

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Сумський державний університет
Факультет електроніки та інформаційних технологій
Кафедра інформаційних технологій

«До захисту допущено»

В.о. завідувача кафедри

_____ Світлана ВАЩЕНКО

_____ 2023 р.

КВАЛІФІКАЦІЙНА РОБОТА

на здобуття освітнього ступеня магістр

зі спеціальності 122 «Комп'ютерні науки» _____,

освітньо-професійної програми «Інформаційні технології проектування»

на тему: «Web-орієнтована система підтримки ведення податкового обліку фізичної особи підприємця»

Здобувача групи ІТ.м-22 Харченка Богдана Володимировича

(шифр групи)

(прізвище, ім'я, по батькові)

Кваліфікаційна робота містить результати власних досліджень. Використання ідей, результатів і текстів інших авторів мають посилання на відповідне джерело.

(підпис)

Богдан ХАРЧЕНКО

(Ім'я та ПРІЗВИЩЕ здобувача)

Керівник к. т. н. доц. Ващенко С.М.

(посада, науковий ступінь, вчене звання, ім'я та ПРІЗВИЩЕ)

_____ (підпис)

Суми – 2023

Сумський державний університет
Факультет електроніки та інформаційних технологій
Кафедра інформаційних технологій
Спеціальність 122 «Комп'ютерні науки»
Освітньо-професійна програма «Інформаційні технології проектування»

ЗАТВЕРДЖУЮ

В.о. завідувача кафедри ІТ

Світлана ВАЩЕНКО

«_____» _____ 2023 р.

ЗАВДАННЯ
НА КВАЛІФІКАЦІЙНУ РОБОТУ СТУПЕНЯ МАГІСТР СТУДЕНТУ

Харченко Богдан Володимирович

1 Тема роботи Web-орієнтована система підтримки ведення податкового обліку фізичної особи підприємця_____

затверджена наказом по університету від «08» листопада 2023 р. № 1249-VI

2 Термін здачі студентом кваліфікаційної роботи « 15 » грудня 2023 р.

3 Вхідні дані до кваліфікаційної роботи технічне завдання на розробку Web-орієнтованої системи підтримки ведення податкового обліку фізичної особи підприємця_____

4 Зміст розрахунково-пояснювальної записки (перелік питань, що їх належить розробити) аналіз предметної області, проектування web-орієнтованої системи, розробка web-орієнтованої системи_____

5 Перелік графічного матеріалу (з точним зазначенням обов'язкових креслень) Актуальність роботи. Постановка задачі. Аналіз предметної області. Структурно-функціональна модель. Діаграма варіантів використання. Схема бази даних. Демонстрація роботи. Висновки_____

6. Консультанти розділів роботи:

Розділ	Консультант	Підпис, дата	
		Завдання видав	Завдання прийняв

Дата видачі завдання _____.

Керівник _____
(підпис)

Завдання прийняв до виконання _____
(підпис)

КАЛЕНДАРНИЙ ПЛАН

№ п/п	Назва етапів кваліфікаційної роботи	Строк виконання етапів роботи	Примітка
1	Дослідження предметної області	04.10.2022 – 10.10.2023	Викон.
2	Розробка шаблону web-орієнтованої системи	11.10.2022 – 14.10.2023	Викон.
3	Проектування сторінок системи	15.10.2023 – 22.10.2023	Викон.
4	Розробка основних функціональних модулів	23.10.2022 – 14.11.2023	Викон.
5	Розробка бази даних	15.11.2023 – 18.11.2023	Викон.
6	Підключення системи до бази даних	19.11.2023 – 24.11.2023	Викон.
7	Тестування	25.11.2023 – 28.11.2023	Викон.
8	Оформлення пояснювальної записки	29.11.2023 – 05.12.2023	Викон.
9	Підготовка демонстраційного матеріалу та доповіді	06.12.2023 – 17.12.2023	Викон.
10	Захист	18.12.2023 – 25.12.2023	Викон.

Магістрант _____
(підпис)

Харченко Б.В.

Керівник роботи _____
(підпис)

к.т.н., доц. Ващенко С.М.

АНОТАЦІЯ

Тема кваліфікаційної роботи магістра «Web-орієнтована система підтримки ведення податкового обліку фізичної особи підприємця».

Пояснювальна записка складається зі вступу, 4 розділів, висновків, списку використаних джерел із 40 найменувань, додатків. Загальний обсяг роботи – 99 сторінок, у тому числі 54 сторінки основного тексту, 4 сторінки списку використаних джерел, 45 сторінок додатків.

Актуальність роботи полягає в постійних змінах у сфері оподаткування України та нагальній потребі у вдосконаленні сервісів підтримки податкового обліку для фізичних осіб підприємців (ФОП). Зростання кількості підприємців у сучасному суспільстві потребує створення ефективних та простих у користуванні інструментів для фінансового управління у відповідності до законодавства.

Web-орієнтована система підтримки ведення податкового обліку стане затребуваною в умовах розвитку податкової системи та динамічних змін у законодавстві. Завдяки ній ФОП зможуть забезпечити своєчасність та точність ведення обліку фінансових операцій, уникнути податкових ризиків та забезпечити відповідність їхньої діяльності податковим нормам, а також система спростить чи навіть автоматизує процеси, такі як аналітика прибутку та формування звітності.

Мета роботи: розробка web-орієнтованої системи підтримки ведення податкового обліку фізичної особи підприємця для спрощення ознайомлення з державною податковою системою та оптимізації ведення звітності.

Ключові слова: WEB-ОРІЄНТОВАНА СИСТЕМА, ПОДАТКИ, КАЛЬКУЛЯТОР, ОБЛІК, ФІНАНСИ, HTML, CSS, JAVASCRIPT, JS, PHP, СУБД, MYSQL.

ЗМІСТ

ВСТУП.....	6
1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ.....	8
1.1 Огляд предметної області.....	8
1.2 Огляд аналогів.....	10
2 ПОСТАНОВКА ЗАДАЧІ.....	14
2.1 Постановка задачі.....	14
3 МОДЕЛЮВАННЯ ТА ПРОЕКТУВАННЯ.....	19
3.1 Структурно-функціональне моделювання.....	19
3.2 Моделювання варіантів використання.....	21
3.3 Модель схеми даних.....	23
4 ПРОГРАМНА РЕАЛІЗАЦІЯ.....	25
4.1 Архітектура web-системи.....	25
4.2 Розробка дизайну web-додатку.....	26
4.3 Реалізація web-додатку.....	28
4.4 Тестування програмного продукту.....	35
4.5 Демонстрація роботи web-додатку.....	39
ВИСНОВКИ.....	50
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ.....	51
ДОДАТОК А.....	55
ДОДАТОК Б.....	66

ВСТУП

Сучасна Україна переживає важкий період: війна, реформи, вибір шляху розвитку та його відстоювання. У такій ситуації сфера оподаткування є однією з найбільш важливих та навантажених. Впровадження реформ та нестабільні умови оподаткування, що постійно змінюються, адаптуючись до реалій війни, породжують потребу в ефективних інструментах підтримки фізичних осіб підприємців у веденні податкового обліку їх діяльності.

В контексті вищезазначених проблем саме інформаційні технології (ІТ) набувають особливої важливості. За останні десять років вплив ІТ на суспільство став настільки суттєвим, що його неможливо ігнорувати. Свідченням цього є факт, що Україна стала не тільки першою країною, яка впровадила електронний паспорт на державному рівні, а й першою країною, що зрівняла чинність електронних документів та фізичних, та зібрала їх в одному мобільному застосунку – Дії. Саме через це велика кількість європейських країн бажає та вже співпрацює з розробниками з України над реалізацією аналогів Дії для внутрішнього користування.

Однією з найважливіших галузей ІТ є web-орієнтовані системи, які характеризуються інтуїтивною зрозумілістю користування і легким доступом, що робить їх універсальними та популярними серед користувачів. Враховуючи ці характеристики, веб-орієнтований додаток було обрано як об'єкт розробки кваліфікаційної роботи. Предметом дослідження є ведення податкового обліку та звітності фізичними особами підприємцями.

Об'єкт дослідження: ведення податкового обліку та звітності фізичними особами підприємцями.

Предмет дослідження: web-орієнтовані додатки підтримки процесу ведення податкового обліку фізичними особами підприємцями.

Метою роботи є розробка web-орієнтованої системи підтримки ведення податкового обліку фізичної особи підприємця, спрямованої на спрощення ознайомлення з державною податковою системою та оптимізацію ведення звітності.

Для досягнення поставленої мети необхідно:

- ознайомитися з нормативною базою діяльності фізичних осіб-підприємців (ФОП);
- проаналізувати наявні аналоги;
- розробити шаблон та модулі web-орієнтованої системи;
- створити базу даних;
- реалізувати розроблюваний продукт;
- провести тестування;
- розмістити web-системи на хостингу.

Практичне значення роботи полягає в заощадженні фінансових та часових витрат підприємців на веденні податкової діяльності за рахунок розроблюваної системи.

1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ

1.1 Огляд предметної області

ІТ в Україні щільно впроваджені в усі сфери державної діяльності. Незважаючи на це податкова система України, є обширною та складною сферою [1]. Хоча в повсякденному житті більшість громадян не помічають цю складність через те, що не приймають безпосередньої участі в податкових процесах, бо за них це роблять бухгалтери, та в момент зустрічі з ними «наодинці», починається нерозуміння [2]. Для багатьох прикладом початку такої «зустрічі» є рішення про ведення підприємницької діяльності як фізичною особою-підприємцем (ФОП) [3].

Для ефективного управління звітністю необхідно повністю розуміти всі аспекти податкового процесу, такі як відсоткові ставки, строки подачі та форми звітності [4]. Без застосування інформаційних технологій (ІТ) це вимагало б великої кількості часу, саме тому є такі спеціалісти, як бухгалтери. Зараз, завдяки відповідним ІТ, які інтегровані у податкову систему України, кожен громадянин може легко отримати необхідну інформацію на web-ресурсах, зекономивши при цьому багато часу та грошей [5]. Саме з другої причини багато ФОП вирішують вести свою фінансову та податкову діяльність самотужки, занурюючись в вивчення цих сфер діяльності. Проте, для повного розуміння все ще необхідно витратити значний час на пошук інформації, що часто призводить до проблем, оскільки інформація на різних web-ресурсах може відрізнятись або бути застарілою, а функціональний калькулятор податків взагалі складно знайти.

Інформаційні технології проявляють найбільшу ефективність саме в фінансовій сфері [6], в тому числі й податковій системі, де для розрахунків використовується велика кількість математичних формул. Саме завдяки ІТ стало можливим проводити розрахунок податків та ведення документообігу

онлайн. Це реалізовано на державному ресурсі – Електронному кабінеті платника податків [7]. У цьому контексті калькулятор податків значно спрощує процес розрахунків, з якими раніше громадянин не мав справи, і дозволяє перевіряти коректність ведення податкового обліку [8]. Однак такий калькулятор податків інтегровано лише у бланки звітності, і без їх заповнення отримати до нього доступ неможливо. Такий підхід є скоріш альтернативним рішенням, ніж єдиним, бо створюючи велику кількість таких бланків, виконуючи розрахунки, дуже легко заплутатись, та відправити не той звіт.

З мого досвіду можу стверджувати, що більшість осіб підприємців користуються послугами бухгалтерів та фінансових консультантів для розрахунку податків та ведення податкового обліку через сприйняття податкових процесів як надскладних [9]. І в цьому інформаційні технології приходять на допомогу найкращим чином. Розробка web-орієнтованої системи підтримки ведення податкового обліку фізичної особи підприємця надасть повноцінний функціонал для самостійного контролювання своєї фінансової діяльності, ведення податкового обліку та інформування актуальними змінами в сфері за допомогою зручного інструментарію.

Основним юридичним джерелом, що містить законодавство з податкової системи, є Податковий кодекс України [10]. Усі податкові процеси базуються на цьому документі, визначаючи порядок оподаткування, розміри податків та категорії платників.

У розроблюваній системі розглядатимуться: податки на прибуток чотирьох груп фізичних осіб підприємців; ведення фінансового обліку діяльності, автоматизована аналітика діяльності підприємця та генерація бланків податкової звітності.

1.2 Огляд аналогів

Останнім часом набувають популярності бухгалтерські послуги з консультуванням щодо оподаткування та розрахунку податків. Декілька прикладів (рис.1.1 – 1.4): Taxer [11] та Talent.com [12]. Ці сервіси поєднані єдиною предметною областю, але мають різний функціонал та методи надання послуг.

Taxer (рис. 1.1 – 1.3) останнім часом є одним з найпопулярніших web-орієнтованих сервісів, що надає функціонал для генерації податкових бланків звітності. Корисним та функціональним є розділ «Календар», в якому користувачі можуть переглянути строки подачі звітів та оплати податків. Окрім цього сервіс має зручну базу знань, де зібрана інформація щодо ведення ФОП. Для отримання повноцінного функціоналу необхідно придбати підписку.

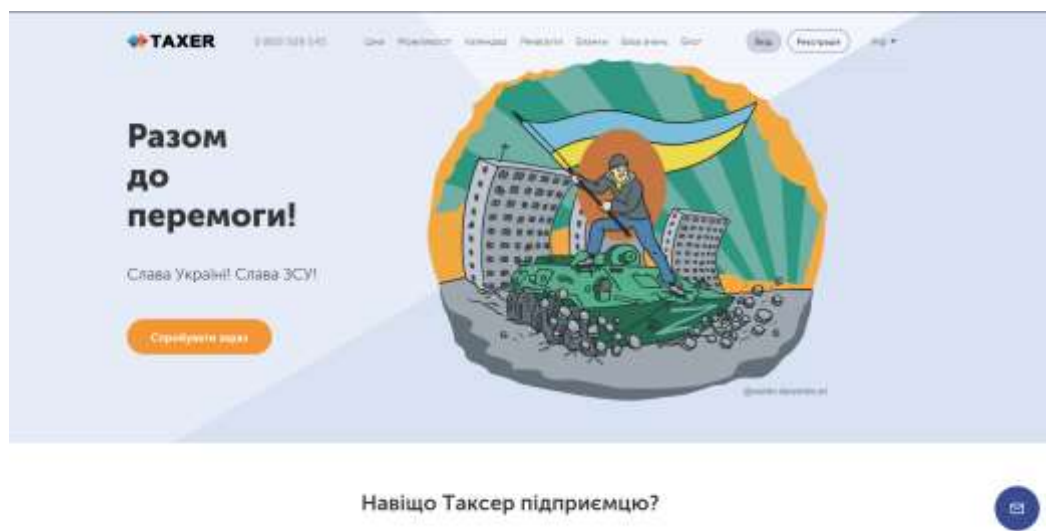


Рисунок 1.1 - Сервіс Taxer: головна сторінка

Джерело:[11]

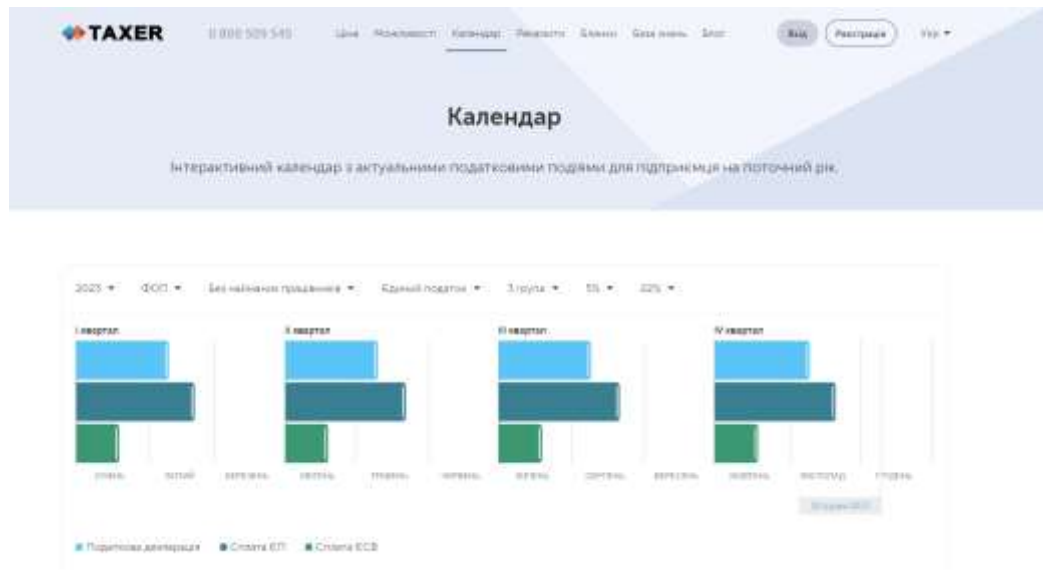


Рисунок 1.2 - Сервіс Taxer: податковий календар

Джерело:[11]

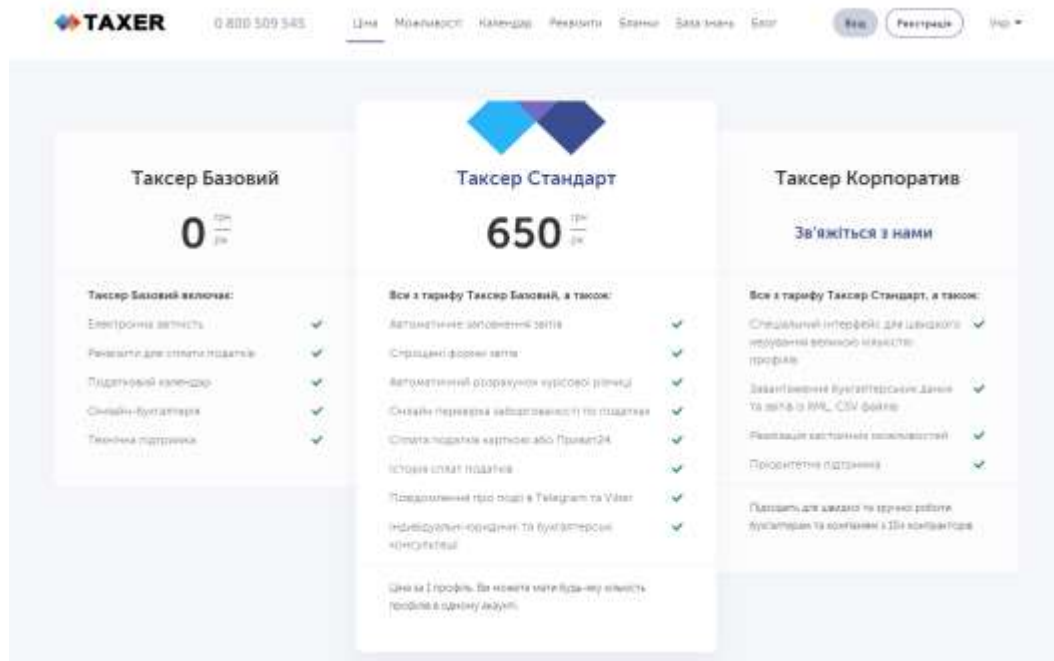


Рисунок 1.3 – Сервіс Taxer: вартість користування

Джерело:[11]

Talent.com [11] – web-сервіс для пошуку роботи. Додаток має калькулятор для розрахунку податків на прибуток найманого робітника але не має калькулятора для розрахунку податків підприємців. Також на сторінці калькулятора міститься інформація щодо податків, що сплачують як

робітники, так і роботодавці, але лише для податку найманого робітника. Будь-яка інформація стосовно ФОП відсутня на сервісі й калькулятор має лише одну функцію, не має гнучких налаштувань. Також сервіс не має модулю формування податкової звітності чи ведення фінансового обліку.

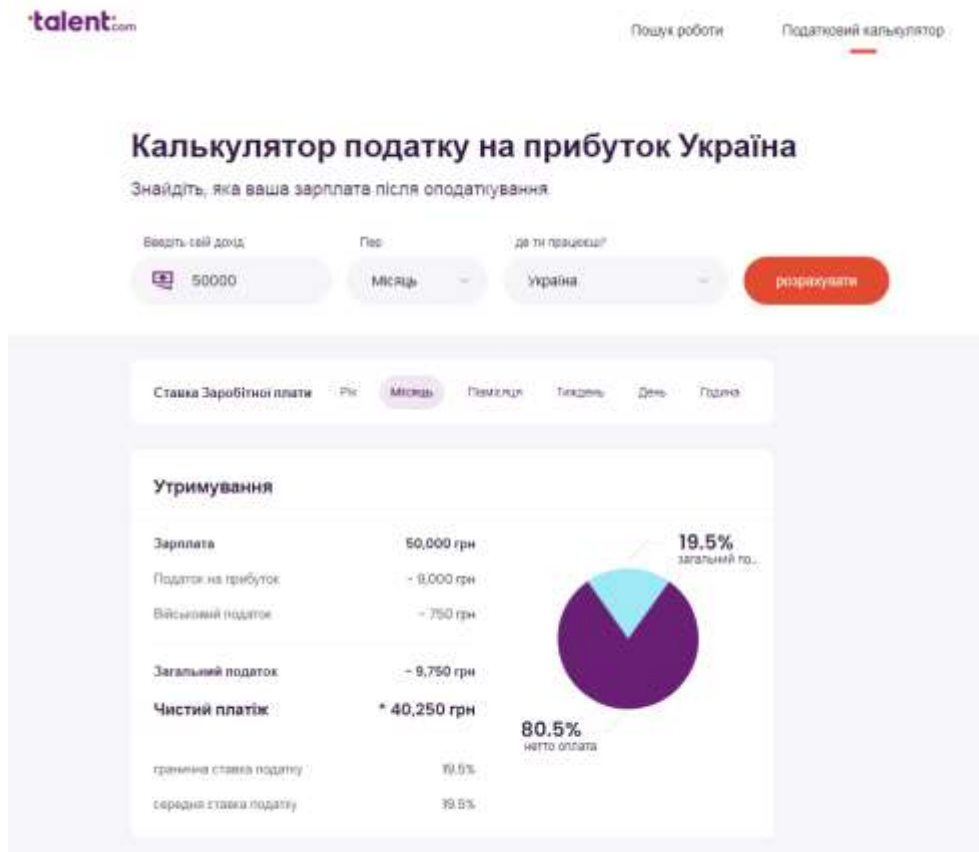


Рисунок 1.4 – Сервіс talent.com

Джерело: [12]

Детальний аналіз аналогів представлений у таблиці 1.1.

