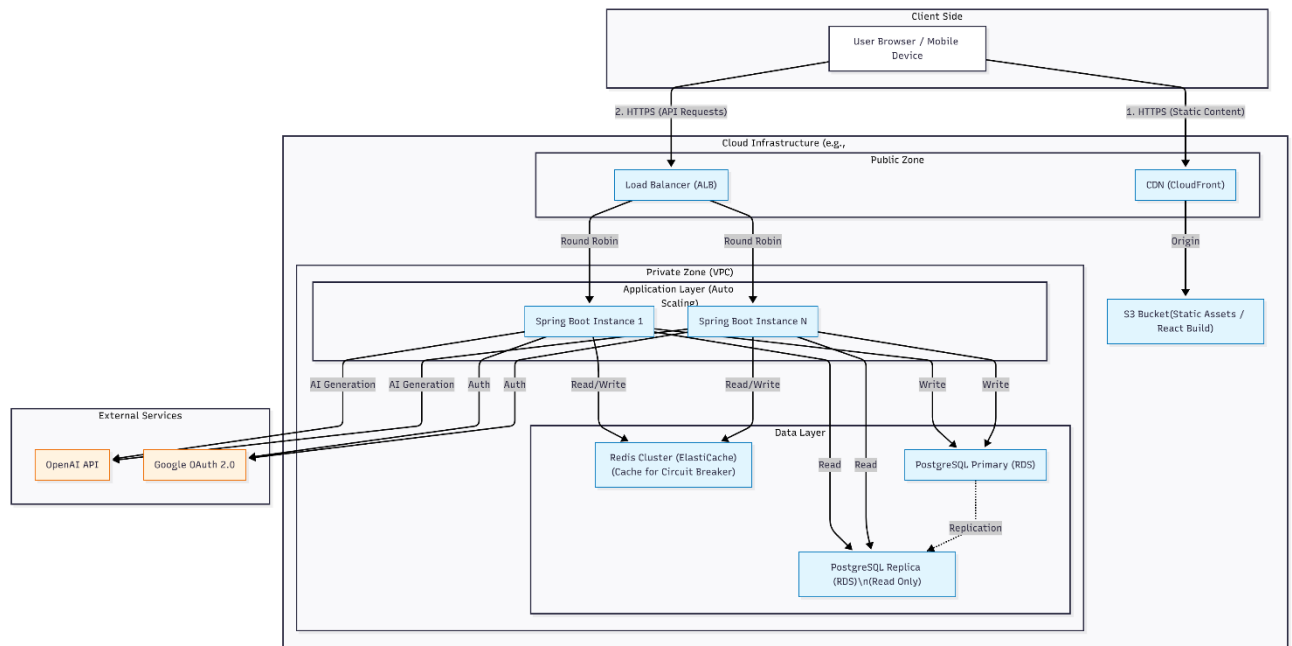


Deployment Model



Застосунок «**Personalized Recipe Generator**» є хмарним вебзастосунком на базі React SPA та Spring Boot, розгорнутих у середовищі AWS Cloud. Архітектура складається з 6 основних зон:

1. **Client Side**
2. **Public Zone (Edge Layer)**
3. **Application Layer (Auto Scaling Group)**
4. **Data Layer**
5. **Private Zone (VPC)**
6. **External Services**

1. Client Side

Browser / Mobile Device

Призначення: Кінцевий клієнт, який взаємодіє з вебзастосунком.

Функції:

- Відображення React SPA
- Надсилання REST API запитів на backend
- Виконання OAuth2 flow з Google
- Завантаження PDF-файлів
- Доступ до сторінки каталогу, улюблених та панелі користувача

Типова активність:

- GET запити до CDN (React bundle, images)
- POST/GET/PUT/DELETE до API (через ALB)

2. Public Zone (Edge Layer)

CDN (CloudFront)

Призначення: Географічне кешування статичного контенту.

Що зберігається / кешується:

- React build (index.html, JS, CSS)
- Зображення рецептів
- Статичні Assets

Переваги:

- Зниження навантаження на S3
- Прискорення завантаження SPA
- Edge-кешування для глобальних користувачів

Технології: AWS CloudFront

Application Load Balancer (ALB)

Призначення: Приймає всі API-запити від клієнта → маршрутизує їх у Auto Scaling Group backend інстансів.

Функції:

- Routing за правилами (host/path-based)
- HTTPS termination
- Health check backend інстансів
- Захист базовим firewall рівнем

Потоки:

- Browser → HTTPS → ALB
- ALB → HTTP/HTTPS → Spring Boot Instances

3. Application Layer (Auto Scaling)

Spring Boot Instances (Instance 1...N)

Знаходяться у приватних сабнетах VPC.

Призначення: Головний сервер застосунку, який виконує всю бізнес-логіку.

Основні модулі:

- AuthModule (JWT + OAuth2 Google login)
- RecipeGenerationModule (інтеграція з OpenAI API, Circuit Breaker, Semaphore)
- RecipeManagementModule (CRUD рецептів, фільтри)
- PDFExportService (ThreadPool для асинхронної PDF генерації)
- Favorites / Notes / History Module
- RateLimiter (захист від надмірних запитів)

Взаємодія:

- Read/Write із Redis

- Write до Primary PostgreSQL
- Read із Replica PostgreSQL
- API-запити до OpenAI
- OAuth2 обмін з Google Auth

Auto Scaling можливості:

- Масштабування за CPU, Memory, AI latency
- Автоматична реплікація інстансів
- Заміна “збитих” інстансів через health check

4. Data Layer

Redis Cluster (ElastiCache)

Призначення: High-performance in-memory кеш.

Використання:

1. Circuit Breaker Cache — зберігає fallback-рецепти у разі відмови OpenAI
2. Rate Limiting — IP/User-based counters
3. Caching популярних рецептів

Переваги:

- Мінімальна затримка (< 1 ms)
- Розподілений доступ для всіх Spring Boot інстансів

PostgreSQL Primary (RDS)

Призначення: Основна база даних для запису операцій.

Що зберігається:

- Користувачі
- Улюблені рецепти
- Історія генерацій
- AI згенеровані рецепти
- Нотатки
- Аналіз фільтрів

PostgreSQL Read Replica

Призначення: Високопродуктивні читання для масштабування.

Застосування:

- Перегляд каталогу
- Користувацькі пошуки
- Сторінка історії рецептів
- Аналітично важкі SELECT

Синхронізація: DB_Primary → async replication → DB_Replica

5. Private Zone (VPC)

VPC

Призначення: Забезпечує сегментацію мережі та безпеку.

Компоненти у приватних сабнетах:

- Spring Boot Instances
- Redis
- PostgreSQL Primary & Replica

Безпека:

- Security Groups (доступ тільки з ALB до backend)
- Немає прямого доступу з Інтернету
- RDS доступний лише з backend інстансів

S3 Bucket

Призначення: Зберігання:

- React build артефактів (для CDN origin)
- Зображень рецептів
- PDF експортів

Переваги:

- Надійне зберігання
- Інтеграція з CloudFront
- Режим Private/Public залежно від ресурсу

6. External Services

OpenAI API

Призначення: Генерує рецепти на основі інгредієнтів, дієти та історії.

Інтеграція:

- HTTPs REST
- Circuit Breaker для fault tolerance
- Semaphore для обмеження паралельних викликів
- Fallback через Redis, якщо OpenAI недоступний

Google OAuth 2.0

Призначення: Альтернативна аутентифікація користувачів.

Флоу:

1. User → Google Auth
2. Authorization Code → наша backend система

3. Backend → Google OAuth token exchange
4. Backend генерує власний JWT для користувача

Переваги:

- Менше паролів у системі
- Вища безпека
- Швидкий login для користувачів