University of British Columbia, Vancouver
Department of Computer Science

CPSC 304 Project Cover Page

Milestone #: 2

Date: October 15th, 2022

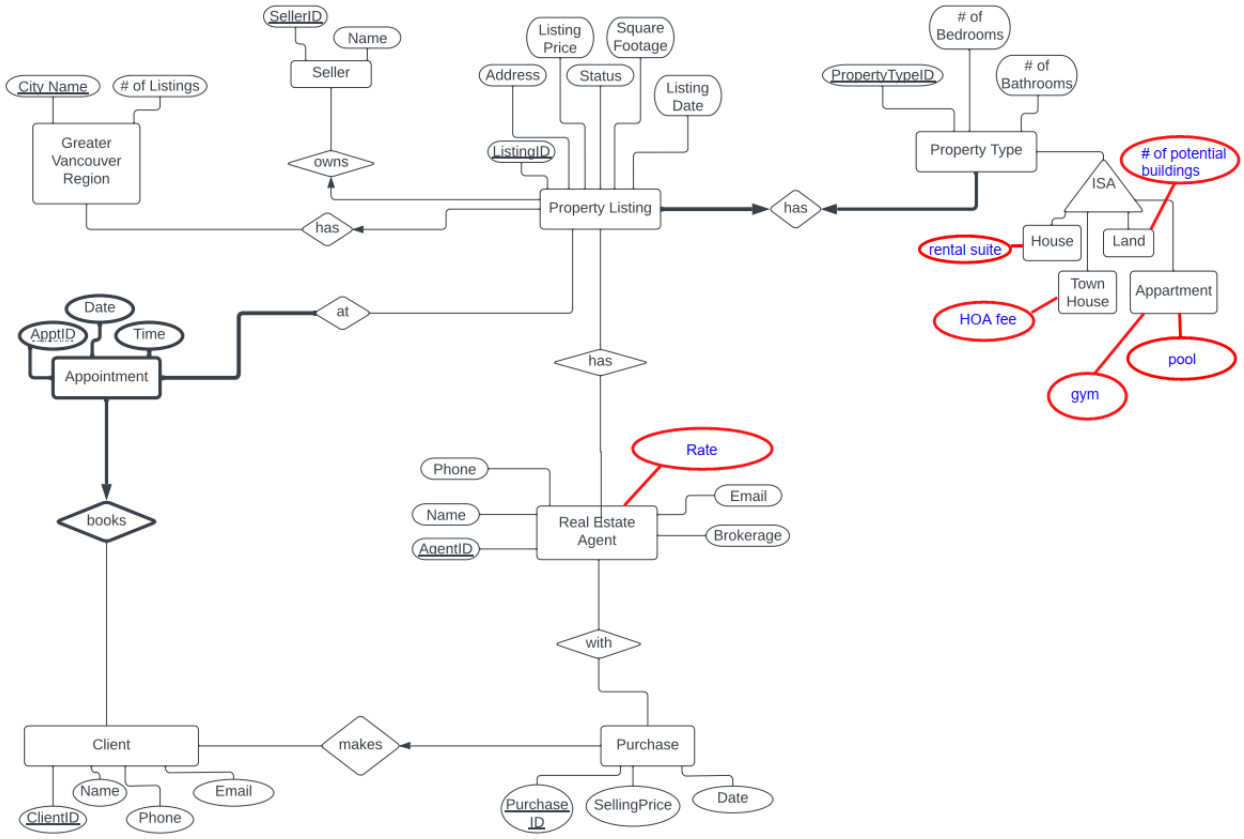
Group Number: 10

Name	Student Number	CS Alias (Userid)	Preferred E-mail Address
Sarah Li	21849476	j9z2b	Sarahli2001@gmail.com
Raifah Rahman	29435245	b2r8v	raifahbooks19@gmail.com
Krisha Iquin	67576298	p0d0a	krishaiquin@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 e-mail 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

2. The ER diagram you are basing your item #3 (below) on.



<u>Additions from previous ER Diagram - added unique attributes to ISA subclasses:</u>

