GNG1103

Design Project User and Product Manual

User and Product Manual Instruction

Submitted by:

Z-Ecosystem (Group 11)

Markos Brown, 300170812

Kaiya Brereton, 300244096

Isaac Winsor, 300226614

Darren Rahnemoon, 300189815

Adam Ottaway, 300076155

Dec 8, 2021

University of Ottawa

Table of Contents

List of Tables

List of Acronyms and Glossary

Table 1. Acronyms

Acronym	Definition
Z-Ecosystem	Zafin Ecosystem

Table 2. Glossary

Term	Acronym	Definition

Abstract

The contents of this manual describes the loyalty program that was created by the Z-EcoSystem to be used by the client, Zafin. It documents the process one would follow to correctly recreate the complete final design using clear guidelines and instructions. Several aspects of the project will be covered such as the bill of materials, the considerations one must take into account when using the product, and detailed explanations of the uses and modes that the design contains. In addition, important files and links to the MakerRepo repository will be included in the manual, which aid in recreating the final design. This document also includes any additional comments and/or recommendations regarding the process of creating this module to ensure any person that wishes to recreate this project may find it easy and understandable.

1 Introduction

Zafin is a world leader, trusted by partners with years of experience worldwide interactions. Their goal is to address the problem of how points are specific and limited in their application (air travel, consumer goods, repay with points, etc.). Their solution was to “unlock” the economic value in loyalty programs by demarcating loyalty programs within financial institutions worldwide and their ecosystem partner.

This program needs to be a decentralized network that allows financial institutions to incentivize their users to spend more points and as a product accumulate more points and spend more money through the institution’s infrastructure. Moreover, the system should be able to receive and interpret data, with big and small financial institutions

By the end of this manual, the reader should have the necessary knowledge to reproduce the loyalty program; however, there are a few key assumptions to consider before undertaking this project. It is expected that the user has sufficient coding skills and has experience with software; in addition, it is assumed that tlocation for the platform and the demographic. With these skills, and assumption in mind the reader will be a suitable candidate for this design.

This User and Product Manual (UPM) provides the information necessary for account holders at banks to effectively use the Z-EcoSystem and for prototype documentation. Note there is no associated security or privacy actions required.

2 Overview

Our platform is designed to be easy to use, allows customers to earn points, redeem as well as exchange their loyalty rewards points. In addition, the system should act as a network that financial institutions can join and easily have access to other insitittion’s points of exchange. With this in mind, it is easy to change rates and relationships with other points systems based on business strategy decisions and allows Zafin to be independent of the system - financial institutions and ecosystem partners communicate directly with one another without the need of an intermediary party (Zafin). Our system employs a few assumptions such as the starting location for the platform and the demographic.

Zafin aims to develop a network of financial institutions that can join and easily have access to other insitittion's points of exchange. Ultimately this system must be easy to implement and unimplement when needed as well as easy to change rates and relationships with other points systems based on business strategy decisions. Although the most fundamental aspect of the system is that decentralized platform in other words, Zafin does not manage the platform. Rather the platform itself resembles a marketplace of companies that Zafin hosts on their servers.

The key features of our product are the base point system and the point exchange system. The base point system allows all entering parties access to the system. In the event that there is a small institute that lacks the infracture to join onto the network they could develop a point system via the base point system. The point exchange system is the "heart" of our design. In essence it is a dating app for banks, in the sense that banks and partner businesses have the choice to reach out to the other members of the network.

The unique thing that our product brings is account base point. This aspect of our product allows financial institutions to award points based on the user accounts/ bundled accounts. This way banks can reward users that have certain accounts/products with the institute.

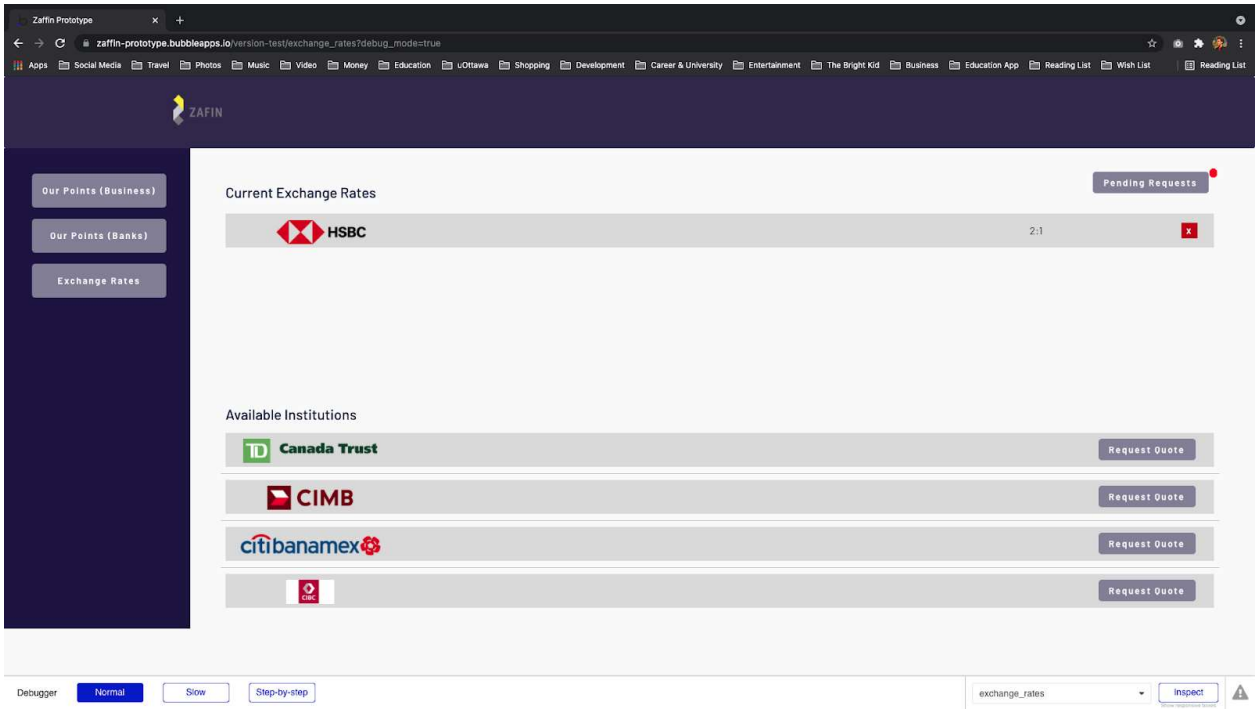


Figure 2.1: Control Panel for requesting new rates from other institutions

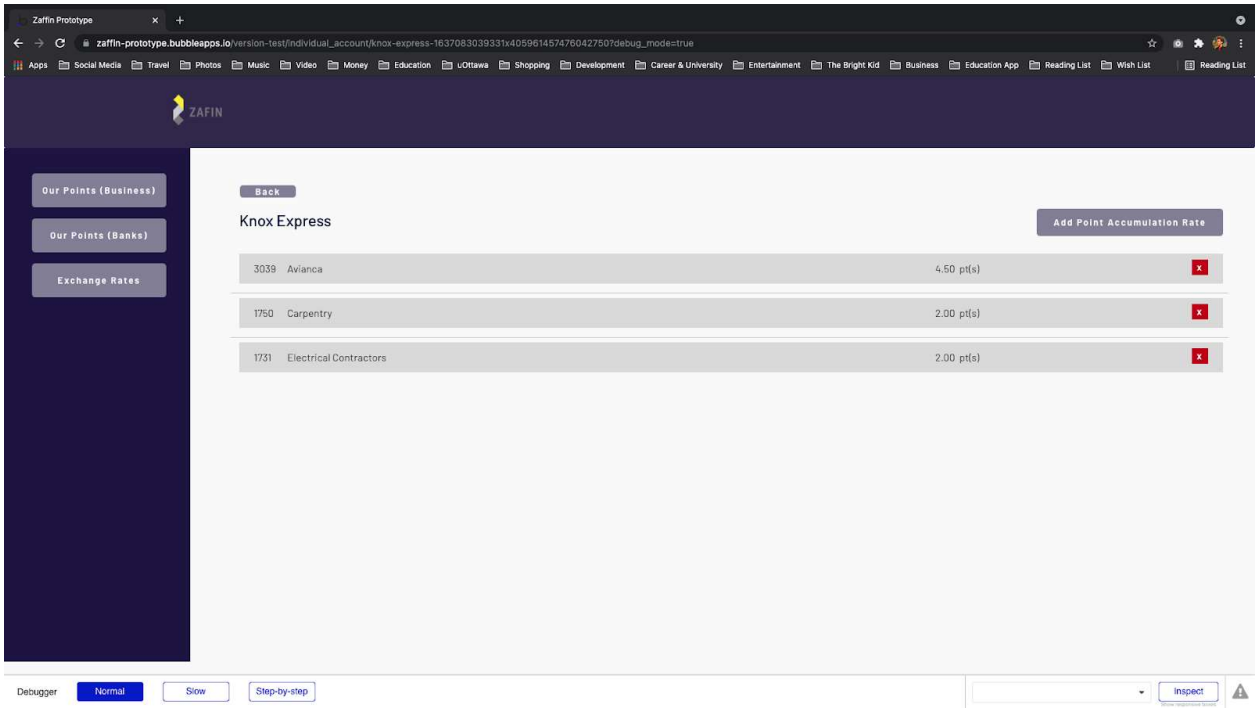


Figure 2.2: Control Panel for creating new point accumulation rates for the account types of an existing bank

