

**MINISTRY OF EDUCATION AND SCIENCE OF UKRAINE**

**Sumy State University**

Academic and Research Institute of Business, Economics and Management

(full name of the institute/faculty)

Economic Cybernetics Department

(full name of the department)

«Admitted to the defense»

Head of Department

\_\_\_\_\_ Vitaliia KOIBICHUK

(signature)

(First and LAST NAME)

\_\_\_\_\_ 20\_\_ p.

**QUALIFICATION WORK**

**to obtain an educational degree**   bachelor  

(bachelor / master)

from the specialty 051 Economics

(code and name)

educational-professional programs Economic Cybernetics

(educational-professional / educational-scientific)

(the name of the program)

on the topic: Development of a Web-Based Accounting and Personal Finance Analysis System

Winner(s) of the group EK-91a

(group cipher)

Shramko Elina Volodymyrivna

(full name)

The qualification work contains the results of own research. The use of ideas, results and texts of other authors are linked to the corresponding source

  
\_\_\_\_\_

(signature)

Elina Shramko

(Name and SURNAME of the acquirer)

Head  
docent, candidate of technical sciences

Kostiantyn HRYTSENKO

(position, academic degree, academic title, Name and SURNAME)



(signature)

Consultant \_\_\_\_\_

(position, academic degree, academic title, Name and SURNAME)

(signature)

**Sumy – 2023**

Міністерство освіти і науки України  
Сумський державний університет  
Навчально-науковий інститут бізнесу, економіки та менеджменту  
Кафедра економічної кібернетики

## КВАЛІФІКАЦІЙНА РОБОТА БАКАЛАВРА

на тему «РОЗРОБКА WEBОРІЄНТОВАНОЇ СИСТЕМИ ОБЛІКУ  
ТА АНАЛІЗУ ОСОБИСТИХ ФІНАНСІВ»

Виконав студент 4 курсу, групи ЕК-91а  
(номер курсу) (шифр групи)

Спеціальності 051 «Економіка»  
(Економічна кібернетика)

Шрамко Е. В.  
(прізвище, ініціали студента)

Керівник доцент, к.т.н. Гриценко К.Г.  
(посада, науковий ступінь, прізвище, ініціали)

Суми – 2023 рік

## АНОТАЦІЯ

кваліфікаційної роботи бакалавра на тему  
«РОЗРОБКА ВЕБОРІЄНТОВАНОЇ СИСТЕМИ ОБЛІКУ  
ТА АНАЛІЗУ ОСОБИСТИХ ФІНАНСІВ»

студента Щрамко Еліни Володимирівни  
(прізвище, ім'я, по батькові)

Актуальність теми, обраної для дослідження, визначається тим, що облік та аналіз особистих фінансів є ключовими елементами фінансової грамотності. Усвідомлення важливості цих процесів людиною надзвичайно корисне при плануванні бюджету та прийнятті майбутніх фінансових рішень. З огляду на сучасний рівень розвитку інформаційних технологій та фінансової сфери, стає очевидним зростаючий попит на прості в користуванні, прозорі інструменти для обліку та аналізу особистих фінансів. Відповідно до цих тенденцій, створення вебсайту, орієнтованого на такий запит, є вкрай важливим. Цей ресурс зможе значно полегшити процеси обліку доходів та витрат, допомогти проводити ідентифікацію та аналіз фінансових звичок. Це, у свою чергу, підвищить якість прийняття обґрунтованих фінансових рішень, що є на меті підвищення ефективності управління особистими фінансами.

Мета кваліфікаційної роботи полягає у створенні веборієнтованої системи, яка дозволить автоматизувати процеси обліку та аналізу особистих фінансів.

Об'єктом дослідження є процеси обліку та аналізу особистих фінансів.

Предметом дослідження є сучасні програмні засоби, моделі та технології створення вебсайту у якості веборієнтованої системи обліку та аналізу особистих фінансів.

Задачами є дослідження стану автоматизації бізнес-процесів та формування вимог до веборієнтованої системи обліку та аналізу особистих фінансів, проектування та реалізація веборієнтованої системи обліку та аналізу особистих фінансів.

Для досягнення поставленої мети та задач дослідження були використані такі методи дослідження: аналіз та синтез, структурний та функціональний аналіз, сучасні методи та технології розробки та проєктування веборієнтованих систем обліку та аналізу особистих фінансів.

Інформаційною базою кваліфікаційної роботи є наукові дослідження, присвячені питанням обліку та аналізу особистих фінансів; існуючі програмні рішення в даній галузі; інформація про особисті доходи та витрати.

Основним науковим результатом кваліфікаційної роботи є розробка та впровадження веборієнтованої системи обліку та аналізу особистих фінансів.

Отримані результати можуть бути використані для аналізу процесів обліку та аналізу особистих фінансів, а також використання веборієнтованих систем у якості інструмента покращення ефективності їхньої діяльності.

Ключові слова: особисті фінанси, автоматизація, веборієнтована система, моделювання бізнес-процесів.

Зміст кваліфікаційної роботи викладено на 35 сторінках. Список використаних джерел із 42 найменувань, розміщений на 3 сторінках. Робота містить 6 таблиць, 15 рисунків, а також 4 додатків, розміщених на 11 сторінках.

Рік виконання кваліфікаційної роботи – 2023 рік.

Рік захисту роботи – 2023 рік.

Міністерство освіти і науки України  
Сумський державний університет  
Навчально-науковий інститут бізнесу, економіки та менеджменту  
Кафедра економічної кібернетики

ЗАТВЕРДЖУЮ  
Завідувач кафедри  
к.е.н., доцент  
\_\_\_\_\_ В.В. Койбічук  
“ \_\_\_ ” \_\_\_\_\_ 2023 р.

ЗАВДАННЯ НА КВАЛІФІКАЦІЙНУ РОБОТУ БАКАЛАВРА  
спеціальність 051 «Економіка (Економічна кібернетика)»  
студенту IV курсу, групи ЕК-91а

Шрамко Еліні Володимирівні

(прізвище, ім'я, по батькові студента)

1. Тема роботи Розробка веборієнтованої системи обліку та аналізу особистих фінансів

затверджена наказом по університету від «23» травня 2023 року № 0554-VI

2. Термін подання студентом закінченої роботи «16» червня 2023 року

3. Мета кваліфікаційної роботи створення веборієнтованої системи, яка дозволить автоматизувати процеси обліку та аналізу особистих фінансів

4. Об'єкт дослідження процеси обліку та аналізу особистих фінансів

5. Предмет дослідження сучасні програмні засоби, моделі та технології створення вебсайту у якості веборієнтованої системи обліку та аналізу особистих фінансів

6. Кваліфікаційна робота виконується на матеріалах наукових досліджень, присвячених питанням обліку та аналізу особистих фінансів; існуючих програмних рішень в даній галузі; інформації про особисті доходи та витрати.

7. Орієнтовний план кваліфікаційної роботи, терміни подання розділів керівникові та зміст завдань для виконання поставленої мети

Розділ 1. Дослідження стану автоматизації процесів обліку та аналізу особистих фінансів і проектування веборієнтованої системи – 23 травня 2023 року

(назва – термін подання)

У розділі 1 проаналізувати стан автоматизації процесів обліку та аналізу особистих фінансів, сформулювати вимоги до веборієнтованої системи, розробити моделі процесів у нотаціях IDEF0, IDEF3 і DFD, описати архітектуру веборієнтованої системи, її функціональну структуру,

підсистеми забезпечення функціональної частини та технології вирішення поставлених завдань.

Розділ 2. Реалізація прототипу веборієнтованої системи – 3 червня 2023 року  
(назва – термін подання)

У розділі 2 розглянути структуру та особливості реалізації інформаційного та алгоритмічного забезпечення. Навести інструкцію по використанню.

(зміст конкретних завдань до розділу, які повинен виконати студент)

## 8. Консультації з роботи:

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв
1			
2			
3			

9. Дата видачі завдання: «03» квітня 2023 року

Керівник кваліфікаційної роботи

  
(підпис)

К.Г. Гриценко  
(ініціали, прізвище)

Завдання до виконання одержав

  
(підпис)

Е.В. Шрамко  
(ініціали, прізвище)

## ЗМІСТ

ВСТУП.....	3
РОЗДІЛ 1. ДОСЛІДЖЕННЯ СТАНУ АВТОМАТИЗАЦІЇ ПРОЦЕСІВ ОБЛІКУ ТА АНАЛІЗУ ОСОБИСТИХ ФІНАНСІВ І ПРОЄКТУВАННЯ ВЕБОРІЄНТОВАНОЇ СИСТЕМИ.....	5
1.1. Характеристика об'єкта дослідження.....	5
1.2. Аналіз стану автоматизації процесів обліку та аналізу особистих фінансів .....	8
1.3. Формування вимог до веборієнтованої системи.....	12
1.4. Розробка моделей бізнес-процесів.....	16
1.5. Архітектура веборієнтованої системи.....	20
РОЗДІЛ 2. РЕАЛІЗАЦІЯ ПРОТОТИПУ ВЕБОРІЄНТОВАНОЇ СИСТЕМИ..	26
2.1. Структура та особливості реалізації інформаційного забезпечення .....	26
2.2. Структура та особливості реалізації алгоритмічного забезпечення .....	30
2.3. Контрольний приклад та інструкція по використанню .....	33
ВИСНОВКИ .....	36
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ .....	38
ДОДАТКИ .....	43

## ВСТУП

Облік та аналіз особистих фінансів є важливими аспектами фінансової грамотності. Вони допомагають ефективно керувати доходами та витратами, свідомо розуміти, скільки грошей витрачається на певні категорії витрат. Облік дозволяє ідентифікувати області, де можна зекономити, а аналіз фінансів допомагає виявити тенденції та зміни. Ця інформація корисна для планування бюджету та майбутніх фінансових рішень. Систематичний облік і аналіз фінансів також сприяють формуванню довгострокової стратегії управління особистими фінансами [44].

На сучасному етапі розвитку інформаційних технологій та фінансової сфери, виникає все більше потреба в зручних та прозорих інструментах для обліку та аналізу особистих фінансів. Розвиток веборієнтованих технологій надає можливість кожному користувачу вести особистий фінансовий контроль в режимі онлайн, що, безумовно, призведе до підвищення рівня фінансової грамотності населення України [36].

Актуальність обраної теми стає очевидною на перехресті цифровізації та фінансової грамотності. Якісний контроль та аналіз особистих фінансів є ключовими етапами у економічному розвитку та фінансовому добробуту сучасних людей. Цифрові технології, спрямовані на допомогу в цих питаннях, відкривають нові горизонти для оптимізації, контролю та прогнозування фінансової діяльності користувачів. Вебсайт для обліку та аналізу особистих фінансів може значно полегшити облік доходів та витрат, допомогти в ідентифікації та аналізі фінансових звичок та тенденцій, а також у прийнятті обґрунтованих фінансових рішень. Таким чином, тема дослідження відображає актуальні потреби суспільства та виклики, з якими зіштовхується фінансова сфера в умовах цифрової трансформації.

Об'єктом дослідження є процеси обліку та аналізу особистих фінансів.

Предметом дослідження є сучасні програмні засоби, моделі та технології створення вебсайту у якості веборієнтованої системи обліку та аналізу особистих фінансів.



Метою кваліфікаційної бакалаврської роботи є створення веборієнтованої системи, яка дозволить автоматизувати процеси обліку та аналізу особистих фінансів.

Для досягнення поставленої мети необхідно виконати наступні завдання:

- скласти характеристику об'єкта дослідження;
- проаналізувати стан автоматизації процесів обліку та аналізу особистих фінансів;
- сформулювати та обґрунтувати вимоги до веборієнтованої системи обліку та аналізу особистих фінансів;
- розробити моделі процесів обліку та аналізу особистих фінансів у нотаціях IDEF0, IDEF3 і DFD;
- описати архітектуру веборієнтованої системи, її функціональну структуру, підсистеми забезпечення функціональної частини та технології вирішення поставлених завдань;
- розглянути структуру та особливості реалізації інформаційного та алгоритмічного забезпечення;
- розробити веборієнтовану інформаційну систему обліку та аналізу особистих фінансів;
- написати коротку інструкцію користувача.

Інформаційною базою для кваліфікаційної роботи бакалавра стануть наукові дослідження, присвячені питанням обліку та аналізу особистих фінансів; існуючі програмні рішення в даній галузі; інформація про особисті доходи та витрати.

# РОЗДІЛ 1. ДОСЛІДЖЕННЯ СТАНУ АВТОМАТИЗАЦІЇ ПРОЦЕСІВ ОБЛІКУ ТА АНАЛІЗУ ОСОБИСТИХ ФІНАНСІВ І ПРОЄКТУВАННЯ ВЕБОРІЄНТОВАНОЇ СИСТЕМИ

## 1.1. Характеристика об'єкта дослідження

В рамках даного дослідження проводиться не лише аналіз процесів обліку та аналізу особистих фінансів, але й розробка веборієнтованої системи, призначеної для автоматизації цих процесів. Тож, об'єктом дослідження є процеси обліку та аналізу особистих фінансів в умовах використання веборієнтованих технологій.

Під обліком та аналізом особистих фінансів мається на увазі весь спектр дій, які здійснюють окремі особи, щоб ефективно використовувати свої грошові ресурси аби максимізувати індивідуальний добробут [37]. Основна мета таких дій полягає в забезпеченні достатньої кількості інформації для прийняття обґрунтованих рішень, які посприятимуть оптимізації управління грошовими ресурсами.

Існують різні методи обліку особистих фінансів (табл. 1.1). Вибір методу обліку особистих фінансів залежить від особистих потреб та вподобань, а також від рівня зручності, доступності та функціональних можливостей, які необхідні для ефективного управління фінансами.

Таблиця 1.1 – Методи обліку особистих фінансів

<b>Метод</b> <b>Критерій</b>	<b>Ручний</b>	<b>Електронні таблиці</b>	<b>Фінансові програми</b>
Час	Довго	Середньо	Швидко
Складність	Проста	Середня	Різна (залежно від програми)
Доступність	Легко	Легко	Легко
Спостереження	Важко	Легко	Легко
Аналітика	Важко	Основна	Розширена
Автоматизація	Відсутня або обмежена	Обмежена	Висока
Синхронізація	Відсутня або обмежена	Обмежена	Висока
Резервне копіювання	Важко	Легко	Легко
Функціональність	Обмежена	Середня	Різна (залежно від програми)
Кооперація	Важко	Легко	Легко

Звідси можемо зрозуміти, що спеціалізовані фінансові програми, такі як веборієнтовані системи, є лідером серед методів обліку, оскільки можуть забезпечити розширену аналітику, високу автоматизацію, легку співпрацю та резервне копіювання. Вони надають більшу функціональність, але складність та функціонал можуть варіюватись залежно від конкретної програми.

Доходи і витрати є двома основними складовими фінансової інформації, які відображають фінансову ситуацію і допомагають керувати особистими фінансами. Розглянемо їх детальніше:

1. Доходи – це грошові суми, які надходять із різних джерел. У статті «Особисте фінансове планування: сутність, специфіка, основні етапи» Кізіма Т. зазначає: «Практика показує, що доходи та майно підрахувати набагато легше, оскільки джерел отримання доходів менше, ніж витрат, і вони частіше мають документальне підтвердження» [37]. Вони можуть включати:

- заробітну плату;
- пенсію або стипендію;
- дивіденди;
- відсотки з банківських депозитів;
- прибуток від бізнесу.

2. Витрати – це грошові суми, які витрачаються на різні потреби і послуги. Категорії витрат – це простий спосіб з'ясувати, на що витрачається найбільше грошей. Вони, звичайно, будуть відрізнятися у кожної людини, тому веборієнтована система може перелічувати загальновідомі категорії витрат (табл. 1.2), а також створити можливість налаштувати їх самостійно.

Таблиця 1.2 – Категорії витрат [37]

№	Узагальнена назва	Підкатегорія
1.1	Обов'язкові витрати	Продукти харчування
1.2		Комунальні платежі
1.3		Побутові товари
1.4		Проїзд у громадському транспорті
1.5		Обслуговування особистого транспорту
2.1	Особисті витрати	Розваги
2.2		Подарунки
2.3		Одяг та взуття
2.4		Кафе та ресторани

## Продовження таблиці 1.2

2.5		Краса та здоров'я
2.6		Освітні послуги
3	Непередбачувані витрати	
4	Інвестиції	
5	Придбання товарів довгострокового користування	
6	Благодійність	
7	Особистий резерв	

Аналіз особистих фінансів включає різні методи, що допомагають зрозуміти, оцінити та планувати ваші фінансові показники. Ось опис деяких з них:

– розрахунок рівня доходів та витрат – включає зіставлення загального рівня доходу з загальним рівнем витрат. Це дозволяє оцінити фінансову стійкість, ефективність управління фінансами та виявити потенційні проблеми;

– перевірка плану витрат – процес співставлення очікуваних витрат з фактичними за певний період (наприклад, місяць чи рік). Це допомагає контролювати витрати, забезпечує основу для прийняття фінансових рішень та планування майбутніх заощаджень;

– виявлення трендів та прогнозування фінансових показників – включає аналіз попередніх фінансових даних для виявлення трендів. Це передбачає використання графіків та діаграм, щоб візуалізувати зміни у фінансових показниках. На основі цього аналізу можна планувати майбутні витрати та доходи.

Завдяки новим технологіям, що активно розвиваються, сьогодні є можливим вести облік своїх доходів, витрат та заощаджень. Перевагою веборієнтованих систем є однаковий інтерфейс як для перегляду зі смартфона, так і для перегляду з комп'ютера, що не змушує ще раз вчитися працювати з системою при зміні пристрою. Таким чином можна побачити чітку картину того, куди людина витрачає свої кошти, проаналізувати доходи і зробити відповідні висновки [28].

