**НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ**

**«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ  
імені ІГОРЯ СІКОРСЬКОГО»**

**Факультет інформатики та обчислювальної техніки**

**Кафедра інформаційних систем та технологій**

|  |  |
| --- | --- |
|  |  |

**Курсова робота**

**з дисципліни «Програмування»**

**на тему: «Електронний каталог товарів»**

Виконав:

студент 1 курсу, групи ІА-32

Варивода Кирил Сергійович \_\_\_\_\_\_\_\_\_\_

(підпис)

Керівник:

асистент кафедри ІСТ

Степанов Андрій Сергійович \_\_\_\_\_\_\_\_\_\_

(підпис)

Засвідчую, що у цій курсовій роботі немає запозичень з праць інших авторів без відповідних посилань. Студент Варивода Кирил Сергійович

(підпис)

Київ – 2024 року

ЗМІСТ

[ВСТУП 3](#_Toc158383261)

[1 ВИМОГИ ДО СИСТЕМИ 4](#_Toc158383262)

[1.1 Функціональні вимоги до системи 4](#_Toc158383263)

[1.2 Нефункціональні вимоги до системи 4](#_Toc158383264)

[2 СЦЕНАРІЇ ВИКОРИСТАННЯ СИСТЕМИ 5](#_Toc158383265)

[2.1 Діаграма прецедентів 6](#_Toc158383266)

[2.2 Опис сценаріїв використання системи 7](#_Toc158383267)

[3 АРХІТЕКТУРА СИСТЕМИ 24](#_Toc158383268)

[4 РЕАЛІЗАЦІЯ КОМПОНЕНТІВ СИСТЕМИ 26](#_Toc158383269)

[4.1 Загальна структура проекту 27](#_Toc158383270)

[4.2 Компоненти рівня доступу до даних 27](#_Toc158383271)

[4.3 Компоненти рівня бізнес-логіки 29](#_Toc158383272)

[4.4 Компоненти рівня інтерфейсу користувача 30](#_Toc158383273)

[ВИСНОВКИ 31](#_Toc158383274)

[ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ 33](#_Toc158383275)

[ДОДАТОК А Лістінг програми 36](#_Toc158383276)

[ДОДАТОК Б Назва додатку Б (за наявності) 37](#_Toc158383277)

[ДОДАТОК В Назва додатку В (за наявності) 38](#_Toc158383278)

# ВСТУП

В еру інформаційних технологій та розвитку технологій стає дедалі більша потреба у сучасних «гаджетах», які будуть корисні для дому. Наприклад, як уявити рутину сьогодення без пральної машини, ноутбуку, вай-фай роутеру, тощо. Вони все далі заполонили нашу буденність і тому стає потреба в місці, де це все можна придбати і тут в нагоді стають сайти з електронним каталогом товарів, які можна придбати не виходячи з дому. Сьогодні, сайти з електронним каталогом товарів є невід’ємною частиною майже кожної людини, яка тісно пов’язана з побутом. Вони допомагають вибирати нові товари для дому, зберігати їх, проглядати та вибирати поміж безлічі варіантів від різних компаній з різних куточків світу. Але задля того, щоб каталог товарів користувався популярністю у споживачів, він повинен мати інтуїтивно зрозумілий інтерфейс, достатній каталог товарів, сортування по категоріям задля легшого пошуку товарів та системи профілів, щоб у користувача була можливість зберігати товари, які йому припали до душі. Також система профілів дозволяє зберігати дані про сесію, такі як: перегляд товарів, пошукові запити та обрані категорії, що в свою чергу дозволяє робити «таргетовану» рекламу, яка в більшості випадків є ефективнішою, оскільки рекомендує користувачу товари з категорій, які він переглядав до цього.

Головна мета роботи є створення застосунку, який дозволить переглядати категорії та товари електронного каталогу, як гість, користувач та адміністратор. Додатково адміністратор може редагувати, видаляти та додавати товари та категорії. Головні вимоги до застосунку є:

* ефективність серверної частини застосунку, щоб користувачу не доводилося чекати на відповідь від «бек енду»;
* додані адміністратором товари мають зберігатися у незалежному від користувача сховищі. Як «незалежне сховище» в даній роботі було використано базу даних яка базується на SQL, в моєму випадку – PostgreSQL;
* інтерфейс повинен мати функціонал для редагування, видалення та створення нових товарів та категорій лише адміністратором.

# 1 ВИМОГИ ДО СИСТЕМИ

## 1.1 Функціональні вимоги до системи

Система має відповідати наступним функціональним вимогам:

– незареєстрований користувач повинен мати можливість переглядати товари в каталозі;

– незареєстрований користувач повинен мати можливість здійснювати сортування товарів по категоріям;

– незареєстрований користувач повинен мати можливість зареєструватися;

– незареєстрований користувач повинен мати можливість здійснити вхід в існуючий акаунт;

– зареєстрований користувач без прав адміністратора повинен мати усі можливості, що є у незареєстрованого користувача, а також він повинен мати можливість переглядати свій профіль, здійснювати вихід з профілю, додавати та видаляти товари з «бажаного»;

– зареєстрований користувач з правами адміністратора повинен мати усі можливості, що є у зареєстрованого користувача без прав адміністратора, а також він повинен мати можливість додавати, редагувати та видаляти категорії та товари.

