|  |
| --- |
| CSC 436 |
| Kitchen Manager |
| Software Design Document |
|  |
| **Cody Adams, Winston Belin, Ryann Case, Andre Edwards, Adanma Nkume** |
| **4/6/2010** |

|  |
| --- |
| The Software Design Document provides a detailed design of the functions of the Kitchen Manager System and of the system itself. Included are charts, diagrams and descriptions that provide the information necessary to comprehend the system’s behavior. |

Table of Contents

[1 Overview 4](#_Toc258277795)

[1.1 \* Purpose 4](#_Toc258277796)

[1.2 Scope 4](#_Toc258277797)

[2 \* System Architecture (Draft) 5](#_Toc258277798)

[3 \* Data Dictionary 6](#_Toc258277799)

[4 \* Software Design 6](#_Toc258277800)

[4.1 Kitchen Manager Component Diagram 6](#_Toc258277801)

[4.1.1 System Organization Class Diagram 7](#_Toc258277802)

[4.1.2 Current User Setup Class Diagram 8](#_Toc258277803)

[4.1.3 Add Food Class Diagram 10](#_Toc258277804)

[4.1.4 Meal Checkout & Shopping List Printout Class Diagram 12](#_Toc258277805)

[5 Sequence Diagrams 16](#_Toc258277806)

[6 Data Design 25](#_Toc258277807)

[6.1 Persistent/Static Data 25](#_Toc258277808)

[6.2 Transient/Dynamic Data 26](#_Toc258277809)

[6.3 External Interface Data 27](#_Toc258277810)

[6.4 Transformation of Data 27](#_Toc258277811)

[7 \* User Interface Design 27](#_Toc258277812)

[7.1 User Interface Design Overview 27](#_Toc258277813)

[7.2 User Interface Design Hierarchy 27](#_Toc258277814)

[7.3 Use Cases 27](#_Toc258277815)

[7.3.1 Use Case Description: Add Custom Recipe 27](#_Toc258277816)

[7.3.2 Use Case Description: User Management – Add User 27](#_Toc258277817)

[7.3.3 Use Case Description: User Management – Remove User 28](#_Toc258277818)

[7.3.4 Use Case Description: Switch User 28](#_Toc258277819)

[7.3.5 Use Case Description: Existing Food Item Entry 29](#_Toc258277820)

[7.3.6 Use Case Description: Non-existing Food Item Entry 29](#_Toc258277821)

[7.3.7 Use Case Description: Change Icon 29](#_Toc258277822)

[7.3.8 Use Case Description: System Preferences - Language 30](#_Toc258277823)

[7.3.9 Use Case Description: System Preferences – Units of Measure 30](#_Toc258277824)

[7.3.10 Use Case Description: System Preferences – Background Change 31](#_Toc258277825)

[8 \* Glossary 31](#_Toc258277826)

[9 Version History 33](#_Toc258277827)

# Overview

## \* Purpose

The purpose of this document is to describe the requirements for creating a Kitchen Manager system. This is to be used in the customer’s home to keep track of food inventory and assist in meal planning. The system will allow users a very easy way to manage what they eat and to further extend that into helping users adjust their diets with as little effort as possible. This document will have all required information for creating software for the entire system and required hardware needed and recommended.

## Scope

Kitchen Manager will be an in home system that will allow users to keep an inventory of their food items. It will allow users to keep track of what they are eating and will do the math on caloric intake and other nutrition information. This will help alleviate users of the tedious task of keeping track of what they eat themselves. Many people are discouraged from maintaining a proper diet because they simply do not want to have to keep track of these things themselves. Kitchen Manager will offer an easy solution for tracking that information.

The system will be a personal computer with a touch-screen LCD monitor, UPC barcode scanner and CPU that are completely integrated. It will also include internal databases and connections to external databases that store the users’ personal information, contain information about the food items such as price and nutritional information, and store information about the users’ daily nutritional intake. The system will allow the user to manage a running inventory of what foods they currently have in their kitchen. Options available to the user will be to scan in any food purchased in order to keep inventory. In the event the item doesn’t have a barcode the touch-screen keyboard will allow the user to manually enter the item into the system.

The system will have built-in Wi-Fi to allow it to connect to either a company branded Wi-Fi printer or any store-bought Wi-Fi printer connected to the user’s home network. The Wi-Fi will also allow the system to connect to the internet through the user’s router. Incorporation of downloadable nutrition information, recipes and meal plans will be enabled through a Wi-Fi connection. The printer will allow users to print off meal plans and/or recipes to assist with cooking,

The system will allow users to keep track of just what exactly they are eating and will do the math on calories and other nutrition facts to alleviate end users of the tedious task of keeping track of what they eat themselves. Many people are discouraged from maintaining a proper diet because they simply don’t want to have to keep track of these things themselves but if an easy solution for tracking that information was available many would be inclined to take advantage of it.

