To get LM Studio, ngrok, and CodeLlama working with Cursor, you need to bridge the gap between Cursor's cloud-based model verification and your local machine. Because Cursor does not currently support localhost URLs directly for privacy and security reasons, ngrok acts as the necessary "public" tunnel.

Step 1: Set up the model in LM Studio
Open LM Studio and search for CodeLlama. Download the version that fits your hardware (usually the Q4_K_M GGUF variant is a good balance).
Go to the Developer/Local Server tab (the double-bracket icon <->).
Select your CodeLlama model from the dropdown at the top.
In the Server Settings on the right:
Ensure the port is set to 1234 (default).
Crucial: Toggle Cross-Origin Resource Sharing (CORS) to ON.
Click Start Server.

Step 2: Create the Tunnel with ngrok
Since Cursor requires a public HTTPS endpoint, you'll use ngrok to expose your LM Studio server.

Open your terminal or command prompt.

Run the following command:
ngrok http 1234

ngrok will generate a "Forwarding" URL (e.g., https://random-id.ngrok-free.app). Copy this URL.

Step 3: Configure Cursor
Open Cursor and go to Settings (gear icon in the top right) > Models.
Scroll down to the OpenAI API section.
Click "Override OpenAI Base URL" and paste your ngrok URL, appending /v1 to the end.
Example: https://your-unique-id.ngrok-free.app/v1
In the API Key field, enter any placeholder text (e.g., lm-studio or 1234). LM Studio doesn't require a real key, but Cursor needs the field filled.
Under Models, click "Add Model" and type the exact name of the model as it appears in LM Studio (usually something like codellama-7b-instruct).
Toggle off other models (GPT-4, etc.) to ensure Cursor defaults to your local setup.
Troubleshooting Tips
Model Name Match: If Cursor says "Model not found," ensure the name you added in Cursor matches the "Model Identifier" shown in the LM Studio server logs.
Context Length: CodeLlama has a specific context window. In LM Studio, you can adjust the "Context Overflow" settings if you notice the model "forgetting" parts of your code.
Speed: Running through ngrok adds a tiny bit of latency, but the actual processing is still happening on your GPU/CPU.

Would you like me to help you optimize the n-gpu-layers setting in LM Studio to ensure CodeLlama runs faster on your specific hardware?

Running Local LLM in Cursor Video

This video provides a quick visual walkthrough of how to link a local model to Cursor using a tunneling service like ngrok.