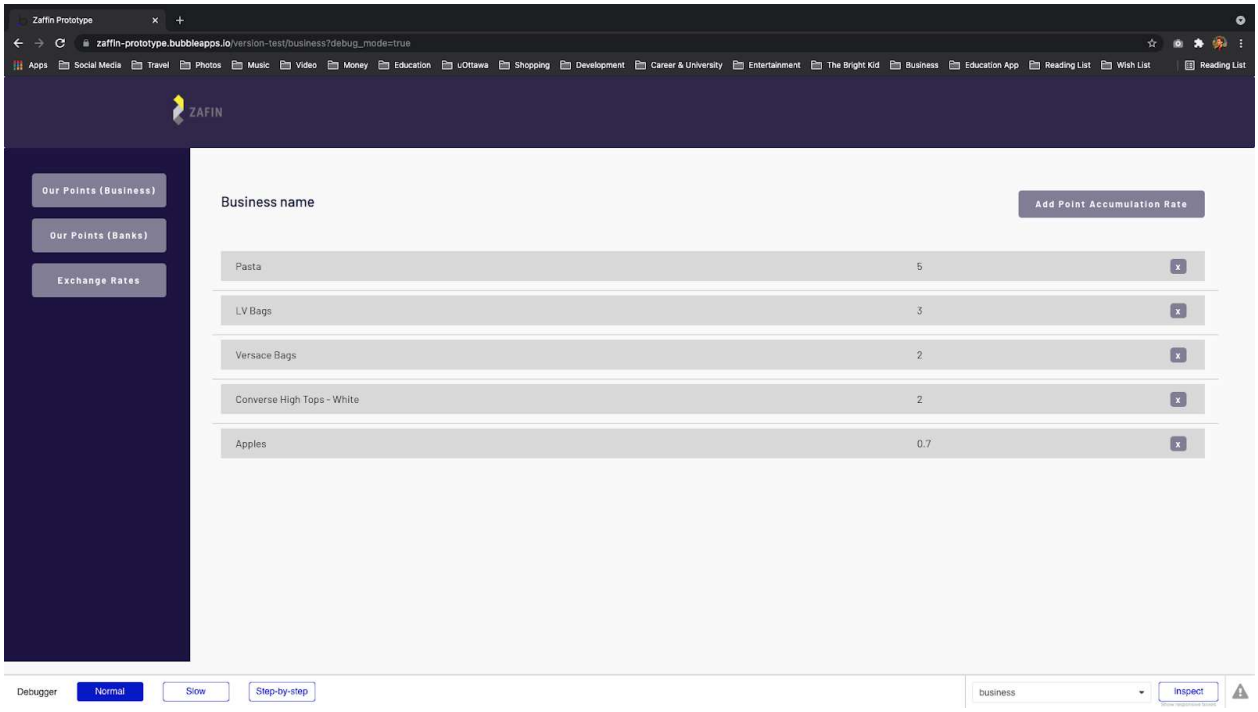


Figure 2.3: Control Panel for creating new point accumulation rates for the products offered of an existing business

The architecture of the system is such that the financial institutes and the partner member of the ecosystem are all linked together via a centreless connection. Our software offers a “back end” or a system that only internal users can see, in other words bank clientele would not have access to this system.

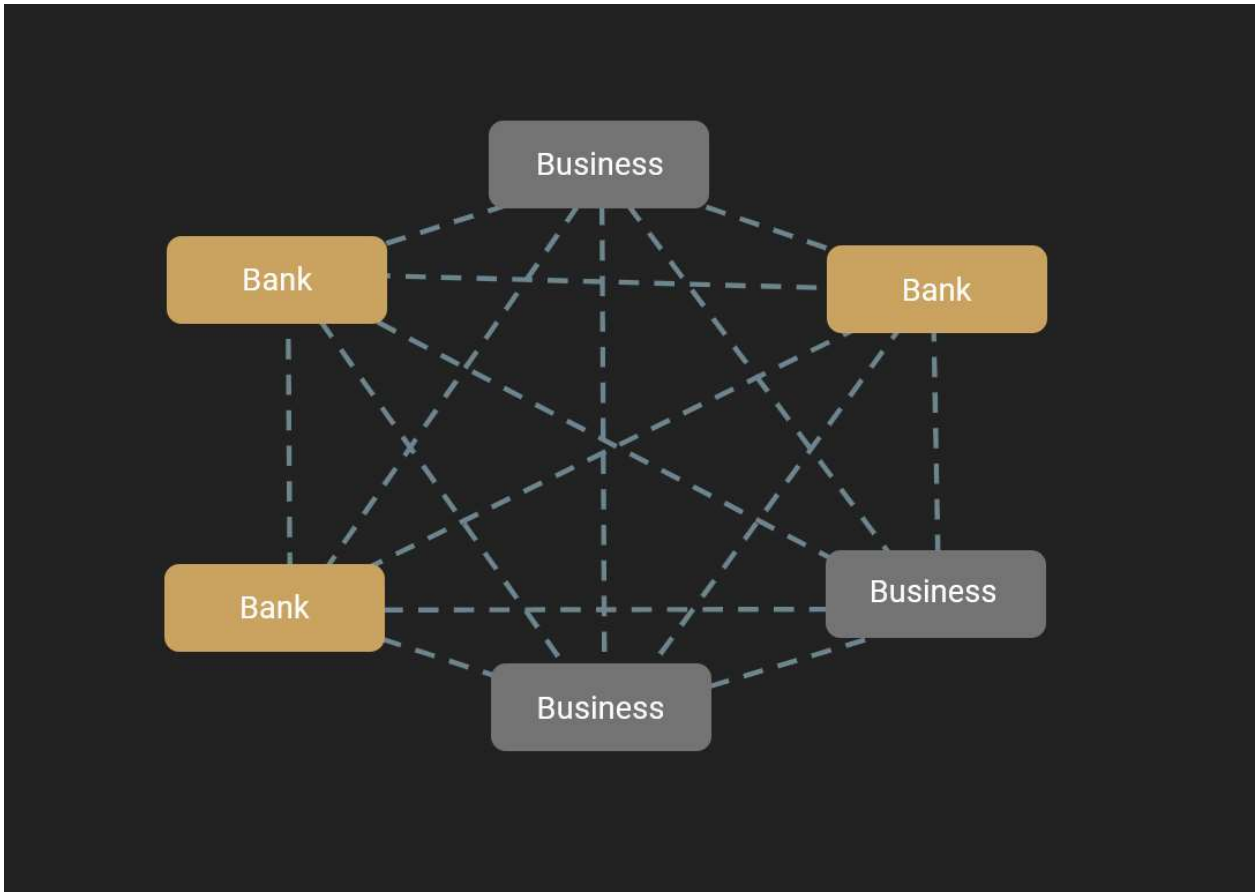


Figure 2.4: Example of the decentralized system, every bank or business that joins immediately has access to everyone on the Z-Ecosystem

2.1 Cautions & Warnings

Do not share your password with anyone even within your own institution. The system is designed in such a way that multiple users can belong to the same institution. If you need access request your administrator to create a user account for you in the node.

3 Getting started

At initiation, the user is placed on the “Our Points (Business)” page of the app. This is what a business using the Z-Ecosystem would see. On this page users can add and remove point accumulation rates for different products.

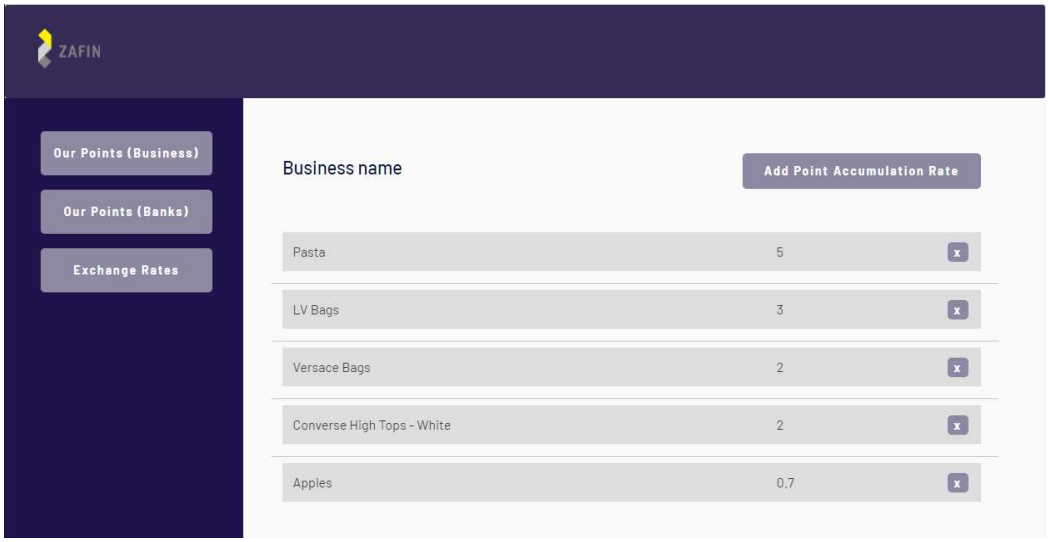


Figure 3.1: Startup page “Our Points (Business)” of Z-ecosystem prototype.

When the “X” beside the product is clicked, the point accumulation rate is removed. When a user clicks the “Add Point Accumulation Rate” button, a popup appears which allows new point accumulation rates to be added.

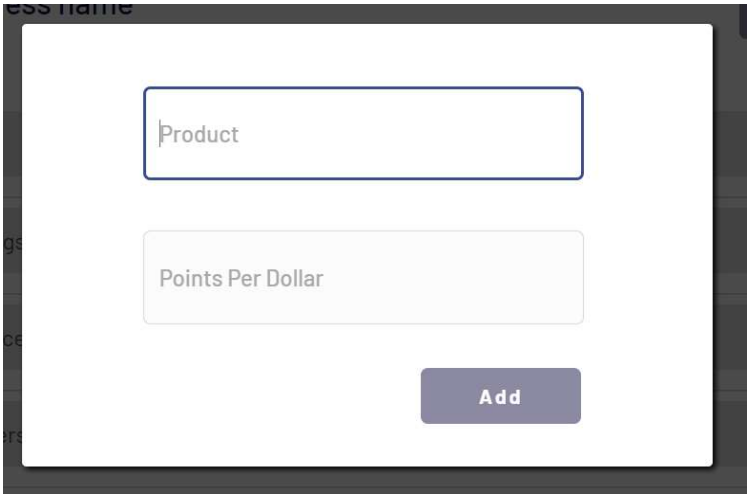


Figure 3.2: “Add Point Accumulation Rate” popup. Allows users to input a product and the points accumulation rate for that product.

By clicking the “Our Points (Banks)” button on the left (see Figure 3.1) users are taken to the banks side of the prototype. This is what banks using the Z-Ecosystem would see. This page allows users to add and remove account types.

