

Fakultät Elektrotechnik Feinwerktechnik Informationstechnik

Prüfungsstudienarbeit von
Sebastian Adam
Matr. – Nr.: 210 81 33
Bachelor Media Engineering 6. Semester



Prüfungsstudienarbeit
für das Projekt „ReMasterBlaster“
im Rahmen der Projektarbeit im 6. Semester
des Studiengangs Media Engineering

Sommersemester 2012

Bestätigung gemäß § 35 (7) RaPO

Adam, Sebastian

Ich bestätige, dass ich die Prüfungsstudienarbeit mit dem Titel:

Prüfungsstudienarbeit
für das Projekt „ReMasterBlaster“
im Rahmen der Projektarbeit im 6. Semester
des Studiengangs Media Engineering

selbständig verfasst, noch nicht anderweitig für Prüfungszwecke vorgelegt, keine anderen als die gegebenen Quellen oder Hilfsmittel benutzt, sowie wörtliche und sinngemäße Zitate als solche gekennzeichnet habe.

Datum: 03.08.12

Unterschrift:

“Abstract

Gemäß Paragraph 21, Absatz Eins der Rahmenprüfungsordnung, kurz RaPo, sind Prüfungsstudienarbeiten Prüfungsleistungen mit überwiegend zeichnerischem, gestalterischem oder sonstigem komplexen Inhalt und offenem Lösungsweg zum Nachweis kreativer Fähigkeiten, die sich wegen der umfassenden Aufgabenstellung und der Art der Aufführung in der Regel über einen längeren Zeitraum erstrecken.

Die einzelnen Prüfungsstudienarbeiten des Projektteams, bestehend aus Sebastian Adam, Sergej Bjakow, Michael Kao, Pavlina Pavlova und Maximilian Seyfert, welches sich mit der Realisierung eines Remakes des AMIGA Klassikers MasterBlaster mit neuen Webtechnologien (HTML5, CSS, JavaScript) befasst hat, sollen einen individuellen Einblick auf die jeweiligen Anforderung, die es mit Lösungsstrategien zu bewerkstelligen galt, gewährleisten.

Diese, meine eigene Studienprüfungsarbeit, will Auskunft über die Bedeutung der Rolle meiner Person in der gemeinsamen Teamarbeit verdeutlichen.“

Projekt-Roadmap

Tätigkeit	Dokument	Beteiligte
Erstellung des Lastenhefts	02_documents/02_requirements/ Lastenheft1_1.pdf	Sebastian Adam
Erstellung des Pflichtenhefts	02_documents/02_requirements/ Pflichtenheft_v1_1.pdf	Sebastian Adam
Erstellung des Zeitplans	02_documents/03_timeline/ Remasterblaster.pdf	Sebastian Adam
Erstellung des Protokolls	02_documents/01_protocols/ ...Protokoll_01.pdf	Sebastian Adam
Erstellung des Protokolls	02_documents/01_protocols/ ...Protokoll_02.pdf	Sebastian Adam
Erstellung des Protokolls	02_documents/01_protocols/ ...Protokoll_03.pdf	Sergej Bjakow
Erstellung des Protokolls	02_documents/01_protocols/ ...Protokoll_04.pdf	Sebastian Adam
Erstellung des Protokolls	02_documents/01_protocols/ ...Protokoll_05.pdf	Maximilian Seyfert
Erstellung des Protokolls	02_documents/01_protocols/ ...Protokoll_06.pdf	Michael Kao
Erstellung des Protokolls	02_documents/01_protocols/ ...Protokoll_07.pdf	Sebastian Adam
Erstellung des Protokolls	02_documents/01_protocols/ ...Protokoll_08.pdf	Sebastian Adam
Erstellung des Protokolls	02_documents/01_protocols/ ...Protokoll_09.pdf	Maximilian Seyfert
Erstellung des Protokolls	02_documents/01_protocols/ ...Protokoll_10.pdf	Sebastian Adam
Erstellung des Protokolls	02_documents/01_protocols/ ...Protokoll_11.pdf	Sergej Bjakow
Erstellung des Protokolls	02_documents/01_protocols/ ...Protokoll_12.pdf	Sebastian Adam
Erstellung des Protokolls	02_documents/01_protocols/ ...Protokoll_13.pdf	Sebastian Adam
Erstellung der Testbench mit der Sprite – Engine	01_development/01_testbench/*	Pavlina Pavlova
Erstellung der ersten Testumgebung	01_development/02_testenvironment/*	Michael Kao Pavlina Pavlova
Programmierung der Spielelogik	01_development/03_final/game.js	Michael Kao
Erstellung der Sprite Zuordnung	01_development/03_final/sprites.js	Michael Kao
Programmierung des Graphical User Interface	01_development/03_final/gui.js	Pavlina Pavlova
Erstellung der Sprites – Quelldatei	04_material/01_rawfiles/ 01_sprites_psd/*	Michael Kao Sergej Bjakow