## 1.2. Аналіз стану автоматизації процесів обліку та аналізу особистих фінансів

State of Mobile 2023 [26], показує, що мобільні додатки, як ніколи раніше, є провідним індикатором змін у поведінці споживачів. Телефони та планшети стали невід’ємною частиною нас самих – часто вони є першою ланкою, до якої ми звертаємося за допомогою, реакцією та зв’язком. Гарною новиною є те, що попит на мобільні додатки ще ніколи не був таким високим, проте спостерігається перше в історії зниження споживчих витрат на магазини додатків через макроекономічні умови [24].

Впровадження мобільних додатків у таких популярних сферах, як мобільний банкінг, цифрові гаманці та платежі, а також персональні кредити, стрімко зростало у 2022 році, продовжуючи прискорений перехід на мобільні пристрої, який розпочався з початком пандемії у 2020 році.

Також відмічається значний прогрес в області управління персональними фінансами. Це було спричинено розвитком FinTech-стартапів та інноваційних рішень, які забезпечують різноманітні можливості для управління особистими фінансами.

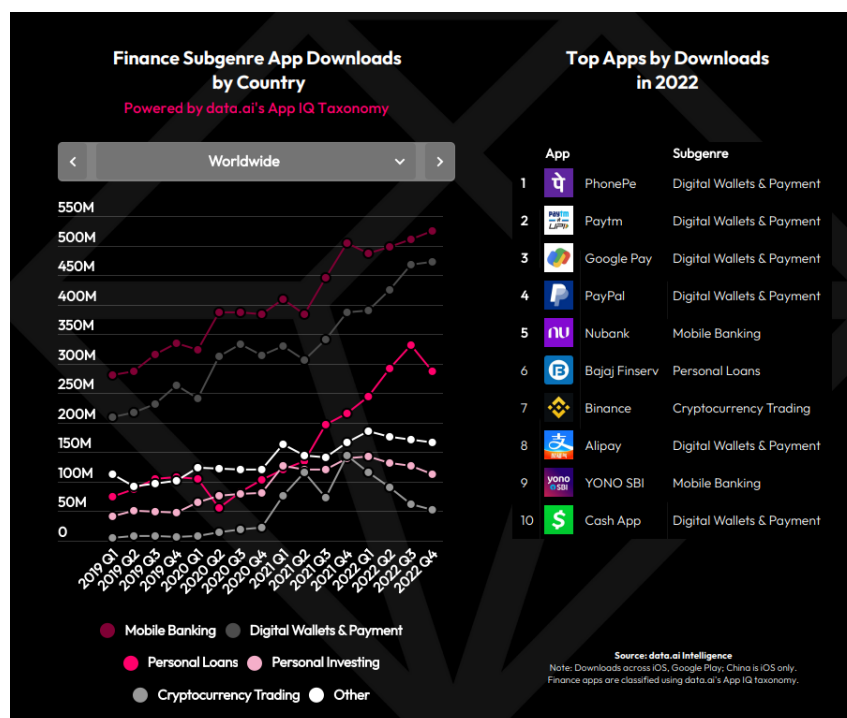


Рисунок 1.1 – Динаміка використання фінансових мобільних додатків [26]

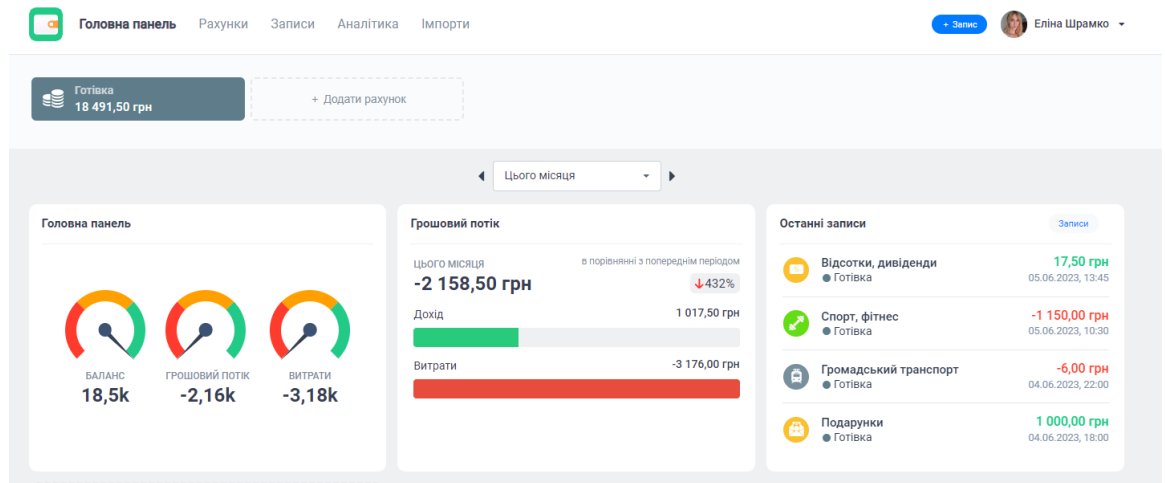
Спостерігаючи за поточним станом сфери фінансових інструментів доступних в Україні, можна констатувати проблему недостатньої кількості вебсайтів, призначених для обліку та аналізу особистих витрат. Це, безперечно, ставить певні обмеження на споживачів, що прагнуть оптимізувати свої витрати, та складає додаткові труднощі для прозорого та ефективного керування особистими фінансами.

У ході здійсненого дослідження було виявлено лише один вебсайт, що надає можливість вести облік та проводити аналіз особистих витрат для українських користувачів.

BudgetBakers є фінансовим вебсайтом, що допомагає вам краще управляти своїми грошима. Наступні є його ключовими характеристиками:

- категорії та теги: є можливість сортувати свої витрати та доходи за допомогою існуючих категорій або створювати власні;
- відстеження витрат: додаток дозволяє вносити витрати вручну або автоматично імпортувати транзакції з банківських рахунків;
- планування бюджету: користувач має можливість створювати бюджети для різних категорій витрат та відстежувати їх у реальному часі;
- синхронізація між пристроями: дані можуть бути синхронізовані між різними пристроями, що дозволяє управляти своїми фінансами з будь-якого місця;
- безпека: BudgetBakers стверджує, що вони використовують передові технології безпеки для захисту персональних даних [9].

Загалом, інтерфейс BudgetBakers (рис 1.2) є досить зручним для використання, а функціонал покриває основні потреби користувачів в області особистого фінансового планування.



Рисунк 1.2 – Інтерфейс вебсайту BudgetBakers

Однак, протягом дослідження, було виявлено декілька недоліків, що, хоча і незначні, можуть певною мірою впливати на комфорт використання сервісу. Специфіку цих мінусів важливо взяти до уваги при формуванні загальної оцінки вебсайту та його потенціалу в контексті українського ринку фінансових технологій.

Таблиця 1.3 – Плюси та мінуси вебсайту BudgetBakers

Плюси	Мінуси
Багатий функціонал і гнучкість у плануванні бюджету	Деякі функції доступні тільки у платній версії
Інтерфейс є простим у використанні	В залежності від регіону та банку, автоматичний імпорт транзакцій може бути недоступний
Підтримує різні валюти	Відсутність можливості створення власних категорій

Протиставляючи ситуацію в мобільній сфері, можна зауважити вражаючу контрастність. Мобільний сегмент фінансових технологій характеризуються широким спектром функціоналу. Цей ринок є динамічним та конкурентним, і він надає користувачам широкий вибір інструментів для оптимізації своїх витрат та управління особистими фінансами.

1. Monefy – це мобільний додаток для відстеження ваших витрат і доходів. Це простий у використанні інструмент, який дозволяє вам краще розуміти, куди йдуть ваші гроші. Ось деякі з його ключових характеристик:

- синхронізація даних: Monefy дозволяє вам синхронізувати дані між різними пристроями за допомогою Dropbox;
- велика кількість валют: Monefy підтримує використання різних валют;
- звіти: додаток генерує графіки і діаграми, що дозволяють візуально аналізувати свої витрати і доходи [21].

Таблиця 1.4 – Плюси та мінуси мобільного додатку Monefy

Плюси	Мінуси
Простота використання і інтуїтивно зрозумілий інтерфейс	Автоматична синхронізація з банківськими рахунками відсутня; всі транзакції потрібно вводити вручну
Швидке додавання витрат і доходів	Деякі функції, такі як синхронізація з Dropbox і розблокування всіх категорій, доступні тільки в платній версії
Графічні інструменти для аналізу витрат	Немає можливості планувати бюджет або встановлювати мети заощадження

2. Money Manager – це додаток для управління особистими фінансами, який пропонує широкий спектр функцій для відстеження і планування витрат, доходів та інвестицій. Основні характеристики Money Manager:

- бюджетування: створення місячних та щоденних бюджетів та встановлення лімітів витрат для різних категорій;
- графіки та звіти: додаток генерує деталізовані графіки і звіти, що допомагають краще розуміти витрати, доходи та загальний фінансовий стан;
- планування майбутніх транзакцій: планування майбутніх витрат, доходів та переказів, що дозволяє краще контролювати свої фінанси [22].

Таблиця 1.5 – Плюси та мінуси мобільного додатку Money Manager

Плюси	Мінуси
Широкий спектр функцій для управління особистими фінансами	Всі транзакції потрібно вводити вручну, що може бути трудомістким
Деталізовані графіки та звіти для аналізу фінансових даних	Інтерфейс може бути складним для новачків
Можливість планувати майбутні транзакції	Деякі функції, такі як синхронізація даних або створення необмеженої кількості аккаунтів, доступні тільки в платній версії

Отже, BudgetBakers є цінним вебресурсом для управління особистими фінансами в Україні, надаючи можливість відстежувати витрати, створювати



бюджети та аналізувати фінансові звички. Він може бути складним для незнайомих користувачів, але він компенсує це широким функціоналом. Мобільні додатки Monefy і Money Manager також допомагають в управлінні особистими фінансами. Monefy орієнтований на швидке відстеження витрат, тоді як Money Manager надає більш детальний аналіз і планування. Враховуючи обмеження вебресурсів в Україні, [web.budgetbakers.com](http://web.budgetbakers.com) стає необхідним для управління особистими фінансами.

### 1.3. Формування вимог до веборієнтованої системи

Веборієнтована система – це інтегрована система, що складається з компонентів програмного забезпечення на стороні клієнта (браузера), які взаємодіють з компонентами програмного забезпечення на вебсервері для обробки та обміну даними. Вона використовує сервіси Інтернету, основані на архітектурі клієнт-сервер, використовуючи не тільки модель "запит-відповідь", але й можливо інші моделі взаємодії, такі як асинхронні зворотні виклики або вебсокети для двосторонньої комунікації в реальному часі. Стандарт обміну даними – HTTP, але можуть бути використані й інші протоколи, що відповідають конкретним потребам системи [12].

Чітко визначені вимоги є важливим елементом на шляху до успішної реалізації веборієнтованої системи. Якісні, детальні вимоги допомагають зменшити фінансові ризики та дотримуватися графіку виконання.

Керівництво з бізнес-аналізу (БАВОК) – всесвітньо визнаний стандарт, випущений Міжнародним інститутом бізнес-аналізу (ІІВА). БАВОК надає структуроване керівництво, в якому відображається передовий досвід і описуються області знань, необхідні дії, завдання та методи в контексті бізнес-аналізу. Він призначений для поліпшення бізнес-аналізу та включає в себе необхідні навички для ефективного виконання завдань [2].

Згідно з довідником БАВОК вимоги до проєкту можна класифікувати наступним чином:

1. Бізнес-вимоги.
2. Вимоги користувачів.
3. Вимоги до рішення (функціональні та нефункціональні) [2].

Бізнес-вимоги визначають стратегічний шлях проєкту. Вони включають вимоги до проєкту високого рівня, які описують загальну бізнес-ціль з точки зору компанії. [4]. Вимоги до сайту обліку та аналізу особистих фінансів можуть диктуватися потребами та трендами ринку. В нашому випадку є зростаюча потреба на ринку в вебінструменті для управління бюджетом, що може стати ключовою бізнес-вимогою для сайту.

Тож, в контексті веборієнтованої системи для управління бюджетом бізнес-вимоги є такими:

- відповідність ринковим трендам;
- залучення та утримання користувачів;
- підвищення фінансової грамотності;
- відповідність законодавству.

Вимоги користувачів зосереджуються на тому, що окремі групи користувачів хочуть або очікують від системи, і це може включати такі речі, як зручність використання, надійність, швидкість, специфічні функції тощо [4].

Якщо ми говоримо про веборієнтовану систему для управління бюджетом, то вимоги користувачів включають:

- легкість у використанні;
- здатність встановлювати і слідкувати за фінансовими цілями;
- безпека персональних даних;
- візуально зрозумілі аналітичні звіти [12].

Вимоги до продукту, які в нашому випадку є вимогами до веборієнтованої системи, – це конкретні описи можливостей і якостей, щоб відповідати вимогам користувачів і самого бізнесу. Вони мають певний рівень деталізації, щоб забезпечити ефективну розробку та впровадження рішення [2].

Функціональні вимоги – вимоги, що характеризують, як продукт або сервіс відповідає потребам клієнта. Вони включають в себе функції та можливості, які відображаються в сценаріях використання, деталізуючи взаємодію користувача з продуктом або послугою. Виходячи з бізнес-вимог та клієнтських вимог, було розроблено функціональні вимоги у формі “user stories” та “use cases” [3].

User story – інструмент, який використовується в Agile розробці програмного забезпечення, щоб зафіксувати опис функції програмного забезпечення з точки зору кінцевого користувача. Інакше кажучи, це неформальний опис природною мовою однієї або декількох функцій програмної системи. Він описує тип користувача, що він хоче і чому [34].

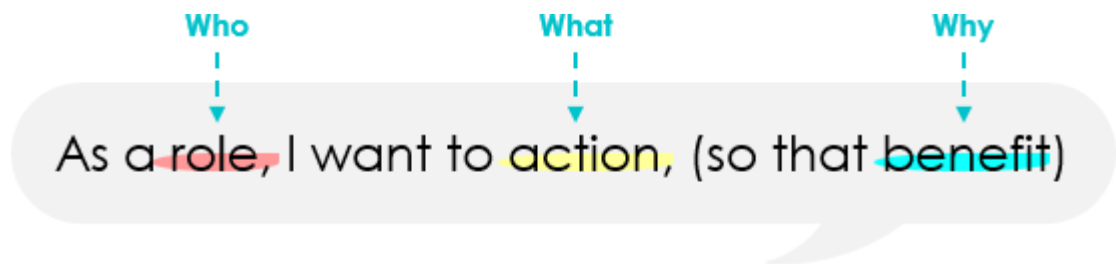


Рисунок 1.3 – Приклад формування user story [11]

Use case – це форма представлення вимог, що визначає взаємодію між користувачем і сервісом, яка може допомогти визначити функціональні вимоги до системи. Іншими словами, що повинен робити продукт або послуга, щоб задовольнити потреби і бажання клієнта [27]. Він має бути розроблений з урахуванням специфічної мети користувача і завжди включає в себе учасника та дію [11]. Наприклад, "незарєєстрований користувач" – це учасник, а "заповнити форму реєстрації" – це дія.

У системі обліку та аналізу особистих фінансів повинні бути присутні дві ключові ролі користувачів: незарєєстрований користувач (anonymous\_user) та зарєєстрований і авторизований користувач (user).

Таблиця 1.6 – Функціональні вимоги користувача до веборієнтованої системи обліку та аналізу особистих витрат

№	Тип вимоги	Роль користувача	Формулювання користувацької вимоги
1	User story	anonymouse_user	Як користувач хоче бачити головну сторінку сайту з описом його мети, переваг та особливостей
2	User story	anonymouse_user	Як користувач я хочу мати можливість реєстрації та подальшої авторизації на сайті
3	User story	user	Як користувач я хочу мати можливість перейти в особистий кабінет, щоб керувати особистими даними
4	User story	user	Як користувач я хочу мати можливість створення, перегляду, редагування та видалення різних рахунків з різними валютами
5	User story	user	Як користувач я хочу бачити актуальний баланс наявних рахунків
6	User story	user	Як користувач я хочу встановлювати бюджет витрат на певний період
7	User story	user	Як користувач я хочу мати можливість додавання та видалення записів про доходи/витрати та присвоювати їм певні категорії та додавати опис
8	User story	user	Як користувач я хочу мати можливість створення, редагування та видалення категорій
9	User story	user	Як користувач я хочу бачити аналітику використання моїх доходів, тенденції та рекомендації
10	Use case	user	Після заповнення форми реєстрації/логіну та натиснення кнопки «Підтвердити» відображається персональна сторінка користувача (або помилка у разі вводу неправильних даних)
11	Use case	user	При створенні рахунку натиснення кнопки «Створити» відображається сторінка цього рахунку
12	Use case	user	При заповненні форми на сторінці створення транзакції та натиснення кнопки «Додати» відображається сторінка рахунку, до якого була додана транзакція
13	Use case	user	При видаленні категорії, в якій існують транзакції, такі транзакції переміщуються у категорію «інше»
14	Use case	user	Після завершення періоду, на який був встановлений бюджет, з'являться повідомлення на сторінці аналітики щодо виконання/невиконання цілі

Нефункціональні вимоги – вимоги, що визначають стандарти зручності та ефективності, встановлюючи вимоги до загальної якості системи. Вони являють собою набір стандартів, які використовуються для оцінки конкретної роботи системи [16]. Невиконання нефункціональних вимог може призвести до того, що система не зможе задовольнити потреби користувачів.

