

Entwicklungsprojekt

Wintersemester 2019/20

David Buglowski & Moritz Langer

Nutzungsproblem : Rechenschwäche

- Anzahl von betroffenen Kindern mind. so hoch wie bei LRS
- Rechenschwäche deutlich weniger Publik
- In manchen Bundesländern nicht als Krankheit angesehen

Was ist Rechenschwäche (Dyskalkulie)?

- Problem: Repräsentation und Verarbeitung von Zahlen

- Visuelle Darstellung
- Räumliche & semantische Größenrepräsentation
- Verbale Zahlenrepräsentation
- Strategisches & konzeptionelles Wissen

$$\begin{array}{r} 1+2 = ? \\ 12 \quad 3 \quad 21 \\ \quad \quad \epsilon \end{array}$$

Dyskalkulie beschreibt eine Beeinträchtigung von mathematischen Denkprozessen. Dieses Denken entwickeln Kinder schon im jungen Alter und erweitern dies in der Grundschule. Die Dyskalkulie bremst diese Entwicklung erheblich ab, weshalb einfache Grundrechenarten und die Basis der mathematischen Logik fehlen kann. Die Kinder verstehen Zahlen als Symbole und nicht als eine Mengenangabe. Da somit zu Beginn der Schulkarriere der wichtigste Baustein der Mathematik fehlt, hat das schwerwiegende Folgen für die Zukunft.. Rund 3-7% der Bevölkerung ist davon betroffen.

<https://www.schlaudino.com/rat-und-hilfe/glossar/dyskalkulie.php>

Folgen der Rechenschwäche

- zunehmende Probleme bei komplexeren Problemen
- 20% nicht bereit für weiterführende Schulen
- Seit 2011 verschlechtert (TIMSS-Studie 2016)

=> Behebung des Defizits = **gesellschaftliches Interesse**

Quelle: <https://www.iflw.de/blog/dyskalkulie-rechenschwaecher/timss-studie-grundschueler-mit-mathe-problem/>

Spezifizierung des Problems

- Neben Therapie, auch Umgang mit Zahlen wichtig
- Spielerischer Umgang ideal zum Lernen für Kinder
- Nachteil Brettspiele: Betreuung durch Erwachsenen

Für das behandeln einer Rechenschwäche ist eine Therapie notwendig, aber nebenbei muss der Umgang mit Zahlen und Mengen erlernt werden. Hierfür sind Spiele sehr gut geeignet, da Kinder durch sie besser lernen als durch andere Methoden.

Einziger Nachteil an herkömmlichen Spielen ist, dass es immer eine Person geben muss die die Ergebnisse kontrolliert, was dazu führt, dass ein Kind nicht alleine lernen könnte.

Unsere Idee

- 3D-Spiel zur Unterstützung des Lernprozesses
- Auf Basis von bereits eingesetzten Brettspielen
- Selbstkontrollierbarkeit durch Algorithmen

Vorgehen

- Recherche, Problemanalyse
- Grundmodelle (Domänenmodell, Stakeholder)
- Funktionalitätsanalyse (Anforderung -> Use Case)
- Analyse der Brettspiele
- Erste Spezifikation eines PoC
- Erstellen eines ersten PoC
- Ausarbeiten des Systems mit Abwägung der Alternativen

Hauptperspektive

| Visual Computing | Social Computing |
|-------------------------------|---|
| Grafische Darstellung | Gamification |
| Systemumsetzung als Spiel | E-Learning |
| Hauptinteresse im Bereich CGA | Interaktion von Gesellschaft und Informatik |

Als Hauptperspektive haben wir Visual Computing ausgewählt, da wir unser System als ein Spiel umsetzen wollen und somit eine Graphische Oberfläche erstellt wird, die für das Produkt von großer Relevanz ist. Für Social Computing würde aber auch das Element des Spiel sprechen, da wir mithilfe von Gamification das Spielerlebnis verbessern können. Letztendlich haben wir uns aber für visual entschieden, da unser Interesse eher im Bereich der Computergrafik liegt.

Nebenperspektive

| Social Computing | Mensch Computer Interaktion |
|---|--|
| Gamification | Interaktion von Menschen mit Computern |
| E-Learning | Starke Analyse der Systemnutzung |
| Interaktion von Gesellschaft und Informatik | |

Als Nebenperspektive kam natürlich social Computing, aus den vorrangegangenen Gründen, in frage. Zusätzlich wäre Mensch Computer Interaktion ebenfalls eine Option gewesen, da unser System ein großes Augenmerkmal auf die Interaktion des Benutzers und unseres Systems ist.

PC oder Tablet?

| Tablet | PC |
|---|---------------------------|
| Mobil | Mehr Geräte in Haushalten |
| Einfachere Bedienung | |
| Berührung | |
| Einfaches & schnelles Starten des Systems | |

Für die weitere Planung unseres System stellte sich die Frage, auf welchen Endgeräten es laufen soll. Die Überlegung war, es entweder auf einem PC oder einem Tablet laufen zu lassen. Entschieden haben wir uns dann für das Tablet bzw. Android, da es für ein Kind dank des Touchscreen eine bessere Bedingung anbietet, zudem ist es auch einfacher zu starten, sodass ein Kind dies problemlos selbstständig tun könnte und außerdem ist ein Tablet mitnehmbar und nicht wie ein PC ortsgebunden ist.

System umsetzen, wie?

- Vorteile einer Game Engine
 - Enormer Aufwand an Programmierung entfällt
 - Zeitaufwand wird kleiner -> mehr Zeit für andere Punkte
 - Bieten viele nützliche Funktionen zur Erstellung von Spielen
 - Bringen eine Vielzahl von Assets mit
 - Ermöglicht einfachen Export auf etliche Plattformen

<https://viscircle.de/alles-was-sie-zu-game-engines-wissen-muessen/>

Titel

| Unity | Unreal Engine | Amazon Lumberyard |
|--|--|------------------------------------|
| Schnelle & einfache Benutzeroberfläche | Benötigt mehr Zeit zum Kennenlernen | Basiert auf CryEngine => komplexer |
| Enthält Android | Enthält Android | Enthält Android |
| Hohe Anzahl an Assets | Asset-Store enthalten | Asset-Store enthalten |
| Projekt benötigt nicht das Maximum | -+ schönere mögliche Grafik (bei mehr Aufwand) | Beste mögliche Grafik |
| C# / JavaScript | C++ | Lua & C++ |
| | | AWS support |
| Große Community | Große Community | Noch recht klein |

Wir haben uns drei größtenteils kostenlose Game Engines ausgesucht und verglichen. Lumberyard haben wir als ersten Kandidaten ausgeschlossen, da wir uns bei allen drei Engines als Anfänger rantasteten und sich dies bei der kleinen Community, die dort, aufgrund der noch recht neuen (im Vergleich zu den anderen beiden Giganten) Engine, leider sehr schwierig gestaltet.

Da Unity und Unreal größtenteils kostenlos sind und somit nicht direkt aus der Auswahl fallen, haben wir uns dazu entschieden die beiden großen angebotenen Game Engines zu vergleichen und unsere Auswahl dementsprechend zu treffen.

Unity zeichnet eine reaktionsschnelle und einfach zu bedienende Benutzeroberfläche aus, welche eine längere Einarbeitungszeit vorbeugt und somit Anfängerfreundlicher ist. Unreal Engine hat da zwar nachgebessert, dennoch ist sie ziemlich komplex und in vielerlei Hinsicht zeitaufwendiger (Import und Speichern von Assets dauert lang, einfache Aufgaben benötigen zusätzliche Schritte).

Im Vergleich der unterstützten Plattformen ist Unity viel präsenter. Zwar unterstützen beide Android und somit unsere gewählte Plattform, wäre es dennoch nicht unvorteilhaft eine größere Auswahl für eine mögliche spätere Ausbreitung zu haben.

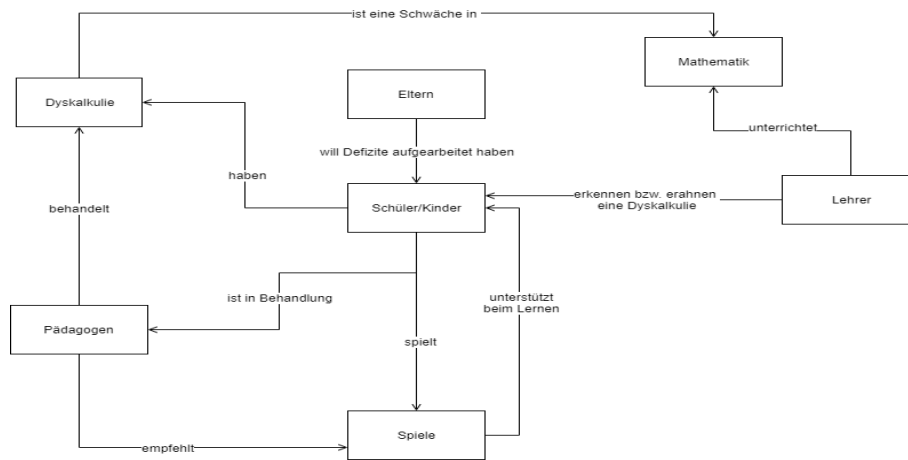
Beide Engines verfügen über eigene Asset Stores, welche prall mit Modellen gefüllt sind. Hier geht Unity mit einer größeren Auswahl als Sieger hervor.

Unreal ermöglicht grafisch zwar ein viel schöneres und ansprechenderes Endprodukt, jedoch bedeutet dies auch viel mehr Aufwand, den man als Anfänger dann betreiben muss.

Beim Punkt Programmiersprache verwendet Unreal C++ und Unity JavaScript bzw. C#. Da unser Studium bisher größtenteils auf Java aufgebaut hat und JavaScript in GDW vorkam, ist hier Unity leicht im Vorteil.

<https://viscircle.de/unreal-engine-vs-unity-welche-software-sollten-gamedeveloper-waehlen/>

Domänenmodell



Stakeholderanalyse

| Bezeichnung | Beziehung zum System | Erfordernis | Erwartung |
|--------------|----------------------|---|---|
| Schüler/Kind | Anspruch | Pädagogisch richtiges lernen durch das System | |
| | Interesse | | Verbesserung der Dyskalkulie |
| | Interesse | | Spaß beim ausführen des Spieles |
| Eltern | Anspruch | Eigenständiges lernen/spielen des Kindes | |
| | Interesse | | Verbesserung der Dyskalkulie des Kindes |
| Pädagogen | Interesse | Richtigkeit der Lernspiele | Unterstützung des Patienten |
| | Anspruch | | |
| Lehrer | Interesse | | Helfen des Schülers zur Aufarbeitung seines Defizites |

Die Spalte Objektbereich wurde von uns bewusst weggelassen, da alle unsere Stakeholder in den Gleichen Bereich fallen würden.

Use Cases

| User's purpose | System response |
|--|---|
| Benutzer möchte ein spezifisches Spiel auswählen | System bietet dem Benutzer verschiedene Spiele an |
| | System startet ausgewähltes Spiel |

| User's purpose | System response |
|--|---|
| Der Benutzer möchte die Verbesserung/Verschlechterung einsehen | System analysiert die gespeicherten Ergebnisse der verschiedenen Sessions |
| | System zeigt die Daten der verschiedenen Sessions an |

| User's purpose | System response |
|--|--|
| | System speichert Ergebnisse während des Spielens |
| Benutzer möchte die Möglichkeiten haben, die Ergebnisse einsehen zu können | System zeigt die gespeicherten Daten an |

| User's purpose | System response |
|---|--|
| | System speichert die Spielzeit während einer Session |
| Benutzer möchte die Möglichkeiten haben, die Spielzeiten einsehen zu können | System zeigt die gespeicherten Daten an |

Use Cases

| User's purpose | System response |
|---|---|
| Benutzer möchte für seine Leistung und Aufwand belohnt werden | System erstellt nach Bestehen bestimmter Aufgaben visuelle Belohnungen(Achievements, Medaillen,Pokale,...) und zeigt diese an |

| User's purpose | System response |
|---|--|
| Benutzer möchte Fortschritte bezüglich seiner Defizite machen | System erhöht die Schwierigkeit, wenn bestimmte Aufgaben problemlos gelöst wurden. |

| User's purpose | System response |
|---|---|
| Benutzer möchte Fortschritte in der Verbesserung visuell wahrnehmen | System berechnet den prozentualen Fortschritt der gelösten Aufgaben einer Schwierigkeitsstufe |
| | System zeigt diesen als individuelles Level(Rang) aus |