Meeting with Cory, John and Brandon on 11/24/21. Dealing with the discount engine MVP for Scrum and also possible future add ons. These notes will be added to the product backlog as user stories.

Story: Salesperson adding items to the cart, and I need for the discount engine to work as it should.

Following manager scenarios:

Name it, then check to see if the discount already exists. If it does, it will throw an error message up top. (how will this be validated?) All the discounts will be pulled in, even if they are inactive.

\*\*This is on the to do list

Edit the name (this is now select existing)

If it’s archived, you can check a box to unarchive or activate it, and then you can edit it.

Discount amount: Percent or dollars

Active switch-or inactive

Archive= will be hidden

We’d like a one pager- all the stuff is flipped behind the scenes (ajax) the page is rewritten.

Instead of pushing to an action page and then a success page.

\*\*Need to update the tabs (active, inactive, archived) once the discounts are moved.

Need to update the data table to reflect these changes.

Customer types: John is wiring it up as we speak. Click box, select all, or select multiple types.

When we pull open list builders, it needs to retain what was previously selected, as well as show you what was selected.

Categories: Same behavior would apply to the categories.

\*\*Based on the item categories selected, we would narrow down the vendors that were associated with those categories. This will take some ajax. Self-limiting field based on the criteria. Maybe instead of ajax call to db we filter the memory objects from within the page. This is a decision that needs to be made.

Ideally John would like DAO’s and call against the object instead of the database.

Items will behave the same way… limited by the vendor which is limited by the category.

Dates and times: Db work needs to be done

\*\*Min Max: only on a per line basis currently. They really want it to be by the cart. Major value add to figure out the cart as a whole (not just the line) Maybe phase 2.

Graphs of cool information would be part of another phase/bonus round. Hard to determine as there are many discounts that can be combined. Need to store somewhere.

Sort: will work if added to the table first. Otherwise doesn’t.

Maximum stackable: staying as it is.

Max # of items in the cart: staying as is.

Go through the cart and for the per, we need to back this stuff out. Loops of loops. High level stuff in the cart.

Possible enterprise level action.

Building it out so we can get accurate data per line so we can get data for charts and graphs.