# The top 5 – 20 reasons that this solution will be used:

1. Easily create, view, store and print recipes
2. Easily view the impact that cost changes to ingredients has on items/products/recipes
3. View items that suppliers carry and related costs; be able to compare costs across suppliers
4. Easily update the annual inventory

# Overview:

* Suppliers sell us ingredients
* Ingredients are used to create a recipe
* Recipes create items that can be sold

# Phase 1:

* Create a form and table to add suppliers. Keep it simple for now.
* Then add a form to add / edit ingredients and select the supplier
* Then add a form to add recipes that include ingredients
* Then add a form to add items using recipes

Should I allow for Product Tags : e.g. Gluten Free, Diabetic, etc

# Relationship

postgres://blogpost\_db\_frcs\_user:LGKGduHR68EpEVLwrqtiKze4hzc7NJ9i@dpg-co6nn5q0si5c73cjg46g-a.ohio-postgres.render.com/blogpost\_db\_frcs

PGPASSWORD=LGKGduHR68EpEVLwrqtiKze4hzc7NJ9i psql -h dpg-co6nn5q0si5c73cjg46g-a.ohio-postgres.render.com -U blogpost\_db\_frcs\_user blogpost\_db\_frcs

# Categories

Categories are things like:

* Breads,
* Cookies
* Muffins
* Pies

I think these are more like filters. Like which muffins we have.

# An Item

Items are sellable things like chocolate chip cookies, cinnamon rolls, cutout cookies, strawberry cream cheese Danish. Each of these have 1 or more associated recipes. For example, cutout cookies use the cutout cookie and the buttercream frosting recipe

Items have name, descriptions, prices are manually entered but periodically evaluated based on cost to maintain margins. Items can be retail only, wholesale only, both retail and wholesale, special order only (e.g. Gluten Free) or Cannoli).

* Do we allow all items to be special ordered? What about wholesalers, does that have a different price?

Does Square have an API that can allow me to add items? What about Quickbooks?

Side note: Can I integrate Square to QB Online to eliminate sales entry from Patti.

## Data Elements:

One-to-many relationship with Recipes

* Item ID
* FK : Recipe ID
* FK: Category
* Name
* Description
* Retail Price
* Wholesale Price
* Out-of-season Special Order Price

# A Recipe

A recipe contains ingredients.

There is a cost to prepare each recipe.

* Labor cost :
  + Time \* labor\_rate
  + Note : need way to store and calculate labor\_rate
* Ingredients

Cutout Cookie Recipe:

|  |  |  |
| --- | --- | --- |
| Flour | 13.5 | lbs |
| Sugar | 6.5 | lbs |
| Eggs | 27 |  |
| Vanilla | 0.5 | Cups |
| Milk | 1 | Cups |
| Butter | 7.5 | lbs |
| Baking Powder | 6.75 | oz |
| **Total Cost of Ingredients** |  |  |

* Unit\_of\_Measure
  + It would be great to easily flip between weight and cups/tblsp/tsp depending on how many units we need to make
  + So… 3 options
    - Maintain a table for each ingredient (cost per lb, cup, tblsp, gallon, cup, etc)
    - Maintain only the default Unit\_of\_Measure in the Ingredient or Supplier table with the associated cost (may be required no matter what).
      * Eg. A 50 lb bag of flower has a cost $25.00. The programming would calculate the cost for the ingredient based on the recipe.
    - Calculate the unit of measure based on a conversion table
      * Dry measure: 1 lb = x cups = y tblsps = z tsps.
      * Liquid measure:
      * Butter: sticks, cups, lbs
      * So would there be different formulas for dry vs liquid vs butter measures? Butter measurements may apply to margarine as well.
* Recipe Cost
  + Depends on how much of each ingredient is used (Unit\_of\_Measure)
  + This could be calculated and probably should
* Some recipes include other recipes
  + E.g. Buttercream frosting for cutouts
  + Mocha Buttercream frosting for Brownies
  + Can we do this with a data element that has the id of “Includes this recipe”? Do we ever connect more than two recipes for a product? Do we need an Item table that has a one to many relationship to our recipes? YES

## Data elements:

One-to-many relationship with Ingredients

Recipe ID

Recipe Name (note that most of the time, the recipe name is the same as the Item name and should default to that during data entry)

Ingredients

* So there should be a one to many relationship between a recipe and ingredients. Figure out how that works

Quantity

Unit of measure

Time it takes

Quantity of items that it makes

Preparation Description (e.g. bake at 450 for 20 minutes)

# Ingredients

Ingredients have a supplier, possibly more than one. But usually we get them from a primary supplier. Each ingredient has an associated cost for the specific supplier. The cost…

* Ingredient cost. There is a cost from each supplier. We currently only update costs a couple of times a year and base our recipe cost and inventory cost on the primary supplier as compared to using an average cost for the year or time period.
* In our ingredients table we include the unit\_of\_measure cost.
  + Eg.

## Data elements:

One-to-many relationship with Suppliers

Ingredient ID

Ingredient Name

Ingredient Description

Ingredient Category / Department

* Dry goods like flour, sugar, chocolate chips

# Supplier

Supplier ID

Supplier Name

Contact Information

One-to-many relationship with Supplier Item

# Supplier Item

Are the Supplier Item and the Ingredient almost synonymous? I think they are – the difference would be an ingredient could have an average cost spread out across multiple suppliers. Also would we need an historical supplier table as prices change. I think we could have that be the order table. As items are ordered and received, we can track quantity and cost. Probably can’t maintain in QB w/o paying for inventory. But that may be ok.

Supplier Item ID

FK Supplier

Item ID / Barcode

Item Order Code (Optional)

Item name

Item Category

* Dry Goods

Item Class

* Flour

Item Size

* Vanilla : How to handle case of 4 gallon jars
* Baking Power: Case of 12 boxes at 10 oz each
* Flour : each bag is 50 lbs

# For reselling:

* Need to add company info
* How to have a multi-tenant solution?
* Kubernetes and containers?

A screenshot of a computer

Description automatically generated