

# CPSC 304 Project Cover Page

Milestone #: 4

Date: 05/08/2024

Group Number: 11

Name	Student Number	CS Alias (Userid)	Preferred E-mail Address
Abdul Khurram	25159641	f0p7l	arkb25@student.ubc.ca
Rehnoor Saini	78995446	g0o9s	rehnoorsaini@outlook.com
Udbhav Kansal	42392381	m8z9e	udbhav27@student.ubc.ca

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

## Short Description of the Final Project and What It Accomplished

Our final project is a comprehensive real estate management system that allows users to manage property listings, make offers, and schedule appointments with realtors. This web-based platform provides a user-friendly interface for buyers and sellers to interact seamlessly with realtors and each other. Key accomplishments of the project include:

- **User Registration and Authentication:** Users can securely register and log in to the system, ensuring personalized experiences and secure data handling.
- **Property Listings Management:** Users can browse, search, and filter property listings by various criteria, such as price, type, and location. Real Estate insights help users make a decision on buying homes.
- **Offers and Appointments:** The system allows users to make offers on properties and schedule appointments with realtors for property viewings.
- **Realtor Management:** Realtors can manage their listings, view offers, and keep track of appointments efficiently through their dashboard.
- **Database Integration:** All operations are backed by a robust database schema, ensuring data integrity and consistency across the platform.

Overall, the project successfully integrates front-end interfaces with a back-end database, achieving the goal of providing a comprehensive and functional real estate management system.

## Description of How the Final Schema Differed from the Schema Turned In

### **1. Consolidation and Simplification:**

- **Properties and Subtypes:**
  - The final schema consolidates property attributes into a central Properties table, with subtypes like Houses, Townhouses, Condos, and Apartments inheriting from it. This reduces redundancy and simplifies the overall structure.
  - The original schema might have had more fragmented or redundant tables for each property type without a central structure.

### **2. User Role Differentiation:**

- **Users Table:**
  - The addition of a UserType attribute in the Users table allows differentiation between buyers and sellers directly within the table.

- This contrasts with the original schema, which might have separated buyers and sellers into distinct tables or lacked clear role differentiation.

**3. Keys and Constraints:**

- **Primary and Foreign Keys:**

- The final schema uses composite primary keys for tables like Properties and Listings (based on Address and PostalCode), ensuring uniqueness and data integrity.
- Foreign key relationships are robust, with ON DELETE CASCADE used extensively to maintain referential integrity across deletions, which might have been less comprehensive in the original schema.

**4. Enhanced Data Integrity:**

- **Unique Constraints:**

- Unique constraints were added on attributes such as Email and Phone in the Realtors and Users tables to prevent duplicate entries and ensure data consistency.
- The original schema might have lacked these constraints, leading to potential data duplication.

**5. Additional Attributes:**

- **Appointments and Listings:**

- New attributes like AppointmentStatus and ListingStatus provide clear status tracking for appointments and listings, which might have been less detailed in the original schema.

**6. Schema Optimization:**

- **Normalization:**

- The schema was further normalized, reducing redundancy and ensuring that each piece of data is stored only once.
- This contrasts with a potentially more denormalized original schema that might have had duplicate information across tables.

**7. Relationships and Integrations:**

- **Amenity Associations:**

- The Amenities table links directly to properties using Address and PostalCode, providing a structured approach to associating facilities with specific properties.

- The original schema might have handled amenities less directly or with a less robust linkage.

#### 8. Weak Entity Relationship:

- **Listings as a Weak Entity:**

- The Listings table is now modeled as a weak entity dependent on the Properties table. It uses the Address and PostalCode as part of its composite primary key, signifying its dependency on a Property for existence.
- This modeling approach ensures that a listing cannot exist without an associated property, which strengthens data integrity and reflects the real-world dependency of listings on properties.
- This is a shift from the original schema, where listings may not have had such clearly defined dependency.

#### Updated Schema and Instances

- Realtors(**RealtorID**, Name, Email, Phone, YearsOfExperience, RealEstateFirm)
- Users(Name, **Email**, Phone, UserType, **RealtorID**)
- Properties(**Address**, **PostalCode**, City, Province, PropertyType, PropertyCondition, NumBaths, NumBeds, YearBuilt, InteriorSpace)
- Houses(**Address**, **PostalCode**, YardSize, NumGarage, NumFloors, HasBasement)
- Townhouses(**Address**, **PostalCode**, NumGarage, NumFloors, HOAFee)
- Condos(**Address**, **PostalCode**, HOAFee, UnitNumber)
- Apartments(**Address**, **PostalCode**, UnitNumber)
- Amenities(**Name**, Type, Distance, Rating, **PropertyAddress**, **PropertyPostalCode**)
- Listings(**ListingID**, **Address**, **PostalCode**, ListingStatus, **SellerEmail**, ListingPrice, ExpirationDate, ListingDate)
- Offers(**OfferID**, OfferStatus, OfferDate, OfferAmount, **BuyerEmail**, **ListingID**, **Address**, **PostalCode**)
- Appointments(**AppointmentID**, AppointmentStatus, AppointmentDate, AppointmentTime, **BuyerEmail**, MeetingPlace, **ListingID**, **Address**, **PostalCode**)

#### Realtors

University of British Columbia, Vancouver  
Department of Computer Science

RealtorID	Name	Email	Phone	YearsOfExperience	RealEstateFirm
1	Emily Brown	emilybrown@gmail.com	3456789012	5	Dream Homes Realty
2	Michael Johnson	michaeljohnson@gmail.com	4567890123	8	Premium Estates
3	Sarah Green	sarahgreen@gmail.com	5678901234	4	Top Choice Realty
4	David Wilson	davidwilson@gmail.com	6789012345	10	Elite Realty Group
5	Jessica Taylor	jessicataylor@gmail.com	7890123456	7	Star Real Estate
6	Daniel White	danielwhite@gmail.com	8901234567	6	Prime Properties

## Users

Name	Email	Phone	UserType	RealtorID
Rehnoor	rehnoorsaini@outlook.com	4316885656	buyer	1
Sam Jane	samjane@hotmail.com	1234567890	seller	1
George Washington	georgewash@laundry.com	2234567990	buyer	2
Will Smith	willsmith@alaadin.com	1237867893	seller	3
Sundar Pichai	sundarpichai@gmail.com	2354557899	buyer	4
Brett Lee	brettlee@rediffmail.com	5154447890	seller	5
Steve Smith	stevesmith@google.com	2345678901	buyer	6

## Properties

Address	City	Province	PropertyType	PostalCode	PropertyCondition	NumBaths	NumBeds	YearBuilt	InteriorSpace
123 Maple St	Vancouver	BC	House	V5K0A1	Good	2	3	1995	2000
456 Oak St	Toronto	ON	Condo	M5H2N2	New	1	1	2015	800
789 Pine St	Calgary	AB	Townhouse	T2P1B3	Fair	2	2	2005	1500
321 Birch St	Montreal	QC	House	H3Z2Y7	Fair	3	4	1980	2500
654 Cedar St	Edmonton	AB	Apartment	T5J1B7	New	1	2	2020	900
987 Elm St	Ottawa	ON	House	K1P5G1	Good	3	5	1975	2300
159 Spruce St	Winnipeg	MB	Townhouse	R3C4G6	Fair	2	3	1990	1600
753 Willow St	Halifax	NS	Apartment	B3J3M8	New	1	1	2022	700

## Houses

Address	PostalCode	YardSize	NumGarage	NumFloors	HasBasement
123 Maple St	V5K0A1	1000	2	2	Yes
321 Birch St	H3Z2Y7	800	1	2	Yes
987 Elm St	K1P5G1	200	1	2	No

## Townhouses

Address	PostalCode	NumGarage	NumFloors	HOAFee
789 Pine St	T2P1B3	1	3	100
159 Spruce St	R3C4G6	0	2	150

## Condos

Address	PostalCode	HOAFee	UnitNumber
456 Oak St	M5H2N2	500	4

Apartments

Address	PostalCode	UnitNumber
753 Willow St	B3J3M8	502
654 Cedar St	T5J1B7	601

Amenities

Name	Type	Distance	Rating	PropertyPostalCode	PropertyAddress
Central Park	Park	0.5	4.5	B3J3M8	753 Willow St
Vancouver Library	Library	1	4.8	V5K0A1	123 Maple St
Community Pool	Pool	0.8	4.2	T2P1B3	789 Pine St
Fitness Center	Gym	1.5	4.7	R3C4G6	159 Spruce St
Shopping Mall	Mall	2	4.3	B3J3M8	753 Willow St
City Theater	Theater	1.2	4.6	T2P1B3	789 Pine St

