

SKIDS Advanced - Enterprise Integration Infrastructure

Comprehensive Child Development Platform with Advanced Vendor Management

Next.js	14	(https://nextjs.org/)
TypeScript	5.0	(https://www.typescriptlang.org/)
Tailwind	3.0	(https://tailwindcss.com/)
License	Proprietary	(LICENSE)

A comprehensive enterprise-grade platform for child development assessment, care plan management, vendor integration, and advanced analytics with real-time monitoring capabilities.

TABLE OF CONTENTS

- 1. [Overview](#)
- 2. [Key Features](#)
- 3. [Architecture](#)
- 4. [Quick Start](#)
- 5. [Project Structure](#)
- 6. [Configuration](#)
- 7. [Testing](#)
- 8. [Deployment](#)
- 9. [Documentation](#)
- 10. [Contributing](#)
- 11. [Support](#)

OVERVIEW

SKIDS Advanced is an enterprise-grade child development platform that combines comprehensive health assessment tools with advanced vendor management, real-time analytics, and AI-powered insights. The platform serves healthcare providers, parents, and administrative staff with role-based access and specialized workflows.

Platform Highlights

- **Healthcare Focus:** Pediatric health assessments and care plan management
 - **AI-Powered:** ROI analysis, predictive analytics, and intelligent recommendations
 - **Enterprise-Ready:** Vendor onboarding, staff management, and compliance tracking
 - **Real-time Analytics:** Unified dashboard with live metrics and performance monitoring
 - **HIPAA Compliant:** Security-first architecture with data encryption and audit logging
 - **High Performance:** Optimized for speed with 99.9% uptime targets
-

KEY FEATURES

Core Platform Features

Discovery & Assessment System

- **Interactive Body Systems:** Brain, cardiovascular, respiratory, digestive, sensory
- **Developmental Domains:** Cognitive, language, social-emotional, motor, sensory
- **Age-Adaptive Content:** Tailored experiences for different age groups
- **Interactive Elements:** 60fps animations and engaging visualizations

Intervention Programs

- **Vision Care:** Amblyopia/myopia treatment with arrest lenses
- **Hearing Solutions:** SoundScout technology integration
- **Nutrition AI:** Personalized meal planning with NutreeAI integration
- **Wellness Programs:** Pranayama coaching and mindfulness training
- **Dental Care:** Comprehensive oral health programs
- **Specialized Care:** Dermatology, allergy, and sleep interventions

Care Plan Management

- **Dynamic Care Plans:** AI-powered plan generation and customization
- **Subscription Tiers:** Essential (₹299), Comprehensive (₹499), Premium (₹799)
- **Real-time Updates:** Live plan modifications and progress tracking
- **Outcome Tracking:** Comprehensive progress monitoring and reporting

Enterprise Integration Features

Vendor Management System

- **8-Step Onboarding:** Comprehensive vendor integration workflow
- **Performance Tracking:** Real-time vendor performance monitoring
- **Relationship Management:** Automated vendor communication and support
- **ROI Analysis:** AI-powered vendor performance and cost-benefit analysis
- **Compliance Monitoring:** HIPAA and regulatory compliance tracking

Staff Management Tools

- **Role-Based Access:** Admin, staff, provider, vendor, specialist roles
- **Performance Analytics:** Individual and team productivity tracking
- **KRA Management:** Key Result Areas with automated performance reviews
- **Training Systems:** Comprehensive onboarding and skill development

Unified Analytics Dashboard

- **Real-time Metrics:** Live system performance and business metrics
- **Predictive Analytics:** AI-powered forecasting and trend analysis
- **ROI Tracking:** Comprehensive return on investment analysis
- **Alert Management:** Automated monitoring and escalation procedures

Payment & Subscription Management

- **Multi-Gateway Support:** Razorpay, Stripe, PayPal, Square integration
- **Subscription Management:** Automated billing and renewal handling
- **Revenue Tracking:** Comprehensive financial analytics and reporting
- **Secure Processing:** PCI-compliant payment handling

ARCHITECTURE

Frontend Architecture

Next.js 14 App Router

- └─ React 18 with TypeScript
- └─ Tailwind CSS for styling
- └─ Framer Motion for animations
- └─ Lucide React for icons
- └─ Custom component library

⚙ Backend Architecture

API Routes (Next.js)

- └─ RESTful API design
- └─ TypeScript for type safety
- └─ Middleware for authentication
- └─ Rate limiting and security
- └─ Error handling and logging