## 1.2 Нефункціональні вимоги до системи

Система має відповідати наступним функціональним вимогам:

– система повинна мати відкриту архітектуру;

– система повинна мати веб-інтерфейс;

– інтерфейс користувача має бути зручним та інтуїтивно-зрозумілим;

– система повинна бути крос-платформною та адаптуватися під пристрій.

# 2 СЦЕНАРІЇ ВИКОРИСТАННЯ СИСТЕМИ

## 2.1 Діаграма прецедентів

Діаграма прецедентів системи представлена на рис. 2.1.

Акторами є користувачі системи: незареєстрований (гість), зареєстрований без прав адміністратора (користувач) та зареєстрований з правами адміністратора (адміністратор).

Зареєстрованому користувачу без прав адміністратора доступна уся функціональність, що і незареєстрованому, а також можливість вийти з облікового запису, переглянути профіль, додати товар до «збереженого», переглянути список «збережених» товарів та видалення товарів з цього списку, а зареєстрованому користувачу з правами адміністратора додається можливість редагувати та видаляти категорії та товари. Детально усі сценарії використання описані у наступному підрозділі.

# 

Рисунок 2.1 – Діаграма прецедентів

## 2.2 Опис сценаріїв використання системи

Детальні описи сценаріїв використання наведено у таблицях 2.1 – 2.13.

Таблиця 2.1 – Сценарій використання «Створення облікового запису»

|  |  |
| --- | --- |
| Назва | Створення облікового запису |
| ID | 1 |
| Опис | Користувач, використовуючи два поля для вводу, вписує унікальний логін та довільний пароль |
| Актори | Гість |
| Вигоди компанії | Якщо гості зможуть зареєструватися на сайті, то у компанії, яка тримає цей сайт, буде можливість обробляти дані користувача та його сесії |
| Частота користування | При першому відвідуванні сайту |
| Тригери | Користувач вводить логін та пароль і натискає кнопку «Register» |
| Передумови | Поле реєстрації доступне у будь–якому вікні |
| Постумови | Користувач потрапляє на вікно каталогу маючи профіль |
| Основний розвиток | Користувач вводить логін і пароль у поле, натискає на кнопку «Register» |
| Альтернативні розвитки | – |
| Виняткові ситуації | У базі даних існує користувач з заданим логіном |

В таблиці 2.2 представлений сценарій використання «Вхід в обліковий запис»

Таблиця 2.2 – Сценарій використання «Вхід в обліковий запис»

|  |  |
| --- | --- |
| Назва | Вхід в обліковий запис |
| ID | 2 |
| Опис | Користувач, використовуючи два поля для вводу, вписує свій логін і пароль |
| Актори | Гість |
| Вигоди компанії | Якщо є можливість зареєструвати користувача та зберігати його акаунт десь у базі даних, то необхідно надати можливість здійснити вхід в існуючий акаунт |
| Частота користування | При повторних відвідуваннях сайту |
| Тригери | Користувач вводить логін та пароль і натискає кнопку «Log in» |
| Передумови | Поле реєстрації доступне у будь–якому вікні |
| Постумови | Користувач потрапляє на вікно каталогу маючи профіль |
| Основний розвиток | Користувач вводить логін і пароль у поле, натискає на кнопку «Log in» |
| Альтернативні розвитки | \_ |
| Виняткові ситуації | * Не існує користувача з заданим логіном * Невірний пароль для заданого логіну |

В таблиці 2.3 представлений сценарій використання «Вихід з облікового запису»

Таблиця 2.3 – Сценарій використання «Вихід з облікового запису»

|  |  |
| --- | --- |
| Назва | Вихід з облікового запису |
| ID | 3 |
| Опис | Користувач, використовуючи кнопку «Log out» здійснює вихід з облікового запису |
| Актори | Користувач, Адміністратор |
| Вигоди компанії | Користувач має можливість вийти зі свого облікового запису, наприклад, для збереження особистих даних, якщо він здійснив вхід в особистий акаунт не на своєму пристрої |
| Частота користування | Відносно часто |
| Тригери | Користувач натискає кнопку «Log out» |
| Передумови | Користувач ввійшов у свій профіль та знаходиться на будь-якій сторінці сайту |
| Постумови | Користувач потрапляє на вікно каталогу без профілю |
| Основний розвиток | Користувач натискає на кнопку «Log out» |
| Альтернативні розвитки | \_ |
| Виняткові ситуації | \_ |

В таблиці 2.4 представлений сценарій використання «Перегляд головної сторінки каталогу»

Таблиця 2.4 – Сценарій використання «Перегляд головної сторінки каталогу»

|  |  |
| --- | --- |
| Назва | Перегляд головної сторінки каталогу |
| ID | 4 |
| Опис | Користувач переглядає головну сторінку каталогу, де показані товари та категорії |
| Актори | Гість, Користувач, Адміністратор |
| Вигоди компанії | Основна інформація та пропозиції компанії знаходяться на головній сторінці, тому важливо зацікавити клієнта з перший секунд на сайті |
| Частота користування | Постійно |
| Тригери | Користувач переходить на сайт/запускає додаток |
| Передумови | Немає |
| Постумови | Користувач потрапляє на головну сторінку |
| Основний розвиток | Користувач запускає додаток/переходить на сайт |
| Альтернативні розвитки | * Якщо користувач не зареєстрований він не буде мати можливості додати товар до «збережених» * Якщо користувач не має прав адміністратора, він не буде мати можливість редагувати, додавати та видаляти товари та категорії |
| Виняткові ситуації | – |

В таблиці 2.5 представлений сценарій використання «Перехід на сторінку товару»

Таблиця 2.5 – Сценарій використання «Перехід на сторінку товару»

|  |  |
| --- | --- |
| Назва | Перехід на сторінку товару |
| ID | 5 |
| Опис | Користувач зможе докладно побачити інформацію про товар - його назву, ціну, опис та категорію, а також додати у «збережене» прямо на сторінці товару |
| Актори | Гість, Користувач, Адміністратор |
| Вигоди компанії | Якщо користувача зацікавив якийсь товар, то він захоче дізнатися про нього більше інформації та, можливо, придбати його |
| Частота користування | Постійно |
| Тригери | Користувач натискає на товар в каталозі |
| Передумови | Користувач знаходиться на головній сторінці |
| Постумови | Користувач потрапляє на сторінку із пошуковими видачами |
| Основний розвиток | Користувач натискає товар та отримує інформацію про нього. Можливість додати до збережених. Можливість перейти на головну сторінку |
| Альтернативні розвитки | * Якщо у користувача немає акаунту у нього не буде можливості додати товар до «збережених» * Якщо адміністратор натиснув кнопку «Edit», то його перекине на сторінку товару з додатковими полями вводу для зміни даних товару |
| Виняткові ситуації | * Дані, введені адміністратором при зміні товару, не є валідними |

В таблиці 2.6 представлений сценарій використання «Редагування товарів та категорій»

Таблиця 2.6 – Сценарій використання «Редагування товарів та категорій»

|  |  |
| --- | --- |
| Назва | Редагування товарів та категорій |
| ID | 6 |
| Опис | Редагування товарів та категорій |
| Актори | Адміністратор |
| Вигоди компанії | Редагування товарів та категорій є чудовим засобом для адміністрування сайту, бо не стоїть потреба видаляти та створювати нові товари та категорії при зміні актуальності даних |
| Частота користування | Рідко |
| Тригери | Адміністратор натиснув на кнопку «Edit» навпроти товару |
| Передумови | Адміністратора перекидує на сторінку товару, де у нього з’являються нові поля для вводу даних |
| Постумови | Товар чи категорія відредаговані |
| Основний розвиток | Адміністратор вводить дані для редагування та натискає на кнопку «Edit» |
| Альтернативні розвитки | – |
| Виняткові ситуації | - Дані, введені адміністратором при зміні товару чи категорії, не є валідними |

В таблиці 2.7 представлений сценарій використання «Видалення товарів чи категорій»

Таблиця 2.7 – Сценарій використання «Видалення товарів чи категорій»

|  |  |
| --- | --- |
| Назва | Видалення товарів чи категорій |
| ID | 7 |
| Опис | Видалення товарів чи категорій |
| Актори | Адміністратор |
| Вигоди компанії | Для легкого адміністрування адміністратор повинен мати можливість видаляти товари та категорії, щоб не робити постійні запити до бази даних для цього |
| Частота користування | Рідко |
| Тригери | Адміністратор натискає на кнопку «Delete» навпроти товару |
| Передумови | Адміністратор знаходиться на головній сторінці |
| Постумови | Товар чи категорія видалені |
| Основний розвиток | Адміністратор натискає на відповідну кнопку біля товару |
| Альтернативні розвитки | \_ |
| Виняткові ситуації | \_ |

В таблиці 2.8 представлений сценарій використання «Створення товарів та категорій»

Таблиця 2.8 – Сценарій використання «Створення товарів та категорій»

|  |  |
| --- | --- |
| Назва | Створення товарів та категорій |
| ID | 8 |
| Опис | Адміністратор створює товар чи категорію |
| Актори | Адміністратор |
| Вигоди компанії | Функціонал для адміністрування є дуже корисним, наприклад створення категорій та товарів не потребує конкретної взаємодії з базою даних, якщо воно створено на сайті |
| Частота користування | Рідко |
| Тригери | Адміністратор натискає кнопку «Create» |
| Передумови | Адміністратор знаходиться на головній сторінці |
| Постумови | Адміністратора перекидує на сторінку з полями, де він може створити категорію або товар |
| Основний розвиток | Адміністратор створює категорію або товар |
| Альтернативні розвитки | – |
| Виняткові ситуації | * Адміністратор ввів невалідні дані |

