 МІНІСТЕРСТВО ОСВІТИ I НАУКИ УКРАЇНИ

НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ

«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ   
ІМЕНІ ІГОРЯ СІКОРСЬКОГО»

ФАКУЛЬТЕТ БІОМЕДИЧНОЇ ІНЖЕНЕРІЇ

КАФЕДРА БІОМЕДИЧНОЇ КІБЕРНЕТИКИ

**Лабораторна робота №1**

з дисципліни «Системи баз даних»

на тему: «Аналіз сучасних систем управління базами даних»

Варіант 7

**Виконав:**

студент гр. БС-03

Затуловський Г. А.

**Перевірив:**

ст.вик. [Сердаковський В. С.](https://schedule.kpi.ua/lecturers?lecturerId=f5dbaf77-ed72-47dd-bf2c-a135a78a44c8)

доц. [Лісневський Р. О.](https://schedule.kpi.ua/lecturers?lecturerId=78dc4be6-c994-41fb-bdf5-c5283e4a6467)

Зараховано від \_\_\_.\_\_\_.\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

(підпис викладача)

Київ-2022

**Мета:** навчитися аналізувати та класифікувати сучасні інструменти діаграм даних.

**Завдання:**

1. Проаналізувати теоретичні відомості з роботи та пошукові системи на вміст сучасних ERD. Обрати довільно безкоштовний інструментарій ERD для побудови діаграми «сутність зв'язок» згідно узгодженої тематики.
2. Обрати та узгодити тематику комп’ютерного практикуму з викладачем систем баз даних (Додаток А).
3. Дослідити згідно варіанту(номер по списку) сучасні інструменти діаграм даних по критеріям та функціональним можливостям інструментарію.
4. Оформити аналіз даних(інструментарій ERD згідно варіанту) та результати проектування БД (діаграми «сутність зв'язок» з узгодженої тематики) у протокол та завантажити у систему Moodle.

**Порядок виконання роботи**

**Eclipse Papyrus** — це інструмент UML, який надає дуже розширені можливості для налаштування інструменту. На сайті ви знайдете навчальні посібники та приклади, які ілюструють використання всіх цих спеціальних засобів. Eclipse Papyrus надає редактори для всіх діаграм UML(Діаграма класів. Діаграма об'єкта. Діаграма упаковки. Композитна структурна схема. Діаграма компонентів. Схема розгортання. Схема профілю. Діаграма варіантів використання. Діаграма діяльності. Схема кінцевого автомата. Діаграма зв'язку. Діаграма послідовності. Часова діаграма. Діаграма огляду взаємодії).

## **SysML 1.1 і 1.4**

Eclipse Papyrus також забезпечує повну підтримку [SysML](http://www.omgsysml.org/) , щоб увімкнути системну інженерію на основі моделі. Також надаються спеціальні табличні та графічні редактори, необхідні для SysML:

* Діаграма визначення блоку
* Внутрішня блок-схема
* Діаграма вимог
* Параметрична діаграма
* Таблиця вимог
* Таблиця розподілу

## **Модельне виконання**

Завдяки [Moka](http://git.eclipse.org/c/papyrus/org.eclipse.papyrus-moka.git/) Eclipse Papyrus може виконувати моделі за допомогою багатої та розширюваної системи анімації та моделювання.

Крім того, оскільки графічне моделювання не завжди є найкращим способом визначення поведінки виконуваних моделей, Eclipse Papyrus надає видання текстових нотацій із підсвічуванням синтаксису, завершенням і підтримкою вмісту. Звичайно, це настроювана функція Eclipse Papyrus.

## **Повністю настроюване середовище**

Усі функції моделювання Eclipse Papyrus розроблено таким чином, щоб їх можна було налаштовувати та максимізувати повторне використання. Тому, якщо ви захочете адаптувати стандартну конфігурацію для певної області, нотації, практики моделювання або використати потужні механізми налаштування Eclipse Papyrus, щоб адаптувати середовище моделювання відповідно до ваших потреб. Багато конфігурацій в Eclipse Papyrus базуються на моделі, тому налаштування можна виконувати в реальному часі.

