

**КИЇВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ
ІМЕНІ ТАРАСА ШЕВЧЕНКА**

Факультет інформаційних технологій

Кафедра технологій управління

Спеціальність 122 «Комп'ютерні науки»
Освітньо-наукова програма «Управління проєктами»

КВАЛІФІКАЦІЙНА РОБОТА МАГІСТРА

на тему:

«Дослідження процесів управління проєктом розробки веб-додатку CRM системи для СРА мережі»

Студента 2-го курсу групи УП-21

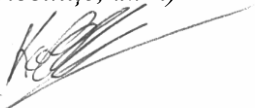
Науковий керівник:

К.Т.Н., доцент

(науковий ступінь, вчене звання)

КОВАЛЕНКА Олександра

(прізвище, ім'я)



(підпис студента)

ТІМІНСЬКИЙ Олександр

(прізвище, ім'я)

(дата)

(підпис)

Попередній захист:

(Висновок: «До захисту в Екзаменаційній комісії»)

Завідувач кафедри

технологій управління, проф. _____
(підпис)

МОРОЗОВ Віктор _____
(прізвище, ім'я)

(дата)

Київ-2024

**КИЇВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ
ІМЕНІ ТАРАСА ШЕВЧЕНКА**

Факультет інформаційних технологій

Кафедра технологій управління

Освітній рівень Магістр

Спеціальність 122 Комп'ютерні науки

Освітня програма Управління проектами

ЗАТВЕРДЖУЮ

Завідувач кафедри

професор **МОРОЗОВ Віктор**

“ _____ ” _____ 20__ року

**З А В Д А Н Н Я
НА ВИКОНАННЯ КВАЛІФІКАЦІЙНОЇ РОБОТИ**

Студент: Коваленко Олександр Володимирович

Група: УП-22

1. Тема кваліфікаційної роботи

«Дослідження процесів управління проектом розробки веб-додатку CRM системи для СРА мережі»

Затверджена протоколом № 6 від “6” листопада 2023 року.

2. Строк подання студенткою готової роботи - “6” травня 2024 р.

3. Цільова установка та вихідні дані до роботи: вибір та аналіз методів, методологій і засобів управління проектами для виконання проекту в поставлений строк, використовуючи заплановані ресурси та бюджет.

4. Зміст роботи:

Обґрунтування доцільності та життєздатності проекту, аналіз предметної галузі, розробка проекту, мета, цілі та продукт проекту, PEST-аналіз, SWOT-аналіз, аналіз методологій і методик управління проектами, розробка програмного забезпечення проекту, математичне та інформаційне забезпечення, функціональний аналіз, архітектура інформаційної системи, організаційна структура проекту, опис модулів програмного забезпечення, декомпозиція процесів, функціональність, беклог проекту, гнучкі технології створення продукту та планування IT-проекту, визначення вимог до проекту, управління бюджетом проекту, розробка дерева проблем та рішень проекту, використання фреймворку Kanban для управління проектом, модулі продукту проекта, WBS процесів, управління ризиками, контроль за прогресом, управління якістю проекту.

5. Перелік графічного матеріалу (слайдів):

Актуальність обраної предметної області, цілі проекту, Вибір фреймворку, модель організаційної структури команди, бюджет проекту, модель WBS проекту за фазами життєвого циклу, діаграма Ганта, архітектури продукту, Функціональний аналіз, концептуальна модель бази даних, математичне та інформаційне забезпечення, ризики проекту, SWOT аналіз, економічна складова проекту, Структура програмного забезпечення, робота зі стейкхолдерами, висновки.

6. Календарний план виконання роботи:

№ з/п	Назва частин роботи	Виконання роботи
1	Аналітичний огляд літератури за темою досліджень з предмету дослідження	01.12.23-24.12.23
2	Аналіз існуючих методологій управління	02.01.24-10.12.24
3	Постановка задачі на розробку веб застосунку для СРА мережі	15.01.24-20.01.24
4	Погодження плану написання кваліфікаційної роботи магістра з науковим керівником	14.02.24
5	Підготовка розділу 1 «ДОСЛІДЖЕННЯ ТА ОБҐРУНТУВАННЯ ДОЦІЛЬНОСТІ ТА ЖИТТЄЗДАТНОСТІ ПРОЄКТУ»	14.02.24-21.03.24
6	Підготовка розділу 2 «АНАЛІЗ ТА УПРАВЛІННЯ ПРОЄКТОМ СТВОРЕННЯ ДОДАТКА ДЛЯ ОРГАНІЗАЦІЇ РОЗВАЖАЛЬНИХ ПОДІЙ»	21.03.24-06.04.24
7	Підготовка розділу 3 «РОЗРОБКА ТА КОНТРОЛЬ ЯКОСТІ ПРОЄКТУ ДЛЯ ОРГАНІЗАЦІЇ РОЗВАЖАЛЬНИХ ПОДІЙ»	06.04.24-12.04.24
8	Підготовка розділу 4 «РЕАЛІЗАЦІЯ ТА ВЗАЄМОДІЯ В ПРОЄКТІ РОЗРОБКИ ДОДАТКА ДЛЯ ОРГАНІЗАЦІЇ РОЗВАЖАЛЬНИХ ПОДІЙ»	12.04.24-01.05.24
9	Оформлення кваліфікаційної роботи	01.05.24-05.05.24
10	Передача кваліфікаційної роботи науковому керівникові	06.05.24
11	Передача кваліфікаційної роботи рецензенту для рецензування	10.05.24
12	Захист кваліфікаційної роботи	23.05.24-27.05.24

Дата видачі завдання “06” листопада 2023 р.

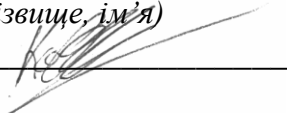
Керівник роботи к.т.н., доцент, ТІМІНСЬКИЙ Олександр
(посада, прізвище, ім'я)

(підпис)

Завдання прийняв до виконання студент групи УП-21

КОВАЛЕНКО Олександр

(прізвище, ім'я)



ЗМІСТ

АНОТАЦІЯ	5
ВСТУП	7
РОЗДІЛ 1. АНАЛІЗ ПРОЦЕСУ РОБОТИ СРА МЕРЕЖІ	10
1.1 Аналітичний огляд літератури за темою досліджень СРА мереж	10
1.2 Аналіз існуючих СРА мереж	15
1.3 Аналіз існуючих методологій управління	19
1.3.1 Scrum	22
1.3.2 Kanban	22
1.3.3 Scrumban	23
1.3.4 Design Thinking	23
1.3.5 Agile Project Management (AgilePM)	23
1.3.6 Scaled Agile Framework® (SAFe®)	24
1.3.7 Large-Scale Scrum (LeSS)	24
1.3.8 Nexus	24
1.3.9 AgileSHIFT	25
1.3.10 Open Space Agility (OSA)	25
1.3.11 Agility Scales	25
1.3.12 Вибір фреймворку	25
1.4 Аналіз Методик опису вимог	26
1.5 Аналіз основних процесів предметного середовища	27
1.6 Постановка задачі на розробку веб застосунку для СРА мережі	33
1.6 Висновки до першого розділу	34
РОЗДІЛ 2. РОЗРОБКА АРХІТЕКТУРИ WEB ДОДАТКУ CRM СИСТЕМИ ДЛЯ СРА МЕРЕЖІ	36
2.1 Розробка архітектури продукту	36
2.1.1 Функціональний аналіз	36
2.1.2 IDEF0 процесу відкриття офери	41
2.1.3 Архітектура інформаційної системи	42
2.1.4 Дерево причин та наслідків (діаграма Ішикави)	44
2.1.5 Дерево проблем та рішень	45
2.1.6 SWOT аналіз	46
2.2 Математичне забезпечення веб додатку	47
2.3 Інформаційне забезпечення веб додатку	50

2.4 Обґрунтування вибору програмних засобів	54
2.5 Структура програмного забезпечення	56
2.6 Висновки до другого розділу	59
РОЗДІЛ 3. УПРАВЛІННЯ ПРОЄКТОМ СТВОРЕННЯ ВЕБ ДОДАТКУ. ОРГАНІЗАЦІЙНА СТРУКТУРА ПРОЄКТУ, КОНТРОЛЬ ЯКОСТІ.	61
3.1 Інтелектуальна карта проекту	61
3.3 Організаційна структура проекту	62
3.4 Планування проектних робіт: визначення складу, тривалості робіт і зв'язків між роботами. Ресурсне планування проекту	62
3.7.1 Функціонування системи, вплив зовнішніх зв'язків	70
3.7.2 Формування команди	74
3.8 Керівництво користувача	77
3.9 Огляд процесу тестування	87
3.10 Економічна складова проекту	88
3.11 Висновки до третього розділу	91
РОЗДІЛ 4. РОБОТА ЗІ СТЕЙКХОЛДЕРАМИ. УПРАВЛІННЯ РИЗИКАМИ	93
4.1 Аналітичний огляд зацікавлених сторін та вплив на проєкт	93
4.2 Управління ризиками	96
4.2.1 Підсумки здійснення PEST-аналізу для компанії	100
ВИСНОВОК	104
СПИСОК ВИКОРИСТАНИХ ІНФОРМАЦІЙНИХ ДЖЕРЕЛ	106
ДОДАТОК А	111
ДОДАТОК Б	121

АНОТАЦІЯ

кваліфікаційної магістерської роботи на тему:

«Дослідження процесів управління проектом розробки веб-додатку CRM системи для CPA мережі»

Студент: Коваленко Олександр Володимирович.

Науковий керівник: Тімінський Олександр Георгійович.

Рік захисту - 2024.

Темою даної роботи є «Дослідження процесів управління проектом розробки веб-додатку CRM системи для CPA мережі». Предметна область розглядає процеси управління проектом в контексті створення веб додатку для CPA мережі.

Метою даної роботи є розробка системи управління проектом та його ресурсам, створення концепції додатка, вибір управлінських рішень та програмного забезпечення для управління та розробки продукту, розробка архітектури платформи, аналіз ризиків та розробку звітів про виконання проекту.

Об'єктом дослідження є система управління проектом розробки веб-додатка для CPA мережі.

Предметом дослідження є процеси управління і розробки проекту, такі як організація команди, прорахунок ресурсів, системне проектування, вибір методологій, імплементація математичних моделей, управління якістю та ризиками, аналіз всіх сутностей проекту.

Наукова новизна полягає в розробці інтегрованого підходу до управління даними в CPA мережах: Впровадження новітніх методик обробки та аналізу даних для оптимізації взаємодії з клієнтами в CPA мережах, що дозволяє збільшити ефективність рекламних кампаній і знизити витрати на маркетинг.

Застосування Agile методологій у процесі розробки CRM-системи: Адаптація та модифікація Kanban фреймворку, для специфіки роботи в CPA мережах, що включає гнучкість у плануванні, ітеративний розвиток та неперервний зворотній зв'язок від користувачів.

Створення адаптивного інтерфейсу для максимальної зручності користувачів: Оптимізація користувацького інтерфейсу CRM-системи з метою поліпшення користувацького досвіду, зручності використання системи на різних пристроях і платформах.

Дипломна робота складається зі вступу, чотирьох розділів з висновками, загальний висновок, список використаних джерел та двох додатків з лістингом коду.

Робота включає 136 сторінок з додатками, 54 рисунків та 21 таблиць.

ВСТУП

У світі швидкими темпами розвивається новий вид маркетингу, заснований на оплаті за результат, базований на win-win співпраці між рекламодавцем та людиною, що просуває його продукт, задля того щоб обидві сторони знаходили одне одного існують так званні CPA мережі (з англ. Cost Per Action - “ціна за дію”, модель партнерської реклами), вони поєднують інтереси рекламодавців та веб-майстрів (майданчики/вітрини/сайти, де рекламуються послуги/продукти для кінцевих клієнтів). Рекламодавець через посередників (веб-майстрів) отримує клієнтів, веб-майстер заробляє комісію за залучених клієнтів.

Через новизну цієї ніші маркетингу, кожен проєкт розвивається індивідуально, а отже і технологічні рішення у всіх різні, від незручного і не функціонального додатку до неоптимізованих витрат проєктних ресурсів на розробку рішення. Тож партнерам таких мереж доводиться обирати між поганим і гіршим. Звідси випливає висновок про відсутність сучасного оптимального і зручного додатку для CRM системи CPA мережі, яке б не мало понаднормових витрат під час розробки чи експлуатації, а отже в цій *предметній області є актуальним* запропонувати інноваційне рішення в управлінні проєктом створення такого продукту, що і буде зроблено в даній кваліфікаційній роботі магістра.

CPA мережа контролює якість та кількість трафіку для рекламодавців, дає асортимент продуктів/послуг, статистику веб-майстрам, а також веде взаєморозрахунки. Веб-майстер у CPA мережі бачить усі пропозиції від усіх рекламодавців зі статистикою, вибирає, з ким працювати, а гроші отримує агреговано від CPA мережі. Аналогічно рекламодавець платить одній CPA-мережі за продаж всіх веб-майстрів і не супроводжує технічні та фінансові процеси.

Темою кваліфікаційної роботи є «Дослідження процесів управління проєктом розробки веб-додатку CRM системи для CPA мережі». Предметна область розглядає процеси управління проєктом в контексті створення веб додатку для CPA мережі.

Метою даної роботи є розробка системи управління проєктом та його ресурсам, створення концепції додатка, вибір управлінських рішень та програмного

забезпечення для управління та розробки продукту, розробка архітектури платформи, аналіз ризиків та розробку звітів про виконання проєкту.

Об'єктом дослідження є система управління проєктом розробки веб- додатка для CPA мережі.

Предметом дослідження є процеси управління і розробки проєкту, такі як організація команди, прорахунок ресурсів, системне проектування, вибір методологій, імплементація математичних моделей, управління якістю та ризиками, аналіз всіх сутностей проєкту.

Наукова новизна полягає в розробці інтегрованого підходу до управління даними в CPA мережах: Впровадження новітніх методик обробки та аналізу даних для оптимізації взаємодії з клієнтами в CPA мережах, що дозволяє збільшити ефективність рекламних кампаній і знизити витрати на маркетинг. Оптимізації роботи партнерських мереж (CPA мереж) на прикладі управління проєктом розробки веб додатку CRM системи для CPA мереж, що дозволить створити продукт з мінімальною кількістю витрат та в заданий термін.

Зручність таких мереж у тому, що web майстрам не доводиться самотійно шукати рекламодавців, укладати договори з усіма компаніями та стежити за статистикою. Працюючи через CPA мережі, веб-майстер скорочує свої операційні витрати та розуміє за статистикою, які пропозиції вибрати.

Завдання продукту проєкту – забезпечити коректний облік статистики, безперебійну систему комунікації, виплат та виконання зобов'язань всіх сторін.

CPA мережа — це унікальний B2B продукт, який забирає на себе найбільш трудомісткі процеси, що зазвичай вичерпують людський час та енергію, такі як підбір продукту для свого майданчика, виплати, врегулювання робочих та іноді конфліктних ситуацій як на стороні рекламодавця, так і на стороні веб-майстра . Сучасні CPA мережі здатні позбавити всі сторони процесів від непотрібних проблем, що призводить не лише до зростання ринку, а й покращує його якість. [1]

Мета продукту проєкту полягає у підвищенні ефективності співпраці рекламодавців та партнерської мережі веб-майстрів за рахунок створення сучасного web застосунку для CPA мережі, який має містити необхідні у роботі web майстрів

статистичні дані, зручний та зрозумілий інтерфейс, функціонал для відкриття, налаштування та запуску оферів в одному застосунку.

В даний час, як за технологіями, так і за змістом, це потужний інструмент, за допомогою якого можна реалізувати програму лояльності для клієнтів, залучити трафік з нових джерел, підвищити середній чек, знизити кількість невікуплених замовлень та багато іншого. Його можна віднести до обов'язкових рішень безлічі завдань маркетингу для бізнесу в інтернеті.

Для досягнення даної мети потрібно виконати наступні завдання:

- провести аналіз існуючих рішень для CPA мереж
- провести аналіз основних потреб користувачів
- провести аналіз алгоритмів та пайплайнів розробки систем подібного типу
- провести User Experience / User Interaction (далі UX/UI) дослідження
- сформулювати функціональні та нефункціональні вимоги для розроблюваного застосунку
- розглянути процеси даної предметної області
- розробити архітектуру додатку
- розробити та протестувати програмний продукт

Актуальність даної роботи і важливість продукту проекту прямо впливає з новизни CPA маркетингу в цілому і відсутністю оптимальних рішень на ринку, зокрема проблемою є неефективна взаємодія між веб майстром, рекламодавцем і системою, а також неефективне використання ресурсів проекту через необхідність наймати менеджерів для роботи з системою в усіх сучасних рішеннях.

РОЗДІЛ 1. АНАЛІЗ ПРОЦЕСУ РОБОТИ CPA МЕРЕЖІ

1.1 Аналітичний огляд літератури за темою досліджень CPA мереж

CPA (Cost Per Action) - модель партнерської реклами, для якої рекламодавець це модель партнерської реклами, при якій рекламодавець залучає так званих помічників для пошуку клієнтів та оплачує лише вчинені користувачами дії, такі як покупка товару, надсилання контактних даних (для подальшої роботи контакт центрів, так звана оплата за “лід”, або встановлення додатку).

З сучасними темпами розвитку цифрових технологій, ведення бізнесу вже не є можливим без застосування інструментів інтернет маркетингу. Одним із таких інструментів є CPA маркетинг, який з'явився не так давно, але вже достатньо розповсюджений і приносить бізнесу прибутки (рис 1.1, рис.1.2).

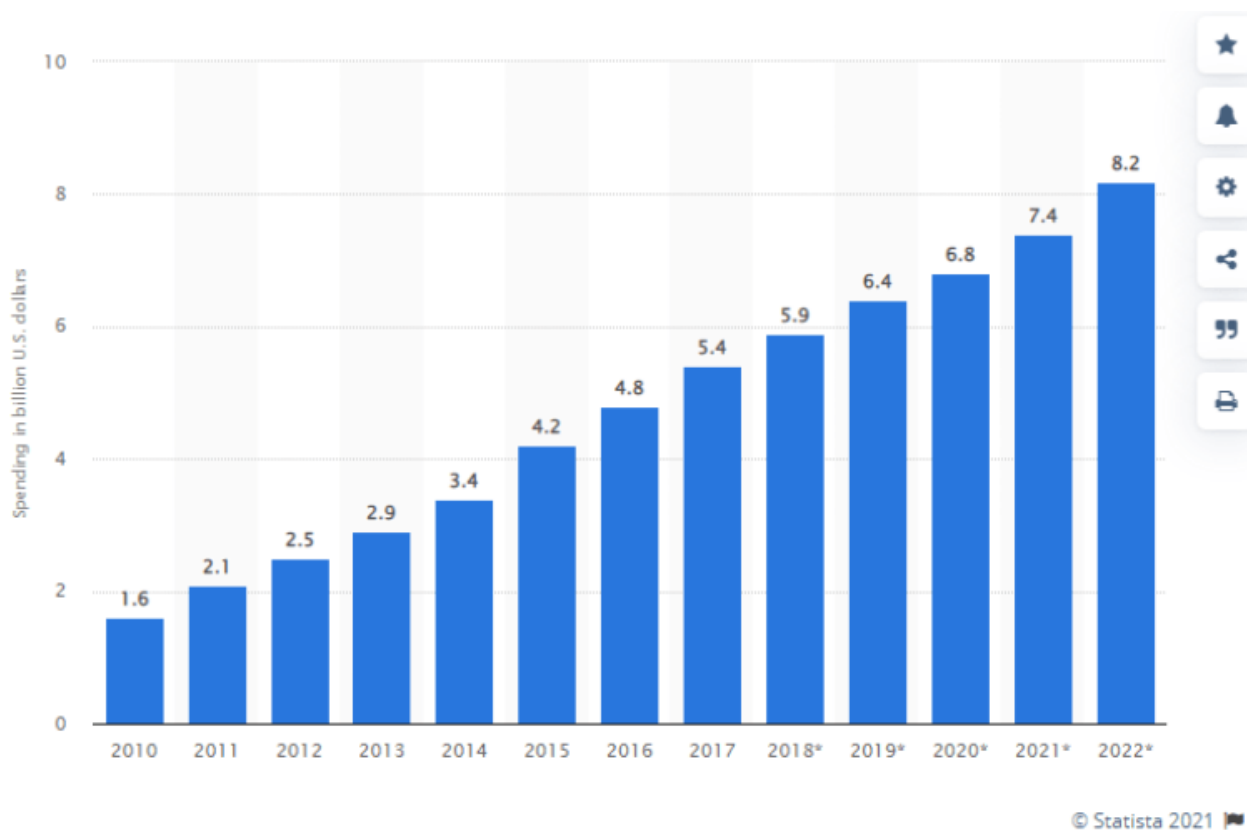


Рис. 1.1. Динаміка зростання об'єму витрат на маркетинг по моделі CPA у світі

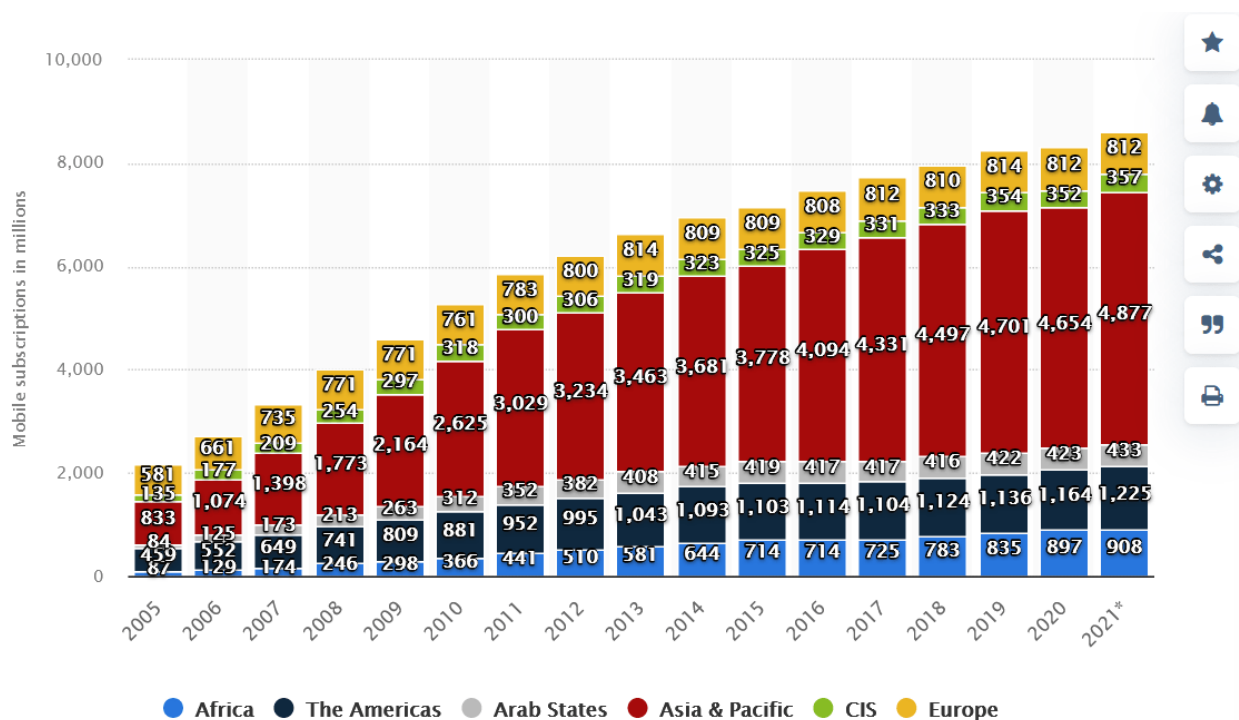


Рис. 1.2. Розподіл витрат по частинам світу

У даній системі партнерами виступають Web-майстри зі своїми власними сайтами, які перенаправляють трафік на цільові сторінки.

Головним плюсом CPA маркетинга є оплата саме за результат, що є найбільш прозорою формою (в порівнянні з оплатою за покази в Бізнес менеджері Facebook).

Так як дана тематика достатньо специфічна, нижче наведені найчастіше вживані у даній сфері та в моїй роботі ключові терміни:

- Офер (англ. Offer) — рекламна пропозиція від компанії, тобто короткий тезисний опис того що потрібно зробити для компанії, мета пропозиції, сума виплати за результат. Web майстер отримавши офер може підключати його до рекламних компаній, які він налаштовує на цільову дію для замовника. Якісна CPA мережа має достатньо високооплачуваних оферів.

- Web майстер — спеціаліст, який вміє налаштовувати, запускати рекламні кампанії та досягати цільової дії.

- Лити (трафік на офер) — комплекс дій web майстра для досягнення мети оферу.

- Конверсія (CR) — відсоток користувачів сайту, що зробили цільову дію, характеризує ефективність лиття на офер.

- Трафік — потік користувачів, що виникає як наслідок роботи рекламної кампанії, відслідковується через сервіси Google Analytics та Facebook Ads.

- Tracking link — унікальне посилання офера, яке потрібно для відслідковування статистики рекламних кампаній.

- Пуш — повідомлення, яке отримує користувач при встановленні додатку, частина вирви продажу.

- Вирва продажу — комплекс взаємодії з клієнтом/користувачем для того, щоб він виконав цільову дію, як правило купив або вніс депозит.

- Депозит — інвестиція, внесення коштів в цільовий додаток або сайт, зроблений користувачем, з нього зазвичай web майстер отримує прибуток.

- Гео — країна(-и), на аудиторію яких розрахований офер, дана характеристика важлива тим, що потенційні клієнти різних країн мають різні поведінкові моделі, а отже і рекламні кампанії відрізняються.

- Хіти — переходи на цільовий сайт/додаток через рекламне оголошення.
- Ліди — користувачі, що пройшли реєстрацію на цільовому сайті/додатку.
- Хости — кількість унікальних переходів на домен.
- Visit to reg — математична модель підрахунку ефективності хітів, отже це $(\text{хіти}/100) \cdot \text{ліди}$

- Affiliate менеджер — менеджер, що управляє оферами, розподіляє між web майстрами, аналізує статистику, також виступає як спеціаліст технічної підтримки.

CPA у якості основної моделі оплати можна задіяти:

→ Facebook: виділення цільової аудиторії для початку

→ В пуш-мережах: модель “Середня ціна конверсії”

→ Google Ads: Цільова ціна за конверсію

→ У CPA мережах:

→ SEO просування сайтів

→ Існуючі моделі розрахунку CPA

Основні моделі розрахунку які існують на даний момент:

→ Ліди CPL (Cost Per Lead) – модель оплати тільки за приведений лід. Йдеться вже про клієнта з підтвердженими контактними даними.

→ Кліки (CPC) — рекламодавець платить за кожен клік по оголошенню. Вартість кліка розраховується за аукціонною системою.

→ Покази (CPM) — рекламодавець сплачує 1000 показів оголошення.

→ Продаж CPS (Cost Per Sale) — використовується для позначення вартості одного оплаченого замовлення. Коли ми говоримо про CPS, то завжди маємо на увазі вже здійснений продаж.

→ Налаштування програм (CPI) — рекламодавець платить за здійснену установку програми.

→ інші, менш поширені цільові дії.[1]

Отже працювати по моделі CPA можна із залученням різних типів трафіку (Рис. 1.3) з різними типами цілових дій.



Рис. 1.3. Канали залучення трафіка, характерні для CPA

Таким чином загальна формула: $CPA = \text{сума затрат на рекламу} / \text{кількість цільових дій}$. Під сумою ми розуміємо весь бюджет, витрачений на рекламну кампанію, а під кількістю цільових дій - сумарну кількість виконаних цільових дій.

Наприклад, припустимо що онлайн-магазин прикрас запускає рекламну кампанію на покупки з бюджетом 100 000 гривень. Рекламний трафік приніс 800 продажів прикрас різних видів. Отже $100\,000 / 800 = 125$ гривень, тобто ціна однієї цільової дії (конверсії). Для таких розрахунків в рекламних кабінетах зазвичай вбудовують мікросервіс, який дозволяє не тільки рахувати конкретну компанію, а і брати загалом декілька.

Цільові дії можуть бути різними на всіх етапах конверсії, загалом можна поділити на такі:

- Здійснення продажу (в CPS моделі)
- Заповнення і відправка форми з контактними даними
- Перехід на визначену Web-сторінку
- Конвертація користувача в ліда (CPL модель)

- Та інші менш розповсюджені в маркетингу дії, навіть перегляд відео на протязі певного часу можна вважати такою дією, все залежить від цілей бізнеса та рекламної кампанії.

Отже можна виділити найрозповсюджені дії для CPA моделі:

- Заовлення товару (покупка)
- Додавання товару в корзину на сайті
- Заовлення Call back (дзвінок менеджера або консультанта)
- Підписка на розсилку
- Перегляд відео
- Встановлення додатків, програм, або скачування файлів
- Заовнення анкетних даних [2]

Тепер, коли ми розуміємо основний принцип роботи CPA моделі маркетингу, можна перейти до аналізу процесів роботи CPA мереж.

Крупні бренди і невеликі бізнеси просувають свої продукти через різні канали, але роблять це дедалі частіше не напряду, а через CPA мережі, тобто таку платформу, яка поєднує коло Web-майстрів з колом брендів, що надають так звані офери (запит на цільову дію, бюджет і готовність оплатити цільовий результат), так як вирішувати дані питання вручну займало би більше часу, та скоріш за все було б менш ефективно для бренду.

Отже CPA мережа це посередник між рекламодавцем і веб майстром, який буде розміщувати рекламу на своїх платформах. Детальніше взаємозв'язки можна розглянути на наведеній схемі (Рис. 1.4.):

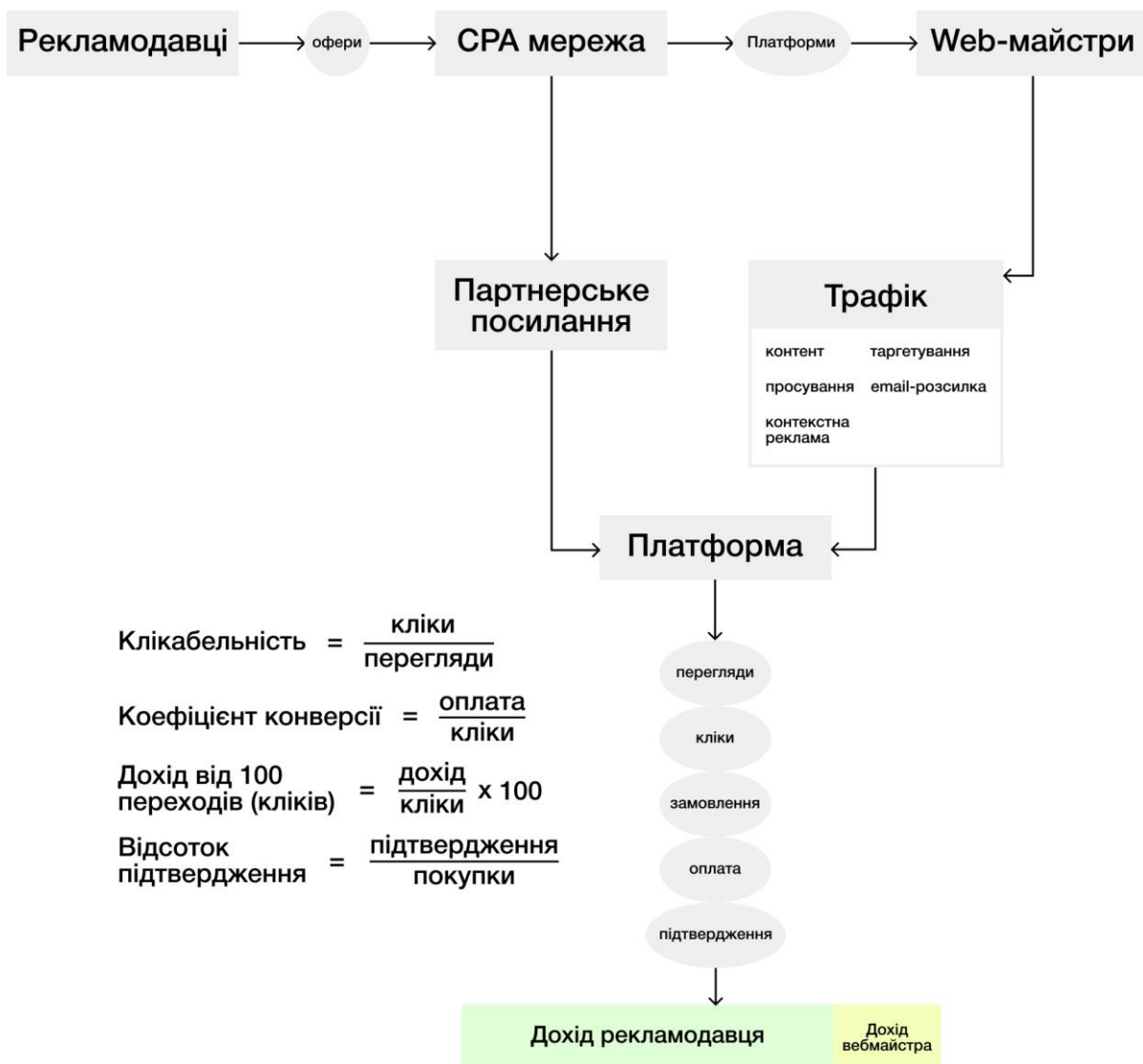


Рис 1.4 схема роботи CPA мережі та важливі показники[3]

1.2 Аналіз існуючих CPA мереж

CPA мереж зараз стає дедалі більше з розвитком CPA маркетингу, найбільш важливими критеріями для вибору є:

1. Кількість і якість ексклюзивних та вигідних оферів.
2. Період заморозки (холда), що кошти можна виводити через деякий час після підтвердження.
3. Кількість партнерів.
4. Кількість “Гео”, тобто місць/країн проживання цільової аудиторії.
5. Наявність захисних від шахрайства сервісів.
6. Наявність технічної підтримки, доступність менеджерів (так званих affiliate manager)

Наведемо приклади найбільших, та найрозвинутіших CPA мереж:

- “Le Elephant” — Платформа, яка спеціалізується в основному на електронній комерції, продаж товарів різного призначення через онлайн сторінки, надає обширну кількість оферів від крупних компаній та малого бізнесу.

Одразу приковує увагу незручність фільтрації та сортування оферів, а вірніше їх відсутність. Платформа має обширний вибір мікросервісів, що можуть допомогти вебмайстрам ефективізувати роботу. Одним із недостатків платформи можна вважати бюрократизовану систему становлення партнером платформи, потрібно пройти всі перевірки згідно законодавства, а також підтвердити якість і реальність послуги, що надається. Цей процес займає достатньо великий проміжок часу і, на мою думку, його можна було б оптимізувати.

- “Everad” — CPA мережа у вертикалі “нутра”, мають достатньо широкий список оферів і гео, але достатньо закриту їх базу, щоб отримати інформацію про них, потрібно стати партнером, а отже реєструючись ми тільки можемо зробити висновки по продаючій лендинг сторінці.

- “Affter” — українська CPA мережа у вертикалі іgaming. Працюють по моделі CPA та Revenue Share, Вийшли на ринок у 2023 році, управління проектом здійснюються досвідченими менеджерами, що працюють в сфері понад 6 років.

- “ADVGame” — CPA-мережа з ігровими оферами, просувають мобільні застосунки і мобільні ігри, в основному реклама в соцмережах і в інших додатках і іграх, обширна кількість оферів від найбільших виробників, таких як Super Cell, Gameloft та ін. Має порівняно невеликий вибір інструментів в особистому кабінеті.

- “Huge Offers” — партнерська мережа в сфері електронної комерції, надають офери фінансових та крипто брендів Має прозору систему перегляду оферів та статистики, але натомість не зручний застосунок псує досвід користування платформою (Рис. 1.5.).

