

**DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING
THE UNIVERSITY OF TEXAS AT ARLINGTON**

**SYSTEM REQUIREMENTS SPECIFICATION
CSE 4316: SENIOR DESIGN I
FALL 2021**



**UTA STEAM
GROCO**

**PATRICK FAULKNER
HOZEFA TANKIWALA
ANDREW HANDS
KIRAN KARKI
UYEN DO**

REVISION HISTORY

Revision	Date	Author(s)	Description
0.1	10.07.2021	PF	document creation
0.2	10.15.2021	PF, HT, AH, KK, UD	first draft
0.3	10.12.2015	PF, HT, AH, KK, UD	complete draft

CONTENTS

1	Product Concept	9
1.1	Purpose and Use	9
1.2	Intended Audience	9
2	Product Description	10
2.1	Features & Functions	10
2.2	External Inputs & Outputs	10
2.3	Product Interfaces	11
3	Customer Requirements	14
3.1	The application must allow to perform grocery items	14
3.1.1	Description	14
3.1.2	Source	14
3.1.3	Constraints	14
3.1.4	Priority	14
3.2	The application must present user with the best grocery item	14
3.2.1	Description	14
3.2.2	Source	14
3.2.3	Constraints	14
3.2.4	Priority	14
3.3	The application must allow user to filter their search	14
3.3.1	Description	14
3.3.2	Source	14
3.3.3	Constraints	15
3.3.4	Priority	15
3.4	The application must allow users to create grocery lists	15
3.4.1	Description	15
3.4.2	Source	15
3.4.3	Constraints	15
3.4.4	Priority	15
3.5	The application must allow users to search their grocery list	15
3.5.1	Description	15
3.5.2	Source	15
3.5.3	Constraints	15
3.5.4	Priority	15
3.6	The application must pull relevant information from more than one store.	15
3.6.1	Description	15
3.6.2	Source	16
3.6.3	Constraints	16
3.6.4	Priority	16
3.7	The application must provide the most optimal route to the user	16
3.7.1	Description	16
3.7.2	Source	16
3.7.3	Constraints	16
3.7.4	Priority	16
3.8	The application must allow users to view and share recipes	16

3.8.1	Description	16
3.8.2	Source	16
3.8.3	Constraints	16
3.8.4	Priority	16
3.9	The application must allow users to add all ingredients from a recipe to their grocery list	16
3.9.1	Description	16
3.9.2	Source	17
3.9.3	Constraints	17
3.9.4	Priority	17
3.10	The application must allow user to keep a list of favorite grocery items	17
3.10.1	Description	17
3.10.2	Source	17
3.10.3	Constraints	17
3.10.4	Priority	17
4	Packaging Requirements	18
4.1	Web Accessible	18
4.1.1	Description	18
4.1.2	Source	18
4.1.3	Constraints	18
4.1.4	Priority	18
5	Performance Requirements	19
5.1	Grocery or Recipe Search Action	19
5.1.1	Description	19
5.1.2	Source	19
5.1.3	Constraints	19
5.1.4	Standards	19
5.1.5	Priority	19
5.2	Time to Load A Menu	19
5.2.1	Description	19
5.2.2	Source	19
5.2.3	Constraints	19
5.2.4	Priority	19
5.3	Add Item to List Action	19
5.3.1	Description	19
5.3.2	Source	19
5.3.3	Constraints	19
5.3.4	Priority	20
5.4	Compile Shopping Plan Action	20
5.4.1	Description	20
5.4.2	Source	20
5.4.3	Constraints	20
5.4.4	Priority	20
5.5	Load Meal Plan Action	20
5.5.1	Description	20
5.5.2	Source	20
5.5.3	Constraints	20

5.5.4	Priority	20
6	Safety Requirements	21
6.1	Laboratory equipment lockout/tagout (LOTO) procedures	21
6.1.1	Description	21
6.1.2	Source	21
6.1.3	Constraints	21
6.1.4	Standards	21
6.1.5	Priority	21
6.2	National Electric Code (NEC) wiring compliance	21
6.2.1	Description	21
6.2.2	Source	21
6.2.3	Constraints	21
6.2.4	Standards	21
6.2.5	Priority	21
6.3	RIA robotic manipulator safety standards	21
6.3.1	Description	21
6.3.2	Source	22
6.3.3	Constraints	22
6.3.4	Standards	22
6.3.5	Priority	22
6.4	University Group Study Room Policy	22
6.4.1	Description	22
6.4.2	Source	22
6.4.3	Constraints	22
6.4.4	Standards	22
6.4.5	Priority	22
7	Maintenance & Support Requirements	23
7.1	Source Code Documentation	23
7.1.1	Description	23
7.1.2	Source	23
7.1.3	Constraints	23
7.1.4	Priority	23
7.2	Amazon Web Services Maintenance	23
7.2.1	Description	23
7.2.2	Source	23
7.2.3	Constraints	23
7.2.4	Priority	23
8	Other Requirements	24
8.1	Programming Language	24
8.1.1	Description	24
8.1.2	Source	24
8.1.3	Constraints	24
8.1.4	Standards	24
8.1.5	Priority	24
8.2	Cloud Database	24

