Devyanshi Chandra and Rahil Badkul
Professor Durant
CS 3200
Final Report

READ ME

First, download the housingdb.sql and housingoperations.sql files to access the housing database and our data on your computer. Then, download the HousingApp_ChandraBadkul.py using Spyder. After going to the correct directory for the spyder file in the terminal, run the command 'python' and the name of the file in order to run the file.

Another way to load our Spyder Python file is to download Spyder via Anaconda. The link to downloading Anaconda is https://docs.anaconda.com/anaconda/install/index.html. Through this, you can launch Spyder and continue to access our code. In order to continue onto the housing application, you must enter your SQL username and password in order to connect to the database. Afterward, you can complete the steps below:

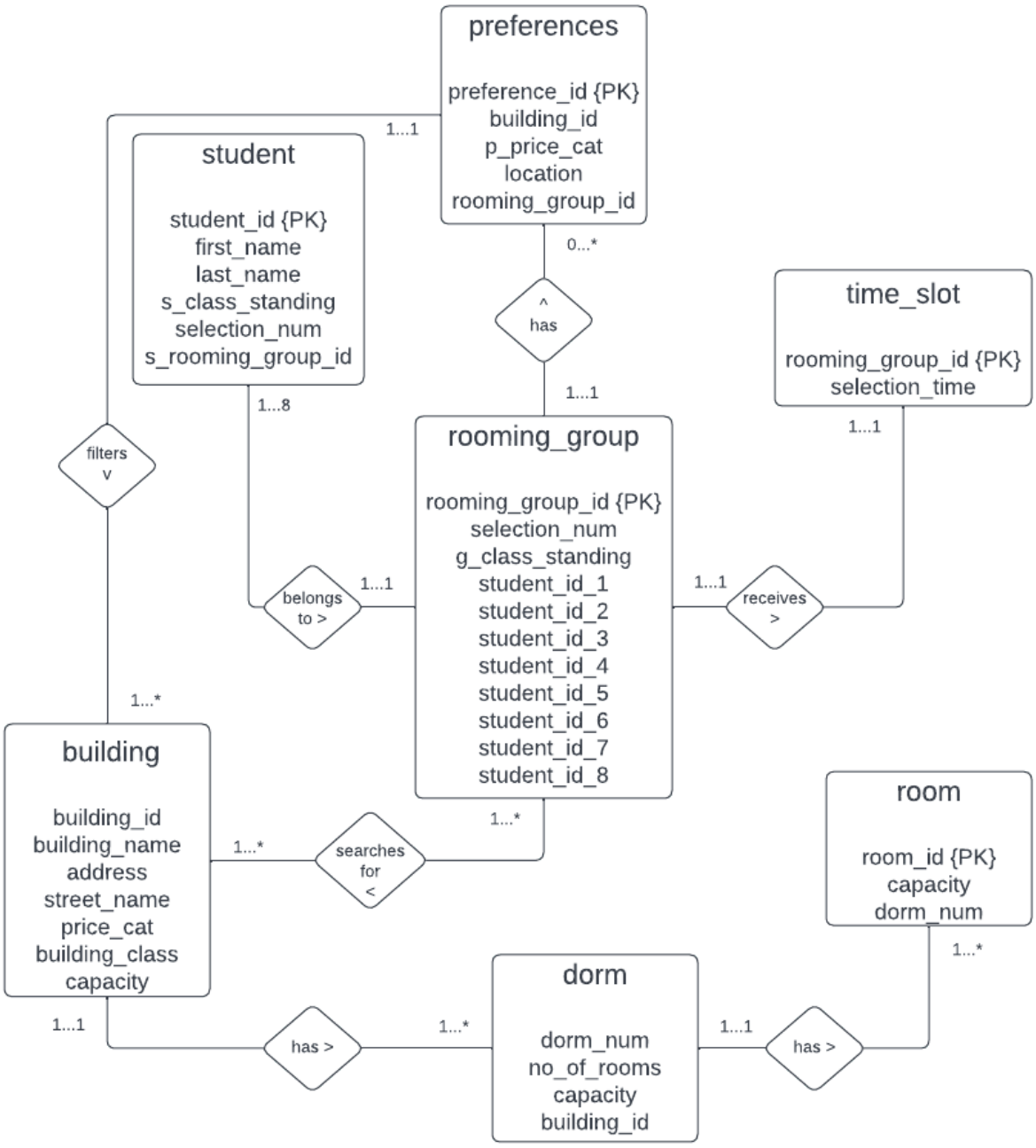
- After entering your username and password, the console will ask if the user currently has a group within our database.
    - If the user has a group, they will be asked if they'd like to search using their group information or whether they'd like to edit the group, then prompted to enter their group ID
        - If a user would like to edit their group, they'll be asked if they'd like to add or delete a member from their group. After choosing an action, they will provide the student ID they'd like to delete/add, and the action will be completed as well as the group being updated
    - If the user does not have a group, they will have to enter all students' IDs who wish to be included within this group. After receiving those IDs, they will then have to input each student's first and last names, selection numbers, and class standings
- Once the group has been finalized, the user will then be asked if they'd like to continue in order to search. If they choose to search, they'll be asked if they have a specific building they'd like to search
    - If they do, they'll be asked to provide the building's name and the console will display basic information about the building, as well as the dorms available that match or are larger than the capacity of their group
    - If they don't, the console will display all buildings that fit the class standing and capacity of the rooming group and ask if they'd like to filter out searches based on price, location, or both. Based on their choice, they'll be prompted to either pick a location/street name or a price category or both.

