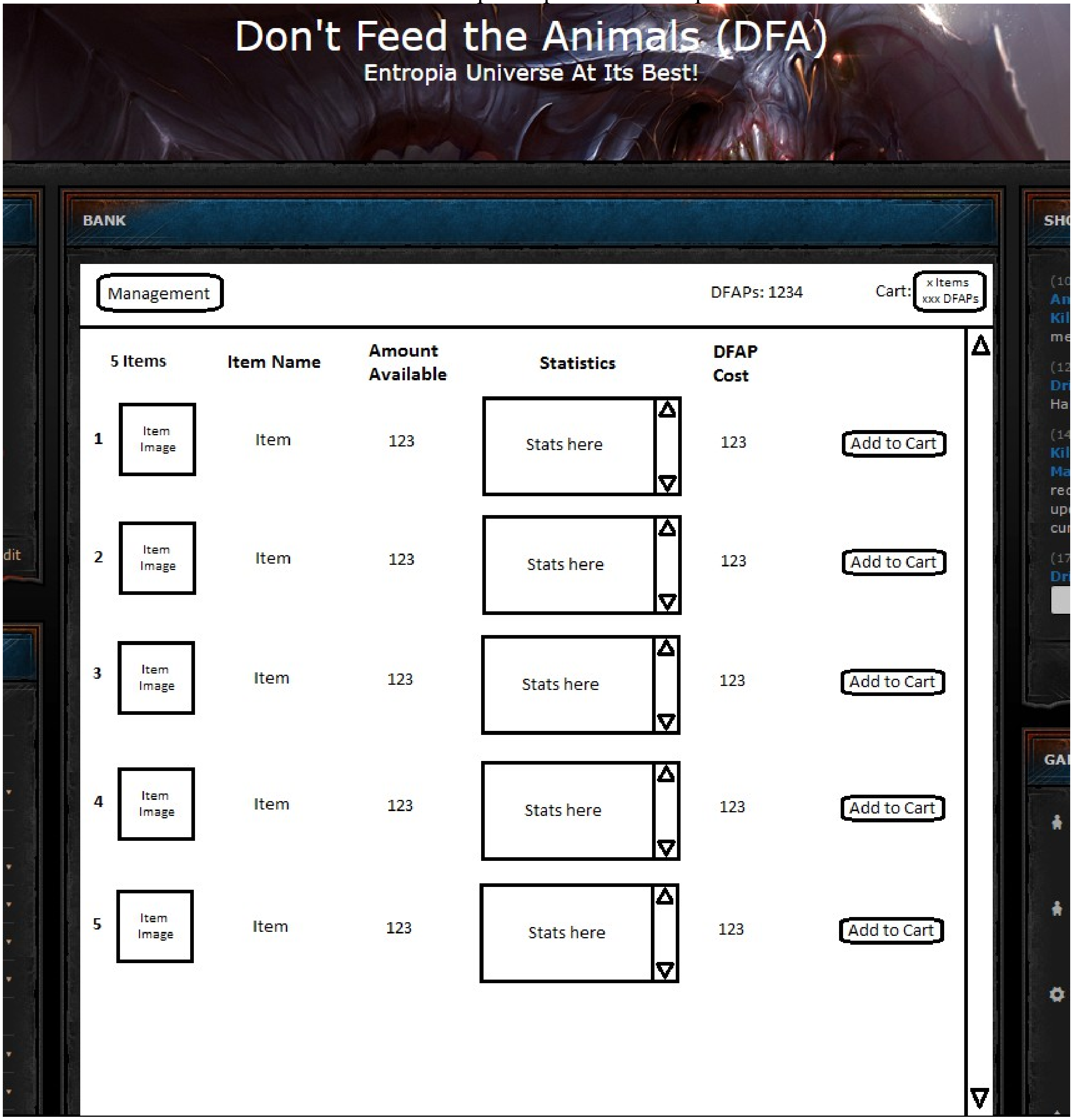
Concept design for DFA Shop Web page design.

Below would be the home view when a user opens up the DFA Shop:



Don't Feed the Animals (DFA)

Entropia Universe At Its Best!

BANK

Management

DFAPs: 1234

Cart: x Items xxx DFAPs

	5 Items	Item Name	Amount Available	Statistics	DFAP Cost	
1	Item Image	Item	123	Stats here	123	Add to Cart
2	Item Image	Item	123	Stats here	123	Add to Cart
3	Item Image	Item	123	Stats here	123	Add to Cart
4	Item Image	Item	123	Stats here	123	Add to Cart
5	Item Image	Item	123	Stats here	123	Add to Cart

- Management button will only show for a person able to do management of the items and DFAPs. This would include add/modify/remove user info such as setting user name, DFAPs available to user etc. This would also be where the manager can add/modify/remove available items in the shop. This is also where the manager would full fill the delivery status of a purchased item or readd it back into the shop.

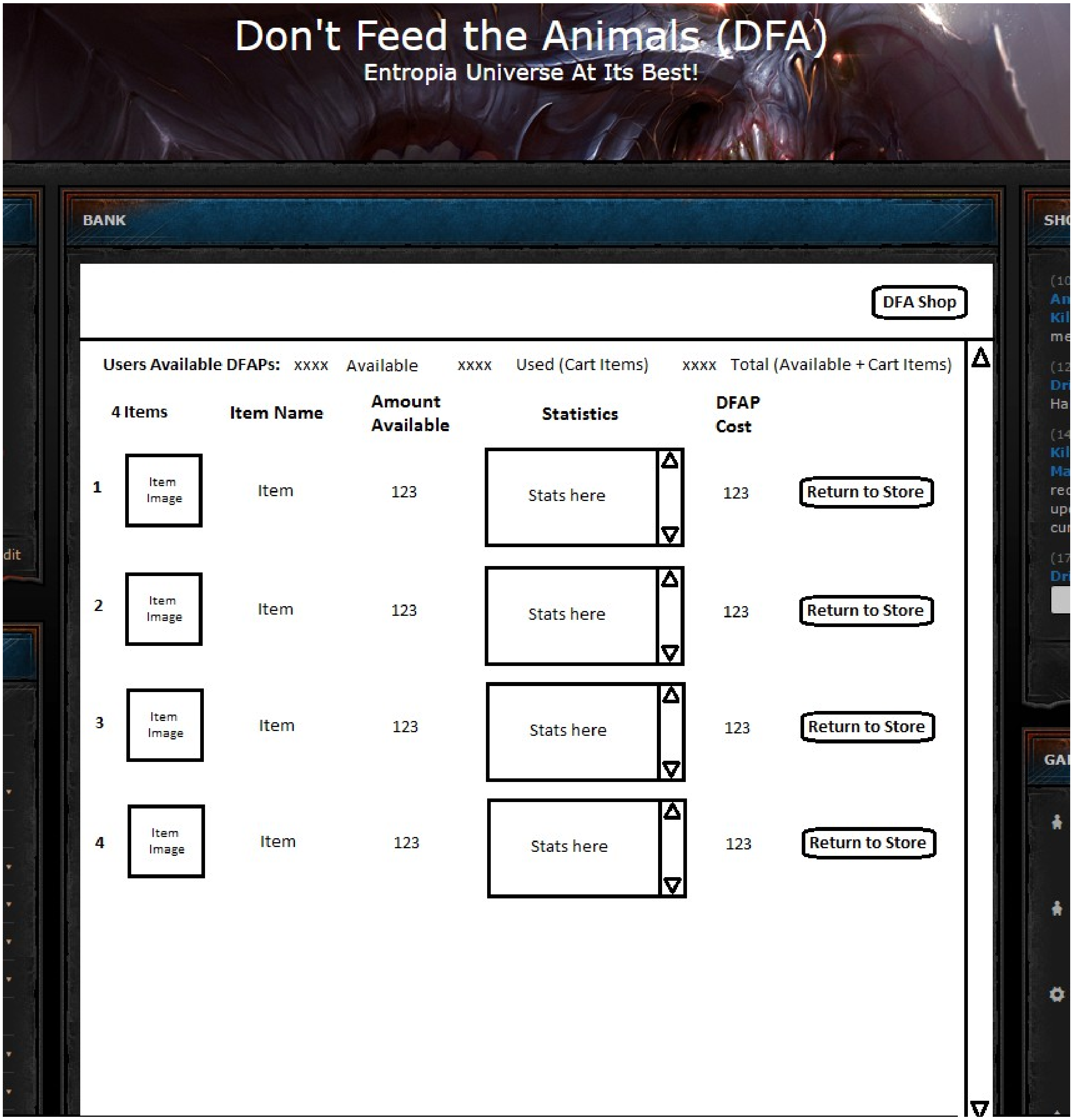
- Cart button will be available for the user to see what they currently have in their cart to be delivered, it will also allow them to remove items from their cart if they don't want it and it will go back into the shop listing.

