🚀 Railway Deployment Guide - AI Product Calculator

Complete Step-by-Step Deployment Instructions

📋 Pre-Deployment Checklist

- [x] ✅ ESLint version conflicts resolved (downgraded to 8.57.0)
- [x] ✅ TypeScript ESLint parser/plugin compatibility fixed
- [x] ✅ Nixpacks configuration optimized for Railway
- [x] ✅ Build process tested and working
- [x] ✅ Prisma schema properly configured
- [x] ✅ Environment variables documented

🔧 1. Upload Project to GitHub

```
# Initialize git repository (if not already done)
git init
git add .
git commit -m "Initial commit - AI Product Calculator"

# Push to GitHub
git remote add origin https://github.com/yourusername/ai-product-calculator.git
git branch -M main
git push -u origin main
```

🗄️ 2. Railway Database Setup

1. **Create New Project in Railway**
   - Go to railway.app (https://railway.app)
   - Click “New Project”
   - Choose “Empty Project”

2. **Add PostgreSQL Database**
   - Click “Add Service”
   - Select “Database” → “PostgreSQL”
   - Wait for database to deploy
   - Note the generated database URL

3. **Get Database Connection String**
   - Click on the PostgreSQL service
   - Go to “Variables” tab
   - Copy the `DATABASE_URL` value
   - It should look like: `postgresql://postgres:password@hostname:port/railway`

🌐 3. Deploy Web Application

1. **Add Web Service**
   - In the same Railway project, click “Add Service”
   - Select “GitHub Repo”

- Connect your GitHub repository
- Select the repository with your AI Product Calculator

2. **Configure Environment Variables**
   `bash`
   ```
   # Required Variables
   DATABASE_URL=postgresql://postgres:password@hostname:port/railway
   NEXTAUTH_URL=https://your-app-name.up.railway.app
   NEXTAUTH_SECRET=your-super-secret-32-character-string
   NODE_ENV=production
   ```

How to generate NEXTAUTH_SECRET:
`bash`
```
# In terminal/command prompt:
openssl rand -base64 32
# OR use any 32+ character random string
```

1. **Set Variables in Railway**
   - Click on your web service
   - Go to "Variables" tab
   - Click "Add Variable" for each:
     - `DATABASE_URL` → Paste the PostgreSQL URL from step 2
     - `NEXTAUTH_URL` → `https://your-app-name.up.railway.app`
     - `NEXTAUTH_SECRET` → Generated secret from above
     - `NODE_ENV` → `production`

🎯 4. Deploy and Verify

1. **Trigger Deployment**
   - Railway should automatically start building after you add variables
   - Or click "Deploy" to manually trigger

2. **Monitor Build Process**
   - Watch the build logs in Railway dashboard
   - Expected phases:
   ```
   ✅ Install: npm ci + prisma generate
       ✅ Build: npm run build
       ✅ Start: npm start
   ```

3. **Verify Deployment**
   - Once deployed, click on the generated Railway URL
   - You should see the AI Product Calculator landing page
   - Test the calculator flow: Landing → Calculator → Results

🔍 5. Troubleshooting Guide

Build Fails at Install Phase:

```
# Check if these variables are set:
- DATABASE_URL (from PostgreSQL service)
- NODE_ENV=production
```

Build Fails at Prisma Generate:

```
# Ensure DATABASE_URL is valid and PostgreSQL service is running
# Check Railway PostgreSQL service status
```

App Starts but Database Errors:

```
# Verify DATABASE_URL format:
postgresql://username:password@host:port/database

# Check PostgreSQL service is active in Railway
```

404 on Calculator Pages:

```
# Ensure all environment variables are set
# Check NEXTAUTH_URL matches your Railway app URL
```

📊 6. Expected Results

After successful deployment:

- **Landing Page**: `https://your-app.up.railway.app/`
- **Calculator**: `https://your-app.up.railway.app/calculator`
- **API Endpoints**: All calculator APIs functional
- **Database**: PostgreSQL tables auto-created via Prisma

🔄 7. Making Updates

```
# Make code changes locally
git add .
git commit -m "Update description"
git push origin main

# Railway will automatically redeploy
```

🎉 8. Success Indicators

✅ **Build Completed Successfully**
✅ **App responding on Railway URL**
✅ **Landing page loads properly**
✅ **Calculator form works**
✅ **Results page displays**
✅ **Email capture functions**
✅ **Database saves calculations**

📞 9. Support

If deployment fails after following this guide:

1. Check Railway build logs for specific errors
2. Verify all environment variables are correctly set
3. Ensure PostgreSQL database service is running
4. Confirm GitHub repository connection is active

Common Success Patterns:
- Build time: 2-4 minutes
- First deployment: May take 5-10 minutes
- Subsequent deployments: 1-2 minutes

🚀 **Your AI Product Calculator will be live and ready to capture leads!**