SOHAM KOTKAR — GURUKUL FRONTEND OWNER
Experience Integrity • Stability • Soul Aligned Interface

———

A. READ THIS FIRST (MANDATORY CONTEXT)

You now own the Gurukul Frontend.

Your responsibility:
• Convert current frontend from “functional demo” → clean, disciplined, production-grade learning interface
• Build around stable backend (Yashika) without forcing backend changes
• Make Gurukul feel human, calm, guided — not like a random edtech app
• Ensure full learning flow continuity (not page-to-page features)
• Prepare UI to support EMS Governance (Teacher/Parent/Admin rails)

You are NOT allowed to:
Break working backend integrations
Introduce random UI experiments
Ship unstable half-baked UX decisions
Ignore polish/documentation

This is not a “UI playground”.
This is the face of Gurukul.

Outcome must represent:
A refined, stable, intuitive, soul-aligned learning platform interface that can be demoed proudly.

———

B. INTEGRATION BLOCK

You work in locked sync with:

Yashika
Backend Owner — Stability • Reliability • Soul + EMS Ready Backend
Provides stable, versioned, reliable API & governance rails

Rishabh Yadav
EMS Core / Institutional Layer
Provides structure for institutions, roles, tenant logic

Nikhil Pawar
Permissions + Role Logic + Behavior Enforcement
Provides safe visibility models for Parent / Teacher / Admin

Rukayya
Adaptive UI + Calm Learning Intelligence Layer (non-blocking support)
Helps UI feel guided, human and emotionally steady

Akash (Consult if required)
Architecture sanity + integration correctness

———

C. TIMELINE

Total Duration: 10 Days
Outcome MUST be production-feeling + demo ready.

———

D. DAY-BY-DAY EXECUTION PLAN

DAY 1 — UX Audit + Truth Check
• Review current frontend honestly
• Identify confusing UX
• Identify weak design spots
• Identify “fake feeling / mechanical” areas
• Map full learning journey
(Onboarding → Learning → Summarizer → Questions → Reflection → Progress)

Output:
docs/UI_TRUTH_REPORT.md

———

DAY 2 — UI Contract Freeze
• Align with Yashika API Freeze
• Standardize component responsibility
• Remove unstable UI hacks
• Define clear UI states:
loading / success / failure / neutral / reflective

Output:

Frontend shall NOT break when backend stabilizes.

———

DAY 3 — Real Experience Layer
• Fix navigation flow
• Ensure student journey continuity
• Reduce friction
• Remove random feeling transitions
• Add structure, calm rhythm

This is where Gurukul starts feeling alive.

———

DAY 4 — Stability Pass
• Handle backend failures gracefully
• Prevent UI crashes
• Proper loaders
• Proper retry
• Proper feedback to user

NO “nothing happens” states allowed.

———

DAY 5 — Governance Friendly UI
Prepare rails (hooks only now) for:
• Teacher View scope
• Parent View scope
• Admin school/institution view scope

DO NOT BUILD full dashboards now.
Just UI readiness + structural correctness.

———

DAY 6 — Performance & Cleanup
• Remove junk UI code
• Reduce unnecessary rerenders
• Remove console spam logs
• Optimize state handling

UI must feel confident and reliable.

———

DAY 7 — Deep Integration Day with Yashika
Full pairing session
Fix friction
Make system behave like one product
Frontend must not request backend to change its truth

———

DAY 8 — Soul Validation Day
Check:
• Is it calm?
• Does it guide or overwhelm?
• Does it respect learning?
• Does it feel warm, human, real?
• Does it support reflection?
• Does it encourage growth without stress?

Fix anything that breaks that spirit.

———

DAY 9 — Documentation + Readiness Note

Create:
/docs/FINAL_FRONTEND_READINESS_REPORT.md

Include:
• What works
• What breaks
• Risks
• UX reasoning
• Implementation truth
• Future safe upgrade direction

No bragging.
Only reality.

———

DAY 10 — Lock + Demo Polish
• Final refinement

• Final aesthetics
• Stability check
• Demo rehearsal mindset

———

E. LEARNING KITS (MANDATORY)

1 Study / Watch / Read
Search & Study:
“Production-grade React UI discipline”
“Calm UI / Calm Technology Design”
“Education UX — human-first interfaces”
“Error tolerant UI design”
“Frontend resiliency patterns”

2 Theory Concepts
• UX rhythm
• Cognitive load
• Trustful UI behavior
• Learning system interface design

3 LLM Learning Prompts
Ask:
“Design a learning platform UI that feels supportive instead of transactional.”
“How to build UI that doesn’t panic when backend fails?”
“What makes educational software feel human rather than tool-like?”

———

F. DELIVERABLES

1 Stable usable frontend
2 No API breaking dependencies
3 True learning journey experience
4 EMS hooks prepared
5 Soul-aligned interface
6 Fully integrated with backend
7 Polished user-facing behavior
8 Demo ready
9 Documentation + Truth note