

# Chat, I'm cooked.

How one reel at 1:30 a.m. took my startup from a few  
hundred users to 18,000, at fifteen.

KANAK RAJ

# Chat, I'm cooked.

How one reel at 1:30 a.m. took my startup from a few hundred users to 18,000, at fifteen.

Kanak Raj

**First edition, 2026.** Published by Avenoric, [avenoric.tech](https://avenoric.tech).

**Every number in this book is real.** Where I don't know something yet, I say so. Numbers are as of October 2026.

**This copy is yours.** It's licensed to you for personal use. Please don't share it, re-upload it or sell it. The only official place to get it is **[avenoric.tech/store](https://avenoric.tech/store)**. If you got it anywhere else, it may have been altered. Full terms: [avenoric.tech/store/license](https://avenoric.tech/store/license).

Nothing in this book is financial, legal or medical advice. It's what happened to me, and what I took from it.

© 2026 Avenoric. All rights reserved.

*For everyone who was told  
to sleep first  
and build later.*

## CONTENTS

- **Prologue: 1:30 a.m.**
- 

### PART ONE · BEFORE ANYONE WAS WATCHING

- 1 One prompt**
- 

- 2 404**
- 

- 3 Nervous as hell**
- 

- 4 Seventeen hours**
- 

- 5 The login button**
- 

### PART TWO · THE VIEWS

- 6 A hundred reels**
- 

- 7 The week before**
- 

- 8 Fifty thousand**
- 

- 9 ₹21,800**
- 

### PART THREE · BEING SEEN

- 10 The youngest in the room**
- 

- 11 Explaining it at home**
- 

- 12 Distribution is the moat**
- 

### PART FOUR · WHAT IT COST, AND WHAT IT TAUGHT ME

- 13 What I gave up**
- 

- 14 What I believe at fifteen**
- 

- 15 Dear fourteen-year-old me**
- 

### AT THE BACK

- **The whole book on one page**
- 

- **The playbook**
- 

- **Kanak Raj**
-

## PROLOGUE

# 1:30 a.m.

It was half past one in the morning, and my family had been telling me to go to sleep for hours.

I had a reel ready. It was about a minute long, just me talking to the camera about how I'd used ExplainMate to study the night before an exam, and it had taken me about two and a half hours to make. I nearly didn't post it.

That week had been bad. A few days earlier I had come very close to deleting everything. ExplainMate, the study app I had been building since I was fourteen, had somewhere between 450 and 600 users, and almost every one of the hundred-odd reels I'd posted had stalled at one or two thousand views. At half past one, with everyone at home asking me to switch the light off, one more reel felt pointless. I knew how it would go.

I posted it anyway. It opened with three words:

| “Chat, I’m cooked.”

Half an hour later it had a thousand views, which had never happened before. On a good day, a thousand was what my reels managed in twenty-four hours. Before I fell asleep I set myself a target of fifty thousand and told myself I'd be happy with that.

When I woke up at nine, it was sitting at fifty thousand.

1K in 30 min

50K by 9 a.m.

600K by night

1M in 48 hours

Over the next three weeks, 18,000 people signed up for ExplainMate. Between that reel and the copies I put out as trial reels, the views went past 2.5 million. I was fifteen, in Class 10, with board exams a few months away.

This book is about how I got to that night and what happened after it. I've tried to keep in the parts that usually get cut: the folder that disappeared and the hour I spent crying over it, eleven rebuilds, weekends that ran to seventeen hours, a login button that was broken just as people finally started turning up, a server bill I wasn't ready for, my parents, and a newspaper my family doesn't even buy.

All the numbers are real. Where I don't know something yet, I've said so.

PART ONE

# Before anyone was watching

A folder that disappeared, eleven rebuilds, a launch night,  
and a bug that cost me thousands of people.

# One prompt

I've been coding since 2020.

Before ExplainMate I had done agency work and some freelancing, so I'd built things for other people and been paid for them. What I had never done was put a product of my own in front of real users. That's production: if something breaks, a stranger sees it, and the stranger doesn't care that you're fourteen.

The idea came from something every Indian student knows, which is rote learning. You memorise an answer word for word, write it out in the exam and forget it a week later, without ever really understanding it. So much of school is built around getting through the next test rather than knowing the subject.

I didn't need to do research to find that problem, because I sit in it every day. ExplainMate started from one belief: students should understand and remember what they study instead of mugging it up.

## The first version

The first version had a single feature, a chat, but it didn't answer you with one paragraph. Ask it something and it gave you the rough work, the scribbled working you'd do in the margin, and the

fair work, the clean answer you'd write in the exam. After that came an explanation, some examples and a mind map, all from one question.

With an ordinary AI chatbot, getting all of that takes five prompts or more. With ExplainMate it took one. It sounds like a small thing, but it was the reason people liked it.

I built it on my own. When you're young and building for people your own age, you don't have to imagine the problem. You already know which apps students open at eleven at night, which ones they delete, and what would actually help. A team of adults making an edtech product has to guess at all of that.

---

#### WHAT I TOOK FROM IT

You don't need to be older to build. You need to be closer to the problem than the people who are older.

# 404

Late in 2025, before ExplainMate was public, I had been building it for one or two months.

One day I deployed to production on Vercel, as I'd done plenty of times before, and opened the site. It showed a Next.js 404: *This page could not be found.*

Broken pages happen, so I went to the repository to find out why. The main app folder wasn't there. It had been deleted, and to this day I'm not sure how. It might have been Windows, or something I ran without noticing. All I know is that it was there, and then it wasn't.

I spent two days trying to get it back, digging through git history and anything else I could think of. But I hadn't been using git properly, so there was nothing to recover. I cried for about an hour, and I'm not embarrassed to say so. One or two months of work had gone, and I hadn't backed any of it up.

Two days later, I started again.

• • •

You'd think rebuilding something you've already built would be easy, since you know what it's meant to look like. It wasn't. The first rebuild didn't come out right, and neither did the second. By

the sixth I was completely fed up, and the same sentence kept going round in my head: *this isn't working*.

Then one thought pushed everything else aside. I would keep working, however long it took, until I was back where I had been. It wasn't a plan so much as a decision to stop asking whether it would work.

The eleventh version worked, and it was far better than the one I'd lost. I think I understand why. The first version was me working out what I was building. By the eleventh I had run into every hard problem ten times, so I knew which parts mattered and I wasn't guessing any more. Losing the code felt like the worst thing that could happen to me. It ended up being the reason the product got good.

There's a practical lesson in it as well: learn git before you need it. I found out what it's for by losing two months of work.

---

#### WHAT I TOOK FROM IT

Rebuilding isn't starting over. It's building again with everything you learned the first time.

# Nervous as hell

ExplainMate launched on 28 November 2025. I was at home, and I was nervous as hell.

It was the first time my code was going in front of real people, and the app was still throwing schema errors. After the 404 I had a new fear as well: what if I deleted something by accident again? I still didn't know git well.

So I tested it, hundreds of times. I'd ask a question, check the answer, fix whatever was wrong and start again, until there was nothing left to check. Then I deployed it.

The first thing I did afterwards was call a friend and ask him to try it. He did, and what he said was, "Bro, it's so good." He'd typed one question and got the explanation, the examples and the working together, which was the whole point.

Mostly I felt relief. After the lost folder, eleven versions and all that testing, it worked, and someone other than me had used it and liked it. For the first time, ExplainMate existed outside my own head.

---

## WHAT I TOOK FROM IT

Your first real user is the first real test. Mine was a friend on the phone, and that was enough.

# Seventeen hours

In early 2026, when ExplainMate was still very new, I was building a quiz feature.

One of its modes was assertion and reason. Anyone who has done CBSE science knows these: you're given a statement, the Assertion, and a Reason, and you have to decide whether each is true and whether the reason actually explains the assertion. They're among the trickiest questions in board papers, so I badly wanted ExplainMate to get them right.

At first it was terrible. It repeated questions, sometimes left the reason out altogether, sometimes wrote an assertion so long it stopped being a question, and every so often the AI simply made things up. On some weekends I spent seventeen hours on my laptop working on that one mode of that one feature. It's still in ExplainMate, and these days it works.

. . .

The most ridiculous thing that happened while I was building was the bug I spent ages fixing that wasn't there. I'd change the code, check the preview on localhost, see the bug still showing, and change the code again. It turned out the code had updated but the

preview hadn't. The bug had been fixed for a while, and I'd been chasing a ghost. If you build things, it will happen to you too. Laugh, and restart the server.

## Building with AI

Around February 2026 I switched to building entirely with AI, the thing people call vibe coding, and it changed how fast I could ship. I kept two rules, though.

The first is that I review everything. The AI writes the code, but I read it and understand it before it goes live. If I can't explain what part of my product does, I don't really own it, and I'm the one who has to fix it when it breaks at 2 a.m.

The second is that nothing confidential goes into the chat: no keys, no passwords, no user data.

---

### WHAT I TOOK FROM IT

Some features take seventeen hours. Some bugs aren't even there. Both are part of building.

# The login button

Around February 2026, one of my reels finally took off. It was about a hackathon, and it went past 500,000 views.

People were going to my profile, tapping the ExplainMate link and landing on the app. I could see them arriving, but I couldn't see anyone logging in, which made no sense. So I opened the app and pressed the login button myself.

Nothing happened. I pressed it again, and still nothing. I remember thinking, *oh hell no*. It had been broken the whole time. The one time Instagram was sending me real people, my own app was turning them away.

I'll never know exactly how many I lost, because people who hit a broken app don't write in to complain. They leave and don't come back. My best guess is somewhere between two and four thousand.

• • •

Everyone in startups says you shouldn't wait for perfect, and I agree. If you wait for perfect, you never launch. But that day showed me where the line is. Something unfinished is fine: a

missing feature, a screen that isn't pretty yet. Something broken isn't. A login that doesn't work isn't a rough edge. It's a locked door with your name on it.

## | Trust lost is lost for life.

Nobody arriving from a reel thinks, *early product, I'll check back next month*. They think it doesn't work, and that's the last thought they ever have about it. These days, before anything goes out, I go through the app the way a brand-new user would, every single time.

---

### WHAT I TOOK FROM IT

Don't ship perfect. But never ship broken.

END OF THE FREE SAMPLE

# That was before anyone was watching.

The rest is the views, and everything that came after them.

- A hundred reels that went nowhere, and the week I almost deleted everything
- Waking up to 50,000 views, and ₹149 at the barber shop
- 18,000 signups in three weeks, and a ₹21,800 server bill
- Being fifteen in a room of twenty-five-year-olds, and ending up in Dainik Jagran
- Explaining it all at home, what I gave up, and a letter to fourteen-year-old me

**Get the full book at [avenoric.tech/store](https://avenoric.tech/store)**