Loyalty Points, Coupons, Promotion Codes, Smart Cart Stuff, and other sales options – 2/8/17

Steve, Alan, Eric, Brandon

* The QR codes might end up helping us take rules and settings into the cart. Meaning, it may take some sort of id and/or instructions to the cart. The cart would then break things apart, parse what is needed, and then apply the values and/or look-up the rules accordingly. Basically, let the QR code help transport some of the logic pieces and/or directions of where to look things up at.
* One of the older methods was built using flex grid and a small self-contained list of prices for known quantities. For example: say you knew that it went 1/8 oz, ¼ oz, ½ oz, 1 oz (full). They, the users, would just add a pricing list such as: 5,9,16,27.
  + One of the benefits of the flex grid was it was a simple list
  + It still needs to be maintained
  + They have to be in a certain order and could break if setup wrong
  + This is somewhat of a hack of sorts
* The ideal would be a table or database type model so that it will be more set fields and structure
* One of the goals is checking the rest of the cart and seeing what else plays into the pricing tiers.
  + This needs to be a pre logic type action. This logic could then change the values in the cart.
* Possible problem with QR codes – If something changes, the QR code (old label) could go stale. If the QR code held all of the logic, it could change. If it only held an id number, it could always just look it up and then roll from there. That would be better.
* Calendar based sales and promotions – date and time based pieces.
* The smart cart is the logic side of the puzzle. The brains.
* Loyalty points – basics – clear back from a paper and pencil based model – usually you have a start date, a way to set an initial balance if needed, some sort of ratio between dollars spent to points. This could items sold, dollars spent, repeat items, etc.
* We need a stored value that is able to searched and manipulated via an actual one-to-many type relationship. We need the concept of an account of some sorts. If you only rely on a on-the-fly calculation, it gets too crazy. You run into crazy logic, time spans, rules, etc.
* What about special deals, double points, points earned in a certain time period.
* We need a place to store the earned points and the redeemed points. It really needs to be tied to promotion codes, campaigns, specific sales, etc.
* What about family members? What about sub customers who get to use a parent discount? Other crazy models.
* If we store points and/or credits, we could then start basing logic off of that. If we do store points and/or credits, that would need to take place when the cart converts from session into a real database.
* We may need to control the usage of points and/or credits. We don’t want to go into a negative points or negative credit. Image a person who has 2,000 points but wants to use 2,225 points. That shouldn’t work and/or happen. We have a max of 2,000 for this transaction. We just need to watch things.
* We need a centralized service that watches things. We are getting (showing or printing) loyalty points in the cart, in the receipt, and in the customer log. We need to standardize things.
* We may need this same centralized logic to show up in ecommerce and outside the secured environment.
* We need to help catch part of the story… not just showing points, but why do they get certain points for certain things. That is huge and great logic that helps tell the story.
* Can we combine loyalty points, gift cards, in-store credits, coupons, punch cards, etc. ??? They are all some kind of a liability of sorts
* What about vendor credits? What about customer credits? What about general non-denominational credits (not tied to a specific vendor or customer)?
* We may need to marry the in-line discount campaigns with the QR codes, with the coupons, with the possible loyalty points. Basically, sew things together.
* Is this standard and/or an add-on that needs to be setup and/or charged extra?
* We get into max, min, expiration dates, time periods
* On printing QR codes and promotions, we need to generate our own codes and then help the publishers either generate their own QR codes or help them know what is needed.
* What about a serialized coupon or a single one-by-one code. Or do you just record when the coupons are used and the system could check to see if it had been used by the current client or not. On 3/15/18 we talked about some options for coupons and being able to cross tie things together. This could be as simple as tracking campaigns via invoice line items or special account line items. We may need both options. Not required, but available.
* What about a digital punch card?
* This whole thing has been a huge problem since the beginning of time. We just need to help use the system to help track what we can and then show them what we have.
* You could get away from print ads and get them into a digital world where each QR code and/or coupon would be customer specific. That takes it to a tighter level.
* Eric was talking about coupons on coupons… basically, a simple entry level, then prompting for some info, and then creating a more specific code.
* If we make a new table, we just need a category and/or type to differentiate between loyalty points, in-store credits, coupons, punch-cards, gift cards, other…
* Coupons and promotion codes need to break out to a sub table that has rules and other restrictions and/or details.
* We need to show some of these liabilities and assets (vendor credits) on the balance sheet.
* What about special line items that could get skipped and/or mapped directly to the balance sheet. We need to make this as smooth as possible.
* We need to be able to search gift cards, search coupons, search promotion codes, etc. Maybe its own home page. We need to see balances. We need to be able to add to and subtract from an existing balance (load and unload).
* We need some reporting on what is still outstanding and/or still owed.
* At one time do you ever write things off and clear the accounts. For example, is there a expiration date on the points/credits?
* Do we want to assign any dollar values to those points and/or credits that are out there? The owners may want to see this in their balance sheet.
* We may need to look into some other products to see what they did. Basically, doing some research on other products.
* Some of the people use my cart favorite buttons to do things like bogo (buy one get one). We may want to tie in some of the my cart favorite logic into promotions, sales, discounts, etc.
* As a note, we have some other notes dealing with discounts, exceptions, and other tracking stuff.

