

# Senior Software Engineer Take-Home Challenge

---

## Take-Home Challenge: Multi-Tenant Marketing Dashboard

---

### What We're Evaluating

---

1. **Does it work?** End-to-end functionality, not polish.
2. **Is the architecture clean?** Sensible data model, proper separation of concerns.
3. **Do you understand multi-tenancy?** Correct data isolation between users.

**Time:** 1-3 hours

---

### The Challenge

---

Build a full-stack app where:

1. **Admin uploads CSVs** → parsed and stored in a real database (SQLite, Postgres, etc.)
2. **Users see a dashboard** → filtered to only data they're authorized to see
3. **User switcher** → navigate to a selection screen, pick a user, return to dashboard with their data

## Data Hierarchy

```
Brand (e.g., "Acme")
└─ Company (e.g., "Acme Downtown LLC")
    └─ Locations (campaigns run per-location)
```

## Permission Levels

- **Brand User** — sees all companies and locations under their brand
- **Company User** — sees only their company's locations

## Users to Implement

User	Role	Scope
User 1	Brand User	Acme (all Acme companies)
User 2	Company User	Acme Downtown LLC
User 3	Company User	Competitor Wellness Inc

CSV upload can live anywhere — a separate page, a button in the header, etc. Doesn't need to be tied to a user role.

## Dashboard Tabs

Simple tables for each — no fancy UI needed:

1. **Campaigns** — the campaign data for this user's scope
  2. **Leads** — the lead data for this user's scope
  3. **Sales** — the sales data for this user's scope
  4. **ROI Report** — a joined/aggregated table (e.g., spend vs revenue by company or location)
-

## Deliverables

---

1. **Code repository** (GitHub or zip)
  2. **README** — how to run it, key decisions, trade-offs
- 

## Tech Stack

---

Use whatever you're productive with. If you want to match our stack: Next.js, TypeScript, Postgres/SQLite.

---

## Bonus

---

**Show your AI workflow.** We use Claude Code, Cursor, and similar tools daily. If you used AI-assisted development, share your chat history or a brief writeup. Not required, but a plus.

---

## Tips

---

- Make assumptions and document them
  - Keep it simple — clean and working beats over-engineered
  - Move fast — use whatever tools help you ship
  - Come prepared to explain and extend your project.
- 

## Sample CSVs

---

See the included `sample-data/` folder containing: - `campaigns.csv` — 50 rows of campaign data - `leads.csv` — 50 rows of lead data - `sales.csv` — 50 rows of sales data

---

Good luck!