

LASTENHEFT

Artillery



Fachhochschule Nordwestschweiz
Hochschule für Technik

3. Semester Systemtechnik / 3Sbb

Fachhochschule Nordwestschweiz

Hochschule für Technik

Dozenten:

Prof. Dr. Norbert Hofmann

Dr. Patrick Coray

Studenten:

Joel Koch

Raphael Waltenspül

Final / V1.00.00 / 28.10.2015

Abstract

Dieses Dokument, beschreibt die Gesamtheit der Forderungen bezüglich Lieferung und Leistungen der Software welche im Matlab Workshop entwickelt wird. Konkret sind nachfolgend die Konzepte und Rahmenbedingungen, der Einsatz und die Funktionen der Software Artillery beschrieben. Die Funktionen sind nach Must, Need und Nice to Have gegliedert. Die Must Funktionen beinhaltet die grundlegenden Elemente um das Spiel laufen zu lassen, wie zum Beispiel den Winkel und Schussstärke einzustellen. In den Need und Nice to Have Funktionen sind den zusätzlichen Gadgets beschrieben, welche das Spiel interessanter machen. Dies können zum Beispiel verschiedene Spielmodi oder Multiplayer Funktionen sein. Das Lastenheft liefert eine detaillierte Liste der Must und Need Funktionen und dient hiermit auch zum messen des Erfüllungsgrad der Softwareanforderungen.

1 Inhaltsverzeichnis

1	Inhaltsverzeichnis.....	3
2	Einleitung	5
2.1	<i>Änderungen / Version.....</i>	<i>5</i>
2.2	<i>Abkürzungen</i>	<i>5</i>
2.3	<i>Gültigkeit und Zweck.....</i>	<i>6</i>
2.4	<i>Ablage, Verweise.....</i>	<i>6</i>
2.5	<i>Mitgeltende Unterlagen.....</i>	<i>6</i>
2.6	<i>Verteiler und Freigabe.....</i>	<i>6</i>
3	Ausgangslage.....	7
3.1	<i>Ist-Konzept.....</i>	<i>7</i>
3.2	<i>Projektidee</i>	<i>7</i>
3.3	<i>Langfristige Strategie</i>	<i>7</i>
4	Konzept und Rahmenbedingungen.....	8
4.1	<i>Zielsetzung.....</i>	<i>8</i>
4.2	<i>Must Kriterien.....</i>	<i>8</i>
4.2.1	<i>Systemelemente</i>	<i>8</i>
4.2.2	<i>Spielelemente</i>	<i>8</i>
4.3	<i>Need Kriterien.....</i>	<i>9</i>
4.3.1	<i>Spielelemente.....</i>	<i>9</i>
4.4	<i>Nice to Have Kriterien.....</i>	<i>9</i>
4.4.1	<i>Systemelemente</i>	<i>9</i>
4.4.2	<i>Spielelemente</i>	<i>9</i>
4.5	<i>Bewertung der Zielerreichung.....</i>	<i>10</i>

5	Produkteinsatz	11
5.1	<i>Benutzer / Zielgruppe.....</i>	<i>11</i>
6	Produktumgebung.....	12
6.1	<i>Systemvoraussetzung</i>	<i>12</i>
6.1.1	Windows.....	12
6.1.2	Mac OS.....	12
6.1.3	Linux	12
7	Anforderungen	13
7.1	<i>Funktionale Anforderungen.....</i>	<i>13</i>
7.1.1	Need Funktionen	14
7.2	<i>Daten Schnittstellen</i>	<i>15</i>
7.2.1	Need Daten Schnittstellen	15
7.3	<i>Benutzer Schnittstellen</i>	<i>15</i>
7.4	<i>Leistungsmerkmale</i>	<i>15</i>
7.5	<i>Testfälle</i>	<i>15</i>
7.6	<i>Nichtfunktionale Anforderungen Benutzbarkeit.....</i>	<i>16</i>
7.6.1	Design.....	16
7.6.2	Zuverlässigkeit	16
7.6.3	Änderbarkeit; Übertragbarkeit; Wartbarkeit	16
8	Lieferumfang	16
9	Abnahmekriterien	17
10	Unterschriften	17

2 Einleitung

2.1 Änderungen / Version

Task:	Wer:	Wann:	Bemerkung:
Erstellt V00.00.01	WR	27.10.2015	
Review V00.00.02	JK	28.10.2015	Ok
Anpassungen V01.00.00	WR	28.10.2015	Die Kategorien wurden um das „Need“ erweitert

2.2 Abkürzungen

Abkürzung:	Bedeutung:
JK	Joel Koch
RW	Raphael Waltenspül
Fxy	Produktfunktion Nummer xy
Dxy	Datenschnittstelle Nummer xy
Lxy	Leistungsmerkmal Nummer xy
Bxy	Benutzerschnittsteller Nummer xy
Exy	Testcase Nummer xy
Spielrunde	Entspricht dem Spielverlauf bis zu einem Treffer
Spielzug	Entspricht einem Feuerbefehl
Software	Software bezeichnet die geschriebenen .m Files zur Ausführung in Matlab.

2.3 Gültigkeit und Zweck

Das Dokument ist für die Dauer des Projektes, das heisst für die Dauer des Matlab Workshops gültig. Mit der Entlastung des Projektteams erlischt der Anspruch auf weiter Implementierungen gemäss Lastenheft. Zweck dieses Dokumentes ist es eine exakte Beschreibung der Erwartungen an die Software „Artillerie“ zu liefern und den Programmierern den Auftrag über die gewünschten Funktionen zu formulieren. Das Lastenheft dient weiter zur Bewertung der Zielerreichung und gilt als Grundlage zur Entlastung des Projektteams.

2.4 Ablage, Verweise

Die Ablage des gesamten Projektes befindet sich in einem Git- Repository. Berechtigung für das Repository wird allen beteiligten erteilt. Das Projekt unterliegt der „GNU General Public License / Version 3, 29 June 2007“

HTTPS clone URL: <https://github.com/fhnwRaphi/fhnwMatlProject>

2.5 Mitgeltende Unterlagen

Bezeichnung	Autor / