

Міністерство освіти і науки України
Луцький національний технічний університет
Факультет комп'ютерних та інформаційних технологій
Кафедра інженерії програмного забезпечення

**КВАЛІФІКАЦІЙНА РОБОТА
ЗА СТУПЕНЕМ ВИЩОЇ ОСВІТИ «БАКАЛАВР»**

**ВЕБ-ПЛАТФОРМА ЗА ДОПОМОГОЮ ЧЕКСБОХ: АСПЕКТИ
ФІСКАЛЬНОГО ОБЛІКУ, ЗВІТНІСТЬ І ВЗАЄМОДІЯ З
ПОДАТКОВИМИ ОРГАНАМИ**

**WEB PLATFORM USING CHECKBOX: ASPECTS OF FISCAL
ACCOUNTING, REPORTING AND INTERACTION WITH TAX
AUTHORITIES**

спеціальність 121 Інженерія програмного забезпечення

(шифр і назва спеціальності)

освітня програма «Інженерія програмного забезпечення»

(назва освітньої програми)

Виконав: здобувач вищої освіти
Групи ІПЗс-21

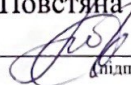
Качула Іван Миколайович


(підпис)

Керівник:

к.т.н., доцент

Повстяна Юлія Славомирівна


(підпис)

Кваліфікаційну роботу

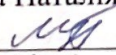
допущено до захисту

« 8 » червне 2024 р.

к.т.н., доцент

Гарант освітньої програми:

Ліщина Наталія Миколаївна


(підпис)

Луцьк – 2024 року

ЛУЦЬКИЙ НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ

Факультет комп'ютерних та інформаційних технологій
Кафедра інженерії програмного забезпечення
Ступінь вищої освіти: бакалавр
Галузь знань: 12 Інформаційні технології
Спеціальність: 121 Інженерія програмного забезпечення
Освітня програма: «Інженерія програмного забезпечення»

ЗАТВЕРДЖУЮ

Завідувач кафедри

М. М. Микуню
« 2 » 02 2024 р.

ЗАВДАННЯ НА КВАЛІФІКАЦІЙНУ РОБОТУ ЗДОБУВАЧУ ВИЩОЇ ОСВІТИ

Качулі Івану Миколайовичу

(прізвище, ім'я, по батькові)

1. Тема кваліфікаційної роботи: Веб-платформа за допомогою Checkbox: аспекти фіскального обліку, звітність і взаємодія з податковими органами

Керівник роботи: к.т.н., доцент Повстяна Юлія Славомирівна

затверджені наказом закладу вищої освіти від «30» грудня 2023 р. № 477/01-02

2. Строк подання здобувачем вищої освіти кваліфікаційної роботи «5» червня 2024 р.

3. Вихідні дані до роботи: технічне та програмне забезпечення ЕОМ, вимоги до розробки програмного забезпечення, ергономічні вимоги до функціонування програмного засобу.

4. Зміст розрахунково-пояснювальної записки (перелік питань, що потрібно розробити):

Аналіз сучасного стану проблеми, існуючих методів і засобів її розв'язання, аналіз, визначення вимог до розроблюваного програмного забезпечення та проектування програмного забезпечення, опис функціонального наповнення об'єкта проектування, практична реалізація об'єкта проектування.

5. Перелік графічного матеріалу: 18 рисунків; 2 схеми; 1 діаграма.

6. Консультанти розділів роботи

Розділ	Прізвище, ініціали та посада консультанта	Підпис	
		завдання видав	завдання прийняв
Аналіз предметної області	Повстяна Ю. С.		
Специфікації вимог до розробленої системи	Повстяна Ю. С.		
Розробка об'єкта проєктування	Повстяна Ю. С.		
Нормоконтроль	Повстяна Ю. С.		
Гарант ОП	Ліщина Н. М.		
Показник запозичень тексту	3,3 %		
Академічна доброчесність	Яцук А. А.		

7. Дата видачі завдання «2» лютого 2024 р.

КАЛЕНДАРНИЙ ПЛАН

№ 3/п	Назва етапів кваліфікаційної роботи бакалавра	Строк виконання етапів роботи	Примітка
1	Огляд літературних джерел по темі кваліфікаційної роботи бакалавра	07.02.2024 р.	
2	Аналіз проблеми розробки та впровадження об'єкту проєктування	27.02.2024 р.	
3	Обґрунтування вибору шляхів, технологій (алгоритмів) і засобів вирішення поставленого завдання	12.03.2024 р.	
4	Розробка функціонально-структурної схеми роботи об'єкта проєктування та проєктування бази даних	17.04.2024 р.	
5	Практична реалізація об'єкта проєктування та розробка бази даних	18.05.2024 р.	
6	Тестування та налагодження об'єкта проєктування	01.06.2024 р.	
7	Здача чистового варіанту кваліфікаційної роботи бакалавра на кафедру	05.06.2024 р.	

Здобувач вищої освіти

Керівник кваліфікаційної роботи

(підпис)

Качула І. М.
(прізвище, ініціали)
Повстяна Ю. С.
(прізвище, ініціали)

Міністерство освіти і науки України
Луцький національний технічний університет

Рецензія
на кваліфікаційну роботу бакалавра

Здобувач вищої освіти:

Качула Іван Миколайович

Тема:

Веб-платформа за допомогою Checkbox: аспекти фіскального обліку, звітність і взаємодія з податковими органами

Коротка характеристика кваліфікаційної роботи:

Кваліфікаційна робота бакалавра присвячена створенню веб-платформи для фіскального обліку, звітності та взаємодії з податковими органами за допомогою технології Checkbox. Автор глибоко дослідив питання автоматизації бухгалтерських процесів та податкової звітності, представивши детальний аналіз існуючих рішень на ринку та їхні обмеження. Робота складається з теоретичної частини, де описано основи фіскального обліку, і практичної частини, яка включає розробку веб-платформи з використанням сучасних технологій і фреймворків.

Самостійні розробки і пропозиції автора:

Автор продемонстрував високу самостійність у розробці веб-платформи, створивши надійну та безпечну архітектуру системи. Особливу увагу було приділено інтуїтивному інтерфейсу користувача та автоматичній передачі даних до податкових органів. Одним із головних досягнень є створення архітектури системи, що забезпечує надійність, масштабованість і безпеку обробки фінансових даних. Ці розробки свідчать про здатність автора вирішувати складні завдання у сфері комп'ютерних технологій.

Практичне значення роботи:

Практичне значення кваліфікаційної роботи полягає у створенні функціональної веб-платформи, яка може бути використана підприємствами для автоматизації процесів фіскального обліку та податкової звітності. Розроблена система дозволяє знизити трудовитрати на ведення бухгалтерії, мінімізувати ризики помилок у звітності та

забезпечити своєчасну взаємодію з податковими органами. Таким чином, робота має вагоме практичне значення і може бути основою для подальших досліджень та вдосконалень у сфері автоматизації бухгалтерських процесів.

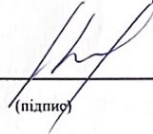
Загальний висновок:

Кваліфікаційна робота є цінним внеском у автоматизацію фіскального обліку та податкової звітності, демонструючи високу кваліфікацію автора та значний практичний потенціал для реального впровадження. Здобувач освіти Качула Іван Миколайович заслуговує позитивну оцінку.

Рецензент: к. тех. н., доцент Ліщина Валерій Олександрович

(прізвище, ім'я, по-батькові, посада)

Рецензент кваліфікаційної роботи


(підпис)

«7» 06 2023р.

Міністерство освіти і науки України
Луцький національний технічний університет
Факультет комп'ютерних та інформаційних технологій
Кафедра інженерії програмного забезпечення

ВІДГУК
КЕРІВНИКА НА КВАЛІФІКАЦІЙНУ РОБОТУ БАКАЛАВРА

Здобувач вищої освіти Качула Іван Миколайович
група ПЗс-21

Тема кваліфікаційної роботи Веб-платформа за допомогою Checkbox: аспекти фіскального обліку, звітність і взаємодія з податковими органами

Керівник к.т.н., доц. Повстяна Ю. С.

Актуальність теми

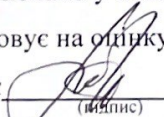
Актуальність теми роботи полягає в необхідності автоматизації процесів фіскального обліку та податкової звітності, що є критично важливим для сучасних підприємств. В умовах постійних змін у законодавстві та збільшення обсягу даних, що потребують обробки, використання сучасних технологій, таких як Checkbox, дозволяє значно підвищити ефективність та точність ведення бухгалтерії. Це допомагає зменшити ризики помилок, скоротити час на обробку даних та забезпечити своєчасну взаємодію з податковими органами, що в кінцевому результаті сприяє покращенню фінансової прозорості та економічної стабільності підприємств.

Об'єкт дослідження є процеси фіскального обліку та податкової звітності підприємств, а також засоби автоматизації цих процесів за допомогою веб-платформи, створеної з використанням технології Checkbox. Дослідження охоплює аналіз існуючих систем автоматизації бухгалтерського обліку, виявлення їхніх недоліків та переваг, а також розробку та впровадження власного програмного рішення для покращення цих процесів.

Характеристика теоретичного рівня, наявності самостійних розробок і практичної значимості роботи. Робота відзначається високим теоретичним рівнем, детально аналізуючи методи фіскального обліку та нормативні вимоги до податкової звітності. Автор самостійно розробив архітектуру системи, інтерфейс користувача, механізми автоматичної передачі даних до податкових органів та засоби захисту даних. Практична значимість роботи полягає у створенні веб-платформи, яка автоматизує фіскальний облік та податкову звітність, підвищуючи ефективність фінансового управління підприємствами.

Загальний висновок

Роботу виконано у повному обсязі у відповідності до поставлених завдань, а здобувач Качула І. М. заслуговує на оцінку «відмінно».

Керівник  (підпис)

Повстяна Ю.С.
(прізвище, ім'я, по батькові)

«07» 06 2024 р.

АНОТАЦІЯ

Качула І. М. Веб-платформа за допомогою Checkbox: аспекти фіскального обліку, звітність і взаємодія з податковими органами. Рукопис.

Кваліфікаційна робота бакалавра ОП «Інженерія програмного забезпечення». Луцький національний технічний університет. Луцьк, 2024.

Кваліфікаційна робота бакалавра складається зі вступу, трьох розділів, висновків і пропозицій, списку використаних джерел та додатків.

У першому розділі здійснено аналіз сучасного стану проблеми веб застосунків із можливістю фіскалізації чеків і сформульовано завдання. В другому розділі визначено вимоги до розроблюваної системи, а також засоби, методи і алгоритми розробки. У третьому розділі розроблено інтернет магазин із можливістю фіскалізації чеків і подання звітності у податкову. У висновках узагальнено інформацію, відображену у попередніх частинах. Результати розробки демонструють можливості взаємодії інтернет магазинів із податковими органами.

Ключові слова: фіскалізація, чек, звітність, інтернет магазин, Checkbox, Laravel, REST API, Blade.

ABSTRACT

Kachula I. M. Web Platform Using Checkbox: Aspects of Fiscal Accounting, Reporting and Interaction With Tax Authorities. Manuscript.

Bachelor's qualifying thesis of the educational program "Software Engineering". Lutsk National Technical University. Lutsk, 2024.

The bachelor's qualifying thesis consists of an introduction, three sections, conclusions and proposals, a list of used sources and annexes.

In the first chapter, an analysis of the current state of the problem of web applications with the possibility of fiscalizing checks and the task is formulated. The requirements are defined in the second section and the task was formulated. The second section defines the requirements for the developed system, as well as means, methods and development algorithms. In the third chapter, an online store with the possibility of fiscalizing checks and submitting reports to the tax office was developed and tested. The conclusions summarize the information presented in the previous parts. The development results demonstrate the possibilities of interaction of online stores with tax authorities.

Keywords: fiscalization, receipt, reporting, online store, Checkbox, Laravel, REST API, Blade.

ЗМІСТ

ВСТУП.....	7
РОЗДІЛ 1 АНАЛІЗ ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ ВЕБ ЗАСТОСУНКІВ ІЗ МОЖЛИВІСТЮ ФІСКАЛІЗАЦІЇ ЧЕКІВ І ПОСТАНОВКА ЗАВДАНЬ НА КВАЛІФІКАЦІЙНУ РОБОТУ.....	9
1.1 Аналіз сучасного стану проблеми.....	9
1.2 Постановка завдання на кваліфікаційну роботу бакалавра.....	16
Висновки до розділу 1.....	17
РОЗДІЛ 2 СПЕЦИФІКАЦІЯ ВИМОГ ДО РОЗРОБЛЮВАНОЇ СИСТЕМИ.....	19
2.1 Аналіз, визначення вимог до розроблюваного програмного забезпечення та проектування програмного забезпечення.....	19
2.2 Вибір засобів, методів і алгоритмів вирішення поставленого завдання...	21
Висновки до розділу 2.....	23
РОЗДІЛ 3 РОЗРОБКА ІНТЕРНЕТ МАГАЗИНУ ІЗ МОЖЛИВІСТЮ ФІСКАЛІЗАЦІЇ ЧЕКІВ.....	24
3.1 Практична реалізація об'єкта проектування.....	24
3.2 Тестування та налагодження інформаційно-комп'ютерної системи.....	35
3.3 Розробка документації для програмного продукту.....	38
3.4 SEO інформаційно-комп'ютерної системи.....	41
Висновки до розділу 3.....	43
ВИСНОВКИ.....	45
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ.....	46
ДОДАТКИ.....	47

ВСТУП

Актуальність роботи. Фіскалізація є однією із найбільш поширених проблем інтернет магазинів. Кожне замовлення яке зроблене на сайті повинне пройти цей процес і отримати свій фіскальний чек. Зі вступом в чинність закону України №129-IX, кожен підприємець повинен видавати чеки клієнтам, тому це на даний час нагальна проблема малого та великого бізнесу. Чеки видаються в будь якому випадку коли є факт розрахункової операції, такі як оплата платіжною карткою в інтернет магазині.

