**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: «Система проведення тендерів»**

Виконав:

студент 1 курсу, групи ІА-33

Мартинюк Юрій Романович \_\_\_\_\_\_\_\_\_\_

(підпис)

Керівник:

асистент кафедри ІСТ

Мягкий Михайло Юрійович \_\_\_\_\_\_\_\_\_\_

(підпис)

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

Студент \_\_\_\_\_\_\_\_\_\_

(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

У сучасному світі, де ефективність і прозорість бізнес-процесів мають першорядне значення, система проведення тендерів забезпечує справедливу конкуренцію та ефективне використання ресурсів. Система, така як конкурентні торги, дозволяє компаніям або організаціям визначати найбільш відповідних постачальників через об'єктивну та методичну процедуру відбору. Це завдання полягає в тому, щоб зрозуміти, як працюють різні сутності тендерної системи, розглядаючи приклади з участю різних акторів, а саме: Користувача, Тендерної пропозиції, Тендеру тощо.

Тендер є формальним запрошенням, яке запитує пропозиції на постачання певних товарів або послуг. Це зазвичай здійснюється будь-якою організацією, яка потребує таких предметів, щоб потенційні постачальники могли представити свої пропозиції. Отже, це створює основу для конкурентного відбору, надаючи рівні шанси всім зацікавленим сторонам. Крім того, цей процес спрямований на створення конкурентних умов, що дозволяють організації отримати найкращі ціни, а також якісні послуги або товари. Тендерна пропозиція становить відповідь постачальника, що містить повну інформацію про умови та ціни. Ці два елементи дуже важливі, оскільки вони сприяють прозорості та ефективності під час закупівель.

Користувач і Хазяїн тендеру, які є гравцями системи, виконують різні ролі з відповідними правами та обов'язками. Користувачі можуть переглядати активні тендери, шукати їх за ключовими словами, створювати тендерні пропозиції або ставати хазяїнами тендерів. Такі особи здебільшого представляють компанії або організації, які шукають можливості для ведення бізнесу, отримуючи нові контракти після перемоги в торгах. З іншого боку, Хазяїн тендеру може видаляти, переглядати, генерувати URL, запускати, зупиняти, переглядати пропозиції та інше. Тому він/вона не лише керує, а й забезпечує відповідність усіх учасників встановленим вимогам та критеріям відбору.

Ця робота розгляне основні аспекти та труднощі, з якими стикаються учасники та власники тендерів, а також запропонує потенційні шляхи підвищення ефективності та автоматизації процесу проведення тендерів. Зокрема, буде розглянуто питання забезпечення рівних можливостей у бізнес-транзакціях; мінімізації ймовірності корупції при укладанні контрактів; підвищення ефективності у виборі постачальників або підрядників та заощадження витрат. Метою цього дослідження є не лише аналітичний підхід, але й рекомендації заходів, які б підвищили рівень продуктивності, водночас забезпечуючи відкритість під час тендерних процесів.

Основна мета цього проекту – глибоко зрозуміти, як працюють тендерні системи, їх структури та операції, що дозволить оцінити, чи є вони корисними для загальної ефективності організацій. Дослідження має намір запропонувати нові ідеї для вирішення проблем, які виникають у сучасних тендерних системах, краще їх зрозумівши. Досліджуючи різні сценарії, де взаємодіють різні користувачі, я сподіваюся встановити місця, де потрібно внести зміни, щоб наші тендери стали ефективнішими та прозорішими.

Крім того, дослідження також заглибиться в правові та нормативні рамки, що стосуються тендерів, зосереджуючись на дотриманні законів та інших нормативних вимог. Це передбачає аналіз місцевих законів, що регулюють закупівлі, разом із міжнародними стандартами, які керують такими процесами. Додатково буде оцінено досвід різних країн щодо успішного впровадження ефективних тендерних систем, щоб запозичити їхній досвід, зробивши його найкращим для нашого контексту, а також навчитися на їхніх помилках для майбутнього успіху.