- Added rental suite attribute to House entity (will be a boolean value denoting whether there is a rental suite or not)
- Added gym and pool attribute to Apartment entity (will be a boolean value denoting whether there is a gym or not), also added HOA(Home Owner's Association) fee attribute to Apartment entity
- Added HOA fee attribute to TownHouse entity
- Added # of potential buildings attribute to Land entity (Land has potential to build a house, or apartment or townhouse)
- Added Rate attribute to Real Estate Agent entity (real estate agents works for a specific rate)

3. The schema derived from your ER diagram (above).

* CKs are italicized, PKs are underlined, FKs are bolded

- GreaterVancouverRegion(<u>City Name</u>, # of Listings)
- PropertyListing(<u>ListingID,</u> **City Name**, **SellerID**, **PropertyTypeID**, **AgentID**, Address, Listing Price, Status, Square Footage, Listing Date)
- PropertyListingHasPropertyType(**<u>ListingID</u>**, **<u>PropertyTypeID</u>**)

*can't enforce many-to-many participation constraints yet here, need assertions

- PropertyListingHasRealEstateAgent(**AgentID**, **ListingID**)
- Seller(SellerID, Name)
- PropertyType(PropertyTypeID, **ListingID**, # of Bedrooms, # of Bathrooms)
- House(**PropertyTypeID**, Rental Suite)
- Apartment(**PropertyTypeID**, Pool, Gym)
- Land(**PropertyTypeID**, # of Potential Buildings)
- TownHouse(**PropertyTypeID**, HOA fee)
- Client(ClientID, Name, *Phone*, *Email*)
- Purchase(PurchaseID, **ClientID**, Selling Price, Date)
- PurchaseWithRealEstateAgent(**AgentID, PurchaseID)**
- ClientBooksAppointment(ApptID, **ClientID,** Date, Time)
- AppointmentAtListing(**ApptID, ListingID**)
- RealEstateAgent(AgentID, Name, *Phone*, *Email*, Rate, Brokerage)

4. Functional Dependencies (FDs)

* needs to be normalized

- GreaterVancouverRegion(City Name, # of Listings)
  - City Name -> # of Listings
- PropertyListing(ListingID, **City Name**, **SellerID**, **PropertyTypeID**, Address, Listing Price, Status, Square Footage, Listing Date)*
  - ListingID -> City Name, SellerID, PropertyTypeID, Address, Listing Price, Status, Square Footage, Listing Date
  - Address -> CityName, Status(if you know the address of the listing, you know which city it's in and whether its been sold or not)
- PropertyListingHasPropertyType(**AgentID**, **ListingID**)
  - AgentID -> ListingID
  - ListingID -> AgentID
- Seller(SellerID, Name)
  - SellerID -> Name
- PropertyType(PropertyTypeID, **ListingID**, # of Bedrooms, # of Bathrooms)
  - PropertyTypeID -> ListingID, # ofBedrooms, # of Bathrooms
- House(**PropertyTypeID**, Rental Suite)
  - PropertyTypeID -> Rental Suite
- Apartment(**PropertyTypeID**, Pool, Gym)
  - PropertyTypeID -> Pool, Gym
- Land(**PropertyTypeID**, # of Potential Buildings)
  - PropertyTypeID -> # of potential buildings
- TownHouse(**PropertyTypeID**, HOA fee)
  - PropertyTypeID -> HOA fee
- Client(ClientID, Name, *Phone*, *Email*)
  - ClientID -> Name, Phone, Email

- Email -> ClientID

- Purchase(PurchaseID, **ClientID**, Selling Price, Date)
  - PurchaseID -> ClientID, SellingPrice, Date
- PurchaseWithRealEstateAgent(**AgentID, PurchaseID)**
  - AgentID -> PurchaseID
  - PurchaseID -> AgentID
- Appointment(ApptID, **ClientID,** Date, Time)
  - ApptID, ClientID -> Date, Time
- AppointmentAtListing(**ApptID, ListingID**)
  - ApptID -> ListingID
  - ListingID -> ApptID
- RealEstateAgent(AgentID, Name, *Phone*, *Email*, Rate, Brokerage)*
  - AgentID -> Name, Phone, Brokerage, Email, Rate
  - Email -> AgentID
  - Rate, Brokerage -> Name (if you know the agent's rate and the brokerage they work for, you know the name of the agent)