Для веборієнтованої системи особистих фінансів, нефункціональні вимоги включають:

1. Зручність використання: інтерфейс має бути простим та зручним для користувачів з різним рівнем комп'ютерної грамотності.

2. Доступність: система має бути доступна для користувачів 24/7,

незалежно від їх місцезнаходження.

3. Безпека: система має забезпечувати захист від несанкціонованого доступу, а також має бути в стані шифрувати і зберігати конфіденційні дані.

4. Сумісність: система має бути сумісна з різними браузерами, операційними системами та пристроями

5. Продуктивність: система має бути в змозі обробляти велику кількість запитів одночасно і без значних затримок.

6. Масштабованість: система має добре адаптуватися до збільшення обсягу користувачів або транзакцій [16].

#### 1.4. Розробка моделей бізнес-процесів

Розробка моделей бізнес-процесів для веборієнтованої системи обліку та аналізу особистих фінансів привносить ясність та розуміння в функціонування системи для всіх зацікавлених сторін. Вона допомагає виявити та оптимізувати потенційно складні місця та зайві бізнес-рішення, що підвищує ефективність системи. Моделі також служать інструментами контролю та підтримки якості, дозволяючи виділити, як дані та завдання просуваються через систему. Крім того, вони дозволяють планувати та прогнозувати вплив потенційних змін, таких як внесення нових функцій або зміну бізнес-стратегії [40].

Існує багато методологій, що використовуються для розробки моделей бізнес-процесів. Ось деякі з них:

– IDEF0 – це структурована методологія, що використовується для представлення функцій бізнесу і їх взаємозв'язку;

– IDEF3 – це процесно-орієнтована методологія, що зосереджена на відображенні процесів і послідовностей дій у бізнес-процесах;

– DFD – це методологія, що використовується для візуалізації потоку даних через систему, включаючи входи, виходи, процеси і збереження даних [25].

Кожна з цих методологій має свої сильні і слабкі сторони, тому вибір залежить від конкретного контексту та вимог до моделювання. Однак, зупинимося на сімействі IDEF через його універсальність та гнучкість: він дозволяє моделювати різноманітні процеси, в тому числі бізнес-процеси [8]. Це також сприяє ясності та структурованості, оскільки він вимагає чіткого визначення елементів процесу, включаючи входи, виходи, функції та взаємозв'язки [23].

Зосередимося на моделюванні за IDEF0, що був розроблений спеціально для моделювання функціональних аспектів систем [18]. Використання IDEF0 зокрема корисне через його візуальну простоту та структурованість: він дозволяє створювати чіткі, логічні діаграми, включаючи лише два основних етапи:

1. Побудова контекстної діаграми – створення контекстної діаграми, яка представляє систему або процес на високому рівні. Контекстна діаграма зазвичай містить лише одну функцію (або процес) та її входи, виходи, механізми та контролю. Цей етап необхідний для встановлення обсягу системи або процесу та для забезпечення спільного розуміння мети моделювання між всіма зацікавленими сторонами [16].

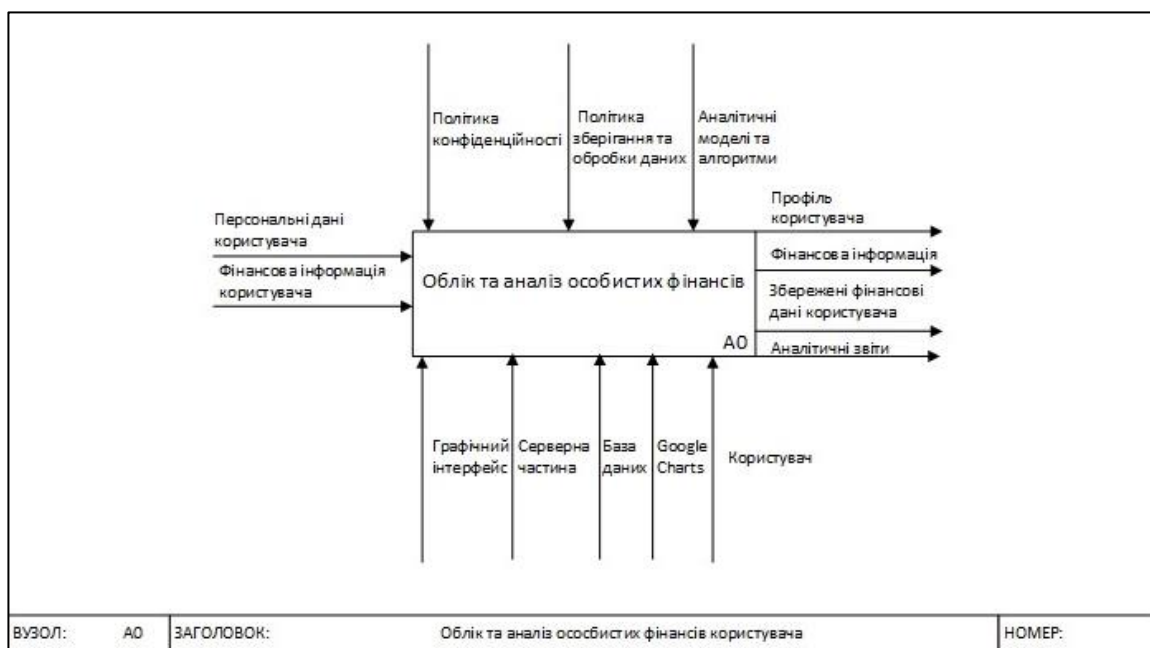


Рисунок 1.4 – Контекстна діаграма IDEF0 верхнього рівня «Облік та аналіз особистих фінансів»

Головний процес "Облік та аналіз особистих фінансів" – це багатофункціональний процес, що охоплює збір, обробку, зберігання та аналіз особистих фінансових даних користувача (рис. 1.4). Контекстна діаграма головного процесу включає 4 види інформації. Вхідні дані включають особисту та фінансову інформацію користувача. Вихідні дані представляють собою профіль користувача, збережені фінансові дані користувача та аналітичні звіти та дашборди, які представляють деталізований аналіз особистих фінансів користувача. Механізми, що підтримують цей процес, включають вебсайт для обліку та аналізу особистих фінансів, систему авторизації, базу даних, інструмент для візуалізації даних Google Charts, а також самого користувача. Керування процесом відбувається за допомогою встановлених політик, включаючи політику конфіденційності та безпеки.

2. Деталізація діаграм – створення детальних діаграм для кожної функції або процесу, представленого на контекстній діаграмі. Кожна діаграма деталізується за допомогою додаткових вхідних, вихідних даних, механізмів та контролю. Цей етап може повторюватися кілька разів для кожної підфункції, що дозволяє створити ієрархічну структуру діаграм [16].

Діаграма, представлена нижче (рис. 1.5), ілюструє та роз'яснює всі підпроцеси системи.

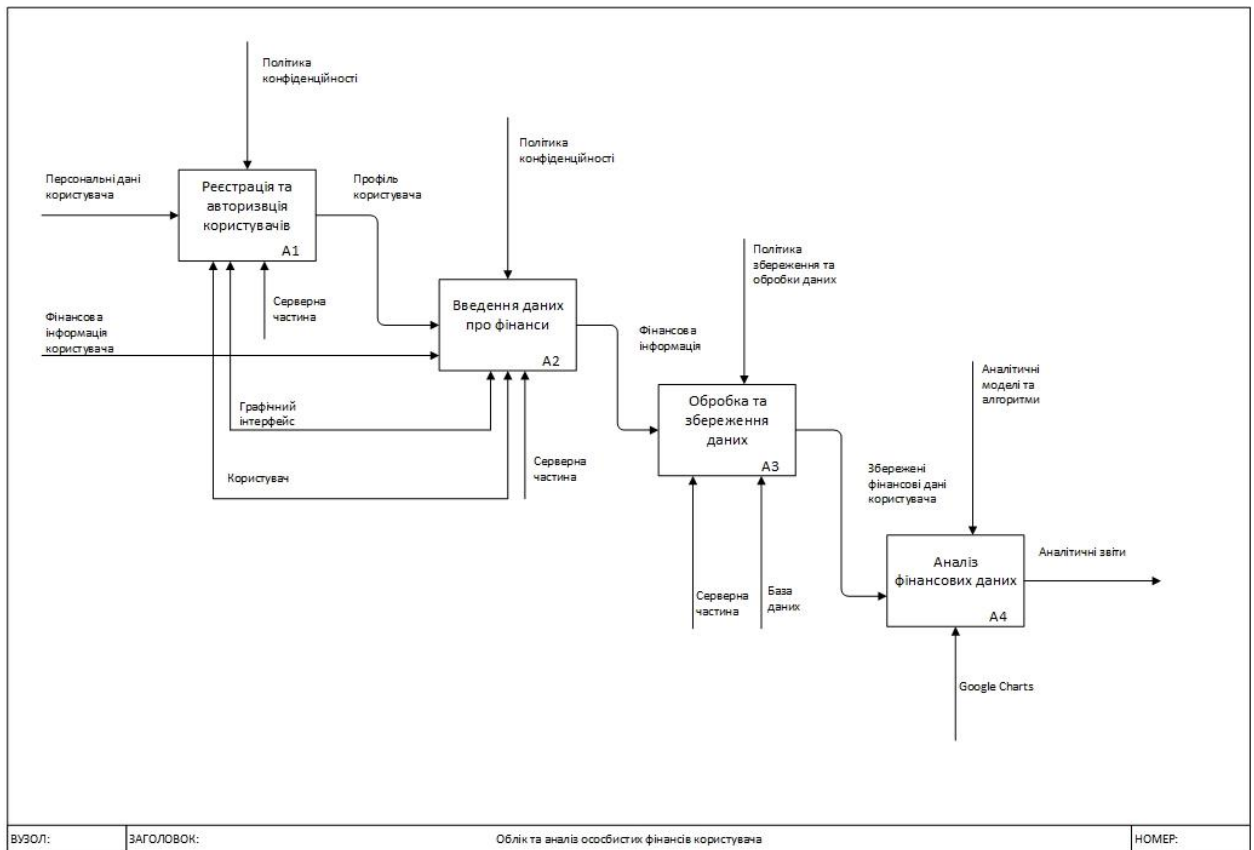


Рисунок 1.5 –Декомпозиція контекстної діаграми роботи вебсайту з обліку та аналізу особистих фінансів

Діяльність веборієнтованої системи обліку та аналізу особистих фінансів складається з 4 основних етапів:

1. Перший етап запускається, коли користувач заходить на вебсайт і реєструється, вводячи свою особисту інформацію. Він охоплює перевірку на валідність вхідних даних і створення облікового запису в системі. Керування процесом визначається політикою, що забезпечує конфіденційність даних. Механізми включають систему авторизації та користувача [41]. Вихідними даними цього процесу є створений профіль користувача.

2. У другому етапі користувач вводить свою фінансову інформацію: доходи та витрати. Керування процесом здійснюється за правилами введення даних, що гарантують валідність інформації. Механізми включають сам вебсайт та користувача. Вихідні дані – це «сирі» фінансові дані користувача.

3. Після того, як користувач ввів свої фінансові дані, система обробляє



цю інформацію та зберігає її в базі даних. Керування цим процесом здійснюється за допомогою стандартів збереження та обробки. Механізми включають вебсайт та базу даних. Вихідними даними цього процесу є збережені фінансові дані користувача.

4. Після збереження фінансових даних відбувається їх аналіз. Система використовує різні аналітичні моделі та алгоритми для виявлення тенденцій, паттернів або аномалій в даних. Керування процесом визначається встановленими налаштуваннями аналізу користувача. Механізми включають Google Charts [17] та користувача. Вихідними даними цього процесу є аналітичні звіти, які представлені користувачу.

Отже, функціонування вебсайту обліку та аналізу особистих фінансів включає різні процеси, що об'єднуються у контекстній діаграмі IDEF0. Від реєстрації та аутентифікації користувача до генерації звітів та візуалізації результатів – кожен етап відіграє важливу роль. Система, будучи належно спроектованою, забезпечить ефективний інструмент для обліку та аналізу особистих фінансів, що допоможе користувачам краще управляти своїми фінансами та приймати обґрунтовані фінансові рішення [10, 23].

### 1.5. Архітектура веборієнтованої системи

Створення ефективної, безпечної та зручної для користувача системи потребує ретельного вибору архітектури та технологій реалізації. Вибір повинен бути здійснений на основі ретельного аналізу сформованих вимог.

Розробка архітектури сайту є критично важливою, оскільки вона допомагає впорядкувати структуру інформації, що покращує навігацію та доступність [30]. Правильно спроектована архітектура сайту сприяє більш ефективному SEO, оскільки пошукові системи легше індексують організований контент. Крім того, вона створює чіткий шлях для майбутнього розвитку та масштабування сайту, зменшуючи потребу в змінах [12].

Щоб створити архітектуру спроектованої системи, слід вивчити існуючий

та потенційний контент, який буде розміщено на сторінках сайту. Крім того, важливо структурувати вебсторінки так, щоб система використовувала свої ресурси ефективно [39]. Однак ключовим етапом є розробка схеми навігації, яка слугує картою майбутнього вебсайту, допомагає оптимізувати розподіл функціонального та інформаційного навантаження сайту і його окремих сторінок [19].

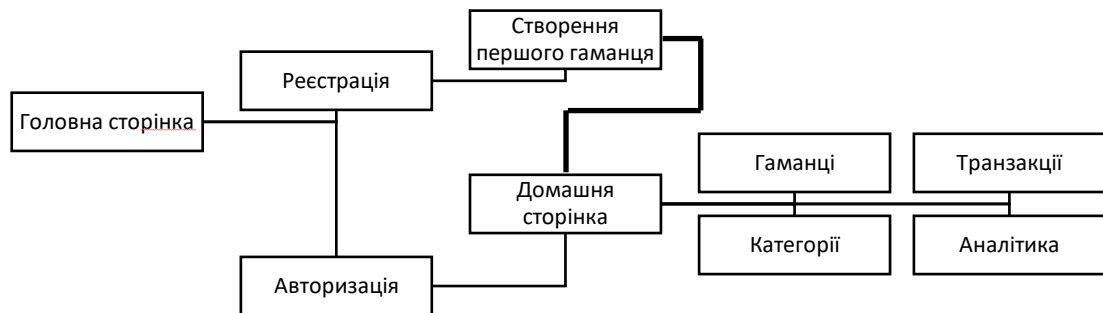


Рисунок 1.6 – Навігаційна схема веборієнтованої системи обліку та аналізу особистих фінансів

Вебсайт буде мати розгалужену структуру сторінок, як показано на рисунку 5.1. Така структура є оптимальною для чіткого представлення контенту сайту і не повинна створювати проблем з навігацією для користувачів. Домашню сторінку зможуть бачити лише зареєстровані користувачі, в той час як незареєстрованим користувачам буде запропоновано зареєструватись на сайті, а потім створити перший власний гаманець з певною валютою.

Для візуалізації проєктованої системи та систематизації всіх її вебелементів було створено структурний макет вебсайту – wireframe, що є схематичним відображенням функціональності сторінок та їх контенту.

Balsamiq Wireframes – це швидкий інструмент для створення макетів, що допомагає дизайнерам та розробникам дозволяє створювати макети, що імітують ескіз на папері. Balsamiq пропонує широкий вибір предвстановлених компонентів, які можна перетягнути та налаштувати за допомогою його

інтуїтивного інтерфейсу. Згенеровані макети можна експортувати в різні формати для подальшого використання або презентацій [1].

Домашня сторінка – це найбільш значуща для користувача сторінка вебсайту. Це та сторінка, яку користувач бачить першою після входу на сайт. Вона повинна лаконічно і повноцінно відображати його поточні фінансові показники. Така сторінка має бути обладнана зручними навігаційними інструментами, які дозволять навіть не дуже досвідченому користувачу швидко ознайомитися з призначенням та контентом інших сторінок. На рисунку 1.7 показано прототип домашньої сторінки вебсайту, де видно, що навігаційне меню розміщене в верхній частині сторінки і дозволяє перейти на всі інші сторінки сайту. Тут також розміщена кнопка для переходу на сторінку особистого кабінету та кнопка створення нової транзакції.

