**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: «Сайт новин»**

Виконав:

студент 1 курсу, групи ІА-33

Панащук Роман Андрійович \_\_\_\_\_\_\_\_\_\_

(підпис)

Керівник:

асистент кафедри ІСТ

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

(підпис)

Засвідчую, що у цій магістерській дисертації 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

Сьогодні, в епоху цифрових технологій та глобалізації, інформація стала невід'ємною частиною нашого повсякденного життя. Інтернет перетворився на головний канал отримання новин, завдяки якому мільйони людей дізнаються про найважливіші події у світі буквально в реальному часі. Чи можемо ми уявити день без актуальних новин про політику, спорт, економіку чи культуру? Навряд чи. Тому створення сучасного, швидкого і зручного новинного сайту – це не просто технічне завдання, а відповідь на нагальні потреби суспільства.

Актуальність цього проєкту очевидна: ми живемо в світі, де своєчасний доступ до новин є критично важливим. Люди хочуть бути в курсі останніх подій, приймати обґрунтовані рішення та залишатися поінформованими. Проте, у потоці інформації легко загубитися. Саме тому створення надійного та зручного новинного ресурсу набуває особливого значення. Він не лише допоможе користувачам швидко знаходити потрібну інформацію, але й стане їхнім надійним компасом у світі новин.

Наша мета – розробити інноваційний новинний сайт, який стане зручним помічником для всіх, хто цінує оперативність та достовірність інформації. Цей сайт повинен забезпечити користувачам швидкий доступ до новин, можливість сортувати їх за категоріями та шукати за ключовими словами. Водночас адміністратори отримають ефективний інструмент для створення та управління новинами.

Для досягнення цієї амбітної мети ми поставили перед собою кілька важливих задач:

1. Створення захоплюючого користувацького інтерфейсу
2. Розробка бази даних для зберігань новин
3. Реалізація функціоналу для користувачів
4. Реалізація функціоналу для адміністраторів
5. Забезпечення безпеки та захисту даних

Створення цього новинного сайту – це не просто технічне завдання. Це проєкт, який може змінити спосіб, у який люди сприймають і отримують інформацію. Це виклик, на який ми готові відповісти з ентузіазмом та відданістю. Наш сайт стане вашим надійним партнером у світі новин, допомагаючи завжди залишатися на хвилі найактуальніших подій.

# 1 ВИМОГИ ДО СИСТЕМИ

## 1.1 Функціональні вимоги до системи

Система має відповідати наступним функціональним вимогам:

– незареєстрований користувач повинен мати можливість переглядати усі новини, які зараз доступні, приблизну дату їх публікації та їх категорії;

– незареєстрований користувач повинен мати можливість здійснювати пошук новин по ключовим словам у заголовку та/або у описі фільму та здійснювати пошук за кожною з категорій;

– адміністратор повинен мати можливість входу у адміністративну панель для створення нових новин

– адміністратор повинен мати можливість створювати нові новини, вказуючи заголовок, текст та категорію

– незареєстрований користувач та адміністратор повинні мати змогу легко переміщуватись по сайту, повинні бути кнопки повернення назад

## 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.

– система повинна забезпечувати високий рівень безпеки даних

– система повинна бути здатною до масштабування

– система повинна бути надійною та стійкою до збоїв

# 2 СЦЕНАРІЇ ВИКОРИСТАННЯ СИСТЕМИ

## 2.1 Діаграма прецедентів

Діаграма прецедентів системи представлена на рис. 2.1.

Акторами є користувачі системи: незареєстрований (користувач) та зареєстрований користувач (адміністратор).

Адміністратору доступна уся функціональність, що і користувачу, а також можливість створювати новини та входити у панель адміністратора. Детально усі сценарії використання описані у наступному підрозділі.

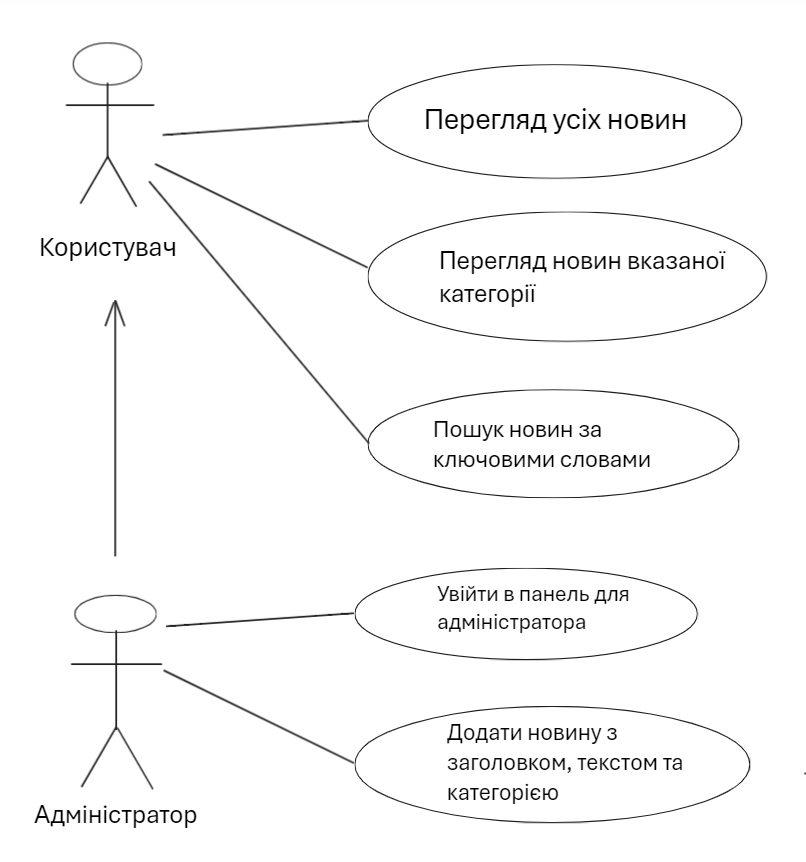


Рисунок 2.1 – Діаграма прецедентів

## 2.2 Опис сценаріїв використання системи

Детальні описи сценаріїв використання наведено у таблицях 2.1 – 2.17.

