

# Realisierung des Spieleklassikers "Archon"

mit 3D– und Webtechnologien

## Bachelorarbeit

im Fachgebiet Software-Engineering

zur Erlangung des akademischen Grades

Bachelor in Engineering



Vorgelegt von:	Kevin Dyes
Matrikelnummer:	2694420
Hochschule:	Technische Hochschule Georg-Simon-Ohm
Studienbereich:	Elektro– und Informationstechnik
Erstgutachter:	Prof. Dr. Röttger
Zweitgutachter:	Prof. Dr. Hopf



# Inhaltsverzeichnis

<b>Inhaltsverzeichnis</b>	<b>I</b>
<b>Vorwort</b>	<b>III</b>
<b>Erklärung</b>	<b>V</b>
<b>1 Einleitung</b>	<b>1</b>
1.1 Hinführung . . . . .	1
1.2 Aktueller Forschungsstand . . . . .	1
1.3 Motivation / Problemstellung . . . . .	1
1.4 Ziele dieser Arbeit . . . . .	2
1.5 Zentrale Begriffe . . . . .	2
1.5.1 Was versteht man unter Webtechnologien? . . . . .	2
1.5.2 Was versteht man unter 3D-Technologien? . . . . .	2
1.6 Aufbau . . . . .	2
<b>2 Hauptteil</b>	<b>5</b>
2.1 Einleitung . . . . .	5
2.2 Analyse . . . . .	5
2.2.1 Analyse des klassischen Spiels . . . . .	5
2.2.2 Anforderungsanalyse an neue Implementierung . . . . .	5
2.3 Implementierung . . . . .	5
2.3.1 Architektur . . . . .	6
2.3.2 benötigte Technologien und Frameworks . . . . .	6
2.3.3 Hilfsmittel und Vereinfachungen . . . . .	6
2.3.4 Schritte der Implementierung . . . . .	6
2.4 Resultate . . . . .	6
2.4.1 fertige Architektur . . . . .	6
2.4.2 Erfüllung der Anforderungen . . . . .	7
2.4.3 Überprüfung der Software mit Unit-Tests . . . . .	7
2.5 Fazit . . . . .	7
<b>3 Fazit und Ausblick</b>	<b>9</b>
3.1 Fazit . . . . .	9
3.1.1 erreichte Ziele . . . . .	9
3.2 Ausblick . . . . .	9

<b>Verzeichnisse</b>	<b>VII</b>
Literaturverzeichnis . . . . .	VII
Abbildungsverzeichnis . . . . .	IX
Tabellenverzeichnis . . . . .	XI
Quellcodeverzeichnis . . . . .	XIII
Abkürzungsverzeichnis . . . . .	XV
Stichwortverzeichnis . . . . .	XIX

## **Vorwort**

Im Jahr 2017 wurden mehr als 1,8 Milliarden browser-fähige Endgeräte weltweit verkauft.



## Erklärung

Ich, Kevin Dyes / Matrikel Nummer 2694420, versichere hiermit, dass ich diese Bachelorarbeit mit dem Thema

*Realisierung des Spieleklassikers "Archon"*  
*mit 3D- und Webtechnologien*

selbständig verfasst und keine anderen als die angegebenen Quellen und Hilfsmittel benutzt habe, wobei ich alle wörtlichen und sinngemäßen Zitate als solche gekennzeichnet habe. Die Arbeit wurde bisher keiner anderen Prüfungsbehörde vorgelegt und auch nicht veröffentlicht.

Nürnberg, den 8. Mai 2018

---

*Kevin Dyes*





# 1 Einleitung

## 1.1 Hinführung

*Die Hinführung soll den Leser an das Thema 3D-Technologien, Web-Entwicklung und Spieleklassiker, wie Archon heranzuführen. Hier sind Quellen und Recherchen zu ein paar Zahlen, wie Remakes, Umsätze zu Uralt-Spielen aktuell und Beliebtheit/Zahlen zum Webbrowser als Spieleplattform angebracht, um Wichtigkeit der Plattform und Technologien, aber auch Relevanz für Spieleklassiker zu zeigen.*

Fliegt vermutlich raus und wird in Vorwort integriert.

