University of British Columbia, Vancouver
Department of Computer Science

CPSC 304 Project Cover Page

Milestone #: 1

Date: July 14, 2024

Group Number: 42

Name	Student Number	CS Alias (Userid)	Preferred Email Address
Dominic Leo	66545906	t6b8l	dpaleo22@gmail.com
Robert Burns	27053263	j4w1e	rdburns96@gmail.com
Shane Mander	38316287	s1u2u	shane.mander31@gmail.com

By typing our names and student numbers in the above table, we certify that the work in the attached assignment was performed solely by those whose names and student IDs are included above. (In the case of Project Milestone 0, the main purpose of this page is for you to let us know your email address, and then let us assign you to a TA for your project supervisor.)

In addition, we indicate that we are fully aware of the rules and consequences of plagiarism, as set forth by the Department of Computer Science and the University of British Columbia

University of British Columbia, Vancouver
Department of Computer Science

GROUP 42 – Real Estate Management System

Project Description

The domain of our proposed application is real estate. We intend for it to serve as a single system that will allow agents to store and view the company's database of listed properties, both past and present. To achieve this, our system must be able to capture the relationships between clients, buyers, lawyers, etc.. The goal of our proposed application is to efficiently encapsulate these relationships and property information and allow for its rapid retrieval. We believe this would be of great benefit to a real estate company, helping them properly manage their listed properties.

Our proposed database aims to model the myriad of ways a party can be tied to a property while also providing information about the property itself. The former includes, but is not limited to, owners, tenants, and prospective buyers as well as the agents and lawyers that may represent them. The latter is even more expansive with some common examples being houses and apartments. An example use case for our proposed database is a prospective buyer would approach an agent of the company who can query our system to find out which listed properties fit their criteria, who owns these properties, and if the owners are represented by lawyers, which ones. The previous example highlights why we believe this system would be useful to real estate companies.

Database Specifications

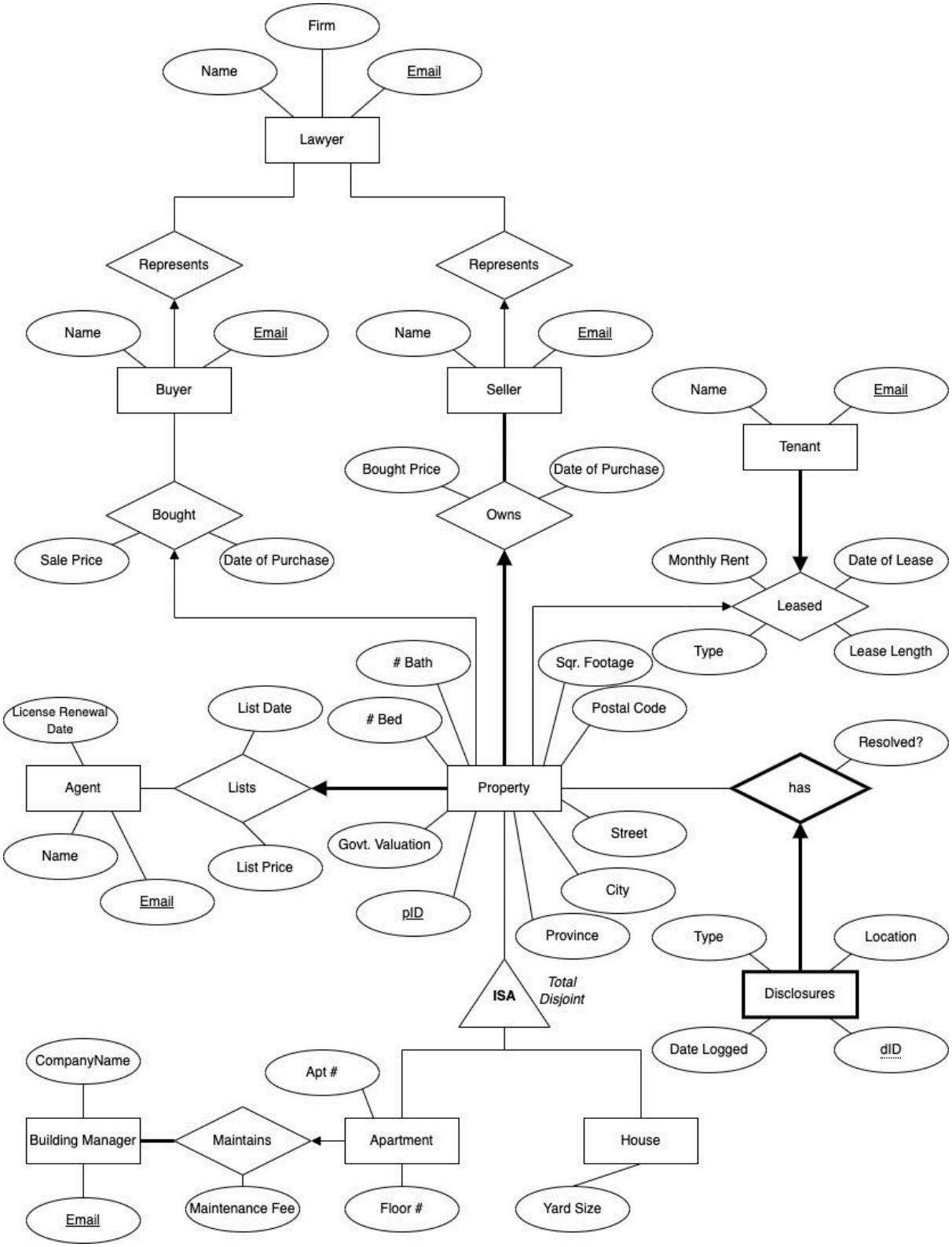
This database will allow employees of our target real estate company to store and query all property listings the company has been involved with (past and present). It will enable our agent to view the property details, which agent listed the property, the contact details of the seller or the seller's lawyer, the building manager's contact details (if the property is an apartment), and any disclosures that were made regarding the property (property damage, etc.). Additionally, agents will be able to view details of past listings including sale price, date at which sale was made, and the buyer's contact info (or their lawyer). Finally, if the property is under lease, the agent will be able to access the tenant's contact info.

