

DIPLOMARBEIT

Great Powers - Eine Prozesssimulation soziopolitischer, konfliktbehafteter Ereignisse

Ausgeführt im Schuljahr 2022/23 von:

Jonas Nagelmaier 5AHIT
Lukas Wagner 5AHIT

Betreuer:

Dipl.-Ing. Paul Panhofer BSc.
Dipl.-Ing. Paul Panhofer BSc.

Krems, am 31. März 2023

EIDESSTATTLICHE ERKLÄRUNG

Ich erkläre an Eides statt, dass ich die vorliegende Diplomarbeit selbstständig und ohne fremde Hilfe verfasst, andere als die angegebenen Quellen und Hilfsmittel nicht benutzt und die den benutzten Quellen wörtlich und inhaltlich entnommenen Stellen als solche erkenntlich gemacht habe.

Krems, am 31. März 2023

Verfasser/Verfasserinnen:

Jonas Nagelmaier

Lukas Wagner

DIPLOMARBEIT

DOKUMENTATION

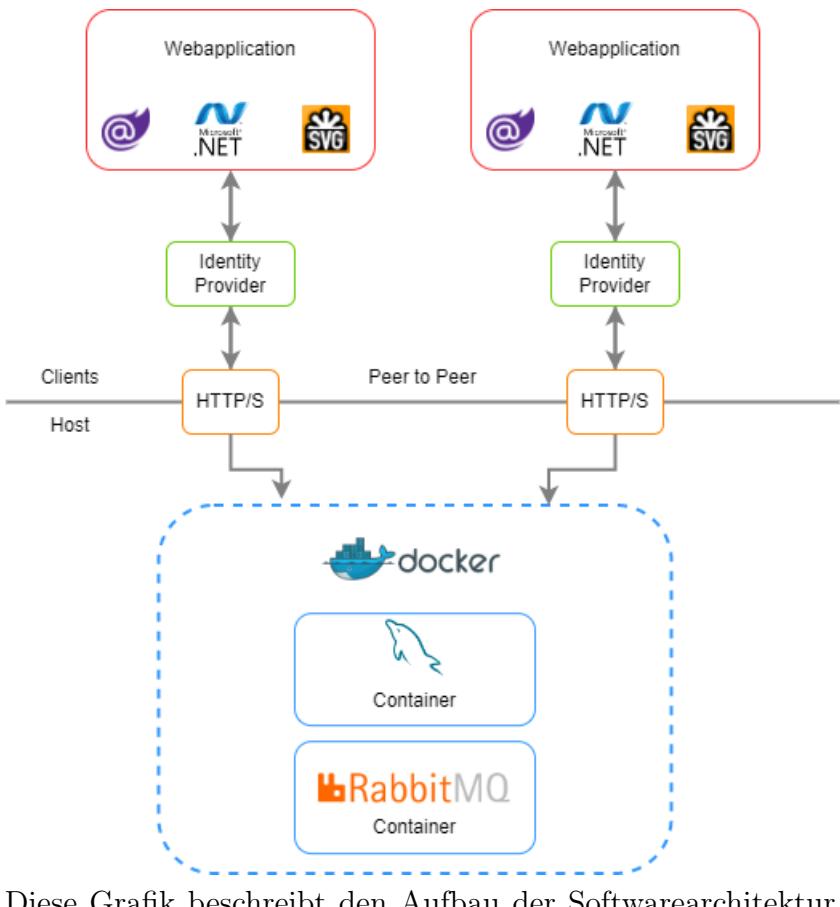
Namen der Verfasser	Jonas Nagelmaier Lukas Wagner
Jahrgang / Klasse Schuljahr	5AHIT 2022/23
Thema der Diplomarbeit	Great Powers - Eine Prozesssimulation soziopolitischer, konfliktbehafteter Ereignisse
Kooperationspartner	HTL Krems

Aufgabenstellung	<ul style="list-style-type: none">• Visualisierung historischer Prozessdaten• Abstrahierung historischer, soziopolitischer Prozesse zur Zeit des Zweiten Weltkrieges
------------------	---

Realisierung	<ul style="list-style-type: none">• Interaktive Clients zur Visualisierung und Adaptierung historischer Prozesse• Webschnittstelle basierend auf SVG Technologien• .NET Controller• Konzeptionierung einer relationalen Datenbankstruktur• Modelschicht zur Verwaltung von Entitäten• Domainschicht zur Kapselung des Datenbankzugriffs
--------------	--

Ergebnisse	Webapplikation zur Simulierung historischer Ereignisse mit Inhaltsschwerpunkt des Zweiten Weltkrieges.
------------	--

Typische Grafik, Foto etc.
(mit Erläuterung)



Diese Grafik beschreibt den Aufbau der Softwarearchitektur der Webapplikation. Sie zeigt die Verbindung zwischen den einzelnen Clientrechnern und erläutert die verwendete Peer to Peer Architektur.

Teilnahme an
Wettbewerben,
Auszeichnungen

Wettbewerb Jugend Innovativ 2022/23

Möglichkeiten der
Einsichtnahme in die
Arbeit

Bibliothek der HTL Krems

Approbation
(Datum / Unterschrift)

Prüfer/in

Abteilungsvorstand /
Direktor/in

DIPLOMA THESIS

Documentation

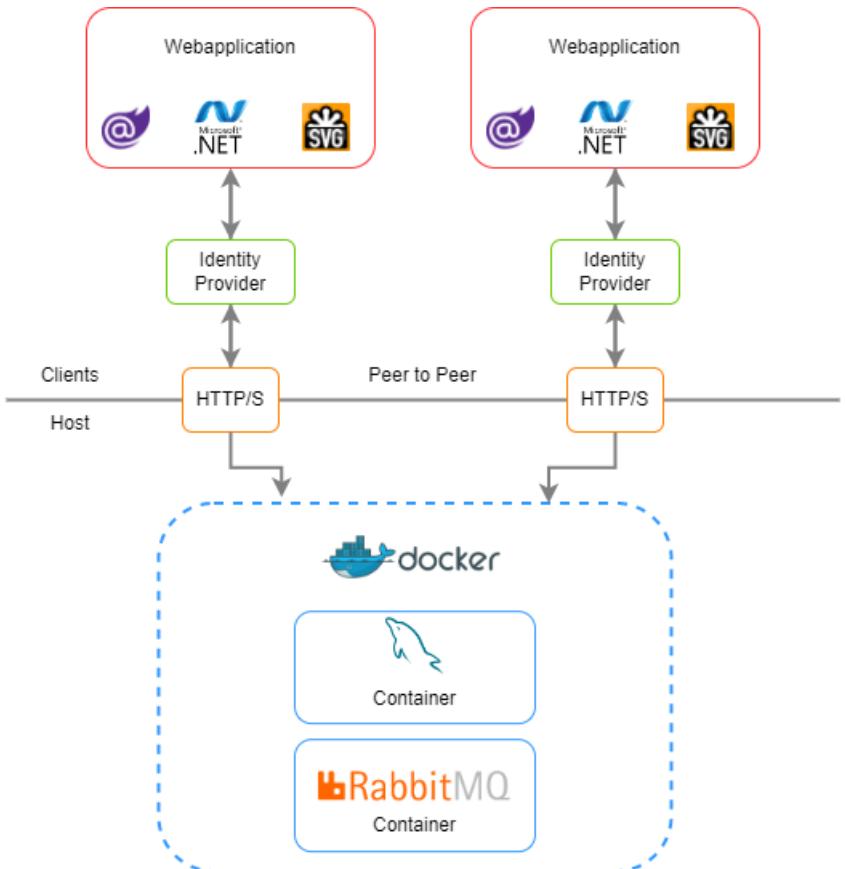
Authors	Jonas Nagelmaier Lukas Wagner
Form	5AHIT
Academic year	2022/23
Topic	Great Powers - A Process Simulation of Sociopolitical Events Involving Conflict
Co-operation partners	HTL Krems

Assignment of tasks	<ul style="list-style-type: none">• Visualization of historical process data• Abstraction of historical, socio-political processes at the time of the Second World War.
---------------------	--

Realization	<ul style="list-style-type: none">• Interactive clients for visualization and adaptation of historical processes• Web interface based on SVG technologies• .NET Controller• Conceptual design of a relational database structure• Model layer for entity management• Domain layer for encapsulation of database access
-------------	---

Results	Web application for simulating historical events with content focus on the Second World War.
---------	--

Illustrative graph, photo (incl. explanation)



This graphic describes the structure of the software architecture used by the web application. It shows the connection between the individual client computers and explains the peer-to-peer architecture that was used.

Participation in competitions
Awards

Competition Jugend Innovativ 2022/23

Accessibility of diploma thesis

Library of the HTL Krems

Approval
(Date / Sign)

Examiner

Head of Department /
College

Inhaltsverzeichnis

1. Präambel	11
1.1. Zusammenfassung	11
1.2. Abstract	11
1.3. Involvierte Personen	11
1.4. Danksagung	11
1.5. Gendererklärung	11
2. Einführung	12
2.1. Ausgangslage und Problemstellung	12
2.1.1. Komplexität der Materie	12
2.1.2. Zielsetzung	12
2.2. Forschungsfrage	12
2.3. Strukturierung der Arbeit	13
3. Prozesssimulation	14
3.1. Definitionen essenzieller Begriffe	14
3.1.1. Systemtheorie	14
3.1.2. Bildung von Modellen	14
3.1.3. Entwicklung von Modellen	14
3.1.4. Simulation	14
3.1.5. Historische Dynamiken	15
3.1.6. Prozess	15
3.1.7. Prozesssimulation	16
3.1.8. Prozesskomponente	16
3.2. Essenzielle Faktoren einer historischen Prozesssimulation	16
3.2.1. Geographische Faktoren	16
3.2.2. Geopolitische Faktoren	16
3.2.3. Wirtschaftliche Faktoren	16
3.2.4. Soziopolitische Faktoren	16
3.2.5. Strategische Faktoren	17
3.3. Simulationsmetriken	17
3.4. Bestehende Konzepte	17
3.4.1. Risiko	17
3.4.2. Hearts of Iron IV	19
4. Visualisierung historischer Ereignisse	23
4.1. Kartographische Visualisierung	23
4.2. Graph Visualisierung	24
4.3. Baum Visualisierung	25
5. Modellbildung	26
5.1. Kriterienkatalog zur Kategorisierung historischer Prozesse	26
5.1.1. Geographische Faktoren	26
5.1.2. Geopolitische Faktoren	26
5.1.3. Wirtschaftliche Faktoren	27
5.1.4. Soziopolitische Faktoren	27
5.1.5. Strategische Faktoren	27

5.2. Mathematische Modelle	28
5.2.1. Disproportionale Repräsentation von Simulationskomponenten	28
5.2.2. Deduktion der Eigenschaften von Simulationskomponenten in auf Zufall basierten Auseinandersetzungen	29
5.2.3. Erhebung wirtschaftlicher Eigenschaften anhand des GDP Modells . .	31
5.2.4. Ausarbeitung von situationsabhängigen Eigenschaften der Simulationskomponenten	31
6. Visualisierungsmethodiken des Prototypen	33
6.1. Implementierung der Kartographischen Visualisierung	33
6.1.1. Fraktionen	33
6.1.2. Aufteilung der Regionen	33
6.1.3. Darstellung der Truppen	34
6.1.4. Sonstige Visualisierungselemente	35
6.1.5. Simulationsinteraktionen	35
6.2. Konzeptionierung der Phaseninteraktion	36
6.3. Bereitstellung der aktuellen Informationen	37
6.4. Visualisierung des Schlachtfeldes	39
7. Diskussion eines vereinfachten Modells	40
7.1. Erhebung eines Referenzmodells	40
7.1.1. Bewertung geographischer Faktoren	40
7.1.2. Bewertung geopolitischer Faktoren	40
7.1.3. Bewertung wirtschaftlicher Faktoren	40
7.1.4. Bewertung soziopolitischer Faktoren	41
7.1.5. Bewertung strategischer Faktoren	41
7.2. Bewertung des Great Powers Modell	41
7.2.1. Bewertung geographischer Faktoren	41
7.2.2. Bewertung geopolitischer Faktoren	42
7.2.3. Bewertung wirtschaftlicher Faktoren	42
7.2.4. Bewertung soziopolitischer Faktoren	42
7.2.5. Bewertung strategischer Faktoren	42
7.3. Kontrastierung mit erhobenem Referenzmodell	43
7.3.1. Anforderungen	43
7.3.2. Modellbeschreibung des Prototypen	43
8. Bewertung von Visualisierungsmethoden zur Darstellung historischer Prozesse	44
8.1. Diskussion und Bewertung der Kartographische Darstellung	44
8.1.1. Einsatzgebiet des Kartographischen Modells im historischen Kontext	45
8.2. Diskussion und Bewertung der Graph Darstellung	46
8.2.1. Einsatzgebiet des Graph Modells	47
8.3. Diskussion und Bewertung der Baum Darstellung	48
8.3.1. Einsatzgebiet des Baum Modells	49
8.4. Konzept Entscheidung	50
8.4.1. Anforderungen	50
8.4.2. Entscheidungsfindung	50

9. Systemarchitektur	51
9.1. Komponentenübersicht	51
9.1.1. Blazor Server App	52
9.1.2. Identity Provider	52
9.1.3. RabbitMq Message Broker	52
9.1.4. MySql Datenbank	52
9.2. Kommunikation	53
9.3. Technologien	54
9.3.1. .NET Core	54
9.3.2. Blazor Server App	54
9.3.3. MySql	55
9.3.4. RabbitMq	56
9.3.5. SVG	56
9.3.6. Panzoom	57
9.3.7. Docker	57
10. Komponentenbeschreibung Great-Powers	58
10.1. Komponentenbeschreibung Program	58
10.1.1. Containerisierung	58
10.1.2. WebApplication Builder	59
10.1.3. ASP.NET Core Identity Framework	59
10.1.4. MudBlazor Service	60
10.1.5. Dependency Injection	60
10.2. Komponentenbeschreibung View	61
10.2.1. Menü Subkomponent	61
10.2.2. Simulations Subkomponent	62
10.3. Komponentenbeschreibung Model	63
10.3.1. Datenbankstruktur	63
10.3.2. Übersicht der Entitäten	63
10.3.3. DbContext	65
10.4. Komponentenbeschreibung Domain	65
10.4.1. ARepository	66
10.4.2. ACreatableRepository	67
10.4.3. SessionInfo-Repository	68
10.5. Komponentenbeschreibung Eventbus	68
10.5.1. Event Publisher	69
10.5.2. Event Subscriber	70
10.5.3. Event Records- und Handler	72
11. Zusammenfassung und Projektaussichten	74
11.1. Beantwortung der Forschungsfrage	74
11.1.1. Vorbereitung	74
11.1.2. Umsetzung	74
11.1.3. Auslieferung	74
11.1.4. Schlussfolgerung	74
11.2. Ausbaupotential	74

12. Installation	75
12.1. Systemanforderungen	75
12.2. Dateiträgerbeschreibung	75
12.3. Installation & Ausführung	75
12.3.1. Installation	75
12.3.2. Ausführung	75
I. Literaturverzeichnis	76
II. Abbildungsverzeichnis	78
III. Tabellenverzeichnis	79
IV. Quellcodeverzeichnis	80
V. Abkürzungsverzeichnis	81
A. Anhang	82
A.1. Grafiken	82
A.2. Kapitelverzeichnis	84
A.3. Besprechungsprotokolle	85
A.4. Pflichtenheft	91
A.5. Arbeitspakete	102
A.6. Projekttagebücher	105
A.6.1. Projekttagebuch Jonas Nagelmaier	105
A.6.2. Projekttagebuch Lukas Wagner	107
A.7. Dateiträgerbeschreibung	109

	<p style="text-align: center;">HÖHERE TECHNISCHE BUNDES - LEHRANSTALT Krems</p>
Abteilung:	Informationstechnologie

1. Präambel

1.1. Zusammenfassung

Die Zielsetzung des Projektes bestand in der Konzeptionierung und Implementierung einer Prozesssimulation des Zweiten Weltkrieges.

Dafür wurde ein Prototyp mit folgenden Komponenten implementiert:

- ASP.NET Core Blazor Webapplikation
- MySql Datenbank
- RabbitMq Message Broker

1.2. Abstract

The objective of the project was to design and implement a process simulation of World War II. For this purpose, a prototype with the following components was implemented:

- ASP.NET Core Blazor Webapplication
- MySql Database
- RabbitMq Message Broker

1.3. Involvierte Personen

Das Projektteam besteht aus dem Projektleiter Jonas Nagelmaier und Kollegen Lukas Wagner. Betreuer und Auftraggeber ist Dipl.-Ing. Paul Panhofer BSc.

1.4. Danksagung

Wir möchten uns besonders bei unserem Betreuer Dipl.-Ing. Paul Panhofer BSc. bedanken. Dieser hat uns während der gesamten Umsetzung unserer Arbeit mit seiner Expertise unterstützt und uns wertvolle Ratschläge gegeben.

Des Weiteren möchten wir uns besonders bei unseren Familien und unseren Freunden bedanken, welche uns in vielerlei Hinsicht bei unserem Projekt unterstützt haben. Wir fanden bei ihnen immer ein offenes Ohr, welches uns emotional unterstützte, indem sie für uns da waren und uns zuhörten, wenn wir uns über unsere Herausforderungen und Sorgen im Zusammenhang mit unserem Projekt unterhalten haben. Ihre Unterstützung hat geholfen, die emotionalen Belastungen, die mit dem Projekt einhergingen, besser zu bewältigen.

1.5. Gendererklärung

In dieser Arbeit wird aus Gründen der besseren Lesbarkeit das generische Maskulinum verwendet. Weibliche und anderweitige Geschlechteridentitäten werden dabei ausdrücklich mitgemeint, soweit es für die Aussage erforderlich ist.

2. Einführung

2.1. Ausgangslage und Problemstellung

2.1.1. Komplexität der Materie

Die Bewertung komplexer soziopolitischer Prozesse im Kontext historischer Studien bleibt auch heute noch das Privileg einer profilierten Expertise.

Durch den Einsatz einer Prozesssimulation soll ein fundiertes Verständnis komplexer soziopolitischer Prozesse erlangt werden.

2.1.2. Zielsetzung

Ziel der Arbeit ist die Einführung gewillter und interessierter Schüler in das komplexe Feld der historischen soziopolitischen Prozesse des Zweiten Weltkrieges. Parallel wird ein Kriterienkatalog zur Komplexitätsreduktion für Prozesssimulationen diskutiert.

2.2. Forschungsfrage

Die Simulation historischer Prozesse beschreibt ein aufwendiges Unterfangen. Zur Vereinfachung solcher Systeme soll geprüft werden, welche Aspekte der Simulation das System signifikant Beeinflussen.

Folgende Aspekte sind dabei von vordergründigen Interesse:

- Geopolitische Faktoren
- Strategische Muster
- Geographische Strukturen
- Wirtschaftslagen
- Bündnisse diverser Länder

ermittelt werden.

Daraus resultiert die Forschungsfrage:

Wie stark können historisch geprägte soziopolitische Prozesse zur Zeit des Zweiten Weltkrieges vereinfacht werden, ohne ihren Simulationscharakter zu verlieren?

2.3. Strukturierung der Arbeit

Die Arbeit ist in 3 Hauptabschnitte gegliedert.

- Recherche
- Empirischer Teil
- Bewertung

Zu Beginn der Arbeit werden bestehenden Konzepte zur Abbildung historischer Ereignisse aufgezeigt. Im zweiten Punkt liegt der Fokus auf der Konzeptionierung des Prototypen, welcher zum beweisen der These herangezogen wird. Schlussendlich werden diverse Modelle und Visualisierungskonzepte verglichen und auf deren Vor- und Nachteile eingegangen.

3. Prozesssimulation

3.1. Definitionen essenzieller Begriffe

3.1.1. Systemtheorie

Unter der Systemtheorie versteht man die teilhafte Beschreibung der Realität. Hierbei werden Grundlagen und Prinzipien der einzelnen Systeme dargestellt. So können beispielsweise Nationen, Gesellschaften, Tiere, Pflanzen, Ökosysteme oder ganze Sternenkonstellationen anhand der Systemtheorie abgebildet werden.

3.1.2. Bildung von Modellen

Modelle geben Teile der Realität abstrahiert wieder. Durch eine anfängliche Frage- und Problemstellung wird der Zweck des Modells definiert. Grundsätzlich können Modelle in dem Maß ihrer Abstraktion von einer realitätsgetreuen Wiedergabe bis hin zu einer sehr stark abstrahierten Abbildung variieren. Es wird im wissenschaftlichen Bereich zwischen zwei Arten von Modellen unterschieden

- Gegenständliche Modelle
- Formale Modelle

Gegenständliche Modelle beziehen sich auf Gegebenheiten der realen Welt, wie etwa Objekte, Ereignisse sowie auch das Sachverhalten zwischen diesen. Formale Modelle bestehen jedoch nur aus mathematischen Formeln und Gleichungen. Ebenfalls unterscheidet man zwischen statischen und dynamischen Modellen. Statische Größen und deren Beziehungen zueinander sind für statische Modelle von großer Bedeutung. Im Gegensatz dazu bilden dynamische Modelle die Dynamiken eines Objekts ab.

3.1.3. Entwicklung von Modellen

Die Modellentwicklung bezieht sich auf das Entwerfen von Modellen für spezifische Probleme, bei dem das reale Problem in ein mathematisch formuliertes Modell übertragen wird. Um ein Modell zu entwickeln, müssen vier Schritte durchgeführt werden. Zunächst wird eine Fragestellung oder ein Problem definiert, das gelöst werden soll. Basierend auf dieser Fragestellung wird ein Modellzweck definiert, der speziell auf die Problemstellung zugeschnitten ist. Anschließend werden die Modellart und Modellierungstiefe evaluiert und ausgewählt. Die Modellierungstiefe gibt an, wie detailliert das Modell sein sollte, um den Ausschnitt der Realität genau zu repräsentieren. Während der Modellbildung wird das Modell erstellt und die Modellierung ausgeführt. Im letzten Schritt der Modellvalidierung wird überprüft, ob das entwickelte Modell das gesteckte Ziel erfüllt.

3.1.4. Simulation

Simulationen verwenden Modelle, um einen Ablauf oder Prozess aus der Wirklichkeit über einen bestimmten Zeitraum nachzubilden. Durch Simulationen können verschiedene Szenarien durchlaufen und getestet werden, was in einer Vielzahl von Anwendungsbereichen

nützlich ist. Zum Beispiel können Simulationen zur Optimierung von Prozessen, zum Testen von Theorien, zur Verbesserung der Leistung und sogar für Unterhaltungszwecke eingesetzt werden.

3.1.4.1. Vorteile von Simulationen

Simulationsmodelle haben verschiedene Vorteile, wie zum Beispiel ein geringes finanzielles Risiko, die Möglichkeit, verschiedene Theorien unter denselben Umständen zu testen sowie die Analyse von langfristigen Auswirkungen und Folgen. Darüber hinaus ermöglichen Simulationsmodelle, komplexe Sachverhalte auf eine einfache Art und Weise darzustellen, was zu einem besseren Verständnis beitragen kann.

3.1.4.2. Nachteile von Simulationen

Obwohl Simulationen verschiedene Vorteile bieten, haben sie auch ihre Einschränkungen. Es ist kaum möglich, mit einer Simulation alle Faktoren der Realität realitätsgerecht abzubilden. Es sei jedoch darauf hingewiesen, dass nicht jeder Anwendungsbereich eine genaue Realitätsabbildung erfordert. In Branchen wie der Unterhaltung und Bildung werden nur Teilespekte simuliert oder dargestellt, da das Ziel nicht darin besteht, die Realität exakt wiederzugeben.

3.1.5. Historische Dynamiken

Historische Dynamiken, die auch oft als historische Simulation bezeichnet werden, beinhalten vor allem die historische Modellierung und umfassen viele Teilbereiche wie z.B. die Populationsmodellierung. In den letzten zwei Jahrzehnten haben historische Simulationen durch die Digitalisierung stark an Popularität gewonnen, insbesondere für Unterhaltungs- und Bildungszwecke. Bekannte Beispiele dafür sind die Prozesssimulationen von EUROPA UNIVERSALIS [1] und Civilization [2], welche eine historische Periode simulieren und dem Benutzer ermöglichen, Einfluss auf deren Verlauf zu nehmen. Historische Dynamiken finden auch Anwendung bei komplexen Modellen, mit denen Prognosen für soziale oder ökonomische Entwicklungen erstellt werden können. Sie können auch dazu beitragen, abstrakte Abläufe wie sozio-politische Prozesse im historischen Kontext zu veranschaulichen.

3.1.6. Prozess

Ein Prozess in der Prozesssimulation ist eine Nachbildung eines realen Ereignisses und dient dem analysieren der unterschiedlichen Einflüsse, welche auf seinen Verlauf wirken. Die Simulation ermöglicht es, die Auswirkungen von Änderungen im Prozessdesign oder in den Bedingungen, unter denen der Prozess durchgeführt wird, zu untersuchen und zu erforschen.

Ein historischer Prozess bezieht sich auf einen tatsächlich stattgefundenen Prozess in der Vergangenheit. Dies kann ein Ereignis oder eine Abfolge von Ereignissen sein, die historische Bedeutung haben. Historische Prozesse können auf verschiedene Arten analysiert werden. Die Untersuchung historischer Prozesse kann dazu beitragen, historische Zusammenhänge besser zu verstehen und wichtige Lehren für die Gegenwart und die Zukunft zu ziehen.

3.1.7. Prozesssimulation

Diese Prozesse tragen alle dazu bei, die historischen Gegebenheiten unter Berücksichtigung der ausschlaggebenden Metriken in der Form einer Prozesssimulation zu abstrahieren.

3.1.8. Prozesskomponente

Prozesskomponenten sind die einzelnen durch ihre spezifischen Eigenschaften unterscheidbaren Elemente, aus welchen die Prozesssimulation aufgebaut ist. Dazu gehören beispielsweise Truppen, Regionen, Industriesektoren oder auch deren übergeordneten Nationen selbst.

3.2. Essenzielle Faktoren einer historischen Prozesssimulation

3.2.1. Geographische Faktoren

Geographische Faktoren beschreiben physische Merkmale und Eigenschaften, welche Auswirkungen auf menschliche Aktivitäten haben können. Hierzu zählen Topographische Verhältnisse, Klima, Agrarproduktion und Bodenkulturen. Diese Faktoren spielen bei der Konzeptionierung von geographischen Plänen eine große Rolle.

3.2.2. Geopolitische Faktoren

Geopolitische Faktoren beziehen sich auf die politischen, wirtschaftlichen, militärischen und kulturellen Faktoren, welche Beziehungen und Interaktionen zwischen Staaten und Regionen beeinflussen. Hierzu zählen politische Ansichten und Aktionen sowie auch die Zugehörigkeiten von Nationen zu Bündnissen und Organisationen.

3.2.3. Wirtschaftliche Faktoren

Wirtschaftliche Faktoren sind Faktoren, welchen die Wirtschaftstätigkeit eines Landes oder einer Region beeinflussen können. Eine wichtige wirtschaftliche Kennzahl bildet das Bruttoinlandsprodukt, welches als Grundlage für den wirtschaftlichen Vergleich diverser Nationen genutzt werden kann.

3.2.4. Soziopolitische Faktoren

Soziopolitische Faktoren sind Faktoren, welche die Interaktionen zwischen Gesellschaft und Politik beschreiben. Ein weiterer Bereich der Soziopolitik wird durch ethnische und kulturelle Werte geprägt.

