

Bachelorarbeit 2015

Augmented Reality for Museums



Abb. 1: Logo Erasmus MMXVI (Quelle: Notter + Vigne Sàrl)

Studentin : Hischier Evelyn

Dozent : Müller Henning

Abgegeben am 27. Juli 2015

Zusammenfassung

2016 nimmt Basel das 500. Jubiläum des *Novum Instrumentum* (neues Testament) von Erasmus von Rotterdam als Anlass diesen zu feiern. Dazu soll es eine Sonderausstellung in und um das Historische Museum in Basel geben. Verantwortlich für diese Ausstellung ist Marcel Henry, wissenschaftlicher Ausstellungsassistent. Die Idee von Marcel Henry ist, dass eine App für iOS und Android eine Verbindung von der heutigen Zeit zu der Zeit von Erasmus von Rotterdam zu erstellen. Das moderne Smartphone soll der Blick in die Vergangenheit ermöglichen.

Das Ziel dieser Arbeit ist einen Prototyp für die Android App zu erstellen. Dabei sollen Augmented Reality und Virtual Reality als neue Technologien erlernt und eingesetzt werden. Dieser Prototyp soll anschliessend von Roger Schaer und Antoine Widmer zu einer finalen Version weiterentwickelt werden. Betreut wurde die Arbeit durch Henning Müller, Antoine Widmer und Roger Schaer.

Zu Beginn werden die zwei neuen Technologien kurz vorgestellt. Des Weiteren musste in einem nächsten Schritt entschieden werden, welche Software und Hardware zum Einsatz kommen wird. Dazu wurden eigene Tests erstellt, um diese miteinander zu vergleichen und zu analysieren. Auf den Ergebnissen basierend wurde entschieden welcher Augmented Reality Anbieter im Prototyp verwendet wird. Im Bereich Virtual Reality war die Entscheidung einfacher, weil zum jetzigen Zeitpunkt erst ein paar wenige Virtual Reality Geräte auf dem Markt sind.

Schlüsselbegriffe: Augmented Reality, Virtual Reality, Museum, Qualcomm Vuforia, Unity, Android

Danksagung

Ich möchte mich bei allen Personen bedanken, welche mich während der Bachelorarbeit unterstützt haben. Einen besonderen Dank geht an die folgenden Personen:

Betreuer des administrativen Aspekts, Henning Müller, für regelmässige Feedbacks und Inputs zur Bachelorarbeit.

Betreuer der technischen Aspekte, Antoine Widmer und Roger Schaer, für wöchentliche Feedbacks, Tipps und Unterstützung.

Institut der Hes-so Wallis für die Zurverfügungstellung von einem Smartphone und Google Cardboard für die Entwicklung.

Marcel Henry für die bereitgestellten Inhalte, rasche Feedbacks zu den Prototypen und gute Zusammenarbeit.

Inhaltsverzeichnis

Tabellenverzeichnis.....	vi
Abbildungsverzeichnis.....	vii
Abkürzungsverzeichnis.....	ix
Einleitung.....	1
1 Was ist Augmented und Virtual Reality?	2
1.1 Augmented Reality	2
1.1.1 Was ist Augmented Reality?	2
1.1.2 Welche AR Projekte gibt es bereits?	2
1.1.3 Forschung	3
1.2 Virtual Reality	4
1.2.1 Definition und Ursprung	4
1.2.2 Wie wird VR heute eingesetzt?	4
2 Methodik.....	6
2.1 Recherche- und Analysephase	6
2.2 Implementierungsphase.....	6
3 Recherche- und Analysephase	7
3.1 Augmented Reality	7
3.1.1 Auswahl der SDKs.....	7
3.1.2 Vergleichskriterien	7
3.1.3 Vergleich.....	8
3.1.4 Tests mit PixLive	10
3.1.5 Entscheid: Passende SDK auswählen	10
3.2 Virtual Reality	11
3.2.1 Auswahl der Head-Mounted-Display Geräte.....	11
3.2.2 Vergleichskriterien	11
3.2.3 Vergleich.....	11
3.2.4 Entscheid	13
3.2.5 Cardboard und 360° Fotos	13
3.3 360° Videos.....	13
4 Verwendete Tools	14
4.1 Vuforia	14
4.2 Android Studio.....	14
4.3 Google Cardboard.....	14
4.4 Unity	15

4.5	Ninjamock.....	15
4.6	Erste Versuche mit Vuforia und Cardboard	15
5	Erasmus – Prototyp App.....	16
5.1	Mockups	16
5.2	Wie entstand dieser Prototyp?	17
5.2.1	Wie erkennt Vuforia die verschiedenen Bilder / QR Codes?	17
5.2.2	Wie werden diese QRCs erweitert?	18
5.2.3	Wie wird die Unity App mit der Normalen App verbunden?.....	20
5.3	Wie funktioniert die App?	21
	Schlussfolgerung	24
	Worin lagen die Schwierigkeiten und wie wurden diese gelöst?	24
	Neue Technologie.....	24
	Keine Integration von Cardboard in Vuforia	24
	Verschiedene Softwareversionen	25
	Limitation der App Grösse.....	26
	Was hätte man anders / besser machen können?	26
	AR und Cardboard Test	26
	Unit Tests.....	27
	Wie geht es weiter mit der Erasmus App?.....	27
	Literaturverzeichnis.....	28
	Anhang I: Product Backlog	31
	Anhang II: Stundenrapport.....	32
	Anhang III: Mockups.....	36
	Anhang IV: Code	38
	Anhang V: QRC	39
	Selbstständigkeitserklärung der Verfasserin.....	41

Tabellenverzeichnis

Tabelle 1: Augmented Reality SDK Vergleich. (Eigene Darstellung)	
Quelle: Siehe Tabellenende	9
Tabelle 2: Vergleich Virtual Reality Head-Mounted-Display Geräte. (Eigene Darstellung)	
Quelle: Siehe Tabellenende	12
Tabelle 3: Product Backlog (Eigene Darstellung)	31
Tabelle 4: Stundenrapport mit Übersicht über Probleme und deren Lösungen (Eigene Darstellung)	32

Abbildungsverzeichnis

Abb. 1: Logo Erasmus MMXVI (Quelle: Notter + Vigne Sàrl)	i
Abb. 2: Screenshot Google Camera (Eigene Darstellung)	13
Abb. 3: Google Cardboard (Quelle: http://thewestsidestory.net/wp-content/uploads/2014/12/Google-Cardboard-VR-headset.jpeg).....	14
Abb. 4: Mockup Sprachauswahl (Eigene Darstellung)	16
Abb. 5: Screenshot Android Studio Sprachauswahl (Eigene Darstellung)	16
Abb. 6: Screenshot von Features auf einem Zielbild (Eigene Darstellung)	17
Abb. 7: Screenshot Startseite (Eigene Darstellung)	21
Abb. 8: Screenshot Einstellungen (Eigene Darstellung)	21
Abb. 9: Screenshot Download (Eigene Darstellung)	22
Abb. 10: Screenshot Bildschirmeinstellung (Eigene Darstellung)	22
Abb. 11: Screenshot Ladebildschirm (Quelle: Unity)	23
Abb. 12: Screenshot duplizierte Vuforiakamera (Eigene Darstellung)	24
Abb. 13: Mockup Startseite (Eigene Darstellung)	36
Abb. 14: Mockup Beispiel Bild (Eigene Darstellung)	36
Abb. 15: Mockup Download (Eigene Darstellung)	36
Abb. 16: Mockup Cardboardansicht (Eigene Darstellung).....	36
Abb. 17: Mockup Bildschirmeinstellung (Eigene Darstellung).....	36
Abb. 18: Mockup Cardboardansicht mit Beispiel Inhalt (Eigene Darstellung).....	36
Abb. 19: Mockup Tabletansicht mit Beispiel Inhalt (Eigene Darstellung).....	37
Abb. 20: Mockup Tabletansicht (Eigene Darstellung).....	37
Abb. 21: Mockup Standortauswahl (Eigene Darstellung)	37
Abb. 22: Mockup Museumsplan (Eigene Darstellung).....	37
Abb. 23: Mockup Tramtour (Eigene Darstellung)	37
Abb. 24: Mockup Beispieltext (Eigene Darstellung).....	37
Abb. 25: Mockup Sponsoren (Eigene Darstellung)	37
Abb. 26: QRC 1 (Quelle: Notter + Vigne Sàrl)	39
Abb. 27: QRC 2 (Quelle: Notter + Vigne Sàrl)	39
Abb. 28: QRC 3 (Quelle: Notter + Vigne Sàrl)	39
Abb. 29: QRC 4 (Quelle: Notter + Vigne Sàrl)	39

Abb. 30: QRC 5 (Quelle: Notter + Vigne Sàrl)	39
Abb. 31: QRC 6 (Quelle: Notter + Vigne Sàrl)	39
Abb. 32: QRC 7 (Quelle: Notter + Vigne Sàrl)	39
Abb. 33: QRC 8 (Quelle: Notter + Vigne Sàrl)	39
Abb. 34: QRC 9 (Quelle: Notter + Vigne Sàrl)	39
Abb. 35: QRC 10 (Quelle: Notter + Vigne Sàrl)	40
Abb. 36: QRC 11 (Quelle: Notter + Vigne Sàrl)	40
Abb. 37: QRC 12 (Quelle: Notter + Vigne Sàrl)	40
Abb. 38: QRC 13 (Quelle: Notter + Vigne Sàrl)	40
Abb. 39: QRC 14 (Quelle: Notter + Vigne Sàrl)	40
Abb. 40: QRC 15 (Quelle: Notter + Vigne Sàrl)	40
Abb. 41: QRC 16 (Quelle: Notter + Vigne Sàrl)	40
Abb. 42: QRC 17 (Quelle: Notter + Vigne Sàrl)	40
Abb. 43: QRC 18 (Quelle: Notter + Vigne Sàrl)	40

Abkürzungsverzeichnis

App	Anwendungssoftware (Englisch: Application Software)
AR	Augmented Reality
DK	Development Kit
GB	Gigabyte
HUD	Head-up-Display
IDE	Integrierte Entwicklungsumgebung (Englisch: Integrated Development Environment)
JAR	Java Archive
MB	Megabyte
px	Pixel
QRC	Quick Response Code
SD Karte	Sichere digitale Speicherkarte
SDK	Software Development Kit
URL	Uniform Resource Locator
VR	Virtual Reality

Einleitung

Das Ziel dieser Bachelorarbeit ist einen Prototyp für das Historische Museum in Basel zu entwickeln. 2016 feiert das Werk *Novum Instrumentum* (neues Testament) von Erasmus von Rotterdam sein 500. Jubiläum. Dazu organisiert das Museum eine Sonderausstellung über Erasmus von Rotterdam. Verantwortlich für diese Ausstellung ist Marcel Henry, wissenschaftlicher Ausstellungsassistent. Die Idee von Marcel Henry ist, dass eine App für iOS und Android eine Verbindung von der heutigen Zeit zu der Zeit von Erasmus von Rotterdam zu erstellen. Das moderne Smartphone soll der Blick in die Vergangenheit ermöglichen. Funktionieren soll die App mit Augmented und Virtual Reality. Mit dieser App erhofft sich das Museum mehr Besucher anzulocken. Erwartet werden während der ganzen Laufzeit um die 20'000 Besucher. Die Herausforderung dieser Arbeit liegt darin, neue Technologien kennen zu lernen und diese in einem Prototyp für Android Smartphones zu verbinden. Ebenfalls muss die App den Ansprüchen des Kunden, dem Historisches Museum, genügen.

Motivation

Seit ca. acht Jahren gibt es das moderne Smartphone. Gemäss Farago (2012) hatten im Juli 2011 bereits 76% der 15- bis 64 Jährigen in der Schweiz ein Smartphone. Tendenz steigend.

Wie lässt sich also ein Museumsbesuch mit der heutigen mobilen Technik verbinden?

Viele Museen bieten eine Audiotour an. Früher geschah dies über spezielle Mediaplayer. Da heute aber fast jeder Besucher einen eigenen Mediaplayer in Form eines Smartphones bei sich trägt, lässt sich dieses gut dafür verwenden. Aber das reicht in der heutigen Zeit nicht mehr für einen aussergewöhnlichen Museumsbesuch.

Wie kann also ein Besucher ein Museum auf einer ganz neuen Weise entdecken? In ein paar einzelnen Museen kann eine Ausstellung oder einzelne ausgewählte Ausstellungsobjekte bereits virtuell mit seinem Smartphone besucht werden. Jedoch noch nicht in der Schweiz. Das historische Museum in Basel ist also das erste Museum in der Schweiz, welches den aktuellen Augmented Reality Trend in einer Ausstellung zeigen wird.

1 Was ist Augmented und Virtual Reality?

In den nächsten Abschnitten wird erklärt was Augmented Reality (AR) und Virtual Reality (VR) ist und wie diese bereits in Museen oder im Alltag zur Anwendung kommen.

1.1 Augmented Reality

1.1.1 Was ist Augmented Reality?

Augmented Reality (AR), zu Deutsch erweiterte Realität, ist ein Mix aus der echten Welt und einer von Computern erzeugten Scheinwelt. In dieser Scheinwelt werden zum Beispiel zusätzliche Informationen zu einem Gebäude, Gemälde, Berggipfel, etc. angezeigt. Unter anderem können Videos, Grafiken, 3D-Objekte, Animationen und Texte angezeigt werden. Mit Hilfe von Smartphones, Tablets, Head-up-Displays (HUD) oder Smart Glasses kann jeder mit einer bestimmten App AR nutzen. Die Einsatzgebiete reichen von der Medizin bis zur Unterhaltungsindustrie. (DATACOM Buchverlag GmbH)

1.1.2 Welche AR Projekte gibt es bereits?

An der Digital-Life-Design Konferenz 2014 wollte Metaio präsentieren, was bereits mit ihrer Software möglich ist. Um dies zu realisieren haben sie beim Bayrischen Nationalmuseum um eine Partnerschaft angefragt. So geschah es, dass fünf Kunstwerke durch Metaio erweitert wurden. Diese konnten mit Google Glass oder iPads besichtigt werden. Von den verschiedenen Kunstwerken wurde zuerst ein 3D Model erstellt. Dieses wurde in einem weiteren Schritt mit Texten, Bildern, Audios und Videos erweitert.

Zum Beispiel wird ein Model der Stadt München aus dem Jahre 1570 erweitert durch eine aktuelle Karte der Stadt. So ist leicht erkennbar was sich im Laufe der Zeit verändert hat und wie sich die Stadt entwickelt hat. (Schleeh, 2014)

Weltweit gibt es erst eine paar einzelne weitere Museen, die eine AR-Ausstellung geplant oder durchgeführt haben. Eines davon ist das Smithonian. Seit Anfang 2015 kann im *National Museum Of Natural History* in Washington D.C. teile der Knochenhalle mit einem iPhone oder iPad neu erlebt werden. Sobald eines der 13 Skelette erkannt wurden, können verschiedene Animationen, Videos und Aktivitäten angeschaut werden. So wird zum Beispiel aus den Knochen eines Schwertfisches ein lebendiger Fisch oder eine Vampirfledermaus beginnt sich aufzustellen und läuft davon. Ausserdem sind in der App verschiedene Spiele

integriert. Beispielsweise muss eine Fledermaus nach seinem Geräusch identifiziert werden. (Sabella & Lavery, 2015)

AR wird jedoch nicht nur in Museen eingesetzt. AR wird immer mehr in den Alltag integriert. So kommt es auch das immer mehr gedruckte Kataloge durch AR erweitert werden. Einer davon ist der Katalog vom Möbelhersteller IKEA. Die Möbel aus dem Katalog lassen sich einfach mit der App ins Wohnzimmer bringen. Dazu muss im Katalog zuerst ein Möbelstück ausgewählt werden und mit der App die Seite gescannt werden. Danach muss der Katalog auf dem Boden platziert werden. Durch die Referenzgrösse des Katalogs weiss die App wie gross zum Beispiel das Sofa in der Wohnung dargestellt werden soll. Die Möbel lassen sich so beliebig platzieren und mit anderen vergleichen. Auch lässt sich das Möbelstück drehen und von allen Seiten betrachten. (Ridden, 2013)

1.1.3 Forschung

An der Technischen Universität in München wird geforscht, wie man AR mit chirurgischen Eingriffen verbinden kann. Das Ziel dabei ist, bei Operationen die Bilder aus einem Computertomografen direkt über der Eingriffsstelle anzuzeigen. Denn bisher werden die Bilder auf einem separaten Bildschirm angezeigt. Der Chirurg muss also jedes Mal seinen Kopf drehen, damit dieser die Bilder betrachten kann.

Getestet wird mit Medizinern von der Chirurgischen Klinik Innenstadt der Ludwig-Maximilians-Universität. Diese geben den Entwicklern ein ständiges Feedback, damit das System den Anforderungen angepasst werden kann. Als Testobjekt dienen zu Beginn Puppen und Tierkadaver. Sobald die Technik ausgereifter ist, wird an lebenden menschlichen Körpern versucht. (Fakultät für Informatik)

1.2 Virtual Reality

1.2.1 Definition und Ursprung

Die virtuelle Realität (VR) ist eine künstliche Umgebung, welche von Computern erzeugt wird. Diese erzeugte Realität kann Bezug zur echten Realität haben, ist aber nicht zwingend. Mit Hilfe von VR-Geräten werden das Sehen und das Hören angesprochen. Durch das „täuschen“ dieser Sinne glaubt sich der Nutzer in einer Scheinwelt. Die echte Welt wird dadurch ausgeblendet. (Rouse, 2009)

VR ist keine neue Erfindung, denn die ersten Projekte entstanden bereits 1962. Jedoch zeigten die ersten Prototypen nur kurze Filme. In den 1990er Jahren gab es einen regelrechten Boom um VR. Diese neuere Generation von VR stand im Zusammenhang mit den damaligen Heimkonsolen. Allerdings erreichten diese Geräte nie den Massenmarkt, weil die damalige Hardware zu teuer und unausgereift für die Allgemeinheit war. Ausserdem war die Programmierung zu kompliziert, was dazu führte, dass es nur sehr wenige passende Spiele gab. Einzig in Japan konnte sich VR in Spielhallen halten. Ausserhalb der Spielewelt wurde ab den 1990ern VR für die Steuerung von Industrierobotern, beim Training von Militärs sowie in der Architektur eingesetzt. (Bonset, 2014)

1.2.2 Wie wird VR heute eingesetzt?

Architekten und Immobilienverkäufer haben das Potential von der heutigen VR-Technologie erkannt. Das Schweizerunternehmen Archilogic erstellt durch intelligente Algorithmen 3D Modelle aus Grundrissplänen eines Hauses oder einer Wohnung. Diese Modelle können dann mit Oculus Rift oder Google Cardboard angeschaut werden. So kann der Hausbauer bzw. -käufer sich ein Objekt schon vorab besichtigen und bereits virtuell einrichten. (Schuppisser, 2015)

Auch Spieleentwickler nutzen VR in ihren Smartphonespielen. Daher gibt es bereits einige VR-Spiele im Play- und App Store. Diese können mit Google Cardboard oder der Samsung Gear gespielt werden. Für die VR-Brillen Oculus Rift und HTC (Re) Vive gibt es bereits für die Entwicklerversionen verschieden Demospiele zum Testen. Diese geben erste Eindrücke zu der neusten VR-Generation, sie sind jedoch noch nicht ausgereift für den Massenmarkt. (Häntschi, 2015)

Viele Personen, welche eine VR-Brille auf hatten, beklagen sich über ein Unwohlsein nach einer bestimmten Nutzungsdauer. Dieses Unwohlsein wird als Motion Sickness, zu Deutsch Reisekrankheit, bezeichnet. Zu dieser Übelkeit kommt es, weil die Augen andere Informationen liefern, als gefühlt werden. Zum Beispiel sitzt ein Spieler virtuell in einer Achterbahn. Die Augen sehen in diesem Fall eine Bewegung, doch andere Organe melden dem Gehirn, dass man immer noch still auf seinem Stuhl sitzt. Abhilfe soll eine virtuelle Nase schaffen. Diese Nase wird im unteren mittleren Bereich des Bildschirms eingeblendet. Sie soll dem Gehirn einen ständigen Anhaltspunkt geben und so die Intensivität der Übelkeit minimieren. In den durchgeführten Tests ist den Probanden die künstliche Nase nicht aufgefallen oder hat zumindest nicht gestört. (Whittinghill, 2015)

2 Methodik

Die Arbeit wurde in zwei grössere Phasen eingeteilt. In der ersten Phase ging es darum die optimale Software und Tools für den Prototyp zu finden. In der zweiten Phase wurde der Prototyp entwickelt.

2.1 Recherche- und Analysephase

In der ersten kürzeren Phase werden Analysen und Recherchen zu Software Development Kits (SDK) und verschiedenen VR-Brillen durchgeführt. Diese werden darauf hin analysiert. Auf den Analysen basierend werden Entscheide für die Implementierungsphase gefällt.

2.2 Implementierungsphase

Die Implementierungsphase ist der wichtigere Aspekt für diese Arbeit. In dieser zweiten Phase wird ein Prototyp erstellt, welcher in einem späteren Zeitpunkt von Antoine Widmer und Roger Schaer weiterentwickelt werden kann. Die endgültige Version soll 2016 im Historischen Museum in Basel zur Anwendung kommen.

3 Recherche- und Analysephase

3.1 Augmented Reality

Das Ziel dieses Kapitels ist die passende AR SDK für den Prototyp festzulegen. Dabei gilt es einige Museumseigenschaften zu berücksichtigen.

3.1.1 Auswahl der SDKs

Die folgenden SDKs wurden ausgewählt, weil diese am häufigsten eingesetzt wurden und gemäss der Auflistung von Davis & Community von Socialcompare die meisten Funktionen anbieten.

- Metaio
- Wikitude
- Vuforia
- Layar
- ARmedia
- Xludia

3.1.2 Vergleichskriterien

Da die App in einem Museum verwendet wird, gibt es ein paar Einschränkungen zu beachten. Nach den folgenden Kriterien wurden die verschiedenen SDKs untersucht:

- Auf welchen Plattformen läuft die SDK? Wichtig sind hier dass Android und iOS verfügbar sind.
- Was kosten die verschiedenen SDKs?
- Wie benutzerfreundlich sind SDKs?
- Wie wird ein neues Bild oder 3D-Objekt hinzugefügt?
- Unter welchen Aspekten wird ein Bild erkannt?
- Unter welchen 3D-Objekte?
- Gibt es Einschränkungen die zu beachten sind?
- Können Videos abgespielt werden?
- Was passiert wenn das Bild durch Personen verdeckt wird?
- Haben schlechte Lichtverhältnisse einen Einfluss auf die Erkennung?

3.1.3 Vergleich

ARmedia und Xloundia konnten nicht getestet werden, weil diese ihre Trial-Versionen nur unter Anfragen herausgeben. Da der Entscheid aber relativ früh gefällt werden musste, konnte nicht auf die Antwort der beiden Lieferanten abgewartet werden.

Getestet wurde mit einem Bild der Mona Lisa, ausgedruckt auf ein A4 Blattpapier und als 3D-Objekt wurde die Maus Roccat Pyra verwendet. Als Ersatz, falls die beiden Objekte nicht geeignet sind, diente ein Gemälde von Leonardo Da Vinci und eine Kaugummiverpackung von Oxycool. Die Apps wurden auf ein Samsung Galaxy S4 geladen und getestet.

Bemerkungen zur Tabelle 1

Die Preise sind nicht vollständig abgebildet. Aus Platzgründen sind nur eine Trial/Starterversion sowie das billigste und teuerste Produkt aufgelistet. Die verschiedenen Preisklassen unterscheiden sich in der Menge der Funktionen bzw. besseren Support.

Beim Testen von Vuforia war das Problem, dass nur Ecken und Kanten erkannt werden. Deswegen mussten auf die Ersatzgegenstände ausgewichen werden. Auch bietet Vuforia kein eigenes Interface an, aus diesem Grunde musste Unity benutzt werden.

Tabelle 1: Augmented Reality SDK Vergleich. (Eigene Darstellung) Quelle: Siehe Tabellenende

	Metaio		Wikitude		Vuforia		Layar	
	Beschreibung	Wert	Beschreibung	Wert	Beschreibung	Wert	Beschreibung	Wert
Betriebs-system	Android, iOS, Windows, Tizen	10	Android, iOS, Smart Glasses	10	Android, iOS, Smart Glasses	10	Android, iOS, BlackBerry	10
Preis	Trial, Basic 4'500€/Jahr, Pro 6'500€/Jahr	8	Trial, Light 590€, Pro 4'490€/Jahr	8	Starter, Classic 499\$, Cloud 999\$/Monat	8	Trial, Basic 3\$/Seite, Premium 300€/Monat	8
Benutzer-freundlich	Drag n Drop	10	Drag n Drop, Webbrowser	10	Drag n Drop, Unity	7	Drag n Drop, Webbrowser	10
Bild hinzufügen	Drag n Drop	10	Drag n Drop	10	Upload, Library download, Import	4	Drag n Drop	10
3D Objekt hinzufügen (Ziel)	Import, Objekt scannen	10	Funktioniert nicht in Trial	0	Objekt scannen, gleiche Schritte wie Bild hinzufügen	8	Wird nicht unterstützt	0
Bild Erkennung	Alle Winkel, Teilausschnitt reicht	10	Alle Winkel (nicht sehr steil), Teilausschnitt: 1/2 Bild	6	Alle Winkel, Teilausschnitt: (1/4 - 1/2)	10	Alle Winkel, Teilausschnitt: (1/2)	8
3D Objekt Erkennung	Umgebung wird auch gescannt, gleicher Winkel	7	Funktioniert nicht in Trial	0	Alle Winkel/Seiten erkannt	10	Wird nicht unterstützt	0
Videos	Möglich	10	Möglich	10	Möglich	10	Möglich	10
Bild verdeckt	AR verschwindet	8	AR verschwindet	8	AR verschwindet	8	AR verschwindet langsam, Fixieren auf Bildschirm, Anzoomen	10
Einschränkungen	Max. ca. 1m Abstand	10	Max. ca. 1m Abstand	10	Ersatzbilder (nur Kanten & Ecken), Grösse begrenzt, Max 50cm Abstand (3D)	6	Max. ca. 1m Abstand, Angebotene Funktionen sehr eingeschränkt	6
Lichtverhältnisse	Solange Bild beleuchtet wird	10	Solange Bild beleuchtet wird	10	Solange Bild beleuchtet wird	10	Solange Bild beleuchtet wird	10
		103		82		91		82
Quelle:	(Metaio GmbH, 2015), Daten durch Autorin gesammelt		(Wikitude GmbH, 2015), Daten durch Autorin gesammelt		(Qualcomm Connected Experiences Inc), Daten durch Autorin gesammelt		(Layar), Daten durch Autorin gesammelt	

3.1.4 Tests mit PixLive

PixLive wurde von der Schule als AR-Tool vorgeschlagen, weil Antoine Widmer persönlichen Kontakt zu Vidinoti hat. Um die Funktionalität sicherzustellen, wurden die gleichen Tests wie bei den anderen SDKs durchgeführt.

