Heirarchy

The heirarchy of the data sent is in order of groups, then by individual table attributes in order of how the attributes are set up in the sql database.

For example:

$$First:\ Groups$$
$$Users$$
$$GroupDebt$$
$$PersonalDebt$$
$$Groceries$$
$$Chores$$
$$Last:\ Rent$$

The rest of the hierarchies are as follows:

Groups	Users	Group Debt	Personal Debt	Groceries	Chores	Rent
group id	user id	debt id	debt id	grocery id	chore id	rent id
group name	first name	debt type	debt type	amount	name	amount
ground rules	last name	amount	amount	paid status	due date	paid status
	email	group id	group id	user id	chore status	user id
	password		group id		user id	group id
	phone				group id	
	facebook					
	twitter					
	instagram					
	group id					

If in the event there is a confliction, the tables are of a higher priority than the attributes. For example: If both a group and a group id have to be sent the order would be group id, group name, ground rules, debt id, debt name, amount, group id. There are two group id's that have to be sent because the group table has a group id and the group debt has a group id. If the group id and user id were to be sent from the groups table and the users table respectively, the order would be: group id , user id since the gorup id is fromt the groups table.

Methods

The list of methods in serverRoutes.js is below in addition to what sort of data will be sent in addition to the data that will be recieved. Define "processed?" as some sort of return code to be interpreted as either a successful or an unsuccessful request.

If there is an attribute that is at the beginning of the attributes that is also in the url definition allow that to be ignored as an object in the attribute list. (ie "api/groups/groupid" has group id sent in both GET and DELETE but since it is in the url it will not be sent as an attribute. However that same url will have group id sent as an attibute when using PUT since that group id belongs to the 'new' group info properties rather than the current group info property)

api/groups
-Method Name: create new group (post)
-Send: group name
-Receive: group id

api/groups/groupid
-Method Name: get group info (get)
-Send: group id
-Receive: group name, users, groups debt, group id

-Method Name: update group info (put)
-Send: group name, users, groups debt, group id
-Receive: processed?

-Method Name: delete group(delete)
-Send: group id
-Receive: processed?

api/users
-Method Name: create new user (post)
-Send: first name, last name, email, password, phone, facebook, twitter, instagram, group id
-Receive: processed?

api/users/userid
-Method Name: get user info (get)
-Send: user id
-Receive: first name, last name, email, password, phone, facebook, twitter, instagram, group id

-Method Name: update user info (post)
-Send: first name, last name, email, password, phone, facebook, twitter, instagram, group id
-Receive: proceessed?

-Method Name: delete user (delete)
-Send: user id
-Receive: proceessed?

api/groupdebts/groupid
-Method Name: get group debt list (get)
-Send: group id
-Receive: debt type, amount, group id

-Method Name: create group debt(post)
-Send: group id, debt type, amount
-Receive: processed?

api/groupdebts/groupid/debtid
-Method Name: get group info (get)
-Send: group id, debt id
-Receive: debt type, amount

-Method Name: update group debt (put)
-Send: group id, debt id, debt type, amount
-Receive: processed?

-Method Name: delete group debt (delete)
-Send: group id, debt id
-Receive: processed?

api/personaldebts
-Method Name: create new personal debt (post)
-Send: user id, amount, lender id, borrower id
-Receive: processed?

api/personaldebts/userid
-Method Name: list personal debts (get)
-Send: user id

-Receive: amount, lender id, debt id (list)

api/personaldebts/userid/debtid
-Method Name: get personal debt info (get)
-Send: user id, debt id
-Receive: amount, lender id, debt id

-Method Name: update personal debt (put)
-Send: user id, debt id, amount, lender id, borrower id
-Receive: processed?

-Method Name: delete personal debt(delete)
-Send: user id, debt id
-Receive: processed?

api/groceries/groupid
-Method Name: get grocery list (get)
-Send: group id
-Receive: amount due, paid status, user id, group id

-Method Name: update grocery list (put)
-Send: group id
-Receive: amount due, paid status, user id, group id

-Method Name: create new grocery item (post)
-Send: group id, amount due, paid status, user id
-Receive: processed?

-Method Name: delete grocery list (delete)
-Send: group id
-Receive: processed?

api/groceries/groupid/groceryid
-Method Name: get grocery item (get)
-Send: group id, grocery id, item name
-Receive: item name, amount due, paid status, user id

-Method Name: update grocery item (put)
-Send: gropu id, grocery id, item name, amount due, paid status, user id
-Receive: processed

-Method Name: delete grocery item (delete)
-Send: gropu id, grocery id, item name
-Receive: processed

api/chores/groupid
-Method Name: get chores list (get)
-Send: group id
-Receive: amount due, paid status, user id, group id

-Method Name: update chores list (put)
-Send: name, due date, chore status, user id, group id
-Receive: processed?

-Method Name: create new chore (post)
-Send: name, due date, chore status, user id, group id
-Receive: processed?

-Method Name: delete chores list (delete)
-Send: group id
-Receive: processed?

api/chores/groupid/choreid
-Method Name: get chore info (get)
-Send: group id, chore id
-Receive: chore name, due date, chore complete, user id, group id

-Method Name: update chore item (put)
-Send: group id, chore id, chore name, due date, chore complete, user id
-Receive: processed?

-Method Name: delete chore item (delete)
-Send: group id, chore id
-Receive: processed?

api/rent/groupid
-Method Name: get rent info (get)
-Send: group id
-Receive: rent amount, rent paid, user id, group id

-Method Name: update group info (put)
-Send: group id, rent amount, rent paid, user id
-Receive: processed?

-Method Name: create new rent item (post)
-Send: group id, rent amount, rent paid, user id
-Receive: processed?