Таким чином, курсова робота ставить амбітну мету: не лише дослідити існуючу систему проведення тендерів, але й запропонувати конкретні заходи для її вдосконалення. Це включає рекомендації щодо використання сучасних технологій, таких як блокчейн та штучний інтелект, для підвищення прозорості та ефективності тендерних процесів. Відтак, результати цього дослідження можуть бути корисними не лише для академічного середовища, але й для практичного застосування в бізнесі та державному управлінні.

# 1 ВИМОГИ ДО СИСТЕМИ

## 1.1 Функціональні вимоги до системи

Функціонал системи проведення тендерів повинен бути задовільним для такого переліку дій:

– користувач, який не є зареєстрованим (гість, або відвідувач) повинен мати можливість для перегляду активних тендерів, інформацію про ці тендери та тендерні пропозиції для приблизного оцінювання потреб хазяїна тендеру;

– незареєстрований користувач повинен мати можливість пошуку тендерів за ключовими словами, які вводить цей користувач;

– незареєстрований користувач повинен мати можливість створити власний акаунт для розширення функціоналу, який надає цей сервіс;

– зареєстрований користувач (гість, який створив власний акаунт) повинен мати повний перелік функціоналу, який надається незареєстрованому користувачу, та розширювати його функціонал;

– зареєстрований користувач повинен мати можливість створити власний тендер та стати його хазяїном;

– хазяїн тендеру повинен мати можливість редагування, активації/деактивації, видалення власного тендеру без перешкод, які теоретично можуть виникнути;

– хазяїн тендеру повинен мати можливість власноруч обрати тендерну пропозицію після закінчення дії тендеру, яка підкріплена під цей тендер та в подальшому стане переможцем (виконавцем) цього тендеру;

– зареєстрований користувач повинен мати можливість написати відгук (перелік власних компетенцій) у відповідь на будь-який тендер з подальшим зворотнім зв’язком;

– зареєстрований користувач повинен мати можливість створення тендерної пропозиції до будь-якого тендеру та стати хазяїном пропозиції;

– хазяїн тендерної пропозиції повинен мати можливість редагування та видалення цієї тендерної пропозиції без перешкод;

– між хазяїном тендеру та переможцем цього тендеру повинен бути зворотній зв’язок у вигляді чату, де обидві сторони матимуть можливість обговорити питання стосовно виконання.

## 1.2 Нефункціональні вимоги до системи

Система проведення тендерів повинна відповідати таким нефункціональним вимогам:

– система повинна мати відкриту архітектуру;

– система повинна мати графічний інтерфейс для зручного використання користувачем;

– інтерфейс користувача повинен бути легким у використанні, візуально приємним та мати інтуїтивно зрозумілий функціонал;

– система повинна задовільняти умовам безпеки для користувача, наприклад, хешування паролів, перевірку на те, чи є цей користувач хазяїном тендеру/тендерної пропозиції при редагування чи видаленні, генерації унікальних ідентифікаторів для кожного створеного тендеру, пропозиції та відгуку;

– система повинна бути продуктивною.

# 2 СЦЕНАРІЇ ВИКОРИСТАННЯ СИСТЕМИ

2.1 Діаграма прецедентів

Діаграма прецедентів системи представлена на рис. 2.1.

Акторами даної системи проведення тендерів є користувачі системи: незареєстрований (гість) та зареєстрований користувачі.

Зареєстрованому користувачу доступна уся функціональність, що і незареєстрованому, а також у зареєстрованого користувача повинна бути розширена функціональність: створення/редагування/активація/видалення тендерів (редагування та інше повинно бути доступним за умови, якщо користувач є хазяїном тендеру), створення/видалення тендерних пропозиції, написання відгуків (переліку компетентностей) та вибір хазяїном тендеру переможця власного тендеру.

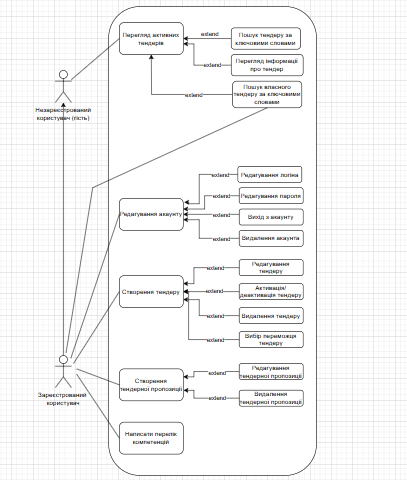


Рисунок 2.1 – Діаграма прецедентів для системи проведення тендерів

## 2.2 Опис сценаріїв використання системи

Детальні описи сценаріїв використання наведено у таблицях 2.1 – 2.9.

Таблиця 2.1 – Сценарій використання «Пошук активних тендерів за ключовими словами»

|  |  |
| --- | --- |
| Назва | Пошук активних тендерів за ключовими словами |
| ID | 1 |
| Опис | Будь-який користувач має можливість власноруч знайти певний тендер, використовуючи ключові слова |
| Актори | Зареєстрований користувач, незареєстрований користувач |
| Вигоди компанії | Пошук тендерів за власними словами надає користувачу можливість знайти тендер з потрібною тематикою, а отже, підвищуються шанси на створення тендерної пропозиції та використання даного сервісу, а це – прибуток для компанії |
| Частота користування | Постійно |
| Тригери | Користувач вводить ключові слова у полі для пошуку |
| Передумови | Пошукове поле доступне на головній сторінці системи |
| Постумови | Користувач потрапляє на сторінку з переліком тендерів, які відповідають даним ключовим словам |
| Основний розвиток | Користувач вводить ключові слова, натискає кнопку «Пошук» біля поля для введення |
| Альтернативні розвитки | – |
| Виняткові ситуації | У разі відсутності тендерів з такими ключовими словами, користувач отримує повідомлення про відсутність тендерів з такими ключовими словами |

В таблиці 2.2 представлений сценарій використання «Перегляд детальної інформації по тендеру»

Таблиця 2.2 – Сценарій використання «Перегляд детальної інформації по тендеру»

|  |  |
| --- | --- |
| Назва | Перегляд детальної інформації по тендеру |
| ID | 2 |
| Опис | Користувач обирає тендер та переглядає детальну інформацію |
| Актори | Зареєстрований користувач, незареєстрований користувач |
| Вигоди компанії | Знання деталей тендеру дає користувачу ширші розуміння суті цього тендеру, а це – утримання клієнтів на цьому сервісі. |
| Частота користування | Постійно |
| Тригери | Користувач натискає на кнопку «Деталі тендеру», яка розташована під кожним тендером |
| Передумови | На головній сторінці або на сторінці знайдених тендерів є список активних тендерів, які можна переглянути |
| Постумови | Користувач бачить детальну інформацію про обраний тендер |
| Основний розвиток | Користувач натискає на кнопку, перед ним з’являється детальна інформація про тендер |
| Альтернативні розвитки | – |
| Виняткові ситуації | – |

В таблиці 2.3 представлений сценарій використання «Перегляд головної сторінки із активними тендерами»

Таблиця 2.3 – Сценарій використання «Перегляд головної сторінки із активними тендерами»

|  |  |
| --- | --- |
| Назва | Перегляд головної сторінки із активними тендерами |
| ID | 3 |
| Опис | Користувач переглядає головну сторінку із активними тендерами |
| Актори | Зареєстрований користувач, незареєстрований користувач |
| Вигоди компанії | Головна сторінка важлива для зацікавлення нових користувачів, та актуальна інформація буде утримувати старих |
| 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 | – |
| Виняткові ситуації | – |

В таблиці 2.4 представлений сценарій використання «Створення/редагування/видалення тендеру»