8.2.1	Description	24
8.2.2	Source	24
8.2.3	Constraints	24
8.2.4	Standards	24
8.2.5	Priority	24
8.3	Registering an account	24
8.3.1	Description	24
8.3.2	Source	24
8.3.3	Constraints	24
8.3.4	Standards	24
8.3.5	Priority	24
8.4	Deleting an account	25
8.4.1	Description	25
8.4.2	Source	25
8.4.3	Constraints	25
8.4.4	Standards	25
8.4.5	Priority	25
8.5	Web Scrape data from a store's website.	25
8.5.1	Description	25
8.5.2	Source	25
8.5.3	Constraints	25
8.5.4	Standards	25
8.5.5	Priority	25
8.6	Update grocery data once a week	25
8.6.1	Description	25
8.6.2	Source	25
8.6.3	Constraints	25
8.6.4	Standards	25
8.6.5	Priority	25
9	Future Items	26
9.1	Android and iOS Applications	26
9.1.1	Description	26
9.1.2	Source	26
9.1.3	Constraints	26
9.1.4	Standards	26
9.1.5	Priority	26
9.2	Revenue Generation	26
9.2.1	Description	26
9.2.2	Source	26
9.2.3	Constraints	26
9.2.4	Standards	27
9.2.5	Priority	27
9.3	Recipe Rating	27
9.3.1	Description	27
9.3.2	Source	27
9.3.3	Constraints	27
9.3.4	Standards	27

9.3.5	Priority	27
9.4	Recipe Review	27
9.4.1	Description	27
9.4.2	Source	27
9.4.3	Constraints	27
9.4.4	Standards	27
9.4.5	Priority	27
9.5	Recipe Reporting	27
9.5.1	Description	27
9.5.2	Source	28
9.5.3	Constraints	28
9.5.4	Standards	28
9.5.5	Priority	28
9.6	Enhanced Search Algorithm	28
9.6.1	Description	28
9.6.2	Source	28
9.6.3	Constraints	28
9.6.4	Standards	28
9.6.5	Priority	28

LIST OF FIGURES

1	High level overview of the system	9
2	Product logo	11
3	End-user product interface	13

1 PRODUCT CONCEPT

This section describes the purpose, use, and intended user audience for the Groco product. Groco is a grocery shopping web application that is accessible on PC, smartphones, and tablets. The user of Groco will be able to search for the grocery items and based on the user's preference, the system will suggest the optimal grocery items. The system also allows users to search for recipes, add their recipes and meal plans into their shopping list to perform the optimization and navigation route.

1.1 PURPOSE AND USE

The purpose of this product is to help users with grocery shopping. The product can:

- look for the item's availability in nearby stores
- save money by comparing prices of the item in different stores
- save traveling time for shopping by comparing the distance to stores
- provide the convenience in planning meals

1.2 INTENDED AUDIENCE

The application is a web application and users can access the application from multiple devices. The product is made available publicly and free of charge. The product contains no objectionable material; therefore it is suitable for general grocery shoppers.