Gleich zu Beginn der Tests ist aufgefallen, dass PixLive 3D-Objekte nicht erkennt, da nur Bilder oder iBeacons auf die Plattform hochgeladen werden können. Mit PixLive Maker kann das Bild einfach per Drag n Drop erweitert werden. Die AR-Gestaltung fällt leicht mit dem Maker, jedoch ist die Integration auf ein Smartphone sehr schwierig – zumindest für Anfänger – und es lässt sich nicht gleich intuitiv bedienen, wie die anderen getesteten SDKs. Die oben genannten Tests konnten nicht durchgeführt werden, weil es nicht möglich war, die AR-Funktion auf das Smartphone zu bringen.

Ein Nachteil dieser SDK ist, dass sie nur kommerziell vertrieben wird. Dadurch arbeitet eine viel kleinere Community mit PixLive als beispielsweise mit Wikitude. Eine grössere Community bietet mehr Tutorials, Guides und schnelle Antworten auf verschiedenen Plattformen im Internet.

3.1.5 Entscheid: Passende SDK auswählen

Gemäss den durchgeführten Tests und dem daraus resultierenden Ergebnis hätte ich mich für die SDK von Metaio entschieden. Weil diese SDK sehr einfach zu benutzen ist, alle gewünschte Funktionen enthält und die zwei Testobjekte ohne Probleme erkannt wurden. Ausserdem können 3D-Objekte von Blender – Software zur Modellierung von 3D-Objekten - direkt als Ziele integriert werden. Somit muss nicht immer das 3D-Objekt mit der App gescannt werden. Jedoch wurde Metaio in der Zwischenzeit von Apple aufgekauft und eingestellt. Aus diesem Grund musste ich eine weitere Entscheidung fällen.

Meine zweite Entscheidung fällt auf Vuforia von Qualcomm. Es ist einfach zu benutzen, nach einer gewissen Einarbeitungszeit, und unterstützt 3D-Modelle. Des Weiteren kann mit Vuforia der Bildschirm für jedes Auge aufgeteilt werden, damit die App mit Google Cardboard verwendet werden kann.

3.2 Virtual Reality

Dieses Kapitel befasst sich mit den verschiedenen VR-Brillen. Dabei geht es darum eine Übersicht über kommende bzw. bestehende Geräte zu erhalten.

3.2.1 Auswahl der Head-Mounted-Display Geräte

Oculus Rift, Google Cardboard, Samsung Gear VR, Project Morpheus und (Re) Vive sind aktuell die am meisten diskutierten und ersehnten VR-Geräte. Einzig Cardboard und Gear VR sind aktuell auf dem Markt verfügbar. Die restlichen haben bisher einen Prototypen vorgestellt und / oder bieten eine Entwicklerversion an.

3.2.2 Vergleichskriterien

Damit sich die einzelnen Geräte gut vergleichen lassen, wurden folgende Vergleichskriterien aufgestellt.

- Welcher Hersteller bietet die Brille an?
- Ab wann sind die Geräte verfügbar?
- Was sind die (geschätzte) Kosten und zusätzliche Kostenfaktoren?
- Wie gut ist die Bildschirmqualität (Auflösung, Refresh rate)?
- Was für Sensoren sind eingebaut?
- Wie gross ist das Gesichtsfeld?

3.2.3 Vergleich

Bemerkungen zur Tabelle 2

¹ entspricht dem geschätzten Preis.

Die Daten der Oculus Rift entsprechen der Entwicklerversion 2. In der zwischen Zeit wurde die finale Version an der Electronic Entertainment Expo (E3) in Los Angeles vorgestellt.

Mittlerweile bietet Samsung seine Gear VR auch für das Smartphone Galaxy S6 an.

Tabelle 2: Vergleich Virtual Reality Head-Mounted-Display Geräte. (Eigene Darstellung) Quelle: Siehe Tabellenende

	Oculus Rift	Cardboard	Gear VR	Project Morpheus	(Re) Vive/ Steam VR
Hersteller	Oculus VR (Facebook)	Google	Samsung	Sony	HTC & Valve
Preis 19.05.2015	Entwicklerversion 350\$ Release: 350\$ ¹	Ab 3\$	Note 4: Ab 750 \$ Gear VR: 199\$	Release: 300\$ ¹	Entwicklerversion gratis Release: 500\$ ¹
Release	1. Quartal 2016	2014	2014	1. Quartal 2016	4. Quartal 2015
Zusätzliche Geräte nötig?	Positional Tracking	Smartphone	Samsung Galaxy Note 4	Dual Shock-Controller Move-Controllern PlayStation 4	Tracking-System von Valve
Bildschirm-auflösung	960 x 1080px pro Auge	Smartphone abhängig	2.560 x 1.440px	1920 x 1080px	1200 x 1080px
Refresh rate	75Hz		60Hz	120Hz	90Hz
Sensoren	Gyroskop Accelerometer Magnetometer		Accelerometer Gyroskop Magnetometer Annäherungsschalter	Bewegungssensor Accelerometer	Gyroskop Accelerometer Laser zur Positionsbestimmung
Gesichtsfeld	100°		96°	90°	unbekannt
Quellen	(Oculus VR & Oculus VR, LLC, 2015) (Kneussel, GIGA Digital AG, 2015) (Winchester, 2015)	(Google Inc) (Heil, 2015) (Winchester, 2015)	(Samsung Electronics Switzerland GmbH) (Mobilzone Holding AG) (Oculus VR & Oculus VR, LLC) (Winchester, 2015)	(Fauster & Yoshida, 2015) (IDG Entertainment Media GmbH) (Shepherd & Mccallion, 2015) (Dunning, 2014) (Winchester, 2015)	(HTC CORPORATION) (Valve Corporation) (Kneussel, GIGA Digital AG, 2015) (Winchester, 2015)

3.2.4 Entscheid

Da die App noch dieses Jahr fertig gestellt werden soll und nächstes Jahr im Einsatz steht, kommen nur die Gear VR von Samsung und Cardboard von Google in die nähere Auswahl. Aus Kostengründen entscheide ich mich für Google Cardboard, weil ich aktuell kein Samsung Galaxy Note 4 besitze und das Historische Museum sich diese Geräte ebenfalls beschaffen müsste. Ausserdem haben schon viele Museums Besucher ihre eigene Smartphones, welche mit Cardboard benutzt werden können.

3.2.5 Cardboard und 360° Fotos

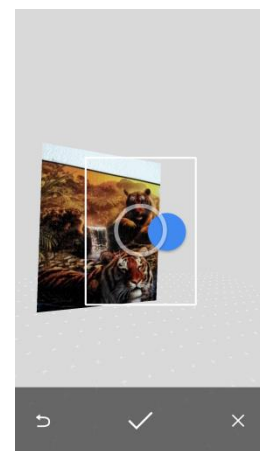
Mit Hilfe der Google Camera auf Androidgeräten oder Photo Sphere Camera auf iOS lassen sich ganz einfach 360° Fotos aufnehmen. Wie auf der Abb. 2 ersichtlich, muss ein Foto nach dem anderen geschossen werden. Sobald der blaue Kreis innerhalb des weissen Ringes ist, wird automatisch ein Foto geknipst. Dies wird so lange wiederholt bis alle Punkte fotografiert sind. Für ein optimales Ergebnis sollte die Höhe und Neigung immer gleich sein.

Statt das 360° Foto selber zu erstellen, können auch Bilder aus anderen Quellen benutzt werden. Entweder werden die Bilder auf das Gerät hoch geladen oder können direkt von einem Webserver abgerufen werden.

In der Demo App von Google Cardboard ist bereits die Photosphere integriert. Diese greift auf die 360° Fotos auf dem Gerät zu. Mit der Neigung und Kopfbewegung wird die Ansicht auf das Foto verändert. (Reeve, 2014)

3.3 360° Videos

Damit ein 360° Video entstehen kann braucht es mehrere Kameras, welche verschiedene Blickwinkel filmen. Aktuell werden die meisten 360° Videos mit Hilfe von GoPros und einer dafür eigens konstruierten Halterung erstellt. Einige junge Unternehmen wie zum Beispiel 360fly Inc. haben Prototypen entwickelt, welche in den nächsten Jahren auf dem Markt erhältlich sind. Diese Geräte haben die verschiedenen ausgerichteten Kameras festinstalliert und können in den Alltag integriert werden (K.L., 2015)



**Abb. 2: Screenshot
Google Camera
(Eigene Darstellung)**

Mit Unity 5 können 360° Videos einfach in Google Cardboard integriert werden. Die Videos werden in sogenannten Sphären dargestellt. Damit der Benutzer das komplette Video anschauen kann, muss die Kamera immer im richtigen Winkel zu den Sphären sein. (K.L., 2015)

4 Verwendete Tools

4.1 Vuforia

Qualcomm wurde 1985 gegründet mit der Idee Kommunikation mit Qualität anzubieten. Sie bieten viele verschiedene Produkte an, eines davon ist Vuforia. Vuforia ist eine AR SDK für Apple- und Androidgeräte. Vuforia kann mit Xcode, Eclipse / Android Studio und Unity verwendet werden. Die SDK erkennt Bilder, 3D-Objekte, Text, Zylinder, Würfel und spezielle Marker. In der Anwendung wird die aktuelle Version 4.2.3 verwendet. (Qualcomm Technologies Inc, 2015)

4.2 Android Studio

Android Studio ist eine integrierte Entwicklungsumgebung (IDE) für die Applikationsentwicklung für Androidgeräte. Android Studio basiert auf IntelliJ IDEA, welches zu Eclipse konkurrenziert. Android Studio bieten einen Emulator zum Testen der erstellen Apps an. Es kann jedoch auch mit dem eigenen Androidgerät getestet werden. Android Studio ist kostenlos erhältlich. Es wird für den Prototyp die aktuelle Version 1.2.1.1 verwendet. (Google Inc)

4.3 Google Cardboard

Cardboard ist eine Karton-VR-Brille von Google. Diese Brille ermöglicht allen für wenig Geld VR zu erleben. Cardboard wird zusammen mit dem eigenen Smartphone benutzt (Vergleiche Abb. 3). Die Schachtel muss vor Beginn selber zusammen gebaut werden. Die mitgelieferten Linsen können ersetzt werden, durch Korrigierte. (Missfeldt)



Abb. 3: Google Cardboard (Quelle: <http://thewestsidestory.net/wp-content/uploads/>)

4.4 Unity

Unity 3D ist eine Engine für 2D-, 3D- und Grafikanwendungen. Mit Unity können vor allem Spiele für Webbrowser, Computer, Smartphones und Konsolen entwickelt werden. Vorteile gegenüber anderen Engines sind: Billigere Lizenz, schnelle und stabile Ergebnisse für Browser und Smartphones. (Steinlechner, 2012)

Unity 3D wurde ausgewählt, weil Google Cardboard und Vuforia ihre SDKs dafür anbieten, grafisches Arbeiten ermöglicht und es eine kostenlose Version gibt.

4.5 Ninjamock

Ninjamock ist ein kleines Startup aus Dänemark. Der gleichnamige Mockupdesigner lässt sich einfach über den Browser bedienen. Mit Ninjamock können Mockups für Android, iPhone, Webseiten und weitere erstellt werden. Die Mockups werden wie Skizzen dargestellt, damit für den Kunden klar ersichtlich ist, dass z.B. die Android App noch nicht fertiggestellt ist. Es gibt eine eingeschränkte Gratisversion sowie zwei weitere Bezahlversionen. Es wurde die Gratisversion zum Erstellen der Mockups verwendet. (Ninjamock)

4.6 Erste Versuche mit Vuforia und Cardboard

Unity war zu Beginn eine unbekannte Software. Um die Funktionalitäten dieser Software kennen zu lernen, wurden verschiedene Tutorials durchgeführt, Videos angeschaut sowie Guides durchgelesen. Es wurde auch explizit nach Tutorials gesucht, welche Vuforia in Google Cardboard verwenden. Es gibt verschiedene Lösungsansätze, jedoch gibt es zu keinem Ansatz ein Tutorial.

5 Erasmus – Prototyp App

5.1 Mockups

Zur Vorbereitung auf den ersten Besuch in Basel wurden Mockups mit Ninjamock erstellt. Diese ersten Mockups wurden anhand von schriftlichen Angaben von Marcel Henry angefertigt. Vor Ort wurden dann die Mockups dem Kunden vorgestellt. Dieser war von diesen ersten Bildern begeistert.

