

Bachelorarbeit

Showroom für den zukunft.bahnhof Lichtensteig

Frühjahrssemester 2023



ZUKUNFT.BAHNHOF

Version: 01.00

Datum: 2023-06-15 20:21:25Z

Git Version: 70182ec

Autoren: Simon Canal
Linard Vincenz

Betreuer: Prof. Dr.-Ing. Frieder Loch
Projektpartner: Martin Hohn (ATMA.life)
Experte: Prof. Dr. Daniel P. Politze
Gegenleser: Adrian Scherrer



OST – Ostschweizer Fachhochschule
Campus Rapperswil-Jona

Abstract

Das Ziel dieser Arbeit ist es, in einem Showroom das Projekt zukunft.bahnhof in digitaler Form erlebbar zu machen. Dazu gehört sowohl die Entwicklung eines Konzepts für ein interaktives Besuchererlebnis, wie auch die prototypische Implementierung in Form einer Applikation. Zudem soll der gewählte Lösungsansatz kritisch reflektiert werden, um Möglichkeiten zur Weiterentwicklung und iterativen Verbesserung aufzuzeigen.

Mithilfe einer Literaturrecherche wurden mögliche Realisierungsansätze sowie Technologien erfasst. Eine Nutzwertanalyse sowie ein Workshop mit einer Fokusgruppe diente dem Vergleich der Ansätze sowie der Wahl eines zu realisierenden Produkts. Als Teil eines Gesamtkonzepts für den Showroom, wurde eine [Virtual Reality](#) Applikation in Form einer WebApp realisiert. Die Applikation besteht dabei aus einem [NGINX](#) Reverse Proxy, welcher sowohl das [React](#) Frontend, wie auch das ASP.Net Core Backend bereitstellt. Das Areal von Lichtensteig ist in einem [Unity](#) Projekt modelliert. Die Daten werden in einer PostgreSQL Datenbank via Entity Framework Core persistiert. Alle Komponenten werden mittels Continuous Delivery getestet, gebildet und in Docker Containern ausgeliefert.

Der Prototyp dieser [VR](#) Applikation für den Showroom ermöglicht es den Besuchenden auf eindrucksvolle und spielerische Art Informationen zum Projekt zukunft.bahnhof zu vermitteln. Benutzende können dabei das Projektareal in Lichtensteig virtuell betrachten und sich spezifisch zu den einzelnen Teilen des Projekts erkunden. Durch die Möglichkeit der eigenen Arealgestaltung, sowie Kommentar- und Feedback-Funktionen können sich Besuchende des Showrooms partizipativ im Projekt einbringen.

Da der Prototyp als WebApp realisiert wurde, kann die Applikation auch auf mobilen Geräten genutzt werden. Somit kann das Nutzererlebnis auch nach einem Besuch im Showroom noch fortgeführt werden.

Inhaltsverzeichnis

1	Management Summary	1
1.1	Ausgangslage	1
1.2	Vorgehen	1
1.3	Ergebnisse	2
2	Einleitung	3
2.1	Ausgangslage	3
2.2	Ziel der Arbeit	4
2.3	Rahmenbedingungen	5
2.4	Vorgehensmodell	6
3	Realisierungsansätze	7
3.1	Analyse und Begriffserklärungen	7
3.2	Vorgehen	10
3.3	Realisierungsansätze	11
4	Auswahl eines Realisierungsansatzes	17
4.1	Kriterien	17
4.2	Gewichtung	19
4.3	Bewertung	19
4.4	Auswertung der Ergebnisse	25
5	Konzept Showroom	26
5.1	Workshop mit Fokusgruppe	26
5.2	Konzeptentwurf	27
5.3	Abgrenzung	29
6	Anforderungsanalyse	30
6.1	Funktionale Anforderungen	30
6.2	Nicht-Funktionale Anforderungen	32
7	Evaluierung der Technologien	35
7.1	Begriffserklärung	35
7.2	Technologiewahl	36
7.3	Experimente	39
8	Design des Prototypen	44
8.1	Domain Model	44
8.2	Systemarchitektur	45
8.3	Wireframes	49
9	Implementierung	52
9.1	Hardware	52
9.2	Technologiestack	52
9.3	Testing	54
9.4	Unity	57

9.5	React-Unity-Webgl	58
9.6	Continuous Integration und Continuous Delivery	59
10	Usability Test	63
10.1	Ablauf	63
10.2	Ergebnisse	63
11	Ergebnisse	65
11.1	Umgesetzte Anforderungen	65
11.2	Zielerreichung der Arbeit	66
11.3	Vorstellung des Prototyps	66
11.4	Ausblick und Erweiterungsmöglichkeiten	71
11.5	Operationelle Hinweise	73
A	Verzeichnisse	75
	Abbildungsverzeichnis	75
	Tabellenverzeichnis	77
	Literaturverzeichnis	78
	Glossar	82
	Abkürzungsverzeichnis	84
B	Projektplanung	85
B.1	Projektorganisation	85
B.2	Zeitplanung	85
B.3	Planung	86
B.4	Besprechungen	86
B.5	Risikomanagement	87
B.6	Qualitätsmassnahmen	89
B.7	Projektmonitoring	90
C	Anhang	93
C.1	Screenshots Prototyp	93
C.2	Ergebnisse Workshop Fokusgruppe	101
C.3	Usability Test	103
C.4	Testabdeckung	107

1. Management Summary

Das Management Summary gibt einen Überblick über die Arbeit und fasst deren Ergebnisse zusammen.

1.1 Ausgangslage

Das Projekt zukunfft.bahnhof gestaltet in Lichtensteig (SG) ein Areal, welches ein Entwurf für eine regenerative, zukunftsfähige Gesellschaft sein wird. Da das Projekt eine Brücke zur digitalen Welt schlagen und während der Planungs- und Bauphase eine innovative Pioniernutzung realisieren will, soll ein Konzept für ein interaktives Besucherlebnis entwickelt werden.

Eine Literaturrecherche soll zur Identifikation ähnlicher Konzepte und möglicher Technologien durchgeführt werden. Zudem soll in Erfahrung gebracht werden, wie man Inhalte digital erlebbar machen kann, sodass Informationen spannend und nachhaltig vermittelt werden können.

Es sollen verschiedene Ansätze zur Erreichung eines interaktiven Besucherlebnisses konzeptionell ausgearbeitet werden. Anhand eines Vergleichs dieser verschiedenen Realisierungsansätze soll ein Ansatz zur Umsetzung ausgewählt werden. Dieser soll in Form eines Prototyps implementiert und anhand von Feedback potenzieller Nutzergruppen verfeinert werden. Zudem soll der Prototyp kritisch reflektiert werden, um Möglichkeiten zur Weiterentwicklung und iterativen Verbesserung aufzuzeigen.

1.2 Vorgehen

Da diese Arbeit kein Projekt mit vordefinierten Anforderungen ist wurde ein eigenes Vorgehensmodell dafür entwickelt.

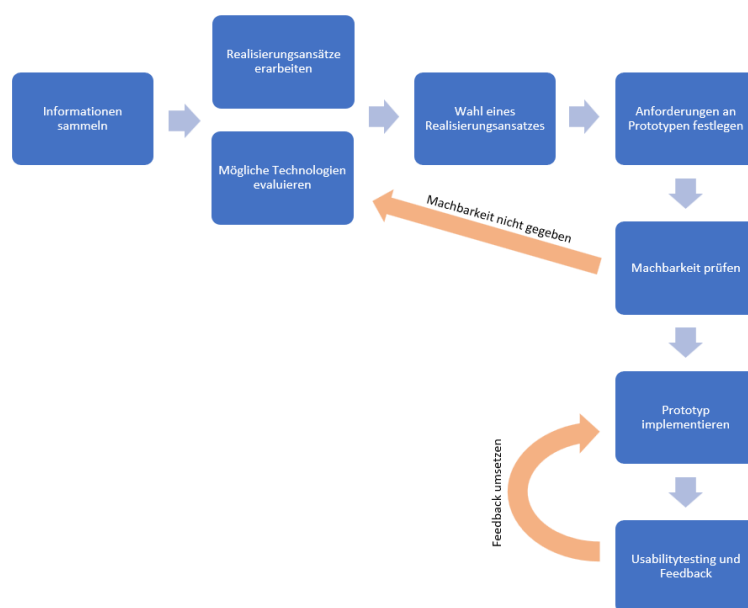


Abbildung 1.1: Gewähltes Vorgehensmodell (Quelle: Eigene Darstellung)

1.3. ERGEBNISSE

Wie der [Abbildung 1.1](#) entnommen werden kann, werden die meisten Schritte sequentiell abgearbeitet. Lediglich die Erarbeitung der Realisierungsansätze, sowie die Evaluierung der möglichen Technologien wird parallel durchgeführt.

Die Wahl des umzusetzenden Realisierungsansatzes wird anhand einer Nutzwertanalyse, sowie einem Workshop mit einer Fokusgruppe durchgeführt. Nach der Festlegung der Anforderungen an den Prototypen, wird eine Machbarkeitsprüfung anhand von Experimenten durchgeführt. Das bedeutet, dass Anforderungen in Form eines [Proof of Concept](#) implementiert werden um zu prüfen, ob der Anforderungskatalog mit den gewählten Technologien umgesetzt werden kann. Ist dies nicht möglich wird eine Reevaluierung der Technologien durchgeführt.

Bei erfolgreichem Abschluss aller Experimente, wird der gewählte Realisierungsansatz implementiert und anhand von Feedback aus einem Usabilitytesting verbessert.

1.3 Ergebnisse

Im umgesetzten Prototypen konnten, trotz relativ kurzer Implementierungsphase, alle zwingenden Anforderungen umgesetzt werden. Sogar von den optionalen Anforderungen wurden einige Punkte umgesetzt.

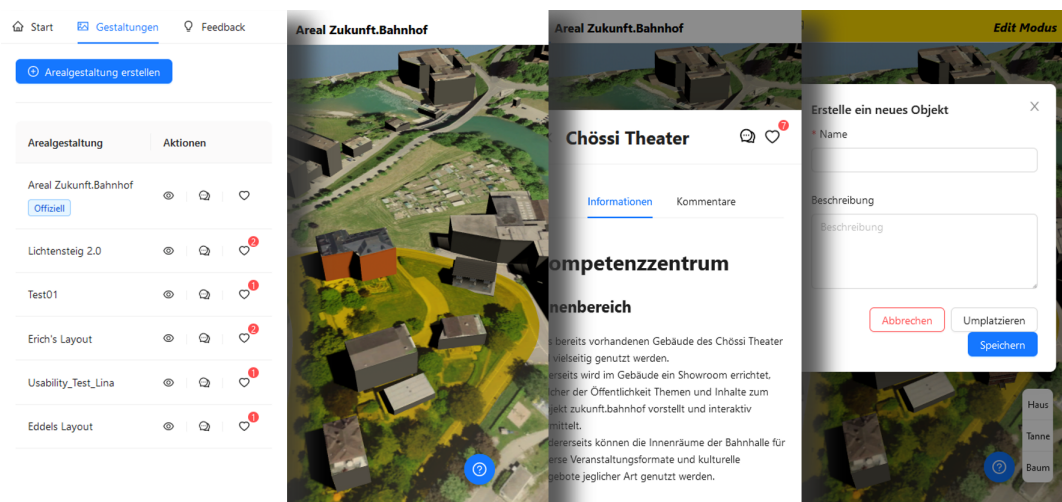


Abbildung 1.2: Impressionen des Prototyps (Quelle: Eigene Darstellung)

Diese Arbeit bringt als Ergebnis ein Konzept eines Showrooms, sowie als Teil davon einen Prototypen einer Web Applikation (<https://srbsci-155.ost.ch>) hervor. Dieser kann in einem Showroom auf einem grossen Monitor, als auch auf mobilen Geräten genutzt werden. Der Prototyp ermöglicht Anwohnern von Lichtensteig, wie auch anderen interessierten Personen sich über das Projekt zukunft.bahnhof auf moderne und eindrückliche Art und Weise zu informieren.

Dazu können vordefinierte Arealgestaltungen in Form von virtuellen 3D Modellen betrachtet werden. Über einzelnen Aspekte, sowie die umzusetzenden baulichen Massnahmen des Projekts, lassen sich dabei gezielt Informationen abrufen.

Zudem können sich Besuchende im Projekt partizipativ einbringen indem sie Dinge kommentieren und bewerten oder direkt eine eigene Arealgestaltung entwerfen, um zu zeigen wie das Bahnhofsareal ihren Wünschen entsprechend umgestaltet werden sollte.

2. Einleitung

In diesem Kapitel wird einerseits die Ausgangslage und das Projekt [zukunft.bahnhof \(ZB\)](#) näher beleuchtet, andererseits wird auf die Aufgabenstellung und die damit verbundenen Ziele eingegangen.

2.1 Ausgangslage

Mit dem Projekt [ZB](#) entsteht in Lichtensteig eine Arealentwicklung, welche ein Modell für eine regenerative, enkeltaugliche Gesellschaft sein soll.

Das Projekt soll auf inspirierende Art zeigen, dass Veränderung möglich ist und diese erlebbar machen. Diese Erlebnisse soll den Besuchenden des Areals, die hier eine Veranstaltung besuchen, in einem Hotelzimmer übernachten, oder auch nur etwas in der Gastronomie essen, ermöglicht werden.

2.1.1 Projekt [zukunft.bahnhof](#)

Für eine enkeltaugliche Zukunft mit mehr Lebensqualität braucht es nachhaltige und gemeinschaftliche Wohnformen, die Raum für Begegnung und Entfaltung bieten sowie Wohnen und Arbeiten kreativ verbinden.

Die [Abbildung 2.1](#) zeigt eine Übersicht über die verschiedenen Teilbereiche des Projekts [ZB](#).



Abbildung 2.1: Konzept Projekt [ZB](#) (Quelle: [zukunft.bahnhof](#))

2.3. RAHMENBEDINGUNGEN

Lösungsansatz zur Umsetzung ausgewählt werden. Dieser soll in Form eines Prototyps implementiert und anhand von Feedback potenzieller Nutzergruppen verfeinert werden. Zudem soll der gewählte Lösungsansatz kritisch reflektiert werden, um Möglichkeiten zur Weiterentwicklung und iterativen Verbesserung aufzuzeigen.

2.3 Rahmenbedingungen

Diese Arbeit wird als Bachelorarbeit im Studiengang Informatik an der OST - Ostschweizer Fachhochschule durchgeführt.

2.3.1 Zeitdauer

Der Beginn der Bachelorarbeit ist der 20. Februar 2023. Die Abgabe kann bis zum 16. Juni 2023 um 17.00 Uhr erfolgen.

Projektdauer:	17 Wochen
Projektbeginn:	20. Februar 2023
Projektende:	16. Juni 2023, 17.00 Uhr

Tabelle 2.1: Zeitdauer

2.3.2 Zeitbudget

Die Teammitglieder beteiligen sich mit jeweils 360 Arbeitsstunden am Projekt. Dies ergibt pro Woche einen Arbeitsaufwand von ca. 21,3 Stunden. Somit stehen dem Projekt insgesamt 720 Stunden zur Verfügung. Dieses Zeitbudget soll, gemäss Leitfaden der OST, nicht mit mehr als 20% über- bzw. unterschritten werden.

Anzahl Projektmitarbeiter:	2
Aufwand pro Woche und Person:	21,2 Stunden
Aufwand Total:	720 Stunden

Tabelle 2.2: Zeitbudget

2.3.3 ECTS Vergütung

Die Bachelorarbeit wird mit 12 ECTS Punkten vergütet. Dies entspricht dem geleisteten Aufwand von 360 Stunden, bei gängiger Berechnung von 30 Stunden Aufwand pro ECTS Punkt.

2.4 Vorgehensmodell

Zuerst werden Informationen zum Projekt *ZB*, wie dessen Stakeholder, Ziele und den bisherigen Projektverlauf gesammelt. Auch über Vorgehensweisen und den aktuellen Stand der Forschung im Bereich der transformativen Lernerfahrung werden Informationen gesammelt, sodass ein Verständnis dafür hervorgeht, wie ein Showroom gestaltet werden kann, welcher bei den Besuchenden für ein Aha-Erlebnis und nachhaltige Erfahrungen sorgt.

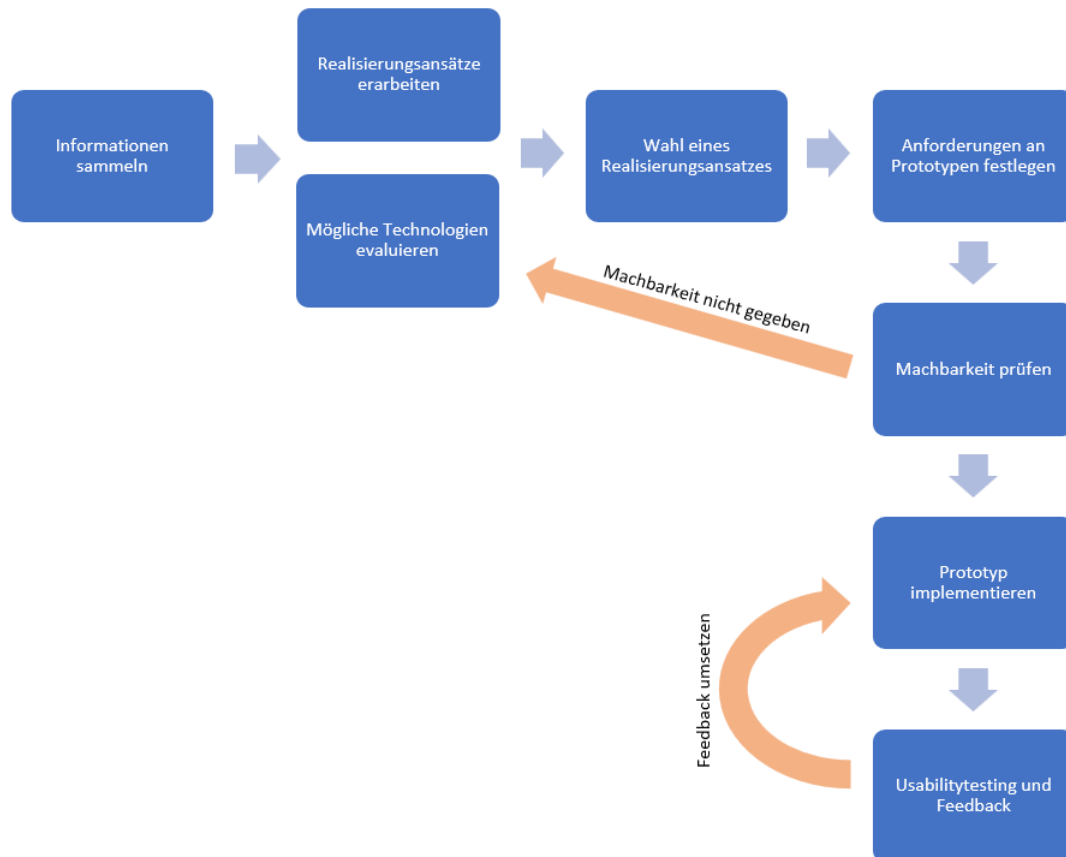


Abbildung 2.3: Gewähltes Vorgehensmodell (Quelle: Eigene Darstellung)

Das Erarbeiten der möglichen Realisierungsansätze sowie das Evaluieren der möglichen Technologien wird parallel durchgeführt. Nach der Wahl eines Realisierungsansatzes wird ein Anforderungskatalog an den zu implementierenden Prototypen aufgesetzt. Die Anschließend durchgeführten Experimente dienen der Prüfung der Machbarkeit und sorgen dafür, nicht umsetzbare Anforderungen früh zuerkennen. Falls also bei der Durchführung der Experimente Probleme auftreten, werden erneut mögliche Technologien evaluiert. Wurden alle Experimente erfolgreich durchgeführt, wird mit der Implementierung begonnen. Abschliessend wird der Prototyp anhand von Feedback potenzieller Benutzenden optimiert. Dazu wird ein Usabilitytesting durchgeführt, sowie allgemeines Feedback zum Prototypen eingeholt. Anpassungen anhand dieses Feedbacks werden bei Möglichkeit direkt am Prototypen umgesetzt.

3. Realisierungsansätze

Um einen Lösungsansatz für die Problemstellung zu entwickeln, werden mögliche Realisierungsansätze erarbeitet. Ein Realisierungsansatz beschreibt konzeptionell eine mögliche Lösung zur Umsetzung des Showrooms. Die Realisierungsansätze werden nur grob umschrieben, ausreichend für eine Bewertung und Auswahl zur Umsetzung.

Damit fundierte Ideen ausgearbeitet werden können müssen zuerst einige Begrifflichkeiten geklärt werden.

3.1 Analyse und Begriffserklärungen

Damit mögliche Realisierungsansätze erarbeitet werden können, muss zuerst verstanden werden, wie die [Ziele](#) erreicht werden können.

3.1.1 Transformative Lernerfahrungen

Der Showroom soll den Besuchenden eine transformative Lernerfahrungen bieten. Jack Mezirow gilt als Begründer der Theorie des transformativen Lernens, welche in den 1970er Jahren in den USA entstand und danach in Kanada und Südamerika weiterentwickelt wurde. [\[30\]](#)

Mezirow unterscheidet zwischen Lernen, bei welchem neues Wissen aufgenommen wird und transformativem Lernen, bei dem vorhandenes Wissen reorganisiert wird und so problematische Bedeutungsstrukturen transformiert werden. Während es also beim Lernen generell um die Aneignung von Wissen geht, zielt das transformative Lernen auch auf die Veränderung der Weltsicht ab. [\[18\]](#)

Rationaler Diskurs und kritische Reflexion

Der rationale Diskurs zählt zusammen mit der Erfahrung und der kritischen Reflexion zu den zentralen Elementen dieser Lernform. Durch ihn können Menschen mit ihrer Außenwelt in Kontakt treten und individuelle Erfahrungen austauschen. Zudem werden sie angeregt, ihre eigenen Sichtweisen zu hinterfragen und auf ihre Gültigkeit zu überprüfen. [\[27\]](#)

Um eine transformative Lernerfahrung zu schaffen, müssen die Besuchenden also dazu angeregt werden, ihre vorherrschenden Sichtweisen und Weltbilder zu hinterfragen und kritisch zu reflektieren. Dies kann beispielsweise erreicht werden, indem ein sachlicher Dialog in einer Gruppe entsteht. Da der Showroom öffentlich zugänglich sein wird, werden sich die unterschiedlichsten Personengruppen treffen. Dadurch bietet sich die Möglichkeit verschiedene Ansätze zu realisieren, die zu einem spannenden Diskurs einladen. Oftmals werden sich wahrscheinlich einzelne Personen oder Familien im Raum einfinden. Durch die gemeinsame Interaktion mit der Ausstellung können spannende Diskussionen stattfinden, die gemeinsam neue Ideen entstehen lassen.

3.1.2 Präsenzerleben und Immersion

Der Ansatz des transformativen Lernens soll dabei unterstützen, das Projekt ZB digital erlebbar zu machen. Schlussendlich soll der Showroom jedoch vor allem eines und zwar interessierten Personen das Projekt näher bringen und aufzeigen welche Veränderungen, Benefits und Innovationen es mit sich bringt. Kolb beschreibt im Konzept des Experiential Learning, dass Wissen durch die Umwandlung von Erlebnissen entsteht. [7] Über Erlebnisse und Erfahrungen können wir Menschen also demnach gut Wissen aneignen. Etwas erlebbar und greifbar zu machen ist jedoch keine einfache Aufgabe.

Hier kommt der Begriff Präsenzerleben ins Spiel. Präsenzerleben ist ein Merkmal von AR und VR und beschreibt das Gefühl des Eintauchens in eine virtuelle Umgebung. Der Begriff Immersion wird oft synonym dafür verwendet. Es vermittelt dem Nutzer das Gefühl, die reale Welt zu verlassen und nur in der virtuellen Umgebung präsent zu sein. In der Filmbranche wird auch teils der Begriff Transportation verwendet, welcher sprechend darstellt, dass man während der Nutzung in eine andere Welt transportiert wird. [8]

Eine hohe Anzahl an Interaktionsmöglichkeiten und deren Realitätsnähe tragen zur Entstehung von Präsenzerleben bei. Stimulierende Medieninhalte wie die Kameraperspektive, Sound und Haptik sowie körperliche Aktionen können das Präsenzerleben zusätzlich steigern. Auch soziale Elemente, wie Gruppenerlebnisse tragen dazu bei, dass sich Nutzende transportiert fühlen. Dabei entscheiden die Natürlichkeit der Objekte, der Umgebung und den Interaktionsmöglichkeiten darüber, in welchem Masse Präsenzerleben geschaffen werden kann. [8]

Immersive Medien können die Nutzenden dazu bringen, eine tiefere Verbindung mit dem Medium herzustellen. Dadurch können Inhalte auf eindruckliche Art und Weise präsentiert werden, was dafür sorgt, dass Nutzende ein einschneidendes Nutzererlebnis haben. Dies ist notwendig, damit sich die Personen im Showroom kritisch reflektieren können und entsprechend eine transformative Lernerfahrung entsteht. Auch die Einbindung der Nutzenden in Form von Interaktion trägt einen wichtigen Teil zu diesem Erlebnis bei. Im Bereich des Marketing gibt es dafür einen speziellen Begriff.

3.1.3 Experiential Marketing

Experiential Marketing[28] oder auch Engagement Marketing ist eine Marketingstrategie, welche ein Publikum einlädt, mit einem Unternehmen in einer realen Situation zu interagieren. Dadurch möchte man eine Marke greifbar und erlebbar machen und eine dauerhafte Verbindung zwischen dem Kunden und der Marke aufbauen und stärken. Erreicht wird dies durch den Einsatz von partizipativem, praxisorientiertem und greifbarem Material.

Im Showroom möchten wir zwar keine Marke, sondern ein Projekt erlebbar machen. Aber diese Marketingstrategie lässt sich gut auf diesen Anwendungsfall adaptieren.

3.1.4 Fazit

Damit den Besuchenden sowohl die Grundidee und Eckdaten des Projekts ZB mitgeteilt, als auch transformative Lernerfahrungen zu den angesprochenen Themen mitgegeben werden können, bedarf es einer vielschichtigen Gestaltung des Showrooms. Einerseits soll das Projekt verständlich der breiten Masse der Bevölkerung präsentiert werden. Andererseits kann mithilfe von immersiven Medien ein Präsenzerleben für die Nutzenden geschaffen werden, damit diese Informationen in Erinnerung bleiben. Somit können bei

3.1. ANALYSE UND BEGRIFFSERKLÄRUNGEN

den Nutzenden auch besser kritische Reflexionen erzeugt werden.

Kollaboration bzw. Partizipation ist ein wichtiger Themenbereich des Projekts [ZB](#) und soll daher auch im Showroom zum Ausdruck kommen. Die Einbindung der Besuchenden durch partizipative Prozesse dient dabei nicht nur der Vermittlung von Projektinformationen, sondern baut zusätzlich eine dauerhafte Verbindung zwischen der Bevölkerung oder anderen interessierten Personen und dem Projekt [ZB](#) auf.

Das Fazit kann folgendermassen zusammengefasst werden.

- Vielschichtgen Gestaltung des Showrooms.
- Mit der Hilfe von immersiven Medien eine spannende und fesselnde Erfahrung schaffen.
- Die Ausstellung soll starke Partizipationsmöglichkeiten bieten um eine bessere Vermittlung von Informationen sowie eine dauerhafte Verbindung zwischen interessierten Personen und dem Projekt [ZB](#) aufzubauen.

3.2 Vorgehen

Um mögliche Realisierungsansätze zu erarbeiten, wurde ein Brainstorming mit dem Projektteam und dem Industriepartner durchgeführt. Dabei wurden die Erkenntnisse aus [Kapitel 3.1](#) eingebracht. Entsprechend haben sich zwei Fragestellungen herauskristallisiert, auf welche wir uns im Brainstorming Prozess fokussiert haben.

- **Aktion:** Was möchten Besuchende des Showrooms tun?
- **Partizipation:** Wie können sich die Besuchenden in dem Projekt einbringen?

3.2.1 Brainstorming

Die Ergebnisse des Brainstormings wurden zur besseren Übersichtlichkeit in einem Mind-map zusammengefasst. Dabei ist auf der linken Seite der Zusammenstellung das Konzept des Projekts **ZB** zusammengefasst. Gegenüber sind die Inputs und Ideen aus dem Brainstorming nach den beiden Fragestellungen kategorisiert dargestellt.

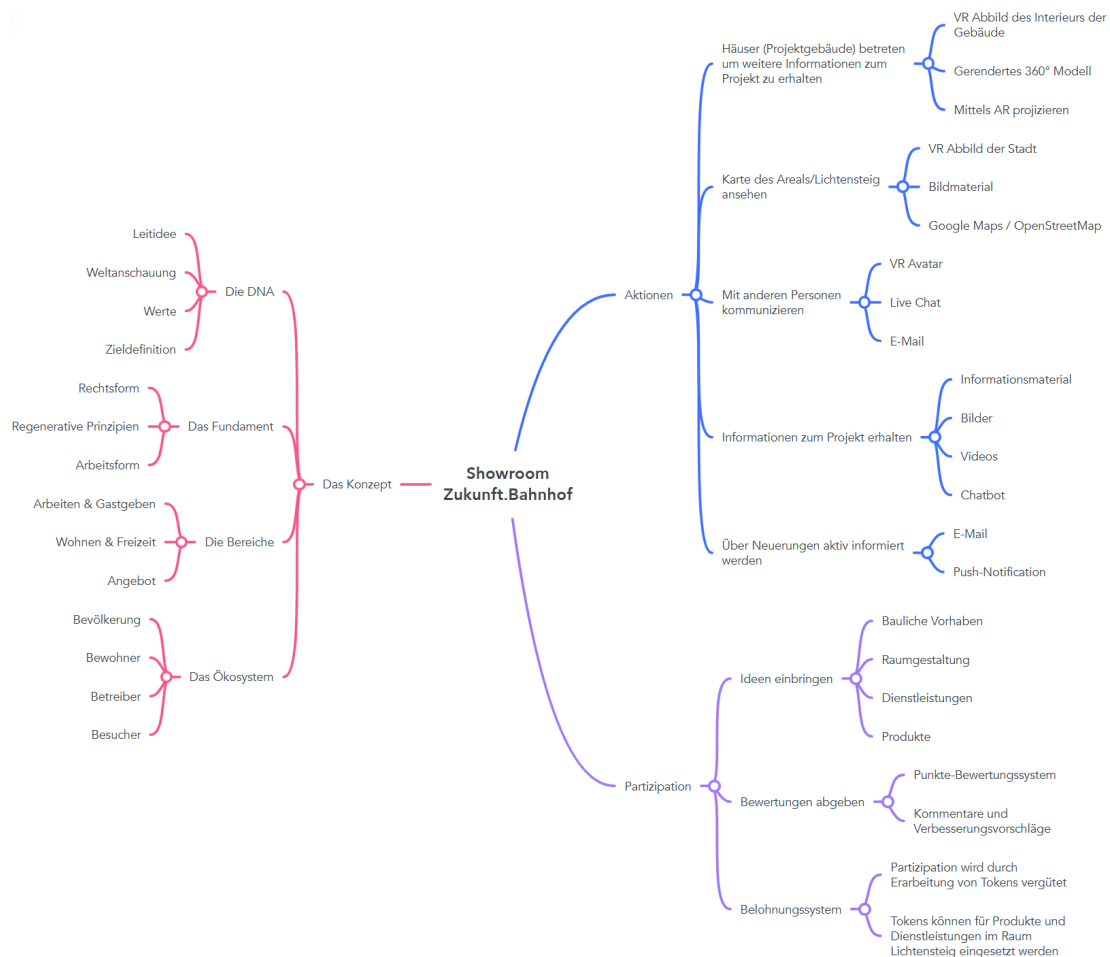


Abbildung 3.1: Resultat des Brainstormings (Quelle: Eigene Darstellung)

Anhand dieser Ideensammlung konnten wir einige konkrete Realisierungsansätze ableiten. Da im Rahmen dieser Arbeit lediglich eine prototypische Umsetzung realisiert wird, haben wir bewusst im Vorfeld nicht auf Realisierungsansätze, welche den zeitlichen Rahmen der Arbeit sprengen würden verzichtet. Somit enthält die nachfolgende Auflistung durchaus Ansätze, welche in der Implementierung sehr zeitaufwändig sind und entsprechend nur zu einem kleinen Teil umgesetzt werden könnten.

3.3 Realisierungsansätze

Die Realisierungsansätze sind nachfolgend aufgeführt und zur besseren Nachverfolgung mit dem Präfix 'RA' gekennzeichnet.

- RA1: VR-Abbild von Lichtensteig
- RA2: App für Partizipation
- RA3: App als Informationsquelle
- RA4: Game zukunft.bahnhof
- RA5: Erlebniswelt zukunft.bahnhof

3.3.1 RA1: VR-Abbild von Lichtensteig

Das Projekt [ZB](#) hat sich bereits mit der Idee befasst, das Projekt Areal oder sogar weitere Teile Lichtensteigs in einem Digital Twin darzustellen. Dabei lag der Fokus jedoch auf der Darstellung der vorhandenen Strukturen, sowie der Erweiterung des Digital Twin um geplante Teile des Projekts, welche noch nicht umgesetzt sind. So könnte im VR-Abbild beispielsweise bereits ein Gebäude abgebildet sein, welches durch das Projekt [ZB](#) erst in einigen Monaten oder Jahren gebaut wird.



Abbildung 3.2: Beispiel einer VR Stadt (Quelle: <https://www.jenamdvacet.cz/>)

Wie die Abbildung 3.2 zeigt, können Stadtteile sehr realitätsnah dargestellt werden. Dieses Beispiel zeigt den Campus der Tomas Bata Universität in Zlín.

3D City Map

Der Kern von Lichtensteig soll in einem 3D-Modell mittels VR Technologien dargestellt werden. Dabei werden sowohl alle Gebäude, Strassen, Öffentlichen Plätze wie auch Grünflächen originalgetreu dargestellt. Zusätzlich werden alle Objekte, welche mit dem Projekt **ZB** in Verbindung gebracht werden, detaillierter ausgearbeitet. Diese Objekte sollen dann auch für weitere Informationen oder Aktionen angewählt werden können. Die gesamte Karte soll aus verschiedenen Perspektiven betrachtet werden. Zudem soll ein näheres Betrachten der Karte mittels Zoom möglich sein.

Interaktion

Da das Projekt **ZB** auf Kollaboration und Partizipation setzt, soll dies auch im Showroom speziell zum Ausdruck kommen.

Besuchende sollen auf der vorhandenen 3D City Map direkt zusätzliche Wünsche und Vorschläge einbringen können. Somit können gute neue Ideen gewonnen werden und das Nutzererlebnis der Besuchenden kann ebenfalls auf eine neue Ebene gehoben werden.

Objekte erfassen

Die Besuchenden sollen während der Navigation auf der 3D City Map neue Objekte wie beispielsweise eine Sitzbank erfassen können. Diese Objekte können dann auf der Karte platziert werden. Jedes erfasste Objekte muss beschrieben werden, sodass schlussendlich nachvollziehbar ist, wer ein neues Objekt platziert hat und mit welcher Intention dies ausgeführt wurde.

Die Liste auswählbarer Objekte könnte je nach Möglichkeiten und Wünschen der Stakeholder und des Projektteams angepasst werden. Mögliche Objekte wären beispielsweise:

- Gebäude
- Grünfläche
- Brunnen
- Spielplatz
- Sitzbank

Ideen einreichen

Auch Ideen für zusätzliche Produkte, Dienstleistungen oder Gesellschaftsangebote sollen auf der Karte erfasst werden können. Analog dem Erfassen von Objekten ist auch hierbei die Angabe von Zusatzinformationen Pflicht.

Bewertungen abgeben

Da sich das Projekt **ZB** über mehrere Jahre erstreckt, ist es für das Projektteam sicherlich essentiell zu erfahren, wie gut die bisher umgesetzten Ideen bei der Bevölkerung von Lichtensteig oder den restlichen Stakeholdern ankommt.

Um das Einholen von Feedback zu vereinfachen, sollen die Besuchenden des Showrooms direkt auf der 3D City Map Objekte und Ideen bewerten können. Dies soll sowohl für bereits umgesetzte Projektteile, wie auch für neue Ideen vom Projektteam oder anderen Showroom Besuchenden möglich sein.

Für die Bewertung wäre ein Sterne- oder Punktesystem denkbar. Zudem sollte es möglich

3.3. REALISIERUNGSANSÄTZE

sein, Kommentare dazu abzugeben. Neue Ideen sollten mit einem Like" versehen werden können, damit gut ersichtlich ist, welche Ideen grossen Anklang finden.

3.3.2 RA2: App für Partizipation

Wie bereits im RA1 beschrieben, sollen Besuchende sich aktiv in das Projekt ZB einbringen können. Dies soll jedoch nicht über eine 3D City Map geschehen, sondern mittels App möglich sein. Am besten würde sich hierzu eine Progressive Web App (PWA) anbieten, welche eine sinnvolle Nutzung sowohl über den Browser, als auch das Smartphone ermöglicht. Die App kann dazu ohne Installation auf dem Smartphone verwendet werden, da sie ähnlich wie eine WebApp über den Browser verwendet wird.

Analog RA1 sollen auch hier Informationen zum Objekt und der erfassenden Person abgespeichert werden. Auch eine Bewertung der Erfassten Objekte und Ideen soll realisiert werden.

AR Integration

Die PWA soll den Benutzenden ermöglichen, vordefinierte Objekte mittels AR in das Stadtbild von Lichtensteig einzufügen. Dabei wird die Kamera des Smartphones eingeschaltet, um die Umgebung anzuzeigen. Danach können die 3D-Modelle der Objekte in das Bild eingefügt werden. So ist erkennbar, wie das gewählte Objekt an diesem Standort hineinpassen würde oder wie es nach der Umsetzung der Idee aussehen könnte.

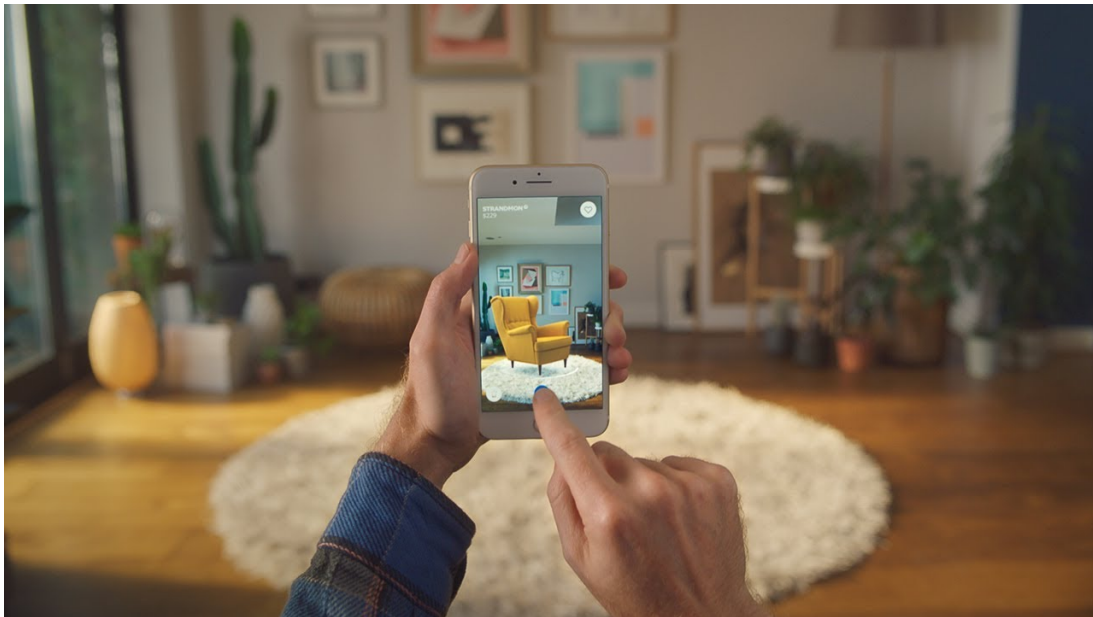


Abbildung 3.3: Ikea AR App

(Quelle: <https://www.ikea.com/ch/de/customer-service/mobile-apps/>)

Das Möbelhaus Ikea beispielsweise verwendet genau dieses Verfahren bereits in einer App, welche es Benutzenden erlaubt anzuschauen, wie Produkte sich in ihrem Wohnraum einfinden. Auch im Outdoor-Bereich wurde bereits mit AR gearbeitet. Ein prominentes Beispiel hierfür ist sicherlich das Spiel Pokémon Go, welches vor einigen Jahren veröffentlicht wurde.

Sobald man mit der Platzierung des Objekts zufrieden ist, soll man ein Foto der Objektplatzierung erstellen können, welches danach für andere Benutzende zugänglich ist.

GPS Koordinaten

Anhand von GPS Koordinaten soll ermittelt werden, wo etwas im Raum Lichtensteig ein Objekt erfasst wurde. Dies erlaubt es schliesslich, die erfassten Objekte auf einer Karte darzustellen.

Dies kann sowohl eine herkömmliche 2D-Karte oder auch eine 3D City Map wie in RA1 beschrieben sein. Die Karte der erfassten Elemente könnte dabei zur besseren Veranschaulichung übersichtlich im Showroom präsentiert werden.

3.3.3 RA3: App als Informationsquelle

In einer PWA soll allen Interessierten die Möglichkeit geboten werden, sich übersichtlich Informationen zum Projekt [ZB](#) zu beschaffen. Die App könnte dabei die bereits vorhandene Website [zukunftbahnhof.ch](#) ablösen, welche nicht wirklich mit einem Responsive Design entwickelt wurde und entsprechend bei einem Aufruf über das Smartphone nicht sehr benutzerfreundlich dargestellt wird.

Kollaboration und Austausch

Zusätzlich soll die App einen durch Login geschützten Bereich erhalten, welcher der Kollaboration und dem Austausch der verschiedenen Stakeholder dient.

Sowohl Einträge mit Informationen zum Projekt, Umfragen, wie auch ein FAQ-Bereich könnte zu einem optimalen Informationsfluss im Projekt beitragen.

Veranstaltungskalender

Um über Veranstaltungen im Raum Lichtensteig informiert zu werden, soll ein Veranstaltungskalender öffentlich zugänglich gemacht werden. Die App kann die Benutzenden mittels Push-Benachrichtigungen über anstehende Events informieren.

Somit ist dafür gesorgt, dass die Angebote und Veranstaltungen im neuen Bahnhof Areal auch umworben und entsprechend gut genutzt werden.

Auch eine Integration der Anlässe, welche auf der Homepage oder dem Mitteilungsblatt der Gemeinde ersichtlich sind, wäre denkbar. Somit könnte die Bevölkerung von Lichtensteig auch über Termine wie zum Beispiel die Sammlung von Abfall oder Karton rechtzeitig erinnert werden.

3.3.4 RA4: Game [zukunft.bahnhof](#)

Das Konzept des Projekts [ZB](#) welches aus DNA, Fundament, Bereichen und dem Ökosystem besteht soll in Form eines Spiels vermittelt werden. Auf diese Art könnte sowohl Kindern wie auch Erwachsenen das Projekt auf unterhaltsame Weise näher gebracht werden.

Diese Art Spiel, auch Serious Game genannt, wurde bereits erfolgreich zur Vermittlung von Lerninhalten eingesetzt. In "Welten der Werkstoffe" wird zum Beispiel Wissen zum Thema Werkstoffkunde vermittelt.



Abbildung 3.4: Screenshot des Spiels "Welten der Werkstoffe"
(Quelle: <https://welt-der-werkstoffe.de/game.html>)

Story

Um die Inhalte spielerisch zu vermitteln, benötigt das Spiel interessante Charaktere, eine angemessene Story und packende Dialoge.

Die Story des Spiels soll aus der Sicht einer Neuzuzügerin oder eines Neuzuzügers erzählt werden. Die Person zieht neu nach Lichtensteig und wird von den Anwohnern über das Projekt **ZB** informiert. Nach und nach bringt sich die Person auch selbst im Projekt ein und erhält somit immer tiefere Einblicke in die Grundidee und das Konzept vom **ZB**.

3.3.5 RA5: Erlebniswelt zukunft.bahnhof

Das Konzept des Projekts **ZB** lässt sich in vier Bereiche aufteilen. Diese 4 Bereiche (DNA, Fundament, Bereiche, Ökosystem) sollen im Showroom anhand einer Erlebniswelt anschaulich vermittelt werden.

Interaktive Infowand

An einer interaktiven Infowand sollen die Besuchenden sich selbständig durch die Informationen navigieren können. Dabei ist der Inhalt in Form einer drehbaren Kugel abgebildet.

Die äusserste Schicht dieser Kugel repräsentiert das Ökosystem mit Bevölkerung, Bewohner, Betreiber und Besucher. Anhand von realen oder auch fiktiven Personen werden Beispiele für die jeweiligen Gruppierungen dargestellt. Texte, Bilder oder auch Kurzvideos sollen einen tieferen Einblick gewähren.

Die Besuchenden sollen danach frei zwischen den verschiedenen Schichten hin und her navigieren können. Somit werden auch die Bereiche, das Fundament und die DNA erlebbar gemacht.

Chatbot

Ein Chatbot soll den Besuchenden Antworten auf viele Fragen zum Projekt [ZB](#) liefern. Die Fragen können dabei entweder in Textform oder auch mittels Spracherkennung ermittelt werden. Ein Katalog von Fragen und deren Antworten ermöglicht es den Betreibenden zu steuern, wie über das Projekt informiert wird.

Angebote

Angebote, welche bereits durch das Projekt entstanden sind, könnten den Besuchenden im Showroom präsentiert werden. Dadurch wird es möglich dass Personen, welche zum ersten Mal mit dem Projekt in Kontakt treten, direkt davon profitieren und entsprechend einfach Zugang dazu finden.

4. Auswahl eines Realisierungsansatzes

In diesem Kapitel wird der Prozess der Auswahl eines Realisierungsansatzes beschrieben. Die Auswahl wird anhand einer Nutzwertanalyse getroffen. Dazu werden die verschiedenen Ansätze gemäss verschiedener Kriterien miteinander verglichen. Aufgrund der Ergebnisse aus dieser Bewertung wird entschieden, welcher Ansatz in Form eines Prototypen umgesetzt wird.

4.1 Kriterien

Der Kriterienkatalog beinhaltet verschiedene, selbst definierte Kriterien mit jeweils einer Gewichtung. Diese entscheidet, wie wichtig das Kriterium ist und entsprechend auch wie stark die Bewertung dieses Kriteriums in die Gesamtpunktzahl einfließt. Es werden alle Realisierungsansätze relativ zueinander bewertet und anhand der Gewichtungen eine Totalpunktzahl berechnet. Anhand dieser Punktzahl wird entschieden, welcher Ansatz in Form eines Prototypen umgesetzt wird.

Die Bewertung der Kriterien erfolgt anhand der nachfolgenden Stufen.

0 Punkte:	Kriterium nicht erfüllt
1 Punkte:	Kriterium teils erfüllt
2 Punkte:	Kriterium erfüllt
3 Punkte:	Kriterium sehr gut erfüllt

Tabelle 4.1: Bewertung der Kriterien

K1: Wissenstransfer

Die Besuchenden des Showrooms sollen bestmöglich über das Projekt [ZB](#) informiert werden. Die Informationen sollen dabei in interessanter und verständlicher Art vermittelt werden.

- Werden alle Informationen zum Projekt [ZB](#) vermittelt?
- Werden die Informationen verständlich und nachvollziehbar vermittelt?
- Sind die Informationen gut zugänglich?

K2: Transformative Lernerfahrungen

Die Besuchenden des Showrooms sollen transformative Lernerfahrungen erleben, welche zu Aha-Momenten führen.

- Werden die Besuchenden zur kritischen Reflexion animiert?

K3: Partizipation

Die Besuchenden des Showrooms sollen sich durch partizipative Prozesse in das Projekt [ZB](#) einbringen können. Zudem soll der Aspekt der Kollaboration, welcher ein wichtiger Bestandteil des Konzepts ist, erlebbar gemacht werden.

4.1. KRITERIEN

- Können sich die Besuchenden partizipativ im Projekt einbringen?
- Kann durch die Partizipation ein sinnvoller Beitrag zum Projekt ZB beigetragen werden?
- Wird der Aspekt der Kollaboration für die Besuchenden erlebbar?

K4: Präsenzerlebnis

Die Besuchenden des Showrooms sollten ein starkes Präsenzerlebnis erfahren. Durch die Wahrnehmung der Inhalte auf verschiedenen Ebenen und mit verschiedenen Sinnen soll das Präsenzerlebnis zusätzlich gesteigert werden.

- Werden die Besuchenden auf moderne und interessante Art Informationen erhalten?
- Sind die Inhalte für die breite Masse der Bevölkerung zugänglich?
- Wird durch Präsenzerleben ein starkes Präsenzerlebnis generiert?

K5: Gestaltungsmöglichkeiten

Der Showroom soll interessant und ideenreich gestaltet werden. Der verfügbare Platz soll gut genutzt werden. Der Realisierungsansatz soll im Verlaufe des Projekts ZB erweitert oder angepasst werden können.

- Wird der Showroom gut in Szene gesetzt und ideenreich gestaltet?
- Ist der Showroom erweiter- oder anpassbar?

K6: Umsetzbarkeit

Der Realisierungsansatz soll gut anhand eines Prototypen, unter Berücksichtigung des zeitlichen Rahmens dieser Arbeit, implementiert werden können. Dadurch soll möglichst gut geprüft werden können, inwiefern sich der Ansatz zu einem kompletten Produkt weiterentwickeln lässt.

- Kann der Realisierungsansatz anhand eines Prototypen, unter Berücksichtigung des zeitlichen Rahmens dieser Arbeit, sinnvoll implementiert werden?
- Kann ein Prototyp erarbeitet werden, welcher danach zu einem Endprodukt weiterverarbeitet werden kann?

4.2 Gewichtung

Die Gewichtung der Kriterien wird auf die Aufgabenstellung und die Ziele abgestimmt. Ein zusätzlicher Workshop mit einer Fokusgruppe aus unbeteiligten Personen sorgt dafür, dass keine voreingenommene Gewichtung gemacht wird. Aus diesem Grund wird die Gewichtung beider Parteien gleich stark in die finale Gewichtungstabelle einfließen. Dazu wird den Teilnehmenden eine Liste aller Kriterien ausgehändigt. Zudem erhalten sie die Aufgabenstellung und werden bezüglich der Ziele aufgeklärt. Anschliessend werden sie innerhalb von drei Minuten jedem Kriterium eine Gewichtung in Form eines Zahlenwerts (1-3) zuteilen. Schliesslich wird der Mittelwert aus diesen Werten und der Beurteilung durch uns errechnet, was die definitive Gewichtung des Kriteriums ergibt.

Nr.	Kriterium	Projektteam	Fokusgruppe	Gewichtung
K1	Wissenstransfer	1	2	1.5
K2	Transformative Lernerfahrungen	3	1	2
K3	Partizipation	2	2	2
K4	Präsenzerlebnis	3	3	3
K5	Gestaltungsmöglichkeiten	1	1	1
K6	Umsetzbarkeit	1	2	1.5

Tabelle 4.2: Gewichtung der Kriterien

4.3 Bewertung

Nachfolgend werden die Kriterien der Realisierungsansätze gemäss dem Schema von [Tabelle 4.1](#) bewertet. Multipliziert mit der Gewichtung ergibt dies die jeweilige Punktzahl pro Kriterium.

4.3.1 K1: Wissenstransfer

RA1: VR-Abbild von Lichtensteig

RA1 beschreibt, dass die Besuchenden des Showrooms Informationen zum Projekt an diversen Stellen in der City Map einsehen können. Dabei sind die Informationen direkt örtlich an ein Objekt, wie beispielsweise ein Gebäude, gebunden. Die Informationen können somit auf verständliche Weise kommuniziert werden. Es sind keine Grenzen gesetzt bezüglich Anzahl eingebundener Informationen. Auch ein separater Bereich auf der City Map, welcher ausschliesslich Informationen in Form von Fliesstext oder Videos beinhaltet, ist denkbar.

RA2: App für Partizipation

RA2 sieht nicht vor Informationen zum Projekt den Nutzenden zur Verfügung zu stellen, diese Informationen sollen von einer anderen Stelle geliefert werden (z.B. Webseite der Stiftung). Stattdessen fokussiert sich dieser Ansatz mit dem Thema Partizipation.

RA3: App als Informationsquelle

RA3 wird in Form einer PWA beschrieben, welche ebenfalls alle Informationen des Projekts abbilden kann. Dies bietet den Vorteil, dass die Informationen für jede Person

4.3. BEWERTUNG

jederzeit zugänglich sind und man dazu nicht zwingendermassen den Showroom besuchen muss.

RA4: Game zukunft.bahnhof

Gemäss RA4 werden die Lerninhalte in Form eines Games präsentiert. Dabei können die Informationen spielerisch und eindrücklich vermittelt werden, was ein grosser Vorteil ist. Gerade Nutzende im Kindesalter sprechen darauf normalerweise besser an als auf herkömmliche Informationsvermittlung. Ein Nachteil ist jedoch, dass die Informationen nicht frei zugänglich sind, sondern erst nach und nach im Verlauf des Spiels mitgeteilt werden. Das Game wird in Form einer PWA implementiert und ist somit sehr gut zugänglich. [19]

RA5: Erlebniswelt zukunft.bahnhof

Eine Erlebniswelt wie in RA5 beschrieben, bietet eine gute Grundlage, Informationen in angemessener Art zu präsentieren. Die Informationen sind zwar nur im Showroom zugänglich, können jedoch auf verschiedene Arten erlebbar und greifbar gemacht werden.

Realisierungsansatz	Punktzahl
RA1: VR-Abbild von Lichtensteig	3
RA2: App für Partizipation	0
RA3: App als Informationsquelle	2
RA4: Game zukunft.bahnhof	2
RA5: Erlebniswelt zukunft.bahnhof	2

Tabelle 4.3: Bewertung K1 Wissenstransfer

4.3.2 K2: Transformative Lernerfahrungen

RA1: VR-Abbild von Lichtensteig

RA1 kann durch den Einsatz von immersiven Medien besser zur kritischen Reflexion animieren als Lösungen, die auf herkömmliche Medien setzen. Die virtuelle City Map ermöglicht es, Konzepte und Ideen des Projekt ZB anschaulich aufzuzeigen.

RA2: App für Partizipation

Der Ansatz RA2 ist nicht wirklich auf die Vermittlung einer transformativen Lernerfahrung ausgerichtet. Da die Partizipation mittels PWA funktioniert, ist es schwierig, die Nutzenden vollständig abzuholen und vertieft mit dem Thema zu konfrontieren.

RA3: App als Informationsquelle

RA3 vermittelt die Informationen zum Projekt klassisch in Form einer responsive Website. Dabei können zwar Fragen, welche die kritische Reflexion anregen gestellt werden, jedoch hält sich dies stark in Grenzen. Gruppenerlebnisse hingegen sind sehr unwahrscheinlich mit diesem Ansatz.

RA4: Game zukunft.bahnhof

Durch Storytelling in RA4, packende Charaktere und Dialoge kann eine gute Grundlage für das Erleben von transformativen Lernerfahrungen geschaffen werden. Die Nutzenden können sich mit den Charakteren identifizieren und daher deren Fragestellungen und Probleme nachvollziehen.

RA5: Erlebniswelt zukunft.bahnhof

RA5 liefert den Besuchenden Informationen anhand einer interaktiven Inforwand im Showroom. Ähnlich wie auch RA1 kann mithilfe von VR Technologien ein Immersives Erlebnis geschaffen werden. Dadurch kann eine Reflexion über die Themen des Projekt [ZB](#) durchaus stärker gefördert werden, als bei der herkömmlichen Informationsvermittlung.

Realisierungsansatz	Punktzahl
RA1: VR-Abbild von Lichtensteig	2
RA2: App für Partizipation	1
RA3: App als Informationsquelle	1
RA4: Game zukunft.bahnhof	3
RA5: Erlebniswelt zukunft.bahnhof	2

Tabelle 4.5: Bewertung K2 Transformative Lernerfahrungen

4.3.3 K3: Partizipation

RA1: VR-Abbild von Lichtensteig

Besuchende des RA1 erfahren den Aspekt der Kollaboration in Bezug auf das Projekt [ZB](#) und können sich zudem auch direkt selber anhand von partizipativen Prozessen aktiv in das Projekt einbringen.

RA2: App für Partizipation

RA2 legt den Fokus komplett auf Partizipation. Durch die sehr gute Verfügbarkeit und Zugänglichkeit der PWA ist Partizipation im Projekt jederzeit und von überall aus möglich. Über die App können problemlos, weitere Formen der Zusammenarbeit mit der Bevölkerung zum Projekt [ZB](#) aufgebaut werden.

RA3: App als Informationsquelle

Die Infoapp von RA3 dient hauptsächlich dem Informationsfluss. Wie bereits beschrieben, kann hingegeben auch hierbei ein durch Login geschützter Bereich für die Kollaboration der Stakeholder untereinander genutzt werden.

RA4: Game zukunft.bahnhof

Bei RA4 wird zwar die Kollaboration in Bezug auf das Projekt [ZB](#) anschaulich aufgezeigt. Jedoch ist es nur schwer umsetzbar, Nutzende aktiv in partizipativen Prozessen im Projekt einzubringen.

RA5: Erlebniswelt zukunft.bahnhof

Auch RA5 konzentriert sich bezüglich Partizipation auf die Informationsvermittlung und bindet die Besuchenden des Showrooms nicht aktiv in die Gestaltung des Projekts mit ein.

Realisierungsansatz	Punktzahl
RA1: VR-Abbild von Lichtensteig	2
RA2: App für Partizipation	3
RA3: App als Informationsquelle	2
RA4: Game zukunft.bahnhof	1
RA5: Erlebniswelt zukunft.bahnhof	0

Tabelle 4.6: Bewertung K3 Partizipation

4.3.4 K4: Präsenzerlebnis

RA1: VR-Abbild von Lichtensteig

VR Technologien, wie sie bei RA1 eingesetzt werden, ermöglichen es den Besuchenden des Showrooms durch Transportation ein starkes Präsenzerlebnis zu bieten. Durch den lokalen Betrieb in einem fixen Raum stehen zudem die Möglichkeiten für den Einsatz von VR-Brillen offen. Das könnte das Präsenzerlebnis weiter verbessern. Auch das Verwenden von grossen Bildschirmen oder Beamer kann sich positiv auf das Präsenzerlebnis auswirken.

RA2: App für Partizipation

RA2 verwendet AR um zukünftige Veränderungen in Lichtensteig durch das Projekts [ZB](#) zu visualisieren. Durch den Fakt, dass die Informationen auf einer PWA konsumiert werden, ist es schwieriger ein gleich gutes Präsenzerlebnis zu bieten, wie dies in einem dafür konzipierten Showroom möglich ist. Im Gegensatz dazu sind die Inhalte jedoch besser zugänglich.

RA3: App als Informationsquelle

RA3 liefert die Informationen zum Projekt [ZB](#) in klassischer Weise. Dadurch wird nicht ein speziell starkes Präsenzerlebnis generiert.

RA4: Game zukunft.bahnhof

Das Game gemäss RA4, präsentiert die Informationen auf sehr moderne und frische Art. Vor allem junge Besuchende werden dadurch erfolgreich abgeholt. Eine gut durchdachte und packende Story kann durchaus eine Transportation verursachen und die Nutzenden fesseln. Die Umsetzung in Form eines Browser Games macht die Inhalte gut zugänglich.

RA5: Erlebniswelt zukunft.bahnhof

In der Erlebniswelt von RA5 können die Inforationen auf verschiedene Arten präsentiert werden. Durch den Einsatz eines Chatbots, via Sprachsteuerung, können Inhalte ebenfalls interessant vermittelt werden.

Realisierungsansatz	Punktzahl
RA1: VR-Abbild von Lichtensteig	3
RA2: App für Partizipation	2
RA3: App als Informationsquelle	1
RA4: Game zukunft.bahnhof	3
RA5: Erlebniswelt zukunft.bahnhof	2

Tabelle 4.7: Bewertung K4 Präsenzerlebnis

4.3.5 K5: Gestaltungsmöglichkeiten

RA1: VR-Abbild von Lichtensteig

RA1 bietet den Besuchenden eine City Map per VR an. Dies kann gut in einen Showroom anhand eines grossen Touchscreens integriert werden. Auch der Einsatz von VR Brillen im Showroom sind denkbar. Dabei ist mehr Präsenzerleben möglich, jedoch wird der Showroom auf den ersten Blick nicht wirklich stark in Szene gesetzt. Je nach Platzanspruch können zusätzliche gestalterische Elemente umgesetzt werden.

RA2: App für Partizipation

Wie in RA2 beschrieben, wird der Showroom selbst gar nicht genutzt. Bei Bedarf könnte die PWA auf einem Screen im Showroom gezeigt werden. Dies würde den Showroom jedoch nicht speziell in Szene setzen.

RA3: App als Informationsquelle

Die Infoapp von RA3 thematisiert den Showroom und dessen Gestaltung nicht, sondern fokussiert sich auf das Bereitstellen von Informationen zum Projekt [ZB](#). Die Infoapp liesse sich gut im Showroom auf einem Touchscreen darstellen. Dies würde den Raum nicht speziell gut gestalten, aber die Besuchenden schnell mit Informationen versorgen.

RA4: Game zukunft.bahnhof

Auch RA4 beschreibt nicht wie der Showroom gestaltet werden soll. Grundsätzlich kann das Browser Game ebenfalls ohne viel Aufwand im Showroom bereitgestellt werden. Eine ideenreiche Gestaltung ist jedoch nicht sehr einfach umzusetzen.

RA5: Erlebniswelt zukunft.bahnhof

In RA5 werden diverse Möglichkeiten vorgestellt, wie der Showroom gestaltet werden kann. Durch das Verwenden von mehreren Elementen kann der Raum abwechslungsreich und interessant gestaltet werden. Auch das Anpassen oder Erweitern des Showrooms kann durch die verschiedenen Elemente gut bewerkstelligt werden.

Realisierungsansatz	Punktzahl
RA1: VR-Abbild von Lichtensteig	2
RA2: App für Partizipation	1
RA3: App als Informationsquelle	1
RA4: Game zukunft.bahnhof	1
RA5: Erlebnisswelt zukunft.bahnhof	3

Tabelle 4.8: Bewertung K5 Gestaltungsmöglichkeiten

4.3.6 K6: Umsetzbarkeit

RA1: VR-Abbild von Lichtensteig

RA1 lässt sich gut in einem Prototypen umsetzen. Es kann in Erfahrung gebracht werden, ob Objekte wie Gebäude, Strassen oder auch Grünflächen ausreichend detailliert in einer VR City Map dargestellt werden können. Auch das interagieren mit Objekten auf der Karte lässt sich gut erproben. Werden die verwendeten Technologien sorgfältig ausgelesen, lässt sich der Prototyp mit hoher Wahrscheinlichkeit zu einem Fertigprodukt weiterentwickeln.

RA2: App für Partizipation

Auch RA2 lässt sich sinnvoll als Prototyp darstellen. Hierbei kann getestet werden, wie gut vordefinierte Objekte mittels [AR](#) in die Umgebung eingefügt werden können. Die Genauigkeit der GPS Position der erfassenden Person kann ebenfalls analysiert werden.

RA3: App als Informationsquelle

Die PWA, wie in RA3 beschrieben, kann sehr gut in einem Prototypen umgesetzt werden. Da keine [MR](#) Technologien benötigt werden, ist eine direkte Überführung in ein Endprodukt gut möglich.

RA4: Game zukunft.bahnhof

RA4 lässt sich schwer in Form eines Prototypen implementieren. Es lässt sich zwar analysieren, ob grafische Elemente wie die Charaktere oder die Umgebung des Spiels entworfen und in einem Game abgebildet werden können. Auch die interaktiven Elemente, wie die Konversation mit den Charakteren oder das Entscheiden, was die Hauptfigur als nächstes tun soll, kann in einem Prototypen testhalber implementiert werden. Doch der Aufwand für einen solchen Prototypen wird den Rahmen dieser Arbeit sprengen. Zudem wird man wahrscheinlich nur wenige Informationen über die Umsetzbarkeit eines vollständigen Produkts erhalten.

RA5: Erlebnisswelt zukunft.bahnhof

Für die Erlebnisswelt RA5 lässt sich überprüfen, inwiefern ein interaktives Infoboard umgesetzt werden kann. Hierbei kann vor allem auf die Steuerung und Navigation eingegangen werden. Beim Chatbot kann auf vorhandene Produkte zurückgegriffen werden, was einen Prototypen schnell umsetzen lässt.

4.4. AUSWERTUNG DER ERGEBNISSE

Realisierungsansatz	Punktzahl
RA1: VR-Abbild von Lichtensteig	2
RA2: App für Partizipation	2
RA3: App als Informationsquelle	3
RA4: Game zukunft.bahnhof	0
RA5: Erlebniswelt zukunft.bahnhof	2

Tabelle 4.9: Bewertung K6 Umsetzbarkeit

4.4 Auswertung der Ergebnisse

Zur Berechnung der Gesamtpunktzahl eines Realisierungsansatzes wurden die Punkte der einzelnen Kriterienbewertungen zusammengerechnet.