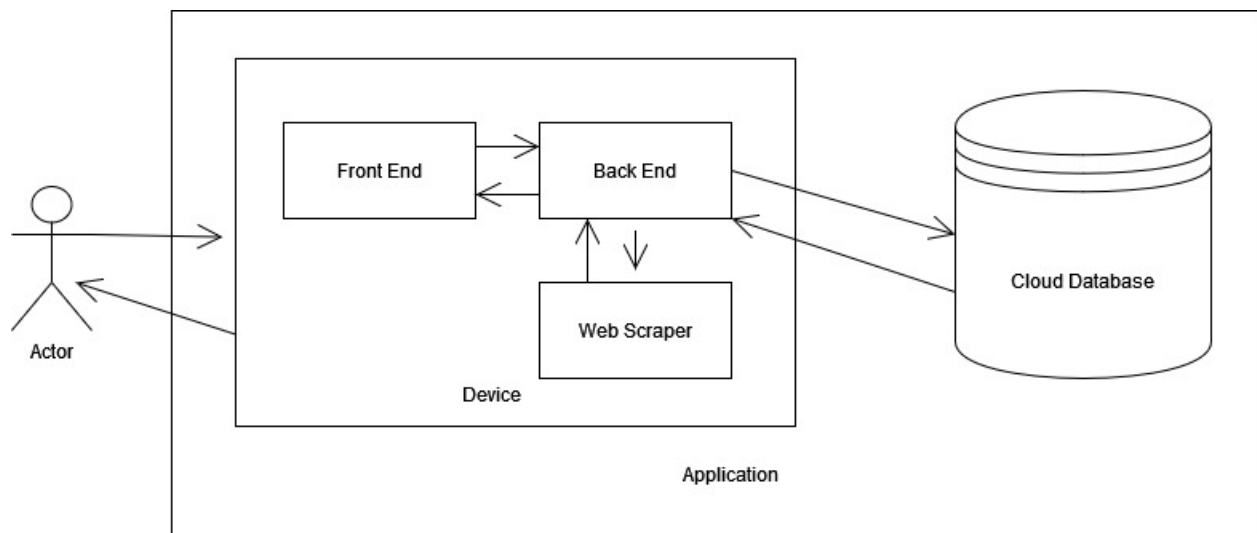


Figure 1: High level overview of the system

2 PRODUCT DESCRIPTION

Groco is a shopping application that provides the optimal suggestion to shop for groceries. The optimization is based on the user's preferences such as brands, prices, and distance. Groco includes five main components: item search, recipes search, adding recipes, adding meal plan, and navigation.

2.1 FEATURES & FUNCTIONS

The product suggests the optimal grocery items based on the user's preference. The product consists of five main components:

- Atomic item search: A user can search for each item to check its availability and prices.
- Recipes search: A user can search for a specific recipe and add it to a shopping list.
- Adding customized recipes: A user can add his or her recipes for later use or share with other users.
- Adding meal plans: A user can multiple recipes to create meal plans. The product will aggregate all items and amounts for shopping.
- Store navigation: A user can click on the map link to navigate to stores.

The product provides the optimal suggestion based on the database and is not responsible for the accuracy or changes regarding prices and discounts at the actual stores. The product does not provide online purchases and online reservation services.

The external requirements for this product include the web browser, internet, a map application, and GPS satellite.

2.2 EXTERNAL INPUTS & OUTPUTS

Describe critical external data flows. What does your product require/expect to receive from end users or external systems (inputs), and what is expected to be created by your product for consumption by end users or external systems (outputs)? In other words, specify here all data/information to flow into and out of your systems. A table works best here, with rows for each critical data element, and columns for name, description and use.

Data	Description and use	In/Out
UserID	Unique username to identify user	In
Password	Input associated with UserID for authentication	In
Email	Input associated with UserID to retrieve password	In
Recipe	Search for recipe	In
Recipe	Add customized recipe	In
Recipe	Display found recipe	Out
Ingredients	Ingredients of some recipe	Out
Grocery	Search for item by keyword	In
Grocery price	Display price of the item	Out
Store's name	Display store that carries the item	Out
Store's distance	Display distance from user's location to store	Out
Total price	Display total price of shopping list	Out
Meal plan	Add meal plan to shopping list	In
Meal plan	Display saved meal plan	Out
Shopping list	Display all items for shopping	Out

Table 2: External inputs and outputs

2.3 PRODUCT INTERFACES

The product is a web application with a user-friendly interface. Below are the sample screenshots of the operational (visible) interfaces for end-user.