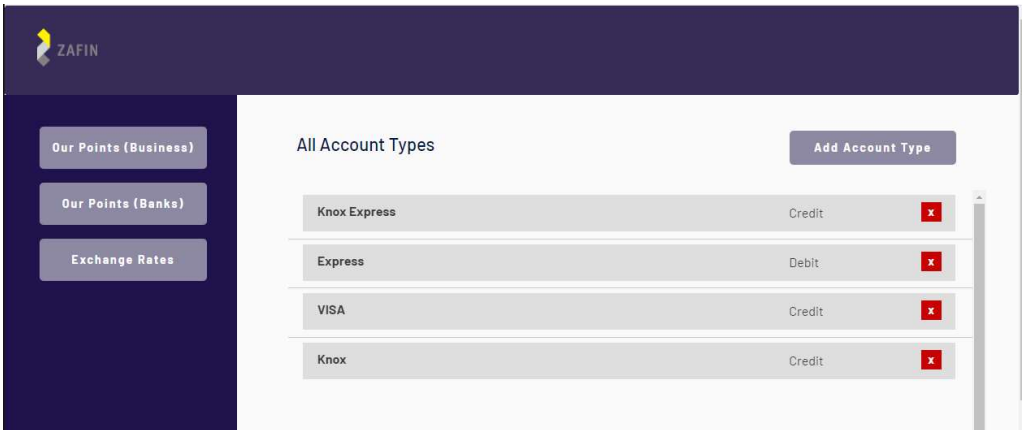


Figure 3.3: Our Points (Banks) page of the Z-Ecosystem

By clicking the “X” beside the account type, the account type is removed. By clicking the “Add Account Type” button, a popup appears that allows users to add a new account type.

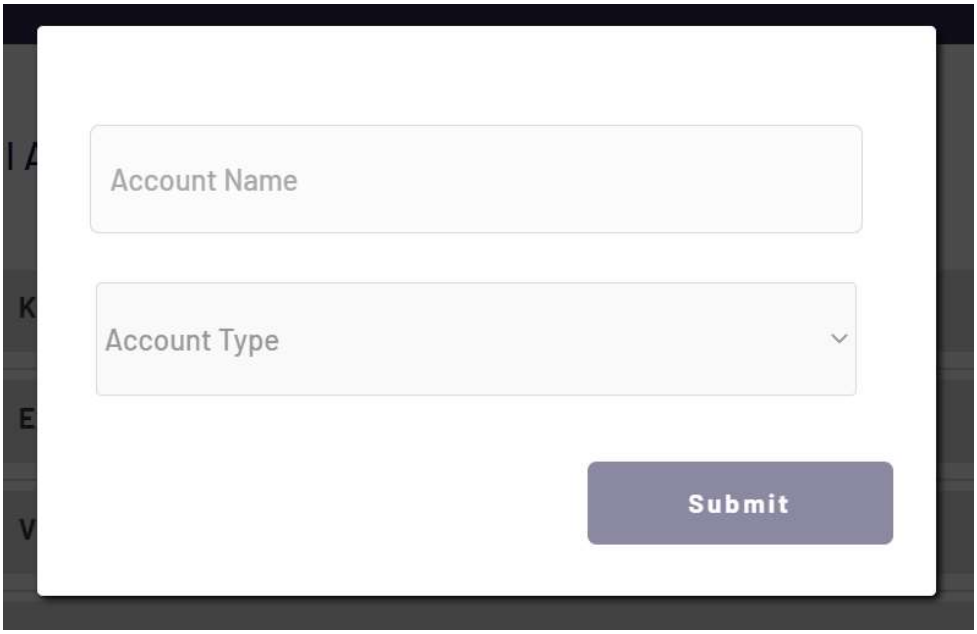


Figure 3.4: “Add account type” popup. Users can enter the account name and choose the account type from a dropdown list that includes the options debit or credit.

After clicking on an account type, users are taken to a page showing all active point accumulation rates for that account type. This page also allows users to add and remove point accumulation rates for that account type. Clicking the back button takes users back to the all account types page.

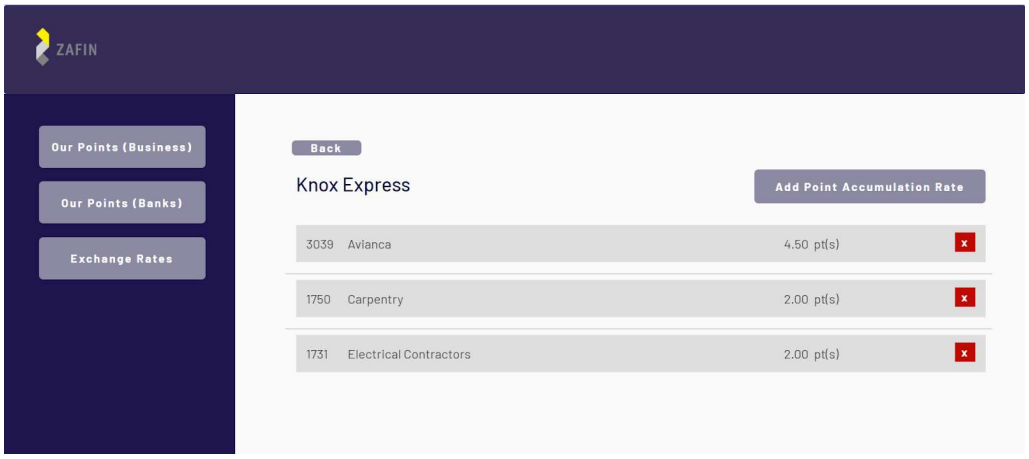


Figure 3.5: Point accumulation rate page for a specific account type.

Clicking the "X" allows users to remove a point accumulation rate from that account type, and clicking the "Add Point Accumulation Rate" button shows a popup which allows users to add a point accumulation rate

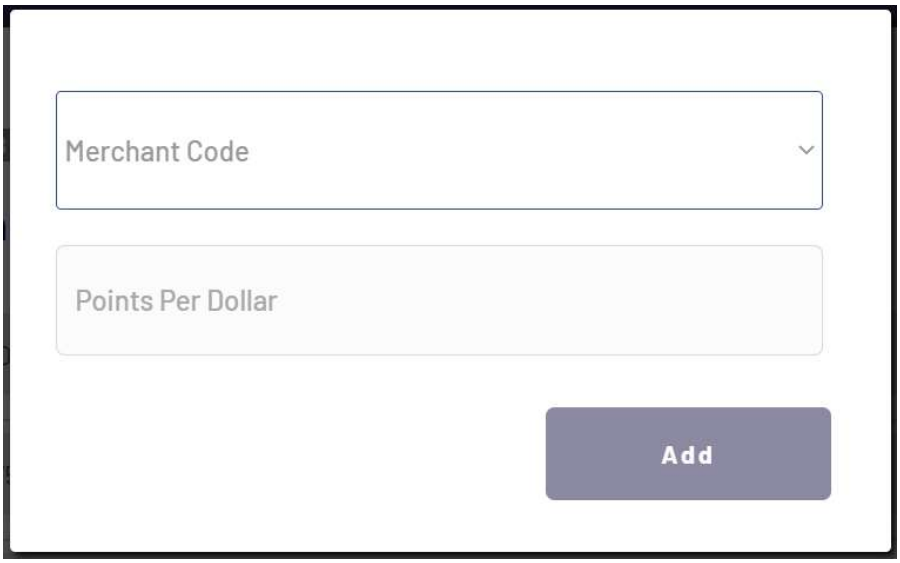


Figure 3.6: "Add Point Accumulation Rate" popup. Users can choose the merchant code from a dropdown list of merchant codes, and input a points per dollar rate for the merchant code.

By clicking the "Exchange Rate" button on the left of the screen users are taken to the exchange rate page. This page is only available to banks. This page allows users to see current exchange rates, request an exchange rate from available institutions, and see pending exchange rates.

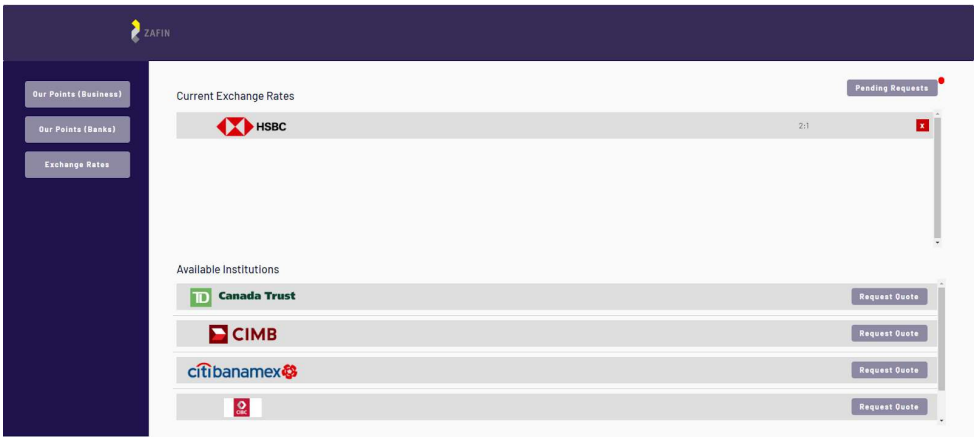


Figure 3.7: Exchange rates page.

By clicking the “Request Quote” button beside an available institution, a popup will appear that allows users to request an exchange rate with the respective bank.

