

Полтавський університет економіки і торгівлі

Кафедра комп'ютерних наук та інформаційних технологій



КОМП'ЮТЕРНІ НАУКИ ТА ІНФОРМАЦІЙНІ ТЕХНОЛОГІЇ (KNIT-2025)



МАТЕРІАЛИ НАУКОВО-ПРАКТИЧНОГО СЕМІНАРУ Випуск 4

Вересень 2024 р. - червень 2025 р.

Полтава 2025

КОМП'ЮТЕРНІ НАУКИ ТА ІНФОРМАЦІЙНІ ТЕХНОЛОГІЇ (КНІТ-2025): матеріали науково-практичного семінару. Випуск 4 / за ред. О.В. Ольховської – Полтава: Кафедра КНІТ ПУЕТ, 2025. – 56 с.

Збірник матеріалів науково-практичного семінару містить добірку праць присвячених актуальній проблематиці, що висвітлює питання галузі сучасних інформаційних технологій, кібернетики, інформатики, математичного моделювання, системного аналізу, програмного забезпечення інформаційних систем та теорії прийняття оптимальних рішень.

У збірці представлено матеріали, що відображають проблематику підготовки фахівців з комп'ютерних наук та інформаційних технологій.

Всі публікації викладено в авторській редакції.

Ум. друк. арк. 3,5
©Кафедра КНІТ ПУЕТ, 2025

ЗМІСТ

Береславич Б.М., Ольховська О.В. СТВОРЕННЯ АНІМАЦІЙНОГО ВІДЕОКОНТЕНТУ ДЛЯ ОСВІТНЬОЇ РЕКЛАМНОЇ СТРАТЕГІЇ ЗАКЛАДУ ВИЩОЇ ОСВІТИ.....	5
Білуха Д.В., Чілікіна Т. В. РОЗРОБКА ІНТЕРФЕЙСУ ДЛЯ ПЗ ПІДГОТОВКИ ЗОБРАЖЕННЯ ДЛЯ ВЕРСТАТУ ПО ВИГОТОВЛЕННЮ ТРИКОТАЖНИХ ВИРОБІВ.....	8
Вакуленко С.Р. МОБІЛЬНИЙ ЗАСТОСУНОК ДЛЯ ПІДТРИМКИ ЗДОРОВОГО ХАРЧУВАННЯ З БАЗОЮ РЕЦЕПТІВ І ПІДРАХУНКОМ КАЛОРІЙ.....	11
Ганзя І. О., Парфьонова Т. О. РОЗРОБКА ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ TELEGRAM-БОТА ДЛЯ ЗАВАНТАЖЕННЯ ТА ОБРОБКИ ВІДЕО З YOUTUBE.....	14
Дзяткевич М. О., Черненко О. О. ВЕБ-ДОДАТОК ДЛЯ ВІДСТЕЖЕННЯ ТА ПРОГНОЗУВАННЯ СТАНУ ЗДОРОВ'Я НА ОСНОВІ ОБРОБКИ БІОМЕДИЧНИХ ДАНИХ.....	17
Довгий І.В., Кошова О.П. РОЗРОБКА ВЕБ-ЗАСТОСУНКУ ДЛЯ УПРАВЛІННЯ СПРАВАМИ.....	20
Караблін Р. В., Олексійчук Ю. Ф. РОЗРОБКА САЙТУ “ПОРІВНЯННЯ УТИЛІТ ДЛЯ МОНІТОРИНГУ СТАНУ АПАРАТНИХ ЗАСОБІВ ПК”.....	24
Кряж О.В., Черненко О. О. ПРОГРАМНА РЕАЛІЗАЦІЯ САЙТУ ЗІ СТАТИСТИКОЮ ІГРОВИХ ПРОФІЛІВ.....	27
Кіліян Д.В., Олексійчук Ю.Ф. РОЛЬ АДАПТИВНИХ НАВЧАЛЬНИХ ПРОГРАМ У ПОТЕНЦІАЛІ ДИСТАНЦІЙНОЇ ОСВІТИ.....	30
Соколов А.В., Олексійчук Ю.Ф. РОЗРОБКА САЙТУ ІНТЕРНЕТ-МАГАЗИНУ «KIDDO».....	33
Тобота Я. О. Чілікіна Т. В. РОЗРОБКА САЙТУ ІНТЕРНЕТ-МАГАЗИНУ «ОТАК».....	36
Тур В., Ольховська О.В. РОЗРОБКА WEB-СЕРВІСУ ДЛЯ ІНТЕРАКТИВНОГО ОЦІНЮВАННЯ СТУДЕНТІВ З ВИКОРИСТАННЯМ NLP-МОДЕЛЕЙ.....	38
Фісун Д. А., Чілікіна Т. В. ПРОЕКТУВАННЯ ВЕБ-СТОРІНКИ ЗА ДОПОМОГОЮ МОВИ ПРОГРАМУВАННЯ PYTHON ТА ФРЕЙМВОРКУ FLASK.....	41
Халявка Г. В., Кошова О. П. ОСОБЛИВОСТІ РОЗРОБКИ ВЕБ-САЙТУ ДЛЯ РЕМОНТУ ТА ПОШИВУ ОДЯГУ З ВИКОРИСТАННЯМ LARAVEL.....	45

Храпко Д. А., Кошова О. П. РОЗРОБКА ЧАТ-БОТА ДЛЯ НАДАННЯ КОНСУЛЬТАЦІЙ В СФЕРІ КІБЕРБЕЗПЕКИ.....	50
Шаповал П.В. РОЗРОБКА НАВЧАЛЬНОГО ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ З ТЕМИ «ОСНОВИ МОВИ ЗАПИТІВ HQL» ДИСЦИПЛІНИ «ТЕХНОЛОГІЇ ОБРОБКИ ТА АНАЛІЗУ ДАНИХ»....	52
Шеліхов А.С., Парфьонова Т.О. АЛГОРИТМІЗАЦІЯ ТА РОЗРОБКА ЗАСТОСУНКУ ПЕРСОНАЛІЗОВАНИХ РЕКОМЕНДАЦІЙ ДЛЯ ОПТИМІЗАЦІЇ ФІЗИЧНОЇ АКТИВНОСТІ НА ОСНОВІ МОНІТОРИНГУ КРОКІВ ТА ВИТРАЧЕНИХ КАЛОРІЙ.....	54

СТВОРЕННЯ АНІМАЦІЙНОГО ВІДЕОКОНТЕНТУ ДЛЯ ОСВІТНЬОЇ РЕКЛАМНОЇ СТРАТЕГІЇ ЗАКЛАДУ ВИЩОЇ ОСВІТИ

Б.М. Береславич, студент групи КН-б-41, спеціальності
«Комп'ютерні науки»,

О.В. Ольховська, науковий керівник, завідувач кафедри, кандидат
фізико-математичних наук,
Полтавський університет економіки і торгівлі

Ключові слова: АНІМАЦІЯ, МОНО PRO, РИГІНГ, ЛІПСИНГ,
ВІДЕОМОНТАЖ, РЕКЛАМНА СТРАТЕГІЯ, ТІКТОК

У світі цифрових комунікацій відео є одним з найефективніших інструментів взаємодії з аудиторією. В умовах жорсткої конкуренції між закладами вищої освіти (ЗВО) саме візуальний контент стає вирішальним чинником у процесі залучення потенційних абітурієнтів [1]. Особливо актуальним є використання коротких, динамічних та емоційно насичених відеоформатів — таких як ролики для TikTok, Instagram Reels чи YouTube Shorts. Анімація, як окрема форма цифрового мистецтва, дозволяє не лише передати інформацію, а й створити візуально унікальний образ навчального закладу, що виділяється серед конкурентів [4].

Було розроблено повноцінний цикл створення анімаційного відеоконтенту з використанням сучасних програмних інструментів. Основним середовищем для анімації обрано Mono Pro — професійну платформу для 2D-анімації [2; 3], що поєднує інструменти ригінгу, ліпсінгу, векторної графіки та ключової анімації. Завдяки цьому забезпечено високу якість фінального відеопродукту при оптимізації часу та зусиль розробника.

Процес створення відео включав кілька етапів:

1. Попередній аналіз і постановка цілей. Було проаналізовано маркетингові потреби університету, досліджено актуальні тренди в освітній відеореklamі, визначено основну цільову аудиторію — підлітки та молодь 16–19 років, які активно використовують соціальні мережі [1; 4].

2. Розробка сценарію та композиційної структури. Визначено сюжетну лінію, побудовано розкадровку з урахуванням принципів сторітелінгу, визначено ключові елементи сцен: персонажі, фон, емоційні акценти, таймінг.

3. Створення анімації. Виконано ригінг персонажа — побудова скелета, що дозволяє керувати рухами. Застосовано автоматизований ліпсинг за аудіо доріжкою — синхронізація руху губ зі звуком. Налаштовано рухи тіла, міміку, вирази обличчя.

4. Постпродакшн. Змонтовано відео в зручний для соціальних мереж формат (9:16). Додано субтитри, фонову музику, адаптовано ролик під TikTok-алгоритми — коротка тривалість, перші 3 секунди з акцентом на емоцію, завершення закликком до дії.

5. Оптимізація та тестування. Відео протестовано на фокус-групі. Зафіксовано позитивну динаміку за такими критеріями: сприйняття інформації, запам'ятовуваність, мотивація до ознайомлення з університетом.

Крім технічної реалізації, особливу увагу було приділено аспектам дизайну, що впливають на психологічне сприйняття глядачем: використано м'яку кольорову гаму, доброзичливі персонажі, стиль «плоскої» графіки, що добре сприймається молоддю.

Результатом стало ефективне маркетингове відео, яке може бути використане на сторінках університету в TikTok, Instagram та інших цифрових платформах. Робота підтвердила актуальність інтеграції сучасних інформаційних технологій — зокрема, 2D-анімації — в процес освітнього маркетингу, а також продемонструвала приклад міждисциплінарного підходу, що поєднує IT, медіа, педагогіку та рекламу.

Практична цінність проекту полягає в можливості багаторазового використання анімаційного шаблону для створення нових відео, а також у потенціалі впровадження технології в інші

освітні процеси — від створення навчальних роликів до інтерактивних курсів.

Література:

1. Методичні рекомендації до виконання кваліфікаційної роботи за освітньою програмою "Комп'ютерні науки". — Полтава: ПУЕТ, 2025. — 34 с.