Таблиця 2.1 – Сценарій використання «Перегляд усіх новин»

|  |  |
| --- | --- |
| Назва | Перегляд усіх новин |
| ID | 1 |
| Опис | Користувач, використовуючи кнопку на головній сторінці сайту переглядає усі новини які були опубліковані |
| Актори | Користувач |
| Вигоди компанії | Якщо користувачі не можуть переглядати усі новини що є на сайті, то вони навряд чи продовжать користуватися сервісом |
| Частота користування | Постійно |
| Тригери | Користувач натискає кнопку на головній сторінці сайту |
| Передумови | Кнопка “Усі новини” доступна на головній сторінці сайту |
| Постумови | Користувач потрапляє на вікно з усіма новинами сайту |
| Основний розвиток | Користувач натискає на кнопку “Усі новини” |
| Альтернативні розвитки | – |
| Виняткові ситуації | – |

В таблиці 2.2 представлений сценарій використання «Перегляд новин вказаної категорії»

Таблиця 2.2 – Сценарій використання «Перегляд новин вказаної категорії»

|  |  |
| --- | --- |
| Назва | Перегляд новин вказаної категорії |
| ID | 2 |
| Опис | Користувач обирає категорію, новини за якою хоче переглянути і переглядає |
| Актори | Користувач |
| Вигоди компанії | Якщо користувач не зможе переглянути всі новини за категорією яка його цікавить, то сайт новин не буде користуватися попитом |
| Частота користування | Постійно |
| Тригери | Користувач вибирає категорію у полі для пошуку |
| Передумови | Пошукове поле доступне на головній сторінці сайту |
| Постумови | Користувач потрапляє на нову сторінку та переглядає новини вказаної категорії |
| Основний розвиток | Користувач вибирає категорію у блоці для пошуку та натискає на кнопку пошуку |
| Альтернативні розвитки | – |
| Виняткові ситуації | Якщо новин не знайдено то користувач побачить напис, що новини за вказаною категорією відсутні |

В таблиці 2.3 представлений сценарій використання «Пошук новин за ключовими словами»

Таблиця 2.3 – Сценарій використання «Пошук новин за ключовими словами»

|  |  |
| --- | --- |
| Назва | Пошук новин за ключовими словами |
| ID | 3 |
| Опис | Користувач, використовуючи поле для пошуку, шукає новини, що відповідають критерію пошуку у заголовках та тексті |
| Актори | Користувач |
| Вигоди компанії | Якщо користувачі не можуть шукати цікаві їм новини, то вони навряд чи продовжать користуватися сервісом |
| Частота користування | Постійно |
| Тригери | Користувач вводить пошуковий запит у полі для пошуку |
| Передумови | Пошукове поле доступне на головній сторінці сайту |
| Постумови | Користувач потрапляє на сторінку з результатами пошуку |
| Основний розвиток | Користувач вводить запит у пошукову строку, натискає на кнопку пошуку |
| Альтернативні розвитки | – |
| Виняткові ситуації | Якщо новин не знайдено то користувач побачить напис, що новини за вказаними критеріями пошуку відсутні |

В таблиці 2.4 представлений сценарій використання «Вхід у панель адміністратора»

Таблиця 2.4 – Сценарій використання «Вхід у панель адміністратора»

|  |  |
| --- | --- |
| Назва | Вхід у панель адміністратора |
| ID | 4 |
| Опис | Адміністратор може зайти у панель адміністратора для додання новин на сайт |
| Актори | Адміністратор |
| Вигоди компанії | Для дотримання безпеки, адміністратор повинен пройти верифікацію, щоб не було стороннього втручання у роботу сайту |
| Частота користування | Часто |
| Тригери | Адміністратор вказує свій логін та пароль і натискає кнопку входу |
| Передумови | Користувач переходить на сторінку входу |
| Постумови | Адміністратор потрапляє на сторінку панелі адміністратора |
| Основний розвиток | Адміністратор натискає на кнопку у шапці сайту для переходу на сторінку авторизації, вказує свій логін та пароль і натискає на кнопку входу |
| Альтернативні розвитки | – |
| Виняткові ситуації | Користувач натискає на кнопку у шапці сайту для переходу на сторінку авторизації, вказує неправильний логін та/або пароль і натискає на кнопку входу. Після цього потрапляє на сторінку де бачить що його логін або пароль не вірні |

В таблиці 2.5 представлений сценарій використання «Створення новини»

Таблиця 2.5 – Сценарій використання «Створення новин»

|  |  |
| --- | --- |
| Назва | Створення новин |
| ID | 5 |
| Опис | Адміністратор має можливість додати нову новину на сайт, для цього він повинен вказати її заголовок текст та категорію |
| Актори | Адміністратор |
| Вигоди компанії | Для підтримання активності сайту, потрібно часто завантажувати нові новини на сайт |
| Частота користування | Користувачи зазвичай досить часто використовують плейлисти |
| Тригери | Адміністратор натиснув кнопку для створення новини |
| Передумови | Адміністратор увійшов у панель адміністратора та заповнив необхідні поля новини |
| Постумови | Адміністратор потрапляє на сторінку де бачить що новина успішно опублікована та може повернутися на головну сторінку або повернутися у панель адміністратора та додати ще одну новину |
| Основний розвиток | У сторінці панелі адміністратора адміністратор вказує заголовок, категорію та текст новини і натискає кнопку опублікувати новину |
| Альтернативні розвитки | – |
| Виняткові ситуації | – |

# 3 АРХІТЕКТУРА СИСТЕМИ

Загальна архітектура системи наведена на рис. 3.1.