Modifikation der Sounds zu mp3 Dateien	01_development/03_final/sounds/*	Sergej Bjakow Maximilian Seyfert
Capturing der einzelnen Elemente (Sprites)	01_development/03_final/img/*	Sergej Bjakow
Zusammenfügen der Sprites	04_material/01_rawfiles/ 01_sprites_psd/*	Michael Kao
Vorbereiten der einzelnen Szenenelemente	01_development/03_final/img/*	Pavlina Pavlova Sergej Bjakow Michael Kao
Erstellung der Zwischenpräsentation	03_presentations/ 2012-05-09_ReMasterBlaster.pdf	Sebastian Adam
Erstellung der Abschlusspräsentation	03_presentations/ 2012-07-20_ReMasterBlaster.pdf	Sebastian Adam Michael Kao Pavlina Pavlova
Erstellung eines Plakats für die Abschlusspräsentation	04_material/04_poster/ extras_plakat.pdf	Sebastian Adam
Erstellung eines Plakats für die Abschlusspräsentationen	04_material/04_poster/ project_presentation.pdf	Michael Kao
Erstellung des Wikis	http://schorsch.efi.fh-nuernberg.de/mewiki/index.php/Project-BME-2012-07-RemasterBlaster/Project-BME-2012-07-RemasterBlaster	Sebastian Adam
Erstellung der Gesamtdokumentation	02_documents/04_documentation/ 02_overalldocumentation/ Gesamtdokumentation.pdf	Sebastian Adam

Die mit **blau** markierten Zeilen stellen die Arbeitsbereiche mit eigener Arbeitsleistung dar.

INHALTSVERZEICHNIS

BESTÄTIGUNG GEMÄß § 35 (7) RAPO	2
“ABSTRACT	3
PROJEKT-ROADMAP	4
1. AUSGANGSSITUATION	7
2. PROJEKTIDEE	7
3. ROLLENVERGABE	8
4. PROJEKTORGANISATION	8
4.1 Meetings	9
4.2 Blog	9
4.3 Zeitplanung	10
4.4 Facebook Gruppe	12
4.5 Google Kalender	13
5. DEFINITION DES LASTEN- UND PFLICHTENHEFTS....	13
5.1 Muss - Kriterien	14
5.2 Soll - Kriterien	14
5.3 Kann - Kriterien	15
6. VERSIONSVERWALTUNG MIT GITHUB	15
7. SOLL – IST ANALYSE	16
8. RESÜMEE	17
ABBILDUNGS- UND QUELLENVERZEICHNIS	20
ANHANG	21

1. Ausgangssituation

Ausgangssituation für die Projektarbeit „ReMasterBlaster“ war die für das 6. Semester im Bachelorstudiengang Media Engineering angesetzte Projektarbeit. Als Auftraggeber traten in unserem Fall Prof. Dr. Matthias Hopf sowie Prof. Dr. Stefan Röttger in Erscheinung. Der zeitliche Rahmen umfasste den Zeitraum vom 26.03.2012 bis letztendlich 03.08.2012. Das Projektteam für das Projekt „ReMasterBlaster“ setzte sich aus folgenden Mitgliedern zusammen: Sebastian Adam, Sergej Bjakow, Michael Kao, Pavlina Pavlova und Maximilian Seyfert.

2. Projektidee

Die ursprüngliche Projektidee stammte von Prof. Dr. Matthias Hopf, welcher auf Nachfrage von mir, Sebastian Adam, das Projektthema zur Verfügung stellte. Hintergrund des Ganzen war, dass der Fokus auf Reverse Engineering¹ gelegt werden sollte. Hierbei gilt es ein bereits vorhandenes Produkt – in unserem Fall der AMIGA Klassiker MasterBlaster – in seine einzelnen Bestandteile zu zerlegen und nachzukonstruieren. Es sollte hierbei jedoch nicht wieder ein Spiel für die AMIGA entstehen, viel mehr wollte das Projektteam das Spiel für die breite Masse zugänglich machen. Also entschieden wir uns für die Umsetzung mittels neuer Webtechnologien im Internetbrowser der jedem Endanwender heutzutage zur Verfügung steht. Hierbei kamen die Markup Language HTML5², CSS³ und JavaScript⁴ sowie die Bibliothek jQuery⁵ und die Crafty JavaScript Sprite Engine⁶ zum Einsatz.