* Визначте власне графічне, текстове або табличне позначення.
* Відфільтруйте існуючі палітри або визначте власні за допомогою конфігурації на основі моделі.
* Визначте спеціальні види властивостей, щоб представити лише важливі для вас характеристики.
* Прочитайте свою модель за допомогою спеціального дослідника моделей для структурування та візуалізації.
* Повторно використовуйте стандартні мови або визначте власну мову моделювання завдяки редактору профілю UML.

**Використання мов програмування**

Пряма генерація коду для конкретної мови програмування. Наразі підтримуються C++ і Java, скоро буде підтримуватися C. Генератор коду також підтримує генерацію коду в зворотному та зворотному напрямку.

* [Генерація коду Java](http://wiki.eclipse.org/Java_Code_Generation)
* [Зворотний код Java](http://wiki.eclipse.org/Java_reverse_engineering)
* [Генерація коду C++](http://wiki.eclipse.org/Papyrus/Codegen/Cpp_description)
* Додавання нового генератора коду: окрім вищезгаданої мови програмування, також можна розробити й інтегрувати інші генератори коду до Papyrus. Щоб дізнатися, як додати новий генератор коду до Papyrus, зверніться до такої [вікі](http://wiki.eclipse.org/Papyrus/Codegen/Adding_a_New_Code_Generator) -сторінки .

## **Eclipse Papyrus у технологіях**

Багато технологій доповнюють, розширюють або використовують Papyrus. Ось основні з них:

* [Papyrus RT](https://www.eclipse.org/papyrus-rt/) для моделювання систем реального часу.
* [Papyrus for Robotics](https://www.eclipse.org/papyrus/components/robotics/) : середовище моделювання на основі Papyrus, присвячене робототехніці.
* [Рішення Papyrus PolarSys](https://www.polarsys.org/solutions/papyrus) : Eclipse Papyrus, упаковане як рішення PolarSys.
* [Репозиторій профілів UML Eclipse](http://projects.eclipse.org/projects/modeling.upr) .
* [Papyrus Tools](http://git.eclipse.org/c/papyrus/org.eclipse.papyrus.tools.git/) : набір інструментів, які доповнюють Eclipse Papyrus.
* [Eclipse Safety Framework](https://www.polarsys.org/esf/) : набір інструментів для інтеграції методів безпеки в керований моделлю інженерний процес.

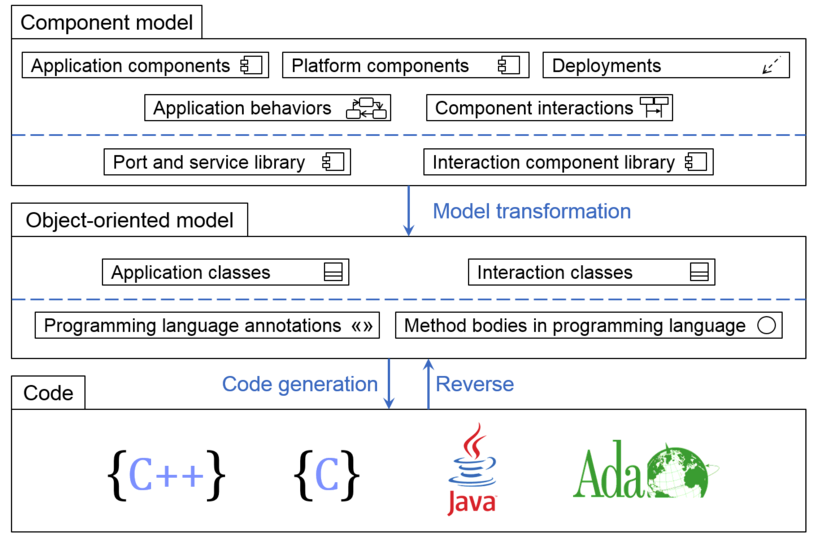


Рис. 1 - Огляд ланцюга інструментів

Тема для створення бази даних: бази даних “Облік побутової техніки на складах”

