



**МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ХЕРСОНСЬКИЙ НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ
КАФЕДРА ПРОГРАМНИХ ЗАСОБІВ І ТЕХНОЛОГІЙ**

**МАТЕРІАЛИ
V ВСЕУКРАЇНСЬКОЇ
НАУКОВО-ТЕХНІЧНОЇ КОНФЕРЕНЦІЇ
«МОЛОДЬ І НАУКА»**

24 травня

Хмельницький – 2024

Молодь і наука: Матеріали V всеукраїнської науково-практичної конференції молодих вчених, аспірантів і студентів (24 травня 2024 р., м. Хмельницький) / за ред.

О.О.Боскіна, Р.М.Захарченко. – Херсон, 2024. – 101 с. (електронне видання)

У збірнику представлені матеріали доповідей молодих вчених, студентів та аспірантів з науковими керівниками на конференції «Молодь і наука», що організовувалася та проходила кафедраю програмних засобів і технологій факультету інформаційних технологій та дизайну Херсонського національного технічного університету 24 травня 2024 року. Доповіді присвячені актуальним питанням розвитку теорії і практики інформаційних технологій та програмування, прикладним аспектам застосування програмних рішень у різних прикладних сферах.

V Всеукраїнська науково-технічна конференція «Молодь і наука» відбулася 24 травня 2024 р. у Херсонському національному технічному університеті на базі кафедри програмних засобів і технологій у заочному форматі (електронне видання). 100 стор.

ТЕМАТИЧНІ НАПРЯМКИ КОНФЕРЕНЦІЇ:

- інструментальні засоби і середовища програмування;
- моделі і процеси життєвого циклу програмного забезпечення;
- методи і технології розробки програмного забезпечення;
- інформаційні системи, бази даних та інтелектуальні системи;
- мультимедійні системи, навчальні та ігрові програми;
- інтернет-технології та WEB-дизайн;
- програмні системи захисту інформації

ОРГАНІЗАТОРИ:

Міністерство освіти і науки України
Херсонський національний технічний університет
Херсонська державна морська академія
Івано-Франківський національний технічний університет нафти і газу
Національний технічний університет "Дніпровська політехніка"
Український державний університет залізничного транспорту

ПРОГРАМНИЙ КОМІТЕТ

Шерстюк Володимир Григорович, д.т.н., професор Херсонський національний технічний університет– Голова програмного Комітету
Бень Андрій Павлович, к.т.н., професор, проректор з науково-педагогічної роботи, Херсонська державна морська академія
Шекета Василь Іванович, д.т.н., професор, Івано-Франківський національний технічний університет нафти і газу
Гнатушенко Володимир Володимирович, д.т.н., професор, Національний технічний університет "Дніпровська політехніка"
Каргін Анатолій Олексійович, д.т.н., професор, Український державний університет залізничного транспорту, Україна
Жарікова Марина Віталіївна, д.т.н., професор Херсонський національний технічний університет
Огнева Оксана Євгенівна, к.т.н., доцент Херсонський національний технічний університет

ОРГАНІЗАЦІЙНИЙ КОМІТЕТ

Козуб Наталія Олександрівна, к.т.н., доцент – Секретар оргкомітету
Захарченко Раїса Миколаївна, к.т.н., доцент
Боскін Олег Йосипович, ст. викладач
Комісаров Олександр Сергійович, асистент

Адреса Оргкомітету:

Україна, 29016 м. Хмельницький, вул. Інститутська, 11.e-mail: pzit316@gmail.com,
+380972776600

Зміст

Агєєнко А.Ф., Величко Ю.І., РОЗРОБКА ІНФОРМАЦІЙНОЇ СИСТЕМИ З ОБЛІКУ ІНФОРМАЦІЇ В ОНЛАЙН-МАГАЗИНІ	6
Антошук А.В., Боскін О.О., ВИКОРИСТАННЯ ANYLOGIC ДЛЯ МОДЕЛЮВАННЯ ТА ОЦІНКИ ПРОФЕСІЙНИХ НАВИЧОК У СКЛАДНИХ ДИНАМІЧНИХ СИСТЕМАХ	8
Атаманчук В.Ю., Захарченко Л.М., ГІБРИДНИЙ МЕТОД ПОБУДОВИ РЕКОМЕНДАЦІЙНИХ СИСТЕМ ПРИ ПІДБОРІ КАДРІВ В ІТ-СФЕРІ	12
Атаманчук В.Ю., Захарченко Р.М., МЕТОДИ ФОРМУВАННЯ РЕКОМЕНДАЦІЙ В РЕКОМЕНДАЦІЙНИХ СИСТЕМАХ	15
Атаманчук В.Ю., Захарченко Л.М., РЕАЛІЗАЦІЯ РЕКОМЕНДАЦІЙНОЇ СИСТЕМИ УПРАВЛІННЯ ПІДБОРОМ ПЕРСОНАЛУ ІТ-ПІДПРИЄМСТВ	17
Балабай Д.Ю., Захарченко Р.М., АНАЛІЗ ТА ОПТИМІЗАЦІЯ ПРОДУКТИВНОСТІ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ	19
Блакитний О.А., Захарченко Р.М., ВИКОРИСТАННЯ WEBASSEMBLY ДЛЯ ПОКРАЩЕННЯ ПРОДУКТИВНОСТІ ВЕБ-СТОРИНОК	21
Блакитний О.А., Захарченко Л.М., ЯК СНАТГРТ МОЖЕ ПОКРАЩИТИ ВЕБ-РОЗРОБКУ	23
Бойко М.Є., Огнєва О.Є., ПРОЄКТУВАННЯ ТА РОЗРОБКА РЕЛЯЦІЙНОЇ БАЗИ ДАНИХ ДЛЯ WEB ДОДАТКУ «ІНТЕРНЕТ МАГАЗИН ЕЛЕКТРОНІКИ»	24
Валюга В.В., Комісаров О.С., ІНТЕГРАЦІЯ ШТУЧНОГО ІНТЕЛЕКТУ У ВЕБ-ДОДАТКИ	26
Горностаєв Д.С., Огнєва О.Є., ДОСЛІДЖЕННЯ РОЗВИТКУ ІІІ, ЙОГО СУЧАСНОГО СТАНУ ТА ВПЛИВУ НА РІЗНІ СФЕРИ ЖИТТЯ	28
Довгаль О.О., Огнєва О.Є., СТВОРЕННЯ СЕРВІСУ ДЛЯ ПЕРЕВІРКИ БРОНЮВАНЬ НА ОСНОВІ КЛІЄНТ-СЕРВЕРНОЇ АРХІТЕКТУРИ	30
Дудник Б.М., Захарченко Р.М., ВИКОРИСТАННЯ ІТ В РОЗРОБЦІ ВЕБ-ДОДАТКУ ДЛЯ МАГАЗИНУ КНИЖОК	33
Зярко Д.В., Захарченко Р.М., ВИКОРИСТАННЯ ІТ В РОЗРОБЦІ ВЕБ-ДОДАТКУ ДЛЯ ТУРИСТИЧНОГО АГЕНСТВА	36
Іваниця М.С., Хохлов В.А., БАЗОВІ ТА СУЧАСНІ ТЕОРІЇ МОТИВАЦІЇ ТА МОТИВАЦІЙНОГО СПОНУКАННЯ В ІТ – СФЕРІ	40
Іваниця М.С., Хохлов В.А., РОЗРОБКА ТА ОПТИМІЗАЦІЯ СИСТЕМИ РЕКОМЕНДАЦІЙ НА ОСНОВІ МАШИННОГО НАВЧАННЯ	42
Комісаров О.С., Захарченко Р.М., ОПТИМІЗАЦІЯ АЛГОРИТМІВ ПОШУКУ ТА РОЗВІДКИ ДЛЯ РОЮ АВТОНОМНИХ АГЕНТІВ НА ОСНОВІ МЕТОДІВ МАШИННОГО НАВЧАННЯ	44
Леба Д. С., Огнєва О.Є., РОЗРОБКА БАЗИ ДАНИХ ВЕБ-ДОДАТКУ "НАЗЕМНА БРОНЕТЕХНІКА"	47

Любимов О.С., ВДОСКОНАЛЕННЯ МОДЕЛІ РОЗПІЗНАВАННЯ ОСЕРЕДКІВ ЛІСОВИХ ПОЖЕЖ ПРИ ДИСТАНЦІЙНОМУ ЗОНДУВАННЮ З БЕЗПЛОТНИХ АПАРАТІВ	49
Лященко М.О., Захарченко Р.М., ВИКОРИСТАННЯ ІТ ПРИ РОЗРОБЦІ ВЕБ-ДОДАТКУ ДЛЯ НАВЧАЛЬНОГО ВІДДІЛУ ЗВО	53
Матвійчук О.В., Огнєва О.Є., РОЗРОБКА ПРОТОТИПУ ШАБЛОНУ ДЛЯ ОПТИМІЗАЦІЇ ІНТЕРНЕТ-МАГАЗИНІВ ПРИ НИЗЬКІЙ ШВИДКОСТІ ІНТЕРНЕТУ	55
Мацків В.О., Огнєва О.Є., ВПЛИВ РІЗНИХ СТРУКТУР ДАНИХ НА ПРОДУКТИВНІСТЬ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ	58
Міщенко Д.М., Фролова М.Е., ВИКОРИСТАННЯ ІТ ДЛЯ РОЗРОБКИ ДОДАТКУ ЧИСЕЛЬНОГО РОЗВ'ЯЗКУ СЛАР	61
Морозов О.Є., Хохлов В.А., РОЗРОБКА ІНФОРМАЦІЙНОЇ СИСТЕМИ З ОБЛІКУ КАРТОТЕК ДЛЯ ПОЛІКЛІНІКИ	63
Нікулінський Д.В., Огнєва О.Є., ДОСЛІДЖЕННЯ МОЖЛИВОСТЕЙ ВИКОРИСТАННЯ ШТУЧНОГО ІНТЕЛЕКТУ ДЛЯ АНАЛІЗУ КОМПЛЕКТУЮЧИХ ПК	65
Орлик О.О., Боскін О.О., РОЗРОБКА СИСТЕМИ ПЛАНУВАННЯ ЗАВДАНЬ	67
Петров В.О., Огнєва О.Є., ВИВЧЕННЯ НОВИХ ТЕНДЕНЦІЙ ТА ТЕХНОЛОГІЙ У ГАЛУЗІ РОЗРОБКИ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ	69
Платошечкін А.М., Козуб Н.О., АРКАДНІ ІГРИ, ЯК ЗАСІБ ДЛЯ КОРОТКОГО ВІДПОЧИНКУ	71
Плутенко Д.В., Огнєва О.Є., ЗАНУРЕННЯ У СВІТ ЖАХУ: ДОСЛІДЖЕННЯ ТА РОЗРОБКА ХОРОР ІГОР	73
Ринденко Є.І., Огнєва О.Є., ПРОЄКТУВАННЯ БАЗИ ДАНИХ ВЕБ-ДОДАТКУ ДЛЯ ПУБЛІКАЦІЇ АВТОРАМИ-ПОЧАТКІВЦЯМИ ТВОРІВ	75
Рогатовська А.А., Огнєва О.Є., ОПТИМІЗАЦІЯ РАЦІОНУ ХАРЧУВАННЯ ДІАБЕТИЧНИХ ХВОРИХ	78
Середюк В.Я., Фролова М.Е., ДОСЛІДЖЕННЯ ВИКОРИСТАННЯ НОРМАЛІЗОВАНИХ ДАНИХ ДЛЯ АНАЛІЗУ ТЕКСТУ	81
Тищенко М.О., Огнєва О.Є., ВИВЧЕННЯ ТА ЗАСТОСУВАННЯ НОВИХ ПАРАДИГМ І ПІДХОДІВ ДО РОЗРОБКИ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ (НАПРИКЛАД DEVOPS, БЕЗСЕРВЕРНА АРХІТЕКТУРА, МІКРОСЕРВІСИ)	83
Фідіна К.В., Огнєва О.Є., РОЗРОБКА ВЕБ-ОРІЄНТОВАНОЇ ІНФОРМАЦІЙНОЇ СИСТЕМИ ДЛЯ ОБСЛУГОВУВАННЯ АБОНЕНТІВ ПРОВАЙДЕРА МЕРЕЖІ ІНТЕРНЕТ	85
Філіпчук В.Р., Огнєва О.Є., ДОСЛІДЖЕННЯ ТА РОЗРОБКА ШАБЛОНІВ ПРОЄКТУВАННЯ ДЛЯ СТВОРЕННЯ ВЕБ-ДОДАТКУ "АПТЕКА"	87
Шатровський В.О., Фролова М.Е., ВИКОРИСТАННЯ НЕЙРОННИХ МЕРЕЖ ДЛЯ СТВОРЕННЯ ТОРГОВИХ АЛГОРИТМІВ НА КРИПТОВАЛЮТНИХ РИНКАХ	89

Шишов М.О., Огнєва О.Є., к.т.н., ДОСЛІДЖЕННЯ ОПТИМІЗАЦІЇ ПРОДУКТИВНОСТІ ТА ШВИДКОДІЇ ВЕБ-САЙТІВ ЗА ДОПОМОГОЮ ФРЕЙМВОРКУ ЛАРАВЕЛ	92
Юдін Д.Ф., Захарченко Р.М., ВИКОРИСТАННЯ ІТ В РОЗРОБЦІ ФУЛСТЕК ВЕБ-ДОДАТКУ ДЛЯ ПРОДАЖУ ЕЛЕКТРОНІКИ НА ФРЕЙМВОРКУ REACT ТА NODEJS	94
Яркова Є.Д., Величко Ю.І., РОЗРОБКА ВЕБ-САЙТУ «ВЕБ-ЛІКАРНЯ»	97

Агєєнко А.Ф., студент кафедри програмних засобів і технологій, Херсонський національний технічний університет, м.Хмельницький, Україна,

Величко Ю.І., к.т.н., доцент кафедри програмних засобів і технологій, Херсонський національний технічний університет, м.Хмельницький, Україна.

РОЗРОБКА ІНФОРМАЦІЙНОЇ СИСТЕМИ З ОБЛІКУ ІНФОРМАЦІЇ В ОНЛАЙН-МАГАЗИНІ

Інформаційні потоки представляють собою сукупність даних, які циркулюють в організації або між організаціями для забезпечення виконання різних функцій управління. Ці потоки можуть включати в себе різноманітну інформацію, починаючи від фінансових звітів і закінчуючи повідомленнями електронної пошти. Інформаційні потоки є життєво важливими для ефективної роботи підприємств, оскільки вони забезпечують необхідні дані для прийняття рішень, координації діяльності та контролю виконання завдань.

Системи інформаційних потоків — це структури та процеси, які забезпечують збір, обробку, зберігання, передачу та використання інформації в організації. Вони включають в себе як технічні засоби (комп'ютерні мережі, бази даних, програмне забезпечення), так і організаційні процедури та регламенти. Ефективна система інформаційних потоків дозволяє своєчасно отримувати точні дані, що сприяє оперативному управлінню та прийняттю обґрунтованих рішень.

Найважливішою особливістю процесу управління є його циклічність та безперервність. Управління складається з постійного аналізу ситуації, прийняття рішень, реалізації цих рішень і подальшого контролю результатів. Весь цей процес базується на інформації, що надходить з різних джерел. Безперервний потік інформації дозволяє менеджерам адаптувати свої дії до змінних умов і забезпечувати досягнення цілей організації.

Інформацію умовно вважають предметом та продуктом управлінської праці, тому що вона є основним ресурсом, який використовується менеджерами для виконання своїх функцій. Як предмет праці, інформація потребує збору, аналізу, обробки та зберігання. Як продукт праці, вона формується в процесі управлінської діяльності у вигляді звітів, планів, прогнозів і інших документів. Таким чином, інформація є як сировиною, так і результатом управлінської діяльності.

Інформація як елемент управління і предмет управлінської праці повинна забезпечувати кілька ключових аспектів. По-перше, вона має бути достовірною, щоб на її основі можна було приймати правильні рішення. По-друге, інформація повинна бути своєчасною, щоб менеджери могли оперативно реагувати на зміни в зовнішньому та внутрішньому середовищі. По-третє, вона повинна бути релевантною, тобто відповідати потребам конкретного управлінського завдання. Нарешті, інформація повинна бути зрозумілою та доступною для тих, хто її використовує.

Інформаційне забезпечення відіграє критично важливу роль в управлінській діяльності. Воно включає в себе організацію інформаційних потоків і систем, які дозволяють ефективно збирати, обробляти і використовувати інформацію. Надійна система інформаційного забезпечення сприяє прийняттю обґрунтованих управлінських рішень, забезпечує контроль і координацію діяльності, а також дозволяє швидко адаптуватися до змінних умов. Тому інвестиції в розробку та підтримку ефективних інформаційних систем є важливим завданням для будь-якої організації.

Облік інформації в онлайн-магазині – виконання завдань з підвищення ефективності роботи онлайн-магазину завдяки автоматизації ведення обліку інформації про товари, користувачів, замовлення, тощо.

Об'єктом дослідження є велика кількість внутрішніх елементів магазину, які взаємно пов'язані. В ході дослідження проводиться аналіз предметної області, визначення моделей, методів та алгоритмів, необхідних для постановки задачі та розробки математичної моделі. Зокрема, увага приділяється вимогам до програмного забезпечення та розробці програмного продукту, що включає в себе базу даних та алгоритми їх обробки та візуалізації.

SQLite Administrator — це інтегроване середовище для управління будь-якою інфраструктурою баз даних SQLite. Використовується для доступу, налаштування, адміністрування та розробки всіх компонентів SQLite баз даних, а також управління ними. Середовище SQLite Administrator надає єдину повнофункціональну службову програму, яка поєднує в собі велику групу графічних інструментів з низкою редакторів сценаріїв для доступу до SQLite для розробників і адміністраторів баз даних всіх професійних рівнів [1-5].

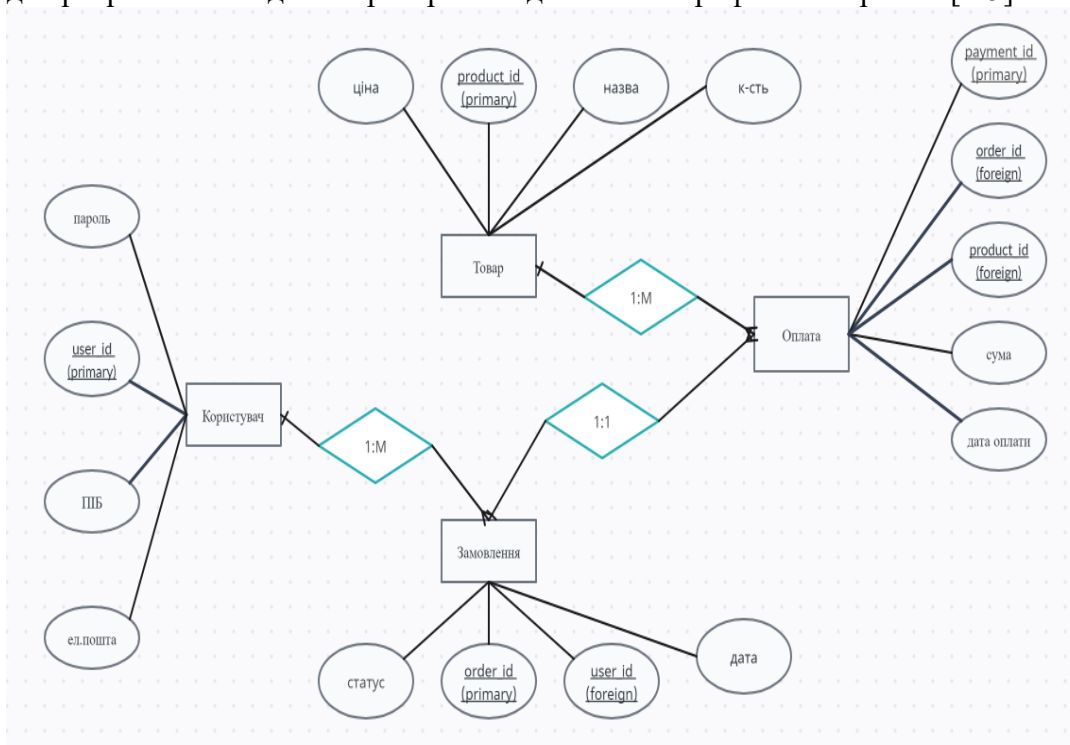


Рисунок 1 - ER-діаграма зпроектованої бази даних

Система інформаційного забезпечення являє собою сукупність даних про цілі, стан, напрямки розвитку об'єкта та його оточення, організованих у взаємопов'язаних інформаційних потоках. Ця система включає методи збору, зберігання, пошуку, обробки та надання даних користувачеві.

Команди та функції, які пропонуються розробниками програмних продуктів у середовищі SQLite Administrator, задовольняють більшість сучасних вимог до представлення та обробки даних, а також створення баз даних.

ПЕРЕЛІК ПОСИЛАНЬ:

1. Introduction to SQL. URL: <https://www.w3schools.com/sql/>
2. Learn SQL. URL: <https://www.codecademy.com/learn/learn-sql>
3. SQL-wasm.js GitHub Repository. URL: <https://github.com/sql-js/sql.js>
4. The SQL Tutorial for Data Analysis. URL: <https://mode.com/sql-tutorial/>
5. TutorialsPoint SQLite Tutorial. URL: <https://www.javatpoint.com/sqlite-tutorial>

Антощук А.В., студент кафедри програмних засобів і технологій, Херсонський національний технічний університет, м. Хмельницький, Україна.

Боскін О.О., ст.викладач кафедри програмних засобів і технологій, Херсонський національний технічний університет, м.Хмельницький, Україна.

ВИКОРИСТАННЯ ANYLOGIC ДЛЯ МОДЕЛЮВАННЯ ТА ОЦІНКИ ПРОФЕСІЙНИХ НАВИЧОК У СКЛАДНИХ ДИНАМІЧНИХ СИСТЕМАХ

Складні динамічні системи, такі як логістичні мережі, виробничі процеси чи системи управління, є невід’ємною частиною сучасних професійних сфер. Їх моделювання дозволяє аналізувати поведінку, прогнозувати результати та оптимізувати процеси в умовах невизначеності. У контексті підготовки фахівців моделювання набуває особливого значення, адже воно дає змогу оцінити професійні навички в безпечному віртуальному середовищі, де можна відтворити реальні робочі ситуації.

Оцінка професійних навичок є критично важливою для забезпечення якості підготовки фахівців, оскільки дозволяє виявити сильні та слабкі сторони, вдосконалити прийняття рішень і адаптивність до змін. Програмне забезпечення AnyLogic, завдяки своїм можливостям системного, агентного та дискретно-подійного моделювання, відкриває широкі перспективи для створення симуляційних моделей, що імітують складні робочі процеси.

Мета – дослідити потенціал AnyLogic для створення симуляційних моделей, які сприяють оцінці та вдосконаленню професійних навичок у складних динамічних системах. Особливу увагу приділено адаптивним сценаріям, які дозволяють моделювати нестандартні ситуації та аналізувати ефективність дій фахівців.

AnyLogic – це потужне програмне забезпечення для симуляційного моделювання, яке підтримує три основні підходи: системно-динамічне, агентне та дискретно-подійне моделювання. Системно-динамічне моделювання дозволяє аналізувати поведінку складних систем через взаємодію змінних у часі. Агентне моделювання фокусується на індивідуальних об’єктах (агентах), їхній поведінці та взаємодії, що ідеально підходить для імітації людських рішень у професійних контекстах. Дискретно-подійне моделювання відтворює послідовність подій, що є ефективним для аналізу процесів, таких як виробничі чи логістичні ланцюги. Унікальність AnyLogic полягає в можливості комбінувати ці підходи в одній моделі, що забезпечує комплексний аналіз складних систем.

Порівняно з іншими інструментами, такими як MATLAB чи Simulink, AnyLogic має низку переваг. MATLAB ефективний для числових обчислень і математичного моделювання, але менш зручний для створення інтерактивних симуляцій із візуалізацією. Simulink орієнтований на моделювання систем керування, але обмежений у гнучкості для агентного чи гібридного моделювання. AnyLogic вирізняється простотою створення інтерактивних моделей, підтримкою Java для кастомізації та можливістю інтеграції з іншими платформами.

Ключові можливості AnyLogic включають:

- Гнучкість: адаптація до різних типів систем і сценаріїв завдяки комбінації методів моделювання.
- Інтеграція даних: імпорт даних із зовнішніх джерел (наприклад, Excel, бази даних) для створення реалістичних моделей.
- Візуалізація: створення 2D- і 3D-моделей, що полегшують інтерпретацію результатів і демонстрацію симуляцій.

Ці можливості роблять AnyLogic ефективним інструментом для моделювання професійних навичок у складних динамічних системах, забезпечуючи точність і практичну цінність симуляцій.

Складні динамічні системи – це системи, що характеризуються великою кількістю взаємопов’язаних елементів, нелінійною поведінкою, динамічними змінами в часі та впливом зовнішніх факторів. Їх особливості включають непередбачуваність, чутливість до початкових

умов, взаємодію між компонентами та здатність до самоорганізації. Такі системи часто зустрічаються в реальному світі, де рішення приймаються в умовах невизначеності та обмеженого часу.

У професійній діяльності прикладами складних динамічних систем є:

- Логістика: управління ланцюгами постачання, де необхідно оптимізувати маршрути, враховувати затримки, попит і ресурси.
- Управління проектами: координація команд, розподіл ресурсів і реагування на зміни в термінах чи бюджеті.
- Виробництво: автоматизовані лінії, де збої в одному процесі можуть вплинути на весь цикл, а також оптимізація використання обладнання.

Симуляційне моделювання відіграє ключову роль в аналізі поведінки таких систем. Воно дозволяє:

- відтворювати реальні сценарії для тестування стратегій і рішень;
- прогнозувати наслідки змін у системі (наприклад, зміна попиту чи поломка обладнання);
- оцінювати ефективність дій фахівців у контрольованому середовищі.

Використання AnyLogic для моделювання таких систем забезпечує створення деталізованих імітаційних моделей, які допомагають виявити слабкі місця, протестувати різні сценарії та вдосконалити професійні навички фахівців у реальних умовах.

Симуляційні моделі в AnyLogic дозволяють створювати адаптивні сценарії, які відтворюють реальні робочі ситуації для оцінки професійних навичок. Завдяки гнучкості платформи можна моделювати різноманітні умови, включаючи типові задачі, стресові ситуації та нестандартні виклики. Адаптивні сценарії в AnyLogic формуються шляхом налаштування параметрів, таких як зміна вхідних даних (наприклад, попит у логістиці), введення випадкових подій (збої обладнання) або динамічних обмежень (скорочення часу на виконання завдання). Це дозволяє імітувати реальні професійні середовища, де фахівці змушені приймати рішення в умовах невизначеності.

Моделі в AnyLogic використовуються для оцінки ключових аспектів професійної діяльності:

- Прийняття рішень: аналіз якості рішень, наприклад, вибір оптимального маршруту в логістиці чи розподіл ресурсів у проєкті.
- Швидкість реакції: вимірювання часу, необхідного для реагування на зміни чи вирішення проблем.
- Ефективність дій: оцінка результатів роботи фахівця, таких як досягнення цілей або мінімізація витрат.
- Для оцінки навичок використовуються конкретні метрики, зокрема:
- Час виконання завдань: тривалість, потрібна для завершення задачі, наприклад, налаштування виробничої лінії чи планування логістичного маршруту.
- Кількість помилок: частота неправильних рішень, таких як вибір неоптимального шляху чи ігнорування критичних змін.
- Адаптивність до змін умов: здатність фахівця швидко реагувати на нові обставини, наприклад, зміну пріоритетів у проєкті чи появу несподіваних перешкод.

Такі моделі дозволяють не лише оцінити поточний рівень підготовки, але й виявити прогалини в навичках, що потребують вдосконалення, сприяючи цілеспрямованому розвитку фахівців.

Інтеграція адаптивних сценаріїв у AnyLogic дозволяє створювати гнучкі моделі, які імітують реальні професійні виклики. Побудова таких сценаріїв передбачає використання динамічних параметрів, що змінюються залежно від умов моделі. У AnyLogic це досягається через налаштування змінних, таких як обсяг ресурсів, часові обмеження чи ймовірність виникнення подій. Наприклад, у логістичній моделі можна задати змінний попит або затримки доставки, а у виробничій – випадкові поломки обладнання. Використання Java-коду та

бібліотек AnyLogic забезпечує можливість створювати складні сценарії з динамічною взаємодією агентів і подій.

Моделювання стресових ситуацій та нестандартних умов є ключовим для перевірки професійних навичок. У AnyLogic можна імітувати критичні сценарії, такі як:

- Раптове зростання попиту в логістиці, що вимагає швидкого перепланування маршрутів.
- Зміна пріоритетів у проєктному управлінні через обмеження бюджету чи термінів.
- Аварійні ситуації на виробництві, що потребують негайного реагування.

Такі сценарії дозволяють оцінити здатність фахівця адаптуватися до непередбачуваних обставин, приймати обґрунтовані рішення та ефективно управляти ресурсами. Наприклад, модель може включати таймер, який фіксує швидкість реакції, або лічильник помилок для оцінки точності дій.

Аналіз результатів симуляцій у AnyLogic забезпечує цінну інформацію для вдосконалення професійної підготовки. Платформа дозволяє генерувати звіти з ключовими показниками, такими як час виконання завдання, кількість помилок чи ефективність використання ресурсів. На основі цих даних можна виявити слабкі сторони фахівців, розробити індивідуальні програми навчання або скоригувати робочі процеси. Наприклад, якщо симуляція показує, що фахівець повільно реагує на зміни умов, можна запропонувати тренінги з управління стресом або прийняття рішень. Таким чином, адаптивні сценарії в AnyLogic сприяють цілеспрямованому розвитку професійних навичок.

Кейс: моделювання роботи менеджера логістики в AnyLogic.

Розглянемо приклад моделювання роботи менеджера логістики, який відповідає за оптимізацію маршрутів доставки в умовах змінного попиту та обмежених ресурсів. У симуляції відтворюється логістична мережа з кількома складами, транспортними засобами та клієнтами. Мета – оцінити навички менеджера в прийнятті рішень щодо вибору маршрутів, розподілу ресурсів та реагування на непередбачувані події, такі як затримки чи зміна замовлень.

Налаштування моделі.

Вхідні дані:

- Кількість складів і клієнтів (наприклад, 3 склади, 20 клієнтів).
- Відстані між об'єктами, витрати на транспортування.
- Обсяг попиту клієнтів (змінюється випадковим чином у заданих межах).
- Доступність транспортних засобів (кількість, місткість).

Змінні:

- Час доставки (залежить від обраного маршруту).
- Рівень пріоритетності замовлень (високий/низький).
- Випадкові події: затримки через погодні умови чи технічні збої.

Вихідні показники:

- Загальний час виконання всіх доставок.
- Витрати на транспортування (паливо, амортизація).
- Кількість помилок (наприклад, доставка не в пріоритетному порядку).
- Рівень задоволення клієнтів (залежить від швидкості доставки).

Модель створена з використанням гібридного підходу в AnyLogic: агентне моделювання для імітації поведінки транспортних засобів і менеджера, дискретно-подійне – для обробки послідовності доставок. Сценарій включає адаптивні умови, наприклад, раптове зростання попиту або вихід з ладу одного транспортного засобу.

Результати симуляції та їх інтерпретація.

Після запуску симуляції AnyLogic генерує звіт із ключовими показниками. Наприклад:

- Менеджер завершив доставку за 8 годин, але перевищив бюджет на 15% через неефективний вибір маршрутів.

- Кількість помилок: 3 (два замовлення доставлено з порушенням пріоритетів, один – із затримкою).
 - Рівень задоволення клієнтів: 85% (через затримки кількох доставок).
- Ці результати дозволяють оцінити навички менеджера.
- Прийняття рішень: неефективний вибір маршрутів вказує на потребу в додатковому навчанні з оптимізації.
 - Швидкість реакції: затримки свідчать про повільне реагування на зміни (наприклад, некоректне перенаправлення транспорту).
 - Адаптивність: здатність менеджера швидко перерозподілити ресурси при збої транспортного засобу.

На основі аналізу можна запропонувати тренінги з логістичного планування або вдосконалення алгоритмів прийняття рішень. Такий підхід демонструє, як AnyLogic допомагає не лише оцінити професійні навички, але й визначити конкретні напрями їх розвитку.

Використання AnyLogic для оцінки професійних навичок у складних динамічних системах має низку переваг. По-перше, платформа дозволяє створювати реалістичні симуляційні моделі, що відтворюють реальні робочі сценарії з високим ступенем деталізації. По-друге, гнучкість AnyLogic у поєднанні системного, агентного та дискретно-подійного моделювання забезпечує комплексний підхід до аналізу навичок, таких як прийняття рішень, швидкість реакції та адаптивність. По-третє, візуалізація та інтеграція даних спрощують інтерпретацію результатів, що робить оцінку доступною для тренерів і фахівців. Це сприяє цілеспрямованому розвитку професійної компетентності.

Проте моделювання в AnyLogic має обмеження та виклики. Створення складних моделей вимагає значних затрат часу на налаштування та тестування. Висока деталізація моделей може ускладнювати їх масштабування для великих систем. Крім того, точність симуляцій залежить від якості вхідних даних, а помилки в даних можуть спотворити результати. Також потрібні спеціалізовані знання для розробки моделей, що може бути викликом для користувачів-початківців.

Перспективи подальшого розвитку включають інтеграцію AnyLogic з технологіями віртуальної (VR) та доповненої реальності (AR), що дозволить створювати імерсивні тренувальні середовища для ще більш реалістичного відпрацювання навичок. Поєднання з штучним інтелектом може автоматизувати аналіз результатів симуляцій, пропонувати персоналізовані рекомендації для навчання та оптимізувати адаптивні сценарії. Ці інновації відкривають нові можливості для підвищення ефективності оцінки та вдосконалення професійних навичок у складних системах.

ПЕРЕЛІК ПОСИЛАНЬ

1. AnyLogic. Офіційна документація AnyLogic [Електронний ресурс] / AnyLogic Company. – Режим доступу: <https://www.anylogic.com/resources/documentation/>. – Дата доступу: 15.07.2025.
2. Борщ В. І. Симуляційне моделювання в AnyLogic: основи та приклади використання / В. І. Борщ, О. П. Сидоренко // Вісник НТУ "ХПІ". – 2023. – № 2. – С. 45–52. – Режим доступу: <http://visnyk.ntu-kpi.kharkov.ua/2023/2/45-52.pdf>. – Дата доступу: 15.07.2025.
3. Law A. M. Simulation Modeling and Analysis / A. M. Law. – 5th ed. – New York: McGraw-Hill Education, 2015. – 804 p. – [Електронний ресурс]. – Режим доступу: <https://www.mhprofessional.com/simulation-modeling-and-analysis-9780073401323-usa>. – Дата доступу: 15.07.2025.
4. Коваленко О. М. Моделювання логістичних систем за допомогою AnyLogic / О. М. Коваленко // Наукові записки УНПК. – 2024. – № 1. – С. 112–120. – Режим доступу: <http://naukovi-zapysky.unpk.edu.ua/2024/1/112-120.pdf>. – Дата доступу: 15.07.2025.
5. Borshchev A. The Big Book of Simulation Modeling: Multimethod Modeling with AnyLogic 8 / A. Borshchev. – AnyLogic North America, 2013. – 614 p. – [Електронний ресурс]. – Режим доступу: <https://www.anylogic.com/resources/books/big-book-of-simulation-modeling/>. – Дата доступу: 15.07.2025.
6. Григор'єв О. В. Використання адаптивних сценаріїв у симуляційному моделюванні для оцінки професійних компетенцій / О. В. Григор'єв, І. С. Петренко // Журнал сучасних інформаційних технологій. – 2024. – № 3. – С. 78–85. – Режим доступу: <http://jsit-journal.org/2024/3/78-85.pdf>. – Дата доступу: 15.07.2025.

Атаманчук В.Ю., студент кафедри програмних засобів і технологій, Херсонський національний технічний університет, м.Хмельницький, Україна,

Захарченко Л.М., викладач-стажист кафедри програмних засобів і технологій, Херсонський національний технічний університет, м.Хмельницький, Україна.

ГІБРИДНИЙ МЕТОД ПОБУДОВИ РЕКОМЕНДАЦІЙНИХ СИСТЕМ ПРИ ПІДБОРІ КАДРІВ В ІТ-СФЕРІ

Гібридний метод побудови рекомендаційної системи для підбору кадрів в ІТ – сфері поєднує використання як колабораційної фільтрації виконаної на базі методу МАІ так і фільтрації вмісту (отриманої інформації за допомогою тестування, співбесіди та інтерв'ювання) [1, 2].