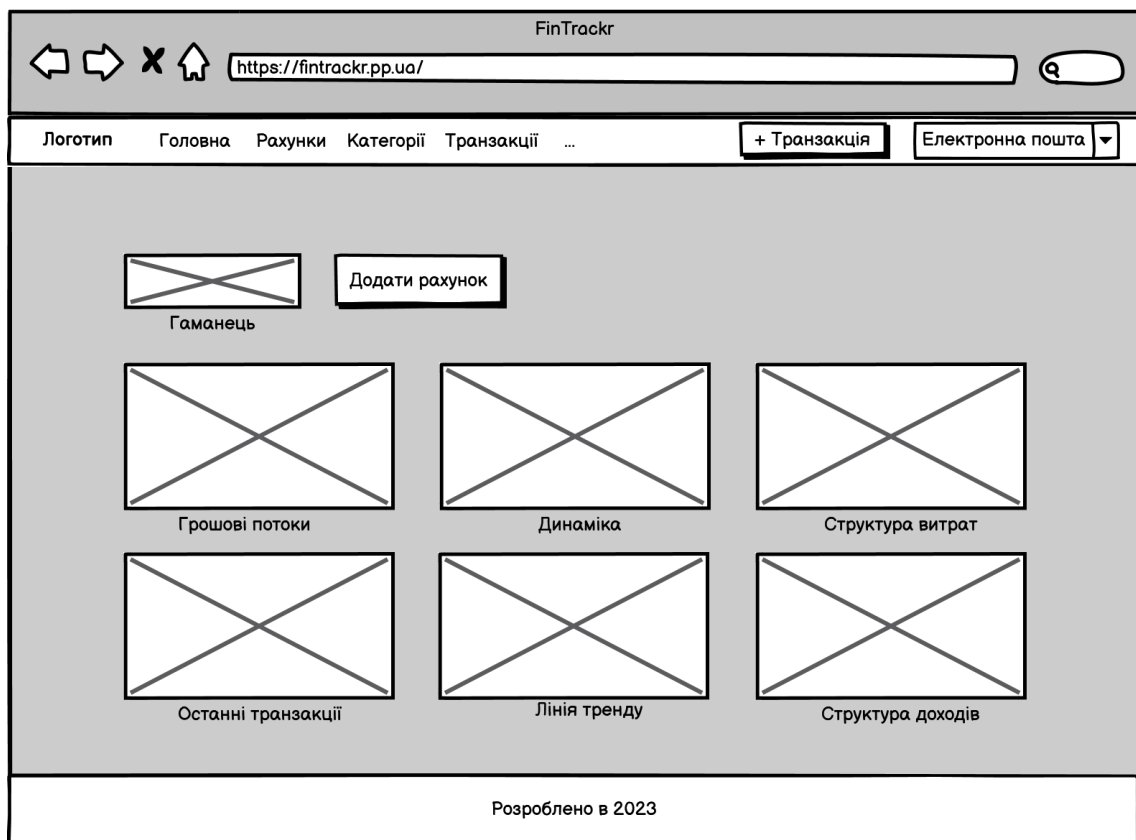


Рисунок 1.7 – Прототип домашньої сторінки вебсайту

Сучасний ринок технологій пропонує величезний вибір різноманітних мов програмування, фреймворків та інструментів для розробки веборієнтованих систем. Кожна з них має свої переваги та недоліки, і вибір

однієї або іншої технології часто залежить від специфічних вимог до проєкту та від досвіду команди розробників.

У процесі проєктування та реалізації веборієнтованих систем, структурна організація програмного коду відіграє важливу роль. Одним з фундаментальних аспектів такої організації є розділення системи на дві основні частини: серверна частина та клієнтська частин [20, 33].

Серверна частина (бекенд) – це частина системи, що відповідає за обробку вхідних запитів, виконання відповідних обчислень або маніпуляцій з даними, а потім відправлення відповідей назад. Бекенд є фундаментом, на якому базується функціонал системи. Його основним завданням є забезпечення надійного та безпечного обміну даними, а також маніпуляції з базою даних [19].

Щоб запустити вебсайт або додаток, потрібен сервер, на якому працює серверне програмне забезпечення. До популярних серверних програм належать Apache HTTP Server, Microsoft IIS та Nginx. Ці програми виконують такі завдання на стороні сервера, як розміщення вебсторінок, прийом запитів користувачів і відправку відповідей [22].

Слід також зазначити, що бекенд часто включає в себе інші компоненти, такі як бази даних, API, а також логіку безпеки та авторизації. Усі ці компоненти вимагають відповідного програмного забезпечення для коректної роботи.

Існує багато мов програмування, які можна використовувати для розробки вебсайтів. Кожна мова має свої сильні та слабкі сторони, тому дуже важливо вибрати правильну мову для проєкту. Ми зосередимося на розробці з Python.

Python – це високорівнева мова програмування з простим синтаксисом і читабельним кодом, що робить його відмінним вибором для новачків і експертів. Він підтримує широкий спектр застосувань, включаючи веброзробку, аналіз даних, машинне навчання і автоматизацію. Велика і активна спільнота постійно створює і підтримує множину корисних бібліотек

і фреймворків [35].

Django – це основний фреймворк для розробки вебдодатків на Python, який надає комплексні функції "з коробки", що підвищують швидкість розробки і сприяють безпеці програмного забезпечення. Django підтримує високо навантажені додатки та пропонує відмінні можливості для масштабування. Його філософія DRY (Don't Repeat Yourself) заохочує повторне використання коду та сепарацію компонентів, що покращує читабельність та упорядкованість коду [13].

Важливо також зазначити необхідність ефективного розгортання та управління додатками, такими як сама програма, написана на Python, та база дани. Docker – це платформа, що дозволяє розробникам "контейнеризувати" свої додатки разом з їх налаштуваннями. Контейнери Docker надають універсальне середовище, яке гарантує, що програмне забезпечення буде працювати однаково, незалежно від того, де воно запускається [14].

Щодо вибору СУБД, то зупинимося на PostgreSQL. По-перше, PostgreSQL підтримує широкий спектр стандартних функцій SQL, що надає велику гнучкість при виконанні запитів до бази даних. По-друге, він має високу надійність та стабільність. По-третє, PostgreSQL підтримує розширені функції, такі як вбудовані та користувацькі типи даних, тригери та збережені процедури [6]. Він також може бути легко інтегрований з різними технологіями, включаючи Django та Docker.

Клієнтська частина (фронтенд), в свою чергу, є частиною системи, з якою безпосередньо взаємодіє користувач. Вона включає в себе візуальне представлення даних, інтерфейси взаємодії та усі елементи, які користувач бачить на екрані свого пристрою. Фронтенд забезпечує приємний користувацький досвід, виконуючи роль посередника між користувачем та бекендом [15, 32].

Синтаксис шаблонів Django був розроблений таким чином, щоб бути інтуїтивно зрозумілим та зручним у використанні, незалежно від того, є ви досвідченим розробником чи новачком. У контексті створення

веборієнтованої системи обліку та аналізу особистих фінансів, клієнтська частина, буде реалізована за допомогою Django Template Language.

Django Template Language (DTL) — це мова шаблонів Django, яка надає гнучкість і потужність для візуалізації даних у вебдодатку. Це вбудована система шаблонів Django дозволяє використовувати HTML, CSS і JavaScript для створення багатофункціонального та візуально привабливого інтерфейсу користувача [7].

Основні компоненти DTL:

– змінна: виводяться в фігурних дужках, наприклад: `{{ variable }}`. Коли шаблон відображається, він замінюється значенням змінної;

– тег: дозволяють виконувати різноманітні операції, такі як цикли, умовні оператори, завантаження статичних файлів та інше. Теги вказуються у фігурних дужках з використанням відсоткового знаку, наприклад: `{% tag %}`;

– фільтр: дозволяють змінювати змінні в шаблонах. Вони використовуються після змінної і розділяються символом `|`. Наприклад, ви можете використовувати фільтр для перетворення тексту в нижній регістр: `{{ variable|lower }}` [7].

Отже, правильно розроблена архітектура сайту та ефективне використання технологій, таких як Django, Python і Docker, та використання надійних баз даних, таких як Postgres, можуть забезпечити якісне функціонування сайту і його сумісність з різними пристроями. При цьому створення вебдодатків з використанням Django Template Syntax дає змогу створювати багатофункціональні інтерфейси, що значно полегшує розробку серверної частини.

## РОЗДІЛ 2. РЕАЛІЗАЦІЯ ПРОТОТИПУ ВЕБОРІЄНТОВАНОЇ СИСТЕМИ

### 2.1. Структура та особливості реалізації інформаційного забезпечення

Інформаційне забезпечення є ключовим компонентом автоматизованих облікових інформаційних систем. Його завданням є відтворення інформації, яка описує стан об'єкта управління, та слугує основою для прийняття керівничих рішень [43].

Опис структури та особливостей реалізації інформаційного забезпечення необхідний для розуміння того, як система збирає, зберігає, обробляє та використовує дані. Ця інформація допомагає у визначенні ефективності, надійності та безпеки системи, а також допомагає в процесі її удосконалення та масштабування.

Розроблена веборієнтована система обліку та аналізу особистих фінансів використовує СУБД PostgreSQL, яка була вибрана з огляду на її широкі можливості та високий рівень безпеки.

Перегляд бази даних відбувається за допомогою терміналу Docker контейнера PostgreSQL (рис. 2.1). Docker дозволяє виконувати PostgreSQL у ізольованому середовищі, що полегшує процес розгортання та управління базою даних (рис. Б.1). Крім того, це дозволяє забезпечити однакові умови роботи на різних платформах та машинах, що сприяє загальній надійності та стабільності системи [14].

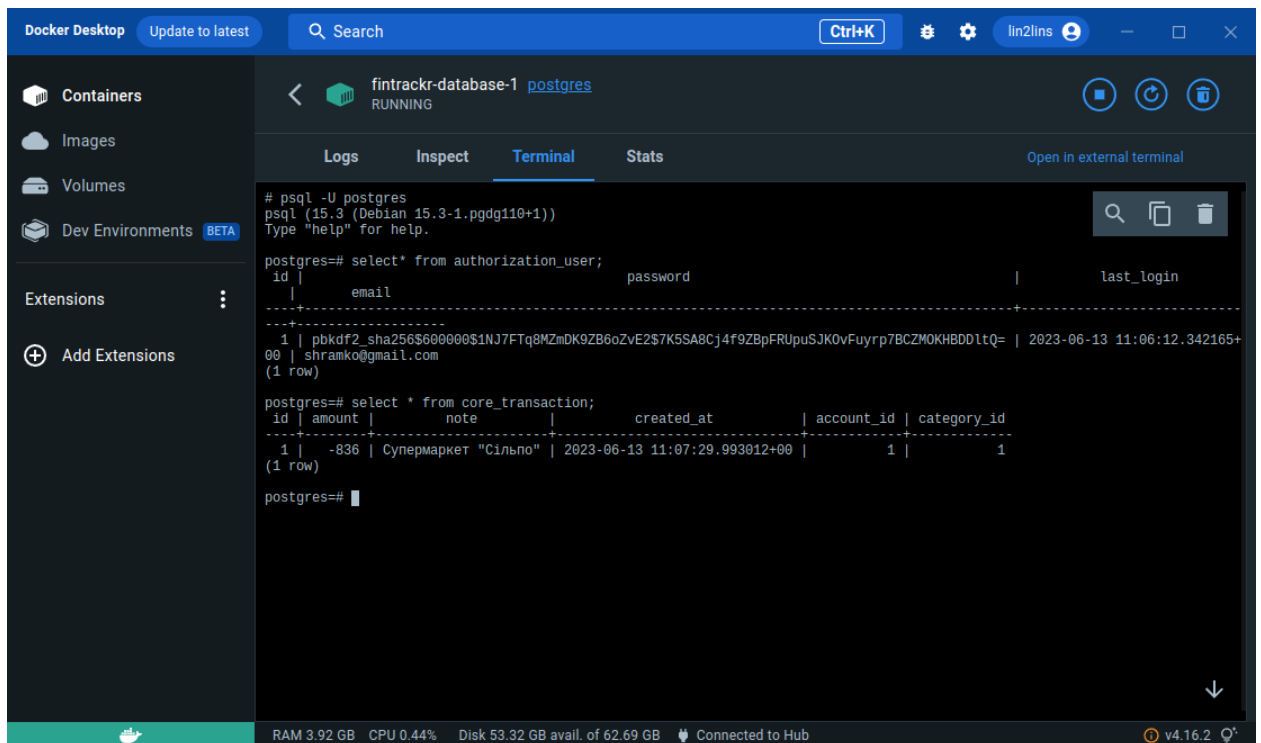


Рисунок 2.1 – Перегляд таблиць бази даних через термінал застосунку Docker Desktop

В контексті розробки веборієнтованої системи обліку та аналізу особистих фінансів, для взаємодії з базою даних PostgreSQL використовується Django ORM (об'єктно-реляційне відображення). Django ORM дозволяє розробникам виконувати запити до бази даних використовуючи Python, що спрощує роботу з даними і забезпечує високий рівень абстракції (рис 2.2) [12].

```

2 usages  Elina Shramko
40 class AccountListView(LoginRequiredMixin, View):
41     login_url = reverse_lazy("login")
42     template_name = "core/account.html"
43     success_url = reverse_lazy("account")
44
45     Elina Shramko
46     def get(self, request):
47         form = AccountForm()
48         user_accounts = Account.objects.filter(user=request.user).all()
49         return render(request, self.template_name, {"accounts": user_accounts, "form": form})

```

Рисунок 2.2 – Використання Django ORM для отримання рахунків користувача

Для візуального представлення та розуміння структури бази даних була використана ER-діаграма, що є графічним засобом, який ілюструє структуру



бази даних, включаючи таблиці (або "сутності"), їх атрибути та зв'язки між ними (рис. 2.3). Це є стандартним методом опису структури бази даних, який дозволяє визначити та розглянути сутності, їх атрибути, а також різноманітні взаємозв'язки [5].

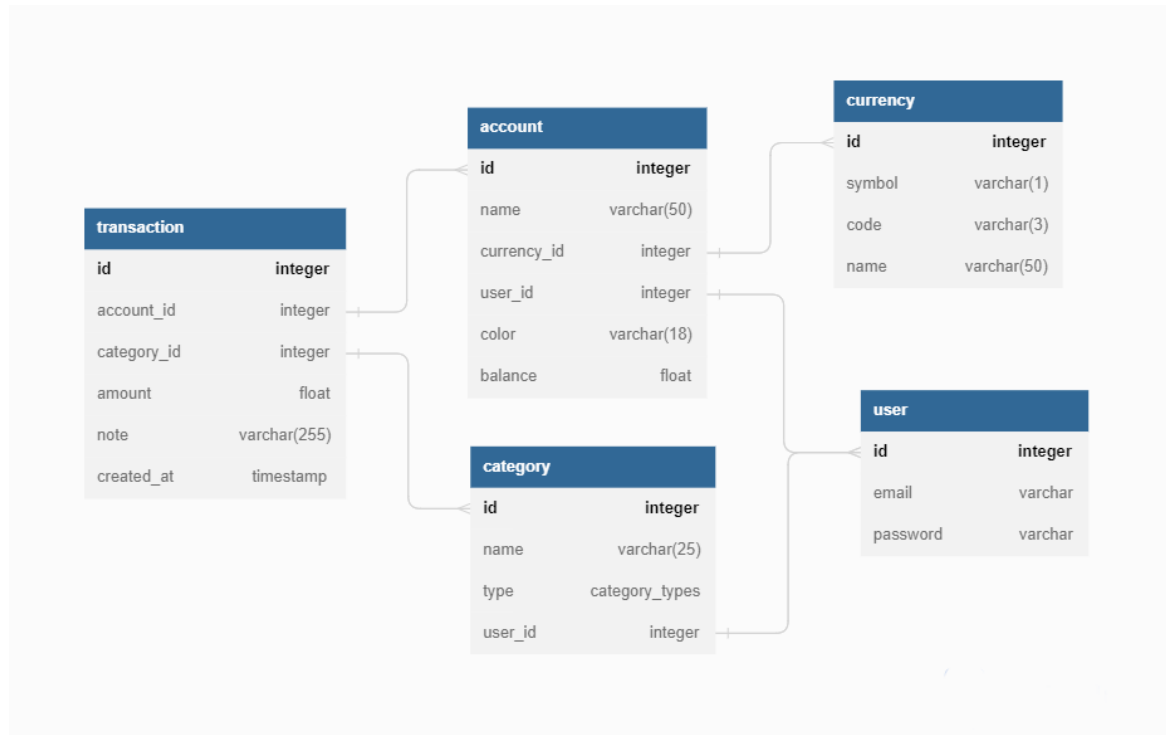


Рисунок 2.3 – ER-діаграма бази даних веборієнтованої системи обліку та аналізу особистих фінансів

Ось короткий опис структури:

1. Таблиця "user": ця таблиця зберігає інформацію про користувачів. У кожного користувача є унікальний ідентифікатор (id), електронна адреса (email) та пароль (password).

2. Таблиця "category": тут зберігається інформація про категорії фінансових транзакцій. Кожна категорія має унікальний ідентифікатор (id), назву (name), тип (type) та ідентифікатор користувача (user\_id), що посилається на таблицю "user".

3. Таблиця "transaction": ця таблиця містить інформацію про конкретні фінансові транзакції. Кожна транзакція має унікальний ідентифікатор (id), ідентифікатор рахунку (account\_id), що посилається на таблицю "account",

ідентифікатор категорії (`category_id`), що посилається на таблицю "category", суму транзакції (`amount`), нотатку (`note`), та час створення (`created_at`).

4. Таблиця "currency": ця таблиця включає інформацію про валюту. Кожна валюта має унікальний ідентифікатор (`id`), символ (`symbol`), код (`code`) та назву (`name`).

5. Таблиця "account": тут зберігається інформація про рахунки користувачів. Кожен рахунок має унікальний ідентифікатор (`id`), назву (`name`), ідентифікатор валюти (`currency_id`), що посилається на таблицю "currency", ідентифікатор користувача (`user_id`), що посилається на таблицю "user", баланс (`balance`) та колір (`color`).

В рамках розробленої системи обліку та аналізу особистих фінансів, основними джерелами вхідної інформації є користувачі системи. Вони активно взаємодіють із системою, вводячи необхідні персональні дані та інформацію про свої фінансові операції. Сюди входить реєстрація та авторизація в системі, створення та управління рахунками, введення даних про доходи та витрати, а також подальше керування цими даними.

Ці введені користувачами дані переходять до внутрішнього середовища системи, де вони зберігаються, обробляються та аналізуються відповідно заданого алгоритму.

Щодо інформації про валюту, система обліку та аналізу особистих фінансів включає в себе передвизначені категорії валют: гривня, долар та євро. Ця інформація є статичною і не підлягає зміні користувачами. Вона служить базовим елементом для виконання фінансових обчислень та аналізу в рамках системи.

Вихідні дані, відображені користувачам, представляють собою результати аналізу введених фінансових даних. Вони включають графіки, діаграми та інші форми візуалізації, які ілюструють різноманітні аспекти фінансового стану користувача.

