

Titel

Anastasia Chernysheva
a.chernysheva@oth-aw.de

Jakob Götz
j.goetz@oth-aw.de

Ardian Imeraj
a.imeraj@oth-aw.de

Patrice Korinth
p.korinth@oth-aw.de

Philipp Stangl
p.stangl1@oth-aw.de

Zusammenfassung—TODO

I. EINLEITUNG

Das erste Videospiel *Tennis for Two* aus dem Jahr 1958 ebnete den Weg für die Spieleindustrie. Über Jahrzehnten hinaus kamen immer mehr verschiedene Spiele auf den Markt, die sich in ihrer Art und Weise unterscheiden. Mit den heutigen Videospielen ist es möglich, sich mit Spielern aus der ganzen Welt zu messen oder gemeinsam zu spielen. Aus den unzählig veröffentlichten Videospielen ist eine breite Palette an Genres entstanden. Dabei wird in der Spieleindustrie grob in sechs Kategorien unterschieden, hierbei zählen die Strategie- und Action-Spiele zu den beliebtesten Genres [1]. Zu den anderen vier Kategorien zählen hier die Rollen-, Adventure-, Simulationen- und Sonstigespiele. Jedes dieser Kategorien besitzt weitere Unterkategorien, die sich in ihrer Art und Weise unterscheiden.

Um einen Überblick zu erhalten, wie die Spiele zusammenhängen, präsentieren wir hier eine plattformunabhängige Anwendung, welche die Spiele in einem Graphen darstellt. In Abschnitt II des Konzeptpapiers werden bereits bestehende Anwendungen vorgestellt, die sich mit der Kategorisierung von Spielen und Darstellung von Graphen beschäftigen. In Abschnitt III werden die Anforderungen in Form von User Stories beschrieben. Zum Schluss wird auf die zu verwendenden Technologien zur Umsetzung des Projekts eingegangen, welche in Abschnitt IV näher beschrieben werden.

II. VERWANDTE ARBEITEN

Die Anwendung Obsidian erstellt aus von Nutzern kreierten Markdown Dateien eine Graphoberfläche. Über Verlinkungen zwischen den Markdown Dateien wächst der Graph an. Der Nutzer kann den Graph filtern oder gewisse Teile daraus hervorheben. Diese Anwendung zeigt hervorragend, wie eine Visualisierung eines Graphen möglicherweise aussehen könnte. Die Website igdb.com gibt einen Eindruck, was der Nutzer von einer Spieldatenbank erwarten kann. Der Funktionsumfang und die Spieleanzahl ist groß, sodass das Suchergebnis meist zufriedenstellend ist. Des Weiteren besitzt die Website eine gut dokumentierte Entwickler-API. Dies ermöglicht es Entwickler, Daten in Form von JSON, REST oder ProtoBuf zu erhalten. Jedoch sind die Daten, welche IGDB bereit stellt nicht für die LOD-Cloud aufbereitet. Des Weiteren ist die Oberfläche der Website sehr klassisch aufgebaut, wodurch kein weiteres Interesse geweckt wird. Hier setzt das Projekt von Team Rot an. Es wird eine Web-Applikation entworfen, welche die Daten anreichert, damit diese in die LOD-Cloud überführt werden können. Zusätzlich wird eine Graphoberfläche entworfen,

welche dazu einlädt, artverwandte Spiele auch ohne Filteroptionen zu entdecken.

III. ANFORDERUNGEN

In der Anforderungsanalyse wurden zwei Stakeholder identifiziert: Benutzer und Entwickler. Deren Anforderungen werden in diesem Abschnitt in Form von User Stories beschrieben.

A. Graphoberfläche

Als Benutzer möchte ich in eine Graphoberfläche, welche visuell Verbindungen zu anderen Spielen aufzeigt. Somit kann ich direkt ähnliche Spiele, welche mir ebenfalls gefallen könnten, sehen. Akzeptanzkriterien sind:

- Graphoberfläche soll direkt sichtbar sein beim besuchen der Website
- Der Graph soll mit ausreichend vielen Knoten angereichert sein
- Die Oberfläche soll visuell ansprechend sein und zum stöbern einladen

B. Knotendetails

Als Benutzer möchte ich mit einem Klick auf einem Knoten im Graphen weitere Details zu den im Knoten enthaltenen Spiel sehen. Somit erhalte ich ohne die Graphoberfläche zu verlassen die wichtigsten Informationen zu einem Spiel. Akzeptanzkriterien sind:

- Knoten im Graphen sollen anklickbar sein
- Der geklickte Knoten öffnet ein Feld mit weiteren Informationen zu den ausgewählten Knoten
- Die wichtigsten Informationen befinden sich zusammengefasst im geöffneten Feld

C. Filter

Als Benutzer möchte ich mit einer Filteroption Knoten und somit Spiele im Graphen visuell hervorheben. Dadurch kann auch in einem großen Graphen der Nutzer eine Gruppe von Spielen problemlos finden. Akzeptanzkriterien sind:

- diverse Filtereinstellungen
- Ergebnis sollte gut erkennbar hervorgehoben werden
- Ausreichend viele Filteroptionen

D. Twitter-Cards für das jeweilige Spiel

Als Benutzer möchte ich Twitter-Cards zu dem Spiel sehen, welches aktuell betrachtet wird. Akzeptanzkriterien sind:

- Aktuelle Twitter-Cards geordnet anzeigen
- Twitter-Cards fügen sich formschön in die Website-Oberfläche ein
- Es existiert eine Verlinkung zum jeweiligen Twitter Beitrag

E. (Optional) Detailseite

Als Benutzer möchte ich eine Detailseite zu einem jeweiligen Spiel, welche über den Knoten im Graphen erreicht wird. Somit muss ich als Nutzer nicht externe Internetseiten für weitere Informationen besuchen. Akzeptanzkriterien sind:

- Alle Details eines Spiels werden aufgelistet
- Spiel wird vorgestellt

F. (Optional) Suche

Als Benutzer möchte ich ein Spiel suchen können, da bei einem größeren Graphen nur schwierig ein spezielles Spiel gefunden werden kann. Akzeptanzkriterien sind:

- Eingabefeld für Suche
- Weiterleitung direkt zum Knotenpunkt, welcher das gewünschte Spiel beinhaltet
- Grafische hervorhebung des Knotenpunkts

G. Testabdeckung

Als Entwickler möchte ich eine ausreichend hohe Testabdeckung, um Fehler frühzeitig zu erkennen. Akzeptanzkriterien sind:

- Testabdeckung von mindestens 50%
- Sowohl Frontend- als auch Backend sind mit den Tests abgedeckt

H. (Optional) API-Nutzung

Als Entwickler möchte ich erkennen können, in welchen Umfang meine API genutzt wird, damit ich Rückschlüsse auf die Auslastung und Interessen erhalte. Akzeptanzkriterien sind:

- Wie häufig wird die API angesprochen
- Welche API-Endpunkte werden am häufigsten genutzt

I. (Optional) Einsehen meist aufgerufener Spiele

Als Benutzer möchte ich, dass mir die meist aufgerufenen Spiele angezeigt werden, damit ich wissen kann, welche Spiele in der Spielebranche momentan gefragt sind. Akzeptanzkriterien sind:

- Ein eigener Bereich für zuletzt aufgerufene Spiele auf der Startseite
- Spiele werden jeweils nur einmal angezeigt
- Hierarchische Anordnung nach Anzahl der Aufrufe in bestimmter Zeit

J. (Optional) Einsehen von Bewertungen

Als Benutzer möchte ich Bewertungen zu dem jeweiligen Spiel einsehen können, damit ich mir ein Bild über die Qualität des Spiels verschaffen kann. Akzeptanzkriterien sind:

- Eine Anzeige in Form von Sternen oder eines Prozentsatzes
- Eine Möglichkeit Spiele anhand der Bewertungen zu sortieren

IV. METHODEN

Ziel bei der Entwicklung und bei der späteren Verwendung ist eine plattformunabhängige Anwendung, die eine Schnittstelle für die *Linked Open Data Cloud* als auch der Bedienoberfläche der Anwendung bereitstellt. Daraus ergeben sich folgende technische Schlüsselbausteine:

Für die Repräsentation der semantischen Daten im Frontend wird das Framework Svelte verwendet. Die Daten werden voraussichtlich in einer Ontotext GraphDB Instanz persistiert. Im Backend wird FastAPI für eine REST-Schnittstelle und RDFLib zum arbeiten mit RDF verwendet. Die Kommunikation zwischen Frontend und Backend wird über eine RESTful-API abgewickelt. Um eine fehlerfreie Anwendung zu entwickeln, werden geeignete Test-Frameworks für Frontend als auch Backend selektiert und verwendet. Die einzelnen Komponenten der Anwendung werden mittels Docker *containerisiert*.

LITERATUR

- [1] Statistica. "Statistica." [Online]. (2022), Adresse: <https://de.statista.com/prognosen/999755/deutschland-beliebteste-genres-bei-videospielen>.