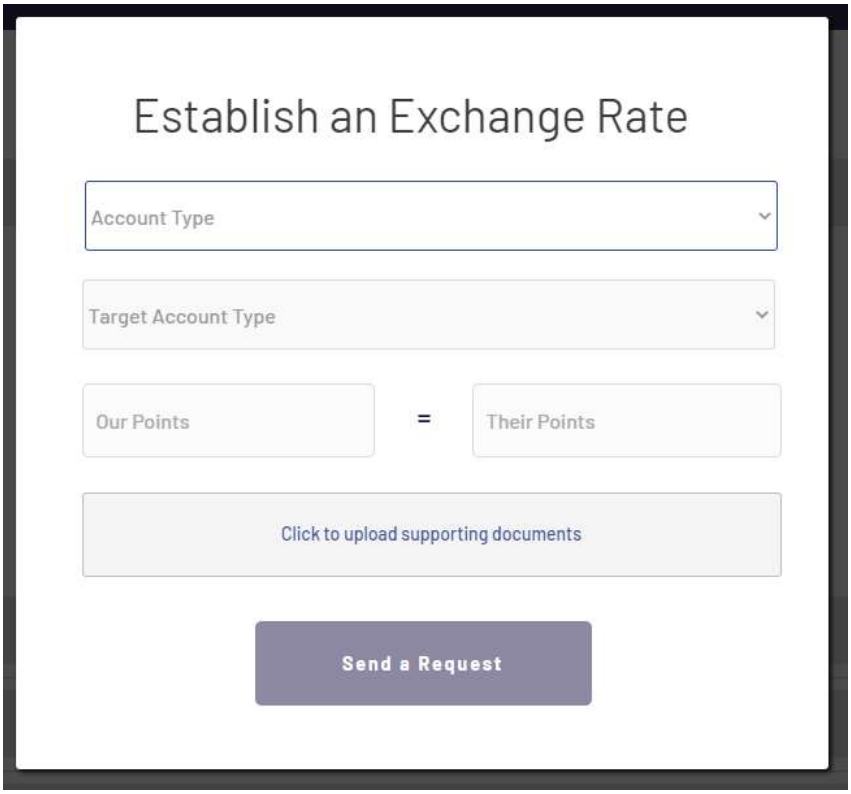


Figure 3.8: “Establish and Exchange Rate” popup. Users can input the account type from their bank, select the target account type from the requested bank, input a suggested rate, and upload supporting documents.

By clicking on the “Pending Requests” button in the top right (See Figure 3.7), users are taken to the “Pending Exchange Rates” page, where all the pending requests from other banks can be seen.

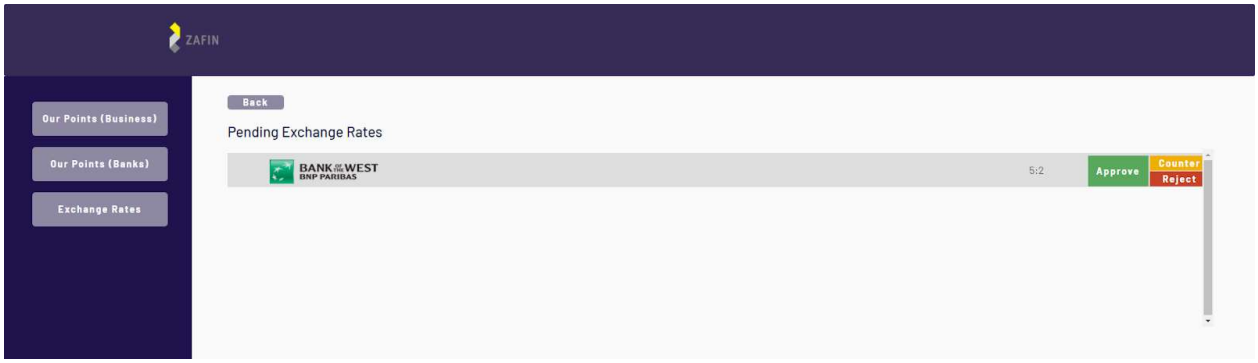


Figure 3.9: Pending Exchange Rates page.

On the “Pending Exchange Rates” page, users can approve, reject, or propose a counter offer to an exchange rate requested by another bank.

By clicking “Approve”, a popup appears that has an “Approve” and a “Cancel” button. This makes sure the user really wants to approve the rate. By clicking “Cancel”, the popup closes and nothing happens. By clicking “Approve”, ideally the exchange rate would be moved to the “Current Exchange Rates” page (See Figure 3.7), however this feature has not been implemented yet, so nothing happens.

By clicking the “Reject” button, ideally the exchange rate would be removed, however this feature has not been implemented yet, so nothing happens.

By clicking the “Counter” button, a popup appears that allows users to submit a counter offer exchange rate to the bank.

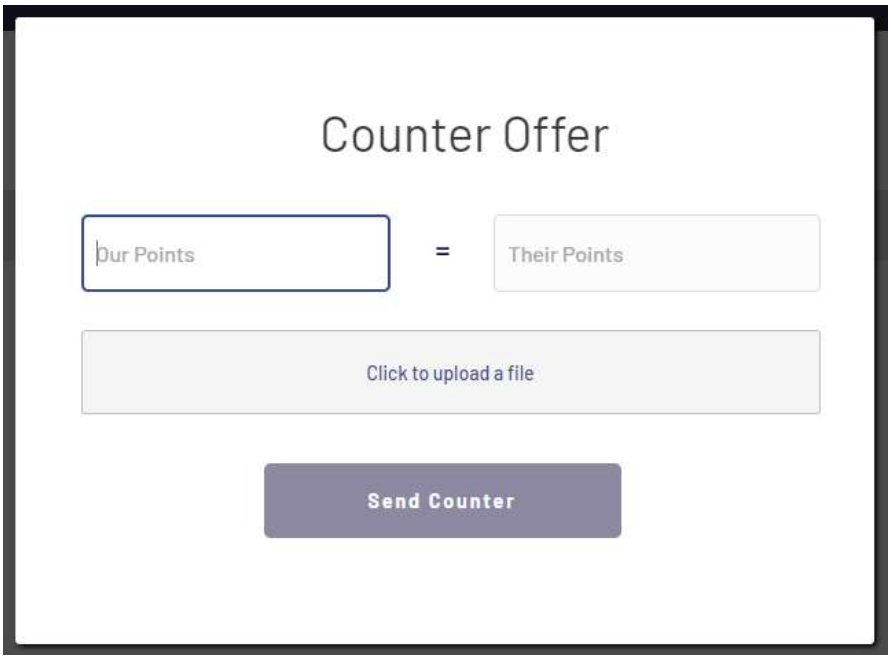


Figure 3.10: “Counter Offer” popup. Allows users to send a counter offer exchange rate and upload supporting documents

3.1 Set-up Considerations

1. Acquire a device that can connect to the internet
2. Use the link below to access the prototype
   https://zaffin-prototype.bubbleapps.io/version-test/business
3. You will immediately gain access to a demo account that gives you access to both the Base Point System and the Point Exchange Rate page

3.2 User Access Considerations

Given the limited time, we were not able to fully implement the authentication system we desired for the control panel but the following user permission system would be a part of our agenda should Zafin wish to continue working with us:

3.2.1 Super User:

Gives the user unrestricted access to the entire node. Useful for the dev team or other administrators or the institution.

3.2.2 Base Point Administrator

Gives the user access to only the Base Point system. Useful for members of the institution who are assigned to maintain the point accumulation rates.

3.2.3 Point Exchange Administrator

Gives the user access to only the Point Exchange Rate system. Useful for members of the institution who are assigned to maintain the rates between their institution and other third-parties on the network.

3.3 Accessing the System

Since we were not able to implement the authentication system the system is currently globally available through the link mentioned in the Set-up Considerations section.

3.4 System Organization & Navigation

Once logged, depending on the institution the user is a part of, the user would either enter the Base Point System for banks or the Base Point System

3.5 Exiting the System

Closing the tab with the prototype open exits the system.

4 Using the System

The following subsections provide detailed, step-by-step instructions on how to use the various functions or features of the Zafin Ecosystem. These functions are the base point and exchange rate systems along with their respective sub-systems.

4.1 Base Points

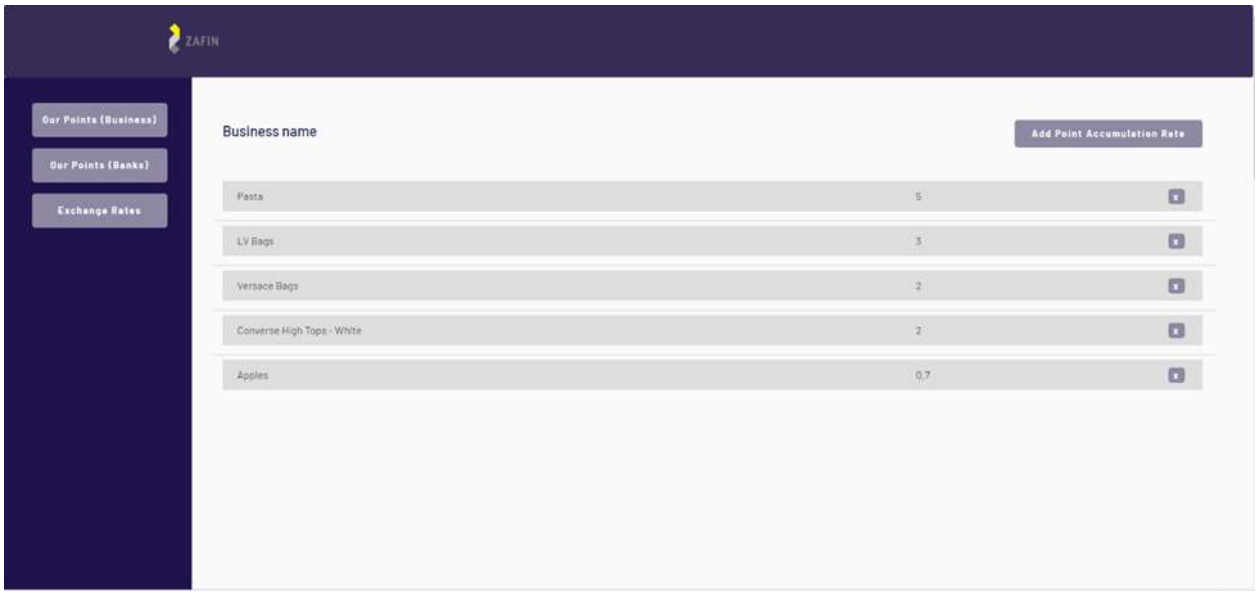
Our base point system is a function that allows banks and businesses access to their own point accumulation rates. This feature was intended for businesses that did not already have access to a loyalty rewards service. This allowed all businesses access to the Zafin Ecosystem.

The base point system was divided into two separate sections. One for banks, and one for businesses.

4.1.1 Points (Businesses)

The “Our Points (Business)” Tab allows individual businesses to apply a specific accumulation rate to items that they are selling. To add a rate, you simply press on the “Add point accumulation rate”, insert the name of your item, and add a rate in the form of points per dollar.

