

MANVSim: Simulation eines Massenanfalls von Verletzten

Bachelor-Abschlussprojekt/Masterprojekt SoSe 2024

Jon Stührwoldt, Lukas Voigt, Peter Gausmann, Yannick Illmann, Simon Ohlsen

27. September 2024

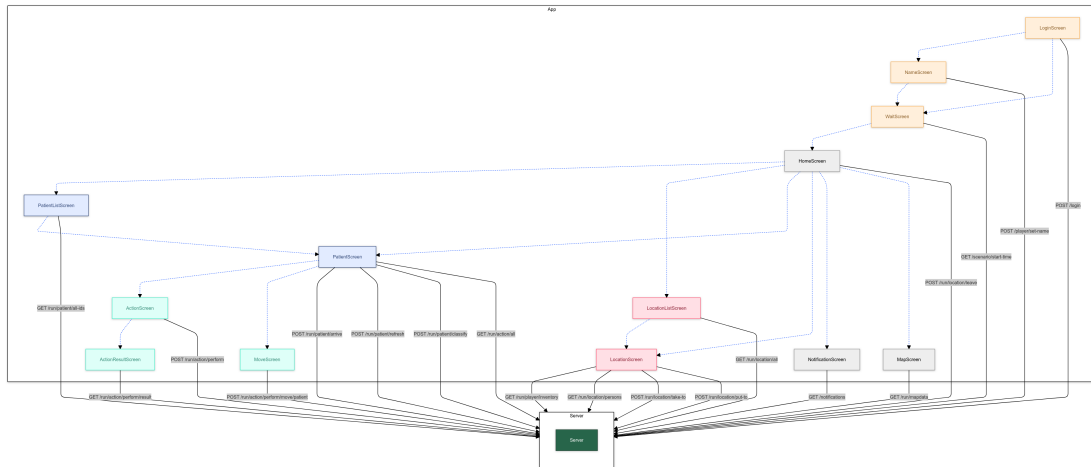
Anwendung MANVSim

- Webanwendung
 - Simulationsverwaltung
 - Stammdatenverwaltung
- Mobile (Web-) App
- Vorstellung:
 - Koordinierungs- und Adminpanels
 - Wichtige Ansichten der App
- MANV-Übung:
 - Übung vorbereitet
 - Thema: Halloween
 - Rettungskräfte gesucht

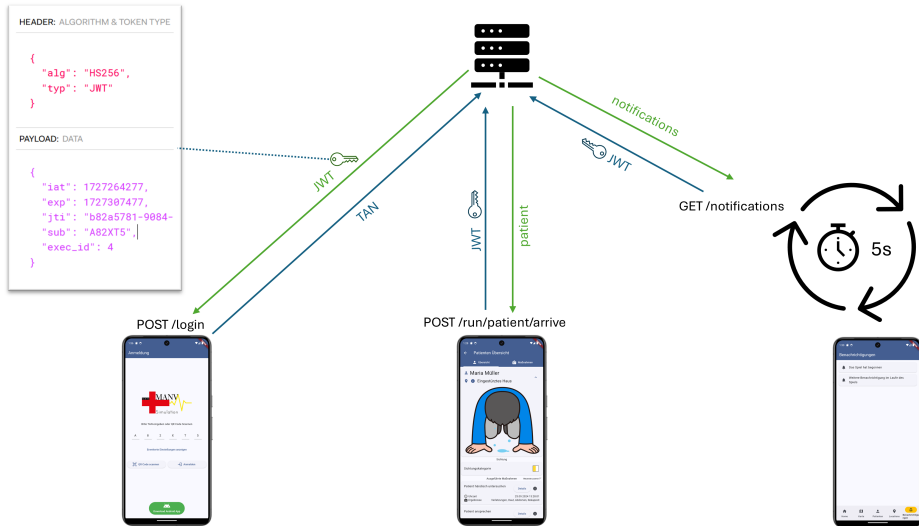
Wechsel zur Anwendung

App - Kommunikation mit Server

- Kommunikation mit Server über REST-API
- 19 API Endpunkte



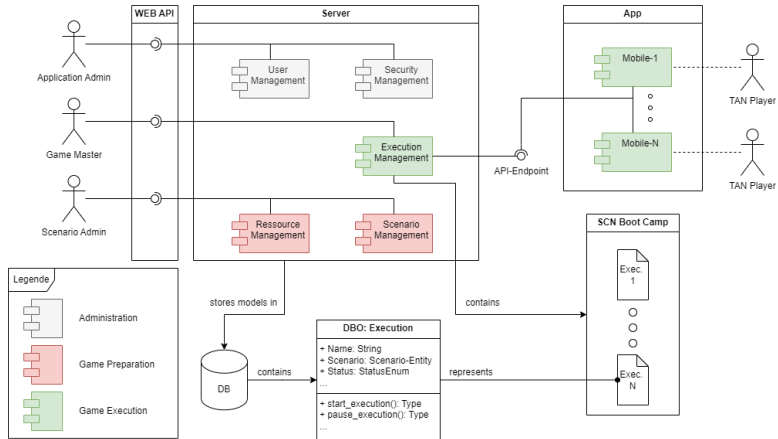
App - Kommunikation mit Server



Python Server mit Flask API

- stellt Endpunkte zum Persistieren der Stammdaten
- stellt Endpunkte zur Live Administration einer Übung
- stellt Endpunkte zur Teilnahme an einer Übung

Backend-Komponenten



Trennung von Basis- und Spieldaten

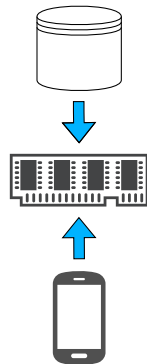
- In-memory Objekte für Spieldaten
- Datenbank für Template-Objekte

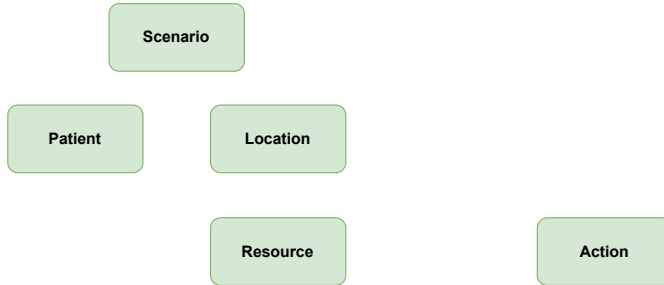
Vorteile:

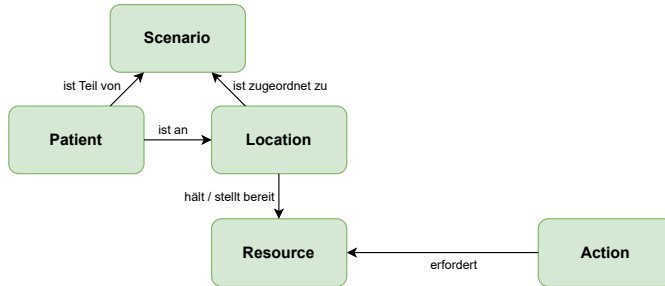
- Unabhängigkeit von ORM
- Konstante schnelle Zugriffszeiten
- Kapselung

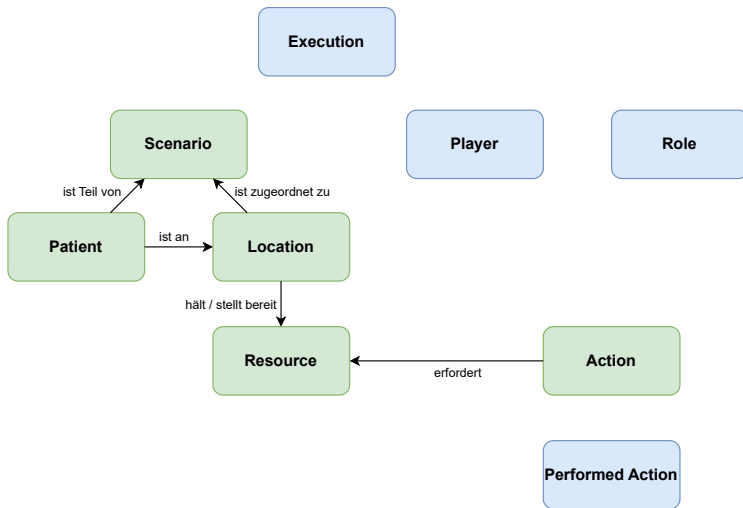
Nachteile:

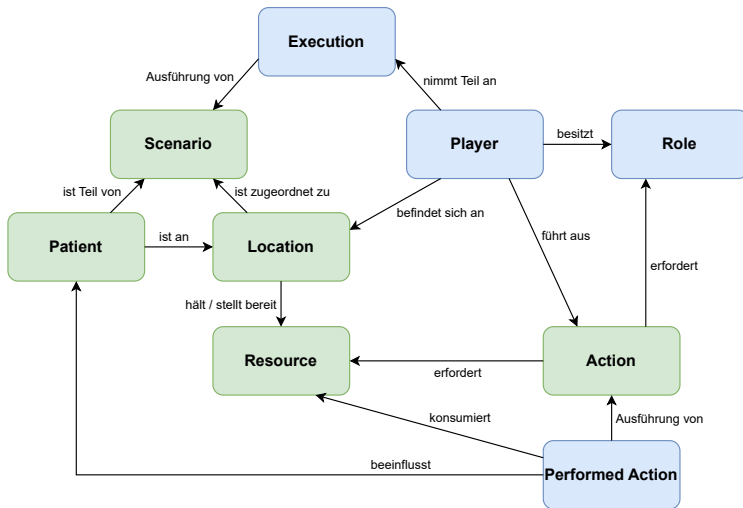
- Erhöhte Komplexität
- Synchronisation

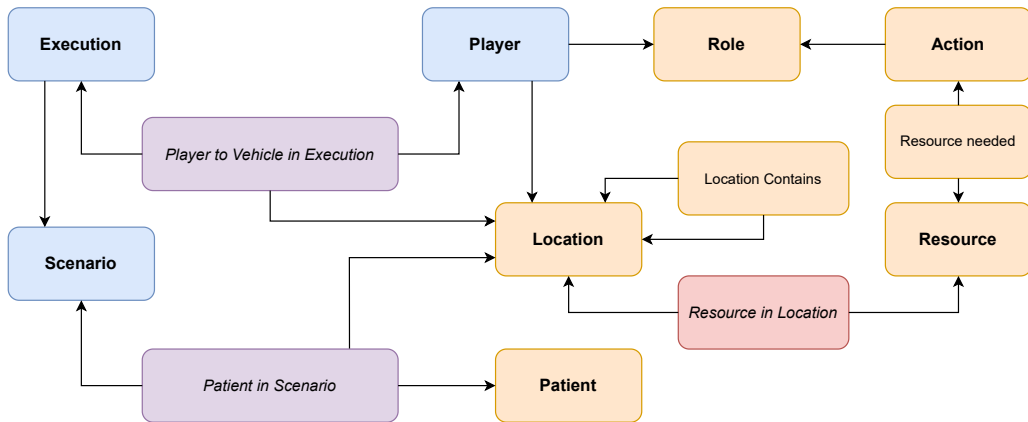








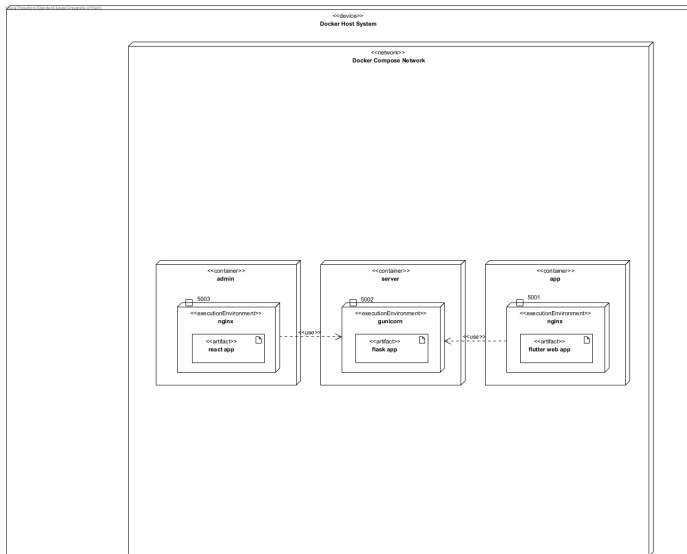




Deployment 1

Deployment in separaten Containern mit Docker Compose:

- server
 - API Server
 - Python + Flask
 - Deployment mit Gunicorn
- app
 - mobile App
 - Flutter + Dart
 - Deployment mit nginx
- admin
 - Admin Webinterface
 - REACT
 - Deployment mit nginx



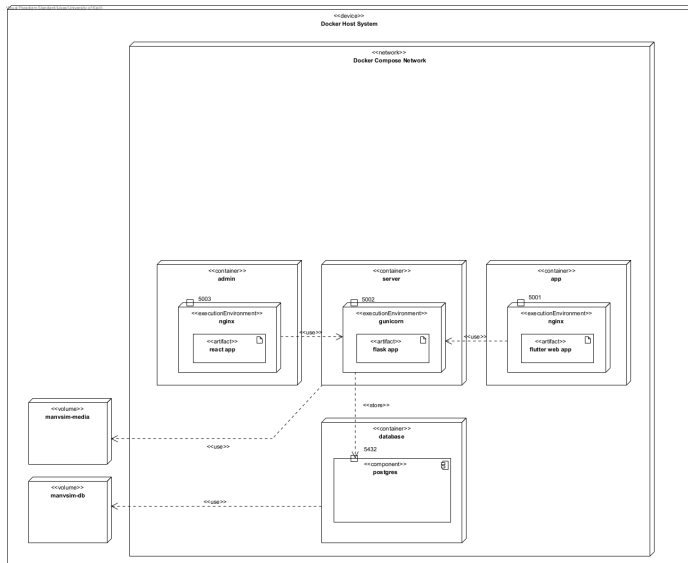
Deployment 2

Datenbank:

- PostgreSQL
- andere Datenbanken konfigurierbar
- Zugriff nur durch API Server

Persistenz über Docker Volumes:

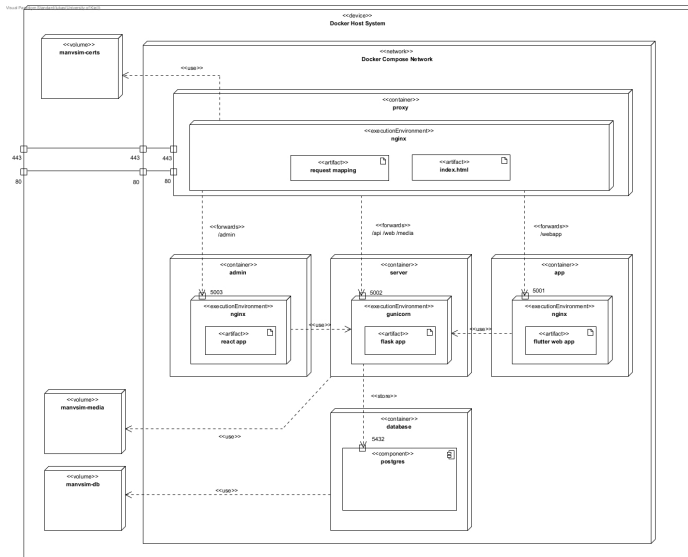
- manvsim-db
- manvsim-media



Deployment 3

Erreichbarkeit der Container über Reverse Proxy:

- nginx
- Deployment von Landing Page
- Path based Routing
- TLS
 - Let's Encrypt Zertifikate über Volume
 - innerhalb des Docker Netzwerks nur HTTP
- Port Freigabe/Mapping
 - 80:80 (http)
 - 443:443 (https)



Projektarbeit MANVSim

- Weekly Termin mit ergänzenden Einzelterminen
- Feature-Branch Implementierung
- Review Prozesse für gemeinsamen Wissensaustausch

Herausforderungen

- Unterschiedliches KnowHow vereinen
- Mehrere Entwicklungsschwerpunkte (Web, Python, Mobile)
- Stundenpläne und (Krank-)Ausfälle ändern Prioritäten

Future Work - Webanwendung

- Erweiterung/Bug-Fixing der Admin-Oberflächen
- Neustarten von abgebrochenen/abgestürzten Übungen

Future Work - Mobile App / GameAPI

- Ressourcen in Inventar Management integrieren
- Patienten-Koordinaten dynamisch 'sinnvoll' generieren