Figure 2: Product logo

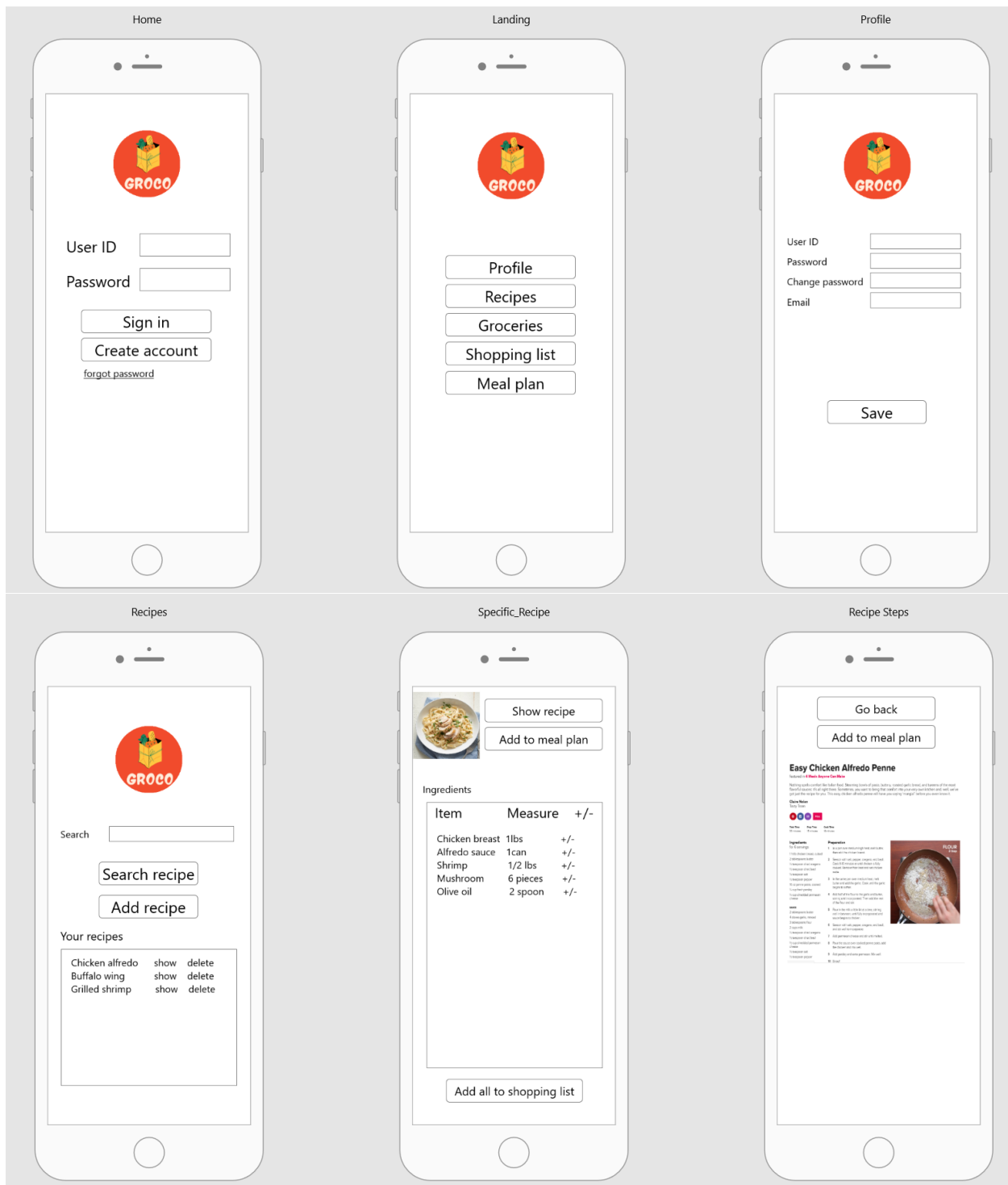




Figure 3: End-user product interface

3 CUSTOMER REQUIREMENTS

The customer requirements for this application have been procured directly from the client, Tim Dockins. The client has a very specific vision about the functionality of this application and has provided us with requirements that he thinks should be a part of the application.

3.1 THE APPLICATION MUST ALLOW TO PERFORM GROCERY ITEMS

3.1.1 DESCRIPTION

A search functionality must be implemented into the application which will allow the user to search for a specific grocery item and then allows the user to add that item to their shopping/grocery list. User must be allowed to type in the grocery item they are looking for and the search should find the matching item for user to add to list.

3.1.2 SOURCE

Customer

3.1.3 CONSTRAINTS

There are chances that the search functionality may not be able to find the item that user is looking for.

3.1.4 PRIORITY

High

3.2 THE APPLICATION MUST PRESENT USER WITH THE BEST GROCERY ITEM

3.2.1 DESCRIPTION

Based on the item that the user searched for, the application should go and look for the best possible match for that item based on the price of the item and location of the stores that item is available in. The application may list multiple options for the same item depending on the item's price and store's location.

3.2.2 SOURCE

Customer

3.2.3 CONSTRAINTS

The item searched for may not be available at any nearby store. Have to deal with the legality of web scraping from big stores.

