

Міністерство освіти і науки України
НТУУ «КПІ ім. Ігоря Сікорського»
Навчально-науковий інститут атомної та теплової енергетики
Кафедра цифрових технологій в енергетиці

Лабораторна робота №5
з дисципліни
«Проектування інформаційних систем»
Тема «Проектування UML-діаграм для ІС засобами CASE
систем»
Варіант №18

Студента 4-го курсу НН ІАТЕ гр. ТР-12
Ковальова Олександра
Перевірів: ст. в., Гурін Артем Леонідович

Мета роботи. Навчитися будувати діаграми UML при проектуванні інформаційної системи.

Завдання: Розробити діаграму прецедентів інформаційної системи відповідно варіанту предметної області. Використати для цього CASE систему Rational Rose або іншу. Розробити діаграму класів (4-6 об'єктів інформаційної системи). Згенерувати в Rational Rose код програмної системи в Java або C++ опису класів. Розробити діаграму кооперації (взаємодії). Розробити діаграму послідовності.

Індивідуальне завдання: *ІС обліку історичних пам'яток м. Житомир.*

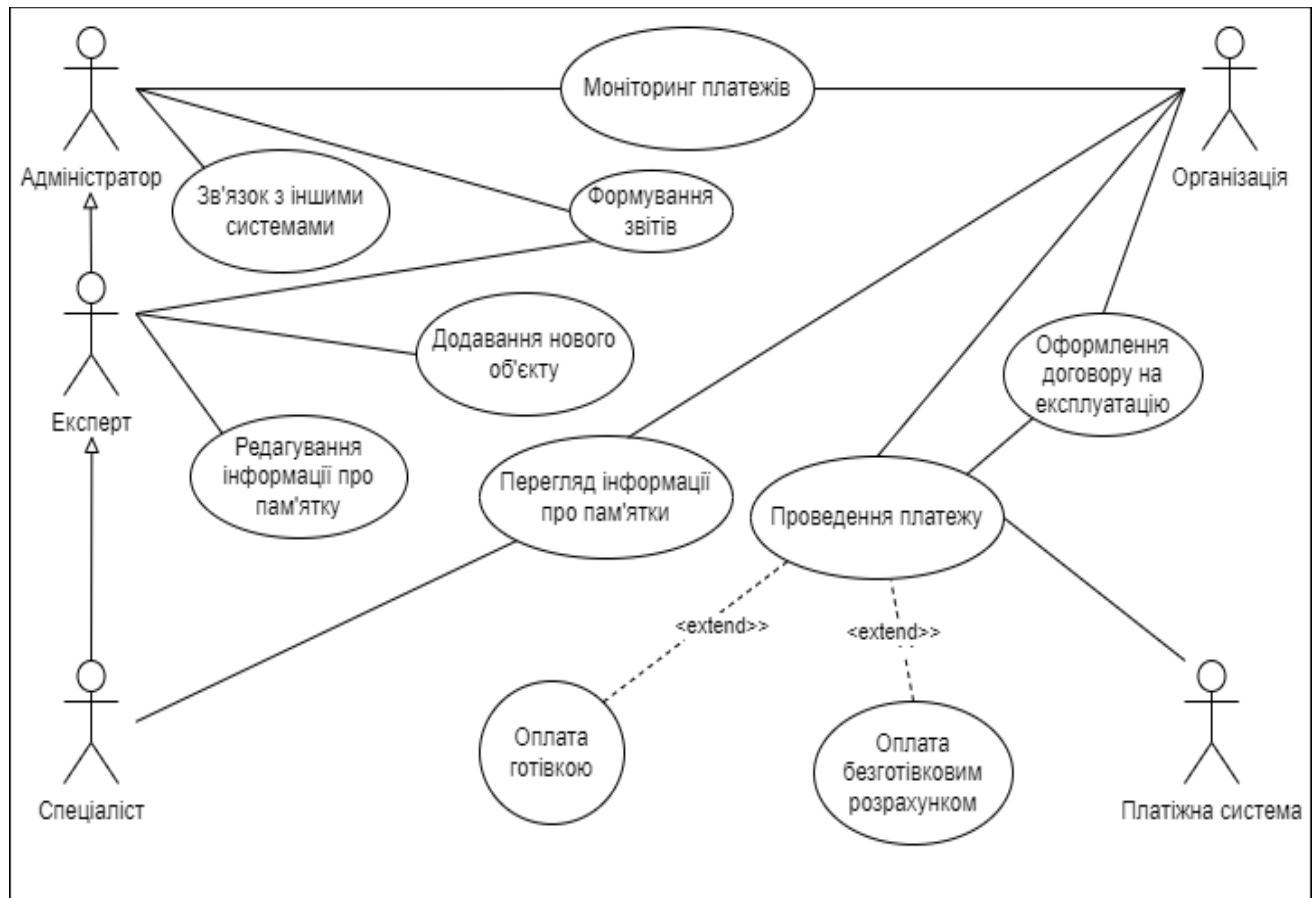
Зразковий зміст бази даних:

- код об'єкта, найменування, фотографії;
- історичні характеристики об'єкта (стародавність та ін.), текстова характеристика об'єкта;
- поточний стан об'єкта, для чого використовується в даний час, вартість об'єкта, якщо об'єкт в експлуатації – ким (інформація про експлуатуючу організацію або приватну особу);
- інше.

