

# Quick Setup Guide

---

## Step-by-Step Installation

### 1. Database Setup (Supabase)

```
# 1. Go to https://supabase.com and sign up/login
# 2. Click "New Project"
# 3. Fill in:
#   - Name: cafe-management (or your choice)
#   - Database Password: (save this!)
#   - Region: Choose closest to you
# 4. Wait 2-3 minutes for provisioning

# 5. Once ready, go to SQL Editor (left sidebar)
# 6. Click "New query"
# 7. Open database_schema.sql in a text editor
# 8. Copy ALL contents and paste into Supabase SQL Editor
# 9. Click "Run" or press Ctrl+Enter
# 10. Verify tables were created in Table Editor

# 11. Get your connection string:
#   - Go to Settings (gear icon) > Database
#   - Find "Connection string" section
#   - Copy the "URI" format
#   - Replace [YOUR-PASSWORD] with your database password
```

### 2. Backend Setup

```
# Navigate to backend folder
cd backend

# Create virtual environment
python -m venv venv

# Activate virtual environment
.\venv\Scripts\Activate.ps1

# Install dependencies
pip install -r requirements.txt

# Create .env file from example
Copy-Item .env.example .env

# Edit .env and update DATABASE_URL with your Supabase connection string:
# Go to Supabase Dashboard > Settings > Database > Connection string (URI)
# DATABASE_URL=postgresql://postgres:[YOUR-PASSWORD]@db.[PROJECT-
```

```
REF].supabase.co:5432/postgres
```

```
# Run the backend server
```

```
uvicorn app.main:app --reload --host 0.0.0.0 --port 8000
```

#### ☑ Verify Backend is Running:

- Open <http://localhost:8000> - Should show API info
- Open <http://localhost:8000/api/v1/docs> - Should show Swagger documentation
- Open <http://localhost:8000/health> - Should return `{"status": "healthy"}`

Backend API will be running at: <http://localhost:8000>

API Documentation: <http://localhost:8000/api/v1/docs>

### 3. Frontend Setup

```
# Open a NEW terminal window
```

```
# Navigate to frontend folder
```

```
cd frontend
```

```
# Install dependencies
```

```
npm install
```

```
# Create .env file
```

```
"VITE_API_URL=http://localhost:8000/api/v1" | Out-File -FilePath .env -Encoding utf8
```

```
# Run the development server
```

```
npm run dev
```

#### ☑ Verify Frontend is Running:

- Open <http://localhost:3000> - Should show login page
- Check terminal - No errors should appear
- Press F12 in browser - Console should be clean (no red errors)

Frontend will be running at: <http://localhost:3000>

### 4. Create Your First User

Option A: Using the API Docs (Recommended)

1. Go to <http://localhost:8000/api/v1/docs>
2. Expand `POST /api/v1/auth/register`
3. Click "Try it out"
4. Enter:

```
{
  "email": "admin@cafe.com",
  "password": "your_secure_password",
  "full_name": "Admin User"
}
```

5. Click "Execute"

Option B: Using curl

```
curl -X POST http://localhost:8000/api/v1/auth/register `
-H "Content-Type: application/json" `
-d
'{"email\":\"admin@cafe.com\", \"password\":\"your_password\", \"full_name\":\"Admin User\"}'
```

## 5. Login to the Application

1. Go to `http://localhost:3000`
2. Login with your credentials
3. Create your first cafe

## Common Issues & Solutions

Issue: "psycopg2-binary" installation fails

**Solution:** Install Visual C++ Build Tools from Microsoft

Issue: "Cannot connect to database"

**Solution:**

- Verify your Supabase project is active (green status in dashboard)
- Check DATABASE\_URL in backend/.env has correct password
- Ensure you're using the URI format from Settings > Database
- Test connection from Supabase SQL Editor with `SELECT 1`
- Check your IP isn't blocked (Supabase allows all IPs by default)

Issue: "Module not found" in frontend

**Solution:**

```
cd frontend
rm -rf node_modules
npm install
```

Issue: CORS errors

**Solution:** Check that ALLOWED\_ORIGINS in backend/.env includes your frontend URL

## Development Tips

1. **Backend Auto-reload:** FastAPI automatically reloads when you change Python files
2. **Frontend Hot Reload:** Vite provides instant updates when you change React files
3. **Database Changes:** After modifying schema, paste updated SQL into Supabase SQL Editor and run
4. **API Testing:** Use the Swagger UI at <http://localhost:8000/api/v1/docs>

## Project Structure Overview

```
Cafe_Gestion/
├── database_schema.sql      # PostgreSQL database schema
├── README.md                # Main documentation
├── SETUP.md                # This file
├── backend/                 # FastAPI backend
│   ├── app/
│   │   ├── main.py         # Application entry point
│   │   ├── api/v1/endpoints/ # API route handlers
│   │   ├── core/           # Configuration & security
│   │   ├── models/         # Database models
│   │   └── schemas/        # Request/response schemas
│   ├── requirements.txt     # Python dependencies
│   ├── .env.example        # Environment template
│   └── .env                 # Your environment (create this)
└── frontend/               # React frontend
    ├── src/
    │   ├── main.tsx        # Application entry point
    │   ├── App.tsx         # Main app component
    │   ├── api/client.ts   # API integration
    │   ├── store/          # State management
    │   ├── components/     # Reusable UI components
    │   └── pages/          # Page components
    ├── package.json        # Node dependencies
    ├── vite.config.ts      # Vite configuration
    └── .env                 # Your environment (create this)
```

## Next Steps

1. ☒ Complete setup steps above
2. ☒ Create your first user
3. ☒ Login and create a cafe
4. ☒ Add stock items
5. ☒ Create menu items with recipes
6. ☒ Add staff members
7. ☒ Start recording daily sales

8. ☒ View reports and track profitability

## Need Help?

- API Documentation: <http://localhost:8000/api/v1/docs>
- Check terminal logs for errors
- Verify all environment variables are set correctly
- Ensure PostgreSQL, Python, and Node.js are properly installed

---

Happy coding! 🐼