3.2.4 PRIORITY

High

3.3 THE APPLICATION MUST ALLOW USER TO FILTER THEIR SEARCH

3.3.1 DESCRIPTION

The user should be able to put conditions on their search to filter the results. The user must be able to filter their search using constraints like price, brands, stores and distance from their current location.

3.3.2 SOURCE

Customer

3.3.3 CONSTRAINTS

There might be just one option available, not allowing the user to filter their search.

3.3.4 PRIORITY

Moderate

3.4 THE APPLICATION MUST ALLOW USERS TO CREATE GROCERY LISTS

3.4.1 DESCRIPTION

Other than searching for a single grocery item, the application must allow user to create grocery lists for their items. The grocery list must store multiple items. The user should be able to items to their grocery list by searching for the item and selecting a match to be added to the grocery list.

3.4.2 SOURCE

Customer

3.4.3 CONSTRAINTS

No applicable constraints.

3.4.4 PRIORITY

High

3.5 THE APPLICATION MUST ALLOW USERS TO SEARCH THEIR GROCERY LIST

3.5.1 DESCRIPTION

The application must allow the user the search all items present in their grocery list all at once. The user should not have to search for each item individually in their grocery list. Grocery list functionality must have options for user to select how far they are willing to travel to get their groceries and how many stores they are willing to visit.

3.5.2 SOURCE

Customer

3.5.3 CONSTRAINTS

There might be no items in the grocery list or just one item in which the grocery list search should act like item search.

3.5.4 PRIORITY

High

3.6 THE APPLICATION MUST PULL RELEVANT INFORMATION FROM MORE THAN ONE STORE.

3.6.1 DESCRIPTION

When the user does a search on their entire grocery list, the application must pull up results for the items from more then one store. The user can select if he wants to go to more than one store or no. Depending on user decision, the application should come up with relevant results and should follow the constraints which are put there by the user.

3.6.2 SOURCE

Customer

3.6.3 CONSTRAINTS

There may be a case where the user constraints might not be satisfied because all items may not be available in one store.

3.6.4 PRIORITY

Moderate

3.7 THE APPLICATION MUST PROVIDE THE MOST OPTIMAL ROUTE TO THE USER

3.7.1 DESCRIPTION

If the user opts to visit to multiple stores to get their groceries then the application should provide the user with an optimal route and the order in which the user should visit those stores. The application must consider multiple factors in deciding the route and redirect user to a maps application.

3.7.2 SOURCE

Customer

3.7.3 CONSTRAINTS

The user must have Google Maps installed on their phone in order for this functionality to work.

3.7.4 PRIORITY

Moderate

3.8 THE APPLICATION MUST ALLOW USERS TO VIEW AND SHARE RECIPES

3.8.1 DESCRIPTION

The user should be able to look up recipes in the app. They must also be allowed to create their own recipes and save them. The recipes will store both the ingredients for that recipe and the procedure to follow. Users must be able to view and share recipes with other people using the application.

3.8.2 SOURCE

Customer

3.8.3 CONSTRAINTS

There must be a database available with recipes of common dishes pre-stored in it.

3.8.4 PRIORITY

Low

3.9 THE APPLICATION MUST ALLOW USERS TO ADD ALL INGREDIENTS FROM A RECIPE TO THEIR GROCERY LIST

3.9.1 DESCRIPTION

Once a user has added a recipe to their account, the application must allow the user to add all the ingredients of that recipe to their grocery list. The application should do that by parsing every item in their recipe's ingredients list and should ignore the procedure.

3.9.2 SOURCE

Customer

3.9.3 CONSTRAINTS

The user could use short forms for the items in their recipe which may not work while searching for that item.

3.9.4 PRIORITY

Low

3.10 THE APPLICATION MUST ALLOW USER TO KEEP A LIST OF FAVORITE GROCERY ITEMS

3.10.1 DESCRIPTION

When a user searches for a grocery item, the application must give them an option to make that item a favorite. Then the application must keep track of all the items that a user has chosen as favorite by putting them in a list and storing that list into the database.

3.10.2 SOURCE

Customer

3.10.3 CONSTRAINTS

The user must also be allowed to remove an item from favorites list.

3.10.4 PRIORITY

Low

4 PACKAGING REQUIREMENTS

One of the goals is to make this application easily accessible to all users. With this in mind, the application will be presented via a web browser. This will eliminate the need for installation files or hardware and access to certain application stores.

