НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ

«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ

імені ІГОРЯ СІКОРСЬКОГО»

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ (повна назва кафедри, циклової комісії)

**КУРСОВА РОБОТА**

з \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

(назва дисципліни)

на тему:\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Студента (ки) \_\_\_\_\_ курсу \_\_\_\_\_\_ групи

Спеціальності 121 «Інженерія програмного забезпечення»

Керівник \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

(посада, вчене звання, науковий ступінь, прізвище та ініціали)

Кількість балів: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Національна оцінка \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Члени комісії

(підпис) (вчене звання, науковий ступінь, прізвище та ініціали)

(підпис) (вчене звання, науковий ступінь, прізвище та ініціали)

Київ- 2024рік

**ЗМІСТ**

ВСТУП.................................................................................................................................. 4

1 ПОРЯДОК ВИКОНАННЯ КУРСОВОЇ РОБОТИ......................................................... 5

1.1 Об'єктно-орієнтований аналіз предметної області ................................................. 5

1.2 Об'єктно-орієнтоване проектування ........................................................................ 6

1.3 Об'єктно-орієнтоване програмування ...................................................................... 6

1.4 Тестування програми ................................................................................................. 7

1.5 Оформлення пояснювальної записки....................................................................... 8

1.6 Захист курсової роботи.............................................................................................. 8

2 ЗМІСТ ПОЯСНЮВАЛЬНОЇ ЗАПИСКИ ....................................................................... 9

3 ПРАВИЛА ОФОРМЛЕННЯ ПОЯСНЮВАЛЬНОЇ ЗАПИСКИ ................................ 14

3.1 Загальні вимоги ........................................................................................................ 14

3.2 Нумерація.................................................................................................................. 15

3.3 Оформлення цитат і переліку посилань................................................................. 16

3.4 Оформлення додатків .............................................................................................. 17

4 ПОРЯДОК ВИКОНАННЯ ТА ЗАХИСТУ КУРСОВОЇ РОБОТИ............................. 18

4.1 Хід виконання та захисту курсової роботи ........................................................... 18

4.2 Критерії оцінювання курсової роботи ................................................................... 20

5 ТЕМИ ІНДИВІДУАЛЬНИХ ЗАВДАНЬ НА КУРСОВУ РОБОТУ........................... 25

5.1 Варіанти завдань ............................................Ошибка! Закладка не определена.

5.1.1 Тематика базових індивідуальних завданьОшибка! Закладка не

определена.

5.1.2 Тематика додаткових індивідуальних завданьОшибка! Закладка не

определена.

СПИСОК ЛІТЕРАТУРИ................................................................................................... 26

ДОДАТОК А ТЕХНІЧНЕ ЗАВДАННЯ.......................................................................... 27

ДОДАТОК Б ТИТУЛЬНА СТОРІНКА........................................................................... 29

ДОДАТОК В ЛИСТ ЗАВДАННЯ.................................................................................... 30

ДОДАТОК Г ТЕКСТИ ПРОГРАМНОГО КОДУ........................................................... 32

**Вступ**

Виконання курсової роботи з дисципліни "Основи програмування" є невід’ємною частиною навчального процесу студентів, які навчаються за спеціальністю 121 "Інженерія програмного забезпечення" на бакалаврському рівні. Робота має на меті не лише поглиблення теоретичних знань, отриманих під час вивчення курсу, але й їх застосування для розв’язання конкретних фахових задач, що сприяє формуванню висококваліфікованих фахівців у галузі програмування.

Тема моєї курсової роботи — "ОСББ. Управління будинком. Електронний довідник даних про мешканців та орендарів багатоквартирного будинку". Проект передбачає розробку програмного забезпечення, що дозволить автоматизувати процес збору, зберігання та обробки інформації про мешканців та орендарів: їхні особисті дані, номер квартири, параметри проживання, вартість оренди та комунальних послуг.

Основними завданнями, які стоять переді мною під час виконання курсової роботи, є:

* Розробка об'єктно-орієнтованої моделі даних;
* Програмування функціоналу для розрахунку квартплати з урахуванням різних факторів, таких як площа квартири, кількість мешканців, споживання води та електроенергії;
* Створення інтуїтивно зрозумілого графічного інтерфейсу користувача;
* Реалізація механізмів збору та аналізу даних для розрахунку доходів і витрат ОСББ.

Курсова робота вимагає застосування сучасних технологій розробки програмного забезпечення і використання навичок, набутих в процесі навчання, як у теоретичному, так і в практичному аспектах. Завершення проекту потребує від студента здатності до самостійної роботи, аналітичного мислення та вміння розв’язувати комплексні задачі. За результатами виконаної роботи будуть сформовані висновки, які покажуть ефективність запропонованого рішення.

Цей проект відображає практичне застосування набутих знань і спрямований на поліпшення управління житловими комплексами через використання інноваційних програмних рішень, що є актуальним і сучасним викликом в галузі інженерії програмного забезпечення.