Таблиця 2.4 – Сценарій використання «Створення/редагування/видалення тендеру»

|  |  |
| --- | --- |
| Назва | Створення/редагування/видалення тендеру |
| ID | 4 |
| Опис | Користувач має можливість створити власний тендер та стати його хазяїном, після створення користувач може рдагувати або видалити тендер |
| Актори | Зареєстрований користувач, хазяїн тендеру |
| Вигоди компанії | Створення тендерів – це суть цієї системи, без неї вона втратить актуальність, а отже, її відсутність не вигідна компанії |
| Частота користування | Часто (коли користувач захоче створити/редагувати/видалити тендер) |
| Тригери | Користувач натискає на відповідні кнопки на головній сторінці та на сторінці користувача |
| Передумови | Користувач знаходиться на головній сторінці або сторінці користувача |
| Постумови | Користувач потрапляє на сторінку для створення/редагування тендеру |
| Основний розвиток | Користувач вводить інформації про власний тендер, після створення потрапляє на сторінку з деталями тендеру. При редагуванні/видаленні все те саме, тільки користувач потрапляє на головну сторінку |
| Альтернативні розвитки | Користувач вирішив не створювати/редагувати тендер, тому переходить на головну сторінку або сторінку користувача |
| Виняткові ситуації | При введенні не того типу даних, користувач отримує повідомлення «Введіть правильний тип даних». Якщо ім’я тендеру уже наявне в базі даних, то користувач отримує відповідне повідомлення |

В таблиці 2.5 представлений сценарій використання «Створення/редагування/видалення тендерної пропозиції»

Таблиця 2.5 – Сценарій використання «Створення/редагування/видалення тендерної пропозиції»

|  |  |
| --- | --- |
| Назва | Створення/редагування/видалення тендерної пропозиції |
| ID | 5 |
| Опис | Користувач має можливість створити пропозиції до певного тендеру, після – редагувати/видалити її |
| Актори | Зареєстрований користувач, автор пропозиції |
| Вигоди компанії | Це також суть цієї системи, без неї – система не є актуальною, тому за її відсутності у компанії не буде прибутків |
| Частота користування | Користувачі зазвичай помірно створюють пропозиції |
| Тригери | Користувач натискає кнопку «Створити тендерну пропозицію» або «Редагувати пропозицію» |
| Передумови | Користувач знаходиться на головній сторінці, де під кожним тендером є кнопка для створення пропозиції. Для редагування – на сторінці користувача |
| Постумови | Користувач потрапляє на сторінку для створення пропозиції |
| Основний розвиток | Користувач створює/редагує/видаляє власну пропозицію |
| Альтернативні розвитки | Користувач передумав створювати/редагувати/видаляти пропозицію, тому може перейти на головну сторінку |
| Виняткові ситуації | При введення не того типу даних, користувач отримує повідомлення «Введіть правильний тип даних» |

В таблиці 2.6 представлений сценарій використання «Написання відгуку (переліку компетенцій) хазяїну тендеру»

Таблиця 2.6 – Сценарій використання «Написання відгуку (переліку компетенцій) хазяїну тендеру»

|  |  |
| --- | --- |
| Назва | Написання відгуку (переліку компетенцій) хазяїну тендеру |
| ID | 6 |
| Опис | Користувач має можливість (за потреби) написати перелік компетенцій хазяїну тендеру |
| Актори | Зареєстрований користувач |
| Вигоди компанії | Надавання зворотного зв’язку збільшує шанси на успішну угоду, а отже – прибуток для компанії |
| Частота користування | Не часто |
| Тригери | Користувач натискає на відповідну кнопку під описом тендеру |
| Передумови | Користувач знаходиться на головній сторінці або на сторінці з детальною інформацією про тендер |
| Постумови | Користувач потрапляє на сторінку для написання переліку компетенцій |
| Основний розвиток | Користувач натискає відповідну кнопку та створює відгук (перелік компетенцій) |
| Альтернативні розвитки | Користувач передумав писати відгук, тому він може перейти на головну сторінку |
| Виняткові ситуації | При введення не того типу даних, користувач отримує повідомлення «Введіть правильний тип даних» |

В таблиці 2.7 представлений сценарій використання «Вибір хазяїном тендеру переможця (виконавця) свого тендеру»