///////////////////// left off here on 3/15/18 – brandon, eric, and Russell

* Get the big picture and then focus on the smaller outlay or requirements
* One of the goals of all of this sales and promotion stuff is getting the data back to see what is working and what is not working. Tracking a campaign is huge. What to use or which ones are working…. That’s what they want to know.
* What about special reports that group and show the campaigns, the promotions, the sales, etc. Help the owners and managers see what is working and what is going on. Basic analyst stuff (sums, max, min, averages, trends, etc.). How effective was the money that I spent for my marketing campaign?
* If we use the same table, we may want to have a flag that helps us know if we are dealing with assets (who owes us) vs. liabilities (who do we owe or potentially owe).
* What about in-house deals… single use, multi-use, time-based, reoccurring, etc.
* Make things configurable… this deals with naming, passwords, aliases, rules, assignments, etc.
* We are seeing a new table for the main account. This can be tied to a vendor, a customer, or a generic account. We then need a table that allows for transactions (load, unload, get, use, etc.). We are also seeing another table that deals more with the promotion codes and campaigns. We need to interconnect all of these pieces.
* Exclusions, per item, per category, rules, and logic. Take things out of the hardcoded level and make it data driven logic and rules. That is somewhat of the goal.
* We could still allow free-form discounts and such, but some of the new pieces will really start helping to define the usage. Another way is to define one of the campaigns as fully open. Either way, we need to group things and bring back the data in a report.
* See a small drawing… we had key players including vendors, users, customers, unknown people (generic), invoices, PO’s, and items. Those pieces already exist. We then create an special tracking account table that could tie to any of those people (persons including the general or unknown). That table would hold the who and the what type of account was it (asset or liability – who owes who money and/or a money equivalent). We then need a sub table that would allow for the accounts to be loaded and unloaded (positive and negative amounts). The other main table would be the promotion codes and/or campaign table. This holds special rules and assignments for items, parts, discounts, pricing, etc. We then would tie them all up together.
* Eric is in favor of bitesize chunks and multiple deliverables… keep it simple.
* Just a question… what about multi-tiered affiliate programs. It has similar needs (account, transactions, and promotions).
* As we create these new tools and features – who will babysit things? maintain things? and what kind of setup is needed? We can make some good money doing custom code, but it might be really awesome to make it part of the whole package.
* As far as funding… Is this an add-on or do we just take into our family model and kick back a percentage back out to the developers who are playing? Let’s look into a monthly percentage kickback type model.
* If we can easily add it to the standard, then let’s do it…
* As a note, we have put some rough costs (development prices) in the adilas community funded projects.
* Support? Who is going to do support? Or do we make it tight and then just roll it in to the whole system. Get more people involved into the project.
* Our plan for next time is reviewing the notes, making a plan, and starting layout database tables. Take small steps, kick things around, let it simmer, and build on last pieces. Treat it like the sub inventory project. Multiple doses, dipping in and out to figure out the game plan.
* Part of this plan is pulling in some of what Eric has already done and making it part of the bigger picture. That is pretty cool. As a side note, instead of paying for one-time custom code model, our goal is somewhat pulling things into a reoccurring revenue model. Pretty cool! ☺
* Steve was talking about a family type model, a trust, and a virtual coop type model.
* Not part of this conversation but… Eric has some expertise in servers, load balancing, database stuff, performance tuning, monitoring, checking for bad code or checking for non-efficient pieces.
* There is not a handbook for what we are doing… We have to look to other companies, history, and we understand that we are somewhat of a pioneer and/or explorer of one way or another.

