🚀 Deploy VCTT-AGI to Render.com NOW

Time Required: 5 minutes
Repository: https://github.com/Counterbalance-Economics/vctt-agi-engine

---

Step 1: Create Web Service (2 minutes)

1. **Open Render Dashboard**: https://dashboard.render.com
   - Sign in with your GitHub account

2. **Click “New +”** (top right) → Select **“Web Service”**

3. **Connect GitHub Repository**:
   - Click “Build and deploy from a Git repository”
   - Click “Next”
   - Find: **Counterbalance-Economics/vctt-agi-engine**
   - Click “Connect”

Step 2: Configure Service (2 minutes)

Basic Settings:

```
Name: vctt-agi-backend
Region: Oregon (US West) - or closest to you
Branch: master
Root Directory: nodejs_space
Runtime: Node
```

Build & Deploy:

```
Build Command: yarn install && yarn build
Start Command: yarn start:prod
```

Instance Type:

```
Free (for testing) or Starter ($7/month for production)
```

Step 3: Add PostgreSQL Database (1 minute)

Option A: Create New Database
1. Scroll down to “Environment” section
2. Click “Add Database” or “New PostgreSQL”
3. Name: `vctt-agi-db`

4. Plan: Free (for testing)
5. Click “Create Database”

Render will auto-configure `DATABASE_URL` for you!

Option B: Use Platform Database
- The DATABASE_URL is already configured in the current environment
- You can use: `postgresql://role_18ace863b:4INg7wn17TfboFK9i-fRD6CFi_9HEPgcK@db-18ace863b.db003.hosteddb.reai.io:5432/18ace863b`

Step 4: Set Environment Variables

Click “Advanced” → “Add Environment Variable”

Required:

```
OPENAI_API_KEY = your-openai-key-here
```

Optional (already have defaults):

```
PORT = 8000
NODE_ENV = production
OPENAI_MODEL = gpt-4
OPENAI_TEMPERATURE = 0.7
```

Step 5: Deploy! 🚀

1. Click **“Create Web Service”**
2. Wait 3-5 minutes for deployment
3. Render will:
   - Clone the repository
   - Install dependencies
   - Build TypeScript
   - Start the server
   - Give you a URL: `https://vctt-agi-backend.onrender.com`

Step 6: Verify Deployment

Once deployed, test these endpoints:

1. Health Check:

```
curl https://your-app-name.onrender.com/health
```

Expected response:

```
{
  "status": "ok",
  "timestamp": "2025-11-17T...",
  "uptime": 123.45,
  "database": "connected"
}
```

2. API Documentation:
Open in browser: `https://your-app-name.onrender.com/api-docs`

3. Create Session:

```
curl -X POST https://your-app-name.onrender.com/api/v1/session/start \
  -H "Content-Type: application/json" \
  -d '{"user_id":"test_user","input":"Hello VCTT!"}'
```

🎯 Expected Deployment URL

After deployment, your backend will be at:

```
https://vctt-agi-backend.onrender.com
```

Save this URL! You'll need it to update the UI.

🐛 Troubleshooting

If build fails:
1. Check Render logs (click "Logs" tab)
2. Verify Node version is 18+ (should auto-detect from `.node-version`)
3. Ensure `nodejs_space` is set as root directory

If database connection fails:
1. Verify PostgreSQL database is attached
2. Check `DATABASE_URL` environment variable exists
3. Ensure database is in the same region as web service

If API returns 500 errors:
1. Check `OPENAI_API_KEY` is set correctly
2. View logs for error details
3. Verify all environment variables are set

📱 Update UI After Backend Deploys

Once backend is live, update the UI:

```
cd /home/ubuntu/vctt_agi_ui
# Update .env or set during deployment
export VITE_API_URL=https://your-backend-url.onrender.com
vercel --prod
```

✅ Success Checklist

After deployment:
- [ ] `/health` endpoint returns `{"status":"ok"}`
- [ ] `/api-docs` shows Swagger documentation
- [ ] Can create a session via POST `/api/v1/session/start`
- [ ] Can retrieve session history
- [ ] Analytics endpoints work
- [ ] Database persists data across requests

🎉 You’re Done!

Once deployed:
1. Copy your backend URL
2. Test the endpoints
3. Update the UI with the new backend URL
4. Test the full application

Your VCTT-AGI Engine will be live in production! 🚀

Need Help? Paste any error messages and I’ll help troubleshoot.