Таблиця 1.1 – Порівняння показників аналогів розроблюваного продукту

Характеристики/Продукт	Taxer	Talent.com
Наявність калькулятора податків	-	+
Наявність новин податкової сфери	+	-
Наявність можливості обирання виду податку в калькуляторі	-	+
Наявність посилань на першоджерела та корисні джерела	-	-

Продовження табл. 1.1.

Наявність детального опису податків за видами	+	+
Можливість ведення фінансового обліку	-	-
Можливість генерації звітності	+	-

Джерело: власна реалізація

За результатами проведеного аналізу можна зробити висновки, що з двох жоден з конкурентів не задовольняє потреби фізичних осіб-підприємців в повній мірі, тому необхідно створити нову систему, яка забезпечить усі необхідні характеристики наведені у таблиці 1.1.

Отже, можна зробити висновок, що розробка web- орієнтованої системи підтримки ведення податкового обліку фізичної особи підприємця має сенс, а це означає, що обрана тема є актуальною.

2 ПОСТАНОВКА ЗАДАЧІ

2.1 Постановка задачі

Головна мета проекту полягає в розробці web-орієнтованої системи підтримки ведення податкового обліку фізичної особи підприємця. Використання такої системи стане запорукою заощадження фінансових та часових витрат підприємців на веденні податкової діяльності.

Розроблюваний продукт орієнтовано на фізичних осіб, що планують, або вже займаються підприємницькою діяльністю в якості фізичної особи-підприємця (ФОП) та мають намір вести фінансову діяльність самостійно, без залучення профільного спеціаліста.

Основні вимоги до web-додатку:

- інтуїтивно зрозумілий інтерфейс [13];
- наявність калькулятора податків для всіх груп ФОП;
- бути простим в управлінні для підтримки та оновлення;
- наявність можливості генерації податкової звітності;
- наявність можливості ведення фінансового обліку та аналізу діяльності.

Етапи виконання проекту такі:

- розробка шаблону та модулів web-орієнтованої системи;
- створення бази даних;
- реалізація розроблюваного продукту;
- проведення тестування;
- розміщення web-системи на хостингу.

Структура сайту має містити наступні розділи:

- профіль – має містити підрозділи:
 - аналітика – містить модуль ведення фінансової діяльності (облік прибутку), а також аналітику щодо наданих даних;

- звітність – містить модуль генерації, зберігання та завантажування бланків податкової звітності;
- калькулятор податків – має містити підрозділи:
 - калькулятор податків ФОП – містить калькулятори для розрахунку податку фізичних осіб-підприємців;
 - калькулятор податків найманого робітника – містить калькулятори для розрахунку податку найманого робітника;
 - калькулятор податків купівлі-продажу – містить калькулятори для розрахунку податку процесу купівлі-продажу;
 - калькулятор податків спадщини – містить калькулятори для розрахунку податку на спадщину;
- новини – містить останні новини податкової сфери України;
- про податки – містить фундаментальні дані щодо податків: їх типи, правила нарахування та розрахунку;
- корисні джерела – містить корисні джерела щодо ведення податкової діяльності, приклад основні державні web-ресурси такі як електронний кабінет платника податків [7], державний реєстр платників податків [3] тощо;

В графічному вигляді структура сайту зображена на рисунку 2.1.

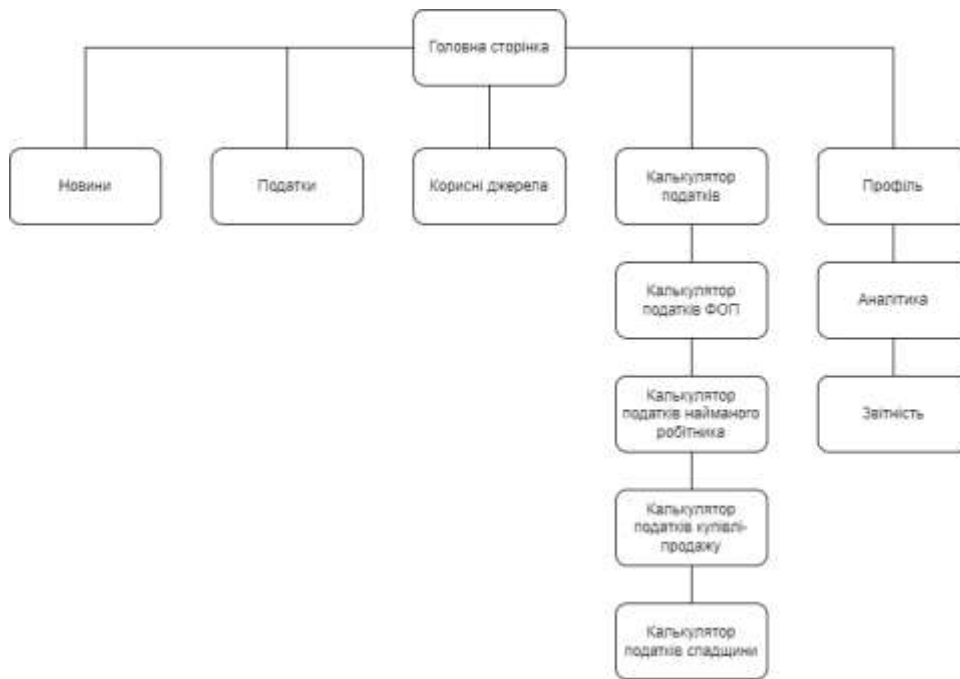


Рисунок 2.1 – Карта розроблюваного web-додатку

Джерело: власна розробка.

У додатку мають бути реалізовані наступні категорії користувачів:

- адміністратор – має можливість на перегляд, додавання, редагування та видалення інформації, створення звітів, ведення особистого фінансового обліку, розрахунок податків, заповнення розділу «Профіль»;
- зареєстрований користувач – має можливість на перегляд інформації, створення звітів, ведення особистого фінансового обліку, розрахунок податків, заповнення розділу «Профіль»;
- гість – має можливість на перегляд інформації, розрахунок податків;

Планування проведення робіт надано у додатку А.

2.2 Засоби реалізації

Засобами реалізації проекту було обрано:

- HTML – мову гіпертекстової розмітки сторінки для створення каркасу web-додатку [14]. Її переваги:
 - універсальність: підтримується всіма web-браузерами;
 - легкість використання: має простий синтаксис;
- CSS – мова стилізації для надання каркасу приємного дизайну [15]. Її переваги:
 - підтримка адаптивного дизайну;
- JavaScript – високорівнева мова програмування для створення інтерактивних функцій додатку [16]. Її переваги:
 - інтерактивність: можливість надавати динамічності та інтерактивності web-контенту;
 - широке застосування та універсальність;
- PHP – серверна мова програмування для зв'язку web-додатку з сервером та роботи з базами даних [17]. Її переваги:
 - гнучкість та інтегрованість: легко інтегрується з базами даних та іншими технологіями;
- MySQL – програмне забезпечення для управління реляційними базами даних [18]. Її переваги:
 - надійність та швидкість: висока продуктивність, надійність та легкість управління;
 - переносимість та сумісність: підтримується більшістю операційних систем та інтегрується з різними мовами програмування.

2.3 Розрахунок податків

Розрахунок податків відбувається у модулі «Калькулятор податків». Усі розрахунки спираються на державні нормативні акти щодо податкової системи України [19]. Формула розрахунку податку для груп фізичної особи-підприємця:

$$П = \frac{С}{100\%} \times Пс + ЄСВ, \quad (2.1)$$

де П – податок,

Пс – податкова ставка групи,

ЄСВ – податок єдиний соціальний внесок, котрий складає 22% від встановленої мінімальної заробітної плати в країні [20].

Податкова ставка групи визначається згідно норм, зазначених в пунктах 291-293 Податкового кодексу України (ПКУ) [10]:

- група 1 – не більше 10% від прожиткового мінімуму для працездатних осіб;
- група 2 – не більше 20% від мінімальної заробітної плати;
- група 3 – 3% для платників ПДВ та 5% для тих, хто не сплачує ПДВ, з можливістю зменшення до 2% під час воєнного стану;
- група 4 – ставки податку залежать від типу та розташування земель, коливаючись від 0.16% до 6.33%.

3 МОДЕЛЮВАННЯ ТА ПРОЕКТУВАННЯ

3.1 Структурно-функціональне моделювання

Процес моделювання складається зі створення абстрактної концептуальної схеми, логічної та фізичної моделей. Етапи є взаємопов'язаними та доповнюють один одного.

Для створення концептуальної схеми майбутнього web-додатку використовуватиметься методологія IDEF0, котра використовує графічний підхід та описує системи й процеси діяльності як комплекс взаємозалежних функцій [21]. Такий підхід надає перевагу у дослідженні функції проекту в незалежності від конкретних об'єктів, які виконують ці функції.

Стандарт IDEF0 відображає об'єкти системи, матеріальні й інформаційні потоки, які перетворюються в бізнес-процесі.

Для контекстної діаграми було визначено наступні дані:

- вхідні дані – запит користувача на розрахунок суми податку;
- управління – чинні закони державної податкової системи;
- механізми – web-система, користувач-підприємець, технічне забезпечення;
- вихідні дані – результати розрахунку податку.

Функціональне моделювання діяльності web-орієнтованої системи підтримки ведення податкового обліку фізичної особи підприємця в IDEF0 можна оглянути на рисунку 3.1.



Рисунок 3.1 – Концептуальна діаграма

Джерело: власна реалізація.

Також було проведено декомпозицію IDEF0 для деталізації внутрішніх процесів (рисунок 3.2) [22].

Декомпозиція діаграми розрахунку податку представлена такими підпроцесами:

- вибір групи ведення ФОП;
- заповнення полів форми калькулятора податку;
- розрахунок податку.

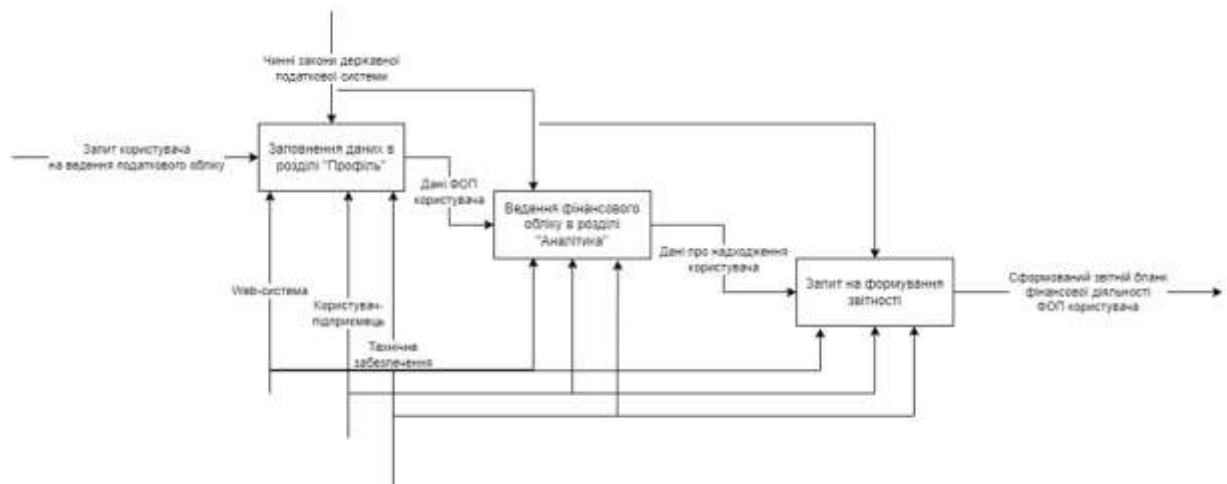


Рисунок 3.2 – Діаграма декомпозиції

Джерело: власна реалізація.

3.2 Моделювання варіантів використання

Після детального моделювання процесів та аналізу структури розроблюваної web-орієнтованої системи виникла необхідність у створенні діаграми варіантів використання.

Use Case діаграма являє собою схематичне зображення варіантів використання системи, описує її функціональне призначення та взаємодію акторів з нею [23]. Основними складовими діаграми є актори та їх взаємодія з системою.

Така схема варіантів використання надає можливість уявити майбутню систему як сукупність об'єктів та суб'єктів, які певним чином взаємодіють з нею. В цьому контексті взаємодія характеризується відповіддю системи на запит актора. Таким чином, варіанти використання системи можна розглядати як набір дій, необхідних для взаємодії з нею.

Для web-орієнтованої системи підтримки ведення податкового обліку фізичної особи підприємця було виділено такі варіанти використання для наступних акторів:

- адміністратор:
 - додавання інформації;
 - редагування інформації;
 - видалення інформації;
- користувача:
 - реєстрація;
 - перегляд інформації у web-системі;
 - розрахунок податків;
 - формування звітності;
 - збереження даних фінансової діяльності.

Діаграма варіантів використання web-орієнтованої системи підтримки ведення податкового обліку представлена на рисунку 3.3.

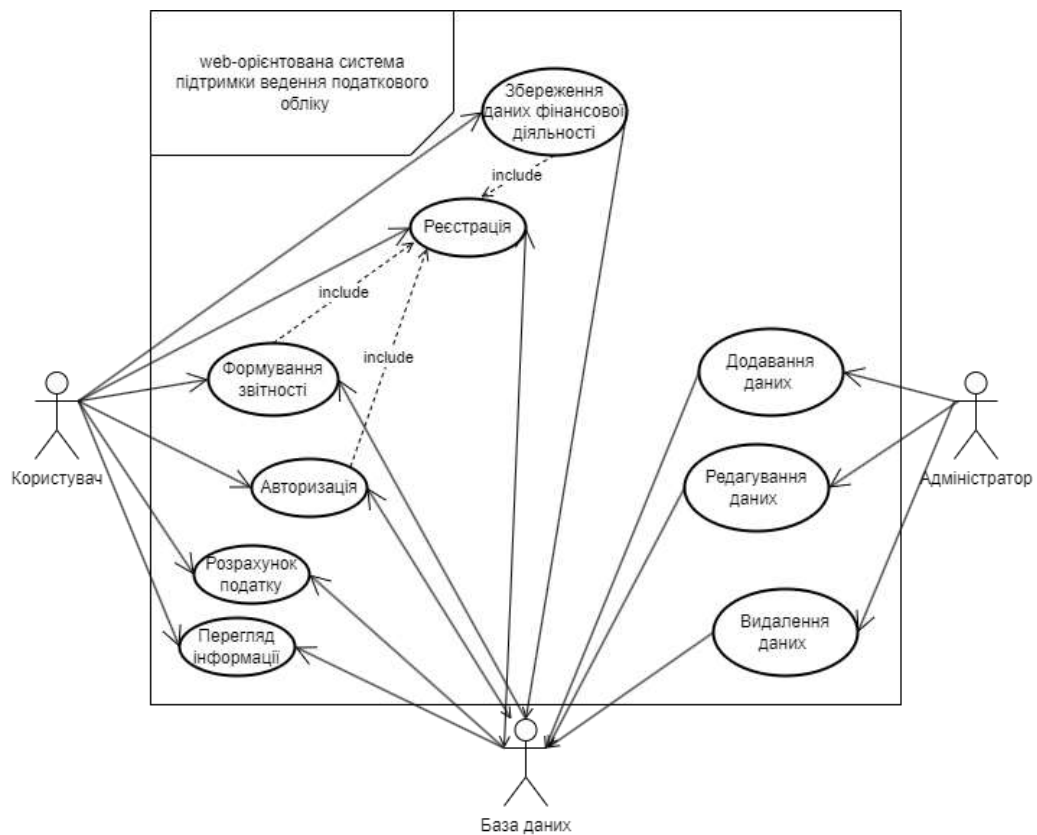


Рисунок 3.3 – Діаграма варіантів використання

Джерело: власна реалізація.

3.3 Модель даних

Після проведення моделювання розроблюваної системи необхідно провести моделювання та розробку сховища для зберігання необхідних даних. В якості сховища даних було обрано використати базу даних (БД).

Початковим етапом розробки інформаційної бази є проектування. Під час проектування створюється чітка структура зберігання даних, яка буде представлена у вигляді таблиць. Структуру таблиць визначають такі параметри, як розмір, тип даних, первинні, вторинні та унікальних ключі [24].

Для створення інформаційної бази важливо визначити сутності – об'єкти предметної області розроблюваного додатку, які необхідно зберігати. Кожна сутність буде відображена окремою таблицею в базі даних. У стовпцях таблиць розміщені атрибути, а в рядках – конкретні екземпляри.

У процесі проектування були виділені наступні сутності:

- новини (news);
- податки – таблиця з їх описом (tax_description);
- податки – таблиця з їх розмірами для розрахунків (tax);
- користувачі – таблиця для зберігання даних зареєстрованих користувачів (users);
- корисні джерела – таблиця переліком корисних джерел (usefull_source);
- профіль клієнта – таблиця з даними з розділу «профіль» (users_profile);
- звіти – таблиця з даними щодо створених звітів (reports);
- надходження – таблиця з даними щодо обраної користувачами валюти в розділі ведення фінансової діяльності (Income);
- щомісячні надходження – таблиця з даними щодо місячного прибутку користувачів з розділу ведення фінансової діяльності (IncomeMonth);

– технічна таблиця – використовується для контролю доступу до платформи під акаунтом користувача (а).

Для оперативної взаємодії з базою даних використовується система управління базою даних MySQL, що дозволяє зручно додавати, редагувати та видаляти дані [25]

Модель виконаної фізичної реалізації бази даних для забезпечення працездатності web-орієнтованої системи підтримки ведення податкового обліку фізичної особи підприємця представлена на рисунку 3.4.

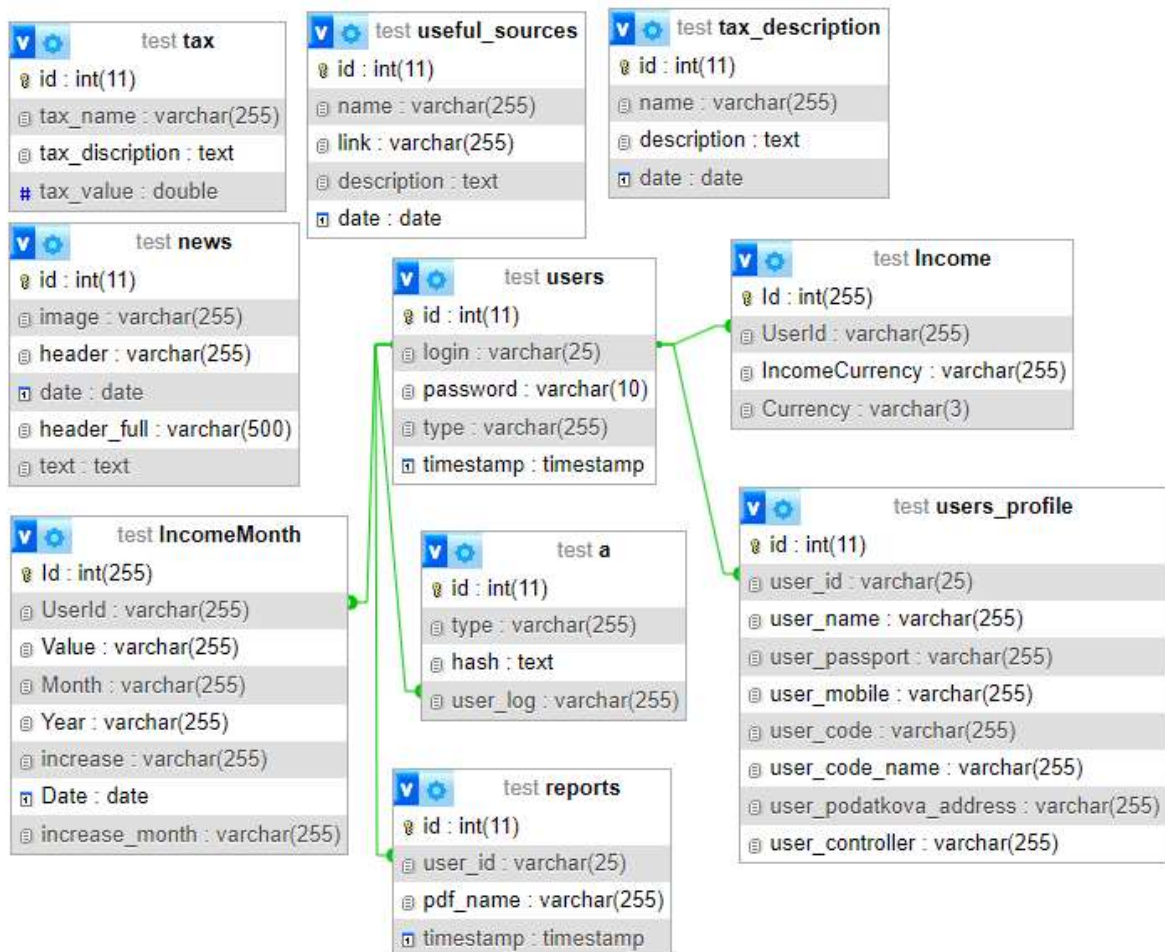


Рисунок 3.4 – Фізична реалізація бази даних

Джерело: власна реалізація.

4 ПРОГРАМНА РЕАЛІЗАЦІЯ

4.1 Архітектура web-системи

На початку програмної реалізації web-орієнтованої системи необхідно провести проектування його архітектури використовуючи діаграму HLD (High Level Design) [26]. HLD – діаграма, яка надає можливість визначити та описати компоненти, що будуть використовуватися в процесі розробки програмного продукту. Інакше кажучи, діаграма дизайну високого рівня являє собою концепцією додатку.

Архітектура web-орієнтованої системи підтримки ведення податкового обліку фізичної особи підприємця складається з:

- вид – відповідає за візуалізацію даних;
- контролер – відповідає за взаємодію з моделлю для отримання та обробки даних;
- модель – складається з двох частин: перша відповідає за взаємодію з системою об'єктно-реляційного відображення; друга – за взаємодію з сервіс-контейнером;
- бази даних – відповідає за зберігання даних.

Діаграма високого рівня представлена на рисунку 4.1.