## 1.2 Aktueller Forschungsstand

*Hier sollen dann Extrema zum einen in Richtung 3D-Entwicklung, in Richtung Web-Entwicklung und aber auch bei der Spiele-Entwicklung kurz herausgearbeitet werden, um dem Leser einen kurzen Einblick in aktuelle Möglichkeiten zu geben, und zu zeigen, dass die Arbeit "State of the Art" ist.*

Fliegt vermutlich raus, oder kommt an andere Position.

## 1.3 Motivation / Problemstellung

*Diese Sektion soll vom vorherigen Stand der Forschung und Möglichkeiten auf Anwendbarkeit überleiten und zeigen, dass die evtl. surreal erscheinende Vermischung von Genres, Plattformen und Technologien anwendbar ist, und das sogar mit beschränktem Aufwand. Vereinbarkeit von aktuellen Technologien mit Spiele-Klassikern und Spiele-Klassiker als Anwendungsbeispiele dafür.* Versionen, Ableger und Kopien des Spieleklassikers Archon gibt es zahlreich. Meist sind diese jedoch auf ein System beschränkt, oder tatsächlich nur mit einem Emulator ausführbar. Das Web leistet an dieser Stelle die perfekte Abhilfe: Es gibt unzählige Geräte, die heutzutage einen Webbrowser installiert haben, somit hat jeder Zugriff auf diesen Ableger, der ein webfähiges Gerät besitzt.

## 1.4 Ziele dieser Arbeit

*Hier soll bloß kurz und prägnant die Fragestellung genannt werden und eine kurze Analyse der einzelnen Worte des Titels stattfinden. Ein Satz, wie "Die exakten Anforderungen für die Umsetzung dieser groben Ziele werden zu einem späteren Zeitpunkt herausgearbeitet", oder "... Anforderungen werden aus dem Aufbau der Arbeit heraus klar." muss hier hinein.*

## 1.5 Zentrale Begriffe

### 1.5.1 Was versteht man unter Webtechnologien?

Webtechnologien betiteln die Sammlung aller nötigen Aspekte zum Erstellen einer Webanwendung. Webanwendungen selbst bestehen aus einem Client-Server-Modell. Typische Bestandteile des Clients sind:

- HTML zur Beschreibung des Inhalts
- CSS zur Beschreibung des Aussehens
- JavaScript zur Dynamisierung des Clients

Ein Webserver gibt auf HTTP-Anfragen die entsprechenden Inhalte und Medien an den Client heraus, welcher durch einen Webbrowser angezeigt wird.

### 1.5.2 Was versteht man unter 3D-Technologien?

Da ein Bildschirm, beispielweise eine Computers, nur zweidimensional ist, muss durch andere Methodiken der Effekt einer dritten Dimension geschaffen werden. Hier kommen 3D-Technologien zum tragen, die mit Hilfe von Berechnungen, Überlagerungen o.ä. Licht bzw. Schatten und damit Tiefe in eine 3D-Szenerie bringen.

## 1.6 Aufbau

*Hier wird dann auf den Aufbau des Haupt- und Schlussteils eingegangen. Also der erste Abschnitt beschäftigt sich mit Anforderungsanalyse beider Plattformen/Seiten. Der zweite Abschnitt dreht sich dann rund um Zusammenführung der Anforderungen und Tool-Auswahl, und somit darum einen Einstiegspunkt und eine Architektur für die weitere Entwicklung festzulegen.*

### 1.6 Aufbau

---

Im ersten Teil dieser Arbeit sollen die Anforderungen an eine Neuauflage von "Archon" erfasst werden und eine Beschreibung der Features und Spielmechaniken zur Umsetzung erstellt werden. Anschließend werden die Anforderungen der technologischen Seite herausgearbeitet, sodass eine Liste an Funktionalitäten entsteht, die von den 3D- und Webtechnologien erfüllt werden muss. Alle Anforderungen werden dann in einer Beschreibung zur Umsetzung und einer entsprechenden Architektur abgearbeitet. Den Abschluss bildet eine Überprüfung der fertigen Neuauflage auf die Erfüllung der Anforderungen, sowie eine Präsentation des Ergebnisses.



## 2 Hauptteil

### 2.1 Einleitung

### 2.2 Analyse

The goal of every video game is to present the user(s) with a situation, accept their input, interpret those signals into actions, and calculate a new situation resulting from those acts. Games are constantly looping through these stages, over and over, until some end condition occurs (such as winning, losing, or exiting to go to bed). Not surprisingly, this pattern corresponds to how a game engine is programmed. The specifics depend on the game. [13]

*In diesem Abschnitt soll die Vorgehensweise bei der Analyse kurz aufgezeigt werden und der Inhalt der einzelnen Teile klar gemacht werden. Also, zum einen, dass hier eben eine Anforderungsanalyse stattfindet, und zum Anderen, dass diese Analyse zweigeteilt ist.*

#### 2.2.1 Analyse des klassischen Spiels

*Hier werden alle Aspekte des klassischen Spiels herausgearbeitet, gegliedert nach Teilen: Generelle Regeln, das Board, die Figuren, Abläufe und Wirkungen, mögliche Einstellungen*  
*Stichwort: Komponentendiagramm!*

#### 2.2.2 Anforderungsanalyse an neue Implementierung

*Hier wird analysiert, welche Aspekte das Spiel von der Seite der technologisch gestellten Anforderungen erfüllen muss. Also: Es muss eine 3D- Engine geben, die irgendwie im Browser renderbar ist.*

*Es muss möglich sein, alle Spiel-Mechaniken mittels Server-Client-Kommunikation umzusetzen, etc.*

*Stichwort: Komponentendiagramm!*

### 2.3 Implementierung

*Hier soll dann aufgezeigt werden, welche Anforderungen vom Spiel durch welche Technologie umgesetzt werden und nötige Bedingungen/grundsätzliche Ausschlüsse aufgezeigt werden.*

*Hier, oder im nächsten Abschnitt muss klar gestellt werden, dass ein KI-Modus nicht implementiert wird!*

### 2.3.1 Architektur

*Aus dem Teil der Zusammenführung ergibt sich dann zwangsläufig eine Art Komponentenarchitektur, die hier verfeinert wird, bis zu dem Punkt, an dem eine Festlegung auf Frameworks und Technologien getätigt werden muss.*

### 2.3.2 benötigte Technologien und Frameworks

*Hier wird dann von den benötigten Technologien die Festlegung auf eine bestimmte Implementierung getroffen, also Frameworks, Programmiersprachen, Toolsets etc. festgelegt.*

*Hier wird dann die Komponentenarchitektur vom vorherigen Teil weiter verfeinert, sodass evtl schon eine grobe Klassenarchitektur dabei herauspringen kann.*

### 2.3.3 Hilfsmittel und Vereinfachungen

*Hier würden Dinge zu lesen sein, wie die Benutzung von TypeScript, oder Browser-Beschränkungen etc, da diese nicht relevant für den eigentlichen Entwicklungsvorgang sind, aber dennoch nützlich sind und eben Vereinfachungen darstellen.*

### 2.3.4 Schritte der Implementierung

*Hier wird dann der Entwicklungsprozess kurz erläutert, also Vorgehen, wie Aufsetzen der Tool-Chain, dann Entwicklung einer ersten Darstellung (Frontend), um direkte Erfolge zu sehen, etc pp.*

## 2.4 Resultate

*Hier soll dann ein Screenshot des Ergebnisses rein und erläutert werden, dass als nächste der Endstand mit seiner Architektur gezeigt wird und anschließend die Erfüllung aller Anforderungen sichergestellt wird. Als letztes (falls genug Zeit!) werden die (hoffentlich) programmierten Unit-Tests erwähnt, und deren Ergebnisse dargestellt.*

