

Міністерство освіти і науки України
Тернопільський національний технічний університет імені Івана Пулюя

(повне найменування вищого навчального закладу)

Факультет комп'ютерно-інформаційних систем і програмної інженерії

(назва факультету)

Кафедра комп'ютерних систем та мереж

(повна назва кафедри)

КВАЛІФІКАЦІЙНА РОБОТА

на здобуття освітнього ступеня

магістр

(освітній рівень)

на тему: Математичне та програмне забезпечення комп'ютерних систем
моніторингу та аналізу постачання мережевих послуг Інтернет-провайдером

Виконав: студент (ка) VI курсу, групи СІМ-61

Спеціальності:

123 "Комп'ютерна інженерія"

(шифр і назва спеціальності)

Залісковий Ю. І.

підпис

(прізвище та ініціали)

Керівник

Лецишин Ю.З.

підпис

(прізвище та ініціали)

Нормоконтроль

Луцик Н. С

підпис

(прізвище та ініціали)

Завідувач кафедри

Осухівська Г. М.

підпис

(прізвище та ініціали)

Рецензент

Готович В.А.

підпис

(прізвище та ініціали)

м. Тернопіль – 2023

Міністерство освіти і науки України
Тернопільський національний технічний університет імені Івана Пулюя

Факультет комп'ютерно-інформаційних систем і програмної інженерії
(повна назва факультету)

Кафедра комп'ютерних систем та мереж
(повна назва кафедри)

ЗАТВЕРДЖУЮ

Завідувач кафедри

Осухівська Г.М.

(підпис)

(прізвище та ініціали)

« ____ » _____ 2023 р.

**ЗАВДАННЯ
НА КВАЛІФІКАЦІЙНУ РОБОТУ**

на здобуття освітнього ступеня магістр
(назва освітнього ступеня)

за спеціальністю 123 «Комп'ютерна інженерія»
(шифр і назва спеціальності)

студенту Залісковому Юрію Ігоровичу
(прізвище, ім'я, по батькові)

1. Тема роботи Математичне та програмне забезпечення комп'ютерних систем моніторингу та аналізу постачання мережеских послуг Інтернет-провайдером

Керівник роботи Лецишин Юрій Зіновійович к.т.н., доцент кафедри КС
(прізвище, ім'я, по батькові, науковий ступінь, вчене звання)

Затверджені наказом ректора від «01» грудня 2023 року № 4/7-1132

2. Термін подання студентом завершеної роботи 26. 12. 2023 р.

3. Вихідні дані до роботи Вимоги до систем моніторингу, наукові літературні дослідження

4. Зміст роботи (перелік питань, які потрібно розробити)

Вступ. 1. Огляд і аналіз існуючих рішень і проблем систем моніторингу і аналізу мережеских послуг. 2. Проектування математичного, програмного і алгоритмічного забезпечення системи моніторингу і аналізу мережеских послуг.

3. Застосування обраних методів розробки для створення системи моніторингу і аналізу мережеских послуг. 4. Охорона праці та безпека в надзвичайних ситуаціях. Висновки

5. Перелік графічного матеріалу (з точним зазначенням обов'язкових креслень, слайдів)

1. Актуальність мети, мета завдання дослідження

2. Об'єкт, предмет, наукова новизна і практичне значення дослідження

3. Структурна схема системи з аналізу і моніторингу

4. Структурна схема набору технологій MERN

5. Блок-схема алгоритму роботи системи аналізу і моніторингу

6. Математичні формули. 7. Результат виведення математичних обчислень

8. Висновки

6. Консультанти розділів роботи

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв
<i>Охорона праці та безпека</i>	<i>Осухівська Г. М.</i>		
<i>в надзвичайних ситуаціях</i>	<i>Стадник І. Я.</i>		

7. Дата видачі завдання _____

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів роботи	Термін виконання етапів роботи	Примітка
1	<i>Аналіз наукових робіт у сфері моніторингу і аналізу постачання мережевих послуг</i>		<i>Виконано</i>
2	<i>Формування актуальності роботи, наукової новизни</i>		<i>Виконано</i>
3	<i>Огляд і аналіз існуючих рішень для систем моніторингу і методів розробки програмного забезпечення</i>		<i>Виконано</i>
4	<i>Обґрунтування ефективності обраного методу розробки</i>		<i>Виконано</i>
5	<i>Написання математичного забезпечення</i>		<i>Виконано</i>
6	<i>Розробка алгоритму роботи системи</i>		<i>Виконано</i>
7	<i>Написання програмного забезпечення для системи моніторингу і аналізу мережевих послуг</i>		<i>Виконано</i>
8	<i>Тестування роботи системи</i>		<i>Виконано</i>
9	<i>Написання розділу «Охорона праці та безпека в надзвичайних ситуаціях»</i>		<i>Виконано</i>
10	<i>Оформлення пояснювальної записки</i>		<i>Виконано</i>
11	<i>Оформлення графічного матеріалу</i>		<i>Виконано</i>
12	<i>Попередній захист кваліфікаційної роботи</i>		<i>Виконано</i>
13	<i>Захист кваліфікаційної роботи</i>		

Студент _____
(підпис)*Залісковий Ю. І.*

(прізвище та ініціали)Керівник роботи _____
(підпис)*Лецишин Ю.З.*

(прізвище та ініціали)

АНОТАЦІЯ

Математичне та програмне забезпечення комп'ютерних систем моніторингу та аналізу постачання мережевих послуг Інтернет-провайдером // Кваліфікаційна робота магістра // Залісковий Юрій Ігорович // ТНТУ, Комп'ютерна інженерія, група СІм-61 // Тернопіль, 2023 // с. – 82, рис. –34, табл. – 1, бібліогр. – 29.

Ключові слова: комп'ютерна система, аналіз, моніторинг, веб-ресурс, сервер, бази даних.

Кваліфікаційну роботу магістра присвячено дослідженню і розробці математичного та програмного забезпечення для моніторингу і аналізу ефективності роботи інтернет-провайдера. Було проаналізовано методи для збору інформації для аналізу постачання мережевих послуг. Здійснено огляд існуючих рішень, обрано засоби розробки і проаналізовано їх недоліки і переваги. Розроблено математичні методи для обрахунку параметрів і характеристик ефективності постачання мережевих послуг. Спроектовано алгоритм роботи системи і запропоновано способи покращення застосованого програмного забезпечення. Налаштовано програму для моніторингу The Dude, створено можливості для виведення інформації в базу даних MySQL з подальшим використанням чат-бота. Практично реалізовано веб-ресурс, забезпечено зв'язок з базою даних MongoDB, реалізовано функції авторизації працівників і математичного обчислення визначених параметрів.

ABSTRACT

Mathematical and software support for computer systems monitoring and analysis of network service delivery by an Internet provider// Master's graduation thesis // Zaliskovyi Yurii Ihorovych // Ivan Pulyu Ternopil National Technical University, Faculty of Computer Information Systems and Software Engineering, Department of Computer Systems and Networks, group CIM-61 // Ternopil, 2023 // p. – 82, fig. –34, tab. – 1, bibliography - 29.

Keywords: computer system, analysis, monitoring, web resource, server, databases.

The master's qualification work is devoted to the research and development of mathematical and software for monitoring and analyzing the effectiveness of the Internet provider. Methods for gathering information to analyze network service delivery were analyzed. An overview of existing solutions was carried out, development tools were selected and their shortcomings and advantages were analyzed. Mathematical methods have been developed for calculating the parameters and characteristics of the efficiency of the supply of network services. The algorithm of system operation was designed and ways to improve the applied software were proposed. The program for monitoring The Dude has been configured, opportunities have been created to output information to the MySQL database with the subsequent use of a chatbot. The web resource is practically implemented, the connection with the MongoDB database is ensured, the functions of authorization of employees and mathematical calculation of the specified parameters are implemented.

ЗМІСТ

ВСТУП.....	8
РОЗДІЛ 1 ОГЛЯД ТА АНАЛІЗ ІСНУЮЧИХ РІШЕНЬ І ПРОБЛЕМ МОНІТОРИНГУ І АНАЛІЗУ МЕРЕЖЕВИХ ПОСЛУГ	11
1.1 Огляд джерел отримання даних для аналізу постачання мережеских послуг	11
1.2 Аналіз рішень з моніторингу мереж	15
1.3. Аналіз рішень з розробки програмного забезпечення для розробки системи збору і аналізу	17
1.3.1. Спосіб розробки веб-ресурсу за допомогою технології WordPress Stack.	19
1.3.2. Спосіб розробки веб-ресурсу за допомогою технологій LAMP та WAMP Stacks.....	21
1.3.3. Спосіб розробки веб-ресурсу за допомогою технологій JavaScript Full Stacks.....	22
1.4. Висновки	24
РОЗДІЛ 2 ПРОЕКТУВАННЯ МАТЕМАТИЧНОГО, ПРОГРАМНОГО І АЛГОРИТМІЧНОГО ЗАБЕЗПЕЧЕННЯ СИСТЕМИ МОНІТОРИНГУ І АНАЛІЗУ МЕРЕЖЕВИХ ПОСЛУГ	26
2.1. Математичне забезпечення аналізу ефективності роботи інтернет- провайдера	26
2.1.1. Математичне забезпечення аналізу ефективності мереж.	26
2.1.2. Розрахунок ефективності використання можливостей мережевої інфраструктури.....	28
2.1.3. Математичне забезпечення обрахунку ефективності обслуговування системи.	30
2.1.4. Математичне забезпечення методу черг.....	33
2.2. Алгоритмічне і програмне забезпечення системи моніторингу і аналізу....	35
2.3. Висновки	43
РОЗДІЛ 3 ЗАСТОСУВАННЯ ОБРАНИХ МЕТОДІВ РОЗРОБКИ ДЛЯ СТВОРЕННЯ СИСТЕМИ МОНІТОРИНГУ І АНАЛІЗУ МЕРЕЖЕВИХ ПОСЛУГ	44
3.1. Реалізація збору даних з мережевого моніторингу	44
3.2. Програмна реалізація створення веб-ресурсу	47
3.2.1 Створення клієнтської частини веб-ресурсу	47
3.2.2. Створення серверної частини веб-ресурсу.....	52

3.2.3. Програмна реалізація математичного забезпечення	55
3.2.4. Публікація ресурсу на хостинг	57
3.3. Результати програмної реалізації системи моніторингу і аналізу	60
3.4. Висновки	65
РОЗДІЛ 4 ОХОРОНА ПРАЦІ І БЕЗПЕКА В НАДЗВИЧАЙНИХ СИТУАЦІЯХ	66
4.1 Охорона праці	66
4.2. Безпека життєдіяльності	69
4.2.1. Забезпечення захисту працівників суб'єкта господарювання від іонізуючих випромінювань.	69
4.2.2. Вплив проникаючої радіації ядерного вибуху на надійність роботи електронного обладнання.....	71
4.3 Висновки	73
ВИСНОВКИ	74
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ	76
Додаток А	79

ВСТУП

Актуальність теми. З безперервним зростанням використання інтернет-мереж, інтернет-провайдери існують в складному і конкурентному середовищі. У зв'язку з постійними намаганнями держави-агресора атакувати цифрову і енергетичну інфраструктуру України, інтернет-провайдери опиняються в дедалі складнішому становищі і постають перед перешкодами для свого існування.

Тому зараз постачальники мережевих послуг не можуть не оптимізувати свої ресурси, вибудувувати стратегії і шукати нові можливості для розвитку. Все це неможливо без попереднього проведення аналітики своєї роботи. Компанії потребують засоби для моніторингу і аналізу своїх можливостей. Оскільки не існує готових систем для обчислення і аналітики даних, то важливо розробити програмне середовище, що буде одночасно проводити моніторинг мереж, збирати дані про клієнтів і надавати можливості для обчислення отриманої інформації. Методи математичного моделювання, теорії масового обслуговування, теорії ймовірності і обчислювальної математики забезпечать об'єктивні можливості для корпоративної аналітики провайдера. Використовуючи сучасні методи веб-розробки програмне забезпечення буде мати можливості для постійної модернізації і оновлення, тому система буде гнучкою для підтримання своєї актуальності довгий період часу.

Мета і задачі дослідження. Метою кваліфікаційної роботи є розробка математичного та програмного забезпечення для моніторингу й аналізу ефективності роботи інтернет-провайдера. Для досягнення визначеної мети необхідно виконати такі задачі:

- проаналізувати методи для збору інформації для аналізу ефективності постачання мережевих послуг;
- здійснити огляд існуючих рішень, обрати засоби розробки проаналізувати їх недоліки і переваги;

- з урахуванням наукових досліджень, розробити математичні методи для обрахунку параметрів і характеристик ефективності постачання мережевих послуг;
- спроектувати алгоритм роботи системи і запропонувати способи покращення застосованого програмного забезпечення;
- програмно реалізувати систему аналізу і моніторингу поєднуючи різні технології розробки.

Об'єкт дослідження. Процес моніторингу і аналіз ефективності надання мережевих послуг

Предмет дослідження. Математичне і програмне забезпечення систем аналізу ефективності роботи інтернет-провайдера

Наукова новизна.

- Уперше запропоновано набір параметрів для обчислення ефективності постачання мережевих послуг отриманих із різних джерел, що обчислюються за допомогою розробленого і реалізованого математичного забезпечення, яке розвинуло застосування відомих математичних методів.
- Поєднано різні методи розробки системи моніторингу і аналізу мережевих послуг у спосіб, в який раніше не були застосовані, що дозволило забезпечити комплексність системи і надало математичні і програмні можливості для різнобічного аналізу постачання мереж.

Практичне значення одержаних результатів. Налаштовано систему моніторингу телекомунікаційних мережами, розроблено виведення інформації з програми в базу даних і сповіщення в чат-боті. Розроблено веб-ресурс для комунікації з клієнтами і отримання від них даних важливих для аналізу. Розроблено засоби для поєднання всієї отриманої інформації в одній системі керування базами даних. Створено програмні засоби для проведення математичного обчислення і порівняльного аналізу різних відділень інтернет-провайдера.

Публікації. Результати дослідження були опубліковані, як тези конференцій у статтях: «Методи проведення моніторингу і аналізу мережевої

інфраструктури Інтернет провайдерами», і «Вибір технологій розробки веб-ресурсу моніторингу мережі Інтернет провайдерами» і апробовані на на XI науково-технічній конференції «Інформаційні моделі, системи та технології».

Структура роботи. Кваліфікаційна робота складається зі вступу, чотирьох розділів, висновків по роботі і переліку використаної літератури. Кваліфікаційна робота містить 82 сторінки, з них 69 сторінок основного тексту, 34 рисунки, 1 таблиця, 1 додаток і 29 найменувань переліку літератури.

РОЗДІЛ 1

ОГЛЯД ТА АНАЛІЗ ІСНУЮЧИХ РІШЕНЬ І ПРОБЛЕМ МОНІТОРИНГУ І АНАЛІЗУ МЕРЕЖЕВИХ ПОСЛУГ

1.1 Огляд джерел отримання даних для аналізу постачання мережеских послуг

Інтернет-провайдери це ключові компанії у сучасному цифровому світі, що забезпечують підключення до глобальної мережі для мільйонів користувачів. Основна функція ІП - це забезпечення доступу до Інтернету. Вони є посередниками між користувачами та глобальною мережею, надаючи їм можливість переглядати веб-сайти, обмінюватися електронною поштою, використовувати соціальні мережі та інші інтернет-сервіси. Зазвичай, ІП- це комерційні компанії, які надають послуги для підприємств та домогосподарств, рідше вони можуть бути неприбутковими організаціями, які обслуговують специфічні групи користувачів. Але оскільки більшість компаній є комерційними, а сфера постачання мережеских послуг є доволі конкурентною, то ІП повинні постійно збільшувати свою ефективність і розвиватись.

Без аналізу ефективності своєї роботи ІП не зможуть прогнозувати свою роботу, розширювати зони покриття власних мереж і створювати стратегії розвитку компанії. Існує багато методів аналізу даних, які можуть бути застосовані для аналітики ефективності, але спочатку треба отримати всю необхідну інформацію[1].

Джерела отримання інформації для аналізу ефективності постачання мережеских послуг:

- Моніторинг мережі;
- Внутрішні бази даних;
- Дані з системи обслуговування клієнтів;
- Веб-аналітика;
- Відкриті дані і дані з соціальних мереж.

Моніторинг є найбільш очевидною і надзвичайно важливою складовою аналізу ефективності постачання мережевих послуг. Системи мережевого моніторингу є одним з предметів даного дослідження, тому що його роль важко переоцінити. Відсутність такої системи може призвести до ряду серйозних проблем, які впливають на якість обслуговування і задоволення користувачів. При відсутності моніторингу і аналізу мережі в ІІ можуть виникнути безліч проблем. Без системи моніторингу провайдер може не вчасно виявляти проблеми в роботі обладнання, а відмови або неполадки можуть виникнути безперервно, і відновлення послуг може затриматися. Під час відмов часто виникають фінансові втрати через відсутність послуг для користувачів.. Без моніторингу це може призвести до непередбачуваних фінансових труднощів для компанії. Відсутність моніторингу може призвести до невиявлення атак або аномальної активності, що загрожує безпеці мережі [2].

Таким чином, системи моніторингу є не тільки необхідним джерелом даних для аналізу, але й незамінною складовою всієї програмної інфраструктури мережевого постачальника. Для аналізу даних про технічні аспекти постачання мережевих послуг отримання даних з моніторингу є незамінним. Можна визначити, які компоненти мережі потребують заміни у зв'язку з частими поломками чи невідповідності до характеристик у зв'язку з застарілістю чи браком. За допомогою моніторингу легко визначити найефективніші компоненти мережі і враховувати ці дані для планів масштабування мереж. Маючи всі ці дані, інтернет провайдер все ще не буде мати достатньо інформації для комплексного аналізу своєї роботи, оскільки технічні характеристики не є єдиним важливим аспектом аналітики даних.

Внутрішні бази даних зазвичай містять фінансову інформацію та інформацію про клієнтів. Сюди входять: дані про прибутки, витрати, кількість користувачів, кількість нових підписок, відмови від послуг та іншу фінансову інформацію зі своїх баз. Вони не потребують частого оновлення і формуються зазвичай з квартальних звітів про роботу чи систем білінгу. Ці дані є основою

для всіх систем призначених для аналітики ефективності інтернет-провайдера і кожна математична модель повинна мати таку інформацію.

Дані з системи обслуговування клієнтів. Маючи технічні дані про інфраструктуру ІІ, його внутрішні дані не можна забувати про необхідність комунікації користувачами. Налагоджуючи можливість спілкування з клієнтами провайдер вже збільшує свою ефективність. Але це дозволить не тільки підвищити якість своїх послуг і покращити ймовірність збільшення користувацької бази, а й отримати незамінні дані для аналізу своєї ефективності. Часто така інформація може здатись занадто суб'єктивною, але потрібно правильно налагодивши комунікацію і коректно вести її облік. Моніторинг прекрасний інструмент для виявлення технічних проблем та помилок, але тільки спілкуючись з клієнтом провайдер зможе дізнатись про нестандартні і неочевидні проблеми своєї роботи. Такі дані також допоможуть краще розуміти приріст нових або потенційних користувачів мережевими послугами, а це незамінний фактор для будь якої аналітики. Серед програмних рішень для такої комунікації здебільшого обирають розробку веб-ресурсу, що дозволить вести збір необхідної інформації в базу даних [1].

Веб-аналітика - це процес збору, обробки та аналізу даних про відвідувачів веб-сайту та їх поведінку. Вона включає в себе різні методи та інструменти: використання cookies для відстеження даних та генерації звіту на основі активності користувача, геотаргетинг, що полягає в визначенні геолокації відвідувачів веб-сайту, процес відстеження мобільних телефонів, а також відстеження прокрутки та теплові карти для розуміння того, як користувачі взаємодіють з сайтом.

