

Webbasierte Multiplayer Schach-App

Bachelorarbeit

vorgelegt von
Jasper Paul Fülle
Matrikelnummer 3367654

Betreut von
Prof. Dr. Thorsten Thormälen

Studiengang

Wirtschaftsinformatik

12. April 2023

Fachbereich Mathematik und Informatik
Philipps Universität Marburg

Inhaltsverzeichnis

Inhaltsverzeichnis	1
1 Einleitung	3
1.1 Motivation	3
1.2 Zielsetzung	3
1.3 Aufbau der Arbeit	4
2 Theoretische Grundlagen	5
2.1 Schach	5
2.2 Web-Technologien	5
2.2.1 Node.js und Express	5
2.2.2 Socket.io	5
2.2.3 React	5
2.2.4 PostgreSQL	5
3 Systemarchitektur	6
3.1 Frontend	7
3.1.1 React-Komponenten	7
3.1.2 Benutzerführung	7
3.1.3 Kommunikation mit dem Backend	7
3.2 Backend	7
3.2.1 API-Endpunkte	7
3.2.2 Datenbankstruktur	7
3.2.3 Echtzeit-Kommunikation	7
4 Implementierung	8
4.1 Frontend-Entwicklung	8
4.2 Backend-Entwicklung	8
4.3 Datenbank-Integration	8
5 Tests und Evaluation	9
5.1 Funktionalitätstests	9
5.2 Usability-Tests	9
5.3 Performancetests	9
5.4 Sicherheitstests	9

6	Fazit und Ausblick	10
6.1	Zusammenfassung der Ergebnisse	10
6.2	Limitationen	10
6.3	Potenzielle Erweiterungen und Weiterentwicklung	10
	Abbildungsverzeichnis	11
	Literaturverzeichnis	13

1. Einleitung

1.1 Motivation

Schach ist ein traditionsreiches und abwechslungsreiches Brettspiel, deren Ursprung nicht genau bestimmt werden kann. Es wird vermutet, dass das erste schachähnliche Spiel seinen Ursprung in Nordindien um 600 n. Chr. hatte und *Tschaturanga* hieß¹. Im Laufe der Jahrhunderte hat Schach eine bedeutende Rolle in der Kultur und Geschichte gespielt. So wurde beispielsweise die Schach-WM 1972 eine Art Machtkampf im kalten Krieg zwischen der UdSSR, welche den damaligen Schach dominierten, und der USA². Schach bleibt bis heute ein beliebtes Spiel, welches 2020 durch die Netflix Serie *Damengambit* und 2022 durch den Betrugsvorwurf von Magnus Carlsen an seinen 19-jährigen Gegner Hans Niemann³ eine breitere Aufmerksamkeit erhielt (siehe Abbildung 1). Darüber hinaus hat Schach im digitalen Zeitalter eine neue Popularität erreicht. Online-Schachplattformen wie chess.com verzeichnen Milliarden von Live-Partien⁴, während Schach Live-Streams auf Plattformen wie Twitch Millionen von Followern anziehen.

Die Entwicklung einer webbasierten Multiplayer-Schach-App bietet eine einzigartige Gelegenheit, ein traditionsreiches und beliebtes Spiel in das digitale Zeitalter zu bringen. Meine Motivation für diese Arbeit besteht darin, eine App zu entwickeln, die die Grundlagen einer Schach-App enthält und gleichzeitig eine solide Basis für zukünftige Erweiterungen und Verbesserungen bietet. Insbesondere plane ich, innovative Funktionen zu integrieren, die bislang in den gängigen Schach-Apps nicht vorhanden sind, wie z.B. die Möglichkeit, unterschiedliche Schachfiguren und -bretter als Belohnungen freizuschalten oder mit Freunden eine Gruppe zu gründen, welche in einer Liga auf- und absteigen kann. Durch die Entwicklung einer Schach-App mit neuen Funktionalitäten kann ich dazu beitragen, die Popularität von Schach zu steigern und vor allem das Spiel einem breiteren Publikum zugänglich zu machen.

1.2 Zielsetzung

Diese Bachelorarbeit hat das Ziel eine Schach-App zu entwerfen und zu implementieren, die eine intuitive User Experience und ein ansprechendes User Interface mit vielen nützlichen Funktionen beinhaltet.

¹van der Linde [1874]

²Hansen

³Sportschau [2022]

⁴chess.com [2018]

User Experience (*UX*) bezieht sich darauf wie ein Nutzer sich auf einer Anwendung bewegt und wie einfach und angenehm es für den Nutzer ist, die Funktionen der Anwendung zu verwenden.

