МІНІСТЕРСТВО ОСВІТИ ТА НАУКИ УКРАЇНИ

ЧЕРКАСЬКИЙ ДЕРЖАВНИЙ БІЗНЕС-КОЛЕДЖ

***Кафедра комп’ютерної інженерії та інформаційних технологій***

Відділення інженерії програмного забезпечення

**Звіт з ПР02**

з курсу «Архітектура програмного забезпечення»

Чернишова Романа Святославовича,Хохлова Костянтина Дмитровича,Шемшура Сергія Олексійовича,Пальонного Михайла Юрійовича

*ПІБ студентів*

студентів групи **1П-21**

Викладач Марченко С. В.

Черкаси-2023

**ПРАКТИЧНА РОБОТА №2**

**Атрибути якості та архітектурно значущі вимоги**

**Система оцінювання**

|  |  |  |
| --- | --- | --- |
| **№** | **Тема** | **К-ть балів** |
| 1. | Завдання 1. Quality Attribute Workshop | 4 |
|  | **За практичну** | **4** |
| 2. | ІНДЗ-1. Додаток А | 2 |
| 3. | ІНДЗ-2. Quality Attribute Workshop для проєкту | 4 |
| 4. | ІНДЗ-3. UML/C4-модель проєкту | 4 |
|  | **Всього** | **14** |

1. ***(Quality Attribute Workshop)*** Виконайте у мінігрупах (орієнтовно 3-4 особи) АТАМ-аналіз для одного з запропонованих доменів на основі курсу «Software Architecture & System Design: Practical Case Studies»:

|  |  |  |  |
| --- | --- | --- | --- |
| **Варіант** | **Домен** | **ПІБ студентів** | **Атрибути якості** |
| 1 | Highly Scalable Image Sharing Social Media Platform | 1П-21б: Чернишов,Хохлов,Шемшур,Пальонний | Reliability, Security, Scalability, Performance |
|  |  |
| 2П-20: Кучерук, Ігнатенко, Ганжуга, Березенко |  |
| 2 | Video-On-Demand (VOD) Streaming Service | 1П-21а: Гончар, Голець, Ванін |  |
| 2П-21б: Ткаченки, Вакуленко, Компанієць |  |
|  |  |
| 3 | Highly Scalable, Real-Time Instant Messaging Service | 1П-20: Нестеренко, Панченко, Анголюк. |  |
| 1П-21а: Гончаренко, Заєць, Шаповал |  |
| 2П-21б: Нечко, Очеретяний, Нех, Шиповалов |  |
| 2П-20: Гончарова, Дворяківська, Короп, Чухало |  |
|  |  | 1П-21б: Чіпенко, Цьома, Устименко |  |
| 4 | Typeahead Autocomplete for a Search Engine | 1П-20, 3П-21: Володін, Воробйова, Дяченко, Ергардт. |  |
|  |  |
|  |  |
| 5 | Scalable Ride Sharing Service | 2П-21б: |  |
| 2П-20: |  |
|  |  |

Обґрунтуйте вибір ключових атрибутів якості для аналізу та підготуйте доповідь щодо оцінювання їх впровадження, враховуючи сценарії атрибутів якості та загальне utility-дерево.

**Проект «ImgHub»**

ImgHub - це проект, спрямований на надання користувачам доступу до фотографій, до яких вони мають індивідуальний інтерес. Наприклад, припустимо, що у вас є веб-сайт із фотографічним портфоліо, де розміщено значну кількість фотографій. Користувачам може сподобатися лише певна фотографія, і їм потрібно легко знайти та отримати доступ до неї серед великого обсягу контенту. Основна проблема, яку вирішує ImgHub, полягає в усуненні цієї перешкоди та наданні швидкого та зручного доступу до фотографій, які цікавлять користувачів.