The inventory system on the unit coupled with the optional printer will allow users to print off grocery list before heading to the grocery store. When a user goes into the kitchen to eat they can use the system to subtract an estimated amount of what they ate. An example would be a box of animal crackers. The nutrition facts provide the serving information and the end-user uses the touch-screen device to enter how many servings they ate. The inventory system subtracts the servings from the known total and updates the information into a database. When the user prompts a list of low quantity foods, it is printed from the optional printer assisting them in knowing what foods they need to buy when shopping.

The system will allow various users to keep track of their own eating patterns and habits by having different user boxes for each person in the family. At the user’s request or user pre-programmed intervals, the meal information can be printed or uploaded to the internet where meal suggestions and other dietary assistance can be provided by professionals.

An option for a partnership with grocery store and retail chains presents the optional ability to have information directly added from the store itself. As an example, users could be provided with a membership card with a barcode that will allow the partner store to upload whatever purchases the end-user makes to servers and ultimately to the end-user’s system itself. The cashier would scan or swipe the card before processing the order and at the end of the order the items purchased would be uploaded then downloaded to that specific user’s system. This would further ease the user’s required work to use the system.

The ultimate goal of the system is to allow users a very easy way to manage what they eat and to further extend that into helping users adjust their diets with as little effort as possible. Various interfaces can be used to achieve this result. At the core of the system is the software itself which will be pre-installed to the Kitchen Manager PC.

# \* System Architecture (Draft)

# \* Data Dictionary

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Entity Name | Element Name | Definition | Type | Storage Format | Scale | Bounds | Display Format | Mandatory Entry/Fill | Default Value | Modified by | Read by | Constraints |
|  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |

# \* Software Design

## Kitchen Manager Component Diagram



### System Organization Class Diagram



### Current User Setup Class Diagram



#### Pseudo code

Current User Details Screen () {

Display user details

}

Add User () {

AddUserForm.Show;

Collect data into variables to push to TABLE;

INSERT INTO USERTABLE (SQL);

AddUserForm.Close

};

Remove User () {

Display user list from USERTABLE (SQL);

RemoveUserForm.Show;

REMOVE SELECTED USER FROM USERTABLE (SQL);

RemoveUserForm.Close

};

Switch User () {

UserForm.Show;

Display user list from USERTABLE (SQL);

Select user from TABLE

Change current User

UserForm.Close

};

### Add Food Class Diagram



#### Pseudo code

Increase Quantity () {

Search Food inventory table for food item

If food item available

Increment item

Else search UPC table for item

Copy from UPC table to food inventory table

Increment item

Fooditementry.close

Re-activate Real-time scanning

};

Add Not Recognized Item Screen (UPC) {

Get food item details

Add UPC table

Add Not Recognized Item Screen.close

Call Food Item Entry Screen (UPC code)

}

### Meal Checkout & Shopping List Printout Class Diagram



#### Pseudocode

Food Item Detail Screen (food item) {

Display food item details

};

Add Food to Meal () {

Search Instockinventory.select

Display food item details

Add food item to meal

Decrement food item inventory

fooditemdetails.close

Select Instockinventory ();

};

QuickAdd Food to Meal (item, quantity) {

Search Instockinventory.select

Add food item to meal

Decrement food item inventory

Select Instockinventory ();

};

Add Food to Shopping list () {

Search Instockinventory.select

Display food item details

Add food item to Shopping list

fooditemdetails.close

Select Instockinventory ();

};

QuickAdd Food to Shopping list (item, quantity) {

Search Instockinventory.select

Add food item to Shopping list

Select depleteinventory ();

};

Per-item Decrement () {

For MealArray (start to finish) {

If MealArray (current) = Food ItemSelected {

MealArray (current)--;

} } };

Add User () {

AddUserForm.Show;

Collect data into variables to push to SQL USER TABLE;

INSERT INTO SQL USER TABLE;

AddUserForm.Close

};

Clear Meal/List () {

FoodInMeal/List [] = New String FoodInMeal/List [];

};

Eat Meal () {

MealArray.Clear;

} };

Print Shopping List () {

For FoodArray (start to finish) {

Print FoodArray (current);

}

ListArray.Clear;

} };

Display Food to Browse Food Screen (Current Starting Position) {

Retrieve data from SQL Food Inventory Table into the run-time food array;

Display specified 5 items to form;

};

Move Forward/Back in Browse Food Screen () {

Get current array starting position;

Increment by 5;

Pass new starting position to Display Food to Browse Food Screen ();

};

Food Item Entry Screen (UPC Code from Keyboard/Scanner) {

Deactivate Real-Time scanning;

