

**ПОЛТАВСЬКИЙ ДЕРЖАВНИЙ АГРАРНИЙ УНІВЕРСИТЕТ
НАВЧАЛЬНО-НАУКОВИЙ ІНСТИТУТ ЕКОНОМІКИ, УПРАВЛІННЯ,
ПРАВА ТА ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ
КАФЕДРА ІНФОРМАЦІЙНИХ СИСТЕМ ТА ТЕХНОЛОГІЙ**

Пояснювальна записка

до кваліфікаційної роботи на здобуття ступеня вищої освіти магістр

на тему: **«Автоматизація вебпроцесів та розроблення сервісу для
моніторингу працездатності вебресурсів»**

Виконав: здобувач вищої освіти
за освітньо-професійною програмою
Інформаційні управляючі системи та
технології спеціальності
126 Інформаційні системи та технології
ступеня вищої освіти магістр
групи 126ІСТмд_21
Кулага Б.А.
Керівник: Уткін Ю. В.
Рецензент: Петраш Р.В.

Полтава – 2023 року

ВСТУП

Інформаційний простір мережі інтернет має колосальний обсяг і вплив, без перебільшення, на всі сфери життя і діяльності світової спільноти. Згідно зі звітом одного з відомих моніторингових вебсайтів [1] станом на вересень 2023 року у світі налічують 1 106 671 903 вебсайти, з яких тільки 201 898 446 вебсайтів, тобто 18 %, є активними. Тенденція на розробку персональних вебсайтів набрала обертів настільки стрімко, що кожної хвилини в світі з'являється приблизно 175 нових.

На сьогодні немає можливості точно визначити кількість вебсайтів, які припинили своє існування. Проте, можна впевнено стверджувати, що існує значна кількість вебресурсів, які закриваються з різних причин, включаючи технічні несправності та кібератаки. Проте, точна кількість таких сайтів і пропорція відносно всіх існуючих сайтів є доволі складною для визначення. Процеси закриття сайтів можуть бути дуже різними і залежать від багатьох факторів, таких як:

- Рівень безпеки сайту: сайти з низьким рівнем безпеки можуть бути більш вразливими перед кібератаками та іншими загрозами. Якщо власники сайту не забезпечують належну захист, це може призвести до закриття сайту.
- Тип атак та їхні наслідки: вид атаки і її наслідки можуть варіюватися від незначних до серйозних. Серйозні атаки, такі як DDoS-атаки або злами безпеки, можуть призвести до неможливості функціонування сайту, що може призвести до його закриття [2].
- Реакція власників сайту: власники сайтів можуть реагувати по-різному на атаки та несправності. Деякі намагаються відновити свої сайти, тоді як інші можуть вирішити закрити їх через серйозні проблеми. Наприклад, втрата бази даних з інформацією про облікові записи користувачів.

Основною проблемою для активного існування вебсайтів є те, що власникам складно вчасно виявити їх недоступність. Це шкодить репутації сайту серед користувачів та пошукових системам, котрі будуть зменшувати рейтинг

вебсайту через недоступність. Тут на допомогу приходять сервіси для моніторингу доступності. Моніторинг – це процес систематичного спостереження, вимірювання і контролю над певними процесами [3].

Актуальність теми кваліфікаційної роботи тісно пов'язана зі зростанням кількості інтернет-ресурсів, важливістю вебпроцесів у різних сферах, конкурентоспроможність бізнесу, потребою в забезпеченні безпеки даних, інноваціями в галузі ІТ, а також прямим впливом на реальні проблеми бізнесу та організацій.

Метою кваліфікаційної роботи є створення та дослідження властивостей інноваційного сервісу, який допоможе автоматизувати та контролювати вебпроцеси для підвищення ефективності, безпеки та надійності вебресурсів.

Завданнями кваліфікаційної роботи є:

- здійснення всебічного аналізу концепції моніторингу вебресурсів;
- обґрунтування рішень та підходів для розробки системи моніторингу доступності вебресурсів;
- розроблення оригінального сервісу моніторингу доступності вебресурсів із використанням готових рішень ;
- оцінювання ефективності застосування систем моніторингу доступності вебресурсів.

Об'єктом дослідження кваліфікаційної роботи є процеси і засоби моніторингу вебресурсів та методи вивчення, аналізу та вдосконалення цих процесів у контексті автоматизації створення і застосування.

Предметом дослідження є системи моніторингу вебпроцесів та готові рішення для їх розроблення.

Методи досліджень: аналітико-синтетичний, інформаційно-пошуковий, дедуктивний, порівняльний, абстрактний, моделювання реалізації розробки сервісу.

Інформаційну базу кваліфікаційної роботи складають наукові публікації, спеціалізована література, освітні відеоматеріали, статистичні дані від

аналітичних компаній, що розміщені у вільному доступі на вебсайтах, і нормативні документи, стандарти в галузі інформаційних технологій.

Практична значущість кваліфікаційної роботи виявляється у можливості розробки інноваційного сервісу, який спрощує і поліпшує процеси вебавтоматизації та моніторингу. Цей сервіс може відігравати важливу роль у підвищенні ефективності та надійності вебпроектів, що корисно для бізнесу, організацій та окремих користувачів.

Апробація результатів дослідження відбувалася шляхом оприлюднення доповідей на міжнародній та студентських конференціях. За результатами проведеного дослідження опубліковано тези: «Порівняння сервісів моніторингу доступності вебсайтів», I Міжнародна науково-практична конференція, 28 вересня 2023 р., м. Полтава; «Функціонал сервісу моніторингу доступності вебсайтів», XX щорічний міждисциплінарний семінар «Студентські роботи за науковою тематикою кафедри інформаційних систем та технологій», 29 листопада 2023 р., м. Полтава.

Структура кваліфікаційної роботи логічно пов'язана з задачами досліджень, пояснювальна записка містить перелік умовних познач, вступ, три розділи основної частини, висновки, список використаних джерел. Загальний обсяг текстової частини кваліфікаційної роботи складає 62 сторінки формату А4, містить 17 рисунків і 3 таблиці. В роботі використано посилання на 41 науково-технічне інформаційне джерело.

РОЗДІЛ 1

АНАЛІЗ КОНЦЕПЦІЇ ВЕБМОНІТОРИНГУ

1.1 Сутність концепції вебмоніторингу

Концепція вебмоніторингу полягає в систематичному спостереженні та контролі за вебресурсами з метою забезпечення їхньої надійності, продуктивності та доступності. Ця ідея передбачає постійний моніторинг вебсайтів і додатків для виявлення можливих проблем, включаючи збої серверів, помилки на вебсторінках та інші аспекти, які можуть впливати на користувацький досвід та ефективність вебресурсів. Аналіз отриманих даних дозволяє приймати рішення щодо оптимізації та виправлення проблем з метою покращення функціонування вебресурсів.

Моніторинги вебсайтів з'явилися після того, як Інтернет став популярним і доступним для користувачів. Початкові інструменти для вебмоніторингу виникли у 1990-х рр., коли стало важливим відстежувати роботу вебресурсів [4].

Якщо описати процес простою мовою то моніторинг – це умовний користувач який регулярно відвідує вебсайт та коли помічає що трапляється помилка або недоступність він одразу ж повідомить про це власника сайту.

Схема того, як працюють моніторинги доступності вебсайтів зображена на рис. 1.1.

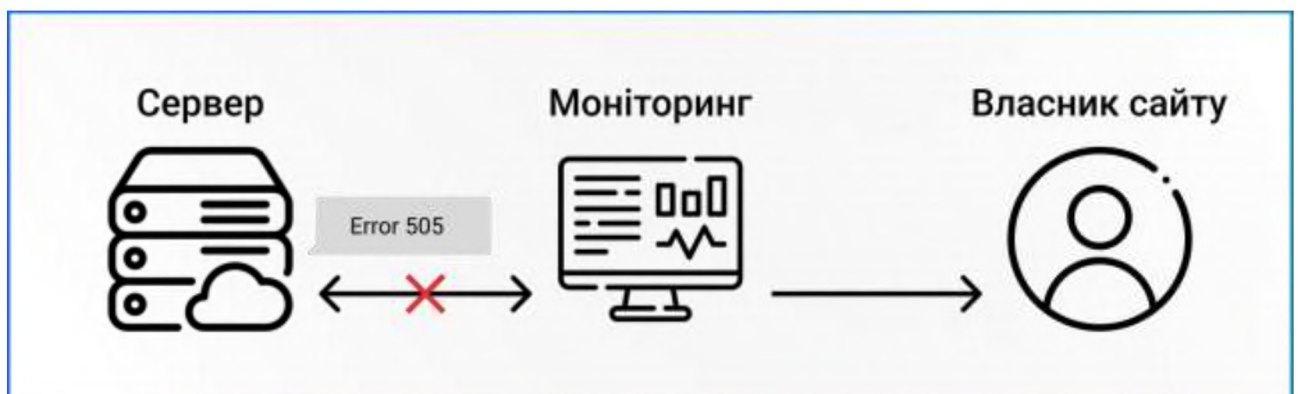


Рисунок 1.1 – Схема роботи моніторингів

Спочатку моніторинги надавали обмежені функції, такі як перевірка доступності вебсайту та повідомлення про недоступність. Протягом часу ці інструменти розвивалися і розширювали свої можливості, включаючи аналіз швидкості завантаження сторінок, виявлення помилок, моніторинг продуктивності серверів та інші функції для забезпечення надійності та ефективності вебресурсів.

Зараз існує безліч різних сервісів та програм для вебмоніторингу, які надають різноманітні інструменти для забезпечення якості та доступності вебсайтів та додатків, такі як:

- Uptimebot.com – це провідна у світі служба моніторингу безвідмовної роботи. На сьогоднішній день їх послугами користуються більше ніж 2 100 000 користувачів з усього світу [5];

- Uptime-monitor.io – це потужний інструмент для забезпечення доступності користувацьких вебсайтів і серверів. За допомогою цього сервісу користувач може легко відстежувати свої онлайн-активи та отримувати сповіщення про будь-які проблеми чи збої [6];

- Visualping.io – це інтернет-сервіс, який дозволяє відстежувати візуальні зміни на вебсайтах. Основна мета Visualping – автоматично відслідковувати будь-які зміни на вебсторінках та повідомляти користувача про них [7].

Більшість цих сервісів були створені за для однієї цілі, відстеження працездатності веб ресурсів, але з часом через велику конкуренцію були зобов'язані розвивати свої сервіси та впроваджувати нові функції, котрі закривали інші проблеми власників сайтів, такі як:

Моніторинг SSL сертифікату – це процес постійного контролю та перевірки дійсності й правильності використання SSL сертифікатів на вебсерверах, вебсайтах. SSL-сертифікати використовуються для забезпечення захищеної передачі даних між вебсервером і користувачем, зокрема шифрування інформації [8].

Моніторинг терміну дії домену – це процес постійного відстеження та контролю за терміном дії доменного імені (вебадреси) в інтернеті [9]. Це важлива

процедура для власників вебсайтів і доменів, оскільки домени мають обмежений строк дії і потребують продовження реєстрації для збереження права на використання цього домену. Моніторинг терміну дії домену за допомогою сервісу моніторингу доступності вебсайтів зазвичай полягає в періодичній перевірці часу, який залишився до закінчення терміну дії домену. Цей процес може бути автоматизованим та виконуватися через інтервали часу, визначені користувачем. Система моніторингу може виконувати запити до DNS-серверів або переглянути термін дії в довіднику Whois (рис. 1.2), щоб отримати інформацію про термін дії домену [10]. Після цього вона може порівняти дату поточного дня з датою закінчення терміну дії домену та визначити залишковий термін дії. Якщо термін дії домену наближається до закінчення або вже минув, система моніторингу може сповістити власника про це, щоб він міг вчасно продовжити реєстрацію домену та уникнути його втрати.

matchmarketing.agency

Інформація реєстру whois домену - matchmarketing.agency

Domain Name: MATCHMARKETING.AGENCY
 Registry Domain ID: 41b677319c194ae8a69d38cca9859dc5-DONUTS
 Registrar WHOIS Server: whois.ukraine.com.ua
 Registrar URL: <https://www.ukraine.com.ua>
 Updated Date: 2022-10-17T12:42:08Z
 Creation Date: 2020-11-17T13:57:08Z
 Registrar Registration Expiration Date: 2024-11-17T13:57:08Z
 Registrar: HOSTING UKRAINE LLC
 Registrar IANA ID: 2374
 Registrar Abuse Contact Email: domain@abuse.team
 Registrar Abuse Contact Phone: +380.443927433
 Domain Status: ok <http://www.icann.org/epp#ok>
 Registry Registrant ID: ICN-442410-BOAUM
 Registrant Name: Privacy Protection
 Registrant Organization: Hosting Ukraine LLC
 Registrant Street: PO Box 65
 Registrant City: Kiev
 Registrant State/Province:
 Registrant Postal Code: 04112
 Registrant Country: UA
 Registrant Phone: +380.443927433
 Registrant Phone Ext:
 Registrant Fax:
 Registrant Fax Ext:
 Registrant Email: aa3466a9545608ad3b@privacyprotected.net
 Registry Admin ID: ICN-442410-BOAUM
 Admin Name: Privacy Protection
 Admin Organization: Hosting Ukraine LLC
 Admin Street: PO Box 65
 Admin City: Kiev
 Admin State/Province:
 Admin Postal Code: 04112
 Admin Country: UA

Рисунок 1.2 – Приклад звіту по запиту довідника Whois

Моніторинг роботи Cron – це процес відстеження та контролю за виконанням завдань на розкладі, який встановлений за допомогою програми Cron в операційній системі. Програма Cron дозволяє користувачам планувати виконання автоматичних завдань, таких як запуск скриптів, виконання резервних копій, відправка повідомлень тощо, за певним графіком.