В таблиці 2.9 представлений сценарій використання «Перегляд профілю користувача»

Таблиця 2.9 – Сценарій використання «Перегляд профілю користувача»

|  |  |
| --- | --- |
| Назва | Перегляд профілю користувача |
| ID | 9 |
| Опис | Перегляд профілю користувача |
| Актори | Користувач, Адміністратор |
| Вигоди компанії | Користувач повинен мати можливість переглядати дані про свій обліковий запис, також на цій сторінці користувач має можливість подивитися, чи є у нього права адміністратора |
| Частота користування | Відносно часто |
| Тригери | Користувач натискає на кнопку «Профіль» |
| Передумови | Користувач знаходиться на будь-якій сторінці сайту |
| Постумови | Користувача перекидує на сторінку з інформацією про профіль та його «збереженими» товарами |
| Основний розвиток | Користувач натискає на кнопку «Профіль» та отримує інформацію про свій профіль |
| Альтернативні розвитки | \_ |
| Виняткові ситуації | \_ |

В таблиці 2.10 представлений сценарій використання «Зберегти товар»

Таблиця 2.10 – Сценарій використання «Зберегти товар»

|  |  |
| --- | --- |
| Назва | Зберегти товар |
| ID | 10 |
| Опис | Зберегти товар до списку збережених товарів |
| Актори | Користувач, Адміністратор |
| Вигоди компанії | Якщо користувачі зможуть додавати товари до збережених це набагато полегшить їх користування сайтом, а отже збільшить шанси на те, що користувач витратить гроші на цьому сайті |
| Частота користування | Часто |
| Тригери | Користувач натискає на кнопку «\*» біля товару |
| Передумови | Користувач знаходиться на головній сторінці |
| Постумови | Товар додається до списку «збережених». Кнопка «\*» біля товару зникає. Біля кнопки «Профіль» з’являється лічильник товарів, доданих до «збереженого» |
| Основний розвиток | Користувач натискає меню поряд із строкою програвання, відкривається меню із слайдерами налаштування, користувач може їх змінювати |
| Альтернативні розвитки | \_ |
| Виняткові ситуації | \_ |

В таблиці 2.11 представлений сценарій використання «Перегляд «збережених» товарів»

Таблиця 2.11 – Сценарій використання «Перегляд «збережених» товарів»

|  |  |
| --- | --- |
| Назва | Перегляд «збережених» товарів |
| ID | 11 |
| Опис | Користувач переглядає «збережені» товари |
| Актори | Користувач, Адміністратор |
| Вигоди компанії | Якщо є можливість додати товар до «збережених», то має бути можливість переглядати їх |
| Частота користування | Часто |
| Тригери | Користувач натискає на кнопку «Профіль» коли біля неї є лічильник |
| Передумови | Користувач знаходиться на будь-якій сторінці сайту |
| Постумови | Користувача перекидує на сторінку профілю, де видно товари, які він додав до «збережених» |
| Основний розвиток | Користувач натискає на кнопку «Профіль» та отримує список «збережених» товарів |
| Альтернативні розвитки | – |
| Виняткові ситуації | * Користувач не додав жодного товару до «збережених» |

В таблиці 2.12 представлений сценарій використання «Сортування по категоріям»

Таблиця 2.12 – Сценарій використання «Сортування по категоріям»

|  |  |
| --- | --- |
| Назва | Сортування по категоріям |
| ID | 12 |
| Опис | Сортування товарів по категоріям |
| Актори | Гість, Користувач, Адміністратор |
| Вигоди компанії | Якщо користувачі зможуть швидко та зручно сортувати товари по своїм потребам, то це позитивно відобразиться на продажах сайту |
| Частота користування | Часто |
| Тригери | Користувач натискає на категорію, яка його цікавить |
| Передумови | Користувач знаходиться на головній сторінці |
| Постумови | Каталог товарів сортується відповідно заданої категорії |
| Основний розвиток | Користувач натискає на потрібну категорію та сортує товари |
| Альтернативні розвитки | \_ |
| Виняткові ситуації | \_ |

В таблиці 2.13 представлений сценарій використання «Перехід на головну сторінку»

Таблиця 2.13 – Сценарій використання «Перехід на головну сторінку»

|  |  |
| --- | --- |
| Назва | Перехід на головну сторінку |
| ID | 13 |
| Опис | Користувач може перейти на головну сторінку |
| Актори | Гість, Користувач, Адміністратор |
| Вигоди компанії | Користувач повинен мати швидкий доступ до головної сторінки в разі, якщо він захоче продовжити вибирати товари |
| Частота користування | Часто |
| Тригери | Користувач натискає на кнопку «До каталогу» на будь-якій сторінці сайту |
| Передумови | Користувач знаходиться на будь-якій сторінці сайту |
| Постумови | Користувача перекидує на головну сторінку |
| Основний розвиток | Користувач натискає на кнопку «До каталогу» та повертається до перегляду товарів |
| Альтернативні розвитки | \_ |
| Виняткові ситуації | \_ |

# 3 АРХІТЕКТУРА СИСТЕМИ

Об’єм: 1-3 сторінки

Описується загальна архітектура системи, з яких компонентів вона складається, як вони пов’язані, які технології використовуються. Приклад як це приблизно може виглядати. Попередження: в рисунок навмисно внесені помилки, щоб студенти його не «копіпастили».