Das User Interface (*UI*) beschäftigt sich mit der visuellen und interaktiven Gestaltung von Benutzeroberflächen. Es umfasst die Gestaltung von Buttons, Formularen und anderen visuellen Komponenten, sowie das Feedback dieser Komponenten, wie zum Beispiel die Rückmeldung eines fehlgeschlagenen Logins. Zusammengefasst beschäftigt sich UX damit, wie man eine Anwendung verwendet und UI damit, wie die Benutzeroberfläche der Anwendung aussieht.⁵ Funktionen der Schach-App sind unter anderem das Registrieren und Einloggen, das Versenden, Annehmen und Ablehnen von Freundschaftsanfragen, das Zuschauen bei laufenden Spielen, das Herausfordern von Freunden zu Schachspielen und natürlich das Spielen von Schachpartien mit verschiedenen Einstellungsmöglichkeiten der Schach Uhren und einem Chat selbst. Dabei wird besonderer Wert auf die Verwendung moderner Web-Technologien wie React, Node.js, Socket.IO, Redis und PostgreSQL gelegt, um eine optimale Benutzererfahrung und Skalierbarkeit zu gewährleisten. Darüber hinaus soll die Arbeit einen Überblick über die technischen Herausforderungen und Lösungen im Zusammenhang mit der Implementierung einer solchen Schach-App bieten.

1.3 Aufbau der Arbeit

Diese Bachelorarbeit gliedert sich in sechs Hauptkapitel, die jeweils unterschiedliche Aspekte der Entwicklung und Implementierung der Schach-App behandeln.

Im ersten Kapitel, der *Einleitung*, werden die Motivation für die Entwicklung der Schach-App, die Zielsetzung der Arbeit und der Aufbau der Arbeit selbst vorgestellt.

Das zweite Kapitel, *Theoretische Grundlagen*, erläutert die Grundlagen von Schach als Spiel sowie die verwendeten Web-Technologien wie Node.js, Express, Socket.io, React und PostgreSQL, die für das Verständnis der nachfolgenden Kapitel wichtig sind.

Im dritten Kapitel, *Systemarchitektur*, wird die Gesamtarchitektur der Schach-App beschrieben, einschließlich der Unterteilung in Frontend und Backend, der Datenbankstruktur und der Kommunikation zwischen den verschiedenen Komponenten.

Das vierte Kapitel, *Implementierung*, geht auf die praktische Umsetzung der Schach-App ein, indem es die Entwicklungsprozesse für das Frontend und das Backend sowie die Integration der Datenbanken erläutert.

Das fünfte Kapitel, *Tests und Evaluation*, behandelt die verschiedenen Tests, die durchgeführt wurden, um die Funktionalität, Usability, Performance und Sicherheit der Schach-App zu bewerten.

Im abschließenden sechsten Kapitel, *Fazit und Ausblick*, werden die Ergebnisse der Arbeit zusammengefasst, eventuelle Limitationen diskutiert und mögliche Erweiterungen und Weiterentwicklungen für die Schach-App vorgeschlagen.

Die Arbeit endet mit dem *Anhang*, der zusätzliche Grafiken und die Liste der verwendeten Literatur enthält.

⁵Robbins [2012]

2. Theoretische Grundlagen

2.1 Schach

Schach ist ein strategisches Brettspiel für zwei Spieler, welches auf einem quadratischen Spielfeld mit 64 Feldern gespielt wird. Jeder Spieler beginnt mit 16 Figuren und das Ziel des Spiels ist es, den König des Gegners schachmatt zu setzen, indem man ihn bedroht, ohne dass der Gegner den Angriff verhindern kann.

Wie sich welche Figuren bewegen und andere Figuren schlagen erkläre ich nicht explizit, lediglich zwei Sonderregeln des Schach werde ich genauer erklären, da diese bei der Umsetzung des Spiels gesondert gehandhabt werden müssen.

Die erste ist das so genannte *en passant*-Regel. Dabei ist es einem Bauern möglich einen gegnerischen Bauer diagonal zu schlagen, falls dieser zwei Felder gezogen ist und nun auf der gleichen Höhe wie der eigene Bauer steht (siehe Abbildung 2).

Die zweite Zusatzregel ist die *Bauernumwandlung*. Sie besagt, dass falls ein Bauer die gegnerische Grundreihe erreicht, dieser Bauer in eine Dame, einen Springer, einen Turm oder einen Läufer umgewandelt werden kann (siehe Abbildung 3).

2.2 Web-Technologien

2.2.1 Node.js und Express

2.2.2 Socket.io

2.2.3 React

2.2.4 PostgreSQL

3. Systemarchitektur

Die Anwendung ist in zwei Hauptkomponenten unterteilt: das Frontend und das Backend. Das Frontend ist für die Darstellung der Benutzeroberfläche und die Interaktion mit dem Benutzer verantwortlich, während das Backend die Spiellogik, die Verwaltung der Benutzerdaten und die Echtzeit-Kommunikation zwischen den Spielern steuert.