¹ Eine genauere Definition finden Sie hier:

http://de.wikipedia.org/wiki/Reverse_engineering

² Hyper Text Markup Language, eine Auszeichnungssprache im Internet

³ Cascading Style Sheets, eine Möglichkeit HTML-Inhalte zu layouten

⁴ eine Skriptsprache, die auf der Clientseite im Browser abläuft

⁵ populäre JavaScript Bibliothek, www.jquery.com

⁶ die von uns verwendete Sprite Engine, mehr Infos unter www.craftyjs.com

3. Rollenvergabe

Die Rollenvergabe soll erzielen komplexe Aufgaben, die sich über einen längeren Zeitraum erstrecken, zu strukturieren und somit die Arbeit innerhalb des Projektteams zu erleichtern sowie einen harmonischen Arbeitsfluss zu gewährleisten.

Das Team der Projektgruppe „ReMasterBlaster“ entschied sich zu Beginn des Projekts ebenfalls eine Rollenvergabe vorzunehmen. Als Programmierer der Spielelogik traten Michal Kao sowie Maximilian Seyfert in Erscheinung. Pavlina Pavlova wurde dem GUI⁷ zugeteilt und Sergej Bjakow kümmerte sich größtenteils um den Teil des Reverse Engineerings um unser neues Spiel mit möglichst allen Details nahe am Original erscheinen zu lassen. Ich selbst wurde nach einer kurzen Besprechung mit allen Teilnehmern zum Projektleiter ernannt. Diese Rolle ist in einem Projekt natürlich von essentieller Bedeutung, da der Projektleiter die Schnittstelle zu den betreuenden Professoren darstellt und für die gesamte Organisation innerhalb des Projektgeschehens zuständig ist.

Um mich mit dieser neuen Rolle vertraut zu machen konnte ich auf Unterlagen aus Vorlesungen sowie auf Literatur aus der Bibliothek der Hochschule zurückgreifen. Ich beschäftigte mich intensiv mit den Aufgaben eines Projektleiters, da ich solch eine Rolle noch nie zuvor inne hatte.

4. Projektorganisation

Eine gelungene Projektdurchführung ist gekennzeichnet von einer akribischen Projektorganisation. Dies erforderte in unserem Fall regelmäßig veranstaltete Treffen, die Errichtung eines Entwicklungsblogs sowie zu Beginn des Projekts die Einrichtung eines öffentlich (für alle

⁷ Graphical User Interface

Teammitglieder) zugänglichen Ordners innerhalb der Dropbox⁸. Diese gemeinsamen Meetings und Dokumentations- bzw. Organisationstools werde ich in diesem Kapitel näher erläutern.

4.1 Meetings

Zur optimierten Kommunikation sowie zur Stärkung und Verbesserung der Teamarbeit wählte ich die Option, dass wir wöchentlich mindestens ein gemeinsames Treffen abhielten. Hier wurden jeweils von den Teammitgliedern die neuesten Arbeitsergebnisse vorgestellt und wiederum neue Aufgaben verteilt. Die Aufgabe eines Jeden von uns war es zudem abwechselnd Sitzungsprotokolle dieser wöchentlich stattfindenden Teammeetings zu erstellen, um sich auch rückblickend über die einzelnen Treffen informieren zu können. Diese Protokolle wurden immer zeitnah nach dem eigentlichen Treffen den anderen Mitgliedern in digitaler Form zur Verfügung gestellt. Einen Großteil der Protokollaktivitäten übernahm im Laufe des Projekts ich, in der Rolle des Projektleiters.

Gegen Ende des Projekts – wo erfahrungsgemäß die Zeit immer knapper wird und noch dazu die reguläre Prüfungszeit ansteht – führte ich zudem sogenannte „Projektstage“ ein bei denen gezielt eine Aufgabe bis zur Vervollständigung verfolgt wurde. Hier ist z.B. das „mergen“⁹ der beiden Programmteile (Spiellogik und GUI) zu erwähnen.