У центрі будь-якої рекомендаційної системи знаходиться так звана матриця переваг. Це матриця, в якій по одній з осей відкладено всі клієнти сервісу (Users), а по іншій - об'єкти рекомендації (Items). На перетині певних пар (user, item) дана матриця заповнена оцінками (Ratings) - це показник зацікавленості користувача в даному об'єкті, виражений за заданою шкалою (наприклад від 1 до 9).

Нехай існує набір m об'єктів $A_1, A_2, \dots, A_m$ і задача полягає у визначенні вагових коефіцієнтів ефективності кожного з цих об'єктів, тобто в знаходженні відповідних їм чисел $w_1, w_2, \dots, w_m$. Згідно з МАІ, матриця попарних порівнянь повинна володіти наступними властивостями [1, 2]:

- 1) Всі елементи матриці A додатні: $a_{ij} > 0$ для всіх номерів $i, j = 1, 2, \dots, m$.
- 2) Матриця A обернено симетрична $a_{ji} / a_{ij} = 1$ для всіх номерів $i, j = 1, 2, \dots, m$. Зокрема $a_{ii} = 1, i = 1, 2, \dots, m$.
- 3) Матриця A сумісна, тобто рівність $a_{ij} = a_{ik} / a_{kj}$ має місце для всіх номерів $i, j, k = 1, 2, \dots, m$.
- 4) Число m є максимальним власним значенням матриці A і для деякого єдиного (нормованого) вектор-стовпця $w = (w_1, w_2, \dots, w_m)^T$ з додатніми компонентами виконується рівність $Aw = mw$.

Розглянуто метод побудови матриці попарних порівнянь, яка задовольняє перші три властивості, що перелічені вище. Відповідно до перших двох властивостей діагональні елементи матриці – це одиниці. Далі експерту пропонується порівняти вагу першого об'єкта з вагою другого об'єкта і вказати додатне число, що відображає у скільки разів вага першого об'єкта більша за вагу другого об'єкта [1, 2]. У результаті виконання такого порівняння експерт назначає деяке додатне число a_{12} . Далі для порівняння з першим об'єктом розглядається третій об'єкт і в результаті порівняння експертом вказується число a_{13} і т.д.

Після виконання порівнянь першого об'єкта з всіма іншими будуть назначені додатні числа a_{12}, a_{13}, a_{1m} . Таким чином буде побудовано перший рядок матриці A . Описаний спосіб визначення елементів першого рядка матриці A можна назвати «схемою порівняння за взірцем», у ролі якого виступає перший об'єкт. Всі інші елементи матриці попарних порівнянь можна розрахувати на основі властивостей 2 і 3 матриці попарних порівнянь. Завдяки цим властивостям виконуються рівності:

$$a_{ij} = a_{i1} a_{1j} = a_{1j} / a_{1i}, i = 2, \dots, m; j = 1, 2, \dots, m \quad (1)$$

з допомогою яких однозначно розраховуються елементи інших рядків матриці A [1, 2].

Після того, як матриця $A = (a_{ij})_{m \times m}$ за допомогою залежності (1) сформована, можна знайти вектор вагових коефіцієнтів важливості $w = (w_1, w_2, \dots, w_m)^T$. Його компоненти визначають залежністю:

$$w_i = a_{1m} / a_{1i}, i = 1, 2, \dots, m \quad (2)$$

Вектор вагових коефіцієнтів важливості, визначений за допомогою (2), не задовольняє вимогу нормалізації, так як його остання компонента рівна одиниці.

Якщо необхідно, щоб він був нормований, кожен його компоненту слід розділити на суму всіх компонент, тобто на величину

$$w_1 + w_2 + \dots + w_{m-1} + 1,$$

де всі доданки w_i , $i = 1, 2, \dots, m-1$, знайдені за залежністю (2).

Здійснюється експертне порівняння першого критерію зі всіма іншими (для умови задачі).

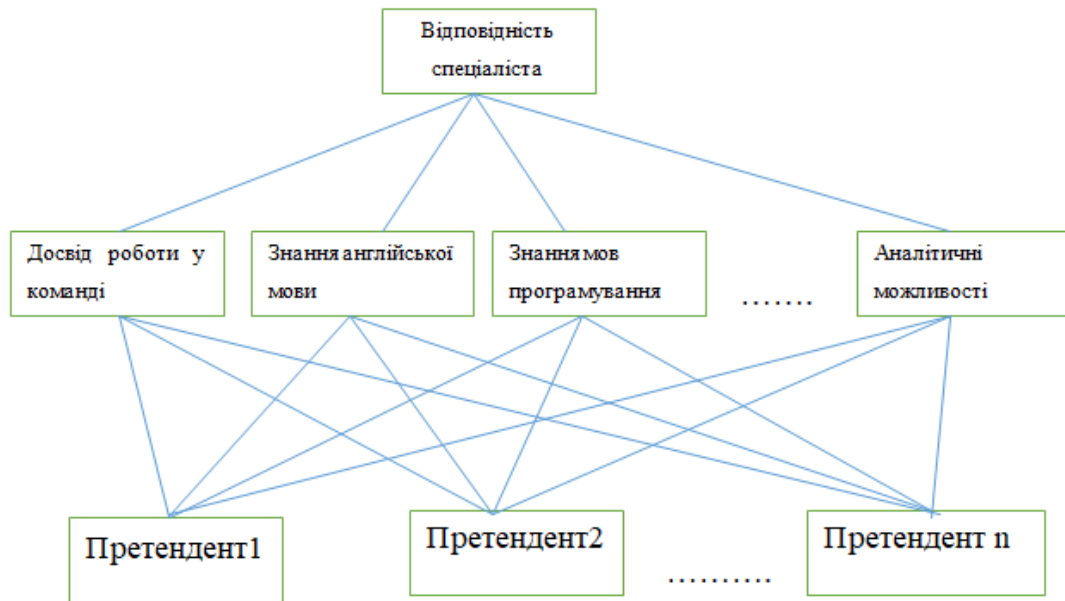


Рисунок 1 - Схема вибору претендента на посаду спеціаліста в ІТ-компанію за обраними критеріями

На рис.1 приведена схема вибору претендента на посаду спеціаліста в ІТ-компанію за обраними критеріями. Був розроблений модуль для рекомендаційної системи підбору кадрів в ІТ-сферу на базі методу МАІ (метод аналізу ієрархій) для визначення відповідності спеціаліста, де з даними працює менеджер [1, 2].

За даним методом було обрано кількість (в додатку їх кількість необмежена) параметрів і їх значення для визначити вагових коефіцієнтів.

Наприклад, були визначені 4 критерії:

K1 - досвід роботи у команді ;

K2 - знання англійської мови;

K3 - знання мов програмування;

K4 - аналітичні можливості.

А претенденти на посаду ІТ-спеціаліста (кількість необхідно обрати) для визначення вагових коефіцієнтів у кількості, наприклад, 4: P1; P2; P3; P4. (в додатку їх кількість необмежена).

В залежності від необхідності попарно за методом МАІ потрібно порівняти критерії у балах від 1 до 9 та обрахувати вагові коефіцієнти значимості за формулою Сааті.

Після внесення всієї інформації, можна отримати результат, натиснувши кнопку Розрахунок. На рис. 2 представлено вікно програми де після внесення даних та натиснення кнопки Розрахунок на екрані вагові коефіцієнти по кожному заданому критерію та загальний.

Результат					
	K1	K2	K3	K4	Коефіцієнт
P1	0,29	0,0892	0,041	0,0787	0,498
P2	0,176	0,00995	0,0282	0,0563	0,27
P3	0,061	0,0584	0,0131	0,0239	0,156
P4	0,0338	0,0338	0,00000	0,0115	0,035

Рисунок 2 - Вікно програми

За методом визначення вагових коефіцієнтів МАІ в програмі автоматично було визначено вагові коефіцієнти після попарного порівняння чотирьох претендентів на посаду ІТ-спеціаліста за чотирма критеріями.

У колаборативній фільтрації на основі користувачів додаток ідентифікує претендентів, які мають подібні значення критеріїв та визначають на базі МАІ вагові коефіцієнти претендентів за кожним критерієм.

Для визначення вагових коефіцієнтів для підбору кадрів ІТ-фахівців за обраними критеріями розроблено додаток.

ПЕРЕЛІК ПОСИЛАНЬ:

1. Saaty, T. L. (2013). On the measurement of intengibles. A principal eigenvector approach to relative measurement derived from paired comparisons. Notices of the American Mathematical Society, 60(2), 192–208. URL: <https://dialnet.unirioja.es/ejemplar/323067>.
2. Проблеми сучасних рекомендаційних систем ресурсу. URL: <https://www.altexsoft.com/blog/engineering/pros-and-cons-of-java-programming/>

Атаманчук В.Ю., студент кафедри програмних засобів і технологій, Херсонський національний технічний університет, м.Хмельницький, Україна,

Захарченко Р.М., к.т.н., доцент кафедри програмних засобів і технологій, Херсонський національний технічний університет, м.Хмельницький, Україна.

МЕТОДИ ФОРМУВАННЯ РЕКОМЕНДАЦІЙ В РЕКОМЕНДАЦІЙНИХ СИСТЕМАХ

Рекомендаційні системи - це про те, що запропонувати клієнту, щоб зробити його щасливим [1, 2].

Рекомендаційні системи порівнюють зібрані дані від користувачів і створюють список елементів, які рекомендовані користувачеві. Вони є альтернативою алгоритму пошуку, оскільки допомагають користувачам швидко знаходити предмети та інформацію, які б ті не змогли знайти самостійно [1, 2]. Особливо широко рекомендаційні системи використовуються в електронній комерції [1, 2]. Застосування рекомендаційних систем поширюється останнім часом на стаціонарну роздрібну торгівлю, довідкові центри, пошук з програмного забезпечення, наукових статей і т. п. Це застосування характеризується наданням рекомендацій користувачам автоматично, на підставі вже вчинених дій (покупок, виставлених рейтингів, відвідувань тощо) і прийомом від них зворотного зв'язку (замовлення в магазинах, перехід за посиланнями).

При розробці рекомендаційних систем зазвичай розробники стикаються з рядом проблем прогнозування: розрідженість даних, холодний старт, бульбашки фільтрів, синонімія, шахрайство, розмаїття, білі ворони [1, 2], та ін.

Для позбавлення від цих проблем використовуються такі підходи, як колаборативна, контентна та гібридна (поєднання колаборативної та контентної) фільтрації [1, 2].

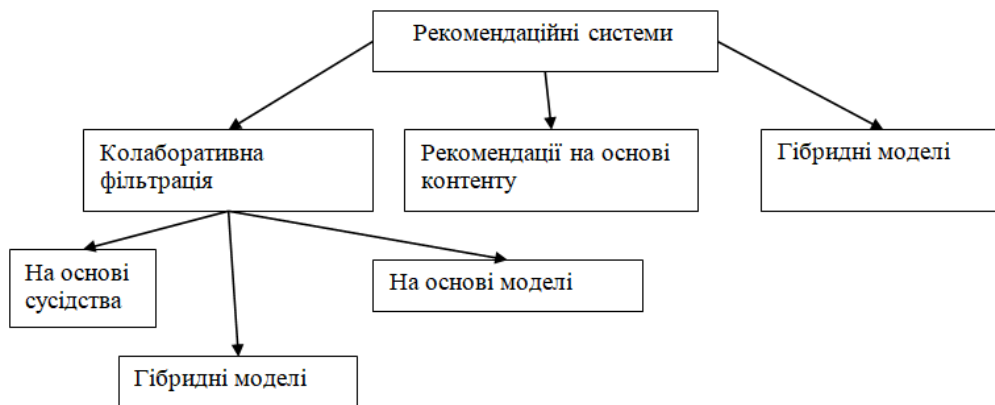


Рисунок 1 - Класифікація рекомендаційних систем

Головною перевагою колаборативної фільтрації є необов'язковість інформації про користувачів або предмети, і тому їх можна використовувати в набагато більшій кількості ситуацій [1, 2].

Фільтрація на основі контенту - тип фільтрації використовує алгоритм для вибору елементів з подібним вмістом, щоб сформувати рекомендацію на підставі того, що використовувалось чи входило до кола уподобань користувача [1, 2].

Гібридні підходи (hybrid) поєднують у собі колаборативну та контентну фільтрацію, підвищуючи ефективність та складність рекомендаційних систем. Об'єднання результатів

колаборативної та контентної фільтрації потенційно дозволяє суттєво підвищити точність рекомендацій. Крім того, гібридний підхід може бути корисним, коли застосування колаборативної фільтрації починається зі значної розрідженості даних («холодний старт»).

Гібридний підхід дозволяє спочатку зважати на результати контентної фільтрації, а потім змішувати ваги відповідних результатів у напрямку колаборативної фільтрації (по мірі наповнення відповідних наборів даних по конкретному користувачеві) [1, 2].

Результат роботи рекомендаційної системи це сформований список рекомендацій.

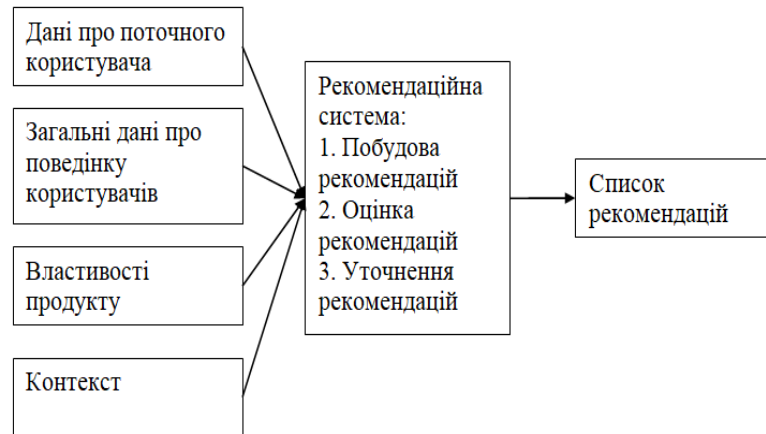


Рисунок 2 - Процес роботи рекомендаційних систем

Для того, щоб система мала високоякісні рекомендації, необхідно постійно оновлювати інформацію про інтереси користувача. Інша проблема – користувач завжди хоче мінімізувати час, що витрачається на взаємодію з системою, і неохоче ділиться інформацією для зворотного зв'язку з системою [1, 2]. Це типова проблема для всіх рекомендаційних систем, яку потрібно вирішувати. Результати аналізу методів формування рекомендацій при побудові рекомендаційних систем – перевагу має гібридний метод формування рекомендацій, в якому враховані зміни у вимогах користувача.

ПЕРЕЛІК ПОСИЛАНЬ:

1. Проблеми сучасних рекомендаційних систем ресурсу. URL: <https://www.altexsoft.com/blog/engineering/pros-and-cons-of-java-programming/>
2. Jalili, M., Ahmadian, S., Izadi, M., Moradi, P., & Salehi, M. (2018). Evaluating collaborative filtering recommender algorithms: a survey. *IEEE access*, 6, 74003–74024. DOI: 10.1109/ACCESS.2018.2883742.

Атаманчук В.Ю., студент кафедри програмних засобів і технологій, Херсонський національний технічний університет, м.Хмельницький, Україна,

Захарченко Л.М., к.т.н., доцент кафедри програмних засобів і технологій, Херсонський національний технічний університет, м.Хмельницький, Україна.

РЕАЛІЗАЦІЯ РЕКОМЕНДАЦІЙНОЇ СИСТЕМИ УПРАВЛІННЯ ПІДБОРОМ ПЕРСОНАЛУ ІТ-ПІДПРИЄМСТВ

Для формування рекомендації в гібридній рекомендаційній системі (РС) управління підбору персоналу до ІТ-фірми необхідно отримати якнайбільше інформації про претендента на ІТ посаду за допомогою колаборативної фільтрації на базі методу МАІ та фільтрації вмісту інформації на базі тестів, співбесіди та інтерв'ю.

Побудова веб-інтерфейсу програми виконана з використанням технології AJAX (Asynchronous Javascript and XML). При оновленні даних на веб-сторінці, система не перезавантажується повністю, а відбувається оновлення потрібної частини контенту веб-сторінки (збільшується швидкість роботи веб-програми) [1-3].

HR менеджер розпочинає роботу зі сторінки вакансій і якщо такі є то вони завантажуються з сайту «rabota.ua» за допомогою API. На сторінці зі списком вакансій є можливість переглянути активні та архівні вакансії. Якщо вакансія обрана то в системі можна переглянути дошку відповідної вакансії, яка була обрана. Виконана реалізація фільтру на дошці вибору вакансії та завантаження претендентів на дошку з використанням технології AJAX без перезавантаження сторінки [1-3].

В системі розроблено множинний вибір претендентів на вакансію для того щоб була можливість надсилання повідомлень на електронну пошту.

Для формування якісної рекомендації в розробленій РС, на основі гібридного методу, управління підбору персоналу до ІТ-фірми, зроблено все для того, щоб отримати якнайбільше інформації про претендента на ІТ посаду за допомогою колаборативної фільтрації на базі методу МАІ та фільтрації вмісту інформації на базі тестів, співбесіди та інтерв'ю.

Для побудови гібридної РС для визначення претендентів на посади ІТ-фахівців використано метод фільтрації вмісту. Відносно даного методу запропоновано додати в програмі, для отримання необхідної інформації про претендентів, можливість пройти тестування, співбесіди чи проведення інтерв'ю. Для зручності користувачеві пропонується обрати тип тесту:

- здібностей;
- особистості;
- проєктивні;
- критеріально-орієнтовані;
- креативності;
- на логіку (Равена 30 хв);
- профільпереваг об'єктів праці (Клімова 5 хв);
- тест «незавершених речень» (15 хв);
- та ін.

Система інтегрована з сайтом по підбору персоналу «rabota.ua», є можливість ведення листування з претендентами та масове відправлення повідомлень. Автоматизація даного процесу підвищує продуктивність праці HR-менеджерів в ІТ-сфері.

ПЕРЕЛІК ПОСИЛАНЬ:

1. Проблеми сучасних рекомендаційних систем ресурсу. URL: <https://www.altexsoft.com/blog/engineering/pros-and-cons-of-java-programming/>
2. Технологія AJAX і сайти в концепції web. URL: <http://www.znannya.org/?view=AJAX-technology>
3. Чалий С.Ф., Лещинський В.О., Лещинська І.О. Моделювання контексту в рекомендаційних системах. Науковий журнал «Проблеми інформаційних технологій», 2018, №. 1(023). С. 21-26.

Балабай Д.Ю., студент кафедри програмних засобів і технологій, Херсонський національний технічний університет, м.Хмельницький, Україна,

Захарченко Р.М., к.т.н., доцент кафедри програмних засобів і технологій, Херсонський національний технічний університет, м.Хмельницький, Україна.

АНАЛІЗ ТА ОПТИМІЗАЦІЯ ПРОДУКТИВНОСТІ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

Продуктивність програмних додатків є критично важливим фактором їхнього успіху та зручності використання. У цьому звіті детально розглянуто аналіз та оптимізацію продуктивності програмного забезпечення, включаючи методології виявлення вузьких місць, інструменти для аналізу продуктивності та стратегії оптимізації. Дослідження включає тематичні дослідження та експериментальні результати, що демонструють вплив методів оптимізації на продуктивність програмного забезпечення.

Оптимізація продуктивності програмного забезпечення передбачає підвищення ефективності програмних додатків для збільшення їхньої швидкості, швидкості відгуку та використання ресурсів. Цей звіт має на меті надати всебічний огляд методів аналізу продуктивності, загальних проблем продуктивності та практичних стратегій оптимізації.

Профілювання - це метод, який використовується для вимірювання продуктивності програмних додатків шляхом збору даних про різні аспекти їх виконання, такі як виклики функцій, використання пам'яті та завантаження процесора. Для профілювання зазвичай використовують такі інструменти, як gprof, VisualVM та Valgrind [1-3].

Бенчмаркінг передбачає проведення серії стандартизованих тестів для оцінки продуктивності програмного забезпечення за різних умов. Бенчмарки можуть бути синтетичними або реальними і корисні для порівняння різних версій програмного забезпечення або різних алгоритмів.

Статичний аналіз досліджує вихідний код без його виконання, щоб виявити потенційні проблеми з продуктивністю, такі як неефективні алгоритми, непотрібні обчислення та витрати пам'яті. Для статичного аналізу використовуються такі інструменти, як SonarQube та Coverity, Clang Static analyzer, і т.і [1-3].

Динамічний аналіз передбачає моніторинг програмного забезпечення під час виконання для збору даних про продуктивність. Сюди входить моніторинг використання пам'яті, завантаження процесора та операцій вводу/виводу. Такі інструменти, як Dynatrace та AppDynamics, надають можливості динамічного аналізу.

Ботлнеки центрального процесора виникають, коли процесор є обмежуючим фактором продуктивності програмного забезпечення. Вони можуть бути спричинені неефективними алгоритмами, надмірними обчисленнями або недостатнім паралелізмом.

Ботлнеки пам'яті виникають через неефективне використання пам'яті, що призводить до надмірного підкачування, витоку пам'яті або фрагментації. Ці проблеми можуть значно сповільнити роботу програм.

І/О Ботлнеки виникають, коли операції вводу/виводу, такі як читання/запис з диска або мережевий зв'язок, обмежують продуктивність. Це може бути пов'язано з високою затримкою або низькою пропускну здатністю пристроїв вводу/виводу [1-3].

Підвищення ефективності алгоритмів - один з найефективніших способів оптимізації програмного забезпечення. Це включає в себе вибір правильних структур даних, оптимізацію циклів та зменшення складності алгоритмів.

Використання паралелізму може значно підвищити продуктивність, особливо на багатоядерних процесорах. Методи включають багатопоточність, паралельну обробку даних з використанням кількох ядер процесору.

Ефективне керування пам'яттю передбачає мінімізацію використання пам'яті, уникнення витоків пам'яті, оптимізації доступу до пам'яті. Методи включають кешування, меморі пулінг, використання ефективних структур даних, ліниву ініціалізація та відкладені алокації і т.і. [1-3].

Оптимізації операцій вводу/виводу можуть включати такі техніки як асинхронний ввід/вивід, буферизацію, кешування, мінімізацію доступу до диску/мережі, і стиснення даних що мають пройти через ввід/вивід.

Веб-додаток мав високу затримку та низьку пропускну здатність. Профілювання виявило вузькі місця процесора через неефективні алгоритми та надмірні запити до бази даних. Оптимізація включала кешування частих запитів, оптимізацію SQL-операторів та впровадження асинхронної обробки. Результатом стало скорочення часу відгуку на 50% та збільшення пропускну здатності на 70%.

Комп'ютерна гра у жанрі пригодницького 3D-екшену мала часті падіння частоти кадрів, статтери та тривалий час завантаження рівнів на більш застарілих конфігураціях комп'ютерів. Інструменти профілювання були використані для виявлення ботлнеків у продуктивності, серед яких були неефективні алгоритми рендерингу, неоптимізоване завантаження ресурсів та погана реалізація багатопотоковості. Процес оптимізації включав такі методи, як оклюжн кулінг, використання ЛОДів, стиснення ассетів, а також краще розпаралелювання завдань що найбільш інтенсивно використовували процесор. В результаті гра досягла стабільної частоти 60 кадрів в секунду на цільовому обладнанні, зі скороченим часом завантаження і більш плавним ігровим процесом [1-3].

Оптимізація продуктивності - це важливий аспект розробки програмного забезпечення, який передбачає системний підхід до виявлення та вирішення проблем продуктивності. Використовуючи профілювання, бенчмаркінг, статичний і динамічний аналіз, а також застосовуючи різні стратегії оптимізації, можна досягти значного покращення продуктивності програмного забезпечення. Цей звіт підкреслює важливість безперервного моніторингу та оптимізації продуктивності для підтримки ефективних і швидких програмних додатків.

ПЕРЕЛІК ПОСИЛАНЬ:

1. Ammons, G., Ball, T., & Larus, J. R. (1997). Exploiting hardware performance counters with flow and context sensitive profiling. ACM SIGPLAN Notices, 32(5), 85-96.
2. Dean, J., & Barroso, L. A. (2013). The tail at scale. Communications of the ACM, 56(2), 74-80.
3. Sutter, H. (2005). The Free Lunch Is Over: A Fundamental Turn Toward Concurrency in Software. Dr. Dobbs's Journal, 30(3).

Блакитний О.А., студент кафедри програмних засобів і технологій, Херсонський національний технічний університет, м.Хмельницький, Україна,

Захарченко Р.М., к.т.н., доцент кафедри програмних засобів і технологій, Херсонський національний технічний університет, м.Хмельницький, Україна.

ВИКОРИСТАННЯ WEBASSEMBLY ДЛЯ ПОКРАЩЕННЯ ПРОДУКТИВНОСТІ ВЕБ-СТОРИНОК

Сучасне інформаційне суспільство це впершу чергу електронна комерція: швидкий доступ до різноманітних товарів і послуг, можливість у порівнянні цін та простота в оплаті та доставці та ін. [1-3]. Вибагливі покупці отримують як необхідний товар так і якісні послуги. В сфері електронної комерції важливу роль відіграють веб-додатки. WebAssembly (WASM) - це інноваційна технологія, яка дає можливість підвищити продуктивність веб-додатків на більш високі рівні, це віртуальна машина для виконання низькорівневого байткоду у веб-браузерах зі значною продуктивністю[1-3]. Це двійковий формат, який запускається разом з JavaScript у веб-браузерах, забезпечуючи виконання коду з швидкістю, близькою до природної. WebAssembly не залежить від платформи і може використовуватися для створення широкого спектру додатків, включаючи ігри, відеоредактори та наукові симулятори [1-3].

WebAssembly це значно швидше виконання коду у порівнянні з традиційним JavaScript, файли WebAssembly менші за обсягом та швидше завантажуються та є можливість використання різних мов програмування (C, C++, Rust тощо) для розробки веб-додатків.

Серед переваг WebAssembly легка інтеграція з існуючим JavaScript кодом.

WebAssembly технологія дозволяє створювати веб-додатки з високою продуктивністю та ефективністю. IT-фахівці мають можливість, використовуючи цю технологію, швидко розроблювати потужні веб-додатки [1-3].

WebAssembly, PWA та ChatGPT є важливими тенденціями у веб-розробці. PWA - це веб-додатки, які можуть запропонувати користувачам досвід, подібний до роботи з додатками, з такими функціями, як офлайн-функціональність, push-сповіщення та можливість встановлення на пристрої користувача. Розробники, поєднуючи WebAssembly з PWA мають можливість створювати високопродуктивні веб-додатки, які забезпечують користувацький досвід, подібний до нативних мобільних додатків [1-3].

В наш час WebAssembly набуває все більшої популярності, оскільки дозволяє розробникам писати код на вже знайомих їм мовах, таких як C++, Rust або Go, а потім компілювати цей код у модуль WebAssembly [1-3]. Ці модулі можна запускати в будь-якому сучасному веб-браузері без необхідності використання плагінів або іншого стороннього програмного забезпечення.

WebAssembly особливо корисна для додатків, які вимагають високої продуктивності або обробляють великі обсяги даних. Він також надає можливість повторно використовувати існуючий код у веб-додатках, що полегшує створення складних додатків з поєднанням фронтенд і бекенд функціональності [1-3].

Це потужний інструмент для веб-розробки, який компілюється в машинний код, який може виконуватися безпосередньо віртуальною машиною браузера. Це дозволяє пришвидшити час завантаження та зробити веб-додатки більш чуйними.

Існує декілька фреймворків та інструментів WebAssembly:

AssemblyScript - це мова проста у використанні для написання високопродуктивних додатків і бібліотек для Інтернету [1-3].

WebAssembly Studio - онлайн-середовище розробки, яке дозволяє розробникам писати, компілювати та налагоджувати код WebAssembly. Надає графічний інтерфейс, який полегшує роботу з WebAssembly навіть для розробників, які не знайомі з цією технологією.

WABT - це набір інструментів, які дозволяють розробникам працювати з бінарними файлами WebAssembly [1-3].

Бібліотека Rust забезпечує міст між Rust і JavaScript, дозволяючи розробникам викликати функції JavaScript з Rust і навпаки.

WebAssembly - це потужна технологія, яка дозволяє розробникам писати високопродуктивні програми на різних мовах і запускати їх безпосередньо в браузері.

Приклади веб-сайтів і додатків, які використовують WebAssembly:

Figma - це браузерний інструмент проєктування, який дозволяє дизайнерам створювати та співпрацювати над дизайном інтерфейсу користувача. Він використовує WebAssembly, щоб забезпечити швидкий і чуйний користувацький інтерфейс, навіть при роботі зі складними проєктами [1-3].

AutoCAD Web - це браузерна версія популярного програмного забезпечення для автоматизованого проєктування. Вона використовує WebAssembly для високопродуктивного рендерингу 2D- і 3D-графіки безпосередньо у браузері.

Doom 3 - це класичний шутер від першої особи, перенесений в Інтернет за допомогою WebAssembly. Вона демонструє можливості WebAssembly для запуску високопродуктивних ігор у браузері.

OpenCV - це популярна бібліотека комп'ютерного зору, яка була перенесена на веб за допомогою WebAssembly.

Unity - популярний ігровий рушій, який було портовано на веб за допомогою WebAssembly.

WebAssembly демонструє продуктивність веб-розробки в напрямку створення високопродуктивних, зручних і персоналізованих веб-додатків. Ці тенденції можуть докорінно змінити спосіб взаємодії з веб-додатками та надати розробникам нові можливості для створення потужних та інноваційних рішень.

ПЕРЕЛІК ПОСИЛАНЬ:

1. WebAssembly і можливості оптимізації продуктивності веб-додатків 2024. URL: <https://it-rating.ua/webassembly-i-mojlivosti-optimizatsii-produktivnosti-veb-dodatviv>
2. Використання WebAssembly для оптимізації продуктивності веб-додатків 2023. URL: <https://it-rating.ua/vikoristannya-webassembly-dlya-optimizatsii-produktivnosti-veb>
3. Майбутнє веб-розробки: Тренди та технології, на які варто звернути увагу. URL: <https://stfalcon.com/uk/blog/post/web-development-trends-part-2>

Блакитний О.А., студент кафедри програмних засобів і технологій, Херсонський національний технічний університет, м.Хмельницький, Україна,

Захарченко Л.М., викладач-стажист кафедри програмних засобів і технологій, Херсонський національний технічний університет, м.Хмельницький, Україна.

ЯК CHATGPT МОЖЕ ПОКРАЩИТИ ВЕБ-РОЗРОБКУ

Chat GPT - цей ШІ є не просто корисною технологією, але його сприймають як безмежні можливості в роботі компаній у різних галузях. Наприклад, в ІТ-сфері Chat GPT може покращити веб-розробку декількома способами [1-4]. Chat GPT можна інтегрувати у веб-додатки для забезпечення покращення користувацького досвіду, залучаючи їх до розмов природною мовою. Це можливість максимально збільшити утримання користувачів. Chat GPT навчають розпізнавати вподобання користувачів і як результат надавати персоналізовані рекомендації, які відповідають їхнім вподобанням. За допомогою такої методики можна отримати максимальну задоволеність веб-сайтом користувачами. Для зменшення навантаження на агентів служби підтримки веб-сайту Chat GPT може давати відповіді користувачам на поширені запитання та виступати як інструмент підтримки клієнта [1-4]. Chat GPT може оптимізувати робочі процеси, автоматизуючи планування зустрічей, прийом замовлень та ін. завдань. Chat GPT можна використовувати для створення високоякісного контенту для веб-сайтів, що може покращити рейтинг у пошукових системах і залучити більше трафіку на сайт. Використовуючи Chat GPT для створення контенту, веб-розробники можуть переконатися, що їхній сайт оптимізований як для користувачів, так і для пошукових систем [1-4].

Приклади веб-сайтів і додатків на основі Chat GPT. Slack є провідною платформою для обміну бізнес-повідомленнями, вона інтегрувала Chat GPT як один із своїх програмних додатків. Користувачі спілкуються зі своїми колегами за допомогою ШІ, а Chat GPT пропонує текст, що налаштовується, для написання повідомлень. ШІ надає резюме цілих каналів або окремих тем для обговорення. WriteMage надає системний доступ до ChatGPT і сумісний з macOS та iOS. Shopify, платформа електронної комерції, інтегрувала ChatGPT у свій додаток для смартфонів Shop, щоб пропонувати користувачам персоналізовані поради щодо товарів [1-4].

Chat GPT демонструє еволюцію веб-розробки в напрямку створення високопродуктивних, зручних і персоналізованих веб-додатків [1-4].

ПЕРЕЛІК ПОСИЛАНЬ:

1. WebAssembly і можливості оптимізації продуктивності веб-додатків 2024. URL: <https://it-rating.ua/webassembly-i-mojlivosti-optimizatsii-produktivnosti-veb-dodatkov>
2. Використання WebAssembly для оптимізації продуктивності веб-додатків 2023. URL: <https://it-rating.ua/vikoristannya-webassembly-dlya-optimizatsii-produktivnosti-veb>
3. <https://stfalcon.com/uk/blog/post/web-development-trends-part-2>
4. Переваги Chat GPT у бізнесі.2023р. URL: <https://webpromoexperts.net/ua/blog/noviy-riven-vzaimodiyi-z-kliientami-chat-gpt-u-biznesi/>

Бойко М.Є., студент кафедри програмних засобів і технологій, Херсонський національний технічний університет, м.Хмельницький, Україна,

Огнєва О.Є., к.т.н., доцент кафедри програмних засобів і технологій, Херсонський національний технічний університет, м.Хмельницький, Україна.

ПРОЄКТУВАННЯ ТА РОЗРОБКА РЕЛЯЦІЙНОЇ БАЗИ ДАНИХ ДЛЯ WEB ДОДАТКУ «ІНТЕРНЕТ МАГАЗИН ЕЛЕКТРОНІКИ»

Інформаційні потоки — це фізична передача інформації від одного співробітника компанії до іншого або від одного відділу до іншого [1].

Система інформаційного потоку – це сукупність фізичних інформаційних рухів, які дають змогу здійснити будь-який процес, здійснити певне рішення. Найбільш загальна система інформаційних потоків – це сума інформаційних потоків, які дають змогу підприємству здійснювати фінансово-господарську діяльність [3-4].

Найважливішою ознакою процесу управління є його інформаційний характер. Організація виконання ухвалених рішень здійснюється системою методів впливу на працівників з використанням інформації про хід виконання ухвалених рішень (інформація зворотного зв'язку). Чим точнішою та об'єктивнішою є інформація, якою володіє система управління, тим краще вона відображає поточний стан і взаємозв'язки в об'єкті управління, тим точніше обґрунтування цілей і реальних заходів, спрямованих на їх досягнення [3].

Оскільки керівник у своїй роботі спирається на інформацію про стан об'єкта і своєю діяльністю створює нову командну інформацію з метою переведення керованого об'єкта з фактичного стану в бажаний, то інформація умовно розглядається предмет і результат управлінської праці.

Інформація як елемент управління і предмет управлінської праці повинна забезпечувати якісне розуміння завдань і стану керованих і керуючих систем і забезпечувати розробку ідеальних моделей їх бажаного стану [1, 3].

Таким чином, інформаційне забезпечення – це частина системи управління, яка являє собою сукупність даних про фактичний і можливий стан факторів виробництва і зовнішніх умов під час функціонування виробничого процесу, про логіку змін і перетворень факторів виробництва [4]. Картотека пацієнтів – виконання завдань з підвищення ефективності роботи реєстратури в поліклініці; автоматизація ведення обліку пацієнтів.

Об'єктом дослідження є робота сайту для інтернет магазину, перегляд складу, додавання товарів, обробка замовлень. Предметом дослідження є побудова бази даних для автоматизації процесів роботи магазину.

Для створення бази даних необхідно вивчити предметну область задачі, визначити моделі, методи та алгоритми, які будуть використовуватися при постановці задачі та її математичну модель, необхідно вибрати програмне забезпечення та розробити програмний продукт, що складається з бази даних і алгоритмів їх обробки та зображення.

phpMyAdmin — це веб-інтерфейс для управління інфраструктурою MySQL та MariaDB. Використовується для доступу, налаштування, адміністрування та розробки всіх компонентів MySQL і MariaDB, а також для їх управління. Середовище phpMyAdmin надає єдину повнофункціональну службову програму, яка поєднує в собі зручні графічні інструменти з потужними редакторами сценаріїв для доступу до MySQL і MariaDB, що робить його ідеальним інструментом для розробників і адміністраторів баз даних всіх професійних рівнів.

