

Jasper Hustedt
7383407
Informatik (BA)
8. Semester
s5414157@rz.uni-frankfurt.de
jasper.hustedt@web.de

Bachelorarbeit

Virtualisierung von Mindmaps in Va.Si.Li-Lab

**Erstellung und Evaluierung eines vernetzten Lehrprogramms auf
Basis von Mindmapping in Virtual und Augmented Reality in
Va.Si.Li-Lab**

Jasper Hustedt

Abgabedatum: 23.01.2024

Institut für Informatik
Text Technology Lab
Prof. Dr. Alexander Mehler

Erklärung zur Abschlussarbeit

gemäß §35, Abs. 16 der Ordnung für den Bachelorstudiengang Informatik vom 17. Juni 2019:

Hiermit erkläre ich

(Nachname, Vorname)

Die vorliegende Arbeit habe ich selbstständig und ohne Benutzung anderer als der angegebenen Quellen und Hilfsmittel verfasst.

Ich bestätige außerdem, dass die vorliegende Arbeit nicht, auch nicht auszugsweise, für eine andere Prüfung oder Studienleistung verwendet wurde.

Zudem versichere ich, dass alle eingereichten schriftlichen gebundenen Versionen meiner vorliegenden Bachelorarbeit mit der digital eingereichten elektronischen Version meiner Bachelorarbeit übereinstimmen.

Frankfurt am Main, den

Datum

Unterschrift der/des Studierenden

Zusammenfassung

Durch den Fortschritt digitaler Lernmethoden und das Ansteigen der Popularität von Virtual Reality in den letzten Jahren stieg die Nachfrage nach einer Kombination eben dieser beiden Fachbereiche stark an. Zugängliche Applikationen dieser Art sind bis jetzt jedoch eher selten, und somit kaum nutzbar für den alltäglichen Gebrauch in Lehreinrichtungen. Viele Lehrende besitzen nicht genug Fachkenntnisse, um für eine Einführung von Methoden der Virtual Reality in ihren Unterricht zu sorgen. Bereits existierende Applikationen sind oft entweder nicht von den Lehrenden selbst anpassbar oder besitzen keine Schnittstelle, die es einfach genug machen würde, dass diese anderen Mitteln vorgezogen werden würden.

Mindmaps sind neben Virtual Reality eine Art des Lehrens, die verstärkt die Kreativität und Interaktion mit zu lernenden Dingen fördern.

In dieser These wurde im Folgenden eine Virtual Reality Applikation namens VA.SI.LI-MINDMAP entwickelt, in der Aufgaben in einem Mindmapformat auf einer Website von Lehrenden erstellt und in einer Virtual Reality Umgebung von Lernenden gelöst werden können.

Daraufhin wurde eine Evaluation dieser vernetzten Applikation anhand von 15 Probanden durchgeführt. Die Probanden wurden befragt und die Befragung ausgewertet.

Inhaltsverzeichnis

1	Einleitung	8
1.1	Einleitung ins Lehren und Lernen mit virtuellen Interaktionen	8
1.2	Ähnliche Arbeiten	9
2	Basis / Ressourcen	11
2.1	Spark	11
2.2	MongoDB	11
2.3	Unity	11
2.4	Va.Si.Li-Lab	11
3	Grundlegende Komponenten und Datenfluss	13
4	Anwendungsfälle	15
4.1	Allgemeiner Unterricht	15
4.2	Sprachenlernen	15
4.3	Erlernen praktischer Aufgaben	15
4.4	Unterhaltung	16
4.5	Abbilden von Datenstrukturen	16
5	Website	17
6	Schnittstelle/Server und Datenbank	20
7	Datenstruktur der Schnittstelle	21
7.1	Umwandlung der „image“ und „audio“ Variablen	22
8	Virtual Reality Komponente	24
8.1	Menü	24
8.2	Mindmap-Objekt	24
8.3	Umwandlung in Aufgabenform	25
8.4	Interaktion	25
8.5	Anordnung des Mindmaps	26
8.6	Handmenüs	26
8.7	Verbinden von Knoten	27
8.8	Ermittlung der Validierung	29
8.9	Erzeugung der Verbindungsbeschriftungen	32
8.10	Berechnung der nächsten Knoten der Aufgabe mit automatischer Knotenverbindung	32

9	Evaluation von VA.SI.LI-MINDMAP	34
9.1	Aufbau	34
9.2	Auswertung	34
9.3	Anmerkungen/Probleme der Evaluationsteilnehmenden	35
9.3.1	VR-Komponente	35
9.3.2	Website	35
10	Diskussion und zukünftige Verbesserungen	37
10.1	Modularität	37
10.2	Weitere Datenquellen	37
10.3	NLP Methoden	37
10.4	Zusätzliche Datentypen	37
10.5	Erweitertes Erstellen eigener Mindmaps	38
10.6	Einbindung der kollaborativen Komponente von Va.Si.Li-Lab	38
10.7	Erweiterung von VA.SI.LI-MINDMAP für Kontexte außerhalb des Lernens	38
11	Fazit	39
12	Anhang	40
	Literatur	45

Abbildungsverzeichnis

3.1	Datenflussdiagramm von VA.SI.LI-MINDMAP	13
5.1	Ein Ausschnitt der gesamten Website mit erstellter Aufgabe	18
5.2	Ein Beispiel für eine Aufgabe mit gerichteten Verbindungen	18
5.3	Ein Beispiel für eine Aufgabe mit ungerichteten Verbindungen und Abbildungen	19
5.4	Ein Beispiel für ein Element, dass einen Knoten darstellen soll, mit allen drei Inhaltsarten Text, Bild und Audio	19
8.1	Menü zum Auswählen einer Aufgabe	24
8.2	Ein mit Text, Bild und Audio gefüllter Knoten	26
8.3	Beispiel eines Mindmaps von vorne betrachtet	27
8.4	Ein von außerhalb betrachtetes Mindmap, mit Knoten und Verbindungen über den Großteil der verfügbaren Fläche verteilt. Die blauen Objekte an den Seiten stellen den Außenrand dar und sind der Interaktionspunkt für das bewegen des gesamten Mindmaps	28
8.5	Menü mit Knotenauswahl auf der linken Hand	29
8.6	Menü mit Informationen zur Aufgabe und Tasten zum Wechseln der Aufga- benseite und der Validierung auf der rechten Hand	30
8.7	Gerichteter Graph aus dem Aufgabentyp mit automatischer Verbindung . . .	30
8.8	Ungerichteter Graph	31
8.9	Korrektur der Aufgabenseite eines Mindmaps mit einheitlich richtig gesetzten Verbindungen	31
8.10	Korrektur der Aufgabenseite eines Mindmaps mit richtig gesetzten Verbin- dungen, falsch gesetzten Verbindungen und richtig gesetzten Verbindungen mit falscher Verbindungsbeschriftung	32
9.1	Auf zwei Nachkommastellen gerundeter Punktedurchschnitt pro Frage	36
12.1	Fragebogen zur Evaluation	41
12.2	Klassendiagramm der VR-Komponente	42
12.3	Klasse MindMapController	43
12.4	Klasse Node	43
12.5	Klasse NodeConnection	44

Tabellenverzeichnis

1.1	Eigenschaftsvergleich verschiedener Applikationen	10
9.1	Bewertungen der Probanden	35

1 Einleitung

1.1 Einleitung ins Lehren und Lernen mit virtuellen Interaktionen

Die digitalen Möglichkeiten und Werkzeuge im Bereich des Lernens haben sich in den letzten Jahren vervielfältigt (Veletsianos 2016). Vor allem seit Beginn der COVID-19-Pandemie hat ein Ausbau der virtuellen Kommunikation, insbesondere auch in Universitäten, Schulen und anderen Bildungseinrichtungen, stattgefunden (Aditya 2021). Mit verschiedenen Applikationen wurden neue Möglichkeiten für moderne und alternative Lernprozesse geschaffen. Sowohl für Lernende, als auch für Lehrende, ergab dies neben neuen Möglichkeiten jedoch auch neue Probleme, denn diese Kommunikationsmöglichkeiten stellen nur einige von vielen Wegen bereit, konventionellen Unterricht in eine digitale Umgebung zu übertragen. Neue und innovative Lernprozesse, die auf virtuellen Interaktionen basieren, und gleichzeitig aber auch benutzerfreundlich und von den Lehrenden selbst anpassbar sind, wurden jedoch vernachlässigt (Radianti u. a. 2020).

Virtual und Augmented Reality Applikationen beschäftigen sich mit der virtuellen Interaktion eines Benutzers in einer simulierten dreidimensionalen Umgebung mithilfe einer virtuellen Brille und Controllern für die Hände (LaValle 2023 S. 8, 31, 43). Diese stellen eine erweiterte Möglichkeit dar, mit der Lernende ihren Lernprozess interaktiv vertiefen können.

Insbesondere Mindmaps sind eine gute Interaktionsmöglichkeit, um Ideen auszudrücken und Aufgaben zu entwerfen. Mindmaps bestehen aus mit Inhalt gefüllten Elementen, auch Knoten genannt, die miteinander auf unterschiedlichste Weise verbunden werden. Sie fördern selbstständiges Denken und gleichzeitig ein räumliches Bewusstsein. Begriffe, mit denen interagiert werden, sind handfester und mehr als nur Worte auf Papier oder in einem digitalen Dokument, wie in Rezapour-Nasrabad 2019 beschrieben.

Lehren mit Virtual Reality, Augmented Reality oder einer Mischung aus beidem, der sogenannten Mixed Reality, kann einen positiven und verbesserten Lerneffekt auf die Lernenden haben. Grinshkun u. a. haben bei einer Versuchsreihe zur Unterrichtung in Schulklassen mit Augmented Reality festgestellt, dass bei den Schülern eine erhöhte Motivation und erhöhtes Interesse am Unterrichtsmaterial bestand, im Vergleich zu Schülern im konventionellen Unterricht (Grinshkun u. a. 2021).

Bei Saxena und Signer wurde ein Vorteil durch Lernen mit Augmented Reality Mindmaps festgestellt, allerdings nur in Veranstaltungen, die nicht in Präsenz stattgefunden haben (Saxena und Signer 2022).

Während der Entwicklung des Mixed Reality Interfaces „Reality Map“ von Giraudeau und Hachet (Giraudeau und Hachet 2017) wurde eine Versuchsreihe zur Evaluation dieser durchgeführt. Personen, die das Mixed Reality Interface benutzt haben, um ein vordefiniertes Thema zu erlernen, schnitten bei einem folgenden Test über diese Themen signifikant besser ab als

Personen, die ein herkömmliches, digitales Programm verwendeten. Durch einen Vortest und eine Verteilung der Probanden in Gruppen mit gleichstarkem Vorwissen wurde dieses Ergebnis außerdem bekräftigt.

Ähnlich aussagekräftige Werte ermittelten auch Sims und Karnik und Kuřák u. a. Bei Sims und Karnik (Sims und Karnik 2021) wurde die Virtual Reality Mindmap Applikation „VERTAS“ entworfen und anschließend eine Nutzerbefragung durchgeführt. Die Applikation war darauf ausgerichtet, Mindmaps in Aufgabenform bereitzustellen. Die Erstellung der eigentlichen Aufgaben wurde jedoch nicht weiter behandelt. Es wurde lediglich die Benutzbarkeit des Programms erprobt und getestet. Als Hauptargumente für die Benutzung der Applikation wurden die Attraktivität, also auch die Anregung für das Lernen bestimmt.

Kuřák u. a. (Kuřák u. a. 2019) entwickelte eine Applikation, die sich auf die Zusammenarbeit von Mitarbeitenden in einer Virtual Reality Umgebung fokussierte. Die Teilnehmenden meldeten meist eine gute bis exzellente Benutzerfreundlichkeit.

Während es also einige Forschungen und Applikationen im Bereich des Lernens mit Virtual und Augmented Reality gibt, von denen einige sich auf Mindmaps fokussieren, fehlt eine benutzerfreundliche Virtual Reality Applikation für Lernende, die mit einer Oberfläche für Lehrende verknüpft ist, mit denen für den tagtäglichen Unterricht selbstständig und ohne Programmiererfahrung oder Virtual/Augmented/Mixed Reality Erfahrungen Aufgaben erstellt und den Lernenden bereitgestellt werden können. In der folgenden Arbeit wird dies weiter aufgegriffen.

1.2 Ähnliche Arbeiten

Nach umfassender Recherche wurde festgestellt, dass es keine Arbeit gibt, deren Bestandteile mit VA.SI.LI-MINDMAP so sehr übereinstimmen, dass man einen direkten, qualitativen Vergleich jedes einzelnen Bestandteils ziehen kann. Allerdings gibt es Applikationen, die zumindest Aufgaben in Form von Mindmaps in Virtual Reality anbieten, und Applikationen zum Erstellen von Mindmaps in Virtual Reality. Somit wurde im Folgenden ein quantitativer Vergleich auf Basis der Bestandteile zwischen den unter anderem in Kap. 1 erläuterten Applikationen erstellt, in dem Unterschiede und Gemeinsamkeiten der Applikationen festgestellt wurden (siehe Tabelle 1.1). VA.SI.LI-MINDMAP punktet vor allem in Funktionen, die nicht in den anderen Applikationen vorhanden sind. Sowohl den Aspekt der Aufgabenentwicklung außerhalb von Virtual Reality und ohne Programmierkenntnisse, als auch das direkte Hochladen und automatische Umwandeln der Aufgaben in ein für Virtual Reality passendes Format, ist in den wenigsten anderen Applikationen vorhanden. Diese spezialisieren sich meist auf das Erstellen von Mindmaps in Virtual Reality. Bei der freien Erstellung von Mindmaps in Virtual Reality kommt VA.SI.LI-MINDMAP jedoch nicht an einige der anderen Applikationen heran. Dies war allerdings auch nicht Fokus der Bachelorarbeit. Trotzdem existiert ein Aufgabentyp, mit dem man ein freies Mindmap erstellen kann, bei dem Inhalte der Knoten und verfügbare Verbindungsbeschriftungen vorgegeben sind. Die Art der Verbindungen, Platzierung der Knoten und Auswahl der Verbindungsbeschriftungen ist hierbei den Lernenden überlassen. Ein

Tabelle 1.1: Eigenschaftsvergleich verschiedener Applikationen

Applikation Eigenschaft	VA.SI.LI-MINDMAP	VERITAS	MVMMfFW ¹	Reality-Map	Mind Map VR	Noda (VR)	ELEwARMM ²
Erstellen von Knoten (VR)	Nein	Nein	Ja	Ja	Ja	Ja	Ja
Erstellen von Verbindungen (VR)	Ja	Ja	Ja	Ja	Ja	Ja	Ja
Beschriftung von Verbindungen (VR)	Ja	Nein	Nein	Nein	Nein	Nein	Nein
Färbung von Verbindungen (VR)	Nein	Nein	Ja	Nein	Ja	Ja	Nein
Gerichtete Verbindungen (VR)	Ja	Ja	Nein	Nein	Ja	Nein	Ja
Knoteninhalt Text	Ja	Ja	Ja	Ja	Ja	Ja	Ja
Knoteninhalt Bild/Symbol	Ja	Ja	Nein	Ja	Nein	Ja	Ja
Knoteninhalt Audio	Ja	Nein	Nein	Nein	Nein	Nein	Nein
Kollaborativ	Ja/Nein ³	Nein	Ja	Ja	Nein	Ja	Nein
Externe Aufgabenerstellung	Ja	Ja	Nein	Ja	Nein	Nein	Ja
... für Nicht-Programmierer	Ja	Nein	Nein	Nein	Nein	Nein	Ja
... mit Lösungsvergleich	Ja	Nein	Nein	Nein	Nein	Nein	Nein

weiterer Punkt sind die gerichteten und ungerichteten Verbindungen. Viele der Applikationen besitzen nur eine Art von Verbindung, und oftmals lassen sie keine genaueren Beschriftungen dieser zu. Durch die Beschriftungen kann man die Verbindungen klassifizieren und so um einiges genauere und aussagekräftigere Mindmaps erstellen.