Als nächstes wurde zusammen mit dem Kunden die Mockups erweitert durch weitere Wünsche des Kunden. Die Abb. 4 zeigt ein Mockup der Sprachauswahl. Die Mockups dienten als Vorlage für die verschiedenen Activities des Prototyps. Als Activity werden die verschiedenen Bildschirme in Android bezeichnet. Auf Abb. 5 ist die fertige Ansicht zusehen. Alle Mockups sind im Anhang III: Mockups aufgelistet.

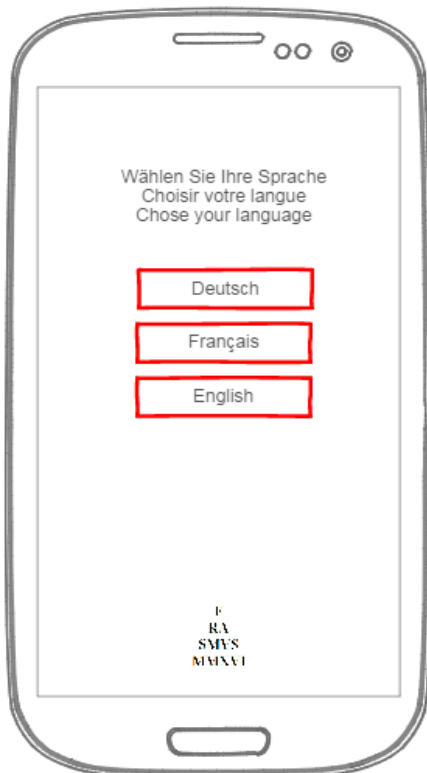


Abb. 4: Mockup Sprachauswahl (Eigene Darstellung)

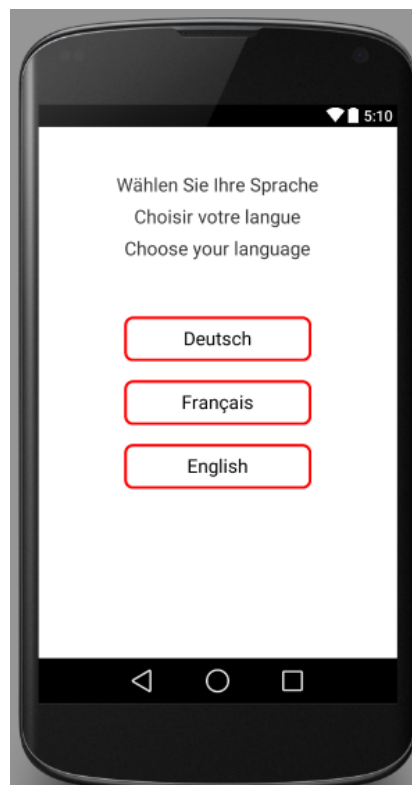


Abb. 5: Screenshot Android Studio Sprachauswahl (Eigene Darstellung)

5.2 Wie entstand dieser Prototyp?

Zunächst mussten die Tools (Vergleiche mit 4 Verwendete Tools) installiert und konfiguriert werden. Um Unity und Vuforia kennen zu lernen und sich ein Basiswissen anzueignen, wurden mehrere Tutorials durchgeführt, bzw. angeschaut. Mit Android Studio wurden die Mockups in einen funktionierenden Prototyp umgewandelt.

5.2.1 Wie erkennt Vuforia die verschiedenen Bilder / QR Codes?

Als erstes muss auf der Entwicklerwebseite von Vuforia ein Lizenschlüssel erstellt werden. Beim Erstellen dieses Schlüssels können je nach Art des Schlüssels Kosten anfallen. Für den Prototyp wurde die Starterversion ausgewählt. Diese ist kostenlos, wobei ein Wasserzeichen auf dem Bild eingeblendet wird.

Als nächstes muss eine Datenbank erstellt werden. In diese Datenbank werden alle Zielobjekte hochgeladen. Als Ziel kann ein Bild, ein Zylinder, ein Quader oder ein 3D-Objekt hochgeladen werden. Wobei ein Bild nicht grösser als 2MB gross sein kann und das 3D-Objekt mit dem *Vuforia Object Scanner*, eine spezielle App zum Scannen von 3D Objekten, von Qualcomm gescannt werden muss.

Sobald ein Ziel hochgeladen wird, wird dieses von Vuforia analysiert und bewertet. Jedem Bild werden sogenannte „Features“ zugewiesen. Auf Abb. 6 sind die Features in Gelb zu sehen. Je nach Anzahl dieser Features erhält das Bild eine bessere Bewertung. Maximal werden fünf Sterne vergeben. Ein Ziel mit weniger als drei Sternen wird nicht oder nicht immer erkannt. Die von Marcel Henry zur Verfügung gestellten QRCS haben alle eine Bewertung von fünf Sternen. Die QRC sind im Anhang V: QRC zu finden.

Des Weiteren muss die eben erstellte Datenbank heruntergeladen werden. Beim Herunterladen kann aus den verschiedenen Entwicklerplattformen und Unity ausgewählt werden. In Unity lässt sich diese ganz einfach per Doppelklick einbinden.

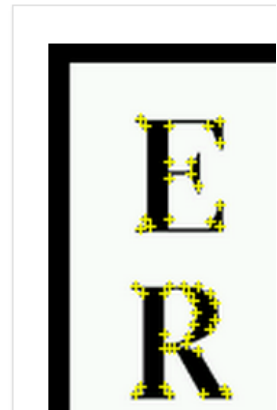


Abb. 6: Screenshot von Features auf einem Zielbild (Eigene Darstellung)

5.2.2 Wie werden diese QRCs erweitert?

Da dieser Teil des Prototyps mit Unity erstellt wurde, wird nicht auf andere Entwicklungsplattformen verwiesen. Nachdem die Datenbank und die Vuforia Erweiterung importiert sind, kann mit der grafischen Erweiterung der Zielobjekte begonnen werden.

Als erstes wird die Standard Kamera durch die *ARCamera* von Vuforia ersetzt. Diese ermöglicht das Erkennen der Zielobjekte. Dieser Kamera muss ein Lizenzschlüssel hinzugefügt werden. Des Weiteren muss in der *ARCamera*-Einstellungen eine Datenbank ausgewählt und aktiviert werden.

Im nächsten Schritt werden die Zielobjekte eingefügt. Beispielsweise wird ein Bild mit dem mitgelieferten Asset *ImageTarget* dargestellt. Dem *ImageTarget* muss mitgeteilt werden, welche Datenbank und welches Objekt benutzt werden soll.

Als nächstes können die Targets grafisch erweitert werden. Im Prototyp wurden die Targets erweitert durch Bilder, Videos, Musik, 3D-Modellen und einer Photosphere.

Bild anzeigen

Weil normale Bilder nicht unterstützt werden, muss das Bild zunächst in ein Sprite umgewandelt werden. Sprites sind einfache 2D Objekte von Unity. Das Sprite wird in einem nächsten Schritt über dem Zielobjekt platziert. Die Grösse, Rotation und Abstand zum Zielobjekt können eingestellt werden.

Musik wiedergeben

Um Musik abspielen zu können, braucht es eine *Audio Source*. Diese *Audio Source* steuert den Ton über die Lautsprecher des Geräts. Ausserdem können verschiedene Effekte eingestellt werden. Eine Audio Source ist kein Objekt, das in Unity irgendwo platziert werden kann, es muss dafür einem *Game Objekt* hinzugefügt werden. Das passende Lied kann direkt in der *Audio Source* hinzugefügt werden, dies setzt aber voraus, dass das Audiofile in Unity importiert wurde.

Es ist aber auch möglich die Datei zu streamen. Dazu benötigt es aber das folgende Script:

```
private void LoadContent(string path)
{
    // get the content from web or directory
    WWW www = new WWW(path);

    audio = GetComponent<AudioSource> ();
    audio.clip = www.GetAudioClip (true, true);

}
```

WWW stellt eine Verbindung zu der angegebenen URL her und fängt an das gefundene Objekt zu downloaden. Der lokalen *Audio Source audio* wird das vorhandene *Game Objekt* zu gewiesen. Damit der Ton auch wirklich abgespielt werden kann. Mit *GetAudioClip* wird das heruntergeladene Objekt der *Audio Source* übergeben. In den Parametern kann eingestellt werden, ob gestreamt werden soll und ob der Ton in 3D wiedergegeben werden soll. 3D bezeichnet in diesem Fall, dass die Distanz und Richtung zu dem Objekt beachtet wird und dem entsprechend der Ton wiedergegeben wird.

Video abspielen

Es gibt zwei Arten ein Video mit Unity abzuspielen. Einerseits kann das Video importiert werden und als *Movie Texture* einem *Game Objekt* hinzugefügt werden. Das Video kann auf diese Weise auf der ganzen Fläche verschoben werden und in der Grösse angepasst werden. Diese Variante benötigt jedoch zum importieren Apples Quicktime.

Andererseits kann das Video direkt auf dem Smartphone per Fullscreen abgespielt werden. Mit dieser Variante wird das Video direkt im Vollbildmodus auf das Smartphone gestreamt. Das Video kann mit einer URL oder Dateipfad verlinkt sein. Falls das Video in eine künstliche Umgebung eingefügt werden soll, ist diese Variante nicht geeignet.

3D-Modell anzeigen

3D-Modelle können selber mit geeigneter Software erstellt werden, im *Asset Store* von Unity gekauft werden oder von anderen Seiten importiert werden. Bei den eigenen bzw. von anderen Seiten geladenen Modellen ist darauf zu achten, dass das jeweilige Format von Unity unterstützt wird. Sobald das Modell importiert wurde, kann es frei auf der Arbeitsfläche von Unity platziert werden. Auch hier kann die Grösse und Rotation eingestellt werden. Durch spezielle Tools lassen sich die Modelle animieren. Die Animationen lassen sich wiederum in Unity steuern.

Photosphere anzeigen

Es gibt verschiedene Arten eine Photosphere in Unity einzubauen. Eine einfach Variante ist das Unity Plugin von Cardboard zu benutzen. Durch dieses Plugin wird das Bild direkt auf zwei Bilder aufgeteilt. Ausserdem werden die Bewegungen des Smartphones direkt auf das Bild übertragen.

Das Panoramabild muss jedoch zu Beginn noch bearbeitet werden. Als erstes muss das Bild in Unity importiert werden. Als nächstes muss der *Texture Type* auf *Cubemap* gewechselt werden sowie das *Mapping* auf *zylindrisch* geändert werden. Das Konvertieren kann je nach Leistung des Rechners länger dauern. Als nächstes muss ein neues *Material* erstellt werden. Bei diesem *Material* muss der *Shader* auf *Skybox > Cubemap* gewechselt werden. Nun kann dem *Material* das Foto zugewiesen werden. Im letzten Schritt muss die *Skybox* im Hintergrund mit dem erstellen *Material* ausgetauscht werden. Diese Anleitung stammt von Felix (2015).

5.2.3 Wie wird die Unity App mit der Normalen App verbunden?

Wenn die App nur aus der AR-Funktion besteht, kann diese nun getestet werden und in den Playstore bzw. App Store geladen werden. Da dieser Prototyp aber weitere Funktionen anbietet, müssen diese auch noch integriert werden.

Normale App

In Android Studio wurden unter anderem die Funktion der Sprachenauswahl sowie der Streamingauswahl implementiert. Jede Activity wurde auf die gleiche bzw. ähnliche Art erstellt. Zunächst wurden die Activities aus einem leeren Template erstellt. Diese enthält eine Referenz auf die Parentactivity, um auf diese mit dem Backbutton zurückzukehren. Danach wurde das Layout so bearbeitet, dass es so ähnlich wie möglich wie das Mockup aussieht. Des Weiteren wurden die Logik implementiert, z.B. die gewählte Sprache speichern und anwenden. Es enthält jedoch nicht jede Activity eine Logikkomponente. Beispielsweise hat die Karte vom Museum keine weitere Funktion beim Anklicken, als die Etage zu wechseln. Im Idealfall sollten die roten Flächen, bei einer Berührung, eine Beschreibung anzeigen.

Wie werden diese zwei Apps miteinander verbunden?

Es gibt zwei Lösungsvorschläge. Einerseits wird die App aus dem Android Studio in ein JAR-File exportiert. Diese Datei wird dann in Unity importiert. Dadurch erhält Unity Zugriff auf die einzelnen Activities.

Andererseits kann in Unity das Projekt in ein Eclipse Projekt exportiert werden. Dazu muss in den *Build Settings* ein Haken für *Google Android Project* ausgewählt werden. Beim Exportieren wird ein Projekt an dem angegebenen Pfad erstellt. Dieses lässt sich dann in Android Studio importieren und mit dem vorhandenen Projekt verbinden. Dadurch erhält Android Studio Zugriff auf die implementierten AR-Funktionen von Unity.

5.3 Wie funktioniert die App?

Als erstes muss die App aus dem Playstore bzw. App Store heruntergeladen werden. Nach der Installation öffnet sich die App. Auf der Startseite ist das Logo der Ausstellung zu sehen (Vergleiche mit Abb. 7). Von diesem Bildschirm (Activity) kann man per Klick auf das Logo auf die nächste Seite gelangen oder über die drei Punkte oben rechts die Einstellungen aufrufen.

In den Einstellungen (Siehe Abb. 8) kann die Sprache auf Deutsch, Französisch oder Englisch gewechselt werden. Des Weiteren kann ausgewählt werden, ob der komplette Inhalt zuerst auf das

Smartphone geladen wird oder bei gebrauch gestreamt wird. In der nächsten Option kann die Cardboardansicht (zwei Kamerabilder) ein- bzw. ausgeschaltet werden. Unter Museum

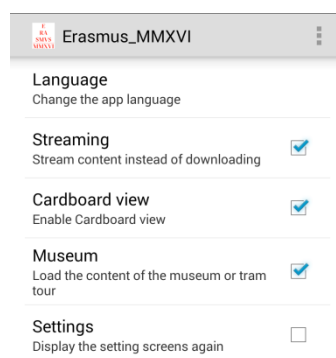


Abb. 8: Screenshot Einstellungen (Eigene Darstellung)

kann gewählt werden, ob man sich gerade im Museum oder in der Tramtour befindet. In der letzten Einstellung kann gewählt werden, dass die Einstellungsbildschirme noch einmal angezeigt werden sollen. Normalerweise werden die Einstellungsbildschirme nur beim ersten Benutzen angezeigt.



Abb. 7: Screenshot Startseite (Eigene Darstellung)

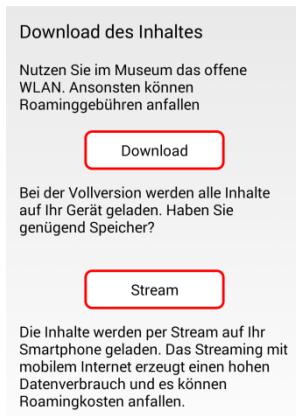


Abb. 9: Screenshot Download (Eigene Darstellung)

Option wird der Inhalt bei dessen gebrauch geladen. Diese Variante kann aber vor allem für ausländische Besucher wegen hohen Roaminggebühren teuer werden.

Auf dem letzten Einstellungsbildschirm ist die BildschirmEinstellung zu sehen (Vergleiche mit Abb. 10). Mit Klick auf das Cardboardlogo oder auf den Cardboardbutton wird die AR-Funktion auf dem Bildschirm dupliziert dargestellt. Ein Bild für das linke Auge, das andere für das Rechte. Bei Klick auf das Tablet oder den Tabletbutton wird das Bild auf dem ganzen Bildschirm angezeigt.

So sollte es am Schluss sein. Da es aber Probleme bei der duplizierten Darstellung gab, wird momentan das Bild immer als Tabletansicht angezeigt (Vergleiche mit *Keine Integration von Cardboard in Vuforia*).

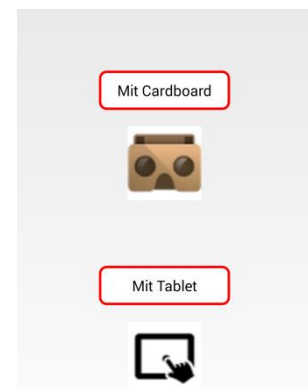


Abb. 10: Screenshot BildschirmEinstellung (Eigene Darstellung)

Die App verfügt nicht nur über Einstellungsbildschirme, sondern auch über andere nützliche Activities. Eine Karte des Museums zeigt die verschiedenen AR und Cardboardzonen an. Auf Google Maps wird die Tramtour mit den verschiedenen Stationen angezeigt. Ein weiterer Bildschirm soll später die verschiedenen Sponsoren auflisten. Im Prototyp wird jedoch nur ein Platzhalterbild dargestellt. Aktuell sind diese drei Activities über einen Bildschirm mit Buttons aufrufbar. In der finalen App wird dieser Bildschirm durch ein Menu ersetzt. Von diesem Menu soll zusätzlich die AR-Activity aufrufbar sein.



Abb. 11: Screenshot Ladebildschirm (Quelle: Unity)

Sobald die AR-Activity gestartet wird, erscheint ein Ladebildschirm von Unity (Siehe Abb. 11). In der kostenlosen Version von Unity lässt sich dieser Ladebildschirm, der sogenannte Splash, nicht verändern. Sobald das Kamerabild geladen ist, kann ein QRC gescannt werden. Mit dem Prototyp können die QRC eins bis sieben getestet werden.

Die verschiedenen QRC sind im Anhang V: QRC zu finden.

Was zeigen die Verschiedenen QRCs an?

Das erste QRC zeigt ein Gemälde von Erasmus von Rotterdam, gemalt 1523 von Hans Holbein. Bei QRC zwei und sechs wird ein Video über Konstantin Wecker abgespielt. Ausgestrahlt von Kulturplatz auf SRF 1 am 20. Mai 2015. Einmal wird das Video gestreamt beim anderen Mal wird das heruntergeladene Video benutzt. Das dritte und fünfte QRC spielen jeweils ein Lied ab. QRC drei streamt das Lied *Born This Way* von Lady Gaga. Nummer fünf spielt *Le déserteur* von Boris Vian ab. Das vierte QRC zeigt einen Soldaten in 3D. Dieser Soldat wurde aus dem Asset Store von Unity heruntergeladen. Das Modell soll im Museum neben eine Kanone der Eidgenossen dargestellt werden. Es soll die Verbindung von damals zu heute herstellen. Das letzte QRC startet eine Photosphere. Durch die Bewegung des Smartphones kann man sich in dieser Sphäre bewegen. Das Panoramafoto wurde von Marcel Henry zur Verfügung gestellt.

Schlussfolgerung

Worin lagen die Schwierigkeiten und wie wurden diese gelöst?

Im Laufe der Bachelorarbeit tauchten immer wieder verschiedene Probleme und Schwierigkeiten auf. Dieses Kapitel widmet sich diesen Schwierigkeiten und wie diese versucht wurden zu lösen.

Neue Technologie

Zu Beginn dieser Arbeit war der Verfasserin bekannt was AR ist, jedoch waren die Schritte zur Integration in eine App unbekannt. Im Bereich VR war bereits der Name Oculus Rift geläufig, jedoch hat man sich nicht genauer damit befasst. Deswegen wurden zuerst Informationen zu den Anbietern von AR und VR gesammelt.

Die ersten AR-Erfahrungen wurden während den Tests der verschiedenen AR-SDKs gesammelt (Siehe 3.1 Augmented Reality). Dabei fiel recht schnell auf, wie unterschiedlich diese Tools sind. Metaio zum Beispiel war sehr leicht und intuitiv zu benutzen. PixLive hingegen war nur die grafische Erweiterung einfach, der Rest war gänzlich unklar und die Anleitung von Vidioti war veraltet. Nach dem Verkauf von Metaio an Apple musste auf die zweitbeste SDK ausgewichen werden. Da Apple alle Produkte von Metaio einstellte, um diese später bei eigenen Produkten zu integrieren.

Mit Vuforia wurden zu Beginn ein paar Tutorials durchgeführt, da Vuforia ein Plugin für die Software Unity3D bereitstellt. Die API und Manuals von Unity mussten einige Male konsultiert werden, um zu verstehen wie die einzelnen Funktionen in Unity umgesetzt werden oder um herauszufinden wie diese funktionieren.

Keine Integration von Cardboard in Vuforia

Die Unity Plugins von Cardboard und Vuforia arbeiten aktuell (noch) getrennt voneinander. Vuforia benötigt eine andere Kamera als Cardboard. Dadurch lassen sich diese beiden Plugins nicht gut

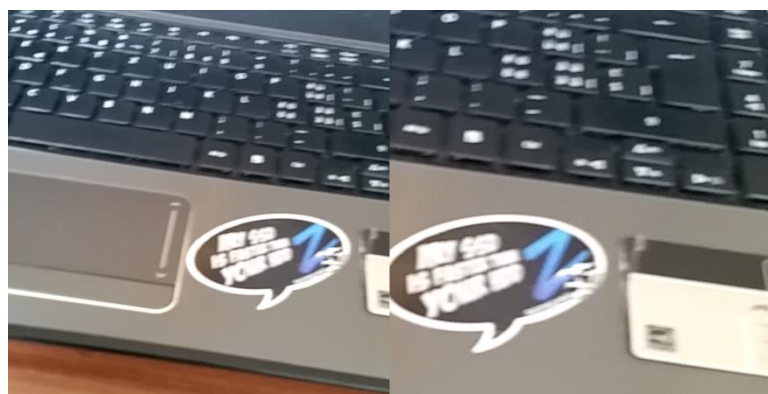


Abb. 12: Screenshot duplizierte Vuforiakamera (Eigene Darstellung)

mit einander verbinden. Aus diesem Grunde wurde versucht mit Vuforia die Ansicht von Cardboard zu simulieren. Die rechte Kamera (Siehe Abb. 12) ist eine Kopie der Linken. Jedoch ist das rechte Bild stark an gezoomt. Es wurde jedoch lediglich die breite auf einen halben Bildschirm eingestellt und der Winkel etwas nach Links und Rechts verschoben. Es wurde ca. 15 Stunden investiert um verschiedene Lösungsansätze auszuprobieren und diesen Fehler zu beheben – jedoch ohne Erfolg. Deswegen bietet der Prototyp aktuell nur die Tablet Version an.

Verschiedene Softwareversionen

Bei der Installation der verschiedenen Softwaretools wurde die (damalige) aktuellste Version von Unity und dem Unity Plugin von Vuforia installiert. Jedoch wurde für Unity die 64Bit Version gewählt. Als Vuforia zum ersten Mal getestet werden sollte, funktionierte die Erkennung nicht. Nach kurzer Recherche über das Vuforia Plugin war klar, dass dieses nur die 32Bit Version unterstützt. Unity musste deinstalliert und die 32Bit Version installiert werden. Dies sollte nicht das letzte Mal der Deinstallation von Unity bleiben.

Mitten während den ersten Versuchen ein Zielobjekt erkennen zulassen, kam ein automatisches Update von Unity. Da Unity erkannt hat, dass auf dem verwendeten Computer ein 64Bit Betriebssystem läuft, hat es natürlich die 64Bit Version installiert. Dies wurde aber erst bemerkt als das Zielobjekt nicht mehr erkannt wurde. Von neuem musste Unity deinstalliert werden. Da es in der zwischen Zeit ein weiteres Update gab, wurde die allerneuste Version installiert.

Das alte Projekt wurde übernommen, aber das Zielbild im Projekt wurde immer noch nicht erkannt. Daher wurde entschlossen ein weiteres Mal Unity zu deinstallieren. Dieses Mal wurde die erste Version, welche funktionierte, wieder installiert. Jedoch konnte das alte Projekt nicht importieren werden, weil dieses mit einer älteren Unity Version nicht mehr kompatibel war – durch das Importieren in die neuste Version wurden einige Projektdaten überschrieben. Daher musste ein neues Projekt erstellt werden.

Limitation der App Grösse

Gemäss Android Developers (2012) darf eine App nicht grösser als 50MB gross sein. Jedoch können zwei Erweiterungsdateien von bis zu zwei GB angefügt werden. Durch diese erste Einschränkung lassen sich nur sehr wenige Videos und Audiodateien integrieren.

Deswegen wird dem Endkunden zwei Möglichkeiten geboten. Erstens alle Inhalte werden per Stream geladen. Zweitens die Inhalte werden per Download auf das Smartphone in einem späteren Zeitpunkt geladen.

Die Daten werden vorab auf einen gratis Webserver von Hosting24 geladen (<http://erasmusmxvi.net76.net/>). Der Video- oder Audioplayer streamt sich die Daten direkt von diesem Webserver. Zum Beispiel erreicht der Audioplayer mit diesem Link http://erasmusmxvi.net76.net/Erasmus/Audio/Boris_Vian-Le_d_serteur.mp3 ein Lied von Boris Vian.

In der Downloadoption werden alle Ordner und Dateien von dem Server lokal auf das Smartphone geladen. Falls eine SD-Karte verfügbar ist, werden die Daten auf diese gespeichert. Beim Abspielen der Dateien wird statt der URL auf die lokale Datei verwiesen.

Was hätte man anders / besser machen können?

AR und Cardboard Test

Beim Testen der verschiedenen AR SDKs wurde nicht darauf geachtet, wie diese im Zusammengang mit Google Cardboard verwendet werden können. Da es fast von Anfang an klar war, dass Cardboard im Prototyp verwendet wird.

Die Recherche und Analysephase sollte so kurz wie möglich gehalten werden. Deswegen wurden die verschiedenen SDKs nicht auf die Kompatibilität mit Google Cardboard getestet, da dies auf der Entwicklungsebene geschehen sollte. Als es darum ging eine Lösung zu finden, wie Vuforia und Cardboard mit einander kombiniert werden können, wurde klar, dass dies ein Auswahlkriterium der SDKs gewesen sein sollte. Glücklicherweise kann mit Vuforia die Cardboardansicht simuliert werden, wenn auch (noch) nicht mit einem optimalen Ergebnis (Vergleiche mit *Keine Integration von Cardboard in Vuforia*).

Unit Tests

Die einzelnen Funktionen der App wurden nach jeder Implementierung manuell getestet. Dazu wurde die App auf ein Samsung Galaxy S5 geladen und die verschiedenen Funktionen durchgespielt. Sobald mehrere Funktionen zusammen kamen, wurden diese Marcel Henry zur Verfügung gestellt. Damit dieser die App auch testen konnte und seine Meinung dazugeben konnte. Auf diese Weise konnte der Prototyp immer wieder den Wünschen des Kunden angepasst werden und dieser wusste über den aktuellen Stand Bescheid.