Таблиця 2.7 – Сценарій використання «Вибір хазяїном тендеру переможця (виконавця) свого тендеру»

|  |  |
| --- | --- |
| Назва | Вибір хазяїном тендеру переможця (виконавця) свого тендеру |
| ID | 7 |
| Опис | Після завершення часу тендеру, хазяїн цього тендеру може обрати переможця (виконавця) цього тендеру |
| Актори | Зареєстрований користувач (хазяїн тендеру) |
| Вигоди компанії | Вибір переможця тендерної пропозиції – важливий функціонал даної системи, тому без неї вона втрачає сенс, а отже – компанія не отримає прибутки |
| Частота користування | Один раз на один тендер |
| Тригери | Користувач натискає відповідну кнопку під своїм тендером на сторінці користувача |
| Передумови | Користувач (хазяїн тендеру) знаходиться на сторінці користувача |
| Постумови | Користувач (хазяїн тендеру) обирає переможця зі списку тендерних пропозиції, які відсортовані за прайсом |
| Основний розвиток | Користувач обирає переможця свого тендеру |
| Альтернативні розвитки | Користувач не хоче вибирати переможця з певних причин, пому він може вернутися назад |
| Виняткові ситуації | – |

В таблиці 2.8 представлений сценарій використання «Пошук власних тендерів за ключовими словами»

Таблиця 2.8 – Сценарій використання «Пошук власних тендерів за ключовими словами»

|  |  |
| --- | --- |
| Назва | Пошук власних тендерів за ключовими словами |
| ID | 8 |
| Опис | Користувач має можливість знайти власний тендер за ключовими словами |
| Актори | Зареєстрований користувач |
| Вигоди компанії | Пошук власних тендерів надає зручні умови для користувачів, а отже, більше шансів утримати користувача на цьому сервісі |
| Частота користування | Нечасто |
| Тригери | Користувач вводить ключові слова у поле для пошуку та натискає відповідну кнопку |
| Передумови | Користувач знаходиться на сторінці користувача |
| Постумови | Користувач отримує список власних тендерів з такими ключовими словами |
| Основний розвиток | Користувач вводить ключові слова у відповідному полі та натискає кнопку «Пошук» |
| Альтернативні розвитки | – |
| Виняткові ситуації | При відсутності тендерів з такими ключовими словами, користувач отримує відповідне повідомлення |

В таблиці 2.9 представлений сценарій використання «Зворотній зв’язок між хазяїном тендеру та виконавцем»

Таблиця 2.9 – Сценарій використання «Зворотній зв’язок між хазяїном тендеру та виконавцем»

|  |  |
| --- | --- |
| Назва | Зворотній зв’язок між хазяїном тендеру та виконавцем |
| ID | 9 |
| Опис | Хазяїн тендеру та переможець цього тендеру мають можливість для зворотного зв’язку один з одним |
| Актори | Хазяїн тендеру, переможець цього тендерцу |
| Вигоди компанії | Більше шансів на успішне виконання тендеру, а отже – утримання цих користувачів, у результаті – прибуток для компанії |
| Частота користування | Часто, враховуючи те, що це дія між вдома користувачами |
| Тригери | Користувач натискає відповідну кнопку під власним тендером, в якого уже визначено переможця, або тендерною пропозицією, яка є переможцем тендеру |
| Передумови | Користувач знаходиться на сторінці користувача |
| Постумови | Зв’язок між хазяїном тендеру та виконавцем |
| Основний розвиток | Користувач потрапляє в чат |
| Альтернативні розвитки | – |
| Виняткові ситуації | – |

# 3 АРХІТЕКТУРА СИСТЕМИ

Загальна архітектура системи наведена на рис. 3.1.

