

РОЗРОБКА ВЕБДОДАТКУ ДЛЯ ЕЛЕКТРОННОЇ КОМЕРЦІЇ

У даній статті проаналізований процес проектування та побудові вебдодатку, який працює за так званою моделлю маркетплейсу, тобто це суб'єкт електронної комерції, який зазвичай не продає товари самостійно, але надає майданчик для приватних підприємців, компанії та приватних осіб для систематичного продажу товарів чи послуг. Ці програмні комплекси відрізняються нестандартними для електронної комерції групами користувачів (не тільки покупці в якості клієнтів, але і покупці, і постачальники послуг) та специфічною манерою взаємодії між користувачами, які виступають у різних ролях, таким чином ці проекти реалізують бізнес-модель *Customer-to-Customer*, вирішуючи задачі посередництва, реклами, а також безпеки та арбітражу конфліктів. Програмних рішень такого типу на ринку вдосталь, але майже всі вони мають надлишковий функціонал, складну систему модерації, а найбільші ще й досить активне і конкурентне середовище. Програмний продукт такого типу має особливу архітектуру, тому що йому доводиться вирішувати різноманітні задачі, частина яких виходить за рамки простих угод купівлі-продажу та транзакції грошей від покупця до продавця. Насамперед, платформа повинна забезпечити взаємодію, безпеку даних, переводів, захист від фішингу, розв'язання конфліктів. Функціонал таких проектів екстенсивно зростає, що, з одного боку, дозволяє залучити більше користувачів із різними потребами, з іншого ж – суттєво ускладнює взаємодію, навіть може робити процес продажів та купівлі неефективним і непрозорим. Мінімалістичний підхід до проектування дозволив зупинитись на головному функціоналі і дослідити на цьому прикладі основні риси даного типу програмного забезпечення. В якості програмного стеку було обрано PHP+Node.JS+MySQL у серверній частині (бекенді) та JavaScript у реалізації фреймворку Vue у якості браузерного. На бекенді для пришвидшення розробки використано фреймворк Laravel, який дозволяє вирішити більшість рутинних задач і створити якісний та стабільний проект з використанням підходу REST API.

Ключові слова: маркетплейс, онлайн-маркет, електронна комерція, архітектура ПЗ, PHP, MySQL, Laravel, Node.js, Vue.

O.A. Tuzenko, S.I. Volodin. E-commerce application development. In the current article a process of designing and building a web-application was researched. The application works as a marketplace, it is a subject of e-commerce which usually does not sell goods or provide services itself, it provides a platform for entrepreneurs, companies and individuals to systematically sell goods, promote and provide services. This software packages use unusual for e-commerce user groups (here not only customers are users, but also merchants and service providers) and a very specific interaction model, which have a different system role, thereby these projects embody business model *Customer-to-Customer* by solving intermediation and advertisement tasks as well as different tasks related to security and conflicts arbitration. There are a lot of development solutions like this on the market, but most of them have excess feature set, and the biggest ones in addition provide an extremely competitive environment. The marketplace applications have specific architecture because they are to solve various problems and tasks, which are partially out of simple buy-sell deals and funds transactions from the buyer to the merchant. First of all, this platform must provide an environment for interaction, data security, secure transactions, phishing

¹ канд. техн. наук, доцент, ДВНЗ «Приазовський державний технічний університет», м. Дніпро, ORCID: 0000-0002-4920-9417, tuzenkoaa@gmail.com

² ст. викладач, ДВНЗ «Приазовський державний технічний університет», м. Дніпро, volodinsi57@gmail.com

protection, conflict solving framework. Number of functions in the projects of this kind grows extensively, which from one side allows to involve more users with different needs, from other hand it significantly complexes the interaction, and could even make the selling and buying non-effective and non-obvious. A minimalistic approach to the design allowed us to focus on main program features and investigate on this example the main features and principles of current software type. As a main programming stack, a set including PHP+Node.js+MySQL on server side was selected, as a frontend Javascript-based framework Vue was used. On the backend for the development improvement a PHP-framework Laravel was used, which allows to solve most of routine tasks and rather easily create a quality and stable project using REST API approach.

Keywords: marketplace, online-market, e-commerce, software architecture, PHP, MySQL, Laravel, Node.js, Vue.

Постановка проблеми. Електронна комерція росте швидкими темпами кожного року, зростання продажів у різних сферах торгівлі значуще навіть у ті місяці і роки, у яких загалом були кризи в економіці. Широке розповсюдження продаж товарів через Інтернет почалось на початку 2010 року, але справжнім стрибком у рості комерції став 2020 рік, коли торгівля була паралізована через карантини у наслідок пандемії коронавірусу. Розвиток онлайн-торгівлі пов'язаний з тим, що віртуальні магазини мають загалом менше географічних обмежень та дозволяють продавцям залучати нових клієнтів з усього світу. Крім того, інтернет-магазини є зручним та швидким способом здійснення покупок, що дозволяє покупцям зекономити свій час та зручно вибрати необхідний товар чи послугу. Деякі компанії не продають товари самі, але надають продавцям можливість користуватись програмними засобами, додатками і сайтами як платформами для продажу, вони зазвичай називаються онлайн-маркетом. Онлайн-маркет є специфічним типом сайту, тому що він надає одразу послуги декільком групам користувачів, які відіграють у роботі сайту різні ролі (продавці, покупці) і повинні мати різні рівні доступу і права. Розробка онлайн-маркету є складним та багатоетапним процесом, який потребує високих технічних знань та досвіду веброзробки, у зв'язку з чим дана тема є актуальною, а дослідження її є раціональним, не дивлячись на те, що існує досить велика кількість матеріалів на цю тему. Функціонування платформ з онлайн-продажів зазнає змін дуже швидко, крім того, цей процес можна розглядати з різних точок зору.

Аналіз останніх досліджень і публікацій. Для успішної розробки онлайн-маркету необхідно вивчити проблеми, пов'язані з проектуванням структури бази даних, розробкою функціоналу та інтерфейсу вебдодатку, тестуванням, налаштуванням, та обрати інструменти.

Онлайн-маркет, або маркетплейс – це віртуальна (онлайн) платформа, яка пов'язує між собою продавців і покупців, де вони зустрічаються, щоб обмінюватись, продавати, купувати та просувати товари, віртуальні активи або послуги. Тобто розробники маркетплейсу надають інструменти для маркетингу та власне продажів товарів і місце, де цей продаж можна здійснити. Але однієї платформи зазвичай замало. Коли на одній платформі зустрічаються люди, в яких є економічна зацікавленість у взаємодії, потрібно як мінімум врегульовувати конфлікти, надавати правила взаємодії і поміч у технічній частині. Тому задача побудови саме маркетплейсу виявляється однією з найскладніших і найбільш багатопланових задач у сфері електронної комерції.

Для ефективного функціонування маркетплейсу він повинен відповідати декільком основним вимогам:

1. Забезпечувати механізми залучення трафіку, а отже залучення нових покупців до платформи.
2. Мати зрозумілий та функціональний каталог продукції, який допоможе покупцям зорієнтуватись в асортименті.
3. Дотримуватися кодексу правил, які є чи не найважливішою частиною платформи поза суто технічними інструментами і який дозволяє регулювати відносини між продавцями і клієнтами.
4. Мати прозору платіжну систему і можливість здійснення безпечних покупок, що дозволить клієнту повернутись знов без небезпеки бути ошуканим [1].

Електронний маркетплейс, на якому зустрічаються представники бізнесу або приватні особи, які пропонують певні товари або послуги, і користувачі цих товарів і послуг, повинен забезпечувати наступні дії для процесу торгівлі:

1. Пропонувати товари і послуги у структурованій манері, крім того дозволяти знаходити і обирати з запропонованих варіантів товари і послуги.
2. Мати змогу домовлятися про ціну і умови угоди.
3. Укладати угоди про продаж певних товарів і послуг.
4. Оплачувати товари.
5. Організовувати доставку куплених товарів чи послуг за умовами угод [2].

Якщо будувати онлайн-маркет з нуля, треба розуміти, що даний проект буде розвиватись у надконкурентному середовищі, в якому працюють гіганти міжнародного ринку на кшталт Amazon [3], eBay [4], Aliexpress [5], Etsy [6], тому зайняти будь-яке місце у ряду маркетплейсів такого типу майже не є можливим. Але більш вузькоспеціалізовані маркетплейси часто з'являються у вигляді стартапів і мають успіх, або якнайменше займають свою особливу нішу. Дослідженню онлайн-маркетів як явища займаються як маркетологи, так і технічні спеціалісти і вчені з області інформаційних наук, тому що процеси, що відбуваються у середовищі онлайн-ринку, обумовлені як маркетинговими явищами, так і звісно технічними особливостями функціонування того чи іншого проекту [7]. Досліджуються усі аспекти діяльності таких проектів, багато уваги приділяється взаємозв'язку технічної реалізації, внутрішньої політики та результативності певного проекту [2].

Класифікація онлайн-маркетів теж одна із великих задач дослідження цього явища, кожен із типів має свої показники успішності і особливості роботи. Загалом вирізняють такі типи маркетплейсів за особливостями функціонування:

1. Вертикальні маркетплейси, призначені для тільки однієї категорії продуктів або послуг, наприклад, Airbnb, Bolt.
2. Горизонтальні маркетплейси, призначені для широкого спектру товарів, наприклад, Amazon.
3. Продуктові маркетплейси, які працюють тільки з фізичними товарами.
4. Віртуальні маркетплейси, які продають тільки віртуальні продукти (наприклад, маркетплейси для геймерів, біржі або онлайн-маркети віртуальних валют та крипто валют, маркетплейси, які продають творчі надбання).
5. Сервісні онлайн-маркети, які пропонують тільки послуги, сюди також частково відносяться біржі фрілансу.
6. Гібридні маркетплейси, які пропонують розміщення як продуктів, фізичних та віртуальних, так і послуг.
7. Нішеві маркетплейси, які займають тільки невелику частину ринку або покривають специфічну вузьку нішу.
8. «Чорні» маркетплейси, вкрай успішні майданчики, де продаються незаконні товари і укладаються угоди про незаконну діяльність [8].

Архітектура проектів може суттєво відрізнятись в залежності від направленості, розміру, віку проекту та інших особливостей проекту, але типові риси архітектури властиві більшості маркетплейсів та близьких до них проектів. Схема архітектури зображена на рис. 1.

Таким чином, типова архітектура такого типу проекту складається з чотирьох залежних один від одного шарів, в кожному із яких виконується частина логіки взаємодії між покупцем, продавцем та системою. Відкидаючи побічні дії на кшталт конфігурацій, налаштувань та логування подій та даних, зосередимось на шарі бізнес-логіки, щоб дослідити, що саме там відбувається та спроектувати свій продукт на основі цього функціоналу.

Мета дослідження – на прикладі мінімалістичного додатку дослідити створення платформи-маркетплейсу за допомогою фреймворка Laravel.

Виклад основного матеріалу. Електронна комерція має декілька переваг порівняно з традиційними методами купівлі та продажу товарів. По-перше, вона дозволяє покупцям здійснювати покупки з будь-якого місця та в будь-який час, що робить процес купівлі більш зручним та доступним. По-друге, вона дає можливість продавцям знизити витрати на зберігання та транспортування товарів, що забезпечує зниження вартості продукції для кінцевих покупців. По-третє,

вона забезпечує зручний та швидкий спосіб оплати товарів та послуг, що сприяє збільшенню обсягів продажів.

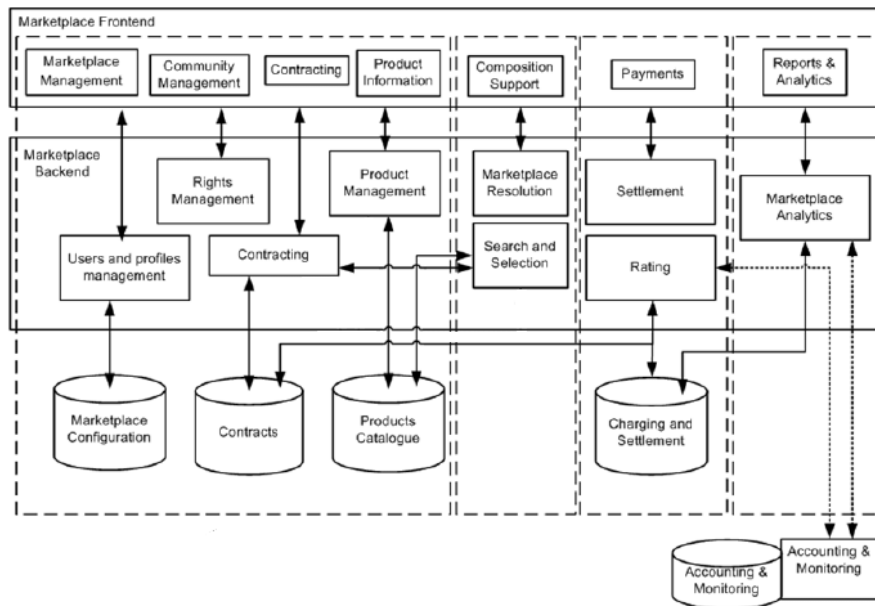


Рис. 1 – Типова архітектура онлайн-маркету

Одним з найбільших викликів електронної комерції є проблеми з безпекою. Зокрема, під час онлайн-платежів можуть виникати ризики викрадення особистих даних клієнтів та крадіжки грошей з їх рахунків. Щоб забезпечити безпеку, компанії повинні використовувати захист від шахрайства та фішингу, а також виконувати вимоги щодо захисту даних клієнтів, включаючи створення безпечних паролів та шифрування даних.