Існують різні інструменти веб-аналітики, такі як Google Analytics, Mixpanel та Adobe Analytic. Ці програми надають повний фреймворк вимірювань, включаючи код для початку відстеження користувачів на веб-сайті, зберігання та обробку даних, а також інструменти візуалізації.

Цей метод дозволяє отримувати дані про користувачів без ведення прямої комунікації з ним. Основною проблемою є те, що для можливості аналітики

потрібно зібрати велику кількість таких даних. Тому поки веб-ресурс не має достатню популярність, такий метод застосовуватись не може.

Відкриті дані і дані з соціальних мереж. Відкриті дані - це великі набори даних, доступні для всіх, хто має підключення до Інтернету. Цей метод може включати в себе все, від публічних даних, зібраних урядовими установами, до економічних тенденцій від банків та фінансових конгломератів. Інтернет-провайдери можуть використовувати відкриті дані для аналізу тенденцій та виявлення нових можливостей. Під даними з соціальних мереж розуміється інформація, яку користувачі публічно діляться, і місцезнаходження та біографічні дані. Компанії зібрали дані з соціальних медіа для цільової реклами через рекламу за допомогою слідкування або поведінкову рекламу. Методі збору відкритих даних і даних з соціальних мереж є ще більш ресурсозатратним ніж веб-аналітика. Разом з складністю аналізу великої кількості таких даних результат такого аналізу може бути не надто корисним для невеликих ІП. Тому такий метод застосовується зазвичай тільки великими постачальниками, що мають покриття в багатьох регіонах країни.

Проблемою аналізу є те, що готові математичні і програмні рішення кожного з методів аналізу мало поєднані між собою. Програмне забезпечення з моніторингу не дозволяє поєднувати аналітику його даних з зовнішніми даними провайдера і часто пропонує доволі вузький аналіз для кожного конкретного пристрою в складі мережі. Не існує й готових математичних рішень, які могли б коректно прогнозувати і аналізувати необхідні дані і підлаштовуватись під необхідний контекст.

Таким чином це дослідження спрямоване на створення комплексної системи, яка буде поєднувати в собі різні технології і забезпечувати методи для збору і аналізу даних в рамках одного проекту. Порівняльний аналіз математичних показників буде застосований, як метод, що повинен вказати інтернет-провайдеру на проблеми у різних регіональних відділеннях компанії. Буде аналізуватись 2 регіональних відділення у містах Збараж і Хоростків. Дані для обчислення будуть отриманні від програмного забезпечення з моніторингу

мереж, що буде аналізувати кожну з цих мереж і окремі її компоненти. А крім технічних показників будуть обраховуватись і більш суб'єктивні отримані від системи взаємодії з клієнтами. Після створення всієї системи інтернет-провайдер може робити висновки про різні аспекти своїх відділень і мати можливість для прогнозування майбутніх планів розвитку. Важливим фактором системи буде гнучкість і готовність до масштабування, тому до методів обрахунку і порівняльного аналізу у майбутньому можуть бути додані і показники отримані від веб-аналітики і відкритих даних з інших масивів інформації.

1.2 Аналіз рішень з моніторингу мереж

Важливо якісно проаналізувати програмне забезпечення для здійснення моніторингу. Для початку потрібно розібратись як працюють програми для моніторингу і який результат цим програм очікує інтернет-провайдер. Моніторинг мережі — це систематичне спостереження за роботою комп'ютерної мережі з метою забезпечення її ефективності, стабільності та безпеки.

Системи моніторингу мереж використовують декілька протоколів забезпечення своїх функцій та. Вони використовують SNMP для отримання інформації про пристрої та їх стан [3]. ICMP використовується для відправки запитів “ping” до пристроїв, щоб перевірити їх доступність. DNS використовується для визначення імен пристроїв. TCP використовується для встановлення з'єднань з пристроями та обміну даними.

Щодо вимог до ПЗ, то програми моніторингу повинні стежити за станом всієї бездротової мережі і відстежувати і порівнювати реальні швидкості і рівні сигналів. Рішення з моніторингу можна знизити ризик появи шкідливих програм і нелегального підключення, оскільки вони дозволяють оперативно реагувати на аномальну діяльність в межах локальної мережі, мати можливість бачити мережеві процеси і автоматизувати частину рутинної діяльності адміністратора. Програми моніторингу дозволяють відстежувати ряд важливих для роботи

маршрутизатора факторів, таких як статус роутера, ступінь навантаження, трафік, рівень сигналу, останній доступ

Для порівняльного аналізу представлені три популярних сервіси для моніторингу мережевого обладнання і трафіку.

Observium - це платформа для моніторингу мереж, яка надає потужні можливості моніторингу, використовуючи технології, такі як SNMP, IPMI, JMX, VMware та користувацькі перевірки. Observium дозволяє збирати метрики, виявляти стани проблем, надсилати сповіщення та відображати зібрані дані. Observium виділяється підтримкою широкого спектру пристроїв, платформ та операційних систем, таких як NetApp, Brocade, FreeBSD, Linux, Windows, Cisco, HP, Dell та Juniper. Однак, Observium може бути складним для налаштування та використання, тому для роботи з ним доволі великий поріг входження серед адміністраторів [4].

Wireshark є, можливо, найпопулярнішою системою моніторингу мережевого трафіку. Ця програма дозволяє перехоплювати і аналізувати дані, що пересилаються по мережі. Wireshark використовує ті самі технології, як і Observium і дозволяє адміністраторам переглядати деталі кожного пакета, що проходить через мережу, в реальному часі. Відомою перевагою Wireshark є інтерфейс, який користувачі вважають інтуїтивно зрозумілим. Це полегшує використання для широкого кола фахівців. Ще одна перевага Wireshark полягає у підтримці фільтрів і кодування кольорами [5].

MikroTik The Dude – це популярний в Україні інтегрований інструмент для моніторингу та управління мережами, розроблений для обладнання MikroTik RouterOS. Очевидною перевагою The Dude є фірма виробник, оскільки MikroTik займає велику частку ринку мережевого обладнання. В The Dude вбудована підтримка пристроїв MikroTik, що робить його оптимальним вибором для тих, хто використовує продукцію даного виробника. Серед своїх конкурентів програма виділяється ефективністю своїх функцій з автоматичного сканування пристроїв в межах вказаних підмереж. Основною проблемою The Dude є відсутність оновлень ПЗ від розробника [6].

Порівнюючи ці системи моніторингу мереж, можна зробити висновок, що вони мають доволі схожі характеристики. Ці ПЗ є здебільшого безкоштовними, використовують однакові технології, виконують здебільшого однакові функції. В кожного з них є свої переваги, та часто вони не такі значні. Тому при виборі програм для моніторингу інтернет-провайдери користуються суб'єктивними аргументами: досвід персоналу у роботі з ПЗ і сумісність з апаратним забезпеченням.

Для цього проекту було вибрано користуватись програмним забезпеченням MikroTik The Dude. Оскільки розглянуті програми мережевого моніторингу не мають принципових відмінностей в характеристиках, то перевага The Dude в виконанні функції автоматичного сканування зіграла важливу роль в обранні її як засобу для реалізації дослідження. Але основним фактором стала велика кількість операторів з досвідом роботи цієї системи в Україні і її сумісність з пристроями від MikroTik.

1.3. Аналіз рішень з розробки програмного забезпечення для розробки системи збору і аналізу

Для початку треба визначити, який тип програмного забезпечення для аналізу даних є пріоритетнішим для даного проекту: додаток чи веб-ресурс. Створити власне програмне забезпечення для системи аналізу ефективності роботи інтернет-провайдера потрібно для виконання двох задач: отримання даних від клієнтів та програмної реалізації математичного обрахунку ефективності. Для першої задачі десктопні додатки не будуть ефективними, оскільки користувач повинен мати можливість оперативно і просто отримати доступ до ресурсу інтернет-провайдера. Тому для зв'язку з клієнтами очевидним засобом є веб-розробка. Але для створення системи з аналізу і обрахунку ефективності такої принципової різниці в методах розробки немає. Потрібно розібратись у різниці методів розробки веб-ресурсів і десктопних додатків, щоб визначити який метод обрати.

Веб-розробка зосереджується на створенні веб-сайтів та веб-додатків, які можуть бути доступні через будь-який веб-браузер. Веб-розробники використовують технології, такі як: HTML, CSS та JavaScript, для створення структури, стилю та функціональності веб-сайтів. Для створення веб-ресурсів з більшою функціональністю використовуються серверні мови програмування та фреймворки. Веб-розробка зазвичай поділена на фронтенд (клієнтська частина) та бекенд (серверна частина) розробку. Веб-ресурси допомагають розробникам донести і обробляти інформацію, що вони доступні та зрозумілі для загальної публіки.

Розробка десктопних додатків зосереджується на створенні програм, які встановлюються на конкретних пристроях, таких як персональні комп'ютери. Розробники настільних додатків використовують мови програмування, такі як: C++, Java, Python, або .NET, для створення додатків, які можуть використовувати специфічні для пристрою функції [7].

Вибір між веб-розробкою та розробкою настільних додатків залежить від ваших конкретних потреб та вимог. Серед переваг додатків над веб ресурсами є продуктивність і функціональність. Настільні додатки працюють швидше, ніж веб-сайти, оскільки вони виконуються безпосередньо на пристрої і мають меншу кількість компонентів. Веб-розробник для отримання таких самих можливостей у функціоналі, як у додатків, повинен застосовувати більше технологій і затрачати більше ресурсу для їх поєднання. У веб-розробці також є низка переваг. Наприклад, оновлювати веб-ресурс під нові задачі і функції набагато легше ніж десктопний додаток. Також немає потреби розробляти різні версії програми під різне апаратне і системне забезпечення.

Програмна реалізація математичних задач для аналізу ефективності роботи інтернет-провайдера не потребує обширного функціоналу ,а також не затрачає таку кількість ресурсів, щоб отримати проблеми з продуктивністю. Таким чином, логічним рішенням є зупинитись на розробці веб-ресурса, на базі якого можна виконувати 2 необхідні задачі: комунікація і збір даних клієнтів, а також обрахунку ефективності постачальника мережевих послуг. Наступним

кроком потрібно оглянути рішення для веб-розробки, щоб мати можливість обирати оптимальні технології.

1.3.1. Спосіб розробки веб-ресурсу за допомогою технології WordPress Stack. WordPress Stack є одним з найпопулярніших наборів технологій для розробки веб-додатків. Він складається з трьох основних компонентів: WordPress, PHP та MySQL.

WordPress- це, мабуть, найпопулярніша система управління контентом, яка дозволяє користувачам легко створювати, редагувати та публікувати вміст на веб-сайті. WordPress є відкритим програмним забезпеченням, тому він безкоштовний для використання та модифікації. Загалом WordPress Stack акумулює в собі всі основні переваги і недоліки систем управління контентом, тому на його прикладі можна буде робити висновки про наборів технологій, які включають такі системи.

Система управління контентом (Content Management System, CMS) - це програмне забезпечення, яке допомагає користувачам створювати, керувати та змінювати вміст на веб-сайті застосовуючи його можливості і не вимагаючи в розробника приділяти велику кількість часу на написання коду. CMS забезпечує всю базову інфраструктуру для розробника, в деяких випадках навіть не вимагаючи його написання верстки сайту за допомогою HTML та CSS, а користуючись готовими шаблонами системи.

PHP є серверною мовою програмування, яка використовується для створення динамічних веб-сторінок. Вона використовується для керування динамічним контентом, базами даних, відстеженням сесій. PHP є дуже популярною мовою, що застосовується багато років, а підтримку рівня її технологічності забезпечує розвиток її фреймворків. PHP може створювати, відкривати, читати, записувати, видаляти та закривати файли на сервері. Вона також може збирати дані з форм, відправляти та отримувати файли cookie,

додавати, видаляти, модифікувати дані в базі даних, контролювати доступ користувачів та шифрувати дані PHP використовується в WordPress для обробки запитів від користувачів та відправки відповідей назад до браузера.

MySQL є системою управління реляційними базами даних. Реляційні бази даних зберігають дані в табличному форматі як рядки та стовпці. Вони визначають відносини між даними за допомогою кількох таблиць і є простими для розуміння і управління для адміністратора бази даних. MySQL є дуже популярною системою керування і відзначається своєю надійністю. Вона забезпечує швидкість виконання команд, гнучкість і підтримує переведення даних в інші SQL бази. MySQL використовується в WordPress для зберігання всіх даних, таких як публікації, коментарі та іншу інформацію від адміністраторів чи користувачів сайту.

Фронтенд в WordPress включає HTML, CSS та JavaScript. HTML створює основну структуру будь-якої сторінки. CSS визначає колірну схему, типи шрифтів та макет. JavaScript забезпечує логічну частину ресурсу, його функціонал. Розробник може використовувати шаблони, які WordPress пропонує для створення структури веб-ресурсу, але кращим рішенням написання власного коду і конвертація його в CMS.

Бекенд частина в WordPress вимагає застосування плагінів. Плагін WordPress - це додатковий шматок програмного коду, який, коли додається до проекту і створює нові функції. Наприклад, форми для комунікації з клієнтом створюються за допомогою плагінів. Код плагінів в WordPress Stack розробляють на мові PHP.

Очевидною перевагою WordPress Stack є його інтуїтивно зрозумілий інтерфейс, тому розробка веб-сайтів за допомогою цієї платформи не вимагає високих навичок та знань у програмуванні. CMS забезпечує інфраструктуру для розробки і не потребує використання великої кількості технологій.

Проте, функціонал WordPress Stack обмежений в рамках своєї CMS і не дозволяє серйозного масштабування, бо стандартні функції не можуть задовільнити всі вимоги. Присутні проблеми з оптимізацією, тому що з кожною

новою функцією і, відповідно, плагіном продуктивність веб-ресурсу зменшується. Також важливою проблемою є безпека, у кожній CMS є стандартні вразливості, які можуть використовувати зловмисника, а оскільки WordPress є однією з найпопулярніших систем, то існує багато способів проводити атаки на ресурс [8].

1.3.2. Спосіб розробки веб-ресурсу за допомогою технологій LAMP та WAMP Stacks. Технологія LAMP є одним з найпоширеніших стеків програмного забезпечення для багатьох найпопулярніших веб-додатків. LAMP - це аббревіатура, що включає в себе чотири технології: Linux для операційної системи, Apache HTTP Server, MySQL для системи управління реляційними базами даних та PHP, як мову програмування.

WAMP ж прямим аналогом LAMP, але включає в себе операційну систему Windows. Однак, варіант розробки за допомогою Linux є популярнішим, ймовірно через те, що деякі функції або налаштування, які доступні в Linux, не доступні або працюють по-іншому в Windows

Фронтенд веб-ресурсів, створених за допомогою LAMP, зазвичай розробляється за допомогою мови програмування PHP. Вона вбудована в Apache, що дозволяє PHP-скриптам взаємодіяти безпосередньо з веб-сервером. На клієнтському рівні розробники, що працюють з LAMP використовують мову JavaScript, але їй приділяється небагато уваги і зазвичай не застосовують фреймворки. Тому часто фронтенд частина LAMP-ресурсів не є перевагою.

Бекенд веб-додатків, створених за допомогою LAMP, зазвичай розробляється за допомогою мови програмування PHP та бази даних MySQL. Оскільки PHP є серверною мовою програмування, то розробник використовує всі переваги мови для забезпечення надійної роботи бекенд частини веб-ресурсу. PHP використовується для обробки даних та взаємодії з базою даних. При

бажанні також можна використовувати іншу реляційну базу даних, оскільки вони часто є взаємозамінні і схожі за синтаксисом.

LAMP є відкритим та гнучким рішенням для розробки веб-додатків. Він надає розробникам можливість використовувати відкрите програмне забезпечення для створення надійних, безпечних та ефективних веб-ресурсів. Але, у зв'язку з акцентом LAMP стеку на серверній мові програмування, клієнтська частина проекту може виглядати застарілою [9].

1.3.3. Спосіб розробки веб-ресурсу за допомогою технологій JavaScript Full Stacks. JavaScript Full Stacks — це загальна назва для одразу трьох технологій- MERN(MongoDB, Express, React, Node.js), MEAN(MongoDB, Express, React, Node.js) і MEVN (MongoDB, Express, React, Node.js). Ці стеки використовують JavaScript як основну мову програмування для розробки як клієнтської (фронтенд), так і серверної (бекенд) частин веб-додатків. Кожна з технологій працює з фреймворками Express і Node.js, а також з базою даних MongoDB. Відрізняються вони основною частиною технологіями для фронтенд-розробки.

React— це найпопулярніша серед представлених бібліотек JavaScript, розроблена Facebook, яка набула популярності завдяки своїй гнучкості, оптимізації продуктивності та швидкому впровадженню великими технологічними компаніями. Він дуже зручний для адаптації верстки веб-сайтів, оскільки пропонує HTML-подібний синтаксис JSX. Відомий React великою кількістю доповнень, які значно розширюють його функціонал [10].

Angular— це фреймворк для розробки веб-додатків, який був розроблений першим, серед представлених конкурентів. Він надає розробникам потужні інструменти для створення складних односторінкових додатків. Angular використовується разом з мовою програмування TypeScript. Одною з його переваг є Model-View-ViewModel — технологія дуже корисна для великих

проектів, тому що дозволяє розробникам координувати свою роботи нуд одними розділами [11].

Vue— це прогресивний JavaScript фреймворк, який забезпечує гнучкість і швидкість розробки. Він називається прогресивним фреймворком, тому що ви можете розширити його функціональність за допомогою офіційних та сторонніх пакетів, таких як Vue Router або Vuex, щоб перетворити його на справжній фреймворк. Його технології є доволі новітніми і прогресивними, але завдяки складності розуміння Vue не здобув великою популярності.

Обравши фронтенд технологію, потрібно розібратись з бібліотеками, що відповідають за бекенд частину. Node.js— це асинхронне середовище виконання JavaScript, яке працює на серверній стороні додатку для керування певними процесами. Node.js використовується для розробки серверних та мережевих додатків. Здатність виконувати код JavaScript на сервері часто використовується для генерації динамічного вмісту веб-сторінки перед відправкою сторінки в браузер користувача. Express— це бібліотека Node.js. Express працює поверх функціональності веб-сервера Node.js, щоб спростити його API та додати корисні нові функції. Він спрощує організацію функціональності вашого додатку за допомогою проміжного програмного забезпечення та маршрутизації. Він додає корисні утиліти до HTTP-об'єктів Node.js та сприяє відтворенню динамічних HTTP-об'єктів.

За бази даних у яких зберігаються дані, які використовує веб-ресурс відповідає MongoDB— система керування нереляційними базами даних. Нереляційні бази даних зберігають дані в гнучкому форматі, такому як пари ключ-значення, документи або графи. Це робить системи керування нереляційними базами даних зручними для управління великою кількістю інформації. MongoDB працює з форматом JSON, який часто використовується для роботи веб-розробників і добре взаємодіє з мовою JavaScript. При управління базами даних MongoDB застосовує JavaScript-подібний синтаксис.

Головна перевага JavaScript Full стеків— використання однієї мови програмування- JavaScript. Різні бібліотеки цієї мови відповідають за фронтенд і

бекенд частини ресурсу і технологічно не поступаються мовам програмування, що спеціалізуються тільки на одній сфері розробки. JavaScript Full стеки— гнучкі, готові до масштабування і чудово оптимізовані [9]. Ці стеки не включають CMS, але вони надають всі необхідні інструменти для створення веб-додатків.

