|  |
| --- |
| http://blog.gfx47.com/wp-content/uploads/2011/02/unity3d1.jpg |
| Arbeitsdokument „unityserver 3d im cave“  Hier steht ein Untertitel  **Art der Arbeit (Semesterarbeit / Bachelorthesis / etc.)** |
| Studiengang: [Studiengang einfügen]  Autor: [Autor/en einfügen]  Betreuer: [Betreuer einfügen]  Auftraggeber: [Auftraggeber einfügen]  Experten: [Experte/n einfügen]  Datum: [Datum einfügen] |

Management Summary

Inhaltsverzeichnis

[1 Einführung 4](#_Toc434763771)

[1.1 Vorarbeiten Projekt 2 4](#_Toc434763772)

[2 Infrastruktur 4](#_Toc434763773)

[2.1 Trackingsystem WorldViz 5](#_Toc434763774)

[2.2 Devices 5](#_Toc434763775)

[2.3 Unity Server 7](#_Toc434763776)

[2.4 Audio 7](#_Toc434763777)

[2.5 Video Matrix Switch 7](#_Toc434763778)

[3 Stereoskopie 8](#_Toc434763779)

[4 Immersion 8](#_Toc434763780)

[5 Devices 8](#_Toc434763781)

[6 VRPN 8](#_Toc434763782)

[7 Unity Package 8](#_Toc434763783)

[8 „Architektur der Komponenten“ 8](#_Toc434763784)

[9 Warping (optional, ev. bei Immersion) 8](#_Toc434763785)

[10 Demo Apps 9](#_Toc434763786)

[10.1 Shooting Gallery 9](#_Toc434763787)

[10.2 Model-Viewer 9](#_Toc434763788)

[10.3 App Drittpartei 9](#_Toc434763789)

[11 Abbildungsverzeichnis 9](#_Toc434763790)

[12 Tabellenverzeichnis 9](#_Toc434763791)

[13 Glossar 10](#_Toc434763792)

[14 Literaturverzeichnis 10](#_Toc434763793)

[15 Anhang 11](#_Toc434763794)

[16 Selbständigkeitserklärung 12](#_Toc434763795)

# Einführung

## Vorarbeiten Projekt 2

# Infrastruktur

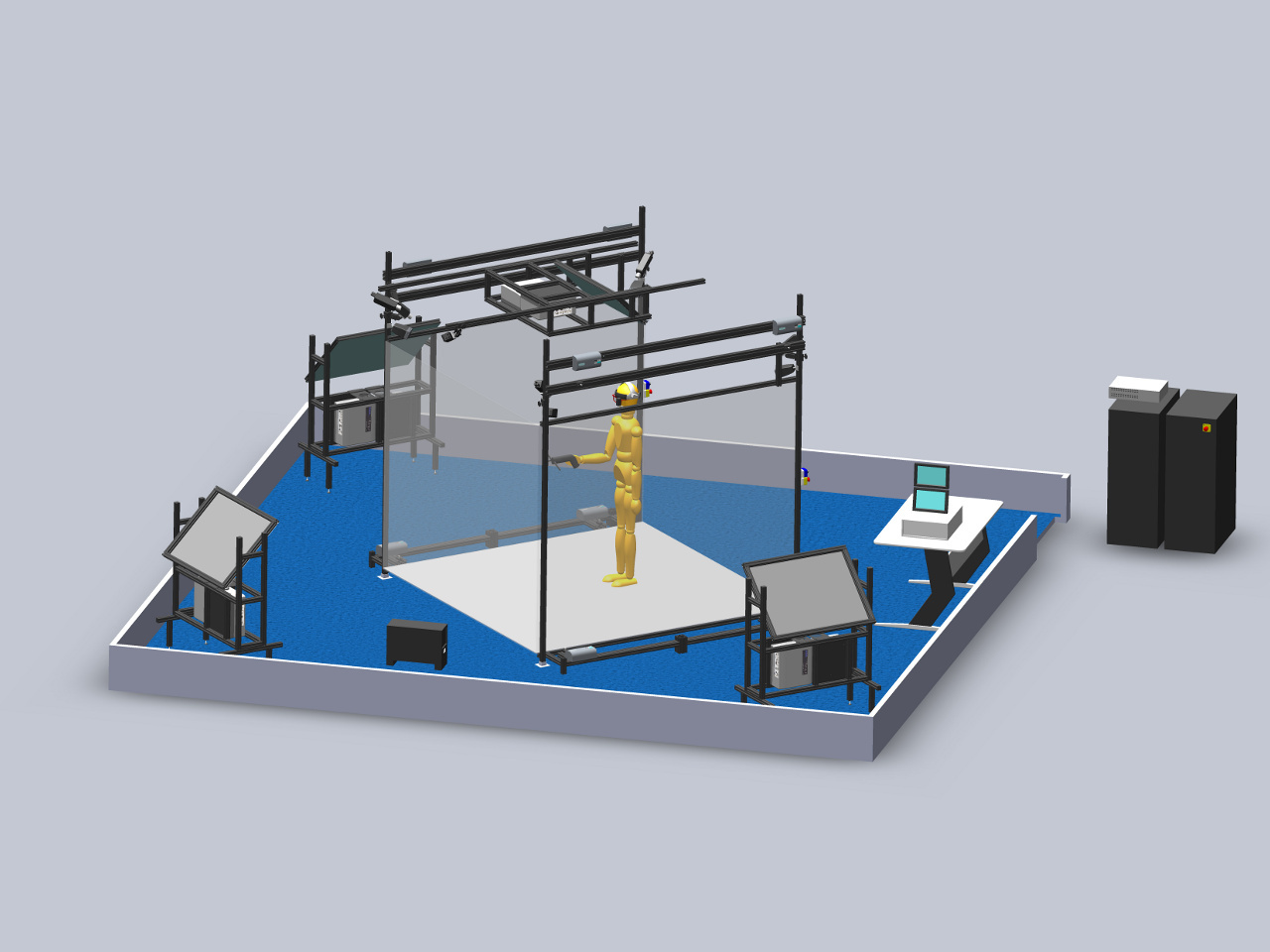


Abbildung : CAVE BFH

Der virtual reality haptic CAVE der BFH bietet Entwicklern und Forschern ein mächtiges Instrument, um hochrealistische immersive Applikationen zu bauen. Der CAVE ist eine kubische Konfiguration mit vier Leinwänden (Links, Front, Rechts, Boden) die von je 2 Projektoren bestrahlt werden. Um eine realistische 3D Projektion zu erschaffen, sind bei den Projektoren Polfilter angebracht und die Benutzer tragen eine polarisierte Brille.

Alle Komponenten vereint und dessen Abhängigkeiten werden in der folgenden Grafik dargestellt:

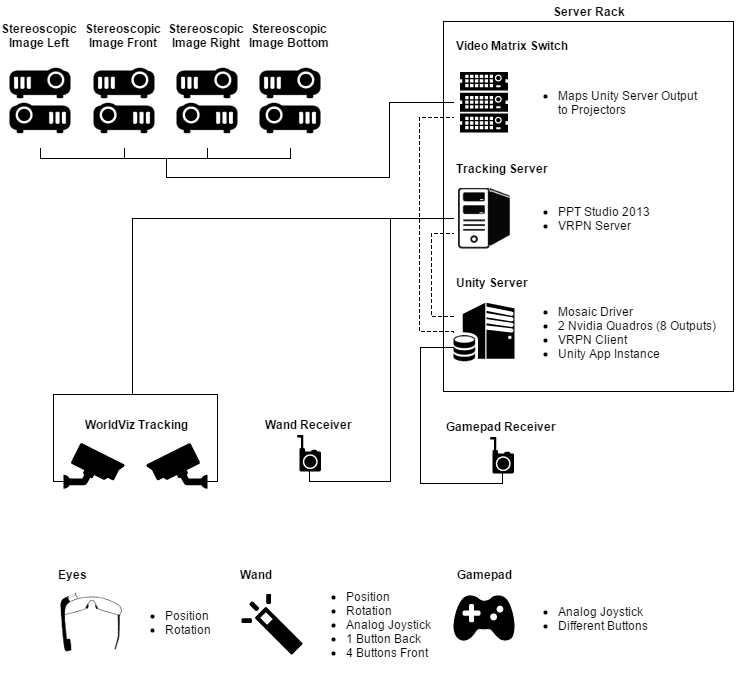


Abbildung : Infrastruktur CAVE

## Trackingsystem WorldViz

Das Trackingsystem von WorldViz wurde integriert, um Positionen und Rotationen von verschiedenen Devices im CAVE zu erfassen. Die von den 10 WorldViz Kameras übermittelten Daten werden vom PPT Studio 2013 zentral auf einem Server interpretiert, d.h. es werden Punkte im Raum und dessen Rotation der Devices berechnet, und können bei Bedarf über das VRPN-Protokoll abgefragt werden.

