**SimplyPie UML Diagrams**

**Storage**

* **Storage ID**
* **Storage Name**
* **Status (e.g. Full/Empty/Space Storage description** e.g. *Perishable goods only*

**User**

* **User ID**
* **Username**
* **Password**
* **Firstname**
* **Surname**
* **Gender**
* **Email address**
* **Living Status**
* **Occupation**

**Recipe**

* Recipe Name
* Ingredients
* Steps
* TotalTime
* *Calories*
* Serving
* Counter (*number of times cooked)*

**1**

**1..\***

**1..\***

**1**

**1**

**1**

**1..\***

**1..\***

**Food Category**

* **Category ID**
* **Category Name**
* **Description**

*(I.e. Meat should be stored in the freezer storage)*

**Food Item**

* **Item ID**
* **Name**
* **Quantity int (not null)**
* **ExpiryDate date (nullable)**
* **Category – to be used as foreign key**

*E.g. Vegetable/Meat/Canned/etc*

* **Image path (optional)**
* **ExpirationHistFlag**
* **Individual Price**
* **Price\***
* **Storage location (foreign key)**
* **Notes**

**1**

**1**

**Meal Diary (calendar)**

* **Date**
* **Breakfast** (*links to recipe ID*)
* **Lunch** (*links to recipe ID*)
* **Dinner** (*links to recipe ID*)
* **UserID (foreign key)**

**1..\***

**1**

**Reject Pile**

* **Item ID**
* **Storage ID**

*This table will be used to define a trend of food items wasted, to ensure that the user knows what sort of items were left unused*

**User Group**

* **User ID**
* **User ID\***

*(This table will be used for Shared Profiles i.e. multiple users can view the same storages + recipes. Designed for users living in the same house i.e. families/flat mates)*

**EXPAND**

How the user would use SimplyPie (+ registering and logging in) **FIRST TIME USER**

Note: when a user first registers, a confirmation email will be sent to the address provided by the user. The user cannot proceed until the email has been verified (this is because the main way for the application to communicate to the user is by email – especially when they are not logged on the system).

1. Create virtual stores, reflecting the food storage areas at home
2. Input items to the stores:

* Select a store then input all the items there
* **OR** input all the items (from the receipt), one of the fields would be a dropdown list showing all the storage areas

1. Once the items have been saved, user will be presented with an expiry alert notification options: ‘*Alert me when products are about to expire: Two weeks before, A week before, 3 days before*’ This could be set individually for each product or general
2. For **Phase 1:** Manually Input Personal Recipes to the system, filling out all the fields for Recipe entries. This will be done using the Calendar functionality e.g. Select Day then **Add Recipe** (set frequency for each recipe: e.g. Fish and Chips on Fridays, recipe planned every 3 days etc.) + Edit, Delete

Start Cooking step 🡪 will display the ingredients required + currently stored items. Display a shopping list of items not already bought.

1. Expiry Process will begin running once the user has saved the contents of the storages

Daily runs of the SimplyPie **WHEN AN ITEM EXPIRES**

1. The application will send the user an email (depending on the frequency set by the user) telling them certain item(s) are about to expire *in x days/weeks.*
2. User logs into the website, a notification is displayed reflecting the email sent. Items presented on the screen
3. User will be shown the following options for each item (check box + drop down list):

**Cook Now**

**Remove**

**Set Aside – equivalent of Snooze (set another alert process email)**

**Plan Recipe**

**Items Expiring *23/10/2016-25/10/2016***

**ACTION**

**EXPIRY DATE**

**QUANTITY**

**ITEM**

**Submit**

**25/10/2016**

**400g**

**CHICKEN**