„VAnnotatoR“ von Abrami, Mehler und Spiekermann ist ein Programm für virtuelle Umgebungen, mit dem man Daten wie Texte, Bilder und andere konzeptuelle Entitäten verlinken kann (Abrami, Mehler und Spiekermann 2019). Ebenfalls unterstützt „VAnnotatoR“ Text2Scene Systeme und benutzt „Natural Language Methods“, kurz NLP, um automatische Textanalysen durchführen zu können. Es kann in Virtual Reality Umgebungen benutzt werden und läuft, wie VA.SI.LI-MINDMAP, ebenfalls auf basis von Unity3D (Unity Technologies 2023) und OpenVR (ValveSoftware 2015). „VAnnotatoR“ besitzt eine Art des Verbindens von Elementen, die sehr ähnlich zu VA.SI.LI-MINDMAP ist. Diese ist allerdings weniger speziell auf Mindmaps und Aufgaben im Mindmapformat ausgerichtet, dafür aber um einiges umfangreicher. Aus diesen Gründen wurde hier auf einen Vergleich verzichtet.

Die hier entwickelte Website ist schwer mit anderen klassischen Mindmap-Werkzeugen zu vergleichen. Die meisten übertreffen die Website der These, da sie für das reine Erstellen von Mindmaps entwickelt worden sind und somit um einiges mehr Werkzeuge und Eigenschaften besitzen, die hier nicht zeitnah eingebaut werden konnten, beziehungsweise nicht benötigt werden. Dies lässt sich damit erklären, dass die hier entwickelte Website mit dem Gedanken erstellt wurde, die Mindmaps in einem passenden Format hochzuladen. Dies fehlt bei anderen zweidimensionalen Mindmapprogrammen.

Ein weiterer Aspekt ist die Inklusion von Audiodateien. Keines der hier untersuchten Applikationen besitzt die Komponenten für in Knoten eingebundene Audiodateien. Gerade für das Lernen von Fremdsprachen und das Verbessern von anderen auf Geräuschen oder Sprache basierten Fähigkeiten ist dies eine Notwendigkeit.

¹Multimodal Virtual Mind Map for Future Workplace

²Enhance Learning Experience with Augmented Reality Mind Map

³Zum Zeitpunkt der These ist VA.SI.LI-MINDMAP zwar in Va.Si.Li-Lab Mehler u. a. 2023 (siehe Kap. 2) integriert, benutzt allerdings noch nicht seine kollaborativen Funktionen. Das heißt, dass eine Person nicht das Mindmap anderer Personen sehen und damit interagieren kann. Allerdings können die Personen sich gegenseitig sehen und miteinander interagieren.

2 Basis / Ressourcen

Die Website basiert auf Javascript, Html und Css Dateien. Sie kann mit der API des Servers kommunizieren. Der Server wurde in Java mithilfe des Web-Frameworks Spark programmiert und kommuniziert mit einer MongoDB Datenbank. Die Virtual Reality Komponente wurde in der Echtzeit-Entwicklungsplattform Unity entwickelt und in das „VR-Lab for Simulation-based Learning“, auch Va.Si.Li-Lab, eingebaut. Die Virtual Reality Komponente kommuniziert ebenfalls mit der API des Servers.

2.1 Spark

Das Web-Framework Spark (Wendel 2015) ist ein Grundgerüst für Webanwendungen in Java oder Kotlin, was das Erstellen von diesen erleichtern soll. Spark ist frei verfügbar und „Open Source“. Hier wurde Spark benutzt, um eine API zu erstellen, mit der sowohl die Website geladen, als auch Daten mit der Website und der Virtual Reality Komponente ausgetauscht werden können.

2.2 MongoDB

MongoDB ist eine dokumentorientierte Datenbank (MongoDB, Inc 2009), in der Daten im BSON-Format gespeichert und gelesen werden können, das einem JSON-Format sehr ähnelt (MongoDB 2024). Das hier verwendete Datenformat ist in Kap. 7 näher beschrieben. In dieser Arbeit wird eine Datenbank genutzt, die durch den Fachbereich „Texttechnologie“ der Goethe Universität Frankfurt betrieben und zur Verfügung gestellt wurde.

2.3 Unity

Unity ist eine Echtzeit-Entwicklungsplattform für Echtzeit-3D-Spiele, -Applikationen und -Erfahrungen, und unterstützt insbesondere die Arbeit mit Virtual Reality (Unity Technologies 2023). Es wird eine Vielzahl von Systemen unterstützt, für die Applikationen entwickelt werden können. In den Virtual Reality Teil von VA.SI.LI-MINDMAP könnte also mit wenig Aufwand Augmented Reality Unterstützung für Mobilgeräte wie Handys integriert werden.

2.4 Va.Si.Li-Lab

Das „VR-Lab for Simulation-based Learning“ (Mehler u. a. 2023) ist eine Umgebung in Virtual Reality, die Nutzern verschiedene Lernszenarien in sozialen Umgebungen in Form von Rollenspielen anbietet. Nutzer können akustische und gestische Interfaces zur Kommunikation nutzen. Va.Si.Li-Lab's Virtual Reality Interaktion basiert auf einer veränderten Version von Ubiq (University College London 2023). Ubiq ist eine Anwendung zum Ermöglichen von

vernetzten Projekten in Forschung, Lehre und Entwicklung. Es stellt eine Reihe von Möglichkeiten der Online-Vernetzung bereit und steuert die Synchronisation zwischen Nutzern.

3 Grundlegende Komponenten und Datenfluss

VA.SI.LI-MINDMAP ist in drei Komponenten aufgeteilt. Eine Website (siehe Kap. 5) wird verwendet, um Aufgaben zu erstellen oder bereits erstellte Aufgaben zu verändern. Ein Server speichert die erstellten Aufgaben als Dokumente auf einer Datenbank (siehe Kap. 7). Hier werden Text, Bilder und Audiodateien gespeichert, als auch Verbindungen zu anderen Knoten, deren Positionen aus der Aufgabenerstellung der Website sowie weitere Informationen. Die Positionen und der Zoom-Wert werden benötigt, um das korrekte erneute Laden von den erstellten Aufgaben in der Website zu gewährleisten. Die Verbindungen dienen der späteren Überprüfung der Aufgaben innerhalb der VR/AR Komponente. Der Datenfluss zwischen den einzelnen Komponenten von VA.SI.LI-MINDMAP ist in Abb. 3.1 dargestellt.

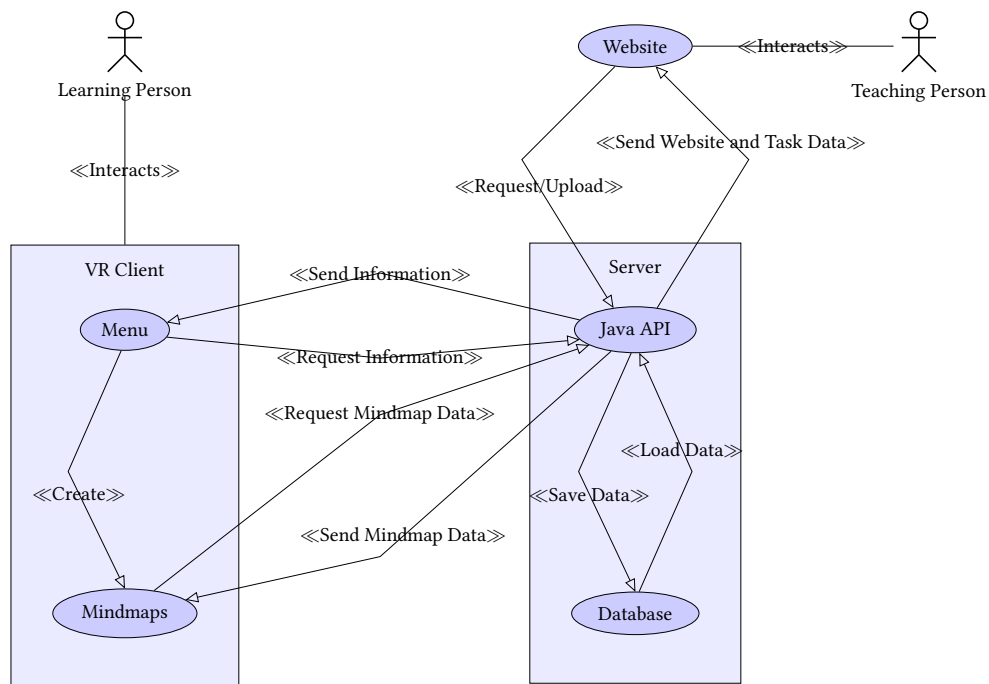


Abbildung 3.1: Datenflussdiagramm von VA.SI.LI-MINDMAP

Die Virtual Reality Komponente (siehe Kap. 8) ist in Unity3D (Unity Technologies 2023) erstellt und mithilfe von Ubiq (University College London 2023) in Va.Si.Li-Lab (Mehler u. a. 2023) integriert, und bietet ein Interface zum Auswählen der schon erstellten Aufgaben. Diese werden vom Server abgerufen. Je nach Aufgabe bekommen die Lernenden Knoten zur Verfügung, welche sie in ihrer Mindmap einfügen und verbinden können. Die Mindmap besteht aus dreidimensionalen Knotenobjekten und Linien. Die Knoten positionieren sich um die Lernenden in einer festen Distanz herum; verschieben kann man diese mit einer Greiffunktion. Die Auswertung/Korrektur erfolgt am Ende durch eine Überprüfung der Verbindungen.

Falsche Verbindungen werden rot hervorgehoben, fehlende Verbindungen gelb, eine richtige Verbindung mit falscher Beschriftung gelb-rot und die korrekte Lösung grün (siehe Kap. 8.8). Fehlende Verbindungen und falsche Verbindungsbeschriftungen werden bei der Überprüfung generiert beziehungsweise ersetzt.

4 Anwendungsfälle

Die folgenden Anwendungsfälle sind für VA.SI.LI-MINDMAP denkbar:

4.1 Allgemeiner Unterricht

Die Applikation ist für eine große Anzahl an Lernszenarien anwendbar, sowohl für den Unterricht in Schulen, Universitäten, Fachhochschulen, Ausbildungsstätten und Ähnlichem, als auch für private Lernkurse, Fort- und Weiterbildungen und eigenständiges Interesse. Lehrende können Aufgaben von Zuhause aus oder innerhalb der Lehreinrichtungen erstellen, die Lernenden können dann entweder in Klassenräumen oder in eigenständiger Heimarbeit diese lösen. Durch zukünftige Einbindung der Mindmaps in die Online-Komponenten aus Va.Si.Li-Lab (Abrami, Mehler, Bagci u. a. 2023) könnten diese Lernenden dann nicht nur miteinander kommunizieren, sondern sogar zusammen an Aufgaben arbeiten, selbst wenn sie sich in Realität nicht im gleichen Raum befinden.

4.2 Sprachenlernen

Weiterführend eignet sich VA.SI.LI-MINDMAP im besonderen für das Sprachenlernen, da durch die Verknüpfung von Text, Bildern und Audio geeignete Lernmaterialien zur Verfügung gestellt werden können. Durch zusätzliche Nutzung der verschiedenen Aufgabentypen (siehe Kap. 5) wird ein Lernprozess erreicht, der mit dem typischen Lernen von Sprachen gleichzusetzen ist, wie Hörtests, Vokabeln lernen, Texte lesen und übersetzen, sowie dem Vermitteln von Hintergrundwissen.

4.3 Erlernen praktischer Aufgaben

Durch das interaktive Bearbeiten der Aufgaben stellt sich als Anwendungsfall ebenfalls das Lernen praktischer Prozesse oder Jobs in den Vordergrund, da so eine stärkere Verinnerlichung der Abläufe gesichert wird als mit dem herkömmlichen Lernen auf Papier, beziehungsweise am Computer. Hier passen unter anderem, aber nicht exklusiv, praktische technische Berufe, Berufe die mit Versorgung oder Wartung zu tun haben, Arbeiten mit Menschen und generell Berufe die das Merken bestimmter Abläufe erfordern.

4.4 Unterhaltung

Eine weitere Kategorie der Anwendungsfälle wäre die Unterhaltung, da man mit VA.SI.LI-MINDMAP Szenarien entwickeln kann wie Gehirnaktivitätsaufgaben, das Erstellen von Comics, Frage- und Antwortquizzes, und das Entwickeln von kleineren Spielen und spielerischen Aufgaben.

4.5 Abbilden von Datenstrukturen

Neben Anwendungsfällen wie Lernprozessen ließen sich auch allgemein Datenstrukturen mit VA.SI.LI-MINDMAP abbilden, die auf Objekten oder Daten basieren, die untereinander vernetzt sind. Ebenfalls möglich sind Netzwerke und Graphen aus der Graphentheorie. Diese oben genannten Anwendungsfälle ließen sich allein durch Anpassung der Datenstruktur an die hier verwendete Datenstruktur der BSON-Dateien einrichten. Zusätzliche Funktionalitäten, wie z.B. eine automatische Erstellung der vernetzten Strukturen, müssten jedoch noch eingebaut werden. Es existiert zwar eine automatische Generierung einer Aufgaben-/Netzwerkstruktur, allerdings ohne automatische Platzierung der Knoten innerhalb der 3D-Umgebung.

