**🚀 Mivton Project - Phase 1.2 Context Prompt**

**I am working on MIVTON - a futuristic multilingual chat platform. You are my technical developer. We have completed Phase 1.1 and now need to implement Phase 1.2 - Database Setup.**

**🎯 Project Overview**

* **Name**: Mivton
* **Purpose**: Friends-only multilingual text chat with real-time OpenAI translation
* **Target**: Gen Z users with futuristic design
* **Status**: Phase 1.1 ✅ COMPLETE, Phase 1.2 🔄 IN PROGRESS

**✅ Phase 1.1 - COMPLETED (July 30, 2025)**

* **Landing Page**: Live at https://mivton-production.up.railway.app
* **Domain**: mivton.com (connected, DNS propagating)
* **Backend**: Express.js server with Socket.IO ready
* **Design**: Futuristic Gen Z aesthetic with glassmorphism
* **Infrastructure**: Railway deployment with environment variables
* **Features**: Waitlist modal, chat preview, animations, mobile responsive

**🔄 CURRENT PHASE 1.2 - Database Setup (Day 1-2)**

**Goal**: Set up PostgreSQL database with core tables and connections **Priority**: IMPLEMENT THIS PHASE

**Phase 1.2 Tasks:**

* [ ] Set up PostgreSQL connection (Railway)
* [ ] Create users table with all required fields
* [ ] Implement session management with PostgreSQL store
* [ ] Create database connection utilities
* [ ] Test: Database connectivity and basic queries

**Required Database Tables for Phase 1.2:**

-- Core users table

CREATE TABLE users (

id SERIAL PRIMARY KEY,

username VARCHAR(20) UNIQUE NOT NULL,

email VARCHAR(255) UNIQUE NOT NULL,

password\_hash VARCHAR(255) NOT NULL,

full\_name VARCHAR(100) NOT NULL,

gender VARCHAR(10) NOT NULL,

native\_language VARCHAR(10) NOT NULL,

is\_verified BOOLEAN DEFAULT TRUE,

is\_admin BOOLEAN DEFAULT FALSE,

admin\_level INTEGER DEFAULT 0,

is\_blocked BOOLEAN DEFAULT FALSE,

status VARCHAR(20) DEFAULT 'offline',

last\_login TIMESTAMP DEFAULT NULL,

created\_at TIMESTAMP DEFAULT CURRENT\_TIMESTAMP,

updated\_at TIMESTAMP DEFAULT CURRENT\_TIMESTAMP

);

-- Session management

CREATE TABLE session (

sid VARCHAR NOT NULL PRIMARY KEY,

sess JSON NOT NULL,

expire TIMESTAMP(6) NOT NULL

);

**🔧 Technical Infrastructure (Already Set Up)**

**Railway Environment Variables ✅**

NODE\_ENV=production

JWT\_SECRET=mivton-super-secret-jwt-key-2025-production

OPENAI\_API\_KEY=sk-proj-ssWG4RYWzRSkC6N5wSrwR-ajRzmcGMLG2agNfBO1IRpplc7a8LzwmHrKNeShj4J2gm8ynWDu\_2T3BlbkFJob8\_1Ny3bs5vVdEKNo48BEsXZOB4kGzgtQ-O0-JAssePAFHU7CFAi-cLPc-xYop2z362vHtd4A

SMTP\_HOST=smtp.hostinger.com

SMTP\_PORT=465

SMTP\_SECURE=true

SMTP\_USER=info@mivton.com

SMTP\_PASS=Bacau@2012

APP\_URL=https://mivton.com

FRONTEND\_URL=https://mivton.com

**Current File Structure ✅**

mivton/

├── server.js # Main Express server ✅

├── package.json # Dependencies ✅

├── .env # Environment variables ✅

├── public/

│ ├── index.html # Landing page ✅

│ ├── css/style.css # Futuristic styles ✅

│ └── js/app.js # Frontend logic ✅

├── database/ # 🔄 NEEDS SETUP

├── routes/ # 🔄 NEEDS SETUP

├── middleware/ # 🔄 NEEDS SETUP

└── utils/ # 🔄 NEEDS SETUP

**🎨 Design System (Established)**

Primary: #6366f1 (Electric Blue)

Secondary: #8b5cf6 (Vibrant Purple)

Accent: #06b6d4 (Cyan)

Success: #10b981 (Green)

Warning: #f59e0b (Amber)

Error: #ef4444 (Red)

Background: #0f172a (Dark Navy)

Surface: #1e293b (Slate)

Text: #f1f5f9 (Light)

**💻 Technology Stack**

* **Backend**: Node.js + Express.js ✅
* **Database**: PostgreSQL (Railway) 🔄 SETUP NEEDED
* **Real-time**: Socket.IO ✅
* **Translation**: OpenAI GPT-4 (ready)
* **Email**: Hostinger SMTP (configured)
* **Frontend**: HTML5 + CSS3 + Vanilla JS ✅
* **Deployment**: Railway ✅

**📋 Phase 1.2 Specific Requirements**

**Database Connection Setup Needed:**

1. **PostgreSQL Connection**: Connect to Railway PostgreSQL instance
2. **Connection Pool**: Implement efficient connection pooling
3. **Error Handling**: Robust database error handling
4. **Migration System**: Basic schema setup and management
5. **Connection Testing**: Health check for database connectivity

**Files to Create in Phase 1.2:**

database/

├── connection.js # PostgreSQL connection setup

├── schema.sql # Core database schema

└── init.js # Database initialization

middleware/

└── database.js # Database middleware

utils/

└── database.js # Database utilities

**Dependencies to Add:**

* pg (PostgreSQL client)
* express-session (session management)
* connect-pg-simple (PostgreSQL session store)

**🔮 Phase 1.3 Preview (Authentication System)**

**After Phase 1.2, we'll implement:**

* User registration with validation
* Secure login with bcrypt
* Session management
* Futuristic login/register pages
* Basic email verification setup
* Complete user registration and login flow

This context helps prepare database structure for authentication needs.

**👥 Our Roles**

* **You (Claude)**: Technical developer - create database schema, connection files, middleware
* **Me**: Can run terminal commands, deploy to Railway, copy/paste code files

**🚀 What I Need Help With**

I need you to help me implement Phase 1.2 - Database Setup. Specifically:

1. **Database connection files** - PostgreSQL setup for Railway
2. **Core schema creation** - Users and session tables
3. **Connection utilities** - Pooling and error handling
4. **Integration with existing server.js** - Add database middleware
5. **Testing instructions** - How to verify database connectivity

Please provide the exact files and terminal commands I need for Phase 1.2!

**END OF PROMPT - Ready for Phase 1.2 Database Setup Implementation**