Особливо важливим для платформ-маркетплейсів є побудування відносин між продавцем та покупцем. Цей процес міститиме розробку правил та гарантій їх дотримання, можливість розв'язання суперечок, розглядання в розумні терміни скарг і ухвалення адекватних рішень по кожній з них, а також захист від нечесних дій за допомогою захисту платежів і гарантій виконання зобов'язань з обох сторін. Обидві групи користувачів (і покупці, і продавці) є клієнтами платформи, тому захист угод та розв'язання конфліктів є однією з важливих частин функціонування платформи. Також необхідно приділяти увагу до захисту даних, паролів, платіжних транзакцій, що є складовою безпекової стратегії додатку.

Вибір мови програмування для подібних проектів обумовлений багатьма факторами і є складним стратегічним рішенням. Зазвичай вибір мови заснований на можливостях команди, але при проектуванні слід враховувати не тільки поточну ситуацію на ринку та у бізнесі, а і перспективи розвитку. Так, треба враховувати перспективу зростання попиту і кількості користувачів і відвідувачів на сайті чи у додатку, яка вимагатиме масштабування кодової бази та бази даних, підвищення пропускну здатності і підвищення продуктивності функціонування програмного продукту. Для великих продуктів або продуктів, які мають швидку перспективу зростання, обирають асинхронні варіанти, комбіновані архітектури та часто мікро-сервісний або модульний підхід. Для програмних продуктів, зростання яких лежить у коридорі від повільного до помірного, а також для продуктів, які треба швидко і досить дешево підтримувати і розвивати, основним технологічним стеком є так званий LAMP. В основі цього стеку лежить мова PHP, яка є однією з найрозповсюдженіших мов програмування. Ця мова є досить швидкою, щоб витримувати помірні та високі навантаження, а також досить безпечною. Вона активно розвивається ком'юніті і використовується у великій кількості проектів. Зазвичай у зв'язці з PHP використовують систему керування базами даних MySQL, яка являє собою систему, що дозволяє створювати, підтримувати та масштабувати реляційні бази даних. Якщо проект не піддається надвисокому навантаженню, то такий технологічний стек є раціональним вибором через його відносно легку підтримуваність, можливості розміщувати на дешевих виділених та хмарних серверах та

відносну стабільність додатків. В якості браузерного клієнту можна використовувати безліч різних варіантів бібліотек і фреймворків на базі мови Javascript або інших мов, що здатні виконуватись у середовищі веббраузера.

Для розробки вебдодатків часто використовують фреймворки. Фреймворк є набором інструментів, бібліотек та методологій, що спрощують розробку програмного забезпечення і дозволяють розробникам швидше створювати додатки. Наявність стандартних методів вирішення типових задач також дозволяє уникнути помилок, збільшує стабільність та безпеку додатку.

Після аналізу різних методів вирішення проблем було зроблено висновок, що використання фреймворку Laravel [9] забезпечить швидку розробку та ефективне використання ресурсів. Основні переваги використання фреймворку Laravel полягають у тому, що він забезпечує зручну інтерфейсну частину, швидкість розробки, високу продуктивність та безпеку вебдодатків. Один з найбільш важливих аспектів Laravel – це його система маршрутизації, яка дозволяє розробнику легко налаштувати маршрути та контролери в додатку. Це дозволяє значно скоротити час розробки та забезпечує простоту управління трафіком в додатку. Laravel також має велику кількість різноманітних додаткових бібліотек і плагінів, що робить його більш гнучким для розробки вебдодатків різних складностей [10].

Для проектування задачі платформи були умовно поділені на наступні модулі:

1. Товари – містить сутності і бізнес логіку, яка включає опис товарів, їх категоризацію, сортування, фільтрування та рекомендації.

2. Модуль роботи з продавцем (Магазин) – система керування товарами, налаштуваннями, статистикою продажів та ін.

3. Модуль покупця – містить роботу з платформою з точки зору покупця, включаючи особистий кабінет із списком замовлень, кошик, персональні дані.

4. Платіжний модуль – модуль додатку, що працює з платіжними системами на стороні покупця та продавця, забезпечує безпечні транзакції, простий Escrow-подібний досвід переказів та ін.

5. Модуль арбітражу – модуль, робота якого присвячена розв'язанню конфліктів та обробці взаємних скарг.

6. Модуль доставки – регулює діяльність та налаштування коду, що відповідає за оформлення доставки і відстежування, якщо воно доступне.

7. Модуль статистики – відповідає за збір, зберігання та аналіз статистики та поставляє дані до адміністраторів системи та продавців.

Основні завдання розробки маркетплейсу включають:

– розробку архітектури системи: структури бази даних, взаємодії між компонентами системи;

– розробку інтерфейсу користувача: дизайн і розміщення елементів, функціональність;

– розробку функціональності системи: авторизація користувачів, створення та редагування товарів, оформлення замовлень, оплата тощо;

– тестування та відлагодження системи.