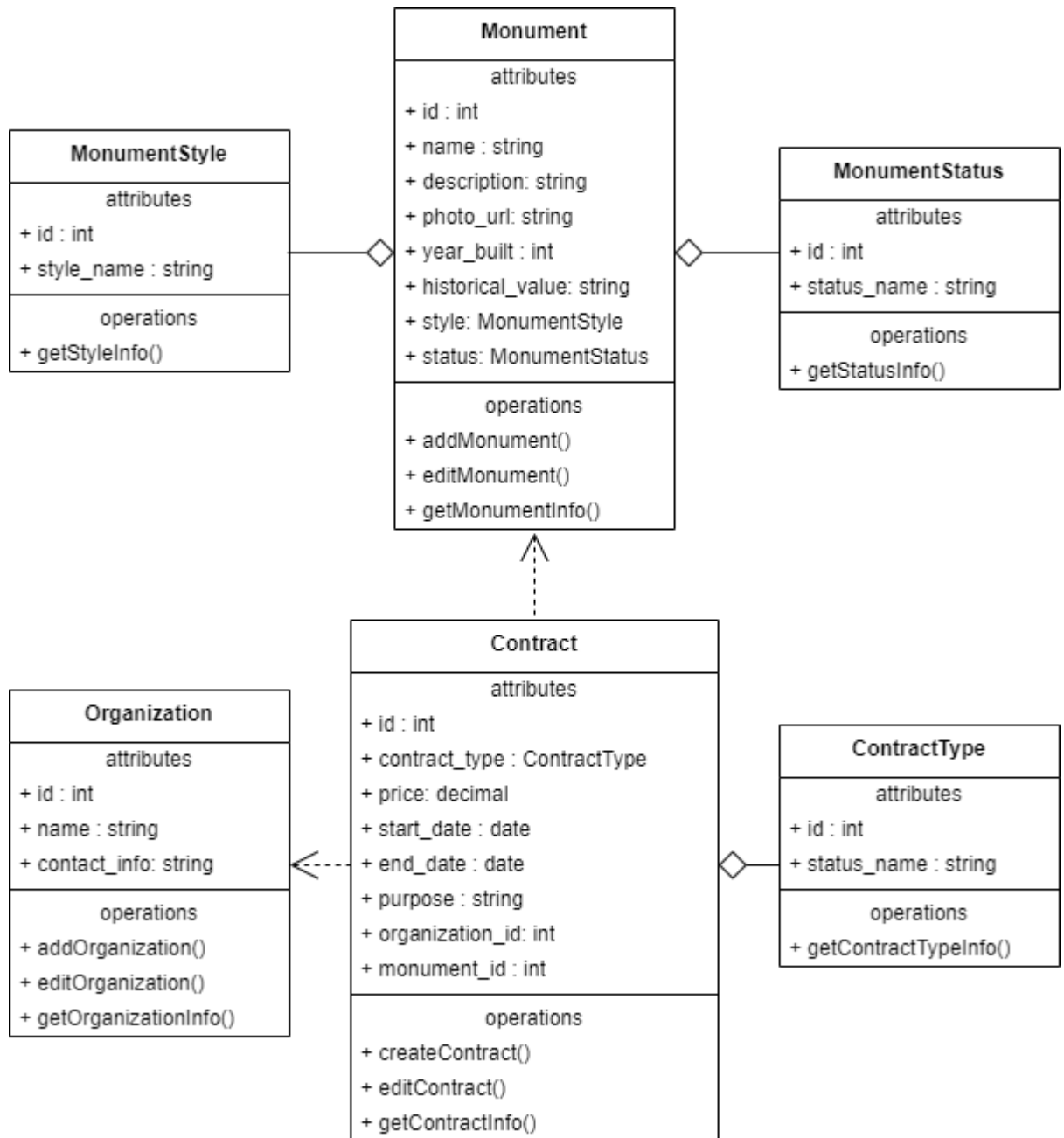
Хід роботи

Для виконання роботи буде використана більш вживана система *draw.io*.

Діаграма прецедентів (*Use-Case*):



Діаграма класів:



Згенерований код (Java):

```

class Monument {
    private int id;
    private String name;
    private String description;
    private String photoUrl;
    private int yearBuilt;
    private String historicalValue;
    private MonumentStyle style;
    private MonumentStatus status;

    public void addMonument() {
        // Логіка додавання пам'ятки
    }
}