4.2 Blog

Da das Projekt „ReMasterBlaster“ ein Open Source Projekt darstellt sollten auch die Fortschritte, die im Projekt getätigt wurden, öffentlich für Jedermann einsehbar sein. Hier entschieden wir uns dazu einen Blog einzurichten. Den Webpace der dafür nötig war stellte ich von meinem

⁸ Eine kostenlos angebotene Cloud Speicherlösung im World Wide Web

⁹ zusammenfügen von einzelnen, vorher unabhängig voneinander entwickelten, Programmteilen

privaten Account bei einem Hoster für das Projekt zur Verfügung. Wir entschieden uns jedoch zusätzlich eine „internationale“ Domain für den Zeitraum des Projekts anzumieten. Hier entschieden wir uns für den Namen: <http://www.remasterblaster.com> – wie innovativ! Als Blogsystem kam Wordpress zum Einsatz. Ich passte das Layout und das Design des Blogs ein wenig an unsere Bedürfnisse an und von nun an war es meine Aufgabe einmal wöchentlich über das Projektgeschehen in englischer Sprache zu informieren. Diese Aufgabe war auch deswegen sehr hilfreich für mich, da ich dadurch öfter als nur einmal (beim wöchentlichen Teammeeting) mit den einzelnen Teammitgliedern in Kontakt trat um den aktuellen Stand der Entwicklungsarbeiten zu erfragen. Diese Tatsache hat mir sehr geholfen den Überblick über das gesamte Projekt zu bewahren. Somit konnten nicht nur die Leser des Blogs, sondern auch ich in meiner Tätigkeit als Projektleiter, von unserem Entwicklungsblog profitieren.

4.3 Zeitplanung

Die Zeitplanung ist wohl eine der Wichtigsten Aufgaben eines Projektleiters. Hierbei versuchte ich mich an mehreren „Projektplanungstools“ von denen viele jedoch eher enttäuschten bzw. nach einiger Zeit kostenpflichtig waren. Um die Zeitplanung für unser Projekt letztendlich durchzuführen nutzte ich das Open Source Tool „Gantt Project“, da dieses das einzige kostenlose Tool war, welches mich einigermaßen zufrieden stellte. Im weiteren Verlauf werde ich kurz die einzelnen Phasen des Projektes vorstellen und anschließend den Gantt – Chart¹⁰ der aus diesen Phasen und Teilaufgaben resultierte erläutern.

1. Planungsphase (26.03. – 10.04.2012)

Einarbeitung in den Bereich der Sprite Engines

Erstellung einer Testumgebung mit diversen Sprite Engines

Definition des Lastenheftes

Erstellung eines Projektablaufplans

¹⁰ Zeitlinie über das gesamte Projekt

2. Entwicklungsphase (10.04. – 03.07.2012)

Entwicklung einer Spielumgebung anhand des Originals
Implementierung der Bewegungsfunktion eines Spielers
Einbindung der Goodies nach einer zufälligen
Verteilungswahrscheinlichkeit
Abstraktion des Player-Objekts um mehrere Spieler einbinden zu
können
Implementierung des Konfigurationsmenüs
Erstellung einer Hall of Fame die die Sieger nach einer Spielrunde
anzeigt
Entwicklung einer Shrinking-Funktion, welche das Spiel nach einer
vorgegebenen Zeit beendet

3. Dokumentationsphase (03.07. – 17.07.2012)

Erstellung der Präsentationen (Zwischen- und
Abschlusspräsentation)
Erstellung der Prüfungsstudienarbeiten der einzelnen
Teammitglieder
Erstellung der Gesamtdokumentation (inkl. Wiki-Eintrag)

