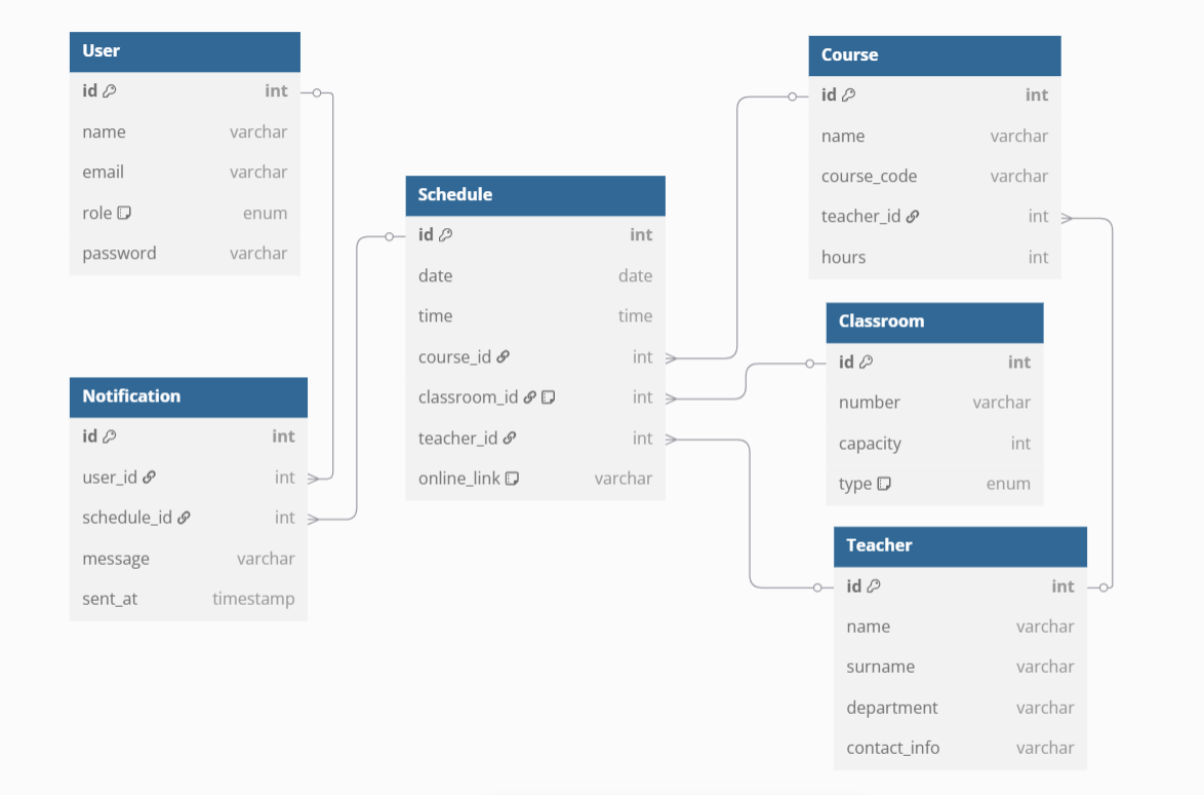
1. Data model

1.1 ER діаграма



1.2 Опис сутностей для ER-діаграми:

1. User

- Опис: Таблиця користувачів, яка зберігає інформацію про студентів, викладачів та адміністраторів системи.
- Атрибути:
  - id: Унікальний ідентифікатор користувача.
  - name: Ім'я користувача.
  - email: Адреса електронної пошти користувача (унікальна).
  - role: Роль користувача (студент, викладач, адміністратор).
  - password: Пароль користувача.

2. Teacher

- Опис: Таблиця, що містить дані викладачів, які читають курси.

- Атрибути:
  - id: Унікальний ідентифікатор викладача.
  - name: Ім'я викладача.
  - surname: Прізвище викладача.
  - department: Відділ або кафедра, до якої належить викладач.
  - contact_info: Контактна інформація викладача (наприклад, телефон, email).

3. Course

- Опис: Таблиця, що містить курси, які викладаються у навчальному закладі. Кожен курс прив'язаний до викладача.
- Атрибути:

  - id: Унікальний ідентифікатор курсу.
  - name: Назва курсу.
  - course_code: Унікальний код курсу.
  - teacher_id: Ідентифікатор викладача, який викладає курс (зв'язок із таблицею Teacher).
  - hours: Кількість годин, відведених на курс.

4. Classroom

- Опис: Таблиця, що містить інформацію про аудиторії або типи онлайн-занять.
- Атрибути:
  - id: Унікальний ідентифікатор аудиторії.
  - number: Номер аудиторії.
  - capacity: Ємність аудиторії (кількість людей, що можуть бути присутні на занятті).

- type: Тип аудиторії (лекційний зал, лабораторія, комп'ютерний клас або онлайн-заняття).

5. Schedule

- Опис: Таблиця, що містить розклад занять для кожного курсу, викладача та аудиторії. Якщо заняття проводиться онлайн, то поле classroom_id може бути NULL.
- Атрибути:
  - id: Унікальний ідентифікатор запису розкладу.
  - date: Дата заняття.
  - time: Час початку заняття.
  - course_id: Ідентифікатор курсу, який викладається (зв'язок із таблицею Course).
  - classroom_id: Ідентифікатор аудиторії (може бути NULL для онлайн-занять, зв'язок із таблицею Classroom).
  - teacher_id: Ідентифікатор викладача, що проводить заняття (зв'язок із таблицею Teacher).
  - online_link: Посилання на онлайн-заняття (якщо є, інакше NULL).

6. Notification

- Опис: Таблиця, що містить сповіщення для користувачів (студентів, викладачів). Сповіщення можуть бути надіслані, коли змінюється розклад або є інші важливі оновлення.
- Атрибути:
  - id: Унікальний ідентифікатор сповіщення.
  - user_id: Ідентифікатор користувача, який отримує сповіщення (зв'язок із таблицею User).

- ❖ schedule_id: Ідентифікатор розкладу, до якого стосується сповіщення (зв'язок із таблицею Schedule).
- ❖ message: Повідомлення, яке буде надіслано користувачу.
- ❖ sent_at: Час, коли сповіщення було надіслано.

1.3 Класифікація даних по Data Retention Policy

Тип даних	**Час збереження**	**Умови видалення**
Дані користувачів	5 років	Видалення після закінчення навчання або за запитом
Розклад занять	1 рік	Архівування після закінчення семестру
Логи змін розкладу	6 місяців	Автоматичне очищення
Дані про аудиторії	Постійно	Оновлення адміністратором
Інформація про курс	3 роки	Видалення після оновлення курсу

2. Resiliency model

2.1 CID-Діаграма (Customer Interaction Diagram)

Interaction ID	***From***	***To***	***Description***
1.	Браузер	API	Виконує HTTP-запити (GET/POST/DELETE/PUT) для доступу чи модифікації ресурсів розкладу.
2.	API	Бізнес-логіка	API перенаправляє запити на рівень бізнес-логіки для обробки, валідації та виконання логіки.
3.	Бізнес-логіка	Шар доступу до даних	Бізнес-логіка надсилає запити чи команди для взаємодії з базою даних, наприклад, перевірка доступності розкладу.
4.	Шар доступу до даних	PostgreSQL DB	Виконує SQL-запити (наприклад, SELECT, INSERT, UPDATE) для читання чи запису даних розкладу.
5.	PostgreSQL DB	Шар доступу до даних	Повертає результати операцій з базою даних (наприклад, оновлені дані розкладу) назад до шару доступу.

6.	Шар доступу до даних	Бізнес-логіка	Подає оброблені дані чи результати з бази даних (наприклад, оновлений розклад) бізнес-логіці.
7.	Бізнес-логіка	API	Надсилає відповідь або оброблені дані назад до API для доставки клієнту, наприклад, оновлений розклад.
8.	API	Браузер	Надсилає HTTP-відповіді (JSON, XML) назад до браузера для взаємодії з користувачем (наприклад, відображення оновленого розкладу).
9.	API	Mailtrap SMTP	Надсилає запити до SMTP-сервісу для відправки транзакційних або сповіщувальних email-лів, таких як зміни в розкладі.
10.	Mailtrap SMTP	Вхідна пошта користувача	Доставляє email-листів

2.2 RMA Workbook (модель відмов)

Інтеракція 1: Browser → API

Помилка	**Опис помилки**
1. Невірний запит	Користувач може надіслати некоректний запит, наприклад, із відсутніми параметрами або неправильним форматом URL.
2. Втрата з'єднання	Може бути проблема з мережею або з’єднанням між браузером і сервером, що призводить до втрати запиту.
3. Проблеми з аутентифікацією	Користувач може бути неавторизованим або сесію було припинено, що блокує доступ до ресурсу.

Інтеракція 2: API → Бізнес-логіка

Помилка	Опис помилки
1. Логічна помилка	API передає некоректний запит до бізнес-логіки, що викликає непередбачувану поведінку чи помилку обробки.
2. Перевантаження запитів	API може відправити забагато запитів до бізнес-логіки, що призводить до перевантаження системи та затримки обробки.
3. Відсутність необхідних даних	У запиті відсутні дані, необхідні для обробки бізнес-логікою (наприклад, відсутня інформація про користувача або параметри).

Інтеракція 3: Бізнес-логіка → Шар доступу до даних

Помилка	Опис помилки
1. Помилка в SQL-запиті	Запит до бази даних містить помилки в синтаксисі або логіці, що призводить до не успішного виконання операції.
2. Відсутність даних	У базі даних відсутні необхідні записи, наприклад, розклад для конкретного дня чи курсу.
3. Проблеми з доступом до бази	Шар доступу до даних не може підключитися до бази даних через помилки в налаштуваннях або мережеві проблеми.

Інтеракція 4: Шар доступу до даних → PostgreSQL DB

Помилка	Опис помилки
1. Часове обмеження	База даних не відповідає через тривалі запити, що викликає затримки в обробці запитів.
2. Пошкодження даних	SQL-запит може викликати пошкодження або неповний запис в базі даних (наприклад, через некоректне оновлення розкладу).

3. Втрата з'єднання з БД	Переривання з'єднання між шаром доступу до даних і базою через мережеві проблеми чи перевантаження БД.

Інтеракція 5: Mailtrap SMTP → Вхідна пошта користувача

Помилка	Опис помилки
1. Невірна адреса email	Неправильна або неіснуюча email-адреса у системі викликає невдачу в доставці повідомлення.
2. Проблеми з SMTP-сервером	Виникають помилки під час взаємодії з сервером SMTP, що призводить до неможливості відправити email.
3. Спам-фільтри	Сповіщення потрапляє в спам або блокується фільтрами, через що користувач не отримує повідомлення.

3. Security model

3.1 Основні флови та моделювання загроз (Threat Model)

Флов 1: Перегляд розкладу користувачем

Етапи:

1. Користувач у браузері надсилає запит до API.
2. API перенаправляє запит до бізнес-логіки.
3. Бізнес-логіка взаємодіє з базою даних.
4. Дані повертаються до браузера.

Загрози:

- **T1:** SQL Injection.
- **T2:** Перегляд розкладу іншого користувача без прав.
- **T3:** Витік даних через HTTP.

Флов 2: Оновлення розкладу адміністратором

Етапи:

1. Адміністратор автентифікується.
2. Надсилає POST/PUT запит до API.
3. Дані оновлюються в БД.
4. Надсилаються email-сповіщення.

Загрози:

- **T4:** Несанкціонований доступ до адміністративної функції.
- **T5:** Масове редагування або видалення даних (або помилково, або зловмисно).
- **T6:** Вставка шкідливого коду (XSS).

Флов 3: Реєстрація/логін користувача

Етапи:

1. Користувач надсилає логін-запит із credentials.
2. API передає логіку авторизації (наприклад, перевірка паролю).
3. Користувачу видається токен або сесія.

Загрози:

- **T7:** Слабкі паролі, brute force атаки.
- **T8:** Відсутність захисту від CSRF при логіні.
- **T9:** Зберігання паролів у відкритому вигляді.

Флов 4: Відправка email-сповіщення

Етапи:

1. Зміна розкладу викликає генерацію повідомлення.
2. Повідомлення формуються та надсилаються SMTP-серверу.
3. Email доставляється користувачу.

Загрози:

- **T10:** Фальшиві email або фішинг.
- **T11:** Проблеми з SMTP (недоступність, втрата повідомлень).
- **T12:** Попадання в спам (відсутність SPF/DKIM/DMARC).
- **T13:** Відправка сповіщення не тому користувачу (data leakage).
- **T14:** Повторна відправка сповіщення або спам.

3.2 Mitigation Plan

Загроза	Назва	Mitigation
T1	SQL Injection	Використовувати параметризовані запити та ORM.
T2	Несанкціонований доступ	Впровадити RBAC, перевірку ідентифікаторів користувача.
T3	Незашифроване з'єднання	Примусово використовувати HTTPS на всіх рівнях.
T4	Привілейована ескалація	Авторизація на рівні маршруту + перевірка ролей.
T5	Масове пошкодження даних	Аудит логів, версійність змін, автоматичні резервні копії.
T6	XSS	Санітаризація введення, Content Security Policy (CSP).
T7	Brute-force	Блокування після X спроб, CAPTCHA, rate limiting.
T8	CSRF	Використання CSRF-токенів на критичних POST/PUT.

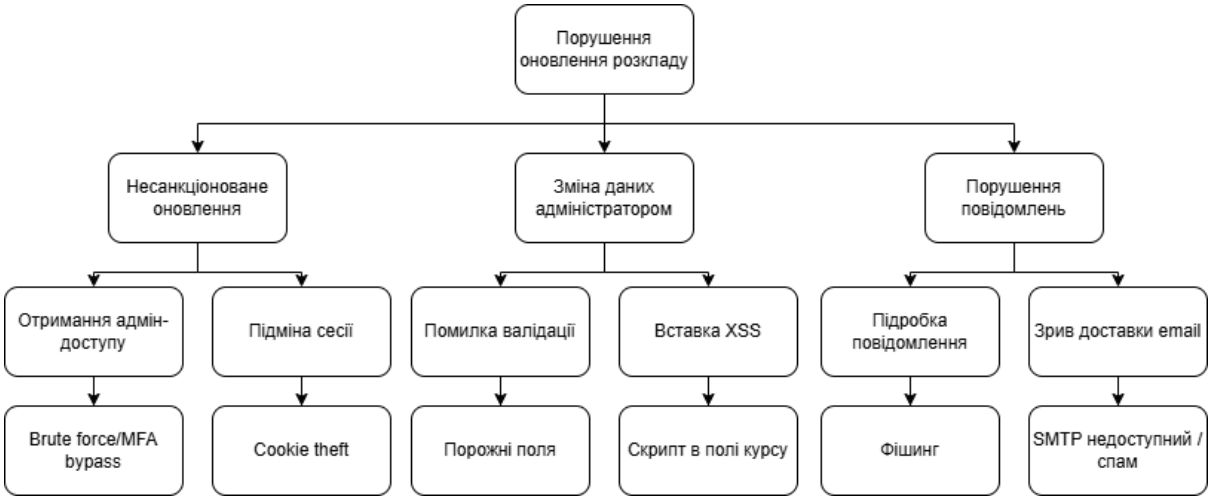
T9	Зберігання паролів у plaintext	Використання bcrypt або Argon2, соль.
T10	Email spoofing/phishing	Налаштування SPF, DKIM, DMARC.
T11	SMTP недоступний	Резервний SMTP, автоматичне повторне надсилання.
T12	Email потрапляє у спам	Моніторинг доставок, коректна конфігурація SMTP.
T13	Витік через email	Логіка перевірки одержувача перед надсиланням.
T14	Спам-розсилка	Rate limiting для email queue, throttle.
T15	DDoS API	Rate limiting, захист API Gateway, Cloudflare/AWS Shield.

3.3 STRIDE-таблиця загроз по фловах

№	Флов	Загроза	Тип STRIDE	Опис
1	Перегляд розкладу	SQL Injection	Tampering	Можливість змінити логіку запиту до БД.
2	Перегляд розкладу	Несанкціонований перегляд чужого розкладу	Information Disclosure	Доступ до приватної інформації інших користувачів.
3	Перегляд розкладу	HTTP-з'єднання	Information Disclosure	Перехоплення чутливих даних у трафіку.
4	Оновлення розкладу	Доступ без авторизації	Elevation of Privilege	Користувач дістає доступ до функцій адміністратора.
5	Оновлення розкладу	Масове редагування/видалення	Tampering	Навмисне або помилкове порушення цілісності даних.
6	Оновлення розкладу	XSS у формі	Tampering	Вставка шкідливого скрипта, що виконується у браузері.
7	Реєстрація/логін	Слабкі паролі / brute-force	Spoofing	Зловмисник підбирає облікові дані користувача.

8	Реєстрація/логін	CSRF під час авторизації	Tampering	Запит надсилається без згоди користувача.
9	Реєстрація/логін	Незахищене зберігання паролів	Information Disclosure	Компрометація бази з паролями.
10	Email-сповіщення	Підробка email / фішинг	Spoofing	Видавання зловмисника за сервіс.
11	Email-сповіщення	Відсутність SMTP або втрати	Denial of Service	Зрив сповіщень, важлива інформація не доходить.
12	Email-сповіщення	Потрапляння в спам	Denial of Service	Повідомлення ігноруються або блокуються.
13	Email-сповіщення	Надсилання не тому користувачу	Information Disclosure	Витік конфіденційної інформації.
14	Email-сповіщення	Масове дублювання email (спам)	Repudiation / DoS	Втрата довіри, перевантаження inbox.
15	Всі API флови	DDoS-атака	Denial of Service	Перевантаження сервісу.

3.4 Приклад Attack Tree: Флов «Оновлення розкладу адміністратором»



Пояснення вузлів:

Несанкціоноване оновлення

- Отримання адмін-доступу – використання слабких паролів, відсутності MFA.
- Підміна сесії – крадіжка токена сесії через XSS або інші методи.

Зміна даних адміністратором

- Помилка валідації – наприклад, відсутня перевірка формату дати або аудиторії.
- XSS у формі – вставлення скрипта, який виконається для інших користувачів.

Порушення повідомлень

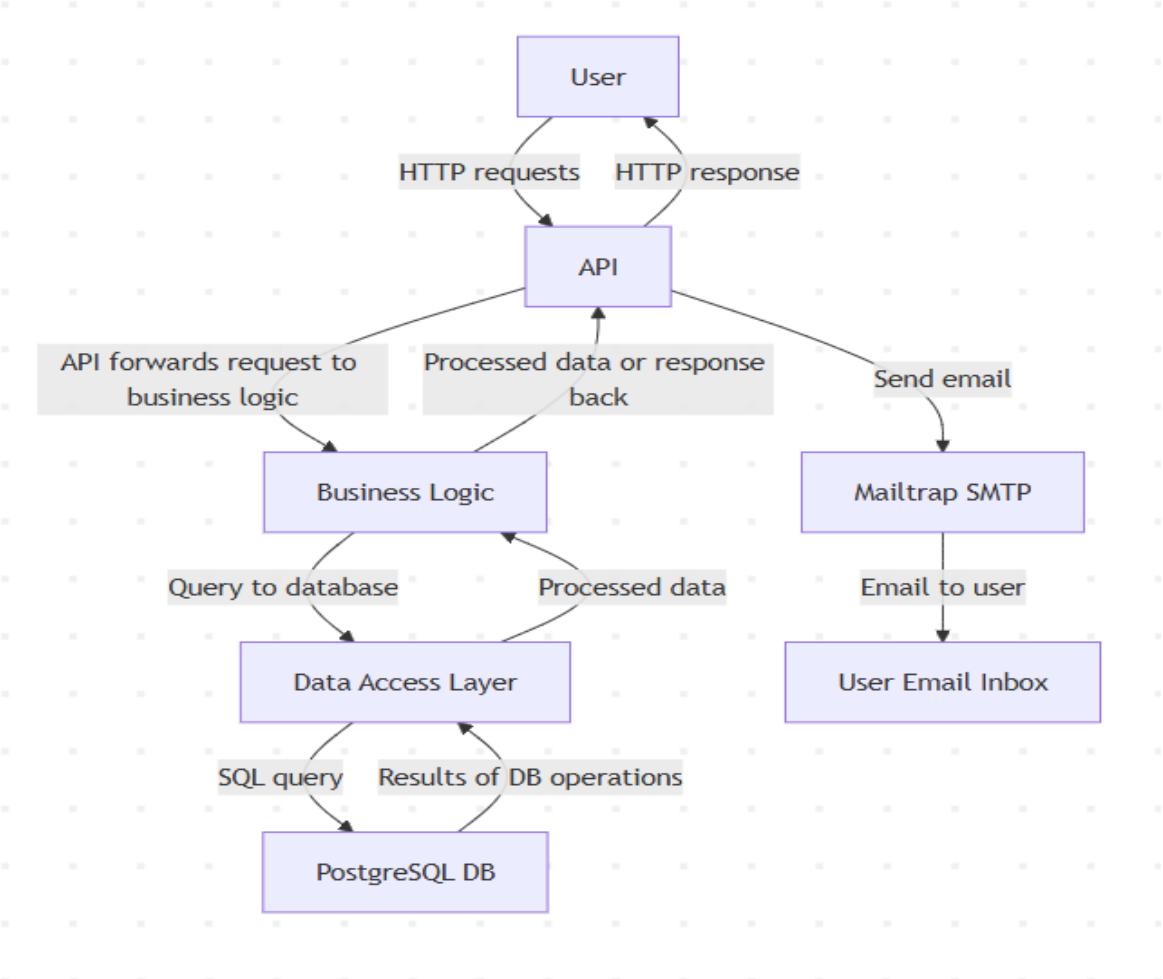
- Підробка повідомлення – фішинговий лист або зміна тексту повідомлення.
- Зрив доставки – SMTP сервер не працює або лист потрапляє в спам.

Захист проти кожної гілки:

Гілка	Захист
Brute force, MFA bypass	CAPTCHA, MFA, обмеження спроб входу
Cookie theft	Secure/HttpOnly cookies, SameSite policy
Валідація	Валідація на бекенді + санітаризація
XSS	HTML-санітаризація, CSP
SMTP/spam	SPF, DKIM, DMARC, моніторинг доставок
Фішинг	Чіткий брендинг email, захист SMTP

4. Deployment model

4.1 Інфраструктурна діаграма проекту:



4.2 Основні компоненти інфраструктури:

1. **Користувачі (Browser)**:
   - Веб-браузер або мобільний додаток користувача, що взаємодіє з веб-сайтом або API.
   - Підключення до Інтернету для доступу до сервісу.
2. **API Gateway**:
   - **Роль**: Це точка входу для всіх клієнтських запитів до системи. API Gateway відправляє запити до відповідних сервісів бізнес-логіки.

- **Тип ресурсу**: Сервер або контейнер, що працює як балансувальник навантаження.
- **Розміщення**: Веб-сервіс (наприклад, AWS API Gateway або Nginx).

3. **Сервіси бізнес-логіки (Business Logic Services)**:
   - **Роль**: Обробка запитів, валідація та виконання логіки. Ці сервіси взаємодіють з даними, здійснюють валідацію, і в залежності від запиту відправляють відповіді.
   - **Тип ресурсу**: Віртуальні машини або контейнери (наприклад, AWS EC2 або Docker).
   - **Розміщення**: Хмарні сервіси або локальні сервери.
4. **Шар доступу до даних (Data Access Layer)**:
   - **Роль**: Взаємодія з базами даних для виконання CRUD (Create, Read, Update, Delete) операцій.
   - **Тип ресурсу**: Сервер доступу до даних, API для взаємодії з БД.
   - **Розміщення**: Хмарний сервіс, зокрема AWS Lambda або серверні інстанси.
5. **База даних (Database)**:
   - **Роль**: Зберігання даних про користувачів, розклад, повідомлення та іншу інформацію.
   - **Тип ресурсу**: Реляційна база даних (наприклад, PostgreSQL, MySQL), хмарне сховище (AWS RDS, Google Cloud SQL).
   - **Розміщення**: В хмарі, віртуалізована інфраструктура.
6. **Мережеві ресурси (Load Balancer, DNS, CDN)**:
   - **Роль**: Балансування навантаження між серверами для забезпечення високої доступності та швидкого доступу.
   - **Тип ресурсу**: Локальні або хмарні ресурси, такі як AWS Elastic Load Balancer, CloudFront (CDN).

- **Розміщення**: Хмарне середовище.

7. **Система сповіщень (Notification System)**:
   - **Роль**: Відправка повідомлень користувачам через електронну пошту, SMS чи інші канали.
   - **Тип ресурсу**: SMTP-сервер (наприклад, AWS SES, SendGrid), вбудовані сервіси для сповіщень.
   - **Розміщення**: В хмарному середовищі.
8. **Система зберігання файлів (File Storage)**:
   - **Роль**: Зберігання медіа-файлів, документів, логів.
   - **Тип ресурсу**: Хмарне сховище (наприклад, AWS S3, Google Cloud Storage).
   - **Розміщення**: Хмарне середовище.
9. **Моніторинг та логування (Monitoring & Logging)**:
   - **Роль**: Збір і аналіз логів і метрик для моніторингу здоров'я системи та визначення аномалій.
   - **Тип ресурсу**: AWS CloudWatch, Prometheus, Grafana для моніторингу.
   - **Розміщення**: Хмарне середовище або локальні сервери.
10. **Система автентифікації та авторизації (Authentication & Authorization)**:
    - **Роль**: Здійснення перевірки і авторизації користувачів.
    - **Тип ресурсу**: OAuth, JWT, OpenID Connect, Firebase Authentication.
    - **Розміщення**: Хмарне середовище або локальні сервери.

4.3 Пояснення типів ресурсів:

1. **API Gateway**:
   - **Ресурси**: Це сервери, які контролюють і маршрутизують запити, балансуючи навантаження між різними сервісами та додаючи додаткові рівні безпеки.
   - **Пояснення**: Збирає запити від користувачів, пересилаючи їх на відповідні бізнес-сервіси. Впроваджує логіку маршрутизації та управління.
2. **Бізнес-логіка**:
   - **Ресурси**: Віртуальні машини або контейнеризовані сервіси, які обробляють логіку програми.
   - **Пояснення**: Приймає запити з API, обробляє їх, виконуючи валідацію, обчислення або інші бізнес-процеси, потім передає результат в базу даних або інші сервіси.
3. **База даних**:
   - **Ресурси**: Реляційні або документоорієнтовані бази даних.
   - **Пояснення**: Зберігає інформацію про користувачів, їхні налаштування, історію і розклад, з якими взаємодіють бізнес-логіка та API.
4. **Сповіщення**:
   - **Ресурси**: SMTP-сервіси для відправки електронних листів або інші сервіси для повідомлень (SMS, Push).
   - **Пояснення**: Використовується для відправки користувачам повідомлень (наприклад, підтвердження реєстрації, зміни розкладу).

5. Analytics model

5.1 Таблиця основних функціональних метрик

Метрика	Вимірювання	Зв'язок з функціональними вимогами
Час відповіді API	Час відправлення запиту до отримання відповіді від API	Створення розкладу, перегляд розкладу
Кількість успішних запитів до API	Кількість запитів, що отримали відповідь 200 (OK)	Перегляд розкладу, редагування розкладу
Кількість неуспішних запитів до API	Кількість запитів, що отримали помилку (наприклад, 404, 500)	Редагування розкладу, видалення заняття
Час виконання запиту до бази даних	Час виконання SQL запиту (SELECT, INSERT, UPDATE)	Створення розкладу, перегляд розкладу
Час оновлення розкладу	Час між внесенням змін в розклад і їх відображенням	Автоматичне оновлення розкладу
Частота доступу до розкладу	Кількість запитів до розкладу за певний період (наприклад, день, тиждень)	Перегляд розкладу
Кількість помилок у відправлених email	Кількість помилок при відправленні email (SMTP помилки)	Сповіщення про зміни в розкладі
Кількість користувачів, які не отримали сповіщення	Кількість користувачів, яким не було надіслано сповіщення	Сповіщення про зміни в розкладі

Час до сповіщення	Час між зміною розкладу та отриманням сповіщення користувачем	Сповіщення про зміни в розкладі
Частота редагування розкладу	Кількість разів, коли адміністратор змінює розклад	Редагування розкладу
Кількість скасованих занять	Кількість занять, які були скасовані або видалені	Видалення заняття з розкладу
Час доступу до онлайн-уроку	Час, необхідний для підключення до онлайн-заняття	Створення розкладу, перегляд розкладу (для онлайн-занять)
Час завантаження сторінки для користувача	Час від запиту користувача до повного завантаження сторінки	Перегляд розкладу
Час редагування заняття адміністратором	Час, необхідний адміністратору для внесення змін	Редагування розкладу
Кількість успішних сповіщень по email	Кількість коректно надісланих email-повідомлень	Сповіщення про зміни в розкладі

5.2 Funnel-метрики

1. **Funnel для перегляду розкладу**:
   - **Метрики**:
     - Час відповіді API
     - Кількість успішних запитів до API
     - Кількість користувачів, які переглядають розклад.
   - **Логіка**:
     - Користувач ініціює запит на перегляд розкладу → API відповідає успішно → Користувач переглядає розклад.

2. **Funnel для сповіщення користувачів про зміни в розкладі**:
    - **Метрики**:
        - Час до сповіщення
        - Кількість користувачів, які не отримали сповіщення
        - Кількість успішних сповіщень по email
    - **Логіка**:
        - Адміністратор змінює розклад → Сповіщення відправляються користувачам → Користувач отримує сповіщення.

6. Monitoring & Alerting model

6.1 Операційні Метрики (Monitoring)

Метричні	Опис	Одиниця виміру	Спосіб збору	Підключення до інфраструктурних ресурсів
CPU використання	Відсоток використання центрального процесора сервером.	Відсотки (%)	Використання системних інструментів моніторингу (Prometheus, Grafana)	Сервери (API сервери, сервери обробки запитів)
Оперативна пам'ять (RAM)	Відсоток використаної оперативної пам'яті на сервері.	Відсотки (%)	Збір через системи моніторингу (Nagios, Zabbix)	Сервери (API сервери, сервери обробки запитів)
Затримка запиту до API (API Latency)	Час, який проходить від запиту до API до отримання відповіді.	Мілісекунди (ms)	Логування через API Gateway або спеціальні моніторингові інструменти	API сервіси, сервери обробки запитів
Пропускна здатність мережі	Швидкість передавання даних між серверами та користувачами.	Мегабіти на секунду (Mbps)	Використання інструментів моніторингу мережі (NetFlow, Grafana)	Мережеві ресурси, сервери (мережеві інтерфейси, маршрутизатори)

Стан підключень до БД	Кількість активних підключень до бази даних.	Кількість підключень	Моніторинг з допомогою бази даних або інструментів для моніторингу баз (PostgreSQL)	База даних (PostgreSQL)
Час відновлення після збою (RTO)	Час, що необхідний для відновлення після системного збою.	Час (в секундах або хвилинах)	Використання журналу подій та моніторингу збереження даних	Сервери, бази даних, резервні копії
Час доставки email	Час між запитом на відправку email і доставкою повідомлення користувачу.	Мілісекунди (ms)	Логування через SMTP сервери або використання сторонніх провайдерів	Сервери SMTP, мережа
Кількість активних сесій	Кількість одночасно активних сесій користувачів в системі.	Кількість сесій	Моніторинг через API або додаткові логи сесій	Сервери, API, база даних
Час відповіді від БД	Час затримки для виконання SQL-запиту до бази даних.	Мілісекунди (ms)	Логування запитів до бази даних (через PgStat)	База даних (PostgreSQL, MySQL)
Навантаження на сервер	Відсоток завантаження серверних ресурсів (CPU, RAM).	Відсотки (%)	Використання моніторингових інструментів для серверів	Сервери (API сервери, сервери обробки запитів)

Проблеми із захистом від DDoS-атак	Кількість заблокованих підозрілих IP адрес або запитів, що відповідають характеристикам DDoS.	Кількість запитів	Використання інструментів моніторингу DDoS-атак (AWS Shield, Cloudflare)	Мережеві ресурси, захист від DDoS
Наявність резервних копій	Перевірка актуальності та наявності резервних копій даних.	Boolean (True/False)	Моніторинг через засоби управління резервними копіями (Backup tools)	Сервери, хмарні ресурси
Частота перевірки систем безпеки	Частота запуску сканувань та перевірок на вразливості системи.	Період (години/дні)	Використання інструментів для моніторингу безпеки (Qualys, Nessus)	Сервери, мережеві інтерфейси, системи безпеки
Кількість збоїв з підключенням до API	Кількість відмов в підключеннях до API серверів.	Кількість помилок	Логування через API Gateway, сервіси моніторингу	API сервери, мережа
Час роботи сервера	Час, протягом якого сервер працює без перерв.	Час (в годинах)	Моніторинг через сервіс управління сервером (Nagios, Datadog)	Сервери

6.2 Alerting – Мін/Макс Допустимі Значення

Метричні	Мінімальне значення	Максимальне значення	Тип метрики	Критичність досягнення критичних значень	Мітки при досягненні критичних значень	Mitigation Plan
CPU використання	10%	90%	Ресурсна метрика	Висока	Відмова у роботі серверів, зависання системи	Оповіщення, автоматичне масштабування серверів, додавання ресурсів
Оперативна пам'ять (RAM)	20%	95%	Ресурсна метрика	Висока	Перевантаження, повільна робота системи	Оповіщення, звільнення пам'яті (закриття процесів), масштабування
Час відповіді від API (API Latency)	100ms	2000ms	Продуктивність сервісу	Висока	Низька продуктивність, відмова запитів користувачів	Оповіщення, оптимізація запитів до API, додавання серверів
Пропускна здатність мережі	1 Mbps	1000 Mbps	Продуктивність мережі	Середня	Затримка передачі даних, повільний доступ до ресурсу	Оповіщення, масштабування мережі, перевірка на перегрузку
Затримка запитів до бази даних	20ms	1000ms	Продуктивність бази даних	Висока	Низька продуктивність запитів,	Оповіщення, оптимізація SQL-запитів, індексація таблиць

					блокування користувачів	
Час доставки email	0ms	1000ms	Інтерфейс повідомлень	Середня	Затримка в доставці, низька швидкість обробки email	Оповіщення, перевірка конфігурацій SMTP, використання резервних каналів
Кількість активних сесій	0	10000	Продуктивність сервісу	Середня	Наявність багатократних підключень, можливість DDoS-атак	Оповіщення, обмеження кількості сесій, перевірка на DDoS-атаки
Навантаження на сервер	10%	85%	Ресурсна метрика	Висока	Завантаження сервера, затримки в обробці запитів	Оповіщення, масштабування серверів, балансування навантаження
Наявність резервних копій	True	False	Операційна метрика	Висока	Відсутність резервних копій, неможливість відновлення	Оповіщення, ініціювання резервного копіювання, перевірка процедур
Збої з підключенням до API	0	5	Продуктивність сервісу	Висока	Втрата з'єднання, відмови в роботі сервісу	Оповіщення, перезапуск сервісу, перевірка інфраструктури API