Для даного дослідження було прийнято рішення використовувати набір технологій MERN. Хоча для забезпечення взаємодії з програмами моніторингу краще підходять стеки, що взаємодіють з релятивними базами даних, все ж переваги JavaScript Full переважають цей недолік. Недостатність технологічності LAMP і Wordpress не дадуть достатність можливостей ні для майбутнього масштабування проекту, ні для повної реалізації всіх завдань цієї роботи у зв'язку з недостатньою кількістю нових готових рішень і доповнень. Саме через велику кількість різноманітних можливостей було прийнято рішення використовувати для фронтенд-частини фреймворк React. Багато проблем, що виникають під час розробки складних проектів можуть бути вирішені простим під'єднанням додаткових модулів MERN і використанням технологій, що вони забезпечують.

1.4. Висновки

У даному розділі одержано наступні результати:

1. Наведено аргументи про важливість забезпечення моніторингу і аналізу ефективності роботи постачальника мережевих послуг. Проведено огляд джерел отримання даних для аналізу ефективності, таких як: моніторинг мережевої інфраструктури, використання корпоративних баз даних, система обслуговування клієнтів, веб-аналітика, відкриті дані і дані з соціальних мереж. Зроблено висновки про доцільність створення системи, що буде отримувати, порівнювати і обраховувати дані отримані за допомогою використання засобів для моніторингу і систем комунікації з клієнтами.

2. Проведено огляд програмного забезпечення з моніторингу мережевих послуг, наведено інформацію необхідну для аналізу програм. Виявлено схожі можливості і характеристики популярних рішень з моніторингу таких як: Observium, Wireshark і The Dude. Через сумісність з апаратним забезпеченням і функціями автоматичного сканування було прийнято рішення використовувати ПЗ The Dude.

3. Розглянуто методи веб-розробки і розробки настільних додатків, зроблено висновки про доцільність використання технологій веб-розробки. Проаналізовано різні засоби розробки веб-ресурсів серед яких: набори технологій Wordpress, LAMP і JavaScript Full. Розглянуто аргументи на користь застосування кожної технології, зроблено висновок про доцільність використання JavaScript Full стеку, зокрема MERN, через його переваги у технологічності і розроблених готових рішеннях

РОЗДІЛ 2

ПРОЕКТУВАННЯ МАТЕМАТИЧНОГО, ПРОГРАМНОГО І АЛГОРИТМІЧНОГО ЗАБЕЗПЕЧЕННЯ СИСТЕМИ МОНІТОРИНГУ І АНАЛІЗУ МЕРЕЖЕВИХ ПОСЛУГ

2.1. Математичне забезпечення аналізу ефективності роботи інтернет-провайдера

2.1.1. Математичне забезпечення аналізу ефективності мереж. Для аналізу ефективності роботи телекомунікаційних джерел інтернет-провайдери використовують програмні засоби для моніторингу. Вони дозволяють отримати багато метрик і параметрів, що дадуть змогу порівнювати мережі різних регіональних відділень, різного апаратного забезпечення, а також бачити зміни в мережі. Проте часто, щоб зібрати дані для аналізу необхідно спочатку провести математичні обчислення цих показників за допомогою вихідних даних програм моніторингу. Кожен постачальник мережеских послуг сам визначає яким характеристикам надати пріоритет для спостереження і аналізу. Декілька важливих параметрів і способи їх обчислення будуть представлені нижче.

Пропускна здатність— це найбільш очевидна характеристика ефективності мережі і дуже важлива, оскільки низькі значення можуть призвести до повільної передачі даних та погіршення якості обслуговування. Це кількість даних, які можуть бути передані через мережу за певний проміжок часу. Тобто пропускна здатність обчислюється шляхом відношення об'єму даних, до часу за який вони передаються в мережі [12].

Наступною важливою характеристикою телекомунікаційної мережі можна визначити відсоток втрат пакетів. Оскільки втрачені пакети часто потребують повторного надсилання, що може спричинити затримки, то високий рівень втрат пакетів може бути ознакою проблем з мережею, таких як

перевантаження або нестабільні з'єднання [2]. Ця характеристика може бути обрахована такою формулою:

$$PLR = \frac{n_l}{n_s} \times 100\%,$$

де: PLR— показник втрачених пакетів;

n_l — втрачені пакети

n_s — загальна кількість пакетів.

Важливим показником ефективності мережі є час простою, що вказує на період часу, протягом якого система не доступна для роботи. Довгі періоди простою вказують на проблеми зі стабільністю, надійністю і продуктивністю системи. Цей показник легко розрахувати віднявши загальний час роботи системи від періоду безперебійної роботи. За допомогою нього можна обчислити показник ефективності виражений формулою:

$$E_t = \frac{T_w - T_p}{T_w} \times 100\%,$$

де: E_t — ефективність мережі за часом роботи;

T_w — час роботи системи;

T_p — час простою.

Ще однією важливою характеристикою прямо пов'язаною з часом є затримка. Затримка означає час, який потрібен для передачі пакета даних від джерела до призначення. Без визначення значень цього показнику і прийняття мір для її зменшення система буде передавати дані все з меншою швидкістю і гіршою якістю. Затримка визначається різницею часу відправи і часу отримання[2].

Основною функцією моніторингу мережевого обладнання є виявлення помилок в роботі системи. Тому кількість помилок, про які сповіщає програма також може застосовуватись як важлива характеристика мережевого

обладнання. Вираховувавши кількість помилок за певний період часу, можна порівнювати надійність роботи окремих компонентів мережі і таким чином робити висновки щодо вибору апаратного забезпечення, або якості його обслуговування персоналом. Також можна провести агреговану оцінку надійності мережі, розрахувавши відношення між загальною кількістю помилок і кількістю пристроїв. Таким чином можна буде отримати загальну метрику для порівняння ефективності обслуговування або якості обладнання різних мереж.

2.1.2. Розрахунок ефективності використання можливостей мережевої інфраструктури. Перед постачальником мережевих послуг завжди стоїть задача-оптимізувати свої ресурси. Важливим показником для цієї задачі є завантаженість мережі. Часто інтернет-провайдер повинен приймати рішення-коли потрібно модернізувати свою інфраструктуру, оскільки вона не справляється з тим об'ємом роботи, яка на неї покладена. Наприклад, коли оптоволоконний кабель, під'єднаний до комутатора здатний надавати певне значення пропускну здатності, то, на перший погляд, обраховувати граничні можливості кабеля потрібно порівнявши значення пропускну здатності і кількість клієнтів помножену на максимальну пропускну здатність, яку вони можуть отримати. Тобто, якщо тариф для користувачів надає послуги до 100 мбіт/с, то максимальна кількість клієнтів має бути не більше ніж 10 осіб, оскільки наступні перевищать ліміт інфраструктури наданої оптоволоконним кабелем. Але на практиці, такого майже ніколи не буває. Усі підключені абоненти не використовують трафік на повну потужність, тим більше усі одночасно. За допомогою програм для моніторингу мереж можна побачити фактичну пропускну здатність, яку використовують поточні клієнти. Для прийняття рішень про вдосконалення інфраструктури потрібно знати ліміт підключень нових користувачів. Завдання полягає у тому, щоб з'ясувати, скільки ще клієнтів можна додати до мережі без прокладання нового оптоволоконна. Для початку після визначення фактичних

значень використаного трафіку, потрібно визначити який об'єм припадає на одного абонента. Це можна дізнатись за такою формулою:

$$M = \frac{B_u}{C_0},$$

де: B_u — пікове значення фактичного використання трафіку;

C_0 — початкова кількість користувачів.

Після визначення яку максимальну кількість трафіку споживає 1 абонент потрібно від загального значення доступної пропускної здатності відняти T_u . Таким чином можна отримати значення T_a - пропускна здатність мережі не використана користувачами. Маючи значення M і T_a можна визначити максимальну кількість користувачів, яка може користуватись наявною мережевою інфраструктурою. Також маючи загальний показник пропускної здатності можна розрахувати ефективність використання ресурсів системою [12] використовуючи таку формулу:

$$E_l = \frac{B_u}{B_a} \times 100\%,$$

де: E_l — ефективність використання ресурсів системою;

B_a — максимальні можливості використання трафіку.

Для того, щоб визначити час коли кількість користувачів досягне моменту, коли інфраструктура мережі не зможе забезпечувати всіх абонентів послугами потрібно отримати ще декілька значень параметрів: R -приріст нової кількості клієнтів за період часу, L -скорочення кількості клієнтів за період часу. Ці параметри можна отримати під заявок користувачів на підключення до мережі, залишених на веб-ресурсі, а також з баз даних провайдера. Знаючи всі ці дані можна створити математичну модель, яка буде рахувати через який

проміжок часу приріст клієнтів змусить збільшувати пропускні спроможності мережі.

Потрібно використати рівняння

$$T = ((R - L) \times t + C_0) \times M,$$

де: T — максимальне значення пропускної здатності забезпечене мережевою інфраструктурою;

t — час за який кількість користувачів досягне максимально можливого значення.

Далі можна отримати формулу призначену для розв'язання рівняння:

$$t = \frac{\frac{T}{M} - C_0}{R - L}, \quad (2.1)$$

Така модель може бути доволі приблизною, зокрема, тому що показник використання даних на одного клієнта може змінюватись, тому важливо оновлювати дані в математичній моделі. Також важливо встановити обмеження на максимальну пропускну здатність забезпечену інфраструктурою при розрахунку, на рівні принаймні 90%, оскільки при непередбачуваних обставинах кількість клієнтів або навантаження на систему може різко зрости і провайдер може не встигнути спрогнозувати необхідність модернізації інфраструктури.

2.1.3. Математичне забезпечення обрахунку ефективності обслуговування системи. Для аналізу ефективності надання послуг інтернет-провайдером потрібно користуватись не тільки технічними даними про стан мережевого обладнання, але й іншими факторами, зокрема тими.

При обчисленні математичної моделі вираженої формулою 2.1 використовувався показник приросту і скорочення клієнтів. Ці значення можна

легко розрахувати маючи дані про загальну кількість клієнтів і дані про приєднання нових. Часто дані про зміну кількості клієнтів обрахують автоматично за допомогою систем білінгу. Дані про нові запити на приєднання будуть допомагати визначати ефективність кожного відділення провайдера[14] залучати нових користувачів.

$$K = \frac{N_n}{N_z},$$

де: K — коефіцієнт залучення нових клієнтів;

N_n — кількість нових клієнтів за період часу у окремого відділення;

N_z — загальна кількість клієнтів відділення.

Також маючи дані про користувачів, що бажають стати абонентами інтернет-провайдера і реальну кількість нових користувачів можна порахувати відсоток клієнтів, що передумали підключатись до послуг.[15]

Цей показник дасть компанії можливість порівнювати клієнтів відсіюється на етапі домовленостей і в подальшому визначати причини. Коли показник відмов в одному регіональному відділенні відчутно більший ніж в інших, це може показувати на низку недоліків, наприклад, що працівники не правильно проводять комунікацію, дані про умови тарифів на ресурсах провайдера застаріли. За допомогою цього показника компанія може звертати увагу на такі проблеми і вирішувати їх.

Отримуючи на веб-ресурсі дані від клієнтів про помилки, що виникають у них при користуванні послугами інтернет-провайдера. Така інформація буде хорошим доповненням до даних про помилки отримані за допомогою програми моніторингу мереж. Наприклад, можна провести обчислення:

$$K_{rp} = \frac{Q_{cp}}{Q_{mp}},$$

де: K_{rp} — коефіцієнт відношення помилок;

Q_{cp} — кількість помилок виявлених користувачами;

Q_{mp} — кількість помилок виявлених користувачами.

Якщо відношення у одного регіонального відділення буде вищим ніж у інших, то це може свідчити про низку проблем. Системні помилки повинні вирішуватись оперативно, до того як користувачі зможуть помітити будь-які проблеми. Повільне вирішення проблем може бути однією з причин високих значень показника Квп. Також це може свідчити про помилки, які не змогла виявити програма для моніторингу мереж, а також про погану комунікацію між абонентами і відділенням інтернет-провайдера.

Ще одним показником, що може свідчити про проблеми у роботі працівників інтернет провайдера можна вираховувати середній час, за який проводиться відновлення роботи мережі. Цей показник можна виконати такою формулою:

$$AT_r = \frac{T_p}{Q_p}, \quad (2.2)$$

де: AT_r — середній час відновлення роботи системи;

T_p — час простою мережі;

Q_p — кількість випадків припинення роботи системи.

У часи проблем з енергетичною системою і можливих блекаутів втрата живлення для системи є типовою складністю, тому до мережевого обладнання повинні бути під'єднані додаткові джерела живлення. Але у разі виявлення довгого періоду може свідчити про проблеми з автоматизацією підключення резервного живлення чи оперативністю роботи працівників.

Отримавши всі показники показники ефективності своєї роботи інтернет-провайдер може, коли це можливо, також розрахувати, які показники він хоче отримувати від своїх відділень. Після цього можна вирахувати відношення

реальних показників до бажаних. Для того щоб мати одне значення що буде узагальнювати всі обчислені характеристики регіонального відділення інтернет провайдера можна розрахувати агрегований показник. Агрегування здійснюється підсумовуванням, групуванням або іншими способами зведення часткових показників в узагальнені, такий показник в даному випадку може використовувати технічні, клієнтські і економічні показники, що не проводяться в цьому дослідженні [16].

2.1.4. Математичне забезпечення методу черг. В галузі математичної статистики і теорії ймовірностей був виведений метод черг, що є ефективним засобом для обрахунку потреб систем масового обслуговування, варіюючи операційні характеристики систем. Він досліджує і обраховує будь-яку послідовність, елементи якої очікують на обслуговування. Для даного дослідження метод черг можна застосовувати для різних даних. Для технічних даних здобутих, наприклад, програмою моніторингу можна використовувати обрахування черг пакетів даних у мережі. Також метод черг може бути доповненням до математичної моделі вираженої формулою 2.1. Використовуючи метод черг можна користуватись не тільки швидкістю надходження нових заявок на підключення, але й можливістю регіональними відділеннями провайдера їх обробки.

За допомогою цього метода можна побудувати багато математичних моделей. Для обчислення основних робочих характеристик базової моделі достатньо знати значення двох основних параметрів моделі: λ - інтенсивність надходження завдань у систему, і μ - інтенсивність обслуговування завдань сервісом [17]. За допомогою цих параметрів можна обчислити і інші характеристики методу черг.

Таблиця 2.1

Робочі характеристики базової моделі

Характеристика	Позначення	Формула
Коефіцієнт завантаження системи	-	λ/μ
Середня кількість завдань у системі	L	$\lambda/(\mu-\lambda)$
Середня кількість завдань у черзі (середня довжина черги)	L_q	$\lambda^2/\mu(\mu-\lambda)$
Середній час перебування у системі	W	$1/(\mu-\lambda)$
Середній час очікування у черзі	W_q	$\lambda/\mu(\mu-\lambda)$
Ймовірність того, що система порожня	P_0	$1-\lambda/\mu$

Кожна з цих характеристик важлива для розуміння та оптимізації систем чергування. Коефіцієнт завантаження системи важливий для визначення ступеню завантаження системи та її ефективності. Середня включає ті завдання, що обслуговуються і ті, що чекають в черзі. Ця характеристика потрібна для оцінки загального навантаження на систему. На відміну від неї характеристика L_q не включає ті завдання, що обслуговуються, але важлива для планування ресурсів. Середній час очікування у черзі- це показник критичний для розуміння задоволення клієнтів і ефективності системи. Ймовірність того, що система порожня, тобто в даний момент система не обслуговує жодного завдання, важлива для визначення необхідних ресурсів і потенційного простою.

Кожна з цих характеристик допомагає зрозуміти, як система черг буде вести себе під різними навантаженнями, і як це впливає на ефективність та задоволеність клієнтів.

Проте слід зауважити, що базова модель передбачає реалізацію записаних у таблиці функцій тільки при умові, що $\lambda < \mu$, тобто працюють при умові виконання експоненціального розподілу. Тому для точніших результатів для обчислення моделей можна застосовувати інші обрахунки, які забезпечуються

моделями черг. Наприклад, формули можна розвинути, обчисливши математичне сподівання і дисперсію. Проте для простих обрахунків метод черг базової моделі все ще буде дуже корисним і придатним для застосування.

2.2. Алгоритмічне і програмне забезпечення системи моніторингу і аналізу.

Після визначення методів збору даних і математичних способів для забезпечення аналізу потрібно визначити принципи розробки системи, що реалізовує мету даного дослідження. Оскільки засоби для реалізації комплексного аналізу ефективності використовують різні технології, як це можна побачити на структурній схема(див. рис. 2.1), то варто розібратись з порядком і особливостями розробки.

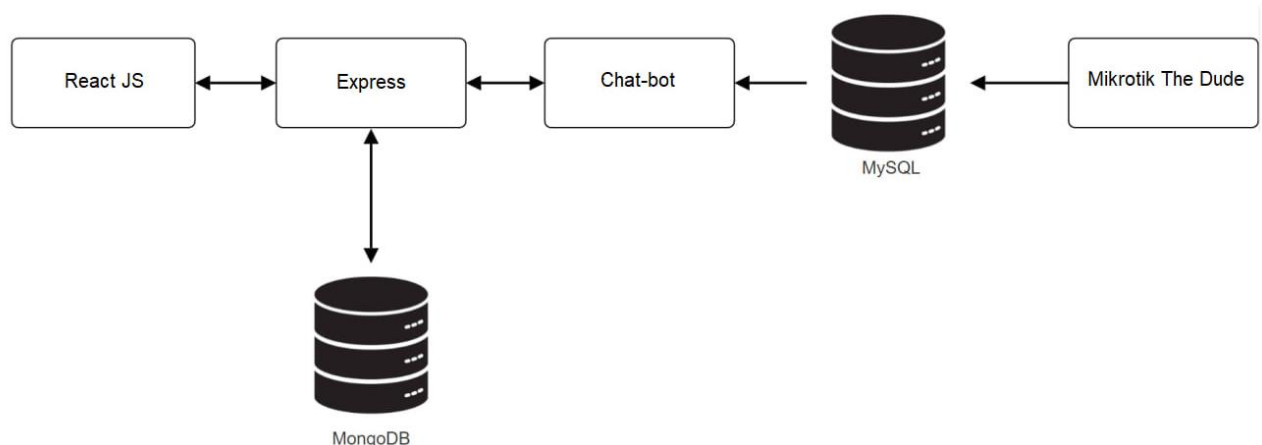


Рис. 2.1. Структурна схема системи з аналізу і моніторингу

Перший метод збору даних буде реалізований за допомогою програми для моніторингу мережевого обладнання The Dude. Моніторинг буде проводитись здебільшого використовуючи технології протоколу SNMP. Алгоритм роботи цього протоколу(див. рис. 2.2) починається з системи управління мережами(NMS). NMS ініціює запити до агентів SNMP для отримання інформації про стан пристроїв. Агенти це програмне забезпечення,

яке відповідає за моніторинг і збір інформації з кожного окремого мережевого обладнання і вбудовані в систему [18].