Загальна архітектура системи наведена на рис. 3.1.

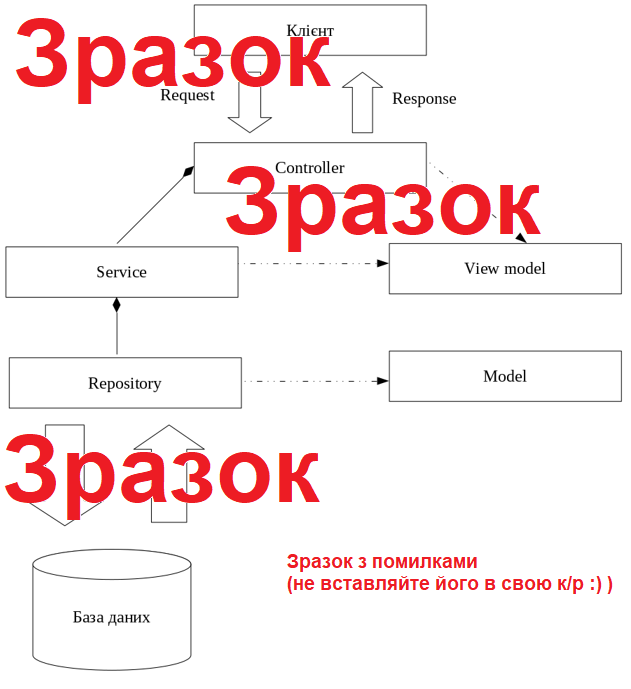


Рисунок 3.1 – Загальна архітектура системи

Система складається з наступних елементів:

– графічний інтерфейс;

– серверна частина;

– база даних.

Графічний інтерфейс необхідний для взаємодії з користувачем. HTTP запит надходить до серверної частини, де оброблюється і повертається відповідь. На серверній частині виконується основна логіка системи. Дані, отриманні з графічного інтерфейсу валідуються, конвертуються. Також, серверна частина формує запит до бази даних та оброблює відповідь і передає її до графічного інтерфейсу. База даних зберігає дані, які були сформовані на серверній частині та повертає їх у разі запиту. Структурна схема зображена в Додатоку Б

До серверної частини належать наступні елементи:

– контролер;

– модель та вигляд;

– сервіс;

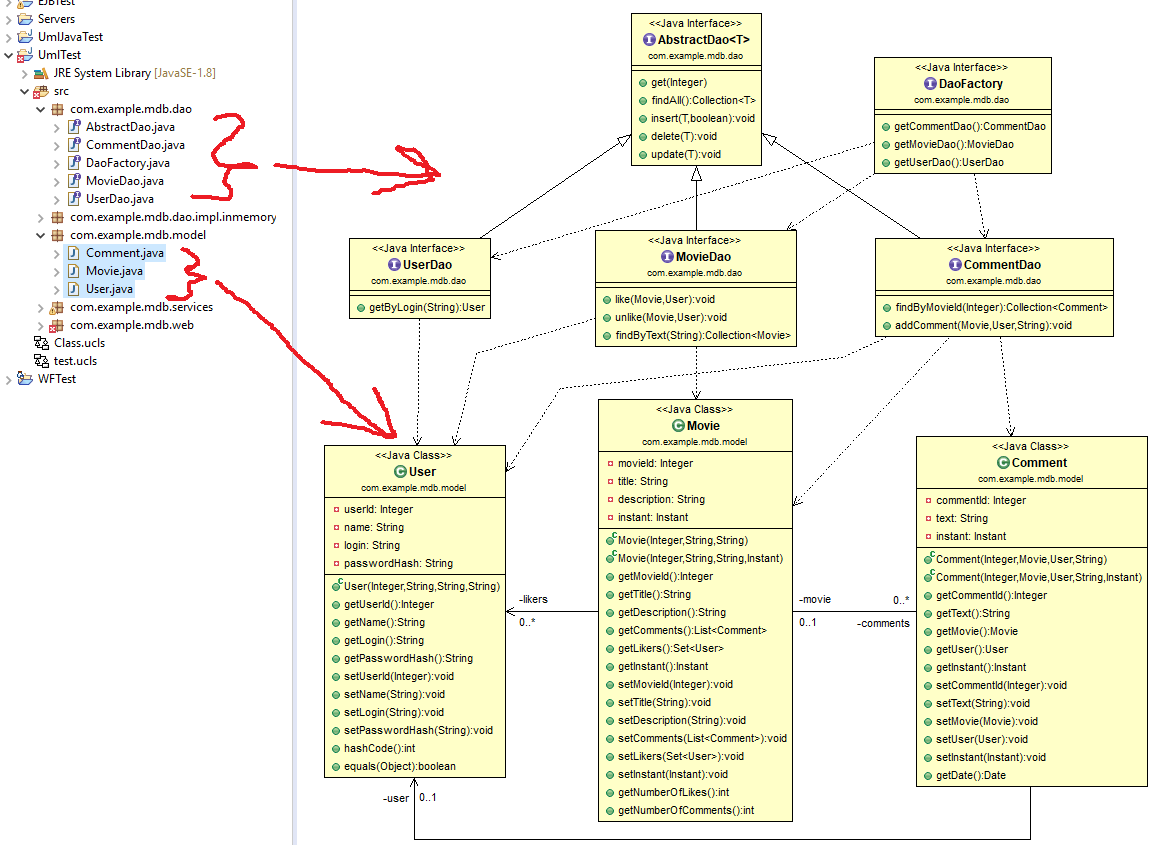
– репозиторій.