- Add to Cart button will add a single instance of an item to the persons cart.

The middle section of the screen is the listing of all the items in the shop, it is an alphabetical list of all the items based on item name. There can be an image attached by the manager when putting the item into the shop, statistics of the item or any useful data can be place in the statistics section, the amount available in the shop is listed and the going value of that particular item is listed.

This is a DFAP only shop currently, later it maybe upgraded to accept DFAP or PED, DFAP should be slightly cheaper then PED purchases, this is to promote participation in society driven events to gain DFAPs.

Below is the Cart View:



Don't Feed the Animals (DFA)

Entropia Universe At Its Best!

BANK

DFA Shop

Users Available DFAPs: xxxx Available xxxx Used (Cart Items) xxxx Total (Available + Cart Items)

	4 Items	Item Name	Amount Available	Statistics	DFAP Cost	
1	Item Image	Item	123	Stats here	123	Return to Store
2	Item Image	Item	123	Stats here	123	Return to Store
3	Item Image	Item	123	Stats here	123	Return to Store
4	Item Image	Item	123	Stats here	123	Return to Store

The Cart view hold a particular person cart information along with their balance information.

The user is able to place items back into the store if he does not want the item anymore by pressing the “Return To Store” button. If there are multiples of the item it will pop up a windows asking how many to return to the store, by default it returns only 1 item but will allow the user to enter how many up to the total in his cart to have returned to the store.

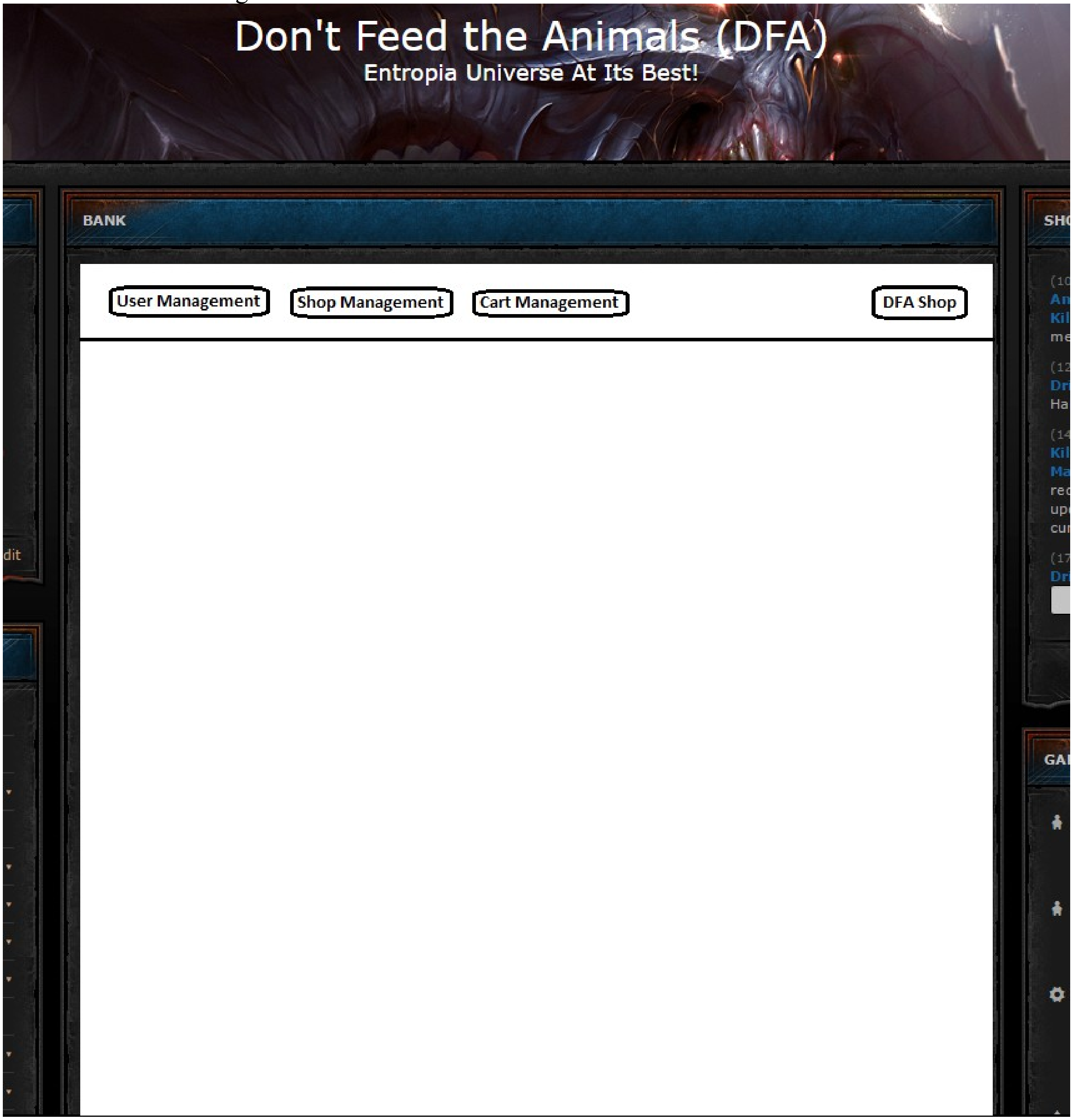
The balance information shows amount of DFAPs left not including items in cart (Available), the balance of items in Cart (Used) and the total of both (Total).

The person also has an option to return to the “DFA Store” at the top by pressing the “DFA Shop” button.

Below are the Management Screens

These screens will only be accessible by administrators that manage the Shop and DFAPs for users.