2. Бомберчік Аніматронік. YouTube-канал. Відеокурс з анімації в Moho Pro: створення персонажів, ригінг, ліпсінг, камера, сцени. — [Електронний ресурс]. — URL: <https://www.youtube.com/@bomberchik/videos>

3. Mult-urokiRu. YouTube-канал. Освітній курс з анімації в Moho Pro. — [Електронний ресурс]. — URL: <https://www.youtube.com/@Mult-urokiRu>

4. Нарисовано нормально. TikTok-акаунт. Приклади коротких анімаційних відео. — [Електронний ресурс]. — URL: <https://www.tiktok.com/@narisovano.normalno>

РОЗРОБКА ІНТЕРФЕЙСУ ДЛЯ ПЗ ПІДГОТОВКИ ЗОБРАЖЕННЯ ДЛЯ ВЕРСТАТУ ПО ВИГОТОВЛЕННЮ ТРИКОТАЖНИХ ВИРОБІВ

*Д.В. Білуха, здобувач освіти спеціальності «Комп'ютерні науки»
Т. В. Чілікіна, науковий керівник к.ф.-м.н., доцент кафедри
комп'ютерних наук та інформаційних технологій
Полтавський університет економіки і торгівлі*

У сучасному виробництві трикотажних виробів важливу роль відіграє якісна підготовка зображень для їх подальшої обробки на автоматизованих верстатах [1, 2]. Технологічний процес потребує точності, ефективності та зручності, а відповідне програмне забезпечення повинно забезпечити користувача простим і логічним інтерфейсом. Основною метою даної роботи є розробка функціонального інтерфейсу для ПЗ, що дозволяє здійснювати підготовку зображень до виводу на текстильні верстати.

Розробка користувацького інтерфейсу є важливою задачею, оскільки існуючі рішення часто мають складний або надмірно технічний дизайн, що ускладнює роботу операторів виробництва. Запропоноване рішення базується на технологіях JavaFX та FXML, що дозволяють створювати адаптивні, інтерактивні та зручні інтерфейси користувача [1].

Функціональні можливості створеного інтерфейсу включають:

- вибір зображення для обробки та завантаження його у програму;
- базову обробку: масштабування, корекцію яскравості, контрастності, конвертацію у відповідні формати;
- попередній перегляд результату обробки у реальному часі;
- можливість експорту зображення у формат, який підтримується текстильними верстатами;
- автоматичне збереження історії змін і шаблонів налаштувань;

- інтеграція з модулями підготовки візерунків;
- збереження метаданих (інформації про авторство, дату, розмір тощо) для автоматизованого звітування.

Окрему увагу приділено зручності навігації — інтерфейс реалізовано із застосуванням принципів інтуїтивної взаємодії користувача з елементами UI, що базується на рекомендаціях з ефективної комунікації [3]. Це забезпечує високу швидкість навчання операторів, а також зменшує ризик помилок у виробничому процесі.

Особливості реалізації на базі JavaFX дозволяють ефективно масштабувати систему, розгортати її на різних операційних системах, інтегрувати з іншими модулями автоматизації. Також реалізована можливість підтримки багатомовного інтерфейсу, що є важливим для підприємств із міжнародною співпрацею.

Інтерфейс був протестований на базовому наборі виробничих зображень, результати тестування підтвердили його ефективність, зручність та відповідність вимогам виробництва. Проведене анкетування серед операторів показало зростання задоволеності процесом роботи з ПЗ на 35% порівняно з попереднім рішенням.

Окрім основних функціональних можливостей, також було реалізовано модуль експорту результатів у формати, сумісні з САД-системами для текстильної промисловості, що дозволяє зменшити втрати при перенесенні візерунків між різними програмами. Передбачено логування операцій з метою забезпечення безпеки та відстеження змін при колективній роботі над виробничими завданнями.

Узагальнюючи, розробка спеціалізованого програмного інтерфейсу для автоматизації підготовки зображень у сфері виробництва трикотажу є інноваційним рішенням, яке може бути масштабоване й на інші сфери графічного виробництва. Впровадження цього програмного забезпечення дозволяє зменшити витрати часу, підвищити якість результату та полегшити роботу операторів. Таке рішення також відкриває нові можливості в адаптації підприємств до індустрії 4.0, де автоматизація та цифровізація процесів є ключовими чинниками успіху.

Література

1. Morato, C. R. (2021). JavaFX 17 by Example. Apress.
2. Norman, D. (2013). The Design of Everyday Things. Basic Books.
3. McKay, E. N. (2013). UI is Communication: How to Design Intuitive, User Centered Interfaces by Focusing on Effective Communication. New Riders.
4. Кулік О.М. Автоматизовані системи управління текстильним виробництвом: сучасний стан та перспективи. // Технології легкої промисловості. — 2021. — №3.
5. Ісаєнко Л.О. Програмні засоби обробки графіки в дизайні тканин. // Вісник ХНУМГ ім. Бекетова. — 2022. — №4.

МОБІЛЬНИЙ ЗАСТОСУНОК ДЛЯ ПІДТРИМКИ ЗДОРОВОГО ХАРЧУВАННЯ З БАЗОЮ РЕЦЕПТІВ І ПІДРАХУНКОМ КАЛОРІЙ

С.Р. Вакуленко, студент гр. КН-М-11, спеціальності
«Комп'ютерні науки»
Полтавський університет економіки і торгівлі

У сучасній розробці мобільних застосунків для підтримки здорового способу життя важливим елементом є створення зручних та ефективних інтерфейсів, що сприяють досягненню цілей користувачів. Адаптивність, масштабованість та високий рівень конфігурації є основними критеріями для успіху таких застосунків, особливо у контексті швидких змін вимог до дизайну та функціональності. Метою цієї роботи є розробка мобільного застосунку для підтримки здорового харчування, який включає базу рецептів і функцію підрахунку калорій, з акцентом на високу продуктивність, адаптивність та гнучкість.

У рамках проекту було проведено аналіз сучасних підходів до розробки мобільних застосунків, які мають функціональність для здорового харчування, використовуючи популярні технології для мобільних платформ. Дослідження включали вивчення підходів до розробки баз рецептів, реалізації підрахунку калорій, інтеграції з базами даних та адаптивності інтерфейсів для різних пристроїв. Використані джерела [1-5] висвітлюють ключові аспекти мобільного дизайну, організації компонентів і оптимізації продуктивності.

Розроблений застосунок включає:

- Адаптивність: забезпечення коректного відображення компонентів інтерфейсу на різних мобільних пристроях з різними розмірами екранів та орієнтаціями.
- Масштабованість: можливість розширення застосунку новими рецептами, функціями або інтеграціями без порушення його структури.

- Підрахунок калорій: автоматичний підрахунок калорій та макроелементів для кожної страви на основі бази даних рецептів.

- Конфігурація: підтримка налаштувань, які дозволяють користувачам змінювати стилі та функціональність компонентів застосунку, наприклад, для дієт, тем або обраних інгредієнтів.

Процес розробки застосунку включав такі етапи:

1. Дослідження вимог аналіз необхідних компонентів, таких як база рецептів, функціональність підрахунку калорій, можливості для індивідуальних налаштувань.

2. Проектування архітектури створення модульної та гнучкої архітектури застосунку, що дозволяє додавати нові компоненти без порушення роботи всього застосунку.

3. Розробка компонентів розробка окремих функціональних блоків, таких як перегляд рецептів, підрахунок калорій, рекомендації по дієтах та планах харчування.

4. Документування створення документації для використання компонентів та їх налаштувань, щоб спростити подальше використання та інтеграцію для інших розробників.

Одним із ключових аспектів проекту стало забезпечення повторного використання коду. Для цього були застосовані такі підходи

- Використання універсальних шаблонів стилізації для компонентів, що дозволяє легко адаптувати інтерфейс під різні екрани мобільних пристроїв.

- Інкапсуляція бізнес-логіки (підрахунок калорій, рекомендації по рецептах) у окремі модулі.

- Впровадження системи управління темами для динамічного налаштування стилів в залежності від уподобань користувача.

У процесі розробки також було впроваджено заходи для підвищення продуктивності застосунку

- Оптимізація стилів та компонентів для зменшення часу завантаження і покращення користувацького досвіду.

- Налаштування кешування та асинхронного завантаження даних для швидшої роботи застосунку.

Цей підхід дозволяє створити ефективний, адаптивний і масштабований мобільний застосунок для підтримки здорового харчування, який надає користувачам можливість стежити за своїм

раціоном та досягати поставлених цілей щодо здорового способу життя.

Література

1. Smith, J. (2020). *Mobile Application Development for Health and Wellness*. Springer. Retrieved from <https://www.springer.com>
2. Johnson, M., & Brown, L. (2019). *React Native for Cross-Platform App Development*. O'Reilly Media. Retrieved from <https://www.oreilly.com>
3. Anderson, D. (2021). *Building Scalable Mobile Apps with Efficient Data Management*. Wiley. Retrieved from <https://www.wiley.com>
4. Lee, C., & Patel, R. (2022). *Tailwind CSS for Modern Web and Mobile Design*. Apress. Retrieved from <https://www.apress.com>
5. White, T., & Harris, S. (2018). *Storybook: A Tool for Component-Driven Development*. Packt Publishing. Retrieved from <https://www.packtpub.com>

РОЗРОБКА ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ TELEGRAM-БОТА ДЛЯ ЗАВАНТАЖЕННЯ ТА ОБРОБКИ ВІДЕО З YOUTUBE

І. О. Ганзя, студент спеціальності «Комп'ютерні науки», група КНб-41

Т. О. Парфьонова, науковий керівник, кандидат фізико-математичних наук, доцент кафедри комп'ютерних наук та інформаційних технологій

Полтавський університет економіки і торгівлі

Ключові слова: TELEGRAM-БОТ, PYTHON, YT-DLP, ОБРОБКА ВІДЕО, ЗАВАНТАЖЕННЯ МУЛЬТИМЕДІА

Keywords: TELEGRAM BOT, PYTHON, YT-DLP, VIDEO PROCESSING, MEDIA DOWNLOAD

У статті представлено програмне рішення у вигляді Telegram-бота для автоматизованого завантаження та обробки відео з YouTube та інших популярних соціальних платформ. В основі розробки лежить мова програмування Python із використанням бібліотек aiogram, yt-dlp та ffmpeg, які забезпечують зручну інтеграцію з Telegram API і дають змогу реалізовувати складну логіку роботи з мультимедійними файлами [1, 2, 3].

Telegram-бот реалізує низку функціональних можливостей:

- завантаження відео у якості до 1080p;
- конвертація відео у аудіоформат MP3;
- обрізання аудіо за часовими параметрами користувача;
- отримання субтитрів, включаючи україномовні, якщо доступні;
- збереження окремих треків із YouTube-плейлистів;
- повідомлення про завершення процесу завантаження або виявлені помилки [4, 5].

Проект було реалізовано з урахуванням обмежень Telegram API щодо обсягу переданих файлів (до 2 ГБ), що стало важливим фактором у побудові архітектури системи. Особливу увагу приділено оптимізації розміру медіа через використання бібліотеки `ffmpeg` для стиснення без суттєвих втрат якості [1].