## 2.2. Структура та особливості реалізації алгоритмічного забезпечення

Алгоритмічне забезпечення – це сукупність математичних алгоритмів, що використовуються в системі для вирішення задач та обробки інформації [43]. Інакше кажучи, це набір специфікацій та описів алгоритмів, які використовуються зокрема у веборієнтованій системі. Одним із способів наглядного представлення цих алгоритмів є використання блок-схем.

Блок-схема – це графічне зображення послідовності процесів, яке допомагає зрозуміти зв'язок між рішеннями та кінцевим результатом. Цей інструмент може бути використаний для опису різних процесів, таких як виробничий процес, адміністративний процес або процес надання послуг. Блок-схеми використовують прості геометричні фігури для представлення кроків процесу, прийняття рішень, збереження даних та виходу з процесу [42].

Опис структури та особливостей реалізації алгоритмічного забезпечення є важливим, оскільки це сприяє кращому розумінню та аналізу логіки роботи системи. Деталізація алгоритмів дозволяє ефективніше управляти процесом розробки, виконувати оптимізацію та усунення помилок. Крім того, якісний опис алгоритмічного забезпечення полегшує процес тестування та впровадження системи.

Алгоритм роботи з веборієнтованою системою обліку та аналізу особистих фінансів для нового користувача має наступний вигляд:

1. Реєстрація: користувач вводить свої особисті дані (ім'я, електронна пошта, пароль тощо). Далі система перевіряє введені дані на правильність і унікальність. Якщо дані коректні, створюється новий користувач у системі.

2. Створення першого гаманця: користувач вводить назву гаманця та початковий баланс, а потім гаманець зберігається в системі і стає доступним для подальшої роботи.

3. Налаштування категорій: користувач має можливість налаштувати категорії для класифікації своїх транзакцій. Система надає типові категорії, які користувач може змінити або додати нові.

4. Створення транзакції: користувач має можливість додавати нові транзакції до свого гаманця. Він обирає тип транзакції (дохід або витрата), вводить суму транзакції та обирає підкатегорію. Система зберігає транзакцію та оновлює баланс гаманця.

5. Перегляд головної сторінки з аналітикою: користувач має можливість переглянути головну сторінку системи, на якій відображаються аналітичні дані по його рахунку. Система показує інформацію про поточний баланс гаманця, статистику доходів та витрат за певний період.

Алгоритм роботи з веборієнтованою системою обліку та аналізу особистих фінансів для існуючого користувача відрізнятиметься тільки тип, що користувач замість реєстрації виконує авторизацію і йому не потрібно створювати перший гаманець, що є необхідним для нового користувача (рис. В.1).

Важливо відмітити, що у розробленій системі користувач може створювати необмежену кількість рахунків, обираючи одну з трьох валют. Це дозволяє користувачеві вести облік своїх фінансів в різних грошових одиницях, що є корисним функціоналом для осіб, що мають різні рахунки або ведуть фінансову діяльність у різних країнах.

Загальний опис алгоритму використання сайту для обліку та аналізу особистих фінансів може бути описаний такою блок-схемою (рис. 2.3):

1. Користувач відкриває вебсайт.
2. Користувач вводить свої дані для входу в систему.
3. Користувач переходить до на свою домашню сторінку.
4. Користувач вибирає одну з доступних дій на навігаційній панелі.
5. Відповідно до вибору користувача виконується відповідна дія.

Наприклад, якщо користувач вибрав "Додати транзакцію", він переходить до форми додавання транзакції, заповнює необхідні поля та надсилає форму.

Важливо звернути увагу, що реальний алгоритм може включатиме подальші кроки залежно від вибору користувача.

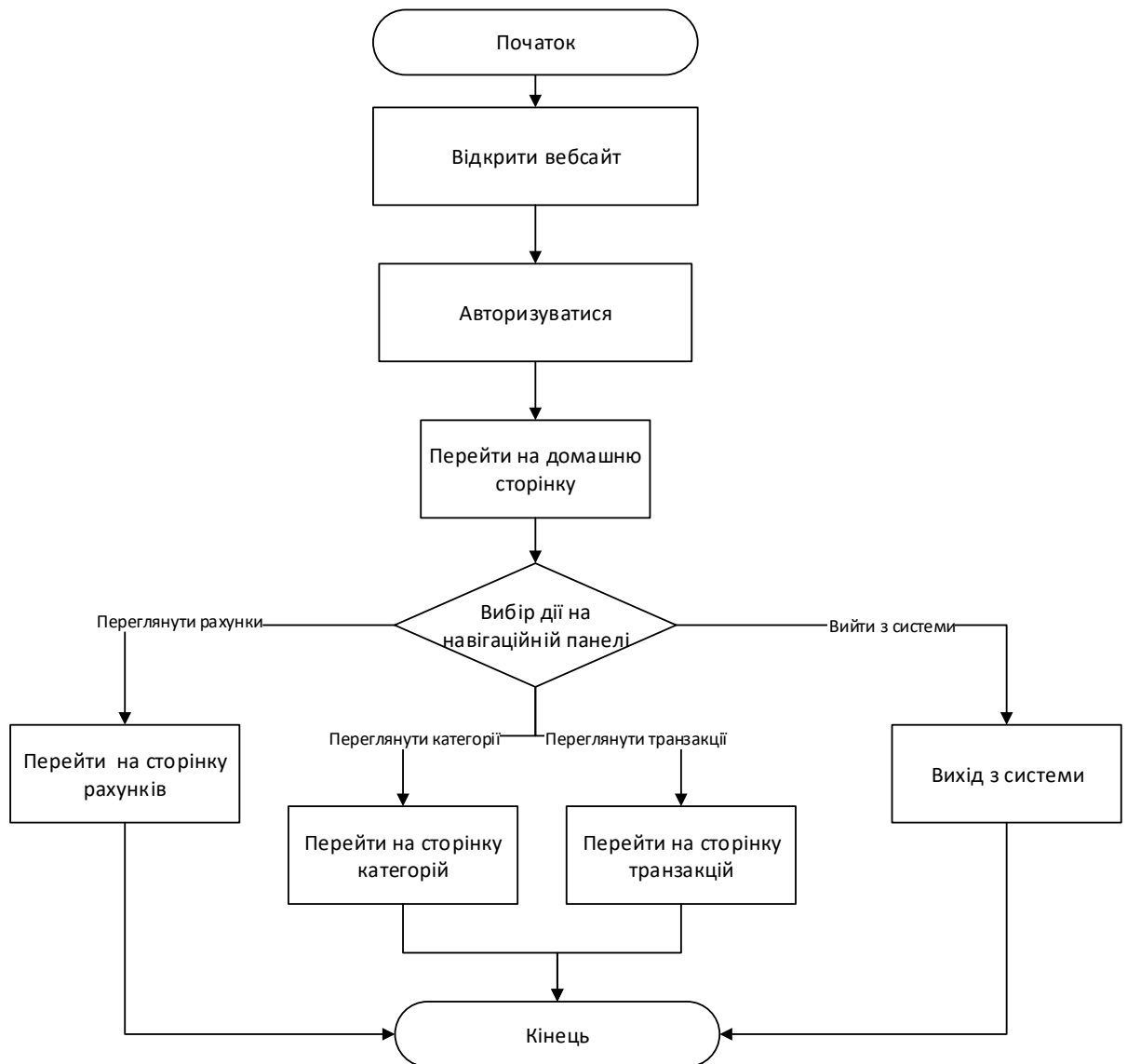


Рисунок 2.4 – Блок-схема використання сайту обліку та аналізу особистих фінансів

Розглянемо приклад, наведений вище в контексті роботи серверної частини сайту. Процес створення транзакції на вебсайті для обліку та аналізу особистих фінансів починається з того, що система отримує запит на створення транзакції. Далі вона відображає форму створення транзакції. Коли користувач заповнює та відправляє форму, система отримує введені дані, виконує необхідні перевірки, вносить запис до бази даних, оновлює баланс відповідного рахунку та діаграми. Після цього система перенаправляє користувача на головну сторінку, щоб він побачив нову транзакцію (рис. В.2).

### 2.3. Контрольний приклад та інструкція по використанню

Розглянемо приклад створення користувачем транзакції, розібраний в контексті серверної частини сайту. Приклад включає активність зареєстрованого користувача (рис. Г.1-Г.3), що бажає записати нову фінансову транзакцію в системі.

Перший крок включає авторизацію. Користувачу потрібно зайти на сайт, ввести електронну пошту й пароль та натиснути кнопку «Увійти» (рис. 2.5).

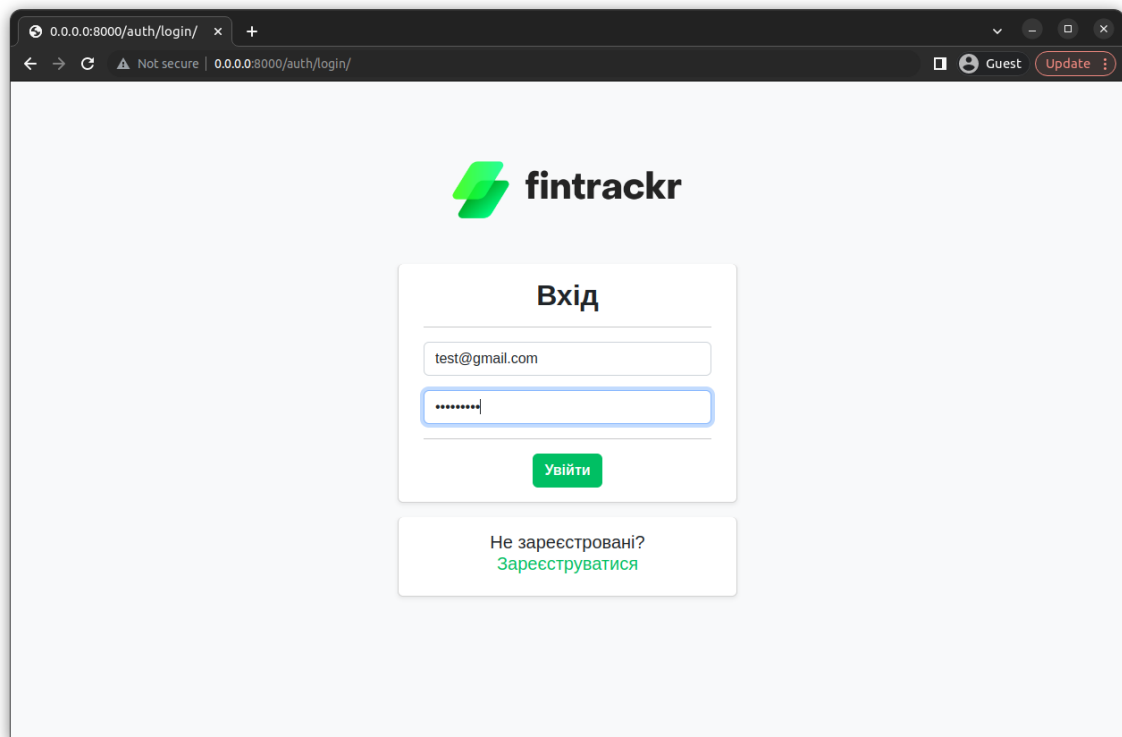


Рисунок 2.5 – Сторінка авторизації

Другий крок включає вибір користувачем опції «+Транзакція» в навігаційному меню сайту. Після цього відкривається форма додавання транзакції, де користувач вводить деталі транзакції: тип (дохід або витрата), суму, дату і категорію. Після заповнення полів користувач натискає кнопку «Створити». Потім система перевіряє правильність введеної інформації, переконуючись, що всі дані введені вірно і відповідають вимогам формату. Як бачимо на рисунку 2.6, відображається помилка про пусте необхідне поле.

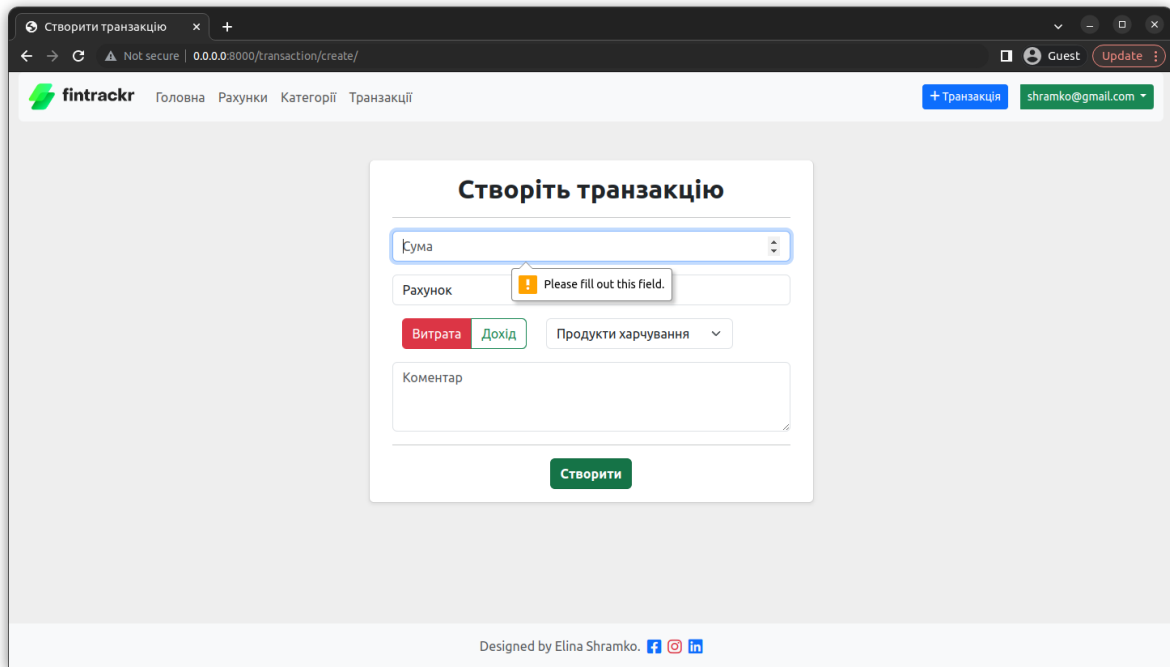


Рисунок 2.6 – Сторінка створення транзакції з помилкою вводу

Вводимо суму в незаповнене поле та натискаємо кнопку «Створити» (рис. 2.7).

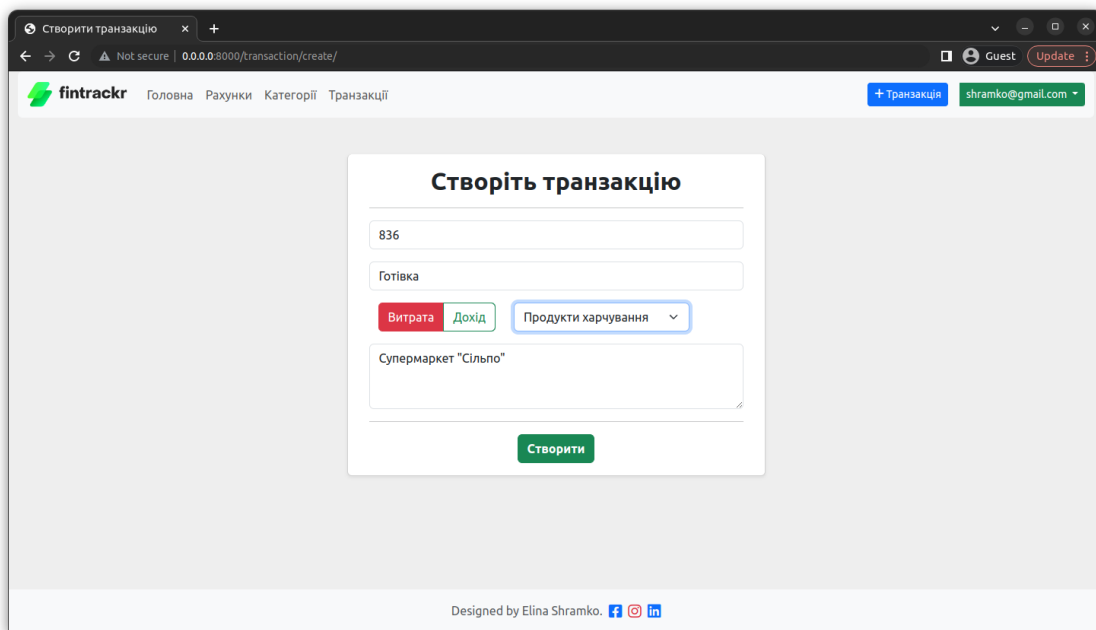


Рисунок 2.7 – Сторінка створення транзакції

Якщо перевірка пройшла успішно, система зберігає запис про транзакцію в базі даних. Користувачу відображається головна сторінка з

новою транзакцією, що підтверджує успішне виконання операції (рис. 2.8).