Data Architecture

Supabase/PostgreSQL

- └─ User management tables
- └─ Care plans and subscriptions
- └─ Vendor and staff data
- └─ Analytics and metrics
- └─ Audit logs and compliance

Security Architecture

Multi-layer Security

- └─ Role-based access control (RBAC)
- └─ JWT token authentication
- └─ Data encryption at rest and in transit
- └─ HIPAA compliance measures
- └─ Audit logging and monitoring

QUICK START

Prerequisites

- **Node.js:** 18.0.0 or higher
- **npm:** 8.0.0 or higher (or yarn/pnpm)

- **Git:** Latest version
- **Database:** PostgreSQL 15+ (or Supabase account)

✂ Installation

1. Clone the Repository

```
git clone https://github.com/your-org/skids-advanced.git
cd skids-advanced
```

2. Install Dependencies

```
npm install
# or
yarn install
# or
pnpm install
```

3. Environment Setup

```
cp .env.example .env.local
```

4. Configure Environment Variables

```
# Edit .env.local with your configuration
# See Configuration section for details
```

5. Start Development Server

```
npm run dev
```

6. Access the Application

- **Main Platform:** <http://localhost:3001>
- **Analytics Dashboard:** <http://localhost:3001/admin/analytics>
- **Vendor Management:** <http://localhost:3001/admin/vendor-management>
- **Staff Management:** <http://localhost:3001/admin/staff-management>

Quick Testing

```
# Run all tests
npm run test:all

# Run development tests
npm run test:dev

# Run with coverage
npm run test:coverage
```

PROJECT STRUCTURE

```
skids-advanced/
├── src/
```

app/	# Next.js 14 App Router
layout.tsx	# Root layout
page.tsx	# Homepage
admin/	# Admin interfaces
analytics/	# Unified analytics dashboard
vendor-management/	# Vendor management system
staff-management/	# Staff management tools
care-plans/	# Care plan management
discovery/	# Discovery journeys
interventions/	# Intervention programs
dashboard/	# User dashboards
api/	# API routes
components/	# Reusable components
layout/	# Layout components
discovery/	# Discovery components
interventions/	# Intervention components
care-plans/	# Care plan components
analytics/	# Analytics components
vendor/	# Vendor components
chat/	# Chat components
ui/	# Base UI components
lib/	# Utility libraries
api/	# API layer
payment/	# Payment gateways
ai/	# AI/ML utilities
utils/	# Helper functions
types/	# TypeScript definitions
analytics.ts	# Analytics types
vendor.ts	# Vendor types
payment.ts	# Payment types
auth.ts	# Authentication types
service-integration.ts	# Integration types
hooks/	# Custom React hooks
styles/	# Global styles
docs/	# Documentation
INTERNAL_TESTING_DEPLOYMENT_PLAN.md	
PRODUCTION_DEPLOYMENT_GUIDE.md	
KRA_SLA_FRAMEWORK.md	
tests/	# Test files
unit/	# Unit tests
integration/	# Integration tests
e2e/	# End-to-end tests
performance/	# Performance tests
README_TESTING.md	# Testing framework guide
.env.example	# Environment template
package.json	# Dependencies
tsconfig.json	# TypeScript config
tailwind.config.js	# Tailwind config
next.config.js	# Next.js config
jest.config.js	# Jest config

CONFIGURATION

Environment Variables

Create .env.local file:

```
# =====  
# SKIDS ADVANCED - COMPREHENSIVE ENVIRONMENT CONFIGURATION  
# =====  
  
# Application Settings  
NODE_ENV=development  
NEXT_PUBLIC_APP_ENV=development  
NEXT_PUBLIC_API_BASE_URL=http://localhost:3001  
  
# Authentication (Clerk)  
NEXT_PUBLIC_CLERK_PUBLISHABLE_KEY=your_clerk_publishable_key_here  
CLERK_SECRET_KEY=your_clerk_secret_key_here  
  
# Payment Gateways  
RAZORPAY_KEY_ID=rzp_test_your_key_here  
RAZORPAY_KEY_SECRET=your_razorpay_secret_here  
STRIPE_PUBLISHABLE_KEY=pk_test_your_stripe_key_here  
STRIPE_SECRET_KEY=sk_test_your_stripe_secret_here  
  
# External Services  
NUTREEAI_API_KEY=your_nutreeai_api_key_here  
SHANTI_API_KEY=your_shanti_api_key_here  
  
# Feature Flags  
ENABLE_VENDOR_ONBOARDING=true  
ENABLE_AI_RECOMMENDATIONS=true  
ENABLE_ROI_ANALYTICS=true  
USE MOCK PAYMENTS=true  
USE MOCK AI=true
```

Available Scripts

```
# Development
npm run dev      # Start development server
npm run build    # Build for production
npm run start    # Start production server

# Testing
npm run test:all  # Run all tests
npm run test:unit # Run unit tests
npm run test:integration # Run integration tests
npm run test:e2e  # Run end-to-end tests
npm run test:performance # Run performance tests
npm run test:coverage # Run tests with coverage

# Code Quality
npm run lint      # Run ESLint
npm run lint:fix   # Fix ESLint issues
npm run type-check # Run TypeScript checks
npm run format     # Format with Prettier

# Deployment
npm run deploy:staging # Deploy to staging
npm run deploy:production # Deploy to production
```

TESTING

Testing Strategy

- **Unit Tests:** Component and function testing with Jest
- **Integration Tests:** API and service integration testing
- **E2E Tests:** Full user workflow testing with Playwright
- **Performance Tests:** Load and stress testing with k6

Running Tests

```
# Run specific test suites
npm run test:unit
npm run test:integration
npm run test:e2e
npm run test:performance

# Run with coverage
npm run test:coverage

# Run in watch mode
npm run test:watch
```

Test Coverage Requirements

- **Unit Tests:** 90% coverage minimum
- **Integration Tests:** All API endpoints covered

- **E2E Tests:** Critical user journeys covered
 - **Performance Tests:** All pages under 2s load time
-

DEPLOYMENT

Deployment Environments

1. **Development:** http://localhost:3001
2. **Staging:** https://staging.skids.clinic
3. **Production:** https://app.skids.clinic

Deployment Process

```
# 1. Run tests
npm run test:all

# 2. Build application
npm run build

# 3. Deploy to staging
npm run deploy:staging

# 4. Run staging tests
npm run test:staging

# 5. Deploy to production
npm run deploy:production
```

Infrastructure Requirements

- **Server:** 8 vCPUs, 16GB RAM, 200GB SSD
 - **Database:** PostgreSQL 15+ with read replicas
 - **CDN:** Cloudflare with global edge locations
 - **Monitoring:** Comprehensive APM and alerting
-

DOCUMENTATION

Technical Documentation

- [Testing Framework Guide \(README_TESTING.md\)](#)
- [Internal Testing Plan \(docs/INTERNAL_TESTING_DEPLOYMENT_PLAN.md\)](#)
- [Production Deployment Guide \(docs/PRODUCTION_DEPLOYMENT_GUIDE.md\)](#)
- [KRA/SLA Framework \(docs/KRA_SLA_FRAMEWORK.md\)](#)

User Documentation

- [Non-IT Team User Manual \(docs/USER_MANUAL_NON_IT.md\)](#)
 - [Admin User Guide \(docs/ADMIN_USER_GUIDE.md\)](#)
 - [Vendor Onboarding Guide \(docs/VENDOR_ONBOARDING_GUIDE.md\)](#)
-

CONTRIBUTING

Development Workflow

1. Create Feature Branch

```
git checkout -b feature/your-feature-name
```

2. Make Changes

- Follow TypeScript best practices
- Write tests for new features
- Update documentation

3. Test Changes

```
npm run test:all  
npm run lint  
npm run type-check
```

4. Submit Pull Request

- Provide clear description
- Include test coverage
- Update documentation

Code Standards

- **TypeScript:** Strict mode enabled
 - **ESLint:** Airbnb configuration
 - **Prettier:** Consistent formatting
 - **Conventional Commits:** Structured commit messages
-

SUPPORT

Team Contacts

- **Project Lead:** project-lead@skids.clinic
- **Technical Lead:** tech-lead@skids.clinic
- **DevOps Lead:** devops@skids.clinic
- **QA Lead:** qa-lead@skids.clinic

Issue Reporting

- **Bug Reports:** Use GitHub Issues with bug template
- **Feature Requests:** Use GitHub Issues with feature template
- **Security Issues:** Email security@skids.clinic

Documentation

- **API Documentation:** Available at /api/docs
- **Component Library:** Available at /storybook

- **Architecture Diagrams:** Available in </docs/architecture>
-

LICENSE

This project is proprietary software developed for SKIDS Advanced platform.

© 2024 SKIDS.CLINIC. All rights reserved.

**ENTERPRISE-READY PLATFORM FOR COMPREHENSIVE CHILD DEVELOPMENT &
HEALTHCARE MANAGEMENT**