4.1 WEB ACCESSIBLE

4.1.1 DESCRIPTION

This product will be a web application and will be accessible to all users via web browsers on smartphones, tablets, PC, and Mac at a URL that will be specified at a later date.

4.1.2 SOURCE

Customer

4.1.3 CONSTRAINTS

Because this application will only accessible to users via web browser, an internet connection will be required.

4.1.4 PRIORITY

Critical

5 PERFORMANCE REQUIREMENTS

The following is a list of requirements regarding performance for the application. For the most part, these requirements are particular for load and query time. The list is not all inclusive, but covers the main features for the application regarding menus and dataflows.

5.1 GROCERY OR RECIPE SEARCH ACTION

5.1.1 DESCRIPTION

The amount of time to perform a recipe or grocery search and print out the results to the user.

5.1.2 SOURCE

The Team

5.1.3 CONSTRAINTS

Once a query is submitted, it can take no longer than four seconds. While the down time on the client can vary, on an average device in a cellular network it should take no more than two seconds, totalling at six seconds to complete. In practice, caching will be an issue so it will not be uncommon for queries to take longer, but the average should be no less than six seconds.

5.1.4 STANDARDS

List of applicable standards

5.1.5 PRIORITY

Low

5.2 TIME TO LOAD A MENU

5.2.1 DESCRIPTION

The amount of time required to any menu, discluding time required to load the contents of any query tables.

5.2.2 SOURCE

The Team

5.2.3 CONSTRAINTS

This action should average no more than 2 seconds and should exceed 5 seconds no more than 5% of the time.

5.2.4 PRIORITY

Low

5.3 ADD ITEM TO LIST ACTION

5.3.1 DESCRIPTION

The amount of time required to add an item to a shopping list.

5.3.2 SOURCE

The Team

5.3.3 CONSTRAINTS

This action should average no more than 500 milliseconds.

5.3.4 PRIORITY

Low

5.4 COMPILE SHOPPING PLAN ACTION

5.4.1 DESCRIPTION

The amount of time required to take a shopping list and choose stores and a travel route for the user.

5.4.2 SOURCE

The Team

5.4.3 CONSTRAINTS

This action should average no more than sixty seconds.

5.4.4 PRIORITY

Moderate

5.5 LOAD MEAL PLAN ACTION

5.5.1 DESCRIPTION

The amount of time required to import a meal plan into the current shopping list.

5.5.2 SOURCE

The Team

5.5.3 CONSTRAINTS

This action should average no more than sixty seconds.

5.5.4 PRIORITY

Moderate

6 SAFETY REQUIREMENTS

Include a header paragraph specific to your product here. Safety requirements might address items specific to your product such as: no exposure to toxic chemicals; lack of sharp edges that could harm a user; no breakable glass in the enclosure; no direct eye exposure to infrared/laser beams; packaging/grounding of electrical connections to avoid shock; etc.

6.1 LABORATORY EQUIPMENT LOCKOUT/TAGOUT (LOTO) PROCEDURES

6.1.1 DESCRIPTION

Any fabrication equipment provided used in the development of the project shall be used in accordance with OSHA standard LOTO procedures. Locks and tags are installed on all equipment items that present use hazards, and ONLY the course instructor or designated teaching assistants may remove a lock. All locks will be immediately replaced once the equipment is no longer in use.

6.1.2 SOURCE

CSE Senior Design laboratory policy

6.1.3 CONSTRAINTS

Equipment usage, due to lock removal policies, will be limited to availability of the course instructor and designed teaching assistants.

6.1.4 STANDARDS

Occupational Safety and Health Standards 1910.147 - The control of hazardous energy (lockout/tagout).

6.1.5 PRIORITY

Critical

6.2 NATIONAL ELECTRIC CODE (NEC) WIRING COMPLIANCE

6.2.1 DESCRIPTION

Any electrical wiring must be completed in compliance with all requirements specified in the National Electric Code. This includes wire runs, insulation, grounding, enclosures, over-current protection, and all other specifications.

6.2.2 SOURCE

CSE Senior Design laboratory policy

6.2.3 CONSTRAINTS

High voltage power sources, as defined in NFPA 70, will be avoided as much as possible in order to minimize potential hazards.

6.2.4 STANDARDS

NFPA 70

6.2.5 PRIORITY

Critical

6.3 RIA ROBOTIC MANIPULATOR SAFETY STANDARDS

6.3.1 DESCRIPTION

Robotic manipulators, if used, will either housed in a compliant lockout cell with all required safety interlocks, or certified as a "collaborative" unit from the manufacturer.