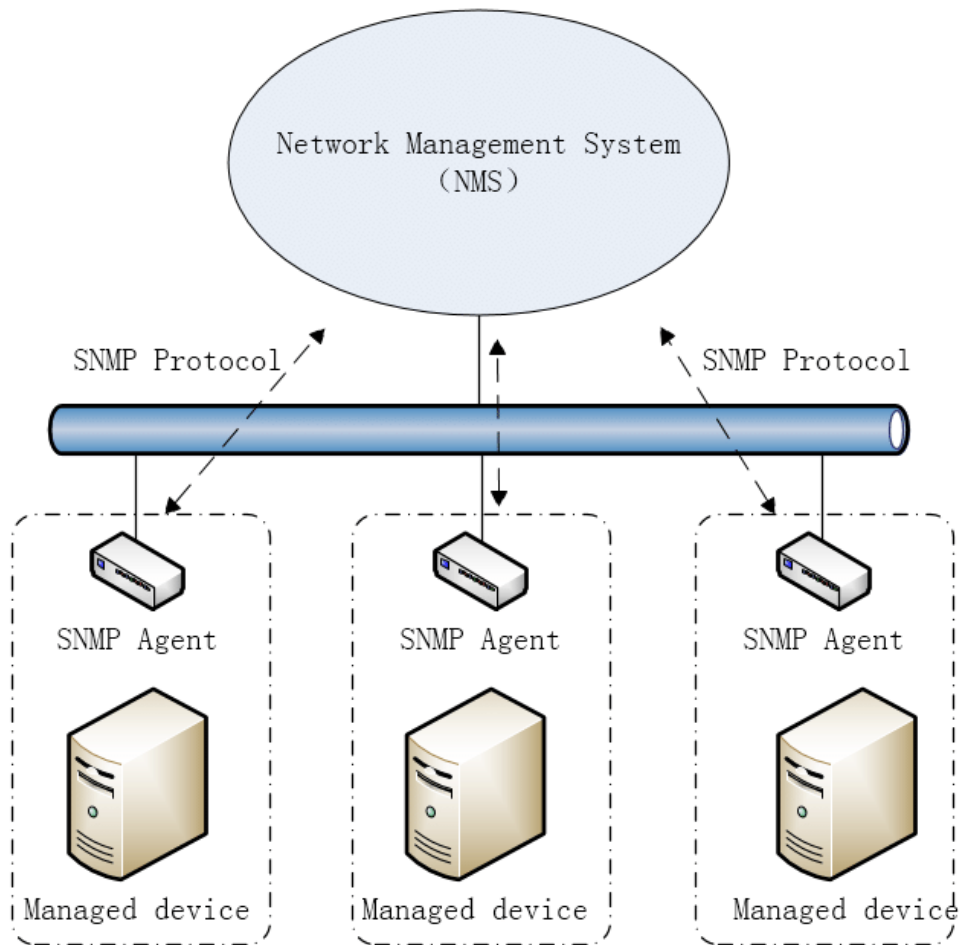


Рис. 2.2 Блок-схема роботи алгоритму протоколу SNMP

SNMP Manager може періодично опитувати SNMP агентів для визначення їх стану. Але в агентів є можливість встановлення "пасток" (trap), що дозволяє агентам сповіщати менеджера про події без заздалегідь визначених запитань. Таким чином це автоматизує систему і дозволяє їй ефективно працювати пре непередбачуваних або неочікуваних обставинах. Операції SNMP використовують методи GET, SET і GETNEXT для доступу інформації або зміни конфігурації. Систему керування можна спрощено визначати, як клієнт, а агентів SNMP, як сервер, тому що перший ініціює взаємодію, а другий виконує відповідні операції. До кожної мережі, що проходить моніторинг повинен бути

підключений блок power control, що перевіряє наявність з'єднання пристроїв з мережею і живлення для них.

Другим процесом в алгоритмі роботи після налаштування роботи системи з SNMP протоколом має стати надання найкращих можливостей операторам мереж слідкувати за їх апаратної і програмною справністю. Для цього програма The Dude використовує свою функцію журналювання(логінгу). Цей процес експорту даних про стан мереж можна використовувати для налаштування сповіщень про несправності для працівників. Враховуючи, що The Dude це одна з ланок системи аналізу ефективності постачання мережевих послуг, то крім функції про сповіщення логінг має постачати важливі дані зібрані з мереж для програмного забезпечення, яка забезпечить математичне обрахування елементів. Для The Dude існує багато готових рішень з автоматичного експорту даних, одною з яких є алгоритм PHP-MySQL-PHP-Месенджер. Тобто спочатку дані експортуються в найпопулярнішу систему керування реляційними базами даних, а з неї для швидшого донесення важливої інформації до операторів мережі має потрапляти у якийсь з месенджерів шляхом надсилання повідомлень через чат-бот. Через необхідність наявності у месенджера розвинутої інфраструктури чат-ботів і готових рішень у цій сфері. Інші рішення в комунікації серед працівників інтернет-провайдера використовують месенджер Telegram. Оскільки він має найбільшу кількість готових рішень з чат-ботів, а також доповнення для MERN, наприклад tgbot-expanion, то прийнято рішення використовувати його для експорту даних. Telegram має проблеми з захистом даних і їх приватністю. Хоча дані експортовані з The Dude не є найбільшим пріоритетом безпеки, а всі дані зберігаються окремо в базах даних і не пропадуть у випадку надзвичайних ситуацій, але код має бути написаний таким чином, щоб у разі проблем Telegram швидко перенести метод комунікації в такі месенджери, як Facebook Messenger або Whatsapp. Оскільки лістинг коду для чат-боту доволі простий, то ввівши API від іншого месенджера дозволить швидко замінити програму для комунікації. Враховуючи, що одна з застосованих технологій- React розроблена компанією

МЕТА, то Facebook Messenger буде найкращою альтернативою Telegram враховуючи сумісність програмного забезпечення цієї компанії.

Оскільки рішення з експорту в The Dude здебільшого використовують релятивні бази даних, то це ускладняє процес надсилання інформації в систему аналізу, що реалізується з застосуванням релятивних баз даних MongoDB. Тому той самий чат-бот повинен стати посередником між різними системами баз даних і бути інструментом не тільки сповіщення про проблеми, але й експорту даних на наступні етапи.

Третім етапом системи є забезпечення отримання системою усіх даних і параметрів для подальшого обрахунку. MERN стек має велику кількість рішень з розробки чат-ботів. Тому в доповнення до існуючого чат бота, що надсилає дані моніторингу на Node.js має бути створений простий бот з єдиною функцією-експортом даних з чату в MongoDB базу даних. Таким чином нівелюються принципові різниці між реляційними і нереляційними базами даних і процес переходу інформації не має перешкод з конвертацією даних. Код цього чат боту має бути розміщений зі всією серверною частиною проекту використовуючи можливості технології Express. Як видно на структурній схемі MERN стеку(див. рис. 2.3), Express є базою для циркуляції даних серед системи [19].

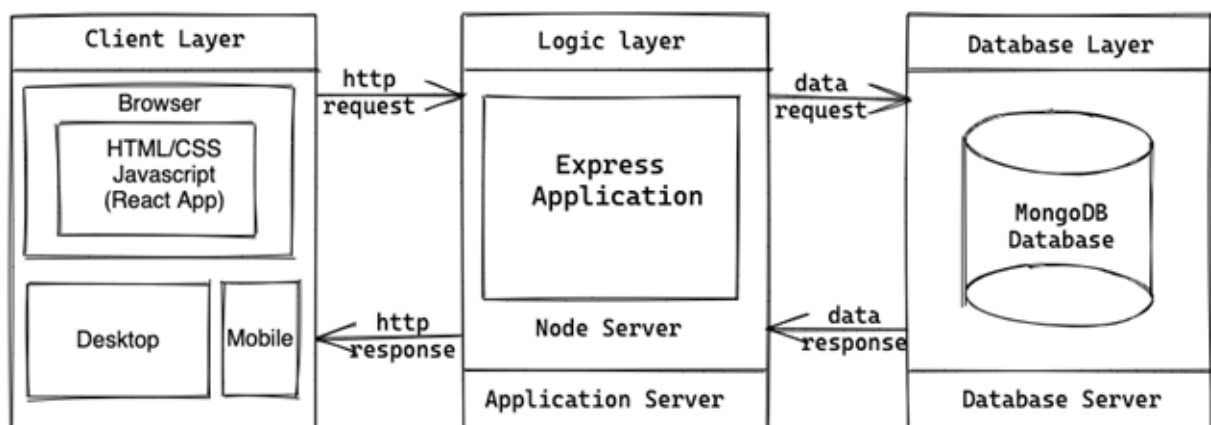


Рис. 2.3. – Структурна схема набору технологій MERN

Ця технологія налаштує також передачу даних від користувачів використовуючи форми на клієнтській частині проекту до MongoDB. Не працююча серверна частина фактично паралізує систему, тому важливо створити можливості для його безперебійної роботи. Для цього можна застосовувати спеціально розроблений модуль helmet, що складається з дрібніших функцій для створення безпечного середовища роботи сервера. Після залучення функцій csp, hidePoweredBy, hsts, ieNoOpen і інших можливостей helmet можна встановлювати серверну частину на хостинг і перейти до створення можливостей з математичного обчислення і аналізу ефективності роботі інтернет-провайдера.

Останньою складовою системи повинна стати реалізація математичного забезпечення. Вона буде реалізована з використанням всіх технологій MERN, а демонстрація даних для працівників буде завдяки React. Варто зазначити, що з використанням самого лише фреймворку React реалізувати клієнтську частину було б доволі складно, тому ця технологія повинна постійно модернізуватись [13]. Найважливішими доповненням до React в цій роботі є react-dom і react-router-dom [20]. Без першого було б неможливо взаємодія з подіями та оновлення вмісту сторінки без повного, а без другого неможливо організувати структуру і налаштувати перехід на різні елементи веб-ресурсу. Використовуючи технології описані раніше повинна бути створена прихована сторінка, призначена для розрахування деяких математичних показників, отриманих попередньо системою з системи моніторингу і форм веб-ресурсу і перенесених в базу даних.

Оскільки такі розрахунки не мають бути доступні пересічному користувачу, то перед отриманням даних працівник повинен пройти авторизацію. У разі успішної авторизації на сторінку виводяться математичні розрахунки. Проте навіть для одного простого розрахунку, наприклад за допомогою формули 2.2, програма повинна перевіряти багато факторів і виконувати різні алгоритмічні кроки, як можна описати блок схемою(див. рис. 2.4). Для початку програма повинна виконати з'єднання з сервером і можливість

приєднання серверу до бази даних. При виконанні цих умов можна здійснити авторизацію користувача, при успішному виконанні авторизації програма починає проводити обчислення. Вона виводить інформацію з бази даних і перевіряє чи була розміщена інформація для обрахунку. У випадку відсутності даних про зупинку роботи системи, програмне забезпечення виводить інформацію про нульовий простій. Якщо ж записи були знайдені то програма спочатку вираховує час кожного простою шляхом віднімання часу вимкнення від часу ввімкнення мережі. Після цього знайдені результати додаються. Далі вираховується скільки сповіщень про відсутність живлення було в базі даних і ділячи загальний час простою на кількість зупинок мереж отримується і виводиться показник середнього простою одного з регіональних відділень компанії, який можна порівняти з показниками інших відділень.

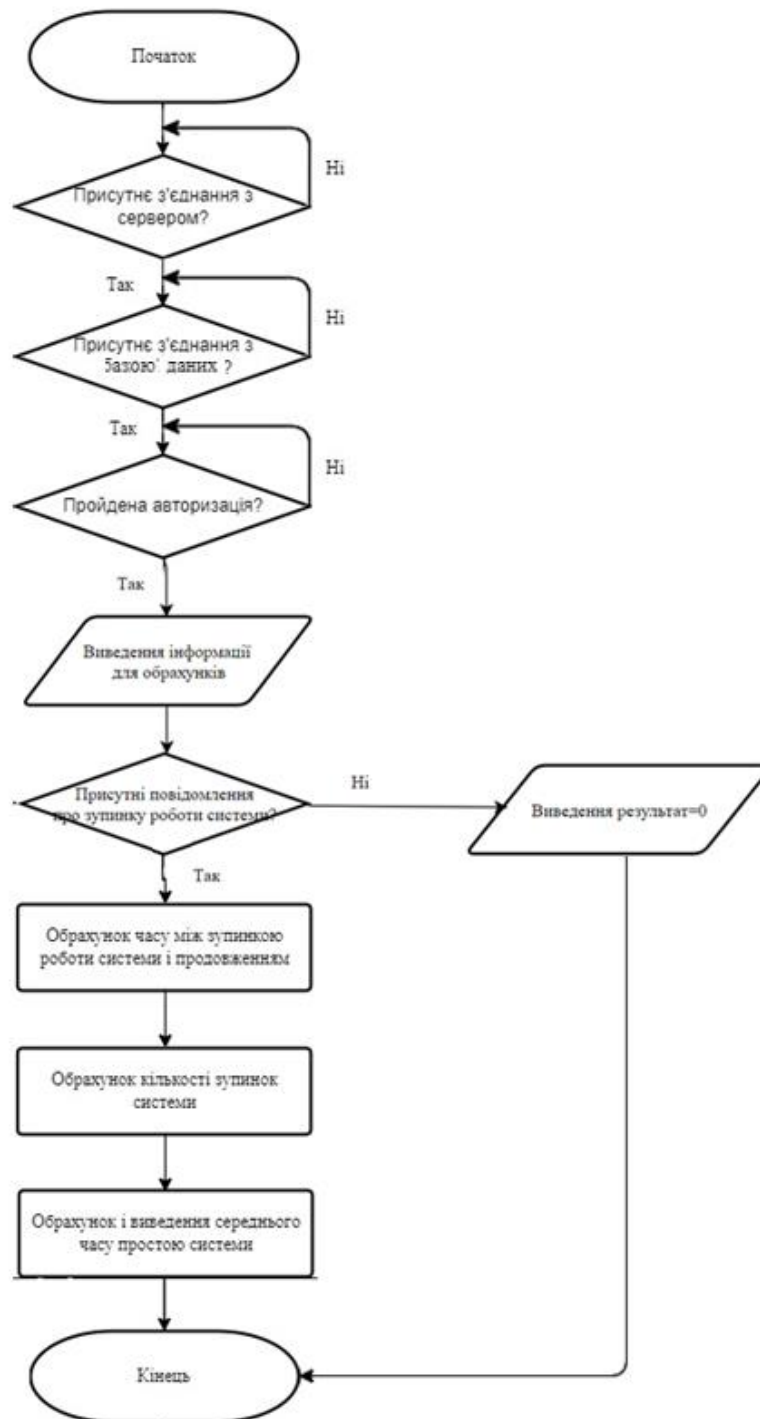


Рис. 2.4. Блок-схема алгоритму математичного обчислення одного з параметрів мережі

Після проектування того, за якими принципами повинна працювати система, можна розробити алгоритм її роботи(див. рис. 2.5), що наглядно продемонструє зв'язаність процесів і порядок роботи.

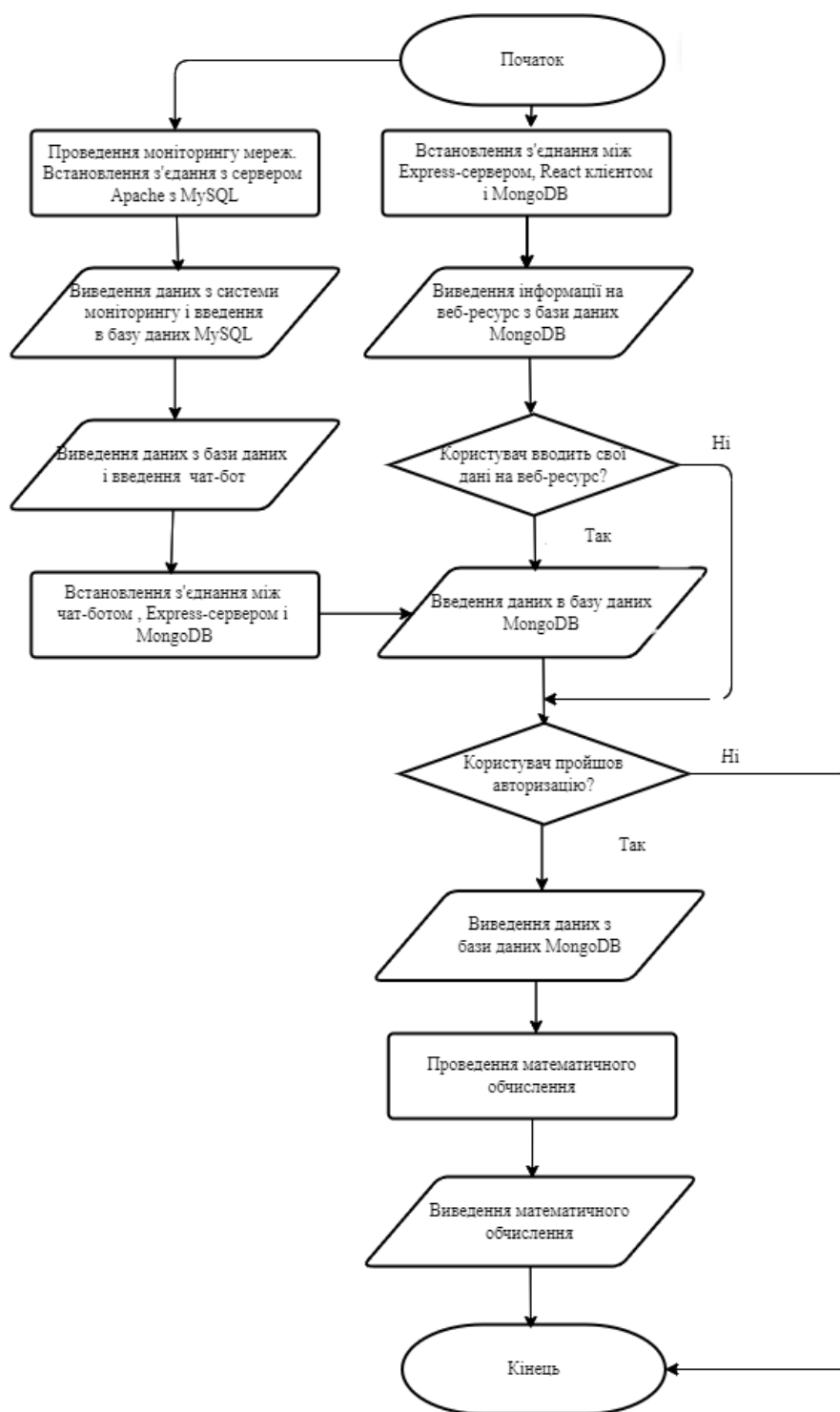


Рис. 2.5. Блок-схема алгоритму роботи системи аналізу і моніторингу ефективності роботи інтернет-провайдера

На початку роботи системи два процеси працюють паралельно-моніторинг мережі і робота веб-ресурсу. У випадку встановлення всіх з'єднань веб-ресурс виводить контент для користувачів. Далі клієнт може вказати свої дані використовуючи форму для відправки подачі заявок і скарг, тоді база даних

MongoDB повинна оновитись. Якщо в паралельному процесі всі з'єднання відбулись, то у разі відправки в чат-бот повідомлень з бази даних і серверу, що транслюють дані програми моніторингу, чат-бот оновлює базу даних MongoDB записами своїх повідомлень. Якщо працівник хоче отримати доступ до математичних обрахунків отриманих даних, то ці процеси будуть відображені після успішного завершення авторизації. Тільки після цього почнеться процес математичного обчислення і виведення результатів. В рамках одного обчислення програма повинна виконати декілька перевірок і функцій. Таким чином в результаті роботи програмного забезпечення інтернет-провайдер повинен здобути комплексні можливості для отримання даних і їх аналізу в рамках однієї системи.

2.3. Висновки

В другому розділі було розглянуто математичні методи розрахунку ефективності різних показників в рамках надання мережевих послуг інтернет-провайдером. Були розглянуті такі способи обчислення ефективності мереж як: пропускна здатність, відсоток втрати пакетів, час простою, затримка, ефективність мережі за часом роботи, ефективність мережі за завантаженістю та інші. Розроблено математичну модель обчислення через який проміжок часу приріст клієнтів змусить збільшувати пропускні спроможності мережі. Наведено математичні обрахунки для виявлення ефективності обслуговування системи і співпраці з клієнтами. Наведено способи для застосування методу черг.

Розглянуто способи виконання поставлених задач і принципи роботи протоколу моніторингу SNMP. Спроектовано порядок розробки системи, розглянуто принципи і особливості технології MERN. Запропоновано засоби модернізації React використовуючи доповнення, а також Express за допомогою модуля helmet. Запропоновано можливості для переносу чат-боту в Telegram на платформи типу Messenger. Описано алгоритми виконання завдань системи і продемонстровано за допомогою блок-схем.

РОЗДІЛ 3

ЗАСТОСУВАННЯ ОБРАНИХ МЕТОДІВ РОЗРОБКИ ДЛЯ СТВОРЕННЯ СИСТЕМИ МОНІТОРИНГУ І АНАЛІЗУ МЕРЕЖЕВИХ ПОСЛУГ

3.1. Реалізація збору даних з мережевого моніторингу

Для того щоб налаштувати моніторинг телекомунікаційних мереж після встановлення програми MikroTik The Dude потрібно створити новий проект і підключати туди необхідні мережі. В головному вікні The Dude потрібно натиснути на "Add Network" вказати назву мережі, а потім параметри, такі як IP-адресу, підмережу та інші налаштування. Для забезпечення роботи протоколу потрібно встановити параметр "Use SNMP".

Щоб провести первинне сканування мережі потрібно в контекстному меню для мережевого інтерфейсу вибрати "Scan". Після завершення сканування пристрої будуть відображені у вікні "Devices", щоб додати їх до моніторингу використовується "Add to map", після цього можна буде переглянути мапу мережі(див. рис. 3.1). У розділі "Links" можна налаштувати з'єднання між пристроями, щоб визначити, як вони пов'язані між собою. Також потрібно вибрати сервіси, для даного проекту використовуються як Ping так і SNMP[6].

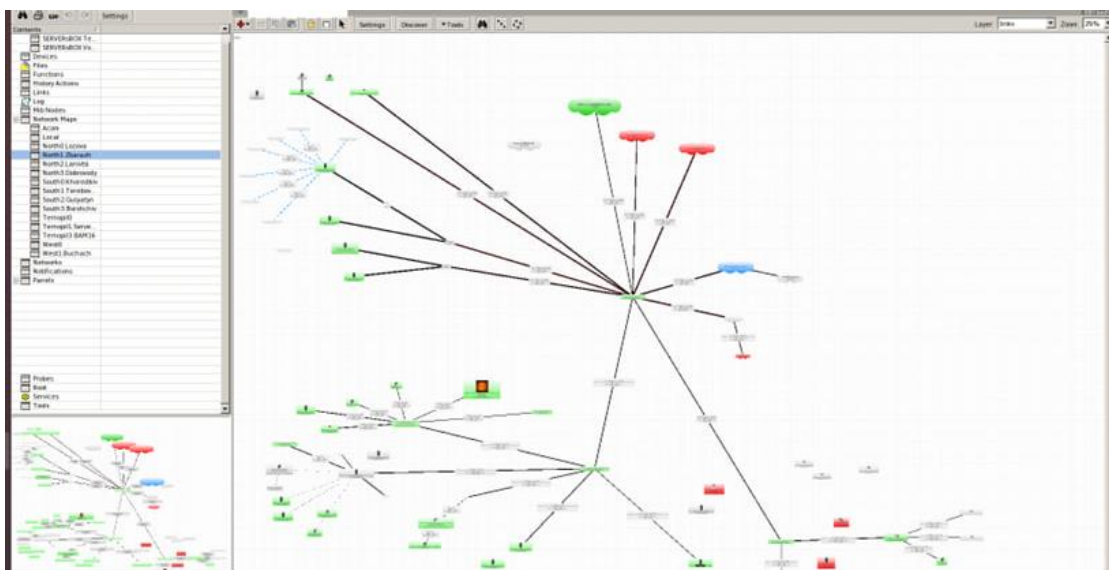


Рис. 3.1. Налаштований і підключений вузол мережі

Щоб реалізувати функцію логінгу в базу даних Mysql потрібно обрати "Logging" у налаштуваннях. Після обрання опції "Log to syslog server" потрібно вказати адресу порту на який відправляє. Серед готових рішень для логінгу використовується корпоративний Apache сервер. Для того щоб приймати логи на базу даних використовується написаний php код, типовий для такої задачі(див. рис. 3.2) .

```
2 $rawLogData = file_get_contents("php://input");
3 $username = "acom";
4 $servername = " ";
5 $password = " ";
6 $dbname = " ";
7
8 $conn = new mysqli($servername, $username, $password, $dbname);
9
10 if ($conn->connect_error) {
11     die("Connection failed: " . $conn->connect_error);
12 }
13
14 $sql = "INSERT INTO mikrotik_logs (log_data) VALUES ('$rawLogData')";
15
16 if ($conn->query($sql) === TRUE) {
17     echo "Log data added to MySQL successfully";
18 } else {
19     echo "Error: " . $sql . "<br>" . $conn->error;
20 }
21
22 $conn->close();
```

Рис. 3.2. Лістинг php-коду для логування

Далі потрібно створити telegram-бота і отримати унікальний API-ключ(див. рис. 3.3). Використовується PHP-скрипт (див. рис. 3.4), який використовує бібліотеку для взаємодії з API Telegram.

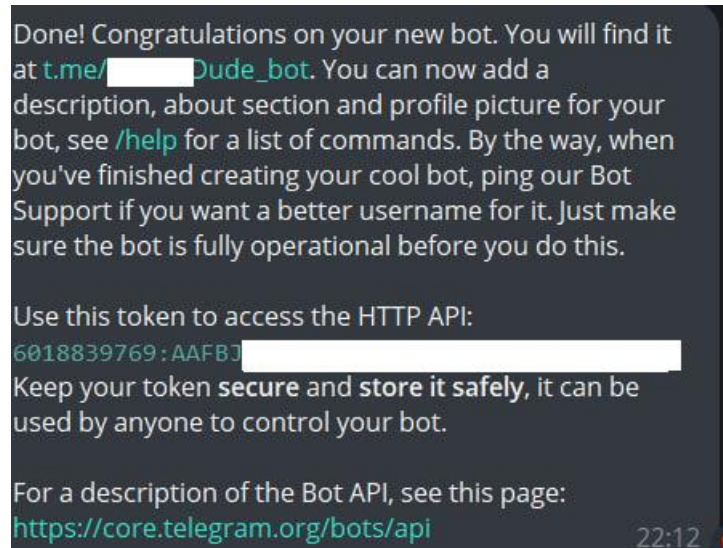


Рис. 3.3. Отримання даних чат-бота

```

1  <?php
2
3  $telegramBotToken = "your_telegram_bot_token";
4  $chatId = "your_chat_id";
5
6  $message = "Time: " . date("M/d/Y H:i:s") . "; dev
   .ap4|eth5; Service: ping; Status: up";
7
8  // Відправка повідомлення в Telegram
9  $telegramApiUrl = "https://api.telegram.org/bot$te
   ?chat_id=$chatId&text=$message";
10 file_get_contents($telegramApiUrl);

```

Рис. 3.4. Лістинг php-коду для чат-бота

Цей скрипт відправляє повідомлень за допомогою чат-бота в потрібний канал або чат, якщо показники, що моніторяться програмою зазнають змін.

3.2. Програмна реалізація створення веб-ресурсу

3.2.1 Створення клієнтської частини веб-ресурсу. Коли йдеться про ефективність роботи інтернет-провайдера, то створення якісного веб-ресурсу дозволить не тільки збирати необхідні дані для їх аналітики, але й необхідний запис для підвищення ефективності компанії. Оскільки веб-сайт допомагає привертати увагу потенційних клієнтів, налагоджувати комунікацію з користувачами і популяризувати продукцію провайдера, важливо дотримуватись правил веб-розробки і SEO для того, щоб клієнти могли знайти і користуватись веб-ресурсом. SEO, як процес що спрямований на покращення видимості сайту у пошукових системах вимагає впровадження певних вимог, які повинні дотримуватись під час розробки.

Розробляючи верстку веб-ресурсу потрібно спочатку визначитись з структурою, для цього потрібно проаналізувати рішення інших інтернет-провайдерів і створити свою структуру. На веб-ресурсі створено 6 окремих сторінок доступних для перегляду кожного користувача— одну початкову з основою інформацією й інші, такі як: «Акції», «Новини», «Тарифи», «Підтримка» і «Контакти», а також одну сторінку, що потребує авторизації і відображає реалізацію математичних обрахунків для аналізу даних.

Під час розробки потрібно розбити структуру сайту на логічні блоки, такі як головне меню(header) заголовки, секції для контенту та підвал/footer). Під час цього етапу робиться акцент на семантиці, обираючи теги, які краще відповідають типу інформації. Залежно від вибраного HTML-тегу пошукові сервіси роблять висновки про тип контенту, який розміщений в цих тегах. Тому в цьому проекті для заголовні використовуються виключно теги h1-h6, при цьому тег h1 не повинен повторюватись на сторінці. Основний контент при розробці розміщений в блоках, таких як main, section, header. Вони зможуть демонструвати алгоритмам пошукових систем, що веб-ресурс має дієву структуру і збільшувати відвідування сайту.

Після завершення розробки верстки потрібно за допомогою HTML, потрібно перейти до налаштування стилів за допомогою CSS. Каскадні таблиці стилів важливі для забезпечення зручності і вигляду інтерфейсу, а обрання різних стратегій стилізації впливає також на технологічність і швидкодію веб-ресурсу. Тому, після проведення аналізу різних технологій CSS, прийнято рішення відкинути застарілі рішення типу float, а обирати стилі розміщення інформації серед Flexbox і Grid. Для даного веб-ресурсу не настільки важливі функції, що надає Grid для позиціонування складних елементів, тому можливості Flexbox стали достатніми для його застосування в стилях проекту на протязі всієї розробки(див. рис. 3.5).

```
react-backend > client > src > # index.css > ...
153 }
154   .login-icon:active .hide-menu{
155     display: block;
156   }
157 }
158 li{
159   list-style-type: none;
160 }
161
162 .title-center{
163   display: flex;
164   justify-content: center;
165 }
166
167 .intro{
168   min-height: 100vh;
169   height: 100%;
170 }
171
172 .intro-first-block{
173   width: 100%;
174   display: -webkit-box;
175   display: -ms-flexbox;
176   display: flex;
177   -ms-flex-pack: distribute;
178   justify-content: space-around;
179 }
180
181 .intro-second-block{
182   width: 100%;
183   display: -webkit-box;
184   display: -ms-flexbox;
185   display: flex;
186   -ms-flex-pack: distribute;
187   justify-content: space-around;
188 }
189 }
```

Рис. 3.5. Лістинг CSS-коду

Важливою частиною цього етапу є робота з різними розмірами екранів, враховуючи адаптивність та мобільність. Оскільки після 2020-го року більшість користувачів користуються веб-ресурсами через мобільні пристрої(див. рис. 3.6), то важливо реалізувати оптимізацію сайту під різні розміри екрану [21]. Також важливим фактором для введення адаптивності у веб-ресурс є кросс-

браузерність. Для того щоб інтерфейс відображався однаково для всіх комп'ютерних браузерів, потрібно використати веб-префікси.

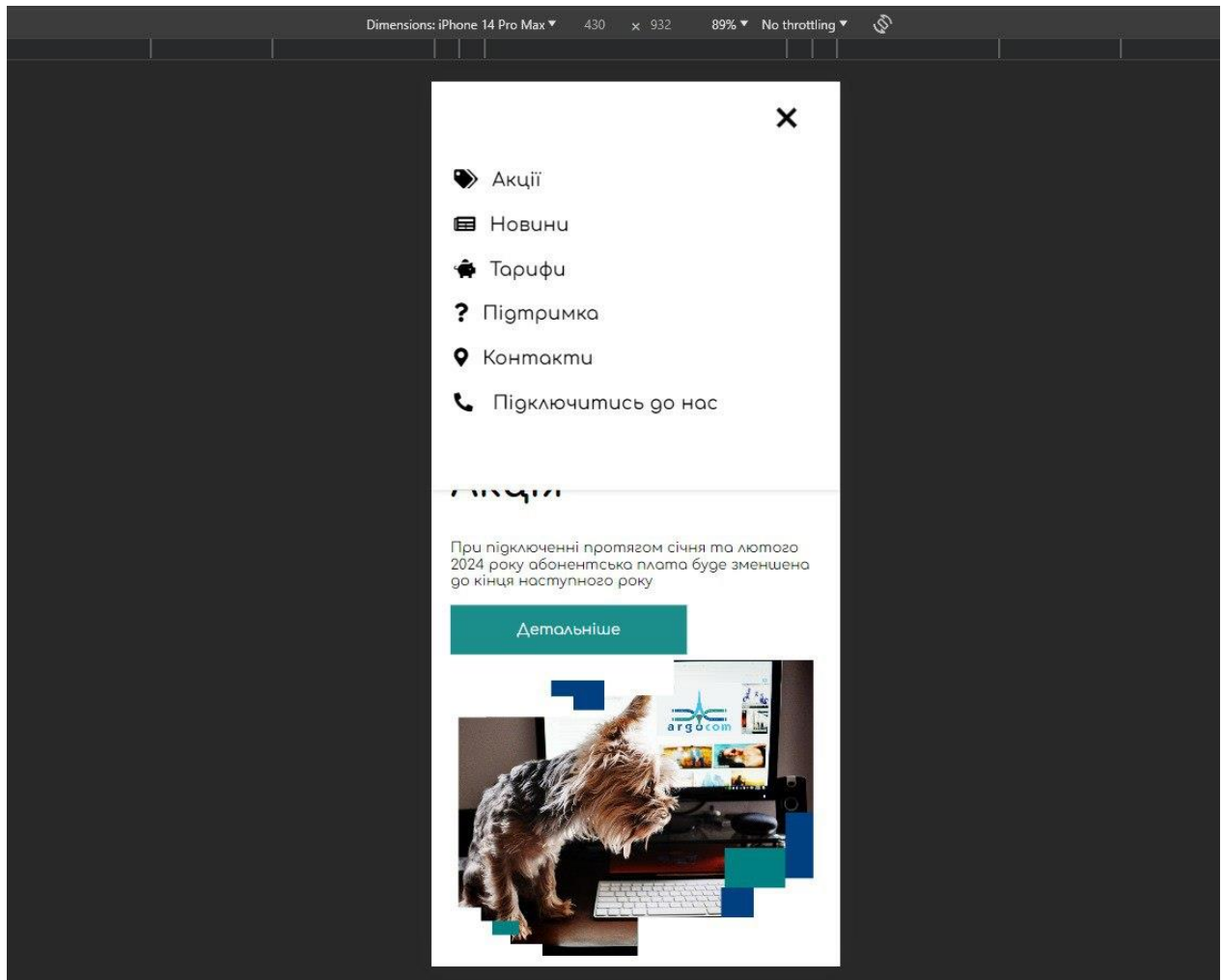


Рис. 3.6. Процес перевірки адаптивності веб-сайту

Дотримавшись всіх правил і розробивши верстку для клієнтської частини веб-ресурсу потрібно перейти до додання логіки, інтерактивності і динамічності у проект. Для цього у веб-розробці використовується основна мова програмування JavaScript. Оскільки веб-ресурс реалізовується із використанням технології MERN, то перехід до розробки за допомогою JavaScript потрібно шляхом створення й інтеграції React-проекту .

Для того, щоб автоматизувати створення веб-сторінок за допомогою React можна використовувати технологію Create-React-App. Цей інструмент пропонує простий та швидкий шлях створення нового проекту, позбавляючи

розробника від необхідності ручної настройки великої кількості конфігураційних параметрів. Відкривши термінал в середовищі розробки, потрібно ввести в термінал команду `npx create-react-app`(див. рис. 3.7). Після цього відбувається автоматичне створення проекту з мінімальною структурою файлів та підтримкою сучасних доповнень, таких як Babel, Webpack, та ESLint.

```
PS C:\Users\User\Desktop\диплом\react> npx create-react-app my-app
Need to install the following packages:
  create-react-app
Ok to proceed? (y) y
npm WARN deprecated tar@2.2.2: This version of tar is no longer supported, an
Creating a new React app in C:\Users\User\Desktop\диплом\react\my-app.

Installing packages. This might take a couple of minutes.
Installing react, react-dom, and react-scripts with cra-template...

added 1463 packages in 2m

242 packages are looking for funding
  run `npm fund` for details

Initialized a git repository.

Installing template dependencies using npm...
```

Рис. 3.7. Команда для створення React- проекту

Перед інтеграцією розробленою верстки налаштовується структура React-проекту. У створеній папці “public” створенні статичні файли, такі як `index.html`. Цей файл служить точкою входу для веб-додатка. Ще одна папка “src” містить код React-додатка, включаючи головні файли `index.js` і `App.js` [23].

Верстку клієнтської частини сайту доволі легко імпортувати в React-проект, оскільки синтаксис JSX має HTML-подібну структуру. Під час процесу інтеграції HTML-коду потрібно змінювати частину коду, оскільки без певних змін веб-ресурс буде вказувати на помилки. Тому потрібно записувати

властивості HTML в JSX з використанням camelCase, JavaScript-код повинен бути оформлений у фігурних дужках. Також атрибут класів повинен мати назву className. Після адаптації HTML-коду (див. рис. 3.8) помилки повинні зникнути і React повинен працювати коректно. Далі верстка веб-ресурсу розділяється на компоненти React. Кожна сторінка на сайті повинна отримати свій файл з розширенням .js і за допомогою доповнення react-router-dom потрібно зв'язати кожен компонент з головною сторінкою веб-ресурсу імпортовано в файл App.js.

```

    <h2>Чому ми?</h2>
  </article>
  <div className="reasons">
    <div className="reason-row">
      <div className="first-reason">
        <div className="circle">
          <i className="fas fa-server big-icon" id="server"></i>
        </div>
        <article className="text">
          <h3>Оперативність</h3>
          <p>Шукаєте провайдера, який буде швидко вирішувати ваші проблеми?> нами Вам не потрібно чекати, нас є предста
        </article>
      </div>
      <div className="second-reason">
        <div className="">
          <i className="fas fa-user-alt big-icon" id="dish"></i>
          <div className="triangle-right"></div>
        </div>
        <article className="text">
          <h3>Розуміння</h3>
          <p>Argosom - одна з тих небагатьох компаній, які завжди готові ввійти в Вашу ситуацію, підлаштуватися
          та подати Вам руку допомоги.<br/> нас індивідуальний підхід до всіх, > ми цінуємо кожного клієнта</p>
        </article>
      </div>
      <div className="reason-row">
        <div className="third-reason">
          <i className="fas fa-satellite-dish big-icon" id="dish"></i>
          <div className="triangle"></div>
          <article className="text">
            <h3>Розвиток</h3>
            <p>Наша компанія постійно ровиває наше обладнання щоб надавати Вам високошвидкісний інтернет за низьку цінуем
          </p>
          </article>
        </div>
        <div className="fourth-reason">

```

Рис. 3.8. Лістинг JSX коду адаптованого з HTML

Після налаштування веб-ресурсу на React можна створити рішення, що позитивно вплинуть на дизайн інтерфейсу, такі як темний режим сайту чи модальні вікна використовуючи модуль styled-components. Але взаємодія з клієнтом і запис його даних у базу даних буде не можливий без серверної частини.