5. Normalization
*we normalized to BCNF

1) PropertyListing(ListingID, **City Name**, **SellerID**, **PropertyTypeID**, Address, Listing Price, Status, Square Footage, Listing Date)

- ListingID -> City Name, SellerID, PropertyTypeID, Address, Listing Price, Status, Square Footage, Listing Date
- Address -> CityName, Status

ListingID+ = {ListingID, City Name, SellerID, PropertyTypeID, Address, Listing Price, Status, Square Footage, Listing Date }

Address+ = {Address, CityName, Status}

ListingID is the only key

Address -> CityName, Status violates BCNF, so we decompose on PropertyListing
PropertyListing_1(Address, CityName, Status)
PropertyListing_2(ListingID, SellerID, PropertyTypeID, Address, Listing Price, Square Footage, Listing Date)

Final Answer: PropertyListing_1(Address, **CityName**, Status), PropertyListing_2(ListingID, **SellerID**, **PropertyTypeID**, Address, Listing Price, Square Footage, Listing Date)

---

2) RealEstateAgent(AgentID, Name, *Phone*, *Email*, Rate, Brokerage)

- AgentID -> Name, Phone, Brokerage, Email, Rate
- Email -> AgentID
- Rate, Brokerage -> Name

AgentID+ = {AgentID, Name, Phone, Email, Brokerage, Rate}
Email+ = {Email, AgentID, Name, Phone, Brokerage, Rate}
Rate, Brokerage+ = {Rate, Brokerage, Name}

AgentID and Email are the only keys

Rate, Brokerage -> Name violates BCNF, so we decompose on Purchase
RealEstateAgent_1(Rate, Brokerage, Name)
RealEstateAgent_2(AgentID, Phone, Email, Rate, Brokerage)

Final Answer: RealEstateAgent_1(Rate, Brokerage, Name),
RealEstateAgent_2(AgentID, *Phone*, *Email*, Rate, Brokerage)

6. The SQL DDL statements required to create all the tables from item #5.

PropertyListing_1(Address, **CityName**, Status)

```
CREATE TABLE PropertyListing_1 (
        Address             CHAR(30) NOT NULL,
        CityName            CHAR(30) NOT NULL,
        Status              CHAR(10) NOT NULL,
        FOREIGN KEY (CityName) REFERENCES GreaterVancouverRegion
                ON DELETE CASCADE)
```

PropertyListing_2(ListingID, **SellerID**, **PropertyTypeID**, Address, Listing Price, Square Footage, Listing Date)

```
CREATE TABLE PropertyListing_2 (
        ListingID           INTEGER PRIMARY KEY,
        SellerID            INTEGER NOT NULL,
        PropertyTypeID      INTEGER NOT NULL,
        Address             CHAR(30) NOT NULL,
        ListingPrice        INTEGER,
        SquareFootage       INTEGER,
        ListingDate         DATE,
        FOREIGN KEY (SellerID) REFERENCES Seller
                ON DELETE CASCADE,
        FOREIGN KEY (PropertyTypeID) REFERENCES PropertyType
                ON DELETE CASCADE)
```

RealEstateAgent_1(Rate, Brokerage, Name)

```
CREATE TABLE RealEstateAgent_1 (
        Rate            INTEGER,
        Brokerage       CHAR(30),
        Name            CHAR(30))
```

RealEstateAgent_2(AgentID, *Phone*, *Email*, Rate, Brokerage)

```
CREATE TABLE RealEstateAgent_2 (
        AgentID         INTEGER PRIMARY KEY,
        Phone           CHAR(30) UNIQUE,
        Email           CHAR(30) UNIQUE,
        Rate            INTEGER,
        Brokerage       CHAR(30))
```

