



University of Puerto Rico Mayagüez, Campus
Faculty of Engineering
Department of Computer Science and Engineering



Team I: Food Choices

(name tbd)

Estefanía Torres Collado

Andrea C. Miranda Acevedo

David Carrión Benítez

Gabriel R. Pantojas Burgos

Christopher Vegerano López

Everson Rodríguez Muñiz

Informative Part

- I. Name, Place, Date
 - a. Food Choices, Mayaguez PR, August 2020
- II. Current Situation
 - a. The current situation is that people are struggling to find an establishment that serves their craving that satisfies their needs based on taste, budget, availability, and location. This causes a delay on the time taken to choose where and what to eat by the person. Furthermore, a wrong decision can lead to unsatisfaction of that person and potentially deteriorate the image or reputation of the establishment.
- III. Needs
 - a. There is therefore a need to enhance discovering the right food establishments that are capable of increasing satisfaction to the customer and decrease the time it takes to reach a decision.
- IV. Ideas (Include more)
 - a. The idea is to provide a platform/system where people can search and find directly the best rated dish of a food establishment that fulfills their craving. Dish ratings will be based by the reviews and votes of numerous food critics and popularity/majority preference. From another perspective, a person could be at a restaurant but doesn't know what to eat, s/he can verify in our system the top or most preferred dishes in that establishment and can read professional reviews, if any, of each dish.
- V. Scope, Span
 - a. The scope is food directory. The span is the creation and management of a crowd-sourced and food critic reviewed food community to explore different food options.
- VI. Synopsis
 - a. The project is to develop from a domain description to the implementation of a food community that helps people make a more precise and faster decision on choosing what and where to eat. The domain model is expected to cover phenomena such as: presentation of a dish with its descriptors like the establishment that serves it, location of the establishment, availability, price of the dish, taste, and popularity. Each dish is expected to be given a rating by a crowd-sourced set of reviews (mostly composed of food critics) and popularity based of the general public.

Descriptive Part

- I. Descriptive Rough Domain Sketches
 - a. A person wants to go out to eat a specific dish that the person is craving at a given moment, but that person wants to eat at the best possible location that serves the best dish that they are desiring. That person could be a tourist or someone that

doesn't know the area all too well, thus reaching a problem that he or she doesn't have a good sense of direction of where to dine where they will satisfy their palate. The person proceeds to look for information on the internet or by asking friends or family members for suggestions of establishments near them that serve the dish they are craving. That way the person can gather different options to decide which is the best option. The best option will be determined by different criteria: economic, distance/location, quantity, quality and/or ratings from source (friend or web). When an option is finalized, the person goes to the establishment, orders the food and proceeds to eat it.

II. Descriptive Domain Narrative

- a. In the domain of food establishments, there are three main notions that further describe and differentiate each from the other. These are: menu, location, price range, rating, and type. A menu is made up of one or more dishes and is created with the food establishment and can be edited. A location is the placement of the establishment in a place during a specific time. A price range is the average amount of money that is spent on one meal at the location. A rating is the overall score given by the clients. The type is the classification or grouping of the food establishment for it to be searchable. By a food establishment we understand that it is a location in which food and drink is sold to customers. A food establishment can be searched for in terms of location, rating, or type. There are several operations which involve or result in food establishments: creating a new food establishment or editing a food establishment. Editing a food establishment can be triggered by a change in rating due to new reviews or a change in location or offering in the menu.

Dishes are a serving available at an establishment that form its menu. They are also composed of other notions to further describe them. These are rating, type, and price range. A rating is the overall score given by the consumers, which is utilized for searching. The type is the classification or grouping of the dish for it to be searchable. The price range is the category of amount of money necessary for purchase. There are several operations which involve or result in dishes. Dishes can be created, edited, or deleted. A dish's rating changes according to the overall score of the users or customers. Dishes can be deleted when an establishment no longer serves it in their menu.

III. Descriptive Domain Terminology

- a. Craving –An intense desire for a particular food that the person wants to eat.
- b. Rating – Score that determines the ranking of a dish based on different categories.
- c. Source –Website or close friend/relative that gives you feedback of a dish.
- d. Dish – Specific plate of food.
- e. Establishment – Place where dishes are prepared and served for consumers that have paid.

