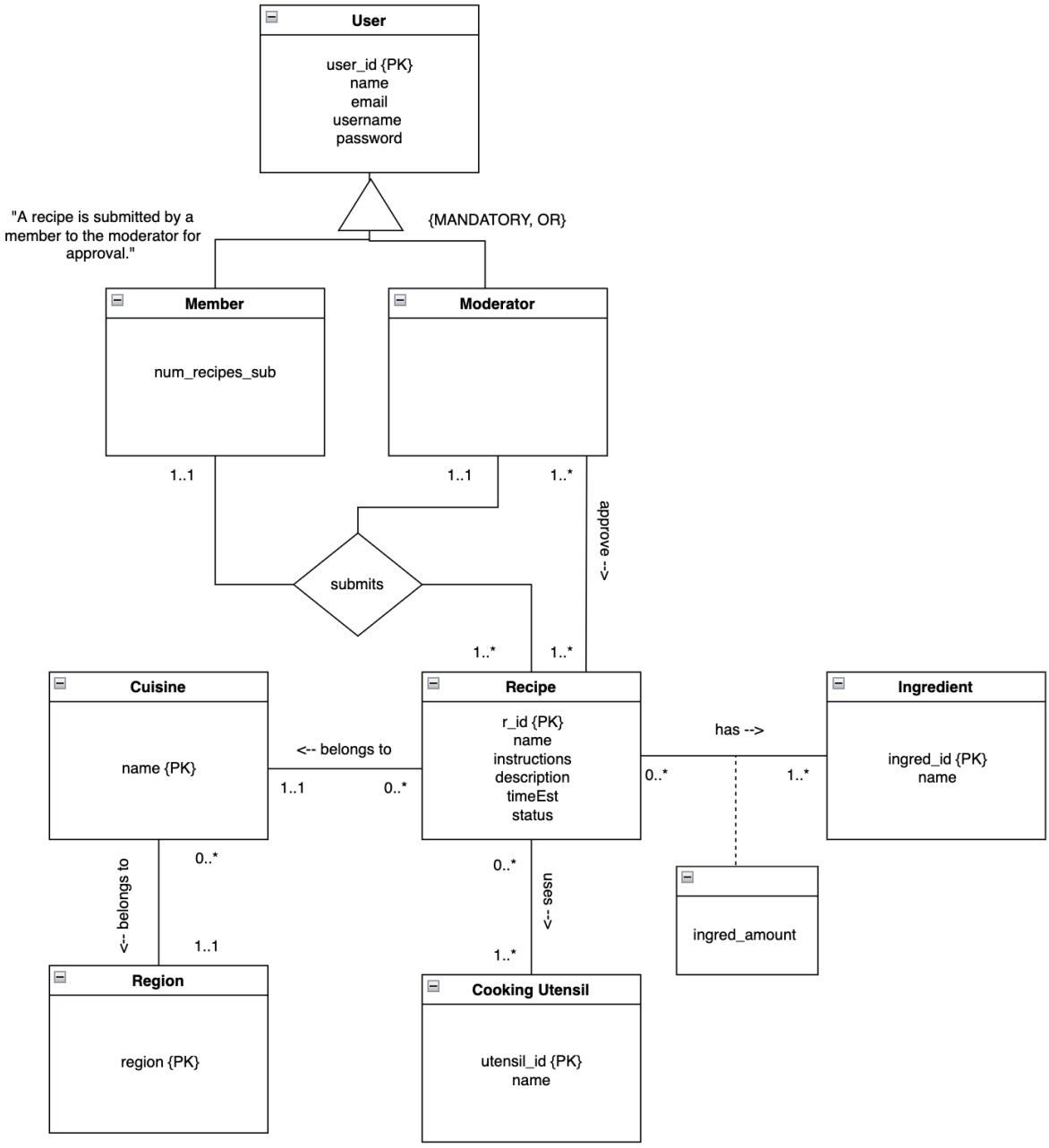
Kirti Magam and Riya Gurnani
Professor Durant
CS 3200
Final Report