Ще донедавна вся фіскальна звітність відбувалась в офлайн режимі із бухгалтерським складом який формував звіти до податкових органів, але вже є всі можливості для перенесення всіх цих процесів в онлайн режим. Автоматизація цих процесів в інтернет магазинах є серед найпопулярніших питань, які підприємці вирішують різними шляхами.

Програмні реєстратори розрахункових операцій вже давно не тільки не поступається в зручності використання і набору функціональності звичайним РРО, а і перевищують їх можливості.

Фіскалізація, повернення, часткове повернення, звітність до податкових органів, всі ці функції допомагає виконувати програмний реєстратор розрахункових операцій (пРРО) Checkbox. Цей сервіс дозволяє українським підприємцям займатися бізнесом, а не бюрократією, яка повністю переходить в онлайн режим.

Правильна реалізація і інтеграція можливостей сервісу Checkbox покажуть наскільки простішою може бути вся бюрократична взаємодія з податковими органами. Показати наскільки простим може бути ця взаємодія із врахуванням всіх законів чинного законодавства України і буде завданням цієї кваліфікаційної роботи.

Метою роботи є розробка веб-платформи інтернет магазину з функціоналом онлайн фіскалізації чеків та з можливістю введення податкової звітності, який буде оптимізовувати процес взаємодії із податковими органами.

Об'єкт дослідження – процес та результат розробки системи із можливістю фіскалізації та звітування в податкових органах.

Предмет дослідження – методи і засоби взаємодії інтернет магазинів із податковою службою та процеси фіскалізації чеків.

Завдання дослідження:

- вивчити питання та проаналізувати існуючі сервіси та методи інтеграції онлайн фіскалізації чеків в інтернет магазинах, а саме на сервісах Checkbox, Cashalot та Вчасно.Каса;
- визначити вимоги для онлайн фіскалізації чеків та звітів в податкову і проаналізувати досвід провідних інтернет магазинів в цьому питанні;
- розробити схему бази даних та обдумати методи для ефективного впровадження онлайн фіскалізації чеків;
- реалізувати інтернет магазин та її адмін частину із всім набором функціональності;
- реалізувати функціонал онлайн фіскалізації чеків для користувачів (покупців) та адміністратора сайту;
- реалізація методів ведення онлайн звітності, а саме формування Х-звіту та Z-звіту;
- описати інструкцію та рекомендації щодо розгортання і налаштування проєкту, написання readme файлу.

РОЗДІЛ 1

АНАЛІЗ ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ ВЕБ ЗАСТОСУНКІВ ІЗ МОЖЛИВІСТЮ ФІСКАЛІЗАЦІЇ ЧЕКІВ І ПОСТАНОВКА ЗАВДАНЬ НА КВАЛІФІКАЦІЙНУ РОБОТУ

1.1 Аналіз сучасного стану проблеми

Світ не стоїть на місці, він постійно розвивається і це вже не новина. Не так давно, для того щоб щось купити, у людей не було інших варіантів як фізично ходити по магазинам, а в сьогодення нікого не здивуєш інтернет магазином. Це настільки звичне явище, що примітивний інтернет магазин сприймається як буденність навіть для студентських робіт.

Переважна більшість офлайн магазинів мають свою онлайн версію. Все це робиться не просто так, це відкриває велетенський потенціал в нових покупцях з різних частин країни, а іноді й світу. Підприємці отримують все більше онлайн замовлень і їхній грошовий потік все частіше стає залежним від онлайн інтернет магазинів. В деяких випадках, весь бізнес будується навколо Інтернету. Ви не прив'язані до місця дислокації ваших магазинів. Наразі це найпростіший спосіб масштабувати бізнес.

Прогрес зростання популярності онлайн покупок можна наочно побачити на графіку зображеному на рисунку 1.1 де зображено тенденцію по зростанню динаміки світових обсягів електронного роздрібного продажу станом за період 2014-2023 роках. Беручи до уваги результати досліджень по графіку, можна дійти висновку що із 2014 року по 2023 рік люди почали робити онлайн покупки фактично більше ніж у 5 разів. Цією інформацією не можна нехтувати, адже вона дає нам наглядно побачити в якому напрямку рухається комерція та комерційні операції.

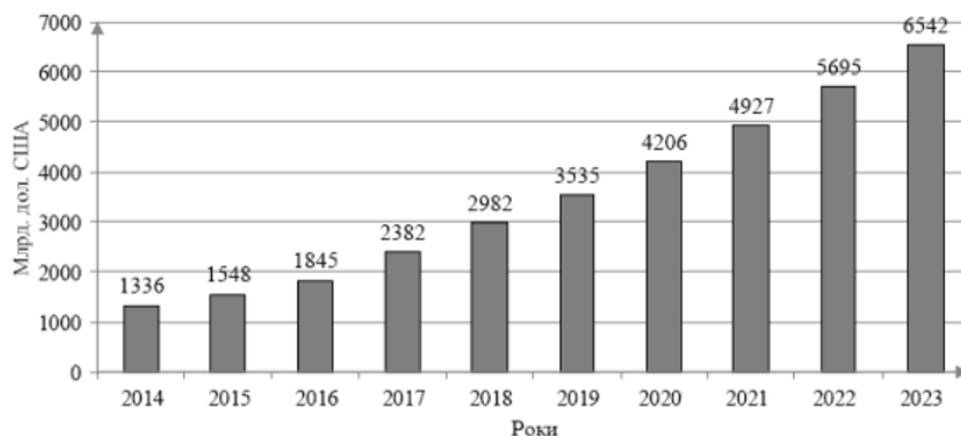


Рисунок 1.1 – Графік «Динаміка світових обсягів електронного роздрібного продажу станом за період 2014-2023 р.» [1]

Існує дуже велика кількість засобів і методів розробки веб-сайтів загалом, починаючи від CMS (таких як WordPress, Joomla та інші), закінчуючи повноцінними фреймворками (Laravel, Django та інші). Для створення простих і не дуже вибагливих по цілях веб-застосунків, цілком може вистачити і CMS, проте для нашого завдання такого скудного функціоналу не вистачає, нам потрібен повноцінний фреймворк.

Різновидів фреймворків досить багато, одні із найпопулярніших це: Laravel, Symfony та Django. Серед цього списку трішки виділяється фреймворк Django, адже він написаний на мові програмування Python, в той час як інші два на PHP. Проте він в порівнянні з іншими не набув настільки широкої популярності і в нього немає такої великої громади однодумців. Laravel і Symfony – це обидва дуже потужні бекенд-фреймворки які багато чим схожі, вони побудовані на одному і тому ж патерні MVC (Model View Controller). Цей патерн має таку архітектуру, що дозволяє виокремити в застосунку окремі логічні компоненти, що допомагає ефективно керувати даними і максимально спростити логіку для користувача і розробника.

MVC найперший патерн проєктування архітектури. Microsoft ще в 2009 році випустив цей каркас із відкритим кодом. У MVC поділяється на три складові, що відповідають за роботу з даними, бізнес-логікою та

користувачьким інтерфейсом. Можна виокремити три значні переваги такого підходу до проєктування структури:

- паралельна розробка;
- чіткий поділ по відповідальності кожного елементу;
- просте тестування.

Наслідком вибору для розробки фреймворку є необхідність в фронтенд частині розробки. Проте це не стає великою проблемою. Весь фронтенд буде побудований на стандартних засобах HTML, CSS та JavaScript із використанням препроцесору SASS, та JS бібліотеки jQuery.

Проте, не все так просто з інтернет магазинами. Інтернет суцільно переплітається із реальним життям, інтернет магазин отримує реальні гроші, а де відбувається обіг реальних фінансів, там і потрібно щоб все відповідало законодавству країни. Тобто, при операціях із грошовим потоком, законодавством України передбачено фіскалізація чеків та звітність у податкову, а саме Х-звіти та Z-звіти. Всі операції мають відбуватися прозоро. Саме всі ці операції більшість інтернет магазинів роблять офлайн, адже раніше реалізація такого функціоналу була неможливою і тільки нещодавно почали з'являтися можливості для опрацювання всіх цих даних онлайн.

Від нещодавна, а саме першого серпня 2023 року, було прийнято закон України №129-IX, який зобов'язує усіх підприємців видавати фіскальні чеки за покупки і надані послуги. Тепер всі товариства з обмеженою відповідальністю і прості особи підприємці повинні фіскалізувати чеки.

Перш ніж щоб чек став фіскальним, йому необхідно пройти процедуру під назвою – фіскалізація. Фіскалізація – це повний алгоритм дій по передачі звітів про власні продажі до податкових органів. Ця операція повинна відбуватися за участю Державної податкової служби.

Тож можна ототожнити, що фіскальний касовий чек – розрахунковий документ, зареєстрований в РРО (реєстратор розрахункових операцій) при проведенні розрахунків за продані товари (надані послуги) встановленої форми, який в обов'язковому порядку повинен бути виданий покупцю, на відміну від

товарного чеку, який видається за вимогою та заповнюється вручну [2]. А сам процес фіскалізації чеків – це процес передачі фіскальних даних операції до податкових органів.

Закон про обов’язкову фіскалізацію був прийнятий не так давно і на ринку ще немає запатентованого і повністю готового рішення по інтеграції пРРО (програмний реєстратор розрахункових операцій) в свої системи інтернет магазинів. Зараз є можливість розробити універсальний метод по автоматизації інтеграції сервісу до онлайн системи магазинів і запатентувати його. Зараз всі інтеграції відбуваються практично з нуля і з чистих запитів до API обраного сервісу.

Аналізуючи статтю «Development of world economy, marketing and management in modern conditions» [3] можна дійти висновку, що за допомогою РРО та програмного РРО здійснюється фіскалізація мережі автоматів самообслуговування: кавомашин, автоматів з водою, мийок, каси самообслуговування тощо. Тому керівництву підприємств або підприємцю потрібно обирати метод, залежно від технічних особливостей обладнання, бюджету та потреб (через онлайн сервіс, на базі міні-комп’ютера, або вбудований класичний РРО).

Всі програмні РРО працюють за схожою схемою. Спочатку формується чек, сервіс відправляє його на сервер податкової служби. Податкова служба присвоює кожному чеку свій фіскальний номер і надсилає чек назад на сервер веб-сайту. Після таких маніпуляцій чек вважається фіскалізованим і його можна надіслати покупцю.

Ось головні переваги пРРО перед традиційною офлайн звітністю:

- гнучкість і мультиплатформенність;
- економія часу та ресурсів (в тому числі фінансових);
- простота впровадження в різні види систем;
- відповідність законодавству України.

Більшість пРРО можуть працювати навіть в умовах відсутності світла і в режимі офлайн. В кожному сервісі і окремому тарифі є своя фіксована кількість чеків зроблена в офлайн режимі перед тим як відправити на сервер податкової, що є дуже зручним у використанні в умовах аварійних відключень електроенергії.

Весь процес від створення чеку на своєму сервері і до отримання його від податкової відбувається автоматично. Касиру лише потрібно оформити продаж товару або зафіксувати факт надання послуги і видати чек.

Важливо зазначити, що програмне забезпечення пРРО надається безкоштовно від податкової служби та доступне для встановлення на широкий спектр платформ, включаючи Windows, Android, iOS та WEB. Це робить пРРО доступним для широкого кола користувачів, дозволяючи платникам податків легко інтегрувати ці рішення у свою діяльність, оптимізувати процеси обліку та забезпечити високий рівень прозорості своїх фінансових операцій [4].

Враховуючи стан і правила сучасного документообігу, створення сайту із можливістю фіскалізації чеків є однією із важливих складових розвитку компанії і бізнесу.

В Україні є декілька сервісів які надають такі можливості. Вони мають відкрите REST API для інтеграції їх в будь-які проєкти. Трьома основними програмними РРО являються Checkbox, Cashalot та Вчасно.Каса. Всі вони мають свої особливості та переваги і недоліки перед іншими і кожен обирає під свої потреби та на свій смак. Вибір між цими сервісами залежить від функціональних вимог, конкретних потреб та бюджету. Всі вони мають безкоштовні версії, але розширені функції доступні лише для платних підписок.