```

```

        public void editMonument() {
            // Логіка редагування пам'ятки
        }

        public void getMonumentInfo() {
            // Логіка отримання інформації про пам'ятку
        }
    }

    class MonumentStyle {
        private int id;
        private String styleName;

        public void getStyleInfo() {
            // Логіка отримання інформації про стиль
        }
    }

    class MonumentStatus {
        private int id;
        private String statusName;

        public void getStatusInfo() {
            // Логіка отримання інформації про статус
        }
    }

    class Organization {
        private int id;
        private String name;
        private String contactInfo;

        public void addOrganization() {
            // Логіка додавання організації
        }

        public void editOrganization() {
            // Логіка редагування інформації про організацію
        }

        public void getOrganizationInfo() {
            // Логіка отримання інформації про організацію
        }
    }

    class Contract {
        private int id;
        private ContractType contractType;
        private decimal price;
        private Date startDate;
        private Date endDate;
        private String purpose;
        private Organization organization;
        private Monument monument;

        public void createContract() {
            // Логіка створення договору
        }

        public void editContract() {
            // Логіка редагування договору
        }

        public void getContractInfo() {

```

```

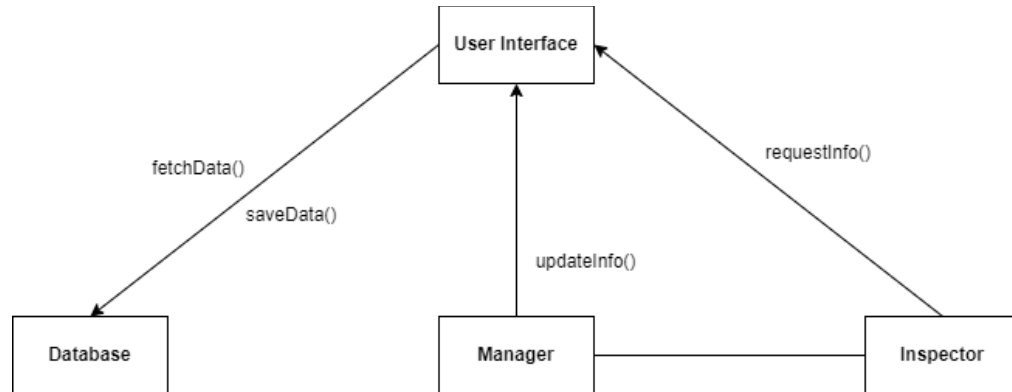
        // Логіка отримання інформації про договір
    }
}

class ContractType {
    private int id;
    private String typeName;

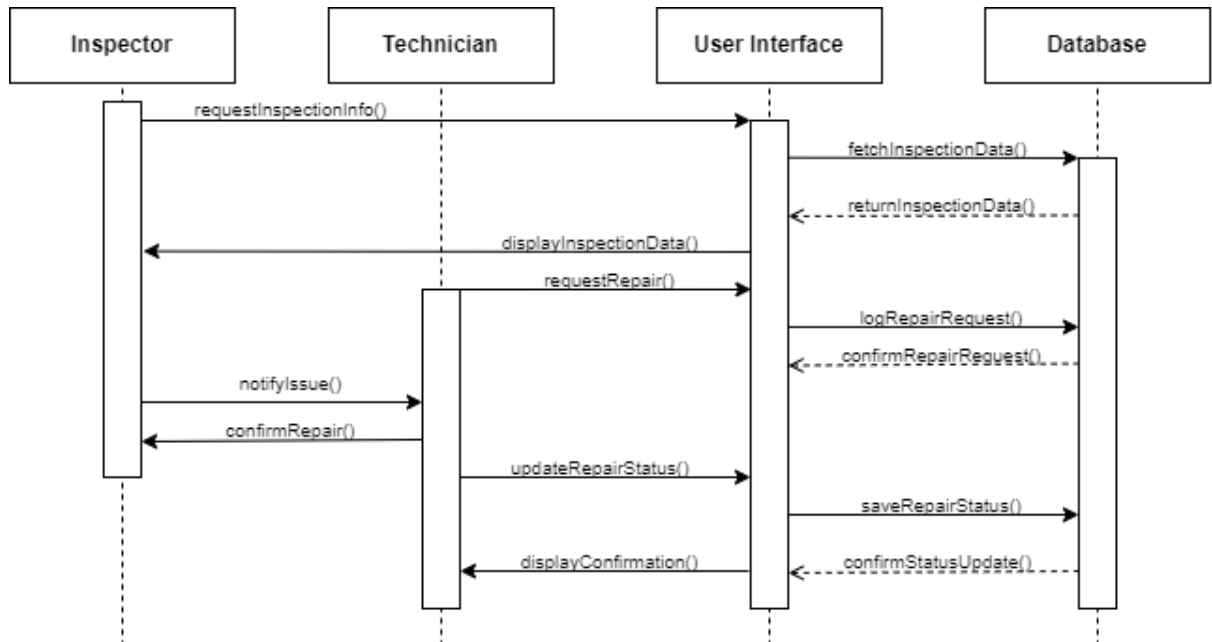
    public void getContractTypeInfo() {
        // Логіка отримання інформації про тип договору
    }
}

```

Діаграма колаборацій:



Діаграма послідовності:



Висновок: У результаті виконання лабораторної роботи була успішно досягнута мета — навчитися будувати UML діаграми. Створені діаграми прецедентів, класів, кооперації та послідовності надали чітке уявлення про структуру та взаємодії компонентів системи. Отримані результати підтверджують їх важливість у процесі проектування та документації інформаційних систем.