Project pitch

Team 75

Team 75

Name	Student number	Email
Lucas Kim	2798807	lucassukeunkim@gmail.com
Sanskar Shrestha	2797647	20216661@life.hkbu.edu.hk
Bora Tarlan	2797649	b.tarlan@student.vu.nl
Sean Hermes	2796616	s.h.hermes@student.vu.nl

Overview

Our LetsCook program is a cookbook system that allows users to easily access & create cooking recipes for their own personal use. Our LetsCook system will provide a variety of features that will ease and combine different steps in cooking dishes into a single system of recipes. The key aspects of our system are:

- creating and editing cooking recipes
- saving & loading recipes as local files
- sorting recipes in customizable categories
- step-by-step execution of recipes

LetsCook is an all-encompassing program perfect for anyone and everyone who likes to cook - not to mention it is very beginner friendly. When the user chooses to execute a recipe, they will be faced with a checklist of all the required ingredients to ensure that they are not missing anything. The user can also create custom categories to sort recipes, resulting in a very personalized and streamlined user experience.

Partial inspiration: https://thecookbookapp.com/

Functional features

ID	Short name	Description	Champion
F1	Customizable categories	On top of default categories of recipes, the users should be able to **customize the categories** of recipes as per their choice. (*In addition to default categories of recipes such as appetizers, desserts, etc., the users should be able to add their own customized categories such as favourites folder.*)	Lucas
F2	Commands	The user shall be able to manage and execute the recipes by typing **command-line instructions** following this syntax: command + [target]. Available commands are: - create + recipe - create + category - update + recipe - update + category - delete + recipe - delete + category - execute + recipe - load + recipe - save + folder (*The project will have a command-line interface so appropriate commands must be typed in order to perform an action.)*	Sanskar

Functional features (Continued)

ID	Short name	Description	Champion
F3	Printable Recipe	**Bonus:** The user should be able to **print the recipe** of their choice without having the need to carry a device in the kitchen. (*It would be convenient for some users to have a printed document of the recipe while cooking.)*	Sean
F4	Convert ingredient measurement	The user shall be able to **convert the measurements** of the ingredients into commonly referenced units such as cups, ounces, grams, kilograms, teaspoons, tablespoons, etc. (*Not all measurements in a recipe would be in similar format, so having the option to convert the measurements into desired format would be helpful for many users.)*	Bora
F5	Recipe ratings	The users shall be able to **rate the recipes** on a scale of 1 to 5 with 1 being the lowest and 5 being the highest. (*Ratings will be stored as an attribute of a Recipe object, and saved locally when the Recipe is saved as a JSON file)*	Lucas

Functional features (Continued)

ID	Short name	Description	Champion
F6	Checklist ingredients	The users shall be able to check whether they have the **ingredients required** for the recipe. (*Checking the list of ingredients might get confusing, so having a checklist for the required ingredients can help the users have an effortless cooking.)*	Sanskar

Quality requirements

ID	Short name	Quality attribute	Description
QR1	System interface coherence	Usability	When the user issues a command, the command line interface will process the command and then display all relevant information and actions available to the user.
QR2	Adaptable recipes	Maintainability	The class structure of the program with respect to recipes will be designed as to allow for new recipe-related features to be easily implemented.
QR3	Speedy runtime	Responsiveness	When the user issues a command, the system will process the command and prompt the user for their next input very quickly.
QR4	Bug-free code	Reliability	All user action paths will execute correctly without ambiguity and all features will work as intended by the programmers.
QR5	Ease of access	Availability	The system can be accessed by the user locally without any dependency on external factors (internet, other users, etc.)

Time log - Assignment 1

Team number	75		
Member	**Activity**	**Week number**	**Hours**
Sanskar Shrestha	Define functional features, define team contract	1	1
Bora Tarlan	Define functional features, define team contract	1	1
Sean Hermes	Define functional features, define team contract	1	1
Lucas Kim	Define functional features, define team contract	1	1
Sanskar Shrestha	Contribute to the creation of the project pitch	1	1
Bora Tarlan	Contribute to the creation of the project pitch	1	1
Sean Hermes	Contribute to the creation of the project pitch	1	1
Lucas Kim	Contribute to the creation of the project pitch	1	1
Sanskar Shrestha	Added functional features to project pitch	1	1
Bora Tarlan	Worked on overview & quality requirements of project pitch	1	1
Sean Hermes	Worked on overview & quality requirements of project pitch	1	1
Lucas Kim	Worked on overview & quality requirements of project pitch	1	1
Sanskar Shrestha	Finalized slideshow	1	0.5
Bora Tarlan	Finalized slideshow	1	0.5
Sean Hermes	Finalized slideshow	1	0.5
Lucas Kim	Finalized slideshow	1	0.5

Signed contract

Google Doc (view):

Team Contract