CS 331 Francois Mukaba

Project Part A + B Jena Lovejoy

11/28/18 Simone Ray

**REPORT**

Design:

All available housing units with descriptions are displayed when an applicant chooses the “Apply” option in the main menu. The applicant is prompted to enter 3 preferences + a roommate preference, and then sequentially enters username, password, and other personal information. Once everything is inputted, the applicant will get the match automatically and it will be printed on the screen with the new assigned address. They can return to the main screen and login as a resident. If no unit is available, they are added to to the waitlist with a dummy housing unit. At the time that a unit becomes available, Admissions admin will have to change the values and correctly assign the waitlisted applicant a housing unit (We did not implement this part of the project).

Assumptions made in EER:

1. Residents and Waitlisted applicants are two types of users in our housing system. They cannot be both types simultaneously (hence are represented as disjoint), and there are no other types of users (hence complete relationship). A user can also be the family head of other users (hence self-relational).
2. There are three types of admin: maintenance, residence, and admissions admin. We make the assumption that there are no other types of admin in our model (hence complete relationship). Admin can hold multiple roles though (hence overlapping). They can handle many maintenance requests, many residents, and many waitlisted applications respectively.
3. An applicant can have many housing preferences, hence it is a multivalued attribute.
4. Many residents can live in AT MOST one housing unit. The housing unit has an attribute type, which is either apartment or suite. The attributes, # of bedrooms and max people, indicate which type of apartment or suite it is. Another attribute called “occupied/not occupied” allows the housing system to know which units are available to lease. A 1 indicates that the housing unit is fully occupied, a 0 indicates that there is still some vacancies in the unit.
5. A maintenance request can be filed by AT MOST one person.
6. A maintenance request, application, resident can be handled by EXACTLY one admin.
7. After submitting an Application and getting approved by the College, the students has to confirm their approval. If there are no available units, or the applicant does not confirm their housing unit, they are added to the Waitlist table.

Design choices in Relational schema:

1. Multivalued attributes, housing\_preference has its own table since it is multivalued.
2. Since admin types are overlapping, maintenance admin, resident admin, and admissions admin are represented with boolean flags in the admin table.

Additional assumptions:

1. All data is stored and kept for a period of time greater than five years, so it can be accessed later for demographic purposes.
2. Roommate preferences ONLY matched if roommate already exists in the system.
3. We are only going to book people based on preferences.
4. In table “Maintenace\_request”, the status column only accepts value “pending” or “completed”. All requests entered should have value “pending” and as soon as a request is dealt with, the administrator will change the status value from “pending” to “completed”.

CS 331 Francois Mukaba

Project Part A + B Jena Lovejoy

11/28/18 Simone Ray

**WORK BREAKDOWN**

Francois Mukaba

* Finished and formatted the EER Diagram
* Added initial data to tables
* Wrote MaintenanceRequestDue.java and added method to confirm resident login credentials in HousingSystem.java
* Wrote query for Task 2
* Helped with debugging
* Helped with the bulk uploads
* Edited Create table queries
* Presented in video

Jena Lovejoy

* Wrote create table statements for part A
* Started and formatted the EER diagram
* Added initial data to tables
* Wrote HousingClient.java, including printing menu interfaces, allowing user interaction with those interfaces, accepting and validating data, connecting to HousingSystem.java
* Added comments and headers to all files
* Debugged HousingClient & HousingSystem
* Presented in video
* Pulled together/edited video

Simone Ray

* Helped design EER
* Designed the relational schema
* Wrote the project report
* Wrote load\_tables
* Wrote bulk of HousingSystem.java and HousingUnit.java (in charge of Task 1)
  + checking availability, booking housing, creating resident, creating applicant
* Added initial data to tables
* Helped with debugging HousingClient, HousingSystem
* Presented in video