- “Admitad” — одна з найдовж діючих на ринку, мають офери по різним вертикалям, в тому числі комп’ютерні ігри і електронну комерцію, мають обширну систему фільтрації оферів, але незручну навігацію між сервісами і особистим кабінетом

- “Get4Click” — партнерська мережа в сфері електронної комерції, надають офери брендів Samsung, Skyeng, Philips, Skillbox та інших. Має зручну систему кастомізації особистого кабінета, але натомість непрозору систему відбору в партнери з боку Web-майстрів і обмежену кількість сервісів.

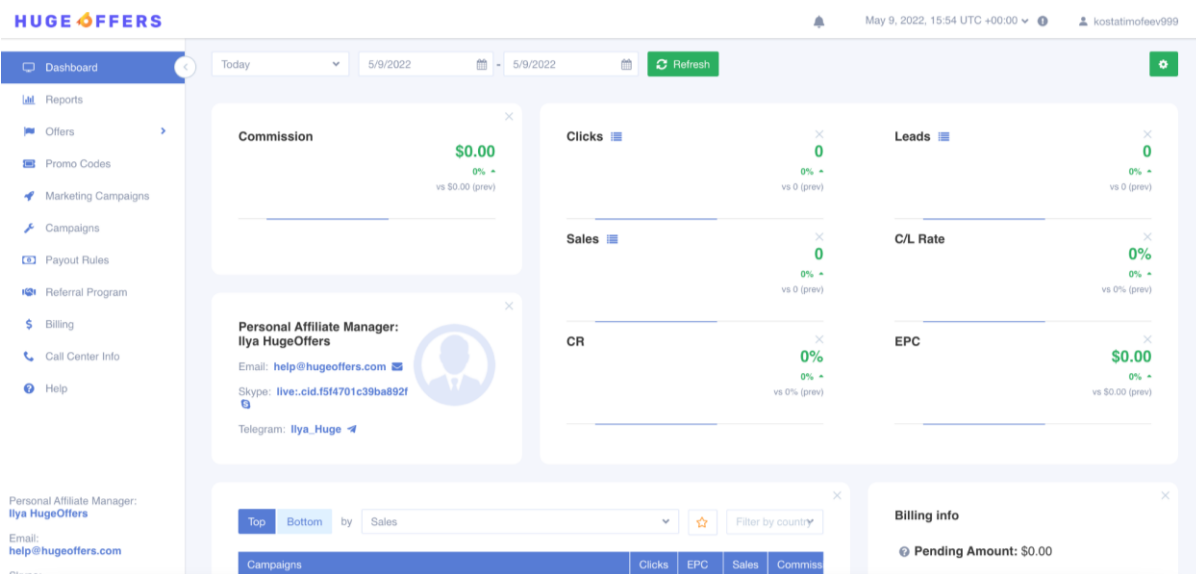


Рис. 1.5. сервіси CPA мережі “Huge Offers”

Висновок: більшість проаналізованих CPA мереж, як наведені вище, так і інші мають невелику кількість сервісів, обмежену систему фільтрації і офери в основному в одній або двох суміжних вертикалях, що може викликати дискомфорт у Web-майстрів, які працюють на декілька вертикалів.

Представлені вище партнерські мережі в основному не конкурують між собою і були обрані як лідери в своїх нішах, але як користувачу мені було достатньо складно орієнтуватись в інтерфейсі, всі представлені платформи мають приблизно однакову структуру блоків, що часто не дозволяє їх відрізнити між собою, особливо при наявності однакового кольорового рішення дизайну. Але загалом контент виглядає чисто і просто для сприйняття інформації, якщо не брати складні системи підтверджень заявок і незручні системи навігації та фільтрації. Більшість мереж мають свої вбудовані мікросервіси, і я зіткнувся з проблемою: або сервісів надто багато і мало корисного, або просто недостатньо для комфортної роботи в рамках мережі, а отже я маю використовувати та підключати сторонні сервіси, які також на даний час вже достатньо широко представлені в інтернеті.

Отже найбільшою потребою ринку наразі є розробка зручної CPA мережі з потенційно великою кількістю оферів з різних вертикалей, зі зручною системою фільтрації, відслідковуванням власної статистики виходячи з CPA моделі обчислень, а також представленими необхідними для роботи сервісами, в тому числі з використанням API технологій.

В кваліфікаційній роботі магістра було обрано дослідження цієї теми на прикладі крипто ніші, яка зараз є найновішою та ще не має розвинутих CPA мереж та сервісів, як наприклад мережі в ніші e-commerce та igaming.

Факторами порівняння за існуючими рішеннями стали: зручність інтерфейсу, розвинена система фільтрації, наявні необхідні вбудовані сервіси, швидка процедура реєстрації та підтвердження реєстрації, адаптивність сайту, сучасний дизайн, відслідковування своєї статистики, та можливість завантажити звіти (таблиця 1.1.).

Таблиця 1.1. Порівняння існуючих рішень за факторами

	Le Elephant	Everad	Affter	ADVGame	Admitad	Get4 Click	Huge Offers
Зручність інтерфейсу	+	-	+-	-	-	+	-
Розвинена система фільтрації	-	-	+	-	+	+	+-
Наявні необхідні вбудовані сервіси	+	-	-	-	+	-	+-
Швидка процедура реєстрації та її підтвердження	-	+	+	-	-	+	+
Адаптивність сайту	+	+	+	+	+	+	+
Сучасний дизайн	+	+	+	-	+	+-	+
Відслідковування своєї статистики	+	+	+	+	+	-	+
Можливість завантажити звіти	-	-	-	+	-	-	-

Актуальність розробки функціональної та зручної платформи для партнерського маркетингу наразі гостра, кожна CPA мережа має свої недоліки і позитивні сторони, але перед клієнтами і партнерами постає складне питання вибору через недосконалість та обмеженість всіх CPA мереж, що існують на даний момент.

Це може бути викликано порівняно невеликим віком галузі, яка розвивається швидкими темпами, компанії отримують більші прибутки і починають перелаштовуватись, але роблять це повільно. Альтернативних платформ з високим рівнем комфорту на даний момент не існує, або існують на рівні представлених. При розробці моєї платформи будуть враховані всі фактори, позитивні і негативні сторони існуючих для створення принципово нової та сучасної СРА мережі.

1.3 Аналіз існуючих методологій управління

Управління проектом розробки СРА платформи вимагає точного планування та використання ефективних методологій управління проектами для досягнення бажаних результатів. Ось детальний аналіз кількох ключових моделей життєвого циклу проєкту, методологій опису вимог та їх особливостей в застосуванні до розробки СРА платформи:

Таблиця 1.2. Аналіз моделей життєвого циклу

Модель ЖЦ	Визначення
Інкрементна модель (Incremental model)	<p>Відповідно до інкрементної моделі (англ. Increment - збільшення, прирощення) програмне забезпечення розробляється з лінійною послідовністю стадій, але в кілька інкрементів (версій). Таким чином поліпшення продукту проходить заплановано весь час, поки життєвий цикл розробки ПЗ не завершиться.</p> <p>Вимоги до системи визначаються на самому початку роботи, після чого процес розробки проводиться у вигляді послідовності версій, кожна з яких є закінченим і працездатним продуктом.</p> <p>Плюси:</p> <ul style="list-style-type: none"> ● замовник може дати свій відгук щодо кожної версії продукту; ● є можливість переглянути ризики, які пов'язані з витратами і дотриманням графіка; ● звикання замовника до нової технології відбувається поступово. <p>Мінуси:</p> <ul style="list-style-type: none"> ● функціональна система повинна бути повністю визначена на початку життєвого циклу для виділення ітерацій; ● при постійних змінах структура системи може бути порушена;

	<ul style="list-style-type: none"> • терміни здачі системи можуть бути збільшені через обмеженість ресурсів (виконавці, фінанси).
V-подібна модель (V-model)	<p>Дана модель стала послідовником каскадної моделі, так як з її допомогою можна усунути недоліки, які були раніше.</p> <p>Суть цієї моделі полягає в тому, що процеси на всіх етапах контролюються, щоб переконатися в можливості переходу на наступний рівень. Вже на стадії написання вимог починається процес тестування.</p> <p>Плюси:</p> <ul style="list-style-type: none"> • сувора етапізація; • мінімізація ризиків і усунення потенційних проблем за рахунок того, що тестування з'являється на самих ранніх стадіях; • вдосконалений тайм-менеджмент. <p>Мінуси:</p> <ul style="list-style-type: none"> • неможливість адаптуватися до змінених вимог замовника; • тривалий час розробки (іноді триває до декількох років) призводить до того, що продукт може бути вже не потрібний замовнику, оскільки його потреби змінюються; • немає дій, спрямованих на аналіз ризиків.
Каскадна модель	<p>При такій моделі кожна з фаз проекту проводиться один раз, слідуючи одна за одною. Для того, щоб почати наступну стадію, необхідне повне завершення попередньої.</p> <p>Плюси:</p> <ul style="list-style-type: none"> • всі стадії проекту виконуються в чіткій послідовності; • чіткість етапів дозволяє планувати терміни завершення всіх робіт і відповідні ресурси (грошові і людські); • вимоги залишаються незмінними протягом усього циклу. <p>Мінуси:</p> <ul style="list-style-type: none"> • складності при формулюванні чітких вимог і неможливість їхньої зміни; • тестування починається тільки з середини розвитку проекту; • до завершення процесу розробки користувачі не можуть переконатися, чи якісний продукт, який розробляється.
Спіральна модель (Spiral model)	<p>У спіральній моделі життєвий шлях продукту, що розробляється зображується у вигляді спіралі, яка, розпочавшись на етапі планування, розкручується з проходженням кожного наступного кроку. Таким чином, на виході з чергового витка отримуємо готовий протестований прототип, який доповнює існуючу збірку. Прототип, що задовольняє всі вимоги, готовий до випуску.</p> <p>Плюси:</p>

	<ul style="list-style-type: none"> ● приділяється особлива увага управлінню ризиками; ● додаткові функції можуть бути додані на пізніх етапах; ● є можливість гнучкого проектування. <p>Мінуси:</p> <ul style="list-style-type: none"> ● оцінка ризиків на кожному етапі є досить витратною; ● постійні відгуки і реакція замовника може провокувати все нові і нові ітерації, які можуть призводити до тимчасового затягування розробки продукту; ● більш застосовується для великих проєктів.
Гнучка модель (Agile model)	<p>Являє собою сукупність різних підходів до розробки ПЗ. Включає серії підходів до розробки програмного забезпечення, орієнтованого на використання ітеративної розробки (в Scrum ітерації називаються спринтами), динамічне формування вимог і забезпечення їхньої реалізації в результаті постійної взаємодії всередині самоорганізованих робочих груп, що складаються з фахівців різного профілю. Окрема ітерація являє собою мініатюрний програмний проєкт. Однією з основних ідей Agile є взаємодія всередині команди і з замовником напряму.</p> <p>Плюси:</p> <ul style="list-style-type: none"> ● швидке прийняття рішень завдяки постійним комунікаціям; ● мінімізація ризиків; ● полегшена робота з документацією. <p>Мінуси:</p> <ul style="list-style-type: none"> ● велика кількість мітингів і обговорень, що може збільшити час розробки продукту; ● складно планувати процеси, так як вимоги постійно змінюються; ● рідко використовується для реалізації великих проєктів.

Проаналізувавши позитивні та негативні сторони, було обрано модель Agile, Розглянемо кілька основних фреймворків та оберемо найоптимальніший для реалізації проєкту.

1.3.1 Scrum

Особливості:

Ролі в команді: Ясно визначені ролі (Scrum Master, Product Owner, Development Team) сприяють чіткій організації та відповідальності.

Регулярні зустрічі: Щоденні стендапи, спринт-планування, ретроспективи та перегляди спринтів допомагають підтримувати високий рівень комунікації та прозорості проекту, але при цьому можуть збільшити кількість часу виконання проекту.

Артефакти Scrum: Product Backlog, Sprint Backlog, інкремент продукту — це інструменти, які допомагають команді візуалізувати прогрес і планувати роботу.

Застосування до SPA платформи:

Scrum є ефективним у розробці SPA платформ завдяки своїй структурованості та зосередженості на досягненні короткострокових цілей через спринти. Це дозволяє розробникам швидко пристосовуватися до змін у вимогах і технологіях, а також забезпечує постійне впровадження нововведень.

1.3.2 Kanban

Особливості:

Візуалізація роботи: Користування Kanban-дошкою допомагає команді бачити статус усіх завдань у реальному часі.

Обмеження поточної роботи: Покращує фокус та продуктивність, мінімізуючи кількість одночасно виконуваних завдань та тиску, що може виникнути з боку менеджменту.

Покращення потоку: Наголос на зменшенні часу циклу завдань та збільшенні постійної доставки.

Застосування до SPA платформи:

Kanban може бути корисним при управлінні неперервними покращеннями та оптимізацією процесів після запуску платформи. Ця система дозволяє ефективно реагувати на потреби в технічній підтримці, обслуговуванні і масштабуванні продукту.

1.3.3 Scrumban

Особливості: Комбінація Scrum і Kanban, зосереджена на гнучкості та відповідності змінам.

Позитиви: Забезпечує більшу гнучкість порівняно з чистим Scrum, дозволяє краще реагувати на зміни, зменшує час на планування.

Негативи: Може бути складним для команд без досвіду у Kanban або Scrum, вимагає сильних навичок самоорганізації.

Застосування: Scrumban ідеально підходить для проектів, де потрібна висока адаптивність до змінюваних вимог користувачів та ринку. У вашому проекті це допоможе швидко реагувати на зміни в маркетингових стратегіях CPA мереж, дозволяючи команді ефективно впроваджувати нововведення і вдосконалювати функціонал системи без довгих планувань.

1.3.4 Design Thinking

Особливості: Процес, орієнтований на ітерації, емпатію та прототипування для вирішення проблем.

Позитиви: Сприяє інноваціям та креативності, зосереджується на користувачеві.

Негативи: Може вимагати більше часу для розвитку концепції, не завжди легко інтегрувати з жорсткими вимогами до часу випуску.

Застосування: Цей підхід може бути використаний на початкових етапах проекту для глибокого аналізу потреб користувачів CRM-системи. Робочі групи можуть організовувати сесії з зацікавленими сторонами для визначення основних болів та потреб, що призведе до створення інноваційного та ефективного продукту.

1.3.5 Agile Project Management (AgilePM)

Особливості: Баланс між чітким плануванням та гнучкістю.

Позитиви: Забезпечує структуру та дисципліну, допомагає управлінню більшими проектами.

Негативи: Може бути менш гнучким порівняно з іншими Agile методиками.

Застосування: AgilePM забезпечить структурований, але гнучкий підхід до управління проектом, дозволяючи чітко визначити ролі, відповідальності та етапи впровадження. Це допоможе управлінню проектом в умовах, коли потрібно скоординувати різноманітні завдання та вимоги стейкхолдерів.

1.3.6 Scaled Agile Framework® (SAFe®)

Особливості: Рамка для масштабування Agile на великі проекти та комплексні організації.

Позитиви: Ефективна координація між різними командами, включає чіткі ролі та відповідальності.

Негативи: Складність та велика навантаження документації можуть сповільнити процес.

Застосування: Якщо ваш проект масштабний і передбачає велику кількість команд, SAFe® допоможе організувати робочий процес так, щоб всі команди мали єдині напрямки розвитку і чітко синхронізовані цілі, забезпечуючи високу продуктивність і координацію.

1.3.7 Large-Scale Scrum (LeSS)

Особливості: Фокус на масштабуванні Scrum без втрати його основних принципів.

Позитиви: Мінімізує бюрократію, сприяє прозорості та співпраці.

Негативи: Може вимагати сильних змін у організаційній структурі.

Застосування: LeSS можна використовувати для спрощення процесів у великих командах без втрати основних принципів Scrum. Це знижує бюрократію і збільшує прозорість робочих процесів, що важливо для швидкого та ефективного розгортання CRM-системи.

1.3.8 Nexus

Особливості: Рамка для інтеграції декількох Scrum команд.

Позитиви: Дозволяє ефективно масштабувати Scrum, підтримує зв'язки між командами.

Негативи: Високі вимоги до координації та спільної роботи.

Застосування: Nexus підходить для проектів, де декілька Scrum-команд працюють над спільним продуктом. Це може допомогти уникнути проблем з інтеграцією та забезпечити високу якість фінального продукту, координуючи зусилля команд.

1.3.9 AgileSHIFT

Особливості: Цілісний підхід до внутрішньої культури змін.

Позитиви: Сприяє культурі адаптації та гнучкості в усій організації.

Негативи: Може вимагати значних культурних змін, що є викликом для традиційних організацій.

1.3.10 Open Space Agility (OSA)

Особливості: Використання Open Space Technology для сприяння швидкому і ефективному впровадженню змін.

Позитиви: Стимулює інновації та активну участь всіх учасників.

Негативи: Може бути непередбачуваним, важко контролювати великі групи.

1.3.11 Agility Scales

Особливості: Центрується на масштабуванні гнучких методів шляхом використання технологій.

Позитиви: Надає інноваційні інструменти для підтримки трансформації.

Негативи: Вимагає високого рівня технічної оснащеності, може бути складним у впровадженні.

Кожен із цих фреймворків має свої сильні та слабкі сторони, і вибір має базуватися на специфічних потребах проєкту, культурі компанії, та вимогах до масштабування і співпраці команд.

1.3.12 Вибір фреймворку

Ці фреймворки Agile методології мають різні підходи до управління проєктами і можуть бути адаптовані для різних фаз розробки СРА платформи, забезпечуючи гнучкість, ефективність та успішність проєкту в довгостроковій перспективі. Але з огляду на ціль завершити проєкт в короткі терміни та з найбільш ефективним використанням наявних ресурсів було обрано методологію *KanBan* за простоту використання, візуалізацію та одночасні процеси, що відбуваються в різних командах.

1.4 Аналіз Методик опису вимог

Оглянемо кілька основних методик опису вимог (Таблиця 1.3)

Таблиця 1.3. Аналіз методик опису вимог

Методології опису вимог	Визначення
IEEE 830-1998	Цей стандарт, розроблений Інститутом інженерів електроніки та електротехніки (IEEE), містить рекомендації щодо написання добре структурованих вимог до програмного забезпечення. Стандарт описує,

	як вимоги повинні бути чіткими, повними, консистентними та перевіряємими. Він також рекомендує включати функціональні, нефункціональні вимоги, а також інтерфейси і обмеження.
Volere Requirements Specification Template	Volere є шаблоном для специфікації вимог, який забезпечує комплексний підхід до збору та аналізу вимог. Шаблон включає різні розділи, такі як цілі проекту, обмеження, припущення, функціональні та нефункціональні вимоги, сценарії використання, інтерфейси користувача, і так далі. Volere також наголошує на важливості залучення стейкхолдерів у процесі визначення вимог..
BABOK (Business Analysis Body of Knowledge)	Розроблений Міжнародним інститутом бізнес-аналізу (ІБА), BABOK є одним з провідних стандартів для бізнес-аналізу, що включає детальні методики збору та аналізу вимог. Стандарт описує різні техніки еліцитації вимог, їх аналіз, документування та управління протягом життєвого циклу проекту.
FURPS+	Це акронім, який використовується для класифікації вимог до програмного забезпечення: Functionality, Usability, Reliability, Performance, Supportability та "+", який включає додаткові категорії, такі як Design, Implementation, Interface, Physical та Others. Цей стандарт допомагає забезпечити всеосяжний підхід до оцінювання та специфікації вимог до продукту.

Оскільки Kanban не є традиційним проектним методом з чіткими вимогами до специфікації або документування, він зазвичай використовується разом з іншими методиками опису вимог для досягнення кращих результатів. Хоча це не методика опису вимог, підходи з AgilePM можуть інтегруватися в Kanban, особливо в аспектах, що стосуються гнучкого управління проектами і взаємодії зі стейкхолдерами. AgilePM може допомогти в структуруванні вимог та забезпеченні чітких напрямків для команди, що працює з Kanban. Отже при визначенні вимог в проєкті створення веб додатку CRM системи буде використано гнучкий підхід, використання цієї методики з Kanban може забезпечити краще розуміння та виконання проектних вимог, сприяючи ефективному потоку робіт і підвищенню загальної продуктивності проекту.

1.5 Аналіз основних процесів предметного середовища

Однією із задач проєкту є створити інформаційну систему для CPA мережі, використовуючи знання про CPA маркетинг, процеси та технології створення подібних систем з метою розробити найбільш ефективне рішення в умовах сучасного ринку. Адміністратор може переглядати дії та підтверджувати офери для користувачів, користувач, у свою чергу може управляти своїми транзакціями, оферами, формувати звіти по конкретним оферам, фільтрувати їх по категоріям, а також переглядати статистику в реальному часі, використовуючи математичні моделі системи, у зручному інтерфейсі.

Задачами є:

1. Провести аналіз літератури
2. Провести порівняльний аналіз існуючих ІС
3. Проаналізувати процес роботи Web-застосунку для CPA-мережі
4. Розробити ефективну CRM систему для користувача платформи

Складові елементи:

1. База даних оферів
2. Підсистема авторизації
3. База даних користувачів
4. Модуль менеджменту оферами
5. Математичний модуль відстежування статистики
6. Модуль фільтрації оферів
7. Модуль поповнення балансу

Опис профілів зацікавлених сторін:

Внутрішні зацікавлені сторони:

- Розробник сайту
- Адміністратор сайту

Зовнішні зацікавлені сторони:

- Клієнти/Вебмайстри
- Рекламодавці

Модель оплати CPA розглянута в першому пункті – це вершина еволюції digital advertising. Головна її цінність у тому, що мережа дає рекламодавцям можливість використовувати майданчики та формати, недоступні у стандартних каналах. Ідея в тому, що самостійно CPA мережа не генерує трафік. Для цього організація співпрацює з тисячами веб-майстрів, які є власниками сайтів, груп у соцмережах, Youtube-каналів або вміють добре налаштовувати різні види інтернет реклами для рекламодавців, які готові платити гроші за цільову дію на своєму web-ресурсі.

Ринок affiliate в Україні якраз перебуває на стадії становлення та формування. Є відчутна необхідність стимулювання зростання веб-майстрів, появи, створенні нових майданчиків. Якщо раніше, щоб заробити в партнерській мережі, потрібно було робити якийсь сайт, розбиратися з безліччю нюансів, то зараз є варіанти, як це можна зробити швидше та в рази простіше. Наприклад, звернутися до актуальних трендів генерації трафіку: Telegram-канали, YouTube, Instagram та Tik Tok.

Можна виділити такі тренди:

1. Збільшення мобільного трафіку та обсягів продажів через mobile завдяки запуску в країні технологій 4G+.
2. Зростання популярності відеоконтенту, який активно використовується для залучення трафіку рекламодавцями та веб-майстрами.
3. Розвиток нових каналів трафіку: месенджери, онлайн-трансляції, модель CPA-offline, SMS, YouTube, Instagram, Tik Tok, сарафанне радіо, термінали та голосовий пошук.

Також можна зробити висновок щодо актуальних методів та модулів при проектуванні CPA мережі:

1. Наскрізна аналітика – трафік – продаж онлайн. Інтеграція за API з банками та мікро фінансовими компаніями для отримання онлайн статистики від них.
2. Технологічність - нові сервіси, процеси для клієнтів та веб майстрів. Наприклад, ми даємо веб майстрам шаблон, який легко інтегрувати в сайт і він автоматично підбирає кредиторів під профіль клієнта в онлайн.

3. Використання SMS, IVR, push-повідомлень, чат-ботів для максимальної монетизації трафіку.

Основною задачею є створення адаптивної, функціональної та зручної системи (веб-додатку) для CPA-мережі, використовуючи математичні CPA-моделі і сучасні технології та інструменти для розробки систем. Система має містити функціонал для управління, редагування оферів та сервісів платформи.

На першій діаграмі відображено процес «відкриття оферу» (рис. 1.6). На вході ми отримуємо персональні дані веб майстра, авторизаційні дані та дані про вертикаль роботи.

На виході, після виконаного процесу, ми отримуємо відкритий офер, його статистичні дані, що будуть відображатись списком та через математичний модуль застосунку.

Варто зазначити важливість дотримання закону України щодо “збору та обробки персональних даних” користувача, якими має керуватись та регулюватись продукти, які вимагають від користувача передачу певних персональних даних та гарантувати йому конфіденційність і безпеку, на що також посилаються користувацькі умови і положення.

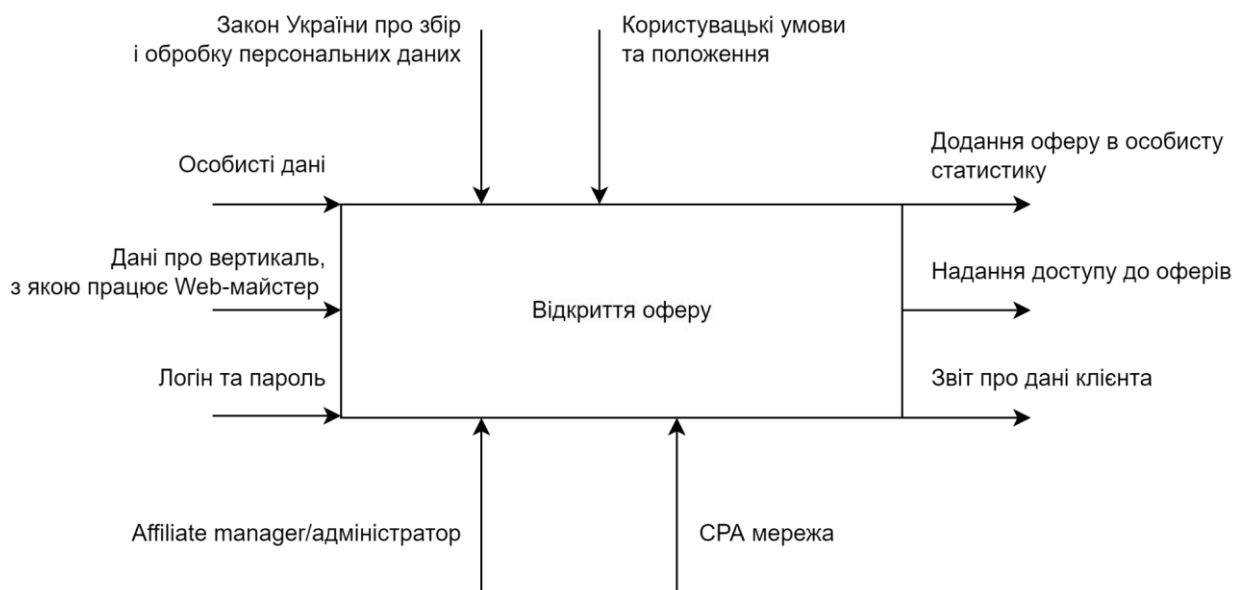


Рис. 1.6 контексна діаграма ЯК Є

Розглянемо детальніше описаний процес, побудувавши декомпозицію його основних функцій (рис. 1.7) та проведемо більш детальний аналіз процесу реєстрації на платформі (рис. 1.8). Для початку роботи веб майстер має пройти реєстрацію на платформі, після чого його дані обробляються системою і за ним закріплюється affiliate manager, який аналізує статистику веб майстра під час відкриття оферу.

У представлених діаграмах (рис. 1.7-1.8) описується процес становлення учасником і партнером CPA мережі та отримання доступу до відкритого оферу.

Така складна схема реєстрації та підтвердження облікового запису потрібна, щоб менеджер, який супроводжує партнера провів особисту листування з ним та впевнився в достовірності наданої інформації, та підтвердив експертизу партнера щодо трафіку та його майстерності лити.

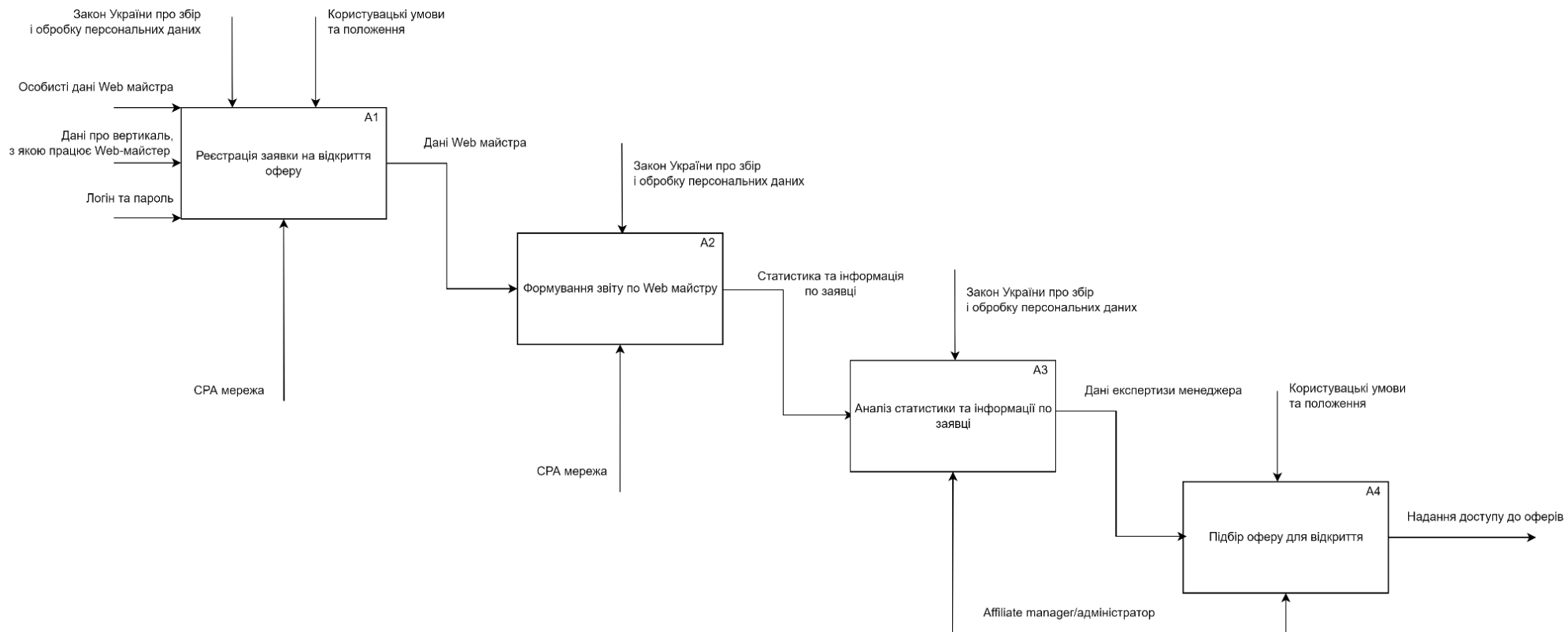


Рисунок 1.7 — Деталізована IDEF0 діаграма процесу відкриття оферу “ЯК Є”

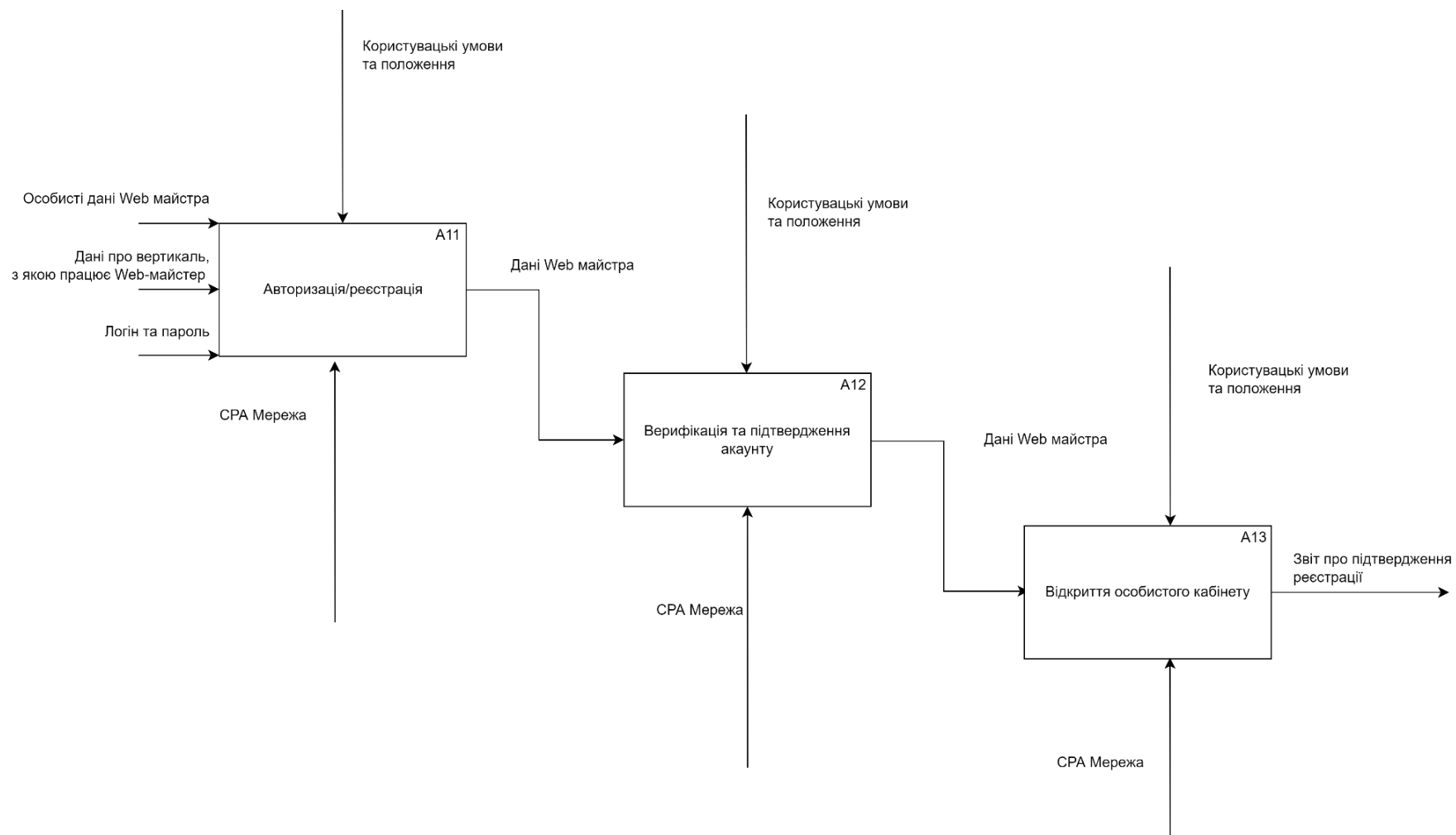


Рисунок 1.8 — Деталізація блоку А1 “JK Є”

1.6 Постановка задачі на розробку веб застосунку для CPA мережі

Основною задачею проекту є створення адаптивної, функціональної та зручної системи (веб-додатку) для CPA-мережі, використовуючи математичні CPA-моделі і сучасні технології та інструменти для розробки систем. Система має містити функціонал для управління, редагування оферів та сервісів платформи.

Основні функціональні вимоги до веб застосунку:

1. Реєстрація та авторизація користувачів, підтвердження акаунту.

Стандартне рішення для всіх систем, додатків, платформ і сервісів задля безпеки особистих даних та цілісності системи. Користувач повинен мати свій особистий логін та пароль для користування системою, доступу до особистої статистики, а також задля безпеки особистих даних і фінансів.

2. Привабливість, зручність та ефективність інтерфейсу у взаємодії з користувачем.

Я задію свій досвід у розробці інтерфейсів, щоб створити максимально зручний та функціональний, згідно з правилами та принципами UX/UI дизайну, проведу дослідження, виявлю проблематику та запроваджу найбільш ефективне рішення

3. Модуль відкриття оферу та присвоєння користувачеві

Особливість розроблюваної платформи, необхідний функціонал для користувача і для адміністратора, відкриття офера одному користувачеві унеможливорює відкриття того ж офера іншому користувачеві.

