

Take Home Touchline - Multi-Client Customization Challenge

Overview

This is a technical assessment meant to gauge your depth of understanding, strength of software design and development principles. We are interested in the thought process, decision making and judgment.

It is all right to use coding agents for this assignment but it is required that you share your conversation history and process.

If something is not clear please make reasonable assumptions and state your assumptions.

There are no right answers. Good luck!

The Scenario

You're building **Touchline**, a game day operations platform for sports teams. You have 2 customers, each with unique operational requirements based on their sport, venue, and league compliance needs.

The 2 Example Customer Requests

Customer A: LA Galaxy (MLS Soccer)

Need: Department-based task organization

Requirements:

- Tasks tagged with department (Operations, Security, Medical, Concessions)
- Dashboard grouped/filtered by department
- Department field REQUIRED for LA Galaxy only
- Department heads see only their tasks

Business Context:

- 8 departments coordinate on game day
 - Department heads are held accountable for their tasks
 - Execs need rollup metrics by department
-

Customer B: Portland Thorns (NWSL Soccer)

Need: Photo verification for compliance

Requirements:

- Mark specific tasks as "photo required"
- Cannot complete task without uploading photo
- Photos stored securely, viewable in audit log
- Only Portland Thorns sees this feature

Business Context:

- League compliance requirements
- Insurance liability (proof of safety checks)
- Previous incident due to incomplete security sweep

Assignment Structure

Part 1: Architecture Design

Deliverable: Architecture document (markdown, diagrams, or whiteboard)

Required sections:

1. System Architecture
 - How will you structure customization? (code, config, data model)
 - Single codebase or customer-specific forks?
 - What's your extensibility strategy?
2. Data Model
 - Proposed schema (tables, columns, relationships)
 - i. How will you design schema to ensure tenant data isolation / cross-pollution between tenants, esp with RAG systems being built on top.
 - How do you avoid "column sprawl"?
 - How do you maintain type safety?
 - Draw Entity Relationship Diagram or show SQL
3. Trade-offs Analysis
 - Why this approach vs alternatives?
 - What are the costs/benefits?
 - What breaks first at 50 customers? At 100?
 - What would you do differently with more time?

Part 2: Implementation

Deliverable: Working code implementing these customer features

Requirements:

- Fully implement their features end-to-end:
- Code should make it easy to add new features

Technical Stack:

- Your choice