Realisierungsansatz	Totale Punktzahl
RA1: VR-Abbild von Lichtensteig	26.5
RA2: App für Partizipation	18
RA3: App als Informationsquelle	17.5
RA4: Game zukunft.bahnhof	21
RA5: Erlebniswelt zukunft.bahnhof	19

Tabelle 4.10: Auswertung Nutzwertanalyse Realisierungsansätze

Der [Realisierungsansatz 1 \(VR-Abbild von Lichtensteig\)](#) erreicht bei der Nutzwertanalyse die höchste Punktzahl.

5. Konzept Showroom

In diesem Kapitel wird der Entwurf eines Konzepts für ein Showroom des Projekts [ZB](#) beschrieben.

5.1 Workshop mit Fokusgruppe

Die bisher erarbeiteten Realisierungsansätze wurden alle durch das Projektteam generiert. Um zusätzlich noch weitere, eventuell ausser Acht gelassene Ideen von unbeteiligten Personen zu erhalten, wurde ein Workshop mit einer Fokusgruppe durchgeführt.

5.1.1 Teilnehmende Personen

Die Fokusgruppe besteht aus fünf Personen im Alter von 20 bis 30 Jahren mit verschiedenen beruflichen und schulischen Hintergründen. Beide Geschlechter sind ausreichend vertreten.

5.1.2 Methodik

Der Workshop wurde im Stil der [Crazy8](#) Methode durchgeführt. Die Teilnehmenden machen sich dazu selbständig innerhalb von fünf Minuten Gedanken wie das Projekt [ZB](#) digital erlebbar gemacht werden kann. In dieser Zeit soll jede Person fünf Ideen erarbeiten. Diese werden nach Ablauf danach im Plenum erläutert und besprochen. Durch eine offene Diskussion können Ideen präzisiert werden oder auch weitere Ideen dazu kommen.

5.1.3 Ergebnisse

Folgende Ideen wurden im Workshop zusammengetragen:

- Film mit Animationen in Dauerschleife laufen lassen
- VR Animation zur Veranschaulichung des [ZB](#) Areal
- 3D-Beamer/Hologramme um Teile des Projekts visuell abzubilden
- Miniatur Modell des Bahnhof Areal
- Bilder von geplanten baulichen Veränderungen zeigen
- PowerPoint mit Informationen bereitstellen
- Infotafel mit Informationen bereitstellen
- VR-Brillen einsetzen, um Animationen oder Modelle anzuschauen
- Durch QR Codes in Lichtensteig auf Informationen aufmerksam machen
- Unterstützungs- und Gönnerbeiträge digital entgegennehmen (via Smart Contracts auf Blockchain)

5.1.4 Weitere Inputs

Nebst diesen Ideen wurden auch weitere Punkte genannt, welche generell bei der Gestaltung des Showrooms und der Ausarbeitung des Konzepts beachtet werden sollten.

- Interessen der Bevölkerung sammeln
- Showroom familienfreundlich gestalten
- Zu Besuchen im Showroom animieren durch Wettbewerbe, Angebot von Essen/Trinken oder dem bieten eines Rahmenprogramms
- Showroom soll mehrere Elemente enthalten und sich nicht auf einen Lösungsansatz beschränken

5.1.5 Fazit

Der Workshop mit der Fokusgruppe hat gezeigt, dass die erarbeiteten Realisierungsansätze Ideen von unbeteiligten Personen widerspiegeln. Interessant ist, dass fast alle Teilnehmenden der Fokusgruppe auf die Idee der Erstellung einer City Map gekommen sind. Sei das in Form eines Miniatur Modells oder durch VR Technologien oder 3D Modelle. Auch die Nutzwertanalyse zeigt auf, dass dieser Realisierungsansatz am meisten Potenzial aufweist.

Die familiengerechte Gestaltung des Showrooms wurde bei den Realisierungsansätzen durch die Nutzwertanalyse bewertet (Kriterium K4). Die Idee, potenzielle Besuchende mithilfe eines Rahmenprogramms, Wettbewerben oder dem Angebot von Essen und Trinken von einem Besuch des Showrooms zu überzeugen, war neu und ist ein spannender Ansatz, der im Rahmen eines Gesamtkonzepts weiterverfolgt wird. Laut der Fokusgruppe sollen mehrere Elemente im Showroom platziert werden, um ein abwechslungsreiches Erlebnis zu bieten. Zudem hilft dies gerade bei grösseren Gruppen im Showroom trotzdem möglichst alle Personen mit Inhalten zu versorgen.

5.2 Konzeptentwurf

Anhand der Erkenntnisse der Nutzwertanalyse und des Workshops mit der Fokusgruppe steht fest, dass der Showroom mehrere Elemente beinhalten sollte, welche auf unterschiedliche Weise das Projekt [ZB](#) näher bringen. Ein Rahmenprogramm rund um den Showroom wird helfen, dauerhaft für gute Besucherzahlen zu sorgen.

Die Elemente im Showroom sind blau gezeichnet und nummeriert.

1. VR City Map
2. Bilder und Infotexte zu Konzepten und Visionen des Projekts
3. Interaktive Infotafel
4. Cafébar zum Austausch mit Besuchenden und Nutzung als aktiver Treffpunkt in Lichtensteig
5. Infomaterial zu verwandten Projekten

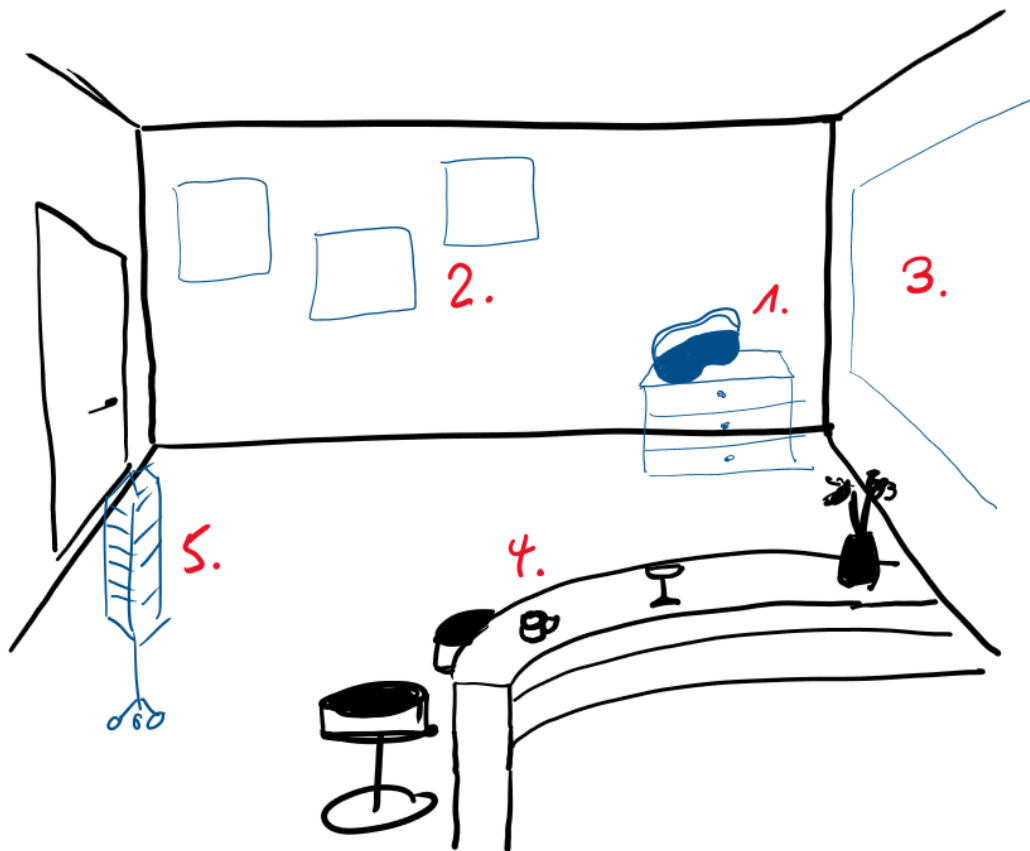


Abbildung 5.1: Beispiel Showrooms Innenansicht (Quelle: Eigene Darstellung)

Die [Abbildung 5.2](#) illustriert, dass der Showroom auch auf mehrere Stockwerke verteilt werden kann. Dabei wird der gesellschaftliche Teil vom Informativen Teil getrennt und das Café als Treffpunkt im unteren Bereich angesiedelt.

Somit kann auf relativ geringer Fläche ein interessanter Showroom entstehen, welcher vielseitig genutzt werden kann. Auch die Möglichkeit einer modularen Erweiterung ist durch die Nutzung von Containern gegeben.



Abbildung 5.2: Beispiel Showroom Aussenansicht
(Quelle: <https://www.kreative-eventcontainer.de/anwendungsbeispiele/presentation-container>)

5.3 Abgrenzung

Im Rahmen dieser Arbeit wird der Fokus auf die prototypische Implementierung der VR City Map zur visuellen Betrachtung des zukünftigen Bahnhof Areal und der Partizipation der Besuchenden gelegt. Jedoch wird der Showroom bei vollständiger Umsetzung, wie in der [Abbildung 5.1](#) beschrieben, im Idealfall aus vielen weiteren Elementen bestehen. Die Abbildung zeigt eine Möglichkeit auf, wie der komplette Showroom gestaltet werden kann.

6. Anforderungsanalyse

In diesem Kapitel wird auf die funktionalen sowie die nicht-funktionalen Anforderungen an die City Map eingegangen.

6.1 Funktionale Anforderungen

Die [Tabelle 6.1](#) zeigt die Muss-Anforderungen bezüglich Funktionalität an den Prototypen auf.

Nr.	Anforderung
FR1	Als benutzende Person möchte ich ein VR-Abbild des Areals ZB aus der Vogelperspektive betrachten können, sodass ich mir einen Überblick über dessen geplante Gestaltung verschaffen kann.
FR2	Als benutzende Person möchte ich die Kameraperspektive 360° schwenken können, sodass ich meinen Blickwinkel auf die Objekte unterschiedlich wählen kann.
FR3	Als benutzende Person möchte ich Objekte auswählen können, um weitere Informationen dazu zu erhalten, sodass ich mich optimal über das Projekt ZB informieren kann.
FR4	Als benutzende Person möchte ich Ideen in Form von Textnachrichten ins Projekt ZB einbringen können, sodass ich mich aktiv am Projekt beteiligen kann.
FR5	Als benutzende Person möchte ich Objekte einer Arealgestaltung bewerten können, sodass ich zeigen kann, welche Elemente mich ansprechen.
FR6	Als benutzende Person möchte ich Objekte in einer Arealgestaltung kommentieren können, sodass ich konstruktives Feedback zu den umgesetzten und geplanten Elementen des Projekts geben kann.

Tabelle 6.1: Funktionale Anforderungen

6.1.1 Optionale Anforderungen

Viele Funktionalitäten sind als optionale Anforderungen eingeplant. Die optionalen Anforderungen können als Erweiterungsmöglichkeiten gesehen werden und bieten eine Grundlage um den Prototypen weiter zu entwickeln. Sie werden nach Möglichkeit umgesetzt.

Nr.	Optionale Anforderung
FR7	Als benutzende Person möchte ich zwischen verschiedenen Tageszeiten umschalten können, sodass erkennbar ist welche Elemente des Projekts ZB wann und wie zur Nutzung angedacht sind.
FR8	Als benutzende Person möchte ich in die Szenerie hineinzoomen können, sodass ich die Objekte näher betrachten kann.
FR9	Als benutzende Person möchte ich die Kameraperspektive komplett frei wählen können, sodass ich Objekte aus allen möglichen Perspektiven betrachten kann.
FR10	Als benutzende Person möchte ich ein leeres Template des Areal einblenden können, sodass ich eine eigene Version der Arealgestaltung erstellen kann.
FR11	Als benutzende Person möchte ich vordefinierte Objekte auf dem leeren Template platzieren können, sodass ich das Areal anhand der bestehenden Möglichkeiten frei gestalten kann.
FR12	Als benutzende Person möchte ich meine Arealgestaltung abspeichern können, sodass die Szenerie zu einem späteren Zeitpunkt angeschaut werden kann.
FR13	Als benutzende Person möchte ich im Vorfeld erstellte Arealgestaltungen anschauen können, damit ich sehen kann, wie andere Personen das Areal gestaltet haben.
FR14	Als benutzende Person möchte ich Arealgestaltungen als solches bewerten können, sodass ich zeigen kann welche Entwürfe mir gefallen.
FR15*	Als benutzende Person möchte ich, von Benutzenden erfasstes, Feedback bewerten können, sodass ich aufzeigen kann, welche Elemente mir gefallen.
FR16*	Als benutzende Person möchte ich von Benutzenden erfasstes Feedback kommentieren können, sodass ich konstruktives Feedback zu den eingereichten Ideen geben kann.

Tabelle 6.2: Funktionale Anforderungen (optional)

* Da im Verlaufe der Arbeit entschieden wurde, dass die Feedbacks nur dem Projektteam zur Verfügung gestellt werden soll und nicht öffentlich, sind beide Anforderungen nicht umsetzbar und werden nicht mehr berücksichtigt.

6.2 Nicht-Funktionale Anforderungen

Die Nicht-funktionale Anforderungen werden nach **FURPS+** in verschiedene Kategorien aufgeteilt.

Functionality

Nr	Anforderung	Measure
NFR1	Die Anwendung soll mit aktuellen Webbrowsern verwendet werden können.	Die Anwendung soll in mindestens zwei gängigen Browsern (aktuelle Version von Chrome, Edge oder Firefox) korrekt dargestellt und genutzt werden können. Ausserdem sollen auch mobile Geräte unterstützt werden.
NFR2 (optional)	Die Anwendung soll mit einer VR-Brille verwendet werden können.	Die Anwendung soll mindestens eine Brille der folgenden Marken unterstützen: HTC Corporation, Meta Platforms Inc., Microsoft Inc., Samsung Electronics oder Sony Group Corporation.
NFR3 (optional)	Fehler, die im Backend der Anwendung auftreten, sollen zentral in Logfiles geschrieben werden.	Die Anwendung soll alle im Backend auftretende Fehler global abfangen und auf dem Server in ein Logfile schreiben.

Tabelle 6.3: Nicht-funktionale Anforderungen (Functionality)

Usability

Nr	Anforderung	Measure
NFR4	Die Steuerung soll einfach zu erlernen und zu bedienen sein, um eine reibungslose Erfahrung zu gewährleisten.	Die Steuerung soll per Maus und Tastatur intuitiv möglich sein. Usabilitytests im Browser sollen mindestens ein Ergebnis von 80% erreichen.
NFR5 (optional)	Die Steuerung mit einem VR-Headset soll einfach zu erlernen und zu bedienen sein, um eine reibungslose Erfahrung zu gewährleisten.	Die Steuerung soll per VR-Headset möglich sein. Usabilitytests mit VR-Headset sollen mindestens ein Ergebnis von 80% erreichen.

6.2. NICHT-FUNKTIONALE ANFORDERUNGEN

Nr	Anforderung	Measure
NFR6	In der Anwendung eingebundene Informationen zum Projekt ZB sollen einfach zugänglich sein.	Beim Auswählen eines mit Informationen versehenen Objekts sollen mit einem einfachen 'Klick' weitere Informationen dazu angezeigt werden.

Tabelle 6.4: Nicht-funktionale Anforderungen (Usability)

Reliability

Nr	Anforderung	Measure
NFR7	Die VR-Anwendung soll robust und fehlertolerant sein, um sicherzustellen, dass sie auch bei unerwarteten Ereignissen oder Bedingungen stabil bleibt.	Im Backend soll durch den Einsatz von Testing sichergestellt werden, dass die Businesslogik sinngemäss funktioniert und nicht zu Abstürzen der Anwendung führt. Die Testabdeckung soll dabei mindestens 80% betragen. Für die benutzende Person relevante Fehlermeldungen sollen im Frontend angezeigt werden.

Tabelle 6.5: Nicht-funktionale Anforderungen (Reliability)

Performance

Nr	Anforderung	Measure
NFR8	Die VR-Anwendung soll in der Lage sein, die Stadt in einer angemessenen Geschwindigkeit zu laden.	Die Anwendung soll auf in weniger als 10 Sekunden geladen und funktionsfähig sein.
NFR9	Die VR-Anwendung soll Interaktionen und Bildwiederholungen in einer angemessenen Geschwindigkeit verarbeiten und anzeigen können.	Die Anwendung soll auf aktuelle Geräte mindesten 60 FPS darstellen können.

Tabelle 6.6: Nicht-funktionale Anforderungen (Performance)

Supportability

Nr	Anforderung	Measure
NFR10	Die VR-Anwendung soll einfach wart- und erweiterbar sein.	Die Architektur der VR-Anwendung soll modular und klar strukturiert sein, um Änderungen und Erweiterungen an einzelnen Modulen zu erleichtern. Für die Kommunikation mit dem Backend sollen DTOs verwendet werden, um die Abhängigkeit zu minimieren und einfacher austauschbar zu machen.
NFR11	Die VR-Anwendung soll weitverbreitete und gut unterhaltene Frameworks und Bibliotheken einsetzen.	Es sollen keine, nicht mehr weiterentwickelte Frameworks und Bibliotheken eingesetzt werden, um die Wartbarkeit in der Zukunft sicherzustellen.
NFR12	Die VR-Anwendung soll zum grössten Teil aus Open-Source Technologien bestehen.	Mhr als die Hälfte der eingesetzten Technologien, Frameworks und Bibliotheken sollen Open-Source sein.

Tabelle 6.7: Nicht-funktionale Anforderungen (Supportability)

7. Evaluierung der Technologien

Für MR werden verschiedene Technologien miteinander verglichen. Da der Einsatz freier und quelloffener Technologien als Ziel vorgegeben ist, fallen diverse Technologien aus. So z.B. Verge3D oder Wonderland Engine sowie viele andere Game Engines. Die Game Engines sind zwar oft frei zugänglich, es fallen aber oft noch Lizenzgebühren an. Ausserdem sind oft nicht alle Features inbegriffen und es handelt sich nicht um Open-Source.

7.1 Begriffserklärung

Um den Zusammenhang zwischen verschiedene Technologien und Begriffe besser zu verstehen wird in diesem Abschnitt auf verschiedene Begriffe eingegangen. In diesem Kapitel wird über Virtual Reality sowie verwandte Begriffe (AR/MR) diskutiert.

Virtual Reality

Die Virtual Reality oder kurz VR steht für eine computergenerierte virtuelle Realität. Diese Umgebung wird immersiv, interaktiv, dreidimensional und multisensorisch computergeneriert. Dadurch fühlen sich die Benutzende der VR-Welt als wären sie Teil dieser Welt und Reize der realen Welt werden nicht mehr wahrgenommen. Als Anschauungsmedium dienen oft so genannte VR-Brillen. Damit lässt sich sehr gut in die virtuellen Realität eintauchen, da über das integrierte Display der Brille nur noch die VR-Welt zu sehen ist. Je nach Brille sind auch Kopfhörer Bestandteil der Brille, was die Audioreize der realen Welt ausblenden lässt und so die Audioreize der virtuellen-Welt wahrgenommen werden können. Jedoch lassen sich VR-Applikationen auch auf Smartphone oder Computer darstellen. Bei solchen Applikationen lässt sich die wirkliche Welt nicht wirklich gut ausblenden. Um eine VR Applikation zu erstellen stehen viele Technologien zur Verfügung. Im Kapitel(Technologiekandidaten) sind mehrere solcher Technologien aufgelistet. [8]

Augmented Reality

Augmented Reality zu Deutsch erweiterte Realität, verbindet die virtuelle Welt mit der realen Welt. Um dies zu bewerkstelligen, gibt es diverse Möglichkeiten. Eine ist die Anwendung einer AR/MR-Brille wie z.B. die Microsoft HoloLens. In einer AR oder MR-Brille ist das Display transparent, wodurch die reale Umgebung wahrgenommen werden kann und gleichzeitig können diverse Objekte in der Umgebung eingeblendet werden. Oftmals werden auch Smartphones für eine AR Anwendung benutzt. Durch die Kamera des Smartphones kann die Umgebung auf dem Display dargestellt werden. Gleichzeitig können computergenerierte Objekte eingeblendet werden (siehe Ikea AR App). [6]

Mixed Reality

MR ist ein Überbegriff von VR und AR. Dieser Begriff vereint Augmented Reality und Virtual Reality, so gibt es Anwendungen und Hardware die als Mixed Reality bezeichnet werden. Z.B. die Microsoft HoloLens-Brille wird als MR-Brille vertrieben, da sie eine virtuelle Welt darstellen kann, jedoch auch mit der realen Welt interagieren kann und Objekte darin projizieren. [16]



Abbildung 7.1: MR-Brille Microsoft HoloLens (Quelle: <https://www.microsoft.com/de-de/hololens>)

Metaverse

In der Literatur scheint der Begriff Metaverse nicht klar definiert zu sein. Oftmals wird von einer virtuellen Welt gesprochen in der Personen mittels Avatare miteinander interagieren können. Die erweiterte Realität ([Augmented Reality](#)) wird aber ebenfalls mit dem Metavers in Verbindung gebracht. Die Interaktion mit der virtuellen Welt wird mittels [AR](#) bewerkstelligt. Das Wort ‘Metaverse’ ist zusammen gesetzt aus ‘meta’ und ‘universe’ was soviel bedeutet wie jenseits unseres Unviersums. [17]

7.2 Technologiewahl

Es stehen viele verschiedene Technologien für die Erstellung von VR Applikationen zur Verfügung.

7.2.1 Technologiekandidaten

Eine Auswahl an verschiedenen Technologien schafft eine Übersicht der aktuell verwendeten Produkte.

QuriVR

QuriVR basiert auf A-Frame einem Framework das ursprünglich von Mozilla entwickelt wurde. QuriVR ist darauf ausgelegt mit möglichst einfachen Mitteln möglichst schnell ein Ergebnis zu erhalten. Es ermöglicht das Erstellen eine VR Umgebung mit lediglichem Beschreiben der gewünschten virtuellen Realität in Worten. [31]

A-Frame

Basierend auf Three.js ist A-Frame ein Open Source Framework, das die Erstellung von VR Objekte und Umgebungen im Browser ermöglicht. Die einfache Beschreibung von Objekte durch HTML-Tags. [26]

Three.js

Die Three.js Bibliothek ist in JavaScript geschrieben und benutzt ebenfalls [WebGL](#). Da [WebGL](#) auf die Hardwarebeschleunigung des jeweiligen Gerätes setzt, ist die Library sehr performant. Auf GitHub hat das Open-Source-Projekt bereits über 1.7K Mitwirkende. [1] [2]

OpenSpace3D

OpenSpace3D ist eine freie Software um VR und AR zu entwickeln. Die Software bietet eine grafische Oberfläche, um die virtuelle Welt/Objekte zu erstellen. [11]

Babylon.js

Babylon.js ist in TypeScript geschrieben und ist somit für TypeScript- oder JavaScript-Projekte geeignet. Die 3D-Engine nutzt [WebGL](#)[24] und lässt sich in allen aktuellen Webbrowsern direkt ausführen. Um [VR/AR](#) Inhalte darzustellen, setzt Babylon.js [WebXR](#) ein. [22] [3]

Unity

[Unity](#) ist eine weit verbreitete [Game Engine](#)[14]. Die Hälfte aller Spiele für Konsole, Computer und Smartphones wurden mit [Unity](#) erstellt. [Unity](#) bietet sehr viel verschiedene Features. Somit kann [Unity](#) nicht nur für Spiele benutzt werden, sondern auch für [MR](#) Anwendungen. Mithilfe des [AR Foundations](#) Paketes z.B. können [Corss-Platform Augmented Reality](#) Applikationen erstellt werden. So ist Z.B. das Spiel Pokémon GO mit [Unity](#) & [AR Foundations](#) erstellt worden. [13]

[Unity](#) ist keine Open Source Applikation und gehört der ‘Unity Software Inc.’ Firma. Sofern die Software nicht von Unternehmen mit mehr als 100’000.- US-Dollar eingesetzt Jahresbruttoumsatz eingesetzt wird, darf die Unity Personal Edition unentgeltlich eingesetzt werden.

Adobe Aero

Adobe Aero ist ein Toolset für die Erstellung von [AR](#) Anwendungen. Da die Software aber Lizenzpflichtig ist und keine gratis Version verfügbar ist, wird Aero für diese Arbeit nicht infrage kommen. [12]

AR.js

Da die AR.js-Bibliothek auf JavaScript basiert, funktioniert eine AR.js-Applikation in jedem aktuellen Webbrowser und somit auch auf iOS und Android. Unter anderem unterstützt die Open Source Bibliothek das Tracken von Bilder sowie standortbezogene AR Anwendungen. [15]

7.2.2 Anforderungen an Technologien

Damit ein Prototyp des gewählten Realisierungsansatzes (siehe [Kapitel 3.3.1](#)) erfolgreich implementiert werden kann, werden Technologien benötigt, welche den Anforderungen aus [Kapitel 6](#) genügen. Nachfolgend ein Auszug jener Anforderungen, welche die Technologiewahl betreffen.

- 3D-Abbild von Lichtensteig erstellen
- VR Applikation über Browser verwenden
- Erstellen von 3D-Objekten für die Karte
- Interaktion mit Objekten auf der Karte
- Darstellung von Inhalten in Text- und Bildform
- Optional:
 - Importieren von Kartendaten aus Google Maps oder Open Street Map
 - VR Applikation über VR-Brillen verwenden
 - Darstellung von Inhalten in Form von Videos

7.2.3 Auswahl der passenden Kandidaten

Aufgrund des vorhandenen Funktionsumfangs und der Tatsache, dass uns im Rahmen der Bachelorarbeit durch Mitarbeitende des Instituts an der OST zusätzliches Knowhow zur Verfügung steht, wird Unity als Haupt-Technologiekandidat gesehen. Unity bietet ein sehr umfangreiches Repertoire an Funktionalität und wird seit vielen Jahren sehr stark im Bereich [MR](#) eingesetzt. Dadurch ist auch eine grosse Community um das Produkt entstanden, was zusätzlich für die Technologie spricht.

Blender bietet im Bereich der 3D-Modellierung sehr interessante Werkzeuge und hat einen enormen Funktionsumfang. Es ist eine verbreitete 3D-Grafiksoftware, die unter anderem auch von kleineren Film- und Gamestudios eingesetzt wird und gerade im Bereich der Modellierung gut ergänzend zu Unity eingesetzt werden kann.

Es kann sein, dass im Verlauf der Arbeit erkannt wird, dass der Einsatz von Unity und Blender durch weitere Technologien ergänzt werden muss, um zu einem optimalen Ergebnis zu kommen oder man komplett auf eine dieser Technologien verzichten sollte. Um einen möglichen Technologiewechsel während der Implementierung zu verhindern, wird daher vorab Unity und Blender genauer untersucht.

In Form von Experimenten wird überprüft, ob die Anforderungen (siehe [Kapitel 7.2.2](#)) welche an die Technologiekandidaten gestellt werden auch damit umgesetzt werden können.

7.3 Experimente

Um sicherzustellen, dass mit den ausgewählten Technologien alle Anforderungen umgesetzt werden können, sind einige Funktionalitäten in Form eines POC implementiert. Diese Anwendungen sind in einem separaten Gitlab Repository abgelegt und werden nachfolgend als Experimente bezeichnet.

7.3.1 Experiment 1: 3D-Abbild von Lichtensteig erstellen

Motivation: Um die Umgebung rund um das ZB Areal in einer virtuellen Welt darzustellen, muss die Umgebung nachgezeichnet oder bereits bestehende Daten dafür verwendet werden. Mit Open-Street-Maps sind als Open Source bereits viele Kartendaten vorhanden. So ist auch Lichtensteig in Open Street Maps abgebildet. Die Quelloffene Kartenapplikation beinhaltet nicht nur eine Verkehrskarte, sondern Höhenangaben der Landschaften und Gebäude. Dank dieser Informationen lässt sich relative einfach eine 3D-Karte erstellen.[25]

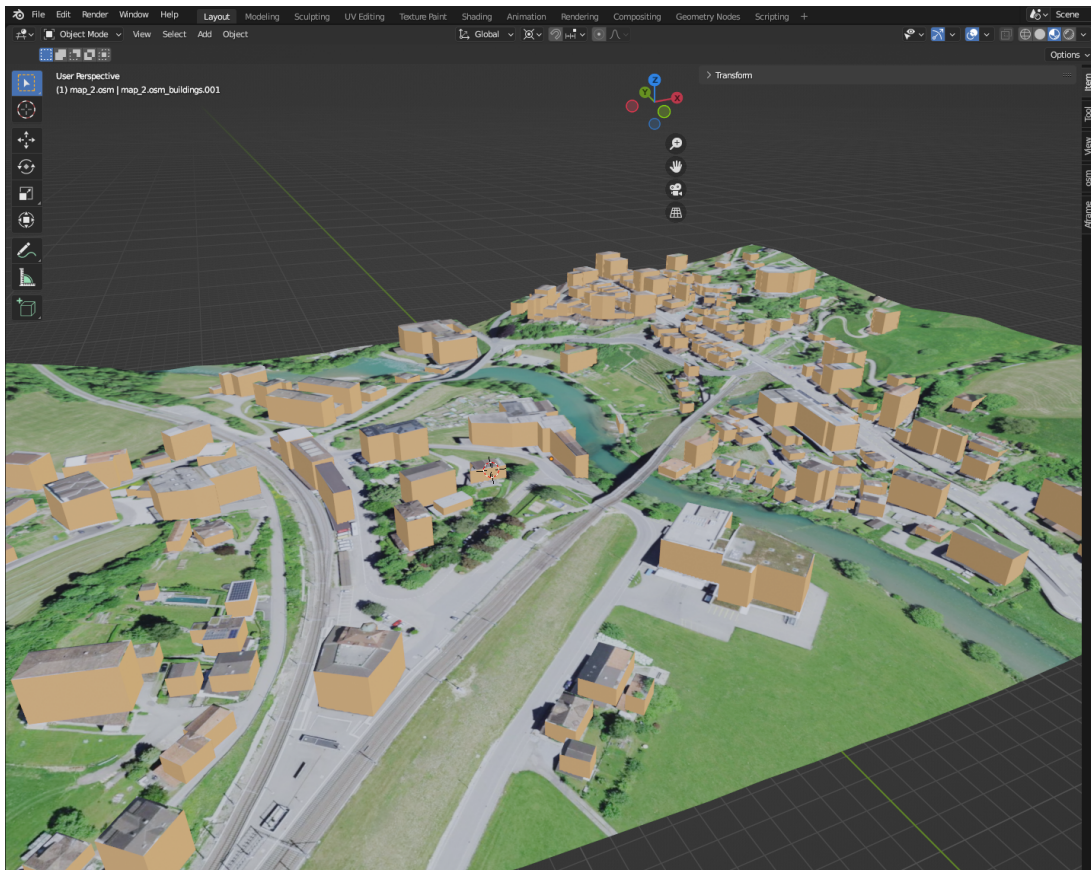


Abbildung 7.2: ZB Areal- und Umgebungs-Darstellung in 3D in Blender (Quelle: Eigene Darstellung)

Umsetzung: Das Add-on [Blender-OSM](#)[4] kann in Blender verwendet werden, um Kartenmaterial direkt von Open Street Maps zu importieren. Dadurch kann eine 3D-Karte mit den Landschafts- und Gebäudedaten des gewünschten Ausschnittes erstellt werden. Durch [Mapbox](#)[20] lassen sich die Kartendaten mit Satellitenbilder ergänzen und dadurch realitätsnah nachbilden. Die Satellitenbilder werden für die Umgebung und für die Dächer der 3D-Gebäude benutzt. Da Umgebungsobjekte wie z.B. Bäume, lediglich aus Satellitenbildern bestehen, werden diese nicht in der dritten Dimension dargestellt, sondern sind als flache Objekte ersichtlich. Weil ebenfalls keine Informationen über die Dachschräge auf Open-Street-Maps vorhanden sind, werden die Dächer als Flachdächer dargestellt.

Die Gebäude sind nur als groben Umriss vorhanden (zumindest in Lichtensteig). Deswegen werden die Gebäude nur mit einem kleinen Detailgrad in 3D dargestellt. Die Anpassung zwischen Gebäude und Satellitenbilder ist mit etwas Aufwand verbunden, da die Bilder und Kartendaten nicht zu 100 % übereinstimmen. Aus diesem Grund ist im Experiment auch ersichtlich, dass die Gebäude nicht ganz exakt mit dem Untergrund übereinstimmen. Dies kann mit entsprechendem Aufwand von Hand nachjustiert werden.