Das Problem bei manuellen Tests ist, dass diese sehr zeitintensiv sind und nicht immer alle verschiedenen Möglichkeiten getestet werden. Deswegen sollte die finale App Unit Tests enthalten. Damit die Qualität immer garantiert ist und alle verschiedenen Funktionen immer automatisch getestet werden.

Wie geht es weiter mit der Erasmus App?

Die App wird voraussichtlich ab Mai 2016 im Historischen Museum Basel eingesetzt. Der Prototyp wird planmässig von Antoine Widmer und Roger Schaer weiterentwickelt und schlussendlich in den Playstore geladen. Der Prototyp soll durch eine weitere Technik erweitert werden: iBeacon oder Geoposition. Durch diese Techniken werden in der Tramtour verschiedene Events ausgelöst, damit in der Tour durch Basel die Geschichte von Erasmus von Rotterdam sich angehört bzw. angesehen werden kann. Des Weiteren müssen die konkreten Inhalte bereitgestellt werden sowie Tests Vorort durchgeführt werden.

Daneben muss noch eine iOS App erstellt werden. Die AR Funktionen können angepasst übernommen werden, da Unity auch ein iOS Plugin bereitstellt. Jedoch müssen die Funktionen, welche in Android Studio erstellt wurden, neu in Objective-C geschrieben werden. Schlussendlich muss die App noch in den App Store geladen werden.

Literaturverzeichnis

- HTC CORPORATION. (kein Datum). Abgerufen im Mai 2015 von HTC CORPORATION:
<http://www.htcvr.com/>
- Valve Corporation. (kein Datum). Abgerufen im Mai 2015 von Valve Corporation:
<http://store.steampowered.com/universe/vr>
- Android Developers. (5. März 2012). *Android Developers Blog*. Abgerufen am 7. Juli 2015 von
<http://android-developers.blogspot.ch/2012/03/android-apps-break-50mb-barrier.html>
- Bonset, S. (24. Mai 2014). *t3n*. Abgerufen am 16. Juni 2015 von <http://t3n.de/news/virtual-reality-oculus-rift-project-morpheus-546904/>
- DATACOM Buchverlag GmbH. (kein Datum). *ITWissen*. Abgerufen am 7. Mai 2015 von
<http://www.itwissen.info/definition/lexikon/augmented-reality-AR-Erweiterte-Realitaet.html>
- Davis, N., & Community von Socialcompare. (11. März 2015). *SocialCompare.com*. Abgerufen im April 2015 von <http://socialcompare.com/en/comparison/augmented-reality-sdks>
- Dunning, J. (19. März 2014). *CraveOnline Media LLC*. Abgerufen im Mai 2015 von
<http://www.playstationlifestyle.net/2014/03/19/lots-of-project-morpheus-details-no-problem-hitting-1080p60fps-with-games/>
- Fakultät für Informatik. (kein Datum). *Technische Universität München*. Abgerufen am 20. Juli 2015 von <https://www.in.tum.de/forschung/forschungs-highlights/medical-augmented-reality.html>
- Farago, P. (27. August 2012). *Flurry Insights Blog*. Abgerufen am 11. Juli 2015 von
<http://flurrymobile.tumblr.com/post/113379358945/ios-and-android-adoption-explodes-internationally>
- Fauster, R., & Yoshida, S. (4. März 2015). *Playstation.Blog*. Abgerufen im Mai 2015 von
<http://blog.de.playstation.com/2015/03/04/gdc-2015-neues-zu-project-morpheus/>
- Felix. (13. Juni 2015). *Stackoverflow*. Abgerufen am 20. Juli 2015 von
<http://stackoverflow.com/a/30821239>
- Google Inc. (kein Datum). *Android*. Abgerufen im Mai 2015 von
<https://developer.android.com/tools/studio/index.html>

- Google Inc. (kein Datum). *Google*. Abgerufen im Mai 2015 von <https://www.google.com/get/cardboard/>
- Häntschi, M. (12. Juni 2015). *Computer Bild*. Abgerufen am 16. Juli 2015 von <http://www.computerbild.de/artikel/cbs-News-Spiele-Hardware-HTC-Vive-11468052.html>
- Heil, P. (5. Februar 2015). *Gadgets-China*. Abgerufen im Mai 2015 von <http://gadgets-china.de/google-cardboard-bausatz-virtual-reality-brille/>
- IDG Entertainment Media GmbH. (kein Datum). *Gamestar*. Abgerufen im Mai 2015 von <http://www.gamestar.de/hardware/tft-monitore/sony-project-morpheus/793.html>
- K.L. (9. März 2015). *stack exchange inc*. Abgerufen im Mai 2015 von <http://stackoverflow.com/a/28939883>
- Kneussel, J. (23. März 2015). *GIGA Digital AG*. Abgerufen im Mai 2015 von <http://www.giga.de/konsolen/oculus-rift/specials/oculus-rift-preis/>
- Layar. (kein Datum). *Layar*. Abgerufen im April / Mai 2015 von <https://www.layar.com/>
- Metaio GmbH. (2015). *Metaio*. Abgerufen im April / Mai 2015 von <http://www.metaio.com/>
- Missfeldt, M. (kein Datum). *Brillen & Sehhilfen*. Abgerufen am 9. Juni 2015 von <http://www.brillen-sehhilfen.de/google-brille/google-vr-brille-pappe-cardboard.php>
- Mobilzone Holding AG. (kein Datum). *Mobilzone*. Abgerufen im Mai 2015 von <http://www.mobilezone.ch/mobiltelefone/13423-samsung-sm-n910f-galaxy-note-4-32gb-black>
- Ninjamock. (kein Datum). *Ninjamock*. Abgerufen im Juni / Juli 2015 von <http://ninjamock.com/>
- Oculus VR & Oculus VR, LLC. (2015). *Oculus*. Abgerufen im Mai 2015 von <https://www.oculus.com/dk2/>
- Oculus VR & Oculus VR, LLC. (kein Datum). *Oculus*. Abgerufen im Mai 2015 von <https://www.oculus.com/gear-vr/>
- Qualcomm Connected Experiences Inc. (kein Datum). *Vuforia Developer Portal*. Abgerufen im April / Mai 2015 von <https://developer.vuforia.com/>
- Qualcomm Technologies Inc. (2015). *Qualcomm*. Abgerufen im April / Mai 2015 von <https://www.qualcomm.com/products/vuforia>

- Reeve, J. (18. Dezember 2014). Abgerufen am 21. Mai 2015 von <http://blog.wsd.net/jreeve/google-cardboard-and-photospheres/>
- Ridden, P. (14. August 2013). *Gizmag*. Abgerufen am 19. Juli 2015 von <http://www.gizmag.com/ikea-augmented-reality-catalog-app/28703/>
- Rouse, M. (Juni 2009). *WhatIs*. Abgerufen am 17. Mai 2015 von <http://whatis.techtarget.com/definition/virtual-reality>
- Sabella, K., & Lavery, R. (13. Januar 2015). *Newsdesk*. Abgerufen am 17. Juli 2015 von <http://newsdesk.si.edu/releases/smithsonian-brings-historic-specimens-life-free-skin-and-bones-mobile-app>
- Samsung Electronics Switzerland GmbH. (kein Datum). *Samsung*. Abgerufen im Mai 2015 von <http://www.samsung.com/ch/promotions/galaxynote4/spec/gearvr/>
- Schleeh, H. (16. April 2014). *Schleeh.de*. Abgerufen am 17. Juli 2015 von <http://schleeh.de/mit-google-glass-und-ipad-durch-das-museum-augmented-reality/>
- Schuppisser, R. (13. März 2015). *Aargauer Zeitung*. Abgerufen am 16. Juni 2015 von <http://www.aargauerzeitung.ch/leben/virtual-reality-brille-ermoeglicht-blick-ins-noch-nicht-gebaute-haus-128932205>
- Shepherd, A., & McCallion, J. (4. März 2015). *Dennis Publishing Limited*. Abgerufen im Mai 2015 von <http://www.pcpro.co.uk/peripherals/1000343/project-morpheus-confirmed-release-date-price-and-features-everything-you-need>
- Steinlechner, P. (11. September 2012). *Golem.de*. Abgerufen am 12. Juli 2015 von <http://www.golem.de/news/interview-unity-3d-der-typische-unity-user-ist-schwer-zu-beschreiben-1209-94476.html>
- Whittinghill, D. (24. März 2015). *Purdue University*. Abgerufen am 16. Juli 2015 von <http://www.purdue.edu/newsroom/releases/2015/Q1/virtual-nose-may-reduce-simulator-sickness-in-video-games.html>
- Wikitude GmbH. (2015). *Wikitude*. Abgerufen im April / Mai 2015 von <https://www.wikitude.com/>
- Winchester, H. (6. März 2015). *Wearable*. Abgerufen im Mai 2015 von <http://www.wearable.com/headgear/the-best-ar-and-vr-headsets>

Anhang I: Product Backlog

Zu Beginn der Arbeit musste ein Product Backlog erstellt werden. Damit alle beteiligten den Product Backlog lesen bzw. verstehen konnten wurde dieser in Englisch verfasst. Dieser diente bei den wöchentlichen Sitzungen als Grundlage, was in der nächsten Woche / Sprint gemacht werden soll. In jeder Sitzung wurde am Anfang präsentiert was gemacht wurde. Danach gab es eine kurze Diskussion, was gut oder weniger gut ist und wo es Probleme gab. Zum Schluss wurde die Planung der nächsten Woche besprochen.

Erklärung zu den Farben der Tabelle 3:

Priorität: Rot = Muss, Orange = Nice-to-have

Status: Grün = erledigt, Orange = in Arbeit, Rot = zu erledigen

Tabelle 3: Product Backlog (Eigene Darstellung)

Nr.	Theme	User Stories As a/an	I want ...	so that ...	More Information	Priorität	Status
1	Release	User	to have a manual	I can get help to install the app		1	3
2	Analysis	Developer	to use the best tool for Augmented Reality (AR) according to my needs	so I can integrated all desired functions	Display text, image, video, audio	1	3
3	Analysis	Developer	to use the best tool for Virtual Reality (VR) according to my needs	so I can integrated all desired functions	Display text, image, video, audio -> 360° Foto	1	3
4	Analysis	Client	information about 360° videos in Switzerland	I can decide, if it is possible to you for my project	vision 360° -> Kontakt, Preis, Miete für 1, 2 Tage	2	1
5	Analysis	User	only to install one single app	I can use Cardboard or a table / Smartphone		1	3
6	Development	User	to fix an image/logo with my Smartphone	I can get more information about the presented object		1	3
7	Development	User	to fix an image/logo with my Smartphone	I can see a video		1	3
8	Development	User	to fix an object with Google CB	I can get more information		1	3
9	Development	User	to be in the near of an object with my Smartphone	I can get more information about the presented object	Beacon	2	1
10	Development	User	to be in the near of an object with my Smartphone	I can see a video	Beacon	2	1
11	Development	User	to fix at point for some seconds with my Google Cardboard	I can watch videos, hear audios or see some pictures or texts (displayed like popups)	360° Foto im Hintergrund, VR erleben --> VR Tunnel	2	1
12	Development	User	to look at a 3D-Object with my Smartphone	I can get more information about the object (text, picture)		2	1
13	Development	User	to look at a 3D-Object with my Smartphone	I can see a video, hear an audio		2	1
14	Development	User	to look at a 3D-Object with my Google CB	I can get more information about the object (text, picture)		2	1
15	Development	User	to look at a 3D-Object with my Google CB	I can see a video, hear an audio		2	1

Anhang II: Stundenrapport

Alle geleisteten Stunden an dieser Arbeit wurden in die Tabelle 4 übertragen. Vom 27. April bis 29. Juni standen drei und ein halber Tag pro Woche für die Arbeit zur Verfügung. Danach wurde jeder Arbeitstag für die Arbeit aufgewendet. Die halben Tage sind in violett markiert. Wochenenden und Feiertage bzw. Ferientage sind in hell rot dargestellt.