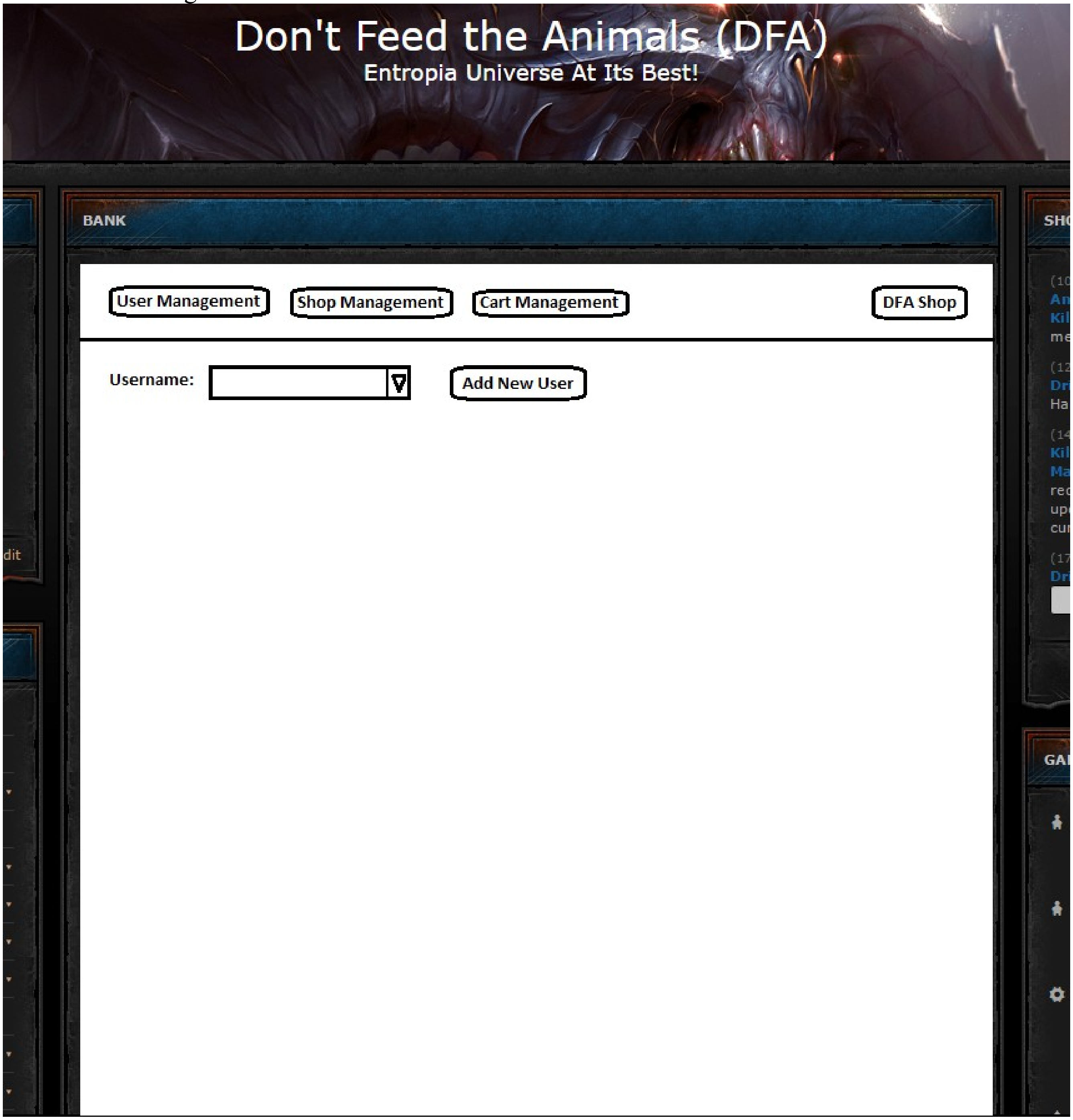
Here is the Base Management Screen:



There are 3 types of Management that can be performed by pressing the corresponding buttons. “User Management”, “Shop Management”, and “Cart Management”.

Each will be explained next in the order listed above.

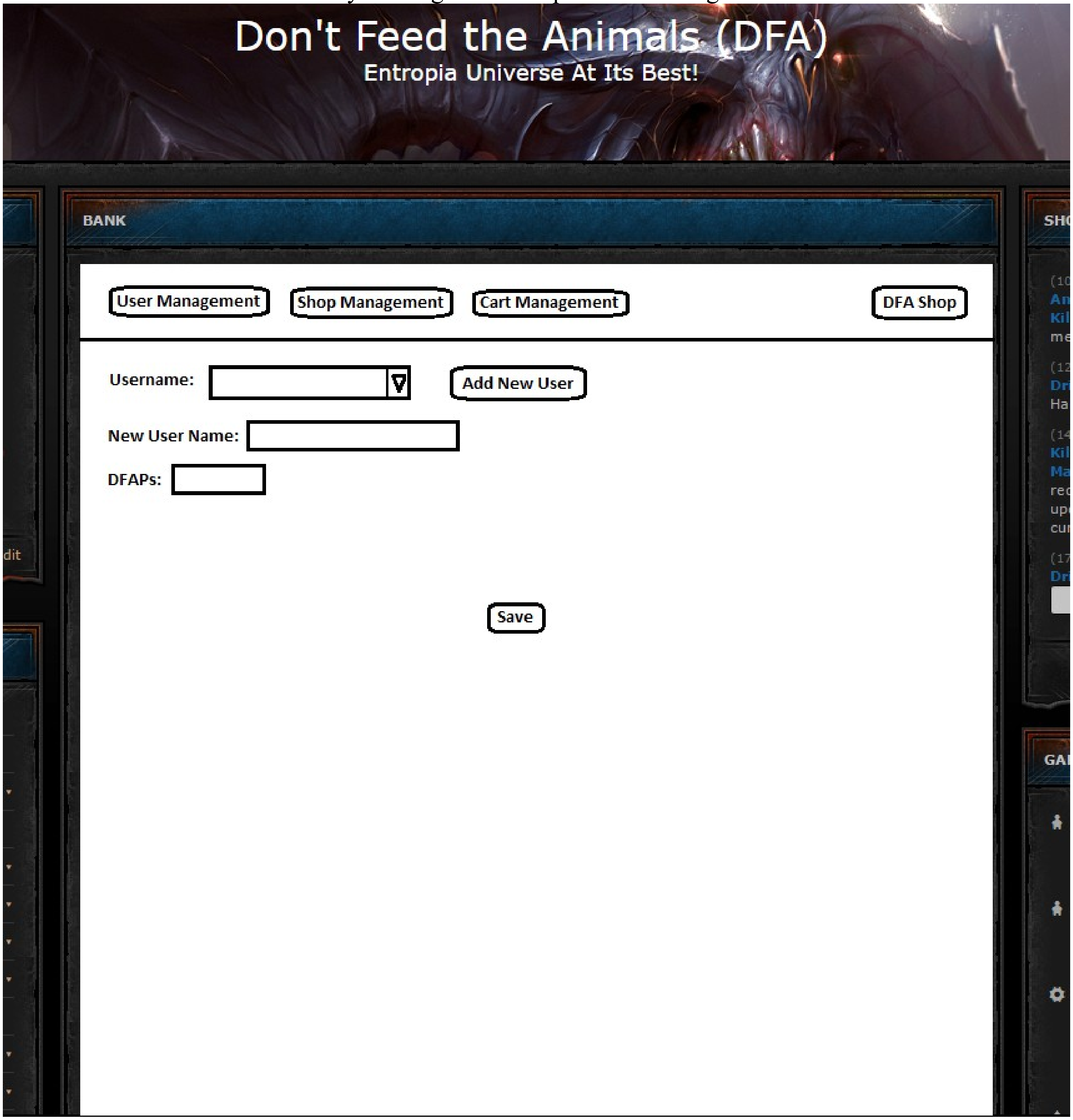
Initial User Management Screen:



This form will have a list of all users registered to use the store. If a User is not in the store an administrator can Add a user manually.