Діаграма станів і переходів наведено на рис. 2.

Отже, для досягнення поставлених завдань розробки онлайн-маркету були використані наступні сучасні технології та програмні засоби:

– фронтенд: Vue 3, Laravel 9 (має вбудований шаблонний движок Blade, який дозволяє розбити HTML-код на невеликі шаблони), Bootstrap 5;

– бекенд: Laravel 9, Node.JS[11];

– СУБД MySQL 8.0 [12].

До додаткових інструментів додаються: пакетні менеджери Composer, NPM, а також інструмент для збірки Vite із мовою препроцесору SCSS.

Архітектура системи вебдодатку на Laravel, Vue [13] та MySQL була побудована на базі клієнт-серверної архітектури з використанням REST API. У цій архітектурі клієнти взаємодіють з сервером через API, виконуючи запити і отримуючи відповіді в форматі JSON.

Вебдодаток онлайн маркету складається з 44 маршрутів, три з котрих стандартні, створені для відслідковування помилок в роботі додатку, а один направлено на генерацію токєну захисту від CSRF-атак. Інші маршрути поділені на дві великі категорії: API, Web.

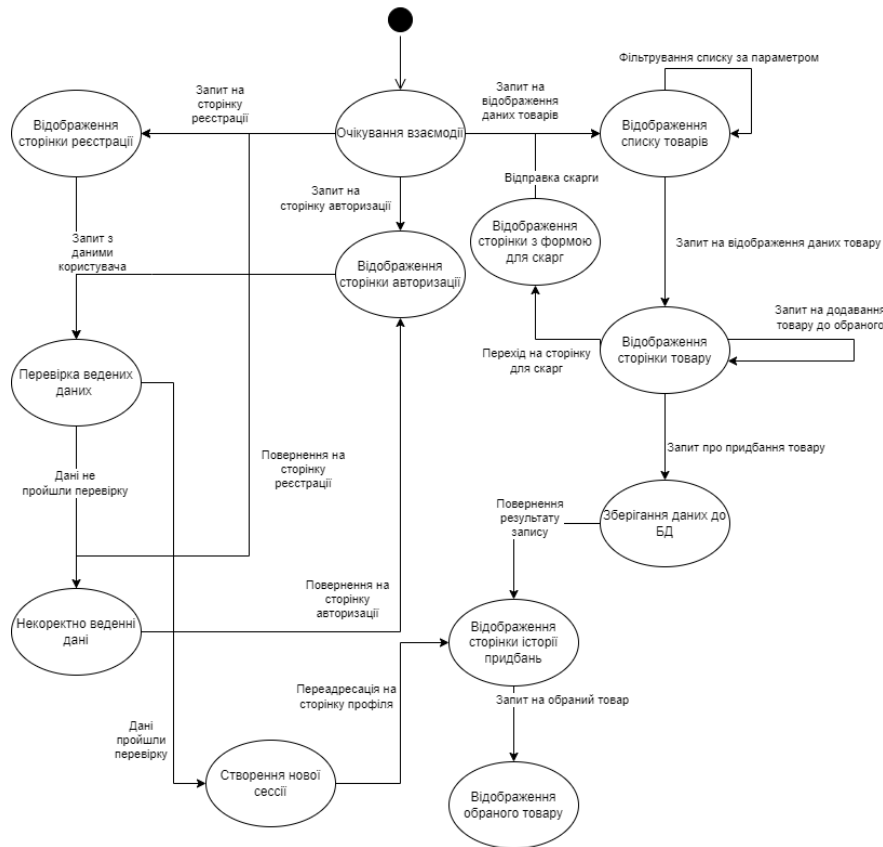


Рис. 2 – Діаграма станів і переходів

Web-маршрути використовуються для забезпечення доступу до статичних HTML-сторінок, які відображаються в браузері. Ці маршрути можуть використовувати middleware, які забезпечують аутентифікацію, авторизацію та інші функції безпеки.

API-маршрути використовуються для доступу до даних через API. Вони використовуються для передачі даних в JSON-форматі та відповідають на запити HTTP-методами, такими як GET, POST, PUT та DELETE. API-маршрути також можуть використовуватись проміжним програмним забезпеченням, які забезпечують аутентифікацію та авторизацію, а також перевірку валідності даних, які надсилаються з клієнтської сторони. У даній роботі API-маршрути були зроблені для взаємодії з Vue-компонентами.