Die Fassaden der Gebäude müssen noch durch zusätzliche Bilder ergänzt werden, ansonsten werden sie nur einfarbig dargestellt (siehe [Abbildung 7.2](#)). Im Experiment wird dies nicht umgesetzt. Diese Darstellung kann anschliessend als Grundbaustein für die virtuelle Welt genutzt werden. Weitere darzustellende Objekte können ebenfalls mit Blender designt und in die VR-Welt eingebunden werden.

7.3.2 Experiment 2: VR Applikation über Browser verwenden

Motivation: Das 3D-Abbild von [Experiment 1](#) muss in [Unity](#) verwendet werden können. Dieses Experiment soll aufzeigen, ob und wie mittels [Unity](#) und Docker das Areal im Browser zur Verfügung gestellt werden kann. Da [Unity WebGL](#) unterstützt, müsste es möglich sein, das Projekt als Website auf einem Webserver zugreifbar zu machen.

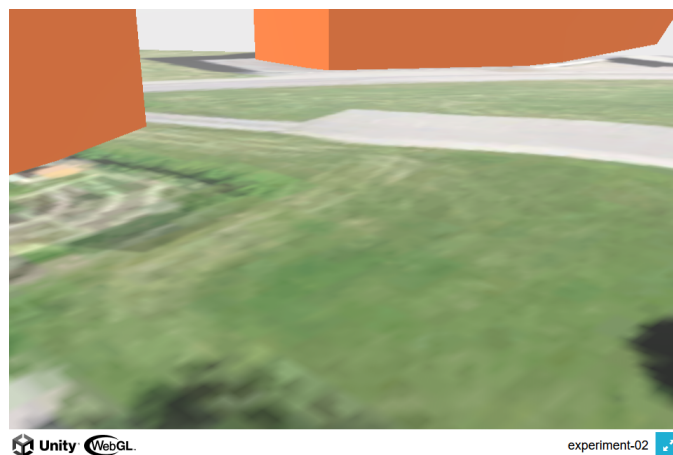


Abbildung 7.3: Screenshot Webseite über Docker zur Verfügung gestellt. (Quelle: Eigene Darstellung)

Umsetzung: [Unity](#) unterstützt verschiedene Plattformen. Neben Desktop und Mobile Plattformen werden auch diverse [MR-Headsets](#) und [WebGL](#) unterstützt. Je nachdem für welche Hardware die Applikation entwickelt werden soll, muss entsprechend eine andere Plattform gewählt werden. Damit die Anwendung im Browser dargestellt

werden kann, wird [WebGL](#) als Plattform verwendet. [Unity](#) erstellt beim Builden die notwendigen Dateien, die für ein Webserver benötigt werden. Standardmässig werden die Daten jedoch im [Gzip](#) Format komprimiert. Wenn der Webserver nicht [Gzip](#) als Content-Encoding unterstützt, kann die Applikation so nicht zur Verfügung gestellt werden.

Ohne eingeschaltete Komprimierung werden Javascript und Web-Assembly Dateien erstellt. Diese Dateien können einfach mit einem Webserver wie [NGINX](#) zur Verfügung gestellt werden. Für die lokale Entwicklung der Applikation führt [Unity](#) ein lokaler Webserver aus, um die Webseite im Browser anzuzeigen. Damit die Applikation der Öffentlichkeit verfügbar gemacht werden kann, wird Docker eingesetzt. Als Docker Container wird ein [Alpine](#) Linux mit [NGINX](#) verwendet. Auf der [Abbildung 7.3](#) ist ersichtlich, wie die Webseite aussieht.

7.3.3 Experiment 3: Kameraführung und Interaktion mit 3D-Objekten

Motivation: Der Prototyp soll die Interaktion mit diversen Objekten ermöglichen. Dieses Experiment soll die Machbarkeit aufzeigen, wie ein Objekt in der virtuellen Welt ausgewählt und angeklickt werden kann. Damit dies möglich ist, muss sichergestellt werden, dass die Ansicht geändert werden kann. So soll das Sichtfeld herumgeschwenkt werden können, um Objekte, die hinter der betrachtenden Person stehen, auswählen zu können.

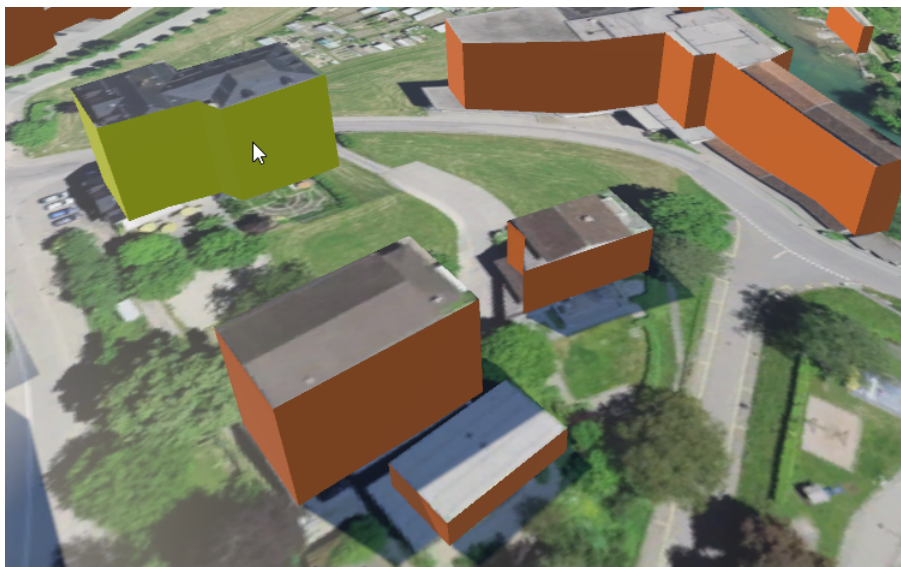


Abbildung 7.4: Objekt Selektion in Unity (Quelle: Eigene Darstellung)

Umsetzung: Um eine Interaktion mit platzierten Objekten zu ermöglichen, müssen Benutzereingaben registriert werden können. In einer Webanwendung sind dies vor allem Maus und Tastatur bzw. Touchscreen auf Mobilgeräte.

Die Erkennung von Eingaben und anschliessende Aktionen, werden in Unity mit C#-Skripte gesteuert. Solche Skripte können allen im Projekt enthaltenen oder sogenannten leeren Objekten zugewiesen werden. So ist ein 'Selection'-Skript einem leeren 'Selection-Objekt' zugewiesen, das die Steuerung und Selektion von Objekten koordiniert. Wenn also auf das Chössi Theater geklickt wird, wird dies von Unity erkannt und zu Demon-

strationszwecken die Wandfarbe geändert (siehe grün/gelbes Gebäude in [Abbildung 7.4](#)).

Die Karte des ZB-Areals ist ein einzelnes Objekt mit unterschiedlichen Layern (Boden, Wände der Gebäude und Dächer). Die Gebäude sind jedoch nicht als einzelnes Objekt vorhanden. Deswegen müssen Gebäude, bei denen weitere Informationen angezeigt werden sollen, als einzelne Objekte aus Blender exportiert und in [Unity](#) importiert werden.

Die Kameraführung wird ebenfalls über C#-Skripte gesteuert. Wenn die Maus bewegt wird, während die linke Maustaste gedrückt gehalten wird, dreht sich das Bild entsprechend in [Unity](#). Mit den Pfeil-Tasten oder ASWD-Tasten ist es möglich, sich in der virtuellen Welt fortzubewegen. Für den Prototypen ist die freie Fortbewegung jedoch nicht relevant.

7.3.4 Experiment 4: Darstellung von Inhalten

Motivation: Zusätzlich zur virtuellen Welt müssen in der Applikation diverse Schaltflächen zur Bedienung dargestellt werden. Es muss möglich sein Text, Informationen und auch eingebettete Videos anzuzeigen. Dies kann innerhalb Unity durch das Erstellen von Objekten umgesetzt werden. Jedoch sind die gestalterischen Möglichkeiten begrenzt und das Kreieren ist mit sehr viel Aufwand verbunden. Zudem muss nach Änderungen dieser Objekte jedes mal wieder ein kompletter Build der Unity App erfolgen. Dies ist nicht wirklich eine komfortable Lösung, vor allem in Anbetracht dessen, dass der Prototyp mit hoher Wahrscheinlichkeit in Zukunft weiterentwickelt und angepasst wird. Gängige UI-Libraries bieten Komponenten, welche als Overlay über andere Inhalte dargestellt werden können. Floating Button, Modal oder Drawer sind Beispiele für solche Komponenten. Ist das Einbinden der Unity App in einer [SPA](#) möglich, können eben solche UI Komponenten für die Darstellung von Inhalten eingesetzt werden.

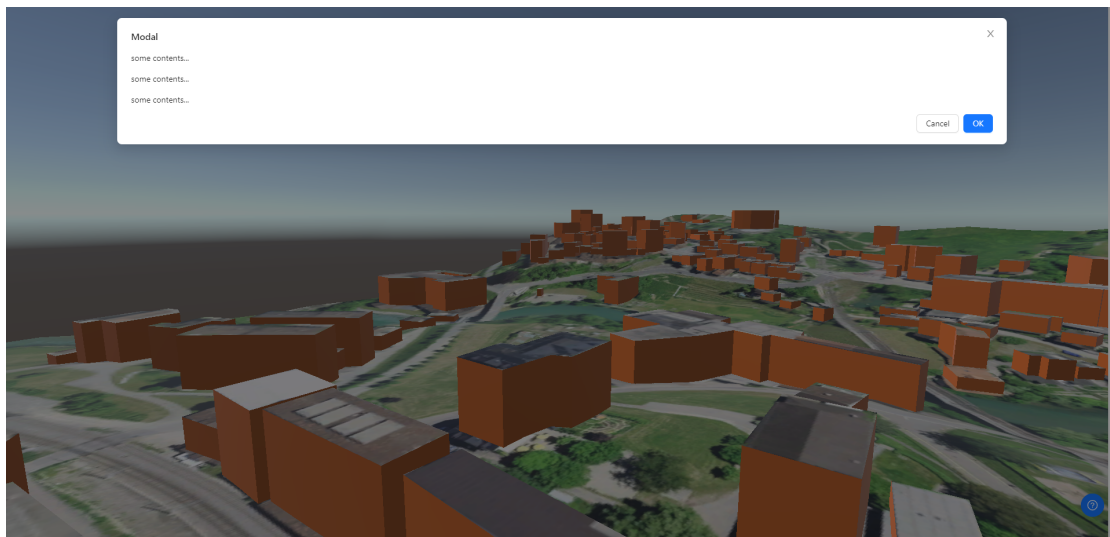


Abbildung 7.5: Floating Button und Modal in React (Quelle: Eigene Darstellung)

Umsetzung: Für React gibt es die Library [React Unity WebGL](#) [29]. Damit können [Unity WebGL](#) Builds sehr einfach in einer React App integriert werden. Innerhalb React steht nach der Installation des Pakets eine Unity-Komponente zur Verfügung, welche

einen vorab erzeugten Build in der WebApp einbindet. Somit kann das [Unity](#) Projekt im Browser auf einer Website dargestellt werden, ohne auf die gewohnten Features einer [SPA](#) zu verzichten.

Die Library stellt Schnittstellen für die Kommunikation von React zu Unity und umgekehrt zur Verfügung. Über Eventlistener kann eine nahtlose Integration des Unity Projekts in der React App erfolgen. Wird auf ein Gebäude aus Experiment 1 geklickt, muss über den Eventlistener eine Nachricht an React gesendet werden, damit React darauf reagieren kann. Als Modal stellt React dann weitere Informationen über das angeklickte Gebäude dar (siehe [Abbildung 7.5](#)).

7.3.5 Experiment 5: Objekte frei platzieren

Motivation: Mit diesem Experiment soll aufgezeigt werden, wie Objekte frei im Areal platziert werden können. Es soll für die Benutzenden möglich sein, vordefinierte Objekte frei auf dem Areal zu platzieren. Diese Objekte sollen im Backend abgespeichert werden, damit sie persistent bleiben und andere Benutzende die platzierte Objekte ansehen können.



Abbildung 7.6: Experiment 5 beim Platzieren einer Kugel. (Quelle: Eigene Darstellung)

Umsetzung: Für die vordefinierte Objekte werden sogenannte Prefabs benutzt. Im Experiment bestehen die aus einem Zylinder, einem Quadrat und einer Kugel. Um das gewünschte Objekt zu platzieren, ist im Experiment je eine Schaltfläche am unteren Bildrand angebracht. Beim Auswählen eines Objektes erscheint das Objekt und kann mit der Maus platziert werden (siehe: [Abbildung 7.5](#)). Anschliessend wird ein weiteres Skript aufgerufen, das die Position in einer JSON-Datei speichert. Anstatt die Koordinaten in einer JSON-Datei zu speichern, werden beim Prototypen die JSON-Daten über die API an das Backend gesendet.

8. Design des Prototypen

Dieses Kapitel beschreibt das Domain Model sowie die Softwarearchitektur des Prototypen. Die Architektur wird anhand von [C4-Diagramms](#) auf verschiedenen Ebenen dargestellt. Ein Deployment-Diagramm gibt zudem eine Übersicht darüber, in welcher Form die Applikation schlussendlich ausgeliefert wird.

8.1 Domain Model

Die 'Scene' beschreibt die übergreifende Komponente im Domain Model. Sie enthält nur ein Attribut welches den Namen abspeichert, steht jedoch im Endeffekt für ein komplettes VR-Modell. Durch das Einführen dieser Abstraktionsebene ist es möglich, die Applikation auch für andere Kartenausschnitte bzw. VR-Modelle einzusetzen.

Auf der Szene kann textuelles Feedback in Form von 'Ideas' hinterlegt werden. Zudem sind die 'Layouts' der Szene angebinden. Diese Layouts beschreiben eine Arealgestaltung, welche sowohl 'Comments', 'Likes' als auch 'ObjectComponents' enthalten kann. Letzteres sind Objekte, welche auf dem VR-Modell platziert werden können und speichern daher Position sowie Rotation in drei Dimensionen (X,Y, Z) ab. Zudem bestehen Objekte aus einem Namen und dem Attribut 'isCustomComponent', welches festlegt, ob es sich um ein vordefiniertes Objekt handelt oder durch Benutzende erfasst wurde.

Objekte und Layouts können mit Reaktionen in Form von Likes und Kommentaren versehen werden. Ein 'Prefab' beschreibt ein platzierbares Objekt und referenziert dabei eine Datei, welche das VR-Objekt dazu definiert. In der 'Description' werden Informationen zu den einzelnen Objekten abgespeichert. Diese Informationen sollen nacher durch Benutzende angezeigt werden können. Vordefinierte Objekte können dabei alle gängigen HTML Elemente verwenden, um den Beschreibungstext zu formatieren und mit Bildern und ähnlichem zu gestalten. Bei den eigens erstellten Objekten kann lediglich Text in der Beschreibung gespeichert werden.

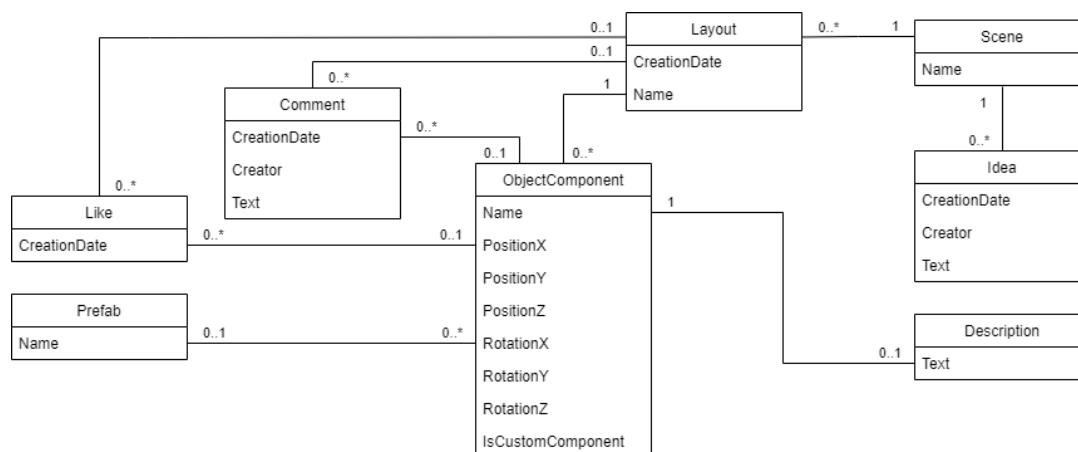


Abbildung 8.1: Domänenmodell (Quelle: Eigene Darstellung)

8.1.1 Constraints

Es gibt einige Einschränkungen, welche zur korrekten funktionsweise der Applikation beitragen und daher hier aufgelistet werden.

- Vordefinierte Layouts müssen mit einer Id <0 erfasst werden. Damit diese durch die Benutzenden nicht verändert werden können.
- Eine Scene für ein jeweiliges VR-Modell muss vorab in der Datenbank erfasst werden. Für diesen Anwendungsfall (zukunft.bahnhof Lichtensteig) wird eine Scene mit der Id = 1 erwartet. Entsprechend wird bei den API Aufrufen diese SceneId verwendet.
- Prefabs müssen vorab in der Datenbank erfasst werden. Der Dateiname der Prefabs muss dabei mit der Id übereinstimmen, damit die korrekten Objekte in das VR-Modell geladen werden.

8.2 Systemarchitektur

C4 Modelle beschreiben die Architektur des Prototypen. Nebst den statischen Modellen, werden die wichtigsten Abläufe auch in Form von Ablaufdiagrammen aufgezeigt. Ein Deploymentdiagramm stellt zudem das Deployment der Applikation dar.

8.2.1 Kontextdiagramm

Die Benutzenden greifen auf das System zu und können dadurch Arealgestaltungen in Form von VR Modellen besichtigen. Informationen zum Projekt [ZB](#) sind den jeweiligen Objekten hinterlegt. Auch Interaktionen wie das hinterlassen von Likes und Kommentaren oder das Erfassen von Ideen können durchgeführt werden.

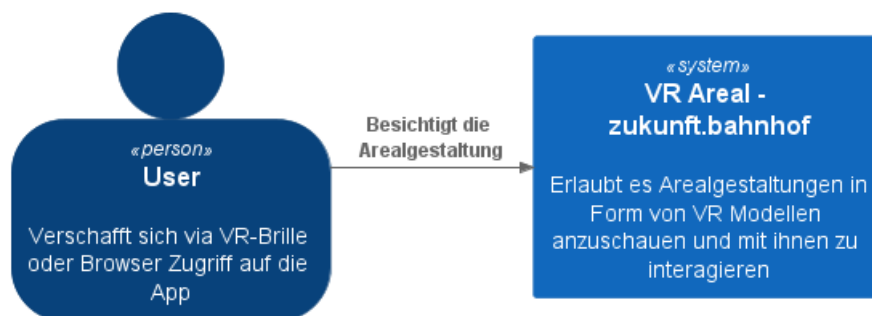


Abbildung 8.2: Kontextdiagramm (Quelle: Eigene Darstellung)

8.2.2 Containerdiagramm

Das System besteht aus einem Reverse Proxy, einem Client, einer API und einer Datenbank.

Der Reverse Proxy wird mit NGINX umgesetzt und liefert die SPA des React Clients an den Browser aus. Zudem leitet er die API-Anfragen an das Backend weiter. Dieses besteht aus einem ASP.NET Core WebAPI, welches fürs Laden und Speichern der Daten in der PostgreSQL Datenbank zuständig ist.

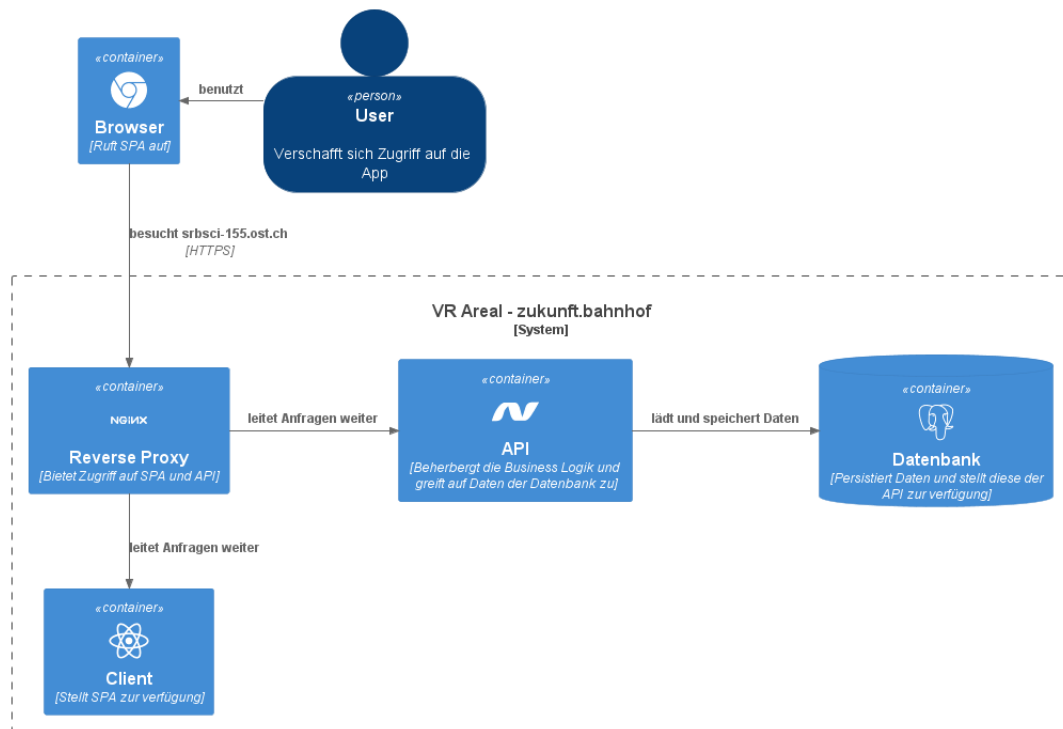


Abbildung 8.3: Containerdiagramm (Quelle: Eigene Darstellung)

8.2.3 Komponentendiagramm - API

Die API wird mit ASP.NET Core WebAPI realisiert. Konzeptionell wird die API pro Aufgabenbereich jeweils in separate Controller aufgeteilt. Die Controller sind für das Handling der Anfragen zuständig. Im Mapping werden Daten transformiert und von Modell-Klassen auf DTO-Klassen und umgekehrt gemappt. Zugriffe auf die Datenbank passieren ausschliesslich über den ApiDbContext.

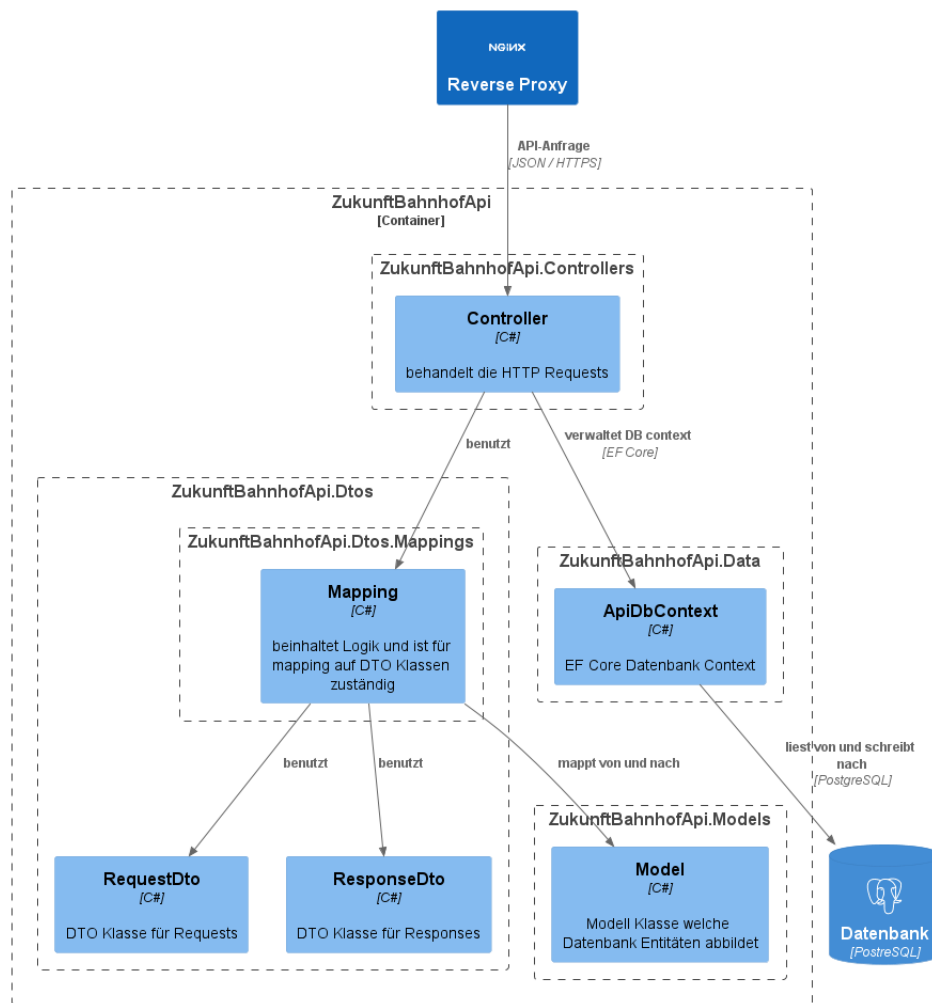


Abbildung 8.4: Komponentendiagramm API (Quelle: Eigene Darstellung)

8.2.4 Patterns und Konzepte

Sowohl im Frontend als auch im Backend werden Patterns und Konzepte eingesetzt, welche in den C4 Diagrammen nicht ausreichend dargestellt werden können. Auf jene wird in diesem Kapitel näher eingegangen.

Data Transfer Object

Für den Datenaustausch von Frontend und Backend werden DTO Klassen implementiert. So werden nur die Daten übermittelt, welche auch wirklich vom Frontend benötigt werden. Dies ist aus Security- aber vor allem auch aus Performancegründen sinnvoll. Zudem wird durch die Verwendung von DTOs eine Unabhängigkeit gegenüber den Datenbank-

tabellen bzw. Modell-Klassen erreicht, was ein grosser Vorteil im Bereich Maintainability darstellt.

Beim Mapping zwischen den DTO- und den Modell-Klassen wird bewusst auf den Einsatz von Libraries und Packages verzichtet. Dadurch werden weitere externe Abhängigkeiten vermieden.

Entity Framework Core

Um die Zugriffe auf die Datenbank sicher und einfach zu vollziehen, wird Entity Framework Core als OR-Mapper eingesetzt. Somit werden alle benötigten Datenbanktabellen auf Modellklassen im Backend abgebildet. Dadurch wird der Einsatz von Loading-Strategien wie Eager- oder Lazyloading ermöglicht sowie das absetzen von SQL-Injections unterbunden.

8.2.5 Deployment

Das Deploymentdiagramm beschreibt wie die einzelnen Komponenten der Applikation ausgeliefert werden.

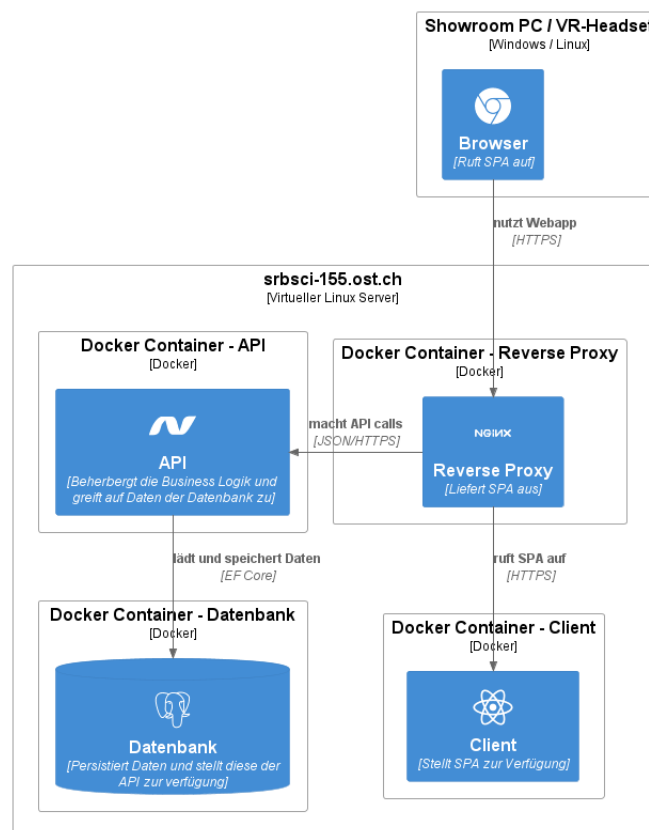


Abbildung 8.5: Deploymentdiagramm (Quelle: Eigene Darstellung)

Die Komponenten Webserver, API und Datenbank werden als Docker Services definiert und in separate Docker Container ausgeliefert. Durch den Einsatz von Dockercompose werden alle Komponenten zentral an einer Stelle konfiguriert und zusammengestellt, was die Wartung vereinfacht. Dies ermöglicht einen einfachen Betrieb auf beliebiger Hardware. Auch der nachträgliche Umzug auf eine andere Maschine oder zu einem Hostingpartner ist durch die Dockerisierung wesentlich einfacher zu bewerkstelligen.

8.3 Wireframes

Wireframes zeigen grafisch auf, wie das User Interface aufgebaut ist.

8.3.1 Startseite

Die Startseite ist schlicht gehalten und zeigt dem Benutzer direkt nach Abschluss des Ladevorgangs das VR Areal des Projekts ZB. In der unteren rechten Ecke kann man über das Info-Icon Informationen zum Projekt einblenden. Das Hilfe-Icon gibt Auskunft über die Bedienung der Applikation. Über das Glühbirnen-Icon kann man neue Ideen zum Projekt verfassen. Alle auswählbaren Objekte auf der Karte sind farblich markiert, sodass man sofort erkennt womit man interagieren kann.



Abbildung 8.6: Wireframe Startseite (Quelle: Eigene Darstellung)

8.3.2 Idee hinzufügen

Beim Erfassen einer Idee kann Name und Idee hinzugefügt werden. Mit einem Klick auf die Schaltfläche 'Speichern' wird die Idee abgespeichert.

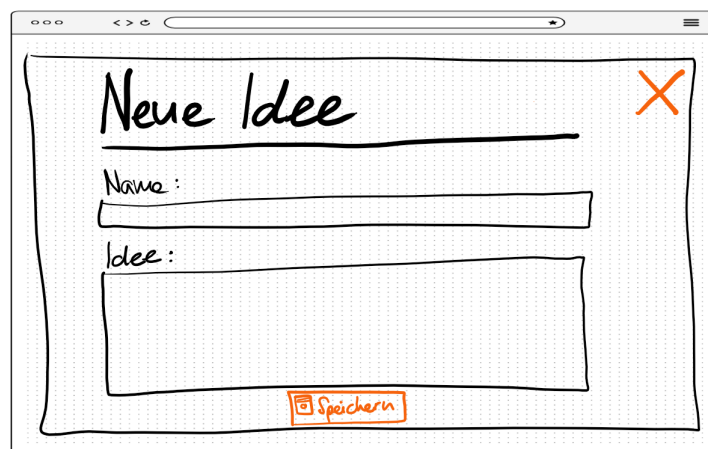


Abbildung 8.7: Wireframe Idee hinzufügen (Quelle: Eigene Darstellung)

8.3.3 Informationen zu einem Objekt

Wird ein markiertes Objekt ausgewählt, werden Informationen in Form eines Overlays dargestellt. Der Inhalt ist scrollbar und kann sowohl aus Text, wie auch aus Bildern oder eingebetteten Videos bestehen.

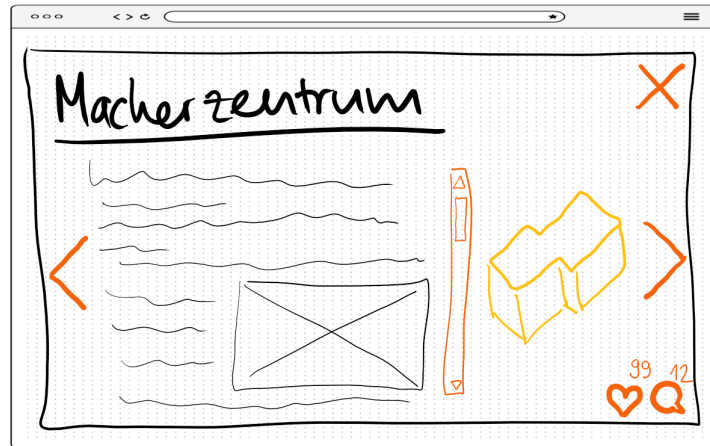


Abbildung 8.8: Wireframe Infoseite 1/2 (Quelle: Eigene Darstellung)

Über die Pfeil-Icons rechts und links, ist eine Navigation auf weitere Objekte möglich. Im Weiteren kann man über die Like- und Kommentar-Icons auf Objekte reagieren. Das Overlay lässt sich über die Schaltfläche oben rechts schliessen.

