

НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ
імені ІГОРЯ СІКОРСЬКОГО»

ФАКУЛЬТЕТ ПРИКЛАДНОЇ МАТЕМАТИКИ

Кафедра системного програмування і спеціалізованих комп'ютерних систем

«До захисту допущено»

Завідувач кафедри

_____ Тарасенко В.П.
(підпис) (ініціали, прізвище)

“ ___ ” червня 2019 р.

Дипломний проект

на здобуття ступеня бакалавра

з напрямку підготовки **6.050102 «Комп'ютерна інженерія»**

на тему: Програмна система управління інформаційними потоками
в медійній рекламі

Виконав: студент IV курсу, групи КВ-51
(шифр групи)

Маслов Вадим Ігорович
(прізвище, ім'я, по батькові) (підпис)

Керівник доц. каф. СПСКС, к.т.н Петрашенко А.В
(посада, науковий ступінь, вчене звання, прізвище та ініціали) (підпис)

Консультант з нормоконтролю, доц.каф.СПСКС, к.т.н. Клятченко Я.М.
(назва розділу) (посада, вчене звання, науковий ступінь, прізвище, ініціали) (підпис)

Рецензент _____
(посада, науковий ступінь, вчене звання, науковий ступінь, прізвище та ініціали) (підпис)

Засвідчую, що у цьому дипломному
проекті немає запозичень з праць інших
авторів без відповідних посилань.

Студент _____
(підпис)

Київ – 2019 року

**НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ
імені ІГОРЯ СІКОРСЬКОГО»**

ФАКУЛЬТЕТ ПРИКЛАДНОЇ МАТЕМАТИКИ

Кафедра системного програмування і спеціалізованих комп'ютерних систем

Рівень вищої освіти – перший (бакалаврський)

Напрямок підготовки 6.050102 «Комп'ютерна інженерія»

ЗАТВЕРДЖУЮ

Завідувач кафедри

_____ Тарасенко В.П.
(підпис) (ініціали, прізвище)

«__» червня 2019 р.

ЗАВДАННЯ

на дипломний проект студента

Маслова Вадима Ігоровича

(прізвище, ім'я, по батькові)

1. Тема проекту

Програмна система управління інформаційними потоками в медійній рекламі, керівник проекту Петрашенко Андрій Васильович к.т.н. доцент, затверджені наказом по університету від «22» травня 2019 р. №1330-С

2. Термін подання студентом проекту «__» червня 2019 р.

3. Вихідні дані до проекту див. Технічне завдання

4. Зміст пояснювальної записки

- Аналіз існуючих рішень та обґрунтування теми дипломного проекту
- Структура програмних засобів
- Опис розроблених алгоритмів
- Випробування та результати

5. Перелік графічного матеріалу (із зазначенням обов'язкових креслеників, плакатів, презентацій тощо)

- ER діаграма бази даних
- Структурна схема проекту
- Аукціон OpenRTB. Структурна схема
- Фільтрування рекламних кампаній. Схема алгоритму

6. Консультанти розділів проекту*

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв
Нормоконтроль	Клятченко Я.М. доцент		

7. Дата видачі завдання «__» _____ 2019 р.

Календарний план

№ з/п	Назва етапів виконання дипломного проекту	Термін виконання етапів проекту	Примітка
1	Вивчення літератури за тематикою проекту	15.04.2019	
2	Розроблення та узгодження технічного завдання	30.04.2019	
3	Аналіз існуючих рішень	05.05.2019	
4	Підготовка матеріалів першого розділу дипломного проекту	10.05.2019	
5	Підготовка матеріалів другого розділу дипломного проекту	18.05.2019	
6	Підготовка графічної частини дипломного проекту	20.05.2019	
7	Оформлення документації дипломного проекту	25.05.2019	
8	Попередній огляд матеріалів диплому на кафедрі	30.05.2019	

Студент

_____ (підпис)

Маслов В.І

(ініціали, прізвище)

Керівник проекту

_____ (підпис)

Петрашенко А.В

(ініціали, прізвище)

* Консультантом не може бути зазначено керівника дипломного проекту.

АНОТАЦІЯ

Кваліфікаційна робота включає пояснювальну записку (51 сторінку, 17 рисунків, 2 графіки)

Об'єкт розробки – створення комп'ютерної системи автоматизації управління процесом продажу медійної реклами яка дозволяє завантажувати рекламні пропозиції та налаштовувати порядок та правила їх показу.

Комп'ютерна система дозволяє: завантажувати медіа рекламу та забезпечувати їх збереження; створення рекламних кампаній, в основі яких лежать попередньо завантажені рекламні креативи; в автоматичному режимі вибрати оптимальну рекламну пропозицію з запропонованих рекламним запитом. В процесі розробки було використано стандарт OpenRTB версії 2.3. Всі описані сервіси було розроблено мовою програмування Node.js. В якості бази даних використовувалась MySQL.

В ході розробки:

- проведено аналіз стандарту OpenRTB;
- сформульовані вимоги до комп'ютерної системи автоматизації управління продажу медійної реклами;
- розроблена система автоматизації вибору оптимальної пропозиції з запропонованих варіантів рекламного запиту;
- розроблено користувацький додаток для завантаження рекламних банерів та створення рекламних кампаній задля їх просування;
- розроблено веб-сервіс для автоматичного завантаження даних створених через додаток та оперування цими даними для вибору оптимальної пропозиції;
- розроблено програмне забезпечення для створення тестового навантаження на веб-сервіс.

Ключові слова:

КОМП'ЮТЕРНА СИСТЕМА АВТОМАТИЗАЦІЇ ПРОДАЖУ МЕДІЙНОЇ РЕКЛАМИ, OPENRTB, DSP, ADVERTISEMENT CAMPAIGNS, NODE.JS, MYSQL.

ANOTATION

Qualification work includes explanatory note (51 page, 17 images, 2 graphs)

Subject of development is creation of computer system that automates the selling process of advertisements and which handles creation of ad inventory and targeting rules.

Developed computer system allows: to download ad banners and handles its storage; creation of ad campaigns, which are based on previously created ad banners; automatically select optimal ad position from provided ad request. Developed system is based on open-source standard OpenRTB version 2.3. All services were developed using Node.js. MySQL was used as main storage.

During development process:

- OpenRTB version 2.3 was analyzed ;
- formulated requirements for the computer system that automates selling process of advertisements;
- create automatization system for optimal deal selection from provided ad request;
- developed user application for ad inventory and ad campaign creation;
- developed web-service to handle automatic download of user provided data and using them to select an optimal deal;
- Created software to perform load tests of main system

Keywords:

AD EXCHANGE, OPENRTB, DSP, ADVERTISEMENT CAMPAIGNS, NODE.JS, MYSQL.

Поз.	Формат	ПОЗНАЧЕННЯ	НАЙМЕНУВАННЯ	Кількість аркушів	№ прим.	Примітки
			<u>Документація загальна</u>			
			<u>Новорозроблена</u>			
	A4	ІАЛЦ.045430.002 ТЗ	Програмна система управління інформаційними потоками в медійній рекламі	4		
			Технічне завдання			
	A4	ІАЛЦ.045430.003 ТП	Програмна система управління інформаційними потоками в медійній рекламі	2		
			Відомість технічного проекту			
	A4	ІАЛЦ.045430.004 ПЗ	Програмна система управління інформаційними потоками в медійній рекламі	51		
			Пояснювальна записка			

					ІАЛЦ.045430.001 ОА							
Змін.	Арк.	№ докум.	Підпис	Дата	Програмна система управління інформаційними потоками в медійній рекламі Опис альбому			Літ.	Аркуш	Аркушів		
Розробив		Маслов В.І.							1	3		
Перевірив		Петрашенко А.В.						НТУУ «КПІ ім. І.Сікорського» Кафедра СПіСКС Група КВ-51				
Консульт.												
Н. контроль		Клятченко Я.М.										
Зав. каф.		Тарасенко В.П.										

Поз.	Формат	ПОЗНАЧЕННЯ	НАЙМЕНУВАННЯ	Кількість аркушів	№ прим.	Примітки
	A4	ІАЛЦ.045430.005 Д1	Програмна система управління інформаційними потоками в медійній рекламі	1		
			ER діаграма бази даних			
	A4	ІАЛЦ.045430.006 Д2	Програмна система управління інформаційними потоками в медійній рекламі	1		
			Структурна схема проекту			
	A4	ІАЛЦ.045430.007 Д3	Програмна система управління інформаційними потоками в медійній рекламі	1		
			Структурна схема аукціону			
	A4	ІАЛЦ.045430.008 Д4	Програмна система управління інформаційними потоками в медійній рекламі	1		
			Блок схема фільтрування рекламних кампаній			
ІАЛЦ.045430.001 ОА						
Змін.	Арк.	№ докум.	Підпис	Дата	Арк.	
					2	

ЗМІСТ

1. НАЙМЕНУВАННЯ ТА ГАЛУЗЬ РОЗРОБКИ.....	2
2. ПІДСТАВА ДЛЯ РОЗРОБКИ.....	2
3. ЦІЛЬ І ПРИЗНАЧЕННЯ РОБОТИ.....	2
4. ДЖЕРЕЛА РОЗРОБКИ.....	2
5. ТЕХНІЧНІ ВИМОГИ.....	2
5.1. Вимоги до програмного продукту, що розробляється.....	2
5.2. Вимоги до апаратного забезпечення.....	3
5.3. Вимоги до програмного та апаратного забезпечення користувача.....	3
6. ЕТАПИ РОЗРОБКИ.....	4



					ІАЛЦ.045430.002 ТЗ					
Зм	Лист	№ докум.	Підп.	Дата	Програмна система управління інформаційними потоками у галузі медійної реклами Технічне завдання			Лім.	Лист	Листів
Розроб.		Маслов В.І						1	4	
Перев.		Петрашенко А.В								
Н. контр.		Клятченко Я.М						НТУУ «КПІ ім. Ігоря Сікорського», ФПМ, КВ-51		
Затв.		Тарасенко В.П								

1. НАЙМЕНУВАННЯ ТА ГАЛУЗЬ РОЗРОБКИ

Назва розробки: «Програмна система управління інформаційними потоками у галузі медійної реклами».

Галузь застосування: організація прийому та обробки вхідних пропозицій продажу рекламних банерів.

2. ПІДСТАВА ДЛЯ РОЗРОБКИ

Підставою для розробки є завдання на виконання роботи першого (бакалаврського) рівня вищої освіти, затверджене кафедрою системного програмування і спеціалізованих комп'ютерних систем Національного технічного університету України «Київський Політехнічний Інститут імені Ігоря Сікорського».

3. МЕТА І ПРИЗНАЧЕННЯ РОБОТИ

Метою даного проекту є створення веб-сервісу, що робить вибір оптимальної рекламної пропозиції на основі попередньо створених налаштувань таргетування.

4. ДЖЕРЕЛА РОЗРОБКИ

Джерелом інформації є технічна та науково-технічна література, технічна документація, публікації у періодичних виданнях та електронні статті у мережі Інтернет.

5. ТЕХНІЧНІ ВИМОГИ

5.1. Вимоги до програмного продукту, що розробляється

- сумісність веб-сайту з будь яким браузером (Chrome, Firefox ...);

					ІАЛЦ.045430.002 ТЗ	Лист
Зм	Лист	№ докум.	Підп.	Дата		2

- можливість створення та видалення рекламних банерів;
- можливість видалення рекламних банерів;
- можливість створення та видалення рекламних кампаній;
- можливість видалення рекламних кампаній;
- можливість налаштування правил таргетування кампаній;
- сумісність веб-сервісу зі стандартом OpenRTB 2.3;

5.2. Вимоги до апаратного забезпечення

- Процесор: 2-ядерний процесор: Intel, AMD;
- Оперативна пам'ять: 2 Гб;
- Наявність доступу до мережі Internet;

5.3. Вимоги до програмного та апаратного забезпечення користувача

- Операційна система: Ubuntu

					ІАЛЦ.045430.002 ТЗ	Лист
Зм	Лист	№ докум.	Підп.	Дата		3

6. ЕТАПИ РОЗРОБКИ

№ з/п	Назва етапів виконання дипломного проекту	Термін виконання етапів
1.	Вивчення літератури за тематикою проекту	15.04.2019
2.	Розроблення та узгодження технічного завдання	30.04.2019
3.	Аналіз існуючих рішень	05.05.2019
4.	Підготовка матеріалів першого розділу дипломного проекту	10.05.2019
5.	Підготовка матеріалів другого розділу дипломного проекту	18.05.2019
6.	Підготовка графічної частини дипломного проекту	20.05.2019
7.	Оформлення документації дипломного проекту	25.05.2019
8.	Попередній огляд матеріалів диплому на кафедрі	30.05.2019

Поз.	Формат	ПОЗНАЧЕННЯ	НАЙМЕНУВАННЯ	Кількість аркушів	№ прим.	Примітки
			<u>Документація загальна</u>			
			<u>Новорозроблена</u>			
	A4	ДП.468300.004 ПЗ	Мікропроцесорна система	51		
			дистанційного моніторингу			
			параметрів обладнання.			
			Пояснювальна записка			
	A4	ІАЛЦ.045430.005 Д1	Програмна система	1		
			управління			
			інформаційними потоками			
			в медійній рекламі			
			ER діаграма бази даних			
	A4	ІАЛЦ.045430.006 Д2	Програмна система	1		
			управління			
			інформаційними потоками			
			в медійній рекламі			
			Структурна схема			
			проекту			

					ІАЛЦ.045430.003 ТП			
Змін.	Арк.	№ докум.	Підпис	Дата				
Розробив		Маслов В.І.			Програмна система управління інформаційними потоками в медійній рекламі Відомість технічного проекту	Літ.	Аркуш	Аркушів
Перевірив		Петрашенко А.В.					1	2
Консульт.						НТУУ «КПІ ім. І.Сікорського» Кафедра СПіСКС Група KB-51		
Н. контроль		Клятченко Я.М.						
Зав. каф.		Тарасенко В.П.						