**1 ПОРЯДОК ВИКОНАННЯ КУРСОВОЇ РОБОТИ**

Основні етапи виконання курсової роботи:

− об'єктно-орієнтований аналіз предметної області задачі;

− об'єктно-орієнтоване проектування архітектури програмної системи;

− об'єктно-орієнтоване програмування;

− тестування програми;

− оформлення пояснювальної записки;

− захист курсової роботи.

* 1. **Об'єктно-орієнтований аналіз предметної області**

А) Предметна область цього програмного забезпечення присвячена управлінню багатоквартирними будинками та обліку інформації про мешканців, орендарів, комунальні витрати та витрати на ремонт.

Терміни та поняття:

* Багатоквартирний будинок
* Квартира
* Кімната
* Орендар (Tenant)
* Власник квартири (LandLord)
* Витрати на комунальні послуги (UtilityExpenses)
* Витрати на ремонт (RepairExpense)
* Оренда
* Дохід від оренди

Суб'єкти:

* Власники квартир (LandLord)
* Орендарі (Tenant)
* Об'єднання співвласників багатоквартирного будинку (ОСББ)

Об'єкти:

* Багатоквартирний будинок (Building)
* Квартира (Apartment)
* Кімната (Room)

Способи взаємодії суб'єктів:

* Власники квартир здають їх в оренду орендарям
* Орендарі сплачують орендну плату власникам квартир
* ОСББ збирає інформацію про мешканців, орендарів, комунальні витрати та витрати на ремонт

Способи використання об'єктів:

* Багатоквартирний будинок є сукупністю квартир
* Квартира складається з кімнат та використовується для проживання
* Кімната є частиною квартири та використовується для різних цілей (спальня, вітальня тощо)

Закономірності:

* Вартість оренди квартири залежить від її площі, кількості кімнат та місцезнаходження
* Комунальні витрати залежать від кількості мешканців, споживання ресурсів (електрика, вода, опалення тощо)
* Витрати на ремонт виникають після закінчення терміну оренди та залежать від ступеня зносу квартири
* Дохід власника квартири формується від орендної плати за вирахуванням витрат на комунальні послуги та ремонт

Отже, ця предметна область охоплює управління багатоквартирними будинками, облік інформації про мешканців, орендарів, комунальні витрати та витрати на ремонт, а також взаємодію між власниками квартир, орендарями та об'єднанням співвласників будинку.

Б) Словник предметної області включатиме такі елементи:

Іменники:

1. Будинок (Building)
2. Квартира (Apartment)
3. Кімната (Room)
4. Орендар (Tenant)
5. Власник (LandLord)
6. Оренда (Rent)
7. Витрати (Expenses)
8. Комунальні послуги (Utilities)
9. Ремонт (Repair)
10. Дохід (Income)
11. Об'єднання співвласників (Owners' Association)
12. Платіж (Payment)
13. Площа (Area)
14. Поверх (Floor)
15. Контракт (Contract)
16. Термін (Term)

Дієслова, пов'язані з іменниками:

1. Будинок: будувати, володіти, управляти, обслуговувати.
2. Квартира: орендувати, винаймати, проживати, ремонтувати, обладнати.
3. Кімната: використовувати, меблювати, прибирати.
4. Орендар: орендувати, винаймати, сплачувати, проживати, звітувати.
5. Власник: володіти, здавати в оренду, отримувати дохід, ремонтувати.
6. Оренда: укладати, сплачувати, поновлювати, припиняти.
7. Витрати: нараховувати, сплачувати, відшкодовувати.
8. Комунальні послуги: споживати, оплачувати, розраховувати.
9. Ремонт: проводити, оцінювати, фінансувати.
10. Дохід: обчислювати, отримувати, розподіляти.
11. Об'єднання співвласників: управляти, координувати, представляти.
12. Платіж: здійснювати, обробляти, відстежувати.
13. Площа: обчислювати, вимірювати, оцінювати.
14. Поверх: розташовувати, будувати.
15. Контракт: укладати, переглядати, поновлювати.
16. Термін: встановлювати, продовжувати, закінчувати.

Цей словник охоплює основні терміни та поняття, пов'язані з управлінням багатоквартирними будинками, орендою квартир, комунальними витратами, ремонтами та взаємодією між учасниками процесу.

В) Функціональні вимоги до розроблюваної системи управління багатоквартирними будинками:

1. Облік інформації про будинки:

- Можливість додавання, видалення та редагування інформації про будинки (назва, кількість поверхів, адреса тощо).

- Перегляд списку всіх будинків у системі.

2. Облік інформації про квартири:

- Можливість додавання, видалення та редагування інформації про квартири (номер, кількість кімнат, площа, поверх, вартість за квадратний метр).

- Прив'язка квартири до певного будинку.

- Перегляд списку всіх квартир у певному будинку.

3. Облік інформації про власників квартир (домовласників):

- Можливість додавання, видалення та редагування інформації про власників (ім'я, контактна інформація, квартири у власності).

- Перегляд списку всіх власників квартир у системі.

- Перегляд інформації про квартири, якими володіє певний власник.