Die Anwendung verwendet moderne Web-Technologien, um eine reaktive und benutzerfreundliche Oberfläche zu schaffen. Das Frontend basiert auf dem React-Framework¹, das es ermöglicht, wiederverwendbare Komponenten zu entwickeln und den Anwendungsstatus effizient zu verwalten. Das User-Interface basiert auf Chakra UI², einem modernen und flexiblen Komponenten-Bibliothekssystem, das die Entwicklung von responsiven und zugänglichen Benutzeroberflächen erleichtert. Die Benutzerführung und die Kommunikation zwischen den React-Komponenten sind so gestaltet, dass sie eine nahtlose und intuitive Benutzererfahrung bieten.

Auf der Backend-Seite wird Node.js³ mit dem Express-Framework verwendet, um einen leistungsstarken und skalierbaren Server bereitzustellen. Die API-Endpunkte und die Kommunikation mittels WebSockets ist so konzipiert, dass sie den Anforderungen der verschiedenen Frontend-Komponenten gerecht werden und die Kommunikation zwischen Frontend und Backend erleichtern. Die Echtzeit-Kommunikation zwischen den Spielern wird mit Hilfe von Socket.io⁴ ermöglicht, einer Bibliothek, die bidirektionale Kommunikation über WebSockets unterstützt. Für die Speicherung und Verwaltung der Benutzerdaten zum Anmelden wird eine PostgreSQL⁵-Datenbank verwendet, die aufgrund ihrer Leistungsfähigkeit und Flexibilität ausgewählt wurde. Freundeslisten und Daten aktiver Spiele werden in einer Redis⁶-Datenbank gespeichert, die sich durch hohe Leistung und niedrige Latenz auszeichnet, insbesondere bei Lese- und Schreibvorgängen. Redis, eine In-Memory-Datenstruktur, eignet sich ideal für Anwendungen, bei denen schnelle Zugriffszeiten und Skalierbarkeit wichtig sind. Die Kombination von PostgreSQL und Redis ermöglicht eine effiziente Verwaltung sowohl persistenter als auch flüchtiger Daten und fördert eine optimale Benutzererfahrung.

¹Meta Platforms [2023]

²Adebayo [2023]

³Foundation [2023]

⁴Socket.IO [2023]

⁵Group [2023]

⁶Ltd. [2023]

3.1 Frontend

3.1.1 React-Komponenten

3.1.2 Benutzerführung

3.1.3 Kommunikation mit dem Backend

3.2 Backend

3.2.1 API-Endpunkte

3.2.2 Datenbankstruktur

PostgreSQL

Redis

3.2.3 Echtzeit-Kommunikation

4. Implementierung

4.1 Frontend-Entwicklung

4.2 Backend-Entwicklung

4.3 Datenbank-Integration

5. Tests und Evaluation

5.1 Funktionalitätstests

5.2 Usability-Tests

5.3 Performancetests

5.4 Sicherheitstests

6. Fazit und Ausblick

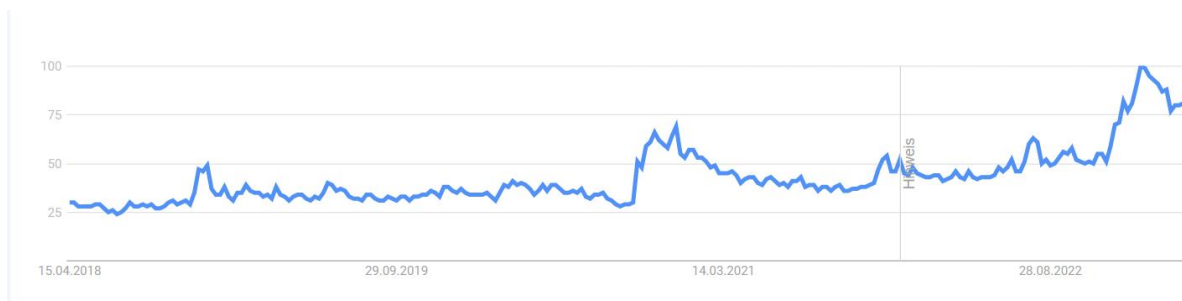
6.1 Zusammenfassung der Ergebnisse

6.2 Limitationen

6.3 Potenzielle Erweiterungen und Weiterentwicklung

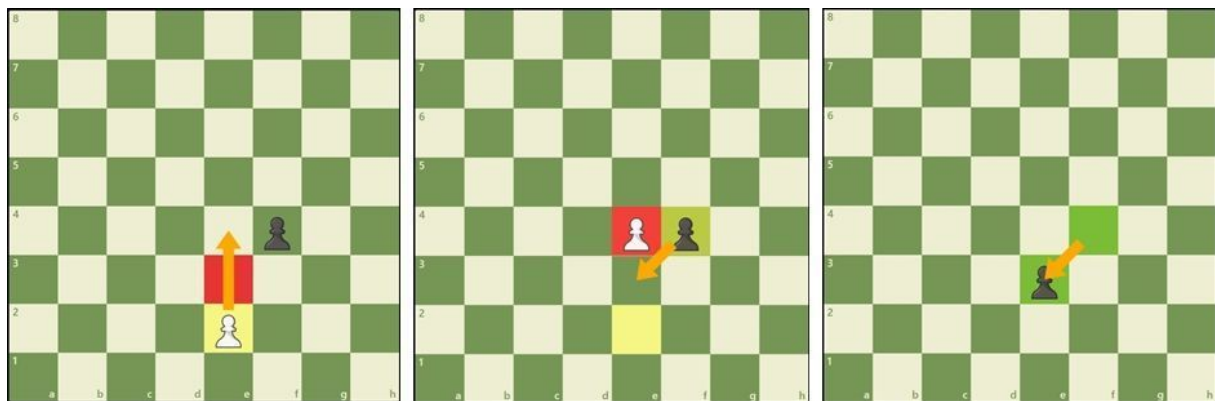
Abbildungsverzeichnis

- 1 Relatives Suchinteresse des Wortes *Chess* auf Google in den letzten 5 Jahren. 11
- 2 Die Zusatzregel *en passant* 11
- 3 Die Zusatzregel *Bauernumwandlung* 12



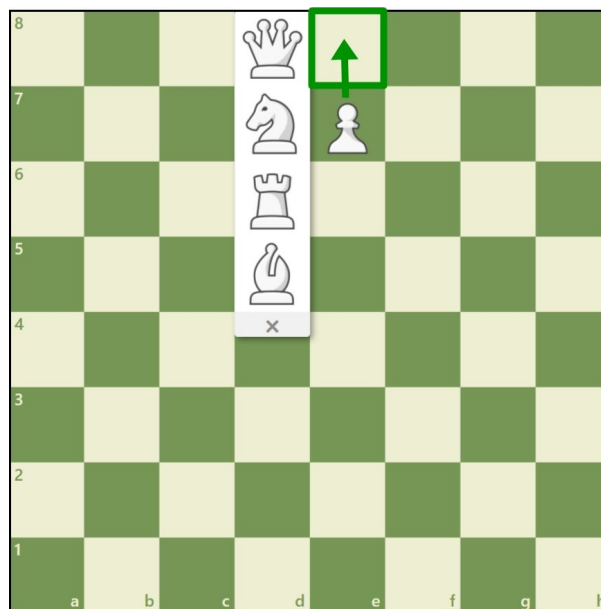
Quelle: <https://trends.google.de/>

Abbildung 1: Relatives Suchinteresse des Wortes *Chess* auf Google in den letzten 5 Jahren.



Quelle: <https://www.chess.com/de/schachregeln>

Abbildung 2: Die Zusatzregel *en passant*



Quelle: <https://www.chess.com/de/schachregeln>

Abbildung 3: Die Zusatzregel *Bauernumwandlung*

Literaturverzeichnis

Segun Adebayo. Chakra ui - a simple, modular and accessible component library for building react applications, 2023. URL <https://chakra-ui.com/>.

chess.com. We've reached 3000000000 live chess games, 2018. URL <https://www.chess.com/forum/view/general/weve-reached-3000000000-live-chess-games>.

OpenJS Foundation. Node.js - an open-source, cross-platform javascript runtime environment., 2023. URL <https://nodejs.org/>.

The PostgreSQL Global Development Group. Postgresql - the world's most advanced open source relational database., 2023. URL <https://www.postgresql.org/>.

Astrid Hansen. Wie ein schachspiel zum wettstreit der systeme wurde. URL <https://www.geo.de/magazine/geo-epoche/19054-rtkl-schachweltmeisterschaft-wie-ein-schachspiel-zum-wettstreit-der>.

Redis Ltd. Redis - the open source, in-memory data store used by millions of developers as a database, cache, streaming engine, and message broker., 2023. URL <https://redis.io/>.

Inc. Meta Platforms. React – a javascript library for building user interfaces, 2023. URL <https://react.dev/>.

Jennifer Niederst Robbins. *Learning Web Design*. O'REILLY, 2012. ISBN 978-1-449-31927-4.

Socket.IO. Socket.io - bidirectional and low-latency communication for every platform., 2023. URL <https://socket.io/>.

Sportschau. Nach schach-eklat – ermittlungen gegen niemann und carlsen, 2022. URL <https://www.sportschau.de/schach/magnus-carlsen-hans-niemann-ermittlungen-100.html>.

Antonius van der Linde. *Geschichte und Litteratur des Schachspiels, Erster Band*. 1874.