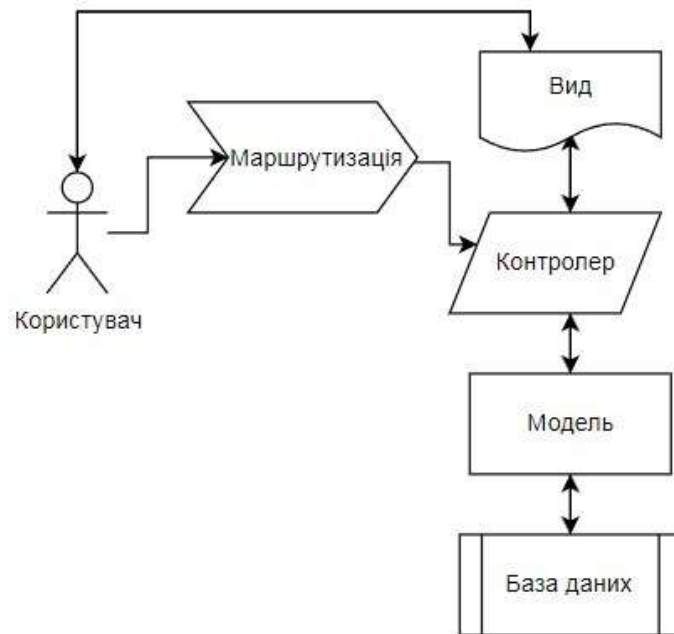


Рисунок 4.1 - Діаграма високого рівня

Джерело:[27]

4.2 Розробка дизайну web-додатку

Перед початком програмної реалізації розроблюваного продукту необхідно створити макет сторінок, єдиний стилістичний шаблон та дизайн сторінок [28]. Для створення макету web-системи необхідно визначити єдину структуру web-сторінок, розміщення основних блоків, єдину кольорову гамму та шрифти. У відповідності з вимогами до програмного продукту, описаних у додатку А, було створено макет сторінки web-системи (рис.4.2).

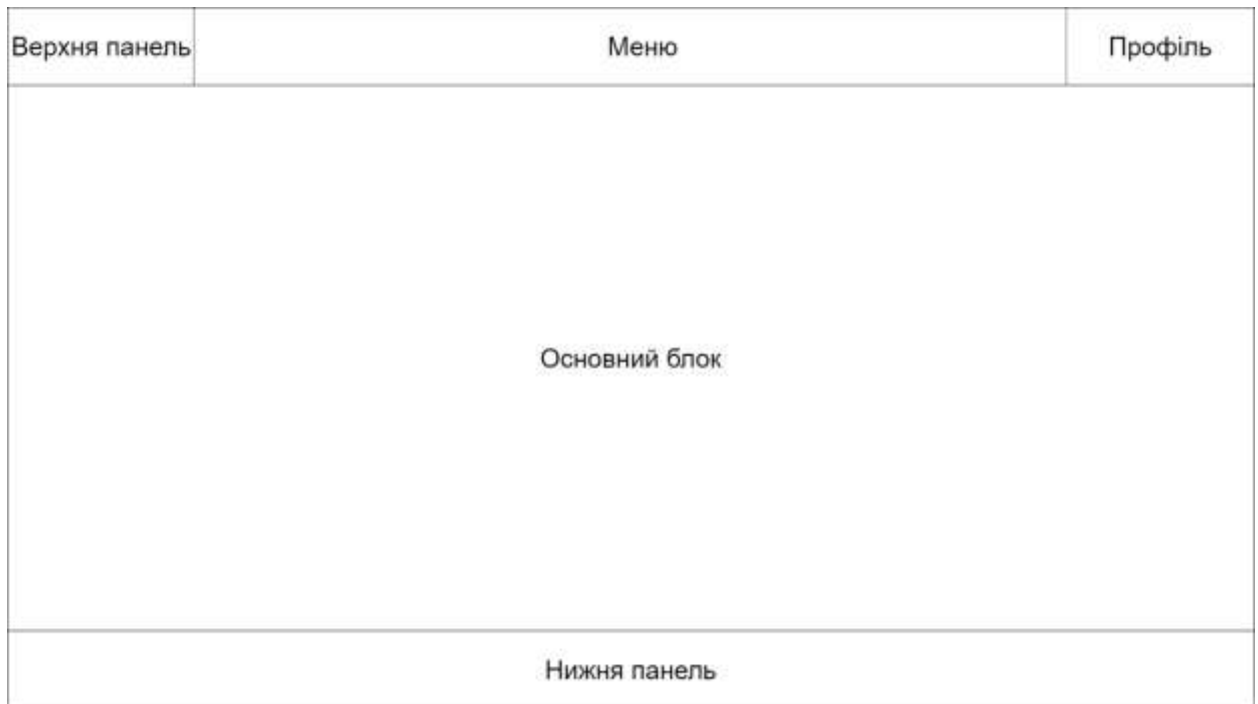


Рисунок 4.2 – Макет сторінки web-системи

Джерело: власна реалізація.

Після погодження макету сторінок web-системи з замовником, було описано єдині стилістичні вимоги до всіх візуальних сутностей розроблюваного проекту. Для вирізнення з-поміж конкурентів, було вирішено обрати м'яку палітру кольорів, а саме відтінки зеленого кольору (#586D73, #AAD0DC), білий та сірий. Також було обрано шрифт-код сторінок, а саме шрифти «Montserrat» та «Roboto».

Після погодження єдиних стилістичних вимог до розроблюваної до web-системи було створено макет дизайну однієї з сторінок – сторінки новин (рис. 4.3).



Рисунок 4.3 – Макет web-сторінки новин

Джерело: власна реалізація.

4.3 Реалізація web-додатку

Першим етапом початку програмної реалізації проекту є налаштування середовища розробки. Розробку було вирішено проводити безпосередньо на хостинг-сервісі. У якості хостинг-сервісу було обрано Fornex.com [29] та в якості рішення було обрано орендувати VPS [30].

Fornex.com має гарну репутацію серед розробників, його сервери розміщені в Європі, що надасть максимальну швидкість взаємодії, та гарантує безпеку. VPS дозволяє гнучко налаштувати необхідну конфігурацію сервера

та має ряд переваг у порівнянні з використанням вже налаштованого рішення, але потребує додаткових знань для налаштування конфігурацій.

Для комфортної розробки безпосередньо на сервері було обрано використати ssh підключення через доповнення ftp-simple до Visual Studio Code [31].

Після здійснення налаштувань конфігурації серверу, було створено структуру папок, для збереження файлів та налаштовано базу даних (рис. 4.4), що зберігатиме таблиці з необхідною інформацією. Також було створено безпосередньо таблиці в БД, приклад структури наведено на рисунку 4.5 для таблиці users_profile.

Таблиця	Дія	Рядки	Тип	Застосування	Розмір	Фрагментовані
*	Переглянути Структура Пошук Вставити Очистити Знищити	317	InnoDB	utf8_general_ci	48.0 КБ	-
income	Переглянути Структура Пошук Вставити Очистити Знищити	3	InnoDB	utf8_general_ci	32.0 КБ	-
incomeMonth	Переглянути Структура Пошук Вставити Очистити Знищити	17	InnoDB	utf8_general_ci	16.0 КБ	-
news	Переглянути Структура Пошук Вставити Очистити Знищити	14	InnoDB	utf8_general_ci	48.0 КБ	-
reports	Переглянути Структура Пошук Вставити Очистити Знищити	3	InnoDB	utf8mb4_general_ci	16.0 КБ	-
tax	Переглянути Структура Пошук Вставити Очистити Знищити	14	InnoDB	utf8_general_ci	16.0 КБ	-
tax_description	Переглянути Структура Пошук Вставити Очистити Знищити	6	InnoDB	utf8_general_ci	36.0 КБ	-
useful_sources	Переглянути Структура Пошук Вставити Очистити Знищити	9	InnoDB	utf8_general_ci	16.0 КБ	-
users	Переглянути Структура Пошук Вставити Очистити Знищити	4	InnoDB	utf8_general_ci	32.0 КБ	-
users_profile	Переглянути Структура Пошук Вставити Очистити Знищити	1	InnoDB	utf8mb4_general_ci	32.0 КБ	-
10 таблиць	Всього	390	InnoDB	utf8mb4_general_ci	272.0 КБ	0 Б

Рисунок 4.4 – Створена база даних

Джерело: власна реалізація.

#	ім'я	Тип	Застосування	Атрибути	Нуль	За замовчуванням	Коментарі	Додатково	Дія
1	id	int(11)			Ні	Немає		AUTO_INCREMENT	Змінити Знищити Більше
2	user_id	varchar(255)	utf8mb4_general_ci		Ні	Немає			Змінити Знищити Більше
3	user_name	varchar(255)	utf8mb4_general_ci		Ні	Немає			Змінити Знищити Більше
4	user_passport	varchar(255)	utf8mb4_general_ci		Ні	Немає			Змінити Знищити Більше
5	user_mobile	varchar(255)	utf8mb4_general_ci		Ні	Немає			Змінити Знищити Більше
6	user_code	varchar(255)	utf8mb4_general_ci		Ні	Немає			Змінити Знищити Більше
7	user_code_name	varchar(255)	utf8mb4_general_ci		Ні	Немає			Змінити Знищити Більше
8	user_podatkova_address	varchar(255)	utf8mb4_general_ci		Ні	Немає			Змінити Знищити Більше
9	user_controller	varchar(255)	utf8mb4_general_ci		Ні	Немає			Змінити Знищити Більше

Рисунок 4.5 – Таблиця users_profile

Джерело: власна реалізація.

Було налаштовано підключення до бази даних та модернізовано раніше створену бібліотеку для спрощення зв'язку з БД, яка зображена на рисунку 4.6.

```
<?php
class DB
{
    private $_connection;
    private static $_instance;

    public static function getInstance()
    {
        if (!self::$_instance) {
            self::$_instance = new self();
        }
        return self::$_instance;
    }

    private function __construct()
    {
        $this->_connection = new mysqli('test', 'test', 'test', 'test');

        if (mysqli_connect_error()) {
            trigger_error("Failed to connect to MySQL: " . mysqli_connect_error(),
                E_USER_ERROR);
        }
    }

    private function __clone()
    {
    }

    public function getConnection()
    {
        return $this->_connection;
    }
}

class Query
{
    public static function getQuery($sql_query)
    {
        $db = DB::getInstance();
        $mysqli = $db->getConnection();

        $sql_result = $mysqli->query($sql_query);
        return $sql_result;
    }

    public static function getLastId()
    {
    }
}
```

Рисунок 4.6 – Розроблений модуль взаємодії з БД

Джерело: власна реалізація.

Кожна сторінка web-системи має 4 обов'язкові файли:

- .html – містить розмітку для відображення сторінки;
- .css – містить стилі сторінки;
- .js – містить код для надання інтерактивності сторінці та налаштування зв'язку зі скриптами .php;

– .php є файлами-скриптами для роботи з базами даних та зв'язку web-системи з серверною частиною.

Усі сторінки містять два спільні блоки – верхнє та нижнє меню (рис. 4.7 – 4.8), ці блоки було розроблено першочергово.

```

<div class="header-menu">
  <div class="header-menu-child">
    <div class="header-menu-child-calc">
      <a href="/gc_modules/certificates/test/public_html/backend/tax_calculator/tax_calculator.html" class="header-menu-calc-link">
        Калькулятор податків
      </a>
    </div>
    <div class="header-menu-child-tax">
      <a href="/gc_modules/certificates/test/public_html/backend/tax_info/tax_info.html" class="header-menu-goals_program">
        Податки
      </a>
    </div>
    <div class="header-menu-child-news">
      <a href="/gc_modules/certificates/test/public_html/backend/news/news_card.html" class="header-menu-progress">
        Новини
      </a>
    </div>
    <div class="header-menu-child-sources">
      <a href="/gc_modules/certificates/test/public_html/backend/useful_sources/useful_sources.html" class="header-menu-income">
        Корисні Джерела
      </a>
    </div>
  </div>
  <div class="header-menu-user">
    <a href="user.html">
      
    </a>
    <div class="header-menu-user-auth">
      
      <div class="header-menu-user-auth-content">
        <a class="header-menu-user-auth-content-item" href="/gc_modules/certificates/test/public_html/backend/profile/profile.html">Профіль</a>
        <a class="header-menu-user-auth-content-item" href="/gc_modules/certificates/test/public_html/backend/analytic/analytica.html">Аналітика</a>
        <a class="header-menu-user-auth-content-item" href="/gc_modules/certificates/test/public_html/backend/report/report.html">Звітність</a>
        <a class="header-menu-user-auth-content-item" onclick="exitInfo()">Вийти</a>
      </div>
    </div>
  </div>
</div>

```

Рисунок 4.7 – Верхнє меню сторінок сайту

Джерело: власна реалізація.

```

<div class="footer">
  <div class="footer-menu">
    <div class="footer-menu-page">
      <a href="/" class="footer-menu-page_name">
        /
      </a>
    </div>
    <div class="footer-menu-links">
      <a href="/gc_modules/certificates/test/public_html/backend/tax_calculator/tax_calculator.html" class="footer-menu-calc-link">
        Калькулятор податків
      </a>
      <br>
      <a href="/gc_modules/certificates/test/public_html/backend/tax_info/tax_info.html" class="footer-menu-goals_programs">
        Податки
      </a>
      <br>
      <a href="/gc_modules/certificates/test/public_html/backend/news/news_card.html" class="footer-menu-progress">
        Новини
      </a>
      <br>
      <a href="/gc_modules/certificates/test/public_html/backend/useful_sources/useful_sources.html" class="footer-menu-income">
        Корисні джерела
      </a>
    </div>
    <div class="footer-menu-address">
      Україна, м. Суми, вул. Пушкіна 4Мод. 24
      <br>
      +38 (096) 991-97-54
      <br>
      chernya2@gmail.com.ua
    </div>
    <div class="footer-menu-socialia">
      <div class="footer-menu-socialia-picture">
        <a href="https://www.facebook.com">
          
        </a>
        <a href="https://www.instagram.com">
          
        </a>
        <a href="https://www.youtube.com">
          
        </a>
      </div>
      <div class="footer-menu-socialia-text">
        Не в соц. мережах
      </div>
    </div>
  </div>
</div>

```

Рисунок 4.8 – Нижнє меню сторінок сайту

Джерело: власна реалізація.

Кожна сторінка окрім верхнього та нижнього меню містить персональне тіло. Приклад коду такого тіла сторінки наведено на рис. 4.9.


```

<div class="user-block">
  <div class="coop-header">
    <div class="coop-title">
      АВТОРИЗАЦІЯ
    </div>
  </div>
  <div class="user-autho-block-form">
    <div class="page-goals-addform-form-main">
      <input class="page-goals-addform-form-inputLogin" type="text" size="70" placeholder="Логін">
      <input class="page-goals-addform-form-inputPassword" type="password" size="70" placeholder="Пароль">
      <button class="page-goals-addform-button-add" onclick="enter()"> Увійти</button>
      <p class="page-goals-addform-registration-button" onclick="registration()">
        Зареєструватись
      </p>
    </div>
  </div>
</div>
</div>

<div id="registration-form" class="registration-form">
  <span class="close-button" onclick="closeForm()">&times;</span>
  <h2>Рєєстрація</h2>
  <form>
    <input class="registration-form-login" type="text" placeholder="Логін" id="login">
    <input class="registration-form-password" type="password" placeholder="Пароль" id="password">
    <input class="registration-form-password" type="password" placeholder="Повторіть пароль" id="confirm-password">
    <button type="button" onclick="register()">Зареєструватись</button>
  </form>
</div>

```

Рисунок 4.9 – Тіло web-сторінки авторизації

Джерело: власна реалізація.

Авторизація користувачів (як клієнтів, так і адміністратора) реалізована через використання хешування [32] та cookies [33]. За процеси хешування, занесення cookies та занесення даних до БД відповідає файл hash.php (рис. 4.10).

```

<?php
if ($_GET['type']== "") {echo "ERROR: type is not set"; exit;}
if ($_GET['login']== "") {echo "ERROR: login is not set"; exit;}

$type = $_GET['type'];
$log = $_GET['login'];

require_once ("db-model-BE.php");

session_start();

function generateCode($length) {
    $chars = "abcdefghijklmnopqrstuvwxyzABCDEFGHIJKLMNOPQRSTUVWXYZ0123456789";
    $code = "";
    $clen = strlen($chars) - 1;
    while (strlen($code) < $length) {
        $code .= $chars[mt_rand(0,$clen)];
    }
    return $code;
}

$hash = md5(generateCode(10));

$result = Query::getQuery("INSERT INTO `a` (`id`, `type`, `hash`, `user_log`) VALUES (NULL, '$type', '$hash', '$log')");
$result1 = Query::getQuery("SELECT max(id) FROM a");

$row = mysqli_fetch_assoc($result1);
$num = $row['max(id)'];

setcookie("type", $type, time()+60*60*24*30, "/", null, null, true);
setcookie("hash", $hash, time()+60*60*24*30, "/", null, null, true);
setcookie("hash_num", $num, time()+60*60*24*30, "/", null, null, true);
setcookie("log", $log, time()+60*60*24*30, "/", null, null, true);

echo $num;
}

```

Рисунок 4.10 – Робота з сесіями користувачів

Джерело: власна реалізація.

Перевірка активної сесії авторизації реалізована шляхом порівняння активних соокієс та даних занесених раніше до бази даних сесій. Також в парочесі перевірки система дізнається статус авторизованого користувача – user або admin. За цей процес відповідає файл check_cookies.php, зображений на рисунку 4.11.

```

<?php

if (isset($_COOKIE["id"])) $id = $_COOKIE["id"];
if (isset($_COOKIE["hash"])) $hash = $_COOKIE["hash"];

require_once ("db-model-BE.php");

$result = Query::getQuery("SELECT `type` FROM `a` WHERE `id_user`=`$id` AND `hash`=`$hash`");

while ($row = mysqli_fetch_assoc($result))
{
    echo $row['type'];
};

}

```

Рисунок 4.11 – Файл check_cookies.php

Джерело: власна реалізація.

Розроблений програмний код основних модулів web-орієнтованої системи підтримки ведення податкового обліку фізичної особи підприємця подано у додатку Б.

4.4 Тестування програмного продукту

Останнім етапом реалізації проекту web-орієнтованої системи підтримки ведення податкового обліку фізичної особи підприємця є тестування та використання програмного продукту.

Функціональне тестування розроблюваного продукту складається з етапів [34]:

- тестування посилань web-сторінок: посилання та кнопки на сторінках;
- тестування модуля «Аналітика»: коректність передачі даних та коректність розрахунків;
- тестування модуля «Звітність»: коректність генерації звіту та його завантаження на девайс;
- тестування модуля «Профіль»: коректність збереження даних;
- тестування розрахунку податків ФОП.

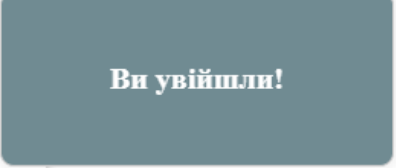
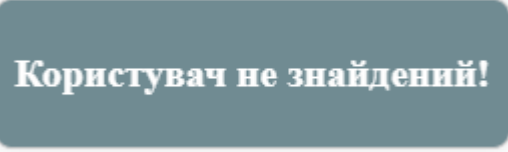
Процес тестування було розпочато з перевірки роботоспроможності пунктів меню. Усі пункти меню та підвала працюють коректно та посилаються на коректні web-сторінки. Також коректно працюють гіперпосилання на соціальні мережі.

Модулі «Аналітика», «Звітність», «Профіль» та калькулятори податків ФОП було протестовано з використанням техніки Statement Coverage (SC) [35]. Statement Coverage передбачає, що кожна функціональна умова належить бути перевірена хоча б один раз. Усі результати проведених тестувань представлено у табл. 4.1.



Таблиця 4.1 – Тестування web-системи

Назва	Очікуваний результат	Фактичний результат	0/1
Перевірка розрахунків податків для ФОП 1 групи	Податок розраховано коректно, користувач отримав результат		1
Перевірка розрахунків податків для ФОП 2 групи	Податок розраховано коректно, користувач отримав результат		1
Перевірка розрахунків податків для ФОП 3 групи	Податок розраховано коректно, користувач отримав результат		1
Перевірка розрахунків податків для ФОП 4 групи	Податок розраховано коректно, користувач отримав результат		1

Продовження табл.4.1

Перевірка розрахунків податків для ФОП загальної групи	Податок розраховано коректно, користувач отримав результат		1
Перевірка реакції калькулятора на від'ємне значення	Податок розраховано коректно, значення взято за модулем		1
Вхід до акаунту користувача з коректними даними	Повідомлення про вдалий вхід та авторизація		1
Вхід до акаунту користувача з некоректними даними	Повідомлення про те, що користувач не існує та не здійснення авторизації		1
Вихід з акаунту користувача	Повідомлення про вихід з акаунта та безпосередньо вихід		1

Продовження табл. 4.1.

Збереження даних в «Профіль»	Збереження даних, та повідомлення про це		1
Додавання нового запису в «Аналітика»	Відображення нового запису та динамічна зміна аналітики	Новий запис відображено та аналітика була змінена динамічно	1
Видалення нового запису в «Аналітика»	Видалення запису та динамічна зміна аналітики	Запис видалено та аналітика була змінена динамічно	1
Генерація звіту	Генерація звіту та відповідне повідомлення про це		1
Завантаження звіту на локальний девайс	Початок завантаження файлу		1

Джерело: власна реалізація.

Завершивши виконання тестування можна зробити висновок: розроблюваний програмний продукт не має значних дефектів статусу Blocker чи Major.

Також було проведено тестування серверу на навантаження під час користування web-додатком. Конфігурація сервера [36]: 2 ядра центрального процесору (ЦП), 2 Гб оперативної пам'яті та 25Гб постійної. Під час користування додатком з чотирьох девайсів навантаження на ЦП не

перевищило 1%, навантаження оперативної пам'яті – не більше 25 Мб. Грунтуючись на цих даних можна зробити висновки, що web-додаток не є вибагливим до серверу, тому працюватиме при середніх навантаженнях навіть на слабких серверах з 1-2 ядрами центрального процесора та 1-2 Гб оперативної пам'яті.

Отже web-орієнтована система підтримки ведення податкового обліку фізичної особи підприємця готова до передачі замовнику.

4.5 Демонстрація роботи web-додатку

Головною сторінкою web-системи є сторінка новин, що складається з трьох областей: верхнього меню, нижнього меню та основного блоку, який складається з двох блоків: із заголовком та саме блок із новинами (рис. 4.12).



