

# Education Analytics Platform

---

A modern, React-based educational analytics dashboard that helps track student performance, course progress, and academic achievements.



## Features

- **Real-time Dashboard**

- Student performance metrics
- Course progress tracking
- Assessment history
- Grade analytics

- **Course Management**

- Active course overview
- Progress tracking
- Module completion status
- Performance trends

- **Performance Analytics**

- Grade visualization
- Progress charts

- Comparative analytics
- Achievement tracking

## Tech Stack

- **Frontend**

- React 18
- TypeScript
- Tailwind CSS
- Lucide React (icons)
- Vite (build tool)

- **Backend**

- Supabase
- PostgreSQL
- Row Level Security (RLS)

## Prerequisites

Before you begin, ensure you have the following installed:

- Node.js (v18 or higher)
- npm (v9 or higher)
- Git

## Getting Started

### 1. Clone the repository

```
git clone <repository-url>
cd education-analytics-platform
```

### 2. Install dependencies

```
npm install
```

### 3. Set up environment variables

Create a `.env` file in the root directory:

```
VITE_SUPABASE_URL=your_supabase_url
VITE_SUPABASE_ANON_KEY=your_supabase_anon_key
```

### 4. Start the development server

```
npm run dev
```

## 📁 Project Structure

```
education-analytics-platform/
├── src/
│   ├── components/
│   │   ├── Dashboard/
│   │   │   ├── AssessmentList.tsx
│   │   │   ├── CourseList.tsx
│   │   │   ├── CourseProgress.tsx
│   │   │   ├── PerformanceChart.tsx
│   │   │   └── StatCard.tsx
│   │   └── Layout/
│   │       └── Header.tsx
│   ├── hooks/
│   │   ├── useCourseProgress.ts
│   │   ├── usePerformanceData.ts
│   │   └── useStudentData.ts
│   ├── lib/
│   │   └── supabase.ts
│   ├── data/
│   │   └── mockData.ts
│   ├── App.tsx
│   └── main.tsx
└── supabase/
    └── migrations/
        └── 0001_heavy_tree.sql
└── public/
```

## 🗄️ Database Schema

### Tables

#### 1. **students**

- Personal information
- Enrollment details
- Academic status

#### 2. **courses**

- Course information
- Difficulty levels
- Credit hours

#### 3. **enrollments**

- Student-course relationships

- Progress tracking
- Completion status

#### 4. **performance\_records**

- Assessment results
- Score tracking
- Performance history

## Security

The application implements comprehensive security measures:

- Row Level Security (RLS) policies
- User authentication
- Data access control
- Secure API endpoints

## Development

### Available Scripts

- `npm run dev` - Start development server
- `npm run build` - Build for production
- `npm run preview` - Preview production build
- `npm run lint` - Run ESLint

### Code Style

The project uses:

- ESLint for code linting
- TypeScript for type safety
- Prettier for code formatting

## Components

### Dashboard Components

#### 1. **StatCard**

- Display key metrics
- Visual indicators
- Trend analysis

#### 2. **CourseProgress**

- Module tracking
- Completion percentage
- Visual progress bar

### 3. PerformanceChart

- Grade visualization
- Progress tracking
- Trend analysis

### 4. AssessmentList

- Recent assessments
- Score history
- Course-wise breakdown

## Layout Components

### 1. Header

- Navigation
- User profile
- Notifications

## ⌚ Custom Hooks

### 1. useStudentData

- Fetch student statistics
- Calculate performance metrics
- Track progress

### 2. useCourseProgress

- Monitor course completion
- Track module progress
- Calculate completion rates

### 3. usePerformanceData

- Analyze performance trends
- Calculate averages
- Track improvements

## 🔒 State Management

- React hooks for local state
- Custom hooks for data fetching
- Supabase real-time subscriptions

## 🎨 Styling

The project uses Tailwind CSS for styling with:

- Responsive design
- Custom color schemes

- Consistent spacing
- Modern UI components

## Deployment

1. Build the project:

```
npm run build
```

2. Preview the build:

```
npm run preview
```

3. Deploy to your preferred hosting platform

## Future Enhancements

- Advanced analytics dashboard
- Real-time notifications
- PDF report generation
- Mobile application
- Integration with LMS platforms

## Contributing

1. Fork the repository
2. Create a feature branch
3. Commit your changes
4. Push to the branch
5. Open a pull request

## License

This project is licensed under the MIT License - see the LICENSE file for details.

## Authors

- Your Name - Initial work

## Acknowledgments

- React team
- Supabase team
- Tailwind CSS team
- Open source community