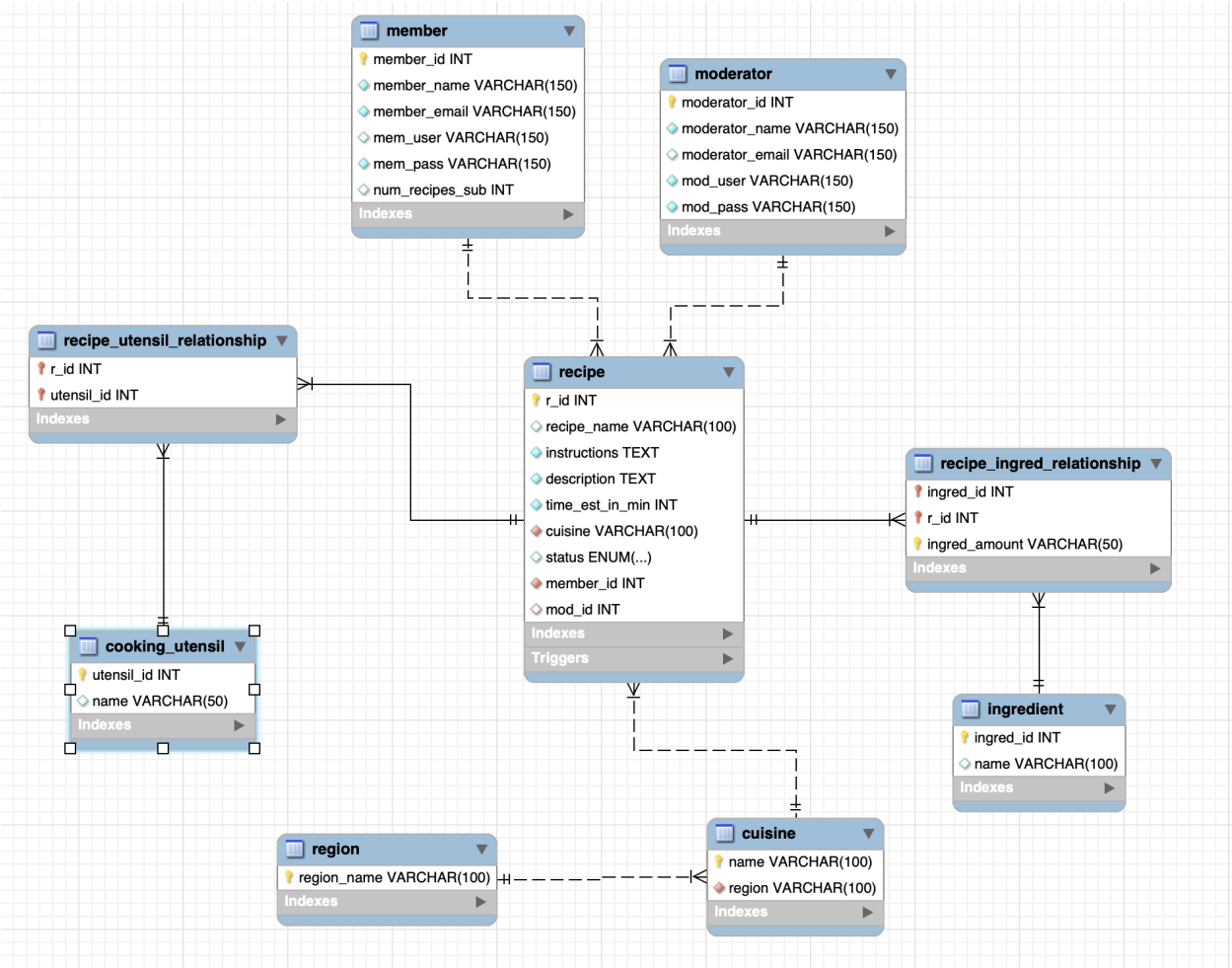
1. README
   Download the GurnaniRMagamK_project.sql file and run it in MySQL Workbench to have access to the recipes database on your device. Download the spyder file which is called GurnaniRMagamK_project.py. After going to the correct directory for the spyder file in the terminal, run the command 'python' followed by the name of the file in the terminal to run the file.
   Another option of running the file is downloading Spyder. Spyder is available to use through Anaconda. Here is the link to downloading Anaconda on your computer: https://docs.anaconda.com/anaconda/install/index.html.
   Once you have the application installed, launch Spyder through there. You can open the file in Spyder and press run at the top of the screen. Answer the input questions in the console.
   Once the code is running, follow the prompts to know what to type based on what action you want to complete. For more detailed instructions, refer to the user flow.

2. We used MySQL Workbench to build our database and its procedures, functions, and triggers. We then utilized Jupyter Notebook and Google Colab to write code with python to add user interaction and connect to our database using pymysql. This allowed for easier collaboration. However, we did end up switching the python code to Spyder for greater ease of testing.

3. UML:


User
user_id {PK}
name
email
username
password
{MANDATORY, OR}
"A recipe is submitted by a member to the moderator for approval."
Member
num_recipes_sub
Moderator
1..1
1..1
1..*
approve -->
submits
1..*
1..*
Cuisine
name {PK}
<-- belongs to
1..1
0..*
Recipe
r_id {PK}
name
instructions
description
timeEst
status
has -->
0..*
1..*
Ingredient
ingred_id {PK}
name
0..*
<-- belongs to
1..1
0..*
uses -->
1..*
ingred_amount
Region
region {PK}
Cooking Utensil
utensil_id {PK}
name

4. Reverse engineer:



5. First the user is prompted to enter in their username and password to connect to the SQL database
    - Then the user types in if they're a 'member'
        - The user is prompted to enter 'yes' if they need to sign up
            - They are prompted to enter their profile name, email, username, and password to be added as a member
                - So long as they are not trying to duplicate another existing member, this will create their registration and log them in
        - Or 'no' if they are already registered as a member and need to just log in
            - They are prompted to enter their member username and password & are successfully logged in as long as that combination is for an actual registered member (exists in the member table)
                - If the login is unsuccessful, they will be prompted again