Рисунок 4.12 – Web-сторінка «Новини»

Джерело: власна реалізація.

Основним модулем розроблюваного проекту є сторінка «Калькулятор податків» (рис. 4.13). Основний блок модуля містить варіанти калькуляторів податків. Головним і найбільш функціональним є калькулятор податків фізичних осіб-підприємців – рисунок 4.14.



Рисунок 4.13 – Web-сторінка «Калькулятор податків»

Джерело: власна реалізація.

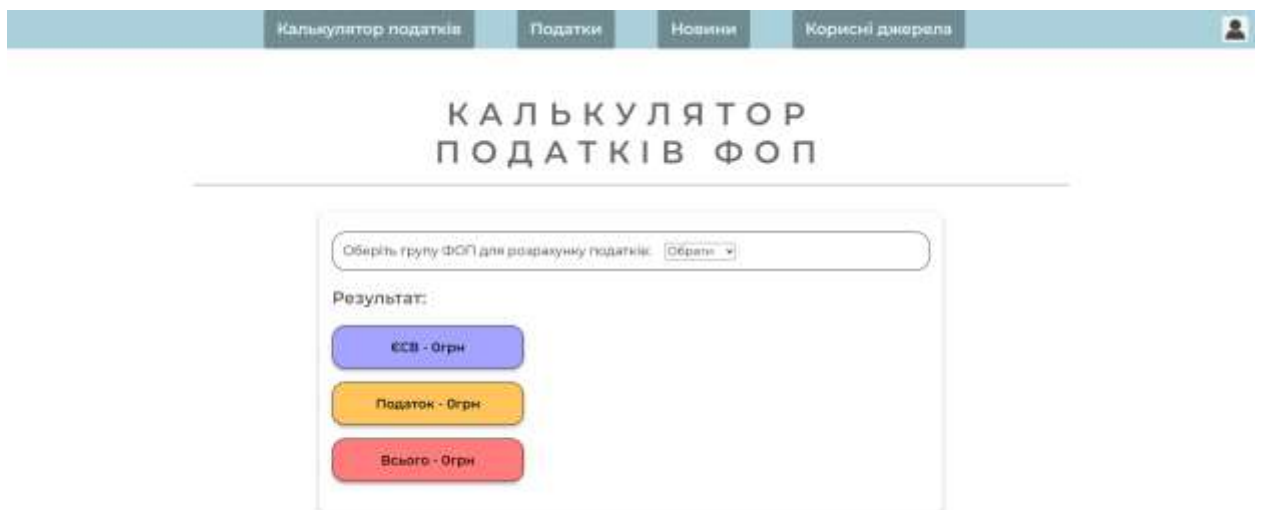


Рисунок 4.14 – Web-сторінка з калькулятором податків фізичних осіб-підприємців

Джерело: власна реалізація.

Калькулятор є інтерактивним [37] та в залежності від обраної групи ФОП змінюються формули розрахунків та зовнішній вигляд форми калькулятора. Розглянемо усі можливі варіації вигляду калькулятора податків ФОП на рисунках 4.15-4.19.

**КАЛЬКУЛЯТОР
ПОДАТКІВ ФОП**

Оберіть групу ФОП для розрахунку податків: Перша ▾

Результат:

ЄСВ - 1430

Єдиний податок - 248.1

Всього - 1678.1

Рисунок 4.15 – Вигляд калькулятора для ФОП першої групи
Джерело: власна реалізація.

**КАЛЬКУЛЯТОР
ПОДАТКІВ ФОП**

Оберіть групу ФОП для розрахунку податків: Друга ▾

Результат:

ЄСВ - 1474

Єдиний податок - 1340

Всього - 2814

Рисунок 4.16 – Вигляд калькулятора для ФОП другої групи
Джерело: власна реалізація.

КАЛЬКУЛЯТОР ПОДАТКІВ ФОП

Оберіть групу ФОП для розрахунку податків: Третя ▾

Вкажіть прибуток за місяць: Прибуток за місяць

Результат:

- ЄСВ - 0
- Єдиний податок - 0
- Всього - 0

Рисунок 4.17– Вигляд калькулятора для ФОП третьої групи
Джерело: власна реалізація.

КАЛЬКУЛЯТОР ПОДАТКІВ ФОП

Оберіть групу ФОП для розрахунку податків: Четверта ▾

Оберіть % за категорією вашої землі: 0,19 ▾

Фактична оцінка власності: Прибуток за місяць

Результат:

- Всього - 0

Рисунок 4.18 – Вигляд калькулятора для ФОП четвертої групи
Джерело: власна реалізація.

КАЛЬКУЛЯТОР ПОДАТКІВ ФОП

Оберіть групу ФОП для розрахунку податків: Загальна ▾

Вкажіть прибуток за місяць:

Результат:

- ЄСВ - 0
- Єдиний податок - 0
- Всього - 0

Рисунок 4.19 – Вигляд калькулятора для ФОП загальної групи
Джерело: власна реалізація.

Розрахунки податків відбуваються у зв'язці .js та .php скриптів, де перші виконують роботу на стороні web-сторінок, а другі – на стороні серверу [38].

Також було розроблено модуль реєстрації нового користувача. Модуль реєстрації зображено на рисунку 4.20.

Калькулятор податків | Податки | Рівняння | Корисні джерела

АВТОРИЗАЦІЯ

Реєстрація

Ім'я:

Прізвище:

Електронна пошта:

Рисунок 4.20 – Web-модуль реєстрації
Джерело: власна реалізація.

Під час реєстрації обов'язково відбувається валідація нового користувача, перевіряються паролі на співпадіння між собою, та на наявність вже зареєстрованого користувача.

Наступним модулем розроблюваного проекту є «Профіль». У даному розділі користувачу пропонується заповнити особисту інформацію для подальшої роботи з нею. Модуль зображено на рисунку 4.21.

Рисунок 4.21 – Web-сторінка «Профіль»

Джерело: власна реалізація.

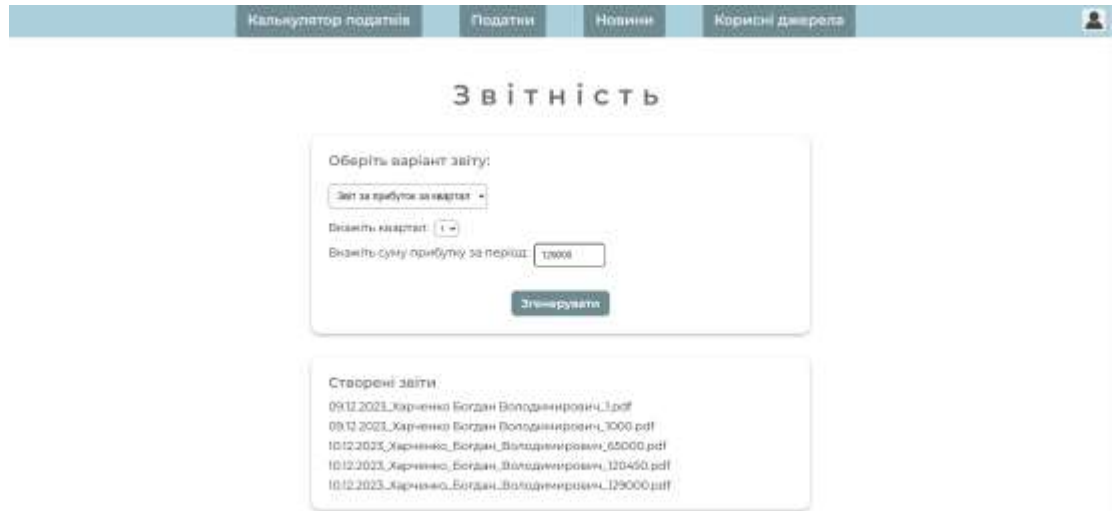
Наступним модулем розроблюваного проекту є «Аналітика». Дана сторінка складається з двох розділів: прибутку та висновків. Перший розділ являє собою інтегровану систему ведення обліку фінансових надходжень у розрізі місяця. Другий розділ містить короткі висновки щодо фінансової діяльності за категоріями місяця, року та тренду. Сторінка зображена на рисунках 4.22.



Рисунки 4.22 – Web-сторінка «Аналітика»

Джерело: власна реалізація.

Також було розроблено модуль «Звітність». Даний модуль має функціонал генерації обраного варіанту звіту для подальшої подачі до податкової служби. Наразі реалізовано функціонал генерації звітності для ФОП 3 групи на спрощеній системі оподаткування. Після генерації звіту він з'явиться у блоці «Створені звіти», звідки його можна завантажити на девайс. Сторінка «Звітність» зображена на рисунку 4.23.



Рисунки 4.23 – Web-сторінка «Звітність»

Джерело: власна реалізація.

Генерація відбувається за допомогою зв'язки .js та .php. За основу було обрано актуальний шаблон звітності, який заповнюється на стороні .php необхідними даними користувача. Усі дані тягнуться з модулю «Профіль» у разі, якщо «Профіль» не було заповнено – користувач отримає відповідне повідомлення. Генерація відбувається в форматі png, після чого відбувається конвертація у більш зручний формат pdf [39] за допомогою інтегрованої бібліотеки TCPDF [40].

Завантаження на локальний пристрій згенерованого звіту відбувається за рахунок report.js та download-report.php файлів, що емітують роботу браузера та починають завантаження файлу за посиланням.

Процес збереження даних в пункті «Профіль» зображено на рисунку 4.24. Видно, як сторінка надає зворотній зв'язок після збереження даних.

Рисунки 4.24 – Web-сторінка «Профіль»

Джерело: власна реалізація.

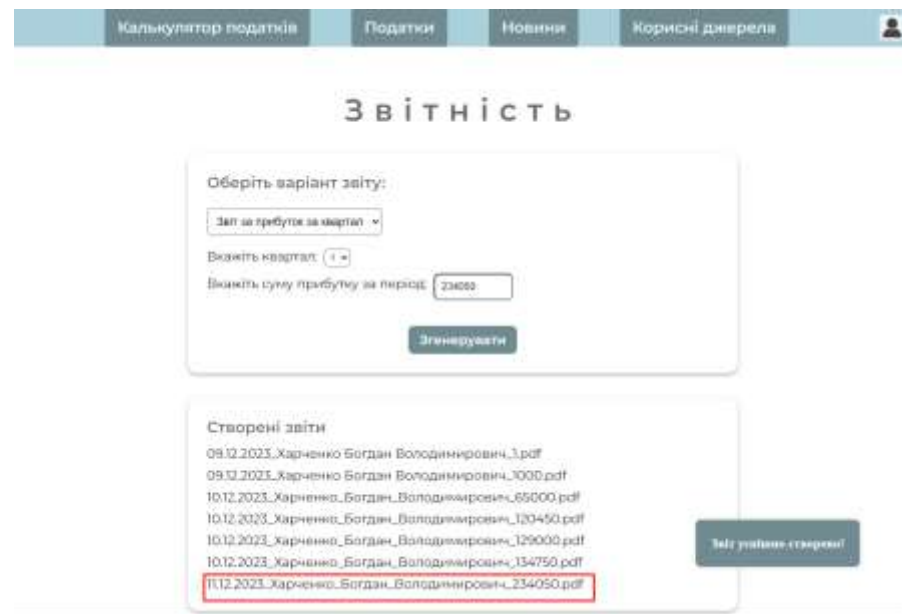
Сторінку «Аналітика» зображено на рисунку 4.25. В модулі ведення обліку фінансової діяльності побудовано графік на базі даних користувача. У розділі «Висновки» знаходяться короткі висновки діяльності у розрізі місяця та року, а також щодо тренду.



Рисунки 4.25 – Web-сторінка «Аналітика»

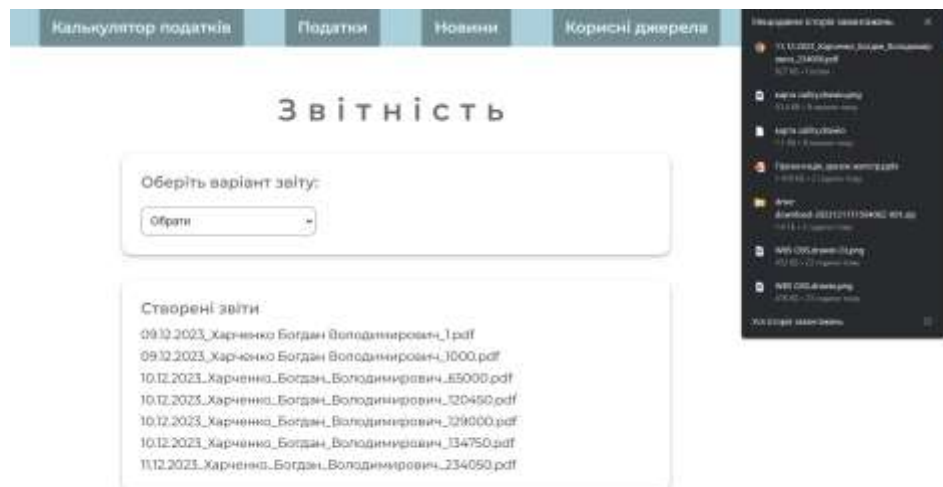
Джерело: власна реалізація.

Сторінку «Звітність» зображено на рисунку 4.26. На рисунку видно, як сторінка надає зворотній зв'язок щодо створення нового звіту на базі заданих користувачем даних, а саме звіт за другий квартал з прибутком 234050 грн. В розділі «Створені звіти» збережено всі створені раніше звіти користувача. Після обирання звіту та натискання на нього починається завантаження документу на локальний девайс (рис. 4.27).



Рисунки 4.26 – Web-сторінка «Звітність» - створення звіту

Джерело: власна реалізація.



Рисунки 4.27 – Web-сторінка «Звітність» - завантаження звіту

Джерело: власна реалізація.

Модуль «Калькулятор податків ФОП» зображено на рисунку 4.28. На рисунку видно, як було розраховано розмір податку для третьої групи ведення фізичної особи-підприємця на базі заданих користувачем даних, а саме 13407 грн.

Калькулятор податків Податки Новини Корисні джерела

КАЛЬКУЛЯТОР ПОДАТКІВ ФОП

Оберіть групу ФОП для розрахунку податків: Третя

Вкажіть прибуток за місяць: 13407

Результат:

- ЄСВ - 1474
- Єдиний податок - 670.35
- Всього - 2144.35

Рисунки 4.28 – Web-сторінка «Калькулятор податків ФОП»

Джерело: власна реалізація.

ВИСНОВКИ

Результатом виконання кваліфікаційної роботи магістр є створений web-орієнтована система підтримки ведення податкового обліку фізичної особи підприємця.

У ході виконання проекту було проведено дослідження предметної області web-орієнтованої системи підтримки ведення податкового обліку, а також проведено детальний аналіз наявних аналогів та визначено вимоги і характеристики системи. У результаті була підтверджена актуальність розроблюваного проекту. Як наслідок, було сформульовано детальні мету та тему задачі, методи та технології для розробки кваліфікаційної роботи на ступінь магістр «Web-орієнтована система підтримки ведення податкового обліку фізичної особи підприємця». Також було проведено планування робіт, визначено реагування на всі ризики, що можуть виникнути під час розробки програмного продукту.

Під час реалізації проектної частини кваліфікаційної роботи було проведено структурно-функціональне моделювання та розроблено діаграму варіантів використання web-системи. Також були визначені сутності системи та їх атрибути, а також розроблена база даних для зберігання інформації.

У практичній частині роботи було розроблено діаграму дизайну високого рівня разом зі шаблоном дизайну сторінок та встановлено єдиний стиль розроблюваної web-системи. Також було виконано програмну реалізацію, продемонстровано роботу системи та проведено його тестування з використанням техніки Statement Coverage.

Розроблена web-система сприятиме заощадженню фінансових та часових витрат підприємців на веденні податкової діяльності. Продукт орієнтований на фізичних осіб, які планують, або вже займаються підприємницькою діяльністю в якості фізичних осіб підприємців.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Державна податкова служба України [Електронний ресурс] – Режим доступу до ресурсу: <https://tax.gov.ua/>.
2. Деякі особливості адміністративно-правового статусу суб'єктів податкових правовідносин [Електронний ресурс] – Режим доступу до ресурсу: <https://science.lpnu.ua/sites/default/files/journal-paper/2018/may/12814/284-290.pdf>.
3. Єдиний державний реєстр фізичних осіб-підприємців [Електронний ресурс] – Режим доступу до ресурсу: <https://minjust.gov.ua/m/edini-ta-derjavni-reestri>.
4. Система оподаткування в Україні та податкове законодавство [Електронний ресурс] – Режим доступу до ресурсу: <https://ank.odessa.ua/konsultatsii/opodatkuvannya-v-ukrayini/>.
5. Податкова карта України [Електронний ресурс] – Режим доступу до ресурсу: <https://map.tax.gov.ua/main>.
6. Міністерство фінансів України [Електронний ресурс] – Режим доступу до ресурсу: <https://www.mof.gov.ua/uk>.
7. Електронний кабінет [Електронний ресурс] – Режим доступу до ресурсу: <https://cabinet.tax.gov.ua/>
8. Реєстрація та облік фізичних осіб - платників податків [Електронний ресурс] – Режим доступу до ресурсу: <https://tax.gov.ua/fizichnim-osobam/reestratsiya-ta-oblik-fiz>.
9. Економіка українських ФОПів в реальному часі [Електронний ресурс] – Режим доступу до ресурсу: <https://opendatabot.ua/open/foronomics>.
10. Податковий кодекс України [Електронний ресурс] – Режим доступу до ресурсу: <https://zakon.rada.gov.ua/laws/show/2755-17#Text>.
11. Taxer.ua [Електронний ресурс] – Режим доступу до ресурсу: <https://taxer.ua/uk/>

12. Talent.com [Електронний ресурс] – Режим доступу до ресурсу: <https://ua.talent.com/tax-calculator?salary=50000&from=month®ion=Ukraine>
13. What is Intuitive Interaction? Balancing Users’ Performance and Satisfaction with Natural User Interfaces [Електронний ресурс] – Режим доступу до ресурсу: <https://academic.oup.com/iwc/article-abstract/27/3/357/2358470?redirectedFrom=fulltext>
14. HTML5 and the evolution of HTML; tracing the origins of digital platforms [Електронний ресурс] – Режим доступу до ресурсу: <https://www.sciencedirect.com/science/article/abs/pii/S0160791X2100004X>.
15. HTML and CSS: Design and Build Websites by Jon Duckett. – м. New York, John Wiley & Sons. 2011. – 490 с.
16. Learn JavaScript VISUALLY Ivelin Demirov. – Japanese, Nai Inc. 2014. – 164 с.
17. PHP & MySQL: Novice to Ninja by Tom Butler and Kevin Yank. – м. Мельбурн, SitePoint. 2017. – 664 с.
18. Learning PHP, MySQL & JavaScript: With jQuery, CSS & HTML5. – м. Себастопол, O'Reilly Media, Incorporated . 2018. – 827 с.
19. Система оподаткування в Україні та податкове законодавство [Електронний ресурс] – Режим доступу до ресурсу: <https://ank.odessa.ua/konsultatsii/opodatкування-v-ukrayini/>.
20. Особливості нарахування та сплати ЄСВ фізичними особами – підприємцями [Електронний ресурс] – Режим доступу до ресурсу: <https://dp.tax.gov.ua/media-ark/local-news/print-444182.html>.
21. What is The IDEF0 Diagram [Електронний ресурс] – Режим доступу до ресурсу: <https://www.edrawmax.com/article/the-complete-guide-to-understand-idef-diagram.html>
22. IDEF0 constraint (decomposition) diagrams [Електронний ресурс] – Режим доступу до ресурсу: https://support.unicomsi.com/manuals/systemarchitect/114100/starthelp.html#page/Architecting_and_designing/IDEF1019213.html.

23. UML Use Case Diagram Tutorial [Електронний ресурс] – Режим доступу до ресурсу: <https://www.lucidchart.com/pages/uml-use-case-diagram>.

24. Database structure: the building blocks of a database [Електронний ресурс] – Режим доступу до ресурсу: <https://www.lucidchart.com/pages/database-diagram/database-design#:~:text=Database%20structure%3A%20the%20building%20blocks%20of%20a%20database,-The%20next%20step&text=Within%20a%20database%2C%20related%20data,and%20columns%2C%20like%20a%20spreadsheet>.

25. Efficient MySQL Performance: Best Practices and Techniques. – м. Себастопол, O'Reilly Media, Inc., 2021. – 322 с.

26. What is High Level Design – Learn System Design [Електронний ресурс] – Режим доступу до ресурсу: <https://www.geeksforgeeks.org/what-is-high-level-design-learn-system-design/>.

27. Харченко Б.В., Ващенко С.М. Web-орієнтований довідник-калькулятор податків [Електронний ресурс] – Режим доступу до ресурсу: https://essuir.sumdu.edu.ua/bitstream-download/123456789/88818/1/Kharchenko_bac_rob.pdf

28. 10 principles of good website design [Електронний ресурс] – Режим доступу до ресурсу: https://www.wix.com/studio/blog/good-website-design?utm_source=google&utm_medium=cpc&utm_campaign=20415922074^153462532217&experiment_id=^^682623789509^&gclid=CjwKCAiApuCrBhAuEiwA8VJ6JrLa17yj3CXml1kzbOJQqy-DsPbIwfwmdjNLbWhY33a4mrcKxEI8-hoCXpgQA_vD_BwE.

29. Fornex.com [Електронний ресурс] – Режим доступу до ресурсу: <https://fornex.com/en/>.

30. What is VPS? Pros & Cons of VPS Hosting [Електронний ресурс] – Режим доступу до ресурсу: <https://hostingjournalist.com/what-is-vps-pros-cons-of-vps-hosting/>.