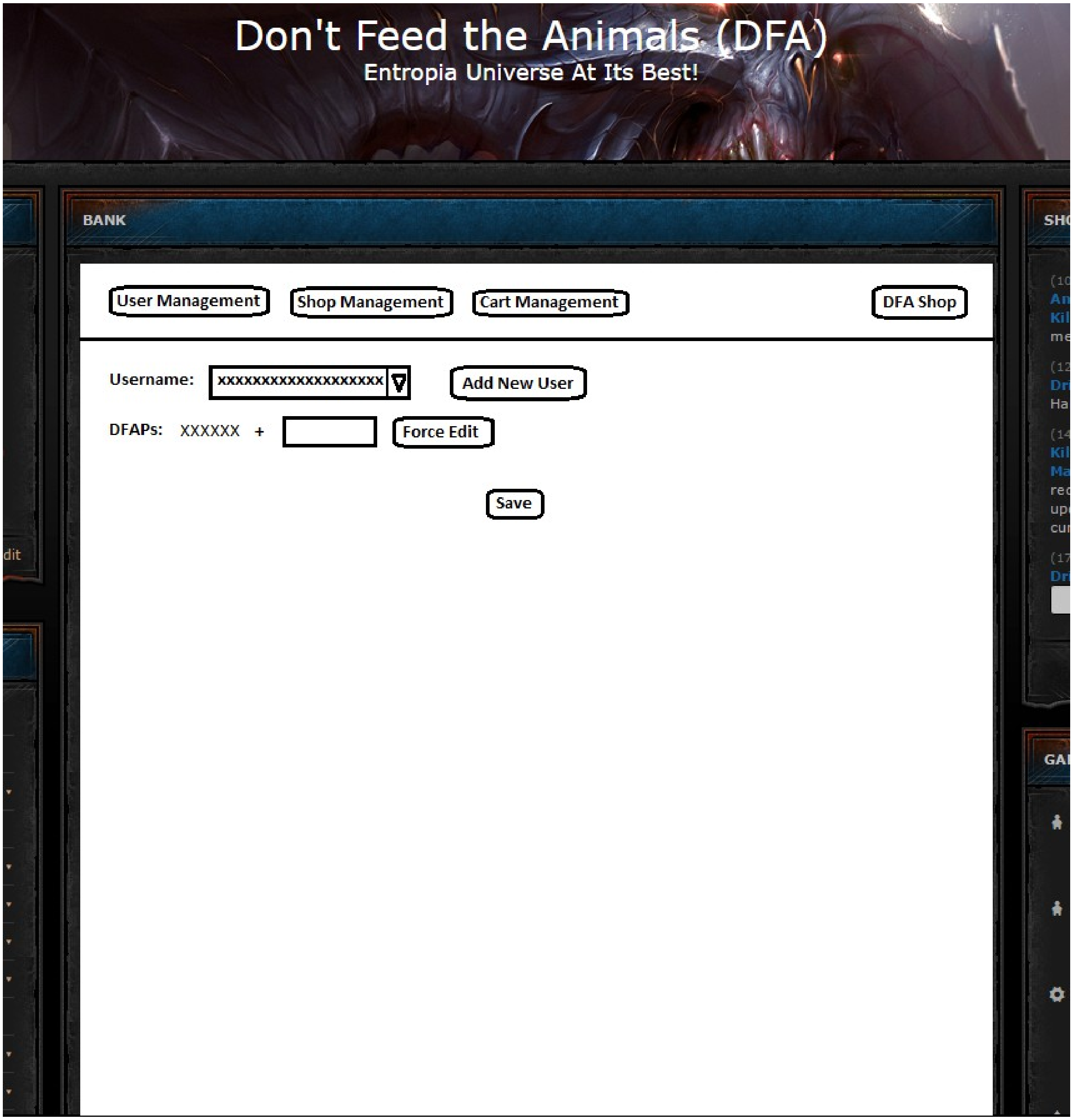
The Add user ability maybe written out of the project depending on ability to use the shivtr.com API to grab registered members instead of managing them within this app manually. (Note: I may be able to use this API and the login token to handle authentication as well. So there would be no need to build a separate authentication model for this application. This is all still in concept review stage.)

The next image is simple Add User Form to add a new user to the system if I am unable to get the API to work. This is the form that may no longer exist as part of the design.



The next image is is what will be used to edit user information. By default it will want the administrator to only Add new DFAP to a users account. If the Admin however needs to force set the DFAP for a user he can do so using the Force Edit Button.

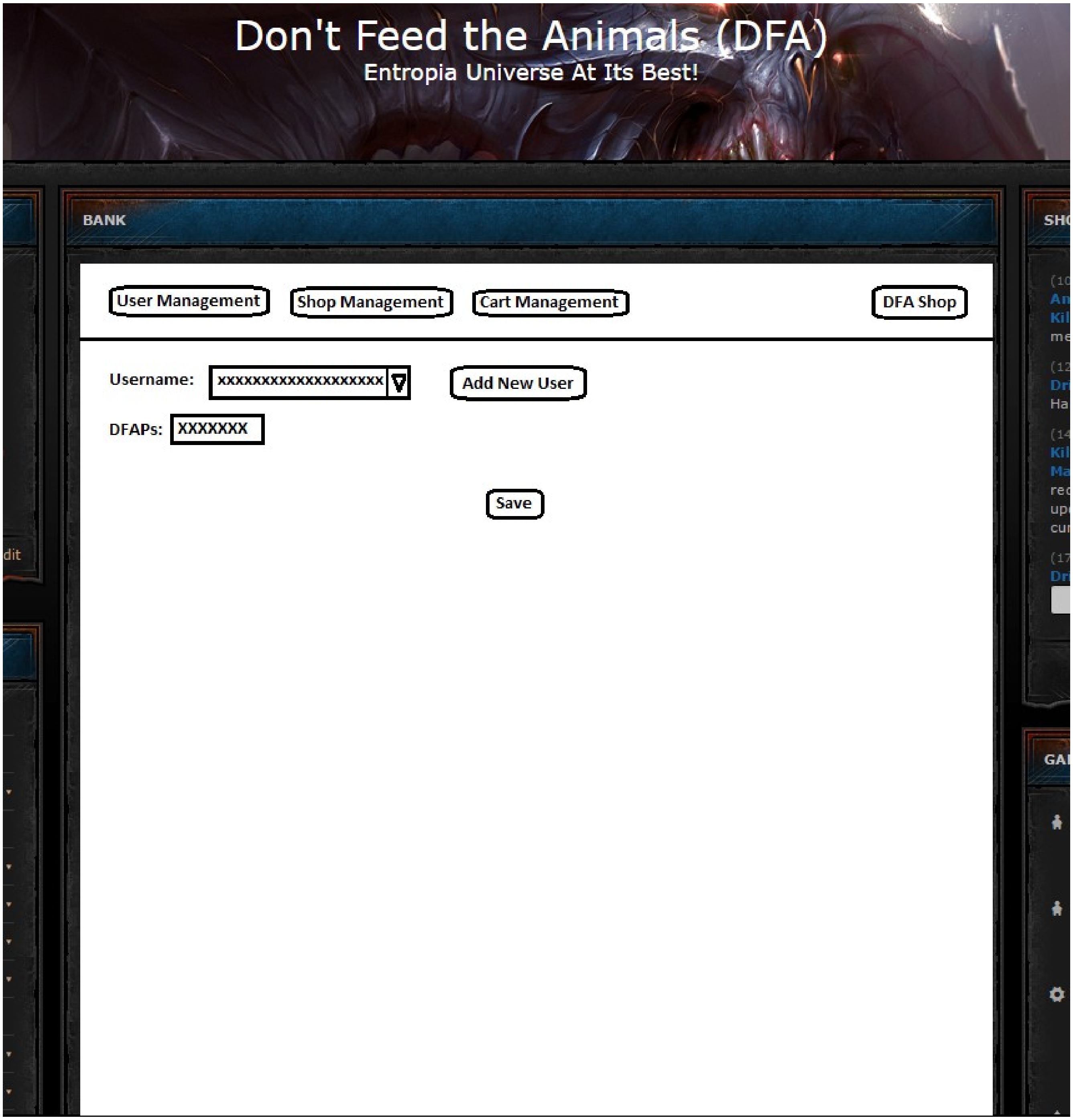
The Save button will save the information for a particular selected user.



