

Continue your app journey on mobile- Rocket Mobile is live on iOS. [Download now.](#)

AI Tutor India

Landing page

Preview

Code

Integrations

Upgrade

AI TUTOR – LANDING PAGE LAYOUT (Tier 2/3/4 India) SECTION 1 — HERO Headline: “AI Tutor – Har Bacche Ka Personal Teacher. 24×7.” Subtext: Class 1–12. All boards. Homework, doubts, tests, revision. CTA: Start Free Learning Trust Bar: 5 Lakh+ Questions Solved | Ex-Microsoft Team | Works on Low-End Phones | Hindi + English SECTION 2 — PROBLEM • No good tutors nearby • High tuition fees • Parents can’t help with homework • English textbooks confusing SECTION 3 — SOLUTION • Doubt Solving (Photo ® Steps) • Homework Help • Free Notes (NCERT) • Practice Tests • Voice Chat (Coming Soon) SECTION 4 — TRUST BOOSTER Parent Testimonials (3 examples) SECTION 5 — PRICING Free Plan n99/month plan n199/month family plan SECTION 6 — FEATURE DEMO Short video/screenshots suggested SECTION 7 — LANGUAGE SUPPORT Hindi, English, Hinglish SECTION 8 — PARENTS’ SAFETY Safe, private, no ads SECTION 9 — SCHOOL PARTNERSHIPS Used in 40+ towns SECTION 10 — FINAL CTA Start Free Learning

Rocket

Perfect! I'll build your AI Tutor landing page with all those detailed sections. Let me help you choose the best technology stack for this project.

Rocket

I'll begin with our default framework. You can customize the framework if you want.

My default framework-

- Next.js
- TypeScript
- Tailwind CSS

Use default framework

Rocket

Perfect! I'll build your AI Tutor landing page using Next.js and TypeScript. The app will initially be frontend-only with all the detailed sections you specified - from the hero section with Hindi/English support to parent testimonials and pricing plans.

Rocket

Select the screens you wish to build from the list below. Token consumption is directly related to the number of screens generated.



Landing Page

1 screen(s) selected

Rocket

I'm generating screens for your app.

Landing Page

Completed

Rocket

- Created mobile-first AI Tutor landing page with 8 conversion sections and bilingual Hindi/English support
- Implemented interactive components including rotating testimonials, FAQ accordion, and multi-step OTP signup form
- Added responsive design with progressive loading, trust signals, and strategic conversion CTAs with countdown timers
- Login credentials for app's functional experience. Phone OTP: 123456

Files (12)

Code diff

Rollback

Here is a clean, 1-page UI/UX design blueprint for the Logged-In User Home Screen of your AI Tutor app — crafted specifically for Tier-2/3/4 Indian students, low-end devices, and clarity-first UX. This is a final, ready-to-build screen layout. AI Tutor – Logged-In User Home (1-Page Design Layout)
Simple, fast, distraction-free, Bharat-friendly. 1 Top Bar (Most Important Actions Only) Left: AI Tutor logo (small) Center: "Hi, Anay! " (Student name auto-fills) Right: Profile icon → Settings, Language, Parent Mode 2 Quick Input Box (Primary Function) Big Search/Doubt Input Card: Upload Photo | Speak | Type your question... One-tap photo upload One-tap voice question Lowest friction possible CTA below: [Ask AI Tutor] (Full-width button) 3 Continue Learning (Personalized) Heading: Continue Learning Shows last 3 active items: "Fractions – NCERT Class 6 · Continue" "Photosynthesis Doubt · View Answer" "Last Test: Algebra Basics · Score: 72% · Improve Now" Tap → resumes instantly. 4 Suggested For You (AI-Recommended) Horizontal Cards

(scrollable): NCERT Chapter Notes eg: "Class 6 → Chapter 4 → Simple Equations" Daily 10-Minute Practice 8–10 auto-generated questions Weak Areas Identified Math: Fractions → 62% Science: Circuit Diagram → 54% Today's Challenge 5 mixed questions for streaks Main Feature Blocks (Grid – 2×2) Doubt Solver Upload → Step-by-step solution. Notes & Concepts NCERT/CBSE/state board. Practice Tests Auto-generate tests, instant score. Voice Tutor Ask a doubt using voice. (If coming soon → label badge: Beta) Study Goals / Streak Zone Simple gamification (no clutter). Today's Study Goal: 10 mins Completed: 4 mins Streak: 3 days One-line motivational text: "Great! Just 6 minutes left for full progress." Parent Mode (Small, Non-intrusive) Card at bottom: Parent Dashboard Track progress Weekly report Safe learning environment Tap → switch to Parent View. Bottom Navigation (4 Tabs, Max) Home | Tests | Notes | Profile (Simple icons, no text overload) Design Principles Used Zero complexity: Tier-2/3/4 students prefer 3–4 obvious actions Photo-first doubt solving → most used feature Notes + Tests = second highest usage Gamification minimal, not flashy English simplified, Hinglish friendly Big touch targets for small screens Designed for 5-inch low RAM devices

Rocket

I'll create the logged-in user home screen for your AI Tutor app with all the specific design elements you've outlined for Tier-2/3/4 Indian students.

Rocket

Screen set in progress

- Student Home Dashboard

Rocket

- Created student home dashboard page with 8 sections optimized for Tier-2/3/4 Indian students and low-end devices
- Implemented bilingual UI components including quick input box, learning progress tracker, and gamified study goals
- Added bottom navigation with Home, Tests, Notes, and Profile tabs for easy mobile navigation
- Integrated third-party APIs for student data analytics and personalized learning recommendations

Files (10)

page.tsx

StudentHomeDashboard.tsx

TopBar.tsx

QuickInputBox.tsx

ContinueLearning.tsx

SuggestedContent.tsx

FeatureGrid.tsx

StudyGoals.tsx

ParentModeCard.tsx

BottomNavigation.tsx

Code diff

Rollback

make the student dashboard functional and also with the database structure

Rocket

I'll make your student dashboard fully functional with Supabase backend integration and create the complete database structure for all the AI Tutor features.

Rocket

I'll connect with your existing database.

Connected to ai-tutor

Rocket

Integrating Supabase project with your app. This will take few minutes.

every question and its response would need the responses to be shown. design the question and answer UI in the student section

Rocket

I'll design a comprehensive question and answer UI interface for your student section where students can view their questions and AI tutor responses with a clean, mobile-friendly layout optimized for Indian students.

every question and its response would need the responses to be shown. Suggest me the design the question and answer UI in the student section but do not generate the design yet

Rocket

Here's my design suggestion for the Question & Answer UI in your student section, optimized for Tier 2/3/4 Indian students:

(opens in new tab) Q&A Interface Design

Recommendations

(opens in new tab) 1. Q&A List View (Main Screen)

Search your doubts... 

Recent Questions

-  Math - Fractions
"How to solve $\frac{2}{3} + \frac{1}{4}$?"
✓ Answered • 2 hours ago
Helpful

-  Science - Photosynthesis
"Why do plants need sunlight?"
X Processing... • 5 mins ago

(opens in new tab) 2. Individual Q&A Detail View

[← Back to Questions](#) [: Menu](#)

-  Your Question • Math • 2 hours ago
[Photo of handwritten math problem]
"How to solve $\frac{2}{3} + \frac{1}{4}$?"

AI Tutor's Answer

- Step 1: Find common denominator
 $\frac{2}{3} + \frac{1}{4}$
LCM of 3 and 4 = 12
- Step 2: Convert fractions
 $\frac{2}{3} = \frac{8}{12}$
 $\frac{1}{4} = \frac{3}{12}$
- Step 3: Add
 $\frac{8}{12} + \frac{3}{12} = \frac{11}{12}$
- Final Answer: $\frac{11}{12}$

Was this helpful?

Yes (124) No (2) Ask Follow-up

(opens in new tab) 3. Key Design Elements to Include:

Question Cards Structure:

- Question type indicator ( Photo,  Text,  Voice)
- Subject badge (Math, Science, etc.)