Було прийнято рішення робити проект без певної головної сторінки, роль головної сторінки відіграє сторінка з рекомендованими або обраними товарами із каталогу. Ця сторінка з товарами має адаптивну верстку під будь-які розміри екрану та товари, які виводяться послідовно по 15 елементів і доповнюються у сторінку при наблизненні користувачем кінця сторінки. Обравши товар, користувач потрапляє до сторінки товару, яка має кілька відмінностей у користувацькому досвіді взаємодії та дизайн, залежно від ролі користувача.

Основні скріншоти програмного забезпечення наведено на рис. 3.

Для покупки товару на платформі покупець повинен надати особисті дані, а для цього необхідна реєстрація. Для реєстрації необхідно перейти на відповідний розділ на сайті, вказати ім'я користувача, адрес електронної пошти, пароль. Для фільтрування товарів за категоріями необхідно натиснути на відповідний розділ у меню, яке розташовано на навігаційному меню. Після того, як користувач обирає необхідний товар, він переходить на сторінку з цим товаром, де може придбати товар, додати його до обраного, перейти до іншого, рекомендованого, товару, ознайомитись з описом, оцінкою та відгуками на нього, переглянути додаткові зображення товару. За бажанням він може залишити свій відгук. Після додавання товару до обраного, користувача поверне на сторінку з товаром. Після придбання, його перенаправить на сторінку профіля, де буде відображена історія придбань. Там же зліва він може обрати перегляд обраного товару, а також, за бажанням, видалити його зі списку.

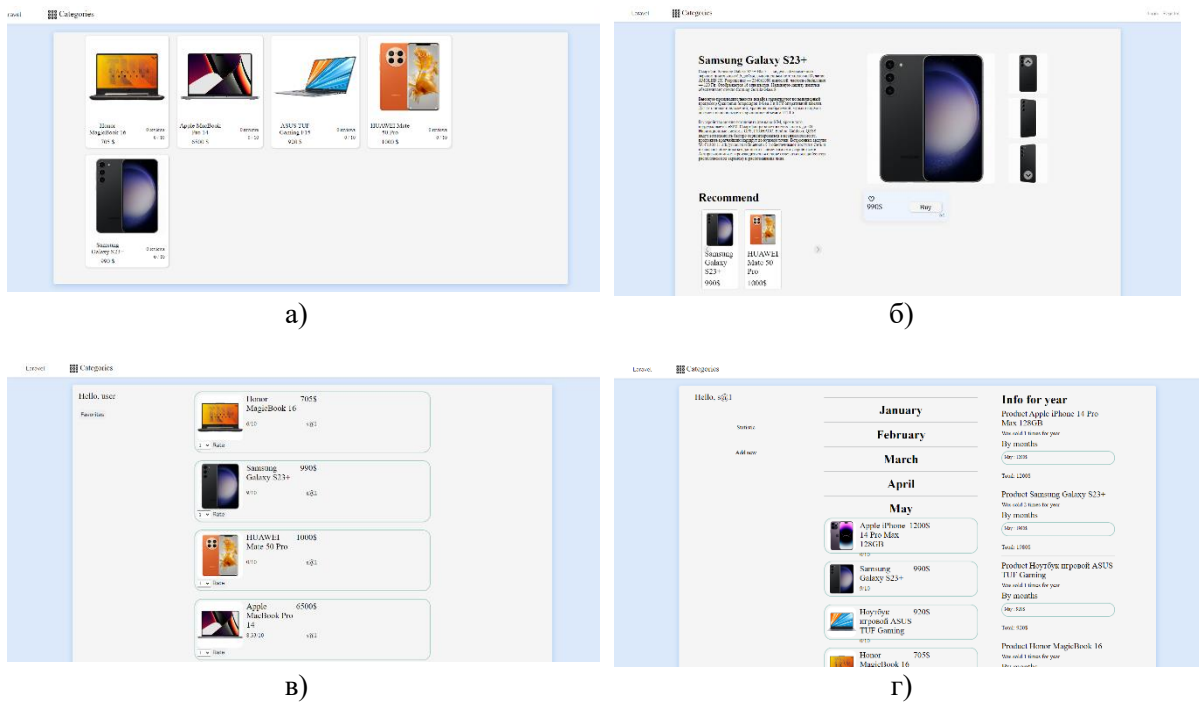


Рис. 3 – Основні скріншоти вебдодатку: а) – головна сторінка; б) – сторінка товару; в) – сторінка профілю покупця зі списком товарів, які він придбав; г) – сторінка профілю продавця зі статистикою продажів

На сторінці профілю продавець може додавати новий товар або переглядати статистику продажів. На першій сторінці він має заповнити форму, яка зберігає його товар у базу даних, після чого вона буде відображена у списку на головній та за фільтрацією по категоріях або продавцях. На другій сторінці продавець буде бачити інформативну статистику продажів. На ній буде зображено товари, відсортовані по місяцях. Також буде підраховано скільки було продано кожного товару щомісячно, а також скільки було отримано грошей за ці продажі, що буде підраховано також щомісячно та за весь рік цілком. Крім цього продавцю дозволяється видалити свій товар на сторінці товару. Останній функціонал на сторінці товару, а також на списках товарів, не відрізняється від функціоналу, до якого має доступ продавець.