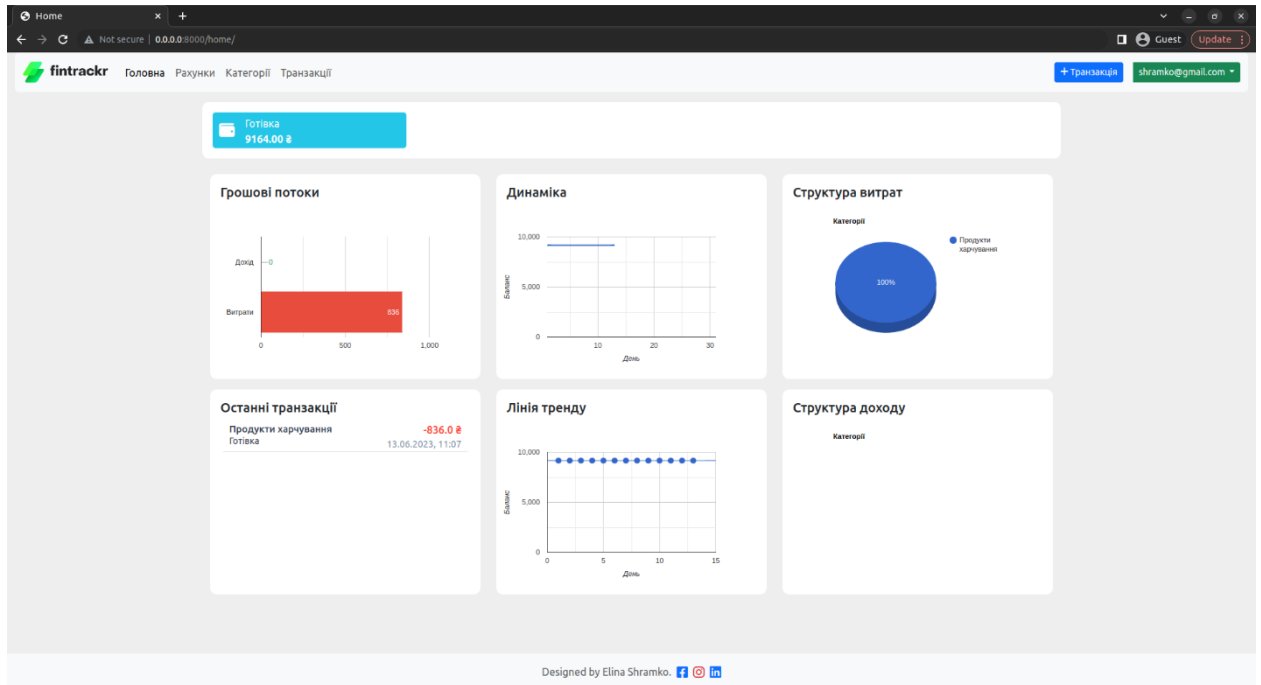


Рисунок 2.8 – Домашня сторінка зі створеною транзакцією

Для того, щоб система могла надати користувачеві значущу та відображаючу реальну ситуацію аналітику особистих фінансів, необхідно ввести достатню кількість транзакцій. Це дозволить системі аналізувати тенденції, виявляти відхилення та патерни в фінансовій поведінці користувача. Чим більше транзакцій користувач введе, тим більш детальною і точною буде аналітика, надана системою.



## ВИСНОВКИ

Важливим етапом для економічного розвитку та фінансового добробуту сучасних людей є контроль та аналіз особистих фінансів. Використання цифрових технологій у цій сфері відкриває нові можливості для оптимізації, контролю та прогнозування фінансової діяльності користувачів. Вебсайт для обліку та аналізу особистих фінансів значно спрощує процеси відстеження доходів та витрат, допомагає ідентифікувати та проаналізувати фінансові звички та тенденції, а також приймати обґрунтовані фінансові рішення.

У процесі виконання випускної кваліфікаційної роботи виконано всі поставлені задачі:

1. Складено характеристику об'єкта дослідження.
2. Проаналізовано стан автоматизації процесів обліку та аналізу особистих фінансів. Проведено короткий огляд безкоштовних систем управління особистими фінансами.
3. Сформульовано та обґрунтовано вимоги до веборієнтованої системи.
4. Побудовано контекстну діаграму та декомпозицію процесів обліку та аналізу особистих фінансів в методології IDEF0.
5. Розроблено архітектуру веборієнтованої інформаційної системи, її функціональну структуру підсистеми забезпечення функціональної частини та технології вирішення поставлених завдань.
6. Розглянуто структуру та особливості реалізації інформаційного та алгоритмічного забезпечення, відображено діяльність сайту у вигляді блок-схеми.
7. Розроблено веборієнтовану систему на базі локального сервера. При розробці вебсайту використано наведені бібліотеки та програми.
8. Написана інструкція щодо використання інформаційної системи.

Розроблений сайт представляє собою веборієнтовану систему обліку та аналізу особистих фінансів, яка надає користувачам зручний інтерфейс для

управління своїми фінансами. Система дозволяє користувачам створювати рахунки, додавати, редагувати та видаляти транзакції, а також визначати категорії доходів та витрат. Завдяки вбудованим аналітичним інструментам, користувачі можуть слідкувати за своїми фінансовими трендами та отримувати деталізовані звіти. Сайт підтримує адаптивний дизайн, що забезпечує зручне використання на різних пристроях. Усі дані користувачів захищені та шифруються для забезпечення приватності та безпеки. Реалізований функціонал включає можливість створення рахунків, категорій та транзакцій, візуалізації даних через графіки та діаграми, аналізу фінансових трендів та відхилень. В загальному, цей вебсайт є потужним інструментом для особистого фінансового менеджменту.

В подальшому планується реалізувати наступні покращення:

- 1) реалізація більшої кількості способів вводу фінансової інформації, наприклад, імпорт даних з банківських карток;
- 2) впровадження більшої кількості аналітичних діаграм;
- 3) впровадження сутності «бюджет» для ефективного планування особистих фінансів;
- 4) розробка можливості експорту фінансової інформації та аналітичних звітів у форматах PDF, MS Word тощо.

16.06.2023.



## СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. 7 Lessons We've Learned Designing SaaS Websites. Balsamiq. URL: <https://balsamiq.com/learn/articles/saas-website-design-lessons/> (Last accessed: 12.05.2023).
2. A Guide to the Business Analysis Body of Knowledge (BABOK Guide). International Institute of Business Analysis. Toronto, Ontario, Canada. 253 p. URL: [https://www.academia.edu/42605272/A\\_Guide\\_to\\_the\\_Business\\_Analysis\\_Body\\_of\\_Knowledge\\_BABOK\\_Guide](https://www.academia.edu/42605272/A_Guide_to_the_Business_Analysis_Body_of_Knowledge_BABOK_Guide) (Last accessed: 03.05.2023).
3. AltexSoft. Functional and Nonfunctional Requirements: Specification and Types. 2021. URL: <https://www.altexsoft.com/blog/business/functional-and-non-functional-requirements-specification-and-types/> (Last accessed: 02.05.2023).
4. AltexSoft. How to Write a Business Requirements Document: Guidelines, Templates, and Useful Tips. 2021. URL: <https://www.altexsoft.com/blog/business-requirements-document/> (Last accessed: 02.05.2023).
5. Bagui S., Earp R. Database Design Using Entity-Relationship Diagrams. – 2<sup>nd</sup> ed. – CRC Press, 2011. – 362 p.
6. Beginning Databases with PostgreSQL: From Novice to Professional Academia.edu. URL: [https://www.academia.edu/19624323/Begining\\_Databases\\_with\\_PostgreSQL\\_from\\_novice\\_to\\_Profesional](https://www.academia.edu/19624323/Begining_Databases_with_PostgreSQL_from_novice_to_Profesional) (Last accessed: 10.05.2023).
7. Beginning Django: Web Application Development and Deployment with Python. Covers 1.11 LTS compatible with Python 2 and 3. Academia.edu. URL: [https://www.academia.edu/39620108/Beginning\\_Django\\_Web\\_Application\\_Development\\_and\\_Deployment\\_with\\_Python\\_Covers\\_1\\_11\\_LTS\\_compatible\\_with\\_Python\\_2\\_and\\_3\\_Daniel\\_Rubio/](https://www.academia.edu/39620108/Beginning_Django_Web_Application_Development_and_Deployment_with_Python_Covers_1_11_LTS_compatible_with_Python_2_and_3_Daniel_Rubio/) (Last accessed: 10.05.2023).
8. Best Business Process Modeling Techniques (With Examples). URL: <https://tallyfy.com/business-process-modeling-techniques/> (дата звернення: 09.05.2023).
9. BudgetBakers: Your Money on Track. URL: <https://budgetbakers.com/what-is-wallet/> (Last accessed: 26.04.2023).
10. C. L. Ang, M. Luo, L.P. Khoo & R.K. L. Gay (1997) A knowledge-based

approach to the generation of IDEF0 models, *International Journal of Production Research*, 35:5, 1385-1412

11. Capturing Functional Requirements with Use Cases and User Stories. Cybermedia. URL: <https://www.cybermedian.com/capturing-functional-requirements-with-use-cases-and-user-stories/> (Last accessed: 05.05.2023).

12. Dissanayake, N. R., Dias, K. A. Web-based Applications: Extending the General Perspective of the Service of Web. 10th International Research Conference of KDU on Changing Dynamics in the Global Environment: Challenges and Opportunities. Rathmalana, Sri Lanka. (August 3-4. 2017, Sri Lanka).

13. Django Documentation: Django 4.2. Django Software Foundation, URL: <https://docs.djangoproject.com/en/4.2/> (Last accessed: 08.05.2023).

14. Docker overview. Docker Documentation URL: <https://docs.docker.com/get-started/overview> (Last accessed: 09.05.2023).

15. Front End Developer: What is Front-End Development? (Explained in Plain English) freeCodeCamp.org. URL: <https://www.freecodecamp.org/news/front-end-developer-what-is-front-end-development-explained-in-plain-english/> (Last accessed: 09.05.2023).

16. Functional vs Non Functional Requirements: What's the difference? Guru99. URL: <https://www.guru99.com/functional-vs-non-functional-requirements.html> (Last accessed: 06.05.2023).

17. Google Developers. Google Charts. URL: <https://developers.google.com/chart/interactive/docs> (Last accessed: 15.05.2023).

18. IDEF0 - Part 1 (understanding it). URL: [http://syque.com/quality\\_tools/tools/Tools19.htm](http://syque.com/quality_tools/tools/Tools19.htm) (Last accessed : 06.05.2023)

19. Information Architecture in UX: The Blueprint of User Experience. AltexSoft. URL: <https://www.altexsoft.com/blog/uxdesign/information-architecture/> (Last accessed: 11.05.2023).

20. Koh M. Anatomy of a website: website architecture, 2016. URL: <https://blog.optimalworkshop.com/anatomy-website-website-architecture/> (Last accessed: 16.05.2023).

21. Monefy. URL: <https://monefy.me/> (Last accessed: 26.04.2023).
22. Money Manager. URL: <https://realbyteapps.com/> (Last accessed: 26.04.2023).
23. Presley, A., Liles, D.H. The use of IDEF0 for the design and specification of methodologies, in: Proceedings of the 4th industrial engineering research conference. 1995.
24. Report: State of Mobile 2023 – Mobile banking comes of age. 2023. URL: <https://www.paymentscardsandmobile.com/report-state-of-mobile-2023-mobile-banking-comes-of-age/> (Last accessed: 25.04.2023).
25. Šerifi, V.; Dašić, P.; Ječmenica, R. and Labović, D. Functional and Information Modeling of Production Using IDEF Methods. *Strojniški vestnik – Journal of Mechanical Engineering*. №55. 2009. p. 131-140.
26. State Of Mobile 2023. URL: <https://www.data.ai/en/go/state-of-mobile-2023/> (Last accessed: 24.04.2023).
27. Stellman. Requirements 101: User Stories vs. Use Cases. URL: <https://www.stellman-greene.com/2009/05/03/requirements-101-user-stories-vs-use-cases/> (Last accessed: 05.05.2023).
28. T. Munohsamy. Personal Financial Management. Institut Teknologi URL: [https://www.researchgate.net/publication/279198054\\_Personal\\_Financial\\_Management/](https://www.researchgate.net/publication/279198054_Personal_Financial_Management/) (Last accessed: 22.04.2023).
29. Tangkawarow, J. Waworuntu. A Comparative of business process modelling technique. International Conference on Innovation in Engineering and Vocational Education. 2016.
30. Web Application Architecture: A Comprehensive Guide On The What, Why And How, 2021. URL: <https://www.intuz.com/guide-on-web-app-architecture> (Last accessed: 16.05.2023).
31. What is a Flowchart? Process Flow Diagrams & Maps. URL: <https://asq.org/quality-resources/flowchart>
32. What Is a Front-End Developer? URL : <https://frontendmasters.com/books/front-end-handbook/2018/what-is-a-FD.html>

(Дата звернення: 14.05.2023)

33. What is Full Stack Development? Simplilearn. URL: <https://www.simplilearn.com/what-is-full-stack-development-article> (Last accessed: 08.05.2023).

34. What is User Story in Agile Software Development? Visual Paradigm Guide. URL: <https://www.visual-paradigm.com/guide/agile-software-development/what-is-user-story/> (Last accessed: 05.05.2023).

35. Why Use Python for Web Development? STX Next. URL: <https://www.stxnext.com/blog/python-for-web-development/> (Last accessed: 09.05.2023).

36. Гончар Л.В., Гарна С.О., Мартиненко А.О. Фінансова грамотність населення України як фактор успішного розвитку держави. Інноваційна економіка. 2019. №1-2. С. 182-186.

37. Кізима Т. В., Куцяк. В. О. Особисте фінансове планування: сутність, специфіка, основні етапи. Світ фінансів. 2015. №3. С. 62-70.

38. Мітал О. Г. Заощадження домогосподарств та їх трансформація в інвестиційні ресурси. Європейський вектор економічного розвитку. 2017. № 1. С. 70-78.

39. Основні етапи створення сайту. URL: <https://pbb.lviv.ua/stattiinovyny/statti-shchodo-stvorennia-saitu/osnovni-etapy-stvorennia-saitu> (Дата звернення: 17.05.2023)

40. Пістунов І. М. Моделювання бізнес процесів: навчальний посібник / І.М. Пістунов. – Д.: НТУ «ДП», 2021. – 130 с.

41. Про захист персональних даних. Режим доступу: <http://zakon4.rada.gov.ua/laws/show/2297-17> (Дата звернення: 11.05.2023)

42. Статкус А. В., Мацалак В. І., Мацалак В. І. Огляд концепції інформаційної технології. Збірник тез: "IPST-2022" одинадцята міжнародна науково-технічна конференція НТУ "ХПІ". М. Харків, 2022. – с. 21-25.

43. Терещенко Л. О., Матієнко-Зубенко І. І. Інформаційні системи і технології в обліку: навч. посіб. Київ: КНЕУ, 2004. 187 с.

44. Управління особистими фінансами: збірник матеріалів IV Міжнародної науково-практичної конференції (8–9 вересня. 2021, м. Тернопіль). Тернопіль, 2021. 247 с.

## ДОДАТКИ

### Додаток А

## SUMMARY

Shramko E.V. Development of a Web-Based Accounting and Personal Finance Analysis System. Qualification work of a bachelor. Sumy State University, Sumy, 2023.

The paper describes the object of study, investigates the state of automation in this area, substantiates the requirements for it, develops business process models, designs the architecture, develops a website, writes a short user manual.

Keywords: personal finance, automation, web-oriented system, business process modeling.

## АНОТАЦІЯ

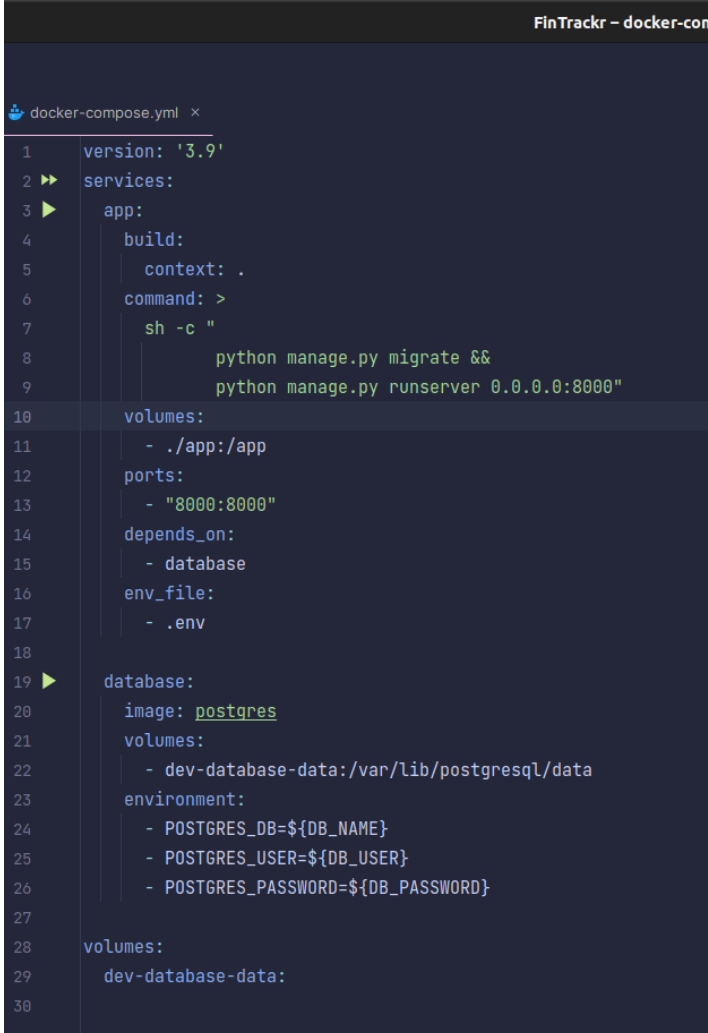
Шрамко Е. В. Розробка веборієнтованої інформаційної системи обліку та аналізу особистих витрат. Кваліфікаційна робота бакалавра. Сумський державний університет, Суми, 2023 р.

У роботі складено характеристику об'єкта дослідження, досліджено стан автоматизації у цій сфері, обґрунтовано вимоги до неї, розроблено моделі бізнес-процесів, спроектовано архітектуру, розроблено вебсайт, написано коротку інструкцію для користувача.