6.3.2 SOURCE

CSE Senior Design laboratory policy

6.3.3 CONSTRAINTS

Collaborative robotic manipulators will be preferred over non-collaborative units in order to minimize potential hazards. Sourcing and use of any required safety interlock mechanisms will be the responsibility of the engineering team.

6.3.4 STANDARDS

ANSI/RIA R15.06-2012 American National Standard for Industrial Robots and Robot Systems, RIA TR15.606-2016 Collaborative Robots

6.3.5 PRIORITY

Critical

6.4 UNIVERSITY GROUP STUDY ROOM POLICY

6.4.1 DESCRIPTION

All group study room users must conform to the University's policies for personal conduct: Visitor Conduct Policy. Damage to equipment in the rooms will result in the loss of the ability to reserve and/or occupy any of these rooms. Noise levels from any conversations and/or equipment must not disturb others. All group study room users are expected to be courteous and leave the room ready for the next group. Unattended personal property may not be used to "hold" a room by any individual or group and may be removed by Libraries' staff to allow others to use the space. Staff reserves the right to enter any group study room at any time. Any obstruction to the windows is prohibited. The University and its staff are not responsible for unattended, lost, stolen, or damaged personal items.

6.4.2 SOURCE

University of Texas - Arlington Group Study Room Policy

6.4.3 CONSTRAINTS

Current UTA students, faculty, and staff may reserve group study rooms with a valid UTA NetID (reserve now). A person may reserve a study room in 30-minute increments up to a maximum of 3 hours per individual per day. Reservations can be made up to 14 days in advance.

6.4.4 STANDARDS

6.4.5 PRIORITY

Critical

7 MAINTENANCE & SUPPORT REQUIREMENTS

For maintenance on this project, the team at present has no expectation for continuing support on this project after its completion. The following are a list of deliverables and notes for the support team, which the customer may choose to appoint at a later date.

7.1 SOURCE CODE DOCUMENTATION

7.1.1 DESCRIPTION

The documentation within and possibly extracted from the source code, which will define how each class and public method behaves.

7.1.2 SOURCE

Development Team

7.1.3 CONSTRAINTS

Can only be implemented as time allows for this project and given the development process.

7.1.4 PRIORITY

Moderate

7.2 AMAZON WEB SERVICES MAINTENANCE

7.2.1 DESCRIPTION

For the application to remain active and usable, Amazon Web Services (AWS) payments must be maintained and monitored for cloud features.

7.2.2 SOURCE

Development Team

7.2.3 CONSTRAINTS

Must be continually paid depending on user traffic and application overhead.

7.2.4 PRIORITY

Moderate

8 OTHER REQUIREMENTS

8.1 PROGRAMMING LANGUAGE

8.1.1 DESCRIPTION

The project code will be written in react native.

8.1.2 SOURCE

Team

8.1.3 CONSTRAINTS

N/A

8.1.4 STANDARDS

N/A

8.1.5 PRIORITY

High

8.2 CLOUD DATABASE

8.2.1 DESCRIPTION

The project will use AWS EC2 to store user data, grocery list, etc.

8.2.2 SOURCE

Team

8.2.3 CONSTRAINTS

N/A

8.2.4 STANDARDS

N/A

8.2.5 PRIORITY

High

8.3 REGISTERING AN ACCOUNT

8.3.1 DESCRIPTION

The application should be able to register a new user. The application will provide a form based interface for the user to enter credentials.

8.3.2 SOURCE

Team

8.3.3 CONSTRAINTS

N/A

8.3.4 STANDARDS

N/A

8.3.5 PRIORITY

High

8.4 DELETING AN ACCOUNT

8.4.1 DESCRIPTION

The application should be able to delete an existing user account. After a successful deletion the user shall be able to register again using the previous credentials.

8.4.2 SOURCE

Team

8.4.3 CONSTRAINTS

N/A

8.4.4 STANDARDS

N/A

8.4.5 PRIORITY

Moderate

8.5 WEB SCRAPE DATA FROM A STORE'S WEBSITE.

8.5.1 DESCRIPTION

The application shall gather information like price of groceries from a store's website.

8.5.2 SOURCE

Team

8.5.3 CONSTRAINTS

N/A

8.5.4 STANDARDS

N/A

8.5.5 PRIORITY

Moderate

8.6 UPDATE GROCERY DATA ONCE A WEEK

8.6.1 DESCRIPTION

The application shall use the data to find deals for a period of 1 week. After a week is completed the system shall update the price of groceries.