## Devices

* **Eyes**

Die Eyes von WorldViz sind polarisierte Brillen mit zwei montierten Infrarot-Trackern. Somit lassen sich die Position des Kopfes und dessen Rotation auf zwei Achsen (Yaw und Roll) bestimmen.

****

Abbildung : PPT Eyes, Quelle: www.worldviz.com

* **Wand**

Der Wand von WorldViz ist das primäre Inputgerät. Nebst zwei Infrarot-Tackern, welche für die Positions- und Rotationsbestimmung verwendet werden, ist ein Gyrometer integriert, um noch präziser Drehungen feststellen zu können. Dadurch wird auch die fehlende Rotationsachse kompensiert und es können Yaw, Roll und Pitch ermittelt werden.

Als Inputs dienen ein analoger Joystick, vier Buttons auf der Vorderseite sowie ein Button auf der Rückseite.

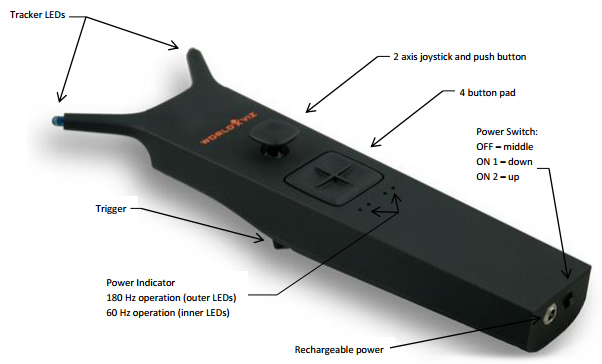


Abbildung : PPT Wand, Quelle: www.worldviz.com

* **Gamepad**

Ein weiteres Inputgerät ist ein handelsübliches Gamepad. Frei von jeglichen Tracking wird einzig die Unity-Applikation gesteuert.



Abbildung : Gamepad, Quelle: www.androidrundown.com

## Unity Server

Der leistungsstarke Untiy Server ist der Knotenpunkt des gesamten Systems. Auf diesem Rechner läuft die Unity-Applikation mit dem konfigurierten UnityPlugin, welches die Trackingdaten vom Trackingserver abgreift und das korrekte Rendering in der Uniy-Applikation für die Projektorenaufteilung sicherstellt. Mittels Mosaic, einem speziellen Treiber von Nvidia der die Aufteilung auf mehrere Grafikkartenausgänge übernimmt, werden alle Projektoren korrekt für die stereoskopische Projektion angesprochen.

## Audio

Um die Immersion weiter zu steigern, ist ein 3D Soundystem mit vier Lautsprechern in Betrieb.

## Video Matrix Switch

Weil parallel weitere Clients in Betrieb sind und Bilddaten für den CAVE liefern können, wird ein Video Matrix Switch eingesetzt, um die verschiedenen Inputquellen auf die 8 bestehenden Projektoren abbilden zu können.

# Stereoskopie

* Kamerasettings
  + Sekundäre Kamera
  + GUI Elemente
* Mosaic Settings

# Immersion

* Frustum angleichen
* Kalibrieren

# Devices

* Wand
  + Virutueller cave
  + Mousecursor
  + Einstellung in Unity (UnityPlugin -> Inspector):  
      
    Channel = markerID (PPTStudio) – 1  
    WorldVizObject = PPT0
* Eyes
* Gamepad

# VRPN

* Wrapper

# Unity Package

* Asset

# „Architektur der Komponenten“

* Unterteilung Module
  + Singleton (Hilfsklasse)
* Logisch und technisch

# Warping (optional, ev. bei Immersion)

# Demo Apps

## Shooting Gallery



Abbildung : Shooting Gallery Ingame

Das Setting dieses Demospiels ist eine Schiessbude im Wilder Westen Stil, wie sie auf einem Jahrmarkt anzutreffen ist. Die Galerien mit den abzuschiessenden Zielen verteilen sich rund um den Spieler. Mit Hilfe des Head Trackings kann sich der Spieler in der gesamten Szenerie umschauen, Bewegungen ausführen und die Objekte aus verschiedenen Perspektiven betrachten. Das Wand-Device steuert das Gewehr, um die Zielobjekte anzuvisieren und abzuschiessen. Die Buttons des Wands werden gebraucht um das Gewehr abzufeuern.

