

Міністерство освіти і науки України
Національний університет «Запорізька Політехніка»

Кафедра програмних засобів

ЗВІТ

Дисципліна «Фреймворки розробки програмного забезпечення»

Робота №1

Тема «Розроблення технічного завдання»

Виконав варіант 19

Студент КНТ-122

Онищенко О. А.

Прийняли

Викладач

Зелік О. В.

2024

ВСТУП

Проект «Застосунок торгівлі нерухомістю» застосовуватиметься у предметній області Нерухомості. В ній люди продають або купують нерухомість: будинки, квартири, ділянки, тощо. Застосунок створюється для полегшення виставлення нерухомості для продавців і спрощення придбання для покупців. Для цього буде створено зручний інтерфейс для проведення процедур купівлі-продажу онлайн. Програма зможе бути використана на персональних комп'ютерах із операційною системою Windows.

1 ПІДСТАВА ДЛЯ РОЗРОБКИ

Підставою розробки є видане завдання університетської дисципліни «Фреймворки розробки програмного забезпечення». Завдання розбите на кілька підзавдань, при завершенні виконання усіх отримаємо кінцевий проєкт.

2 ВИМОГИ ДО ПРОГРАМИ

Вимоги до програми формуватимуться на основі індивідуального завдання. Завдання обране за номером варіанта 19, текст наведено нижче.

Програмне забезпечення продажу та придбання нерухомості. Всю нерухомість поділено на приватні будинки, окремі квартири та квартири в новобудовах. Приватні будинки та окремі квартири можуть виставлятися на продаж звичайними користувачами. Квартири в новобудовах можуть виставлятися лише спеціальними менеджерами. Користувачі можуть подавати заявки на перегляд житла, передивлятися стан заявок, а також після перегляду виставляти оцінку. Власники можуть передивлятися

заявки на всі їх пропозиції або на обрані, підтверджувати перегляд або скасовувати, знімати пропозиції (кількість кімнат, площа, рік будівництва, зображення, планування, тощо), змінювати.

2.1 Функціональні вимоги

Програма має містити хоча б два розділи: оголошення, заявки.

У розділі оголошень користувачі мають могли переглядати доступні оголошення. Для кожного має бути зазначено кількість кімнат, площа, рік будівництва (або десятиліття), зображення, тип планування. Під кожним оголошенням має бути зазначена його ціна, ім'я продавця та можливість організувати зустріч.

У розділі заявок користувачі мають могли бачити подані заявки на зустрічі. Для кожної має бути зазначено її статус, оцінка (недоступна якщо оголошення не переглянуто), назва оголошення.

Для забезпечення торгівлі необхідно додано можливість створювати користувацькі профілі. Для цього можна використати базу даних з даними про кожного користувача з їхнім іменем, паролем та особистими даними програми. Також має бути реалізована властивість менеджера як статус. Користувач має мати можливість запитувати статус менеджера. Отримати статус менеджера можна якщо придбано або виставлено хоча б одне оголошення.

2.2 Вимоги до надійності

Надійність роботи програми має бути забезпечена через використання бази даних та обробки виключних ситуацій. Потенційні виключні ситуації: користувач вже існує, неправильні облікові дані користувача, оголошення не існує, зустрічі не існує, не можна поставити

оцінку бо статус зустрічі не оновлено, не можна придбати оголошення бо зустріч не відбулася. Обробка виключних ситуацій має здійснюватись виведенням повідомлення про помилку та, можливо, записом помилки до журналу або консолі.

2.3 Умови експлуатації

Для полегшення користуванням програми має бути зроблена секція з поясненням функцій та інформацією про можливості програми. У секції мають бути наведені різні розділи для функцій та зображення для легшого розуміння.

Також для легшого користування має бути розроблений простий світлий графічний інтерфейс з підказками користувачеві.

2.4 Вимоги до технічних засобів

Оскільки програма розробляється сучасними засобами під сучасні системи, вимоги до її використання мають бути мінімальними. Потенційними речима, які можуть сповільнювати програму можуть бути: зображення, надто довгі списки. Для вирішення обох питань можна обмежити кількість елементів які завантажуються на сторінку за один раз. Якщо елементів більше ніж обмежена кількість, тоді мають бути створені додаткові сторінки, а елементи перенесені до них.

2.5 Вимоги до інформційної і програмної сумісності

Сумісність такої програми не може бути широкою, бо створюватиметься вона засобами призначеними для системи Windows.

Робота з іншими програмами також не передбачена. Сумісність може бути хіба між окремими компонентами програми. Це можна забезпечити створенням класів для груп функцій та розробку функцій, що відповідають лише за одну задачу.

2.6 Вимоги до маркування та упакування

Пакування програми може бути здійснене вбудованими засобами середовища розробка. Це може дати можливість створити програму-встановлювач яка дозволить користувачам встановлювати проєкт на свої системи.

Маркування здійснюватиметься написанням назви програми у заголовку вікна та на інформаційній сторінці.

2.7 Вимоги до транспортування та збереження

Програму має мати встановлювальник. При транспортуванні можна буде завантажувати лише його, а розпаковану версію він буде ставити на систему користувачеві. Зберігатися програма буде на диску користувача як окрема тека.

2.8 Спеціальні вимоги

Вимоги до користувацького інтерфейсу: згідно завдання має бути зроблений засобами C# і фреймворком WPF. Має бути простий, світлий та зрозумілий користувачеві з підказками де потрібно та інформаційною сторінкою.

3 ВИМОГИ ДО ПРОГРАМНОЇ ДОКУМЕНТАЦІЇ

Код програми має бути зрозуміло написаним: всі змінні мають бути чітко названі, класи об'єднувати схожі функції, і код має бути форматований охайно.

Документування має бути зроблене написанням коментарів у місцях де це необхідно. Коментарі ці мають бути Українською, стислим й зрозумілими.

4 ТЕХНІКО-ЕКОНОМІЧНІ ПОКАЗНИКИ

Програма має коректно запускатись, виводити діагностичну інформацію у консоль та працювати без перешкод. При виникненні помилок вони мають бути оброблені виведенням помилки на екран та повідомлення у консоль розробникам.

5 СТАДІЇ ТА ЕТАПИ РОЗРОБКИ

Розробка програми, згідно завдання, має складатися з восьми етапів:

1. Розробка й написання технічного завдання
2. Розробка архітектури програми
3. Проектування бази даних
4. Розробка консольного застосунку для тестування функцій
5. Розширення функціональу
6. Написання додаткових функцій
7. Розробка графічного інтерфейсу
8. Написання авторського свідоцтва

6 ПОРЯДОК КОНТРОЛЮ ТА ПРИЙМАННЯ

Приймання кожної роботи має здійснюватись онлайн на заняттях або на платформі Google Classroom. Виконавець має завантажити звіти з робіт на платформу, викладач має перевірити й оцінити їх.

ВИСНОВКИ

Проєкт використовуватиме сучасні засоби розробки що полегшить процес. Основні етапи розбито на окремі роботи що також полегшить виконання. При реалізації буде багато змін тому конкретний функціонал ані прототипи інтерфейсу не вказані у технічному завданні.