Username: xxxxxxxxxxxxxxxxxxxx

DFAPs: XXXXXX +

The Next image is example of the Force Edit screen for the DFAPs.



Username: xxxxxxxxxxxxxxxxxxx

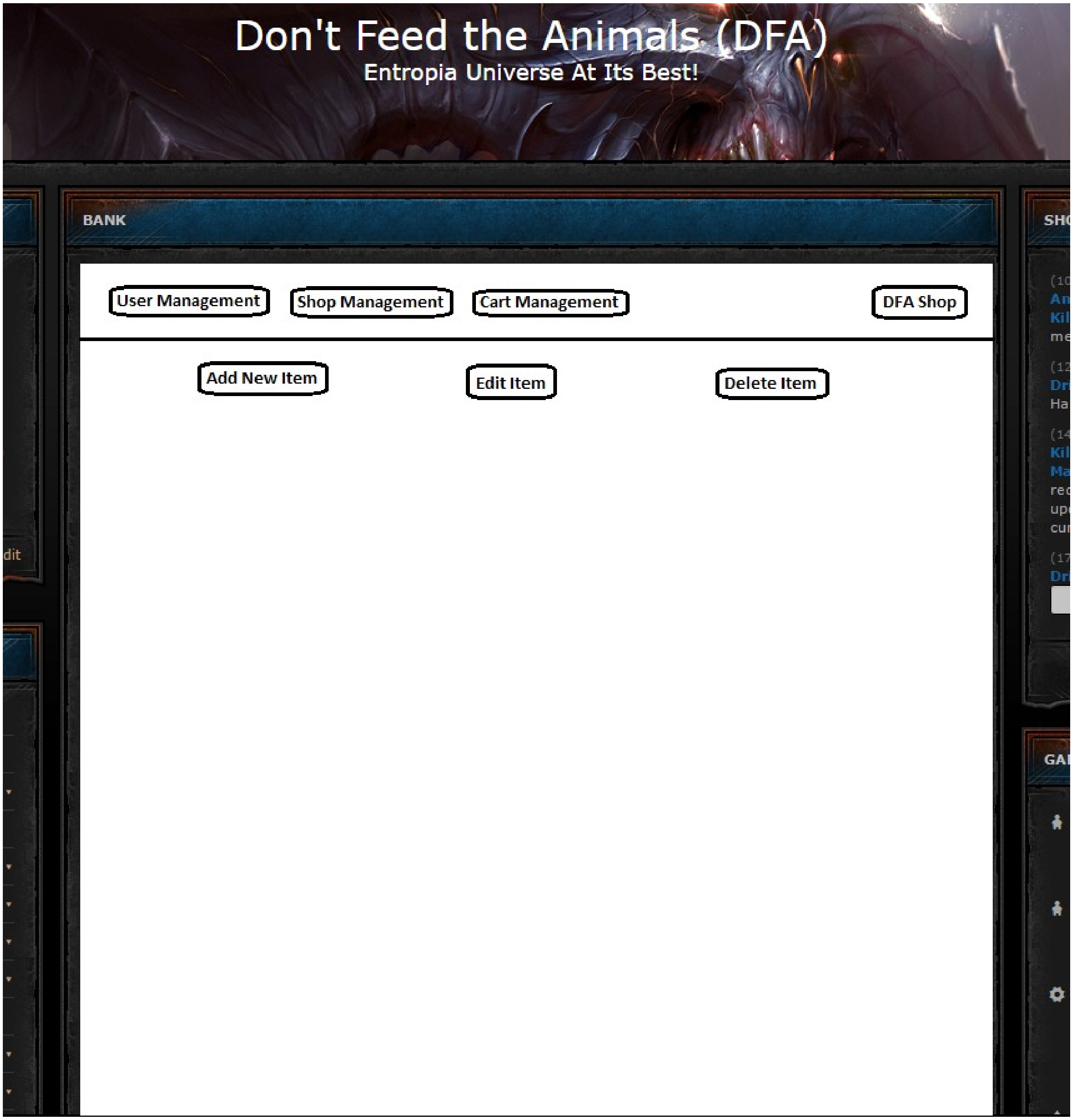
DFAPs: XXXXXXX

It will display the users available DFAPs and the Admin can change it accordingly and save the changes.

Next Section is the Shop Management:

This will display 3 new buttons for the Admin to use, “Add New Item”, “Edit Item”, and “Delete Item”

Note: Some of the following screens could be condensed maybe so Add Item is at top and list of all items in shop are listed right away with Edit and Delete buttons off to the right side.

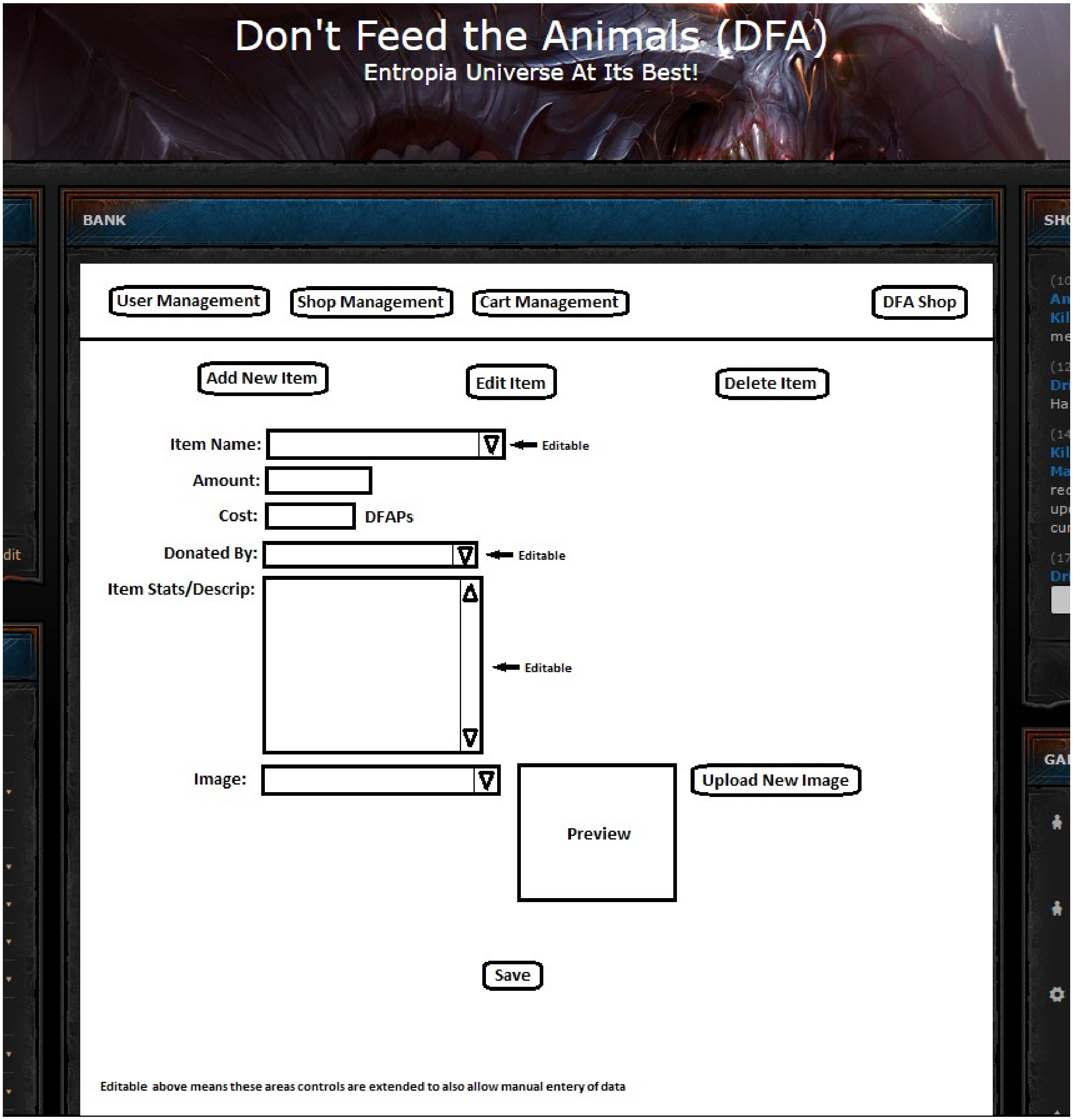


The Next image is an concept of what the Add New Item and Edit screens would looks like.

Add New Item all default values would be blank, Edit would have populated data for the item that can be changed.

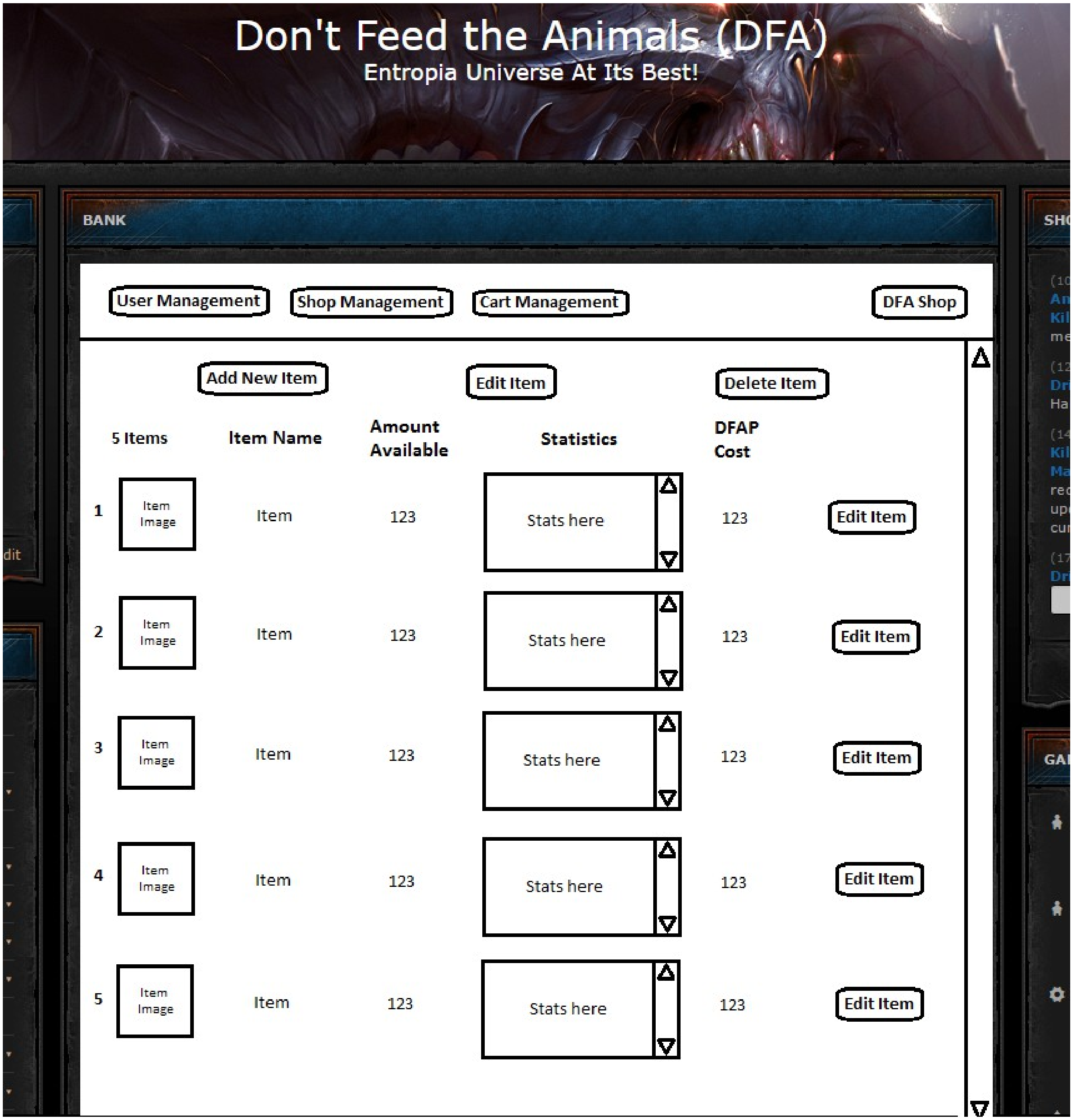
Drop Down items would list items that were already added before and default information could be pulled automatically and edited accordingly for the new item as desired.

Save would save the item and add it to the Store inventory.