Моніторинг пінгу (англ. ping) – це процес вимірювання часу, необхідного для передачі сигналу (пакету даних) від пристрою до сервера та повернення його назад [10]. Цей метод використовується для визначення доступності сервера та часу відповіді. Він дозволяє перевіряти частоту відповідей сервера та виявляти можливі проблеми зі з'єднанням, такі як затримки, втрати даних або недоступність сервера. Відслідковування пінгу може бути важливим для визначення роботи сервера та його доступності для користувачів. У контексті сервісу моніторингу доступності, перевірка пінгу використовується для визначення часу відповіді сервера та його доступності. Система відправляє пакет даних до сервера та очікує на його відповідь. Якщо відповідь отримана, система вимірює час, який затрачений на відправку та повернення цього пакету. Якщо цей час перевищує певне налаштоване значення або якщо відповідь не надійшла, то це може вказувати на проблеми з доступністю сервера або затримки у з'єднанні.

1.2 Технології та підходи моніторингу доступності вебресурсів

На сьогоднішній день будь-який користувач інтернету або велика компанія може безкоштовно забезпечити моніторинг свого сайту всього в декілька кліків. Більшість компаній користуються послугами моніторингу для економії часу та фінансової частини на розробку такого сервісу. Але є й такі, котрі бажають розробити власний сервіс моніторингу з різних причин. Їх може бути безліч, найпоширеніші з них – це відсутність сервера моніторингу в необхідній користувачу країні. Наприклад, користувач із Казахстану не бажає, щоб його

сервер моніторингу знаходився у Франції. Це обумовлюється тим, що час відповіді на запит (Request Response Time) буде некоректний через велику відстань між серверами, що може мати вплив на статистику аптайму.

Request Response Time – це показник, що вказує на час затримки між відправленням запиту до сервера і отриманням відповіді від сервера [12]. Цей показник вимірюється в мілісекундах або секундах і визначає швидкість, з якою сервер обробляє запити. Час відповіді запиту важливий для оцінки продуктивності вебсайту або вебдодатку. Він включає час, який затрачається на передачу запиту через мережу до сервера, обробку запиту на сервері та повернення відповіді до клієнта. Цей показник важливий для користувачів, оскільки відображається на швидкодії вебсайту: маленьке значення часу відповіді свідчить про швидкий вебсайт, тоді як велике значення може вказувати на проблеми з продуктивністю сервера або мережі.

Аптайм (англ. uptime) – це період часу, протягом якого система, сервер, комп'ютер або сервіс працює та функціонує без відмови або перебоїв в доступності [13]. Термін аптайм використовується для вимірювання часу, коли система або пристрій є доступним та готовим до роботи. На відміну від аптайму, даунтайм (англ. downtime) вказує на період часу, коли система або сервіс недоступний або не працює. Аптайм вимірюється у відсотках часу, і вираз "99,9% аптайм" вказує, що система доступна приблизно 99,9% часу, або приблизно 8 годин 45 хвилин на місяць недоступності. Аптайм є важливим параметром для оцінки надійності системи та її доступності для користувачів.

Для розроблення власного сервісу моніторингу доступності вебсайту можна виділити три підходи, які наведені нижче.

1. Скористатися готовим рішенням для моніторингу вебсайту та модифікувати систему отримання сповіщень.
2. Створити власну систему моніторингу за допомогою інтеграції з лідерами ринку, таким як Prometheus.io.
3. Розробити власний підхід за допомогою програмного забезпечення (ПЗ) zennoposter.

Перший підхід більш легкий, оскільки не потребує багато часу та специфічних знань для налаштування та розгортання системи. Для цього буде достатньо всього двох сервісів, а саме:

1. Make.com – це онлайн-сервіс для автоматизації робочих процесів та інтеграції різних вебдодатків і сервісів без необхідності писати код. Він надає користувачам зручний інтерфейс для створення власних автоматизованих сценаріїв, які можуть об'єднувати дії в різних додатках і послугах, щоб автоматизувати рутинні завдання та оптимізувати робочі процеси [14].

2. Hetrixtools.com – це безкоштовний вебсервіс для моніторингу доступності вебсайтів і серверів. Цей сервіс надає можливість власникам вебсайтів та адміністраторам серверів відстежувати роботу своїх онлайн-проектів і вчасно виявляти будь-які проблеми або відмови [15].

Другий підхід більш складний в реалізації через те, що потребує розроблення за допомогою написання коду та інтеграції з іншими сервісами. Або як один з варіантів, взяти до роботи готове ПЗ з відкритим кодом Prometheus та інтегрувати його під свої потреби.

Prometheus – це система моніторингу та спостереження, яка використовується для збору та аналізу метрик інфраструктури, додатків та сервісів у реальному часі [16]. Вона спеціально призначена для моніторингу різноманітних систем та компонентів у складних середовищах. Prometheus використовує модель збору метрик через HTTP, де додатки та системи відправляють свої метрики до Prometheus сервера, який потім збирає, обробляє і зберігає ці дані. Він дозволяє виконувати моніторинг стану різноманітних ресурсів, включаючи сервери, мережеві компоненти, контейнери, бази даних, сервіси та інші складові системи. Основні переваги Prometheus – це гнучкість у зборі метрик, широкий спектр підтримуваних інтеграцій з різними системами, можливість моніторингу в реальному часі та можливість використання мови запитів для аналізу метрик. Ця система дозволяє оперативно виявляти та реагувати на проблеми в інфраструктурі або додатках для підтримки їх ефективної роботи.

Цей метод доволі кропіткий та потребує більше часу, адже користувачу доведеться вчити технічну документацію та проводити безліч тестів, аби кінцевий продукт відповідав усім вимогам надійності та безпеки, адже його основною задачею буде моніторинг доступності вебсайту.

Третій метод є компромісом між першими двома, адже в цьому випадку користувачу доведеться знайомитися з технічною документацією ПЗ zennoposter, але при цьому не потрібні знання в програмуванні, оскільки всі дії виконуються в зручному графічному інтерфейсі. Також користувачів, які щойно відкрили для себе ПЗ, існує форум з великою кількістю користувачів, котрі допоможуть з вирішенням проблем [17].

У порівнянні підходів найбільш цікавими з міркувань ефективності на виділений час можна вважати перший та третій методи. Адже вони більше підходять для користувачів або власників сайтів, які хочуть отримати власний моніторинг без вивчення для цього мови програмування та їх аспектів.

Велику увагу приділяють методам сповіщення власників сайтів або технічних працівників про недоступність вебресурсу, які можуть здійснюватися за допомогою різних методів і засобів (рис 1.3).

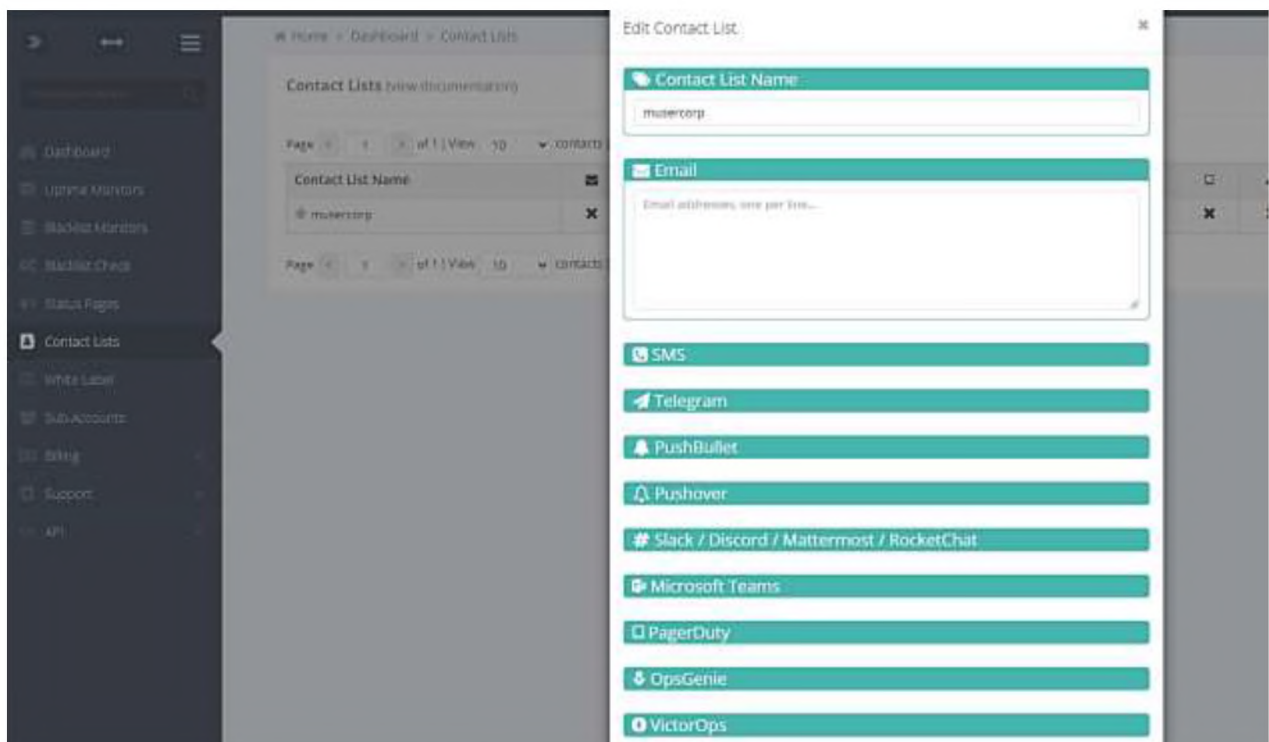


Рисунок 1.3 – Перелік способів сповіщення власника сайту про недоступність

Найпопулярнішими методами (каналами) сповіщень є:

- Електронна пошта: сповіщення можуть автоматично відправлятися на вказану електронну адресу власника сайту або технічного адміністратора. Це може бути важливим каналом комунікації для оперативного реагування на проблеми.
- SMS-повідомлення: сповіщення можуть також надсилатися через SMS на мобільний телефон власника сайту або технічного працівника. Це особливо корисно, якщо потрібна швидка реакція.
- Повідомлення месенджерів: деякі системи моніторингу можуть надсилати повідомлення через популярні месенджери, такі як Telegram, Slack або WhatsApp.
- Дзвінки: в деяких випадках, коли проблема надзвичайно важлива, може бути використана система автоматичних дзвінків для сповіщення.
- Внутрішні системи слідкування: технічні працівники можуть мати доступ до внутрішніх систем слідкування або моніторингу, які відображають стан вебресурсу.
- Інтеграція з системами керування завданнями: якщо організація використовує системи керування завданнями (наприклад, JIRA або ServiceNow), сповіщення можуть автоматично створювати та оновлювати завдання для вирішення проблеми.
- Зовнішні системи сповіщення: користувач може використовувати сторонні системи сповіщення, такі як PagerDuty, VictorOps або Opsgenie, для організації та управління сповіщеннями.
- Інші канали сповіщення: в залежності від потреби організації можуть використовуватися інші канали сповіщення, такі як факси, робочі сповіщення або автоматичні записи на голосовій пошті.

Вибір методу сповіщення може залежати від конкретних вимог вашої організації та тимчасових обмежень. Головна мета – забезпечити оперативне сповіщення та реагування на проблеми з доступністю вебресурсу [18].

1.3 Методи і засоби моніторингу доступності

Обрана система моніторингу встановлює підключення до вебресурсу шляхом визначення інтерфейсу зв'язку між самою системою та вебресурсом, який потрібно відстежувати. Це об'єднує можливість системи взаємодіяти з вебресурсом, надсилати запити для перевірки його доступності та отримувати відповіді для подальшого аналізу. Підключення встановлюється на основі налаштованих параметрів, включаючи URL-адресу вебресурсу, методи моніторингу, інтервал перевірок та очікуваний статус. Цей етап дозволяє системі почати процес постійного моніторингу та відстежування доступності вебресурсу згідно з заданими параметрами [19].

Задані параметри моніторингу визначають конфігурацію системи моніторингу та способи перевірки доступності вебресурсу. Це включає в себе визначення URL-адреси вебресурсу, який потрібно відстежувати, вибір методів моніторингу (наприклад, HTTP-запити, пінг, перевірка SSL-сертифікату), встановлення інтервалу моніторингу (частоти перевірок), визначення очікуваного статусу вебресурсу при нормальному функціонуванні та інших параметрів, що впливають на процес моніторингу. Задані параметри дозволяють системі налаштовувати, яким чином вона буде виконувати перевірки та реагувати на можливі проблеми з доступністю вебресурсу згідно з користувацькими вимогами та налаштуваннями.

Система виконує перевірки доступності шляхом виконання певних дій та операцій, визначених в параметрах моніторингу. Це включає в себе надсилення запитів до вебресурсу, який вона відстежує, та отримання відповідей. Способи перевірки можуть варіюватися від використання HTTP-запитів для перевірки статусу вебсайту до відправлення пінг-запитів для перевірки доступності сервера. Під час кожної перевірки система порівнює отримані результати з очікуваним статусом, визначеним користувачем в налаштуваннях. Якщо система виявляє, що вебресурс недоступний або відповідає з помилкою, це реєструється

як проблемний стан та може спровокувати подальші дії, які вказані в системі для реагування на такі ситуації.

Основний процес моніторингу доступності включає постійні з'єднання з сервером, на якому працює вебсайт, що перевіряється, та перевірку відповіді цього сервера при спробі відкрити конкретну URL-адресу.

Вебсайт розглядається як доступний, якщо сервер повертає відповідь «200 ОК», яка може включати код перенаправлення (наприклад, «301» або «302») [20]. Будь-яка інша відповідь, а також відсутність відповіді протягом певного часу, призведе до реєстрації та повідомлення про недолік – це зазвичай відбувається після перевірки проблеми з декількох джерел.

Таблиця статусів сервера HTTP має наступні коди:

- 1xx (Інформаційні): ці коди - інформаційні повідомлення, які показують, що запит отримано і обробляється. Наприклад, код «100» означає, що сервер отримав запит і готовий до продовження.

- 2xx (Успішні): коди цієї групи показують, що запит клієнта було успішно оброблено сервером. Наприклад, «200» вказує на успішну відповідь на запит.

- 3xx (Перенаправлення): ці коди використовуються для вказівки клієнтові на те, що додаткові дії потрібні для завершення запиту. Наприклад, «301» або «302» вказують на потребу перенаправлення на іншу сторінку або URL.

- 4xx (Помилки клієнта): ці коди вказують на помилки, що виникли через неправильні або недійсні запити від клієнта. Наприклад, «404» - це код для помилки «сторінка не знайдена».

- 5xx (Помилки сервера): коди цієї групи показують, що виникли помилки через проблеми на сервері під час обробки запиту клієнта. Наприклад, «500» - це внутрішня помилка сервера.

Ці коди дозволяють явно передавати інформацію про статус виконання запиту і можуть бути використані для адміністрування, відлагодження та вирішення проблем, пов'язаних з вебсайтами.

Один із прикладів повідомлення про недоступність сайту як результат моніторингу показано на рис. 1.4.

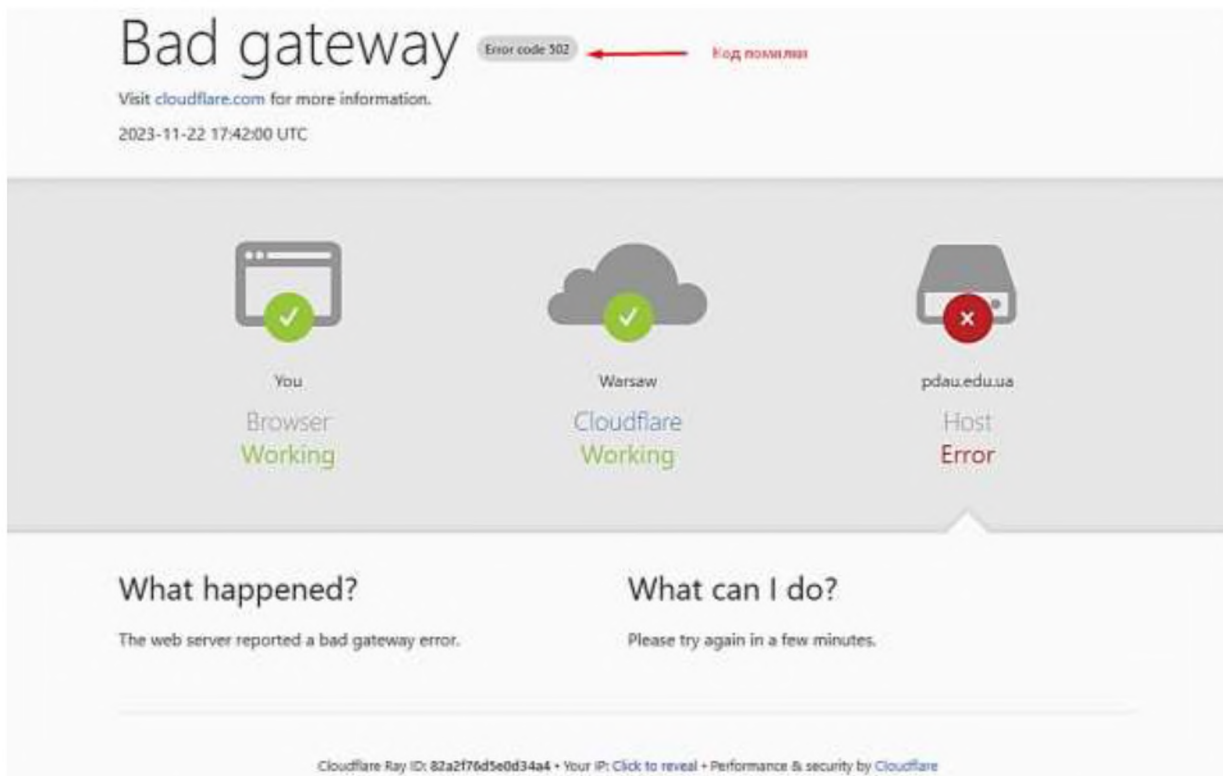


Рисунок 1.4 – Приклад повідомлення про недоступність сайту за кодом 502

При моніторингу доступності завантажуються лише HTTP-заголовки, що спричиняє мінімальне навантаження на сервер та залишається невидимим для пакетів відстеження трафіку, таких як Google Analytics [21].

Google Analytics – це безкоштовний вебаналітичний сервіс, який надає детальну статистику та звіти про вебсайт. Цей інструмент від Google дозволяє власникам вебресурсів відстежувати різноманітні показники, такі як кількість відвідувачів, їхні поведінкові патерни, джерела трафіку, конверсії, взаємодію з контентом та багато іншого [22].

За допомогою Google Analytics можна детально проаналізувати трафік на вебсайті, визначити ефективність маркетингових кампаній, зрозуміти, які сторінки привертають більше уваги від користувачів, ідентифікувати проблеми у вебпроцесах та приймати обґрунтовані рішення для покращення присутності в мережі.

Завдяки Google Analytics можна побачити проблеми з доступом або недоступність сайту за конкретний період: це завжди видно по діаграмі кількості сеансів (рис 1.5) та користувачів. Повний екран сервісу наведено в додатку Б.

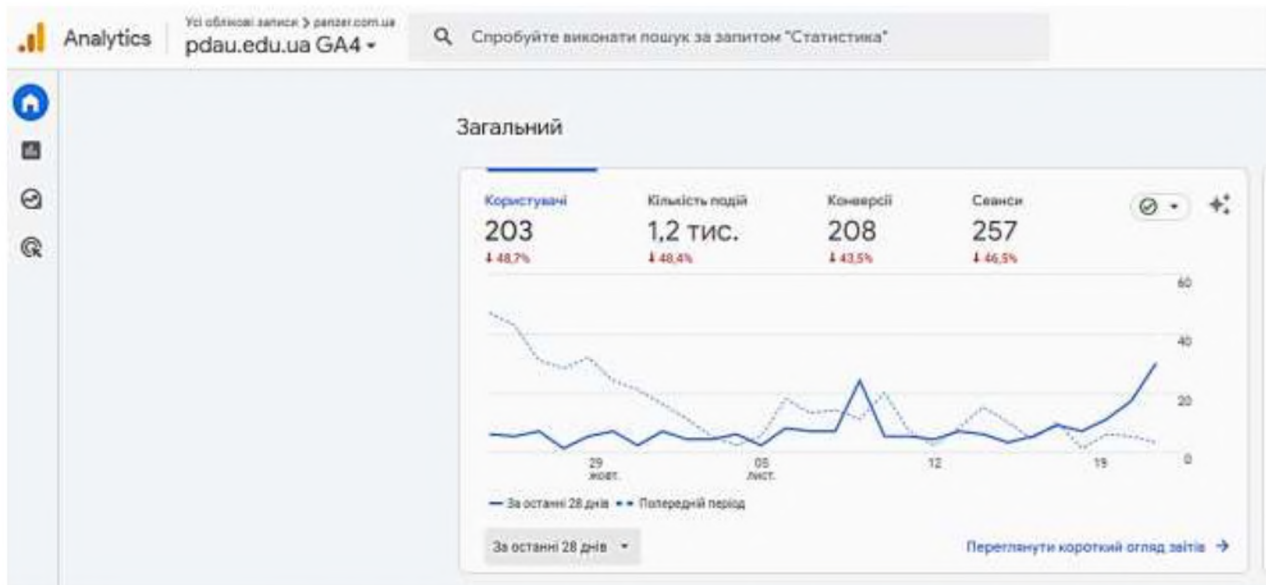


Рисунок 1.5 – Фрагмент інтерфейсу Google Analytics

Виявлення проблем з доступністю вебресурсу запускає процес визначення неполадок або аномалій, які можуть виникнути з доступністю вебресурсу. Це включає в себе аналіз результатів перевірок, які система виконує на основі налаштованих параметрів. Якщо система виявляє, що вебресурс не відповідає на запити, видає помилки або відсутній для доступу, це реєструється як проблема. Виявлення проблем може також включати аналіз інших аномалій, таких як помилки SSL-сертифікатів чи низька швидкість завантаження сторінки. Детальне виявлення проблем дозволяє оперативно вживати заходи для вирішення проблем та забезпечення максимальної доступності вебресурсу для користувачів.

Окрім того, можливі й автоматичні заходи: якщо система виявляє проблеми або аномалії під час перевірок доступності вебресурсу, то вона автоматично активує визначені заходи для вирішення проблеми. Це може включати в себе, наприклад, спроби автоматичного відновлення роботи вебсайту, перезавантаження сервера, встановлення попередньо налаштованих заходів для захисту від кібератак або надсилання повідомлень адміністраторам для подальшої ручної відладки. Також система може перенаправити трафік на резервний сервер або CDN для забезпечення найкращої доступності для користувачів. Автоматичні заходи допомагають забезпечити швидку реакцію на

проблеми та зменшити вплив недоступності вебресурсу на користувачів і бізнес-процеси.

Отже, дослідження методів та засобів моніторингу показує, що моніторинг є ключовим елементом для забезпечення працездатності вебресурсів та має стратегічне значення для бізнесу. Основний аналіз показав, що цей інструмент дозволяє вчасно виявляти проблеми доступності вебресурсів, забезпечуючи власників сайтів необхідною інформацією для оперативного реагування на виниклі проблеми.

1.4. Огляд аналітичних можливостей сервісів моніторингу доступності

Сервіси моніторингу доступності надають різноманітну корисну аналітику, яка допомагає розуміти стан доступності вебсайтів, додатків або сервісів для користувачів (рис. 1.6).



Рисунок 1.6 – Приклад екрану аналітики в системі моніторингу доступності

Аналітика доступності вебсайтів, додатків та сервісів є ключовою для забезпечення стабільної та надійної роботи цих систем. Вона дозволяє виявляти

та вирішувати проблеми, що можуть виникнути під час їх експлуатації або детальніше зрозуміти передумови та наслідки проблеми. Аналізуючи продуктивність, швидкість завантаження, час відновлення після відмови та інші метрики, ця аналітика допомагає розуміти, наскільки ефективно працює система. Вона також сприяє вчасному реагуванню на проблеми, покращує користувацький досвід, допомагає в плануванні майбутніх вдосконалень та збільшує загальну надійність та ефективність системи. Таким чином, аналітика доступності не лише запобігає відмовам, а й сприяє покращенню роботи системи та задоволенню користувачів.

Одним з ключових показників аналітики є час відновлення після відмови MTTR – це метрика, яка вимірює середній час, необхідний для відновлення роботи системи після виявлення проблеми або відмови [23]. Ця метрика важлива для розуміння того, як швидко та ефективно можна відновити функціональність системи після виникнення проблеми. MTTR включає в себе не лише час виявлення проблеми, але й час, необхідний для виявлення її причини, виправлення та повернення системи до робочого стану. Ця метрика допомагає оцінити ефективність команди підтримки та управління технічною підтримкою.

Мінімізація MTTR важлива для підтримки безперебійної роботи системи та задоволення потреб користувачів. Чим швидше проблему виявлено, виправлено та відновлено роботу системи, тим менш вплив на користувачів та бізнес. У зусиллях по зниженню MTTR важливо вдосконалювати процеси моніторингу, автоматизації виявлення та виправлення проблем, а також підвищувати навички та ефективність команди підтримки, щоб максимально скоротити час відновлення після відмови та забезпечити найкращий користувацький досвід.

Аналітика сервісу доступності, також тісно пов'язана з показником «Продуктивність та швидкість завантаження», оцінює швидкість та продуктивність роботи вебсайту, додатку чи сервісу для кінцевих користувачів. Цей аспект аналізу включає в себе декілька важливих метрик та показників, що допомагають зрозуміти, наскільки ефективно система працює для користувачів:

- Час завантаження сторінки/додатку: вимірює час, який потрібний для завантаження вмісту сторінки або додатку. Швидке завантаження важливе для забезпечення позитивного користувацького досвіду та уникнення втрати потенційних користувачів через довгий час очікування.

- Швидкість відповіді сервера: цей показник вказує на час, який сервер витрачає на обробку запитів користувачів. Швидка відповідь сервера дозволяє забезпечити швидке взаємодію з системою та знизити час очікування.

- Час рендерингу сторінки: це вимірює час, необхідний браузеру для відображення вмісту сторінки після отримання від сервера. Швидкий час рендерингу важливий для відчуття користувачем швидкості та плавності роботи вебсторінки чи додатку.

- Використання ресурсів: аналіз того, як ефективно система використовує свої ресурси (наприклад, пам'ять, процесор). Це важливо для оптимізації роботи системи та забезпечення її стабільності при високому навантаженні.

Аналітика продуктивності та швидкості завантаження надає важливі дані для розуміння того, наскільки швидко та ефективно працює система для користувачів. Вона допомагає виявляти слабкі місця в роботі системи та покращувати її продуктивність для досягнення кращого користувацького досвіду.

Географічний розподіл доступності – це іще один важливий аспект аналітики сервісу доступності, який вивчає доступність системи з різних географічних регіонів чи місць. Цей пункт оцінює, наскільки ефективно система працює для користувачів з різних частин світу або конкретних локацій. Моніторинг географічного розподілу доступності допомагає виявити можливі проблеми, які можуть виникнути у певних регіонах або при підключенні з конкретних місць. Основні аспекти цієї аналітики включають:

- Перформанс залежно від місця з'єднання: оцінюється, як швидко та ефективно користувачі отримують доступ до сервісу з різних локацій. Наприклад, чи може впливати велика відстань до сервера на швидкість завантаження або роботу системи.

- Доступність з різних регіонів: вимірює, наскільки ефективно система працює для користувачів з різних частин світу. Це дозволяє виявити можливі проблеми з доступністю для певних географічних аудиторій.

- Стійкість доступності у різних регіонах: аналізує, як часто виникають перерви у роботі системи у різних місцях. Це дозволяє виявити можливі технічні проблеми зв'язку чи інфраструктури, що впливають на доступність.

- Тестування стійкості в різних мережевих умовах: оцінює, як система працює при різних умовах мережі, таких як швидкість інтернет-з'єднання, тип підключення (мобільна мережа, Wi-Fi, провідний доступ), що важливо для користувачів з різних мереж та умов.

Аналітика географічного розподілу доступності допомагає підтримувати стабільну і надійну роботу системи для користувачів з будь-якої точки світу, виявляючи можливі проблеми та сприяючи у покращенні доступності для всіх аудиторій.

Також деякі сервіси ведуть аналітику стійкості до високого навантаження, що оцінює, наскільки ефективно система працює під великим та інтенсивним навантаженням. Цей пункт аналізу є важливим для визначення того, як система реагує на збільшення обсягу запитів та користувацького трафіку, забезпечуючи стабільність та продуктивність у таких умовах.

