

ОДЕСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ ІМЕНІ І. І. МЕЧНИКОВА

(повне найменування закладу вищої освіти)

Факультет математики, фізики та інформаційних технологій

(повне найменування факультету)

Кафедра інформаційних технологій

(повна назва кафедри)

Кваліфікаційна робота

на здобуття ступеня вищої освіти «Бакалавр»

«Розробка інформаційної системи класифікації та зберігання 3D-моделей»

(тема кваліфікаційної роботи українською мовою)

«Development of an information system for classifying and storing 3D models»

(тема кваліфікаційної роботи англійською мовою)

Виконав: здобувач денної форми навчання спеціальності 122 Комп'ютерні науки

(код, назва спеціальності)

Освітня програма Комп'ютерні науки

(назва)

САЙКІН Олексій Сергійович

(прізвище, ім'я, по-батькові здобувача)

Керівник викл. Гадяцький І. А.

(науковий ступінь, вчене звання, прізвище, ініціали)

(підпис)

Рецензент Департамент інформації та цифрових рішень Одеської міської ради, головний спеціаліст відділу впровадження інформаційних технологій, Клепатська В.В.

(науковий ступінь, вчене звання, прізвище, ініціали)

Рекомендовано до захисту:
Протокол засідання кафедри
Інформаційних технологій

№ від 2025 р.

Завідувачка кафедри

КАЗАКОВА Надія

(підпис)

(прізвище, ім'я)

Захищено на засіданні ЕК №
протокол № від 2025 р.

Оцінка / /
(за національною шкалою/шкалою ECTS/ бали)

Голова ЕК

КОПИЧЕНКО Іван

(підпис)

(прізвище, ім'я)

Одеса 2025

АНОТАЦІЯ

У дипломній роботі розроблено інформаційну систему для класифікації та зберігання 3D-моделей, яка дозволяє структурувати, систематизувати та забезпечити зручний доступ до цифрового тривимірного контенту.

Об'єктом дослідження є процеси зберігання та класифікації 3D-моделей у цифрових середовищах. Предметом дослідження – методи й інструменти створення інформаційної системи для організації, пошуку та доступу до 3D-моделей.

Метою роботи є створення інформаційної системи, яка дозволяє ефективно організувати колекцію 3D-моделей із можливістю завантаження, категоризації, пошуку та збереженню додаткових метаданих (опис, формат, полігони, теги тощо).

У роботі проведено аналіз існуючих рішень у сфері зберігання 3D-контенту, обґрунтовано вибір технологій реалізації, спроектовано архітектуру програмного забезпечення, розроблено базу даних і реалізовано прототип системи на основі веб-технологій, як бекенду так і фронтенду (Node.js, PostgreSQL, HTML/CSS/JS).

Результатом є функціональний веб-сервіс для зберігання 3D-моделей, який може бути використаний як для особистих цілей, так і в якості основи для подальшої розробки професійної платформи для поширення тривимірного контенту.

ABSTRACT

This thesis presents the development of an information system for the classification and storage of 3D models, which enables structuring, systematizing, and providing convenient access to digital three-dimensional content.

The object of the research is the processes of storing and classifying 3D models in digital environments. The subject of the research is the methods and tools used to create an information system for organizing, searching, and accessing 3D models.

The aim of the work is to create an information system that efficiently organizes a collection of 3D models, with support for uploading, categorization, search, and storing additional metadata (description, format, polygon count, tags, etc.).

The thesis includes an analysis of existing solutions in the field of 3D content storage, a justification for the choice of implementation technologies, the design of the software architecture, the development of a database, and the implementation of a prototype system based on web technologies, both backend and frontend (Node.js, PostgreSQL, HTML/CSS/JS).

The result is a functional web service for storing 3D models, which can be used for personal purposes or serve as a foundation for developing a professional platform for sharing three-dimensional content.

ЗМІСТ

ПЕРЕЛІК СКОРОЧЕНЬ, УМОВНИХ ПОЗНАЧЕНЬ І ТЕРМІНІВ	4
ВСТУП	6
1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ.....	7
1.1 Поняття інформаційної системи.....	7
1.2 Актуальність створення.....	9
1.3 Аналіз існуючих рішень	12
1.4 Постановка завдання.....	14
2 ПРОЄКТУВАННЯ СИСТЕМИ.....	16
2.1 Платформа для розробки.....	16
2.2 Використані веб-технології.....	20
2.3 Бази даних	21
2.4 Проєктування бізнес-процесів	26
3 ПРОГРАМНА РЕАЛІЗАЦІЯ СИСТЕМИ	29
3.1 Загальна характеристика реалізованої ІС	29
3.2. Структура інформаційної системи	30
3.3 Модулі та компоненти сайту	31
3.3.1 Головна сторінка	32
3.3.2 Каталог моделей	35
3.3.3 Персональна сторінка моделі.....	36
3.3.4 Інші сторінки та функції.....	37
3.4 Модуль адміністратора.....	37
3.5 Тестування інформаційної системи	40
ВИСНОВКИ.....	42
ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ.....	43

ПЕРЕЛІК СКОРОЧЕНЬ, УМОВНИХ ПОЗНАЧЕНЬ І ТЕРМІНІВ

- ІС – Інформаційна система (англ. Information system);
- ПЗ – Програмне забезпечення;
- 3D – Тривимірна комп'ютерна графіка (англ. Three Dimensional);
- 3D-model – Тривимірна модель або об'єкт у тривимірному просторі;
- SQL – Structured Query Language;
- CSS – Cascading Style Sheets;
- HTML – HyperText Markup Language;
- HTTP – HyperText Transfer Protocol;
- HTTPS – HTTP Secure (Захищений протокол HTTP);
- NPM – Node Package Manager;
- JS – JavaScript;
- JSON – JavaScript Object Notation;
- URL – Uniform Resource Locator;
- IP – Internet Protocol;
- REST – Representational State Transfer;
- IDE – Integrated Development Environment;
- СУБД, СКБД, DBMS – Система управління або керування базами даних (англ. Database Management System);
- BPMS – Business Process Management Systems;
- BPMN – Business Process Model and Notation;
- Physical Data Model – Фізична модель;
- Conceptual Data Model – Концептуальна модель;
- Logical Data Model – Логічна модель;
- БД – Бази даних;

SEO – Search Engine Optimization;

API – Application Programming Interface (Інтерфейс прикладного програмування);

UI/UX – User interface/User experience;

REST – Representational State Transfer;

CRUD – Create, Read, Update, Delete;

MVC – Model-View-Controller;

CDN – Content Delivery Network;

XML – Extensible Markup Language.

ВСТУП

У сучасному світі тривимірна графіка займає важливе місце у багатьох сферах діяльності – від комп'ютерних ігор і анімації до архітектурної візуалізації, віртуальної та доповненої реальності. Збільшення попиту на тривимірні ресурси призводить до необхідності створення зручних платформ, які дозволяють зберігати, впорядковувати та надавати доступ до моделей, текстур, сцен та інших цифрових об'єктів.

Існуючі рішення не завжди відповідають потребам конкретних спільнот або спеціалізованих задач. У зв'язку з цим актуальним є створення гнучкої інформаційної системи, яка дозволяє не лише завантажувати та зберігати 3D-моделі, але й класифікувати їх за категоріями, технічними характеристиками та іншими параметрами. Така система повинна забезпечити простий і швидкий доступ як для кінцевих користувачів, так і для адміністраторів, із можливістю подальшого масштабування, редагування та аналізу контенту.

Об'єктом дослідження є процеси створення, зберігання та класифікації цифрових тривимірних моделей в контексті веб-орієнтованому середовищі.

Предметом дослідження є архітектура, методи та інструменти розробки інформаційної системи, яка забезпечує завантаження, збереження, фільтрацію та організацію файлів моделей, а також механізм керування доступом до них.

Метою роботи являється розробка інформаційної системи для класифікації та зберігання 3D-моделей, що забезпечує користувачам можливість зручного доступу до цифрових тривимірних ресурсів, їх фільтрації та завантаженню.

1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ

1.1 Поняття інформаційної системи

Інформаційна система – це організована сукупність ресурсів (людей, технічних засобів, програмного забезпечення, баз даних та процедур), яка призначена для збирання, обробки, зберігання, передавання та використання інформації для підтримки прийняття рішень, координації, аналізу й контролю в організації або окремій сфері діяльності [1].

Інформаційні системи відіграють ключову роль у цифровому середовищі. Вони забезпечують ефективну обробку великих обсягів даних, автоматизацію бізнес-процесів, управління інформацією, надання доступу до ресурсів тощо. Залежно від сфери застосування, ІС можуть бути бухгалтерськими, логістичними, медичними, освітніми, інженерними, веб-орієнтованими тощо.

Основні компоненти будь-якої ІС включають:

- технічне забезпечення (апаратна частина);
- програмне забезпечення (системне й прикладне);
- бази даних (структуроване зберігання інформації);
- система користувачів (як адміністраторів, так і кінцевих користувачів);
- процедури та правила функціонування системи.

Сучасні веб-орієнтовані інформаційні системи реалізуються через інтернет або локальні мережі. Вони базуються на клієнт-серверній архітектурі, де інтерфейс користувача доступний через браузер, а обробка даних відбувається на сервері. Прикладами таких систем є системи електронного документообігу, онлайн-магазини, навчальні платформи, а також сайти для зберігання й поширення цифрових медіа-ресурсів. Як у прикладі рисунку 1.

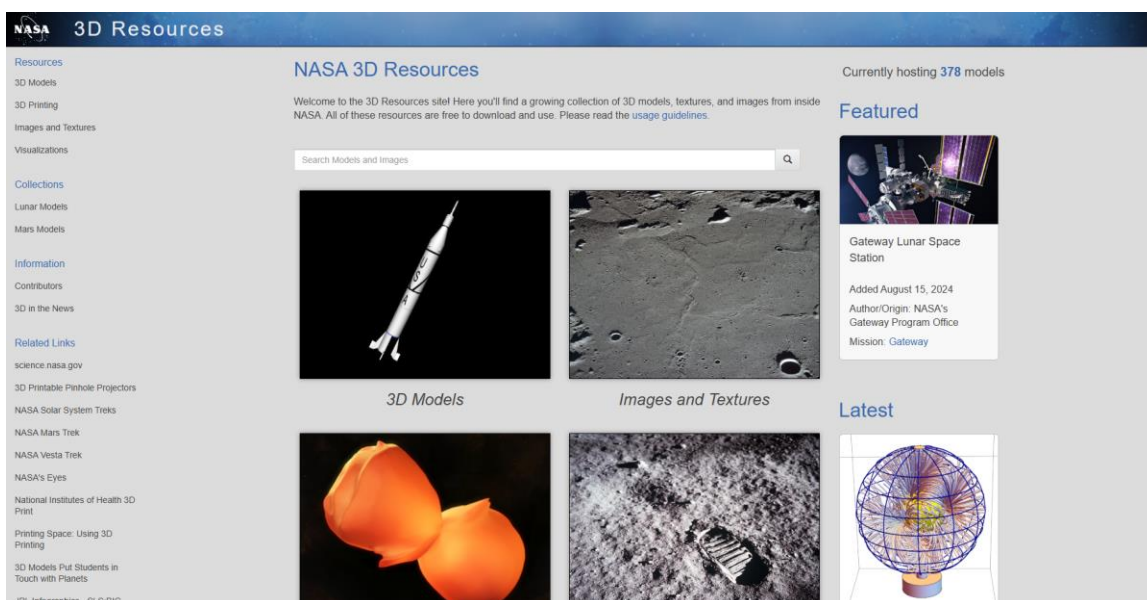


Рисунок 1 – Приклад інформаційної системи

NASA 3D Resources – це офіційна платформа ІС Національного управління з аеронавтики і дослідження космічного простору США (NASA), яка надає відкритий доступ до колекції 3D-моделей, текстур та візуалізацій космічних апаратів, супутників, астронавтів тощо [2].