3.2.5. Strategische Faktoren

Unter strategischen Faktoren versteht man den Vorgang einer Entscheidungsfindung, welche zu einem gewünschten Ausgang führt. Im historischen Kontext kann dies eine Militäroperation, Sanktionen oder die Umstrukturierung von Infrastruktur im eigenen Land sein.

3.3. Simulationsmetriken

Zur Identifikation der prozesssignifikanten Metriken wurden unterschiedliche Konzepte miteinander verglichen. Durch diesen Vergleich wurden Faktoren wie beispielsweise die Vielfalt der eingesetzten Technologien als nicht essenziell eingestuft. Gleichzeitig wurden andere Faktoren als unverzichtbar angesehen wurden, darunter unter anderem die geographische Aufteilung einzelner Regionen.

3.4. Bestehende Konzepte

Folgende Simulationsmodelle sollen miteinander verglichen werden:

- Risiko [3], welches als Beispiel einer stark abstrahierten Prozesssimulation für konfliktbehaftete Prozesse diente
- Hearts of Iron IV [4], welches als Beispiel einer komplexen Prozesssimulation für konfliktbehaftete Prozesse diente

3.4.1. Risiko

Bei Risiko [3] werden historische, konfliktbehaftete Ereignisse auf stark abstrahierte und eher simple Art und Weise sowohl in Aspekten der Visualisierung als auch der Modellbildung dargestellt. Es bietet mehrere verschiedene Simulationsszenarien, darunter auch ein Szenario zur Zeit des Zweiten Weltkrieges, was zu einer vereinfachten Repräsentation von Simulationskomponenten führt. So gibt es für Truppen nur eine einzige generelle Repräsentation und auch die Aufteilung von Regionen, welche nur Landmassen darstellen, ist simpel gehalten und vernachlässigt jegliche aquatische Simulationsaspekte. Zu solchen aquatischen Aspekten zählen die vielen verschiedenen Seeeinheiten, wie etwa Flugzeugträger oder Kreuzer. Auch geopolitische Aspekte werden abstrahiert und repräsentieren so nur die 6 Kontinente. Inwiefern das Risiko Modell die bereits aufgestellten essenziellen Faktoren, siehe Kapitel 3.2, umsetzt, wird nun genauer beschrieben.



Abbildung 3.1.: Risiko Prozesssimulation

3.4.1.1. Diskussion des Simulationsmodells

- Geographische Faktoren

Die Wiedergabe von geographischen Gegebenheiten geschieht in Risiko äußerst abstrahiert, da hier die gesamte Karte auf lediglich 42 Gebiete aufgeteilt wird. Seeregionen werden hierbei nicht berücksichtigt, was auf spätere strategische Faktoren einen maßgebenden Einfluss hat. Um Truppen zwischen Kontinenten zu bewegen, gibt es einige fixierte Punkte, an denen diese von einer Landmasse zur anderen kommen.

- Geopolitische Faktoren

Die in Risiko vorhandenen geopolitischen Faktoren umfassen die Definition der unterschiedlichen Nationen sowie die Verteilung der geographischen Regionen an diese. Aufgeteilt sind die Regionen auf 6 Nationen, welche jeweils einen Kontinent repräsentieren: Nordamerika, Südamerika, Afrika, Europa, Asien und Australien.

- Soziopolitische Faktoren

In dem Risiko Modell werden soziopolitische Faktoren nicht weiter berücksichtigt. Das bedeutet, dass Faktoren wie die Religion, Ethnie oder Kultur von Bevölkerungsgruppen und der Einfluss dieser auf die Entscheidungsfindung einer Nation nicht abgebildet sind.

- Wirtschaftliche Faktoren

Wirtschaftliche Faktoren werden in Risiko nur durch die Anzahl der Regionen, welche im Besitz einer Nation sind, dargestellt. So hat nun eine Nation mit 10 Regionen auch

eine Wirtschaftsstärke von 10. Für jeden Punkt an Wirtschaftskraft einer Nation kann pro Runde dann eine neue Einheit rekrutiert werden.

- Strategische Faktoren

Die Abbildung von strategischen Faktoren erfolgt im Risiko Modell ebenfalls stark abstrahiert. Hier wird zur Repräsentation von Einheiten aller verschiedenen Typen eine einzige Komponente verwendet. Diese können sich dann pro Runde um ein Gebiet bewegen, wobei es bei einer Bewegung auf ein feindliches Gebiet zu einem Gefecht kommt. Gefechte werden geführt, indem Angreifer und Verteidiger wiederholt mehrere Würfel werfen und die Seite, wessen höchste gewürfelte Zahl geringer ist als die der anderen, eine Einheit verliert. Hat eine Seite keine Einheiten mehr, hat diese verloren.

3.4.2. Hearts of Iron IV



Abbildung 3.2.: Hearts of Iron Prozesssimulation

Hearts of Iron IV [4] ist der vierte Eintrag in eine Reihe von aufeinander bauenden Prozesssimulationen historischer, konfliktbehafteter Ereignisse, welches Ereignisse zur Zeit des Zweiten Weltkrieges simuliert. Hierbei handelt es sich um eine äußerst komplexe Repräsentation von historischen Ereignissen, welche eine umfangreiche Wiedergabe von einer Vielzahl an Aspekten darbietet, darunter etwa das technologische Fortschreiten von Nationen, komplexe politische Interaktionen zwischen Nationen, verwalten von Versorgungslinien und Logistik oder auch die Steuerung von politischen und marktdynamischen Ausrichtungen einer Nation. Auf die Umsetzung der vorab definierten essenziellen Faktoren - siehe Kapitel 3.2 - im Hol4 Modell wird nun genauer eingegangen.

3.4.2.1. Diskussion des Simulationsmodells

- Geographische Faktoren

Wie in der folgenden Grafik ersichtlich, wird in HoI4 ebenfalls auf die Einteilung der Landschaft in Regionen gesetzt, wobei die Einteilung hier jedoch um einiges granulärer erfolgt als im Great-Powers Modell. Weiteres werden im HoI4 Modell auch eine Vielzahl an bedeutsamen Städten repräsentiert, welche hier ihre eigenen Regionen sind. Zusätzlich wird zwischen den unterschiedlichen Gegebenheiten von Regionen wie z.B. Wäldern, Gebirgen oder eben Städten unterschieden, was eine Auswirkung auf die späteren Faktoren besitzt.



Abbildung 3.3.: Geographische Darstellung in HoI4

- Geopolitische Faktoren

Ebenfalls aus der vorherigen Grafik ersichtlich ist die Aufteilung dieser geographischen Regionen in den Besitz verschiedener Nationen, wobei hier wieder auf eine umfangreiche Menge an unterschiedlichen Nationen gesetzt wird. Diese haben alle ein großes Maß an Autonomie, was zusammen mit einem komplexen System zur Abbildung politischer Handlungen zwischen Nationen die nicht konfliktbasierten Interaktionen unter Nationen äußerst detailgetreu wiedergeben. Außerdem haben alle Nationen ihren eigenen sogenannten Fokusbaum, welcher die vielen verschiedenen Entscheidungen und Zielsetzungen der Regierung einer Nation repräsentiert. Der Umfang dessen soll in folgender Grafik illustriert werden

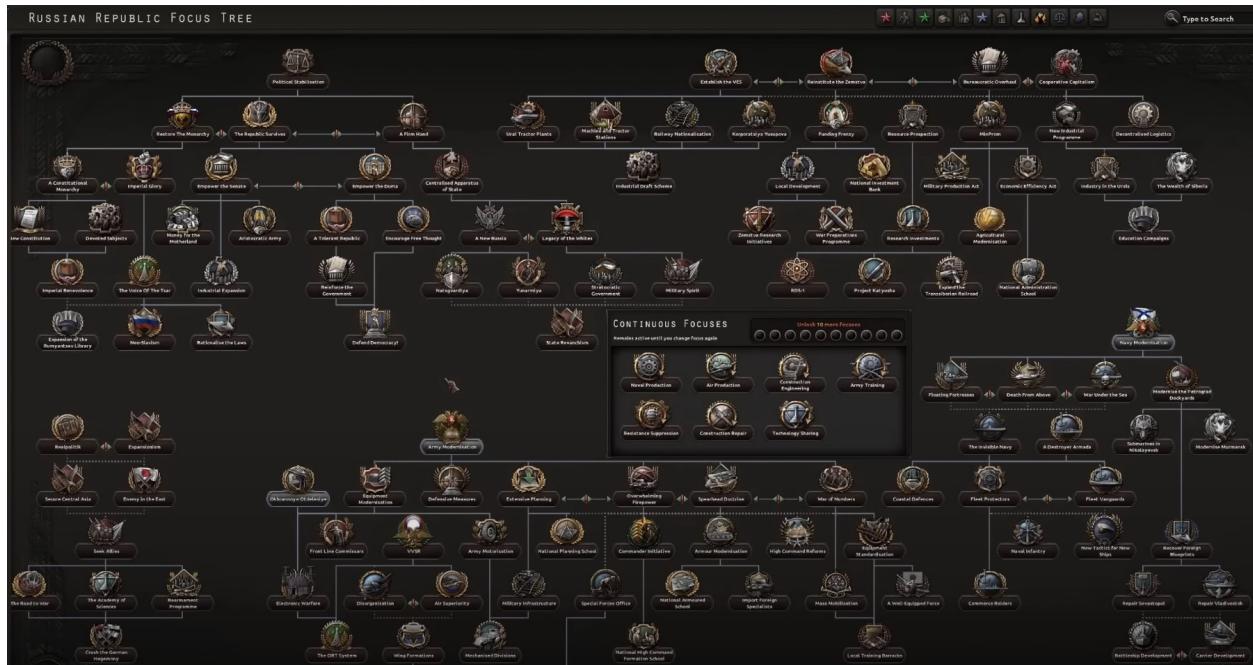


Abbildung 3.4.: Fokusbaum in HoI4

- Wirtschaftliche Faktoren

In HoI4 existieren einige äußerst umfangreiche Systeme zur Abbildung der wirtschaftlichen Entwicklung sowie dem Ausbau der Infrastruktur in den unterschiedlichen Gebieten einer Nation. Beispielsweise können in Gebieten Industriekomplexe errichtet werden, um sowohl militärische Ausrüstung als auch Ressourcen und Produkte, welche entweder zur weiteren Verarbeitung oder zum Handel mit anderen Nationen verwendet werden, herzustellen. Der Bau solcher Industriekomplexe wird in folgender Grafik dargestellt



Abbildung 3.5.: Wirtschaftlicher Ausbau in HoI4

Des Weiteren kann für den schnellen und einfachen Transport von Einheiten und deren Versorgung ein Schienennetzwerk durch Regionen konstruiert werden, was die historischen Gegebenheiten sehr akkurat widerspiegelt.

Letztlich gilt noch zu erwähnen, dass der Bau von Industriekomplexen und Infrastruktur selbstverständlich Ressourcen benötigt, was so einen Kreislauf von der Ressourcenschöpfung aus besetzten Regionen über die Aneignung neuer Regionen durch Gefechte und schließlich die Investition von Ressourcen in den Ausbau gewonnener Regionen darstellt.

- Soziopolitische Faktoren

HoI4 bildet die historischen Gegebenheiten von verschiedenen Kulturen, Religionen, Ethnien und Ideologien und den Interaktionen zwischen diesen ab, indem die Bevölkerung jeder Region eine gewisse Akzeptanz oder das Fehlen dieser gegenüber der ideologischen und politischen Einstellungen der ihre Region besitzenden Nation aufweist. Diese Akzeptanz kann durch weitere Aktionen der Nation sowohl steigen als auch sinken, wobei eine hohe Akzeptanz für die effektive Verwendung von Gebieten nötig ist und eine zu niedrige Akzeptanz zu Aufständen oder sogar Bürgerkriegen führen kann.

- Strategische Faktoren

Die strategischen Faktoren, welche in konfliktbasierten Prozessen maßgebend sind, werden in HoI4 in großem Detail wiedergegeben. So werden Faktoren wie das Gelände, die Versorgung von Einheiten, die spezifischen Stärken unterschiedlicher Einheitentypen, die Organisation und Ausrüstung von Truppen sowie auch die Planung und Anwendung von gewissen strategischen Mustern in Auseinandersetzungen abgebildet. Genaueres hierzu ist aus der folgenden Grafik abzulesen

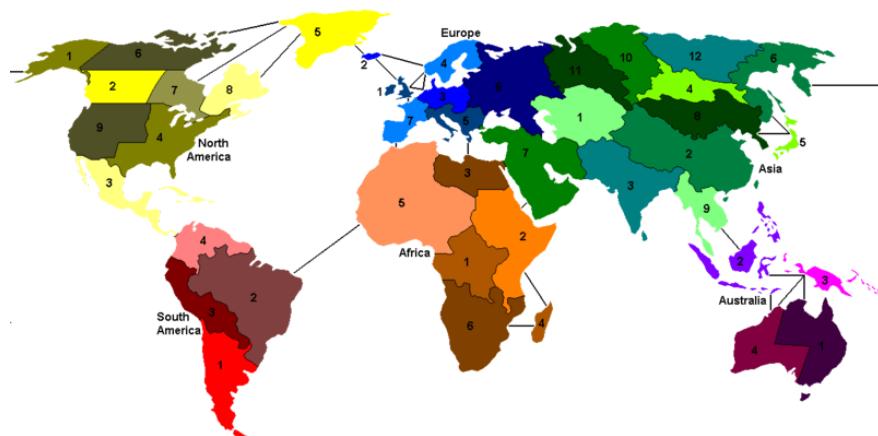


Abbildung 3.6.: Abbildung von Konflikten in HoI4

Ebenfalls ist durch die granuläre Abbildung von Gebieten das Nachstellen gewisser taktischer Vorgänge möglich. So kann man beispielsweise feindliche Truppen umkreisen, um diese von ihrer Versorgungslinie abzuschneiden, was zu der schnellen Entscheidung eines Gefechts führen kann.

4. Visualisierung historischer Ereignisse

4.1. Kartographische Visualisierung



Eine kartographische Visualisierung ist eine graphische Darstellung von Informationen auf einer Karte. Sie kann verschiedene Elemente enthalten, wie beispielsweise Muster, Symbole oder Farben, um verschiedene Daten darzustellen. Eine einfache Form der kartographischen Visualisierung wäre beispielsweise eine Karte, welche Länder oder ganze Regionen zeigt.

Die Verwendung unterschiedlicher Farben soll dabei zusätzliche Informationen liefern. Zum Beispiel können die Länder nach ihrer geographischen Lage eingefärbt werden, um Kontinente oder Regionen zu unterscheiden. Alternativ können auch spezifische Daten wie Bevölkerungszahl oder Wirtschaftsleistung mit Hilfe von Farben oder Schattierungen dargestellt werden.

Eine historische Karte kann auch mit Symbolen und Beschriftungen versehen werden, um wichtige geographische Merkmale hervorzuheben oder zusätzliche Informationen darzustellen. Zum Beispiel können Städte, Hauptstädte oder wichtige Häfen mit Symbolen oder Namen gekennzeichnet werden, um ihre Bedeutung auf der Karte zu verdeutlichen.

Die historische Karte ist eine einfache Form der kartographischen Visualisierung, die trotzdem viele nützliche Informationen vermitteln kann. Diese Art von Karte wird oft verwendet, um politische Zusammenhänge und Beziehungen zwischen verschiedenen Ländern darzustellen oder um einen schnellen Überblick über die geographische Verteilung von Bevölkerung und Wirtschaftsleistung zu geben.

4.2. Graph Visualisierung

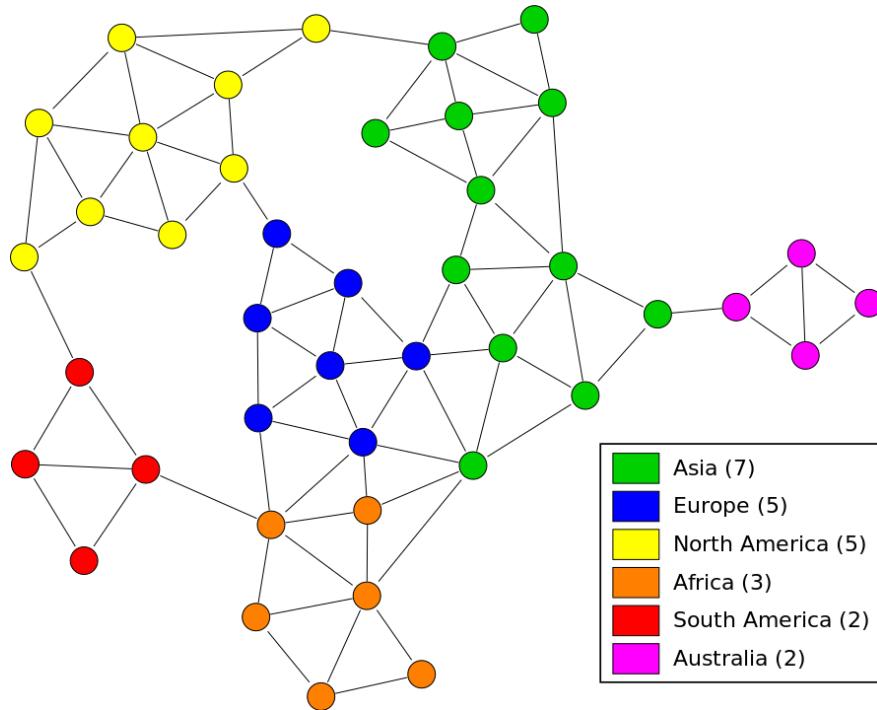


Abbildung 4.2.: Darstellung eines historischen Ereignisses in Form eines Graphen [6]

Die Visualisierung einer Karte kann auch in Form eines Graphen dargestellt werden. Hierbei bilden die Regionen die Knoten. Die Kanten stellen die Verbindungen zwischen den Regionen dar. Unter diesem Modell ist klar ersichtlich, welche Regionen miteinander verknüpft sind.

Durch die Darstellung einer Karte als Graphen können auch weitere räumliche Informationen wie Entfernungen, Verbindungen oder weitere Parameter in die Analyse einbezogen werden. Dies kann besonders nützlich sein, um die effizienteste Route zwischen verschiedenen Orten zu finden oder um komplexere Analysen über einen Abschnitt der Karte zu erstellen.

Zusätzlich können komplexe Netzwerke und Beziehungen zwischen verschiedenen Regionen dargestellt werden. Zum Beispiel können Verbindungen zwischen verschiedenen Städten, Flughäfen oder Häfen in einem bestimmten Gebiet in einem Netzwerkgraphen dargestellt werden, um die wichtigsten Verbindungen zu identifizieren.

Durch die Verwendung von Algorithmen zur Analyse von Graphen können auch Vorhersagen über mögliche Entwicklungen in einem Gebiet gemacht werden.

4.3. Baum Visualisierung

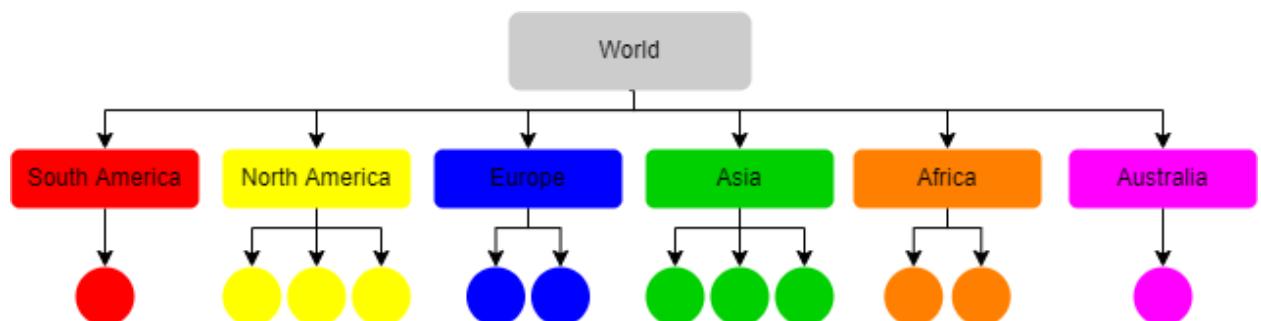


Abbildung 4.3.: Darstellung eines historischen Ereignisses in Form eines Baumdiagramms

Eine Baumvisualisierung kann verwendet werden, um hierarchische geographische Daten darzustellen. Hierbei wird die höchste Ebene der Baumstruktur durch eine breite Region wie ein Land, ein Kontinent oder die gesamte Welt repräsentiert. Eine Ebene darunter wird dann das nächstkleinere Element dargestellt.

Jeder Knoten in der Baumstruktur kann mit weiteren geographischen Informationen wie Wirtschaftsstärke, historische Merkmale wie Truppenkapazitäten oder geographischen Merkmalen wie Grenzen oder Lage verknüpft werden. Darüber hinaus ist es möglich Infrastruktur, wie in etwa Häfen, Industriekomplexe, Flughäfen oder Handelsrouten als weitere Knoten anzuhängen.

Eine solche Darstellung kann nützlich sein, um ein Verständnis für die Hierarchie und Struktur geographischer Daten zu vermitteln und komplexe räumliche Beziehungen zwischen verschiedenen geographischen Einheiten zu visualisieren.

Beispielsweise könnte eine Baumstruktur verwendet werden, um die geographischen Merkmale eines Landes zu erfassen, indem die Baumstruktur die verschiedenen Ebenen der Verwaltungseinheiten darstellt, beginnend mit den höchsten Ebenen wie Kontinenten und Regionen und dann weiter nach unten bis hin zu Ländern.

Durch die Verwendung von Baumstrukturen zur Darstellung historischer geographischer Daten können historische Trends und Entwicklungen leichter verstanden werden, was dazu beitragen kann, ein umfassenderes Verständnis der Geschichte von Ländern und Regionen zu erlangen.

5. Modellbildung

5.1. Kriterienkatalog zur Kategorisierung historischer Prozesse

Die maßgebenden Kriterien des Simulationsmodells, bereits in Kapitel 3.2 beschrieben, können auf 5 zentrale Punkte heruntergebrochen werden:

1. Geographische Faktoren
2. Geopolitische Faktoren
3. Wirtschaftliche Faktoren
4. Soziopolitische Faktoren
5. Strategische Faktoren

5.1.1. Geographische Faktoren

Geographische Faktoren haben einen starken Einfluss auf die Prozesssimulation und dienen als Grundlage für die restlichen Simulationskomponenten. Um die Gegebenheiten zur Zeit des Zweiten Weltkriegs möglichst akkurat wiederzugeben, ohne die Simulation zu komplex zu gestalten, wurde eine Kompromisslösung zwischen den bereits gefundenen Konzepten gewählt, bei der insgesamt 161 Regionen definiert werden. Davon sind 65 davon Seeregionen. Länder, welche nicht am Kriegsgeschehen teilnehmen werden als neutrale Länder repräsentiert. Verbindungen zwischen Seeregionen, welche durch Landmassen verlaufen, werden durch einen Kanal dargestellt und ermöglichen so das leichte Fortbewegen. Inselregionen werden von Seezonen umfasst und sind nur per Transporter oder Lufteinheiten erreichbar. Diese Aufteilung repräsentiert die historischen Gegebenheiten detailliert genug, um den Simulationscharakter nicht zu verletzen, steigert den Komplexitätsgrad jedoch nicht übermäßig.

5.1.2. Geopolitische Faktoren

Geopolitische Faktoren wie politische Maßnahmen, Bündnisse und Nationalitäten, welche in diesem Simulationskonzept vorhanden sind, bauen direkt auf den soeben definierten geographischen Faktoren auf und bilden die historische Situation der unterschiedlichen Nationen zur Zeit des Zweiten Weltkriegs abstrahiert ab. So wurde die Vielzahl an Nationen zu den folgenden fünf großen Nationen zusammengefasst

- Vereinigte Staaten Amerikas
- Vereinigtes Königreich
- Sowjetunion
- Japanisches Kaiserreich
- Deutsches Reich

auf welche der Besitz der einzelnen Regionen zu Beginn der Simulation aufgeteilt wurde. Diese Nationen teilen sich wiederum in zwei Bündnisse, die Alliierten und die Axenmächte, welche die historische Einteilung der Nationen widerspiegeln. Ebenfalls wurden hier jene Regionen berücksichtigt, welche zur Zeit des Zweiten Weltkriegs eine neutrale Haltung einnahmen und wurden so als unbeteiligte Gebiete festgelegt. Die Ziele der beiden Bündnisse

innerhalb der Simulation, die 12 Hauptstädte von großer Bedeutung, wurden ebenfalls ihren historischen Gegebenheiten entsprechend platziert.

5.1.3. Wirtschaftliche Faktoren

Die in diesem Simulationskonzept behandelten wirtschaftlichen Faktoren beschränken sich auf die Wirtschaftsstärken der einzelnen Regionen, das Startvermögen der Nationen und die Verteilung von Industriekomplexen. Hier wurde beim Komplexitätsgrad wieder ein Mittelweg zwischen den verglichenen Konzepten gegangen, da Regionen zwar unterschiedliche Wirtschaftsstärken aufweisen, dieses sich jedoch nicht ändert oder durch den Bau von Infrastruktur verbessert wird und Industriekomplexe lediglich zur Aneignung neuer Truppen dienen.

5.1.4. Soziopolitische Faktoren

Ein weiteres Kriterium ist das Abbilden von Ereignissen soziopolitischer Natur. Dazu gehören vor allem, wie sich die Einstellung der Bevölkerung einer Nation auf diese auswirkt und der Umgang von Nationen mit unterschiedlichen religiösen, kulturellen oder ethnischen Gruppierungen in deren Bevölkerung. Derartige Faktoren wurden in dieser Prozesssimulation nicht eingebunden, um eine rein objektive Betrachtung von historisch konfliktbehafteten Prozessen und deren Vorgänge ohne die Einbindung jeglicher politischer oder ethischer Thematik zu ermöglichen.

5.1.5. Strategische Faktoren

Bei den strategischen Faktoren liegt bei dieser Prozesssimulation das Augenmerk, da hier eine vergleichsweise hohe Komplexität erreicht wurde, mit einer Vielzahl von strategischen Mustern zur Repräsentation von verschiedenen Truppenbewegungen und Gefechtslagen. Zu den unterschiedlichen Faktoren, welche zur Simulation von Truppenmanöver beitragen, zählen unter anderem

- Die Bewegungsreichweite unterschiedlicher Truppen
- Erreichbarkeit von Landeplätzen bei Flugmanöver
- Das schnelle Vorrücken von gepanzerten Divisionen durch sogenannte Blitz-Manöver
- Den defensiven Vorteil, welchen eine Verteidigende Nation in Gefechten besitzt
- Der Transport von Truppen über Seeregionen durch Transporter bzw. Flugzeugträger
- Unterstützendes Artillerie-Feuer bei Infanterie-Manöver
- Amphibische Angriffe auf Küstenregionen
- Spezielle Manöver von U-Booten zum Untertauchen und Umgehen von Feinden
- Das Verhindern von eben diesen U-Boot Manövern durch die Präsenz von Zerstörer

Des Weiteren sind die unterschiedlichen Eigenschaften von Einheiten in Gefechten ein essentieller Faktor, da diese den Verlauf der Simulation vorgeben und so für den Erhalt des Simulationscharakters von äußerster Wichtigkeit sind. Hierfür wird, um die chaotische und unbestimmbare Natur solcher Auseinandersetzungen abzubilden, auf eine chancenbasierte Gefechtsabhandlung gesetzt, welche in der Form von abwechselndem Beschuss von Angreifer und Verteidiger durch das Werfen eines Würfels gegen eine bestimmte, von den Gefechtseigenschaften der feuern den Einheit abhängigen Zahl. Wie diese unterschiedlichen Gefechtseigenschaften zustande gekommen sind, wird in dem folgendem Kapitel behandelt.

5.2. Mathematische Modelle

5.2.1. Disproportionale Repräsentation von Simulationskomponenten

Zur Vereinfachung der zugrunde liegenden historischen Prozesse, wurden diese stark abstrahiert. Wie genau die verschiedenen Verhältnisse für die abstrahierte Darstellung der Simulationskomponenten hergeleitet wurden ist aus folgender Tabelle abzulesen

Realitätswerte	Großbritannien	Sowjetunion	United State	Deutsches Reich	Japan
Infanterie	4 090 000	11 340 000	3 970 000	12 220 000	2 840 000
Artillerie	36 400	541 900	219 000	101 200	24 000
Panzer	28 500	110 000	91 270	61 250	7 200
Flugzeuge	133 000	162 000	329 000	126 000	90 000
U-Boot	76	482	193	312	63
Zerstörer	205	81	359	162	216
Kreuzer	90	10	93	39	73
Flugzeugträger	13	0	18	2	16
Schlachtschiff	24	3	32	23	13

Tabelle 5.1.: Realitätswerte

	Maßstab
Infanterie	1 : 300 000
Artillerie	1 : 12 500
Panzer	1 : 2 500
Flugzeuge	1 : 7 500
U-Boot	1 : 20
Zerstörer	1 : 15
Kreuzer	1 : 7
Flugzeugträger	1 : 2
Schlachtschiff	1 : 3

Tabelle 5.2.: Maßstab

Simulationswerte	Großbritannien	Sowjetunion	United State	Deutsches Reich	Japan
Infanterie	14	38	13	41	10
Artillerie	3	43	18	8	2
Panzer	12	44	37	25	3
Flugzeuge	18	22	44	17	12
U-Boot	4	24	10	16	3
Zerstörer	14	6	24	11	15
Kreuzer	13	2	13	6	11
Flugzeugträger	7	0	9	1	8
Schlachtschiff	8	1	11	8	4

Tabelle 5.3.: Simulationswerte

Die hier angeführten Werte zeigen den Stand von Truppenzahlen der unterschiedlichen Nationen im Jahre 1942 und wurden von folgender Quelle [7] übernommen. Ebenfalls zu beachten ist, dass die aquatischen Streitkräfte Frankreichs, welche in der hierfür bezogenen Quelle [7] aufgelistet werden, aufgrund der britischen Operation Catapult ein nicht Faktor sind. Informationen über Operation Catapult wurden von hier [8] und hier [9] bezogen.

5.2.2. Deduktion der Eigenschaften von Simulationskomponenten in auf Zufall basierten Auseinandersetzungen

Um die Gefechtseigenschaften der Einheiten darzustellen, wird ein Erwartungswert basiertes Konzept umgesetzt.

Um die Stärke einer Infanterie Einheit zu berechnen wurden folgende Daten aus der Schlacht um Stalingrad [10] [11] gesammelt:

1. angreifende Infanterie Einheiten der Deutschen Armee (850 000)
2. verlorene Einheiten der Sowjetunion (320 000)

Anhand dieser Werte können nun 2 Erwartungswerte berechnet werden, welche miteinander verglichen werden können.

Zuerst kann der Erwartungswert der reellen Schlacht berechnet werden, welcher sich wie folgt zusammensetzt:

$$850000 * \frac{1}{6} = 141666$$

Somit beträgt der Erwartungswert bei einer Trefferquote von $\frac{1}{6}$ und einer Truppenanzahl von 850 000:

$$\mu \approx 141666$$

Danach wird der Erwartungswert mit der gegebenen Trefferquote ($\frac{1}{6}$) aus dem Prototypen berechnet.

$$3 * \frac{1}{6} = 0.5$$

Hierbei ist folgender wert bei einer im Maßstab 1 : 300 000 getreuen Truppenstärke von ≈ 3 abzulesen:

$$\mu \approx 0.5$$

Um einen logischen Vergleich zu ziehen, werden beide Werte auf eine ansehnliche Basis gebracht.

$$0.5 * 2 = 1$$

$$141666 * 2 = 283333$$

Somit ist das berechnete Ergebnis von 1 : 283 333 und der tatsächliche Verlust von 320 000 annähernd gleich und es kann gesagt werden das die Trefferquoten realitätsgetreu bleiben.

$$283333 \approx 320000$$

Berechnet man nun die prozentuale Genauigkeit des Modells, kann man feststellen, das diese bei rund 89% liegt.

$$\frac{283333}{320000} \approx 89\%$$

Folgende weitere historische Gefechte wurden als Grundlage für jeden Truppentypen herangezogen: die Schlacht von Dünkirchen [12], die Schlacht um Rostow [13], die Schlacht am Asowschen Meer [14], bezogen von folgender Quelle [15]

Daraus resultierten folgende Werte und Trefferquoten:

	Angreifer	Verteidiger
Infanterie	1 ($\frac{1}{6}$)	2 ($\frac{2}{6}$)
Artillerie	2 ($\frac{2}{6}$)	2 ($\frac{2}{6}$)
Luftwaffe	-	-
Panzer	3 ($\frac{3}{6}$)	3 ($\frac{3}{6}$)
Jäger	3 ($\frac{3}{6}$)	4 ($\frac{4}{6}$)
Bomber	4 ($\frac{4}{6}$)	1 ($\frac{1}{6}$)
U-Boot	2 ($\frac{2}{6}$)	1 ($\frac{1}{6}$)
Transporter	-	-
Zerstörer	2 ($\frac{2}{6}$)	2 ($\frac{2}{6}$)
Kreuzer	3 ($\frac{3}{6}$)	3 ($\frac{3}{6}$)
Flugzeugträger	1 ($\frac{1}{6}$)	2 ($\frac{2}{6}$)
Schlachtschiff	4 ($\frac{4}{6}$)	4 ($\frac{4}{6}$)

Tabelle 5.4.: Trefferquoten

5.2.3. Erhebung wirtschaftlicher Eigenschaften anhand des GDP Modells

Um repräsentative Werte für eine Wirtschaftsstärke einer Region zu erheben, wurde das GDP [16] Modell herangezogen. Daraus resultierte eine Liste von wirtschaftlichen Werten, welche auf eine Skala von 1-12 eingegliedert wurden:

Nation	Wert
Vereinigte Staaten	12
Deutschland	10
Frankreich	6
Norwegen	2
...	...

Tabelle 5.5.: Wirtschaftsstärken

5.2.4. Ausarbeitung von situationsabhängigen Eigenschaften der Simulationskomponenten

Folgende Überlegungen wurden aus der Analyse der betrachteten Schlachten gewonnen.

- In einer Vielzahl von Schlachten, wie z.B. in der Schlacht von Dünkirchen [12], wird gezeigt, dass ein Verteidiger durch Gegebenheiten wie Engpässe und das Halten von defensiven Positionen einen substanziellen Vorteil gegenüber einem Angreifer besitzt,

was in der Form eines um 1 erhöhten Würfelwert für verteidigende Infanterie in der Simulation reflektiert wird.

- Ebenfalls ersichtlich ist der Fakt, dass Infanterie-Regimenter [17] unbedingt in Kooperation mit Artillerie agieren wollen, da alle der unterschiedlichen Nationen ihre Infanterie-Divisionen [18] aus sowohl Infanterie als auch Artillerie und unterstützenden Truppen aufbauen. Dies wird in der Simulation dargestellt indem Infanterie durch die Präsenz von verbündeter Artillerie einen um 1 erhöhten Würfelwert erhalten.
- Aus einigen Schlachten wird ersichtlich, dass die um ein vielfaches höhere Geschwindigkeit von Panzerwagen im Vergleich zu Infanterie einen erheblichen Vorteil bietet, da diese neuen Strategien wie den Sichelschnittplan [19], welcher während der Schlacht um Belgien [20] eine große Rolle spielte, ermöglicht. Deshalb sind Panzerwagen in der Simulation dazu im Stande, sich durch feindliche Gebiete hindurch zu bewegen und Fronten so schnell zu verschieben.
- Eine weitere spezielle Eigenschaft von Einheiten, welche in die Simulation eingebunden wurde, ist die Fähigkeit von U-Booten, sich durch Untertauchen unentdeckt fortzubewegen, was sowohl zur Vermeidung von unerwünschten Auseinandersetzungen als auch zum Umgehen von feindlichen Flotten genutzt werden kann. Auskunft über den Einsatz von U-Booten wurde sich in dieser Quelle [21] verschaffen.
- Als Kehrseite zu den Eigenschaften von U-Booten unerkannt zu bleiben, dienen die Zerstörer, welche durch den Einsatz des Sonars und Wasserbomben die U-Boote bekämpfen können. Dies wird in der Simulation dadurch repräsentiert, dass U-Boote sich nicht an Zerstörern vorbei bewegen können und auch nicht in Kämpfen untertauchen können, in welchen feindliche Zerstörer vorhanden sind. Dies ist wieder aus der folgenden Quelle [21] ersichtlich.

6. Visualisierungsmethodiken des Prototypen

6.1. Implementierung der Kartographischen Visualisierung

Zur Darstellung der prozessrelevanten Informationen wurde eine kartographische Darstellung gewählt. Dargestellt werden Regionen und Fraktionen. Diese beinhalten wiederum Truppen, Industriekomplexe und Hauptstädte.

6.1.1. Fraktionen

Grundsätzlich wird zur Unterscheidung der Fraktionen mit 5 Farben gearbeitet.



Abbildung 6.1.: Fraktionsfarben

Diese repräsentieren folgende Fraktionen:

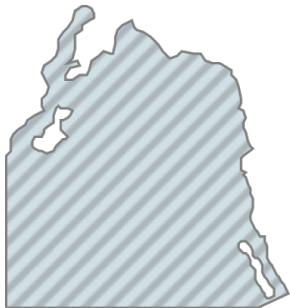
1. Sowjetunion
2. Deutsches Reich
3. Vereinigtes Königreich
4. Japan
5. Vereinigte Staaten

6.1.2. Aufteilung der Regionen

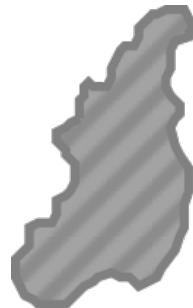
Regionen teilen sich auf 3 verschiedene Typen auf.

1. Seezonen
2. Landregionen
3. neutrale Regionen

Seezonen: können Einheiten befahren, aber nicht besetzt werden können. Landregionen: sind eine Fraktion zugewiesen, und neutrale Regionen, mit denen keine Interaktion stattfinden kann.



(a) Seezone



(b) Neutrale Region



(c) Fraktions Region

Abbildung 6.2.: Darstellungsarten verschiedener Regionen

6.1.3. Darstellung der Truppen

Die Visualisierung der Truppen auf einer Region erfolgt als Truppenabzeichen. Hierbei wird zwischen Land und Seeregionen unterschieden. Die Anzahl der Truppen wird mithilfe einer Zahl dargestellt. Truppen können im Verlauf der Simulation hinzugefügt und entfernt werden.



(a) Infanterie



(b) Panzer



(c) Luftabwehr



(d) Artillerie



(e) Bomber



(f) Jäger

Abbildung 6.3.: Darstellung Landtruppen [22]



(a) Kreuzer



(b) Schlachtschiff



(c) U-Boot



(d) Zerstörer



(e) Flugzeugträger



(f) Transporter

Abbildung 6.4.: Darstellung Seetruppen [22]

6.1.4. Sonstige Visualisierungselemente

Regionen mit der Möglichkeit zur Truppenausbildung besitzen einen Industriekomplex angezeigt. Die Lage von Hauptstädten wird ebenfalls in einer Region repräsentiert.



(a) Industriekomplex



(b) Hauptstadt

Abbildung 6.5.: Sonstige Visualisierungselemente

6.1.5. Simulationsinteraktionen

Wird eine Region ausgewählt, ändert sich die Farbe des Landes auf einen gelben Farbton. Zusätzlich werden Basisinformationen für ein Land angezeigt. Wird eine bereits ausgewählte Region erneut angeklickt, ändert sich die Farbe des Landes auf die ursprüngliche Fraktionsfarbe und die Informationen verschwinden.

Bei Bewegungsvorgängen einer Truppe werden mögliche Regionen, welche angesteuert werden können, mit einer blauen Farbe hinterlegt. So ist es für einen Benutzer klar ersichtlich, welche Wege genutzt werden können und wie hoch die Reichweite einer Truppe ist.

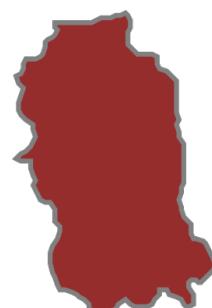
Werden Truppen in eine feindliche Region gesendet, wird ein Kampf ausgelöst. Regionen auf denen gekämpft wird, werden mit einem dunkelroten Farbton hinterlegt.



(a) Aktive Region



(b) Zielregion



(c) Schlachtfeld

Abbildung 6.6.: Darstellungsarten verschiedener Interaktionen

Um die Bewegungen von Truppen zu visualisieren, werden Pfeile zwischen zwei Regionen generiert. Diese verlaufen durch den Mittelpunkt aller Regionen, durch welche Truppen verschoben werden.

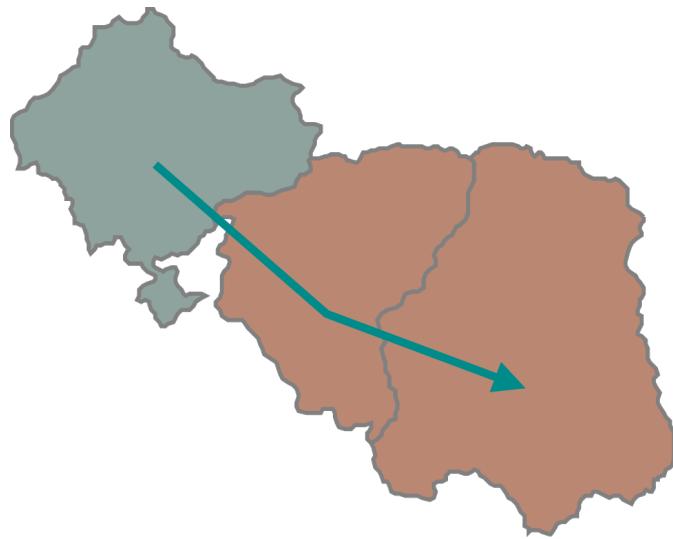


Abbildung 6.7.: Bewegungsaktion

6.2. Konzeptionierung der Phaseninteraktion

Um mit den Simulationskomponenten zu interagieren wird ein Drawer für jede Phase zur Verfügung gestellt.

Während der Kaufphase wird für jede Truppe eine Schaltfläche zum Hinzufügen und Entfernen bereitgestellt.

LAND	MEER	LUFT	INDUSTRIE	
Ang	Ver	Bew	Kosten	Kaufen
	1	2	1	3

Abbildung 6.8.: Kaufbildschirm

In einer Region werden pro Truppe zwei Schaltflächen angezeigt. Diese ermöglichen es dem Benutzer, ein Ziel festzulegen oder zu entfernen.



Abbildung 6.9.: Bewegungsaktion

Um gekaufte Truppen zu mobilisieren, steht eine Funktion zur Verfügung, die für jede Truppe einen eigenen Knopf bereitstellt. Durch Betätigung dieses Knopfes kann eine Region mit einem Industriekomplex ausgewählt werden, auf der die entsprechende Truppe platziert wird.

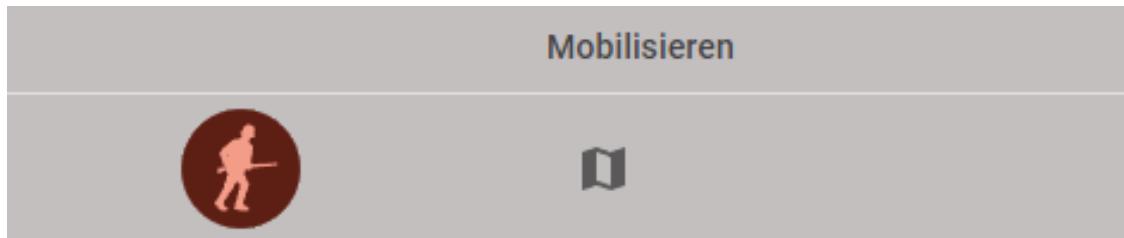


Abbildung 6.10.: Mobilisierung

6.3. Bereitstellung der aktuellen Informationen

Die Informationsleiste bietet dem Benutzer eine Vielzahl an wichtigen Daten und Informationen, die für eine erfolgreiche Simulationsstrategie von großer Bedeutung sind.

Dazu gehören nicht nur die Anzahl der Runden, sondern auch die aktuelle Fraktion und Phase der Simulation. Zusätzlich werden eroberte Hauptstädte in verschiedenen Farben hervorgehoben, um einen schnellen Überblick über den aktuellen Stand der territorialen Kontrolle zu geben. Hauptstädte der Achsenmächte sind in Rot markiert, während die der Alliierten in Blau hervorgehoben werden.

Auf diese Weise kann der Benutzer auf einen Blick erkennen, welche Gebiete im Besitz welcher Fraktionen sind und seine Strategie entsprechend anpassen.



Abbildung 6.11.: Infoleiste

Die Truppenleiste enthält Informationen darüber, wie viele Einheiten sich in der Region befinden und welcher Fraktion sie angehören. Jede Fraktion wird durch eine bestimmte

Farbe repräsentiert, die in der Truppenleiste angezeigt wird. So kann der Benutzer auf einen Blick sehen, welche Fraktionen in einer Region vertreten sind und wie stark jede Fraktion dort vertreten ist.

Darüber hinaus gibt die Truppenleiste auch Auskunft darüber, ob die Region einen Industriekomplex oder eine Hauptstadt enthält.

Insgesamt liefert die Truppenleiste dem Benutzer also wichtige Informationen darüber, welche Truppen in einer Region stationiert sind, welche Fraktionen dort vertreten sind und welche strategischen Ressourcen in der Region vorhanden sind. Damit hat der Benutzer eine bessere Übersicht über die Karte und kann seine Strategie optimal planen.



Abbildung 6.12.: Truppenleiste

6.4. Visualisierung des Schlachtfeldes

Ein Schlachtfeld setzt sich aus verschiedenen Truppengruppen zusammen, auf denen die Würfelanzahl angegeben ist, ab der eine Truppe einen Treffer erzielt. Die Karten ermöglichen eine schnelle Einschätzung der Stärken und Schwächen einer Einheiten und erleichtern die Planung von taktischen Manövern auf dem Schlachtfeld.

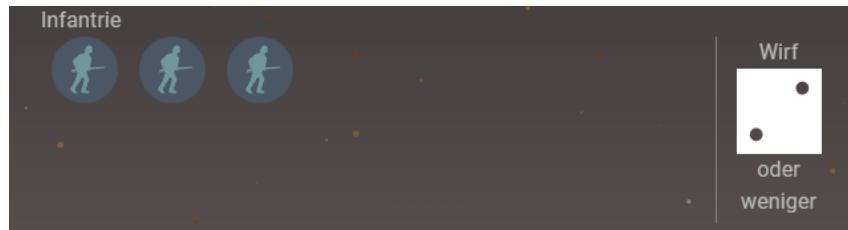


Abbildung 6.13.: Truppenanzeige

In der Mitte des Schlachtfelds befinden sich sechs Würfel, die jeweils eine Zahl von 1 bis 6 repräsentieren. Diese Würfel sind Teil der Würfelanzeige und zeigen die aktuellen Treffer an, die von den angreifenden Truppen erzielt wurden. Jeder Würfel ist mit einer bestimmten Zahl beschriftet und hat auch einen kleinen Text, der anzeigt, wie oft diese Zahl gewürfelt wurde.

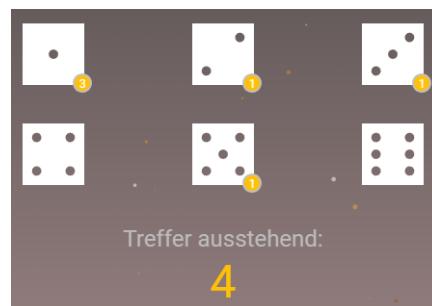


Abbildung 6.14.: Trefferanzeige

Wenn eine Truppe auf dem Schlachtfeld verwundet wird, wird dies durch eine Farbänderung gekennzeichnet. Die Farbänderung zeigt an, dass die Truppe nun nicht mehr angegriffen werden kann und in der nächsten Runde vom Schlachtfeld entfernt wird.

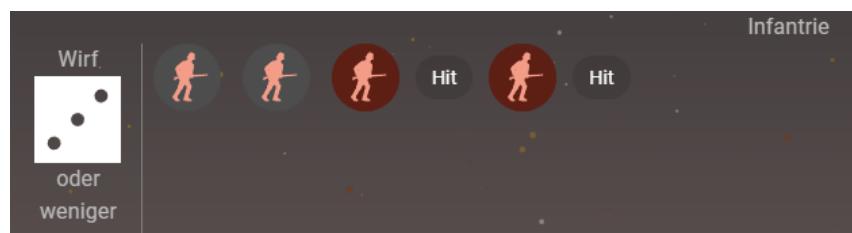


Abbildung 6.15.: Verwundetenanzeige

7. Diskussion eines vereinfachten Modells

Prozesssimulationen historischer Ereignisse unterscheiden sich in ihrer Komplexität aufgrund des zugrundeliegenden Prozessmodells und der Visualisierung.

7.1. Erhebung eines Referenzmodells

Hier wird die eigens entwickelte Prozesssimulation mit dem Simulationsmodell von Hearts of Iron IV [4] verglichen. Dazu soll die Umsetzung der essenziellen Faktoren im HoI4 Modell, siehe Kapitel 3.2, bewertet werden.

7.1.1. Bewertung geographischer Faktoren

- Vorteile: Durch die umfangreiche und detailliertere Abbildung von geographischen Faktoren wird die Realität sehr treu wiedergegeben. Dadurch dass die folgenden Faktoren maßgeblich von den geographischen Faktoren abhängen, bildet dies die Grundlage für eine sehr realitätsnahe Simulation, was einen sehr vertiefenden Wissensgewinn ermöglicht.
- Nachteile: Durch die hoch komplexe Natur der hier abgebildeten historischen Prozesse kann dies jedoch dazu führen, dass ein Simulant den Überblick über die vielen Vorgänge der Simulation verliert.

7.1.2. Bewertung geopolitischer Faktoren

- Vorteile: Die Abbildung von einem Großteil der Nationen, welche zur Zeit des Zweiten Weltkrieges existierten, ermöglicht eine sehr realitätstreue Wiedergabe von unterschiedlichen politischen Verhältnissen zwischen Nationen. Eine solche Abbildung von politischen Prozessen ermöglicht das Simulieren einer Vielzahl an Faktoren, welche zur Entstehung von konfliktbasierten Ereignissen beitragen.
- Nachteile: Viele der inkludierten Nationen haben nur einen geringen Einfluss auf den Verlauf der Simulation und erhöhen so deren Komplexität ungerechtfertigt. Außerdem gilt abermals, dass die hoch komplexe Wiedergabe von politischen Prozessen zur Verwirrung des Benutzers führen kann.

7.1.3. Bewertung wirtschaftlicher Faktoren

- Vorteile: Das detaillierte Darstellen von wirtschaftlichen Faktoren ermöglicht das Simulieren der unterschiedlichen Bedingungen, welche zum Geschehen von konfliktbasierten Ereignissen gegeben sein müssen. Dies macht es dem Simulanten möglich, die grundlegenden Einflüsse auf konfliktbasierte Ereignisse besser zu verstehen.
- Nachteile: Solche wirtschaftlichen Faktoren sind nur indirekt an den historischen Ereignissen beteiligt, wodurch eine detaillierte Abbildung dieser von dem Fokus auf konfliktbasierte Ereignisse abweicht. Das kann zur Folge haben, dass die Simulation ohne eine direkte Verbesserung des Verständnisses von eben diesen Ereignissen an Komplexität gewinnt.

7.1.4. Bewertung soziopolitischer Faktoren

- Vorteile: Durch die Abbildung von soziopolitischen Faktoren wird konfliktbehafteten Ereignissen Kontext gegeben. Da eine Vielzahl an historischen Ereignissen entweder direkt oder indirekt von diesen Faktoren beeinflusst waren, ermöglicht die realitätsnahe Wiedergabe dieser ein verbessertes Verständnis über die verschiedenen Einflüsse auf konfliktbehaftete Ereignisse für den Simulanten.
- Nachteile: Soziopolitische Faktoren haben wieder nur indirekten Einfluss auf das Geschehen von konfliktbehafteten Ereignissen, was deren detaillierte Abbildung aus Gründen der Übersichtlichkeit in Frage stellt. Außerdem kann die Inkludierung dieser Faktoren zum Aufkommen von Fragen über ethische Thematiken führen, was für die rein objektive Betrachtung von konfliktbehafteten Ereignissen nicht von Vorteil ist.

7.1.5. Bewertung strategischer Faktoren

- Vorteile: Durch die Repräsentation der vielen unterschiedlichen Einflüsse auf den Verlauf und Ausgang von konfliktbehafteten Ereignissen wird eine essenzielle Wissensbasis für das Verstehen solcher Prozesse geboten.
- Nachteile: Die Komplexität, in welcher diese vielen Einflüsse und Taktiken wiedergegeben werden, kann für historisch Interessierte abschreckend wirken. Dies erfolgt daraus, dass es einen gewissen Aufwand erfordert, um sich mit den vorhandenen strategischen Faktoren vertraut zu machen.

7.2. Bewertung des Great Powers Modell

Nun wird die Umsetzung der essenziellen Faktoren im Great Powers Modell, siehe Kapitel 5.1, bewertet.

7.2.1. Bewertung geographischer Faktoren

- Vorteile: Das Modell von Great Powers bietet in Hinsicht auf geographische Faktoren ein akzeptables Maß an Realitätstreue. Dadurch, dass die restlichen Faktoren maßgebend von diesen abhängen, bietet dieses Modell durch seine Unterteilung der Weltkarte in 161 Regionen eine gute Darstellung der historischen Gegebenheiten für die restliche Simulation.
- Nachteile: Es werden einige Aspekte von sowohl den geographischen als auch den anderen von diesen abhängigen Faktoren nur abstrahiert dargestellt, was zu gewissen Ungenauigkeiten in der Simulation führen kann. Dadurch entspricht das gewonnene Wissen eventuell nicht ganz den historischen Gegebenheiten.

7.2.2. Bewertung geopolitischer Faktoren

- Vorteile: Durch die Zusammenfassung auf nur 5 Nationen schafft das Great Powers Modell einen klaren Überblick über sowohl die Aufteilung der Regionen als auch die Bündnisse zwischen den Nationen.
- Nachteile: Auf die Abbildung einer Vielzahl von politischen Prozessen wird durch dieses Zusammenfassen verzichtet, was zu einer erhöhten Entfernung zu den historischen Gegebenheiten führen kann.

7.2.3. Bewertung wirtschaftlicher Faktoren

- Vorteile: Dadurch, dass wirtschaftliche Faktoren nur so abgebildet werden, dass sie einen direkten Einfluss auf konfliktbehaftete Ereignisse haben, wird der Fokus der Simulation auf das Darstellen eben solcher Ereignisse klar. Dies ermöglicht es einem Simulanten, einen guten Überblick über die unterschiedlichen Prozesse zu erlangen.
- Nachteile: Wirtschaftliche Faktoren liegen der Entwicklung von konfliktbehafteten Ereignissen zugrunde. Durch eine starke Abstrahierung dieser geht das Verständnis von grundlegenden Einflüssen auf eben diese Ereignisse eventuell verloren.

7.2.4. Bewertung soziopolitischer Faktoren

- Vorteile: Da soziopolitische Faktoren im Great Powers Modell nicht repräsentiert werden, bleibt die Komplexität der Simulation gering. Außerdem führt das Weglassen dieser Faktoren zur Vermeidung von Fragen zu ethischen Thematiken, was den Fokus auf konfliktbehaftete Ereignisse stärkt.
- Nachteile: Soziopolitische Faktoren haben jedoch einen maßgebenden Einfluss auf das Geschehen und den Verlauf von konfliktbehafteten Ereignissen, wodurch einige äußerst wichtige Faktoren nicht abgebildet sind. So können Simulanten diese Einflüsse nicht simulieren, wodurch abermals weniger über die komplexen Vorgänge von historischen, konfliktbehafteten Prozessen gelernt werden kann.

7.2.5. Bewertung strategischer Faktoren

- Vorteile: Durch das Darstellen von einigen strategischen Faktoren werden konfliktbasierte Ereignisse auf einer abstrahierten Weise wiedergegeben, was den Wissensgewinn erleichtert. Dies geschieht durch eine verringerte Anzahl von Simulationsaspekten, welche das Verstehen der verschiedenen Vorgänge durch eine reduzierte Simulationskomplexität ermöglicht.
- Nachteile: Einige maßgebende Aspekte von historischen, konfliktbehafteten Ereignissen werden nur teils oder gar nicht abgebildet, was den Verlust von Wissen über eben diese Ereignisse aufweisen kann.

7.3. Kontrastierung mit erhobenem Referenzmodell

7.3.1. Anforderungen

Um eine einfache, jedoch realitätsgestreue Simulation eines historischen Ereignisses zu schaffen ist es notwendig, folgende Anforderungen zu berücksichtigen:

1. Übersichtlichkeit: Es ist notwendig dass ein Benutzer den Überblick über das Geschehen behält.
2. Reduzierung: Nur durch eine Reduzierung überflüssiger Faktoren ist es möglich den Fokus auf die relevanten Thematiken zu lenken.
3. Simplizität: Um eine lehrreiche Simulation ohne komplexe Vorgänge zu schaffen ist es von Nöten dem Benutzer gegenüber eine gewisse Einfachheit zu behalten.

7.3.2. Modellbeschreibung des Prototypen

Der Prototyp erfüllt diese Anforderungen folgendermaßen:

- Übersichtlichkeit: Um die Übersichtlichkeit zu gewährleisten werden detaillierte geographische Faktoren vernachlässigt. Eine genaue soziopolitische Erläuterung ist überflüssig und ist somit nicht relevant für den Prototypen.
- Reduzierung: Die Reduzierung des Modells wird in Form der Zusammenfassung auf 5 Fraktionen umgesetzt. Weiteres werden Angriffs- und Verteidigungswerte auf einen nummerischen Wert zwischen 1-6 zusammengefasst. Die Flexibilität der Truppen wird auf eine gerade Anzahl beschränkt, welche angibt, wie viele Länder diese durchqueren kann.
- Simplizität: Um dem Benutzer die komplexe Thematik von politischen Entscheidungen hinwegzunehmen wird auf diese Funktion verzichtet. Auch komplexe militärische Operationen werden vereinfacht abgebildet um die Einfachheit zu behalten

Somit wurde die Adaptierung des HoI4 Modells nicht weiter in Betracht gezogen da dieses Modell eine weitaus komplexere Repräsentation der Ereignisse bereitstellt, und diese den Anforderungen nicht entsprechen.

8. Bewertung von Visualisierungsmethoden zur Darstellung historischer Prozesse

8.1. Diskussion und Bewertung der Kartographische Darstellung

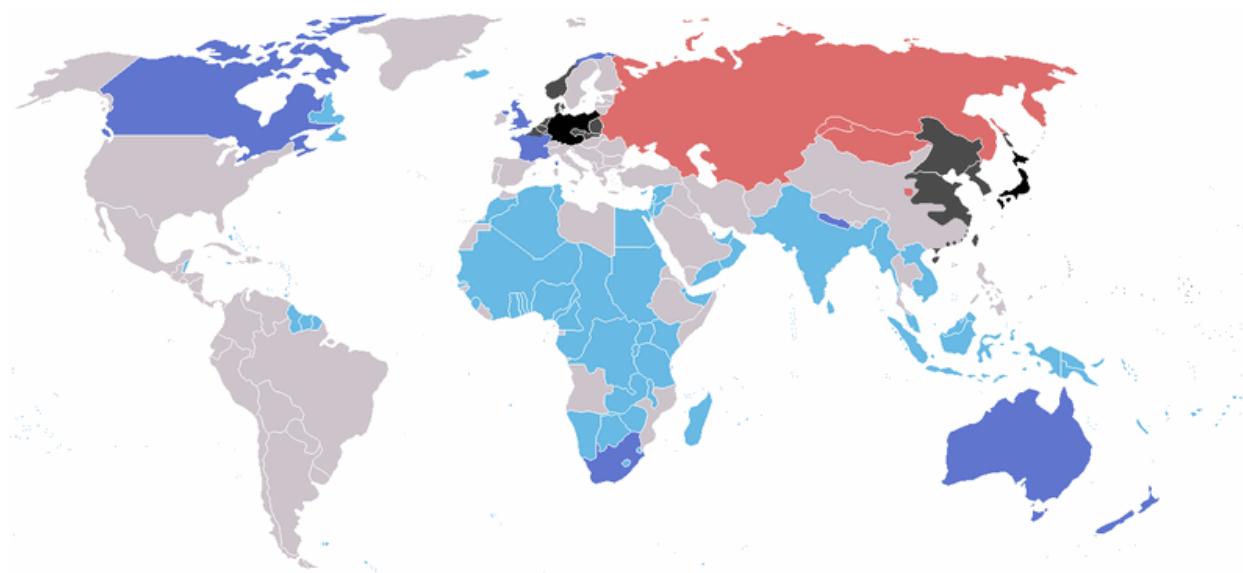


Abbildung 8.1.: Verwendung des kartographischen Modells im Kontext historischer Ereignisse

Vorteile der kartographischen Visualisierung:

1. Übersichtlichkeit: Kartographische Visualisierungen bieten eine klare Übersicht über geographische Daten und Informationen und ermöglichen es, komplexe Zusammenhänge und Muster schnell zu erkennen.
2. Präzision: Karten können sehr präzise sein, wenn sie auf hochwertigen Daten basieren und die richtige Technik verwendet wird.
3. Verständlichkeit: Kartographische Visualisierungen sind oft leichter zu verstehen als numerische Daten oder Tabellen. Durch die Verwendung von Karten können komplexe geographische Informationen einfach und schnell vermittelt werden.

Nachteile der kartographischen Visualisierung:

1. Verzerrungen: Eine der größten Herausforderungen bei der kartographischen Visualisierung ist, dass die Darstellung der Erdoberfläche auf einer flachen Karte zwangsläufig zu Verzerrungen führt. Je nach Art der Projektion können Entfernung, Flächen oder Winkel falsch dargestellt werden.
2. Überlappung: Bei der Darstellung von mehreren Elementen auf einer Karte kann es vorkommen, dass sich diese überlappen und somit schwer zu unterscheiden sind.

8.1.1. Einsatzgebiet des Kartographischen Modells im historischen Kontext

Anwendungsbereiche des kartographischen Modells mit Bezug zu historischen Ereignissen sind:

Strategie Erprobung: Das kartographische Modell kann in der Militärstrategie eingesetzt werden, um die räumlichen Aspekte von militärischen Operationen und Taktiken zu visualisieren und zu analysieren. Historische Karten können verwendet werden, um die Regionen und Einheiten zu identifizieren, welche Einfluss auf den Ausgang einer Strategie hatten.

Kriegsberichterstattung: Das kartographische Modell kann in der Kriegsberichterstattung eingesetzt werden, um die Verteilung von Truppen, Industriekomplexe und anderen Ressourcen zu analysieren. Historische Karten und Daten können verwendet werden, um die Positionen der Armeen während Schlachten zu veranschaulichen und die Verluste und Gewinne auf beiden Seiten zu erörtern.

Grenzziehung und Territorialkonflikte: Das kartographische Modell kann in Grenzziehungs- und Territorialkonflikten eingesetzt werden, um die räumliche Verteilung von Gebieten und Grenzen zu visualisieren. Aufzeichnungen und Daten können beispielsweise verwendet werden, um die Entstehung und Veränderung von Bündnissen und Territorien im Laufe der Geschichte zu untersuchen.

Insgesamt kann das kartographische Modell in einer Vielzahl von Anwendungsbereichen im Zusammenhang mit historischen Konflikten eingesetzt werden, um wichtige räumliche Zusammenhänge zu visualisieren.

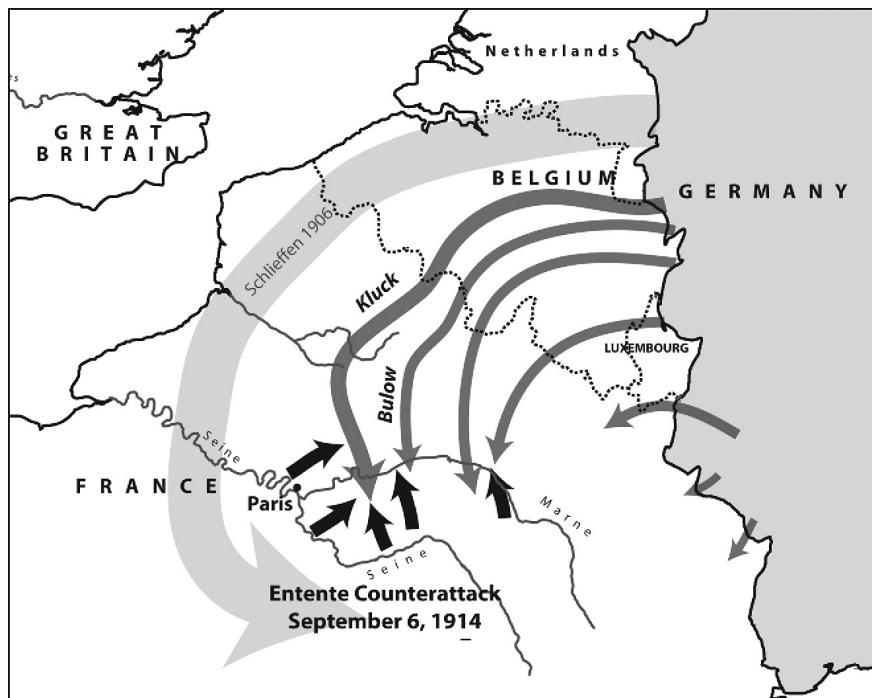


Abbildung 8.2.: Darstellung des Schlieffen-Plan auf einem kartographischen Modell [23]

8.2. Diskussion und Bewertung der Graph Darstellung



Abbildung 8.3.: Verwendung des Graph Modells im Kontext geographischer Daten [24]

Vorteile der Graph Visualisierung:

1. Einfache Erfassung von räumlichen Beziehungen: Die Visualisierung von Kartendaten als Graph ermöglicht es dem Betrachter, schnell räumliche Beziehungen zwischen verschiedenen Punkten zu erfassen, da die Knoten des Graphen die Punkte auf der Karte repräsentieren und die Kanten die Wege zwischen den Punkten darstellen.
2. Reduzierung der Datenkomplexität: Wenn große Mengen an Kartendaten vorhanden sind, kann die Visualisierung als Graph dazu beitragen, die Komplexität der Daten zu reduzieren, indem nur die relevanten Verbindungen und Routen dargestellt werden. Dadurch können wichtige Informationen schnell erfasst und genutzt werden.

Nachteile der Graph Visualisierung:

1. Begrenzte Informationsdichte: Graphen sind in der Regel nicht in der Lage, eine große Menge an Daten auf kleinem Raum darzustellen. Sie können jedoch durch die Verwendung von Farben, Größen und Formen von Knoten und Kanten verbessert werden.
2. Überlappung: Bei der Darstellung von vielen Knoten auf einem Graphen können sich diese oft überlappen und schwer zu unterscheiden sein.

8.2.1. Einsatzgebiet des Graph Modells

Anwendungsbereiche des Graphmodells mit Bezug zu historischen Ereignissen sind:

Analyse von militärischen Operationen: Das Graph-Modell kann eingesetzt werden, um militärische Operationen und die Beziehungen zwischen verschiedenen militärischen Einheiten darzustellen. Hiermit können Bewegungen von Truppen und die strategischen Entscheidungen, die zu bestimmten militärischen Ereignissen führten, besser verstanden werden.

Analyse von politischen Ideologien: Das Graph-Modell kann auch dazu genutzt werden, politische Ideologien und ihre Ausbreitung in verschiedenen Regionen darzustellen. Die Knoten im Graphen repräsentieren dabei politische Gruppierungen oder Ideologien, während die Kanten die Beziehungen zwischen ihnen darstellen, wie beispielsweise ideologische Einflüsse oder politische Allianzen.

Analyse der Bündnisse: Das Graph-Modell kann auch genutzt werden, um Regionsgrenzen zwischen historischen Bündnissen darzustellen. So können Verbindungen zwischen Kontinenten und Regionen analysiert werden. Dadurch können Historiker die Entwicklung von politischen Kooperationen zwischen Nationen besser verstehen.

Insgesamt kann das Graph Modell in einer Vielzahl von Anwendungsbereichen im Zusammenhang mit historischen Konflikten eingesetzt werden, um wichtige Zusammenhänge zu analysieren.

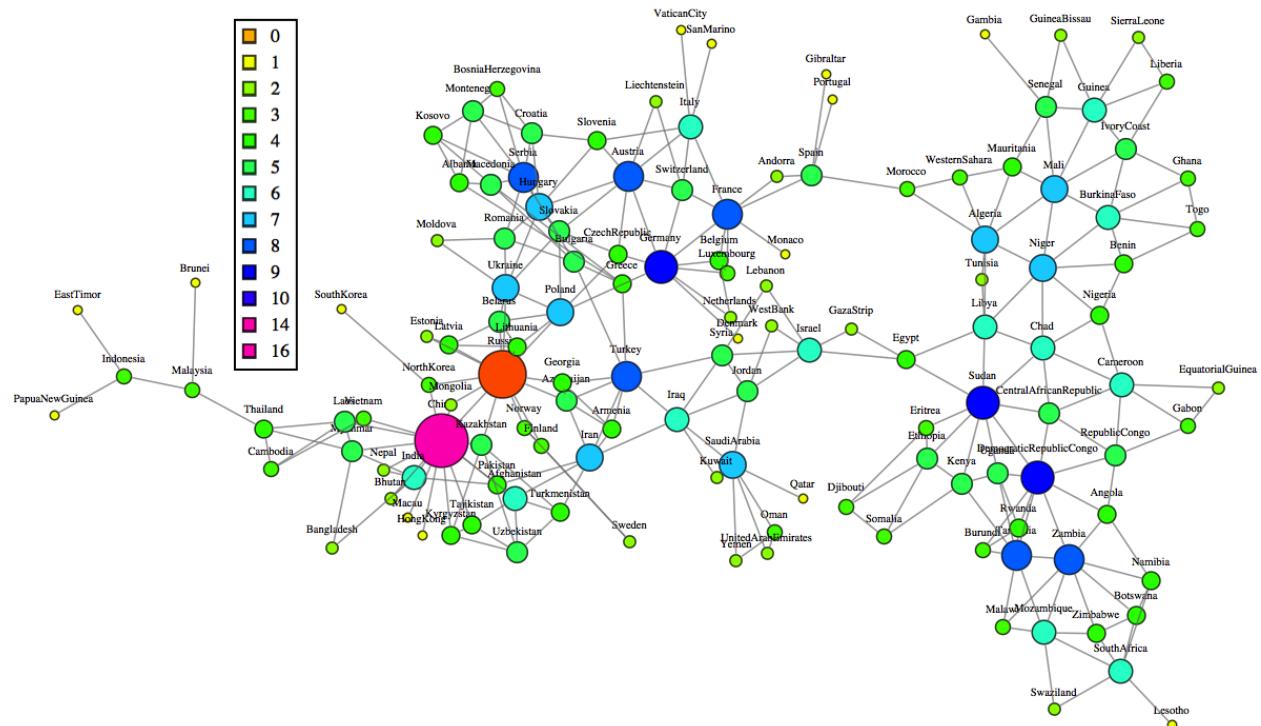


Abbildung 8.4.: Analyse von Nachbarländer anhand eines Graphen [25]

8.3. Diskussion und Bewertung der Baum Darstellung

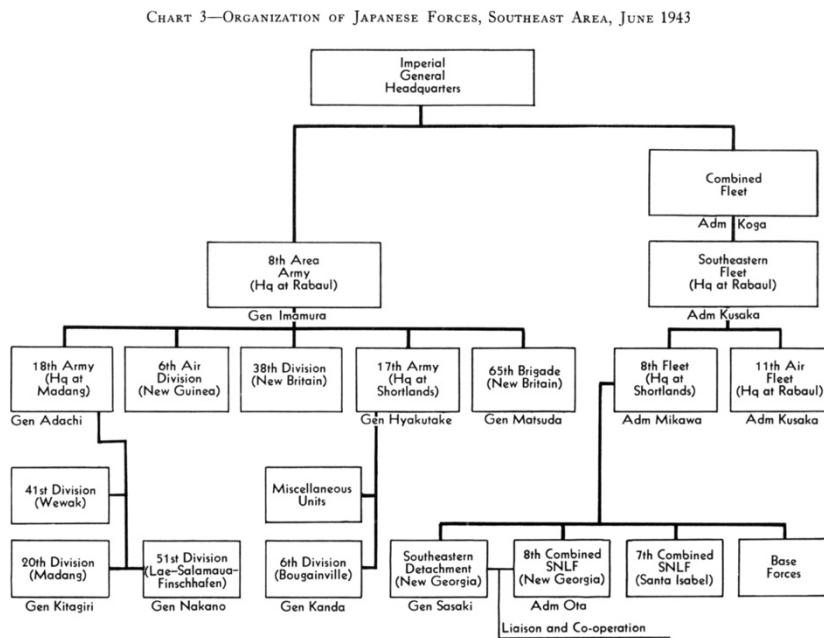


Abbildung 8.5.: Darstellung der Japanischen Truppen in Form eines Baumdiagrammms [26]

Vorteile der Baum Visualisierung:

1. Hierarchische Strukturen: Wenn die Daten auf der Weltkarte in hierarchischen Strukturen organisiert sind, wie z.B. Nationen, Regionen oder Truppen, kann die Darstellung in einer Baumstruktur sinnvoll sein. Dadurch kann der Benutzer Relationen zwischen verschiedenen Knotentypen schnell erkennen.
2. Übersichtlichkeit: Eine Weltkarte kann sehr komplex sein, insbesondere wenn sie viele Daten enthält. Durch die Darstellung in einer Baumstruktur kann der Benutzer schnell einen Überblick über die Daten erhalten und Zusammenhänge erkennen. Dies kann insbesondere bei großen Datensätzen von Vorteil sein.
3. Effizienz: Eine Baumstruktur kann auch bei der Datenanalyse und -verarbeitung effizient sein, da der Benutzer schnell zwischen den verschiedenen Ebenen navigieren und die Daten filtern kann, die für ihn relevant sind.

Nachteile der Baum Visualisierung:

1. Platzbedarf: Baumvisualisierungen können viel Platz auf einem Bildschirm einnehmen, insbesondere wenn der Baum sehr tief ist oder viele Knoten enthält.
2. Begrenzte Hierarchie: Baumvisualisierungen eignen sich am besten für die Darstellung von hierarchischen Beziehungen, bei denen es eine klare Ordnung der Elemente gibt. In anderen Fällen können sie jedoch weniger effektiv sein.

8.3.1. Einsatzgebiet des Baum Modells

Anwendungsbereiche des Baummodells mit Bezug zu historischen Ereignissen sind:

Historische Zusammenhänge: Das Baummodell kann dazu beitragen, komplexe Zusammenhänge und Beziehungen in historischen Ereignissen zu präsentieren.

Erklärung von historischen Entwicklungen: Das Baummodell kann auch dazu beitragen, historische Entwicklungen und Veränderungen zu erklären. Beispielsweise kann ein Entscheidungsbaum erstellt werden, um politische Entscheidungen und Konflikte zu erklären.

Vergleich historischer Ereignisse: Das Baummodell kann auch verwendet werden, um historische Ereignisse zu vergleichen und zu analysieren.

Darstellung historischer Entscheidungsprozesse: Das Baummodell kann verwendet werden, um Entscheidungsprozesse historischer Ereignisse darzustellen und zu visualisieren

Veranschaulichung von historischen Veränderungen: Das Baummodell kann auch dazu beitragen, historische Veränderungen und Entwicklungen zu veranschaulichen.

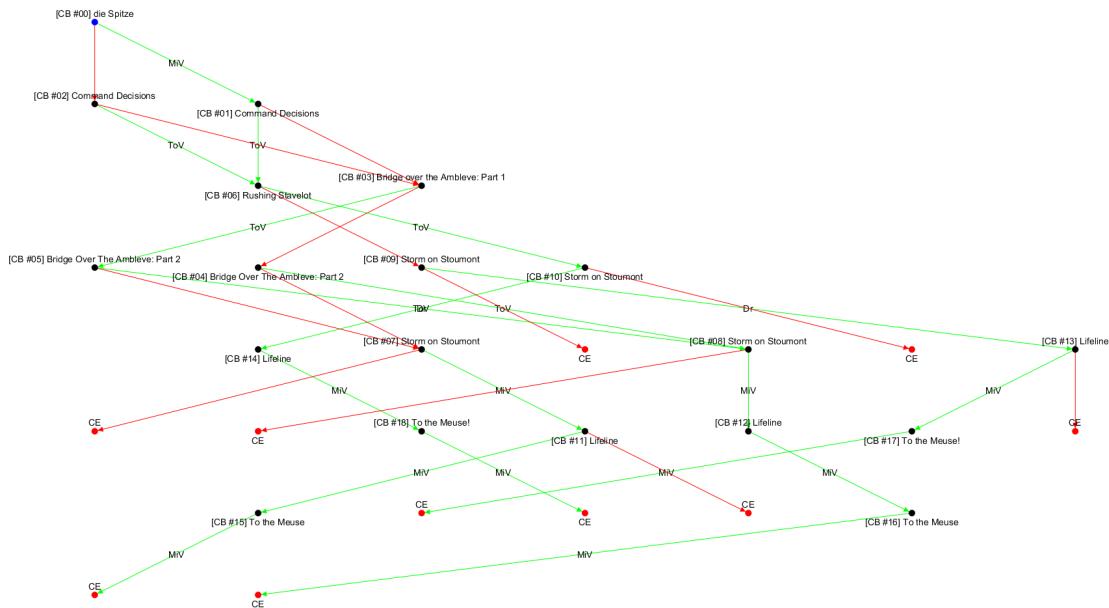


Abbildung 8.6.: Entscheidungsbaum diverser Strategien der Kampfgruppe Peiper [27]

8.4. Konzept Entscheidung

8.4.1. Anforderungen

Die Reduzierung auf

- klare Verständlichkeit der Daten
- intuitive Interaktion
- Abbildung von komplexen Daten
- Reduzierung der benötigten Systemleistung auf ein Minimum
- Bezug zu bereits bekannten Modellen

sind für die Implementierung des Prototypen obligatorisch.

8.4.2. Entscheidungsfindung

Ein kartographisches Modell eignet sich am besten für die Visualisierung des Prototypen aus verschiedenen Gründen:

1. Intuitiv: Eine kartographische Darstellung ist für einen historischen Kontext das intuitivste Modell, da es eine visuelle Repräsentation von geographischen Informationen bietet, die leicht verständlich ist.
2. Detailgenauigkeit: Eine kartographische Darstellung kann sehr detailliert sein und eine Vielzahl von Informationen auf kleinem Raum vermitteln. Dies ermöglicht es, komplexe Zusammenhänge und Beziehungen zwischen verschiedenen Elementen aufzuzeigen.
3. Übersichtlichkeit: Kartographische Modelle sind in der Regel übersichtlicher als Graph-Darstellungen oder Baum Visualisierungen

Die Darstellung mithilfe eines Graphen ist im Kontext historischer Ereignisse nicht von Vorteil, da es schwierig ist, räumliche Informationen, historische Kontexte und Ereignisse in einem einzelnen Graphen zusammenzufassen.

Ein Baum-Modell ist weniger geeignet für die Darstellung einer Karte in einer historischen Simulation, da es nicht in der Lage ist, komplexe räumliche Informationen, historische Kontexte und zeitliche Veränderungen darzustellen.

9. Systemarchitektur

9.1. Komponentenübersicht

Der Prototyp besteht aus folgenden Komponenten:

- Blazor Server App
- Identity Provider
- RabbitMq Message Broker
- MySql Datenbank

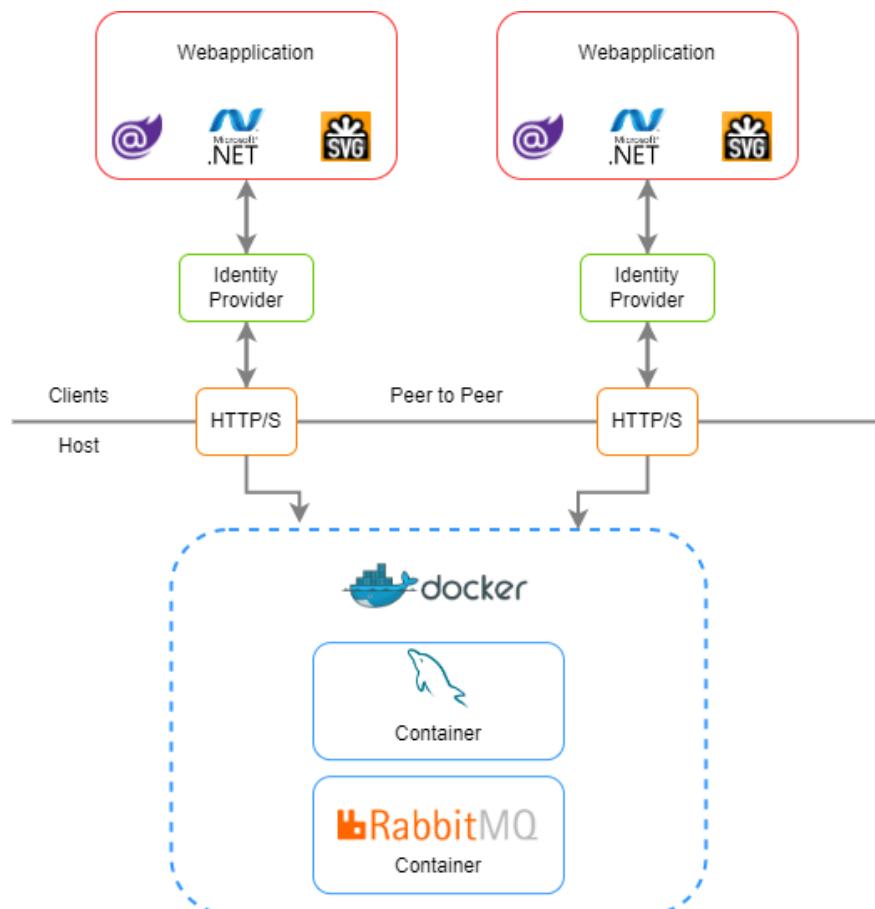


Abbildung 9.1.: Komponentenübersicht

9.1.1. Blazor Server App

Die Implementierung des Prototypen basiert auf folgenden Schichten:

Die Model-Schicht ist für das Datenbankschema zuständig. Sie beinhaltet die Datenbankanbindung Konfiguration zu dem MySql Container.

Die Domain-Schicht ist für den Datenbankzugriff zuständig. Um dies zu ermöglichen wird das Repository-Pattern mit den CRUD Methoden umgesetzt.

Die View-Schicht ist für die Präsentation der Anwendung verantwortlich. Hier werden die Benutzeroberfläche und die Benutzerinteraktionen implementiert.

Darüber hinaus ermöglicht eine zusätzliche Schicht die Kommunikation mit dem RabbitMq Container und steuert sämtliche Events welche für die Synchronisierung zwischen den Clients erforderlich ist.

9.1.2. Identity Provider

Um die Sicherheit des Systems zu gewährleisten, wird mithilfe des „ASP.NET Core Identity“ Frameworks eine Authentifizierungsstufe implementiert. Diese sorgt dafür, dass bei mehreren Zugriffen auf Daten sichergestellt wird, dass der jeweilige Benutzer auch die Rechte hat, diese zu laden.

9.1.3. RabbitMq Message Broker

Die Synchronisierung der Daten ist ein zentraler Punkt, welcher beachtet werden muss. Hierbei wird auf das Designpattern des „Eventbus“ gesetzt. Dieser stellt eine zentrale Schnittstelle zwischen Client-Rechnern da, welche untereinander synchronisiert werden müssen. Dafür wird die Software RabbitMq verwendet.

Jeder Client implementiert einen Background Service, welcher die Aufgabe hat auf eine Queue des RabbitMq Service zu hören und Events an einen Eventprozessor weiterzuleiten. Dieser verarbeitet das Event und fordert von der MySql Datenbank die neuen Daten an. Ist dies geschehen, gibt er dem System den Befehl die Oberfläche zu aktualisieren.

9.1.4. MySql Datenbank

Um Daten zu präservieren, wird eine Standardkonfiguration eines MySql Docker Container verwendet.

9.2. Kommunikation

Dem Prototyp liegen folgende grundlegende Kommunikationsabläufe zugrunde. Der Prozess der Benutzerinteraktion auf der Simulationsoberfläche läuft folgendermaßen ab:

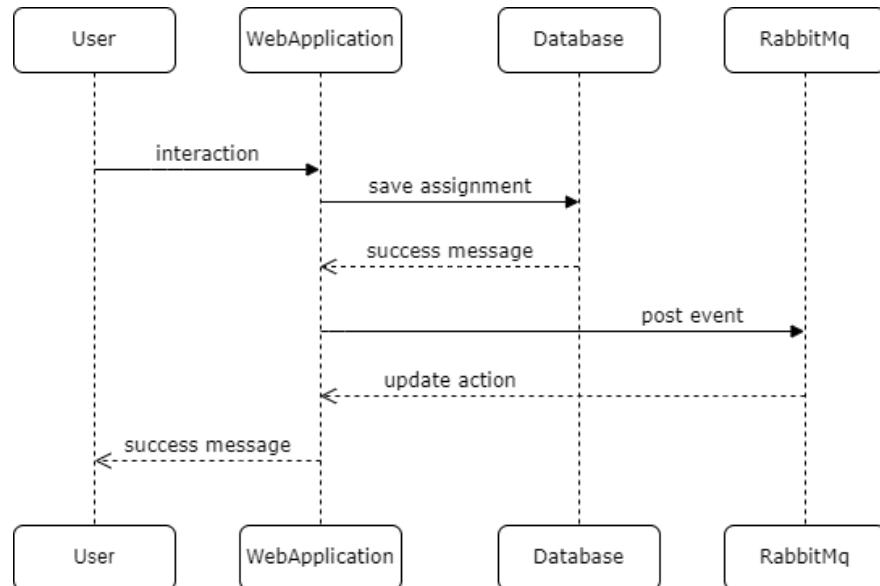


Abbildung 9.2.: Interaktion auf der Simulationsoberfläche

Die Antwortverarbeitung gestaltet sich folgendermaßen:

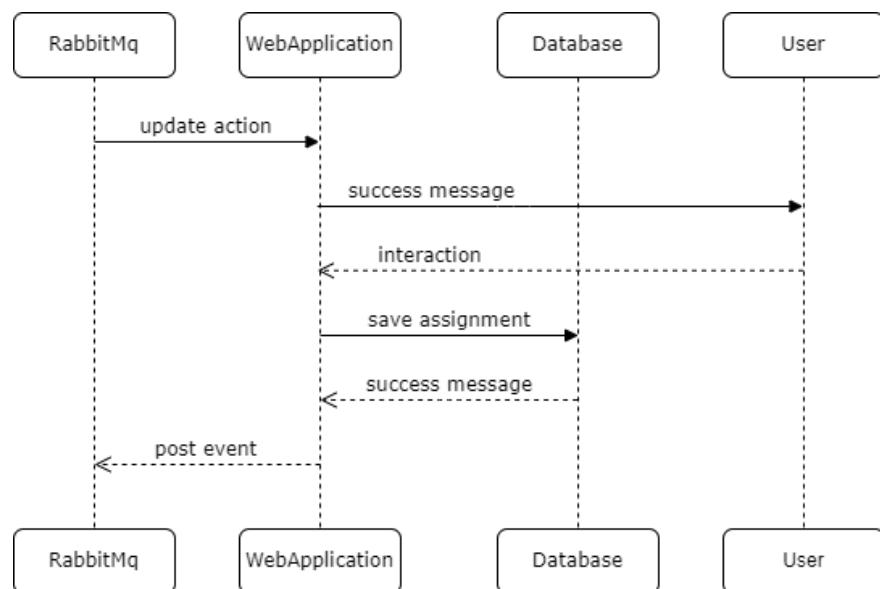


Abbildung 9.3.: Empfangen einer Interaktion

9.3. Technologien

9.3.1. .NET Core

Für die Implementierung der Webapplikation wird die Programmiersprache C# verwendet. Das verwendete Framework .NET Core von Microsoft bietet folgende Vorteile:

- Kompatibilität mit RabbitMq und Docker
- einfache Anbindung der MySQL Datenbank
- leichte Implementierung einer Modellschicht durch EF
- Unterstützung von vielen Erweiterungspaketen

Alternativ dazu wurde JavaScript mit dem React Framework in Betracht gezogen, welches jedoch durch die Komplexität im Bezug zur Einbindung der RabbitMq Komponente nicht weiter in Betracht gezogen wurde.

9.3.2. Blazor Server App

Blazor ist ein Web-Framework zur Erstellung von Web-UI-Komponenten (Razor-Komponenten), welche auf verschiedene Weise gehostet werden können.

Mit der Blazor Server Technologie wird die App auf dem Server von einer ASP.NET Core-App aus ausgeführt. UI-Updates, Ereignisse und JavaScript-Aufrufe werden über eine SignalR-Verbindung unter Verwendung des WebSockets-Protokolls abgewickelt.

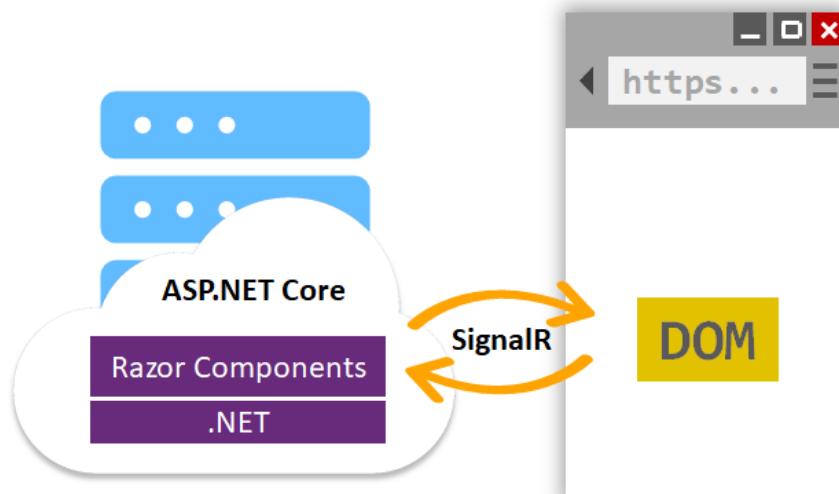


Abbildung 9.4.: Aufbau einer Blazor Server App [28]

Diese Technologie bringt folgende Vorteile mit sich:

- die Downloadgröße ist deutlich kleiner und die Ladezeit der App wesentlich kürzer
- Blazor Server-Apps funktionieren auch mit Browsern, die WebAssembly nicht unterstützen
- .NET Tools wie Debuggen funktioniert erwartungsgemäß

Nachteile sind folgende:

- ständige Verbindung zum Server wird benötigt
- höhere Latenzen
- serverlose Bereitstellung ist nicht möglich

Es gibt eine Vielzahl an Alternativen für die Blazor Server App.

Wie zum Beispiel:

- React
- Angular
- Vue

Da diese Frameworks allerdings auf eine zusätzliche API-Schnittstelle setzen, welche implementiert werden muss, entschieden wir uns hierbei für die Blazor Server Technologie.

9.3.3. MySql

Für die Persistierung der Daten wird eine MySql Datenbank eingesetzt.

Vorteile einer relationalen Datenbank gegenüber eines dokumentbasierten Modells sind:

- Verhinderung von duplizierten Daten
- Gewährleistung von Datenintegrität
- Gewährleistung von Datenkonsistenz
- Flexibilität bei Datenänderungen

Durch diese Vorteile wird eine Systemumgebung geschaffen, welche strukturiert abgefragt werden kann. Zusätzlich bietet das Modell eine effiziente Speicherung der Daten.

In einer MySql Datenbank werden Daten in mehreren Tabellen gespeichert welche mittels Schlüssel in Beziehung gesetzt werden. Alle Tabellen bestehen aus Spalten und Zeilen wobei jeder Spalte ein fixer Datentyp zugeordnet wird.

	ID	NAME	TREASURY	PLAYER_ID	COLOR	TYPE
1	1	Soviet_Union	24	35c0c9db-10cc-45cf-a07e-2cbabb1910ee	#ba8772	SovietUnion
2	2	Germany	41	35c0c9db-10cc-45cf-a07e-2cbabb1910ee	#8ea39e	Germany
3	3	United_Kingdom	31	35c0c9db-10cc-45cf-a07e-2cbabb1910ee	#b59b68	UnitedKingdom
4	4	Japan	30	35c0c9db-10cc-45cf-a07e-2cbabb1910ee	#d29151	Japan
5	5	United_States	42	35c0c9db-10cc-45cf-a07e-2cbabb1910ee	#97a95f	UnitedStates
6	6	Neutral	0	35c0c9db-10cc-45cf-a07e-2cbabb1910ee	#4f4d49	Neutral

Abbildung 9.5.: Beispiel einer MySql Tabelle

9.3.4. RabbitMq

Zur Synchronisierung der Clients wird ein Event Bus eingesetzt. Aus technischer Sicht wird dabei auf RMQ zugegriffen. Dieser empfängt Events und benachrichtigt alle Clients, dass eine Änderung vorgenommen wurde.

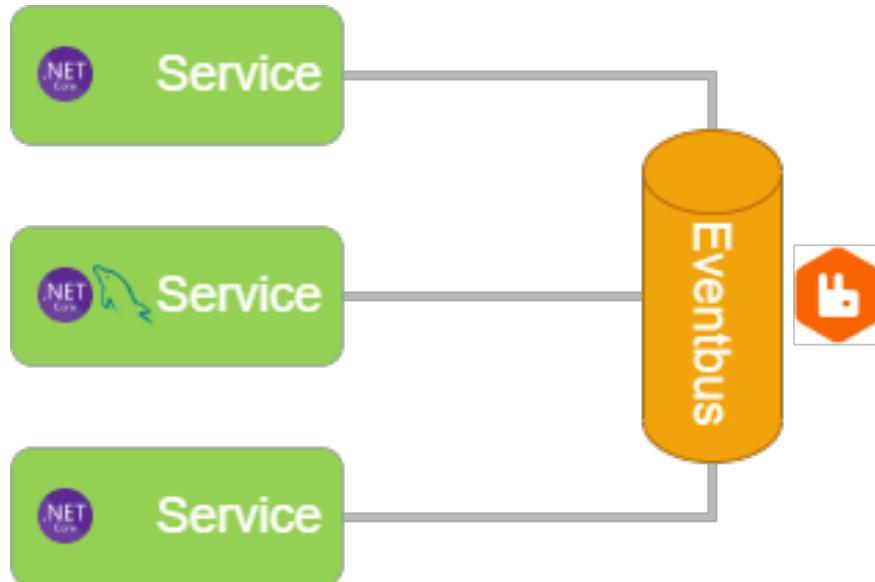


Abbildung 9.6.: Kommunikation zwischen Clients über den Event Bus

9.3.5. SVG

SVG wird eingesetzt, um eine interaktive Oberfläche bereitzustellen, werden Scalable Vector Graphics benutzt. Diese werden mathematisch berechnet und stellen so sicher, das jedes System dieselbe Darstellung aufweist.

```
1  <svg xmlns="http://www.w3.org/2000/svg" class="path-arrow">
2      <defs>
3          <marker id="arrow" viewBox="0 0 10 10" refX="5" refY="5"
4              markerWidth="4" markerHeight="4" orient="auto"
5              fill="darkcyan">
6              <path d="M 0 0 L 10 5 L 0 10 z"/>
7          </marker>
8      </defs>
9      <path d="@path" stroke="darkcyan" fill="none"
10         marker-end="url(#arrow)" stroke-width="4"/>
11  </svg>
```

Quellcode 9.1: Beispiel einer SVG Implementation

9.3.6. Panzoom

BlazorPanzoom ist eine Bibliothek für Blazor, welche sämtliche "Panzoom JavaScript-Bibliothek" Funktionen implementiert. Sie bietet eine einfache Möglichkeit, das Bewegen und Zoomen von Webkomponenten/Elementen mithilfe von CSS-Transformationen zu ermöglichen.

```
1  <div class="panzoom-parent" style="height: 100vh; @Style">
2      <Panzoom @ref="_panzoom" PanzoomOptions="@_panzoomOptions"
3          WheelMode="WheelMode.Custom" OnWheel="OnWheel">
4              // Map render
5          </Panzoom>
6      </div>
7
```

Quellcode 9.2: Einbinden eines Panzoom Containers

9.3.7. Docker

Docker bietet eine einfache Möglichkeit zur Containerisierung von Softwarekomponenten. Damit ist eine einfache und schnelle Installation gewährleistet. Durch die Isolierung der einzelnen Service Schichten ist eine Unabhängigkeit der Komponenten gegeben.

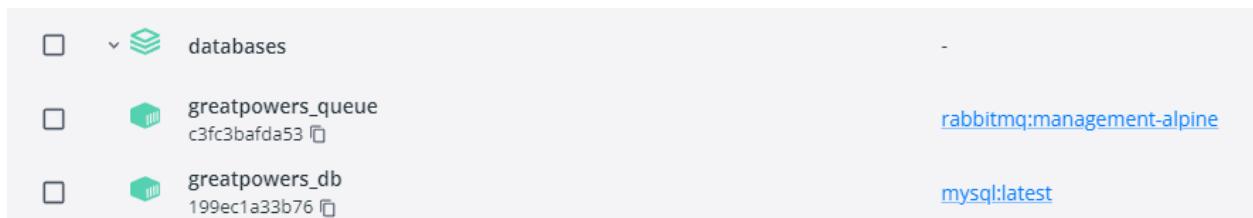


Abbildung 9.7.: Aufbau der Docker Architektur

10. Komponentenbeschreibung Great-Powers

10.1. Komponentenbeschreibung Program

Die "Program.cs" Datei ist eine zentrale Datei in einer ASP.NET Core-Anwendung, da sie die Einstiegspunkte für die Anwendung definiert und konfiguriert.

10.1.1. Containerisierung

Um sicherzustellen, dass die Anwendung ordnungsgemäß funktioniert und nicht fehlschlägt, muss ein neues Prozessobjekt angelegt werden, um den Befehl "docker-compose" in der Windows-Eingabeaufforderung ausführen zu können.

Bereits laufende Container werden mithilfe von "docker-compose down" gestoppt und dann mit "docker-compose up -d -build" neu gestartet. Die -d Flag sorgt dafür, dass die Container im Hintergrund ausgeführt werden.

Der Pfad zum Verzeichnis, in dem sich die Datenbankdateien befinden, wird zuvor durch die Verwendung von "Directory.GetParent" und "Environment.CurrentDirectory" ermittelt.

```
1  var databasePath =
2      Directory.GetParent(Environment.CurrentDirectory)
3          .Split("Great-Powers-Thesis")[0] +
4          "\\Great-Powers-Thesis\\Databases";
5  var process = new Process();
6  var startInfo = new ProcessStartInfo{
7      WindowStyle = ProcessWindowStyle.Hidden,
8      FileName = "cmd.exe"
9  };
10 process.StartInfo = startInfo;
11 process.StartInfo.Arguments = $"{"/c cd {databasePath} &&
12 docker-compose down && docker-compose up -d --build"};
13 process.Start();
14 process.WaitForExit();
```

Quellcode 10.1: Starten der Dockerdienste

10.1.2. WebApplication Builder

In der Datei "Program.cs" wird die "WebApplicationKlasse konfiguriert, welche die Verarbeitung von HTTP-Anforderungen in der ASP.NET Core-Anwendung handhabt.

```
1  var builder = WebApplication.CreateBuilder(args);
2  builder.Services.AddDbContext<GreatPowersDbContext>(
3      options => {
4          options.UseMySql(
5
6              builder.Configuration.GetConnectionString("DefaultConnection"),
7                  new MySqlServerVersion(new Version(8, 0, 27)),
8                  retry => retry.EnableRetryOnFailure()
9              );
10             options.UseLoggerFactory(new NullLoggerFactory());
11         });
12 
```

Quellcode 10.2: Konfiguration des WebApplication Builder

ASP.NET Core Identity ist ein Framework zur Authentifizierung und Autorisierung von Benutzern in ASP.NET Core Webanwendungen. Es enthält eine Reihe von Funktionen zur Verwaltung von Benutzerkonten, wie z.B. Registrierung, Anmeldung, Zurücksetzen von Passwörtern und Verwaltung von Benutzerrollen.

Diese Funktionen können durch eine Registrierung der User Klasse als AddDefaultIdentity-User genutzt werden.

10.1.3. ASP.NET Core Identity Framework

```
1  builder.Services.AddDefaultIdentity<User>
2      (options => options.SignIn.RequireConfirmedAccount = false)
3          .AddEntityFrameworkStores<GreatPowersDbContext>();
4
5  app.UseAuthentication();
6  app.UseAuthorization();
7 
```

Quellcode 10.3: Konfiguration des ASP.NET Core Identity Framework

10.1.4. MudBlazor Service

MudBlazor ist eine Open-Source-Bibliothek für das Blazor-Webframework. Sie bietet eine Vielzahl von UI-Komponenten und -Funktionen, die Entwicklern helfen, ansprechende und benutzerfreundliche Webanwendungen zu erstellen. MudBlazor enthält eine große Auswahl an Komponenten, darunter Schaltflächen, Formulare, Tabellen oder Diagramme und unterstützt eine Vielzahl von Anpassungsmöglichkeiten.

Um diesen Service zu integrieren, muss dies als Service registriert werden.

```

1  builder.Services.AddMudServices(config => {
2      config.SnackbarConfiguration.PositionClass =
3          Defaults.Classes.Position.BottomEnd;
4      config.SnackbarConfiguration.PreventDuplicates = true;
5      config.SnackbarConfiguration.NewestOnTop = false;
6      config.SnackbarConfiguration.ShowCloseIcon = true;
7      config.SnackbarConfiguration.VisibleStateDuration = 10000;
8      config.SnackbarConfiguration.HideTransitionDuration = 500;
9      config.SnackbarConfiguration.ShowTransitionDuration = 500;
10     config.SnackbarConfiguration.SnackbarVariant = Variant.Filled;
11 });

```

Quellcode 10.4: Konfiguration des MudBlazor Service

10.1.5. Dependency Injection

Dependency Injection (DI) ist ein Entwurfsmuster, das zur Entkopplung von Komponenten in einer Anwendung verwendet wird. Es ermöglicht, Abhängigkeiten zwischen Objekten zur Laufzeit aufzulösen, anstatt sie zur Kompilierungszeit festzulegen. Dadurch wird die Flexibilität, Erweiterbarkeit und Testbarkeit von Anwendungen verbessert. Um diese Funktionalität in .NET 7.0 nutzen zu können, muss Folgendes konfiguriert werden:

```

1  builder.Services.AddScoped<IService, Service>();
2  builder.Services.AddScoped<Service>();
3

```

Quellcode 10.5: Registrierung einer Systemverwalteten Klasse

Zusammengefasst ist die "Program.cs"-Datei eine wichtige Komponente einer ASP.NET Core-Anwendung, da sie die grundlegende Konfiguration und den Einstiegspunkt für die Anwendung definiert.

10.2. Komponentenbeschreibung View

10.2.1. Menü Subkomponent

Die Menü Komponente bildet die Oberfläche zur Navigationslogik. Der Zweck dieser Komponente ist es, die Navigation und die Einstellungen dem Nutzer zu präsentieren.

Ausgangspunkt dieser Komponente ist das *Main Menu*, welches sich in zwei Untermenüs aufteilt. Je nach Menüs ist es möglich, eine neue Simulation zu erstellen, zu laden oder einer bereits bestehenden Simulation beizutreten.

Hierbei ist zu beachten, dass zu jedem gegebenen Zeitpunkt eine valide SessionId des ASP.NET Identity Framework gefordert ist. Sollte diese Id nicht im Browser Cache vorliegen, wird der Benutzer auf die entsprechende Identity Seite weitergeleitet.

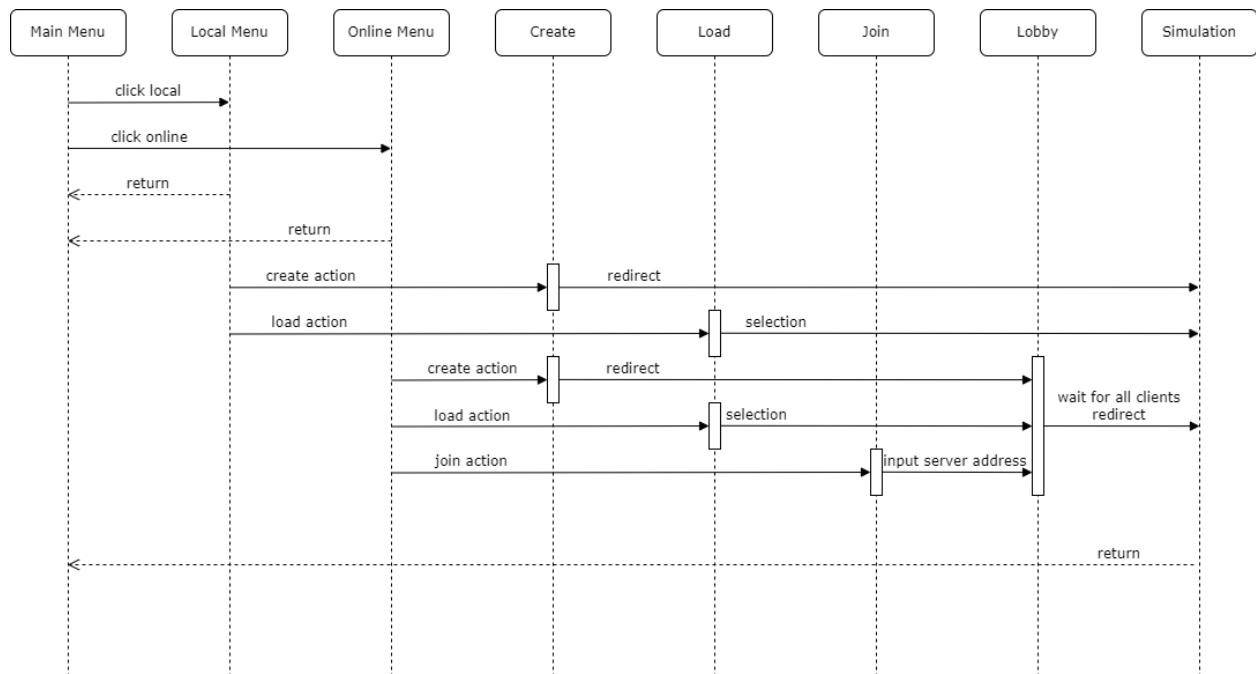


Abbildung 10.1.: Sequenzdiagramm der Menüinteraktion

10.2.2. Simulations Subkomponentent

Die Simulationskomponente bildet die Oberfläche der Simulationslogik. Der Zweck dieser Komponente ist es, die implementierte Logik der Schichten Model, Domain und Eventbus graphisch darzustellen.

Zentraler Ausgangspunkt dieser Komponente ist die Simulation, welche für die Darstellung der *Informationsleiste*, *Karte* und des *Drawers* zuständig ist.

Je nach Phase werden der *Drawer Komponente* spezifische Daten übergeben, um die passenden Informationen darstellen zu können.

Die *Karten Komponente* ist für die Darstellung der Truppenleiste und des *Panzoom Containers* zuständig.

Der *Panzoom Container* gliedert sich wiederum in folgende Unterkomponenten auf:

- Industriekomplexe
- Wirtschaftsstärke
- Länder
- Kanäle
- Bewegungspfeile
- Truppen

Die Darstellung der Subtruppen wird von der Truppenkomponente übernommen.

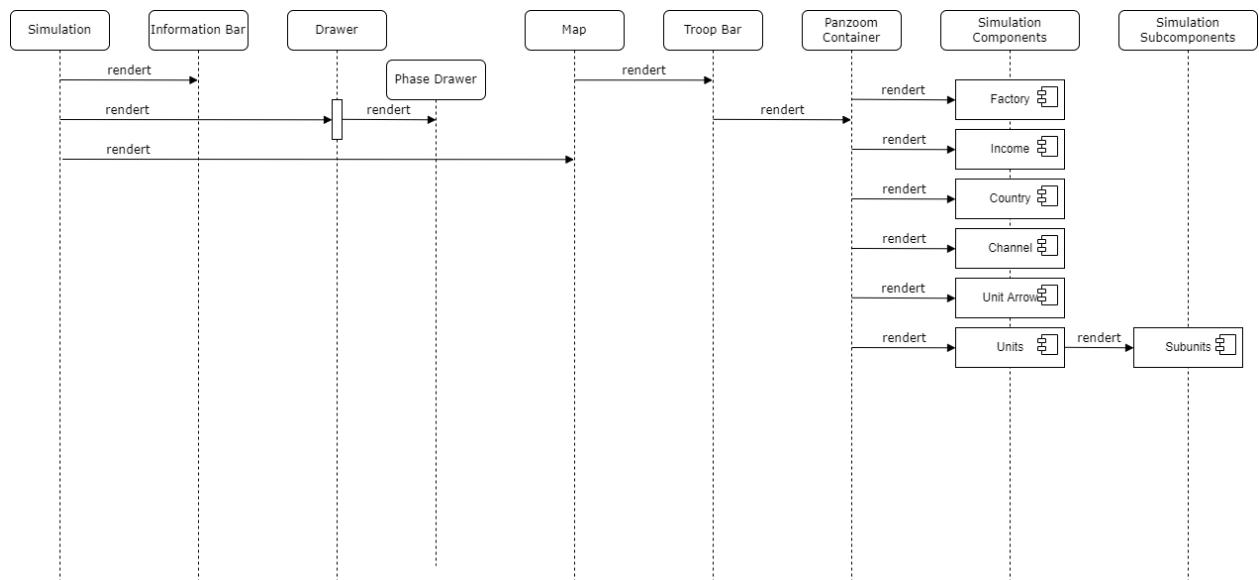


Abbildung 10.2.: Render Sequenz

10.3. Komponentenbeschreibung Model

Die Klassen der Model-Schicht spiegeln die Struktur der Datenbank wieder und dienen so dazu, die gespeicherten Daten als sogenannte Entitäten zu repräsentieren. Hier werden somit die verschiedenen Simulationskomponenten in der Form von C# Code abgebildet und auch deren Verhalten für die Simulationslogik definiert.

10.3.1. Datenbankstruktur

Die Simulationskomponenten werden in der Datenbank durch die folgende Struktur dargestellt.

Siehe Anhang Grafik A.1

10.3.2. Übersicht der Entitäten

In folgendem Klassendiagramm wird eine Übersicht über die verschiedenen Entitäten der Model-Schicht verschaffen.

Siehe Anhang Grafik A.2

10.3.2.1. User

Die User Entität stellt die Benutzer der Simulation dar und wird somit zur Identifikation der Simulanten verwendet.

10.3.2.2. SessionInfo

In der SessionInfo Entität werden die Grunddaten der Simulation verwaltet, darunter z.B. die aktuelle Phase und Runde, die Bedingungen für die Vollendung der Simulation oder welche Nation momentan Simuliert wird.

10.3.2.3. Nation

Die Nation Entität repräsentiert die Nationen der Simulation mit ihren dazugehörigen Daten wie Name, zuständiger User, momentanes Einkommen aus deren besetzten Regionen oder auch deren angespartes Vermögen. In der Allies Entität werden die Bündnisse zwischen den Nationen festgehalten.

10.3.2.4. AUnit

Die AUnit Entität stellt die Gemeinheiten zwischen den verschiedenen Truppen wie Infanterie oder Kreuzer dar und stellt auch die Logischen Methoden für die Kalkulation der Pfade bereit, welche Einheiten zur Fortbewegung nutzen. Solche geteilten Attribute wären beispielsweise die restliche Bewegungsreichweite der Einheit oder die Nation, zu welcher die Einheit zugehörig ist. Die unterschiedlichen Einheitentypen werden anschließend nach ihren spezifischen Eigenschaften weiter in 3 Gruppen unterteilt

- Land Truppen wie Infanterie Kompanien oder Panzerdivisionen
- Luft Truppen, darunter Jäger und Bomber Staffeln
- Aquatische Truppen wie U-Boote oder Schlachtschiffe

Diese Gruppen werden jeweils durch ihre eigenen Entitäten ALandUnit, APlane und AShip weiter repräsentiert, wobei hier wieder jegliche Eigenschaften und Logik definiert wird, welche von allen Einheiten der jeweiligen Gruppe geteilt werden. Darunter fällt nun etwa die aktuelle Position, da Landeinheiten nur am Land und aquatische Einheiten nur auf dem Wasser stationiert sein können, sowie auch die Logik für die Ausführung einer geplanten Bewegung. Diese 3 Einheitentypen umfassen schließlich die einzelnen, konkreten Einheiten. Die verschiedenen Einheiten, welche in dieser Simulation vorhanden sind setzen sich für Landeinheiten aus

- Infanterie-
- gepanzerte Fahrzeug-
- Artillerie-
- und Flak-Divisionen,

für Lufteinheiten aus

- Jäger-
- und Bomber-Staffeln

und für aquatische Einheiten aus

- U-Boot-
- Zerstörer-
- Kreuzer-
- Flugzeugträger-
- Transporter-
- und Schlachtschiff-Flotten

zusammen und haben letztlich durch ihre eigenen Entitäten ihre spezifischen Werte wie Trefferpunkte oder Bewegungsreichweite sowie die Logik für ihre spezifischen Regeln bei der Fortbewegung, darunter z.B. das Blitzmanöver von Panzerdivisionen oder das Untertauchen von U-Booten. Auch Industriekomplexe werden als Einheit behandelt, da sie Attribute wie Position und Kosten, sowie auch die Eigenschaft, in der Kaufphase produziert zu werden mit den anderen Einheiten teilen.

10.3.2.5. ARegion

In der ARegion Entität werden die unterschiedlichen geographischen Regionen der Simulation verwaltet, wobei sich hier nur der Name der Region und generelle Logik zur Bestimmung von in der Region stationierten, bestimmte Bedingungen erfüllenden Einheiten befinden. Anschließend werden Regionen auf Land- und Seeregionen aufgeteilt, wobei die Eigenschaften dieser in ihren jeweils eigenen Entitäten weiter definiert werden, so wird beispielsweise in Landregionen die Nation, welche sich im Besitz der Region befindet und die Wirtschaftsstärke der Region gespeichert. Die Capitals Entität repräsentiert eine Hauptstadt von großer

Bedeutung, die Eroberung welcher das Ziel der Nationen ist und hat die Landregion gespeichert, in welcher sie sich befindet. Weiters wird in der Neighbours Entität festgehalten, welche Regionen miteinander benachbart sind, was als Grundlage für die Bewegung der Einheiten dient und in der CanalOwners Entität wird letztlich noch vermerkt, welche Landregionen jeweils den Panama- und Suezkanal kontrollieren.

10.3.2.6. Battle

Die Battle Entität verwaltet die Daten und Logik zur Simulation eines Gefechtes. Zu den verwalteten Daten gehören unter anderem die teilnehmenden Truppen, aufgeteilt auf Angreifer und Verteidiger, die aktuelle Phase und Runde des Gefechts oder die Nation, welche sich momentan in der Offensive befindet. Die Logik umfasst das Würfeln der Truppen der aktuellen Nation, das zuteilen der erzielten Treffer oder auch das Abhandeln der entstandenen Verluste.

10.3.3. DbContext

Ebenfalls in der Model-Schicht ist der DbContext, eine Klasse welche auf der DbContextBase Klasse des Pomelo Entity Frameworks aufbaut und als Schnittstelle zwischen der wirklichen Datenbank und den Entitätsklassen dient. Hier werden die verschiedenen Entitäten vermerkt, die Verbindungen zwischen ihnen festgelegt und auch die Datenbank mit ihren initialen Daten befüllt.

10.4. Komponentenbeschreibung Domain

Die Klassen der Domain-Schicht dienen zur Steuerung des Datenbank-Zugriffs, dabei gibt es für jede Entität eine eigene Repository Klasse, welche ihre allgemeinen Methoden von den ARepository oder ACreatableRepository Klassen erben.

10.4.1. ARepository

Im ARepository werden Methoden zum Lesen und Verändern von Entitäten definiert, dieses wird durch seine Einbindung in das ACreatableRepository von allen Entitäten verwendet. Genaueres ist aus folgendem Codeabschnitt abzulesen

```
1  public abstract class ARepository<TEntity> : IRepository<TEntity>
2      where TEntity : class{
3          protected readonly GreatPowersDbContext _context;
4          protected readonly DbSet<TEntity> _set;
5
6          public ARepository(GreatPowersDbContext context){
7              _context = context;
8              _set = _context.Set<TEntity>();
9          }
10
11         public virtual async Task<TEntity?> ReadAsync(int id) => await
12             _set.FindAsync(id);
13
14         public virtual async Task<List<TEntity>>
15             ReadAsync(Expression<Func<TEntity, bool>> filter) =>
16             await _set.Where(filter).ToListAsync();
17
18         public virtual async Task<List<TEntity>> ReadAllAsync() => await
19             _set.ToListAsync();
20
21         public virtual async Task<List<TEntity>> ReadAsync(int start, int
22             count) =>
23             await _set.Skip(start).Take(count).ToListAsync();
24
25     }
26
```

Quellcode 10.6: Inhalt des ARepositories

10.4.2. ACreatableRepository

Im ACreatableRepository werden sowohl Methoden zum Erstellen und Entfernen von Entitäten definiert als auch die Methoden des ARepositories durch Vererbung übernommen. Verwendung findet dieses Repository in den Repositories der verschiedenen Einheiten. Genaueres ist aus folgendem Codeabschnitt abzulesen

```
1  public abstract class ACreatableRepository<TEntity> :  
2      ARepository<TEntity>, ICreatableRepository<TEntity> where TEntity :  
3          class  
4  {  
5      public ACreatableRepository(GreatPowersDbContext context) :  
6          base(context){  
7          }  
8  
9      public async Task<TEntity> CreateAsync(TEntity entity){  
10         _context.ChangeTracker.Clear();  
11         _set.Add(entity);  
12         await _context.SaveChangesAsync();  
13         return entity;  
14     }  
15  
16     public async Task DeleteAsync(TEntity entity){  
17         _context.ChangeTracker.Clear();  
18         _set.Remove(entity);  
19         await _context.SaveChangesAsync();  
20     }  
21  
22     public async Task DeleteAsync(List<TEntity> entities){  
23         _set.RemoveRange(entities);  
24         await _context.SaveChangesAsync();  
25     }  
26 }
```

Quellcode 10.7: Inhalt des ACreatableRepository

10.4.3. SessionInfo-Repository

Für die SessionInfo Entität gibt es ein eigenes Repository welches nicht die Methoden von ARepository oder ACreatableRepository erbt, sondern seine eigenen Methoden definiert, da es nur eine Instanz dieser Entität gibt. Im folgendem Codeausschnitt ist genaueres ersichtlich

```
1  public class SessionInfoRepository{
2      private readonly GreatPowersDbContext _context;
3      protected readonly DbSet<SessionInfo> _set;
4
5      public SessionInfoRepository(GreatPowersDbContext context)
6      {
7          _context = context;
8          _set = _context.Set<SessionInfo>();
9      }
10
11     public async Task<SessionInfo?> ReadAsync(){
12         return await _set.FirstOrDefaultAsync();
13     }
14
15     public async Task UpdateAsync(SessionInfo entity)
16     {
17         _context.ChangeTracker.Clear();
18         _set.Update(entity);
19         await _context.SaveChangesAsync();
20     }
21
22 }
```

Quellcode 10.8: Inhalt des SessionInfo-Repository

10.5. Komponentenbeschreibung Eventbus

Die Klassen des Eventbusses dienen zur Kommunikation zwischen Benutzern der Simulationssoftware, auch wenn deren Anwendungen in verschiedenen Netzwerken operieren, wobei hier immer dann eine Nachricht versandt wird, wenn sich der Zustand der Simulation ändert. Der Eventbus besteht aus den folgenden Komponenten

10.5.1. Event Publisher

Der Event Publisher schreibt die Benachrichtigungen in den RabbitMQ Exchange, daraufhin werden sie in die Queues der Benutzer verteilt. Konfiguriert wird der Event Publisher folgendermaßen

```
1  public class EventPublisher : IEventPublisher {  
2  
3      private readonly IConfiguration _configuration;  
4      private readonly string _exchangeName;  
5      private readonly IModel _channel;  
6      private readonly IConnection _connection;  
7  
8      public EventPublisher(IConfiguration configuration) {  
9          _configuration = configuration;  
10  
11         var factory = new ConnectionFactory() {  
12             HostName = AppSettings.IpAddress,  
13             Port = int.Parse(AppSettings.RabbitPort),  
14             Password = "greatpowers",  
15             UserName = "greatpowers",  
16             VirtualHost = "greatpowers"  
17         };  
18         try {  
19             _connection = factory.CreateConnection();  
20             _channel = _connection.CreateModel();  
21             _connection.ConnectionShutdown += ShutDownMessageBroker;  
22             _exchangeName = _configuration["EventBusExchange"];  
23         }  
24         catch (Exception e) {  
25             Console.WriteLine(e);  
26             throw;  
27         }  
28     }  
29  
30     private void ShutDownMessageBroker(object? sender,  
31     ShutdownEventArgs e) => Console.WriteLine("Message broker  
connection shut down");  
32 }
```

Quellcode 10.9: Konfiguration des Event Publishers

Hier werden alle Daten festgelegt, die zum Veröffentlichen von Nachrichten benötigt werden.

Die Methode zum Veröffentlichen von Benachrichtigungen sieht wiederum so aus

```
1  public void Publish(string message) {  
2      var body = Encoding.UTF8.GetBytes(message);  
3  
4      _channel.BasicPublish(_exchangeName, "", null, body);  
5  }  
6
```

Quellcode 10.10: Publish Methode des Event Publishers

Diese wird in der View Komponente an jeglichen Stellen aufgerufen, an welchen sich der Status der Simulation ändert. Dies kann dann wie folgt aussehen

```
1  _EventPublisher.Publish(JsonSerializer.Serialize(new  
2  StateHasChangedEvent()));  
3
```

Quellcode 10.11: Aufruf der Publish Methode

10.5.2. Event Subscriber

Der Event Subscriber ist ein vom System verwalteter Hintergrund Prozess, welcher dauerhaft mit der Anwendung läuft und darauf wartet, dass eine Nachricht in die Queue des Benutzers geschrieben wird, woraufhin der Event Subscriber die Benachrichtigung dann an den Event Processor zur Verarbeitung weiterleitet. Um als Background Service registriert zu werden, muss man nun folgenden Code im Program.cs einfügen

```
1 var builder = WebApplication.CreateBuilder(args);  
2 ...  
3 ...  
4 builder.Services.AddHostedService<EventSubscriber>();  
5 ...  
6 ...  
7 var app = builder.Build();  
8
```

Quellcode 10.12: Registrieren des Event Subscribers

Ebenfalls muss der Event Subscriber folgendermaßen konfiguriert werden

```
1  public class EventSubscriber : BackgroundService {
2      private readonly IConfiguration _configuration;
3      private IConnection _connection;
4      public IModel _channel;
5      public string _queueName;
6      private string _exchangeName;
7
8      public EventSubscriber(IConfiguration configuration,
9          IEventProcessor processor) {
10         _configuration = configuration;
11         _eventProcessor = processor;
12         Init();
13     }
14
15     public void Init() {
16         var factory = new ConnectionFactory() {
17             HostName = AppSettings.IpAddress,
18             Port = int.Parse(AppSettings.RabbitPort),
19             UserName = "greatpowers",
20             Password = "greatpowers",
21             VirtualHost = "greatpowers"
22         };
23         _connection = factory.CreateConnection();
24
25         _channel = _connection.CreateModel();
26         _queueName = $"{_configuration["EventBusQueue"]}";
27         _exchangeName = _configuration["EventBusExchange"];
28     }
29 }
```

Quellcode 10.13: Konfiguration des Event Subscribers

Hier wird zusammen mit generellen Daten für die Kommunikation festgelegt, auf welche Queue dieser Event Subscriber gebunden ist.

Das Warten auf und Auslesen von Nachrichten aus der Queue geschieht durch diese Methode im Event Subscriber

```
1  protected override Task ExecuteAsync(CancellationToken
   stoppingToken) {
2      stoppingToken.ThrowIfCancellationRequested();
3
4      var consumer = new
5          EventingBasicConsumer(_eventSubscriber._channel);
6
7      consumer.Received += (moduleHandle, message) => {
8          var body = message.Body;
9          var eventMessage =
10             Encoding.UTF8.GetString(body.ToArray());
11
12
13      _eventSubscriber._eventProcessor.ProcessEvent(eventMessage);
14      };
15
16      return Task.CompletedTask;
17  }
18
```

Quellcode 10.14: Auslesen von Nachrichten im Event Subscribers

10.5.3. Event Records- und Handler

Event Records sind die Objekte, welche bei einer Änderung des Simulationsstatus über den Event Bus verschickt werden, um die Anwendungen anderer Simulanten über diese Änderung zu informieren. Durch die Verwendung solcher Records wird die Kommunikation einheitlich und übersichtlich gestaltet. Die Event Handler wiederum dienen dazu, erhaltene Nachrichten über Statusänderungen zu verarbeiten und die Anwendung dementsprechend zu refreshen. Die Zuweisung von erhaltenen Nachrichten an ihren korrespondierenden Event Handler geschieht im Event Processor, welcher wie folgt aufgebaut ist

```
1  public class EventProcessor{  
2  
3      public EventProcessor(IServiceScopeFactory scopeFactory) :  
4          base(scopeFactory){  
5          this["STATE_HAS_CHANGED"] = new  
6          StateHasChangedEventHandler(scopeFactory);  
7          this["READY_EVENT"] = new ReadyEventHandler(scopeFactory);  
8      }  
9  
10     public void ProcessEvent(string eventMessage) {  
11         var eventRecord =  
12         JsonSerializer.Deserialize<EventRecord>(eventMessage);  
13         var eventHandler = this[eventRecord.Type];  
14         eventHandler.Execute();  
15     }  
16 }
```

Quellcode 10.15: Event Processor

In dieser Simulationsanwendung werden 2 verschiedene Event Records verwendet, wobei Beide ihre grundlegenden Eigenschaften von der Event Record Klasse erben. Dies sieht dann folgendermaßen aus

```
1  public record EventRecord(string Type);  
2  
3  public record ReadyEvent() : EventRecord("READY_EVENT");  
4  public record StateHasChangedEvent() :  
5          EventRecord("STATE_HAS_CHANGED");  
6
```

Quellcode 10.16: Event Records

11. Zusammenfassung und Projektaussichten

11.1. Beantwortung der Forschungsfrage

Ziel dieser Arbeit war die Reduzierung sämtlicher Einflussfaktoren von historischen Ereignissen, auf einen Kriterienkatalog, mithilfe dessen ein Prototyp entwickelt wurde.

11.1.1. Vorbereitung

Für die Entwicklung der Prozesssimulation wurden verschiedene Architekturstrukturen entwickelt und miteinander verglichen. Kriterien wie Kompatibilität, Komplexität und Geschwindigkeit wurden berücksichtigt, um eine ideale Wahl zu treffen.

11.1.2. Umsetzung

Nach Abschluss der Planungsphase wurde mit der Umsetzung eines Prototypen begonnen. Der Entwicklungsstatus wurde in dem Projektmanagement Tool YouTrack [29] dokumentiert und schriftlich festgehalten.

11.1.3. Auslieferung

Zur Überprüfung der festgelegten Ziele in den Managementdokumenten wurde ein Abnahmemeeting vereinbart, in welchem die Erfüllung der genannten Punkte sichergestellt wurde.

11.1.4. Schlussfolgerung

Somit ist die Zielsetzung und die Aufgestellte These bewiesen.

Es ist in der Tat möglich ein komplexes Geschehen auf wenige zentrale Faktoren herunterzubrechen ohne den Simulationscharakter zu verzerren.

11.2. Ausbaupotential

Es wäre möglich, mithilfe des entwickelten Kriterienkatalogs und des Prototypen zu diversen geschichtlichen Ereignissen eine Simulation zu entwickeln.

Damit kann ein breit aufgestellter Themenkatalog aus Prozesssimulationen im Bezug auf historische Ereignisse zur Verfügung gestellt werden.

12. Installation

12.1. Systemanforderungen

- Windows 11 / 10
- Docker v 4.13.1
- .NET 7.0 SDK (v7.0) - Windows x64
- 64-bit Prozessor
- 8GB System RAM

12.2. Dateiträgerbeschreibung

Auf der beigelegten DVD befinden sich folgende Dateien:

- Die Auszuführende Programmdatei für Windows befindet sich unter **thesis/programm/win/app.cmd**
- Der Quellcode der Webapplikation befindet sich unter **thesis/programm/data**

12.3. Installation & Ausführung

12.3.1. Installation

Bei der Ausführung der cmd Datei wird eine Systemvariable gesetzt.

Diese kann in den Windows Einstellungen wieder entfernt werden.

*Systemvariable: **ASPNETCORE_ENVIRONMENT "Development"***

12.3.2. Ausführung

Um das Programm in Betrieb zu nehmen, ist eine Internetverbindung von Nöten.

- Starten des Docker Dienstes
- Ausführen der 'app.cmd' Datei unter **thesis/programm/win/app.cmd**

I. Literaturverzeichnis

- [1] *Europa Universalis*. Onlinequelle: URL: https://de.wikipedia.org/wiki/Europa_Universalis.
- [2] *Civilization*. Onlinequelle: URL: https://de.wikipedia.org/wiki/Sid_Meier%C3%A4%C5%BDs_Civilization.
- [3] *Risiko*. Informationsquelle: URL: [https://de.wikipedia.org/wiki/Risiko_\(Spiel\)](https://de.wikipedia.org/wiki/Risiko_(Spiel)).
- [4] *Risiko*. Informationsquelle: URL: https://de.wikipedia.org/wiki/Hearts_of_Iron_IV.
- [5] *Darstellung eines historischen Ereignisses in Kartographischer Form*. Bildquelle: URL: https://de.m.wikipedia.org/wiki/Datei:Risk_game_map.png.
- [6] *Darstellung eines historischen Ereignisses in Form eines Graphen*. Bildquelle: URL: https://upload.wikimedia.org/wikipedia/commons/2/24/Risk_Game_Graph.svg.
- [7] *Militärstreitkräfte 1942*. Onlinequelle: URL: https://de.wikipedia.org/wiki/Zweiter_Weltkrieg.
- [8] *Operation Catapult 1*. Onlinequelle: URL: https://de.wikipedia.org/wiki/Operation_Catapult.
- [9] *Operation Catapult 2*. Onlinequelle: URL: <https://www.welt.de/geschichte/zweiter-weltkrieg/article166194112/So-versenkten-die-Briten-1940-Frankreichs-Flotte.html>.
- [10] *Schlacht um Stalingrad*. Onlinequelle: URL: https://de.wikipedia.org/wiki/Schlacht_von_Stalingrad.
- [11] *Schlacht um Stalingrad*. Onlinequelle: URL: <https://www.studysmarter.de/schule/geschichte/nationalsozialismus/schlacht-um-stalingrad/>.
- [12] *Schlacht von Dunkirk*. Onlinequelle: URL: https://de.wikipedia.org/wiki/Schlacht_von_D%C3%A4nkirchen.
- [13] *Schlacht um Rostow*. Onlinequelle: URL: https://de.wikipedia.org/wiki/Schlacht_um_Rostow.
- [14] *Schlacht beim Asowschen Meer*. Onlinequelle: URL: https://de.wikipedia.org/wiki/Schlacht_am_Asowschen_Meer.
- [15] *Schlachten des Zweiten Weltkrieges*. Onlinequelle: URL: https://de.wikipedia.org/wiki/Kategorie:Schlacht_des_Zweiten_Weltkriegs.
- [16] *GDP*. Onlinequelle: URL: https://de.wikipedia.org/wiki/Liste_der_L%C3%A4nder_nach_Bruttoinlandsprodukt.
- [17] *Truppenverband Regiment*. Onlinequelle: URL: <https://de.wikipedia.org/wiki/Regiment>.
- [18] *Truppenverband Division*. Onlinequelle: URL: [https://de.wikipedia.org/wiki/Division_\(Milit%C3%A4r\)](https://de.wikipedia.org/wiki/Division_(Milit%C3%A4r)).
- [19] *Sichelschnittplan der Schlacht um Belgien*. Onlinequelle: URL: <https://de.wikipedia.org/wiki/Sichelschnittplan>.
- [20] *Schlacht um Belgien*. Onlinequelle: URL: https://de.wikipedia.org/wiki/Schlacht_um_Belgien.

- [21] *U-Boote und Zerstörer.* Onlinequelle: URL: https://www.planet-wissen.de/technik/schiffahrt/geschichte_der_u_boote/index.html.
- [22] *Darstellung der Truppen.* Onlinequelle: URL: <https://axisallies.com/rules/axis-allies-rules-1942-2nd-edition.pdf>.
- [23] Ransom, Roger L.: *Schlieffen's Gamble*, Seite 51–84. Cambridge University Press, 2018.
- [24] *Graph Map.* Onlinequelle: URL: <https://mapsontheweb.zoom-maps.com/image/650167979980554240>.
- [25] *Graph Analyse.* Onlinequelle: URL: <https://technology.amis.nl/wp-content/uploads/2018/12/image-8.png>.
- [26] *Truppen Diagramm.* Onlinequelle: URL: <https://www.ibiblio.org/hyperwar/USA/USA-P-Rabaul/charts/USA-P-Rabaul-3.jpg>.
- [27] *Entscheidungsbaum.* Onlinequelle: URL: <https://www.combatmission.lesliesoftware.com/FinalBlitzkrieg/Campaigns/Kampfgruppe%20Peiper.html>.
- [28] *Aufbau einer Blazor Server App.* Bildquelle: URL: <https://learn.microsoft.com/de-de/aspnet/core/blazor/?view=aspnetcore-7.0>.
- [29] *YouTrack.* Software: URL: <https://www.jetbrains.com/de-de/youttrack/>.
- [30] *Anwendung einer Zweidimensionalen Darstellung.* Onlinequelle: URL: https://de.wikipedia.org/wiki/Chronologie_des_Zweiten_Weltkrieges#/media/Datei:WWII.gif.
- [31] *Ardennenoffensive.* Onlinequelle: URL: <https://de.wikipedia.org/wiki/Ardennenoffensive>.
- [32] *Operation Bagration.* Onlinequelle: URL: https://de.wikipedia.org/wiki/Operation_Bagration.

II. Abbildungsverzeichnis

3.1. Risiko Prozesssimulation	18
3.2. Hearts of Iron Prozesssimulation	19
3.3. Geographische Darstellung in HoI4	20
3.4. Fokusbaum in HoI4	21
3.5. Wirtschaftlicher Ausbau in HoI4	21
3.6. Abbildung von Konflikten in HoI4	22
4.1. Darstellung eines historischen Ereignisses in Kartographischer Form [5]	23
4.2. Darstellung eines historischen Ereignisses in Form eines Graphen [6]	24
4.3. Darstellung eines historischen Ereignisses in Form eines Baumdiagramms	25
6.1. Fraktionsfarben	33
6.2. Darstellungsarten verschiedener Regionen	34
6.3. Darstellung Landtruppen [22]	34
6.4. Darstellung Seetruppen [22]	34
6.5. Sonstige Visualisierungselemente	35
6.6. Darstellungsarten verschiedener Interaktionen	35
6.7. Bewegungsaktion	36
6.8. Kaufbildschirm	36
6.9. Bewegungsaktion	37
6.10. Mobilisierung	37
6.11. Infoleiste	37
6.12. Truppenleiste	38
6.13. Truppenanzeige	39
6.14. Trefferanzeige	39
6.15. Verwundetenanzeige	39
8.1. Verwendung des kartographischen Modells im Kontext historischer Ereignisse	44
8.2. Darstellung des Schlieffen-Plan auf einem kartographischen Modell [23]	45
8.3. Verwendung des Graph Modells im Kontext geographischer Daten [24]	46
8.4. Analyse von Nachbarländer anhand eines Graphen [25]	47
8.5. Darstellung der Japanischen Truppen in Form eines Baumdiagramms [26]	48
8.6. Entscheidungsbaum diverser Strategien der Kampfgruppe Peiper [27]	49
9.1. Komponentenübersicht	51
9.2. Interaktion auf der Simulationsoberfläche	53
9.3. Empfangen einer Interaktion	53
9.4. Aufbau einer Blazor Server App [28]	54
9.5. Beispiel einer MySql Tabelle	55
9.6. Kommunikation zwischen Clients über den Event Bus	56
9.7. Aufbau der Docker Architektur	57
10.1. Sequenzdiagramm der Menüinteraktion	61
10.2. Render Sequenz	62
A.1. Aufbau der MySQL Datenbank	82
A.2. Klassendiagramm der Model Schicht	83

III. Tabellenverzeichnis

5.1. Realitätswerte	28
5.2. Maßstab	28
5.3. Simulationswerte	29
5.4. Trefferquoten	31
5.5. Wirtschaftsstärken	31
A.1. Kapitelverzeichnis	84
A.2. Arbeitstagebuch Nagelmaier	106
A.3. Arbeitstagebuch Wagner	108

IV. Quellcodeverzeichnis

9.1. Beispiel einer SVG Implementation	56
9.2. Einbinden eines Panzoom Containers	57
10.1. Starten der Dockerdienste	58
10.2. Konfiguration des WebApplication Builder	59
10.3. Konfiguration des ASP.NET Core Identity Framework	59
10.4. Konfiguration des MudBlazor Service	60
10.5. Registrierung einer Systemverwalteten Klasse	60
10.6. Inhalt des ARepositories	66
10.7. Inhalt des ACreatableRepository	67
10.8. Inhalt des SessionInfo-Repository	68
10.9. Konfiguration des Event Publishers	69
10.10 Publish Methode des Event Publishers	70
10.11 Aufruf der Publish Methode	70
10.12 Registrieren des Event Subscribers	70
10.13 Konfiguration des Event Subscribers	71
10.14 Auslesen von Nachrichten im Event Subscribers	72
10.15 Event Processor	73
10.16 Event Records	73

V. Abkürzungsverzeichnis

EF Entity Framework

UI User Interface

API Application Programming Interface

CRUD Create - Read/Retrieve - Update - Delete/Destroy

DI Dependency Injection

HoI4 Hearts of Iron IV

GDP Gross domestic product

A. Anhang

A.1. Grafiken

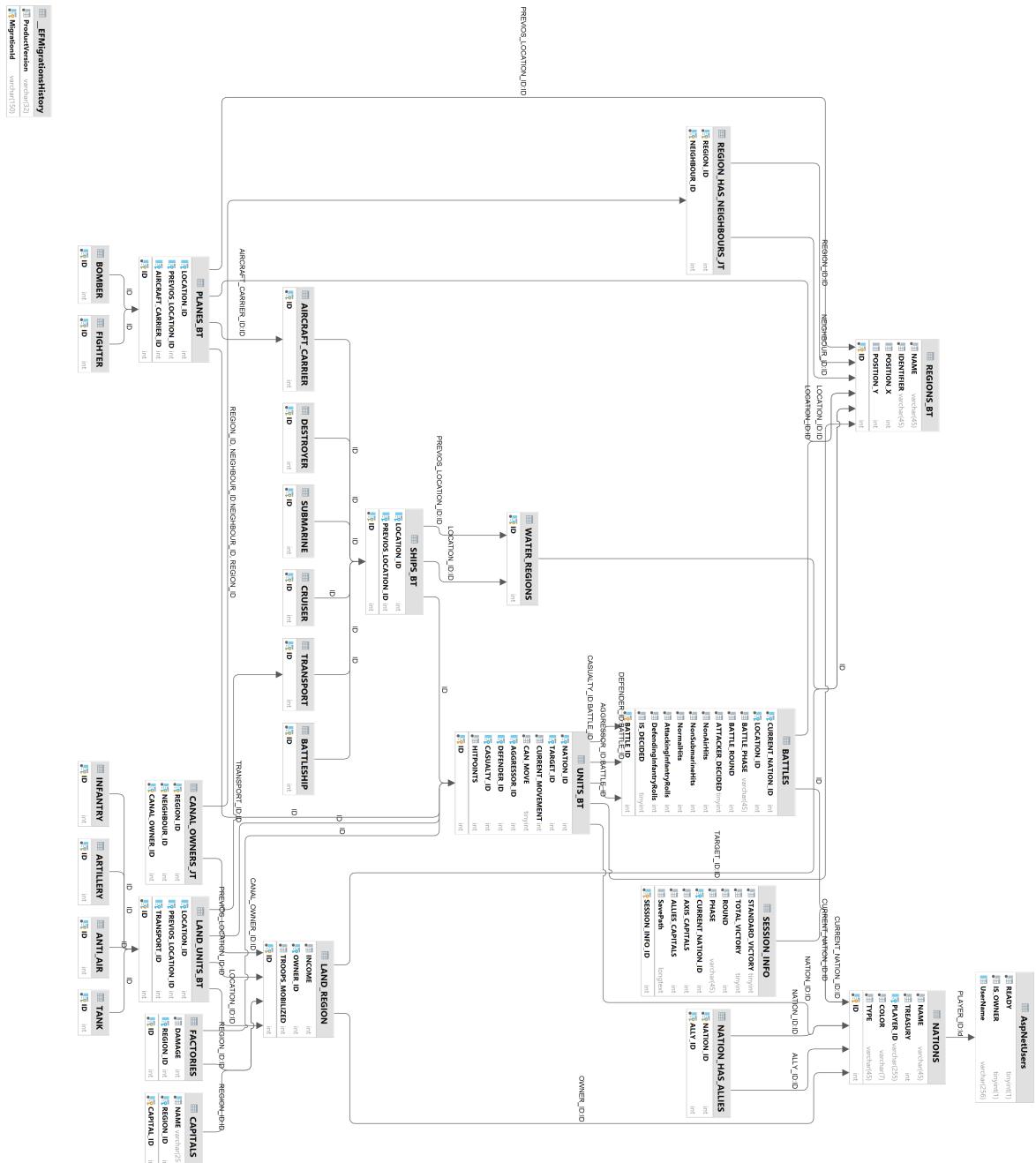


Abbildung A.1.: Aufbau der MySQL Datenbank

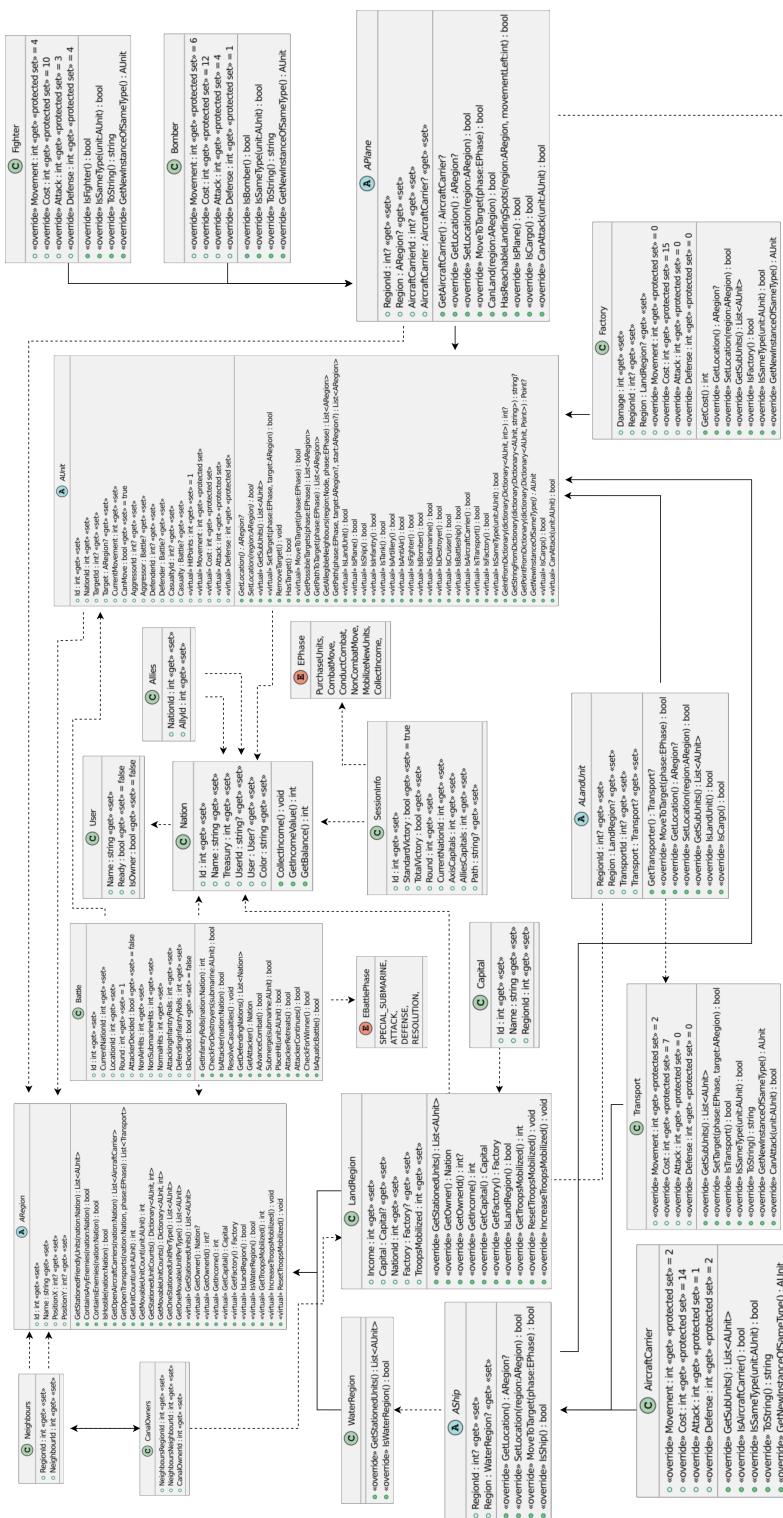


Abbildung A.2.: Klassendiagramm der Model Schicht

A.2. Kapitelverzeichnis

Kapitel	Autor
1	Nagelmaier Jonas
2	Nagelmaier Jonas
3	Wagner Lukas
4	Nagelmaier Jonas
5	Wagner Lukas
6	Nagelmaier Jonas
7	Wagner Lukas
8	Nagelmaier Jonas
9	Nagelmaier Jonas
10.1	Nagelmaier Jonas
10.2	Nagelmaier Jonas
10.3	Wagner Lukas
10.4	Wagner Lukas
10.5	Wagner Lukas
11	Nagelmaier Jonas
12	Nagelmaier Jonas

Tabelle A.1.: Kapitelverzeichnis

A.3. Besprechungsprotokolle

htlkrems Bautechnik & IT	HTBL Krems Höhere Lehranstalt für Informationstechnologie	Reife- und Diplomprüfung
------------------------------------	--	-------------------------------------

Gespräch Nr. 1

Ort: HTL Krems

Datum: 18.10.2022

Zeit: 8:40 – 9:00

Anwesend:

- Dipl. Ing. Msc Paul Panhofer Bsc
- Jonas Nagelmaier
- Lukas Wagner

Besprechungsinhalt:

Name	Notiz
Pflichtenheft	<ul style="list-style-type: none">• Ergänzung von vollständigen Mockups• Detailbeschreibung einzelner Mockups• Hinzufügen eines Ablaufdiagramms• Restlicher Inhalt ist in Ordnung
Ticketsystem	<ul style="list-style-type: none">• Projektarbeit auf Tickets in einem Rahmen von 1-8 Stunden aufteilen• Ticketdauer schätzen und eintragen• Ticket einer Person zuweisen• PSP erstellen

Aufgaben:

Name	Notiz	zu erledigen bis
Pflichtenheft	Fehlende Punkte in das Pflichtenheft einpflegen	03.11.2022
Ticketsystem	Projekttickets in das Programm YouTrack einpflegen	03.11.2022

Nächstes Gespräch:

Ort: HTL Krems / 4AHIT

Datum: 03.11.2022

Zeit: 12:15 – 12:30

Voraussichtlich anwesend:

- Dipl. Ing. Msc Paul Panhofer Bsc
- Jonas Nagelmaier
- Lukas Wagner

Gespräch Nr. 2

Ort: HTL Krems

Datum: 03.11.2022

Zeit: 12:15 – 12:30

Anwesend:

- Dipl. Ing. Msc Paul Panhofer Bsc
- Jonas Nagelmaier
- Lukas Wagner

Besprechungsinhalt:

Name	Notiz
Spezifizieren	Spezifizierung des Pflichtenheftes in Bezug auf den Zweiten Weltkrieg
Top-Down Chart	Erstellung einer Produktübersicht
Mockup Beschreibung	Zu jedem Mockup eine Beschreibung
Übersicht über Funktionen	Eine Übersichtstabelle über alle verfügbaren Funktionen

Aufgaben:

Name	Notiz	zu erledigen bis
Besprechungsinhalte	Umsetzung der Besprechungsinhalte	02.12.2022
Docker	<ul style="list-style-type: none">• Recherchieren nach Möglichkeiten, zur Implementierung eines lokalen Docker Netzwerkes• Prototypen ausprogrammieren	02.12.2022

Nächstes Gespräch:

Ort: HTL Krems / 5AHIT

Datum: 02.12.2022

Zeit: 07:45 – 8:00

Voraussichtlich anwesend:

- Dipl. Ing. Msc Paul Panhofer Bsc
- Jonas Nagelmaier
- Lukas Wagner

Gespräch Nr. 3

Ort: HTL Krems

Datum: 02.12.2022

Zeit: 7:50 – 8:10

Anwesend:

- Dipl. Ing. Msc Paul Panhofer Bsc
- Jonas Nagelmaier
- Lukas Wagner

Besprechungsinhalt:

Name	Notiz
Docker Port	Ist ein Docker Port belegt werden die nächsten 5 Ports überprüft und danach eine Fehlermeldung geworfen

Aufgaben:

Name	Notiz	zu erledigen bis
Aktualisierung des Pflichtenhefts	Übernehmen der Aktualisierungen aus Meeting Nr.2	20.12.2022
Überprüfung der Ports	Fehlervermeidung durch Überprüfen der verfügbaren Ports	20.12.2022
Oberfläche	Implementierung einer SVG Karte inklusiver Interaktion per Pfeile	20.12.2022
Logik	Implementierung der Bewegungslogik und Teile der Kampflogik	20.12.2022

Nächstes Gespräch:

Ort: HTL Krems / COW-W

Datum: 20.12.2022

Zeit: 09:40 – 10:00

Voraussichtlich anwesend:

- Dipl. Ing. Msc Paul Panhofer Bsc
- Jonas Nagelmaier
- Lukas Wagner

Gespräch Nr. 4

Ort: HTL Krems / 5AHIT

Datum: 22.12.2022

Zeit: 08:00 – 08:10

Anwesend:

- Dipl. Ing. Msc Paul Panhofer Bsc
- Jonas Nagelmaier
- Lukas Wagner

Besprechungsinhalt:

Name	Notiz
Code refactor	Aufteilen der Methoden von der abstrakten Klasse in die Kindklassen

Aufgaben:

Name	Notiz	zu erledigen bis
Code refactor	Implementierung des besprochenen Inhaltes	20.01.2023
Fertige Karte	Fertigstellung der Svg Länder Pfade	20.01.2023
Initialdaten fertig	Implementierung Database Seeding	20.01.2023
Kampflogik + Kampf Bewegung	Implementierung der Kampflogik + Kampfbewegung	20.01.2023
Kampf Darstellung	Erstellen einer Kampf Darstellung	20.01.2023
Phasen Menüs	Implementierung der Phasen Menüs	20.01.2023

Nächstes Gespräch:

Ort: HTL Krems / 5AHIT

Datum: 20.01.2023

Zeit: 08:00 – 08:10

Voraussichtlich anwesend:

- Dipl. Ing. Msc Paul Panhofer Bsc
- Jonas Nagelmaier
- Lukas Wagner

Gespräch Nr. 5

Ort: HTL Krems

Datum: 20.01.2023

Zeit: 7:50 – 8:10

Anwesend:

- Dipl. Ing. Msc Paul Panhofer Bsc
- Jonas Nagelmaier
- Lukas Wagner

Besprechungsinhalt:

Name	Notiz
Abgabe der Alphaversion	Sämtliche Features sind implementiert

Aufgaben:

Name	Notiz	zu erledigen bis
Eventbus	Adaptierung und Fertigstellung des Eventbus	24.02.2023
Simulation	Adaptierung und Fertigstellung der Simulation	24.02.2023
Containerisierung	Adaptierung und Fertigstellung der Containerisierung	24.02.2023
Cleanup	Aufräumen von veralteten Codeabschnitten	24.02.2023

Nächstes Gespräch:

Ort: HTL Krems / COW-W

Datum: 24.02.2023

Zeit: 09:40 – 10:00

Voraussichtlich anwesend:

- Dipl. Ing. Msc Paul Panhofer Bsc
- Jonas Nagelmaier
- Lukas Wagner

Gespräch Nr. 6

Ort: HTL Krems

Datum: 24.02.2023

Zeit: 9:40 – 10:00

Anwesend:

- Dipl. Ing. Msc Paul Panhofer Bsc
- Jonas Nagelmaier
- Lukas Wagner

Besprechungsinhalt:

Name	Notiz
Abnahme des technischen Produktes	Produkt wurde abgenommen

Aufgaben:

Name	Notiz	zu erledigen bis
Finalisieren des technischen Produktes	Das Programm in eine .exe Datei konvertieren	10.03.2023
März Präsentation	Inhalt: <ul style="list-style-type: none">• Beantwortung der Forschungsfrage• 7 – 10 Folien• Ablaufdiagramme	10.03.2023
Juni Präsentation	Inhalt: <ul style="list-style-type: none">• Fokus auf der Forschungsfrage• Prozesssimulation• Visualisierungskonzepte (Nagelmaier)• Kriterienkatalog (Wagner)• Wahrscheinlichkeitsrechnungen (Wagner)	10.03.2023
schriftlicher Teil	Inhalt: <ul style="list-style-type: none">• Prozesssimulation• Visualisierung (Nagelmaier)• Modellbildung (Wagner)• Beschreibung der Zielsetzung• Beschreibung des Prototyps• Bewertung der Forschungsfrage	10.03.2023

Nächstes Gespräch:

Ort: HTL Krems / COW-W

Datum: 10.03.2023

Zeit: 09:40 – 10:00

Voraussichtlich anwesend:

- Dipl. Ing. Msc Paul Panhofer Bsc
- Jonas Nagelmaier
- Lukas Wagner

htlkrems	HÖHERE TECHNISCHE BUNDES - LEHRANSTALT Krems
Abteilung:	Informationstechnologie

A.4. Pflichtenheft

Pflichtenheft

für

HTL Krems
z.H. Paul Panhofer
Alaunatalstrasse 29
3500 Krems an der Donau

im Folgenden Auftraggeber genannt

von

Nagelmaier Jonas und Wagner Lukas

im Folgenden Auftragnehmer genannt

Inhaltsverzeichnis

1 Einleitung	3
1.1 Ausgangslage	3
2 Entwicklungsteam	3
2.1 Mitglieder	3
2.2 Rollen und Verantwortung	3
3 Zielbestimmungen	4
3.1 Ziele des Benutzers	4
3.2 Ziele der Software	4
4 Produkteinsatz	4
4.1 Anwendungsbereiche und Zielgruppen	4
4.2 Betriebsbedingungen	4
5 Produktfunktionen	5
5.1 Benutzer	5
5.2 Hostsystem	7
6 UML - Chart	8
7 Architekturbeschreibung	9
7.1 Systemarchitektur	9
8 Installation	10
9 Rechtliche Bestimmungen	10
9.1 Gewährleistungs- und Garantiehinweise	10
9.2 Veröffentlichung	10
10 Lieferumfang	10
11 Kommunikationsverzeichnis	11

	<p style="text-align: center;">HÖHERE TECHNISCHE BUNDES - LEHRANSTALT Krems</p> <p>Abteilung: Informationstechnologie</p>
---	---

1 Einleitung

1.1 Ausgangslage

Die Bewertung komplexer soziopolitischer Prozesse im Kontext historischer Studien bleibt auch heute noch das Privileg einer entsprechend profilierten Ausbildung. Durch den Einsatz einer Prozesssimulation soll die Möglichkeit geschaffen werden, der Gesellschaft ein Tool zur Vertiefung ihres Verständnisses von komplexen soziopolitischen Prozessen bereitzustellen.

2 Entwicklungsteam

2.1 Mitglieder

Das Projektteam besteht aus zwei Mitgliedern:

- Nagelmaier Jonas
- Wagner Lukas

Betreuende Lehrkraft:

Dipl.-Ing. Panhofer Paul MSc. BSc.

für Kontaktdaten siehe „Kommunikationsverzeichnis“

2.2 Rollen und Verantwortung

Die Rollen sind folgendermaßen aufgeteilt:

Nagelmaier Jonas

- Interaktive Clients zur Visualisierung und Adaptierung historischer Prozesse
- Webschnittstelle basierend auf SVG Technologien
- .NET Controller

Wagner Lukas

- Konzeptionierung einer relationalen Datenbankstruktur
- Modelschicht zur Verwaltung von Entitäten
- Domainschicht zur Kapselung des Datenbankzugriffs

3 Zielbestimmungen

3.1 Ziele des Benutzers

- Simulationsstände speichern und laden
- Simulationen online hosten
- Großmächte simulieren
- Strategien erproben

3.2 Ziele der Software

- Erstellen/Speichern und laden einer Simulationssession
- Verwaltung der Systementitäten
- Bereitstellung einer Datenbankverbindung
- Bereitstellung einer nicht dedizierten Host Session

4 Produkteinsatz

4.1 Anwendungsbereiche und Zielgruppen

Der Anwendungsbereich der Webapplikation beschränkt sich auf die Personengruppe mit Interesse an historischen Ereignissen. Zielgruppe sind Personen mit historischen Interessen und Grundkompetenzen im Umgang mit geeigneter Hardware.

4.2 Betriebsbedingungen

Es wird ein System zur Ausführung der Webapplikation benötigt. Um alle Funktionen der Software nutzen zu können benötigt der Benutzer eine stabile Internetverbindung und einen Internet-Browser.

5 Produktfunktionen

5.1 Benutzer

Menüprozess

Startet der Benutzer die Webapplikation, so beginnt folgender Prozess:

1. Zuerst wählt der Benutzer aus folgenden Optionen:
 - (a) Erstellung einer neuen Session
 - (b) Laden einer vorhandenen Session
 - (c) Beitreten einer gehosteten Session
2. Entscheidet sich der Benutzer für:
 - (a) den Erstellungsprozess beginnt Konfigurationsfluss
 - (b) den Ladeprozess beginnt der Interaktionsfluss

Konfigurationsfluss

1. Folgende Einstellungen stehen dem Benutzer zur Auswahl:
 - (a) Lokale Simulation – Onlinehosting
 - (b) Fraktionsauswahl
 - (c) Simulationseigenschaften

Interaktionsfluss

1. Dem Benutzer ist es möglich zu jedem Zeitpunkt die Simulation zu speichern
2. Die Fraktionen werden rundenbasiert durchgegangen
 - (a) Die aktuelle Fraktion kann nun seine Phasen durchlaufen
 - i. Kaufphase
 - ii. Spezialkampfphase
 - iii. Kampfphase
 - iv. Bewegungsphase
 - v. Mobilisierungsphase
 - vi. Statusphase
 - (b) Die nächste Fraktion beginnt ihren Durchlauf
3. Wenn eine Fraktion die festgelegten Siegeskriterien erfüllt, ist die Simulation beendet

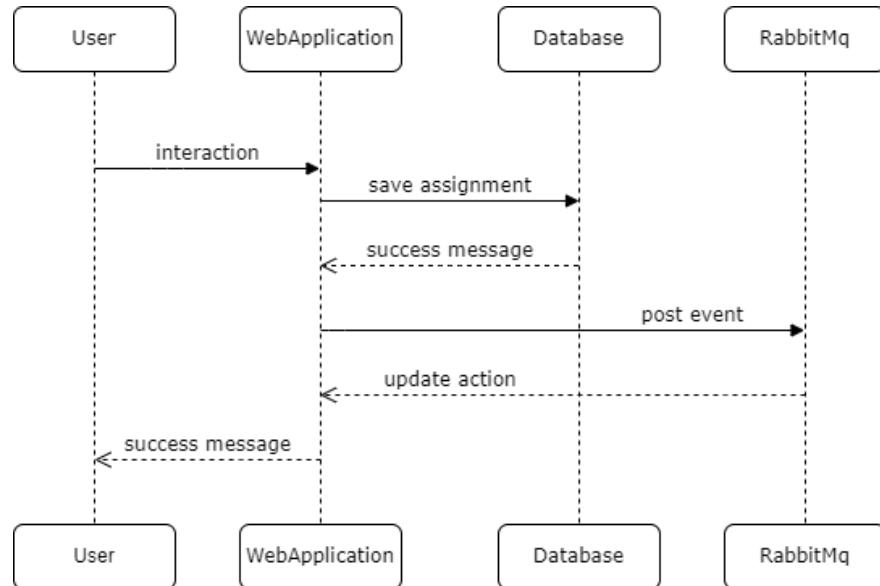


Abbildung 1: Interaktion auf der Simulationsoberfläche

Host-Joinprozess

1. Um eine Simulation zu hosten oder einer vorhandenen Simulation beizutreten, benötigt jeder Benutzer einen Usernamen
2. Entscheidet sich ein Benutzer für die online Funktion, erscheint je nach Art ein Fenster mit folgenden Informationen:
 - (a) Host
 - i. Ip-Adresse + Port unter der die Session erreichbar ist
 - (b) Beitretnende
 - i. Eingabefeld für die Session Adresse

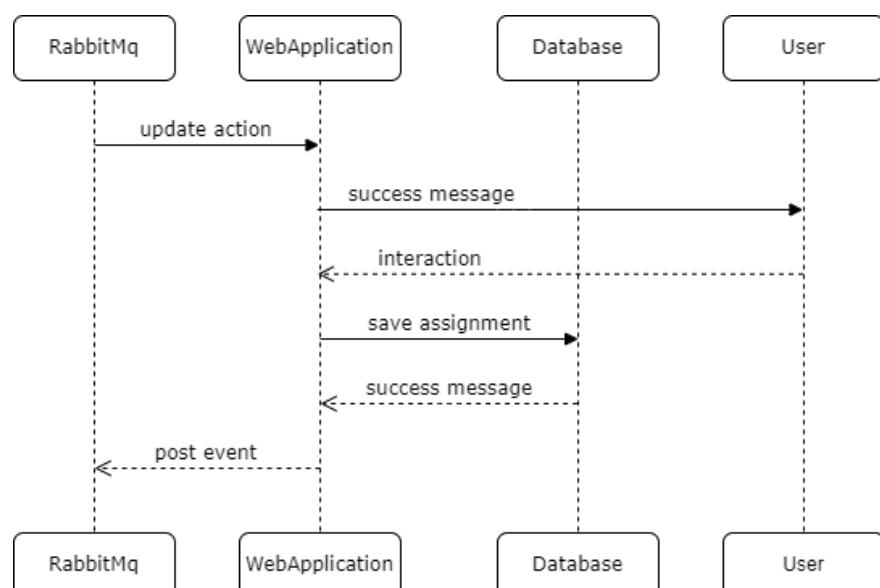


Abbildung 2: Empfangen einer Interaktion

5.2 Hostsystem

Um zwischen mehreren Benutzern in einer Session differenzieren zu können, muss bei einem Erstbeitritt einer Session ein Username mitgegeben werden.

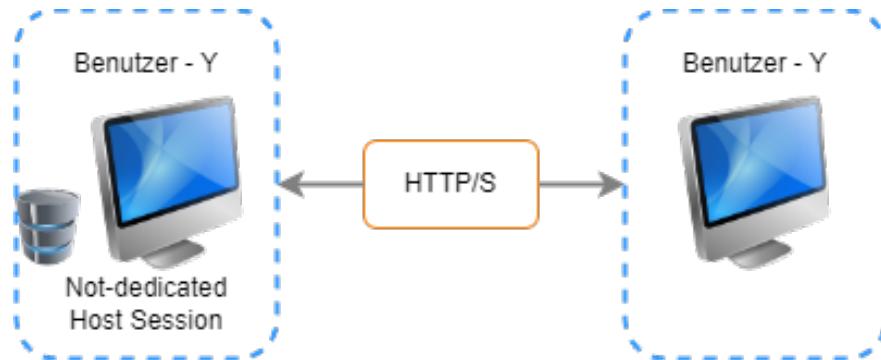
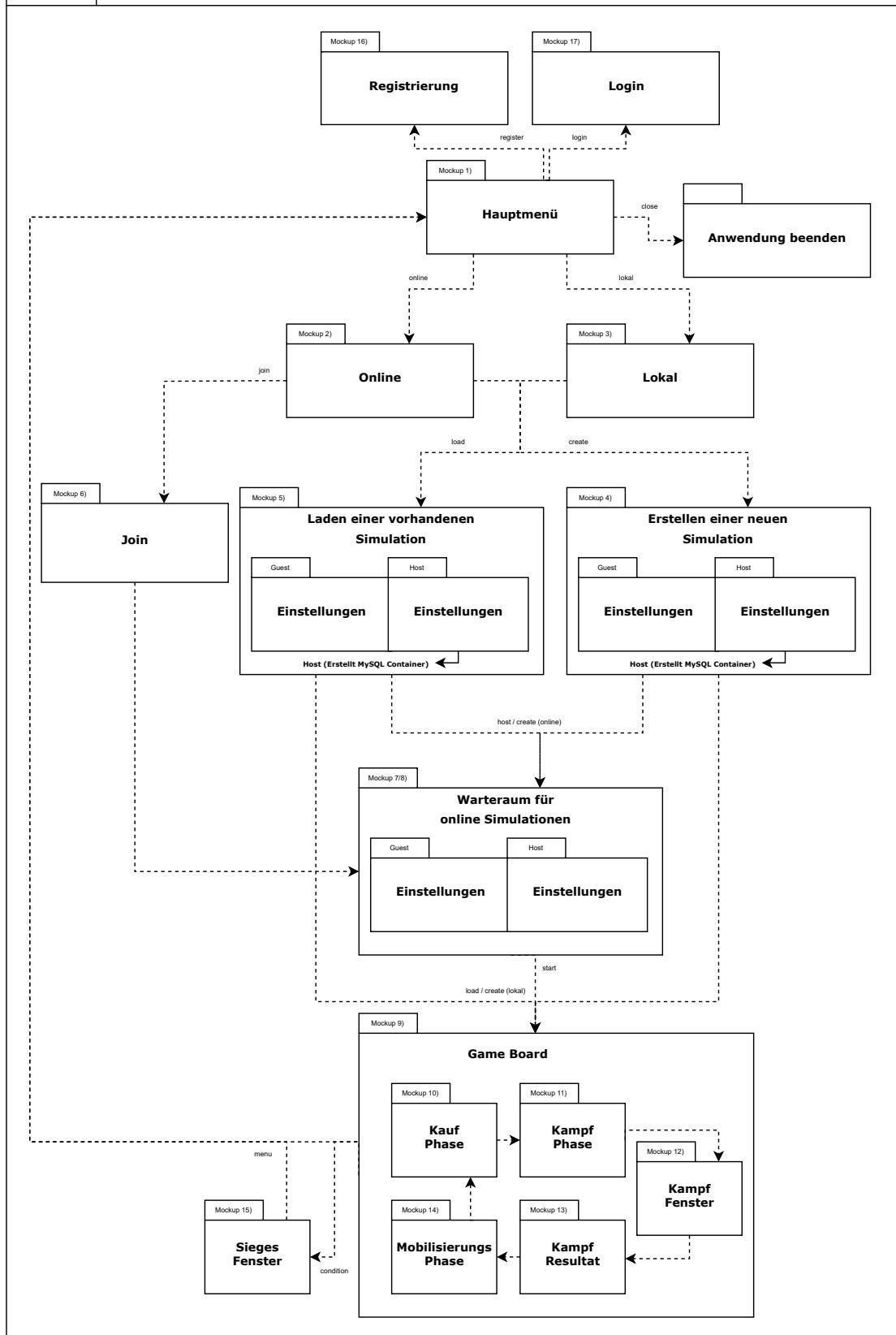


Abbildung 3: Menüprozess

6 UML - Chart

Thesis - UML



7 Architekturbeschreibung

7.1 Systemarchitektur

Die serverseitige Komponente wird mit Docker realisiert. Alle Server werden containerisiert. Die Simulationssessions werden am Client als nicht dedizierte host Session gehostet. Zur Speicherung der Simulation wird die relationale Datenbank MySql benutzt. Die Web-Applikation wird mit Blazor erstellt. Für die grafische Abstrahierung wird die SVG-Technologie eingesetzt. Die Architekturstruktur zeigt die Kommunikationswege der einzelnen Komponenten.

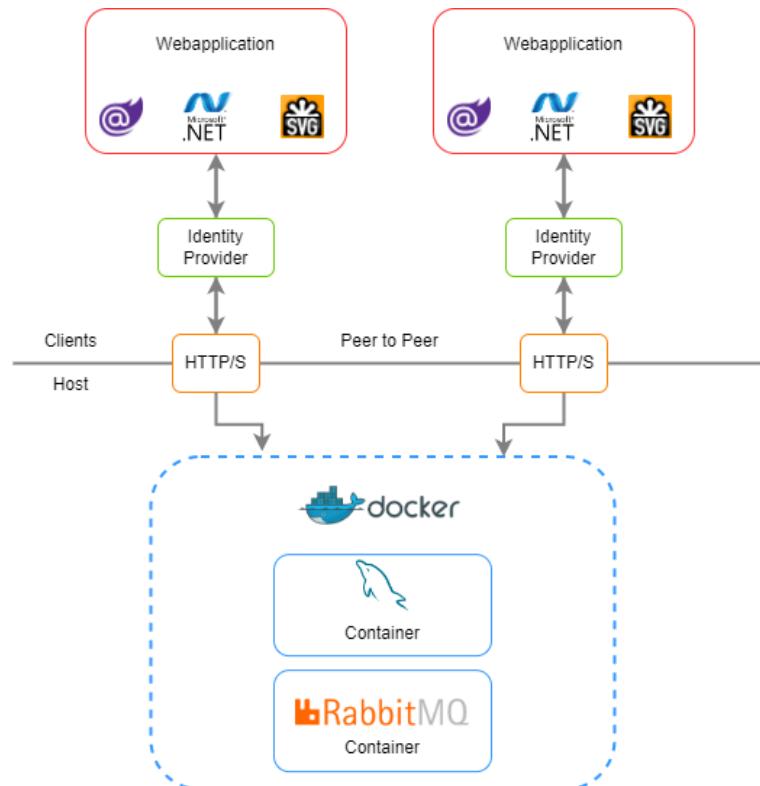


Abbildung 4: Architekturdiagramm

8 Installation

Zur Installation wird dem Auftraggeber eine schriftliche Installationsanleitung bereitgestellt.

9 Rechtliche Bestimmungen

9.1 Gewährleistungs- und Garantiehinweise

Jegliche Gewährleistung sowie Garantie sind ausgeschlossen.

9.2 Veröffentlichung

Da diese Anwendung im Rahmen einer Diplomarbeit an einer höheren technische Bundeslehranstalt entsteht, ist es notwendig, dass alle elektronisch gefertigten Dokumente und Niederschriften, sowie der gesamte Quellcode auf analogem und digitalem Wege der Beurteilung zugeführt werden. Des Weiteren erklärt sich der Auftraggeber damit einverstanden, dass die vorher erwähnten Inhalte in vollem Umfang der Öffentlichkeit zur Verfügung gestellt werden dürfen.

10 Lieferumfang

Das fertige Projekt wird in folgendem Umfang dem Auftraggeber übergeben:

- (a) .exe Datei inklusive Programmdatein
- (b) Installationsanleitung

	HÖHERE TECHNISCHE BUNDES - LEHRANSTALT Krems Abteilung: Informationstechnologie
---	--

11 Kommunikationsverzeichnis

 Bautechnik & IT	HTBL Krems Höhere Lehranstalt für Informationstechnologie	Reife- und Diplomprüfung
---	---	---

Projekt Name Great - Powers	Team Nagelmaier Jonas Wagner Lukas Dipl. Ing. Msc Paul Panhofer Bsc
Dokument Nr G_01 Name Kommunikationsverzeichnis	

Versionsgeschichte

Version	Datum	Autor/in	Änderungen
1.0	02.02.2022	Nagelmaier Jonas	Initiale Erstellung

Planungsteam

Rolle	Name	Mail	Anmerkungen
Planungsleitung	Jonas Nagelmaier	j.nagelmaier@htlkrems.at	Erreichbar Mo-Fr
Mitarbeit	Lukas Wagner	l.wagner@htlkrems.at	Erreichbar Mo-Fr

Team-externe Kontakte

Rolle	Name	Mail	Anmerkungen
Betreuer	Paul Panhofer	p.panhofer@htlkrems.at	Erreichbar Di-Fr

Voraussichtliches Projektteam

Rolle	Name	Mail	Anmerkungen
Projektleiter	Jonas Nagelmaier	j.nagelmaier@htlkrems.at	Erreichbar Mo-Fr
Mitarbeiter	Lukas Wagner	l.wagner@htlkrems.at	Erreichbar Mo-Fr

A.5. Arbeitspakte

YouTrack Issues Dashboards New Issue

Project-Flow order by: {issue id} asc

» Matches 47 issues

N	Thesis-1	Arbeitspakte erstellen	Normal	Task	Submitted	Jonas Nagelmaier	3h	?	Jonas Nagelmaier	Yesterday	
N	Thesis-2	Pflichtenheft erstellen	Normal	Task	Submitted	Jonas Nagelmaier	1d 6h	?	Jonas Nagelmaier	15:31	
N	Thesis-11	Mockup modellierung	Normal	Task	Submitted	Jonas Nagelmaier	1d	?	Jonas Nagelmaier	14:51	
N	Thesis-47	Pflichtenheft verfassen	Normal	Task	Submitted	Jonas Nagelmaier	4h	?	Jonas Nagelmaier	15:31	
M	Thesis-9	Blazor Webapplikation	Major	Epic	Submitted	Unassigned	5w 1d 2h	?	Jonas Nagelmaier	15:37	
N	Thesis-3	Model Schicht	Normal	Task	Submitted	Lukas Wagner	1d 2h	?	Jonas Nagelmaier	14:51	
N	Thesis-4	Modellierung in MySQL Workbench	Normal	Task	Submitted	Lukas Wagner	2h	?	Jonas Nagelmaier	15:52	
N	Thesis-5	Implementierung der Konzeptionierung	Normal	Task	Submitted	Lukas Wagner	2h	?	Jonas Nagelmaier	15:52	
N	Thesis-6	Implementierung - DbContext	Normal	Task	Submitted	Lukas Wagner	2h	?	Jonas Nagelmaier	15:52	
N	Thesis-14	Database seeding	Normal	Task	Submitted	Lukas Wagner	4h	?	Jonas Nagelmaier	15:52	
N	Thesis-7	MySQL Datenbank	Normal	Task	Submitted	Jonas Nagelmaier	3h	?	Jonas Nagelmaier	16:01	
N	Thesis-8	Dockerfile schreiben	Normal	Task	Submitted	Jonas Nagelmaier	3h	?	Jonas Nagelmaier	15:09	
M	Thesis-10	Implementierung der Logik	Major	Epic	Submitted	Lukas Wagner	2w 2d 7h	?	Jonas Nagelmaier	16:01	
N	Thesis-12	Erstellen des Abstract Repositories	Normal	Task	Submitted	Lukas Wagner	1h	?	Jonas Nagelmaier	14:51	
N	Thesis-13	Implementierung der Repositories	Normal	Task	Submitted	Lukas Wagner	3h	?	Jonas Nagelmaier	14:51	

▼	N	Thesis-15	Implementierung der Simulationsphasenlogik	🔒	
	Normal	Task	Submitted	Lukas Wagn ● 1w 4d 7h ?	Jonas Nagelmaier 15:02
▼	N	Thesis-16	Kaufphase	🔒	
	Normal	Task	Submitted	Lukas Wagn ● 1d 2h ?	Jonas Nagelmaier 14:51
	N	Thesis-17	Einheiten Auswahl	🔒	
	Normal	Task	Submitted	Lukas Wag ● 5h ?	Jonas Nagelmaier 14:51
	N	Thesis-18	Plazierung der Einheiten	🔒	
	Normal	Task	Submitted	Lukas Wag ● 5h ?	Jonas Nagelmaier 14:51
▼	N	Thesis-19	Kampfphase	🔒	
	Normal	Task	Submitted	Lukas Wagn ● 1w 1d 7h ?	Jonas Nagelmaier 16:01
	N	Thesis-20	Kampfbewegung	🔒	
	Normal	Task	Submitted	Lukas Wag ● 1d 7h ?	Jonas Nagelmaier 15:02
	N	Thesis-21	Kampfabhandlung	🔒	
	Normal	Task	Submitted	Lukas Wag ● 1w ?	Jonas Nagelmaier 14:51
	N	Thesis-22	Bewegungsphase	🔒	
	Normal	Task	Submitted	Lukas Wagn ● 1d 2h ?	Jonas Nagelmaier 15:02
	N	Thesis-23	Mobilisierungsphase	🔒	
	Normal	Task	Submitted	Lukas Wagn ● 2h ?	Jonas Nagelmaier 14:51
	N	Thesis-24	Siegesbedingungen überprüfen	🔒	
	Normal	Task	Submitted	Lukas Wag ● 2h ?	Jonas Nagelmaier 14:52
	N	Thesis-26	Implementierung des Hosting Systems	🔒	
	Normal	Task	Submitted	Lukas Wagn ● 2d 4h ?	Jonas Nagelmaier 15:01
▼	M	Thesis-25	Weboberfläche	🔒	
	Major	Epic	Submitted	Jonas Nagel ● 2w 1d 6h ?	Jonas Nagelmaier 16:01
▼	N	Thesis-29	Menü Oberfläche	🔒	
	Normal	Task	Submitted	Jonas Nagel ● 4d 5h ?	Jonas Nagelmaier 16:01
	N	Thesis-27	Oberfläche des Hauptmenüs	🔒	
	Normal	Task	Submitted	Jonas Nagel ● 5h ?	Jonas Nagelmaier 15:37
	N	Thesis-28	Oberfläche des Online Menüs	🔒	
	Normal	Task	Submitted	Jonas Nagel ● 2h ?	Jonas Nagelmaier 15:27
	N	Thesis-30	Oberfläche des Lokalen Menüs	🔒	
	Normal	Task	Submitted	Jonas Nagel ● 2h ?	Jonas Nagelmaier 15:08
	N	Thesis-31	Oberfläche Simulationskonfiguration	🔒	
	Normal	Task	Submitted	Jonas Nagel ● 1d ?	Jonas Nagelmaier 15:27
	N	Thesis-32	Oberfläche des Warteraums	🔒	
	Normal	Task	Submitted	Jonas Nagel ● 1d ?	Jonas Nagelmaier 15:27

N	Thesis-33	Oberfläche Laden der Session	🔒								
Normal		Task		Submitted		Jonas Nagel		6h		?	
Jonas Nagelmaier	15:27										
N	Thesis-34	Oberfläche Online Beitritt	🔒								
Normal		Task		Submitted		Jonas Nagel		6h		?	
Jonas Nagelmaier	15:27										
▼	N	Thesis-35	Simulations Oberfläche	🔒							
Normal		Task		Submitted		Jonas Nagel		1w 2d 1h		?	
Jonas Nagelmaier	15:27										
N	Thesis-36	Erstellung einer SVG Karte	🔒								
Normal		Task		Submitted		Jonas Nagel		1d 2h		?	
Jonas Nagelmaier	15:27										
N	Thesis-37	Basis Oberfläche	🔒								
Normal		Task		Submitted		Jonas Nagel		4h		?	
Jonas Nagelmaier	15:27										
N	Thesis-38	Länder Interaktion	🔒								
Normal		Task		Submitted		Jonas Nagel		4h		?	
Jonas Nagelmaier	15:27										
N	Thesis-39	Darstellung der Einheiten	🔒								
Normal		Task		Submitted		Jonas Nagel		1d 7h		?	
Jonas Nagelmaier	15:27										
N	Thesis-40	Kauffenster	🔒								
Normal		Task		Submitted		Jonas Nagel		6h		?	
Jonas Nagelmaier	15:27										
N	Thesis-41	Kampfenster	🔒								
Normal		Task		Submitted		Jonas Nagel		6h		?	
Jonas Nagelmaier	15:27										
N	Thesis-42	Kampfzusammenfassung	🔒								
Normal		Task		Submitted		Jonas Nagel		4h		?	
Jonas Nagelmaier	15:27										
N	Thesis-43	Mobilisierungsfenster	🔒								
Normal		Task		Submitted		Jonas Nagel		6h		?	
Jonas Nagelmaier	15:27										
N	Thesis-44	Siegesfenster	🔒								
Normal		Task		Submitted		Jonas Nagel		2h		?	
Jonas Nagelmaier	15:28										
M	Thesis-45	Dokumentierung der Resultate	🔒								
Major		Task		Submitted		Lukas Wagner		1w		?	
Jonas Nagelmaier	16:02										
M	Thesis-46	Dokumentierung der Resultate	🔒								
Major		Task		Submitted		Jonas Nagelmaier		1w		?	
Jonas Nagelmaier	16:02										

A.6. Projekttagebücher

A.6.1. Projekttagebuch Jonas Nagelmaier

Tag	Zeit	kumulativ	Fortschritt
Di 04.10.22	4h	4h	Ausarbeitung Pflichtenheft
Sa 15.10.22	4h	8h	Ausarbeitung Pflichtenheft
So 30.10.22	1h	9h	Ausarbeitung Arbeitspakete
Mo 31.10.22	2h	11h	Ausarbeitung Arbeitspakete
Mo 31.10.22	2h	13h	Modellierung der Datenbank
Di 01.11.22	2h	15h	Modellierung der Datenbank
Mi 02.11.22	4h	19h	Erstellung Mockups
Sa 05.11.22	3h	22h	Recherche Konzeptionierung Docker
So 06.11.22	3h	25h	Implementierung Docker-Containerisierung
So 06.11.22	3h	28h	Implementierung Docker-Containerisierung
Mo 28.11.22	4h	32h	Blazor Menu-Prototype
Di 29.11.22	3h	35h	Menu-Prototype Design
So 19.12.22	4h	39h	Gestaltung der Simulationsoberfläche
Mo 20.12.22	6h	45h	Gestaltung der Simulationsoberfläche
Di 21.12.22	6h	51h	Gestaltung der Simulationsoberfläche
Mi 28.12.22	6h	57h	Generierung der Regionspfade
Do 29.12.22	6h	63h	Generierung der Regionspfade
So 01.01.23	3h	66h	Implementierung der Informationsleiste
Di 03.01.23	2h	68h	Implementierung der Truppenleiste
Di 03.01.23	6h	74h	Generierung der Regionspfade
Mi 04.01.23	6h	80h	Generierung der Regionspfade
Do 05.01.23	6h	86h	Einbindung der Karte
Sa 07.01.23	6h	92h	Einbindung der Karte
So 08.01.23	3h	95h	Konzeptionierung der Kaufphase
Mo 09.01.23	4h	99h	Implementierung der Kaufphase
Di 10.01.23	6h	105h	Implementierung der Kaufphase
Mi 11.01.23	6h	109h	Adaptierung der Kaufphase
Do 12.01.23	6h	115h	Konzeptionierung der Kampfbewegung
Fr 13.01.23	2h	117h	Implementierung der Kampfbewegung
Sa 14.01.23	6h	123h	Implementierung der Kampfbewegung
So 15.01.23	6h	129h	Adaptierung der Kampfbewegung
Mo 16.01.23	6h	135h	Konzeptionierung des Schlachtfeldes
Di 17.01.23	8h	143h	Implementierung des Schlachtfeldes
Mi 18.01.23	5h	148h	Implementierung des Schlachtfeldes
Do 19.01.23	5h	153h	Adaptierung des Schlachtfeldes
Fr 20.01.23	5h	158h	Konzeptionierung der Bewegung
Mo 23.01.23	6h	164h	Implementierung der Bewegung
Di 24.01.23	2h	166h	Implementierung der Bewegung
Mi 25.01.23	6h	172h	Adaptierung der Bewegung
Do 26.01.23	4h	176h	Konzeptionierung der Mobilisierung
Fr 27.01.23	2h	178h	Implementierung der Mobilisierung

Mo 30.01.23	4h	182h	Implementierung der Mobilisierung
Di 31.01.23	4h	186h	Adaptierung der Mobilisierung
Mi 01.02.23	3h	189h	Implementierung der dezentralen Verteilungsschnittstelle
Mo 06.02.23	4h	193h	Implementierung der dezentralen Verteilungsschnittstelle
Di 07.02.23	2h	195h	Implementierung der dezentralen Verteilungsschnittstelle
Mi 08.02.23	6h	201h	Implementierung der dezentralen Verteilungsschnittstelle
Do 09.02.23	4h	205h	Implementierung der dezentralen Verteilungsschnittstelle
Fr 10.02.23	5h	210h	Implementierung der dezentralen Verteilungsschnittstelle
Sa 11.02.23	4h	214h	Implementierung der dezentralen Verteilungsschnittstelle
Mo 13.02.23	4h	218h	Testen und Bugfixing
Di 14.02.23	2h	220h	Testen und Bugfixing
Mi 15.02.23	6h	226h	Testen und Bugfixing
Do 16.02.23	4h	230h	Testen und Bugfixing
Fr 17.02.23	3h	233h	Testen und Bugfixing
Sa 18.02.23	3h	236h	Testen und Bugfixing
Mo 20.02.23	4h	240h	Testen und Bugfixing
Mi 22.02.23	2h	242h	Testen und Bugfixing
Do 23.02.23	6h	248h	Testen und Bugfixing
Fr 03.03.23	3h	251h	Verfassen der schriftlichen Arbeit
Sa 04.03.23	4h	255h	Verfassen der schriftlichen Arbeit
Mo 06.03.23	3h	258h	Verfassen der schriftlichen Arbeit
Di 07.03.23	2h	260h	Verfassen der schriftlichen Arbeit
Mi 08.03.23	4h	264h	Verfassen der schriftlichen Arbeit
Do 09.03.23	3h	267h	Verfassen der schriftlichen Arbeit
Sa 11.03.23	6h	273h	Verfassen der schriftlichen Arbeit
So 12.03.23	6h	279h	Verfassen der schriftlichen Arbeit
Di 14.03.23	2h	281h	Verfassen der schriftlichen Arbeit
Mi 15.03.23	4h	285h	Verfassen der schriftlichen Arbeit
Do 16.03.23	4h	289h	Verfassen der schriftlichen Arbeit
Sa 18.03.23	6h	295h	Verfassen der schriftlichen Arbeit
So 19.03.23	6h	301h	Verfassen der schriftlichen Arbeit

Tabelle A.2.: Arbeitstagebuch Nagelmaier

A.6.2. Projekttagebuch Lukas Wagner

Tag	Zeit	kumulativ	Fortschritt
Mo 31.10.22	2h	2h	Ausarbeitung Arbeitspakete
Mo 31.10.22	2h	4h	Modellierung der Datenbank
So 27.11.22	2h	6h	Erstellen der Entitäten
Fr 09.12.22	2h	8h	Konzeptionierung der Simulationslogik
Fr 09.12.22	1h	9h	Adaption der Datenbank und der Entitäten
Fr 09.12.22	2h	11h	Database Seeding
Sa 10.12.22	1h	12h	Unit Factories
Sa 10.12.22	5h	17h	Implementierung der Bewegungslogik
So 11.12.22	4h	21h	Implementierung der Bewegungslogik
Mo 19.12.22	2h	23h	Implementierung der Bewegungslogik
Mo 19.12.22	1h	24h	Database Seeding
Di 20.12.22	4h	28h	Implementierung der Bewegungslogik
Di 20.12.22	4h	32h	Überarbeitung der Entitäten
Mi 21.12.22	8h	40h	Implementierung der Bewegungslogik
Di 27.12.22	2h	42h	Database Seeding
Mi 28.12.22	8h	50h	Database Seeding
Do 05.01.23	5h	55h	Implementierung der Bewegungslogik
Fr 06.01.23	4h	59h	Implementierung der Bewegungslogik
So 08.01.23	3h	62h	Implementierung der Bewegungslogik
Di 10.01.23	4h	66h	Implementierung der Bewegungslogik
Di 10.01.23	4h	70h	Testen der Bewegungslogik
Mi 11.01.23	8h	78h	Testen der Bewegungslogik
Mi 11.01.23	3h	81h	Überarbeitung der Entitäten
Do 12.01.23	3h	84h	Überarbeitung der Entitäten
Do 12.01.23	3h	87h	Implementierung der Game Engine
Do 12.01.23	3h	90h	Implementierung der Multiplayer Funktionalitäten
Fr 13.01.23	2h	92h	Implementierung der Multiplayer Funktionalitäten
Sa 14.01.23	2h	94h	Implementierung der Multiplayer Funktionalitäten
Sa 14.01.23	6h	100h	Implementierung der Kampflogik
Mo 16.01.23	3h	103h	Testen der Bewegungslogik
Di 17.01.23	7h	110h	Testen der Bewegungslogik
Mi 18.01.23	2h	112h	Implementierung der Multiplayer Funktionalitäten
Mi 18.01.23	3h	115h	Überarbeitung der Bewegungslogik
Do 19.01.23	5h	120h	Überarbeitung der Bewegungslogik
Mi 01.02.23	10h	130h	Überarbeitung der Bewegungslogik
Di 14.02.23	2h	132h	Implementierung der Multiplayer Funktionalitäten
Di 14.02.23	3h	135h	Überarbeitung der Kampflogik
Mi 15.02.23	3h	138h	Überarbeitung der Kampflogik

Do 16.02.23	2h	140h	Überarbeitung der Kampflogik
Do 16.02.23	3h	143h	Implementierung der Multiplayer Funktionalitäten
So 19.02.23	1h	144h	Implementierung der Multiplayer Funktionalitäten
So 19.02.23	2h	146h	Überarbeitung der Kampflogik
Mo 20.02.23	7h	153h	Überarbeitung der Kampflogik
Di 21.02.23	1h	154h	Überarbeitung der Kampflogik
Mi 22.02.23	2h	156h	Überarbeitung der Kampflogik
Do 23.02.23	4h	160h	Überarbeitung der Kampflogik
Mi 01.03.23	3h	163h	Code Cleanup
Mo 13.03.23	4h	167h	Verfassen der schriftlichen Arbeit
Di 14.03.23	4h	171h	Verfassen der schriftlichen Arbeit
Mi 15.03.23	3h	174h	Verfassen der schriftlichen Arbeit
Do 16.03.23	4h	178h	Verfassen der schriftlichen Arbeit
Sa 18.03.23	5h	183h	Verfassen der schriftlichen Arbeit
So 19.03.23	3h	186h	Verfassen der schriftlichen Arbeit
Mo 20.03.23	1h	187h	Verfassen der schriftlichen Arbeit
Di 28.03.23	5h	192h	Verfassen der schriftlichen Arbeit

Tabelle A.3.: Arbeitstagebuch Wagner

A.7. Dateiträgerbeschreibung

Auf der beigelegten DVD befinden sich folgende Dateien:

- Die schriftliche Arbeit befindet sich unter **thesis/documentation/final.pdf**
- Die Installationsanleitung befindet sich unter **thesis/documentation/installation.pdf**
- Die schriftliche Arbeit in L^AT_EXFormat befindet sich unter **thesis/latex/**
- Sämtliche Projektmanagement Dateien befinden sich unter **thesis/management/**
- Die auszuführende Programmdatei für Windows befindet sich unter **thesis/programm/win/app.cmd**
- Der Quellcode der Webapplikation befindet sich unter **thesis/programm/data**