Three-Layer Architecture in MuleSoft API Development Explained

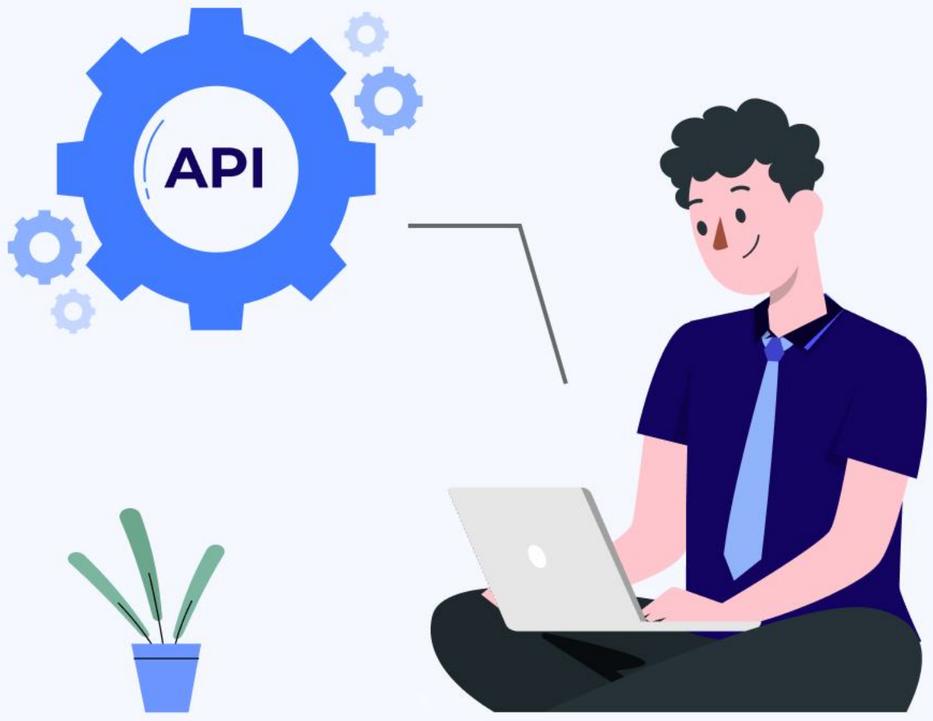
A structured approach to API development, enabling businesses to streamline their integration processes and improve overall system performance.

01

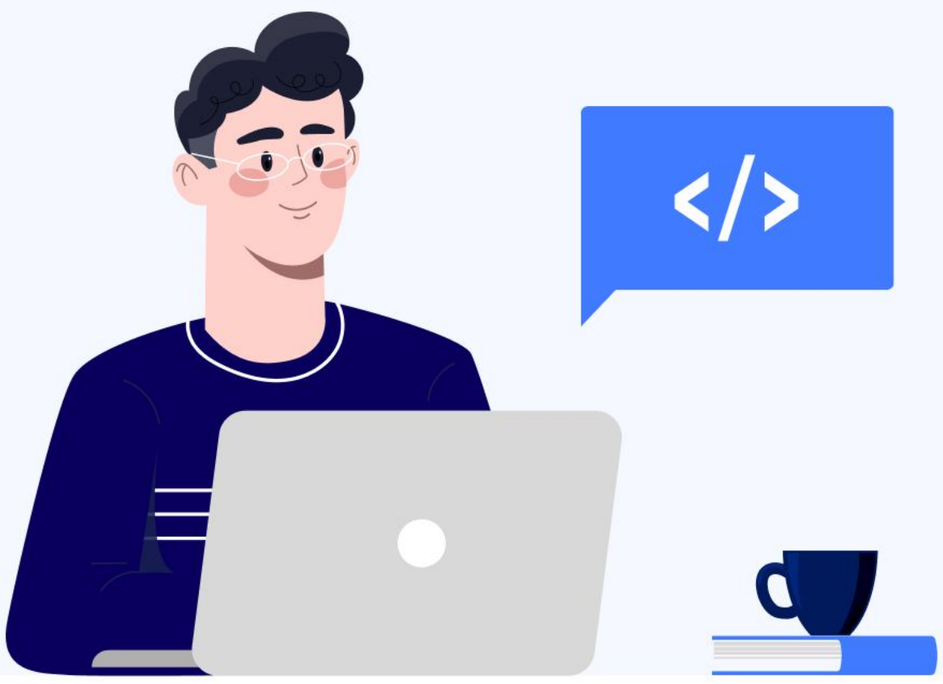
Experience Layer
Purpose: Presents information to users and manages interactions.

Components: User interfaces like web pages, mobile apps, and desktop applications

Key Considerations:
- No orchestration or business logic.
- Implement strong authentication (e.g., OAuth 2.0).
- Ensure extra security against DoS and DDoS attacks.



02



Process Layer
Purpose: Contains core application logic and rules.

Function: Acts as an intermediary, processing requests from the Experience Layer and orchestrating interactions with the System Layer.

Key Considerations:
- Generally not exposed publicly; use API proxies if needed.
- Provide internal URLs for upstream communication.
- Secure with policies like Client ID Enforcement.

System Layer
Purpose: Interacts with data storage systems (databases, external APIs).

Function: Manages data retrieval, storage, and manipulation while abstracting data access.

Key Considerations:
- Not exposed publicly; deploy within a private network.
- Provide internal URLs for communication.
- One System Layer API per backend system.

03