Listings

ListingID	Address	PostalCode	ListingStatus	SellerEmail	ListingPrice	ExpirationDate	ListingDate
1	123 Maple St	V5K0A1	Active	rehnoorsaini@outlook.com	550000	2024-12-31	2024-01-01
2	456 Oak St	M5H2N2	Pending	samjane@hotmail.com	350000	2024-11-30	2024-02-15
3	789 Pine St	T2P1B3	Active	georgewash@laundry.com	450000	2024-10-15	2024-03-10
4	321 Birch St	H3Z2Y7	Active	willsmith@alaadin.com	500000	2024-09-30	2024-04-20
5	654 Cedar St	T5J1B7	Pending	sundarpichai@gmail.com	400000	2024-08-30	2024-05-25
6	987 Elm St	K1P5G1	Active	brettlee@rediffmail.com	600000	2024-07-31	2024-06-15

Offers

OfferID	OfferStatus	OfferDate	OfferExpiryDate	OfferAmount	BuyerEmail	ListingID	Address	PostalCode
1	Pending	2024-03-01	2024-03-15	510000	samjane@hotmail.com	1	123 Maple St	V5K0A1
2	Accepted	2024-04-01	2024-04-10	310000	georgewash@laundry.com	2	456 Oak St	M5H2N2
3	Pending	2024-05-01	2024-05-15	460000	willsmith@alaadin.com	3	789 Pine St	T2P1B3
4	Rejected	2024-06-01	2024-06-15	360000	sundarpichai@gmail.com	4	321 Birch St	H3Z2Y7
5	Pending	2024-07-01	2024-07-10	560000	brettlee@rediffmail.com	5	654 Cedar St	T5J1B7
6	Accepted	2024-08-01	2024-08-10	410000	stevesmith@google.com	6	987 Elm St	K1P5G1

Appointments

AppointmentID	AppointmentStatus	AppointmentDate	AppointmentTime	MeetingPlace	BuyerEmail	ListingID	Address	PostalCode
1	Scheduled	2024-03-05	10:00	123 Elm St	samjane@hotmail.com	1	123 Maple St	V5K0A1
2	Completed	2024-04-05	14:00	456 Oak St	georgewash@laundry.com	2	456 Oak St	M5H2N2
3	Scheduled	2024-05-10	11:00	789 Pine St	willsmith@alaadin.com	3	789 Pine St	T2P1B3
4	Cancelled	2024-06-10	15:00	321 Maple St	sundarpichai@gmail.com	4	321 Birch St	H3Z2Y7
5	Scheduled	2024-07-15	09:00	654 Birch St	brettlee@rediffmail.com	5	654 Cedar St	T5J1B7
6	Completed	2024-08-20	13:00	987 Cedar St	stevesmith@google.com	6	987 Elm St	K1P5G1

## SQL Queries

### **Fetch Listings for Seller**

**File:** appService.js

**Lines:** 61-70

**SQL Query:**

```
SELECT ListingID, l.Address, City, Province, ListingPrice, ListingStatus, NumBaths, NumBeds,
l.PostalCode
FROM PROPERTIES p, LISTINGS l
WHERE SellerEmail = :sellerEmail AND
    p.Address = l.Address AND
    p.PostalCode = l.PostalCode AND
    ListingPrice >= :minPrice AND
    ListingPrice <= :maxPrice AND
    NumBaths >= :minBath AND
    NumBeds >= :minBed AND
    InteriorSpace >= :minSpace AND
    InteriorSpace <= :maxSpace AND
    YearBuilt >= :minYear
```

### **Fetch Listings for Non-Seller**

**File:** appService.js

**Lines:** 81-89

**SQL Query:**

```
SELECT ListingID, l.Address, City, Province, ListingPrice, ListingStatus, NumBaths, NumBeds,
l.PostalCode
FROM PROPERTIES p, LISTINGS l
WHERE p.Address = l.Address AND
    p.PostalCode = l.PostalCode AND
    ListingPrice >= :minPrice AND
    ListingPrice <= :maxPrice AND
    NumBaths >= :minBath AND
    NumBeds >= :minBed AND
    InteriorSpace >= :minSpace AND
    InteriorSpace <= :maxSpace AND
    YearBuilt >= :minYear
```

### **Get Maximum Offer ID**

**File:** appService.js



**Lines:** 97

**SQL Query:**

```
SELECT NVL(MAX(OfferID), 0) + 1 AS NextOfferID FROM OFFERS
```

**Insert Offer**

**File:** appService.js

**Lines:** 102-106

**SQL Query:**

```
INSERT INTO OFFERS (  
    OfferID, OfferStatus, OfferDate, OfferAmount, BuyerEmail, OfferExpiryDate, ListingID,  
    Address, PostalCode  
) VALUES (  
    :offerID, 'Pending', :offerDate, :offerAmount, :buyerEmail, :offerExpiryDate, :listingID,  
    :address, :postalCode  
)
```

**Fetch Offers for Listing**

**File:** appService.js

**Lines:** 125-128

**SQL Query:**

```
SELECT OfferID, OfferStatus, OfferDate, OfferAmount, BuyerEmail, OfferExpiryDate, ListingID,  
    Address, PostalCode  
FROM OFFERS  
WHERE ListingID = :IID
```

**Update Offer Status**

**File:** appService.js

**Lines:** 134-137

**SQL Query:**

```
UPDATE OFFERS  
SET OfferStatus = :status  
WHERE OfferID = :offerID
```

**Fetch Property Details**

**File:** appService.js

**Lines:** 142-144

**SQL Query:**

## University of British Columbia, Vancouver

### Department of Computer Science

---

```
SELECT *  
FROM PROPERTIES  
WHERE ADDRESS = :addr AND POSTALCODE = :pc
```

#### Fetch Listing Details

**File:** appService.js

**Lines:** 145-148

**SQL Query:**

```
SELECT ListingID, ListingStatus, SellerEmail, ListingPrice  
FROM LISTINGS  
WHERE Address = :addr AND PostalCode = :pc AND ListingID = :listingId
```

#### Get Realtor Info

**File:** appService.js

**Lines:** 155-159

**SQL Query:**

```
SELECT *  
FROM REALTORS r  
WHERE r.RealtorID IN  
    (SELECT RealtorID  
     FROM USERS  
     WHERE Email = :sEmail)
```

#### Insert House

**File:** appService.js

**Lines:** 236-238

**SQL Query:**

```
INSERT INTO HOUSES(Address, PostalCode, YardSize, NumGarage, NumFloors, HasBasement)  
VALUES  
(:addr, :pCode, :ySpace, :hGarage, :hFloors, :basement)
```

#### Get Maximum Listing ID

**File:** appService.js

**Lines:** 246

**SQL Query:**

```
SELECT NVL(MAX(ListingID), 0) + 1 AS NextListingID FROM LISTINGS
```

#### Insert Listing

## University of British Columbia, Vancouver

### Department of Computer Science

---

**File:** appService.js

**Lines:** 251-253

**SQL Query:**

```
INSERT INTO LISTINGS (ListingID, Address, PostalCode, ListingStatus, SellerEmail, ListingPrice,
ExpirationDate, ListingDate)
VALUES
(:nextListingID, :addr, :pCode, 'Active', :email, :price, :expDate, :todayDate)
```

**Authenticate User**

**File:** appService.js

**Lines:** 274

**SQL Query:**

```
SELECT * FROM USERS WHERE EMAIL = :email
```

**Check User Existence**

**File:** appService.js

**Lines:** 292

**SQL Query:**

```
SELECT Email FROM USERS WHERE Email = :email
```

**Check Realtor Existence**

**File:** appService.js

**Lines:** 296

**SQL Query:**

```
SELECT RealtorID FROM REALTORS WHERE RealtorID = :realtorID
```

**Register User**

**File:** appService.js

**Lines:** 303-305

**SQL Query:**

```
INSERT INTO USERS (Name, Email, Phone, UserType, RealtorID)
VALUES (:name, :email, :phone, :userType, :realtorID)
```

**Fetch Amenities**

**File:** appService.js

**Lines:** 319-321

## University of British Columbia, Vancouver

### Department of Computer Science

---

#### SQL Query:

```
SELECT *  
FROM AMENITIES  
WHERE PropertyPostalCode = :pc AND PropertyAddress = :addr
```

#### Delete Listing

**File:** appService.js

**Lines:** 331-333

#### SQL Query:

```
DELETE FROM PROPERTIES WHERE Address = :address AND  
PostalCode = :postalCode
```

#### Update Listing Price

**File:** appService.js

**Lines:** 347

#### SQL Query:

```
UPDATE Listings SET ListingPrice=:price where ListingID=:lid AND Address=:addr AND  
PostalCode=:pCode
```

#### Update Property Details

**File:** appService.js

**Lines:** 351-353

#### SQL Query:

```
UPDATE Properties SET NumBaths=:nBaths, NumBeds=:nBeds, YearBuilt=:yBuilt,  
InteriorSpace=:space  
WHERE Address=:addr AND PostalCode=:pCode
```

#### Update House Details

**File:** appService.js

**Lines:** 360-362

#### SQL Query:

```
UPDATE HOUSES SET YardSize=:ySpace, NumGarage=:hGarage, NumFloors=:hFloors,  
HasBasement=:basement  
WHERE Address=:addr AND PostalCode=:pCode
```

#### Update Townhouse Details

**File:** appService.js

**Lines:** 366-368

## University of British Columbia, Vancouver

### Department of Computer Science

---

#### SQL Query:

```
UPDATE TOWNHOUSES SET NumGarage=:tGarage, NumFloors=:tFloors, HOAFee=:tFee
WHERE Address=:addr AND PostalCode=:pCode
```

#### Update Condo Details

**File:** appService.js

**Lines:** 372-374

#### SQL Query:

```
UPDATE CONDOS SET HOAFee=:cFee, UnitNumber=:cNum
WHERE Address=:addr AND PostalCode=:pCode
```

#### Update Apartment Details

**File:** appService.js

**Lines:** 378-380

#### SQL Query:

```
UPDATE APARTMENTS SET UnitNumber=:aNum
WHERE Address=:addr AND PostalCode=:pCode
```

#### Fetch Appointments for User

**File:** appService.js

**Lines:** 390-392

#### SQL Query:

```
SELECT *
FROM APPOINTMENTS
WHERE BuyerEmail = :email
```

#### Average Offer Amount Per Property Type

**File:** appService.js

**Lines:** 401-404

#### SQL Query:

```
SELECT p.PropertyType, AVG(o.OfferAmount) AS AverageOfferAmount
FROM OFFERS o
JOIN LISTINGS l ON o.ListingID = l.ListingID
JOIN PROPERTIES p ON l.Address = p.Address AND l.PostalCode = p.PostalCode
GROUP BY p.PropertyType
```

---

**Book Appointment**

**File:** appService.js

**Lines:** 417-419

**SQL Query:**

```
INSERT INTO APPOINTMENTS(AppointmentID, AppointmentDate, AppointmentTime,
MeetingPlace, BuyerEmail, ListingID, Address, PostalCode, AppointmentStatus)
VALUES
(:nextAppointmentID, :appDate, :appTime, :appMeetingPlace, :buyerEmail, :listingID, :address,
:postalCode, 'Scheduled')
```

**Update Appointment Status**

**File:** appService.js

**Lines:** 428-431

**SQL Query:**

```
UPDATE APPOINTMENTS
SET AppointmentStatus = :status
WHERE AppointmentID = :appointmentID
```

**Count Offers Per Listing**

**File:** appService.js

**Lines:** 441-444

**SQL Query:**

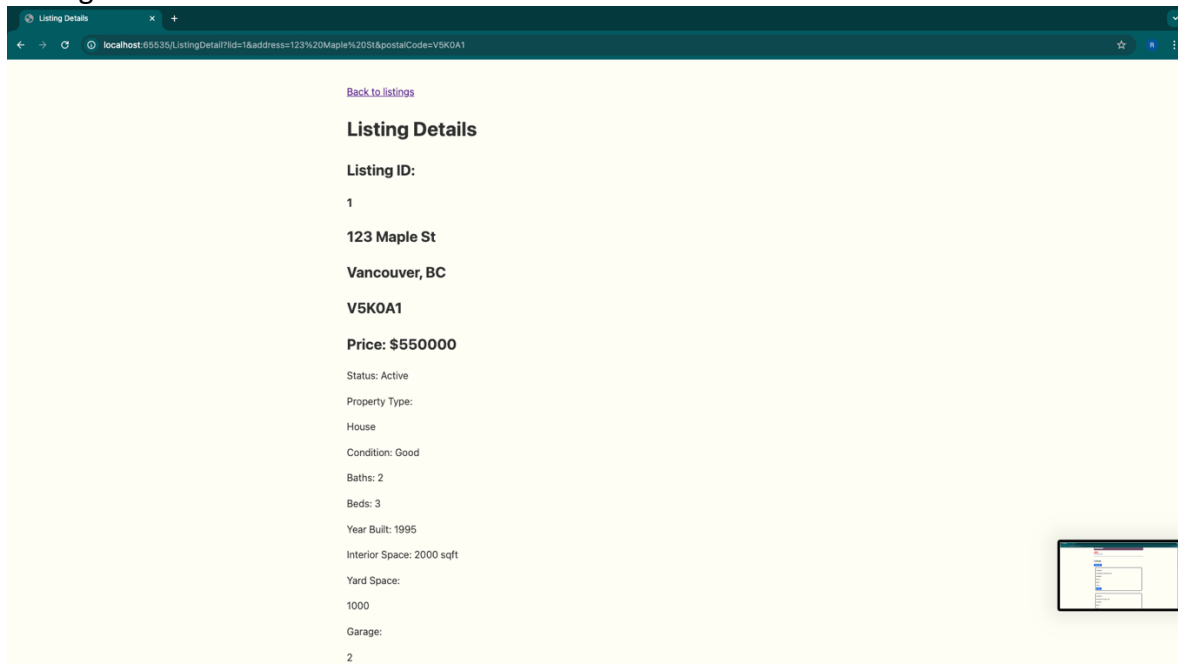
```
SELECT ListingID, COUNT(*) AS OfferCount
FROM OFFERS
GROUP BY ListingID
```