2/16/17

* On the loyalty points and special accounts... maybe treat it like elements of time where you could setup as many templates as needed and then you have these virtual accounts for all kinds of things like: gift cards, loyalty points, affiliate programs, in-store credits, vendor credits, rewards points, etc. Let the people setup and maintain their own mini virtual accounts. They all use similar type functions and features. Maybe just let them name it, set it up, set the rules, and have it play into the system.
* We would love to piggy back as much of this possible. Basically create a special tracking account and then allow it to be loaded and unloaded (earned and spent) as needed.

3/15/18 (over a year later) – Meeting with Russell and Eric – both have projects for this type of account or special tracking.

* We need types of special accounts – be able to have multiple and be able to name the types.
* Eric has a liability report that he runs… it is the biggest pig out there… he doesn’t even release that to the clients… he pulls it. It is that big. We need to fix that.
* Summary, some good rules and logic, a backend to hold things, and being able to use and/or get to it.
* Russell wants to possibly move the coupons into their own feature. They may tie into the logic for loyalty points, but they could be different. We may need to add a varchar called “campaign” to both invoice line items and on special accounts (the line items and/or transactions).
* For the future… we may want to be able to record invoice numbers and even possible line item numbers. We would like it going both ways… if I look up the gift card, we can see or get to the line items. If we go to the invoice, we can get back to the gift card. Going back to the id’s, we use 0’s as a default and then plug them in if needed. We may even want app types, main id’s, and sub id’s on the transactional table. Possible options to tie into.
* The invoice payments need a way to possibly hook into the gift cards – basically from invoice payment to gift card.
* There was a conversation about normalizing data and de-normalizing data – transactional vs analytical databases. Details (transactions) or aggregate values (sums, counts, maxes, mins, averages, etc.).
* We want a quick search option for these special accounts and promotion codes… text, numeric, home, all, last, adv, etc.
* Points are a customer language… dollars and/or value are a different language (what is the ratio… we need to be able to get to both values). For example: 1 point = $1, or 1,000 points = $5, etc. How do they relate and what is the discount? Earn rate, value for redemption rate. These could just be settings per account type.
* Eric’s idea on an I-tunes type build your own rules. Build your own settings and ratios (rules and logic by the user). Once a feature is selected… dynamically show the options that apply. For example, if I choose a date setting… it asks about single dates, ranges, expirations, etc. If a numeric option is chosen, it would ask for max, min, ranges, greater than, less than, equal to, not equal to, etc. You get the idea. What are the switches and what are the values for those switches?
* Some of our logic needs to be called as a shopping cart refreshes. What will apply and what may still be in a dormant stage.
* On discounts and allowing and not allowing discounts… Eric would love to see that setting way up high on the part category level. Think way of high vs on the individual parts. We may need both levels. Eric has been seeing that the part categories seem to get the campaigns, discounts, and/or coupons. We can run things per item, but often it tends to be up higher on the part category level. Nesting hierarchy and what is the main driver?
* Part id’s… how to load the special accounts and how to unload and/or redeem things. Is this dynamic and/or system generated.
* Russell was concerned with the brain load on the shopping cart. Quick and light checks… before just processing everything. Basically, we know that the shopping cart is the hot spot. Be aware of that.
* Use the customer’s monies to help build out the system. We have multiple clients who are wanting these same things… that is awesome.
* Being able to schedule discounts and promotions (tied to dates and times). For example: We know that this Saturday we need all of the soda pop be on sale with a 25% discount. Be able to prep the system for future sales and promotions.