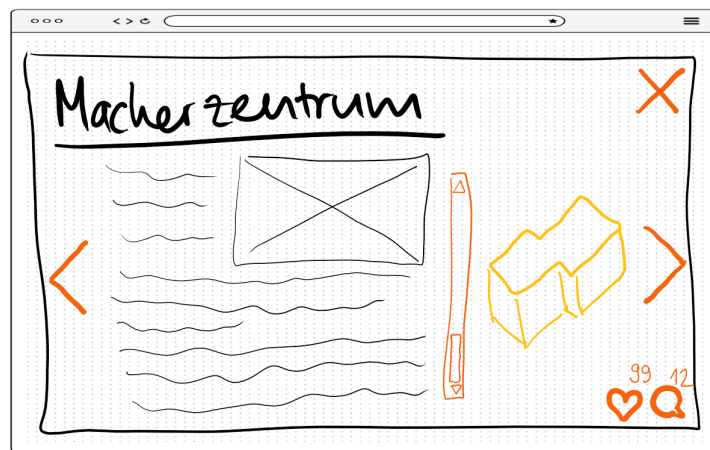


Abbildung 8.9: Wireframe Infoseite 2/2 (Quelle: Eigene Darstellung)

8.3.4 Kommentare

Kommentare werden in Form einer scrollbaren Liste dargestellt. Über den Pfeil oben links kann zurück zum Objekt navigiert werden. Via Schaltfläche oben rechts kann das Overlay geschlossen werden. Danach wird wieder die Karte des Areals dargestellt. Über das Stift-Icon ist es möglich ein neuer Kommentar zu verfassen.

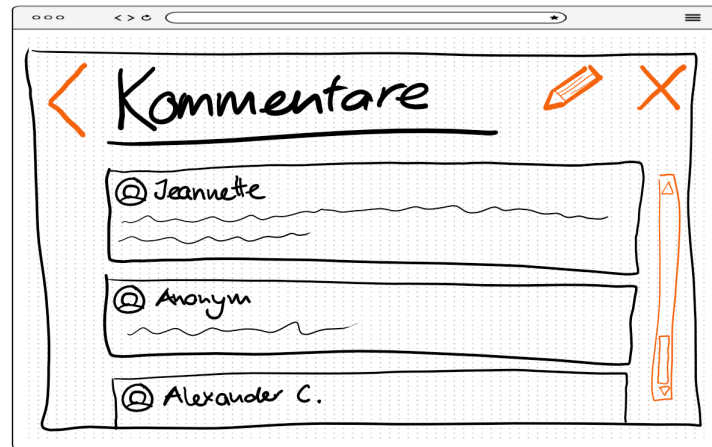


Abbildung 8.10: Wireframe Kommentare (Quelle: Eigene Darstellung)

8.3.5 Kommentar hinzufügen

Bei einem neuen Kommentar kann sowohl der Name, wie auch der Kommentartext verfasst werden. Mit einem Klick auf die Schaltfläche 'Speichern' wird der Kommentar abgespeichert.

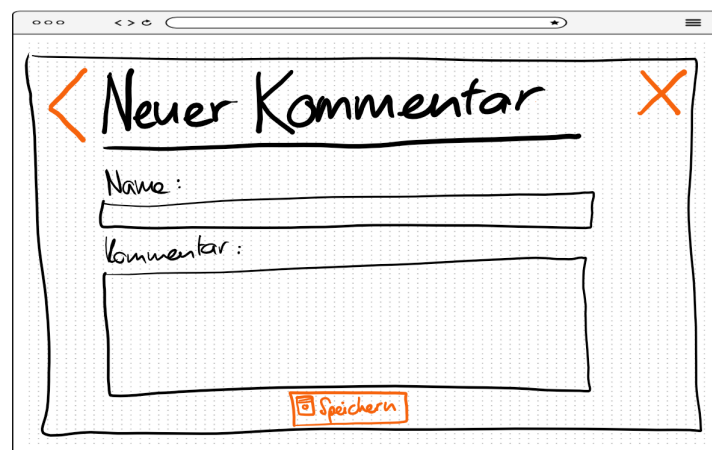


Abbildung 8.11: Wireframe Kommentar hinzufügen (Quelle: Eigene Darstellung)

9. Implementierung

Dieses Kapitel beschreibt ausgewählte Implementationsaspekte sowie eingesetzte Tools, Frameworks und Libraries. Zudem wird auf das Testing eingegangen.

9.1 Hardware

Für die Auslieferung des Prototyps wird ein virtueller Linux Server der OST verwendet.

Bezeichnung	Wert
Hostname (FQDN):	srbsci-155.ost.ch
IP-Adresse:	152.96.86.155
Subnetzmaske:	255.255.254.0
Betriebssystem:	Ubuntu 20.04 LTS
Arbeitsspeicher:	4GB
Festplattenspeicher:	50GB

Tabelle 9.2: Informationen zur eingesetzten Hardware

9.2 Technologiestack

In diesem Projekt wird eine Vielzahl von externen Technologien wie Tools, Frameworks und Libraries eingesetzt. Nachfolgend wird beschrieben, zu welchem Zweck und wo diese eingesetzt werden.

9.2.1 Blender

Blender wird für das Erstellen des 3D-Modells vom Areal rund um den Bahnhof in Lichtensteig genutzt. Dabei werden Kartendaten von Open Street Map importiert. Dies ermöglicht, die Gebäude wie z.B. das Chössli Theater in 3D darzustellen und als separates Objekt in Unity einzubinden. Auch ein Layer mit einer Satellitenaufnahme wird in Blender über das Modell gelegt, sodass sowohl der Untergrund wie auch die Dächer der Gebäude realitätsgetreu dargestellt werden können.

9.2.2 Unity

Das vorab mit Blender erstellte Modell wird in [Unity](#) importiert und dort erweitert. In [Unity](#) werden zusätzlich die Kameraführung wie auch die Beleuchtung bzw. der Schattwurf definiert. Nebst den bestehenden Objekten im Modell werden im Unityprojekt zusätzlich Prefabs hinterlegt. Prefabs sind Vordefinierte Objekte, welche Benutzende des Prototyps auf dem VR-Modell platzieren können.

Auch ein wenig Programmlogik befindet sich im Unityprojekt in Form von Skripten. Dies ist C# Code, welcher für verschiedenste Dinge, wie beispielsweise die Kameraführung im VR-Modell, genutzt werden kann.

9.2.3 Asp.NET Core WebAPI

Die API im Backend wird mit Asp.NET Core realisiert. Somit wird eine dem Projektteam bereits bekannte Technologie eingesetzt, welche weit verbreitet ist und den Einsatz von Entity Framework Core ermöglicht.

Da auch die [Unity](#) Skripte in C# implementiert werden, bietet es sich an weitere Produkte des .NET Stacks zu verwenden, um die Anzahl verwendeter Programmiersprachen klein zu halten.

9.2.4 React

Das Frontend wird als [SPA](#) mit der [React](#) Library implementiert. Auch [React](#) wurde vom Projektteam bereits in anderen Projekten verwendet und ist somit keine Unbekannte. Durch eine sehr breite Community und viele Anwender, wird diese Library seit vielen Jahren sehr gerne genutzt.

9.2.5 NGINX

[NGINX](#) fungiert sowohl als Webserver, wie auch als Reverse Proxy.

9.2.6 PostgreSQL

Zur Persistierung der Daten wird ein PostgreSQL Server eingesetzt.

9.2.7 Docker und Dockercompose

Um die gesamte Applikation portierbar zu machen und somit einfacher auszuliefern, wird Docker sowie Dockercompose eingesetzt. Jede Komponente der Applikation wird dabei als separater Docker Container konfiguriert. Die Dockercompose Konfiguration beschreibt die Abhängigkeiten der einzelnen Containern und enthält weitere Definitionen, wie beispielsweise Umgebungsvariablen.

9.2.8 React-Unity-WebGL

Das [Unity](#) Projekt wird mithilfe dieser Library in die [React SPA](#) eingebunden. Zudem wird sie für die Kommunikation zwischen [Unity](#) und [React](#) verwendet.

9.2.9 Entity Framework Core

Die Datenbank wird mittels dem OR Mapper Entity Framework Core angebunden. Somit kann die API auf sichere und einfache Art auf die Datenbank zugreifen.

9.2.10 react-cookie

Die JavaScript Library react-cookie wird zur Erstellung von Cookies im Browser verwendet.

9.2.11 xUnit

Das Testing Framework xUnit wird im Backend zur Erstellung von Interationstests eingesetzt.

9.2.12 Testcontainers

Das NuGet Package Testcontainers wird eingesetzt, um Testdatenbanken fürs Integrationstesting zur Laufzeit programmatisch zu starten.

9.2.13 Ant Design

Als UI Library wird Ant Design eingesetzt. Ant Design ist für verschiedene Plattformen/Frameworks verfügbar und bietet für React eine sehr grosse Auswahl an Komponenten und eine ausführliche Dokumentation an. Durch eine umfangreiche Community wird die Library stets weiterentwickelt und verbessert.

9.3 Testing

Die Businesslogik in der API muss auf ihre Korrektheit überprüft werden. Da beim Unit Testing lediglich die internen Komponenten wie beispielsweise Controllerklassen getestet werden können und die Anbindung an die Datenbank nur gemockt wird, wird bewusst darauf verzichtet.

Stattdessen wird die API mittels Integrationstesting getestet. Dies bietet den Vorteil, dass nebst den einzelnen Units (Methoden, Funktionen) auch das komplette Zusammenspiel der API mit den umliegenden Systemkomponenten auf Korrektheit und Funktionalität untersucht werden kann.

9.3.1 Setup

Für das Testing wird die Visual Studio Solution ZukunftBahnhofApi um ein neues Projekt ZukunftBahnhofApi.Test.Integration ergänzt. Die Namensgebung erlaubt es zukünftig weitere Testarten wie beispielsweise ein Projekt für Unit Tests hinzuzufügen.

Das Projekt ZukunftBahnhofApi.Test.Integration verwendet folgende NuGet Packages:

Package	Version	Einsatzzweck
coverlet.collector	3.2.0	Coverlet wird zur Erfassung und Dokumentation der Code Coverage verwendet.
FluentAssertions	6.11.0	Mithilfe von FluentAssertions können die Assertions der Tests auf natürlichere Weise definiert werden.
JunitXml.TestLogger	3.0.124	Dieser Testlogger wird verwendet, um Reports im XML Format zu erstellen, welche danach direkt in der Gitlab Pipeline eingebunden werden können.
Microsoft.AspNetCore.Mvc.Testing	6.0.16	Stellt die WebApplicationFactory Klasse zur Verfügung.

Package	Version	Einsatzzweck
Testcontainers	3.1.0	Mithilfe von Testcontainers werden zur Laufzeit Wegwerf-Instanzen von Docker Containern erstellt, welche für das Integration Testing benötigt werden.
Testcontainers.PostgreSQL	3.1.0	Dieses Package unterstützt die Erstellung von PostgreSQL Containern.
xUnit	2.4.2	Das Testing Framework xUnit bildet die Grundlage der Testumgebung für die API.
xUnit.Priority	1.1.6	Das Testing Framework xUnit bildet die Grundlage der Testumgebung für die API.
xUnit.runner.visualstudio	2.4.5	Dieser Testrunner ermöglicht es, die Tests direkt im Testexplorer des Visual Studio auszuführen.

Tabelle 9.3: Setup Testumgebung

9.3.2 WebApplicationFactory

Die Klasse `WebApplicationFactory` wird über das Standardpackage `Microsoft.AspNetCore.Mvc.Testing` zur Verfügung gestellt. Mithilfe dieser Klasse wird eine In-Memory Applikation der `Asp.NET Core WebAPI` erstellt. Es wird ein Bootstrapping der `TestServer` Klasse durchgeführt und ein `HttpClient` zurückgegeben, welcher in den Tests für das Erstellen von `HTTP` Anfragen an die API verwendet werden kann. [21]

Durch Vererbung der `WebApplicationFactory` Klasse wird üblicherweise ein In-Memory `DbContext` anstelle des eigentlichen `DbContext` eingebunden, da durch die Testfälle nicht die produktive Datenbank verändert werden soll.

Da alle Komponenten des Prototyps, auch die Datenbank, in einem Docker Container ausgeführt wird, ist es sinnvoller auch das Testing entsprechend aufzubauen. Somit wird für den `DbContext` kein In-Memory Speicher verwendet. An dieser Stelle kommt dann das Package `Testcontainers` ins Spiel.

9.3.3 Testcontainers

Mit `Testcontainers` werden Docker Images vordefiniert, welche dann zur Laufzeit der Tests hochgefahren werden, um die Tests darauf auszuführen. Sobald alle Testfälle abgeschlossen sind, werden die Docker Container wieder durch `Testcontainers` abgeräumt. Ein separater Docker Container übernimmt dabei diese Orchestrierung.

Wie man in [Abbildung 9.1](#) gut sehen kann, wird dieser Container mit dem `Testcontainers` Image zuerst erstellt und gestartet. Dieser wiederum fährt danach die benötigten `PostgreSQL` Container hoch. Pro Testklasse wird jeweils ein separater Container erstellt. Da für jeden API Controller eine eigene Testklasse verwendet wird, werden hier mehrere Container erstellt. Auch ist erkennbar, dass die Tests auf die Erstellung der Docker Container warten, bevor sie ihre Anfragen an das Testsystem absenden.

9.3. TESTING

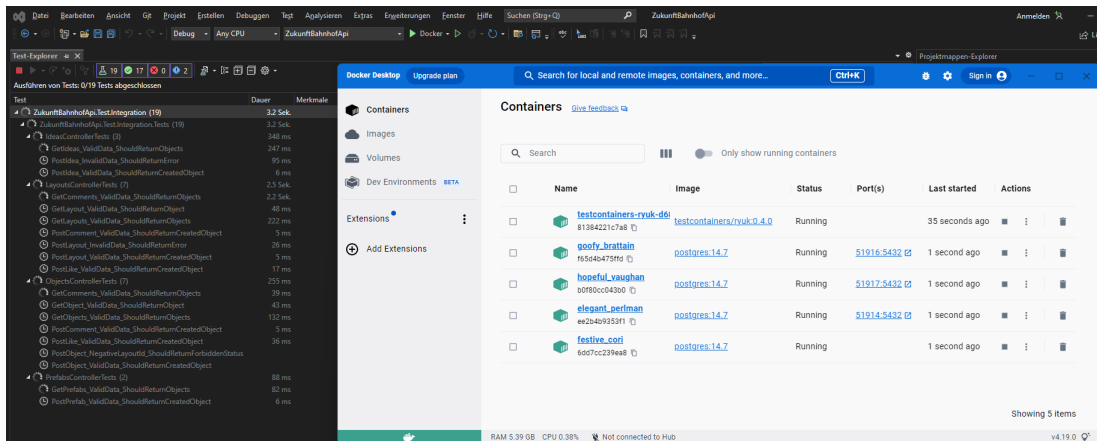


Abbildung 9.1: Lokale Ausführung der Integration Tests (Quelle: Eigene Darstellung)

9.3.4 Integration in CI/CD

Damit die Tests immer bei Codeänderungen erneut durchgeführt werden, wird das Ausführen der Tests in die Gitlab Pipeline integriert. Im Kapitel 9.6 wird dies genauer erläutert.

Bei jedem Push ins Gitlab Repository werden die Tests ausgeführt. Nach der Ausführung der Tests werden die Testresultate sowie die Testabdeckung gesammelt und direkt in Gitlab dargestellt (siehe Abbildung 9.2). Dies bietet eine sehr schnelle Übersicht über die Testergebnisse und die Testabdeckung, was gerade bei Merge Requests extrem nützlich ist.

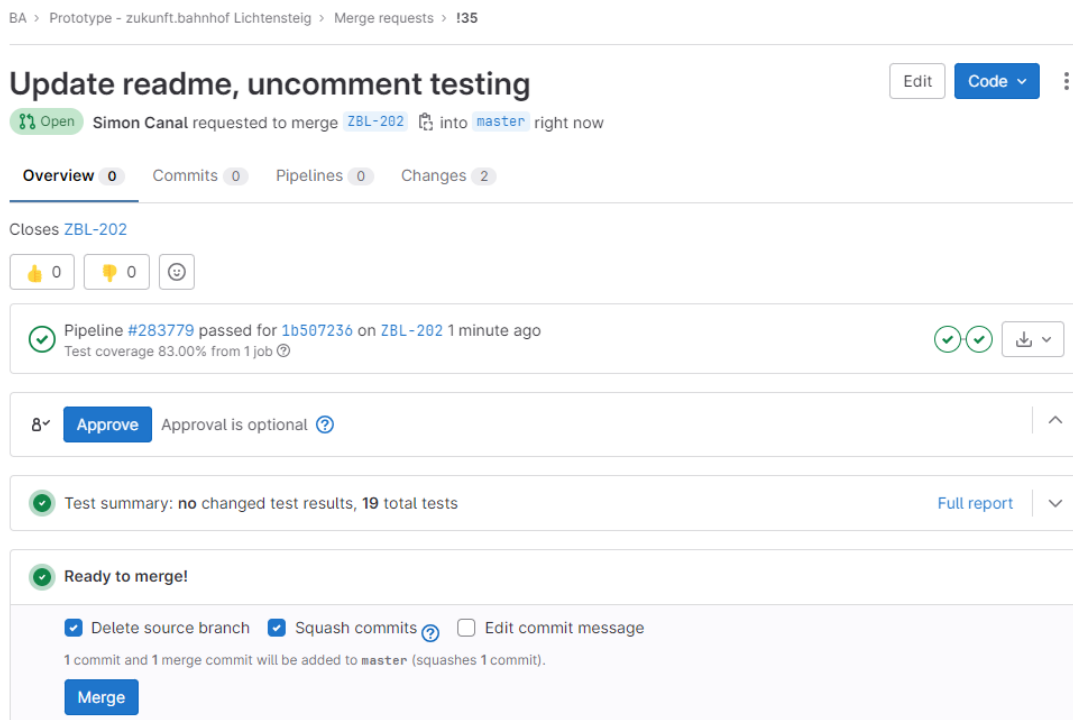


Abbildung 9.2: Testcoverage in Gitlab Merge Request (Quelle: Eigene Darstellung)

Möchte man die Testergebnisse der einzelnen Tests anschauen, ist dies über den jewei-

9.4. UNITY

ligen Pipeline Run möglich. Wie in [Abbildung 9.3](#) zu sehen ist, wird im Register Tests jeder Test und dessen Ergebnis aufgeführt. Somit ist für den Reviewer des Merge Request gut ersichtlich, ob die Testabdeckung weiterhin eingehalten wird und ob Tests fehlgeschlagen sind. Bei Letzterem lässt sich zudem direkt anzeigen, welche Tests nicht erfolgreich durchgeführt werden konnten.

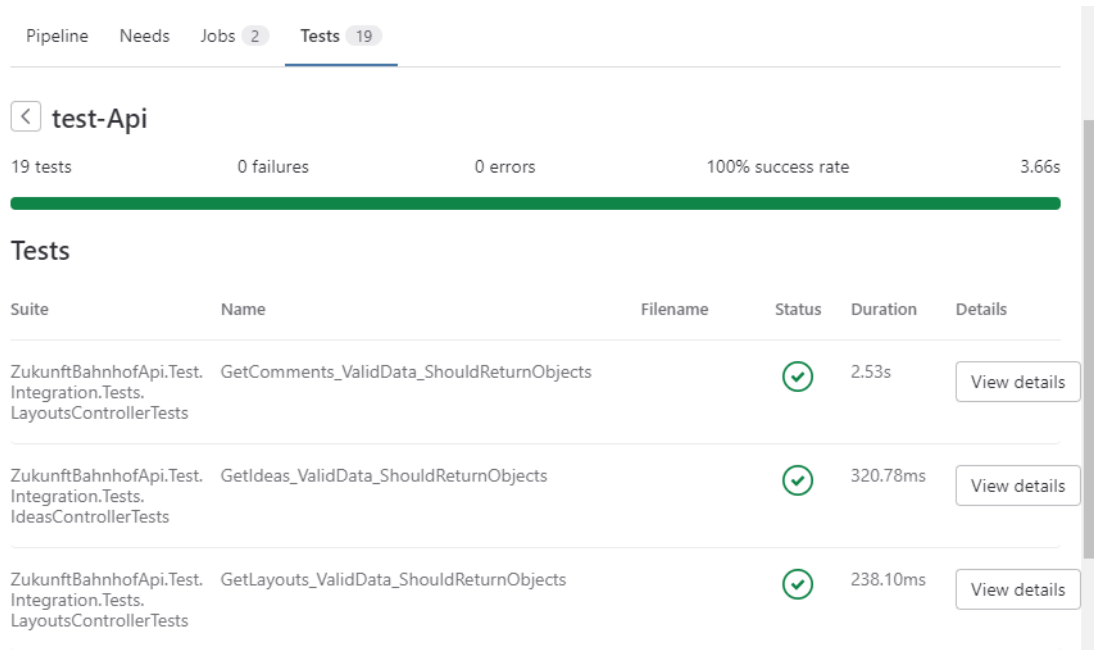


Abbildung 9.3: Testergebnisse in Gitlab Pipeline (Quelle: Eigene Darstellung)

9.4 Unity

Für die Darstellung des 3D-Modells wird Unity eingesetzt, was danach als [WebGL](#) kompiliert wird. Das Unity Projekt wird in sogenannten Assets organisiert. Nachfolgend sind die wichtigsten Assets aufgelistet:

- FBX-Dateien: 3D-Modell und alle abgebildete Gebäude von Lichtensteig, bzw. alles was mit Blender erstellt ist.
- Prefabs: Objekte, die bei einer Arealgestaltung platziert werden können.
- Materials: Das Aussehen eines Objektes in Unity, z.B. die Farbe der Gebäudefassaden.
- Jslib: Das [WebGL](#) Plugin was für die Kommunikation zwischen Unity und React benötigt wird.
- Scripts: C# Code was für die gesamte Logik im Unity-Projekt zuständig ist.

9.4.1 Controllers

Im Controller Verzeichnis der Scripts ist die gesamte Steuerung der Kamera untergebracht sowie die Logik für das Auswählen von Objekten. Die Kameraführung unterscheidet zwischen Maus und Touchscreen Eingabe. Wird ein Finger oder der Mauszeiger

erkannt, wird die Sicht der Kamera entsprechend der Bewegung um das Areal herumgeschwenkt. Sofern ein gewisser Schwellwert nicht über bzw. unterschritten wird, ansonsten stoppt die Bewegung um nicht unter den Grund sehen zu können. Bei mehr als einem erkannten Finger wird davon ausgegangen, dass versucht wird heran oder hinaus zu zoomen. Um das sogenannte **Pinch-Zoom** zu erkennen, wird die Position am Anfang der Bewegung gespeichert. Anschliessend wird der Betrag der verändernden Distanz zwischen den zwei Fingern errechnet. Die Ansicht wird um diesen Betrag vergrössert oder verkleinert, je nachdem ob der Betrag positiv oder negativ ist.

Ein weiterer Controller ist für das Selektieren und Deselektieren von anklickbaren Objekten zuständig. Bei diesem Controller muss beachtet werden, dass Objekte nur dann anklickbar sind, wenn sich der Finger oder die Maus nicht bewegen. Ansonsten kann es vorkommen, dass ein Objekt selektiert wird, auch wenn eigentlich nur die Ansicht geändert werden will.

9.4.2 Datenpersistenz

Beim Aufrufen der offiziellen Arealgestaltung wird nur das Unity Projekt geladen. Erst wenn ein von Besuchern gestaltetem Areal aufgerufen wird, wird über der React-Unity-Webgl Library die Url und die Id der gewünschten Gestaltung an Unity mitgeteilt. Mit diesen Informationen lädt Unity die gesetzten Objekte vom Backend nach und stellt sie auf der Karte dar.

9.5 React-Unity-Webgl

Die Library React-Unity-WebGL stellt eine Kommunikation zwischen **Unity** und React her. Durch den Einsatz dieser Bibliothek ist es möglich, Methoden jeweils vom einen zum anderen System aufzurufen und Daten hin und her zu versenden. Der Methoden-Aufruf ist jedoch etwas limitiert. Es ist lediglich möglich, eine Methode mit höchstens einem Parameter aufzurufen. Dieser eine Parameter ist zudem die einzige Möglichkeit, Daten zu transferieren. Ausserdem muss dieser Parameter entweder eine Zahl oder ein String sein. Aus diesem Grund werden die Objekte, die von React zu **Unity** und umgekehrt übergeben werden, als Json konvertiert und anschliessend wieder zurück zu einem Objekt umgewandelt werden.

Auf der React-Seite müssen Event Listener erstellt werden, die auf der Kommunikation von **Unity** lauschen, um entsprechend Aktionen ausführen zu können. Wenn ein Objekt platziert wird, muss React dies erfassen und daraufhin das Formular zum Ausfüllen des Namens des Objekts sowie einer Beschreibung anzeigen.

9.6 Continuous Integration und Continuous Delivery

Die CI/CD Pipeline ist in vier Stages aufgeteilt, von welchen die beiden Stages 'prepare' und 'test' bei jedem Push ins Repository durchlaufen werden. Die Stage 'prepare' lizenziert das Unity-Projekt und bestimmt die Unity-Version, was später zum Kompilieren des Unity-Projekts in der Pipeline benötigt wird.

Name	Zweck	Ausführung
prepare	Vorbereitende Schritte, wie Lizenzierung des Unity Projekts und Bestimmen der Unity Version.	Immer
test	Backend Integration Tests ausführen.	Immer
build	Unity Projekt als WebGL builden und als Artifact zur Verfügung stellen.	Merge into master
deploy	Applikation auf den Server ausliefern.	Merge into master

Tabelle 9.5: CI/CD Stages

Nachfolgend sind die drei wichtigsten Stages genauer erklärt.

9.6.1 Test

Die Integrationstests vom Backend werden ebenfalls in der Pipeline ausgeführt. Da die CI/CD Pipeline in Docker läuft, laufen die Tests in der Pipeline im sogenannten Docker in Docker Modus. Als Artifact steht anschliessend ein Testcoverage xml zur Verfügung. Dieser wird in der Pipeline im Reiter Tests entsprechend grafisch dargestellt. Solange das Artifact vorhanden ist, werden alle Tests mit dem entsprechenden Status angezeigt.

9.6.2 Build

In der Build Stage wird das gesamte Unity Projekt für WebGL kompiliert. Die Pipeline ist so aufgebaut, dass auf Wunsch das Projekt auch für eine andere Plattform gebaut werden kann. Der Build steht nach dem Kompilieren als Artifact in Gitlab zur Verfügung. Dieser wird beim Deploy Prozess dann wieder verwendet. Die Stages Build und Deploy werden nur nach dem Mergen im Master ausgeführt.

9.6.3 Deploy

Dieser Abschnitt erläutert den gesamten Deploy-Prozess, der von der Git Pipeline aus getriggert wird. Unter Vorbereitungen ist aufgelistet, was alles auf dem Server eingerichtet sein muss, damit eine erfolgreiche Verteilung möglich ist.

Vorbereitungen

Um auf dem Ost-Server deployen zu können, müssen gewisse Vorbereitungen auf dem Server getroffen werden. So muss SSH aktiviert und lauffähig sein. Ausserdem muss der SSH-Port auf 40010 geändert werden, da nur die Ports 80,443 und von 40000 bis 40010 offen sind. Auf dem Server muss Docker installiert sein, dies kann mit 'sudo apt install docker' erledigt werden. Zusätzlich ist auch die Installation von npm notwendig ('sudo apt install npm'). Zu Beginn muss ein Verzeichnis namens '/zb' auf dem Server erstellt werden. In diesem Verzeichnis wird die Anwendung schliesslich heruntergeladen, um im

Docker-Container ausgeführt werden zu können. Um ein reibungsloses Deployment zu ermöglichen, ist das vorgängige Clonen des Repositorys im entsprechenden Ordner notwendig. Die `deploy.sh` Datei muss ebenfalls ins selbe Verzeichnis kopieren werden. Im Projektverzeichnis unter `ZukunftBahnhofClient` muss ausserdem einmalig `'npm install'` ausgeführt werden. Dies installiert alle nötige NodeJs Module, die später dann im entsprechend Docker Container zur Verfügung gestellt werden.

Pipeline

Die `'deploy.sh'` Datei sorgt dafür, dass mögliche Änderungen, die direkt auf dem Server vorgenommen wurden zurückgesetzt werden und der aktuelle Stand vom Master deployed wird. Ausserdem wird mit `'docker system prune -f'` dafür gesorgt, dass nicht zu viel Speicherplatz verwendet wird. Jedes Mal beim Deployen werden die Docker Container von neuem gebildet. Dies kann schnell zu Speicherplatzmangel führen.

Als Vorbereitung in der Pipeline wird der `ssh-agent` aktualisiert, `sshpass`, `rsync` installiert, den SSH Port entsprechend angepasst und mit `ssh-keyscan` die `'known_host'` Datei mit dem Public-Key vom Ost-Server ergänzt.

SSHpass wird verwendet, um eine SSH Verbindung mit Eingabe des Passworts zu ermöglichen. So werden alle Befehle mithilfe von `sshpass` auf dem Deploymentserver ausgeführt. `Rsync` wird benötigt, um die gebildeten Dateien aus dem Unity Build auf dem Server zu kopieren. Weiter werden Umgebungsvariablen bzw. die `client.env` und `database.env` Dateien von der Pipeline aus auf dem Server abgefüllt. Um dies dynamisch anpassbar zu gestalten, werden dafür CI/CD Variablen in Gitlab benutzt. Folgende Variablen müssen definiert sein:

Variable	Beschreibung
HTTP_HTTPS	Definiert ob die Applikation über http oder https zur Verfügung gestellt werden soll.
POSTGRES_DB	Der Datenbankname der Postgres Datenbank.
POSTGRES_PWD	Das Passwort Postgres Datenbank.
POSTGRES_USER	Den User Postgres Datenbank.
SSH_PORT	Port über welchem die SSH Verbindung aufgebaut werden kann.
UNITY_LICENSE	Unity Lizenz, die auf der Unity Seite erstellt werden muss.
VM_HOST	Hostname des Servers auf dem die Applikation deployed werden soll.
VM_PWD	Password vom Server.
VM_SSH_PUB_KEY	Den Public Key vom Server der in der Datei <code>'known_host'</code> geschrieben wird.
VM_USER	User zum einloggen auf dem Server.

Tabelle 9.7: CI/CD Variablen

Wenn alle Daten auf dem Server korrekt abgefüllt sind, wird geprüft, ob ein Zertifikat von Let's Encrypt vorhanden ist. Die Zertifikatsdatei muss unter folgendem Verzeichnis abgelegt sein: `‘/zb/prototype-zukunft.bahnhof-lichtensteig/ZukunftBahnhofWeb/certbot/conf/live/srbsci-155.ost.ch/fullchain.pem’`. Trifft dies nicht zu, muss erst noch ein Zertifikat bei Let's Encrypt beantragt werden (siehe [Abbildung 9.4](#)).

