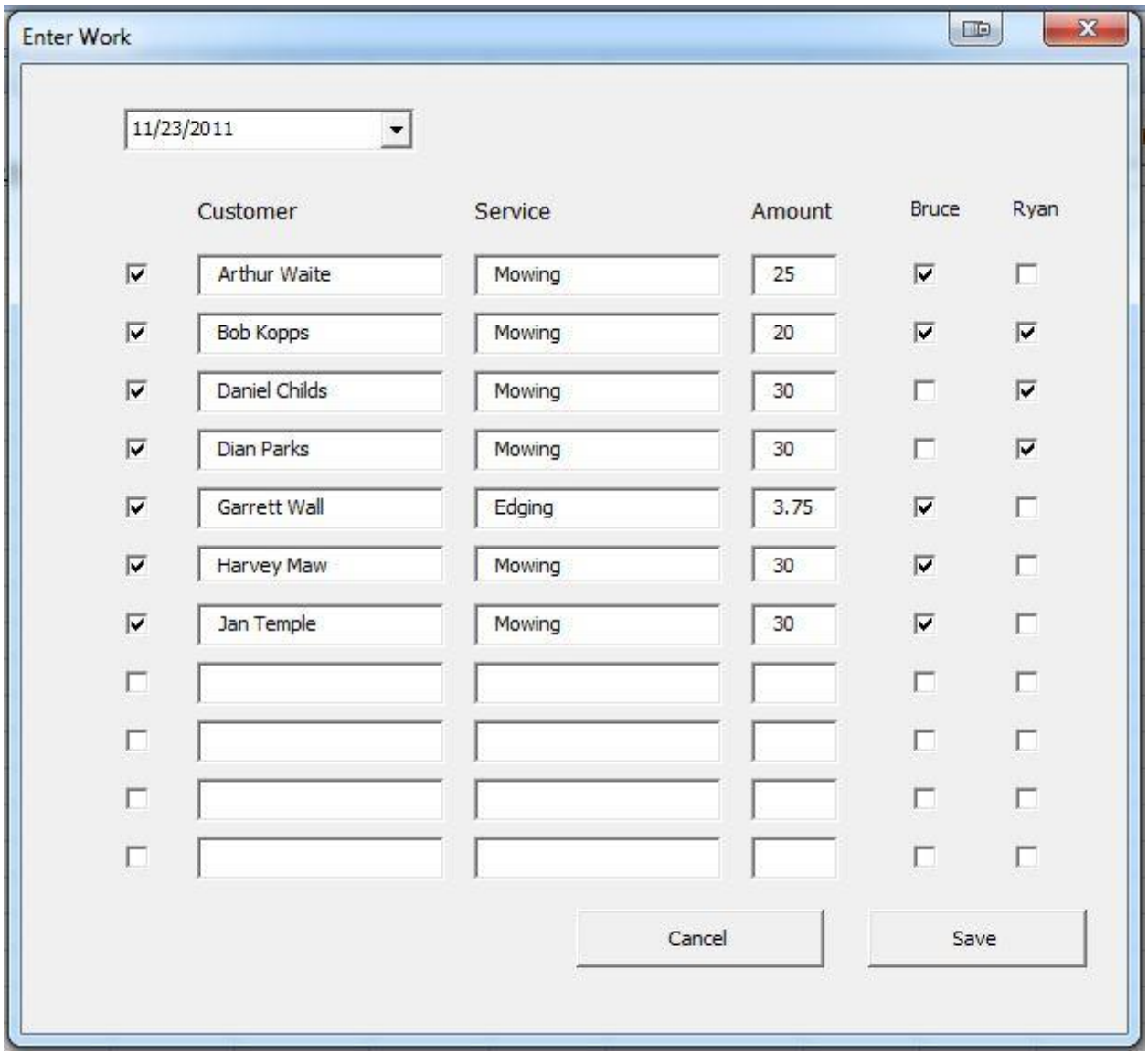
Two to three years ago, we began emailing our bills to our customers who provided their email address to us rather than mailing a physical copy of it. This decreased the cost of billing to us as we had to use fewer envelopes and stamps and it also made the process go faster since we could just email the bill rather than have to print it off and put it in the envelope. However, it was still labor intensive as we would manually create every bill for each customer.

In order to decrease the time needed to process billing, as well as to reduce the amount of errors in the bills, I have developed a spreadsheet that will automate a portion of the data entry process as well as automate the billing process and facilitate the movement of data between the different sheets on the spreadsheet.

Note: To help protect the identity of our customers, all names and contact information has been changed in the spreadsheet. This does not affect how the spreadsheet operates in any way.

Implementation

The first step was to automate the data entry process. This was done by creating the Enter Work user form. The Enter Work form appears when you click a button at the top of the "Work" sheet. It has a date time picker in the top left corner. The default value for this is the current date but you can change it if you are entering work from a different day. Based off the value that is picked for the date, the sub procedure converts the value into the name of the day of the week that the date represents. It then looks at the "Customers" tab and displays all customers who were supposed to have work done that day of the week. It enters the customer name, default service, and default amount into the user form text boxes. The user is then required to check who performed the work on the right side of the form. If a service other than the default service was performed, the user can edit the contents of the text box to reflect that or add an additional row at the bottom of the form.



When the user clicks the save button, the work indicated in the form is inserted onto the “Work” tab of the spreadsheet. The amount of the total allocated to each employee is calculated based on whether or not his name was checked in the Enter Work form.

WorkID	Date	Customer	Service	Amount	Bruce	Ryan	LOM	Check Number
	23-Nov-11	Arthur Waite	Mowing	$ 25.00	$ 16.67	$ -	$ 8.33	
	23-Nov-11	Bob Kopps	Mowing	$ 20.00	$ 6.67	$ 6.67	$ 6.67	
	23-Nov-11	Daniel Childs	Mowing	$ 30.00	$ -	$ 20.00	$ 10.00	
	23-Nov-11	Dian Parks	Mowing	$ 30.00	$ -	$ 20.00	$ 10.00	
	23-Nov-11	Garrett Wall	Edging	$ 3.75	$ 2.50	$ -	$ 1.25	
	23-Nov-11	Harvey Maw	Mowing	$ 30.00	$ 20.00	$ -	$ 10.00	
	23-Nov-11	Jan Temple	Mowing	$ 30.00	$ 20.00	$ -	$ 10.00	

This will be done routinely throughout the month. At the end of the month, the user will do the billing process.

Billing

When doing bills, the user will have the option of processing all bills or of processing one bill for a single customer. When the user clicks the button to process one bill for a single customer, an input box appears asking the user to enter the name of the customer.

When the user enters in the name, the sub procedure will look in the “customers” tab for the name that was entered in. If the name was not found either because they aren’t a customer or because of a misspelling, the program displays a message box indicating that no customer was found and reopens the input box for the user to enter in a different name. If a name is found, the program then checks to make sure the customer is an active customer. If the customer is an active customer the sub procedure creates a bill for that customer as seen below.

Lawn-O-Matic

Invoice

Professional Lawncare Service
975 E 800 N
Mapleton, UT 84664

Bill To
Bob Kopps
975 E. 800 N.
Mapleton, UT 84664

Invoice Date: November 25, 2011
Due Date: December 10, 2011

Date	Service	Amount
November 2, 2011	Mowing	$ 20.00
November 9, 2011	Mowing	$ 20.00
November 16, 2011	Mowing	$ 20.00
November 23, 2011	Mowing	$ 20.00
November 30, 2011	Mowing	$ 20.00

Balance Due: $ 100.00

Payment is due 15 days from the invoice date. Late payments will result in a $10 late fee. Please make all checks payable to "Lawn-O-Matic". If you have any questions concerning your bill, please call us at (801)-427-1948.

The sub procedure then outputs the “Bill” sheet to a .pdf version and saves it in the same location as the workbook with the name of the file being CustomerName_Month. The program then refers to the “customers” tab to see if we have an email address for that customer. If we do, it creates an email and attaches the file created to the email and sends it to the customer. If the email sends correctly, the program puts the date, customer, and file name in the “sent” tab as well as makes the file name a link to the file. This allows us to refer to that invoice in the future by simply click on the link from Excel. If the email doesn’t send correctly then it does the same output except it indicates “Bad Email” instead of the date on the “Sent” tab. The sub procedure will then move all records that were included in the bill from the “Work” tab to the “Billed” tab.

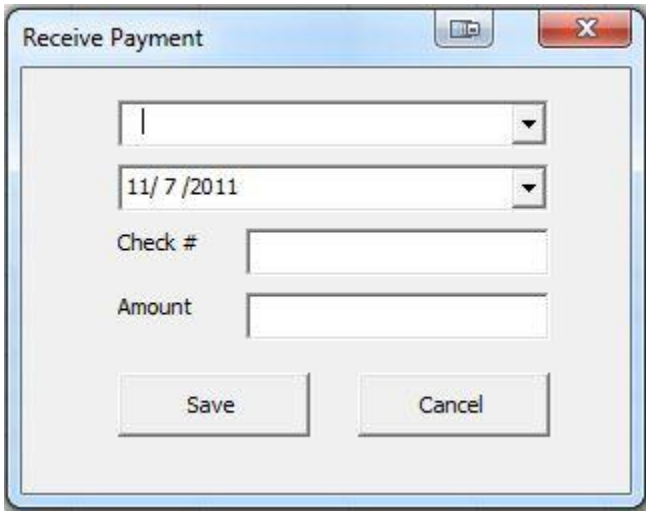
Date Sent	Customer	File Name
Bad Email	Suzanne Ricks	C:\Users\Sommerfeldts\Documents\LOM\Suzanne Ricks_Nov.pdf
25-Nov-2011	Brian Jensen	C:\Users\Sommerfeldts\Documents\LOM\Brian Jensen_Nov.pdf
25-Nov-2011	Daniel Childs	C:\Users\Sommerfeldts\Documents\LOM\Daniel Childs_Nov.pdf
25-Nov-2011	Dian Parks	C:\Users\Sommerfeldts\Documents\LOM\Dian Parks_Nov.pdf

If we don't have the customer's email address on file, the program will automatically print out the bill to the default printer of the computer using the program.

If the user clicks on the "All Bills" button, the program will loop through all the customers and do the same process as the single bill. The sub procedure also checks to make sure that the amount being billed is not $0. If it is $0, the procedure will not save, print, or email the file. After all bills have been sent or printed, the program moves all those records to the "Billed" tab.

Receive Payments

When customers send in checks, the user will click on the "Record Payment" button. This brings up the following form:



The user will then select the customer who is paying from the drop down combo box, change the date if required, enter the check #, and enter the amount. The program checks to make sure all fields have values, and then records the information in the "Payments" tab.

Customer	Date	Check Number	Amount
Lance Kirkland	November 7, 2011	4582	$ 75.00

After recording the information in the payments tab, the program will compare the amount paid to the amount that has been billed. If the amount paid is the same as the amount billed, the sub procedure will enter the check number next to the records of work that correlate with the billing and move those records to the "Paid" tab.

WorkID	Date	Customer	Service	Amount	Bruce	Ryan	LOM	Check Number
	4-Nov-11	Lance Kirkland	Edging	$ 5.00	$ 3.33	$ -	$ 1.67	4582
	11-Nov-11	Lance Kirkland	Edging	$ 5.00	$ 3.33	$ -	$ 1.67	4582
	18-Nov-11	Lance Kirkland	Edging	$ 5.00	$ -	$ 3.33	$ 1.67	4582
	25-Nov-11	Lance Kirkland	Edging	$ 5.00	$ 1.67	$ 1.67	$ 1.67	4582

If the amount the customer paid is greater than the amount billed, the sub procedure will move all records for that customer from the “billed” tab to the “paid” tab and enter the difference as a credit in the “work” tab.

WorkID	Date	Customer	Service	Amount	Bruce	Ryan	LOM	Check Number
	25-Nov-11	Lance Kirkland	Credit	$ (55.00)				4582

If the amount the customer paid is less than the amount billed, the sub procedure will move as many records for that customer as are covered by the payment from the “billed” tab to the “paid” tab and will then display a message box that indicates how much the customer still owes.

Learning and Conceptual Difficulties

I had a pretty good idea of what I wanted the program to accomplish conceptually. I have been dealing with this spreadsheet to track the work that we do for 10 years so I already knew what it did and did not do. I also have managed the billing process for years so I knew what would make that go smooth and easy. I had tried to automate the billing process some in the past through using Outlook but that still took awhile and required a lot of user input. Because of those past experiences, I wanted a program that would do almost all the work automatically.

The biggest problem I faced was in the debugging process. I worked through the program in the same way that we would use the spreadsheet manually. I started off by figuring out how to record the work performed automatically. Then I went on to work through the billing process and then finished with the receiving payments process. With each step, a lot of debugging had to occur.

The debugging proved to be difficult in large part because I worked on the project in spurts. I would work on it for a few hours and then come back the next day to debug it. This caused some headaches because I would go back to debug and not remember why I had some portion of the code there. It would then take a few minutes to figure out what that particular code was supposed to do and why it wasn’t doing it. I often found myself completely re-writing the code that I was debugging as an idea about a different way to do it would come to my mind during that process.

Another issue I faced was integrating it all together. When I first wrote the code, I wrote each portion so it could stand alone by itself. When I tried to integrate, I found errors that didn't present themselves when the code stood by itself. One particular issue that occurred was when I changed how the bills were emailed. I originally wrote that portion of the code to email it using Microsoft Outlook. After we learned how to send the email directly from Excel in class, I went back and re-wrote that section of code. When I was done, the single bill email would work correctly but when I tried to send out all the bills it wouldn't work correctly. It would send the first bill and then not send any of the others.

I thought the problem must be with the code that sent the email but as I further examined it I finally figured out that it was actually not populating the invoice correctly which made it so the total amount being invoiced was $0. Since I wrote a line of code to not send or print an invoice that was for $0 these were not getting sent. I was able to find the line of code that wasn't working properly and fix it but it took several hours to completely fix it.

In conclusion, I believe the program I wrote will save me and my brothers who run the company with me a lot of time as so much of our process is now automated. In doing the project, I came to see how programming can be used to solve every day issues and save us a lot of time. I also learned that the debugging process is often long and arduous but in the end, it is worth it.