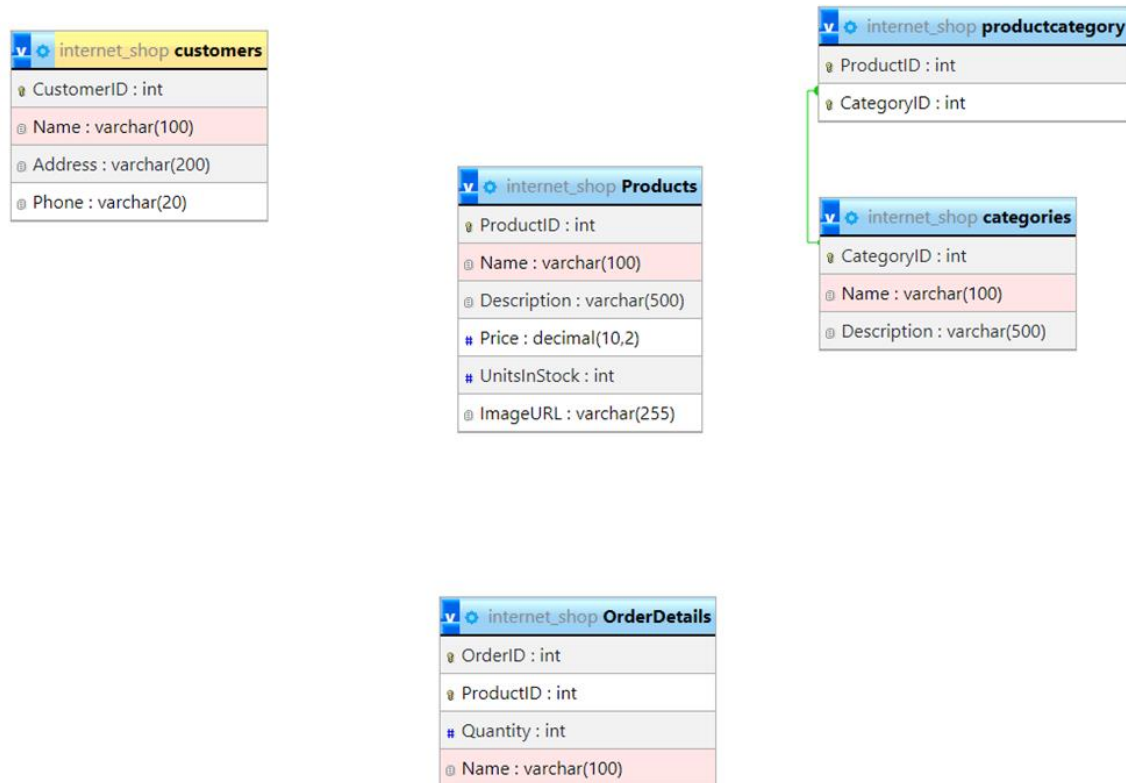


Рисунок 1 - Схема зв'язку даних в базі даних, яка зпроектована

Система інформаційного забезпечення-це сукупність даних про цілі, стан, напрямки розвитку об'єкта і навколишнього його середовища, організована під взаємопов'язаних потоках відомостей. Ця система включає методи одержання, зберігання, сортування, обробки даних і видачі їх користувачеві.

Набір команд і функцій, що запропоновані розробниками програмних продуктів у середовищі PhpMyAdmin, відповідають більшості сучасних вимог до подання та обробки даних, створення баз даних.

ПЕРЕЛІК ПОСИЛАНЬ:

1. Дударенко А.І., Пивоваров М.В. Інформаційні системи в менеджменті: Навчальний посібник. Київ: Знання, 2020. 367 с.
2. Гнатенко О.Ю., Чернявський О.О. Сучасні інформаційні системи та технології: Підручник. Харків: ХНЕУ ім. С. Кузнеця, 2021. 512 с.
3. Головачова Л.О. Інформаційні технології в управлінні: Навчальний посібник. Одеса: ОНУ, 2020. 248 с.
4. Сидоренко С.В. Інформаційні системи і технології в управлінні організацією: Підручник. Львів: ЛНУ ім. І. Франка, 2022. 395 с.

Валюга В.В., студент кафедри програмних засобів і технологій, Херсонський національний технічний університет, м.Хмельницький, Україна,

Комісаров О.С., ст.викладач кафедри програмних засобів і технологій, Херсонський національний технічний університет, м.Хмельницький, Україна.

ІНТЕГРАЦІЯ ШТУЧНОГО ІНТЕЛЕКТУ У ВЕБ-ДОДАТКИ

Штучний інтелект (ШІ) — це галузь комп'ютерних наук, яка займається створенням комп'ютерних систем, здатних виконувати завдання, що зазвичай вимагають людського інтелекту. Це можуть бути завдання, пов'язані з навчанням, розпізнаванням мови, візуальним сприйняттям, прийняттям рішень та іншими когнітивними функціями.

ШІ в даний час використовується не лише користувачами для пошуку інформації, створення чи малювання, але й широко застосовується в робочих проєктах у різноманітних сферах. Однією з таких сфер є веб-програмування, де штучний інтелект надає велику кількість інструментів та функцій для прискорення та покращення роботи додатків. Можна відокремити наступні функції[1]:

- розробка комерційних пропозицій для потенційних замовників веб-студій;
- скорочення часу на технічний аспект у ході проєктування, формування коду та подальшого тестування. З цією метою ШІ можна поєднувати з ІСР (інтегрованими середовищами розробки);
- персоналізація, тобто створення веб-ресурсів, що повністю відповідають індивідуальним потребам замовників від стилістики до функціоналу;
- формування технічних завдань на кожному з етапів реалізації стратегії розробки та просування сайту;
- розробка унікального дизайну, генерування зображень найвищої якості за особистими ескізами та напрацюваннями;
- збір та кластеризація семантики;
- генерація великих обсягів унікального, професійного текстового контенту: статей для блогів, описів товарів інтернет-магазинів, продаючих текстів для лендингів, e-mail розсилок і т.д. Оскільки успішна розкрутка сайту багато в чому залежить від якісного контенту, це робить ШІ для генерації текстів дуже затребуваним;
- професійний переклад контенту іншими мовами з метою розробки мультимовних веб-ресурсів;
- ефективне спілкування з клієнтами, відвідувачами сайтів через чат-боти, які використовують моделі природної мови;
- автоматизація процесу лінкбілдингу, що базується на закупівлі зовнішніх посилань.

Чималу функцію у наданні інструментарію зі штучним інтелектом відіграють сервіси, які володіють своїм штучним інтелектом, або інтегрують вже існуючі рішення у свої бізнес-проєкти. Найпоширенішими та найбільш актуальними є такі сервіси, як: Google Cloud AI, що надає набір інструментів і API від Google для машинного навчання, включаючи Vision API, Speech-to-Text API, Natural Language API, OpenAI, яка відома своїми моделями на зразок GPT-3, OpenAI надає API для інтеграції передових моделей природної мови у додатки. Дуже актуальною також є платформа EdenAI, яка надає єдиний API для доступу до різних постачальників штучного інтелекту. Вона дозволяє інтегрувати різні сервіси ШІ, такі як обробка природної мови, розпізнавання зображень, генерація тексту та інші, через єдиний інтерфейс. Це спрощує процес вибору та інтеграції ШІ-інструментів від різних постачальників у ваші веб-додатки.

Інтеграція штучного інтелекту (ШІ) у веб-додатки надає численні переваги, які підвищують ефективність, продуктивність та якість взаємодії з користувачами. Ось деякі з основних переваг [2]:

- автоматизація рутини та повторювальних завдань, такі як відповіді на запити клієнтів через чат-боти, управління контентом та аналіз даних;
- надання розумних рекомендацій, що підвищують релевантність контенту для користувачів та персоналізують взаємодію з ними;
- виявлення та попередження загроз безпеці за допомогою аналізу поведінкових даних, що дозволяє знизити ризик атак;
- оптимізація використання серверних ресурсів та мінімізація витрат на інфраструктуру за рахунок алгоритмів машинного навчання;
- аналіз великих обсягів споживчих даних для виявлення нових ринкових тенденцій та уподобань, що сприяє розробці інноваційних функцій та продуктів;
- автоматизація тестування та контроль якості продуктів, що забезпечує швидше виявлення помилок та підвищення якості;
- аналіз великих даних з метою оптимізації процесів, виявлення закономірностей та створення моделей прогнозування;
- побудова симулятивних сценаріїв, які допомагають у прийнятті рішень та адаптації до змін ринкових умов;
- автоматизація небезпечних завдань, що зменшує ризик травматизму та помилок, пов'язаних з людським фактором;
- моніторинг виконання безпекових вимог та попередження про потенційні порушення.
- автоматичне надання користувачам персоналізованих пропозицій, відповідей та підтримки, що покращує якість обслуговування;
- автоматизація обробки клієнтських запитів, що значно скорочує час реакції та підвищує задоволеність користувачів.

Отже, інтеграція штучного інтелекту у веб-додатки є дуже актуальною та прогресивною тематикою, яка привносить багато інновацій у розробку. Використання ШІ не тільки спрощує процеси розробки та підвищує ефективність, але й створює нові можливості для персоналізації, автоматизації та оптимізації. Застосування інструментів ШІ дозволяє розробникам створювати більш інтерактивні та інтуїтивні веб-ресурси, що відповідають індивідуальним потребам користувачів.

ПЕРЕЛІК ПОСИЛАНЬ:

1. Основні функції ШІ у веб-розробці. URL: <https://voll.com.ua/uk/blog/shtuchnij-intelekt-cifrove-majbutnye-dostupne-kozhnomu>
2. Переваги використання штучного інтелекту у веб-додатках. URL: <https://wezom.com.ua/ua/blog/shtuchnij-intelekt-u-virobnitstvi-perevagi-ta-prikladi-zastosuvannya>

Горностаєв Д.С., студент кафедри програмних засобів і технологій, Херсонський національний технічний університет, м.Хмельницький, Україна,

Огнєва О.Є., к.т.н., доцент кафедри програмних засобів і технологій, Херсонський національний технічний університет, м.Хмельницький, Україна.

ДОСЛІДЖЕННЯ РОЗВИТКУ ШІ, ЙОГО СУЧАСНОГО СТАНУ ТА ВПЛИВУ НА РІЗНІ СФЕРИ ЖИТТЯ

Розвиток штучного інтелекту (ШІ) є однією з найважливіших і найперспективніших галузей сучасних технологій, що має значний вплив на різні аспекти життя суспільства [1-2].

Витоки ШІ можна простежити до 1950-х років, коли Алан Тьюрінг запропонував свій тест для визначення машинного інтелекту. Тьюрінг припустив, що люди використовують розум для розв'язання завдань та прийняття рішень, то чому б машинам не робити те ж саме? Ця ідея встала в основу його статті 1950 року "Обчислювальна техніка та інтелект", в якій він описав, як створювати розумні машини та як перевіряти їх інтелект [1-2].

З 1957 по 1974 рік штучний інтелект процвітав. Комп'ютери могли зберігати більше інформації, ставали швидшими, дешевшими та доступнішими. Алгоритми машинного навчання також удосконалювалися, і люди стали краще розуміти, який алгоритм застосовувати для вирішення завдань. У 1960-1970-х роках були створені перші програми, що могли грати в шахи та проводити діалог з людиною [1-2].

1990-ті роки відзначилися значним прогресом завдяки розвитку комп'ютерної техніки та алгоритмів. 1997 році чемпіон світу з шахів та гросмейстер Гаррі Каспаров зазнав поразки від комп'ютерної програми для гри в шахи (IBM Deep Blue). Цей широко розрекламований матч став першою поразкою чинного чемпіона світу з шахів від комп'ютера та значним кроком для розвитку штучного інтелекту [1-2].

Розвиток нейронних мереж і алгоритмів глибокого навчання зробив можливим створення систем, що можуть розпізнавати зображення, голос та мову. Робототехніка та автономні транспортні засоби, що використовують ШІ, стали реальністю. ШІ застосовується в діагностиці хвороб, розробці нових ліків та персоналізованій медицині.

Можливість виконувати повторювані операції в обробній промисловості полегшує життя людей завдяки сучасним технологіям. У результаті працівники мають можливість зосередитись на складніших завданнях. Нейромережа дозволяє персоналізувати рекомендації щодо продуктів. ШІ допомагає підприємствам у сфері маркетингу та комерції, надаючи персоналізовані рекомендації, які можуть покращити взаємодію з клієнтами.

Щороку Україна продовжує розвивати сферу інноваційних технологій та запроваджувати їх у сучасне життя суспільства. Штучний інтелект не став виключенням. У грудні 2020 року Кабмін схвалив план розвитку ШІ до 2030 року. Ухвалення цієї концепції продемонструвало, що Україна готова йти в ногу з сучасними технологіями.

Етика штучного інтелекту, у свою чергу, є сукупністю принципів відповідального ШІ — як він має чинити, а як не повинен. Ці принципи часто мають на увазі безпеку, справедливість та орієнтованість на людину.

Популяризатором явища вважається Айзека Азімова — автора оповідання «Хоровод». У ньому письменник уперше сформулював «Три закони робототехніки». Пізніше у романі «Роботи та імперія» Азімов доповнив список нульовим правилом. Вони звучать приблизно так: Робот не може своєю дією завдати шкоди людству, або ж внаслідок бездіяльності допустити, щоб людству було завдано шкоди. Робот не може своєю дією завдати шкоди людині, або ж внаслідок бездіяльності допустити, щоб людині було завдано шкоди. Робот повинен підкорятися наказам людини, за винятком випадків, коли ці накази суперечать

Першому закону. Робот повинен захищати свою безпеку доти, доки вона не суперечить Першому чи Другому закону [1-2].

13 березня 2024 року Європейський парламент прийняв історичний документ – Закон про штучний інтелект (Artificial Intelligence Act). Його метою є поліпшення функціонування внутрішнього ринку та сприяння впровадженню людино-центрованого та надійного штучного інтелекту, забезпечуючи при цьому високий рівень захисту здоров'я, безпеки та основних прав. Пріоритетом є забезпечення того, що системи штучного інтелекту будуть безпечними, прозорими, можуть бути відслідковуваними, не дискримінаційними та екологічно безпечними [1].

Розвиток штучного інтелекту має великий потенціал для покращення життя людей і сприяє науково-технічному прогресу. Подальші дослідження в галузі ШІ мають на меті створення більш безпечних, ефективних та етичних систем, що сприятимуть сталому розвитку суспільства.

ПЕРЕЛІК ПОСИЛАНЬ:

1. Еволюція штучного інтелекту (ШІ): Визначні моменти в історії та застосування. URL: <https://cases.media/en/article/evolyuciya-shtuchnogo-intelektu-shi-viznachni-momenti-v-istoriyi-ta-zastosuvannya>
2. Meta та ШІ: як компанія вирішує проблему дезінформації. URL: <https://cases.media/en/article/meta-ta-shi-yak-kompaniya-virishuye-problemu-dezinformaciyi>

Довгаль О.О., студент кафедри програмних засобів і технологій, Херсонський національний технічний університет, м.Хмельницький, Україна,

Огнєва О.Є., к.т.н., доцент кафедри програмних засобів і технологій, Херсонський національний технічний університет, м.Хмельницький, Україна.

СТВОРЕННЯ СЕРВІСУ ДЛЯ ПЕРЕВІРКИ БРОНЮВАНЬ НА ОСНОВІ КЛІЄНТ-СЕРВЕРНОЇ АРХІТЕКТУРИ

Ми живемо у світі де люди з усього світу мандрують по світу та бронюють багато готелів, орендують машини, або замовляють квитки на авіарейси. Усі ці типи продуктів реєструються за допомогою спеціально розроблених глобальних розподільчих систем, або GDS (Global Distribution System). На даний момент, найвпливовішими компаніями які підтримують такі системи є Amadeus, Sabre, Travelport (Apollo, Galileo, Worldspan).

Глобальна система розподілу, або GDS, є комп'ютерною мережею, що діє як посередник між турагентами та численними постачальниками туристичних послуг. Він збирає інвентаризацію, розклади та тарифи від постачальників і дає OTA (Online Travel Agency) та іншим посередникам можливість шукати та бронювати їх за допомогою API підключення або агентських платформ.

Існують багато компаній які працюють над розробкою систем бронювань та виконують незліченну кількість бронювань для проведення тестування їх систем на працездатність. Тестові бронювання є такими бронюваннями які потребують використання тестових даних у реальних (!) бронюваннях. Такі дії потребують високої уваги, бо якщо забути відмінити бронювання, то готель, або компанія яка його зареєструвала, буде вимагати грошей на його оплату. Якщо брати у загальному плані те, які бронювання є найнебезпечнішими, то їх можна проставити за таким пріоритетом, від безпечних до найризикованіших:

1. Автомобільні — оренда автомобілів, виклик автомобіля на якийсь час та сплата за його використання на період який ми взяли. Може бути як звичайний із карткою, так із оплатою на місці: мається на увазі що якщо бронювання здійснено таким чином, то оплата буде здійснена лише при наявності людини, яка бронювала, тому це є найбезпечніший тип бронювання.

2. Готельні — цей тип бронювання є середнього рівня ризику. Цей тип можна виконувати як через спеціальний механізм тестового бронювання, вставляючи спеціально зарезервований формат кредитної картки (може бути або VISA, або Mastercard), але тестування такого бронювання є неповним, тому використовують тестову кредитну картку, але у даному випадку, якщо прийде час оплати за бронювання, готель або компанія буде вимагати оплату, що може спричинити дуже неприємні моменти для тих, хто робить розробку систем.

3. Літакові — це найризикованіший тип бронювань. Ризик полягає у тому що зазвичай, компанії авіаліній хочуть оплату одразу або оплату після деякого часу. Таким чином такі бронювання треба відмінити дуже швидко, рекомендовано відмінити протягом години. Ще ризик у тому що літакові бронювання білетів є дуже дорогим. Зазвичай від 200 і більше Євро.

Якщо перейти до суті діла, а саме до кореня роботи системи бронювань, або як їх ще називають «reservation system», то найголовнішим у структурі бронювань є так званий PNR (Passenger Name Record) — це запис номеру пасажирів. На початку він слугував як загальний запис для усіх літакових бронювань, рейсів у аеропортах. Приклад запису PNR наведений на рисунку 1 на наступній сторінці.

RP/SF01S216Q/SF01S216Q AA/SU 9DEC14/0447Z 7JSGUA
 1.DIBLAZIO/BILL
 2 NNL IL HK1 SDX IN23APR OUT24APR 1C1KRAC USD115.00 DLY 021
 OAK CREEK TERRACE/BC-C1KRAC/BS-05772782/CF-BY9LD86HP
 /G-CCVIXXXXXXXXXXXXX4905EXP0817 *IL+
 SEE RTSVCH
 3 AP E ASTRYGIN@GETGOING.COM
 4 AP H+1-510-813-7969
 5 TK OK08DEC/SF01S216Q

Рисунок 1 - Приклад PNR готельного бронювання із сегментом NNL

З цього ми можемо сказати, що завдяки такій властивості записів PNR ми маємо можливість відмінити окремі сегменти, тобто мати декілька бронювань у одному PNR але активними у БД будь лише ті, де їх сегмент існує.

Також є одна особливість для бронювань білетів літаків — вони мають окремо ще аналог PNR, а саме локатори.

Локатор авіакомпаній — це комбінація коду з 6 літер, яка використовується для пошуку та ідентифікації бронювання авіакомпанії. Кожне дійсне бронювання, підтверджене авіакомпаніями, матиме локатор записів авіакомпаній. Приклад локаторів: REL33W, K14GX5, PFFYCS.

Таким чином, ми приходимо до основної проблеми яку треба вирішити: перевірка статусу бронювань. Для вирішення цієї проблеми нам потрібно розуміти спосіб роботи із GDS та провайдерами. Ми маємо можливість «спілкуватися» із ними через їх API.

Задля цього ми можемо створити на основі клієнт-серверної архітектури сервіс, який буде приймати дані про бронювання, а саме його унікальний ідентифікатор у БД де зберігаються усі виконані бронювання та PNR у якому він зареєстрований.

Для створення серверної частини підходять 2 бібліотеки [1-5]:

1. Flask — це мікрофреймворк Python, що використовується для створення веб-застосунків. Він простий у вивченні та використанні, роблячи його чудовим вибором для початківців. Flask також дуже гнучкий, що робить його ідеальним для широкого спектру застосунків.

2. FastAPI — це сучасний фреймворк Python, що використовується для створення API. Він заснований на Flask, але пропонує ряд додаткових функцій, таких як автоматична генерація документації OpenAPI та вбудована підтримка аутентифікації та авторизації.

Для бази даних можна використати PostgreSQL (також відомий як PostgreSQL) - це вільна та відкрита реляційна система керування базами даних (СКБД), що відрізняється надійністю, потужністю та гнучкістю.

Приклад того як буде виглядати так званий звіт від розробленого сервісу перевірок бронювань у випадку їх активного статусу на рисунку 2:

Product	Provider	Unique ID / Additional ID	Booker email / Traveler email	Environment / Host	Checkin / Checkout, Pickup / Dropoff, Departure / Arrival	Suspicion	Booking errors / Cancellation errors	Created at	ID
hotel	EPS	4dudv3zmy5xmf / 9033783407990	TFA-UAT / [REDACTED]	prod / [REDACTED]	Tue, Jul 30, 2024 / Wed, Jul 31, 2024	Uncheckable provider eps	Booking error: No errors / Cancellation errors: No cancellations found	2024-02-15 13:16:09	7550
hotel	EPS	66iqu4baszlwbyb / 9033784214608	Cytric QA / [REDACTED]	uat / [REDACTED]	Mon, Mar 25, 2024 / Fri, Mar 29, 2024	Uncheckable provider eps	Booking error: No errors / Cancellation errors: No cancellations found	2024-02-15 14:24:58	7553
hotel	SABRE	khlpg3jph67sq / GHYRRA	tsh_air_api / [REDACTED]	uat / [REDACTED]	Fri, Apr 19, 2024 / Sat, Apr 20, 2024	Segment(s) 3484453784 not canceled	Booking error: No errors / Cancellation errors: No cancellations found	2024-02-19 14:51:44	7945
hotel	AMADEUS	532audv2qf47x26 / P4XNP5	tsh_air_api / [REDACTED]	uat / [REDACTED]	Fri, Apr 19, 2024 / Sat, Apr 20, 2024	Segment(s) 2 not canceled	Booking error: No errors / Cancellation errors: No cancellations found	2024-02-19 17:49:24	7998

Рисунок 2 - Приклад сповіщення про невідмінені бронювання

У додаток до унікальності можна сказати, що у світі поки немає такого програмного продукту, який би регулював процес бронювання та відміни і перевіряв бронювання на їх статус і потребу відмінити. Дану систему можна також налаштувати під свої потреби, наприклад для звітувань, аудитів або також спростити алгоритм перевірки зменшуючи кількість запитів та роблячи декомпозицію можна зробити склад програми більш логічним та оптимізованим, який можна буде легко підтримувати.

Схема подібної системи показана на рис. 3:

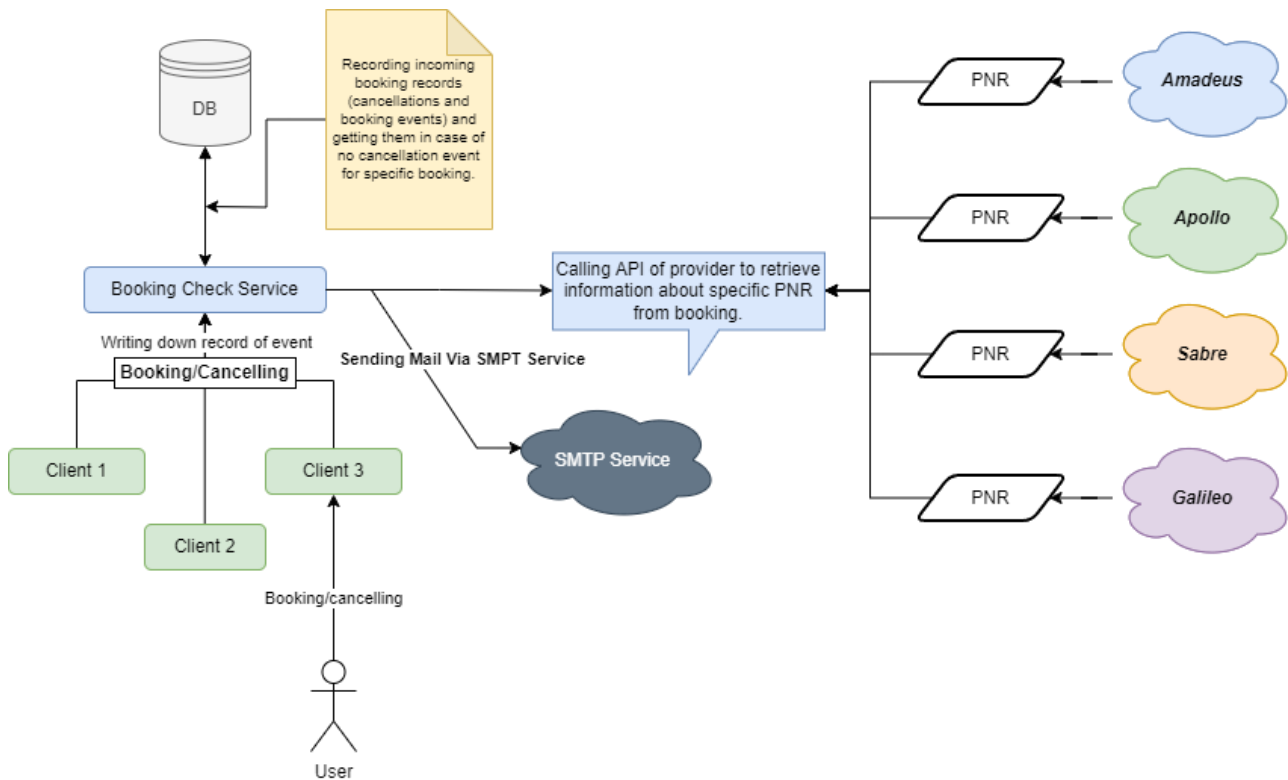


Рисунок 3 - Діаграма будови сервісу для перевірки бронювань.

ПЕРЕЛІК ПОСИЛАНЬ:

1. Пояснення та історія PNR. URL: <https://hasbrouck.org/articles/PNR.html>
2. Пояснення про локатори. URL: <https://www.cheapair.com/help/flights/what-is-an-airline-record-locator/>
3. Flask. URL: <https://flask.palletsprojects.com/en/2.3.x/>
4. FastAPI. URL: <https://fastapi.tiangolo.com/>
5. PostgreSQL. URL: <https://www.postgresql.org/>

Дудник Б.М., студент кафедри програмних засобів і технологій, Херсонський національний технічний університет, м.Хмельницький, Україна,

Захарченко Р.М., к.т.н., доцент кафедри програмних засобів і технологій, Херсонський національний технічний університет, м.Хмельницький, Україна.

ВИКОРИСТАННЯ ІТ В РОЗРОБЦІ ВЕБ-ДОДАТКУ ДЛЯ МАГАЗИНУ КНИЖОК

У сучасному інформаційному суспільстві роль веб-додатків у сфері комерції та обміну інформацією стає все більш значимою. Швидкий розвиток інтернет-технологій та поширення електронної комерції сприяють ефективному впровадженню онлайн-платформ для здійснення покупок, замовлень та інших трансакцій [1, 2].

Було проведено дослідження онлайн-магазинів книжок, які забезпечують зручний доступ до літературних творів та розширюють можливості користувачів у виборі та пошуку необхідної інформації.

Поставлене завдання - розробка зручного веб-додатку для магазину книжок з використанням мови програмування PHP та фреймворка Bootstrap.

Метою даного проєкту стала розробка функціонального та зручного для користувача веб-додатку, який забезпечує можливість здійснення покупок книжок он-лайн, а також надає інформацію про нові надходження та акції магазину.

За даною темою було проведено дослідження можливостей реалізації веб-сайтів та вивчено теоретичні аспекти веб-розробки, зокрема бази даних та архітектурні підходи до створення веб-додатків. Також у роботі описано функціональність та особливості магазину книжок, а також їх реалізація з використанням PHP. Особливу увагу приділено розробці інтерфейсу користувача з використанням Bootstrap для забезпечення адаптивності та естетичності веб-додатку. В результаті розроблено пропозиції щодо подальшого вдосконалення додатку.

На рис. 1-6 представлено вікна розробленого веб-додатку.