### 2.4.1 fertige Architektur

*Die fertige Architektur kann und soll durchaus von der geplanten Abweichen und das schlussendliche Ergebnisse wird hier in Form von Diagrammen gezeigt, die dann einzeln erklärt werden.*

### 2.4.2 Erfüllung der Anforderungen

*Hier werden die Anforderungen aus dem Analyse-Teil aufgegriffen und mit der fertigen Anwendung und ihrer Architektur abgeglichen, also so was wie "die einzelnen Figuren und ihre Unterschiede, sind hier und hier da und da durch umgesetzt worden."*

### 2.4.3 Überprüfung der Software mit Unit-Tests

*Hier wird dann die breite der Unit-Tests gezeigt, deren Anzahl und die Beschränkungen, also Code-Abdeckung. Anschließend ein Ergebnis-Log von einem Lauf auf dem finalen Stand.*

## 2.5 Fazit





## 3 Fazit und Ausblick

*Die Ergebnisse wurden bisher nur dargestellt und erklärt und nicht bewertet und analysiert, dass soll hier geschehen unter "erreichte Ziele". Das Fazit soll Punkte zu Anwendbarkeit der Technologien, Entwicklung/Nachbau eines Spieleklassikers und Problempunkte, aber auch positives zu Support und Inbetriebnahme enthalten. Der Ausblick soll kommende Technologien und weitere Entwicklungspunkte für das Spiel beleuchten.*

### 3.1 Fazit

#### 3.1.1 erreichte Ziele

### 3.2 Ausblick

*Hier kann auch noch in der bisher fehlende KI-Modus in Aussicht gestellt werden.*



## Literaturverzeichnis

- [1] BOWDEN KELLY: *TypeScript Node Starter*. <https://github.com/Microsoft/TypeScript-Node-Starter>. Version: 0.1.0, Abruf: 31.03.2018
- [2] CONTRIBUTORS, Mozilla: *MDN-Web-Dokumentation*. <https://developer.mozilla.org/de/>, Abruf: 31.03.2018
- [3] EXPRESSJS.COM CONTRIBUTORS: *express.js*. <https://expressjs.com/>. Version: 4.16.2, Abruf: 31.03.2018
- [4] [HTTPS://GITHUB.COM/MRDOOB](https://github.com/MRDOOB): *three*. <https://threejs.org/>. Version: 0.91.0, Abruf: 31.03.2018
- [5] [HTTPS://GITHUB.COM/SOCKETIO/SOCKET.IO/GRAPHS/CONTRIBUTORS](https://github.com/socketio/socket.io/graphs/contributors): *Socket.IO*. <https://socket.io/>. Version: 2.1.0, Abruf: 31.03.2018
- [6] IDC: *Absatz von Tablets, PCs und Smartphones weltweit von 2010 bis 2017 und Prognose für 2022 (in Millionen Stück)*. <https://de.statista.com/statistik/daten/studie/256337/umfrage/prognose-zum-weltweiten-absatz-von-tablets-pcs-und-smartphones/>, Abruf: 31.03.2018
- [7] MEDARCH: *The Secrets Of Archon*. <http://www.vintagecomputing.com/index.php/archives/44>. Version: 1.0, Abruf: 31.03.2018
- [8] MICROSOFT: *TypeScript - JavaScript that scales*. <https://www.typescriptlang.org/index.html>. Version: 2.8, Abruf: 31.03.2018
- [9] MICROSOFT: *Visual Studio Code*. <https://code.visualstudio.com/>, Abruf: 31.03.2018
- [10] NODE.JS FOUNDATION: *node.js*. <https://nodejs.org/en/>. Version: 9.4.0, Abruf: 31.03.2018
- [11] NYSTROM, Bob: *Game Programming Patterns*. 1. Auflage 2014. Genever Benning, 2014 <http://gameprogrammingpatterns.com/>. – ISBN 0990582906
- [12] THE JQUERY FOUNDATION: *jQuery - Webportal*. <http://jquery.com/>, Abruf: 31.03.2018
- [13] US/DOCS/GAMES/ANATOMY\$HISTORY <https://developer.mozilla.org/en-US/docs/Games/Anatomy> *Anatomy of a video game*. <https://developer.mozilla.org/en-US/docs/Games/Anatomy>
- [14] W3SCHOOLS, Refsnes Data: *W3Schools Online Web Tutorials*. <http://www.w3schools.com/>. Version: 2015, Abruf: 11.02.2015
- [15] WIKIPEDIA: *Archon (Computerspiel)*. [https://de.wikipedia.org/wiki/Archon\\_\(Computerspiel\)](https://de.wikipedia.org/wiki/Archon_(Computerspiel)). Version: Dezember 2014, Abruf: 31.03.2018



# **Abbildungsverzeichnis**



# **Tabellenverzeichnis**





# Quellcodeverzeichnis



## Abkürzungsverzeichnis

Abb. ....	Abbildung
AJAX ....	ermöglicht asynchronen Datenaustausch mit z. B. Webservern (engl. für Asynchronous JavaScript and XML)
API ....	Programmschnittstelle nach aussen (engl. für Application Programming Interface)
ASCII ....	7-Bit Zeichencodierung (engl. für American Standard Code for Information Interchange)
AWL ....	Anweisungsliste, Assembler ähnlich
Code-Folding .....	logisch zusammengehörende Quelltextabschnitte werden in Abschnitte gruppiert, um diese einfach ein- bzw. auszublenden, erhöht die Lesbarkeit und Übersichtlichkeit
CPU ....	Elektronischer Rechner (engl. für Central Processing Unit)
CR ....	Wagenrücklauf (engl. Carriage Return)
CRC ....	zyklische Redundanzprüfung (engl. Cyclic Redundancy Check)
CSS ....	gestufte Gestaltungsbögen, legt die Darstellung des HTML Quellcodes im Browser fest (engl. für Cascading Style Sheets)
DB ....	Datenbaustein, Baustein zur Datenhaltung
DOM ....	Dokumentstruktur der Webseite (engl. für Document Object Model)
FB ....	Funktionsbaustein, wie FC nur mit Gedächtnis (Speicher in Form eines Datenbaustein
FC ....	Funktion
Field PG .....	Spezieller Laptop (Computer) für industrielle Umgebungen zum Programmieren einer Steuerung
FUP ....	Funktionsplan, Digitalen Logik Gattern ähnlich
GIF ....	Grafikaustausch Format, Animationsfähig (engl. für Graphics Interchange Format)
GUI ....	Grafisches Benutzer Interface (engl. für Graphical User Interface)
HAP ....	HTML Agility Pack, C#HTML Bibliothek zum verarbeiten von Webdokumenten
HMI ....	Mensch-Maschine Interface (engl. für Human Machine Interface)
HTML ....	Hypertext Auszeichnungssprache, HTML-Dateien sind die Grundlage des World Wide Web und werden von einem Webbrowser dargestellt (engl. für Hypertext Markup Language)

---

HTTP	.....	Hypertext-Übertragungsprotokoll, Protokoll zur Übertragung von Daten über ein Netzwerk (engl. für Hypertext Transfer Protocol)
HTTPS	.....	sicheres Hypertext-Übertragungsprotokoll s. a. HTTP (engl. für Hypertext Transfer Protocol Secure)
IDE	.....	integrierte Entwicklungsumgebung (engl. für Integrated Development Environment)
Interface	.....	Schnittstelle
IP	.....	Internet Protocol
IPC	.....	Industrie PC - Computer für Industriellen Einsatz
Java-Applet	.....	Hilfsprogramm oder Tool, was in eine Webseite integriert wird
JavaScript	.....	Skriptsprache, ursprünglich für dynamisches HTML in Webbrowsern entwickelt
JPEG	.....	komprimierte Grafikdatei, auch JPG (engl. für Joint Photographic Expert Group)
JSON	.....	kompaktes Datenformat zum Datenaustausch mit z. B. Webservern (engl. für JavaScript Object Notation)
KOP	.....	Kontaktplan, Schaltplan ähnlich
LF	.....	Zeilenvorschub (engl. Line Feed)
LRC	.....	Längsparitätsprüfung (engl. Longitudinal Redundancy Check)
Mockup	.....	Attrappe oder auch rudimentärer Prototyp (auch Maquette)
MSDN	.....	Das Microsoft Entwickler Netzwerk (engl. für MicroSoft Developer Network)
MWSL	.....	Mini Web Server Language, Serverbasierende Skriptsprache
o.V.	.....	ohne Verfasser (bei Literaturverweisen)
OB	.....	Organisationsbaustein
OS	.....	das Betriebssystem (engl. für Operating System)
Parser	.....	engl. to parse - „analysieren“ bzw. lateinisch pars - „Teil“ im Deutschen gelegentlich auch Zerteiler, Analysiert die Semantik des Scripts um daraufhin Aktionen durchzuführen
PC	.....	Elektronischer Rechner (engl. für Personal Computer)
PG	.....	Programmier Gerät, meist ein PC
PLC	.....	s. a. SPS (engl. für Programmable logic controller)
PLCVarTab	.....	Variablentabelle (Symboltabelle)
PN	.....	s. a. Profinet
PNG	.....	Grafikaustausch Format (engl. für Portable Network Graphics)

---

PROFINET	.....	Bezeichnung für industriellen Netzwerkstandard (engl. für Process Field Network)
SCL	.....	Strukturierter Text (engl. für Structured Control Language)
SDK	.....	s. a. IDE (engl. für Software Development Kit)
SFB	.....	System Funktionsbaustein
SFC	.....	System Funktion
SOP	.....	Same Origin Policy
SPS	.....	Speicher Programmierbare Steuerung
ST	.....	Strukturierter Text, s. a. SCL
SVG	.....	skalierbare Vektorgrafik (engl. für Scalable Vector Graphics)
TCP	.....	Transmission Control Protocol
TIA	.....	Totally Integrated Automation
TN	.....	Teilnehmer
UDT	.....	Benutzerdefinierter Datentyp in Form einer Struktur (engl. für User Defined Typ)
URI	.....	Uniform Resource Identifier
URL	.....	Uniform Resource Locator
URN	.....	Uniform Resource Name
W3C	.....	Organisation zur Standardisierung von Webtechnologien (engl. für World Wide Web Consortium)
Webbrowser	.....	auch kurz Browser (engl. to browse) steht für durchstöbern, abgrasen, durchsuchen - Software zum Darstellen von Daten, hauptsächlich Webseiten und deren Inhalt, können zu diesem Zweck mit Webservern kommunizieren
Webseite	.....	s. a. HTML-Datei
Webserver	.....	Ein Webserver speichert Webseiten und stellt diese zur Verfügung. Der Webserver ist eine Software, die Dokumente mit Hilfe standardisierter Übertragungsprotokolle (HTTP, HTTPS) an einen Webbrowser überträgt.  In einer CPU mit PROFINET-Schnittstelle ist ein Webserver integriert, der mit anwenderdefinierten Webseiten erweiterbar ist
WPO	.....	Webdaten Optimierung (engl. für Web Performance Optimization)
WWW	.....	Internet (engl. für World Wide Web)
WYSIWYG	.....	Man sieht im Editor sofort was man bekommt, sowohl textuell als auch grafisch dargestellt (engl. für what you see is what you get)



## Stichwortverzeichnis

Das Verzeichnis ist in Haupt- und Unterbegriffe gegliedert. Ist ein Stichwort nicht unter den Hauptbegriffen gelistet, so ist es womöglich als Untereintrag zu finden.

Analyse, 5

Analyse neue Implementierung, 5

Architektur, 6

Aufbau, 2

Ausblick, 9

Erfüllung, 7

Fazit, 7, 9

Forschungsstand, 1

Hilfsmittel, 6

Hinführung, 1

Implementierung, 5, 6

Motivation, 1

Resultate, 6

Spiel Analyse, 5

Technologien, 6

Unittests, 7

Zentrale Begriffe, 2

Ziele, 2