3.2.2. Створення серверної частини веб-ресурсу. Express є центральним елементом, як MERN стеку, так і розробленої системи загалом. Без Express, як програмної оболонки для серверного фреймворку Node.js неможливо налаштувати взаємодію Клієнтської частини з системами керування базами даних і іншими зовнішніми об'єктами. Аналогом create-react-app для Express є Express-Generator. Він представляє собою інструмент для автоматизованого створення шаблонів проектів на базі Express.js. Express-Generator спрощує початковий процес створення базового каркасу серверної частини проекту, включаючи каталоги та файли, які забезпечують структуру веб-додатка. Після його активації у терміналі командою `npm run express-generator` (див. рис. 3.10), інструмент генерує конфігураційні файли, такі як `package.json` та файли налаштувань сервера, зменшуючи необхідність вручну налаштовувати всі параметри[22].

```
PS C:\Users\User\Desktop\диплом\react> npx express-generator
Need to install the following packages:
  express-generator
Ok to proceed? (y) y
npm WARN deprecated mkdirp@0.5.1: Legacy versions of mkdirp are no longer
warning: the default view engine will not be jade in future releases
warning: use '--view=jade' or '--help' for additional options

create : public\
create : public\javascripts\
create : public\images\
create : public\stylesheets\
create : public\stylesheets\style.css
create : routes\
```

Рис. 3.10. Команда для встановлення Express

Ініціалізуючи проект створюється новий каталог, що містить базовий шаблон Express-додатка. У файлі із назвою `bin` налаштовуються команди запуску серверу. У папку `public` потрібно помістити попередньо розроблену клієнтську

частину на React. А у папці routes потрібно встановити і налаштувати файли маршрутизації для різних частин додатка. Перед початком написання коду встановлюється модуль helmet(див. рис. 3.11) для покращення безпеки сервера.

```
PS C:\Users\User\Desktop\диплом\react> npm install --save helmet
npm WARN deprecated transformers@2.1.0: Deprecated, use jstransformer
npm WARN deprecated constantinople@3.0.2: Please update to at least constantinople
npm WARN deprecated jade@1.11.0: Jade has been renamed to pug, please install the
added 100 packages, and audited 101 packages in 13s
1 package is looking for funding
```

Рис. 3.11. – Команда для встановлення helmet

У файлах app.js і index.js, що є основний файлом в Express-додатку потрібно почати налаштування зв'язків у MERN проєкті. Для початку здійснюється підключення бази даних до проєкту. MongoDB дозволяє декілька методів підключення, але для даної роботи використовується підключення за допомогою підключення MongoClient з унікальним гіперпосиланням на базу даних(див. рис. 3.12).

```
getdb();
async function getdb() {
  global.connection = await MongoClient.connect("mongodb+srv://cart0n
    useUnifiedTopology: true,
    useNewUrlParser: true
  })
  global.database = await global.connection.db("acom");
  global.news = await global.database.collection('news');
  global.actions = await global.database.collection('actions');
  global.tariff = await global.database.collection('tariff');
  global.otariff = await global.database.collection('otariff');
  global.call = await global.database.collection('any');
  global.oproblem = await global.database.collection('oproblem');
  global.login = await global.database.collection('login');
  global.Zbarazherr = await global.database.collection('Zbarazherr');
  global.Zbarazh = await global.database.collection('Zbarazh');
```

Рис. 3.12. Лістинг коду для підключення до бази даних і її колекцій

Після підключення бази даних потрібно налаштувати її з'єднання з клієнтською частиною, для цього варто використати функції з методами `get` і `post`. Метод `get` (див. рис. 3.13) відповідає за виведення інформації в базу даних, після його імплементації інформацію на сторінках сайту, в тому числі для реалізації математичних обчислень, можна буде генерувати і оновлювати в базі даних, не взаємодіючи з кодом веб-ресурсу.

```
router.get('/tariff', async function (req, res) {
  let data;

  data = await global.tariff.findOne({});

  console.log(data)
  res.send(JSON.stringify(data));
});
```

Рис. 3.13. Лістинг коду для реалізації методу `get`

Метод `post` (див. рис. 3.14), навпаки, відправляє дані введені клієнтом у базу даних. Для цього на веб-ресурсі створені форми для взаємодії з клієнтом, за допомогою яких нові користувачі зможуть подавати заявки на відключення послуг, а діючі абоненти зможуть скаржитись на виникнення помилок в мережевих послугах провайдера.

```

router.post('/tariffs', function (req, res) {
  console.log(req.body)
  console.log("Тариф: " + req.body.type)
  console.log("Імя: " + req.body.name)
  console.log("Номер: " + req.body.phone)

  let data = req.body;
  global.otariff.insertOne(data, function (err, ans) {
    if (err)
      console.log(err);
    else console.log("ok")
  });
  res.end();
});

```

Рис. 3.14. Лістинг коду для реалізації методу post

3.2.3. Програмна реалізація математичного забезпечення

Можливості MERN стеку також забезпечуються для обрахунку математичних функцій. Налаштувавши з'єднання усіх технологій на веб-ресурсі потрібно створити сторінку, що буде виводити з бази даних показники, оброблювати їх і виводити на клієнтську частину веб-ресурсу. Для того, щоб налаштувати авторизацію і тільки перевірені користувачі могли переглядати математичний аналіз ефективності роботи провайдера ресурси MERN пропонують декілька способів. Один з них це підключення модуля Passport.js, це доповнення дасть багато можливостей в тому числі створення особистого кабінету для клієнтів. Але оскільки даний проект не потребує таких широких можливостей то можна справитись без додаткових модулів. Можна створити одну колекцію у базі даних з іменем користувача і паролем(див. рис. 3.15) і реалізувати просту авторизацію для адміністраторів, без необхідності реєстрації.

```
QUERY RESULTS: 1-1 OF 1

  _id: ObjectId('657afc2b154cdc0b262e6308')
 password: "admin_F2$1c)"
 username: "acom"
```

Рис. 3.15. Дані для авторизації записані у колекції MongoDB

На фронтенд-частині створюється форма в яку можна відправляти дані для авторизації. Після витягнення даних з бази даних в серверній частині веб-ресурсу пишеться запит до серверу на клієнтській частині з метою перевірки авторизаційних даних вказаних користувачем і записаних у базі даних. Якщо дані співпадають, то на клієнтській частині виводяться математичні обрахунки. Обрахунки ці попередньо вказуються в коді, як клієнтської, так і серверної частини. Для математичних обрахунків можна підключити доповнення MERN стеку Math.js[25],але основна складність в реалізації програмного забезпечення є не проблема з обрахунками, а виведення правильних даних з бази MongoDB. Тому можна використовувати вбудовані математичні функції, але приділяти увагу умовам обрахунку у коді(див. рис. 3.16). У випадку потреби ці обрахунки можна записати у базу даних.


```

const AverageTimeComponent = () => {
  const [averageTime, setAverageTime] = useState(null);

  useEffect(() => {
    fetch('/averageTime')
      .then((response) => response.json())
      .then((data) => {
        setAverageTime(data.averageTime);
      })
      .catch((error) => {
        console.error('Error fetching average time:', error);
      });
  }, []);

  return (
    <div>
      <h2>Average Time:</h2>
      {averageTime !== null ? (
        <p>`Tc= ${averageTime} `</p>
      ) : (
        <p>Loading...</p>
      )}
    </div>
  );
};

```

Рис. 3.16. Лістинг коду математичного обчислення параметра перед виведенням в клієнтську частину

Для виведення математичних графіків що дозволять наглядно продемонструвати зміни в параметрах ефективності роботи інтернет-провайдера можна використовувати доповнення Recharts. Воно надає можливість використовувати багато додаткових модулів що генерують динамічні і гнучкі графіки та діаграми.

3.2.4. Публікація ресурсу на хостинг. Після завершення роботи з розробкою програмного забезпечення потрібно експортувати його на глобальний хостинг. Веб-ресурси розроблені за допомогою MERN технологій мають обмежені ресурси для хостингу. Для бекенд-частини додатку можна використати сервіс Render. Він безкоштовно публікує Express проект в глобальну мережу і робить доступним запити до нього

з клієнтської частини у будь-який момент. Для того щоб встановити послуги хостингу потрібно використати можливості git. Клієнтську частину слід видалити з папок Express проекту. Після цього створюється git-репозиторій, до нього додаються файли серверної частини(див. рис. 3.17) і публічно публікуються в інтернет (див. рис. 3.18).

```
PS C:\Users\User\Desktop\диплом\backendfirsttry> git push -u origin main
info: please complete authentication in your browser...
Enumerating objects: 1847, done.
Counting objects: 100% (1847/1847), done.
Delta compression using up to 4 threads
Compressing objects: 100% (1683/1683), done.
Writing objects: 100% (1847/1847), 2.38 MiB | 1.46 MiB/s, done.
```

Рис. 3.17. Експорт серверної частини за допомогою git

```
Dec 14 07:01:11 PM (Use `node --trace-deprecation ...` to show where the warning was created)
Dec 14 07:01:16 PM Your service is live
```

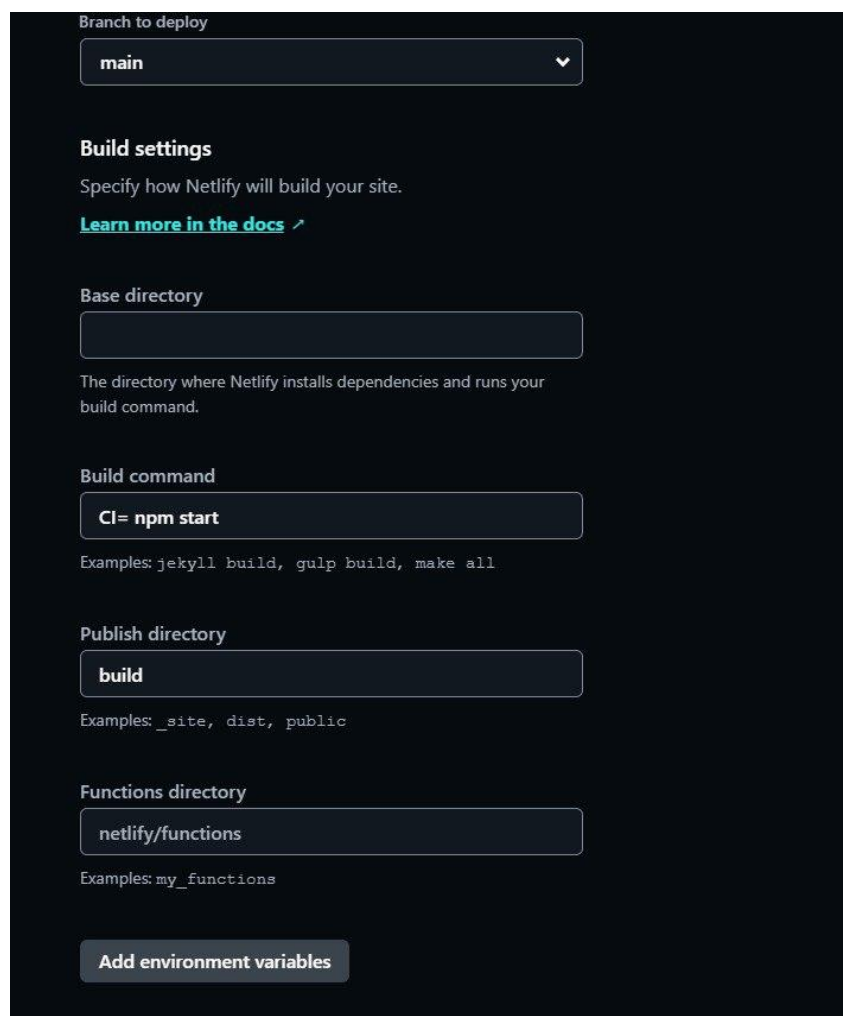
Рис. 3.18. Публікація серверної частини веб-ресурсу

Після публікації бекенд-частини веб-ресурсу потрібно у файлі “package.json” клієнтської частини у рядку “proxy” прописати наявне гіперпосилання на сервер, замість посилання на локальну мережу (див. рис. 3.19).

```
Debug
"scripts": {
  "start": "react-scripts start",
  "build": "react-scripts build",
  "test": "react-scripts test",
  "eject": "react-scripts eject"
},
"proxy": "https://backend2-rbf6.onrender.com",
"eslintConfig": {
  "extends": "react-app"
},
```

Рис. 3.19. Введення глобальної адреси сервера у далі клієнтської частини

Експорт клієнтської частини веб-ресурсу є складнішим процесом, оскільки виникають проблеми з наявністю безкоштовного хостингу, а можливості сервісу Render не призначені для фронтенд-частини. Можна використовувати сервіс Netlify, він хоч і має обрізані функції, але є надійним безкоштовним постачальником послуг хостингу для React проектів. Спосіб його публікації є таким самим, як з публікацією Express-проекту, тобто з використанням git. Для побудови проекту всередині хостингу при публікації потрібно вказати команду “npm run build”(див. рис. 3.20).



The image shows a dark-themed configuration interface for Netlify. At the top, there is a dropdown menu labeled "Branch to deploy" with "main" selected. Below this is a section titled "Build settings" with the subtitle "Specify how Netlify will build your site." and a link "Learn more in the docs". There are four input fields: "Base directory" (empty), "Build command" (containing "CI= npm start" with examples "jekyll build, gulp build, make all" below it), "Publish directory" (containing "build" with examples "_site, dist, public" below it), and "Functions directory" (containing "netlify/functions" with examples "my_functions" below it). At the bottom, there is a button labeled "Add environment variables".

Рис. 3.20. Публікація клієнтської частини веб-ресурсу

Після цього слідуючи за рекомендаціями сервісу Netlify можна надати додаткові параметри веб-ресурсу і дозволити опублікувати його.

3.3. Результати програмної реалізації системи моніторингу і аналізу

Для того щоб перевірити роботу систему варто почати з перевірки виведення інформації про моніторинг мереж з чат-боту розміщеного у месенджері Telegram (див. рис. 3.21).



Рис. 3.21. Виведення результатів моніторингу

Далі потрібно перевірити чи експортуються ці дані у MongoDB за допомогою серверу Express (див рис. 3.22).



Рис. 3.22. Запис результатів моніторингу в базі даних

Після цього варто перевірити чи коректно працює зв'язок бази даних із клієнтською частиною веб-ресурсу. Для цього спочатку можна перевірити метод `get`, тобто вивід даних на веб-сайт. Одна з сторінок “Акції” виводить для клієнтів інформацію розміщену на MongoDB, тому перейшовши на цю сторінку перевіряю наявність цієї інформації(див. рис. 3.23) і звіряю з інформацією з бази даних (див. рис. 3.24).

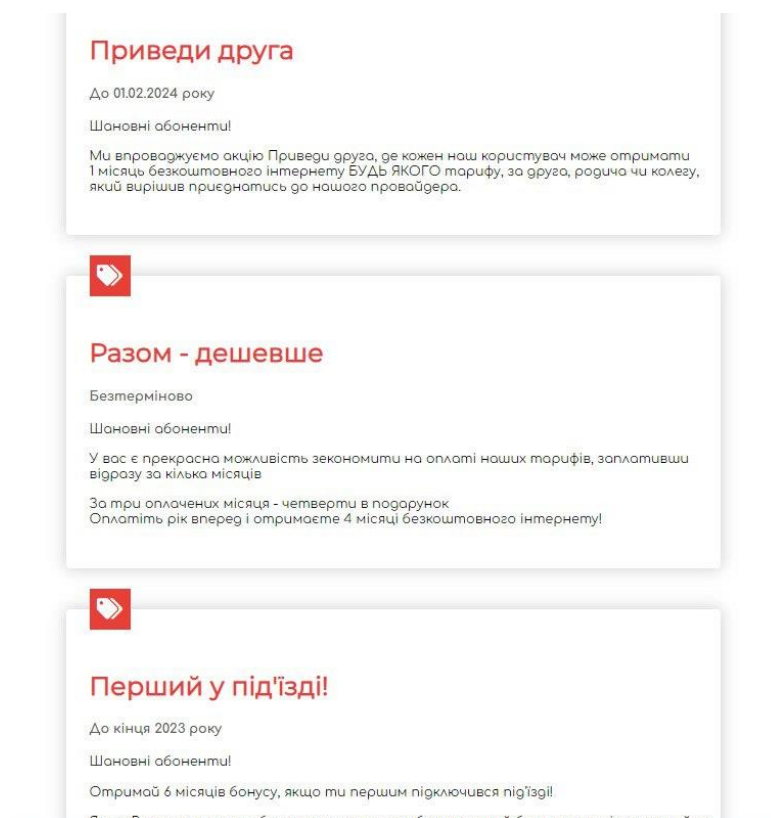


Рис. 3.23. Вивід інформації з бази даних

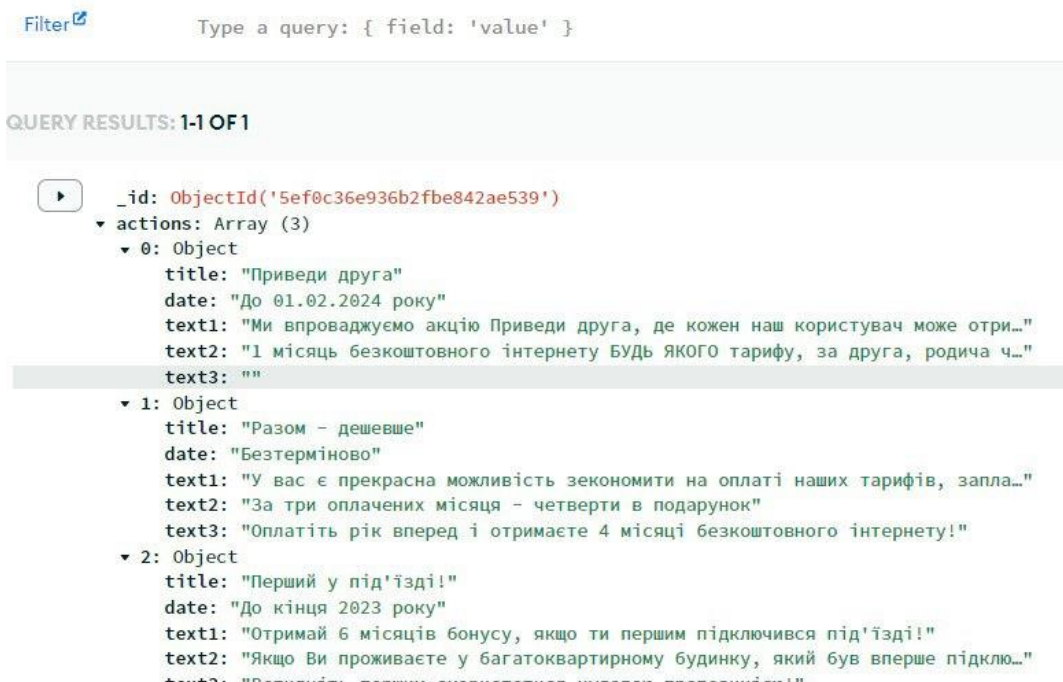


Рис. 3.24. Запис цієї інформації в базі даних

Після успішної перевірки методу `get`, варто перевірити метод `post`, це можна зробивши ввівши інформацію в форму залишення заявок на підключення, користуючись модальним вікном(див. рис. 3.25), якщо інформація надійшла в базу даних(див. рис. 3.26), то метод `post` працює коректно.

Хочете підключити наші послуги?
Залиште свої данні і наш оператор зв'яжеться з Вами

Хоростків
Юрій
0983758620
Пігтвердити

Рис. 3.25. Введення даних у форму на веб-сайті

```

_id: ObjectId('657d7676c9436c002b5cc0b3')
type: "Хоростків"
name: "Юрій"
phone: "0983758620"

```

Рис. 3.26. Запис отриманих даних у базу даних

Неперевіреною функцією веб-ресурсу залишилась авторизація, тому потрібно перейшовши на приховану сторінку “/login” ввели дані користувача записані в базу даних. Коли методи post і get працюють коректно, то сторінка має вивести таблицю з інформацією про математичні обрахунки і параметри (див. рис. 3.27), а також графік, що надає змогу відображати динаміку змін певних параметрів, що обчислюються на веб-ресурсі.