Адміністратор також проходить етап реєстрації, після якого його акаунту надають роль адміністратора. Адміністратор має можливість видалити будь-який товар з публічного доступу на сторінці товару. На сторінці профілю у адміністратора знаходиться список скарг, на які він має можливість відповісти двома варіантами. У скарзі може бути відмовлено, якщо вона не відповідає дійсності.

Таким чином, було створено вебдодаток для електронної комерції, який дозволяє:

- користувачам реєструватись та переглядати товари в онлайн-маркету, фільтруючи їх за категоріями або продавцями; слідкувати за історією покупок та обраними товарами;
- продавцям створювати нові позиції на маркету, а також підраховування статистичних даних щодо продажів;
- адміністраторам сайту швидко реагувати на скарги або видалити товари, які порушують правила сервісу чи є незаконними.

Висновки

- 1) Було проведено огляд принципів побудування додатків-маркетплейсів, їх розмаїття, особливості функціонування та архітектури.
- 2) Було проаналізовано основні етапи розробки, такі як проектування бази даних, розробка моделей та контролерів, налаштування маршрутів та створення користувацького інтерфейсу, та

створено вебдодаток для електронної комерції, який забезпечує високий рівень безпеки та швидкість роботи.

3) Практична значущість розробленого додатку полягає у тому, що він надає можливість зручно та швидко розгорнути продавцям свій бізнес; користувачам – отримати необхідний товар в будь-який час та з будь-якої точки світу.

4) Лімітований функціонал додатку дозволяє розробити тільки основні модулі і функції, а потім у гнучкому режимі із попереднім тестуванням додавати те, що необхідно саме цільовій аудиторії покупців і продавців, а також залучати тих, для кого функціонал та конкурентне сервіси великих маркетплейсів є надлишковим.

Перелік використаних джерел:

1. Sedziewski M. Building an online marketplace from scratch – introduction [Online]. Available: <https://medium.com/hackernoon/building-an-online-marketplace-from-scratch-introduction-738839e4e76> (дата звернення 30.09.2023).
2. A marketplace framework for trading cloud-based services / A. Menychtas, S. Garcia Gomez, A. Giessmann, A. Gatzoura, K. Stanoevska, J. Vogel. *Proceedings of the 8th international conference on Economics of Grids, Clouds, Systems, and Services GECON 2011*, Paphos, Cyprus, 5 December 2011. Pp. 76-89. DOI: https://doi.org/10.1007/978-3-642-28675-9_6.
3. Amazon. URL: <https://www.amazon.com/> (дата звернення 30.06.2023).
4. EBay. URL: <https://www.ebay.com/> (дата звернення 30.06.2023).
5. Aliexpress. URL: <https://www.aliexpress.com/> (дата звернення 30.06.2023).
6. Etsy. URL: <https://www.etsy.com/> (дата звернення 30.06.2023).
7. MarketPlace – a digital materials modelling marketplace / G. Goldbeck et al. 2023. 39 p. DOI: <https://doi.org/10.5281/zenodo.8330333>.
8. 20+ Best marketplaces in Germany. URL: <https://ecommercegermany.com/blog/20-best-online-marketplaces-in-germany> (дата звернення 01.10.2023).
9. Pecoraro C. *Mastering Laravel*. Birmingham : Packt Pub., 2015. 204 p.
10. Wilson R. *Planning your internet marketing strategy: a doctor Ebiz guide*. New York : Wiley, 2001. 276 p.
11. Syed B. *Beginning Node.js*. New York : Apress, 2014. 308 p.
12. Stokes D. *MySQL and JSON: a practical programming guide*. New York : McGraw Hill, 2018. 128 p.
13. Hanchett E., Listwon B. *Vue.js in action*. Shelter Island : Manning, 2018. 304 p.