Editable above means these areas controls are extended to also allow manual entery of data

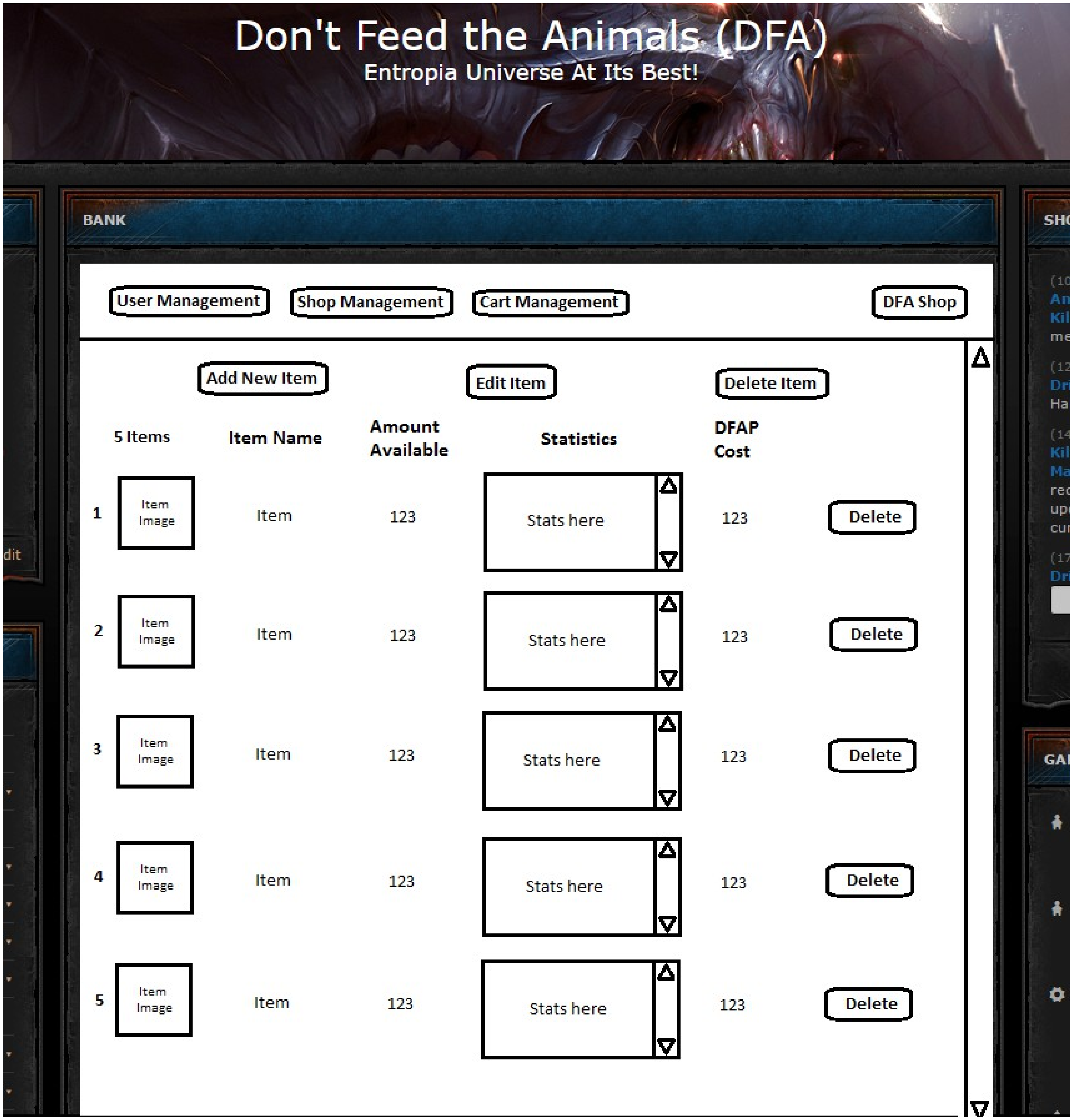
The Next two screens are what could be condensed, and be part of the default display of the shop management screen.



Don't Feed the Animals (DFA)
Entropia Universe At Its Best!

BANK

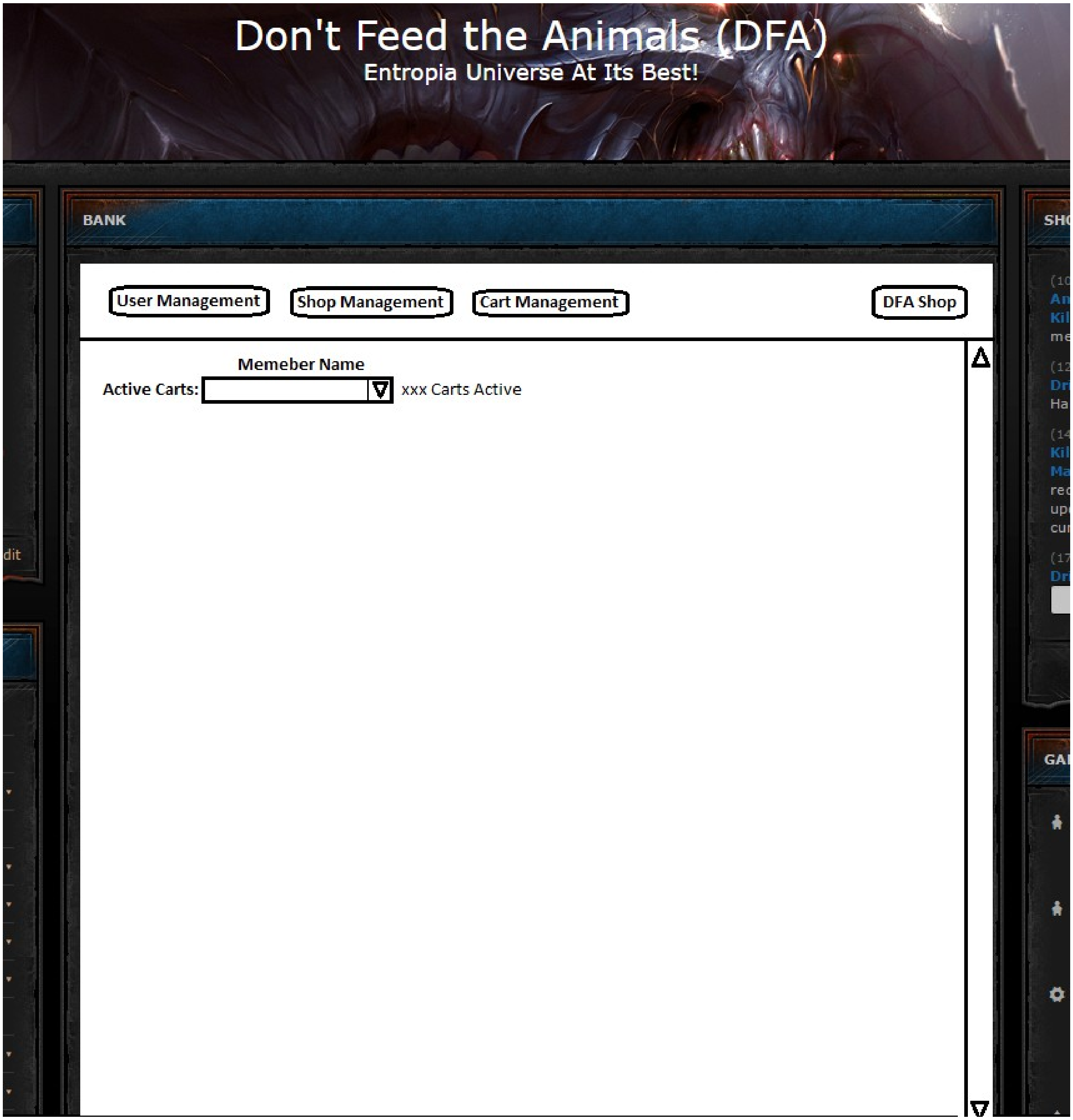
	5 Items	Item Name	Amount Available	Statistics	DFAP Cost	
1	Item Image	Item	123	Stats here	123	Edit Item
2	Item Image	Item	123	Stats here	123	Edit Item
3	Item Image	Item	123	Stats here	123	Edit Item
4	Item Image	Item	123	Stats here	123	Edit Item
5	Item Image	Item	123	Stats here	123	Edit Item