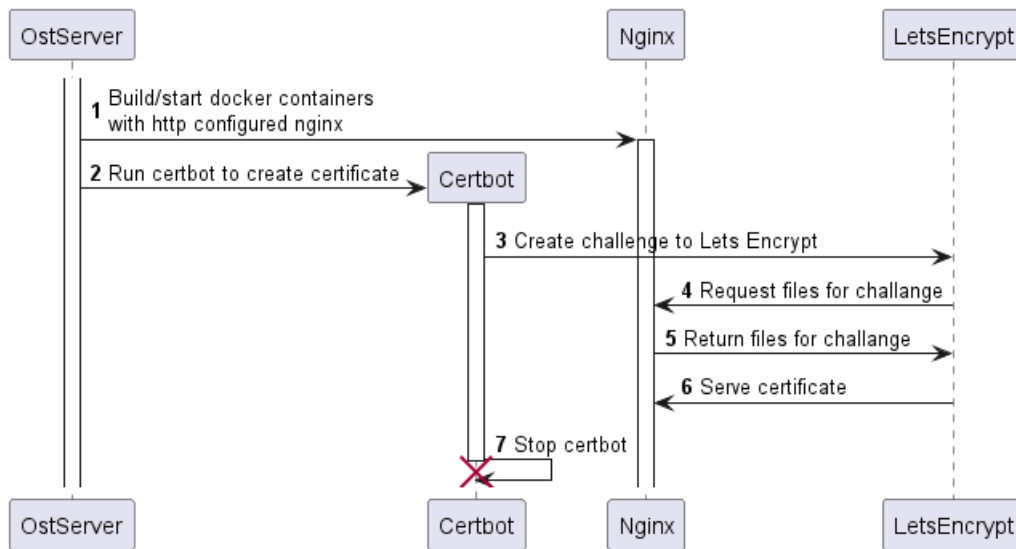


Abbildung 9.4: Zertifikat Erhaltungsablauf mithilfe von certbot (Quelle: Eigene Darstellung)

Für die Beantragung eines Zertifikats bei Let's Encrypt wird Certbot verwendet. Damit Certbot ein Zertifikat beantragen kann, muss erst ein Webserver gestartet werden. Dies wird mit den nginx Docker Container erledigt. Dieser wird so gestartet und konfiguriert, dass er in der Route `‘/.well-known/acme-challenge/’` das Volumen `‘/ZukunftBahnhofWeb/certbot/www’` bzw. innerhalb des Docker Containers `‘/var/www/certbot’` publiziert wird. Certbot erstellt dann eine Challenge an Let's Encrypt. Let's Encrypt wiederum greift auf die Route `‘/.well-known/acme-challenge/’` zu und erstellt anschließend ein Zertifikat, was Certbot dann wieder in einem gemeinsamen Volume für nginx bereitstellt. Anschliessend können alle Docker Container (mit Ausnahme von Certbot, der wird heruntergefahren) neu gestartet werden. Dafür wird eine zusätzliche `docker-compose.deploy.yml` Datei benutzt, die nginx so umkonfiguriert das alle Routen auf https umgeleitet werden.

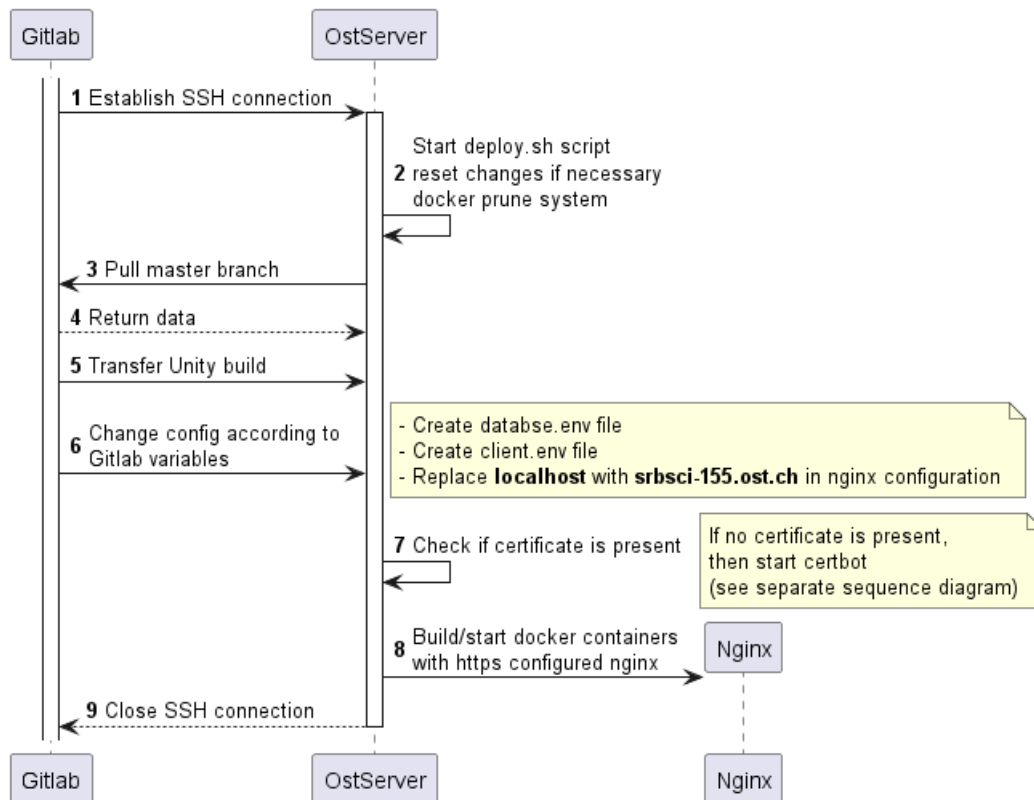


Abbildung 9.5: Deploymentablauf (Quelle: Eigene Darstellung)

In der [Abbildung 9.5](#) ist der Deployment Prozess dargestellt. Wobei nur der Nginx Container abgebildet ist. Insgesamt werden folgende Docker Container gestartet:

- database: postgres Datenbank
- api: ASP .Net Core, das Backend
- client: NodeJs stellt die React Applikation zur Verfügung
- proxy: nginx Webserver, fungiert als Reverse Proxy und handhabt das Zertifikat.

10. Usability Test

In diesem Kapitel wird zunächst der Ablauf des durchgeführten Usability-Tests beschrieben. Dabei werden die einzelnen Schritte erläutert, die bei der Durchführung des Tests angewendet werden. Anschliessend werden die Ergebnisse des Usability-Tests präsentiert. Die gesammelten Daten und Feedbacks der Testpersonen werden analysiert und ausgewertet, um Erkenntnisse über die Benutzerfreundlichkeit der Anwendung zu gewinnen. Die detaillierte Darstellung der Testergebnisse bietet Einblicke in Stärken und Schwächen der Anwendung aus Benutzersicht.

10.1 Ablauf

Der Usability Test wurde in Lichtensteig am 02.06.2023 auf dem Areal bzw. im Restaurant des Chössi Theaters durchgeführt. Zwei Personen von den ‘Summer of Pioniers’ haben sich bereit erklärt am Usability Test teilzunehmen. Die Teilnehmenden wurden gebeten 10 verschiedene Aufgabe nacheinander abzuarbeiten. Dies sollte möglichst ohne Hilfe geschehen. Dadurch lässt sich gut erkennen, ob der Prototyp benutzerfreundlich und intuitiv zu bedienen ist. Als Erstes soll ein neues Layout (nachträglich in Arealgestaltung umbenannt) erstellt werden und in sich in diesem umsehen. Anschliessen soll über das Chössi Theater Informationen eingeholt werden und ein Kommentar hinterlegen. Danach soll in der soeben erstellte Arealgestaltung mindestens ein Objekt der Wahl erstellt, benannt und beschrieben werden. Ausserdem sollen andere Gestaltungen angeschaut kommentiert und bewertet werden. All diese Aufgaben sind im Anhang C.3 detailliert beschrieben mit der jeweiligen Bewertung und Bemerkung der testenden Personen.

10.2 Ergebnisse

Der Usability Test ist mehrheitlich positiv ausgefallen. Das Feedback der Testpersonen ist sehr hilfreich und ausführlich, was für die Optimierung und Erweiterung des Prototypen äusserst wertvoll ist. Nachfolgend sind die wichtigsten Erkenntnisse/Verbesserungsvorschläge aufgelistet.

- Auf der Startseite der Applikation fehlen Informationen über das Projekt [ZB](#).
- Die Startseite ist generell nicht sehr ansprechend, da lediglich eine Liste von Layouts gezeigt wird (siehe [Abbildung 10.1](#)). Auch der Name Layout ist ungünstig gewählt.
- Beim Anzeigen von Informationen eines Gebäudes ist der Name zu wenig präsent dargestellt.
- Um das Areal zu gestalten, könnten bessere Möglichkeiten geboten werden, um freier in der Gestaltung zu sein. So ist es beispielsweise sehr mühselig einen Wald zu modellieren, da dafür etliche Objekte (Bäume und/oder Tannen) gesetzt und benannt werden müssen. Hier wäre es wünschenswert eine Möglichkeit zu bieten mehrere identische Objekte zusammen zu platzieren, oder zumindest die Benennung der Objekte als optionales Feld zu pflegen.

10.2. ERGEBNISSE

- Die Icons fürs Kommentieren und Liken dürften grösser ausfallen.
- Einmal platzierte und abgespeicherte Objekte können nicht mehr editiert oder gelöscht werden.
- Bei der Arealgestaltungsübersicht wäre eine Vorschau des Areals in Form eines Thumbnails wünschenswert.
- Erweiterungsvorschlag: Eine geführte Tour durch die Gebäude (z.B. Chössli Theater und Neubau).
- Anklickbare Objekte dürften besser hervorgehoben werden.

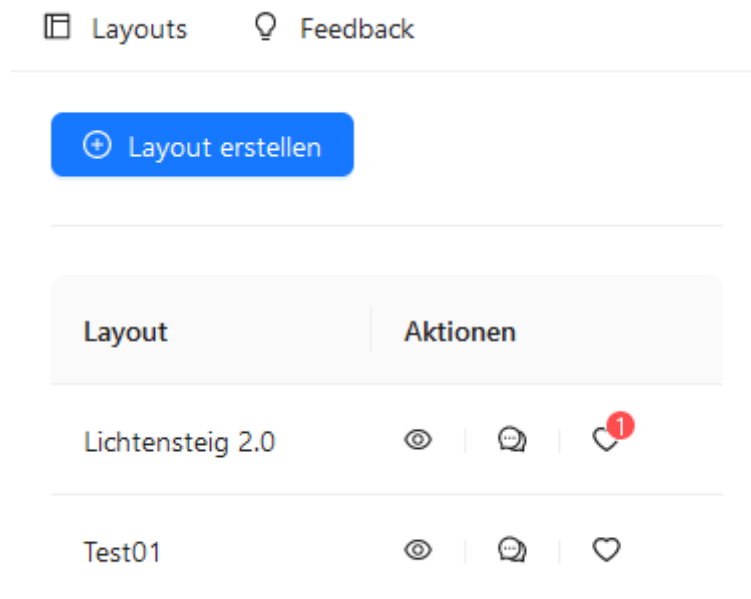


Abbildung 10.1: Layouts - Alte Startseite der Applikation (Quelle: Eigene Darstellung)

11. Ergebnisse

In diesem Kapitel wird das Endergebnis der Implementation analysiert. Es wird überprüft, welche Anforderungen umgesetzt wurden und anhand von Screenshots aufgezeigt, wie der Prototyp aussieht.

11.1 Umgesetzte Anforderungen

Nachfolgend wird auf alle funktionalen und nicht-funktionalen Anforderungen eingegangen, um aufzuzeigen, was bei der Implementierung des Prototypen alles umgesetzt wurde.

11.1.1 Funktionale Anforderungen

Alle zwingenden funktionalen Anforderungen konnten erfolgreich umgesetzt werden. Da die Implementierung der Anforderungen besser voranschritt als vorerst angenommen, konnte zudem ein Grossteil der optionalen Anforderungen ebenfalls im Prototypen umgesetzt werden.

Gerade die Anforderung [FR11](#), welche das Platzieren von Objekten beschreibt, wertet den Prototypen zusätzlich auf und zeigt auf, wo das Potenzial einer solchen Applikation liegt.

Anforderung	Beschreibung	Umgesetzt
FR1	VR-Abbild aus Vogelperspektive betrachten	✓
FR2	Kameraperspektive um 360° schwenken	✓
FR3	Weiter Informationen anzeigen	✓
FR4	Feedback zum Projekt	✓
FR5	Objekte bewerten	✓
FR6	Objekte kommentieren	✓
Optionale Anforderungen		
FR7	Zwischen Tageszeiten umschalten	✗
FR8	Zoomen	✓
FR9	Kameraperspektive komplett frei wählen	✗
FR10	Eigene Arealgestaltung	✓
FR11	Vordefinierte Objekte in einer individuellen Arealgestaltung platzieren können	✓
FR12	Arealgestaltung persistieren	✓
FR13	Andere Arealgestaltung betrachten	✓
FR14	Arealgestaltungen bewerten	✓
FR15*	Feedback bewerten	!
FR16*	Feedback kommentieren	!

Tabelle 11.1: Stand der funktionalen Anforderungen

* Anforderungen wurden bewusst nicht umgesetzt, da das Feedback nicht öffentlich sein soll.

11.1.2 Nicht-funktionale Anforderungen

Auch alle nicht-funktionalen Anforderungen konnten in der prototypischen Implementierung berücksichtigt werden. Für die Umsetzung der optionalen Anforderungen in diesem Bereich reichte die vorhandene Zeit nicht aus.

Anforderung	Beschreibung	Umgesetzt
NFR1	Funktionsfähig in aktuelle Webbrowsern	✓
NFR4	Einfache Bedienung	✓
NFR6	Einfaches Anzeigen von Informationen	✓
NFR7	Testing im Backend	✓
NFR8	Kurze Ladezeit	✓
NFR9	Gute Performance	✓
NFR10	Einfache Wartung	✓
NFR11	Einsatz von weitverbreitete und gut unterhaltene Frameworks	✓
NFR12	Grösstenteils Open Source Einsatz	✓
Optionale Anforderungen		
NFR2	VR-Brillen Support	✗
NFR3	Fehler im Backend in einem Logfile loggen	✗
NFR5	Einfache Bedienung für VR-Headset	✗

Tabelle 11.3: Stand der nicht-funktionalen Anforderungen

11.2 Zielerreichung der Arbeit

Das Resultat der Elaborations-Phase sind fünf verschiedene Realisierungsansätze die das Projekt **ZB** auf verschiedene Arten digital erlebbar machen. Mithilfe einer Nutzwertanalyse und eines Workshops ist **RA1** als geeignetster Realisierungsansatz ausgewählt. Durch sehr gute Möglichkeiten in Bereich Wissenstransfer und Partizipation und einem starken Präsenzerlebnis, sticht **RA1** im Gegensatz zu den anderen Ansätzen heraus. Das Konzept für den Showroom sieht vor, verschiedene Ansätze im Raum zu vereinen. Diese Arbeit konzentrierte sich auf die prototypische Umsetzung einer VR City Map (**RA1**). Der Prototyp ist lauffähig und kann auf <https://srbsci-155.ost.ch> besucht werden. Somit ist eine Installation in einem Showroom z.B. mithilfe eines grossen Monitors möglich. Durch Usability Testing direkt vorort auf dem Bahnhofsareal in Lichtensteig, konnte der Prototyp durch potenzielle Benutzende ausführlich getestet werden. Dadurch konnten viele Verbesserungs- und Weiterentwicklungsmöglichkeiten erfasst werden, von welchen bereits einige im Prototypen umgesetzt wurden.

11.3 Vorstellung des Prototyps

In diesem Abschnitt wird auf den Prototypen eingegangen und aufgezeigt was alles möglich ist. Nachfolgend werden die wichtigsten Funktionen der Applikation anhand Screenshots dargestellt. Weitere Bilder des Prototypen sind im [Anhang C.1](#) zu finden.

11.3.1 Start der Applikation

In diesem Abschnitt wird auf den Start der Applikation eingegangen. Als Einstiegspunkt der Applikation wird das Projekt **ZB** kurz erklärt ([Abbildung 11.1](#)).



Abbildung 11.1: Start der Applikation (Quelle: Eigene Darstellung)

Am unteren Ende der Startansicht befindet sich eine Schaltfläche mit der die VR-Welt geladen werden kann. Die offizielle Arealgestaltung kann nicht verändert werden. Beim ersten Aufruf wird zudem eine Tour angezeigt, die die Benutzenden Schritt für Schritt durch die Bedienung der VR-Welt hindurchführt. Die Bedienung ist auf Maus, Tastatur und Touchscreen ausgelegt. In der Tour wird mit Symbolbildern gearbeitet, die aufzeigen, mit welchen Aktionen was möglich ist. Falls eine Schaltfläche auf der Seite angeklickt werden soll, wird diese hervorgehoben, wie hier in Schritt 4 der Tour ersichtlich. Die Sicht auf das Areal kann um 360° geschwenkt werden. Wenn ein Gebäude näher betrachtet werden möchte, kann dies mit dem Mauseisrad oder bei einem Gerät mit Touchscreen mit [Pinch-Zoom](#) erreicht werden.



Abbildung 11.2: Tour Schritt 4 (Quelle: Eigene Darstellung)

11.3.2 Eigene Gestaltung

Um bei der Gestaltung von Lichtensteig mitzuwirken, kann eine eigene Arealgestaltung erstellt werden oder eine bereits bestehende Gestaltung erweitert werden. Unter dem Menüpunkt 'Gestaltungen' werden alle bereits erfassten Arealgestaltungen angezeigt. Beim Tippen/Klicken auf dem Auge in der Liste ([Abbildung 11.3](#)) kann die gewünschte VR-Welt gewählt werden oder eine neue mit dem dafür vorgesehene Button erstellt werden.

Ist die gewünschte Gestaltung geladen, können Objekte beliebig in dem abgebildeten Ausschnitt von Lichtensteig platziert werden. Nach der Auswahl eines vordefinierten Objektes (Haus, Tanne oder Baum) wird in den sogenannten Edit Modus gewechselt. In diesem Modus, das mit einem gelb/orangen Balken gekennzeichnet ist, ist die Kameraführung deaktiviert und das gewählte Objekt kann auf der Karte platziert werden. Weil das Platzieren des Objektes ebenfalls mit der linken Maustaste erfolgt, wird die Kameraführung deaktiviert.

Mit der rechten Maustaste kann das Objekt im Uhrzeigersinn um 45° gedreht werden. Ist das Objekt platziert, muss noch ein Name und auf Wunsch eine Beschreibung des Objektes hinterlegt werden. Falls fälschlicherweise das Objekt an einem ungewünschten Ort platziert wurde, kann mit der Hilfe des Buttons 'Umplatzieren' das Objekt erneut an einem anderen Ort platziert werden. (siehe [Abbildung 11.4](#))

11.3. VORSTELLUNG DES PROTOTYPS

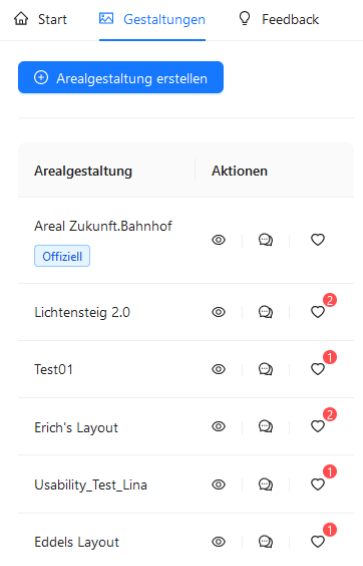


Abbildung 11.3: Arealgestaltungsübersicht (Quelle: Eigene Darstellung)

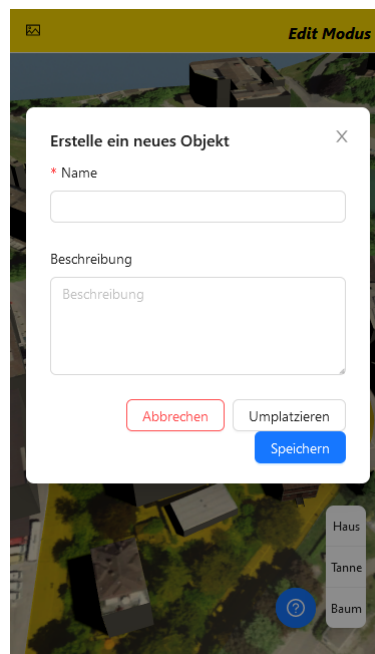


Abbildung 11.4: Neues Objekt erstellen (Quelle: Eigene Darstellung)

11.3.3 Kommentieren und bewerten

Die Arealgestaltungen sowie die einzelnen Objekte der jeweiligen Gestaltungen können kommentiert und bewertet werden. Da im Prototypen keine Benutzerverwaltung umgesetzt wurde, kann ohne Einschränkungen kommentiert und bewertet werden.

Die Kommentar- und Bewertungsfunktion ist für die einzelnen Objekte sowie auch für das gesamte Areal identisch aufgebaut. Neben den von Benutzenden erstellten Gebäude

11.3. VORSTELLUNG DES PROTOTYPUS

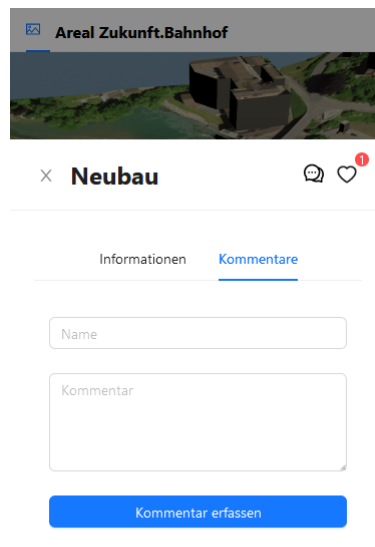


Abbildung 11.5: Kommentare zu einem Gebäude (Quelle: Eigene Darstellung)

können auch das Chössli Theater und der Neubau bewertet und kommentiert werden. Diese beiden Gebäude sind in allen Arealgestaltungen vorhanden. Die Bewertungen und Kommentare der beiden Gebäude werden ebenfalls in allen Gestaltungen angezeigt. Beim Bewerten oder Kommentieren eines Objektes werden zusätzlich noch unter dem Reiter Informationen, Name und Beschreibung des Objektes angezeigt.

11.3.4 Feedbackfunktion

Um ein allgemeines Feedback zum Projekt abzugeben, wird die extra dafür vorgesehene Funktion benützt. Dieses Feedback wird allerdings nur in der Datenbank gespeichert und ist somit für die Benutzenden nicht einsehbar. Das Feedback soll auch nur für die Stiftung [ZB](#) verfügbar sein.

The screenshot shows a web interface for a feedback form. At the top, there is a navigation bar with three items: 'Start' (with a house icon), 'Gestaltungen' (with a grid icon), and 'Feedback' (with a speech bubble icon and a blue underline). Below the navigation bar is the title 'Feedback' in a large, bold font. Underneath the title is a paragraph of text: 'Hier können Feedback, Anregungen und Ideen zum Projekt zukunfts.bahnhof eingereicht werden. Das Projektteam freut sich über jegliche Inputs und nimmt ihre Anliegen gerne auf. Auch allgemeines Feedback zum Projekt nehmen wir gerne entgegen.' Below this text is a light blue box containing a blue circle with the number '1' and the text 'Nur das Projektteam hat Einsicht in die erfassten Nachrichten.' Below this box are two input fields: one for 'Name' and one for 'Feedback'. At the bottom of the form is a blue button with the text 'Feedback erfassen'.

Abbildung 11.6: Feedback Seite (Quelle: Eigene Darstellung)

11.4 Ausblick und Erweiterungsmöglichkeiten

In diesem Abschnitt wird, basierend auf den Erkenntnissen aus dem Usability Test, ein Ausblick auf die mögliche Weiterentwicklung des Prototyps gegeben. Darüber hinaus werden bereits umgesetzte Verbesserungsvorschläge aufgeführt.

Informationen zum Projekt **Bereits umgesetzt** ✓

Die Startseite soll eine kurze Erklärung des Projektes **ZB** beinhalten. Somit werden Besuchende informiert, um was es sich bei diesem Projekt handelt und können anschliessend das Areal betrachten. Ausserdem soll die Benutzerführung verbessert werden indem nicht lediglich eine Liste von 'Layouts' als Startseite präsentiert wird (siehe [Abbildung 10.1](#)). Die Namensgebung ist ebenfalls verbesserungswürdig. Unter 'Layout' versteht man nicht automatisch, dass es sich um eine Gestaltung des Areals rund um Lichtensteig handelt. Aus diesem Grund wäre 'Arealgestaltung' ansprechender und selbsterklärender als 'Layout'.

Objekt-Informationen besser darstellen **Bereits umgesetzt** ✓

Wenn ein Objekt angeklickt wird, sollten die Informationen dazu besser dargestellt werden. So soll der Name des Objekts prägnanter sein. Dies erleichtert für die Benutzenden die Erkenntnis, welches Gebäude nun ausgewählt wurde.

Mehr Möglichkeiten ein Areal zu gestalten

Eine Erweiterung könnte es ermöglichen, Objekte mehrfach unter demselben Namen zu platzieren. Sodass ein Wald erstellt werden kann, ohne jeden Baum einzeln benennen zu müssen. Zudem könnten noch mehr platzierbare Objekte hinzukommen, so dass die Wahl nicht nur zwischen Haus, Tanne und Baum besteht.

Grössere Kommentier- und Bewertungsicons **Bereits umgesetzt** ✓

Die Icons zum Kommentieren und Liken bei der Auswahl eines Objekts dürfen prägnanter sein. Vor allem auf grösseren Monitoren erscheinen die Schaltflächen sehr klein und

können daher übersehen werden. Auf Mobilien Geräten kann es je nach Auflösung und Schriftgrösse schwierig sein die Icons richtig zu mit dem Finger zu treffen.

Editieren und Löschen von Objekten

Falls ein Objekt platziert und abgespeichert ist, kann es nicht wieder verändert werden. Weder der Beschreibungstext, noch der Name kann nachträglich geändert werden. Auch die Position und Rotation kann nicht geändert werden. Auch das Löschen von platzierten Objekten ist nicht möglich. Hier würde es sich anbieten, ein zusätzliches Icon bei der Auswahl eines Objektes darzustellen, womit ein Editieren der Daten sowie Umplatzieren ermöglicht wird.

Eine andere Möglichkeit wäre, dass nicht direkt nach dem Abspeichern des Objektes der 'Edit Modus' verlassen wird, sondern das dies über eine separate Schaltfläche geschieht. So könnten die Objekte in diesem Modus noch umplatziert, gelöscht oder umbenannt werden.

Vorschau aller Arealgestaltungen

Es wäre wünschenswert, wenn in der Liste der Arealgestaltungen jedes Element mit einer kleinen Übersichtsbild ausgestattet wäre. So müsste nicht jede Gestaltung geöffnet werden um zu sehen wie sie in groben Zügen aussieht. Dies könnte zumindest für die Mausbedienung mit einem Hover-Effekt umgesetzt werden. Oder man ergänzt die Liste mit einer kleinen Grafik die das Areal zeigt.

Geführte Tour

Eine weitere mögliche Erweiterung ist eine geführte Tour durch die wichtigsten Gebäude des Areals. Dies könnte auf verschiedene Arten umgesetzt werden. Eine Möglichkeit wäre eine geführte Kamerabewegung von einem Objekt zum nächsten, sodass die Benutzende Person nur 'Weiter' oder 'Zurück' auswählen muss.

Bessere Markierung von auswählbaren Objekten

Objekte die anklickbar sind, könnten mit ihrem Namen oder einem Icon gekennzeichnet werden. Eine andere Möglichkeit wäre diese Objekte durch leuchtende Konturen besser erkennbar zu machen.

Benutzerverwaltungssystem

Eine sinnvolle Erweiterung ist ein Login oder Benutzerverwaltungssystem. Somit könnte dafür gesorgt werden, dass nur angemeldete Personen ihre eigene Gestaltung editieren können. Ausserdem könnte so auch ein Berechtigungssystem eingeführt werden, welches eine Möglichkeit bieten würde, abgegebenes Feedback selber direkt in der Benutzeroberfläche einzusehen. Die Applikation könnte so umgebaut werden, das weitere Informationen zu den vordefinierten Gebäuden (Chössli Theater und Neubau) ebenfalls über die Benutzeroberfläche ergänzen und geändert werden können.

11.5 Operationelle Hinweise

Dieser Abschnitt dient der Weiterbetreuung des Prototypen. Die Datenbank wird mit Port 5439 von Docker Compose zur Verfügung gestellt. Auf dem OST Server ist dieser Port aus Sicherheitsgründen jedoch nicht von aussen erreichbar.

11.5.1 Platzierbare Objekte hinzufügen

Um die Liste der verfügbaren Objekte für die Platzierung zu erweitern, müssen die folgenden Schritte durchgeführt werden.

1. Eintrag in der Datenbanktabelle *Prefabs* erstellen. Z.B. (4, 'Sitzbank')
2. Erstellen eines Prefabs in Unity (Dateiendung .prefab).
3. Ablegen des Prefabs im Unity Projekt unter 'Assets/Resources/Prefabs'.
4. Der Name des Prefabs muss mit der Id in der Datenbank übereinstimmen. Z.B. '4.prefab'
5. Dem Prefab muss ein Collider (z.B. Mesh Collider) hinzugefügt werden, sofern noch keiner vorhanden ist. Dieser wird benötigt, damit das Objekt später selektiert werden kann.
6. Dem Prefab muss der Tag 'Selectable' zugewiesen werden.
7. Dem Prefab das Unity Skript 'CustomObjectProperties' zugewiesen werden. Zudem muss sichergestellt werden, dass die Eigenschaft 'Is Custom Component' angewählt ist (siehe [Abbildung 11.7](#)).
8. Zudem muss im Game Object 'BuildingManager' (zu finden in der Hierachy unter Managers) das neu erstelle Prefab hinzugefügt werden (siehe [Abbildung 11.8](#)).

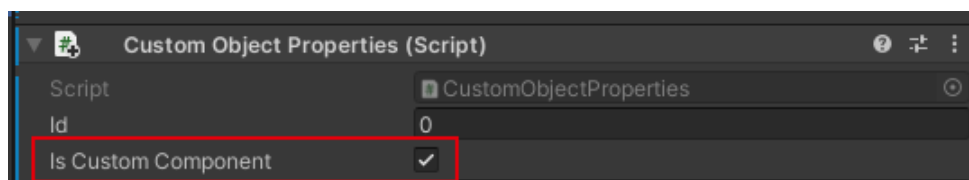


Abbildung 11.7: Zuordnung CustomObjectProperties zu neuem Prefab (Quelle: Eigene Darstellung)

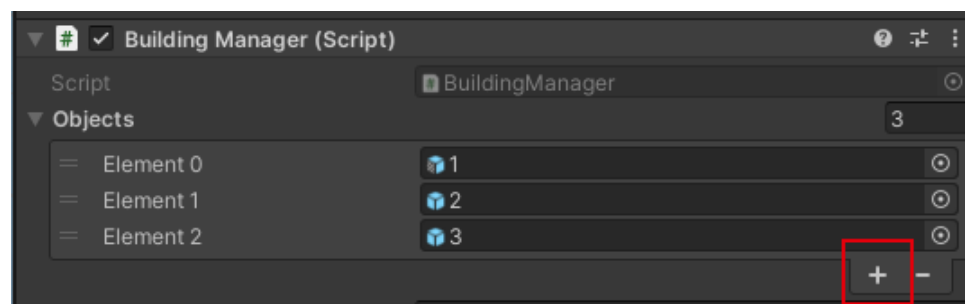


Abbildung 11.8: Prefab in BuildingManager hinzufügen (Quelle: Eigene Darstellung)

11.5.2 Vordefinierte Objekte hinzufügen

Vordefinierte Objekte, wie z.B. das Chössli Theater, müssen in Unity erstellt und am gewünschten Ort auf der City Map platziert werden. Anschliessend sind folgende Schritte notwendig.

1. Eintrag in der Datenbanktabelle *Description* erfassen. Im Attribut *Text* können HTML-Elemente verwendet werden, um den Beschreibungstext zu gestalten.
2. Eintrag in der Datenbanktabelle *ObjectComponents* mit **negativer** *Id* erfassen. Da es bereits zwei vordefinierte Objekte auf der City Map gibt, müsste ein neues Objekt mit der *Id* -3 erfasst werden.
 - *Id* = negative *Id*
 - *Name* = Name des Gebäudes
 - *PositionX* = Positionswert X-Achse
 - *PositionY* = Positionswert Y-Achse
 - *PositionZ* = Positionswert Z-Achse
 - *RotationX* = Rotationswert X-Achse
 - *RotationY* = Rotationswert Y-Achse
 - *RotationZ* = Rotationswert Z-Achse
 - *IsCustomComponent* = **false**
 - *DescriptionId* = *Id* der vorab erfassten Beschreibung
 - *LayoutId* = null
 - *PrefabId* = null
3. Sicherstellen, dass dem Objekt ein Collider hinzugefügt ist.
4. In Unity muss das Skript 'CustomObjectProperties' dem Objekt zugewiesen werden. Zudem muss sichergestellt werden, dass 'Is CustomComponent' **nicht** angekreuzt ist (siehe [Abbildung 11.9](#)).
5. Die *Id* im 'CustomObjectProperties' muss der *Id* des Objektes in der Datenbank übereinstimmen.