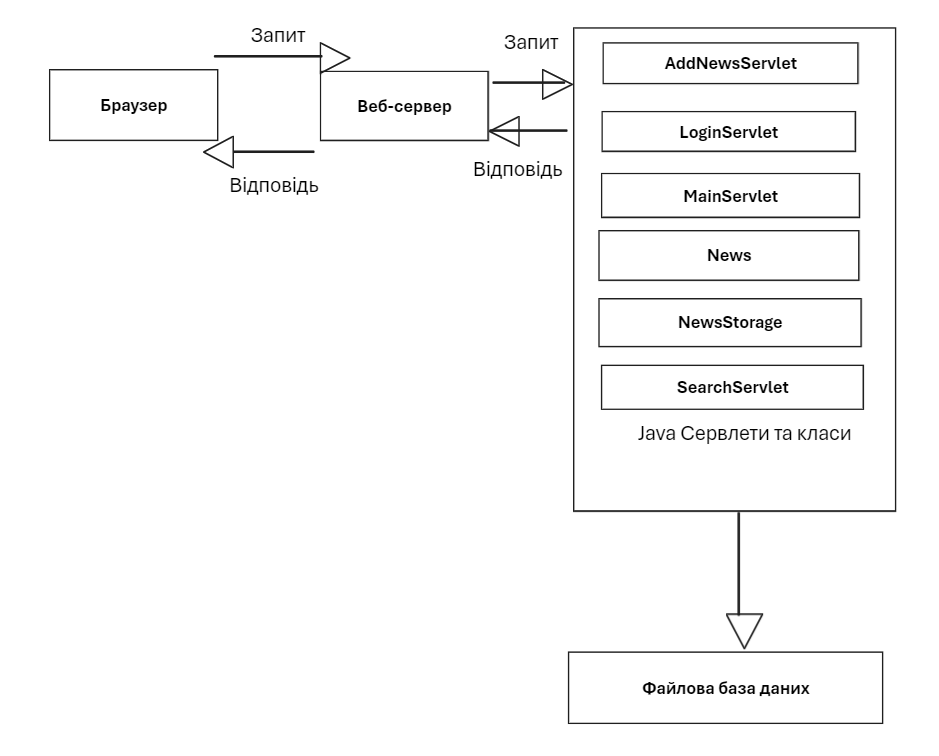


Рисунок 3.1 – Загальна архітектура системи

Система складається з наступних елементів:

– графічний інтерфейс;

– серверна частина;

– база даних.

Графічний інтерфейс необхідний для взаємодії з користувачем. 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 файлової бази даних та оброблює відповідь і передає її до графічного інтерфейсу. Файлова база даних зберігає дані, які були сформовані на серверній частині та повертає їх у разі запиту.

До серверної частини належать наступні елементи:

– Веб сервер;

– Java сервлети;

– Класи логіки додатку;

На веб сервер надходять дані з графічного інтерфейсу. З веб серверу, дані обробляються у сервлетах. Сервлети надсилають запити у файлову базу даних і створюють нові файли з новою інформацією. Також в сервлеті формується вид, тобто об’єкт і його ім’я для відображення на графічному інтерфейсі.

