# SPORTS INTELLIGENCE ASSISTANT: OVERVIEW

## BUILDING A SPORTS-FOCUSED AI USING OPENAI'S MODELS

1. Use OpenAI's Existing Model (like GPT-4o or GPT-4.1)  
These models already understand language, reasoning, context, and general knowledge. No need to start from zero.  
  
2. Inject Your Domain Knowledge (sports, fan behavior, trends, etc.)  
- Feed it custom documents, research, and writing you've produced  
- Use tools like embeddings + vector databases to let it "recall" relevant info on demand  
- Add custom prompts or fine-tuned instructions so the model speaks with your voice, expertise, and priorities  
  
3. Wrap it in a Custom Interface  
Create a branded tool or chat assistant for:  
- Client-facing work  
- Internal strategy sessions  
- Fan sentiment analysis  
- Storytelling generation

## WHAT THIS LETS YOU DO

- Create a sports-savvy AI assistant trained on your style of thinking  
- Analyze game data, player sentiment, social media, cultural moments, brand tie-ins, and more  
- Let clients chat with an "insight engine" that reflects your firm’s POV  
- Build a tool that "gets" fandom like you do — not just stats, but emotion, behavior, loyalty, and psychology

## WHAT YOU'D NEED

- Access to the OpenAI API  
- A developer (or partner to guide the structure)  
- Your archive: PDFs, decks, insights, audio, articles  
- Optional: vector database like Pinecone or Weaviate for context memory

## PROJECT: SPORTS INTELLIGENCE ASSISTANT

A custom AI trained on fan culture, sports behavior, and your strategic POV — built for client-facing use, internal insights, or pitch augmentation.

## CORE CAPABILITIES

- Answers questions with a fan- and brand-savvy POV  
- Analyzes trends, social sentiment, or past campaigns  
- Generates content in your tone (headlines, insights, campaign copy)  
- Assists with decks, creative ideas, and strategy prompts  
- Pulls from your proprietary material for customized output

## HOW LONG & HOW HARD?

If you're a strategist, not a dev:  
You'll need one smart developer (or team) to build the backend + integrations. You steer the content and training.  
  
Timeline:  
Phase 1: Scope & Structure (1–2 weeks)  
Phase 2: MVP Build (2–4 weeks)  
Phase 3: Testing & Iteration (2–3 weeks)  
Phase 4: Optional Polish (2–4 weeks)  
  
Total: 6–12 weeks for a polished product  
(Much faster if it's internal only or MVP-style)

## COST ESTIMATE (BALLPARK)

Developer / Team: $5K–20K+  
OpenAI API Usage: $50–$500+/mo  
Vector Database: Free–$100/mo  
Hosting / UI Tools: Minimal  
  
Scrappy MVP possible for $5K–10K + your time

## YOUR ADVANTAGE

- 10+ years of proprietary material  
- A clear, recognizable POV and tone  
- Proven client-facing experience  
- An actual use case + market demand  
  
You're not starting from scratch. You're turning your brain, voice, and archive into a usable interface.