If Food Item is in Food Inventory Database;

Display Food Item Details to Screen;

Else If Food Item is in UPC Database;

Display Food Item Details to Screen;

Else

Display food is not available;

Deactivate Increase Quantity Button;

Activate Add Not Recognized Food Item button;

};

Select In-Stock Inventory (SortType) {

SELECT \* FROM FOOD INVENTORY DATABASE WHERE QUANTITY > 0 SORTED BY SortType;

Build Linked-List to facilitate scrolling;

Display SQL Query Results (Linked-List) to Browse Food Screen;

};

Select Depleted Inventory (SortType) {

SELECT \* FROM FOOD INVENTORY DATABASE WHERE QUANTITY >= 0 SORTED BY SortType;

Build Linked-List to facilitate scrolling;

Display SQL Query Results (Linked-List) to Browse Food Screen;

};

# Sequence Diagrams



















# Data Design

## Persistent/Static Data



## Transient/Dynamic Data

## External Interface Data

## Transformation of Data

# \* User Interface Design

## User Interface Design Overview

## User Interface Design Hierarchy

## Use Cases

### Use Case Description: Add Custom Recipe

Actor: User

System: 0. User open Kitchen Manager

1. System display Main Screen
2. User log into user account
3. System display Current User Details Screen
4. User select Add Custom Recipe option
5. System display Recipe Screen
6. User add custom recipe

#### Screen shot of forms related to Use Case

##### (Description of fields on form)

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Entity Name | Element Name | Definition | Type | Storage Format | Scale | Bounds | Display Format | Mandatory Entry/Fill | Default Value | Modified by | Read by | Constraints |
|  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |

### Use Case Description: User Management – Add User

Actor: User

System: 0. User open Kitchen Manager

1. System display Main Screen
2. User select Current User Details
3. System display Current User Details Screen
4. User select User Management screen
5. System display User Management options
6. User select Add User option
7. System display Add User option
8. User enter user information
9. System save user information

#### Screen shot of forms related to Use Case

##### (Description of fields on form)

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Entity Name | Element Name | Definition | Type | Storage Format | Scale | Bounds | Display Format | Mandatory Entry/Fill | Default Value | Modified by | Read by | Constraints |
|  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |

### Use Case Description: User Management – Remove User

Actor: User

System: 0. User open Kitchen Manager

1. System display Main Screen
2. User select Current User Details
3. System display Current User Details Screen
4. User select User Management screen
5. System display User Management options
6. User select Remove User option
7. System display Remove User option
8. User remove user information
9. System accept and remove user information

#### Screen shot of forms related to Use Case

##### (Description of fields on form)

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Entity Name | Element Name | Definition | Type | Storage Format | Scale | Bounds | Display Format | Mandatory Entry/Fill | Default Value | Modified by | Read by | Constraints |
|  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |

### Use Case Description: Switch User

Actor: User

System: 0. User open Kitchen Manager

1. System display Main Screen
2. User log into user account
3. System display Current User Details Screen
4. User select Switch User option
5. System display Switch User option
6. User select from available users
7. System display User details

#### Screen shot of forms related to Use Case

##### (Description of fields on form)

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Entity Name | Element Name | Definition | Type | Storage Format | Scale | Bounds | Display Format | Mandatory Entry/Fill | Default Value | Modified by | Read by | Constraints |
|  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |

### Use Case Description: Existing Food Item Entry

Actor: User

System: 0. User open Kitchen Manager

1. System display Main Screen
2. User select Food Item Entry Screen
3. System display Food Item Entry Screen
4. User select Increase Quantity option
5. System display Increase Quantity option
6. User scans new food
7. System accept updated quantity

#### Screen shot of forms related to Use Case

##### (Description of fields on form)

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Entity Name | Element Name | Definition | Type | Storage Format | Scale | Bounds | Display Format | Mandatory Entry/Fill | Default Value | Modified by | Read by | Constraints |
|  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |

### Use Case Description: Non-existing Food Item Entry

Actor: User

System: 0. User open Kitchen Manager

1. System display Main Screen
2. User select Food Item Entry Screen
3. System display Food Item Entry Screen
4. System display Increase Quantity option
5. User scans new food
6. System does not accept updated quantity
7. User select Add Not Recognized Item Screen
8. System display Add Not Recognized Item option
9. User enters food item manually
10. System accept updated quantity

#### Screen shot of forms related to Use Case

##### (Description of fields on form)

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Entity Name | Element Name | Definition | Type | Storage Format | Scale | Bounds | Display Format | Mandatory Entry/Fill | Default Value | Modified by | Read by | Constraints |
|  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |

### Use Case Description: Change Icon

Actor: User

System: 0. User open Kitchen Manager

1. System display Main Screen
2. User log into user account
3. System display Current User Details Screen
4. User select Change Icon option
5. System display Change Icon option
6. User select from available icon
7. System display icon