5 Website

Die Website ermöglicht es, Aufgaben in einem Mindmapformat zu erstellen, zu verändern und hochzuladen. Verschiedenste Optionen stehen bereit, um die Aufgaben den eigenen Wünschen anzupassen (siehe Abb. 5.1).

Die Aufgabentypen entscheiden die grundlegende Art der Bearbeitung der Aufgaben innerhalb der Virtual Reality Komponente. Der Typ „Verbinden“ bestimmt, dass die Knoten in einer korrekten Weise verbunden werden müssen. Der Typ „Fehlende Wörter“ wurde speziell eingebaut, um aus einem Satz heraus Wörter und Wortarten zu entnehmen, die später wieder korrekt eingesetzt werden sollen. Wenn man keine Aufgabe lösen möchte, kann man mit dem Typ „Erkundung“ Knoten automatisch in einer festgelegten Reihenfolge erscheinen lassen. Diese Funktion dient vor allem dem Lernen, und nicht der Abfrage von Gelerntem. Ein Typ „Freies Mindmapping“ wurde ebenfalls eingebaut, um den Lernenden einen Bereich zu geben, um frei und ohne Bewertung Knoten zu verknüpfen.

Ein Feld für eine Zeitangabe kann benutzt werden um eine Zeitspanne vorzugeben, die später in der Virtual Reality Komponente die Dauer bestimmt, die Lernende haben, um die Aufgabe zu lösen.

Ein weiteres Feld dient dazu, eine Aufgabenbeschreibung hinzuzufügen (siehe Abb. 5.1).

Man kann innerhalb der Aufgabe verschiedene Seiten erstellen, um die Aufgabe zu unterteilen. Dies dient einer besseren Sichtbarkeit und einem geregelten Ablauf. Die Seiten können per Einstellung später entweder in Reihenfolge oder zufällig erscheinen. Jede Seite enthält ebenfalls eine optionale Beschreibung, falls sich die Seiten in der Art der Aufgabe unterscheiden (siehe Abb. 5.1). Man kann die Elemente einer Aufgabe auf der Website vergrößern oder verkleinern. Dies wird ebenfalls in der Datenstruktur für späteres Laden der Aufgabe gespeichert.

Die eigentlichen Aufgaben werden aus Elementen, den sogenannten Knoten, erstellt (siehe Abb. 5.4). Die Bewegung der Knoten wurde anhand eines Beispiels aus W3 Schools (W3 Schools 2023) erstellt. Diese Knoten können Text, Bild und/oder Audiodateien enthalten. Für das erneute Laden der Bild- und Audiodateien wurde der Inhalt eines Beitrags von Stackoverflow, Benutzer: cuixiping (Stackoverflow, Benutzer: cuixiping 2023) verwendet. Die Knoten können miteinander je nach Aufgabentyp durch gerichtete (siehe Abb. 5.2) oder ungerichtete (siehe Abb. 5.3) Verbindungen, beziehungsweise Kanten, verbunden werden. Diese gerichteten Kanten bestimmen die für später gesuchte Verbindung der Elemente, die Zuordnung untereinander oder die Reihenfolge des Erscheinens der Knoten, ebenfalls je nach Aufgabentyp.

Order of Sheet Appearance:

In order
Randomized

Task Description:
Eine Beispielaufgabe

Task Time (min):
10

Sheets:

Sheet 1
Sheet 1702520125960
Sheet 1702521293268
Sheet 1702522070418
Sheet 1702522334648

Add Sheet

Delete Current Sheet

Task Creator for "BeispielHauptaufgabe"

Type of Task on this Sheet:

Connection Task
Discovery Task
Missing Words Task
Free Mindmapping Task

Sheet Description:
Ordne nach Größe der Lösungen und

Nodes:
New Node

Zoom:
+ -

Abbildung 5.1: Ein Ausschnitt der gesamten Website mit erstellter Aufgabe

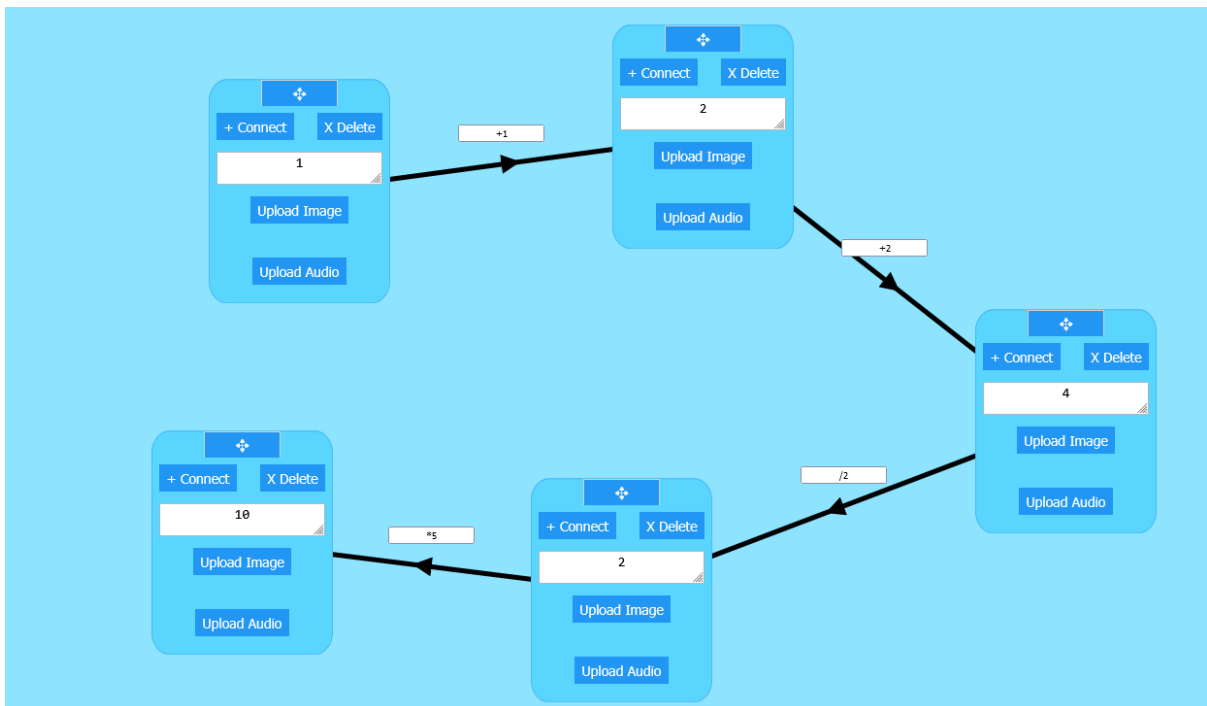


Abbildung 5.2: Ein Beispiel für eine Aufgabe mit gerichteten Verbindungen

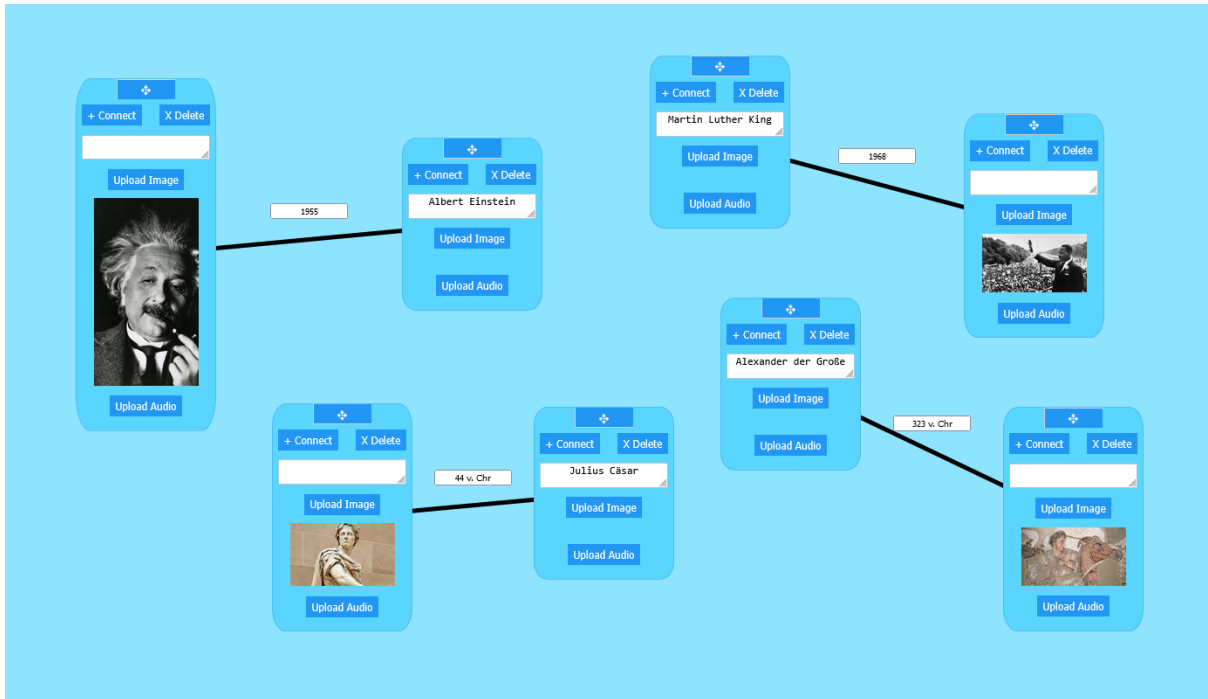


Abbildung 5.3: Ein Beispiel für eine Aufgabe mit ungerichteten Verbindungen und Abbildungen

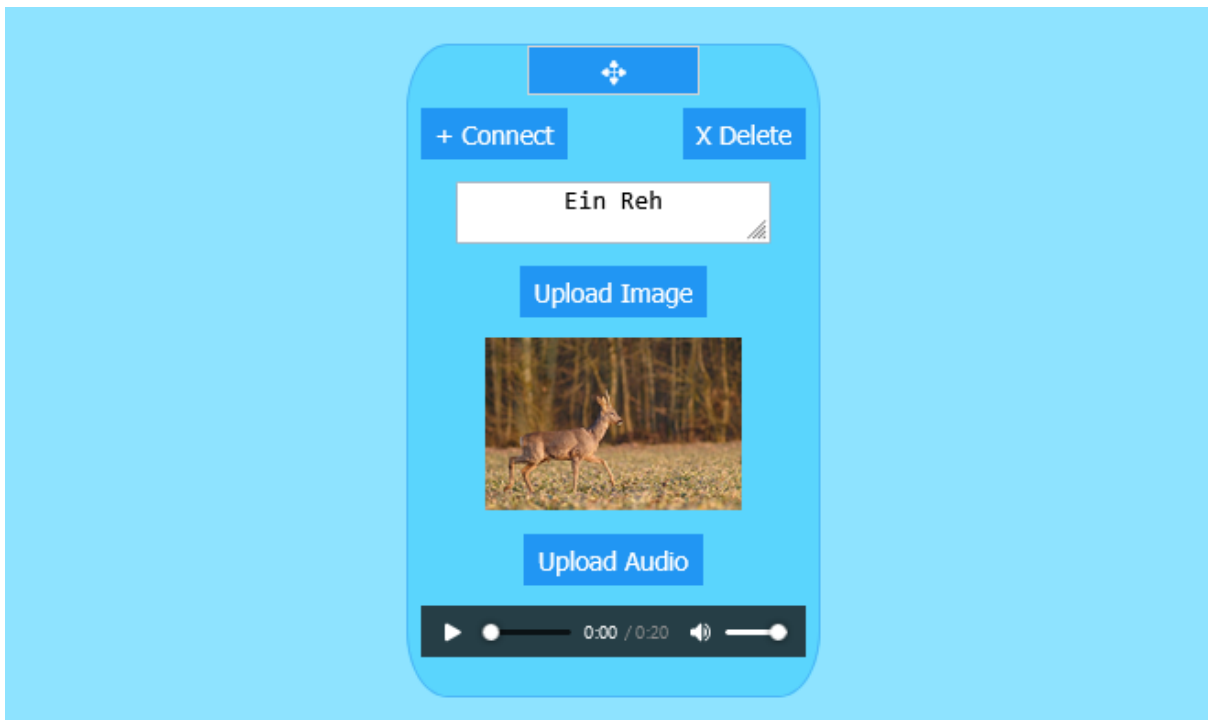


Abbildung 5.4: Ein Beispiel für ein Element, das einen Knoten darstellen soll, mit allen drei Inhaltsarten Text, Bild und Audio

6 Schnittstelle/Server und Datenbank

Ein Server stellt die Website bereit und enthält Befehle, die die Kommunikation zwischen Website und der Datenbank ermöglichen. Dies sind unter anderem Funktionen wie das Abrufen und das Hochladen der erstellten Aufgaben, als auch das Hochladen der Bild- und Audiodateien in GridFS, ein Werkzeug, mit dem man große Dateien in MongoDB (MongoDB, Inc 2009) hochladen kann. Ebenfalls werden die Dateien vorher auf dem Server aus einem Base64 String in ein für GridFS geeignetes ByteArray-Format umgewandelt.

Bei einem erneuten Hochladen einer aktualisierten Form schon bestehender Aufgaben werden die alten Versionen überschrieben, sowie die zugehörigen Bild- und Audiodateien aus GridFS gelöscht. Die Variablen für Bild- und Audiodateien haben somit je nach derzeitigem Ort ein anderes Format. Dieses Austauschen des Formats der Bild- und Audiovariablen ermöglicht eine effizientere Nutzung und ein Minimieren leerer Variablen innerhalb der Datenstruktur. Der Server stellt neben der oben genannten Kommunikation zwischen der Website und der Datenbank auch eine Kommunikation zwischen der Datenbank und der Virtual Reality Komponente in Unity bereit. Unity kann hier Aufgaben mit allen dazugehörigen Informationen abrufen.