* **Umgebung**

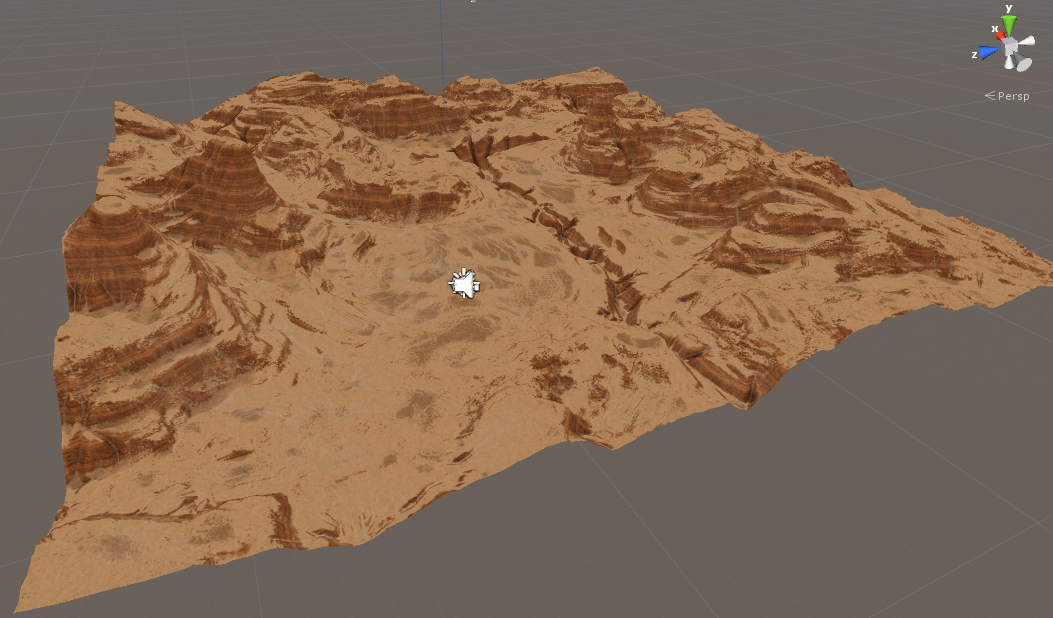
Die Umgebung ist ein gekauftes Model aus dem Asset Store von Unity mit riesigen Ausmessungen.  
  


Abbildung : Desert Model, Quelle: Unity Asset Store

* **Gewehr**

Das Model des Gewehrs ist ebenfalls aus dem Unity Asset Store. Die Enten und Zielscheiben werden mit der Cursorposition anvisiert, welche durch das UnityPlugin vom Wand gesetzt wird.

**Rotation**

Das Gewehr dreht sich entsprechend dessen Position. Die Rotation wird zweierlei beeinflusst.

1. **Rotation der Eyes**

Basierend auf der Rotation der Eyes auf der y-Achse (Yaw) und der z-Achse (Roll) dreht sich auch das Gewehr im Spiel. Somit wird ermöglicht, dass sich der Benutzer des CAVEs drehen kann und das Gewehr immer in seine Blickrichtung zielt. Neigt er den Kopf leicht auf eine Seite, übernimmt das Gewehr ebenfalls diese Manipulation rollt sich auf die Seite.

1. **Relativer Winkel zwischen Eyes und Wand**

Zusätzlich zu der Blickrichtung, welche mittels Eyes festgestellt wird, folgt das Gewehr der aktuellen Cursorposition. Dazu wird der relative Winkel zwischen dem Wand und den Eyes berechnet und darauf basierend erfolgt eine zusätzliche Rotation des Gewehrs.