Окремі аспекти аналітики стійкості систем до високого навантаження включають:

- Тестування продуктивності системи при максимальному навантаженні: Оцінюється, як система працює при піковому навантаженні, коли кількість запитів або користувацький трафік зростає значно вище середнього.

- Складність у роботі під великим трафіком: аналізується, як система обробляє та відповідає на запити при великій кількості одночасних користувачів чи великому обсязі даних.

- Тестування навантаження: проводяться тести для виявлення меж, до яких може витримати система під час великого навантаження, щоб визначити, коли може виникнути перевантаження та проблеми зі швидкістю реакції.

– Моніторинг в пікові часи: систематичне спостереження за роботою системи під час періодів пікового навантаження для виявлення можливих проблем та вдосконалення її продуктивності.

Аналіз стійкості до високого навантаження допомагає визначити, наскільки ефективно система працює в умовах великого обсягу запитів та трафіку. Ця аналітика допомагає виявляти потенційні межі та слабкі місця у роботі системи, що дозволяє підготувати та оптимізувати її для ефективної роботи під високим навантаженням.

Кілька слів про аналітику журналів подій (logs), аналіз та використання журналів подій для моніторингу та виявлення проблем в системі [24]. Основні аспекти аналізу журналів подій включають:

– Збір та збереження даних: системи моніторингу збирають дані про події, які виникають у системі. Це може бути інформація про запити користувачів, помилки, дії адміністраторів тощо.

– Аналіз та інтерпретація інформації: зібрані журнали подій аналізуються для виявлення подій, відмінностей або аномалій, що можуть вказувати на проблеми в системі. Цей аналіз може включати пошук патернів або несправностей.

– Виявлення проблем та налагодження системи: аналіз журналів подій допомагає ідентифікувати проблеми та потенційні небезпеки для системи, або вживати заходів для їх попередження.

– Моніторинг відповідності до стандартів та регуляторних вимог: журнали подій часто використовуються для виконання вимог щодо звітності, аудиту та відповідності стандартам безпеки даних.

– Аналіз кореляції подій: виявлення причинно-наслідкових зв'язків та складних взаємозв'язків.

Аналітика журналів подій допомагає забезпечити контроль та відстеження дій у системі, виявляти та усувати проблеми, забезпечуючи стабільність, безпеку та доступність сервісів. Вона є важливою частиною моніторингу та діагностики для забезпечення надійності та ефективності системи.

Висновки до розділу 1

Інтенсивність збільшення кількості вебсайтів запустила процес розробки сервісів моніторингу доступності вебсайтів. Вебмоніторинг з'явився на етапі популяризації інтернету та став важливим для відстеження функціонування вебресурсів. Початково ці моніторинги були обмежені лише перевіркою доступності, але зараз вони надають багато інструментів для забезпечення якості та доступності вебсайтів і додатків. Тепер вони мають в рази більший функціонал у порівнянні з попередниками, на основі прогнозування здатні попереджати про проблеми ще до того, як вони настали.

Сьогодні майже будь-хто за наявності вебресурсу може легко скористатися послугами моніторингу для ефективного контролю його доступності. Велика частина компаній обирає такий підхід, оскільки це заощаджує час і фінанси, які раніше були б витрачені на розробку власних моніторингів. Однак, деякі організації можуть бути зацікавлені у створенні власних моніторингів з різних причин, наприклад, за відсутності серверів моніторингу в потрібній локації.

Особлива увага приділяється методам сповіщень власників сайтів чи технічних працівників про недоступність вебресурсу. Вибір методу сповіщення залежить від вимог організації та часових обмежень, але головна його мета полягає в оперативному сповіщенні та реагуванні на проблеми доступності вебресурсу.

Важливо також визначити параметри моніторингу, які встановлюють конфігурацію системи та способи перевірки доступності вебресурсу. Ці параметри допомагають системі виконувати перевірки та реагувати на проблеми доступності згідно з користувацькими налаштуваннями.

Цей розділ надає важливі концептуальні аспекти моніторингу та підкреслює його важливість у сфері управління вебпроцесами. Розглянуті аспекти дозволять визначити ключові переваги використання моніторингу доступності вебресурсів та його вплив на підвищення ефективності бізнесу й користувацького досвіду.

РОЗДІЛ 2

ОБҐРУНТУВАННЯ ВИБОРУ РІШЕНЬ ТА ПІДХОДІВ ДЛЯ РОЗРОБКИ МОНІТОРИНГУ ДОСТУПНОСТІ ВЕБРЕСУРСІВ

2.1 Вибір підходу розроблення моніторингу доступності вебресурсів

Вибір підходу розроблення моніторингу доступності часто базується на таких складових як час розробки, витрати на розробку, ефективність, тощо. В даній роботі розглянуто два підходи, котрі будуть найшвидші в реалізації та найменш затратні для власника сайту або технічного працівника.

Перший підхід – найлегший в реалізації та економічно вигідний, оскільки, завдяки правильному підбору технологій, може бути безкоштовним для власника сайту. Це підхід із застосуванням готового рішення для моніторингу доступності веб ресурсів від компанії HetrixTools та інтеграція його з Telegram bot арі за допомогою сервісу автоматизації вебпроцесів «Make».

У цьому підході інструменти hetrixtools відповідають за моніторинг доступності вебсайту, вебсерверу, SMTP-серверу та сповіщень про недоступність ресурсу за допомогою інструменту WebHook.

WebHook – це механізм, який дозволяє вебдодаткам отримувати реальні чи миттєві сповіщення про зміни або події, які відбуваються на іншому вебсервісі чи додатку [25]. Зазвичай, WebHook використовується для автоматизації та спрощення обміну даними між різними вебсервісами. Коли відбувається певна подія чи зміна в джерелі, до якого підключено WebHook, він надсилає HTTP-запит до попередньо визначеного URL-адреси вебдодатку-приймача, що вказує на те, що подія відбулася. Приймаюча сторона може обробити це сповіщення та виконати необхідні дії. WebHook дозволяє реалізувати взаємодію між різними додатками та сервісами в реальному часі, сприяючи автоматизації процесів, обміну даними та обробці подій на вебплатформах.

Далі приходить час для інструменту Make, котрий являє собою міст між першим сервісом та ботом в месенджері Telegram. Його основна задача –

прийняти повідомлення в форматі JSON, переформовувати отримане повідомлення в зручний для користувача формат та надіслати до бота Telegram.

JSON – це формат обміну даними, який використовується для передачі структурованих даних між комп'ютерами [26]. Він є легким текстовим форматом, що базується на синтаксисі JavaScript, але використовується для зберігання та обміну даними в багатьох сучасних мов програмування. Використовується для представлення об'єктів та масивів у вигляді пар ключ-значення (рис 2.1). Він є популярним форматом обміну даними в мережевому програмуванні, особливо в веброзробці та взаємодії з вебсерверами. JSON дозволяє передавати структуровані дані, такі як текст, числа, об'єкти та масиви, що полегшує їх обробку та читання як людьми, так і комп'ютерами.



```

1  {
2    "data": [
3      {
4        "type": "articles",
5        "id": "1",
6        "attributes": {
7          "title": "Working with JSON Data in python",
8          "description": "This article explains the various ways to work with JSON data in python.",
9          "created": "2020-12-28T14:56:29.000Z",
10         "updated": "2020-12-28T14:56:28.000Z"
11       },
12       "author": {
13         "id": "1",
14         "name": "Aveek Das"
15       }
16     ]
17   }
18 }

```

Рисунок 2.1 – Приклад ключів JSON

Месенджер Telegram в цьому випадку відіграє роль зручного методу отримання інформації для власника сайту чи адміністратора вебресурсу. Мінуси цього підходу полягають в тому, що інструмент hetrixtools не має великої кількості серверів по світу, тому, більше за все, підійде не всім користувачам мережі інтернет, а сервіс Make має обмеження по кількості дій в середині інструменту для безкоштовного тарифного плану до 1000 дій в місяць. Завдяки

цьому, якщо автоматизація багатокрокова, можна вичерпати ліміт дій і в потрібний момент не отримати повідомлення про недоступність сервісу.

Другий підхід базується на застосуванні готового програмного забезпечення ZennoPoster та його інтеграції з Telegram Bot Api.

ZennoPoster – це програмне забезпечення для автоматизації дій в інтернеті яке дозволяє створювати та виконувати різноманітні автоматизовані завдання [27], такі як парсинг даних з вебсайтів, автоматичне заповнення форм, взаємодія з вебсервісами, робота з соціальними медіа та багато іншого. ПЗ має графічний інтерфейс, що дозволяє користувачам створювати скрипти шляхом перетягування та розміщення блоків. Ці блоки представляють різноманітні дії, такі як клікання, заповнення форм, робота з базами даних, взаємодія з API і т. ін. Використовується для автоматизації рутинних завдань в інтернеті, що дозволяє зекономити час та зусилля у порівнянні з виконанням цих завдань вручну.

API – це набір правил, протоколів та інструментів, які дозволяють різним програмам взаємодіяти одна з одною [28]. Він визначає способи комунікації між різними програмами, додатками або сервісами, щоб вони могли обмінюватися даними та виконувати певні функції. API може бути у формі бібліотек, які визначають правила виклику функцій, HTTP-методів, які дозволяють отримувати чи відправляти дані, або набору інструкцій, які програми використовують для спілкування між собою. Він може бути використаний для побудови інтеграцій, розширення функціональності програм або спрощення розробки нових додатків. API може мати різні види, такі як Веб API, бібліотеки, оперативні системи, апаратне забезпечення тощо. Вони є ключовою складовою сучасних програмних систем, які дають можливість програмістам створювати взаємодіючі програми та сервіси.

Завдяки цьому ПЗ власник сайту може створити бота який буде заходити на сайт та перевіряти його доступність. Основною перевагою використовувати цей метод є те що на відміну від інструменту «hetrixtools» бот в «ZennoPoster» може робити комплексну перевірку вебсайту на наявність елементів на сторінці або в цілому доступності сайту.

Основним мінусами є те що ПЗ ZennoPoster не має безкоштовної версії, а функціонал місцями настільки складний, що новачку доведеться звертатися за допомогою до аудиторії на офіційному форумі чи шукати відповідь в довіднику. Також ZennoPoster не підтримує жодних операційних систем, окрім сімейства Windows.

2.2 Обладнання для розгортання сервісу перевірки доступності

Розгортання сервісу моніторингу доступності вебресурсів вимагає наявності серверних ресурсів. Це може бути фізичний сервер або віртуальний сервер, який розміщений в хмарі. При використанні фізичного сервера необхідно вибрати апаратне забезпечення, що відповідає потребам проекту: процесор, обсяг оперативної пам'яті, місце на диску тощо. В разі використання віртуального сервера в хмарі, можна вибрати певний план хостингу, який підходить за обсягом ресурсів [29].

Також важливо вибрати оптимальну операційну систему для сервера. Зазвичай використовуються операційні системи, такі як Linux або Windows Server, залежно від сумісності з використовуваним програмним забезпеченням.

Крім того, для роботи сервера можуть знадобитися інші компоненти, такі як вебсервер, наприклад, Apache, Nginx, база даних, а також додаткове ПЗ для налаштування середовища та забезпечення безпеки, фаєрволи, програмне забезпечення антивірусної захисту, системи моніторингу безпеки тощо.

Nginx – це вебсервер з відкритим кодом та проксі-сервер, який забезпечує високопродуктивне обслуговування вебсайтів [30]. Він відомий своєю ефективністю та швидкодією у роботі з великим обсягом одночасних з'єднань, а також здатністю ефективно обробляти статичний контент, асинхронно виконувати запити та функціонувати як проксі-сервер. Nginx також може використовуватися як балансувальник навантаження для розподілу трафіку між серверами. Він широко використовується у сфері хостингу, веброботи та

вебсервісів завдяки своїй надійності, продуктивності та можливостям конфігурації. Усі ці складові обираються враховуючи потреби та специфіку проекту, а також можливості та вимоги до безпеки та надійності інфраструктури.

Для глобального сервісу моніторингу доступності вебсайтів необхідно мати розподілену мережу серверів, розміщених у різних частинах світу. Це забезпечить швидкий та ефективний доступ до сервісу для користувачів з різних географічних регіонів [31]. Розподілена мережа серверів також дозволить зменшити затримки при виконанні перевірок доступності вебресурсів і підвищить загальну продуктивність сервісу. Технічно надійне розміщення серверів у відповідних дата-центрах забезпечить стабільну та безперебійну роботу сервісу, а також гарантуватиме високу доступність для користувачів.

Можливим недоліком може бути вартість управління та підтримки розподіленої мережі серверів. Збільшення кількості серверів, розташованих у різних частинах світу, зазвичай призводить до збільшення витрат на їх оренду, обслуговування та адміністрування. Крім того, забезпечення стабільної роботи всіх цих серверів потребує постійної уваги до мережесхемних проблем, резервування, оновлення програмного забезпечення, що також може збільшувати витрати на технічне обслуговування. Компанії повинні бути готові до вкладення додаткових ресурсів у підтримку та управління такою розподіленою мережею, щоб забезпечити її ефективну та безперебійну роботу для користувачів усього світу.

Мінімальні системні вимоги для сервера, який виконує моніторинг доступності вебсайтів, можуть бути наступними [32]:

1. Процесор: двоядерний процесор з частотою 2 ГГц або вище.
2. Оперативна пам'ять (RAM): мінімум 2 ГБ оперативної пам'яті.
3. Місце на диску: достатньо простору для зберігання баз даних та інших системних файлів, зазвичай 20 ГБ або більше.
4. Мережевий інтерфейс: підтримка Ethernet з достатньою швидкістю для взаємодії з Інтернетом та перевіркою вебсайтів.

Рекомендовані системні вимоги можуть бути вищими в залежності від розміру системи та кількості вебсайтів, які планується моніторити. Наприклад:

1. Процесор: чотирьох ядерний процесор зі значним запасом потужності, наприклад, Intel Core i5 або еквівалентний.
2. Оперативна пам'ять (RAM): рекомендується мінімум 8 ГБ RAM або більше для оптимальної продуктивності.
3. Місце на диску: додаткове місце на диску для резервних копій та збереження логів, які можуть займати значну кількість простору.
4. Мережевий інтерфейс: гігабітне підключення до Інтернету для швидкої перевірки доступності вебсайтів.

