Orchids Full Stack SWE Takehome - Database Agent

Background

At Orchids, we are building the best agentic AI for building stunning apps and websites. One essential tool that we have implemented to achieve this is a database agent - an agent that can go into a user’s Next.js project and implement any database related features (such as setting up database schemas, implementing database operations, etc) and integrate database functionality into the project’s frontend. In this challenge, you will be building out a simplified version of the database agent.

Implementation Guideline

You will be building a CLI tool (similar to Claude Code) that will sit at the root of a Next.js project. Given a user query for any database related feature, the CLI tool should spin up a “database agent" that will modify project files locally to achieve the user’s request. For the sake of simplicity, please assume that all queries being made will require a database. The end result of a query should be the completed database features along with a successful integration into the frontend of the project. Utilize existing AI models via API (i.e. from OpenAI, Anthropic, Google, etc.). If you need an API key let us know and we’re happy to provide.

Codebase (Link to template)

You will be working off a template codebase, which is a Next.js + TypeScript project containing a clone of Spotify built with Orchids. This project only contains the frontend for the Spotify project so the database agent you will be building will need to take into context the current setup of the project and be able to add relevant database/backend functionalities. You can assume all queries being made to the agent will be related to the Spotify project.

Database framework

A core piece of the database agent is the database framework in which the agent operates. This dictates largely how the agent will work. For this part, we leave it at your own discretion and you are free to choose whatever framework that you think might work best (such as types of databases - relational/document or providers such as Supabase/MongoDB) but we highly recommend using Drizzle, a headless TypeScript ORM. Drizzle makes it easy to connect to any database, define database schemas, run migration commands, write database operations, etc - anything you would need to manage a relational database.

Pipeline

1. **CLI tool**: Create a script that will spin up a CLI tool. The CLI tool should display the current process of the agent (such as what it is thinking, what file it is editing, etc)

2. **Database agent**: Gather enough context on the project and implement database features. It should be able to write database schema setup files, run migration scripts, implement database operations, set up API endpoints, etc. Then, it should be able to integrate the newly created database features into the UI/UX of the site.

Test Queries:

1. “Can you store the recently played songs in a table”
    a. The agent should create the table, populate it, and also create a route to fetch information from that table.
    b. *BONUS*: integrate the route into the existing code so that the site actually fetches the data and properly displays it on the frontend.
2. “Can you store the ‘Made for you’ and ‘Popular albums’ in a table”
    a. The agent should create the tables, populate them, and also create a route to fetch information from those tables.
    b. *BONUS*: integrate the route into the existing code so that the site actually fetches the data and properly displays it on the frontend.

Deliverable

There are 2 things needed for a submission:

- **A Github link** of the codebase you have made changes to. If it requires extra set up steps, please include it in a README.md file
- **A Youtube link** to a short video of you walking through the features implemented and the agent executing on the two test queries.
- **SUBMIT HERE**

DUE DATE: 7/20 at 11:59 PM PST

***Note**:* If you run into compatibility issues with React 19 when setting up the project, feel free to downgrade to React 18 or run `npm install –legacy-peer-deps`

That is all from us! Thank you for your interest in Orchids and please let us know if you have any questions by emailing us at kevinlu@orchids.app