Поз.	Формат	ПОЗНАЧЕННЯ	НАЙМЕНУВАННЯ	Кількість аркушів	№ прим.	Примітки	
	А4	ІАЛЦ.045430.007 ДЗ	Програмна система управління інформаційними потоками в медійній рекламі	1			
			Структурна схема аукціону				
	А4	ІАЛЦ.045430.008 Д4	Програмна система управління інформаційними потоками в медійній рекламі	1			
			Блок схема фільтрування рекламних кампаній				
		Диск CD-ROM	Текст ПЗ. Тексти програм. Графічний матеріал	1			
		ІАЛЦ.468300.003 ТП				Арк.	
Змін.	Арк.	№ докум.	Підпис	Дата	2		

ЗМІСТ

ПЕРЕЛІК СКОРОЧЕНЬ, УМОВНИХ ПОЗНАЧЕНЬ, ТЕРМІНІВ _____	3
ВСТУП _____	5
1. АНАЛІЗ ІСНУЮЧИХ РІШЕНЬ ТА ОБҐРУНТУВАННЯ ТЕМИ ДИПЛОМНОГО ПРОЕКТУ _____	6
2. СТРУКТУРА ПРОГРАМНИХ ЗАСОБІВ _____	20
3. ОПИС РОЗРОБЛЕНИХ АЛГОРИТМІВ _____	31
4. ВИПРОБУВАННЯ ТА РЕЗУЛЬТАТИ _____	44
ВИСНОВКИ _____	47
СПИСОК ВИКОРИСТАНИХ ЛІТЕРАТУРНИХ ДЖЕРЕЛ _____	49

					ІАЛЦ.045430.004 ПЗ			
<i>Зм</i>	<i>Лист</i>	<i>№ докум.</i>	<i>Підп.</i>	<i>Дата</i>				
<i>Розроб.</i>		Маслов В.І			Програмна система управління інформаційними потоками в медійній рекламі <i>Пояснювальна записка</i>	<i>Лім.</i>	<i>Лист</i>	<i>Листів</i>
<i>Перев.</i>		Петрашенко А.В					1	51
						НТУУ «КПІ ім. І. Сікорського», ФІМ, КВ-51		
<i>Н. контр.</i>		Клятченко Я.М.						
<i>Затв.</i>		Тарасенко В.П						

ДОДАТКИ

Додаток 1. Копії графічних матеріалів

- ІАЛЦ.045430.005 Д1. ER діаграма бази даних
- ІАЛЦ. 045430.006 Д2. Структурна схема проекту
- ІАЛЦ. 045430.007 Д3. Аукціон OpenRTB. Структурна схема
- ІАЛЦ. 045430.008 Д4. Блок схема фільтрування рекламних кампаній

					ІАЛЦ.045430.004 ПЗ	Лист
Зм	Лист	№ докум.	Підп.	Дата		2

ПЕРЕЛІК СКОРОЧЕНЬ, УМОВНИХ ПОЗНАЧЕНЬ, ТЕРМІНІВ

Ad Exchange – платформа-посередник між DSP та SSP

CPI (cost per impression) - модель монетизації рекламного трафіку за якого оплачується кожен окремий перегляд користувачем

CPM (cost per mille) - формат оцінювання вартості послуг, за якого ціна виставляється за тисячу одиниць товару, передбаченого моделлю

DSP (Demand-Side Platform) – технологічна система-учасник автоматизованого аукціону, яка намагається придбати покази реклами кінцевим користувачам

Highload – стан ПЗ за якого його навантаження наближається до теоретичного мінімуму можливостей мови програмування, на якій написано додаток

KPI (key performance indicators) – загальне поняття, що описує всі метрики рекламної кампанії

KV (key-value) сховище - вид баз даних, які орієнтовано на високу швидкість читання-запису. На основі таких сховищ зазвичай будують системи кешування “гарячих” даних

ORM (Object-relational mapping) - механізм, що дозволяє описувати несумісні з мовою програмування типи даних (наприклад таблицю бази даних) на основі конструкцій доступних в цій мові програмування для наступного маніпулювання полями сутності за допомогою вбудованих методів

PPC (pay per click) – модель монетизації рекламного трафіку за якого оплачується лише перегляд рекламної позиції

ROI (return of investment) – метрика, що показує ефективність інвестицій і є відношенням між чистим прибутком та інвестиціями

RTB (Real Time Bidding) – технологія закупки медійної реклами за допомогою програмованих, автоматичних аукціонів, що відбувається у реальному часі

					ІАЛЦ.045430.004 ПЗ	Лист
Зм	Лист	№ докум.	Підп.	Дата		3

SSP (Supply-Side Platform) – технологічна система-учасник автоматизованого аукціону, яка намагається продати покази реклами рекламодавцям

Рекламна конверсія – відношення загального числа користувачів до користувачів що виконали дію що, передбачена моделлю монетизації (перехід за посиланням, встановлення додатку, тощо)

					ІАЛІЦ.045430.004 ПЗ	Лист
Зм	Лист	№ докум.	Підп.	Дата		4

ВСТУП

Задача що стоїть перед RTB включає в себе велику кількість розрізаних функціональних компонентів. Принцип його функціонування вимагає від розроблених додатків зовсім інших навантажувальних можливостей, аніж у звичних веб-сервісах, оскільки технологія працює з зовсім іншими піковими навантаженнями, що при досить низькому допустимому часу відповіді диктує зовсім інші правила написання ПЗ.

Специфіка технології вимагає відповіді на запити у реальному часі і, оскільки, рекламні запити приходять від кожного користувача, то настає так званий Highload, тобто упирання ПЗ в ліміт можливостей тієї мови програмування, на якій він був написаний а також апаратних можливостей системи. Для цього використовуються різноманітні техніки при проектуванні мови програмування та ПЗ, що дозволяє покращити максимальні можливості мови.

В даному дипломному проєкті було розглянуто функціональну частину технології під назвою DSP, що надасть можливість завантажувати рекламні банери та налаштовувати правила їх таргетування до кінцевого користувача, а також виступати як одна зі сторін процесу автоматичного аукціону.

					ІАЛЦ.045430.004 ПЗ	Лист
Зм	Лист	№ докум.	Підп.	Дата		5

1. АНАЛІЗ ІСНУЮЧИХ РІШЕНЬ ТА ОБҐРУНТУВАННЯ ТЕМИ ДИПЛОМНОГО ПРОЕКТУ

Оскільки сама технологія RTB – відносно молода, то ринок представлений досить невеликою кількістю великих компаній. Сам процес автоматичного проведення аукціонів двосторонній, тому представники галузі діляться на дві категорії: DSP та SSP.

Також існують різні допоміжні сервіси, такі як, наприклад:

DMP – сервіс-посередник, що зберігає дані про користувачів, що дозволяє прямим учасникам аукціону не витратити значні кошти на зберігання повної інформації про користувача або доповнення відсутньої інформації за необхідністю

Ad Exchange – сервіси, що не представляють інтереси жодної зі сторін аукціону, а працюють лише у якості посередника між ними, формуючи мережу, з однієї сторони якої будуть знаходитись SSP, а з іншої – DSP поєднуючи їх задля підвищення ефективності показів

Як основний учасник процесу DSP являє собою рекламодавця – самостійну компанію чи мережу, що має певні рекламні матеріали (або рекламний інвентар), який потрібно продати, донести до кінцевого користувача за якомога меншу ціну. Різні DSP по-різному підходять для різних рекламних кампаній, оскільки їхні направленості чи моделі монетизації можуть кардинально впливати на результат аукціону.

З іншої сторони аукціону знаходиться інший учасник аукціону – SSP, що являє собою платформу – веб-сайт, мобільний додаток, тощо. Який надає рекламний трафік, тобто потік користувачів що знаходяться на даній платформі і яким можна показати рекламу. Також SSP може збирати дані про користувачів, для того щоб в майбутньому надавати більш детальну інформацію про користувача – все для того щоб підвищити ефективність показу реклами.

					ІАЛЦ.045430.004 ПЗ	Лист
Зм	Лист	№ докум.	Підп.	Дата		6

Сам термін RTB означає real-time bidding, тобто ставки в реальному часі. В момент завантаження сторінки створюється аукціон, на якому відбуваються ставки за певне рекламне місце на платформі. Аукціон проводиться як аукціон другої ціни, тобто у разі виграшу виплачується не ціна ставки, а певна ціна, що визначається на основі другої найбільшої ставки. Причиною вибору саме аукціону другої ціни можна вважати те, що одні й ті самі користувачі з'являються на сервісах ad exchange. Якби рекламодавці платили рівно стільки скільки б вони ставили (тобто більш звичний аукціон першої ціни), це б не описувало істинні оцінювання вартості показу такому користувачу, вони б просто вручну підлаштовували свої ставки у відповідь на ставки інших гравців на ринку. Така нестабільність в поведінці ставок була розглянута на ринку аукціонів першої ціни, наприклад як на пошукових ринках [1].

Для розуміння цього можна розглянути наступний приклад. Уявимо двох рекламодавців, що розділяють групи таргетування та виставляють наступні ціни за тисячу показів: \$6 та \$8. Мінімальною вартістю ставки в даній ситуації можна вважати \$2. Таким чином протягом аукціону, ставки будуть рости від \$2 до \$6, в цей період обидва рекламодавці намагатимуться перебити ставку одне одного. При ставках в \$6 перший рекламодавець перестане приймати участь в аукціоні. В цей самий момент, другий учасник скине вартість ставки до мінімуму, після чого повернеться перший і після цього замкнеться цикл дій.

Таким чином в даній роботі використовується аукціон другої ціни. При отриманні рекламного запиту, кожен рекламодавець, маючи дані про рекламну позицію, матиме різні уявлення про її вартість. Варто зазначити, що це уявлення є таємницею кожного з учасників, оскільки воно створюється на основі інформації про продаж та загального бюджету. На практиці, такі аукціони відбуваються за ієрархічною системою, оскільки початкове розміщення аукціону може бути на стороні видавницької мережі SSP, яка може передати інформацію про цей аукціон до всіх своїх партнерських мереж ad exchange або навіть рекламних мереж DSP напямую, для того щоб зібрати максимум

					ІАЛЦ.045430.004 ПЗ	Лист
Зм	Лист	№ докум.	Підп.	Дата		7

доступного рекламного інвентарю та обрати найкращу пропозицію з можливих. Таким само чином, всі партнерські мережі також можуть проводити аукціони на своїй стороні для того щоб обрати найкращий варіант серед своїх партнерів.

Аукціони, як інструмент маркетингу, створює конкуренцію між рекламодавцями. Найкращим способом є мотивування їх на те, щоб вони розкривали свої оцінки пропозиції при виставленні ставок. Аукціон другої ціни створює механізм впливу на учасників, що забезпечує розуміння про те, що набагато краще коли вони створюють ставки повністю відповідно до їх чесних оцінок [2]. Для того щоб запевнитися в тому що в даній ситуації чесність допомагає максимізувати прибутковість, розглянемо наступний приклад.

Уявимо показ реклами, що представлений у вигляді векторної функції x , може бути проданий одному з n учасників аукціону. Рекламодавці роблять ставки одночасно. Кожен рекламодавець i ($i \in [1, n]$) має оцінює успішність цього показу як деяку функцію $c_i(x)$. Уявимо що вартість показу для кожного рекламодавця рівна та дорівнює 1. Таємне уявлення кожного учасника про вартість v_i таким чином рівне $c_i(x)$ кожного учасника i . Кожен з учасників знає своє уявлення про вартість показу, проте не знає яке це уявлення у інших учасників. Проте вони можуть мати припущення про це уявлення у інших. Це припущення можна описати розподілом. Зазначимо, що дане уявлення кожного з учасників виконується незалежно від загального розподілу $F(\cdot)$ з її щільністю $f(\cdot)$ на проміжку $[0, +\infty]$. Таким чином вираз $F_V(v) = P(V \leq v)$ означає що ймовірність випадкової змінної V менше або рівне за деяке значення v .

Можна сказати, що учасник 1, для якого це значення дорівнює v_1 , вирішує створити ставку b_1 та для максимізації прибутку від інших учасників 2, ..., n використовує певну стратегію $b(\cdot)$. Уявимо, що рекламодавець i робить найбільшу ставку серед 2, ..., n учасників. Прибутки першого учасника таким чином можна описати у якості π_1

					ІАЛЦ.045430.004 ПЗ	Лист
Зм	Лист	№ докум.	Підп.	Дата		8

$$\pi_1(v_1, b_i, b(\cdot))$$

$$= \begin{cases} v_1 - b_i, & \text{при } b_1 > b_i > \max(b(v_2, \dots, b(v_{i-1}), b(b_{i+1}), \dots, b(v_n))) \\ 0, & \text{при } b_1 < \max(b(v_2, \dots, b(v_{i-1}), b(b_{i+1}), \dots, b(v_n))) \end{cases}$$

Де $b_1 > b_i > \max(b(v_2, \dots, b(v_{i-1}), b(b_{i+1}), \dots, b(v_n))$ це ймовірність

$\int_0^{b_1} dF(x)^{n-1} = \int_0^{b_1} (v_1 - x)(n - 1)f(x)F(x)^{n-2} dx$, таким чином маємо

$$\pi_1(v_1, b_i, b(\cdot)) = \int_0^{b_1} (v_1 - x)(n - 1)f(x)F(x)^{n-2} dx$$

Рекламодавець 1 обирає b_1 таким чином, щоб максимізувати свої прибутки. При $b_1 > v_1$ ми маємо

$$\pi_1(v_1, b_i, b(\cdot))$$

$$= \int_0^{v_1} (v_1 - x)(n - 1)f(x)F(x)^{n-2} dx$$

$$+ \int_{v_1}^{b_1} (v_1 - x)(n - 1)f(x)F(x)^{n-2} dx$$

де другий інтеграл – від’ємний. Таким чином ми можемо очікувати ріст прибутків при $b_1 \rightarrow v_1$. З іншої сторони, при $b_1 < v_1$ другий інтеграл стає додатнім. При $b_1 \rightarrow v_1$ очікувана винагорода зростає на суму

$$\int_{b_1}^{v_1} (v_1 - x)(n - 1)f(x)F(x)^{n-2} dx$$

Таким чином можна сказати що винагорода максимізована за умови, що $b_1 = v_1$. Тому правдивість у відносинах – це домінантна стратегія ведення угод. Домінантною можна вважати таку стратегію, що є кращою для учасника, при цьому стиль ставок інших учасників не важливий.

Проте, на практиці, зазвичай рекламодавець приходить на аукціон з фіксованим бюджетом та може бути задіяний у декількох аукціонах другої ціни за все життя рекламної кампанії. Таким чином повна правдивість може і не бути домінантною стратегією.

Хоч теорія ігор і приносить деяке розуміння поведінок стратегій для всіх учасників аукціону, практичним підходом буде дослідження статистики ринкової ціни та об'ємів продажу

Аукціон за замовчуванням відкритий, тобто не відбувається фільтрація за партнером і в аукціоні може приймати участь кожен бажаючий.

Проте, великі платформи чи мережі можуть створювати РМР (private marketplace або приватний майданчик) в аукціонах яких приймають участь тільки певні підтверджені партнери та в якому торги відбуваються тільки між ними.

Всі існуючі рішення розповсюджуються за певною системою монетизації. Сама по собі модель монетизації означає спосіб за допомогою якого буде визначатися одиниці рекламного товару та спосіб їх оплати.

Першою та найпростішою моделлю монетизації є РРС, що передбачає оплату за кожне натиснення на рекламу.

Від користувача не очікується будь-який вид взаємодії після переходу. Очікується тільки натискання на рекламну пропозицію. В рамках зазначеної моделі не враховується загальний результат чи вплив на кінцевого користувача рекламної пропозиції, рекламодавець платить лише за переходи за посиланням. Натискаючи на рекламу, користувач виявляє певну міру зацікавленості в пропозиції, тому ми можемо сказати, що оплата в рамках даної моделі відбувається за таргетовану комунікацію між рекламодавцем та кінцевим користувачем.