31. Ftp-simple [Електронний ресурс] – Режим доступу до ресурсу: <https://marketplace.visualstudio.com/items?itemName=humy2833.ftp-simple>
32. How to create hash from string in JavaScript? [Електронний ресурс] – Режим доступу до ресурсу: <https://www.geeksforgeeks.org/how-to-create-hash-from-string-in-javascript/>.
33. Using HTTP cookies [Електронний ресурс] – Режим доступу до ресурсу: <https://developer.mozilla.org/en-US/docs/Web/HTTP/Cookies>.
34. Функціональне тестування сайту [Електронний ресурс] – Режим доступу до ресурсу: <https://brainlab.com.ua/uk/blog-uk/yak-testuvati-veb-sayt-osnovn-etapi-poradi>
35. Statement Coverage [Електронний ресурс] – Режим доступу до ресурсу: <https://www.codium.ai/glossary/statement-coverage/#:~:text=Statement%20coverage%20testing%20is%20a,during%20the%20preliminary%20code%20review>.
36. How should I choose a new server hardware configuration? [Електронний ресурс] – Режим доступу до ресурсу: <https://www.techtarget.com/searchdatacenter/answer/How-should-I-spec-our-new-server-hardware-configuration>.
37. Як створити привабливі та інтерактивні веб-сайти [Електронний ресурс] – Режим доступу до ресурсу: <https://redstone.media/yak-stvoryty-privablyvi-ta-interaktyvni-veb-saity>
38. How to Use PHP, JavaScript (or jQuery), and HTML Together [Електронний ресурс] – Режим доступу до ресурсу: <https://cullenwebservices.com/how-to-use-php-javascript-or-jquery-and-html-together/>
39. PDF Explained by John Whittington. – м. Севастопол, O'Reilly Media, Inc., 2011. – 140 с.
40. TCPDF Open Source PHP class for generating PDF documents [Електронний ресурс] – Режим доступу до ресурсу: <https://tcpdf.org/>.

ДОДАТОК А

ПЛАНУВАННЯ РОБІТ

А.1 МЕТА ПРОЕКТУ

Метою даного дипломного проекту є заощадження фінансових та часових витрат підприємців на веденні податкової діяльності за рахунок web-орієнтованої системи підтримки ведення податкового обліку.

Створюваний програмний продукт дасть змогу скоротити час на розрахунку податків, веденні фінансового обліку та створенні податкової звітності. Також розроблюваний додаток дозволить фізичним особам підприємцям відмовитись від послуг бухгалтера.

Web-додаток орієнтований на фізичних осіб, що планують, або вже займаються підприємницькою діяльністю в якості ФОП. Також дане програмне забезпечення стане в допомогу громадянам в питаннях ознайомлення з нормами податкової системи України, а також розрахунку їх податків.

Для успішності було деталізовано мету web-додатку за допомогою SMART-методу. Результати деталізації представлені у таблиці А.1.

Таблиця А.1 – Деталізація мети проекту методом SMART

Specific	Розробка web-орієнтованої системи підтримки ведення податкового обліку
Measurable	Коректно розраховані податки та створена податкова звітність користувача.
Achievable	Мета досяжна, є затверджене технічне завдання, необхідні програмні засоби.
Relevant	Для полегшення ведення податкового обліку.
Time-framed	Термін встановлено – до кінця 5 курсу.

Джерело: власна реалізація.

А.2 ПЛАНУВАННЯ ЗМІСТУ РОБІТ

Ієрархічна структура робіт (WBS) є ієрархічним планом завдань, що є основою кожного проекту. WBS дозволяє поділити проект на його складові частини. За кожною такою частиною є закріплений перелік виконуваних робіт, що необхідні для реалізації проекту.

Використовуючи WBS всі задіяні в проекті працівників отримують перевагу в чіткому розумінні етапів виконання завдання шляхом встановлення ієрархічно структурованого розподілу робіт з реалізації проекту.

В процесі планування тривалості кожного етапу методами WBS враховуються попередні та наступні завдання. У результаті такого планування отримується чіткий план, що дозволяє на будь-якому етапі бачити, як проект має розвиватися. Такий інструмент дозволяє здійснювати керування проектом належним чином, корегуючи його згідно плану.

Частини WBS включають:

- завдання: ідентифікатор, назва та опис;
- залежність завдань: об'єднання декількох завдань, якщо між ними є залежність від виконання іншого;
- дата початку та завершення завдання: оцінка часу, який займе кожен етап та весь проект вцілому;
- оцінка роботи: скільки часу потрібно для виконання завдання;
- статус завдання: розпочато, виконується, завершено тощо;
- діаграма Ганта: візуалізація WBS із завданнями, представленими графічно на часовій прямій.

WBS діаграму, що була створена згідно діаграми Ганта та технічного завдання зображено на рисунку А.1.

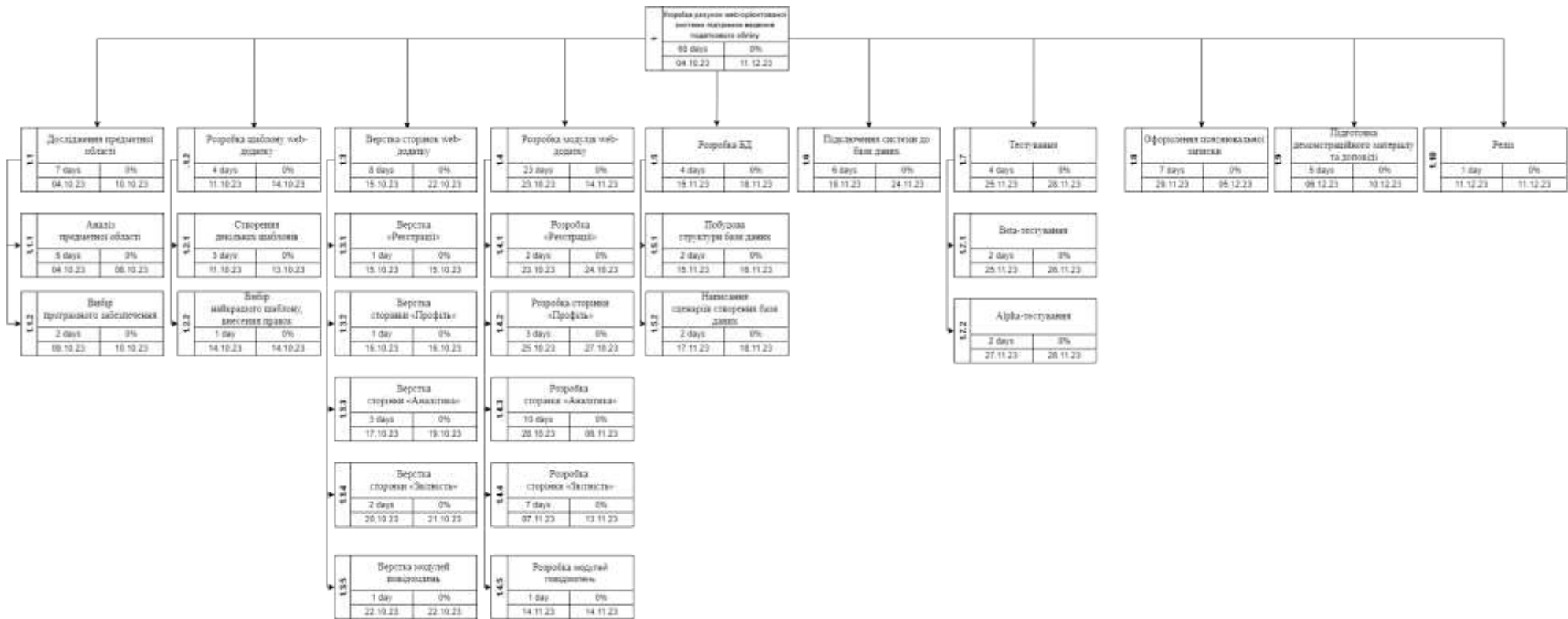


Рисунок А.1 – WBS структура робіт проекту

Джерело: власна реалізація.

А.3 ПЛАНУВАННЯ СТРУКТУРИ ВИКОНАВЦІВ

Організаційна структура проекту (OBS) є ієрархічною моделлю, яка описує установлену організаційну структуру для планування проекту, управління ресурсами, відстежування часу та витрат, розподілу витрат, звітності про доходи/прибутки та управління роботою.

Графічне відображення учасників проекту та їх відповідальних осіб, що задіяні в реалізації проекту, представлене в організаційній структурі, яка об'єднує подібні заходи проекту та пов'язує їх із структурою організації. OBS використовується для призначення відповідальних за управління проектом, звітності про витрати, виставлення рахунків, складання бюджету та контролю проекту, встановлюючи організаційну перспективність проекту.

Після етапу розподілення обов'язків та призначення завдань, структури OBS і WBS надають обширний аналітичний спектр для вимірювання продуктивності проекту та робочої сили.

Етапи розробки структури розподілу організації включають:

- зображення всієї організації у вигляді ієрархії;
- визначення відділів та груп;
- визначення функціональних груп (розподілення вартості роботи)

та груп схвалення (схвалення роботи) для кожного користувача.

OBS діаграма, створена згідно з діаграмою Ганта та технічного завдання, зображена на рисунку А.2.

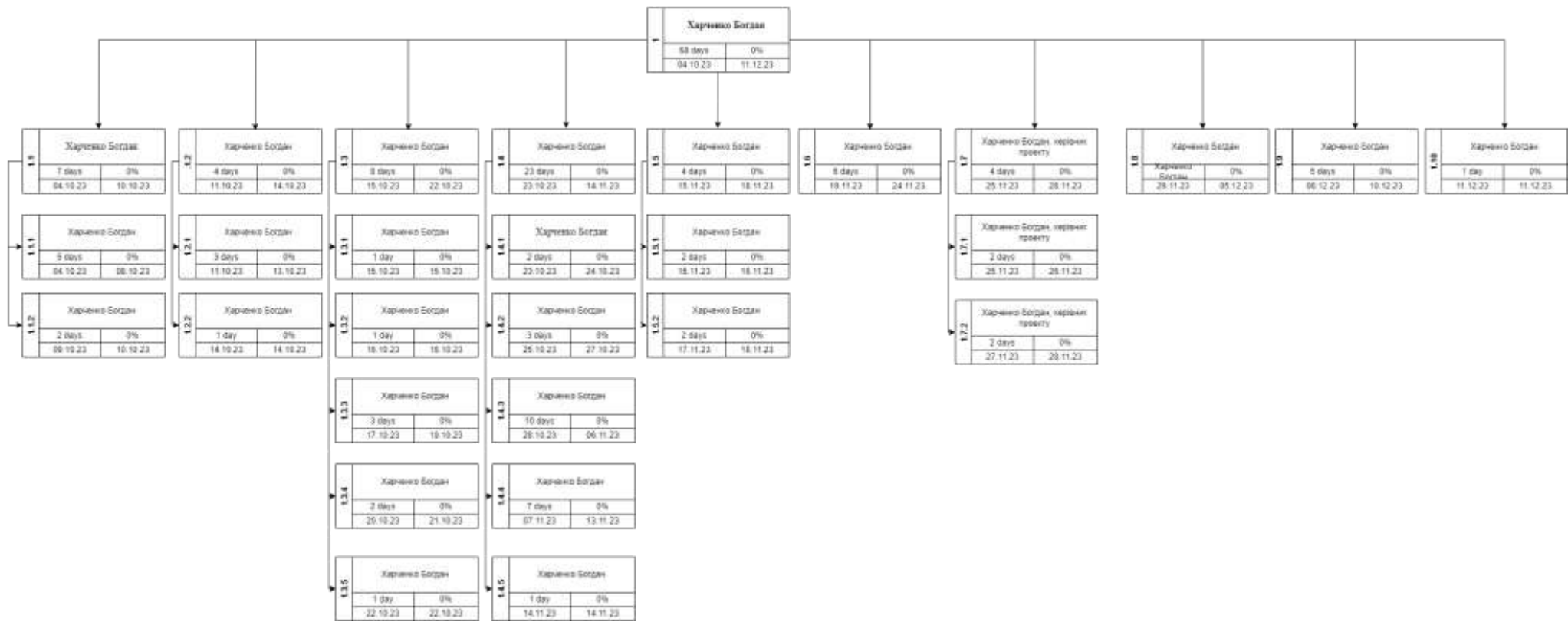


Рисунок А.2 – OBS структура робіт проекту

Джерело: власна реалізація.

Таблиця А.2 – Виконавці проекту

Роль	Ім'я	Проектна роль
Розробник	Харченко Б. В.	Розробляє проект
Проектувальник	Харченко Б. В.	Розроблює структуру web-додатку
Дизайнер	Харченко Б. В.	Виконує дизайн усіх елементів додатку
Тестувальник	Харченко Б. В., керівник проекту	Відповідає за тестування продукту
Керівник проекту	Ващенко С. М.	Формує технічне завдання на розробку проекту

Джерело: власна реалізація.

А.4 ДІАГРАМА ГАНТА

Планування проекту включає важливий етап – розробку календарного графіку або діаграми Ганта. Цей графік представляє собою розклад виконання завдань із реальними датами, надаючи можливість точно оцінити тривалість процесів з урахуванням обмежень ресурсів, вихідних днів та свят.

Діаграма Ганта є ефективним інструментом управління проектами та ілюструє план проекту, надаючи графічне представлення кожного етапу на часовій шкалі та містить інформацію про завдання виконання, їхні дати початку та завершення, залежності між завданнями та виконавців.

На рисунку А.3 зображено діаграму Ганта, яку було побудовано відповідно до технічного завдання.

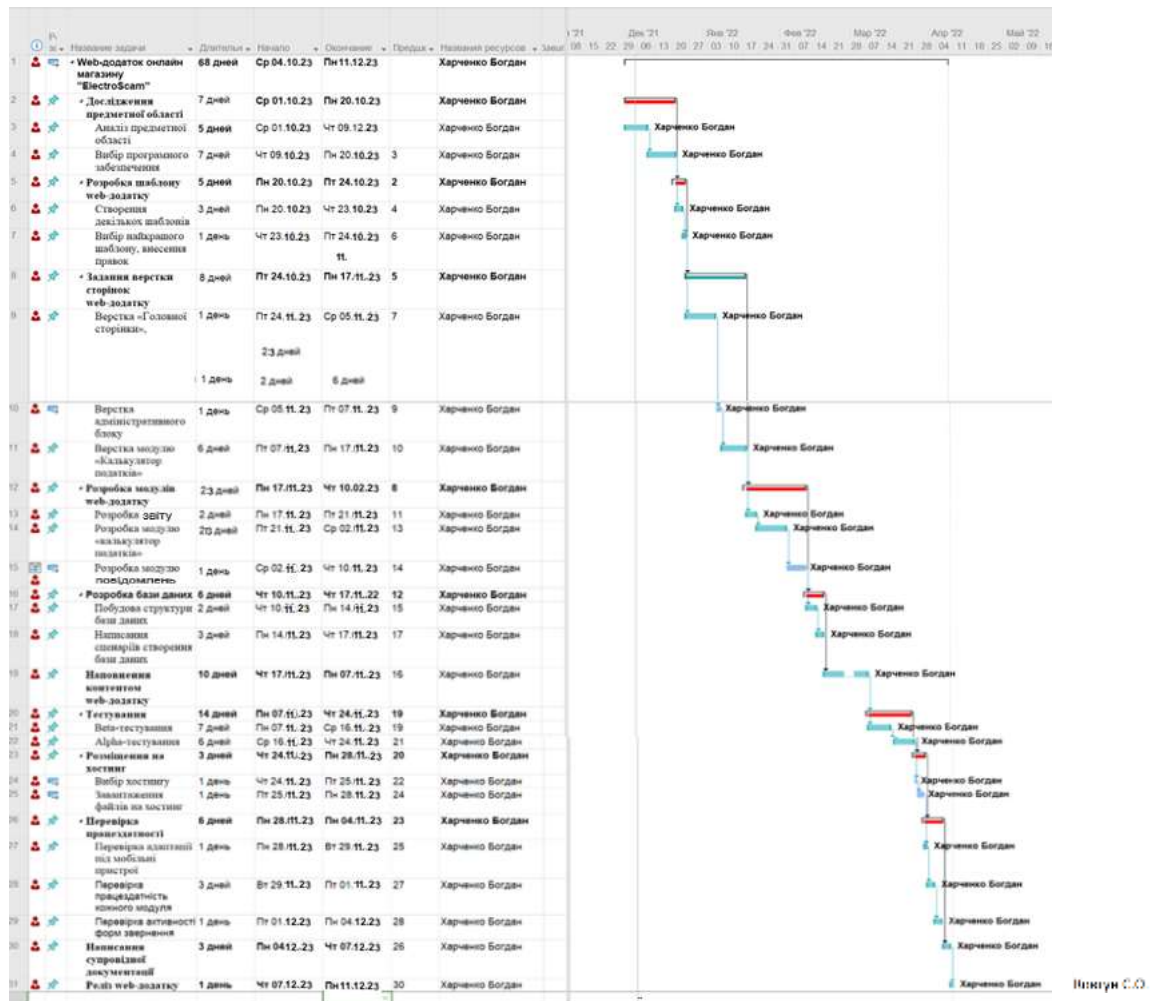


Рисунок А.3 – Календарний графік проекту

Джерело: власна реалізація.

А.5 УПРАВЛІННЯ РИЗИКАМИ ПРОЕКТУ

Для управління ризиками розроблюваного проекту здійснюється аналітична робота для виявлення усіх можливих факторів. Реагування на можливі ризики залежить від рівня його важливості. Після виявлення факторів здійснюється кількісна оцінка.

Якісне та кількісне оцінювання ризиків допомагає побудувати стратегію уникнення проблем під час розробки проекту. Оцінювання можуть проводитись окремо, або одночасно.

Шкалу для класифікації ризиків за величиною впливу на проект та ймовірністю виникнення представлено у таблиці А.3.

Таблиця А.3 – Шкала класифікації ризиків за ймовірністю виникнення та величиною впливу.

Оцінка	Ймовірність	Вплив ризику	Тип ризику
1	Низька	Низький	Прийнятні
2	Середня	Середній	Виправдані
3	Висока	Високий	Недопустимі

Джерело: власна реалізація.

Для зменшення негативного впливу ризиків на проект проводиться планування реагування на них. Це планування містить визначення ефективності розробки та оцінку наслідків їх впливу на проект. Оцінювання здійснюється за показниками, описаними у таблиці А.3.

Ризики розроблюваного проекту та стратегії реагування на них описані у таблиці А.4.

Таблиця А.4 – Матриця ймовірності та впливу згідно проекту

ID	Статус ризику	Опис ризику	Ймовірність виникнення	Вплив ризику	Ранг ризику	План А	Тип стратегії реагування	План Б
1	Відкритий	Вибір неефективних технологій розробки	Малий	Малий	0,1	Відповідально підійти до питання вибору технологій реалізації проекту	Зменшення	Переоцінка обраних технологій для реалізації проекту та корегування цього списку
2	Відкритий	Помилки проектування та налагодження бази даних	Дуже малий	Середній	0,05	На етапі створення БД урахувати усі майбутні маніпуляції з даними у базі	Зменшення	Переробити БД згідно з виявленими вимогами.
3	Відкритий	Відсутність резервних копій даних	Дуже малий	Великий	0,05	Налаштувати автоматичне збереження даних	Зменшення	Відновити данні
4	Відкритий	Вихід зі строю комп'ютерного обладнання	Дуже малий	Великий	0,13	Провести аналіз обладнання на працездатність	Ухилення	Замінити або відновити обладнання
5	Відкритий	Збій в роботі програмного забезпечення	Дуже малий	Середній	0,07	Мати аналогове програмне забезпечення	Зменшення	Перевстановити програмне забезпечення.

Продовження табл. А.4

ID	Статус ризику	Опис ризику	Ймовірність виникнення	Вплив ризику	Ранг ризику	План А	Тип стратегії реагування	План Б
6	Відкритий	Часте внесення змін у ТЗ	Дуже малий	Малий	0,09	Визначити точну мету та потреби проекту	Зменшення	Перевизначити технічне завдання
7	Відкритий	Низька кваліфікація розробника	Дуже малий	Великий	0,12	Знати стеки технологій які використовуються	Зменшення	Провести аналіз та визначити аналогові технології
8	Відкритий	Неоптимальний розподіл часу	Малий	Середній	0,13	Провести дослідження та розробити план виконання проекту	Ухилення	Перевизначити строки виконання етапів проекту
9	Відкритий	Поява альтернативного продукту	Дуже великий	Дуже малий	0,09	Провести дослідження на наявність аналогів та слідкувати за новинами	Передача	
10	Відкритий	Виявлення великої кількості помилок при тестуванні	Дуже великий	Малий	0,13		Ухилення	Провести аналіз та усунути поилки

Джерело: власна реалізація.

Результатом планування реагування стала матриця ймовірності виникнення та впливу ризиків, що зображена у таблиці А.5. Зеленим кольором позначено прийнятні, жовтим – виправдані, червоним – недопустимі.

Таблиця А.5 – Матриця ймовірностей

Ймовірність	Вплив загрози (ризик)				
	Дуже малий 0,05	Малий 0,15	Середній 0,3	Великий 0,5	Дуже великий 0,8
0,9	R9(0.09)	R10(0.13)			
0,7					
0,5					
0,3		R1(0.1)	R8(0.13)		
0,1		R6(0.09)	R2(0.05), R5(0.07)	R3(0.05), R4(0.13), R7(0.12)	

Джерело: власна реалізація.

Класифікація ризиків за рівнем, відповідно до отриманого значення індексу, представлена у таблиці А.6.

Таблиця А.6 – Класифікація ризиків

№	Назва	Межі	Ризик, які входять (номера)
1	Прийнятні	$0,005 \leq R \leq 0,05$	1, 2, 3, 4, 5, 6, 7, 9
2	Виправдані	$0,05 < R < 0,14$	8, 10
3	Недопустимі	$0,14 \leq R \leq 0,72$	-

Джерело: власна реалізація.

