HACKATHON BATTLE PLAN

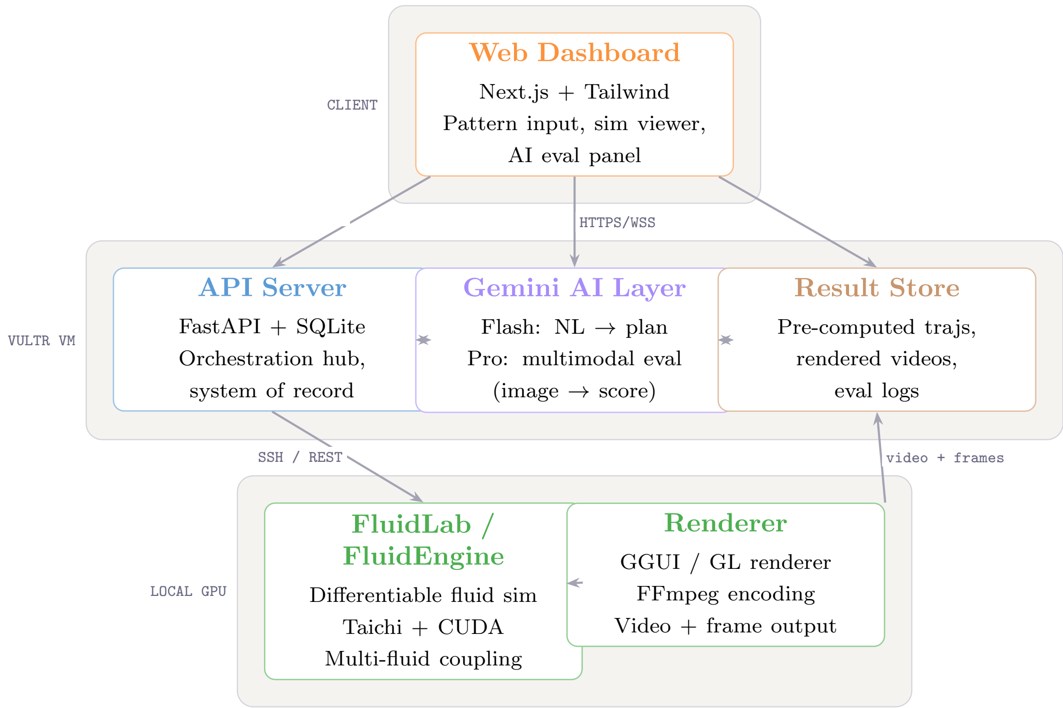
LatteBot

Track 2 --- Sim-to-Real Training Pipelines

Sim-to-real training platform for precision fluid manipulation, powered by differentiable physics & multimodal AI evaluation

lablab.ai × Surge — Launch & Fund Your Startup Edition 1 — Feb 6–15, 2026

01 System Architecture



02 Tech Stack

Simulation Engine

FluidLab	Fluid manipulation env
FluidEngine	Differentiable physics
Taichi	GPU-accelerated compute
exp_latteart.yaml	Built-in latte art task

Backend (Vultr)

FastAPI	REST + WebSocket server
SQLite	Jobs, results, metadata
Nginx	Reverse proxy + SSL

AI / Intelligence

Gemini 3 Flash	NL → pour parameters
Gemini 3 Pro	Multimodal eval + critique
Diff. Optimization	Gradient-based traj. opt.

Frontend

Next.js 14	React framework
Tailwind CSS	Utility-first styling
WebSocket client	Real-time sim updates

Infrastructure

Vultr VM	Backend host (mandatory)
Local NVIDIA GPU	FluidLab compute
SSH Tunnel	Vultr ↔ GPU bridge
GitHub	Code + documentation

Mandatory Deliverables

Public Web URL	App hosted on Vultr
GitHub Repo	Code + setup docs
Demo Video	Full flow recording
X Post	@lablabai + @Surgexyz_
Pitch Deck	Slide presentation

03 Request Lifecycle (End-to-End Flow)

1. **User Input**
User types `"Make a rosetta"` or selects a preset pattern in the web dashboard. Request sent to Vultr backend via HTTPS.

2. **Gemini Planning (Flash)**
Backend calls Gemini 3 Flash to parse natural language into structured pour specification:
`{ pattern: "rosetta", pour_height: 0.08, oscillation_freq: 3.2, flow_rate: 0.6, pull_through: true }`

3. **Sim Job Dispatch**
Backend checks for pre-computed trajectory cache. Cache hit → skip to step 5. Cache miss → dispatch parameters to local GPU worker via SSH tunnel.

4. **Differentiable Trajectory Optimization**
FluidEngine computes gradients *through* the multi-fluid simulation. Optimizes pitcher trajectory (XY position, height, tilt angle, flow rate over time) to minimize distance between resulting milk particle distribution and target pattern. **Gradient-based — not brute-force RL.** This is the core technical innovation.

5. **Simulation Render**
Optimized trajectory executed in FluidEngine. Multi-fluid interaction rendered (coffee surface + milk particles). GGUI/GL renderer outputs video frames + top-down final cup capture. Results uploaded to Vultr.

6. **Gemini Evaluation (Pro — Multimodal)**
Top-down cup image sent to Gemini 3 Pro for visual evaluation:
`{ score: 8.2, feedback: "Good symmetry. Milk density uneven on left. Reduce flow_rate mid-pour." }`

7. **Closed-Loop Feedback**
Dashboard updates via WebSocket: sim replay video, AI evaluation card, trajectory visualization. User can accept result, adjust parameters based on Gemini feedback, or re-run. Complete training loop demonstrated.

04 Why Track 2

Track 2 asks for *"systems that use simulation as the primary environment for training, testing, or validating robotic behaviors before deployment on real robots."*

LatteBot is exactly this:

- **Training:** Differentiable physics enables gradient-based trajectory optimization — the simulation *is* the training environment.
- **Evaluation:** Gemini Pro serves as an automated benchmarking framework, scoring pattern accuracy, symmetry, and quality from rendered images.
- **Validation:** The FluidLab paper (ICLR 2023 Spotlight) already demonstrated successful sim-to-real transfer of optimized trajectories. This isn't theoretical — there's published evidence the pipeline works.
- **Scalability:** The platform generalizes beyond latte art to any precision fluid manipulation task, which is exactly the "scales beyond a single demo" requirement in the track description.

05 Risks & Mitigations

Trajectory optimization is slow

Differentiable optimization through fluid sim can take minutes–hours per pattern.

Mitigation: Pre-compute 3–4 trajectories offline. Serve cached results during demo. Live optimization is a bonus, not a requirement.

GPU ↔ Vultr connection drops

SSH tunnel between local GPU and Vultr is fragile. Could break during demo.

Mitigation: Pre-render all demo videos and store on Vultr. App works in "replay mode" even if GPU is offline. Use `autossh` for tunnel persistence.

FluidLab setup / dependency issues

Taichi, CUDA, and Python version pinning can be painful to get working.

Mitigation: Day 1 priority. Use provided `environment.yml`. Fall back to Docker if conda fails. Don't spend more than 1 day on setup.

Latte art visuals look bad

Particle-based rendering may not be visually impressive enough for judges.

Mitigation: Use GL renderer (better than GGUI). Colorize milk particles clearly. Top-down view hides 3D limitations. Emphasize pattern accuracy over photorealism.

06 Submission Checklist

☐	Backend deployed on Vultr VM as central orchestrator	MUST
☐	Public web URL accessible via browser	MUST
☐	Gemini used for planning + multimodal evaluation	MUST
☐	GitHub repo with setup docs and architecture README	MUST
☐	Demo video posted on X tagging @lablabai + @Surgexyz_	MUST
☐	X post link pasted in official submission form	MUST
☐	Slide presentation / pitch deck	MUST
☐	FluidLab sim running with real multi-fluid physics	CORE
☐	Differentiable trajectory optimization producing visible results	CORE
☐	Closed loop: NL → Gemini plan → sim → Gemini eval → feedback	CORE
☐	Multiple patterns supported (heart, rosetta, tulip)	NICE
☐	REST API documented for external integration	NICE
☐	Trajectory comparison view (before/after optimization)	NICE

07 Pitch Framing

The One-Liner

"LatteBot is a sim-to-real training platform for precision fluid manipulation — using differentiable physics to optimize robotic pour trajectories and multimodal AI to evaluate results, all before touching real hardware."

Problem

Training robotic baristas costs real milk, real espresso, and real time for every failed pour. There's no fast, safe way to iterate on fluid manipulation policies. Existing sim tools don't support multi-fluid coupling or provide gradients for optimization.

Solution

A simulation-first training pipeline where robots learn precision pouring through differentiable physics. Trajectories are optimized via gradients (not brute-force RL), evaluated by multimodal AI (Gemini), and validated before real deployment. FluidLab (ICLR 2023 Spotlight) already proved sim-to-real works.

Expansion Beyond Latte Art

The platform generalizes to any precision pouring domain: pharmaceutical dispensing, chemical mixing, paint application, food plating, lab automation. Any industry where fluid manipulation quality matters and real-world iteration is expensive or dangerous.

Judging Criteria Alignment

Technology	Differentiable physics (FluidEngine) + multimodal AI (Gemini Flash/Pro) deeply integrated, not superficial wrappers.
Presentation	Memorable demo — type a pattern name, watch a robot pour it in sim, get AI feedback.
Business Value	Real market (robotic coffee $2B+), clear path to generalization, published sim-to-real evidence.
Originality	No other team will combine differentiable fluid sim with multimodal AI evaluation in a closed loop.

--- end of document ---