- They will then be prompted to enter ‘yes’ to submitting a recipe
    - They are prompted to enter a recipe title, recipe instructions, recipe description, recipe cuisine, the region that cuisine belongs to, and a time estimate for the recipe in minutes
        - This will add the recipe to the recipe table as a ‘Submitted’ one to be approved
    - They are prompted to enter the utensils needed in a list separated by commas
        - This adds the utensils to the utensils table
    - They are prompted to enter the ingredients needed in a list separated by commas
    - They are prompted to enter the amount for each ingredient in a list separated by commas
        - This will add to the ingredients table
- Or ‘no’ to submitting a recipe
- They will then be prompted to to answer ‘yes’ to searching for a recipe
    - They will answer ‘yes’ to searching for a recipe by its name
        - They will be asked to to enter a recipe name
            - The recipe with that name will be returned
            - Or they will be given an error message that the name doesn’t exist and are asked to enter another one
    - Or ‘no’ to searching a recipe by its name
    - They will answer ‘yes’ to searching a recipe by a keyword
        - They will be prompted to enter a keyword
            - Any recipe names with that keyword will be returned
    - Or ‘no’ to searching a recipe by a keyword
    - They will answer ‘yes’ to searching a recipe by cuisine
        - They will be prompted to enter a cuisine
            - If the cuisine name doesn’t exist in the cuisine table an error message will appear and they will be prompted again
            - Any recipes with that cuisine will be printed out
    - Or ‘no’ to searching a recipe by cuisine
    - They will answer ‘yes’ to searching a recipe by the time estimate in minutes
        - They will then enter a time in minutes
            - Recipes with time estimate within a range of +10 or -10 minutes will be returned

            - Or ‘no’ to searching a recipe by the time estimate in minutes
            - Or ‘no’ to searching for a recipe
        - They will then answer ‘yes’ to selecting a recipe from the search(s)
            - They will be prompted to give the recipe id of the recipe they would like to select
                - The recipe, utensil, and ingredients will be printed for that recipe
        - Or ‘no’ to selecting a recipe from the search(s)
        - They will answer ‘yes’ if they want to know how many recipe submissions they have
            - Their submission count will be returned
        - Or ‘no’ if they don’t want to know how many recipe submissions they have
        - They will answer ‘yes’ if they want to continue searching and selecting recipes
            - They will be redirected to begin the search/select process again
        - Or they will answer ‘no’ if they don’t want to continue searching and selecting recipes
            - They will be logged out
    - Or if they’re a ‘moderator’
        - They will be prompted to answer ‘yes’ if they need to sign up
            - They will enter their profile name, email, username, and password
                - If that moderator doesn’t already exist, they will be added
        - They will answer ‘no’ if they don’t need to sign up & instead just log in
            - They will enter their moderator username and password to log in
            - If they don’t exist, an error message will appear & they will be prompted to log in again
        - They will type ‘view’ to review the submitted recipes that are waiting for approval
            - It will return each recipe with the ‘Submitted’ status one at a time
                - The moderator will type ‘approve’ to approve it, thus changing its status to ‘Posted’
                - Or they will type ‘deny’ to deny it, thus deleting it from the recipe table
                - If they type neither of these options, an error message will appear to prompt them again
        - They moderator will be logged out

6. Lessons Learned

We gained a lot of technical knowledge from completing this project. Both of us have not had any prior experience with databases or SQL so this project really strengthened our knowledge from this course about how to design and create databases. We also learned more about how to connect a database to python so we can create more user interaction and the nuances that come with that. We also developed new skills beyond our current knowledge of Python that will be beneficial for the future. As with most long term project assignments, time management can always become difficult. We just needed to keep in mind that it is important to split up the work over time and not procrastinate too much because otherwise this work cannot be done in a short time frame. We made time to work through how we wanted it to work, designed the uml and sql database, created procedures, and then worked on python and user interactions. We made sure to constantly communicate and check to see how our project was progressing. It was interesting to work with our recipe data domain. There are so many out there and as we were trying to find files / places that we could scrape to put data in our table, we learned new recipes that we can now practice making. We also realized how many functions a user can utilize to create a recipe database. Our initial design approach was pretty similar to how the final product works, however we definitely had a lot more ideas that we did not have time to implement and some that would have made it a bit too complex for our knowledge. For example, we considered adding functionality for a user to rate a recipe and also add a review to it that other users could then view. We do not have any code that is not working correctly.

7. Future Work

There is certainly a lot more functionality that could be added to our recipes database. Currently, we do not have plans to continue to work on this project. However, we do have ideas on how it can be expanded upon if we change our minds in the future. This database can be incredibly useful to users who are looking for an easy way to find recipes and share their own with others. We would definitely like to add greater functionality for users to rate and review recipes that other users can view when they are searching for and selecting recipes. It could also be useful to optimize the interface through which users interact with the database. As of now, we have terminal commands that users would respond to for these interactions. However, creating a GUI interface would definitely improve this function and allow users to more clearly see their options with how to use the recipes database.