ДОДАТОК Б

ЛІСТИНГ ПРОГРАМНОГО КОДУ ОСНОВНИХ МОДУЛІВ WEB-ДОДАТКУ

Profile.js:

```

window.onload = function () {
    loadPageData();
    loadData();
}

async function exitInfo(){
    var sendBody =
"https://tolik.app/gc_modules/certificates/test/public_html/backend/exit_cookies.php
";
    var response = await fetch(sendBody);
    document.getElementById('file_exit').innerText = "Ви вийшли з акаунту";
    let success_message = document.getElementById("file_exit");
        success_message.style.display = "block";
        success_message.style.opacity = 100;
        setTimeout(function () {
            success_message.style.display = "none";
            success_message.style.opacity = 0;
        }, 5000);
    await new Promise(r => setTimeout(r, 1500));
    document.location.href = "../news/news_card.html";
}

async function loadPageData()
{
    var sendBody1 =
"https://tolik.app/gc_modules/certificates/test/public_html/backend/check_cookies.ph
p";
    var response1 = await fetch(sendBody1);
    var inputJson1 = await response1.text();

    if(inputJson1==="admin"){
        document.getElementsByClassName('header-menu-user-picture')[0].style.display
= 'none';
        document.getElementsByClassName('header-menu-user-picture2')[0].style.display
= 'block';
    }else if(inputJson1==="user"){
        document.getElementsByClassName('header-menu-user-picture')[0].style.display
= 'none';
        document.getElementsByClassName('header-menu-user-picture2')[0].style.display
= 'block';
    }else{
        document.getElementsByClassName('header-menu-user-picture2')[0].style.display
= 'none';
        document.getElementsByClassName('header-menu-user-picture')[0].style.display
= 'block';
    }
}

async function saveProfile() {

```

```

    var sendBodyLog =
"https://tolik.app/gc_modules/certificates/test/public_html/backend/analytics/get_log
_cookies.php";
    var responseLog = await fetch(sendBodyLog);
    var inputJsonLog = await responseLog.text();

    var name = document.getElementById('profile-input-name').value;
    var passport = document.getElementById('profile-input-passport').value;
    var phone = document.getElementById('profile-input-phone').value;
    var code = document.getElementById('profile-input-code').value;
    var codeName = document.getElementById('profile-input-codeName').value;
    var address = document.getElementById('profile-input-address').value;
    var authority = document.getElementById('profile-input-authority').value;

    var sendBody =
"https://tolik.app/gc_modules/certificates/test/public_html/backend/profile/set_profi
le_data.php?UserId="+inputJsonLog+"&Name="+name+"&Passport="+passport+"&Phone="+phone
+"&Code="+code+"&CodeName="+codeName+"&Address="+address+"&Authority="+authority;
    var response = await fetch(sendBody);
    var inputJson = await response.text();

    if(inputJson=="Ok"){
        document.getElementById('file_exit').innerText = "Успішно збережено!";
        let success_message = document.getElementById("file_exit");
        success_message.style.display = "block";
        success_message.style.opacity = 100;
        setTimeout(function () {
            success_message.style.display = "none";
            success_message.style.opacity = 0;
        }, 5000);
    }else{
        document.getElementById('file_exit').innerText = "Сталась помилка!";
        let success_message = document.getElementById("file_exit");
        success_message.style.display = "block";
        success_message.style.opacity = 100;
        setTimeout(function () {
            success_message.style.display = "none";
            success_message.style.opacity = 0;
        }, 5000);
    }
}

}

async function loadData() {

    var sendBodyLog =
"https://tolik.app/gc_modules/certificates/test/public_html/backend/analytics/get_log
_cookies.php";
    var responseLog = await fetch(sendBodyLog);
    var inputJsonLog = await responseLog.text();

    try {
        const url =
'https://tolik.app/gc_modules/certificates/test/public_html/backend/profile/get_profi
le_data.php?UserId='+inputJsonLog;
        const response = await fetch(url);

        if (response.ok) {
            const data = await response.json();
            document.getElementById('profile-input-name').value = data.user_name
|| '';

```

```

        document.getElementById('profile-input-passport').value =
data.user_passport || '';
        document.getElementById('profile-input-phone').value =
data.user_mobile || '';
        document.getElementById('profile-input-code').value = data.user_code
|| '';
        document.getElementById('profile-input-codeName').value =
data.user_code_name || '';
        document.getElementById('profile-input-address').value =
data.user_podatkova_address || '';
        document.getElementById('profile-input-authority').value =
data.user_controller || '';
    } else {
        console.error('Ошибка HTTP: ' + response.status);
    }
} catch (error) {
    console.error('Ошибка:', error.message);
}
}
}

```

set_profile_data.php:

```

<?php

require_once("db-model-BE.php");

if ($_GET['UserId']== "") {echo "Error: UserId is not set"; exit;};
if ($_GET['Name']== "") {echo "Error: Name is not set"; exit;};
if ($_GET['Passport']== "") {echo "Error: Passport is not set"; exit;};
if ($_GET['Phone']== "") {echo "Error: Phone is not set"; exit;};
if ($_GET['Code']== "") {echo "Error: Code is not set"; exit;};
if ($_GET['CodeName']== "") {echo "Error: CodeName is not set"; exit;};
if ($_GET['Address']== "") {echo "Error: Address is not set"; exit;};
if ($_GET['Authority']== "") {echo "Error: Authority is not set"; exit;};

$UserId = $_GET['UserId'];
$Name = str_replace("'", "", $_GET['Name']);
$Passport = $_GET['Passport'];
$Phone = $_GET['Phone'];
$Code = $_GET['Code'];
$CodeName = str_replace("'", "", $_GET['CodeName']);
$Address = str_replace("'", "", $_GET['Address']);
$Authority = str_replace("'", "", $_GET['Authority']);

$searchQuery1 = "SELECT * FROM `users_profile` WHERE `user_id` = '$UserId'";
$resultSearch1 = Query::getQuery($searchQuery1);

if ($resultSearch1 && mysqli_num_rows($resultSearch1) > 0) {
    $result = Query::getQuery("UPDATE `users_profile` SET
`user_name`=''$Name'', `user_passport`=''$Passport'', `user_mobile`=''$Phone'', `user_code`=''$
Code'', `user_code_name`=''$CodeName'',
`user_podatkova_address`=''$Address'', `user_controller`=''$Authority' WHERE
`user_id` = '$UserId'");
} else {
    $result = Query::getQuery("INSERT INTO `users_profile` (`id`, `user_id`,
`user_name`, `user_passport`, `user_mobile`,
`user_code`, `user_code_name`, `user_podatkova_address`, `user_controller`)

```

```

VALUES (NULL, '$UserId', '$Name', '$Passport', '$Phone', '$Code', '$CodeName',
'$Address', '$Authority');"");
}

$result = Query::getQuery($query);

$searchQuery = "SELECT * FROM `users_profile` WHERE `user_id` = '$UserId' AND
`user_name` = '$Name' AND `user_passport` = '$Passport' AND `user_mobile` = '$Phone'
AND `user_code` = '$Code' AND `user_code_name` = '$CodeName' AND
`user_podatkova_address` = '$Address' AND `user_controller` = '$Authority'";

$resultSearch = Query::getQuery($searchQuery);

if ($resultSearch && mysqli_num_rows($resultSearch) > 0) {
    echo "Ok";
} else {
    echo "Error";
}

?>

```

get_profile_data.php

```

<?php

require_once("db-model-BE.php");

if ($_GET['UserId'] == "") {
    echo json_encode(["error" => "UserId is not set"]);
    exit;
}

$UserId = $_GET['UserId'];

$result = Query::getQuery("SELECT `user_name`, `user_passport`, `user_mobile`,
`user_code`, `user_code_name`, `user_podatkova_address`, `user_controller` FROM
`users_profile` WHERE `user_id`='$UserId'");

if ($result) {
    $data = mysqli_fetch_assoc($result);
    echo json_encode($data);
} else {
    echo json_encode(["error" => "Error fetching profile data"]);
}

?>

```

load_profile_data.php:

```

<?php

require_once("db-model-BE.php");

if (!isset($_GET['UserId']) || empty($_GET['UserId'])) {
    echo json_encode(["error" => "UserId is not set"]);
    exit;
}

```

```

$userId = $_GET['UserId'];

$query = "SELECT * FROM `users_profile` WHERE `user_id` = '$userId'";
$result = Query::getQuery($query);

if ($result) {
    if (mysqli_num_rows($result) > 0) {
        $row = mysqli_fetch_assoc($result);
        echo json_encode($row);
    } else {
        echo json_encode(["error" => "No data found for UserId: $userId"]);
    }
} else {
    echo json_encode(["error" => "Error executing query: " . $query . "<br>" . $conn->error]);
}

?>

```

report.js:

```

window.onload = function () {
    loadPageData();
    loadData();
}

async function exitInfo(){
    var sendBody =
"https://tolik.app/gc_modules/certificates/test/public_html/backend/exit_cookies.php
";
    var response = await fetch(sendBody);
    let success_message = document.getElementById("file_exit");
    success_message.style.display = "block";
    success_message.style.opacity = 100;
    setTimeout(function () {
        success_message.style.display = "none";
        success_message.style.opacity = 0;
    }, 5000);
    await new Promise(r => setTimeout(r, 1500));
    document.location.href = "../news/news_card.html";
}

async function loadPageData()
{
    var sendBody1 =
"https://tolik.app/gc_modules/certificates/test/public_html/backend/check_cookies.ph
p";
    var response1 = await fetch(sendBody1);
    var inputJson1 = await response1.text();

    if(inputJson1==="admin"){
        document.getElementsByClassName('header-menu-user-picture')[0].style.display
= 'none';
        document.getElementsByClassName('header-menu-user-picture2')[0].style.display
= 'block';
    }else if(inputJson1==="user"){
        document.getElementsByClassName('header-menu-user-picture')[0].style.display
= 'none';

```

```

        document.getElementsByClassName('header-menu-user-picture2')[0].style.display
= 'block';
    }else{
        document.getElementsByClassName('header-menu-user-picture2')[0].style.display
= 'none';
        document.getElementsByClassName('header-menu-user-picture')[0].style.display
= 'block';
    }
}

```

```

function chooseReportType() {
    var block = document.getElementById("report-main-choice-list-child-data");
    var selectElement = document.getElementById("report-main-choice-list");
    var selectedIndex = selectElement.selectedIndex;
    var block3 = document.getElementById("report-main-choice-list-child-data-
quarter");

```

```

    if(selectedOptionIndex == "2"){
        block3.style.display = "none";
    }else{
        block3.style.display = "flex";
    }

```

```

    if (block.style.display == 'block') {
    } else {
        block.style.display = 'block';
    }
}

```

```

function isAnalyticsBlock(){
    var block = document.getElementById("report-main-choice-list-child-data-
analyticsList");
    var none = document.getElementById("report-main-choice-list-child-data-income");

    if (block.style.display == 'block') {
        none.style.display = "block";
        block.style.display = "none";
    } else {
        none.style.display = "none";
        block.style.display = "block";
    }
}

```

```

async function generateReport(){
    var sendBodyLog =
"https://tolik.app/gc_modules/certificates/test/public_html/backend/analytics/get_log
_cookies.php";
    var responseLog = await fetch(sendBodyLog);
    var inputJsonLog = await responseLog.text();

    try {
        const url =
'https://tolik.app/gc_modules/certificates/test/public_html/backend/profile/get_profi
le_data.php?UserId='+inputJsonLog;
        const response = await fetch(url);

        if (response.ok) {
            const data = await response.json();

```

```

        var user_name = data.user_name;
        var user_passport = data.user_passport;
        var user_mobile = data.user_mobile;
        var user_code = data.user_code;
        var user_code_name = data.user_code_name;
        var user_podatkova_address = data.user_podatkova_address;
        var user_controller = data.user_controller;
    } else {
        console.error('Ошибка HTTP: ' + response.status);
    }
} catch (error) {
    console.error('Ошибка:', error.message);
}

if(document.getElementById("report-main-choice-list-child-data-quarter").style.display == "none"){
    var quarter_num = 5;
    var quarter_sum = document.getElementById('report-main-choice-list-child-data-income-area').value;
} else {
    var quarter_num = document.getElementById('report-main-choice-list-child-data-quarter-area').value;
    var quarter_sum = document.getElementById('report-main-choice-list-child-data-income-area').value;
}

var sendBody =
"https://tolik.app/gc_modules/certificates/test/public_html/backend/report/create_report.php?UserId="+inputJsonLog+"&quarter_num="+quarter_num+"&quarter_sum="+
    quarter_sum+"&user_name="+user_name+"&user_passport="+user_passport+"&user_mobile="+user_mobile+"&user_code="+user_code+"&user_code_name="+user_code_name+"&user_podatkova_address="+
    user_podatkova_address+"&user_controller="+user_controller;
var response = await fetch(sendBody);

var today = new Date();
var day = today.getDate();
var month = today.getMonth() + 1;
var year = today.getFullYear();
day = (day < 10) ? "0" + day : day;
month = (month < 10) ? "0" + month : month;

var formattedDate = day + "." + month + "." + year;

var filename = formattedDate+"_"+user_name.replace(/ /g, '_')+"_"+quarter_sum;

var sendBody2 =
"https://tolik.app/gc_modules/certificates/test/public_html/backend/report/set_report_to_db.php?UserId="+inputJsonLog+"&filename="+filename;
var response2 = await fetch(sendBody2);
var inputJson2 = await response2.text();

if(inputJson2=="Ok"){
    document.getElementById('file_exit').innerText = "Звіт успішно створено!";
    let success_message = document.getElementById("file_exit");
    success_message.style.display = "block";
    success_message.style.opacity = 100;
    setTimeout(function () {
        success_message.style.display = "none";
        success_message.style.opacity = 0;
    }, 5000);
} else {

```



```

document.getElementById('file_exit').innerText = "Сталась помилка!";
let success_message = document.getElementById("file_exit");
success_message.style.display = "block";
success_message.style.opacity = 100;
setTimeout(function () {
    success_message.style.display = "none";
    success_message.style.opacity = 0;
}, 5000);
}
loadData();
}

async function loadData() {

    var sendBodyLog =
"https://tolik.app/gc_modules/certificates/test/public_html/backend/analytics/get_log
_cookies.php";
    var responseLog = await fetch(sendBodyLog);
    var inputJsonLog = await responseLog.text();

    try {
        const url =
'https://tolik.app/gc_modules/certificates/test/public_html/backend/report/load_repor
ts.php?UserId='+inputJsonLog;
        const response = await fetch(url);

        if (response.ok) {
            const data = await response.json();

            var reportList = document.getElementById("report-main-reports-list");

            reportList.innerHTML = "";

            for (var i = 0; i < data.length; i++) {
                reportList.innerHTML += ""
                + "<div class='report-main-reports-list-child'>"
                + "<div class='report-main-reports-list-child-header'
onclick='downloadReport(\"" + data[i].pdf_name + "\"" + ">"
                + data[i].pdf_name+".pdf"
                + "</div>"
                + "</div>";
            }
        } else {
            console.error('Ошибка HTTP: ' + response.status);
        }
    } catch (error) {
        console.error('Ошибка:', error.message);
    }
}

async function downloadReport(filename) {

    var downloadUrl =
"https://tolik.app/gc_modules/certificates/test/public_html/backend/report/download-
report.php";
    var fileUrl = downloadUrl + "?filename=" + filename;
    const response = await fetch(fileUrl);

    if (response.ok) {
        var blob = await response.blob();
        var link = document.createElement('a');
        link.href = window.URL.createObjectURL(blob);
    }
}

```

```

        link.download = filename+".pdf";

        document.body.appendChild(link);
        link.click();
        document.body.removeChild(link);
    } else {
        alert('Ошибка при скачивании файла');
    }
}

```

set_report_to_db.php:

```

<?php

require_once("db-model-BE.php");

if ($_GET['UserId']== "") {echo "Error: UserId is not set"; exit;};
if ($_GET['filename']== "") {echo "Error: filename is not set"; exit;};

$UserId = $_GET['UserId'];
$filename = $_GET['filename'];

$result = Query::getQuery("INSERT INTO `reports` (`id`, `user_id`, `pdf_name`,
`timestamp`) VALUES (NULL, '$UserId', '$filename', current_timestamp());");
$result = Query::getQuery($query);

$searchQuery = "SELECT `id` FROM `reports` WHERE `user_id`='$UserId' AND
`pdf_name`='$filename'";

$resultSearch = Query::getQuery($searchQuery);

if ($resultSearch && mysqli_num_rows($resultSearch) > 0) {
    echo "Ok";
} else {
    echo "Error";
}

?>

```

load_reports.php:

```

<?php

require_once("db-model-BE.php");

if ($_GET['UserId'] == "") {
    echo "Error: UserId is not set";
    exit;
}

$UserId = $_GET['UserId'];

$query = "SELECT `pdf_name` FROM `reports` WHERE `user_id`='$UserId'";
$result = Query::getQuery($query);

if ($result && mysqli_num_rows($result) > 0) {
    $rows = array();

```

```

    while ($row = mysqli_fetch_assoc($result)) {
        $rows[] = $row;
    }

    echo json_encode($rows);
} else {
    echo json_encode(["error" => "Error executing query: " . $query . "<br>" . $conn->error]);
}

?>

```

converte-png-to-pdf.php:

```

<?php
require_once('../../../TCPDF/TCPDF.php');

function converteToPDF($filename){

    $imageFile = 'report-results/report-result-png/'.$filename.'.png';
    list($image_width, $image_height) = getimagesize($imageFile);

    $pdf = new TCPDF_TCPDF(P, 'pt', array($image_width, $image_height), true, 'UTF-8',
false);

    $pdf->SetMargins(0, 0, 0, true);
    $pdf->AddPage();

    $pdf->Image($imageFile, 0, 0, $image_width, $image_height, '', '', '', false, 300,
'', false, false, 0, false, false, false);
    $pdf->
>Output($_SERVER['DOCUMENT_ROOT'].'/gc_modules/certificates/test/public_html/backend/
report/report-results/report-result-pdf/'.$filename.'.pdf', 'F');
}
?>

```

create_report.php:

```

<?php

header("Content-type: image/png");

include 'converte-png-to-pdf.php';
//include 'download-report.php';

if ($_GET['UserId']== "") {echo "ERROR: UserId is not set"; exit;}
if ($_GET['quarter_num']== "") {echo "ERROR: quarter_num is not set"; exit;}
if ($_GET['quarter_sum']== "") {echo "ERROR: quarter_sum is not set"; exit;}
if ($_GET['user_name']== "") {echo "ERROR: user_name is not set"; exit;}
if ($_GET['user_passport']== "") {echo "ERROR: user_passport is not set"; exit;}
if ($_GET['user_mobile']== "") {echo "ERROR: user_mobile is not set"; exit;}
if ($_GET['user_code']== "") {echo "ERROR: user_code is not set"; exit;}
if ($_GET['user_code_name']== "") {echo "ERROR: user_code_name is not set"; exit;}
if ($_GET['user_podatkova_address']== "") {echo "ERROR: user_podatkova_address is not
set"; exit;}

```

```

if ($_GET['user_controller']== "") {echo "ERROR: user_controller is not set"; exit;}

$UserId = $_GET['UserId'];
$quarter_num = $_GET['quarter_num'];
$quarter_sum = $_GET['quarter_sum'];
$user_name = $_GET['user_name'];
$user_passport = $_GET['user_passport'];
$user_mobile = $_GET['user_mobile'];
$user_code = $_GET['user_code'];
$user_code_name = $_GET['user_code_name'];
$user_podatkova_address = $_GET['user_podatkova_address'];
$user_controller = $_GET['user_controller'];
$tax_sum = round($quarter_sum*0.05, 2);
$currentYear = date("Y");
$currentDate = date("d.m.Y");
$filename = $currentDate."_".str_replace(' ', '_', $user_name)."_" . $quarter_sum;

$font = "fonts/arial.ttf";
$template = imagecreatefrompng("template/quarter_template.png");
$quarter_num_x = 0;

if($quarter_num == 1){
    $quarter_num_x = 263;
}else if($quarter_num == 2){
    $quarter_num_x = 523;
}else if($quarter_num == 3){
    $quarter_num_x = 763;
}else if($quarter_num == 4){
    $quarter_num_x = 1073;
}else{
    $quarter_num_x = 1073;
}

$quarter_sum_x = find_end_position_x($template, 8, 0, $font, $quarter_sum);
$tax_sum_x = find_end_position_x($template, 8, 0, $font, $tax_sum);

imagefttext($template, 8, 0, $quarter_num_x, 312, $black, $font, "X");
imagefttext($template, 8, 0, 1650, 312, $black, $font, $currentYear);
imagefttext($template, 8, 0, 80, 468, $black, $font, $user_controller);
imagefttext($template, 8, 0, 446, 520, $black, $font, $user_name);
imagefttext($template, 8, 0, 447, 588, $black, $font, $user_podatkova_address);
imagefttext($template, 8, 0, 1350, 661, $black, $font, $user_mobile);
imagefttext($template, 8, 0, 1401, 708, $black, $font, $user_passport);
imagefttext($template, 8, 0, 79, 961, $black, $font, $user_code);
imagefttext($template, 8, 0, 421, 961, $black, $font, $user_code_name);
imagefttext($template, 8, 0, $quarter_sum_x, 1474, $black, $font, $quarter_sum);
imagefttext($template, 8, 0, $quarter_sum_x, 1600, $black, $font, $quarter_sum);
imagefttext($template, 8, 0, $tax_sum_x, 1708, $black, $font, $tax_sum);
imagefttext($template, 8, 0, $tax_sum_x, 1744, $black, $font, $tax_sum);
imagefttext($template, 8, 0, $tax_sum_x, 1816, $black, $font, $tax_sum);
imagefttext($template, 8, 0, $tax_sum_x, 1888, $black, $font, $tax_sum);
imagefttext($template, 8, 0, 855, 2580, $black, $font, $currentDate);
imagefttext($template, 8, 0, 1278, 2777, $black, $font, $user_name);
$path = "report-results/report-result-png/" . $filename . ".png";
imagepng($template, $path);

imagedestroy($template);

convertToPDF($filename);

function find_end_position_x($image, $font_size, $angle, $font, $text){
    $bbox = imageftbbox($font_size, $angle, $font, $text);

```

```

    $width = abs($bbox[2] - $bbox[0]);
    $x = 1797 - $width;

    return $x;
}
?>

```

download-report.php:

```

<?php

if ($_GET['filename']== "") {echo "ERROR: filename is not set"; exit;}
$filename = $_GET['filename'];
downloadPDF($filename);

function downloadPDF($filename){
    $pdf_file = 'report-results/report-result-pdf/'.$filename.'.pdf';
    $pdf_name = $filename.".pdf";
    header('Content-Type: application/pdf');
    header('Content-Disposition: attachment; filename="'.$pdf_name.'");
    readfile($pdf_file);
}
?>