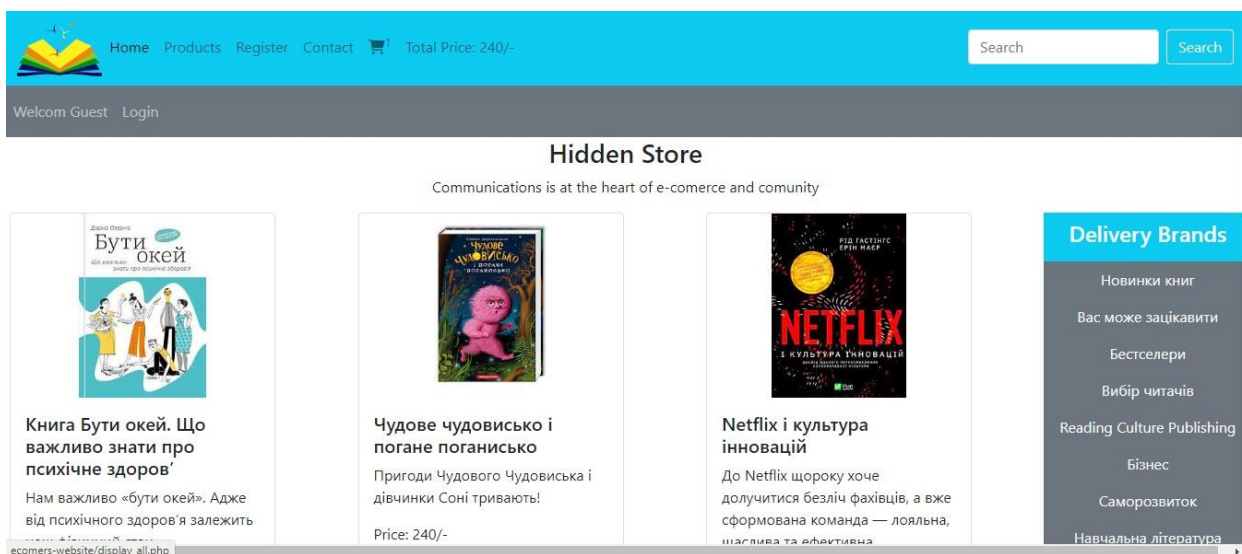


Рисунок 1 - Головна сторінка

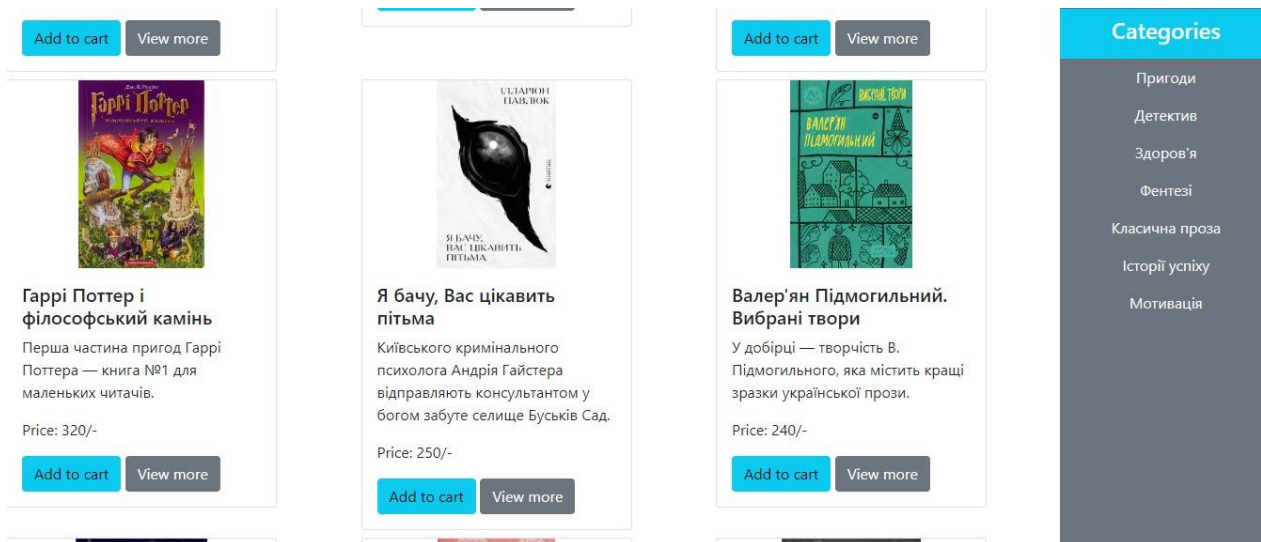


Рисунок 2 - Головна сторінка

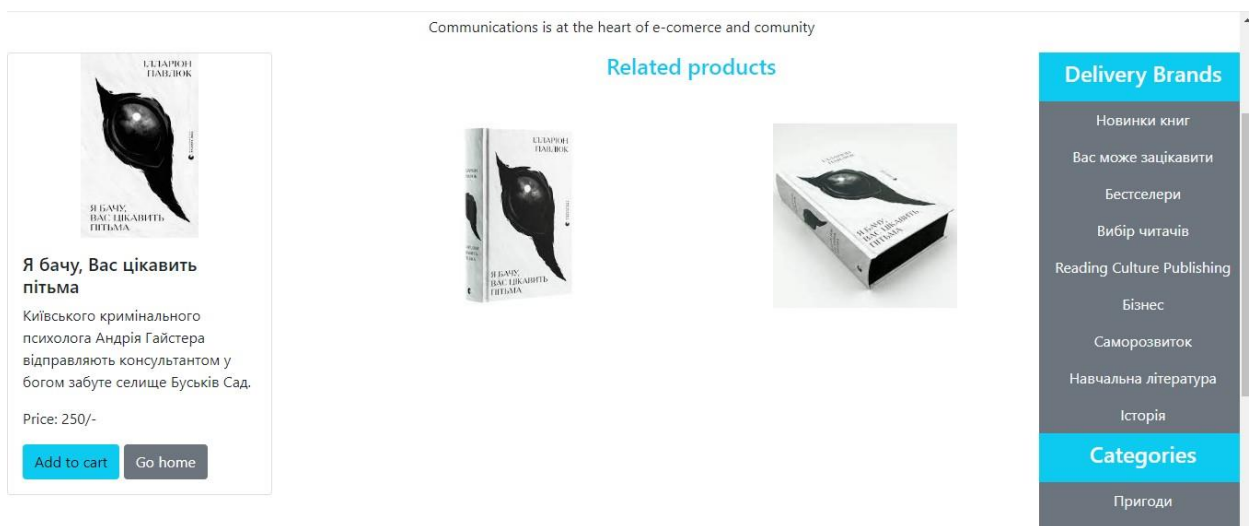


Рисунок 3 - Інформація про товар

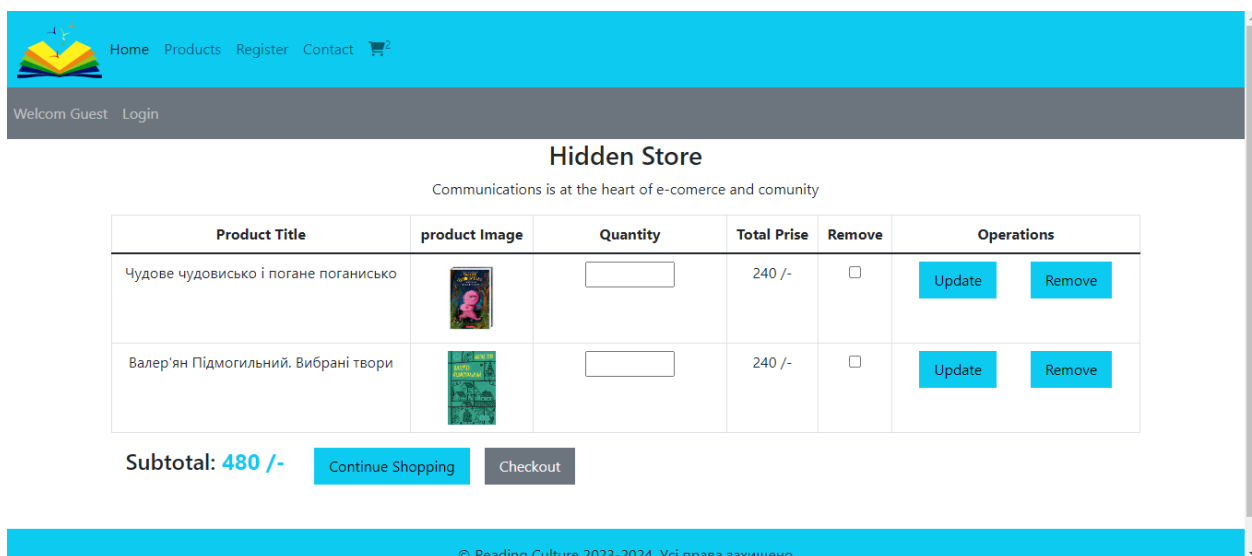


Рисунок 4 - Кошик

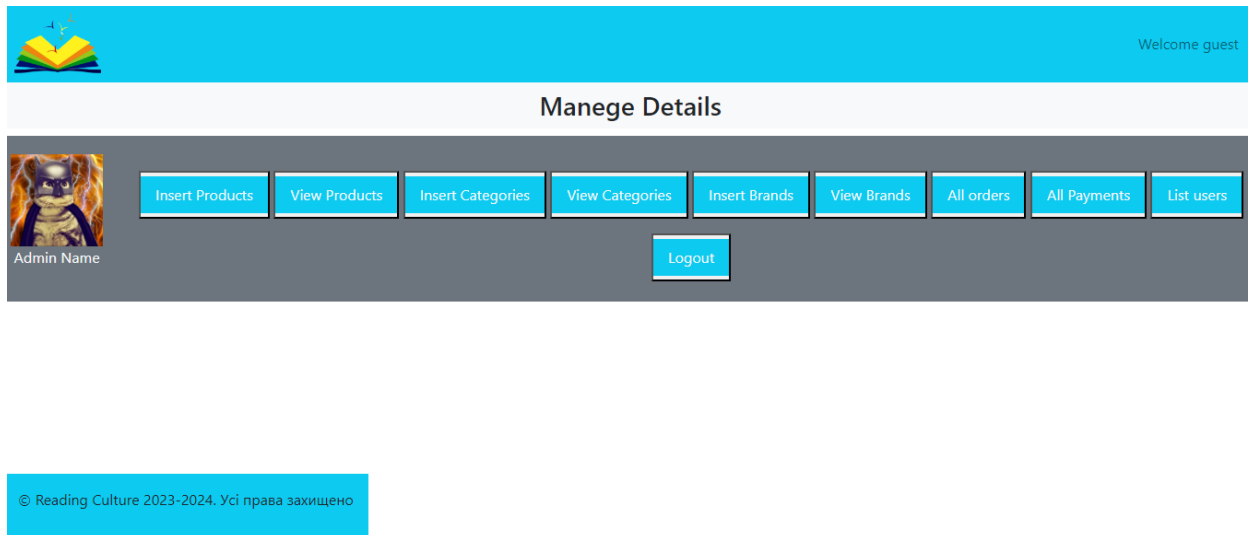


Рисунок 5 - Адмін панель

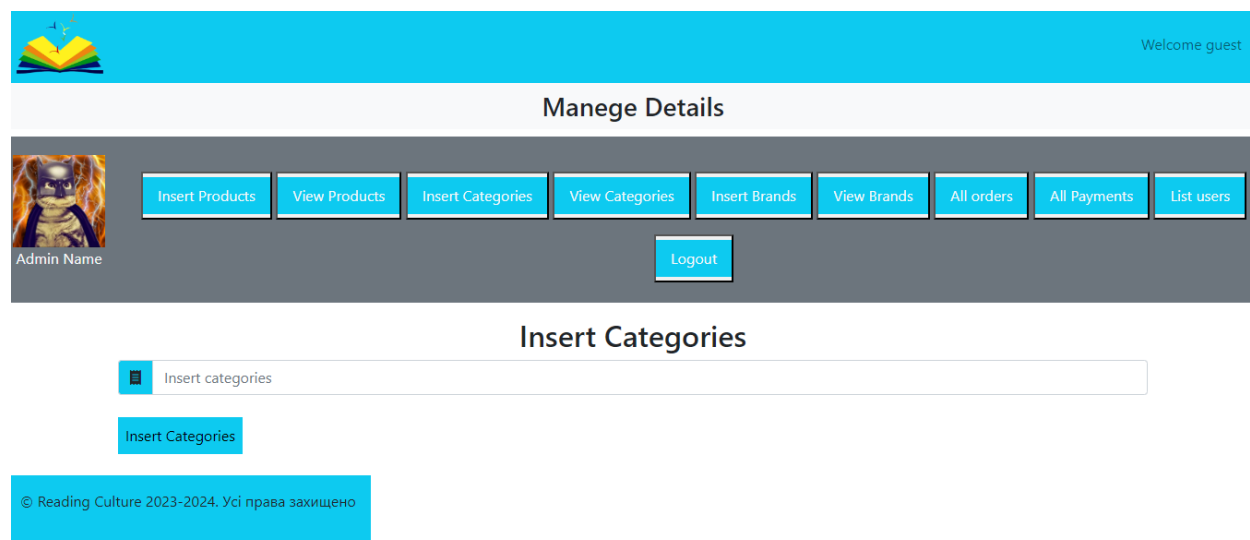


Рисунок 6 - Додавання нової категорії

Ця робота має важливе практичне значення для розробників веб-додатків та підприємств, що прагнуть покращити свою присутність у сегменті електронної комерції, особливо у контексті зростання популярності онлайн-торгівлі та веб-технологій.

ПЕРЕЛІК ПОСИЛАНЬ:

1. Цеслів О.В. WEB-програмування : навч. посібник / О.В. Цеслів ; М-во освіти і науки, молоді та спорту України, Нац. техн. ун-т України "Київ. політехн. ін-т". Київ : НТУУ "КПІ", 2011. 296, с. .
2. Куленко М.Я. Основи графічного дизайну : підручник для студентів вищих навч. закладів / Михайло Куленко; МОНУ; Київський нац. ун-т будівництва і архітектури. 2-ге вид., випр. та доп. Київ : Кондор, 2007. 492с.

Зярко Д.В., студент кафедри програмних засобів і технологій, Херсонський національний технічний університет, м.Хмельницький, Україна,

Захарченко Р.М., к.т.н., доцент кафедри програмних засобів і технологій, Херсонський національний технічний університет, м.Хмельницький, Україна.

ВИКОРИСТАННЯ ІТ В РОЗРОБЦІ ВЕБ-ДОДАТКУ ДЛЯ ТУРИСТИЧНОГО АГЕНСТВА

У сучасному світі інформаційних технологій туристичні агентства стикаються з необхідністю постійного оновлення своїх сервісів для забезпечення зручності та задоволення потреб клієнтів. Із швидким розвитком онлайн-технологій та зростанням популярності онлайн-бронювань та пошуку інформації, туристичні агентства примушені адаптуватися до цих змін, надаючи своїм клієнтам зручні та інноваційні веб-додатки [1-4]. Отже, розробка веб-додатку для туристичного агентства стає актуальною задачею, що вимагає проведення досліджень та впровадження нових технологічних рішень для підвищення конкурентоспроможності на ринку туризму. Цілі дослідження даного напрямку визначаються наступним чином:

- аналіз сучасного стану ринку туристичних послуг та веб-додатків для туризму з метою ідентифікації актуальних тенденцій та потреб користувачів;
- визначення основних вимог та потреб користувачів щодо функціональності веб-додатку та його інтерфейсу;
- розробка концепції та функціональних вимог до веб-додатку на основі зібраної інформації;
- реалізація веб-додатку з використанням сучасних технологій веб-розробки та кращих практик;
- проведення тестування та оцінка ефективності розробленого додатку згідно з встановленими критеріями.

Новизна роботи за темою розробки зручного та інноваційного програмного продукту полягатиме в розробці веб-додатку, спрямованого на полегшення процесу планування та бронювання подорожей для клієнтів туристичного агентства. Результати дослідження та розробки мають значення як для практичного застосування в туристичній індустрії, так і для подальших досліджень у сфері розробки веб-додатків для туризму. Враховуючи швидкі темпи розвитку технологій та зміну уявлень користувачів про зручність та ефективність веб-сервісів, пошук та реалізація новаторських рішень стає запорукою успіху та конкурентоспроможності на ринку туризму.

На рис.1 – 6 представлено екранні форми розробленого веб-додатку для туристичного агентства

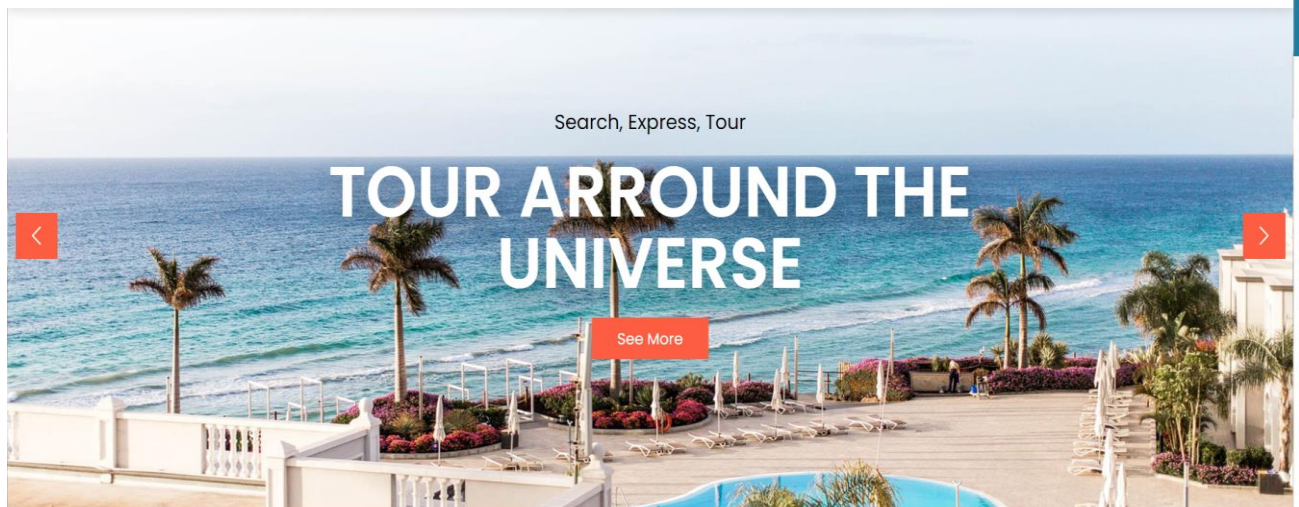
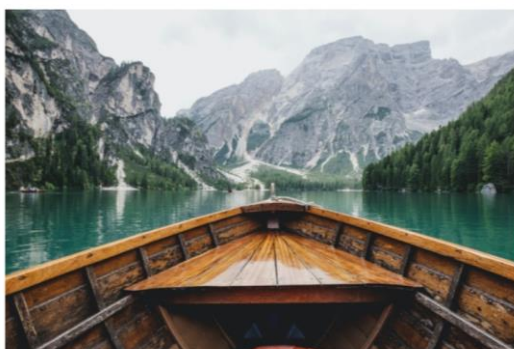


Рисунок 1 - Головний банер

OUR SERVICES



Рисунок 2 - Сервіси



About Us

Amazing Tours A Best Tour Operator And Travel Agent In Bangladesh. We Are Tour Package, Hotel Booking, Bangladesh Tours & Travel Agent. Find International And Domestic Tour Packages From Bangladesh At Low Prices Including Best ... Travel Package | Honeymoon Package | Tour Package

[Read More](#)

Рисунок 3 - Інформація про компанію

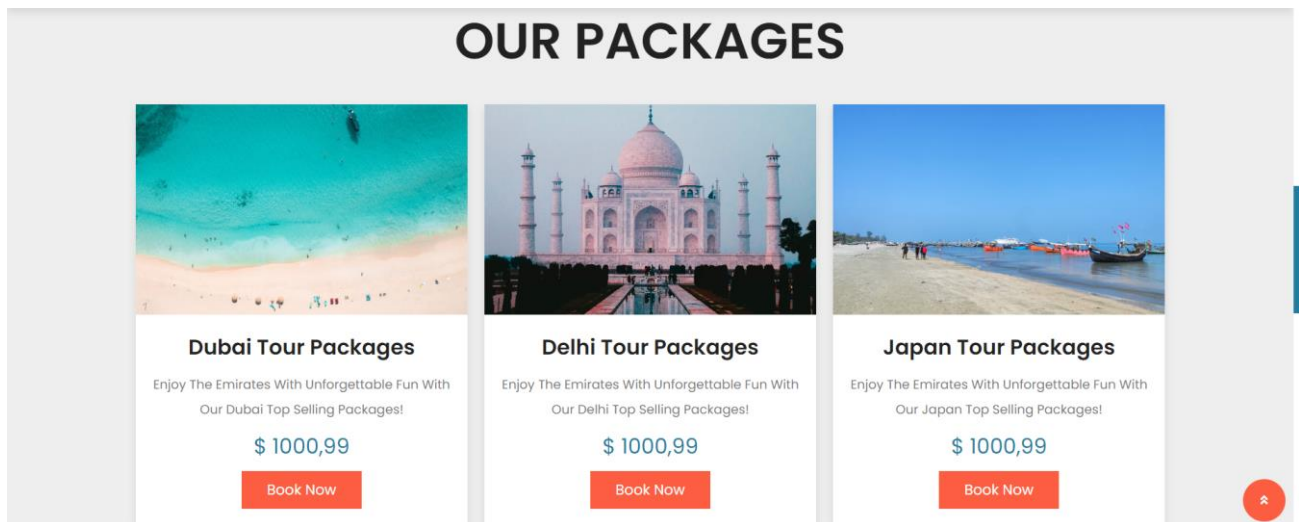


Рисунок 4 - Тури на головній сторінці

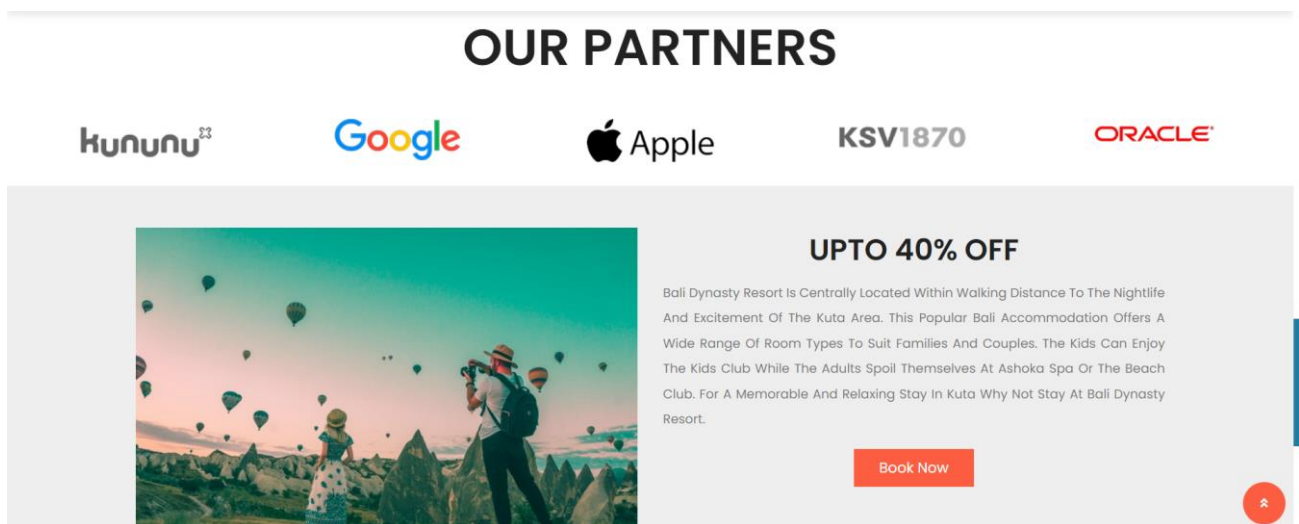


Рисунок 5 - Наші партнери і особлива пропозиція для користувача

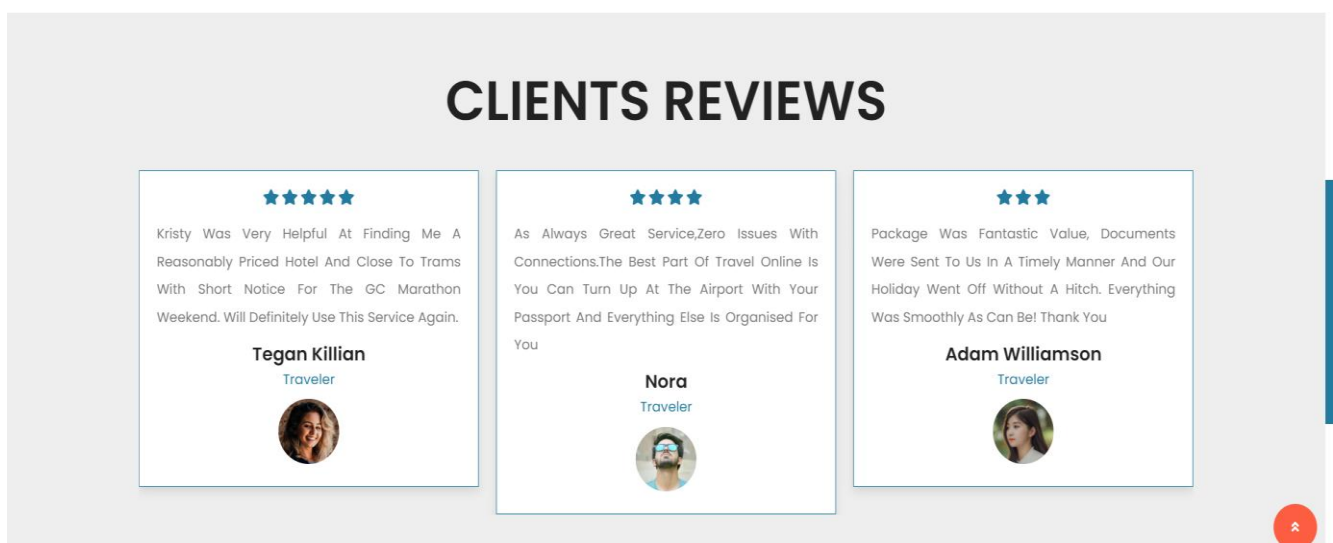


Рисунок 6 - Відгуки

Перед розробкою веб-додатку для туристичного агентства був проведений аналіз та вибір оптимальних технологій та інструментів для розробки фронтенду та бекенду, а також для вибору бази даних. Після цього була розроблена структура бази даних, враховуючи потреби програмного продукту.

Процес реалізації додатку це реалізація головної сторінки, системи замовлення, сторінки «Про нас» та прописані необхідні скріпти. Розроблено керівництво з використання додатку, що дозволить користувачам ефективно взаємодіяти з програмним продуктом.

Результатом роботи стала успішна реалізація веб-додатку для туристичного агентства, який відповідає всім необхідним вимогам та сприятиме покращенню ефективності та конкурентоспроможності агентства.

ПЕРЕЛІК ПОСИЛАНЬ:

1. Зацепина, Т. В. Сучасні тенденції розвитку туризму в Україні. URL: https://tourlib.net/statti_ukr/zacepina.htm
2. Садова О. В. Проблеми та перспективи розвитку туризму в Україні. URL: https://tourlib.net/statti_ukr/sadova.htm
3. Аспекти розвитку туристичного бізнесу в Україні. URL: <http://dspace.wunu.edu.ua/bitstream/316497/20989/1/211-212.pdf>
4. Visit Ukraine Today: The Biggest Travel Trends for 2024. URL: <https://visitukraine.today/uk/blog/3159/the-biggest-travel-trends-for-2024>

Іваниця М.С., студент кафедри програмних засобів і технологій, Херсонський національний технічний університет, м.Хмельницький, Україна,

Хохлов В.А., к.т.н., доцент кафедри програмних засобів і технологій, Херсонський національний технічний університет, м.Хмельницький, Україна.

БАЗОВІ ТА СУЧАСНІ ТЕОРІЇ МОТИВАЦІЇ ТА МОТИВАЦІЙНОГО СПОНУКАННЯ В ІТ – СФЕРІ

Відомо, що мотивувати робітників не можливо без знання їх потреб, а особливо без врахування ієрархії цих потреб. Найпоширеним видом мотивації є фінансовий вид мотивації. Але керівники в ІТ-сфері замислюються не тільки над грошовою винагородою, а намагаються враховувати нематеріальні види мотивації персоналу також [1]. Існуючі матеріальні способи мотивації ІТ-фахівців: конкурентоспроможна і стабільна заробітна плата; різні бонуси; регулярні підвищення заробітної платні; заохочення особистих ініціатив співробітника та понаднормової роботи; знижки або безкоштовне користування продуктами діяльності компанії (програмне забезпечення, бібліотека тощо) [1-3]. До нематеріальних видів мотивації ІТ-фахівців відносять: оплату занять спортом, різних навчальних курсів, необхідних для роботи; оплачувану відпустку; позапланові вихідні; корпоративні курси з іноземної мови; оплату участі в спеціалізованих конференціях, «meet-up»; фінансування закордонних робочих поїздок; «соціальні прогладжування» (публічна похвала робітника, реалізація ідей співробітника, публічне визнання його авторства, кар'єрне зростання, збільшення потоку цікавих, креативних та відповідальних завдань, поліпшення рівня комфорту робочого місця та офісу загалом, розважальні заходи, корпоративні свята, тимбилдинги, колективне привітання співробітників з днем народження, допомога та підтримка в прийнятті рішень, регулярний зворотній зв'язок від керівництва з підкресленням досягнень співробітника та пропозиціями щодо допомоги і т.д.). Але в ІТ-сфері до персоналу застосовуються і різні штрафи [1-3].

Винагороди стимулюють тенденцію прагнення до успіху, а штрафи – поведінку, спрямовану на уникнення можливої невдачі.

У теорії управління мотивацію розглядають як процес спонукання, стимулювання себе чи інших (окремої людини або групи людей) до цілеспрямованої поведінки або виконання певних дій, спрямованих на досягнення власної мети або мети організації. Сутність мотивації можна визначити як сили, що змушують людей поводитися певним чином.

Сучасний підхід до класифікації теорії мотивації полягає у виділенні двох великих груп теорії мотивації - змістовних та процесних [1-3].

Процесні теорії побудовані на поведінці людей з урахуванням їх сприйняття та пізнання.

Ці дві групи теорій взаємодоповнюють одна одну, в той же час вони можуть мати різні площини застосування.

Змістова теорія мотивації - теорія, яку розробив А. Маслоу (1908-1970). А. Маслоу запропонував психологічний механізм динаміки розвитку мотивації поведінки людей через задоволення потреб [1-3].

Менше відома теорія мотивації М. Туган-Барановського.

Концепція мотивації за потребами представлена теорією, розробленою Д. Мак-Клелландом. Ця теорія концентрує увагу переважно на потребах вищого рівня, до яких віднесено потреби успіху, влади та причетності [1-3].

Німецький вчений К. Альдерфер зробив спробу модифікувати теорію потреб А. Маслоу. Він виділив три основні класи мотивів, запропонувавши теорію ERG.

Однією із змістовних теорій, що побудована на описі подій та ситуацій, які викликають у людини задоволення в процесі роботи є теорія, автор якої Ф. Герцберг. Результатом дослідження Ф.Герцберга стало виявлення «критичних подій» щодо приємних та неприємних ситуацій [1-3].

Отже, до факторів задоволення роботою віднесені характеристики, які стосуються самої роботи, тобто відображають внутрішній зміст роботи, а до факторів незадоволеності - зовнішні по відношенню до роботи. За Ф. Герцбергом, зміна зовнішніх факторів може ліквідувати те, що викликає незадоволення нею, але не може забезпечити задоволення працею, більш високого почуття.

Теорія професора Мічиганського університету Дугласа Мак Грегора ґрунтується на характеристиці явищ, ситуацій, умов, які пов'язані із забезпеченням задоволення [1-3].

У однієї і тієї ж людини не може бути одна й та ж сама мотивація до ефективної праці. Зміни в житті змінюють особистість та пов'язану з нею мотивацію.

Науковці довели, що найбільш потужним мотивуючим засобом є безпосередньо робота, саме вона мотивує трудову поведінку людини. Чим більшим знанням володіє людина, тим більше її прагнення до цікавої змістовної роботи, роботи, в якій людина може само актуалізуватися [1-3].

В ІТ-сфері керівники приділяють особливу увагу мотиваційного спонукання своїх працівників. Адже успіх підприємства залежить в першу чергу від висококваліфікованих та мотивованих айтишників.

ПЕРЕЛІК ПОСИЛАНЬ:

1. Мотивація персоналу в ІТ-сфері. URL: http://bses.in.ua/journals/2018/35_1_2018/14.pdf
2. Карнегі Д. Як здобувати друзів і впливати на людей. Київ: КМ-Букс, 2017. 256 с.
3. Цимбалюк С.О. Детермінанти трудової активності персоналу в системі мотивації. Актуальні проблеми економіки. 2017. № 1 С. 264–272. URL: http://nbuv.gov.ua/UJRN/ape_2017_1_31.

Іваниця М.С., студент кафедри програмних засобів і технологій, Херсонський національний технічний університет, м.Хмельницький, Україна,

Хохлов В.А., к.т.н., доцент кафедри програмних засобів і технологій, Херсонський національний технічний університет, м.Хмельницький, Україна.

РОЗРОБКА ТА ОПТИМІЗАЦІЯ СИСТЕМИ РЕКОМЕНДАЦІЙ НА ОСНОВІ МАШИННОГО НАВЧАННЯ

Сучасний цифровий світ характеризується стрімким зростанням обсягів інформації, доступної для користувачів. У таких умовах важливим стає не лише збереження та обробка даних, але й надання релевантної інформації, що відповідає індивідуальним потребам. Рекомендаційні системи стали невід'ємною складовою багатьох онлайн-платформ, зокрема стрімінгових сервісів, інтернет-магазинів, соціальних мереж та освітніх ресурсів. Вони дозволяють підвищити зручність користувацького досвіду, сприяють утриманню клієнтів та збільшенню їхньої активності [1-4].

На сьогоднішній день технології машинного навчання відкривають нові горизонти у створенні рекомендаційних систем, забезпечуючи їхню високу точність, адаптивність та масштабованість. Завдяки використанню таких підходів, як колаборативна фільтрація, контентні рекомендації та гібридні моделі, можливо не лише прогнозувати інтереси користувачів, але й динамічно реагувати на зміну їхніх вподобань [1-4].

Актуальність дослідження теми зумовлена зростанням потреби у високоефективних системах рекомендацій, що відповідають сучасним вимогам до точності, швидкості та персоналізації. Зокрема, у контексті стрімінгових платформ, таких як YouTube Music або Netflix, розробка й оптимізація рекомендаційних систем є критично важливим завданням для покращення користувацького досвіду та підвищення конкурентоспроможності.

В результаті проведеного дослідження реалізовано ефективну рекомендаційну систему для потокового сервісу, яка забезпечує персоналізовані пропозиції користувачам на основі їхніх вподобань та історії взаємодій. Система побудована з використанням сучасних методів машинного навчання, включаючи контентну, колаборативну та гібридну фільтрацію, що дозволяє забезпечити високу точність і адаптивність рекомендацій. Проведені експериментальні дослідження підтвердили ефективність розроблених алгоритмів і їхню здатність працювати з великими обсягами даних, надаючи рекомендації в режимі реального часу.

Розроблена архітектура системи підтримує масштабованість і інтегрується з сучасними базами даних для зберігання структурованої інформації про користувачів, контент та їхні взаємодії. Проведена оптимізація обчислень дозволяє системі забезпечувати високу продуктивність навіть при значному навантаженні. Під час тестування було досягнуто високих показників точності рекомендацій, які відповідають сучасним стандартам у галузі.

Таким чином, результати дослідження підтверджують ефективність обраних підходів і демонструють перспективи подальшого розвитку рекомендаційних систем для застосування у стрімінгових сервісах.

ПЕРЕЛІК ПОСИЛАНЬ:

1. Годовиченко М. А., Горбатенко А. А. Рекомендаційні системи: моделі, виклики та можливості. Herald of Advanced Information Technology, 2022, т. 5, № 1, с. 15–25.
2. Димченко О. В., Смиш О. Р., Жежерун О. П. Графічний інтерфейс для рекомендаційної системи. Наукові записки НаУКМА, 2021, т. 4, с. 93–97.

3. Жежерун О. П., Яремко О. В. Побудова рекомендаційних систем на основі онтологій. Наукові записки НаУКМА, 2017, т. 194, с. 45–50.
4. Zhang, Y., Yang, Q. (2021). A survey on deep learning for recommender systems. IEEE Transactions on Knowledge and Data Engineering, 34(2), 1005–1020.

Комісаров О.С., старший викладач кафедри програмних засобів і технологій, Херсонський національний технічний університет, м.Хмельницький, Україна,
Захарченко Р.М., к.т.н., доцент кафедри програмних засобів і технологій, Херсонський національний технічний університет, м. Хмельницький, Україна.

ОПТИМІЗАЦІЯ АЛГОРИТМІВ ПОШУКУ ТА РОЗВІДКИ ДЛЯ РОЮ АВТОНОМНИХ АГЕНТІВ НА ОСНОВІ МЕТОДІВ МАШИННОГО НАВЧАННЯ

Використання роїв автономних агентів для вирішення задач пошуку та розвідки стає все більш актуальним у різних сферах застосування. Інтеграція методів машинного навчання в алгоритми керування роєм відкриває нові можливості для підвищення ефективності та адаптивності таких систем [1]. Актуальність дослідження обумовлена необхідністю розробки більш ефективних методів координації та прийняття рішень у групах автономних агентів.

У роботі представлено комплексний підхід до оптимізації поведінки рою з використанням глибокого навчання з підкріпленням та еволюційних алгоритмів. Архітектура розробленої системи керування роєм зображена на рис. 1.

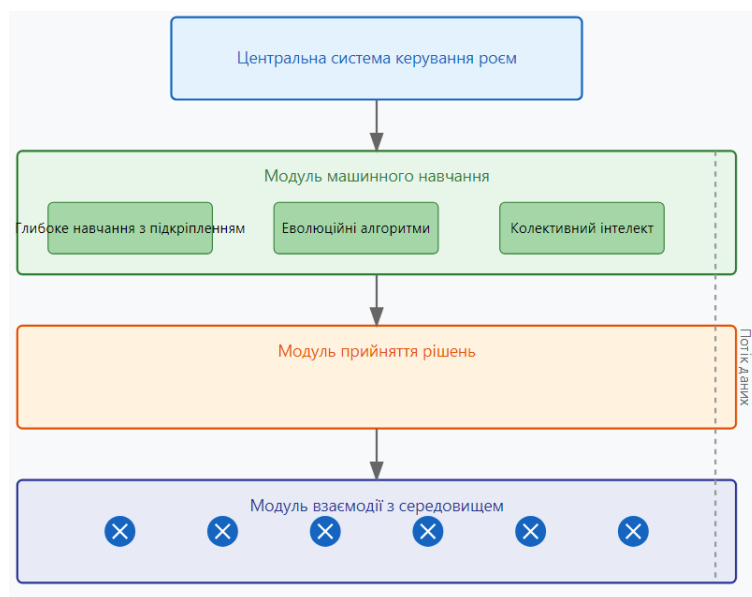


Рисунок 1 - Архітектура системи керування роєм

Дослідження включало розробку та порівняльний аналіз різних підходів до навчання та оптимізації поведінки рою. Основні характеристики розглянутих методів представлені в таблиці 1.

Таблиця 1.

Порівняльний аналіз методів оптимізації поведінки рою

Метод	Швидкість навчання (год)	Ефективність пошуку (%)	Адаптивність (1-10)	Обчислювальна складність	Енергоефективність (1-10)
Q-Learning	4-6	82	7	$O(n^2)$	8
DQN	8-12	89	8	$O(n^2 \log n)$	6
A3C	6-8	91	9	$O(n^3)$	5

DDPG	10-14	94	9	$O(n^3)$	4
Genetic Algorithm	12-16	87	6	$O(n \log n)$	7
PSO	3-5	85	7	$O(n)$	9
Hybrid Approach	8-10	96	9	$O(n^2)$	7

Розроблений гібридний підхід поєднує переваги глибокого навчання з підкріпленням та еволюційних алгоритмів [2]. Кожен агент рою оснащений нейронною мережею, що навчається за допомогою алгоритму DDPG (Deep Deterministic Policy Gradient), в той час як параметри колективної поведінки оптимізуються за допомогою генетичного алгоритму.

Основними компонентами системи є модуль сприйняття навколишнього середовища, модуль прийняття рішень та модуль координації дій агентів. Модуль сприйняття використовує комбінацію сенсорних даних та алгоритмів комп'ютерного зору для створення карти середовища [3]. Модуль прийняття рішень базується на глибокій нейронній мережі, що оптимізує стратегію пошуку на основі отриманого досвіду.

Експериментальне дослідження проводилось на симуляторі з різними сценаріями пошуку та розвідки. Тестування включало варіації кількості агентів (від 5 до 50), розмірів області пошуку та типів перешкод. Результати показали значне покращення ефективності пошуку порівняно з традиційними методами [4].

Важливим аспектом дослідження стала розробка механізмів адаптації поведінки рою до змін у навколишньому середовищі. Впроваджено алгоритм динамічної реконфігурації рою, що дозволяє оптимізувати розподіл агентів у просторі залежно від поточної задачі та умов середовища [5].

Особлива увага приділена питанням енергоефективності та оптимізації ресурсів. Розроблено алгоритм балансування навантаження між агентами, що враховує залишковий заряд батарей та обчислювальні можливості кожного агента. Система демонструє здатність підтримувати ефективну роботу навіть при виході з ладу окремих агентів.

Практичне застосування розробленої системи було протестовано в задачах:

- Пошуку та моніторингу об'єктів у важкодоступних місцях
- Картографування невідомої території
- Виявлення аномалій та небезпечних зон
- Координації дій у надзвичайних ситуаціях

Результати тестування показали підвищення ефективності виконання поставлених задач на 35-40% порівняно з існуючими методами. Час, необхідний для повного обстеження заданої території, скоротився в середньому на 45%, а точність виявлення цільових об'єктів збільшилась на 28%.

Запропонований підхід до оптимізації поведінки рою автономних агентів демонструє високу ефективність у вирішенні задач пошуку та розвідки. Інтеграція методів машинного навчання дозволяє значно покращити адаптивність та автономність системи. Розроблені алгоритми можуть бути застосовані в широкому спектрі практичних задач, де потрібна координувана робота групи автономних агентів.

ПЕРЕЛІК ПОСИЛАНЬ:

1. Smith J., Wang R. Reinforcement Learning in Swarm Robotics. IEEE Transactions on Robotics. 2023. Vol. 39(4). P. 789-804.
2. Chen X., Miller T. Hybrid Optimization for Multi-Agent Systems. Journal of Artificial Intelligence Research. 2024. Vol. 75. P. 234-256.
3. Zhang L., Anderson K. Vision-Based Navigation in Robotic Swarms. Robotics and Autonomous Systems. 2023. Vol. 158. P. 104207.

4. Brown M., Davis R. Adaptive Behavior in Autonomous Agent Systems. IEEE Transactions on Systems, Man, and Cybernetics. 2024. Vol. 54(2). P. 312-328.
5. Wilson P., Thompson J. Energy-Efficient Swarm Control Algorithms. Journal of Intelligent & Robotic Systems. 2023. Vol. 102. P. 45-67.

Леба Д. С., студент кафедри програмних засобів і технологій, Херсонський національний технічний університет, м. Хмельницький, Україна,

Огнєва О.Є., к.т.н., доцент кафедри програмних засобів і технологій, Херсонський національний технічний університет, м. Хмельницький, Україна.

РОЗРОБКА БАЗИ ДАНИХ ВЕБ-ДОДАТКУ "НАЗЕМНА БРОНЕТЕХНІКА"

Зростання кількості специфічних проєктів з власними розробками є причиною зростання попиту на створення списку з фільтрами. Але таку базу даних можна розвивати в залежності від задач [1-3].