## Screenshots:

### Aggregation with Groupby

Before

During



After

#### Offer Counts

Number of Offers for this Listing: 1

Create Offer

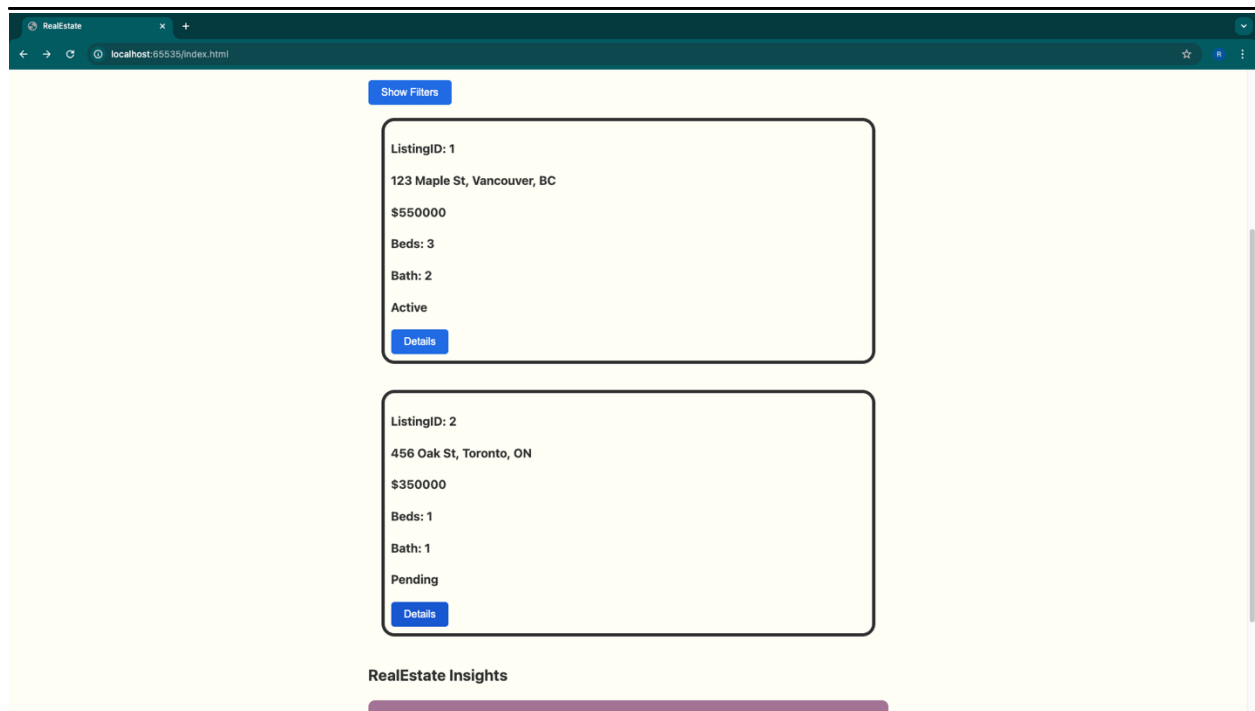
Book Appointment

### DELETE

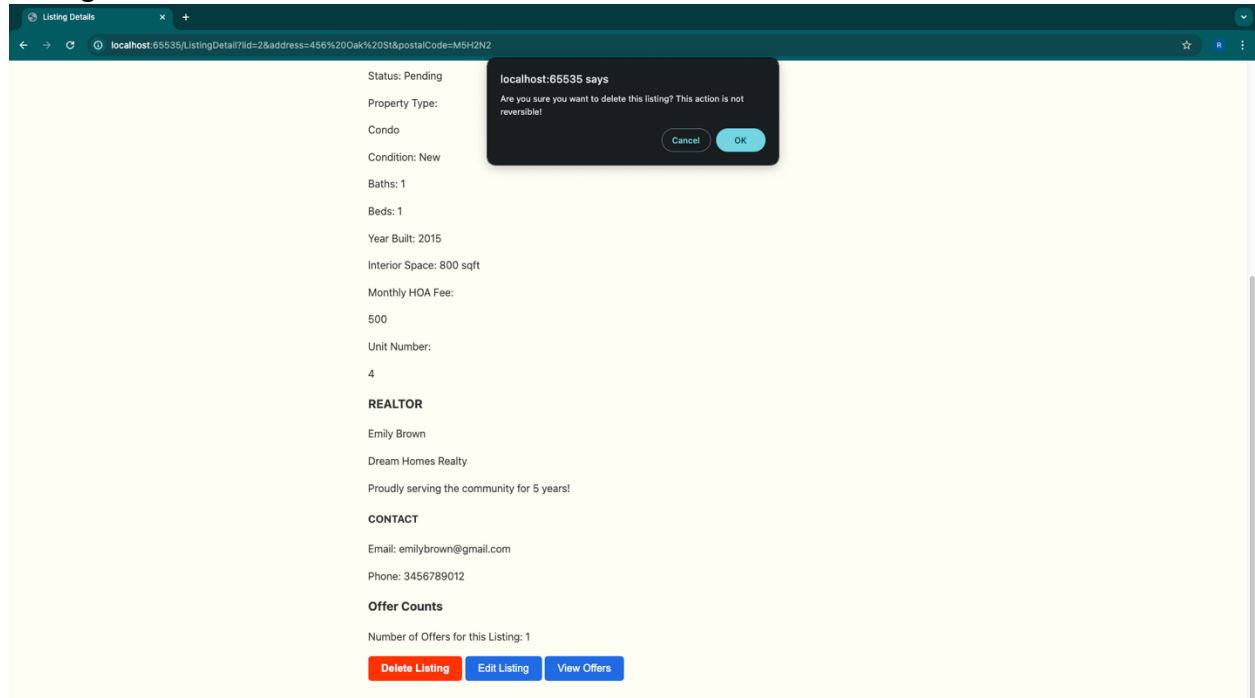
Before

# University of British Columbia, Vancouver

## Department of Computer Science



During

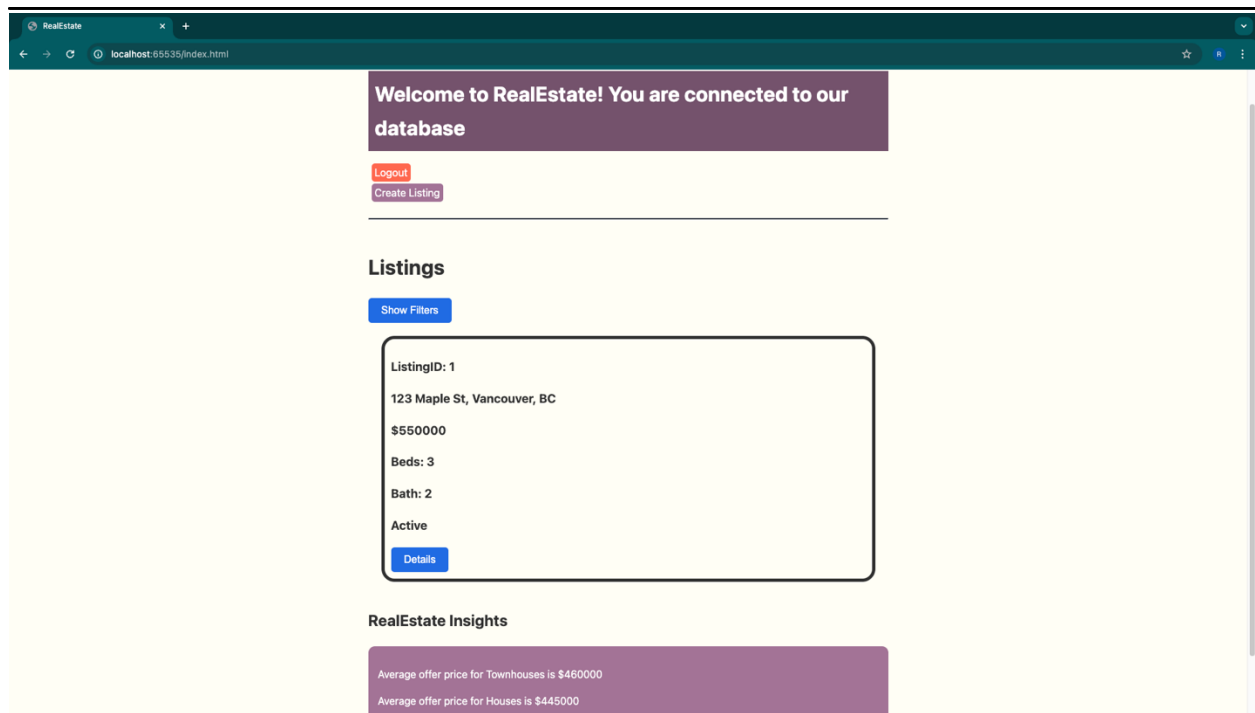


After



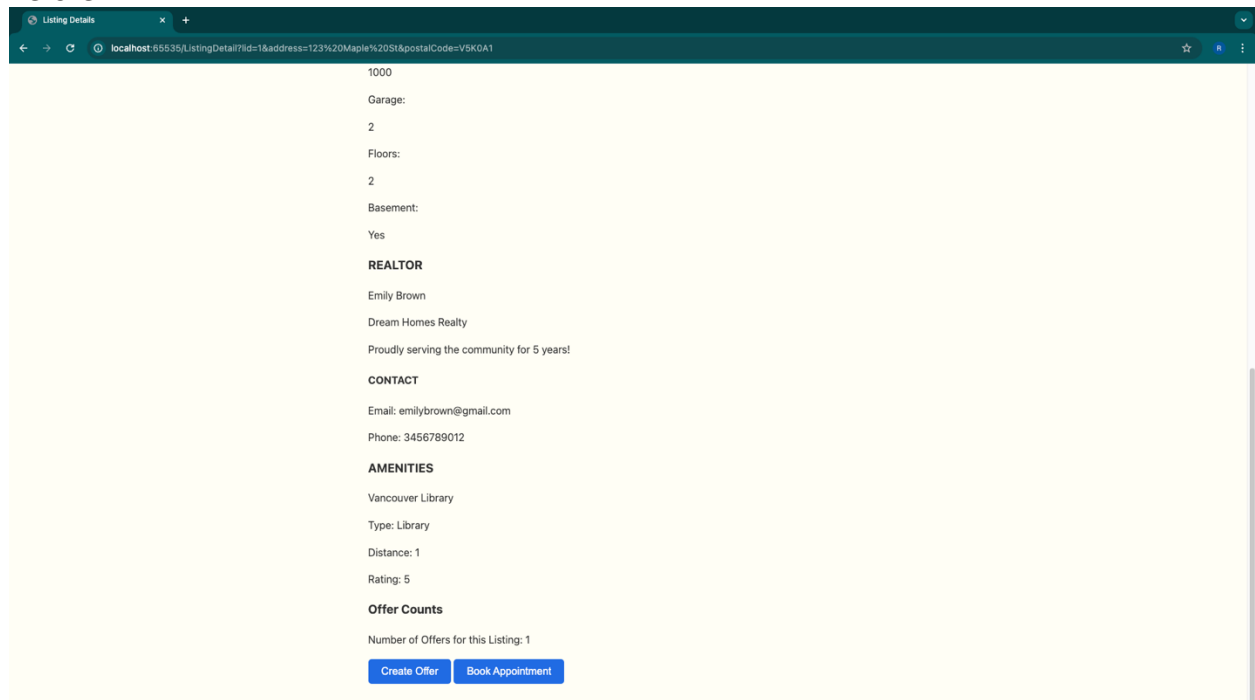
# University of British Columbia, Vancouver