Зазвичай компанії розробники вебсервісів орендують серверні потужності для розробки своїх сервісів. Адже це економічно вигідніше за рахунок того що немає необхідності купувати серверне обладнання, займатися його «колокацією» та постійним обслуговуванням.

Колокація – це послуга, яку надають центри обробки даних (ЦОД). Вона полягає в тому, що компанії можуть розмістити своє обладнання (сервери, обладнання мережі, зберігання даних тощо) у спеціально обладнаному приміщенні, яке належить зовнішньому постачальнику послуг ЦОД [33]. Це може бути приватний сервер або обладнання, яке використовується для підтримки онлайн-сервісів та додатків. Колокація дозволяє компаніям мати повний контроль над своїм обладнанням, при цьому вони забезпечуються високою швидкістю і доступністю мережі завдяки інфраструктурі ЦОД. Найкращім варіантом отримання серверних потужностей є аренда VPS/VDS

VPS (Virtual Private Server) та VDS (Virtual Dedicated Server) є схожими концепціями в галузі віртуалізації серверів, де один фізичний сервер розділяється на окремі віртуальні сервери (рис 2.2).

VPS – це віртуальне просторове середовище, де декілька користувачів використовують один фізичний сервер, розділяючи ресурси цього сервера [34]. Він надається у вигляді віртуального середовища з власною операційною системою, обладнанням та ресурсами. Користувачі можуть встановлювати власні програми і мати контроль над окремими параметрами свого віртуального сервера.

VDS – це схоже поняття, але з дещо іншим підходом. Тут ресурси фізичного сервера також розділяються між користувачами, проте кожен віртуальний сервер має власну операційну систему, що робить його більш незалежним від інших віртуальних серверів на цьому фізичному сервері. У випадку VDS користувачі мають більше контролю над конфігурацією та можливостями свого сервера.

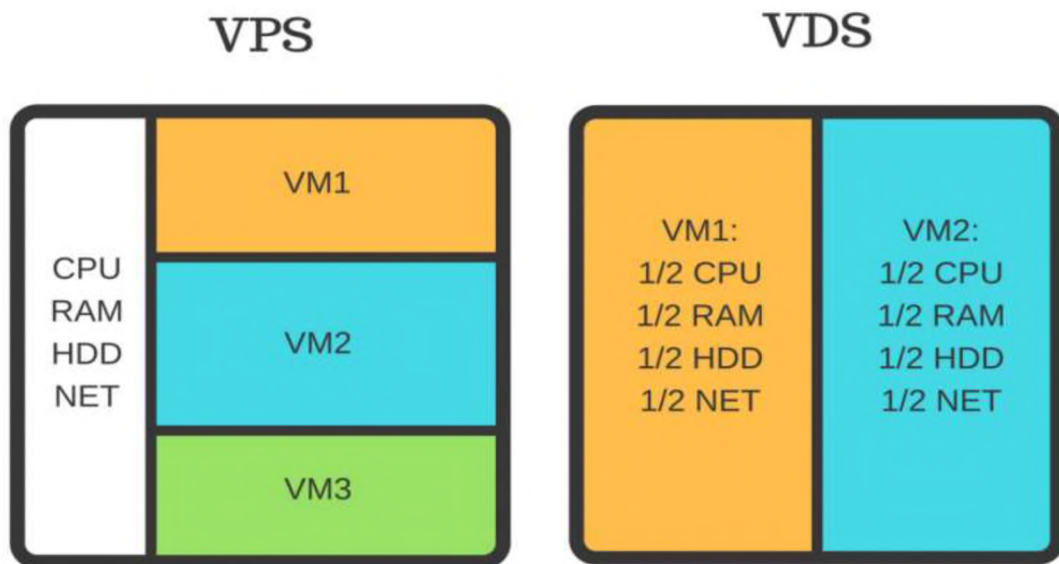


Рисунок 2.2 – Візуалізація різниці між VPS та VDS [34]

VPS і VDS пропонують користувачам кілька переваг:

- Гнучкість: полягає в їх здатності відповідати змінюваним потребам користувача. Ці рішення дозволяють масштабувати ресурси відповідно до поточних вимог проекту. Вони забезпечують гнучкість у зміні обсягів процесорного часу, обсягів пам'яті та потреб дискового простору. Користувачі можуть легко збільшувати або зменшувати ресурси відповідно до потреб, що дозволяє оптимізувати продуктивність та ефективність роботи віртуальних серверів в реальному часі.

- Вартість: використання віртуальних приватних серверів (VPS) або віртуальних серверів з відведеною пам'яттю (VDS) може мати економічні переваги без необхідності інвестування в фізичне серверне обладнання. Не потрібно вкладати велику суму грошей у придбання фізичного обладнання, що дозволяє економити на витратах на придбання та підтримку обладнання.

Користувач може легко масштабувати ресурси сервера вгору або вниз, потрібне тільки підключення до потрібної кількості обчислювальних ресурсів. Вартість використання VPS або VDS базується на оплаті за фактично використані ресурси, що може бути економічніше, ніж оплата за повне фізичне обладнання, навіть у випадку, коли потрібно тимчасово збільшити ресурси.

– Ізоляція ресурсів: ресурси відрізняються від спільних вебхостингових рішень у тому, що вони забезпечують ізольоване середовище для кожного користувача. Це означає, що ресурси, такі як CPU, RAM, дисковий простір, мережевий трафік і таке інше, виділяються конкретним користувачам і не діляться з іншими користувачами на фізичному сервері. Така ізоляція ресурсів забезпечує більшу стабільність та безпеку, оскільки вплив на ресурси інших користувачів обмежений. Це також означає, що користувачі можуть вільно керувати своїм віртуальним сервером, встановлювати програмне забезпечення, налаштовувати параметри та використовувати ресурси без втручання інших.

– Швидкість налаштування: швидкість налаштування віртуальних приватних серверів (VPS) та віртуальних серверів з відведеною пам'яттю (VDS) може значно варіюватися в залежності від провайдера послуг хостингу або хмарних сервісів. Зазвичай, процес налаштування VPS або VDS може бути досить швидким та ефективним. Коли користувач замовляє віртуальний сервер, найчастіше він отримує його протягом кількох хвилин або навіть секунд. Деякі провайдери навіть пропонують можливість миттєвого розгортання (instant deployment) для VPS, що означає миттєвий доступ до сервера без зайвих затримок. Цей швидкий процес налаштування дозволяє користувачам швидко розпочати роботу з їхнім віртуальним сервером і використовувати його згідно з власними потребами без значних затримок або часових витрат на очікування..

Перелічені вище фактори роблять VPS та VDS популярними серед бізнесів, розробників та користувачів, які шукають ефективні рішення для хостингу та обслуговування вебсайтів, додатків та інших вебпроектів. На сьогоднішній день лідерами з надання послуги оренди VDS/VPS є [36]:

– Amazon Web Services (AWS);

- Microsoft Azure;
- Google Cloud Platform (GCP);
- DigitalOcean;
- Linode;
- Vultr;
- OVHcloud;
- Hetzner Cloud.

Ці компанії мають різні плани, функціонал, ціни та підтримку, і вибір кращого провайдера може залежати від ваших конкретних потреб та бюджету.

Рекомендації вибору кращого провайдера можуть бути індивідуальними і можуть змінюватися залежно від конкретної ситуації та вимог. Перед вибором провайдера, слід ретельно дослідити їхні можливості, рецензії користувачів, ціни та служби підтримки.

2.3 Фактори впливу на оцінювання ресурсів та витрат для розроблення сервісу моніторингу доступності вебресурсів

Цей етап включає в себе оцінку людських, фінансових, технічних та часових ресурсів, необхідних для успішного завершення проекту. Оцінка витрат і ресурсів є важливим кроком у плануванні та визначенні потенційних витрат на реалізацію даної системи моніторингу.

Людські ресурси у контексті оцінки для розробки системи моніторингу доступності вебресурсів охоплюють аналіз необхідних кваліфікацій та чисельності персоналу для успішного втілення проекту[37].

Визначення необхідних спеціалістів та їх навичок є важливою складовою оцінки. Розробники, інженери з веброботки, тестувальники, фахівці з безпеки, а також технічна підтримка можуть бути включені до складу команди. Не менш важливим є аналіз їхнього досвіду та компетенцій у відповідній області. Крім того, потрібно враховувати можливості залучення зовнішніх партнерів, які

мають необхідний досвід у розробці та підтримці подібних систем, може бути важливим фактором для успішного завершення проекту. Такий аналіз допомагає забезпечити витрати на персонал в межах бюджету та отримати необхідний рівень експертизи для реалізації проекту.

Вартість розроблення системи моніторингу доступності вебресурсів може значно варіюватися в залежності від багатьох факторів, таких як обсяг та складність функціоналу, потреби в спеціалістах, обрані технології, локація розробки, терміни проекту та інші. Наприклад, більш прості системи моніторингу, які орієнтовані на базові перевірки доступності вебсайтів, можуть мати більш доступні витрати. Проте, якщо потрібні додаткові функції, такі як розширені аналітичні можливості, моніторинг безпеки або глибокі технічні аспекти, вартість розробки може збільшитися.

Загальною практикою є проведення детальної оцінки витрат з боку команди розробки або ІТ-фахівців, щоб визначити конкретні витрати на реалізацію такого проекту в конкретних умовах.

Розробка системи моніторингу доступності вебресурсів може коштувати від кількох тисяч до декількох десятків тисяч доларів США, або навіть більше в разі складних або розширених проектів. Вартість визначається наступними факторами.

1. Функціональність системи: чим більше функцій, тим більше розробка може вартувати. Наприклад, базовий моніторинг може бути менш коштовним, а отже, додавання розширених функцій, таких як аналітика, автоматизовані звіти або інтеграція з іншими системами, збільшить ціну.

2. Обсяг робіт: великий обсяг робіт зазвичай означає більше витрат. Це включає в себе розробку, тестування, підтримку, технічну документацію та інші аспекти.

3. Команда та спеціалісти: вартість може змінюватися в залежності від того, чи використовується внутрішній персонал або залучаються зовнішні спеціалісти.

4. Технології: вибір певних технологій чи платформ може впливати на вартість, оскільки деякі технології можуть бути більш витратними у розробці та підтримці.

5. Терміни проекту: швидкий термін виконання зазвичай вимагає більше витрат через прискорену роботу та залучення додаткових ресурсів.

Ці конкретні фактори варіюються від проекту до проекту, тому варто провести докладний аналіз та обговорення з командою розробки, щоб уточнити приблизні витрати для конкретного проекту.

Ризики та резерви у проекті розробки системи моніторингу вебресурсів охоплюють різноманітні аспекти, що можуть вплинути на успішність проекту. Це включає технічні ризики, пов'язані з непередбачуваними проблемами у технічній частині, ризики з ресурсами такі як недостатність персоналу або фінансів. Зовнішні фактори наприклад: зміни вимог або регуляцій та ризики безпеки й конфіденційності, втрата даних або порушення безпеки.

Заходи з ризик-менеджменту включають резерви, які забезпечують додатковий час, бюджет чи ресурси для реагування на непередбачені обставини. Це важливо для зниження впливу негативних ситуацій та ефективного управління будь-якими можливими загрозами проекту.

Висновки до розділу 2

Безумовно, вибір підходу для розробки системи моніторингу доступності вебресурсів є важливим етапом, що базується на декількох факторах, включаючи час розробки, витрати та ефективність. Було визначено та проаналізовано два головних підходи, які найбільше відповідають критеріям швидкості реалізації та витрат на підтримку для власників вебсайтів або технічних працівників, а також розглянуто два популярних підходи розробки моніторингу доступності.

Перший підхід передбачає використання готового рішення для моніторингу доступності вебресурсів від компанії HetrixTools та його інтеграцію з Telegram Bot Api через сервіс автоматизації вебпроцесів Make. Це дозволяє

оперативно отримувати повідомлення про стан доступності вебсайту через зручний месенджер.

Другий підхід базується на використанні готового програмного забезпечення ZennoPoster та його інтеграцію з Telegram Bot API. Це також відмінний спосіб забезпечити моніторинг та отримання повідомлень про стан доступності вебресурсу через популярний месенджер.

Наголошено на тому, що для розгортання сервісу моніторингу доступності вебресурсів необхідно мати серверні ресурси - фізичний сервер або віртуальний сервер в хмарі. Останній варіант дозволяє легко масштабувати серверні можливості та швидко реагувати на зміни в потребах проекту.

Для стабільної та ефективної роботи глобального сервісу моніторингу доступності вебресурсів важливо мати розподілену мережу серверів у різних частинах світу, що забезпечить швидкий доступ для користувачів з різних географічних регіонів та знизить затримки при виконанні перевірок.

Аналізуючи різні технічні та фінансові аспекти, наведено обґрунтування вибору конкретних стратегій, технологій та підходів для розробки системи моніторингу доступності вебресурсів. Ці підходи є потенційно ефективними для контролю доступності вебплатформ та максимізації їх продуктивності.

РОЗДІЛ 3

РОЗРОБЛЕННЯ МОНІТОРИНГУ ДОСТУПНОСТІ ВЕБРЕСУРСІВ ТА АВТОМАТИЗАЦІЯ ВЕБПРОЦЕСІВ

3.1 Розроблення сервісу моніторингу доступності вебресурсів із використанням готових рішень

На початку розробки було прийнято рішення починати з розроблення шаблону бізнес-моделі, до уваги було взято Lean Canvas.

Lean Canvas – це інструмент, який допомагає зрозуміти ключові аспекти бізнесу при створенні стартапу або розвитку нового продукту [38]. Це спрощена версія бізнес-плану, яка дозволяє уявити, протестувати та визначити стратегію розвитку бізнесу на одному аркуші паперу. Lean Canvas включає в себе дев'ять ключових блоків, які охоплюють основні аспекти бізнесу: пропозиція для користувачів, цільова аудиторія, проблема, унікальність пропозиції, канали збуту, доходи, ресурси, вартість та ключові метрики (рис.3.1). Цей інструмент дозволяє зосередитися на головних аспектах бізнесу, спрощує процес виявлення сильних та слабких сторін проекту, допомагає зробити швидке оцінювання його потенціалу та зрозуміти його вартість для клієнтів.



Рисунок 3.1 – Візуалізація бізнес моделі Lean Canvas

Після розроблення бізнес моделі настає час дизайну користувацького досвіду або UX, тобто необхідно розробити послідовність дій користувача.

UX (англ. User Experience) – означає "користувацький досвід" у перекладі з англійської. UX відноситься до способу, яким користувачі сприймають і взаємодіють з продуктом, послугою, вебсайтом, програмою або будь-яким іншим об'єктом [39].

Головна мета UX – забезпечити користувачам приємний, ефективний та задовільний досвід під час використання продукту чи послуги. Для цього використовуються різноманітні методи та підходи, включаючи дослідження користувачів, проектування інтерфейсів, тестування взаємодії тощо [40].

При розробленні нового сервісу з інтерфейсом користувача, важливо приділяти увагу до вже сформованого користувацького досвіду, тобто якщо зробити всі кнопки на сайті круглими а для тексту використовувати не читабельний шрифт, скоріш за все користувачі не оцінять такого перформансу.

Невдале рішення UX дизайну зображено на рис. 3.1

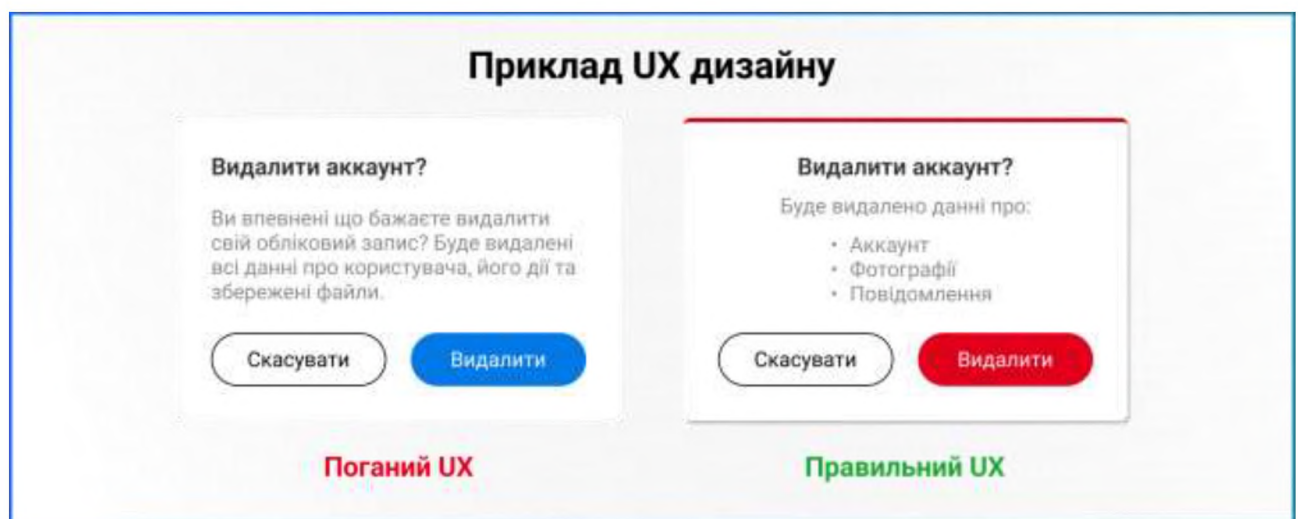


Рисунок 3.1 – Приклад невдалого UX дизайну

UX для моніторингу доступності вебсайтів орієнтується на створення зручного та ефективного інструменту, який допомагає спеціалістам з доступності вебсайтів виявляти, аналізувати та виправляти проблеми, що перешкоджають користувачам з обмеженими можливостями.

Основні аспекти UX для моніторингу доступності вебсайтів включають:

1. Простий та інтуїтивний інтерфейс: інструмент для моніторингу повинен мати зрозумілий та легкий у використанні інтерфейс, що дозволить користувачам швидко зорієнтуватися в функціях та можливостях інструменту без додаткових навчань.

2. Зручність у використанні функцій аналізу: інструмент повинен надавати можливості проведення різноманітних аналізів доступності вебсайтів, таких як перевірка на відповідність стандартам WCAG (Web Content Accessibility Guidelines), аналіз коду та інші. Функції аналізу повинні бути чітко структурованими та легкими у використанні [41].

3. Гнучкість налаштувань та зручність фільтрації результатів: користувачам може бути корисно мати можливість налаштовувати параметри аналізу та фільтрувати результати, щоб концентруватися на конкретних аспектах доступності або проблемах, що потребують негайного виправлення.

4. Повідомлення та звіти з результатами аналізу: інструмент повинен надавати зрозумілі та докладні звіти про виявлені проблеми доступності разом з рекомендаціями щодо їх виправлення. Це допоможе командам розробників та дизайнерів швидко реагувати на виявлені проблеми.

5. Підтримка та навчання користувачів: інструмент повинен мати доступну документацію, онлайн-підтримку та, можливо, навчальні матеріали або тренувальні ресурси для користувачів з метою максимального використання його можливостей.

6. Адаптивність та швидкість роботи: важливо, щоб інструмент працював швидко та був доступний на різних пристроях і платформах, враховуючи потреби користувачів.

Розробка інструменту для моніторингу доступності вебсайтів з фокусом на UX допомагає забезпечити, що процес виявлення та виправлення проблем доступності стає більш ефективним і зручним для спеціалістів, які відповідають за цей аспект розробки вебсайтів. Наступним після UX дизайну настає час візуальної складової а саме UI дизайну.

UI (англ. User Interface) дизайн – означає «інтерфейс користувача» у перекладі з англійської мови. UI відноситься до всіх елементів та засобів, які користувач використовує для взаємодії з програмним забезпеченням, пристроями або вебсайтами [42].

Основна мета UI – забезпечити користувачам зручну, ефективну та приємну взаємодію з продуктом. Це охоплює елементи дизайну, такі як кнопки, меню, поля введення, графіка, шрифти, кольори, анімація та будь-які інші візуальні та інтерактивні елементи, що використовуються для спілкування користувача з продуктом.

Гарний UI дозволяє користувачам легко розуміти, як працювати з програмним забезпеченням або пристроєм, дозволяє їм швидко знаходити необхідну інформацію та виконувати бажані дії без зайвих труднощів чи заплутування. Порівняння UX та UI дизайну зображено на рис. 3.2.

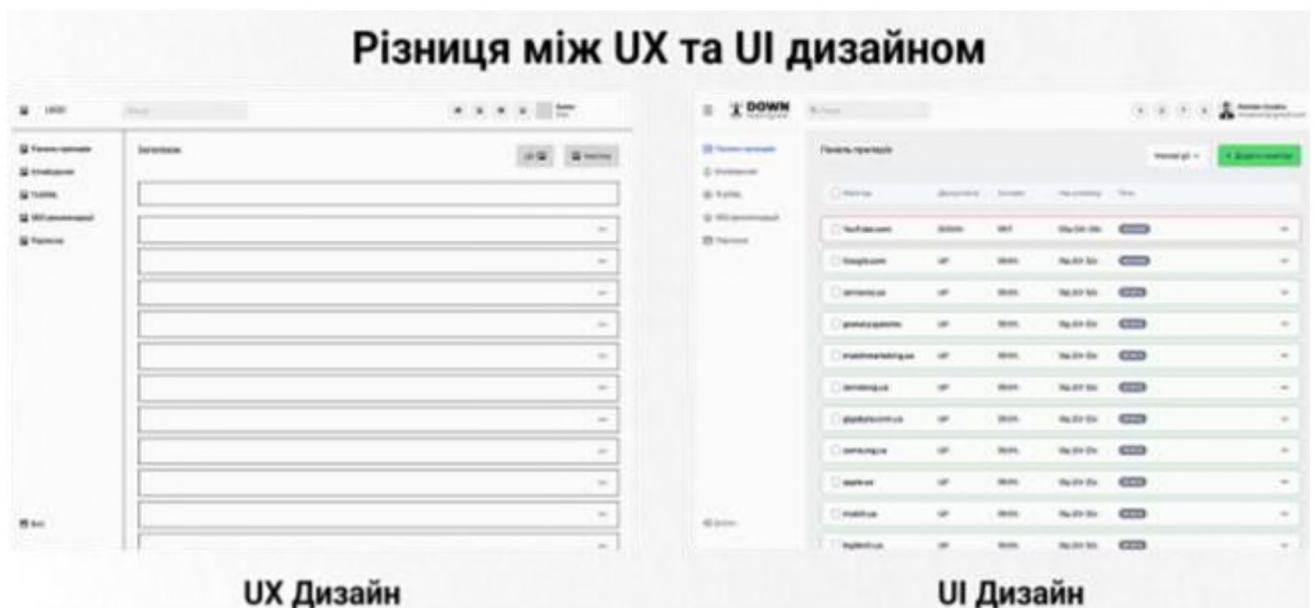


Рисунок 3.2 – Порівняння UX та UI дизайну

Разом, UX та UI впливають на зручність, ефективність та задоволення користувача від взаємодії з продуктом. Ці дві області співпрацюють для створення користувацького досвіду, що є зрозумілим, зручним та приємним для використання.

Розробивши дизайн інтерфейсу застосунку настає час для розроблення інструментів сервісу моніторингу доступності.

У даній роботі запропоновано використовувати перший метод розробки а саме: скористатися готовим рішенням для моніторингу вебсайту та модифікувати систему отримання сповіщень.

Спершу необхідно знайти надійний інструмент або сервіс котрий буде відповідати усім вимогам моніторингу. На основі попередньої аналітики було обрано сервіс HetrixTools, оскільки він має безліч серверів на території Європи, що чудово підходить для перевірки доступності вебресурсів на території України. Ще однією перевагою є те, що сервіс має безкоштовний тариф, якого цілком вистачить для компанії з одним сайтом та декількома серверами.

Першим кроком виконується реєстрація на сайті сервісу HetrixTools [--], після чого у користувача автоматично відкривається «Панель приборів». У цій панелі можна додати свій ресурс, який необхідно відслідковувати, натиснувши на кнопку «Add Monitor». Використання кнопки ресурсу показано в додатку В.

Після натискання на кнопку, відкриється меню «Додавання монітору». На першому кроці сервіс запропонує користувачу обрати один із варіантів моніторингу, серед варіантів:

1. Website monitor
2. Ping/Service monitor
3. SMTP monitor
4. Heartbeat monitor

Оскільки стоїть задача моніторити доступність вебсайту, то слід обрати пункт «Website monitor»

Далі необхідно заповнити всі необхідні поля для налаштування критеріїв (показників) моніторингу:

1. Monitor name: назва монітору;
2. Website link: посилання на вебсайт;
3. Category: категорія;
4. Monitor from: вибір країни для сервера моніторингу;

5. Accepted HTTP Codes: прийняті HTTP коди;
6. Timeout: час очікування;
7. Maximum Redirects: максимальна кількість переадресацій;
8. Checkup Frequency: частота перевірок;
9. Number of Tries: кількість спроб;
10. Number of Triggering Locations: кількість активних місць моніторингу;
11. Alert on After X minutes Of Downtime: повідомлення після X хвилин відсутності;
12. Verify SSL Certificate: перевірка SSL сертифіката.

Наступник кроком вибираємо спосіб сповіщення, WebHook. Після налаштування слід натиснути кнопку «Додати монітор», щоб зберегти налаштування та додати монітор до списку моніторів.

Після додавання сайту на моніторинг слід перейти до другого кроку, та підключити hetrixtools.com до make.com.

Спочатку, потрібно увійти у свій обліковий запис на make.com або створити його та перейти до списку сценаріїв.

Далі, потрібно обрати або створити новий сценарій для редагування.

Після цього, додається модуль WebHook, який використовується для прийому HTTP запитів.

Далі, отримується URL-адреса WebHook від make.com для подальшого використання його в методах отримання сповіщень в hetrixtools.com.

Потім, додається модуль Telegram та підключається обліковий запис Telegram до [Make.com](https://make.com).

Наступним кроком, необхідно отримати ключ API для бота в Telegram. Спочатку потрібно знайти [@BotFather](https://t.me/BotFather) в месенджері та розпочати діалог. Після початку спілкування з ним, користувач повинен натиснути кнопку «Start», щоб почати взаємодію. Далі, слід відправити команду `/newbot`, щоб створити нового бота.

Після цього, [@BotFather](https://t.me/BotFather) запропонує вибрати ім'я для бота. Необхідно надати боту унікальне ім'я. Наступним кроком буде вказання

користувальницького імені для бота, яке повинно закінчуватися на "bot". Наприклад, «example_bot».

Після успішного створення бота, @BotFather надішле повідомлення з ключем API для бота. Цей ключ забезпечує доступ до API Telegram для доступу до бота, і його потрібно зберегти його у безпечному місці. Важливо не передавати цей ключ третім особам, оскільки він надає повний доступ до вашого бота в Telegram.

Далі, налаштовується модуль Telegram для відправлення повідомлень у бота. Вказується текст повідомлення та необхідні параметри для відправлення через Telegram BOT, включаючи ключ API з попереднього кроку.

Після цього, використовуються отримані дані з WebHook (наприклад, JSON або параметри) для налаштування контексту повідомлення в модулі Telegram.

На завершення, зберігаються налаштування та активується сценарій в Make.com.

Ці кроки дозволять приймати дані через WebHook та відправляти вказані повідомлення в Telegram BOT через Make.com [43]. Процес створення сценарію в make.com описано на рис. 3.3.



Рисунок 3.3 – Процес створення сценарію в Make.com

Після цих дій власник сайту отримує повністю робочі моніторинг доступності, котрий, у разі збою, відправляє повідомлення у зручний месенджер, а час виявлення недоступності ресурсу скоротився до лічених хвилин. В залежності від того, як налаштований сайт, ця система моніторингу може бути повністю безкоштовною.

3.2 Розроблення сервісу моніторингу доступності вебресурсів за допомогою автоматизації вебпроцесів

Розроблення сервісу моніторингу доступності вебресурсів за допомогою автоматизації вебпроцесів представляє собою важливий етап у високотехнологічному світі інтернет-послуг. Проект створення такого сервісу вимагає великого рівня уваги до деталей, систематичного підходу та використання передових технологій для забезпечення надійного та ефективного моніторингу вебресурсів.

У цьому контексті розробка програмних засобів для автоматизації вебпроцесів стає ключовим аспектом, оскільки вона спрямована на постійний та систематичний контроль за доступністю та працездатністю вебсайтів. Використання автоматизованих інструментів дозволяє здійснювати швидке виявлення та реагування на можливі проблеми, що виникають у функціонуванні вебресурсів, що, в свою чергу, сприяє підвищенню їхньої ефективності та доступності для користувачів.

Розробка такого сервісу вимагає комплексного підходу, починаючи від аналізу потреб користувачів та виявлення основних вимог до функціональності системи, й до реалізації інструментів моніторингу, що забезпечують надійність та швидкість виявлення проблем. Такий сервіс має на меті забезпечення безперервної доступності вебресурсів та вчасного реагування на будь-які неполадки, тим самим покращуючи користувацький досвід і забезпечуючи стабільність функціонування інтернет-платформ.

За для автоматизації веб процесу в представлений роботі було використане програмне забезпечення ZennoPoster, з ліцензією «lite».

ZennoPoster – це програмний продукт, який є універсальним засобом для автоматизації вебпроцесів. Цей інструмент дозволяє створювати скрипти для виконання різноманітних завдань в інтернеті без необхідності програмування. ZennoPoster використовується для автоматизації дій у браузері: від взаємодії з вебсторінками до заповнення форм, навігації по сайтах, збирання даних, роботи з API і багатьох інших завдань, які потребують роботи в мережі.

Цей інструмент має інтуїтивний інтерфейс (додаток Г), що дозволяє користувачам без спеціальних знань у програмуванні створювати скрипти за допомогою графічного інтерфейсу. ZennoPoster дозволяє автоматизувати багато видів робіт у веб, що робить його корисним для великої кількості завдань у сфері маркетингу, аналітики, реклами, SEO та багатьох інших галузях. Робота в ZP нагадує модульну конструкцію, кожен модуль відповідає за якусь дію. Опишемо функції основних модулів ZP.

1. Емуляція дій користувача. Програма здатна моделювати поведінку користувача в Інтернеті. ZennoPoster дозволяє створювати імітацію кліків мишею, натискання клавіш, введення тексту, перегляду вебсторінок, заповнення форм та інші типові дії, які зазвичай виконує користувач на вебсайтах. Ця функція дозволяє автоматизувати рутинні завдання, які зазвичай виконуються в ручному режимі, шляхом створення скриптів, які відтворюють та автоматизують дії користувача. Завдяки емуляції дій, користувач може створювати автоматизовані робочі процеси, які виконуються відповідно до заданих властивостей і параметрів, що значно полегшує виконання рутинних завдань в мережі Інтернет. Парсинг даних: Збір, обробка та аналіз інформації з вебсторінок, включаючи вилучення тексту, зображень, таблиць, URL-адрес і багато іншого.

2. Робота з динамічним вебконтентом: можливість взаємодіяти з вебсторінками, які містять елементи, що змінюються або оновлюються під час завантаження або після взаємодії з користувачем. Це включає в себе можливість

взаємодії з вебелементами, такими як кнопки, поля введення, списки, таблиці та інші, що змінюються динамічно через виконання JavaScript або AJAX-запитів. ZennoPoster має інструменти для взаємодії з такими елементами на вебсторінках, дозволяючи автоматизовано виконувати дії з динамічним вебконтентом, наприклад, заповнення полів введення, натискання кнопок, отримання або оновлення інформації на сторінці. Це дозволяє створювати скрипти, які можуть працювати з вебелементами, які змінюються або оновлюються під час виконання завдання, що робить робочі процеси більш гнучкими та ефективними.

3. Робота з API та базами даних: взаємодія з різними API для обміну даними між програмами та можливість роботи з різними базами даних.

4. Робота з куки і проксі: Можливість керування куки-файлами для зберігання інформації та проксі-серверами для зміни IP-адреси. Також дозволяє забезпечити ефективну взаємодію з вебсайтами під час автоматизації процесів. Куки використовуються для зберігання даних сесії, таких як інформація про авторизацію, персоналізовані налаштування тощо. ZennoPoster надає зручний інтерфейс для роботи з куки: читання, запис, видалення. Проксі-сервери дозволяють змінювати IP-адресу, що корисно для анонімності чи отримання доступу до обмежених ресурсів. ZennoPoster дає можливість використовувати проксі для надання додаткової анонімності під час взаємодії з вебсайтами.

5. Створення складних автоматизованих робочих процесів: відбувається завдяки використанню блок-схемного режиму, що дозволяє послідовно об'єднувати дії в єдину послідовність. У цьому контексті користувач має змогу створювати складні алгоритми, використовуючи різні блоки: робота з елементами сторінок, умови, цикли, виконання JavaScript коду тощо. Такий підхід дозволяє створювати досить складні програми, які автоматизують різноманітні завдання в мережі.

6. Взаємодія з іншими додатками і сервісами: Можливість інтеграції з іншими програмами та сервісами через API для обміну даними та виконання спільних завдань. На початку треба додати модуль «Таби» та вибрати

налаштування «Перехід на сторінку»: ця дія дозволить відкрити внутрішній браузер ZP для взаємодії з вебсайтом.

Наступним кроком необхідно вказати в налаштуваннях модуля посилання на сайт який необхідно тестувати на доступність. Для прикладу сайт університету «<https://www.pdau.edu.ua/>».

Одним з найлегших варіантів відстеження доступності за допомогою автоматизації є прив'язка до доступності елемента на сторінці вебсайту. Тобто це може бути текстовий блок, зображення, елемент декору та інші елементи які є на сторінці. У конкретному випадку було вирішено прив'язуватися до доступності надпису `<h1>Полтавський державний аграрний університет</h1>`, що є заголовком першого рівня вебсторінки.

Таким чином, необхідно здійснити парсинг заголовку сторінки й порівнювати його з заголовком, який необхідно отримати. Якщо після парсингу заголовки відрізняються, це означає збій сайту або недоступність сторінки.

Парсинг - це процес аналізу тексту або даних для виділення конкретної інформації або структури з них. У вебконтексті парсинг як дія використовується для вилучення корисної інформації з вебсторінок або документів у форматі HTML, XML, JSON або інших (рис. 3.4).

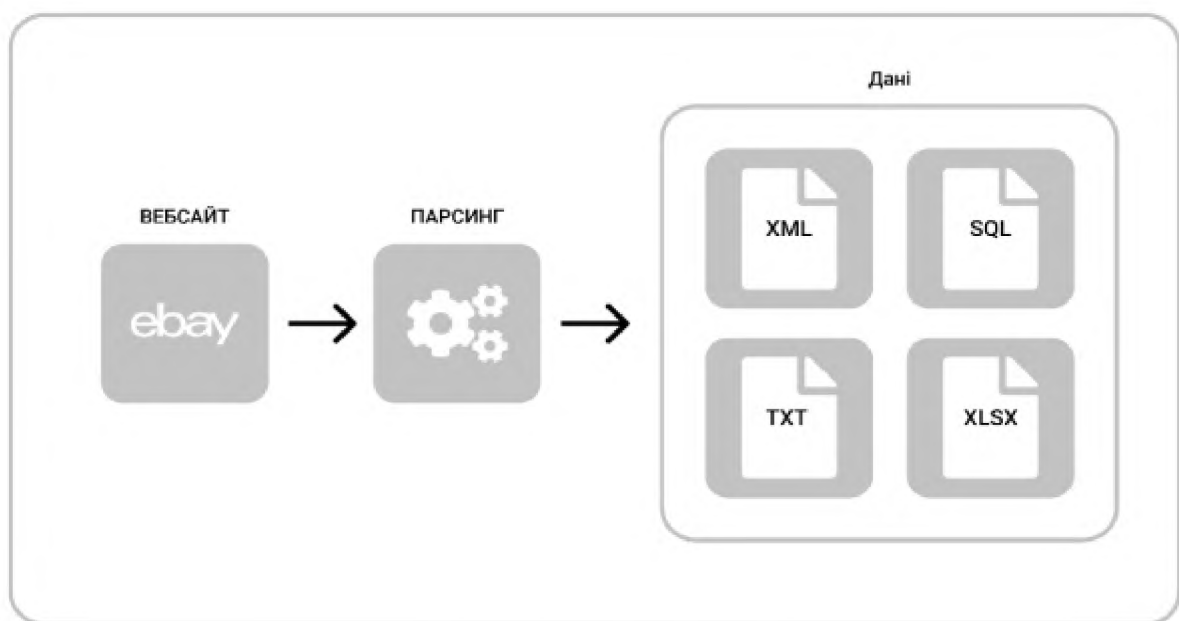


Рисунок 3.4 – Візуалізація роботи парсингу

XML – це розширювана мова розмітки, що використовується для подання та обміну даними в структурованому форматі. Він є текстовим форматом, який використовується для опису даних у вигляді розмічених тегів, аналогічно до HTML, але з більшою гнучкістю та можливістю створення власних тегів та структур даних. XML використовується для обміну даними між програмами та системами, а також для зберігання даних в структурованому форматі. Він часто застосовується в веброзробці, базах даних, конфігураційних файлах та інших областях програмування.

Під час парсингу програми або скрипти переглядають вихідний код вебсторінки, аналізують його структуру та вилучають необхідні дані за допомогою різноманітних методів, таких як регулярні вирази, бібліотеки парсингу, синтаксичний аналіз та інші алгоритми.

Парсинг дозволяє отримати конкретні дані (такі як заголовки, текст, URL-адреси, таблиці, зображення тощо) з великої кількості інформації, що дозволяє використовувати ці дані для подальшого аналізу, збору статистики, автоматизації процесів та багато іншого.

Наступним кроком є додавання модулю «Дані». Цей модуль відповідає за парсинг інформації з веб сайту за допомогою пошуку контенту в об'єктній моделі сайту DOM (від англ. Document Object Model).

DOM – це стандартна модель програмування та представлення структури документів у веброзробці. Вона використовується для опису вебсторінок, їхньої структури та інформації у вигляді дерева об'єктів. DOM представляє структуру HTML-документа у вигляді дерева, де кожен елемент сторінки (текст, зображення, посилання, форма і т.д.) є окремим вузлом цього дерева. Він надає програмам та скриптам можливість доступу до цих елементів, їх зміни та взаємодії з ними. Завдяки архітектурі DOM розробники можуть маніпулювати вмістом вебсторінки за допомогою JavaScript або інших мов програмування, змінюючи структуру сторінки, додавати, видаляти або змінювати елементи, обробляти події користувача та багато іншого. При цьому існують засоби управління властивостями окремих елементів завдяки наслідуванню.

Задля парсингу заголовку сторінки був написаний регулярний вираз Regex, $(?<=<h1>).*(?=</h1>)$, котрий відповідає за пошук на сторінці тегу «h1» та контентом, який в нього «загорнутий».

Regex – це послідовність символів, яка утворює шаблон «пошуку» для пошуку та відповідності тексту в рядках. Це потужний інструмент для роботи з рядками, що дозволяє шукати, вилучати, аналізувати та замінювати текст на основі певних правил чи шаблонів (рис 3.5).



Рисунок 3.5 – Візуалізація регулярного виразу

За допомогою регулярних виразів можна виконувати наступні дії.

1. Пошук певного тексту в рядках: пошук текстових відповідей за допомогою регулярних виразів, знаходження певних текстових патернів або шаблонів у рядку чи текстовому документі. Регулярні вирази визначають вимоги до тексту, за яким потрібно проводити пошук, і можуть включати символи, які відповідають конкретним символам або шаблонам. Наприклад, шукаючи адресу електронної пошти у тексті, можна скористатися регулярним виразом, який визначає патерн електронної адреси.

2. Заміна тексту: заміна тексту за допомогою регулярних виразів полягає у зміні певних частин тексту на інші символи або фрази відповідно до заданого шаблону. Це може включати заміну конкретних слів, фраз або символічних послідовностей на інший текст. Регулярний вираз визначає патерн, який треба

замінити, і може бути використаний для пошуку конкретних фрагментів тексту. Потім визначається, на що ці фрагменти будуть замінені. Прикладом може бути заміна всіх входжень слова "кіт" на слово "собака" у тексті. Регулярний вираз шукає слово "кіт", а потім замінює його на "собака", перетворюючи всі входження цього слова в тексті.

3. Парсинг: використання шаблонів або правил для пошуку, виділення та вилучення певних частин інформації зі строки або тексту. Регулярні вирази визначають шаблон, за яким відбувається пошук та зчитування даних. Після визначення відповідного шаблону програма може виконати пошук певних паттернів або тексту в масиві даних і витягти інформацію, що відповідає цьому шаблону.

4. Валідація даних: за допомогою регулярних виразів можна перевіряти введенні або отримані дані на відповідність певному шаблону або формату. Регулярні вирази використовуються для визначення правил перевірки, де вони фільтрують, порівнюють або перевіряють введені дані на наявність відповідних символів, кількості або інших параметрів. Наприклад, валідація електронної пошти може використовувати регулярний вираз для перевірки, чи введена адреса відповідає формату електронної пошти (наприклад, "ім'я@домен.com"). Якщо введені дані не відповідають заданому шаблону, вони можуть бути відхилені або виправлені користувачем для забезпечення правильності та цілісності інформації.

5. Фільтрація: фільтрація з використанням регулярних виразів – це процес відбору певних даних з текстового чи символьного потоку, які відповідають певному шаблону. Регулярні вирази визначають цей шаблон, відображаючи те, що потрібно знайти або вилучити з вихідних даних. Наприклад, якщо є текст із різними рядками, але ми хочемо знайти тільки ті рядки, які містять певне ключове слово або певний формат, ми можемо скористатися регулярним виразом, щоб вилучити тільки потрібну інформацію. Фільтрація з регулярними виразами дозволяє ефективно та точно виділяти певні дані для подальшого використання чи обробки.

Після успішної відповіді сервера користувач отримує свій «UserID» у вигляді комбінації цифр. UserID та ChatID в Telegram – це унікальні ідентифікатори, які використовуються для ідентифікації користувачів та чатів у цій платформі. Кожен користувач має свій унікальний UserID, який дозволяє виконувати такі операції, як надсилення повідомлень, додавання до груп тощо.

ChatID – це унікальний ідентифікатор кожного чату або групи в Telegram, що використовується для ідентифікації конкретних спілкувань чи групових обговорень у месенджері. Обидва ідентифікатори є важливими для взаємодії з ботами, відправки повідомлень, а також для здійснення різноманітних дій у Telegram. Далі необхідно сформувати повідомлення, яке бот буде надсилати у випадку недоступності ресурсу. Наприклад: Сервіс «назва сервісу» недоступний. Час виявлення «поточний час».

Для надсилення цього повідомлення необхідно сформувати також HTTP запит. На основі Telegram BOT API (рис. 3.7).



Рисунок 3.7 – Приклад сповіщення в Telegram за допомогою HTTP запиту

Після успішної конфігурації послідовності дій бота настає час розвернути його на серверному обладнанні за для постійної та безперебійної роботи 24

години, 7 днів на тиждень. Для цієї задачі було вибрано Українського постачальника VPS а саме компанію Ukraine.com.ua адже їх сервери знаходять безпосередньо на території України що значно покращить показник «RRT». Ще одна з причин вибору саме цього сервісу це доступна ціна.

Після оренди серверу була обрана операційна система сімейства Windows Server 2016, адже ZP не підтримує жодні інші платформи.

Після того як все готово, залишається розвернути шаблон ZP та налаштувати автозапуск шаблону, кожні 5 хв. Приклад готового сценарію зображено на рис. 3.8.

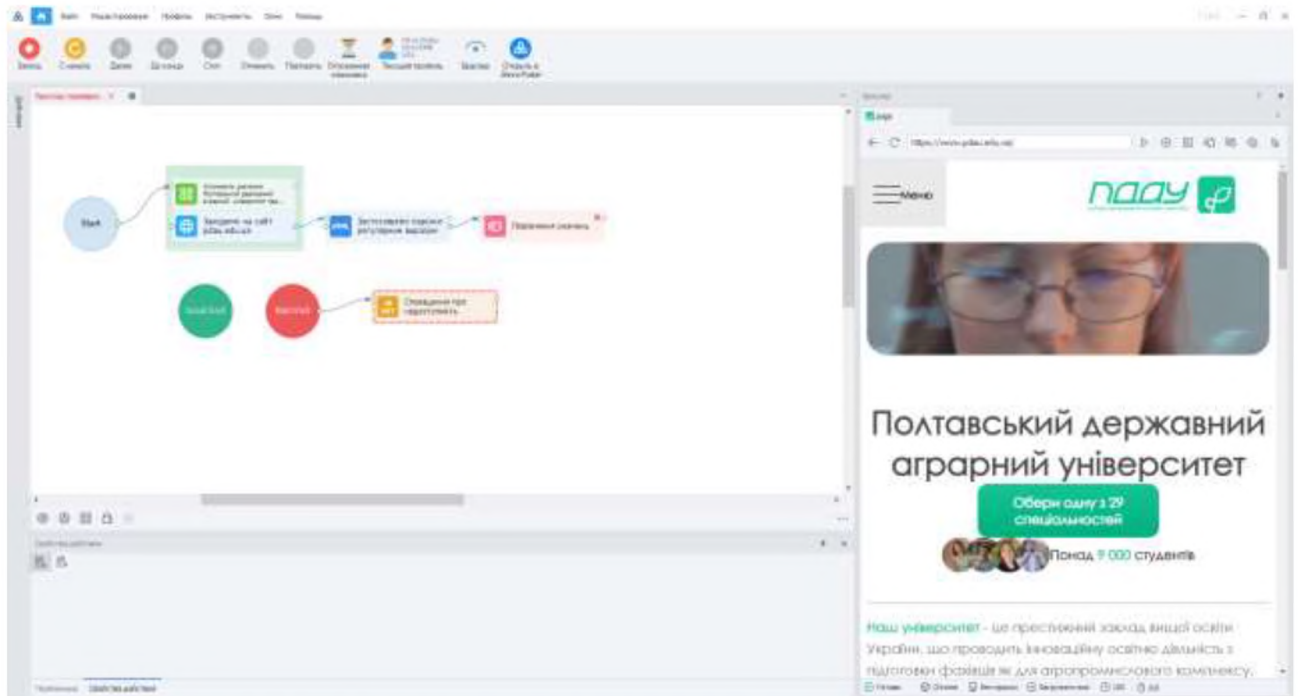


Рисунок 3.8 – Приклад готового сценарію в застосунку ZennoPoster

Серед мінусів цього підходу можна вважати середню точність моніторингу, умовну зручність, та високу ціну.

3.3 Техніко-економічне обґрунтування ефективності застосування моніторингів доступності веб ресурсів

За даними інтернет видання Forbes Ukraine, інтернет магазин Rozetka.ua за 2022 р. отримала виторгу на суму 18,4 млрд. грн. Тобто інтернет-магазин заробляє приблизно 38000 грн. за хвилину. Таким чином, якщо на сайті відбудеться збій або недоступність певних сторінок, система моніторингу на протязі хвилини надішле повідомлення про недоступність або застосує протокол відновлення працездатності сайту (наприклад, презавантажить сервер), що призведе до відновлення працездатності ресурсу на протязі п'яти хвилин. Компанія втратить всього 175 000 грн.

Зазвичай, у великих компаній є співробітник, який відповідає за доступність ресурсів компанії або це декілька системних адміністраторів чи інших технічних працівників, які зобов'язані перевіряти справність систем та доступність вебсайтів.

Для прикладу наведена табл. 3.1, в якій внесено усереднені дані, за допомогою яких наочно відображено вигоду застосування сервісів моніторингу доступності при порівнянні співвідношення витрат на працівника і підписку на моніторинговий сервіс.

Таблиця 3.1 – Співвідношенні витрат на працівника і підписку на моніторинговий сервіс

Виконавець	Заробітна плата, грн	ЄСВ, грн	ПДФО, грн	Підписка, грн
Найманий працівник/міс	16000	1100	2800	0
Система моніторингу/міс	0	0	0	300

Якщо дивитися на сервіс моніторингу доступності вебсайту зі сторони бізнесу, тобто заробляти надаючи послугу моніторингу варто одразу ж зрозуміти вартість розроблення та утримання працездатності сервісу (табл. 3.2).

Таблиця 3.2 – Вартість розробки сервісу моніторингу на рік

Найменування	Кількість, одиниць	Ціна за одиницю, грн	Вартість за рік, грн
Оренда трьох серверів на платформі Digital Ocean/1 міс	3	148	5328
Ліцензія на Windows Server	3	18 537	55611
Ліцензія на програмний комплекс ZennoPoster	3	1087	3261
Зарплатня тех. Працівникам та розробникам/міс	2	25000	600000
Рекламні послуги			100000
Всього			764 200

Зрозуміло, що це наближений обрахунок витрат. Варто також порахувати дохід сервісу моніторингу на 1000 користувачів (табл 3.3.). враховуючи середню ціну та умови підписки на популярних сервісах моніторингу доступності.

Таблиця 3.3 – Дохід на 1000 користувачів при різних тарифних планах

Тарифні плани	Ціна підписки на одного користувач, грн	Дохід на 1000 користувачів, грн
Basic (можливість додати 1 ресурс для моніторингу)	300	300000
Standart (можливість додати 10 ресурсів для моніторингу)	2400	2400000
Agency (можливість додати 50 ресурсів для моніторингу)	12000	12000000

Згідно з інформацією на сайті uptimerobot.com, їх сервісом користуються приблизно 2100000 користувачів (рис. 3.9), а найдешевший тарифний план коштує 259 грн, що в місяць приносить компанії 543 900 000 грн.

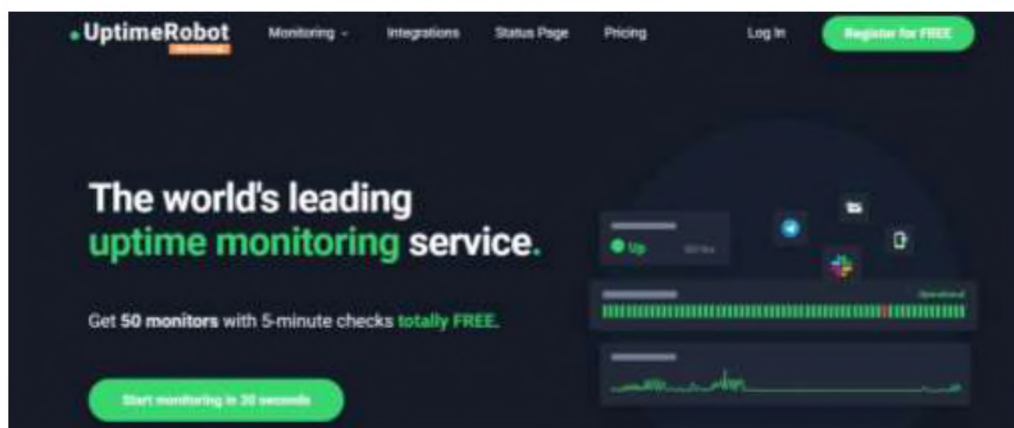


Рисунок 3.9 – Головний сайт UptimeRobot

Результати оцінювання показують, що впровадження моніторингових рішень дозволить ефективно контролювати та уникати перерв у доступності вебресурсів.

Цей розділ ретельно розглянув витрати на розробку та підтримку моніторингової системи, порівнявши їх з очікуваними користями. Обґрунтування ефективності включає аналіз ризиків, вартості інфраструктури, інтеграції технічних рішень та потенційних вигод для бізнесу.

Висновок вказує на те, що впровадження системи моніторингу доступності вебресурсів вигідно з економічної та технічної точок зору. Ця система може підвищити ефективність бізнесу, знизити ризики перерв у роботі сайту та забезпечити кращий контроль за доступністю вебресурсу для користувачів.

Висновки до розділу 3

У третьому розділі роботи вивчено та проаналізовано два ключових методи розробки моніторингу доступності вебресурсів. Перший метод базувався на використанні готових рішень, зокрема, на платформі hetrixtools.com, яка була адаптована за допомогою шаблону бізнес-моделі Lean Canvas. Цей підхід передбачав докладний розгляд концепцій UX/UI дизайну та інтеграції із сервісом автоматизації вебпроцесів Make від розробника "integromat".

Пошагово розглянуто процес додавання та конфігурування ресурсу на hetrixtools.com, після чого реалізовано послідовну інтеграцію з Telegram за допомогою сервісу Make.com. Важливий етап цього методу - докладна настройка та інтеграція, що дозволяє автоматизувати та забезпечити швидку реакцію на проблеми з доступністю вебресурсів.

Другий метод пропонував розробку сервісу моніторингу доступності з використанням програмного забезпечення автоматизації ZennoPoster. У рамках цього методу було розроблено шаблон послідовності дій для перевірки доступності сайту pdau.edu.ua та проведено інтеграцію з Telegram Bot Api для отримання сповіщень у месенджер.

Останній етап дослідження включав техніко-економічне обґрунтування ефективності застосування моніторингів доступності вебресурсів. Цей аспект виявився ключовим, оскільки він дозволив оцінити потенційні переваги та витрати на розглянуті методи моніторингу. Чітко виокремлені етапи та методи розробки дозволяють зробити обґрунтований вибір між різними підходами до моніторингу доступності вебресурсів, враховуючи якість, ефективність та фінансові аспекти.

ВИСНОВКИ

У світі, де вебсайти постійно зростають за кількістю, розробка сервісів моніторингу доступності вебресурсів стала надзвичайно важливою. Вебмоніторинг виник на етапі розцвіту Інтернету і став вирішальним для контролю функціонування вебсайтів. Початково ці моніторинги обмежувалися лише перевіркою доступності, але сьогодні вони пропонують широкий спектр інструментів для забезпечення якості та доступності вебресурсів та додатків. Нові сервіси моніторингу можуть прогнозувати проблеми до їх виникнення, що є значним покращенням.

Зараз практично будь-який власник вебсайту може легко скористатися моніторингом для ефективного контролю доступності своїх ресурсів. Багато компаній обирають цей підхід через економію часу та грошей, які раніше були витрачені на розробку власних інструментів моніторингу. Однак деякі організації можуть мати інші мотиви для створення власних систем через різноманітні фактори, такі як локалізація серверів.

Особлива увага приділяється методам сповіщення про недоступність вебресурсу. Вибір методу залежить від вимог користувача та часових обмежень, але головна мета – оперативно сповіщати про проблеми доступності вебсайту.

Важливо визначити параметри моніторингу, які відображають конфігурацію системи та методи перевірки доступності вебресурсів. Ці параметри допомагають системі реагувати на проблеми доступності відповідно до вимог користувача.

Було розглянуто важливі концептуальні аспекти моніторингу та підкреслено їх вплив на управління вебпроцесами. Розглянуті аспекти дозволяють визначити ключові переваги використання моніторингу доступності вебресурсів для підвищення ефективності бізнесу та користувацького досвіду.

Аналізуючи технічні та фінансові характеристики, було обґрунтовано вибір стратегій, технологій та підходів для розробки системи моніторингу доступності вебресурсів. Ці підходи мають потенціал стати ефективними

інструментами для контролю доступності вебплатформ та підвищення їх продуктивності.

Досліджено два ключові методи розробки моніторингу доступності вебресурсів. Перший метод базувався на використанні готових рішень, таких як hetrixtools.com, з інтеграцією з Telegram через сервіс автоматизації Make. Цей підхід передбачав докладний розгляд концепцій UX/UI дизайну та інтеграцію із сервісом автоматизації вебпроцесів.

Другий метод пропонував розробку сервісу моніторингу доступності з використанням програмного забезпечення ZennoPoster. У рамках цього методу було розроблено шаблон послідовності дій для перевірки доступності вебресурсу та проведено інтеграцію з Telegram для отримання сповіщень.

Техніко-економічне обґрунтування ефективності застосування моніторингу доступності вебресурсів виявилось ключовим етапом, оскільки дозволило оцінити переваги та витрати на обидва методи моніторингу. Чітко виокремлені етапи та методи розробки дозволяють зробити обґрунтований вибір між різними підходами до моніторингу доступності вебресурсів, враховуючи якість, ефективність та фінансові аспекти.