Контент розміщується виключно адміністраторами NASA, тоді як будь який користувач може безкоштовно переглядати та завантажувати ресурси без реєстрації.

У випадку тривимірної графіки інформаційна система дозволяє не тільки зберігати 3D-моделі, текстури сцени, а й забезпечує їх класифікацію, пошук, сортування, візуалізацію та завантаження – це дозволяє ефективно обслуговувати потреби дизайнерів, архітекторів, ігрових розробників та інших фахівців.

Варто розрізняти поняття "веб-сайт" і "ІС". Звичайний веб-сайт може бути лише засобом представлення інформації, тоді як інформаційна система реалізує складніші функції: зберігання, пошук, сортування, керування даними та

взаємодію з користувачем. У контексті цієї роботи веб-сайт виступає інтерфейсною частиною інформаційної системи для роботи з 3D-ресурсами.

ІС може бути реалізована у вигляді веб-сайту, веб-порталу або веб-платформи – залежно від обсягу функцій, аудиторії або спільноти та цілей. Наприклад, веб-портал орієнтований на централізований доступ до різних сервісів та обміну інформацією і ресурсами між користувачами, тоді як веб-платформа надає інтерактивне середовище для завантаження, керування і спільного використання контенту. У даній системі додавання ресурсів виконується виключно адміністратором, тоді як користувачі мають доступ до пошуку, перегляду та завантаження 3D-файлів без потреби реєстрації.

1.2 Актуальність створення

Тривимірна комп'ютерна графіка за останні десятиліття набула широкого поширення завдяки розвитку апаратного забезпечення, ігрових рушіїв та програм для моделювання та дизайні продуктів (Blender, 3ds Max, Maya). Потреба в повторному використанні цифрових ресурсів таких як: моделей, текстур, сцен, HDRI-зображень, матеріалів та шейдерів виникла вже на ранніх етапах становлення CGI та візуалізації: спочатку для внутрішніх потреб студій, а згодом – і серед незалежних розробників.

У 2000-х роках почали з'являтися перші онлайн-бібліотеки моделей, де користувачі могли завантажувати прості 3D-об'єкти – здебільшого у форматах OBJ чи 3DS. З розвитком інтернету та технологій веб-візуалізації (наприклад, WebGL) платформи стали дозволяти попередній перегляд моделей прямо в браузері, а також реалізовувати інші функції: сортування за категоріями або форматом.

На ринку вже існують спеціалізовані ІС платформи, які виконують ці функції, серед них: CGTrader (рисунок 2), KitBash3D та інші [3].

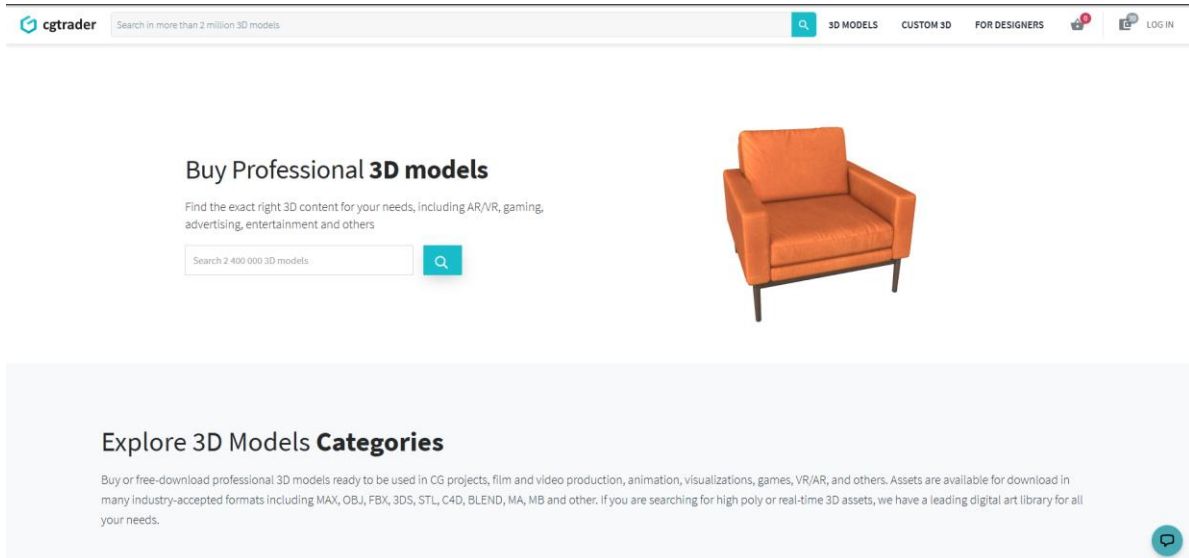


Рисунок 2 – CGTrader

CGTrader – професійний майданчик для продажу 3D-асетів. Платформа орієнтована на архітекторів, розробників ігор та AR/VR-контенту. Автори виступають як продавці, а користувачі можуть купувати моделі за ліцензіями. CGTrader також пропонує вбудовані функції пошуку за специфікаціями, які часто важливі у виробничому середовищі: полігональність, текстуризація, формат.

KitBash3D – приклад вузькоспеціалізованої платформи, зображений на рисунку 3, яка надає набори 3D-ресурсів таких як asset racks, призначені для кіновиробництва, концепт-арту та AAA-ігор (високобюджетних ігор). Кожен набір включає уніфіковані елементи (будівлі, середовища, транспорт), що значно прискорюють виробничий конвеєр, її також називають "production pipeline" – це структурований ланцюжок етапів, де результат одного етапу передається на наступний. Тобто поетапний процес створення продукту, де кожен етап з

моделювання до композиції, логічно пов'язаний і передає результати наступному кроку.

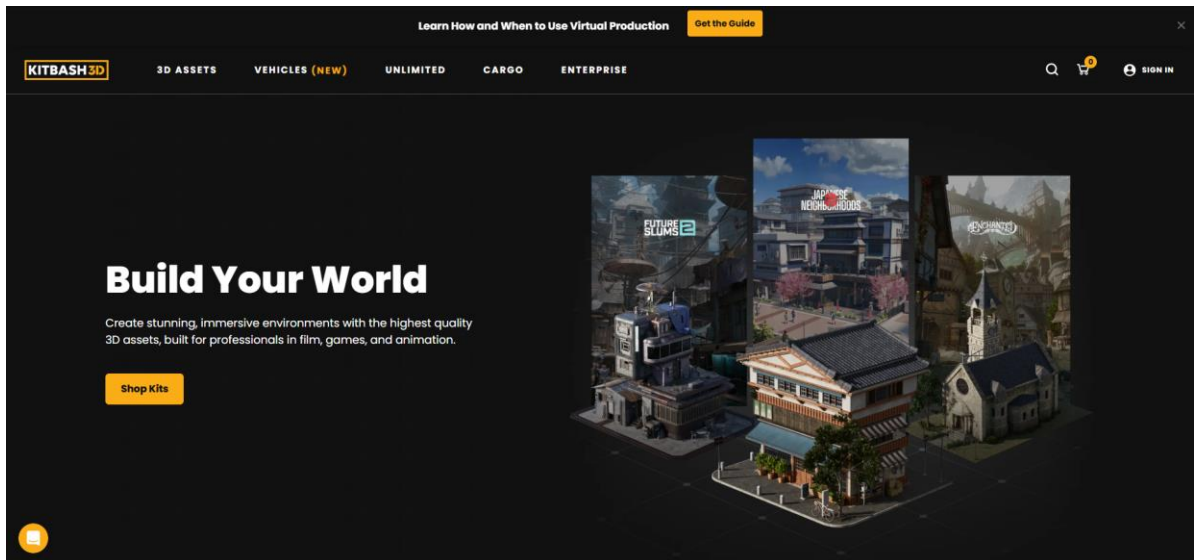


Рисунок 3 – KitBash3D

Більшість таких ІС або вимагають реєстрації, або мають змішану модель поширення (частково платну), або орієнтовані на комерційних користувачів. Крім того, не завжди реалізовані зручні механізми фільтрації моделей за категоріями, жанрами, ліцензіями чи технічними характеристиками (формат, полігональність, наявність текстур). Проте більшість з них або орієнтовані на велику кількість незалежних авторів з відповідними наслідками для якості й структури, або мають комерційну модель доступу.

Система орієнтована на користувачів, які бажають швидко знайти потрібні моделі або текстури для свого проєкту без зайвих бар'єрів. Контент додається виключно адміністратором, що дозволяє забезпечити високу якість, узгодженість форматів та відсутність дублікатів. Такий підхід дозволяє створити публічну бібліотеку з контрольованим контентом, що підходить для навчальних цілей,

невеликих ігрових або анімованих проєктів та прототипування, не вимагаючи реєстрації чи комерційної моделі доступу.

У межах цієї роботи розглядається створення вузькоспеціалізованої інформаційної системи, яка забезпечує:

- адміністратор самостійно керує наповненням, щоб уникнути дублікатів, шкідливих файлів та неякісного контенту;
- користувачі можуть лише переглядати, шукати та завантажувати 3D-моделі;
- структура сайту забезпечує зручну навігацію за категоріями, тегами, параметрами моделей (розмір, формат, кількість полігонів).

Таким чином, предметна область охоплює не лише сферу тривимірної графіки, але й питання управління цифровими ресурсами, структуризації, доступу та інтеграції в професійні робочі процеси.

1.3 Аналіз існуючих рішень

На сучасному етапі розвитку цифрових технологій існує велика кількість інформаційних систем, що забезпечують зберігання, поширення та демонстрацію 3D-моделей.

CGtrader. Комерційна платформа, що надає можливість купівлі та продажу 3D-моделей. Вона підтримує складну систему категоризації, високоякісні прев'ю, а також дозволяє авторам монетизувати свої роботи. Доступ до ресурсів здебільшого платний, користувачі мають обмежену взаємодію без реєстрації.

KitBash3D або Free3D. Спеціалізований сервіс для професійних художників, дизайнерів і студій. Пропонує платні набори моделей для використання у фільмах, іграх та інших візуалізаціях. Сайт орієнтований на високу якість та унікальність ресурсів. Основна мета – комерційне ліцензування та професійне застосування.

PolyHaven та FreeStylized. Некомерційні платформи з відкритим доступом до 3D-моделей, текстур та HDRI. Всі матеріали поширюються під ліцензією CC0, тобто можуть вільно використовуватись без зазначення авторства. Платформа забезпечує попередній перегляд моделей, категоризацію та метадані.

Artstation. Спільнота митців, де користувачі можуть демонструвати свої роботи, зокрема 3D-моделі, проте завантаження часто реалізується через сторонні сервіси. Сайт слугує переважно як візуальне портфоліо, а не сховище даних.

Порівняльний аналіз показує, що існуючі рішення поділяються на дві основні типи:

- Комерційні з фокусом на продаж;
- Некомерційні, які забезпечують відкритий доступ, але можуть мати обмежений функціонал і тематику.

Особливістю розроблюваної системи є:

- відкритий доступ до матеріалів без реєстрації;
- акцент на контент, доступний для ознайомлення, перегляду та завантаження;
- реалізація базових функцій інформаційної системи (зберігання, пошук, фільтрація, опис);
- адміністрування контенту здійснюється лише власником платформи.

1.4 Постановка завдання

Метою даної роботи є розробка веб-орієнтованої інформаційної системи для класифікації, зберігання та поширення 3D-моделей. Така система має забезпечити зручний інтерфейс для пошуку, перегляду та завантаження цифрових 3D-ресурсів, з акцентом на простоту використання, ефективну навігацію та підтримку базових функцій інформаційної системи. Основні задачі які:

- 1) Реалізувати клієнтську частину системи (веб-інтерфейс) для перегляду, сортування й пошуку моделей;
- 2) Реалізувати серверну частину, що обробляє запити, взаємодіє з базою даних та керує файлами;
- 3) Розробити базу даних для збереження інформації про моделі (назва, опис, дата, категорія, формат, кількість полігонів тощо);
- 4) Забезпечити можливість завантаження моделей адміністратором сайту;
- 5) Реалізувати механізми фільтрації, пошуку за тегами, категоріями, назвами;
- 6) Забезпечити адаптивний інтерфейс, що коректно працює на різних пристроях;
- 7) Створити мінімалістичний, функціональний та інтуїтивно зрозумілий дизайн.

Але обмеженням проєкту є:

- 1) Додавання нових моделей, зображень та їх колекціонування можливе лише через адміністративний інтерфейс;
- 2) Користувачі не мають можливості реєстрації або авторизації;

3) Система не включає можливість завантаження контенту сторонніми користувачами;

4) Мультимедійний контент (зображення, прев'ю) зберігається локально, без використання сторонніх API або хмарних сервісів.

Очікуваний результат:

1) Функціональний веб-сайт, що є частиною ІС;

2) Серверна частина з API для обробки запитів;

3) База даних для збереження медіа та їх метаданих;

4) Можливість завантаження, перегляду, пошуку та сортування моделей через інтерфейс користувача.

2 ПРОЄКТУВАННЯ СИСТЕМИ

2.1 Платформа для розробки

Інформаційна система для зберігання та класифікації веб-середовища побудована за принципом клієнт-серверної архітектури. Клієнт-серверне орієнтування – це спосіб організації взаємодії між програмами чи пристроями, де одна сторона з боку клієнта запитує ресурси чи послуги, а інша сторона з серверу надає їх. Подібна архітектура широко використовується в різних областях включаючи веб-додатки, мережі та БД. Такий підхід здійснюється з використанням сучасних веб-технологій програмного забезпечення, для швидкої розробки, масштабованості та легкості підтримки [4].

Figma. Інструмент для створення інтерфейсу, створення попередньої моделі або зразка цифрового продукту для перевірки його функціональності та дизайну веб-сторінок як на рисунку 4. Вона працює безпосередньо як у веб-браузері так і через програмне забезпечення, що робить її доступною з будь якого пристрою.

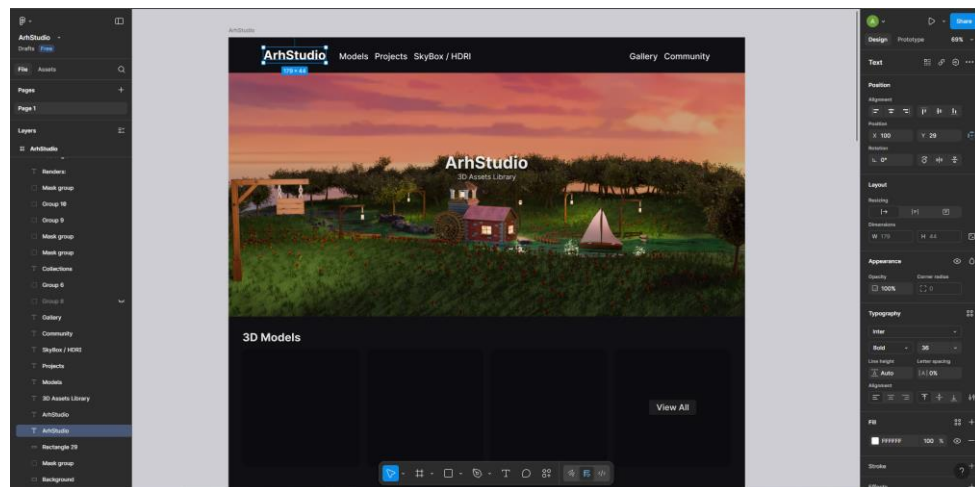


Рисунок 4 – Фінальний прототип сайту у Figma

навіть при роботі з не надто великими веб-проєктами. Незважаючи на те, що Visual Studio переважно використовується для розробки десктопних або корпоративних застосунків, воно забезпечує повноцінну підтримку для JavaScript та Node.js через відповідні розширення.

Хоча для веб-розробки частіше обирають легше середовище – Visual Studio Code, повноцінна IDE Visual Studio була обрана завдяки попередній наявності в системі, знайомству з її інтерфейсом і можливості налаштувати необхідні компоненти без додаткових установок. У поєднанні з інтеграцією систем контролю версій, підтримкою терміналу та гнучкими шаблонами проєктів Visual Studio дозволила ефективно реалізувати основні функціональні компоненти серверної частини системи та її фронтенду.

PgAdmin (рисунок 6). Офіційне графічне середовище адміністрування для системи керування базами даних PostgreSQL. Цей інструмент є кросплатформним та доступним у вигляді програмного забезпечення. У рамках розробки інформаційної системи PgAdmin використовувався для створення та структурування БД, перегляду таблиць, виконання SQL-запитів, а також для моніторингу стану сервера бази.

Однією з головних переваг PgAdmin є його зручність для візуальної роботи з БД. Вбудований SQL-редактор дозволяє записувати та тестувати запити в інтерактивному режимі, переглядати результати виконання, отримувати підказки по синтаксису. Завдяки цьому забезпечується швидкий перехід від логічного проєктування до практичної реалізації таблиць, зв'язків та обробки запитів.

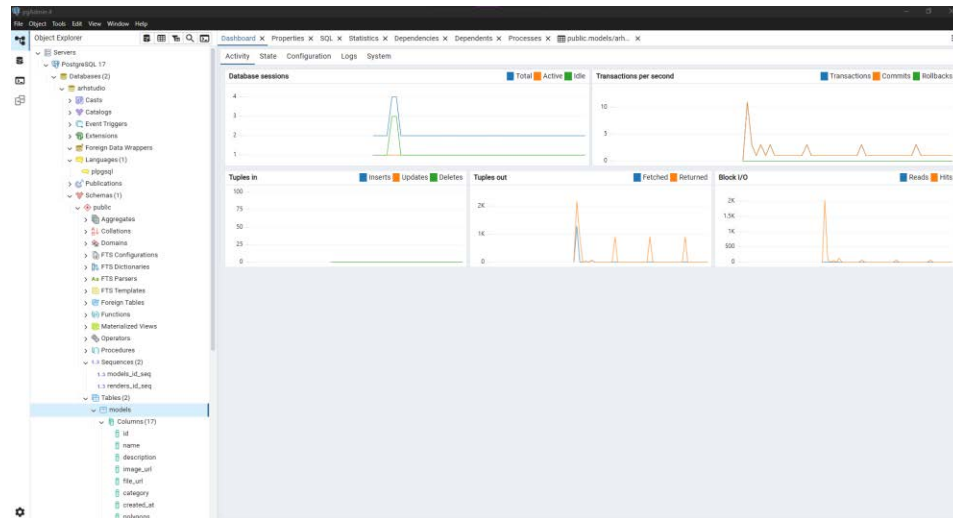


Рисунок 6 – програмне забезпечення "PgAdmin"

Pgadmin також дозволяє управляти правами доступу, створювати резервні копії бази backup, автоматизувати задачі за допомогою скриптів. Його інтеграція з PostgreSQL є безпосередньою, що робить його стандартним інструментом для розробників, які обирають цю систему управління базами даних для побудови серверної частини застосунку. У даному проєкті використовувався на етапі тестування структури БД, відладки запитів та візуальної перевірки взаємозв'язків між ними.

Причини вибору PostgreSQL та PgAdmin:

- підтримка складних структур (JSON, масиви, зв'язки багато-до-багатьох);
- надійність та відповідність ACID-принципам (атомарність узгодженість, ізолюваність, довговічність);
- активна спільнота та велика кількість розширень (наприклад PostGIS для геоданих);
- гнучкість у моделюванні відносин між сутностями;

- добре підходить для веб-додатків, які потребують чіткого контролю над структурою даних.

Недоліками ПЗ є:

- складніше в налаштуванні у порівнянні з легшими рішеннями типу SQLite;
- висока потужність функцій може бути надмірною для простих задач;
- для роботи з великими обсягами даних можуть знадобитись додаткові ресурси оптимізації (індекси, кеш, реплікація).

Notepad++. Простий текстовий редактор з відкритим кодом, призначений для роботи з різними форматами файлів, зокрема з кодом на мовах програмування. Він підтримує підсвічування синтаксису, автозаповнення, роботу з декількома вкладками, плагіни та інші корисні функції, що полегшують написання і редагування коду.

2.2 Використані веб-технології

HTML – стандартизована мова гіпертекстової розмітки [5]. Основна мова розмітки проста у використанні та розумінні для створення структури веб-сторінок. Яка використовується для побудови базової структури інтерфейсу користувача, включаючи розміщення текстових блоків, зображень, форм та інших елементів. HTML має широку підтримку всіма сучасними веб-браузерами, а також є можливість семантичної розмітки, що покращує доступність та оптимізацію для пошукових систем.

CSS – формальна мова стилю сторінок (декорування) та опису зовнішнього вигляду документу веб-сторінки, написаного з використанням мови розмітки для стилізації HTML-елементів, визначення їх зовнішнього вигляду, розміщення та адаптивності дизайну. CSS забезпечує візуальні оформлення інтерфейсу,

включаючи кольори, шрифти, відступи та адаптивний дизайн для різних пристроїв. Каскадні таблиці стилів мають розділені структури, що покращує підтримку коду. CSS може підтримувати анімацію переходів для покращення користувацького досвіду [6].

JavaScript [7]. Мова програмування з одночасним використанням множини парадигм, широкою екосистемою бібліотек та фреймворків, підтримкою об'єктно-орієнтованого, імперативного та функціонального програмування. JS, яка дозволяє додавати інтерактивність до веб-сторінок, використовується для обробки подій, валідації форм, динамічного оновлення контенту та взаємодії з сервером через API.

Node.js [8]. Середовище виконання JavaScript на стороні сервера, яке дозволяє створювати масштабовані та високопродуктивні веб-додатки. Воно додає можливість JS взаємодіяти з пристроями вводу-виводу через свій API, написаний на мові C++, а також підключати зовнішні бібліотеки, написані іншими мовами програмування, забезпечуючи виклики до них із JavaScript коду. Node.js застосовується переважно на сервері, виконуючи роль веб-сервера.

Завдяки асинхронній обробці запитів, Node.js забезпечує високу продуктивність і масштабованість. Однією з ключових переваг є використання єдиної мови програмування JS для клієнтської та серверної частини, що спрощує розробку. Крім того, велика екосистема модулів, доступна через Node Package Manager, дозволяє швидко розширювати функціональність застосунку.

2.3 Бази даних

У контексті системи класифікації та зберігання 3D-моделей та її інформації, база даних виконує роль основного інструменту для структурованого представлення інформації про моделі, їхні технічні параметри, зображення,

категорії, теги, дати публікації, забезпечує можливість керування вмістом з боку адміністратора [9].

Процес проєктування БД охоплює кілька взаємопов'язаних рівнів моделювання: концептуальний, логічний та фізичний. Кожен з них відображає різний рівень абстракції: від загального уявлення про предметну область до конкретного технічного втілення в системі керування базами даних.

Загальні поняття баз даних – це організована сукупність взаємопов'язаних даних, що зберігаються в електронному вигляді та можуть ефективно оброблятися і керуватися за допомогою спеціального програмного забезпечення СКБД.

SQL – це мова структурованих запитів, що використовується для створення, модифікації, пошуку та керування даними у реляційних баз даних.

SQL є стандартом для більшості СКБД, включаючи PostgreSQL, MySQL тощо. Мова дозволяє створювати таблиці та встановлювати зв'язки між ними (Create Table, Alter Table), додавати, змінювати або видаляти запити (Insert, Update, Delete), здійснювати фільтрацію, сортування та агрегацію даних (Select, Where, Order By, Group By), керувати правами доступу, транзакціями, індексами тощо.

SQL забезпечує ефективний доступ до великих обсягів структурованих даних і є ключовим інструментом у роботі з будь-якою реляційною БД.

Бази даних класифікують за різними критеріями. Один із ключових моделей організації даних, відповідно до якої розрізняють такі типи:

- реляційні БД (таблична структура, зв'язки через ключі) найпоширеніші в прикладному програмуванні;
- документно-орієнтовані БД (зберігання документів формату JSON або XML);

- графові БД (представлення у вигляді вузлів і ребер – для складних зв'язків між об'єктами);
- об'єктно-орієнтовані БД (поєднують принципи об'єктного програмування з зберіганням даних);
- ключ-значення – для зберігання простих асоціативних пар.

Концептуальна модель – це абстрактне уявлення предметної області, що описує сутності предметної області, їхні атрибути та зв'язки між ними без урахування технічної реалізації (рисунок 7). Вона не залежить від конкретної СУБД, не містить типів даних і не враховує фізичне зберігання.

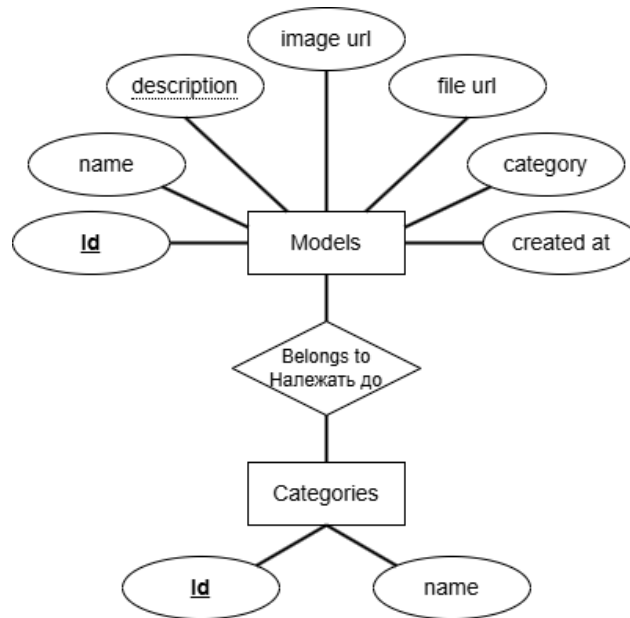


Рисунок 7 – Перша концептуальна модель

Модель створюється на етапі аналізу вимог і використовується для формалізації того, які об'єкти існують у системі, як вони пов'язані між собою та яку інформацію мають зберігати. Вона зазвичай представлена у вигляді ER-діаграм з сутностями (Entity), зв'язками (Relationship) та атрибутами (Attribute).

Концептуальна модель – це перший етап проєктування, який допомагає й дозволяє узгодити бачення структури даних між розробниками та замовником, і служить основою для подальшого логічного моделювання.

Логічна модель або логічна схема – це структуроване уявлення даних, яке вже адаптоване до обраної парадигми бази даних, але ще не враховує фізичної реалізації.

Логічна модель (рисунок 8) деталізує концептуальну, але вже з урахуванням реляційної структури бази даних. Тут визначаються назви таблиць, атрибути, первинні та зовнішні ключі, типи зв'язків.

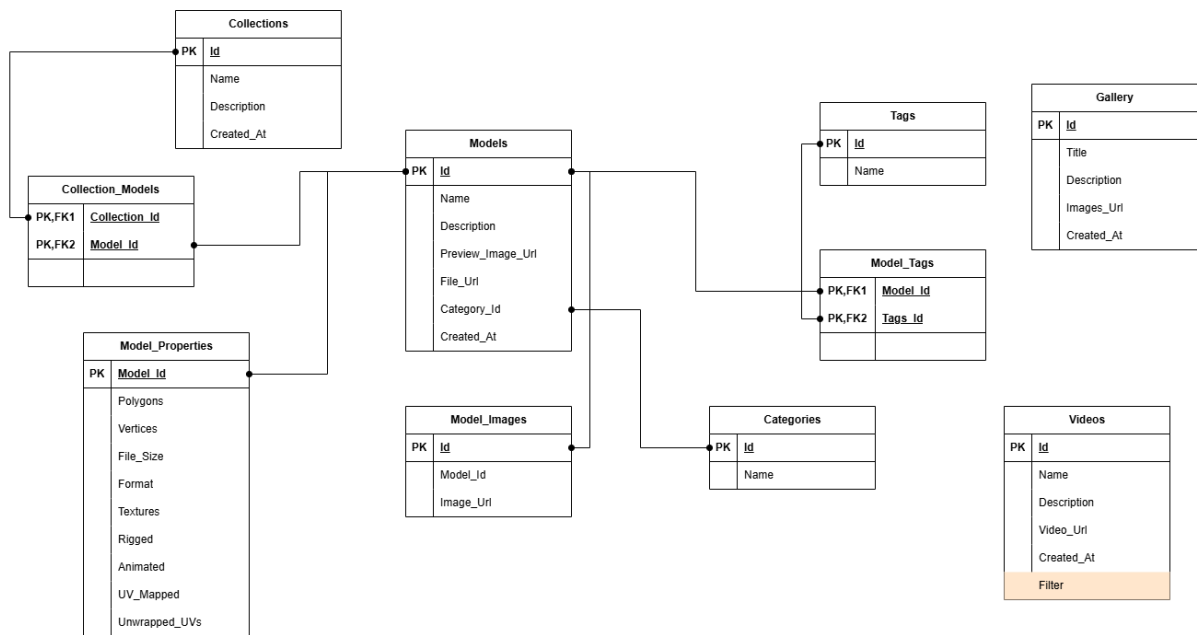


Рисунок 8 – Логічна модель БД

На цьому етапі кожна сутність із концептуальної моделі перетворюється в таблицю. Атрибути перетворюються на поля таблиць з відповідним типом даних, а зв'язки між сутностями – на зовнішні ключі або зв'язувальні таблиці (у випадку відносин типу "багато до багатьох").

Крім цього, визначаються правила унікальності, обов'язковості заповнення полів (NOT NULL), а також індекси для полів, що будуть часто використовуватись у пошуку або сортуванні.

Фізична модель є втіленням логічної структури у конкретній СКБД PostgreSQL. Вона являє собою конкретну логічну структуру БД, яка описує, як дані фактично зберігаються у фізичному реальному сховищі (наприклад, на жорстких дисках) і як до них звертаються СУБД. Фізична модель визначає структуру таблиці, поля, індексів, обмеження та інші деталі необхідні для фактичного створення та функціонування БД.

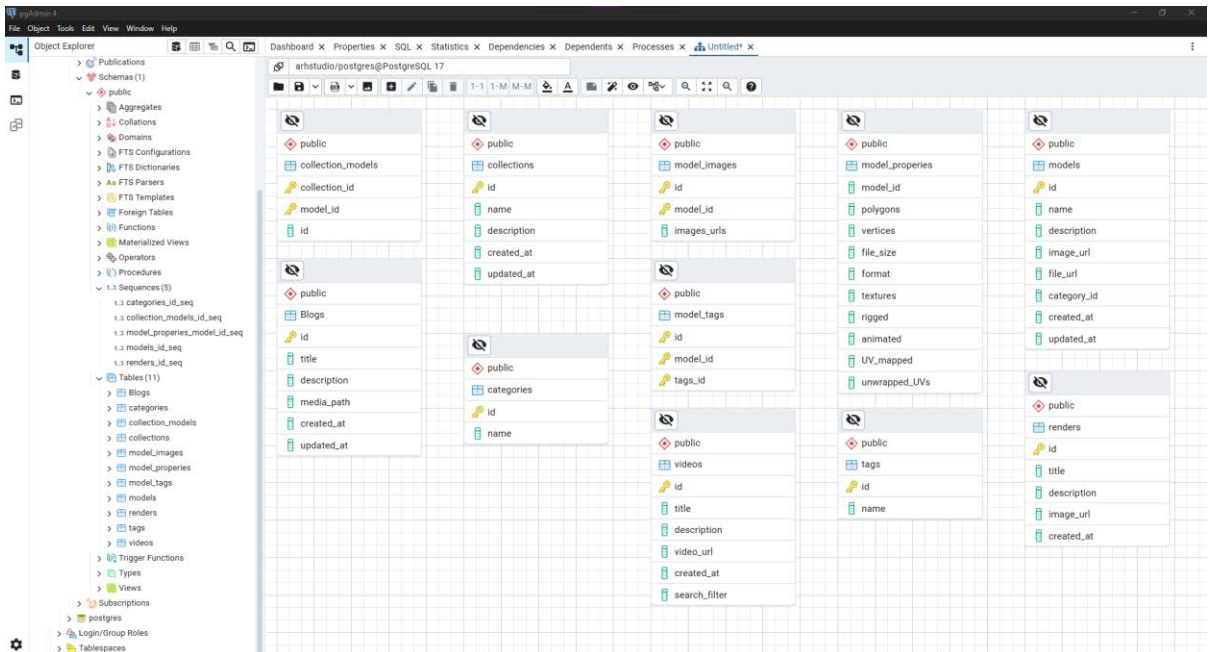


Рисунок 9 – Фізичні бази даних

Зовнішні ключі забезпечують зв'язність, а унікальні ключі та індекси – швидкість доступу до даних. Різниця між фізичною та логічною моделлю полягає в тому, що остання описує дані в абстрактному вигляді, не звертаючи уваги на

фізичні аспекти реалізації, тоді як фізична модель конкретно визначає структуру та взаємозв'язки об'єктів БД.

2.4 Проєктування бізнес-процесів

Модель та нотація бізнес-процесів BPMN – це графічна нотація для моделювання бізнес-процесів, що широко використовується у проєктуванні інформаційних систем. Вона призначена для опису, аналізу та оптимізації процесів у зрозумілому вигляді як для бізнес-користувачів, так і для технічних фахівців. Модель може створювати діаграми, придатні як для документації, так і для автоматизації у BPMS.

BPMN дозволяє візуалізувати послідовність дій у процесі за допомогою блок-схем, використовувати основні елементи: події (events), дії (activities), розгалуження (gateways) та потоки (flows).

IDEF0 – це методологія моделювання функцій (функціонального аналізу), що робить система, організація або процес. Основна увага в IDEF0 – структура та взаємодія функцій, а не їх послідовність.

У рамках практичної реалізації було обрано BPMN, оскільки дана нотація дозволяє більш наочно представити логіку виконання процесів у часі та взаємодію між учасниками системи, що є ключовим при моделюванні роботи інформаційної системи веб-платформи (рисунок 10).

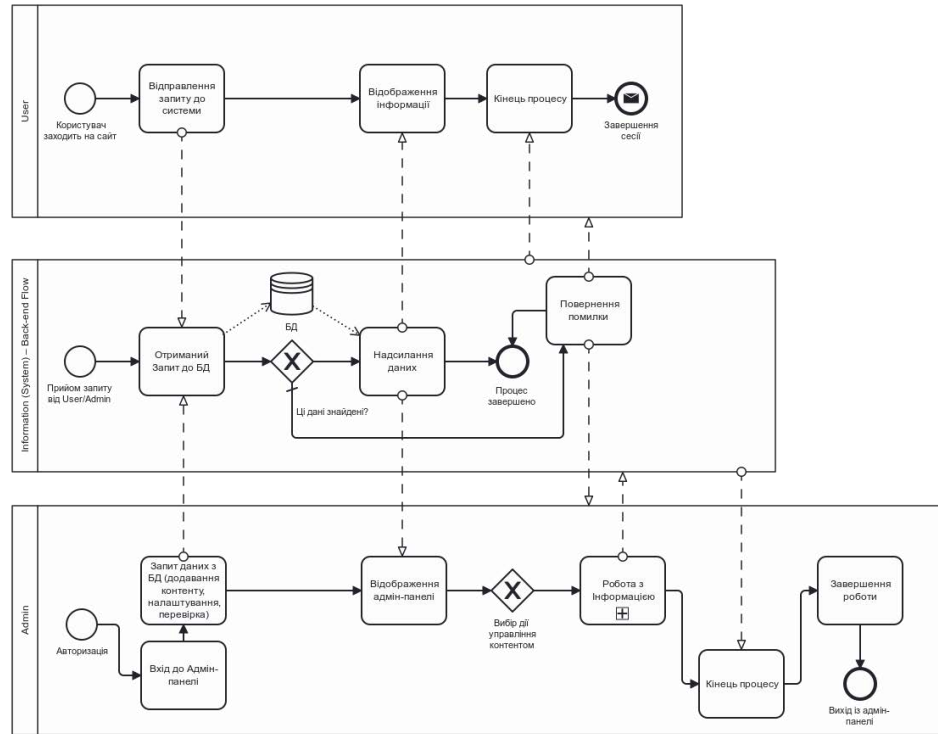


Рисунок 10 – Схема процесів Інформаційної системи

Опис функціонування ІС платформи. Робота інформаційної системи організована навколо трьох основних процесів: взаємодії користувача з інтерфейсом, дій адміністратора через панель керування і обробка запитів серверною частиною системи. У реалізованій ІС важливу роль відіграє розділення функцій між адміністрацією, користувачем і системою. Такий підхід дозволяє чітко розмежувати відповідальність, права доступу, та забезпечити вищий рівень безпеки. Адміністрація має права доступу до створення облікових записів і їх керуванням, у той час як система керує ресурсами, процесами і виконує критичні внутрішні операції. Це розділення спрощує масштабування, контроль і моніторинг.

Взаємодія з боку користувача. Після відкриття сайту користувач отримує доступ до основного контенту – каталогу моделей, галереї зображень, колекцій та інших матеріалів. Ці дані завантажуються з бази даних на основі первинного

запиту до серверної частини. Користувач може переглядати доступний контент, застосовувати фільтри за тегами та категоріями для зручного пошуку. Запит з урахуванням фільтрів обробляється системою, і відфільтровані результати відображаються на сторінці.

При виборі окремої моделі відкривається її персональна сторінка, де представлено зображення, опис та інші параметри. Якщо модель доступна для завантаження, користувач отримує посилання на файл. Це посилання формується сервером, який також відповідає за перевірку наявності файлу та коректність даних перед передачею.

Дії адміністратора. Адмін отримує доступ до функцій керування контентом через окрему панель. Після входу в систему відбувається завантаження поточних даних (моделі, медіа, пости новин). У адміністрації є можливість додавати нові матеріали та перевіряти коректність перед публікацією, редагувати наявні або видаляти їх.

Усі дії адміністратора автоматично передаються на сервер для обробки. Після успішного оновлення БД зберігає зміни, які повертаються до панелі керування у вигляді актуалізованої інформації.

Обробка запитів сервером. Серверна частина ІС виконує роль центрального модуля, який забезпечує стабільну роботу системи та зв'язок між користувачами, адміністраторами і базою даних. Вона обробляє запити, зберігає та оновлює записи в БД, здійснює пошук інформації за заданими параметрами. Крім того, сервер керує файлами, тобто перевіряє їх відповідність формату, фізично зберігає на сервері та створює посилання для завантаження. У разі виникнення помилок система формує відповідні повідомлення і передає їх до клієнтської або адміністративної частини.

3 ПРОГРАМНА РЕАЛІЗАЦІЯ СИСТЕМИ

3.1 Загальна характеристика реалізованої ІС

Інформаційна система, що розробляється являє собою веб-платформу для публікацій, класифікації та завантаження 3D-моделей, текстур, візуалізацій і супутніх матеріалів. Основною метою створення цієї системи є надання зручного середовища для пошуку, перегляду та завантаження як для авторів контенту пов'язаним з 3D-індустрією, так і для користувачів, які шукають якісні ресурси для своїх проєктів, таких як візуалізацій, ігор, освітніх або наукових розробок [10].

Система підтримує відкритий доступ до матеріалів, не вимагаючи реєстрації, що робить її максимально доступною та простою у використанні. Водночас адміністратор має повний контроль над наповненням платформи, модерацією матеріалу, а також проведенням операцій через окрему панель керування.

Основні функціональні можливості інформаційна система реалізує наступні ключові функції, що сприяють зручній та ефективній роботі.

Перегляд та пошук матеріалів. Користувачі можуть переглядати каталог моделей, застосовувати пошук та фільтрувати їх за тегами, категоріями чи популярністю. Завдяки інтерактивному інтерфейсу, пошук дозволяє оперативно знаходити потрібні ресурси.

Відображення детальної інформації про модель. Для кожного елемента створено окрему сторінку, на якій представлена галерея зображень, розгорнутий опис, технічні характеристики про кількість полігонів, розмір файлу тощо, та можливість завантаження 3D-моделі або архіву файлів з текстурами.

Управління контентом. Адміністратор через спеціальну панель отримує повноваження завантажувати нову інформацію, редагувати чи змінювати назву, додавати опис або видаляти вже існуючі матеріали, перевіряти відповідність файлів вимогам системи, а також виконувати класифікацію за категоріями та тегами.

Публікація супутнього матеріалу. Платформа надає можливість розміщення візуалізацій, відео-презентацій а точніше анімацій, та статті з новинами, що збільшує охоплення інформації та робить ресурс більш цікавим для користувачів.

Збір звернень від користувачів. Реалізація форми зворотного зв'язку дозволяє користувачам надсилати повідомлення адміністрації щодо запитань, пропозицій або проблем, забезпечуючи оперативну комунікацію.

3.2. Структура інформаційної системи

З архітектурної точки зору платформа побудована за принципом розділення функціональних компонентів, що дозволяє забезпечити масштабованість та зручність підтримки:

- клієнтська інтерфейсна частина – забезпечує взаємодію через браузер. Користувачі здійснюють перегляд контенту, фільтрацію та детальний перегляд моделей та їх завантаження, що знаходять з серверної частини;

- адміністративна частина – відокремлена область від зайвих користувачів, доступна лише адміністратору, через яку здійснюється керування контентом платформи, контроль за даним та модеруванням ресурсів;

- серверна логіка (системний рівень) – забезпечує обробку запитів користувачів і адміністраторів, виконує роботу з базою даних, обробляє збереження файлів та іншого медіа-контенту і управляє бізнес-логікою.

Взаємодія між компонентами здійснюється через стандартні запити до бази даних та передачу файлів, що гарантує розділення відповідальності та сприяє масштабуванню системи.

3.3 Модулі та компоненти сайту

Для зручності підтримки, модернізації та розширення функціональності платформа розділена на ряд окремих модулів або сторінок, кожен з яких має чітко визначені межі функціоналу. Всі елементи системи спроектовано таким чином, щоб забезпечити максимальну гнучкість, модульність та взаємозв'язок між розділами. На рисунку 11 представлено загальну схему структури модуля для користувача, який є центральним у взаємодії з інформаційною системою.

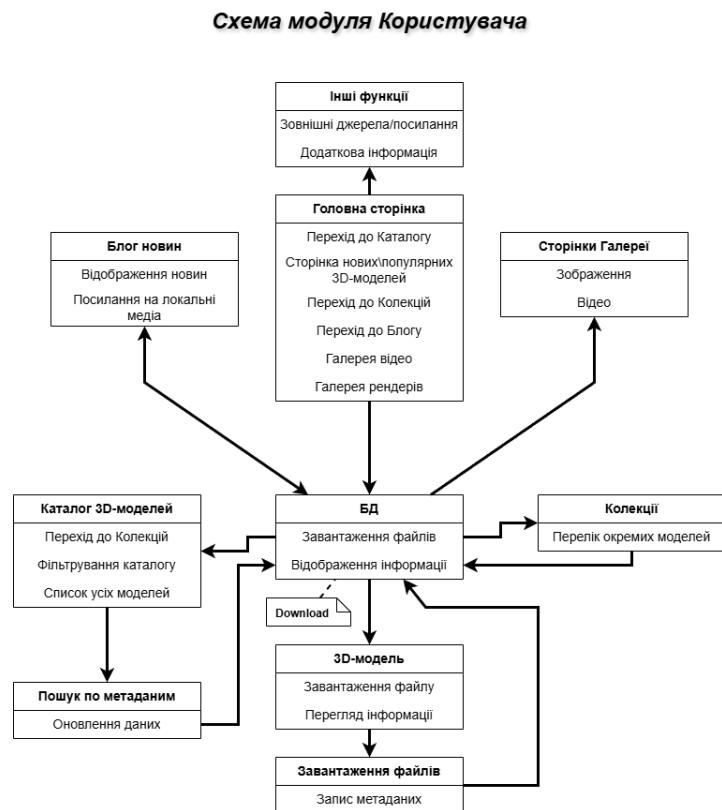


Рисунок 11 – Основний модуль схеми користувачів

3.3.1 Головна сторінка

Головна сторінка є центральним вузлом платформи, який об'єднує усі ключові розділи ресурсу (рисунок 12). При її створенні було враховано принципи зручності навігації, першого враження та презентаційності, оскільки саме ця сторінка найчастіше є першою взаємодією користувача з платформою.

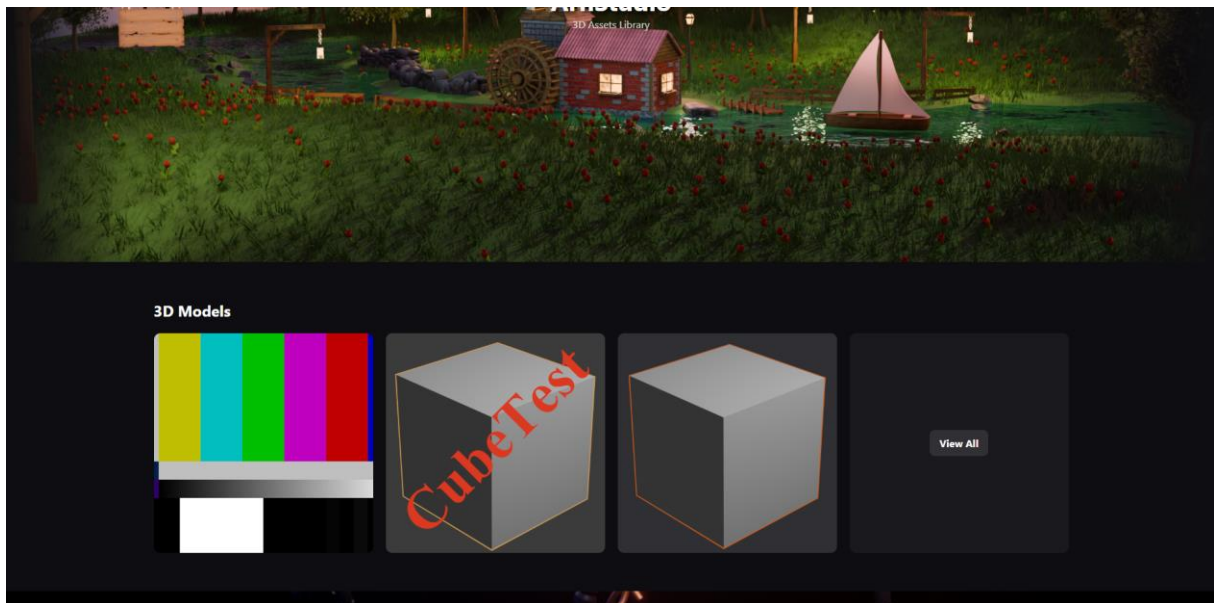


Рисунок 12 – Вигляд основних моделей та перехід до каталогу

У верхній частині інтерфейсу відображається інтерактивна галерея трьох останніх або найпопулярніших 3D-моделей. Цей блок призначений для того, щоб миттєво зацікавити відвідувача найбільш актуальним або візуально привабливим контентом. Поряд блоком передбачено елемент переходу до повного каталогу моделей, що дозволяє розгорнуто ознайомитися з наявною бібліотекою.

Під основним вітриною моделей розташований блок колекцій (рисунок 13). Колекції виконують функцію тематичного або стилістичного угруповання об'єктів. Завдяки цьому користувачі можуть не просто переглядати окремі

моделі, а занурюватися в цілісні підбірки – наприклад, "архітектурні елементи", "стилістичні набори" або "процедурно генеровані ноди". Такий підхід широко використовується в платформах на кшталт ArtStation або Sketchfab і дозволяє сприймати контент більш цілісно і структуровано.

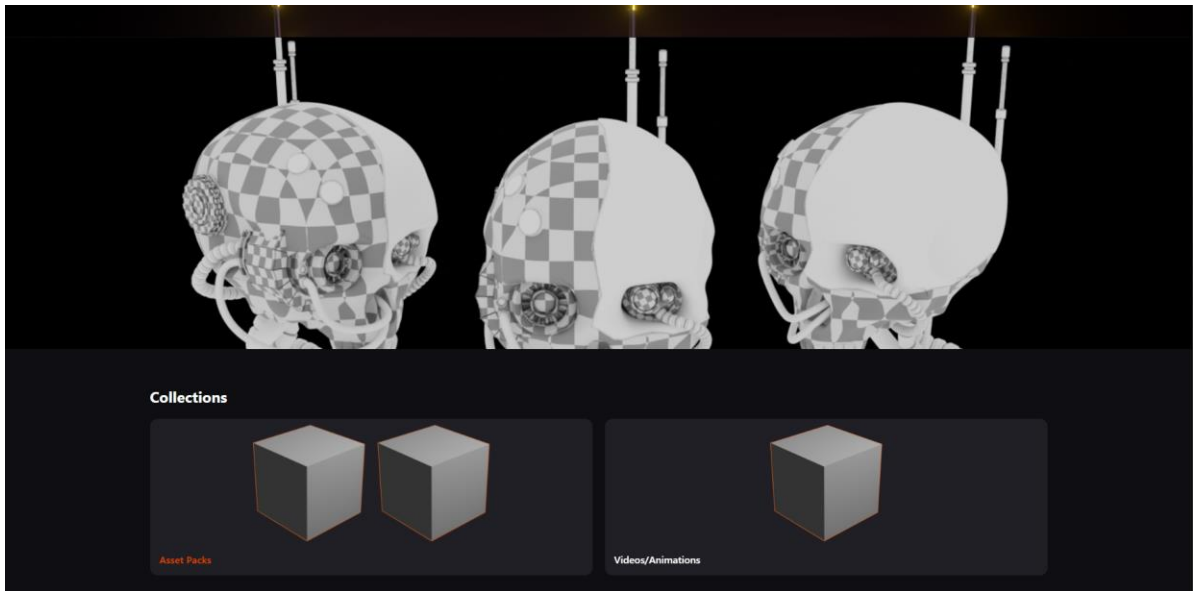


Рисунок 13 – Вигляд колекцій і відео на даний момент

Поруч розміщено мультимедійний блок, присвячений відеоматеріалам анімацій. Тут можуть демонструватися як презентації окремих моделей, так і загальні демонстраційні ролики, створені авторами або командою сайту. Це розширює можливості презентації проєктів, надаючи динамічну альтернативу статичним зображенням.

Центральну частину сторінки завершує інформаційна секція "About Us" – короткий опис концепції, призначення та цінностей платформи. Її завдання познайомити нових користувачів з ідеєю проєкту, підкреслити безкоштовність, відкритість та орієнтацію на спільноту 3D-художників (рисунок 14).

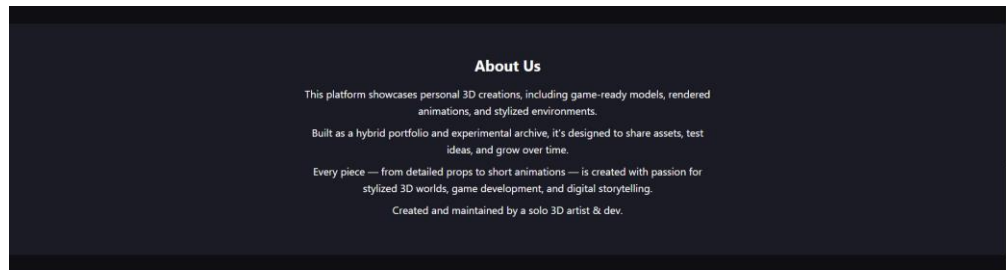


Рисунок 14 – секція "About Us"

Перед футером розміщується компактна галерея зображень – міні-прев'ю, які ведуть до повноцінної сторінки з усіма візуалізаціями, що були завантажені або пов'язані з 3D-творчістю. Така галерея дозволяє в інтерактивній формі ознайомитися з якістю рендерів та оформленням проєктів (рисунок 15).

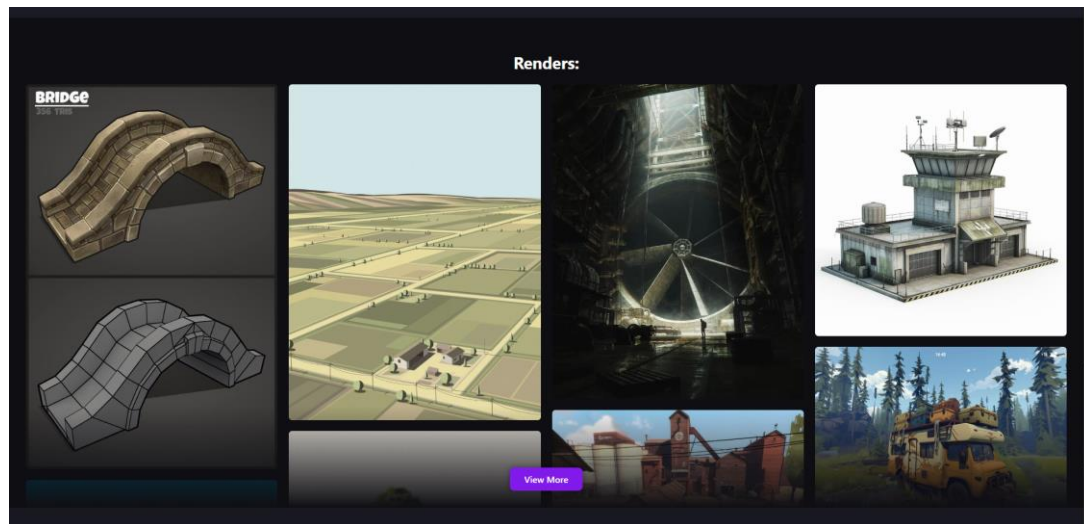


Рисунок 15 – Міні-каталог зображень

Наприкінці сторінки подано контактні дані та зовнішні посилання на пов'язані ресурси або соціальні платформи (рисунок 16). Завершується головна сторінка традиційним футером з компактною навігацією – повторенням

основних розділів сайту для зручного повернення або швидкого переходу до іншого розділу.

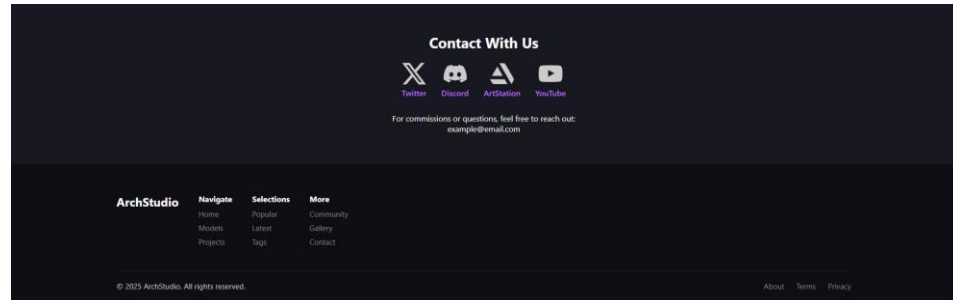


Рисунок 16 – Контактні дані та кінець головної сторінки

3.3.2 Каталог моделей

Модуль сторінки каталогу моделей є основним розділом, з усією базою доступного матеріалу. Користувачі мають змогу швидко перемикатися між різними категоріями завдяки блоку фільтрації розміщеному збоку (рисунок 17).

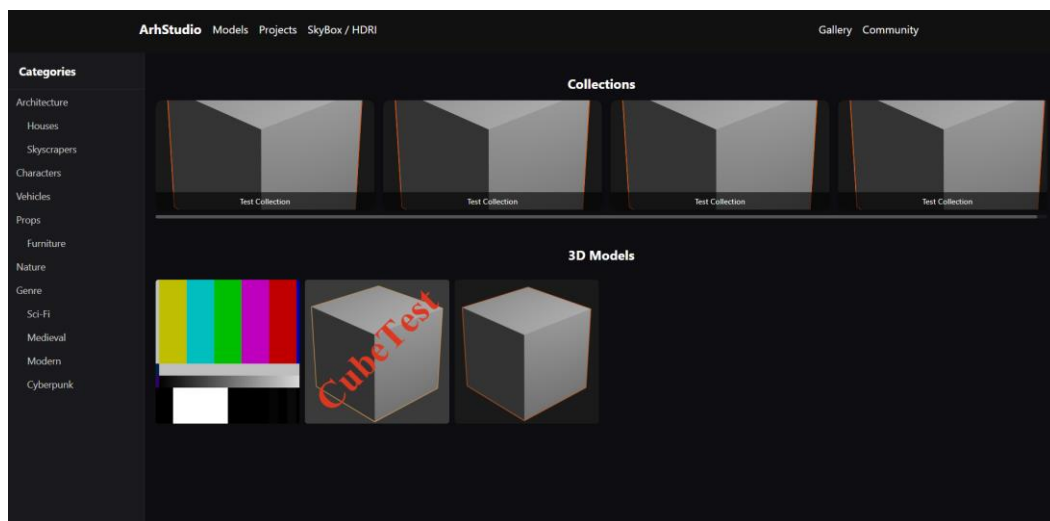


Рисунок 17 – Каталог 3D-моделей

Передбачена можливість поєднувати кілька тегів для точнішого результату. У верхній частині сторінки окремо винесені добірки колекцій – це дозволяє ознайомитися з модельними наборами, що поєднані темою, та іншими пов’язаними даними. Решта простору займає список моделей, що постійно оновлюється в міру додавання нового контенту.

3.3.3 Персональна сторінка моделі

Для кожної моделі реалізовано окрему сторінку, де представлено повну інформацію щодо конкретного елемента. Ліва частина, як правило, відведена під основне зображення як рендер або прев’ю, та невелику кількість додаткових візуалізацій, яке максимально точно відображає вигляд об’єкта. Зображення адаптується під розмір екрану й може бути відкрито в повноекранному режимі (рисунок 18).

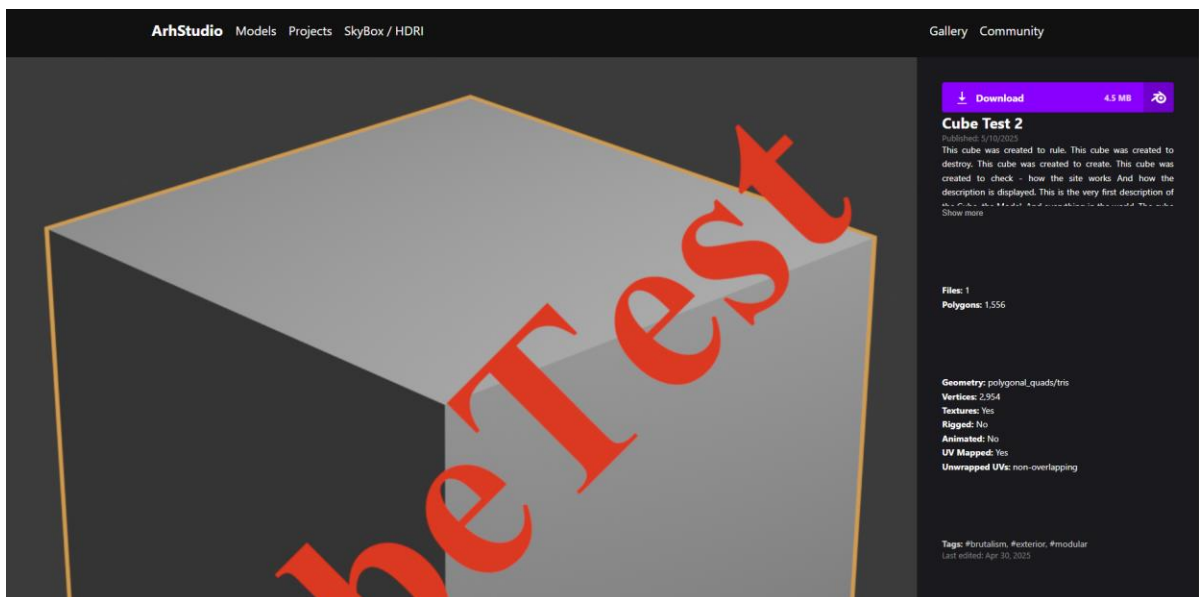


Рисунок 18 – Сторінка 3D-моделі

У правій частині сторінки розміщено інформаційний блок із кнопкою завантаження. Тип файлу або формат вказується поряд. Нижче наведено назву, короткий та розгорнутий опис моделі, технічні характеристики, дата завантаження та відповідні теги й категорії. Також відображається дата останньої модифікації, що дозволяє відстежувати оновлення.

3.3.4 Інші сторінки та функції

Окрім основних сторінок, система також передбачає реалізацію низки додаткових розділів, які на момент написання ще перебувають у процесі активної розробки. Зокрема, це розділ галереї з детальним переглядом рендерів, відеоархів, блог із публікаціями новин, а також сторінка контактної форми, яка дозволяє залишити повідомлення адміністрації. Попри те, що ці модулі ще не завершені повністю, їхня структура вже закладена в загальний дизайн системи, що забезпечує їх плавну інтеграцію в майбутньому.

3.4 Модуль адміністратора

Модуль адміністратора забезпечує інтерфейс для керування інформацією платформи, зокрема 3D-моделями, те є ключовим для підтримки та оновлення ресурсу. Уся функціональність реалізована на одній сторінці, інтерфейс якої поділені на основні логічні блоки, як приклад, панель додавання нових матеріалів та каталог збережених записів. Схема взаємодії адміністратора з модулем наведена на рисунку 19.

Схема модуля Адміністратора

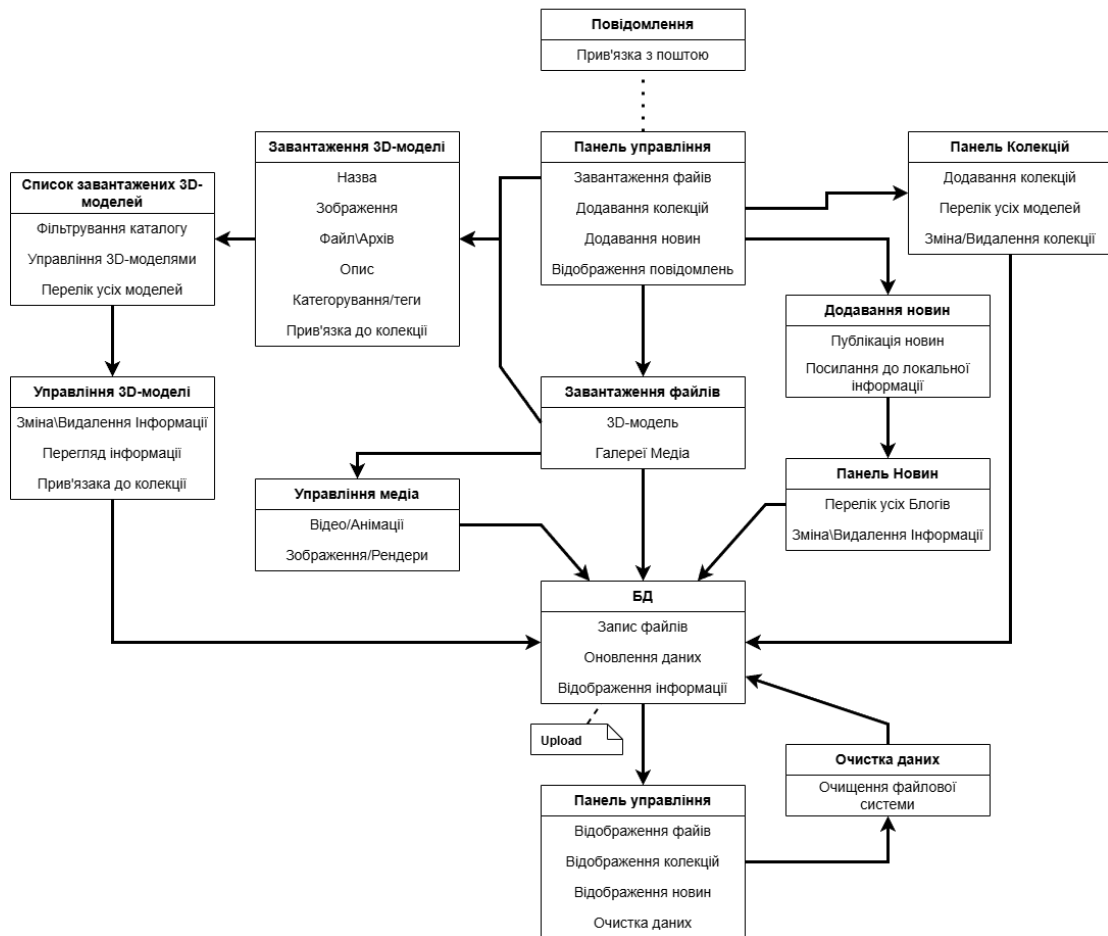


Рисунок 19 – Схема роботи модуля адміністрації

Верхня частина адміністративної сторінки містить зручну форму додавання нових моделей. У відповідних полях адміністратор заповнює назву моделі, обирає основний файл, додає зображення попереднього перегляду, та вказує супровідну інформацію – опис, технічні параметри, категорії, тощо. Ця форма слугує базовим засобом публікації нового контенту на платформі.

Нижче розташований каталог уже завантажених моделей. Кожен запис у цьому списку представлений у вигляді окремого блоку з візуальним прев'ю, короткою інформацією та службовими кнопками. Адміністратор може

редагувати існуючі дані моделі або повністю видалити запис. Це дозволяє швидко оновлювати вміст або коригувати помилки без необхідності перемикання між сторінками (рисунок 20).

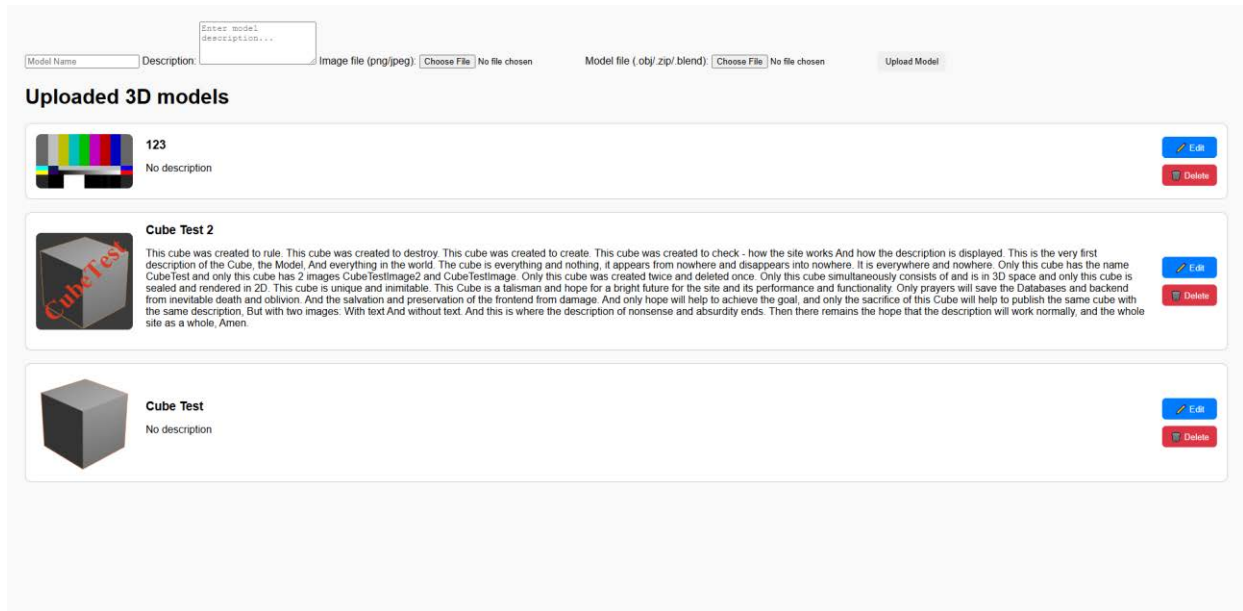


Рисунок 20 – Панель адміністратора

Модуль також передбачає розширення функціональності у майбутньому. Планується інтеграція системи тегів з підтримкою швидкого пошуку, автозаповненням та класифікацією за жанрами й категоріями. Крім того, інтерфейс буде доповнено засобами для керування кількома зображеннями, налаштуванням метаданих, таких як, полігональність, текстурування, дата зміни, відображенням статистики завантажень та можливістю вбудованої перевірки файлів на коректність структури.

Завдяки централізації всіх функцій на одній сторінці, модуль адміністратора забезпечує швидкий доступ до інструментів керування, підвищуючи ефективність роботи без перевантаження інтерфейсу зайвими переходами. Це рішення особливо зручне на початковому етапі розвитку

платформи, коли адміністрування здійснюється однією особою або невеликою командою.

3.5 Тестування інформаційної системи

Тестування інформаційної системи проводилося з метою перевірки коректності реалізованих функцій, виявлення потенційних помилок, а також перевірки надійності обробки даних на стороні клієнта та сервера. Застосовувався ручний метод тестування із використанням браузера, локального сервера та інтерфейсу адміністрування.

Тестування додавання 3D-моделі. Для додавання моделі було перевірено обов'язковість заповнення всіх полів перед відправленням. У разі, якщо деякі з них, наприклад, файл, назва чи зображення не заповнені – сервер не приймає запит і не додає модель у базу даних. Перевірка за наявності додавання файлу за допомогою консолі на рисунку 21.

```

Added to database: {
  id: 71,
  name: 'Cube Test 2',
  description: 'This cube was created to rule. This cube was created to destroy. This cube was created to create. This cube was created to check - how the site works And how the description is displayed. This is the very first description of the Cube, the Model, And everything in the world.\r\n' +
    '\r\n' +
    'The cube is everything and nothing, it appears from nowhere and disappears into nowhere. It is everywhere and nowhere. Only this cube has the name CubeTest and only this cube has 2 Images CubeTestImage2 and CubeTestImage.\r\n' +
    '\r\n' +
    'Only this cube was created twice and deleted once. Only this cube simultaneously consists of and is in 3D space and only this cube is sealed and rendered in 2D.\r\n' +
    '\r\n' +
    'This cube is unique and inimitable.\r\n' +
    '\r\n' +
    'This Cube is a talisman and hope for a bright future for the site and its performance and functionality. Only prayers will save the Databases and backend from inevitable death and oblivion. And the salvation and preservation of the front end from damage.\r\n' +
    '\r\n' +
    'And only hope will help to achieve the goal, and only the sacrifice of this Cube will help to publish the same cube with the same description, But with two images: With text And without text.\r\n' +
    '\r\n' +
    'And this is where the description of nonsense and absurdity ends. Then there remains the hope that the description will work normally, and the whole site as a whole, Amen.',
  image_url: '/uploads/images/1746873783092-CubeTestImage2.png',
  file_url: '/uploads/models/1746873783095-CubeTest.zip',
  category: null,
  created_at: 2025-05-10T10:43:03.259Z,
  polygons: null,
  file_size: 115874,
  format: '.zip',
  vertices: null,
  textures: null,
  rigged: null,
  animated: null,
  uv_mapped: null,
  unwrapped_uv: null,
  images_urls: null
}

```

id	name	description
70	Cube Test	[null]
71	Cube Test 2	This cube was created to rule. This cube was created to destroy. This cube was created to create. This cube was created to check - how the site works And how the description is displayed. This is the very first description of the Cube, the Model, And everything in the world.
72	123	[null]

Рисунок 21 – Завантаження файлу до серверу

Під час завантаження зображень та формату файлу, перевірено зв'язок із відповідними полями та категоріями, що забезпечує правильну структурування при виведенні на головній сторінці та в каталозі.

Моделі, що успішно завантажені, зберігаються у базі даних та виводяться на головній сторінці, у розділі моделей, а також у формі перегляду конкретної моделі.

У системі реалізовано дві основні функції очищення:

1) Ручне видалення моделі через інтерфейс адміністратора. Функція дозволяє повністю видалити модель з бази даних, а також пов'язані з нею файли з сховища.

2) Автоматичне очищення – якщо в директоріях зберігання знаходяться файли, які не мають записів у базі даних, вони видаляються автоматично під час запуску сервера.

ВИСНОВКИ

У процесі виконання дипломної роботи на тему "Розробка інформаційної системи класифікації та зберігання 3D-моделей" було розглянуто етапи аналізу, проєктування та її реалізації. На основі вивчення предметної області сформовано вимоги до функціональності, архітектурі, створено базу даних та розроблено веб-інтерфейс користувача й адміністратора.

Отримане рішення є гнучким і масштабованим, його можна розширити для підтримки нових типів 3D-галузі – текстур, HDRІ, анімацій, сцен, а також для додавання нових функцій, таких як метадані, статистика завантажень, модерація та фільтрація.

Необхідність розробки подібної системи обумовлена зростаючим попитом на цифрові 3D-ресурси в галузях дизайну, ігрової індустрії, візуалізації та освіти. Запропонована інформаційна система є універсальною і може використовуватись як платформа для особистого портфолію, як внутрішній інструмент для організації бібліотек 3D-ресурсів, або як основа для повноцінного онлайн-сервісу. Її гнучка структура забезпечує адаптацію до конкретних потреб користувачів, що робить систему актуальною та корисною у сучасному цифровому середовищі.

ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Wikipedia. Інформаційна система [Електронний ресурс] – https://uk.wikipedia.org/wiki/Інформаційна_система
2. NASA. 3D Resources [Електронний ресурс] – <https://nasa3d.arc.nasa.gov>
3. Maker Trainer. List of 3d models sharing websites [Електронний ресурс] – https://makertrainer.com/wiki/List_of_3D_model_sharing_websites
4. Wikipedia. Клієнт-серверна архітектура [Електронний ресурс] – https://en.wikipedia.org/wiki/Client-server_model
5. Wendy Willard. A beginner`s Guide, Fourth Edition [Електронне видання] – <https://www.snggdcg.ac.in/pdf/study-material/computer-science/0071611436%20HTML.pdf>
6. Duckett J. HTML and CSS: Design and Build Websites [Електронне видання] – Wiley, 2011 – Режим доступу: https://fita.vnua.edu.vn/wp-content/uploads/2018/12/TH03109_HTML-and_CSS-design-and-build-websites_giaotrinhchinh2.pdf
7. JavaScript Info. Сучасний підручник [Електронний ресурс] – <https://uk.javascript.info>
8. Node.js. Express Documentation [Електронний ресурс] – <https://expressjs.com>
9. Medium. Database Management [Електронний ресурс] – <https://medium.com/@ansh.magon19/databases-for-web-development-e8bcc9cf4f29>
10. Geekforgeeks. What is a web server Working and Architecture [Електронний ресурс] – <https://www.geeksforgeeks.org/how-web-works-web-application-architecture-for-beginners/>