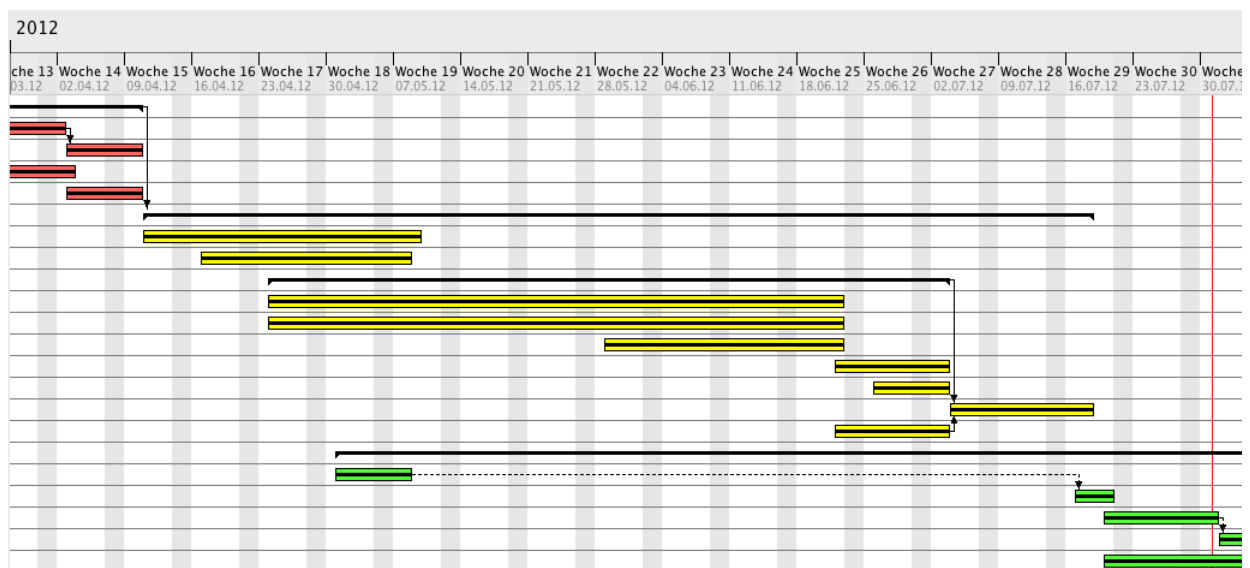


Abbildung 1 zeigt einen Auszug aus dem Gantt-Chart des Zeitplans des Projektes „ReMasterBlaster“. Dieser ist auch noch komplett als Anhang an diese Prüfungsstudienarbeit angehängt, weil hier die zugehörigen Aufgaben nicht angezeigt werden können, da es sonst überhaupt nicht mehr lesbar wäre. Was man jedoch sehen kann ist, dass sich der Zeitplan um 17 Tage nach hinten verschoben hat. Diese Tatsache kam dadurch zustande, da ich die Dokumentationsphase eigentlich komplett in den Prüfungszeitraum eingeplant habe und die anstehenden Prüfungen etwas unterschätzt habe. Zudem gab es Mitglieder im Team die noch Wiederholungsprüfungen aus vorangegangenen Semestern nachholen mussten. Was auch zu sehen ist, dass sich die Phasen teilweise überlappen. Beispielsweise fällt eine Aufgabe, nämlich die Vorbereitung und Erstellung der Zwischenpräsentation am 09. Mai 2012, mitten in die Entwicklungsphase. Dies ist aber nur der Tatsache geschuldet, dass wir die Zwischenpräsentation ebenfalls der Dokumentationsphase zugeordnet haben. Alles in allem bin ich vom zeitlichen Ablauf her sehr zufrieden mit dem Projekt. Die Verschiebung um ca. 2 Wochen ist auch aufgrund der mangelnden Erfahrung bzgl. einer Projektleitung entstanden. Jedoch habe ich in diesem Projekt auch sehr viele Erfahrungen in Sachen Zeit- und Projektplanung mitnehmen können, die ich bei einem nächsten Projekt auf jeden Fall mit einfließen lassen kann.

4.4 Facebook Gruppe

Es mag im Zusammenhang mit einer Projektorganisation vielleicht etwas profan klingen wenn man sagt, dass man eine Facebook Gruppe erstellt, jedoch erwies sich dies in unserem Fall als sehr nützlich. Ich wage zu behaupten, dass jeder der Projektteilnehmer mehrmals täglich auf Facebook online ist. Somit war dies eine gute Möglichkeit um kurzfristige Änderungen an alle Teammitglieder zu kommunizieren. Auch wurden hier Umfragen gestartet welcher Termin am Geeignetsten für ein nächstes Projekttreffen bzw. zum Schluss den Projekttag erscheint. Da man heutzutage Änderungen und Mitteilungen auch per Push-Nachricht auf

sein Smartphone erhält konnte man so gezielt und schnell die Projektgruppe erreichen.

4.5 Google Kalender

Der Google Kalender erschien mir ebenfalls als probates Mittel um die zeitliche Organisation der Teammitglieder zu optimieren. Hier verhält es sich ähnlich zu Punkt 4.4, dass jeder Teilnehmer der Projektgruppe einen Google Account hat. Diese Tatsache nahm ich zum Anlass einen gemeinsamen Kalender für das Projekt zu erstellen welchen ich dann für alle in der Gruppe freigab. So konnte jeder diesen „Projektkalender“ in seinen eigenen Kalender mit einfügen und hatte stets die anstehenden Termine im Blick. Für mich galt es dabei natürlich vorrangig den Kalender zu pflegen, aktuell zu halten und immer die richtigen (Raum-) Informationen einzutragen. Alles in allem war dies eine sehr einfache Möglichkeit die Termine zu koordinieren und im Blick zu halten.

5. Definition des Lasten- und Pflichtenhefts

Da zu einem Projekt normalerweise natürlich immer Auftraggeber und Auftragnehmer gehören musste ich als Projektleiter das Lasten- sowie das Pflichtenheft erstellen. Im Normalfall wird ein Lastenheft vom Auftraggeber spezifiziert und z.B. für eine Ausschreibung des gewünschten Projekts verwendet. Es beschreibt die Anforderungen die der Auftraggeber an das fertige Produkt stellt. Die Formulierungen hierbei sollten natürlich so allgemein wie möglich und so einschränkend wie nötig gewählt werden damit man während der Entwicklung immer noch Spielraum für etwaige Änderungen hat.

Im Pflichtenheft hingegen wird in konkreter Form beschrieben wie der Auftragnehmer gedenkt die an ihn gestellten Anforderungen zu lösen. Hier wurden von mir die von der Projektgruppe gesteckten Ziele niedergeschrieben. In den folgenden drei Abschnitten beschreibe ich die

selbst gesetzten Muss-, Soll- und Kann-Kriterien, da diese die wichtigsten Punkte aus dem Pflichtenheft darstellen.

5.1 Muss - Kriterien

Am Ende des Projekts muss ein spielbares Produkt von „ReMasterBlaster“ vorliegen wobei zwei Spielfiguren auf einem Spielfeld mittels einer Tastatur gesteuert werden können. Es müssen Bomben gelegt und Wände gesprengt werden können. Des Weiteren wird die Spielfigur eines Gegenspielers vom Spielfeld entfernt wenn diese innerhalb eines Bombenradius getroffen wird.

5.2 Soll - Kriterien

Es sollen weitere Spieldetails umgesetzt werden. Beim Wegsprengen einer Wand sollen verschiedene Goodies, welche aus dem Original bekannt sind, erscheinen und der Spielfigur, die das jeweilige Goodie einsammelt, zusätzliche Eigenschaften übertragen. Hierbei sollen folgende Goodies berücksichtigt werden:

- Additional Bomb (man erhält eine extra Bombe)
- Higher Range (der Bombenradius erhöht sich um ein Feld)
- Superman (man kann Mauernblöcke verschieben)
- Ghost (man ist nur 1x1px groß und kann durch Wände gehen)
- Controller (man kann die Bombe fernsteuern)
- Invincible (man ist für eine Aktion, die die Spielfigur des Spielers normalerweise vom Feld entfernen würde, geschützt → Spielfigur wird weiß)
- Remote Detonated Bomb (Bombenexplosion ist zeitgesteuert)
- Speed Up (man wird schneller → max. 4x möglich)
- Stopschild (alle anderen Spieler drehen sich auf der Stelle)
- Fragezeichen (hier verbirgt sich eines der folgenden Goodies: Protection, Ghost, Stopschild oder Krankheit → Timer für Bombenexplosion verkürzt sich)

- Death (die Spielfigur des Spielers wird entfernt)

Weiterhin sollen verschiedene Szenen definiert werden, die in einer vordefinierten Reihenfolge ablaufen. Am Anfang soll ein Startscreen mit dem Logo erscheinen, auf den das Konfigurationsmenü folgt. Die Konfigurationsmöglichkeiten sollen mittels eines JSON-Objekts eingebunden werden. Nachdem man die Konfiguration abgeschlossen hat soll man in die eigentliche Spielszene gelangen wo man dann das Spiel spielen kann.

5.3 Kann - Kriterien

Im Optimalfall, vor Allem aber falls am Ende der Muss- und Sollkriterien noch Zeit vorhanden sein sollte, kann überprüft werden ob mehr als 2 Spieler möglich sind. Des Weiteren kann eine Analyse stattfinden ob andere Eingabegeräte eingebunden werden können. Ebenfalls kann überprüft werden, ob das Spiel auch in Firefox, IE (v9 oder grösser) und Safari läuft bzw. lauffähig gemacht werden kann.

6. Versionsverwaltung mit github

Bereits ganz zu Beginn des Projekts empfahlen uns Prof. Dr. Matthias Hopf und Prof. Dr. Stefan Röttger ein Versionierungssystem für die Umsetzung des Projekts zu verwenden. Hierfür wurde uns dann github¹¹ von den beiden Projektbetreuern nahe gelegt. Github ist ein auf git basierender web-basierter Hostingdienst für Softwareentwicklungsprojekte. Git selbst stellt wiederum das Versionsverwaltungs-System dar. Auf github werden sehr viele bekannte Open-Source Projekte gehostet, darunter beispielsweise auch jQuery, PHP, Ruby on Rails, etc.

Da das Projekt ReMasterBlaster ebenfalls als Open-Source Projekt angelegt wurde bot sich github als Hosting für unseren Quellcode und andere Dokumente natürlich ebenfalls an. Am Anfang war die Verwendung

¹¹ <http://www.github.com>

von github, mit der sich dann natürlich jeder auseinander setzen sollte, erst einmal sehr schwierig. Ich selbst musste mich auch mit den Begrifflichkeiten „Repository“, „commit“ und „push“ erst vertraut machen. Das Repository stellt im Wesentlichen den eigentlichen Projektordner dar in dem alle Dateien gespeichert werden. Ein „Commit“ ist eine Beschreibung einer Änderung, die man an einer vorhandenen Datei vorgenommen hat. Mit einem „Push“ synchronisiert man den lokalen Ordner auf seiner Festplatte mit dem Online-Speicherplatz auf github. Es gab zwar einige Tutorials und Dokumentationen bezüglich github, jedoch dauerte es dennoch eine gewisse Zeit bis sich das Team in die Verwendung und den Workflow dieser Entwicklungsmethode eingefunden hatte.

7. Soll – Ist Analyse

Die Soll-Ist Analyse des Projekts ReMasterBlaster stellt sich wie folgt dar:

Muss – Kriterien	<ul style="list-style-type: none"> ▪ Spielbares Produkt von ReMasterBlaster ▪ Zwei steuerbare Spieler mittels Tastatureingabe ▪ Spieler muss Bomben legen und Wände sprengen können ▪ Gegenspieler wird bei einem Treffer von Spielfeld entfernt 	
Soll – Kriterien	<ul style="list-style-type: none"> ▪ Einbindung der Extras im Spiel (Goodies) ▪ Konfigurationsmenü zu Beginn des Spiels ▪ Szenenabfolge mit Credits, Config, Countdown, Spiel ▪ Hall of Fame der Gewinner einer Runde 	
Kann – Kriterien	<ul style="list-style-type: none"> ▪ Einbindung von mehreren spielbaren Charakteren ▪ zusätzliche Eingabegeräte verwendbar machen ▪ Shop um Extras nach einer Spielrunde kaufen zu können ▪ Sprite - Auswahl für die einzelnen Spielcharaktere 	
Kann – Kriterien	<ul style="list-style-type: none"> • ALLE Goodies wie im Originalspiel implementieren 	

In dieser Analyse zeigt sich, dass alle Muss- und Soll-Kriterien vollständig erfüllt werden konnten. Somit war der „Grunderfolg“ des Projekts bereits gesichert. Jedoch stellten wir bereits während der Umsetzung fest, dass wir noch „Reserven“ für zusätzliche Anforderungen bzw. so genannte „Kann-Kriterien“ hatten. Es wurden zwar bereits im Pflichtenheft solche festgelegt, jedoch kamen dann im weiteren Projektverlauf noch weitere „on the fly“ hinzu. Das waren natürlich immer die besten Momente bei einer Teambesprechung, wo die Ideen gerade so aus den Teilnehmern „raus sprudelten“. Hier konnte ich feststellen, dass das Team begeistert bei der Sache ist auch persönliches Engagement und Hingabe für die „gemeinsame Sache“ zeigte. Von diesen selbstgesteckten Kann-Zielen konnten wir dann auch den größten Teil gemeinsam bewältigen. Einzig und allein die Implementierung von wirklich allen Goodies, wie sie auch im Originalspiel vorhanden sind, blieb uns mangels vorhandener Zeit verwehrt. Doch ich bin mir sicher, dass auch die restlichen Goodies mit ausreichend vorhandener Zeit definitiv möglich gewesen wären was die technische und organisatorische Umsetzung betrifft. Somit bin ich mit der Soll – Ist Analyse als Projektleiter des Projekts „ReMasterBlaster“ mehr als zufrieden, da wir anfangs noch davon ausgegangen sind nur einen kleinen Teil dieser Ziele wirklich erreichen zu können.

8. Resümee

Das Resümee soll versuchen das komplette Projekt in all seinen Bereichen und besonders meine eigenen Erfahrungen bezüglich des Projekts und den anderen Projektmitgliedern zu reflektieren.