- Then the user has the option to either continue to display information about the dorms or rooms, or to stop where they are. Afterwards, they're asked if they'd like to add this building to their preferences.
    - If no, then they're asked if they'd like to continue searching
    - If yes, they can input their group id and the building id to add the preference into their preference table
- The user can then choose to continue searching or stop here. If they choose to stop, they have the choice of viewing their time slot, and if they choose yes, it's displayed in the console. Afterwards, the current search closes and all inserts to tables are saved

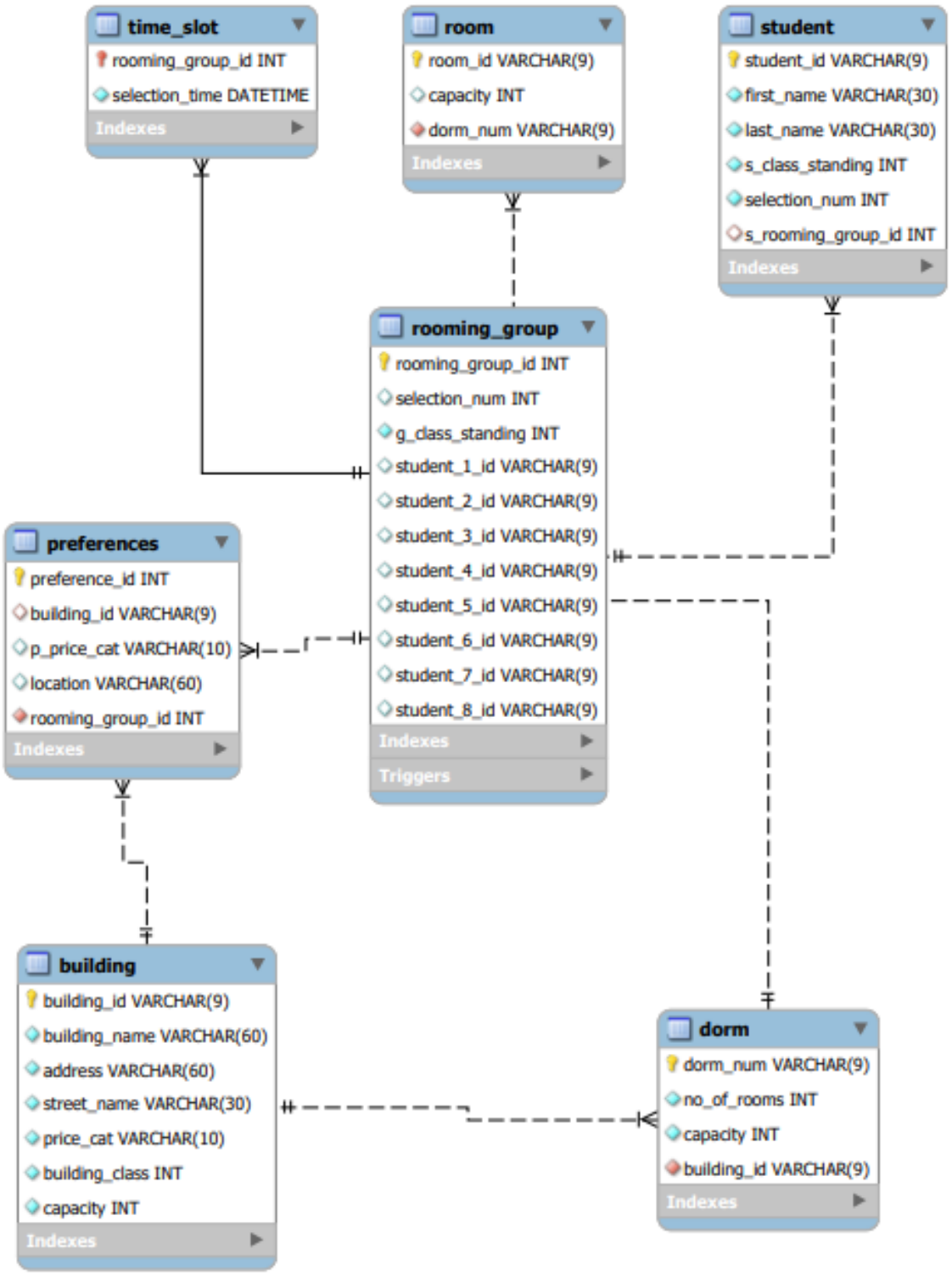
Technical Specifications

- ❖ MySQL Workbench (SQL)
- ❖ Spyder (Python)
- ❖ Excel

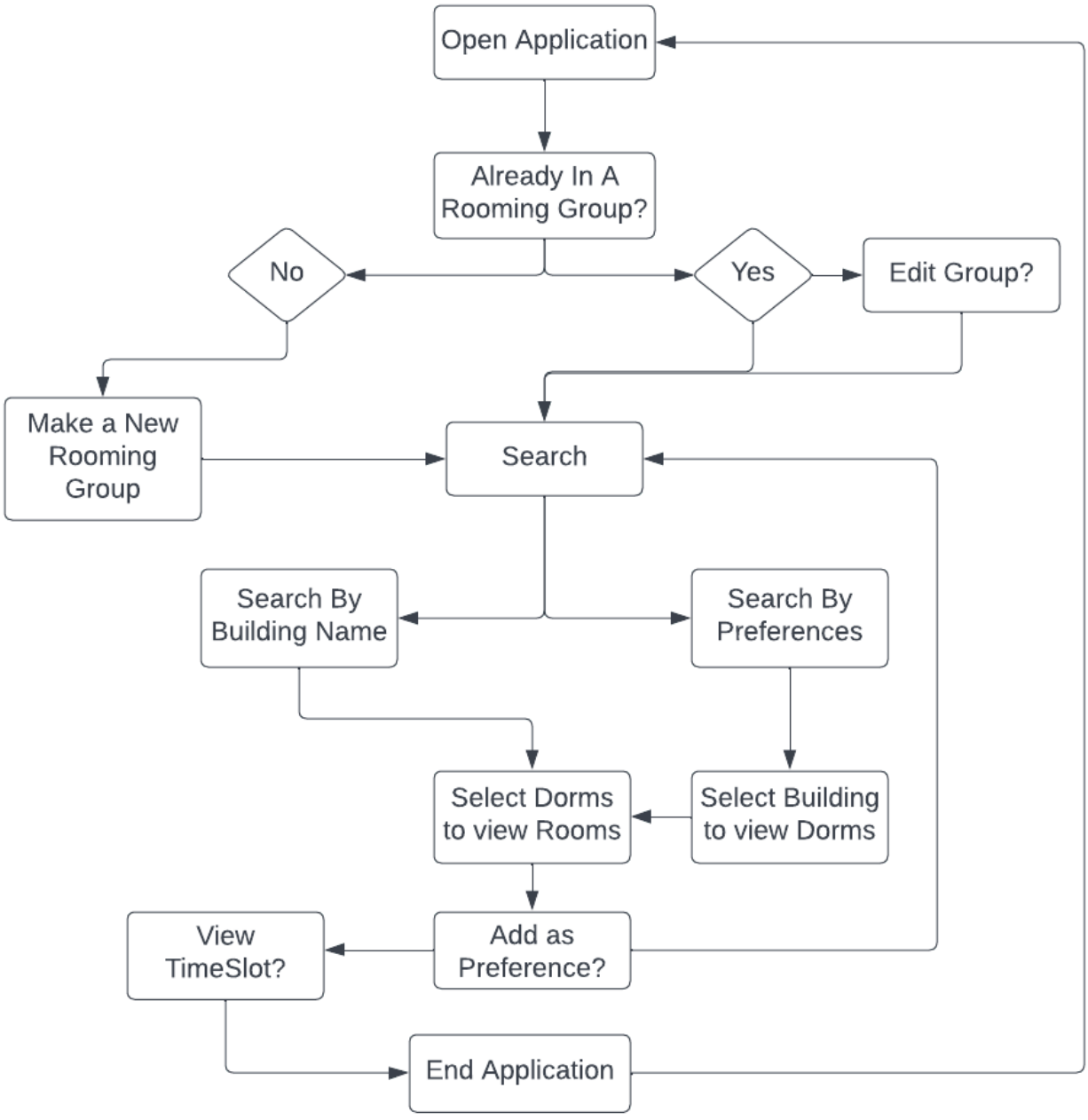
Conceptual Design (UML)



Logical Design (Reverse Engineer)



User FlowChart



Lessons Learned

This project helped us become adept at using multiple programming languages and technologies simultaneously to create a cohesive application. We now understand the integration between database and programming softwares, as well as the importance of properly communicating and documenting our progress. Now that we have finished the application, we realize that we could have extended our design to include more preferences for choosing housing such as wheelchair accessibility and air conditioning units. Another way to extend the application is to offer off campus apartments as well for housing options and to filter buildings by concrete prices rather than price categories. Student reviews of buildings would also aid the selection process by offering insights on living in the dorms.

Future Work

This application would be helpful in order to make the process of selecting and finding housing more seamless. As it stands, however, it would require more functionality in order to be truly useful. If a user would be able to add their most liked buildings into a list and then be able to access this list during their selection time, the housing selection process could be made more helpful and intuitive. This application would also help as a mandatory activity that first year students need to do before their selection time. It would expose them to the potential places they could live at and make a daunting task less stressful.