

**МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
КИЇВСЬКИЙ НАЦІОНАЛЬНИЙ ЕКОНОМІЧНИЙ УНІВЕРСИТЕТ
ІМЕНІ ВАДИМА ГЕТЬМАНА
Навчально-науковий інститут
«Інститут інформаційних технологій в економіці»
Кафедра інформаційних систем в економіці**

ОСВІТНЬО ПРОФЕСІЙНА ПРОГРАМА «КОМП'ЮТЕРНІ НАУКИ»

галузь знань 12 «Інформаційні технології»

спеціальність 122 «Комп'ютерні науки»

Форма навчання: денна

КВАЛІФІКАЦІЙНИЙ БАКАЛАВРСЬКИЙ ПРОЕКТ

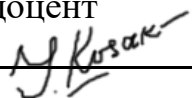
на тему

РОЗРОБКА БЕСТІАРІЮ НАСТІЛЬНОЇ ГРИ «D&D»

здобувача Єрмакова Іллі Євгеновича _____

Науковий керівник:

к.е.н., доцент

_____  Козак І.А.

**Проект допущений до захисту
перед екзаменаційною комісією з
атестації здобувачів вищої освіти
завідувач кафедри:**

к.е.н., доцент.

_____ Тішков Б.О.

Київ 2024

Міністерство освіти і науки України
Київський національний економічний університет імені Вадима Гетьмана
Навчально-науковий інститут «Інститут інформаційних технологій в економіці»
Кафедра інформаційних систем в економіці

ОСВІТНЬО_ПРОФЕСІЙНА ПРОГРАМА «КОМП'ЮТЕРНІ НАУКИ»

галузь знань 12 «Інформаційні технології»
спеціальність 122 «Комп'ютерні науки»

ПОГОДЖЕНО:

Керівник проектної групи(гарант)
освітньо-професійної програми

_____ Іванченко Г.Ф.
“ ____ ” _____ 2024 р.

ЗАТВЕРДЖУЮ:

Завідувач кафедри
_____ Тішков Б.О.

“ ____ ” _____ 2024 р.

ІНДИВІДУАЛЬНЕ ЗАВДАННЯ

здобувача вищої освіти ***Єрмакова Іллі Євгеновича***

очної (денної) форми навчання

на підготовку кваліфікаційного бакалаврського проекту
на тему: «Розробка Бестіарію настільної гри «D&D»»

Тему затверджено наказом ректора Університету від «11» березня 2024 р. № 529 ст.

Кваліфікаційний бакалаврський проект виконується на матеріалах
Аналізу предметної області та літературних джерел

План кваліфікаційного бакалаврського проекту

Розділ I _____ **Характеристика та аналіз предметної галузі** _____

Розділ II ____ **Розробка вимог і моделювання інформаційної системи** _____

Розділ III ____ **Проектування та реалізація компонентів підсистеми** _____

Об'єкт дослідження _____ **Юніт (істота) в настільній грі** _____

Предмет дослідження _____ **Організація роботи бестіарію** _____

Мета кваліфікаційного бакалаврського проекту _____ **створення**
повноцінної системи управління і створення істот на основі D&D5e

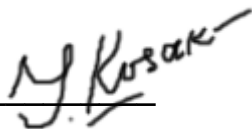
Конкретні завдання, які студент повинен виконати для досягнення поставленої мети:

У розділі I ____ Проаналізувати предметну галузь, обґрунтувати причини необхідності існування системи. Проаналізувати існуючі варіанти рішень, їх недоліки та переваги. _____

У розділі II ____ На основі рішень і інформації отриманої з першого розділу, сформулювати вимоги до системи. Змодельовати її структуру, алгоритми розв'язання. Сформулювати чітку постановку задачі. _____

У розділі III ____ Визначити інформаційне, технічне і програмне забезпечення. Виокремити вхідні та вихідні повідомлення а також структуру таблиць для даної системи. _____

Завдання підготував
науковий керівник _____



Козак Ірина Антонівна

“15”березня 2024 р.

Завдання одержав
студент _____

Єрмаков Ілля Євгенович

“15”березня 2024 р.

АНОТАЦІЯ

Кваліфікаційного бакалаврського проекту студента 4 курсу
Навчально-наукового інституту «Інститут інформаційних систем в
економіці» **Єрмакова Іллі Євгеновича**, виконаної на тему:
«Розробка Бестіарію настільної гри «D&D»»
Київ: кафедра інформаційних систем в економіці, 2024 р.

Кваліфікаційний бакалаврський проект присвячений актуальним потребам цифровізації інформації, перенесення даних з паперового в електронний вигляд. В даному випадку це стосується настільної гри Dungeons and Dragons, а точніше її частини, що пов'язана з ворогами у цій грі.

Кваліфікаційний бакалаврський проект містить 3 розділи, що показують логічний ланцюжок побудови алгоритмів проектування і розробки.

В першому розділі дана характеристика предметної галузі і об'єкта дослідження, а також проаналізовано аналоги системи, даної предметної галузі.

Другий розділ присвячено моделюванню і проектуванню системи. Аналізу і розробці її архітектури і структури, а також чітко окреслені вимоги і функції системи, для забезпечення розуміння готового продукту і послуг, які він надає.

Третій розділ присвячений конструктивним питанням. В ньому показано технічне, програмне і інформаційне забезпечення системи Бестіарію. Розроблена структура забезпечення, виокремлені вхідні та вихідні повідомлення. Також обґрунтовано і наведено перелік технічних засобів, необхідних для розробки.

Висновки містять рекомендації і тези, які зародились під час проектування і розробки системи. Також наведено перелік можливих покращень для даної системи.

РЕФЕРАТ

Кваліфікаційний бакалаврський проект містить 56 сторінок, 10 таблиць, 20 рисунків, список літератури з 5 найменувань, 3 додатків.

РОЗРОБКА БЕСТІАРІЮ НАСТІЛЬНОЇ ГРИ "D&D"

Перелік ключових слів: Бестіарій, система, істота, адміністрування, цифровізація, D&D5e

Предметом дослідження є організація роботи бестіарію

Об'єктом дослідження виступає юніт (істота) в настільній грі

Мета Кваліфікаційного бакалаврського проекту полягає в створенні повноцінної системи управління і створення істот на основі D&D5e

Завданнями кваліфікаційного бакалаврського проекту є дослідити методи цифровізації істот для їх простого пошуку і створення

Апаратні та програмні засоби, що використовувались на при проектуванні: ноутбук, ПК (сервер), PhpStorm, Figma, Github, Linux 20.04 LTS, Windows 11, Google Chrome, Trello

Результати досягнуті в процесі роботи (також їх новизна та ступінь упровадження): Отриманий перший україномовний бестіарій настільної рольової гри D&D

Одержані результати можуть бути використані (рекомендації щодо практичного застосування здобутих результатів, із зазначенням ефективності, сфери використання) для проведення настільних ігор і запису власних ідей монстрів для майбутніх ігор.

Рік виконання кваліфікаційного бакалаврського проекту 2024

Рік захисту кваліфікаційного бакалаврського проекту 2024

Зміст

ВСТУП	1
РОЗДІЛ 1. ХАРАКТЕРИСТИКА ТА АНАЛІЗ ПРЕДМЕТНОЇ ГАЛУЗІ.....	4
1.1 Характеристика предметної галузі та об'єкта дослідження	4
1.2 Аналіз літературних джерел та практичного досвіду використання ІС і технологій в предметній галузі.....	6
РОЗДІЛ 2. РОЗРОБКА ВИМОГ І МОДЕЛЮВАННЯ ІНФОРМАЦІЙНОЇ СИСТЕМИ	12
2.1 Аналіз і специфікація вимог до інформаційної системи.....	12
2.2. Постановка та алгоритм розв'язання задачі	14
2.2.1. Постановка задачі	14
2.2.2. Алгоритм розв'язання задачі.....	19
2.3. Моделювання інформаційної підсистеми.....	21
РОЗДІЛ 3. ПРОЕКТУВАННЯ ТА РЕАЛІЗАЦІЯ КОМПОНЕНТІВ ПІДСИСТЕМИ	30
3.1 Інформаційне забезпечення.....	30
3.2 Технічне забезпечення.....	36
3.3 Програмне забезпечення.....	39
3.4 Результати Реалізації Інформаційної підсистеми.	41
ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ.....	45
Додаток А Створення Істоти.....	2
Додаток В Головна сторінка	4
Додаток С Код Головного Модуля Backend.....	5

ВСТУП

З плином часу все переноситься на цифрові носії. Документацію, підрахунки і багато чого іншого. Інформацію значно зручніше і надійніше зберігати на цифрових носіях, адже це займає набагато менше місця і також дає змогу маніпулювати інформацією набагато оперативніше і швидше. Пошук необхідної інформації у книжці і в інтернеті відрізняються по швидкості в сотні разів, і це є безумовним плюсом комп'ютерів.

Цифровізація, як зазначалось вище, торкнулась багатьох сфер, і однією з них є ігри. Комп'ютерні розваги розвиваються семимильними кроками, з кожним роком дивуючи нас своїми можливостями. Але настільні ігри теж все ще існують і до сих пір користуються попитом.

Найяскравішим прикладом такої гри є Dungeon And Dragons (D&D). Що перекладається як «Підземелля та Дракони». В цій грі є майстер, і гравці. Майстер – творець світу, НІП, історії, моментів, атмосфери і всього що є навколо персонажів гравців. І особливо противників, так. І для того щоб майстер міг давати найзахопливіші бойові сутички і битви, для нього випущена окрема книга Monster Manual, яка на 300 сторінках виклала читачу сотні істот, що стануть перепорою в пригодах героїв.

Всіх цих істот можна відцифрувати, щоб дати змогу як мінімум швидко і зручно шукати.

Інформаційна система Бестіарію – містить відомості про ворогів, з яким гравці настільної гри можуть зіткнутись у своїх захопливих пригодах.

Актуальність ІС: створення «Бестіарію по настільній грі D&D» зумовлена тим, що сьогодні є зростає популярність західних настільних ігор.

Тому планується створити повноцінну систему для Бестіарію, де данні будуть зберігатись і та оновлюватись. Це значно покращить та зробить менш трудомісткою роботу ведучого гри. База даних буде служити як сховище даних. А front-end та back-end дадуть змогу відображати цю інформацію користувачу у зручному форматі.

Об'єктом дослідження у даній бакалаврській роботі є юніти в настільній грі. Предметом дослідження – організація роботи бестіарію.

Дана тема була вибрана через високі темпи зростання популярності даної настільної гри в Україні. Все більше і більше люди починають пізнавати західну культуру блага і розваги, а тому для досягнення успіху – необхідно чіплятись за те – що починає потроху розцвітати і де не так багато сформованої конкуренції. Також важливим чинником є моя власна зацікавленість у цій грі і бажанні просувати її в україномовному середовищі.

Метою даної роботи є створення інформаційної системи "Бестіарію", яка містить відомості про ворогів, з якими гравці настільної гри можуть зіткнутись у своїх захопливих пригодах. Ця система має забезпечити зручний і швидкий доступ до інформації про істот, що зробить гру більш динамічною і менш трудомісткою для ведучого.

Також було досліджено такі літературні джерела для формування розуміння актуальності цієї теми:

1. Wikipedia [Електронний ресурс] – Режим доступу до ресурсу: [https://uk.wikipedia.org/wiki/Dungeons %26 Dragons](https://uk.wikipedia.org/wiki/Dungeons_%26_Dragons)
2. Dungeon And Dragons у військовій розвідці [Електронний ресурс] – Режим доступу до ресурсу: _ <https://dnd.in.ua/articles/vidkryjte-v-sobi-botanika-dungeons-dragons-u-vijskovij-rozvidczy/>
3. Все що потрібно знати про Dungeon and Drag [Електронний ресурс] – Режим доступу до ресурсу: <https://desktopgames.com.ua/ua/article/vse-scho-treba-znati-pro-dungeon-and-dragons.html>
4. Monster Manual / Wizards of the Coast. — Renton, WA : Wizards of the Coast, 2014. — 352 p. — (Dungeons & Dragons).
5. Dungeon Master Guide / Wizards of the Coast. — Renton, WA : Wizards of the Coast, 2014. — 320 p. — (Dungeons & Dragons).

Для досягнення мети проекту було використано ряд методів дослідження. Значну роль відіграв емпіричний метод, що полягав у застосуванні мого власного досвіду гри у Dungeons & Dragons (D&D). Це

дозволило глибше зрозуміти потреби ведучих гри та ефективно перевіряти запропоновані рішення. Також був проведений аналіз існуючих аналогів інформаційних систем для D&D, що допомогло виявити їх сильні та слабкі сторони, а також визначити можливі покращення. Метод порівняльного аналізу дозволив вибрати оптимальні підходи до створення баз даних і організації інформаційної системи. Вербально-описовий метод використовувався для структурування та опису результатів дослідження.

Значущість цієї роботи полягає в тому, що при аналізі існуючих рішень – не було знайдено жодного україномовного варіанту. Створення бестіарію рідною мовою дозволить привернути увагу до нашої гри, адже в нинішніх реаліях для людей дуже важливо, щоб контент був українською.

У звіті буде показано аналіз існуючих рішень, розписано в схемах і текстом – основні кроки організації роботи і створення структури проекту, а також наведені результати роботи, разом з кодом, що відповідає за це.

РОЗДІЛ 1. ХАРАКТЕРИСТИКА ТА АНАЛІЗ ПРЕДМЕТНОЇ ГАЛУЗІ

1.1 Характеристика предметної галузі та об'єкта дослідження

У наш час все більше набирають популярність ігри, які через СРСР не могли потрапити до наших людей, а тому знайомство з ними почалось зовсім недавно. Однією з таких ігор є Dungeons and Dragons (D&D) — неймовірно захопливий проект, що дозволяє учасникам поринути у світ пригод, магії та секретів.

Розроблена в 80-ті роки настільна гра стала культовою і завоювала серця мільйонів людей по всьому світу. Wizards of The Coast можна вважати праотцями формування цілого культу мрійників і фантазерів.

Dungeons and Dragons (D&D) – це настільна рольова гра, у якій один з гравців виконує роль Майстра (Dungeon Master, DM) — творця всесвіту гри. Він вигадує локації, сюжет, а також керує всіма неігровими персонажами (NPC), що зустрічаються гравцям. Задача ж гравців — створити цікавих персонажів, які доповнюють світ і можуть впливати на нього.

Предметна галузь даного дослідження включає інформаційні системи та технології, які використовуються для організації та управління даними про істот у грі D&D. Це забезпечує зручний доступ до інформації про ворогів, з якими гравці можуть зіткнутись під час своїх пригод.

Об'єктом дослідження є процес створення та організації цифрового бестіарію для гри D&D, що включає перенесення даних з фізичної книги Monster Manual у цифрову базу даних.

Dungeons and Dragons є складною настільною грою, яка потребує ретельної підготовки та великого обсягу даних для ведучого гри. Основною особливістю цієї гри є необхідність швидкого доступу до інформації про різноманітних істот, що описані в Monster Manual. Традиційний паперовий формат ускладнює пошук і обробку інформації, що може займати багато часу.

Для полегшення цього процесу доцільно створити інформаційну систему, яка б дозволила швидко і ефективно знаходити потрібну інформацію про істот. Це значно спростить роботу ведучого гри, зменшить

час на підготовку та підвищить загальну ефективність проведення ігрових сесій.

Основні складові інформаційної системи:

1. База даних:

1.1.Зберігання інформації про всіх істот, включаючи їх характеристики, здібності та інші важливі дані.

1.2.Можливість швидкого пошуку і фільтрації даних за різними параметрами.

2. Front-end та back-end компоненти:

2.1.Інтерфейс користувача для зручного доступу до інформації.

2.2.Серверна частина для обробки запитів і управління даними.

Інформаційна система для бестіарію значно спрощує процес підготовки та ведення гри, забезпечуючи швидкий доступ до необхідної інформації та можливість оновлення даних у режимі реального часу. Це дозволяє ведучому гри зосередитися на створенні захоплюючих пригод і взаємодії з гравцями, а не на тривалому пошуку інформації в паперових книгах.

Опис рішень та особливостей:

1. Сутність проблемних ситуацій:

1.1.Тривалий час на пошук інформації у паперовому форматі.

1.2.Ризик втрати або пошкодження фізичних копій книг.

2. Цілі розв'язання:

2.1.Забезпечення швидкого та зручного доступу до даних.

2.2.Створення надійного сховища для інформації про істот.

3. Особи, що приймають рішення:

3.1.Ведучий гри (DM), який користується системою для підготовки та проведення ігрових сесій.

4. Фактори, що впливають на середовище прийняття рішень:

4.1.Зручність користування системою.

4.2.Швидкість доступу до даних.

4.3.Надійність і точність інформації.

Отже, створення цифрового бестіарію для гри Dungeons and Dragons є важливим кроком для покращення ефективності та зручності проведення ігор. Така система забезпечить надійне зберігання даних, швидкий доступ до необхідної інформації та зменшить трудомісткість підготовки ігрових сесій.

1.2 Аналіз літературних джерел та практичного досвіду використання ІС і технологій в предметній галузі

Оскільки ця настільна гра існує вже кілька десятиріч, в інтернеті є багато рішень. Давай розглянемо їх.

9

TTG.club - Це сайт, що представляє собою збірник всіх необхідних для гри знань. Сюди включені не лише Monster Manual (MM), на основі якої ми будемо робити наш сайт, але також Dungeon Master Guide (DMG) і Players Hand Book (PHB). Окрім цих офіційних книг, сайт підтримує багато різних «Homebrew», що написані на ентузіазмі гравців і дають майстрам і гравцям простіше поринати в різні атмосфери, «Одіссея Повелителів Драконів» дасть відчуття дух грецької міфології з магією і драконами, а «Керівництво гравця у похмурій лощині» - важке «Темне» фентезі, періоду середньовіччя. Головна сторінка сайту – рис 1.1

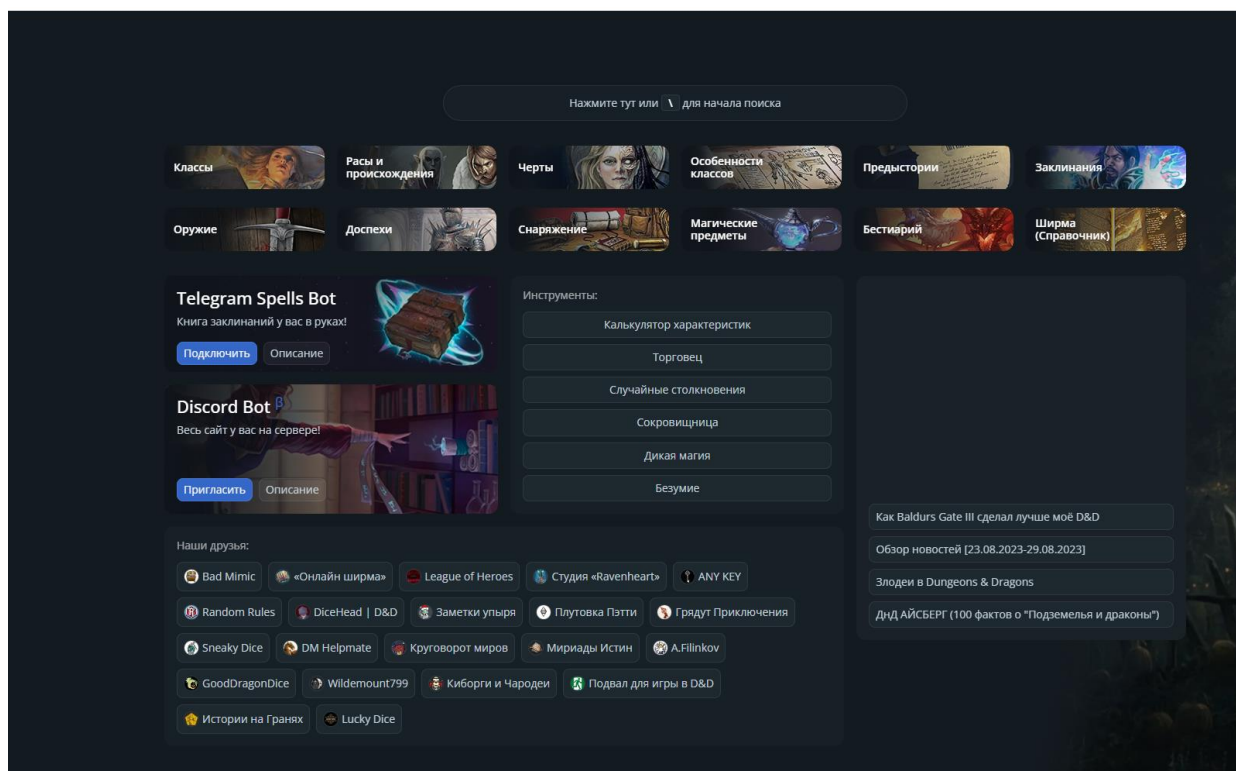


Рисунок 1.1 – Головна сторінка ttg.club

Перше що кидається в оформленні – це темна тема. Вона виглядає дуже сучасно і красиво, а також цей колір передає дух містики і таємниці, що допомагає додадковому понуренню в атмосферу, під час гри.

Розглянемо сторінку Бестиарію – рис 1.2 – 1.3

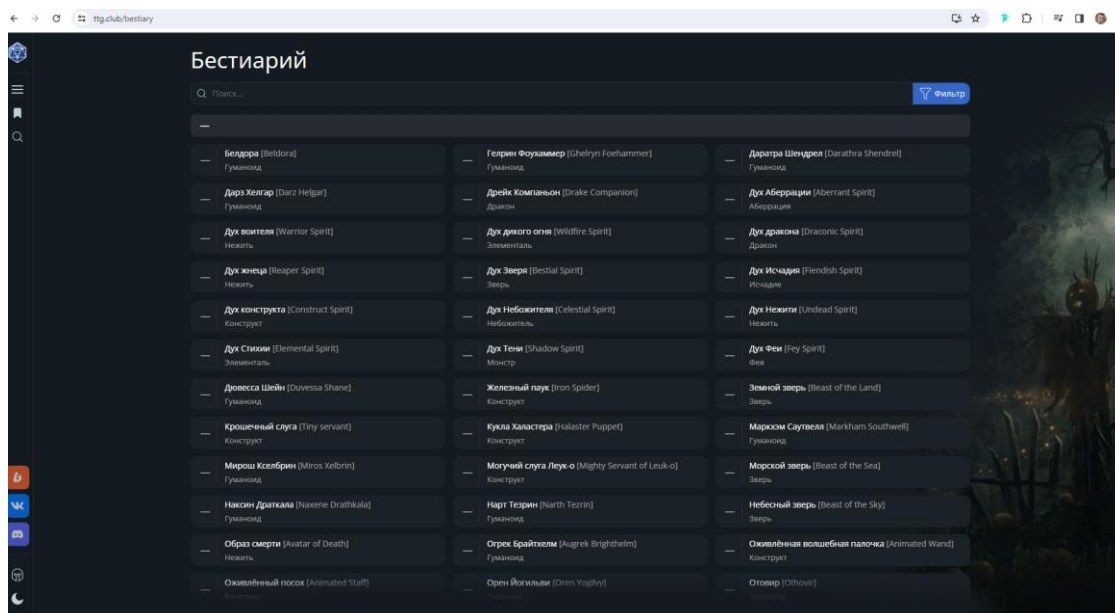


Рисунок. 1.2 – Сторінка Бестиарію

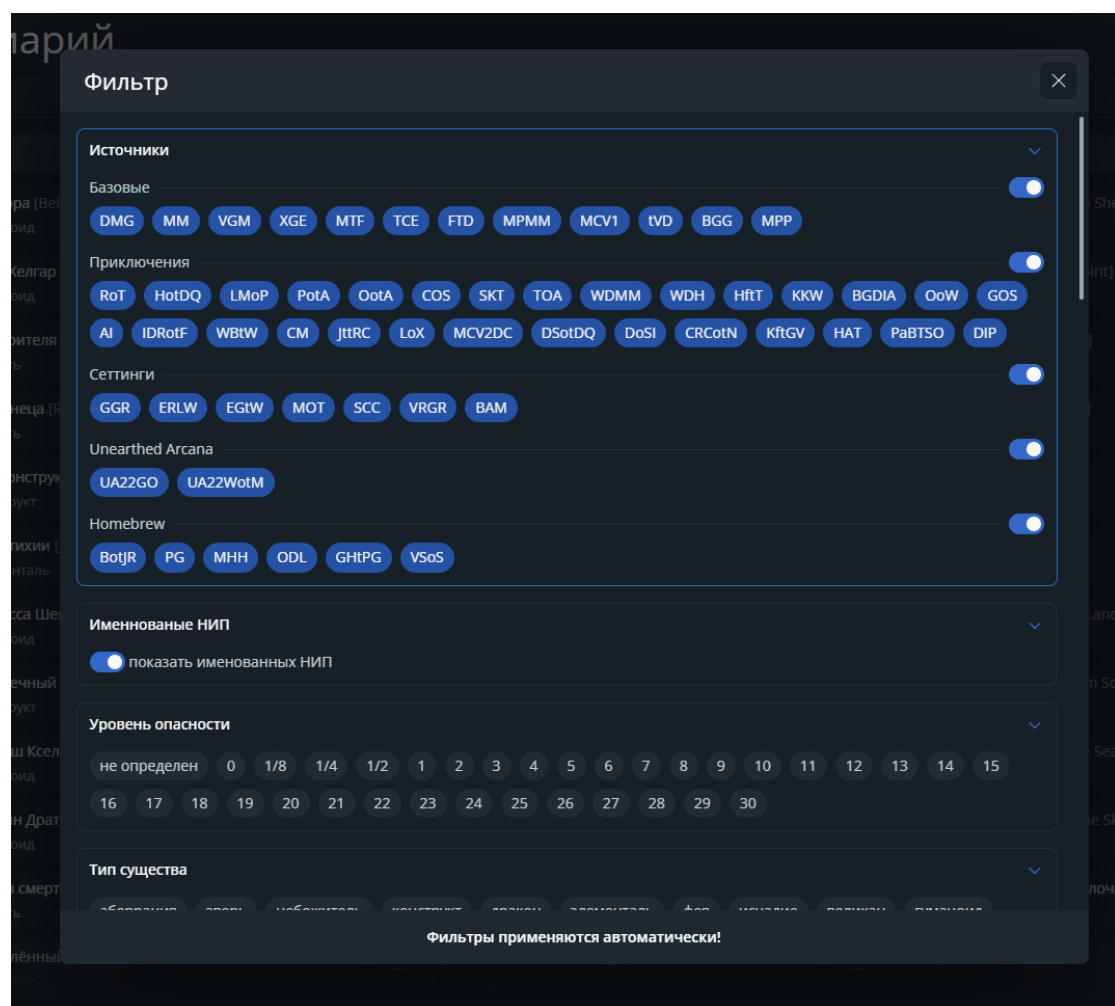


Рисунок. 1.3 – Фільтри

Дизайн містить в собі простоту і функціонал. Зручно переглядати великий обсяг істот, а також шукати конкретну за назвою, особливостями, і джерелом. Завдяки цьому забезпечується швидка доступність інформації, при її великому об'ємі.

Сторінка Істоти теж відповідає стандартам. Інформація добре виокремлена, основні записи розділені на секції і блоки, а мала, але важливі інформація по здібностях істоти – виділена окремим кольором. Сторінка істоти – Рис.1.4

Тараск [↗](#)
Tarrasque

Громадний монстр (титан), без мировоззрения / gargantuan 4x4 клетки или больше

Источник: ММ

Класс доспеха 25 (природный доспех)
Хиты 676 (33x20 + 330)
Скорость 40 фт.

СИЛ	ЛОВ	ТЕЛ	ИНТ	МДР	ХАР
30 (+10)	11 (+0)	30 (+10)	3 (-4)	11 (+0)	11 (+0)

Спасброски Инт +5, Мдр +9, Хар +9
Иммунитет к урону огонь, яд, дробящий, колющий и рубящий урон от немагических атак
Иммунитет к состояниям очарование, искуп, паралич, отравление
Чувства слепое зрение 120 фт., пассивная Внимательность 10
Языки —
Уровень опасности 30 (155 000 опыта)
Бонус мастерства 9

Легендарное сопротивление (3/день). Если тараск проваливает спасбросок, он может вместо этого сделать спасбросок успешным.
Сопротивление магии. Тараск совершает с **преимуществом** спасброски от заклинаний и прочих магических эффектов.
Отражающий панцирь. Каждый раз, когда тараск становится целью заклинания волшебная стрела [magic missile], заклинания с эффектом «линия», или заклинания, требующего бросок дальнбойной атаки, бросьте к6.
При результате от «1» до «5» тараск не попадает под эффект заклинания.
При результате «6» тараск не попадает под эффект заклинания, и эффект отражается в заклинателя, как если бы он исходил от тараска, делая целью заклинателя.
Осадное чудовище. Тараск причиняет двойной урон предметам и строениям.

Действия

Мультиатака. Тараск может использовать **Ужасающую внешность**. Затем он совершает пять атак: одну **Укусом**, две **Когтями**, одну **Рогами**, и одну **Хвостом**. Вместо укуса он может использовать **Проглатывание**.
Укус. Рукопашная атака оружием: +19 к попаданию, досягаемость 10 фт., одна цель. **Попадание:** Колющий урон 36 (4k12 + 10). Если цель — существо, она становится опутана, и тараск не может кусать другую цель.
Коготь. Рукопашная атака оружием: +19 к попаданию, досягаемость 15 фт., одна цель. **Попадание:** Рубящий урон 28 (4k8 + 10).

Рисунок 1.4 – Сторінка істоти.

Dungeons.fandom.com – теж гарний приклад подібного вікі, де міститься інформація для DM та гравців. Гарно відсортовані дані мають швидкий доступ до них, що забезпечує зручне користування. Рис. 1.6 – головна сторінка сайту.

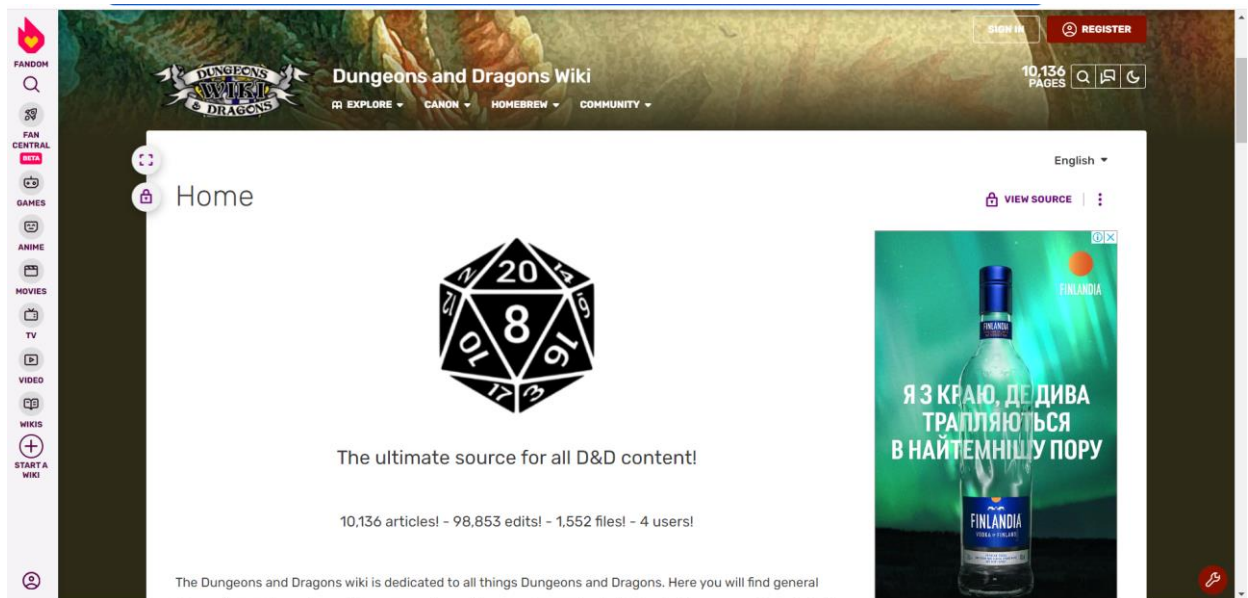


Рисунок. 1.5 – головна сторінка сайту

На відміну від попереднього прикладу, тут доступ до основних сторінок сайту знаходиться лише збоку, з однієї сторони – воно завжди під рукою, і є швидкий перехід між потрібними сторінками, з іншої – це інтуїтивно не зрозуміло, оскільки користувач в першу чергу дивиться на центр екрану, і знайти спосіб переходу між сторінками – займає час.

Пошук на сайті здійснюється по всім речам, що існують у грі, в той час як на ttg.club – він здійснюється по певним категоріям, це не є зручно, оскільки є багато пересікаючих одне-одного понять, що стосуються різних аспектів гри, але з іншої сторони, це не заставляє переключатись між сторінками для пошуку інформації. Пошук на сайті зображений на Рис. 1.7

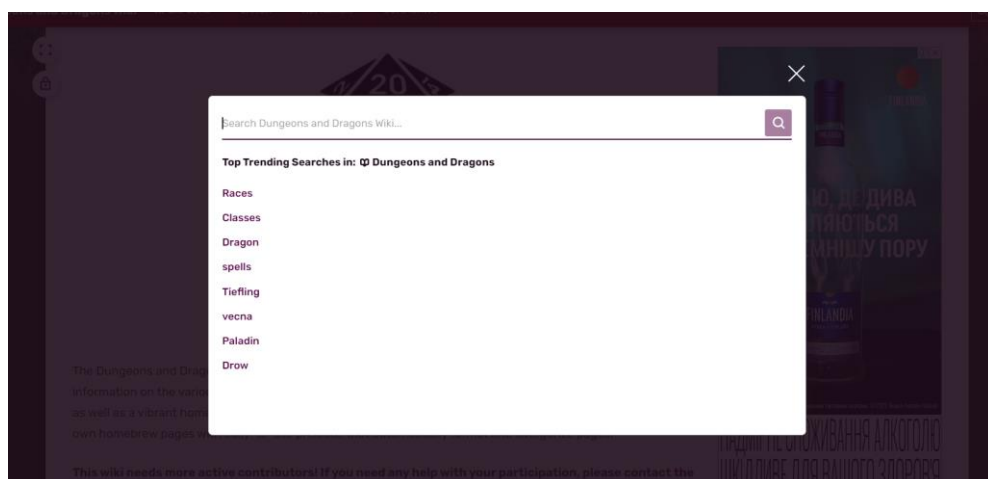


Рисунок. 1.6 – пошук на сайті dungeon.fandom

Сторінка Істоти явно уступає аналогу, оскільки тут вона виглядає дуже скудно, текст розділений набагато гірше, антураж вибиває з атмосфери, а основні

поняття не виділяються іншим кольором, а також велика кількість реклами, що псує загальні враження. Сторінка істоти зображена на Рис. 1.8

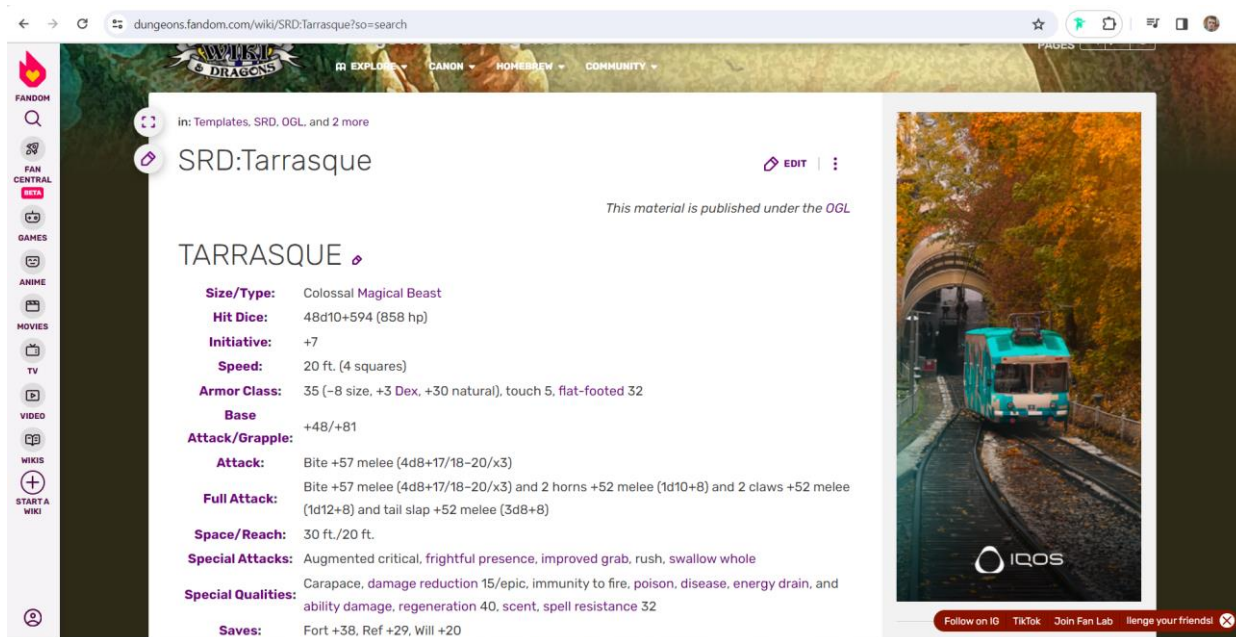


Рисунок. 1.7 – сторінка істоти dungeons.fandom.

Великим мінусом також є відсутність фотографій істоти, що ускладнює опис істоти майстром, аде немає наочного прикладу. Пошук хоч і швидкий, але немає додаткових фільтрів, через що, щоб знайти істоту – треба знати її найменування. Іноді це може бути складним для пошуку.

В розробці нашої інформаційної системи, ми маємо врахувати всі переваги та недоліки, що ми спостерігали на інших сайтах. Наш дизайн має бути привабливим і атмосферним, а функціонал – зручним і інтуїтивно зрозумілим. Для отримання ідеального балансу в цьому, треба проаналізувати сильні і слабкі сторони обох вищезгаданих прикладів. Щоб при розробці власної системи не увіткнутись в проблеми наших конкурентів – зробимо зведену таблицю, де визначимо плюси і мінуси по аспектах, що нас цікавлять. В Таблиці 1.1 показано зведену оцінку плюсів і мінусів кожної з системи.

Таблиця 1.1 - Зведена таблиця переваг і недоліків

Критерій	Ttg.club	Dungeon.fandom
Стилізація сайту для погруження в атмосферу	+	-
Пошук за ключовими словами	+	+
Налаштування фільтрів пошуку	+	-
Розділення інформації на окремі складові	+	-
Доступність для користувачів на безкоштовній основі	+	-
Можливість роздрукувати Істоту	+	-
Можливість реєстрації на сайті	-	+
Швидке переключеннями між різними сторінками сайту	-	+

РОЗДІЛ 2. РОЗРОБКА ВИМОГ І МОДЕЛЮВАННЯ ІНФОРМАЦІЙНОЇ СИСТЕМИ

2.1 Аналіз і специфікація вимог до інформаційної системи

При розробці системи Бестіарію, необхідно враховувати потреби користувачів у якості і зручності надання інформації, зручності для створення істот адмінами, також забезпечити адаптивність додатку і максимізувати охоплення аудиторії.

Бізнес-вимоги:

- **Спростити процес створення істот:** система має надавати адміну швидко заповнювати «повторювану інформацію» яка базується на основних переліках характеристик істоти і які при цьому обмежуються кількістю. Також слід надати адміну виділяти текст основних вмінь істоти, щоб покращити читабельність тексту;
- **Мінімізація часу на пошук істот:** система повинна надавати зручні інструменти для швидкого пошуку істот користувачем, за їх особливостями і назвою;
- **Адаптивність сайту:** важливо забезпечити можливість доступу до системи через планшети і мобільні пристрої, оскільки це буде основними інструментами для використання сайту;
- **Максимізація для охоплення більшої аудиторії:** система має бути доступна на двох мовах, для охоплення більшої аудиторії;
- **Забезпечити автентичний дизайн сайту:** система повинна виглядати органічно і не вибивати користувача із атмосфери фентезі.

Ці бізнес-вимоги є основними для розробки та впровадження системи, яка задовольнить потреби користувачів і адмінів Бестіарію.

Функціональні вимоги системи Бестіарію визначають ключові можливості, які має мати система для задоволення потреб користувачів. Нижче наведено детальний опис кожної функції, яка повинна бути реалізована для забезпечення ефективного функціонування системи та задоволення потреб різних зацікавлених сторін.

- **Авторизація і Автенифікація:** ця функція забезпечує можливість доступу до деяких особливостей і функціоналу сайту. Відповідає за безпеку і цілісність даних від дій пересічного користувача, обмежуючи його функціонал на відміну від адмінського;

- **Управління атрибутами істоти:** має забезпечити зручність в управлінні і виправленні помилок атрибутів істоти, а також легкість у створенні нових.

- **Робота з істотами:** має забезпечувати повноту функціоналу для внесення даних про істот, гнучкість і простоту у їх створенні. Відокремлення і обмежений огляд закінчених істот.

- **Доступність інформації про істоту:** Швидкий пошук істоти на основі тегів, атрибутів, назви. Можливість перегляду істоти на сайті, а також роздруківки листа.

- **Автентичний інтерфейс:** має бути виконаний у приємні кольоровій палатрі золотого і синього, для створення атмосфери «високий риск – висока нагорода» для більшого захоплення грою і сюжетом.

Нижче наведено нефункціональні вимоги до інформаційної системи Бестіарію. Ці вимоги встановлюються для забезпечення зручності системи, а також для задоволення потреб користувачів і адмінів.

- **Зручний спосіб реєстрації:** користувачам має бути надано легкий спосіб для реєстрації на платформі. Також необхідно створити можливість налаштування профілю в майбутньому.

- **Система фідбеку і зворотнього зв'язку:** система повинна надавати можливість зворотнього зв'язку для поповнення адмінського складу і відслідковування баг-репортів.

- **Система управління адмінським складом:** визначає хто може управляти адміністраторським складом. Видаляти і додавати адміністраторів.

- **Ефективний спосіб представлення інформації про істот:** має бути швидкий і надійний спосіб розділення інформації для представлення користувачу.

Ці нефункціональні вимоги спрямовані на забезпечення зручності та ефективності системи наведені на рис 2.1 - діаграмі вимог з поданнями ієрархії вимог.

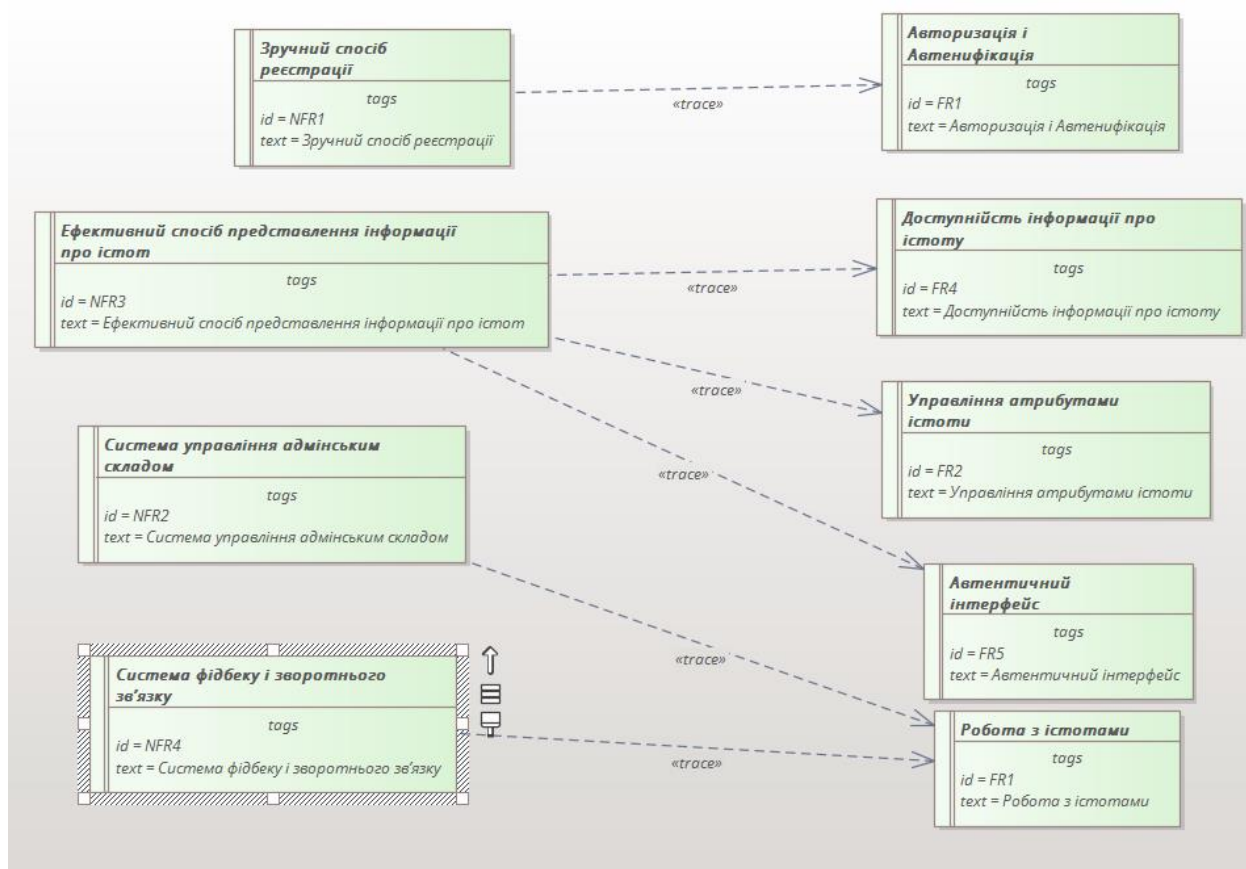


Рисунок. 2.1 – Діаграма вимог з поданням ієрархії вимог.

2.2. Постановка та алгоритм розв’язання задачі

2.2.1. Постановка задачі

Важливим кроком перед виконанням задачі – є її постановка. Чітке формулювання запиту – веде до найкращого результату. Ми вже сформулювали бізнес вимоги, функціональні вимоги і нефункціональні вимоги. Тепер сфокусуємося на чіткому формулюванні задачі, яку треба розв’язати для задоволення цих вимог.

Почнімо із характеристики задачі:

- **Призначення і техніко-економічна сутність** системи бестіарію полягає в допомозі майстру підземелля з швидким пошуком, а адміністраторам -створенням істот. Використання ЕОМ для швидкого сортування і знаходження інформації – є найвірнішим рішенням під час

розв'язання задачі, адже це дозволяє економити час, знижуючи його витрати на підготовку до гри.

- **Об'єктами управління в цій системі є:** Істоти (монстри, персонажі), їх атрибути і розмірності деяких з цих атрибутів, вміння і дії у бою, а також переклади для цих атрибутів і вмінь. Іншою частиною об'єктної структури є користувачі і їх ролі.

- **Призначення і Використання вихідної інформації:** Вихідна інформація використовується для зображення цілісного листа статистики, вмінь і дій конкретної істоти, не зважаючи на пересічність деяких характеристик з іншими об'єктами вихідної інформації, набір унікальних комбінацій цих атрибутів – формує унікальну істоту, яка буде використовуватись майстром для швидкого їх перегляду.

- **Періодичність розв'язання і термін видачі вихідної інформації:** Інформація буде видаватись постійно під час гри, або підготовки до неї, а термін видачі – буде миттєвим, після запиту користувача.

- **Умови, за яких припиняється автоматизоване розв'язання задачі:** Як тільки користувач перестає її запитувати, ври проблемах з під'єднанням користувача до мережі, або інших технічних проблемах зі сторони клієнту. Також при проблемах зі сторони серверу.

- **Зв'язками з іншими задачами можуть бути:** Генерація карт підземелля, створення записів для власної кампанії, додавання медіа плеєра, для музикальної атмосфери під час гри.

- **Розподіл дій між персоналом і технічними засобами за різних ситуацій розв'язання задачі:** Персоналом є Майстер підземелля, який шукає інформацію по потрібним йому істотам і керує їх набором під час гри, а також адміністратори – які створюють цих істот.

Інформаційна модель для Системи Бестіарію з настільної гри D&D наведена на рисунку 2.2.1

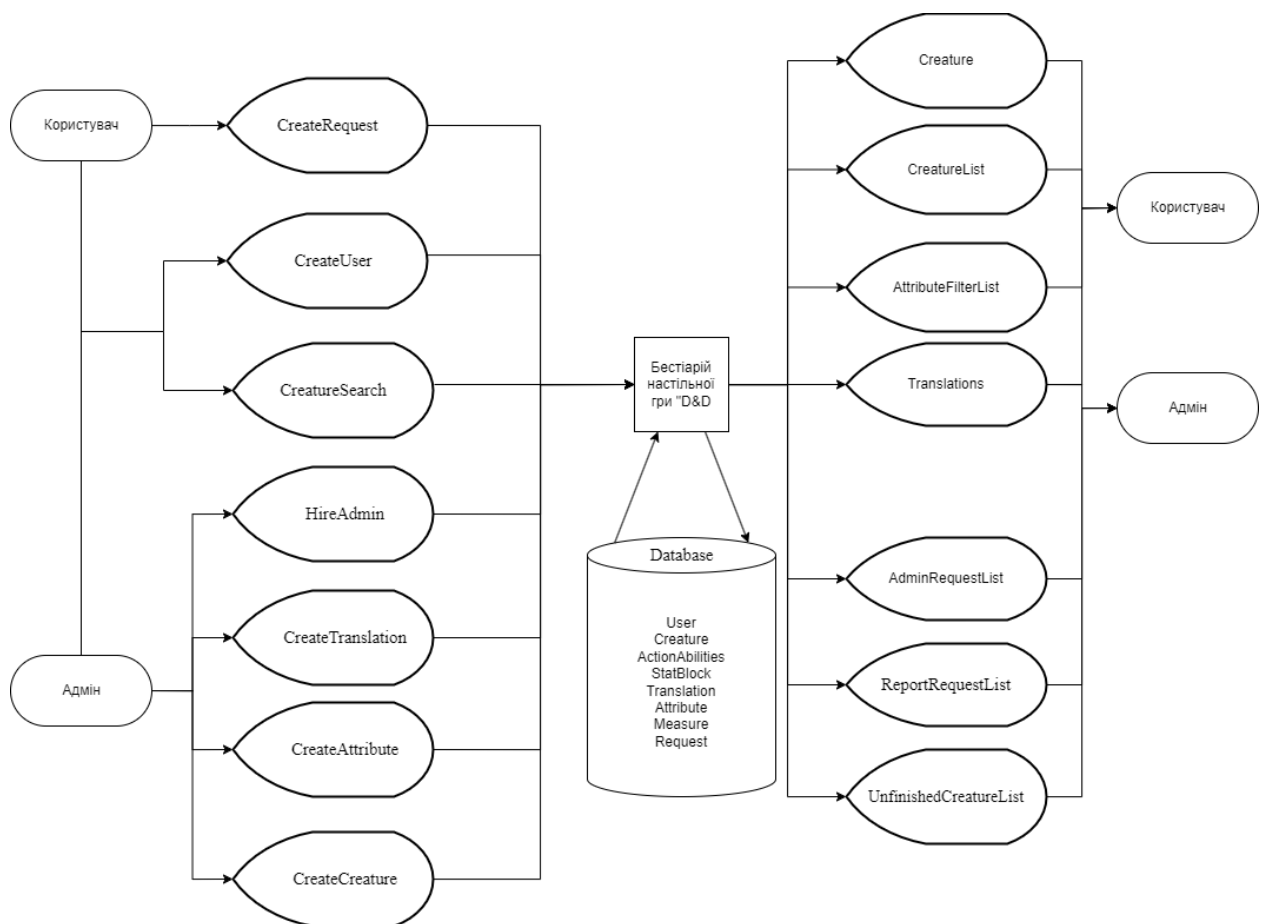


Рисунок 2.2.1 – Інформаційна модель Системи Бестиарію з настільної гри D&D

Тепер опишемо вихідну інформацію, тобто дані які генеруються системою після обробки вхідної інформації і надаються користувачу. Це можуть бути звіти, повідомлення, файли і сторінки.

В системі Бестиарію, вихідна інформація використовується для відображення характеристик, вмінь та атрибутів істот. Це допомагає майстру підземелля в управлінні грою та прийнятті рішень під час гри. В свою чергу майстер підземелля використовує цю інформацію для швидкого доступу до деталей про істот під час гри, а також для підготовки до майбутніх сесій гри. Вихідна інформація представлена у таблиці 2.2.1.

Таблиця 2.2.1 – Вихідна інформація Бестіарію

З/п	Назва вихідного повідомлення	Ідентифікатор	Форма подання і вимоги до неї	Періодичність видання	Термін видання і допустимий час затримки	Користувачі інформації
1	Інформація про істоту	Creature	Екранна форма. Повна відповідність аналогу із книги (якщо є)	Постійно	Негайно після запиту користувача, 2с	Користувач, Адмін
2	Список істот	CreatureList	Екранна форма. Повна відповідність фільтрам	Постійно	Негайно після запиту користувача, 3с	Користувач, Адмін
3	Список Атрибутів (фільтрів)	Attribute FilterList	Екранна форма. Повна відповідність категорії атрибутів	Постійно	Негайно після запиту користувача, 3с	Користувач, Адмін
4	Список перекладів	Translations	Екранні форми та роздруківки. Відповідність перекладів	Постійно	Негайно після запиту користувача, 4с	Користувач, Адмін
5	Список заявок на роль адміна	Admin RequestList	Екранна форма. Відповідність БД	Постійно	Негайно після запиту користувача, 2с	Адмін
6	Список репортів	Report RequestList	Екранна форма. Відповідність БД	Постійно	Негайно після запиту користувача, 2с	Адмін
7	Список незакінчених істот	UnfinishedCreatureList	Екранна форма. Відповідність БД, точність даних на момент закінчення попереднього редагування	Постійно	Негайно після запиту адміністратора, 3с	Адмін

Наступним кроком визначимо вхідну інформацію. Вона відповідає за те, як дані заносяться в БД і як вони там зберігаються. Перелік вхідної інформації подано в таблиці 2.2.2.

Таблиця 2.2.2 – Вхідна інформація Бестіарію

з/п	Назва вхідного повідомлення	Ідентифікатор	Форма подання	Термін і частота надходження	Джерело
1	Створення/редагування істоти	CreateCreature	Екранна форма (форма створення істоти)	При створенні/редагуванні істоти	Адмін
2	Створення Атрибута	CreateAttribute	Екранна форма (форма створення атрибута)	При створенні/редагуванні атрибута	Адмін
3	Створення / редагування перекладу	CreateTranslation	Екранна форма (будь-яка форма, де фігурує переклад)	Під час більшості дій зі створенням інформації	Адмін
4	Створення аккаунта	CreateUser	Екранна форма (форма реєстрації)	При реєстрації користувача	Адмін, користувач
5	Пошук істоти	CreatureSearch	Екранна форма (Форма пошуку і фільтрації істот)	При запиті на пошук істоти	Адмін, користувач
6	Створення запита до адміністрації.	CreateRequest	Екранна форма (форма зворотнього зв'язку)	При створенні форми зворотнього зв'язку	Користувач
7	Прийняття нового адміністратора	HireAdmin	Екранна форма (форма адміністрування запитів)	При перегляді вхідних повідомлень від користувачів	Адмін
8	Список Користувачів	User	Файл БД	При логіні, реєстрації і написання звернення	Користувач, адмін
9	Список Істот	Creature	Файл БД	При створенні, редагуванні або отриманні істоти	Користувач, адмін
10	Список Вмінь і дій істот	ActionAbilities	Файл БД	При створенні, редагуванні або отриманні істоти	Користувач, адмін
11	Список Блоків Характеристик	StatBlock	Файл БД	При створенні, редагуванні або отриманні істоти	Користувач, адмін
12	Список Перекладів	Translation	Файл БД	Завжди	Користувач, адмін
13	Список Атрибутів	Attribute	Файл БД	При створенні, редагуванні, отриманні атрибутів.	Користувач, адмін
14	Список вимірюваних Атрибутів	Measure	Файл БД	При до редагуванні або отриманні істоти	Користувач, адмін
15	Список звернення	Request	Файл БД	При отриманні списку звернень. При створенні нового	Адмін

Отже, ми визначили вхідну і вихідну інформацію для системи, що задовольняють наші цілі та вимоги.

2.2.2. Алгоритм розв'язання задачі.

Для опису алгоритмів розв'язання задачі, необхідно описати використовувану інформацію, результати розв'язання, математичний опис, і алгоритм розв'язання задачі на ЕОМ.

Для нашої системи бестіарію існують такі математичні моделі:

1. Пошук за назвою

а) Формула:

$$F(N) = \{C \in CreaturesData \mid C.name = N\} \quad (2.2.2)$$

б) Де $F(N)$ – Множина істот з назвою N

2. Пошук за атрибутом

а) Формула

$$F(A, V) = \{C \in CreaturesData \mid C.A = V\} \quad (2.2.2)$$

б) Де $F(A, V)$ – Множина істот, де атрибут A має значення V

3. Сортування істот за атрибутом

а) Формула

$$S(A) = sorted(CreaturesData, key = \lambda x: x.A) \quad (2.2.3)$$

б) Де $S(A)$ – відсортований список істот з атрибутом A

Алгоритм розв'язання задачі включатиме наступні кроки:

1. Створення користувача

а) Введення даних для реєстрації;

б) Перевірка чи є створений аккаунт з такою поштою;

в) Збереження у БД;

2. Подача заявки на адміністрування або зворотнього зв'язку (Для не адміна)

а) Заповнення Форми заявки;

б) Перевірка правильності заповнення;

в) Запис у БД;

3. Створення Істоти

а) Перевірка на права доступу (роль Адмін) для користувача;

б) Заповнення форми істоти;

- в) Валідація тексту, створеного в редакторі, через safeHTML
 - г) Вибір, чи закінчена істота;
 - д) Збереження результатів у БД;
4. Створення Атрибута
- а) Перевірка чи існує потрібний атрибут в системі;
 - б) Створення атрибута і перекладу;
 - в) Прив'язання характеристики істоти (якщо це можливо);
 - г) Запис Атрибута в БД
5. Пошук істот
- а) Введення назви та/або атрибутів
 - б) Відправка запиту на back-end
 - в) Пошук істот і відправка результатів на front-end
 - г) Показ інформації користувачу
6. Адмін панель
- а) Перевірка ролі користувача і відображення кнопки входу в адмін-панель
 - б) Підвантаження незакінчених істот, атрибутів і заявок
 - в) Редагування форми незакінченої істоти
 - г) Редагування атрибутів
 - д) Менеджмент повідомлень
 - Помітка повідомлень як прочитаних
 - Видалення повідомлень
 - Прийняття в адміністрацію за запитом (якщо форма помічена як заявка на адміністрування)
7. Відображення інформації про істоту
- а) Пошук істоти в БД
 - б) Відправка користувачу, разом з прив'язаними атрибутами
 - в) Парсинг тексту на вміст спеціального коду позначень і переведення його в HTML для користувача

Алгоритм роботи системи зображений у вигляді схеми, що зображена на рисунку 2.2.2

Діаграма прецедентів системи Бестіарію описує різні дії, які можуть виконувати користувачі та адміністратори.

Користувачі системи мають можливість виконувати такі дії:

- реєстрація: цей прецедент дозволяє користувачам зареєструватись для входу в систему;
- Вхід до акаунту: цей прецедент дозволяє користувачам що вже зареєстровані – повторно увійти в систему зі старими кредами.
- Перегляд істот: дозволяє користувачам дивитись інформацію про істот.
- Роздруківка істот: Дозволяє роздрукувати переглянуту інформацію на папері.
- Подача заявки до адмінів: дозволяє отримати можливість зареєстрованому користувачу зв'язатись з адміністрацією
- Використання фільтрів: Сприяє кращому пошуку істот.

Адміністратори системи, окрім вище перелічених і крім подачі заявки на адміністрування, мають ще й такі можливості:

- Створення істоти: Додавання нової істоти у загальний список;
- Створення атрибутів істоти: дозволяє створити одну з основних структурних часток істоти;
- Управління заявками, атрибутами та істотами: дає можливість адміністрації ефективно здійснювати обслуговування системи.

Діаграма прецедентів представлена на рисунку 2.3

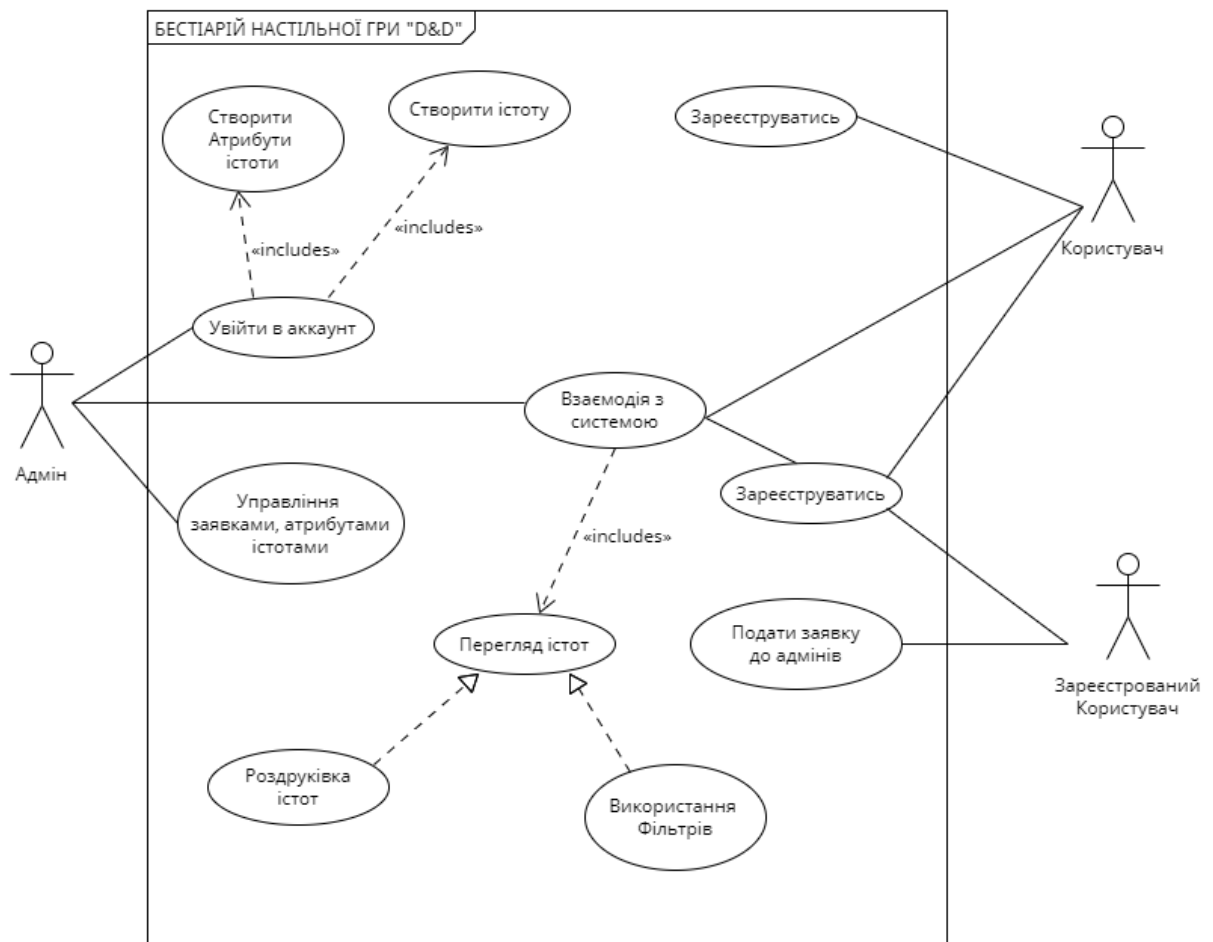


Рисунок 2.3 – Діаграма прецедентів для

Далі будуть наведені діаграми послідовності для нашої системи:

1. Користувач заповнює форму входу до аккаунта:

Нормальний сценарій:

- 1) користувач вводить свої дані в форму;
- 2) після завершення введення даних, система присилає сесійний токен;
- 3) після успішної операції система повідомляє користувача про закінчення операції та надає доступи.

Альтернативний сценарій:

- якщо виникають проблеми у формі, користувач отримує повідомлення про помилку. Він має можливість спробувати знову пізніше. Діаграма послідовності прецеденту «Вхід до аккаунта» представлена на рис. 2.4.

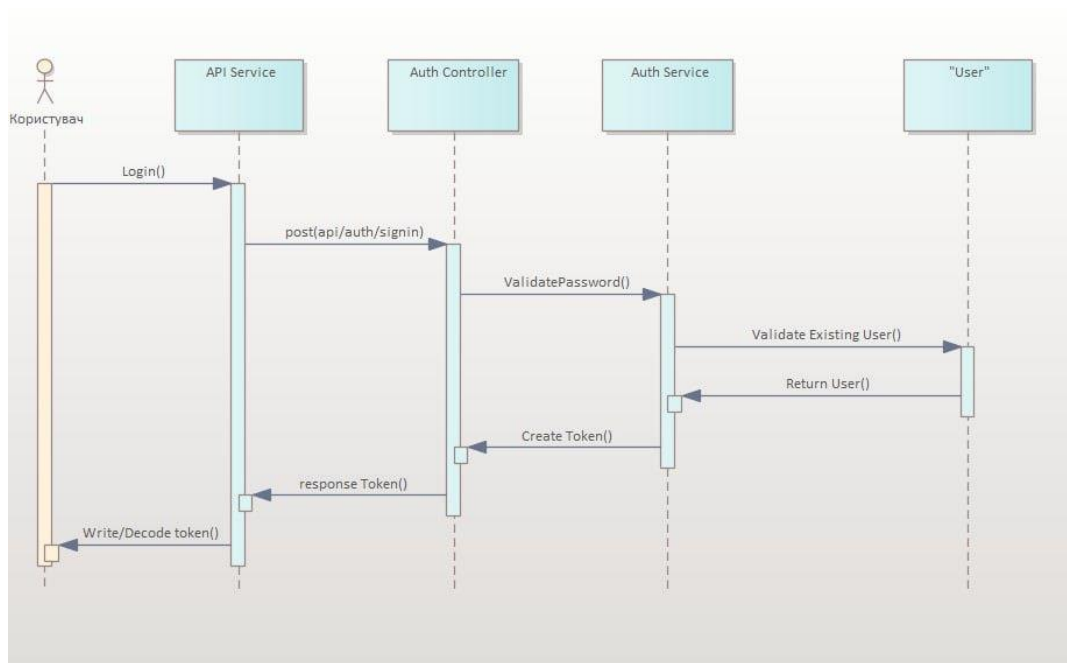


Рисунок 2.4 – Діаграма послідовності прецеденту «Вхід до аккаунта»

2. Користувач надсилає запит на отримання даних:

Нормальний сценарій:

- 1) Користувач обирає параметри для пошуку істоти;
- 2) Сервер перевіряє інформацію і дозволи
- 3) Сервер повертає шукану інформацію

Альтернативний сценарій:

- Система не знаходить інформацію по запити;
- Доступи користувача – не відповідають рівню доступу до інформацію;

Діаграма послідовності прецеденту «отримання даних для відображення» - представлена на рис 2.5.

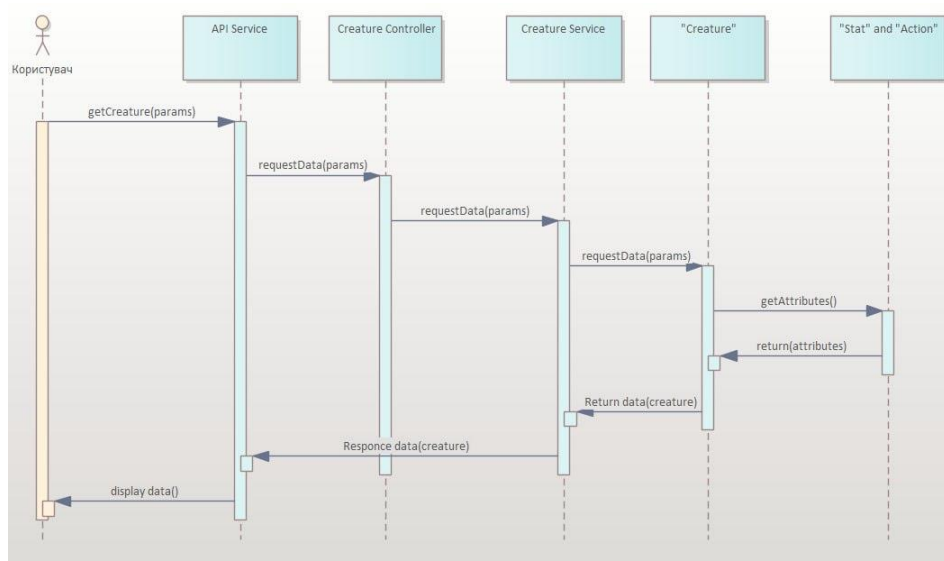


Рисунок 2.5 – Діаграма послідовності прецеденту «отримання даних для відображення»

3. Користувач заходить на форму:

Нормальний сценарій:

- 1) Користувач обирає пункт «заява на адміністрування» і заповнює даними лист;
- 2) Система обробляє лист і повідомляє адміна.
- 3) Адмін змінює роль користувача з заявки
- 4) Користувачу відкриваються відповідні зміни і доступи

Альтернативний сценарій:

- Користувачу відмовляють у праві на адміністрування.

Діаграма послідовності прецеденту «подача запиту на права адміністратора» - зображена на рис. 2.6.

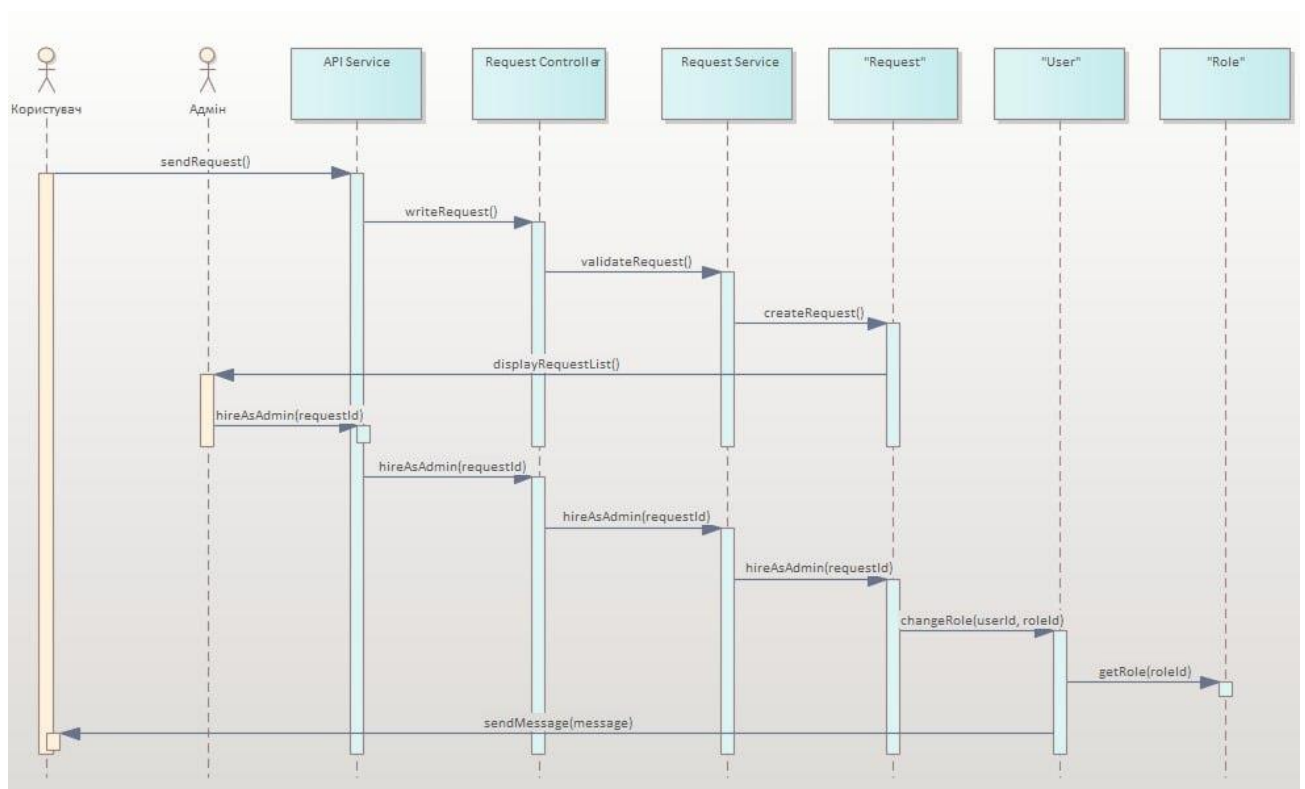


Рисунок 2.6 – Діаграма послідовності прецеденту «подача запиту на права адміністратора»

4. Користувач оновлює дані істоти:

Нормальний сценарій:

- 1) Відправлення даних на сервер;
- 2) Обробка нових даних. Редагування старих і додавання нових;

3) Після завершення редагування, система оновлює дані і надсилає звіт.

Альтернативний сценарій:

- У користувача недостатньо прав для редагування;
- Дані введені некоректно.

Діаграма послідовності прецеденту «Редагування веб-сторінок» - показана на рис 2.7

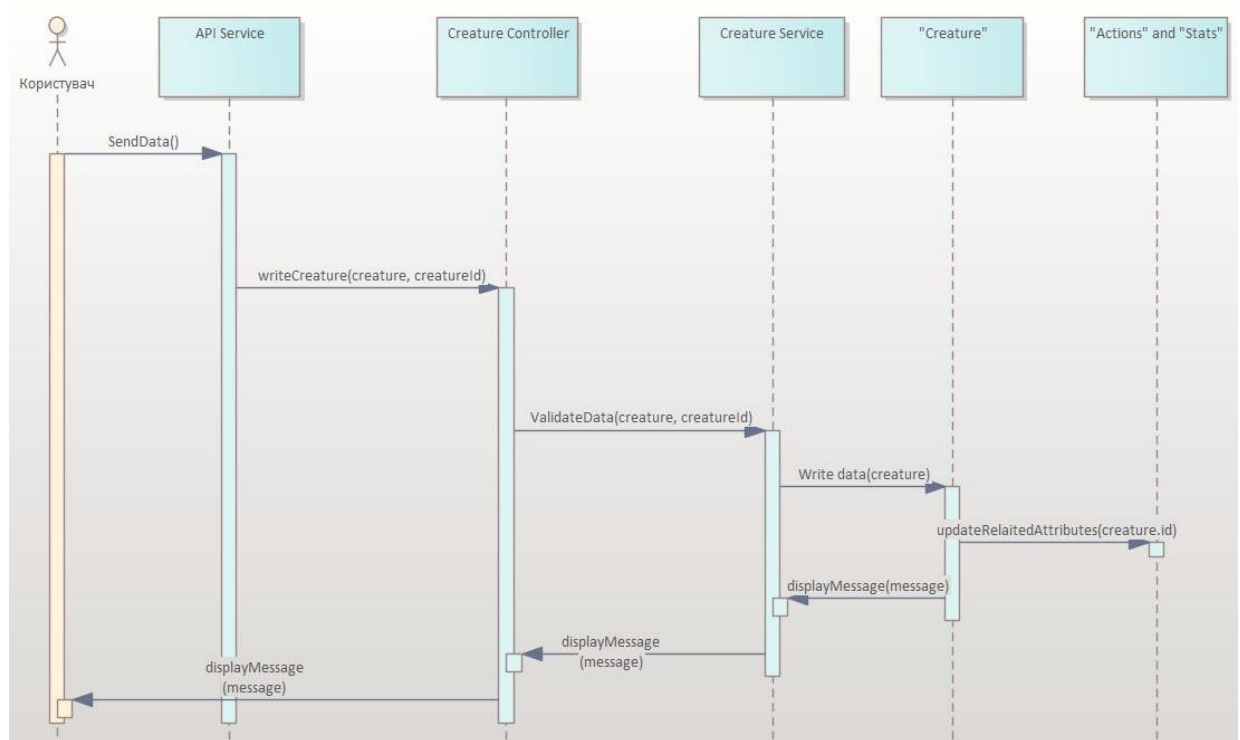


Рисунок 2.7 – Діаграма послідовності прецеденту «Редагування веб-сторінок»

Таким чином, для кожного прецеденту визначено «нормальний» сценарій використання, а також альтернативний сценарій, який описує можливі відхилення

На сайті Бестіарію, користувачі мають доступ до багатьох сторінок. На сторінці «Головна» користувач має швидкий доступ до різних роутів сайту. Перейшовши на вкладку Бестіарій – їм відкриється доступ до перегляду істот, а також налаштування фільтрів, для їх пошуку. На Сторінці «Логін» користувачі зможуть зареєструватися або авторизуватися. Зареєстрованих користувачів – адмін зможе зробити модераторами, які теж будуть створювати істот. На Сторінці «Редагування» адміністраторам буде надана можливість створювати і редагувати. Рисунок 2.8 – діаграма Діяльності.

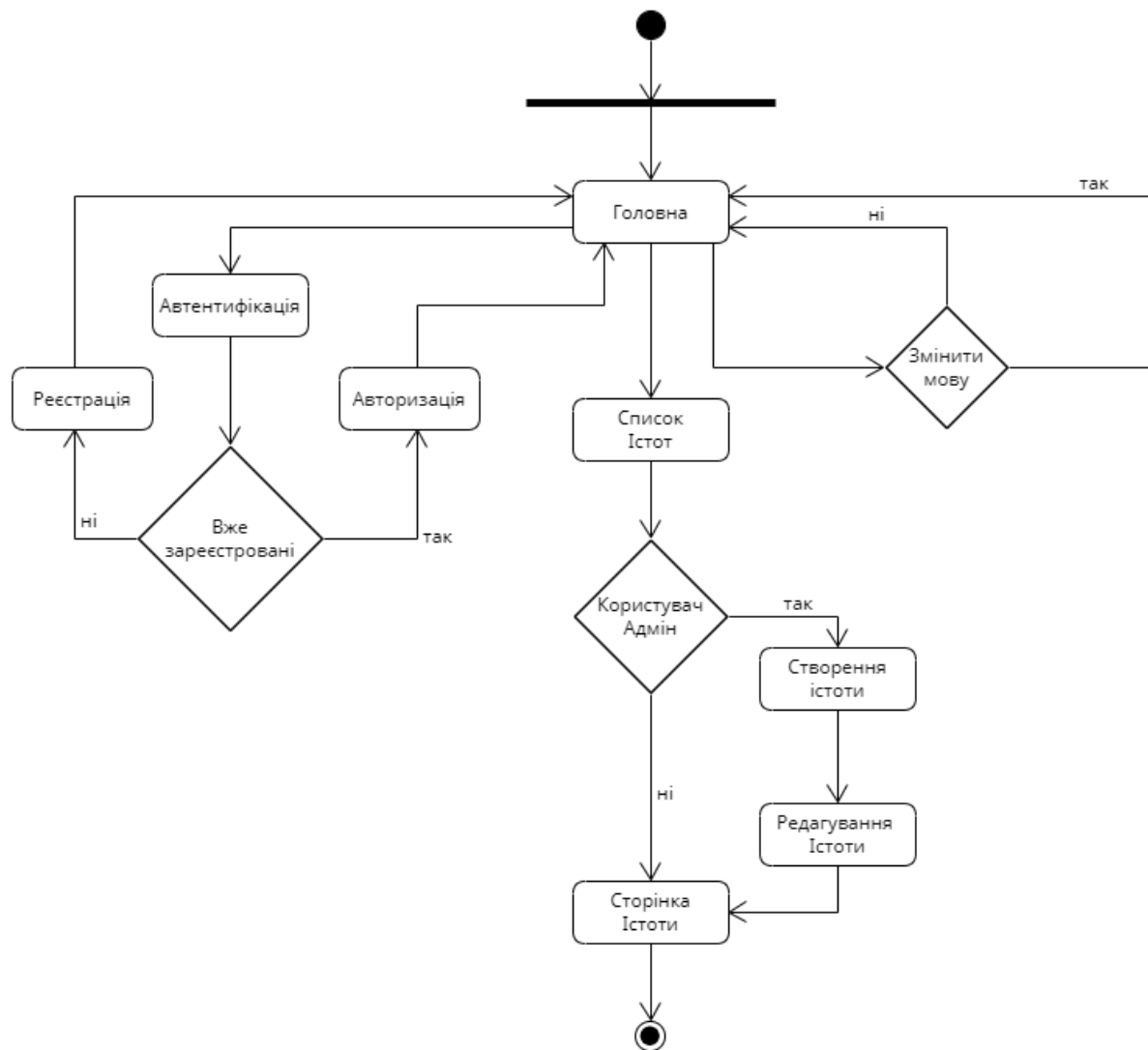


Рисунок 2.8 –Діаграма «Діяльності»

Для представлення класів системи і БД ми використаємо діаграму класів, на ній буде представлені головний модуль, що зв'язує окремі блоки коду у собі, в одну систему.

Головним модулем виступатиме Bestiary Module, в ньому будуть з'єднані Контролери, які можуть існувати по окремоті, але для системи ми опишемо їх в одному масиві, також тут будуть сервіси, по аналогії з контролерами. І Сутності, не зважаючи на відокремленість таблиць користувача і таблиць Істоти – це буде одна БД.

Контролери описуватимуть роути, до яких буде звертатись фронт, Гварди і інтерсептори які перехоплюватимуть вхідні запити і аналізуватимуть чи можна його пропустити далі і які методи будуть відпрацьовувати при активації цього шляху.

Сервіси і Сутності відповідатимуть за структуру і наповнення БД.
Діаграма представлена на рисунку 2.9.

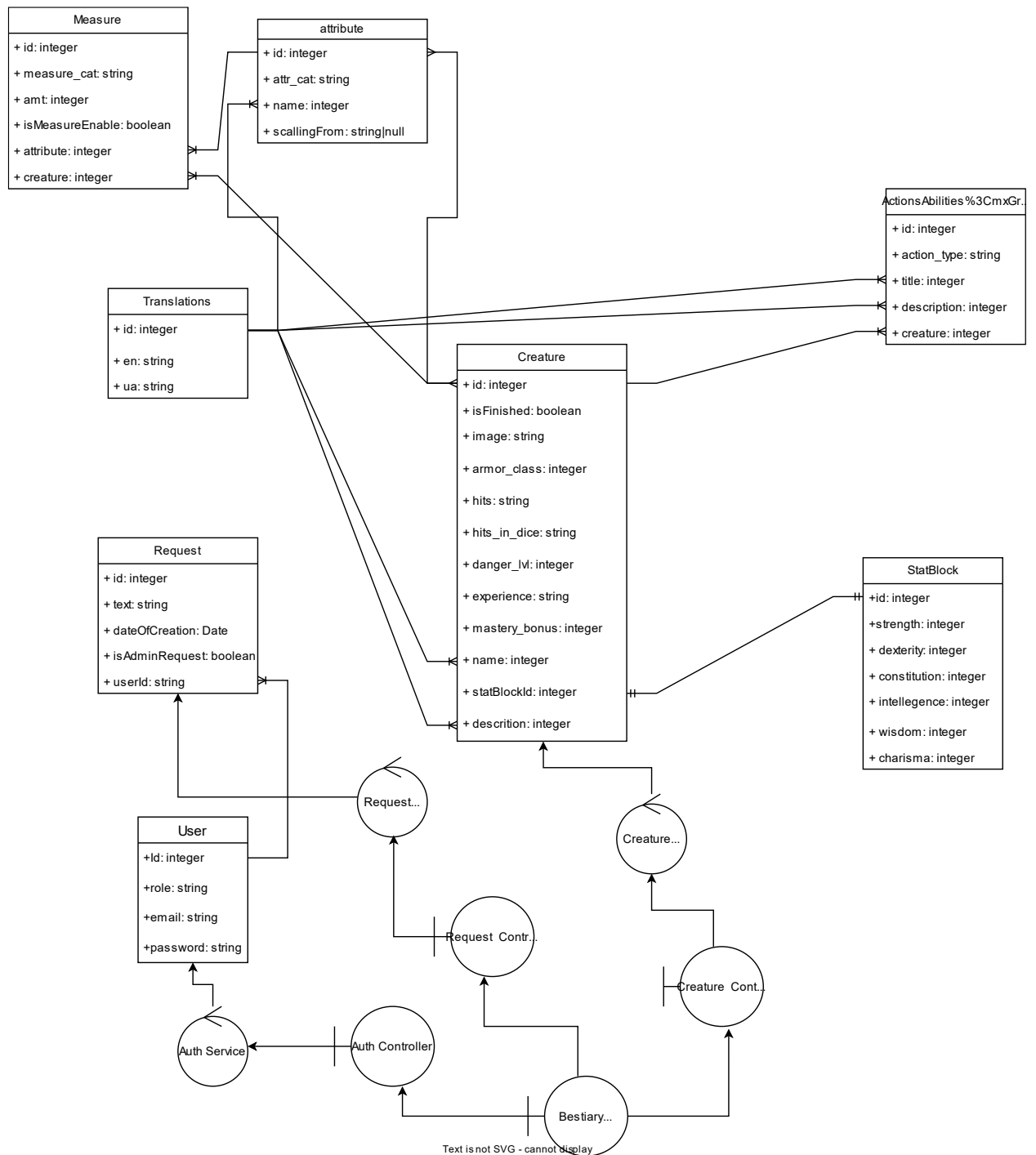


Рисунок 2.9 – Діаграма Класів

Останнім кроком у моделюванні інформаційної системи – стане трасування, завдяки якій ми поглянемо на систему з більшого масштабу. Діаграма Трасування представлена на рисунку 2.10.

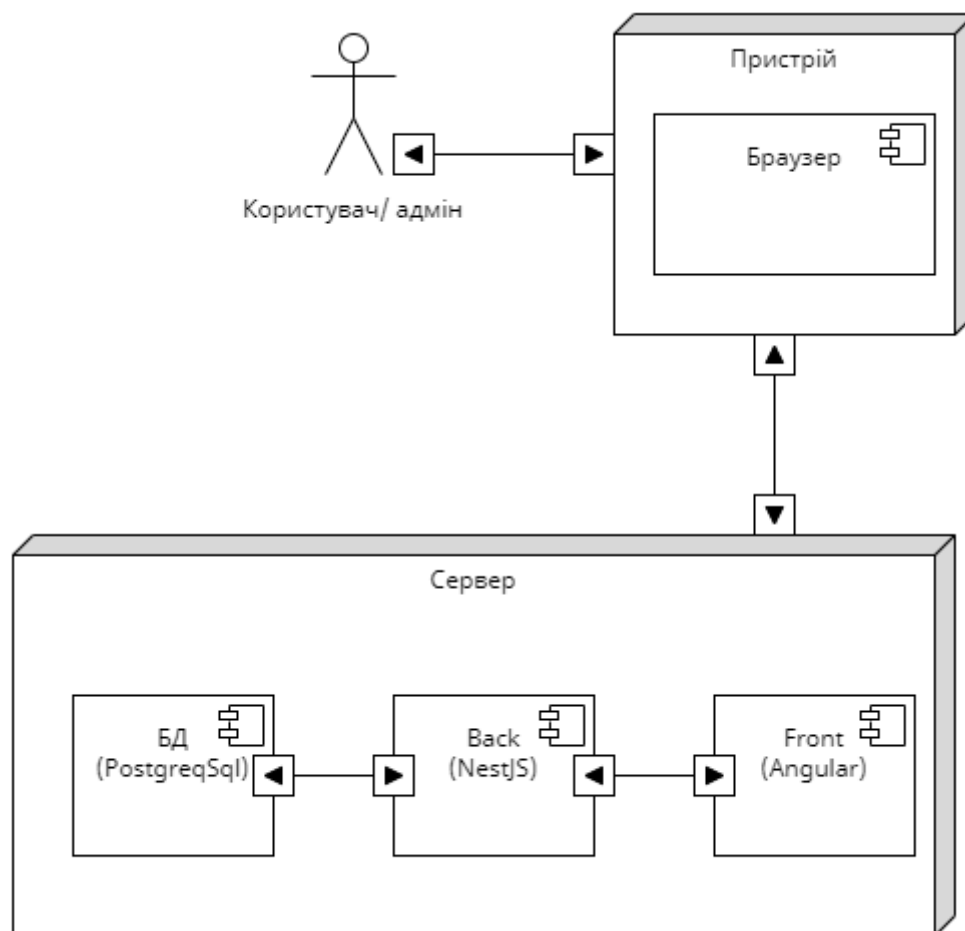


Рисунок 2.10 –Діаграма Трасування

РОЗДІЛ 3. ПРОЕКТУВАННЯ ТА РЕАЛІЗАЦІЯ КОМПОНЕНТІВ ПІДСИСТЕМИ

3.1 Інформаційне забезпечення

Система бестіарію є інформаційно незалежною від даних із зовні, тобто для користування системою – не потрібно мати сторонніх ресурсів. Істоти створюються власними силами і обмежені системою D&D5e, і хоч по ідеї адміністратор має розуміти, як вона функціонує, а значить прочитати хоча б одну книжку правил. Він не зможе вийти за її рамки, як би він не хотів цього зробити.

До складу ІЗ входить лише База Даних, яка в свою чергу поділяється на таблиці, що були детально розписані в минулих пунктах. Також ним є наступні книжки: «Dungeon Master Guide», «Monster Manual», «Player Handbook», Які потрібні для розуміння для чого створена система і як нею користуватись. А для адмінів окремо виділю «Monster Manual» оскільки там містяться початкова інформація про істот.

Структура ІЗ:

- Забезпечення цілісності та достовірності даних
- Регулярне резервне копіювання
- Контроль доступу і аутентифікація користувачів

Тип носія даних: Реляційна база даних на основі PostgreSQL.

Обґрунтування вибору СКБД:

- PostgreSQL: Вибір PostgreSQL обумовлений її відкритим кодом, широкою підтримкою спільноти, наявністю великої кількості інструментів для адміністрування та масштабування, а також здатністю ефективно обробляти великі обсяги даних.

Методи контролю інформації:

- Валідація введених даних на рівні додатка
- Перевірка цілісності даних на рівні бази даних
- Регулярні перевірки на дублювання та некоректні дані

Вимоги до надійності і достовірності інформації:

- Високий рівень доступності даних
- Мінімальний час відновлення у випадку збою
- Захист даних від несанкціонованого доступу

Загальна схема ІЗ:

- Користувачі вводять дані через веб-інтерфейс (Angular, SCSS, Tailwind)
- Дані зберігаються у реляційній базі даних (PostgreSQL)
- Автоматичні процеси забезпечують обробку і збереження даних (Nest, TypeORM)
- Звіти і результати доступні для перегляду і експорту

Методи контролю інформації:

- Валідація введених даних на рівні додатка
- Перевірка цілісності даних на рівні бази даних
- Регулярні перевірки на дублювання та некоректні дані

Вимоги до надійності і достовірності інформації:

- Високий рівень доступності даних
- Мінімальний час відновлення у випадку збою
- Захист даних від несанкціонованого доступу

Загальна схема ІЗ:

- Користувачі вводять дані через веб-інтерфейс (Angular, SCSS, Tailwind)
- Дані зберігаються у реляційній базі даних (PostgreSQL)
- Автоматичні процеси забезпечують обробку і збереження даних (Nest, TypeORM)
- Звіти і результати доступні для перегляду і експорту

Інформаційне забезпечення Бестіарію – показане в таблицях 3.1 – 3.8. там будуть показані поля для кожної окремої таблиці в БД

Таблиця 3.1 – Інформаційне забезпечення Бестіарію «Таблиця User»

Найменування	Ідентифікатор у програмі	Умове позначення у формулах	Формат	Бізнес-правила				Логічні чи семантичні зв'язки
				Первинний / вторинний ключ	Умова на значення	Обов'язкове поле	Індексне поле	
Ідентифікатор	id	id	number	PK	Id > 0	так	так	З таблицею Requests
Роль	role	Role	String	-	Admin / User values	Так	Hi	-
Емейл	email	Email	String	-	Email	Так	Hi	-
Пароль	password	Password	String	-	Length > 8	Так	Hi	-

Таблиця 3.2 – Інформаційне забезпечення Бестіарію «Таблиця Translations»

Найменування	Ідентифікатор у програмі	Умове позначення у формулах	Формат	Бізнес-правила				Логічні чи семантичні зв'язки
				Первинний / вторинний ключ	Умова на значення	Обов'язкове поле	Індексне поле	
Ідентифікатор	id	id	number	PK	Id > 0	так	так	Creature, attribute, Action_abilities,
Англомовний переклад	En	en	String	-	-	Hi	Hi	-
Україномовний переклад	ua	Ua	String	-	-	hi	Hi	-

Таблиця 3.3 – Інформаційне забезпечення Бестіарію «Таблиця Stat-block»

Найменування	Ідентифікатор у програмі	Умове позначення у формулах	Формат	Бізнес-правила				Логічні чи семантичні зв'язки
				Первинний / вторинний ключ	Умова на значення	Обов'язкове поле	Індексне поле	
Ідентифікатор	id	id	number	PK	Id > 0	так	так	Creature
Сила	Strength	Str	number	-	>0, <30	Hi	Hi	-
Спритність	Dexterity	Dex	number	-	>0, <30	hi	Hi	-
Тілобудова	Constitution	Con	number	-	>0, <30	Hi	hi	-
Інтелект	Intelligence	Int	number	-	>0, <30	Hi	Hi	-
Мудрість	Wisdom	Wis	number	-	>0, <30	Hi	Hi	-
Харизма	Charisma	chr	number	-	>0, <30	hi	Hi	-

Таблиця 3.4 – Інформаційне забезпечення Бестіарію «Таблиця Creature»

Найменування	Ідентифікатор у програмі	Умове позначення у формулах	Формат	Бізнес-правила				Логічні чи семантичні зв'язки
				Первинний / вторинний ключ	Умова на значення	Обов'язкове поле	Індексне поле	
Ідентифікатор	id	id	Number	PK	Id > 0	так	так	-
Чи закінчене	isFinished	isFinished	Boolean	-	-	Так	Hi	-
Картинка	Image	Image	String	-	-	Hi	Hi	-
Клас обладунку	Armor_class	Armor_class	Number	-	> 0	Hi	Hi	-
Хіти	Hits	Hits	Number	-	> 0	Hi	Hi	-
Хіти в кістках	Hits_in_dice	Hits_in_dice	String	-	-	Hi	Hi	-
Р-нь небезпеки	Danger_lvl	Danger_lvl	Number	-	> 0	Hi	Hi	-
Досвід	Experience	Experience	String	-	> 0	Hi	Hi	-
Бонус майстерності	Mastery_bonus	Mastery_bonus	Number	-	> 2	Hi	Hi	-
Ідентифікатор імені	nameId	nameId	Number	FK	> 0	Hi	Так	Translations
Ідентифікатор статблоку	StatBlockId	StatBlockId	Number	FK	> 0	Hi	Так	Statblock
Ідентифікатор опису	descriptionId	descriptionId	Number	FK	> 0	Hi	Так	Translations

Таблиця 3.5 – Інформаційне забезпечення Бестіарію «Таблиця Request»

Найменування	Ідентифікатор у програмі	Умове позначення у формулах	Формат	Бізнес-правила				Логічні чи семантичні зв'язки
				Первинний / вторинний ключ	Умова на значення	Обов'язкове поле	Індексне поле	
Ідентифікатор	id	id	Number	PK	Id > 0	так	так	Creature
текст	text	text	string	-	-	Так	Hi	-
Чи Прочитано	isRead	isRead	Boolean	-	-	так	Hi	-
Дата створення	dateOfCreation	dateOfCreation	Date	-	-	Так	ні	-
Запит на Адміністрування	isAdminRequest	isAdminRequest	Boolean	-	-	Так	Hi	-
Ідентифікатор користувача	userId	userId	Number	FK	> 0	Так	Так	User

Таблиця 3.6 – Інформаційне забезпечення Бестіарію «Таблиця Measure»

Найменування	Ідентифікатор у програмі	Умове позначення у формулах	Формат	Бізнес-правила				Логічні чи семантичні зв'язки
				Первинний / вторинний ключ	Умова на значення	Обов'язкове поле	Індексне поле	
Ідентифікатор	id	id	Number	PK	Id > 0	так	так	-
Категорія розмірності	Measure_cat	Measure_cat	String	-	-	так	Hi	-
Кількість	Amt	Amt	Number	-	> 0	Hi	Hi	-
Чи включена розмірність	isMeasureEnable	isMeasureEnable	Boolean	-	-	Hi	Hi	-
Ідентифікатор Атрибути	attributeId	attributeId	Number	FK	> 0	Так	Так	Attribute
Ідентифікатор істоти	CreatureId	CreatureId	Number	FK	> 0	Так	так	Creature

Таблиця 3.7 – Інформаційне забезпечення Бестіарію «Таблиця Attribute»

Найменування	Ідентифікатор у програмі	Умове позначення у формулах	Формат	Бізнес-правила				Логічні чи семантичні зв'язки
				Первинний / вторинний ключ	Умова на значення	Обов'язкове поле	Індексне поле	
Ідентифікатор	id	id	Number	PK	Id > 0	так	так	-
Категорія Атрибутів	Attr_Cat	Attr_Cat	string	-	-	Так	Hi	-
Ідентифікатор імені	NameId	NameId	number	FK	-	Так	Так	Translations
Від чого залежить	scalingFrom	scalingFrom	String	-	-	Hi	Hi	-

Таблиця 3.8 – Інформаційне забезпечення Бестіарію «Таблиця Request»

Найменування	Ідентифікатор у програмі	Умовне позначення у формулах	Формат	Бізнес-правила				Логічні чи семантичні зв'язки
				Первинний / вторинний ключ	Умова на значення	Обов'язкове поле	Індексне поле	
Ідентифікатор	id	id	Number	PK	Id > 0	так	так	-
Вид дії	Action_type	Action_type	string	-	-	Так	Hi	-
Ідентифікатор Заголовка	TitleId	TitleId	number	FK	>0	Так	Так	Translations
Ідентифікатор Опису	DescriptionId	Description Id	number	FK	>0	Так	Так	Translations
Ідентифікатор Істоти	CreatureId	CreatureId	number	FK	>0	Так	Так	Creature

Вибір СКБД

Обґрунтування вибору PostgreSQL:

Опис: PostgreSQL є потужною реляційною системою керування базами даних з відкритим кодом, яка підтримує складні запити і транзакції, а також забезпечує високу продуктивність і масштабованість.

Вимоги та обмеження:

- Висока швидкість обробки запитів
- Можливість обробки великих обсягів даних
- Підтримка складних запитів і транзакцій
- Легкість у налаштуванні та адмініструванні
- Інфологічна модель бази (сховища) даних

Інфологічна модель бази даних:

Моделі: ER-діаграма, що показує зв'язки між таблицями бази даних.

Основні елементи: Таблиці для істот, користувачів, логів дій, перекладів.

Даталогічна модель бази (сховища) даних

Фізична модель бази даних:

Діаграма: Відображає фізичну структуру бази даних у PostgreSQL.

Опис складових бази даних:

Таблиці: Creatures, Users, UserActions, Translations

Зв'язки між таблицями: Зовнішні ключі, індекси

3.2 Технічне забезпечення.

Фізичне розташування обладнання: Система "Бестиарій настільної гри Dungeon And Dragons" функціонує на сервері, перетвореному з ПК, розміщеного у домашніх умовах. Сервер під керуванням ОС Linux 20.04 LTS забезпечує обробку даних і зберігання бази даних. Програмування та тестування здійснюється на ноутбучі Acer Nitro 5 AN515-45.

Схема автоматизації:

Сервер забезпечує базу даних і обробку запитів користувачів.

Клієнтська частина працює на ноутбучі, підключеному до сервера через локальну мережу або інтернет. На рисунку 3.1 – зображено загальну схему автоматизації.

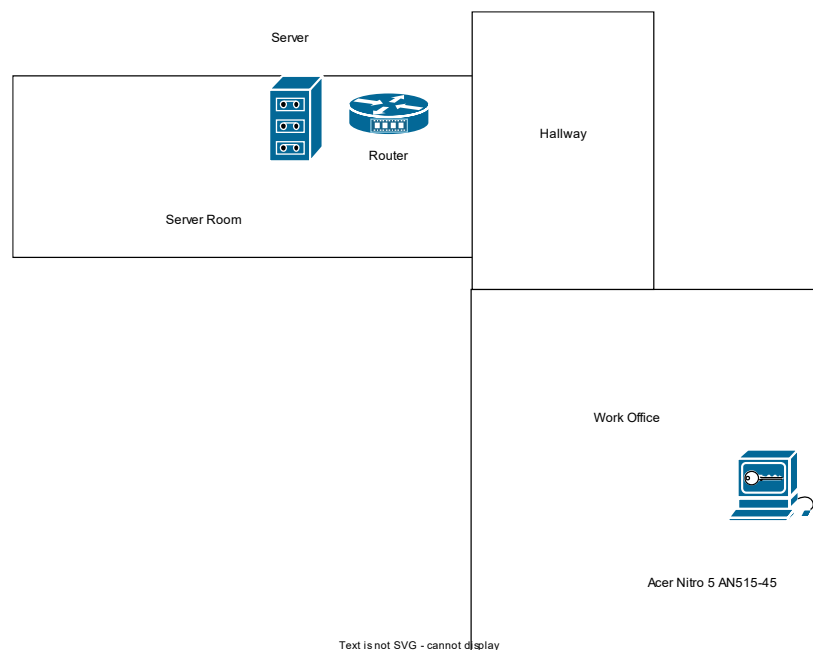


Рисунок 3.1– Загальна схема автоматизації

Структура комплексу технічних засобів

Вибір структури комплексу технічних засобів:

1. Сервер:
 - а) CPU: Intel Core I3-2100
 - б) RAM: DDR3 4GB
 - в) SSD: 128GB

г) GPU: Intel UHD Graphics

д) ОС: Linux 20.04 LTS

2. Робоче місце для розробки:

а) Ноутбук Acer Nitro 5 AN515-45

б) CPU: AMD Ryzen 7 5800H

в) RAM: 16GB

г) SSD: 512GB

д) GPU: Nvidia GeForce RTX 3060

е) ОС: Windows 11

Обґрунтування вибору:

- Сервер: Вибраний ПК як сервер забезпечує достатню потужність для обробки запитів і зберігання бази даних.

- Ноутбук: Висока продуктивність ноутбука дозволяє ефективно здійснювати розробку і тестування програмного забезпечення.

Методи захисту:

- Дані користувачів шифруються.
- Встановлено розподіл доступу за ролями, що обмежує доступ користувачів до адміністративних функцій.

Опис автоматизованого робочого місця (АРМ)

Робоче місце розробника:

- Ноутбук: Acer Nitro 5 AN515-45
- CPU: AMD Ryzen 7 5800H
- RAM: 16GB
- SSD: 512GB
- GPU: Nvidia GeForce RTX 3060
- ОС: Windows 11

Робоче місце сервера:

- ПК:
- CPU: Intel Core I3-2100
- RAM: DDR3 4GB

- SSD: 128GB
- GPU: Intel UHD Graphics
- ОС: Linux 20.04 LTS

Периферійні технічні засоби:

- Монітори, клавіатури, миші, веб-камери для розробки і тестування.

Підключення до мережі:

- Ноутбук: Підключення через Wi-Fi
- Сервер: Підключення через Ethernet

Схема мережі передачі даних

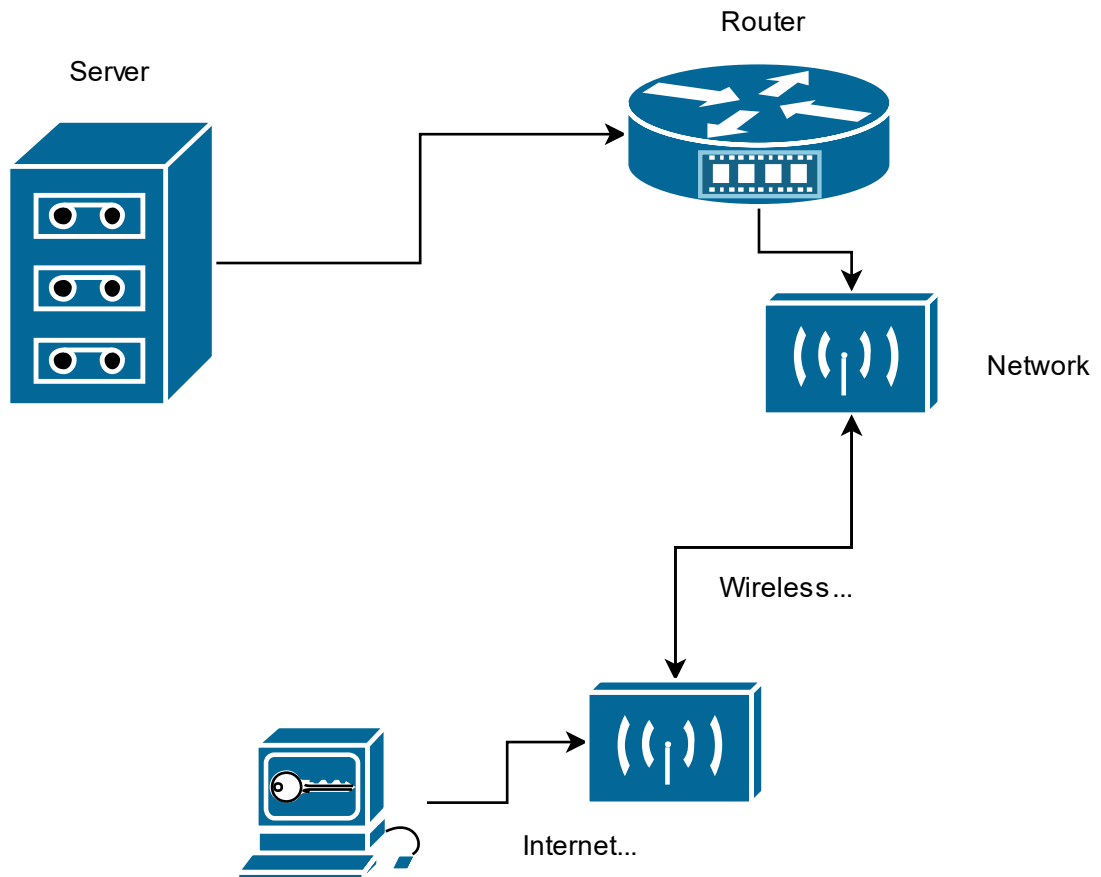
Загальна схема мережі передачі даних:

- Ноутбук розробника підключається до сервера через домашню мережу.
- Сервер підключений до інтернету для доступу до бази даних і обробки запитів.
- Вибір мережного апаратного забезпечення:
- Wi-Fi: Підключення ноутбука через домашній роутер.
- Ethernet: Підключення сервера через кабель.
- Вимоги до каналів зв'язку:
- Швидкість: Мінімальна швидкість інтернет-з'єднання 100 Мбіт/с.
- Тип IP: Статичний IP для сервера.

Розміщення абонентів і об'ємно-тимчасові характеристики передаваних даних:

- Розробник працює з ноутбука, передаючи і отримуючи дані з сервера протягом робочого дня (8 годин на добу, 5 днів на тиждень).
- Основні показники надійності:
- Гарантійний термін використання обладнання: 3 роки
- Періодичність планового ремонту: Раз на рік
- Граничні значення навантаження: Здатність обробляти до 100 запитів на секунду
- Стійкість до електромагнітних завад: Використання екранованих кабелів

Схема мережі передачі даних – зображена на рис 3.2.



Acer Nitro 5 AN515-45 Text is not SVG - cannot display

Рисунок 3.2 – Схема мережі передачі даних

3.3 Програмне забезпечення.

Структура програмного забезпечення для системи "Бестіарій настільної гри Dungeon And Dragons" включає наступні компоненти:

Системне програмне забезпечення

- Операційні системи: Windows 11, Linux 20.04 LTS
- Сервісні програми: антивірусні засоби, утиліти

Прикладне програмне забезпечення

- Розробка: PhpStorm, Node.js, npm, GitHub
- База даних: PgAdmin
- Дизайн: Figma
- Веб-оглядач: Google Chrome

Програмна документація

- Формуляр, керівництво користувача, опис контрольного прикладу, тексти програм

Системне програмне забезпечення

Операційні системи:

- Windows 11: Використовується на ноутбуці для розробки.
- Linux 20.04 LTS: Використовується на сервері для забезпечення надійності та безпеки.

RAD-засоби розробки проекту:

- PhpStorm: Інтегроване середовище розробки для роботи з мовами програмування.
- Node.js і npm: Використовуються для розробки серверної частини, управління пакетами та бібліотеками.

Основні веб-технології:

- JavaScript, HTML, CSS: Використовуються для розробки інтерфейсу користувача.

Інші системні компоненти:

- PgAdmin: Інструмент для управління базами даних PostgreSQL.
- Figma: Інструмент для дизайну інтерфейсу.
- Google Chrome: Веб-оглядач для тестування і відладки веб-додатків.
- Прикладне програмне забезпечення

Сукупність прикладного програмного забезпечення:

Розробка серверної частини:

- Node.js: Серверна платформа для виконання JavaScript-коду.
- npm: Менеджер пакетів для встановлення бібліотек і залежностей.

Розробка інтерфейсу користувача:

- Angular: Фреймворк для створення веб-додатків.
- SCSS, Tailwind: Інструменти для стилізації і верстки.

Управління версіями і співпраця:

- GitHub: Платформа для зберігання коду і спільної роботи над проектом.

Управління базою даних:

- PostgreSQL: Реляційна база даних для зберігання інформації про істот, користувачів і логів.
- PgAdmin: Інструмент для управління базою даних.
- Програмна документація

Основні компоненти програмної документації:

Формуляр: Опис основних характеристик програмного забезпечення, його комплектність і відомості про експлуатацію.

Відомості про призначення програми: Опис використання програмного забезпечення для розв'язання конкретних завдань.

Обмеження на застосування: Мінімальна конфігурація технічних засобів для коректної роботи програмного забезпечення.

Керівництво системного програміста: Відомості для перевірки, забезпечення функціонування і налаштування програмного забезпечення.

Опис контрольного прикладу: Приклади використання програмного забезпечення для перевірки його функціональності.

Тексти програм: Вихідний код програмного забезпечення з коментарями і поясненнями.

3.4 Результати Реалізації Інформаційної підсистеми.

Оригінальні пропозиції та наукові розробки: В процесі виконання дипломної роботи було розроблено інформаційну підсистему "Бестіарій настільної гри Dungeon And Dragons", яка включає в себе комплекс оригінальних пропозицій та наукових розробок. Основна мета системи - забезпечити зручне створення, зберігання та пошук істот для використання майстрами підземелля під час гри.

Основні досягнення:

1. **Створення бази даних для зберігання інформації про істот:** Включає структуру для зберігання даних про атрибути, вміння і навички істот.
2. **Реалізація інтерфейсу для створення і редагування істот:** Забезпечує зручність і швидкість введення даних, включаючи перевірку на унікальність і відповідність атрибутів.

3. **Пошук істот:** Впроваджено зручний інтерфейс для швидкого пошуку істот за різними критеріями (назва, атрибути, вміння).
4. **Підтримка багатомовності:** Введено таблицю перекладів, що дозволяє зберігати назви і описи істот на різних мовах.
5. **Забезпечення безпеки:** Дані користувачів зашифровані, а доступ до функціоналу розподілений за ролями (користувач, адміністратор).

Пілотне впровадження: Система була успішно впроваджена і протестована у контрольному середовищі. Для тестування були використані наступні вхідні дані:

- Інформація про 50 істот різних типів.
- Дані користувачів з різними ролями.

Результати тестування:

1. **Швидкість обробки запитів:** Система здатна обробляти до 100 запитів на секунду.
2. **Час відгуку:** Середній час відгуку на запити користувачів складає 0.2 секунди.
3. **Безпека:** Дані користувачів успішно шифруються, а доступ до адміністративного функціоналу обмежено.
4. **Пошук істот:** Результати пошуку видаються миттєво, навіть при великих обсягах даних.

Аналіз потенційної практичної цінності: Система "Бестіарій" має високу потенційну цінність для майстрів підземелля, надаючи зручний інструмент для управління істотами під час гри. У порівнянні з існуючими альтернативами, система надає більш широкий функціонал і кращу підтримку багатомовності.

Рекомендації щодо подальших досліджень та удосконалень:

1. **Розширення функціоналу:** Додати можливість автоматичного створення істот на основі заданих параметрів.
2. **Покращення інтерфейсу:** Оптимізувати інтерфейс для мобільних пристроїв.

3. **Додаткові інтеграції:** Розглянути можливість інтеграції з іншими системами для імпорту та експорту даних.
4. **Аналіз даних:** Впровадити модулі для аналітики та звітності, що дозволять краще відстежувати використання системи і ефективність її роботи.

Результати аналізу готовності до реального використання: Система готова до реального використання і може бути впроваджена у будь-яку групу майстрів підземелля. Основні компоненти системи пройшли тестування і показали високу надійність і продуктивність.

Перспективи впровадження: Система може бути впроваджена в комерційних умовах як платформа для підтримки настільних ігор. Потенційними користувачами можуть бути клуби настільних ігор, індивідуальні майстри підземелля і розробники нових ігор.

ВИСНОВОК

Бакалаврський проект присвячений розробці ІС «Бестіарію по настільній грі D&D».

Вирішені наступні завдання:

- Зроблено системний аналіз предметної області;
- Побудована інфологічна модель;
- Побудована реляційна модель;
- Реалізація проекту;
- Вирішена поставлена задача.

Проектування ІС було виконано згідно з усіма етапами: зовнішнього, інфологічного, даталогічного та внутрішнього.

Розроблена ІС призначена для зберігання, обробки та зміни інформації про різних істот.

Для грамотного користування ІС ДМ повинні мати мінімальні знання з роботи на комп'ютері.

Необхідність створення ІС «Бестіарію по настільній грі D&D»: організація даних є ключовим моментом при роботі з великими обсягами інформації. Надзвичайно важливо впорядкувати інформацію таким чином, щоб легко і швидко знаходити потрібні відомості. У бестіаріях міститься велика кількість інформації, тому і було необхідно систематизувати процес пошуку та внесення.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. Електронний бестіарій [Електронний ресурс] – Режим доступу до ресурсу: <https://ttg.club/>
2. Електронний бестіарій [Електронний ресурс] – Режим доступу до ресурсу: https://rpg.fandom.com/wiki/Dungeons_%26_Dragons
3. Ситник Н.В. Проектування баз і сховищ даних: Навч. Посібник. – К.:КНЕУ, 2004. – 348 с.
4. Figma [Електронний ресурс] – Режим доступу до ресурсу: <https://www.figma.com/>
5. JetBrains [Електронний ресурс] – Режим доступу до ресурсу: <https://www.jetbrains.com>

ДОДАТКИ

Додаток А Створення Істоти

Home

Barbarian

Classes

Races

Equipment

EN

Creature name:

EN

UA

Alignment:

Select Alignment

+

Type:

Select Type

+

Size:

Select Size

+

Armor class:

0

Armor Type:

Select armor:

+

Hits:

0

Hits in dice:

Hits in dice:

Speeds:

Select speed:

+

Amt:

0

+

STR

0

DEX

0

CON

0

INT

0

WIS

0

CHA

0

Immunities

Select damage:

+

+

Saving throws

Saving throw:

+

Amt:

0

+

Resists

Select damage:

+

+

Skills

Select skill:

+

Amt:

0

+

Vulnerabilities

Select damage:

+

+

Conditions Immunities

Select condition:

+

+

Feelings

Select Feat:

+

Amt:

0

Mer:

+

Languages

Select language:

+

Amt:

0

+

Danger Lvl:

0

Experience:

Experience

Mastery bonus:

0

Abilities

Title EN

Title UA

Description EN

Description UA

ADD

Actions

Title EN

Title UA

Description EN

Description UA

ADD

Bonus Actions

Title EN

Title UA

Description EN

Description UA

ADD

Legendary Actions

Title EN

Title UA

Description EN

Description UA

ADD

Legendary Actions

Title EN

Title EN

Title UA

Title UA

Description EN

Description UA

ADD

Habitats:

Select region:

+

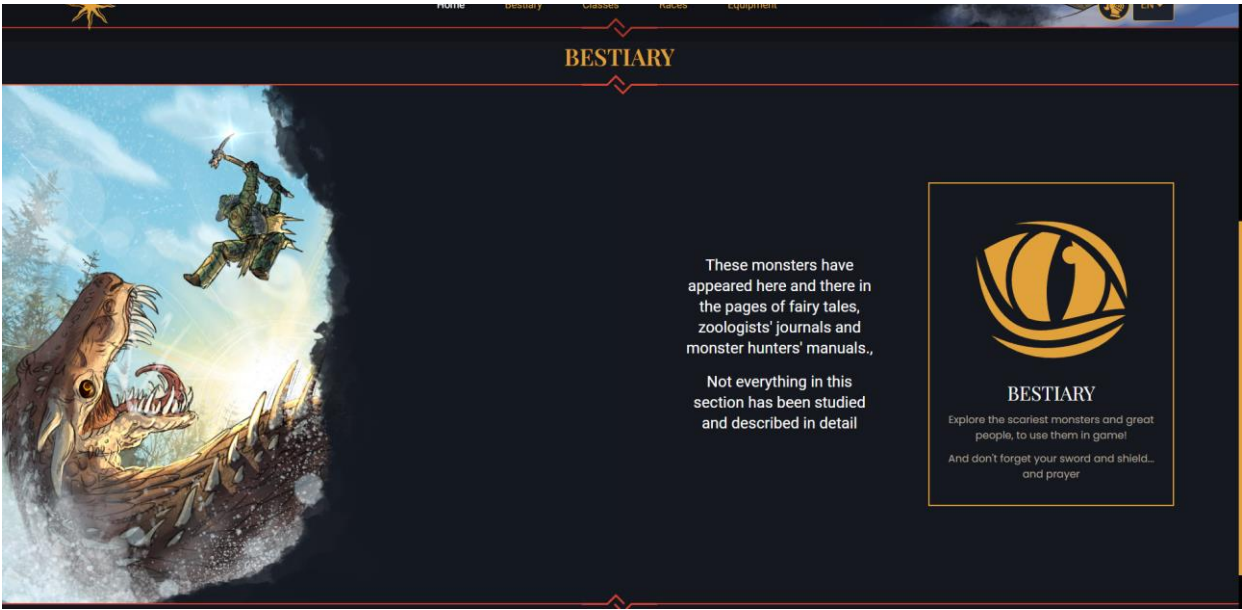
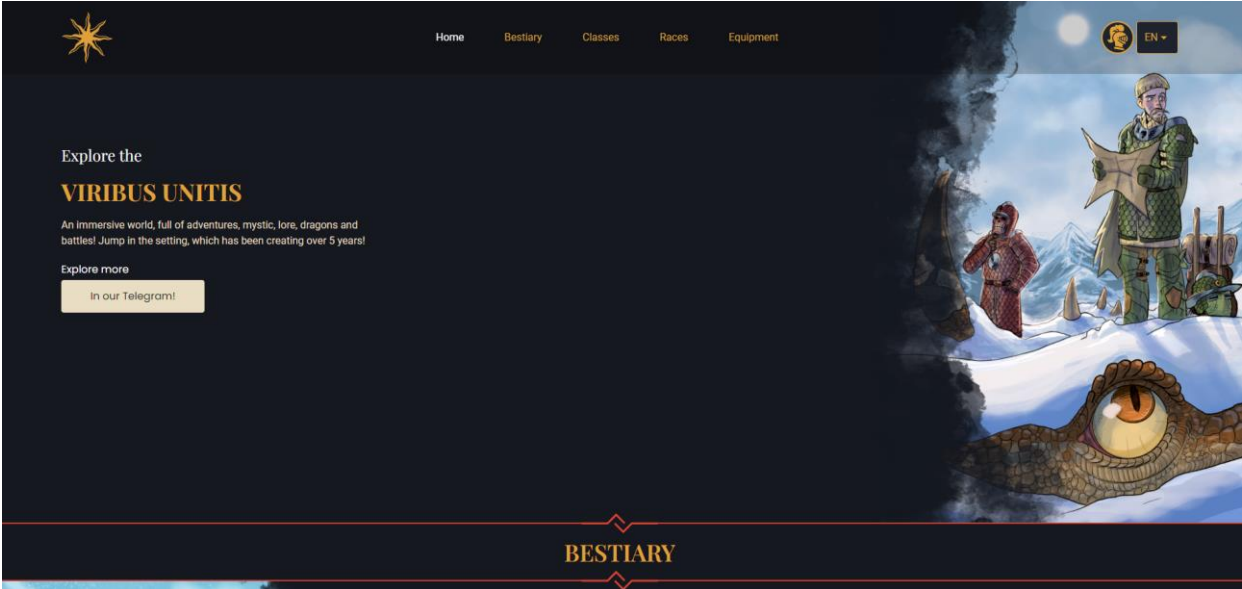
Description

EN

UA

Save Creature

Додаток В Головна сторінка



Додаток С Код Головного Модуля Backend

```
import { Module } from '@nestjs/common';
import { AppController } from '../app.controller';
import { AppService } from '../app.service';
import { TypeOrmModule } from '@nestjs/typeorm';
import { CreatureModule } from '../creature/creature.module';
import { MulterModule } from '@nestjs/platform-express';
import { AuthModule } from '../auth/auth.module';
import { User } from '../auth/entities/user.entity';
import { Alignment } from '../creature/entities/attributes/alignment.entity';
import { Type } from '../creature/entities/attributes/type.entity';
import { Translation } from
'../creature/entities/translations/translation.entity';
import { Size } from '../creature/entities/attributes/size.entity';
import { ArmorTag } from '../creature/entities/attributes/armor-tag.entity';
import { Speed } from '../creature/entities/attributes/speed.entity';
import { Damage } from '../creature/entities/attributes/damage.entity';
import { Feeling } from '../creature/entities/attributes/feeling.entity';
import { SavingThrow } from '../creature/entities/attributes/saving-
throw.entity';
import { Skill } from '../creature/entities/attributes/skill.entity';
import { Condition } from '../creature/entities/attributes/condition.entity';
import { Language } from '../creature/entities/attributes/language.entity';
import { Region } from '../creature/entities/attributes/region.entity';
import { ConditionsMeasure } from '../creature/entities/attribute-
measure/conditions-measure.entity';
import { FeelingsMeasure } from '../creature/entities/attribute-
measure/feelings-measure.entity';
import { ImmunitiesDamageMeasure } from '../creature/entities/attribute-
measure/immunities-damage-measure.entity';
import { LanguagesMeasure } from '../creature/entities/attribute-
measure/languages-measure';
import { ResistsDamageMeasure } from '../creature/entities/attribute-
measure/resists-damage-measure.entity';
import { SavingThrowMeasure } from '../creature/entities/attribute-
measure/saving-throw-measure.entity';
import { SkillsMeasure } from '../creature/entities/attribute-
measure/skills-measure';
import { SpeedsMeasure } from '../creature/entities/attribute-measure/speeds-
measure.entity';
import { VulnerabilitiesDamageMeasure } from '../creature/entities/attribute-
measure/vulnerabilities-damage-measure.entity';
import { Ability } from '../creature/entities/actions-
abilities/abilities.entity';
import { Action } from '../creature/entities/actions-abilities/action.entity';
import { BonusAction } from '../creature/entities/actions-abilities/bonus-
action.entity';
import { LegendaryAction } from '../creature/entities/actions-
abilities/legendary-action.entity';
import { Creature } from '../creature/entities/creature.entity';
import { RegionsMeasure } from '../creature/entities/attribute-
measure/regions-measure.entity';

@Module({
  imports: [
    CreatureModule,
    AuthModule,
    MulterModule.register({ dest: './uploads' }),
    TypeOrmModule.forRoot({
      type: 'sqlite',
      database: 'ViribusUnitis.sqlite',
```

```
    entities: [
      User,
      Type,
      Alignment,
      Size,
      ArmorTag,
      Speed,
      Damage,
      Feeling,
      SavingThrow,
      Skill,
      Condition,
      Language,
      Translation,
      Region,
      ConditionsMeasure,
      FeelingsMeasure,
      ImmunitiesDamageMeasure,
      LanguagesMeasure,
      ResistsDamageMeasure,
      SavingThrowMeasure,
      SkillsMeasure,
      SpeedsMeasure,
      VulnerabilitiesDamageMeasure,
      Ability,
      Action,
      BonusAction,
      LegendaryAction,
      Creature,
      RegionsMeasure,
    ],
    synchronize: true,
  )),
],
controllers: [AppController],
providers: [AppService],
))
export class AppModule {}
```