7 Datenstruktur der Schnittstelle

Die Daten der erstellten Aufgaben werden in einem BSON Format auf einer MongoDB Datenbank gespeichert. Die oberste Schicht der Datenstruktur besteht aus BSON Dateien, die Aufgaben darstellen. Es werden Name der Aufgabe, Aufgabenbeschreibung, Typ der Aufgabe, vorgegebene Zeit und eine Liste von Aufgabenseiten gespeichert. Eine Aufgabenseite ist ebenfalls ein BSON, das den Namen der Seite, deren Beschreibung, einen für das erneute Laden von Aufgaben auf der Website benötigten Wert, der angibt, welche Skalierung die Knoten haben sollen, so wie eine Liste von auf der Aufgabenseite vorhandenen Knoten enthält. Die Knoten jeweils sind weiterhin im BSON Format und bestehen aus einer ID und Inhalten wie Text, Audio und Bild, wobei Audio und Bild jeweils eine ID sind, die mit den in GridFS gespeicherten Dateien verknüpft werden kann. Außerdem sind zweidimensionale Koordinaten vorhanden, was beim erneuten Erzeugen der Aufgabe auf der Website ein korrektes Platzieren der Knoten ermöglicht, und durch eventuelle Erweiterung der Virtual Reality Komponente eine automatische Platzierung von Aufgaben zuließe. Zuletzt wird eine Liste mit Verbindungen gespeichert. Diese sind gleichfalls im BSON-Format, und enthalten die IDs der zwei der Verbindung angehörigen Knoten. Eine der beiden IDs ist zwangsläufig die ID des Knotens, der die Verbindung enthält. Die Reihenfolge der gespeicherten IDs gibt die Richtung der Verbindung an.

Im Folgenden ist das Format der Datenstruktur dargestellt:

```
1 {
2   "result": [
3     {
4       "taskName": string,
5       "taskDescription": string,
6       "sheetOrder": string,
7       "taskTime": int,
8       "sheets": [Document]
9     }
10  ]
11 }
```

Die Aufgabenseiten, hier „sheets“ genannt, haben das folgende Format:

```
1 {
2   "sheetName": string,
3   "taskType": string,
4   "sheetDescription": string,
5   "zoomLevel": float,
6   "nodes": [Document]
7 }
```

Bei Knoten, hier „nodes“, haben die „image“ und „audio“ Variablen Base64-Strings gespeichert, wenn sie außerhalb der Datenbank vorliegen. Innerhalb der Datenbank besitzen sie IDs, die auf Dokumente in GridFS weisen. Im folgenden werden die Datenstrukturen so dargestellt, als wären die Datentypen dieser Variablen Base64-Strings:

```

1 {
2   "nodeID": string,
3   "x": int,
4   "y": int,
5   "text": string,
6   "image": string,
7   "audio": string,
8   "connections": [Document]
9 }

```

Innerhalb der Knoten sind deren Verbindungen, hier „connections“, zu anderen Knoten gespeichert:

```

1 {
2   "connectionID": string,
3   "node1ID": string,
4   "node2ID": string,
5   "connectionTag": string
6 }

```

7.1 Umwandlung der „image“ und „audio“ Variablen

Die Variablen, die größere Dateien beinhalten, wie Bild- und Audiodateien, müssen in der MongoDB-Datenbank (MongoDB, Inc 2009) in ein System hochgeladen werden, das GridFS genannt wird. An diesem Punkt werden also Inhalte der Variablen, die vorher mit Base64-Strings der Dateien gefüllt waren, in GridFS hochgeladen, und die Inhalte der Variablen werden zu IDs umgewandelt, die Zugriff auf die in GridFS liegenden Dokumente enthalten:

```

1 String base64 = node.getString("image");
2 byte[] imageBytes = Base64.getDecoder().decode(base64);
3 bytes = imageBytes;
4 try(GridFSUploadStream uploadStream = MongoDBConnectionHandler.gridFSBucket.openUploadStream("
   image"))
5 {
6   uploadStream.write(imageBytes);
7   uploadStream.close();
8   node.replace("image", uploadStream.getObjectId());
9 }
10 catch (Exception e)
11 {
12   System.out.println("The image upload failed");
13   e.printStackTrace();
14 }

```

Die Umwandlung beim Abrufen der Daten aus der Datenbank über den Server funktioniert wie folgt:

```

1 try (GridFSDownloadStream downloadStream = MongoDBConnectionHandler.gridFSBucket.
   openDownloadStream(node.getObjectId("image")))
2 {
3   int fileLength = (int) downloadStream.getGridFSFile().getLength();
4   byte[] bytesToWriteTo = new byte[fileLength];
5   int bytesRead = 0;

```

```
6 String base64String = "";
7 while (bytesRead < fileLength)
8 {
9     int newBytesRead = downloadStream.read(bytesToWriteTo);
10    if (newBytesRead == -1)
11        throw new Exception();
12    byte[] actualBytes = new byte[newBytesRead];
13    System.arraycopy(bytesToWriteTo, 0, actualBytes, 0, newBytesRead);
14    base64String = base64String.concat(Base64.getEncoder().encodeToString(actualBytes));
15    bytesRead += newBytesRead;
16 }
17 node.replace("image", base64String);
18 }
19 catch (Exception f)
20 {
21     System.out.println("The [insert data type] download failed");
22     f.printStackTrace();
23 }
```

8 Virtual Reality Komponente

Den wichtigsten Bestandteil von VA.SI.LI-MINDMAP stellt die Virtual Reality Komponente dar. Hier können vorher erstellte Aufgaben geladen und bearbeitet werden. VA.SI.LI-MINDMAP ist in Va.Si.Li-Lab integriert und lässt sich von dort aus starten.

8.1 Menü

In einem Menü (siehe Abb. 8.1) wird aus verfügbaren Aufgaben ausgewählt. Jedes Objekt stellt eine Aufgabe dar und zeigt deren Name, Beschreibung, die Anzahl der Aufgabenseiten, die Anzahl der gesamten Knoten der Aufgabe, sowie eine Zeit, in der die Aufgabe gelöst werden muss, an. Beim Interagieren mit einem der Objekte wird ein neues Objekt erstellt, das das zugehörige Mindmap darstellt.

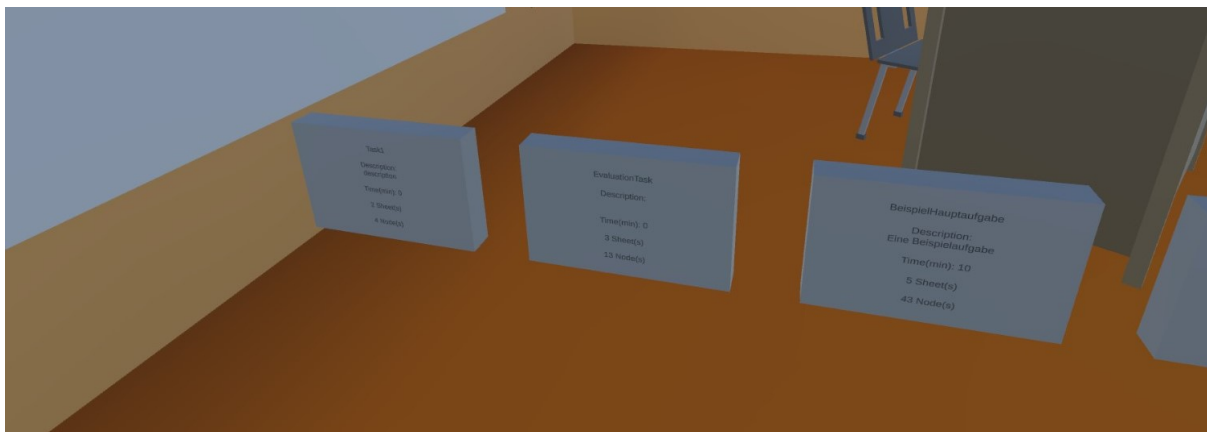


Abbildung 8.1: Menü zum Auswählen einer Aufgabe

8.2 Mindmap-Objekt

Eine Aufgabe wird nach Auswahl des entsprechenden Menüobjektes als Objekt in Unity generiert. Dieses kann gegriffen und bewegt werden, damit man Aufgaben an einer für sich ansprechenden Stelle innerhalb der Va.Si.Li-Lab Umgebung bearbeiten kann. Dies gilt auch noch während der Bearbeitung der Aufgabe, da alle weiteren Objekte, die durch das Mindmap erstellt werden, daran hängen. Man kann es ebenfalls wieder löschen, indem man es greift, und währenddessen die Interaktionstaste (siehe Kap. 8.4), definiert durch Ubiq (University College London 2023) und Va.Si.Li-Lab (Mehler u. a. 2023), an einem der Controller drückt. Die Verbindung zwischen verschiedenen Klassen innerhalb der Virtual Reality Komponente wird in einem UML-Klassendiagramm in Abb. 12.2 genauer betrachtet. Inhalte mehrerer

Klassen wurden für eine verbesserte Übersicht dem Klassendiagramm entnommen und in die Abbildungen 12.3, 12.4 und 12.5 verschoben.

8.3 Umwandlung in Aufgabenform

Die in der Datenbank gespeicherten Aufgaben entsprechen der jeweiligen Lösung dieser. Wenn eine Aufgabe in Unity geladen wird, muss aus dieser Lösung eine zu bearbeitende Aufgabe erstellt werden. Zunächst werden die Handmenüs (siehe Kap. 8.6) erstellt und mit Inhalten gefüllt. Ein Teil davon ist die Anzeige für vorhandene Verbindungsbeschriftungen. Diese werden berechnet und für jede Aufgabenseite gespeichert (siehe Kap. 8.9). Die Reihenfolge der Bearbeitung der Aufgabenseiten wird durch eine Variable bestimmt, die vorgibt, ob die Aufgabenseiten der lernenden Person in vorgegebener oder in zufälliger Reihenfolge angezeigt werden. Ebenfalls werden alle Knoten der gesamten Aufgabe schon am Anfang erzeugt, um eventuelle Verzögerungen und Erstarrungen durch Ladezeiten während der Bearbeitung, die vor allem durch das Laden der Bild- und Audiodateien entstehen, zu verhindern. Die Knoten werden auf Basis eines Knotenobjektes generiert. Dieses ist so aufgebaut, als würde es alle drei Arten von Inhalten, nämlich Text, Bild und Audio, enthalten (siehe Abb. 8.2). Es wird nun geprüft, welche davon im Kontext des jeweiligen Knotens existieren. Teile des Objektes, die Inhalten entsprechen, die nicht für den zu erstellenden Knoten vorgesehen sind, werden gelöscht. Alle anderen Teile werden mit den entsprechenden Inhalten gefüllt.

Das Laden der Audiodateien erfordert eine andere Funktionalität als das Laden der Texte und Bilder. Zur Hilfe wurde ein Beitrag aus Stackoverflow, Benutzer: Çağatay IŞIK (Stackoverflow, Benutzer: Çağatay IŞIK 2023) in VA.SI.LI-MINDMAP eingebaut und angepasst.

Knotenobjekte werden nun nach ihren zugehörigen Aufgabenseiten gruppiert und deaktiviert. Bis auf bei einem Aufgabentyp (siehe Kap. 8.10) werden die Knoten zufällig durchmischt, sodass beim Bearbeiten kein Abschätzen der Lösung durch die Anordnung der Knotenauswahl getätigt werden kann. Das spätere Aktivieren beziehungsweise Anzeigen der Knoten wird durch die aktuelle Aufgabenseite und Handmenüs (siehe Kap. 8.6) bestimmt. Ebenfalls bis auf diesen Aufgabentyp (siehe Kap. 8.10) werden Verbindungen, die in der Lösung definiert wurden, nur innerhalb der Validierung wieder verwendet, mit Ausnahme der oben genannten Verbindungsbeschriftungen (siehe Kap. 8.9).

8.4 Interaktion

Die Art der Interaktion wird von Va.Si.Li-Lab (Mehler u. a. 2023), basierend auf Ubiq (University College London 2023), bestimmt. Man benutzt jeweils einen Virtual Reality Controller pro Hand, sowie eine Virtual-Reality-Brille. Im Verlauf der Entwicklung wurde hauptsächlich die Oculus Quest 2 benutzt, bei der Evaluation (siehe Kap. 9) eine Oculus Quest Pro. Die für die VR-Komponente von VA.SI.LI-MINDMAP benutzten Tasten auf den Controllern sind eine Taste für das Greifen von Objekten und eine Taste für das Benutzen/Interagieren von Objekten. Andere Tasten werden nur über Va.Si.Li-Lab generell benutzt, nicht speziell in VA.SI.LI-MINDMAP.



Abbildung 8.2: Ein mit Text, Bild und Audio gefüllter Knoten

8.5 Anordnung des Mindmaps

Innerhalb einer Aufgabe steht man im Mittelpunkt des zu bearbeitenden Mindmaps (siehe Abb. 8.3). Es werden zwei Hände angezeigt, mit denen man mit Elementen interagieren kann. Knoten, die Text, Bilder und/oder Audiodateien enthalten können, werden hier durch ein Menü am linken Arm gegriffen und daraufhin auf die für das Mindmap vorgesehene Fläche gesetzt. Die vorgesehene Fläche ist eine imaginäre Kugel mit Zentrum am Mittelpunkt des Mindmapobjektes. Die Knoten des Mindmaps positionieren sich auf der Außenfläche dieser imaginären Kugel (siehe Abb. 8.4). Die Knoten können durch Greifen (siehe Kap. 8.4) verschoben werden und orientieren sich dabei weiterhin an der Außenseite der Kugel. Diese Position wird berechnet durch eine Funktion, die eine Koordinate auf der Außenseite der Kugel berechnet, die der Position der Hand, die den Knoten gegriffen hat, am nächsten ist. Die Elemente sind außerdem zum Mittelpunkt des Kreises hin gerichtet.

