Data Architecture Challenge

Objective: Design a scalable data storage system for handling diverse data types for different areas within Nomura. Students must choose appropriate databases (RDBMS, NoSQL, Data Warehouse) based on hidden requirements in each scenario.

Project Modules

1. Module 1: Real-Time Market Data Ingestion

- **Scenario:**
    - Data Sources:
        - High-frequency trading feeds (5M records/sec).
        - Social media sentiment streams (semi-structured JSON).
        - Bloomberg/Reuters API data (structured time-series).
    - Requirements:
        - Sub-millisecond latency for trade execution alerts.
        - Storage for 5 years of historical market data.
    - *Hint*: "Some systems prioritize write speed over complex querying."

---

2. Module 2: Client Portfolio Management

- **Scenario:**
    - Data Sources:
        - Structured client transaction logs (ACID compliance required).
        - Portfolio performance reports (PDF/Excel).
        - Regulatory compliance documents (SEC filings).
    - Requirements:
        - GDPR-compliant audit trails for all transactions.
        - Support for complex joins across client portfolios.
    - *Hint*: "Certain data relationships cannot tolerate eventual consistency."

---

3. Module 3: Research & Analytics

- **Scenario:**

- Data Sources:
  - Equity research reports (unstructured text).
  - Analyst ratings (relational tables).
  - Alternative data (satellite imagery, shipping logs).
- Requirements:
  - Fast full-text search across research documents.
  - Integration with Python/R for machine learning.
- *Hint*: "Schema flexibility is critical for evolving data types."

---

4. Module 4: Regulatory Reporting

- **Scenario:**
  - Data Sources:
    - MiFID II transaction reports (batch CSV uploads).
    - FATCA compliance logs (petabyte-scale).
    - Real-time AML (anti-money laundering) alerts.
  - Requirements:
    - Batch processing for overnight regulatory submissions.
    - Sub-10s query response for auditors.
  - *Hint*: "Not all systems handle batch and real-time workloads equally."

---

5. Module 5: Fraud Detection

- **Scenario:**
  - Data Sources:
    - Trade reconciliation logs (relational).
    - Dark web scraping feeds (graph-like relationships).
    - Employee access patterns (time-series).
  - Requirements:
    - Anomaly detection across 1B+ data points.
    - Real-time alerts for suspicious activity.

- *Hint*: "Some data models excel at uncovering hidden connections."

Final Integration Challenge

- **Task:**
  - Combine all modules into a unified architecture.
  - Justify database choices for each module in a single design document.
  - *Constraints*:
    - Budget: "Minimize costs for high-volume data."
    - Latency: "Prioritize real-time systems for trading."

Deliverables

Come up with a data architecture using appropriate technology that supports each of the above scenarios.

1. **Architecture Diagram:**
   - Label databases used in each module (no explanations).
   - Database design, schema, fields and entities as required.
2. **Justification Document:**
   - Explain why each database type was chosen for specific modules.
   - Highlight trade-offs (e.g., cost vs. scalability).