Мета дослідження – створення реляційної бази даних для веб-додатку «Наземна бронетехніка». За допомогою такої інформаційної системи надається можливість автоматизувати списки розробленої техніки, присвоєння потенційних операторів, розробників. За допомогою неї розробник моделей бронетехніки зможе заносити нову інформацію, скоротити час та автоматично фільтрувати дані.

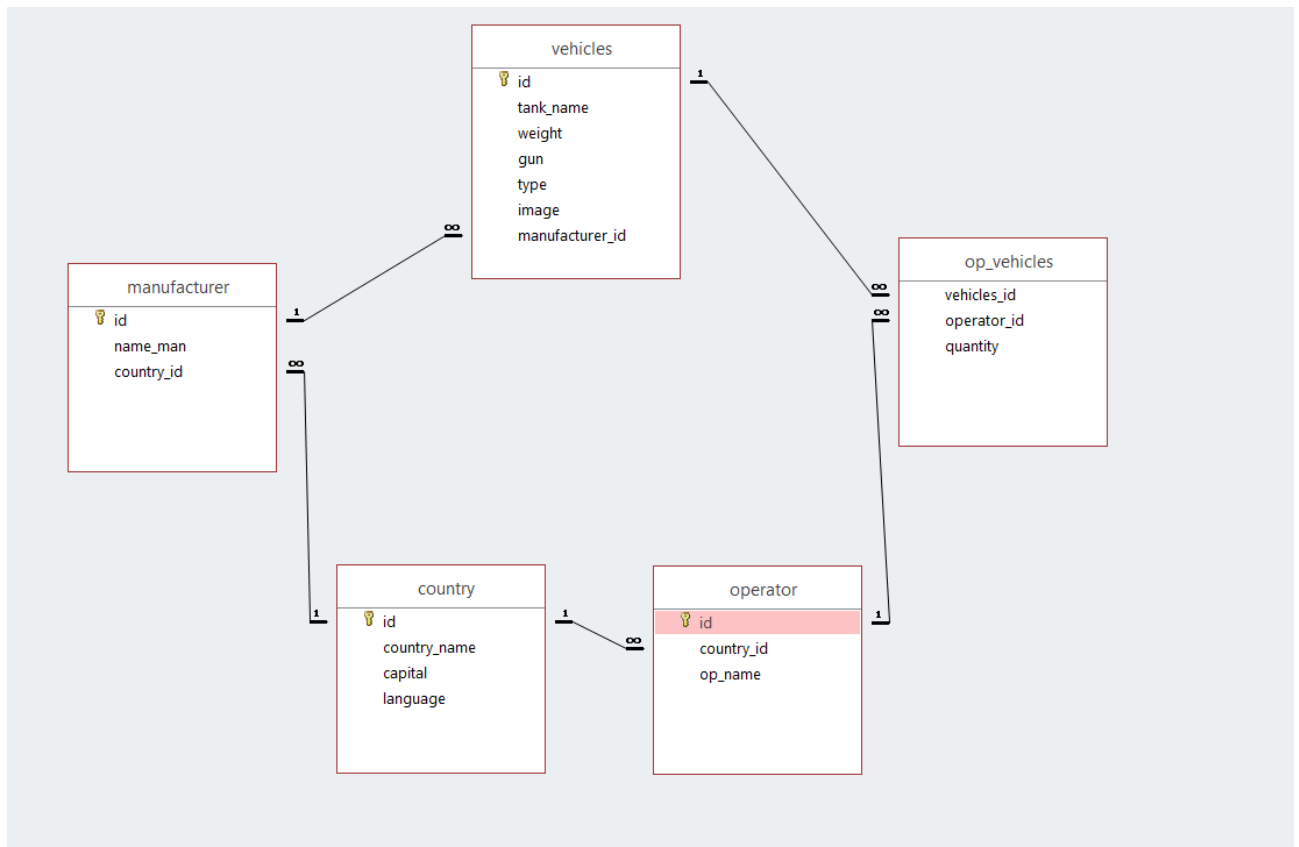


Рисунок 1 - Схема реалізованої бази даних веб-додатку.

Базу даних створено за допомогою phpMyAdmin. Ця СУБД забезпечує зручний інтерфейс для управління базою даних через веб-браузер, що спрощує створення, редагування та видалення даних [2]. Також було розроблено інтерфейс для роботи з базою даних, що дозволяє користувачам легко взаємодіяти з даними та фільтром.

Бронетехніка

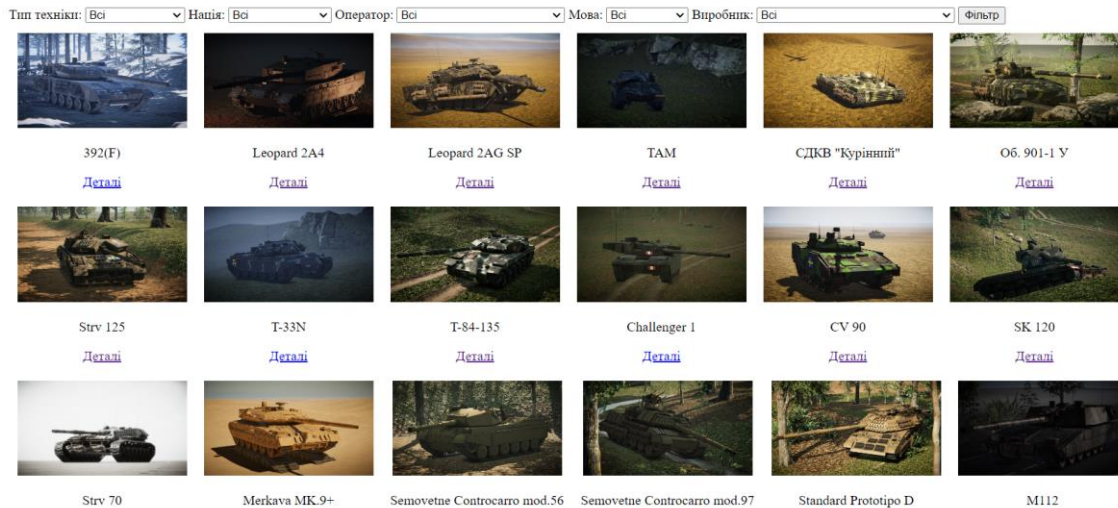


Рисунок 2 - Зовнішній вигляд веб-додатку

Розробка цієї бази даних робить можливим розвиток в залежності від задач. Ось деякі приклади:

- Сайт з розіграшами моделей бронетехніки. Для цього треба як мінімум створити додатково бази даних для акаунтів, розробити систему розіграшу та зв'язати з основною базою даних. Така тема є досить актуальна в наш час, оскільки моделісти організовують подібні розіграші, передаючи зібрані гроші на благодійну діяльність.
- БД для ігрового додатку. Це скоріше буде вікі-подібна сторінка для ігрової платформи, де буде детальна інформація про кожну техніку з усіма відомими ігровими тактико-технічними характеристиками. Є корисним в рамках ігрової платформи
- БД для сайту волонтерської організації. В реалізацію потрібно зробити новинну сторінку(+ БД новини), сторінку форуму(+ БД форум) та інші
- Концепція полягає в конвертуванні спау фантазій в вибірку реальних цікавих проєктів, що може спричинити розвиток військових технологій. Тут вже можуть бути не тільки наземна техніка, а й водна й повітряна. В час війни це є дуже актуальним й йде в пару з розвитком технологій. Й саме тому концепція спау ідей є дуже непоганим варіантом, яке можна спробувати.

Таким чином була представлена база даних із специфічними задачами, але із великими перспективами.

ПЕРЕЛІК ПОСИЛАНЬ:

1. Базові структури: класи та схеми. Об'єкти/речі, змодельовані класами UML. URL: <https://web.csulb.edu/colleges/coe/cecs/dbdesign/dbdesign.php?page=class.php>
2. Документація PHPMyAdmin. URL: <https://docs.phpmyadmin.net/en/latest/>
3. Концепції реляційної бази даних. Розділ 2 Терміни та поняття бази даних. URL: <https://www.upi.pr.it/docs/easfg/easvrfp7.htm>

Любимов О.С., аспірант кафедри програмних засобів і технологій, Херсонський національний технічний університет, м. Хмельницький, Україна,

ВДОСКОНАЛЕННЯ МОДЕЛІ РОЗПІЗНАВАННЯ ОСЕРЕДКІВ ЛІСОВИХ ПОЖЕЖ ПРИ ДИСТАНЦІЙНОМУ ЗОНДУВАННЮ З БЕЗПІЛОТНИХ АПАРАТІВ

Суттєве прискорення кліматичних змін, посилення урбанізації, індустріалізація та багато інших негативних наслідків діяльності людини стали викликом для природи, і чим більш інтенсивними та руйнівними вони стають, тим більше людство стикається з проблемами. Наразі світ перевантажений стихійними лихами - масштабними повеннями, лісовими пожежами, раптовими зсувами ґрунтів, землетрусами, цунамі, торнадо та іншими природними явищами, що утворюють ланцюги небажаних подій і викликають катастрофи через вплив низки людських факторів, в т.ч. неухважність та недбалість [1].

Оскільки людство ще не адаптувалося до природних катастроф та не створило адекватних методів прогнозування та запобігання стихійним лихам, то займається переважно локалізацією та ліквідацією їх наслідків. Більшість природних явищ є випадковими, нестаціонарними, нелінійними та непередбачуваними процесами, які швидко розвиваються у просторі та часі та, як правило, характеризуються слабкою спостережуваністю [2]. Такі природні явища складають клас швидкоплинних динамічних просторово-розподілених процесів руйнівного характеру.

На сьогоднішній день не існує способів побудови надійних, адекватних або навіть прийнятних математичних моделей швидкоплинних руйнівних процесів (ШРП) для їх прогнозування та попередження. Слабка спостережуваність ШРП, їх значні масштаби, інтенсивність та швидкість розповсюдження ускладнюють умови вирішення задач локалізації, реагування та ліквідації наслідків природних катастроф, які ґрунтуються переважно на візуальних спостереженнях, що є неповними і неточними. Отже, проблема інформаційної підтримки вирішення задач локалізації, реагування та ліквідації наслідків ШРП є надзвичайно актуальною.

Але ж, крім нових викликів з'явилися і нові можливості, пов'язані з розвитком безпілотних літальних апаратів (БПЛА), технологій дистанційного зондування, обробки зображень, машинного навчання тощо. Досягнутий рівень розвитку дає можливість використовувати БПЛА для успішного вирішення задач спостереження в умовах, небезпечних для здоров'я і навіть життя спостерігача. БПЛА можуть спостерігати за розвитком ШРП «зверху» та «збоку», що дозволяє максимально наблизитися до спостережуваних подій та утримуватись на безпечній відстані від них [3].

Можливість спостереження ШРП за допомогою БПЛА «зверху» та «збоку» дозволяє значно ширше охоплювати просторово-розподілені події, досягаючи принципово іншого рівня ситуаційної обізнаності осіб, що приймають рішення щодо локалізації, реагування та ліквідації наслідків, на відміну від візуального спостереження. Оскільки ефективність виявлення, локалізації та моніторингу ШРП найбільше залежить від обізнаності осіб, які приймають рішення, щодо динаміки процесів, застосування методів дистанційного зондування за допомогою БПЛА у системах моніторингу ШРП є перспективним напрямком досліджень.

Розглянемо питання застосування дистанційного зондування з БПЛА для моніторингу лісових пожеж – найбільш поширеного та небезпечного класу стихійних явищ, що мають всі ознаки ШРП. Якщо ЛП вже сталася, задача її моніторингу полягає у спостереженні за її розповсюдженням, відтворенні моделі розповсюдження фронту вогню, прогнозування його динаміки.

Просторово-часова модель ШРП має бути тривимірною, що спонукає вирішувати задачу моніторингу одразу з декількох точок спостереження, одночасно і синхронно застосовуючи групу БПЛА, що дозволяє оцінювати в реальному часі висоту фронту полум'я під час лісової пожежі (ЛП) або глибину зсуву ґрунту тощо.

Однією з таких сфер застосування груп БПЛА є системи моніторингу ЛП [5]. БПЛА можуть використовувати оптичні, інфрачервоні камери, лідари та інші сучасні датчики для спостереження ЛП в режимі реального часу, і, на відміну від людей, можуть літати досить близько до вогню. Однак вогонь, дим і вітер спотворюють картину поширення ЛП, а мінливість погоди, палива і рельєфу місцевості ускладнює прогнозування напрямку і швидкості поширення ЛП, що заважає отримати достовірну інформацію для планування та проведення операції з ліквідації ЛП. Свій вплив спричиняють також турбулентність і вібрації БПЛА. Крім того, існують певні просторові обмеження, такі як множинність осередків ЛП, сегментація фронту ЛП, мінливість палива та рельєфу місцевості.

Тож під впливом всіх цих чинників при одночасному використанні для спостереження декількох БПЛА, їхні спостереження є неповними і неточними, але важливою перевагою їх використання є те, що вони можуть спостерігати за рухомим процесом, тобто змінювати точку зору на ситуацію, і можуть наблизитися до певної цільової точки набагато ближче, ніж людина.

Оскільки подолати невизначеність спостережень, не наближаючись до небезпечних подій ближче, ніж дозволяють міркування безпеки, неможливо, особливості просторово-розподілених ШРП не дозволяють повноцінно спостерігати процес з однієї точки спостереження навіть у разі безперервного руху БПЛА, тому розв'язання задачі відтворення моделі процесу потребує одночасного спостереження з кількох різних точок [6]. При використанні декількох БПЛА повнота спостережень істотно додається, але виникає проблема їх несумісності та суперечливості.

Відомо, що особи, які приймають рішення, повинні мати ситуаційну інформацію для планування операцій реагування, тобто вони повинні розуміти розвиток процесу, оцінювати напрямки та швидкість поширення ЛП. На щастя, це не потребує занадто детальної інформації, але має надходити своєчасна інформація про висоту, ширину, силу, напрямки та швидкість фронту вогню. Крім того, суттєвий обмежуючий вплив має фактор часу: чим швидше буде визначена динаміка процесу і розташування фронту пожежі, тим швидше особа, яка приймає рішення, зможе спланувати операцію реагування та придушення ЛП.

Моделлю ЛП як певного класу ШРП є модель розповсюдження фронту вогню [7]. Отже, для побудови тривимірної просторової моделі поширення фронту пожежі в динаміці необхідно адекватно врахувати всі фактори невизначеності спостережень, включно з їх неповнотою, неточністю і суперечливістю. Аналіз підходів до коректного подання багаторакурсних спостережень з кількох БПЛА з врахуванням факторів невизначеності під час моделювання ШРП з використанням сучасних технологій дистанційного зондування та обробки зображень є предметом нашого дослідження.

Прийmemo за основу, що у кожному процесі завжди є певна детермінанта, яка відображає його динаміку. Наприклад, у процесі ЛП таким визначальником є фронт пожежі, який формується під впливом високотемпературної реакції горіння і переміщується, залишаючи вже вигорілі ділянки, де не залишилося палива для реакції горіння, і охоплюючи нові території, де є паливо [8]. Чим сильніший вогонь, тим вищий і ширший фронт пожежі. Таким чином, реконструкція процесу ЛП – це тривимірне відтворення фронту пожежі за даними спостережень її ознак в певних точках простору, а саме вогню, полум'я, диму, високої температури. Наявність перших трьох ознак можна розпізнати за допомогою оптичних камер, а четвертого – за допомогою інфрачервоних камер. У будь-якому випадку необхідно використовувати сучасні методи комп'ютерного зору та дистанційного зондування для оцінки поверхні (контуру) пожежі, що фіксується датчиками. Це завдання досить схоже на завдання розпізнавання об'єктів. Тривимірну модель об'єкта можна відтворити шляхом послідовного сканування його поверхні з декількох точок спостереження. Детальний огляд методів розпізнавання об'єктів наведено в [9], але в ній об'єкти розглядають як статичні, в той час як ЛП – динамічний процес. Таким чином, відомі методи розпізнавання об'єктів є недостатніми у контексті вирішуваної задачі [10], однак вони забезпечують можливість послідовного розпізнавання положення та стану фронту пожежі з отриманням моделі його руху, яка

відображає процес поширення ЛП. Для подолання обмежень видимості при спостереженні було запропоновано метод багаторакурної реконструкції об'єкта [11], що дозволяє сканувати об'єкти в реальному часі, але є досить чутливим до непевних спостережень.

Найбільш релевантним до вирішуваної задачі є об'ємний підхід [12], де розглядаються як реальні 3D-об'ємні методи, які описують об'єкти за допомогою 3D-хмар точок, так і 2,5D-методи, які описують лише виміряну висоту (або глибину) для певної 2D-області. І 3D-, і 2,5D-методи є надчутливими до шумів сенсорів і не можуть розрізняти підозрілі ділянки [13]. Очевидно, що полум'я є певною суттєвою ознакою фронту вогню, яка має змінну форму та колір. Крім того, вплив диму, вітру, відблисків і мерехтіння істотно ускладнюють розпізнавання ознак полум'я.

Таким чином, різного роду спотворення та перешкоди призводять до появи множини нерозпізнаних і невизначених елементів у сканованому зображенні, тому потрібно оцінити деякі ознаки пожежі (полум'я, дим тощо) у певних областях за ступенем їх достовірності на основі невизначеності. спостереження. Об'ємний 3D-метод, запропонований у [6], використовує м'яку наближену модель невизначеності для опису фронту пожежі хмарою розмитих 3D-комірок. Інший підхід є імовірнісним, що подає невизначеність у термінах шансів [14], але є релевантним лише стохастично стабільним даним. Отже, для вирішення поставленої задачі розпізнавання образів більш підходять різні неімовірнісні методи, на кшталт м'яких, нечітких, наближених множин та сірих чисел [15].

Зазвичай реконструкція фронту ЛП виконується в невизначених умовах різного характеру, які неможливо змоделювати окремо неповною, неточною або нечіткою інформацією. Ступені впевненості просторово розподілені та динамічні, тоді як простір спостереження досить розріджений і містить багато невизначених і підозрілих областей. Оскільки нечітка, наближена, м'яка та «сіра» моделі тісно пов'язані між собою та доповнюють одна одну [16], доцільно об'єднати їх у певну комплексну модель, що відображатиме як об'єктивну, так і суб'єктивну невизначеність.

Таким чином, незважаючи на загальний прогрес у галузі реконструкції об'єктів та обробки зображень в умовах невизначеності, багато проблем ще мало відображені в літературі, залишаються недостатньо дослідженими та потребують подальших досліджень. Щоб подолати недоліки, з одного боку, потрібно прискорити процес реконструкції фронту пожежі, уникаючи ітераційних «важких» процесів, а з іншого боку, запобігти впливу спотворень, перешкод і шумів на зорове сприйняття.

Основна ідея вирішення проблеми багаторакурсного дистанційного зондування ЛП з БПЛА полягає в тому, що певні спостережувані ознаки ШРП (у випадку ЛП – полум'я або дим) відповідають певному стану (горить, вигоріло, ще не зайнялося тощо) певної просторової області. Спостерігаючи ці ознаки в межах просторової області, можна її віднести з певним ступенем впевненості до відповідного (за станом) класу просторових областей. Оскільки простір спостереження є розрідженим, а спостереження – неточними і неповними, різні невпевнені, невизначені та підозрілі області також можливо апроксимувати з певним ступенем належності до конкретного класу просторових областей.

Враховуючи одночасний вплив декількох факторів невизначеності різної природи, для визначення ступеня належності деякої просторової області до певного класу просторових областей треба додатково побудувати складну модель, що об'єднуватиме нечіткі, м'які множини та «сірі» числа.

Отже, подання фронту пожежі може базуватися на моделі комплексної невизначеності, яка дозволяє згладити вплив спотворень, перешкод і шумів, а комплексна модель невизначеності може базуватися на одночасному використанні нечітких, наближених, м'яких множин, а також сірих чисел. Це дасть можливість враховувати динаміку поширення фронту пожежі та забезпечує правильну візуальну картину відтвореної 3D-моделі лісової пожежі.

Запропонований підхід дає можливість прискорити процес відтворення тривимірної моделі фронту пожежі та підвищити якість візуалізації для особи, яка приймає рішення, дозволяючи їй зосередитися на спостереженні змодельованого фронту вогню та зменшити

вплив спотворень та перешкод. Розроблений метод забезпечує достатню продуктивність для виконання вимог оперативного реагування на лісові пожежі.

ПЕРЕЛІК ПОСИЛАНЬ:

1. Shen, G., Zhou, L., Wu, Y., Cai, Z.: A global expected risk analysis of fatalities, injuries, and damages by natural disasters. *Sustainability*, 2018, vol. 10(7), p. 2573.
2. Newman, J., Maier, H., Riddell, G., Zecchin, A., Daniell, J., Schaefer, A., van Delden, H., Khazai, B., O'Flaherty, M., Newland, C.: Review of literature on decision support systems for natural hazard risk reduction: current status and future directions. *Env. modeling & software*, 2017, vol. 96(C), pp. 378–409.
3. Yuan, C., Liu, Z., Zhang, Y.: Aerial images-based forest fire detection for firefighting using optical remote sensing techniques and unmanned aerial vehicles. *J. Intel. & Robotic Syst.*, 2017, vol. 88, pp. 635–654.
4. Merino, L., Martínez de Dios, J., Ollero, A.: Cooperative Unmanned Aerial Systems for Fire Detection, Monitoring, and Extinguishing. In: Valavanis K., Vachtsevanos G. (eds). *Handbook of Unmanned Aerial Vehicles*, 2015, pp. 2693–2722.
5. Sherstjuk, V., Zharikova, M., Dorovskaja, I., Sheketa, V.: Assessing Forest Fire Dynamics in UAV-Based Tactical Monitoring System. *Advances in Intelligent Systems and Computing*, 2020, vol. 1246, pp. 285–301.
6. Sherstjuk, V., Zharikova, M., Dorovskaja, I.: 3D Fire Front Reconstruction in UAV-Based Forest-Fire Monitoring System. In: *Proc. of IEEE Third International Conference on Data Stream Mining & Processing (DSMP)*, 2020, pp. 243–248.
7. Sherstjuk, V., Zharikova, M.: Fire-Front Recognition in UAV-Based Forest-Fire Monitoring System Using Fuzzy Rough Soft Sets. In: *Proc. of IEEE 2nd Ukraine Conf. on Electrical and Computer Engineering (UKRCON)*, Lviv, Ukraine, 2019, pp. 1091–1096.
8. Andrews, P.L.: The Rothermel surface fire spread model and associated developments: A comprehensive explanation. *Gen. Tech. Rep. RMRS-GTR-371*. Fort Collins, CO: U.S. Department of Agriculture, Forest Service, Rocky Mountain Research Station. 2018, 121 p.
9. Mendez, O., Hadfield, S., Pugeault, N., Bowden, R.: Taking the Scenic Route to 3D: Optimising Reconstruction from Moving Cameras. In: *2017 IEEE Int. Conf. on Computer Vision (ICCV)*, Venice, 2017, pp. 4687–4695.
10. Scott, W.R., Roth, G., Rivest, J.-F.: View planning for automated three-dimensional object reconstruction and inspection. *ACM Computing Surveys*, 2003, vol. 35(1), pp. 64–96.
11. Galliani, S., Lasinger, K., Schindler, K.: Massively Parallel Multiview Stereopsis by Surface Normal Diffusion. In: *IEEE Int. Conf. on Computer Vision (ICCV)*, Santiago, 2015, pp. 873–881.
12. Isler, S., Sabzevari, R., Delmerico, J., Scaramuzza, D.: An information gain formulation for active volumetric 3D reconstruction. In: *IEEE Int. Conf. on Robotics and Automation (ICRA)*, Stockholm, 2016, pp. 3477–3484.
13. Vasquez-Gomez, J., Sucar, L., Murrieta-Cid, R., Lopez-Damian, E.: Volumetric Next-Best-View Planning for 3D Object Reconstruction with Positioning Error. *Int. J. of Advanced Robotic Syst.*, 2014, vol. 11(10), pp. 1–13.
14. Alon, N., Spencer, J.: *The Probabilistic Method*. 2016. Wiley, New York, 4th. ed.
15. Sherstjuk, V., Zharikova, M., Dorovskaja, I., Chorny, D., Gusev, V., Sokol, I.: 3D Fire Front Reconstruction Based on Multi-View Observation and Complex Uncertainty Model. *Advanced Computer Information Technologies: Proc. of 11th Int. Conf.*, Deggendorf, Germany, 2021, pp. 761–765.
16. Yamaguchi, D., Li, G.-D., Chen, L.-C., Nagai, M.: Reviewing crisp, fuzzy, grey and rough mathematical models. In: *Proc. of IEEE Int. Conf. on Grey Systems and Intelligent Services*, 2007, Nanjing, China, pp. 547–552.

Лященко М.О., студент кафедри програмних засобів і технологій, Херсонський національний технічний університет, м.Хмельницький, Україна,

Захарченко Р.М., к.т.н., доцент кафедри програмних засобів і технологій, Херсонський національний технічний університет, м.Хмельницький, Україна.

ВИКОРИСТАННЯ ІТ ПРИ РОЗРОБЦІ ВЕБ-ДОДАТКУ ДЛЯ НАВЧАЛЬНОГО ВІДДІЛУ ЗВО

Розробка веб-додатку для навчального відділу закладу вищої освіти може значно полегшити процес взаємодії між студентами, викладачами та адміністрацією. Основні функціональні можливості, які такий додаток може включати: електронна реєстрація та планування занять, система онлайн звітності та оцінювання успішності студентів, можливість доступу до навчальних матеріалів та ресурсів, функція обміну повідомленнями та комунікації між учасниками навчального процесу, модуль для організації та відслідковування академічних досягнень. Робота над проєктом може значно покращити ефективність роботи навчального відділу та зробити навчальний процес більш зручним та доступним для всіх його учасників. При розробці додатку є необхідність в захисті персональних даних та конфіденційності інформації користувачів.

Можливості розробленого додатку повинні враховувати потреби користувачів. Відносно функціональності додатку – це створення списку можливостей, які повинен мати додаток, включаючи можливість зареєструватися, отримувати доступ до навчального матеріалу, спілкуватися з викладачами тощо. Необхідно врахувати наявність інтуїтивного користувацького інтерфейсу та зручної навігації для забезпечення комфортного користування додатком. Необхідно також врахувати можливості інтеграції додатку з іншими системами, які використовуються в навчальному закладі для забезпечення одностайності та ефективності роботи [1- 4].

Для розробки такої програми необхідно:

1. Вибрати мову програмування для реалізації додатку.
2. Розглянути можливість створення зручного інтерфейсу.
5. Ретельно протестувати розроблену програму.

Виконуючи такі кроки, було розроблено програму, яка може бути корисною для навчального відділу ЗВО.

За результатами дослідження використання ІТ для створення веб-додатку для навчального відділу закладу вищої освіти може значно полегшити процес розробки навчального плану в ЗВО. У роботі описано вибір мови програмування та її переваги і недоліки. Для розробленого додатку представлено та описано логічну структуру та представлено інструкцію для користувача.

ПЕРЕЛІК ПОСИЛАНЬ:

1. Портал освітньо-інформаційних послуг «Студентська консультація» - Навчальний план вищого навчального закладу. URL: <http://studcon.org/navchalnyy-plan-vyshchogo-navchalnogo-zakladu>
2. Роберт Агулар "HTML і CSS. Основа будь-якого сайту". Ексмо, 2018 р.
3. Розробка та впровадження сучасних методів організації освітнього процесу у закладах вищої освіти. URL: <https://2023.moodlemoot.in.ua/course/view.php?id=9>
4. Складання навчальних планів вищих навчальних закладів. URL: http://www.dut.edu.ua/uploads/1_889_77736557.pdf

Матвійчук О.В., студент кафедри програмних засобів і технологій, Херсонський національний технічний університет, м.Хмельницький, Україна,

Огнєва О.Є., к.т.н., доцент кафедри програмних засобів і технологій, Херсонський національний технічний університет, м.Хмельницький, Україна.

РОЗРОБКА ПРОТОТИПУ ШАБЛОНУ ДЛЯ ОПТИМІЗАЦІЇ ІНТЕРНЕТ-МАГАЗИНІВ ПРИ НИЗЬКІЙ ШВИДКОСТІ ІНТЕРНЕТУ

Існують певні практики та патерни розробки, спеціально призначені для покращення роботи веб-сайтів, включаючи інтернет-магазини, при поганому інтернет-з'єднанні. Ось декілька ключових підходів [1-4]:

- Прогресивний веб-додаток PWA дозволяє веб-додаткам завантажуватися швидше та працювати в офлайн режимі, використовуючи технології, такі як Service Workers, Cache API, та IndexedDB для кешування ресурсів та даних на стороні клієнта.
- Lazy Loading дозволяє завантаження контенту за запитом (наприклад, зображень, відео) лише тоді, коли вони потрібні (наприклад, коли користувач прокручує сторінку до них), що дозволяє зменшити початковий час завантаження сторінки.
- Зменшення розміру файлів CSS, JavaScript, та зображень через мінімізацію та компресію може значно покращити час завантаження сторінки.
- Використання CDN допомагає зменшити затримку, зберігаючи копії веб-ресурсів на серверах, розташованих ближче до кінцевих користувачів.

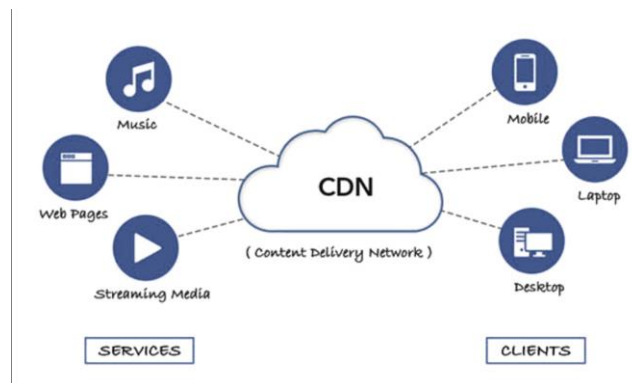


Рисунок 1 - Використання CDN

- Використання форматів зображень, оптимізованих для вебу (наприклад, WebP), та імплементація адаптивного завантаження зображень з різною роздільною здатністю залежно від швидкості з'єднання та розміру екрану.



Рисунок 2 - Оптимізація зображень

- Використання `<link rel="preload">` для попереднього завантаження критичних ресурсів, які будуть потрібні наступними.
- Конфігурація HTTP заголовків для кешування може допомогти зберегти статичні ресурси в кеші браузера, зменшуючи необхідність їх повторного завантаження.

Переваги використання CDN [1-4]:

- Завдяки тому, що контент доставляється з найближчого до користувача сервера, час завантаження сторінки знижується.
- Розподіл навантаження по всій мережі CDN знижує навантаження на основний сервер, дозволяючи обслуговувати більшу кількість користувачів одночасно.
- У разі відмови одного з серверів CDN, запити автоматично перенаправляються на інший доступний сервер, що забезпечує вищу ступінь доступності контенту.

Формат від Google, який забезпечує значно краще стиснення зображень (як з втратою, так і без втрати якості) порівняно з традиційними форматами, такими як JPEG та PNG. WebP підтримує прозорість (альфа-канал) і анімацію [1-4].

Переваги:

- Покращення продуктивності завдяки оптимізації ресурсів та зменшенню часу завантаження.
- Зменшення використання даних для користувачів з обмеженим інтернет-з'єднанням.
- Підвищення доступності контенту в офлайн режимі через PWA.

Недоліки:

- Складність розробки та потреба в додаткових знаннях та інструментах для реалізації деяких оптимізацій.
- Потреба в регулярному оновленні та тестуванні для підтримки сумісності з новими браузерами та технологіями.
- Можливість перевантаження кешу при неправильній конфігурації політик кешування, що може призвести до зберігання застарілого контенту.

Враховуючи зазначені переваги та недоліки, вибір та реалізація шаблону для програмування повинні базуватися на глибокому аналізі потреб користувачів та технічних можливостей розробників. Застосування сучасних підходів і технік оптимізації забезпечить створення ефективних, швидких і доступних веб-сайтів, зокрема інтернет-магазинів, незалежно від швидкості інтернет-з'єднання.

ПЕРЕЛІК ПОСИЛАНЬ:

1. Connolly, T., Begg, C. (2015). Database Systems: A Practical Approach to Design, Implementation, and Management. Pearson.
2. Silberschatz, A., Korth, H., & Sudarshan, S. (2020). Database System Concepts. McGraw-Hill.

3. Elmasri, R., & Navathe, S. B. (2017). Fundamentals of Database Systems. Pearson.
4. Date, C. J. (2019). An Introduction to Database Systems. Addison-Wesley.

Мацків В.О., студент кафедри програмних засобів і технологій, Херсонський національний технічний університет, м.Хмельницький, Україна,

Огнєва О.Є., к.т.н., доцент кафедри програмних засобів і технологій, Херсонський національний технічний університет, м.Хмельницький, Україна.

ВПЛИВ РІЗНИХ СТРУКТУР ДАНИХ НА ПРОДУКТИВНІСТЬ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

У сучасному програмуванні ефективність програмного забезпечення значною мірою залежить від вибору структури даних, оскільки це впливає на продуктивність операцій, таких як доступ, вставка, видалення та сортування. Правильний вибір структури даних здатний значно підвищити швидкість роботи програми, знизити витрати пам'яті та забезпечити масштабованість.

Існує безліч структур даних, кожна з яких має свої переваги в залежності від задачі. Лінійні структури даних, такі як масиви, списки, черги та стеки, часто використовуються для швидкого доступу до елементів або обробки даних у певній послідовності. Для складніших завдань, таких як збереження зв'язків між елементами або впорядкування даних, застосовують дерева і граfi. Хеш-таблиці оптимізують пошук даних, що робить їх ідеальними для роботи з великими обсягами даних у реальному часі.

Практичне застосування цих структур у програмуванні вимагає врахування таких аспектів, як ефективність використання пам'яті та швидкість виконання операцій. Для цього необхідно проводити експериментальний аналіз продуктивності структур даних, щоб визначити, які з них є найбільш ефективними для конкретних завдань, зокрема в умовах великих обсягів даних та багатопоточності.

Метою дослідження в цьому напрямку є проведення аналізу впливу різних структур даних на продуктивність програмного забезпечення з акцентом на оптимізацію швидкості операцій та використання пам'яті в реальних задачах[1, 2, 4].

В результаті дослідження розглянуто конкретні приклади застосування результатів експериментів для реальних задач, таких як сортування, пошук, управління даними в реальному часі та аналіз великих даних.

Для задач, пов'язаних із зберіганням великих обсягів структурованих даних (наприклад, в реляційних базах даних), використання В-дерев або хеш-таблиць дозволяє значно зменшити час пошуку, вставки та видалення.

Для задач, що потребують обробки даних у реальному часі (наприклад, системи управління потоками даних або реєстрації подій), черги та хеш-таблиці забезпечують оптимальну швидкість обробки запитів. Черги дозволяють ефективно організувати обробку запитів у порядку їх надходження (FIFO), а хеш-таблиці дозволяють здійснювати швидкий доступ до елементів на основі ключа [1-4].

Для машинного навчання часто використовуються матриці (як частина обчислювальних тензорів) для організації великих наборів числових даних. Для ефективної обробки таких даних використовуються масиви та списки для збереження вхідних даних і обчислень, а також дерева рішень для класифікації та регресії.

У соціальних мережах для моделювання відносин між користувачами використовуються граfi, оскільки вони дають можливість моделювати складні зв'язки між об'єктами, такі як "дружба", "підписка", "пошук спільних інтересів" тощо. Алгоритми, побудовані на графах, забезпечують ефективне управління зв'язками між користувачами.

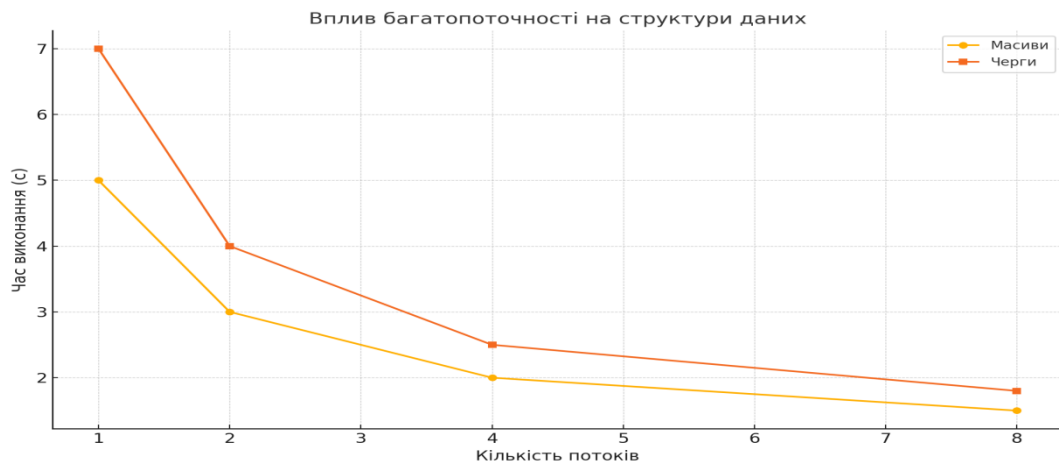


Рисунок 1 - Вплив багатопоточності на продуктивність різних структур даних (масиви, черги). Час виконання операцій порівнюється для різних структур даних при різній кількості потоків.