# 4 РЕАЛІЗАЦІЯ КОМПОНЕНТІВ СИСТЕМИ

4.1 Загальна структура проекту

Загальна структура проекту представлена на рис.4.1

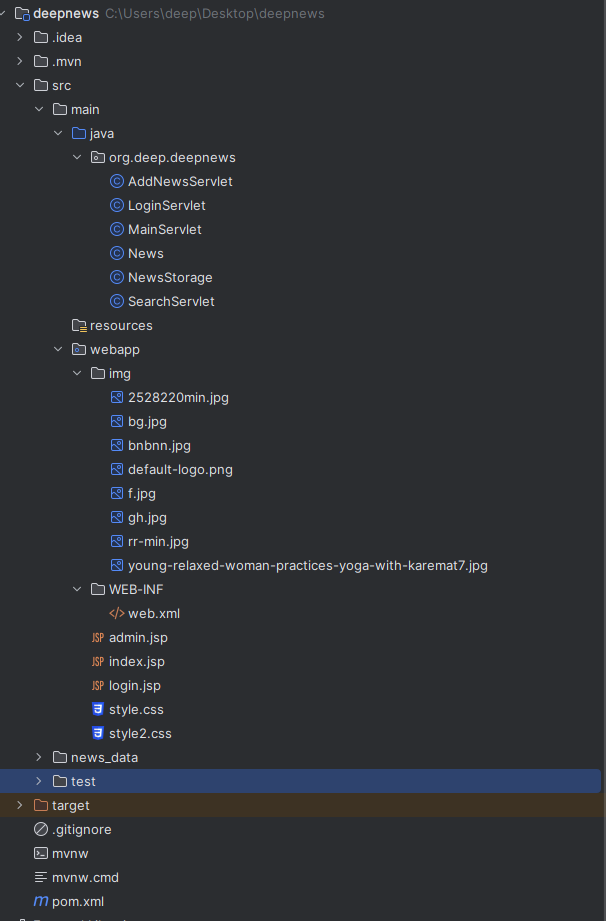


Рисунок 4.1 – Загальна структура проекту

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

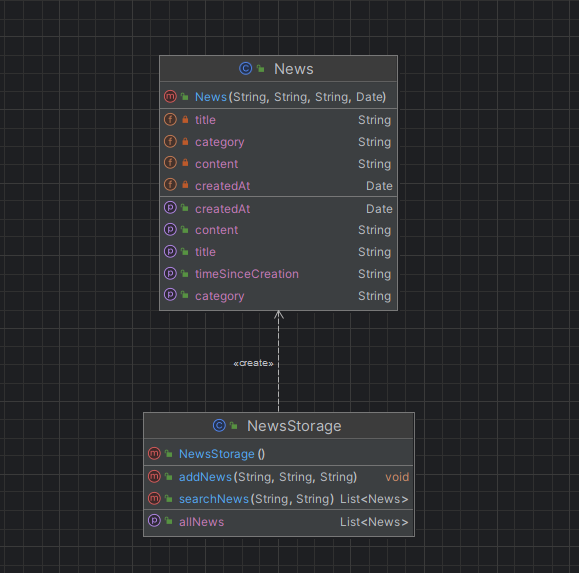


Рисунок 4.2 – Основні сутності та інтерфейси рівня доступу до даних

## Сутності:

## News

## Це клас, який представляє новину. Має такі поля:

## title (String) — заголовок новини

## content (String) — вміст новини

## category (String) — категорія новини

## createdAt (Date) — дата створення новини

## NewsStorage

## Це клас для роботи з новинами на рівні зберігання даних. Має такі методи:

## getAllNews() — повертає список всіх новин, зчитаних з файлів

## searchNews(String keywords, String category) — повертає список новин, що відповідають заданим ключовим словам та категорії

## addNews(String title, String content, String category) — додає нову новину, зберігаючи її у файл

1. Папка з усіма новинами news\_data яка виступає у ролі файлової бази даних

## Зв’язки:

## News та NewsStorage

## NewsStorage використовує клас News для створення об'єктів новин.

## Метод getAllNews() зчитує файли і створює об'єкти News.

## Метод searchNews() працює з колекцією об'єктів News.

## Метод addNews() приймає параметри, з яких створює новий об'єкт News і зберігає його у файл.

## Способи використання:

## Зчитування всіх новин

## Метод getAllNews() читає всі файли в директорії NEWS\_DIRECTORY, створює об'єкти News і повертає список цих об'єктів.

## Пошук новин

## Метод searchNews(String keywords, String category) проходить через всі новини, отримані від getAllNews(), і відбирає ті, що відповідають ключовим словам та категорії.

## Додавання новин

## Метод addNews(String title, String content, String category) створює файл, записує в нього дані новини (заголовок, категорія, дата створення, вміст).

## 4.3 Компоненти рівня бізнес-логіки

Основні сутності та інтерфейси рівня бізнес-логіки на рис. 4.3

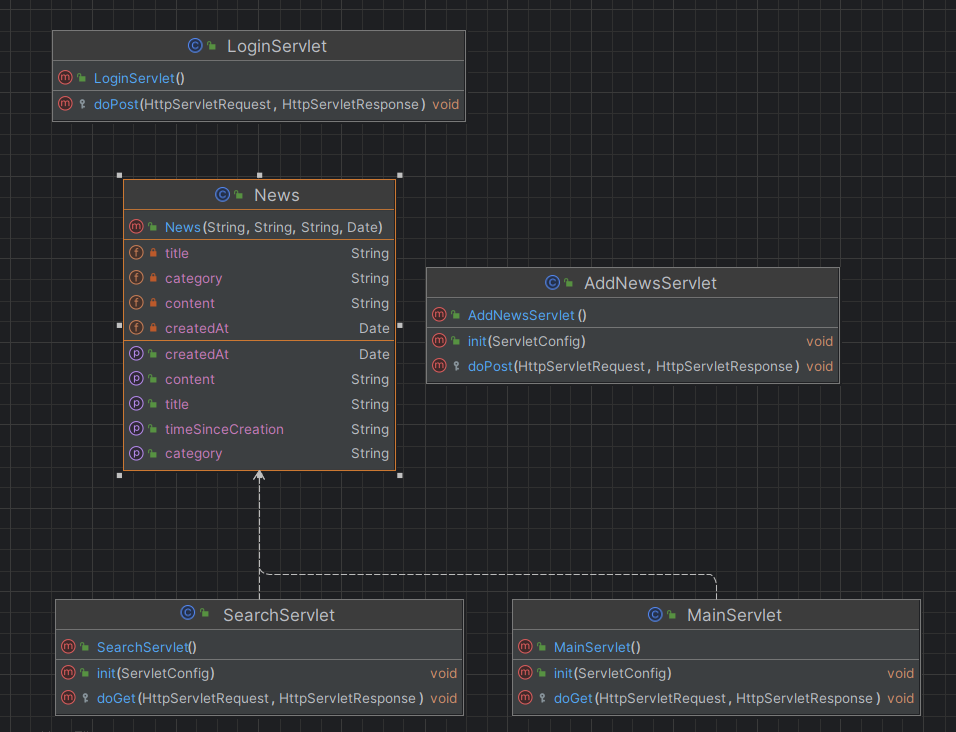


Рисунок 4.3 – Основні сутності та інтерфейси рівня бізнес-логіки

Сутності:

1. AddNewsServlet

Цей сервлет обробляє запити на додавання новин.

1. LoginServlet

Цей сервлет обробляє аутентифікацію користувачів.

1. MainServlet

Цей сервлет відповідає за відображення новин. Він отримує список новин з бізнес-логіки та передає їх на фронтенд.

1. SearchServlet

Цей сервлет обробляє запити на пошук новин.

Зв’язки:

1. Сервлети та NewsStorage

Сервлети викликають методи з NewsStorage для виконання операцій над новинами.