```

analytics.js:

```

window.onload = function () {
    loadPageData();
    loadData();
    loadAnalysisMonth();
    loadAnalysisYear();
    loadAnalysisTrend();
}

async function exitInfo(){
    var sendBody =
"https://tolik.app/gc_modules/certificates/test/public_html/backend/exit_cookies.php
";
    var response = await fetch(sendBody);
    let success_message = document.getElementById("file_exit");
        success_message.style.display = "block";
        success_message.style.opacity = 100;
        setTimeout(function () {
            success_message.style.display = "none";
            success_message.style.opacity = 0;
        }, 5000);
    await new Promise(r => setTimeout(r, 1500));
    document.location.href = "../news/news_card.html";
}

async function loadAnalysisMonth(){
    var sendBodyLog =
"https://tolik.app/gc_modules/certificates/test/public_html/backend/analytics/get_log
_cookies.php";

```

```

var responseLog = await fetch(sendBodyLog);
var inputJsonLog = await responseLog.text();

var sendBody =
"https://tolik.app/gc_modules/certificates/test/public_html/backend/analytics/load_mo
nth_analytics.php?UserId="+inputJsonLog;

var response = await fetch(sendBody);

if (response.ok) {
  try {
    var jsonResponse = await response.json();

    var latestMonth = jsonResponse.latestMonth;
    var latestYear = jsonResponse.latestYear;
    var percentageDifference = jsonResponse.percentageDifference;

    var displayString = `${latestMonth}.${latestYear} у порівнянні з
попереднім місяцем прибуток`;

    if (percentageDifference > 0) {
      displayString += ` зріс на ${percentageDifference}%`;
    } else if (percentageDifference < 0) {
      displayString += ` зменшився на ${Math.abs(percentageDifference)}%`;
    } else {
      displayString += ` залишився незмінним`;
    }

    document.getElementById('analytics-body-analysis-month-text').innerText =
displayString;
  } catch (error) {
    document.getElementById('analytics-body-analysis-month-text').innerText =
"Недостатньо даних для аналізу.";
  }
} else {
  console.error("Помилка отримання даних");
}
}

async function loadAnalysisYear(){
  var sendBodyLog =
"https://tolik.app/gc_modules/certificates/test/public_html/backend/analytics/get_log
_cookies.php";
  var responseLog = await fetch(sendBodyLog);
  var inputJsonLog = await responseLog.text();

  var sendBody =
"https://tolik.app/gc_modules/certificates/test/public_html/backend/analytics/load_ye
ar_analytics.php?UserId=" + inputJsonLog;
  var sendBody2 =
"https://tolik.app/gc_modules/certificates/test/public_html/backend/analytics/load_2y
ears_analytics.php?UserId=" + inputJsonLog;

  var response = await fetch(sendBody);
  var response2 = await fetch(sendBody2);

  var key1 = true;
  var key2 = true;

  if (response.ok && response2.ok) {
    try {
      var jsonResponse = await response.json();

```

```

var year = jsonResponse.year;
var percentageDifferenceYear = jsonResponse.percentageDifferenceYear;

var displayString = `За ${year} рік прибуток`;

if (percentageDifferenceYear > 0) {
  displayString += ` зріс на ${percentageDifferenceYear}%`;
} else if (percentageDifferenceYear < 0) {
  displayString += ` зменшився на
${Math.abs(percentageDifferenceYear)}%`;
} else {
  displayString += ` залишився незмінним`;
}
} catch (error) {
  key1 = false;
}

try {
  var jsonResponse2 = await response2.json();

  var year1 = jsonResponse2.currentYear;
  var year2 = jsonResponse2.previousYear;
  var percentageDifferenceYear2 = jsonResponse2.percentageDifferenceYear;

  var displayString2 = `У ${year1} році відносно ${year2} року прибуток`;

  if (percentageDifferenceYear2 > 0) {
    displayString2 += ` зріс на ${percentageDifferenceYear2}%`;
  } else if (percentageDifferenceYear2 < 0) {
    displayString2 += ` зменшився на
${Math.abs(percentageDifferenceYear2)}%`;
  } else {
    displayString2 += ` залишився незмінним`;
  }
} catch (error) {
  key2 = false;
}

if(key1 && key2){

  document.getElementById('analytics-body-analysis-year-text').innerText =
displayString;
  document.getElementById('analytics-body-analysis-year-text').innerText +=
"\n"+displayString2;
}else if(key1){
  document.getElementById('analytics-body-analysis-year-text').innerText =
displayString;
}else if(key2){
  document.getElementById('analytics-body-analysis-year-text').innerText =
displayString2;
}else{
  document.getElementById('analytics-body-analysis-year-text').innerText =
"Недостатньо даних";
}

} else {
  console.error("Помилка отримання даних за рік");
}
}

```

```

async function loadAnalysisTrend(){
    var sendBodyLog =
    "https://tolik.app/gc_modules/certificates/test/public_html/backend/analytics/get_log
_cookies.php";
    var responseLog = await fetch(sendBodyLog);
    var inputJsonLog = await responseLog.text();

    var sendBody =
    "https://tolik.app/gc_modules/certificates/test/public_html/backend/analytics/load_tr
end_analytics.php?UserId=" + inputJsonLog;
    var response = await fetch(sendBody);

    if (response.ok) {
        try {
            var jsonResponse = await response.json();

            var maxIncreaseStreak = jsonResponse.maxIncreaseStreak;
            var startMonthIncrease = jsonResponse.startMonthIncrease;
            var startYearIncrease = jsonResponse.startYearIncrease;
            var endMonthIncrease = jsonResponse.endMonthIncrease;
            var endYearIncrease = jsonResponse.endYearIncrease;
            var percentageIncomeIncrease = jsonResponse.percentageIncomeIncrease;

            var maxDecreaseStreak = jsonResponse.maxDecreaseStreak;
            var startMonthDecrease = jsonResponse.startMonthDecrease;
            var startYearDecrease = jsonResponse.startYearDecrease;
            var endMonthDecrease = jsonResponse.endMonthDecrease;
            var endYearDecrease = jsonResponse.endYearDecrease;
            var percentageIncomeDecrease = jsonResponse.percentageIncomeDecrease;

            var displayString = "";

            displayString += `Максимальна тривалість росту: ${maxIncreaseStreak}
місяців (з ${startMonthIncrease}.${startYearIncrease} по
${endMonthIncrease}.${endYearIncrease}). Прибуток зріс на
${percentageIncomeIncrease}%. \n`;
            displayString += `Максимальна тривалість падіння: ${maxDecreaseStreak}
місяців (з ${startMonthDecrease}.${startYearDecrease} по
${endMonthDecrease}.${endYearDecrease}). Прибуток впав на
${percentageIncomeDecrease}%. \n`;

            document.getElementById('analytics-body-analysis-trend-text').innerText =
displayString;
        } catch (error) {
            document.getElementById('analytics-body-analysis-trend-text').innerText =
"Недостатньо даних для аналізу.";
        }
        } else {
            console.error("Помилка отримання даних за рік");
        }
    }
}

async function loadPageData()
{
    var sendBody1 =
    "https://tolik.app/gc_modules/certificates/test/public_html/backend/check_cookies.ph
p";
    var response1 = await fetch(sendBody1);

```



```

var inputJson1 = await response1.text();

if(inputJson1==="admin"){
    document.getElementsByClassName('header-menu-user-picture')[0].style.display
= 'none';
    document.getElementsByClassName('header-menu-user-picture2')[0].style.display
= 'block';
}else if(inputJson1==="user"){
    document.getElementsByClassName('header-menu-user-picture')[0].style.display
= 'none';
    document.getElementsByClassName('header-menu-user-picture2')[0].style.display
= 'block';
}else{
    document.getElementsByClassName('header-menu-user-picture2')[0].style.display
= 'none';
    document.getElementsByClassName('header-menu-user-picture')[0].style.display
= 'block';
}
}

async function changeCurrency()
{
    var Currency = document.getElementById('iafCurrency').value;

    var sendBodyLog =
"https://tolik.app/gc_modules/certificates/test/public_html/backend/analytics/get_log
_cookies.php";
    var responseLog = await fetch(sendBodyLog);
    var inputJsonLog = await responseLog.text();

    if (Currency=="") {
        document.getElementById('file_exit').innerText = "Оберіть валюту в якій
будете вести прибуток!";
        let success_message = document.getElementById("file_exit");
        success_message.style.display = "block";
        success_message.style.opacity = 100;
        setTimeout(function () {
            success_message.style.display = "none";
            success_message.style.opacity = 0;
        }, 5000);
        return;
    };

    document.getElementById("currencyChangeForm").style.display="none";
    document.getElementById('incomePanel').innerHTML = "<img
src='https://tolik.app/images/custom_img/loading-25.gif' class='loading-img'>";
    document.getElementById('incomeChart').innerHTML = "<img
src='https://tolik.app/images/custom_img/loading-25.gif' class='loading-img'>";
    document.getElementById('incomeCurrencyValue').innerHTML = "<img
src='https://tolik.app/images/custom_img/loading-25.gif' style='height:20px; margin-
top:3px; margin-bottom:3px;'>";

    var sendBody =
"https://tolik.app/gc_modules/certificates/test/public_html/backend/analytics/set_cur
rency.php?UserId="+inputJsonLog+"&IncomeCurrency="+Currency;

    var response = await fetch(sendBody);
    if (response.ok) {
        loadData();
    }
}

```

```

    } else {
        document.getElementById('file_exit').innerText = "Помилка HTTP: " +
response.status;
        let success_message = document.getElementById("file_exit");
        success_message.style.display = "block";
        success_message.style.opacity = 100;
        setTimeout(function () {
            success_message.style.display = "none";
            success_message.style.opacity = 0;
        }, 5000);
    }
}

async function addIncomeItem()
{
    var sendBodyLog =
"https://tolik.app/gc_modules/certificates/test/public_html/backend/analytics/get_log
_cookies.php";
    var responseLog = await fetch(sendBodyLog);
    var inputJsonLog = await responseLog.text();

    if(document.getElementById('incomeCurrencyValue').innerHTML == "¤ - У.Е. - Умовна
одиниця" || document.getElementById('incomeCurrencyValue').innerHTML == "Не
выбрана"){

        document.getElementById("incomeAddForm").style.display="none";

        var progress = document.getElementById("error_link_2");
        progress.style.display = "block";
        setTimeout(function() {
            var progress = document.getElementById("error_link_2");
            progress.style.display = "none";}, 5000);
    }else{

        var Income = document.getElementById('iafIncome').value;
        var Month = document.getElementById('iafMonth').value;
        var Year = document.getElementById('iafYear').value;

        if (Income=="") {
            document.getElementById('file_exit').innerText = "Будь-ласка, введіть
значення прибутку за період.";
            let success_message = document.getElementById("file_exit");
            success_message.style.display = "block";
            success_message.style.opacity = 100;
            setTimeout(function () {
                success_message.style.display = "none";
                success_message.style.opacity = 0;
            }, 5000);
            return;
        };
        if (Month=="||Month=="Месяц") {
            document.getElementById('file_exit').innerText = "Будь-ласка, введіть
місяць періоду за котрий ви вносите прибуток.";
            let success_message = document.getElementById("file_exit");
            success_message.style.display = "block";
            success_message.style.opacity = 100;
            setTimeout(function () {
                success_message.style.display = "none";
                success_message.style.opacity = 0;
            }, 5000);
            return;
        };
    }
};

```

```

    if (Year=="||Year=="Год") {
        document.getElementById('file_exit').innerText = "Будь-ласка, введіть рік
періоду за котрий ви вносите прибуток.";
        let success_message = document.getElementById("file_exit");
        success_message.style.display = "block";
        success_message.style.opacity = 100;
        setTimeout(function () {
            success_message.style.display = "none";
            success_message.style.opacity = 0;
        }, 5000);
        return;
    };

    document.getElementById("incomeAddForm").style.display="none";
    document.getElementById('incomePanel').innerHTML = "<img
src='https://tolik.app/images/custom_img/loading-25.gif' class='loading-img'>";
    document.getElementById('incomeChart').innerHTML = "<img
src='https://tolik.app/images/custom_img/loading-25.gif' class='loading-img'>";
    document.getElementById('incomeCurrencyValue').innerHTML = "<img
src='https://tolik.app/images/custom_img/loading-25.gif' style='height:20px; margin-
top:3px; margin-bottom:3px;'>";
    var issetSame = 0;

    inputArr.forEach(
        function(curVal, Index) {
            if (curVal['month']==Month&&curVal['year']==Year)
            {
                inputArr[Index]['income']=Income;
                issetSame = 1;
            }
        }
    )

    if (issetSame==0)
    {
        var tempArr = {income:Income, month:Month, year:Year};
        inputArr.push(tempArr);
        inputArr.sort(function(a,b){
            if (a.year == b.year) return a.month.localeCompare(b.month);
            return a.year-b.year;
        });
        inputArr.reverse();
    }

    var sendBody =
"https://tolik.app/gc_modules/certificates/test/public_html/backend/analytics/set_mon
th_income.php?UserId="+inputJsonLog+"&IncomeMonth="+JSON.stringify(inputArr);

    var response = await fetch(sendBody);

    if (response.ok) {
        loadData();
    } else {

        document.getElementById('file_exit').innerText = "Помилка HTTP: " +
response.status;
        let success_message = document.getElementById("file_exit");
        success_message.style.display = "block";
        success_message.style.opacity = 100;
        setTimeout(function () {
            success_message.style.display = "none";
            success_message.style.opacity = 0;

```

```

        }, 5000);
    }
}

loadAnalysisMonth();
loadAnalysisYear();
loadAnalysisTrend();
}

function deterIncome(id){

    document.getElementsByClassName('IPI-value-value')[id].innerHTML =
document.getElementsByClassName('IPI-value-
value')[id].innerHTML.replace(/(\d)(?=(\d\d\d)+([\^d]|$))/g, '$1 ');
}

async function deleteIncomeItem(id){

    var sendBodyLog =
"https://tolik.app/gc_modules/certificates/test/public_html/backend/analytics/get_log
_cookies.php";
    var responseLog = await fetch(sendBodyLog);
    var inputJsonLog = await responseLog.text();

    document.getElementById('incomePanel').innerHTML = "<img
src='https://tolik.app/images/custom_img/loading-25.gif' class='loading-img'>";
    document.getElementById('incomeChart').innerHTML = "<img
src='https://tolik.app/images/custom_img/loading-25.gif' class='loading-img'>";
    document.getElementById('incomeCurrencyValue').innerHTML = "<img
src='https://tolik.app/images/custom_img/loading-25.gif' style='height:20px; margin-
top:3px; margin-bottom:3px;'>";

    var sendBody =
"https://tolik.app/gc_modules/certificates/test/public_html/backend/analytics/delete_
month_income.php?UserId="+inputJsonLog+"&Month="+inputArr[id]['month']+&Year="+input
Arr[id]['year'];

    var response = await fetch(sendBody);
    if (response.ok) {
        loadData();
    } else {
        document.getElementById('file_exit').innerText = "Помилка HTTP: " +
response.status;
        let success_message = document.getElementById("file_exit");
        success_message.style.display = "block";
        success_message.style.opacity = 100;
        setTimeout(function () {
            success_message.style.display = "none";
            success_message.style.opacity = 0;
        }, 5000);
    }

    loadAnalysisMonth();
    loadAnalysisYear();
    loadAnalysisTrend();
}

async function loadData(){

    var sendBodyLog =
"https://tolik.app/gc_modules/certificates/test/public_html/backend/analytics/get_log
_cookies.php";

```

```

var responseLog = await fetch(sendBodyLog);
var inputJsonLog = await responseLog.text();

document.getElementById('incomePanel').innerHTML = "<img
src='https://tolik.app/images/custom_img/loading-25.gif' class='loading-img'>";
document.getElementById('incomeChart').innerHTML = "<img
src='https://tolik.app/images/custom_img/loading-25.gif' class='loading-img'>";
document.getElementById('incomeCurrencyValue').innerHTML = "<img
src='https://tolik.app/images/custom_img/loading-25.gif' style='height:20px; margin-
top:3px; margin-bottom:3px;'>";

var sendBody2 =
"https://tolik.app/gc_modules/certificates/test/public_html/backend/analytics/load_cu
rrency.php?UserId="+inputJsonLog;

var response2 = await fetch(sendBody2);

if (response2.ok) {
  var inputJson2 = await response2.text();

  if (inputJson2=="") {
    document.getElementById('incomeCurrencyValue').innerHTML = "Не выбрана";
  }
  else{
    document.getElementById('incomeCurrencyValue').innerHTML = inputJson2;
  }

  var echoCurrency = "?";

  if (inputJson2=="₴ - UAH - Гривня") echoCurrency = "₴";
  if (inputJson2=="₸ - KZT - Тенге") echoCurrency = "₸";

  if (inputJson2=="$ - USD - Долар США") echoCurrency = "$";
  if (inputJson2=="€ - EUR - Євро") echoCurrency = "€";
  if (inputJson2=="£ - GBP - Фунт стерлінгів") echoCurrency = "£";
  if (inputJson2=="¥ - JPY - Єна") echoCurrency = "¥";

  if (inputJson2=="¥ - CNY - Юань") echoCurrency = "¥";
  if (inputJson2=="₺ - TRY - Турецька ліра") echoCurrency = "₺";
  if (inputJson2=="₼ - AZN - Манат") echoCurrency = "₼";
  if (inputJson2=="ლ - GEL - Лари") echoCurrency = "ლ";

  if (inputJson2=="leu - MDL - Молдавський лей") echoCurrency = "leu";
  if (inputJson2=="﷮ - AED - Дирхам ОАЕ") echoCurrency = "﷮";
  if (inputJson2=="₹ - У.Е. - Умовна одиниця") echoCurrency = "₹";

} else {
  document.getElementById('file_exit').innerText = "Помилка HTTP: " +
response.status;
  let success_message = document.getElementById("file_exit");
  success_message.style.display = "block";
  success_message.style.opacity = 100;
  setTimeout(function () {
    success_message.style.display = "none";
    success_message.style.opacity = 0;
  }, 5000);
}

var sendBody =
"https://tolik.app/gc_modules/certificates/test/public_html/backend/analytics/load_mo
nth_income.php?UserId="+inputJsonLog;

```

```

var response = await fetch(sendBody);
if (response.ok) {

    var inputJson = await response.text();
    if (inputJson==" " || inputJson=="[]")
    {
        document.getElementById('incomePanel').innerHTML = "Тут буде список
вашого прибутку. Почніть вносити дані за допомогою кнопки\"+\\" у нижній часті цього
блоку";
        document.getElementById('incomeChart').innerHTML = "Тут буде графік
вашого прибутку. Почніть вносити дані за допомогою кнопки\"+\\" в меню справа";
        inputArr=[];
        return;
    }
    inputArr = JSON.parse(inputJson);
    document.getElementById('incomePanel').innerHTML = "";

    inputArr.sort(function(a,b){
    if (a.year == b.year) return a.month.localeCompare(b.month);
    return a.year-b.year;
});
    inputArr.reverse();

    inputArr.forEach(
    function(curVal, Index) {
        document.getElementById('incomePanel').innerHTML += "
+<div class='income-panel-item'>
+<div class='IPI-value-box'>
+<div class='IPI-value-box-sub'>
+<div class='IPI-value-value'>
+curVal['income']
+</div>
+<div class='IPI-value-currency'>
+ " "+echoCurrency
+</div>
+</div>
+<div class='IPI-date-box'>
+<div class='IPI-date-month'>
+curVal['month']
+</div>
+<div class='IPI-date-year'>
+curVal['year']
+</div>
+</div>
+<div class='IPI-delete' onclick='deleteIncomeItem("+ Index +"')>
+<img src='https://tolik.app/images/custom_img/krest2_red.png'>
+</div>
+</div>
";
    };

    deterIncome(Index);
    }
)

} else {
//alert("Помилка HTTP: " + response.status);
document.getElementById('file_exit').innerText = "Помилка HTTP: " +
response.status;
let success_message = document.getElementById("file_exit");
success_message.style.display = "block";
success_message.style.opacity = 100;

```

```

        setTimeout(function () {
            success_message.style.display = "none";
            success_message.style.opacity = 0;
        }, 5000);
    }

    google.charts.load('current', {'packages':['corechart']});
    google.charts.setOnLoadCallback(drawChart);

    var chartArr_2018 = [];
    var chartArr_2019 = [];
    var chartArr_2020 = [];
    var chartArr_2021 = [];
    var chartArr_2022 = [];
    var chartArr_2023 = [];

    var check_2018;
    var check_2019;
    var check_2020;
    var check_2021;
    var check_2022;
    var check_2023;
    inputArr.forEach(
        function(curVal, Index) {
            if(curVal['year'] == "2018"){
                chartArr_2018[Number(curVal['month'])] =
(Number(curVal['income']));
                check_2018 = true;
            }
            else if(curVal['year'] == "2019"){
                chartArr_2019[Number(curVal['month'])] =
(Number(curVal['income']));
                check_2019 = true;
            }
            else if(curVal['year'] == "2020"){
                chartArr_2020[Number(curVal['month'])] =
(Number(curVal['income']));
                check_2020 = true;
            }
            else if(curVal['year'] == "2021"){
                chartArr_2021[Number(curVal['month'])] =
(Number(curVal['income']));
                check_2021 = true;
            }
            else if(curVal['year'] == "2022"){
                chartArr_2022[Number(curVal['month'])] =
(Number(curVal['income']));
                check_2022 = true;
            }
            else if(curVal['year'] == "2023"){
                chartArr_2023[Number(curVal['month'])] =
(Number(curVal['income']));
                check_2023 = true;
            }
        }
    )

    var title_2018 = '2018';
    var title_2019 = '2019';
    var title_2020 = '2020';
    var title_2021 = '2021';