7. INSERT statements to populate each table with at least 5 tuples

1) PropertyListing_1(Address, **CityName**, Status)

```
INSERT
INTO    PropertyListing_1(Address, CityName, Status)
VALUE   ('1885 Student Union Boulevard', 'Vancouver','sold')

INSERT
INTO    PropertyListing_1(Address, CityName, Status)
VALUE   ('14211 84th Avenue', 'Surrey', 'on market')

INSERT
INTO    PropertyListing_1(Address, CityName, Status)
VALUE   ('7960 Bennett Rd', 'Richmond', 'sold')

INSERT
INTO    PropertyListing_1(Address, CityName, Status)
VALUE   ('5553 Eglinton St', 'Burnaby', 'sold')

INSERT
INTO    PropertyListing_1(Address, CityName, Status)
VALUE   ('2712 Woodland Drive', 'Coquitlam', 'on market')
```

2) PropertyListing_2(<u>ListingID</u>, **SellerID**, **PropertyTypeID**, Address, Listing Price, Square Footage, Listing Date)

INSERT
INTO PropertyListing_2(ListingID, SellerID, PropertyTypeID, Address, Listing Price, Square Footage, Listing Date)
VALUE (1, 11, 1, '1885 Student Union Boulevard', 1100, 100, '2022-08-19')

INSERT
INTO PropertyListing_2(ListingID, SellerID, PropertyTypeID, Address, Listing Price, Square Footage, Listing Date)
VALUE (1, 12, 2, '14211 84th Avenue', 1 600 000, 200, '2022-03-17)

INSERT
INTO PropertyListing_2(ListingID, SellerID, PropertyTypeID, Address, Listing Price, Square Footage, Listing Date)
VALUE (2, 13, 3, '7960 Bennett Rd', 500 000, 300, '2021-11-14')

INSERT
INTO PropertyListing_2(ListingID, SellerID, PropertyTypeID, Address, Listing Price, Square Footage, Listing Date)
VALUE (4, 14, 4, '5553 Eglinton St', 2000, 'sold', '2020-05-09')

INSERT
INTO PropertyListing_2(ListingID, SellerID, PropertyTypeID, Address, Listing Price, Square Footage, Listing Date)
VALUE (5, 15, 1, '2712 Woodland Drive', 1 300 000, 500, '2022-10-15')

3) RealEstateAgent_1(Rate, Brokerage, Name)

INSERT
INTO RealEstateAgent_1(Rate, Brokerage, Name)
VALUE (30, 'Oakwyn Realty', 'Amy Arnold')

INSERT
INTO RealEstateAgent_1(Rate, Brokerage, Name)
VALUE (50, 'RE/MAX', 'Bob Barney')

INSERT
INTO RealEstateAgent_1(Rate, Brokerage, Name)
VALUE (10, 'Macdonald Realty', 'Carl Cooper')

INSERT
INTO RealEstateAgent_1(Rate, Brokerage, Name)
VALUE (15, 'Century 21', 'Dani Devito')

INSERT
INTO RealEstateAgent_1(Rate, Brokerage, Name)
VALUE (25, 'Keller Williams', 'Erin Evans')

4) RealEstateAgent_2(AgentID, *Phone*, *Email*, Rate, Brokerage)

INSERT
INTO RealEstateAgent_2(AgentID, *Phone*, *Email*, Rate, Brokerage)
VALUE (1, '604-123-4567', 'amy@gmail.com', 30, 'Oakwyn Realty')

INSERT
INTO RealEstateAgent_2(AgentID, *Phone*, *Email*, Rate, Brokerage)
VALUE (2, '778-123-4567', 'bob@gmail.com', 50, 'RE/MAX')

INSERT
INTO RealEstateAgent_2(AgentID, *Phone*, *Email*, Rate, Brokerage)
VALUE (3, '236-123-4567', 'carl@gmail.com' 10, 'Macdonald Realty')

INSERT
INTO RealEstateAgent_2(AgentID, *Phone*, *Email*, Rate, Brokerage)
VALUE (4, '604-111-0000', 'dani@gmail.com', 15, 'Century 21')

INSERT
INTO RealEstateAgent_2(AgentID, *Phone*, *Email*, Rate, Brokerage)
VALUE (5, '778-000-1111', 'erin@gmail.com', 25, 'Keller Williams')