4. Облік інформації про орендарів:

- Можливість додавання, видалення та редагування інформації про орендарів (ім'я, контактна інформація, орендована квартира).

- Перегляд списку всіх орендарів у системі.

- Перегляд інформації про квартиру, яку орендує певний орендар.

5. Управління орендою квартир:

- Можливість встановлення терміну оренди для квартири.

- Розрахунок вартості оренди квартири на основі площі, кількості кімнат та вартості за квадратний метр.

- Облік доходу від оренди для власника квартири.

6. Облік комунальних витрат:

- Можливість додавання та редагування інформації про комунальні витрати для певної квартири (вартість електроенергії, води, опалення тощо).

- Перегляд історії комунальних витрат для квартири.

7. Облік витрат на ремонт:

- Можливість додавання та редагування інформації про витрати на ремонт для певної квартири (опис ремонту, вартість).

- Перегляд історії витрат на ремонт для квартири.

8. Звітність та аналітика:

- Генерація звітів про дохід від оренди для власників квартир.

- Генерація звітів про комунальні витрати та витрати на ремонт для квартир.

- Аналіз та візуалізація даних (наприклад, графіки доходів або витрат за певний період).

9. Обробка виключних ситуацій:

- Належна обробка виключних ситуацій, таких як некоректний введений користувачем вхідний дані, порушення обмежень даних тощо.

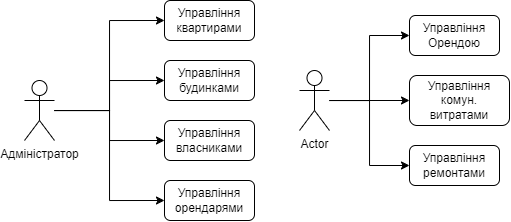
Ці функціональні вимоги охоплюють основні аспекти управління багатоквартирними будинками, обліку інформації про власників, орендарів, квартири, оренду, комунальні витрати та ремонти, а також генерацію звітності та аналітики.

Приблизний сценарій поведінки об'єктів в системі управління багатоквартирними будинками:

1. Створення об'єктів класу Building (будинки):
   * Адміністратор системи створює нові об'єкти класу Building, вказуючи назву будинку та кількість поверхів.
2. Додавання квартир до будинку:
   * Для кожного будинку адміністратор створює об'єкти класу Apartment, вказуючи номер квартири, кількість кімнат, поверх, вартість за квадратний метр та прив'язуючи її до відповідного об'єкту Building.
3. Створення об'єктів класу LandLord (власників квартир):
   * Адміністратор створює об'єкти класу LandLord для кожного власника квартири, вказуючи їх контактні дані.
   * Для кожного власника квартири встановлюється зв'язок з відповідними об'єктами класу Apartment, які він володіє.
4. Здача квартир в оренду:
   * Власник квартири (об'єкт LandLord) встановлює терміни оренди для своїх квартир (об'єктів Apartment) та визначає вартість оренди.
   * Система розраховує дохід власника квартири від оренди на основі вартості оренди та терміну.
5. Створення об'єктів класу Tenant (орендарів):
   * Адміністратор або власник квартири створює об'єкти класу Tenant для кожного орендаря, вказуючи їх контактні дані.
   * Для кожного орендаря встановлюється зв'язок з об'єктом класу Apartment, який він орендує.
6. Облік комунальних витрат:
   * Власник квартири або адміністратор створює об'єкти класу UtilityExpenses для кожної орендованої квартири, вказуючи витрати на електроенергію, воду, опалення тощо за певний період (місяць).
   * Об'єкти UtilityExpenses зберігаються у відповідних об'єктах класу Apartment.
7. Облік витрат на ремонт:
   * Після закінчення терміну оренди власник квартири створює об'єкти класу RepairExpense для своїх квартир, вказуючи опис ремонту та вартість.
   * Об'єкти RepairExpense зберігаються у відповідних об'єктах класу Apartment.
8. Генерація звітів та аналітики:
   * Адміністратор або власник квартири може запросити звіти про дохід від оренди для певного власника квартири або для всіх власників.
   * Адміністратор або власник квартири може запросити звіти про комунальні витрати та витрати на ремонт для певної квартири або для всіх квартир.
   * Система може генерувати графіки та візуалізації на основі даних про доходи, витрати тощо.
9. Обробка виключних ситуацій:
   * У випадку некоректного введення даних користувачем або порушення обмежень даних, система повинна належним чином обробляти ці виключні ситуації та повідомляти користувача про помилки.

Цей сценарій демонструє, як об'єкти різних класів (Building, Apartment, LandLord, Tenant, UtilityExpenses, RepairExpense) взаємодіють між собою для виконання різних функцій системи управління багатоквартирними будинками, таких як облік інформації про будинки, квартири, власників та орендарів, управління орендою, облік комунальних витрат та ремонтів, а також генерація звітності та аналітики.

Діаграммe прецедентів можна побачити на - **рис 1.1**



**Рис 1.1**

* 1. **Об'єктно-орієнтоване проектування**
  2. **Об'єктно-орієнтоване програмування**