- f. Distance – Proximity between a person and an establishment.
- IV. Domain Entities
 - a. Establishment: location where customers go and purchase food.
 - b. Person: user of the web browser and the one that goes to the establishment to satisfy their craving.
 - c. Dish: food that is served in an establishment. In the web application it is the main search engine.
 - i. Price: amount of money that a person must pay for the dish in the establishment to consume it. In the web application it can be utilized as a search to show you the best dish that a person can purchase according to their budget.
 - ii. Rating: the voting system that is utilized in the web application where the dish can get upvoted by different user determining which establishment server the best dish that the user is searching for.
 - iii. Ingredients: ingredients are the components that make up the dish when they are combined.
 - iv. Category:
 - d. Smartphones/computer: Devices that a user can utilize to use the web application to search for the dish that they want to eat at a given time.
 - e. Map: the map consists of the different location of the establishments that have been enlisted in the web application. It will also determine how long it will take you to arrive to the establishment from your current location.
 - f. Menu: List of different dishes that an establishment serve.
 - g. Drink: refreshment that a user drinks after they finish eating to soothe the thirst after they finish eating.
- V. Domain Functions
 - a. Rate_dish(dish, score): assigns/updates the rating score of the specified dish.
 - b. Search_item(): determine all the searchable items within the scope of the domain.
 - c. Delete_dish(): deletes a dish from a menu when the food establishment no longer is serving it, thus making it unavailable for the users.
 - d. Get_menu(): returns the menu of a food establishment, including all the dishes which it is made up of.
 - e. Create_dish(name, ingredients, type): creates a dish in the menu of the food establishment in which it is created.
 - f. Create_establishment(name, location): creates a new food establishment with a menu and a location for which to start adding dishes.
- VI. Domain Events
 - a. Dish was searched

- i. After the user knows what dish they want to consume, the user searches for that dish on the website returning the best rated dish for the search category that they utilized.
 - b. Return location
 - i. After the user decides on the establishment that they want to dine at, the browser will give them the location for the establishment to get there through the fastest route.
 - c. Rating a dish
 - i. A user can upvote a dish within its own category.
- VII. Domain Behaviors
 - a. Rating search
 - i. The website will return the best possible establishment based on the ratings of all the dishes that have been listed on the website. This is the search that will be the most beneficial due to it giving you the best possible dish without any kind of limitation in the search.
 - b. Location search
 - i. The website will return the best possible establishment within a perimeter based on the location that the user inputs.
 - c. Price search
 - i. The website will return the best possible establishment based on the price of all the dishes that have been listed on the website, whether it is the most expensive one or the least expensive.
 - d. Dish rating
 - i. A user can go to the website and decide to upvote an establishment's dish within that dish's category. The user can only upvote for one dish which will determine their favorite establishment that serves that dish. At any point the user can change their mind and upvote another establishment, but they would have to revoke their previous vote.
- VIII. Domain Requirements
 - a. Search a Dish
 - i. A user is able to search for a particular dish based on the search criteria that they decide to utilize whether it's by the dish with the highest rating, by search based on a radius of a specified location (your current location for example), or search based on the cost of the dish.
 - b. Verify if a dish was rated by a "verified" account
 - i. A user can search for dishes that have been rated by a professional chef (that is not affiliated with said establishment) or by a professional food critique/taster, which is someone who makes a living out of tasting different types of dishes or drinks. These types of critiques will have a major impact

on the rating of the dish due to their professional backgrounds, but certain criteria must be met before their rating is said to be “valid”.

c. Best Dish Searcher Website

- i. The final end of the product should be a website where users can login and search for the best dishes due to the ratings of the general public by an upvote system where the user can only vote for a single dish of a particular type (for example mofongo, chicken, steak...). Which will help the user decide on a place that they want to go given a specific dish that they are craving now.

IX. Interface Requirements

- a. shared data initialization requirements – All users should have access to the same data for searching places to eat.
- b. shared data refreshment requirements – Data of the dishes and establishments must be constantly updated according to new reviews changing the score of any of the items.

X. Machine Requirements

a. Performance

- i. The machine shall serve 500 common and 200 verified users, a total of 700 users.
- ii. The machine’s average response time shall be at most 1.6 seconds, when the system is on a heavy load.

b. Dependability

- i. The machine shall always be accessible for users. Every user shall have the appearance that s/he has exclusive access to the system.
- ii. The machine shall always be available, except when under maintenance.
- iii. The machine shall encrypt all stored user sensitive data (passwords, etc.).

c. Maintenance

- i. The machine’s average time between failures shall be at least 30 days, and downtime due to failure shall be less than 3 hours.

d. Platform

- i. The machine shall be developed on a UNIX operating system.
- ii. The machine shall be compatible with all the libraries and 3rd party software for correct execution.

e. Documentation

- i. We shall provide installation, support, user, contribution, and development guidelines.

XI. Software Architecture Design

The Use Case Diagram is used as a primary tool for a new software project underdeveloped. The UML use case diagram represent and specified the expected behavior of the system. Figure 1 represent the connection and behavior of the software that will be developed. There is two kind of user, verified and common users. Those users could see the top-rated dishes and make a review. The system will be capable of show the top/recommended dishes based on the reviews. The common user could search a dish by its location, restaurant, or category. The expected goal is the user could get the best rated dishes of what he/she want to eat at this exact moment, or the best dish to a certain location.

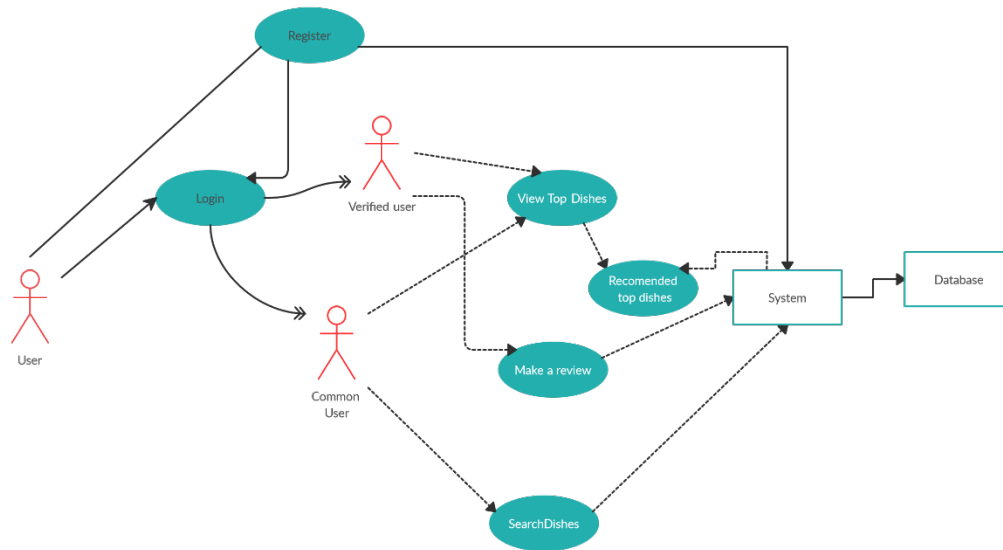
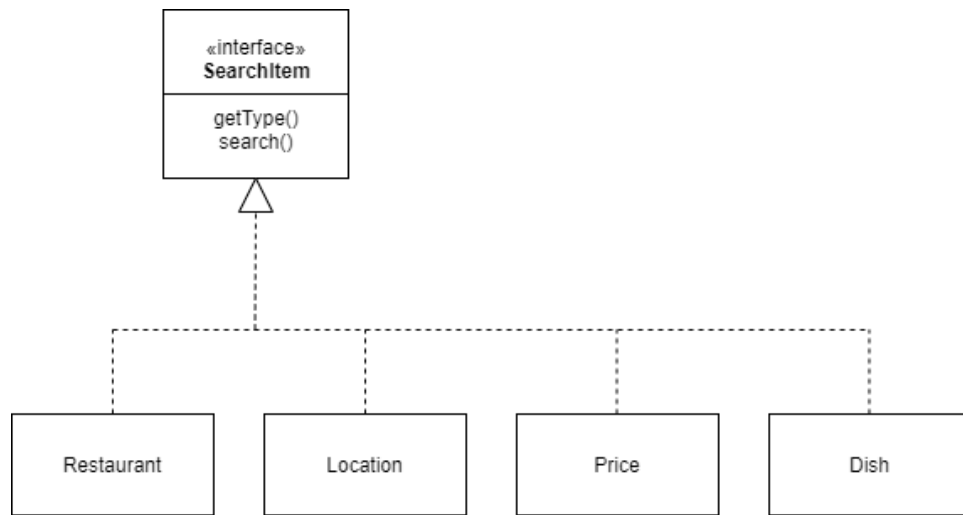


Figure 1: UML Use Case Diagram.

XII. Software Component Design

a. Principles

- i. Design by Contract: It denotes that the relationship between the class and its clients constitute an agreement. In our case, if the user searches for dishes that contain the property of City = "Mayagüez", then the Search List agrees to contain only dishes that are in Mayagüez.
- ii. Open-closed Principle: This principle establishes that there should always be room for extension of a module, but the module should be closed to modifications. Let's take the example of a SearchItem interface. Here, it is ideal to keep adding classes that implement SearchItem, yet it is not ideal to modify the SearchItem interface, when various classes already depend on it.



b. Identify and Elaborate Classes

- i. Dish - has properties like price, rating, and establishment where it is served. It also has functions like rate() to allow a user to give a rating of such dish.
- ii. Restaurant - has properties like location and rating. Has a list data structure that contains the menu items they serve. Has functions like getTopDishes() to get the best they have to offer.
- iii. Location - has properties like latitude and longitude. Has functions like getDistance() that returns the distance of that location with respect to the user.
- iv. User - has properties like name, age, isVerified (for users whose food opinions weigh more). Has functions like getLocation() to determine the current location of the user.

XIII. Selected Fragments of Implementation

- a. No code for now (will be added in the future, hopefully starting next week)

Analytic Part

I. Concept Formation

- a. There are the concrete phenomena of meals, drinks, desserts, and starters. We abstract these concepts into one representation: dishes. There are also the concrete phenomena of bars, restaurants, fast food restaurants, and “chinchorros”. These can be abstracted into the concept of food establishments.

II. Validation

- a. Domain validation report:
 - i. Descriptive Domain Terminology – The craving terminology is not necessary for the domain. It is not a factor present at all instances of the

domain. The source is also not always present and does not account to any other terminology such as rating.

- ii. Domain Entities – The drink entity can be abstracted with dishes because they serve the same function in the scope of the food establishment, they are not treated differently by the users if they are searching a drink. It can be specified as a type of dish
- iii. Domain behaviors – In the rating search it must be specified that dishes are being searched by type for the results to reflect the best rated dish in that specified type.
- iv. Domain requirements – It is not specified how an account is “verified” and how much it affects the overall rating of a dish.

III. Verification

- a. Verification by informal reasoning: A dish is a representation of an item that is sold for consumption; therefore, a dish can be of different types, it is not only defined by complete “meals”, but as an item in the menu of the establishment. Those include drinks and snacks, depending on the type of establishment.

Roles

Note: the role of SCRUM Master will be changed at the beginning of each new sprint.

1. Estefanía Torres Collado
 - a. Role: Frontend Leader
 - b. Responsibilities
 - i. Create wireframes for the UI
 - ii. Help in the frontend development
2. Andrea C. Miranda Acevedo
 - a. Role: Frontend member
 - b. Responsibilities
 - i. Review peer code
 - ii. Work in the frontend development
3. David Carrion Benítez
 - a. Role: Frontend
 - b. Responsibilities
 - i. Review peer code
 - ii. Part of frontend development
4. Gabriel R. Pantojas Burgos
 - a. Role: back end member
 - b. Responsibilities
 - i. Review peer code

- ii. Do part of the back-end development
- 5. Christopher Vegerano Lopez
 - a. Role: Back end
 - b. Responsibilities:
 - i. Review peer code
 - ii. Do part of the back-end development
- 6. Everson Rodríguez Muñiz
 - a. Role: Back End Leader
 - b. Responsibilities
 - i. Organize the team for weekly meetings and determine what is going to be discussed.
 - ii. In charge of development and correct integration of the back end.
 - iii. Create initial ERD diagram.
 - iv. Scrum master for the first sprint (starting next week).