8.6 Handmenüs

Sowohl die linke als auch die rechte Hand in der VR-Komponente besitzen jeweils ein Menü, das an eine Geste gebunden ist, um sichtbar zu werden. Die Geste ist für jede Hand gleich,

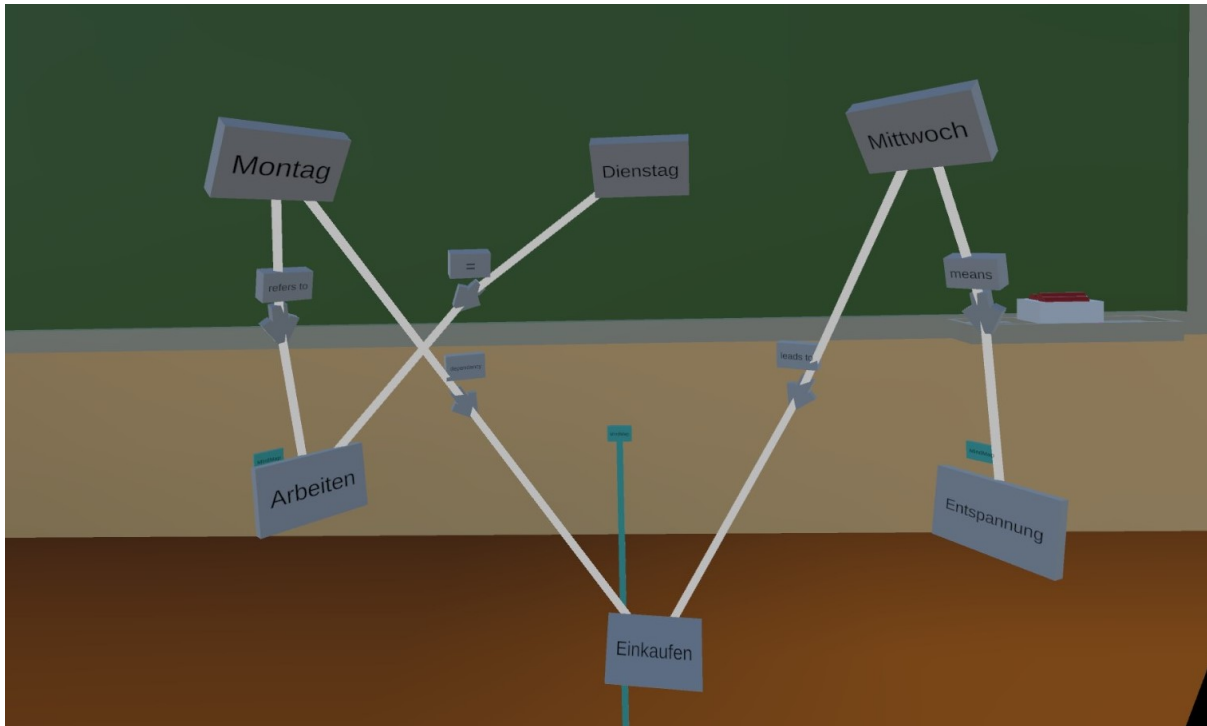


Abbildung 8.3: Beispiel eines Mindmaps von vorne betrachtet

und ähnelt dem Betrachten einer Armbanduhr am eigenen Handgelenk.

Das Menü an der linken Hand (siehe Abb. 8.5) enthält die auf der aktuellen Aufgabenseite verfügbaren Knoten, die noch nicht in der Umgebung des Mindmaps platziert wurden. Diese werden verkleinert über der Hand angezeigt und folgen dieser bei Bewegungen. Beim Aufgabentyp, bei dem nacheinander in bestimmter Reihenfolge Knoten platziert werden müssen und diese sich automatisch verbinden (siehe Kap. 8.10), wird nur der nächste Knoten angezeigt, der platziert werden soll. Bei allen anderen Aufgabentypen werden alle verfügbaren Knoten angezeigt. Diese Knoten werden verkleinert dargestellt. Beim Greifen werden sie in ihre für das Mindmap vorgesehene Größe skaliert, und in eine Gruppe von bereits platzierten Knoten verschoben, für die es gilt, der Außenseite der imaginären Kugel (siehe Kap. 8.5) zu folgen.

Das Menü an der rechten Hand (siehe Abb. 8.6) enthält Informationen wie die Aufgabenbeschreibung der aktuellen Aufgabenseite, die verbleibende Zeit der gesamten Aufgabe, sowie eine Menge an auf der aktuellen Aufgabenseite verfügbaren Verbindungsbeschriftungen. Ebenfalls enthält dieses Menü zwei interagirbare Objekte, die als Tasten dienen. Die eine Taste lässt Nutzer die Aufgabenseite wechseln, die andere validiert (siehe Kap. 8.8) die aktuelle Aufgabenseite.

8.7 Verbinden von Knoten

Knoten können miteinander verbunden werden. Diese Verbindungen erstellt man durch Interaktion mit einem Knoten, der sich daraufhin als markiert anzeigt. Danach interagiert man

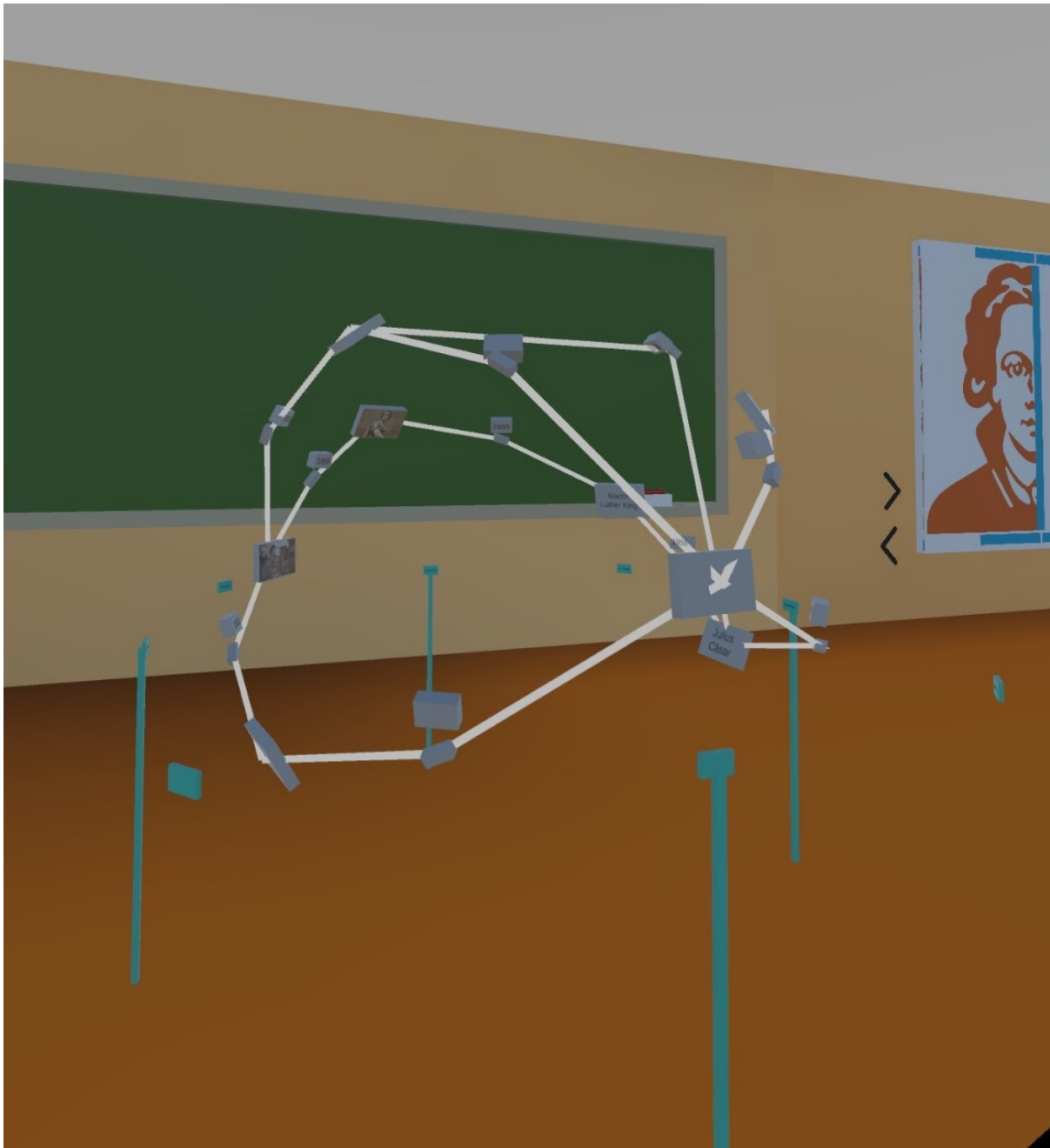


Abbildung 8.4: Ein von außerhalb betrachtetes Mindmap, mit Knoten und Verbindungen über den Großteil der verfügbaren Fläche verteilt. Die blauen Objekte an den Seiten stellen den Außenrand dar und sind der Interaktionspunkt für das bewegen des gesamten Mindmaps

mit einem anderen Knoten. Verbindungen zwischen Knoten sind entweder gerichtet (siehe Abb. 8.7) oder ungerichtet (siehe Abb. 8.8), abhängig vom Aufgabentyp, und können durch Auswahl eines Knotens und darauf folgend der Auswahl eines zweiten Knotens erstellt werden. Die Ausführung der gleichen Aktion bei schon bestehender Verbindung, als auch die

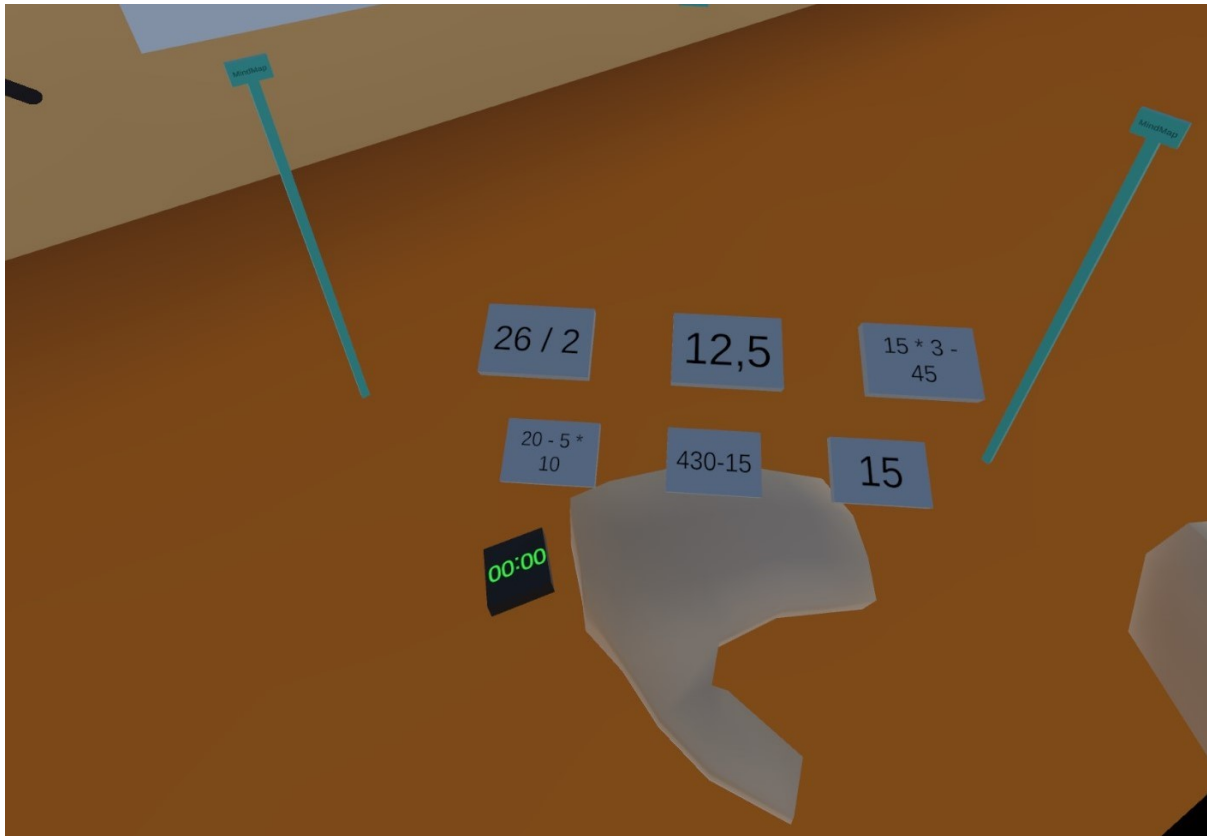


Abbildung 8.5: Menü mit Knotenauswahl auf der linken Hand

Interaktion mit der Verbindung selbst, kann diese wieder entfernen.

Durch das Menü am Arm kann die Aufgabenseite gewechselt und eine Evaluation der aktuellen Aufgabenseite durchgeführt werden. Es kann auch in das Hauptmenü zurückgegangen werden.

Bei der Evaluation einer Aufgabe werden die erstellten Kanten mit der vorhandenen Lösung verglichen. Übereinstimmende Kanten werden grün markiert, Kanten die gesetzt wurden, aber nicht in der Lösung vorhanden sind, rot. Kanten, die in der Lösung gesetzt wurden, aber nicht in der Bearbeitung der Aufgabe, werden neu erzeugt und gelb gefärbt.

8.8 Ermittlung der Validierung

Um eine Aufgabenseite zu validieren, werden die erstellten Verbindungen zwischen Knoten verglichen. Zum einen haben wir die entsprechende Aufgabenseite der von der Schnittstelle stammenden BSON-Datei, die die Lösung der Aufgabe enthält. Zum anderen benutzen wir die Verbindungen aus dem erstellten Mindmap in VR.