## Department of Computer Science



## INSERT

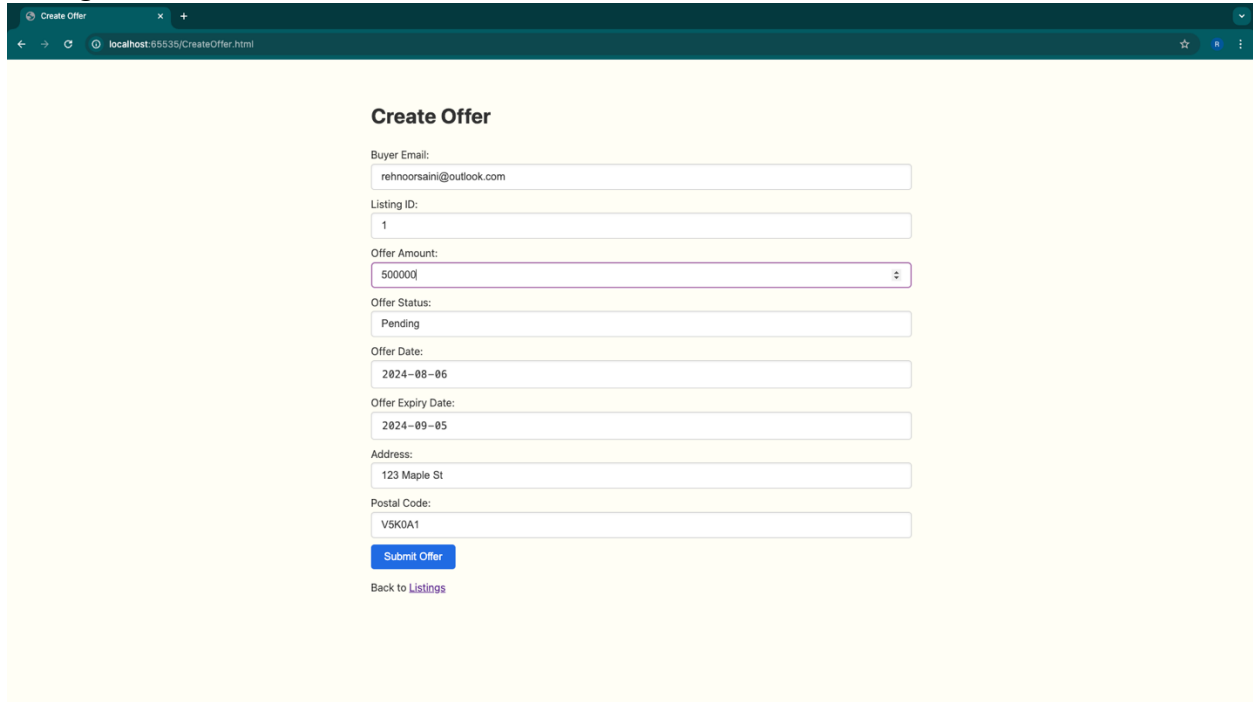
## Before



# University of British Columbia, Vancouver

## Department of Computer Science

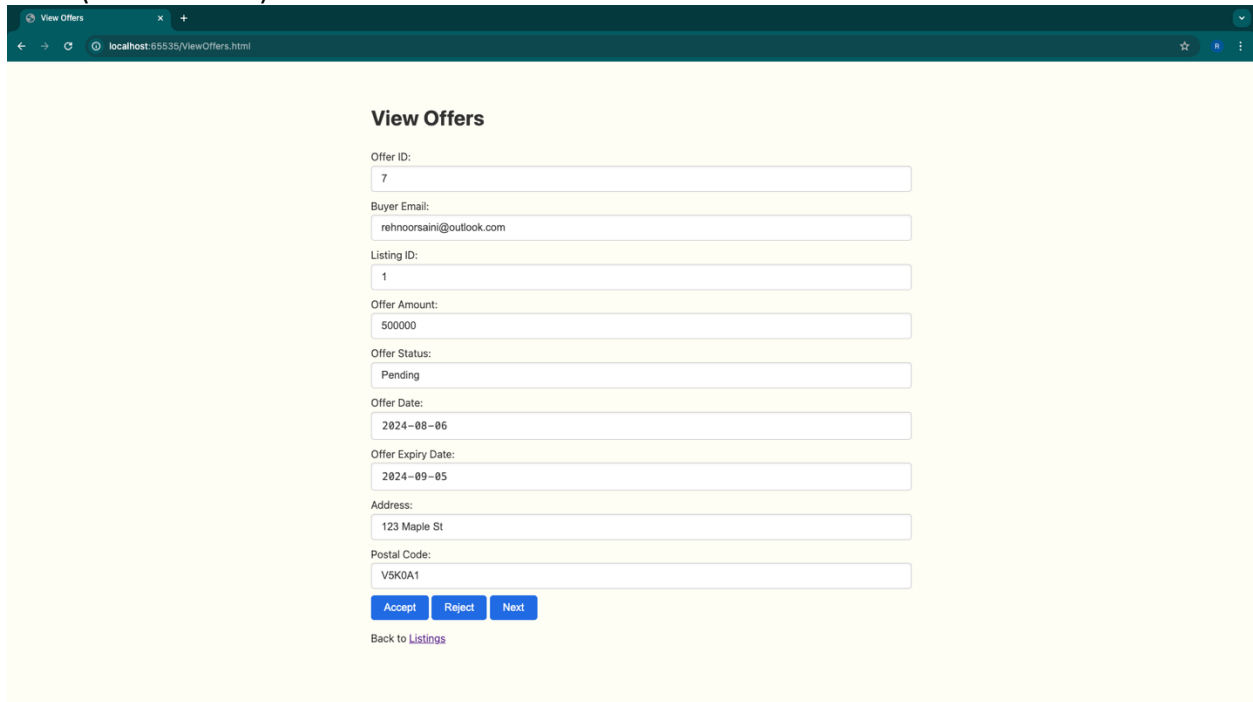
### During



The screenshot shows a web browser window with the address bar displaying 'localhost:65535/CreateOffer.html'. The page title is 'Create Offer'. The form contains the following fields and controls:

- Buyer Email:
- Listing ID:
- Offer Amount:
- Offer Status:
- Offer Date:
- Offer Expiry Date:
- Address:
- Postal Code:
- 
- [Back to Listings](#)

### After (in seller view)



The screenshot shows a web browser window with the address bar displaying 'localhost:65535/ViewOffers.html'. The page title is 'View Offers'. The form contains the following fields and controls:

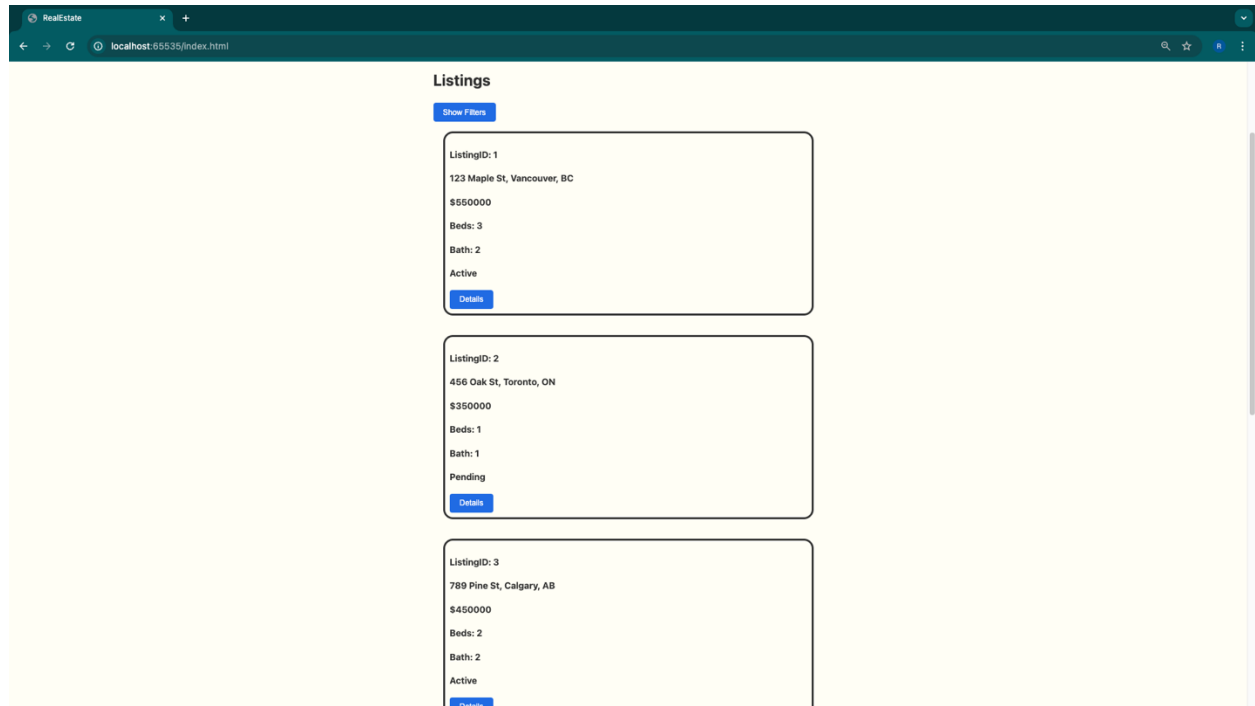
- Offer ID:
- Buyer Email:
- Listing ID:
- Offer Amount:
- Offer Status:
- Offer Date:
- Offer Expiry Date:
- Address:
- Postal Code:
- 
- [Back to Listings](#)

### JOIN

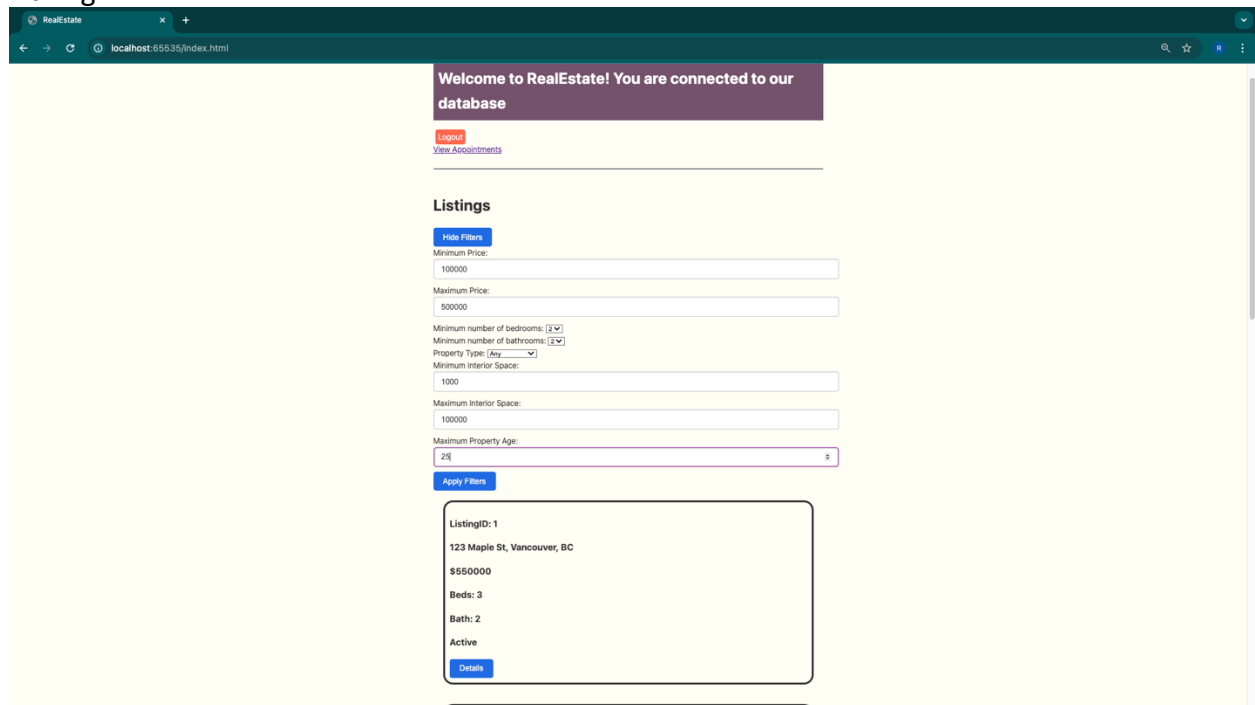
# University of British Columbia, Vancouver

## Department of Computer Science

### Before



### During



## After

Welcome to RealEstate! You are connected to our database

[Logout](#)  
[View Appointments](#)

### Listings

[Hide Filters](#)

Minimum Price: 100000

Maximum Price: 500000

Minimum number of bedrooms: 2

Minimum number of bathrooms: 2

Property Type: Any

Minimum Interior Space: 1000

Maximum Interior Space: 100000

Maximum Property Age: 25

[Apply Filters](#)

ListingID: 3

789 Pine St, Calgary, AB

\$450000

Beds: 2

Bath: 2

Active

[Details](#)

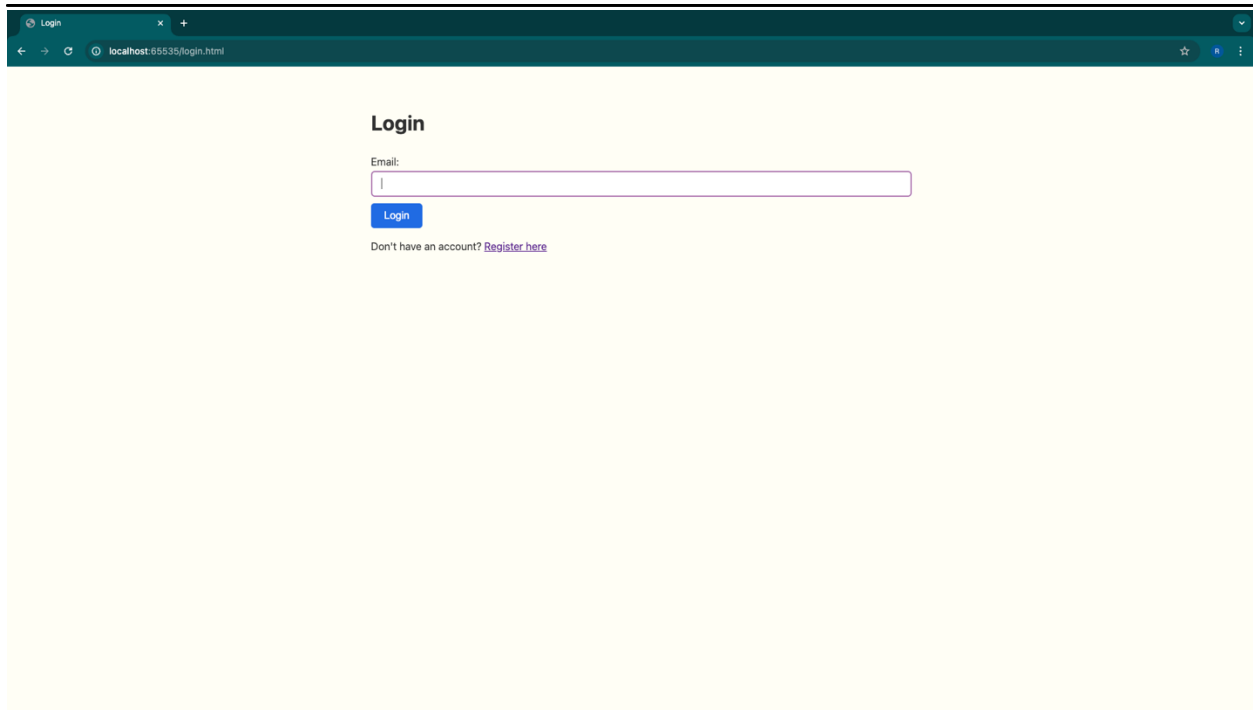
## Nested Aggregation with group by

Before

# University of British Columbia, Vancouver

## Department of Computer Science

---



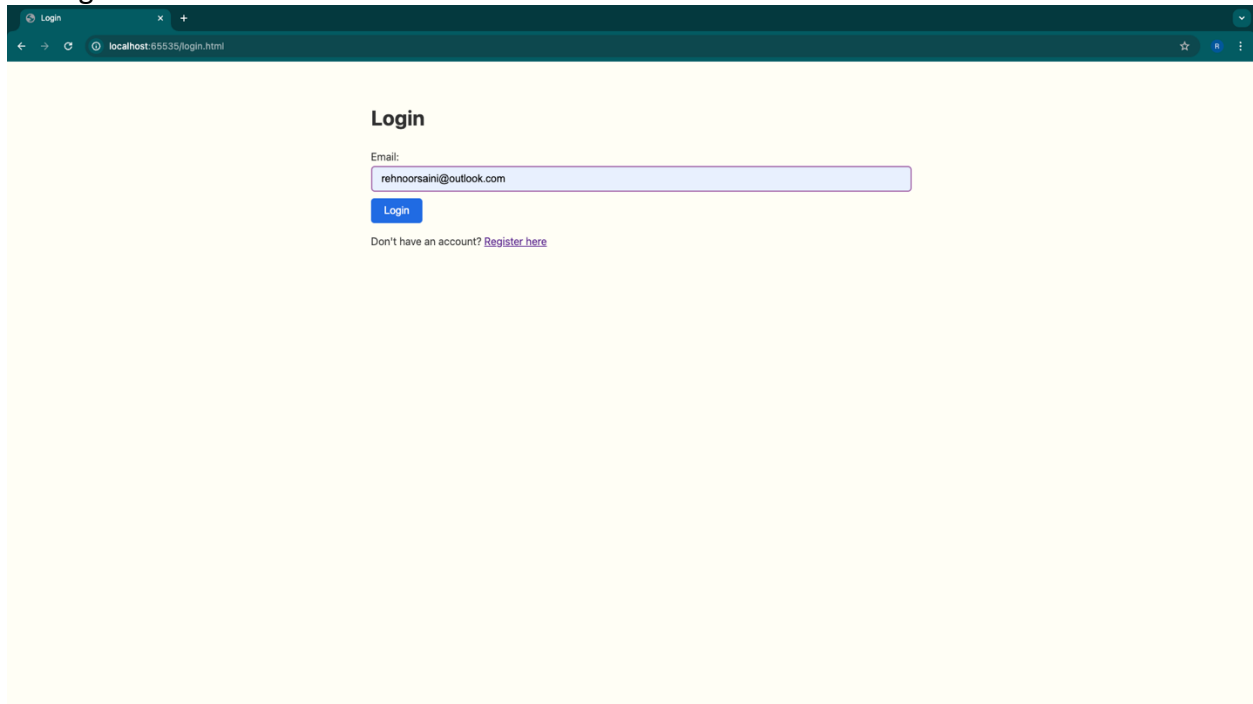
Login

Email:

Login

Don't have an account? [Register here](#)

During



Login

Email:

Login

Don't have an account? [Register here](#)

After

# University of British Columbia, Vancouver

## Department of Computer Science

---

---

### RealEstate Insights

Average offer price for Condos is \$310000  
Average offer price for Townhouses is \$460000  
Average offer price for Houses is \$426666  
Average offer price for Apartments is \$560000

## Projection

### Before

The screenshot shows a web browser window with the title 'Login'. The address bar displays 'localhost:65535'. The page content is as follows:

**Login**

Email:

Don't have an account? [Register here](#)

# University of British Columbia, Vancouver

## Department of Computer Science

### During

Login

Email:

Login

Don't have an account? [Register here](#)

### After

RealEstate

Welcome to RealEstate! You are connected to our database

[Logout](#)

[View Appointments](#)

Listings

Show Filters

ListingID: 1

123 Maple St, Vancouver, BC

\$550000

Beds: 3

Bath: 2

Active

Details

ListingID: 2

456 Oak St, Toronto, ON

\$350000

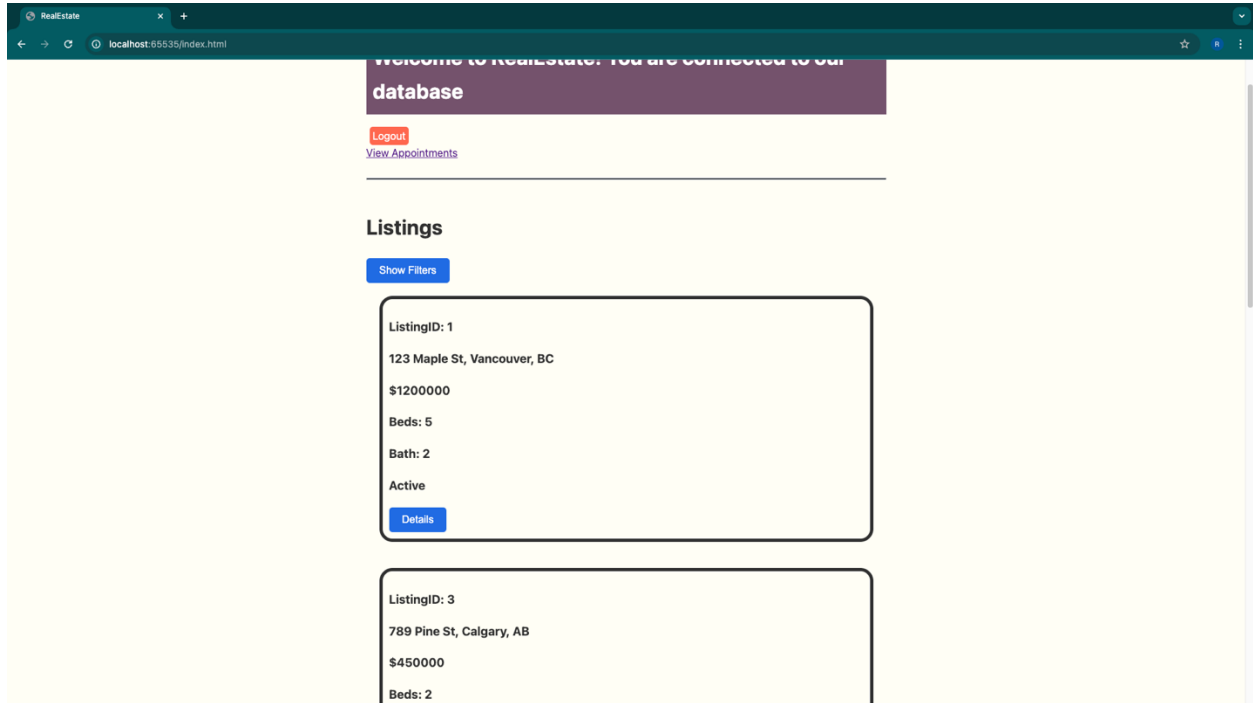
Details

# University of British Columbia, Vancouver

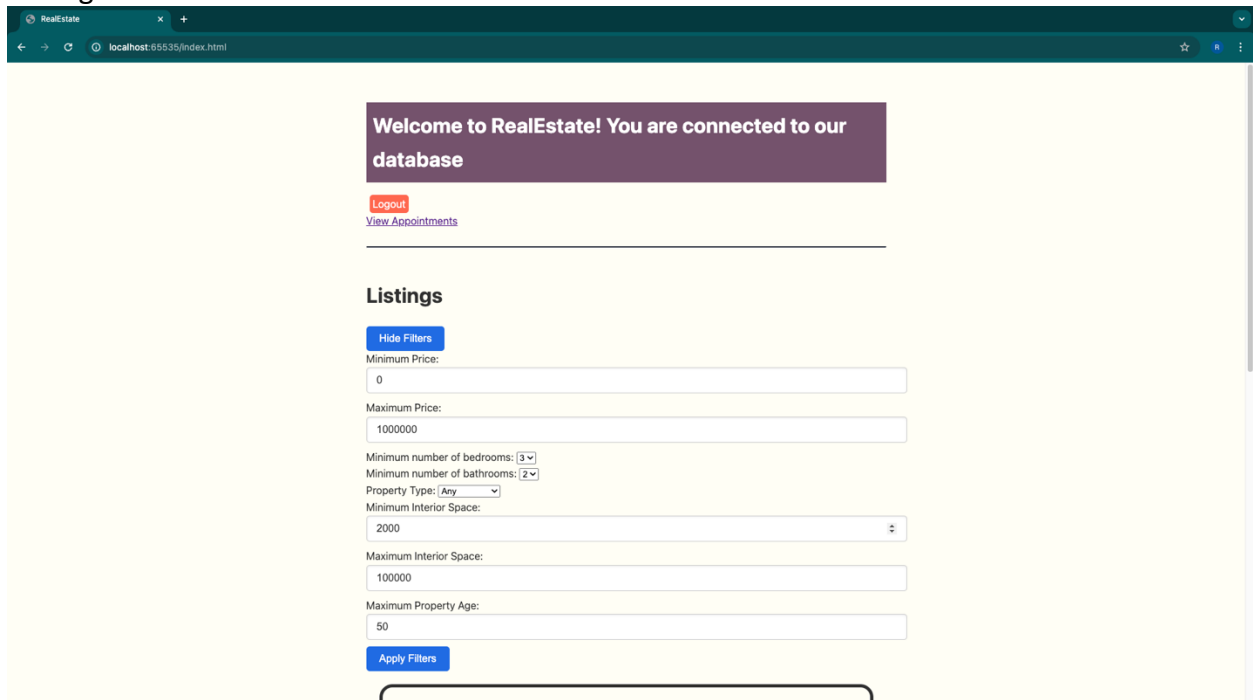
## Department of Computer Science

### Selection and Join

#### Before



#### During

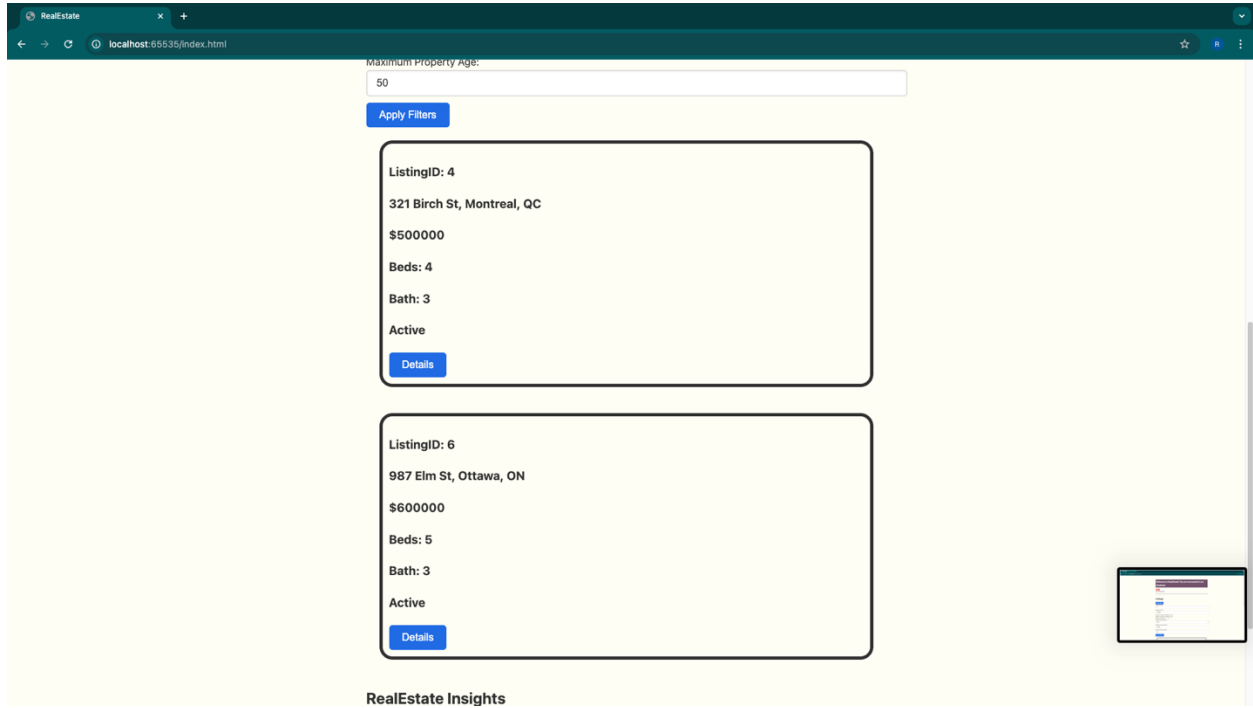