На контролер надходять дані з графічного інтерфейсу. З контролеру, дані формуються в сервісі для запиту в репозиторій. З репозиторію дані надсилаються до бази даних і зберігаються. Також в контролері формується вид, тобто об’єкт і його ім’я для відображення на графічному інтерфейсі.

# 4 РЕАЛІЗАЦІЯ КОМПОНЕНТІВ СИСТЕМИ

Обсяг: 10-20 сторінок, включаючи фрагменти коду, рисунки з UML-діаграмами класів або іншими типами діаграм.

Для економії часу рекомендується не малювати діаграми класів «руками», а генерувати їх на основі написаного коду. Наприклад, це вміє робити компонент plug-in компонент ObjectAid для Eclipse (<https://www.objectaid.com/>). Достатньо просто перетягти необхідні елементи з дерева проекту на полотно діаграми, і усі елементи з’являться автоматично:



Рекомендується розбити цей розділ на підрозділи відповідно до архітектури системи

## 4.1 Загальна структура проекту

Загальна структура проекту представлена на рис.4.1

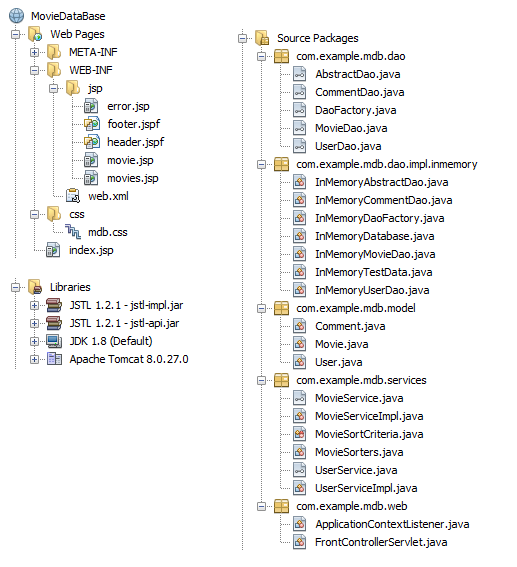


Рисунок 4.1 – Загальна структура проекту

Проект складається з веб-ресурсів, бібліотек, та вихідного коду, який в свою чергу можна поділити на компоненти рівня доступу до даних, компоненти бізнес-логіки та веб-компоненти.