Zuerst geht eine Schleife durch alle in der BSON-Datei existierenden Verbindungen und überprüft, ob diese auch in VR existieren. Wenn eine Verbindung existiert, die gesucht ist, dann wird diese in VR grün markiert (siehe Abb. 8.9). Falls sie existiert, aber die Beschriftung der

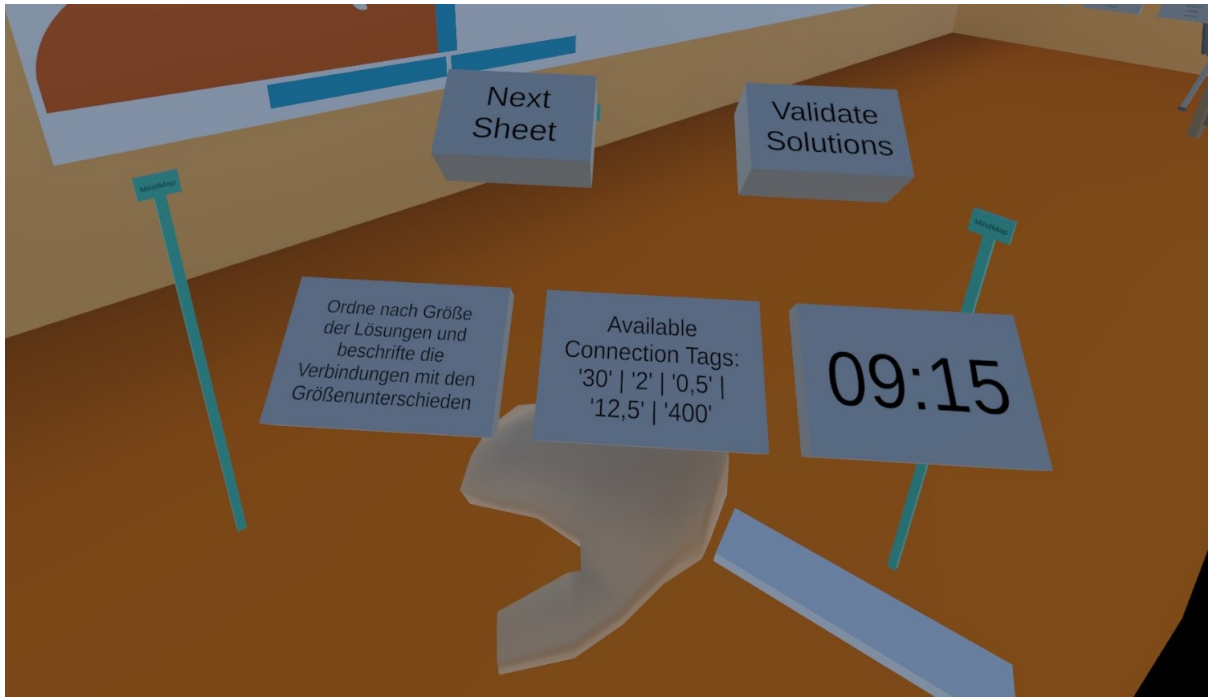


Abbildung 8.6: Menü mit Informationen zur Aufgabe und Tasten zum Wechseln der Aufgabenseite und der Validierung auf der rechten Hand

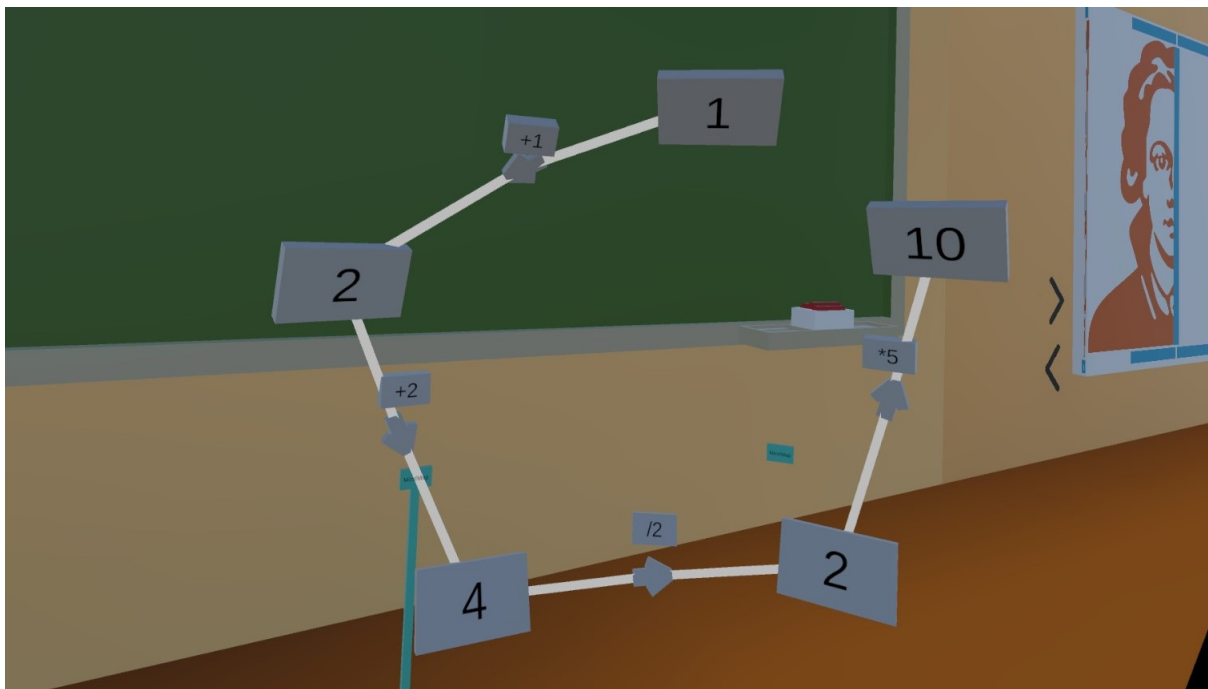


Abbildung 8.7: Gerichteter Graph aus dem Aufgabentyp mit automatischer Verbindung

Verbindung (siehe Kap. 8.9) falsch gesetzt wurde, dann wird die Verbindung gelb gefärbt, und die Verbindungsbeschriftung rot. Wenn sie in VR nicht existiert, wird sie erstellt, korrekt be-

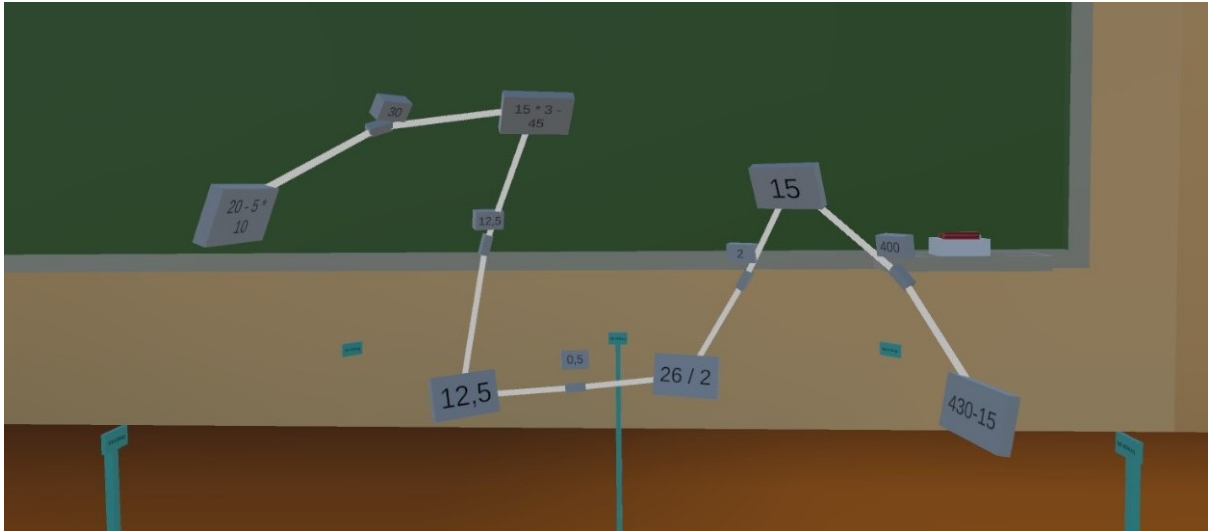


Abbildung 8.8: Ungerichteter Graph

schriftet und gelb gefärbt (siehe Abb. 8.10). In einer zweiten Schleife wird nun durch alle in VR erstellten Verbindungen gegangen, die noch nicht in der vorherigen Überprüfung als korrekt festgestellt wurden. Alle diese verbliebenen Verbindungen werden rot gefärbt, da diese von den Lernenden gesetzt wurden, aber nicht in der Lösung gefordert sind, und somit einen Fehler darstellen.

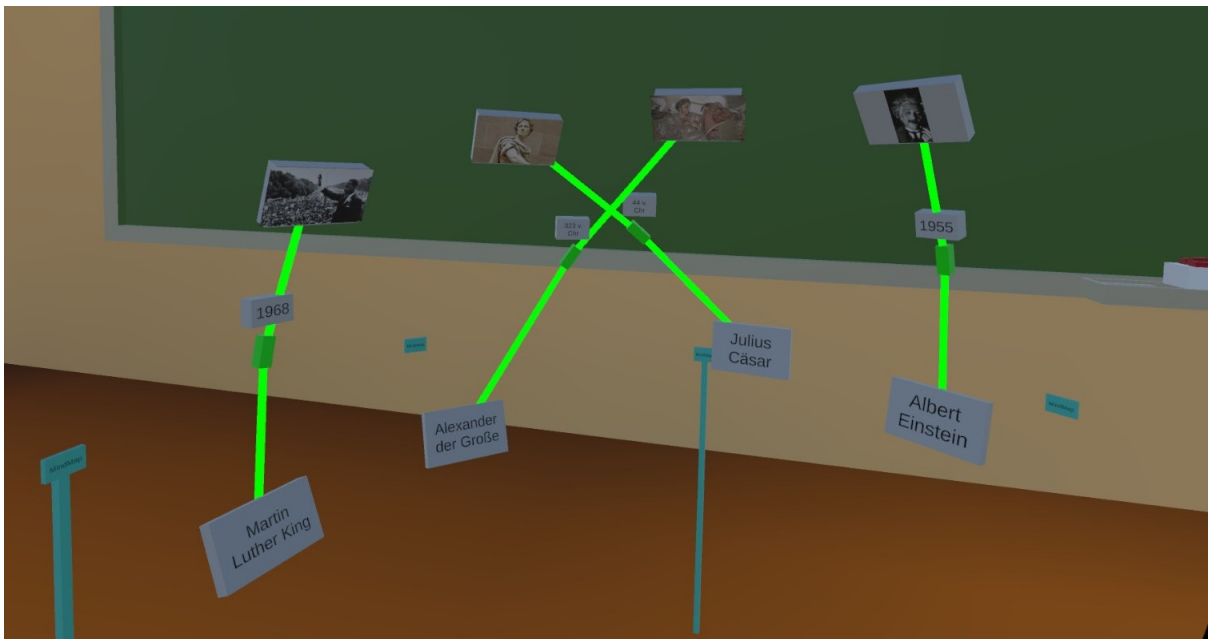


Abbildung 8.9: Korrektur der Aufgabenseite eines Mindmaps mit einheitlich richtig gesetzten Verbindungen

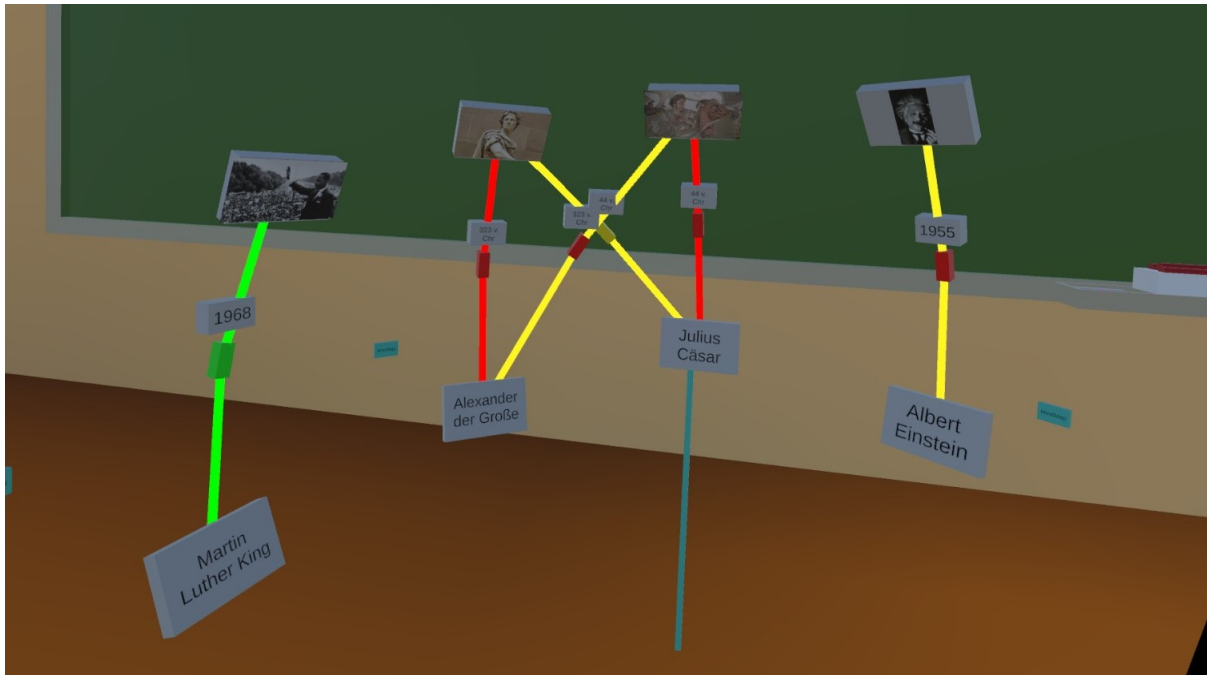


Abbildung 8.10: Korrektur der Aufgabenseite eines Mindmaps mit richtig gesetzten Verbindungen, falsch gesetzten Verbindungen und richtig gesetzten Verbindungen mit falscher Verbindungsbeschriftung

8.9 Erzeugung der Verbindungsbeschriftungen

Beim Erstellen der Aufgabe auf der Website können Verbindungen beschriftet werden. Dies stellt dann die gesuchte Lösung dar, und muss je nach Aufgabentyp von den Benutzern in der VR-Komponente eingestellt werden. Eine Auswahl wird generiert, indem alle auf einer Aufgabenseite verfügbaren Verbindungsbeschriftungen gesammelt und in einer Liste ohne mehrfaches Vorkommen der gleichen Beschriftungen gespeichert werden. Diese Beschriftungen sind nun die Auswahl der Beschriftungen in VR, die gesetzt werden können. Sie werden im Menü der rechten Hand (siehe Kap. 8.6) für die Aufgabenseite angezeigt, in der sich die Lernenden gerade befinden.

8.10 Berechnung der nächsten Knoten der Aufgabe mit automatischer Knotenverbindung

Bei Aufgabentypen, deren Knoten in der Virtual Reality Komponente automatisch verbunden werden sollen, muss eine Reihenfolge der Knoten festgelegt werden, die der Reihenfolge der gerichteten Verbindungen aus der Datenstruktur (siehe Kap. 7) folgt. Um dies zu berechnen, wird innerhalb der Knoten der Datenstruktur versucht, einen Knoten zu finden, der die geringste Anzahl eingehender Verbindungen besitzt (Tittmann 2003 S. 128-146). Graphentheoretisch ist dies einer der Knoten mit den wenigsten Eltern. Falls der Knoten keine Eltern besitzt, dann redet man sogar von einer Wurzel. Dieser ermittelte Knoten wird nun entfernt und in einer Liste an erster Stelle gesetzt. Diese Liste gibt vor, in welcher Reihenfolge die Knoten

in der VR-Komponente von den Lernenden erstellt werden sollen. Im Folgenden wird nun in dem Graphen, abzüglich des schon ermittelten Knotens sowie dessen ein- und ausgehenden Verbindungen, der nächste Knoten gesucht, der nun die wenigsten Eltern besitzt.