Попри все, слід зазначити, що всі ці три сервіси програмного РРО надають зручний інтерфейс для онлайн фіскалізації та дозволяють автоматизувати цей процес.

Cashalot виділяється своїми розширеними можливостями та великим вибором налаштувань. Також можна відмітити простоту у використанні та

можливість обробляти великі обсяги даних продуктів. Позиціонують себе як сучасну альтернативу касовим апаратам (рис. 1.2).

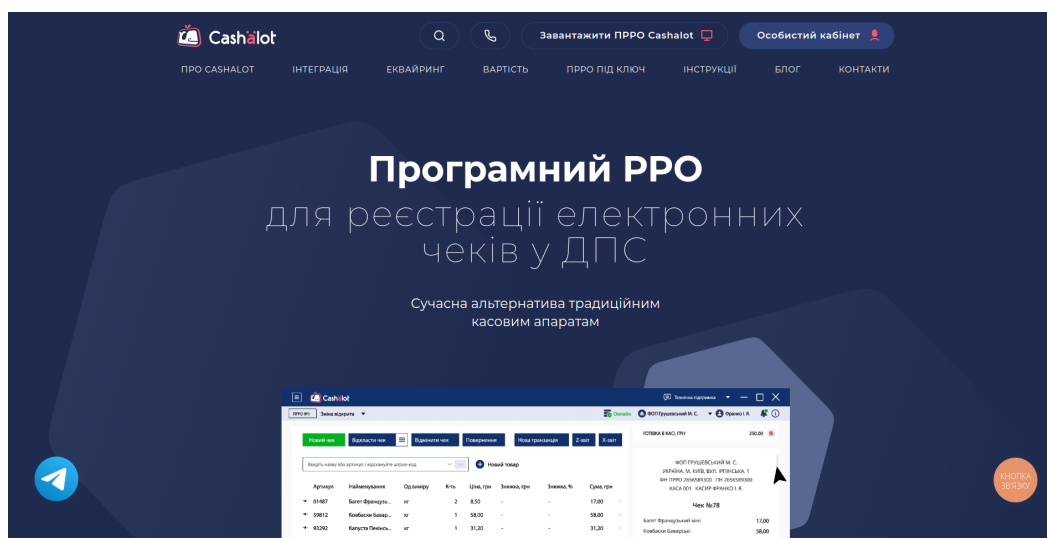


Рисунок 1.2 – Користувачський інтерфейс сервісу «Cashalot» [5]

Вчасно.Каса тільки недавно з'явилася на ринку, тому вона і відрізняється інноваційністю та новішими технологіями. Саме це забезпечує високу швидкість в роботі. Як стверджується на головній сторінці їхнього веб-сайту [6] вони видають чеки навіть без інтернету та при збоях АЦСК або ДПС (рис. 1.3).

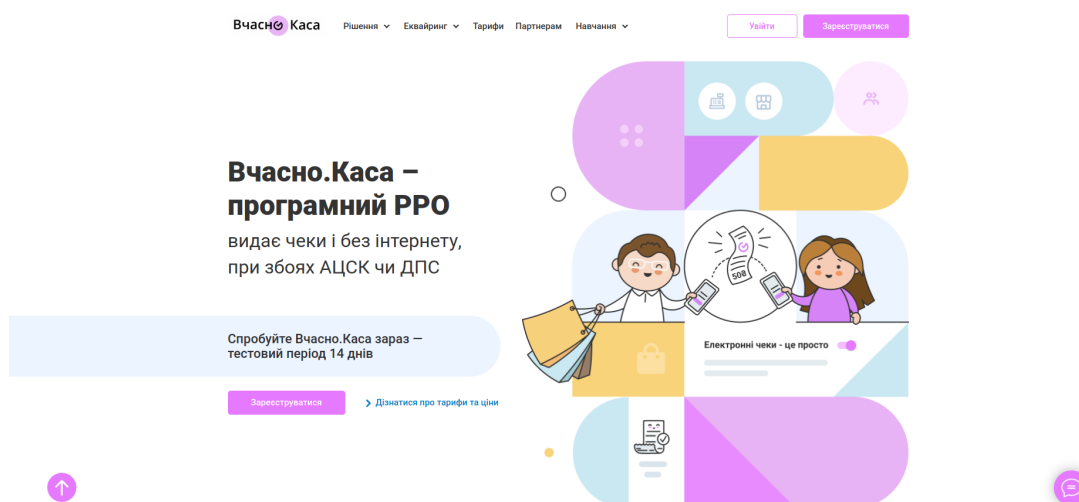


Рисунок 1.3 – Користувачський інтерфейс сервісу «Вчасно.Каса» [6]

Checkbox є одним із провідних сервісів які надають послуги онлайн фіскалізації чеків. Через цей сервіс зручно формуються всі необхідні звіти в податкову та він має легку та інтуїтивно зрозуміло описану документацію через це він і набуває свою популярність як серед розробників, так і бізнесу із інтернет магазинами (рис. 1.4).

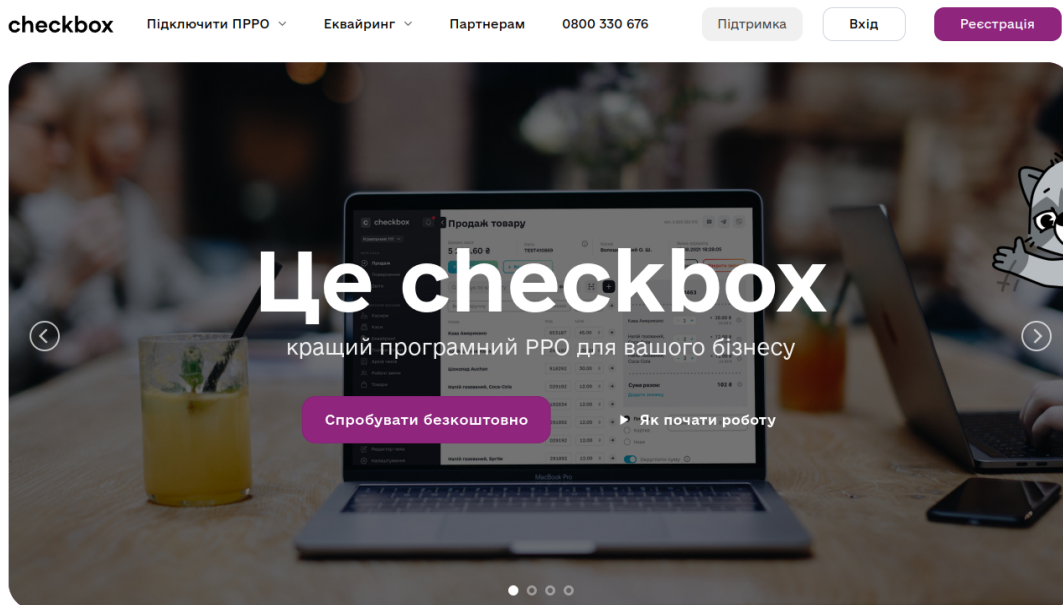


Рисунок 1.4 – Користувацький інтерфейс сервісу «Checkbox» [7]

Розглянувши всі ці сервіси програмного ПРРО, можна зробити вибір в сторону Checkbox, адже саме він як найкраще задовольняє вимоги до поставлених функціональних завдань системи, а саме:

- фіскалізація чеків і відправлення їх покупцям;
- формування Х-звіту в будь-який момент часу;
- формування Z-звіту в кінці кожної касової зміни;
- повна синхронізація між складами в базі даних веб-застосунку та сервісу програмного ПРРО.

Інтернет магазини стають не просто альтернативою для звичайних магазинів, а важливою складовою їхнього обігу товарообміну та економіки. Їх поширення спонукає світ активно розвивати сферу торгівлі та бізнесу. Це

забезпечує споживачам широкий вибір та зручність під час шопінгу і звичайних покупок продуктів незалежно від їхнього місця проживання.

Світ рухається і бізнесу треба рухатись разом із ним. Онлайн чеки повинні ставати все більш поширені серед інтернет магазинів, адже це не тільки спрощує всі бюрократичні процеси, а й зберігає довкілля, тому що задня бумажних чеків щороку вирубується дуже велика кількість дерев. Найбільш провідні інтернет магазини, такі як: Rozetka, Mozo, КСД вже перейшли на онлайн чеки та онлайн звітність в податкову і скоро це буде відбуватись на всіх сайтах.

Однією з головних особливостей електронної комерції є її сильна залежність від технічних можливостей підприємця, надійності та рівня програмного забезпечення із підтримки та налагодження веб-сайтів [8].

Таким чином, розвиток цього напрямку впровадження онлайн фіскалізації чеків стає необхідністю для інтернет магазинів. Тому треба бути готовим до потреб ринку і вже визначати найкращі шляхи для реалізації відповідного функціоналу.

1.2 Постановка завдання на кваліфікаційну роботу бакалавра

Все наше завдання можна розділити на низку цілей, які будуть ключовими для вирішення завдання та досягнення мети роботи. Дані цілі, які мають бути досягнуті за результатом виконання роботи, можна сформулювати так:

- 1) вивчити питання та проаналізувати існуючі сервіси та методи інтеграції онлайн фіскалізації чеків в інтернет магазинах, а саме на сервісах Checkbox, Cashalot та Вчасно.Каса;
- 2) визначити вимоги для онлайн фіскалізації чеків та звітів в податкову і проаналізувати досвід провідних інтернет магазинів в цьому питанні;
- 3) розробити схему бази даних та обдумати методи для ефективного впровадження онлайн фіскалізації чеків;

- 4) реалізувати інтернет магазин та її адмін частину із всім набором функціональності;
- 5) реалізувати функціонал онлайн фіскалізації чеків для користувачів (покупців) та адміністратора сайту;
- 6) реалізація методів ведення онлайн звітності, а саме формування Х-звіту та Z-звіту;
- 7) описати інструкцію та рекомендації щодо розгортання і налаштування проєкту, написання readme файлу.

Після досягнення всіх цих цілей, завдання можна буде вважати виконаним. Це дозволить нам добре дослідити цю тему та в майбутньому з легкістю впроваджувати такі масштабні системи в різні інтернет магазини із врахуванням всіх побажань та вимог.

В результаті у нас повинен бути розроблений інтернет магазин із каталогом товарів та блогом, адміністративна частина (адмінка) сайту із повним контролем всіх процесів в магазині, реалізований функціонал фіскалізації та податкової звітності. Обіг чеків та звітність можна буде контролювати безпосередньо як на сайті інтернет магазину, так і на сайті сервісу програмного РРО. Чеки повинні надсилатись на електронну пошту як покупця, так і продавця. Х-звіти та Z-звіти також повинні надсилатись на пошту адміністратора магазину.

Висновки до розділу 1

У висновку можна сказати що поки переважна більшість інтернет магазинів залишаються примітивними дошками оголошень, на яких тільки картинка товару. Виявлено, що багато інтернет магазинів стикаються із проблемами визваними з необхідністю фіскалізувати всі замовлення, що донедавна робилось вручну, вже після успішної інтернет покупки. В наш час вже почали з'являтися і використовуватись системи для вдосконалення таких

сайтів, також для автоматизації великої кількості процесів, в тому числі і фіскалізації чеків із всіма витікаючими моментами, такими як ведення звітності.

Узагальненим завданням роботи є розробка інтернет магазину, її клієнтської та адміністраторської частини та розробка програмного забезпечення для онлайн фіскалізації чеків і формуванні податкових звітів.

РОЗДІЛ 2

СПЕЦИФІКАЦІЯ ВИМОГ ДО РОЗРОБЛЮВАНОЇ СИСТЕМИ

2.1 Аналіз, визначення вимог до розроблюваного програмного забезпечення та проектування програмного забезпечення

Для того щоб зрозуміти що повинен робити інтернет магазин, можна наглядно розписати всі які можливості він повинен мати щоб задовольнити описані раніше цілі, а саме:

- авторизація;
- менеджерська частина сайту (адмінка);
- блог і статті;
- швидкий пошук по сайту;
- можливість оформити покупку;
- оплата онлайн;
- отримання фіскального чеку;
- формування X та Z-звітів.

Це головні задачі які повинен виконувати інтернет магазин. Цей список ще можна продовжувати такими задачами як можливість обирати продукти і статті в обрані, зміна статичних даних через адміністративну частину, створення нових сторінок безпосередньо через сайт і вбудований редактор, але це все в порівнянні із головними, другорядні задачі.

Щоб краще зрозуміти як має відбуватись фіскалізація і поетапність роботи, було створено діаграму послідовності, в якій описується весь процес від оформлення замовлення до отримання клієнтом його фіскального чеку. Саму діаграму можна побачити на рисунку 2.1.