## 4.2 Компоненти рівня доступу до даних

Основні сутності та інтерфейси рівня доступу до даних наведені на рис. 4.2

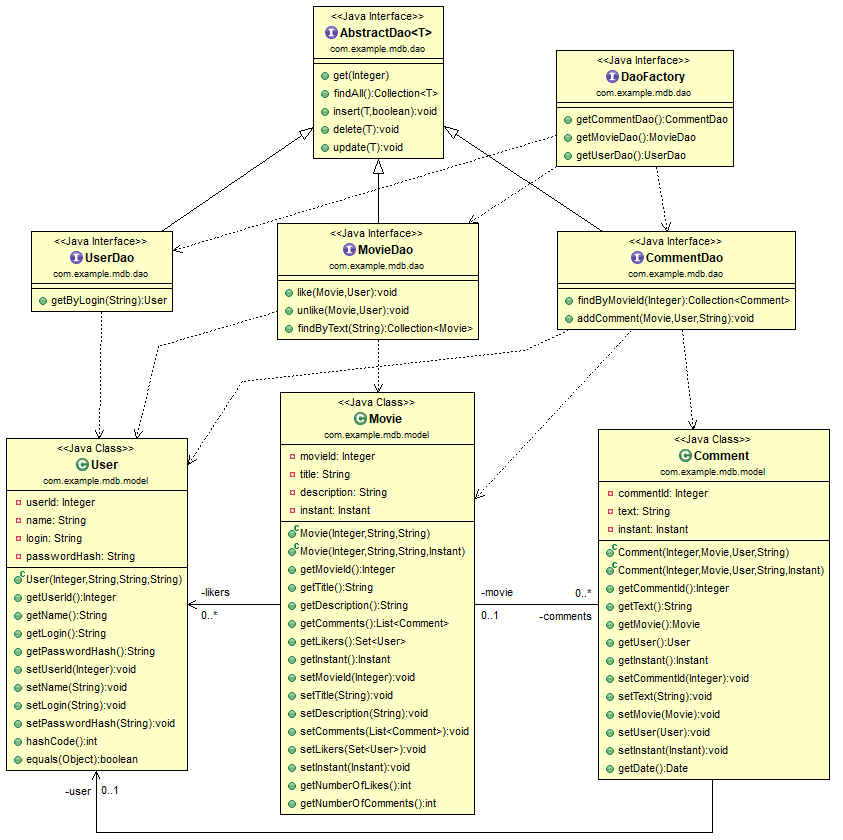


Рисунок 4.2 – Основні сутності та інтерфейси рівня доступу до даних

Далі йде опис сутностей, зв’язків та способів їх використання...

## 4.3 Компоненти рівня бізнес-логіки

## 4.4 Компоненти рівня інтерфейсу користувача

# ВИСНОВКИ

Обсяг: 1-2 сторінки. Коротко підсумовується, що було зроблено в роботі, які цілі були досягнуті, які задачі були вирішені. Наводиться загальна оцінка результату, та робляться пропозиції щодо використання або шляхи подальшого розвитку чи вдосконалення даної системи.

Приклад:

Під час написання роботи, було проаналізовано ринок CRM-систем в Україні, переваги та недоліки існуючих систем. Було вирішено розробити систему, яка дозволить користувачу самостійно керувати своїми телекомунікаційними послугами.

Першим кроком було сформулювати функціональні та нефункціональні вимоги до системи, що визначило очікувану поведінку системи. Наступним, обрано технології на яких буде написана система. Java була обрана в якості мови програмування, середа розробки – IntelliJ IDEA за її можливості та потужність. В якості системи управління базами даних було вирішено використовувати MySQL за легкість її налаштування та підтримки. Для роботи з запитами та налаштуванням з’єднання між системою та базою даних було вирішено використати Hibernate та Spring Data. Величезна перевага фреймворку Spring Data, за яку його було обрано, в можливості не писати запити на SQL, а будувати їх за допомоги назв методів. Для налаштування безпеки системи було вирішено використовувати Spring Security. Цей фреймворк дозволяє власноруч зробити необхідні налаштування, такі як дозволенні ресурси, url-адреси та як саме система визначає ці права доступу. Система повинна мати графічний інтерфейс, тож вирішено було використовувати HTML та Thymeleaf за їх легкість у впровадженні.

Наступним кроком були описані сценарії використання. Система підтримує функціонал перегляду та активації тарифів, сервісів, тарифів домашнього Інтернету. Також була впроваджена бонусна система. Під бонусом мається на увазі – промо-код, активувавши який, користувач може отримати знижку на сторонні продукти. Бонус нараховується у розмірі 10% від вартості купленого продукту на бонусний рахунок користувача. Бонус має термін дії, тож по завершені її – деактивується. Також для симуляції підключення тарифів домашнього Інтернету (проведення кабелів та інше) було описано та впроваджено відповідний сценарій, за яким можна переглянути як змінюється статус по завершенні технічних робіт.

Далі, було розроблено загальну архітектуру системи. Завдяки шаблону проектування MVC, систему було розділено на логічні рівні. Також, окремо було виділені конвертери, помічники та моделі, які роблять систему більш гнучкою та зрозумілою.

Далі, була розроблена ER-модель. Завдяки Hibernate, для встановлення відповідності між базою даних та Java-кодом було створено класи-сутності, що прискорило розробку системи. Для кожної сутності були визначенні необхідні поля та зв’язок з іншими сутностями, якщо в ньому була необхідність.

Отже, сумуючи наведене вище, можна сказати, що основі переваги створеної системи в простоті її налаштування та оновлення функціоналу. Завдяки тому, що система має відкриту архітектуру, її розширення не є проблемою. Також графічний інтерфейс системи є легким в налаштуванні. Система є конкурентоспроможною та має потенціал до розвинення.

# ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ

Обсяг: 1-2 сторінки.