Dieser Ablauf wiederholt sich solange, bis kein Knoten mehr im Graphen verfügbar ist. Falls es mehrere Knoten mit der gleichen minimalen Anzahl eingehender Verbindungen gibt, wird der erste verfügbare Knoten ausgewählt.

Für das Sicherstellen der Reihenfolge in der Virtual Reality Komponente wird nun auf dem Menü der linken Hand (siehe Kap. 8.6) immer nur der Knoten angezeigt, der sich an erster Stelle der Reihenfolge befindet. Beim Herausnehmen und Setzen des Knotens wird dieser aus der Reihenfolge entfernt.

9 Evaluation von VA.SI.LI-MINDMAP

9.1 Aufbau

Eine Evaluation von VA.SI.LI-MINDMAP wurde durch Tests mit 15 Probanden durchgeführt. Diese bearbeiteten vorgegebene Aufgaben in der Virtual Reality Komponente und auf der Website. Der VR-Teil bestand aus einer Einführungsaufgabe, bei der Hilfestellungen sowohl mündlich durch den Prüfer als auch durch Textfelder in der Applikation bereitgestellt wurden, und einer anspruchsvolleren Aufgabe, die bis auf wenige Ausnahmen ohne Hilfe durchgeführt werden sollte.

Danach wurde noch ein kurzer Test auf der Website veranstaltet, bei dem die Probanden einer Schritt-für-Schritt Anweisung folgten, um deren Funktionen zu erlernen.

Im Abschluss wurde allen Probanden ein User Experience Questionnaire, auch UEQ, gestellt, der 18 Fragen bezüglich der Benutzbarkeit und Zufriedenstellung der getesteten Bestandteile beinhaltet (siehe Abb. 12.1). Dieses UEQ wurde den Probanden direkt im Anschluss gestellt, und anonym und ohne Einfluss des Prüfers beantwortet. Jede Frage konnte durch Ankreuzen von Feldern mit Werten in der Spanne von -2 bis 2 vergeben werden, wobei -2 als „Wenig“ und 2 als „Sehr“ oder „Viel“ zu verstehen war.

9.2 Auswertung

Die Ergebnisse der Befragung wurden in eine Tabelle eingetragen, und der Durchschnitt pro Frage wurde errechnet (siehe Tabelle 9.1). Alle Probanden bis auf eine Person kreuzten ganze Zahlen an. Die durchschnittlichen UEQ-Ergebnisse pro Frage, in Abb. 9.1, befinden sich ausschließlich im positiven Bereich von 0,2 bis 1,67. Nach Ausschluss der Frage 9 (siehe Abb. 12.1) über die Vorkenntnisse der Probanden mit Virtual Reality liegen nur 11,76% der Fragen mit ihrem Durchschnitt im Bereich von 0 bis 1, während die anderen 15 Fragen im Bereich von 1 bis 2 liegen. Die beiden Fragen im Bereich von 0 bis 1 betreffen zum einen den visuellen Eindruck der VR-Komponente, und zum anderen die für VA.SI.LI-MINDMAP benötigten Vorkenntnisse in Virtual Reality. Frage 9 bezog sich auf die eingeschätzten Vorkenntnisse der Probanden in Virtual Reality. Hier liegt der Durchschnitt bei einem Wert von 0,2. Da für die Tests keine Vorkenntnisse mit Virtual Reality Applikationen vorausgesetzt waren, kann man hier also bei einem Durchschnitt, der fast der Mitte der Bewertungsskala entspricht, und einem Streubereich von -2 bis 2 auf eine ausgeglichene Menge an Probanden schließen.

Minima und Maxima der Bewertungen gehen bei den Probanden teils stark auseinander. Während jeder Proband mindestens zwei Fragen mit einer 2 bewertet hat, haben nur 20% der Probanden überhaupt eine -2 in mindestens einer Frage angekreuzt, wobei unter erneutem Ausscheiden der Frage 9 nur ungefähr 6,7% der Probanden, also eine Person, mindestens eine -2 vergeben haben. Dieser Proband hat ein Drittel der Fragen mit einer -2 bewertet, von denen sich jedoch 83% auf die Website beziehen. Hier scheint es sich jedoch, unter Berücksichtigung der Durchschnittswerte dieser Fragen, um eine Abweichung unter den Probanden zu handeln.

Es konnte eine Relation festgestellt werden zwischen Frage 9, den eigenen Vorkenntnissen in VR, und Frage 10 über die für VA.SI.LI-MINDMAP benötigten Vorkenntnisse in VR. 50% aller Probanden, die weniger Vorkenntnisse in Virtual Reality angaben, gaben bei Frage 10 ebenfalls eine negative Bewertung, während 62,5% aller Probanden, die gute Vorkenntnisse in VR angaben, einen positiven Wert bei Frage 10 eingetragen hatten.

Tabelle 9.1: Bewertungen der Probanden

Frage Proband	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
1	2	1	2	2	2	2	2	2	-1	-1	2	2	2	2	2	2	2	2
2	1	1	2	0.5	1.5	1	1	1	1	2	2	0	2	2	2	0	2	2
3	2	2	2	2	2	2	2	2	1	2	2	2	2	2	2	2	2	2
4	2	-1	1	2	2	2	1	2	-1	1	1	-2	-2	-1	-2	-2	-2	-2
5	1	2	2	1	2	2	1	2	2	0	0	1	1	2	2	2	2	2
6	1	2	2	1	1	0	1	0	2	1	1	0	1	1	1	1	2	1
7	2	2	2	2	2	2	2	2	-1	0	2	2	2	2	2	2	2	2
8	1	2	1	0	1	1	0	0	-1	2	2	1	1	1	2	1	0	1
9 (Frage 15-18 unbeantwortet)	1	0	1	0	2	2	0	1	1	-1	0	0	2	2	-	-	-	-
10 (Frage 15-18 unbeantwortet)	2	1	2	1	1	2	1	1	1	-1	1	1	2	2	-	-	-	-
11	2	1	2	1	2	1	1	2	0	2	2	2	2	2	2	2	2	2
12	2	2	2	1	2	1	1	2	2	1	2	2	1	2	2	2	2	2
13	1	-1	1	2	1	1	0	1	-2	-1	1	1	1	0	1	1	0	2
14	1	1	2	-1	1	1	1	2	-2	-1	0	0	1	2	0	1	-1	-1
15	2	1	2	1	1	1	2	2	1	2	2	2	2	2	2	2	1	2
Gerundeter Durchschnitt	1.53	1.07	1.67	1.1	1.57	1.4	1.07	1.47	0.2	0.53	1.33	0.93	1.33	1.53	1.2	1.07	0.93	1.13
Gerundete Varianz	0.27	1.07	0.24	0.86	0.25	0.4	0.5	0.55	2.03	1.7	0.67	1.35	1.1	0.84	1.46	1.35	1.78	1.7

9.3 Anmerkungen/Probleme der Evaluationsteilnehmenden

9.3.1 VR-Komponente

Mehrere Probanden merkten an, dass einige Texte auf den Knoten zu klein und somit schlecht lesbar waren. Ebenfalls kam es zu Problemen mit der Benutzung der Menüs, die an die rechte und linke Hand gebunden waren (siehe Kap. 8.6). Die Winkel, in denen die Arme gehalten werden mussten, um die Menüs sichtbar zu machen, wurden oft entweder nicht richtig erkannt, oder die Halteposition war für Probanden umständlich. Mehrere Personen wünschten sich ein erweitertes Tutorial für die Applikation. Ein weiteres Problem in vieler Probanden schien die Verbindung der Knoten zu sein. Durch fehlerhafte Benutzung der Knoten und auch noch vorhandener kleiner Fehler in der Programmierung des VR-Teils von VA.SI.LI-MINDMAP zur Zeit der Evaluation, kamen ab und zu für die Probanden unverständliche Probleme auf. Weitere Vorschläge beinhalteten eine Verbesserung der Bewegung der Knoten, Knoten mithilfe einer in der VR verbreiteten Laserpointer-Komponente bewegbar zu machen, Richtungen der Verbindungen weniger umständlich zu ändern, und weitere Arten von Aufgaben.

9.3.2 Website

Teilnehmende merkten an, dass sie gerne Aufgabenseiten benennen würden. Eine Person wünschte sich ein Tutorial. Eine Person merkte ebenfalls an, dass eine Zeitvorgabe pro Aufgabenseite, im Gegensatz zur existierenden Zeitvorgabe für die gesamte Aufgabe, eine Verbesserung wäre.

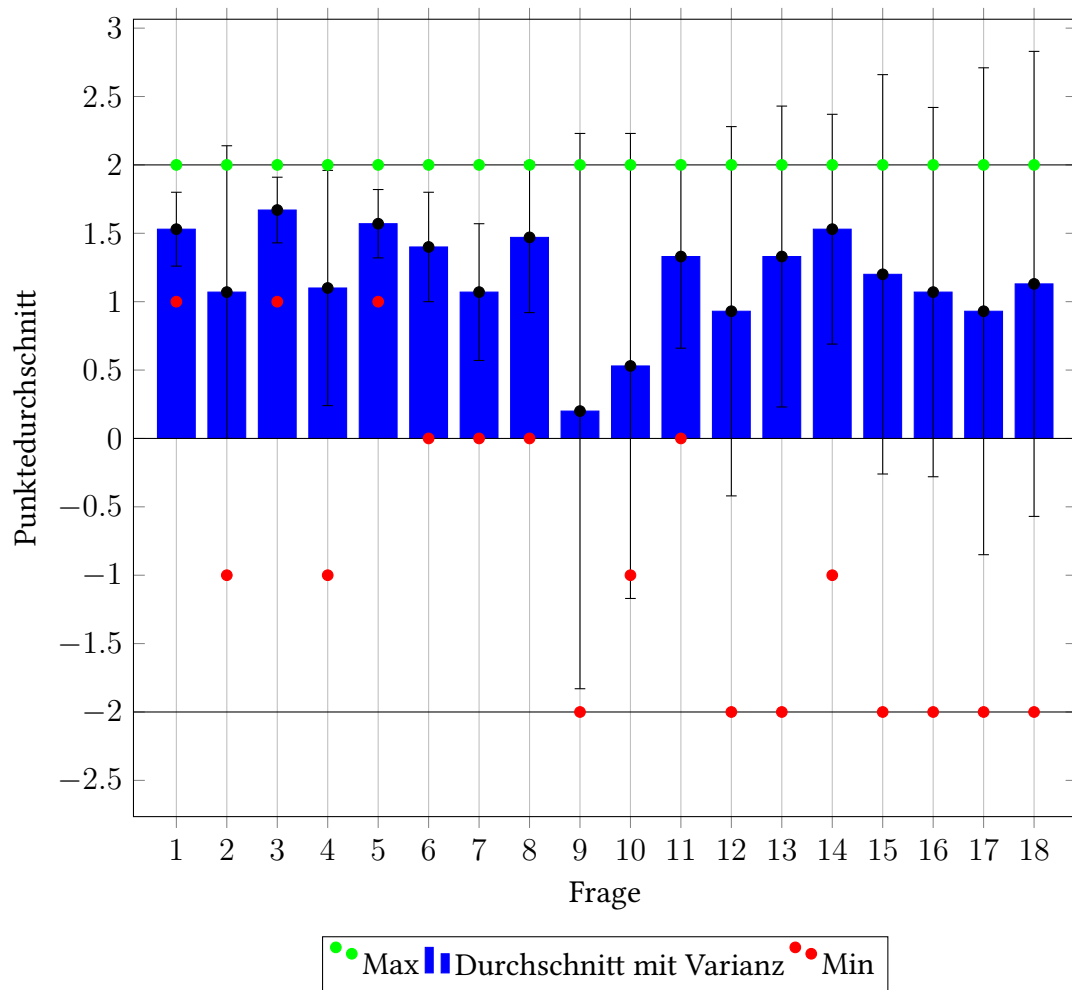


Abbildung 9.1: Auf zwei Nachkommastellen gerundeter Punktedurchschnitt pro Frage

10 Diskussion und zukünftige Verbesserungen

Sowohl während der Entwicklung als auch nach Abschluss der Evaluation kamen Ideen und Verbesserungsvorschläge für VA.SI.LI-MINDMAP auf. Diese reichten von einfachen Bedienungs-erleichterungen bis hin zu dem Hinzufügen weiterer Komponenten. Im Folgenden werden diese noch einmal erläutert.

10.1 Modularität

Verbesserungen an der Modularität könnten in Unity in Form von dynamischen Klassen für den Import der BSON-Dateien eingebaut werden. Dies würde zu einer Reduktion der Arbeit bei Veränderungen der Datenquellen in der Datenbank führen. Des Weiteren könnten Anpassungen vorgenommen werden, dass Inhalte der Datenstruktur der Datenbank dynamisch gelesen und daraus Knoten-Objekte in Unity erstellt werden könnten. Dies ist im Moment nur für 3 vorgegebene Datentypen (Text, Bild und Audio) möglich und könnte durch zum Beispiel Video, interaktive Tabellen und Segmente, und Animationen innerhalb der Virtual Reality Komponente erweitert werden.

10.2 Weitere Datenquellen

Anstatt die hier erstellte Website zu benutzen, um Aufgaben zu erstellen und in das Datenformat der Datenbank zu übersetzen, könnten zum Beispiel andere Datenquellen wie bereits erstellte Aufgaben auf Papier (digital) oder ähnliche Aufgabenarten per Servermethoden in das Datenformat übersetzt werden. Dies würde eine eigens für die VR-Komponente benötigte Erstellung der Aufgaben überflüssig machen, und könnte dann aus vielen bereits bestehenden Quellen Aufgaben ziehen.

10.3 NLP Methoden

Es könnten Methoden des „Natural Language Processing“, auch NLP, für eine vereinfachte Erstellung der Aufgaben eingebaut werden. Diese sollten in der Lage sein, bei Auswahl bestimmter Aufgabentypen von Nutzern eingegebene Inhalte zu analysieren und selbstständig Knoten und Verbindungen zu erstellen. Dies wäre zum Beispiel die automatische Aufgabengenerierung aus einem eingegebenen Text heraus, oder der Bewertung und Einordnung von erstellten Aufgaben an einem Schwierigkeitsgrad.

10.4 Zusätzliche Datentypen

Zusätzliche Datentypen wie Videodateien, interaktive Elemente und Kommunikationsmöglichkeiten durch Elemente hindurch, könnten ein besseres Lernerlebnis erzeugen. Kommu-

nikative Elemente könnten Notizsammlungen sein, die durch andere Teilnehmende editiert werden könnten.