Tabelle 4: Stundenrapport mit Übersicht über Probleme und deren Lösungen (Eigene Darstellung)

	Tag	Datum	Stunden	Was	Probleme / Bemerkung	Lösungen
Woche 0	Freitag	24.04.2015	2	Sitzung mit Kunden (Marcel Henry), Grafiker, Antoine Widmer und Roger Schaer		
	Total		2			
Woche 1	Analysephase					
	Montag	27.04.2015	8	- Dokumente erstellen und formatieren nach Vorgaben - Lastenheft beginnen - AR und VR Tools suchen	Sehr grosse Auswahl an Tools	Nach Top SDKs suchen
	Dienstag	28.04.2015	0		Aufsicht 2. Semester Praktisches Arbeiten (TP)	
	Mittwoch	29.04.2015	8	- Vergleichskriterien aufstellen - Über AR informieren - Tools vergleichen		
	Donnerstag	30.04.2015	3	- Metaio installieren - Metaio testen (Bild und 3D-Objekt)	Kamera kalibrieren	Viel Geduld nötig
	Freitag	01.05.2015	3	- Sitzung mit Antoine Widmer, Henning Müller und Roger Schaer - Wikitude installieren - Wikitude testen - Product Backlog anpassen	Wikitude hat keine grafische Oberfläche	Wikitude Studio benutzen
	Samstag	02.05.2015				
	Sonntag	03.05.2015				
Total		22				
Woche 2	Tag	Datum	Stunden	Was	Probleme / Bemerkung	Lösungen
	Montag	04.05.2015	8	- Product Backlog fertig stellen - Wikitude testen - Vuforia installieren - Unity installieren - Vuforia testen	- nullpointer exeption beim starten von wikitude app -> manuelles hinzufügen eines Bildes zu den sampeln - Falsche unity version installiert	project in samples_ir.Ist hinzufügen
	Dienstag	05.05.2015	2	- Maus scannen mit Vuforia Scanner		
	Mittwoch	06.05.2015	8	- Vuforia / Unity testen - Layar installieren - Layar testen - ARmedia installieren	Mona Lisa und Maus funktionieren nicht als Testobjekte (zu wenig Anhaltspunkte)	Neue Objekte suchen
	Donnerstag	07.05.2015	3	- ARmedia testen - Xloudia installieren - Internet recherche AR/VR	ARmedia Trial erlaubt keine 3D-Objekte zu verwalten	
	Freitag	08.05.2015	2	- Sitzung - Internet recherche AR/VR - Arbeit schreiben		
	Samstag	09.05.2015				
	Sonntag	10.05.2015				
Total		23				
Woche 3	Tag	Datum	Stunden	Was	Probleme / Bemerkung	Lösungen
	Montag	11.05.2015	4	- SDK entscheiden - An Arbeit schreiben		
	Dienstag	12.05.2015			Aufsicht 2. Semester TP	
	Mittwoch	13.05.2015	4	- Entscheid SDK - Arbeit schreiben	Vorstellungsgespräch	
	Donnerstag	14.05.2015				
	Freitag	15.05.2015				
	Samstag	16.05.2015				
	Sonntag	17.05.2015	3	- Arbeit schreiben - Internet recherche		
Total		11				

	Tag	Datum	Stunden	Was	Probleme / Bemerkung	Lösungen
Woche 4	Montag	18.05.2015	8	- Sitzung - Internet recherche - Arbeit schreiben - 360° Foto testen - Vidinoti testen	Vidinoti: - Probleme beim Testen (App erstellen) - Demo abspielen - Demo App hat Probleme beim Starten / Bild erkennen	Nach Tutorials / Guides suchen
	Dienstag	19.05.2015	3	- Arbeit schreiben - VR Brillen vergleichskriterien aufstellen - VR Brillen vergleichen		
	Mittwoch	20.05.2015	8	- VR Brillen vergleichen - Arbeit schreiben - VR Brillen Entscheid		
	Donnerstag	21.05.2015	3	- 360° Foto testen - 360° Video recherche - Arbeit schreiben		
	Freitag	22.05.2015			Hackathon	
	Samstag	23.05.2015				
	Sonntag	24.05.2015				
Total			22			

	Tag	Datum	Stunden	Was	Probleme / Bemerkung	Lösungen
Woche 5	Montag	25.05.2015	2	- 360° Videos mit Google Cardboard recherche		
	Dienstag	26.05.2015	3	- PixLive Lösung suchen - 360° Videos recherche - Arbeit schreiben		
	Mittwoch	27.05.2015	9	- PixLive Lösung suchen - Arbeit schreiben		
	Donnerstag	28.05.2015				
	Freitag	29.05.2015	1	PixLive Lösung suchen		
	Samstag	30.05.2015				
	Sonntag	31.05.2015				
Total			15			

	Tag	Datum	Stunden	Was	Probleme / Bemerkung	Lösungen
				Programmierungsphase		
Woche 6	Montag	01.06.2015	8	- Sitzung - Arbeit schreiben - Download unityextension Google Cardboard - Android-Test-Gerät einstellen	Metaio wurde an Apple verkauft	
	Dienstag	02.06.2015	3	- Neuer Entscheid - Vuforia oder Wikitude - Vuforia Bild erkennen -> Text nicht in 90° zu Bild - Arbeit schreiben	Aufsicht 2. Semester TP	
	Mittwoch	03.06.2015	8	- Vuforia Android SDK testen - Arbeit schreiben	Probleme beim Ausführen der App: Kann keine Sample aufmachen	
	Donnerstag	04.06.2015	3	- Unity: Basis lernen - Cardboardsicht simulieren	Bilder sind nicht gleich gross / zeigen nicht das Gleiche	
	Freitag	05.06.2015	3	- Unity: Basis lernen - Problem mit Cardboardsicht simulieren lösen		
	Samstag	06.06.2015				
	Sonntag	07.06.2015				
Total			25			

	Tag	Datum	Stunden	Was	Probleme / Bemerkung	Lösungen
Woche 7	Montag	08.06.2015	8	- Unity: Basis lernen - Sprite und Audio hinzufügen - Arbeit schreiben	Unity 64bit funktioniert nicht mit Vuforia	
	Dienstag	09.06.2015	4	- Arbeit schreiben - Audio / Video hinzufügen - Linkes und rechtes Bild anpassen	Rechtes Bild ist zu nahe angezoomt	
	Mittwoch	10.06.2015	8	- Bilder anpassen - Audio hinzufügen	Rechtes Bild ist zu nahe angezoomt	
	Donnerstag	11.06.2015	3	- AR wieder funktionieren lassen	App Licence Key konnte nicht mehr hinzugefügt werden	
	Freitag	12.06.2015	4	- Sitzung - Telefon Marcel Henry --> Aktueller Stand schicken - Text einfügen - Guide erstellen -> APK auf Smartphone laden	Text verschwindet nicht immer wie er sollte	
	Samstag	13.06.2015				
	Sonntag	14.06.2015				
Total			27			

	Tag	Datum	Stunden	Was	Probleme / Bemerkung	Lösungen
Woche 8	Prüfungsvorbereitungswoche					
	Montag	15.06.2015				
	Dienstag	16.06.2015	2	Mockup erstellen		
	Mittwoch	17.06.2015				
	Donnerstag	18.06.2015				
	Freitag	19.06.2015				
	Samstag	20.06.2015				
	Sonntag	21.06.2015				
Total			2			

	Tag	Datum	Stunden	Was	Probleme / Bemerkung	Lösungen
Woche 9	Prüfungswoche					
	Montag	22.06.2015				
	Dienstag	23.06.2015	4	- Sitzung - Photoshop erstellen -> nach Wunsch		
	Mittwoch	24.06.2015	11	- Sitzung in Basel: Ausstellung besichtigen - Planung/Konzept - Mockups zeichnen		
	Donnerstag	25.06.2015				
	Freitag	26.06.2015	4	- Android Activities nach Vorlage erstellen		
	Samstag	27.06.2015				
	Sonntag	28.06.2015				
Total			19			

	Tag	Datum	Stunden	Was	Probleme / Bemerkung	Lösungen
Woche 10	Prototyp entwickeln & Arbeit schreiben					
	Montag	29.06.2015	9	- Android Activities nach Vorlage erstellen		
	Dienstag	30.06.2015	9	- Unity app erstellen (Video streaming, Bild, 3D Model, Musik)	Gratis Server war nicht zuverlässig. Nach 3h wurde der Account blockiert Audio streaming funktioniert anders als Video	Neuen gratis Server suchen
	Mittwoch	01.07.2015	9	- Audio streaming - Integration Unity & Android App - Photosphere einbinden	Photosphere funktioniert nicht	
	Donnerstag	02.07.2015	9	- Arbeit schreiben - Buttonstyle ändern - Fehlende Activites ergänzen		
	Freitag	03.07.2015	9	- Sitzung - Settings speichern / laden - App: Bugfixing		
	Samstag	04.07.2015	3	- Prototyp an Kunden - App: Bugfixing, Sprache		
	Sonntag	05.07.2015				
Total			48			

	Tag	Datum	Stunden	Was	Probleme / Bemerkung	Lösungen
Woche 11	Montag	06.07.2015	9	- App: Bugfixing, Download von URL	Probleme beim Downloaden: - Falscher Pfad - Download bricht ab nach 17KB	
	Dienstag	07.07.2015	9	- App: Downloadbug beheben, Arbeit schreiben		Name darf kein Whitespace haben
	Mittwoch	08.07.2015	9	- Arbeit an Dozenten schicken - Unity: Stream von Folder - Arbeit schreiben	Musik wird nicht gestreamt	Keine Rechte verteilt
	Donnerstag	09.07.2015	9	- Unity: Photosphere - Unity: Recherche - Bilder "streamen" - Arbeit schreiben		
	Freitag	10.07.2015	9	- Sitzung - Unity: Photosphere		
	Samstag	11.07.2015	2	- Arbeit schreiben / Recherchieren		
	Sonntag	12.07.2015				
Total			47			

	Tag	Datum	Stunden	Was	Probleme / Bemerkung	Lösungen
Woche 12	Montag	13.07.2015	9	- Arbeit schreiben		
	Dienstag	14.07.2015	9	- Photosphere einbauen - Sprachsettings: Bugfixing - Photospherenprototyp an Kunden		
	Mittwoch	15.07.2015	9	- Arbeit schreiben		
	Donnerstag	16.07.2015	9	- Arbeit schreiben - Recherche existierende Projekte VR		
	Freitag	17.07.2015	9	- Sitzung - Arbeit schreiben		
	Samstag	18.07.2015				
	Sonntag	19.07.2015	3	- Arbeit schreiben - Recherche zu Forschungsprojekten		
Total			48			

	Tag	Datum	Stunden	Was	Probleme / Bemerkung	Lösungen
Woche 13	Arbeit fertigstellen					
	Montag	20.07.2015	9	- Arbeit schreiben		
	Dienstag	21.07.2015	9	- Arbeit schreiben - Unity Projekt neue strukturieren		
	Mittwoch	22.07.2015	9	- Unity und Android Studio Projekt zusammenfügen - Bugfixen		
	Donnerstag	23.07.2015	9	- Arbeit korrigieren, anpassen - Code cleanen - Kommentare im Code vervollständigen		
	Freitag	24.07.2015	9	- Sitzung - Arbeit drucken - Arbeit binden lassen - Bugfixing		
	Samstag	25.07.2015	3	- Online Dokument ausfüllen - Arbeit an Urkund senden - Arbeit auf Laufwerk O Laden - CD brennen		
	Sonntag	26.07.2015				
Total			48			
Abgabe						
	Montag	27.07.2015		Abgabe der Bachelorarbeit		

Anhang III: Mockups

Die Folgenden Mockups wurden von der Verfasserin und Marcel Henry in Zusammenarbeit erstellt. Sie bilden die grafische Grundlage des Prototyps. Auf den Mockups abgebildete Bilder wie zum Beispiel das Logo stammen von Marcel Henry bzw. dem Grafikerbüro Notter + Vigne Sàrl.



Abb. 13: Mockup Startseite (Eigene Darstellung)

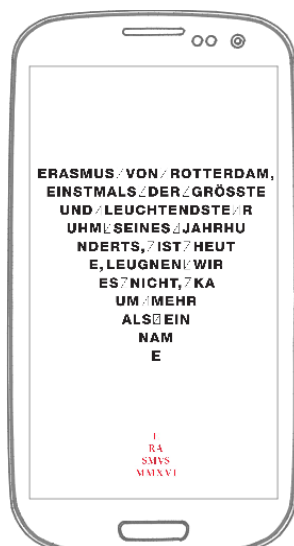


Abb. 14: Mockup Beispiel Bild (Eigene Darstellung)

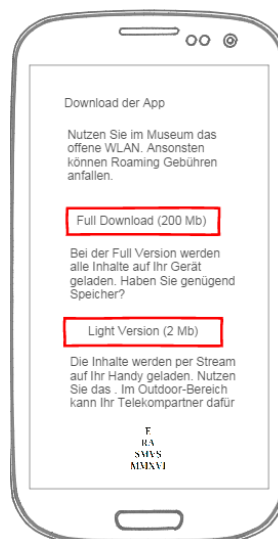


Abb. 15: Mockup Download (Eigene Darstellung)

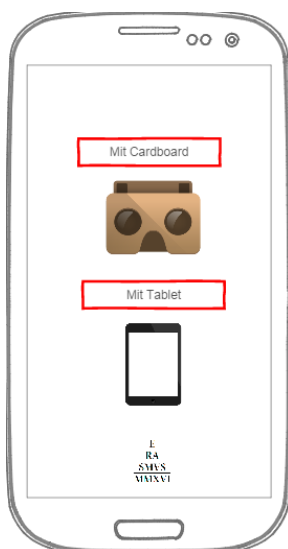


Abb. 17: Mockup BildschirmEinstellung (Eigene Darstellung)

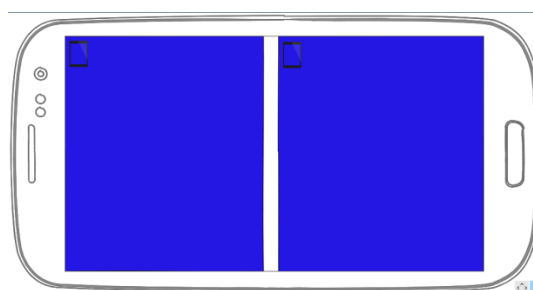


Abb. 16: Mockup Cardboardansicht (Eigene Darstellung)