Кожен рекламодавець може визначати точний місячний бюджет та максимальну ціну за перехід.

Переваги даної моделі для рекламодавця:

- Реклама має більше розповсюдження і без рекламних переходів
- Швидка доставка високоякісного, таргетованого інтернет трафіку на веб-сайт

- Можна виміряти метрику ROI, та за допомогою неї зрозуміти які з напрямлень працюють найкраще
- Є контроль за бюджетом
- Можливість вказати ключові слова чи регіон за якими буде відбуватися таргетування

Недоліки даної моделі для рекламодавця:

- Дуже висока вартість та вимагає значний бюджет
- Висока ймовірність шахраювання на переходах

Переваги для видавництва (платформи)

- Можна мати партнерські відносини з більшою кількістю рекламодавців, оскільки метрика ROI досить прозора
- Можна отримати більше інформації про тих хто переглядає рекламу
- Середній рівень ризику

Недоліки для платформи:

- Вимагає високого рівня CTR метрики
- Не всі переходи можуть зарахуватися
- Прибутки важко розрахувати, адже невідомо яка кількість користувачів вирішить перейти за посиланням

Модель монетизації CPI або CPM передбачає оплату за кожну тисячу переглядів рекламної позиції користувачем. Дана модель дуже схожа на вищеописану, з мінімальними відмінностями у вигляді більш дешевої одиниці товару (оскільки в даній ситуації від користувача очікується мінімальна інтеракція) та, власне, методів оцінки правдивості трафіку. Дана модель більшою мірою опирається лише на кількість і не важливо чи користувач перейшов за посиланням або проявив будь-яку іншу форму зацікавленості. Найбільше підходить для показу реклами орієнтованої на бренди. З точки зору

рекламодавця CPM є найкращим вибором для передбачуваного прибутку та результатів.

Зазвичай, модель зазначає гарантовану мінімальну кількість показів і тому ціна формується на основі даного числа. Наприклад, якщо сайт виставляє ціну CPM \$2 та гарантує при цьому 100000 переглядів, то рекламодавцю буде виставлена ціна в \$200.

Переваги даної моделі для рекламодавця:

- Найменша вартість за одиницю, найкращий варіант для тих ситуацій коли у рекламодавця обмежений бюджет і є потреба в передбачуваних витратах
- Дуже просте впровадження – заплатив за 1000 показів та забув
- За умови високих показників переходів є досить дешевим рішенням
- Дуже добре підходить для рекламування бренду і не потребує високих показників продуктивності

Недоліками даної для рекламодавця є:

- Потрібно оплачувати кожен показ, а не кожен індивідуальний
- Головною ціллю є кількість, а не якість
- Дуже висока ймовірність шахраювання на показах

Перевагами для платформи є:

- Досить низький ризик
- Оскільки рахуються тільки перегляди, то такі метрики успішності як CTR не мають великого значення
- Перегляди можна перевірити та порахувати
- Фіксована вартість та потік трафіку

Недоліками для платформи можуть бути:

- Потреба в великій кількості трафіку
- Дешеві рекламні пропозиції
- Навіть якщо перехід відбувся, він не буде оплачений

					ІАЛЦ.045430.004 ПЗ	Лист
Зм	Лист	№ докум.	Підп.	Дата		12

Загалом, описані вище моделі являються найпоширенішими серед усіх, оскільки їх дуже просто реалізувати та впровадити. Проте, оскільки показники продуктивності невисокі, то звідси впливає їх, в середньому, невисока вартість за одиницю. В контексті продажу гарантованими пакетами, модель PPC також може бути поєднана з моделлю CRM, що дозволяє покращити показники та прогнози, оскільки остання модель надає певні гарантії та дозволяє чітко конфігурувати наступні кроки рекламної кампанії.

Також існують моделі монетизації, що очікують більшу взаємодію кінцевого користувача з рекламною пропозицією.

Таким чином модель CPA передбачає, що користувач виконає якусь дію після переходу за рекламним посиланням, наприклад встановлення додатку чи покупка запропонованого товару і оплата буде виконана тільки після того як користувач виконає цю дію. Рекламодавець самостійно обирає яку дію повинен виконати користувач: завантажити файл, зареєструватися, тощо.

Модель CPI є варіацією CPA, за якої оплата відбувається після встановлення користувачем. Вона є ідеальним варіантом для рекламодавців – розробників додатків, оскільки дозволяє сфокусуватися на високоякісному трафіку, що в свою чергу дає хоч і невелику кількість користувачів, проте досить сильно зацікавлену в цьому. CPI дуже популярна в мобільній екосистемі оскільки вона надає ряд значних переваг. Хоч вартість за кожне завантаження і висока, проте витрати будуть значно ефективнішими оскільки рекламодавець платить за те, щоб отримати зацікавленого користувача

Основними перевагами даної моделі для рекламодавця є:

- Усі ризики несе платформа, що публікує рекламу за такою моделлю
- Оплата йде тільки за успішне виконання
- Модель формує зацікавлених користувачів

Основними недоліками даної моделі для рекламодавця є:

- Найвища вартість за одиницю товару

- Найменший рівень конверсії
- Висока ймовірність шахраювання

Основними перевагами для платформи є:

- Важко спрогнозувати прибутки
- Більш високі прибутки через високу базову вартість
- Можливість заробити репутацію за допомогою правильного підбору часу та розташування реклами

Основними недоліками для платформи є

- Перехід веде на іншу сторінку, що унеможлиблює можливість перевірити чи перехід став повноцінною дією. Через це трекінг може бути досить проблематичним
- Переходи не оплачуються
- Безкоштовні покази якщо користувачі ніяк не взаємодіють з рекламою
- Найбільші ризики шахраювання

Середня ціна реклами за такою моделлю на Google AdWords складає близько \$60. Також ціна значно залежить від бізнес моделі та індустрії. Наприклад CPA в правовій індустрії може сягати \$135, а в сфері охорони здоров'я \$126 на платформі Google AdWords.

Моделі монетизації це потужна річ, що дозволяє гнучко налаштувати процес проведення рекламної кампанії, проте кожна ітерація вимагає певні метрики ефективності. Їх називають KPI, ці метрики дозволяють оцінити успішність того чи іншого етапу проведення рекламної кампанії.

Однією з фундаментальних метрик можна вважати відношення конверсії трафіку. Під рекламною конверсією мають на увазі виконану користувачем дію, що передбачена моделлю монетизації. Відношення рахується між загальним числом користувачів та числом сконвертованих користувачів. В загальному

сенсі дана метрика являє собою ККД рекламної кампанії, а також визначити вплив на цільову аудиторію чи платформу загалом.

Також базовими метриками можна вважати ті, що підраховують прохідність та вартість однієї дії. Такими метриками є CTR, CPC, CPA.

CTR (click-through rate) означає відношення між числом переходів по рекламі та загальним числом переглядів, та показує вплив пропозиції на відвідувачів платформи (оскільки висока кількість переглядів не дуже корисна за умови, що за посиланням ніхто не переходить)

CPC (cost per click) – це метрика що показує середню вартість одного переходу за рекламним посиланням, рахується відношенням всіх витрат на загальне число переходів

CPA (cost per action) – метрика, що показує середню вартість певної дії передбаченої моделлю монетизації (реєстрація, завантаження файлу, тощо)

Два останні показники дуже корисно поєднувати з ROI, таким чином можна буде порівнювати витрати з сумами що повертаються з цих витрат.

ROI (return of investment) – одна з фундаментальних метрик, що показує як повертаються вкладені інвестиції, показує ефективність обраної стратегії.

Єдиним способом переконатися, що виконана робота – прибуткова, це підрахувати дану метрику. Рахується вона досить просто – це відношення між чистим доходом та інвестиційними витратами

CAC (customer acquisition cost) – метрика, що показує вартість отримання нового користувача на платформу, є важливою тільки для того щоб була змога перевірити чи вартість отримання не перевищує доходи від користувача за час його перебування на платформі.

CLV (customer lifetime value) – виражає загальну користь що приносить користувач за час його знаходження на платформі у якості покупця. Є важливою тільки для знаходження чи не є даний користувач занадто дорогим та витрати на його заманювання на платформу не перевищують прибуток від його перебування.

Швидкість відтоку – метрика, що показує скільки користувачів відмінило свої замовлення чи не змогли поновити підписку на послуги. Дана метрика має сенс тільки за умови, що послуги надаються на основі послуг з періодичною оплатою за них.

Всі описані метрики являються ситуаційними і не обов'язково повинні бути використані одночасно, кожна з них має свої плюси та мінуси. Обираючи модель монетизації завжди потрібно робити аналіз цільової аудиторії та продукту та підбирати метрики згідно проведеним аналізом.

Так само можна сказати і про моделі монетизації. В загальному, модель описує бажаний результат та способи його досягнути, що означає що в залежності від ситуації можна брати краще від різних моделей та поєднувати їх в нові, такі що найкраще підходять для обраної аудиторії, продукту та формату розповсюдження.

Описані метрики та моделі монетизації об'єднують заради однієї мети – підвищити конверсію трафіку та при цьому зменшити вартість кожного показу. Оптимізації ставок були добре досліджені [3][4][5]. Але проблема в тому, що більшість досліджень було проведено тільки у сфері аукціонів за ключовими словами у контексті контекстуальної реклами[6]. Зазвичай, попередньо виставляючи ключові слова за якими будуть відбуватися ставки (не на рівні окремого перегляду), вартість та обсяги попередньо оцінюються, а тільки потім виконуються операції з оптимізації відповідно до метрик вказаних замовником (рекламодавцем)[7]. Взявши бюджет кампанії в якості верхньої межі, оптимізації продуктивності показу підпадають під категорію задач з оптимізації бюджету[8]. Більше того, фокус[9][10] на створення ставок та оптимізація на широких ключових словах, де мови запитів залучено до зменшення ціни пов'язаних ключових слів. Є навіть пропозиції[11], для спільної оптимізації ставок на рівні ключових слів та розміщення бюджету кампанії в рамках аккаунту контекстної реклами з декількома кампаніями. Наприклад, в деяких роботах[12], було використано Марковський процес прийняття рішень для

роботи онлайн для зміни вартості ставки, де залишковий обсяг аукціону та бюджет слугували в якості станів та налаштування вартості – у якості дій. Пропозиція [13] перераховувати план розміщення ставок протягом часу життя кампанії, де вартість ставки на кожне ключове слово визначається в кожен дискретний момент часу враховуючи ринкову конкуренцію та CTR на різних рекламних позиціях. Проте, жоден з запропонованих варіантів не розглядає обробку вартості на рівні окремого перегляду, всі ставки співвідносяться з ключовими словами з рідким включенням факторів окремого перегляду, особливо в ситуації рекламодавців та їх представництв. До того ж, в оптимізації контекстної реклами, пошукові системи грають дві ролі: встановлюють ставки на ключові слова та створюють аукціони. Цільові функції можна відхилити на користь збільшення загального прибутку пошукової системи[14] на відміну від загальної продуктивності кожної окремої кампанії рекламодавця.

Оптимізація для функціонування в контексті RTB фундаментально відрізняється. По-перше, ставки не визначаються попередньо створеними ключовими словами[15], а визначаються на рівні кожного окремого показу. Хоч загалом, рекламодавці мають визначити правила таргетування, їм потрібно оцінити вартість кожного окремого показу, який буде виставлятися на аукціоні в реальному часі та створювати нову ціну ставки для кожного такого аукціону. По-друге, у RTB, в основному використовується модель CPM. Виграш у аукціоні на показ, напряду впливає на вартість, не зважаючи на той факт, що переходи за рекламою та їх конверсія може бути оптимізована рекламодавцями. Таким чином, залежність мір ефективності, таких як модель монетизації та підбір метрик, мають вивчатися разом. Було запропоновано алгоритм[16], що визначає розподіл ставок в залежності від повної чи неповної інформації на аукціонах. В іншому запропонованому алгоритмі[17], вартість кожної ставки може контролюватися видавництвом в реальному часі та його ціль – це максимізувати дохід видавництва. Також було досліджені інші проблеми RTB.

Наприклад, дослідження в області швидкості вичерпання бюджету кампанії[18], тобто рівномірного його використання.

Протягом всієї історії, з самого початку людства графічні та інші форми зображення тих речей, що представляли інтерес людини що їх виражала були тісно з самими людськими інтересами, проте з початком розвитку торгівельної справи ці вирази отримали нову функцію – донесення до широких мас інформації про певний набір товарів чи послуг, що надає інша людина.

З часом розміри торгівельної справи розростаються та збільшується аудиторія, що користується послугами торговців. Зі зростанням аудиторії потрібно знаходити нові, простіші та кращі способи донесення інформації до цих мас, наприклад глашатаї в різних частинах міста. З розвитком типографії значно подешевшали нові способи розповсюдження інформації у вигляді памфлетів чи рекламних листівок. Після появи інтернету та початку глобалізації з'явився новий спосіб донесення рекламної інформації до ще більших мас людей – реклама на веб сторінках.

З моменту появи першого банеру рекламні відносини відбувалися за допомогою усних чи писемних домовленостей, даний формат домовленостей потребує безпосереднього втручання людини на кожному етапі її впровадження на кінцевій платформі. Проте, темпи росту інтернету створюють все більшу кількість рекламних місць і з часом посередники, що проводили такого роду домовленості між сторонами перестали справлятися з напливом навантаження. Дана проблема логічним завершенням попереднього витку історії реклами. Наступним кроком домовленостей повинен стати більш продуктивний підхід до проведення цих угод. Використання програмованих угод дозволяє вирішити досить велику кількість проблем старого підходу, а саме – скоротити кількість проміжних ланок між зацікавленими сторонами до однієї, тобто розміщення точки входу скрипту, що виконує автоматизацію, підняти рівень релевантності рекламної пропозиції користувачу, оскільки таргетування виконується щоразу як скрипт звертається до серверу аукціону, спростити та покращити збір метрик

і перевірку на шахрайство, оскільки в скрипт окрім реклами можна додати допоміжну логіку відслідковування, що надасть додаткові можливості.

Програмний продукт, що дозволяє один раз завантаживши рекламний інвентар та налаштувавши правила таргетування та бюджету в автоматичному порядку переглядати всі можливі пропозиції та обирати найбільш підходящі кампанії без людського втручання являє собою значне покращення існуючих раніше процесів, оскільки майже повністю ліквідується людський фактор та збільшується конверсія рекламних пропозицій.

					ІАЛЦ.045430.004 ПЗ	Лист
Зм	Лист	№ докум.	Підп.	Дата		19

2. СТРУКТУРА ПРОГРАМНИХ ЗАСОБІВ

Операційною системою розробки було обрано Ubuntu.

Ubuntu – це open-source дистрибутив GNU/Linux, що базується на основі іншого дистрибутива під назвою Debian.

Дистрибутивом GNU/Linux називають операційну систему на базі ядра Linux, що була створена на основі низки допоміжних утиліт, бібліотек та інструментів GNU і також, зазвичай, власного пакетного менеджера.

Дистрибутиви поширюються безкоштовно у формі образів, готових до установки або роботи з ПК без установки, прямо з накопичувача, разом з набором різних програм, за вибором власника дистрибутиву – це можуть бути як безкоштовні, вільні програми, так і закриті комерційні рішення.

Ubuntu – це такий само дистрибутив. Дана ОС в першу чергу адресує простоту використання та налаштування під потреби користувача. Поширюється вона попередньо встановленим набором інструментарію розробки та власним пакетним менеджером aptitude. Одним з найбільших плюсів як Ubuntu так і інших дистрибутивів є низькі вимоги до апаратних можливостей пристрою. Рекомендовані вимоги до апаратного забезпечення для встановлення Ubuntu разом з графічним інтерфейсом становлять:

- 512 МБ RAM
- Мінімум 5 ГБ на жорсткому диску

Що дає змогу встановити даний дистрибутив на практично кожен пристрій.

Основною мовою розробки було обрано JavaScript на основі виконуваного середовища Node.js.

JavaScript – це високорівнева, інтерпретована мова програмування з простим синтаксисом з широким спектром функціоналу, включеним до стандартної бібліотеки.

JS – це мультипарадигмова мова програмування, з превалюванням функціонального підходу до розробки ПЗ. Основними особливостями даної мови програмування є – динамічна слабка типізація, автоматичне керування пам'яттю, однопоточне виконання коду та зручна система базових типів. Код формується у модулі який потім можна експортувати з них.

Оскільки JavaScript це інтерпретована мова програмування, то для неї потрібно поставляти інтерпретатор та віртуальну машину на якій код буде виконуватися. Для цих задач на frontend частині використовується браузер, а для створення backend частини можна використовувати Node.js, яка включає в себе середовище для виконання коду JS, що створене на основі віртуальної машини V8, яка також використовується в браузері Google Chrome.

Даний дипломний проект побудовано на основі Node.js та реалізовано у вигляді окремих модулів, що виконують логічно завершені кроки виконання загального процесу виконання.

Таким чином було створено модуль, що забезпечує доступ до KV сховища Redis в якому зберігаються дані про сесії користувачів та їх токени авторизації у вигляді JSON об'єктів зображених на рисунку 2.1.

```
{
  "id": 1,
  "name": "Name Surname",
  "username": "username",
  "email": "email@domain.com"
}
```

Рисунок 2.1 - Формат зберігання даних у Redis

Модуль реалізує функціонал створення нових записів, читання вже створених як повністю так і за окремими полями, а також поновлення окремих полів вже створених записів. Ці дані також можна назвати "гарячими" оскільки частота запитів до них висока, оскільки перевірка токенів авторизації відбувається для кожного запиту.

Для збереження даних що використовуються не так часто було створено модуль роботи з реляційною базою даних MySQL що працює на основі ORM в якій кожна таблиця разом з усіма її колонками заздалегідь описана, як це показано на рисунку 2.2

```
{
  id: {
    type: DataTypes.INTEGER
    primaryKey: true,
    autoIncrement: true
  },
  name: DataTypes.STRING,
  username: DataTypes.STRING,
  email: {
    type: DataTypes.STRING,
    unique: true,
    validate: {
      isEmpty: true
    }
  },
  password_hash: {
    type: DataTypes.STRING,
    allowNull: false,
    validate: {
      notNull: true
    }
  },
  password_salt: {
    type: DataTypes.STRING,
    allowNull: false,
    validate: {
      notNull: true
    }
  }
}
```

Рисунок 2.2 - Схема таблиці бази даних

Для кожної таблиці що буде використовуватися потрібно створити таку модель. Створені схеми під'єднуються до класу, що представляє собою

з'єднання до бази даних, який надає стандартний інтерфейс взаємодії з базою даних.

Методами даного класу є, наприклад, отримання записів з таблиці за певною ознакою або поновлення окремих записів. Приклад зображений на рисунку 2.1 демонструє формат даних, які можна отримати у відповідь на запит про отримання інформації щодо аккаунту.

Таким чином було створено всі точки входу в сховища даних - реалізовано інтерфейси внесення та отримання даних з них. Ці модулі, а також інші, що виконують певну бізнес-логіку зручно об'єднувати у функціональні групи за сферою їх використання таким чином створюючи веб-сервіси.

Веб-сервісна архітектура - це такий підхід до розробки програмного забезпечення який, на відміну від монолітної архітектури, сприяє роздробленню структури проекту на мінімальні компоненти, що відокремлені один від одного та можуть знаходитися як у різних процесах, так і навіть на різних серверах в різних дата центрах. Кожен такий компонент має стандартизований інтерфейс та описаний формат відповідей на всі задачі, такий набір форматів називається контрактом. В межах загальної структури проекту дотримання контракту сприяє спрощенню процесу розробки і тестування взаємодії між окремими сервісами, а також виявляти помилки в роботі в процесі розробки, оскільки виконується так зване контрактне тестування, що перевіряє роботу сервісу на відповідність контракту.

На рисунку 2.3 зображено приклад реалізації даної архітектури.

Даний дипломний проект було розділено на дві функціональні частини, кожна з яких складається з декількох сервісів, що виконують вузькоспеціалізовані задачі.

Перша половина - це та частина програмного комплексу, що являється інтерфейсом вводу даних від користувачів і складається вона безпосередньо з frontend додатку, та набору сервісів на стороні backend, що займаються такими

задачами як прийом інформації від користувача, її обробка та збереження в базі даних, обслуговування авторизації запитів.

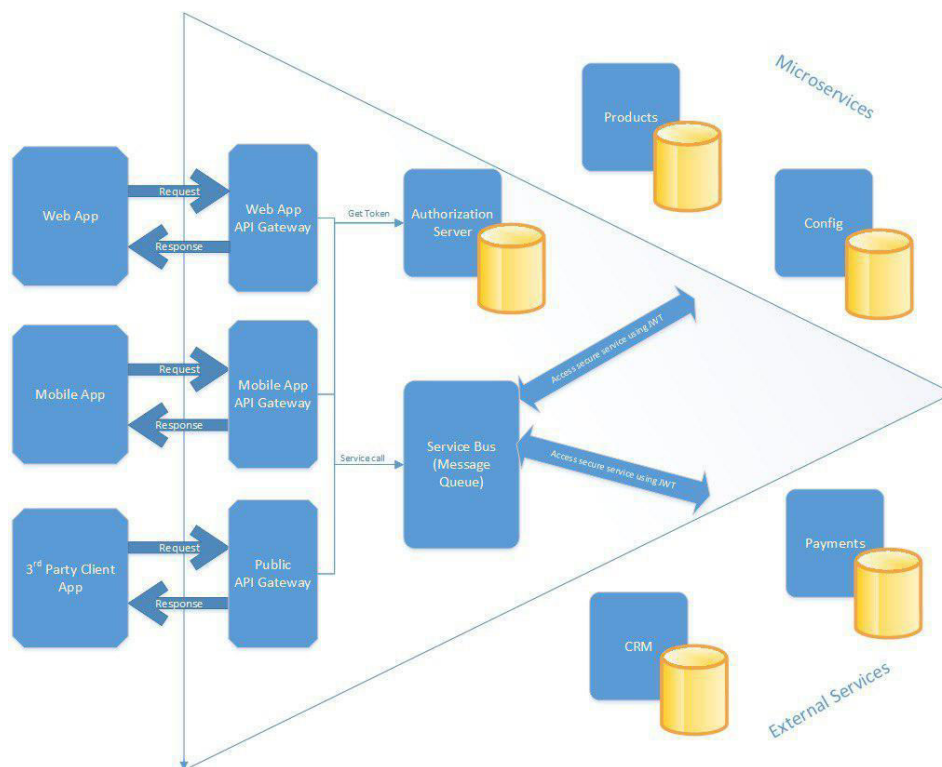


Рисунок 2.3 - Приклад реалізації веб-сервісної архітектури

Друга половина - поєднана з першою опосередковано, за допомогою бази даних. Ця частина реалізує логіку автоматичного завантаження створеної користувачами інформації та взаємодію з іншими автоматизованими системами проведення аукціону на основі стандарту OpenRTB[19] і ніяким чином не взаємодіє з користувачем.

Однією з головних проблем даної архітектури є авторизація запитів. Оскільки компоненти дуже часто розгорнуті на різних серверах, то потрібно перевіряти доступ та створювати області дозволів для запитів що виконуються від імені різних користувачів.

Для виконання таких простих операцій, як створення рекламної інформації достатньо буде використати базову авторизацію за токеном, що передається по заголовках. При першому вході, користувач повинен ввести своє ім'я та пароль (рис 2.4), у відповідь на які буде згенеровано UUIDv4 токен, за яким буде

вестися сесія користувача. Даний токен буде передаватися в заголовках кожного HTTP запиту на сервер.

```
{
  "username": "user",
  "password": "password"
}
```

Рисунок 2.4 - Формат запиту на авторизацію

Згідно зі стандартом OpenRTB версії 2.3, для передачі інформації про майбутню рекламну пропозицію використовується HTTP метод POST, в тілі запиту якого розміщена інформація про такі дані запиту на ставки як:

- Ідентифікатор рекламної пропозиції
- Тип рекламної пропозиції
- Список заблокованих рекламодавців за категоріями контенту IAB
- Список заблокованих рекламодавців за їх доменними іменами
- Інформація про запропоноване рекламне місце у вигляді масиву хеш-таблиць
 - Порядковий номер рекламного місця в контексті рекламного запиту
 - Мінімальна ставка на аукціоні за місце
 - Поле, що показує тип пропонованого рекламного місця (банер, відеореклама чи нативна реклама)
- Один з двох об'єктів що описує платформу на якій розміщено рекламне місце
 - У разі якщо місце знаходиться у додатку, то інформація про нього буде знаходитися в об'єкті з відповідною назвою та зберігати інформацію про цю платформу включаючи, але не обмежуючись полями: ідентифікатор платформи, її назва, категорії що описують тип її діяльності, посилання на неї, інформація про видавця

- Інформація про пристрій з якого відбувся рекламний запит
 - Даний розділ містить всю інформацію про пристрій яку можна використати для таргетування реклами. Наприклад, інформацію про User-Agent, тобто ідентифікатор додатку що діє від імені користувача, IP адресу, мову, модель та виробника пристрою, тип пристрою та з'єднання та також інформацію про приблизне місцезнаходження
- Повністю анонімізована інформація про поточного користувача
 - Інформація про користувача може включати його ідентифікатор, інформацію про його стать, вік та дані що описують його характеристики як покупця, але не можуть розкрити інформацію про те хто це такий

```

{
  "id": "IexxyLDIik",
  "at": 2,
  "bcat": [
    "IAB25",
    "IAB7-39",
    "IAB8-18",
    "IAB8-5",
    "IAB9-9"
  ],
  "badv": [
    "apple.com",
    "go-text.me",
    "heywire.com"
  ],
  "imp": [
    {
      "id": "1",
      "bidfloor": 0.5,
      "instl": 0,
      "tagid": "ag1tb381v1pbmNyDQsSBFNpdGUV7fD0FAw",
      "banner": {
        "w": 728,
        "h": 90,
        "pos": 1,
        "btype": [4],
        "battr": [14],
        "api": [3]
      }
    }
  ],
  "app": {
    "id": "ag1tb381v1pbmNyDAsSA0FwCBIjKfIUDA",
    "name": "Yahoo Weather",
    "cat": ["IAB15", "IAB15-10"],
    "ver": "1.0.2",
    "bundle": "com.yahoo.wxapp",
    "storeurl": "https://itunes.apple.com/id628677149",
    "publisher": {
      "id": "ag1tb381v1pbmNyDAsSA0FwCBIjKfIUCV",
      "name": "yahoo",
      "domain": "www.yahoo.com"
    }
  },
  "device": {
    "dnt": 0,
    "ua": "Mozilla/5.0 (iPhone; CPU iPhone OS 6_1 like Mac OS X) AppleWebKit/534.46 (KHTML, like Gecko) Version/5.1 Mobile/9A334 Safari/7534.48.3",
    "ip": "123.145.167.189",
    "ifa": "AA080DFE7416847C70D291f574D344790E0BB11",
    "carrier": "VERIZON",
    "language": "en",
    "make": "Apple",
    "model": "iPhone",
    "os": "iOS",
    "osv": "6.1",
    "jsv": 1,
    "connectiontype": 3,
    "devicetype": 1,
    "geo": {
      "lat": 35.012345,
      "lon": -115.12345,
      "country": "USA",
      "metro": "803",
      "region": "CA",
      "city": "Los Angeles",
      "zip": "90049"
    }
  },
  "user": {
    "id": "fffffd5135596709273b31a07e466ea2bf4fff",
    "yob": 1984,
    "gender": "M"
  }
}

```

Рисунок 2.5 - Приклад запиту для мобільної реклами

Оскільки стандартом OpenRTB архітектура спілкування між сервісами вже була визначена, то задля одноманітності підходу до розробки модулів доцільно буде використовувати той самий архітектурний підхід, а саме REST.

REST (REpresentational State Transfer) – підхід до розробки ПЗ (зазвичай веб сервісів) з певним набором правил та обмежень. Сервіси, що відповідають даному підходу називають RESTful веб сервісами (або RWS, що означає RESTful Web Service). RESTful дозволяє робити запити до зовнішніх систем для читання та маніпулювання текстовими представленнями веб ресурсів використовуючи однакові операції без збереження стану.

Архітектурний підхід REST дуже просто поєднується з мікросервісною архітектурою. Асинхронність процесу передачі повідомлень, відсутність ведення контексту передачі та невеликі розміри даних є взаємодоповнюючими рисами обох підходів.

Сервіс роботи аукціону, як вже було зазначено, матиме контракт, що забезпечений стандартом. Допоміжні сервіси початкового розміщення рекламного інвентарю, кампаній та їх налаштувань, що виконані у вигляді frontend та backend додатків обмінюватимуться мінімальною, базовою інформацією про сутності у форматі JSON. Сервіс реалізує певне API на основі REST. Таким чином відбуваються мережеві виклики на певний шлях на веб сервері.

Так, наприклад, для створення рекламної кампанії достатньо інформації про її назву та перелік налаштувань (рис 2.6) і передати її у вигляді POST запиту на веб сервер за шляхом /campaigns. Наприклад POST <http://domain.com/campaigns> якщо веб-сервер працює за адресою domain.com.

					ІАЛЦ.045430.004 ПЗ	Лист
Зм	Лист	№ докум.	Підп.	Дата		27


```

{
  "name": "Test Ad Campaign",
  "targets": {
    "os": ["Android", "iOS"],
    "country": ["UKR", "USA"],
    "ct": ["Ethernet", "WiFi"]
  }
}

```

Рис 2.6 Приклад запиту на створення кампанії

Інформація про вже створену кампанію має включати всі дані про неї, а саме її назву, параметри таргетування, а також під'єднаний до неї рекламний інвентар у вигляді масиву банерів. На рисунку 2.7 зображено приклад такого JSON об'єкту. Отримати такий об'єкт можна виконавши GET запит за шляхом /campaigns на веб-сервері. Наприклад GET <http://domain.com/campaigns>

```

{
  "id": 1,
  "name": "Test Ad Campaign",
  "targets": {
    "os": ["Android", "iOS"],
    "country": ["UKR", "USA"],
    "ct": ["Ethernet", "WiFi"]
  },
  "creatives": [{
    "id": 1,
    "name": "name",
    "url": "http://domain.com/path/to/image"
  }]
}

```

Рисунок 2.7 - Повна інформація про створену рекламну кампанію

При передачі інформації на створення рекламного банера або інакше – креатива, потрібно передати лише його назву та зображення. Оскільки зображення передається у вигляді бінарної інформації, то для передачі використовується не JSON, а інший формат передачі, що по своїй суті повторює

структуру JSON, але завдяки іншому методу серіалізації дозволяє пересилати бінарні дані у тілі запиту HTTP методу POST.

```
{  
  "name": "Creative name",  
  "file": "<Binary Data>"  
}
```

Рисунок 2.8 - Схематичне зображення формату даних на створення рекламного банера

Створений рекламний банер не зберігає інформацію про зображення у двійковому вигляді, при створенні надіслане зображення завантажується на CDN і в базу даних записується тільки посилання на це зображення. Зображений на рисунку 2.9 приклад показує таку структуру.

```
{  
  "id": 1,  
  "name": "Creative name",  
  "url": "http://domain.com/path/to/image"  
}
```

Рисунок 2.9 - Схема інформації про рекламний банер

Всі перераховані приклади включають не тільки зображені поля, а й додаткові службові, такі як, наприклад, ідентифікатор власника або дата створення.

Варто зазначити, що звичайний веб-браузер вміє виконувати лише GET запити і для ручного виконання всіх інших типів потрібно використовувати спеціальне ПЗ.

Також можна помітити, що URI для запиту може бути один і той самий (як у випадку з кампаніями) проте вибір, яку з операцій потрібно виконати базується на комбінації HTTP методу та посилання. Тому в залежності від контексту, для одного й того самого посилання можна реалізувати різні обробники, що будуть виконувати різні задачі, але в межах одного контексту.

Також, для обробки процесу фільтрування було розроблено стекову структуру даних, що містить у собі набори етапів фільтрування.

Кожна така операція представляє собою послідовне виконання стандартного набору задач. До цих задач входять: читання множини з дворівневого словника, проведення операції перетину прочитаної множини з результируючою. Кожна додана операція повинна прочитати значення зі словника за назвою операції та переданої до неї значенням.

Описані операції виконують роботу на основі лінійної моделі обчислень, тобто в стеку розміщені не результати роботи етапів, а етапи описані у якості функцій. Такий підхід дозволяє зменшити постійне навантаження та зайнятий об'єм в пам'яті оскільки обчислення будуть виконуватися тільки тоді коли дані з них будуть потрібні, також дуже зручно проводити збір сміття з такими даними, оскільки дані не будуть зберігатися довгий час в пам'яті, а виникати тільки перед використанням та зникати одразу як тільки необхідність в них зникає.

					ІАЛЦ.045430.004 ПЗ	Лист
Зм	Лист	№ докум.	Підп.	Дата		30

3. ОПИС РОЗРОБЛЕНИХ АЛГОРИТМІВ

Кожен функціональний модуль реалізує певний алгоритм взаємодії з даними. Найважливішим місцем проекту є авторизація запитів, на рисунку 3.1 зображено блок схему, що описує процес отримання первинних даних авторизації.

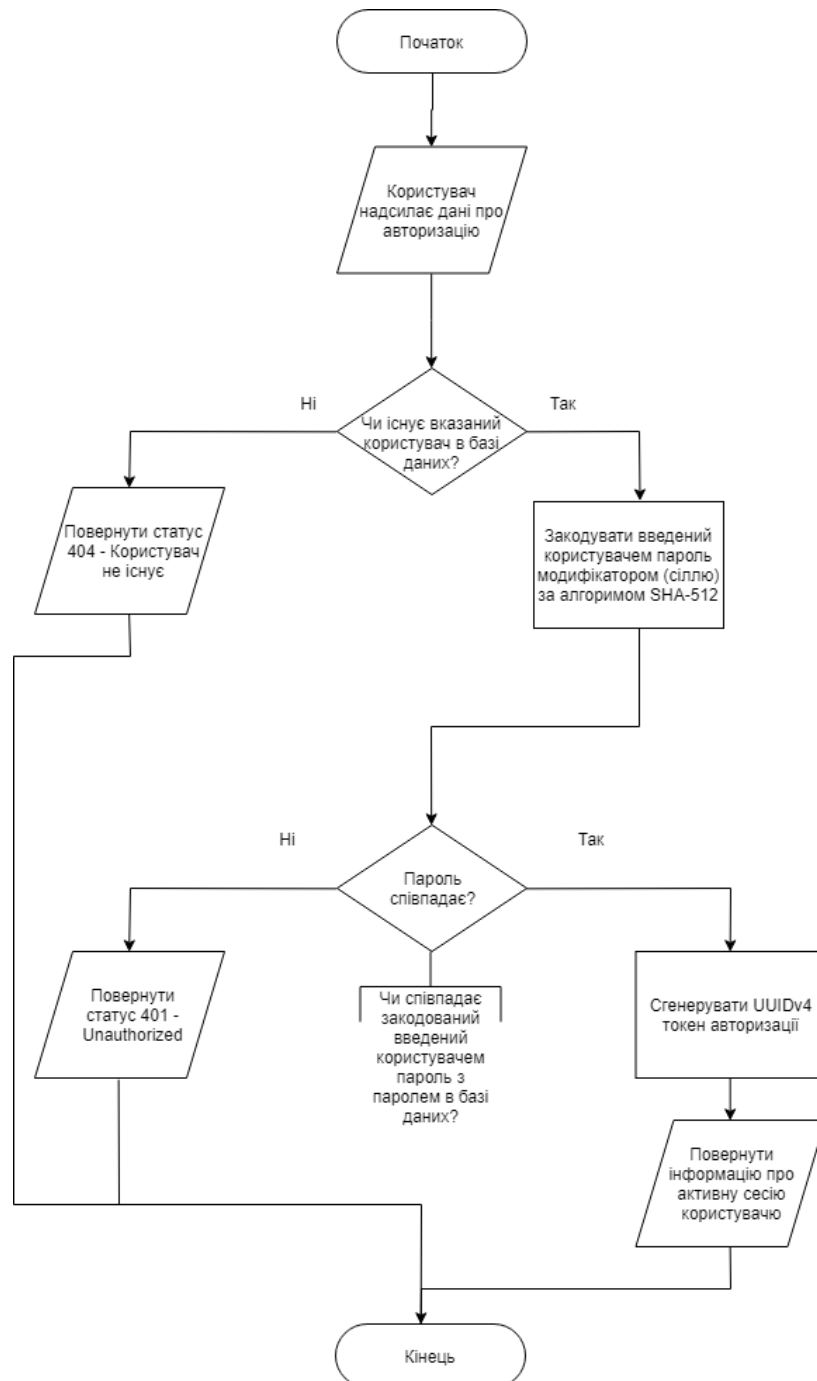


Рисунок 3.1 - Алгоритм роботи сервісу по видачі даних авторизації користувачу

Даний алгоритм ініціюється користувачем, який хоче авторизуватися для роботи із сервісом обслуговування. Для авторизації достатньо лише імені користувача та його паролю (рис 2.4). Далі в базі даних відбувається пошук даних вказаного користувача та якщо його було знайдено, відбувається шифрування введеного користувачем паролю за алгоритмом SHA-2 з 512-бітним розміром результуючого хешу. Алгоритм SHA-2 використовується з міркувань безпеки, результат хешування неможливо декодувати назад за розумний, оскільки даний алгоритм є одностороннім. Криптографічна сіль або модифікатор - це послідовність бітів, що слугують в якості початкових вхідних значень криптографічної функції. У разі співпадіння існуючого хешу функції та результату хешування введеного користувачем паролю буде створено UUIDv4 токен, що закріплюється за сесією користувача. Інформацію про користувацьку сесію та згенерований токен буде повернено користувачу у відповідь на запит.

Отриманий токен потрібно додавати в заголовки кожного запиту до сервісу. На рисунку 3.2 зображено блок схему, що описує процес авторизації кожного запиту. При виконанні запиту до сервісу, в першу чергу перевіряється наявність заголовку авторизації, в якому має зберігатися токен за принципом Bearer Authentication, за яким заголовок авторизації повинен бути заповнений у вигляді “Bearer <token>”, де token - це токен що було видано на попередньому кроці. Переконавшись в тому що заголовок присутній, потрібно перевірити наявність токenu в заголовку, після чого звернутися до кешу, в якому зберігаються дані про сесії всіх авторизованих користувачів, та отримати дані про сесію користувача за даним токеном. В разі якщо буде відсутній заголовок, токен чи користувацька сесія за даним токеном, у відповідь буде повернено відповідь зі статусом 403, що каже про те що користувач не має прав авторизації.

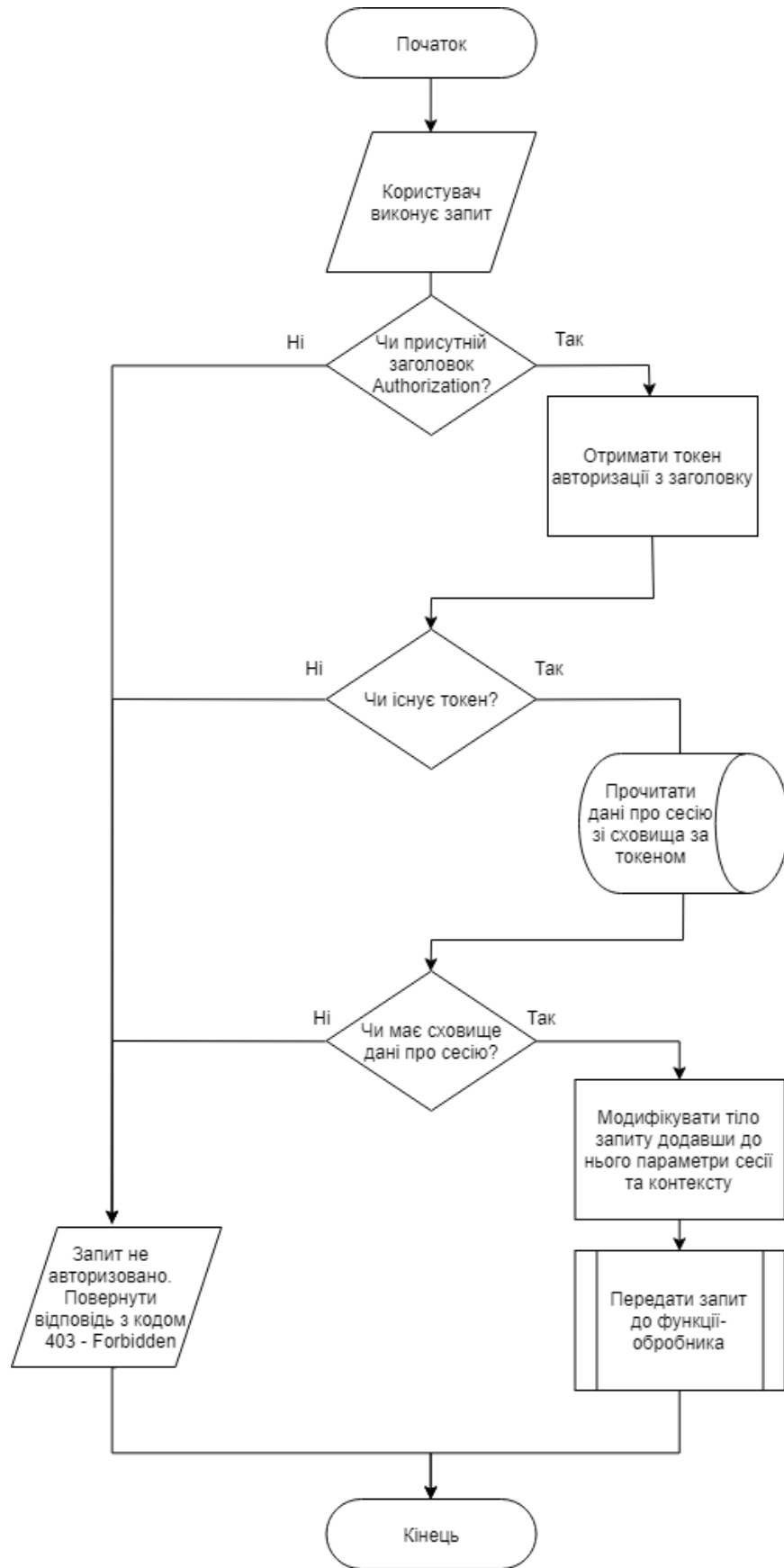


Рисунок 3.2 - Алгоритм авторизації запиту

У випадку, якщо токен сесії вірний та кеш сесій має про це запис, то в тіло запиту додається інформація про сесію - різні додаткові дані користувача, які потім буде корисно мати в функції обробнику. Після успішної роботи користувача по завантаженню рекламного інвентарю та створення кампаній до дії приступає сервіс завантаження даних з бази даних до серверу DSP алгоритм роботи якого зображено на рисунку 3.3.

Сервіс завантаження працює в одному адресному просторі з сервісом DSP, тому словник ініціаліується в процесі завантажувача та відкривається для читання сервісу DSP за допомогою влаштованих в Node.js засобів міжпроцесорної комунікації. Оскільки інший процес виконує тільки операції читання, тому необхідність у синхронізації відсутня.

Варто зазначити, що для підтримання актуальності даних, вказаний алгоритм виконується циклічно з певним інтервалом часу.

У відповідь на кожен запит до бази даних повертається масив рядків налаштувань таргетування, що групується за назвою таргетування та значенням фільтрів. Отриманий масив даних алгоритм обробляє по чергово виштовхуючи з нього елементи щоразу перевіряючи чи залишилися в ньому елементи, у разі якщо елементів більше немає, то дана ітерація алгоритму вважається завершеною, нова версія словника встановлюється поточною для використання та створюється таймер на повторне виконання алгоритму.

Дворівневий словник є словником з оберненим індексуванням в якому перший рівень це назва таргетування, другий - це значення таргетування, а саме значення, що зберігається за ключем – це масив ідентифікаторів кампаній для яких встановлено це правило. Після читання та встановлення ключів словника, масив ідентифікаторів кампаній додається до масиву за цим ключем, результат сортується за зростанням, повторні входження видаляються, виконується перезапис значення за ключем та перехід до наступного значення.

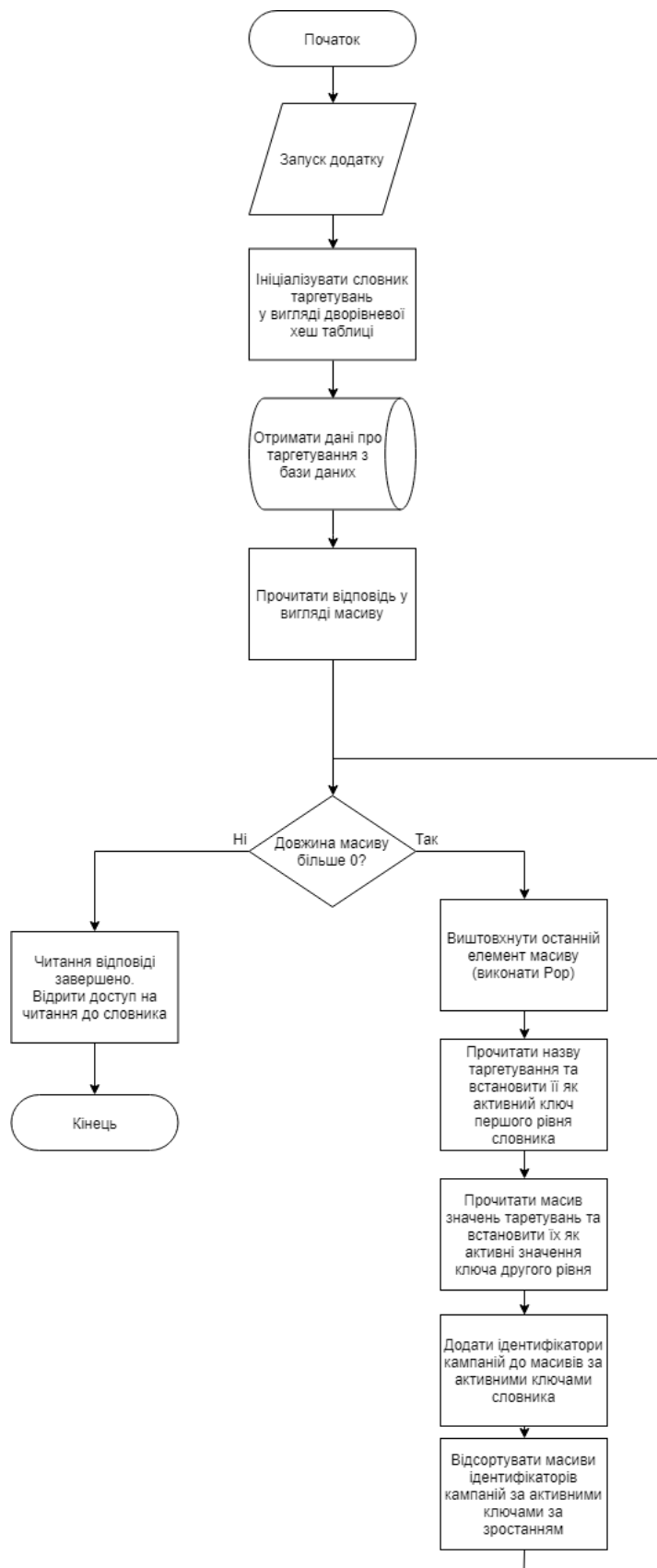


Рисунок 3.3 - Алгоритм завантаження даних

Одним з головних кроків у процесі функціонування DSP є валідація рекламного запиту та підбір відповідного рекламного інвентарю.

На кожен отриманий рекламний запит сервер DSP повинен перевірити на відповідність стандарту та відповідність підтримуваним технологіям зі стандарту, наприклад сервер може підтримувати відео рекламу, але не підтримувати банери, підтримувати певні версії протоколу відео реклами чи способи показу реклами на сторінці, тощо.

В першу чергу важливо перевірити існування головних функціональних частин запиту, тобто таких, без яких запит до даного серверу не має ніякого смислу та відсутність яких має в той же момент переривати обслуговування запиту. Оскільки в даному дипломному проєкті реалізований сервер DSP підтримує тільки банерну рекламу, тому важливо щоб рекламний запит мав хоча б одну вільну позицію для банера.

Також важливо щоб отриманий запит мав інформацію про майданчик на якому розташоване описане рекламне місце. Майданчик може бути одним з двох типів: браузерний додаток або не браузерний додаток. Інформація про майданчик зберігається у полі рекламного запиту `impr`, який в свою чергу є масивом, що описує доступні позиції для показу реклами в контексті даного запиту. Кожна позиція може мати один з трьох існуючих типів: банерна реклама, відео реклама чи нативна реклама. Кожна з яких описується полем `banner`, `video` та `native` відповідно. Описані позиції повинні мати тільки один з описаних типів, в цьому разі описаний об'єкт вважається хибним. Також кожна така позиція повинна мати ідентифікатор, який можна вказати у рекламній відповіді та значення мінімальної очікуваної вартості ставки та, якщо це не було попередньо обговорено, валюта ставки.

Оскільки реалізований сервер DSP підтримує лише банерну рекламу, то кожна описана позиція не повинна описувати будь-які інші типи реклами і такі позиції будуть відкидатися. Якщо підходящих пропозицій не було знайдено, то

такий запит вважатиметься неправильним та буде зупинене його обслуговування. Схема роботи алгоритму зображена на Рис. 3.4

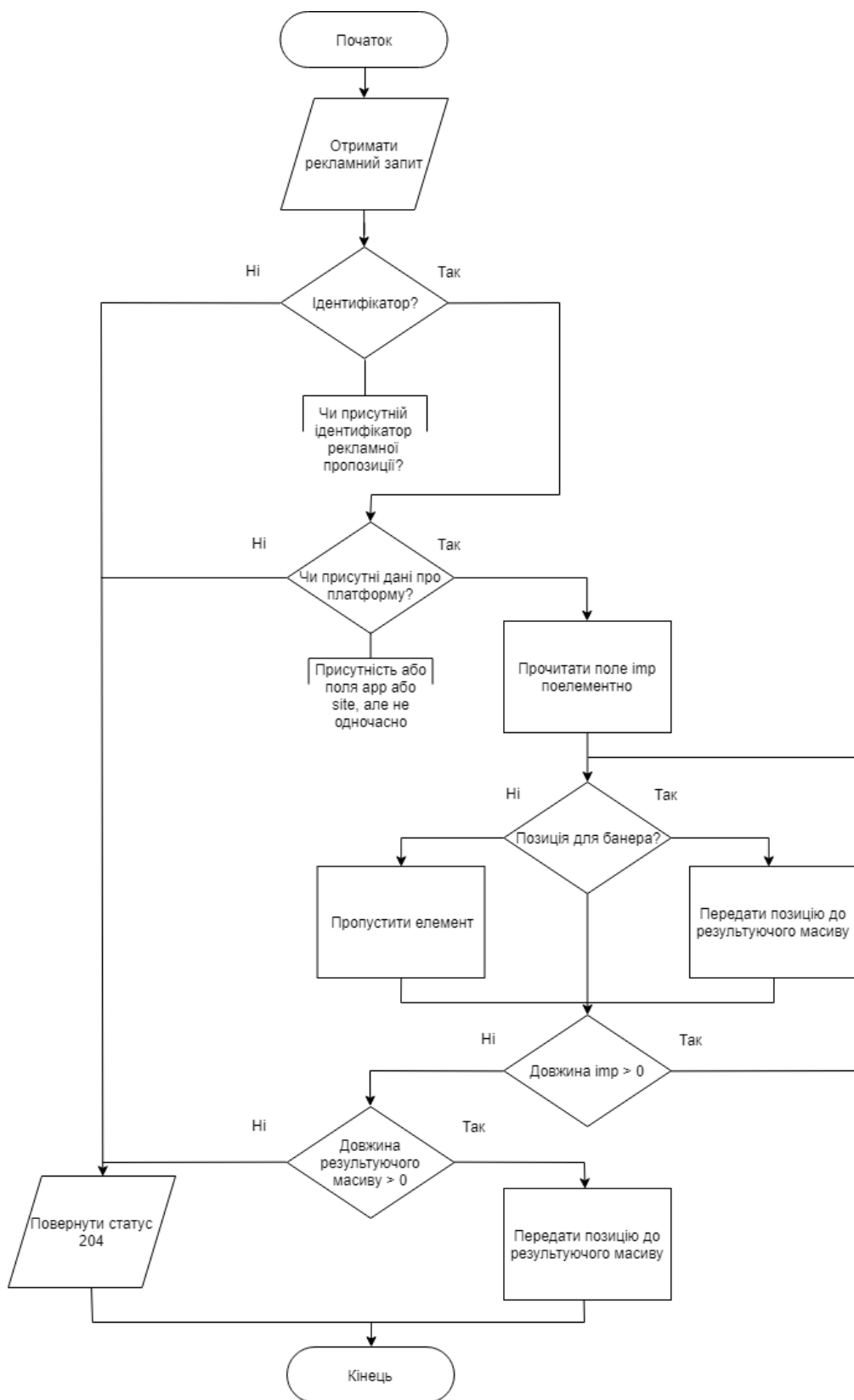


Рисунок 3.4 - Алгоритм валідації рекламного запиту

Так само, валідації підлягають ідентифікатори рекламної пропозиції, користувача, рекламного майданчику, видавництва, а також доменне ім'я майданчика та видавництва та категорії за якими вони працюють. Перечисленні поля не являються ключовими у функціонуванні рекламного запиту, проте їх відсутність не дає змогу точно визначити з якою платформою відбуватиметься співпраця, що недоречно в контексті автоматизованих угод та погіршує можливість ведення статистики роботи сервера і тому хоч їх відсутність і не впливає на роботу, але в контексті роботи серверу їх необхідно забезпечити.

Оскільки основним плюсом функціонування автоматизованих аукціонів є широкі можливості таргетування на конкретну цільову аудиторію, то важливим аспектом можна вважати інформацію щодо користувача та його пристрою. Ці надають широкий спектр налаштувань, за яким можна потрібного глядача які, до того ж, за стандартом можна розширити додатковими даними за попередньою домовленістю сторін. Дані атрибути рекламного запиту не впливають на пряму на його функціональність і тому за їх відсутності запит не буде не вірним, відсутні поля просто будуть вказуватися як деяке стандартне значення за допомогою якого буде вимикатися фільтрування за цим параметром.

Поле user являє собою об'єкт, що складається з інформації про користувача. В контексті створеного сервісу нас цікавлять: стать користувача та рік народження. У разі якщо якийсь з полів відсутніє - воно буде замінене на стандартне значення, що буде вказувати на вимикання таргетування за цим полем.

Поле device - це об'єкт, що містить інформацію щодо пристрій користувача. В контексті розробки дипломного проекту важливо знати тільки про наступні: геоінформація, інформація про виробника пристрою, його модель, операційна система та її версія, спосіб з'єднання та оператор зв'язку (мобільного чи широкосмугового доступу).

Вказана геоінформація описується полем geo та також містить набір важливої інформації щодо приблизного місця перебування користувача. Для правильної роботи нам важливо щоб були вказані наступні поля - регіон знаходження, вказаний за форматом ISO-3166-2 та місто знаходження.

Валідація полів повинна відбуватися швидко і тому прочитані значення для валідації ніяк не впливатимуть на результуюче тіло запиту.

Наступним кроком обробки запиту є його читання та представлення у вигляді зручному для внутрішньої обробки. Наприклад провести читання User-Agent, виділити тільки підходящі рекламні позиції, нормалізувати всі значення полів, тощо.

Читання рядка User-Agent дозволяє визначити додаткову інформацію про користувацький пристрій - назву операційної системи та її версію а також назву та версію агенту за допомогою якого відбувається взаємодія між користувачем та сервером. Реалізацію алгоритму зображено на рисунку 3.5 у якості блок схеми.

Також рекламний запит містить багато іншої інформації яку краще за все прочитати у зручний для внутрішньої обробки вигляді. Таким чином для зменшення часу обробки запиту доцільно використовувати першу підходящу та найдешевшу рекламну позицію, замість постійного підбору та фільтрування кожної окремої пропозиції. Також, оскільки типи пристроїв та підключення закодовано у вигляді чисел, потрібно перевести ці значення до рядків, що їх описують. Інформація, що передається рекламним запитом зазвичай збиткова та деякі можуть повторюватися у різних частинах тіла запиту. Таким чином такі поля можуть мати пріоритет, що зазвичай визначається шириною поняття поля (наприклад поле site в порівнянні з полем publisher має конкретніше значення). Приклад такого читання зображено на рисунку 3.6.

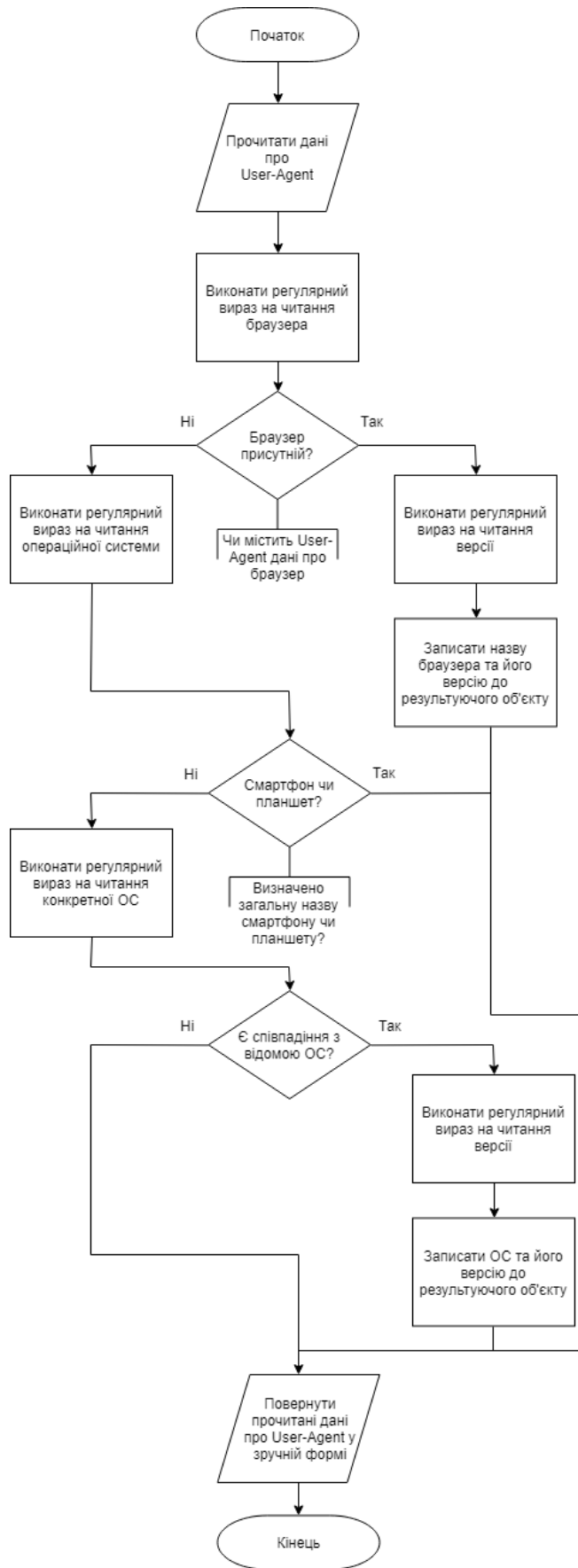


Рисунок 3.5 - Алгоритм читання User-Agent користувача

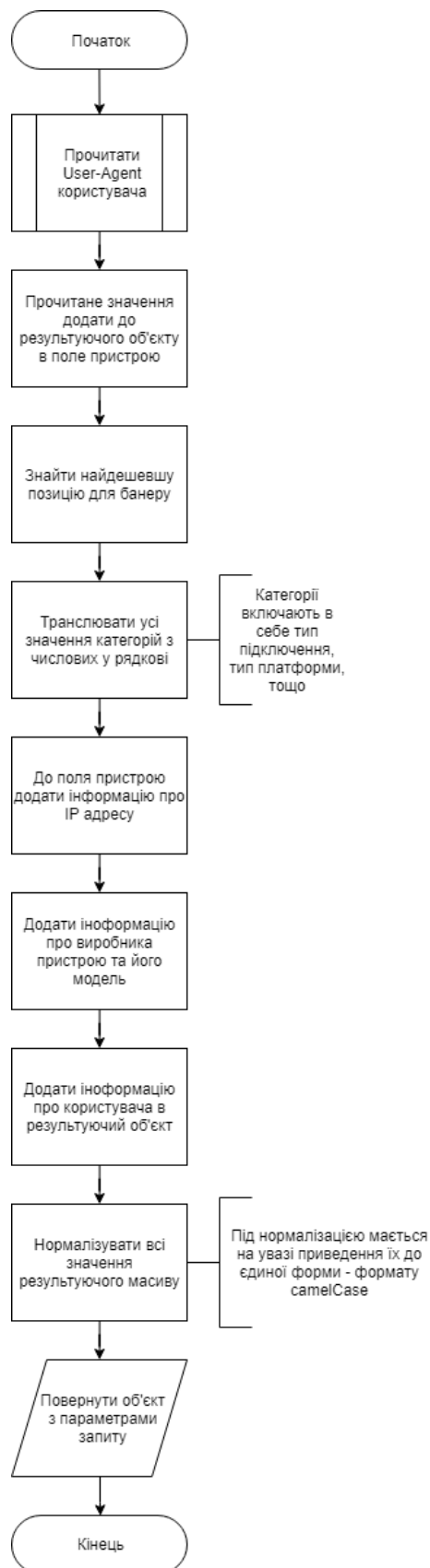


Рисунок 3.6 - Алгоритм приведення рекламного запити до внутрішньої форми

<i>Зм</i>	<i>Лист</i>	<i>№ докум.</i>	<i>Підп.</i>	<i>Дата</i>

Останнім, та найважливішим аспектом роботи серверу DSP є процес таргетування. Малі часові рамки накладають жорсткі умови на розроблений алгоритм фільтрації рекламного інвентарю.

Перш за все, описані раніше етапи обробки та читання запиту дають нам певні спрощення процесу фільтрації. Наприклад, сортування ідентифікаторів за спаданням дасть нам можливість без перевірок відсікати діапазони підходящих ідентифікаторів інвентарю між стадіями вибірок, а використання дворівневого оберненого словника спрощує доступ до цих ідентифікаторів. Вказані підходи до структурування даних дають значний приріст швидкодії, що забезпечено простотою роботи з попередньо впорядкованими структурами даних.

Кожний запит, вже прочитаний та нормалізований для внутрішнього використання потрібно розбити на етапи фільтрування, тобто створити стек дій читання активного словника кампаній. Кожен створений фільтр повинен відповідати одній з існуючих категорій фільтрування або, іншими словами, для кожного такого етапу повинен існувати ключ на першому рівні словника ідентифікаторів. В кожен етап потрібно постачати поточне значення за яким повинна відбуватися фільтрація, що також можна назвати ключем другого рівня словника ідентифікаторів. Таким чином, назва фільтру обирається з попередньо заданих, а значення фільтру передається з запиту на рекламу. Наприклад в ситуації якщо потрібно прочитати кампанії що таргетуються на Україну, потрібно встановити фільтр країн одним з етапів фільтрування та передати до нього нормалізовану назву України за стандартом ISO 3166-1 alpha-2 або ISO 3166-1 alpha-3, тобто 2 чи 3-х символним представленням назви країни. Відбудеться читання за першим рівнем по країнах, та за другим по назві країни. Результатом такого читання буде множина кампаній що потрібно провести в даній країні, у разі якщо такого ключа другого рівня не буде, то буде використано деяке стандартне значення, що характеризує всі кампанії, які явно не вказали країну проведення.

Після читання першого етапу фільтрації отримане значення встановлюється в якості результуючого, тобто такого, яке буде повернено користувачу в разі якщо етапів фільтрації більше не буде. Кожен наступний етап фільтрації читає зі словника нову множину ідентифікаторів, що відповідають заданому фільтру, між результуючою множиною та множиною, що була отримана зі словника проводиться операція перетину, яка визначає кампанії, що відповідають усім прочитаним фільтрам. Результат перетину встановлюється як нове результуюче значення.

Після кожної операції результуюче значення перевіряється на порожність, у разі якщо результуюча множина – порожня, то таку рекламну пропозицію вважають невідповідною і оброблення запиту переривають. Якщо ж результуюча множина не порожня, то оброблений етап виштовхується зі стеку виконання та відбувається перевірка на те, чи залишилися ще операції в стеку. Якщо стек не пустий, то повторюється оброблення наступного етапу фільтрування. У разі якщо стек більше не має задач, то такий процес таргетування вважається успішним і результуючу множину кампаній повертають у зовнішній код для обробки.

					ІАЛЦ.045430.004 ПЗ	Лист
Зм	Лист	№ докум.	Підп.	Дата		43

4. ВИПРОБУВАННЯ ТА РЕЗУЛЬТАТИ

Реалізована система реалізує вирішення однієї з найбільших проблем яка стоїть перед рекламними системами – одночасне обслуговування великих кількостей рекламних запитів при покращенні обслуговування кожного окремого такого запиту. Дана система повинна витримувати великі кількості одночасних запитів та обслуговувати їх в межах дозволених цим самим запитом.

Дозволений час на відповідь вимірюється у мілісекундах (мс) та зазвичай коливається у діапазоні 400-600мс, при його перевищенні рекламний запит вважається недійсним або невірним та відповідь на нього відкидається на стороні іншої рекламної мережі або видавництва. Проте верхнє значення не є рекомендованим, зазвичай максимальним можна вважати $\frac{3}{4}$ від заданого значення у запиті, тому що потрібно враховувати накладені витрати на пересилання відповіді та обробку на стороні приймача.

Тестування відбуватиметься на простому наборі даних, що складається з 5 попередньо створених рекламних банерів (креативів) та 5 рекламних кампаній. До кожної такої кампанії було під'єднано по одному банеру, кожна кампанія має стандартні налаштування таргетування.

Для перевірки робочих можливостей було реалізовано простий сервіс генерації навантаження, що періодично виконує запити серіями довільного розміру. Кожна така серія має певний відсоток хибних та не таргетованих запитів, які система повинна відкинути. Кожен дійсний запит генерується на основі попередньо завантажених даних, що підбираються таким чином, щоб результуючий запит був коректним з точки зору системи яка тестується. Результати тестування зображено на графіках 4.1 та 4.2.

Можна помітити, що з ростом запитів, росте час відповіді. Це пов'язано з тим що систему було тестовано на користувацькому ПК, де ресурси доступні для середовища виконання не є максимально можливими на конкретній

платформі і тому на обслуговування нових запитів потрібно все більше загальної пам'яті.

В цілому система працювала за планом, час відповіді не перевищував ні заплановані значення, ні значення, що становлять $\frac{3}{4}$ від максимального часу.



Графік 4.1 – Результати навантажувального тестування (середні значення)



Графік 4.2 – Результати навантажувального тестування (медіанні значення)

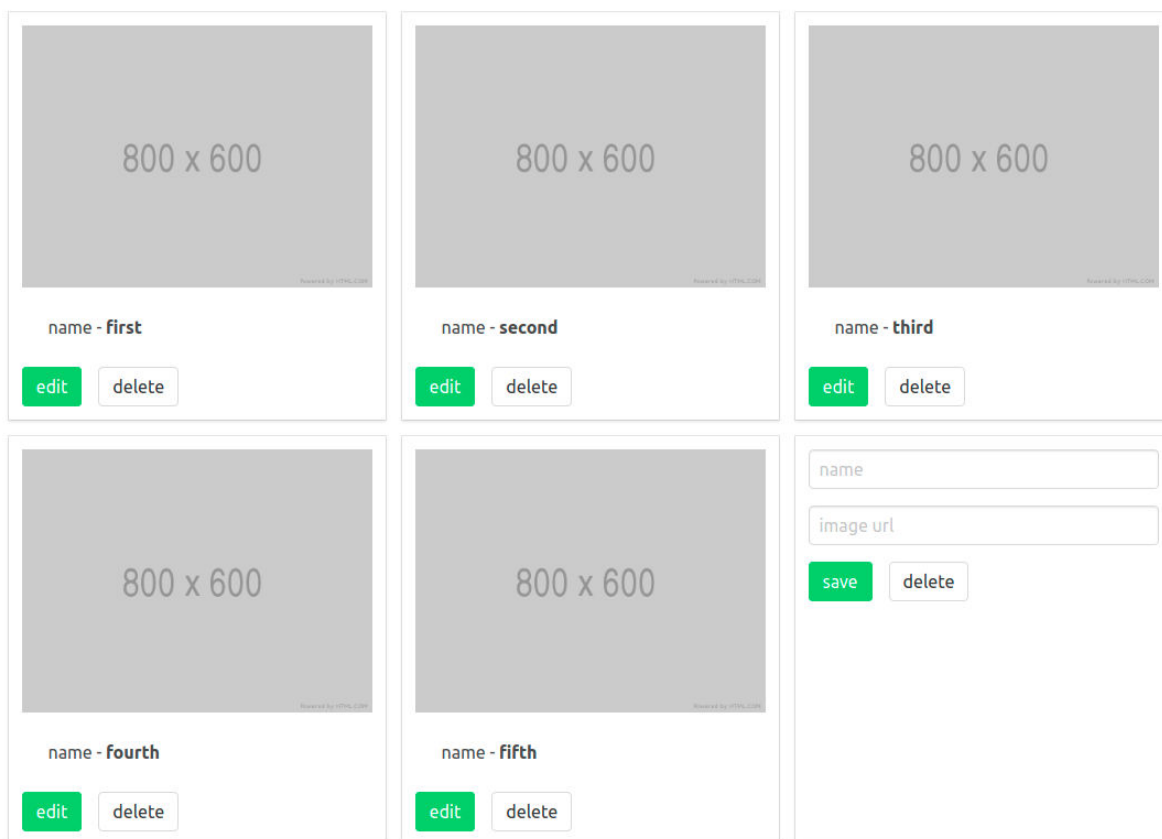


Рисунок 4.3 – Набір створених рекламних банерів

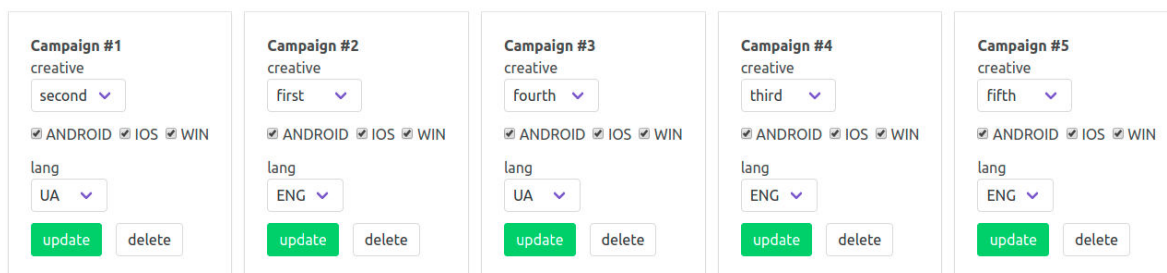


Рисунок 4.4 – Набір створених рекламних кампаній

ВИСНОВКИ

Реалізована система вирішує декілька головних проблем. Насамперед – це проблема швидкодії системи під великих об’ємів навантаження, реалізований програмний комплекс повинен обслуговувати великі об’єми навантаження за розумний, допустимий час, що диктується іншим сервісом, який його надсилає.

Другою проблемою є гарантування обслуговування кожного окремого запиту. Тобто кожен запит повинен бути прочитаний та для нього повинна бути підібрана саме та пропозиція яка для нього підходить найкраще, а саме – прочитати інформацію та на її основі відфільтрувати пропозиції та обрати найкращий варіант.

Також варто зазначити, що реалізована система має значний простір на вдосконалення. Оскільки було вирішено тільки основні проблеми галузі, то система має способи її вдосконалення шляхом вирішення супутніх проблем.

Вектори розвитку включають в себе, але не обмежуються такими варіантами як:

- Підрахунок загальної кількості надісланих запитів від кожного партнера
- Введення метрики використання кожної окремої кампанії та їх консолідація для відображення користувачу, що їх створив
- Створення сервісу кешування рекламних креативів для прискорення доступу
- Реалізація логування роботи сервісу
- Додання нових правил таргетування
- Введення системи ведення бюджету кампаній та аккаунтів

Модульність реалізованої архітектури забезпечує можливість додавання нових функціональних частин без необхідності у внесенні критичних чи глобальних змін до структури проекту.

					ІАЛЦ.045430.004 ПЗ	Лист
Зм	Лист	№ докум.	Підп.	Дата		47

Серед вирішених проблем також можна помітити той факт, що в даній сфері, при таких об'ємах навантажень, навіть оптимізовані алгоритми не справляються з навантаженням і при їх роботі виникають проблеми пов'язані з фундаментальними обмеженнями апаратної платформи на якій розгортається програмний комплекс, такі як, наприклад, обмеження у швидкості читання/запису, обмежений об'єм пам'яті та обмеження у максимальній пропускній можливості мережі зв'язку. При досягненні таких обмежень, можливим варіантом вирішення є тільки масштабування апаратної платформи шляхом покращення конфігурації серверу чи збільшення їх кількості.

					ІАЛЦ.045430.004 ПЗ	Лист
Зм	Лист	№ докум.	Підп.	Дата		48

СПИСОК ВИКОРИСТАНИХ ЛІТЕРАТУРНИХ ДЖЕРЕЛ

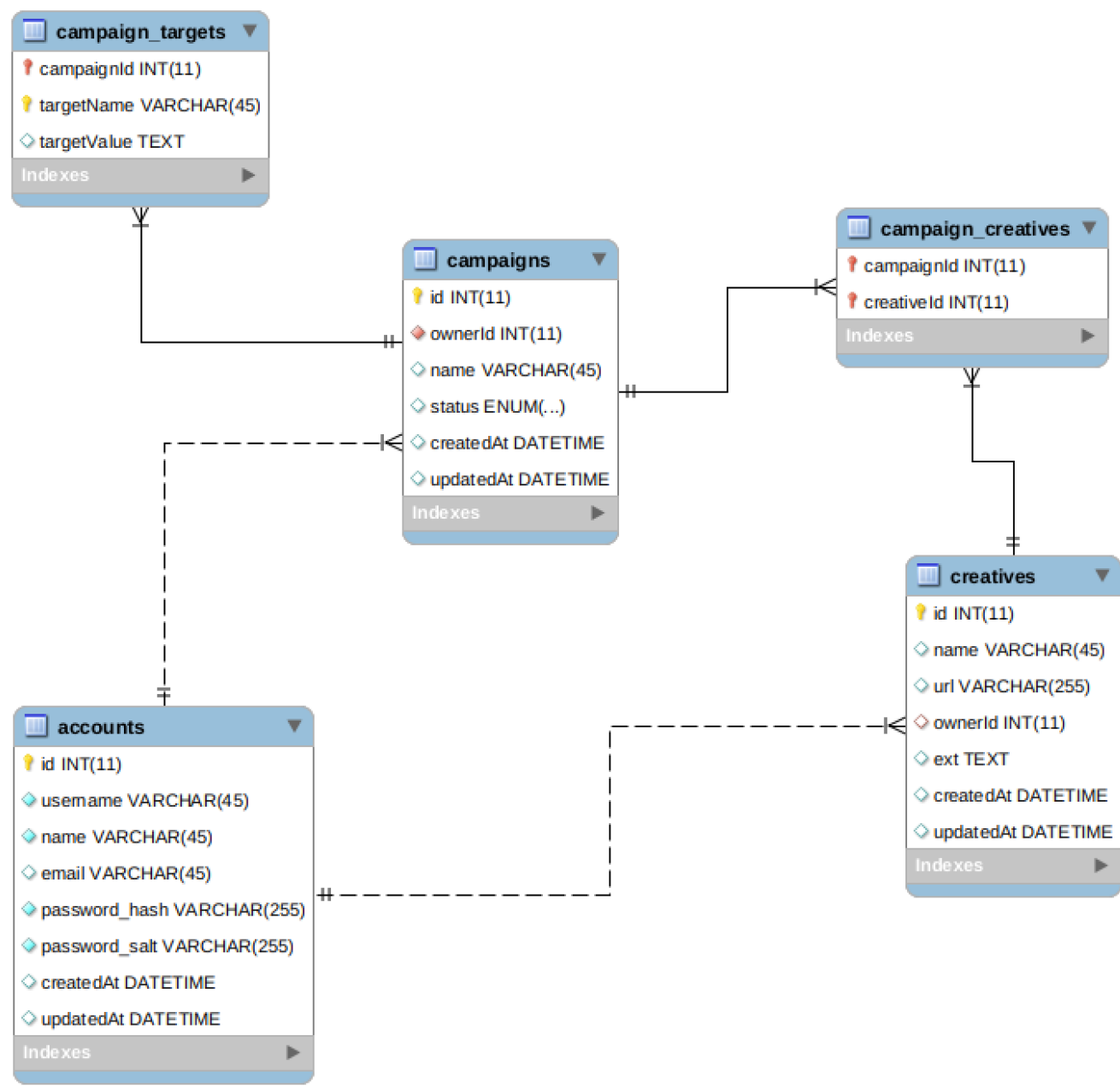
1. Едельман, Б. Strategic bidder behavior in sponsored search auctions. Decision support systems [Текст] / Б. Едельман, М.Островський 43(1) 192–198.
2. Мілґром, П.Р. Putting auction theory to work. [Текст] / П.Р Мілґром — Cambridge University Press, 2004.
3. Фельдман, Дж. Budget optimization in search-based advertising auctions. In Proceedings of the 8th ACM conference on Electronic Commerce [Текст] / Дж. Фельдман 40-49.
4. Хосангар К., Черепанов В., Optimal bidding in stochastic budget constrained slot auctions. In Proceedings of the 9th ACM conference on Electronic Commerce К. Хосангар, В. Черепанов 20-22.
5. Гош А., Рубінштейн Б.І., Adaptive bidding for display advertising. In Proceedings of the 18th International Conference on World Wide Web [Текст] / А Гош, Б.І. Рубінштейн 251-260.
6. Едельман Б., Островський М., Internet advertising and the generalized second price auction: Selling billions of dollars worth of keywords. Technical report, National Bureau of Economic Research / Б. Едельман, М. Островський
7. Борґс К., Чейс Дж., Dynamics of bid optimization in online advertisement auctions. In Proceedings of the 16th International Conference on World Wide Web [Текст] / К. Борґс, Дж. Чейс 531-540
8. Фельдман Дж., Budget optimization in search-based advertising auctions. In Proceedings of the 8th ACM conference on Electronic Commerce [Текст] / Дж. Фельдман 40-49
9. Бродер А., Bid generation for advanced match in sponsored search. In Proceedings of the Fourth ACM International Conference on Web Search and Data Mining [Текст] / А. Бродер, Є. Габрилович 515-524

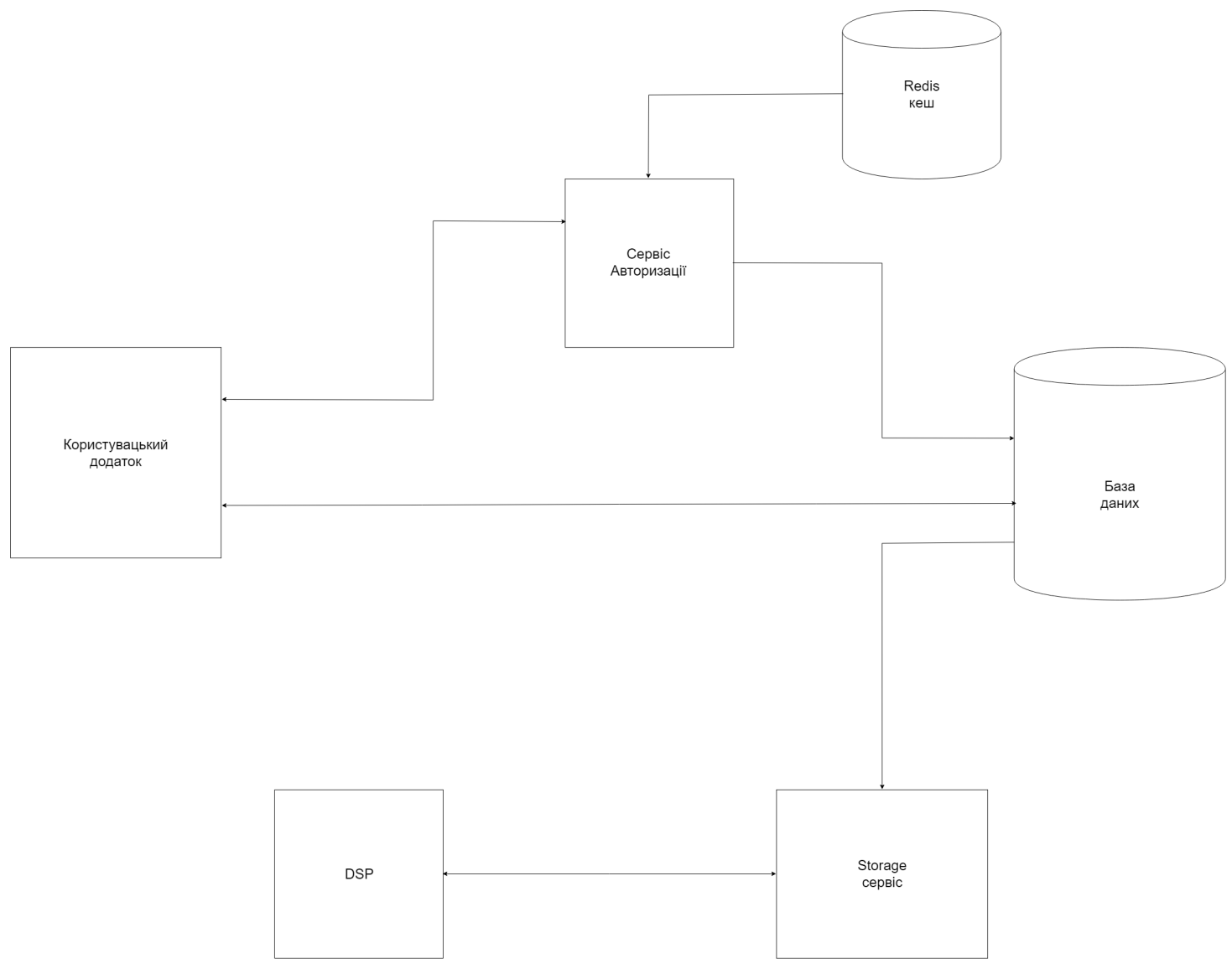
10. Івен Дар Є. Bid optimization for broad match ad auctions. In Proceedings of the 18th International Conference on World Wide Web [Текст] / Є. Івен Дар 231-240
11. Цзян В. Real-time bidding benchmarking with ipinyou dataset. / В. Цзян, С. Чжан. Б. Гао
12. Амін К. Budget optimization for sponsored search: Censored learning in MDPs. In Proceedings of the 28th Conference on Uncertainty in Artificial Intelligence / К. Амін, М. Кернс, П. Б. Кей
13. Кітс Б. Optimal bidding on keyword auctions. Electronic Markets [Текст] / Б. Кітс, Б. Леблан 14(3) 186-201
14. Абрамс З., Revenue maximization when bidders have budgets. In Proceedings of the Seventeenth Annual ACM-SIAM Symposium on Discrete Algorithm [Текст] / З. Абрамс 1074-1082
15. Юян С. Real-time bidding for online advertising: measurement and analysis. In Proceedings of the Seventh International Workshop on Data Mining for Online Advertising [Текст] / С. Юян, Дж. Ван
16. Гош А. Adaptive bidding for display advertising. In Proceedings of the 18th International Conference on World Wide Web [Текст] / А. Гош, Б. І Рубінштейн 251-260
17. Чен Ю. Real-time bidding algorithms for performance-based display ad allocation. In Proceedings of the 17th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining [Текст] / Ю. Чен, П. Беркін, Б. Андерсон 1307-1315
18. Дасдан А. Real time bid optimization with smooth budget delivery in online advertising. In Proceedings of the Seventh International Workshop on Data Mining for Online Advertising [Текст] / А. Дасдан

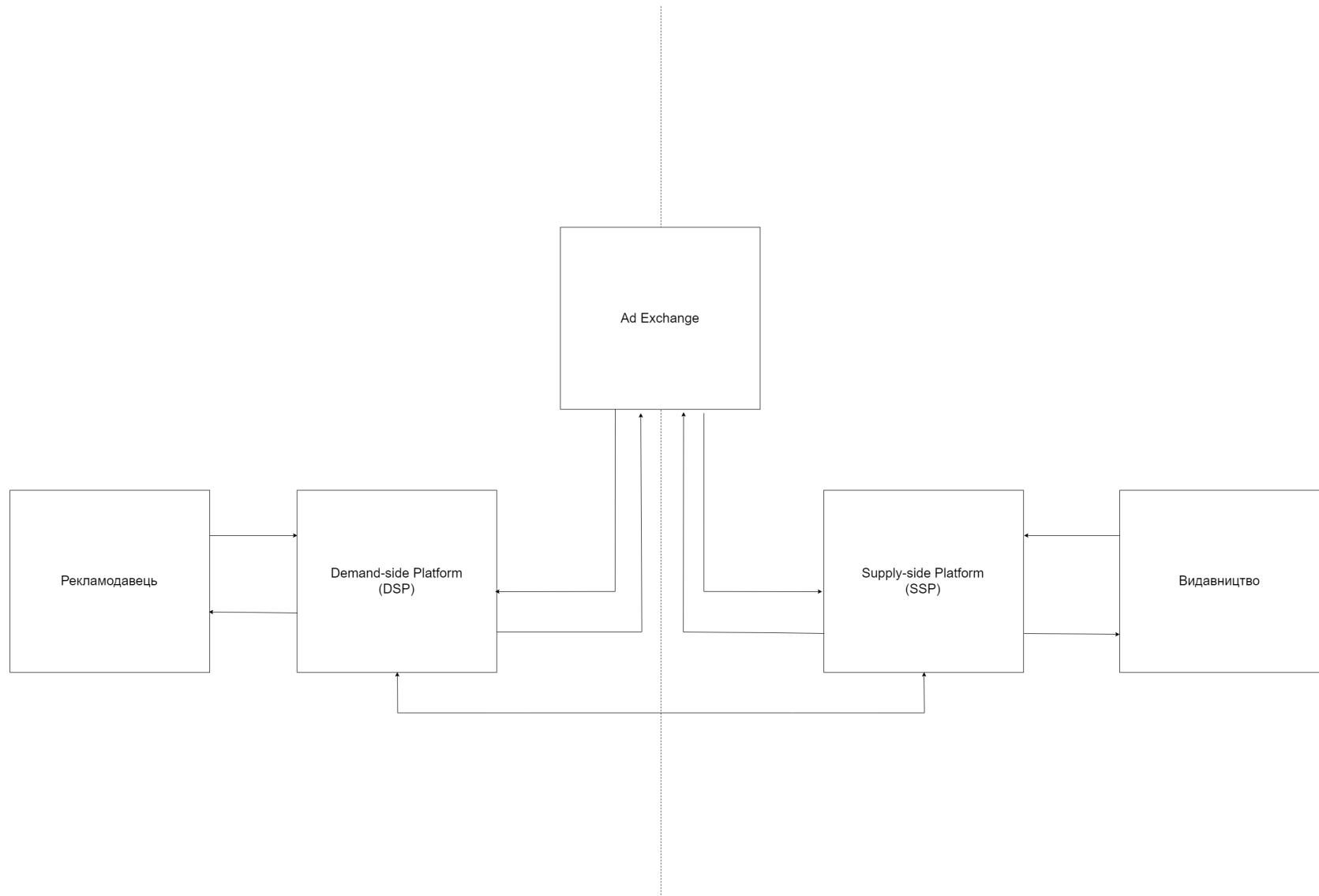
19.Стандарт OpenRTB [Електронний ресурс]. — Режим доступу:
https://www.iab.com/wp-content/uploads/2015/05/OpenRTB_API_Specification_Version_2_3_1.pdf

					ІАЛІЦ.045430.004 ПЗ	Лист
Зм	Лист	№ докум.	Підп.	Дата		51

Додаток 1
Копії графічних матеріалів







					ІАЛЦ.045440.007 Д3		
Зм	Арк	№ докум	Підпис	Дата	Програмна система управління інформаційними потоками у галузі медіайної реклами		
Розроб		Маслов В.І			Ліп	Маса	Масштаб
Перев		Литвищенко А.В			Аркуш 1		Аркуш 1
Техніч					Лукціон OpenRTB. Структурна схема.		
Наказ		Клименко В.М			КПІ ім. Ігоря Савурського, ФТМ, КБ-31		
Затв		Тарасенко В.П					