Показник	Збараж	Хоростків
Показник втрати пакетів(PLR)	0	0
Середня затримка	1.55	1.87
Вігтик потенційних клієнтів(K)	0.67	1.11
Скарги клієнтів до реальних проблем(Kгр)	0.16	0.15
Ефективність використання ресурсів(Et)	96%	17.5%
Час простою(ATr)	2.00 хв.	63.00хв.
Час до заміни обладнання(t)	3.75 м.	1077.55 м.
Коефіцієнт завантаженості	1	1.3333333333333333
Помилки антени AP1	4	8
Помилки антени AP2	4	9
Помилки антени AP3	4	4
Помилки антени AP4	4	

Рис. 3.27. Виведення математичних розрахунків

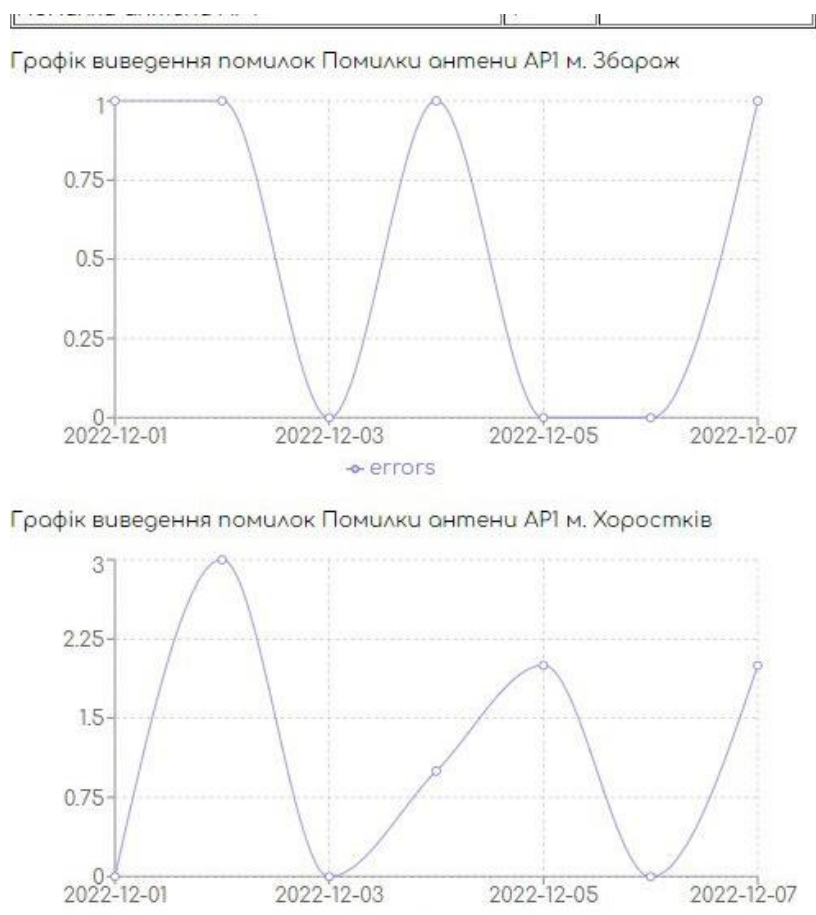


Рис. 3.28. Виведення порівняльних графіків

Якщо авторизаційні дані було введено невірно, користувачу виводиться інформацію про помилку(див. рис. 3.29).

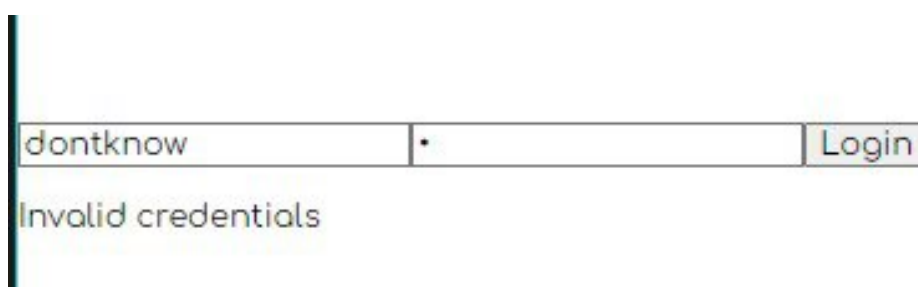


Рис. 3.29. Виведення помилки у разі невірної авторизації

Таким чином, маючи можливість порівняти дану інформацію інтернет-провайдер може порівняти ефективність різних регіональних відділень інтернет-провайдера і враховуючи кожен з параметрів роботи висновки про необхідність зміни або розвитку певних напрямків роботи.

3.4. Висновки

У даному розділі одержано наступні результати:

1. Показано способи для практичної реалізації налаштування і під'єднання телекомунікаційних мереж інтернет-провайдера до програми Mikrotik The Dude, показано можливості для підключення сервісів ping та SNMP. Реалізовано виведення логів моніторингу у базу даних а також розроблено чат бот, що оперативно сповіщає операторів системи про проблеми мережевої інфраструктури.

2. Реалізовано створення веб-ресурсу за допомогою технології MERN. Показано способи створення клієнтської частини з використанням фреймворку React і розроблено засоби для зв'язку з користувачами через форми веб-сайту. Реалізовано зв'язок клієнтської частини із серверним програмним забезпечення Express. Проведено налаштування Express і реалізовано зв'язок із базою даних MongoDB, а також зв'язок із попередньо створеним чат-ботом. Створено функцію авторизації і математичного обчислення необхідних компонентів. Показано реалізацію експорту програмного забезпечення в мережу інтернет за допомогою засобів хостингу.

3. Виконано тестування функцій системи. Наведено результати її роботи.

РОЗДІЛ 4

ОХОРОНА ПРАЦІ І БЕЗПЕКА В НАДЗВИЧАЙНИХ СИТУАЦІЯХ

4.1 Охорона праці

Темою кваліфікаційної роботи є математичне і програмне забезпечення комп'ютерних систем моніторингу і аналізу постачання мережевих послуг інтернет-провайдером. Таким чином дана тема зачіпає як галузь телекомунікаційних мереж так і роботи і розробки з програмним забезпеченням.

Під час роботи з телекомунікаційними мережами працівники повинні дотримуватись вимог ДСТУ ETSI EN 300 386:2017. Цей стандарт, що має назву “Обладнання телекомунікаційних мереж. Вимоги до електромагнітної сумісності. Гармонізований стандарт, який охоплює основні вимоги Директиви 2014/30/ЄС” зосереджений на імплементації європейських норм ETSI EN 300 386:2016. Цей європейський стандарт встановлює конкретні вимоги до електромагнітної сумісності (EMC) для обладнання телекомунікаційних мереж, щоб мережева інфраструктура могла ефективно працювати в електромагнітному оточенні, не викликаючи недопустимих електромагнітних завад для будь-якого обладнання, що знаходиться в цьому оточенні. Згідно з цим стандартом обладнання повинне бути стійким до електростатичного розряду, що може виникнути внаслідок взаємодії з людьми або іншим обладнанням. Обладнання повинне бути стійким до швидких перехідних процесів, таких як ті, що виникають під час перемикання електричних контактів і бути стійким до перепадів напруги, що можуть виникнути внаслідок змін у мережі живлення.

Також обладнання в сфері телекомунікаційних мереж регулюється за допомогою міжнародних стандартів розроблених ІТУ-Т і ІТУ-Д, що є секторами Міжнародного союзу електров'язку. ІТУ-Т відповідає за координацію стандартів для телекомунікацій та інформаційно-комунікаційних технологій. Зокрема при роботі з багатомодовим оптичним волокном потрібно застосовувати стандарти G.651.1, а для оптомодового стандарти G.652 , G.655 і G.657.

Серед інших державних стандартів України, що повинні застосовуватись при роботі з мережевим обладнанням, або з ПК при розробці програмного забезпечення є:

- ДСТУ EN 41003:2014 «Обладнання, яке підключають до телекомунікаційних мереж та/або кабельних розподільчих систем. Додаткові вимоги щодо безпеки».
- ДСТУ EN 60335-1:2015 «Прилади побутові та аналогічні електричні. Безпека. Частина 1. Загальні вимоги».
- ДСТУ EN 60950-1:2015 «Обладнання інформаційних технологій. Безпека. Частина 1. Загальні вимоги».
- ДСТУ EN 61140:2015 «Захист проти ураження електричним струмом. Загальні аспекти щодо установок та обладнання».

Останній стандарт особливо важливий для впровадження мір безпеки на робочому місці. Він визначає основні засоби захисту, такі як: основна ізоляція, огорожі і оболонки, бар'єри, обмеження напруги та інші. Він також регламентує норми кожної з цих мір. Наприклад, при застосуванні твердої основної ізоляції має унеможливлуватись контакт з небезпечними струмовідними частинами. Огорожі або оболонки мають унеможливлувати:

- у разі низьковольтних установок і устаткування, доступ до небезпечних струмовідних частин, забезпечуючи ступінь захисту проти ураження електричним струмом не менше за IPXXB (або IP2X) згідно з ІЕС 60529;
- у разі високовольтних установок і устаткування доступ у небезпечну зону, забезпечуючи ступінь захисту не менше за IPXXB (або IP2X) згідно з ІЕС 60529.

Для розробки даної системи з моніторингу і аналізу мережевих послуг, зокрема під час створення програмного забезпечення активно використовуються екрани типу LED/OLED. Тому під час перебування на робочому місці з використанням персонального комп'ютера важливо дотримуватись певних

нормативно-правових актів, щоб зменшити негативний вплив обладнання для здоров'я людини.

Закон України «Про затвердження вимог щодо безпеки та захисту здоров'я працівників під час роботи з екранними пристроями» вказує, що для забезпечення безпеки та захисту здоров'я працівників усе випромінювання від екранних пристроїв має бути зведене до гранично допустимого рівня з погляду безпеки та охорони здоров'я працівників. Також освітлення робочого місця працівника з екранними пристроями має створювати відповідний контраст між екраном і навколишнім середовищем та відповідати вимогам ДСанПІН 3.3.2.007-98. Згідно з ДСанПІН 3.3.2.007-98:

- розміщення робочих місць з ВДТ ЕОМ і ПЕОМ у підвальних приміщеннях, на цокольних поверхах заборонено.
- Площа на одне робоче місце має становити не менше ніж 6,0 м², а об'єм не менше ніж 20,0 м³.
- Природне освітлення має здійснюватись через світлові прорізи, орієнтовані переважно на північ чи північний схід і забезпечувати коефіцієнт природною освітленості (КПО) не нижче ніж 1,5%.
- Віконні прорізи приміщень для роботи з ВДТ мають бути обладнані регульованими пристроями.
- Для внутрішнього оздоблення приміщень з ВДТ слід використовувати дифузно-відбивні матеріали з коефіцієнтами відбиття для стелі 0,7-0,8, для стін 0,5-0,6.
- Покриття підлоги повинне бути матовим з коефіцієнтом відбиття 0,3-0,5. Поверхня підлоги має бути рівною, неслизькою, з антистатичними властивостями.
- Полімерні матеріали для внутрішнього оздоблення приміщень з ВДТ можуть бути використані при наявності дозволу органів та установ державної санітарно-епідеміологічної служби.
- У приміщеннях з ВДТ слід щоденно робити вологе прибирання.

- Приміщення з ВДТ мають бути оснащені аптечками першої медичної допомоги.

Мікроклімат виробничих приміщень з робочими місцями працівників з екранними пристроями має підтримуватись на постійному рівні та відповідати вимогам Санітарних норм мікроклімату виробничих приміщень ДСН 3.3.6.042-99. А робочий стіл або робоча поверхня повинні бути достатнього розміру та мати поверхню з низькою відбивною здатністю, допускати гнучкість під час розміщення екрана, клавіатури, документів і відповідного устаткування.

Згідно з законом не допускається:

- виконувати технічне обслуговування, ремонт і налагодження екранних пристроїв безпосередньо на робочому місці працівника під час роботи з екранними пристроями;

- відключати захисні пристрої, самочинно проводити зміни у конструкції та складі екранних пристроїв або їх технічне налагодження;

- працювати з екранними пристроями, у яких під час роботи виникають нехарактерні сигнали, нестабільне зображення на екрані та інші несправності.

Правила дотримання пожежної безпеки повинні бути з дотриманням норм Кодексу цивільного захисту України, що прийшов на заміну Закону “Про пожежну безпеку”. Кодекс затверджує правила пожежної та техногенної безпеки, норми, інструкції і методики, а також інші нормативно-правові акти з питань пожежної та техногенної безпеки, що є загальнообов’язковими для виконання, а також здійснює нормативно-правове регулювання щодо порядку організації та здійснення державного нагляду (контролю) у сфері пожежної та техногенної безпеки.

4.2. Безпека життєдіяльності

4.2.1. Забезпечення захисту працівників суб’єкта господарювання від іонізуючих випромінювань.

Одним з основних методів захисту від іонізуючого випромінювання є зменшення активності. Це означає зменшення кількості радіоактивного матеріалу в області. Це може бути досягнуто шляхом використання матеріалів з низькою радіоактивністю, обмеження використання радіоактивних матеріалів або зменшення часу використання радіоактивних матеріалів.

Іншим методом захисту від іонізуючого випромінювання є захист часом. Це може бути досягнуто шляхом обмеження часу роботи в області випромінювання, використання автоматизованих систем для виконання роботи в області випромінювання або використання дистанційного керування для виконання роботи в області випромінювання.

Захист відстанню є ще одним методом захисту від іонізуючого випромінювання. Він може бути досягнуто шляхом використання дистанційного керування, використання захисних екранів або використання автоматизованих систем для виконання роботи в області випромінювання.

Захист екраном є методом захисту від іонізуючого випромінювання, який включає в себе використання матеріалів, які блокують або поглинають випромінювання. Захист екраном може включати в себе використання свинцевих екранів, бетонних стін або водяних щитів.

Екранування, як правило, не потрібне для альфа-частинок, оскільки зовнішнє опромінення альфа-частинками не дає дози радіації. Там, де присутні тверді частинки, забруднені альфа-частинками можуть знадобитися засоби захисту органів дихання, щоб запобігти внутрішньому впливу та дозі. Під час роботи з джерелами рідини, які містять альфа-частинки, можуть знадобитися додаткові ЗІЗ, такі як рукавички, лабораторний халат і захисні окуляри, щоб запобігти забрудненню або потраплянню в очі. Під час роботи з бета-опромінюванням варто використовувати засоби екранування, разом з тими ж ЗІЗ. Гамма-промені і рентгенівські промені можуть долати кілометри в повітрі і проникати глибоко в організм людини або проходити через нього повністю, тому при роботі з ними має бути належне екранування.

4.2.2. Вплив проникаючої радіації ядерного вибуху на надійність роботи електронного обладнання. Ядерний вибух генерує інтенсивне проникаюче випромінювання, яке може серйозно пошкодити електронне обладнання. Це може призвести до втрати даних, пошкодження апаратного забезпечення та, у крайньому випадку, до відмови системи.

Види пошкоджень, які можуть виникнути внаслідок впливу проникаючої радіації можна розділити на одномоментні і поступові пошкодження. Одномоментні пошкодження можуть включати в себе вигорання або розряд, які можуть миттєво знищити електронний компонент. Оскільки електромагнітний імпульс, що виникає під час вибуху ядерної зброї є настільки потужним, що розглядається як один з факторів ураження від цієї зброї. Вибух створює потужне короткочасне електромагнітне поле що може досягати довжини більше 1 кілометра. Електричні поля та магнітні поля, що швидко змінюються, можуть з'єднуватися з електричними/електронними системами, створюючи руйнівні стрибки струму та напруги. Від цього також може постраждати апаратна інфраструктура, така як: металеві опори, антени, дроти ліній зв'язку та електропередач, трубопроводи тощо. При наземному і низькому повітряному вибухах вплив ураження від електромагнітного імпульсу спостерігається на відстані до декількох кілометрів від епіцентру вибуху. Також враховуючи одномоментні пошкодження радіація може викликати перенапругу в електронному пристрої, що може призвести до його вигорання, або викликати розряд в ізоляції, що може призвести до короткого замикання.

Проникаюча радіація також може викликати поступові пошкодження в електронних системах. Це може включати в себе втому матеріалу або деградацію ізоляції, які можуть з часом погіршити роботу системи. Радіація може викликати зміни в структурі матеріалу, що може призвести до його поступового зношування. Або вона може викликати деградацію ізоляції, що може призвести до втрати її електричних властивостей.

Втома матеріалу є процесом, який виникає внаслідок постійного впливу радіації на матеріал. Радіація може викликати зміни в структурі матеріалу, що може призвести до його поступового зношування. Також радіація може викликати дислокації в кристалічній структурі матеріалу, що може призвести до зміни його механічних властивостей. Це може призвести до зниження надійності електронного обладнання. Деградація ізоляції є іншим процесом, який може виникнути внаслідок впливу радіації. Радіація може викликати зміни в хімічній структурі ізоляційного матеріалу, що може призвести до зниження його електричних властивостей. Також вона може викликати окислення або вуглецеву деградацію в полімерних матеріалах, що може призвести до зниження їх діелектричної проникності, що приведе до зниження надійності електронного обладнання.

Цифрові пристрої, такі як комп'ютери та мікроконтролери, можуть бути особливо вразливими до пошкоджень від радіації. Це пов'язано з тим, що вони залежать від точності передачі даних. Проникаюча радіація може викликати помилки в передачі даних, що може призвести до неправильної роботи системи. Це може включати в себе помилки в програмному забезпеченні, що може призвести до неправильних обчислень або неправильної роботи системи. Аналогові пристрої, такі як сенсори та підсилювачі, можуть бути менш вразливими до радіації, але все ж можуть виявляти зміни в своїй роботі внаслідок впливу радіації.

Проникаюча радіація ядерного вибуху може серйозно пошкодити електронне обладнання, викликаючи різні види пошкоджень та впливаючи на їх функціональність та надійність. Однак, з розумінням цих впливів та відповідними стратегіями захисту, можна мінімізувати цей вплив і забезпечити надійність роботи системи.

4.3 Висновки

В даному розділі було вказано на стандарти умов праці на робочому місці, виявлено шкідливі і небезпечні фактор виробничого середовища і порівняно з діючими нормативами, розглянуто нормативні акти, що регламентують забезпечення охорони праці при роботі за ЕОМ

Було розглянуто питання про забезпечення захисту працівників суб'єкта господарювання від іонізуючих випромінювань, наведено методи захисту часом, відстанню, зменшення активності і способи екранування. Розглянуто вплив проникаючої радіації ядерного вибуху на надійність роботи електронного обладнання, розказано про існуючі небезпеки і виклики. Зроблено висновки про міри для захисту, яких повинно дотримуватись підприємство

ВИСНОВКИ

В рамках даного дослідження було розроблено математичне і програмне забезпечення системи моніторингу і аналізу постачання мережевих послуг. Було отримано низку вагомих результатів, що полягають в наступному.

1. Було проаналізовано джерела отримання даних для аналізу ефективності постачання мережевих послуг і було прийнято рішення про використання, окрім внутрішніх баз даних провайдера, даних моніторингу мережевого обладнання і системи для комунікації з клієнтами. Це дозволило забезпечити систему порівняльного аналізу ефективності роботи відділень інтернет-провайдера різнобічною інформацією.

2. Проведений аналіз програмного забезпечення для моніторингу телекомунікаційних мереж, зокрема сервіс Mikrotik The Dude, що дозволило мінімізувати кількість непотрібних функцій і максимізувати ефективність процесу моніторингу.

