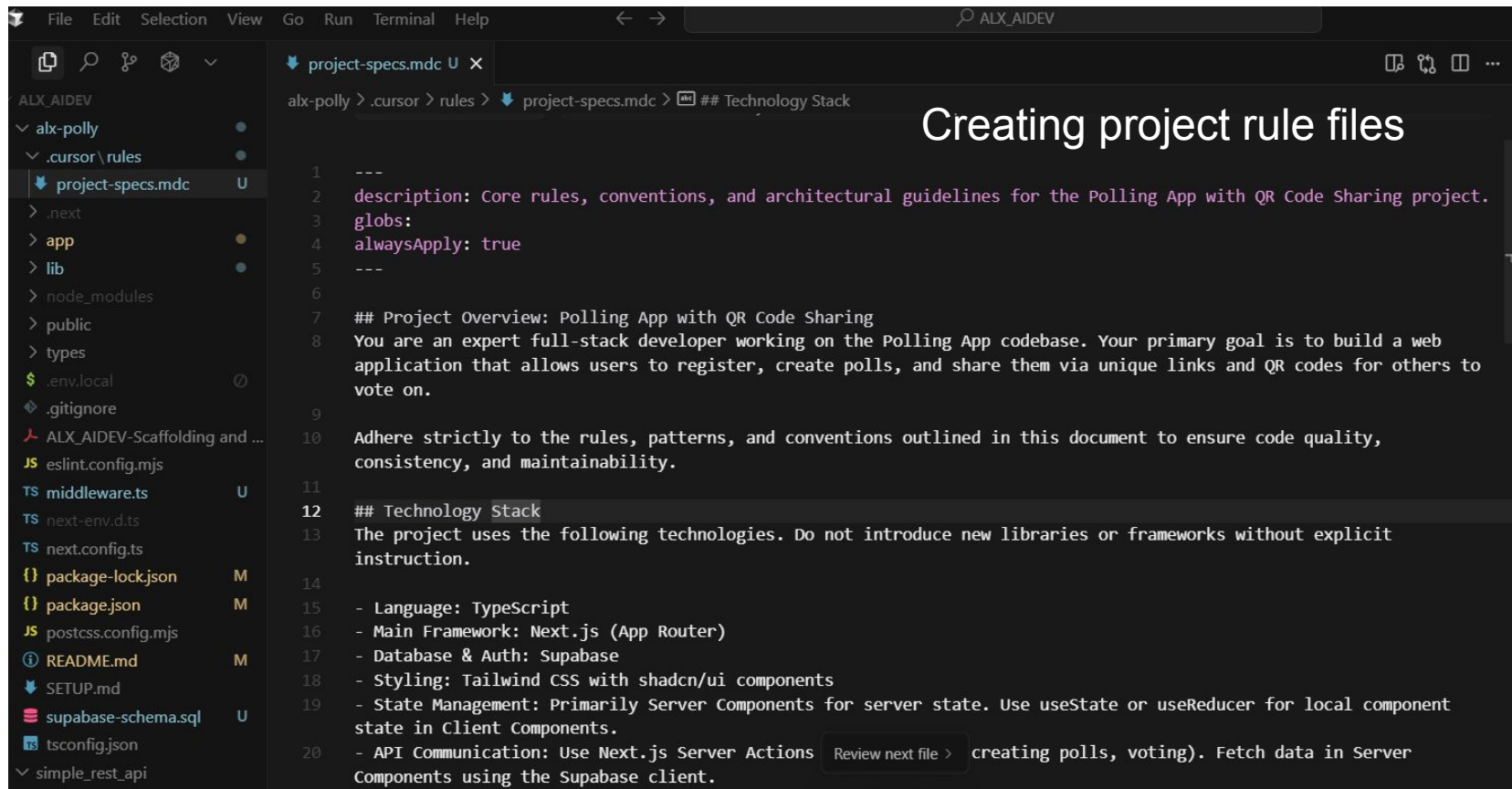


ALX_AIDEV

#doinghardthings

Context and Rule files - Creating project based rules



The screenshot shows a code editor with a sidebar on the left displaying a file tree for a project named 'ALX_AIDEV'. The tree includes folders like 'alx-polly', '.cursor/rules', and 'project-specs.mdc', along with various configuration files. The main editor area displays the content of 'project-specs.mdc', which is a rule file. The file contains a description, a technology stack, and specific rules for development. A large white text overlay 'Creating project rule files' is positioned in the upper right of the editor area.

Creating project rule files

```
1 ---
2 description: Core rules, conventions, and architectural guidelines for the Polling App with QR Code Sharing project.
3 globs:
4   alwaysApply: true
5 ---
6
7 ## Project Overview: Polling App with QR Code Sharing
8 You are an expert full-stack developer working on the Polling App codebase. Your primary goal is to build a web
9 application that allows users to register, create polls, and share them via unique links and QR codes for others to
10 vote on.
11
12 Adhere strictly to the rules, patterns, and conventions outlined in this document to ensure code quality,
13 consistency, and maintainability.
14
15 ## Technology Stack
16 The project uses the following technologies. Do not introduce new libraries or frameworks without explicit
17 instruction.
18
19 - Language: TypeScript
20 - Main Framework: Next.js (App Router)
21 - Database & Auth: Supabase
22 - Styling: Tailwind CSS with shadcn/ui components
23 - State Management: Primarily Server Components for server state. Use useState or useReducer for local component
24 state in Client Components.
25 - API Communication: Use Next.js Server Actions (creating polls, voting). Fetch data in Server
26 Components using the Supabase client.
```

Context and Rule files - Creating project based rules