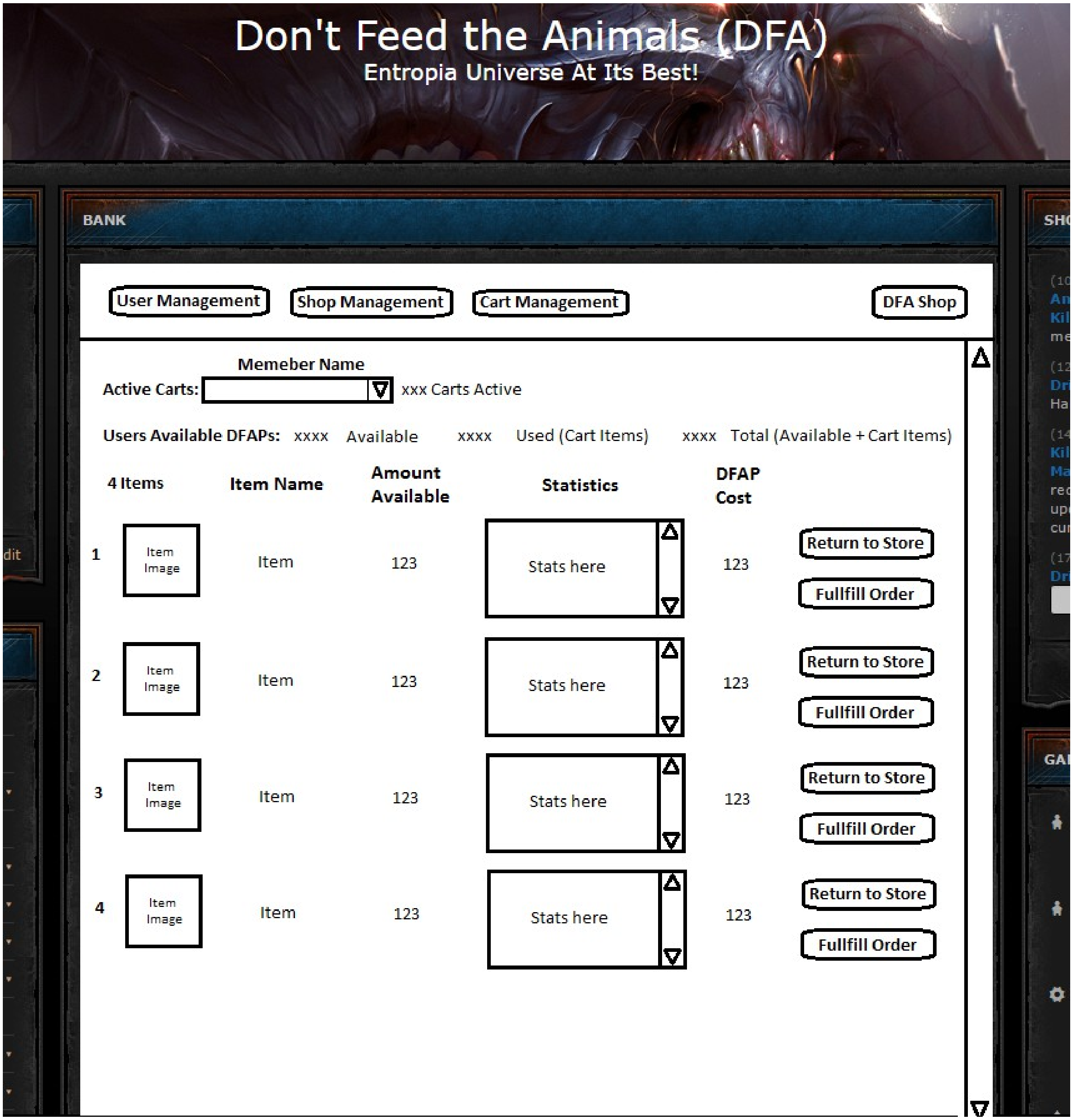


	5 Items	Item Name	Amount Available	Statistics	DFAP Cost	
1	Item Image	Item	123	Stats here	123	Delete
2	Item Image	Item	123	Stats here	123	Delete
3	Item Image	Item	123	Stats here	123	Delete
4	Item Image	Item	123	Stats here	123	Delete
5	Item Image	Item	123	Stats here	123	Delete

Next are examples of the Cart Management Screens:

The first thing an administrator will need to do is review members with active carts. When the member with active item in cart is located the Admin can push the item back to the store or set the transaction as completed once the item is delivered.


Don't Feed the Animals (DFA)
Entropia Universe At Its Best!
BANK
User Management
Shop Management
Cart Management
DFA Shop
Memeber Name
Active Carts:
xxx Carts Active



Memeber Name

Active Carts: xxx Carts Active

Users Available DFAPs: xxxx Available xxxx Used (Cart Items) xxxx Total (Available + Cart Items)

4 Items		Item Name	Amount Available	Statistics	DFAP Cost	
1	Item Image	Item	123	Stats here	123	Return to Store Fullfill Order
2	Item Image	Item	123	Stats here	123	Return to Store Fullfill Order
3	Item Image	Item	123	Stats here	123	Return to Store Fullfill Order
4	Item Image	Item	123	Stats here	123	Return to Store Fullfill Order

Returning the item back to the store will return the DFAPs to the users Available amount, reduce the used amount and place item back into the store.

Fullfilling the Order will decrease the Used amount, Update the Total.

Background Actions and other Notes:

Though out the actions all data will be logged and archived so if really needed the system administrator of the site can do a roll back on the data and fix a mess up if one occurs.

Items are not removed from the store until an order is Full filled it is only tagged as in a cart so it does not show as available until it is placed back into the store.

In the future DFAPs will be managed by the DFAP Bank Manager application but until then DFAPs are managed manually by the active secretary in the society, it is his responsibly to keep this number up to date with values in the on going spreadsheet on google.

I will be writing the application using Python with Flask (MVC Framwork) as the restful web service. I will be use PostgreSQL as the database engine where all the data will be stored. I plan to host the final product on a Public and free PaaS Server https://www.heroku.com/ is one place I could be hosting it on. For development purposes I am hosting it on my local Raspberry Pi units, so it maybe somewhat slow but for development it is all that is needed.

I am investigating the use of the SHIVTR API to handle member lists and credential management to enforce logins. I do not want to have to build a login service for this application, should be able to use shivtr's authentication tokens and and API to perform any authentications, much more secure as shivtr is only place where any login is performed.

I will be hosting all the documentation and code on github: https://github.com/KillariousMaximillion/DFAStore it is all public so unless you wish to contribute to the project editing you should not need to create logins or require me to give you access to the data.

I am planning on setting up a JIRA issue tracking server where I can track issues and tasks that need to be done, so if anyone wants to contribute they can, they can pull tasks and work on anything they wish. Any work will be reviewed prior to going into production to ensure integrity of the code to be working.