Проект ImgHub зосереджений на забезпеченні чотирьох важливих атрибутів якості, а саме Performance, Security, Reliability та Scalability, але особливу увагу слід приділити атрибуту Performance. Цей атрибут грає вирішальну роль у проекті і має надзвичайну вагу з наступних причин.

**Атрибут якості "Performance" у проекті "ImgHub":**

Атрибут якості **"Performance"** у проекті **"ImgHub"** відіграє ключову роль у забезпеченні ефективного функціонування системи та задоволенні потреб користувачів. Важливість цього атрибуту полягає в забезпеченні таких аспектів:

1) **Швидкість завантаження фотографій:** Висока швидкість завантаження фотографій дозволяє користувачам швидко знаходити та переглядати фотографії без зайвих затримок. Це важливо для задоволення користувацьких очікувань та створення позитивного досвіду використання системи.

2) **Швидкість обробки пошуку:** Користувачі часто виконують пошук серед великого обсягу фотографій. Швидкий час відповіді на пошуковий запит дозволяє їм ефективно взаємодіяти з системою та знаходити потрібні фотографії.

3) **Ефективність обробки великих фотографій:** Багато користувачів завантажують великі фотографії, і система повинна швидко та ефективно обробляти їх для забезпечення швидкого перегляду та завантаження.

4) **Продуктивність при одночасних запитах користувачів:** Здатність системи обробляти багато одночасних запитів користувачів визначає, наскільки багато користувачів можуть одночасно взаємодіяти з системою без втрати продуктивності.

5) **Відновлення після відмови (Failover):** Важливо мати механізми відновлення, які дозволяють системі швидко відновитися після відмови чи збою, щоб забезпечити стабільність та доступність сервісу для користувачів.

6) **Збереження кешу для швидкого доступу до фотографій:** Кешування фотографій та запитів допомагає прискорити доступ користувачів до часто використовуваних фотографій та даних.

7) **Оптимізація обробки фотографій на стороні сервера:** Можливість оптимізації фотографій на сервері сприяє покращенню продуктивності та швидкості завантаження.

8) **Моніторинг продуктивності та автоматичне масштабування:** Система повинна бути здатною виявляти зростання навантаження та автоматично масштабуватися для забезпечення стабільної продуктивності.

9) **Впровадження CDN для прискорення доставки фотографій:** Використання мережі доставки контенту допомагає прискорити завантаження фотографій для користувачів з різних регіонів світу.

10) **Кешування запитів на сервері:** Кешування запитів на сервері допомагає зменшити навантаження та прискорити доступ до даних.

11) **Оптимізація завантаження віддалених зображень:** Оптимізація завантаження фотографій з інших віддалених ресурсів дозволяє покращити продуктивність системи.

Забезпечення високої продуктивності у всіх аспектах допомагає створити систему, яка задовольняє користувацькі потреби та надає позитивний досвід використання, що, в свою чергу, сприяє популярності та успіху проекту "ImgHub".

**Сценарії атрибуту якості "Performance" у проекті "ImgHub":**

**Сценарій №1:** Користувач шукає фотографію та очікує її негайного завантаження.

Метрика: Час завантаження фотографії.

Математична формула: Час завантаження = Час передачі даних + Час обробки запиту на сервері.

Приклади: Якщо час передачі даних становить 0,5 секунди, а час обробки запиту на сервері - 1 секунда, то загальний час завантаження буде 1,5 секунди.

**Сценарій №2 :** Користувач виконує пошук серед великого обсягу фотографій.

Метрика: Час відповіді на пошуковий запит.

Математична формула: Час відповіді = Час обробки пошукового запиту.

Приклади: Якщо час обробки пошукового запиту становить 0,2 секунди, то час відповіді на запит також дорівнює 0,2 секунди.

**Сценарій №3 :** Користувачі можуть завантажувати великі фотографії, і система повинна швидко обробляти та відтворювати їх.

Метрика: Час обробки великих фотографій.

Математична формула: Час обробки = Розмір фотографії / Продуктивність обробки.

Приклади: Якщо розмір великої фотографії становить 20 МБ, і продуктивність обробки - 10 МБ в секунду, то час обробки буде 2 секунди.

**Сценарій №4 :** Декілька користувачів одночасно шукають та завантажують фотографії.

Метрика: Максимальна кількість одночасних запитів, яку система може обробити.

Математична формула: Кількість запитів \* Час обробки одного запиту.

Приклади: Якщо система може обробити 20 одночасних запитів, і час обробки одного запиту - 0,5 секунди, то максимальна продуктивність дорівнює 10 запитів/с.

**Сценарій №5 :** Одна з серверних одиниць відмовляє, і необхідне автоматичне відновлення.

Метрика: Час відновлення та стабільність системи після відмови.

Математична формула: Час відновлення = Час виявлення відмови + Час переключення на резервний сервер.

Приклади: Якщо система виявляє відмову через 5 секунд і вдало переключається на резервний сервер за 2 секунди, то час відновлення складає 7 секунд.

**Сценарій №6 :** Користувачі часто переглядають одні й ті ж фотографії.

Метрика: Час доступу до кешованих фотографій.

Математична формула: Час доступу до кешу = Час пошуку фотографії в кеші.

Приклади: Якщо час пошуку фотографії в кеші становить 0,1 секунди, то користувачі отримають доступ до кешованих фотографій дуже швидко.

**Сценарій №7 :** Фотографії можуть бути автоматично оптимізовані на сервері для прискорення завантаження та перегляду.

Метрика: Час обробки фотографій на сервері.

Математична формула: Час обробки фотографії = Складність обробки \* Кількість фотографій.

Приклади: Якщо складність обробки однієї фотографії становить 0,5 секунди, і користувачі обробляють 100 фотографій, то загальний час обробки буде 50 секунд.

**Сценарій №8 :** Обсяг користувачів може зростати, і система повинна автоматично масштабуватися для забезпечення стабільної продуктивності.

Метрика: Час виявлення зростання навантаження та час автоматичного масштабування.

Математична формула: Час реакції на зростання навантаження.

Приклади: Якщо система виявляє зростання навантаження і масштабується протягом 2 хвилин, то вона залишається продуктивною при збільшенні обсягу користувачів.

**Сценарій №9 :** Контент доставляється через мережу доставки контенту (CDN) для підвищення швидкості завантаження.

Метрика: Час завантаження з використанням CDN.

Математична формула: Час завантаження з CDN = Час передачі даних через CDN + Час обробки запиту на сервері.

Приклади: Якщо час передачі даних через CDN становить 0,2 секунди, а час обробки запиту на сервері - 0,5 секунди, то загальний час завантаження буде 0,7 секунди.

**Сценарій №10 :** Запити користувачів можуть кешуватися на сервері для прискорення відповідей.

Метрика: Час доступу до кешованих запитів.

Математична формула: Час доступу до кешованого запиту = Час пошуку запиту в кеші.

Приклади: Якщо час пошуку запиту в кеші становить 0,1 секунди, то користувачі отримають доступ до кешованих даних дуже швидко.

**Сценарій №11 :** Зображення завантажуються з інших віддалених ресурсів.

Метрика: Час завантаження віддалених зображень.

Математична формула: Час завантаження = Час передачі даних + Час обробки запиту на сервері.

Приклади: Якщо час передачі даних становить 0,3 секунди, а час обробки запиту на сервері - 0,2 секунди, то загальний час завантаження.

Ці сценарії і математичні формули демонструють, чому атрибут **"Performance"** є критичним для успішного функціонування проекту ImgHub. Висока продуктивність допомагає забезпечити швидке та ефективне обслуговування користувачів, надаючи можливість знайти та завантажити бажані фотографії без затримок. Це важливо для задоволення потреб користувачів та створення позитивного користувацького досвіду на ImgHub.