Prototype Design [Software]

* User Login/Registration
* Tabs
* Profile
* Profile Info [Username, …]
* Kitchen Info [Item count, Wasted KG, …]
* Logout
* Fridge
* Inventory – separated section
* Nearly Expired Items – separated section [hidable] + notification
* Recipe
* Featured / Suggestive Recipes
* Serving Amount Calculation – fixed 4

ValidIngredient Table [SYSTEM’S INGREDIENTS]

ItemID [PK]

defitemName <- default item name

AnalogousNames <- other equivalent names

Expiry <- how long will the item last

imageURL

Inventory Table [USER’S INVENTORY]

InvID [PK]

ItemID [FK]

AmExp Table [AMOUNT & EXPIRATION || USER’S INVENTORY EXPIRY]

amexpID [PK]

ItemID [FK]

itemAmount <- item amount

itemDateExp <- item date expiration

Recipe Table

recipeID [PK]

scrIngredient <- string array mula sa webscrape

scrInstruction <- string array mula sa webscrape

scrTimeReq <- string mula sa webscrape

sysIngredient <- string array mula sa system [removing all invalid ingredient]

Serving Units [singulars] \*add ‘s’ for another condition

Pound

Ounce

Cup

Tablespoon

Large

Clove

Teaspoon

Variables na need para sa Recipe Algorithm

sysIngredient

itemDateExp

CapstoneDatabase

1. SysIngredient

* id – holds the primary key for this table
* defItemName – the name of the item that will be displayed in the app. This is also the string name that the device will return.
* analogousNames – different naming convention of a specific item name from different recipe
* expiry – how long the item will last (day)
* imageURL – URL link of an upload image from firebase (may handler pag walang image url)
* itemType – this can either be perishables or non perishables. ‘p’ or ‘np’

1. SysRecipe

* id – holds the primary key for this table
* scrName – recipe name
* scrIngredient – webscraped data ingredients, string array, lalabas sa instruction part ng recipe.
* scrInstruction - webscraped data instructions, string array, lalabas sa instruction part ng recipe.
* scrTimeReq – webscpared time required to cook a recipe, total cooktime
* sysIngredient\_id – filtered scrIngredient string array. Only valid ingredients should remain. String array of ingredient IDs.
* imageURL – URL link of an uploaded image from firebase.
* scrDescription – recipe description
* scrDefServing – default serving

1. UserInventory\_Items

* id – holds the primary key for this table
* itemId – SysIngredient ID, Foreign Key
* itemWeightExp = [itemWeight,expiryDate],[ … , … ], [ , ], … (String)

\*expiryDate – Tells when will be the exact expiration date; A string with date format.

* user\_id – the user that owns the item

1. UserInventory\_Amount\_Expiration
2. WebRecipeDetail
3. WebRecipe\_Unit – collection of units from the website
4. WebRecipe\_Ingredient – collection of ingredients from the website
5. UserProfile
6. id – holds the primary key for this table
7. User\_Name – user’s profile name
8. User\_Email – user’s email contact / used as an id / 1 email per account
9. User\_Password
10. SysMessage
11. Mess\_State – on or off – will read by the rpi
12. Mess\_ItemName – item string name message from rpi
13. Mess\_ItemName\_Others – list ng ibang possible item names. Name,name,name,name
14. Mess\_ItemWeight – item weight message from rpi
15. Mess\_ItemImageURL – item image url uploaded by the rpi and link.
16. \*NOTE\* Sa mobile app, sya yung mag tuturn on and off ng mess\_state. Every time na mag siswitch ng state, set all to ‘none’ yung data. It means na null yung value at wala pang valid input galing rpi.

Sa rpi naman, may continuous time interval sya. For every given second, babasahin nya yung Mess\_State to know kung mag poprocess na ba sya ng data o hindi. Pag naka on na yung state, mag poprocess na sya, gagawin na nya yung dapat nyang gawin. Once may result na, before nya isend yung data, dapat nya muna icheck kung naka on parin yung Mess\_State para malaman kung isesend pa ba ng rpi yung result o hindi. In short, si rpi ay taga send lang ng data at nag hihintay lang sya ng signal kung kelan nya gagawin ito.

Sa rpi, may dalawang group of results syang isesend. Yung unang group, consisted ito ng item\_name, other possible item name, at yung image url ng image na naupload sa storage. Yung pangalawang group naman ay yung mismong item weight lang.

Para kahit papaano makabawas ng processing part kay rpi, need muna natin ng conditions before mag send kay rpi. Bukod sa pag naka on yung Mess\_State, sa item name, dapat mag sesend lang sya ng data pag yung top suggestive name nya ay nabago o iba sa unang suggestion nya. Sa pag update ng item name, kung ito nga ay nabago, need din maupdate yung image\_url at yung other suggestive names. Yung condition na ito will run in intervals for every given second.

Sa condition ng weight, maari muna na icheck kung yung weight value na galing sa sensor ay nag stay at walang malaking changes sa loob ng ilang segundo. Pag na satisfy na ng weight ito, dun lang sya mag sesend ng data sa firebase.

Ito yung dalawang conditions na kailangan. Once na ma-satisfy ang mga ito at may result na, before isend, need muna icheck kung naka on parin yung Mess\_State.

Continual ang pag check ng mga conditions for every given second sa rpi. Ganito yung setup kasi habang ginagamit natin yung device, may mga changes na pede mangyari at para maging flexible din.

Pag naka off si Mess\_State, naka off yung main processes ni rpi, continual din na nag babasa si rpi ng Mess\_State. Pag naka on naman na, continual nya irurun yung mga conditions at main processes tapos before nito isend ang result, need muna icheck kung naka on parin si Mess\_State. Pag napansin nan aka off na, balik sya dun sa nag hihintay sya mag on yung Mess\_state.

1. Set up all valid ingredients gamit yung binigay nila poch
2. Add ng item
3. **Update Item Weight**
4. Recommend ng recipe with the scoring algorithm
5. Select Recipe to Cook
6. Record all items that needed to be updated

Filter ng mga prioritized recipes

Fridge tabs

Update Fridge Item

Firebase data, units gawing grams

wednesday

Reccomendation

Device

**Kelan ulit yung defense**

**Need ng video and images for presentation**

**Images o videos ng mga nabili, natira, yung pag weight o pag gamit ng device, yung pag gamit ng mobile app.**

Before

Total Weight ng mga ginamit, sum ng mga nagamit per day

Weight ng mga natira, sum ng mga natira sa araw araw.

After

Total Weight ng mga ginamit, sum ng mga nagamit per day

Weight ng mga natira, sum ng mga natira sa araw araw.

Natira/Nagamit

50g/100g

To do:

Add an user inventory

make a search bar pag mag aadd ng item.

Prompts

List of items that are needed to be updated due to cooking recipes

List of items na malapit na mag expire

Prompt for the device pag mag stastart mag detect at wala na detect.

Dapat, kung ano yung ginagamit na unit sa recipes, ganun din ang unit sa inventory. Isang unit lang para sa lahat.

SysIngredient

Example:

Avocado

Used in

“Recipe1” : amount, unit, avocado

“Recipe2”: amount, unit, avocado