4. Система підрахунку показників у реальному часі

В особистому кабінеті на головній сторінці реалізована можливість у реальному часі відслідковувати показники по своїй статистиці, використовуючи математичні CPA-моделі. Користувач повинен мати змогу швидко візуально оцінити динаміку показників.

5. Індивідуальна система адміністратора з можливістю управління та редагування оферів та інформацією користувачів.

Необхідна для ефективного управління системою в бізнес проектах

6. Сервіс для виділення працюючих телефонних номерів на визначений час по заданій геолокації.

Додатковий модуль для більшої функціональності платформи, який закриває необхідність користувачів в сторонніх сервісах.

7. Система збереження та менеджменту статистикою та оферами.

Користувач повинен мати можливість завантажувати звіти, по його оферам.

8. Зручна система фільтрації.

Необхідна для швидкого знаходження потрібної інформації в системі.

9. Модуль поповнення балансу та виведення виплат на банківський рахунок

Користувач повинен мати можливість виводити зароблені кошти зручним для нього методом оплати, також для більшої зручності поповнювати баланс, щоб потім кошти розподіляти між рекламними кампаніями.

10. Додатковий сервіс паркування доменів що дозволяє реєструвати доменні імена на DNS серверах, дозволяє зберегти за собою право використовувати доменне ім'я для своїх сервісів або сайтів.

Нефункціональні вимоги:

1. Підтримка різних мов.

Стандартне рішення для всіх систем, додатків, платформ і сервісів задля зручності користування та комфорту користувачі з різних країн. Найпопулярніше рішення це використання англійської та української мови

2. Кастомізація теми.

Можливість змінювати колір інтерфейсу на власний смак, тобто користувач зможе налаштувати світлу/темну тему застосунку.

1.6 Висновки до першого розділу

Були проаналізовані основні методології управління проектами, була обрана методологія Agile за її гнучкість та можливість розробляти проєкт в умовах постійних змін та невеликої кількості залучених працівників, також після детального аналізу фреймворків було обрано фреймворк KanBan за можливість візуального представлення задач та невелику кількість необхідних мітингів команди на спринті.

Були проаналізовані основні методики визначення вимог і сформовані задачі проєкту. Обґрунтовано вибір предметної галузі та розкрито суть наукової новизни проєкту.

На даний момент на ринку мало детально спроектованих CPA мереж, так як це найсучасніший вид маркетингу, який достатньо швидко набирає популярність, особливо в новій ніші криптовалют, які вже завоювали прихильність аудиторії та її потребу і ній, а з цього випливає необхідність в створенні системи, що підтримує різні ніші партнерського маркетингу, є ефективним інструментом та зручною системою для користувачів. Задіявши досвід та набуті навички, буде створено бізнес-рішення, продукт, що забезпечить користувача рішенням його потреб. Такий проєкт має високу цінність та конкурентоспроможність, так як всі подібні не мають всіх представлених систем і сервісів, а також не зручні у використанні. Створювана система має бути гнучкою до змін з можливістю адаптовувати її під різні потреби завдяки математичним модулям та гнучкій системі редагування інформації. Тож система являє собою мультизадачну платформу для виконання затребуваних задач бізнесу і є вкрай необхідною в сучасності.

РОЗДІЛ 2. РОЗРОБКА АРХІТЕКТУРИ WEB ДОДАТКУ CRM СИСТЕМИ ДЛЯ СРА МЕРЕЖІ

2.1 .Розробка архітектури продукту

2.1.1 Функціональний аналіз

Побудова діаграм є чи не найголовнішим етапом розробки системи. Зокрема, якщо ми розглядаємо кінцеву ціль як бізнес продукт, ми маємо уявити картину функціонування, отримання вигоди та дослідити поведження користувача. Отже в цілому представлення про продукт ми сформуваали в першому розділі, але тепер потрібно ще структурувати в тому числі бізнес-процеси. В основі будь-якої платформи такого роду лежить дві версії, а саме користувацька і версія для адміністратора, в моєму випадку менеджера.

Affiliate менеджер виконує функцію адміністратора, додає, видаляє, редагує, а найголовніше - відкриває доступ до оферів користувачу (Web майстру).

Web майстер зі свого боку має доступ до математичного модуля, що вираховує статистику та дохід з його трафіку, звичайно що система має модуль реєстрації та модуль відкриття оферу. Web майстер має змогу обирати та надсилати запити на відкриття оферу (рис. 2.1) менеджеру і в подальшому, після відкриття, користуватись платформою.

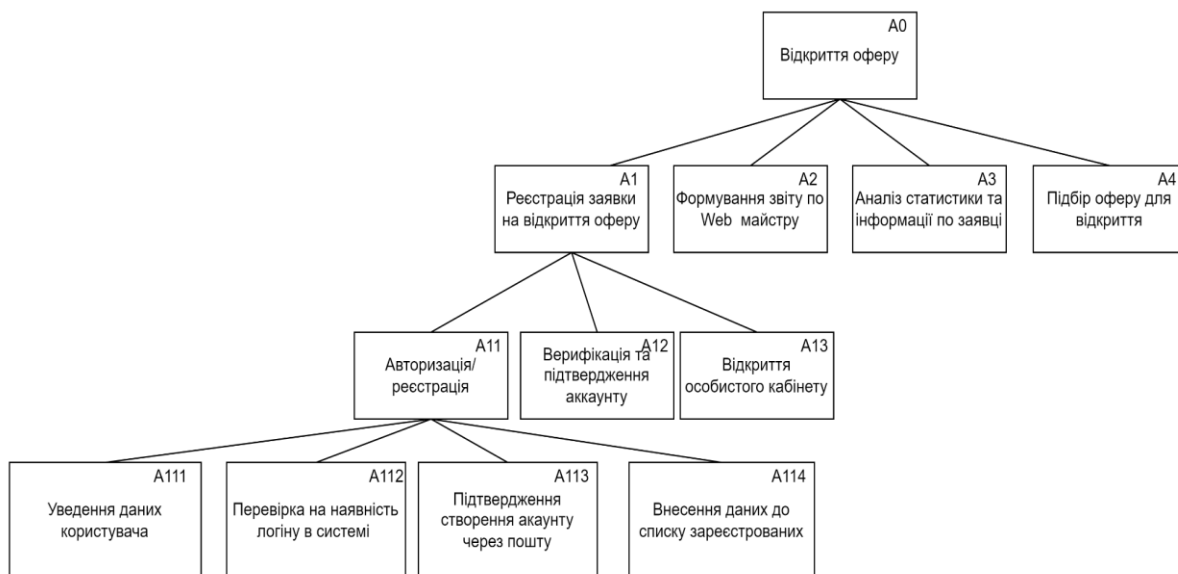


Рис. 2.1. Дерево функцій на прикладі Відкриття оферу

Таким чином, спочатку користувач має зайти на сайт і, натиснувши кнопку “увійти”, ввести свій логін та пароль та авторизуватись, або зареєструватись, якщо ще немає акаунта в системі і потім проробити описані вище кроки. За умови реєстрації, дані клієнта записуються до списку web майстрів. Якщо він вже був зареєстрований, то система перевіряє наявність авторизаційних даних у системі. На головній сторінці користувач має можливість у зручному форматі користуватися автоматичним математичним модулем розрахунку статистики, переглянути динаміку та короткі контекстні меню переходу на сторінку офферів. Офери спочатку знаходяться не закріпленими ні за ким і для відкриття користувачеві потрібно надіслати запит на потрібний оффер, який система та affiliate-менеджер оброблюють. Після обробки запиту оффер додається на сторінку “Мої офери” та закріплюються в базі даних за ним, після цього майстер може переходити до налаштувань і працювати з офером, підключивши його до Google або Facebook Ads. Дані процеси відображено на діаграмі Event-Driven Process Chain (рис. 2.2, 2.3).

Отже до бізнес процесів можна додати діаграму ланцюжка процесів, де основний процес це робота з платформою, що поділяється на підпроцеси, в тому числі реєстрацією, відкриттям оферів, відслідковуванням статистики, переглядом та редагуванням особистих даних та налаштуванням особистих оферів, знаходженням потрібного, відслідковування статистики по кожному окремому оферу, процеси поєднані між собою, мають декілька рівнів ієрархії та розташовані послідовно (рис. 2.4).

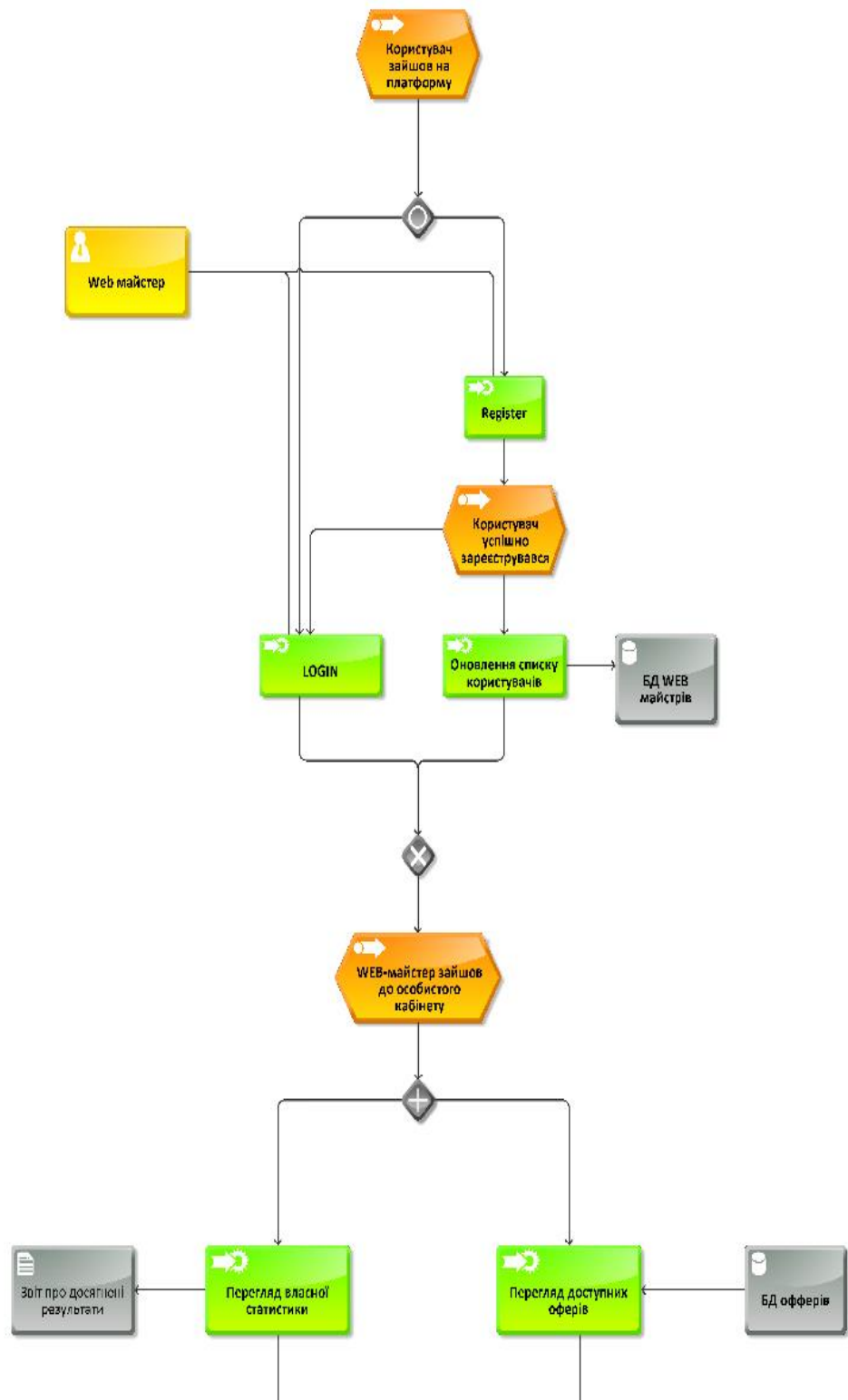


Рисунок 2.2. Діаграма Event-Driven Process Chain

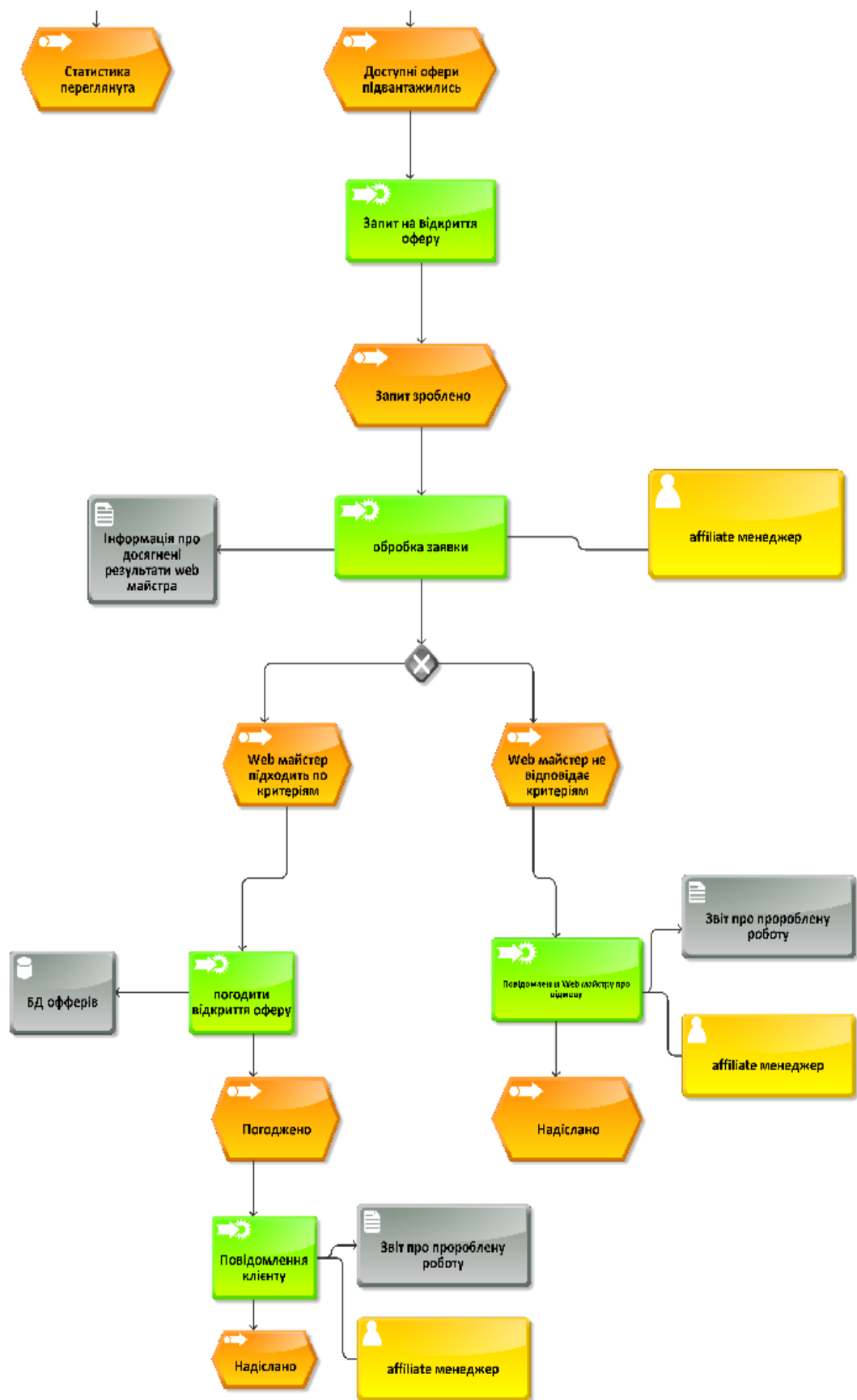


Рис. 2.3 продовження діаграми Event-Driven Process Chain

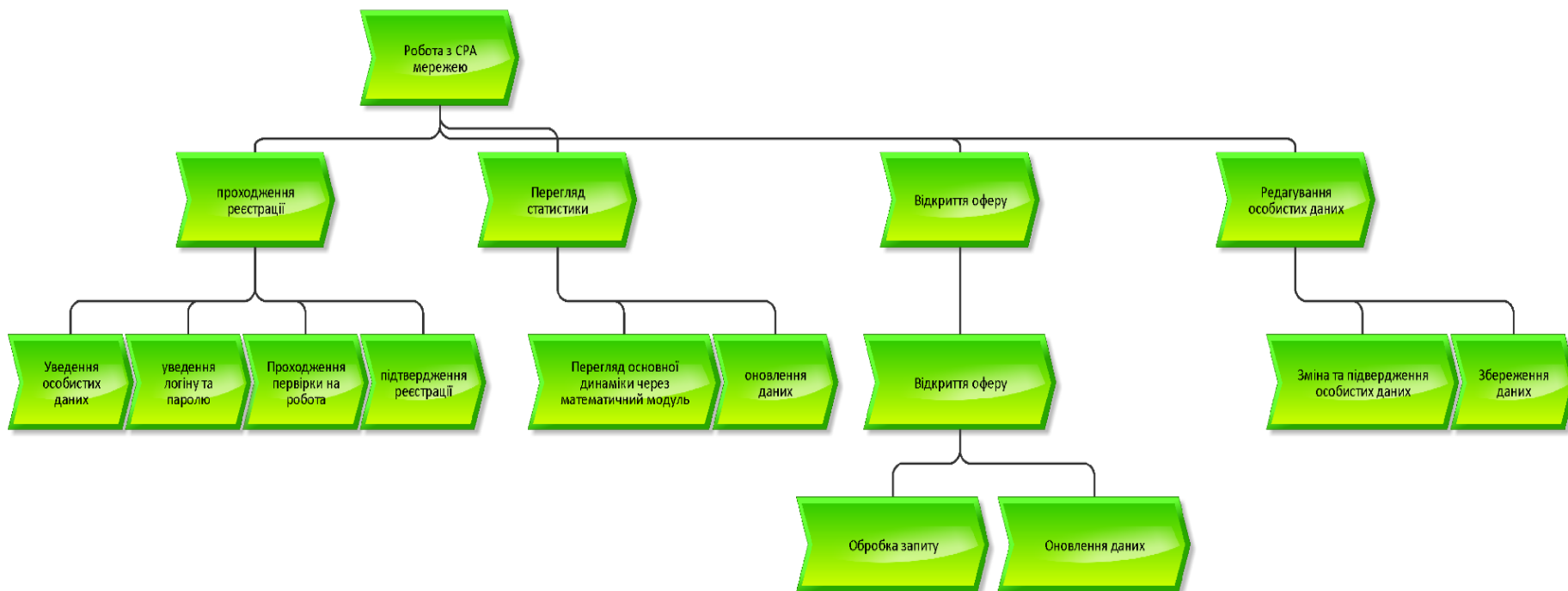


Рис. 2.4. Діаграма ланцюжка процесів Value Added Chain

2.1.2 IDEF0 процесу відкриття оферу

Представимо виконувані процеси у предметному середовищі за допомогою діаграм «ЯК БУДЕ» у нотації IDEF0, тобто вже з використанням web застосунку.

Спочатку будемо контекстну діаграму, де вихідні дані це особисті та авторизаційні дані web майстра, а на виході отриманий звіт по статистиці оферу і дані про офер веб майстра.

Елементом керування є закон України про «Обробку і зберігання персональних даних», а також користувацькі умови та положення, так як користувач дає нам згоду на обробку його персональних даних, також збір статистики його веб переглядів і ми гарантуємо їх безпеку та захист.

Механізмами у даному випадку є CPA мережа та affiliate менеджер.

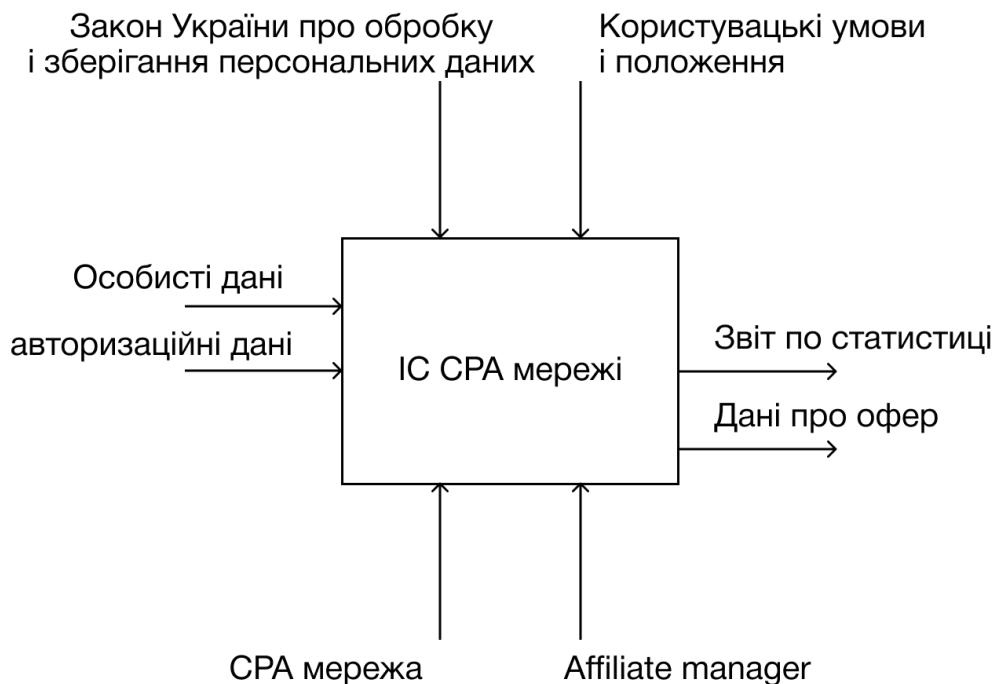


Рис. 2.5. Контекстна діаграма “ЯК БУДЕ”

На наступному етапі проведемо декомпозицію основних процесів діаграми ЯК БУДЕ (рис. 2.6).

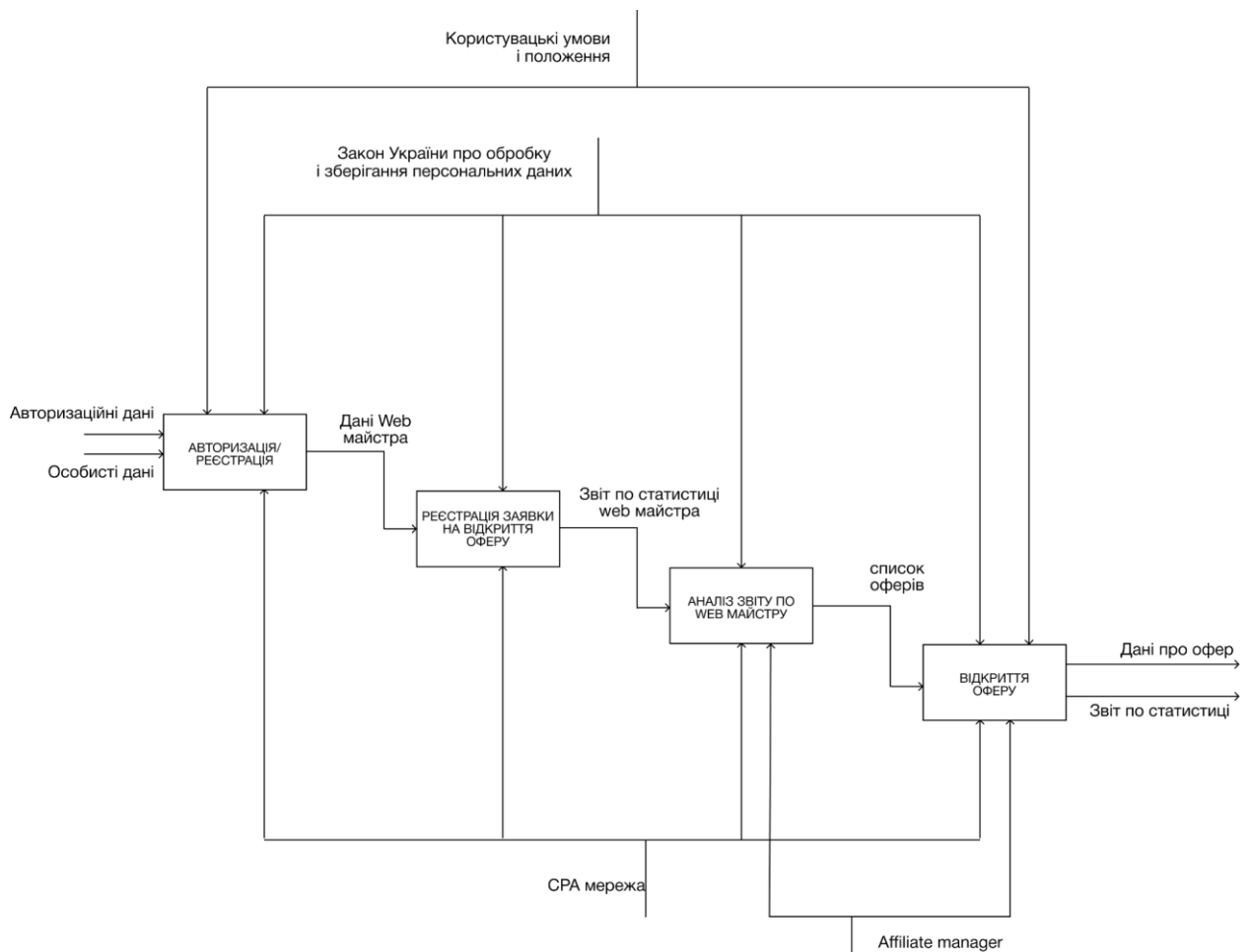


Рис. 2.6. Деталізована IDEF0 діаграма “ЯК БУДЕ”

Спочатку проходить авторизація користувача в системі, після чого дані користувача передаються в системі далі на реєстрацію заявки на відкриття оферу, система і менеджер аналізують статистичні показники веб майстра, на основі чого можуть приймати рішення щодо відкриття оферу для нього. На виході отримуємо офер, його дані і звіти про статистику.

2.1.3 Архітектура інформаційної системи

Наступний крок це побудова архітектури системи, що включає в себе модулі відповідно до функцій системи. Загалом присутні 3 модулі: клієнтський, математичний, модуль affiliate менеджера (рис. 2.7).

Клієнтський модуль включає модуль обробки персональних даних, а також пошуку та фільтрування оферів відповідно до потреб користувача

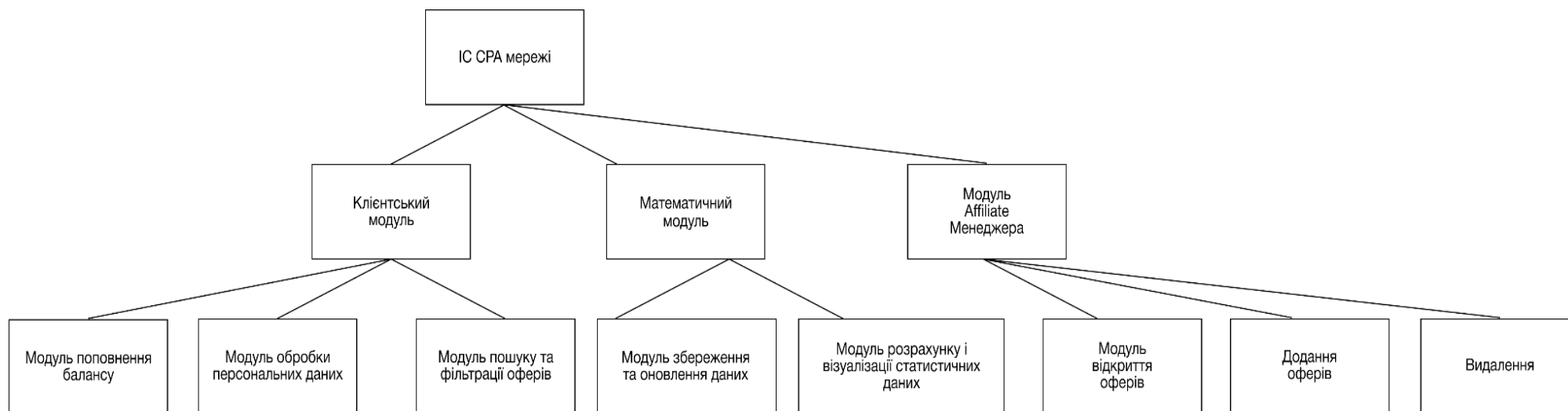


Рис. 2.7. Архітектура Інформаційної системи

2.1.4 Дерево причин та наслідків (діаграма Ішикави)

Діаграма Ішикави, також відома як діаграма риб'ячого скелета або діаграма причин і наслідків, дозволяє систематично аналізувати причини проблеми. Це важливий інструмент у управлінні якістю та аналізі процесів. Для визначеної проблеми повільного відкриття оферу, можна побудувати таку діаграму (рис.2.8):

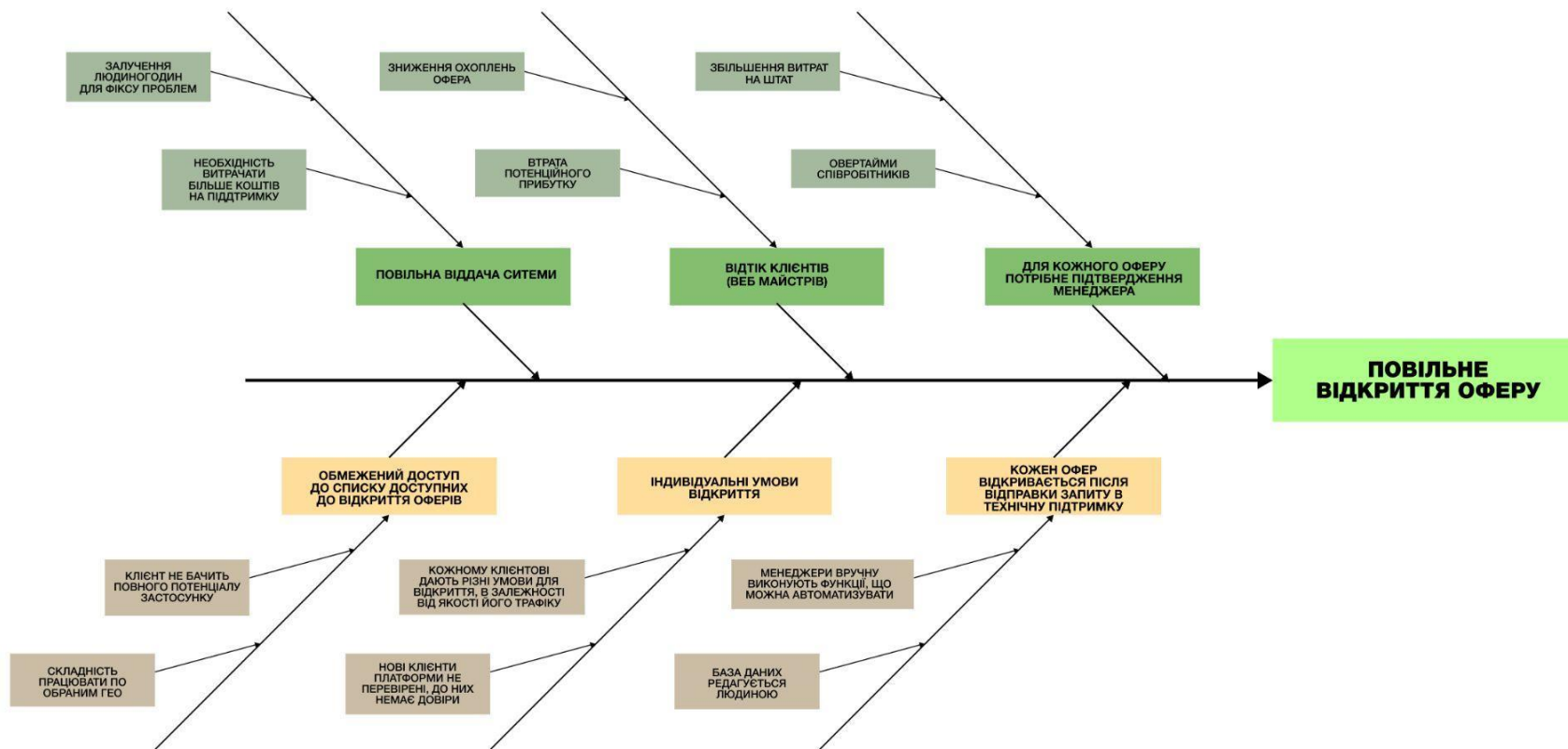


Рис. 2.8. Діаграма Ішикави

2.1.5 Дерево проблем та рішень

Dod: Усунення проблеми повільного відкриття оферу для веб майстра

Problem: Повільне відкриття оферу

Дерево проблем та рішень є корисним інструментом для візуального представлення проблем та їх потенційних рішень у ієрархічній структурі. Це дозволяє аналізувати проблему повільного відкриття оферу та визначати можливі дії для її вирішення (Рис. 2.9).

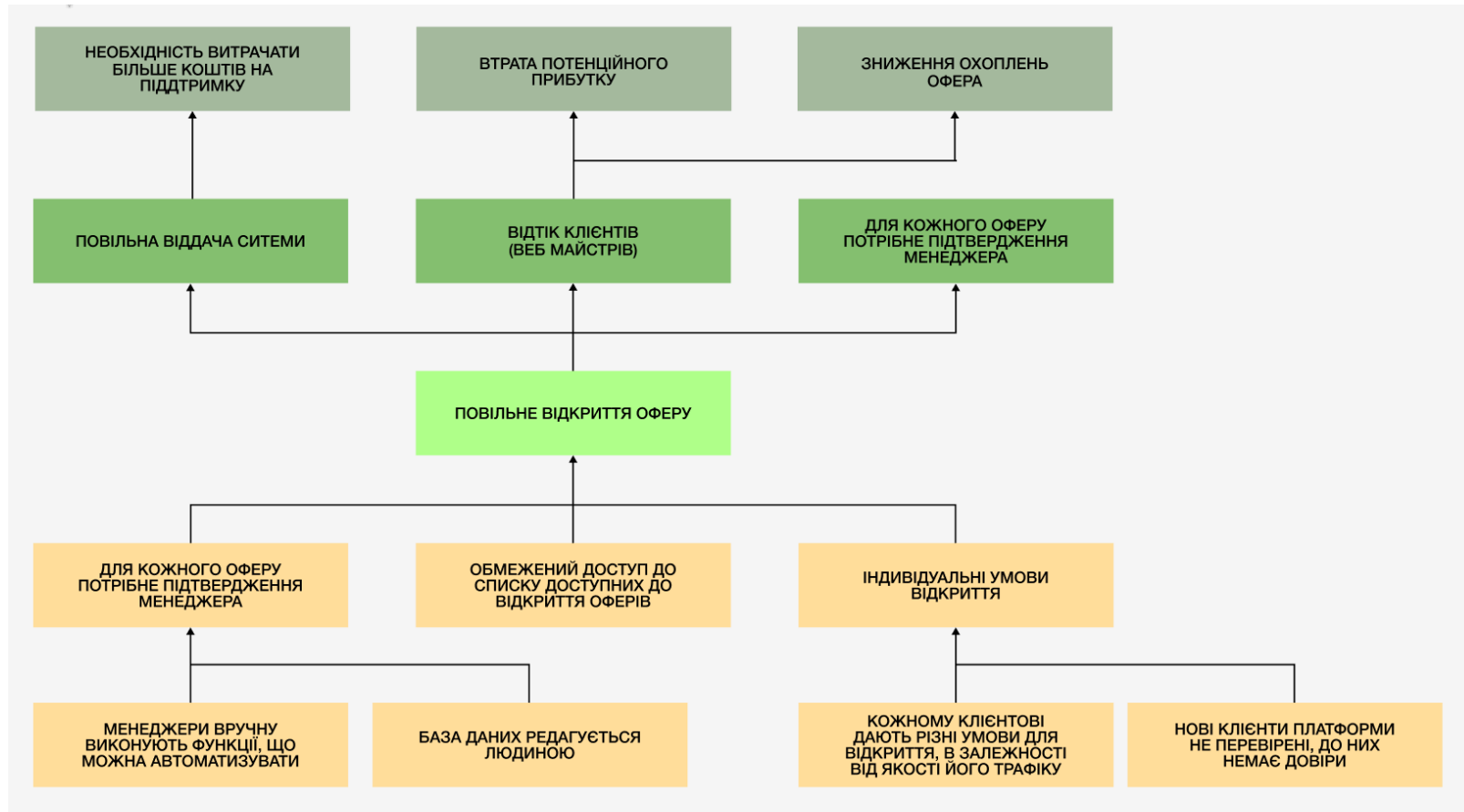


Рис. 2.9. Дерево проблем та рішень

2.1.6 SWOT аналіз

Dod: SWOT аналіз проекту створення веб-додатку для CPA мережі

Таблиця 2.1. SWOT аналіз

Внутрішнє середовище	Сильні сторони (S)	Слабкі сторони (W)
	<ol style="list-style-type: none">1. Оптимізований штат співробітників і відділів2. Молодий колектив (сер вік 25р.)3. Лояльна політика керівництва (соц пакет + умови для праці)4. Ефективне управління ресурсами5. Лояльний до клієнтів сервіс та технічна підтримка	<ol style="list-style-type: none">1. Повільна інтеграція з новими партнерами2. Неавтоматизовані бізнес процеси3. Недостатня кількість співробітників4. Велике операційне навантаження Тім лідів5. Недотримання дедлайнів по виконанню робіт
Зовнішнє середовище	Можливості (O)	Загрози (T)
	<ol style="list-style-type: none">1. Розширення існуючих команд, створення нових (найм нових співробітників)2. Розвиток нових продуктів3. Модернізація та оновлення основного продукту	<ol style="list-style-type: none">1. Конкуренти швидше завершують випуск нових продуктів.2. Збільшення кількості конкурентів3. Вичерпання потенціалу основного продукту4. Банкрутство5. Зниження репутації6. Промисловий шпіонаж

2.2 Математичне забезпечення веб додатку

Для математичного забезпечення використовуються метрики із підключених особистих кабінетів Google та Facebook Ads, тобто дані про переходи за рекламним оголошенням, відслідковування дій користувача напряму пов'язані з API цих сервісів. Для підключення потрібна tracking link з налаштувань оферу (рис. 2.10).