Архітектурно рішення побудовано на асинхронній моделі обробки запитів за допомогою `asyncio`, що дозволяє працювати з великою кількістю користувачів без зниження продуктивності. Для зберігання даних та логування дій використано `SQLite3` [6].

Конфігурація бота здійснюється через `.env`-файли, що дозволяє адаптувати розгортання на різних середовищах (локально, VPS, хмарні хости). Передбачено обробку виключень та введено систему логування дій з можливістю подальшого аудиту [6].

Інтерфейс взаємодії користувача побудовано максимально просто: після введення посилання на відео, бот надсилає варіанти якості для завантаження, а також можливість вибору формату. У випадку помилки користувач отримує відповідне повідомлення з поясненням, що дозволяє зменшити кількість повторних запитів і підвищити ефективність [5].

Рішення було протестовано на різних платформах — Android, iOS та десктоп — і продемонструвало стабільність роботи. У ході тестування зафіксовано зменшення середнього часу обробки запитів до 3 секунд завдяки використанню кешування та потокової обробки даних.

Telegram-бот може бути використаний як інструмент у навчальному процесі (наприклад, під час вивчення тем цифрового контенту або розробки API), а також як приклад реалізації мікросервісної архітектури у рамках невеликого IT-проекту. Його гнучкість дозволяє масштабувати функціонал та додавати підтримку нових джерел відео (TikTok, Instagram, Facebook тощо) за допомогою адаптації `yt-dlp`.

Таким чином, розроблений Telegram-бот є прикладом ефективного застосування сучасних мов і технологій програмування для автоматизації завдань обробки мультимедійного контенту. Його подальший розвиток може включати створення панелі адміністратора, розширення функціональності та інтеграцію з базами даних користувачів.

Література:

1. FFmpeg. Official site. [Електронний ресурс]. – URL: <https://ffmpeg.org/>
2. Python documentation. [Електронний ресурс]. – URL: <https://docs.python.org/3/>
3. Aiogram documentation. [Електронний ресурс]. – URL: <https://docs.aiogram.dev/>
4. yt-dlp GitHub repository. [Електронний ресурс]. – URL: <https://github.com/yt-dlp/yt-dlp>
5. Telegram Bot API documentation. [Електронний ресурс]. – URL: <https://core.telegram.org/bots/api>
6. SQLite documentation. [Електронний ресурс]. – URL: <https://www.sqlite.org/docs.html>

ВЕБ-ДОДАТОК ДЛЯ ВІДСТЕЖЕННЯ ТА ПРОГНОЗУВАННЯ СТАНУ ЗДОРОВ'Я НА ОСНОВІ ОБРОБКИ БІОМЕДИЧНИХ ДАНИХ

М. О. Дзяткевич, магістр

Полтавський університет економіки і торгівлі

О. О. Черненко, доцент, канд. фіз.-мат. наук,

Полтавський університет економіки і торгівлі

У сучасних умовах збільшення хронічних захворювань, зниження рівня фізичної активності та високої зайнятості населення, набувають популярності інноваційні ІТ-рішення у сфері охорони здоров'я. Веб-додатки, що дозволяють користувачам самостійно контролювати показники свого здоров'я, допомагають вчасно виявляти відхилення, отримувати рекомендації та запобігати серйозним ускладненням [1, 5].

Мета даного проєкту — розробка веб-додатку, що забезпечує моніторинг та базове прогнозування стану здоров'я користувача на основі біомедичних даних, таких як частота серцевих скорочень, рівень кисню в крові, артеріальний тиск, кількість пройдених кроків та витрачених калорій [1, 6, 7].

Інструментарій:

- Back-end: Flask (Python), база даних SQLite [4];
- Front-end: HTML/CSS, Bootstrap, JavaScript;
- Бібліотеки для обробки даних: Pandas, Scikit-learn [2, 3];
- API: Google Fit, Apple HealthKit (для отримання біометричних даних) [6, 7];
- Хмарна платформа: Heroku для розгортання.

Архітектура системи реалізована у вигляді клієнт-серверної моделі. Користувач взаємодіє з веб-інтерфейсом через браузер, відправляє дані або дозволяє зчитування з API-фітнес-трекера. Зібрані дані обробляються серверною частиною і на основі моделі

машинного навчання прогнозується ймовірність відхилення параметрів здоров'я від норми [3].

Ключові функції веб-додатку:

1. Реєстрація та аутентифікація користувачів;
2. Збір біомедичних показників;
3. Візуалізація історії змін здоров'я (графіки, діаграми);
4. Алгоритмічний аналіз та виявлення ризиків;
5. Генерація індивідуальних рекомендацій;
6. Повідомлення про критичні відхилення.

Для прогнозування використовувалася модель логістичної регресії, яка навчалася на відкритому наборі даних про стан здоров'я (наприклад, HealthData.gov) [1]. Модель дозволяє визначати ймовірність критичних змін у здоров'ї та пропонує користувачу рекомендації щодо активності, сну або звернення до лікаря [3].

Особливу увагу приділено забезпеченню безпеки та конфіденційності. Всі дані шифруються, реалізовано базовий рівень аутентифікації та HTTPS-з'єднання. Крім того, структура бази даних спроектована таким чином, щоб уникнути дублювання та забезпечити швидкий доступ до історичних записів користувача [4].

Результати тестування веб-додатку з фокус-групою із 10 учасників засвідчили високу зручність використання та коректність обробки даних. Близько 80% учасників відзначили зростання власної обізнаності щодо змін у фізичному стані, що підтверджує практичну ефективність запропонованого інструменту.

Отже, веб-додаток може бути використаний як допоміжний засіб у щоденному моніторингу здоров'я, а також стати корисним компонентом у складі системи електронної охорони здоров'я. У майбутньому можлива інтеграція з медичними закладами для попереднього аналізу стану пацієнтів.

Література

1. HealthData.gov: Free Data for a Healthier America. [Електронний ресурс]. – Режим доступу: <https://healthdata.gov>

2. McKinney W. Python for Data Analysis. – O'Reilly, 2022.
3. Pedregosa F. та ін. Scikit-learn: Machine Learning in Python. – JMLR 12, 2011.
4. Grinberg M. Flask Web Development. – O'Reilly, 2018.
5. Панков А. Медична інформатика. – Харків: Фоліо, 2021.
6. Google Fit API Documentation. [Електронний ресурс]. – Режим доступу: <https://developers.google.com/fit>
7. Apple HealthKit API Overview. [Електронний ресурс]. – Режим доступу: <https://developer.apple.com/healthkit/>

РОЗРОБКА ВЕБ-ЗАСТОСУНКУ ДЛЯ УПРАВЛІННЯ СПРАВАМИ

І.В. Довгий, здобувач освіти КНб-41 спеціальності

«Комп'ютерні науки»,

О.П. Кошова, науковий керівник, доцент кафедри, кандидат педагогічних наук,

Полтавський університет економіки і торгівлі

В роботі розглядається веб-застосунок для управління особистими справами.

I. V. Dovhyi, O. P. Koshova Development of a web application for task management

Ключові слова: веб-застосунок, управління справами, локальне збереження, адаптивний дизайн, SPA, JavaScript.

Keywords: web-application, task management, local storage, responsive design, SPA, JavaScript.

У сучасному цифровому світі ефективно управління часом та завданнями стає критично важливим. Зі зростанням кількості особистих і професійних завдань зростає потреба у простих, автономних та надійних засобах управління ними. Веб-застосунки для керування справами дозволяють користувачам структурувати свій час, контролювати дедлайни та підтримувати високу продуктивність.

Метою даної роботи є розробка автономного веб-застосунку для управління справами, що функціонує без серверної частини та забезпечує базовий функціонал для взаємодії з завданнями: додавання, редагування, фільтрація, збереження даних.

Об’єкт дослідження – клієнтський веб-застосунок для персонального управління завданнями.

Предмет дослідження – методи проєктування, реалізації та тестування SPA-застосунків із використанням локального сховища браузера та адаптивної верстки.

Для визначення ключових функціональних та UX-вимог було проаналізовано такі сервіси:

- Todoist – один із найпопулярніших застосунків для управління завданнями. Має продуману структуру завдань, підтримку проєктів, тегів, нагадувань [2];
- Google Tasks – проста інтегрована система завдань, зручна для користувачів екосистеми Google [3];
- Microsoft To Do – інструмент для особистих і командних списків, з інтеграцією в Microsoft Outlook [4].

Проаналізовано підходи до збереження даних, інтерактивність інтерфейсу, UX/UX практики, зручність навігації та мобільну адаптацію.

Для реалізації застосунку використано такі технології:

- HTML5/CSS3 – для структуризації та оформлення сторінки;
- JavaScript (ES6) – як основна мова логіки [1];
- Web Storage API (localStorage) – для збереження даних без серверу [5];
- Flexbox та media queries – для адаптивного відображення [6];
- UML-діаграми – для візуалізації архітектури [7].

Архітектура веб-застосунку

Застосунок реалізовано у форматі SPA (Single Page Application), де навігація та оновлення контенту відбуваються без повного перезавантаження сторінки.

Ключові компоненти:

- Головна панель управління завданнями (інпут + кнопка створення);
- Список завдань з можливістю фільтрації за статусом (усі / активні / завершені);
- Форма редагування завдання;
- Інтерфейс підтвердження видалення;
- Локальне збереження через localStorage.

UML-діаграми включають:

- use-case діаграму (додати / редагувати / фільтрувати / видалити завдання);
- блок-схему логіки управління подіями;
- діаграму класів (Task, TaskManager, UIController).

Основні функції застосунку

1. Додавання завдань. Користувач вводить текст завдання, яке зберігається як об'єкт із полями id, title, completed, timestamp.

2. Редагування та видалення. Можна змінити назву, відмітити як виконане або повністю видалити завдання.

3. Фільтрація. Використовується фільтр для відображення:

- усіх завдань;
- лише активних;
- лише завершених.

4. Збереження в localStorage. Завдання зберігаються у вигляді JSON у браузері, що дозволяє застосунку працювати без інтернету.

Проведено ручне тестування за критеріями:

- додавання/видалення/редагування працюють стабільно;
- дані не втрачаються при перезавантаженні;
- адаптивність – інтерфейс коректно відображається на телефонах, планшетах, ПК;
- UX – кнопки, елементи керування логічно розташовані, зручні для користувача.

У результаті дослідження та реалізації проєкту досягнуто таких результатів:

- створено повнофункціональний прототип застосунку для управління особистими завданнями;
- забезпечено локальне збереження даних без використання серверної частини;
- реалізовано адаптивний дизайн, що відповідає вимогам UX;
- підготовлено архітектурну документацію у вигляді UML-діаграм;
- застосунок може бути основою для подальшого розвитку або комерціалізації як легкий автономний task-менеджер.