- Question preview (first 50 characters)
- Status indicator (☑ Answered, ✅ Processing, ✗ Failed)
- Timestamp in simple format ("2 hours ago")
- Helpful rating count

Answer Display Features:

- Step-by-step breakdown with numbers
- Visual highlighting for final answers
- Code/equation formatting where needed
- Image support for diagrams
- Audio playback for voice answers

Mobile-Optimized Elements:

- Large touch targets (48px minimum)
- Swipe gestures for navigation
- Pull-to-refresh functionality
- Infinite scroll for question history
- Offline viewing for answered questions

Smart Organization:

- Filter by subject (Math, Science, English, etc.)
- Filter by status (Answered, Processing, Bookmarked)
- Search within questions and answers
- Sort by recent, most helpful, unanswered

Engagement Features:

- Bookmark favorite answers
- Rate answer helpfulness
- Ask follow-up questions directly
- Share solutions with study groups
- Progress tracking (questions asked per subject)

Bilingual Support:

- Toggle between Hindi/English interface
- Mixed language question support
- Audio answers in preferred language
- Regional language math terminology

Performance for Low-End Devices:

- Lazy loading of images
- Compressed answer content
- Minimal animations
- Efficient caching strategy
- Progressive web app features

Would you like me to implement this Q&A interface design in your student dashboard?

since this has phone based signup, how will we capture student name etc. also generate the tab specific view ideas

Rocket

Nov 30, 2025, 10:38 AM message timestamp

Great question! Since you're using phone-based OTP signup, you'll need a smart onboarding flow to capture student details. Here are my recommendations:

[**\(opens in new tab\)**](#) **Student Onboarding Flow (Post-OTP)**

[**\(opens in new tab\)**](#) **Step 1: Welcome & Name Collection**

 **Welcome to AI Tutor!**

What should we call you?

Enter your name
[<input type="text"/>
]

This helps us personalize your learning

[[Continue](#)]

[**\(opens in new tab\)**](#) **Step 2: Class & Board Selection**

 **Tell us about your studies**

Which class are you in?

[1] [2] [3] [4] [5] [6] [7] [8] [9] [10] [11] [12]

Which board?

- CBSE
- ICSE
- State Board
- Other

[Next Step]

(opens in new tab)Step 3: Subject Preferences

Q What subjects do you need help with?

Select all that apply:

- Mathematics
- Science
- English
- Social Studies
- Hindi
- Computer Science

[Start Learning!]

(opens in new tab) 📚 Tab-Specific View Ideas

(opens in new tab) 🏠 HOME TAB(Already designed)

- Quick input for doubts
- Continue learning section
- Suggested content
- Study streaks & goals

(opens in new tab) 💬 TESTS TAB

📊 Practice Tests

Q Quick Practice (5-10 mins)

-  Math: Basic Algebra
- 10 Questions • 8 mins

[Start Now]

📋 Chapter Tests

-  Science: Acids & Bases

Class 7 • 15 Questions

Best Score: 80%

[Retake Test]

Test History

Math Quiz • Yesterday • 72%

Science Test • 2 days ago • 85%

English Grammar • 3 days ago • 90%

 Weekly Challenge

Advanced problems for extra practice

(opens in new tab)  NOTES TAB

 Study Notes

Search notes...  Download

 By Subject

 Mathematics (24 chapters)

 Science (18 chapters)

 English (12 chapters)

 Social Studies (15 chapters)

 Bookmarked Notes

 Triangles & Properties

 Photosynthesis Process

 Grammar Rules

 Downloaded (Offline)

 Algebra Basics

 Chemical Reactions

NEW Recently Added

Latest notes from your syllabus

(opens in new tab)  PROFILE TAB

 Hi, Anay!

 Your Progress

Questions Asked: 127

Tests Completed: 23

Study Streak:  5 days

Favorite Subject: Mathematics

Settings

Language: English
Class: 8th Grade
Board: CBSE
Notifications: ON

Family

- Parent Dashboard Access
- Weekly Report to Parents
- Privacy Settings

Support

- Chat Support
- FAQ
- Call Support
- Rate App

(opens in new tab) Database Structure for Student Data

sql

```
-- Students table (extends Supabase auth.users)
students (
    id (uuid, foreign key to auth.users.id)
    phone_number (text)
```

Would you like me to implement this complete student onboarding flow and tab structure with the database integration?

what is the idea with the parent dashboard and the db structure.

Rocket

(opens in new tab) Parent Dashboard - Complete Concept & DB Structure

(opens in new tab) Parent Dashboard Purpose

The parent dashboard addresses key concerns of Tier 2/3/4 Indian parents:

- **Progress Monitoring:** Track child's learning without being intrusive
- **Safety Assurance:** Ensure child is in a safe learning environment
- **Academic Support:** Understand where child needs help

- **Value Validation:** See ROI of the tutoring investment

[**\(opens in new tab\)**](#) Parent Dashboard UI Concept

[**\(opens in new tab\)**](#) Parent Home Screen

 Parent Dashboard

 Anay's Progress This Week

Questions Asked:	23
Study Time:	2hr 45min
Tests Completed:	4
Avg Score:	78%  (+12%)

 Subject Performance

Math:	85% (Strong)
Science:	70% (Improving)
English:	92% (Excellent)
Social:	65% (Needs Focus)

 Areas Needing Attention

- Fractions & Decimals
- Chemical Equations
- Geography Maps

 Recent Activity

Asked doubt in Math -	2 hrs ago
Completed Science test -	Yesterday
Studied for 45 mins -	Yesterday

[**\(opens in new tab\)**](#) Detailed Reports Screen

 Weekly Report

 Study Pattern

Mon	45min
Tue	30min
Wed	60min
Thu	20min
Fri	65min
Sat	80min
Sun	40min

Learning Goals Progress

Daily 30min target:	✓ 6/7 days
Weekly test goal:	✓ 4/3 completed
Doubt clearing:	✓ 23 questions

[Email Report] [Share with Spouse]

(opens in new tab) Safety & Privacy Screen

Safety Dashboard

Safe Learning Environment

✓ No inappropriate content
✓ No external links or ads
✓ AI responses monitored
✓ No peer chat/social features

Time Controls

Daily Limit:	2 hours
Study Time:	6 PM - 8 PM
Break Reminders:	Every 30 min
[Edit Settings]	

Content Monitoring

Questions reviewed:	127/127
Inappropriate flags:	0
Last content check:	2 hours ago

(opens in new tab) Complete Database Structure

(opens in new tab) Parent-Student Relationship Tables

sql

```
-- Parent Users (extends Supabase auth.users)
parents (
    id uuid PRIMARY KEY REFERENCES auth.users(id),
    phone_number text UNIQUE NOT NULL,
```

(opens in new tab) Enhanced Student Tables (Updated)

sql

```

-- Updated Students table
ALTER TABLE students ADD COLUMN IF NOT EXISTS
parent_oversight_enabled boolean DEFAULT true,
last_activity_at timestamp DEFAULT now(),
total_study_minutes integer DEFAULT 0,
safety_flags integer DEFAULT 0;

-- Updated Student Questions table
ALTER TABLE student_questions ADD COLUMN IF NOT EXISTS
is_flagged boolean DEFAULT false,
flag_reason text,
parent_reviewed boolean DEFAULT false,
content_safety_score integer DEFAULT 100; -- 0-100, lower = more concerning

-- Question Content Monitoring
question_content_flags (
    id uuid PRIMARY KEY DEFAULT gen_random_uuid(),
    question_id uuid REFERENCES student_questions(id) ON DELETE CASCADE,
    flag_type text NOT NULL, -- 'inappropriate_language', 'non_academic',
    'external_help_seeking'
    flag_severity text DEFAULT 'low', -- 'high', 'medium', 'low'
    auto_flagged boolean DEFAULT false,
    reviewer_notes text,
    resolution_status text DEFAULT 'pending', -- 'resolved', 'dismissed', 'escalated'
    flagged_at timestamp DEFAULT now(),
    resolved_at timestamp
);

```

(opens in new tab)📊 Parent Analytics Views

sql

```

-- Weekly Performance Summary View
CREATE VIEW parent_weekly_summary AS
SELECT
    s.id as student_id,

```

(opens in new tab)⌚ Key Parent Dashboard Features

- Real-time Progress Tracking:** Live updates when child studies
- Safety Monitoring:** Content filtering and time controls
- Weekly Email Reports:** Automated summaries sent to parent email
- Achievement Notifications:** Celebrate milestones and improvements
- Concern Alerts:** Notify parents of academic struggles or safety issues
- Tutor Communication:** Direct channel for parent-AI tutor discussions about child's progress

(opens in new tab)🔒 Privacy & Access Model

sql

```
-- Row Level Security Policies
CREATE POLICY "Parents can only see their children's data" ON weekly_reports
FOR SELECT USING (
    student_id IN (
        SELECT student_id FROM student_parent_relationships
        WHERE parent_id = auth.uid()
    )
);
```

This creates a comprehensive parent ecosystem that builds trust while maintaining student privacy and autonomy.

Would you like me to implement any specific part of this parent dashboard system?

how do you plan to manage continue learning, "suggested for you" & streakssection and how does it look like on the DB?

Rocket

[**\(opens in new tab\)**](#) Continue Learning, Suggested Content & Streaks - Implementation Strategy

Here's how I'd architect these personalized learning features with intelligent data tracking:

[**\(opens in new tab\)**](#) 1. CONTINUE LEARNING Section

[**\(opens in new tab\)**](#) Logic & Behavior

Shows last 3-5 incomplete or resumable activities:

- Unfinished tests (auto-saved progress)
- Partially read chapter notes
- Unanswered follow-up questions
- Incomplete practice sessions
- Bookmarked difficult problems

[**\(opens in new tab\)**](#) Database Structure

sql

```
-- Learning Sessions (tracks all learning activities)
learning_sessions (
    id uuid PRIMARY KEY DEFAULT gen_random_uuid(),
    student_id uuid REFERENCES students(id),
```

(opens in new tab) 📱 Continue Learning UI Logic

typescript

```
// Query to get continuable activities
const getContinueLearning = async (studentId: string) => {
  return await supabase
    .from('learning_sessions')
```

(opens in new tab) ⚡ 2. SUGGESTED FOR YOU Section

(opens in new tab) 📊 Recommendation Engine Logic

AI-driven suggestions based on:

- Weak subject areas (low test scores)
- Frequently asked question topics
- Upcoming curriculum (syllabus-based)
- Peer learning patterns (anonymous)
- Time-of-day learning preferences
- Seasonal academic calendar

(opens in new tab) 🗂 Database Structure

sql

```
-- Student Learning Profile
student_learning_profile (
  id uuid PRIMARY KEY DEFAULT gen_random_uuid(),
  student_id uuid REFERENCES students(id) UNIQUE,
```

(opens in new tab) ⚡ Recommendation Generation Function

sql

```
-- Stored procedure to generate personalized recommendations
CREATE OR REPLACE FUNCTION generate_student_recommendations(p_student_id uuid)
RETURNS void AS $$
DECLARE
```

(opens in new tab) 🎮 3. STREAKS Section

(opens in new tab) 🎮 Gamification Strategy

```
Multiple streak types:  
- Daily study streak (studied each day)  
- Question asking streak (asked question daily)  
- Test completion streak (completed 1 test daily)  
- Subject mastery streak (consistent performance)  
- Learning goal streak (met daily targets)
```

(opens in new tab) 📁 Database Structure

sql

```
-- Student Streaks  
student_streaks ()  
    id uuid PRIMARY KEY DEFAULT gen_random_uuid(),  
    student_id uuid REFERENCES students(id),
```

(opens in new tab) ⚙️ Streak Management Functions

sql

```
-- Function to update streaks after any student activity  
CREATE OR REPLACE FUNCTION update_student_streaks(p_student_id uuid, p_activity_type text)  
RETURNS void AS $$  
DECLARE
```

(opens in new tab) 📱 Frontend Implementation

Strategy

(opens in new tab) React Hook for Continue Learning

typescript

```
const useContinueLearning = (studentId: string) => {  
  const [activities, setActivities] = useState([]);  
  
  useEffect(() => {
```

(opens in new tab) Real-time Streak Updates

typescript

```
const useStreaks = (studentId: string) => {  
  const [streaks, setStreaks] = useState({});
```

```
useEffect(() => {
```

This architecture creates a comprehensive personalized learning system that adapts to each student's behavior and academic needs!