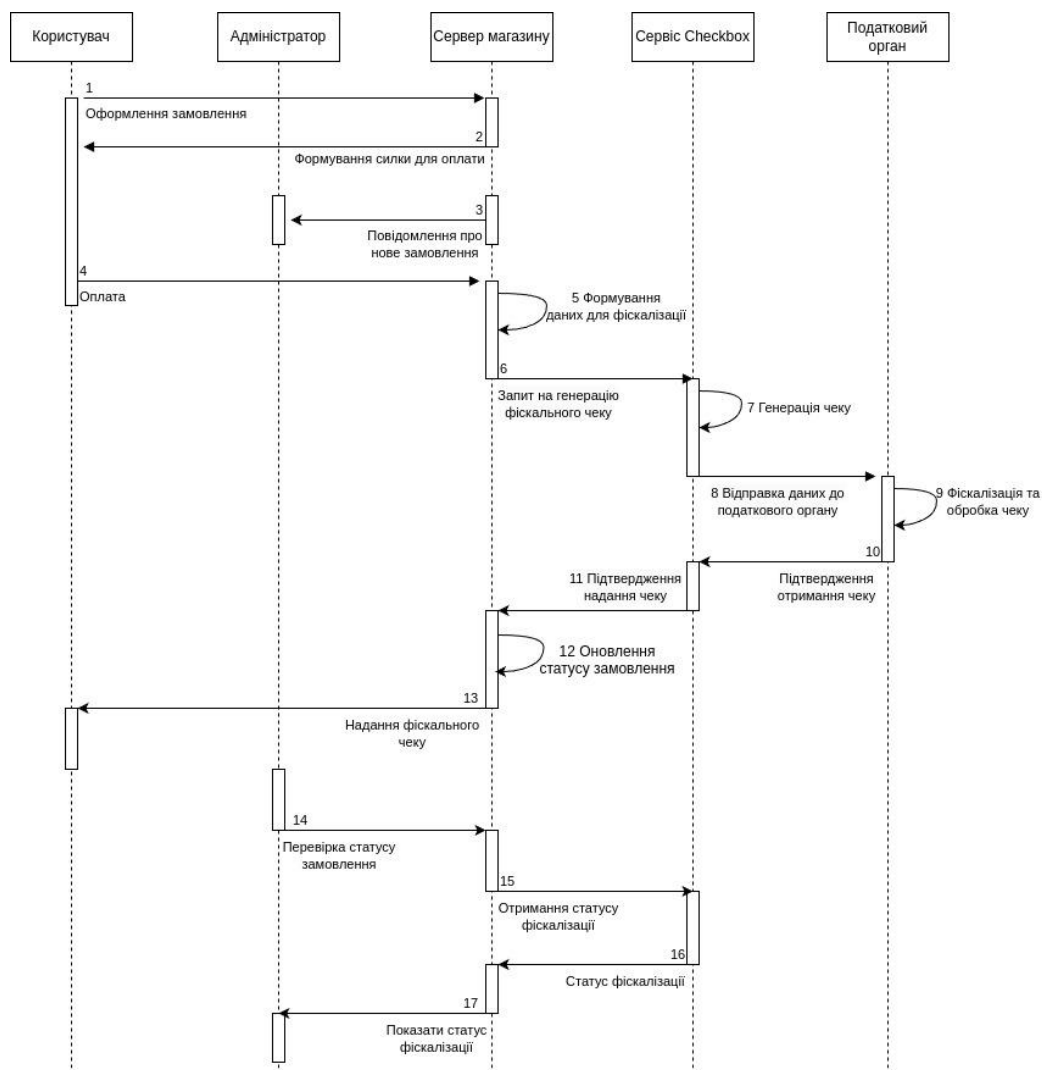


Рисунок 2.1 – Діаграма послідовності процесу фіскалізації чеку

Веб-сайт повинен мати простий, але і в той час гарний зовнішній вигляд і інтуїтивно зрозумілий користувацький інтерфейс. Для того щоб мати представлення який має мати вигляд веб-сайт було розроблено UX/UI-дизайн в застосунку Figma. На рисунку 2.2 зображено макет десктопної версії магазину разом із адміністративною частиною.

Адміністративна частина буде розроблятися за допомогою бібліотеки Bootstrap, тому і відповідно макет розроблявся із урахуванням цього, тому вся адмінка умовно поділена сіткою на 12 стовпців, що відповідає стандартам Bootstrap.

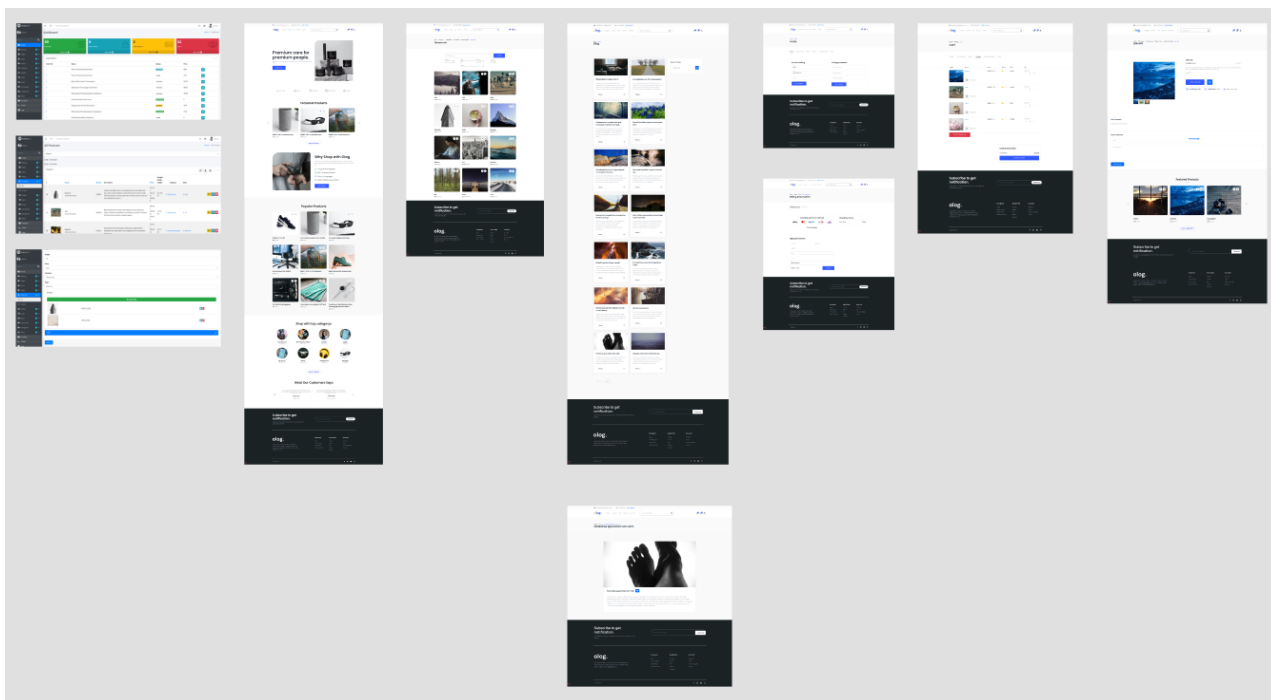


Рисунок 2.2 – Макет сайту

Всі сторінки мають однакову стилістику і добре розміщені пункти навігації. В адмінці одразу видно найголовніші функції які там є, а саме робота із замовленнями. Також вся функціональність розміщена інтуїтивно зрозуміло, тому люба функція знаходиться дуже швидко.

2.2 Вибір засобів, методів і алгоритмів вирішення поставленого завдання

Проаналізувавши поставлені цілі, які були наведені в минулому розділі, було проведено аналіз і вибір найбільш доцільних методів розробки та стеку які задовольняють наші потреби.

В першу чергу потрібно обрати підхід для створення веб-сайту інтернет магазину між CMS та повноцінним фреймворком. Як було сказано в розділі 1, CMS не задовольняє наші потреби, то результатом вибору побудови сайту на повноцінному фреймворку є потреба у виборі між підходом розробки до фронтенд і бекенд частин.

Для фронтенд частини найлогічніше було взяти чистий HTML, CSS та JavaScript. Лише для прискорення розробки буде використано препроцесор SASS та бібліотека jQuery. Такий стек для розробки нашого веб застосунку цілком задовольняє наші потреби, адже цей підхід є швидким в розробці і оптимізованим до використання.

Бекенд частина сайту буде написана використовуючи PHP фреймворк. Вибір між Laravel і Symfony можна робити тільки виходячи із конкретних цілей і задач, адже тільки так можна зрозуміти що краще обрати. Laravel це більш гнучке рішення під величезну кількість проблем. Саме на цьому фреймворку написанні більшість веб-застосунків. Тому Laravel нам якнайкраще задовольняє нашу задачу, тому що він має хорошу підтримку і структуру побудови коду.

При тому є ще декілька підходів розробки сайту на шаблонізаторах і на API. При шаблонному підході, спочатку робиться верстка сайту, а потім бекенд натягується поверх верстки за допомогою шаблонізатору. В Laravel це шаблонізатор Blade. Він вже поверх готової гіпер-розмітки HTML виконує натягування бекенду за допомогою своїх директив. Другий підхід передбачає написання REST API всього функціоналу веб-сайту і взаємодія фронтенд частини із API виконуючи запити до нього.

Ці два підходи є дуже хорошою практикою, тому ми використаємо комбінований, а саме для адміністративної частини використаємо шаблонний підхід, а для клієнтською буде написання REST API.

Базою даних було обрано MySQL. Вона має готовий шаблон підключення в фреймворку Laravel і все що залишається, це ввести свої дані для входу. Дана база даних є простою і інтуїтивно зрозумілою із дуже схожим на англійську мову синтаксис. Взаємодіяти із базою будемо із сервісом Adminer. Саме цей сервіс є більш оптимізованим в порівнянні із іншим популярним сервісом Phpmyadmin.

Для фіскалізації чеків і взаємодії із податковими органами, відштовхуючись від задач які були покладені, я обрав сервіс Checkbox. Саме він буде якнайкраще влаштовувати наші потреби. Також він має приємний

користувацький інтерфейс та добре описану документацію, що є дуже важливим під час процесу розробки.

Висновки до розділу 2

Можна ототожнити, для реалізації поставлених задач було обрано такий стек технологій і засобів:

- UI/UX дизайн – Figma;
- фронтенд – HTML, CSS, SASS, JavaScript та jQuery;
- бекенд – фреймворк Laravel;
- підхід розробки – шаблонізатори Blade та REST API;
- база даних – MySQL;
- сервіс фіскалізації – Checkbox.

Весь цей набір засобів і інструментів підходить нам якнайкраще для розробки і виконання поставлених задач. Кожен елемент чудово взаємодіє один з одним і всі доповнюють одне одного.

РОЗДІЛ 3

РОЗРОБКА ІНТЕРНЕТ МАГАЗИНУ ІЗ МОЖЛИВІСТЮ ФІСКАЛІЗАЦІЇ ЧЕКІВ

3.1 Практична реалізація об'єкта проєктування

Згідно із поставленими цілями в першому розділі, зараз настав час розробити схему бази даних. Цей етап є одним із найголовніших, адже саме від бази даних буде залежати вся майбутня розробка і сам проєкт загалом. Схема бази даних повинна бути добре обдуманого та розширювальною. При хорошій побудові бази даних, вся швидкодія сайту буде справлятися із поставленими завданнями спритніше.

Для побудови схеми бази даних будемо використовувати програму MySQL Workbench. Ця програма весь необхідний нам функціонал та є професійним рішенням для такого роду задач.

При побудові схеми важливо враховувати всі зв'язки між сутностями і грамотно їх проставляти, щоб база даних була максимально нормалізованою. Допускається незначне відхилення нормалізації, але лише у тих випадках де це позитивно впливає на швидкодію та не шкодить іншим функціям.

Фактично база даних для нашого інтернет магазину має три головні сутності, це: користувач (user), товари (products) та замовлення (orders). Вони є головними дійовими особами любого інтернет магазину і вже від цього все будується.

Вже при побудові схеми бази даних, потрібно врахувати що медіа-файли такі як фотографії, відео, файли ми будемо зберігати в окремій таблиці, а не безпосередньо в таблиці до якої вони відносяться. Тобто фотографії товару будуть зберігатися не в таблиці «Products», а в таблиці «Media». Всі ці медіа-файли будуть зв'язуватися із потрібною сутністю через поліморфний зв'язок. Такий підхід оптимізовує роботу бази і робить її більш нормалізованою. Саму схему можна побачити на рисунку 3.1.