10.5 Erweitertes Erstellen eigener Mindmaps

Das Erstellen eigener Mindmaps existiert in VA.SI.LI-MINDMAP nur in Form vorgegebener Knoten mit Inhalten. Die Verbindungsbeschriftungen sind ebenfalls vorzugeben. User können Knoten anordnen, verbinden und Verbindungsbeschriftungen nach eigenen Wünschen auswählen, jedoch keine neuen Knoten erstellen, sowie eigenen Knoteninhalte einfügen. Dies wäre eine geeignete Verbesserungsmöglichkeit.

10.6 Einbindung der kollaborativen Komponente von Va.Si.Li-Lab

Zum Zeitpunkt der Evaluation konnten Mindmaps nur von jeder Person individuell benutzt werden. Während Va.Si.Li-Lab eine kollaborative Komponente besitzt, in der Personen miteinander an Dingen arbeiten können, ist diese Komponente in VA.SI.LI-MINDMAP nicht fertiggestellt.

10.7 Erweiterung von VA.SI.LI-MINDMAP für Kontexte außerhalb des Lernens

Wie in Kap. 4 in der Sektion 4.5 beschrieben, ließe sich VA.SI.LI-MINDMAP auch für Möglichkeiten außerhalb des Lehrbereichs verwenden. Es existiert bereits ein Programmteil, mit dem sich vernetzte Graphen automatisch generieren und verbinden lassen; es fehlt jedoch noch eine automatische Platzierung der Knoten. Dies kann sich je nach Räumlichkeiten unterscheiden, auf denen die Datenstruktur generiert wird, da eine größere Datenstruktur idealerweise nicht auf der hier verwendeten Kugelaußenfläche generiert wird. Dies eignet sich nur für das gleichzeitige Anzeigen kleinerer Netzwerke für eine Person, was in dieser These hauptsächlich verwendet wurde.

11 Fazit

Das Ziel dieser These war das Entwickeln einer Virtual Reality Applikation und einer Schnittstelle zum leichten Erstellen von Aufgaben in einem Mindmapformat für die Verwendung in Virtual Reality. Dieses Ziel wurde im Verlauf der These mit VA.SI.LI-MINDMAP erreicht, und an einer signifikanten Auswahl von Probanden getestet.

Die Komponenten von VA.SI.LI-MINDMAP haben gute Bewertungen erzielt (siehe Kap. 9) und wurden als ansprechendes Lehrmittel empfunden.

Während sie Aspekte enthalten, die es in vergleichbaren Applikationen nicht gibt (siehe Kap. 1.2), unterliegt VA.SI.LI-MINDMAP im Vergleich zu ein paar anderen Applikationen jedoch in einigen sich gleichenden Punkten, wie dem Erstellen eigener Knoten in VR und einer besseren Benutzeroberfläche auf der Website. Dies war jedoch vorherzusehen entsprechend des Umfangs der These und dem Fokus auf einige spezielle Aspekte. Durch die begrenzte Entwicklungszeit während der Bachelorarbeit sind einige Verbesserungs- und Erweiterungsmöglichkeiten offen geblieben, die in zukünftigen Arbeiten aufgegriffen werden könnten.

12 Anhang

Umfragebogen zur Nutzererfahrung in "Virtualisierung von Mindmaps in Va.Si.Li-Lab" von Jasper Hustedt

1 Virtual Reality Anwendung

	Wenig				Sehr/Viel
1. Wie verständlich war die Anwendung?	-2	-1	0	1	2
2. Wie sicher war ich im Umgang mit der Anwendung?	-2	-1	0	1	2
3. Wie schnell konnte ich die Benutzung erlernen?	-2	-1	0	1	2
4. War die Anwendung visuell ansprechend?	-2	-1	0	1	2
5. Wurde mein Interesse an der Anwendung geweckt?	-2	-1	0	1	2
6. Hat die Anwendung das Interesse am Lernen gesteigert?	-2	-1	0	1	2
7. Wie gut kann die Anwendung mein Lernen unterstützen?	-2	-1	0	1	2
8. Ich würde die Anwendung erneut benutzen	-2	-1	0	1	2
9. Meine Vorkenntnisse mit Virtual Reality	-2	-1	0	1	2
10. Benötigt die Anwendung wenig Vorkenntnisse in Virtual Reality?	-2	-1	0	1	2
11. Hat die Anwendung bereits bestehende Funktionen gut integriert?	-2	-1	0	1	2
12. Hat die Anwendung ausreichend Funktionen integriert?	-2	-1	0	1	2
13. War die Website verständlich?	-2	-1	0	1	2
14. War die Website einfach zu benutzen?	-2	-1	0	1	2

15. Konnte man mit der Website Aufgaben gut umsetzen?

-2	-1	0	1	2
----	----	---	---	---
16. Waren auf der Website ausreichend Funktionen integriert?

-2	-1	0	1	2
----	----	---	---	---
17. Wie verständlich war die Beziehung zwischen der VR Anwendung und der Website?

-2	-1	0	1	2
----	----	---	---	---
18. Ist die Art der Aufgabenerstellung auf der Website eine gute Wahl für die Erstellung der VR Aufgaben?

-2	-1	0	1	2
----	----	---	---	---

2 Verbesserungsvorschläge und Anmerkungen:

Abbildung 12.1: Fragebogen zur Evaluation

MindMapController
<pre> standardNodePrefab : GameObject connectionObjectPrefab : GameObject textObjectPrefab : GameObject functionButtonPrefab : GameObject selectionMaterial : Material preDefinedConnectionTags : List<string> radius : float mindMapMidPoint : Vector3 timeInSeconds : float sheetDescriptions : List<string> leftHand : Transform rightHand : Transform currentTask : GameObject sheetPointers : List<int> currentPointer : int lineToObjects : Dictionary<GameObject, Tuple<GameObject, GameObject> selectionToConnect : GameObject </pre>
<pre> «get/set» MindMapMidPoint : Vector3 «get/set» LeftHand : Transform «get/set» RightHand : Transform «get/set» CurrentTaskJSON : JsonTask «get/set» CurrentTask : GameObject «get/set» CurrentPointer : int «get/set» LineToObjects : Dictionary<GameObject, Tuple<GameObject, GameObject> «get/set» SelectionToConnect : GameObject Awake() : void Update() : void createTask(jsonTask : JsonTask) : GameObject SetNodeOrderByParentChildRelations(jsonNodes : JsonNode[]) : List<JsonNode> GetNodeWithLeastUnorderedParents(unorderedNodes : List<JsonNode>, orderedNodes : List<JsonNode>) : JsonNode GetUnorderedParentAmount(jsonNode : JsonNode, unorderedNodes : List<JsonNode>) : int randomizeNodeOrder(jsonNodes : JsonNode[]) : List<JsonNode> SetCurrentUnusedNodePositionsTest2() : void SwitchSheet() : void CreateConnectionTagDisplayString() : string NextNode() : void EmptyNode() : void ConnectNodeToAllExistingConnectionPartners(node : Node) : void ConnectOrDisconnectNodes(node1 : GameObject, node2 : GameObject) : GameObject DisconnectNodeConnections(node : GameObject) : void SpawnAllNodes(jsonNodes : List<JsonNode>, unusedNodes : GameObject) : void SpawnNodeEnumerator(jsonNode : JsonNode, unusedNodes : GameObject) : IEnumerator SpawnNode(jsonNode : JsonNode) : GameObject CompareSolutions() : void GetNearestPositionOnBorder(oldPosition : Vector3) : Vector3 LoadAudio(sourceNode : JsonNode, audioNode : GameObject) : void CurrentSheet() : GameObject CurrentConnectionParent() : GameObject CurrentUnusedNodesParentGameObject() : GameObject CurrentUsedNodesParentGameObject() : GameObject CurrentSheetJson() : JsonSheet switchPagePointer() : void </pre>

Abbildung 12.3: Klasse MindMapController

Node
<pre> follow : Hand graspPointGameObject : GameObject body : Rigidbody mindMapMidPoint : Vector3 usedThisStep : bool graspedThisStep : bool meshRenderers : MeshRenderer[] ownMaterials : Dictionary<MeshRenderer, Material> nodeID : string </pre>
<pre> «get/set» NodeID : string «get/set» MeshRenderers : MeshRenderer[] «get/set» OwnMaterials : Dictionary<MeshRenderer, Material> Awake() : void Update() : void LateUpdate() : void SetNodeOwnMaterials() : void </pre>

Abbildung 12.4: Klasse Node

NodeConnection
ownMaterial : Material meshRenderers : MeshRenderer[] isHighlighted : bool isCollidingWithHand : bool availableConnectionTags : List<string> currentConnectionTagIndex : int currentConnectionTag : string
«get/set» CurrentConnectionTag : string «get/set» ConnectionTagChanger : FunctionButton «get/set» AvailableConnectionTags : List<string> Awake() : void Start() : void Update() : void OnTriggerStay(other : Collider) : void DisableHighlighting() : void FixedUpdate() : void SwitchCurrentConnectionTag() : void SetCurrentConnectionTag(index : int) : void

Abbildung 12.5: Klasse NodeConnection

Literatur

- Abrami, Giuseppe, Alexander Mehler, Mevlüt Bağcı u. a. (2023). „Va.Si.Li-Lab as a Collaborative Multi-User Annotation Tool in Virtual Reality and Its Potential Fields of Application“. In: *Proceedings of the 34th ACM Conference on Hypertext and Social Media*. HT '23. Rome, Italy: Association for Computing Machinery. ISBN: 979-8-400-70232-7. DOI: 10.1145/3603163.3609076. URL: <https://doi.org/10.1145/3603163.3609076>.
- Abrami, Giuseppe, Alexander Mehler und Christian Spiekermann (2019). „Graph-based Format for Modeling Multimodal Annotations in Virtual Reality by Means of VAnnotatoR“. In: *Proceedings of the 21th International Conference on Human-Computer Interaction, HCII 2019*. Hrsg. von Constantine Stephanidis und Margherita Antona. HCII 2019. Orlando, Florida, USA: Springer International Publishing, S. 351–358. ISBN: 978-3-030-30712-7.
- Aditya, David Sulistiawan (2021). „Embarking Digital Learning Due to COVID-19: Are Teachers Ready?“. In: *Journal of Technology and Science Education* 11.1, S. 104–116.
- Giraudeau, Philippe und Martin Hachet (2017). „Towards a Mixed-Reality Interface for Mind-Mapping“. In: *Proceedings of the 2017 ACM International Conference on Interactive Surfaces and Spaces*. ISS '17. Brighton, United Kingdom: Association for Computing Machinery, S. 384–389. URL: <https://doi.org/10.1145/3132272.3132275>.
- Grinshkun, A.V., M.S. Perevozchikova, E.V. Razova und I.Y. Khlobystova (Juni 2021). „Using Methods and Means of the Augmented Reality Technology When Training Future Teachers of the Digital School“. In: *European Journal of Contemporary Education* 10, S. 358–374.
- Kučák, David u. a. (2019). „An Interactive and Multimodal Virtual Mind Map for Future Workplace“. In: *Frontiers in ICT* 6. URL: <https://www.frontiersin.org/articles/10.3389/fict.2019.00014>.
- LaValle, Steven M (2023). *Virtual reality*. Cambridge university press.
- Mehler, Alexander u. a. (2023). „A Multimodal Data Model for Simulation-Based Learning with Va.Si.Li-Lab“. In: *Digital Human Modeling and Applications in Health, Safety, Ergonomics and Risk Management*. Hrsg. von Vincent G. Duffy. Cham: Springer Nature Switzerland, S. 539–565. ISBN: 978-3-031-35741-1. DOI: 10.1007/978-3-031-35741-1_39.
- MongoDB (2024). *JSON and BSON*. <https://www.mongodb.com/json-and-bson> [Accessed: (2024-01-17)].
- MongoDB, Inc (2009). *MongoDB: The Developer Data Platform*. <https://www.mongodb.com> [Accessed: (2023-12-27)].
- Radianti, Jaziar, Tim A Majchrzak, Jennifer Fromm und Isabell Wohlgenannt (2020). „A systematic review of immersive virtual reality applications for higher education: Design elements, lessons learned, and research agenda“. In: *Computers & Education* 147, S. 103778.
- Rezapour-Nasrabad, Rafat (Aug. 2019). „Mind Map Learning Technique: An Educational Interactive Approach“. In: *International Journal of Pharmaceutical Research* 11, S. 1593–1597.
- Saxena, H. ((PhD) Student) und B. (Promotor) Signer (2022). „Enhancing Learning Experience With Augmented Mind Maps“. Magisterarb. Vrije Universiteit Brussel.

- Sims, Robert und Abhijit Karnik (2021). „VERITAS: Mind-Mapping in Virtual Reality“. In: *2021 7th International Conference of the Immersive Learning Research Network (iLRN)*, S. 1–8.
- Stackoverflow, Benutzer: Çağatay IŞIK (2023). *How to convert base64 mpeg to AudioClip in Unity?* <https://stackoverflow.com/questions/75124607/how-to-convert-base64-mpeg-to-audioclip-in-unity> [Accessed: (2023-10-27)].
- Stackoverflow, Benutzer: cuixiping (2023). *How to convert Base64 String to javascript file object like as from file input form?* <https://stackoverflow.com/questions/35940290/how-to-convert-base64-string-to-javascript-file-object-like-as-from-file-input-f> [Accessed: (2023-10-24)].
- Tittmann, Peter (2003). „Graphentheorie“. In: *Hanser Fachbuchverlag 2*.
- Unity Technologies (2023). *Unity Real-Time Development Platform*. <https://unity.com> [Accessed: (2023-12-27)].
- University College London, Virtual Environments and Computer Graphics Group (2023). *Ubiq*. <https://github.com/UCL-VR/ubiq> [Accessed: (2023-12-27)].
- ValveSoftware (2015). *OpenVR*. <https://github.com/ValveSoftware/openvr> [Accessed: (2024-01-18)].
- Veletsianos, George (2016). *Emergence and innovation in digital learning: Foundations and applications*. Athabasca University Press.
- W3 Schools (2023). *How TO - Create a Draggable HTML Element*. https://www.w3schools.com/howto/howto_js_draggable.asp [Accessed: (2023-04-14)].
- Wendel, Per (2015). *Spark - a tiny web framework for Java 8*. <https://github.com/perwendel/spark> [Accessed: (2023-12-27)].