Ключові слова: особисті фінанси, автоматизація, веборієнтована система, моделювання бізнес-процесів.



## Додаток Б



```
FinTrackr – docker-con
docker-compose.yml x
1 version: '3.9'
2 services:
3   app:
4     build:
5       context: .
6     command: >
7       sh -c "
8         python manage.py migrate &&
9         python manage.py runserver 0.0.0.0:8000"
10    volumes:
11      - ./app:/app
12    ports:
13      - "8000:8000"
14    depends_on:
15      - database
16    env_file:
17      - .env
18
19   database:
20     image: postgres
21     volumes:
22       - dev-database-data:/var/lib/postgresql/data
23     environment:
24       - POSTGRES_DB=${DB_NAME}
25       - POSTGRES_USER=${DB_USER}
26       - POSTGRES_PASSWORD=${DB_PASSWORD}
27
28 volumes:
29   dev-database-data:
30
```

Рисунок Б.1 – Файл docker-compose.yml для створення контейнерів сайту та бази даних Postgres

## Додаток В

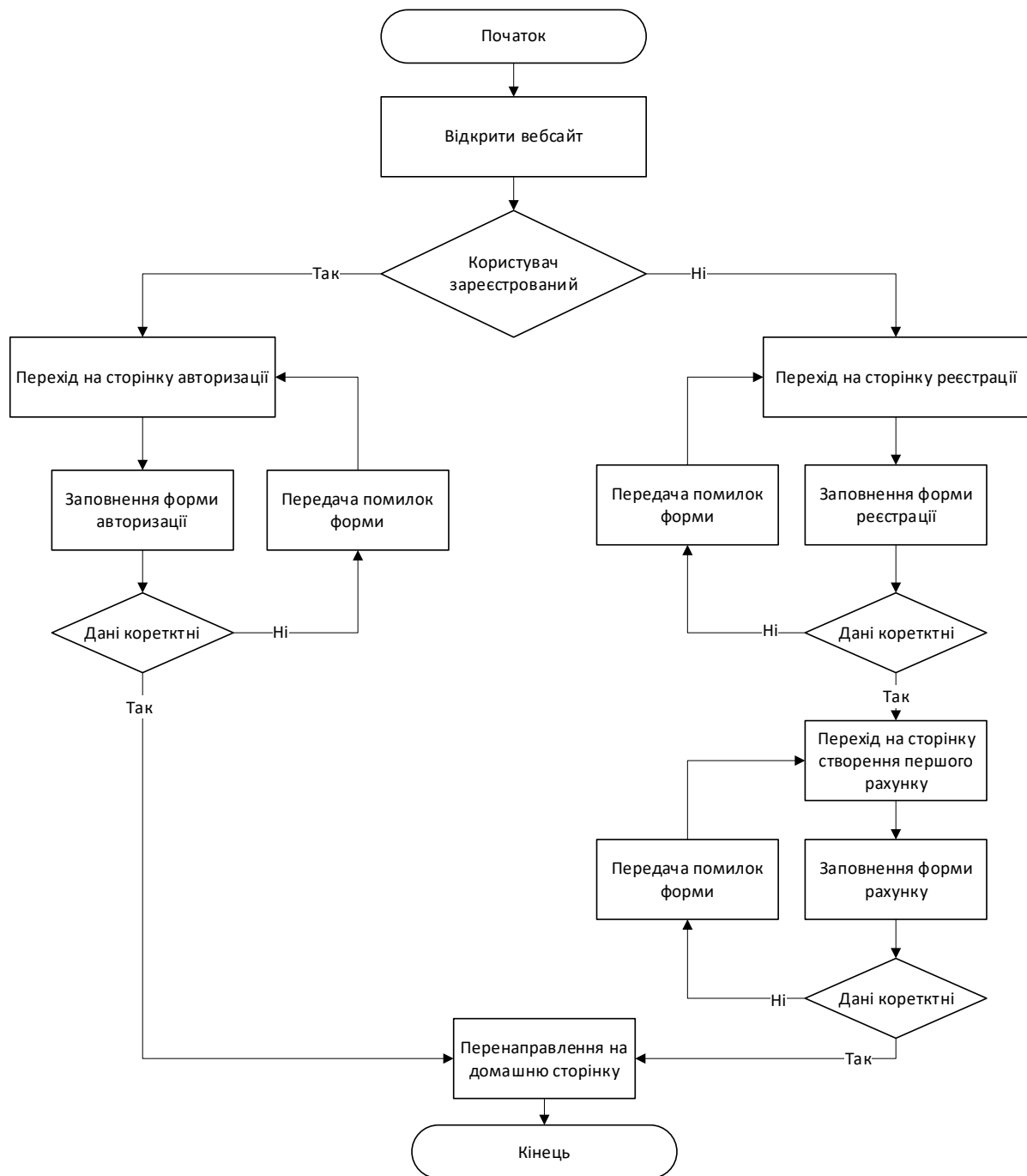


Рисунок В.1 – Блок-схема алгоритму входу на сайт

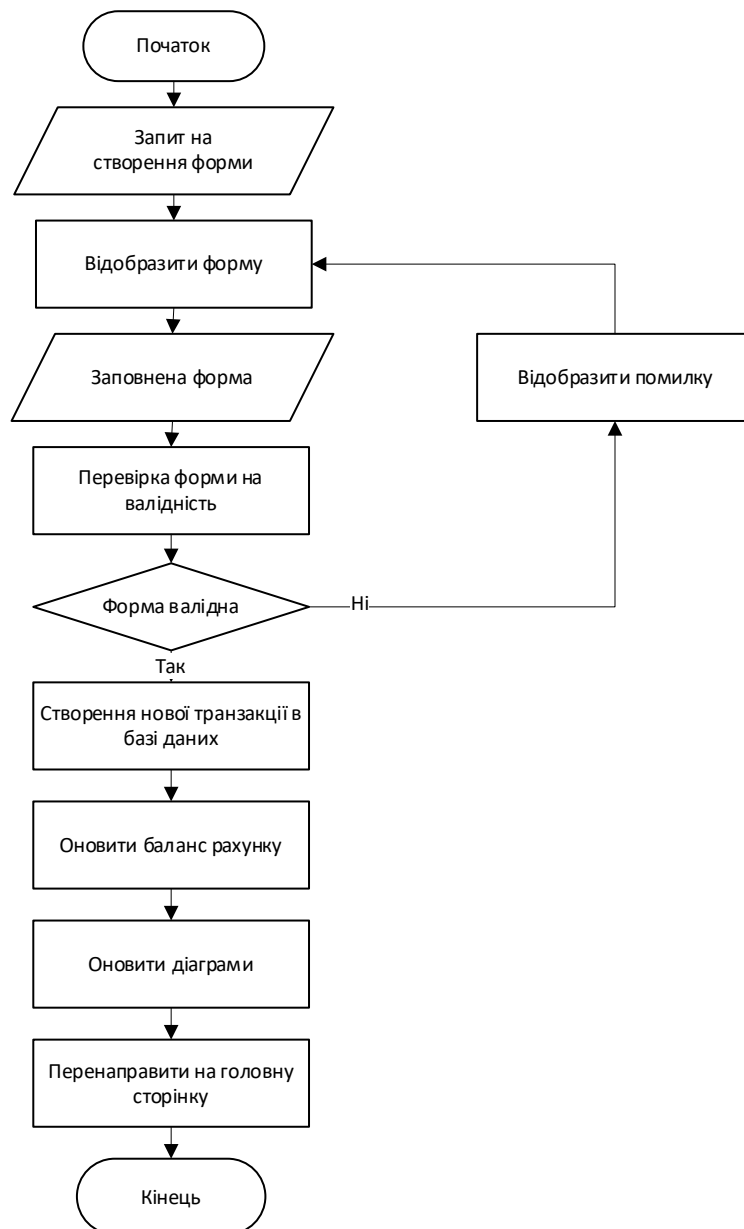


Рисунок В.2 – Блок-схема додання нової транзакції

## Додаток Г

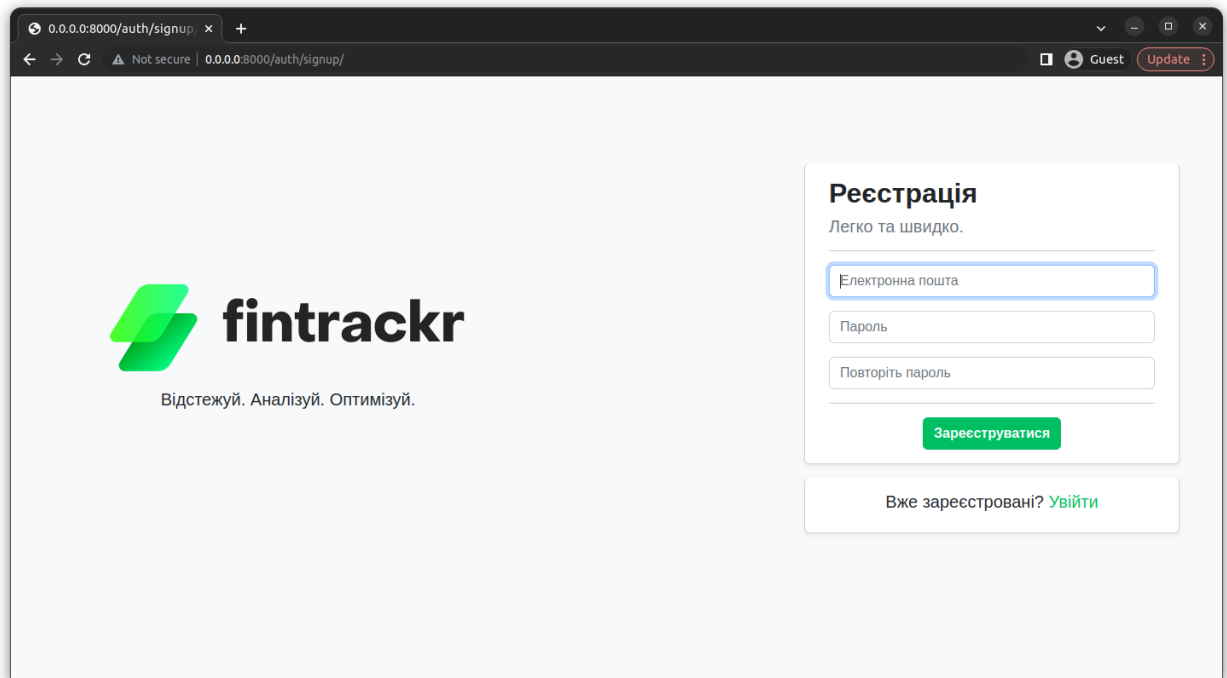


Рисунок Г.1 – Сторінка реєстрації користувача

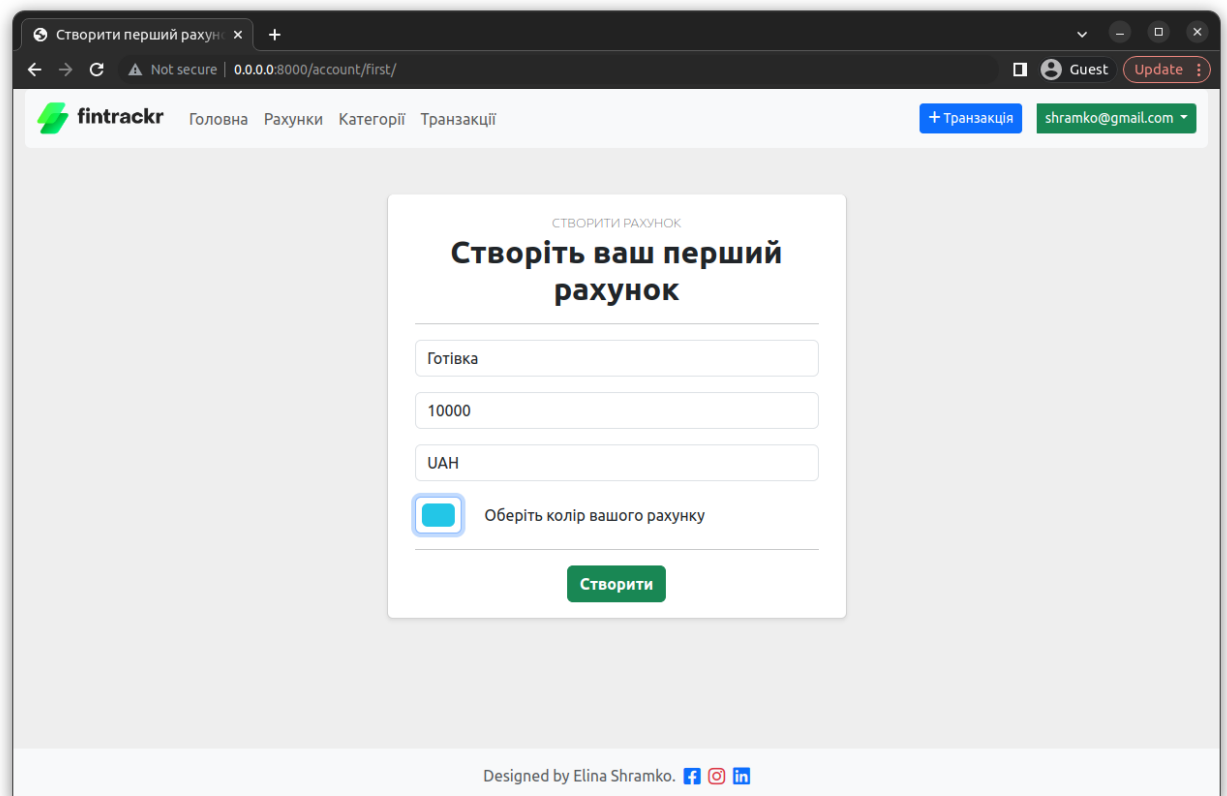


Рисунок Г.2 – Сторінка створення першого рахунку користувача

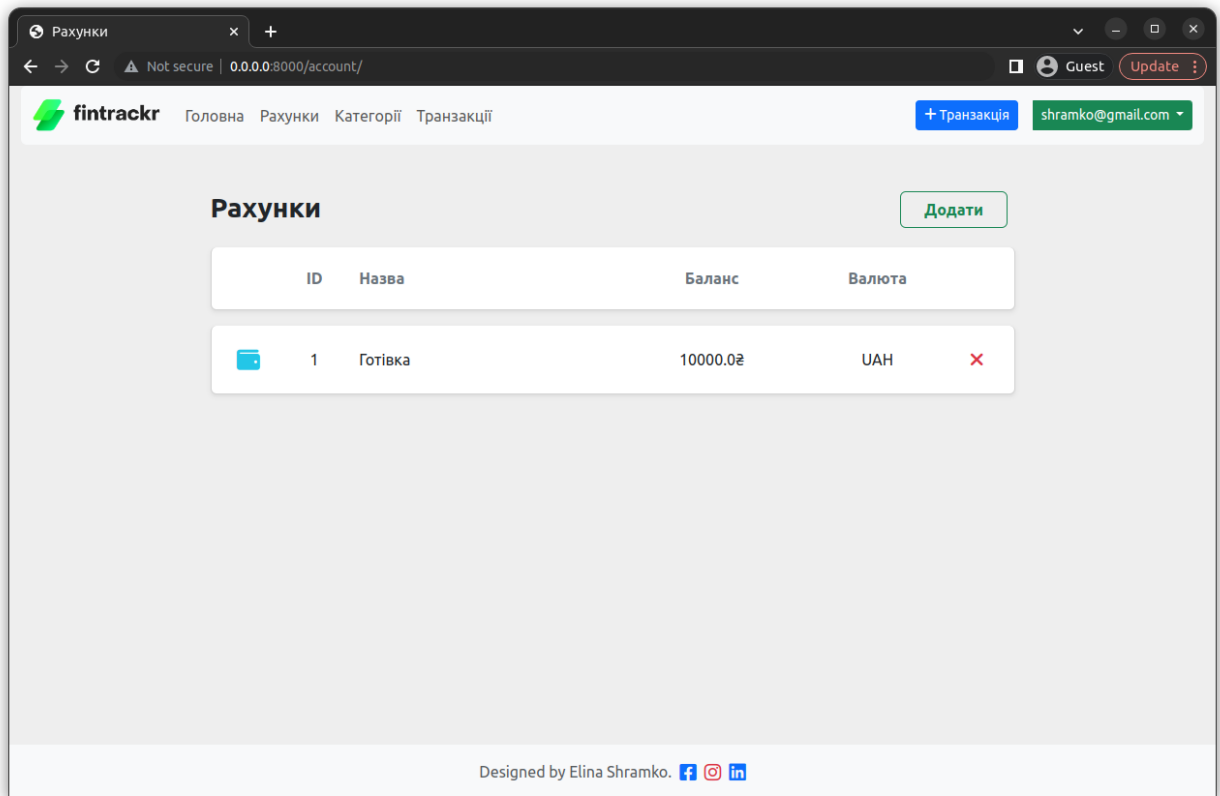


Рисунок Г.3 – Сторінка зі створеним рахунком

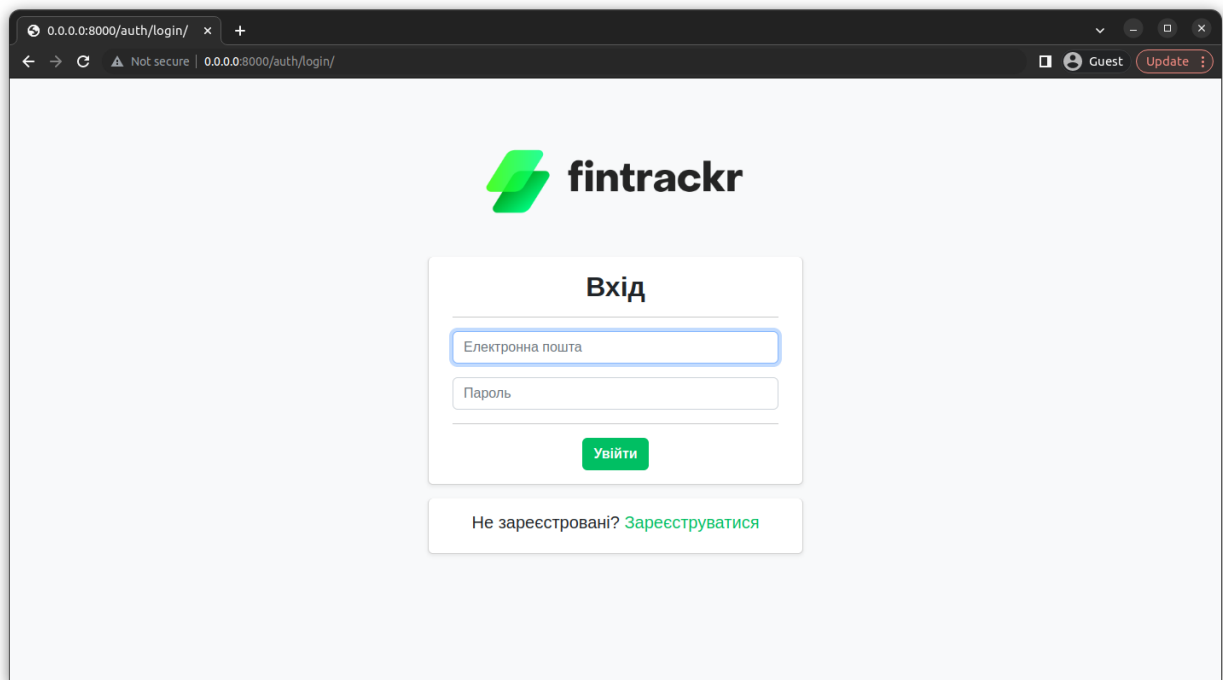


Рисунок Г.4 – Сторінка авторизації користувача

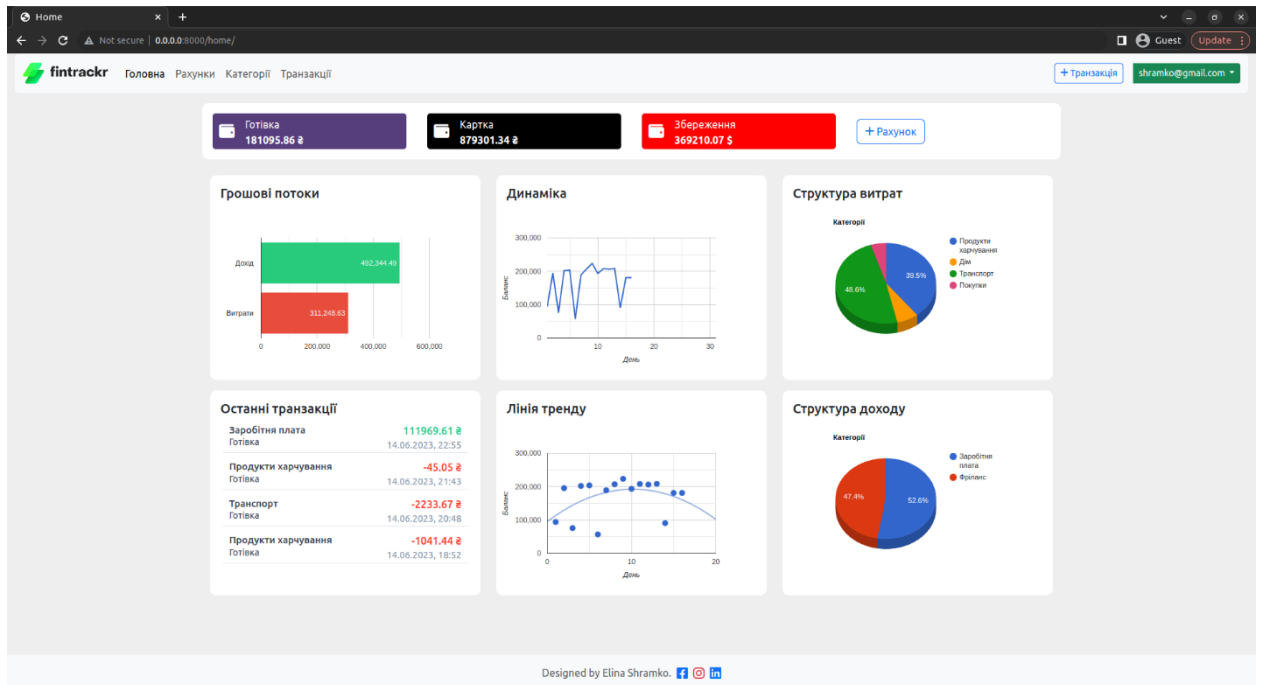


Рисунок Г.5 – Домашня сторінка. Перегляд аналітики для першого рахунку

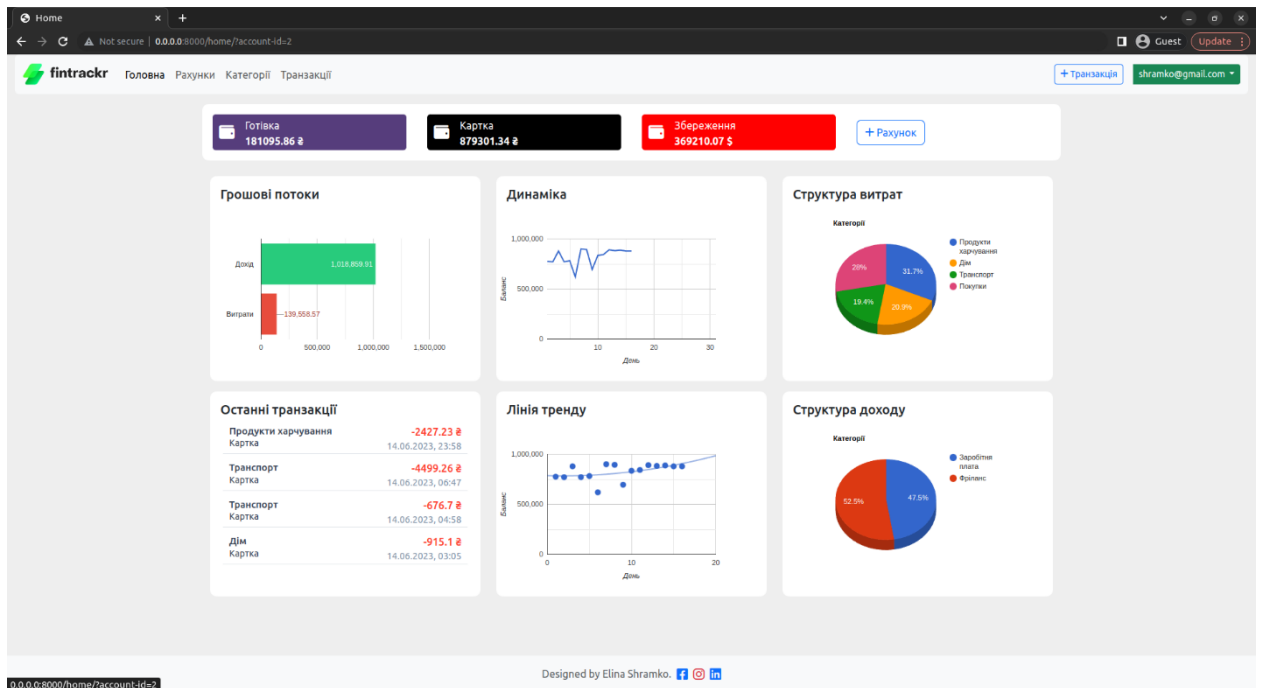


Рисунок Г.6 – Домашня сторінка. Перегляд аналітики для другого рахунку

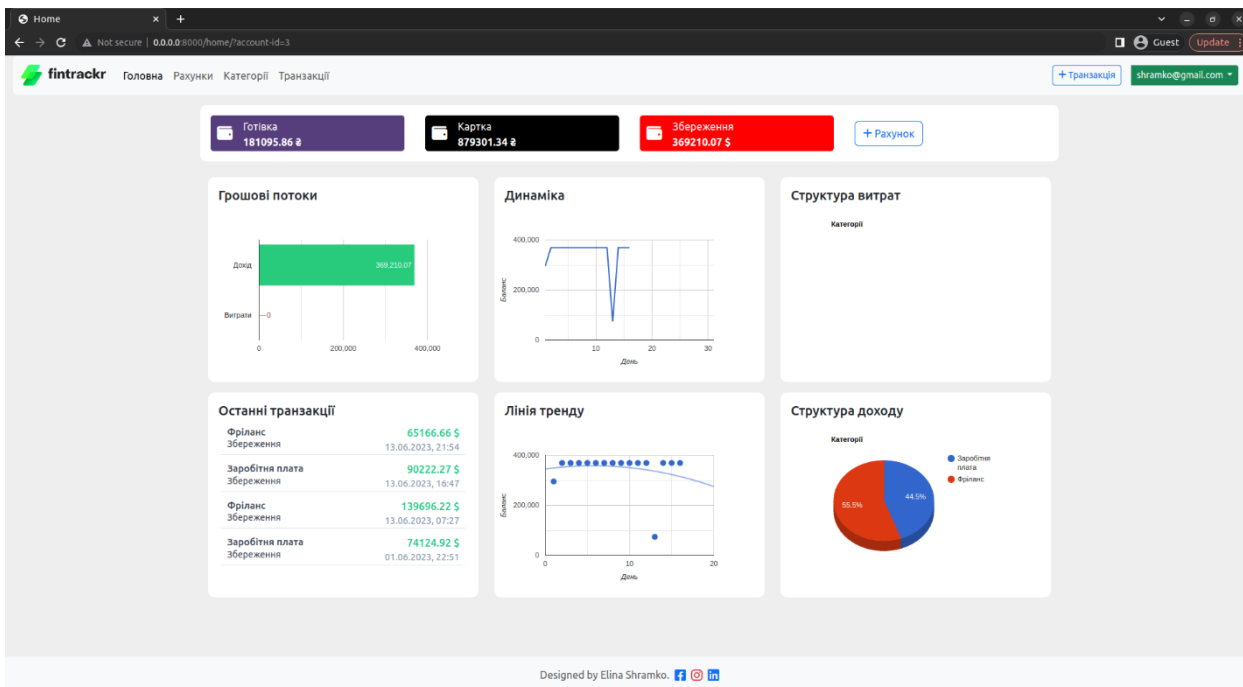


Рисунок Г.7 – Домашня сторінка. Перегляд аналітики для третього рахунку

The accounts table is as follows:

ID	Назва	Баланс	Валюта
1	Готівка	181095.85999999999₴	UAH
2	Картка	879301.33999999999₴	UAH
3	Збереження	369210.07000000007\$	USD

Рисунок Г.8 – Сторінка зі списком рахунків користувача

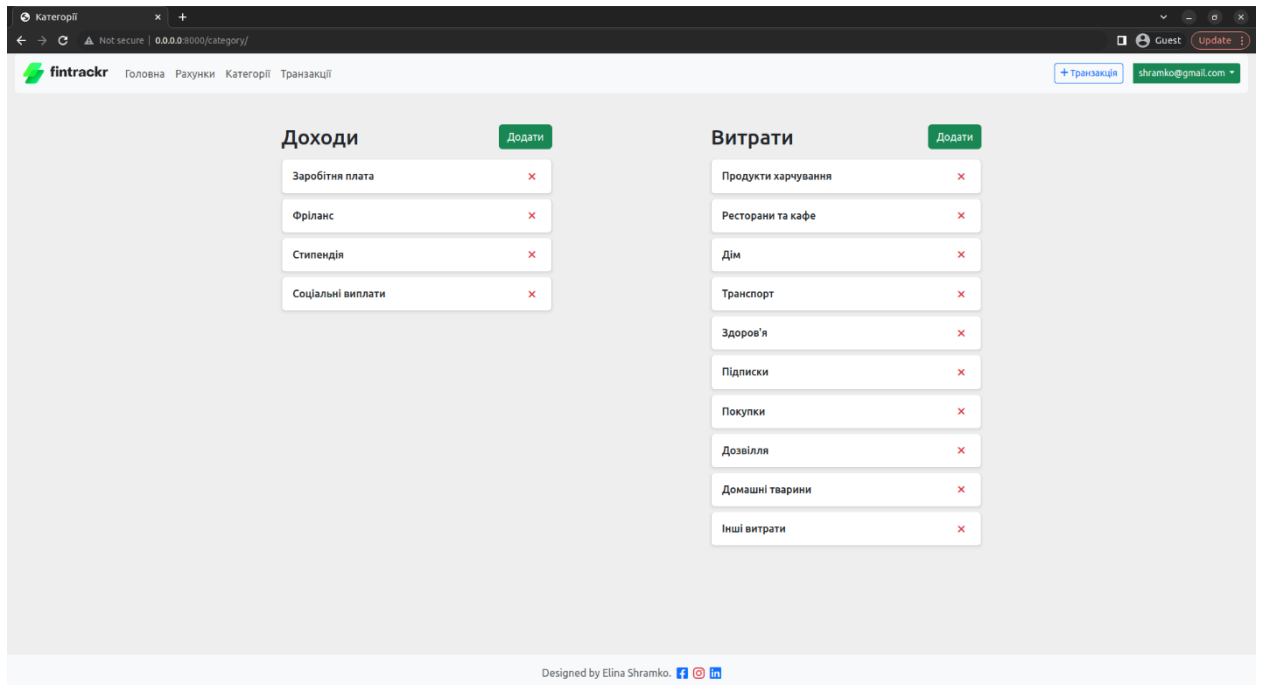


Рисунок Г.9 – Сторінка зі списком категорій витрат та доходів користувача

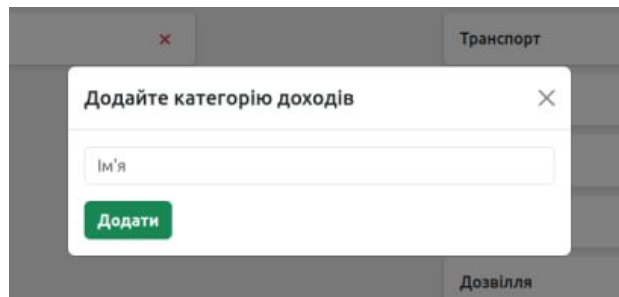