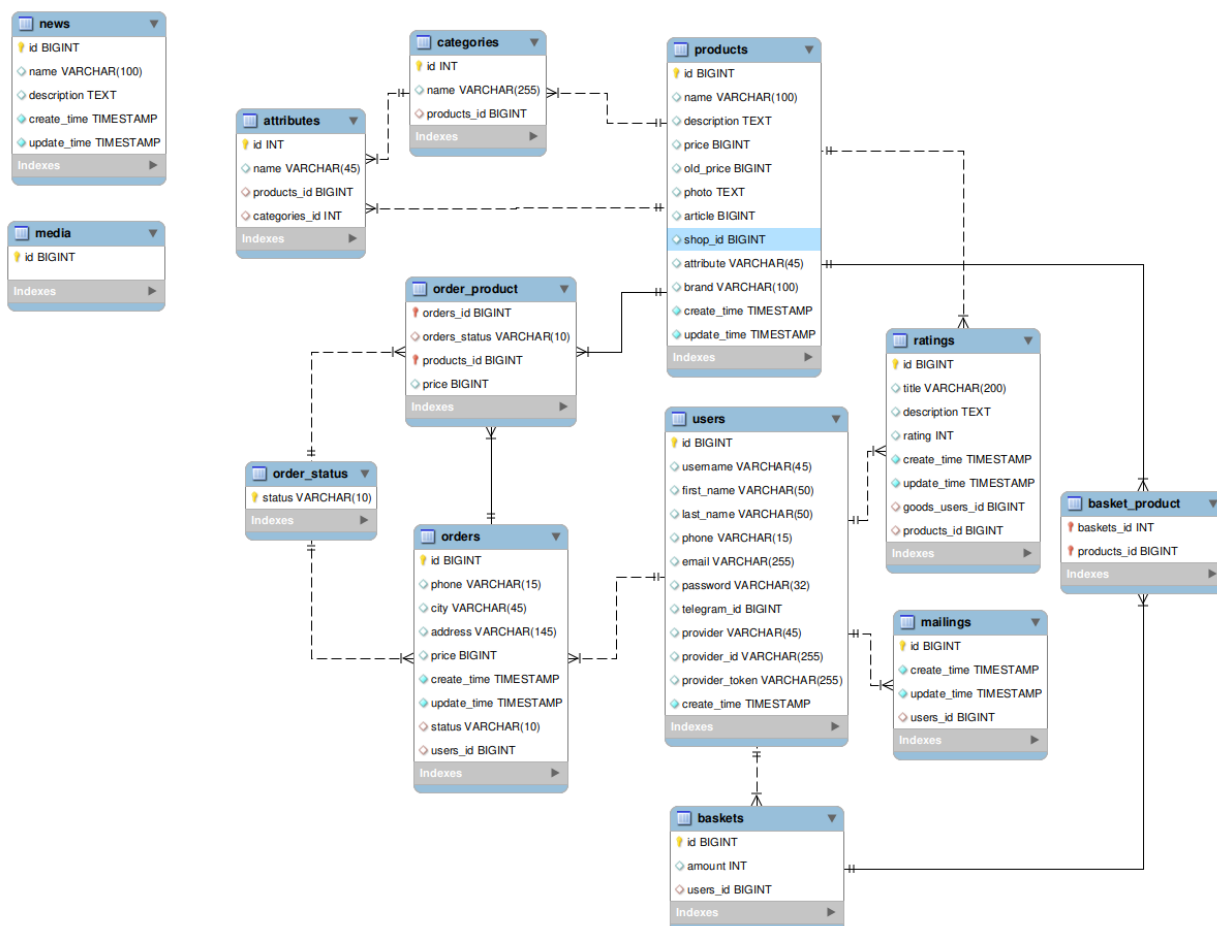


Рисунок 3.1 – Схема бази даних

На схемі чітко видно всі сутності та зв'язки між ними. Вся схема побудована згідно вимог до програмного продукту і якнайкраще підходить під наші задачі. За такою схемою вся наша система є легко масштабуємою і при необхідності всі нові побажання будуть легко інтегровані у робочу систему.

Після побудови схеми, можна створювати пустий проєкт Laravel та описувати базу і зв'язки безпосередньо в проєкті. Створення пустого проєкту Laravel відбувається через пакетний менеджер Composer. Це пакетний менеджер для мови програмування PHP, яка дозволяє структурувати і використовувати різні пакети при розробці. Composer це дуже потужний засіб, який використовується практично у всіх проєктах на PHP.

Для того щоб встановити Composer та створити новий проєкт Laravel потрібно виконати такі команди, які зображені на лістингу 3.1. Враховуючи що,

для розробки ми використовуємо UNIX-подібне середовище, а саме Kubuntu, то і встановлення буде відбуватись через консоль.

Лістинг 3.1 – Створення нового проєкту Laravel

```
sudo apt-get update
sudo apt-get install curl
sudo apt-get install php php-curl
curl -sS https://getcomposer.org/installer -o composer-setup.php
sudo php composer-setup.php --install-dir=/usr/local/bin
    --filename=composer
sudo composer self-update
composer create-project laravel/laravel shop-fiscal
```

Кінець лістингу 3.1

Тепер коли вже є створений новий проєкт та намальована схема бази даних, потрібно описати всі залежності БД. Все це робиться дуже легко коли є зарання побудована схема. Все що потрібно, це відштовхуючись від неї, описати все на рівні фреймворка. Враховуючи лаконічно описану документацію та хорошу архітектуру фреймворку, це не викликає великих проблем. Навіть більш складні поліморфні зв'язки описуються і використовуються дуже легко. Проте у нашому проєкті знадобиться тільки один поліморфний зв'язок, це зв'язок для улюблених, адже в улюблені можна буде додавати не тільки товар, а і новини в блозі.

За приклад опису зв'язку в Laravel можна взяти сутності «Користувач» і «Замовлення». Це буде зв'язок «один до багатьох», це означає що одна сутність може мати зв'язок із багатьма іншими моделями іншої сутності. В нашому випадку це один користувач може мати багато замовлень, але одне замовлення відноситься тільки до одного користувача. В фреймворку це описується, як користувач має багато замовлень («hasMany»), а замовлення відноситься до користувача («belongsTo»). Приклад цього зв'язку зображено на лістингу 3.2.

Лістинг 3.2 – Зв'язок користувача до замовлень

```
Class App\Models\User
public function orders(): \Illuminate\Database\Eloquent\Relations\HasMany
{
    return $this->hasMany(Order::class);
}

Class App\Models\Order
public function user(): \Illuminate\Database\Eloquent\Relations\BelongsTo
{
    return $this->belongsTo(User::class);
}
```

Кінець лістингу 3.2

Для того щоб реалізувати роути і описати всі url потрібно їх описати в файлі «web.php». Проте, враховуючи що ми розробляємо не звичайний веб-сайт, а великий інтернет магазин із розробкою на Blade і REST API, то будемо використовувати 5 файлів:

- 1) «web.php» – головний файл в якому описуються роути для клієнтської частини яка побудована на шаблонізаторі Blade і підключаються інші чотири файли;
- 2) «web-auth.php» – файл в якому описуються роути авторизації користувача;
- 3) «api.php» – файл в якому описуються роути для клієнтської частини, побудованої на REST API;
- 4) «webhook.php» – файл в якому описуються роути для вебхуків, такі як вебхук від платіжної системи;
- 5) «admin.php» – файл в якому описуються роути для адміністративної частини сайту із всіма CRUD роутами.

В файлах роутів описується не тільки url по якому буде доступний той чи інший контент, а і посередники («Middleware»), такі як:

- «Guest» – посередник який пускає в конкретний роут тільки не авторизованих користувачів;

- «Auth» – посередник, протилежний до «Guest», пускає на роут тільки авторизованих користувачів;
- «Admin» – самописний посередник, який пускає на роути тільки менеджерів сайту.

Тепер лишилось тільки описати всю взаємодію із запитами і бізнес логіку. Найважливішою бізнес логікою в інтернет магазинах є функціонал корзини. Вона має велику кількість перевірок і потрібно її завжди тримати в активному стані.

Щоб користувач міг бачити сайт, потрібно пройти весь шлях, спочатку по url користувач звертається до роута, в роуті прописаний контролер до якого він звернеться, а вже у контролері формуються всі дані і іде взаємодія із вхідними даними. Якщо це розробка на REST API, контролер вертає JSON із даними, які вже проходять парсинг на фронтенд частині. У випадку розробки на шаблонізаторах Blade, то дані формуються і відправляються на «View», де вже і виконується розмітка і вивід даних із готових змінних. На рисунку 3.2 зображено вже готову головну сторінку сайту.

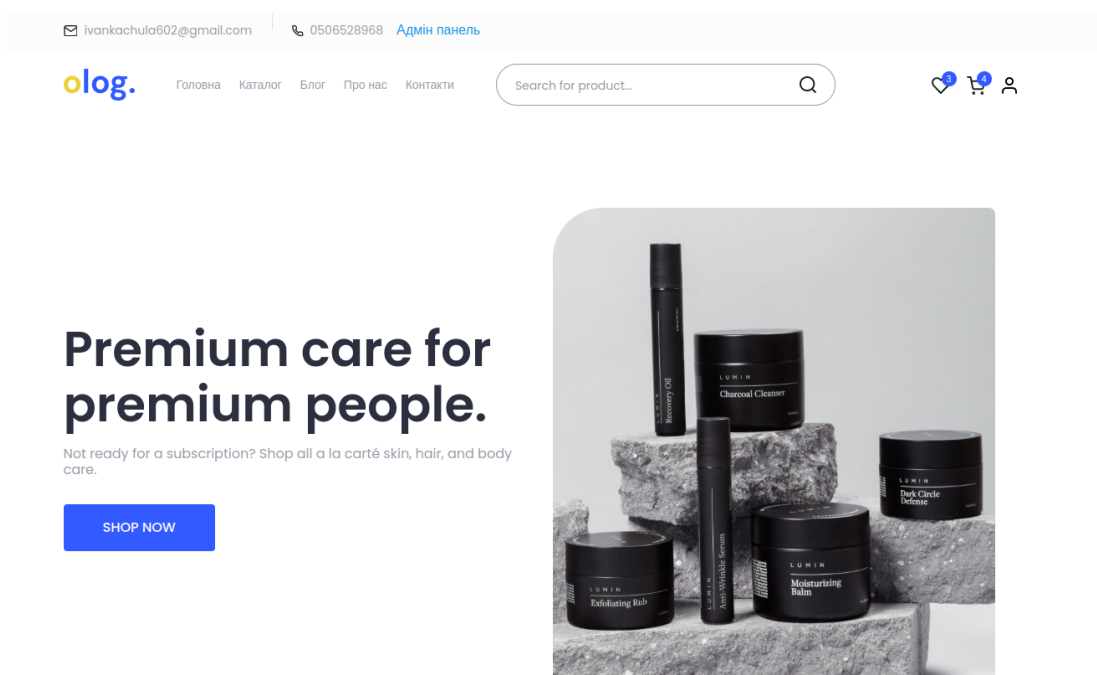


Рисунок 3.2 – Головна сторінка сайту

Слідкуючи за всіма хорошими практиками Laravel, а саме тонкі контролери, товсті моделі, ми не будемо описувати бізнес логіку в контролерах. Вся бізнес логіка буде описана в сервісних класах і для більш потужних функціональностей у фасадах. У них буде описана вся реалізація і у контролерах буде просто визиватися весь цей функціонал.

Для того щоб зробити корзину більш оптимізованою і зменшити навантаження на базу даних, ідентифікатор корзины будемо зберігати в cookie браузера. Ідентифікатор буде отримуватись із локального середовища без зберігання на сервері, а вже із цього ідентифікатора ми легко віддамо дані корзины і товарах в ній з бази.

На рисунку 3.3 зображено який має вигляд товари який був доданий в корзину.

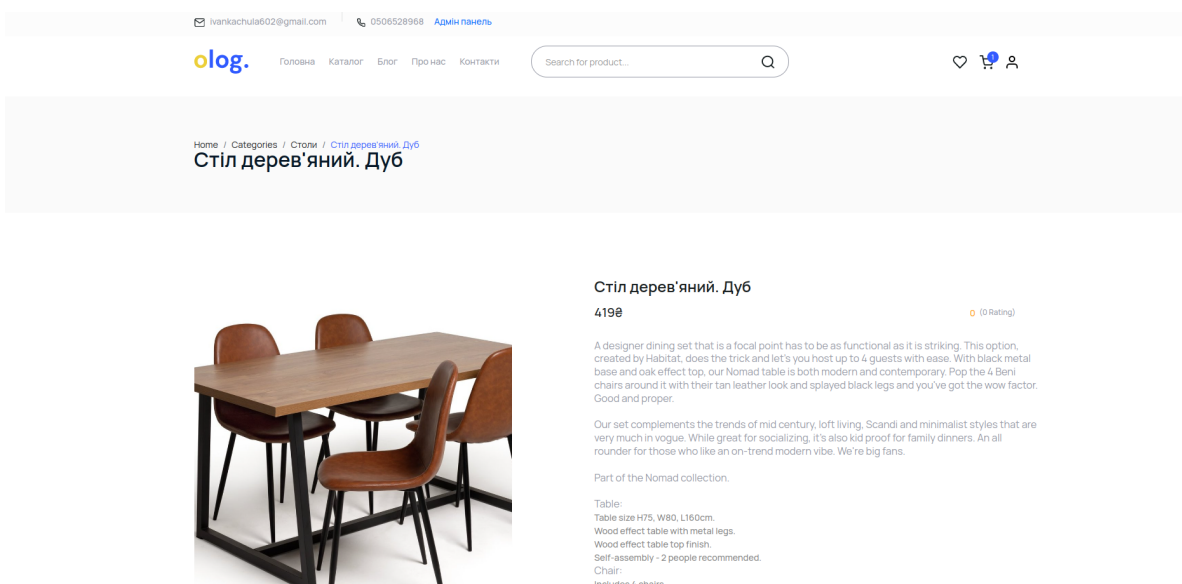


Рисунок 3.3 – Сторінка товару який доданий в корзину

В адміністративній частині сайту є всі функції, які потрібні в інтернет магазинах, від додавання нових продуктів в магазин до оформлення замовлення через адмінку. Адмінка має досить простий вигляд щоб не навантажувати систему і виконувати функції максимально швидко, також це дозволяє швидко знаходити всі необхідні функції.