Різновид використання, варіант використання, прецедент — у розробці програмного забезпечення та системному проектуванні це опис поведінки системи, як вона відповідає на зовнішні запити. Іншими словами, різновид використання описує, «хто» і «що» може зробити з розглянутою системою. Методика різновидів використання застосовується для виявлення вимог до поведінки системи, відомих також як функціональні вимоги. Особливостюю прецедентів є те, що вони завжди ініціюються актором та прецеденти завжди описуються з точки зору акторів

Стислий формат – анотація на один абзац, зазвичай вона описує лише основний успішний сценарій.

Актори – ролі, що виконуються людьми або сутностями, що використовують систему.

**Перелік використаних прецедентів:**

1. Входження користувача у систему
2. Клієнт переглядує каталог товарів
3. Клієнт купує техніку
4. Компанія переглядає вміст складу
5. Компанія продає техніку
6. Компанія редагує каталог
7. Каталог відображає вміст складу
8. Склад зберігає техніку
9. Ордер товару

**Опис** **прецедентів використання в стислому форматі:**

1. Прецедент: Входження користувача у систему
   1. Користувач – Людина

Мета: Входження в систему або створення нового обліку для подальшої роботи з нею

Завдання: Надати особисті дані (ПІБ, пароль, електронна пошта, номер телефону)

1. Прецедент: Клієнт переглядує каталог товарів
   1. Користувач – Клієнт

Мета: Надання можливості клієнту познайомитись з асортиментом складу

Завдання: Показати інформацію про кількість, наявність та ціну товару

1. Прецедент: Клієнт купує техніку
   1. Користувач – Клієнт

Мета: Купівля товару

Завдання: Створення ордеру на побутову техніку (ціни, кількість), що знаходиться на складі

1. Прецедент: Компанія переглядає вміст складу
   1. Користувач – Компанія

Мета: Перегляд кількості побутової техніки на складі

Завдання: Перевірка складу на наявність техніки

1. Прецедент: Компанія продає техніку
   1. Користувач – Компанія

Мета: Продаж техніки зі складу

Завдання: Отримання інформації про побутову техніку та продаж клієнту за ордером

1. Прецедент: Компанія редагує каталог
   1. Користувач – Компанія

Мета: Редагування каталогу товарів

Завдання: Змінення інформації товару, її кількість, наявність та ціну

1. Прецедент: Каталог відображає вміст складу
   1. Користувач – Каталог

Мета: Перегляд складу на наявність товару

Завдання: Перевірка кількості товару на складі

1. Прецедент: Склад зберігає техніку
   1. Користувач – Склад

Мета: Збереження побутової техніки

Завдання: Зберігати інформацію про побутову техніку (назву, тип, виробник, модель, дата)

1. Прецедент: Ордер товару
   1. Користувач – Клієнт

Мета: Продати товар

Завдання: Виставити ордер на продаж товару

**Морфологічний аналіз текстів з виділенням кандидатів сутностей, зв'язків та атрибутів**:

Аналіз іменників:

Людина – сутність, синонім до “користувача”

Аккаунт – атрибут користувача

Компанія – сутність

Клієнт – сутність

Ордер – сутність

Склад – сутність

Побутова техніка – сутність

Каталог - сутність

Ціна – атрибут “Ордер” та “Каталог”

Кількість – атрибут “Склад ” та “Ордер”

Назва - атрибут “Побутовий товар”

Модель - атрибут “Побутовий товар”

Тип - атрибут “Побутовий товар”

Виробник - атрибут “Побутовий товар”

Дата - атрибут “Побутовий товар”

