

Seed API Endpoint Usage Guide

Overview

A secure API endpoint has been created to trigger database seeding on your Render deployment. This allows you to seed the database without needing shell access.

What Was Implemented

1. New Files Created

- `backend/src/controllers/seed.controller.ts` : Controller with authentication logic and seed execution
- `backend/src/routes/seed.routes.ts` : Route definitions for the seed API

2. Modified Files

- `backend/src/routes/index.ts` : Integrated seed routes under `/api/seed`
- `backend/src/app.ts` : Added seed endpoint to API documentation

Endpoints

Health Check

```
GET https://workigom-backend.onrender.com/api/seed/health
```

Purpose: Check if the seed service is available and configured

Response:

```
{
  "success": true,
  "message": "Seed service is available",
  "configured": true,
  "timestamp": "2025-10-25T20:15:30.123Z"
}
```

Run Seed

```
POST https://workigom-backend.onrender.com/api/seed/run
```

Purpose: Trigger database seeding

Headers Required:

- `x-seed-secret` : `workigom_seed_2024` (matches your `SEED_SECRET` environment variable)

Response on Success:

```
{
  "success": true,
  "message": "Database seeded successfully",
  "timestamp": "2025-10-25T20:15:30.123Z",
  "output": "🌱 Seeding database...\n✅ Admin user created...",
  "warnings": null
}
```

Response on Authentication Failure:

```
{
  "success": false,
  "message": "Unauthorized: Missing x-seed-secret header"
}
```

How to Use

Option 1: Using cURL (Command Line)

```
curl -X POST https://workigom-backend.onrender.com/api/seed/run \
  -H "x-seed-secret: workigom_seed_2024" \
  -H "Content-Type: application/json"
```

Option 2: Using Postman

1. Create a new POST request
2. URL: `https://workigom-backend.onrender.com/api/seed/run`
3. Go to **Headers** tab
4. Add header:
 - Key: `x-seed-secret`
 - Value: `workigom_seed_2024`
5. Click **Send**

Option 3: Using JavaScript/Fetch

```
fetch('https://workigom-backend.onrender.com/api/seed/run', {
  method: 'POST',
  headers: {
    'x-seed-secret': 'workigom_seed_2024',
    'Content-Type': 'application/json'
  }
})
.then(response => response.json())
.then(data => console.log('Seed result:', data))
.catch(error => console.error('Error:', error));
```

Option 4: Using Python

```
import requests

url = 'https://workigom-backend.onrender.com/api/seed/run'
headers = {
    'x-seed-secret': 'workigom_seed_2024',
    'Content-Type': 'application/json'
}

response = requests.post(url, headers=headers)
print(response.json())
```

Deployment Status

✓ Changes Committed and Pushed

- Commit: feat: add secure API endpoint for database seeding
- Branch: master
- Repository: <https://github.com/volkanakbulut73/workigom.git>

Next Steps for You

1. Wait for Render to Deploy

- Render will automatically detect the new commit and start deployment
- Check your Render dashboard: <https://dashboard.render.com>
- Look for the “workigom-backend” service
- Wait for the “Live” status

2. Verify the Endpoint is Available

```
bash
curl https://workigom-backend.onrender.com/api/seed/health
```

Expected response:

```
json
{
  "success": true,
  "message": "Seed service is available",
  "configured": true
}
```

1. Run the Database Seed

```
bash
curl -X POST https://workigom-backend.onrender.com/api/seed/run \
-H "x-seed-secret: workigom_seed_2024"
```

2. Verify the Data was Created

- Log into your application at <https://workigom.vercel.app>
- Try logging in with test credentials:
 - **Admin:** admin@workigom.com / admin123
 - **Company:** company1@workigom.com / company123
 - **User:** mehmet@example.com / user123

What the Seed Creates

The seed script creates:

- **1 Admin user** (admin@workigom.com)
- **4 Corporate users** (Companies/Employers/Donors)
- **5 Individual users** (Job seekers/Donation receivers)
- **5 Jobs** across different categories
- **5 Applications** for the jobs
- **5 Donations** of various food types
- **4 Messages** between users
- **6 Notifications** for various events

Security Features

1. **Authentication Required:** The endpoint requires the `x-seed-secret` header
2. **Environment Variable:** The secret is stored securely in Render's environment variables
3. **No Credentials Exposed:** The seed secret is not hardcoded in the codebase
4. **Error Handling:** Comprehensive error handling and logging
5. **Non-Production Warning:** Consider disabling this endpoint in production or using a different approach

Troubleshooting

Issue: "Unauthorized: Missing x-seed-secret header"

Solution: Make sure you're including the `x-seed-secret` header in your request

Issue: "Forbidden: Invalid seed secret"

Solution: Verify that `SEED_SECRET` is set to `workigom_seed_2024` in Render's environment variables

Issue: "Server configuration error: SEED_SECRET not configured"

Solution:

1. Go to Render dashboard
2. Select your backend service
3. Go to Environment section
4. Add/verify: `SEED_SECRET=workigom_seed_2024`
5. Restart the service

Issue: "Failed to seed database"

Solution:

1. Check Render logs for specific error messages
2. Verify `DATABASE_URL` is correctly set
3. Ensure database is accessible
4. Check if migrations have been run

Best Practices

1. **Run Once:** You typically only need to run this once after deployment

2. **Idempotent:** The seed script uses `upsert` operations, so running it multiple times won't create duplicates for some entities
3. **Backup First:** If running on a database with existing data, consider backing up first
4. **Monitor Logs:** Check Render logs while seeding to catch any issues early
5. **Verify Results:** Always verify the seeded data by logging into the application

Example Full Workflow

```
# 1. Check if backend is live
curl https://workigom-backend.onrender.com/api/health

# 2. Check if seed endpoint is configured
curl https://workigom-backend.onrender.com/api/seed/health

# 3. Run the seed
curl -X POST https://workigom-backend.onrender.com/api/seed/run \
  -H "x-seed-secret: workigom_seed_2024" \
  -H "Content-Type: application/json"

# 4. Wait for completion (usually 10-30 seconds)

# 5. Test login at https://workigom.vercel.app
# Use: admin@workigom.com / admin123
```

Support

If you encounter any issues:

1. Check Render logs for detailed error messages
2. Verify all environment variables are set correctly
3. Ensure database is accessible and migrations are applied
4. Review the seed script at `backend/prisma/seed.ts` for data details

Note: Once your database is seeded and the application is working, you may want to remove or secure this endpoint further for production use.