Troubleshooting

Common Issues

1. Database Connection Errors

Problem: Cannot connect to PostgreSQL/MongoDB

```
ERROR NOT CONNECTED to the database!
```

Solutions:

- Verify database is running
- Check connection credentials in `.env`
- Ensure firewall allows database port
- Check database host/port configuration

2. Redis Connection Failed

Problem: Redis client connection error

```
Redis Error - Connection refused
```

Solutions:

- Start Redis server: `redis-server`
- Check Redis password configuration
- Verify Redis port (default: 6379)
- Check if Redis is running: `redis-cli ping`

3. Minio Connection Issues

Problem: Cannot connect to Minio

```
Minio Error - Connection refused
```

Solutions:

- Start Minio server

- Verify endpoint, port, and credentials
- Check bucket exists
- Ensure SSL configuration matches

4. File Upload Failures

Problem: Files not uploading or temp directory errors

Solutions:

- Ensure temp directories exist and have write permissions

```
mkdir -p resources/temp/upload
mkdir -p resources/temp/review
mkdir -p resources/temp/pdf
chmod -R 755 resources/
```

- Check disk space
- Verify Multer configuration

5. JWT Token Invalid

Problem: Authentication fails with valid-looking token

Solutions:

- Verify JWT_SECRET matches token generation
- Check token expiration
- Ensure token format: `Bearer <token>`
- Verify token signature

6. Scheduled Tasks Not Running

Problem: Cron jobs not executing

Solutions:

- Check node-cron syntax
- Verify server timezone
- Check logs for errors
- Ensure process stays running (use PM2)

7. WebSocket Connection Failed

Problem: Chat not working, Socket.IO errors

Solutions:

- Check CORS configuration
- Verify WebSocket port is open
- Check Nginx/proxy configuration for WebSocket support
- Ensure HTTP server is created correctly

8. Payment Webhook Not Receiving

Problem: Xendit webhooks not being received

Solutions:

- Verify webhook URL is publicly accessible
- Check callback token matches
- Examine webhook logs in Xendit dashboard
- Test with webhook testing tools
- Ensure HTTPS in production

Debugging Tips

1. **Enable Debug Logging**

```
// In index.js
const morgan = require('morgan');
app.use(morgan('combined'));
```

2. **Check Database Queries**

```
// Sequelize logging enabled by default in development
// Check console for SQL queries
```

3. **Monitor Process**

```
pm2 logs onmarket-api
pm2 monit
```

4. **Check Port Availability**

```
lsof -i :3610
netstat -tulpn | grep 3610
```

5. **Test Database Connections**

```
# PostgreSQL
psql -h localhost -U onmarket_user -d onmarket_db
```

```
# MongoDB
mongo mongodb://localhost:27017/onmarket

# Redis
redis-cli -h localhost -p 6379 ping
```

Revision #1
Created 24 February 2026 09:30:08 by ondeliveloper
Updated 24 February 2026 09:30:51 by ondeliveloper