```

```

var title_2022 = '2022';
var title_2023 = '2023';
if(check_2018 != true){
  if(check_2022 == true){
    for (let i = 1; i < 12; i++) {
      if(chartArr_2022[i]!=null){
        chartArr_2018[i]=chartArr_2022[i];
        break;
      }
    }
  }
  if(check_2021 == true){
    for (let i = 1; i < 12; i++) {
      if(chartArr_2021[i]!=null){
        chartArr_2018[i]=chartArr_2021[i];
        break;
      }
    }
  }
  else if(check_2020 == true){
    for (let i = 1; i < 12; i++) {
      if(chartArr_2020[i]!=null){
        chartArr_2018[i]=chartArr_2020[i];
        break;
      }
    }
  }
  else if(check_2019 == true){
    for (let i = 1; i < 12; i++) {
      if(chartArr_2019[i]!=null){
        chartArr_2018[i]=chartArr_2019[i];
        break;
      }
    }
  }
  else if(check_2023 == true){
    for (let i = 1; i < 12; i++) {
      if(chartArr_2023[i]!=null){
        chartArr_2018[i]=chartArr_2023[i];
        break;
      }
    }
  }
  else{
    chartArr_2018[1]=0;
  }

  title_2018 = '';
}
if(check_2019 != true){
  if(check_2022 == true){
    for (let i = 1; i < 12; i++) {
      if(chartArr_2022[i]!=null){
        chartArr_2019[i]=chartArr_2022[i];
        break;
      }
    }
  }
  if(check_2021 == true){
    for (let i = 1; i < 12; i++) {
      if(chartArr_2021[i]!=null){
        chartArr_2019[i]=chartArr_2021[i];

```



```

        break;
    }
}
else if(check_2020 == true){
    for (let i = 1; i < 12; i++) {
        if(chartArr_2020[i]!=null){
            chartArr_2019[i]=chartArr_2020[i];
            break;
        }
    }
}
else if(check_2018 == true){
    for (let i = 1; i < 12; i++) {
        if(chartArr_2018[i]!=null){
            chartArr_2019[i]=chartArr_2018[i];
            break;
        }
    }
}

else if(check_2023 == true){
    for (let i = 1; i < 12; i++) {
        if(chartArr_2023[i]!=null){
            chartArr_2019[i]=chartArr_2023[i];
            break;
        }
    }
}
else{
    chartArr_2019[1]=0;
}

title_2019 = '';
}
if(check_2020 != true){
    if(check_2022 == true){
        for (let i = 1; i < 12; i++) {
            if(chartArr_2022[i]!=null){
                chartArr_2020[i]=chartArr_2022[i];
                break;
            }
        }
    }
    if(check_2021 == true){
        for (let i = 1; i < 12; i++) {
            if(chartArr_2021[i]!=null){
                chartArr_2020[i]=chartArr_2021[i];
                break;
            }
        }
    }
    else if(check_2019 == true){
        for (let i = 1; i < 12; i++) {
            if(chartArr_2019[i]!=null){
                chartArr_2020[i]=chartArr_2019[i];
                break;
            }
        }
    }
    else if(check_2018 == true){
        for (let i = 1; i < 12; i++) {

```

```

        if(chartArr_2018[i]!=null){
            chartArr_2020[i]=chartArr_2018[i];
            break;
        }
    }
}

else if(check_2023 == true){
    for (let i = 1; i < 12; i++) {
        if(chartArr_2023[i]!=null){
            chartArr_2020[i]=chartArr_2023[i];
            break;
        }
    }
}
else{
    chartArr_2020[1]=0;
}

title_2020 = '';
}
if(check_2021 != true){
    if(check_2022 == true){
        for (let i = 1; i < 12; i++) {
            if(chartArr_2022[i]!=null){
                chartArr_2021[i]=chartArr_2022[i];
                break;
            }
        }
    }
    if(check_2018 == true){
        for (let i = 1; i < 12; i++) {
            if(chartArr_2018[i]!=null){
                chartArr_2021[i]=chartArr_2018[i];
                break;
            }
        }
    }
    else if(check_2019 == true){
        for (let i = 1; i < 12; i++) {
            if(chartArr_2019[i]!=null){
                chartArr_2021[i]=chartArr_2019[i];
                break;
            }
        }
    }
    else if(check_2020 == true){
        for (let i = 1; i < 12; i++) {
            if(chartArr_2020[i]!=null){
                chartArr_2021[i]=chartArr_2020[i];
                break;
            }
        }
    }
    else if(check_2023 == true){
        for (let i = 1; i < 12; i++) {
            if(chartArr_2023[i]!=null){
                chartArr_2021[i]=chartArr_2023[i];
                break;
            }
        }
    }
}
}

```

```

else{
    chartArr_2021[1]=0;
}

title_2021 = '';
}
if(check_2022 != true){
    if(check_2021 == true){
        for (let i = 1; i < 12; i++) {
            if(chartArr_2021[i]!=null){
                chartArr_2022[i]=chartArr_2021[i];
                break;
            }
        }
    }
    else if(check_2018 == true){
        for (let i = 1; i < 12; i++) {
            if(chartArr_2018[i]!=null){
                chartArr_2022[i]=chartArr_2018[i];
                break;
            }
        }
    }
    else if(check_2019 == true){
        for (let i = 1; i < 12; i++) {
            if(chartArr_2019[i]!=null){
                chartArr_2022[i]=chartArr_2019[i];
                break;
            }
        }
    }
    else if(check_2020 == true){
        for (let i = 1; i < 12; i++) {
            if(chartArr_2020[i]!=null){
                chartArr_2022[i]=chartArr_2020[i];
                break;
            }
        }
    }
    else if(check_2023 == true){
        for (let i = 1; i < 12; i++) {
            if(chartArr_2023[i]!=null){
                chartArr_2022[i]=chartArr_2023[i];
                break;
            }
        }
    }
    else{
        chartArr_2022[1]=0;
    }

    title_2022 = '';
}

if(check_2023 != true){
    if(check_2021 == true){
        for (let i = 1; i < 12; i++) {
            if(chartArr_2021[i]!=null){
                chartArr_2023[i]=chartArr_2021[i];
                break;
            }
        }
    }
}

```

```

    }
    else if(check_2018 == true){
        for (let i = 1; i < 12; i++) {
            if(chartArr_2018[i]!=null){
                chartArr_2023[i]=chartArr_2018[i];
                break;
            }
        }
    }
    else if(check_2019 == true){
        for (let i = 1; i < 12; i++) {
            if(chartArr_2019[i]!=null){
                chartArr_2023[i]=chartArr_2019[i];
                break;
            }
        }
    }
    else if(check_2020 == true){
        for (let i = 1; i < 12; i++) {
            if(chartArr_2020[i]!=null){
                chartArr_2023[i]=chartArr_2020[i];
                break;
            }
        }
    }
    else if(check_2022 == true){
        for (let i = 1; i < 12; i++) {
            if(chartArr_2022[i]!=null){
                chartArr_2023[i]=chartArr_2022[i];
                break;
            }
        }
    }
    else{
        chartArr_2023[1]=0;
    }

    title_2023 = '';
}

function drawChart() {
    var data = google.visualization.arrayToDataTable([
        ['Месяц', title_2018, title_2019, title_2020, title_2021, title_2022,
title_2023],
        ['01', chartArr_2018[1], chartArr_2019[1], chartArr_2020[1],
chartArr_2021[1], chartArr_2022[1], chartArr_2023[1] ],
        ['02', chartArr_2018[2], chartArr_2019[2], chartArr_2020[2],
chartArr_2021[2], chartArr_2022[2], chartArr_2023[2] ],
        ['03', chartArr_2018[3], chartArr_2019[3], chartArr_2020[3],
chartArr_2021[3], chartArr_2022[3], chartArr_2023[3] ],
        ['04', chartArr_2018[4], chartArr_2019[4], chartArr_2020[4],
chartArr_2021[4], chartArr_2022[4], chartArr_2023[4] ],
        ['05', chartArr_2018[5], chartArr_2019[5], chartArr_2020[5],
chartArr_2021[5], chartArr_2022[5], chartArr_2023[5] ],
        ['06', chartArr_2018[6], chartArr_2019[6], chartArr_2020[6],
chartArr_2021[6], chartArr_2022[6], chartArr_2023[6] ],
        ['07', chartArr_2018[7], chartArr_2019[7], chartArr_2020[7],
chartArr_2021[7], chartArr_2022[7], chartArr_2023[7] ],
        ['08', chartArr_2018[8], chartArr_2019[8], chartArr_2020[8],
chartArr_2021[8], chartArr_2022[8], chartArr_2023[8] ],

```

```

        ['09', chartArr_2018[9], chartArr_2019[9], chartArr_2020[9],
chartArr_2021[9], chartArr_2022[9], chartArr_2023[9] ],
        ['10', chartArr_2018[10], chartArr_2019[10], chartArr_2020[10],
chartArr_2021[10], chartArr_2022[10], chartArr_2023[10] ],
        ['11', chartArr_2018[11], chartArr_2019[11], chartArr_2020[11],
chartArr_2021[11], chartArr_2022[11], chartArr_2023[11] ],
        ['12', chartArr_2018[12], chartArr_2019[12], chartArr_2020[12],
chartArr_2021[12], chartArr_2022[12], chartArr_2023[12] ]
    ]);

    var options = {
        title: 'Графік прибутку:',
        curveType: 'function',
        legend: { position: 'bottom' },
        pointSize:10,
        series: {
            0: { color: '#FFDF36', pointShape: 'circle' },
        },
        lineWidth: 5,
    };

    var chart = new
google.visualization.LineChart(document.getElementById('incomeChart'));

    chart.draw(data, options);
}
}

```

delete_month_income.php:

```

<?php

if ($_GET['UserId']== "") {echo "Error: UserId is not set"; exit;};
if ($_GET['Month']== "") {echo "Error: Month is not set"; exit;};
if ($_GET['Year']== "") {echo "Error: Year is not set"; exit;};

require_once ("db-model-BE.php");

$UserId = $_GET['UserId'];
$Month = $_GET['Month'];
$Year = $_GET['Year'];

$result = Query::getQuery("DELETE FROM `IncomeMonth` WHERE `UserId`='$UserId' AND
`Month`='$Month' AND `Year`='$Year'");

?>

```

get_log_cookies.php:

```

<?php

if (isset($_COOKIE["log"])) $log = $_COOKIE["log"];
echo $log;

?>

```

load_2years_analytics.php:

```

<?php
if ($_GET['UserId'] == "") {
    echo "Error: UserId is not set";
    exit;
}

require_once("db-model-BE.php");

$UserId = $_GET['UserId'];

$resultCurrentYear = Query::getQuery("SELECT SUM(`Value`) as totalIncome FROM
`IncomeMonth` WHERE `UserId` = '$UserId' AND `Year` = YEAR(CURDATE())");
$resultPreviousYear = Query::getQuery("SELECT SUM(`Value`) as totalIncome FROM
`IncomeMonth` WHERE `UserId` = '$UserId' AND `Year` = YEAR(CURDATE()) - 1");

if ($resultCurrentYear && $resultPreviousYear && mysqli_num_rows($resultCurrentYear)
> 0 && mysqli_num_rows($resultPreviousYear) > 0) {
    $rowCurrentYear = mysqli_fetch_assoc($resultCurrentYear);
    $rowPreviousYear = mysqli_fetch_assoc($resultPreviousYear);

    $currentYearIncome = $rowCurrentYear['totalIncome'];
    $previousYearIncome = $rowPreviousYear['totalIncome'];

    if($previousYearIncome > 0 && $currentYearIncome > 0){
        $percentageDifferenceYear = (($currentYearIncome - $previousYearIncome) /
abs($previousYearIncome)) * 100;
        $roundedPercentageDifferenceYear = round($percentageDifferenceYear, 2);

        $jsonResultYear = [
            'currentYear' => date('Y'),
            'previousYear' => date('Y') - 1,
            'percentageDifferenceYear' => $roundedPercentageDifferenceYear,
        ];

        echo json_encode($jsonResultYear);
    }else{
        echo "Недостатньо даних.";
    }
} else {
    echo "Недостатньо даних.";
}

?>

```

load_currency.php:

```

<?php

if ($_GET['UserId']== "") {echo "Error: UserId is not set"; exit;};

require_once ("db-model-BE.php");

$UserId = $_GET['UserId'];

```

```

$result = Query::getQuery("SELECT `IncomeCurrency` FROM `Income` WHERE
`UserId`='$UserId'");
while ($row = mysqli_fetch_assoc($result))
{
    echo $row['IncomeCurrency'];
};
?>

```

load_month_analytics.php:

```

<?php
if ($_GET['UserId']== "") {echo "Error: UserId is not set"; exit;};

require_once ("db-model-BE.php");

$UserId = $_GET['UserId'];

$result = Query::getQuery("SELECT `Value`, `Month`, `Year` FROM `IncomeMonth` WHERE
`UserId` = '$UserId' ORDER BY `Date` DESC LIMIT 2;");

if ($result && mysqli_num_rows($result) >= 2) {

    $row = mysqli_fetch_assoc($result);
    $latestIncome = $row['Value'];
    $latestMonth = $row['Month'];
    $latestYear = $row['Year'];

    $row = mysqli_fetch_assoc($result);
    $previousIncome = $row['Value'];

    $percentageDifference = (($latestIncome - $previousIncome) /
abs($previousIncome)) * 100;
    $roundedPercentageDifference = round($percentageDifference, 2);

    $jsonResult = [
        'latestMonth' => $latestMonth,
        'latestYear' => $latestYear,
        'percentageDifference' => $roundedPercentageDifference,
    ];

    echo json_encode($jsonResult);
} else {

    echo "Недостатньо даних.";
}

?>

```

load_month_income.php:

```

<?php
if ($_GET['UserId']== "") {echo "Error: UserId is not set"; exit;};

require_once ("db-model-BE.php");

$UserId = $_GET['UserId'];

```

```

$result = Query::getQuery("SELECT `Value`, `Month`, `Year` FROM `IncomeMonth` WHERE
`UserId`='{$UserId}");

echo "[";
$i=0;
while ($row = mysqli_fetch_assoc($result))
{
    if ($i==0) $i=1;
    else echo ",";
    echo "{\"income\":\":".$row['Value']. "\", \"month\":\":".$row['Month']. "\",
\"year\":\":".$row['Year']. "\"}";
};
echo "]";
?>

```

load_trend_analytics.php:

```

<?php
if ($_GET['UserId'] == "") {
    echo "Error: UserId is not set";
    exit;
}

require_once("db-model-BE.php");

$UserId = $_GET['UserId'];

$resultAllMonths = Query::getQuery("SELECT `Value`, `Month`, `Year` FROM
`IncomeMonth` WHERE `UserId` = '{$UserId}' AND `Year` = YEAR(CURDATE()) ORDER BY `Date`
ASC;");

if ($resultAllMonths && mysqli_num_rows($resultAllMonths) > 0) {
    $dataRows = [];
    $currentIncreaseStreak = $currentDecreaseStreak = 1;
    $maxIncreaseStreak = $maxDecreaseStreak = 0;
    $startMonthIncrease = $endMonthIncrease = $startYearIncrease = $endYearIncrease =
    $startMonthDecrease = $endMonthDecrease = $startYearDecrease = $endYearDecrease =
    null;
    $startValueIncrease = $endValueIncrease = $startValueDecrease = $endValueDecrease
    = 0;

    $monthlyIncome = [];
    $firstRow = mysqli_fetch_assoc($resultAllMonths);
    $monthlyIncome[0] = $firstRow['Value'];

    $prevIncome = $firstRow['Value'];
    $prevMonth = $firstRow['Month'];
    $prevYear = $firstRow['Year'];
    $currentStreakStartMonth = $prevMonth;
    $currentStreakStartYear = $prevYear;
    $temp = 0;

    while ($row = mysqli_fetch_assoc($resultAllMonths)) {

        $temp++;
        $currentIncome = $row['Value'];
        $currentMonth = $row['Month'];
        $currentYear = $row['Year'];

```



```

$monthlyIncome[$temp] = $currentIncome;

$percentageDifference = (($currentIncome - $prevIncome) / abs($prevIncome)) *
100;

if ($percentageDifference > 0) {
    $currentIncreaseStreak++;
    if ($currentIncreaseStreak > $maxIncreaseStreak) {
        $maxIncreaseStreak = $currentIncreaseStreak;
        $startMonthIncrease = $currentStreakStartMonth;
        $startYearIncrease = $currentStreakStartYear;
        $endMonthIncrease = $currentMonth;
        $endYearIncrease = $currentYear;
        $startValueIncrease = $prevIncome;
        $endValueIncrease = $currentIncome;
    }

    $currentDecreaseStreak = 1;
    $currentStreakStartMonth = $currentMonth;
    $currentStreakStartYear = $currentYear;
} else {
    $currentDecreaseStreak++;
    if ($currentDecreaseStreak > $maxDecreaseStreak) {
        $maxDecreaseStreak = $currentDecreaseStreak;
        $startMonthDecrease = $currentStreakStartMonth;
        $startYearDecrease = $currentStreakStartYear;
        $endMonthDecrease = $currentMonth;
        $endYearDecrease = $currentYear;
        $startValueDecrease = $prevIncome;
        $endValueDecrease = $currentIncome;
    }

    $currentIncreaseStreak = 1;
    $currentStreakStartMonth = $currentMonth;
    $currentStreakStartYear = $currentYear;
}

$prevIncome = $currentIncome;
$prevMonth = $currentMonth;
$prevYear = $currentYear;
}

$temp_1 = $endMonthIncrease-$maxIncreaseStreak;
$temp_2 = $endMonthIncrease-1;
$temp_3 = $endMonthDecrease-$maxDecreaseStreak;
$temp_4 = $endMonthDecrease-1;

$jsonResult = [
    'maxIncreaseStreak' => $maxIncreaseStreak,
    'startMonthIncrease' => str_pad($endMonthIncrease-$maxIncreaseStreak+1, 2,
'0', STR_PAD_LEFT),
    'startYearIncrease' => $startYearIncrease,
    'endMonthIncrease' => str_pad($endMonthIncrease, 2, '0', STR_PAD_LEFT),
    'endYearIncrease' => $endYearIncrease,
    'percentageIncomeIncrease' =>
round(calculatePercentageIncrease($monthlyIncome[$temp_1], $monthlyIncome[$temp_2]),
2),
    'maxDecreaseStreak' => $maxDecreaseStreak,
    'startMonthDecrease' => str_pad($endMonthDecrease-$maxDecreaseStreak+1, 2,
'0', STR_PAD_LEFT),

```

```

        'startYearDecrease' => $startYearDecrease,
        'endMonthDecrease' => str_pad($endMonthDecrease, 2, '0', STR_PAD_LEFT),
        'endYearDecrease' => $endYearDecrease,
        'percentageIncomeDecrease' =>
round(calculatePercentageIncrease($monthlyIncome[$temp_3], $monthlyIncome[$temp_4]),
2),
    ];

    echo json_encode($jsonResult);
} else {
    echo "Недостатньо даних для аналізу.";
}

function calculatePercentageIncrease($startValue, $endValue) {
    return (($endValue - $startValue) / abs($startValue)) * 100;
}
?>

```

load_year_analytics.php:

```

<?php
if ($_GET['UserId'] == "") {
    echo "Error: UserId is not set";
    exit;
}

require_once("db-model-BE.php");

$UserId = $_GET['UserId'];

$resultTwelfthMonth = Query::getQuery("SELECT `Value`, `Month`, `Year` FROM
`IncomeMonth` WHERE `UserId` = '$UserId' AND `Year` = YEAR(CURDATE()) AND `Month` =
12 ORDER BY `Date` DESC LIMIT 1;");

$resultFirstMonth = Query::getQuery("SELECT `Value`, `Month`, `Year` FROM
`IncomeMonth` WHERE `UserId` = '$UserId' AND `Year` = YEAR(CURDATE()) AND `Month` = 1
ORDER BY `Date` DESC LIMIT 1;");

if ($resultTwelfthMonth && mysqli_num_rows($resultTwelfthMonth) > 0 &&
$resultFirstMonth && mysqli_num_rows($resultFirstMonth) > 0) {
    $rowTwelfthMonth = mysqli_fetch_assoc($resultTwelfthMonth);
    $incomeTwelfthMonth = $rowTwelfthMonth['Value'];
    $yearTwelfthMonth = $rowTwelfthMonth['Year'];

    $rowFirstMonth = mysqli_fetch_assoc($resultFirstMonth);
    $incomeFirstMonth = $rowFirstMonth['Value'];

    $percentageDifferenceYear = (($incomeTwelfthMonth - $incomeFirstMonth) /
abs($incomeFirstMonth)) * 100;
    $roundedPercentageDifferenceYear = round($percentageDifferenceYear, 2);

    $jsonResultYear = [
        'year' => $yearTwelfthMonth,
        'percentageDifferenceYear' => $roundedPercentageDifferenceYear,
    ];

    echo json_encode($jsonResultYear);
} else {

```

```

        echo "Недостатньо даних.";
    }
    ?>

```

set_currency.php:

```

<?php
if ($_GET['UserId']== "") {echo "Error: UserId is not set"; exit;};

require_once ("db-model-BE.php");

$UserId = $_GET['UserId'];
$IncomeCurrency = $_GET['IncomeCurrency'];

$result = Query::getQuery("SELECT `UserId` FROM `Income` WHERE `UserId`='$UserId'");
$row = mysqli_fetch_assoc($result);

if (isset($row['UserId'])) Query::getQuery("UPDATE `Income` SET
`IncomeCurrency`='$IncomeCurrency' WHERE `UserId`='$UserId'");

else Query::getQuery("INSERT INTO `Income` (`UserId`, `IncomeCurrency`) VALUES
('$UserId', '$IncomeCurrency')");
?>

```

set_month_income.php:

```

<?php
if ($_GET['UserId']== "") {echo "Error: UserId is not set"; exit;};

require_once ("db-model-BE.php");

$UserId = $_GET['UserId'];
$IncomeMonth = $_GET['IncomeMonth'];

echo $IncomeMonth;
$inputArr = json_decode($IncomeMonth, true);

for ($i=0; $i<count($inputArr); $i++)
{
    $value = $inputArr[$i]['income'];
    $month = $inputArr[$i]['month'];
    $year = $inputArr[$i]['year'];

    $result = Query::getQuery("SELECT `UserId` FROM `IncomeMonth` WHERE
`UserId`='$UserId' AND `Month`='$month' AND `Year`='$year'");
    $row = mysqli_fetch_assoc($result);

    if (isset($row['UserId'])) {
        $result2 = Query::getQuery("UPDATE `IncomeMonth` SET `Value`='$value' WHERE
`UserId`='$UserId' AND `Month`='$month' AND `Year`='$year'");
    } else {
        $result2 = Query::getQuery("INSERT INTO `IncomeMonth` (`UserId`, `Value`,
`Month`, `Year`) VALUES ('$UserId', '$value', '$month', '$year')");
    }
}
?>

```