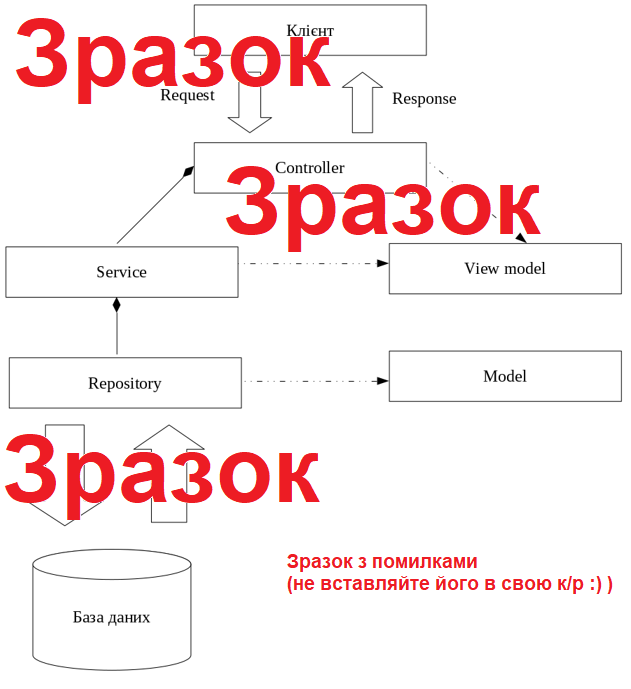


Рисунок 3.1 – Загальна архітектура системи

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;

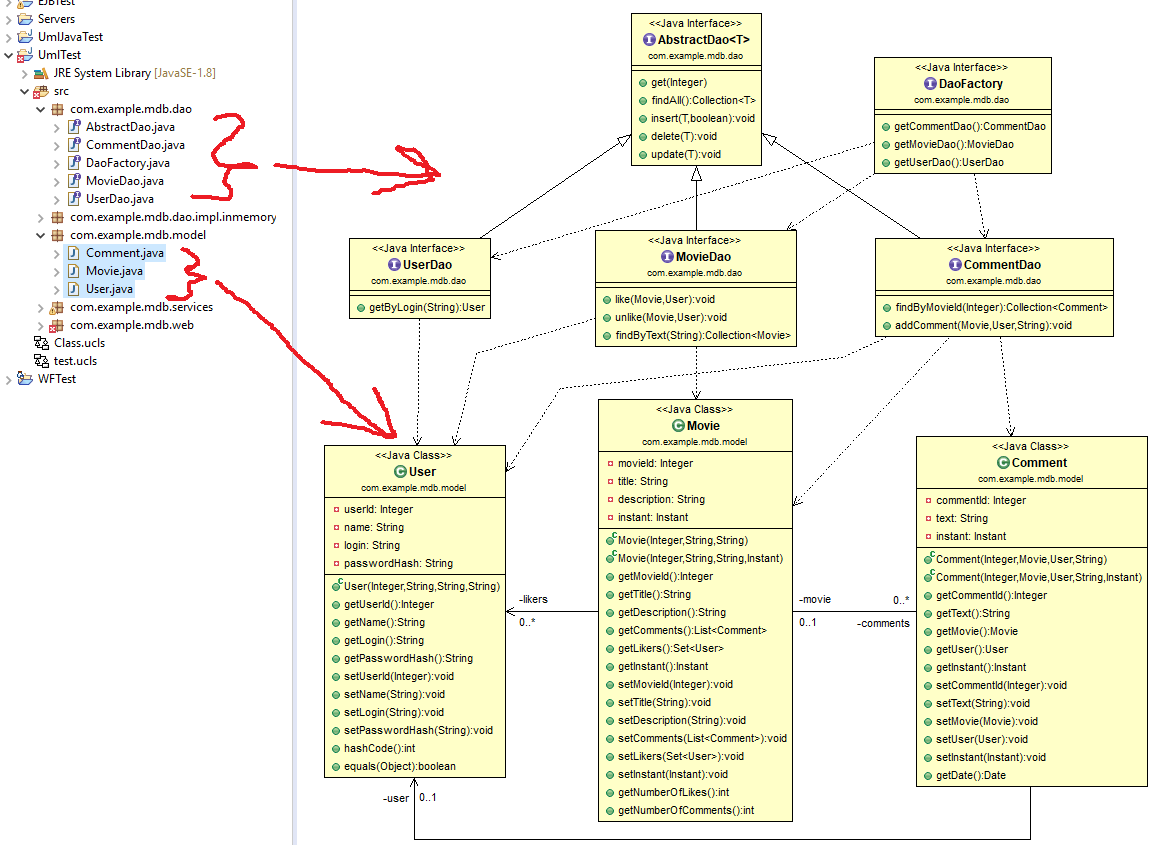
– репозиторій.