#### Screen shot of forms related to Use Case

##### (Description of fields on form)

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Entity Name | Element Name | Definition | Type | Storage Format | Scale | Bounds | Display Format | Mandatory Entry/Fill | Default Value | Modified by | Read by | Constraints |
|  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |

### Use Case Description: System Preferences - Language

Actor: User

System: 0. User open Kitchen Manager

1. System display Main Screen
2. User log into user account
3. System display System Preferences
4. User select language option
5. System display available language options
6. User select from available languages
7. System change system language

#### Screen shot of forms related to Use Case

##### (Description of fields on form)

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Entity Name | Element Name | Definition | Type | Storage Format | Scale | Bounds | Display Format | Mandatory Entry/Fill | Default Value | Modified by | Read by | Constraints |
|  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |

### Use Case Description: System Preferences – Units of Measure

Actor: User

System: 0. User open Kitchen Manager

1. System display Main Screen
2. User log into user account
3. System display System Preferences
4. User select Unit of Measure options
5. System display available Unit of Measure options
6. User select from available unit of Measure
7. System change units of measurement

#### Screen shot of forms related to Use Case

##### (Description of fields on form)

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Entity Name | Element Name | Definition | Type | Storage Format | Scale | Bounds | Display Format | Mandatory Entry/Fill | Default Value | Modified by | Read by | Constraints |
|  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |

### Use Case Description: System Preferences – Background Change

Actor: User

System: 0. User open Kitchen Manager

1. System display Main Screen
2. User log into user account
3. System display System Preferences
4. User select Background Change options
5. System display available Background Change options
6. User select from available background options
7. System update background

#### Screen shot of forms related to Use Case

##### (Description of fields on form)

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Entity Name | Element Name | Definition | Type | Storage Format | Scale | Bounds | Display Format | Mandatory Entry/Fill | Default Value | Modified by | Read by | Constraints |
|  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |

# \* Glossary

A

Add New Recipe Screen – Allows user to manually enter new recipe ingredients and cooking instructions

C

Current User Details Screen – Shows current users details and preferences

D

Depleted Food Items Browser – Allows user to browse through depleted food items

Depleted Food Items Screen – Shows items that are depleted

F

Food Inventory Browser - Allows user to see the inventory for each food item

Food Inventory Database – All food in inventory

Food Inventory Screen

Food Item Entry Screen – Screen that pops up after a bar code has been scanned. Allows user to update quantity

Food Items Browser – Allows user to browse through food on main screen

Food Items Screen -

Food Object Details List – Allows user to see caloric information on food items

Food Object – Food item

G

Grocer Databases – Items that grocer offers for sale

Grocer Membership Card w/ Personal Barcode (Optional)

I

In-Stock Food Items Inventory Screen – Shows items that are in stock

In-stock Food Items Browser – Allows user to browse through in-stock food items

L

Language Translator – Allows user to switch between available languages

M

Main Screen – System loads to main screen, user interaction screen

Menu List -

Menu Screen-

N

Non-existing Food Item Entry Screen – Allows user to manually enter item that is not in UPC database

Nutrition Information Database – Database that stores nutritional information for most known foods

P

Printer – Allows user to print shopping list

R

Recipe List – List of recipes that have been put into the system

Recipe Relevancy Relationship Engine – Engine that matches recipes depending on what kind of food items are in the cart

Recipe Screen - Shows ingredients and instructions of how to make recipe

S

Shopping list – Shows depleted items for user to buy

Shopping List Screen – Shows items user has added to shopping list

System Color Editor – Allows user to switch between available screen colors

System Preferences Browser – Allows user to look through system preferences

System Preferences Screen – Allows user to change preferences

T

Touch Screen Keyboard – Pop up keyboard that is sensitive to touch. Allows user to input information

Touch-Screen Personal Computer – Devise that user interacts with

U

Units of Measurement Converter – Allows user to switch between different measurement standards

UPC Database – Database that has most known UPC codes

UPC Scanner – Devise reads barcode, used to scan in items

USB Weight Scale (Optional) – A scale to measure users’ weight

User – Person that interacts with system

User Database – Stores users profiles

User Management Screen -

# Version History

|  |  |
| --- | --- |
| Release Date | Description |
| 14-Mar-10 | Version 1.0 - Setup Table of Contents; Defined chapters and sections |
| 16-Mar-10 | Version 1.1 – System Hierarchy, Class Diagrams, Pseudo code, Sequence Diagrams, Use Cases, Data design, Glossary inserted |
| 6-Apr-10 | Version 2.0 – Working Final draft with updated sections. Sections that are not complete or that need updating are marked with an asterisk |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |