1 слайд:

Розробка веб-додатку для профорієнтації абітурієнтів з урахуванням специфіки спеціальностей університету (назва роботи), Доповідач: Бєліков Артем Євгенови, Керівник: Жеваго Олександр Олександрович.

2 слайд: Мета роботи та постановка задачі

Розробити веб-додаток для профорієнтаційного тестування абітурієнтів, який дозволяє на основі відповідей на питання, пов’язані з інтересами та навичками, визначати найбільш релевантні спеціальності. Система здійснює порівняння відповідей з еталонними даними студентів, що вже навчаються за відповідними програмами, і надає персоналізовані рекомендації щодо вибору спеціальності відповідно до навчальних програм які вказав адміністратор.

**Ця програма є актуальнольною оскільки** відсутній зручного інструменту для вибору майбутньої спеціальності та зростання кількості напрямів навчання ускладнює ухвалення рішення

Цільова аудиторія включає:

**Абітурієнтів**, які шукають обґрунтоване рішення щодо вступу;

**Студентів-експертів**, які беруть участь у формуванні бази еталонів;

**Адміністратори**

3 слайд: Опис функцій програми

Програма дозволяє створювати нових користувачів, забезпечує перегляд таких сторінок як: головна, про університет, тест та спеціальності. За допомогою зручної адмін панелі дозволяє ефективно управляти даними: категоріями, відповідями користувачів, питаннями та еталонними даними. Колекції питань та категорій пов'язані напряму з заповненням сайту, тому коли користувач додає дані вони відповідно будуть відображатися на сторінках: тест, спеціальності. Також передбачена на сайті авторизація експерта. За допомогою методу косинусної подібності абітурієнтам будуть рекомендовані топ 3 спеціальності з їх описом. У випадку якщо користувач авторизований як експерт, то його відповіді будуть відправлятися на бекенд і відобразяться на адмін панелі в колекції ExpertAnswer (відповіді експерта). Ці дані можуть впливати на еталонні дані спеціальності та рекомендацій в тестуванні.

4 слайд: Опис обраних технологій

Backend / CMS: Strapi · SQLite · Node.js

Frontend: React · TypeScript/JavaScript · CSS · Material UI

Контроль версій та розгортання: GitHub

Текст для доповіді

Strapi (CMS на Node.js)

Дозволяє швидко створити гнучку адмін-панель для керування тестовими питаннями, спеціальностями та відповідями експертів без писання зайвого коду. Має готові API, плагіни й систему ролей, що ідеально відповідає вимозі розмежування прав доступу

SQLite

Вбудована файл-орієнтована СУБД, яку легко інтегрувати зі Strapi. Не потребує окремого сервера — достатньо одного файлу, що спрощує розгортання й робить систему портативною для демонстрацій і тестування.

Node.js

Серверна платформа для Strapi та запуску бізнес-логіки. Широко розповсюджена, має велику екосистему пакетів (node\_modules) й добре масштабується.

React + TypeScript/JavaScript + CSS + Material UI

React дозволяє будувати інтерфейс із окремих блоків (компонентів), які швидко оновлюються на екрані, а за допомогою механізмів hooks і контексту легко зберігати й змінювати стан додатку.

TypeScript додає перевірку типів ще під час написання коду, тому помилки виявляються одразу і простіше підтримувати проєкт, а JavaScript залишається гнучким інструментом для невеликих скриптів.

CSS (модулі чи styled-components) забезпечує адаптивність і кастомізацію стилів.

Material UI пропонує готові компоненти (кнопки, форми, випливаючи меню) з сучасним дизайном та підтримкою темування, що прискорює розробку й уніфікує зовнішній вигляд.

GitHub (контроль версій і розгортання)

Зберігання всієї історії змін.

5 слайд:

Архітектура програми й патерни проєктування

Frontend: React + TypeScript з використанням MUI; компоненти Header, Main, Footer та інші підключаються через ThemeProvider

Backend/CMS: Strapi на Node.js та SQLite. Структура включає сутності Category, Question, CategoryVectorEntry і ExpertAnswer з полем vector для збереження оцінок

Лайфцикли Strapi: після створення чи оновлення категорії запускається функція populateCategoryVectorEntries, яка додає записи до CategoryVectorEntry

Загальний підхід: поєднання MVC на бекенді (Strapi) та компонентної архітектури на фронтенді, з використанням сервісного шару для перерахунку векторів.

Текст для доповіді

Фронтенд є односторінковим застосунком на React, де окремі компоненти відповідають за різні частини інтерфейсу. Бекенд реалізовано за допомогою Strapi, що дотримується MVC‑підходу та надає REST‑API. У Strapi визначені основні сутності (категорії, питання, відповіді експертів), а також lifecycles та сервіс, які автоматично оновлюють зв’язки та обчислюють вектори. Таким чином, дані зручно керуються через CMS, а клієнтська частина отримує їх у готовому вигляді через API.

6 слайд:

Алгоритми

Розрахунок рекомендацій: на сторінці тесту функція cosine обчислює косинусну схожість між відповідями користувача та вектором кожної категорії

Оновлення векторів категорій: після підтвердження відповідей експерта виконується recalcCVE, що вираховує середні значення по кожному питанню та оновлює CategoryVectorEntry

Перерахунок після зміни питань: при додаванні чи видаленні питання викликається recalculateCategory для всіх категорій, щоб синхронізувати вектори

Текст для доповіді

Алгоритм рекомендації базується на косинусній подібності. Після проходження тесту відповіді перетворюються у вектор, який порівнюється з векторами спеціальностей. Найвищі коефіцієнти подібності визначають рекомендовані напрями. Вектори спеціальностей формуються на основі відповідей експертів: Strapi зберігає їх у моделі ExpertAnswer і через сервіс recalcCVE обчислює середні значення. При зміні питань система автоматично перераховує вектори, щоб результати тесту залишалися актуальними.

8 слайд:

Висновки

Реалізовано веб-додаток із модулями тестування, перегляду спеціальностей та інформаційних сторінок

Абітурієнти отримують персоналізовані рекомендації (топ-3 спеціальностей) на основі відповідей

Адміністратори керують контентом і модерує дані через інтуїтивний Strapi

Фронтенд побудовано на React + Material UI, бекенд — Strapi + SQLite

Система забезпечує високу продуктивність, гнучкість та можливість масштабування

Доповідь

У результаті виконання дипломного проєкту створено універсальний веб-додаток для профорієнтації абітурієнтів, що поєднує три ключові блоки: самостійне тестування, перегляд списку спеціальностей та ознайомчі сторінки про університет.

Абітурієнти проходять серію запитань і одразу отримують три найвідповідніші напрями підготовки. Це дозволяє зробити вибір усвідомлено та швидко ознайомитися з перевагами кожної спеціальності.

Адміністратори в адмін-панелі Strapi мають повний контроль: можуть додавати чи змінювати питання, керувати списком спеціальностей і переглядати результати тестів, що гарантує актуальність і якість контенту.

Для реалізації інтерфейсу обрано React із Material UI — швидке створення адаптивних компонентів. Серверна частина на Strapi разом із SQLite забезпечує легке розгортання, автогенерацію REST-API та збереження даних без додаткового налаштування СУБД.

Поєднання сучасних технологій гарантує високу швидкодію, простоту підтримки й можливість масштабувати рішення відповідно до зростання кількості користувачів.

9 слайд:

Дякую за увагу, питання.