Рисунок Г.10 – Форма створення нової категорії доходів

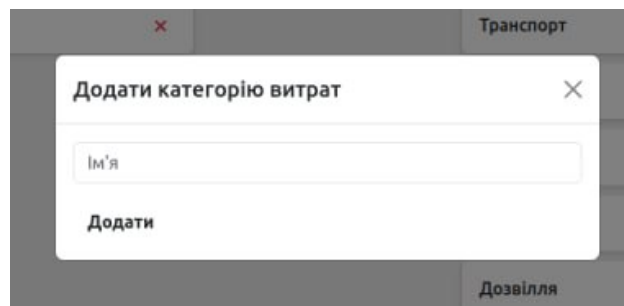


Рисунок Г.11 – Форма створення нової категорії доходів



Транспорт

Редагувати категорію

Заробітня плата

Застосувати

Дозвілля

Рисунок Г.12 – Форма редагування категорії

Транзакції

Додати

ID	Сума	Категорія	Рахунок	Дата
1	-2427.23	Продукти харчування	Картка	14.06.2023, 23:58
2	111969.61	Заробітня плата	Готівка	14.06.2023, 22:55
3	-45.05	Продукти харчування	Готівка	14.06.2023, 21:43
4	-2233.67	Транспорт	Готівка	14.06.2023, 20:48
5	-1041.44	Продукти харчування	Готівка	14.06.2023, 18:52
6	-4811.66	Покупки	Готівка	14.06.2023, 12:24
7	-3461.74	Транспорт	Готівка	14.06.2023, 11:07
8	-1492.23	Транспорт	Готівка	14.06.2023, 06:52
9	-4499.26	Транспорт	Картка	14.06.2023, 06:47

Рисунок Г.13 – Сторінка зі списком транзакцій

The screenshot shows a web browser window with the URL `0.0.0.0:3000/transaction/create/`. The page title is "Створити транзакцію". The header includes the "fintrackr" logo and navigation links: "Головна", "Рахунки", "Категорії", and "Транзакції". A user profile dropdown shows "shramko@gmail.com" and an "Update" option. The main content area features a form titled "Створіть транзакцію" with the following fields: "Сума" (text input), "Рахунок" (text input), "Витрата" (red button), "Дохід" (green button), "Категорія" (dropdown menu), and "Коментар" (text area). A green "Створити" button is at the bottom of the form. The footer text reads "Designed by Elina Shramko." with social media icons for Facebook, Instagram, and LinkedIn.

Рисунок Г.13 – Сторінка створення транзакції

The screenshot shows a web browser window with the URL `0.0.0.0:3000/account/create/`. The page title is "Створіть рахунок". The header is identical to the previous screenshot. The main content area features a form titled "Створіть рахунок" with the following fields: "Назва" (text input), "Баланс" (text input), "Валюта" (text input), and a color selection section with a purple swatch and the text "Оберіть колір вашого рахунку". A green "Створити" button is at the bottom of the form. The footer text reads "Designed by Elina Shramko." with social media icons for Facebook, Instagram, and LinkedIn.

Рисунок Г.14 – Сторінка створення рахунка