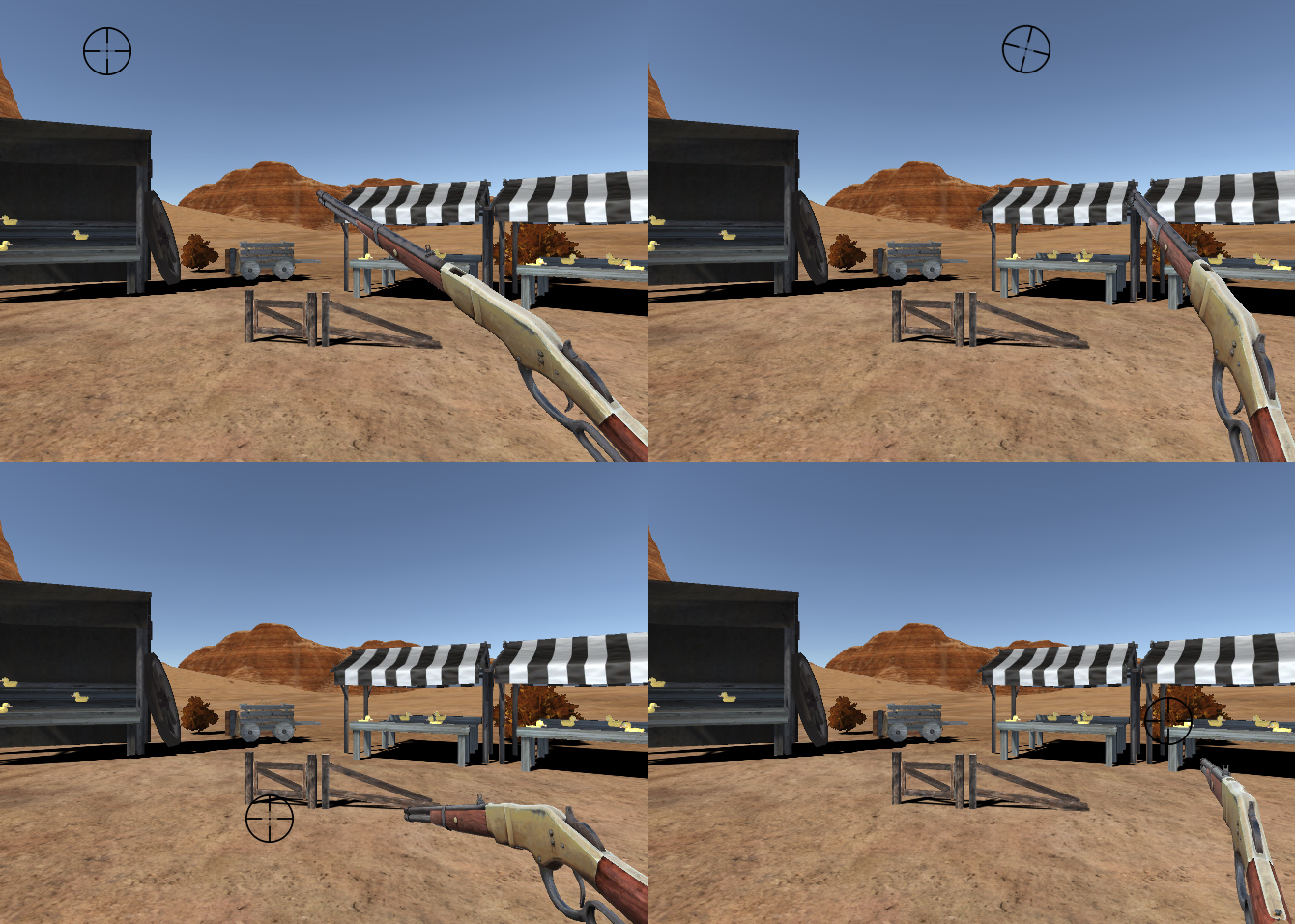


Abbildung : Shooting Gallery, Rotation des Gewehrs

**Schuss**

Dort wo sich der Cursor auf der 2D-Ebene befindet, wird auch präzise der Schuss in der 3D-Welt auftreffen. Dazu wird ein Raycast mit der Ausrichtung der Kamera an der Position des Cursors in die Umgebung geschossen und geschaut, welches Objekt als erstes im Wege steht.

Beim Auftreffen dieses virtuellen Schusses wird dem Ziel mitgeteilt, ob nun eine Interaktion erfolgen soll oder nicht. Zusätzlich wird ein Partikeleffekt, welcher den Einschuss verdeutlich, an der getroffenen Stelle erzeugt.



