University of British Columbia, Vancouver
Department of Computer Science

CPSC 304 Project Cover Page

Milestone #: 1

Date: July 12, 2024

Group Number: 11

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

University of British Columbia, Vancouver
Department of Computer Science

Project Description

The domain of the application is Real Estate Management and Transactions, which encompasses activities related to buying, selling, and managing real estate properties. This includes property listings, viewings, appointments with realtors, making offers, and completing transactions, integrating service providers such as inspectors and contractors to enhance the process.

The database models several key aspects of this domain. Users, including buyers, sellers, and realtors, are differentiated by roles with attributes like UserID, Name, Email, Phone, and Role. Properties are detailed with attributes such as PropertyID, Address, City, Province, PostalCode, InteriorSpace, NumBeds, NumBaths, PropertyType, YearBuilt, and PropertyCondition. Listings manage the properties on the market with attributes like ListingID, ListingDate, ExpirationDate, Status, SellerID, and PropertyID.

Appointments facilitate scheduling property viewings, including attributes such as AppointmentID, Date, Time, Status, BuyerID, RealtorID, and MeetingPlace. Offers track the negotiation process with attributes like OfferID, Date, Amount, Status, BuyerID, SellerID, RealtorID, and OfferExpiryDate. Realtors manage property listings and facilitate transactions, with attributes such as RealtorID, Name, Email, Phone, RealEstateFirm, and YearsOfExperience. Amenities associated with properties are modeled with attributes like Name, Type, Distance, and Rating. The ISA relationships specialize property types into Houses, Townhouses, Apartments, and Condos, each with specific attributes relevant to their type.

For example, first-time home buyers can use the platform to search for properties based on their preferences, schedule viewings, make offers, and contact service providers for inspections. Property sellers can list their properties, manage viewings, and receive offers. Real estate agents can efficiently manage multiple listings, appointments, and transactions. Investors can find and manage investment properties, and landlords can list rental properties and manage applications.

This comprehensive modeling supports the entire lifecycle of real estate transactions, making the application robust and user-friendly for real-life scenarios in the real estate market.

University of British Columbia, Vancouver
Department of Computer Science

Database Specifications

The database will provide comprehensive functionality to support real estate transactions and management. Users will be able to register and log in, search for properties based on various criteria, and view detailed property information. They can schedule appointments for property viewings, make and track offers, and communicate with realtors and service providers. Additionally, the database will facilitate the management of property listings, user feedback, and transaction records, ensuring a seamless and organized process for buying, selling, and managing real estate properties.

Application Platform

The project will use the department-provided Oracle database to manage all data related to users, properties, appointments, offers, and transactions. The expected application technology stack includes HTML, CSS, and JavaScript for the frontend, with Node.js and Express or Python with Flask/Django for the backend. This stack ensures a robust, scalable, and efficient platform, though adjustments may be made as the project progresses based on further learning and requirements.

University of British Columbia, Vancouver
Department of Computer Science

ER Diagram