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

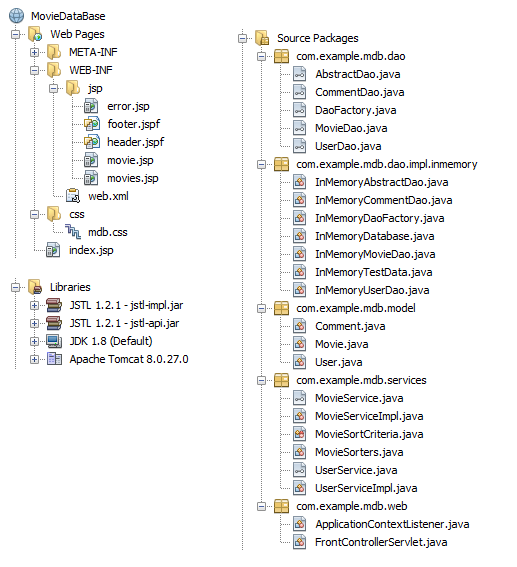


Рисунок 4.1 – Загальна структура проекту

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

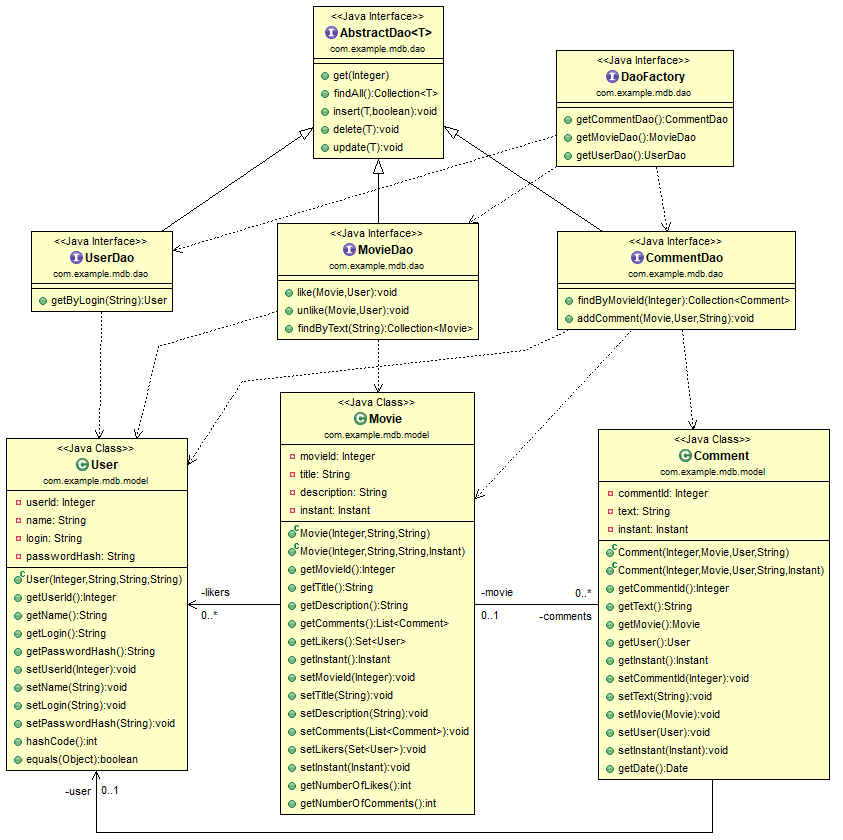


Рисунок 4.2 – Основні сутності та інтерфейси рівня доступу до даних

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

[masonabor/CourseWork\_Martyniuk\_Yurii\_IA-33: Курсова робота (github.com)](https://github.com/masonabor/CourseWork_Martyniuk_Yurii_IA-33)

# 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.