Abbildung : Shooting Gallery, Rauch

**Audio**

* **Zielscheibe**

Eines der beiden abzuschiessenden Ziele ist eine Holzkonstruktion mit drei Zielscheiben, welche sich zu zufälligen Zeitpunkten nach oben, bzw. nach unten klappen und somit angreifbar, bzw. nicht angreifbar werden.

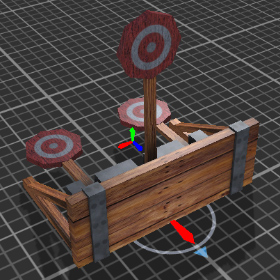


Abbildung : Shooting Gallery, Zielscheibe

Die Animationen sind im FBX-Model gespeichert und werden mittels einem AnimationController ausgelöst. Zu zufälligen Zeitpunkten wird eine der Show-States aktiviert um die Animation abzuspielen. Trifft während einer gewissen Zeitspanne kein Schuss die Zielscheibe, aktiviert sich der Hide-State und geht anschliessend zurück in den Idle-State. Im Gegenzug, trifft der Spieler auf die Zielscheibe, wird die Animation beim Hit-State abgespielt und es werden Punkte gutgeschrieben. Anschliessend fängt die Sequenz von vorne an.

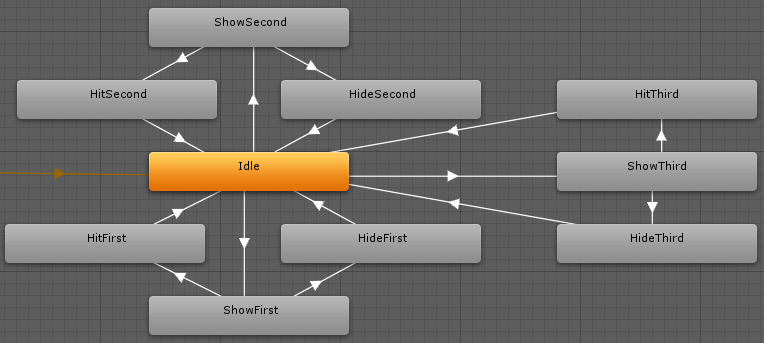


Abbildung : Shooting Gallery, Zielscheibe Animation Controller

* Ente
* Punktesystem
* Devices

## Model-Viewer

## App Drittpartei

# Abbildungsverzeichnis

[Abbildung 1: CAVE BFH 4](#_Toc434763800)

[Abbildung 2: Infrastruktur CAVE 5](#_Toc434763801)

[Abbildung 3: PPT Eyes, Quelle: www.worldviz.com 6](#_Toc434763802)

[Abbildung 4: PPT Wand, Quelle: www.worldviz.com 6](#_Toc434763803)

[Abbildung 5: Gamepad, Quelle: www.androidrundown.com 7](#_Toc434763804)

[Abbildung 6: Shooting Gallery Ingame 9](#_Toc434763805)

# Tabellenverzeichnis

[Tabelle 1: Et ut aut isti repuditis qui ium 3](#_Toc371572893)

# Glossar

**Auinweon**

Et ut aut isti repuditis qui ium 7

**Batnwpe**

Et ut aut isti repuditis qui ium 9

**Cowoll**

Et ut aut isti repuditis qui ium 11

# Literaturverzeichnis

**Literatureintrag**

*Autorname, Autorvorname, Buchtitel, Verlag, Ort, Ausgabe, Jahr* 7

**Literatureintrag**

*Autorname, Autorvorname, Buchtitel, Verlag, Ort, Ausgabe, Jahr* 9

**Literatureintrag**

*Autorname, Autorvorname, Buchtitel, Verlag, Ort, Ausgabe, Jahr* 11

# Anhang

Et ut aut isti repuditis qui ium nonsecturia quis incientiae laborem elliquis et quatur, sitiur aut od moluptatur aut ea conseque peri sim erro essequisit remporia dem et landi dest, cone poris quunt volecab ipidero quatur ad quibusamus.

# Selbständigkeitserklärung

Ich bestätige, dass ich die vorliegende Arbeit selbstständig und ohne Benutzung anderer als der im Literaturverzeichnis angegebenen Quellen und Hilfsmittel angefertigt habe. Sämtliche Textstellen, die nicht von mir stammen, sind als Zitate gekennzeichnet und mit dem genauen Hinweis auf ihre Herkunft versehen.

Ort, Datum:

Unterschrift: