**Q1. Architecture Style + Justification**

**Selected Architecture Style: Layered Architecture**

The Smart Traffic Signal Automation System follows a **Layered Architecture** where the system is divided into logical layers, each responsible for a specific function and interacting only with adjacent layers.

**A. Justification based on Granularity of Components**

**1. Presentation Layer**

- Contains only components responsible for user interaction and visualization.
- Includes the Simulation Interface and Manual Control Dashboard.
- Displays traffic status, alerts, and system output.
- Does not perform processing, decision-making, or database operations.

---

**2. Application Layer (Control & Decision Granularity)**

- Contains the main control logic of the system.
- Responsible for signal timing, coordination, and system decisions.
- Includes:
    - Traffic Signal Controller
    - Emergency Handling Logic
    - Violation Management Logic
- Processes inputs from the detection layer and generates control actions.
- Does not directly interact with raw database queries.

---

**3. Detection Layer (Sensing & Analysis Granularity)**

- Responsible for collecting and analyzing real-world traffic data.
- Includes the Computer Vision module for vehicle detection and classification.
- Identifies traffic density, emergency vehicles, and rule violations.
- Sends processed information to the application layer for decision-making.
- Works independently of storage mechanisms.

---

**4. Data & Infrastructure Layer (Persistence & Platform Granularity)**

- Responsible for data storage and system infrastructure services.
- Includes:
    - Database connector and query handling
    - MySQL database and system tables
    - Logging and authentication support
- Stores vehicle counts, violations, system logs, and user data.
- Operates independently from UI and control logic.

**B. Why Layered Architecture is Best Choice**

**1. Performance**

- Vision module can directly pass data to the controller.
- No network overhead like microservices.
- Enables fast signal decisions (within seconds).

**2. Maintainability**

- UI, Logic, and Database are independent.
- Database changes do not affect other layers.
- Easier debugging and updates.

**3. Scalability**

- New features can be added in logic layer only.
- Example: pedestrian detection or AI prediction module.
- UI and database remain unchanged.

**4. Reliability**

- Fail-Safe module in logic layer can instantly override signals.
- Ensures system keeps working even if camera fails.

Therefore, Layered Architecture provides the best balance of **performance, maintainability, scalability, and reliability**.

**Why Layered Architecture is the Best Choice (Compared to Others)**

Layered Architecture is the most suitable design for the Smart Traffic Signal Automation System because it provides a balanced solution for performance, maintainability, scalability, and reliability while fitting the real-time nature of the system.

**1. Better Performance than Microservices / SOA**

- Traffic decisions must be taken within seconds.
- In layered architecture, modules communicate through direct function calls.
- Microservices or SOA would require network communication between services, increasing latency.
- Therefore, layered design ensures low response time, which is critical for traffic control.

**2. Easier Maintenance than Monolithic Architecture**

- In monolithic systems, UI, logic, and database are tightly coupled.
- Any small change may affect the whole system.
- In layered architecture, each layer is independent.
- Example: Database schema changes only affect the persistence layer.

Hence, layered architecture provides better separation of concerns than monolithic design.

**3. More Structured than Event-Driven Architecture**

- Event-driven systems are good for distributed asynchronous processing.
- But traffic control requires predictable and controlled execution.
- Layered architecture keeps processing steps clear and sequential.
- This improves debugging and system understanding.

Therefore, layered architecture gives better control and clarity

**Q2. Application Components of the Smart Traffic Signal Automation System**

The system consists of the following components organized according to architectural layers.

**1. Presentation Layer Components**

- **Simulation Interface**
    - Displays the traffic intersection visually.
    - Shows vehicle movement and signal colors in real time.
    - Used for monitoring and demonstration.
- **Manual Control Dashboard**
    - Provides secure login for traffic authorities.
    - Allows manual override of signals.
    - Enables emergency stop and reset functions.
- **System Status Panel**
    - Displays alerts, camera connectivity, and processing status.
    - Shows whether the system is in automatic or fail-safe mode.

**2. Application Layer Components**

- **Traffic Signal Controller**
    - Implements the round-robin signal sequence.
    - Calculates dynamic green time based on traffic density.
    - Maintains coordination between all signals.
- **Emergency Priority Manager**
    - Receives emergency detection signals.
    - Immediately switches the corresponding road to green.
    - Ensures safe passage for ambulances and fire trucks.
- **Violation Management Module**
    - Processes violation alerts from the detection layer.
    - Generates violation events for logging.
    - Supports rule enforcement monitoring.

- **System Decision Coordinator** *(extra depth for marks)*
    - Combines inputs from controller, detection, and safety logic.
    - Prevents conflicting commands.
    - Ensures smooth control flow across modules.

**3. Detection Layer Components**

- **Computer Vision Processor**
    - Captures camera frames using OpenCV.
    - Detects vehicles and classifies type.
    - Calculates traffic density on each road.
- **Emergency Vehicle Detector**
    - Identifies ambulances and fire trucks.
    - Sends priority alerts to the application layer.
- **Violation Detection Engine**
    - Detects wrong-way driving.
    - Monitors overspeeding vehicles.
    - Sends violation data for processing and storage.
- **Health Monitoring Sensor Module** *(good addition for marks)*
    - Checks camera availability and frame quality.
    - Reports detection failure to the application layer.

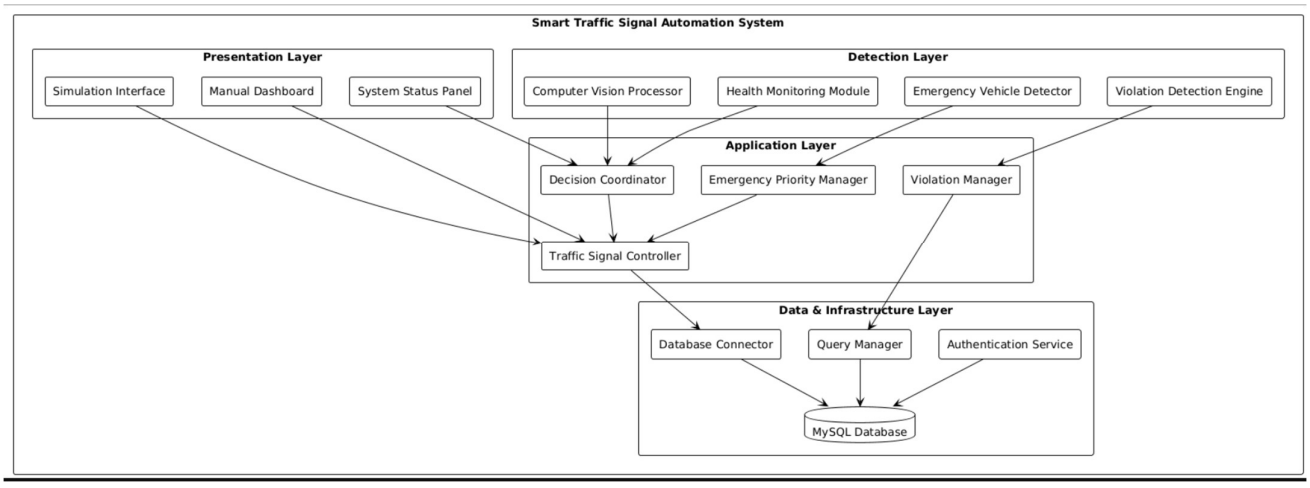
**4. Data & Infrastructure Layer Components**

- **Database Connector**
    - Maintains secure connection between Python system and MySQL database.
    - Handles sessions and transactions.
- **Query Manager**
    - Performs CRUD operations for traffic data.
    - Stores counts, violations, emergency logs, and system events.
- **Authentication Service**

- Verifies dashboard users.
- Protects the system from unauthorized access.

- **MySQL Database**
  - Stores all system records and logs.
- **Core Tables**
  - VehicleCounts
  - ViolationRecords
  - SystemLogs
  - AuthUsers
- **Backup & Logging Storage** *(extra detail = better marks)*
  - Maintains historical data for analysis and auditing.

**It shows component of layers:**



**Communication Rule for Your System**

**Proper flow should be:**

**Presentation → Application → Detection → Data & Infrastructure**