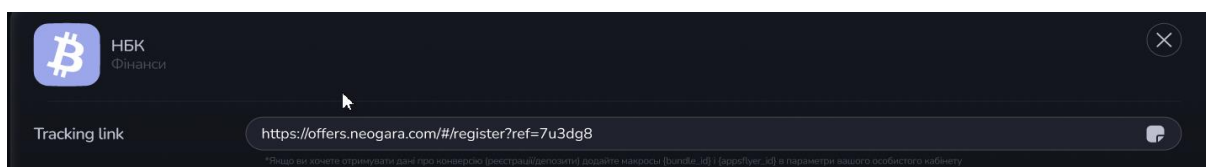


Рис. 2.10. Tracking link оферу

Далі потрібно підключити сам кабінет Google або Facebook, для цього потрібно скопіювати ID кабінету або рекламної кампанії та вставити у відповідне поле (рис. 2.11, 2.12).

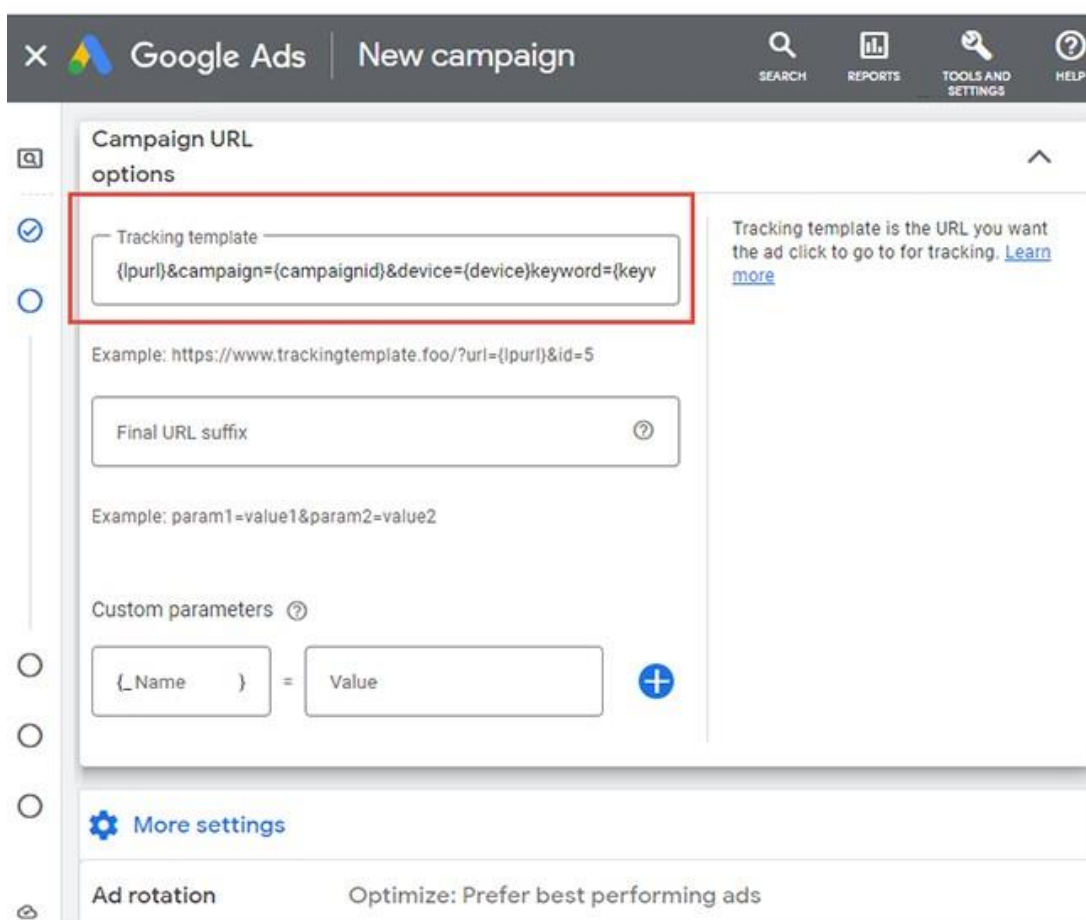


Рис. 2.11 — Google Ads ID

Рис. 2.12. Підключення в застосунку

Після створення офери і підключення його метрик, система застосовує математичні формули розрахунку ефективності роботи рекламної кампанії, а саме CR (коефіцієнт конверсії), а також Visit to reg (коефіцієнт ефективності реєстрацій щодо прибутків).

$$CR = Leads/Hits * 100$$

$$Visit\ to\ reg = Deposits/Leads * 100$$

$$CPA = (marketing\ costs + sales\ costs) / new\ customers$$

$$Profitable\ ROAS = Average\ order\ value / Maximum\ CPA$$

Максимальна CPA (ціна за придбання) — це максимальна сума, яку ви готові витратити, щоб залучити одного клієнта без шкоди для маржі операційного прибутку.

$$Max.\ CPA = operating\ profit\ per\ customer - operating\ profit\ per\ customer\ you\ want\ to\ keep$$

Операційний прибуток на клієнта – це валовий дохід, який ви отримали від клієнта протягом його існування.

Нижче наведено формулу для розрахунку операційного прибутку на клієнта:

$$Операційний\ прибуток\ на\ клієнта = довготривала\ вартість\ клієнта - (середнє\ відшкодування\ на\ клієнта + середні\ прямі\ витрати\ на\ клієнта + середні\ операційні\ витрати\ на\ клієнта)$$

Загальна вартість клієнта (також відома як LTV) – це прогнозований дохід, який ваш клієнт згенерує протягом свого існування.

Прямі витрати включають вартість виробництва/заготівлі товарів + вартість доставки товарів (вартість доставки). У випадку послуг чи інтернет продукту прямі витрати включають вартість надання послуг.

Операційні витрати – це поточні витрати на ведення бізнесу. Операційні витрати можуть включати заробітну плату співробітників, адміністративні витрати, оренду офісу, воду, газ та електроенергію тощо.

Цінність клієнта за весь період (також відома як LTV) – це прогнозований дохід, який отримає клієнт створювати протягом свого існування.

Різні типи клієнтів (наприклад, клієнти з високою ціною та клієнти з низькою цінністю) зазвичай мають різну цінність протягом існування. Тому має сенс розрахувати LTV для кожного унікального сегмента клієнтів.

Розрахунок LTV має сенс лише тоді, коли ваш продукт може забезпечувати зацікавленість і послуги неодноразово (наприклад підписка в мобільному застосунку може оновлюватись раз на місяць).

Нижче наведено формулу для розрахунку LTV:

*Довічна вартість клієнта = середня вартість замовлення * середня частота покупок * середня тривалість життя клієнта*

Середня вартість замовлення (AOV) – це середня вартість транзакції. Звіт Google Analytics про AOV.

Середня частота покупок – це те, як часто в середньому клієнти роблять покупки на вашому веб-сайті. Наприклад, ваші клієнти можуть робити покупки раз на 6 місяців.

Середня тривалість існування клієнта – це середня тривалість (кількість тижнів, місяців або років), протягом якої люди залишаються клієнтами вашого бізнесу.

Маржа операційного прибутку (також відома як операційна маржа) — це відсоток загального обсягу продажів, який залишається після вирахування всіх прямих і операційних витрат із загального обсягу продажів.

Нижче наведено формулу для розрахунку операційного прибутку:

*Маржа операційного прибутку = (операційний прибуток / тривалість клієнта) * 100*

Операційний прибуток - це операційний прибуток компанії.

Загальна формула для розрахунку маржі операційного прибутку компанії така:

$$\text{Маржа операційного прибутку} = (\text{операційний прибуток} / \text{дохід}) * 100$$

Тут замість використання загального обсягу продажів ми використовуємо загальну вартість клієнта за весь період, вимірюється у %.

І саме в операційному прибутку і його маржі в контексті CPA мережі закладені витрати на роботу affiliate менеджерів, які формуються в залежності від об'єму трафіку веб майстрів та його ROI. В спрощені процесу відкриття оферу, яким у всіх існуючих рішеннях оперує affiliate manager полягає покращення взаємодії веб майстра з веб додатком та зменшення витрат компанії, тим самим збільшуючи прибуток. За результатами попередніх досліджень, після автоматизації процесу відкриття оферів, постановки CAP на день, в тому числі за допомогою AI модуля, розробка якого планується в майбутньому, *швидкість зворотного відгуку збільшиться, дохід збільшиться, а витрати ресурсів зменшаться*, що з точки зору проектного менеджменту є оптимізацією ресурсів та по суті науковою новизною.

2.3 Інформаційне забезпечення веб додатку

Так як система оперує великими масивами даних, що стосуються оферів, користувача та налаштувань, потрібно описати і створити базу даних (табл. 2.2.).

Спочатку розробимо концептуальну модель бази даних нотації Чена (рис. 2.13), де будуть відображені сутності системи:

1. Web майстер
2. Affiliate manager
3. Офер
4. Мій офер
5. Конверсія

Кожна з них має свої атрибути, та всі сутності пов'язані між собою. Кожен web майстер має свого менеджера, має доступ до списку оферів, а після відкриття оферу, може переглядати свої офери і статистику, яка в свою чергу і є конверсією.

Таблиця 2.2 Опис зв'язків між сутностями

№	Сутності, що утворюють зв'язок	Тип зв'язку	Пояснення
1	Affiliate manager-Web майстер	Один до багатьох	Один менеджер може бути закріплений за багатьма користувачами, але кожен користувач має тільки одного менеджера.
2	Web майстер – Мої офери	Один до багатьох	Кожен користувач може мати багато оферів, але один офер може бути закріплений тільки за одним користувачем
3	Мої офери – Конверсії	Один до багатьох	Один мій офер має загальну статистику, що вираховується від багатьох конверсій.
4	Мої офери – Офери	Один до одного	Кожен офер спочатку не закріплений ні за ким, після відкриття цей ще офер передає свої параметри і переходить до таблиці моїх оферів.

Наступною є даталогічна модель (рис. 2.14), яка більш детально ілюструє зв'язки між сутностями бази даних із вказанням ключових полів таблиць.

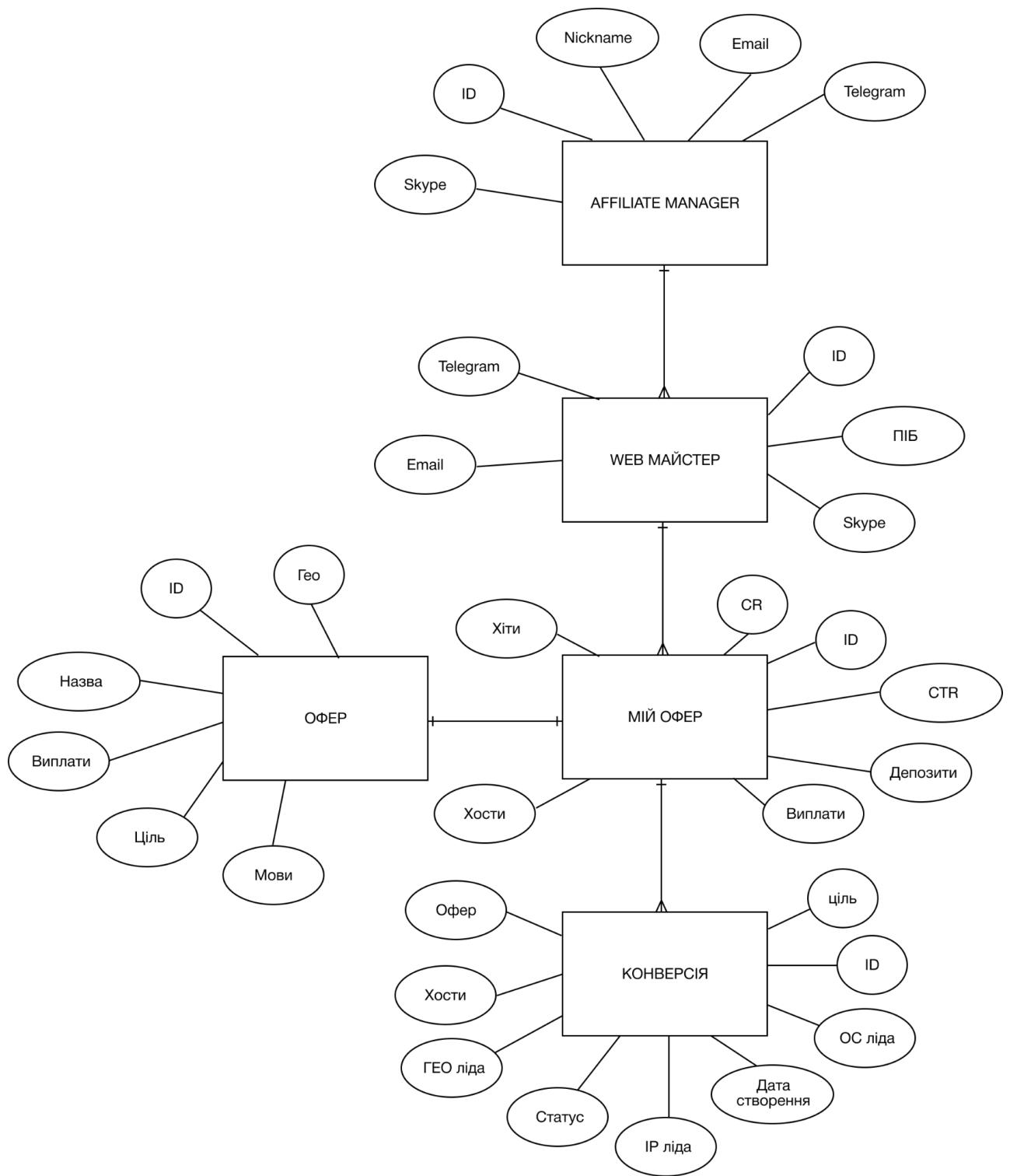


Рис. 2.13. Концептуальна модель бази даних

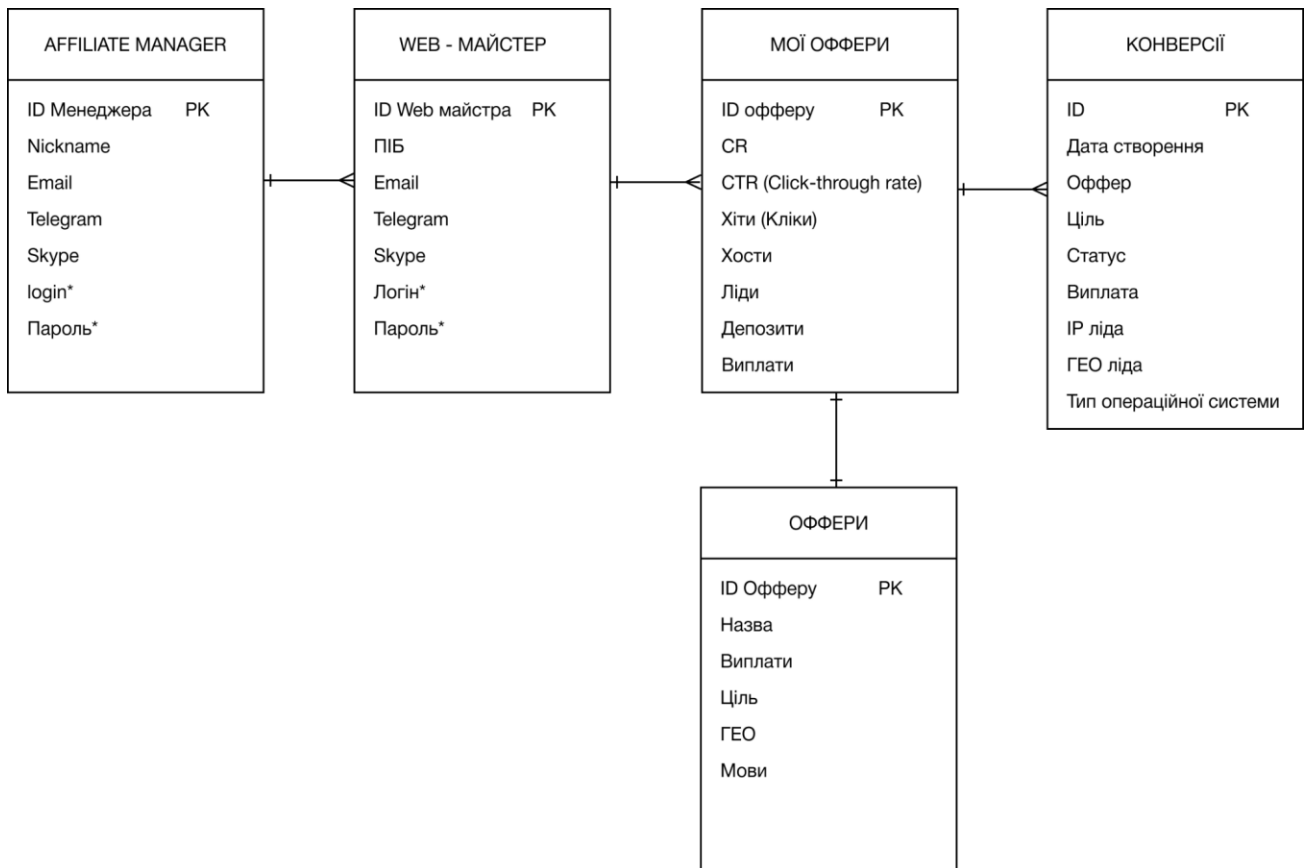


Рис. 2.14. Даталогічна модель бази даних

Також необхідно сформулювати обмеження для атрибутів, при розробці фізичної моделі бази даних, що робиться на основі даталогічної моделі. Обмеження описані у таблиці 2.3.

Таблиця 2.3 Опис атрибутів таблиць

№ п/п	Назва елемента даних	Тип елемента даних	Обов'язкове значення	Обмеження	Ключ
Таблиця Web майстер «веб майстер»					
1	ID	Ціле число	так	4	ПК
2	password	Символьний	так	20	
3	login	Символьний	так	20	
4	Email	Символьний	так	20	
5	Skype	Символьний	ні	20	
6	Telegram	Символьний	ні	20	
Таблиця Affiliate manager «Affiliate менеджер»					
1	ID	Ціле число	так	4	ПК
2	password	Символьний	так	20	
3	login	Символьний	так	20	
4	Email	Символьний	так	20	
5	Skype	Символьний	ні	20	

6	Telegram	Символьний	ні	20	
Таблиця Offers «Офери»					
1	ID	Ціле число	так		ПК
2	name	Символьний	так	20	ЗК
3	cost	Ціле число	так		ЗК
4	goal	Символьний	так	20	ЗК
5	country	Символьний	так		ЗК
6	language	Символьний	так		ЗК
7	info	Символьний	ні	150	ЗК
Таблиця MyOffers «Мої Офери»					
1	ID	Ціле число	так		ПК, ЗК
2	name	Символьний	так	20	ЗК
3	cost	Ціле число	так		ЗК
4	goal	Символьний	так	20	ЗК
5	country	Символьний	так		ЗК
6	language	Символьний	так		ЗК
7	info	Символьний	ні	150	ЗК
1	CR	Числовий	так	3	
2	hits	Ціле число	так	10	
3	leads	Ціле число	так	10	
4	deposits	Ціле число	так	10	
5	host	Ціле число	так	10	
6	proc	Ціле число	так	20	
Таблиця Conversions «Конверсії»					
1	ID	Ціле число	так	150	ПК
2	Date create	Дата	так	9	
3	offer	Ціле число	так	10	ЗК
4	goal	Символьний	так	10	
5	status	Символьний	так	10	
6	money	Ціле число	так	10	
7	lead IP	Символьний	так	20	
8	lead GEO	Символьний	так	50	
9	lead OS	Символьний	так	10	

2.4 Обґрунтування вибору програмних засобів

Для розробки продукту проєкту були обрані наступні інструменти та технології:

- Live Server
- Figma
- Google Analytics

- Facebook Ads
- JavaScript
- CSS
- HTML
- Bootstrap
- Visual Studio Code

Visual Studio Code — це один з найбільш популярних редакторів коду, розроблений корпорацією Microsoft, це швидка, зручне і доволі адаптивне програмне середовище, яке широко використовується в сучасній розробці веб сайтів. Тож я вирішив використувувати саме його для розробки web застосунка.

Figma — Найсучасніший засіб для створення макетів інтерфейсів сайтів, веб або мобільних додатків. Figma виділяється своїм порівняно широким функціоналом, можливістю використовувати плагіни для створення ілюстрацій, іконок, макетів, та в проектуванні складних дизайн систем, прототипуванню інтерфейсів та дослідженню користувацького досвіду. Figma Дозволяє розробити проект одночасно з колегою в реальному часі, без витрати часу на перекидання файлів. Як дизайнер з досвідом роботи, можу рекомендувати використовувати саме її.

Bootstrap — вільний набір інструментів для сайтів і додатків. використовувався для створення динамічної адаптивності сайтів і Web-застосунків.

Live Server — додаткове програмне забезпечення для Visual Studio Code, що надає можливість емулювати сервер на своєму комп'ютері.

Google Analytics — сервіс, що надається Google, для створення детальної статистики відвідувачів веб-сайтів. Статистика збирається на сервері Google, користувач лише розміщує JS-код на сторінках свого сайту. Код відстеження спрацьовує, коли користувач відкриває сторінку у своєму веб-браузері (за умови дозволеного виконання Javascript у браузері).[6]

Facebook Ads — це інструмент Facebook, який дає змогу створювати рекламу у Facebook і керувати нею. Тут ви можете переглядати всі свої кампанії,

набори реклами й окремі оголошення у Facebook, вносити до них зміни й дивитися результати. [7]

JavaScript — мультипарадигменна мова програмування. Підтримує об'єктно-орієнтований, імперативний та функціональний стилі. Є реалізацією специфікації ECMAScript. [8]

CSS — спеціальна мова, що використовується для запису оформлення сторінок, написаних мовами розмітки даних.

HTML — мова розмітки даних.

Проаналізувавши технічне завдання та спланувавши етапи розробки, я почав виконувати завдання.

2.5 Структура програмного забезпечення

Розглянемо структуру розробленого програмного забезпечення web застосунку для SPA мережі (рис. 2.15).

файли розподілені за розширеннями:

→ .html

→ .js

→ .css

Опишемо специфікацію програмних модулів, вихідні і вхідні дані, пояснення щодо функціоналу (табл. 2.4).

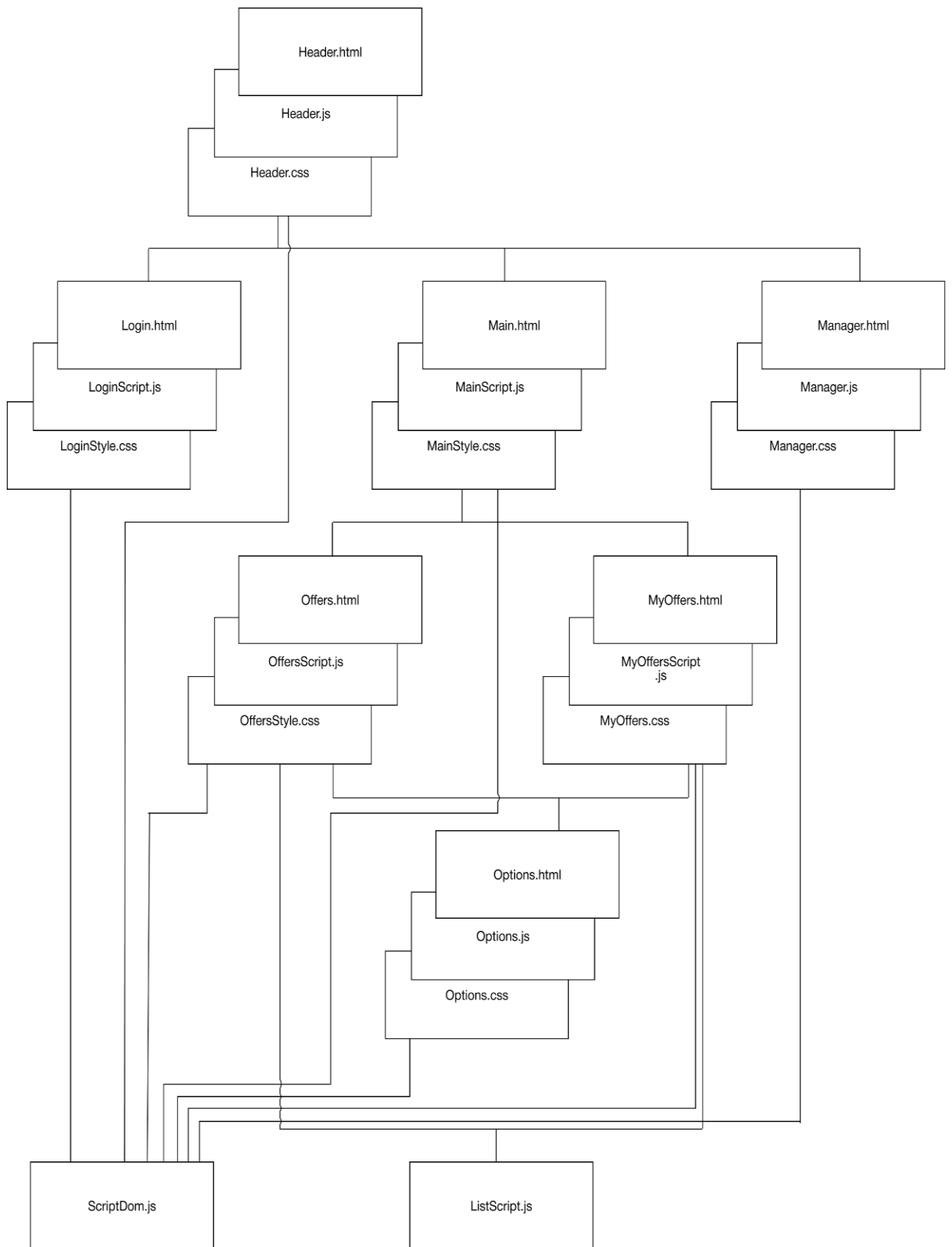


Рисунок 2.15 – Структурна схема програмних модулів

Таблиця 2.4 Специфікація програмних модулів

Модуль	Опис
Header	<p>Меню системи, що потрібне для зручної навігації, завжди супроводжує користувача, тобто модуль переходу та навігації між сторінками системи.</p> <p>Вхідна інформація:</p> <ul style="list-style-type: none"> - Бажана сторінка <p>Вихідна інформація:</p> <ul style="list-style-type: none"> - Інтерфейс відповідної сторінки
Login	<p>Модуль входу до системи, включає реєстрацію та вхід.</p> <p>Вхідна інформація:</p> <p>Дані користувача:</p> <ul style="list-style-type: none"> - при вході: логін та пароль - при реєстрації: ім'я, логін, пароль, email, telegram нікнейм <p>Вихідна інформація:</p> <p>Зареєстрований та залогінений користувач, персональні дані про нього.</p>
Main	<p>Модуль головної сторінки, де відображається збірна статистика, візуалізація даних та особиста інформація користувача.</p> <p>Вхідна інформація:</p> <ul style="list-style-type: none"> - Вид доступу - персональні дані користувача <p>Вихідна інформація:</p> <ul style="list-style-type: none"> - Статистика по рекламним кампаніям щодо відкритих оферів веб майстра - Графічна візуалізація основних показників - Дані про наявність відкритих оферів - Дані профілю та особистого менеджера
Manager	<p>Модуль сторінки з часто запитуваними проблемами, а також контактами особистого менеджера..</p> <p>Вхідна інформація:</p> <ul style="list-style-type: none"> - Вид доступу - дані про менеджера <p>Вихідна інформація:</p> <ul style="list-style-type: none"> - інформація для зв'язку з affiliate менеджером - відповіді на Frequently Asked Questions
Offers	<p>Модуль сторінки оферів доступних для відкриття,</p>

	<p>відображаються всі офери.</p> <p>Вхідна інформація:</p> <ul style="list-style-type: none"> - Вид доступу, таблиця оферів <p>Вихідна інформація:</p> <ul style="list-style-type: none"> - Відображення усіх оферів з БД, з можливою фільтрацією за критерієм - можливість запросити офер на відкриття
MyOffers	<p>Модуль сторінки моїх оферів доступних для відкриття, відображаються всі офери.</p> <p>Вхідна інформація:</p> <ul style="list-style-type: none"> - Вид доступу, таблиця оферів <p>Вихідна інформація:</p> <ul style="list-style-type: none"> - Відображення усіх оферів веб майстра, з можливою фільтрації - Статистика рекламних кампаній по відкритим оферам
Options	<p>Модуль сторінки налаштування відкритого оферу.</p> <p>Вхідна інформація:</p> <ul style="list-style-type: none"> - Вид доступу, вибраний офер <p>Вихідна інформація:</p> <ul style="list-style-type: none"> - Унікальне посилання Tracking link - Відкритий офер
ListScript.js	<p>Модуль сервісу, що відповідає за відображення полів з бази даних у веб застосунку</p> <p>Вхідна інформація:</p> <ul style="list-style-type: none"> - Дані про користувача, тип запиту <p>Вихідна інформація:</p> <ul style="list-style-type: none"> - Результати запитів
ScriptDom.js	<p>Модуль сервісу для побудови архітектури системи у вигляді дерева, що спрощує орієнтацію в коді, оптимізує деякі процеси розробки, такі як заповнення бракуючих елементів, редагування відповідних полів за потреби.</p>

2.6 Висновки до другого розділу

Було проведено функціональний аналіз та розроблено архітектуру системи за допомогою побудови дерева функцій, було змодельовано бізнес-процеси інформаційної системи діаграмами Event-Driven Process Chain та Value Added Chain Diagram, що ілюструють логіку роботи програмного застосунку. Також було

представлено діаграми IDEF0 "ЯК БУДЕ", які показують процеси відкриття оферів для веб майстрів за допомогою інформаційної системи.

На наступному етапі була розроблена архітектура системи, що відображає внутрішні модулі та підсистеми, а також взаємозв'язок між ними.

Було досліджено процес застосування математичних розрахунків і отримання даних від API сервісів Google та Facebook.

Було розроблене інформаційне забезпечення, а саме виявлені та описані сутності інформаційної системи за допомогою концептуальної логічної моделі бази даних, а також детально описані атрибути полів таблиць.

РОЗДІЛ 3. УПРАВЛІННЯ ПРОЄКТОМ СТВОРЕННЯ ВЕБ ДОДАТКУ. ОРГАНІЗАЦІЙНА СТРУКТУРА ПРОЄКТУ, КОНТРОЛЬ ЯКОСТІ.

3.1 Інтелектуальна карта проекту

Інтелектуальні карти можуть використовуватися для планування проєктів, розставляння пріоритетів і відстеження завдань. Вони дозволяють легко ідентифікувати ключові завдання, розподілити ресурси і встановити часові рамки, а також відслідковувати прогрес (рис. 3.1).



Рис. 3.1. Інтелектуальна карта проєкту

3.3 Організаційна структура проекту



Рисунок 3.2 – Організаційна структура

3.4 Планування проектних робіт: визначення складу, тривалості робіт і зв'язків між роботами. Ресурсне планування проекту

Діаграма Ганта є одним із найефективніших інструментів для планування та управління проектами, який візуалізує часову послідовність та тривалість різних завдань або етапів проекту. Застосування діаграми Ганта для проекту створення веб-додатка CRM-системи для CPA мережі дозволить чітко відстежувати прогрес, управляти ресурсами, і координувати роботу команди (Рис. 3.3). Умовно проект можна поділити на такі етапи:

- Розробка концепції CRM-системи.
- Проектування архітектури бази даних.
- Розробка фронтенду і бекенду.
- Інтеграція платіжних систем.
- Тестування та оптимізація.
- Розгортання та налаштування сервера.
- Підготовка документації та матеріалів для користувачів.