Надалі можлива інтеграція з серверною частиною, синхронізація між пристроями та реалізація функціоналу авторизації.

Література:

1. JavaScript documentation. Mozilla Developer Network. URL: <https://developer.mozilla.org/en-US/docs/Web/JavaScript/>
 2. Todoist – система управління завданнями. URL: <https://todoist.com/>
 3. Google Tasks – офіційний сайт. URL: <https://tasks.google.com/>
 4. Microsoft To Do. Офіційна сторінка. URL: <https://todo.microsoft.com/>
 5. Web Storage API documentation. Mozilla Developer Network. URL: https://developer.mozilla.org/en-US/docs/Web/API/Web_Storage_API
 6. Responsive Design. Mozilla Developer Network. URL: [https://developer.mozilla.org/en-US/docs/Learn/CSS/CSS layout/Responsive Design](https://developer.mozilla.org/en-US/docs/Learn/CSS/CSS_layout/Responsive_Design)
- UML diagrams overview. Visual Paradigm. URL: <https://www.visual-paradigm.com/guide/uml-unified-modeling-language/uml-diagram-types/>

РОЗРОБКА САЙТУ “ПОРІВНЯННЯ УТИЛІТ ДЛЯ МОНІТОРИНГУ СТАНУ АПАРАТНИХ ЗАСОБІВ ПК”

Караблін Р. В., здобувач освіти спеціальності 122 “Ком’ютерні науки”, студент групи КН б-41.

*Олексійчук Ю. Ф., кандидат фізико-математичних наук
Полтавський університет економіки і торгівлі*

У сучасному користувацькому середовищі персональних комп’ютерів важливим аспектом є оперативний моніторинг стану апаратного забезпечення. Це особливо актуально в умовах інтенсивного використання ПК для професійних завдань, ігор, обробки відео, а також у серверному середовищі. Регулярне відстеження температури, навантаження на процесор, відеокарту та інші компоненти дозволяє запобігти критичним збоєм, перегрівам і втраті даних [1; 2].

На ринку існує велика кількість утиліт для моніторингу, серед яких найпоширенішими є: HWMonitor, AIDA64, Open Hardware Monitor, MSI Afterburner, Speccy [3–5]. Проте вибір між ними ускладнюється через різний функціонал, моделі ліцензування, інтерфейсну реалізацію та технічні можливості.

Мета і завдання дослідження

Мета проєкту — створення інтерактивного вебресурсу, який би дозволяв користувачам легко знаходити, вивчати та порівнювати утиліти для моніторингу стану ПК за ключовими параметрами.

Основні завдання:

1. Аналіз функціональних характеристик утиліт;
2. Створення структури та розробка вебсайту;
3. Реалізація функціоналу порівняння, відгуків, рейтингу;
4. Забезпечення швидкодії, кросплатформенності та адаптивності сайту.

Для забезпечення повноти оцінки до кожної утиліти було застосовано такі критерії:

- Наявність функцій моніторингу температури, частоти, напруг;
- Зручність користувацького інтерфейсу;
- Вплив на продуктивність системи;
- Актуальність оновлень;
- Модель розповсюдження (безкоштовна чи платна) [1; 2; 4].

Проект реалізовано у вигляді вебсайту з адаптивним дизайном і модульною структурою.

Використані технології:

- Frontend: HTML, CSS, JavaScript (реалізація адаптивності, інтерактивності, таблиць порівняння);
- Backend: PHP (робота з формами, базою даних, рейтингами);
- Сторінки сайту: головна, каталог утиліт, окремі сторінки оглядів, форма порівняння, сторінка з відгуками.

Функціональність:

- каталог з фільтрацією за параметрами;
- детальні сторінки з описами програм;
- інтерактивна таблиця для порівняння 2–3 обраних утиліт;
- форма залишення коментарів та оцінок.

Нефункціональні вимоги: оптимізація коду, швидке завантаження сторінок, безпечне оброблення запитів, наявність SSL-захисту.

Після реалізації сайт було протестовано на різних браузерях (Chrome, Firefox, Edge), пристроях (ПК, планшет, смартфон). Особлива увага приділялася:

- правильному відображенню контенту;
- стабільності динамічного порівняння;
- захисту даних користувача.

Висновки

У результаті реалізації проекту:

- проаналізовано основні популярні утиліти моніторингу стану ПК;

- створено повнофункціональний сайт, який дозволяє здійснювати швидке порівняння та вибір утиліти під конкретні потреби;
- реалізовано інтерфейсну частину з урахуванням сучасних вимог юзабіліті;
- сайт може бути легко адаптований до інших категорій програмного забезпечення або технічних сервісів.

Розроблений ресурс має практичну цінність для користувачів, які шукають оптимальні інструменти для контролю роботи свого ПК, і може бути масштабований для професійного IT-супроводу в сервісних компаніях.

Література

1. Бондаренко І. В. Методи та засоби моніторингу апаратного забезпечення комп'ютерних систем // Вісник НТУУ «КПІ». – 2022.
2. Романенко П. В. Аналіз інструментів для моніторингу температурних показників комп'ютерної системи // Комп'ютерні науки та інформаційні технології. – 2021.
3. Офіційна документація CPUID HWMonitor. – [Електронний ресурс]. – URL: <https://www.cpubid.com/software/hwmonitor.html>
4. Документація AIDA64. – [Електронний ресурс]. – URL: <https://www.aida64.com>
5. Open Hardware Monitor User Guide. – [Електронний ресурс]. – URL: <https://openhwaremonitor.org>

ПРОГРАМНА РЕАЛІЗАЦІЯ САЙТУ ЗІ СТАТИСТИКОЮ ІГРОВИХ ПРОФІЛІВ

О.В. Кряж, бакалавр,
kriazh.sasha@gmail.com,

О. О. Черненко, доцент, канд. фіз.-мат. наук,
Полтавський університет економіки і торгівлі

В роботі розглядається проблема моніторингу статистики профілів за допомогою веб-застосунків для подальшої торгівлі ігровими акаунтами. Проаналізовано вимоги і особливості сайтів, які використовують інтерфейс програмування застосунків для отримання загальної і особистої статистики.

Kriazh O.V. Software implementation of a website with statistics of gaming profiles. The article discusses the problem of monitoring profile statistics using web applications for further trading of gaming accounts. The requirements and features of websites that use the application programming interface to obtain general and personal statistics are analyzed.

Ключові слова: ВЕБ-ДОДАТОК, API, PYTHON, DESTINY 2

Keywords: WEB APPLICATION, API, PYTHON, DESTINY 2

Протягом останніх десяти років у сфері комп'ютерних розваг розвивається новий напрям торгівлі цифровими продуктами. Об'єктом продажу можуть бути внутрішньоігрові предмети, допомога від досвідчених гравців або ігрові акаунти. У такій грі, як Destiny 2, збір інформації, яка повинна зацікавити потенційних покупців ускладнює процес торгівлі для продавців. Створення оголошення для одного акаунту може займати багато часу. Відсутність інструменту для швидкого перегляду необхідної

інформації може сповільнити роботу і знижує зацікавленість покупців.

У такій ситуації зручний веб-застосунок з динамічним оновленням інформації покращить якість процесу для усіх його учасників, підвищить швидкість угод і загальну якість торгівлі, а також стане зручним інструментом для не комерційного застосування звичайними користувачами для перегляду власної статистики.

Метою даної бакалаврської роботи є розробка та програмна реалізація веб-застосунку для інтерактивного перегляду ігрових профілів у зручному форматі. Такий інструмент оптимізує процес створення оголошень та покращує навігацію у перегляді товарів для покупців.

Головні вимоги для веб-застосунку, це зрозумілий, мінімалістичний інтерфейс та оновлення даних у режимі реального часу, завдяки доступу до бази даних Bungie API.

Розробка складалася з таких етапів:

- Аналіз потреб користувачів;
- Вибір технологій для реалізації (Python, Flask, JavaScript, HTML/ CSS);
- Проектування бекенду з інтеграцією Bungie API;
- Створення клієнтської частини з пошуком і переглядом профілів у мінімалістичному стилі.

Веб-застосунок дозволяє користувачу швидко знайти будь-який ігровий профіль та переглядати таку інформацію, як: придбані доповнення (DLC), кількість унікального (екзотичного) екіпірування та відкриті стихійні підкласи, а також відстежувати оновлення даної інформації у реальному часі.

Проект має потенціал для розширення у вигляді збільшення предметів статистики, що відстежується та порівняння профілів між собою прямо у веб-застосунку за допомогою паралельних запитів.

Розроблений веб-застосунок було випробувано на практиці. Продавці відмітили значне зменшення часу, витраченого на створення оголошень до акаунтів, зростання інтересу з боку клієнтів і оцінили лаконічний дизайн.

Підсумовуючи, використання веб-застосунків для збору статистики ігрових профілів покращує якість торгівлі ігровими

акаунтами, підвищує ефективність взаємодії між продавцями та покупцями, а також забезпечує зручний доступ до персональної статистики звичайним користувачам.

Література

1. Bungie API [Електронний ресурс]: Bungie, Inc. Bungie.net API Documentation version 2.20.1, 2025, URL: <https://bungie-net.github.io/multi/index.html>.
2. Mark Lutz. Learning Python 5th Edition / Mark Lutz - O'Reilly Media – 2013 - 1640 с. - ISBN: 978-1-449-35573-9
3. Leonard Richardson. RESTful Web APIs: Services for a Changing World / Leonard Richardson, Mike Amundsen, Sam Ruby - O'Reilly Media - 2013 - 406 с. - ISBN: 9781449359737.
4. Mat Marquis. Javascript For Web Designers / Mat Marquis - A Book Apart - 2016 - 135 с. - ISBN-13: 978-1937557461.
5. Patrick Viafore. Robust Python: Write Clean and Maintainable Code 1st Edition / Patrick Viafore - O'Reilly Media - 2021 - 378 с. - ISBN-13: 978-1098100667.

РОЛЬ АДАПТИВНИХ НАВЧАЛЬНИХ ПРОГРАМ У ПОТЕНЦІАЛІ ДИСТАНЦІЙНОЇ ОСВІТИ

Д.В. Кіліян, студент КН м-11, спеціальності «Комп'ютерні науки»,

*Ю.Ф. Олексійчук, науковий керівник, к ф.-м. наук,
Полтавський університет економіки і торгівлі*

У наш час цифрові технології проникають у всі сфери життя. Неупинно цей процес розвивається і у навчанні. Онлайн-навчання стало невід'ємною частиною сучасного освітнього процесу [2; 4], надаючи доступ до знань широкому колу людей, незалежно від їхнього місця розташування та віку. Разом з цим розвитком зростає і роль інформаційних технологій, зокрема, мов програмування [5]. Вони використовуються для розробки веб-додатків, мобільних додатків, десктопних додатків та, що не менш важливо, програм для дистанційної освіти. Володіння мовами програмування дозволяє студентам отримати цінні навички, які будуть жаданими у наш час.