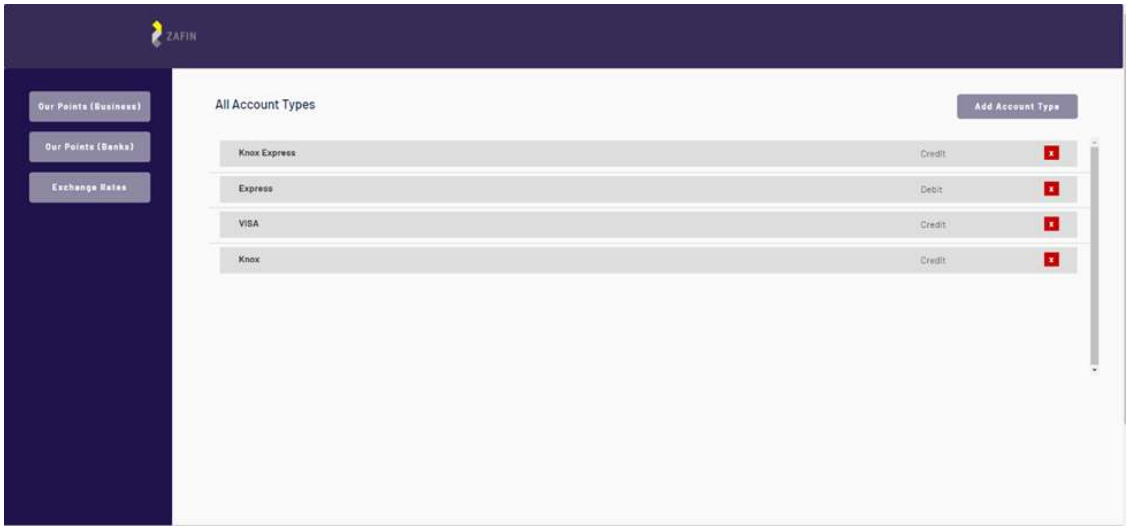
(Figure 4.1) Our Points (Business) page, with multiple accumulation rates for diverse items



4.1.2 Points (Banks)

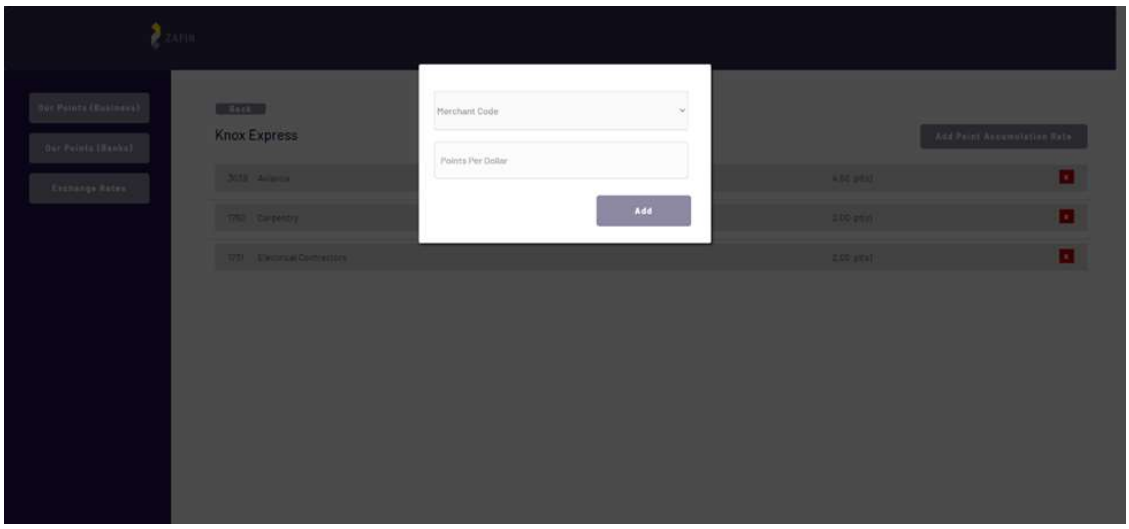
The “Our Points (Banks)” Tab allows banks to create different account types from the back end. This was created to allow banks to create their own accumulation rates for points and to attach them to a given card to be distributed to users.

(figure 4.2) Our points (Banks) page, with multiple credit and debit accounts already created.



From the main page, users can add and edit accounts. When editing an account, you are given many merchant codes as options from our database, and the ability to attach an accumulation rate to said merchant. This will determine how many points are accumulated for shopping with any given merchant.

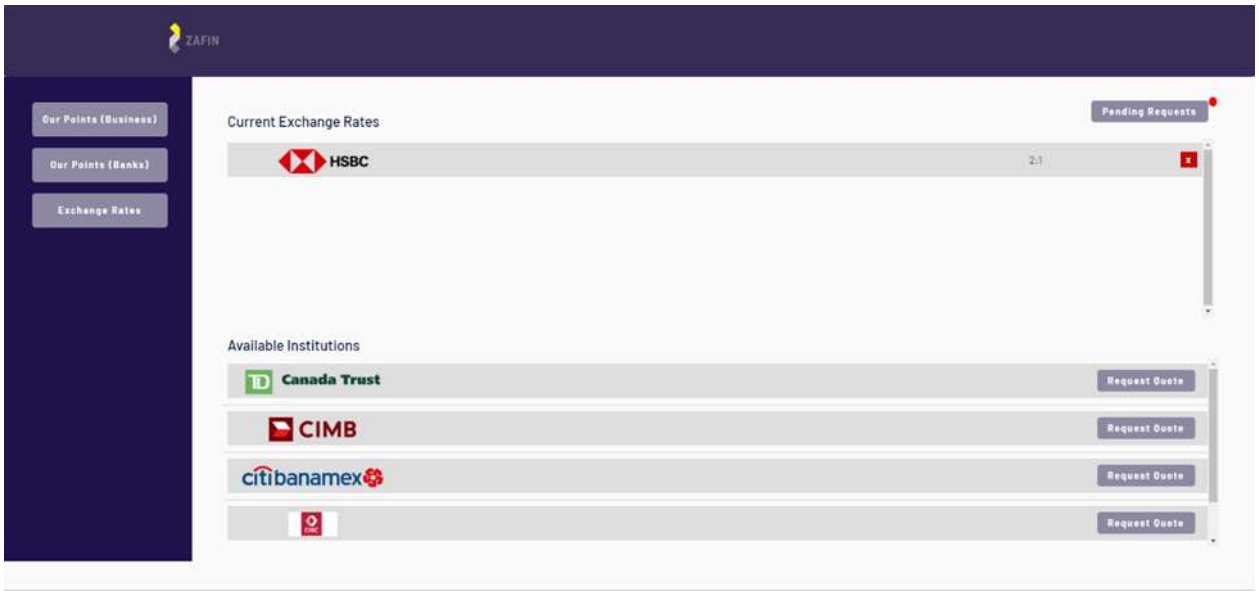
(figure 4.3) Prompt that appears when adding merchants to a given account



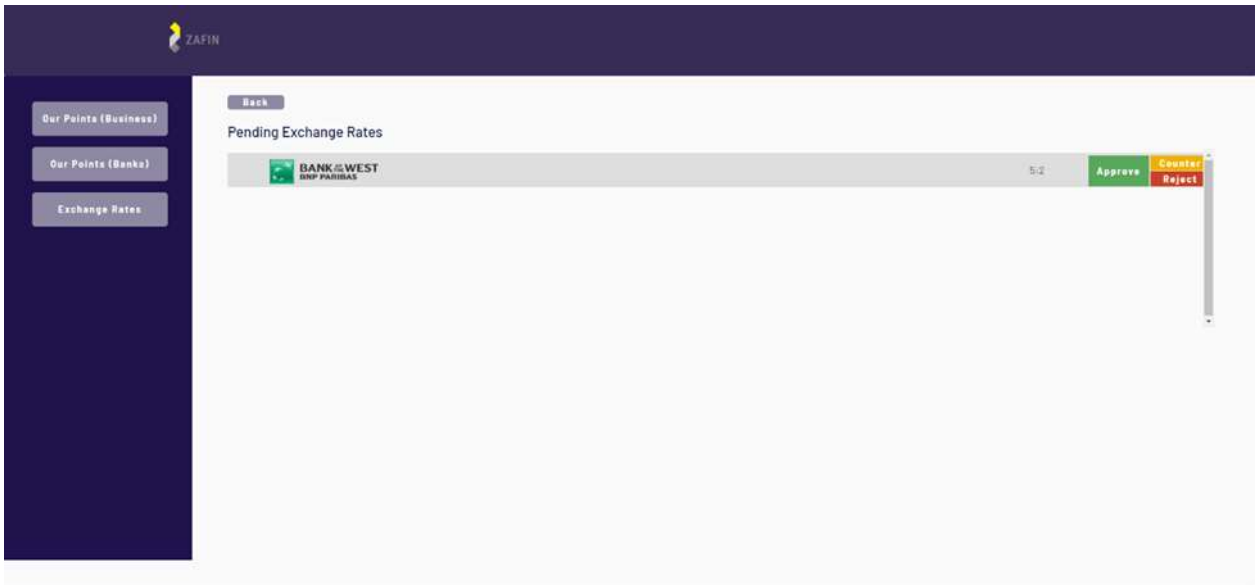
4.2 Exchange Rates

The “Exchange Rates” tab allows banks and businesses to request, accept, deny, and counter exchange rates among themselves. From the main page, you are shown a list of banks/businesses that are a part of the system, and you may choose who you would like an exchange rate with.

(figure 4.4) Main page. From here you can choose from which bank/business you would like to request an exchange rate.



(figure 4.5) Pending request page. From here you can view and interact with requests from other businesses.



5 Troubleshooting & Support

In the case that an error is observed, ensure that all values within the prototype are positive integers. If the issue persists, turn the system off and on again. If an issue is still being observed, please contact a member of our team and we will help as soon as possible.

5.1 Error Messages or Behaviors

If a negative number is added to the accumulation rates, the system may not work as expected. To prevent this, please ensure that all values inserted into the system are positive integers.

5.2 Maintenance

The system should be routinely checked to ensure that all conversion and accumulation rates are present and functioning.

5.3 Support

For customer support related to our prototype, aotta060@uottawa.ca for assistance.