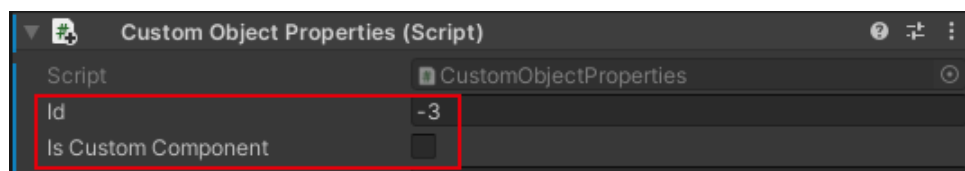


Abbildung 11.9: Zuordnung CustomObjectProperties zu neuem Objekt (Quelle: Eigene Darstellung)

Um die Beschreibungen der bereits bestehenden Objekte (Chössli Theater und Neubau) zu bearbeiten, müssen nur die Einträge in der Tabelle *Description* der entsprechenden Objekte angepasst werden.

A. Verzeichnisse

Abbildungsverzeichnis

1.1	Gewähltes Vorgehensmodell (Quelle: Eigene Darstellung)	1
1.2	Impressionen des Prototyps (Quelle: Eigene Darstellung)	2
2.1	Konzept Projekt ZB (Quelle: zukunft.bahnhof)	3
2.2	Areal ZB in Lichtensteig (Quelle: zukunft.bahnhof)	4
2.3	Gewähltes Vorgehensmodell (Quelle: Eigene Darstellung)	6
3.1	Resultat des Brainstormings (Quelle: Eigene Darstellung)	10
3.2	Beispiel einer VR Stadt (Quelle: https://www.jenamdvacet.cz/)	11
3.3	Ikea AR App le: https://www.ikea.com/ch/de/customer-service/mobile-apps/)	13
3.4	Screenshot des Spiels "Welten der Werkstoffe" (Quelle: https://welt-der-werkstoffe.de/game.html)	15
5.1	Beispiel Showrooms Innenansicht (Quelle: Eigene Darstellung)	28
5.2	Beispiel Showroom Aussenansicht le: https://www.kreative-eventcontainer.de/anwendungsbeispiele/praesentations-container)	29
7.1	MR-Brille Microsoft HoloLens (Quelle: https://www.microsoft.com/de-de/hololens)	36
7.2	ZB Areal- und Umgebungs-Darstellung in 3D in Blender (Quelle: Eigene Darstellung)	39
7.3	Screenshot Webseite über Docker zur Verfügung gestellt. (Quelle: Eigene Darstellung)	40
7.4	Objekt Selektion in Unity (Quelle: Eigene Darstellung)	41
7.5	Floating Button und Modal in React (Quelle: Eigene Darstellung)	42
7.6	Experiment 5 beim Platzieren einer Kugel. (Quelle: Eigene Darstellung)	43
8.1	Domänenmodell (Quelle: Eigene Darstellung)	44
8.2	Kontextdiagramm (Quelle: Eigene Darstellung)	45
8.3	Containerdiagramm (Quelle: Eigene Darstellung)	46
8.4	Komponentendiagramm API (Quelle: Eigene Darstellung)	47
8.5	Deploymentdiagramm (Quelle: Eigene Darstellung)	48
8.6	Wireframe Startseite (Quelle: Eigene Darstellung)	49
8.7	Wireframe Idee hinzufügen (Quelle: Eigene Darstellung)	49
8.8	Wireframe Infoseite 1/2 (Quelle: Eigene Darstellung)	50
8.9	Wireframe Infoseite 2/2 (Quelle: Eigene Darstellung)	50
8.10	Wireframe Kommentare (Quelle: Eigene Darstellung)	51
8.11	Wireframe Kommentar hinzufügen (Quelle: Eigene Darstellung)	51
9.1	Lokale Ausführung der Integration Tests (Quelle: Eigene Darstellung)	56
9.2	Testcoverage in Gitlab Merge Request (Quelle: Eigene Darstellung)	56
9.3	Testergebnisse in Gitlab Pipeline (Quelle: Eigene Darstellung)	57
9.4	Zertifikat Erhaltungsablauf mithilfe von certbot (Quelle: Eigene Darstellung)	61
9.5	Deploymentablauf (Quelle: Eigene Darstellung)	62

10.1	Layouts - Alte Startseite der Applikation (Quelle: Eigene Darstellung) . . .	64
11.1	Start der Applikation (Quelle: Eigene Darstellung)	67
11.2	Tour Schritt 4 (Quelle: Eigene Darstellung)	68
11.3	Arealgestaltungsübersicht (Quelle: Eigene Darstellung)	69
11.4	Neues Objekt erstellen (Quelle: Eigene Darstellung)	69
11.5	Kommentare zu einem Gebäude (Quelle: Eigene Darstellung)	70
11.6	Feedback Seite (Quelle: Eigene Darstellung)	71
11.7	Zuordnung CustomObjectProperties zu neuem Prefab (Quelle: Eigene Darstellung)	73
11.8	Prefab in BuildingManager hinzufügen (Quelle: Eigene Darstellung) . . .	73
11.9	Zuordnung CustomObjectProperties zu neuem Objekt (Quelle: Eigene Darstellung)	74
B.1	Grober Projektplan (Quelle: Eigene Darstellung)	86
B.2	Zeitauswertung nach Projektmitglied	90
B.3	Zeitauswertung nach Kategorie	91
B.4	Soll-Ist-Vergleich	92
C.1	Areal Zukunft.Bahnhof (Quelle: Eigene Darstellung)	93
C.2	Tour Schritt 1 (Quelle: Eigene Darstellung)	94
C.3	Tour Schritt 2 (Quelle: Eigene Darstellung)	95
C.4	Tour Schritt 3 (Quelle: Eigene Darstellung)	96
C.5	Tour Schritt 4 (Quelle: Eigene Darstellung)	97
C.6	Tour Schritt 5 (Quelle: Eigene Darstellung)	98
C.7	Edit Modus (Quelle: Eigene Darstellung)	99
C.8	Informationen über das Chössli Theater (Quelle: Eigene Darstellung) . . .	100
C.9	Eregebnisse Workshop Fokusgruppe 1/2 (Quelle: Eigene Darstellung) . . .	101
C.10	Eregebnisse Workshop Fokusgruppe 2/2 (Quelle: Eigene Darstellung) . . .	102
C.11	Usability Test 1/4 (Quelle: Eigene Darstellung)	103
C.12	Usability Test 2/4 (Quelle: Eigene Darstellung)	104
C.13	Usability Test 3/4 (Quelle: Eigene Darstellung)	105
C.14	Usability Test 4/4 (Quelle: Eigene Darstellung)	106
C.15	Testabdeckung Integrationstesting (Quelle: Eigene Darstellung)	107

Tabellenverzeichnis

2.1	Zeitdauer	5
2.2	Zeitbudget	5
4.1	Bewertung der Kriterien	17
4.2	Gewichtung der Kriterien	19
4.3	Bewertung K1 Wissenstransfer	20
4.5	Bewertung K2 Transformative Lernerfahrungen	21
4.6	Bewertung K3 Partizipation	22
4.7	Bewertung K4 Präsenzerlebnis	23
4.8	Bewertung K5 Gestaltungsmöglichkeiten	24
4.9	Bewertung K6 Umsetzbarkeit	25
4.10	Auswertung Nutzwertanalyse Realisierungsansätze	25
6.1	Funktionale Anforderungen	30
6.2	Funktionale Anforderungen (optional)	31
6.3	Nicht-funktionale Anforderungen (Functionality)	32
6.4	Nicht-funktionale Anforderungen (Usability)	33
6.5	Nicht-funktionale Anforderungen (Reliability)	33
6.6	Nicht-funktionale Anforderungen (Performance)	33
6.7	Nicht-funktionale Anforderungen (Supportability)	34
9.2	Informationen zur eingesetzten Hardware	52
9.3	Setup Testumgebung	55
9.5	CI/CD Stages	59
9.7	CI/CD Variablen	60
11.1	Stand der funktionalen Anforderungen	65
11.3	Stand der nicht-funktionalen Anforderungen	66
B.1	Organisationsstruktur	85
B.2	Zeitplanung	85
B.3	Risiken	87
B.4	Risikomatrix	87
B.6	Vorgehen mit Risiken	88
B.8	Qualitätsmassnahmen	89

Literaturverzeichnis

- [1] Ricardo Cabello alias mrdoob. mrdoob / three.js. <https://github.com/mrdoob/three.js/>, 2023. [Online; aufgerufen am 12.03.2023].
- [2] Three.js Authors. ThreeJs. <https://threejs.org>, 2023. [Online; aufgerufen am 12.03.2023].
- [3] Babylon.js. Babylon.js: Powerful, Beautiful, Simple, Open - Web-Based 3D At Its Best. <https://www.babylonjs.com/>, 2023. [Online; aufgerufen am 12.03.2023].
- [4] Blender-OSM. Blender-OSM: OpenStreetMap and Terrain for Blender. <https://prochitecture.gumroad.com/l/blender-osm>, 2023. [Online; aufgerufen am 24.03.2023].
- [5] Nadja Brändle. Stiftung zukunft.bahnhof ermöglicht Probewohnen auf dem Land. <https://thephilanthropist.ch/stiftung-zukunft-bahnhof-ermoeglicht-probewohnen-auf-dem-land/>, 2022. [Online; aufgerufen am 25.02.2023].
- [6] Alan B. Craig. *Understanding Augmented Reality : : Concepts and Applications*. Elsevier Science & Technology, Oxford, 2013.
- [7] David A. Kolb, Richard E. Boyatzis, and Charalampos Mainemelis. Experiential learning theory: Previous research and new directions. In *Perspectives on Thinking, Learning, and Cognitive Styles*, pages 227–247. Routledge, 2014.
- [8] Langer Elle. *Medieninnovationen AR und VR, Erfolgsfaktoren für die Entwicklung von Experiences*. Springer, 2020. [Buch].
- [9] Blender Foundation. blender.org - Home of the Blender project - Free and Open 3D Creation Software. <https://www.blender.org/>, 2023. [Online; aufgerufen am 24.03.2023].
- [10] Martin Hohn. Showroom Zukunft.Bahnhof FH. Bei Projektstart ausgehändigt, 2023. [Word Dokument; aufgerufen am 25.02.2023].
- [11] I-maginer. OpenSpace3D - Open Source Platform For 3D Environments. <https://www.openspace3d.com/>, 2023. [Online; aufgerufen am 12.03.2023].
- [12] Adobe Inc. Aero. https://www.adobe.com/ch_de/products/aero.html, 2023. [Online; aufgerufen am 18.03.2023].
- [13] Unity Software Inc. Wondering what Unity is? Find out who we are, where we've been and where we're going — Unity. <https://unity.com/our-company>, 2023. [Online; aufgerufen am 17.03.2023].
- [14] University of Silicon Valley Jennifer Martin. What is a Game Engine? — University of Silicon Valley. <https://usv.edu/blog/what-is-a-game-engine/>, 2023. [Online; aufgerufen am 18.03.2023].
- [15] nicolocarpignoli jeromeetienne. AR.js Documentation. <https://ar-js-org.github.io/AR.js-Docs/>, 2023. [Online; aufgerufen am 18.03.2023].

- [16] Timothy Jung. *XR case studies : : using augmented reality and virtual reality technology in business*. Management for professionals. Springer, Cham, Switzerland, 2021.
- [17] Timothy. Jung. *Extended Reality and Metaverse : : Immersive Technology in Times of Crisis*. Springer Proceedings in Business and Economics Ser. Springer International Publishing AG, Cham, 2023.
- [18] Anna Laros. *Transformatives Lernen nach Mezirow*, pages 19–35. Springer Fachmedien Wiesbaden, Wiesbaden, 2015.
- [19] Katharina Looks. Studie zeigt: Kinder lernen am liebsten mit digitalen Medien. <https://www.scoyo.de/magazin/schule/studie-zeigt-kinder-bevorzugen-digitales-lernen/>, 2023. [Online; aufgerufen am 14.06.2023].
- [20] Mapbox. Maps, geocoding, and navigation APIs & SDKs — Mapbox. <https://www.mapbox.com/>, 2023. [Online; aufgerufen am 23.03.2023].
- [21] Microsoft. Integration tests in ASP.NET Core. <https://learn.microsoft.com/en-us/aspnet/core/test/integration-tests?view=aspnetcore-6.0>, 2023. [Online; aufgerufen am 08.06.2023].
- [22] mozilla. Fundamentals of WebXR - Web APIs — MDN. https://developer.mozilla.org/en-US/docs/Web/API/WebXR_Device_API/Fundamentals, 2023. [Online; aufgerufen am 12.03.2023].
- [23] mozilla. SPA (Single-page application) — MDN. <https://developer.mozilla.org/en-US/docs/Glossary/SPA>, 2023. [Online; aufgerufen am 12.03.2023].
- [24] mozilla. WebGL: 2D and 3D graphics for the web - Web APIs — MDN. https://developer.mozilla.org/en-US/docs/Web/API/WebGL_API, 2023. [Online; aufgerufen am 12.03.2023].
- [25] Open Street Map Foundation (OSMF). OpenStreetMap. <https://www.openstreetmap.org>, 2023. [Online; aufgerufen am 24.03.2023].
- [26] supermedium. A-Frame – Make WebVR. <https://aframe.io/>, 2023. [Online; aufgerufen am 12.03.2023].
- [27] Edward Woodbury Taylor. *The handbook of transformative learning: Theory, research, and practice*. Jossey-Bass higher and adult education series. Jossey-Bass a Wiley Imprint, San Francisco, 1st ed. edition, 2012.
- [28] unternehmer.de. Was ist Experiential Marketing? 5 Beispiele. <https://unternehmer.de/marketing-vertrieb/236878-experiential-marketing>, 2023. [Online; aufgerufen am 03.03.2023].
- [29] React Unity WebGL. React Unity WebGL: Bringing your Unity Games to the Web since 2017! <https://react-unity-webgl.dev/>, 2023. [Online; aufgerufen am 16.04.2023].
- [30] wikipedia. Transformatives Lernen. https://de.wikipedia.org/wiki/Transformatives_Lernen, 2023. [Online; aufgerufen am 10.03.2023].
- [31] Dan Zajdband. GuriVR. <https://gurivr.com/>, 2023. [Online; aufgerufen am 12.03.2023].

LITERATURVERZEICHNIS

- [32] Zukunft.Bahnhof. Zukunft.Bahnhof. <https://www.zukunftbahnhof.ch/>, 2023.
[Online; aufgerufen am 25.02.2023].

Glossar

- Alpine** Alpine ist eine leichtgewichtige Linux Distribution. [41](#)
- AR Foundations** AR Foundations ist ein ein Software Paket für [Unity](#). [37](#)
- Augmented Reality** Unter Augmented Reality auf Deutsch erweiterte Realität, versteht man die computergestützte Erweiterung der Realitätswahrnehmung. [35–37](#)
- Blender-OSM** Plugin für Blender um Open Street Map Daten abzurufen. [40](#)
- C4-Diagramm** Notation zur grafischen Darstellung der Architektur von Softwaresystemen in diversen Schichten. [44](#)
- Corss-Platform** Englisch für Plattformunabhängigkeit. Eine Applikation die Corss-Platform unterstützt kann auf unterschiedliche System ausgeführt werden, z.B. für Andorid und iOS. [37](#)
- Crazy8** Crazy8 ist eine Brainstorming-Methode, bei der jedes Teammitglied in acht Minuten und acht Feldern auf einem Blatt Papier acht Lösungsansätze für ein Problem oder eine Fragestellung visualisiert. Durch die strikte Begrenzung der Zeit müssen Ideen schnell und mit bedingungsloser Kreativität entwickelt werden. [26](#)
- FURPS+** Abkürzung für Functionality, Usability, Reliability, Performance, Supportability und Other (+). Wird in der Softwareentwicklung verwendet, um nicht-funktionale Anforderungen zu kategorisieren. [32](#)
- Game Engine** Eine Game Engine[[14](#)], ist eine Software die die Erstellung von Computer-/Konsolen-Spiele stark vereinfacht. Eine Spiel-Engine ist dafür zuständig die gesamte Spiel Welt darzustellen und zu steuern. [35, 37](#)
- Gzip** Gzip ist ein Kompressions-Format um die Grösse von Dateien auf einem Speichermedium zu verkleinern. [41](#)
- Mapbox** Mapbox stellt unter anderem Satellitenbilder zur Verfügung. [40](#)
- Mixed Reality** Mixed Reality ist ein Überbegriff für [Virtual Reality](#) und [Augmented Reality](#). [35](#)
- NGINX** Nginx ist ein leichtgewichtiger auf Linux basierter Webserver, welcher auch als Reverse Proxy eingesetzt werden kann. [i, 41, 53](#)
- Pinch-Zoom** Pinch-Zoom ist die Möglichkeit auf einem Touchscreen zwei Fingern auseinander oder zueinander zu bewegen und somit hinein- oder herauszuzoomen. [58, 67](#)
- Proof of Concept** Proof of Concept demonstriert die Machbarkeit eines Produktes. Damit kann evaluiert werden wie gut eine Umsetzung machbar ist. [2](#)
- React** React ist eine JavaScript Library zur Gestaltung von Benutzeroberflächen. [i, 53](#)

Scrum Scrum ist ein Vorgehensmodell zur agilen Softwareentwicklung. [84](#)

Unity Unity ist eine sehr bekannte [Game Engine](#). [i](#), [37](#), [40–43](#), [52](#), [53](#), [58](#)

Virtual Reality Unter Virtual Reality auf Deutsch virtuelle Realität, versteht man die computergenerierte virtuelle Realität. [i](#), [35](#)

WebGL WebGL ist eine JavaScript API um hoch performante interaktive 3D und 2D Grafiken darzustellen. [36](#), [37](#), [40–42](#), [57](#)

Abkürzungsverzeichnis

AR Kurzform von [Augmented Reality](#). 8, 24, 35–37

MR Kurzform von [Mixed Reality](#). 24, 35, 37, 38, 40

POC Englisch für [Proof of Concept](#). 39

RUP Kurzform von [Rational Unified Process](#). 84, 85, 90

SPA Kurzform von [Single Page Application](#). 42, 43, 46, 53

VR Kurzform von [Virtual Reality](#). i, 8, 35, 37

ZB Abkürzung für 'zukunft.bahnhof'. 3, 4, 6, 8–18, 20–23, 26, 27, 30, 31, 33, 39, 42, 45, 49, 63, 66, 67, 70, 71, 76

B. Projektplanung

In diesem Kapitel wird auf die Projektplanung während der gesamten Bachelorarbeit eingegangen. So werden die Projektorganisation, die Zeitplanung und das Risikomanagement aufgezeigt. Zudem sind Informationen über das Qualitätsmanagement ersichtlich.

B.1 Projektorganisation

Die Organisation setzt sich aus drei Parteien zusammen. Dies sind zum einen die Teammitglieder des Projektteams, als auch Betreuer und Industriepartner, welche beratende Funktionen einnehmen.

Name	Rolle
Simon Canal	Teammitglied
Linard Vincenz	Teammitglied
Prof. Dr.-Ing. Frieder Loch	Betreuer
Martin Hohn	Industriepartner
Prof. Dr. Daniel P. Politze	Experte

Tabelle B.1: Organisationsstruktur

B.2 Zeitplanung

Die Zeitplanung erfolgt mit der Projektmanagementsoftware YouTrack. So ist eine agile Arbeitsweise gewährleistet, da man die Erfassung der Aufgaben stetig anpassen kann. Die geplanten sowie die effektiven Zeitaufwände werden auf 30 Minuten genau erfasst. Zur Planung des Projekts wird [RUP](#) sowie [Scrum](#) verwendet. Dabei wird für die grobe Planung [RUP](#) und für die Feinplanung [Scrum](#) eingesetzt. Ein Sprint in [Scrum](#) erstreckt sich jeweils über eine Woche. Vor jedem neuen Sprint wird jeweils ein Refinement Meeting und ein Sprint Planning durchgeführt, um die Tasks für den neuen Sprint zu definieren. Während eines Sprints wird pro Woche mindestens ein "DailyScrum" abgehalten, um den Status sowie den Zeitplan zu überprüfen und allfällige Fragen zu klären. Am Ende eines Sprints werden dann abschliessend jeweils sowohl ein Sprint Review als auch eine Sprint Retrospective durchgeführt, um in Erfahrung zu bringen, ob alle Tasks abgearbeitet wurden und gegebenenfalls Justierungen vorgenommen werden müssen.

Phase	Start	Ende
Inception	20.02.2023	05.03.2023
Elaboration	06.03.2023	16.04.2023
Construction	17.04.2023	04.06.2023
Transition	05.06.2023	16.06.2023

Tabelle B.2: Zeitplanung

B.3 Planung

Innerhalb der verschiedenen Phasen von RUP wurden die Arbeitspakete in Epics, User Stories und Tasks aufgeteilt, wobei diese in der genannten Reihenfolge ineinander verschachtelt sind. Im groben Projektplan von **Abbildung B.1** sind alle User Stories und deren Zugehörigkeit zu einem Epic dargestellt. Die geplanten sowie die geleisteten Stunden können ebenfalls dem Projektplan entnommen werden. Auch die Meilensteine mitsamt Erreichungszeitpunkt sind aufgeführt.

Weitere Informationen zu den einzelnen Arbeitspaketen, sowie den geplanten Sprints können direkt in YouTrack (<https://canal.youtrack.cloud/>) abgerufen werden.

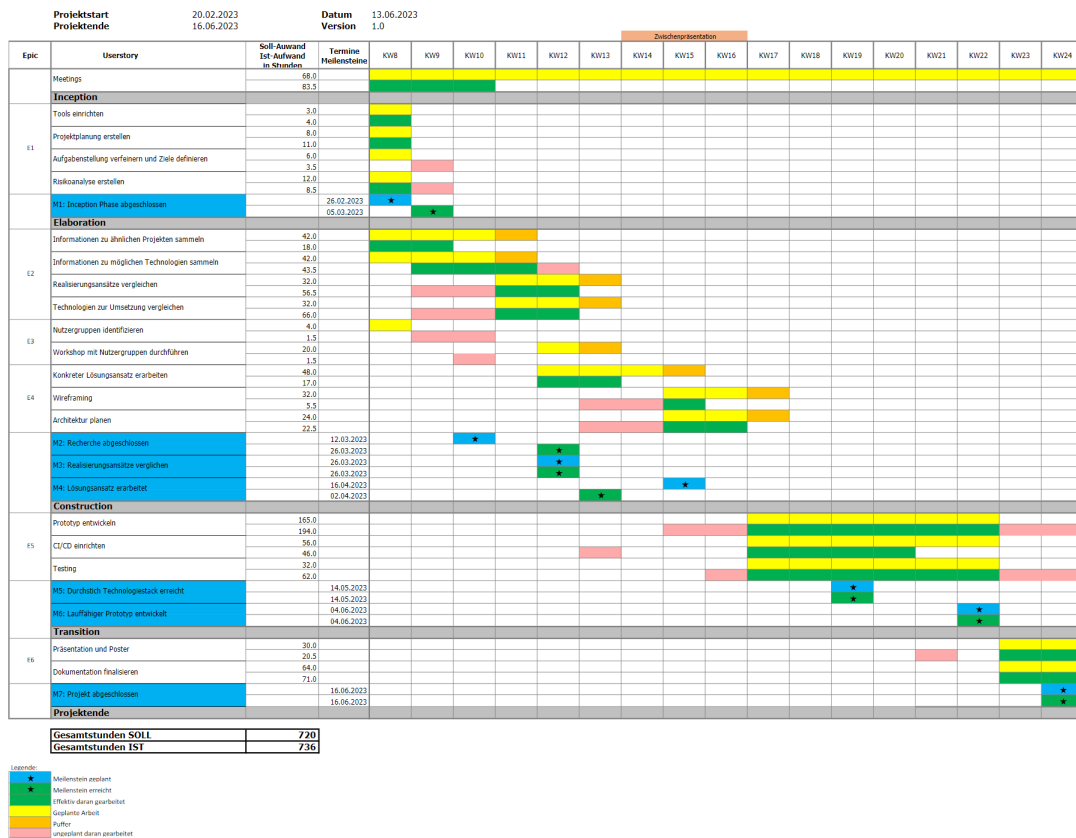


Abbildung B.1: Grober Projektplan (Quelle: Eigene Darstellung)

B.4 Besprechungen

Nachfolgend sind Informationen zu finden, welche über die unterschiedlichen Besprechungen innerhalb des Projektteams sowie mit Betreuer und Industriepartner Auskunft geben.

B.4.1 Besprechung mit Betreuer und Industriepartner

Das Projektteam trifft sich wöchentlich am Freitag über Microsoft Teams mit dem Betreuer. Das Meeting dauert maximal eine Stunde lang und thematisiert den aktuellen Stand, das weitere Vorgehen und anfallende Fragen. Falls das Projektteam keinen Mehrwert in einer Sitzung sieht, kann man diese im Vorfeld absagen. Die Sitzungen werden

protokolliert und das Besprechungsprotokoll in einem dafür vorgesehenen Teams Channel in Form von One Note Notizen abgelegt. Vor jeder Sitzung werden Traktanden festgelegt, welche ebenfalls im Teams Channel hinterlegt sind.

Meetings mit dem Industriepartner werden bei Bedarf geplant und durchgeführt.

B.5 Risikomanagement

Zu diesem Kapitel erhält man Einblicke in die Risikofestlegung und wie mit Risiken umgegangen wird.

B.5.1 Erkannte Risiken

Im Verlauf eines Projekts können jederzeit Risiken eintreten. Um einen möglichen Schaden besser einzugrenzen, sind die erkannten Risiken mit potenzieller Eintrittswahrscheinlichkeit [%] und dadurch vorstellbar entstandene Schäden [h] erfasst. Da es sich hierbei um Schätzungen handelt, sind diese Zahlen nicht als endgültig zu betrachten.

Nr.	Beschreibung	[h]	[%]	Gewichtung
R1	Ungenauere Zeitschätzung, was zu Zeitmangel führt	12	55	6.6
R2	Ausfall eines Teammitgliedes	40	10	4
R3	Wenig bis unbekannte Technologie für den Prototypen	20	50	10
R4	Finanzielle Probleme je nach Prototype	7	10	0.7
R5	Verlust von Daten	15	5	0.75
R6	Lösungsansatz nicht realisierbar	20	15	3
R7	Abhängigkeit von Dritten (Nutzungsgruppe)	8	50	4
R8	Konflikte im Team	8	5	0.4
Total Schaden:		130		Total Gewichtung: 29.45

Tabelle B.3: Risiken

B.5.2 Risikomatrix

In der Risikomatrix sind die ursprünglich erkannten und dokumentierten Risiken dargestellt.

Probabilität	Schweregrad			
	Vernachlässigbar	Marginal	Kritisch	Katastrophal
Hoch				
Wahrscheinlich				
Möglicherweise	R7	R1	R3	
Unwahrscheinlich			R6	
Selten	R4, R5, R8			R2

Tabelle B.4: Risikomatrix

B.5.3 Vorgehen mit Risiken

Zur Risikominimierung sind in nachfolgender Tabelle Erklärungen enthalten, welche dazu dienen sollen, den Risikoeintritt bestmöglich zu verhindern. Zusätzlich sind der Tabelle Massnahmen zu entnehmen, falls ein Risiko tatsächlich eintreten sollte.

Nr.	Vorbeugung	Verhalten beim Eintreten
R1	Planung regelmässig anpassen, damit man frühzeitig erkennt, falls etwas nicht wie gewünscht funktionieren sollte.	Sinnvolle Priorisierung der Aufgaben, um trotzdem alle Muss-Ziele zu erreichen.
R2	Gesunde Ernährung und Sport sowie Puffer-Slots einbauen.	Sinnvolle Neupriorisierung der Aufgaben, allenfalls Anforderungen streichen.
R3	Genügend Zeit für die Informationsbeschaffung, offizielle Dokumentationen hinzuziehen.	Offizielle Dokumentationen hinzuziehen, mehr Zeit für die Recherche über entsprechende Technologie einplanen.
R4	Zu kostspielige Gadgets wenn möglich vermeiden. Im Vorfeld klären wie hoch das Budget ist.	Prototyp so anpassen, dass er mit weniger finanziellen Mitteln umsetzbar ist.
R5	Teammitglieder behalten ihre Änderungen auch nach dem Mergen auf ihren lokalen Maschinen. Bei der Recherche die Ergebnisse direkt dokumentieren und auf Gitlab pushen.	Backup aus lokalen Daten wiederherstellen.
R6	Genaue Recherche über möglichen Lösungsansatz.	Anderen Lösungsansatz wählen, bzw. die Lösung so ändern, dass eine Umsetzung eines Prototyps möglich ist.
R7	Möglichst früh im Projekt die Bedürfnisse der Nutzungsgruppe erfassen.	-
R8	Offener und direkter Austausch untereinander.	Intervention durch Betreuer.

Tabelle B.6: Vorgehen mit Risiken

B.6 Qualitätsmassnahmen

Damit das Produkt eine hohe Qualität aufweist, werden nachfolgende Vorkehrungen getroffen.

Massnahme	Zeitraum	Ziel
Sprint Reviews	Sprintende	Es wird regelmässig über den aktuellen Stand informiert, damit man Probleme frühzeitig detektieren kann
Code Reviews	Merge Request	Einschleichung von Fehlern vermeiden und sauberen Code garantieren
Meilensteine	Fixer Tag	Erfolgreiche Beendigung des Projekts soll so gewährleistet sein.
Automatisiertes Testing	Push	Testen der Funktionalität des Codes
Definition of done	Beenden der Aufgabe	DoD ist gegeben, falls B.6.1 eingehalten sind

Tabelle B.8: Qualitätsmassnahmen

B.6.1 Definition of done

- CI erfolgreich. Bei Status 'Failed' zwingend Fehlerkorrektur durchführen.
- Code Review erfolgreich.
- Bei Sprint Reviews erfolgt die Beendigung des Tasks endgültig

B.6.2 Dokumentation

Die gesamte Dokumentation ist auf dem Gitlab Server der OST gespeichert. Das Projektteam arbeitet direkt auf dem Master Branch. Bei jedem Push wird das Dokument im PDF-Format als Artefakt neu kompiliert. Die Versionisierung wird dadurch sichergestellt.

B.6.3 Projektmanagement

Um das Projektmanagement zu überwachen, wird YouTrack verwendet. Die Aufgaben werden durch eine Aufwandsschätzung erfasst. Der tatsächliche Aufwand wird nachgetragen. Unter dem Reiter Bericht sind die Gesamtzeiten der Mitglieder, Zeiten nach Aufgabe sowie nach Sprint und Epics einzusehen.

B.6.4 Code Reviews

Jedes Projektmitglied erstellt nach der Fertigstellung einer Aufgabe einen Merge Request und weist diesen dem anderen Teammitglied zu. Das zugewiesene Teammitglied reagiert

darauf und überprüft zeitnah, ob im Merge Request alle Qualitätsmassnahmen sauber eingehalten sind. Darauf hin wird der Merge Request bestätigt.

B.7 Projektmonitoring

In diesem Kapitel sind Auswertungen zum Projektverlauf und der Zeitplanung abgebildet.

B.7.1 Zeitauswertung

Die nachfolgenden Zeitauswertungen basieren auf den in YouTrack erfassten Daten. Weitere Auswertungen lassen sich direkt in YouTrack unter <https://canal.youtrack.cloud/reports/timeSheet/149-17> einsehen.

Arbeitsverteilung

Die Grafik veranschaulicht den Zeitaufwand in Stunden für jedes Projektmitglied.

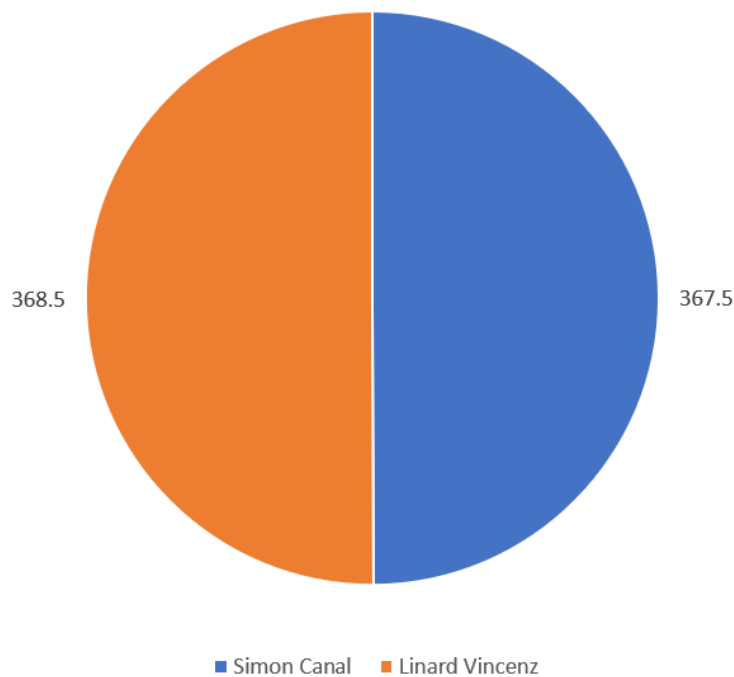


Abbildung B.2: Zeitauswertung nach Projektmitglied

Arbeitsbereich

Die nachfolgende Grafik zeigt die Aufteilung der Zeitaufwände auf die verschiedenen Phasen von RUP. Alle Werte in der Grafik sind in Stunden angegeben.

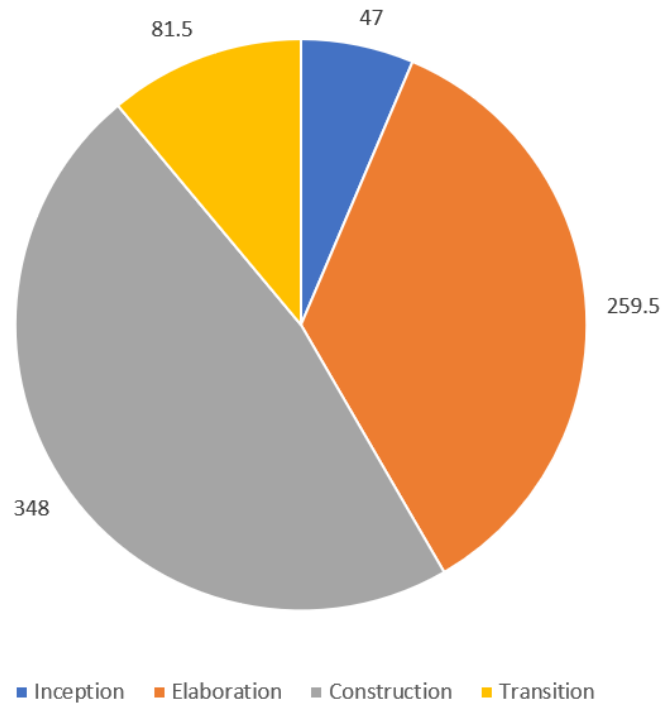


Abbildung B.3: Zeitauswertung nach Kategorie

Soll-Ist-Vergleich

Bei Projektbeginn wurden Zeitschätzungen für alle User Stories erfasst. In der [Abbildung B.4](#) sind diese Schätzungen im Vergleich zu den effektiven Zeitaufwänden dargestellt. Auch in dieser Abbildung sind die Werte in Stunden angegeben.

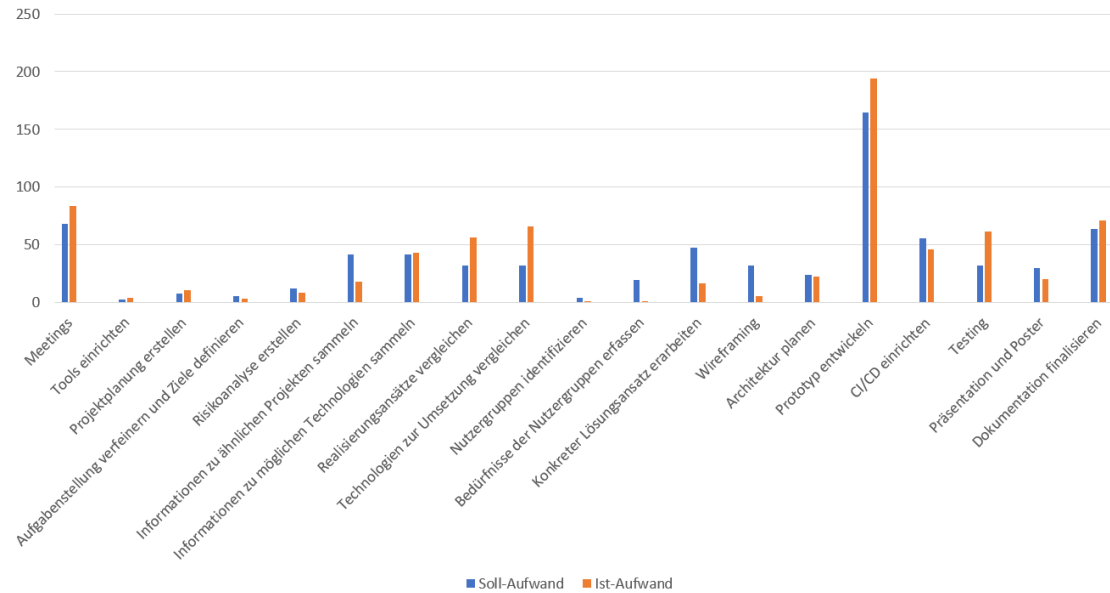


Abbildung B.4: Soll-Ist-Vergleich

C. Anhang

C.1 Screenshots Prototyp

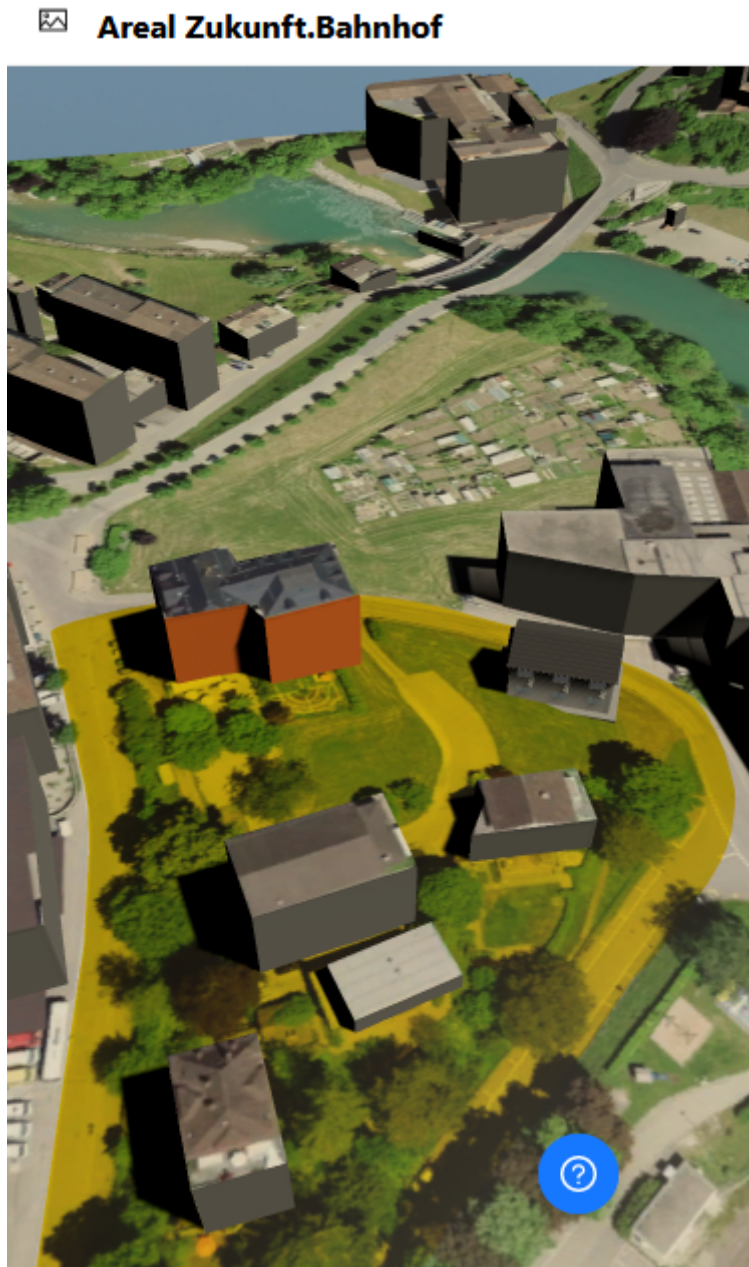


Abbildung C.1: Areal Zukunft.Bahnhof (Quelle: Eigene Darstellung)

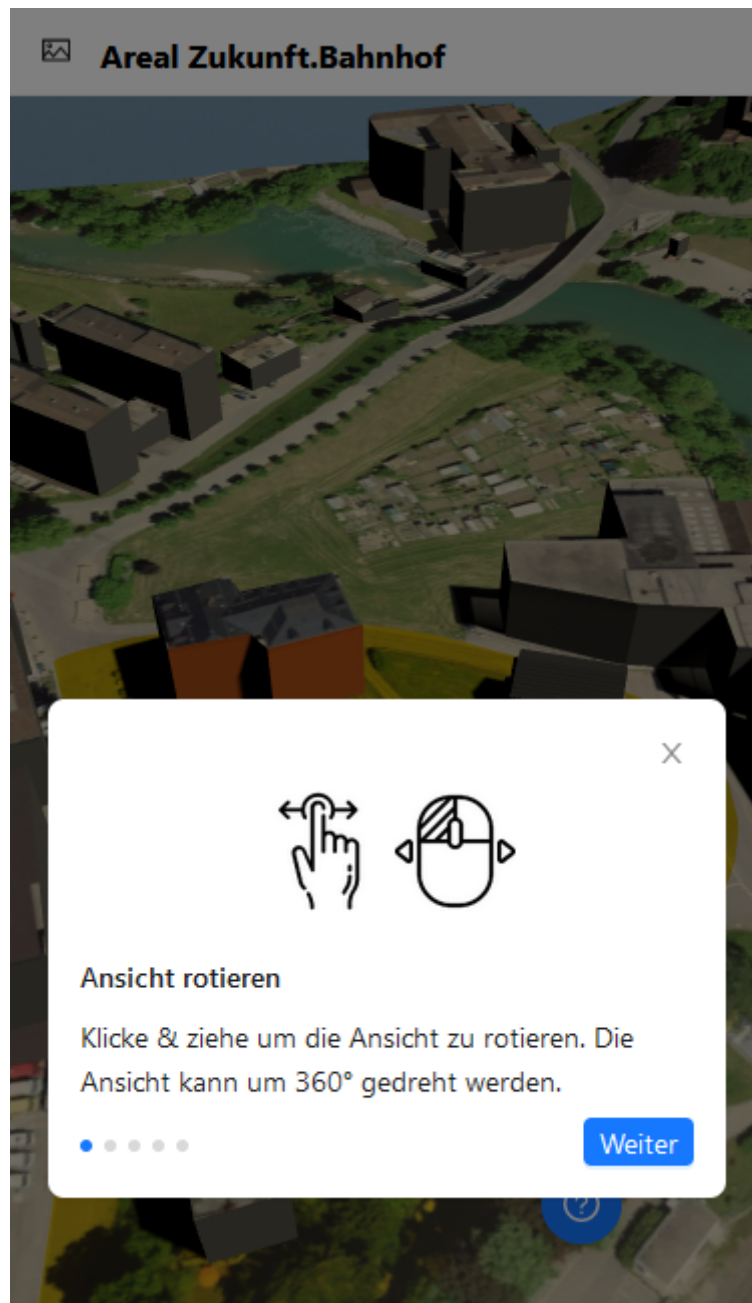


Abbildung C.2: Tour Schritt 1 (Quelle: Eigene Darstellung)



Abbildung C.3: Tour Schritt 2 (Quelle: Eigene Darstellung)

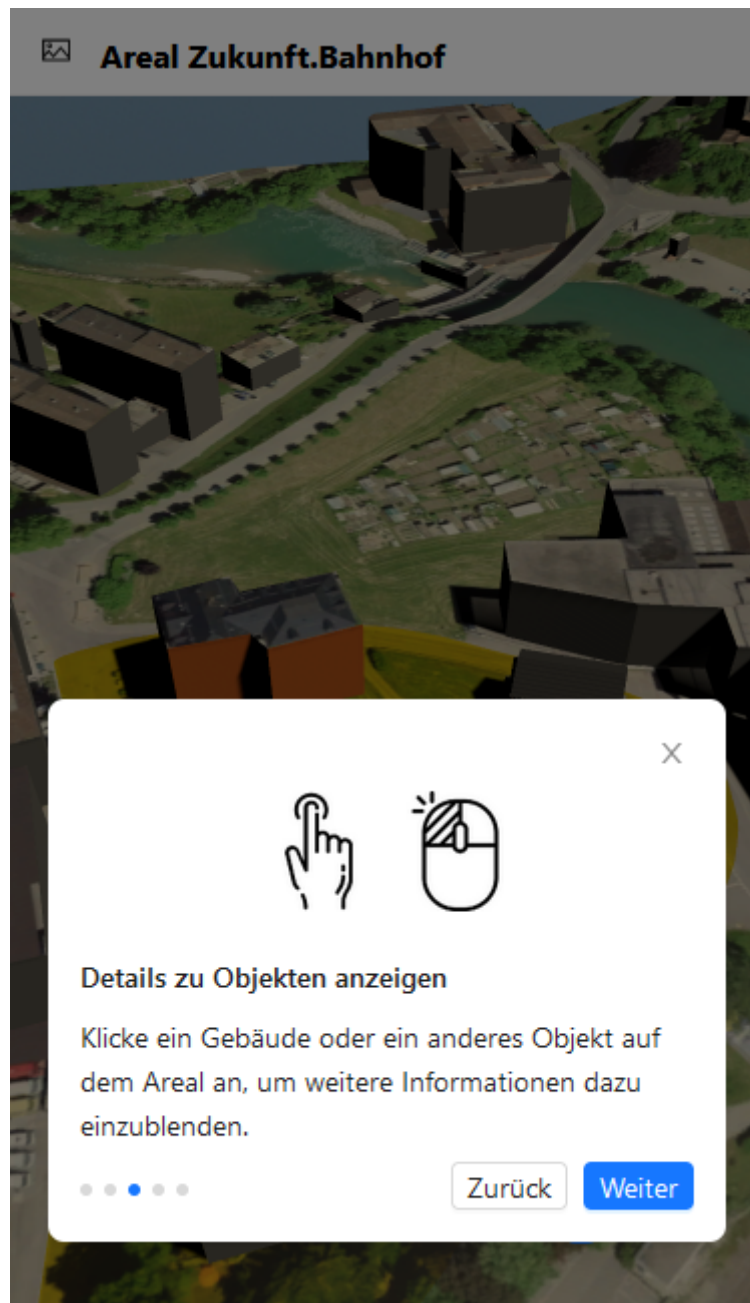


Abbildung C.4: Tour Schritt 3 (Quelle: Eigene Darstellung)



Abbildung C.5: Tour Schritt 4 (Quelle: Eigene Darstellung)



Abbildung C.6: Tour Schritt 5 (Quelle: Eigene Darstellung)

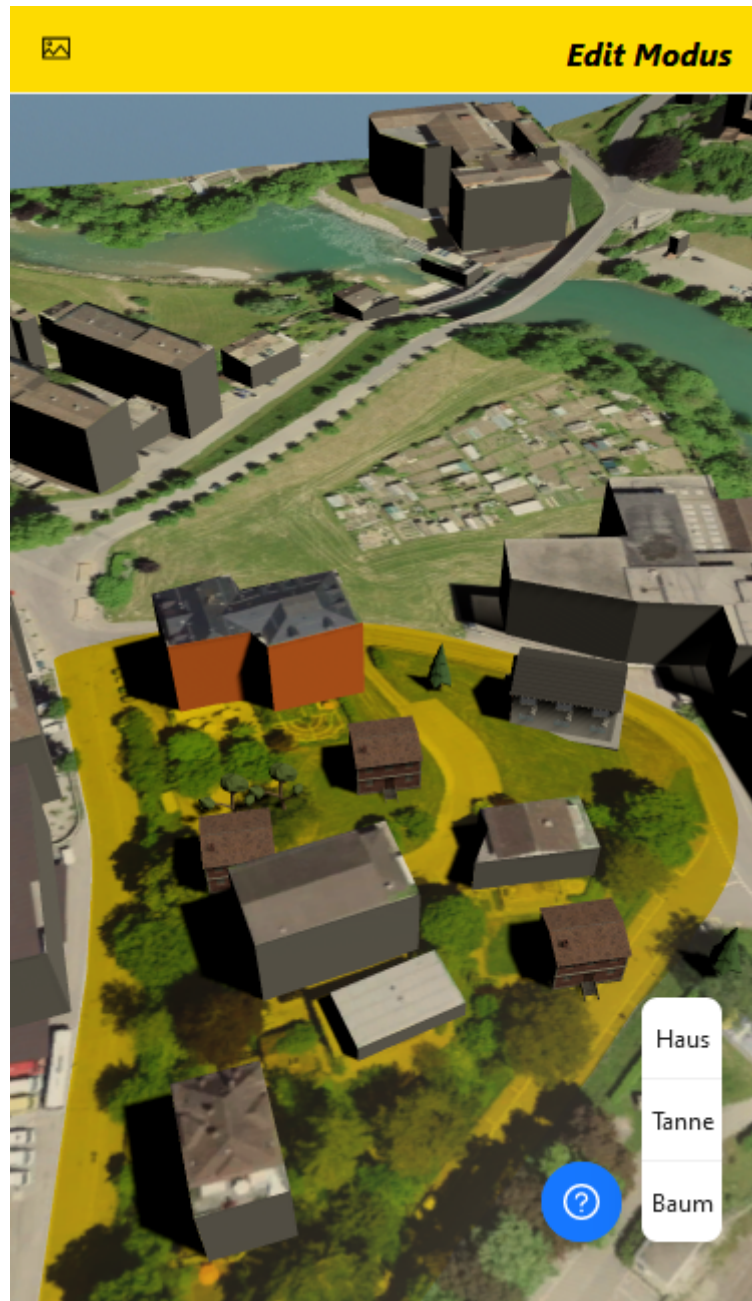
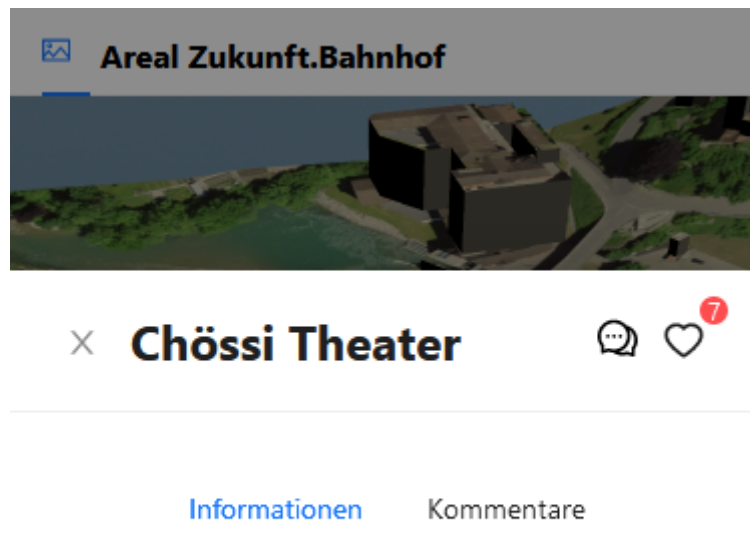


Abbildung C.7: Edit Modus (Quelle: Eigene Darstellung)



Kompetenzzentrum

Innenbereich

Das bereits vorhandenen Gebäude des Chössli Theater soll vielseitig genutzt werden.

Einerseits wird im Gebäude ein Showroom errichtet, welcher der Öffentlichkeit Themen und Inhalte zum Projekt zukunft.bahnhof vorstellt und interaktiv vermittelt.

Andererseits können die Innenräume der Bahnhalle für diverse Veranstaltungsformate und kulturelle Angebote jeglicher Art genutzt werden.

Abbildung C.8: Informationen über das Chössli Theater (Quelle: Eigene Darstellung)

C.2 Ergebnisse Workshop Fokusgruppe



- Interessen der Bevölkerung sammeln
- Film mit Animationen
- Interessant gestalten für Familien
- VR animation

Abbildung C.9: Ergebnisse Workshop Fokusgruppe 1/2 (Quelle: Eigene Darstellung)

C.2. ERGEBNISSE WORKSHOP FOKUSGRUPPE

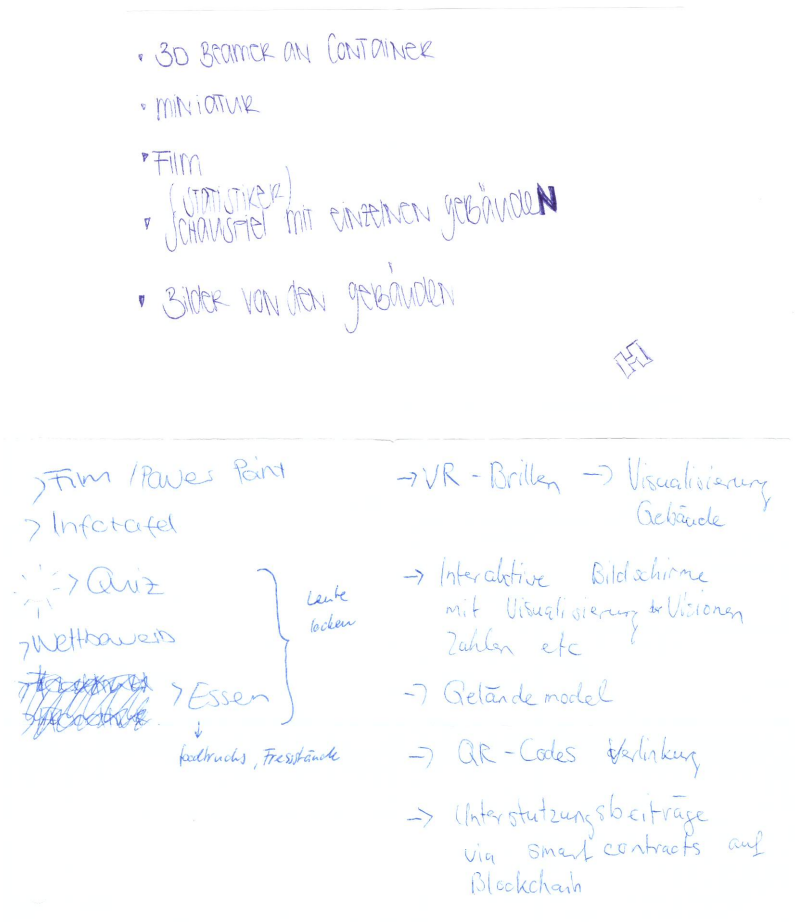


Abbildung C.10: Ergebnisse Workshop Fokusgruppe 2/2 (Quelle: Eigene Darstellung)

C.3 Usability Test

BA - Entwicklung eines Showrooms für den zukunf.bahnhof Lichtensteig

Zukunft.Bahnhof Usability Test



Nr.	Aufgabe	Zufriedenheit (1-10)	Bemerkung
1.	Neues Layout erstellen. Gewünschter Name des Layouts eingeben.	8	Es ist klar wie + was zu tun ist, aber das Interface ist noch nicht so ansprechend
2.	Erstelltes Layout ansehen. Wenn dies nicht möglich war, kann ein vorhandenes Layout angesehen werden (Lichtensteig 2.0)	10	funktioniert gut
3.	Sich in der virtuellen Welt umsehen. Hinweis: Zoomen ist auf mobilen Geräten nur im Beta Stadium und funktioniert nicht allzu verlässlich.	9	mir hat keine ironie, dass ich nur das Chossi - Gebäude für mehr Informationen anschauen konnte *
4.	Sich über das Chössi Theater informieren. Hinweis: Das Gebäude mit der orangen Fassade ist das Chössi Theater.	10/9	nach dem klick auf das Gebäude könnte der Name des Gebäudes im Text präsent sein
5.	Einen Kommentar zum Chössi Theater oder eines anderen Gebäudes hinterlegen.	10	
6.	Ein Objekt in der virtuellen Welt platzieren. 1. Zur Auswahl steht eine Tanne , ein Haus oder ein Baum . 2. Das platzierte Objekt benennen und eine sinnvolle Beschreibung eingeben. Tipp: Das Objekt kann vor dem Speichern auch nochmals umplatziert werden. Zudem kann, bei der Verwendung von Maus und Tastatur, das Objekt zusätzlich um die eigene Achse gedreht werden (Rechtsklick).	9	finde ich nett + klappt gut, ich hätte Lust gehabt einen kleinen Wald zu pflanzen, fand es dann aber nicht so für jeden Baum einen Namen eingeben zu müssen

Simon Canal und Linard Vincenz

Abbildung C.11: Usability Test 1/4 (Quelle: Eigene Darstellung)

C.3. USABILITY TEST

BA - Entwicklung eines Showrooms für den zukunft.bahnhof Lichtensteig

7.	Dem soeben platzierten Gebäude ein Like geben. <i>Hinweis: Es können auch andere Gebäude mit einem Like versehen werden.</i>	9	die Icons könnten parallel zum Namen des Gebäudes etwas präsentierbarer sein. KONNTE! ☺
8.	Sehen was andere über das Projekt denken: 1. Zurück zur Übersicht gehen. 2. Ein anderes Layout ansehen (z.B. Lichtensteig 2.0) 3. Sich hier umsehen und gegebenenfalls Gebäude kommentieren oder liken.		Klappt und erklärt sich selbst, aber denkt hier ist mir das Gebäudetitel nicht präsent genug (habe erst vor absenden keinen Kommentar gemacht, dass es sich um eine KIT handelt)
9.	Zurück zu Layouts: Das angesehene Layout kommentieren und auf Wunsch liken.	10	Klappt gut, spricht auch aber nicht so an das zu machen
10.	Unter Feedback ein Feedback über das Projekt geben.	10	Klappt alles super, like face finde ich nicht so ansprechend

vllt. anderes wording?

"Layouts" → "wie sehen andere Lichtensteig"
"wie stellst du dir die Zukunft vor?"
"wie sieht die Zukunft aus?"

* ich finde es angebraten schon einen Hinweis auf das Gebäude zu bekommen bevor ich klicke. Wenn ich über die Karte ~~so~~ fahren könnte ich so schneller erfassen welche Infrastruktur besteht.

Simon Canal und Linard Vincenz

Abbildung C.12: Usability Test 2/4 (Quelle: Eigene Darstellung)

C.3. USABILITY TEST

BA - Entwicklung eines Showrooms für den zukunfft.bahnhof Lichtensteig

Zukunft.Bahnhof Usability Test



Nr.	Aufgabe	Zufriedenheit (1-10)	Bemerkung
1.	Neues Layout erstellen. Gewünschter Name des Layouts eingeben.	2	- Keine intuitiv verständliche Oberfläche → Handlungsaufforderung fehlt!
2.	Erstelltes Layout ansehen. <i>Wenn dies nicht möglich war, kann ein vorhandenes Layout angesehen werden (Lichtensteig 2.0)</i>	7	+ Auge symbol macht Sinn.
3.	Sich in der virtuellen Welt umsehen. <i>Hinweis: Zoomen ist auf mobilen Geräten nur im Beta Stadium und funktioniert nicht allzu verlässlich.</i>	8	+ Einführung o.k. ✓
4.	Sich über das Chössli Theater informieren. <i>Hinweis: Das Gebäude mit der orangen Fassade ist das Chössli Theater.</i>	8	~ wäre schön, wenn es animiert wäre?!
5.	Einen Kommentar zum Chössli Theater oder eines anderen Gebäudes hinterlegen.	9	+ funktioniert.
6.	Ein Objekt in der virtuellen Welt platzieren. <ol style="list-style-type: none"> Zur Auswahl steht eine Tanne, ein Haus oder ein Baum. Das platzierte Objekt benennen und eine sinnvolle Beschreibung eingeben. <i>Tipp: Das Objekt kann vor dem Speichern auch nochmals umplatziert werden. Zudem kann, bei der Verwendung von Maus und Tastatur, das Objekt zusätzlich um die eigene Achse gedreht werden (Rechtsklick).</i>	5	- Handlungsaufforderung fehlt! - keine Veränderung nach Speichern möglich!

Simon Canal und Linard Vincenz

Abbildung C.13: Usability Test 3/4 (Quelle: Eigene Darstellung)

C.3. USABILITY TEST

BA - Entwicklung eines Showrooms für den zukunft.bahnhof Lichtensteig

7.	Dem soeben platzierten Gebäude ein Like geben. <i>Hinweis: Es können auch andere Gebäude mit einem Like versehen werden.</i>	8	- Herzli ist zu klein ↳ sollte rot werden
8.	Sehen was andere über das Projekt denken: 1. Zurück zur <u>Übersicht</u> gehen. 2. Ein anderes Layout ansehen (z.B. Lichtensteig 2.0) 3. Sich hier umsehen und gegebenenfalls Gebäude kommentieren oder liken.	5	~wäre evtl. besser das in das Layout (kopfzeile?) zu integrieren?! -Kommentar funktion ist doppelt?!
9.	Zurück zu Layouts: Das angesehene Layout Kommentieren und auf Wunsch liken.	6	~evtl. Vorschau funktion?
10.	Unter Feedback ein Feedback über das Projekt geben.	8	+ die Rückmeldung ist gut!

Vorschlag → "fliegender Besuch" aller Objekte.

Simon Canal und Linard Vincenz

Abbildung C.14: Usability Test 4/4 (Quelle: Eigene Darstellung)

C.4 Testabdeckung

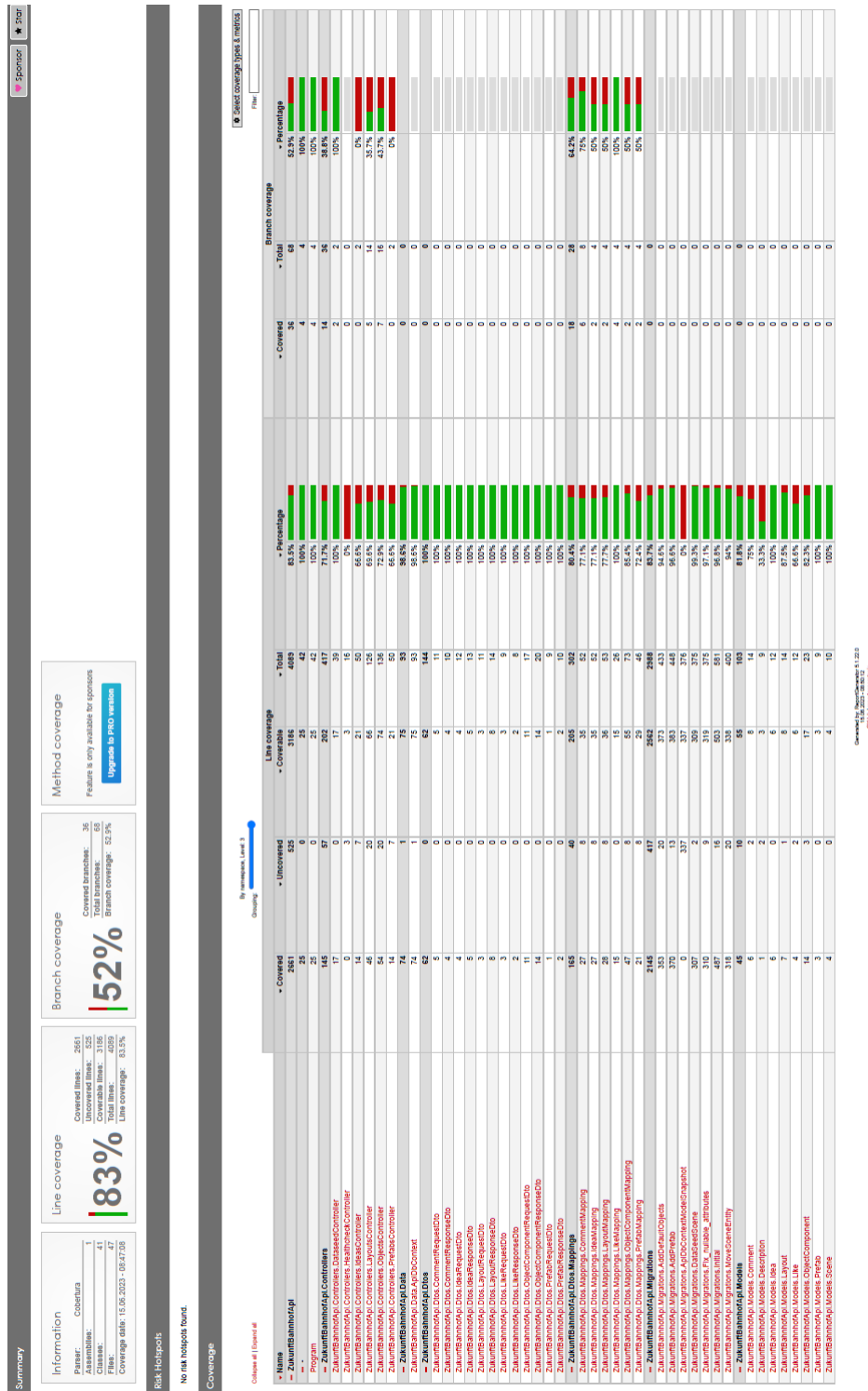


Abbildung C.15: Testabdeckung Integrationstesting (Quelle: Eigene Darstellung)