Дуже зручно винесено все меню в ліву бокову частину сайту і одразу зрозуміло де, та що потрібно шукати. Також є можливість переглядати логи і користуватись системною консоллю `tinker` не заходячи прямо на сервер, де розташований репозиторій із веб-сайтом.

В окрему групу винесені всі динамічні зміни на сайті, такі як: телефони, емейли, вся контактна інформація і копірайт сайту. Можна динамічно змінювати інформацію в футері, не міняючи розмітки HTML. Це дуже полегшує роботу коли треба змінити будь-які статичні дані, все що потрібно, це зайти в адмінку та змінювати їх там, а не лізти на віддалений сервер і міняти файли з відкритим кодом.

На рисунку 3.4 можна побачити вигляд сторінки в адміністративній частині сайту із всіма продуктами.

#	Name	Article	Description	Price	Weight, Width, Height	Category
104	Habitat Recycled Cosy Plain Shaggy Rug 2024-05-27 18:12:12	9342381	<p>Who knew recycling could feel this good? Sumptuously soft underfoot with a long pile, this cosy natural rug is made using recycled plastic bottles. 128 of them to be precise. The perfect way to...	95 € 2.37 \$ 2.20 €	28 . Ролики	
103	Habitat Malta 4 Seater Rattan Effect Garden Bar Set - Natural 2024-05-27 18:12:12	1242245	<p>Take the party outdoors with Malta. A 4-seater swishy bar set that'll add flair to your patio. Each chair flaunts a rattan and rope effect seat with matching cushion for a dash of boho chic. W...	650 € 16.24 \$ 15.05 €	28 . Ролики	
102	Habitat Nomad 2 Door 2 Drawer Wardrobe 2024-05-27 18:12:11	8491594	<p>Industrial heritage with to die-for looks. Our wardrobe evokes loft living and authentic industrial styling with metal accents and an oak-effect finish. The 2-door, 2-drawer wardrobe has plenty of...	300 € 7.50 \$ 6.94 €	28 . Ролики	

Рисунок 3.4 – Сторінка всіх товарів на адміністративній частині сайту

Щоб реалізувати головний функціонал фіскалізації чеків, було застосовано підхід схожий до корзини. Було створено фасад і контролер який буде працювати із вхідними даними. Проте це найскладніший функціонал і цього набору нам не вистачить, тому було створено ще декілька «Laravel action», в яких структуровано по різним файлам відбувається реалізація різних функцій, такий як створення чеку, або звіту.

Тож як відбувається весь процес фіскалізації із середини, враховуючи підхід до написання коду:

- 1) користувач звертається до сторінки до роуту;
- 2) контролер отримує всі надіслані дані і звертається до фасаду;
- 3) фасад робить з'єднання із API Checkbox і виконує «laravel action»;
- 4) action виконує конкретний запит до вже відкритого з'єднання API Checkbox і видає результат;
- 5) всі дані повертаються в контролер і він оперуючи ними видає відповідь із чеком.

На лістингу 3.3 можна побачити реалізацію самого функціоналу фіскалізації чека в «Laravel action».

Лістинг 3.3 – Реалізація фіскалізації чеку

```
public function handle(Order $order)
{
    $order->load('orderProducts.product');
    $api = $this->initApi();
    $cashier = $api->getCurrentCashierShift(true);
    if (Arr::get($cashier, 'status') !== 'OPENED') {
        Log::warning('CHECKBOX: Зміну не відкрито');
        return false;
    }
    $data = [
        'id' => Str::uuid(),
        'goods' => $this->getGoods($order),
        'delivery' => $this->getDelivery($order),
        'payments' => $this->getPayments($order),
        'total_sum' => \intval($order->getTotal() * 100),
    ];
    $res = $api->createReceipt($data);
    return $res;
}
```

Кінець лістингу 3.3

На лістингу 3.3 можна побачити що ми в першу чергу підгружаємо всі дані, це робиться для оптимізації процесу взаємодії із базою даних. Потім ініціюємо з'єднання з API Checkbox через фасад і через нього же перевіряємо

чи відкрита робоча зміна. Якщо зміна не відкрита, то повертаємо із помилкою де вказуємо на це, тому що фіскалізація неможлива при закритій зміні. Якщо зміна відкрита, то формується масив із даними і відправляється у фасад, який вже додасть у header запиту всі токени авторизації і відправить підготовлені дані у Checkbox де чек вже і фіскалізується і повернеться назад.

Чек можна отримати натиснувши на жовту кнопку з іконкою чека, яку зображено в формі на рисунку 3.4. Ця форма знаходиться на адміністративній частині сайту, на сторінці всіх замовлень.

13	12. Вікторія Євгенівна Мірошніченко 2024-05-26 18:31:03	new	Хмельницький	Інститутська	+380923898269	  
14	8. Панасюк Софія Олександрівна 2024-05-26 18:31:03	cancelled	Запоріжжя	Михайла Грушевського	0677043722	  
15	11. Богдан Андрійович Тарашук 2024-05-26 18:31:03	cancelled	Харків	М. Коцюбинського	+380976335653	  
1	20. Пономаренко Єлизавета Федорівна 2024-05-26 18:31:02	shipped	Полтава	М. Коцюбинського	+38(0025)947961	  

Рисунок 3.4 – Вигляд таблиці замовлень

На чекові є вся необхідна інформація як і в звичайних бумажних чеках, а саме інформація про товари, компанію, касира, дату, час та ціну. Загальний вигляд і додатковий текст можна налаштовувати в графічному інтерфейсі сервісу Checkbox.

На самому верху чеку зображено назву, місце знаходження і ID організації. Трішки нижче пішов перелік товарів. Одна позиція складається із штрих-коду товару, назви, кількості штук, ціна за штуку, загальна ціна, знижка, остаточна ціна позиції. Далі ідуть дані по оплаті, а саме тип оплати, якщо це оплата карткою, то маска картки, платіжна система. Нижче знаходиться загальна сума замовлення. І в самому кінці інформація про сам чек, а саме номер чеку, дата його створення, вид створення онлайн чи офлайн, фіскальний номер ПРРО і справої сторони QR-код в електронний кабінет платника податків, в якому можна перевірити чек вже в державному органі. На рисунку 3.5 можна побачити який має вигляд чек який вже повертається із Checkbox.

Тестова організація
М.КИЇВ ГОЛОСІЇВСЬКИЙ Р-Н, Тестова, 41а
ІД test_jEatU5J5SO

Штрих-код: ENIQ-male	10 x 2716.56	27165.60
Подушка ортопедична		
Знижка		-55.44
Сума зі знижкою		27110.16

Банк	PB
Продаж	
ЕПЗ	***4441
ПЛАТІЖНА СИСТЕМА	liqpay
Картка	27110.16 ГРН
Сума	27110.16 ГРН

Чек № TEST-6Hkk1e
22.12.2023 об 11:59:47
ОНЛАЙН

ФН ПРРО TEST760312
ТЕСТОВИЙ ЧЕК
checkbox



Рисунок 3.5 – Готовий чек на замовлення

Для того щоб виконати будь-який із звітів, потрібно в адміністративній частині, на формі із всіма замовленнями натиснути на відповідну до звіту кнопку, на рисунку 3.6 зображені ці кнопки. В додатку А наведений код функціоналу реалізації створення різного типу звітів.

All Orders Home / All Orders

Orders X-звіт Z-звіт — X

Рисунок 3.6 – Кнопки для створення звітів

X-звіти можна робити будь коли, якщо каса відкрита і його створення є по суті тільки заради в інформативних інтересах. Вигляд X-звіту зображено на рисунку 3.7.

```

Тестова організація
УКРАЇНА, М.КИЇВ ГОЛОСІЇВСЬКИЙ Р-Н,
Тестова, 41а
ІД test_P1bT6JK3le
X-звіт №6
Номер каси 1
за 08.06.2024
-----
Зміна відкрита 07.06.2024 20:30:36
Останній фіскальний чек
08.06.2024 00:36:40
Номер останнього чеку 32
Фіскальний номер чеку TEST-zny6Bh
Валюта звіту ГРН
----- Реалізація -----
Одержано по формах оплати
Картка 155705.00
Загальний оборот 155705.00
Кількість чеків 32
----- Повернення -----
Видано по формам оплати
Загальний оборот 0.00
Кількість чеків 0
----- Готівкові кошти в касі -----
Початковий залишок 0.00
Службове внесення 0.00
Службове вилучення 0.00
Кінцевий залишок 0.00
----- Виручка -----
Готівкою 0.00
Безготівкова 155705.00
Всього 155705.00
-----
ФН ПРРО TEST935830
ТЕСТОВИЙ ДОКУМЕНТ ЧЕСКВ0X
08.06.2024 00:46:43

```

Рисунок 3.7 – Вигляд X-звіту

Сам X-звіт являє собою сукупну інформацію про створені чеки за певний період часу. В цьому звіті міститься ID чека, його тип (продаж чи повернення), дата та час, фіскальний номер, тип оплати, сума в грн, сума з урахування заокруглення, локальний номер ПРРО, вид чеку (онлайн чи офлайн), знижка на замовлення, посилання на чек.

А от Z-звіт можна створювати тільки один раз в кінці зміни, тому що саме ці дані будуть відправлені в податковий орган. Цей звіт вже є більш обширним і із загальними даними. Вигляд Z-звіту зображено на рисунку 3.8.

адреса каси	Дата відкриття	Дата закриття	Загальна к-т	Фіскальний	Фіскальний	Номер зміни	Виручка	Останній фін	Кількість че	Кількість че	Сума пдв	Ставки подат	Назва подат	Суми подат	Оборот по	Суми подат	Оборот по	Форми оплат	Одержано г	Видано по о	Початковий	Кінцевий за	Службові в	Службові в	Виручка без П
КРАІНА, М.	2023-12-21	2023-12-22	16	TEST760312	TEST-8xxxx	14	277583	21-12-23 14	16	0	0.00	0.0%-0%	Без ПДВ з	0.00	0.00	0.00	0.00	Карта	277583.00	0.00	212.1	212.1	0	0	277583
КРАІНА, М.	2023-12-21	2023-12-21	2	TEST760312		13	2960	21-12-23 12	2	0	0.00	0.0%-0%	Без ПДВ з	0.00	0.00	0.00	0.00	Карта	2960.00	0.00	212.1	212.1	0	0	2960
КРАІНА, М.	2023-12-21	2023-12-21	1	TEST760312		12	1480	21-12-23 11	1	0	0.00	0.0%-0%	Без ПДВ з	0.00	0.00	0.00	0.00	Карта	1480.00	0.00	212.1	212.1	0	0	1480
КРАІНА, М.	2023-12-21	2023-12-21	0	TEST760312		11	0		0	0	0.00	0.0%-0%	Без ПДВ з	0.00	0.00	0.00	0.00				212.1	212.1	0	0	0
КРАІНА, М.	2023-12-21	2023-12-21	0	TEST760312		10	0		0	0	0.00	0.0%-0%	Без ПДВ з	0.00	0.00	0.00	0.00				212.1	212.1	0	0	0
КРАІНА, М.	2023-12-21	2023-12-21	0	TEST760312		9	0		0	0	0.00	0.0%-0%	Без ПДВ з	0.00	0.00	0.00	0.00				212.1	212.1	0	0	0
КРАІНА, М.	2023-12-21	2023-12-21	0	TEST760312		8	0		0	0	0.00	0.0%-0%	Без ПДВ з	0.00	0.00	0.00	0.00				212.1	212.1	0	0	0
КРАІНА, М.	2023-12-21	2023-12-21	5	TEST760312		7	7400	21-12-23 11	5	0	0.00	0.0%-0%	Без ПДВ з	0.00	0.00	0.00	0.00	Карта	7400.00	0.00	212.1	212.1	0	0	7400
КРАІНА, М.	2023-12-21	2023-12-21	0	TEST760312		6	0		0	0	0.00	0.0%-0%	Без ПДВ з	0.00	0.00	0.00	0.00				212.1	212.1	0	0	0
КРАІНА, М.	2023-12-11	2023-12-12	2	TEST760312		5	17738	11-12-23 17	2	0	0.00	0.0%-0%	Без ПДВ з	0.00	0.00	0.00	0.00	Карта	17738.00	0.00	212.1	212.1	0	0	17738
КРАІНА, М.	2023-12-11	2023-12-11	2	TEST760312		4	33384.24	11-12-23 16	2	0	0.00	0.0%-0%	Без ПДВ з	0.00	0.00	0.00	0.00	Карта	33384.24	0.00	212.1	212.1	0	0	33384.24
КРАІНА, М.	2023-12-11	2023-12-11	35	TEST760312		3	422948.36	11-12-23 15	35	0	0.00	0.0%-0%	Без ПДВ з	0.00	0.00	0.00	0.00	Готівка / Наг	0.00 / 42294	0.00 / 0.00	169.68	212.1	42.42	0	422948.36
КРАІНА, М.	2023-12-07	2023-12-07	0	TEST760312		2	0	07-12-23 13	0	0	0.00	0.0%-0%	Без ПДВ з	0.00	0.00	0.00	0.00	Готівка	0.00	0.00	127.26	169.68	42.42	0	0

Рисунок 3.8 – Вигляд Z-звіту

В той час як в X-звіті кожен рядок це чек і вся інформація по ньому, то в Z-звіті кожен рядок це кожна зміна і в звіті подається загальна інформація по всім касам, враховуючи їхні чеки. Сам звіт має такі дані: адреса каси, дата відкриття/закриття зміни, загальна кількість чеків, фіскальний номер каси, номер змінної, виручка, дата і час останнього фіскального чеку, кількість чеків на покупку і повернення, сума ПДВ, сума податків. Всі ці дані обов'язково повинні бути присутніми, адже податковий орган контролює всі ці процеси.