На усі джерела мають бути посилання у тексті

1. J. Stephen Downie. Music information retrieval [Електронний ресурс] – Режим доступу до ресурсу: <http://www.music-ir.org/archive/downie_mir_arist37.pdf>
2. Дискретное преобразование Фурье. [Електронний ресурс] – Режим доступу до ресурсу: <http://ru.dsplib.org/content/dft.html>
3. Мел (высота звука) [Електронний ресурс] – Режим доступу до ресурсу: <https://en.wikipedia.org/wiki/Mel_scale>
4. Pulse-code modulation [Електронний ресурс] – Режим доступу до ресурсу: <https://en.wikipedia.org/wiki/Pulse-code_modulation>
5. Теорема Котельникова [Електронний ресурс] – Режим доступу до ресурсу: <https://en.wikipedia.org/wiki/Nyquist%E2%80%93Shannon_sampling_theorem>
6. Waveform Audio File Format. [Електронний ресурс] – Режим доступу до ресурсу: <https://en.wikipedia.org/wiki/WAV>
7. TIMIT Acoustic-Phonetic Continuous Speech Corpus [Електронний ресурс] – Режим доступу до ресурсу: <https://catalog.ldc.upenn.edu/LDC93S1>
8. CMU\_ARCTIC speech synthesis databases Електронний ресурс] – Режим доступу до ресурсу: <http://festvox.org/cmu_arctic/>
9. Lee et. al. Unsupervised feature learning for audio classification using convolutional deep-belif networks. [Електронний ресурс]. Режим доступу: <https://papers.nips.cc/paper/3674-unsupervised-feature-learning-for-audio-classification-using-convolutional-deep-belief-networks.pdf>
10. Tao Feng. Deep learning for music genre classification. [Електронний ресурс]. Режим доступу: <https://courses.engr.illinois.edu/ece544na/fa2014/Tao_Feng.pdf>
11. Haggblade et. al. Music genre classification. [Електронний ресурс]. Режим доступу: <http://cs229.stanford.edu/proj2011/HaggbladeHongKao-MusicGenreClassification.pdf>
12. George Tzanetakis et. al. Automatic Musical Genre Classification Of Audio Signals [Електронний ресурс]. Режим доступу: http://ismir2001.ismir.net/pdf/tzanetakis.pdf
13. About Last.fm [Електронний ресурс]. Режим доступу: <https://www.last.fm/about>
14. Last.fm – Wikipedia [Електронний ресурс]. Режим доступу <https://en.wikipedia.org/wiki/Last.fm>
15. Spotify – [Електронний ресурс]. Режим доступу: Wikipedia <https://en.wikipedia.org/wiki/Spotify>
16. М. Тім Джонс. Рекомендательные системы: Часть 1. Введение в подходы и алгоритмы [Електронний ресурс]. Режим доступу: <https://www.ibm.com/developerworks/ru/library/os-recommender1/>
17. Clustering
18. Pandora Radio [Електронний ресурс]. Режим доступу: <https://www.pandora.com>
19. How Pandora Radio Works [Електронний ресурс]. Режим доступу: http://computer.howstuffworks.com/internet/basics/pandora2.htm
20. VV Show #54 - Tim Westergren of Pandora [Електронний ресурс]. Режим доступу: <http://www.venturevoice.com/2009/03/vv_show_54_tim_westergren_of_p.html>
21. Mikolov et al. Distributed Representations of Words and Phrases and their Compositionality [Електронний ресурс]. Режим доступу: [http://web2.cs.columbia.edu/~blei/seminar/2016\_discrete\_data/readings/MikolovSutskeverChenCorradoDean2012.pdf](http://web2.cs.columbia.edu/~blei/seminar/2016_discrete_data/readings/MikolovSutskeverChenCorradoDean2013.pdf)
22. Vector Representations of Words [Електронний ресурс]. Режим доступу: <https://www.tensorflow.org/tutorials/word2vec>
23. How does Shazam work [Електронний ресурс]. Режим доступу: <http://coding-geek.com/how-shazam-works/>
24. Li-Chun Wang. An Industrial-Strength Audio Search Algorithm [Електронний ресурс]. Режим доступу: <https://www.ee.columbia.edu/~dpwe/papers/Wang03-shazam.pdf>
25. Kyunghyun Cho, Bart van Merrienboer, Learning Phrase Representations using RNN Encoder-Decoder for Statistical Machine Translation [Електронний ресурс]. Режим доступу: <https://arxiv.org/abs/1406.1078>
26. Yaroslav Yuriiovych Dorogyi, Vasyl Vasylovych Tsurkan, Oleksandr S Khapilin Автоматичне розпізнавання музичних жанрів глибокими згортковими нейронними мережами [Електронний ресурс]. Режим доступу: <http://feltran.kpi.ua/article/view/105159/119678>
27. Aaron van den Oord, Yazhe Li, Oriol Vinyals Representation Learning with Contrastive Predictive Coding [Електронний ресурс]. Режим доступу: <https://arxiv.org/abs/1807.03748>

# ДОДАТОК А Лістінг програми

В першому рядочку – посилання на github

Далі – код системи (інтерфейси, класи, шаблони сторінок, конфігураційні файли, файли з тестовими даними і т.д.)

Враховуючи загальний обсяг коду системи, в додатках допускається використовувати шрифт, що відрізняється від стандартного Times New Roman 14 міжрядковий 1.5 (наприклад, розмір – 10, міжрядковий – одинарний).

# ДОДАТОК Б Назва додатку Б (за наявності)

Вміст додатку Б (за наявності)

В цьому додатку може бути:

* лістинг фрагментів коду;
* скан статті/тез доповіді на конференції/акта впровадження і т.п.;
* довідкові таблиці/рисунки.

# ДОДАТОК В Назва додатку В (за наявності)

Вміст додатку В (за наявності)

В цьому додатку може бути:

* лістинг фрагментів коду;
* скан статті/тез доповіді на конференції/акта впровадження і т.п.;
* довідкові таблиці/рисунки.