3. Проведений аналіз методів розробки програмного забезпечення і зроблено висновки про доцільність використання набору технологій MERN, запропоновано рішення для розвитку технологій React і Express шляхом підключення додаткових модулів, що дозволило збільшити ефективність роботи веб-ресурсу, забезпечило більше технологічних рішень для розробки й одночасно не вплинуло на швидкодію ресурсу.

4. Використовуючи методи обчислювальної математики, математичного моделювання, теорії ймовірності і масового обслуговування, було визначено параметри і розроблено математичне забезпечення, що створило умови для об'єктивного обрахунку ефективності роботи інтернет-провайдера використовуючи різні методи обчислень.

5. Налаштовано програму для моніторингу The Dude, створено можливості для виведення інформації в базу даних MySQL з подальшим використанням чат-бота, що дозволило забезпечити одночасне сповіщення працівників про виявлені проблеми разом з конвертацією повідомлень

моніторингу в нерелятивну базу даних і дозволило уникнути конфліктів в системі пов'язаних з використанням різних типів даних.

6. Практично реалізовано веб-ресурс, забезпечено зв'язок з базою даних MongoDB, реалізовано функції авторизації працівників і математичного обчислення визначених параметрів, що дало змогу поєднати в собі всі результати дослідження, забезпечити функціонування системи з аналізу і моніторингу, що дає можливість об'єктивно порівнювати різні параметри відділень інтернет-провайдера і забезпечити можливості для подальшого масштабування системи.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Kh R. “How Internet Service Providers Use Big Data Analytics To Help Customers” URL: <https://www.smartdatacollective.com/how-isp-use-big-data-analytics-to-help-customers> (дата звернення: 10.12.2023).
2. Lamberti A. “19 Network Metrics: How to Measure Network Performance” URL: <https://obkio.com/blog/how-to-measure-network-performance-metric> (дата звернення: 10.12.2023).
3. Mocerri P. “SNMP and Beyond: A Survey of Network Performance Monitoring Tool” Washington University in St.Louis, 2006.
4. “Observium documentation” URL: docs.observium.org (дата звернення: 10.12.2023).
5. Bock L. “Learn Wireshark: A definitive guide to expertly analyzing protocols and troubleshooting networks using Wireshark” , Packt Publishing, 2022.
6. “Mikrotik The Dude Documentation” URL: wiki.mikrotik.com/wiki/Manual:The_Dude/Interface (дата звернення: 10.12.2023).
7. Kredpattanakul K., Limpiyakorn Y. “Transforming JavaScript-Based Web Application to Cross-Platform Desktop with Electron. In: Kim, K., Baek, N. (eds) Information Science and Applications 2018.
8. Geetha S., Shah S., Shah S. “Coded Websites vs Wordpress Websites”, IJARST Vol. 12, i. 1, 2021.
9. Singh C. “MERN vs LAMP: Choices for Newbie Developers” International Journal of creatice research thoughts col 8, i. 8, 2020.
10. Holmes S. “Getting MEAN with Mongo, Express, Angular, and Node”, 2015.
11. Bhavyaa S. “Gupta Comprehensive Study of MERN Stack - Architecture, Popularity and Future Scope”, International Journal of Scientific Research in Computer Science, Engineering and Information Technology vol.3, i.6, 2021.

12. Ю.І. Залісковий, Ю.З. Лецишин, А.В. Варавін “Методи проведення моніторингу і аналізу мережевої інфраструктури Інтернет провайдерами”, Матеріали XI науково-технічної конференції Тернопільського національного технічного університету імені Івана Пулюя «Інформаційні моделі, системи та технології» (13-14 грудня 2023 року). Тернопіль: ТНТУ. 2023.

13. Ю.І. Залісковий, Ю.З. Лецишин, А.В. Варавін “Вибір технологій розробки веб-ресурсу моніторингу мережі Інтернет провайдерами”, Матеріали XI науково-технічної конференції Тернопільського національного технічного університету імені Івана Пулюя «Інформаційні моделі, системи та технології» (13-14 грудня 2023 року). Тернопіль: ТНТУ. 2023.

14. Петріна К. “Churn Rate під контролем: як обчислити відтік клієнтів”, URL: <https://esputnik.com/uk/blog/churn-rate-pid-kontrolem-yak-obchyslyty-vidtik-kliientiv> (дата звернення: 10.12.2023).

15. Юдін А. “Лідогенерація в девелопменті, гіпотези, лід-менеджмент і метрики”, URL: <https://marketer.ua/ua/lead-generation-in-development-hypotheses-lead-management-and-metrics> (дата звернення: 10.12.2023).

16. Головенко І.П. “Теоретико-методичні підходи до оцінювання економічної ефективності функціонування підприємств”, Миколаїв, 2016.

17. О.В. Жумік, М.О. Колінько “Застосування моделей черги для оптимізації роботи засобів зв’язку та електронного обладнання підприємства”, Міжнародний науковий журнал "Інтернаука", 2023. №4.

18. Wiley J. “Engineering Information Security: The Application of Systems Engineering Concepts to Achieve Information Assurance”, 2015.

19. Nandapurkar D. “Mastering MERN: The Comprehensive Guide to Full Stack Development” URL: https://medium.com/@gdsc_32043/mastering-mern-the-comprehensive-guide-to-full-stack-development-fb6c7d2f90c4 (дата звернення: 10.12.2023).

20. Subramanian V. “Pro MERN Stack”. 2019, 542 с.

21. Пристай Д. Онлайн 2020. Як пандемія вплинула на онлайн-користування“” URL: <https://suspilne.media/74631-onlajn-2020-ak-pandemia-vplnula-na-onlajn-koristuvanna> (дата звернення: 10.12.2023).
22. “Express.js documentation” URL: <https://expressjs.com/en/starter/generator.html> (дата звернення: 10.12.2023).
23. Samson-Ebere N. “How to Build a Full-Stack Authentication App With React, Express, MongoDB, Heroku, and Netlify” URL: <https://www.freecodecamp.org/news/how-to-build-a-fullstack-authentication-system-with-react-express-mongodb-heroku-and-netlify> (дата звернення: 10.12.2023).
24. Лещишин Ю. З., Романишин Н.Р., Наконечний В. В., Паламарчук А.О. Розробка системи зв'язку як інтегрованого елементу роботизованих систем // 36. тез доповідей XXI Всеукр. наук.-пр. конф. Житомир, 2016. С. 102.
25. Марків В.А., Осухівська Г.М., Лещишин Ю.З., Луцків А.М. Комп'ютерна система аутентифікації осіб // Матеріали XX наукової конференції ТНТУ ім. І. Пулюя. 2017. С. 90–91.
26. Leschyshyn Y., Scherbak L., Nazarevych O., Gotovych V., Tymkiv P., Shymchuk G. Multicomponent Model of the Heart Rate Variability Change-point // IEEE XVth International Conference on the Perspective Technologies and Methods in MEMS Design (MEMSTECH). 2019. P. 110–113.
27. Tymkiv P., Leshchyshyn Y. Algorithm Reliability of Kalman Filter Coefficients Determination for Low-Intensity Electroretinosignal // IEEE 15th International Conference on the Experience of Designing and Application of CAD Systems (CADSM). 2019. P.1-5.
28. Leschyshyn Y., Semchyshyn O. Periodically correlated heart rate variability detection by Neyman - Pearson criterion // 9th International Conference - The Experience of Designing and Applications of CAD Systems in Microelectronics. 2007. P. 139–140.
29. Стадник І.Я., Зварич Н.М. “Оцінка хімічної обстановки при аваріях на хімічно небезпечних об'єктах з викидом (вилівом) небезпечних хімічних речовин та застосуванні хімічної зброї ”ТНТУ. 2020. 36С.

Додаток А
Тези конференцій

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ТЕРНОПІЛЬСЬКИЙ НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ
УНІВЕРСИТЕТ ІМЕНІ ІВАНА ПУЛЮЯ

МАТЕРІАЛИ

XI НАУКОВО-ТЕХНІЧНОЇ КОНФЕРЕНЦІЇ

**«ІНФОРМАЦІЙНІ МОДЕЛІ,
СИСТЕМИ ТА ТЕХНОЛОГІЇ»**



13-14 грудня 2023 року

ТЕРНОПІЛЬ
2023

О.А. Дачук; Р.О. Жаровський УПРАВЛІННЯ ПОТОКОМ ЗА КРИТЕРІЯМИ ДОСТУПНОСТІ O.A. Diachuk; R.O. Zharovskyi FLOW CONTROL BY ACCESSIBILITY CRITERIA	151
Ю.І. Залісковий, Ю.З. Лещишин, А.В. Варавін МЕТОДИ ПРОВЕДЕННЯ МОНИТОРИНГУ І АНАЛІЗУ МЕРЕЖЕВОЇ ІНФРАСТРУКТУРИ ІНТЕРНЕТ ПРОВАЙДЕРАМИ Y.I. Zaliskovyi, Y.Z. Leshchyslyn, A.V. Varavin METHODS OF MONITORING AND ANALYSIS OF NETWORK INFRASTRUCTURE BY INTERNET PROVIDERS	152
Ю.І. Залісковий, Ю.З. Лещишин, А.В. Варавін ВИБІР ТЕХНОЛОГІЙ РОЗРОБКИ ВЕБ-РЕСУРСУ МОНИТОРИНГУ МЕРЕЖІ ІНТЕРНЕТ ПРОВАЙДЕРАМИ Y.I. Zaliskovyi, Y.Z. Leshchyslyn, A.V. Varavin SELECTION OF TECHNOLOGIES FOR THE DEVELOPMENT OF A WEB RESOURCE FOR NETWORK MONITORING BY INTERNET PROVIDERS	153
І. Кардаш, Ю. Лещишин, А. Варавін КРИТЕРІЇ ЕФЕКТИВНОСТІ РОБОТИ ДЛЯ ЗАДАЧІ МОНИТОРИНГУ ЛОКАЛЬНОЇ МЕРЕЖІ I. Kardash, Yu. Leshchyslyn, A. Varavin WORK EFFICIENCY CRITERIA FOR THE LOCAL NETWORK MONITORING TASK	154
І. Кардаш, Ю. Лещишин, А. Варавін МОНИТОРИНГ ЕФЕКТИВНОСТІ РОБОТИ ЛОКАЛЬНИХ МЕРЕЖ I. Kardash, Yu. Leshchyslyn, A. Varavin MONITORING OF THE EFFICIENCY OF LOCAL NETWORKS	155
Н.М. Ковтун; Р.О. Жаровський АЛГОРИТМІЧНЕ ЗАБЕЗПЕЧЕННЯ СИСТЕМ ВИЯВЛЕННЯ ВТОРГНЕНЬ N.M. Kovtun; R.O. Zharovskyi ALGORITHMIC PROVISION OF INTRUSION DETECTION SYSTEMS	156
Д. Козарук, Ю. Лещишин МОДЕЛЮВАННЯ МЕТОДІВ ПОТОКОВОГО ШИФРУВАННЯ ТА ПЕРЕДАВАННЯ ФОТОГРАФІЧНИХ ЗОБРАЖЕНЬ D. Kozaryk; Yu. Leshchyslyn SIMULATION OF STREAM ENCRYPTION METHODS AND TRANSMISSION OF PHOTOGRAPHIC IMAGES	157
Д. Козарук; Ю. Лещишин МЕТОДИ ТА ЗАСОБИ ПОБУДОВИ КОМП'ЮТЕРНОЇ СИСТЕМИ ДЛЯ ПОТОКОВОГО ШИФРУВАННЯ ТА ПЕРЕДАВАННЯ ФОТОГРАФІЧНИХ ЗОБРАЖЕНЬ D. Kozaryk; Yu. Leshchyslyn METHODS AND MEANS FOR CONSTRUCTING A COMPUTER SYSTEM FOR STREAM ENCRYPTION AND TRANSMISSION OF PHOTOGRAPHIC IMAGES	158
Т. І. Крамар; С. В. Туш СУЧАСНІ ТЕХНОЛОГІЇ РОБОТИ КОМП'ЮТЕРИЗОВАНИХ СИСТЕМ ЕКСПОРТУ ЕЛЕКТРОЕНЕРГІЇ T. I. Kramar; Ie. Tysh MODERN WORK TECHNOLOGIES COMPUTERIZED ELECTRICITY EXPORT SYSTEMS	159
Т. І. Крамар; С. В. Туш МЕТОДИ ТА ЗАСОБИ ЗАБЕЗПЕЧЕННЯ СТАБІЛЬНОГО ФУНКЦІОНУВАННЯ ЕЛЕКТРОМЕРЕЖ ПІД ЧАС ПОГОДНИХ АНОМАЛІЙ. T. I. Kramar; Ie. Tysh METHODS AND MEANS OF ENSURING STABLE FUNCTIONING OF ELECTRICAL NETWORKS DURING WEATHER ANOMALIES	160

УДК 004.738

Ю.І. Залісковий, к.т.н.; Ю.З. Лещишин, к.ф.-м. н. А.В. Варавін

Тернопільський національний технічний університет імені Івана Пулюя, Україна

МЕТОДИ ПРОВЕДЕННЯ МОНІТОРИНГУ І АНАЛІЗУ МЕРЕЖЕВОЇ ІНФРАСТРУКТУРИ ІНТЕРНЕТ ПРОВАЙДЕРАМИ

Y.I. Zaliskovyi, Ph.D.; Y.Z. Leshchynshyn, Ph.D. N. A.V. Varavin

METHODS OF MONITORING AND ANALYSIS OF NETWORK INFRASTRUCTURE BY INTERNET PROVIDERS

Інтернет- провайдери відіграють незамінну роль у сучасному світі, забезпечуючи доступ до інформації та різноманітних онлайн- сервісів. Однак, щоб залишатися конкурентоспроможними, провайдерам необхідно постійно покращувати якість своїх послуг і розвивати територіальні зони своєї роботи, забезпечуючи здатність до розширення [1].

Одним з основних способів покращення продуктивності роботи постачальника мережеских послуг є модернізація його інфраструктури [2]. Щоб забезпечити ефективну реалізацію модернізації інфраструктури основним і незамінним методом є моніторинг мережі. Проте, одної тільки наявності даних отриманих від програм моніторингу, зазвичай, недостатньо. Тому для аналізу ефективності роботи мереж можна використовувати отримані дані в математичних формулах і моделях [1].

Отримати інформацію про всі аспекти моніторингу можна за допомогою Mikrotik The Dude. Вибір програмного забезпечення з моніторингу мереж є доволі обширним, але The Dude є простим у використанні, надійним і пристосованим до українських реалій. Після налаштування процесу перевірки отримати дані моніторингу можна перевіряючи кожен окремий пристрій. Проте, для масштабних мереж використовують журналювання (логінг)- процес експорту отриманих даних про мережу. Знаючи характеристики і отримавши дані моніторингу мережі можна для розрахунку важливих параметрів пропускну здатності [1] і ефективності застосовувати формули, такі як:

$$E = \frac{D_{\text{факт}}}{D_{\text{макс}}} \cdot 100 \quad (1)$$

де: E - ефективність мережі (у відсотках), $D_{\text{факт}}$ - фактично передані дані за певний період часу. $D_{\text{макс}}$ - максимально можлива кількість даних, які могли б бути передані за той самий період часу.

$$B = \frac{D}{t} \quad (2)$$

де: B - пропускну здатність в бітах на секунду, D - загальний обсяг переданих даних в бітах, t - час передачі в секундах.

Отже, збільшення ефективності роботи інтернет-провайдера вимагає комплексного підходу. Покращення інфраструктури потребує не тільки аналіз даних з моніторингу мереж, але і якісної аналітики даних за допомогою математичних функцій.

Література

1. А. А. Скопа, С. Л. Волков, і К. Б. Айвазова, «ПРОБЛЕМАТИКА ЯКОСТІ ПОСЛУГ ІНТЕРНЕТ-ПРОВАЙДЕРІВ», Збірник наукових праць Одеської державної академії технічного регулювання та якості, вип. 1 (2), с. 27-31, Чер 2013.
2. R. Khan, "An Efficient Network Monitoring and Management System," Int. J. Inf. Electron. Eng., vol. 3, no. 1, 2013.

УДК 004.43

Ю.І. Залісковий, к.т.н.; Ю.З. Лещишин, к.ф.-м. н. А.В. Варавін
(Тернопільський національний технічний університет імені Івана Пулюя, Україна)

ВИБІР ТЕХНОЛОГІЙ РОЗРОБКИ ВЕБ-РЕСУРСУ МОНІТОРИНГУ МЕРЕЖІ ІНТЕРНЕТ ПРОВАЙДЕРАМИ

Y.I. Zaliskovyi, Ph.D.; Y.Z. Leshchynshyn, Ph.D. N. A.V. Varavin

SELECTION OF TECHNOLOGIES FOR THE DEVELOPMENT OF A WEB RESOURCE FOR NETWORK MONITORING BY INTERNET PROVIDERS

Створення веб-ресурсу є важливою частиною розвитку і підвищення ефективності інтернет-провайдера. Якісний веб-сайт дозволить не тільки покращити і налагодити комунікацію з користувачами, але й збільшити їхню кількість. Перед розробкою веб-ресурсу потрібно визначити відповіді на два головні запитання “Які функції повинен виконувати сайт” і “Які технології найкраще підійдуть для його розробки”.

Веб-ресурси для інтернет-провайдерів повинні інформувати користувачів про актуальну інформацію компанії, таку як: тарифи, акції чи новини. Також веб-сайт повинен підтримувати можливість комунікації з клієнтами і збору їхніх даних. Ця та інша інформація повинна бути проаналізована, можливості для цього також повинні бути доступні за обмеженим доступом на сайті. З останньої вимоги також витікає створення можливостей для авторизації працівників, а в майбутньому для поширення технології до особистих кабінетів.

З визначених вимог стає зрозуміло що розробка потребуватиме залучення різних технологій. Під час аналізу сукупностей технологій для розробки можна виділити три популярні технології: LAMP Stack, що включає Linux, Apache, MySQL та PHP, WordPress Stack (WordPress, PHP, MySQL), MERN Stack (MongoDB, Express.js, React, Node.js). Вони дають змогу легко поєднувати користувацьку і серверну частину проекту. Технологія MERN виділяється технологічністю та можливостями для масштабування [1]. Релятивні бази даних представлені в технології LAMP і WordPress більш поширені серед програмної інфраструктури при роботі з мережами, що дає їм перевагу, але технологічність MERN [2].

Використовуючи React розробники можуть замінити інші фреймворки: Angular.js і Vue, але завдяки поширеності і великій кількості готових рішень варто працювати з React [1]. За допомогою синтаксису JSX від React створюється інтерфейс з яким можуть взаємодіяти користувачі. Дані залишені ними повинні оброблятися серверною частиною за допомогою зв'язки Express та Node.js та передаватися в базу даних MongoDB. Для окремих задач для реалізації функцій веб-ресурсу застосовується головна перевага MERN- технологічність і наявність готових засобів. Для авторизації можна застосовувати модуль Passport.js, для створення модальних вікон з формами- бібліотек “react-modal”, для аналізу даних за допомогою математичних функцій — модуль Math.js, для демонстрації функцій з аналітикою даних — Chart.js.

Таким чином, правильно визначивши необхідні функції і підібравши технології можна створити веб-ресурс який буде відповідати всім критеріям і зберігати можливості для масштабування в майбутньому.

Література

1. Bhavyaa, S. “Gupta Comprehensive Study of MERN Stack - Architecture, Popularity and Future Scope”, International Journal of Scientific Research in Computer Science, Engineering and Information Technology vol.3, i.6, 2021.
2. A. Kapoor, “MERN vs LAMP”, Enlear Academy, 2021.