Product Requirements Document (PRD): **Video Generation Validator (1-Week MVP)**

1) Problem & Goal

AI-generated videos often deviate from their prompts (missing objects, wrong actions, visual artifacts). Manual QC is slow and subjective. Goal: ship a minimal web tool that grades a ≤10s MP4 against its original prompt, surfaces top issues, and suggests an improved prompt the user can iterate with.

2) MVP Scope

- **Input**
  - Upload: single MP4 (≤10s)
  - Prompt: text field (original generation prompt)
  - Threshold: confidence slider 0-100%
- **Grader Engine (TwelveLabs)**
  - Marengo (Search + Embed): semantic video search and embeddings to compare prompt intent vs. content
  - Pegasus (Analyze): generate structured descriptions/summaries/chapters/highlights to detect inconsistencies and support issue explanations + prompt rewrite
- **Results**
  - Score Card: overall confidence with pass/fail vs threshold

    - Key Issues: up to 3 prominent problems (e.g., “Missing red car”)
    - Improved Prompt: concise suggestion that increases specificity and reduces artifacts
    - Re-run: copy improved prompt to clipboard (manual iteration for MVP)

3) TwelveLabs API Usage

- Marengo
    - Search API: `POST /v1.3/search` for any-to-video (text or image query) against an index
    - Embed API (optional path): `POST /v1.3/embed/tasks` → retrieve segment embeddings for custom scoring logic
- Pegasus
    - Analyze/Generate text from video: summaries, chapters, highlights, or open-ended text to extract scene/objects/actions and narrative consistency

Typical flow options (pick one or combine):

- Option A (Fastest to MVP): Upload video to an index (Tasks → Ready) with Marengo enabled → Use Search with the full prompt (and optionally sub-queries) to gauge alignment → Use Pegasus to produce a concise analysis → Fuse into a single confidence score + issues + prompt rewrite.
- Option B (More control): Use Embed API to retrieve segment embeddings (Marengo 2.7) and compute custom similarity vs. prompt-embedded text chunks; use Pegasus outputs as auxiliary signals for issue labeling and prompt rewriting.

4) Scoring & Issue Detection (guidelines, not strict)

- Signals to consider (choose subset that fits timebox):
    - Prompt alignment: Marengo search relevance for the full prompt (average top-K scores)
    - Coverage checks: break the prompt into entities/actions/attributes; search each sub-query to verify presence

    - Narrative/visual summary: Pegasus summaries/chapters as cross-check; detect contradictions (e.g., night vs day)
    - Simple artifact heuristics (stretch): detect black/frozen frames or duplicates with ffmpeg/OpenCV
- Confidence (0–100): combine selected signals via a simple weighting; calibrate on a few sample pairs so scores match human intuition
- Key issues: rank the weakest matches and clearest contradictions; cap at 3 for clarity
- Prompt rewrite: emphasize specificity (counts, colors, actions, camera/view, time-of-day) and add negatives (e.g., "no extra limbs, no distortions")

5) UX Flow

1. User uploads MP4, enters original generation prompt, adjusts threshold, clicks "Grade"
2. Backend processes the video (upload/index if needed), runs Marengo search and Pegasus analysis
3. Tool returns: confidence score + pass/fail, top issues, improved prompt, and "Copy & re-run" button

6) Architecture (keep flexible)

- Frontend: Next.js + TypeScript (single page: upload, prompt, slider, results)
- Backend: Next.js API routes or a minimal Python/FastAPI service; no DB required

7) Milestones (1 week)

- Day 1: UI scaffold (upload/prompt/threshold), file validation
- Day 2–3: TwelveLabs integration (Option A preferred for speed): index/upload flow, Search API and Analyze API usage
- Day 4: Scoring fusion + top-3 issues; improved prompt generation; progress + error states
- Day 5: Calibration on 5–7 examples; polish + deploy; quick demo script

8) Acceptance Criteria

- Processes ≤10s MP4 and returns results within 10s
- Produces an overall confidence with pass/fail and up to 3 actionable issues
- Improved prompt is materially more specific and addresses flagged gaps