8.6.2 SOURCE

Team

8.6.3 CONSTRAINTS

N/A

8.6.4 STANDARDS

N/A

8.6.5 PRIORITY

Moderate

9 FUTURE ITEMS

This section will cover all future requirements for this application. There is currently no plan to complete these requirements for the prototype version that will be completed for the scope of CSE 4316 and CSE 4317.

9.1 ANDROID AND IOS APPLICATIONS

9.1.1 DESCRIPTION

The prototype version, though accessible on Android and iOS devices, will only be accessible through the browser. Having applications for each type of mobile device is the highest priority of all future requirements. Having these applications will allow users to more easily access the application while creating a more efficient User Interface for mobile devices. The mobile applications should function exactly the same way on both mobile operating systems while looking as close to the same as possible, all of which should be the same as the web application.

9.1.2 SOURCE

UTA STeam

9.1.3 CONSTRAINTS

The application will have to be developed for each operating system at the same time to ensure they perform and look the same. After completion of the mobile applications, they should be made available on both the Apple App Store and Google Play Store. If any updates occur to the system, the team will have to ensure that they work and are pushed to the web, Android, and iOS applications at the same time.

9.1.4 STANDARDS

List of applicable standards

9.1.5 PRIORITY

Future

9.2 REVENUE GENERATION

9.2.1 DESCRIPTION

This application has a lot of potential to generate revenue. The two main ways this can be done would be through data collection or membership access. Data collection could be useful to companies that want to track what certain demographics are shopping for, which zip codes people are shopping in, or how far people are will to drive for certain stores, etc. Memberships could allow for partnerships with other companies. This could allow certain people or companies to release cookbooks in the application, dietitians to create personalized meal plans for users, or allow stores to offer exclusive deals all of which would only be accessible to members.

9.2.2 SOURCE

Customer

9.2.3 CONSTRAINTS

The stakeholders must come to an agreement on how the application will generate revenue. If data collection is decided upon, the development team must ensure that all laws and regulations are followed and allow users to opt out of collection. If membership is chosen, the stakeholders must find partners that could make a membership worth the user's money.

9.2.4 STANDARDS

List of applicable standards

9.2.5 PRIORITY

Future

9.3 RECIPE RATING

9.3.1 DESCRIPTION

Users will have the ability to rate recipes. The rating system will be a standard five-star system. This rating system will allow users to make the best choice when looking at similar recipes.

9.3.2 SOURCE

UTA STeam

9.3.3 CONSTRAINTS

The development team will have to find a way to prevent users from spamming recipes with five star ratings to boost traffic to a recipe. There will also need to be a way to prevent users from spamming a recipe with bad reviews for illegitimate reasons.

9.3.4 STANDARDS

List of applicable standards

9.3.5 PRIORITY

Future

9.4 RECIPE REVIEW

9.4.1 DESCRIPTION

The review system will allow user to give any general comments about the recipe. This will also allow users to share their critiques, helpful tips, and recommendations.

9.4.2 SOURCE

UTA STeam

9.4.3 CONSTRAINTS

There will have to be a way to regulate these comments to ensure that nothing offensive is said in the reviews. There should also be a way to ensure that these reviews are only relevant to the recipe.

9.4.4 STANDARDS

List of applicable standards

9.4.5 PRIORITY

Future

9.5 RECIPE REPORTING

9.5.1 DESCRIPTION

This recipe reporting system will allow users to report recipes. Once they are reported, someone within the team should be tasked for screening these reporting to ensure their legitimacy. This reporting system will help the team ensure that shared recipes are in fact recipes, that recipes do not contain or create

anything harmful or illegal, while also regulating any vulgar or offensive language that might appear in the recipes.

9.5.2 SOURCE

UTA STeam

9.5.3 CONSTRAINTS

The team will have to regulate reporting to ensure that they are legitimate.

9.5.4 STANDARDS

List of applicable standards

9.5.5 PRIORITY

Future

9.6 ENHANCED SEARCH ALGORITHM

9.6.1 DESCRIPTION

The algorithm for the prototype version of this application will provide the most optimal grocery item based only on price and location. In the future, customers will be able to choose the criteria that chooses the most optimal grocery item. This will allow users to give their own priority to brand, price, or location.

9.6.2 SOURCE

Customer

9.6.3 CONSTRAINTS

The team will have to create an algorithm that is able to change based on user preference.

9.6.4 STANDARDS

List of applicable standards

9.6.5 PRIORITY

Future

REFERENCES