Deliverable G - Prototype II & Customer Feedback

By: Ryan Dick, Hamza Khattab and Tuba Minhaj

#

# Table Of Contents

[**Table Of Contents**](#_kiw1zsgqxe2c) **1**

[**Introduction**](#_pfws5qwseg5o) **2**

[**Experimental Model**](#_hwxlo1hzhhnd) **2**

[**Results & Analysis**](#_127y1ba83omf) **4**

[Test 1: Addition and removal of businesses from the list of partners](#_mx5wkihnpc2y) 4

[Test 2: Assigning tiers to partners](#_ncsbxpz0046t) 5

[Test 3: Change how points are calculated by different tiers](#_mrr89aixlxsq) 5

[Test 4: Feasibility of automating the implementation of businesses](#_lplssmuvu47q) 5

[Test 5: Feasibility of combining both prototypes in the time constraint](#_qkjx0ixc9g43) 6

[Test 6: Will the implementation of businesses change Data analytics?](#_titpxgth25ki) 6

[Test 7: Time it takes for a prototype to update when the data is changed](#_1b9tngvb4ckg) 7

[**Feedback**](#_v77xfx3e9dhj) **8**

[**Test Plan For Prototype 3**](#_x060g8lio7fx) **8**

[**Conclusion**](#_x42hh92moscf) **9**

#

# Introduction

In this deliverable, we made our prototype and tested it according to the test plan outlined in the last deliverable. This document intends to discuss the results and feedback from the prototype. The first section aims to explain the objectives of the experimental model made, explaining what new additions to the previous prototype were made and what new subsystems were created. The subsequent section analyzes the test results, followed by the analysis of the feedback gathered from the potential users. Finally, the last section is the plan for the third prototype.

# Experimental Model

For this prototype, we created a separate Excel file that contains a list of businesses with the status of either High Tier, Low Tier or None (ie. businesses that didn’t join the Loyalty Program).



*Figure 1 - Tiered Business File*

The Rate column (bolded) is calculated using an if statement. High Tiered businesses have a rate of 5pts/$, Low Tired businesses have 3pts/$ and businesses that aren’t part of the program have a rate of 2pts/$. If the business changes from “None” to “High Tier”, for example, their rate of points will change according to the named cells. If the banks decide that the high tiered businesses will have double the points this week, all they need to do is change the value of the named cell.

When a business is added to the list, it updates the “Query 1 (2)” sheet in the Prototype 2 Test Case Excel File. That sheet is connected to the “Bank Data” sheet through the XLOOKUP function, and it is used in the “Points Earned” column to multiply the amount of money they spent by the rate of the business. Thus, when a transaction is made at this newly added business, their earned points will be calculated according to the businesses tier.

This excel file is important as it is where the information regarding business partners is held. Without this, we cannot add or remove businesses from the program, making the whole premise of the program impossible to complete. The “Tiered Points” table is used to calculate the points assigned to each tier. This allows the Banks to change the value of the points attributed by each tier, giving them control over how the points would be calculated. The business table (the blue one) is where each business in the program is assigned a tier. This allows the banks to choose how many points would be given by the business. As of now, the Type column does not serve any purpose, but it is there for the next prototype, which will be explained in the appropriate section. This subsystem was completed in a different Excel file as it is a separate component from the data analytics, and the information contained within this file does not directly interact with the Power BI analytics.



*Figure 2 - Formula Used to Calculate Points*

We have also improved upon the data analytics prototype made in the previous deliverable. This was done by programming some functions in VBA. These functions allow for the creation of a new sheet for every new user added to the table. This sheet will include all of the transactions corresponding to the specific user, allowing for the data to be filtered by each user. In addition, any updates made to the cells will also refresh this, updating the data in each individual sheet to reflect these changes.



*Figure 3 - Sheets Created for Every Unique User*

# Results & Analysis

### Test 1: Addition and removal of businesses from the list of partners

In the picture below, Home Hardware was added to the list.



*Figure 4 - Adding a Business to the List of Businesses Table*

To add the business, you currently need to manually type in the business and set the business type. This was done because this is the most efficient method that we could do in the time constraint. Another method we thought of was to have the Excel file access an online database that contains a list of all the businesses in a certain region, but this would require a significant amount of time to program, so we decided to stick with manually inputting the information. The status or the tier of the business is by default set to none, or 2pts/$. Businesses that have not joined the program default to the “None” tier, giving cardholders a default of 2pts/$. The rates chosen are also just arbitrary values, used only due to the noticeable differences that could be seen when the points are calculated. This default setting was done to simulate how banks would add a business to their program database, as they do not start as part of the program, resulting in the business defaulting to no tier and requiring an employee at the bank to update the business tier as required. Adding a business took 1 minute to test.

### Test 2: Assigning tiers to partners

We manually changed the status of Home hardware from No Tier to High tier, so now the rate is 3pts/$, as seen in the figure below.



*Figure 5 - Setting the Tier of the Business*

The assigning of tiers is done through a drop-down menu in the table. The drop-down menu includes the three tiers we used for this prototype; “None”, “Low Tier”, and “High Tier”. These tiers were used as they allowed for easily distinguishable results in this test as well as other tests conducted with this prototype. The drop-down menu also makes it very easy to assign different tiers to each business, and it is more convenient than having to manually type the tier.

### Test 3: Change how points are calculated by different tiers

Below is a picture of part of the “Bank Data” sheet that calculates the Points Earned by users.



*Figure 6 - Bank Data Table Demonstrating Calculation of Different Point Rates*

It calculates this value by multiplying the amount spent by the rate of the business. For example, Staples (first row) is a Low Tier business so the $78 that was spent there is multiplied by 3 to get 234 pts and $455 spent at Home Depot (second row) would get a user double the number of points. This is how the tiered partner system works.

### Test 4: Feasibility of automating the implementation of businesses

As previously mentioned in test 1, we did not fully automate the implementation of businesses. Due to the time constraint of this prototype, as well as our small group size, it simply was not feasible to do. The idea we had was to connect it to some online database that had a list of businesses in a region, such as a list on Wikipedia. The program would then go through the list, adding each business to the table in Excel. This would require a significant amount of time spent both coding this and learning the programming knowledge required to complete something like this. We have decided to put this part of the project on hold, and to only look at ways to do this if time permits after the next deliverable.

### Test 5: Feasibility of combining both prototypes in the time constraint

The combining of both prototypes turned out to be very simple. We had used the built-in feature within Excel that allows for data to be imported through a Sharepoint folder. This feature was used as the Excel file containing the tiered businesses was located in the Sharepoint folder. This then automatically creates a query from the Data Analysis file to the Tiered Business File. Before using this query feature, we had used a direct URL in the XLOOKUP. While this method worked in the desktop app, it resulted in a #REF! Error in the Excel web app. Because of this, we had decided to switch to the built-in Power Query, which not only solved this problem but also gives the Excel file access to the data in any file that may be added to the SharePoint folder in the future.



*Figure 7 - Table Created by the Query Between the Two Prototypes*

### Test 6: Will the implementation of businesses change Data analytics?

The point of this test is to see if the businesses added (Home Hardware) will appear on the Power BI Interface. Currently, this is a graph that shows which businesses users tend to earn their points. Notice Home hardware is not part of the graph.



*Figure 8 - Graph Showing Points Earned by Business Before Changes*

Now after refreshing the data, Home Hardware appears as the last bar in the graph shown below.



*Figure 9 - Graph Showing Points Earned by Business After Changes*

This graph is important as it allows us to see how the new prototype could affect the data analytics made in the previous deliverable. We also picked this specific graph, which shows the distribution of points across each business, as this is one of the data types we have decided to allow banks to filter for in the next prototype. This is important for the concept of the tiered system, as higher-tiered businesses will receive more benefits from the program, and this graph allows banks to see where most of their cardholders tend to earn their points. This could then be used by banks to offer increased benefits to the specific business by creating a new tier, specific for them. This test took about 10 minutes to complete.

### Test 7: Time it takes for a prototype to update when the data is changed

Home Hardware was added to the list of businesses in Test 1. All that is left is to refresh the data on the Prototype 2 test cases and then add a transaction to the Bank Data Sheet. This whole process took less than 1 minute.



*Figure 10 - Demonstration of the Updating of Data Between Prototypes*

# Feedback

This prototype was designed for the banks and was meant to be shown to the client during the meeting we were supposed to have on Tuesday, but since it got moved to another date, we didn’t get any feedback from them.

# Test Plan For Prototype 3

| Test ID and Priority | Test Objective**(why)** | Description of Prototype used and of Basic Test Method**(what)** | Description of results to be recorded and how these results will be used **(how)** | Estimated Test duration and planned start date **(when)** |
| --- | --- | --- | --- | --- |
| 1 | Create a Separate Chart for Each User in Power BI | Using Power Automate, have Power BI create automatically create/update charts for each user sheet made | * Test the Creation of Different “Pages” of Charts for Each User
* Test How Updates to Each Sheet Affect the Charts
* Test How Creating New User Would Make New Charts
 | Nov 11 - Nov 16 |
| 2 | Improve Upon the Implementation of Tiers | Using the built-in functions of Excel and VBA, allow for the creation and removal of tiers | * Test the creation and removal of tiers
* Test the addition of incentives to each tier (offers made to partners)
* Test how the updated method interacts with the previous system
* Test to see how Excel will update when the tiers are updated
 | Nov 11 - Nov 14 |
| 3 | Create a UI for ‘housing’ the different parts and for demonstration purposes | Using Power Apps, create a working UI that allows users to interact with the different parts of the system for demonstration purposes | * Test how the main navigation page will connect to the rest of the components
* Test to have transactions (bank data) be made in the Power App
* Test to have the List of Businesses File edited through the Power Apps (inlc. Tiers and partners)
* Test to see how the different Power BI datasets can be accessed from Power Apps
 | Nov 16 - Nov 24 |
| 4 | Allow for Filtering of Data | Using the built-in features of Power BI, allow users to filter data as they please | * Test how users can switch between different views of data
* Test ability to revert filters
* Test option to keep filters every time the specific page is accessed
 | Nov 16 - Nov 24 |

# Conclusion

Using the feedback from our first prototype, we outlined a number of key factors in improving our design focused around, data filtering, UI improvements and tier implementation. The prototype that came, addressed most of the factors, more importantly, we also conducted feasibility tests. Taking the successes and tests from this prototype, our group plans one final prototype to bring together features to create a “full-flow” prototype. We plan for our third prototype to include an improved app interface through direct feedback, and a tiered benefits system. Once this final prototype is complete, our goal is to approach design-day with confidence that we have a well-developed product.