Попит на освоєння мов програмування кожного дня зростає. Пропозиція котра спроможна відповісти на зростаючий попит - навчальні програми. Інтуїтивні та доступні, їх може освоїти кожен, але зазвичай вони не враховують рівень знань користувача, що сильно знижує мотивацію та заважає його прогресу. Цю проблему вирішують адаптивні навчальні програми [1; 3].

Адаптивна навчальна програма дозволяє кожному студенту рухатися в своєму темпі, вивчаючи матеріал, який є найбільш релевантним для його потреб. Адаптивний підхід дозволяє підвищити ефективність навчання завдяки індивідуалізації, персоналізації та постійному зворотному зв'язку. Студенти можуть отримувати більш точні та детальні пояснення складних тем, а також виконувати практичні завдання, які відповідають їхньому рівню знань.

Особливості адаптивних навчальних програм:

1. Індивідуалізація: Кожен студент отримує навчальний матеріал, який відповідає саме його рівню знань та темпу навчання.

2. Динамічність: Навчальний план постійно змінюється в залежності від прогресу студента. Якщо він швидко освоює матеріал, програма ускладнює завдання, а якщо виникають труднощі – спрощує.

3. Автоматизація: Багато рутинних процесів, таких як оцінювання завдань та підбір матеріалу, автоматизуються, що дозволяє викладачеві зосередитися на більш творчих завданнях.

4. Інтерактивність: Адаптивні програми часто включають інтерактивні елементи, такі як симуляції, ігри та вікторини, що робить навчання більш цікавим та ефективним.

Сучасна освіта все більше переходить у цифровий простір, а навчальне програмне забезпечення стає її невід'ємною частиною. Її інструменти не лише дозволяють отримувати доступ до знань в будь-який час і з будь-якої точки світу, але й відкривають нові можливості для інтерактивного навчання та персоналізації освітнього процесу [4; 6]. Якісно розроблене навчальне програмне забезпечення є ключем до успішного дистанційного навчання. Воно забезпечує доступ до структурованого навчального матеріалу, різноманітні інтерактивні завдання та зручні інструменти для спілкування з викладачем та одногрупниками. Завдяки використанню сучасних технологій, онлайн-платформи дозволяють створити ефективне навчальне середовище, яке не поступається традиційному очному навчанню.

Література

1. Грибанова, С. А. (2024). Педагогічна концепція цифрового інтерактивного навчання для розвитку пізнавальної діяльності. У Авіація, промисловість, суспільство: матеріали V міжнар. наук.-практ. конф.(м. Кременчук, 16 трав. 2024 р.) – Харків: ХНУВС. – с. 367-370.
2. В.І. Соловйов "Дистанційне навчання: теорія, технології, практика" – 2006. – 448 с.
3. Сергій Омельчук “Дистанційне навчання студентів: теорія і практика” – 2020. – 78с.

4. Вільям Андерсон та Майкл Грехам Мур "The Handbook of Distance Education". – 2003. – 898с.
5. Олексійчук Ю. Ф. – Розробка та впровадження дистанційного курсу з дисципліни «Програмування» / Ю. Ф. Олексійчук // Дистанційна освіта: забезпечення доступності та неперервної освіти впродовж життя (e-learning and university education-2017): матеріали XLII Міжнародної науково-методичної конференції (м. Полтава, 9–10 лютого 2017 року) – Полтава: ПУЕТ, 2017. – с. 167-169.
6. Мазур М.П. Розвиток дистанційного навчання в Україні як складової інформатизації сучасного суспільства / М.П.Мазур // Інформатика та інформаційні технології в навчальних закладах. – 2007.

РОЗРОБКА САЙТУ ІНТЕРНЕТ-МАГАЗИНУ «KIDDO»

Соколов А.В., студент групи КН-б-41, спеціальності
«Комп'ютерні науки»,

Олексійчук Ю.Ф., науковий керівник, кандидат фізико-
математичних наук,

Полтавський університет економіки і торгівлі

Ключові слова: WORDPRESS, WOOCOMMERCE, ІНТЕРНЕТ-
МАГАЗИН, UX/UI, CMS, ДИТЯЧІ ТОВАРИ, PHP,
АДАПТИВНИЙ ДИЗАЙН

Інтернет-магазини стали одним із найпоширеніших засобів реалізації товарів у цифрову епоху. Зокрема, попит на товари для дітей стрімко зростає, і відповідно підвищується актуальність розробки зручних, безпечних та інформативних вебресурсів, орієнтованих саме на батьків як основну цільову аудиторію. У межах цієї кваліфікаційної роботи було поставлено завдання реалізувати повнофункціональний інтернет-магазин, який би задовольняв вимоги до адаптивного дизайну, високої швидкодії, зручної структури та простоти в адмініструванні.

В якості платформи для розробки обрано CMS WordPress [1; 3] у поєднанні з плагіном WooCommerce [2], що забезпечує гнучкість, розширюваність і мінімальний поріг входу як для розробників, так і для адміністраторів магазину. Таке рішення дозволило швидко розгорнути базову архітектуру системи та сконцентруватися на налаштуванні інтерфейсної та логічної складової.

У результаті було створено інтерфейс головної сторінки з адаптивним слайдером, каталог із динамічним фільтруванням і сортуванням, картки товарів з детальним описом, кошик із

підтримкою зміни кількості товарів і сторінку оформлення замовлення зі зручною формою. Кожен елемент був протестований на мобільних пристроях, що забезпечило повну відповідність сучасним вимогам UX/UI [4].

В особистому кабінеті реалізовано можливість перегляду історії замовлень, редагування особистих даних та керування налаштуваннями. Для адміністратора сайту було розгорнуто контрольну панель, яка дозволяє додавати товари, змінювати ціни, керувати замовленнями, переглядати відгуки тощо.

Окрему увагу приділено **питанням безпеки**: реалізовано захищене з'єднання HTTPS на основі Let's Encrypt, увімкнено проксірування трафіку через Cloudflare для захисту від DDoS-атак та кешування, застосовано плагіни безпеки (iThemes Security) і SEO-оптимізації (Yoast SEO) [1; 2].

З технічної точки зору, серверна частина побудована на стеку Apache + MySQL, підтримано автоматизацію налаштувань через SSH, налаштовано резервне копіювання бази даних та безпечне оновлення ядра CMS і плагінів.

Проект містить також інструкцію для користувача, що описує процес реєстрації, вибору товару, оформлення замовлення та взаємодії з персональним кабінетом. Цей документ супроводжується скріншотами, що особливо корисно для нових користувачів, які не мають технічної підготовки.

Таким чином, результатом роботи став готовий до комерційного використання інтернет-магазин, який поєднує простоту управління, сучасний дизайн, функціональність і відповідність вимогам безпеки. Проект може бути масштабований та адаптований до різних типів товарів або бізнес-моделей.

Література:

1. Офіційна документація WordPress [Електронний ресурс]. – Режим доступу: <https://wordpress.org/support/>.
2. Документація WooCommerce [Електронний ресурс]. – Режим доступу: <https://woocommerce.com/documentation/>.
3. Українська спільнота WordPress [Електронний ресурс]. – Режим доступу: <https://uk.wordpress.org/>.

4. Elementor: Як створити веб-сайт на WordPress
[Електронний ресурс]. – Режим доступу:
<https://elementor.com/help/>.

РОЗРОБКА САЙТУ ІНТЕРНЕТ МАГАЗИНУ «ОТАК»

Я. О. Тобота, здобувач освіти КН б-41,
Т. В. Чілікіна, доцент кафедри комп'ютерних наук та
інформаційних технологій, к. ф.-м. н.,
Полтавський університет економіки і торгівлі

У сучасних умовах активного розвитку електронної комерції створення інтернет-магазинів набуває особливого значення. Зважаючи на зростаючу потребу у швидких, адаптивних та легких у використанні веб-застосунках, було розроблено навчальний проект інтернет-магазину «Отак», який реалізує базову функціональність сучасного e-commerce ресурсу.

Метою роботи є створення клієнтської частини сайту інтернет-магазину з використанням сучасних вебтехнологій, яка демонструє логіку взаємодії з API, ефективну роботу з локальними даними та адаптивний дизайн [1].

Основні етапи реалізації проекту:

- побудова структури застосунку з урахуванням принципів модульності;
- підключення до тестового API для завантаження асортименту товарів [2];
- створення шаблонних сторінок для відображення категорій та детальної інформації про товари;
- реалізація кешування даних у локальному сховищі браузера для підвищення швидкості завантаження та зменшення навантаження на API [3];
- розробка функціоналу «Кошик» з можливістю додавання, видалення та редагування кількості товарів;
- створення інтерфейсу оформлення замовлення з валідацією даних.

Особливу увагу приділено адаптивності інтерфейсу, що забезпечує коректне відображення на різних типах пристроїв, зокрема на мобільних телефонах та планшетах. Дизайн інтерфейсу відповідає актуальним вимогам user-friendly та доступності [4].

Проект не передбачає використання серверної бази даних і платіжної системи, що дозволяє зосередитися на демонстрації клієнтської логіки, структури SPA (Single Page Application) та сучасних технологій розробки [5]. Такий підхід дозволяє використовувати проект як навчальний приклад або як прототип для подальшої комерційної реалізації.

У результаті реалізації проекту створено повнофункціональний макет інтернет-магазину з базовим функціоналом і гнучкою архітектурою, що може бути доопрацьована для впровадження серверної частини, підключення реальних платіжних систем і управління замовленнями.

Розроблений веб-застосунок демонструє ефективне використання сучасного стеку технологій для побудови фронтенд-рішень у сфері електронної комерції.

Література

1. Поліщук О. М. Основи веб-програмування : навч. посіб. / О. М. Поліщук. Київ : КНЕУ, 2020. 192 с.
2. Zakas, N. C. (2021). Understanding ECMAScript 6: The Definitive Guide for JavaScript Developers. No Starch Press.
3. Li, H., & Li, H. (2020). Progressive Web Apps with React: Create fast, reliable, and engaging user experiences for the web. Packt Publishing.
4. Krug, S. (2014). Don't Make Me Think, Revisited: A Common Sense Approach to Web Usability. New Riders.
5. Flanagan, D. (2020). JavaScript: The Definitive Guide. O'Reilly Media.