Проведено експерименти на великих наборах даних (від 1000 до 10 мільйонів елементів). Оцінено час виконання основних операцій: пошук, вставка, видалення.

Для кожної структури даних оцінювалась стійкість до великого навантаження, а також ефективність при паралельних операціях.

Таблиця 1.

Результати тестування на великих обсягах даних

Розмір даних	Масив (с)	Список (с)	Хеш-таблиця (с)	Дерево (AVL) (с)
1000	0.0003	0.0004	0.0001	0.0002
10000	0.003	0.004	0.001	0.001
100000	0.03	0.04	0.01	0.02

У ході дослідження було вивчено основні типи структур даних та їх вплив на продуктивність програмного забезпечення. Були проведені численні експерименти, що дозволили оцінити ефективність операцій пошуку, вставки, видалення та обробки великих даних для різних структур даних.

Хеш-таблиці продемонстрували найкращу продуктивність для задач, що вимагають швидкого доступу до даних.

Дерева (зокрема AVL-дерева) показали високу ефективність при виконанні збалансованих операцій.

Масиви і списки продемонстрували свою перевагу в простих задачах із малими наборами даних, однак виявились менш ефективними при великих обсягах інформації.

Основні наукові та практичні результати: визначено, що правильний вибір структури даних безпосередньо впливає на швидкість виконання програм, розроблені рекомендації щодо вибору структур даних для різних типів задач, проведено експериментальні дослідження, що дозволяють порівняти продуктивність різних структур даних.

Перспективи подальших досліджень можуть бути спрямовані на:

- розробку нових оптимізованих структур даних для спеціалізованих задач;
- вивчення впливу нових технологій, таких як квантові обчислення, на ефективність алгоритмів і структур даних;
- дослідження паралельних та розподілених структур даних для великих обсягів даних та їх обробки в реальному часі.

Також важливим напрямком є оптимізація існуючих структур даних для хмарних обчислень, де вимоги до масштабованості та ефективності обробки великих даних особливо актуальні. Застосування новітніх технологій, таких як штучний інтелект для адаптивних структур даних, може стати важливим етапом у розвитку галузі.

ПЕРЕЛІК ПОСИЛАНЬ:

1. Cormen T. H., Leiserson C. E., Rivest R. L., Stein C. Introduction to Algorithms. MIT Press, 2022.
2. Sedgewick R., Wayne K. Algorithms. Addison-Wesley, 2020.
3. Goodrich M., Tamassia R. Data Structures and Algorithms in Java. Wiley, 2022.
4. Heineman G. T., Pollice G., Selkow S. Algorithms in a Nutshell. O'Reilly Media, 2021.

Міщенко Д.М., студент кафедри програмних засобів і технологій, Херсонський національний технічний університет, м.Хмельницький, Україна,

Фролова М.Е., старший викладач кафедри загальноосвітніх гуманітарних та природничих дисциплін, Херсонський національний технічний університет, м.Хмельницький, Україна.

ВИКОРИСТАННЯ ІТ ДЛЯ РОЗРОБКИ ДОДАТКУ ЧИСЕЛЬНОГО РОЗВ'ЯЗКУ СЛАР

Звичайно, розробка програми для чисельного розв'язання систем лінійних рівнянь алгебри – це важливе завдання. Для ефективного вирішення таких систем рівнянь часто використовують різні алгоритми, такі як метод Гауса або метод LU-розкладання. Ці методи дозволяють знайти рішення системи рівнянь, що може бути корисним у багатьох областях, включаючи інженерні розрахунки, фізику та економіку. Цей тип застосування може бути дуже корисним для економічного аналізу та моделювання [1, 2].

Для розробки такої програми необхідно:

1. Вибрати мову програмування для реалізації алгоритмів розв'язування систем лінійних рівнянь.
2. Дослідити та вибрати відповідні чисельні методи для розв'язування систем лінійних алгебраїчних рівнянь, такі як елімінація Гауса, LU-розклад або ітераційні методи, такі як Якобі чи Гаусса-Зейделя.
3. Написати функції на обраній мові програмування для реалізації чисельних методів розв'язування лінійних алгебраїчних рівнянь.
4. Розглянути можливість створення зручного інтерфейсу для введення коефіцієнтів лінійних рівнянь і перегляду розв'язків.
5. Ретельно протестувати розроблену програму за допомогою різних наборів лінійних рівнянь, щоб забезпечити точні та надійні результати.

Виконуючи такі кроки, можна розробити корисну програму для вирішення систем лінійних алгебраїчних рівнянь для економіки [19].

Обрати мову програмування дуже важко через їх велику кількість в наш час. Адже на перший погляд мови не відрізняються між собою і зробити вибір - це найвідповідальніший крок до своєї майбутньої професії. Для реалізації програмного додатку було обрано мову Java, яка дозволяє створювати практично все.

Продукти Java можна зустріти на кожному кроці. Її використовують для рішень в: державній сфері, в науці, освіті, сфері охорони здоров'я, розваг, в приватному секторі при створенні програм для трейдингу, серверних додатків для банкінгу та для багатьох інших корпоративних цілей [1, 2].

Найголовніша перевага цієї мови: на ній вже написано настільки багато програм, що відійти від неї буде дуже важко і майже неможливо. До того ж, відбуваються постійні оновлення та перезапуски продуктів на мові, та оновлення самої мови. Вона не мертва, її постійно удосконалюють та зміцнюють позиції серед інших конкурентів [1, 2].

Ще один аргумент на користь Java - вона зайняла нішу розробки додатків під Android.

За результатами проведеного дослідження відносно використання методу Гауса для знаходження рішення СЛАР можна сказати, що він не зовсім піддається узагальненим методам програмування. Метод Гауса використовується більше для систем невеликої розмірності (до порядку 10^4). Для великих розмірностей чи розріжених матриць ефективніше є використання ітераційних методів [1, 2]. Модифікований метод Гауса з вибором головного елемента являється достатньо стійким до похибок і при цьому не вимагається великих допоміжних витрат. Він є не набагато але точнішим ніж інші методи.

За результатами дослідження було розроблено додаток для пошуку рішення СЛАР по модифікованому методу Гауса, а саме з вибором головного елементу.

ПЕРЕЛІК ПОСИЛАНЬ:

1. Архітектура мови програмування Java. URL: <https://promoter.net.ua/articles/arxitektura-movi-programuvannia-java.html>
2. Розробка додатку знаходження рішень системи лінійних алгебраїчних рівнянь методом Гауса. URL: <https://www.bestprog.net/uk/2016/03/06/011-c-builder>

Морозов О.Є., студент кафедри програмних засобів і технологій, Херсонський національний технічний університет, м.Хмельницький, Україна,

Хохлов В.А., к.т.н., доцент кафедри програмних засобів і технологій, Херсонський національний технічний університет, м.Хмельницький, Україна.

РОЗРОБКА ІНФОРМАЦІЙНОЇ СИСТЕМИ З ОБЛІКУ КАРТОТЕК ДЛЯ ПОЛІКЛІНІКИ

У сучасному світі інформаційних технологій ІТ фахівці постійно стикаються з необхідністю доступу до актуальних знань та ресурсів. З розвитком інтернету та цифрових технологій виникла потреба у створенні веб-орієнтованих інформаційних систем, які дозволяють швидко знаходити, зберігати та використовувати інформацію [1-5]. Бібліотеки для ІТ фахівців, оснащені сучасними інформаційними системами, стають важливим інструментом для підтримки професійного розвитку та підвищення кваліфікації.

Необхідність розробки бази даних для веб-орієнтованої інформаційної системи створення бібліотеки для ІТ фахівців обумовлена кількома факторами:

- Інформаційні технології розвиваються швидкими темпами, і ІТ фахівці потребують постійного доступу до новітніх досліджень, статей, книг, курсів та інших ресурсів.
- Веб-орієнтовані системи дозволяють забезпечити зручний та швидкий доступ до великої кількості ресурсів з будь-якої точки світу, що особливо важливо в умовах глобалізації та дистанційної роботи.
- Створення централізованої бази даних дозволяє ефективно організувати та керувати великим обсягом інформації, що спрощує процес пошуку та використання необхідних матеріалів.
- Інформаційні системи дозволяють організувати спільну роботу, обмін знаннями та досвідом між фахівцями, що сприяє розвитку професійних спільнот та підвищенню загального рівня знань.

Основною метою розробки бази даних для веб-орієнтованої інформаційної системи створення бібліотеки для ІТ фахівців є створення ефективної, зручної та масштабованої платформи для зберігання, пошуку та управління різноманітними ресурсами. Досягнення цієї мети передбачає вирішення наступних завдань:

- Проєктування структури бази даних, визначення основних сутностей, їх атрибутів та взаємозв'язків, що забезпечить надійне зберігання інформації.
- Створення потужних інструментів для пошуку інформації за ключовими словами, категоріями, авторами та іншими параметрами.
- Забезпечення можливості інтеграції з іншими бібліотеками, науковими базами даних та освітніми платформами для розширення доступу до ресурсів.
- Розробка зручного та інтуїтивно зрозумілого інтерфейсу для доступу до інформаційної системи, що дозволить користувачам ефективно взаємодіяти з базою даних.
- Реалізація механізмів захисту даних та управління доступом, що забезпечить конфіденційність та цілісність інформації.

В результаті проведеної роботи була розроблена база даних для веб-орієнтованої інформаційної системи створення бібліотеки для ІТ фахівців, яка включає наступні компоненти: визначено основні таблиці (ресурси, категорії, автори, користувачі тощо), їх атрибути та взаємозв'язки; реалізовано потужний пошуковий механізм, який дозволяє здійснювати пошук за різними критеріями та фільтрувати результати; забезпечено можливість інтеграції з іншими інформаційними системами та базами даних, що розширило доступ до різноманітних ресурсів; розроблено зручний веб-інтерфейс, який дозволяє користувачам

ефективно взаємодіяти з системою, здійснювати пошук, додавати та редагувати ресурси, а також обмінюватися знаннями; реалізовано механізми аутентифікації та авторизації користувачів, захисту даних та управління доступом.

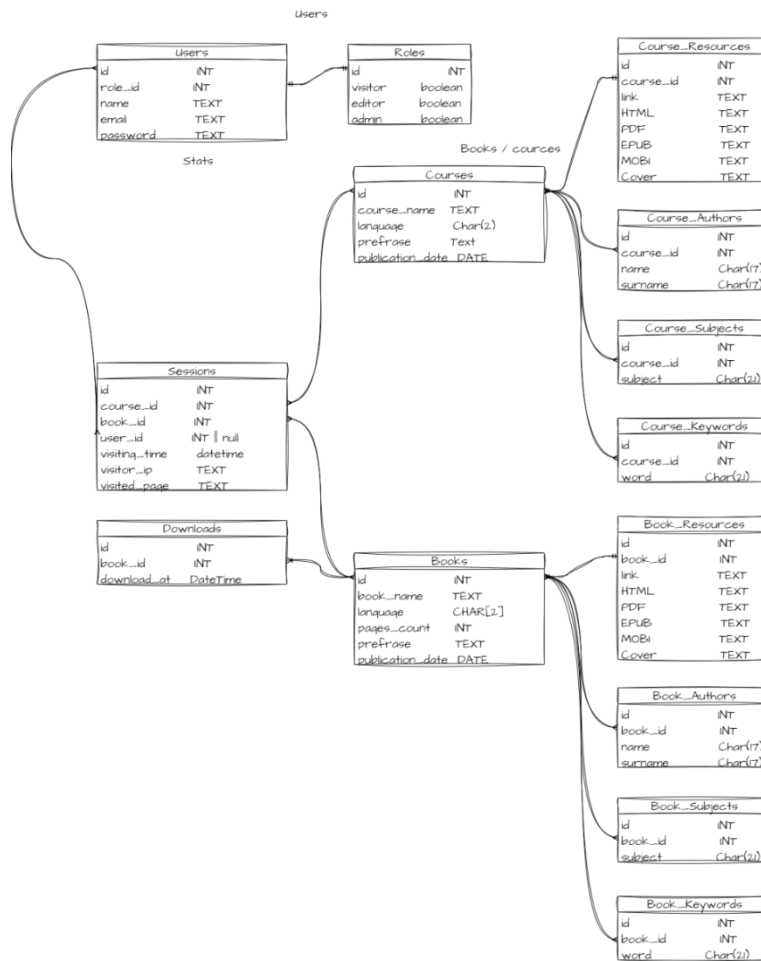


Рисунок 1 – Схема БД

Розробка бази даних для веб-орієнтованої інформаційної системи створення бібліотеки для ІТ фахівців є важливим кроком до забезпечення ефективного доступу до актуальних знань та ресурсів. Така система сприяє підвищенню кваліфікації ІТ фахівців, дозволяє швидко знаходити та використовувати необхідну інформацію, підтримує спільну роботу та обмін знаннями. Впровадження такої інформаційної системи не лише покращує якість професійної діяльності, але й сприяє розвитку інновацій та прогресу у сфері інформаційних технологій.

ПЕРЕЛІК ПОСИЛАНЬ:

1. Connolly, T., Begg, C. (2015). Database Systems: A Practical Approach to Design, Implementation, and Management. Pearson.
2. Silberschatz, A., Korth, H., & Sudarshan, S. (2020). Database System Concepts. McGraw-Hill.
3. Elmasri, R., & Navathe, S. B. (2017). Fundamentals of Database Systems. Pearson.
4. Date, C. J. (2019). An Introduction to Database Systems. Addison-Wesley.
5. Ramakrishnan, R., & Gehrke, J. (2003). Database Management Systems. McGraw-Hill.

Нікулінський Д.В., студент кафедри програмних засобів і технологій, Херсонський національний технічний університет, м.Хмельницький, Україна,

Огнєва О.Є., к.т.н., доцент кафедри програмних засобів і технологій, Херсонський національний технічний університет, м.Хмельницький, Україна.

ДОСЛІДЖЕННЯ МОЖЛИВОСТЕЙ ВИКОРИСТАННЯ ШТУЧНОГО ІНТЕЛЕКТУ ДЛЯ АНАЛІЗУ КОМПЛЕКТУЮЧИХ ПК

Штучний інтелект у сучасному світі стає все більш важливим інструментом для оптимізації та автоматизації різних сфер людської діяльності. Однак, в контексті використання штучного інтелекту для аналізу комплектуючих ПК, існують певні проблеми та виклики [1-3].

Об'єктом дослідження є використання штучного інтелекту для аналізу комплектуючих ПК. Застосування штучного інтелекту в цій сфері може значно полегшити процес вибору та підбору компонентів для побудови персональних комп'ютерів, забезпечуючи оптимальну сумісність та високу продуктивність системи.

У роботі використовується аналіз даних технічних характеристик комплектуючих, розробка та оптимізація алгоритмів машинного навчання, експериментальне тестування алгоритмів на реальних конфігураціях ПК.

Результати дослідження вносять новаторський підхід до використання штучного інтелекту в сфері аналізу комплектуючих ПК. Практично, це дозволяє кінцевим користувачам та фахівцям з інформаційних технологій здійснювати швидкий та точний вибір компонентів для оптимальної конфігурації ПК.

Програма зручна та легка в користуванні. Аналіз виконується за допомогою вибору комплектуючих та натискання кнопки “Аналізувати”, приклад роботи програми можна побачити на рис. 1,2.

Штучний інтелект для аналізу комплектуючих ПК

Вибірть процесор:
Intel Core i3

Вибірть відеокарту:
NVIDIA GTX 1050

Вибірть оперативну пам'ять:
8 GB

Аналізувати

Результат аналізу буде тут

Рисунок 1 - Головна сторінка додатку

Аналізувати

Бюджетний ПК.

Обрані комплектуючі:

Процессор: Intel Core i3

Відеокарта: GTX 1050

Оперативна пам'ять: 8 GB

Рисунок 2 - Результат

Дослідження вказує на необхідність подальших розвідок у напрямку вдосконалення алгоритмів та їх застосування в реальних умовах. Перспективи включають розширення обсягу дослідження на нові покоління комплектуючих та інтеграцію з іншими галузями інформаційних технологій.

Матеріал дослідження розкриває актуальність використання штучного інтелекту для аналізу комплектуючих ПК та визначає його значущість у сучасному інформаційному суспільстві. Обрана проблематика виникла з необхідності покращення процесу вибору та підбору компонентів для персональних комп'ютерів, щоб забезпечити оптимальну сумісність та високу продуктивність систем.

ПЕРЕЛІК ПОСИЛАНЬ:

1. Jaspin K, Sanjeev S, Shyam S,. (2023). Chatbot for Trading Cryptocurrency. 1-9. 10.1109/ACCAI58221.2023.10200405.
2. Plakhotna, Y. Zahreba, Central. (2021). Chatbot as a Trading Tool in the Cryptocurrency Market. Business Inform.
3. Огляд Cryptohopper. URL: <https://tradingbot.info/ua/obzor-cryptohopper/>

Орлик О.О., студент кафедри програмних засобів і технологій, Херсонський національний технічний університет, м. Херсон, Україна
Боскін О.О., ст. викладач кафедри програмних засобів і технологій, Херсонський національний технічний університет, м. Хмельницький, Україна.

РОЗРОБКА СИСТЕМИ ПЛАНУВАННЯ ЗАВДАНЬ

Архітектура програмної системи базується на архітектурному патерні. Архітектурний патерн - це опис типової організації системи. Архітектурні патерни фіксують суть архітектури, яка використовувалася в різних програмних системах. Архітектурний патерн можна розглядати як узагальнений опис належної практики, перевіреної в різних системах та середовищах. Таким чином, архітектурний патерн повинен описати системну організацію, яка була успішною в попередніх системах. Для побудови архітектури системи було використано архітектурний патерн «Модель-Подання-Контролер» (Model-View-Controller) MVC [1-5].

Патерн відокремлює подання системи та взаємодію із системою від даних системи. Система поділяється на три логічних компоненти, які взаємодіють один з одним. Компонент «Модель» керує системними даними та операціями над даними. Компонент «Подання» відображає дані для користувача. Компонент «Контролер» взаємодіє з користувачем, ініціює операції в моделі і керує роботою подання. Патерн MVC дозволяє змінювати дані незалежно від їхнього подання. Зміна даних, зроблена в одному поданні, відображається у всіх інших поданнях програмної системи [1-5].

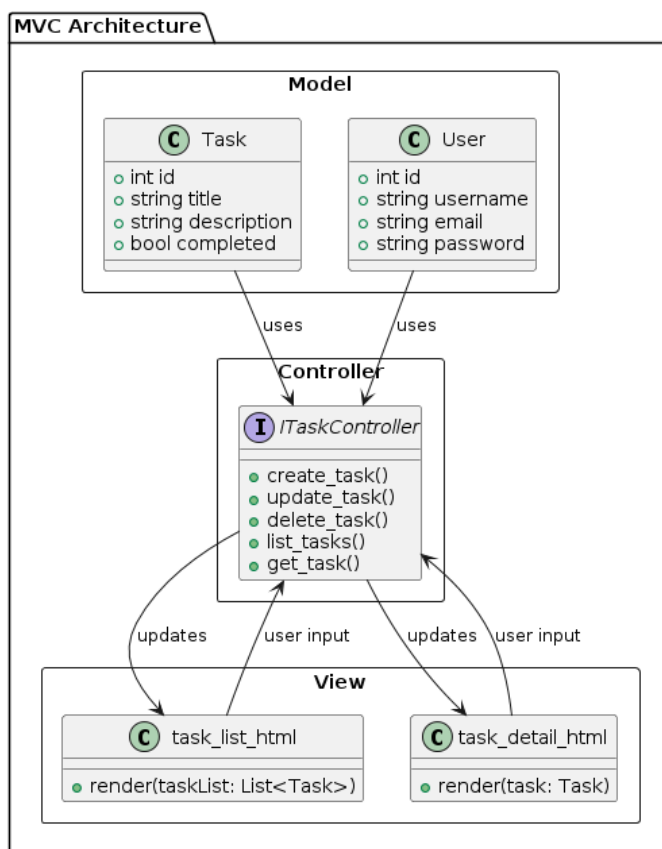


Рисунок 1 - Архітектура програмної системи на основі патерна MVC

Після проєктування системної структури та визначення принципів управління структурою слід виконати поділ підсистем на модулі. Фактично цю роботу можна вважати введенням у детальне проєктування, яке конкретизує архітектурні рішення. Завдання декомпозиції - це завдання визначення внутрішнього змісту кожної підсистеми (компонента). Результатом її вирішення є формування структури підсистеми - набору модулів та відносин їх взаємодії. Відомі два підходи та два типи моделей для декомпозиції підсистем на модулі: модель потоку даних та об'єктно-орієнтована модель. В роботі було застосовано об'єктно-орієнтовану модель, в якій дані подаються як об'єкти-сутності, що мають власні набори даних, стани, набори операцій та їх описи (класи).

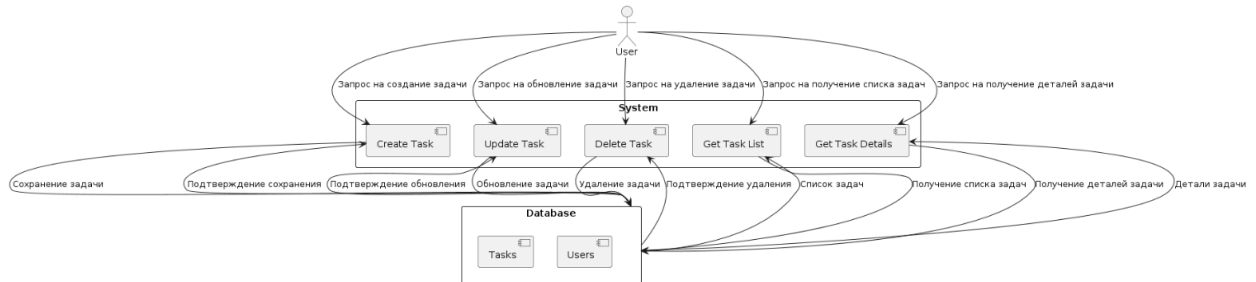


Рисунок 2 - Модуль розподілу завдань. Поток даних

Розроблено архітектуру програмної системи на основі патерна MVC. Описано процес розроблення структури модулів програмної системи. Розроблено діаграму потоків даних для модуля розподілу завдань, призначеного для збирання та зберігання інформації про завдання.

ПЕРЕЛІК ПОСИЛАНЬ:

1. Бендер, Р., Нейм, Т. (2007). Beginning Database Design: From Novice to Professional. Apress.
2. Вейзер, С. (2012). Pro Web 2.0 Application Development with GWT. Apress.
3. Гамма, Е., Хелм, Р., Джонсон, Р., Вліссідес, Дж. (1994). Design Patterns: Elements of Reusable Object-Oriented Software. Addison-Wesley.
4. Карні, Г. (2002). "PostgreSQL: Introduction and Concepts. Addison-Wesley.
5. Ларман, К. (2004). Applying UML and Patterns: An Introduction to Object-Oriented Analysis and Design and Iterative Development. Prentice Hall.

Петров В.О., студент кафедри програмних засобів і технологій, Херсонський національний технічний університет, м.Хмельницький, Україна,

Огнєва О.Є., к.т.н., доцент кафедри програмних засобів і технологій, Херсонський національний технічний університет, м.Хмельницький, Україна.

ВИВЧЕННЯ НОВИХ ТЕНДЕНЦІЙ ТА ТЕХНОЛОГІЙ У ГАЛУЗІ РОЗРОБКИ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

Сучасна індустрія розробки програмного забезпечення (ПЗ) зазнає постійних змін під впливом нових технологій та підходів. Зокрема, хмарні обчислення, Інтернет речей (IoT) та великі дані (Big Data) є одними з ключових напрямів, які активно впливають на процеси створення, розгортання та використання програмних продуктів. Вивчення цих тенденцій дозволяє оцінити їх вплив на галузь і визначити майбутні перспективи розвитку [1-5].

Хмарні обчислення (Cloud Computing) являють собою модель, яка забезпечує зручний мережевий доступ до спільного пулу конфігурованих обчислювальних ресурсів (наприклад, мережі, сервери, сховища, додатки та сервіси). Це значно спрощує управління інфраструктурою та дозволяє зосередитись на розробці програмного забезпечення [1-5].

Основні переваги хмарних обчислень включають зменшення витрат на фізичну інфраструктуру, гнучкість у масштабуванні ресурсів, підвищення надійності та доступності систем. Наприклад, використання платформ як-от AWS, Google Cloud Platform або Microsoft Azure дозволяє розробникам швидко розгорнути та тестувати нові програми, забезпечуючи при цьому високу продуктивність та безперебійний доступ до сервісів [1-5].

Проте, використання хмарних технологій також має свої виклики, зокрема питання безпеки даних та відповідності нормативним вимогам. Необхідність забезпечення конфіденційності та захисту інформації стає ключовою проблемою, яка вимагає ретельного підходу до управління правами доступу та використання шифрування даних.

Інтернет речей (IoT) являє собою мережу фізичних пристроїв, транспортних засобів, будівель та інших об'єктів, оснащених електронікою, програмним забезпеченням, сенсорами та мережевою підключеністю. Це дозволяє цим об'єктам збирати та обмінюватися даними, створюючи можливості для автоматизації та підвищення ефективності різних процесів [1-5].

У галузі розробки програмного забезпечення IoT відкриває нові горизонти для створення додатків, які можуть керувати пристроями, аналізувати дані в реальному часі та забезпечувати інтеграцію різних систем. Наприклад, у сфері розумних міст (smart cities) використовуються IoT-рішення для управління трафіком, енергоспоживанням та безпекою. Водночас, розробка ПЗ для IoT включає виклики, пов'язані з сумісністю пристроїв, забезпеченням безпеки та обробкою великих обсягів даних [1-5].

Великі дані (Big Data) стосуються технологій і методів для збирання, зберігання, обробки та аналізу величезних обсягів даних. Завдяки аналізу великих даних, організації можуть отримувати нові знання, виявляти тенденції та робити прогнози, які раніше були недоступні.

У розробці програмного забезпечення великі дані застосовуються для створення аналітичних платформ, які можуть обробляти і аналізувати дані з різних джерел у реальному часі. Наприклад, у фінансовому секторі великі дані використовуються для виявлення шахрайства, аналізу ринкових тенденцій та управління ризиками. Основні виклики, пов'язані з великими даними, включають складність управління даними, забезпечення їхньої конфіденційності та інтеграцію різнорідних джерел інформації.

Нові тенденції та технології, такі як хмарні обчислення, Інтернет речей та великі дані, значно змінюють підходи до розробки програмного забезпечення. Вони не лише підвищують ефективність та продуктивність процесів створення ПЗ, але й відкривають нові можливості

для інноваційного розвитку. Однак, впровадження цих технологій вимагає врахування нових викликів, таких як безпека, конфіденційність та управління складністю систем. Подальші дослідження у цій галузі повинні зосереджуватися на розробці методів та інструментів, які дозволять максимально ефективно використовувати нові технології, забезпечуючи при цьому високу якість та надійність програмного забезпечення.

ПЕРЕЛІК ПОСИЛАНЬ:

1. Bahga, A., Madiseti, V. (2014). Internet of Things: A Hands-On Approach. VPT.
2. Gubbi, J., Buyya, R., Marusic, S., Palaniswami, M. (2013). Internet of Things (IoT): A vision, architectural elements, and future directions. Future Generation Computer Systems, 29(7), 1645-1660.
3. Kavis, M. J. (2014). Architecting the Cloud: Design Decisions for Cloud Computing Service Models (SaaS, PaaS, and IaaS). Wiley.
4. Marz, N., Warren, J. (2015). Big Data: Principles and best practices of scalable realtime data systems. Manning Publications.
5. Provost, F., Fawcett, T. (2013). Data Science for Business: What you need to know about data mining and data-analytic thinking. O'Reilly Media.

Платошечкін А.М., студент факультету
інформаційних технологій та дизайну,
Херсонський національний технічний
університет

Козуб Н.О., к.т.н, доцент, доцент кафедри
програмних засобів і технологій

АРКАДНІ ІГРИ, ЯК ЗАСІБ ДЛЯ КОРОТКОГО ВІДПОЧИНКУ

Аркадні ігри як засіб для короткого відпочинку мають значний потенціал у сучасному світі, де темп життя постійно зростає, а потреба в ефективному відпочинку стає все більш нагальною. Ці ігри відзначаються кількома ключовими характеристиками, що роблять їх ідеальним інструментом для швидкого розслаблення та відновлення енергії [1-4].

Перш за все, розроблена аркадна гра вирізняється своєю простотою та доступністю. Її правила легко зрозуміти, а геймплей не вимагає тривалого освоєння, що дозволяє гравцям негайно зануритися в гру. Це особливо важливо для людей, які мають обмежений час для відпочинку, оскільки вони можуть одразу отримати задоволення від гри, не витрачаючи час на навчання чи складні стратегії [1-4].

Крім того, аркадні ігри мають швидкий ігровий цикл. Кожна сесія гри зазвичай триває лише кілька хвилин, що дозволяє гравцям швидко переключитися від робочих завдань до гри та назад. Такий формат ідеально підходить для коротких перерв під час робочого дня, коли потрібно відновити увагу та працездатність. Дослідження показують, що короткі, але часті перерви на відпочинок можуть значно підвищити продуктивність і знизити рівень стресу [1-4].

Аркадні ігри також є доступними практично для кожного, незалежно від віку чи технічних навичок. Вони можуть бути встановлені на будь-який пристрій - комп'ютер, смартфон, планшет, що робить їх легкодоступними для гри в будь-який момент. Це особливо важливо в контексті сучасного мобільного способу життя, де люди часто знаходяться в русі та потребують швидких способів відволіктися та розслабитися.

Окрім описаних переваг, аркадні ігри можуть мати й інші позитивні впливи на гравців. Вони сприяють розвитку певних когнітивних навичок, таких як швидкість реакції, координація рухів та прийняття рішень. Багато аркадних ігор вимагають від гравців швидко оцінювати ситуацію та реагувати на змінні умови, що може покращувати здатність до швидкого мислення та адаптації в реальних життєвих ситуаціях.

Не менш важливою є і роль аркадних ігор у формуванні позитивних емоційних звичок. Вони можуть навчити гравців справлятися з поразками, розвивати наполегливість та покращувати настрій навіть після короткої ігрової сесії. В умовах високих вимог сучасного життя це може бути корисним для підтримки емоційного здоров'я та стійкості до стресу.

Найцікавіше в цьому жанрі є те, що він може бути будь-чим, там мати свій особистий шарм. Від шутерів від першої особи, з неймовірно динамічним та швидким геймплеєм по типу Warhammer 40,000: Boltgun, де гра проходить в стилі Doom, але з можливостями сучасних технологій, сам процес став неймовірно плавним. Також важливою частиною жанру є симулятори різної техніки, де вони починаються з тракторів на полі, до космічних літаків. Одними із найяскравіших прикладів симуляторних-аркад є наприклад ACE COMBAT™ 7: SKIES UNKNOWN. В цьому прикладі, необхідно пілотувати бойовим реактивним винищувачем, та перемагати інші літаки які пілотують штучний інтелект різної складності, або в мультиплеєрному режимі проти інших гравців, де в них є приблизно однакові здібності та озброєння як у гравця. Або класичні гонки серії Need for Speed, де необхідно прийти до фінішу першим, коли на трасі можуть знаходитися різні перегороди. Другим жанром який часто асоціюють з аркадою це ігри beat 'em up, одним з старіших прикладів цього жанру є Streets of Rage, де потрібно перемогти велику кількість ворогів у рукопашному бою, іноді використовуючи різні речі, розкидані по карті. Також, одним із класичних жанрів аркад є

файтинги, яскравим прикладом є серія ігор Guilty Gear, де два опоненти борються в дуелі, використовуючи усі свої навички, щоб перемогти свого опонента.

Аркадні комп'ютерні ігри мають нескінченні можливості в різних жанрах, забезпечуючи гравцям широкий спектр розваг та викликів. Від класичних шутерів та платформерів до головоломок і спортивних симуляторів, аркадні ігри можуть задовольнити будь-які вподобання. Кожен жанр пропонує унікальні механіки та стилі геймплею, дозволяючи гравцям відчувати різноманітні емоції та задоволення. Таке багатство жанрових можливостей робить аркадні ігри невичерпним джерелом розваг і відпочинку для всіх вікових груп та інтересів.

Переваги, які були описані, добре ілюструє моя власна аркадна гра. Розроблена з використанням сучасних технологій та інструментів програмування, таких як Unity та C#, ця гра є прикладом того, як аркадні ігри можуть бути інтегровані в повсякденне життя для швидкого відпочинку. Простий інтерфейс та інтуїтивно зрозумілий геймплей дозволяють гравцям миттєво зануритися в гру.

Розроблений проєкт брав натхнення з гри Killing Floor, де гравця спавнять на карту, де його мета перемогти хвилі ворогів, отримуючи при цьому в нагороду кошти, за які гравець може узяти собі більш ефективне озброєння, для більш ефективного проходження сесії. Call of Duty: Zombies має близьку по мотиву ідею, але має можливість до відкриття нової частини карти, та покупати зброю в процесі бою з хвилею ворогів. Має свої модифікатори та можливість отримати випадкову зброю, з спеціальних коробок які знаходяться на карті.

У підсумку, моя аркадна гра пропонує багатогранний підхід до короткого відпочинку, поєднуючи розвагу, соціальну взаємодію, розвиток когнітивних навичок та емоційну підтримку. Завдяки своїй доступності, простоті та здатності швидко залучати гравців, вона стає інструментом для швидкого та ефективного відновлення сил у будь-який час та в будь-якому місці.

ПЕРЕЛІК ПОСИЛАНЬ:

1. NIH. Turning Discovery Into Health. Video gaming may be associated with better cognitive performance. 24 жовтня 2022. URL: <https://www.nih.gov/news-events/news-releases/video-gaming-may-be-associated-better-cognitive-performance-children>
2. Shalini Wa. Impact of Games on the Performance and Engagement of Employees of the IT Industry. Січень 2023. URL: <https://surl.lu/unpkrq>
3. SkillGigs.com. How Gaming Can Improve Your Work Productivity. 27 липня 2022. URL: <https://www.linkedin.com/pulse/how-gaming-can-improve-your-work-productivity-skillgigs>
4. Sultan Hammad Alshammari. The Effectiveness of Video Games in Enhancing Students' Learning. Січень 2015. URL: <https://surli.cc/xjuhzy>

Плутенко Д.В., студент факультету інформаційних технологій та дизайну, Херсонський національний технічний університет, м.Хмельницький, Україна,
Огнєва О.Є., к.т.н., доцент кафедри програмних засобів і технологій, Херсонський національний технічний університет, м.Хмельницький, Україна.

ЗАНУРЕННЯ У СВІТ ЖАХУ: ДОСЛІДЖЕННЯ ТА РОЗРОБКА ХОРОР ІГОР

Хорор ігри займають особливе місце в індустрії відеоігор завдяки своїй здатності викликати сильні емоції та занурювати гравців у світ жаху та напруженості. Цей жанр використовує різноманітні механіки та техніки для створення незабутнього досвіду, який залишає відбиток у пам'яті гравців. Дослідження та розробка хорор ігор вимагає детального розуміння психологічних аспектів страху та способів їх ефективного застосування у грі [1-3].

Одним із ключових аспектів, що роблять хорор ігри ефективними, є використання психологічних тригерів страху. Розробники використовують темні та закриті простори, непередбачувані події, атмосферну музику та звуки, щоб створити напружену атмосферу. Важливим елементом є також елементи сюжету, які поступово розкривають таємниці та загрози, змушуючи гравця відчувати постійний дискомфорт та тривогу. Наприклад, у грі «Tristis Anima», яку я розробляв для своєї кваліфікаційної роботи бакалавра, ці елементи були інтегровані для досягнення максимальної ефективності. Хорошим прикладом використання психології страху є гра «Five Nights at Freddy's», де гравець повинен виживати протягом ночі в ролі охоронця, керуючи обмеженими ресурсами та спостерігаючи за аніматроніками, які можуть атакувати в будь-який момент. Несподівані напади та необхідність постійного моніторингу камер створюють атмосферу постійного напруження та страху [2].

Сучасні технології дозволяють розробникам хорор ігор створювати високоякісну графіку та звукові ефекти, які посилюють відчуття занурення. Використання інструментів, таких як Unity, дозволяє створювати реалістичні моделі та середовища, а також інтегрувати складні анімації та ефекти. Наприклад, у «Tristis Anima» я використав технологію ProBuilder для моделювання ігрових локацій, а також додав динамічні тіні та освітлення для створення відповідної атмосфери. Прикладом видатного використання технологій для створення атмосфери є демо-гра «P.T.», розроблена Хідео Кодзіма. Ця гра відзначається своєю реалістичною графікою та звуковими ефектами, що створюють неперевершене відчуття жаху та невизначеності. Темні коридори, несподівані звуки та інтерактивні елементи створюють незабутній досвід, який справляє сильне емоційне враження на гравців [3].

Важливим аспектом хорор ігор є розробка інтелектуальних ворогів, які реагують на дії гравця та створюють додатковий рівень напруженості. У «Tristis Anima» вороги мають складну систему патрулювання та переслідування, що залежить від поведінки гравця. Вороги реагують на звук, світло та інші фактори, що робить їх непередбачуваними та небезпечними. Цей підхід дозволяє створити унікальний досвід кожного проходження гри, підвищуючи її реіграбельність. Класичним прикладом використання AI для створення напруженої атмосфери є гра «Slender: The Eight Pages». У цій грі гравець повинен знайти вісім записок, уникаючи Слендермена, який постійно переслідує гравця. Слендермен з'являється несподівано та поступово наближається до гравця, що створює постійне відчуття страху та невизначеності. AI в цій грі був налаштований так, щоб зберігати елемент несподіванки та напруження [1].

Багато сучасних хорор ігор використовують елементи соціальної взаємодії для підсилення відчуття страху. Кооперативні режими гри дозволяють гравцям ділитися своїми страхами та переживаннями, що створює відчуття колективного страху. У «Tristis Anima» передбачена можливість багатокористувацької гри, де гравці можуть допомагати один одному виживати у світі жаху.

Тестування є невід'ємною частиною розробки хорор ігор. Важливо перевіряти, як гравці реагують на різні елементи гри, щоб вносити корективи та оптимізувати досвід. У процесі розробки «Tristis Anima» я провів кілька етапів тестування, збираючи відгуки гравців та вносячи зміни для покращення геймплею та атмосфери.

Розробка хорор ігор вимагає глибокого розуміння психології страху, використання сучасних технологій та ретельного тестування. Завдяки комплексному підходу можна створити ігри, які не тільки розважають, але й залишають незабутній слід у пам'яті гравців. Проєкт «Tristis Anima» є прикладом того, як використання різноманітних технік та інструментів дозволяє створити занурюючий світ жаху, який викликає сильні емоції та забезпечує унікальний досвід гри.

ПЕРЕЛІК ПОСИЛАНЬ:

1. Five Nights at Freddy's. URL: https://freddy-fazbears-pizza.fandom.com/wiki/Five_Nights_at_Freddy%27s
2. P.T. URL: <https://silenthill.fandom.com/wiki/P.T.>
3. Slender. URL: <https://theslenderman.fandom.com/wiki/Slender>

Ринденко Є.І., студент факультету інформаційних технологій та дизайну, Херсонський національний технічний університет, м.Хмельницький, Україна,
Огнєва О.Є., к.т.н., доцент кафедри програмних засобів і технологій, Херсонський національний технічний університет, м.Хмельницький, Україна.

ПРОЄКТУВАННЯ БАЗИ ДАНИХ ВЕБ-ДОДАТКУ ДЛЯ ПУБЛІКАЦІЇ АВТОРАМИ-ПОЧАТКІВЦЯМИ ТВОРІВ

База даних є важливою складовою будь-якої системи, адже за допомогою неї відбувається структуроване зберігання та ефективне керування великими об'ємами інформації, які зазвичай містять інформаційні системи. Вона допомагає забезпечувати цілісність і послідовність даних, а також спрощує пошук, фільтрацію і аналіз інформації.

Створення ефективної бази даних для веб-додатку публікації творів повинно допомогти авторам-початківцям донести свої творчі ідеї до широкої аудиторії та встановити контакт зі своїми читачами слухачами. Розробка такої бази даних вимагає ретельного аналізу потреб цільової аудиторії та розуміння особливостей творчого процесу.

При проєктуванні бази даних для цієї інформаційної системи варто враховувати основні її цілі:

- створення профілів авторів (реєстрація та автентифікація);
- публікацію та подальший супровід авторами творів;
- взаємодію користувачів з творами;
- здійснення параметризованого пошуку.

Після аналізу цілей системи, визначаються основні об'єкти, з якими вона буде працювати, та взаємодія між ними (рис. 1). Цими об'єктами стали: автори (zareєстровані користувачі), твори, одиниці творів – розділи, теги (характеристики творів), збірки (структуризація творів за участю користувачів).

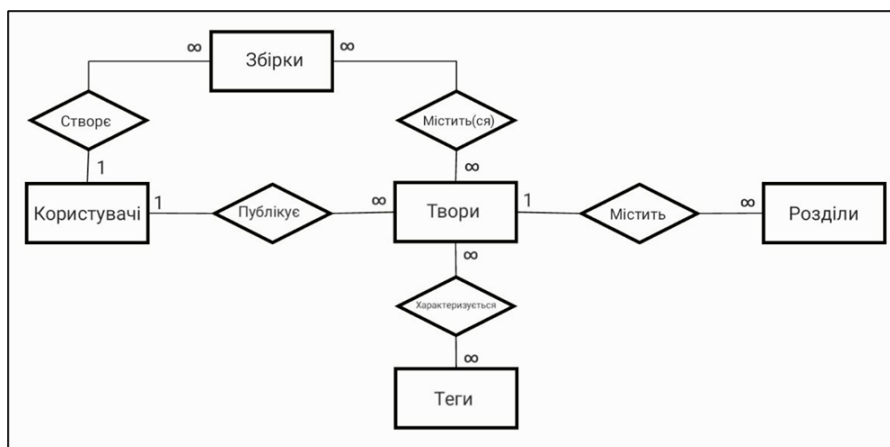


Рисунок 2 – Об'єкти системи та взаємодія між ними

Наступним кроком відбувається перетворення об'єктів у сутності, що є основою для створення таблиць бази даних [1]; визначаються їх атрибути та зв'язки [2]. Визначеними сутностями стали: аватари, автори, мови, твори, розділи, збірки, типи тегів та самі теги (рис. 2).

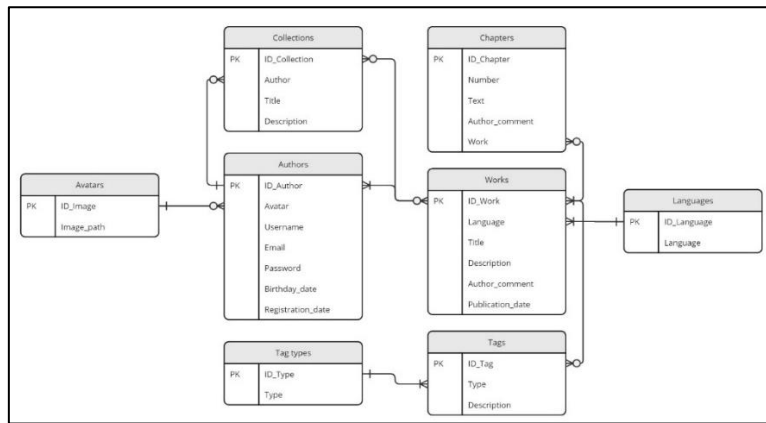


Рисунок 3 – Сутності бази даних, визначені на основі об'єктів

Зв'язки будуються на основі того, що кожен автор у своєму профілі матиме зображення («аватар»), створюватиме збірки та твори. Твори характеризуватимуться тегами, мовою написання, міститимуть розділи та належатимуть до збірок.

Після визначення сутностей, наступним етапом є створення таблиць, що зберігатимуть ці дані. Кожна таблиця відповідає сутності або зв'язку (для реалізації зв'язку типу «багато до багатьох» використовуються проміжні таблиці) та включає всі необхідні атрибути. Для кожного поля таблиці задається певний тип даних, з урахуванням можливостей, які надає система керування базами даних MySQL [3]. Проводиться нормалізація таблиць до 3НФ [4] задля мінімізації надлишкових даних та підвищення ефективності роботи бази даних.

У результаті проєктування виходить база даних, яка складається усього з 11 таблиць (8 основних – аватари, автори, збірки, твори, розділи, типи тегів та теги; 3 проміжних для побудов зв'язків) з визначеними атрибутами та їх типами даних, поєднаними різними типами зв'язків, яка забезпечує ефективне зберігання, доступ та обробку інформації, якої потребує веб-додаток (рис. 3).

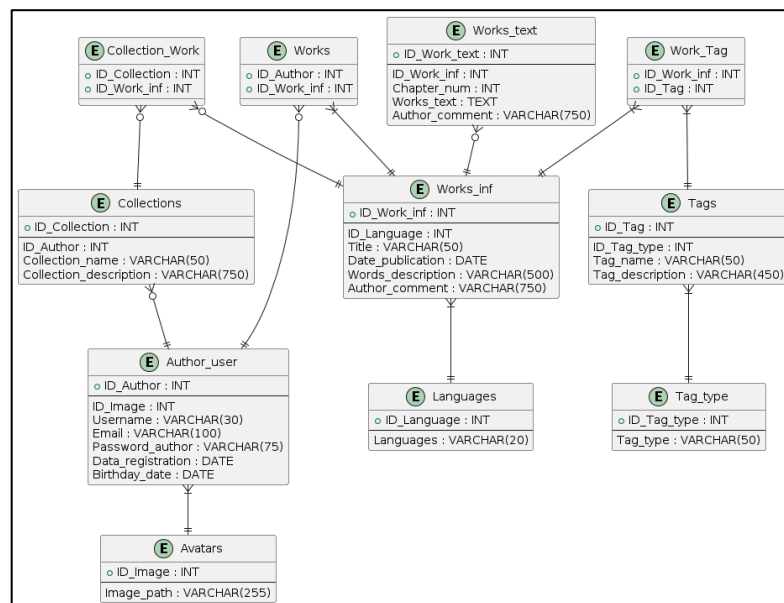


Рисунок 4 – Спроектowana база даних

ПЕРЕЛІК ПОСИЛАНЬ:

1. Поняття сутності, атрибута, ключа, зв'язку. URL: <https://surl.li/dtbxfb>
2. Посібник зі зв'язків між таблицями. URL: <https://surl.li/hhuulh>
3. SQL: Типи даних [URL]: <http://apserver.org.ua/peregl.php?d=view&tid=91>.

4. Hibernate

<https://javarush.com/ua/quests/lectures/ua.questhibernate.level17.lecture02>.

[URL]:

Рогатовська А.А., студентка кафедри програмних засобів і технологій, Херсонський національний технічний університет, м.Хмельницький, Україна,

Огнєва О.Є., к.т.н., доцент кафедри програмних засобів і технологій, Херсонський національний технічний університет, м.Хмельницький, Україна.

ОПТИМІЗАЦІЯ РАЦІОНУ ХАРЧУВАННЯ ДІАБЕТИЧНИХ ХВОРИХ

Людині, хворій на цукровий діабет другого типу, необхідно щодня споживати певну кількість вуглеводів, яка вимірюється в хлібних одиницях.

Отже, ця людина має вживати певну кількість їжі, з якої ці вуглеводи (тобто хлібні одиниці) будуть отримані [1].

Згідно з дослідженнями, кожен продукт харчування має різний рівень вмісту вуглеводів через природу свого походження, що призводить до ситуацій, коли споживання одного продукту призводить до недотримання норми споживання хлібних одиниць, тоді як споживання іншого – до перевищення цієї норми.

Як наслідок, якщо стоїть питання змодельовати оптимальний раціон харчування діабетичних хворих, то необхідно розробити такі обмеження моделі, які б гарантували, що загальна кількість хлібних одиниць у спожитій їжі знаходиться в діапазоні від мінімального до максимального запропонованого значення, залежно від стану конкретного пацієнта.

Суто математично, такі обмеження можна записати у вигляді нерівності (1):

$$C_1 < \sum_{i=0}^{n-1} x_i * BU_i < C_2, \quad (1)$$

де C_1 – мінімальна денна рекомендована до спожиття кількість хлібних одиниць;

C_2 – максимальна денна рекомендована до спожиття кількість хлібних одиниць;

x_i – i -й продовольчий продукт;

BU_i – вміст хлібних одиниць в i -му продукті;

$i = 0, 1, 2 \dots n - 1$ – номер обраного продукту харчування; n – загальна кількість обраних продуктів харчування.

На додаток до різного вмісту хлібних одиниць, кожен продукт має свою ціну, яка може коливатися в залежності від різних змінних, таких як неврожай, військові дії, неможливість імпорту або експорту тощо.

Враховуючи необхідність розподілу коштів та складність їх використання виключно для придбання продуктів харчування, варто додати до моделі обмеження, щоб загальна кількість придбаних продуктів харчування не перевищувала суму, яку ця конкретна людина виділила б як бюджет на ці покупки. Математично, це можна представити у вигляді співвідношення (2).

$$\sum_{i=0}^{n-1} p_i x_i = h, \quad h > 0 \sum_{i=0}^{n-1} p_i x_i = h, \quad h > 0, \quad (2)$$

де x_i – кількість придбаного i -го продукту харчування;

p_i – вартість придбаного i -го продукту харчування;

h – загальна сума, виділена на покупки продовольчих товарів.

Важливо також обмежити максимальне добове споживання кожного i -го продукту, який братиме участь в обчисленнях, щоб не виникали сценарії з нелогічними або навіть безглуздими значеннями, наприклад, 1 кілограм хліба на день. Формула (3) є тим самим обмеженням, яке дозволить це регулювати.

$$0 \leq x_i \leq Am_{max}, \quad (3)$$

де x_i – кількість придбаного i -го продукту харчування в грамах;

Am_{max} – максимальна рекомендована кількість щоденного споживання продукту.

Слід зазначити, що максимальна кількість щоденного споживання в грамах залежить від виду продукту. Наприклад, 300 г для хліба, 450 г для картоплі, 1,5 г для солі та цукру і так далі.

Оскільки сума, витрачена на харчування, яке обирає пацієнт, не повинна становити основну частину бюджету, доречно мінімізувати цю суму, визначивши функцію (4).

$$h \rightarrow \min \quad (4)$$

Отримана функція h може слугувати цільовою для створюваної моделі оптимізації.

Для вирішення задачі оптимізації було розглянуто декілька методів, включаючи метод найменших квадратів, лінійної оптимізації, метод Пауелла та метод Нелдера-Міда [2].

Для порівняння результатів цих методів оптимізації було створено тестовий набір даних (табл. 1).

Таблиця 1.

Тестовий набір даних для перевірки результатів оптимізації

№ продукту	Ціна продукту, грн	Вага / об'єм продукту	Кількість ХО, що міститься у продукті	Максимальна сума витрат, грн	Мінімальна кількість ХО до споживання	Максимальна кількість ХО до споживання
1	33,63	240 г	1,21	1000	23	28
2	28,40	1 кг	0,20			
3	105,90	1 кг	0,56			
4	46,20	1 л	0,36			

На рис. 1 зображено результати оптимізації з використанням підходів, описаних вище. При здійсненні оптимізації максимальна кількість продуктів була вказана 250 г.

```

Лінійна оптимізація:
Кількість кожного продукту, гр: 250.0, 250.0, 250.0, 250.0
Загальна вартість, грн: 80.156
Загальна кількість хлібних одиниць: 1.54

МНК з обмеженнями:
Кількість кожного продукту, гр: 239.153, 250.0, 250.0, 250.0
Загальна вартість, грн: 78.636
Загальна кількість хлібних одиниць: 1.486

Нелдер-Мід:
Кількість кожного продукту, гр: 206.612, 1250.0, 446.429, 694.444
Загальна вартість, грн: 143.812
Загальна кількість хлібних одиниць: 1.792

Пауелл:
Кількість кожного продукту, гр: 206.612, 1250.0, 446.429, 694.444
Загальна вартість, грн: 143.812
Загальна кількість хлібних одиниць: 1.792

```

Рисунок 1 - Отримані результати оптимізації методами найменших квадратів, лінійної оптимізації, Пауелла та Нелдера-Міда

Як можна побачити, методи Пауелла та Нелдера-Міда виявилися більш ефективними з точки зору загальної кількості хлібних одиниць в отриманому раціоні, хоча їхні результати

перевищують максимально рекомендовану добову норму споживання різних продуктів, що змушує відкинути їх у якості методів оптимізації для вирішення поставленого оптимізаційного завдання.

Порівняно з методом найменших квадратів, лінійна оптимізація дає дещо кращі результати, але з вищою загальною вартістю продуктів, що не відповідає обмеженням цільової задачі.

В результаті, для оптимізації рішення цієї задачі найбільш релевантним є метод найменших квадратів, який використовується, щоб вирішити проблему визначення значень змінної x таким чином, щоб кожне з них на першій ітерації $i = 1, \dots, m$ було якомога ближче до заданого значення y_i . Математично, це описується виразом (5).

$$\sum_i e_i^2 = \sum_i (y_i - f_i(x))^2 \rightarrow \min_x \quad (5)$$

При розв'язанні задачі лінійного програмування виду (6)

$$\min_x f(x) \quad (6)$$

з обмеженнями (7) і (8),

$$b(x) \geq 0; \quad (7)$$

$$c(x) = 0, \quad (8)$$

метод найменших квадратів шукає значення функції Лагранжа типу (9).

$$L(x, \lambda, \sigma) = f(x) - \lambda^T b(x) - \sigma^T c(x), \quad (9)$$

де, λ та σ – значення множників Лагранжа.

Отже, отримані результати оптимізації тестового набору даних кількома методами оптимізації показали, що з попередньо обраних для оптимізації методів найбільш підходящим виявився метод найменших квадратів, який має бути обраним для програмної реалізації алгоритму пошуку оптимального рішення поставленої задачі – здійснювати пошук оптимального рішення за актуальними, з точки зору здоров'я хворої на цукровий діабет другого типу людини, критеріями кількості споживання хлібних одиниць.

ПЕРЕЛІК ПОСИЛАНЬ:

1. Про зв'язок вуглеводів і діабету. URL: Все про діабет. <https://diabet.org.ua/pro-zvyazok-vuhlevodiv-i-diabetu/>
2. Зайченко Ю.П. Дослідження операцій. Підручник. К.: Видавничий дім «Слово», 2006. 816 с.

Середюк В.Я., студент кафедри програмних засобів і технологій, Херсонський національний технічний університет, м.Хмельницький, Україна,

Фролова М.Е., старший викладач кафедри загальноосвітніх гуманітарних та природничих дисциплін.

ДОСЛІДЖЕННЯ ВИКОРИСТАННЯ НОРМАЛІЗОВАНИХ ДАНИХ ДЛЯ АНАЛІЗУ ТЕКСТУ

Обробка природної мови (Natural Language Processing (NLP)) з використанням машинного навчання, це тема, яку ще недостатньо вивчена. Обробка природної мови, NLP[1-3] — технологія, яка активно розвивається та допомагає отримати максимальну користь від використання штучного інтелекту. Це та частина використання штучного інтелекту, яка вивчає взаємодію комп'ютерів з людьми з використанням природної мови. Застосування NLP: у програмах для перекладу мов, в текстових процесорах, в програмах інтерактивної голосової відповіді, в персональних програмах-помічниках та ін. Задачі, які вирішуються в області обробки природної мови, це є вирішення проблеми представлення тексту у машинно зрозумілому форматі. Якщо необхідно використання алгоритмів машинного навчання, то є потреба представлення вхідного тексту у числовому просторі так, щоб будь-який текстовий документ з характеристиками фіксованої кількості числових значень, або вектором заданої розмірності [1-3]. За допомогою векторного представлення текстів є можливість вирішувати задачі різної складності: класифікація текстів за тематикою; групування схожих за сенсом документів; ефективне ранжування веб-сторінок у пошукових системах; генерація скорочених текстів зі збереженням сенсу та ін. Методи векторизації текстів, які є, вони незавжди ефективні для високоточного виконання поставлених задач. Пошук більш ефективних методів є актуальною проблемою в галузі обробки природної мови.

Мета дослідження - векторне подання природомовних текстів для використання у системах машинного навчання, що працюють з текстовими даними [1-3].

Результати дослідження: проаналізовані етапи попередньої обробки неструктурованих текстів на природній мові:

- токенизація за реченнями;
- токенизація за словами.

Виконано видалення стоп-слів, від якого залежить якість отриманих токенів.

Було виконане порівняння алгоритмів нормалізації: лематизація має більш точні результати, проте потребує більше часу, що часто є недопустимим у системах, що використовують NLP, тому перевага частіше віддається стемінгу, але потрібно звернути увагу і на недоліки алгоритмів стемінгу (надстемінг і недостемінг). Практичне тестування показало, що для токенизації добре спрацює просте розбиття по пробілах із подальшою нормалізацією токенів. Для стемінгу використано підхід, який відсікає закінчення та суфікси, максимально наближуючи слово до його кореня [1-3].

Аналіз існуючих підходів - алгоритм Портера є оптимальним по результату за складності імплементації та необхідних машинних ресурсів.

ПЕРЕЛІК ПОСИЛАНЬ:

3. Mathangi Sri Practical Natural Language Processing with Python. Apress, 2021. 253 с.
4. Окунькова О.О. Сучасні інформаційні технології аналізу україномовних текстів. Вісник Кременчуцького національного університету імені Михайла Остроградського. Серія: Комп'ютерні науки. Кременчук: КрНУ, 2023. Випуск 1(138). С. 73-79.

5. Що таке обробка природної мови (NLP) та як вона може використовуватися у бізнесі.
URL: <https://metinvest.digital/ua/page/1052>

Тищенко М.О., студент кафедри програмних засобів і технологій, Херсонський національний технічний університет, м.Хмельницький, Україна,

Огнєва О.Є., к.т.н., доцент кафедри програмних засобів і технологій, Херсонський національний технічний університет, м.Хмельницький, Україна.

ВИВЧЕННЯ ТА ЗАСТОСУВАННЯ НОВИХ ПАРАДИГМ І ПІДХОДІВ ДО РОЗРОБКИ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ (НАПРИКЛАД DEVOPS, БЕЗСЕРВЕРНА АРХІТЕКТУРА, МІКРОСЕРВІСИ)

Актуальність вивчення та застосування нових парадигм і підходів до розробки програмного забезпечення полягає в їхній універсальності та ефективності. Вони можуть використовуватися в різних галузях, починаючи від фінансових технологій і закінчуючи медичними системами. Нові парадигми, такі як DevOps, безсерверна архітектура та мікросервіси, розвиваються в нещадному темпі, надаючи більше можливостей для автоматизації рутинних завдань та забезпечення зручності для розробників і кінцевих користувачів [1-4].

Однією з важливих переваг є ефективність цих підходів у вирішенні проблем та наданні інформації. Вони можуть оперативно реагувати на змінні вимоги ринку, забезпечуючи швидку допомогу та сприяючи покращенню якості розробки програмного забезпечення. Наприклад, використання DevOps підходу дозволяє команді швидко виявляти та виправляти помилки через безперервну інтеграцію та доставку (CI/CD), що значно скорочує час випуску продукту.

Унікальність нових парадигм полягає також у їхній здатності адаптуватися до змінних умов і вдосконалювати свої функції, забезпечуючи тривалу та ефективну взаємодію між командами розробників і користувачами. DevOps, наприклад, сприяє інтеграції та співпраці між командами розробників та ІТ-операцій, що дозволяє швидше та ефективніше випускати нові версії програмного забезпечення. Відома компанія Amazon активно використовує DevOps для управління своїми сервісами, що дозволяє їм впроваджувати зміни в систему тисячі разів на день [1-4].

В сучасному технологічному середовищі нові підходи до розробки програмного забезпечення відіграють ключову роль у поліпшенні якості та швидкості розробки. Їхнє застосування не тільки спрощує комунікацію між командами, але й відкриває нові можливості для створення більш масштабованих і надійних систем. Безсерверна архітектура, наприклад, дозволяє розробникам зосередитися на написанні коду, не турбуючись про управління серверами, що значно скорочує час на розробку та впровадження нових функцій. Компанія Netflix широко використовує безсерверні обчислення для обробки даних і масштабування своїх сервісів, що дозволяє їм ефективно обслуговувати мільйони користувачів по всьому світу [1-4].

Однією з важливих переваг нових парадигм в технологічному середовищі є їхня здатність до масштабування і персоналізації. Мікросервіси дозволяють розробникам створювати окремі компоненти, які можуть працювати незалежно один від одного, що сприяє оптимальному розвитку кожного елемента системи з урахуванням його унікальних вимог. Це також дозволяє легко оновлювати та розширювати систему, не впливаючи на її загальну стабільність. Google використовує мікросервіси для управління своїми численними сервісами, що дозволяє їм підтримувати високу продуктивність та надійність навіть при великих навантаженнях [1-4].

Нові парадигми і підходи до розробки програмного забезпечення також можуть бути ефективними інструментами для взаємодії розробників і кінцевих користувачів. Вони можуть надавати своєчасні оновлення, інформацію про нові функції та сприяти активній участі

користувачів у процесі вдосконалення програмного забезпечення. Наприклад, мікросервісна архітектура дозволяє швидко впроваджувати нові функції і отримувати зворотний зв'язок від користувачів, що підвищує задоволеність користувачів і сприяє більш швидкому розвитку продукту [1-4].

Унікальність нових підходів до розробки програмного забезпечення полягає в їхній здатності створювати інноваційні та ефективні рішення для сучасних технологічних викликів. Це може включати в себе використання автоматизації, інтеграцію з хмарними сервісами та інші інноваційні методи, що зроблять процес розробки більш гнучким та продуктивним. Наприклад, використання автоматизації в DevOps дозволяє значно зменшити людські помилки та підвищити ефективність процесів, що є ключовим фактором для успіху в умовах швидко змінюваних ринкових умов.

Так, Amazon активно використовує DevOps для управління своїми сервісами, що дозволяє їм впроваджувати зміни в систему тисячі разів на день. Завдяки безперервній інтеграції та доставці (CI/CD), команда може швидко виявляти та виправляти помилки, що значно скорочує час випуску продукту. Це забезпечує високу якість і швидкість доставки нових функцій та покращень для користувачів [1-4].

Etsy використовує DevOps для автоматизації тестування та розгортання. Завдяки цьому підходу, вони можуть швидко випускати нові функції та забезпечувати стабільність свого інтернет-магазину. Це дозволяє Etsy забезпечувати безперервний потік оновлень без зниження якості обслуговування [1-4].

Netflix використовує безсерверні обчислення для обробки даних і масштабування своїх сервісів. Вони застосовують AWS Lambda для виконання коду у відповідь на події, що дозволяє їм ефективно обробляти великі обсяги даних без необхідності керувати інфраструктурою. Це допомагає знизити витрати та підвищити гнучкість системи [1-4].

Airbnb використовує безсерверні функції для автоматизації рутинних завдань і обробки великих обсягів даних. Наприклад, вони застосовують Google Cloud Functions для автоматичного оброблення фотографій користувачів, що завантажуються на платформу. Це дозволяє зменшити навантаження на сервери та покращити швидкість обробки даних [1-4].

Google використовує мікросервіси для управління своїми численними сервісами. Кожен сервіс працює незалежно, що дозволяє командам розробників швидко впроваджувати нові функції та виправляти помилки без впливу на інші частини системи. Це забезпечує високу продуктивність і надійність сервісів навіть при великих навантаженнях [1-4].

Uber використовує мікросервісну архітектуру для підтримки своїх додатків у реальному часі. Кожен мікросервіс відповідає за окрему функцію, наприклад, управління поїздками, платежами або відстеженням водіїв. Це дозволяє Uber швидко масштабувати свої сервіси та забезпечувати стабільну роботу навіть при пікових навантаженнях [1-4].

Таким чином, вивчення та застосування нових парадигм і підходів до розробки програмного забезпечення, таких як DevOps, безсерверна архітектура та мікросервіси, є ключовими для забезпечення ефективності, швидкості та якості програмних продуктів. Ці підходи дозволяють компаніям швидко реагувати на змінні вимоги ринку, підвищувати продуктивність і знижувати витрати. Приклади провідних компаній, таких як Amazon, Netflix, Google та Uber, демонструють успішне впровадження цих технологій, що сприяє їхньому стабільному зростанню та інноваціям. Використання нових парадигм у розробці програмного забезпечення відкриває нові можливості для автоматизації, масштабування та покращення користувацького досвіду.

ПЕРЕЛІК ПОСИЛАНЬ:

1. AWS Lambda. URL: <https://aws.amazon.com/ru/lambda>
2. Cloud Run functions. URL: <https://cloud.google.com/functions>
3. DevOps и AWS. URL: <https://aws.amazon.com/ru/devops>
4. Your data. Your models. Your choice. URL: <https://www.redhat.com>

Фідіна К.В., студентка кафедри програмних засобів і технологій, Херсонський національний технічний університет, м.Херсон, Україна,
Огнєва О.Є., к.т.н., доцент кафедри програмних засобів і технологій, Херсонський національний технічний університет, м.Херсон, Україна.

РОЗРОБКА ВЕБ-ОРІЄНТОВАНОЇ ІНФОРМАЦІЙНОЇ СИСТЕМИ ДЛЯ ОБСЛУГОВУВАННЯ АБОНЕНТІВ ПРОВАЙДЕРА МЕРЕЖІ ІНТЕРНЕТ

Зростання популярності Інтернет-послуг потребує від провайдерів вдосконалення обслуговування абонентів. Сучасні умови вимагають розробки ефективних інформаційних систем, які автоматизують процеси обслуговування та підвищують рівень задоволеності клієнтів [1-5].

Традиційні системи обслуговування абонентів мають обмежену функціональність та зручність. Наприклад, типовий локальний CRM (Customer Relationship Management) для провайдерів може не включати можливості інтеграції з онлайн-платформами та мобільними додатками, що значно знижує зручність використання для клієнтів та ефективність обслуговування. Такі системи також можуть не підтримувати багатомовний інтерфейс або мати обмежені можливості для налаштування під конкретні потреби провайдера. Окрім цього, такі системи часто вимагають значної кількості ручної роботи для обробки заявок, налаштування нових послуг та оновлення даних, що підвищує ризик помилок і затримок. Веб-орієнтовані системи, що використовують сучасні веб-технології (HTML5, CSS3, JavaScript, MySQL), мають великий потенціал для покращення обслуговування абонентів [1-5].

Розробка веб-орієнтованої інформаційної системи повинна враховувати потреби як абонентів, так і адміністраторів. Система повинна бути інтуїтивно зрозумілою, забезпечувати високий рівень безпеки даних та підтримувати масштабування. Важливо забезпечити сумісність з різними пристроями, включаючи мобільні телефони та планшети. Система повинна забезпечувати можливість автоматичного збирання та аналізу статистичних даних про використання послуг провайдера абонентами. Це дозволить проводити моніторинг якості послуг, виявляти та прогнозувати проблемні ситуації, а також приймати обґрунтовані рішення щодо покращення послуг та їхньої оптимізації. Програмну реалізацію веб-орієнтованої інформаційної системи доцільно здійснювати на базі відкритих стандартів та технологій, що забезпечить її незалежність від конкретних постачальників програмного забезпечення та апаратного забезпечення, а також спростить подальшу підтримку та розвиток системи [1-5].

Подальші дослідження будуть спрямовані на розробку архітектури системи, визначення функціональних модулів, їх інтеграцію та тестування в реальних умовах. Планується експериментальне впровадження системи для оцінки її ефективності та внесення необхідних коригувань.

Основна проблема, яку ставить це дослідження, полягає у розробці ефективної веб-орієнтованої інформаційної системи для обслуговування абонентів провайдера мережі Інтернет. Подальші дослідження зосереджені на вирішенні цієї проблеми через розробку, впровадження та тестування нової системи.

Впровадження автоматизованих систем скоротить час на обробку запитів і знизить ймовірність помилок, що покращить загальний досвід користувачів. Розроблена система буде здатна інтегруватися з існуючими платформами та масштабуватися відповідно до зростаючих потреб провайдера, що забезпечить довготривалу ефективність. Система буде розроблена з урахуванням сучасних вимог до кібербезпеки, що є критично важливим у контексті захисту персональних даних абонентів. Використання новітніх веб-технологій забезпечить високу продуктивність і надійність системи.

ПЕРЕЛІК ПОСИЛАНЬ:

1. Браун М. Впровадження веб-орієнтованих систем в корпоративне середовище. Інформатика і кібернетика. 2021. №4. С. 27-35.
2. Василенко А.В., Коваленко О.Г. Методологічні підходи до створення масштабованих веб-систем. Кібернетика і системний аналіз. 2018. №3. С. 12-20.
3. Іваненко П.В. Інформаційні системи для обслуговування клієнтів. Харків: Технодрук, 2020. 190 с.
4. Мельников, О.Ю., Спориш, Д.К. (2020). Розробка системи для інтелектуального аналізу даних інтернет-провайдера. ВІСНИК Донбаської державної машинобудівної академії, УДК 004.89:519.7
5. Ходоровський М.І., Соловйов В.А. Сучасні підходи до створення інформаційних систем. Інформаційні технології і комп'ютерні системи. 2019. №2. С. 45-52.

Філіпчук В.Р., студент кафедри програмних засобів і технологій, Херсонський національний технічний

університет, м.Херсон, Україна,

Огнєва О.Є., к.т.н., доцент кафедри програмних засобів і технологій, Херсонський національний технічний університет, м.Херсон, Україна.

ДОСЛІДЖЕННЯ ТА РОЗРОБКА ШАБЛОНІВ ПРОЄКТУВАННЯ ДЛЯ СТВОРЕННЯ ВЕБ-ДОДАТКУ "АПТЕКА"

Веб-додатки для аптек забезпечують користувачам зручний доступ до інформації про лікарські засоби, медичні послуги та консультації. Ефективне та швидке створення таких додатків можливе завдяки використанню шаблонів проєктування. В даній роботі проведено дослідження та розробку шаблонів проєктування для створення веб-додатку «Аптека» з використанням сучасних інструментів та технологій.

Метою даного дослідження є вивчення та впровадження шаблонів проєктування для розробки веб-додатку «Аптека», що забезпечить ефективну та якісну розробку програмного забезпечення для медичних закладів.

Дослідження охоплює огляд існуючих шаблонів проєктування, які можуть бути використані для створення веб-додатку «Аптека». Основна увага приділяється розробці оптимального підходу до реалізації додатку на основі вивчених шаблонів, зокрема забезпечення сумісності з різними браузерами та пристроями, а також тестуванню на різних веб-сторінках.

Веб-додаток «Аптека» надає користувачам можливість пошуку та фільтрації лікарських засобів, відображення детальної інформації про кожний препарат, включаючи інструкції та побічні ефекти, а також можливість замовлення ліків онлайн з доставкою. Крім того, додаток включає інтерактивний календар для запису на консультації та отримання нагадувань, а також систему управління користувачами та їх замовленнями.

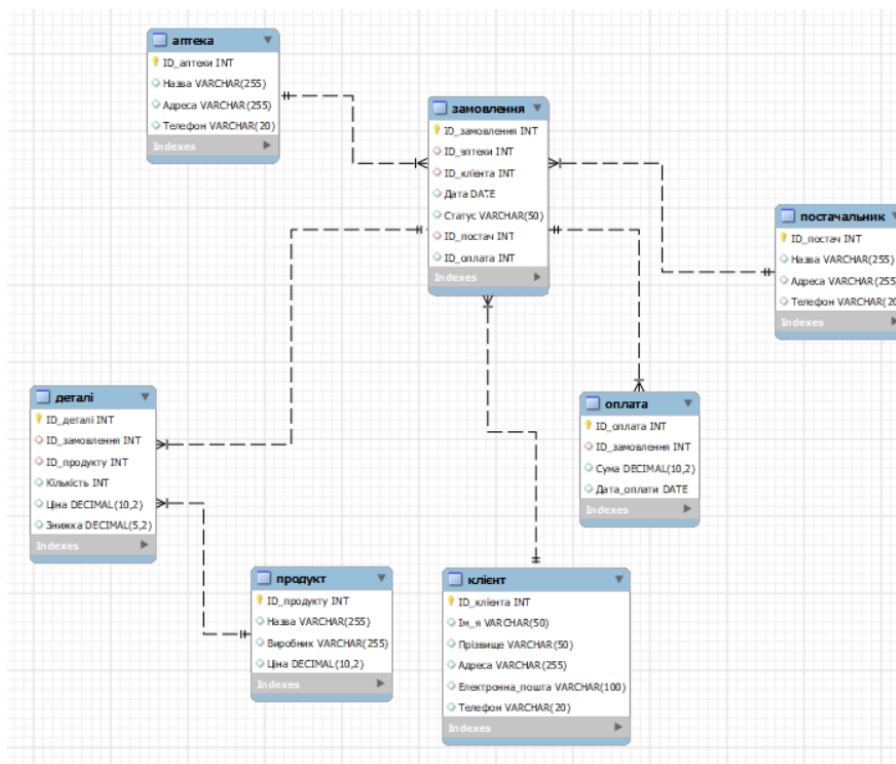


Рисунок 1 - Фізична схема бази даних аптеки

Для розробки веб-додатку «Аптека» використано PHP, HTML, CSS та SQL для бази даних. Готовий веб-додаток містить велику кількість компонентів, які ієрархічно вкладаються одна в одну та відповідають за певний функціонал чи частину сторінки [1-2].

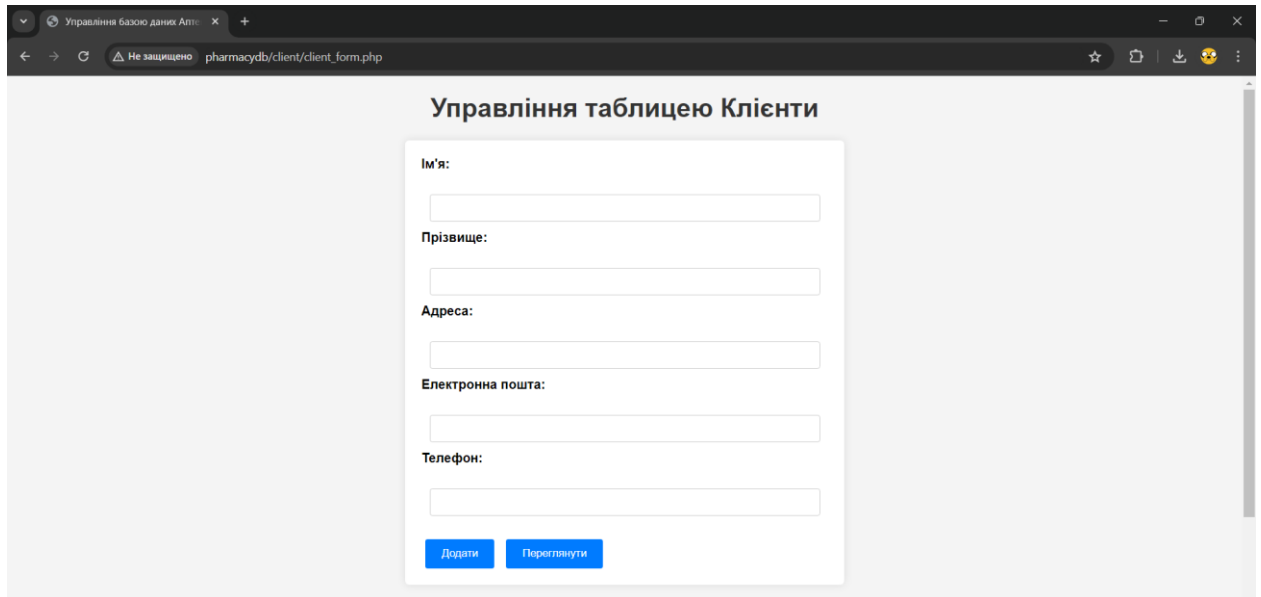


Рисунок 2 - Заповнення таблиці Клієнта

Результати проведеного дослідження показали, що розроблені шаблони проєктування для створення веб-додатку «Аптека» дозволяють ефективно та швидко розробляти функціональні компоненти додатку з можливістю їх подальшого розширення та модифікації. Розроблений продукт може бути використаний для створення веб-додатків різного типу та рівня складності, що є важливим для веб-розробників та їх замовників. Використання шаблонів проєктування є ключовим фактором для ефективної та якісної розробки програмного забезпечення, включаючи веб-додатки.

ПЕРЕЛІК ПОСИЛАНЬ:

1. Дерід І.О., Ночвіна І.О. Розвиток електронної торгівлі в Україні. 2021. № 56. с. 3-5. URL: http://market-infr.od.ua/journals/2021/56_2021/4.pdf.
2. Стась О. Ф., Оніщук О. В. Інформаційні системи та технології в торгівлі: Навчальний посібник. К.: Центр навчальної літератури, 2018. 360 с.

Шатровський В.О., студент кафедри програмних засобів і технологій, Херсонський національний технічний університет, м.Хмельницький, Україна,
Фролова М.Е., старший викладач кафедри загальноосвітніх гуманітарних та природничих дисциплін.

ВИКОРИСТАННЯ НЕЙРОННИХ МЕРЕЖ ДЛЯ СТВОРЕННЯ ТОРГОВИХ АЛГОРИТМІВ НА КРИПТОВАЛЮТНИХ РИНКАХ

Стрімкий розвиток криптовалют за останні 12 років є одним з найбільш суперечливих нововведень в сучасній світовій економіці. Численні непередбачувані коливання обмінного курсу криптовалют, а також відсутність розумного та належного управління операціями з цим типом валюти у більшості країн, що розвиваються, та користувачів цього типу валюти збільшили ризик та недовіру до криптовалют серед інвесторів. Інвестори вважають за краще вкладати кошти в програми, які мають найменший ризик і найкоротші терміни отримання максимальної вигоди і основного прибутку. Тому питання розробки відповідних методів і моделей прогнозування ціни криптовалюти актуальне як для наукового співтовариства, так і для фінансових аналітиків, інвесторів і трейдерів.

З метою покращення точності та ефективності прогнозування цін на криптовалюту або прогнозування потенційної зміни тренду виникає необхідність у використанні нейронних мереж для навчання та вирішення безлічі можливих варіантів на основі величезних об'ємів даних.

Метою цього дослідницького проєкту є розробка нейронної моделі, яка використовує великі обсяги даних для прогнозування того, куди будуть рухатися ціни відповідно до майбутніх тенденцій або змінять свій напрямок у зворотному напрямку.

Основною метою дослідження в цьому напрямку є створення надійної і точної моделі, яка може аналізувати великі обсяги даних з криптовалютних бірж, включаючи такі фактори, як графіки свічок та дані індикаторів на основі свічного аналізу.

Використання методів створення нейронних мереж при розробці системи зробить прогнозування більш об'єктивним і точним. Система використовує різні алгоритми, такі як RNN та LSTM, для вивчення існуючих історичних даних з криптобіржі Binance.

Після навчання розроблена модель готова передбачити потенційний розворот цінового тренду в новому наборі даних. Результати цього дослідження і розробки мають велике значення для ринку криптовалют і допоможуть різним зацікавленим сторонам прийняти рішення про покупку або продаж активів за вигідною ціною. Крім того, впровадження системи може сприяти підвищенню ефективності та зниженню ризиків на волатильному ринку криптовалют.

У цьому дослідницькому проєкті були розглянуті основні методи створення нейронних мереж, для єдиної та найефективнішої моделі прогнозування потенційного розвороту цінового тренду, а також проаналізовані наявні історичні дані найкращих криптовалют по капіталізації для навчання та перевірки моделі. Окрім того, були проведені експерименти для порівняння різних моделей та оцінки їхньої прогностичної здатності.

Наукова новизна проведених досліджень: розроблено систему, що дозволяє точно прогнозувати зміну тренду ціни для криптовалютних пар для ринку криптовалют.

Розробка нейронної моделі включає в себе весь процес створення та підготовки моделі, яка здатна вирішувати поставлену задачу у прогнозуванні поведінки ціни на криптовалютному ринку за допомогою різних алгоритмів.

Першим етапом стало розбиття отриманих даних на тренувальні та тестувальні дані. Тренувальних даних - 80%, тестувальних - буде 20%. На їх основі модель навчалася та покращувалася.

```

1 usage  ▴ Vladyslav Shatrovskyi
52 def transform_train_set_to_tensor(self):
53     """
54     Transform the training set to PyTorch tensors.
55     """
56     features = self.train_data[self.feature_cols]
57     target = self.train_data[self.target_col]
58
59     # Standardize the feature columns
60     features = self.scaler.fit_transform(features)
61
62     # Convert to tensors
63     x_train = torch.tensor(features, dtype=torch.float32)
64
65     y_train_class = target.replace({0: 0, 1: 1, -1: 2})
66     y_train = torch.tensor(y_train_class.values, dtype=torch.long)
67
68     return x_train, y_train
69
1 usage  ▴ Vladyslav Shatrovskyi
70 def transform_test_set_to_tensor(self):
71     """
72     Transform the test set to PyTorch tensors.
73     """
74     features = self.test_data[self.feature_cols]
75     target = self.test_data[self.target_col]
76
77     # Standardize the feature columns using the scaler fitted on the train set
78     features = self.scaler.transform(features)
79
80     # Convert to tensors
81     x_test = torch.tensor(features, dtype=torch.float32)
82
83     y_test_class = target.replace({0: 0, 1: 1, -1: 2})
84     y_test = torch.tensor(y_test_class.values, dtype=torch.long)
85
86     return x_test, y_test
87

```

Рисунок 1 - Реалізація двох методів для отримання тестового та тренувального набору даних

Наступним кроком стала розробка простої нейронної моделі RNN на базі бібліотеки PyTorch.

Нейронна модель має кілька важливих параметрів, кожен з яких відіграє ключову роль у роботі моделі [1-4]:

input_size: кількість вхідних особливостей на кожному кроці послідовності. Вказує на кількість індикаторів, що використовуються для навчання моделі;

hidden_size: кількість нейронів у прихованому шарі RNN. Визначає здатність моделі запам'ятовувати та аналізувати послідовності. Більше значення означає більшу здатність до вивчення складних залежностей, але може призвести до збільшення обчислювальної складності;

num_layers: кількість шарів RNN у моделі. Збільшує глибину моделі, дозволяючи їй вивчати більш складні патерни. Однак занадто велике значення може спричинити перенавчання або збільшити час навчання;

output_size: Кількість вихідних нейронів моделі, в нашому випадку це -1, 0 та 1. Визначає розмір вихідних даних, які моделі потрібно передбачити;

dropout_rate: вірогідність відключення нейронів під час навчання для запобігання перенавчанню. Використовується для регуляризації моделі, щоб уникнути надмірного підлаштування під навчальні дані. Значення 0.6 означає, що 60% нейронів прихованого шару випадково вимикаються під час навчання.

Ці параметри налаштовуються відповідно до складності задачі та обсягу доступних даних для досягнення оптимальної точності моделі.

Наступним кроком було розроблено алгоритм за допомогою якого всі частини програми могли працювати разом та модель мала можливість навчатися, а її результати зберігалися в CSV форматі та в графічному варіанті.

Після завершення етапу навчання та валідації, було проведено тестування моделі на тестовому наборі даних. Мета цього етапу - перевірити, як модель поводить себе на реальних невідомих даних, оцінити її точність, стабільність і здатність до узагальнення.

Аналіз результатів трьох моделей (LSTM, RNN, BiLSTM) на різних криптовалютних парах показав варіативність у значеннях F1 Score та Accurasy. Моделі BiLSTM зазвичай показують найвищі значення F1 Score і Accurasy, що вказує на їх здатність краще виявляти шаблони в даних і давати точніші прогнози. Наприклад, модель BiLSTM для пар ETHUSDT та DOGEUSDT демонструє найкращу продуктивність з найвищими значеннями як F1 Score, так і Accurasy.

У більшості випадків існує висока кореляція між F1 Score і Accurasy, але оцінка F1 Score важлива для оцінки ефективності моделі на основі фактичних торгових даних.

Моделі BiLSTM і LSTM часто демонструють високі результати на деяких парах, таких як DOGEUSDT, де обидва показники досягають значення більше 0.75 для F1 Score і більше 0.97 для Accurasy. Моделі RNN зазвичай мають набагато гірші результати, знижуючи продуктивність порівняно з іншими моделями.

Різні криптовалютні пари мають різну складність для моделей. Наприклад, DOGEUSDT і ADAUSDT показують високі результати для всіх моделей, що свідчить про більш стабільну поведінку цих активів на основі наданих даних. З іншого боку, символи типу XRPUSDT або TONUSDT мають менші значення F1 Score, що може свідчити про складніші або менш передбачувані поведінки ціни.

Таким чином, результати показують, що BiLSTM є найкращою моделлю для передбачення поведінки цін на криптовалютних даних, хоча для деяких символів інші моделі також можуть показувати хороші результати.

ПЕРЕЛІК ПОСИЛАНЬ:

1. Matplotlib Documentation. URL: <https://matplotlib.org/stable/index.html>
2. Mimmo Parente, Luca Rizzuti, Mario Trerotola. A profitable trading algorithm for cryptocurrencies using a Neural Network model. 2023. URL: <https://www.sciencedirect.com/science/article/pii/S0957417423023084>
3. Understanding the crypto-asset phenomenon, its risks and measurement issues. 2019. URL: https://www.ecb.europa.eu/press/economic-bulletin/articles/2019/html/ecb.ebart201905_03~c83aeaa44c.en.html#toc2
4. Xihan Xiong. Junliang Luo. 2024. Global Trends in Cryptocurrency Regulation URL: <https://arxiv.org/html/2404.15895v2>

Шишов М.О., студент кафедри інженерії програмного забезпечення, Херсонський національний технічний університет, м.Хмельницький, Україна,

Огнєва О.Є., к.т.н., доцент кафедри програмних засобів і технологій, Херсонський національний технічний університет, м.Хмельницький, Україна.

ДОСЛІДЖЕННЯ ОПТИМІЗАЦІЇ ПРОДУКТИВНОСТІ ТА ШВИДКОДІЇ ВЕБ-САЙТІВ ЗА ДОПОМОГОЮ ФРЕЙМВОРКУ ЛАРАВЕЛ

Оптимізація продуктивності та швидкодії веб-сайтів – це важлива та актуальна проблема в сучасному інтернет-середовищі, яке визначається високою конкуренцією та вимогливістю користувачів. Досягнення оптимального рівня продуктивності та швидкодії стає ключовим фактором для успіху веб-сайтів, впливаючи на їхню популярність, задоволеність користувачів та конверсію.

Затримки у завантаженні сторінок можуть призводити до негативного враження користувачів, зменшення рівня задоволеності та виходу з веб-сайту. Оптимізований веб-сайт, який працює швидко, підвищує користувацький комфорт та задоволення.

Алгоритми пошукових систем, зокрема Google, враховують швидкість завантаження сторінок при ранжуванні результатів пошуку. Швидкий веб-сайт має більше шансів потрапити в топові позиції, що важливо для забезпечення високої видимості.

Оптимізовані веб-сайти сприяють покращенню конверсії, оскільки користувачі швидше здійснюють бажані дії, такі як покупка товарів чи замовлення послуг. Зменшення часу очікування може значно підвищити впізнаваність бренду та лояльність клієнтів.

Швидкодія важлива для успішного впровадження рекламних кампаній та маркетингових стратегій. Рекламні матеріали краще сприймаються, а результативність кампаній збільшується на швидких веб-сайтах. Оптимізація дозволяє ефективніше використовувати ресурси серверів, що може призвести до зменшення витрат на інфраструктуру та обслуговування, забезпечуючи економію ресурсів.

Згідно з метою під час дослідження було проведено аналіз фреймворку Ларавел, його функціоналу який поліпшує продуктивності та швидкодії веб-сайтів [1-2].

Laravel — це міжплатформовий PHP-фреймворк для створення веб-додатків з використанням архітектурної моделі MVC, що має багато переваг у використанні, оскільки дозволяє розробнику скористатися можливостями великої бібліотеки попередньо запрограмованих функцій (таких як автентифікація, маршрутизація та створення шаблонів HTML). Доступ до цієї бібліотеки спрощує швидке створення надійних веб-додатків, зводячи до мінімуму кількість необхідного коду.

MVC – «модель-представлення-контролер» (від англ. model-view-controller). Це спосіб організації коду, який передбачає виділення блоків, відповідальних рішення різних завдань. Один блок відповідає за дані програми, інший відповідає за зовнішній вигляд, а третій контролює роботу програми (рис. 1).

Laravel пропонує високофункціональне середовище розробки, а також інтуїтивно зрозумілі та виразні інтерфейси командного рядка. Крім того, Laravel використовує об'єктно-реляційне відображення (ORM) для спрощення доступу до даних і маніпулювання ними.

Ряд вбудованих інструментів відкриває широкі можливості для розробників у створенні ефективних та продуктивних веб-додатків.

Використання Eloquent ORM є надзвичайно корисним для простоти та ефективності роботи з базою даних. Можливість визначення відносин та зручний синтаксис дозволяють розробникам зосередитися на логіці додатку, забезпечуючи високий рівень абстракції від бази даних.

Blade шаблонізатор представляє з себе ефективний інструмент для створення зручного та читабельного коду у представленнях. Його можливості компактного виразного синтаксису дозволяють розробникам швидко створювати динамічний контент та зберігати чистоту коду.

Механізми кешування в Laravel, такі як кешування запитів та кешування видів, грають важливу роль у поліпшенні швидкодії. Це дозволяє зберігати результати важких операцій та уникати повторного їх виконання, зменшуючи час завантаження сторінок [1-2].

Використання пагінації в Laravel стає ключовим елементом у роботі з великими обсягами даних. Механізм пагінації дозволяє розбити великі результати запитів на менші частини, що полегшує роботу з великими колекціями даних та поліпшує швидкість завантаження сторінок [1-2].

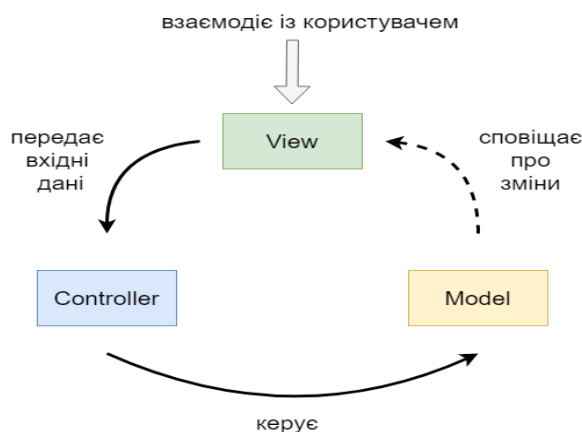


Рисунок 1 – Робота MVC архітектури

Загально кажучи, Laravel надає розробникам ряд ефективних інструментів, які сприяють оптимізації продуктивності та швидкодії веб-додатків. Використання цих можливостей може значно спростити розробку та покращити користувацький досвід, забезпечуючи високу швидкість та ефективність.

ПЕРЕЛІК ПОСИЛАНЬ:

1. Що таке Laravel? URL: <https://probotiuk.com/blog/sho-take-laravel>
2. Laravel. URL: . <https://laravel.com/docs/8.x>

Юдін Д.Ф., студент кафедри програмних засобів і технологій, Херсонський національний технічний університет, м.Хмельницький, Україна,

Захарченко Р.М., к.т.н., доцент кафедри програмних засобів і технологій, Херсонський національний технічний університет, м.Хмельницький, Україна.

ВИКОРИСТАННЯ ІТ В РОЗРОБЦІ ФУЛСТЕК ВЕБ-ДОДАТКУ ДЛЯ ПРОДАЖУ ЕЛЕКТРОНІКИ НА ФРЕЙМВОРКУ REACT ТА NODEJS

У сучасному інформаційному суспільстві, де швидкість і простота здійснення покупок набувають все більшого значення, електронна комерція стає справжнім рушієм економічного розвитку, виходячи за рамки простого інструменту купівлі. Швидкість доступу до товарів і послуг, різноманітність продукції, легкість порівняння цін і простота оплати роблять електронну комерцію дуже привабливою для сучасного споживача [1-3].

Проведено дослідження ролі веб-додатків для електронної комерції. Ці додатки не тільки відображають тенденції розвитку сучасних технологій, але й створюють нові можливості для покупців зручно та швидко отримувати новітню електронну техніку.

Фреймворки React та NodeJS для розробки веб-додатків відомі своєю ефективністю та гнучкістю, за їх допомогою розробка стала ефективнішою та швидшою, а поєднання їхньої потужності з функціональністю електронної комерції відкрило шлях до створення високоефективних інструментів електронних продажів, які задовольняють навіть найвибагливіших споживачів.

Метою цієї роботи було не лише розробка функціонального веб-додатку, але й вивчення сучасних тенденцій та аналіз їхнього впливу на електронну комерцію. Було розібрано математичні аспекти та деталі програмного додатку, щоб гарантувати високу якість та ефективність продукту.

Таким чином, отримані результати будуть не тільки сприяти подальшому розвитку електронної комерції, але й нададуть споживачам нові можливості для зручних та ефективних онлайн-покупок електронних пристроїв, що відповідають потребам сучасного ринку.

На рис.1 – 6 представлено екранні форми розробленого веб-додатку для

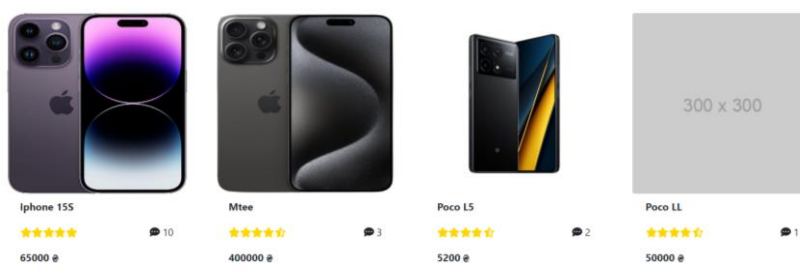


Рисунок 1 - Головна сторінка сайту

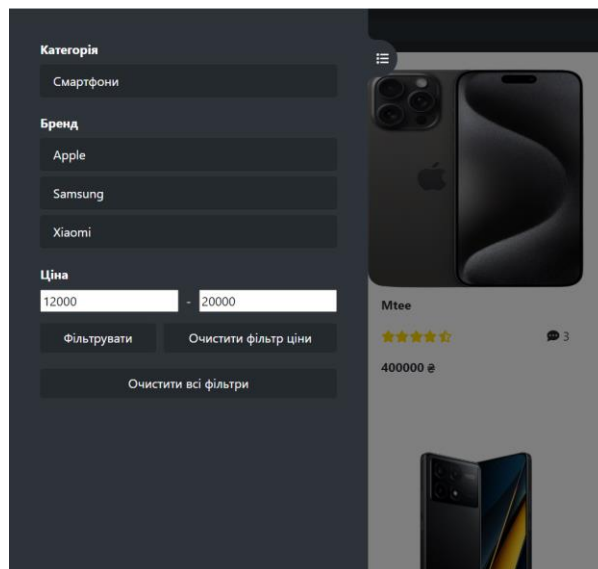


Рисунок 2 - Фільтр товарів

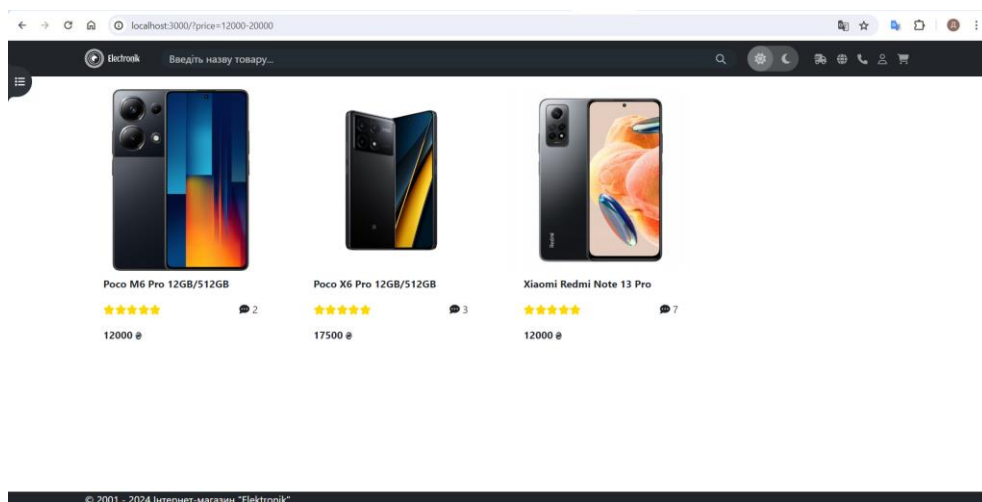


Рисунок 3 - Відфільтровані товари по ціні

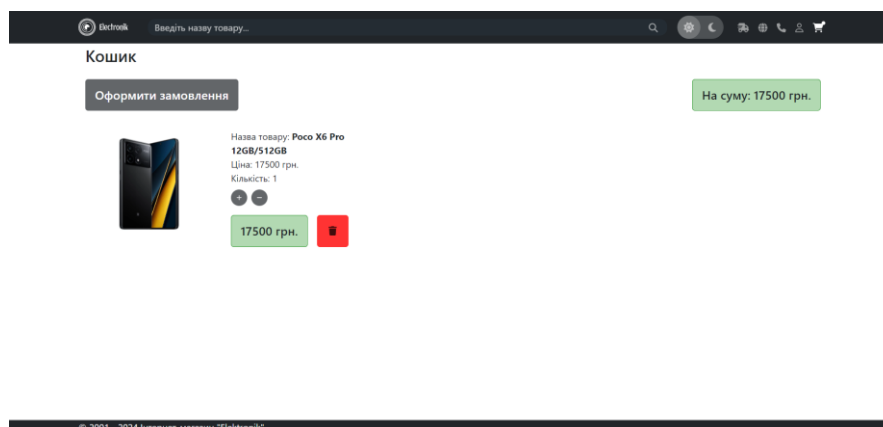


Рисунок 4 - Кошик

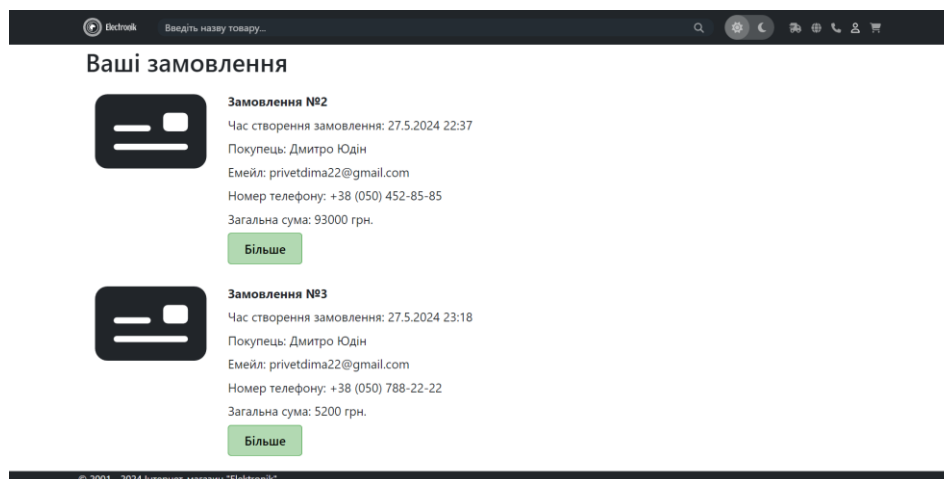


Рисунок 5 - Замовлення користувача

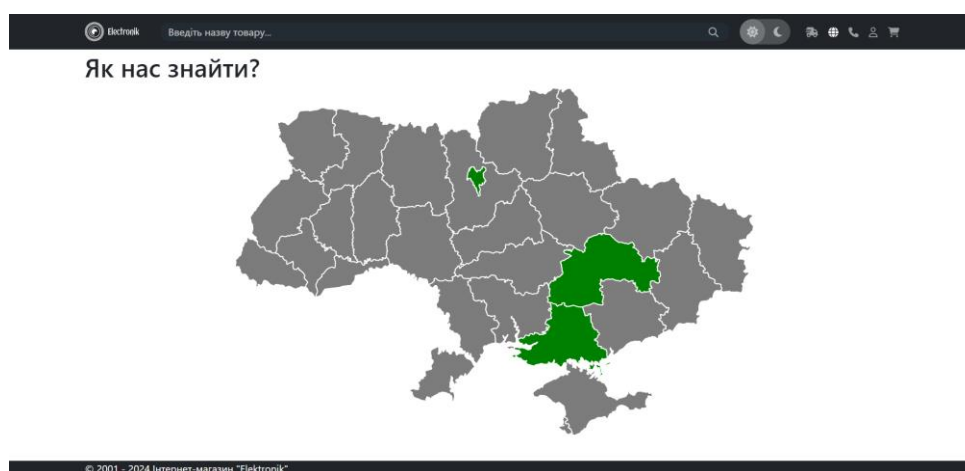


Рисунок 6 - Як нас знайти

За результатами дослідження використання ІТ для створення веб-додатку, продемонстровано високий рівень функціональності і зручності користувача до якого потрібно було дійти. Використання архітектурних шаблонів, моделей баз даних та захисту даних сприяє створенню надійної та безпечної системи електронної комерції. В роботі підтверджується актуальність і перспективи розробки веб-додатків для електронної комерції, ефективного і конкурентоспроможного продукту.

ПЕРЕЛІК ПОСИЛАНЬ:

1. Дерід І.О., Ночвіна І.О. Розвиток електронної торгівлі в Україні. 2021. № 56. с. 3-5.
URL: http://market-infr.od.ua/journals/2021/56_2021/4.pdf.
2. Козицька Г.В. Тенденції та прогнозування розвитку електронної торгівлі в Україні. Вісник Національного університету «Запорізька політехніка». 2021. № 33. с. 1-8.
URL: http://easterneurope-cbm.in.ua/journal/33_2021/18.pdf.
3. Стась О. Ф., Оніщук О. В. Інформаційні системи та технології в торгівлі: Навчальний посібник. К.: Центр навчальної літератури, 2018. 360 с.

Яркова Є.Д., студентка кафедри програмних засобів і технологій, Херсонський національний технічний університет, м.Хмельницький, Україна,

Величко Ю.І., к.т.н., доцент кафедри програмних засобів і технологій, Херсонський національний технічний університет, м.Хмельницький, Україна.

РОЗРОБКА ВЕБ-САЙТУ «ВЕБ-ЛІКАРНЯ»

Актуальність розробки. Актуальність створення веб-сайту для лікарні полягає в зростаючій потребі в цифрових рішеннях для медичних закладів. В умовах сучасних реалій, веб-сайт стає важливим інструментом для залучення пацієнтів, надання інформації та автоматизації процесів обслуговування. Це особливо важливо в умовах пандемій, коли дистанційні медичні консультації та онлайн-сервіси стають необхідністю.

Тематика дослідження. Тематикою даного дослідження є створення веб-орієнтованої інформаційної системи для автоматизації процесів обслуговування пацієнтів лікарні. Завдання включає розробку клієнтської частини веб-застосунку, яка буде взаємодіяти з серверною частиною, забезпечуючи пацієнтам можливість реєстрації, запису на прийом до лікаря, отримання результатів аналізів та інших медичних документів, а також перегляду історії захворювань.

Запис на консультацію

Для запису на прийом з лікарем необхідно зателефонувати в нашу реєстратуру або відвідати нашу поліклініку особисто. Для зручності, ми також надаємо можливість запису на прийом через наш веб-сайт.

ПІБ:

Телефон:

Email:

Виберіть лікаря

Виберіть лікаря

День прийому:

Виберіть день навілі

Час прийому:

Виберіть час відвідування

Відправити

Консультації на цей тиждень

Ім'я	Лікар	День	Час
------	-------	------	-----

Рисунок 1 - Приклад онлайн запису до лікаря

Предметною областю проєкту є лікарня, яка надає широкий спектр медичних послуг, включаючи консультації, діагностику, лікування та реабілітацію. Лікарня прагне мати веб-сайт, який дозволить їй представити свої послуги в інтернеті, залучити нових пацієнтів та покращити якість обслуговування завдяки автоматизації процесів.

Метою дослідження теми розробки веб-сайту є розширення знань у сфері розробки веб-орієнтованих застосунків та набуття досвіду в застосуванні сучасних методів створення фронтенд-частини веб-сайтів за допомогою JavaScript та інших технологій, таких як PHP [1-3].

Практична мета проєкту полягає у реалізації функціонального та зручного фронтенду для веб-орієнтованої інформаційної системи, яка буде використовуватися лікарнею для автоматизації обслуговування пацієнтів. Така система допоможе покращити ефективність роботи медичного закладу, зменшити навантаження на персонал та задовольнити потреби пацієнтів.

При розробці веб-сайту використовувались теоретичні та практичні знання по блоку професійних дисциплін, а також навички розробки веб-застосунків, використовуючи сучасні технології, такі як PHP, JavaScript, CSS та інші. Під час розробки веб-додатку були використані різні методології розробки, методи аналізу, шаблони проєктування та алгоритми [1-3].



Рисунок 2 - Приклад простого онлайн веб-сайту

Знання та навички. Під час роботи над проєктом було використано знання:

- Як створювати веб-застосунки на PHP;
- Як програмувати на мові JavaScript;
- Як налагоджувати помилковий код;
- Як розв'язувати поставлені задачі, використовуючи обґрунтовані методики.

Необхідність створення веб-сайту. Необхідність створення веб-сайту для лікарні полягає у наступних причинах:

- Веб-сайт дозволить лікарні презентувати свої послуги в інтернеті, що збільшить її відомість та привабливість для потенційних пацієнтів;
- Веб-сайт дозволить лікарні зручно та оперативно приймати та обробляти заявки пацієнтів, що покращить якість обслуговування та зменшить навантаження на персонал;

- Веб-сайт дозволить лікарні ефективно взаємодіяти з пацієнтами, надавати їм інформацію про послуги, ціни, акції, оплату, консультації та інше, що підвищить задоволення та лояльність клієнтів;
- Веб-сайт дозволить лікарні адаптуватися до сучасних тенденцій розвитку ринку та конкурентного середовища, де веб-присутність є необхідною умовою успішного бізнесу.

Таким чином, проєкт створення веб-сайту для лікарні є актуальним та необхідним для покращення ефективності роботи медичного закладу, збільшення його дохідності та конкурентоспроможності.

ПЕРЕЛІК ПОСИЛАНЬ:

1. PHP 8.3. URL: <https://www.php.net/releases/8.3/uk.php>
2. Freeman, Adam. (2020). Pro ASP.NET Core 3: Develop Cloud-Ready Web Applications Using MVC, Blazor, and Razor Pages. 10.1007/978-1-4842-5440-0.
3. Durano, Vincent Maverick, Price, Ed, Chilberto, Jeffrey, Helland, Andreas. ASP. NET Core 5 for Beginners: Kick-Start Your ASP. NET Web Development Journey with the Help of Step-by-step Tutorials and Examples. Published by Packt Publishing, Limited, 2020. ISBN 10: 1800567189 / ISBN 13: 9781800567184

ВИМОГИ ДО ОФОРМЛЕННЯ МАТЕРІАЛІВ

Для включення у програму конференції розглядаються заявки і тези доповідей, отримані в електронному вигляді оргкомітетом до **7 травня 2023 року включно**.

1. Обсяг - до 3 стор., текст друкувати у редакторі *Microsoft Word*, без нумерації сторінок, шрифтом «Times New Roman», кеглем 12 pt через один інтервал без ущільнення тексту та переносів, поля - 20 мм з усіх боків, абзац – 10мм.

2. На початку тез в правому куті вказується прізвище та ініціали авторів, місце навчання або роботи, науковий керівник, після чого за вирівнюванням по центру вказується назва.

3. В тексті не повинно бути шрифтових виділень. Рисунки, діаграми, схеми, таблиці мають бути тільки чорного кольору.

4. Рисунки повинні бути згруповані в один графічний об'єкт; формули слід друкувати за допомогою редактора формул *Microsoft Equation* і нумерувати у круглих дужках, наприклад, (2).

5. Перелік посилань слід навести у відповідності до вимог Держстандарту 7.1:2006 кеглем 12 pt. Допускається не більше 5 посилань.

Робочі мови конференції: українська, англійська.