Da sich die Projektfindung am Anfang des Sommersemesters 2012 als sehr schwierig gestaltete und ich an dem Tag der eigentlichen Projektwahl wegen eines wichtigen und schon lange im Voraus terminierten Arzttermins nicht anwesend sein konnte, informierte ich mich im Nachhinein dieser Veranstaltung bei meinen Kommilitonen über die angebotenen Themen und die daraus resultierenden Projektgruppen. Es

galt eine Entscheidung zu treffen zwischen der Projektgruppe „Matrix“ und der Gruppe „MasterBlaster Remake“.

Da ich mich persönlich mehr für die Webentwicklung interessiere und die Projektidee von Prof. Dr. Matthias Hopf sogar auf meine Nachfrage nach einem möglichen Projektthema hin entstanden ist entschied ich mich schlussendlich für den MasterBlaster Remake. Die Gruppe bestand zu dem Zeitpunkt bereits aus Sergej Bjakow, Michael Kao, Pavlina Pavlova und Maximilian Seyfert. Nach einem ersten gemeinsamen Treffen legten wir uns gemeinsam darauf fest, dass ich den Part des Projektleiters übernehmen werde. Da von den anderen Teilnehmern niemand gezielt diese Position einnehmen wollte stellte ich mich zur Verfügung, weil es mich immer schon einmal interessierte in so einer Rolle zu sein. Im Nachhinein muss ich für mich selbst feststellen, dass es eine sehr gute Wahl war, da mir die Arbeit als Projektleiter mit der Organisation und Kommunikation innerhalb der Gruppe und auch mit den betreuenden Professoren sehr viel Spaß gemacht hat. Leider hatte ich zu Beginn sehr viel Aufwand mit organisatorischen Dingen (Projektplan, Lastenheft) und konnte mich bei den Programmierarbeiten nicht wirklich produktiv beteiligen. Ich habe immer versucht in allen Bereichen den Durchblick zu wahren und hier muss ich besonders Michael Kao hervorheben, der sich des Öfteren die Zeit genommen hat „Code Reviews“ mit mir durchzuführen und mir aufgezeigt hat was wo in dem Code passiert und wo momentan die Probleme liegen. Falls es Probleme gab bezog mich Michael Kao auch immer mit in die Programmierarbeiten mit ein und wir redeten viel über die dahinter steckende Logik. Nachdem wir zu einer Lösung gekommen waren setzte er diese Lösung programmtechnisch um. Ich hoffe und denke, dass ich ihm auch bei dem ein- oder anderen Problem eine Hilfe gewesen bin. Auch mit Pavlina Pavlova war es stets eine angenehme Arbeitsatmosphäre, weswegen ich hier betonen möchte, dass die beiden, Michael und Pavlina, in meinen Augen die Stützpfeiler des Projektes „ReMasterBlaster“ gewesen sind. Bei zwei von drei angesetzten

„Projekttagen“ in der heißen Schlussphase waren wir nur zu dritt anwesend. Natürlich muss man auch berücksichtigen, dass die beiden anderen Projektteilnehmer noch Wiederholungsprüfungen abzuleisten hatten, welche nicht vernachlässigt werden sollten. Jedoch trug Maximilian Seyfert auch außerhalb der Prüfungszeit nur sehr wenig zum Projekt bei. So wurden ihm Aufgaben mit einem angesetzten Fertigstellungsdatum aufgetragen, die schlichtweg nicht oder nicht in ausreichendem Maß erfüllt wurden, sodass oft Nacharbeit der anderen Teammitglieder von Nöten war. Dies ist der einzige kritische Punkt, den ich hier bezüglich der Projektgruppe anmerken möchte.

Alles in Allem hat mir das Projekt sehr viel Freude bereitet und auch meine Aufgabe als Projektleiter hat mir sehr viel Spaß gemacht. Ich konnte viele neue Dinge bezüglich einer gelungenen Projektorganisation erlernen und habe neue Kommunikationswege kennen gelernt. Auch der Zeitplanungsfehler, die Dokumentationsphase mitten in die Prüfungsvorbereitung zu legen, wird mir in dieser Form mit Sicherheit nicht mehr unterlaufen.

Abschließend möchte ich mich bei Prof. Dr. Hopf und Prof. Dr. Röttger für die immer freundliche und hilfreiche Projektbetreuung sowie bei meinem gesamten Projektteam für die lustige und aufschlussreiche Zeit, sowie die absolut gelungene Abschlusspräsentation bedanken!

Abbildungs- und Quellenverzeichnis

Abb. 1: Auszug aus dem Gantt-Chart der Zeitplanung
des Projekt, Quelle: GANTTproject

Seite 12

Anhang

Gesamter Zeitplan als Gantt-Chart:

