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

Кафедра системного програмування

і спеціалізованих комп‘ютерних систем

«На правах рукопису» УДК 004.77

До захисту допущено:

Завідувач кафедри

_____Віталій РОМАНКЕВИЧ

«___»_____2022 р.

Магістерська дисертація

на здобуття ступеня магістра

за освітньо-професійною програмою «Системне програмування та спеціалізовані комп‘ютерні системи»

зі спеціальності 123 «Комп‘ютерна інженерія»

на тему: «Засоби для надсилання коротких повідомлень стільниковою мережею»

Виконав:
студент ІІ курсу, групи КВ-11мп
Качанов Віталій Ігорович _____

Науковий керівник:
доцент кафедри СПіСКС к.т.н.,
доцент Тарасенко-Клятченко Оксана Володимирівна _____

Консультант з нормоконтролю:
доцент, с.н.с., к.т.н.,
Боярінова Юлія Євгенівна _____

Рецензент: к.т н., доц.каф.
обчислювальної техніки
Корочкін А.В. _____

Засвідчую, що у цій магістерській дисертації немає запозичень з праць інших авторів без відповідних посилань.
Студент _____

Київ – 2022 року

Національний технічний університет України

«Київський політехнічний інститут імені Ігоря Сікорського»

Факультет прикладної математики

Кафедра системного програмування та спеціалізованих комп‘ютерних систем

Рівень вищої освіти – другий (магістерський)

Спеціальність – 123 «Комп‘ютерна інженерія»

Освітньо-професійна програма «Системне програмування та спеціалізовані комп‘ютерні системи»

ЗАТВЕРДЖУЮ

Завідувач кафедри

_____Віталій РОМАНКЕВИЧ

«___»_____20__р.

ЗАВДАННЯ

на магістерську дисертацію

студенту Качанов Віталій Ігорович

1. Тема дисертації «Засоби для надсилання коротких повідомлень стільниковою мережею», науковий керівник дисертації Тарасенко-Клятченко Оксана Володимирівна, доцент кафедри СПСКС к.т.н., доцент, затверджені наказом по університету від «25» листопада 2022 р. № 4317-С

2. Термін подання студентом дисертації. 13 грудня 2022р.

3. Об’єкт дослідження: система надсилання коротких повідомлень стільниковою мережею.

4. Вихідні дані. Розроблена система надсилання коротких повідомлень стільниковою мережею.

5. Перелік завдань, які потрібно розробити: специфікувати вимоги до системи, аналіз аналогічних систем, дослідження переваг та недоліків, формування логічної та фізичної структури, побудова концептуальної та об’єктно орієнтованої моделі, реалізувати систему.

6. Орієнтовний перелік графічного (ілюстративного) матеріалу: презентація, діаграма варіантів використання, діаграма розгортання, діаграма потоків даних

7. Орієнтовний перелік публікацій. За тематикою проведених досліджень опубліковано 2 наукові праці, тези доповідей на конференціях.

8. Дата видачі завдання. 1 жовтня 2021р.

Календарний план

№ з/п	Назва етапів виконання магістерської дисертації	Термін виконання етапів магістерської дисертації	Примітка
1	Постановка мети та цілі роботи	01.11.2021	
2	Вивчення теоретичного матеріалу	01.03.2022	
3	Аналіз існуючих систем	01.05.2022	
4	Проєктування системи	01.07.2022	
5	Реалізація нової системи	01.09.2022	
6	Тестування системи	01.10.2022	
7	Написання дисертації	20.10.2022	
8	Попередній розгляд магістерської дисертації на кафедрі	30.11.2022	

Студент Віталій КАЧАНОВ

Науковий керівник Оксана ТАРАСЕНКО-КЛЯТЧЕНКО

РЕФЕРАТ

Актуальність теми. Швидкість доставки інформації до кінцевого користувача збільшується з кожним роком. Такі темпи розвитку технологій змушують переглядати методи роботи з даними та інформацією. Все частіше люди замінюють голосове спілкування на текстове листування. Звикання до читання великих обсягів текстової інформації дозволяє розширити сфери, у яких можливе застосування автоматизованих інформаційних систем. Але при впровадженні будь-якої інформаційної системи виникає питання щодо організації комунікації з кінцевими користувачами. Дуже часто це питання дозволяє вирішити сервіс коротких повідомлень – SMS.

Незважаючи на те, що технологія SMS була розроблена понад 30 років тому, вона й досі відповідає сучасним вимогам бізнесу. SMS - розсилка є одним із найдоступніших способів просувати свої товари та послуги, інформувати про будь-які події ваших клієнтів. Сервіси СМС розсилки доставляють клієнту всю важливу інформацію не тільки в рекламних цілях, але і для підтримки зв'язку.

Отже, моделювання системи посилання SMS повідомлень абонентам стільникової мережі є актуальним завданням.

Об’єктом дослідження є система надсилання коротких повідомлень стільниковою мережею

Предметом дослідження короткі повідомлення надіслані стільниковою мережею

Мета роботи: аналіз існуючих систем надсилання коротких повідомлень стільниковою мережею. Розробка архітектури системи надсилання коротких повідомлень стільниковою мережею та реалізація її.

Наукова новизна полягає в наступному:

1. Закладення в архітектуру можливість як надсилати повідомлення так і отримувати нові від отримувача назад.

Практична цінність отриманих в роботі результатів полягає в тому, що запропонована система може масштабуватись та досягати до позначки 60 тисяч меседжів в секунду.

Апробація роботи. Основні матеріали роботи були представленні на:

- Науково-технічній конференції магістрантів кафедри СКС ФПМ НТУУ "КПІ" ПМК-2022.
- IX Міжнародна науково-технічна Internet-конференція «Сучасні методи, інформаційне, програмне та технічне забезпечення систем керування організаційно-технічними та технологічними комплексами».

Публікації. За тематикою проведених досліджень опубліковано 2 наукові праці, а саме тези доповідей на 2-х конференціях.

Структура та обсяг роботи. Магістерська дисертація складається з вступу, чотирьох розділів та висновків.

У вступі подано загальну характеристику роботи, зроблено оцінку сучасного стану проблеми, обґрунтовано актуальність напрямку досліджень, сформульовано мету і задачі досліджень, показано наукову новизну отриманих результатів і практичну цінність роботи, наведено відомості про апробацію результатів і їхнє впровадження.

У першому розділі розглянуто існуючі рішення систем надсилання коротких повідомлень стільниковою мережею. Проаналізовано процес надсилання коротких повідомлень стільниковою мережею та визначені

вимоги до системи надсилання коротких повідомлень стільниковою мережею.

У другому розділі розроблена концептуальна модель системи надсилання коротких повідомлень стільниковою мережею.

У третьому розділі розроблена об’єктно-орієнтована модель системи надсилання коротких повідомлень стільниковою мережею.

У четвертому розділі продемонстровані реалізація, тестування та робота системи надсилання коротких повідомлень стільниковою мережею.

У висновках представлені результати проведеної роботи.

Ключові слова: стільникова мережа, короткі повідомлення.

ABSTRACT

Relevance of the topic. The speed of information delivery to the end user increases every year. Such a pace of technology development forces us to revise the methods of working with data and information. Increasingly, people are replacing voice communication with text correspondence. Getting used to reading large amounts of textual information allows you to expand the areas in which the use of automated information systems is possible. But when introducing any information system, a question arises regarding the organization of communication with end users. Very often, this issue allows you to solve the service of short messages – SMS.

Despite the fact that SMS technology was developed more than 30 years ago, it still meets modern business requirements. SMS sending is one of the most affordable ways to promote your products and services, to inform your customers about any events. SMS mailing services deliver all important information to the client not only for advertising purposes, but also to maintain communication.

So, modeling the system of linking SMS messages to subscribers of a cellular network is an urgent task.

The object of the study is the system of sending short messages by the cellular network

The subject of the study short messages sent by the cellular network

Objective: analysis of existing systems for sending short messages by a cellular network. Development of the architecture of the system for sending short messages by a cellular network and its implementation.

Scientific novelty is as follows:

1. Embedding in the architecture the ability to both send messages and receive new ones from the recipient back.

The practical value of the results obtained in the work lies in the fact that the proposed system can scale and reach up to 60 thousand messages per second.

Approbation of work. The main materials of the work were presented on:

1. Scientific and technical conference of undergraduates of the department SCS FPM NTUU "KPI" PMK-2022.
2. IX International Scientific and Technical Internet-conference "Modern methods, information, software and technical support of control systems of organizational, technical and technological complexes".

Publication. 2 scientific works were published on the subject of the research, namely abstracts of reports at 2 conferences.

Structure and scope of work. The master's thesis consists of an introduction, four chapters and conclusions.

In the introduction, a general description of the work is presented, an assessment of the current state of the problem is made, the relevance of the research direction is substantiated, the purpose and objectives of the research are formulated, the scientific novelty of the results and the practical value of the work are shown, information about the approbation of the results and their implementation is given.

The first section discusses existing solutions for systems for sending short messages over a cellular network. The process of sending short messages over a cellular network is analyzed and the requirements for the system for sending short messages over a cellular network are defined.

In the second section , a conceptual model of the system for sending short messages over a cellular network has been developed.

In the third section, an object-oriented model of the system for sending short messages over a cellular network has been developed.

The fourth section demonstrates the implementation, testing and operation of the system for sending short messages over a cellular network.

The conclusions present the results of the work done.

Keywords: cellular network, short messages.

ЗМІСТ

СПИСОК ТЕРМІНІВ, СКОРОЧЕНЬ ТА ПОЗНАЧЕНЬ.....12

ВСТУП.....14

1. СПЕЦИФІКАЦІЯ ВИМОГ ДО СИСТЕМИ.....15

1.1. Аналіз процесу посилання SMS повідомлень абонентам стільникового зв'язку.....15

1.2. Аналіз систем посилання SMS повідомлень абонентам стільникової мережі.....20

1.3. Визначення вимог до системи надсилання коротких повідомлень користувачам стільникового зв'язку.....27

Висновки.....31

2. КОНЦЕПТУАЛЬНА МОДЕЛЬ СИСТЕМИ.....32

2.1. Формалізування процесу відправки SMS повідомлень абонентам стільникової мережі.....32

2.2. Формулювання зв'язків між етапами процесу посилання SMS повідомлень абонентам стільникової мережі.....37

2.3. Формулювання потоків інформації поміж етапами посилання SMS повідомлень абонентам стільникової мережі.....40

Висновки.....42

3. ОБ'КТНО-ОРІЄНТОВАНА МОДЕЛЬ СИСТЕМИ.....43

3.1. Формулювання варіантів використання системи.....43

3.2. Логічна структура системи.....53

3.3. Фізична структура системи.....58

Висновки.....63

4. РОБОЧИЙ ПРОЄКТ СИСТЕМИ.....64

4.1. Реалізування системи.....64

4.2. Тестування системи.....72

4.3. Демонстрація системи.....79

4.4. Порівняння швидкодії роботи клієнтської частини.....85

Висновки.....89
ВИСНОВКИ.....91
СПИСОК ВИКОРИСТАНИХ ЛІТЕРАТУРНИХ ДЖЕРЕЛ.....92
ДОДАТКИ.....94

СПИСОК ТЕРМІНІВ, СКОРОЧЕНЬ ТА ПОЗНАЧЕНЬ

API – (англ. Application programming interface) опис методів, за допомогою яких одне програмне забезпечення може взаємодіяти з іншим.

CSV – текстовий формат, для відображення табличних даних.

DFD – (англ. data flow diagrams*)* діаграми «потоків» даних.

EMS – стандарт передачі повідомлень, який являє собою розширенням стандарту SMS для стільникових мереж стандартів GSM, TDMA.

GSM – міжнародний стандарт для мобільного цифрового стільникового зв'язку з розділенням каналу за принципом TDMA та високим рівнем безпеки за рахунок шифрування з відкритим ключем.

IDEF0 – методологія функціонального моделювання (англ. function modeling) і графічна нотація, призначена для формалізації і опису бізнес-процесів.

IDEF3 – стандарт документування процесів, що відбуваються в системі.

MariaDB – СУБД розроблене на основі MySQL, що розробляється спільнотою під ліцензією GNU GPL.

MySQL – реляційна система управління базами даних.

Nats Jetstream –

Node.js – фреймворк для використання JavaScript, як мови для розробки серверних додатків.

Redis – сховище пар ключ-значення в оперативній пам'яті, що використовується в якості розподіленої бази даних ключ-значення в пам'яті, кеша і брокера повідомлень.

SMS – (англ. *SMS, Short Message Service*) послуга обміну (передачі і прийому) короткими текстовими повідомленнями в телекомунікаційних мережах.

TDMA – (англ. Time division multiple access) це метод часового поділу одного фізичного каналу зв'язку.

TypeScript – типізована мова програмування, для побудови вебдодатків.

UCS-2 – (англ. Universal Coded Character Set) набор стандартних символів.

ВСТУП

Швидкість доставки інформації до кінцевого користувача збільшується з кожним роком. Такі темпи розвитку технологій змушують переглядати методи роботи з даними та інформацією. Все частіше люди замінюють голосове спілкування на текстове листування. Звикання до читання великих обсягів текстової інформації дозволяє розширити сфери, у яких можливе застосування автоматизованих інформаційних систем. Але при впровадженні будь-якої інформаційної системи виникає питання щодо організації комунікації з кінцевими користувачами. Дуже часто це питання дозволяє вирішити сервіс коротких повідомлень – SMS.

Незважаючи на те, що технологія SMS була розроблена понад 30 років тому, вона й досі відповідає сучасним вимогам бізнесу. SMS - розсилка є одним із найдоступніших способів просувати свої товари та послуги, інформувати про будь-які події ваших клієнтів. Сервіси СМС розсилки доставляють клієнту всю важливу інформацію не тільки в рекламних цілях, але і для підтримки зв'язку.

Отже, моделювання системи посилання SMS повідомлень абонентам стільникової мережі є актуальним завданням.

1. СПЕЦИФІКАЦІЯ ВИМОГ ДО СИСТЕМИ

1.1. Аналіз процесу посилання SMS повідомлень абонентам стільникового зв'язку

Служба коротких повідомлень — з точки в точку, відома як Short Message Service—Point to Point або SMS-PP — спочатку була визначена в стандарті протоколу GSM 03.40, яка зараз підтримується в 3GPP як TS 23.040 GSM 03.41[1] (тепер 3GPP TS 23.041) визначає службу коротких повідомлень — стільникову трансляцію (Short Message Service—Cell Broadcast), яка дозволяє надсилати повідомлення (рекламу, публічну, приватну інформацію тощо) усім користувачам стільникового зв'язку у визначеній географічній зоні.

Повідомлення надсилаються до центру обслуговування коротких повідомлень (SMSC), який забезпечує механізм «зберігання та пересилання». Він намагається надіслати повідомлення одержувачам SMSC та дізнається про стан SIM-карти за допомогою HLR (Home Location Register) сервісів, які надають інформацію про стан активності абонента стільникової мережі. Якщо одержувач недоступний, SMSC ставить повідомлення в чергу для повторної спроби пізніше. Деякі SMSC також надають опцію «переслати й забути», коли передача здійснюється лише один раз без підтвердження надсилання короткого повідомлення.

SMS можна відправляти кількома способами:

- З мобільного пристрою на інший мобільний пристрій - MO-MT (Mobile Origined - Mobile Terminated)
- Від мобільного пристрою до постачальника контенту (ESME) - називається MO-AT (Mobile Originated - Application Terminated)
- Від програми до мобільного пристрою - називається AO-MT (Application Origined - Mobile Terminated)

. Доставка повідомлень — це «негарантована доставка», тому немає жодних гарантій, що коротке повідомлення буде доставлено абоненту

стільникового зв'язку, але затримка або повна втрата повідомлення є рідкісним явищем, зазвичай це стосується менше 5 відсотків повідомлень. Деякі постачальники дозволяють користувачам запитувати звіти про доставку або через налаштування SMS більшості сучасних телефонів, або додаючи до кожного повідомлення *0# або *N#. В звітах про доставку бувають різні статуси отримання повідомлення, а саме відправлено, доставлено, не доставлено, відхилено, помилка.

Під час передачи коротких повідомлень між SMSC і абонентом використовується протокол SS7[1] Повідомлення надсилаються за допомогою операцій MAP MO та MT-ForwardSM, довжина корисного навантаження яких обмежена протоколом, а саме рівно 140 байтами або 1120 біт.

Короткі повідомлення можна кодувати різними алфавітами:

- стандартним 7-бітним алфавітом GSM
- 8-бітним алфавітом даних
- 16-бітним алфавітом UCS-2.

Залежно від налаштуваної мови в телефоні абонента, це призводить до максимальних розмірів окремих коротких повідомлень 160 7-бітних символів, 140 8-бітних символів або 70 16-бітних символів. Підтримка 7-розрядного алфавіту GSM є обов'язковою для GSM-телефонів і мережевих елементів, але символи таких мов, як китайська, арабська, японська, корейська, хінді або кириличні мови (наприклад, українська, болгарська тощо) повинні бути закодовані за допомогою 16-бітного кодування UCS-2. Дані маршрутизації та інші метадані є додатковими до розміру корисного навантаження

Більший вміст (сегментовані SMS або «довгі SMS») можна надсилати за допомогою кількох повідомлень, у даному випадку повідомлення міститимуть в собі заголовка даних користувача (UDH), який містить інформацію про сегментацію. Оскільки UDH є частиною корисного навантаження, кількість доступних символів на сегмент менша:

153 для 7-бітного кодування, 134 для 8-бітного кодування та 67 для 16-бітного кодування. Потім в свою чергу мобільний пристрій складання повідомлення та представлення його користувачеві як одне довге повідомлення. У той час як стандарт теоретично дозволяє до 255 сегментів, 10 сегментів є практичним максимумом для деяких операторів , і довгі повідомлення часто виставляються як еквівалентні кільком SMS-повідомленням. У деяких випадках підтримується 127 сегментів, але обмеження програмного забезпечення в деяких програмах SMS не дозволяють це зробити. Деякі провайдери пропонують орієнтовані на довжину схеми ціноутворення для повідомлень, хоча така структура ціноутворення швидко зникає.

Потокові SMS – це візуальна орієнтація стилю історії SMS-повідомлень, яка впорядковує повідомлення до та від абонента в хронологічному порядку на одному екрані.

На відміну від спеціальних текстових систем, таких як Simple Network Paging Protocol і протокол ReFLEX від Motorola доставка SMS-повідомлень не гарантується, і багато реалізацій не забезпечують механізму, за допомогою якого відправник міг би визначити, чи було SMS-повідомлення доставлено вчасно. SMS-повідомлення, як правило, розглядаються як трафік з нижчим пріоритетом, ніж голосовий, і різні дослідження показали, що приблизно від 1% до 5% повідомлень втрачається повністю, навіть у нормальних умовах роботи, а інші можуть бути доставлені лише через довгий час після того, як їх релевантність пройде. Зокрема, піддається сумніву використання SMS як служби екстреного сповіщення.

Глобальна служба мобільного зв’язку (GSM), яка має найбільшу кількість користувачів у світі, являється дуже уразливою. У GSM тільки повітряний трафік між мобільною станцією (MS) і базовою приймально-передавальною станцією (BTS) додатково шифрується за допомогою слабкого та зламаного потокового шифру (A5/1 або A5/2). Автентифікація

є односторонньою і також вразливою. Існує також багато інших вразливостей і недоліків. Такі вразливості притаманні SMS як одній із кращих і добре перевірених послуг із глобальною доступністю в мережах GSM. Обмін SMS-повідомленнями має деякі додаткові вразливості через функцію зберігання та пересилання, а також проблему фальшивих SMS, які можна відправляти через Інтернет. Коли користувач перебуває в роумінгу, вміст SMS-повідомлень проходить через різні мережі і піддається різноманітним уразливостям і атакам. Ще одна проблема виникає, коли зловмисник отримує доступ до телефону та читає попередні незахищені повідомлення.

Центр обслуговування коротких повідомлень (Short Message Service Center, SMSC) — це один з послідовних елементів стільникової мережі. Повне позначення SMSC відповідно до 3GPP – це служба коротких повідомлень – центр обслуговування (Short Message Service - Service Center).

Функції які виконує SMSC можна описати так:

- Прийом текстових повідомлень (SMS) від користувачів стільникової мережі
- Зберігання текстових повідомлень
- Пересилання текстових повідомлень
- Доставка текстових повідомлень (SMS) користувачам стільникової мережі
- Ведення унікальних позначок часу в текстових повідомленнях

Коли користувач надсилає текстове повідомлення (SMS) іншому користувачеві, це повідомлення зберігається в SMSC, який доставляє його користувачеві призначення, коли він буде доступний. Це варіант зберігання та пересилання.

SMSC відповідає за обробку SMS-операцій бездротової мережі:

- Коли SMS-повідомлення надсилається з мобільного телефону, воно спочатку надходить до SMS-центру.

- Потім SMS-центр пересилає SMS-повідомлення до пункту призначення.
- Основним обов'язком SMSC є маршрутизація SMS-повідомлень і регулювання процесу. Якщо одержувач недоступний (наприклад, коли мобільний телефон вимкнено), SMSC збереже повідомлення.
- Він пересилає SMS-повідомлення, коли одержувач доступний і термін дії повідомлення не перевищено.

SMSC може використовувати для взаємодії з іншими додатками, наприклад, Excel таблиця може взаємодіяти з SMSC, дозволяючи надсилати SMS-повідомлення з електронної таблиці Excel.

SMS-повідомлення тимчасово зберігається в SMS-центрі, якщо мобільний пристрій абонента недоступний. На більшості мобільних телефонів можна вказати термін дії, після якого SMS-повідомлення буде видалено з SMS-центру. Після видалення SMS-повідомлення більше не буде доступним для надсилання на мобільний пристрій абонента (навіть якщо він перейде в активний режим в мережі).

Також однією з головних задач SMSC – є функція маршрутизації повідомлень не лише остаточним абонентам стільникової мережі, а й також маршрутизація повідомлень до інших SMSC в іншій мережі та активація певних правил для того чи інакшого користувача, наприклад: переадресація повідомлення на інший номер абонента, тобто SMSC отримує повідомлення, переглядає правила задані користувачем (як і отримувачем так і тим хто надсилає) і в залежності від них може замінити номер отримувача, змінити ім'я абонента який надсилає повідомлення, заміна тексту повідомлення (часткова або повна). У випадку часткової заміни можуть бути знайдені, певні ключові слова, які не допустимі для надсилання, або ж взагалі повна відміна надсилання повідомлення абоненту стільникової мережі.

1.2. Аналіз систем посилання SMS повідомлень абонентам стільникової мережі

Attentive Mobile - це найповніше рішення для маркетингу текстових повідомлень. Вони також стверджують, що створюють «в середньому 20,5% загального онлайн-доходу для сучасних брендів електронної комерції». Вони пропонують усі необхідні функції для повної платформи маркетингу текстових повідомлень.

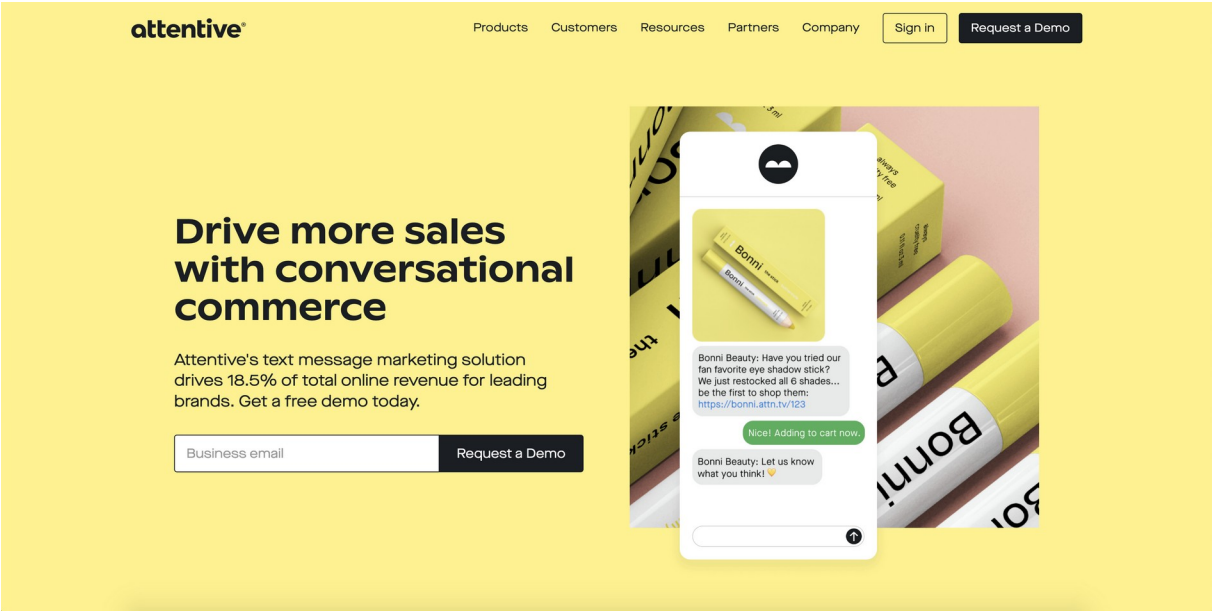


Рисунок 1.1 - Attentive Mobile

Attentive — це комплексна платформа обміну текстовими повідомленнями, орієнтована на розвиток електронної комерції. Вони мають чудові функції в шести різних категоріях:

Зростання аудиторії: стосується створення та розширення вашого списку отримувачів SMS і електронної пошти. Ви можете створити форми реєстрації та рекламувати їх на своєму веб-сайті для максимальної конверсії. Створюйте форми та відображайте їх як спливаючі вікна на своєму веб-сайті (мобільні та настільні версії). Показуйте посилання для реєстрації через облікові записи соціальних мереж.

Менеджер аудиторії: містить набір інструментів для персоналізованого націлювання на аудиторію. Сегментуйте аудиторію за різними критеріями. Поведінка на сайті, історія веб-перегляду та покупок, а також демографічні дані.

Обмін повідомленнями: стосується розширених функцій обміну повідомленнями. Ви можете створювати повідомлення про маркетингові кампанії і надсилати автоматичні повідомлення про транзакції. Вони також пропонують функції A/B тестування та аналітику в реальному часі, щоб вирішити, що вам найкраще підходить.

Бізнес-аналітика: відноситься до їхніх функцій аналітики та звітності. Ви можете створювати звіти про ефективність маркетингових кампанії, ефективність автоматизованих повідомлень, конверсії.

Інтеграція: мають велику кількість інструменітв, які можна інтегрувати з Attentive. Вони пропонують інтеграцію з популярними платформами електронної комерції, такими як Shopify, BigCommerce або Magento. Їхня нещодавня інтеграція Zapier або їхній API дозволяють підключити будь-який інструмент до Attentive.

Відповідність і доставляємість: вони обіцяють високу доставляємість повідомлень і дотримання правил. Вони також пропонують інструменти, за допомогою яких можна переконатися, що ваші форми реєстрації відповідають нормам США

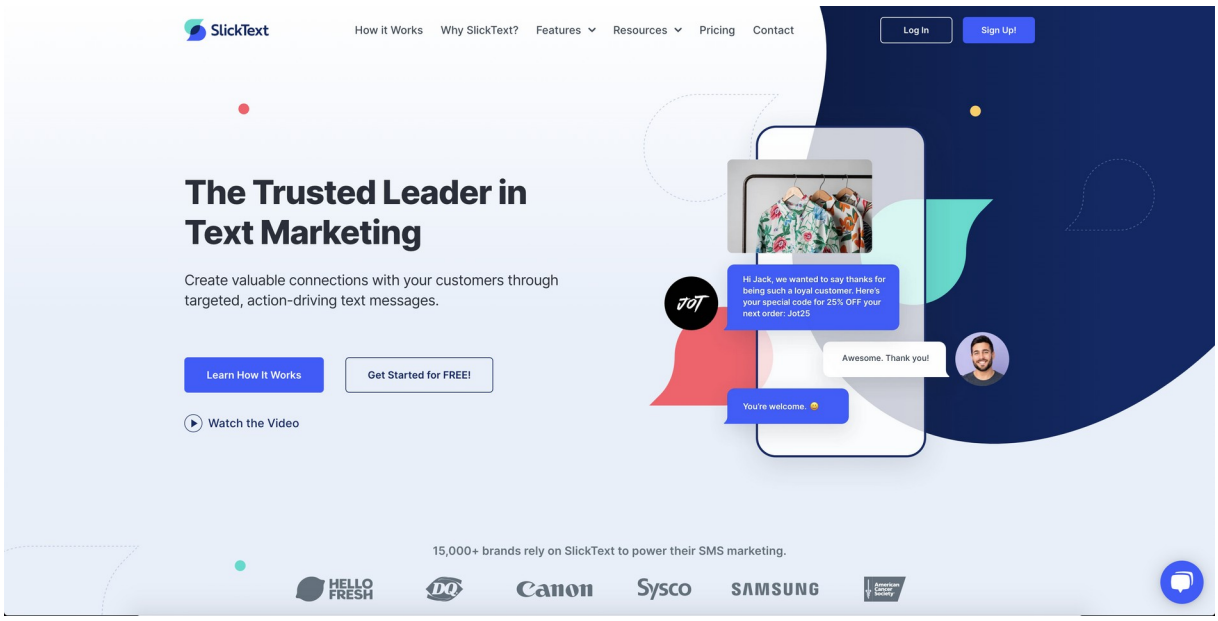


Рисунок 1.2. SlickText

SlickTexting – застосунок, орієнтований на бізнес будь якого розміру та будь яких потреб. Можливості SlickTexting, які виділяють додаток з поміж інших :

Текстові опитування: Користувач може налаштувати опитування та анкети на інформаційній панелі. SlickText пропонує аналітику та звіти для опитувань. Таким чином, користувачу надається можливість проаналізувати відгуки клієнтів. Це важливий крок для забезпечення чудового досвіду клієнтів. 89% клієнтів хочуть, щоб компанії запитували у них відгуки користувачів. Для клієнтів це знак того, що ви готові слухати та вчитися.

Автоматизовані робочі процеси: SlickTexting пропонують чудові функції робочого процесу. Користувач можете створювати автоматизовані послідовності повідомлень, які залучатимуть потенційних клієнтів і підвищуватимуть їх якість.

Детальна аналітика дає користувачу такі показники, як рейтинг кліків. Користувач також може переглядати в режимі реального часу, як його клієнти взаємодіють з SlickTexting.

Подвійна згода: користувач можете вимагати від своїх клієнтів підтвердити свою підписку, надіславши «Так». Або подібні повідомлення підтвердження. Це корисний функціонал, щоб переконатися, що у вас є чистий список телефонних номерів і уникнути додаткових витрат або спаму.

Списки абонентів, що самоочищаються: система виявляє недійсні номери та видаляє їх.

100% відповідність операторам і CTIA (Cellular Telecommunications Industry Association).: вони гарантують доставку повідомлень. Вони також підтримують відповідність операторам стільникового зв'язку та CTIA.

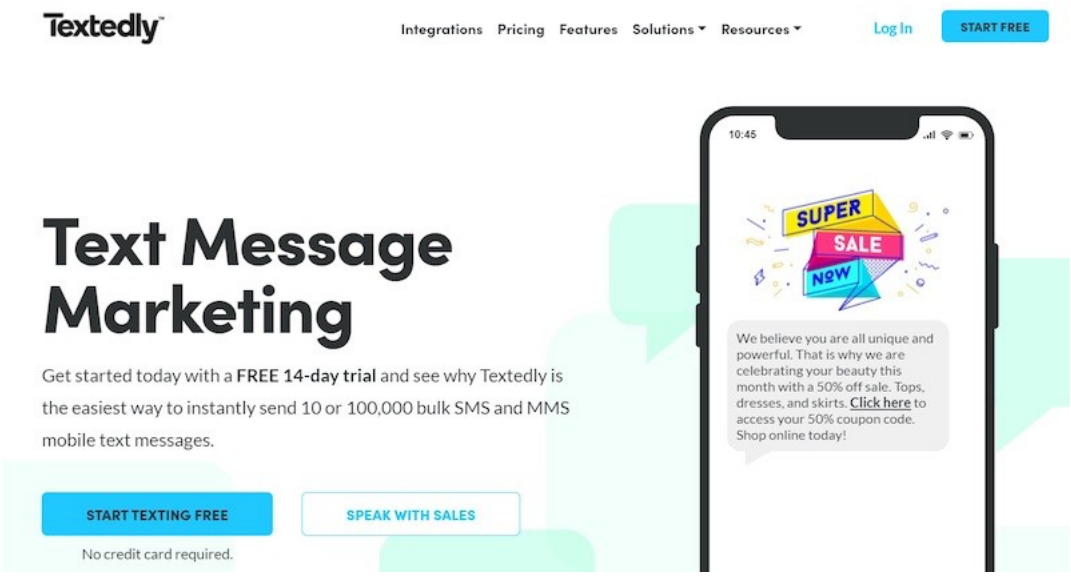


Рисунок 1.3. Textedly

Textedly — це ще одна популярна маркетингова платформа текстових повідомлень із різноманітними чудовими функціями. Textedly має всі стандартні функції найкращих маркетингових SMS платформ. Доступні всі стандартні функції, такі як:

- Масове розсилання текстових повідомленнь.
- Двосторонній обмін повідомленнями.
- Автовідповідачі.

- Маркетингові кампанії.
- Комплексна аналітика
- MMS.

Деякі з інших важливих функцій:

Голосові послуги: користувач може перетворити номер телефону з якого відправлялось повідомлення на call center. Таким чином ваші клієнти можуть легко зателефонувати на той самий номер, на який вони надіслали повідомлення, якщо їм це зручно.

Вхідні MMS-повідомлення: ваші абоненти можуть надсилати вам мультимедійні повідомлення так само, як і користувач їм.

Надсилання текстових повідомлень в Instagram: Textedly добре інтегровані з Instagram. Користувач можете додати кнопку «Надіслати нам повідомлення» у свій профіль Instagram. Це дозволить клієнтам надіслати вам повідомлення, коли вони натиснуть цю кнопку.

Інтеграція з сторонніми сервісами: Textedly надає пряму інтеграцію з популярними платформами маркетингу та продажів. Крім того, користувач можете використовувати Zapier для інтеграції з більшою кількістю програм сторонніх розробників. Наприклад, користувач може підключити Textedly до Jotform. Отже, коли хтось надсилає дані до ваших налаштованих веб-форм Jotform, він додається до вашого списку клієнтів.

Збір даних текстових повідомлень: платформа самостійно збирає та опрацьовує данні про ваших клієнтів, а саме : ім’я, фамілія, адресу проживання, та надає можливість використовувати їх при наступному розсиланні SMS повідомлень.

Підтримка кількох користувачів: можливість додавати стільки користувачів, скільки вам потрібно, до свого облікового запису безкоштовно. Це функція для спільного використання вашого облікового запису з членами вашої команди.

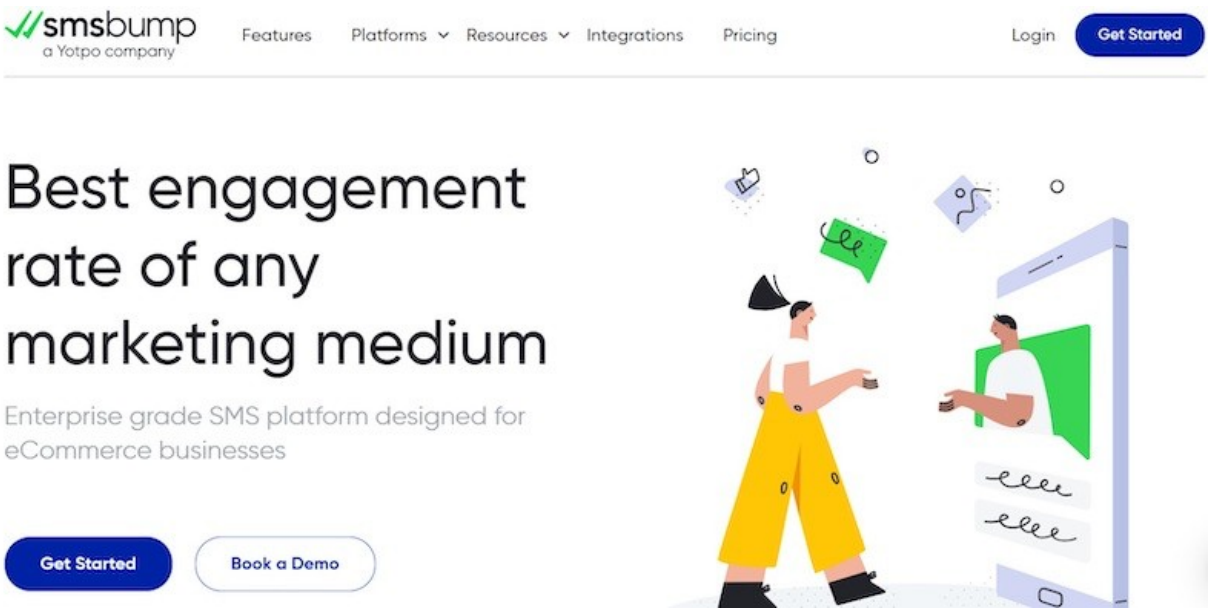


Рисунок 1.4. *SMSBump*

Основні функції SMSBump:

SMSBump надає користувачу багато функцій для створення та розширення списків: вони пропонують багато варіантів для розширення ваших списків. До них належать спливаючі вікна, плаваючі кнопки, вбудовані форми, форми нижнього колонтитула, підключення до соціальних мереж.

Автоматизовані потоки SMS: це послідовність автоматизованих дій на основі різних тригерів. Надсилайте повідомлення після таких дій, як нова реєстрація, замовлення, підтвердження доставки.

A/B-тестування: перш ніж використовувати будь-які повідомлення у своїй маркетинговій кампанії, ви можете запустити їх A/B-тестування, щоб підібрати найкращий текст з найбільшою конверсією.

Планування кампанії: можливість надсилати свої повідомлення в певну дату та час за допомогою Планувальника кампанії.

Міжнародне надсилання: можливість надсилати SMS до США, Канади, Великобританії, Австралії та Франції. MMS-повідомлення доступні лише в США.

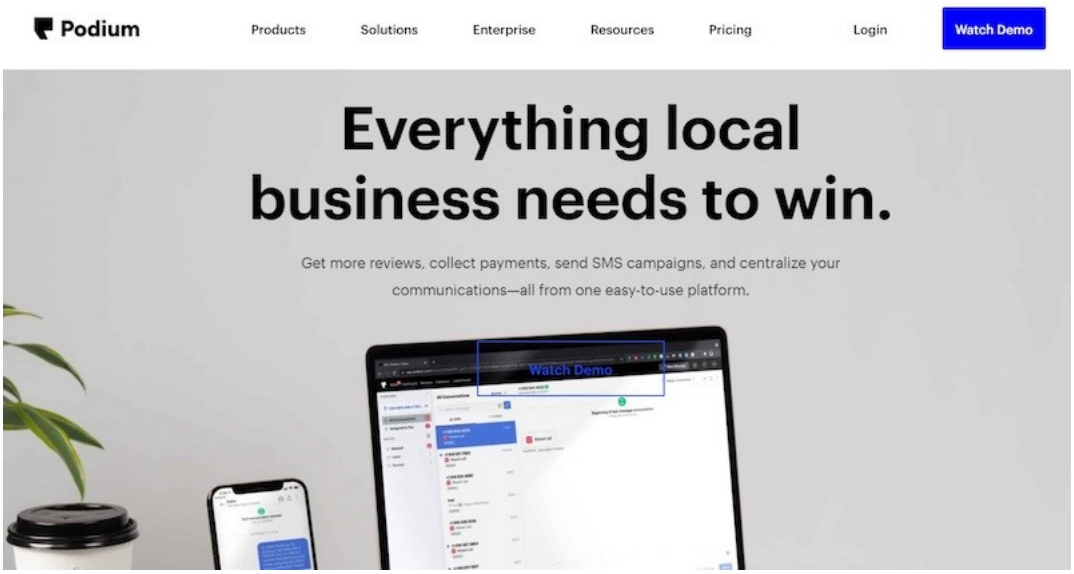


Рисунок 1.5. Podium

Особливості Podium:

Багатоканальний обмін повідомленнями: за допомогою папки «Вхідні» Podium користувач може керувати всіма повідомленнями з різних джерел в одному місці. Він переносить усі повідомлення з Google, Facebook, Instagram, iMessage, вашого веб-сайту, електронну пошту та текстові повідомлення у папку "Вхідні".

Веб-чат: користувач може отримувати повідомлення від людей і відповідати на них на своєму веб-сайті. Вони повинні заповнити свою контактну інформацію, щоб надіслати вам повідомлення, щоб це могло збільшити ваш список контактів.

Командний чат: додаток надає користувачу можливість спілкуватися з товаришами по команді в рамках їхньої платформи обміну повідомленнями. Ви також можете призначати завдання членам команди.

Відеочати: користувачу надається можливість безпечно спілкуватись через відеодзвінок зі своїми клієнтами, надіславши їм посилання за допомогою текстового повідомлення.

Podium також надає користувачам голосовий сервіс, щоб вони також могли надсилати голосові повідомлення своїм клієнтам.

Інструменти створення списку контактів: за допомогою їхніх інструментів користувач може створювати та розширювати свій список контактів. Є різні варіанти, а саме корисутвач може створювати форми та спливаючі вікна для свого веб-сайту, чат свого веб-сайту, інтегрувати сторінку реєстрації в кошик для виписки або POS-систему або навіть запитувати контактну інформацію своїх клієнтів після того, як вони залишать відгук.

Text-to-pay: можливість надсилати клієнтам запит на оплату за допомогою текстових повідомлень. Також клієнтам надається гнучкість для оплати кредитом, банківськими переказами, Google Pay або Apple Pay.

1.3. Визначення вимог до системи надсилання коротких повідомлень користувачам стільникового зв'язку

На рис. 1.6 відтворена схема взаємодії модулів, які використовуються під час надсилання SMS повідомленнь.

У межах магістерського дипломного проєкту будуть запроектовані та відтворені наступні модулі з графічної схеми (рис. 1.6):

- *MW-FRONT* – модуль, який відповідає за графічне відтворення інтерфейсу користувача одноканальних відправлень.
- *OMNI-FRONT* – модуль, який відповідає за графічне відтворення інтерфейсу користувача двоканальних відправлень
- *API* – REST API модуль для взаємодії з сутностями додатку.
- *SCHEDULER* – модуль для контролю конверсій та надсилання запланованих повідомлень.
- *PRODUCER* – модуль для збирання метадати повідомлень, її опрацювання та подальшого відправлення.
- *STATISTIC* – модуль для збору звітів про доставку повідомлень та генерацію аналітичних звітів.
- *Link Shortener* - модуль для генерації та опрацювань посилань.

- *NATS* – модуль, який виконує роль меседж брокера між *PRODUCER* та *TRX*
- *TRX* – модуль, який на пряму взаємодіє з будь яким вендором або SMSC, через їх API

В таблицях 1.1 та 1.2 наведені основні функціональні вимоги, що ставляться до системи.

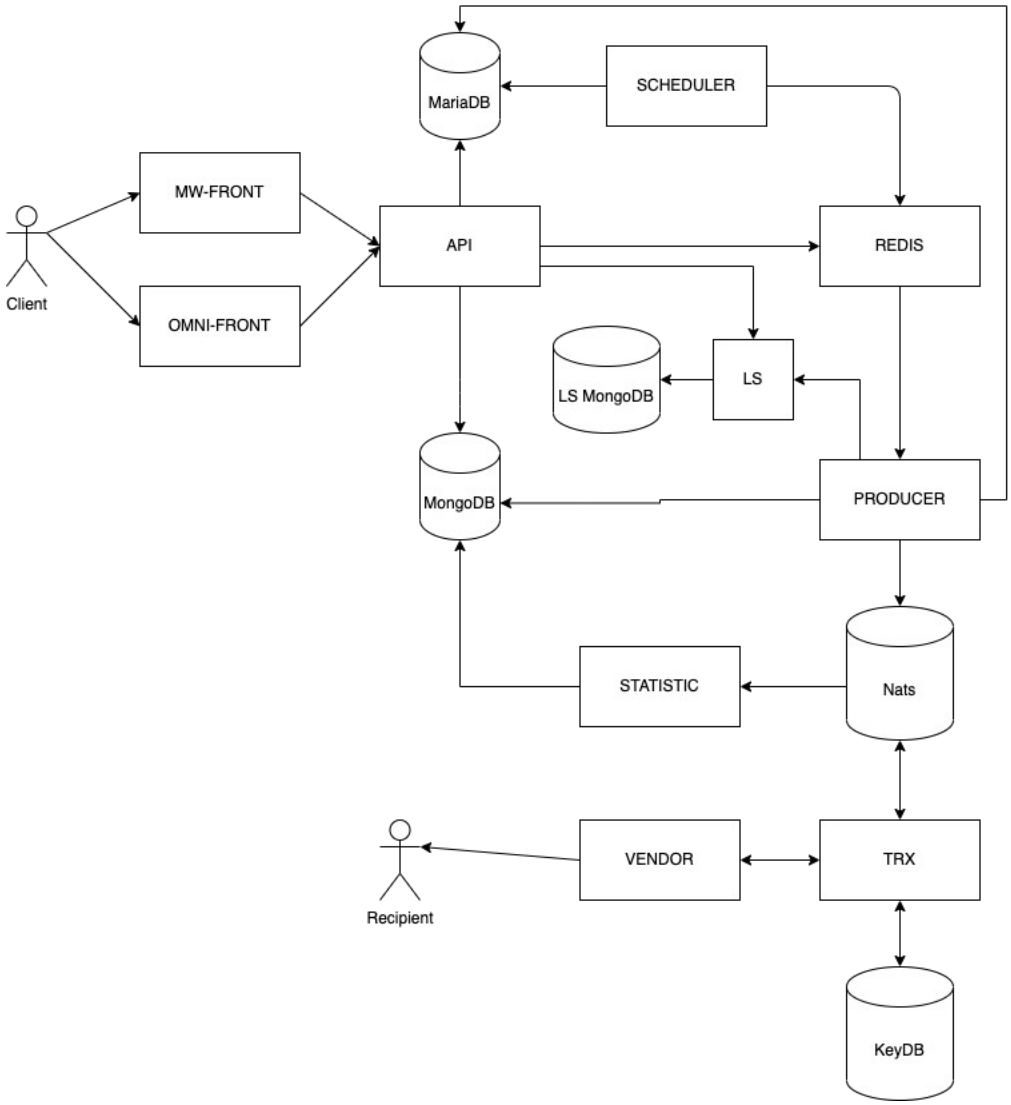


Рисунок 1.6. Графічна схема системи посилання SMS повідомлень

Таблиця 1.1 - Функціональні вимоги до серверної частини системи

Кодовий номер	Вимога

В1	Можливість надсилати SMS абонентам стільникової мережі

Продовження табл. 1.1

В2	Можливість створювання абонентських списків
В3	Можливість надсилати посилання абонентам
В4	Можливість переглядати звіти про надсилання повідомлень абонентам
В5	Безперервна робота програмного забезпечення
В6	Можливість налаштовувати ім’я відправника
В7	Можливість відправлять відкладені повідомлення
В8	Можливість створювати одночасно відправлення повідомлень багатьом абонентам
В9	Можливість спілкуватись з користувачем через двоканальне відправлення
В10	Можливість надсилати MMS
В11	Можливість для абонентів відписуватись від прийняття повідомлень
В12	Можливість створювати персоналізовані повідомлення
В13	Можливість автоматизації процесів відправлення, в залежності від розміру конверсії
В14	Можливість генерувати аналітику маркетингових компаній

Таблиця 1.2 Функціональні вимоги до клієнтських частин системи

Кодовий номер	Вимога
В15	Можливість перегляду аналітичних звітів про відправлення
В16	Можливість перегляду списків звітів отримання повідомлень
В17	Можливість реєстрації та авторизації у додатку

Продовження табл. 1.2

В18	Можливість надсилати заплановані повідомлення
В19	Можливість надсилати повідомлення

Таблиця 1.3 Матриця залежності функціональних вимог до системи

	В1	В2	В3	В4	В5	В6	В7	В8	В9	В10	В11	В12	В13	В14	В15	В16	В17	В18	В19
В1	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
В2		X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
В3			X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
В4				X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
В5					X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
В6						X	X	X	X	X	X	X	X	X	X	X	X	X	X
В7							X	X	X	X	X	X	X	X	X	X	X	X	X
В8								X	X	X	X	X	X	X	X	X	X	X	X
В9									X	X	X	X	X	X	X	X	X	X	X
В10										X	X	X	X	X	X	X	X	X	X
В11											X	X	X	X	X	X	X	X	X
В12												X	X	X	X	X	X	X	X
В13													X	X	X	X	X	X	X

В 14														X	X	X	X	X	X
В 15															X	X	X	X	X

Продовження табл. 1.3

В 16																X	X	X	X
В 17																	X	X	X
В 18																		X	X
В 19																			X

Дивлячись на матрицю залежності, наведену в табл. 1.3, можемо зробити висновок щодо вимог : вони не суперечать одна одній.

Висновки

У даному розділі було зроблено аналіз процесу посилання коротких повідомлень абонентам стільникового зв’язку. Також було проаналізовані подібні системи надсилання SMS повідомлень стільниковою мережею. Зроблено порівняння їх функціональних особливостей та можливостей, знайдено переваги та недоліки. В результаті проведеного аналізу були сформовані функціональні вимоги до системи надсилання коротких повідомлень стільниковою мережею.

2. КОНЦЕПТУАЛЬНА МОДЕЛЬ СИСТЕМИ

2.1. Формалізування процесу відправки SMS повідомлень абонентам стільникової мережі

Шоб уникненути ризиків та проблема, котрі, зазвичай, виникають під час розроблення програмних модулів, змоделюємо предметну царину, для того щоб описати та формалізувати її. Для функіонального моделювання процесу буде використано метод *IDEF0*[6].

IDEF0, або ж Function Modeling – це методологія функціонального моделювання різних бізнес процесів. Процес буде представлено у вигляді графічної нотації, яка потім буде використана як основа для створення функціональних моделей. Дана графічна нотація відображає функції та структуру системи. Головною особливістю методології IDEF0 є те що вона також дає розуміння всіх потоків інформації та материальних речей, які пов’язують ці функції. Методологія IDEF0 вважається новим революційним кроком розвитку мов декларування функціональних систем. Стандарт *IDEF0* був розроблений департаментом ВПС (військово-повітряних сил) США в 1981 році.

Особливості *IDEF0*:

- Контекстна схема - найвища схема в моделі IDEF0.
- Ієрархія декомпозиції IDEF0 з використанням батьківських та дитячих зв'язків.
- Дерева вузлів - структури вузлів у вигляді дерева, кореневі на вибраному вузлі і використовуються для представлення повної декомпозиції IDEF0 в одній схемі.

На рис. 2.1. зображена діаграма, яка показує абстрактний опис системи, на якому відтворено діяльність надсилання SMS повідомлень абонентам стільникової мережі,а саме:

- вихідні дані.
- вхідні дані.

- ресурси(матеріальні та не матеріальні).
- механізми управління.

.

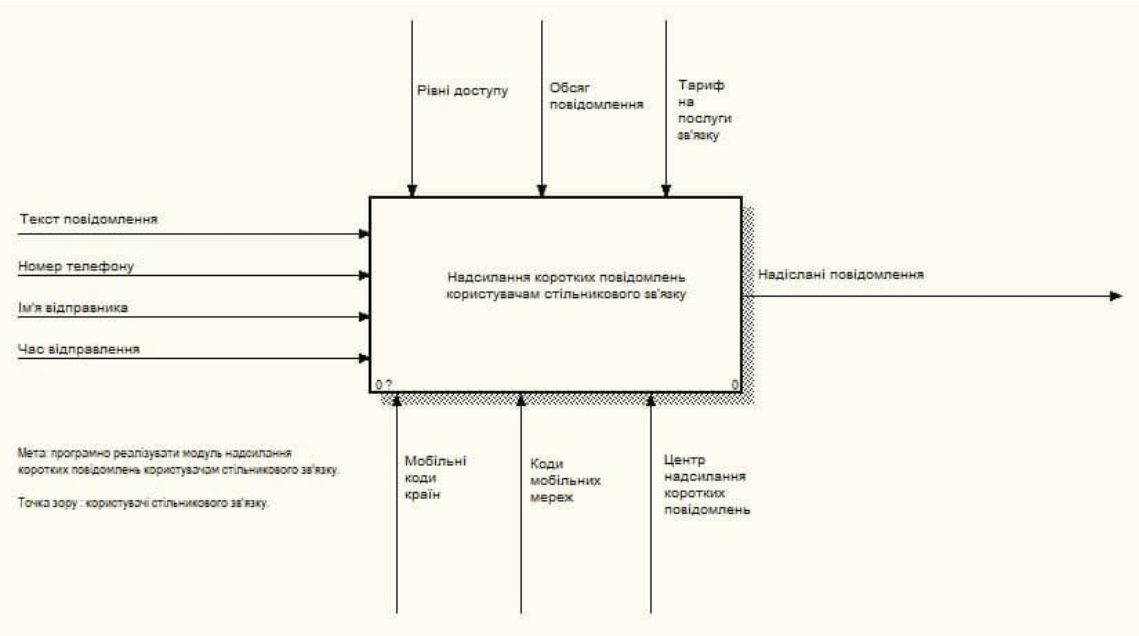


Рисунок 2.1. Функціональна модель системи

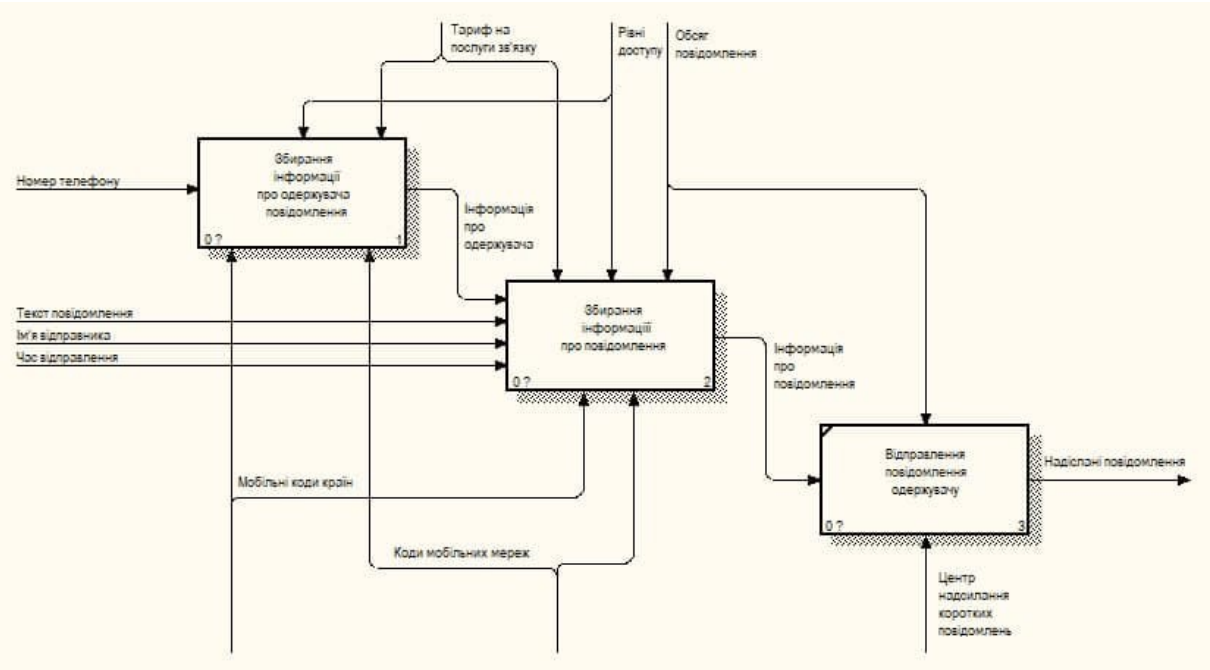


Рисунок 2.2. Декомпозиція функціональної моделі системи

Згідно з структурою системи надсилання коротких повідомлень стільниковою мережею, яка була проаналізовна у підрозділі 1.1, відображено його головні функціональні можливості.

На рис. 2.2 зображена модель процесів та послідовності їх виконання при роботі системи:

- Збирання інформації про абонента стільникової мережі – на даному етапі процесу відбувається збір повної інформації про абонента та перевіряється його MCC (Mobile country code) і MNC (Mobile network code).
- Збирання інформації про текстове повідомлення – на даному етапі процесу відбувається збір даних про саме повідомлення,а саме : персональна інформація абонента, для створення персоналізованого тексту, знаходження посилань в тексті та заміна їх на скорочені та сегментація самого повідомлення за її потреби
- Відправлення повідомлення користувачу – на останньому кроці виконується надсилання коротких повідомлень до SMSC з всією зібраною інформацією під час виконання попередніх етапів.

Вхідними даними є :

- Номер телефону – особистий номер абонента стільникової мережі.
- Час відправлення – орієнтовний час в який система надішле повідомлення користувачу.
- Ім’я відправника – ім’я абонента від якого користувач отримає повідомлення.
- Текст повідомлення – сформований і персоналізований текст, який отримає абонент.

Також на схемі відображені механізми для роботи системи надсилання SMS повідомлень абонентам стільникової мережі:

- Коди мобільних мереж (MNC) – механізм, який визначає мобільного оператора та його мережу, якою користується абонент.
- Мобільні коди країн (MCC) – механізм, який визначає країну походження мережі та де вона використовується.
- Центр посилання SMS повідомлень (SMSC) – механізм, який бере повідомлення в опрацювання та відповідає за подальше відправлення його на мобільний пристрій абонента, отримання звітів про доставлення.

Вихідними даними системи надсилання коротких повідомлень абонентам стільникової мережі є лише «Надіслане повідомлення» – це текстове повідомлення надіслане абоненту.

На рис. 2.1 та 2.2 зображені обмеження та умови, які є зовнішніми правила системи надсилання SMS повідомлень абонентам стільникової мережі.

На рис. 2.3 зображена діаграма у графічній нотації IDEF0, що є декомпозицією другого блоку процесу «Збирання інформації про повідомлення». Дана декомпозиція допомагає описати процес надсилання SMS повідомлень абонентам стільникової мережі більш глибше, точніше та детальніше.

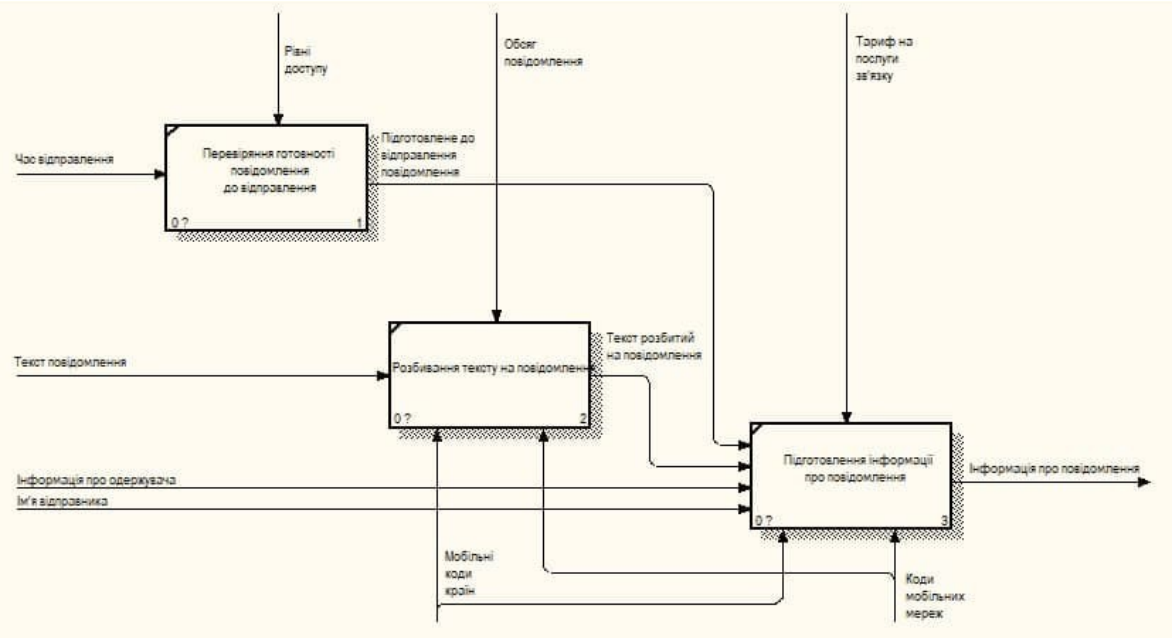


Рисунок 2.3. Декомпозиція другої функції з процесу «Збирання інформації про повідомлення»

Дана декомпозиція дає ширше бачення процесів, які лежать в основі діяльності системи надсилання SMS повідомлень абонентам стільникової мережі.

Вхідні дані процесу :

- Номер телефону – особистий номер абонента стільникової мережі.
- Час відправлення – орієнтовний час в який система надішле повідомлення користувачу.
- Ім'я відправника – ім'я абонента від якого користувач отримає повідомлення.
- Текст повідомлення – сформований і персоналізований текст, який отримає абонент.

Також на рис. 2.3 відображені механізми роботи декомпонованого процесу «Збирання інформації про повідомлення» системи надсилання SMS повідомлень абонентам стільникової мережі:

- Коди мобільних мереж (MNC) – механізм, який визначає мобільного оператора та його мережу, якою користується абонент.
- Мобільні коди країн (MCC) – механізм, який визначає країну походження мережі та де вона використовується.

Вихідними даними є :

- Інформація про повідомлення – сформовані дані, які були агреговані з різних систем.

Також на діаграмі зображеній на рис. 2.3 відображені основні процеси, які беруть участь у роботі системи:

- На першому етапі перевіряється час відправки повідомлення і виконуються відповідні кроки.
- На другому етапі відбувається персоналізація повідомлення, тобто заміна токенів з повідомлення на персональну інформацію абонента та розбиття повідомлення на сегменти в залежності від розміру повідомлення яке буде надсилатись.
- На третьому етапі агрегується інформація про абонентів та саме повідомлення.

2.2. Формулювання зв'язків між етапами процесу посилання SMS повідомлень абонентам стільникової мережі

Для формулювання зв'язків між етапами процесу надсилання SMS повідомлень абонентам стільникової мережі було використано графічну нотацію IDEF3 – методологія моделювання процесів, яким надається можливість збору інформації про процес та подальше його документування.

IDEF3 або Integrated DEFinition for Process Description Capture Method — це метод моделювання бізнес-процесів, який доповнює IDEF0, сам IDEF3 являється теж графічною нотацією для опису різних бізнес процесів. Під час побудови графічної нотації IDEF3 застосовується

організайційний метод, що надає експерту в даній предметній галузі представити стан речей, як налогджену черговість подій з синхронним описом об'єктів, які мають пряме відношення до процесу. IDEF3 є гарно налагодженою технологією, для знаходження даних, які потрібні для проведення своєчасного та структурного аналізу системи [7].

У IDEF3 є головна відмінність від інших технологій для моделювання процесів, а саме, відсутність дуже жорстких семантичних та синтаксичних обмежень, що роблять опис не повноцінних та не цілісних систем не комфортним. Також одним з найголовніших плюсів цієї графічної нотації є те що, аналітик, який її розробляє не повинен за необхідності змішувати свої власні здогадки щодо функціювання бізнес процесів з експертними твердженнями для заповнення пропусків в описі предметної галузі [7]. IDEF3 надає такі засоби: моделювання та документування, що дозволяють виконувати наступні дії:

- Документувати технології які наявні в процесі.
- Створювати імітаційні моделі процесів, які розглядаються.
- Аналізувати вплив потоків документообігу на різні сценарії процесів.
- Впливають на прийняття, насамперед, безумовно кращих рішень під час організації процесів.

На рисунку 2.4. система представлена у вигляді упорядкованої послідовності бізнес процесів, які здійснюються під час надсилання SMS повідомлень абонентам стільникової мережі.

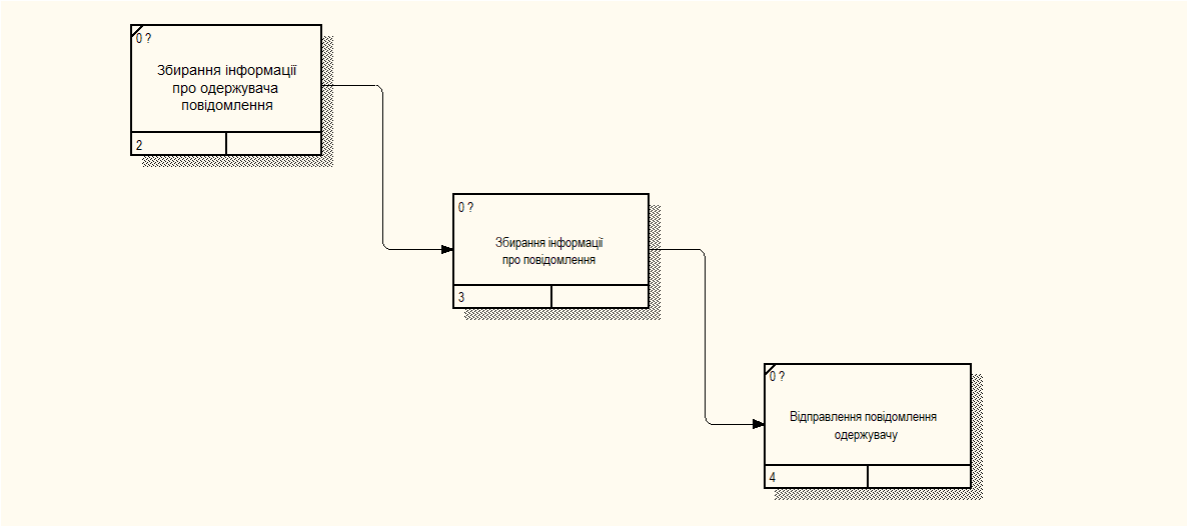


Рисунок 2.4. Декомпозиція діаграми процесів в нотації IDEF3

На рис. 2.4 зображені зв’язки між процесами, що здійснюються під час надсилання SMS повідомлень абонентам стільникової мережі , а саме:

- «Агрегація інформації про абонента повідомлення».
- «Агрегація інформації про саме повідомлення».
- «Надсилання повідомлення одержувачу».

Оскільки, основні дії з даними здійснюються в блоці «Агрегація інформації про абонента повідомлення», а саме агрегація та опрацювання, тому було прийнято рішення декомпозувати в нотації IDEF3 саме цей блок.

Процес відображений на рисунку 2.4 «Агрегація інформації про абонента повідомлення» обробляє дані абонента стільникової мережі та дані про SMS повідомлення. На наступному етапі посилання коротких повідомлень система отримує інформацію з черги повідомлень та всю прилеглу інформацію з баз даних. Наступним кроком система фільтрує абонентів стільникової мережі та робить запит до підсистеми створення посилань, за умови наявності токена з посиланням в тексті повідомлення, для їх генерації, з метою заощадження корисного навантаження повідомлення. У випадку, коли підсистема створення посилань не змогла згенерувати посилання, спрацьовує процес повторного опрацювання

повідомлення і воно повертається до черги. Цей процес буде відбуватись до моменту відправки повідомлення або ж до досягнення ліміту повторних відправлень повідомлень. Коли підсистема створення посилань замінює текст то наступним відбувається процес генерації задач для відправлення повідомлень та запис задачі до SMSC.

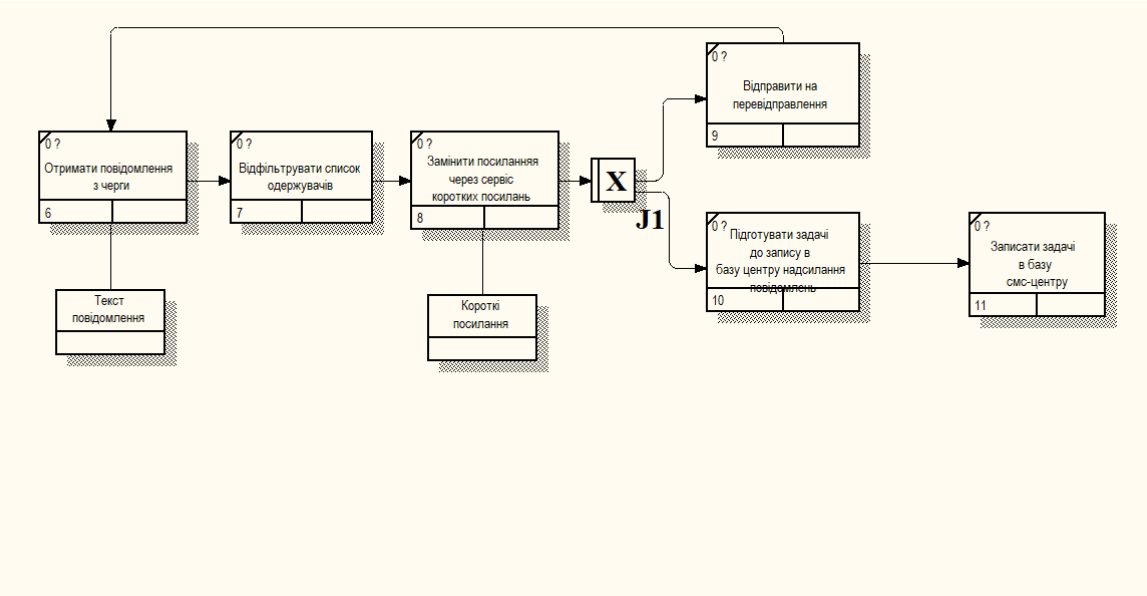


Рисунок 2.5. Декомпозиція процесу «Агрегація інформації про саме повідомлення»

Останнім етапом відправлення повідомлень є відправлення повідомлення самим SMSC до абонента стільникової мережі.

2.3. Формулювання потоків інформації поміж етапами посилання SMS повідомлень абонентам стільникової мережі

Діаграма DFD наочно відображає перебіг інформації в межах процесу чи системи. Для зображення вхідних та вихідних даних, точок зберігання інформації та шляхів її пересування між джерелами та пунктами доставки у таких діаграмах застосовуються стандартні фігури, такі як прямокутники та кола, а також стрілки та короткі текстові мітки.

Діаграми DFD варіюються від найпростіших нарисів процесів (включаючи намальовані вручну) до детальних багаторівневих схем з глибоким аналізом способів обробки даних.

Діаграми DFD застосовуються для аналізу існуючих та моделювання нових систем. У кращих традиціях візуалізації даних діаграми DFD часто наочно «розповідають» про процеси, які складно пояснити словами, і дозволяють ефективно донести інформацію і до «фізиків», і до «ліриків», тобто до всіх учасників організації від розробників до генеральних директорів. Ось чому діаграми DFD не втратили популярності за довгі роки існування. Однак варто згадати, що хоча діаграми DFD відмінно підходять для програм і систем потоків даних, в наші дні вони далеко не завжди відповідають вимогам програмного забезпечення та систем, орієнтованих на інтерактивність, роботу в реальному часі та бази даних.

Правила та поради щодо побудови діаграм DFD

- Кожен процес має супроводжуватися як мінімум одним вхідним та одним вихідним потоком;
- У кожне сховище повинен впадати щонайменше один потік даних і щонайменше один — витікати;
- Дані, що зберігаються у системі, повинні проходити через процес;
- Кожен процес діаграми DFD повинен вести або до іншого процесу, або до сховища даних.

Діаграми DFD поділяються на дві категорії. Логічні діаграми DFD наочно показують переміщення потоку даних, життєво важливих функціонування організації. У центрі уваги таких діаграм — сам бізнес та необхідна йому інформація, а не те, як працює чи має працювати система. На відміну від них, фізичні діаграми DFD ілюструють саме поточний або планований пристрій системи. Наприклад, у логічній діаграмі DFD як

процеси виступають різні види комерційної діяльності, а у фізичній — програми та ручні процедури

На рисунку 2.6 зображено декомпозицію контекстної діаграми, побудованої до цього, що знаходиться на рисунку 2.1. «Посилання SMS повідомлень абонентам стільникової мережі» являється основним процесом.

Вхідні дані:

- ім’я користувача,
- ідентифікаційний номер абонента.
- дата та час відправлення
- текст,

Вихідні дані

- запис в БД
- повідомлення надіслане абоненту стільникової мережі.

Сховища даних:

- БД абонентів.
- БД текстових повідомлень.
- БД створених посилань.

Функціональні блоки:

- «Агрегація інформації про абонента».
- «Агрегація інформації про текстове повідомлення».
- «Надсилання текстового повідомлення абоненту стільникової мережі».

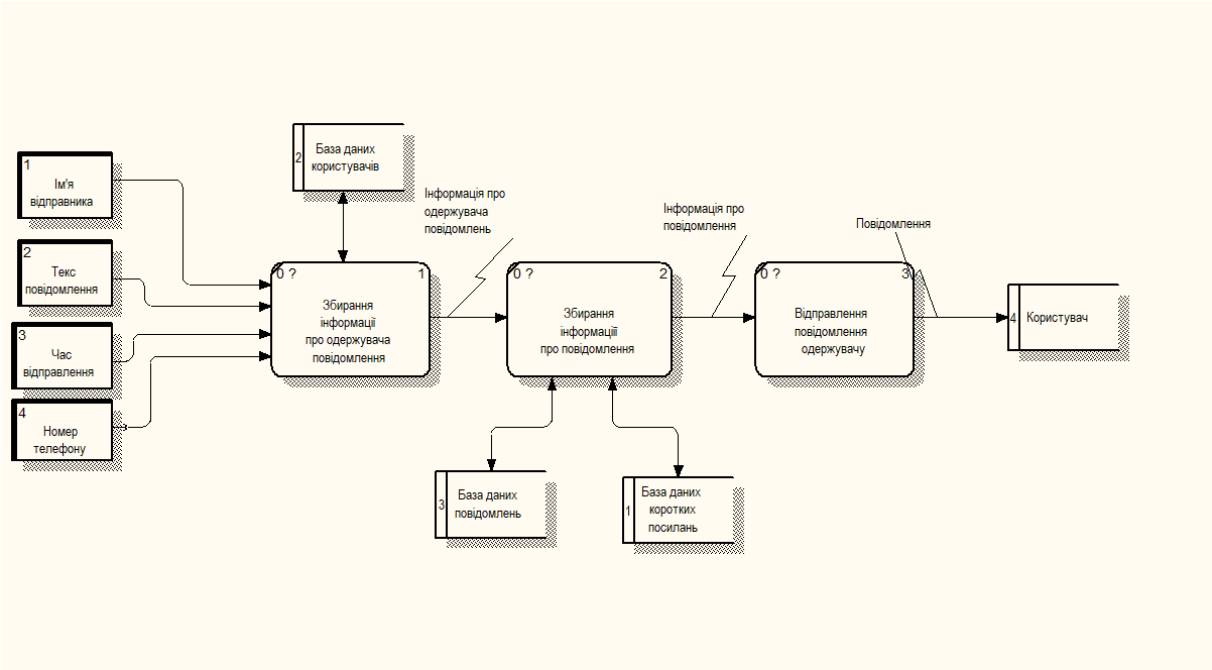


Рисунок. 2.6. Декомпозиція контекстної діаграми в графічній нотації DFD

В результаті виконання описаних вище процесів є посилання текстових повідомлень абоненту стільникової мережі.

Висновки

Завдяки тому що ми формалізували процеси відправки SMS повідомлень абонентам стільникової мережі, сформулювали зв'язки між етапами процесу відправки SMS повідомлень абонентам стільникової мережі та визначили потоки даних між етапами посилання SMS повідомлень абонентам стільникової мережі, ми змогли минути різних ризиків, що зазвичай виникають в момент розробки різних складних систем. Також було покращене розуміння галузі. Це все вдалось зробити за допомогою побудови графічних моделей системи відправки SMS повідомлень абонентам стільникової мережі в нотаціях IDEF0, IDEF3, DFD

В результаті роботи було отримано дані, яких цілком достатньо для побудови об'єктно-орієнтованої моделі системи відправки SMS повідомлень абонентам стільникової мережі

3. ОБ’КТНО-ОРІЄНТОВАНА МОДЕЛЬ СИСТЕМИ

3.1. Формулювання варіантів використання системи

Дійовою особою, котра буде користуватись системою – це користувач та адміністратор. На рис. 3.1. зображені варіанти використання системи користувачем у графічній нотації UML [11] [10].

Рис. 3.1. Варіанти використання системи користувачем

В табл. 3.1 – 3.13. наведено опис усіх варіантів використання користувачем системи зображених на рис. 3.1.

Таблиця 3.1Варіант використання «Авторизація у додатку»

Найменування	Авторизація у додатку
Первинний актор	Користувач
Інші актори	немає
Опис	Можливість авторизуватись у системі
Попередні умови	Наявний вірний логін, пароль та відкрита сторінка авторизації у системі
Вихідні умови	Відкрита сторінка “Dashboard” системи

Таблиця 3.2Варіант використання «Вивантаження списків у форматі CSV»

Найменування	Вивантаження списків у форматі CSV
Первинний актор	Користувач
Інші актори	Немає
Опис	Вивантаження списків номерів абонентів , що отримали або не отримали повідомлення
Попередні умови	Повідомлення відправлене
Вихідні умови	Сформований CSV файл

Таблиця 3.3 Варіант використання «Посилання коротких повідомлень»

Найменування	Посилання коротких повідомлень
Первинний актор	Користувач
Інші актори	Немає

Продовження табл. 3.3

Опис	Посилання коротких повідомлень абонентам стільникової мережі
Попередні умови	Відкрита сторінка «Send Broadcast», номер абонента стільникової мережі, текст повідомлення, наявність достатньої кількості коштів на балансі
Вихідні умови	Відправлене SMS повідомлення, відкрита сторінка «Broadcasts»

Таблиця 3.4 Варіант використання «Завантаження списків у форматі CSV»

Найменування	Завантаження списків у форматі CSV
Первинний актор	Користувач
Інші актори	Немає
Опис	Завантаження CSV списків з інформаціє про абонента
Попередні умови	Відкрита сторінка «Recipients»
Вихідні умови	Створений список абонентів

Таблиця 3.5 Варіант використання «Поповнення балансу»

Найменування	Поповнення балансу
Первинний актор	Користувач
Інші актори	Немає
Опис	Поповнення балансу облікового запису користувача
Попередні умови	Відкрита сторінка «Payments»
Вихідні умови	Баланс облікового запису користувача поповнено

Таблиця 3.6 Варіант використання «Керування текстовими шаблонами»

Найменування	Керування текстовими шаблонами
Первинний актор	Користувач
Інші актори	Немає
Опис	Керування текстовими шаблонами для повідомлень
Попередні умови	Відкрита сторінка «Templates»
Вихідні умови	Новий шаблон

Таблиця 3.7 Варіант використання «Керування посиланнями»

Найменування	Керування посиланнями
Первинний актор	Користувач
Інші актори	Немає
Опис	Генерація, редагування посилань
Попередні умови	Відкрита сторінка «Links»
Вихідні умови	Згенероване або відредаговане посилання

Таблиця 3.8 Варіант використання «Керування списками відписки від розсилки»

Найменування	Керування списками відписки від розсилки
Первинний актор	Користувач
Інші актори	Немає
Опис	Створення та редагування списків відписки від розсилки
Попередні умови	Відкрита сторінка «Unsub lists»
Вихідні умови	Згенерований список абонентів, що не бажають отримувати повідомлення

Таблиця 3.9 Варіант використання «Керування списками імен для відправки повідомлень»

Найменування	Керування списками імен для відправки повідомлень
Первинний актор	Користувач
Інші актори	Немає
Опис	Генерація, редагування списків імен для відправки повідомлень
Попередні умови	Відкрита сторінка «Senders»
Вихідні умови	Згенерований або видалений список імен

Таблиця 3.10 Варіант використання «Отримання звітів про доставку»

Найменування	Отримання звітів про доставку
Первинний актор	Користувач
Інші актори	Немає
Опис	Перегляд звітів по доставлення повідомлень
Попередні умови	Відкрита сторінка «Broadcasts», вибрано потрібне повідомлення, натиснути «Download report»
Вихідні умови	Список звітів про доставлення

Таблиця 3.11 Варіант використання «Надсилання двухсторонніх повідомлень»

Найменування	Надсилання двухсторонніх повідомлень
Первинний актор	Користувач
Інші актори	Немає

Опис	Двухстороннє спілкування з абонентом стільникової мережі

Продовження табл. 3.11

Попередні умови	Відкрита сторінка з інформацією про ціни
Вихідні умови	Відкрита сторінка з інформацією про ціни

Таблиця 3.12 Варіант використання «Вибір SMSC»

Найменування	Вибір SMSC
Первинний актор	Користувач
Інші актори	немає
Опис	Можливість вибрати SMSC через який буде надсилатись повідомлення
Попередні умови	Змінити налаштування за допомогою технічної підтримки Надіслати повідомлення
Вихідні умови	Отримане повідомлення

Таблиця 3.13 Варіант використання «Редагування тексту повідомлень»

Найменування	Редагування тексту повідомлень
Первинний актор	Користувач
Інші актори	немає
Опис	Можливість редагувати повідомлення до моменту його відправки
Попередні умови	Відкрита сторінка «Broadcasts», вибрано потрібне повідомлення, натиснути «Edit», відредагувати потрібні дані, натиснути «Submit»
Вихідні умови	Відредаговане повідомлення

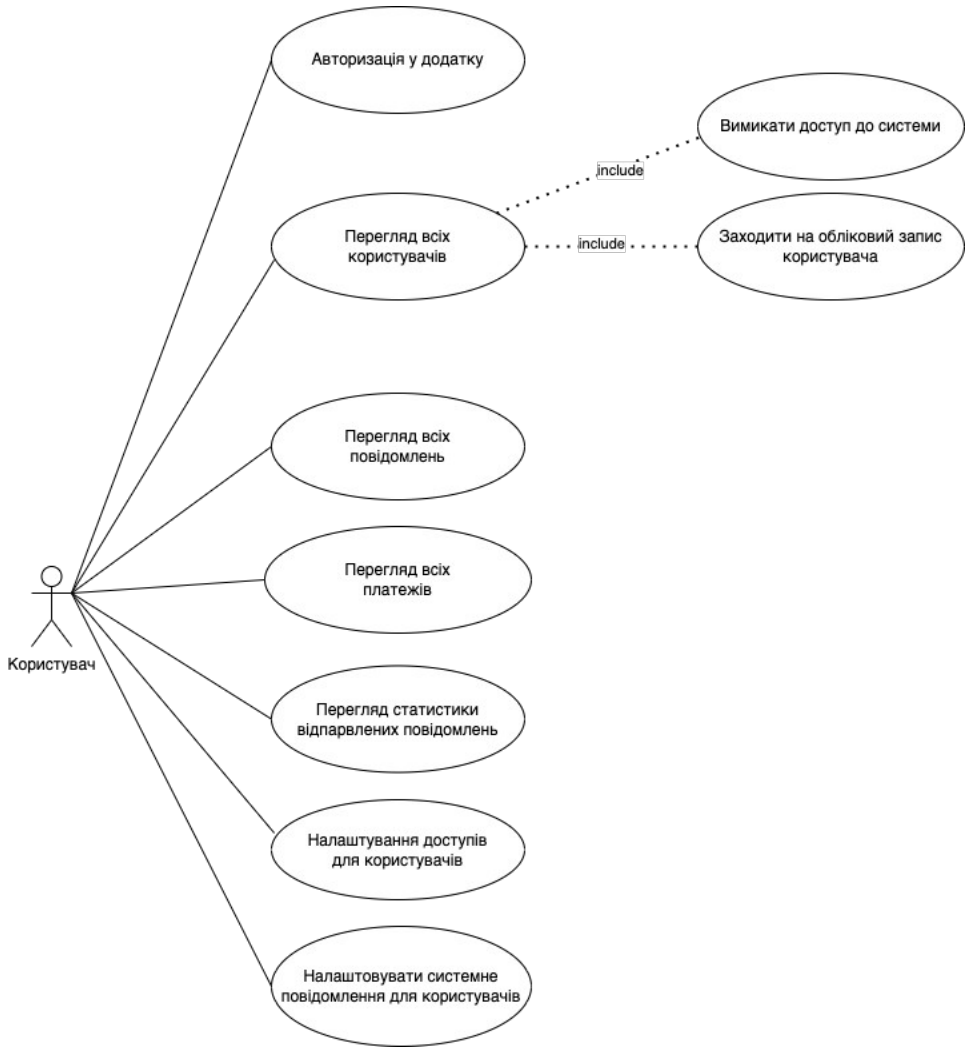


Рис. 3.2. Варіанти використання системи адміністратором

В табл. 3.14 – 3.22. наведено опис усіх варіантів використання системи адміністратором зображених на рис. 3.2.

Таблиця 3.14

Варіант використання «Авторизація у додатку»

Найменування	Авторизація у додатку
Первинний актор	Адіністратор
Інші актори	немає
Опис	Можливість авторизуватись у системі
Попередні умови	Наявний вірний логін, пароль адміністратора та відкрита сторінка авторизації у системі
Вихідні умови	Відкрита сторінка “Dashboard” системи

Таблиця 3.15

Варіант використання «Перегляд всіх користувачів»

Найменування	Перегляд всіх користувачів
Первинний актор	Адміністратор
Інші актори	немає
Опис	Можливість переглянути всіх існуючих користувачів
Попередні умови	Перейти на вкладку для адміністраторів Вибрати вкладку «Users»
Вихідні умови	Список всіх користувачів системи

Таблиця 3.16

Варіант використання «Вимкнути доступ користувачу до системи»

Найменування	Вимкнути доступ користувачу до системи
Первинний актор	Адміністратор
Інші актори	Користувач
Опис	Можливість вимикати доступ користувачам до системи
Попередні умови	Перейти на вкладку для адміністраторів Вибрати вкладку «Users» Натиснути кнопку «Disable» навпроти імені потрібного користувача
Вихідні умови	Вимкнений користувач не може увійти в систему

Таблиця 3.17

Варіант використання «Заходити на обліковий запис користувача»

Найменування	Заходити на обліковий запис користувача
Первинний актор	Адміністратор
Інші актори	Користувач

Продовження табл. 3.17

Опис	Можливість адміністратору авторизуватись від імені іншого користувача
Попередні умови	Перейти на вкладку для адміністраторів Вибрати вкладку «Users» Натиснути кнопку «Impersonate» навпроти імені потрібного користувача
Вихідні умови	Відкрита сторінка “Dashboard” системи від імені іншого користувача

Таблиця 3.18

Варіант використання «Перегляд всіх повідомлень»

Найменування	Перегляд всіх повідомлень
Первинний актор	Адміністратор
Інші актори	немає
Опис	Можливість переглянути всі повідомлення в системі
Попередні умови	Перейти на вкладку для адміністраторів Вибрати вкладку «Broadcasts»
Вихідні умови	Список всіх повідомлень

Таблиця 3.19

Варіант використання «Перегляд всіх платежів»

Найменування	Перегляд всіх платежів
Первинний актор	Адміністратор
Інші актори	немає
Опис	Можливість переглянути всі платежі створені користувачами
Попередні умови	Перейти на вкладку для адміністраторів Вибрати вкладку «Payments»

Вихідні умови	Список всіх платежів наявних в системі

Таблиця 3.20

Варіант використання «Перегляд статистики відправлених повідомлень»

Найменування	Перегляд статистики відправлених повідомлень
Первинний актор	Адміністратор
Інші актори	немає
Опис	Можливість переглядати статистику відправлених повідомлень
Попередні умови	Наявний вірний логін, пароль та відкрита сторінка авторизації у системі
Вихідні умови	Сторінка з графіками статистики відправлених повідомлень

Таблиця 3.21

Варіант використання «Налаштування доступів для користувачів»

Найменування	Налаштування доступів для користувачів
Первинний актор	Адміністратор
Інші актори	Користувач
Опис	Можливість адміністратору налаштовувати доступ до різного функціоналу
Попередні умови	Перейти на вкладку для адміністраторів Вибрати вкладку «Permissions» Додати або прибрати доступ
Вихідні умови	Функціонал доступний користувачу змінений

Таблиця 3.22

Варіант використання «Налаштовувати системне повідомлення для користувачів»

Найменування	Налаштовувати системне повідомлення для користувачів

Первинний актор	Адміністратор

Продовження табл. 3.22

Інші актори	Користувач
Опис	Можливість налаштовувати системне повідомлення для користувачів в хедері сайту
Попередні умови	Перейти на вкладку для адміністраторів Вибрати вкладку «Settings» Ввести нове повідомлення та зберегти його
Вихідні умови	Користувач бачить системне повідомлення в хедері додатка

3.2. Логічна структура системи

Першим та найголовнішим кроком до створення успішної системи – є правильно побудована логічна структура , адже це основа основ проектування складних систем. Кожна система є індивідуальною, тож і проєктування логічної структури для кожної системи буде відрізнятись. Проте варто зважати на певні вимоги до високонавантажених систем.

Логічна структура сама по собі являється набором функціонально-логічних модулів, які об’єднуються по предметній області, а саме сервіси які відповідаю за взаємодію лише з однією сутністю. Такі логічні модулі присутні як і на серверній частині (мікросервіси) так і на клієнтській (мікрофонтенди). Мікросервіси відповідають за опрацювання сутності та взаємодію користувача з базою даних цієї сутності. Мікрофонтенди це різні форми, таблиці, модальні вікна через які користувач може робити запити до мікросервісів [12].

Для формування логічної структури було побудовано UML діаграму схему класів системи посилання SMS повідомлень абонентам стільникової мережі, діаграма зображена на рисунку 3.3 [13].

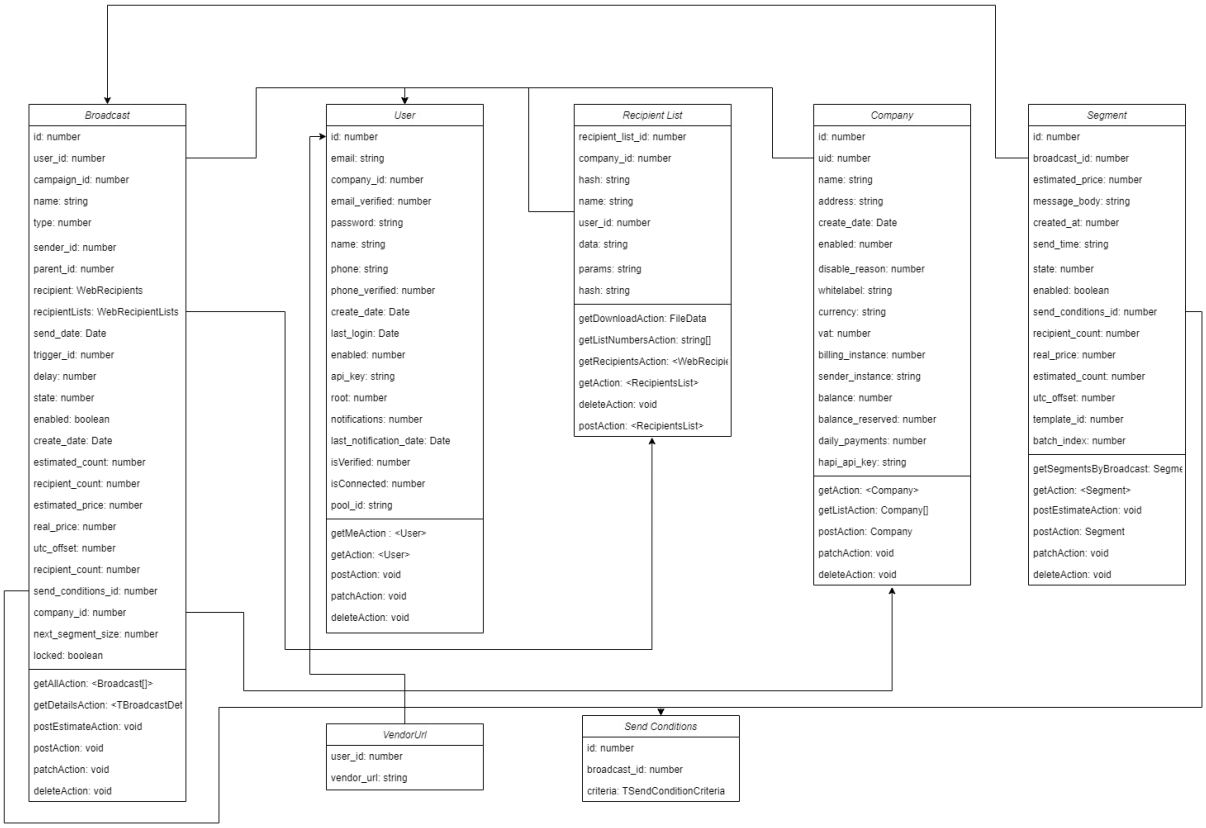


Рис. 3.3. Схема класів системи

Опис класів наявних в системі посилання SMS повідомлень абоненту стільникової наведено в табл. 3.23 .

Таблиця 3.23 Список класів системи

Клас	Опис
Broadcast	Сервіс для виконання маніпуляцій з сутністю повідомлення
User	Сервіс для виконання маніпуляцій з сутністю користувача
RecipientList	Сервіс для виконання маніпуляцій з сутністю списку відправників
Company	Сервіс для виконання маніпуляцій з сутністю компаній
Segment	Сервіс для виконання маніпуляцій з сутністю сегмента повідомлення

Продовження табл. 3.23

SendCondition	Сервіс для виконання маніпуляцій над даними часу відправки повідомлення
VendorUrl	Клас для надання можливості користувачу вибирати SMSC

В таблицях 3.24 – 3.28 описані основні методи для взаємодії з сутностями системи.

Таблиця 3.24 Список методів інтерфейсу IBroadcast

Тип даних	Метод та опис
Broadcast[]	getAll Отримати усі існуючі SMS повідомлення
TBroadcastDetailsResult	getDetails Отримати усі існуючі SMS повідомлення з детальною інформацією
Number	postEstimate Отримати вартість повідомлення
Broadcast	post Створити нове повідомлення
Broadcast	patch Редагувати існуюче повідомлення
Broadcast	delete Видалити існуюче повідомлення

Таблиця 3.25 Список методів інтерфейсу IUser

Тип даних	Метод та опис
User	getMe Запит для отримання токена аутентифікації
User	get

	Отримати дані користувача

Продовження табл. 3.25

User	post Створити та повернути користувача
User	patch Редагувати існуючого користувача
User	delete Видалити існуючого користувача

Таблиця 3.26 Список методів інтерфейсу IRecipientList

Тип даних	Метод та опис
FileData<RecipientList>	getDownload Отримати CSV файл з абонентами стільникової мережі
RecipientList []	getListNumbers Отримати список у JSON форматі абонентів стільникової мережі
RecipientList	getRecipients Отримати CSV файл з абонентами стільникової мережі та їх повні дані
RecipientsList	post Створити CSV список абонентів стільникової мережі
RecipientList	delete Видалити існуючий список абонентів стільникової мережі

Таблиця 3.27 Список методів інтерфейсу ICompany

Тип даних	Метод та опис

Продовження таблиці 3.27

Company	get Отримати одну маркетингову компанію користувача
Company[]	getList Отримати відфільтрований список маркетингових компанії користувача
void	post Створити нову маркетингову компанію
void	patch Редагувати існуючу маркетингову компанію
void	delete Видалити існуючу маркетингову компанію

Таблиця 3.28 Список методів інтерфейсу ISegment

Тип даних	Метод та опис
Segment	get Отримати інформацію про сегмент
Segments[]	getSegsByBrId Отримати всі існуючі сегменти пов’язані з ідентифікатором броадкаста
void	postEstimate Отримати вартість сегменту повідомлень
Segment	post Створити новий сегмент
void	patch Редагувати існуючий сегмент
void	delete Видалити існуючий сегмент

3.3. Фізична структура системи

З метою визначення фізичної структури системи посилання SMS повідомлень абонентам стільникової мережі була створена UML діаграма компонентів, вона знаходиться на рисунку 3.3 [13, 14].

Система посилання SMS повідомлень абонентам стільникової мережі розбита на дві частини, а саме серверну, структура якої зображена на рисунку 3.3, та клієнтську, структура якої зображена на рисунку 3.4. Класи, які були подані вище в підрозділі 3.2 [15] реалізовані в наступних файлах, а саме:

- Index.ts – основний файл, який розпочинає роботу системи посилання SMS повідомлень абонентам стільникової мережі.
- BroadcastController.ts – контроллер або проксі, який являється обробником даних, які приходять через транспорту від клієнтської частини та надсилає їх до сервісу BroadcastService.ts для виконання бізнес логіки.
- UserController.ts – контроллер або проксі, який являється обробником даних, які приходять через транспорту від клієнтської частини та надсилає їх до сервісу UserService.ts для виконання бізнес логіки.
- RecipientListController.ts – контроллер або проксі, який являється обробником даних, які приходять через транспорту від клієнтської частини та надсилає їх до сервісу RecipientListService.ts для виконання бізнес логіки.
- CompanyController.ts – контроллер або проксі, який являється обробником даних, які приходять через транспорту від клієнтської частини та надсилає їх до сервісу CompanyService.ts для виконання бізнес логіки.
- SegmentController.ts – контроллер або проксі, який являється обробником даних, які приходять через транспорту від

клієнтської частини та надсилає їх до сервісу SegmentService.ts для виконання бізнес логіки.

- BroadcastService.ts – сервіс наслідник інтерфейсу IBroadcast, що реалізує методи наслідуємого інтерфейсу.
- UserService.ts – сервіс наслідник інтерфейсу IUser, що реалізує методи наслідуємого інтерфейсу.
- RecipientListService.ts – сервіс наслідник інтерфейсу IRecipientList, що реалізує методи наслідуємого інтерфейсу.
- CompanyService.ts – сервіс наслідник інтерфейсу ICompany, що реалізує методи наслідуємого інтерфейсу.
- SegmentService.ts – сервіс наслідник інтерфейсу ISegment, що реалізує методи наслідуємого інтерфейсу.

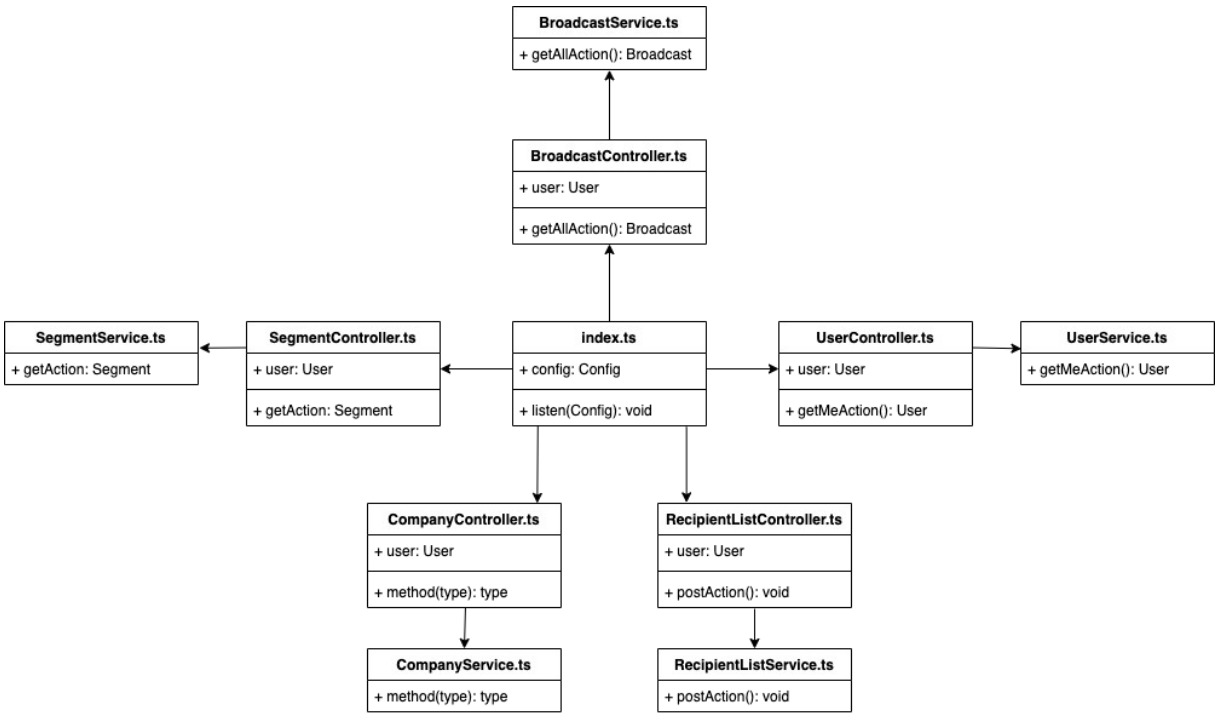


Рис. 3.3. Діаграма компонентів backend частини системи

На рисунку 3.4 зображена структуру frontend частини системи посилання SMS повідомлень абонентам стільникової мережі:

- Index.tsx – основний файл, який відповідає за навігацію по клієнтському застосунку модулю надсилання коротких повідомлень користувачам стільникового зв'язку.
- Dashboard.tsx – сторінка, на якій користувач може переглянути статистику відправлених повідомлень по маркетингових компаніях.
- SendCampaign.tsx – сторінка, на якій користувач може створювати SMS повідомлення.
- PhoneBook.tsx – сторінка, на якій користувач може створювати списки абонентів стільникової мережі з їхніми даними для подальших розсилок.
- Workspace.tsx – сторінка, на якій користувач може переглянути всі надіслані повідомлення.
- Billing.tsx – сторінка, на якій користувач може поповнити баланс облікового запису та переглянути всі створені транзакції на обліковому записі.
- MyAccount.tsx – сторінка, на якій користувач може переглянути інформацію про обліковий запис.

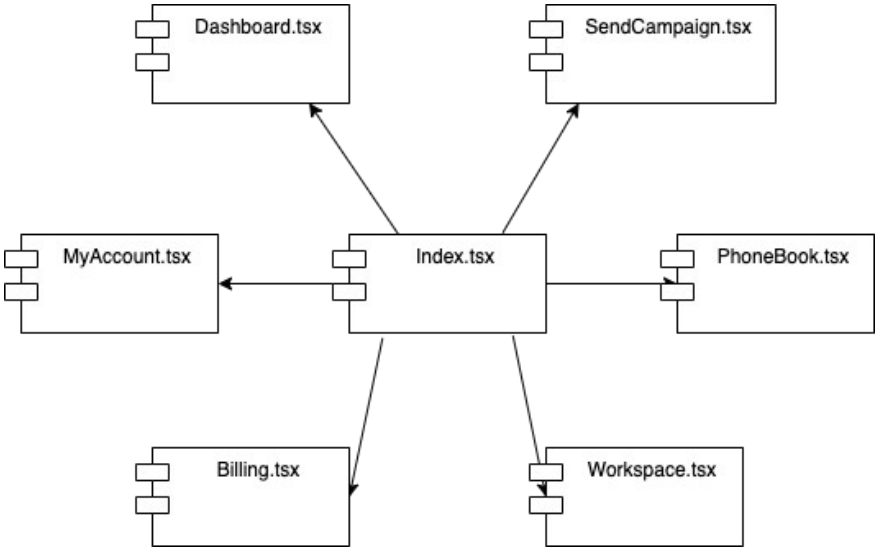


Рис. 3.4. Діаграма компонентів frontend частини системи

На рис. 3.5 зображена структуру файлів частини адміністрування системи надсилання SMS повідомлень абонентам стільникової мережі:

- Index.tsx – основний файл, який імпортує модулі всіх сторінок та відповідає за маршрутизацію веб-додатку системи надсилання SMS повідомлень абонентам стільникової мережі.
- Broadcasts.tsx – компонент, що відповідає за відображення списку всіх повідомлень.
- Payments.tsx – компонент, що відповідає за відображення списку всіх платежів.
- Permissions.tsx – компонент, що відповідає за відображення сторінки налаштування доступів для користувачів.
- Stats.tsx – компонент, що відповідає за відображення статистики відправлених повідомлень.
- Settings.tsx – компонент, що відповідає за відображення сторінки налаштувань системи.
- Users.tsx – компонент, що відповідає за відображення списку всіх користувачів.

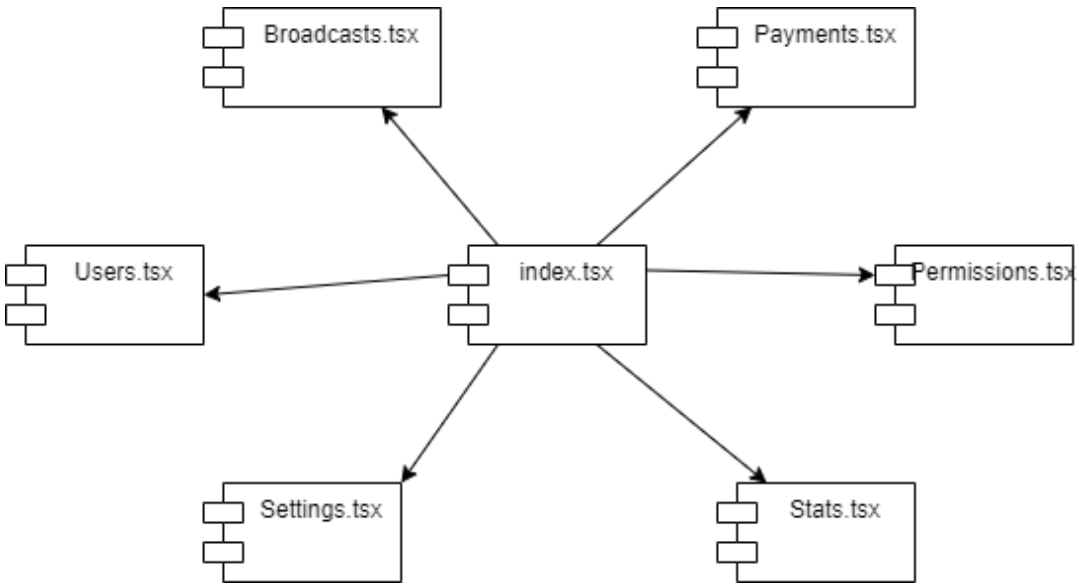


Рис. 3.5. Діаграма компонентів частини адміністрування системи

Система посилання SMS повідомлень абонентам стільникової мережі розгорнута за допомогою Google платформи, а саме сервісу Google Cloud за допомогою технології Google Cloud Kubernetes[16]. Схема розгортання системи відображена на рисунку 3.6. В Google Cloud Kubernetes виконане розгортання наступних баз даних, Redis, Nats Jeatstream, SqlMariaDB, MongoDB та всіх сервісів системи, а саме [16]:

- *MW-FRONT*
- *API*
- *SCHEDULER*
- *PRODUCER*
- *STATISTIC*
- *Link Shortener*

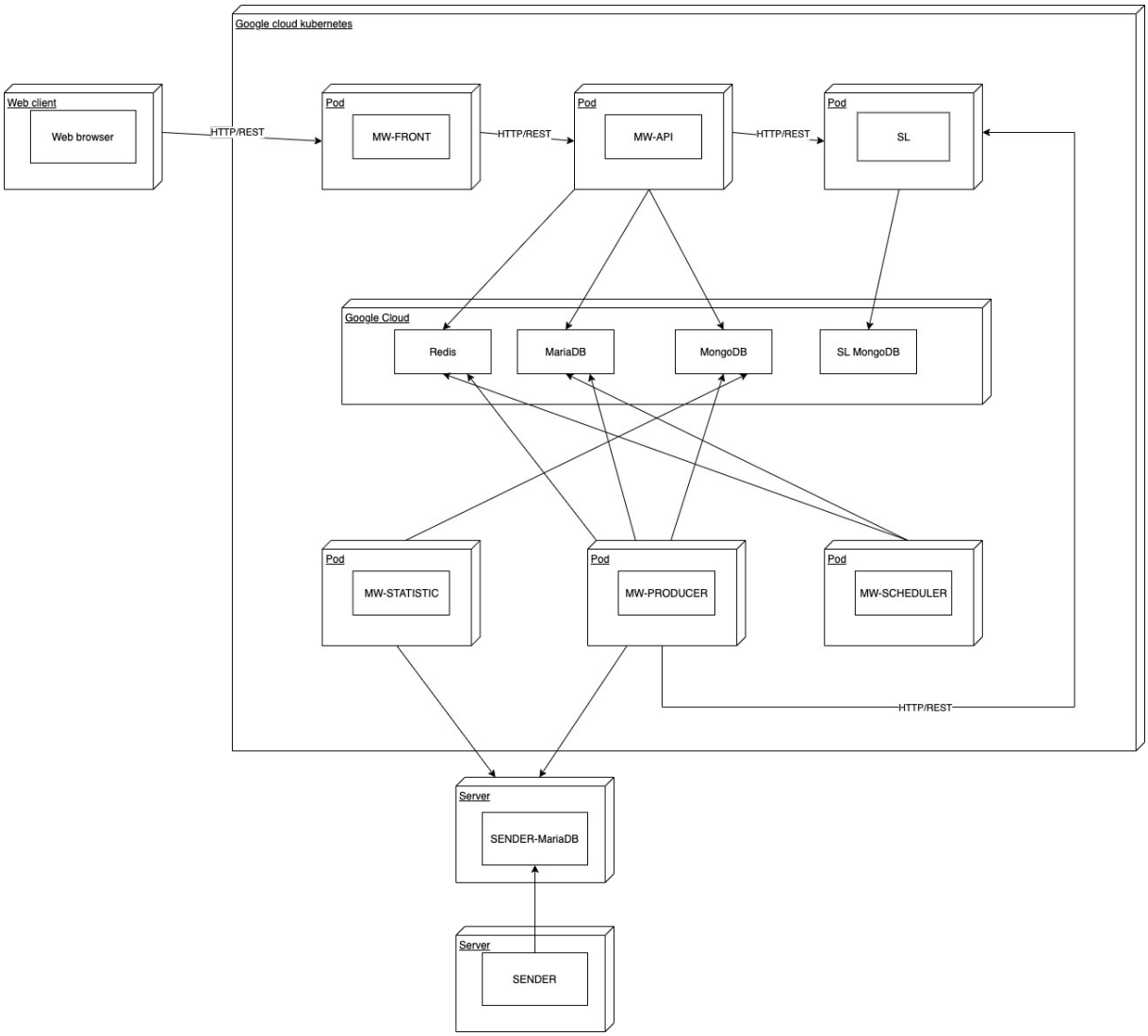


Рис. 3.6. Діаграма розгортання системи

Sender MariaDB, SENDER відокремлені в інший Google Cloud Kubernetes cluster.

Висновки

Було сформульовано варіанти використання системи, побудована логічна та фізична структури системи, а саме об’єктно-орієнтовану модель frontend та backend частин. На основі отриманих даних було реалізовано систему посилання SMS повідомлень абонентам стільникової мережі.

4. РОБОЧИЙ ПРОЄКТ СИСТЕМИ

4.1. Реалізування системи

В першу чергу на початку проєктування системи було поставлене питання, яку мову програмування краще використовувати для написання системи. Було порівняно декілька мов програмування та знайдено їх недоліки та переваги.

Java: одна з найпопулярніших мов програмування у світі. Тривалість її підтримки та її популярність означають, що це надійний вибір. Існує приказка, що «нікого ніколи не звільняли за покупку IBM», і те саме можна сказати про Java — нікого ніколи не звільняли за вибір Java як мови реалізації.

Код на мові Java виконується на віртуальній машини JVM, крос-платформенному середовищі виконання, яке дозволяє коду Java «написати один раз і запускати будь-де». JVM забезпечує економію за рахунок масштабу.

Ви навряд будете розгортати свою систему на разніх операційних системах, але ви отримаєте переваги від економії масштабу, створеної розробниками, які можуть перемикатися між цільовими операційними системами. Це означає, що існує більший ринок розробників, які націлені на JVM для найму.

Тепер давайте подивимося, що може запропонувати Java:

- Довга історія. Java має понад 20 років історії. За цей період мова програмування здобула величезну спільноту багатьох фахівців.
- Широкий вибір комунальних послуг. У Java є багато бібліотек і фреймворків, які роблять розробку легкою та зручною.

Незважаючи на всю потужність цієї мови програмування, вона має два ключові обмеження:

- Жорсткий процес завантаження
- Високе споживання ресурсів під час виконання.

Golang: Go була вперше представлена компанією Google у 2009 році. Golang — це термін, наданий цій мові через назву домену. В архітектурі мікросервісів фреймворк мікросервісів Golang добре відомий своєю підтримкою API та можливостями одночасної роботи. Його одночасна можливість також підвищує продуктивність кількох пристроїв. Крім того, можна легко створювати мікросервіси, оскільки їх функція полегшує всю розробку Golang.

Нарешті, давайте оглянемо переваги Golang:

- Широкий вибір різних утиліт. Golang пропонує потужну стандартну бібліотеку, безліч драйверів бази даних, готові HTTP-сервери, серіалізацію даних у різні формати та багато інших функцій, які допоможуть вам уникнути сторонніх бібліотек.
- Швидка компіляція. Практично всі проекти на базі Golang (незалежно від їх розміру) можна скомпілювати протягом хвилини.
- Простий у використанні. Компілятор Golang генерує лише один файл з усіма необхідними залежностями, що означає, що виконуваний файл можна легко включити в контейнер без додаткових налаштувань.
- Легка інтеграція Kubernetes. Зараз важко уявити написання мікросервісів без використання інструментів контейнерів. Будь-які мікросервісні рішення можна перемістити в контейнер Docker за допомогою інструментів таких як Kubernetes. Однак Golang має трохи більшу перевагу порівняно з іншими мовами програмування. Оскільки Kubernetes розроблено з Golang, це дає вам можливість імпортувати утиліти керування кластером у ваш проект.

- Простота коду. Golang вважається простою мовою, що робить її безпечнішою за інші мови програмування. Чим складніший код, тим менш безпечним він може бути.

Однак Golang має один головний недолік:

- Це молода мова програмування, і всі її компоненти є відкритими. Це робить його бібліотеки менш безпечними порівняно з C#, наприклад.

C# .NET є чудовим міжплатформним рішенням. Це дає вам змогу працювати з надійною та усталеною мовою, яка підтримується Microsoft. В неї вбудовані контейнери Docker допомагають у розробці мікросервісів. Мікросервіси .NET можна легко інтегрувати з програмами, розробленими на Node.JS, Java чи будь-якій іншій мові. Це полегшує перехід до основних технологій .NET.

Результати наших досліджень показали, що ця мова програмування має такі переваги:

- Кросплатформенність. C# використовує технологію .NET і Common Language Runtime (CLR). Ця технологія надає можливість працювати з кодом, написаним на Visual Basic .NET або іншими мовами програмування, які підтримують .NET.
- Зручність IDE: C# працює з Visual Studio IDE, що спрощує розробку програмного забезпечення.
- Готові утиліти: Пропонуючи багато готових рішень, .NET може сприяти розвитку продуктів на основі мікросервісів.
- Підтримка Microsoft: Завдяки підтримці Microsoft C# матиме довгострокові інвестиції, а це означає, що ця мова розвиватиметься в майбутньому.
- Якісна документація: MSDN має дуже хорошу документацію, яка допомагає розробникам використовувати технологію .NET найбільш ефективним способом.

Однак C# має один серйозний недолік, про який слід пам’ятати:

- Windows-орієнтований. Хоча багато крос-платформних продуктів для розробки .NET уже випущено, є частини, які краще працюють у Windows (наприклад, SignalR).

Node JS: протягом багатьох років Node JS став основною платформою для організацій і підприємств, які прагнуть запровадити мікросервіси. Node JS базується на середовищі виконання V8. Крім того, цей продукт дуже швидко прогресує та розвивається. Node.js — популярний фреймворк мікросервісів, який використовують розробники по всьому світу. Розробка веб-додатків за допомогою Node JS також дуже проста.

Переваги Node.js

- Асинхронний IO, тобто можливість паралельно опрацьовувати запити.
- Використовує мову програмування JavaScript, яка є легкою для використання.
- Код мовою JavaScript можна ширити між клієнтським додатком та серверним.
- Npm дуже великий та зручний пакетний менеджер, який весь час продовжує рости.
- Активна та динамічна спільнота, з великою кількістю коду, який шириться через github.

Недоліки Node.js

- Node.js не забезпечує масштабованість. Одного процесора буде недостатньо, платформа не надає можливості масштабування, щоб скористатися перевагами кількох ядер, які зазвичай присутні в сучасному обладнанні серверного класу.
- Node.js не підходить для завдань, що інтенсивно використовують ЦП. Він підходить лише для речей введення-виведення (наприклад, веб-серверів).

Дивлячись на всі недоліки та переваги різних мов програмування та фреймворків для них, було вирішено використовувати мову програмування JavaScript та фреймворк Node.js відповідно.

Сама система буде побудована на змішаній архітектурі, а саме з використанням монолітів та мікросервісів. Такий підхід відповідає кращим рекомендація для початку побудови високонавантажених систем та має свої переваги, а саме [19]:

- Масштабування.
- Кожен мікросервіс малий за розміром та легко переписується.
- Час підняття серверів.
- Швидкість знаходження проблемних місць.
- Гнучкість у реалізації нових модулів.
- Простота написання CI/CD .
- Можливість вибору мови програмування під відповідний мікросервіс.

Список технологій, які використовувались під час розробки системи:

- TypeScript (типізована надстройка для JavaScript).
- Node.js (фреймворк для JavaScript).
- Nest (HTTP фреймворк для Node.js).
- React.js (бібліотека для клієнтського додатку).
- MariaDB (представник сімейства SQL).
- MongoDB (noSQL база даних).
- Redis (кеш сховище).
- Nats Jetstream (меседж брокер) [28].

Система була зпроєктована на мові TypeScript. Вона являється надбудовою над мовою JavaScript та додає типізацію та багато інших бонусів, а саме зворотню сумісність. TypeScript чудово підходить для проєктування та побудови великих високонавантажених вебзастосунків.

Node.js являється повноцінним середовище розробки з відкритим кодом, яка складається з різних модулів, таких як zlib, libuv, V8[20].

Використання JavaScript для написання не лише клієнтського застосунку, а також серверної частини відразу надає такі переваги [21]:

- Розробник краще фокусується на одній мові програмування.
- Можливість поширювати модулі.
- Продуктивність.
- Обмін знаннями між розробниками.
- Асинхронність.
- Кросплатформенність.

MariaDB – це представник з сімейства SQL, а отже реляційна база дани. Вона надає такі переваги [22]:

- Чудова продуктивність.
- Має велику кількість використань та підтримку розробників.
- Має весь функціонал MySQL та свої надбудови.
- Транзакційність.
- Простота у підтримці.
- Чудово задокументований функціонал.

MongoDB має чудовий функціонал у вигляді GridFS, за допомогою якого система може зберігати великі файли розбиваючи їх на частини [23].

NestJS — це фреймворк, що розроблений для полегшення життя розробникам, використовуючи правильні архітектурні підходи та диктуючи власні правила.

Таким чином, NestJS — це не лише бекенд-фреймворк, але й можливість увійти у світ передових концепцій, таких як Domain-driven design, Event Sourcing та мікросервісна архітектура. Все упаковано в просту та легку форму.

Серед основних особливостей цього фреймворку можна виділити наступне:

- NestJS базується на Typescript, який дозволяє розробникам додавати типи до наших змінних і надає помилки компіляції та попередження на їх основі.
- Модульна структура, використання якої спрощує поділ проекту на окремі блоки. Це полегшує використання зовнішніх бібліотек у проектах.
- NestJS містить вбудований контейнер dependency injection. Впровадження залежностей — це шаблон проектування, який використовується, щоб зробити наші програми більш ефективними та модульними. Його часто використовують, щоб зберегти ваш код чистим, легким для читання та використання.
- Підтримується модульна організація проектів, які будуються відповідно до приналежності кожної логічної частини проекту (модуля) до певної предметної області.
- Використання NestJS спрощує тестування завдяки підтримці таких функцій, як контейнери та модулі dependency injection.
- Фреймворк дозволяє створювати розширювані програмні рішення, де немає сильного зв'язку між компонентами.

Клієнтська частина була реалізован за допомогою бібліотеки React. Плюси React:

Велика спільнота розробників: для комерційної розробки це найважливіша річ. Якщо вибрана технологія має велику аудиторію розробників, це набагато спрощує та прискорює життєвий цикл розробки. Коли ви стикаєтеся з будь-якою помилкою, ви можете запитати на публічному форумі, і є висока ймовірність, що серед мільйонів розробників React деякі з них вирішували цю помилку раніше. Завдяки нашій великій спільноті ми також маємо безліч сторонніх бібліотек React, посібників і заходів.

Зворотна сумісність: Це одна з найкращих речей React, що він не вносить критичних змін. Після кожного оновлення React API залишається

майже незмінним. Навіть якщо вони збираються внести більші зміни, ви будете бачити попередження протягом тривалого часу, тому ви можете легко перенести свій код.

Багаторазові компоненти: розробники React можуть створювати багаторазові компоненти. Це дозволяє розробникам створювати невеликі фрагменти інтерфейсу користувача та включати їх у будь-яке місце програми. Ви можете поєднати багато менших компонентів, щоб створити більший складний інтерфейс користувача. Досить часто розробники створюють додаток, використовуючи готові компоненти інтерфейсу користувача, наприклад. MaterialUI.

У світі програмування багато технологій побудовано на основі React. Серед усіх цих речей у нас є фреймворки, які дозволяють створювати додатки React, які відображаються сервером, або просто генерувати файли html на основі коду React. Прикладом SSG є Gatsby, який робить створення швидких і зручних для SEO веб-сайтів у React простим. Ще один корисний фреймворк — це Next.js — з ним ми маємо переваги SSG, а також можливість створювати сторінки, відображені в React сервером, що полегшує створення динамічних і зручних для SEO веб-додатків.

Віртуальний DOM: Virtual DOM (Document Object Model) — відповідає за підтримку абстракції програми React і синхронізацію в актуальному стані з реальним DOM (це те, що ми дійсно бачимо на нашому дисплеї). Це дозволяє покращити продуктивність і швидкість додатка, оскільки реалізація React Virtual DOM порівнює абстракцію додатка між змінами та повторно відтворює лише змінені частини додатка.

Що таке Redis? Згідно з офіційним документом Redis, Redis є сховищем структури даних у пам’яті, яке використовується як база даних, кеш-пам’ять і брокер повідомлень. Він підтримує такі структури даних, як рядки, хеші, списки, набори, відсортовані набори із запитами діапазону, растрові зображення, гіперлоглоги, геопросторові індекси з радіусними запитами та потоками.

Плюси Redis:

- Швидка робота.
- Завдяки легкому налаштуванню Redis простий і легкий у використанні.
- Redis має гнучкі структури даних, він підтримує майже всі структури даних.
- Redis дозволяє зберігати пари ключів і значень розміром до 512 МБ.
- Redis використовує власний механізм хешування під назвою Redis Hashing.
- Відкритий код і стабільність

4.2. Тестування системи

Для тестування системи було використано димне тестування, яке передбачає створення та проходження мінімального набору тестів, завдяки якому ми визначили працездатність системи.

Сценарій димного тестування:

1. Створення облікового запису в системі.
2. Авторизуватись в створений обліковий запис в системі.
3. Завантажити CSV список з абонентами.
4. Відправити Single Broadcast.
5. Відправити Scheduled Single Broadcast.
6. Редагувати Scheduled Single Broadcast.
7. Відправити Base Broadcast.
8. Відправити Scheduled Base Broadcast.
9. Редагувати Scheduled Base Broadcast.
10. Завантажити CSV список з відгуками на повідомлення
11. Відправити Scheduled Smart Broadcast.
12. Редагувати Scheduled Smart Broadcast.
13. Відправити Smart Broadcast.

В табл. 4.1 наведено димний набір тестів.

Таблиця 4.1 Набір тестових випадків

№	Кроки виконання	Результат
1	1.Зайти на сторінку «Signup». 2.Ввести ім'я користувача в поле «Full name». 3.Ввести назву компанії в поле «Company name». 4.Ввести адресу електронної пошти в поле «Email». 5.Ввести мобільний номер користувача в поле «Phone number». 6.Вибрати валюту в полі «Currency». 7.Ввести пароль в полі «Password». 8.Повторити пароль для його підтвердження в полі «Retype your password». 9.Підтвердити форму реєстрації	1.Зараєстровано новий обліковий запис. 2.Відкрита сторінка «Login».
2	1.Відкрити сторінку «Login». 2.Ввести адресу електронної пошти в поле «Email». 3.Ввести пароль в полі «Password». 4.Натиснути кнопку «Login».	1.Авторизація під існуючим користувачем. 2.Відкрита сторінка «Workspace».
3	1. Відкрити сторінку «Recipients». 2. Натиснути кнопку «Add Recipient List». 3. Завантажити CSV файл з даними.	1. Успішно завантажений CSV список.

Продовження табл. 4.1

4	1.Відкрити сторінку «Send Broadcast». 2.Ввести значення «Single» в полі «Type». 3.Ввести номер абонента в полі «Choose recipient». 4.Ввести номера абонентів, якім не хочуть отримувати повідомлення «Choose unsubscribe». 5.Ввести ім'я відправника в полі «Choose sender». 6.Вибрати дату та час відправлення в полі «Starting condition». 7.Вибрати шаблон тексту повідомлення в полі «Template». 8.Ввести текст повідомлення в полі «Write Message». 9.Натиснути кнопку «Estimate». 10. Натиснути кнопку «Send».	1. Відкрита сторінка «Broadcasts». 2. Повідомлення наявне в списку надісланих.
5	1.Створити Scheduled Single Broadcast 2. Натиснути кнопку «Edit». 3.Змінити текст для Scheduled Single Broadcast. 4. Вибрати дату відправлення «Start now». 5. Зберегти зміни.	1. Відредаговане повідомлення надіслане відразу після редагування.

Продовження табл. 4.1

6	1.Відкрити сторінку «Send Broadcast». 2.Ввести значення «Single» в полі «Type». 3.Ввести номер абонента в полі «Choose recipient». 4.Ввести номера абонентів, якім не хочуть отримувати повідомлення «Choose unsubscribe». 5.Ввести ім'я відправника в полі «Choose sender». 6.Вибрати дату та час для відкладеного повідомлення в полі «Starting condition». 7.Вибрати шаблон тексту повідомлення в полі «Template». 8.Ввести текст повідомлення в полі «Write Message». 9.Натиснути кнопку «Estimate». 10. Натиснути кнопку «Send».	1. Відкрита сторінка «Broadcasts». 2. Повідомлення наявне в списку надісланих з відкладеною датою відправлення.
7	1. Створити Scheduled Base Broadcast. 2. Натиснути кнопку «Edit». 3.Змінити текст для Scheduled Base Broadcast. 4. Вибрати дату відправлення «Start now». 5. Зберегти зміни.5. Зберегти зміни.	1. Відредаговане повідомлення надіслане відразу після редагування.

Продовження табл. 4.1

8	.Відкрити сторінку «Send Broadcast». 2.Ввести значення «Single» в полі «Broadcast». 3.Ввести номер абонента в полі «Choose recipient». 4.Ввести номера абонентів, якім не хочуть отримувати повідомлення «Choose unsubscribe». 5.Ввести ім'я відправника в полі «Choose sender». 6.Вибрати дату та час відправлення в полі «Starting condition». 7.Вибрати шаблон тексту повідомлення в полі «Template». 8.Ввести текст повідомлення в полі «Write Message». 9.Натиснути кнопку «Estimate». 10. Натиснути кнопку «Send»	1. Відкрита сторінка «Broadcasts». 2. Повідомлення наявне в списку надісланих.
9	1. Відкрити сторінку «Broadcast». 2. Натиснути кнопку «Download report» для відповідного повідомлення.	1. CSV список звітів завантажений.

Продовження табл. 4.1

10	1.Відкрити сторінку «Send Broadcast». 2.Ввести значення «Base» в полі «Type». 3.Ввести номер абонента в полі «Choose recipient». 4.Ввести номера абонентів, якім не хочуть отримувати повідомлення «Choose unsubscribe». 5.Ввести ім’я відправника в полі «Choose sender». 6.Вибрати дату та час для відкладеного повідомлення в полі «Starting condition». 7.Вибрати шаблон тексту повідомлення в полі «Template». 8.Ввести текст повідомлення в полі «Write Message». 9.Натиснути кнопку «Estimate». 10. Натиснути кнопку «Send».	1.Відкрита сторінка «Broadcasts». 2. Повідомлення наявне в списку надісланих з відкладеною датою відправлення.
11	1. Створити Scheduled Smart Broadcast. 2. Натиснути кнопку «Edit». 3.Змінити текст для Scheduled Base Broadcast. 4. Вибрати дату відправлення «Start now». 5. Зберегти зміни.	1 . Відредаговане повідомлення надіслане відразу після редагування.

Продовження табл. 4.1

12	1. Відкрити сторінку «Send Broadcast». 2. Заповнити поле «Type» значенням «Smart Broadcast». 3. Ввести значення «Smart» в полі «Type». 4. Ввести ім’я повідомлення в поле «Broadcast name». 5. Вибрати список абоненті в полі «Choose recipient». 6. Вибрати список абонентів що не хочуть отримувати повідомлення в полі «Choose unsubscribe». 7. Ввести розмір сегменту в полі «Segment size». 8. Ввести ім’я відправника в полі «Choose sender». 9. Вибрати дату та час відправлення в полі «Starting condition». 10. Вибрати шаблон тексту повідомлення в полі «Template». 11 Ввести текст повідомлення в полі «Write Message». 12.Натиснути кнопку «Estimate». 13. Натиснути кнопку «Send».	1. Відкрита сторінка «Broadcasts». 2. Повідомлення наявне в списку надісланих.

Продовження табл. 4.1

13	1.Відкрити сторінку «Send Broadcast». 2. Заповнити поле «Type» значенням «Smart Broadcast». 3. Ввести значення «Smart» в полі «Type». 4. Ввести ім'я повідомлення в поле «Broadcast name». 5. Вибрати список абоненті в полі «Choose recipient». 6. Вибрати список абонентів що не хочуть отримувати повідомлення в полі «Choose unsubscribe». 7. Ввести розмір сегменту в полі «Segment size». 8. Ввести ім'я відправника в полі «Choose sender». 9. Вибрати відкладену дату та час відправлення в полі «Starting condition». 10. Вибрати шаблон тексту повідомлення в полі «Template». 11 Ввести текст повідомлення в полі «Write Message». 12.Натиснути кнопку «Send».	1.Відкрита сторінка «Broadcasts». 2.Повідомлення наявне в списку надісланих з відкладеною датою відправлення.

4.3. Демонстрація системи

Головною сторінкою системи є сторінка авторизації в веб застосунку. Вона має досить зручний та простий інтерфейс, на якій користувач може зайти в систему. Вона зображена на рисунку 4.1.

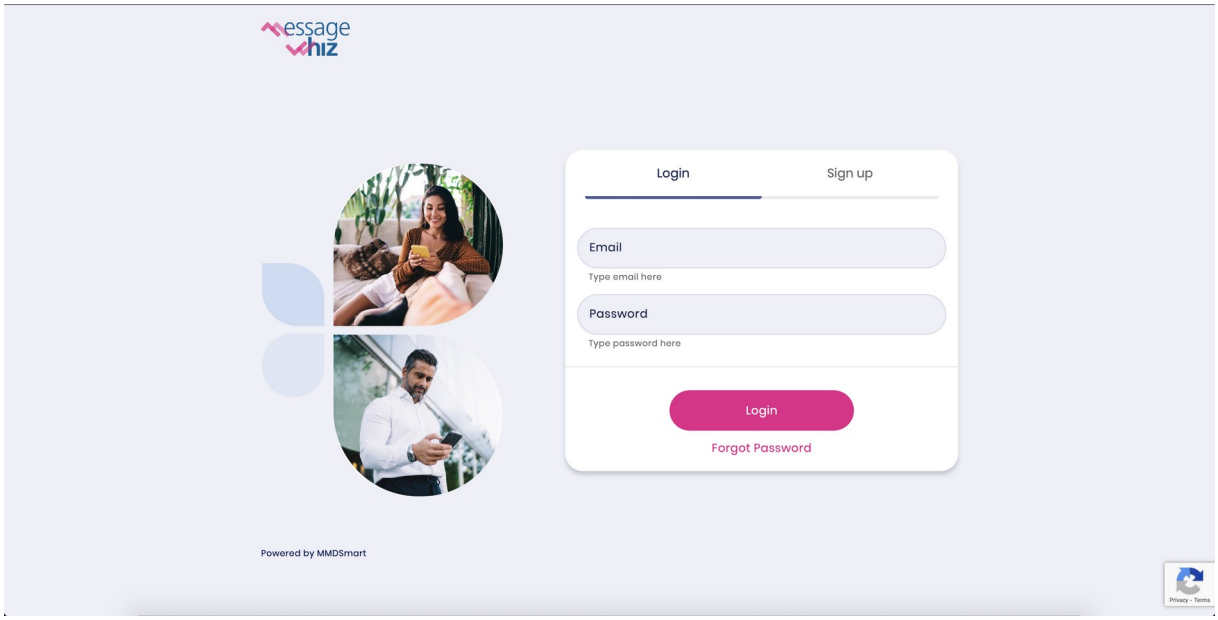


Рисунок 4.1. сторінка Login

Сторінка реєстрації представлена на рисунку 4.2. Вона має дуже простий та зрозумілий інтерфейс для користувачів.

Для того щоб зареєструватись користувачу потрібно ввести певні дані про себе, а саме:

- Повне ім’я.
- Номер телефону.
- Назву компанії.
- Поштову електронну адресу.
- Вибрати валюту.
- Ввести пароль.
- Підтвердити введений пароль.

Після цього натиснути кнопку «Sign Up» та система створить обліковий запис користувача та всі потрібні записи в базі даних.

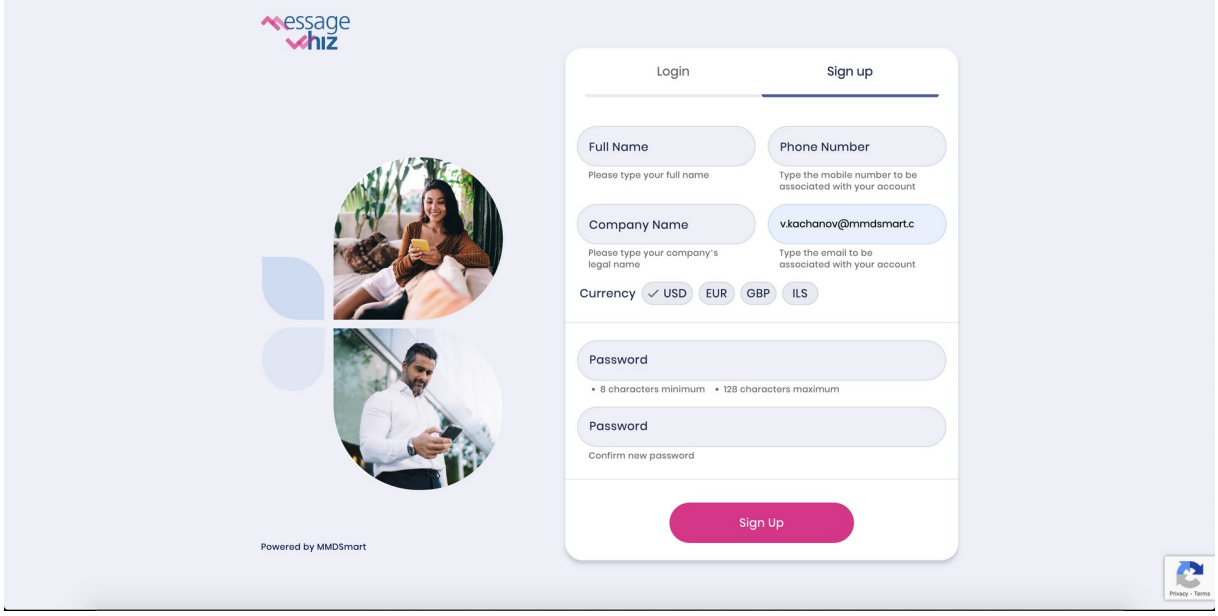


Рис. 4.2. Сторінка реєстрації

Сторінка на якій користувач може створити повідомлення зображена на рисунку 4.3. Користувач має широкий спектр налаштування для створення повідомлень, а саме:

Вибрати маркетингову компанію в межах якої здійснюється надсилання повідомлень.

Визначити тип створюваного повідомлення.

Ідентифікувати повідомлення відповідною назвою.

Вибрати абонентів для надсилання повідомлень.

Вибрати абонентів, що не хочуть отримувати повідомлення.

Вибрати ім’я відправника або список імен, які будуть призначатись у випадковому порядку повідомленням.

Вибрати час та дату надсилання повідомлення.

Написати текст повідомлення або вибрати заздалегідь підготовлений шаблон

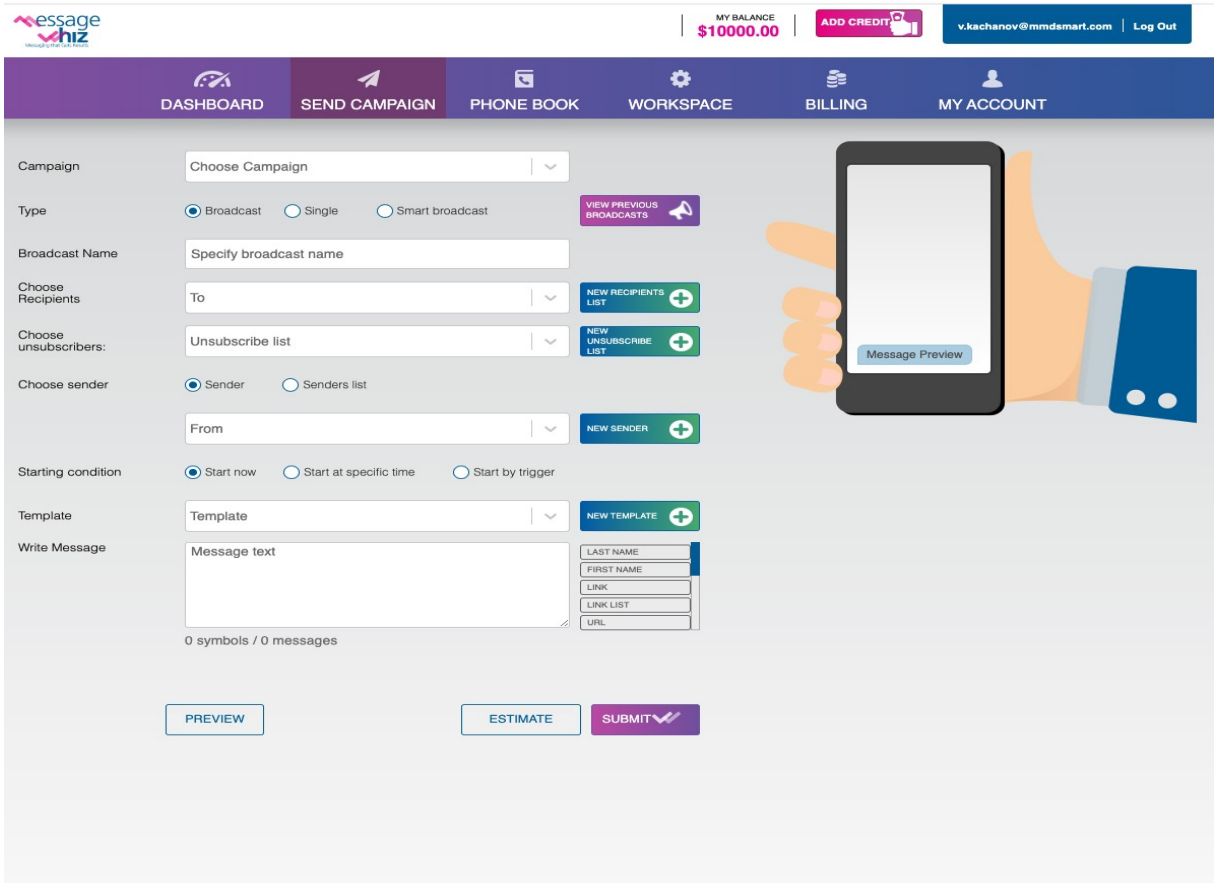


Рис. 4.3. Сторінка створення SMS повідомлень

Сторінка для перегляду всіх надісланих SMS повідомлень знаходиться на рисунку 4.4. Користувач на цій сторінці може завантажувати CSV списки з різною інформацією про повідомлення та робити пошук за різними фільтрами, а саме :

- Пошук за назвою повідомлення.
- Пошук за типом повідомлення.
- Пошук за реальною датою відправлення.
- Пошук за фактичною датою створення.
- Сортування за реальною датою відправлення.
- Сортування за фактичною датою створення.

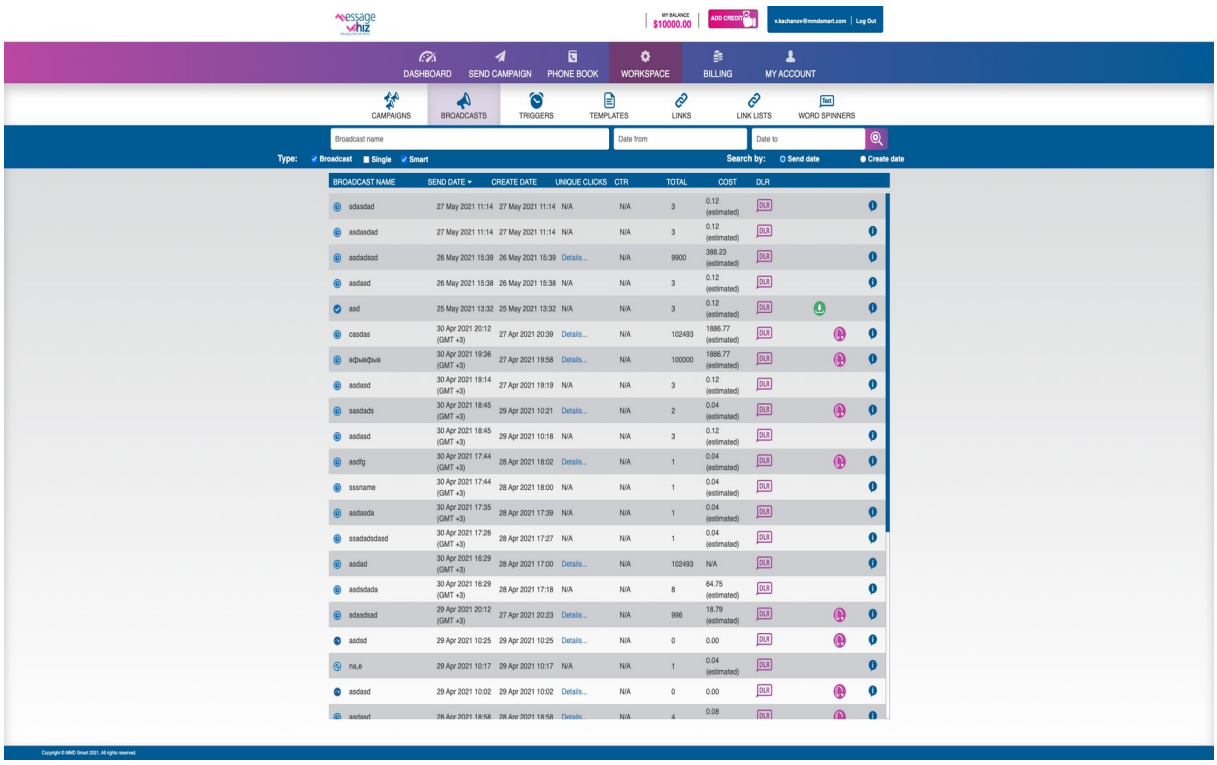


Рис. 4.4. Сторінка перегляду списку повідомлень

Сторінка для перегляду всіх завантажених абонентів в систему знаходиться на рисунку 4.5. На ній видно легкий та зручний інтерфейс для створення, редагування та видалення списків абонентів.

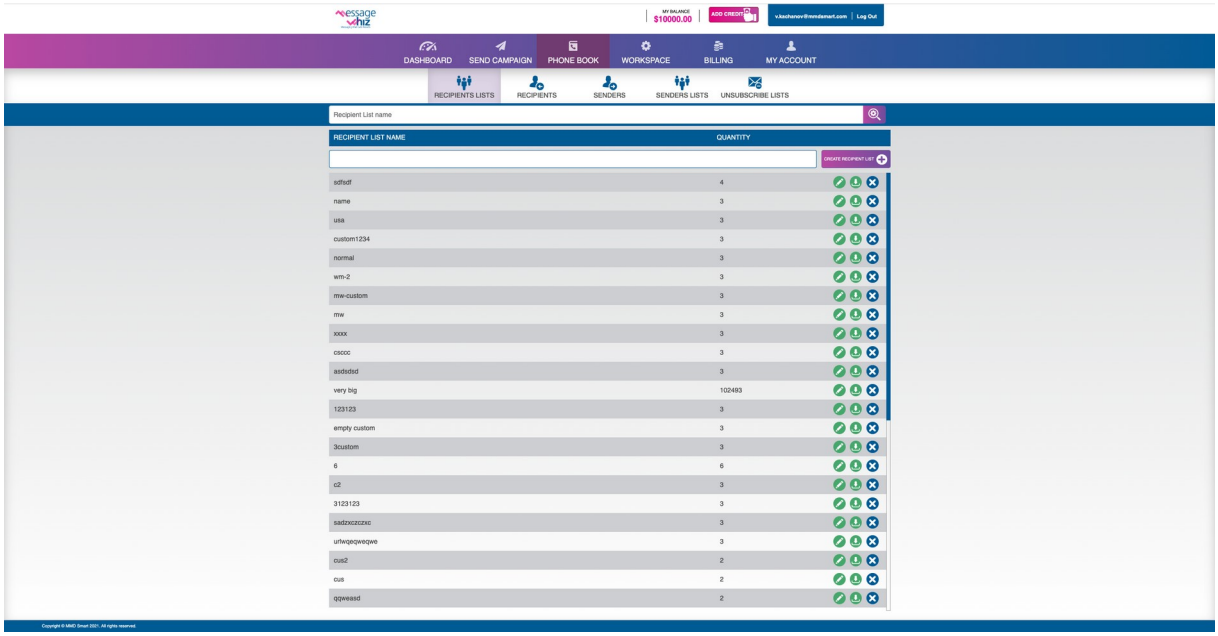


Рис. 4.5. Сторінка перегляду списків одержувачів

На рисунку 4.6 зображена сторінка поповнення балансу облікового запису.

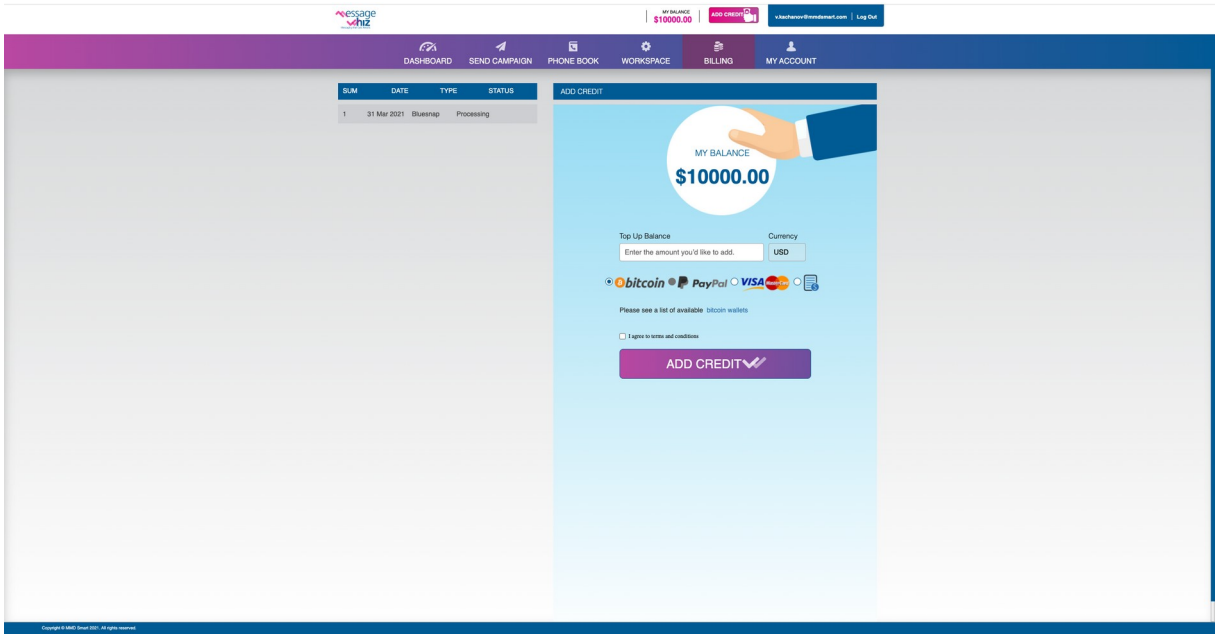


Рис. 4.6. Сторінка поповнення балансу

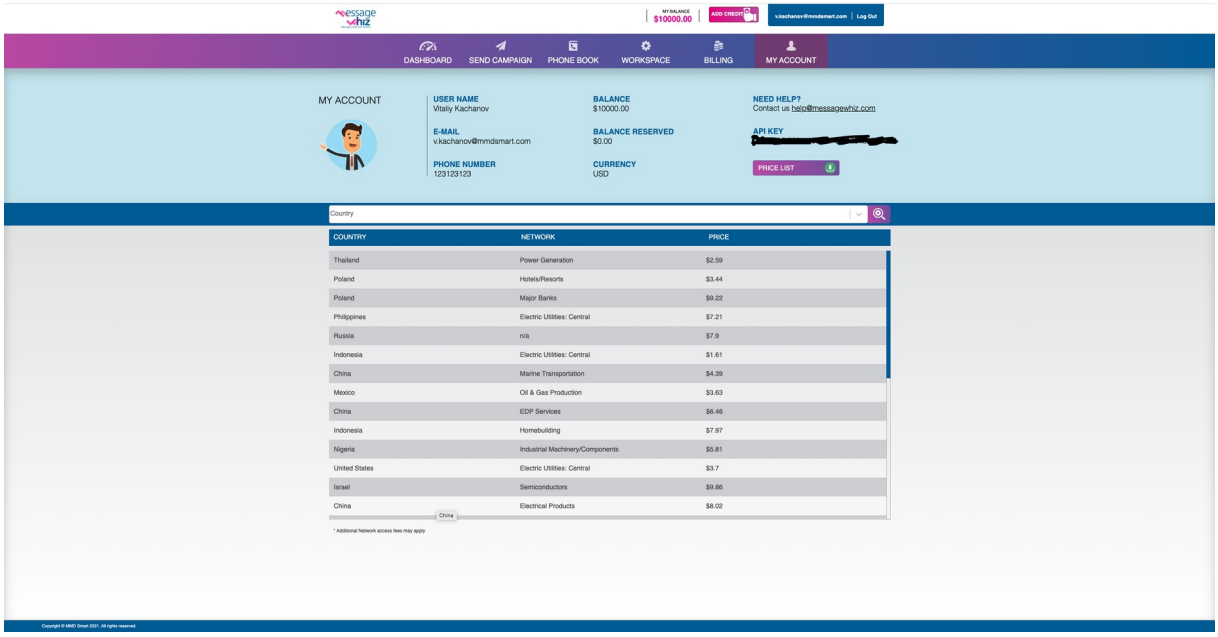


Рис. 4.7. Сторінка перегляду інформації про обліковий запис

Інформація про обліковий запис користувача знаходиться на сторінці «My account». Дану сторінку зображено на рисунку 4.7. На ній користувач може знайти свій APIkey для інтеграцій та переглянути ціни по регіонам.

4.4. Порівняння швидкодії роботи клієнтської частини

У світі веб-розробки стає доречним не лише створювати функціональне програмне забезпечення, але й робити його високопродуктивним, щоб воно могло надавати користувачам незабутній досвід, використовуючи мінімальні ресурси.

Тому було проведено тестування нашої системи та порівняння його з іншими аналогами. Тестування клієнтської частини проводилось за допомогою Google Chrome Tools, а саме частиною Performance.

Була сформована вибірка з понад 100 разів проходження тестового сценарію, а саме відкриття головної сторінки на різних платформах та різного заліза.

Характеристики системи на якій відбувалось тестування веб-додатка:

- CPU : Ryzan 5 3600 3.2 GHz
- GPU: MSI 3060 12GB
- Motherboard: Asus prime b 350 plus
- RAM: 16 GB (2x8) 2900 GHz

На рис. 4.8 зображені результати для системи аналогічній, ту що ми розробляли, на ньому видно час затрачений на

- Завантаження -15 мс
- Виконання скриптів – 691мс
- Рендерінг – 242мс
- Рисування – 86мс
- Система – 380мс
- Загальний час – 1414мс

На рис. 4.9 зображений результати розробленої системи, на ньому видно час затрачений на

- Завантаження -27 мс
- Виконання скриптів – 680мс
- Рендерінг – 51мс
- Рисування – 17мс
- Система – 186мс
- Загальний – 961мс

Отже, на десктопній версії веб-додатку нова розроблена система показала себе краще, а саме різницю в 453мс або 48%.

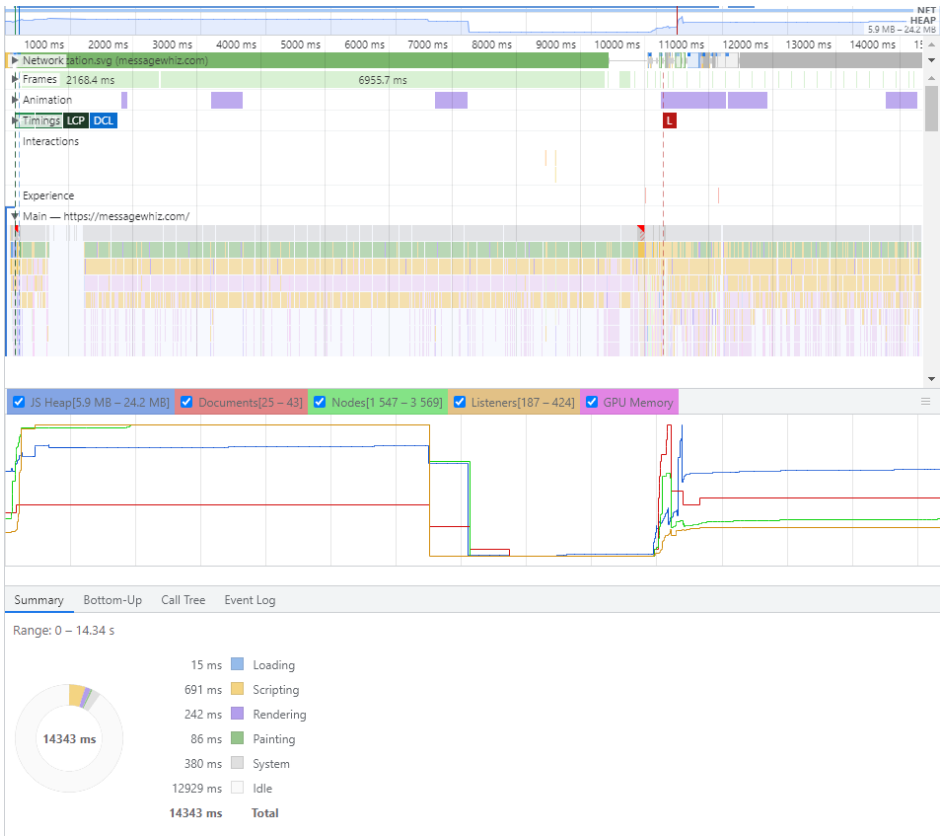


Рис. 4.8 Результати аналогічної системи

Рис. 4.9 Результати розробленого веб-додатку

Мобільний пристрій на якому відбувалось тестування веб-додатка для мобільних пристроїв: Iphone 12 plus.

На рис. 4.10 зображені результати для системи аналогічній, ту що ми розробляли, на ньому видно час затрачений на

- Завантаження -130 мс
- Виконання скриптів – 3343мс
- Рендерінг – 923мс
- Рисування – 183мс
- Система – 1552мс
- Загальний час – 6131мс

На рис. 4.11 зображений результати розробленої системи, на ньому видно час затрачений на

- Завантаження -250 мс

- Виконання скриптів – 3958мс
- Рендерінг – 731мс
- Рисування – 102мс
- Система – 1215мс
- Загальний – 6256мс

Отже, на мобільній версії веб-додатку нова розроблена система показала себе краще, але не на багато, приріст яскраво не виражений, а саме 125мс або 3%.

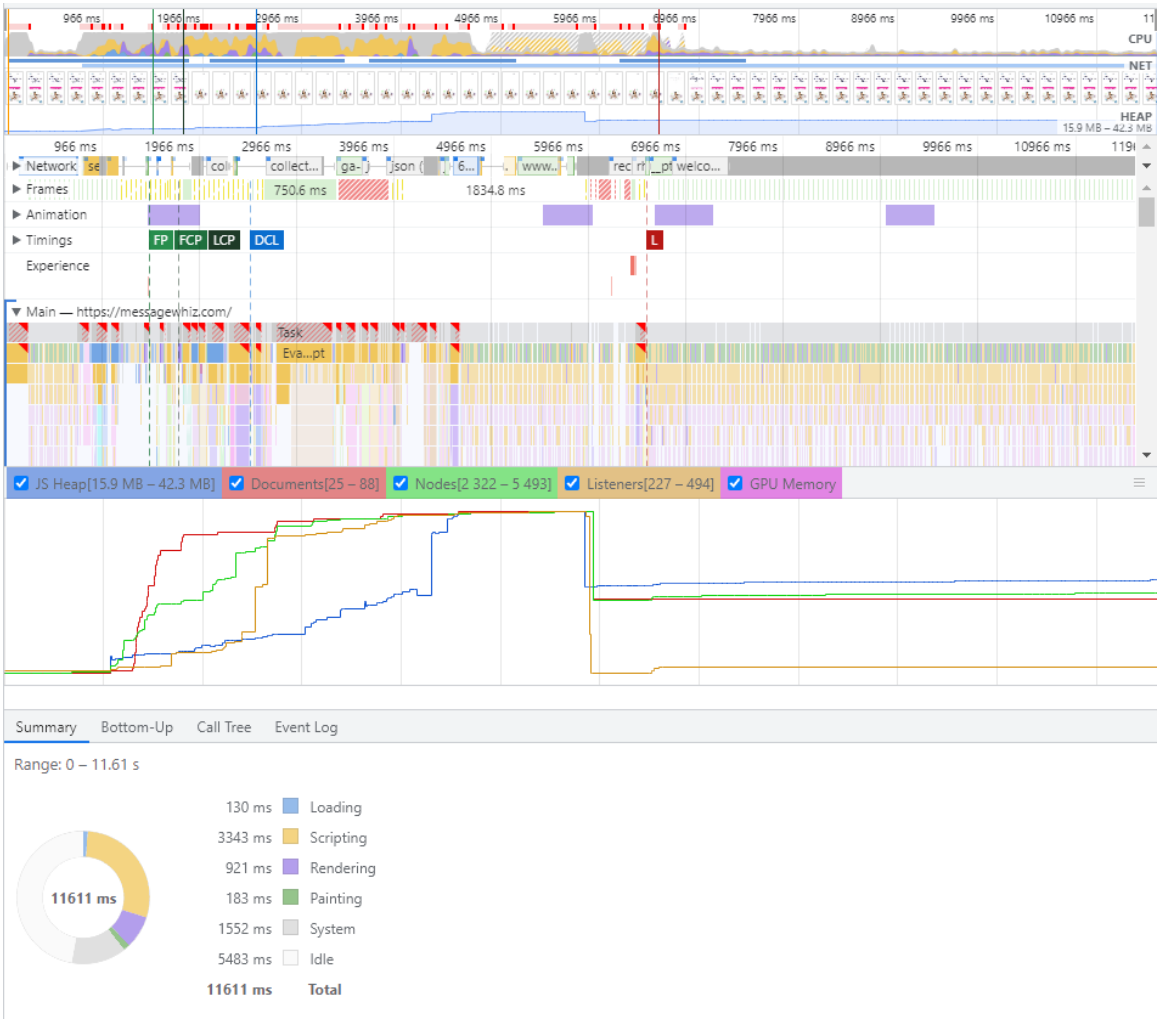


Рис. 4.10 Результати розробленої системи для телефонів

Рис. 4.11 Результати аналогічної системи для телефонів

Висновки

Обрали необхідні технології для реалізації системи посилання SMS повідомлень абонентам стільникової мережі, а саме: мова програмування, бази даних, меседж брокери, кеш сховища, транспортний протокол для передачі даних та бібліотеки для створення мікро фронтенду та мікро сервісів для бекенду.

Створено список тестових випадків для мінімальної перевірки працездатності системи посилання SMS повідомлень абонентам

стільникової мережі. Працездатність системи перевірена за допомогою димного тестування.

Продемонстровано можливості системи посилання SMS повідомлень абонентам стільникової мережі, а саме: створення облікового запису, авторизацію, створення повідомлень, перегляд усіх створених повідомлень, інформацію про обліковий запис.

ВИСНОВКИ

За результатами виконання магістерського дипломного проєкту змодельовано систему надсилання коротких повідомлень стільниковою мережею.

Проаналізовано інші системи надсилання SMS повідомлень абонентам стільникової мережі. На основі проведених досліджень було сформульовано їх переваг та недоліків, з отриманих даних було сформовано функціональні вимоги до системи.

Побудовано концептуальну модель системи посилання SMS повідомлень абонентам стільникової мережі. Визначено основні її функції, етапи та потоки різних даних у графічних нотаціях DFD, IDEF3, IDEF0.

Побудовано об’єктно-орієнтовану модель системи посилання SMS повідомлень абонентам стільникової мережі. Встановлено варіанти її використання, також створено логічну та фізичну структуру.

Обґрунтовано вибір технологій, мов програмування та реалізовано модуль надсилання коротких повідомлень користувачам стільникового зв’язку як вебзастосунок. Проведено його тестування та продемонстровано використання.

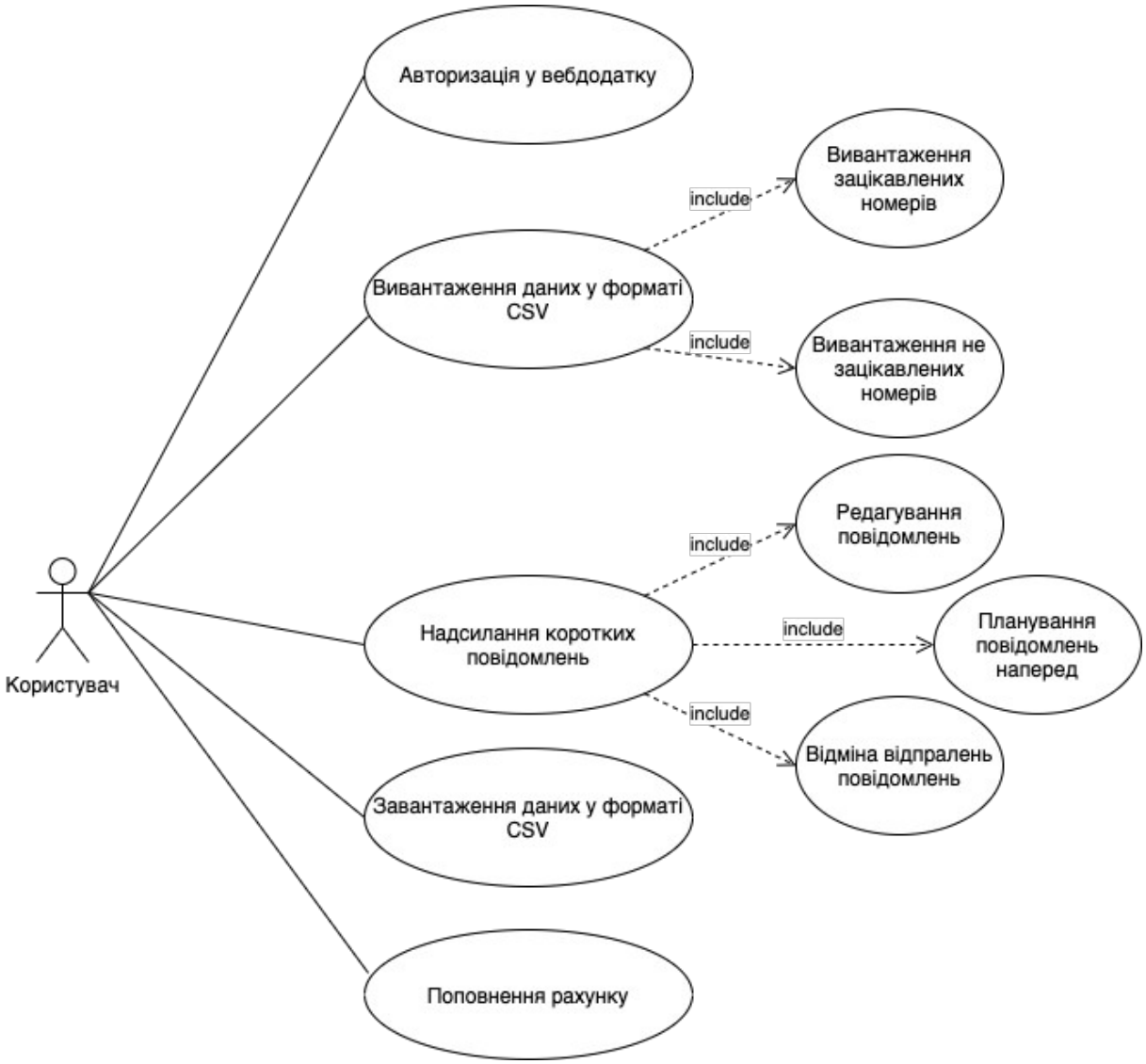
СПИСОК ВИКОРИСТАНИХ ЛІТЕРАТУРНИХ ДЖЕРЕЛ

1. SMSC [Електронний ресурс] – Режим доступу до ресурсу: https://bit.ly/3AO2GsG
2. Groupe Spécial Mobile [Електронний ресурс] – Режим доступу до ресурсу: https://bit.ly/2SJBNmw
3. TextMagic [Електронний ресурс] – Режим доступу до ресурсу: https://bit.ly/3p8iwaQ
4. 11 Best SMS Marketing Platforms for Businesses in 2023[Електронний ресурс] – Режим доступу до ресурсу: https://bit.ly/3PfpkQZ
5. The 5 best SMS apps in 2021 [Електронний ресурс] – Режим доступу до ресурсу: https://bit.ly/34z3VM0
6. SalesMessage [Електронний ресурс] – Режим доступу до ресурсу: https://bit.ly/3c3ZeOc
7. IDEF0 [Електронний ресурс] – Режим доступу до ресурсу: https://bit.ly/2S46iDI
8. IDEF3 [Електронний ресурс] – Режим доступу до ресурсу: https://bit.ly/34yKVgL
9. Что такое DFD [Електронний ресурс] – Режим доступу до ресурсу: https://bit.ly/3pa0fd4
10. DFD [Електронний ресурс] – Режим доступу до ресурсу: https://bit.ly/3ibNJIu
11. UML для самых маленьких [Електронний ресурс] – Режим доступу до ресурсу: https://bit.ly/2R8GGFm
12. UML для бизнес моделирования [Електронний ресурс] – Режим доступу до ресурсу: https://bit.ly/3uE04I9
13. Класифікація ПЗ [Електронний ресурс] – Режим доступу до ресурсу: https://bit.ly/3fEWPfm
14. Структура програмного забезпечення [Електронний ресурс] – Режим доступу до ресурсу: https://bit.ly/34I6xHf

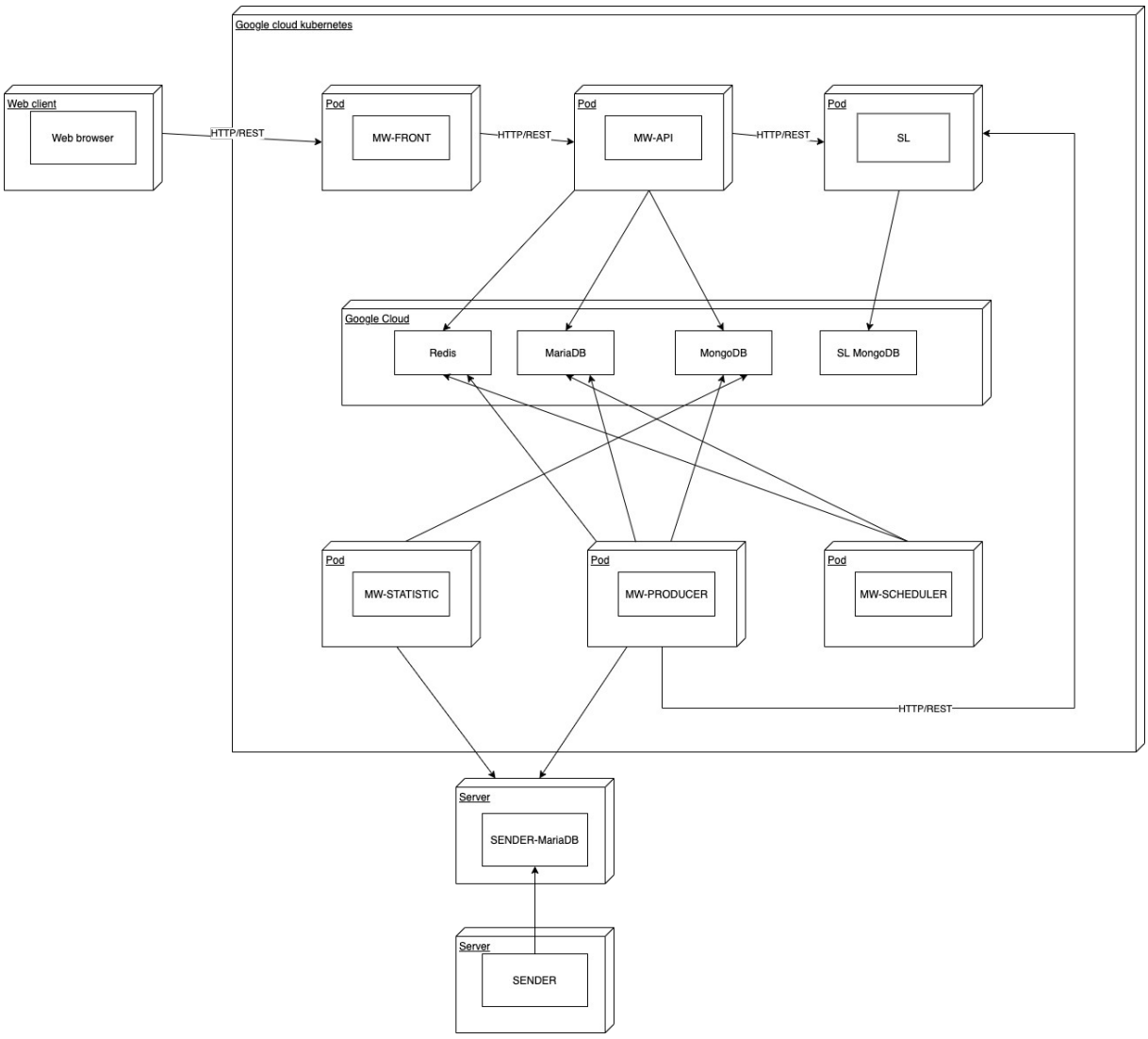
15. Структура програмного забезпечення [Електронний ресурс] – Режим доступу до ресурсу: https://bit.ly/34ywJnM
16. Програмне забезпечення [Електронний ресурс] – Режим доступу до ресурсу: https://bit.ly/34FHUuM
17. Google Cloud [Електронний ресурс] – Режим доступу до ресурсу: https://bit.ly/2RSeLKr
18. GSP: Разбор вычислительного стека Google Cloud Platform [Електронний ресурс] – Режим доступу до ресурсу: https://bit.ly/3yT0ihS
19. Просто о микросервисах [Електронний ресурс] – Режим доступу до ресурсу: https://bit.ly/3x1jXKX
20. Микросервисы [Електронний ресурс] – Режим доступу до ресурсу: https://bit.ly/3ibYE4W
21. What is TypeScript? [Електронний ресурс] – Режим доступу до ресурсу: https://bit.ly/3fUchDa
22. Node.js [Електронний ресурс] – Режим доступу до ресурсу: https://bit.ly/34P4T77
23. MariadDB proc and cons [Електронний ресурс] – Режим доступу до ресурсу: https://bit.ly/34z9vht
24. MongoDB docs[Електронний ресурс] – Режим доступу до ресурсу: https://bit.ly/3c8OlL3
25. Выбор лучшего Node.js фреймворка [Електронний ресурс] – Режим доступу до ресурсу: https://bit.ly/34Iasnr
26. React proc and cons [Електронний ресурс] – Режим доступу до ресурсу: https://bit.ly/34yzk10
27. Redis proc and cons [Електронний ресурс] – Режим доступу до ресурсу: https://bit.ly/2Rgk3z2
28. Nats [Електронний ресурс] – Режим доступу до ресурсу: https://nats.io/

ДОДАТКИ

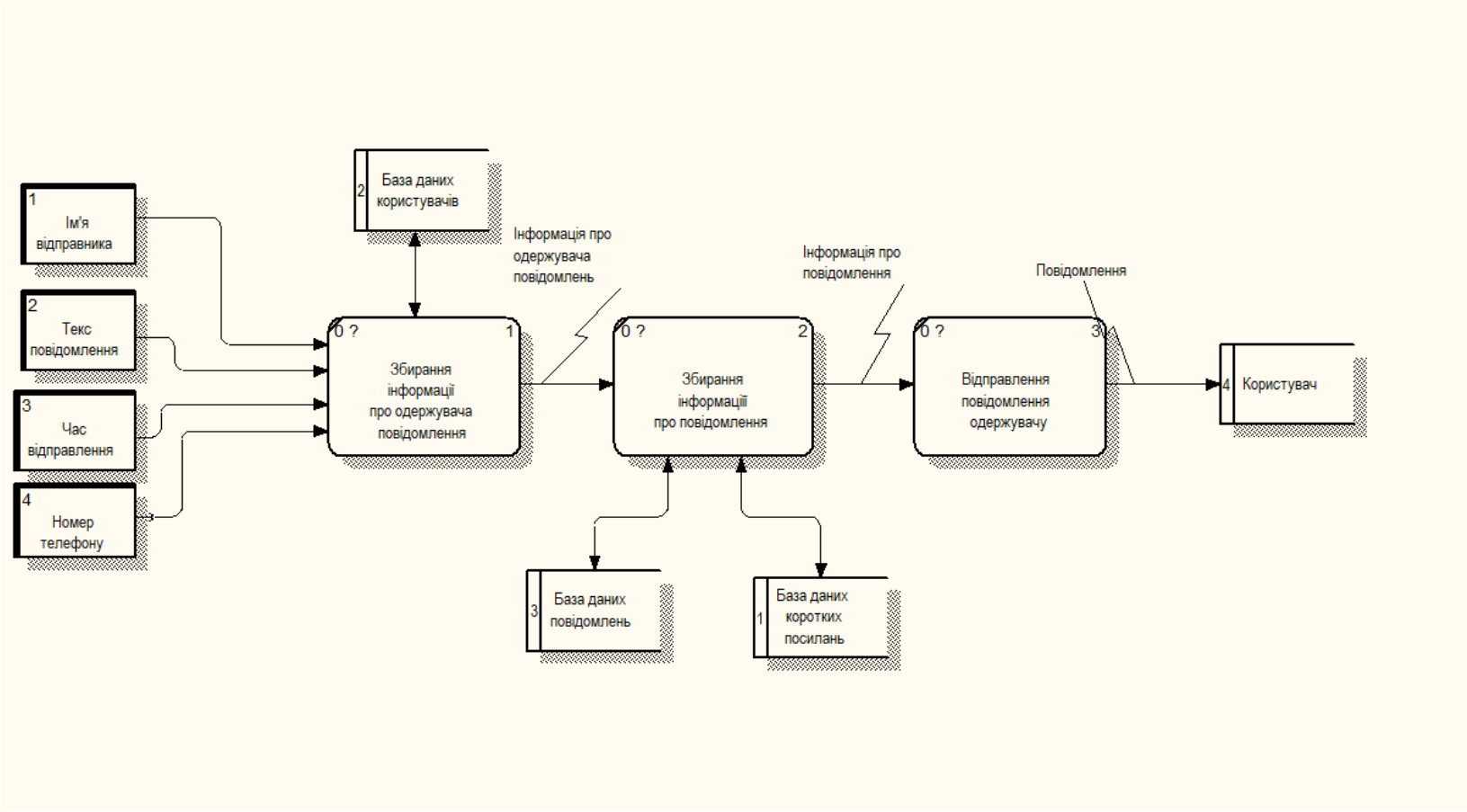
Додаток 1
Копії графічних матеріалів



ДП.045440-06-99
Система надсилання коротких повідомлень абонентам стільникової мережі. Варіанти використання системи. Діаграма варіантів використання

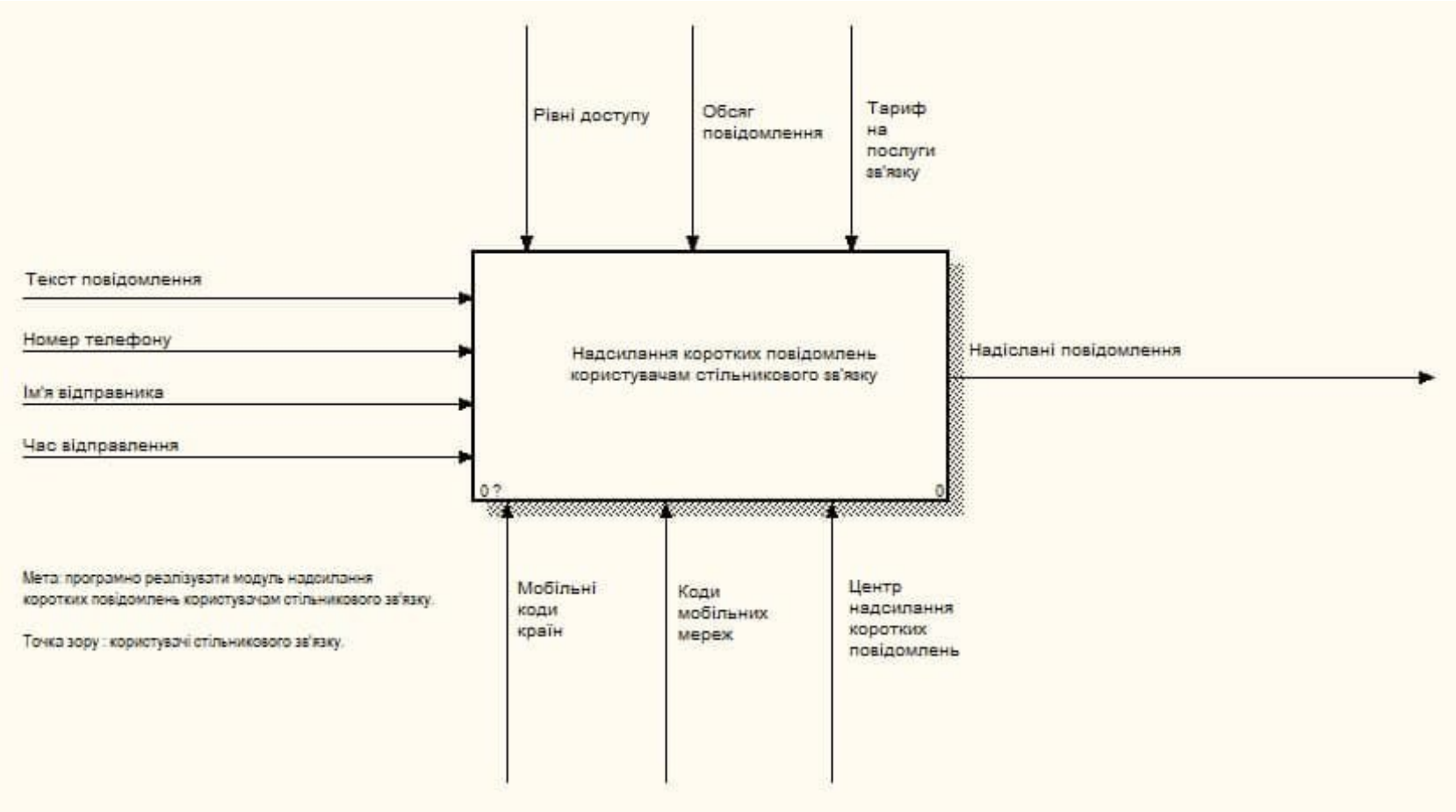


ДП.045440-06-99
Система надсилання коротких повідомлень абонентам стільникової мережі. Вузли системи та зв’язки між ними. Діаграма розгортання



Діаграма потоків даних системи надсилання коротких повідомлень абонентам стільникової мережі

Качанов В.І., група КВ-11мп



Функціональна модель системи надсилання коротких повідомлень абонентам стільникової мережі в нотації IDEF0
Качанов В.І., група КВ-11мп

Додаток 2
Копія презентації

Додаток 3
Копія довідки про впровадження результатів магістерського дипломного проєкту