AddNewsServlet використовує метод addNews для додавання новин.

SearchServlet використовує метод searchNews для пошуку новин.

MainServlet використовує метод getAllNews для отримання всіх новин.

1. Сервлети та Користувачі

LoginServlet взаємодіє з файловою базою даних користувачів для аутентифікації.

Способи використання:

1. Додавання новин

AddNewsServlet отримує дані від користувача (через форму), і передає дані в NewsStorage для збереження.

2.Аутентифікація користувачів

LoginServlet отримує дані для входу, використовує бізнес-логіку для перевірки користувача (порівняння на правильність написання логінів іпаролів), і встановлює сесію для аутентифікованих користувачів.

1. Пошук новин

SearchServlet отримує ключові слова і категорії, використовує метод searchNews з NewsStorage для пошуку відповідних новин і передає результати на фронтенд.

1. Відображення сторінки з новинами

MainServlet отримує всі новини за допомогою методу getAllNews з NewsStorage і передає їх на фронтенд для відображення.

## Компоненти рівня інтерфейсу користувача

Основні сутності та інтерфейси рівня бізнес-логіки на рис. 4.4

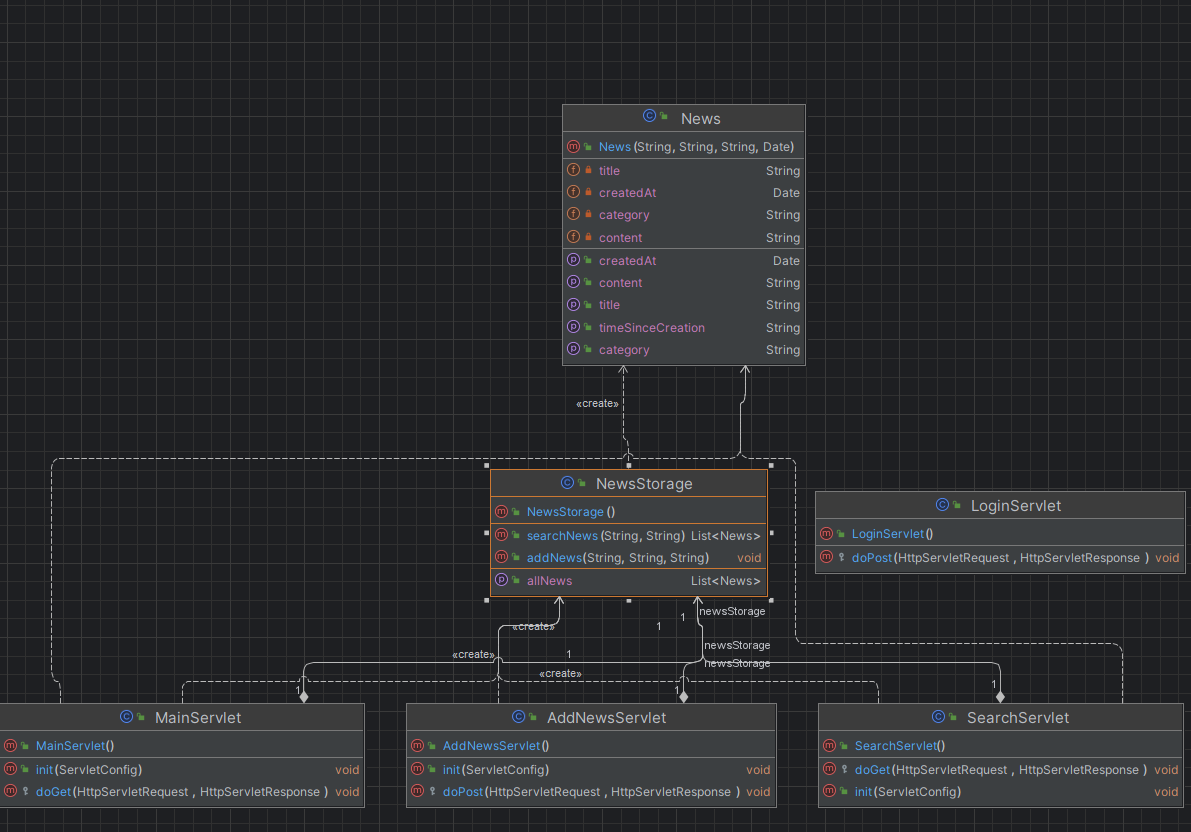


Рисунок 4.4 – Основні сутності та інтерфейси рівня інтерфейсу користувача

Сутності:

1. JSP сторінки

Використовуються для відображення інформації користувачу і збору даних від нього.

1. Сервлети

Взаємодіють з JSP сторінками для обробки запитів від користувача і відображення відповідей.

Зв’язки:

1. Сервлети та JSP

2. Сервлети передають дані на JSP сторінки

Способи використання:

1. Додавання новин

AddNewsServlet обробляє дані, введені у форму, додає новину і перенаправляє користувача на головну сторінку.

1. Аутентифікація користувачів

login.jsp відображає форму для входу.

LoginServlet обробляє введені дані, перевіряє їх і перенаправляє користувача на головну сторінку або повертає його на сторінку входу у разі невдачі.

1. Пошук новин

SearchServlet обробляє введені дані, виконує пошук новин.

1. Відображення сторінки з усіма новинами

MainServlet отримує всі новини і відображає їх.

# 

# ВИСНОВКИ

У процесі виконання курсової роботи створено сучасний сайт новин, що відповідає викликам цифрової епохи і задовольняє потреби користувачів у швидкому та зручному доступі до найсвіжіших новин. Проєкт став живим прикладом того, як новітні технології можуть змінити спосіб сприйняття та отримання інформації.