3.2 Тестування та налагодження інформаційно-комп'ютерної системи

Спеціально для тестування функціоналу інтернет магазину було розроблено фабрики, які наповняють базу даних фейковими даними. При розгортанні проєкту буде можливість запустити фабрику і всі необхідні дані, такі як товари, замовлення, користувачі, блог наповняться випадковими даними і при необхідності фотографіями.

Є два види фабрик, основна і додаткова. Основна фабрика повинна обов'язково виконатись при розгортанні проєкту, вона додасть в базу записи про адміністратора і всі налаштування які зберігаються в базі даних. Додаткова фабрика запускається при необхідності. Це саме вона буде генерувати всі випадкові записи про замовлення і т.д. Додаткова фабрика розробляється спеціально для розробки і тестування веб-сайту, для того щоб вручну не наповнювати базу вигаданими даними.

Для тестування всього функціоналу фіскалізації і звітності сервіс Checkbox надає тестові API-ключі, тестову касу і касира. Проте весь процес ідентичний із реальними касами і касирами. Всі етапи відбуваються в точній послідовності як при реальній фіскалізації. На рисунку 3.9 зображено вигляд тестових даних які надає сервіс Checkbox.

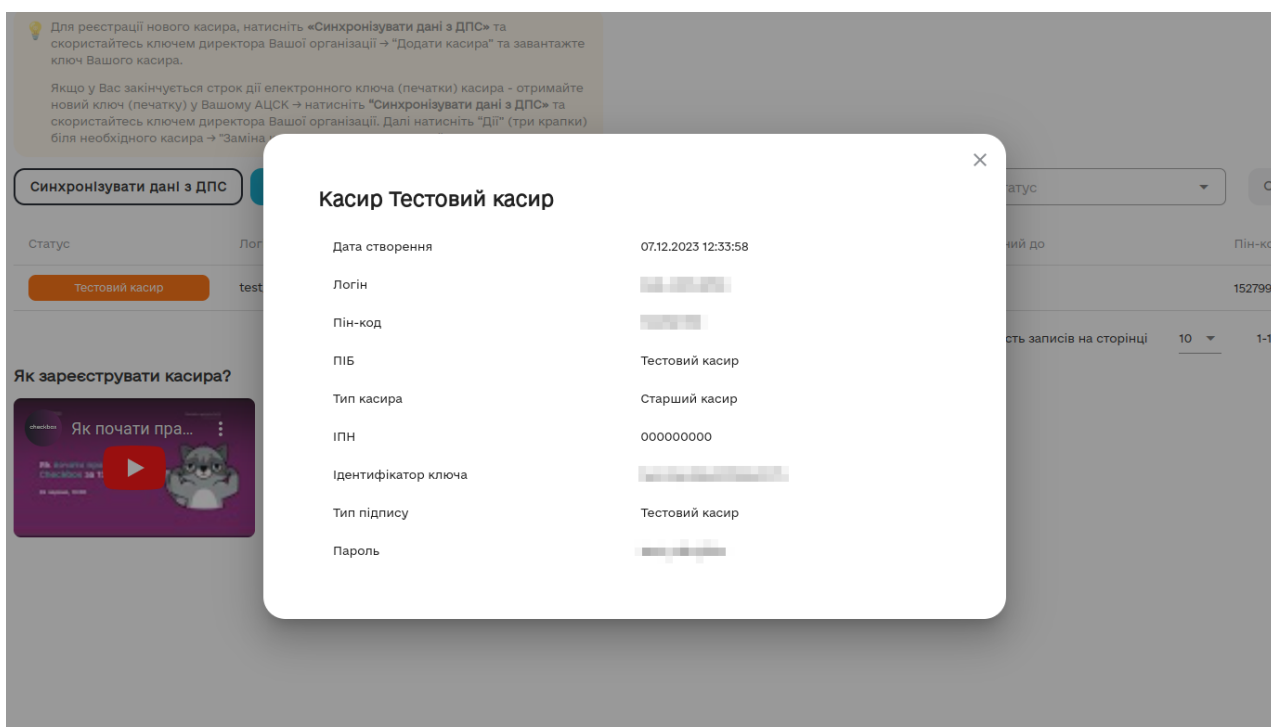


Рисунок 3.9 – Дані тестового касира надані сервісом Checkbox [9]

Головними даними для відкриття зміни і запуску роботи є ключ ліцензії каси та логін, пароль, пін-код касира. Ці дані є конфіденційними, тому нікому не можна їх передавати.

Все що потрібно для початку роботи в тестовому режимі це заповнити тестовими ключами в файлі «.env». Після цього все налаштовано і повинно працювати в штатному режимі. В першу чергу відкривається зміна і далі вже проводяться всі необхідні операції по фіскалізації чи звітності. При тому всі основні функції будуть працювати як під час реальної роботи.

Для тестових чеків також є налаштування їхнього зовнішнього вигляду і всіх інших налаштувань. Йому також можна налаштувати не тільки графічний

вигляд, а і вигляд чеку в текстовому файлі. Проте візуально він відрізняється від звичайного великим червоним попередженням що це тестовий чек і це попередження не можна ні зняти, ні поміняти. Цей текст зникне тільки вже для повноцінних фіскалізованих чеків. Зовнішній вигляд тестового чеку зображено на рисунку 3.10.

Тестовий чек

Тестова організація
М.КИЇВ ГОЛОСІВСЬКИЙ Р-Н, Тестова, 41а
ІД test_jEatU5J5SO

Штрих-код: 482655608940	1 x 1480.00	1480.00
Постільна білизна Gilette		

Банк	РВ
Продаж	
ЕПЗ	****4441
ПЛАТІЖНА СИСТЕМА	liqpay
Картка	1480.00 ГРН
Сума	1480.00 ГРН

Чек № TEST-zijlkj
22.12.2023 об 11:58:49
ОНЛАЙН

ФН ПРРО TEST760312
ТЕСТОВИЙ ЧЕК
checkbox



Тестовий чек

Рисунок 3.10 – Дані тестового касира надані сервісом Checkbox

Тестовий чек ні в якому разі не можна використовувати під час роботи, він не вважається фіскальним і тому його не можна відправляти покупцям. Тестовий чек не надходить до податкових органів і насправді не проходить фіскалізацію через них, проте якщо тестовий чек створюється і приходить, то і зі справжнім все буде працювати. Тестовий чек створений лише для того щоб

зробити відлагодження його зовнішнього вигляду і загальної роботи інтернет магазину для усіх етапів фіскалізації.

Тестові звіти також можна робити як X-звіти, так і Z-звіти. Вони формуються для тестових токенів і сформовані повністю ідентично, як приходили б справжні. На рисунку 3.11 зображено приклад як приходить X-звіт на електронну пошту у вигляді повідомлення із вкладеним файлом.

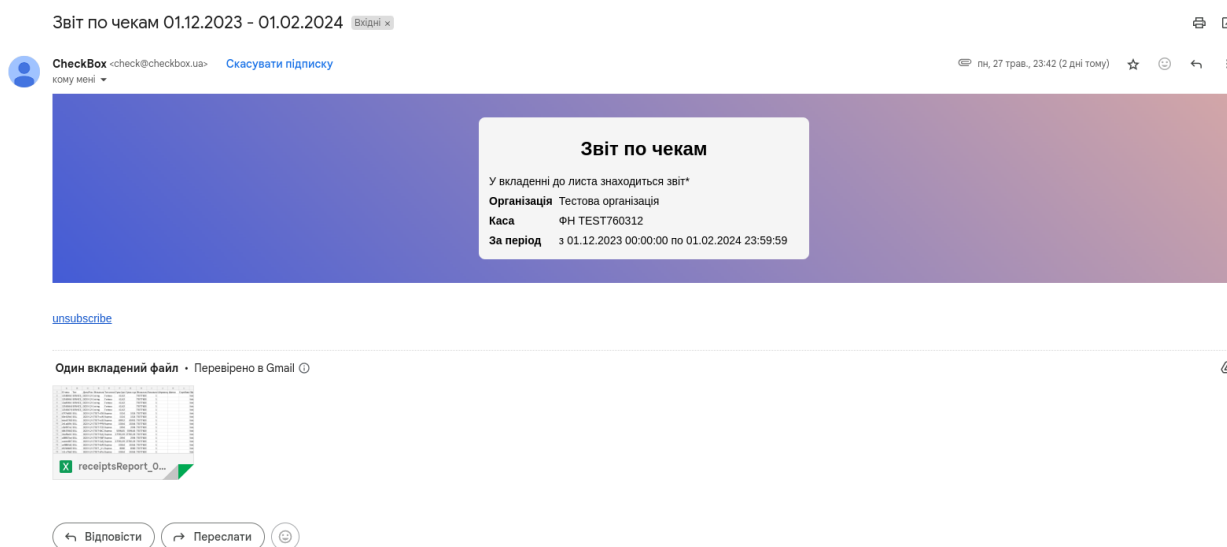


Рисунок 3.11 – Лист із X-звітом

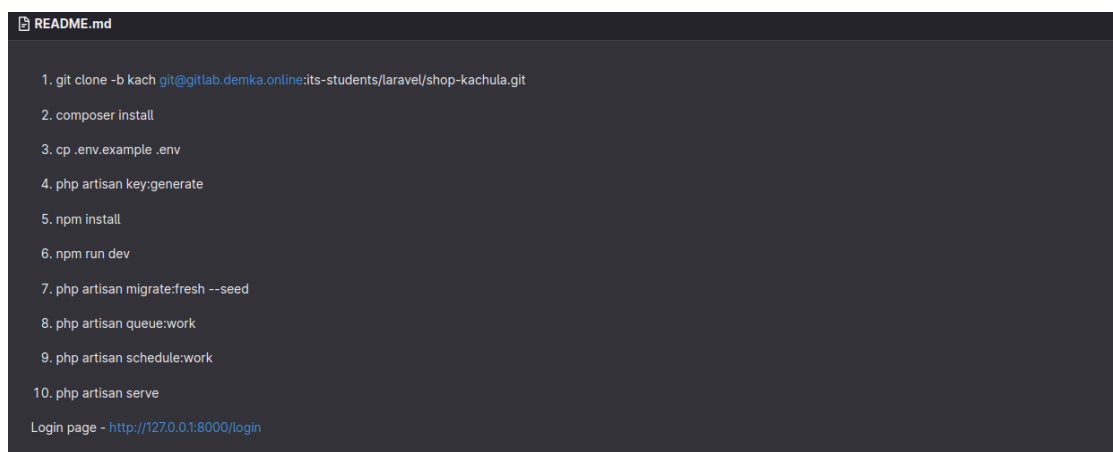
API сервісу Checkbox надає потужні можливості для тестування і налагодження процесу зсередини. Практично всі функції які є в реальній робочій касі, можна протестувати на тестовій. Для тестування навіть не потрібно підписувати договори, верифіковувати документи чи щось подібне, все що треба, це зареєструватися на сервісі і вписати тестові токени, які надаються одразу після реєстрації.

3.3 Розробка документації для програмного продукту

Для того щоб розгорнути проєкт у себе локально чи на сервері було спеціально описано документацію в файлі «README.md». В цьому файлі

описані всі кроки від клонування репозиторію до заповнення бази даних тестовими даними і повноцінного запуску на сервері.

В readme файлі знаходиться покрокова інструкція по розгортанню проєкту, це обов'язковий елемент кожного проєкту який потребує розгортання на сервері. На рисунку 3.12 зображено сам вигляд цього файлу.



```
1. git clone -b kach git@gitlab.demka.online:its-students/laravel/shop-kachula.git
2. composer install
3. cp .env.example .env
4. php artisan key:generate
5. npm install
6. npm run dev
7. php artisan migrate:fresh --seed
8. php artisan queue:work
9. php artisan schedule:work
10. php artisan serve

Login page - http://127.0.0.1:8000/login
```

Рисунок 3.12 – Інструкція в файлі «README.md»

Слідкуючи інструкції із файлу, в першу чергу потрібно клонувати репозиторій з серверу Git. Наступним кроком буде встановлення всіх пакетних залежностей Composer. Далі потрібно налаштувати «.env» файл і згенерувати командою ключ продукта який запишеться в цьому ж файлі. Наступні два кроки налаштовують і встановлюють залежності для фронтенд частини сайту. Коли репозиторій клоновано, всі залежності налаштовані, потрібно запустити міграції бази даних. Laravel з версії 10 може сам створювати не тільки таблиці, а і саму базу даних, тому необхідно лише запустити команду яка не тільки створить базу даних і її таблиці, а і наповнить їх тестовими даними через фабрики. Лишилися тільки налаштування сервера, а саме підключення черг та крон які будуть виконувати задачі в фоні від користувачів. Лишилось тільки запустити сервер.

Так як інтернет магазин написаний не лише на шаблонізаторах Blade, а і REST API, то для цього було описано повноцінну документацію із використанням пакету apiDoc [10].

ApiDoc дозволяє робити опис документації прямо в коді, що значно прискорює процес написання і розміщення певних частин документації, адже можна описувати документації безпосередньо біля методу який описується. Цей сервіс для розробки документації має свій синтаксис і його приклад зображено на лістингу 3.4, в якому описано метод отримання списку продуктів.

Лістинг 3.4 – Опис роуту apiDoc

```
/**
 * @api {get} /api/products Products
 * @apiName GetProducts
 * @apiGroup Catalog
 *
 * @apiParam {Integer} [page] Number Page
 * @apiParam {Integer} [per_page] Per Page
 * @apiParam {String="price","brand","name"} [sort] Sort Field
 * @apiParam {String="desc","asc"} [order] Sort Direction
 * @apiParam {String} [category] Category slug
 * @apiParam {String} [q] Search string
 *
 * @apiExample {curl} Example usage:
 *   /api/products?category=komplektuiushhie&order=desc&sort=name
 *
 */
```

Кінець лістингу 3.4

Документація описана на кожний API роут, адже це роути якими будуть користуватися інші розробники які будуть використовувати написане API. На кожен запит описано його тип, роут, приклад запиту і описані всі додаткові параметри, такі як фільтри.

Весь цей код компілюється однією командою в окремий файл. Доступ до цього файлу описується як через звичайний роут. На рисунку 3.13 зображено вже скомпільовану частину документації із лістингу 3.4.

GET			
/api/products			
Example usage:			
/api/products?category=komplektuushhie&order=desc&sort=name			
Parameter			
Field	Type	Description	
page	optional Integer	Number Page	
per_page	optional Integer	Per Page	
sort	optional String	Sort Field Allowed values: "price", "brand", "name"	
order	optional String	Sort Direction Allowed values: "desc", "asc"	
category	optional String	Category slug	
q	optional String	Search string	

Рисунок 3.13 – API документація роуту «/api/products»

Кожен додатковий параметр має опис свого типу даних, показник чи він обов’язковий та опис змінної. При необхідності вказуються дозволені значення та значення за замовчуванням.

3.4 SEO інформаційно-комп’ютерної системи

Яким би крутим функціоналом і гарним інтерфейсом не мав ваш веб-сайт, всього цього не достатньо для просування його в пошукових запитах. Пошукові системи, такі як Google Chrome, не дивляться наскільки круто виглядає інтернет магазин, вони розуміють тільки SEO-метатеги які створені спеціально для них та індексації карти сайту («Sitemap»).

Карта сайту створюється спеціально для індексації веб-застосунку в пошукових системах, для її просування в пошукових запитах, щоб пошуковий робот міг визначити сторінки сайту і їх зміст. Головні дані які вносяться в карту сайту це URL сторінки, дата останнього оновлення даних на цій сторінці та її пріоритетність серед інших сторінок. Є ще додаткові поля, такі як дата створення сторінки, але їх вже не обов’язково заповнювати.

В SEO описується головна інформація про сторінки веб-сайту, такі як назва сторінки, опис і різні налаштування (robots). Саме їх ми бачимо коли

шукаємо якийсь веб-сайт блукаючи в інтернеті. Для того щоб задати SEO для сторінок, в адміністративній частині, при створенні або редагуванні є форма де задаються метатеги. Форму з SEO можна побачити на рисунку 3.14.

The screenshot shows a web form titled 'SEO'. It has three main sections: 'Title' with a text input containing 'Про нас'; 'Description' with a text area containing a paragraph in Ukrainian about a company's mission; and 'Keywords' with a text input containing 'About, Page, Static'.

Рисунок 3.14 – Блок «SEO» на формі редагування сторінки

Дані з цієї форми беруться і підставляються в HTML розмітку сторінки у вигляді SEO-метатегів. Проте спеціально під месенджери звичайних метатегів не вистачає, для них потрібно виконувати спеціальну розмітку Open Graph. Це потрібна для коректного відображення зображення сторінки в месенджері та коректному наповненню текстом при надсиланні повідомлення із URL на наш веб-сайт.

На лістингу 3.5 можна побачити вже підставленні теги в розмітці HTML головної сторінки сайту.

Лістинг 3.5 – SEO-метатеги

```
<title>Головна</title>
<meta name="description" content="Ласкаво просимо на наш веб-сайт" />
<meta property="og:site_name" content="Laravel" />
<meta property="og:locale" content="uk" />
<meta property="og:title" content="Головна" />
<meta property="og:description" content="Ласкаво просимо на наш веб-сайт!" />
<meta property="og:image" content="http://dev.test/images/default.jpg" />
<meta property="og:url" content="http://dev.test" />
<meta property="twitter:title" content="Головна">
<meta property="twitter:description" content="Ласкаво просимо на наш веб-сайт! ">
```

Кінець лістингу 3.5

У наведеному вище лістингу можна побачити всі необхідні метатеги для веб-сайту. Всі вони відіграють свою роль при розробці SEO і в просуванні сайту в пошукових системах. Також для швидшого індексування пошуковою системою сайту, було створено карту сайту, приклад якої зображено на лістингу 3.6.

Лістинг 3.6 – Карта сайту

```
<url>
  <loc>http://dev.test/pro-nas</loc>
  <changefreq>monthly</changefreq>
  <priority>0.8</priority>
  <lastmod>2024-06-05 19:29:14</lastmod>
</url>
<url>
  <loc>http://dev.test/kontakty</loc>
  <changefreq>monthly</changefreq>
  <priority>0.8</priority>
  <lastmod>2024-05-26 18:29:12</lastmod>
</url>
```

Кінець лістингу 3.6

Таким чином розробивши систему налагодження SEO через адміністративну частину сайту і генерації карти сайту, було досягнуто більшої ефективності при пошуку його в інтернеті. Через він буде вище знаходитися в пошукових запитах, відповідно на нього будуть частіше заходити і робити більше замовлень.

Висновки до розділу 3

У висновку можна підбити підсумки що було розроблено та які функції реалізовано:

- фронтенд та бекенд інтернет магазину;
- користувацьке REST API;
- адміністративна частина сайту;

- можливість фіскалізувати зроблені замовлення та видавати фіскальні чеки;
- можливість автоматичної введення звітності в податкових органах, таких, формування X-звіту та Z-звіту;
- описана документація по розгортанню та використанню користувацької REST API.

Беручи до уваги виконане програмне забезпечення і його функціональність, можна сказати, що завдання до кваліфікаційної роботи було успішно виконано. Проєкт є повністю життєздатним, з добре описаною документацією і легким масштабуванням в майбутньому.

ВИСНОВКИ

Метою даної кваліфікаційної роботи була розробка веб-платформи, яка за допомогою сервісу Checkbox надавала можливість введення фіскального обліку та звітності і ефективно взаємодіяти із податковими органами для інтернет магазинів, які реалізують продаж онлайн.

Для досягнення мети було поставлено цілі, виконуючи які, поступово досягалась мета роботи. В першу чергу було виконано аналіз сучасного стану проблеми, та поставлено завдання до роботи. Вивчивши питання, було проаналізовано методи фіскалізації чеків в інтернет магазинах і дійшовши висновку що сервіс Checkbox підходить під поставленні задачі якнайкраще. Також було визначено вимоги до розроблюваного продукту, а саме програмне забезпечення повинне мати хорошу швидкодію, в онлайн форматі фіскалізувати всі грошові операції та мати можливість звітувати до податкових органів.

Після визначення всіх конструкційних питань, було складено схему бази даних, в якій було враховано ефективність системи загалом та легке впровадження фіскалізації чеків із збереженням їх у базі даних і майбутню масштабованість з хорошою підтримкою продукту протягом довгого періоду часу (рис. 3.1, аркуш 25). Наступним була сама реалізація програмного продукту, під час якого було розроблено інтернет магазин та її адміністративну частину. Вже потім відбулась реалізація функціоналу можливості видавати фіскалізовані чеки та надавати звітність (X-звіт, Z-звіт) в податкові органи. Останнім етапом було тестування та описання документації по розгортанню проєкту (readme файлу) і користуванню її REST API частиною (apiDoc).

Отже, якщо провести аналіз виконаної роботи, то можна дійти висновку що реалізацію всіх фіскальних операцій можна реалізувати в онлайн режимі використовуючи сервіс, який найбільше задовольняє потреби. Фіскальна звітність використовуючи онлайн методи, це те що спрощує життя і роботу підприємцям.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. E-commerce в Україні 2023: створення інтернет-магазину, законодавство, як підвищити довіру споживачів. URL: <https://www.site2b.ua/web-blog/e-commerce-v-ukraine-cifry-fakty-perspektivy-razvitiya-onlajn-torgovli.html> (дата звернення: 14.04.2024).
2. Exellio – Все про РРО та платіжні термінали. URL: <https://rro-info.com.ua/osoblyvosti-fiskalnogo-cheku> (дата звернення: 16.04.2024).
3. Development of world economy, marketing and management in modern conditions, 2023. 226 с. URL: https://isg-konf.com/wp-content/uploads/2023/11/ISG.2023.MONO_.ECON_.4.pdf (дата звернення: 16.04.2024).
4. Програмні РРО: обраний інструмент великими платниками податків у 2024 році. URL: <https://xn----7sbcickxnhd6alkoeinny1mti.xn--j1amh/uk/zrostayuchij-popit-na-programni-rro-vid-velikix-platnikiv-podatkov> (дата звернення: 19.04.2024).
5. Програмний РРО для будь-якого бізнесу в Україні – ПРРО Cashalot. URL: <https://cashalot.ua/> (дата звернення: 21.04.2024).
6. Програмний РРО «Вчасно.Каса». URL: <https://kasa.vchasno.com.ua> (дата звернення: 21.04.2024).
7. Кращий програмний РРО (пРРО) для бізнесу в Україні – Checkbox. URL: <https://checkbox.ua> (дата звернення: 21.04.2024).
8. Шепель І. В. Ведення бухгалтерського обліку та дотримання розрахунково-касової дисципліни торговельних операцій. URL: <https://repo.btu.kharkov.ua/bitstream/123456789/44269/1/tezy-dop-conf-02-11-23-86-87.pdf> (дата звернення: 22.04.2024).
9. Документація. Налаштування друку чеків | Checkbox. URL: <https://wiki.checkbox.ua/uk/portal/settings/receipts> (дата звернення: 04.05.2024).
10. apiDoc – вбудована документація для RESTful web API. URL: <https://apidocjs.com> (дата звернення: 05.05.2024).

ДОДАТКИ

Додаток А

Реалізація функціоналу створення різних типів звітів

```
<?php

namespace App\Actions\Checkbox;

use Illuminate\Support\Arr;
use Illuminate\Support\Facades\Log;
use Lorisleiva\Actions\Concerns\AsAction;

class CreateReportAction extends BaseAction
{
    use AsAction;

    public function handle(string $type, array $params = null)
    {
        $api = $this->initApi();

        $cashier = $api->getCurrentCashierShift();

        if (Arr::get($cashier, 'status') !== 'OPENED') {
            Log::warning('CHECKBOX: Зміну не відкрито');
            return false;
        }

        switch ($type) {
            case 'x':
                $res = $api->createReportX();
                break;
            case 'z':
                if (is_array($params)) {
                    $res = $api->createReportZ($params);
                } else {
```

```
Log::warning('CHECKBOX: Немає обов\'язкових  
параметрів');  
  
        return false;  
    }  
  
    break;  
default:  
    Log::warning('CHECKBOX: Невідомий тип звіту');  
  
    return false;  
}  
  
return $res;  
}  
}
```