Application Platform

The DBMS our project will use will be the department-provided Oracle server. We expect to use PHP to write the backend (connecting to SQL) and HTML/CSS for the front end.

University of British Columbia, Vancouver
Department of Computer Science

ER Diagram



University of British Columbia, Vancouver
Department of Computer Science

Comments

- Email address was chosen as the primary key for Seller, Buyer, Agent, Building Manager, and Tenant. Other candidate keys included a combination of name and address. However email was chosen as the primary key as not only is it guaranteed to be unique for each seller/buyer/tenant, but email is the primary form of contact for our real estate company, thus making it the optimal candidate key.
- The Building Manager entity refers to the company that built the apartment complex (and not the person who manages the complex), as houses typically do not require external maintenance fees while apartments do, this is why Building Manager has no relationship to House.
- The primary key for Disclosures is dID (Disclosure ID). An artificial primary key was used for this entity as the other attributes for Disclosures may not always be unique. DateLogged, Type, and Description are not guaranteed to be unique for each Disclosure entity, therefore, an artificial primary key must be created for this entity.
- The attribute of Property 'Govt.Valuation' refers to the value at which the provincial government assesses the property at. For example, in British Columbia, there exists the BC Assessment. This is helpful for our agents, as it will enable them to more easily gauge the value of a listed property.
- In the ER diagram, Buyer and Seller are two separate entities. Our group discussed combining both entities into an ISA relationship. However, upon discussion with another TA, they explained the approach above is simpler and easier to implement.
- As per TA advice, our group decided to strip unnecessary attributes from our ER diagram including but not limited to: Address (for Seller, Buyer, Agent, and Building Manager), Phone Number (for Seller, Buyer, Agent, Lawyer, and Tenant), additional attributes for Property (e.g. Year built in, property name, property tax, etc.) etc..
- Disclosures_type refers to an in-house code assigned to each disclosure to describe its nature (e.g. #001 for water damage, #002 for mold etc.).
- The primary key for Property is pID. An artificial primary key was used for this entity as using the address of a property alone does not guarantee uniqueness. This is because different apartment numbers in the same apartment block will have the same address. Our team considered adding Apt # as a primary key to Property, but this would not work as a primary key cannot be null (and House sub-entities would have a null Apt # value).