

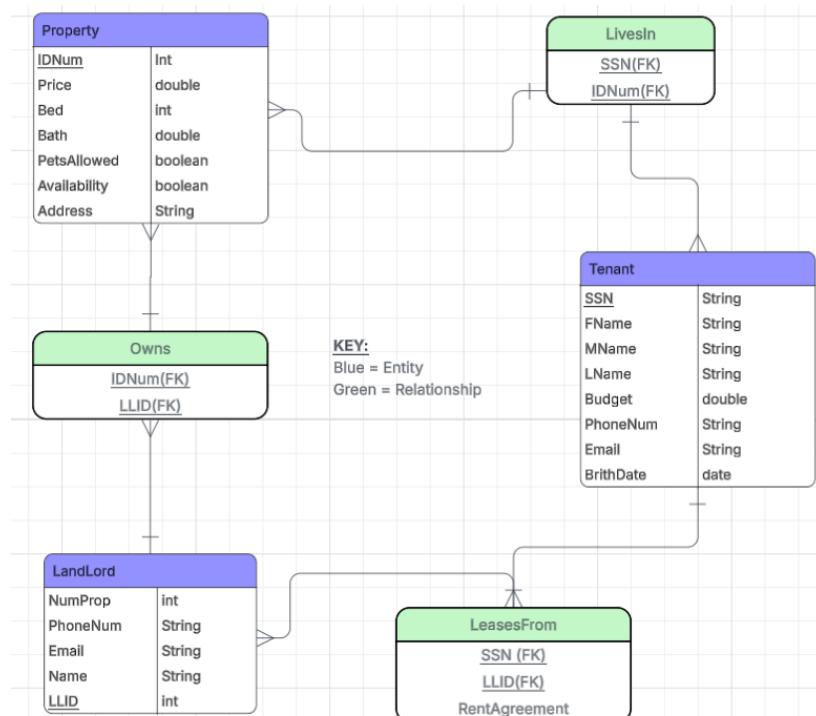
Project Proposal

Problem Statement

Trying to find an apartment that meets your unique needs in the ever-changing market is not only a difficult task, but also nearly impossible for those who do not regularly scan the renting market for the best options that meet their needs and wants. Our goal for this project is twofold: make finding your perfect apartment easy and ensure that you have all the relevant details available to you at the quick and easy push of a button. No more searching through long and endless apartment postings to find out if they will allow you to live with your cat or dog.

Our intended users will be all people who wish to make the finding and renting of an apartment quick, easy and even a little fun. Using a database of apartments in the area, we will be able to easily narrow down and find the perfect apartment for the user based on their own unique specifications.

Database Design



This is the Entity

Relationship (ER) diagram of the database that we intend to use. The database will have three entities: Property, Tenant, and LandLord. As well as three relationships: LivesIn, Owns, and LeasesFrom. Detailed descriptions of the entities and relationships can be found below.

CS 364 Project Proposal

Andrew Pierce, Jacob Rogers, Rohan Hari

The Property table, that describes a rental property, will hold the attributes: IDNum, Price, Bed, Bath, PetsAllowed, Availability, and Address. The Tenant table, which describes the potential tenants who are renting or looking to rent properties, will hold the attributes: SSN, FName, MName, LName, Budget, PhoneNum, Email, BirthDate. The final table, the Landlord table, will hold the attributes: NumProp, PhoneNum, Email, Name, LLID (landlord ID).

The Property table will connect to the Tenant table through the LivesIn relationship. The Tenant table will connect to LandLord though the LeasesFrom relationship and finally the LandLord table will connect to the Property table through the Owns relationship.

Functionality

Our database will provide functionality in two main categories: CRUD operations and advanced queries.

CRUD operations:

- Users looking to rent will be able to enter their budget so they can find the perfect apartment
- Landlords will be able to create new listings
- Renters will be able to view existing listings, with details displayed clearly for comparison
- Both renters and property managers will be able to update existing information
- Property managers will be able to delete listings that are no longer available

Advanced queries:

- Renters will be able to search for apartments within a specific budget range
- Renters will be able to filter searches by key features like number of bedrooms, number of bathrooms, or amenities (parking, laundry, pet-friendly options)
- Renters will be able to search by location

Technical Requirements

The languages we will use for this project are going to be Java and SQL with the JDBC connector. We will primarily use our own personal laptops as the machines for this project. As for the specific tools of the project, we will be using Java Swing to create the GUI, which makes this a desktop application. Another tool we will be using is Jsoup, which is a web scraping tool that gathers data about each apartment. We plan to use MySQL on the class database and use GitHub to share our code. We all have experience with Java, and some of us used Swing in our CS120 classes.