References:

1. M. Sedziewski, Building an online marketplace from scratch – introduction [Online]. Available: <https://medium.com/hackernoon/building-an-online-marketplace-from-scratch-introduction-738839e4e76>. Accessed on: September 30, 2023.
2. A. Menychtas, S. Garcia Gomez, A. Giessmann, A. Gatzoura, K. Stanoevska, and J. Vogel, «A marketplace framework for trading cloud-based services», in Proceedings of the 8th international conference on Economics of Grids, Clouds, Systems, and Services GECON 2011, Paphos, Cyprus, 2011, pp. 76-89. doi: 10.1007/978-3-642-28675-9_6.
3. Amazon [Online]. Available: <https://www.amazon.com/>. Accessed on: June 30, 2023.
4. EBay [Online]. Available: <https://www.ebay.com/>. Accessed on: June 30, 2023.
5. Aliexpress [Online]. Available: <https://www.aliexpress.com/>. Accessed on: June 30, 2023.
6. Etsy [Online]. Available: <https://www.etsy.com/>. Accessed on: June 30, 2023.
7. G. Goldbeck et al., *MarketPlace – a digital materials modelling marketplace*. 2023. 39 p. doi: 10.5281/zenodo.8330333.
8. 20+ Best marketplaces in Germany [Online]. Available: <https://ecommercegermany.com/blog/20-best-online-marketplaces-in-germany>. Accessed on: October 1, 2023.
9. C. Pecoraro, *Mastering laravel*. Birmingham, UK: Packt Publ., 2015.
10. R. Wilson, *Planning your internet marketing strategy: a doctor Ebiz guide*. New York, USA: Wiley Publ., 2001.
11. B. Syed, *Beginning Node.js*. New York, USA: Apress Publ., 2014.

12. D. Stokes, *MySQL and JSON: a practical programming guide*. New York, USA: McGraw Hill Publ., 2018.
13. E. Hanchett, and B. Listwon, *Vue.js in Action*. Shelter Island, USA: Manning Publ., 2018.

Рецензент: О.І. Проніна
канд. техн. наук, доц., ДВНЗ «ПДТУ»

Стаття надійшла 18.10.2023

Стаття прийнята 23.11.2023

УДК 004.056.53:519.17

doi: 10.31498/2225-6733.47.2023.299981

© Кажан Д.В.¹, Левицька Т.О.², Кривенко О.В.³

ВИЗНАЧЕННЯ ТИПУ ВИБУХОНЕБЕЗПЕЧНОГО ПРЕДМЕТУ ЗА ДОПОМОГОЮ ДЕРЕВА РІШЕНЬ

В роботі наведено метод дерева рішень для класифікації типу вибухонебезпечних предметів. Це дозволяє робити висновки відносно типу вибухонебезпечного предмету, що заховано в залежності від типу ґрунту. Дана проблематика є актуальною, оскільки дозволяє обґрунтовано пояснювати тип небезпеки в залежності від різних зовнішніх факторів. В роботі запропоновано загальне представлення побудови дерева рішень для визначення виду вибухонебезпечних предметів. Крім цього в роботі надано візуалізацію розбиття на вузли для наочного представлення побудови дерева рішень з урахуванням розбиття на піддерева та вузли. У якості вузлів виступають основні атрибути: напруга, висота датчика над рівнем ґрунту, тип ґрунту. Наведено алгоритм побудови дерева рішень та структурну блок-схему алгоритму ID3. Було проведено дослідження, на основі якого було обрано для вимірювання хаосу розбиття дерева рішень ентропію. Були перелічені основні характеристики алгоритму ID3 з урахуванням предметної галузі, в якій він застосовується. В роботі наведено одне з побудованих дерев рішень, що було побудовано в рамках досліджень для визначення типу вибухонебезпечного предмета в залежності від можливого типу ґрунту, на прикладі це сухий та піщаний ґрунт.

Ключові слова: дерева рішень, візуалізація даних, вибухонебезпечні предмети, алгоритм ID3.

D.V. Kazhan, T.O. Levytska, O.V. Kryvenko. Determining the type of explosive item using a decision tree. The present circumstances in Ukraine delineate specific areas of study. The imperative to identify an explosive device is increasingly paramount for adults and particularly for children. Failing to recognize this danger poses a threat to health and life. The paper presents a decision tree method for classifying the type of explosive objects. This allows you to draw conclusions about the type of explosive object hidden depending on the type of soil. This problem is relevant, as it allows to reasonably explain the type of danger depending on various external factors. The paper offers a general representation of the construction of a decision tree for determining the type of explosive objects. In addition, the work provides a visualization of the division into nodes for a visual representation of the construction of a decision tree, taking into account the division into subtrees and nodes. The nodes are the main attributes: voltage, sensor height above ground level, soil type. The

¹ магістрант, ДВНЗ «Приазовський державний технічний університет», м. Дніпро

² канд. техн. наук, доцент, ДВНЗ «Приазовський державний технічний університет», м. Дніпро, ORCID: 0000-0003-3359-1313, levitskaya_t_a@pstu.edu

³ канд. техн. наук, доцент, ДВНЗ «Приазовський державний технічний університет», м. Дніпро, ORCID: 0009-0006-2860-6575, krivenko_o_v@pstu.edu