✅ Project Title: SmartAdX – A Scalable, AI-Driven Ad Serving Platform

🚀 Objective:

Build a **miniature version of an ad serving platform** that mimics core elements of Google Ads — including ad creation, targeting, serving, and analytics — while highlighting your **backend system design**, **data processing**, **full-stack capabilities**, and **ML integration**.

🔧 Key Features:

1. **Ad Management**:
    - Users (advertisers) can create campaigns with targeting criteria (location, interests, age group, budget).
    - Ads can be categorized (text, banner, video URL).
2. **User Simulation**:
    - Simulate a stream of users visiting a page, each with metadata (IP/location, age, interest tags).
    - Use RabbitMQ to simulate live traffic (optional but impressive).
3. **Ad Targeting Engine**:
    - Match the best ad based on targeting rules using a scoring algorithm.
    - Bonus: Use ML to improve ad scoring (e.g., based on CTR prediction).
4. **Analytics Dashboard**:
    - Show ad impressions, clicks, CTR, and budget depletion in real time using something like **React + Chart.js** or **Tableau embedded**.
5. **Architecture**:
    - Use **Spring Boot (Java 17)** for the backend.
    - **PostgreSQL** or **MongoDB** for storing campaigns and metrics.

- **Redis** for caching active campaigns.
- **Kafka** (optional) for ad request stream simulation.
- **Docker & Kubernetes** to show deployment readiness.

6. **Frontend**:
   - Admin UI for advertisers to create and manage ads.
   - User-facing ad surface that displays ads based on targeting.

🧠 Technologies Highlighted:

Skill/Area	Covered In
Distributed Systems	Event streaming, service scaling
Data Processing & Analytics	Campaign metrics, CTR calculation
AI/ML	Ad scoring engine (CTR predictor)
Backend Engineering	Spring Boot microservices
Full-Stack Capability	UI + API + DB + CI/CD
Leadership/Design	Modular architecture + documentation

📌 Suggested Next Services (in order):

1. AdServingService *(Core of any ad platform)*

- **Purpose**: Serve the most relevant ad campaign based on user keywords, impressions, and budget.
- **Responsibilities**:
    - Filter active campaigns (`startDate ≤ now ≤ endDate`)
    - Match by `targetingKeywords`
    - Ensure budget/impressions are not exhausted
    - Rank (e.g., by budget, CTR, recency)
    - Return a single ad (or top N)
- **Why next?** This connects your ad campaigns to real-time requests.

2. AnalyticsService

- **Purpose**: Log impressions, clicks, and conversions.
- **Responsibilities**:
    - Track ad interactions (e.g., store logs in DB or stream to Kafka)
    - Update metrics on campaign entities
    - Possibly calculate CTR, engagement, ROI
- **Why?** You need to measure performance and potentially pause underperforming campaigns.

3. AdvertiserService

- **Purpose**: Manage advertiser accounts and associated campaigns.
- **Responsibilities**:

    - CRUD for advertisers
    - Link campaigns to specific advertisers
- **Why?** It helps build a multi-tenant system.

4. BillingService *(Optional for MVP)*

- **Purpose**: Charge advertisers based on impressions/clicks.
- **Responsibilities**:
    - Track usage
    - Generate invoices
    - Integrate with payment gateway