

# Dokumentation Mongoose

Julia Pabst

## Install and sign up for Docker





The screenshot shows the Docker Desktop application window. The top bar is blue with the Docker logo and a search bar. The left sidebar contains navigation options: Containers, Images, Volumes, Builds, Docker Scout, Extensions, and Add Extensions. The main area is titled 'Containers' and displays system usage statistics for CPU and memory. Below these are search and filter controls. A table lists the running containers, showing one container named 'backend' in a 'Running' state. At the bottom, there is a 'Walkthroughs' section with two cards: 'Multi-container applications' and 'Containerize your application'. The status bar at the very bottom indicates the engine is running and provides system resource information.

**Containers** [Give feedback](#)

Container CPU usage ⓘ **0.68%** / 1200% (12 CPUs available)


Container memory usage ⓘ **77.57MB** / 7.48GB [Show charts](#)


Search  ☐ Only show running containers

<input type="checkbox"/>	Name	Image	Status	Port(s)	CPU (%)	Last started	Actions
<input type="checkbox"/>	>  <b>backend</b>		Running (1/1)		0.68%	32 seconds ago	  




Showing 1 item

**Walkthroughs** ×


 **Multi-container applications**  
8 mins

 **Containerize your application**  
3 mins

[View more in the Learning center](#)

Engine running    RAM 1.57 GB CPU 0.00% Disk 56.77 GB avail. of 62.67 GB Signed in New version available 1

## Docker Setup für MongoDB

```
backend >  docker-compose.yml
1  version: '3.8'
2  services:
3    mongodb:
4      image: mongo:latest
5      container_name: mongodb
6      ports:
7        - "27017:27017"
8      environment:
9        MONGO_INITDB_ROOT_USERNAME: root
10       MONGO_INITDB_ROOT_PASSWORD: example
11      volumes:
12        - mongo-data:/data/db
13
14  volumes:
15    mongo-data:
16
```

# Run docker-compose up -d in terminal

```
> testadmin@Julias-MBP-2 backend % docker-compose up -d
```

```
WARN[0000] /Users/testadmin/Desktop/Knowledge_Centre/FH_Technikum/Semester2/Webframeworks/Highscore/einheit4/backend/docker-compose.yml: `version` is obsolete
```

```
[+] Running 7/9
```

:: mongodb [■■■■■■■■■■] 175.8MB / 265.5MB Pulling	44.0s
✓ 7646c8da3324 Pull complete	28.0s
✓ 11f24d004130 Pull complete	28.1s
✓ 1dfef8821fc6 Pull complete	28.5s
✓ bd2b135d9110 Pull complete	28.6s
✓ 76154bf34598 Pull complete	28.7s
✓ a93134489020 Pull complete	28.7s
:: fd1fa78ffcfb Downloading [=====>	] 143.7...
✓ d91b3273507c Download complete	3.4s

# Install Mongoose in project folder

```
● testadmin@Julias-MBP-2 backend % npm install mongoose

npm warn deprecated @npmcli/move-file@2.0.1: This functionality has been moved to @npmcli/fs

added 994 packages, and audited 995 packages in 20s

99 packages are looking for funding
  run `npm fund` for details

15 vulnerabilities (6 moderate, 8 high, 1 critical)

To address all issues, run:
  npm audit fix

Run `npm audit` for details.
```

# Create db.js file

backend > JS db.js > ...

```
1  const mongoose = require('mongoose');
2
3  const connectDB = async () => {
4    try {
5      await mongoose.connect('mongodb://root:example@localhost:27017/highscoredb?authSource=admin', {
6        useNewUrlParser: true,
7        useUnifiedTopology: true,
8        useCreateIndex: true
9      });
10     console.log('MongoDB connected');
11   } catch (err) {
12     console.error(err.message);
13     process.exit(1);
14   }
15 };
16
17 module.exports = connectDB;
18
```

# Connect to database in app.js

backend > JS app.js > ...

```
1  const express = require("express");
2  const cors = require('cors');
3  const connectDB = require('./db');
4  const app = express();
5
6  // Datenbankverbindung herstellen
7  connectDB();
8
9  app.use(cors());
10 app.use(express.json());
11
12 const User = require('./models/userModel');
13 const { HighScore, HighScoreListe } = require('./models/highscoreModel');
14
15 // Beispiel-Routen für Benutzer- und Highscore-Verwaltung
16 app.post("/users", async (req, res) => {
17   const { email, password, address, city, zipCode } = req.body;
18   if (!email || !password || !address || !city || !zipCode) {
```

# run server.js and see how it connects to MongoDB

```
testadmin@Julias-MBP-2 backend % node server.js
Server started
(node:80941) [MONGODB DRIVER] Warning: useNewUrlParser is a deprecated option: useNewUrlParser has no effect since Node.js Driver version 4.0.0 and will be removed in the next major version
(Use `node --trace-warnings ...` to show where the warning was created)
(node:80941) [MONGODB DRIVER] Warning: useUnifiedTopology is a deprecated option: useUnifiedTopology has no effect since Node.js Driver version 4.0.0 and will be removed in the next major version
MongoDB connected
```