PROJECT NAME
 проєкт створення веб додатку CRM системи для CPA мережі

PROJECT MANAGER
 Олександр Коваленко

PROJECT LOCATION
 Kyiv

START DATE
 Monday, January 22, 2024

END DATE
 Friday, March 08, 2024

TASK NAE	TIMELINE		
	START DATE	END DATE	DURATION in days
Аналіз цільової аудиторії продукту та інших схожих CPA мереж	11/06/23	11/11/23	6
Розрахунок бюджету	11/13/23	11/17/23	5
Прогнозування прибутку на перший місяць після запуску	11/13/23	11/17/23	5
Зведення і затвердження ТEO проєкту веб додатку	11/20/23	11/21/23	2
Проведення переговорів з інвесторами	11/22/23	11/22/23	1
Формування функціональних обов'язків команди	11/23/23	11/27/23	5
Аналіз архітектури кроссплатформених додатків	11/24/23	11/30/23	7
Вибір архітектури додатку	11/28/23	11/30/23	3
Вибір ОС для розробки	11/29/23	11/30/23	2
Вибір мови програмування (можливі фреймворки), сервіси розробки	12/01/23	12/02/23	2
Визначення функціоналу додатку	12/01/23	12/02/23	2
Написання ТЗ	12/06/23	12/11/23	6
Визначення загальних положень	12/12/23	12/12/23	1
Визначення вимог	12/13/23	12/14/23	2
Затвердження критеріїв	12/15/23	12/16/23	2
Оцінка ризиків	12/18/23	12/19/23	2
Складання фінансового плану	12/19/23	12/22/23	4
Затвердження dealайнів	12/25/23	12/25/23	1
Розробка макету додатку, візуальної концепції	01/08/24	01/15/24	8
Верстання фронт-енд частини проєкту	01/12/24	01/24/24	13
Розробка Бекенд частини	01/08/24	01/25/24	18
Інтеграція бекенд частини	01/22/24	01/26/24	5
Тестування додатку	01/15/24	01/31/24	17
Тестування роботи СУБД	01/23/24	01/31/24	9
Виправлення багів	01/15/24	02/02/24	19
Повторне тестування	02/05/24	02/07/24	3
Опис помилок	02/05/24	02/07/24	3
Приймання робіт	02/08/24	02/09/24	2
Формування звіту витрат та виконаних ТЗ на проєктування та розробку	02/12/24	02/14/24	3
Підготовка документації додатку	02/12/24	02/16/24	5
Підготовка документації для нових розробників	02/14/24	02/21/24	8
Підготовка фінансового звіту	02/19/24	02/23/24	5
Підготовка інструкції користувача	02/26/24	02/29/24	4
			1
			1
			1

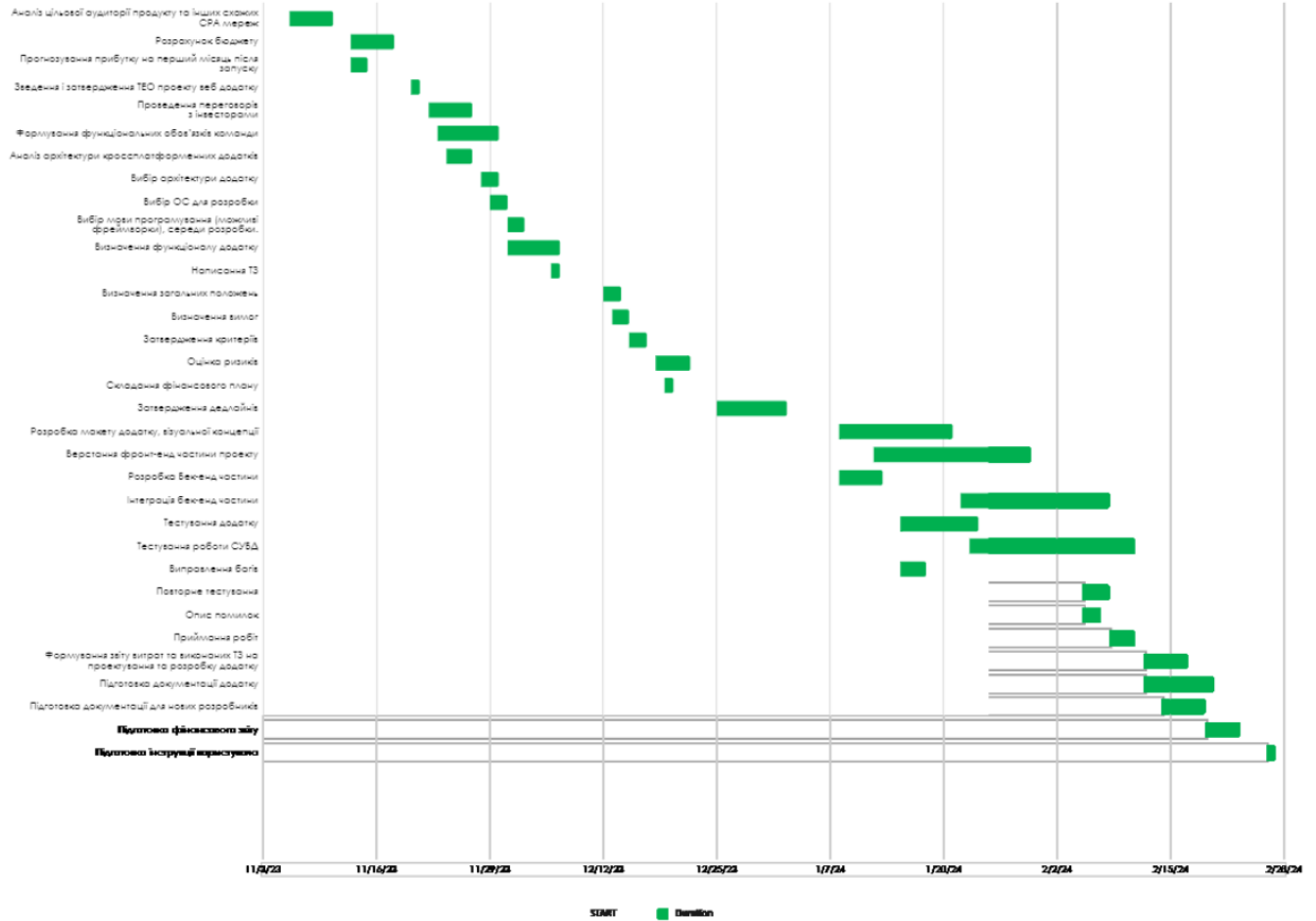


Рис. 3.3. Діаграма Ганта

3.5 Декомпозиція проекту на складові

Основним продуктом проекту є веб-додаток для CPA мережі. Основою його роботи є база даних, API запити на збір статистичних даних з їх подальшою обробкою математичними модулями додатку.

№	Елемент конфігурації	Характеристика
1	Сайт	Лендінг-пейдж та особистий кабінет
2	Кроссплатформенний додаток	Створений додаток на платформах Windows, MacOS, Android, IOS
3	Брендинг та фірмовий стиль	Власний дизайн, логотип та айдентика
4	Сервіси платформи	Система трекінгу маркетингових кампаній та графічне зображення аналізу статистики в реальному часі
5	База даних	СУБД для оферів
6	Маркетингова компанія	Використовуючи TOV (Tone Of Voice), досягнуто ~1 000 000 осіб цільового сегменту
7	Офіс	Офіс («Крипто-хаус») 350 кв.м. на 20-30 осіб
8	Юридична особа	ПАТ

Рис. 3.4. Конфігурація продукту проекту

Основою його роботи є база даних, та API запити, що є другим продуктом проекту. Планується розробити Мобільну версію для роботи з телефону, це буде третім продуктом. В процесі роботи над проектом, було розроблено WBS по фазам проекту (рис. 3.5-3.6).

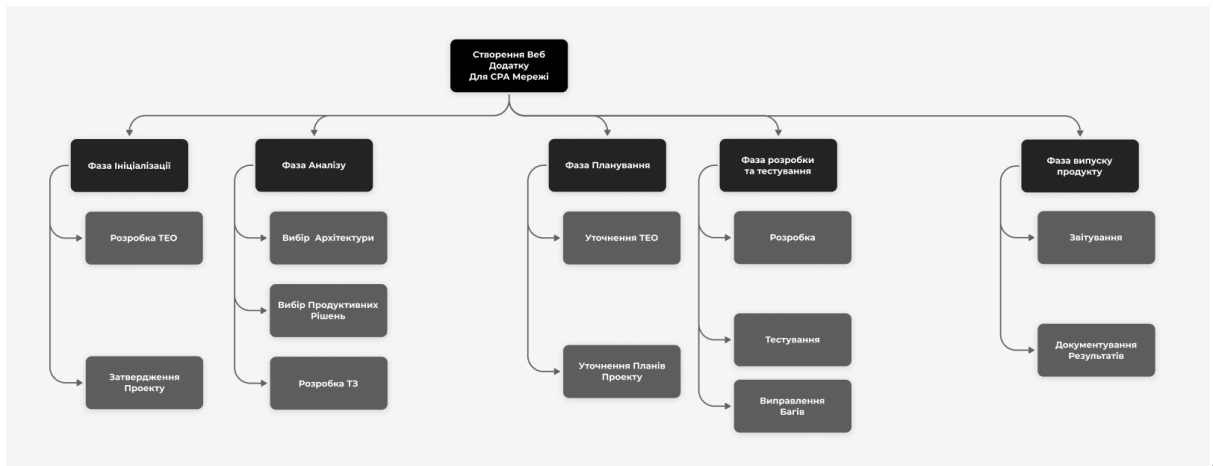


Рис. 3.5. WBS життєвого циклу проекту

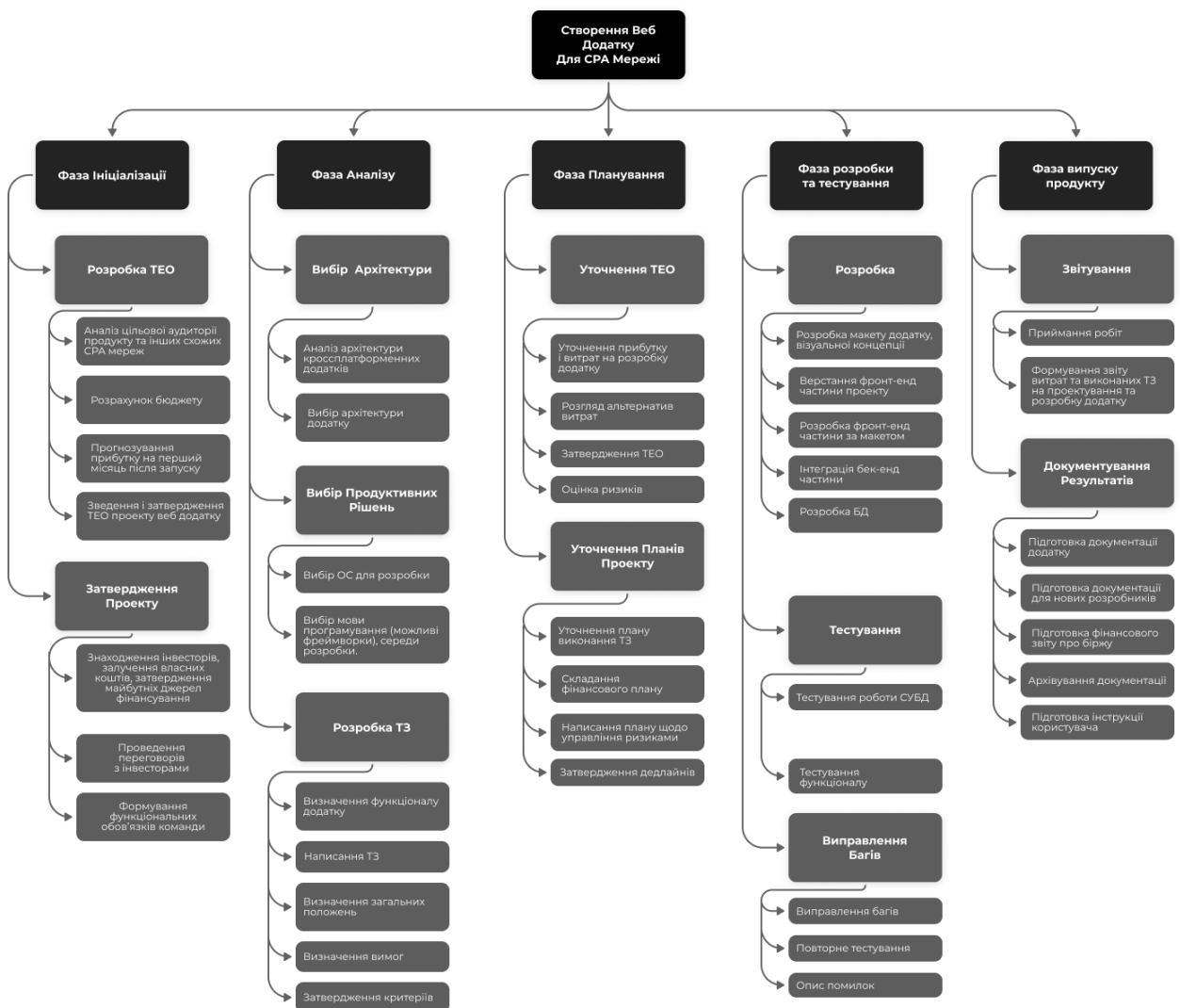


Рис. 3.6. Декомпозиція WBS життєвого циклу проекту

3.6 Календарний план відповідно до завдань

Календарний план виконання робіт — це деталізований графік, який показує, коли кожне завдання або етап проекту має бути розпочатий та завершений:

Таблиця 3.1 Специфікація програмних модулів

Назва задачі	Дата початку	Дедлайн	Термін виконання
Аналіз цільової аудиторії продукту та інших схожих СРА мереж	11/06/23	11/11/23	6
Розрахунок бюджету	11/13/23	11/17/23	5
Прогнозування прибутку на перший місяць після запуску	11/13/23	11/17/23	5
Зведення і затвердження ТЕО проекту веб додатку	11/20/23	11/21/23	2
Проведення переговорів з інвесторами	11/22/23	11/22/23	1
Формування функціональних обов'язків команди	11/23/23	11/27/23	5
Аналіз архітектури кроссплатформених додатків	11/24/23	11/30/23	7
Вибір архітектури додатку	11/28/23	11/30/23	3
Вибір ОС для розробки	11/29/23	11/30/23	2
Вибір мови програмування (можливі фреймворки), середі розробки.	12/01/23	12/02/23	2
Визначення функціоналу додатку	12/01/23	12/02/23	2
Написання ТЗ	12/06/23	12/11/23	6
Визначення загальних положень	12/12/23	12/12/23	1
Визначення вимог	12/13/23	12/14/23	2
Затвердження критеріїв	12/15/23	12/16/23	2
Оцінка ризиків	12/18/23	12/19/23	2
Складання фінансового плану	12/19/23	12/22/23	4
Затвердження дедлайнів	12/25/23	12/25/23	1
Розробка макету додатку, візуальної концепції	01/08/24	01/15/24	8
Верстання фронт-енд частини проекту	01/12/24	01/24/24	13
Розробка Бек-енд частини	01/08/24	01/25/24	18

Інтеграція бек-енд частини	01/22/24	01/26/24	5
Тестування додатку	01/15/24	01/31/24	17
Тестування роботи СУБД	01/23/24	01/31/24	9
Виправлення багів	01/15/24	02/02/24	19
Повторне тестування	02/05/24	02/07/24	3
Опис помилок	02/05/24	02/07/24	3
Приймання робіт	02/08/24	02/09/24	2
Формування звіту витрат та виконаних ТЗ на проектування та розробку додатку	02/12/24	02/14/24	3
Підготовка документації додатку	02/12/24	02/16/24	5
Підготовка документації для нових розробників	02/14/24	02/21/24	8
Підготовка фінансового звіту	02/19/24	02/23/24	5
Підготовка інструкції користувача	02/26/24	02/29/24	4

В процесі роботи над проектом виникли затримки на двох етапах, Задача з аналізу архітектури СРА мереж та кроссплатформенних додатків затягнувся майже на тиждень, а також розробка UX/UI дизайну зайняла більше днів ніж очікувалось, але так як в управлінні проектом використовувався фреймворк KanBan, на кожному з етапів більшість задач мала виконуватись паралельно, через що затримка не вплинула на кінцеві терміни завершення проекту. Жоден з критичних шляхів не був затриманий, отже можна зробити висновок, що гнучкість обраного фреймворка KanBan методології Agile є ефективною при розробці великих проєктів (рис. 3.8).

Було розроблено календарний план проєкту, діаграму Ганта; після зміни тривалості робіт тривалість проєкту не змінилась. Задачі, що враховувались в діаграмах Ганта також були декомпозовані на менші задачі до виконання, а контроль виконання робіт здійснювався за допомогою трекера задач, члени команди самостійно відмічали час, який вони витратили на роботу в застосунку Jira (рис. 3.7)

	03 SAT	04 SUN	05 MON	06 TUE	07 WED	08 THU	09 FRI	10 SAT	11 SUN	12 MON	13 TUE	14 WED	15 THU	16 FRI	Total
Anastasia P							2h			2.25h	4.58h	6.25h	6.5h		31,58h
Milena Zenkevych			3h	2.25h	4.5h	3.5h	4.67h			3.25h	4.17h	3.33h	3.75h		32,42h
Oleksandr Kovalenko			2.75h	3.5h	3.17h	3.5h	5h			4.83h	3.83h	3.67h	3.5h		33,75h
Sofia Shepshuk			3.33h	2.5h	3.75h	3.25h	3.5h			4.33h	4.67h	3.83h	2.6h		31,77h
Tetiana Slipchenko			2.72h	3.92h	3.47h	4.68h	5.02h			3.93h	5.57h	5.72h	5.1h		40,12h
Veronika Marchenko			5.42h	3.42h	4.75h	4.33h	4.17h			4.42h	3.83h	4.25h			34,58h
Total (65 Issues)			20.72h	17.42h	21.8h	21.77h	24.35h			23.02h	26.65h	27.05h	21.45h		204.22h

Рис. 3.7. Трекінг задач співробітниками команди

PROJECT NAME
проект створення веб додатку CRM системи для CPA мережі

PROJECT MANAGER
Олександр Коваленко

PROJECT LOCATION
Kyiv

START DATE
Monday, January 22, 2024

END DATE
Friday, March 08, 2024

TASK NAME	TIMELINE		
	START DATE	END DATE	DURATION in days
Аналіз цільової аудиторії продукту та інших схожих CPA мереж	11/06/23	11/11/23	6
Розрахунок бюджету	11/13/23	11/17/23	5
Прогнозування прибутку на перший місяць після запуску	11/13/23	11/17/23	5
Зведення і затвердження ТСО проекту веб додатку	11/20/23	11/21/23	2
Проведення переговорів з інвесторами	11/22/23	11/22/23	1
Формування функціональних обов'язків команди	11/23/23	11/27/23	5
Аналіз архітектури кроссплатформених додатків	11/27/23	12/09/23	13
Вибір архітектури додатку	11/28/23	11/30/23	3
Вибір ОС для розробки	12/01/23	12/04/23	4
Вибір мови програмування (мажориті фреймворки), середі розробки.	12/01/23	12/02/23	2
Визначення функціоналу додатку	12/01/23	12/02/23	2
Написання ТЗ	12/08/23	12/13/23	6
Визначення загальних положень	12/13/23	12/14/23	2
Визначення вимог	12/13/23	12/14/23	2
Затвердження критеріїв	12/15/23	12/16/23	2
Оцінка ризиків	12/18/23	12/19/23	2
Складання фінансового плану	12/19/23	12/22/23	4
Затвердження дедлайнів	12/25/23	12/25/23	1
Розробка макету додатку, візуальної концепції	01/08/24	01/18/24	11
Верстаття фронт-енд частини проекту	01/16/24	01/26/24	11
Розробка Бек-енд частини	01/08/24	01/25/24	18
Інтеграція бек-енд частини	01/29/24	02/01/24	4
Тестування додатку	01/15/24	02/02/24	19
Тестування роботи СУБД	01/23/24	02/03/24	12
Виправлення багів	01/15/24	02/05/24	22
Повторне тестування	02/05/24	02/07/24	3
Опис помилок	02/05/24	02/08/24	4
Приймання робіт	02/08/24	02/09/24	2
Формування звіту витрат та виконаних ТЗ на проєктування та розробку	02/12/24	02/14/24	3
Підготовка документації додатку	02/12/24	02/16/24	5
Підготовка документації для нових розробників	02/14/24	02/21/24	8
Підготовка фінансового звіту	02/19/24	02/23/24	5
Підготовка інструкції користувача	02/26/24	02/29/24	4
			1
			1
			1

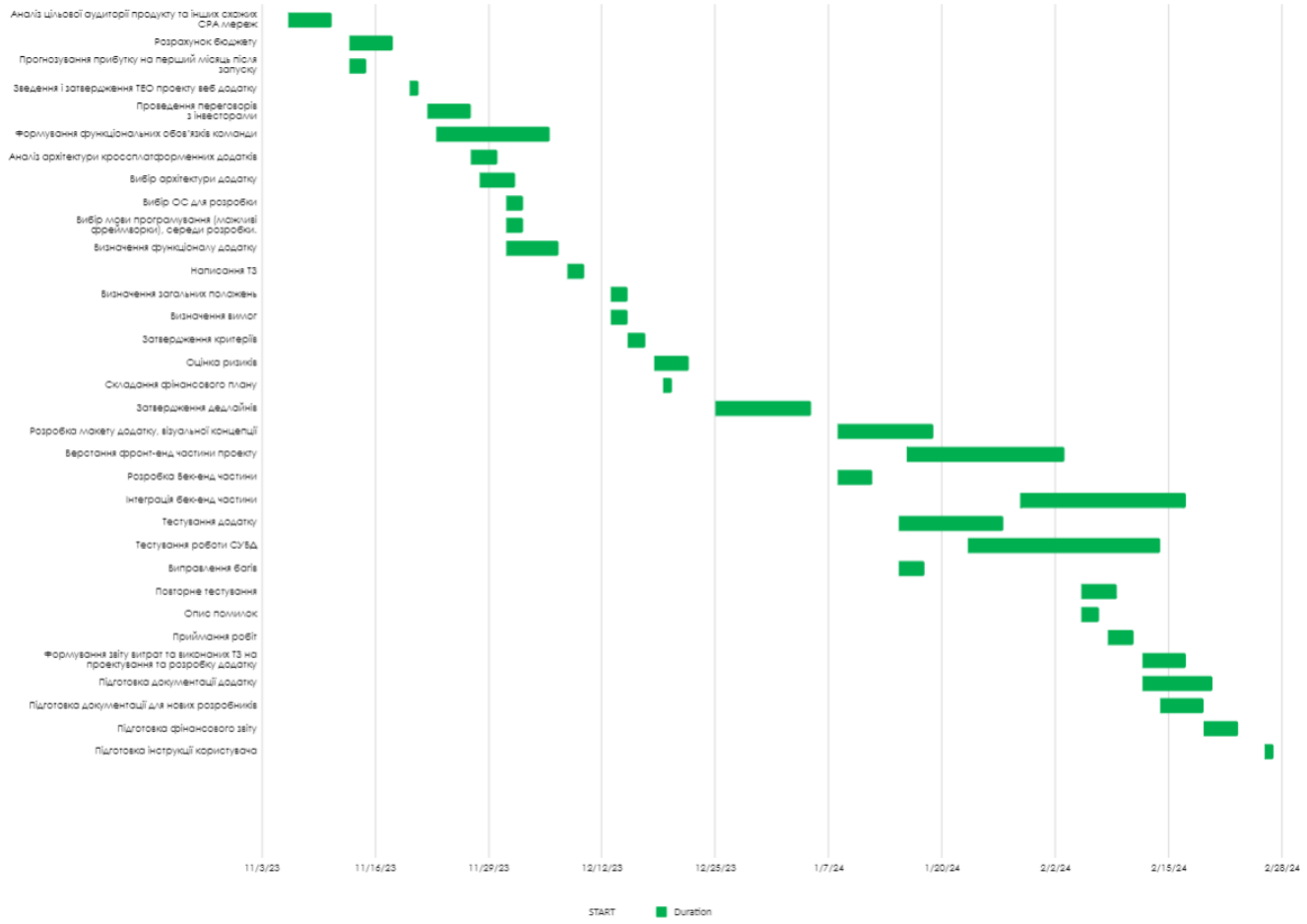


Рис. 3.8. Моніторинг діаграма Ганта, фактичне виконання робіт після затримки

3.7 Функціонування системи, вплив зовнішніх зв'язків. Формування складу команди

3.7.1 Функціонування системи, вплив зовнішніх зв'язків

Система - веб додаток CRM системи для CPA мережі.

Аналіз системи з фізичної точки зору:

Надсистема: Платіжні системи, платформи для розміщення реклами (джерела трафіку).

Підсистеми: Веб-сторінка/сторінка в соціальних мережах, СУБД, CRM система, офіс, співробітники, сервери, комп'ютерне обладнання, юрист, бухгалтерія, співробітники.

Визначимо зовнішні зв'язки системи з надсистемою:

Зовнішні фактори впливу:

- Користувачі: Запускають рекламні кампанії, створюють запити на відкриття оферів, здійснюють взаємодію всередині спільноти.
- Інші ПП або CPA мережі: конкуренція.
- Постачальники обладнання: постачають необхідне обладнання.
- Постачальники хмарних послуг: використання зовнішніх обчислювальних ресурсів.
- Закони про комерційну діяльність: підпорядковує сплатення податків до держбюджету через ФОП.
- Юридична фірма: проводять юридичні консультації.

Складемо повний перелік всіх підсистем.

- Веб-сторінка
- Сторінка в соц. мережах
- Обладнання
- Співробітники
- Офіс
- Сервери
- Бухгалтерія

- Юрист
- Оффер
- CRM система

Складемо повний перелік елементів для кожної підсистеми.

Веб-сторінка:

- лендінг;
- форма реєстрації
- політика конфіденційності
- політика обробки персональних даних
- дизайн елементи

Сторінка в соц. мережах:

- Instagram сторінка
- Facebook
- Linkedin
- Dou
- Twitter
- Youtube

Обладнання:

- Сервери
- Комп'ютери (блоки, монітори, кабелі, роутери)
- телефони
- локальна мережа

Співробітники: СЕО

- тех відділ
- маркетинг відділ
- відділ дизайну
- відділ тестування
- Affiliate менеджери
- відділ фінансів

- юридичний відділ
- HR відділ
- RnD відділ
- Web-розробники

Офіс:

- мебель
- побутова техніка (холодильник, мікрохвильова піч, кофеварка, кондиціонери, світильники)

- зона відпочинку (PS5, пінг-понг, пуфики)
- Серверна
- їдальня
- переговорна кімната
- кабінети
- туалет
- система охорони (вхід по ключ картам, камери, лазерна сигналізація)
- продукти харчування (обіди, фрукти, печеньки, чаї, кава, пиво, інші

напої)

- система пожежогасіння

Сервери:

- окрім власних
- хмарні сервери

Бухгалтерія:

- система обліку документів фінансових потоків

Юрист:

- людина з профільною освітою.

Токен:

- блокчейн.

CRM система:

- особистий кабінет історія транзакцій
- NFT у власності

- інструменти для трейдингу
- інструменти для створення своїх NFT
- Інструменти для редагування особистої інформації.

Подамо у вигляді системи процес функціонування системи (рис. 3.9):

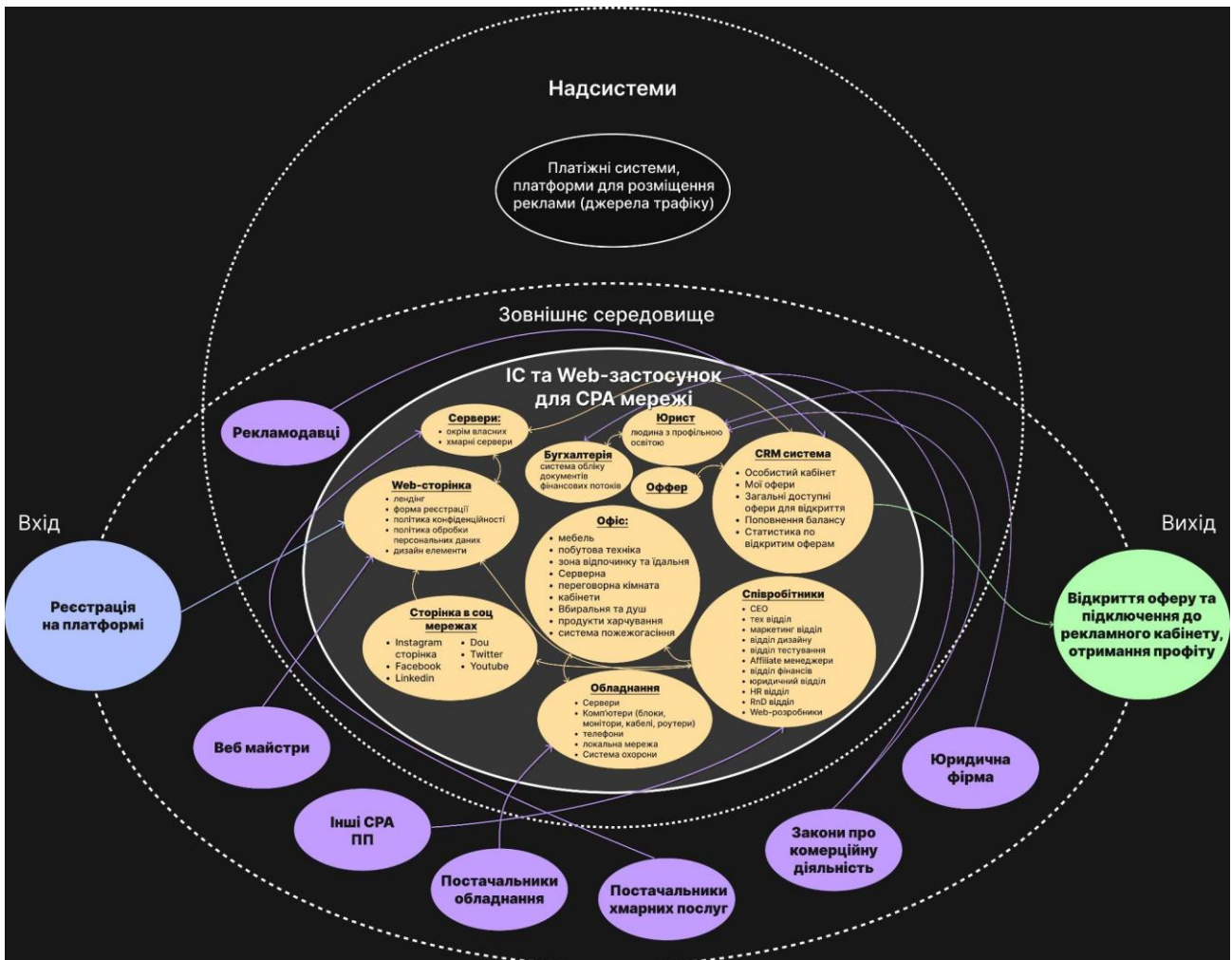


Рис. 3.9. Внутрішнє, зовнішнє середовище функціонування системи Відкриття офферу.

→ Клієнт оформляє запит через веб-додаток для поповнення поточного рахунку.

→ Клієнт заповнює необхідні дані для підтвердження операції для поповнення особистого рахунку.

→ Зовнішня частина веб-сторінки передає заповнену інформацію на обробку серверу.

→ Сервер обробляє повну інформацію від користувача, обробляє транзакцію та заповнює БД користувача(історія транзакцій, особистий рахунок або повідомлення про помилку транзакції) та передає запит зовнішній частині веб додатку.

→ Зовнішня частина відображає інформацію про транзакцію особистого рахунку.

Результатом користування платформою було затверджене системою, відкриття офери та підключення рекламних метрик в кабінет, в результаті чого веб майстер зможе отримувати прибуток за зроблений результат.

3.7.2 Формування команди

За результатами проведених досліджень були сформовані вимоги до кандидатів в команду.

Таблиця 3.2 Вимоги до членів команди

Проектний менеджер	<ul style="list-style-type: none"> → Досвід управління проектами від 2х років в ІТ проектах. → Здатність чітко та ефективно спілкуватися як усно, так і письмово. → Здатність мотивувати та керувати командами, вирішувати конфлікти, інспірувати довіру та співпрацю. → Організаційні навички: Ефективне планування, організація ресурсів, управління часом та дотримання термінів. → Аналітичні навички: Здатність аналізувати складні дані, визначати ризики та виробляти стратегічні рішення. → Адаптивність та гнучкість: Готовність швидко адаптуватися до змін та управляти кількома проектами одночасно. → Знання інструментів управління проектами: Володіння програмним забезпеченням для управління проектами, таким як Microsoft Project, JIRA, Asana, Trello або інші аналогічні інструменти. → Знання технологій розробки → Стресостійкість: Здатність зберігати виваженість і ефективність під час роботи під тиском.
Маркетолог	<ul style="list-style-type: none"> → Досвід роботи маркетологом від 2х років → Досвід роботи і володіння інструментом управління контекстною рекламою Google Adwords → Досвід роботи в просуванні нового товару → Мінімальний досвід в керівництві. → Досвід роботи і володіння інструментами веб-аналітики: Google Analytics
Дизайнер	<ul style="list-style-type: none"> → Досвід роботи понад 2 роки на посаді UX/UI Designer;

	<ul style="list-style-type: none"> → Досвідчений користувач Figma, Photoshop, Illustrator, Invision; → Розуміння принципів usability та UX методологій; → Розуміння принципів графічного дизайну, типографіки, кольору; → Досвід дизайну веб-сайтів та мобільних додатків; → Досвід створення responsive/adaptive design; → Розуміння суті кількісних та якісних досліджень; → Знання гайдлайнів Google Material Design; → Знання гайдлайнів Apple Human Interface Guidelines; → Аналітичне та креативне мислення; → Досвід участі в імплементації дизайну у frontend та backend розробку; → Уміння приймати самостійні дизайн-рішення та обґрунтовувати їх; → Розуміння можливостей та обмежень HTML, CSS, JavaScript;
Back-end розробник	<ul style="list-style-type: none"> → Досвід роботи від 3х років → Мови програмування: Java/Python/JavaScript (Node.js). → Бази даних: MySQL/PostgreSQL/MongoDB/Oracle/SQL Server. → Інструменти розробки API: REST/GraphQL. → Фреймворки: Express.js/Django/Flask/Ruby on Rails/Spring Boot. → Інструменти для управління версіями: Git. → Аналітичний склад розуму. → Увага до деталей. → Здатність до вирішення проблем. → Ефективна комунікація з членами команди.
WEB - розробник	<ul style="list-style-type: none"> → Досвід роботи від 3х років → Гарні знання JavaScript/TypeScript, React+Redux, SCSS\SASS, Jest, Node.js. → Рівень знання англійської мови, достатній для читання і створення внутрішньої технічної документації → Знання Agile/Scrum, Kanbas; → Досвід роботи у розробці нетривіальних рішень → Відмінні знання SQL, бажано PostgreSQL → Рівень знання англійської мови, достатній для читання і створення внутрішньої технічної документації
Тестувальник	<ul style="list-style-type: none"> → Досвід роботи від 2х років. → Написання тест-кейсів і складання чек-листів; → Виявлення, документація й верифікація помилок, а також ведення документації по тестам; → Планування процесу тестування й тестування функціоналу;

	<p>оптимізація процесів розробки, щоб уникнути повторної появи виявлених помилок;</p> <p>→ Повторний тест виправлених помилок, аналіз тестування і його оптимізація, моніторинг та відстеження поправок;</p> <p>→ Деталізація/збір вимог до ПЗ.</p>
Юрист	<p>→ знання цивільного, господарського, договірної, податкового та іншого законодавства, що регулює діяльність онлайн організацій та інших прибуткових суб'єктів;</p> <p>→ вміння складати проекти документів, в т.ч. процесуальні, проекти договорів, додатків та додаткових угод до них, проекти корпоративних та внутрішніх документів організації, тощо;</p> <p>→ навички роботи з нормативно-правовою базою, судовою практикою, реєстрами та іншими базами даних, необхідними для виконання посадових обов'язків;</p> <p>→ навички ведення документообігу, баз даних, реєстрів, тощо;</p> <p>→ гарні комунікаційні навички;</p>
HR-менеджер	<p>→ Підбір персоналу;</p> <p>→ Проведення первинного інтерв'ю (онлайн);</p> <p>→ Ведення бази кандидатів;</p> <p>→ Адаптація нових колег і комунікація (он-лайн);</p> <p>→ Збір та обробка документів;</p> <p>→ Вирішення конфліктних ситуацій.</p> <p>→ Досвід роботи від 2х років.</p>
Фінансист	<p>→ Вища економічна освіта.</p> <p>→ Досвід роботи на аналогічній посаді від 2 року.</p> <p>→ Навики аналізу, планування та прогнозування.</p> <p>→ Навики обробки, систематизації та інтерпретації інформації.</p> <p>→ Аналітичне та критичне мислення, високий рівень уважності і концентрації на деталях, системний підхід в роботі.</p> <p>→ Знання розподілу постійних і змінних витрат.</p> <p>→ Знання в області управлінського обліку і звітності, базові знання бухгалтерського обліку.</p> <p>→ Впевнений користувач програм(Excel, 1C, Power Point, Power BI).</p>

Також були сформовані вимоги щодо організації проекту відносно досягнення його цілі

Таблиця 3.3 Вимоги до організації проекту

Сформовано команду проекту	Всі члени команди мають необхідну кваліфікацію та досвід роботи
----------------------------	---

	в проєктах для виконання своїх посадових задач. Створенні необхідні відділи для виконання завдань для створення продукту. Налаштована система для керівників відділів для передачі завдань всім працівникам. Всі працівники офіційно влаштовані в компанію та укладений договір праці в компанії. Підписано документ про нерозголошення конфіденційної інформації
Команду забезпечено всіма необхідними умовами для ефективної роботи	У випадках роботи в офісі (якщо дозволяє ситуація в країні) у приміщеннях, де працюють всі члени команди, дотримуються правил для забезпечення особистої безпеки під час повітряної тривоги. Будівля та приміщення повинні мати укриття для безпеки кожного працівника компанії. Також приміщення повинно відповідати вимогам нормативно-технічної та експлуатаційної документації, чинних санітарних норм, відповідати умовам безпеки при пожежі, та мати більше чим необхідно робочих місць та комп'ютерів. В разі віддаленої роботи керівництво повинно отримати підтвердження від співробітників, що в них вдома є всі необхідні умови для роботи. При відсутності комп'ютера у працівника компанії - виділити ноутбук.
Виділено фінанси для роботи проєктної команди та менеджерів.	Заробітна плата для проєктної команди та керівників проєкту передбачена на весь час роботи над проєктом. Виплачується вчасно та у розмірі, що вказувався в контрактах. Плата відповідає актуальній ринковій вартості відповідних послуг.
Виділено фінанси для закупівлі необхідної техніки	Ліцензія для програмних продуктів повинна бути закуплена вчасно та на весь термін роботи над проєктом. Закуплено всі необхідні сервіси для хмарного збереження інформації та користування сервісами серверів повинні бути сплачені на весь час роботи над проєктом.

3.8 Керівництво користувача

Для того щоб почати користуватись системою web майстер має пройти процес реєстрації або ж аутентифікації за допомогою логіну та паролю, якщо він вже має створений акаунт.

Під час реєстрації (рис. 3.10, 3.11, 3.12) він має заповнити всі необхідні поля, пройти перевірку на бота за допомогою ReCaptcha та натиснути кнопку “зареєструватись”, після чого він зможе перейти на сторінку входу, де є обов’язкові для заповнення поля логіну та паролю. Після успішного заповнення та натиснення кнопки “увійти” система перевірить наявність акаунту в базі даних і користувач потрапить на головну сторінку системи.

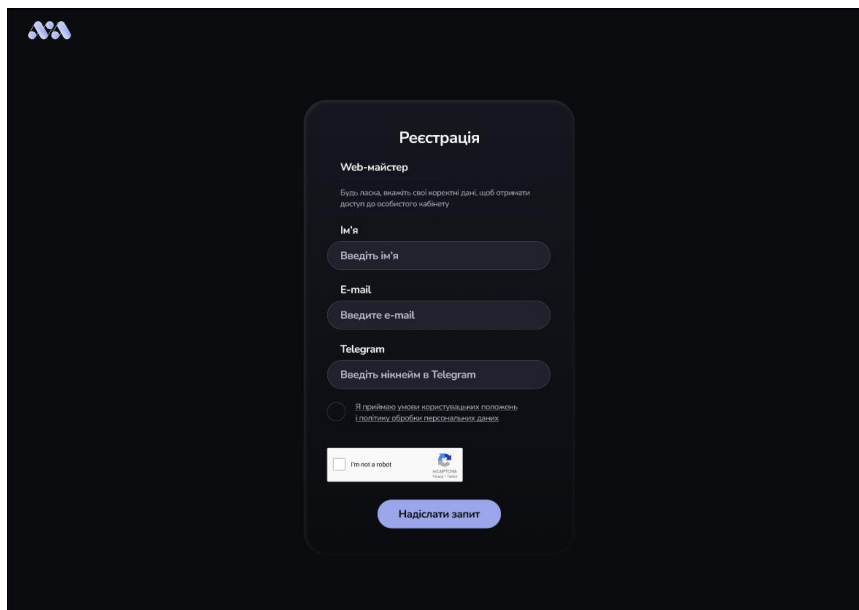


Рис. 3.10. Сторінка реєстрації

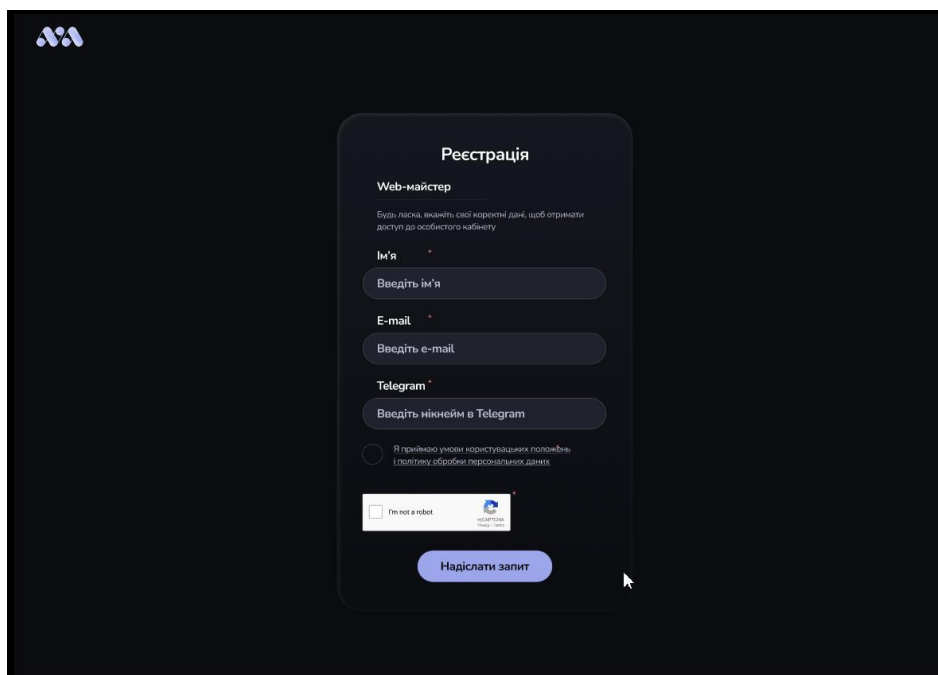


Рис. 3.11. Повідомлення системи про обов'язкове заповнення полів

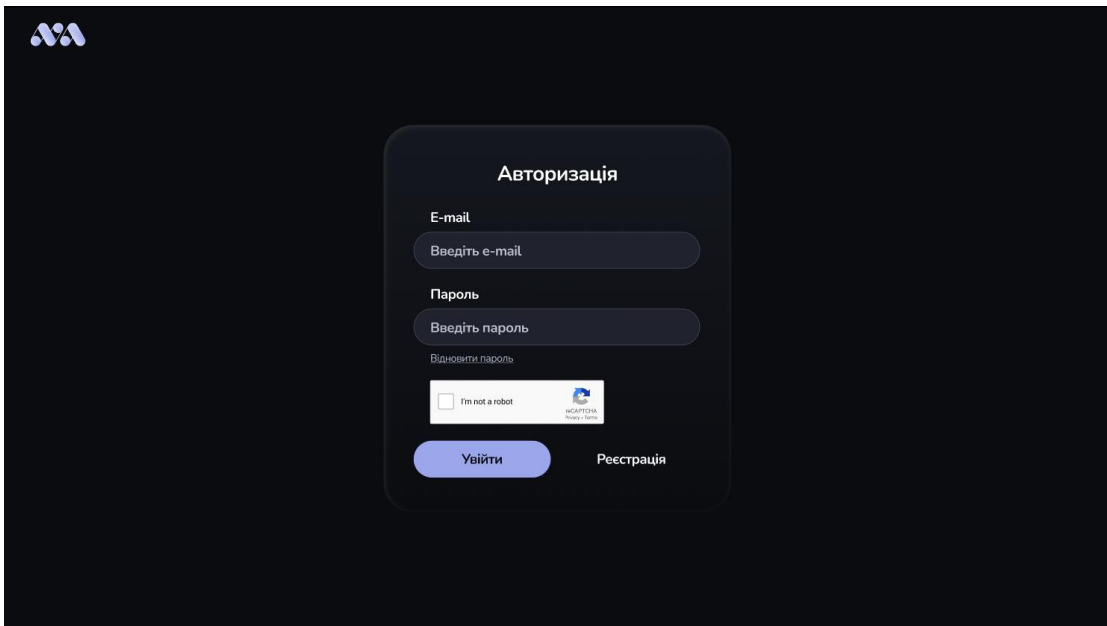


Рис. 3.12. Сторінка авторизації

На головній сторінці веб майстер має можливість відслідкувати актуальну статистику по його відкритим оферам, переглянути динаміку через візуальне представлення математичних формул і метрик рекламних кампаній (рис. 3.13).



Рис. 3.13. Головна сторінка

При наведенні на зони графіку, можна переглянути статистику за вибраний день (рис.3.14).



Рис. 3.14. Відображення статистики за вибрану дату

У правій частині екрану (рис. 3.15) веб майстер має можливість переглянути короткий список його відкритих оферів, а також офери, що оброблюються системою та affiliate-менеджером.

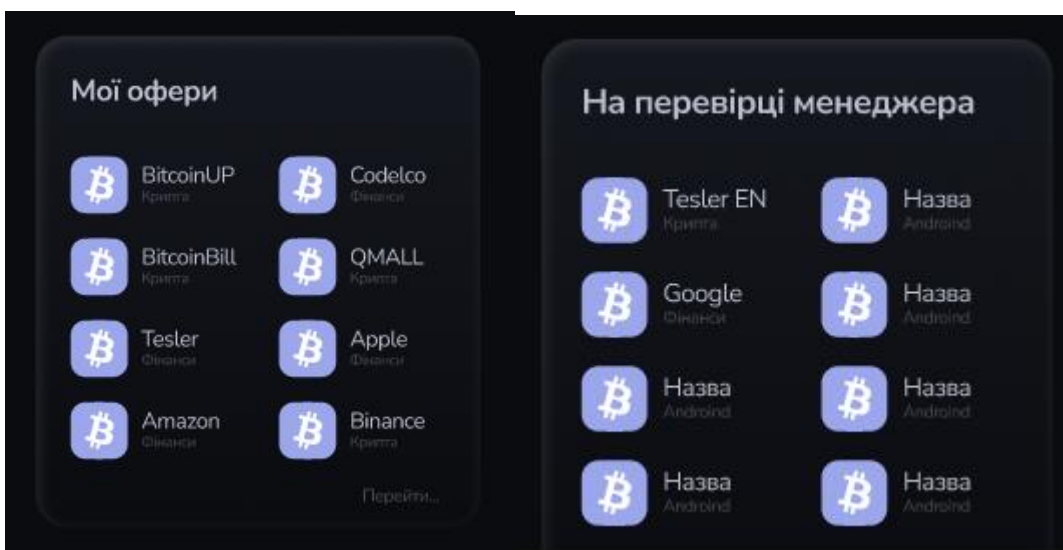


Рис. 3.15. Відображення коротких списків оферів

Натиснувши на кнопку “Поповнити” (рис. 3.16) на фіолетовій картці web майстер може поповнити свій баланс зручним для себе способом (рис. 3.17).

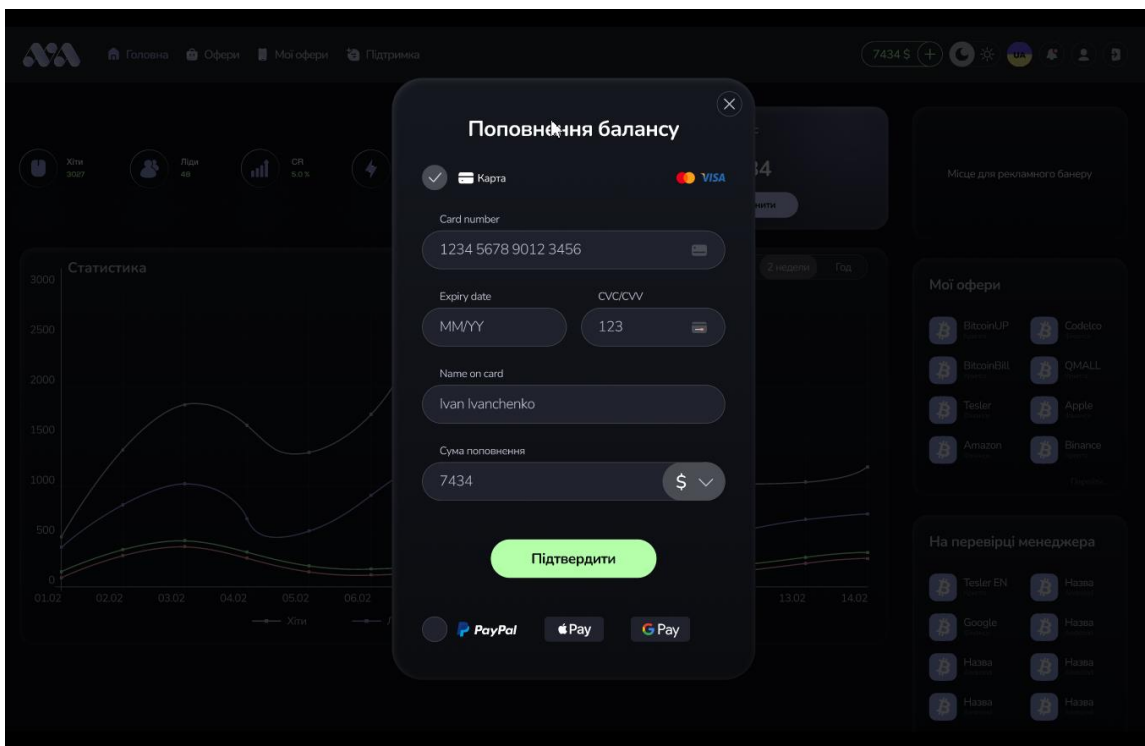


Рис. 3.16 – Заповнення даних картки для поповнення

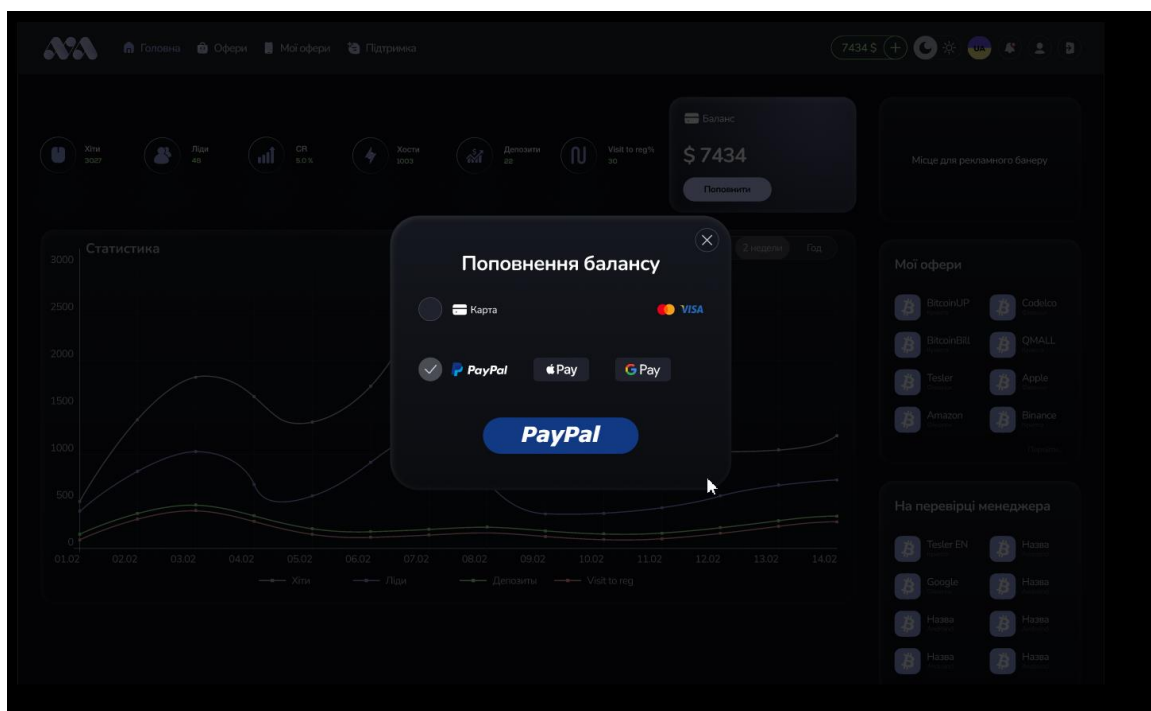


Рис. 3.17. Можливість скористатись іншим зручним способом поповнення
Натиснувши на кнопку “сонце і місяць”, розташовану в хедері, веб майстер може змінити кольорову тему застосунку(рис. 3.18).



Рис. 3.18. Світла кольорова тема застосунку

Також поряд з кнопкою зміни кольорової теми можна переглянути актуальні повідомлення системи, натиснувши кнопку дзвіночка (рис. 3.19).

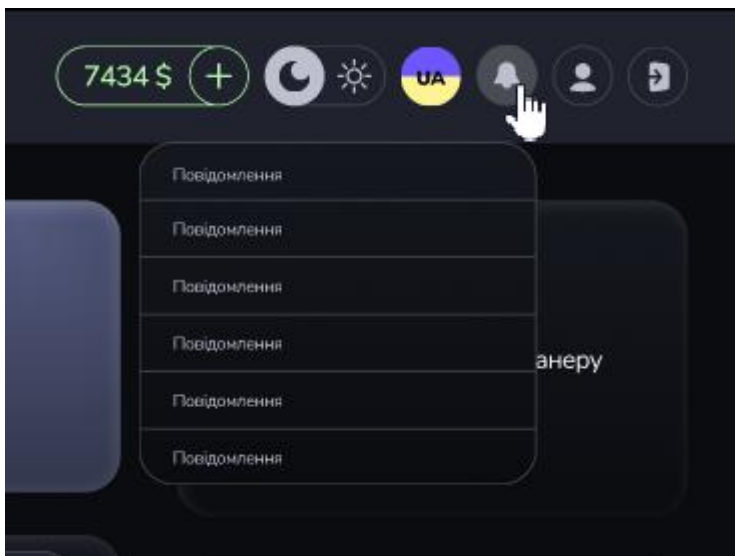


Рис. 3.19. Перегляд повідомлень

Поряд знаходиться кнопка переходу на сторінку з особистими даними (рис. 3.20), що вводились під час реєстрації з можливістю редагувати дані, а також доповнювати їх.

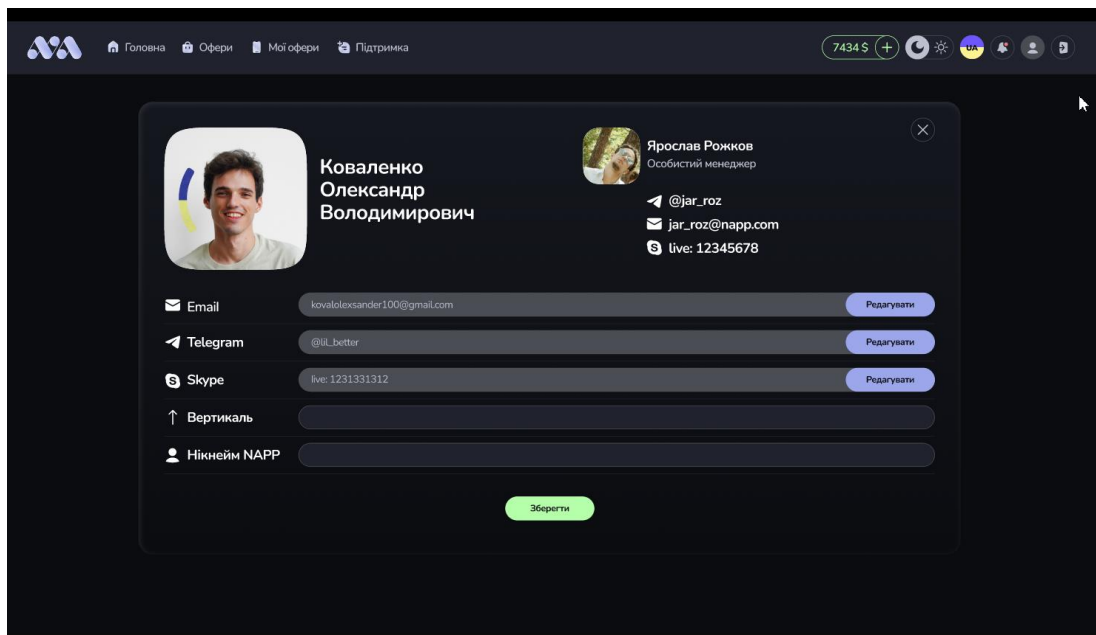


Рис. 3.20. Особистий кабінет

Також є можливість вийти із акаунту, натиснувши кнопку в правому верхньому куточку, в такому випадку ми потрапимо на сторінку авторизації.

Розглянемо більш детально основні функціональні сторінки застосунку. Між ними можна переключатись, використовуючи хедер застосунку (рис. 3.21).

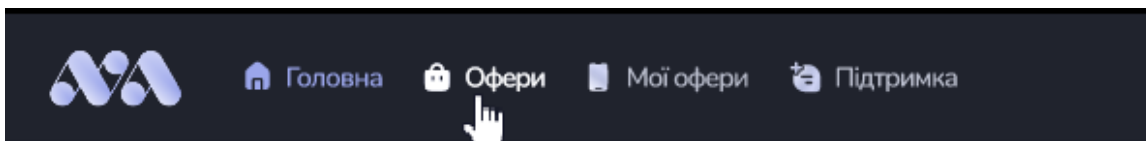


Рис. 3.21 Хедер

Перейдемо на сторінку оферів, тут відображається список доступних до відкриття оферів, що мають свої відповідні атрибути. На даній сторінці реалізовано зручну систему фільтрації для швидкого пошуку оферу за потрібним критерієм або характеристикою (рис. 3.22). Кожному оферу відповідає кнопка відкриття оферу “запросити”, при натисненні якої відбувається підтвердження оферу зі списанням з балансу відповідної суми коштів, що буде використана для запуску рекламної кампанії, про що інформує відповідний поп-ап (рис 3.23)

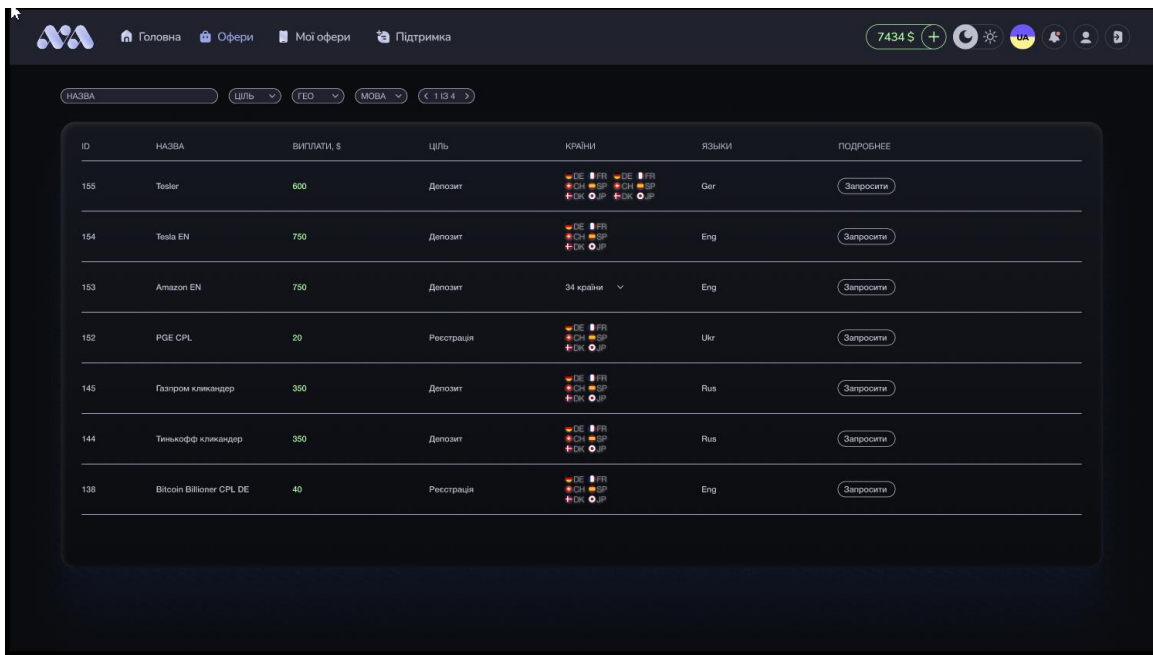


Рис. 3.22. Сторінка оферів

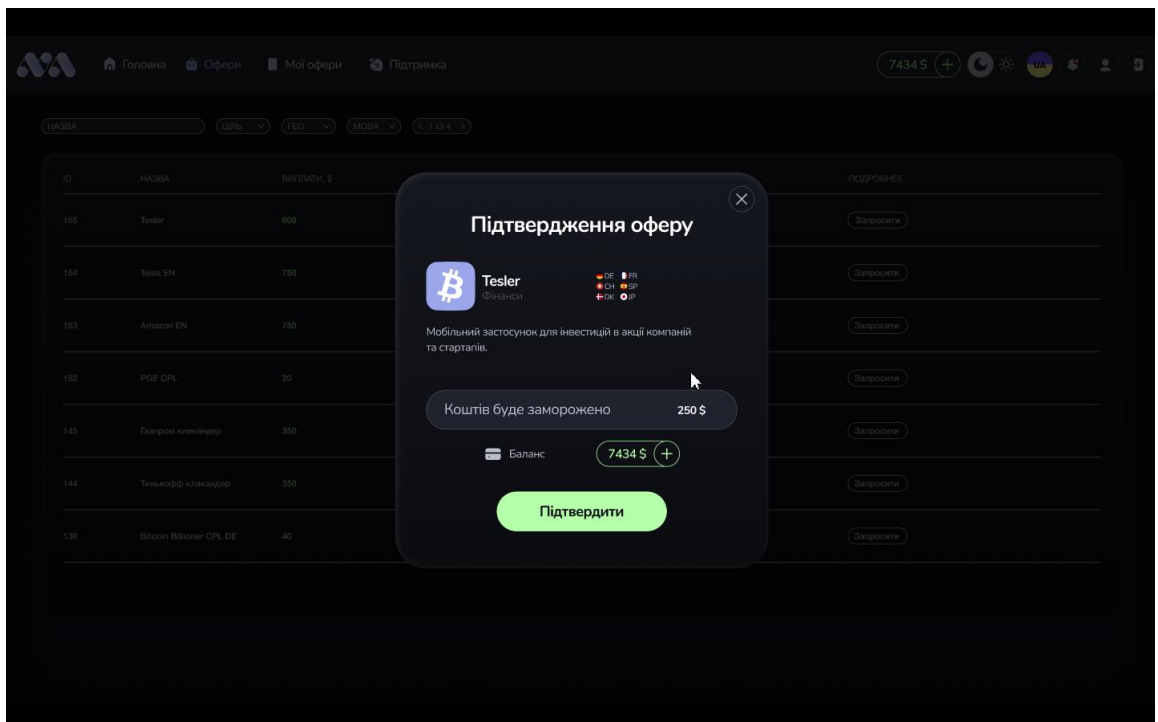


Рис. 3.23. Поп-ап підтвердження відкриття оферу

Після підтвердження відкриття оферу веб майстер має можливість відразу налаштувати рекламну кампанію(-ї) по ньому з отриманням необхідних скриптів та Tracking Link від системи (рис. 3.25), а також підключити рекламну кампанію з Google Analytic або Facebook Ads (рис. 3.24) для відслідковування статистики, налаштувати пуш повідомлення (рис. 3.26) потенційним клієнтам. Пуш повідомлення бувають трьох типів:

- По події (рис. 3.28)
- За розкладом (рис. 3.27)
- Моментальні

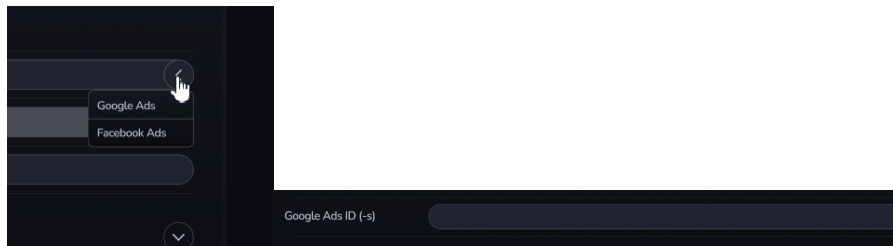


Рис. 3.24. Підключення Google Analytic

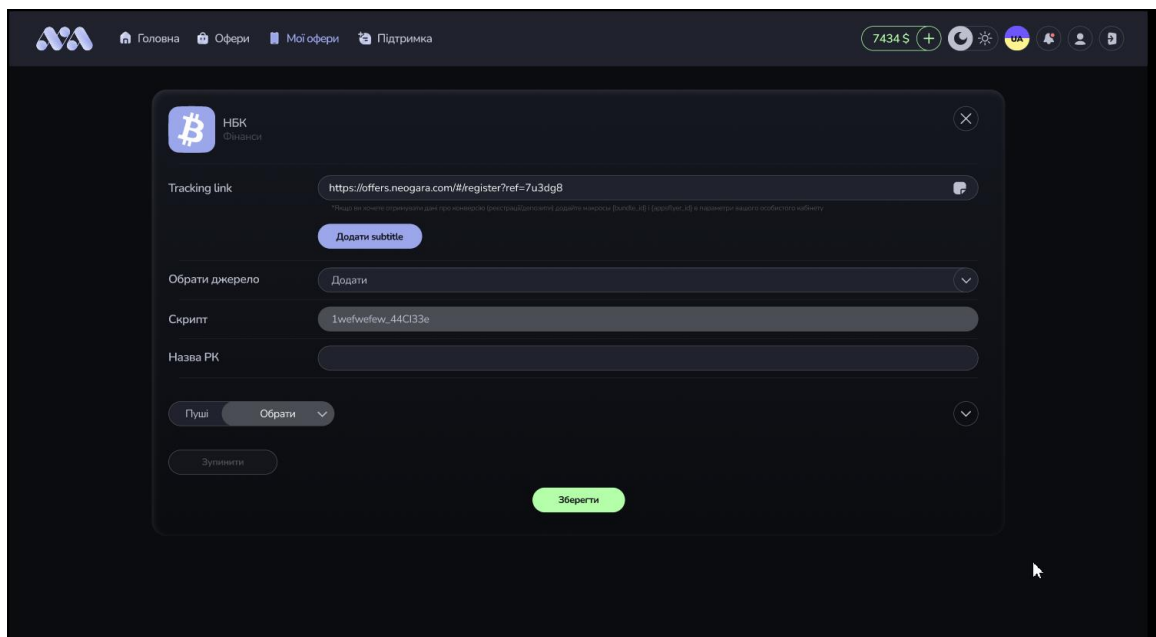


Рис. 3.25. Налаштування офери

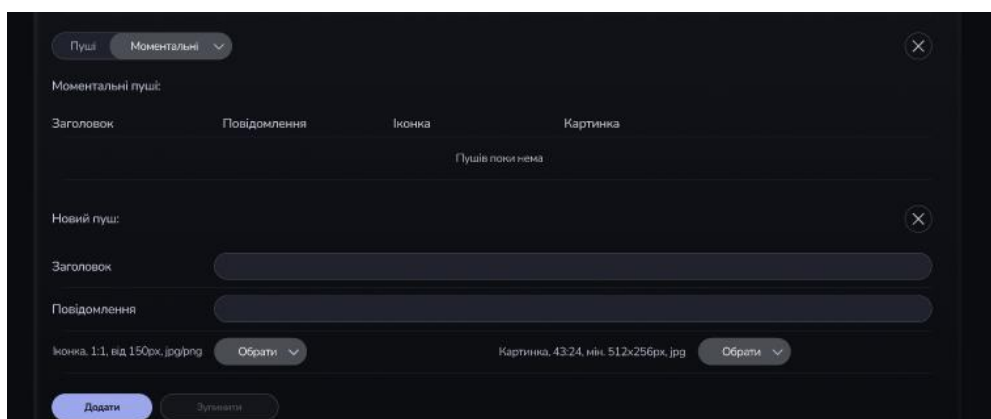


Рис. 3.26. Налаштування моментальних пушів

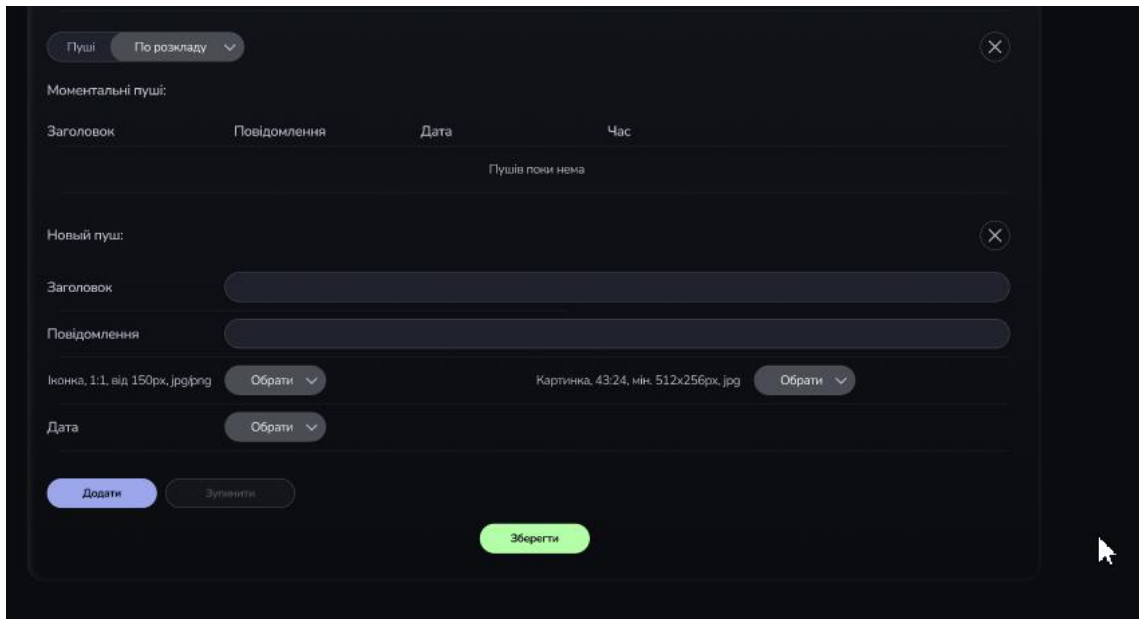


Рис. 3.27. Налаштування пушів за розкладом

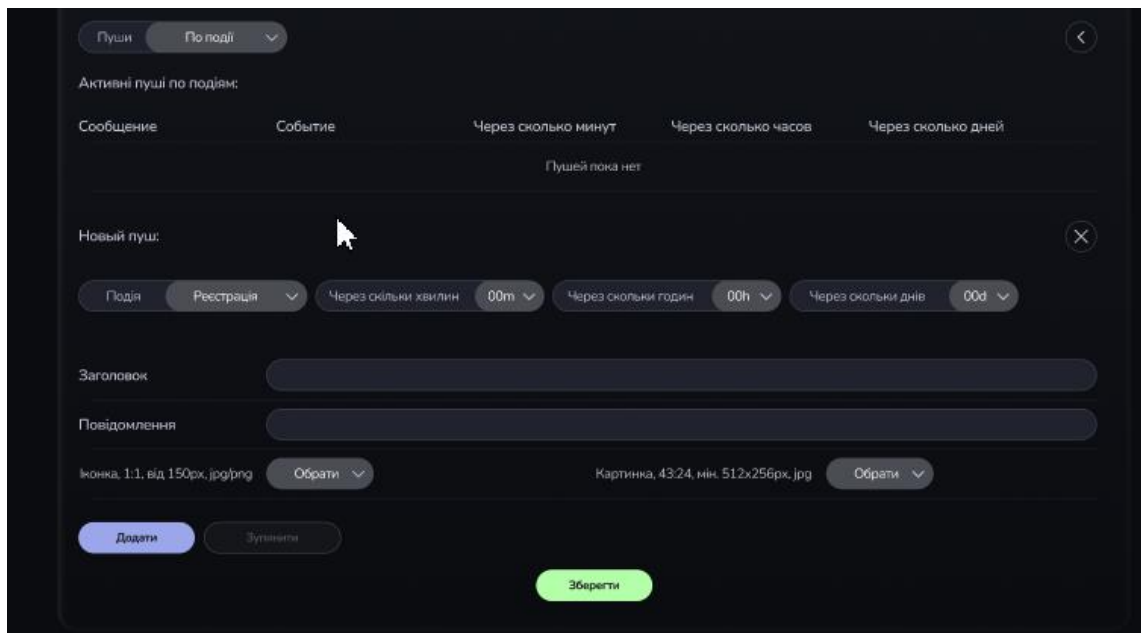


Рис. 3.28. Налаштування пушів по події

Після завершення налаштувань веб майстер натискає кнопку Зберегти, дані зберігаються в системі і вона починає відслідковувати статистику по відкритим оферам (рис. 3.29), також він одразу потрапляє на сторінку “Мої офери”, де він зможе переглянути конверсію кожного окремого оферу. Зелені поля означають успішне виконання клієнтом цільових дій, сині поля це цільові дії що ще обробляються і червоні це цільові дії, які були виконані з помилкою, тобто веб майстер не отримує прибутку з них.

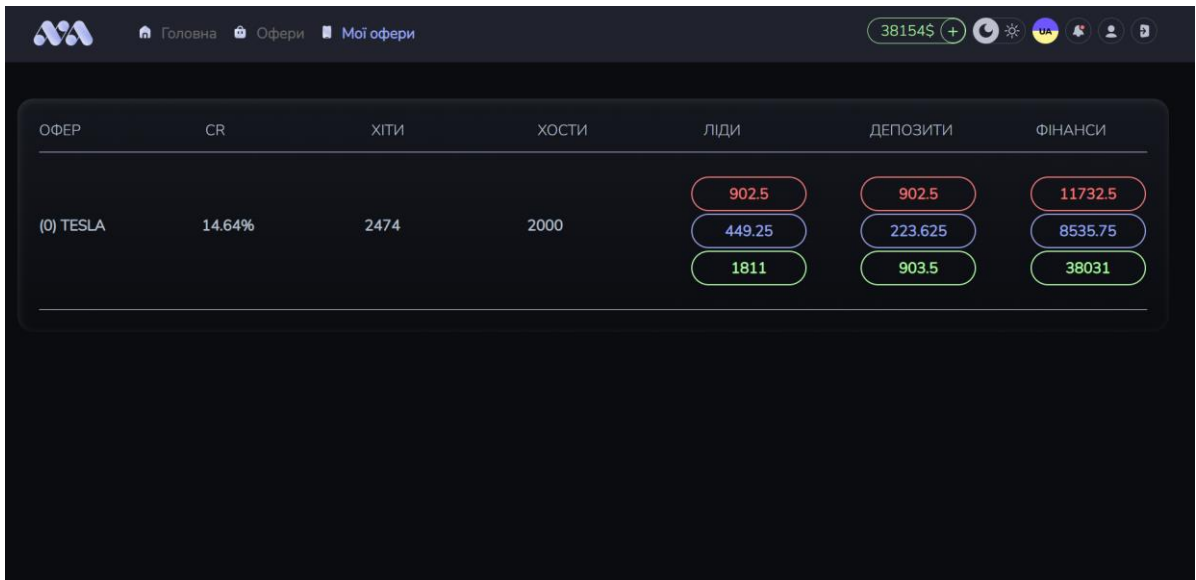


Рис. 3.29. Відображення статистики по оферам веб майстра

Також у веб майстра є можливість переглянути сторінку “Підтримка”, де є інформація про його особистого менеджера (рис. 3.30), та відповідний функціонал для перегляду відповідей на питання, що найчастіше виникають.

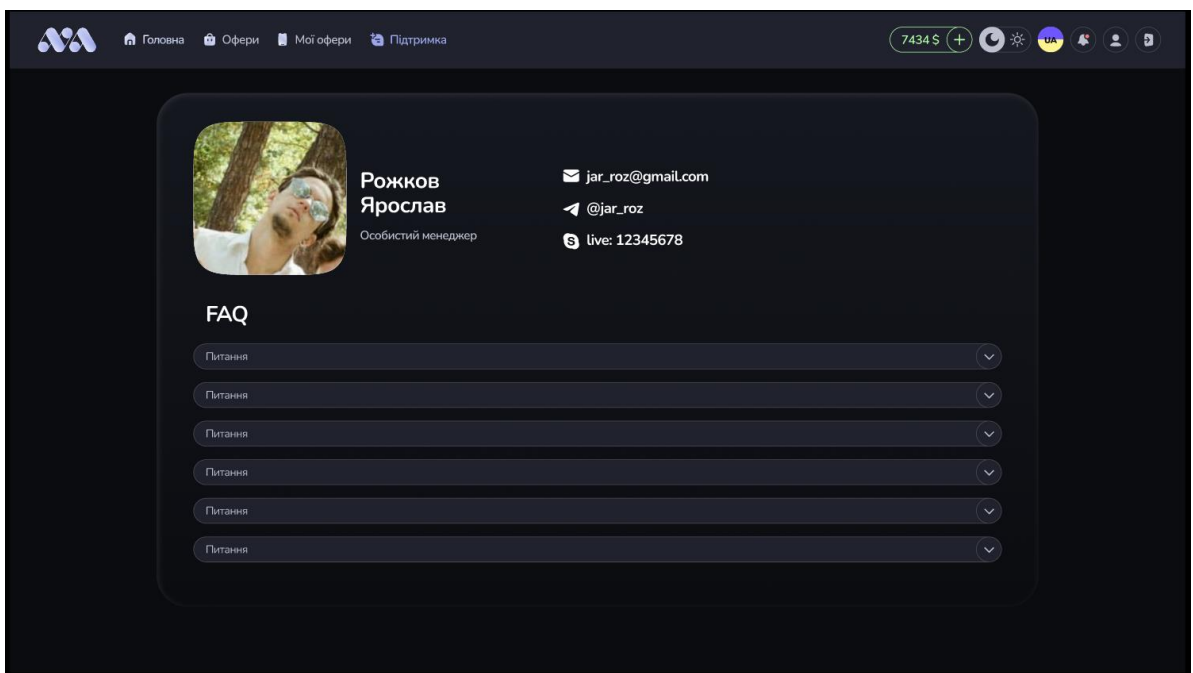


Рис. 3.30. Сторінка підтримки

3.9 Огляд процесу тестування

Фінальною стадією розробки будь-якого програмного продукту є процес тестування застосунку, опишемо Test case у таблиці 3.4.

Таблиця 3.4 Опис Test case

ID	Модуль	Подія	Очікувані результати
----	--------	-------	----------------------

1	Login.js	Реєстрація нового користувача: - Заповнити всі обов'язкові поля - Натиснути кнопку "Погоджуюсь з..." - Натиснути кнопку "ReCaptcha" - Натиснути кнопку "Відправити заявку"	1. Користувач переходить на сторінку авторизації 2. Всі обов'язкові поля заповнені
2	Login.js	Вхід до особистого кабінету: - введення правильного логіну в поле логін - введення правильного паролю в поле пароль - натиснення кнопки "увійти"	1. Успішна авторизація 2. Пароль відображено як крапки у полі паролю 3. Веб майстер переходить на головну сторінку
3	Login.js	Неправильні дані для авторизації: - Введення неправильного логіну - Ввести неправильного паролю - Ненатиснення кнопки "ReCaptcha"	1. Користувач залишається на сторінці авторизації, незаповнені поля підсвічуються червоною зірочкою.
4	MyOffer.js	Перегляд нового відкритого офери: - Натиснути кнопку "запросити" - Підключити Google або Facebook Ads - Натиснути кнопку "Зберегти"	1. Офер з'явився на сторінці "Мої офери" 2. Веб майстер потрапив на сторінку "Мої офери" 3. Веб майстер переглядає статистику по офери

4	MyOffer.js	Новий відкритий офер неправильно підключено: - Натиснути кнопку “запросити” - Не підключити Google або Facebook Ads - Натиснути кнопку “Зберегти”	1. Офер з’явився на сторінці “Мої офери” 2. Веб майстер потрапив на сторінку “Мої офери” 3. Статистика по оферу не відображається
---	------------	--	---

3.10 Економічна складова проєкту

Процес оцінки тривалості діяльностей включає визначення часу, необхідного для конкретних проєктних завдань.

1. Збір необхідної інформації.
2. Вибір відповідних методів оцінки.
3. Розбиття складних завдань у разі потреби.
4. Співпраця з членами команди та експертами.
5. Оцінка тривалості діяльності за допомогою обраного методу.
6. Документування оцінок та обмежень.
7. Перегляд та підтвердження з зацікавленими сторонами.
8. Оновлення графіка проєкту.
9. Моніторинг та контроль прогресу діяльностей.

Інвестиційна стратегія зменшення вартості проєкту включає:

1. Оцінку вартості: точна оцінка вартості проєкту шляхом ідентифікації ресурсів та врахування зовнішніх факторів.
2. Аналіз вартості та користі: оцінка ROI для компонентів проєкту, зосередження на високоефективних діях та усунення неважливих.
3. Інженерія вартості: аналіз компонентів проєкту для пошуку вартісних альтернатив без ушкодження якості.
4. Управління ризиками: виявлення та зменшення ризиків на ранніх етапах, щоб уникнути неочікуваних витрат.

5. Закупівля та управління постачанням: оптимізація витрат за рахунок конкурентного торгівлі, переговорів та ефективних відносин з постачальниками.
6. Постійний моніторинг: відстеження витрат, порівняння фактичних витрат з планованими та прийняття коригувальних заходів за необхідності.
7. Залучення зацікавлених сторін: залучення зацікавлених сторін до прийняття рішень, запит їхнього внеску та узгодження цілей.

Під час виконання цих стратегій можна досягти оптимізації вартості проекту, що призводить до покращення ефективності витрат, кращого використання ресурсів та збільшення вартості проекту. Це допомагає організаціям реалізувати проекти в межах бюджетних обмежень, з максимізацією віддачі від інвестицій.

Тож за результатами досліджень були визначені приблизні терміни, відповідно до попередньо змодельованої діаграми Ганта та опису робіт.

Завдання 1: Планування проекту

Тривалість: 10 днів

Ресурс: Витрати (Проектний менеджер)

Завдання 2: Збір вимог

Тривалість: 6 днів

Ресурс: Витрати (Бізнес-аналітик)

Завдання 3: Проектування та розробка

Тривалість: 30 днів

Ресурс: Витрати (Дизайнер)

Завдання 4: Тестування та контроль якості

Тривалість: 15 днів

Ресурс: Витрати (Тестувальник)

Завдання 5: Реклама

Примітка. Реклама продукту здійснюється після виконання робіт по розробці, тобто проєкт маркетингової кампанії це окремий проєкт після реалізації продукту проєкту.

Тривалість: 28 днів

Ресурс: Витрати (Маркетингова та дизайнерська команда)

У цьому розкладі кожне завдання має визначену тривалість, а ресурси, призначені для кожного завдання, є типу "Витрати". Ці ресурси представляють фінансові витрати, пов'язані з кожним завданням, такі як витрати на управління проєктом, витрати на аналіз бізнесу, витрати на розробку та витрати на тестування.

1 дизайнер коштує 40 доларів за годину, 1 керівник проєкту - 100 доларів за годину, 1 програміст - 60 доларів за годину, менеджер по PR - 30, а тестувальники - 45. Проєкт називається платформою CPA мережі, на якому працюють 3 дизайнери, 5 програмістів, 1 керівник проєкту, 2 тестувальники та 2 менеджери по PR.

Таблиця 3.5 Економічне планування

Task	Duration	Resource	Ціна за годину (\$)	Total Cost (in \$)
Project Planning	10 days	Project Manager	100	8000
Requirement Gathering	3 days	Business Analyst	40	960
Design and Development	30 days (22 working)	Development and Design Team	Programmer: 60	10560
			Designer: 40	7040

			Веб розробник: 25	3000
Testing and QA	15 days	Testing Team	Tester: 45	5400
Advertising	28 days	Marketing and Design Team	Designer: 40	8960
			PR Manager: 30	6720
Total Cost				50,640

В результаті було отримано середню вартість ІТ-проекту в сучасній індустрії. Якщо додати супутні витрати, отримаємо +-60 тис.\$, а також потрібно буде враховувати витрати на закупку трафіку.

3.11 Висновки до третього розділу

Для розробки програмного продукту було проведено ряд досліджень, у тому числі UX/UI експертиза. Були досліджені основні потреби користувача і знайдено ефективні рішення для покращення користувацького досвіду, а отже і ефективності його взаємодії з інформаційною системою.

Була розроблена інтелектуальна карта проекту, сформована організаційна структура проекту. Побудована діаграма Ганта, яка проілюструвала ефективність обраної методології управління проектом в контексті затримок на різних етапах. Було проведено ресурсне планування та визначення вимог до команди та організації роботи, що дозволило закласти необхідний бюджет на розробку проекту.

Досліджений процес функціонування системи, вплив зовнішніх зв'язків.

Було детально описане керівництво користувача та проведений огляд процесу тестування, що дало змогу оцінити якість проведених робіт та перевірити працездатність продукту проекту. Продукт повністю функціонує у відповідності з поставленими задачами.

Для розробки були обрані сучасні інструменти та під час розробки була побудована структура програмного веб застосунку і описані специфікації кожного модуля.

Окрім цього, на фінальному етапі був проведений огляд випадків тестування для перевірки функціонування, які були описані в таблиці.

РОЗДІЛ 4. РОБОТА ЗІ СТЕЙКХОЛДЕРАМИ. УПРАВЛІННЯ РИЗИКАМИ

4.1 Аналітичний огляд зацікавлених сторін та вплив на проєкт

Після завершення проєкту розробки веб додатку CRM системи для CPA мережі, на базі основної команди будуть створені підрозділи, що будуть підпорядковуватись керівнику проєкту для подальшої роботи і розвитку продукту на ринку, отже організаційна структура буде трансформована і із однієї команди сформується 5 нових з новими співробітниками (Рис. 4.1).

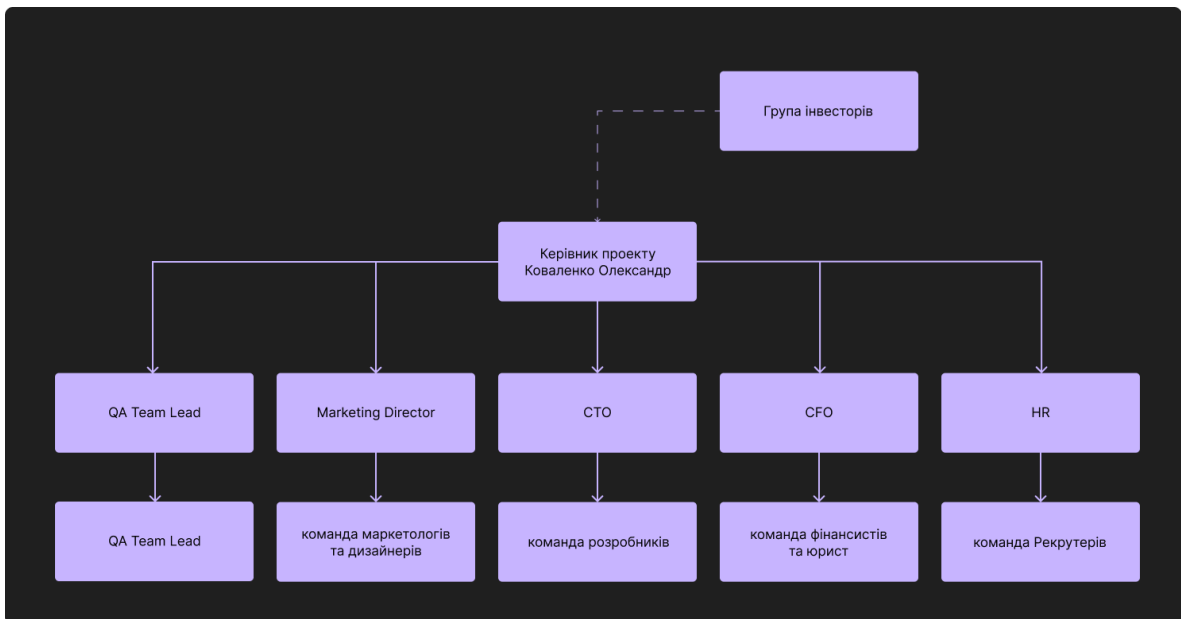


Рис. 4.1. Організаційна структура після реалізації проєкту

Проєкт розробки веб-додатка CRM-системи для CPA мережі включає взаємодію з різноманітними стейкхолдерами, кожен з яких має свої інтереси та внесок у проєкт. Визначення та розуміння цих стейкхолдерів є критично важливим для успішного впровадження проєкту. Ось основні категорії стейкхолдерів:

Внутрішні стейкхолдери

Керівництво компанії:

Інтереси: Рентабельність інвестицій, успішне впровадження системи, зростання бізнесу.

Вплив: Надають фінансування та стратегічне керівництво.

Проєктна команда (розробники, дизайнери, тестувальник. тощо):

Інтереси: Чітке визначення завдань та термінів, професійний розвиток.

Вплив: Повсякденна робота над проектом, визначення технічних деталей та забезпечення якості.

Відділ маркетингу:

Інтереси: Ефективність CRM-системи, здатність приваблювати та утримувати клієнтів.

Вплив: Впливають на функціональні вимоги та маркетинговий успіх продукту.

Зовнішні стейкхолдери

Клієнти та користувачі (веб майстри, рекламодавці та їх афілійовані партнери):

Інтереси: Надійність, інтуїтивність та функціональність CRM-системи.

Вплив: Визначають успішність проекту через свої потреби та відгуки.

Постачальники технологічних рішень:

Інтереси: Продаж технологічних компонентів або сервісів.

Вплив: Можуть обмежувати або розширювати технічні можливості CRM-системи.

Законодавці та регулятори:

Інтереси: Дотримання законодавчих рамок, особливо в частині захисту даних (наприклад, GDPR).

Вплив: Встановлюють правила, яким повинен відповідати продукт.

Інвестори та фінансові партнери:

Інтереси: Віддача від інвестицій, стабільність і зростання компанії.

Вплив: Можуть впливати на фінансування та розширення проекту.

Для ефективного управління проектом важливо розуміти та постійно аналізувати очікування та потреби всіх стейкхолдерів, забезпечуючи їхнє залучення та підтримку. Регулярна комунікація та зворотний зв'язок допоможуть уникнути непорозумінь і сприятимуть успішній реалізації проекту. Розберемо більш детально стейкхолдерів проекту створення веб застосунку CRM системи для CPA мережі.

Таблиця 4.1 Аналітичний огляд стейкхолдерів проекту

	Зацікавлені сторони	Вплив ЗС на проєкт	Вплив результатів проєкту на ЗС
Внутрішні зацікавлені сторони			
1	Product Owner	<ul style="list-style-type: none"> + Визначає стратегічні цілі продукту + Перевірка відповідності проєкту цілям + Створює можливості для якісної реалізації продукту + Забезпечує фінансування продукту 	<ul style="list-style-type: none"> + Отримання прибутку + Отримання позитивної репутації + Створить можливість підтримки та подальшого розвитку проєкту + Допоможе закріпити високі позиції компанії на ринку
2	Project Manager	<ul style="list-style-type: none"> + Керує обмеженнями проєкту + Забезпечує якісні комунікації із зацікавленими сторонами + Забезпечує ефективний розподіл задач у команді + Забезпечує досягнення результатів проєкту 	<ul style="list-style-type: none"> + Поповнення портфеля проєктів + Отримання досвіду + Отримання матеріальної нагороди + Отримання позитивної репутації
3	Команда проєкту	<ul style="list-style-type: none"> + Забезпечує розробку продукту + Забезпечує якість продукту + Забезпечує успішне впровадження і просування продукту 	<ul style="list-style-type: none"> + Отримання досвіду + Отримання матеріальної винагороди + Приналежність до ідеї, що має позитивний вплив на суспільство + Отримання позитивної репутації
4	Інвестори	<ul style="list-style-type: none"> + Забезпечують фінансування продукту + Додатковий моніторинг якості продукту 	<ul style="list-style-type: none"> + Отримання прибутку + Отримання акцій компанії + Проєкт для інвестиційного портфелю
Зовнішні зацікавлені сторони			
1	Кінцеві користувачі	<ul style="list-style-type: none"> + Підвищення іміджу продукту + залучення нових клієнтів + Отримання прибутку 	<ul style="list-style-type: none"> + Забезпечення якісними послугами + Зручна та швидка робота екосистеми
2	Конкуруючі платформи	<ul style="list-style-type: none"> + Можуть чинити спротив на законному рівні + Можуть дискредитувати продукт 	<ul style="list-style-type: none"> + Зменшення кількості клієнтів + Зменшення прибутку + Зниження довіри клієнтів

4.2 Управління ризиками

PEST аналіз є інструментом стратегічного планування, який дозволяє оцінити зовнішні фактори, які можуть вплинути на проект. Для проекту розробки веб-додатка CRM системи для CPA мережі, розглянемо чотири основні аспекти: Політичні (*Political*), Економічні (*Economic*), Соціально-культурні (*Social*), Технологічні (*Technological*) фактори.

Політичні фактори можуть мати вагомий вплив на діяльність компаній, що працюють у різних галузях та регіонах. Характер та ступінь впливу цих факторів варіюються залежно від специфіки бізнесу, його місцезнаходження, міжнародної присутності та інших параметрів. Провівши експертне оцінювання можна зробити припущення щодо політичних факторів та їх впливу на проект:

Таблиця 4.2 Характер та ступінь впливу політичних факторів на проект

Фактори впливу політичного середовища	Характер впливу фактору	Оцінка експертів ступеню впливу фактору			Середній бал
		Експерт 1	Експерт 2	Експерт 3	
Кількісні та якісні обмеження на імпорт	-	1	1	1	-1
Бюрократизація та рівень корупції	-	2	2	1	-1.66
Стійкість політичної влади та існуючого уряду	+	3	1	2	+2
Податкова політика держави	+	3	3	3	+3
Вірогідність розвитку військових подій в країні	+	2	2	2	+2
Тенденції до регулювання та дерегулювання галузі	-	1	2	2	-1.66
Антимонопольне та трудове законодавство	+	2	3	2	+2.33
Майбутнє та поточне законодавство, що регулює правила роботи галузі	-	3	2	1	-2
Підтримка інноваційних компаній з боку держави	+	3	2	3	+ 2.66

Лібералізація зовнішньоекономічного співробітництва	+	3	3	3	+3
---	---	---	---	---	----

Як і очікувалось, в державі, правовими нормами якої керується компанія має тенденції до політичної еволюції та реформ, при цьому існує дестабілізуючий фактор війни на території країни.

Економічні фактори мають значний вплив на діяльність компаній і їх стратегічне планування. Вони включають широкий спектр елементів, таких як макроекономічні умови, валютні коливання, інфляція, рівень безробіття та багато іншого. Провівши експертне оцінювання можна зробити припущення щодо економічних факторів та їх впливу на проєкт:

Таблиця 4.3 Характер та ступінь впливу економічних факторів

Фактори економічного середовища	впливу	Характер впливу фактору	Оцінка експертів ступеню впливу фактору			Середній бал
			Експерт 1	Експерт 2	Експерт 3	
Темпи економіки	зростання	+	3	3	2	+2.66
Рівень інфляції		-	2	3	2	-2.33
Курси основних валют		-	1	1	1	-1
Рівні наявних доходів населення		+	3	3	3	+3
Ступінь глобалізації та відкритості економіки		+	3	3	2	+2.66
Монетарна та фіскальна політика держави		-	1	1	1	-1
Рівень розвитку підприємництва та бізнес середовища		+	2	3	3	+2.66
Цінова конкуренція з боку зарубіжних компаній		+	3	2	3	+2.66

Інвестиційний клімат в галузі	+	3	3	3	+3
-------------------------------	---	---	---	---	----

Вести бізнес доводиться в нестабільній економічній обстановці, через це потрібно забезпечити потік коштів у стабільній валюті, наприклад доллар США. Це допоможе компанії залишитись на плаву у випадку краху економіки країни через високу інфляцію або вплив інших дестабілізуючих чинників.

Соціально-культурні фактори включають демографічні тенденції, суспільні цінності, культурні норми, освітні рівні, та зміни у стилях життя.

Таблиця 4.4 Характер та ступінь впливу соціально-культурних факторів на компанію

Фактори впливу соціально-культурного середовища	Характер впливу фактору	Оцінка експертів ступеню впливу фактору			Середній бал
		Експерт 1	Експерт 2	Експерт 3	
Рівень підготовки молодих спеціалістів в галузі	+	2	2	1	+1.66
Вимоги до якості продукції та рівня сервісу	+	3	3	3	+3
Рівень міграції та імміграційні настрої	+	1	1	1	+1
Культура формування заощаджень і кредитування суспільства	+	2	2	2	+2
Розвиток релігії та інших вірувань	+	0	0	0	0
Ставлення до імпортованих товарів та послуг	+	1	1	1	+1
Спосіб життя і звички споживання	+	2	2	2	+2
Темпи росту населення	+	2	2	1	+1.66

Один із найпозитивніших результатів, висока концентрація молоді, високий рівень освіти та діджиталізація держави створюють сприятливі умови для створення нових молодих команд, а згодом і компаній.

Технологічні фактори відіграють *критичну* роль у сучасному бізнес-середовищі, особливо в предметній галузі кваліфікаційної роботи магістра. Інновації, оновлення технологій та їх швидке впровадження можуть значно вплинути на стратегії компанії, її конкурентоспроможність і майбутній розвиток. Ось основні аспекти технологічних факторів та їх вплив на компанію:

Таблиця 4.5 Характер та ступінь впливу технологічних факторів на компанію

Фактори впливу технологічного середовища	Характер впливу фактору	Оцінка експертів ступеню впливу фактору			Середній бал
		Експерт 1	Експерт 2	Експерт 3	
Рівень інновації та технологічного розвитку галузі	+	3	3	3	+3
Ступінь використання, впровадження та передачі технологій	+	3	3	2	+2.66
Доступ до новітніх технологій	+	3	3	3	+3
Витрати на дослідження та розробки	+	2	2	3	+2.33
Можливість виробництва якісно нової продукції (розвиток конкурентних технологій)	+	2	3	3	+2.66
Законодавство в галузі технологічного оснащення галузі	-	1	2	2	-1.66

Єдиний негативний чинник, це правові обмеження щодо технологічних компаній, через велике податкове навантаження, зарубіжні гіганти не зацікавлені відкривати підприємства та представництва в Україні, через що доступ та регулювання відношень між ІТ компаніями та зарубіжними гігантами Індустрії є обмеженим.

Таблиця 4.6 Фактори, які спричиняють найбільший вплив на діяльність компанії за результатами PEST-аналізу

Політичні		Економічні	
Фактор	Вага	Фактор	Вага
Майбутнє та поточне законодавство, що регулює правила роботи галузі	-2	Рівень інфляції	-2.33
Підтримка інноваційних компаній з боку держави	+2.66	Інвестиційний клімат в галузі	+3
Соціально-культурні		Технологічні	
Розвиток релігії та інших вірувань	0	Законодавство в галузі технологічного оснащення галузі	-1.66
Вимоги до якості продукції та рівня сервісу	+3	Рівень інновації та технологічного розвитку галузі	+3

4.2.1 Підсумки здійснення PEST-аналізу для компанії

На основі проведеного PEST аналізу проекту розробки веб-додатка CRM-системи для CPA мережі можна зробити декілька важливих висновків, що допоможуть формулювати стратегії та приймати обґрунтовані рішення

Таблиця 4.7 Підсумки PEST аналізу та дії

Фактори	Зміни в галузі	Зміни в організації	Дії
Політичні	<ul style="list-style-type: none"> Підсилення спроб зі сторони уряду регулювати галузь криптовалют Наявність державних інвестицій та спрощення користування криптовалютами продуктами 	<ul style="list-style-type: none"> Додаткові витрати на формування сприятливого середовища Спрощення реалізації інноваційних ідей та зменшення терміну впровадження 	<ul style="list-style-type: none"> Створення і впровадження нових контрмір для боротьби з впливом уряду на сферу діяльності Збільшення ефективності відділу з впровадження інновацій
Економічні	<ul style="list-style-type: none"> Нестабільність на 	<ul style="list-style-type: none"> Нестійкість 	<ul style="list-style-type: none"> Залучення експертів

	<p>ринку інвестицій</p> <ul style="list-style-type: none"> • Наявність великого обсягу інвестицій 	<p>розвитку та додаткові труднощі у плануванні подальших дій</p> <ul style="list-style-type: none"> • Впевненість у майбутньому розвитку та додаткові можливості покращення продукту 	<p>зادля покращення фінансового плану та мінімізації втрат у темпах розвитку.</p> <ul style="list-style-type: none"> • Розширення якісної та кількісної складової команди та реалізація додаткових проєктів компанії.
Соціально-культурні	<ul style="list-style-type: none"> • Вплив відсутній • Постійні зміни в стандартах продукту та послуг 	<ul style="list-style-type: none"> • Зміни відсутні • Постійні потреби у вдосконаленні послуг та продукту 	<ul style="list-style-type: none"> • Додаткові дослідження та спроби впровадження змін на користь компанії • Додатковий контроль якості продукту. Покращення ефективності відділу інновацій, тестування та розробки.
Технологічні	<ul style="list-style-type: none"> • Додаткові лімітації у галузі обладнання • Залучення нових технологій, поширення впливу галузі на життя людей 	<ul style="list-style-type: none"> • Складнощі з технічним забезпеченням • Впровадження нових технологій, створення інноваційних продуктів 	<ul style="list-style-type: none"> • Пошуки нових шляхів отримання необхідного обладнання. Посилення юридичного відділу компанії. • Забезпечити інноваційні проєкти якісною командою розробників.

Висновки, отримані з PEST аналізу, є фундаментальними для прийняття обґрунтованих стратегічних рішень. Вони дозволяють компанії адекватно реагувати на зовнішні виклики, адаптуватися до змінюваних умов ринку та оптимізувати свої ресурси для досягнення максимальної ефективності. Ігнорування цих висновків може призвести до втрати конкурентних переваг, зниження прибутків або навіть до згорання бізнесу. Тому кожен аспект PEST аналізу має бути ретельно проаналізований та врахований у стратегічному плануванні проєкту, щоб забезпечити його сталість і успішність на довготривалій період.

Таблиця 4.8 Цілі, показники досягнення та ризики

Цілі	Показники досягнення	Вимірювачі (індикатори)	Припущення та ризики
Стати першою CPA мережею з оптимізованим веб додатком CRM системи з автоматичним відкриттям офери	Швидкість зворотнього зв'язку системи, кількість витрат на підтримку проекту	Скорочення витрат на Affiliate менеджерів, збільшення швидкості віддачі системи	Незручність технології на перших етапах запуску
Створити єдину екосистему	Можливість взаємодії між CRM-системою, рекламодавцем, веб майстром та рекламним кабінетом	Показники отриманої аналітики в особистому кабінеті веб додатку	Технічні недоліки системи, високі комісії постачальників послуг
Обслуговувати 10 000 веб майстрів на рік	Популярність в соціальних мережах, кількість онлайн у веб додатку	10 000 унікальних акаунтів, що залили хоча б одну кампанію	Сильні конкуренти, лімітації серверів, антимонопольна політика держави
Створити ком'юніті веб майстрів в Україні	Популярність CPA маркетингу у сфері	Збільшення кількості оферів, апуцених кампаній та об'ємів трафіку	Втрата актуальності технології
Створити власний AI інструмент для роботи з рекламними кампаніями	Поява AI додатку CRM системи	Запуск AI інструментарію, та отримання першого профіта	Неефективне машинне навчання AI інструменту
Залучити 1000 користувачів від реферальної програми	Активне розповсюдження реферальних посилань у соціальних мережах	Показники аналітики реферальних посилань	Непривабливі умови реферальної програми
Залучити 5.000 користувачів через соц мережі та рекламу в інтернеті	Активне поширення інформації про платформу у соціальних мережах	Показники аналітики FB, Google та статистики переходів з тематичних форумів	Погана маркетингова стратегія. Непривабливий дизайн реклами.
Забезпечити мінімальну комісію на ринку	Високі показники retention у веб майстрів	Процент повернення користувачів 90%	Низька прибутковість від мінімальної комісії

<i>Цілі</i>	<i>Показники досягнення</i>	<i>Вимірювачі (індикатори)</i>	<i>Припущення та ризики</i>
Створити онлайн курси з CPA маркетингу	Залучення талантів до роботи з CPA мережею	Конверсія з курсів до платформи 10%	Незацікавленість аудиторії у курсах
Налагодити партнерські зв'язки	Проведення кросплатформної маркетингової кампанії з топами індустрії	Наявність бонусів та партнерства з топ-10 компаній індустрії	Незацікавленість партнерів. Низька вигода від партнерства

PEST аналіз підкреслює важливість комплексного підходу до планування та реалізації проекту розробки CRM-системи для CPA мережі. Врахування політичних, економічних, соціально-культурних та технологічних аспектів дозволить мінімізувати ризики та максимізувати можливості, забезпечуючи успішний запуск та експлуатацію продукту.

ВИСНОВОК

CPA маркетинг кожного дня дедалі більше набирає популярність, тисячі людей приймають рішення розвиватись і працювати у цій сфері, більшість крупних компаній вже користуються послугами партнерських програм. А це все означає, що наразі є актуальною тема розробки веб застосунку для CPA мережі, так як існуючі рішення не задовольняють потреб сучасного користувача. Новий програмний продукт, що закриває “болі” веб майстрів і при цьому є прозорим для рекламодавцям здатен змінити цінність компанії на ринку і залучити більше веб майстрів та рекламодавців.

В кваліфікаційній роботі магістра на тему *“Дослідження процесів управління проектом розробки веб-додатку CRM системи для CPA мережі”* була розроблена система управління, яка забезпечить успішне виконання всіх поставлених задач та досягнення цілей розробки проекту в досліджуваній галузі.

Було запропоноване інноваційне рішення щодо оптимізації роботи партнерських мереж (CPA мереж) на прикладі продукту проекту, що дозволило створити продукт з мінімальною кількістю витрат та в заданий термін. Також було оптимізоване використання математичних моделей CPA маркетингу і запропоноване рішення з підключенням API Google та Facebook.

Кожен етап життєвого циклу проекту був проаналізований та запланований. Було досліджено предметну галузь та існуючі рішення, проаналізовано існуючі методології управління проектами та обрано найбільш ефективну *Agile* методологію на основі фреймворку *KanBan*, ефективність якого була доведена під час стадії розробки продукту проекту. Були *сформульовані вимоги* до проекту та *чітко поставлені задачі* на розробку, обгрунтовано проблематику досліджуваного предметного середовища через *діаграми Ішикави та дерева проблем та рішень*. *Проведено SWOT аналіз проекту*, проведене детальне *системне проектування* продукту, включаючи функціональний аналіз, декомпозицію основних процесів, побудову архітектури інформаційної системи використовуючи контекстні діаграми, діаграми декомпозиції процесів, бізнес моделі, архітектура інформаційної системи, була спроектована база даних, що зберігає необхідні для

роботи дані. Був досліджений процес роботи API сервісів Google і Facebook, що є необхідним для роботи веб майстрів.

Для коректного функціонування продукту також було впроваджене математичне та інформаційне забезпечення веб додатку CRM системи для CPA мережі. Для розробки також було проведено ряд *UX/UI досліджень* для побудови правильної логіки, *User flow*, ефективного розміщення функціоналу системи у форматі зручного інтерфейсу з модулями візуалізації даних та проведено тестування готового продукту.

Для проєкту розроблена організаційна структура з подальшим планом розвитку компанії. З використанням *діаграми Ганта* спланована робота всіх членів команди проєкту, відслідковане виконання кожної задачі проєкту з *максимально ефективним використанням часу та трудових ресурсів*. Завдяки *гнучкості підходу* попри затримку на 2х етапах розробки продукту проєкту, весь проєкт був завершений в призначений термін, що відображений на моніторинг діаграмі Ганта.

Було змодельоване середовище існування проєкту, включаючи систему, *надсистему*, зовнішні сутності, *внутрішнє середовище* та зв'язки між ними.

Також були дослідженні 4 основні зовнішні фактори впливу на проєкт, що було відображено проведеним *PEST аналізом*.

Під час написання кваліфікаційної роботи магістра також було досліджено роботу зі стейкхолдерами проєкту, створене керівництво користувача веб додатка, досліджено більше 50 джерел, а також представлено лістинг частини коду в додатку до цієї роботи.

В результаті було розроблено систему управління проєктом створення повноцінного веб додатку CRM системи для CPA мережі, а також розроблено сам продукт проєкту, що має цінність на сучасному ринку і в перспективі може стати прибутковим проєктом.

СПИСОК ВИКОРИСТАНИХ ІНФОРМАЦІЙНИХ ДЖЕРЕЛ

1. Стаття про загальні моделі та особливості CPA маркетингу: [Електронний ресурс] // Режим доступу: <https://www.optimizesmart.com/how-to-calculate-maximum-cpa-and-minimum-roas/> [1]
2. Стаття опис роботи з Google Analytic: [Електронний ресурс] // Режим доступу: https://www.2checkout.com/how-to/interview/avinash-kaushik_7.htm [2]
3. “Бази даних. Проектування, реалізація і супроводження. Теорія і практика”, Томас Конноллі, Каролін Бегг, сс. 103-110 [3]
4. “Алан Купер про інтерфейс. Основи проектування взаємодії” [4]
5. Стаття опис роботи з Facebook Ads: [Електронний ресурс] // Режим доступу: <https://www.facebook.com/business/help/200000840044554> [7]
6. Create entire apps using JavaScript and Ajax (англ.). Aptana Inc (2009)., pp 34-55 [8]
7. M. Skalic, J. Jiménez, D. Sabbadin, i G. De Fabritiis, «Shape-Based Generative Modeling for de Novo Drug Design», J. Chem. Inf. Model., [9]
8. Морозов В.В. Чередніченко А.М., Шпильова Т.І. «Формування, управління та розвиток команди проекту». Київ, 2009. Бушуєв С. Д. Методології управління проектами на моделях класу "рушійні сили – опори" / С. Д. Бушуєв, Р. Ф. Ярошенко // Управління розвитком складних систем. - 2010.
9. PMBOK Guide. <https://www.pmi.org/pmbok-guidestandards/foundational/PMBOK>
10. Морозов В. В. Модель впливу зовнішнього оточення на процес управління конфігурацією в проекті [Електронний ресурс] / В. В. Морозов, С. И. Рудницький // Управління розвитком складних систем. - 2013. - Вип. 16. - С. 46-52.
11. Тімінський О. Г. Інформаційний захист управлінських та технологічних систем від зовнішніх негативних впливів в сучасному середовищі / О. Г. Тімінський // Вентиляція, освітлення та теплогазопостачання. - 2009. - Вип. 13. - С. 71-7

12. Ілляшенко С.М., Баскакова М.Ю. Маркетингові дослідження: Навч. посіб. Центр навчальної літератури, 2006. — 192 с.
13. Karlsen, J. T. Project stakeholder management / J. T. Karlsen // Engineering Management in the Global Environments, Proceedings. 2002. – pp. 65–70.
14. Chatfield, Carl. A short course in project management. Microsoft.
15. * The Definitive Guide to Project Management. Nokes, Sebastian. 2nd Ed.n. London (Financial Times / Prentice Hall): 2007. ISBN 978-0-273-71097-4
16. Paul C. Dinsmore et al (2005) The right projects done right! John Wiley and Sons, 2005. ISBN 0-7879-7113-8. p.35 and further.
17. Lewis R. Ireland (2006) Project Management. McGraw-Hill Professional, 2006. ISBN 0-07-147160-X. p.110.
18. Joseph Phillips (2003). PMP Project Management Professional Study Guide. McGraw-Hill Professional, 2003. ISBN 0-07-223062-2 p.354.
19. Dennis Lock (2007) Project management (9e ed.) Gower Publishing, Ltd., 2007. ISBN 0-566-08772-3
20. Young-Hoon Kwak (2005). «A brief history of Project Management». In: The story of managing projects. Elias G. Carayannis et al. (9 eds), Greenwood Publishing Group, 2005. ISBN 1-56720-506-2
21. David I. Cleland, Roland Gareis (2006). Global project management handbook. "Chapter 1: «The evolution of project management». McGraw-Hill Professional, 2006. ISBN 0-07-146045-4
22. Martin Stevens (2002). Project Management Pathways. Association for Project Management. APM Publishing Limited, 2002 ISBN 1-903494-01-X p.xxii
23. Morgen Witzel (2003). Fifty key figures in management. Routledge, 2003. ISBN 0-415-36977-0. p. 96-101.
24. David I. Cleland, Roland Gareis (2006). Global project management handbook. McGraw-Hill Professional, 2006. ISBN 0-07-146045-4. p.1-4 states: «It was in the 1950s when project management was formally recognized as a distinct contribution arising from the management discipline.»
25. Booz Allen Hamilton — History of Booz Allen 1950s.

26. Bjarne Kousholt (2007). Project Management -. Theory and practice.. Nyt Teknisk Forlag. ISBN 87-571-2603-8. p.59.
27. F. L. Harrison, Dennis Lock (2004). Advanced project management: a structured approach. Gower Publishing, Ltd., 2004. ISBN 0-566-07822-8. p.34.
28. Guidance on project management: ISO 21500:2012 [Електронний ресурс]. – Режим доступу: <https://www.iso.org/standard/50003.html>.
29. What are Agile Methodologies? How & When To Use Them [+Example] <https://thedigitalprojectmanager.com/projects/pm-methodology/agile-methodologies/>
30. CPA Marketing: A Low-Risk, High-ROI Approach to Affiliate Marketing <https://www.bigcommerce.com/articles/ecommerce/cpa-marketing/>
31. В. Бакуменко. Управління проектами // Політична енциклопедія. Редкол.: Ю. Левенець (голова), Ю. Шаповал (заст. голови) та ін. — К.: Парламентське видавництво, 2011. — с.738 ISBN 978-966-611-818-2
32. Kenton W. (2023, October 23) Investopedia: *SWOT Analysis: How To With Table and Example* URL: <https://www.investopedia.com/terms/s/swot.asp>
33. Different ways to calculate your CPA <https://www.bigcommerce.com/articles/ecommerce/cpa-marketing/>
34. What is Affiliate Marketing and How to Start in 2024 <https://www.simplilearn.com/tutorials/digital-marketing-tutorial/what-is-affiliate-marketing>
35. Gantt Charting: Definition, Benefits, and How They're Used. [Електронний ресурс]. – Режим доступу: <https://www.investopedia.com/terms/g/gantt-chart.asp>
36. Моделі й методи прийняття рішень: навч. посіб. / С.А. Ус, Л.С. Коряшкіна; М-во освіти і науки України, Нац. гірн. ун-т. – Д. : НГУ, 2014. – 300 с.
37. Best Crypto CPA Networks in 2024: Your Path to Profit <https://allpushnetworks.com/best-crypto-cpa-networks/>
38. Guidance on project management: ISO 21500:2012 [Електронний ресурс]. – Режим доступу: <https://www.iso.org/standard/50003.html>.

39. Library of PMI Global Standards. [Електронний ресурс]. – Сайт Інституту управління проектами PMI. – <http://www.pmi.org/en/PMBOK-Guideand-Standards/Standards-Library-of-PMI-Global-Standards>
40. Michel Crouhy, Dan Galai, Robert Mark. The Essentials of Risk Management [Навчальний посібник] / McGraw-Hill. . – 2005. – p. 416.
41. Olander, S. Evaluation of stakeholder influence in the implementation of construction projects [Текст] / Stefan Olander, Anne Landin // International Journal of Project Management. – 2005. – №23(4). – p. 321-328.
42. PCI Security Standards Council. (2023. November 25) URL: • <https://www.pcisecuritystandards.org>
43. A Guide On How To Get Payment Card Industry Security Standard For Payment Processing Companies (2023). URL: <https://getpci.com/vidi-sertifikativ/pci-dss>
44. Chen J. Investopedia: Know Your Client (KYC): What It Means, Compliance Requirements (2023. November 01) URL: <https://www.investopedia.com/terms/k/knowyourclient.asp>
45. Kenton W. Investopedia: Anti Money Laundering (AML) Definition: Its History and How It Works? (2023. October 9) URL: <https://www.investopedia.com/terms/a/aml.asp>
46. The International Organization for Standardization 20022. (2023) URL: • <https://www.iso20022.org>
47. Advapay.EU: What is Payment Services Directive 2 (PSD2)? (2023. January 2021) URL: <https://advapay.eu/what-is-payment-services-directive-2-psd2/>
48. Barlow M. Evolving Architectures of FinTech. // O'Reilly Media, Inc. 2016
49. Chao W. Systems Architecture of Mobile Payment FinTech Cloud Applications and Services IoT System: Structure-Behavior Coalescence Method for Model Singularity // Createspace Independent Publishing Platform 29 July 2018, 128 pages.

50. Snyder K. Forbes Advisor: Business. Payment Gateway vs. Payment Processor (2023. January 6) URL: <https://www.forbes.com/advisor/business/payment-gateway-vs-payment-processor/>
51. Martin R.C. Agile Software Development, Principles, Patterns, and Practices / R.C. Martin. – London: Pearson, 2002. – 552 p.
52. What is Header Bidding or how to earn more if you're a gaming publisher? <https://palai.media/en/article/what-is-header-bidding-or-how-to-earn-more-if-youre-a-gambling-publisher/>

Лістинг Front-end частини

Main.html

```
<!DOCTYPE html>

<html lang="en">

<head>

  <meta charset="UTF-8">

  <meta http-equiv="X-UA-Compatible" content="IE=edge">

  <meta name="viewport" content="width=device-width, initial-scale=1.0">

  <link href="style/main.css" rel="stylesheet" type="text/css">

  <title>Document</title>

</head>

<body>

  <header class="header__main">

    <div class="header__logo">

    </div>

    <div class="header__link">

      <div class="link">

        <a id="link1" onclick="nextPage(1)">

          <span>Головна</span></a>

        </div>

      <div class="link">

        <a id="link2" onclick="nextPage(2)">

          Офери</a>

        </div>

      </div>

    </div>

  </body>

</html>
```

```
</div>
<div class="link">
    
    <a id="link3" onclick="nextPage(3)">Мої офери</a>
</div>
```

```
</div>
<div class="header__rifht-info">
    <div class="money">
        <p class="balanceUser">7434 $</p><button>+</button>
    </div>
    <div class="day-night">
        
    </div>
    <div class="circle">
        
    </div>
    <div class="circle">
        
    </div>
    <div class="circle">
        
    </div>
    <div class="circle">
        <a href=""></a>
    </div>
</div>
```

```
</div>
```

```
</header>
```

```
<header class="header">
```

```
<div class="header__logo">
    
</div>
</header>
<div class="container">
    <div class="main">
        <div class="container__left-main">
            <div class="left-main__upper">
                <div class="info">
                    <div class="img-info">
                        
                    </div>
                    <div class="text">
                        <p>Хіти</p>
                        <h3 class="info-hit">3027</h3>
                    </div>
                </div>
            </div>
            <div class="info">
                <div class="img-info">
                    
                </div>
                <div class="text">
                    <p>Ліди</p>
                    <h3 class="info-lid">48</h3>
                </div>
            </div>
            <div class="info">
                <div class="img-info">
                    
                </div>
```

```
<div class="text">
    <p>CR</p>
    <h3 class="info-cr">5.0 %</h3>
</div>
</div>
<div class="info">
    <div class="img-info">
        
    </div>
    <div class="text">
        <p>Хости</p>
        <h3 class="info-host">1003</h3>
    </div>
</div>
<div class="info">
    <div class="img-info">
        
    </div>
    <div class="text">
        <p>Депозити</p>
        <h3 class="info-dep">22</h3>
    </div>
</div>
<div class="info">
    <div class="img-info">
        
    </div>
    <div class="text">
        <p>Visit to reg%</p>
        <h3 class="info-reg">30</h3>
```

```
        </div>
    </div>
    <div class="balance">
        <div class="balance__title">
            
            Баланс
        </div>
        <h1 class="balanceUser">$ 7434</h1>
        <button
class="purple"><b>Поповнити</b></button>
    </div>
</div>
<div class="left-main__bottom">
    <canvas id="myChart"></canvas>
</div>

</div>
<div class="container__right-main">
    <div class="advertising">

        <div class="user-img">
            <img class="user-avatar" src="" alt="фотка
юзера">
        </div>
        <div class="user-name__maneg-name">
            <div class="user-name">

        </div>
        <div class="meneg">
            <p>Менеджер: Рожков Ярослав</p>
        </div>
    </div>
</div>
```

</div>

</div>

<div class="advertising-info">

<p>Мої офери</p>

</div>

</div>

</div>

<div id="my" class="container__list-ofer">

<div id="my-ofer-list" class="ofer-list">

<div class="ofer-list__header">

<p onclick="myId()" class="ofer">Офер</p>

<p onclick="myCr()" class="cr">CR</p>

<p onclick="myhits()" class="hits">Хіти</p>

<p onclick="myhosts()" class="hosts">Хости</p>

<p>Ліди</p>

<p>Депозити</p>

<p>Фінанси</p>

</div>

</div>

</div>

<div id="all" class="container__list-ofer">

<div id="ofer-list-all" class="ofer-list">

<div class="ofer-list__header">

<p onclick="allId()">ID</p>

<p onclick="allName()">Назва</p>

<p onclick="allCost()">виплати, \$</p>

<p onclick="allGoal()">ціль</p>

<p onclick="allCountry()">країни</p>

<p onclick="allLanguage()">мова</p>

<p>Докладніше</p>

</div>

</div>

</div>

<div class="window-big-list">

<div class="window-big-list__title">

<div class="name-offer-title">

<h3 id="name-title">Назва</h3>

<p>Android</p>

</div>

</div>

<div class="block-add">

<h3>Tracking link</h3><input value="https://offers.neogara.com/#/register?ref=7u3dg8">

</div>

<div class="block-add">

<h3>Обрати джерело</h3><input>

</div>

<div class="block-add">

<h3>Скрипт</h3><input value="1wefwefew_44C133e">

</div>

<div class="block-add">

```
        <h3>Назва ПК</h3><input>
    </div>
    <div class="block-add">
        <h3>Facebook ID (-s)</h3><input>
    </div>
    <div class="block-add">
        <h3>Заголовок</h3><input>
    </div>
    <div class="block-add">
        <h3>Повідомлення</h3><input>
    </div>
    <button onclick="closeBigWindow()" class="button-add-big-
window">Зберегти</button>
    <button onclick="closeBigWindow()" class="buttonclose-
window">Закрити</button>
    </div>
</div>
<div class="login__container">
    <div class="login">
        <h1>Авторизація</h1>
        <div class="container__input">
            <h2>E-mail</h2>
            <input class="email" placeholder="Введіть e-mail">
        </div>
        <div class="container__input">
            <h2>Пароль</h2>
            <input id="password" class="email" placeholder="Введіть
пароль">
        </div>
        <button onclick="login()" class="purple"><b>Увійти</b></button>
    </div>
</div>
</div>
</body>
```

```
<script src="https://cdn.jsdelivr.net/npm/chart.js"></script>
```

```
<script src="script/script.js"></script>
```

```
<script>
```

```
  const labels = [
```

```
    '10.05',
```

```
    '14.05',
```

```
    '18.05',
```

```
    '22.05',
```

```
    '26.05',
```

```
    '30.05',
```

```
    '03.06',
```

```
  ];
```

```
  const data = {
```

```
    labels: labels,
```

```
    datasets: [
```

```
      {
```

```
        label: 'Хіти',
```

```
        backgroundColor: 'rgb(186, 191, 206)',
```

```
        borderColor: 'rgb(186, 191, 206)',
```

```
        data: [2000, 1500, 1400, 1700, 2000, 1300, 2000],
```

```
      },
```

```
      {
```

```
        label: 'Ліди',
```

```
        backgroundColor: 'rgb(155, 166, 235)',
```

```
        borderColor: 'rgb(155, 166, 235)',
```

```
        data: [1200, 700, 400, 450, 540, 400, 700],
```

```
      },
```

```
      {
```

```
        label: 'Депозити',
```

```
        backgroundColor: 'rgb(182, 255, 170)',
```

```
        borderColor: 'rgb(182, 255, 170)',
```

```
        data: [1100, 600, 300, 350, 240, 300, 600],
```

```
        },
        {
            label: 'Visit to reg',
            backgroundColor: 'rgb(255, 170, 170)',
            borderColor: 'rgb(255, 170, 170)',
            data: [1050, 550, 250, 300, 200, 250, 550],
        },
    ]
};

const config = {
    type: 'line',
    data: data,
    options: {}
};

</script>
<script>
    const myChart = new Chart(
        document.getElementById('myChart'),
        config
    );
</script>
</html>
```

Лістинг Back-end частини

Script.js

```
//перевірка логіну та паролю
function login() {
  const email = document.querySelector(".email").value
  const password = document.querySelector("#password").value

  for (let i = 0; i < user.length; i++) {
    if (user[i].email == email && user[i].password == password) {
      idUser = i

      userImgName(idUser)
      randomOffer(idUser)
      balanceUser()

      const headerLogin = document.querySelector(".header")
      headerLogin.remove()

      const divLogin = document.querySelector(".login__container")
      divLogin.remove()

      const mainDiv = document.querySelector(".main")
      mainDiv.style.display = "flex"

      const headerMain = document.querySelector(".header__main")
      headerMain.style.display = "flex"

    }
  }
  login()

  //рендер аватарки юзера і імені
  function userImgName(id) {
    const userAvatar = document.querySelector(".user-avatar")
    userAvatar.setAttribute("src", user[id].img)

    const userName = document.querySelector(".user-name")
    userName.innerHTML = user[id].name

  }

  function nextPage(idPage) {
```

```

const mainPage = document.querySelector(".main")
const oferList = document.querySelector("#all")
const myOferList = document.querySelector("#my")

if (idPage == 1) {
    mainPage.style.display = "flex"
    oferList.style.display = "none"
    myOferList.style.display = "none"

    const link1 = document.querySelector("#link1")
    const link2 = document.querySelector("#link2")
    const link3 = document.querySelector("#link3")

    link1.innerHTML = "<span>Головна</span>"
    link2.innerHTML = "Офери"
    link3.innerHTML = "Мої офери"

}
else if (idPage == 2) {
    mainPage.style.display = "none"
    oferList.style.display = "flex"
    myOferList.style.display = "none"

    const link1 = document.querySelector("#link1")
    const link2 = document.querySelector("#link2")
    const link3 = document.querySelector("#link3")

    link1.innerHTML = "Головна"
    link2.innerHTML = "<span>Офери</span>"
    link3.innerHTML = "Мої офери"

    offerlistAll(offers)

}
else if (idPage == 3) {
    mainPage.style.display = "none"
    oferList.style.display = "none"
    myOferList.style.display = "flex"

    const link1 = document.querySelector("#link1")
    const link2 = document.querySelector("#link2")
    const link3 = document.querySelector("#link3")

    link1.innerHTML = "Головна"
    link2.innerHTML = "Офери"
    link3.innerHTML = "<span>Мої офери</span>"

```

```

        offerlistMy(myOffers)

    }
}

function offerlistAll(offers) {

    var oferList = document.querySelector("#ofer-list-all")

    const blockRemuve = document.querySelectorAll("#ofer-list__block-ALL")
    if (blockRemuve) {
        for (let i = 0; i < blockRemuve.length; i++) {
            blockRemuve[i].remove()
        }
    }

    for (let i = 0; i < offers.length; i++) {
        const offerBlock = document.createElement("div")
        offerBlock.className = "ofer-list__block"
        offerBlock.setAttribute("id", "ofer-list__block-ALL")

        const p1 = document.createElement("p")
        p1.className = "idlist"
        p1.innerHTML = offers[i].id

        const p2 = document.createElement("p")
        p2.innerHTML = offers[i].name

        const p3 = document.createElement("p")
        p3.className = "cost"
        p3.innerHTML = offers[i].cost

        const p4 = document.createElement("p")
        p4.innerHTML = offers[i].goal

        const p5 = document.createElement("p")
        p5.innerHTML = offers[i].country

        const p6 = document.createElement("p")
        p6.innerHTML = offers[i].language

        const button = document.createElement("button")
        button.setAttribute("onclick", `ask(${offers[i].id})`)
        button.className = "ofer-button"
        button.innerHTML = "Запросити"
    }
}

```

```

        offerBlock.append(p1)
        offerBlock.append(p2)
        offerBlock.append(p3)
        offerBlock.append(p4)
        offerBlock.append(p5)
        offerBlock.append(p6)
        offerBlock.append(button)

        oferList.append(offerBlock)
    }

}

function offerlistMy(offers) {
    var oferList = document.querySelector("#my-ofer-list")

    const blockRemuve = document.querySelectorAll("#ofer-list__block-MY")
    if (blockRemuve) {
        for (let i = 0; i < blockRemuve.length; i++) {
            blockRemuve[i].remove()
        }
    }

    for (let i = 0; i < offers.length; i++) {
        const offerBlock = document.createElement("div")
        offerBlock.className = "ofer-list__block"
        offerBlock.setAttribute("id", "ofer-list__block-MY")

        const p1 = document.createElement("p")
        p1.className = "ofer"
        p1.innerHTML = `(${offers[i].id}) ${offers[i].name} `

        const p2 = document.createElement("p")
        p2.className = "cr"
        p2.innerHTML = offers[i].cr + "%"

        const p3 = document.createElement("p")
        p3.className = "hits"
        p3.innerHTML = offers[i].hit

        const p4 = document.createElement("p")
        p4.className = "hosts"
        p4.innerHTML = offers[i].host

        const div1 = document.createElement("div")

```

```

div1.className = "color"
const h11 = document.createElement("h3")
h11.innerHTML = offers[i].lid[0]
const h12 = document.createElement("h3")
h12.innerHTML = offers[i].lid[1]
const h13 = document.createElement("h3")
h13.innerHTML = offers[i].lid[2]
div1.append(h11)
div1.append(h12)
div1.append(h13)

const div2 = document.createElement("div")
div2.className = "color"
const h21 = document.createElement("h3")
h21.innerHTML = offers[i].lid[0]
const h22 = document.createElement("h3")
h22.innerHTML = offers[i].dep[1]
const h23 = document.createElement("h3")
h23.innerHTML = offers[i].dep[2]
div2.append(h21)
div2.append(h22)
div2.append(h23)

const div3 = document.createElement("div")
div3.className = "color"
const h31 = document.createElement("h3")
h31.innerHTML = offers[i].finans[0]
const h32 = document.createElement("h3")
h32.innerHTML = offers[i].finans[1]
const h33 = document.createElement("h3")
h33.innerHTML = offers[i].finans[2]
div3.append(h31)
div3.append(h32)
div3.append(h33)

offerBlock.append(p1)
offerBlock.append(p2)
offerBlock.append(p3)
offerBlock.append(p4)
offerBlock.append(div1)
offerBlock.append(div2)
offerBlock.append(div3)

oferList.append(offerBlock)
}

```

```

}

var aId = 1
function allId() {

```

```

    if (aId == 0) {
        offers.sort(function (a, b) {
            return a.id - b.id
        })
        aId++
        offerlistAll(offers)
    } else {
        offers.sort(function (a, b) {
            return b.id - a.id
        })
        aId--
        offerlistAll(offers)
    }
}

var aNa = 0
function allName() {
    if (aNa == 0) {
        offers.sort((a, b) => a.name > b.name ? 1 : -1)
        aNa++
        offerlistAll(offers)
    } else {
        offers.sort((a, b) => a.name < b.name ? 1 : -1)
        aNa--
        offerlistAll(offers)
    }
}

var aCo = 1
function allCost() {
    if (aCo == 0) {
        offers.sort(function (a, b) {
            return a.cost - b.cost
        })
        aCo++
        offerlistAll(offers)
    } else {
        offers.sort(function (a, b) {
            return b.cost - a.cost
        })
        aCo--
        offerlistAll(offers)
    }
}

```

```

    }
}

var aGo = 0
function allGoal() {
    if (aGo == 0) {

        offers.sort((a, b) => a.goal > b.goal ? 1 : -1)
        aGo++
        offerlistAll(offers)
    } else {
        offers.sort((a, b) => a.goal < b.goal ? 1 : -1)
        aGo--
        offerlistAll(offers)
    }
}

var aCountry = 0
function allCountry() {
    if (aCountry == 0) {

        offers.sort((a, b) => a.country > b.country ? 1 : -1)
        aCountry++
        offerlistAll(offers)
    } else {
        offers.sort((a, b) => a.country < b.country ? 1 : -1)
        aCountry--
        offerlistAll(offers)
    }
}

var aLa = 0
function allLanguage() {
    if (aLa == 0) {

        offers.sort((a, b) => a.language > b.language ? 1 : -1)
        aLa++
        offerlistAll(offers)
    } else {
        offers.sort((a, b) => a.language < b.language ? 1 : -1)
        aLa--
        offerlistAll(offers)
    }
}

function ask(idOffer) {

    const windowSearch = document.querySelector(".window")
    if (windowSearch) windowSearch.remove()
}

```

```
var selectedOffer

for (let i = 0; i < offers.length; i++) {
    if (offers[i].id == idOffer) selectedOffer = offers[i]
}

const container = document.querySelector(".container")

const window = document.createElement("div")
window.className = "window"

const title = document.createElement("h1")
title.innerHTML = "Підтвердження оферу"

const divImg = document.createElement("div")
divImg.className = "divImg"

const imgName = document.createElement("div")
imgName.className = "imgName"

const img = document.createElement("img")
img.setAttribute("src", "img/ofer/img.png")

const name = document.createElement("h2")
name.innerHTML = selectedOffer.name

const country = document.createElement("p")
country.innerHTML = selectedOffer.country

imgName.append(img)
imgName.append(name)
divImg.append(imgName)
divImg.append(country)

const info = document.createElement("p")
info.className = "info"
info.innerHTML = selectedOffer.info

const divCost = document.createElement("div")
divCost.className = "divCost"
const infoCost = document.createElement("h3")
infoCost.innerHTML = `Коштів буде заморожено:`
const cost = document.createElement("h3")
cost.innerHTML = `${selectedOffer.cost}`

divCost.append(infoCost)
divCost.append(cost)

const balance = document.createElement("h3")
balance.className = "balanceDom"
balance.innerHTML = `Баланс: ${user[idUser].balance}``
```

```

const buttonAdd = document.createElement("button")
buttonAdd.setAttribute("onclick", `openBigInfo(${idOffer})`)
buttonAdd.className = "buttonAdd"
buttonAdd.innerHTML = `Підтвердити`

const buttonclose = document.createElement("button")
buttonclose.setAttribute("onclick", "closeWindow()")
buttonclose.className = "buttonclose"
buttonclose.innerHTML = `Закрити`

window.append(title)
window.append(divImg)
window.append(info)
window.append(divCost)
window.append(balance)
window.append(buttonAdd)
window.append(buttonclose)

container.append(window)
}
//Закрити вікно
function closeWindow() {
  const windowSearch = document.querySelector(".window")
  if (windowSearch) windowSearch.remove()
}

//кнопка відправки
function buttonDone(idOffer) {

  const buttonAddSearch = document.querySelector(".buttonAdd")
  buttonAddSearch.innerHTML = "Відправлено ✓ "

  buttonAddSearch.setAttribute("class", "done")
  buttonAddSearch.setAttribute("onclick", "")

  const buttonclose = document.querySelector(".buttonclose")
  buttonclose.remove()

  setTimeout(() => {
    closeWindow()
  }, 4000);

  for (let i = 0; i < offers.length; i++) {
    if (offers[i].id == idOffer) user[idUser].balance -= offers[i].cost
    balanceUser()
  }

  setTimeout(() => {
    randomOffer(idOffer)
  }

```

```

        balanceUser()
    }, 2000);

}

//Рандом наших офферов)
function randomOffer(idOffer) {

    var newOffer

    for (let i = 0; i < offers.length; i++) {
        if (offers[i].id == idOffer) newOffer = offers[i]
    }

    const crNewRandom = 600 + getRandomInt(2000)
    const hitRandom = crNewRandom - getRandomInt(500)

    const lidMath = []

    lidMath[2] = hitRandom - getRandomInt(200)
    lidMath[0] = (lidMath[2] / 2) - getRandomInt(10)
    lidMath[1] = (lidMath[0] / 2) - getRandomInt(20)

    const depMath = []

    depMath[2] = lidMath[2] / 2 - getRandomInt(10)
    depMath[1] = lidMath[1] / 2 - getRandomInt(10)
    depMath[0] = lidMath[0] / 2 - getRandomInt(10)

    const finansMath = []

    finansMath[2] = lidMath[2] * (20 + getRandomInt(50))
    finansMath[1] = lidMath[1] * (5 + getRandomInt(30))
    finansMath[0] = lidMath[0] * (5 + getRandomInt(30))

    const crMath = rounded(((lidMath[2] / crNewRandom) * 100) / (3 + getRandomInt(6)))

    myOffers[myOffers.length] = {
        id: newOffer.id,
        name: newOffer.name,
        cr: crMath,
    }
}

```

```

        hit: crNewRandom,
        host: hitRandom,
        lid: lidMath,
        dep: depMath,
        finans: finansMath,
    }

myOffers = myOffers.reduce((o, i) => {
    if (!o.find(v => v.id == i.id)) {
        o.push(i);
    }
    return o;
}, []);

user[idUser].balance += finansMath[2]

myOffersMainMenu()

```

```

var hit = 0, lid = 0, cr = 0, host = 0, dep = 0, proc = 0

```

```

for (let i = 0; i < myOffers.length; i++) {

    const divhit = document.querySelector(".info-hit")
    const divlid = document.querySelector(".info-lid")
    const divcr = document.querySelector(".info-cr")
    const divhost = document.querySelector(".info-host")
    const divdep = document.querySelector(".info-dep")
    const divproc = document.querySelector(".info-reg")

    hit += myOffers[i].hit
    lid += myOffers[i].lid[2]
    cr += myOffers[i].cr
    host += myOffers[i].host
    dep += myOffers[i].dep[2]
    proc = 10 + getRandomInt(30)

    divhit.innerHTML = hit
    divlid.innerHTML = lid
    divcr.innerHTML = cr
    divhost.innerHTML = host
    divdep.innerHTML = dep

```

```

        divproc.innerHTML = proc
    }

}

function getRandomInt(max) {
    return Math.floor(Math.random() * max);
}

function rounded(number) {
    return +number.toFixed(2);
}
//Перевірка балансу
function returnBalanceUser() {
    balanceUser()
}
//Оновлення балансу
function balanceUser() {
    const balanceUser = document.querySelectorAll(".balanceUser")
    for (let i = 0; i < balanceUser.length; i++) {
        balanceUser[i].innerHTML = `${user[idUser].balance}$`
    }
    setTimeout(() => {
        returnBalanceUser()
    }, 10000);
}

//Сортування по ID для мого списку
var mId = 1
function myId() {
    if (mId == 0) {
        myOffers.sort(function (a, b) {
            return a.id - b.id
        })
        mId++
        offerlistMy(myOffers)
    } else {
        myOffers.sort(function (a, b) {
            return b.id - a.id
        })
        mId--
        offerlistMy(myOffers)
    }
}

}
//Сортування по сг для мого списку

```

```

var mCr = 1
function myCr() {

    if (mCr == 0) {

        myOffers.sort(function (a, b) {
            return a.cr - b.cr
        })
        mCr++
        offerlistMy(myOffers)
    } else {
        myOffers.sort(function (a, b) {
            return b.cr - a.cr
        })
        mCr--
        offerlistMy(myOffers)
    }

}

//Сортування по хостам для мого списку
var mHO = 1
function myhosts() {

    if (mHO == 0) {

        myOffers.sort(function (a, b) {
            return a.host - b.host
        })
        mHO++
        offerlistMy(myOffers)
    } else {
        myOffers.sort(function (a, b) {
            return b.host - a.host
        })
        mHO--
        offerlistMy(myOffers)
    }

}

//Сортування по хітам для мого списку
var mHi = 1
function myhits() {

    if (mHi == 0) {

        myOffers.sort(function (a, b) {
            return a.hit - b.hit
        })
        mHi++
    }
}

```

```

        offerlistMy(myOffers)
    } else {
        myOffers.sort(function (a, b) {
            return b.hit - a.hit
        })
        mHi--
        offerlistMy(myOffers)
    }

}
//оновелння списку моїх оферів на головній
function myOffersMainMenu() {
    const advertisingInfo = document.querySelector(".advertising-info")

    const listInfoSearch = document.querySelectorAll(".list-info")

    if (listInfoSearch) {
        for (let i = 0; i < listInfoSearch.length; i++)listInfoSearch[i].remove()

    }

    if (myOffers) {
        for (let i = 0; i < myOffers.length; i++) {
            const listInfocreate = document.createElement("div")
            listInfocreate.className = "list-info"
            if (i % 2 === 0) {
                const block1 = document.createElement("div")
                block1.className = "block"

                const blockImg1 = document.createElement("div")
                blockImg1.className = "block-img"

                const img1 = document.createElement("img")
                img1.setAttribute("src", "img/ofer/ofer.png")
                img1.setAttribute("alt", "иконка")

                blockImg1.append(img1)

                const text1 = document.createElement("div")
                text1.className = "text"

                const text1h4 = document.createElement("h4")
                text1h4.innerHTML = myOffers[i].name

                const text1p = document.createElement("p")
                text1p.className = "text"
                text1p.innerHTML = "Крипта"
            }
        }
    }
}

```

```
text1.append(text1h4)
text1.append(text1p)
```

```
block1.append(blockImg1)
block1.append(text1)
```

```
listInfocreate.append(block1)
```

```
if (myOffers[i + 1]) {
  const block2 = document.createElement("div")
  block2.className = "block"

  const blockImg2 = document.createElement("div")
  blockImg2.className = "block-img"

  const img2 = document.createElement("img")
  img2.setAttribute("src", "img/ofer/ofer.png")
  img2.setAttribute("alt", "иконка")

  blockImg2.append(img2)

  const text2 = document.createElement("div")
  text2.className = "text"

  const text2h4 = document.createElement("h4")
  text2h4.innerHTML = myOffers[i + 1].name

  const text2p = document.createElement("p")
  text2p.className = "text"
  text2p.innerHTML = "Крипта"

  text2.append(text2h4)
  text2.append(text2p)

  block2.append(blockImg2)
  block2.append(text2)

  listInfocreate.append(block2)
}
```

```
advertisingInfo.append(listInfocreate)
```

```
}
```

```
}
```

```
}
```

```
}
```

```
function openBigInfo(id) {  
    buttonDone(id)  
    setTimeout(() => {  
        const bigWindow = document.querySelector(".window-big-list")  
        bigWindow.style.display = "flex"  
        const title = document.querySelector("#name-title")  
        for (let i = 0; i < offers.length; i++) {  
            if (offers[i].id == id) {  
                title.innerHTML = offers[i].name  
            }  
        }  
    }, 4000);  
}  
  
function closeBigWindow() {  
    const bigWindow = document.querySelector(".window-big-list")  
    bigWindow.style.display = "none"  
}
```