**Дієслова в тексті та їх аналіз:**

Надання людиною особистих даних (для регістрації нового обліку: ПІБ, пароль, електронна пошта, номер телефону; для входження у систему: ПІВ, пароль)

Клієнт переглядає каталог (ціни, кількість)

Компанія продає побутову техніку

Клієнт купує побутову техніку (указує кількість товару, отримує ціну)

Зберігання побутової техніки на складі (кількість товару на складі)

Ордер містить товар (для подальшого продажу клієнту)

Відображення товарів складу

Компанія редагує каталог (змінює параметри товарів: ціна)

Компанія переглядає склад

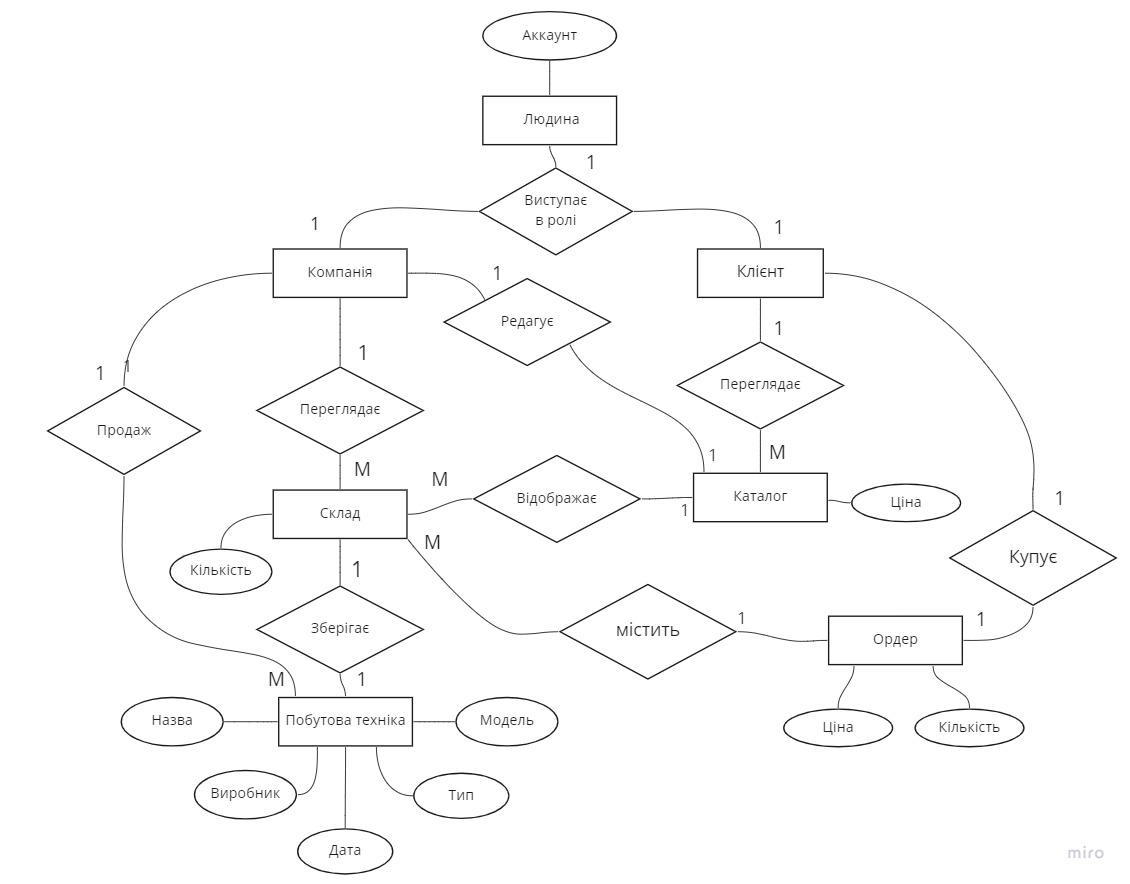


Рис. 2 - Діаграми «сутність зв'язок» бази даних “Облік побутової техніки на складах”