У результаті нашої роботи були досягнуті всі поставлені цілі:

1. Захоплюючий користувацький інтерфейс: розроблено інтерфейс, який не лише зручний, але й візуально привабливий.
2. Міцна структура даних: Створено надійну файлову базу даних, яка забезпечує ефективне зберігання та обробку всіх новин, категорій і дат публікацій.
3. Інтуїтивно зрозумілий функціонал для користувачів: Користувачі можуть переглядати всі новини та шукати за ключовими словами і категоріями.
4. Потужний інструмент для адміністраторів: Адміністратори мають можливість легко створювати новини, що дозволяє оперативно оновлювати контент і тримати користувачів в курсі найсвіжіших подій.

Таким чином, створено не просто сайт новин, а справжній інформаційний хаб, який допомагає користувачам бути в курсі найактуальніших подій. 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

<https://www.freepik.com/> - сайт з зображеннями які були використані в проєкті

<https://nicepage.com/> - сайт з якого була скопійована стилі для адаптивності сайту

# ДОДАТОК А Лістінг програми

Github - https://github.com/SpringlesMcOutrage/kr

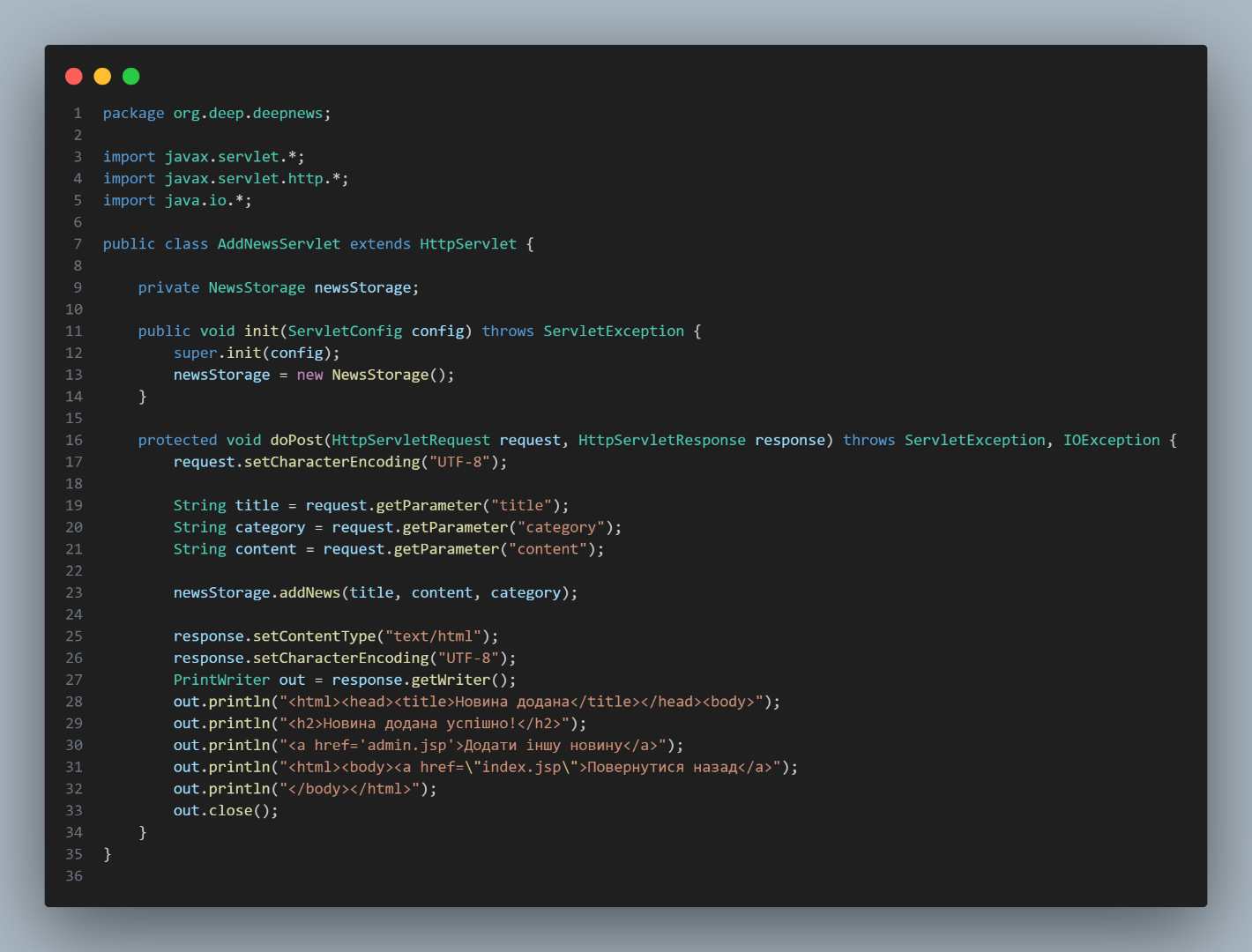


Рисунок 1 – файл AddNewsServlet.java



Рисунок 2 – файл LoginServlet.java

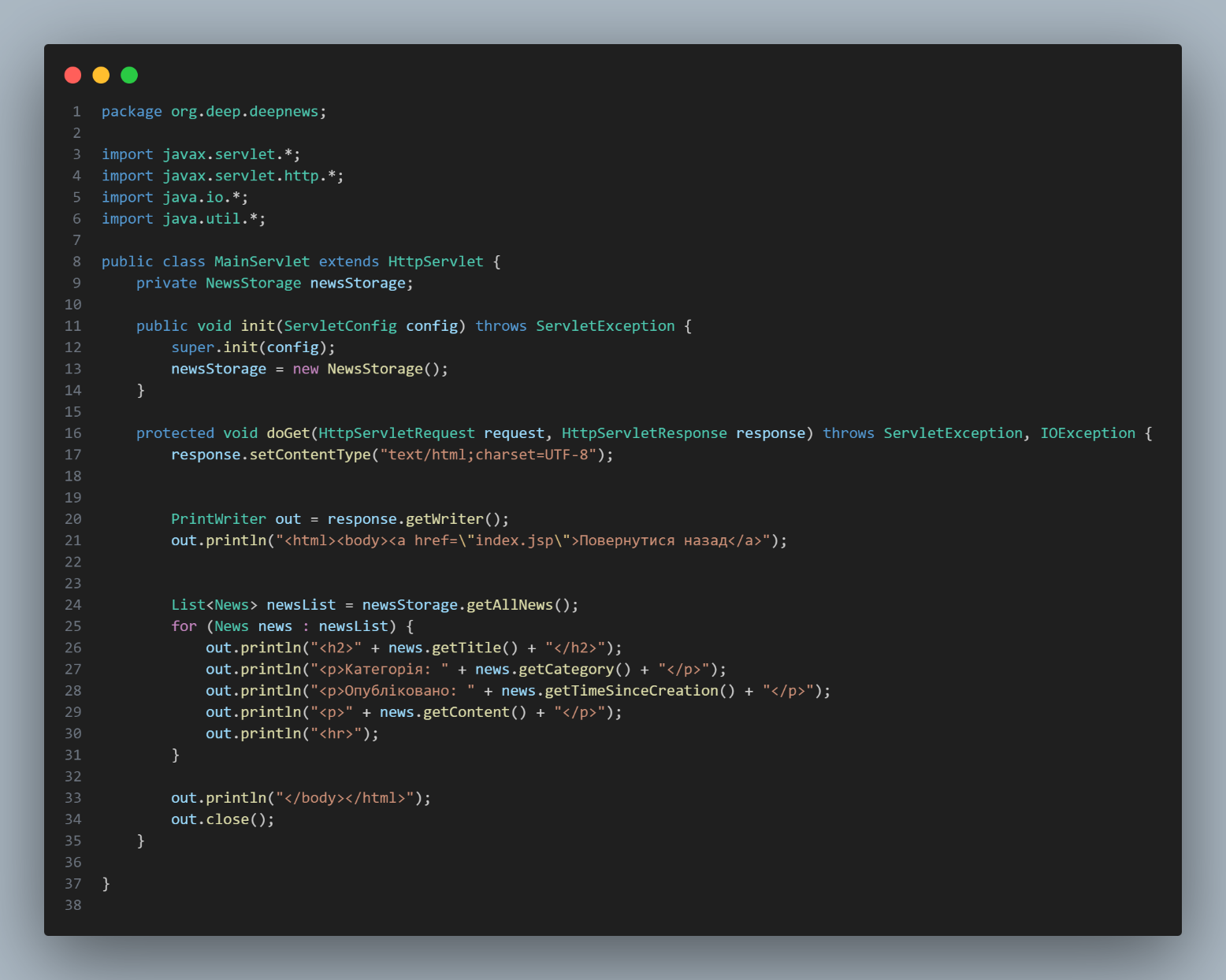


Рисунок 3 – файл MainServlet.java



Рисунок 4 – файл News.java



Рисунок 5 – файл NewsStorage.java



Рисунок 6 – файл SearchServlet.java

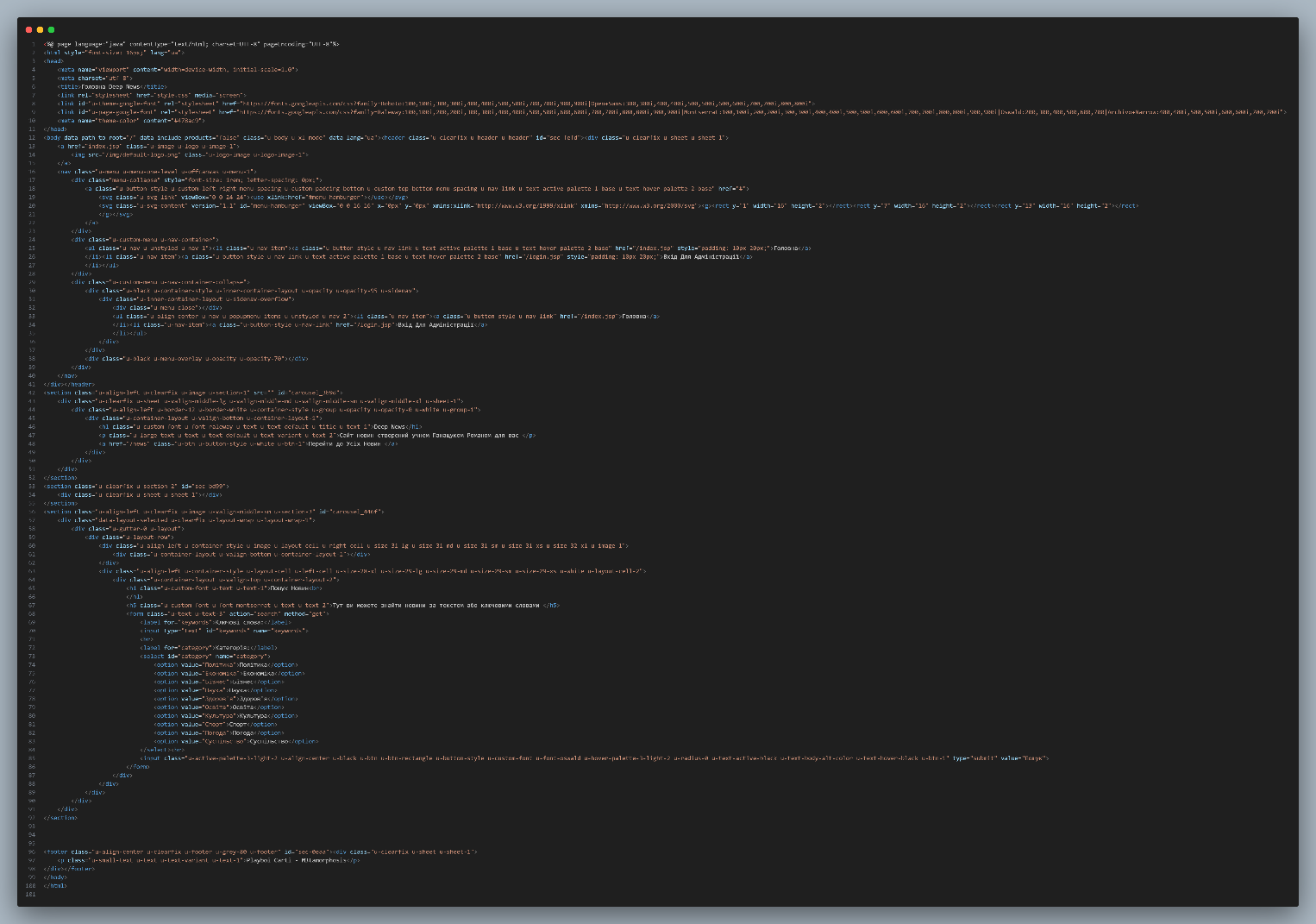


Рисунок 7 – файл index.jsp

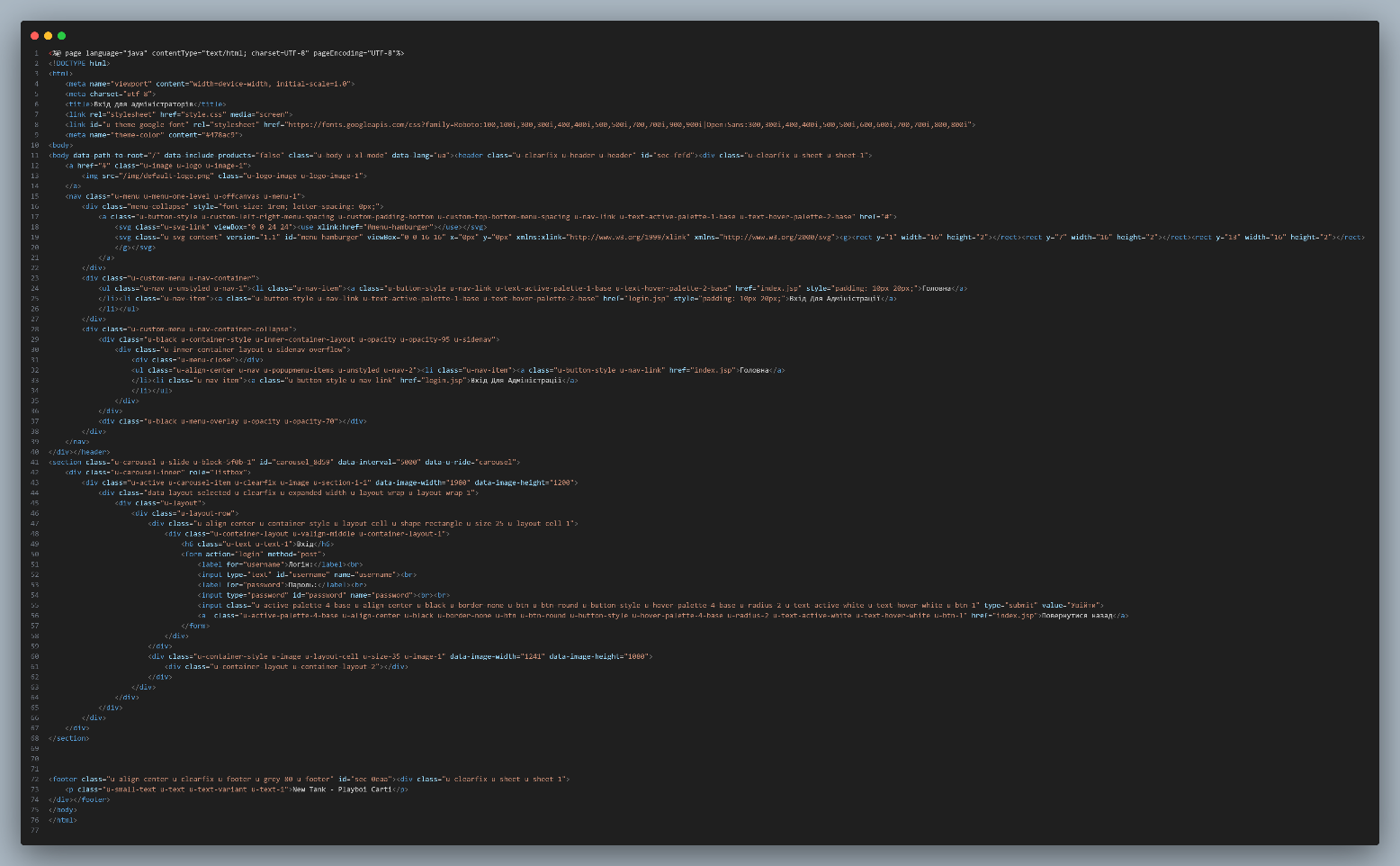


Рисунок 8 – файл login.jsp

Рисунок 9 – файл admin.jsp

Рисунок 10 – файл web.xml