6 Product Documentation

6.1 Base Points

6.1.1 BOM (Bill of Materials)

- Bubble: The best way to build web apps without code | Bubble Cost: Free

6.1.2 Equipment list

- Computer
- Internet connection

6.1.3 Instructions

We start by designing our database structure and what each entity will look like. For our case we came up with the following structure

- User
    - String: Email
    - String: Password
    - Belongs to one: Institution
- Institution
    - String: Name
    - Has many: Users
    - Has many: Account Types
- Account Type
    - String: Name
    - String: Type
    - Belongs to one: Institution
    - Has many: Point Accumulation Rate
- Point Accumulation Rate
    - Number: Points per Dollar
    - String: Merchant Code
    - Belongs to one: Account Type

Once the database structure is designed, we now begin on focusing on the basic layout of the app and wireframes through Bubble. Using the drag and drop interface provided by Bubble, we implement the following view:

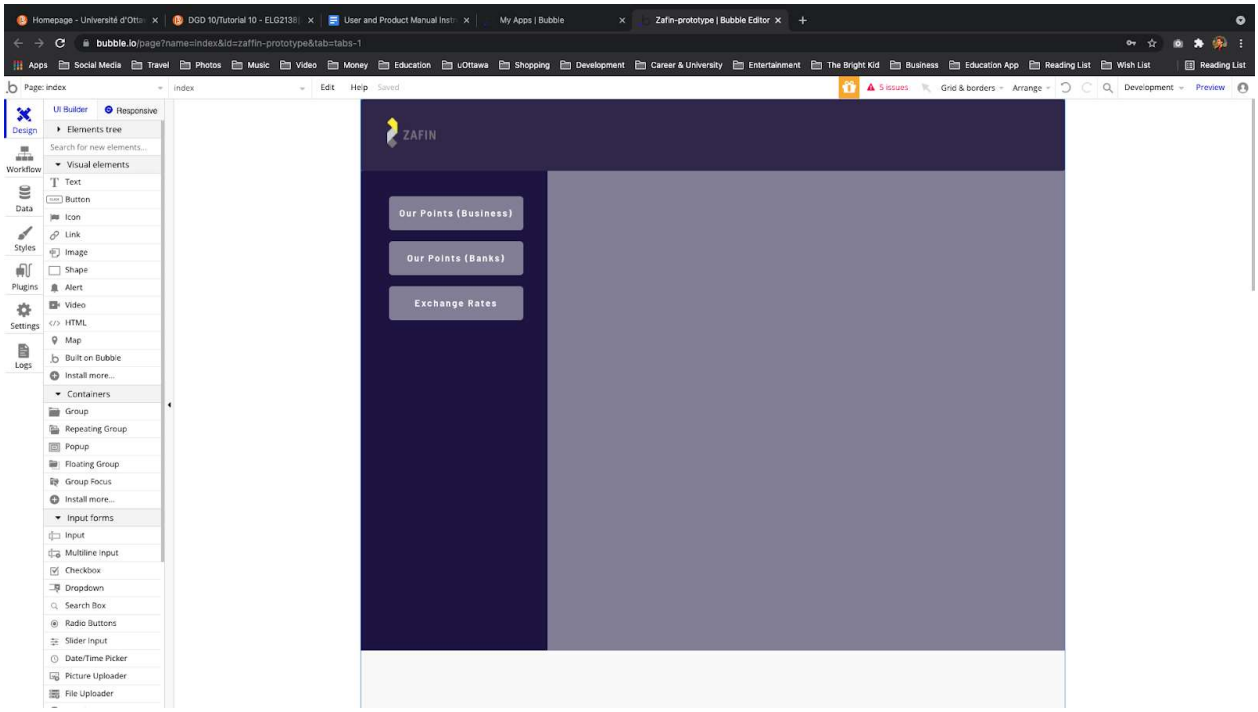


Figure 6.1.3.1: Designing the Basic Layout of the App

We then continue by adding a new page for the Account Types list and another page for the Point Accumulation Rate sub page.

We add a list element and attach it to the Account Types DB entry we created earlier.

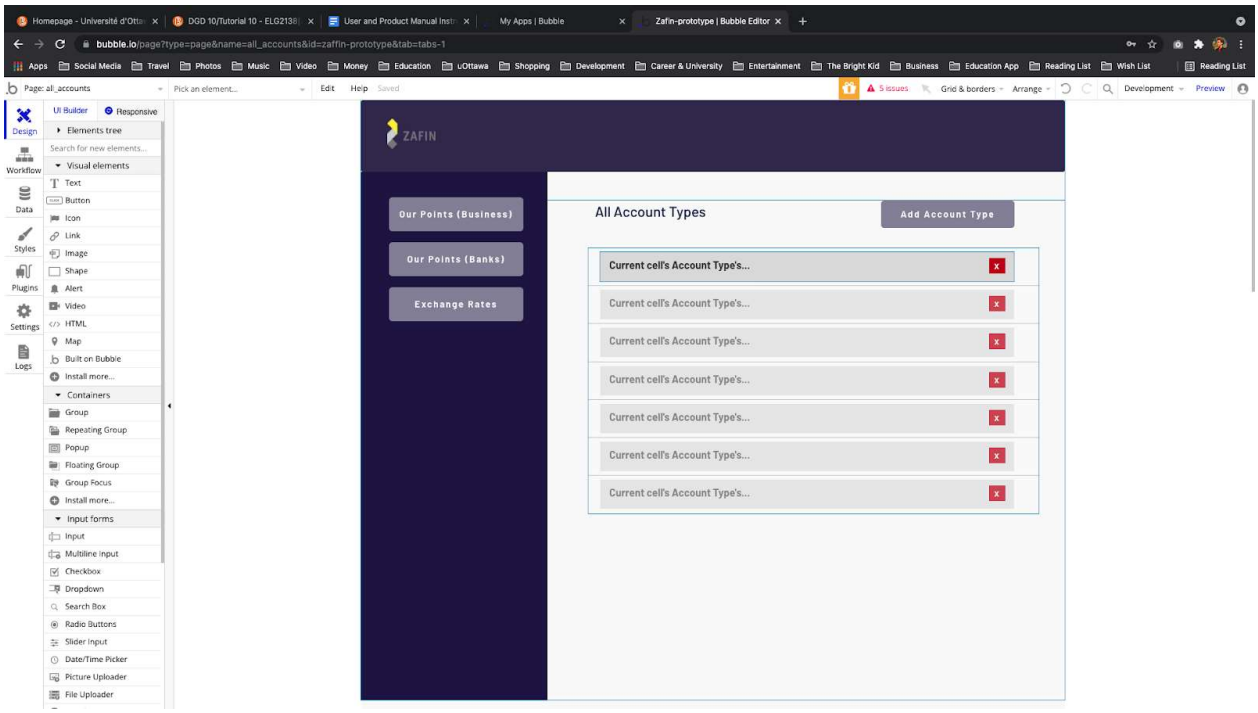


Figure 6.1.3.2: Designed Accounts List page

For each item when clicked, we'd like it to take us to the dedicated account page so we add a click event listener and point towards the point accumulation rate page passing in the chosen account in the context of the next page.

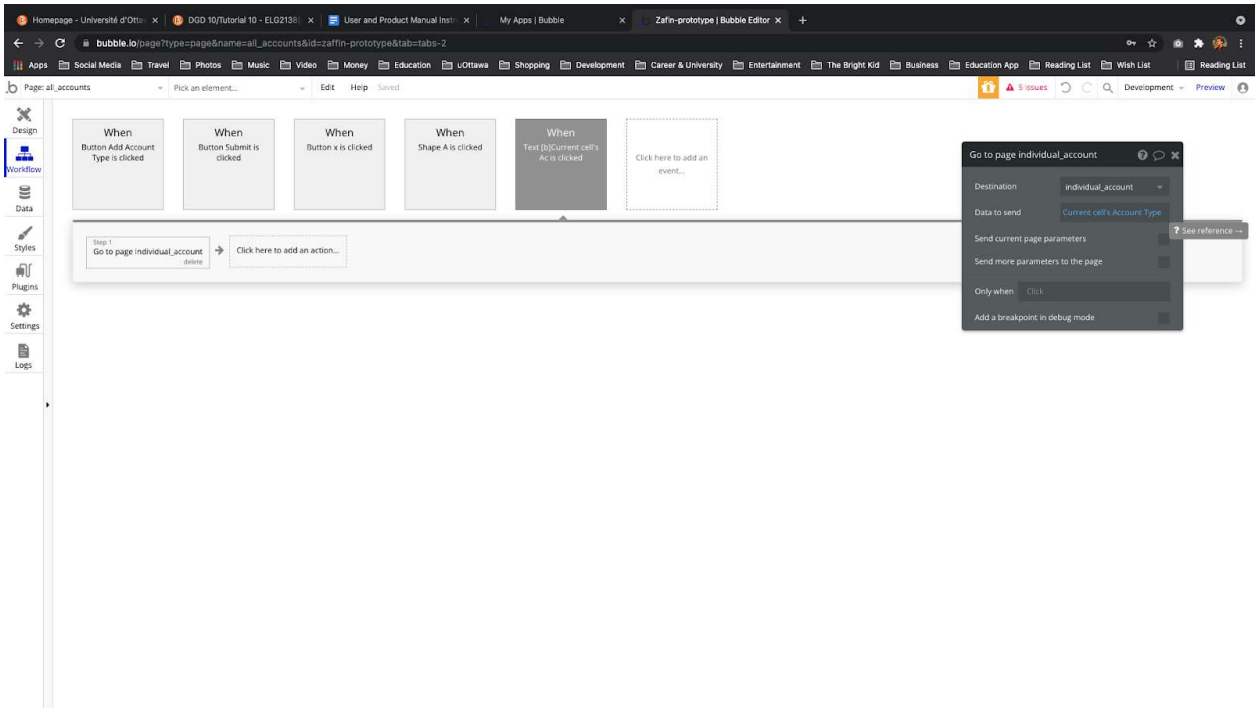


Figure 6.1.3.3: Added a click workflow and pointed to the point accumulation rate page

Once the listener is created we can now focus on the Point Accumulation Page. Given that the page has access to which Account Type was requested we can do a DB search and pull up all accumulation rates for the specified account and list them out in this page.