3/19/18 back on the call with Russell and Eric. We started out looking for other assets, media, and other brainstorming files.

* Did a quick search of media/content in the adilas shop site for “loyalty”. We got some good info on that. We then went into the different media/content files and looked at the different things and made comments.
* We found one screen shot that showed a normal balance sheet item. We then saw a number of graphics and drawings off to the side. We added to that drawing and we ended up numbering some of the sub tables and then took another screen shot. We set the normal balance sheet item as the main (already exists), normal balance sheet subs (already exists – not going to be used other than special changes), then main account types (new # 1), account rules (new # 1.5), special accounts (new # 2), and special account transactions (new # 3).
* We also talked about adding in master list of what is available – this would be a top-level table of concepts such as lunch cards, gift cards, loyalty points, in-store credits, etc. We then use that master list and let the different corps and users setup their own objects based off of the generic master list.
* These master lists of things will help us present the logic, rules, and validation for the general items. Basically, think of this as a master template of sorts and then we let the users lightly modify existing rules and logic values.
* Eric was asking the question… How does this gift card and loyalty points look on the invoice side? It seems like a form of payment (money\_types table). On the money type id’s… if we go to custom per corp, we need a master list and then API that data to all servers… that way we don’t ever have a problem with data assignments and merging future data. Think about using a global id and then matching up all servers… this is a very small table and could make it more portable in the future.
* Originally, we were talking about monies. But in reality, it might be better called resources… this doesn’t have to be money… it could be time, expertise, counts, quantities, talents, abilities, virtues, smoke and mirrors, phantom numbers, etc. We were talking money types… but not everything is fully related to money.
* Do we need to flag the money types? Where do they really go and how do they play in? Maybe a money destination of sorts – P&L (income statement) or BS (balance sheet).
* Are the special tracking accounts quantity based or dollar based? How do we speak their (our users and/or clients) language? Ratios come into play and also displaying data back to the users/clients. We may need to show both quantities (points or values) and dollars (real or equivalent dollar value).
* We know that we need to allow parts (general inventory items) to be skipped and/or mapped to the balance sheet. We don’t want them showing up on the P&L under revenue. Maybe pattern them after the special line items such as “Collected”, “Other”, etc. This would be nice if each item has some mapping on where they should and/or could show up. The special line items do have that, but those are the only ones.
* What if we allowed the users to pick from existing money types as part of the setup of the main account types? Say things like “Internal”, “Other”, “Gift Card/Certificate”, “Coupon”, etc. We then record something special on the payment notes to tie it into the special account tracking piece. For example: Redeemed $50 Lunch Card or Redeemed $100 Gift Card. We could also allow for real id number tie-ins but don’t skip the notes section. That is an easy way to get things done and allow for tons of options.
* We talked with Steve about some of the accounting pieces… loyalty points show up as potential points and/or potential future dollar values (future discounts). When really used… it will be discount against the revenue portion. We may need a switch that shows if it is a phantom liability of points earned and possible monies owed or real monies owed. This is true with coupons and loyalty points (smoke and mirrors of potential discounts that will be applied). These don’t feed into the real A, L, E (assets, liabilities, and equity - balance sheet stuff). The BSI still exists, but is more of an informational type account. We show the balance sheet item as a $0 real value… if you drill down to see it, you will see the details and sub transaction values but the real balance sheet will still be zero. Long story made short… redemption of loyalty points is basically a form of discounts… that could be in-line or general or standalone discounts.
* We need a way to redeem or claim these things back into the system and claim things as revenue (real monies). How do you clear out gift card liabilities and/or customer loyalty points? You need some kind of clean-up tool and/or feature. This may end up being part of the standalone declarations, watchers, and feeders. Lots of fun stuff.
* Russell – we need to consider the user side of the transactions. It needs to either be so easy that they just figure it out, or we provide some training and show the basic steps. Basically, we will have all kinds of moving pieces behind the scenes with a simple user interface on the frontend.