Abb. 18: Mockup Cardboardansicht mit Beispiel Inhalt (Eigene Darstellung)

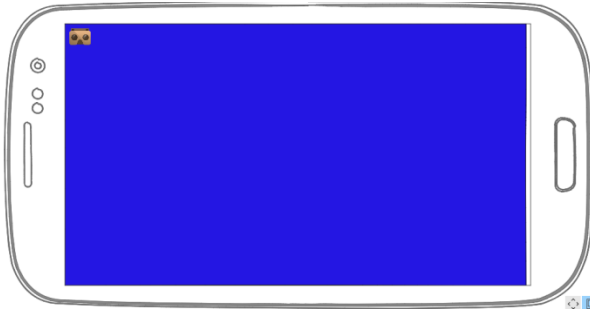


Abb. 20: Mockup Tabletansicht (Eigene Darstellung)



Abb. 19: Mockup Tabletansicht mit Beispiel Inhalt (Eigene Darstellung)

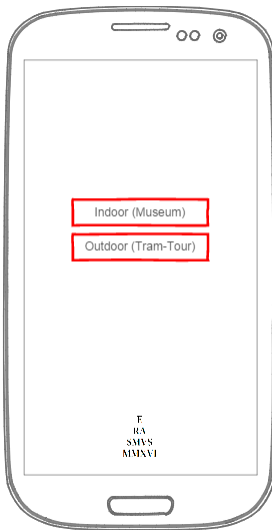


Abb. 21: Mockup Standortauswahl (Eigene Darstellung)

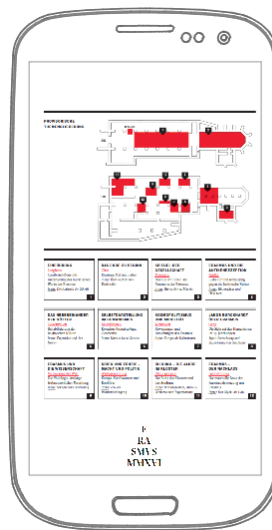


Abb. 22: Mockup Museumsplan (Eigene Darstellung)

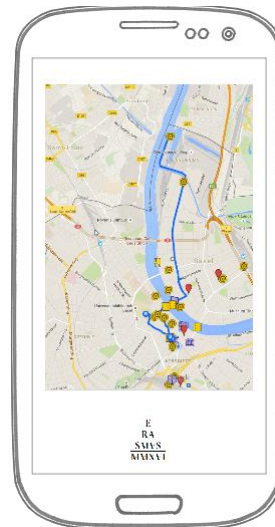


Abb. 23: Mockup Tramtour (Eigene Darstellung)



Abb. 24: Mockup Beispieltext (Eigene Darstellung)

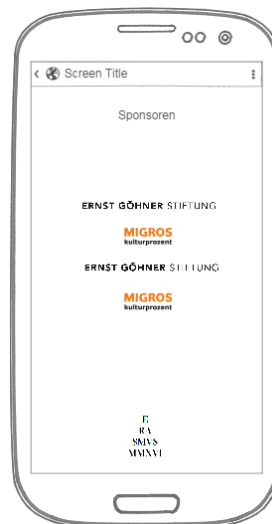


Abb. 25: Mockup Sponsoren (Eigene Darstellung)

Anhang IV: Code

Das Unity und Android Projekt befinden sich in der angehängten CD am Ende dieser Arbeit. Durch ein Kopieren des Codes würde die Übersicht, welches von den beiden Entwicklungsplattformen gegeben ist, verloren gehen.

Anhang V: QRC

Die Folgenden QRCs sind durch das Grafikerbüro Notter + Vigne Sàrl für die Erasmusausstellung angefertigt worden. Diese QRC dienen als Zielbilder für die AR-Funktionen.

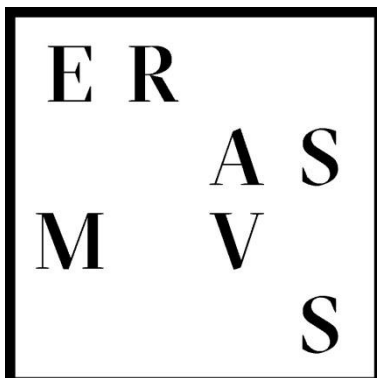


Abb. 26: QRC 1 (Quelle: Notter + Vigne Sàrl)

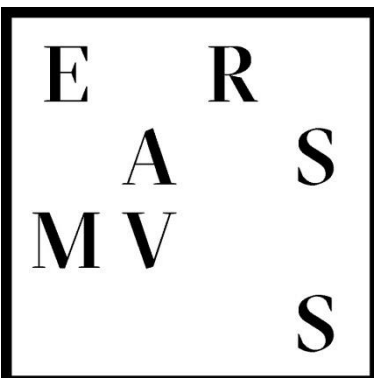


Abb. 27: QRC 2 (Quelle: Notter + Vigne Sàrl)

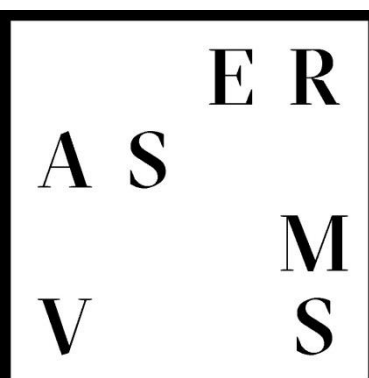


Abb. 28: QRC 3 (Quelle: Notter + Vigne Sàrl)

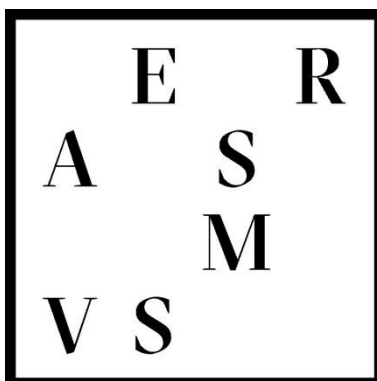


Abb. 29: QRC 4 (Quelle: Notter + Vigne Sàrl)

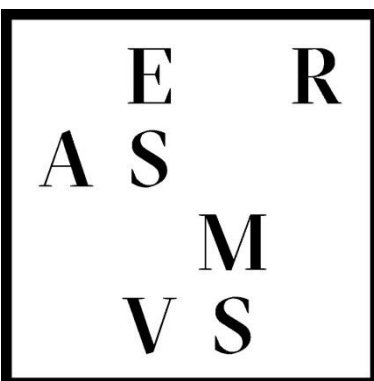


Abb. 30: QRC 5 (Quelle: Notter + Vigne Sàrl)

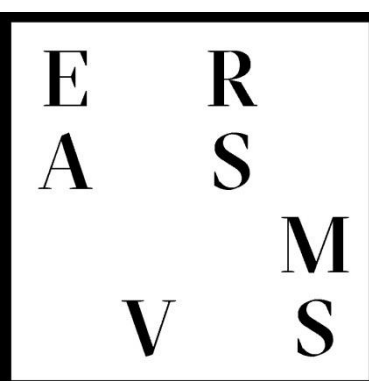


Abb. 31: QRC 6 (Quelle: Notter + Vigne Sàrl)

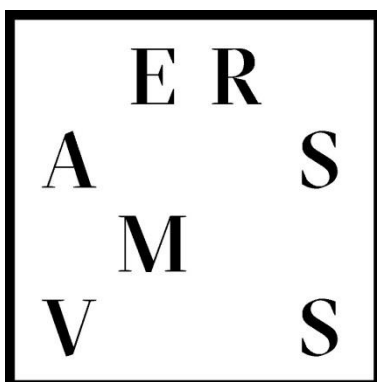


Abb. 32: QRC 7 (Quelle: Notter + Vigne Sàrl)

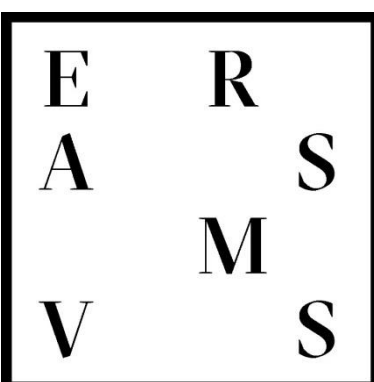


Abb. 33: QRC 8 (Quelle: Notter + Vigne Sàrl)

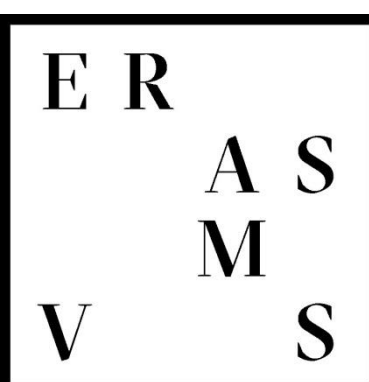


Abb. 34: QRC 9 (Quelle: Notter + Vigne Sàrl)



Abb. 35: QRC 10 (Quelle: Notter + Vigne Sàrl) Abb. 36: QRC 11 (Quelle: Notter + Vigne Sàrl) Abb. 37: QRC 12 (Quelle: Notter + Vigne Sàrl)

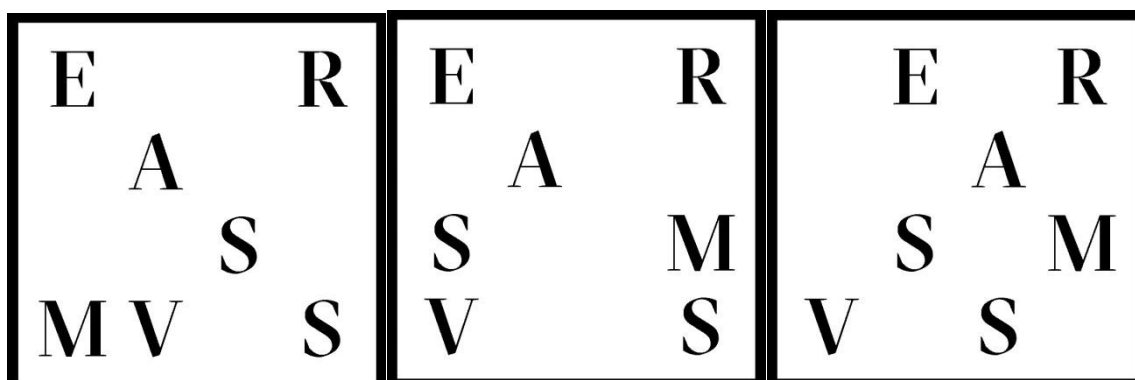


Abb. 38: QRC 13 (Quelle: Notter + Vigne Sàrl) Abb. 39: QRC 14 (Quelle: Notter + Vigne Sàrl) Abb. 40: QRC 15 (Quelle: Notter + Vigne Sàrl)

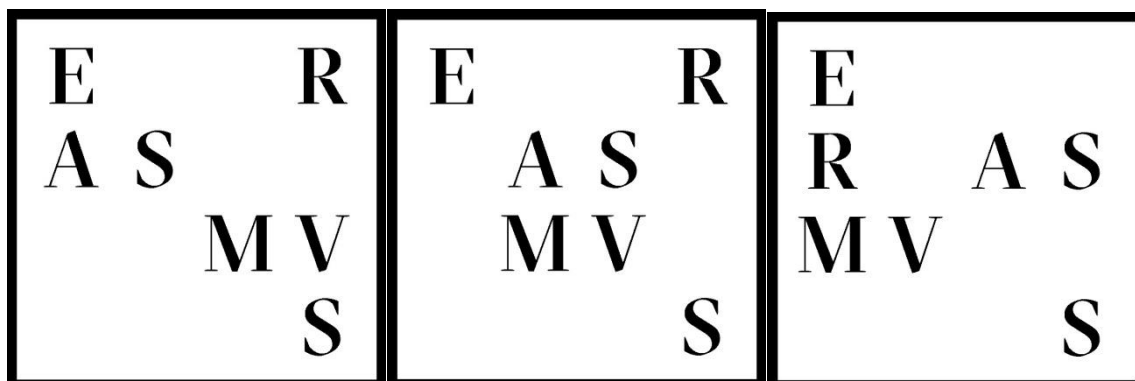


Abb. 41: QRC 16 (Quelle: Notter + Vigne Sàrl) Abb. 42: QRC 17 (Quelle: Notter + Vigne Sàrl) Abb. 43: QRC 18 (Quelle: Notter + Vigne Sàrl)

Selbstständigkeitserklärung der Verfasserin

Ich bestätige hiermit, dass ich die vorliegende Bachelorarbeit alleine und nur mit den angegebenen Hilfsmitteln realisiert habe und ausschliesslich die erwähnten Quellen benutzt habe. Ohne Einverständnis des Studiengangsleiters und des für die Bachelorarbeit verantwortlichen Dozierenden, werde ich diesen Bericht an niemanden verteilen, ausser an die Personen, die mir die wichtigsten Informationen für die Verfassung dieses Berichts geliefert haben und die ich nachstehend aufzähle: Henning Müller, Antoine Widmer, Roger Schaer.

Evelyn Hischer

Juli 2015