РОЗРОБКА WEB-СЕРВІСУ ДЛЯ ІНТЕРАКТИВНОГО ОЦІНЮВАННЯ СТУДЕНТІВ З ВИКОРИСТАННЯМ NLP- МОДЕЛЕЙ

В. Тур, аспірант

О.В. Ольховська, кандидат фізико-математичних наук, доцент
кафедри комп'ютерних наук та інформаційних технологій
Полтавський університет економіки і торгівлі

В роботі розглядається проблема вдосконалення процесу оцінювання знань студентів за допомогою інтерактивного web-сервісу, який використовує сучасні моделі обробки природної мови (NLP). Розроблений підхід дозволяє автоматизувати аналіз відповідей студентів, підвищуючи точність та об'єктивність оцінювання.

Ключові слова: ІНТЕРАКТИВНЕ ОЦІНЮВАННЯ, WEB-SERVICE, ОБРОБКА ПРИРОДНОЇ МОВИ, NLP.

Keywords: INTERACTIVE ASSESSMENT, WEB SERVICE, NATURAL LANGUAGE PROCESSING, NLP.

У сучасному світі цифровізації освіти одним із головних викликів є створення ефективних інструментів для оцінювання знань студентів. Традиційні методи оцінювання, такі як письмові роботи, тестування або усні заліки, можуть бути трудомісткими, суб'єктивними та часозатратними. Використання інтерактивних web-сервісів з інтеграцією сучасних NLP-моделей є інноваційним рішенням, яке дозволяє автоматизувати процес оцінювання, забезпечуючи об'єктивність та високу точність.

Одним із ключових аспектів такого підходу є обробка природної мови (Natural Language Processing, NLP). Ця технологія дозволяє аналізувати текстові відповіді студентів у режимі

реального часу, виявляючи не лише їхню правильність, а й логіку, глибину аргументації та якість викладу. Проблема полягає в тому, що традиційні алгоритми оцінювання не здатні враховувати контекст і семантику відповідей, що часто призводить до недооцінки роботи студента.

NLP-моделі, такі як BERT (Bidirectional Encoder Representations from Transformers) або GPT (Generative Pre-trained Transformer), відкривають нові можливості для аналізу тексту. Вони здатні розуміти контекст відповіді студента, аналізувати синтаксичну та семантичну структуру тексту, визначати ключові поняття та відповідність відповіді навчальним вимогам. Використання таких моделей дозволяє системі оцінювання бути більш адаптивною до різних стилів викладу студентів, а також враховувати індивідуальні особливості їхнього мислення.

Інтерактивний web-сервіс для оцінювання знань на основі NLP-моделей забезпечує об'єктивність та виключає людський фактор, що сприяє підвищенню довіри студентів до системи. Автоматичне оцінювання здійснюється швидко, що дозволяє аналізувати відповіді в реальному часі та суттєво економити час викладачів. Масштабованість таких систем дозволяє використовувати їх для великих груп студентів, особливо у форматі дистанційного навчання. Сервіс також може надавати студентам персоналізовані рекомендації, аналізуючи їхні помилки та пропонуючи додаткові матеріали для поглиблення знань.

Особливу увагу слід приділити аспектам конфіденційності. Сучасні web-сервіси зберігають дані студентів у зашифрованому вигляді, що забезпечує відповідність стандартам захисту персональної інформації. Крім того, хмарні технології дозволяють зберігати великі обсяги даних без зниження продуктивності системи.

Інтеграція NLP-моделей у web-сервіси для оцінювання знань сприяє розвитку компетенцій студентів у формулюванні чітких, логічно структурованих відповідей. Система може визначити не лише правильність, але й якість відповіді, враховуючи її аргументованість, що мотивує студентів покращувати навички письмової комунікації.

У майбутньому такі системи можуть стати основою для розробки комплексних платформ, які не лише оцінюють знання студентів, але й формують їхній освітній шлях, враховуючи сильні та слабкі сторони. Завдяки інтерактивним web-сервісам з використанням NLP-моделей освіта стає більш персоналізованою, ефективною та доступною для широкого кола студентів. Розробка таких сервісів є важливим кроком у цифровізації освіти, забезпечуючи об'єктивність, ефективність та індивідуальний підхід до навчання, що допомагає студентам розкрити свій потенціал та досягати кращих результатів.

Література

1. Devlin, J., Chang, M.-W., Lee, K., & Toutanova, K. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. *arXiv preprint*, 2018. Available at: <https://arxiv.org/abs/1810.04805>.
2. Brown, T. B., Mann, B., Ryder, N., et al. Language Models are Few-Shot Learners. *arXiv preprint*, 2020. Available at: <https://arxiv.org/abs/2005.14165>.
3. Zhang, Y., Chen, X., & Yang, Q. Automatic Short Answer Grading Using Natural Language Processing Techniques. *IEEE Transactions on Learning Technologies*, 2020. DOI: 10.1109/TLT.2020.3024876.
4. Vaswani, A., Shazeer, N., Parmar, N., et al. Attention is All You Need. *Advances in Neural Information Processing Systems*, 2017. Available at: <https://arxiv.org/abs/1706.03762>.
5. NLP in Assessments – Revolutionizing Education and Evaluation. *Excelsoft*, 2023. Available at: <https://www.excelsoftcorp.com/blog-post/nlp-in-assessments-revolutionizing-education-and-evaluation/>.

ПРОЕКТУВАННЯ ВЕБ-СТОРІНКИ ЗА ДОПОМОГОЮ МОВИ ПРОГРАМУВАННЯ PYTHON ТА ФРЕЙМВОРКУ FLASK

Д. А. Фісун, студент спеціальності «Комп'ютерні науки», група КНб-41,

Т. В. Чілікіна, науковий керівник кандидат фізико-математичних наук, доцент кафедри комп'ютерних наук та інформаційних технологій

Полтавський університет економіки і торгівлі

Ключові слова: *Flask, Python, тестування, вебсайт, база даних, UI/UX, SQLite3*

Keywords: *Flask, Python, testing, website, database, UI/UX, SQLite3*

У світі, де цифрова трансформація торкається практично всіх сфер життя, особливу роль відіграють вебзастосунки, які поєднують у собі зручність доступу, платформену незалежність та інтерактивність. Однією з актуальних задач сучасної освіти є створення інструментів для електронного тестування та оцінювання знань. У цьому контексті особливої популярності набуває мова програмування Python, яка, завдяки своїй простоті та великій кількості бібліотек, використовується для створення повноцінних вебдодатків за допомогою фреймворків, таких як Flask [1].

Мета - розробка повнофункціонального веб-застосунку, який дозволяє користувачам проходити тестування з теми "Штучний інтелект", фіксувати результати, працювати з інтерфейсом персонального кабінету, використовуючи серверну логіку на Flask і базу даних SQLite3.

Завдання:

1. Реалізувати систему реєстрації/авторизації користувачів.

2. Створити структуру тестування з випадковою генерацією варіантів.
3. Організувати збереження даних у базі SQLite3.
4. Розробити адаптивний інтерфейс із використанням Bootstrap.
5. Забезпечити вивід аналітики результатів користувача.

Інструменти та технології.

1. Flask — мікрофреймворк для Python, який дозволяє створити серверну логіку, маршрути, обробку POST/GET-запитів [1].
2. Python 3 — мова, що підтримує всі необхідні функції обробки даних та логіки [5].
3. SQLite3 — легка реляційна база даних, яка інтегрується у Flask безпосередньо, не потребує окремого сервера [2].
4. Jinja2 — шаблонізатор для Flask, що дозволяє вбудовувати динамічні елементи в HTML [3].
5. Bootstrap 4 — CSS-фреймворк для створення адаптивного, візуально привабливого дизайну [4].

Функціональна структура вебзастосунку.

1. Головна сторінка (/)

Містить вітальний текст, короткий опис функцій платформи та кнопку «Зареєструватися» або «Увійти».

2. Реєстрація користувача (/register)

Користувач заповнює форму: ім'я, прізвище, номер групи. Дані зберігаються в таблиці users бази даних SQLite. Після реєстрації користувач автоматично перенаправляється на сторінку тесту.

3. Проходження тесту (/test)

Тест включає 10 питань на тему штучного інтелекту. Особливості:

- варіанти відповідей перемішуються;
- неправильна відповідь супроводжується короткою підказкою;
- всі відповіді зберігаються разом з ідентифікатором користувача.

4. Обробка результатів (/test_results)

Система підраховує кількість правильних/неправильних відповідей, відображає пояснення, та зберігає результати у таблиці test_results.

5. Особистий кабінет (/profile)

Містить історію проходження тестів, загальний бал, список пройдених тем, можливість повторного проходження.

Переваги та інновації.

- Адаптивність: Сайт коректно працює на ПК, планшетах, смартфонах.
- Персоналізація: Кожен користувач має свій обліковий запис.
- Безпека: Реалізоване базове шифрування даних форм, захист від SQL-ін'єкцій.
- Освітня цінність: Тематика III актуалізує тестування в контексті сучасних IT-трендів.

Під час тестування вебзастосунок було перевірено на предмет:

- стабільності роботи серверної частини;
- кросбраузерності (Chrome, Firefox, Edge);
- відгуку інтерфейсу на мобільних пристроях;
- точності обліку результатів;
- захисту від несанкціонованого доступу.

Всі модулі працювали коректно. Опитування 10 тестових користувачів показало, що 90% із них оцінили інтерфейс як зручний, а навігацію — як інтуїтивно зрозумілу.

Розроблений вебзастосунок демонструє приклад ефективного поєднання сучасних IT-інструментів, методик навчання та вебтехнологій. Платформа може бути використана:

- у навчальному процесі як модуль для самоперевірки;
- для організації тестування з інших тем;
- у процесі підготовки до сертифікацій.

У майбутньому планується реалізувати:

- адміністративну панель для викладача;
- вивантаження результатів у форматі PDF;
- інтеграцію з чат-ботами для Telegram або Viber.

Лінепамыпа

1. Flask Documentation – <https://flask.palletsprojects.com/en/stable/>
2. SQLite3 Official Documentation – <https://www.sqlite.org/docs.html>
3. Jinja2 Templates – <https://jinja.palletsprojects.com/en/stable/>
4. Bootstrap v4.1 Getting Started – <https://getbootstrap.com/docs/4.1/getting-started/introduction/>
5. Python 3 Official Documentation – <https://docs.python.org/3/>

ОСОБЛИВОСТІ РОЗРОБКИ ВЕБ-САЙТУ ДЛЯ РЕМОНТУ ТА ПОШИВУ ОДЯГУ З ВИКОРИСТАННЯМ LARAVEL 12

*Г. В. Халявка, здобувач освіти, КНб-41,
О. П. Кошова, к. пед. н., доцент кафедри комп'ютерних наук та
інформаційних технологій,
Полтавський університет економіки і торгівлі*

У публікації розглядаються особливості створення веб-сайту для ремонту та пошиву одягу з використанням сучасного фреймворку Laravel 12.

Khaliavka H., Koshova O., Peculiarities of developing a website for clothing repair and tailoring using Laravel 12 are considered in this article.

Ключові слова: LARAVEL 12, ГРУПУВАННЯ МАРШРУТІВ, TAILWIND CSS, САЙТ РЕМОНТУ ОДЯГУ, ВЕБ-РОЗРОБКА.

Keywords: LARAVEL 12, ROUTE GROUPING, CLOTHING REPAIR WEBSITE, TAILWIND CSS, WEB DEVELOPMENT.

У сучасних веб-застосунках маршрутизація відіграє ключову роль у створенні зручного та інтуїтивного інтерфейсу. Laravel 12 пропонує гнучкий, але структурований механізм маршрутизації, що дозволяє групувати маршрути, застосовувати префікси та middleware. Такий підхід допомагає логічно розподілити доступ за ролями і водночас зробити URL зрозумілими без зайвих дублювань. На сучасному етапі розвитку веб-технологій важливою складовою успішного веб-проєкту є не лише його функціональність, а й візуальна привабливість, адаптивність та зручність інтерфейсу для користувачів. Саме тому розробники дедалі частіше використовують сучасні CSS-фреймворки, серед яких особливої популярності набув Tailwind CSS. У поєднанні з

потужністю PHP-фреймворку Laravel 12, Tailwind дає змогу створювати не лише ефективні, але й естетично привабливі веб-застосунки.

Якщо подивитися на приклад використання `Route::middleware(['auth', 'verified'])`, то видно, що всі перевірки автентифікації та підтвердження email зібрані в одному місці. Це означає, що кожен маршрут у цій групі гарантує: користувач не тільки авторизований, а й підтвердив свою електронну пошту. Такий підхід усуває повторюваність коду та зменшує ризик помилок.

Маршрути згруповані логічно. Користувач починає з дашборду (dashboard), де відображаються його попередні замовлення, сповіщення та рекомендації (рис. 1.). Далі він може перейти до сторінки створення нового замовлення, а всі URL побудовані за стандартними REST-паттернами, що робить навігацію зрозумілою.

```
//
// Маршрути для авторизованих користувачів
Route::middleware(middleware: ['auth', 'verified'])->group(callback: function () { void {
    // Головна сторінка користувача
    Route::get(uri: 'dashboard', action: [UserController::class, 'dashboard'])->name(name: 'dashboard');
    // Створення замовлення
    Route::get(uri: 'create', action: [OrderController::class, 'showCreateForm'])->name(name: 'create-order');
    Route::post(uri: 'create', action: [OrderController::class, 'store'])->name(name: 'store-order');

    // Маршрути для замовлень
    Route::controller(controller: OrderController::class)
        ->prefix(prefix: 'orders')
        ->name(name: 'orders')
        ->group(callback: function () { void {
            Route::get(uri: '/', action: 'index')->name(name: 'index'); // Список замовлень
            Route::get(uri: '/{order}', action: 'show')->name(name: 'show'); // Перегляд замовлення
            Route::post(uri: '/', action: 'store')->name(name: 'store'); // Збереження замовлення
            Route::post(uri: '/{order}/cancel', action: 'cancel')->name(name: 'cancel'); // Скасування замовлення
        });
    Route::post(uri: '/orders/{order}/cancel', action: [OrderController::class, 'cancel'])->name(name: 'orders.cancel');
    // Редагування замовлення
    Route::get(uri: '/orders/{order}/edit', action: [OrderController::class, 'edit'])->name(name: 'orders.edit');

    // Оплата замовлення
    Route::get(uri: '/orders/{order}/payment', action: [PaymentController::class, 'showPaymentForm'])->name(name: 'orders.payment');
    Route::post(uri: '/orders/{order}/pay-online', action: [PaymentController::class, 'payOnline'])->name(name: 'orders.payOnline');
});
});
```

Рис. 1. -Маршрути сторінок на сайті з ремонту та пошиву одягу
Адреси виглядають логічно: `/create` — створення нового замовлення, `/orders` — управління замовленнями. Тут же знаходиться весь функціонал, пов'язаний із замовленнями: список, деталі, редагування та скасування. Завдяки такій структурі розробникам і дизайнерам легше працювати з кодом: шаблони кнопок та посилань використовують іменовані маршрути (`route('orders.show', $order)`), а при зміні структури достатньо оновити один виклик у конфігурації маршрутизації.

З часом у будь-якому проєкті з'являються нові вимоги: програма лояльності, інтеграція з службами. Завдяки правильному розподілу префіксів і middleware нові маршрути додаються легко, не впливаючи на існуючі.

Наприклад, для замовлень можна створити окрему групу:

```
Route::prefix('orders')    ->middleware(['auth','verified']) -  
>group(function() { ... });
```

Коли в проєкті працює велика команда, важливо, щоб нові розробники могли швидко розібратися в коді. Чітке групування маршрутів та єдині стандарти іменування суттєво спрощують процес навчання. Достатньо переглянути `routes/web.php`, щоб зрозуміти структуру проєкту.

- Менше дублювань коду — завдяки централізованому використанню middleware

- Зрозуміла структура — маршрути логічно згруповані

- Гнучкість — легко додавати нові ролі та розділи

- Безпека — виключає ризик пропущених перевірок

Грамотна маршрутизація в Laravel 12 створює міцний фундамент для надійного, зручного та масштабованого сервісу. Це вигідно і для клієнтів, і для розробників.

У межах розробки сайту з ремонту та пошиву одягу, Tailwind CSS був інтегрований у проєкт на Laravel 12 як основний інструмент для стилізації інтерфейсу. Tailwind — це утилітарний CSS-фреймворк, що дозволяє будувати інтерфейси напряму в HTML-кодi, без потреби створювати окремі CSS-файли з класами. Це значно пришвидшує розробку, зменшує кількість CSS-коду та сприяє кращому контролю над візуальним оформленням елементів.

Інтеграція Tailwind у Laravel здійснюється через Laravel Mix або Vite (рис. 2.), що дозволяє автоматично компілювати CSS з урахуванням використаних класів. Завдяки цьому, підсумковий розмір стилів значно менший, ніж у класичних фреймворках, таких як Bootstrap. У моєму випадку використовувався Vite як сучасний інструмент збірки, який підтримується з версії Laravel 9 і далі.

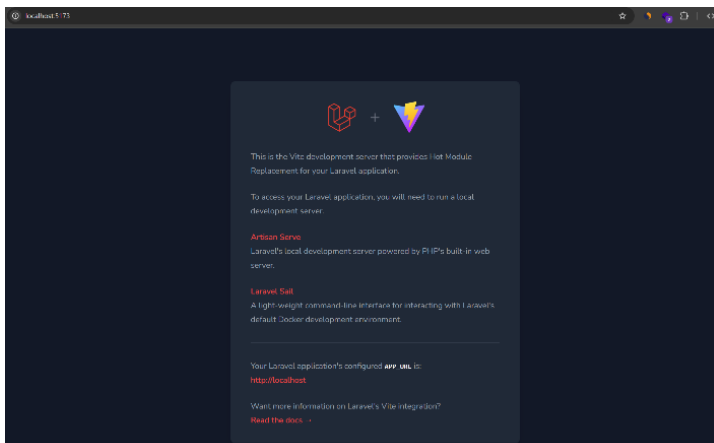


Рис. 2. -Запущений сервер Vite

Основні компоненти сайту — такі як форма створення замовлення, каталог послуг, панель адміністратора — були оформлені за допомогою Tailwind класів. Наприклад, таблиці замовлень, кнопки підтвердження, повідомлення про успішну дію були реалізовані з використанням класів `bg-violet-500`, `text-sm`, `rounded-xl`, `shadow-md`, `hover:bg-slate-500` тощо, що надало сайту сучасного вигляду без перевантаження сторонніми бібліотеками.

Tailwind також дозволяє легко реалізувати адаптивність інтерфейсу, забезпечуючи коректне відображення елементів на мобільних пристроях. Наприклад, через медіакласи `sm:`, `md:`, `lg:`.

Ще однією перевагою Tailwind є наявність плагінів (наприклад, для форм, типографіки, анімацій), які були використані у проєкті для покращення візуальних ефектів — таких як плавна зміна кольорів, кнопки, тощо (рис. 3).

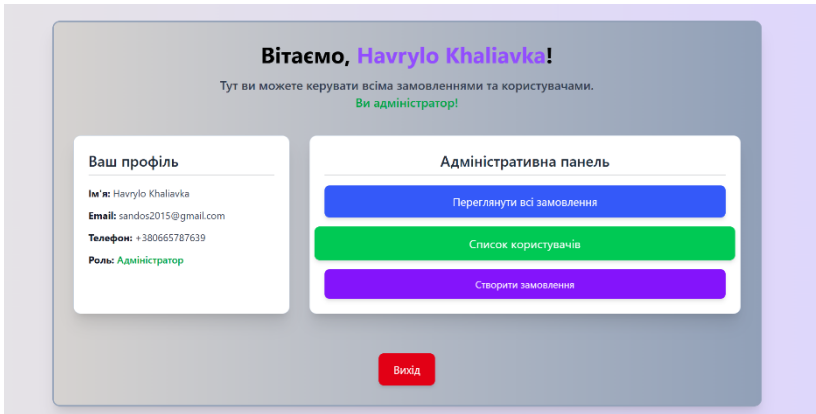


Рис. 3. - Кнопки на адмін панелі сайту веб-сайту для ремонту та пошиву одягу

Загалом, поєднання Laravel 12 і Tailwind CSS дозволило розробити сучасний, легкий, адаптивний і зручний сайт для надання послуг ремонту та пошиву одягу. Це рішення дало змогу зосередитись на функціоналі та логіці сайту, не витрачаючи зайвого часу на стилізацію.

Таким чином, використання Tailwind CSS у проєкті на Laravel — це сучасний підхід до фронтенд-розробки, який ідеально підходить для мікросервісів, додатків та веб-сайтів малого і середнього бізнесу, що вимагають як функціональності, так і візуальної привабливості.

Література

1. Laravel Docs URL: <https://laravel.com/docs/12.x>
2. Laravel Daily URL: <https://laraveldaily.com/lesson/laravel-from-scratch/admin-users-route-groups-middleware>
3. PHP Docs URL: <https://www.php.net/docs.php>
4. TailwindCSS Docs URL: <https://tailwindcss.com/>

РОЗРОБКА ЧАТ-БОТА ДЛЯ НАДАННЯ КОНСУЛЬТАЦІЙ В СФЕРІ КІБЕРБЕЗПЕКИ

*Д. А. Храпко, студент спеціальності «Комп'ютери науки», група
КН 6-41*

*О. П. Кошова, к.ф.-м. н., кафедри комп'ютерних наук та
інформаційних технологій*

Полтавський університет економіки і торгівлі

У сучасних умовах стрімкої цифровізації та зростання кіберзагроз актуальність швидкого й доступного консультування з питань кібербезпеки стає критичною. Метою даної роботи є розробка інтелектуального чат-бота для надання користувачам оперативних консультацій щодо захисту від кібератак, що дозволить зменшити ризики втрати даних, фінансових збитків та підвищити цифрову грамотність населення.

Актуальність теми обумовлена дефіцитом кваліфікованих фахівців з кібербезпеки, високою вартістю послуг експертів та тривалим часом очікування на допомогу. Це змушує користувачів шукати рішення в ненадійних джерелах, що часто призводить до критичних помилок. Чат-бот усуває цю проблему завдяки цілодобовій доступності, миттєвому наданню стандартизованих відповідей на основі перевірених міжнародних стандартів (таких як NIST, OWASP) [1, 2] та зменшенню навантаження на фахівців.

Технологічною основою бота є штучний інтелект (ШІ) та обробка природної мови (NLP), які дозволяють системі аналізувати текстові запити користувачів, розпізнавати їхні наміри, визначати контекст проблеми та генерувати точні рекомендації [3]. Наприклад, у відповідь на запит щодо фішингової атаки бот надає покрокову інструкцію з перевірки листів, блокування шкідливих посилань і оновлення паролів. Для інтеграції з користувачами використовуються популярні платформи: Telegram, WhatsApp та веб-інтерфейси.

Ключовою перевагою рішення є його ефективність. Чат-бот економить ресурси організацій, усуваючи необхідність тримати великий штат консультантів, і масштабується для одночасної роботи з тисячами запитів. Він також виконує проактивну

функцію: попереджає користувачів про новітні загрози (наприклад, шантажні програмні забезпечення) та навчає базовим принципам безпеки (створення складних паролів, увімкнення двофакторної автентифікації).

Порівняно з традиційними методами (електронна пошта, call-центри), чат-бот забезпечує безпрецедентну швидкість відповідей (секунди замість годин очікування), консистентність інформації (немає ризику людських помилок) та здатність миттєво оновлювати базу знань у разі виявлення нових загроз. Перспективи вдосконалення системи включають аналітику запитів для виявлення тенденцій у кіберзагрозах, інтеграцію з SIEM-системами (наприклад, Splunk) для автоматизації реагування на інциденти та розширення функціоналу шляхом додавання розпізнавання зображень (наприклад, аналіз скріншотів шахрайських сайтів).

Розробка чат-бота для кібербезпеки є стратегічним кроком у боротьбі з цифровими загрозами. Він не лише надає доступну допомогу, але й формує культуру кіберобізнаності, що є фундаментом безпеки в епоху глобальних викликів.

Література

1. NIST Cybersecurity Framework. URL: <https://www.nist.gov/cyberframework>
2. OWASP Top Ten Project. URL: <https://owasp.org/www-project-top-ten/>
3. IBM: Natural Language Processing in Cybersecurity. URL: <https://www.ibm.com/cloud/learn/natural-language-processing>

**РОЗРОБКА НАВЧАЛЬНОГО ПРОГРАМНОГО
ЗАБЕЗПЕЧЕННЯ З ТЕМИ «ОСНОВИ МОВИ ЗАПИТІВ HQL»
ДИСЦИПЛІНИ «ТЕХНОЛОГІЇ ОБРОБКИ ТА АНАЛІЗУ
ДАНИХ»**

***П.В. Шаповал**, магістр спеціальності «122 Комп'ютерні науки»,
освітньої програми «Комп'ютерні науки», група КН м-11,
Полтавський університет економіки і торгівлі*

*Запропоновано програмну реалізацію навчального програмного
забезпечення з теми «Основи мови запитів HQL» дисципліни
«Технології обробки та аналізу даних».*

*Shapoval P.V. Development of educational software on the topic
"Basics of HQL query language" for the discipline "Data Processing
and Analysis Technologies"*

Ключові слова: НАВЧАЛЬНЕ ПРОГРАМНЕ ЗАБЕЗПЕЧЕННЯ,
МОВА ЗАПИТІВ HQL, АНАЛІЗ ДАНИХ.

EDUCATIONAL SOFTWARE, HQL QUERY LANGUAGE,
DATA ANALYSIS.

Сьогодні нові інформаційні технології суттєво змінюють спосіб життя людини та проникають у кожну сферу життя суспільства, включаючи освіту. В умовах зростання обсягів даних важливим є навчання студентів ефективним методам їх обробки та аналізу, а також використанню спеціалізованих мов запитів, таких як HQL (Hibernate Query Language) [1]

Основною метою проекту є розробка навчального програмного забезпечення для засвоєння базових понять мови запитів HQL. Програмне забезпечення дозволить студентам

дистанційного курсу «Технології обробки та аналізу даних» самостійно вивчати матеріал, виконувати практичні завдання та тестувати свої знання.

Програмне забезпечення містить такі функціональні можливості:

- Вивчення теоретичного матеріалу, а саме містить розділи про синтаксис HQL. Пояснення основних операторів SELECT, WHERE, JOIN, GROUP BY, ORDER BY [3]
- Практичні завдання які включають в себе написання HQL-запитів та модуль перевірки правильності запиту.
- Також присутнє формування тестів для перевірки теоретичних знань та навичок написання HQL-запитів та система оцінювання.

Перевагами розробленого програмного забезпечення є:

1. Простий дизайн, зручний для користувачів із різним рівнем технічних знань.
2. Можливість отримати миттєвий зворотний зв'язок щодо правильності виконаного завдання.
3. Студенти можуть обирати темп навчання залежно від своїх можливостей і потреб.
4. Програмне забезпечення можна інтегрувати з іншими системами дистанційного навчання, такими як Moodle або Google Classroom.

Розроблене програмне забезпечення сприяє поглибленому вивченню мови запитів HQL та підвищенню якості дистанційного навчання. Запропонований підхід дозволяє студентам закріпити знання через практичне застосування, що є важливим для їхньої майбутньої професійної діяльності.

Література

1. Hibernate ORM Documentation [Електронний ресурс]. – Режим доступу: <https://hibernate.org/orm/documentation/>
2. MySQL Reference Manual [Електронний ресурс]. – Режим доступу: <https://dev.mysql.com/doc/>
3. HQL Basics Tutorial [Електронний ресурс]. – Режим доступу: <https://www.javatpoint.com/hibernate-query-language-hql>

АЛГОРИТМІЗАЦІЯ ТА РОЗРОБКА ЗАСТОСУНКУ ПЕРСОНАЛІЗОВАНИХ РЕКОМЕНДАЦІЙ ДЛЯ ОПТИМІЗАЦІЇ ФІЗИЧНОЇ АКТИВНОСТІ НА ОСНОВІ МОНІТОРИНГУ КРОКІВ ТА ВИТРАЧЕНИХ КАЛОРІЙ

*А.С. Шеліхов, магістр спеціальності «122 Комп'ютерні науки»,
освітньої програми «Комп'ютерні науки», група КН м-11,
Т.О. Парфьонова, науковий керівник, доцент кафедр, к ф.-м наук,
Полтавський університет економіки і торгівлі*

Ключові слова: fitness tracker, React Native, Expo, моніторинг активності, трекер кроків, персоналізовані рекомендації.

Актуальність теми обумовлена глобальними змінами у способі життя населення. Сидячий спосіб життя, недостатня фізична активність та підвищений рівень стресу призвели до значного зростання захворюваності на хронічні недуги. У відповідь на ці виклики інформаційні технології пропонують нові підходи до підтримки здоров'я та формування активного стилю життя. Одним із таких підходів є створення мобільних застосунків для персоналізованого моніторингу фізичної активності на основі аналізу даних про кроки та витрачені калорії.

Метою роботи є алгоритмізація процесу формування персоналізованих рекомендацій та реалізація мобільного застосунку з кросплатформним інтерфейсом для оптимізації фізичної активності користувачів. Реалізація здійснюється за допомогою React Native у середовищі Expo з використанням бібліотек для моніторингу активності та обчислень енергоспоживання [3; 4].

Застосунок включає наступні модулі:

- реєстрація користувача з урахуванням особистих параметрів (вік, вага, стать);

- трекінг кроків у реальному часі із синхронізацією з вбудованими датчиками пристрою;
- розрахунок витрачених калорій з урахуванням індивідуальних даних;
- аналіз досягнутого рівня активності та формування рекомендацій;
- push-нагадування про необхідність фізичних навантажень;
- створення щотижневих звітів з візуалізацією результатів активності.

Алгоритмічна частина проекту передбачає розрахунок добової норми кроків, визначення енергетичного балансу та оцінювання динаміки змін активності користувача. Наприклад, застосовується формула Harris-Benedict для розрахунку базової потреби у калоріях, яка коригується з урахуванням рівня активності. Далі відстежується фактична витрата енергії й формується рекомендація (наприклад, «зробіть ще 1200 кроків для досягнення цілі дня»).

Важливою особливістю застосунку є персоналізований підхід: кожен користувач отримує рекомендації, які базуються на його унікальних фізичних параметрах та історії активності. Це дозволяє уникнути шаблонних порад і забезпечує високу мотивацію для досягнення поставлених цілей [2; 5].

Під час реалізації було враховано UX/UI аспекти: використано мінімалістичний інтерфейс, зрозумілу навігацію, кольорове кодування результатів активності. Важливе значення також мала безпека персональних даних: передбачено локальне шифрування результатів та обмеження доступу через автентифікацію.

Практичне значення роботи полягає у можливості використання застосунку як окремими користувачами, так і фітнес-центрами або медичними організаціями для надання рекомендацій у сфері активного способу життя. Додаток може стати базою для подальшого розширення — наприклад, підключення Bluetooth-пристроїв, інтеграції з медичними системами або додавання функції дієтичного трекінгу.

Таким чином, реалізовано програмне рішення, що дозволяє підвищити ефективність моніторингу фізичної активності та сприяє формуванню здорових звичок користувачів.

Литература

1. World Health Organization. Guidelines on Physical Activity and Sedentary Behaviour. – Geneva: WHO, 2020. – URL: <https://www.who.int/publications/i/item/9789240015128>
2. Tudor-Locke C., Bassett D.R. How many steps/day are enough? // Sports Medicine. – 2004. – Vol. 34(1). – P. 1–8.
3. Expo Documentation. – URL: <https://docs.expo.dev>
4. React Native Documentation. – URL: <https://reactnative.dev>
5. Zhang J., Rowhani-Rahbar A. A Mobile App for Personalized Physical Activity Planning // Journal of Medical Internet Research. – 2019. – Vol. 21(3).