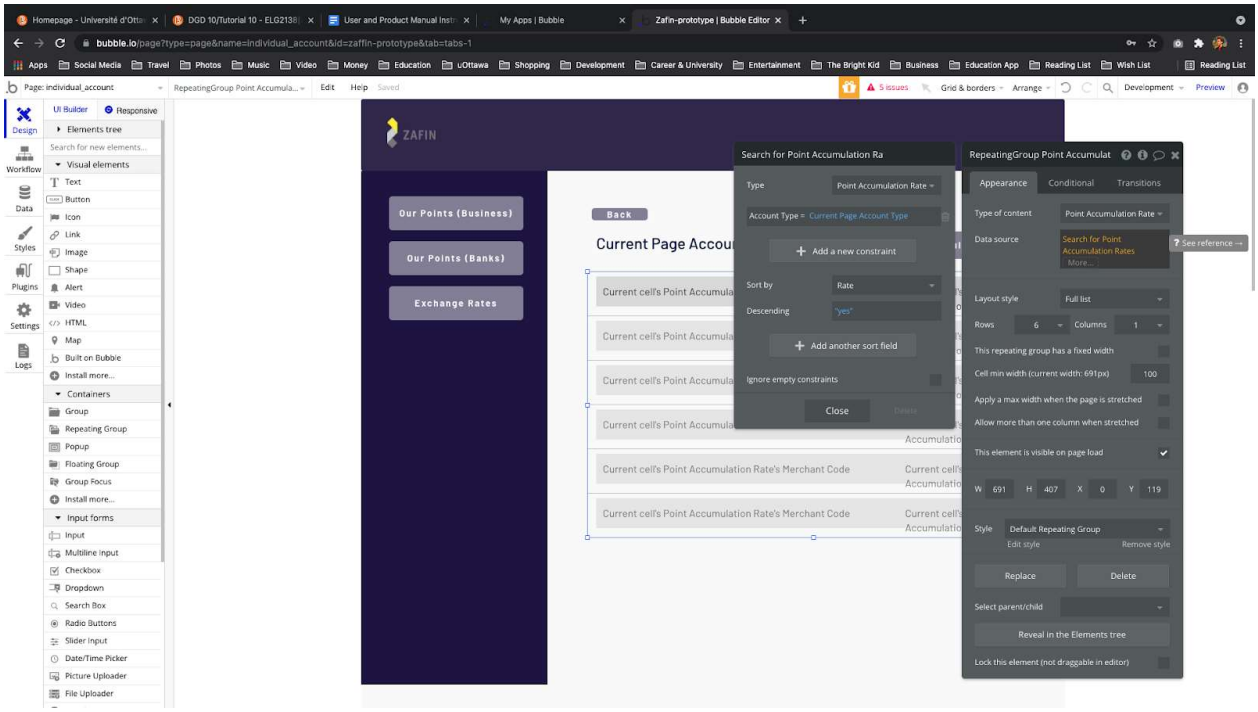


Figure 6.1.3.4: DB Lookup for all point accumulation rates filtered by the current account type

We can now implement the “adding” functionality by using a popup modal hooked up to a button.

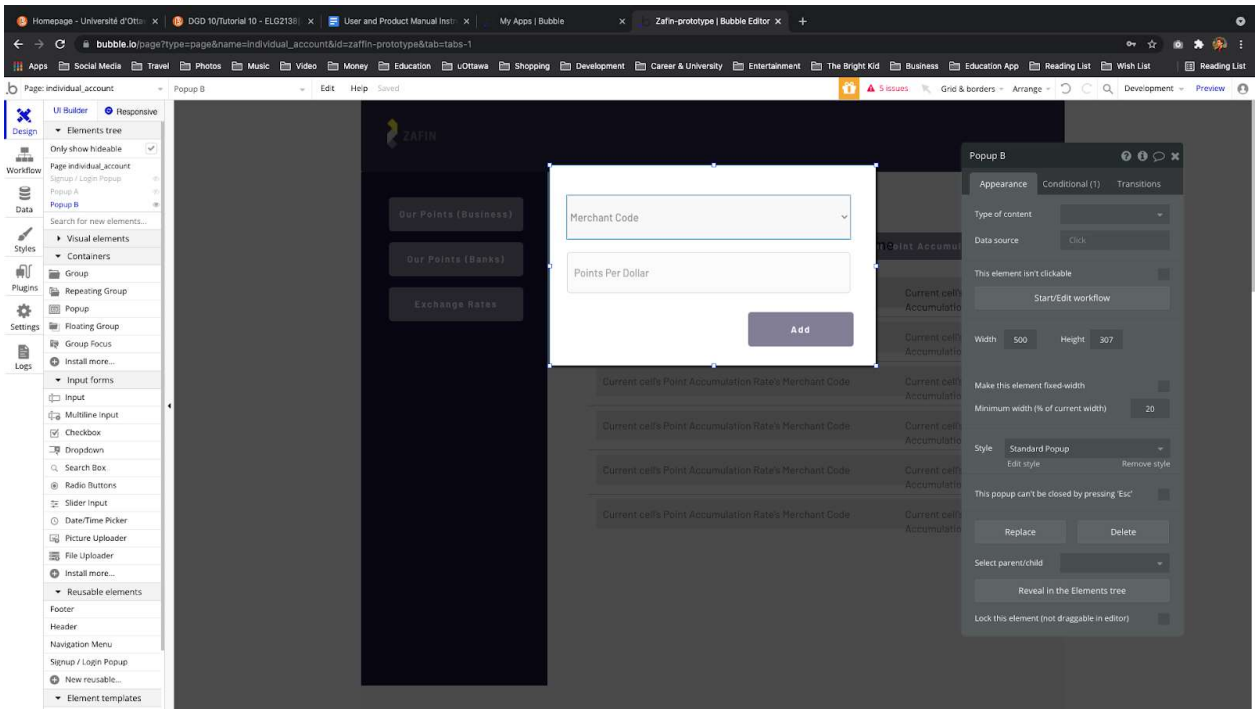


Figure 5: Adding a Point Accumulation Rate

We then follow the same pattern for Businesses and instead of merchant code use the

6.2 Exchange Rates

6.2.1 BOM (Bill of Materials)

- Bubble: The best way to build web apps without code | Bubble Cost: Free

6.2.2 Equipment list

- Computer
- Internet connection

6.2.3 Instructions

We continue from subsystem 1 by defining our new data structure

- Exchange Rate
    - Number: Local Points
    - Number: Foreign Points
    - Has one: Local Account Type
    - Has one: Foreign Account Type
    - String: Status

We begin by adding two lists one for the current exchange rates (i.e. Exchange rates with status = approved) and one for available institutions which loads up all of the institutions in the DB except the ones we have a pending request or an existing exchange rate.

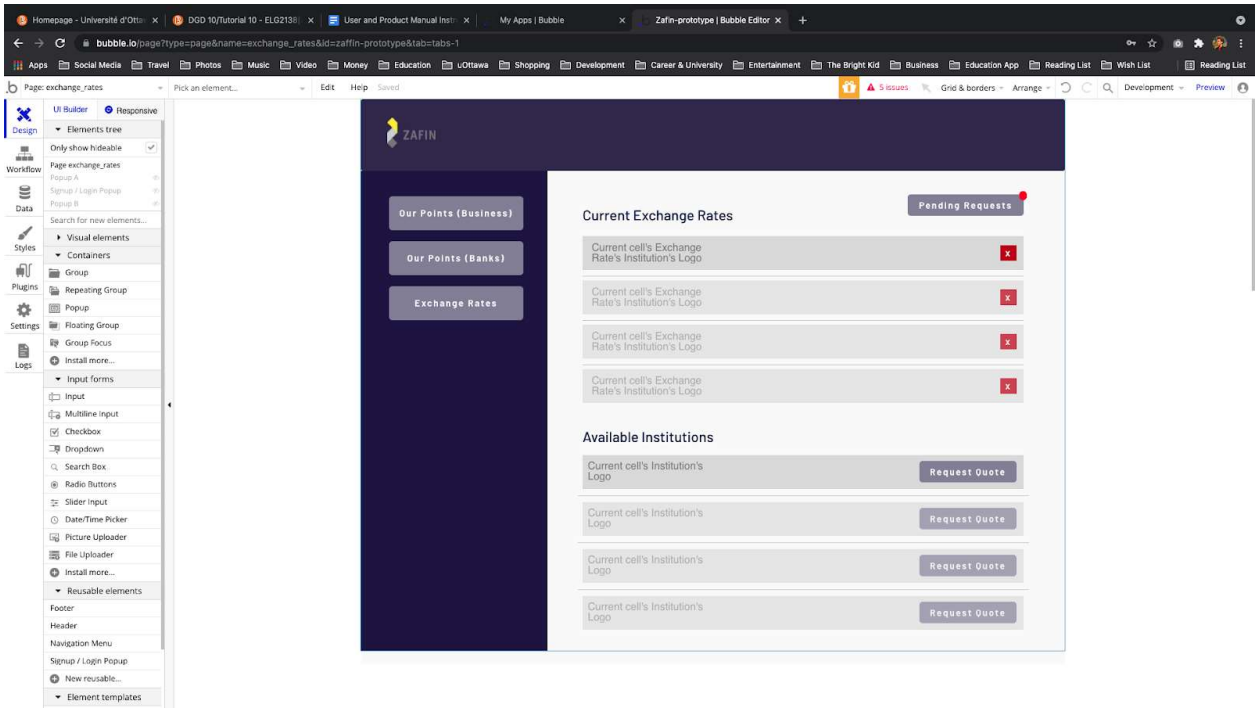


Figure 6.2.3.1: Current Exchange Rates and Available Institutions

We then add an event listener for the Request Quote button and make it open a modal with the following configurations:

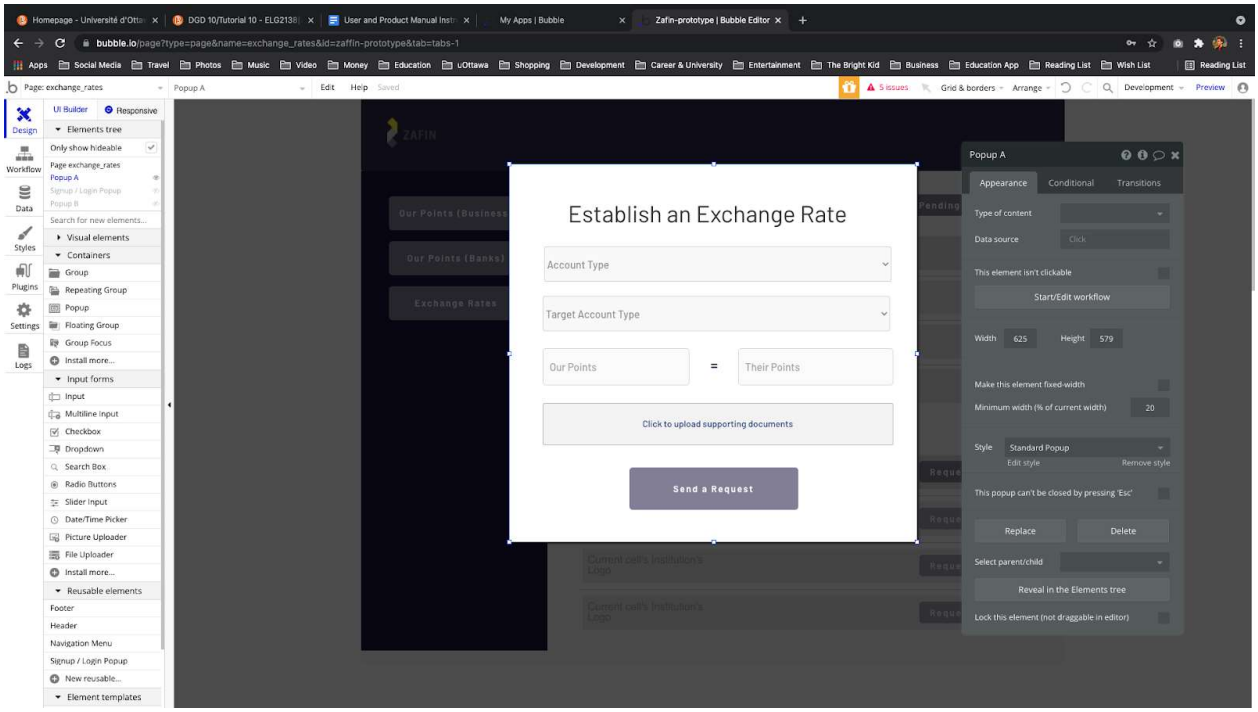


Figure 6.2.3.2: The modal dedicated to adding a new Exchange Rate

Once the submit button is clicked a new Exchange Rate record should be added to the DB with status being “pending”.

We now can focus on the recipient side. For the Pending Requests page we need to list out all of the Exchange Rates with status = pending that belong to the current user’s instutution.

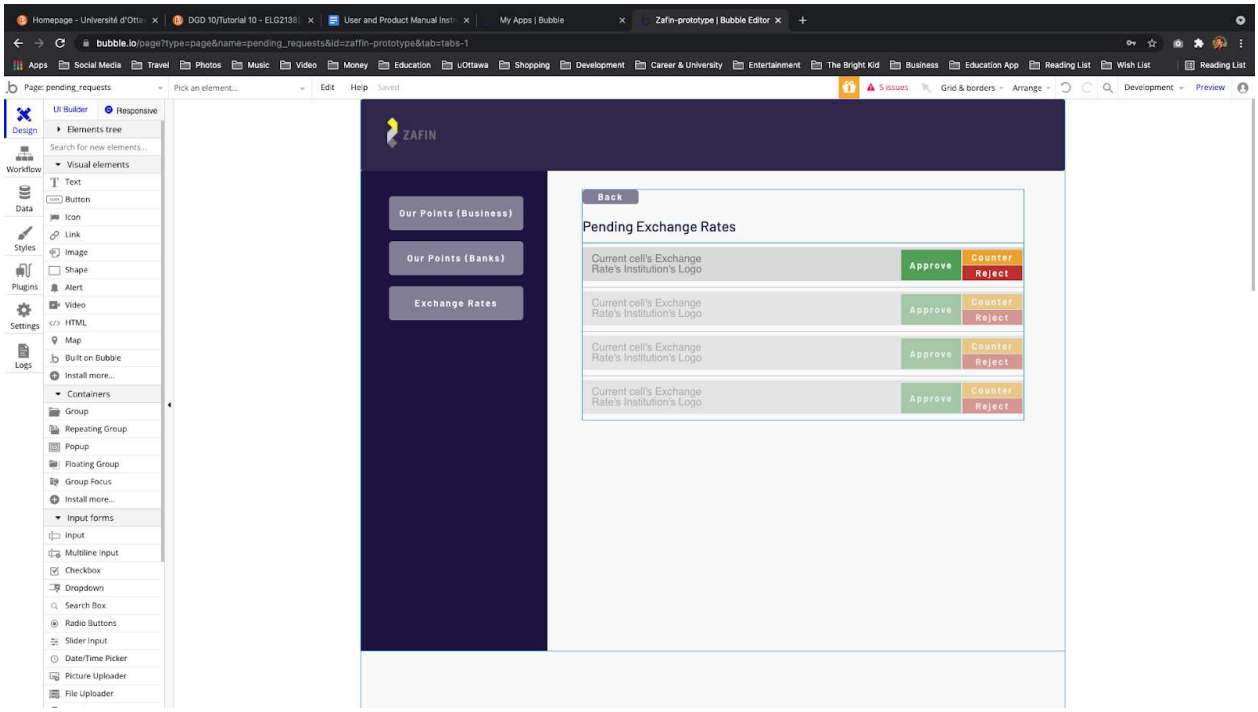


Figure 6.2.3.3: Pending Exchange Rates

Due to time limitations we were not able to complete this flow and were only able to show the pending requests. Ideally the next steps would be to add the approve, counter and reject mechanism.

6.3 Testing & Validation

For our final prototype, we conducted multiple tests to ensure that all of our subsystems were working properly. This included the exchange system and the base point systems.

The tests we conducted and their analyses were as follows:

(Table 6.1) Tests conducted, as well as their goals and passing criteria

Test #	Objective	Description	Passing criteria	Estimated	Conducting test

				test duration	
1	Logic flow	Test the look and feel of the points exchange section	• Ease of use • Modern interface	5 mins	Group members and or volunteers have sufficiently explored our prototype and given feedback
2	End to end test	Ensure everything in the point exchange section is properly linked. Check to make sure the whole prototype is still connected to the section	• If anything is properly linked • User has access to all required features	5 mins	Reach every portion of our app to make sure everything works, with a focus on the newly added section
3	User experience	Have users go through the point exchange section of the prototype and give feedback on the look and feel	• Use results to makes changes to the looks • Feel of the app to improve user experience	20 mins	Group members and or volunteers have sufficiently explored our prototype and given feedback
4	Request exchange rate	Have users request an exchange rate to ensure all information can be inputted correctly	• Successfully make an exchange between different financial institutions	15 mins	Ensure that all of the passing criteria is accessible and with a logical order
6	Accept exchange rate	Have users accept an exchange rate from another institution	• Successfully accept and apply an exchange rate from another	5 mins	Ensure that pressing the accept button updates exchange rate
7	Delete exchange rate	Have user delete an existing exchange rate	successfully remove an exchange rate	5 mins	Delete an exchange rate and make sure it disappears
8	Counter-offer	Have user counter offer an exchange rate request	Successfully counter offer	5 mins	Send a counter-offer and make sure that the offer updates

(Table 6.2) Results and analysis of our testing

Test Objective	**Results**	**Analysis**
Logic flow	Flows logically	No further steps required, the program's

		order makes sense
End to end test	Successfully navigates from end to end	All sections are connected and are accessible
User experience	“Pretty smooth”	The software has become more intuitive
Request exchange rate	All features work as intended, users can request exchange rates	No further steps required
Accept exchange rate	All features work as intended, users can accept exchange rates	No further steps required
Delete exchange rate	All features work as intended, users can delete exchange rates	No further steps required
Deleting accumulation rates	All features work as intended, users can delete accumulation rates	No further steps required

7 Conclusions and Recommendations for Future Work

Our team has learned a lot during the course of this project. We learned early on the importance of time management, as we found ourselves often scrambling to get deliverables finished on time. However, we started scheduling meetings much earlier and getting our work done ahead of time. Another skill we learned was how to distribute the work fairly between members. Not everyone has the same schedule and/or commitments, so some members may not be able to contribute as much one week, and may make up for it with extra work on a week that works better for them. The team learned how to properly manage conflict between members in order to keep the project moving along. Although our team was lucky enough to have minimal conflicts there were a few debates on what direction we would take the project. We learned to quickly resolve these small issues so as not to waste our already limited time. Our group found that it was most productive to work on every step together rather than divide it up into parts. We would get together for meetings and brainstorm together. This way, our work could flow much nicer without someone needing to piece all the parts together and smooth it over. The workload was not always even with this method, but it is not difficult to spot someone who isn’t carrying their weight in a group of five.

If we had a few more months to work on this project, our group would spend a lot more time on our final prototype, as we weren’t able to make it fully functional. Originally, our prototype was also supposed to include a campaigns feature to incentivise spending points, and a savings feature for users to save points and gain interest. The campaigns feature would basically just recommend to users rewards based on past purchases and redemptions as well as deals or events that users can spend their points at. That was a fairly basic idea, so our group decided not to

pursue it and focus on our more original ideas due to lack of time. The points savings feature was meant to be a savings account for people's points. This way, they would be able to gain interest on their points so they can save up for something that costs a lot. Zafin told us that the reason a savings account exists is because the banks have use for the money in that account, but they have no use for points. They said it would be an interesting feature if we found a way the banks could use those points, but again, we lacked the time to split our focus.

8 Bibliography

G2 Search: stripe

How the PC Optimum program works | Ratehub.ca

How the Scene Rewards Program Works | Ratehub.ca

How does the Air Miles rewards program work| Ratehub.ca

PC Optimum

https://www.scene.ca

Get rewarded for your everyday shopping | AIR MILES

Stripe Reviews - 180 Reviews of Stripe.com | Sitejabber

Stripe Reviews | Read Customer Service Reviews of stripe.com (trustpilot.com)

APPENDICES

9 APPENDIX I: Design Files

All documents related to the production and testing of our prototype can be found on our MakerRepo page linked below.

This project was inspired by and created for the course GNG1103 at the University of Ottawa and was supervised by David Knox, who was our instructor throughout the term.

Table 3. Referenced Documents

Document Name	Document Location and/or URL	Issuance Date
Bubble Prototype	https://zaffin-prototype.bubbleapps.io/version-test/business	Nov 11 2021
MakerRepo	https://makerepo.com/IsaacW/955.zafin-ecosystem-a11	Nov 11 2021