# University of British Columbia, Vancouver

## Department of Computer Science

### After



The screenshot shows a web browser window with the title "RealEstate" and the address bar displaying "localhost:65535/index.html". The browser's address bar also shows navigation icons (back, forward, refresh) and a star icon for bookmarks. The page content is a real estate application interface. At the top, there is a label "Maximum Property Age:" followed by a text input field containing the number "50". Below this input field is a blue button labeled "Apply Filters". The main content area displays two property listings, each enclosed in a rounded rectangular box. The first listing, labeled "ListingID: 4", shows the address "321 Birch St, Montreal, QC", the price "\$500000", "Beds: 4", "Bath: 3", and the status "Active". A blue "Details" button is located at the bottom of this listing box. The second listing, labeled "ListingID: 6", shows the address "987 Elm St, Ottawa, ON", the price "\$600000", "Beds: 5", "Bath: 3", and the status "Active". A blue "Details" button is also at the bottom of this listing box. In the bottom right corner of the page, there is a small, floating window or modal that appears to be a preview of a property listing. At the very bottom of the page, the text "RealEstate Insights" is visible.

Maximum Property Age:

50

Apply Filters

ListingID: 4

321 Birch St, Montreal, QC

\$500000

Beds: 4

Bath: 3

Active

Details

ListingID: 6

987 Elm St, Ottawa, ON

\$600000

Beds: 5

Bath: 3

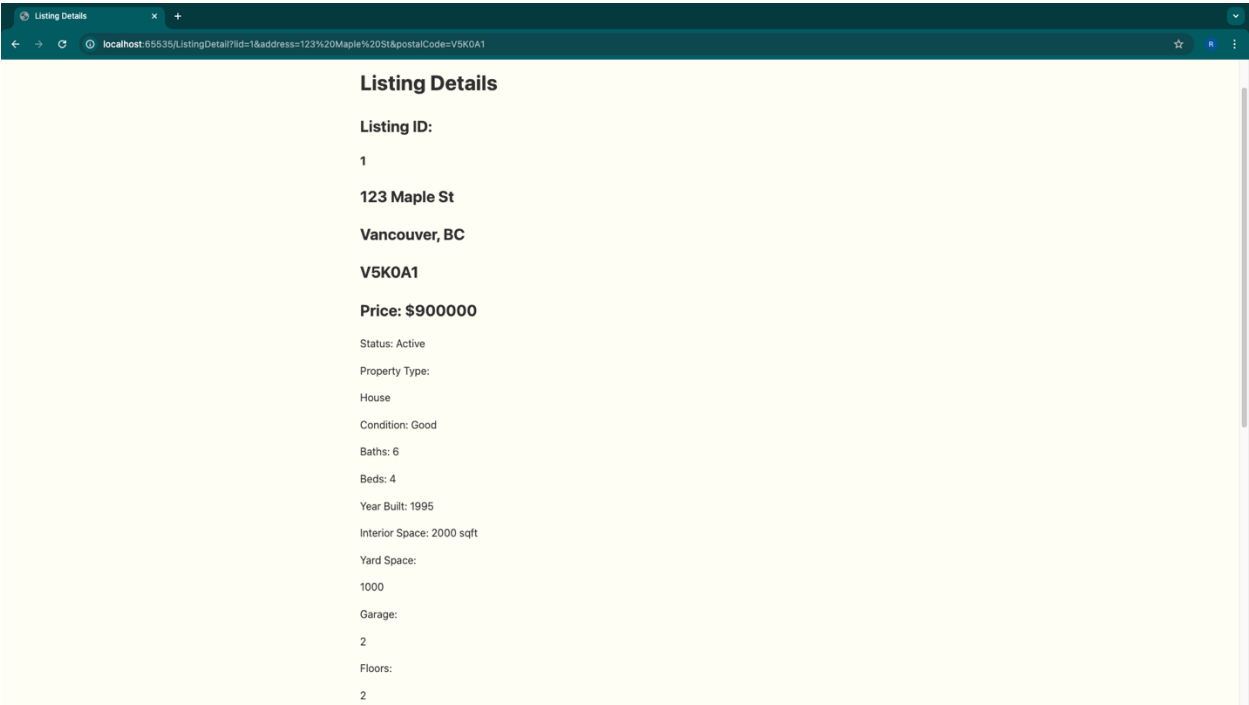
Active

Details

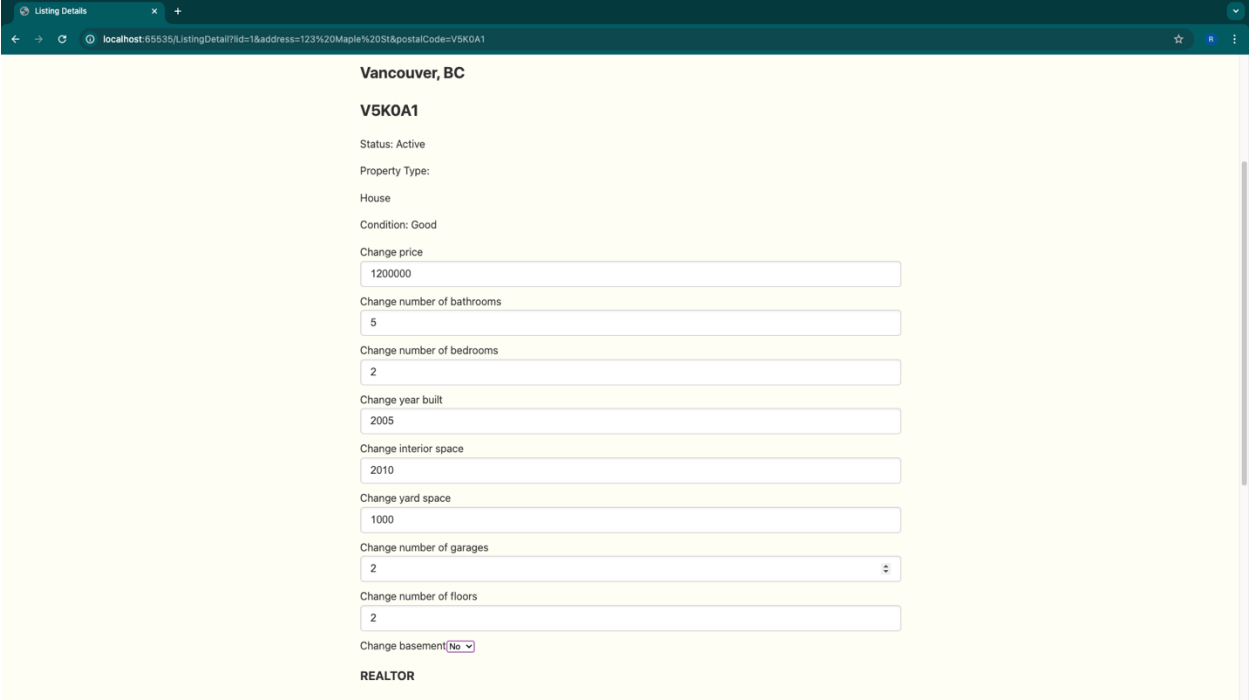
RealEstate Insights

Update

Before



During



# University of British Columbia, Vancouver

## Department of Computer Science

---

### After

