Describe the requirements of the problem with a simple document that lists the rules of the database in the problem domain language. Then from that list of rules and notes highlight the list of possible nouns and actions you identified. I'm expecting this to be a short 1 or 2 pages document.

Verbs & nouns

Problem:
Within the local community, there is a challenge in identifying and promoting the best drinks and desserts available across various cafes. Visitors often find it difficult to decide where to order, and once they make a choice, there is a risk of the selected item being misrepresented on the menu. This discrepancy between expectation and reality can result in unsatisfied cravings and a less-than-optimal dining experience. The lack of accurate and detailed information about the offerings at different cafes makes it challenging for people to discover the hidden gems within the community's beverage and dessert scene.

Objective:
Our objective is to create a database that catalogs cafes within the local community, listing the various foods and drinks available at each cafe. This database allows users to explore and compare different cafes based on the specific types of drinks and foods they offer. By compiling this information the goal is to recommend cafes to the community and allow them to see which cafe is best for a type of craving, encouraging them to discover different locations in the area.

Rules:
- Catalog the different drinks and desserts in each cafe
- Identify and compare cafes to find which ones are most to least recommended
- Identify and compare drinks and desserts of each cafe to find which ones are most to least recommended
- Promote recommended cafes based on different communities and locations

List of Verbs
- Identify
- Promote
- Challenging
- Discover
- Catalog
- Explore
- Compare
- compile
- Recommend
- Encourage

List of Nouns
- Cafe

- Desserts
- Drinks
- Menu
- Database
- Community
- Location

Potential Tables:
User
- user_id: PK
- username
- password
- first_name
- Last_name

User Preference
- MinPrice
- MaxPrice
- cafeType
- location

User Feedback
- FeedbackText: String
- FeedbackDate: Date

cafe
- cafe_id: PK
- cafe_name
- location
- cafe_type

Menu
- sections: [food, drinks]
- seasonal: String

Fooditem
- foodItem_id: PK
- price
- description
- date_added

Beverage
- beverage_id: PK
- price
- description
- date_added

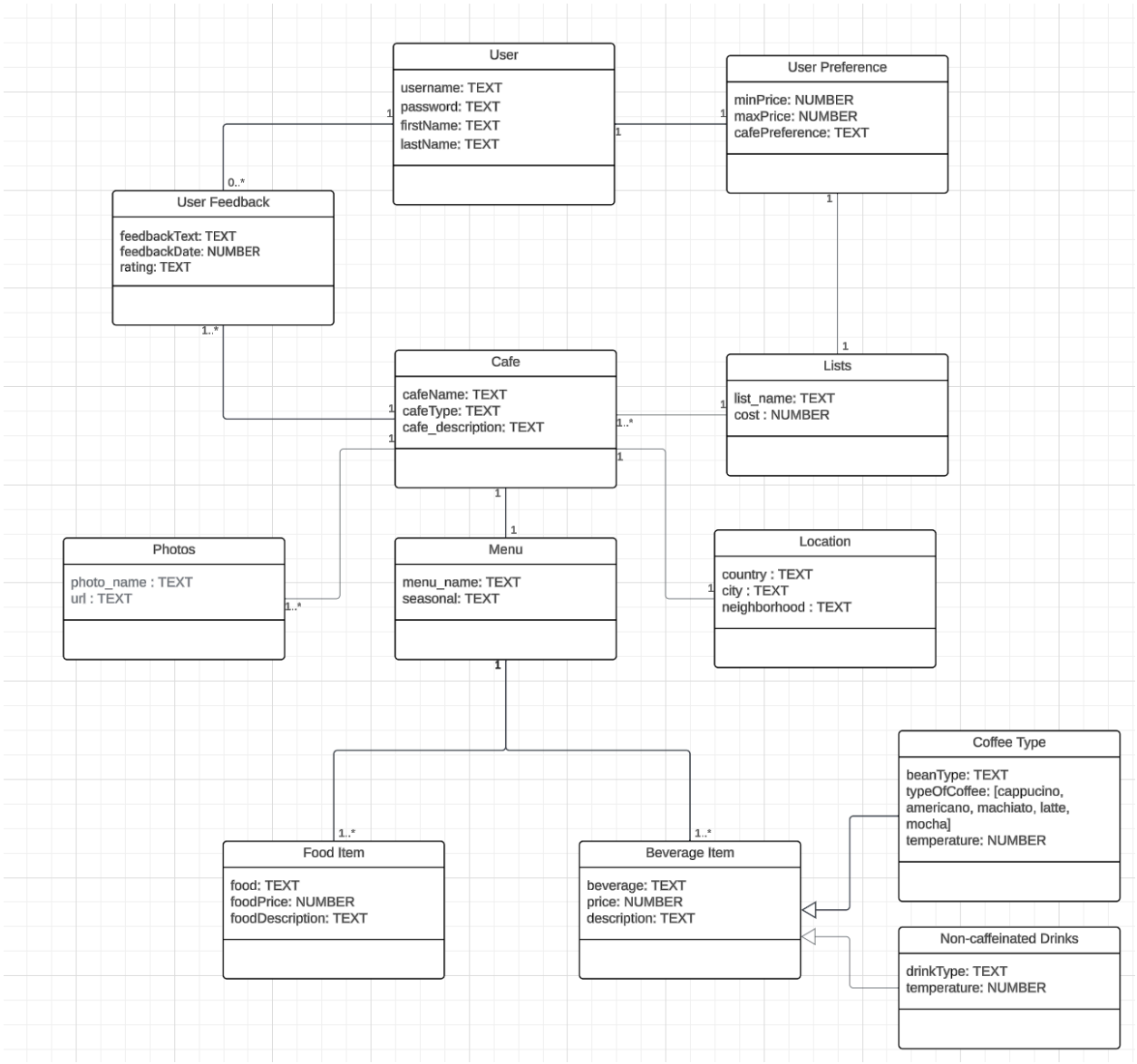
CoffeeType
- coffee_id: PK
- beanType
-

Ingredients
- Ingredients
- possibleAllergens

Reviews
- review_id: PK
- foodItem_id: PK
- drinkItem_id: PK
- rating

foodImage
- review_image: PK
- foodItem_id: FK
- drinkItem_id: FK

Analyze the problem and create a conceptual model in UML using a tool of your choice (e.g., LucidChart, Enterprise Architect, ArgoUML, Visual Paradigm, ERwin, TOAD) as discussed during class and provided in the references and resources below. Additional requirements and clarifications will be provided in the #general channel on Slack. The diagram must contain at least three classes, at least one to many relationship and one many to many. All relationships, except generalization, must have full multiplicity constraints and labeled as appropriate. Classes must have proper names, descriptions, and attributes with domain types. Key attributes and derived attributes must be marked. Don't build a model with more than 10 entities.



https://lucid.app/lucidchart/b5bcaf42-4e43-4189-8512-967e79676a02/edit?viewport_loc=1796%2C-2170%2C4042%2C1819%2C0_0&invitationId=inv_d7bb0b18-79b1-4113-9f65-92067be66e73

Project part 2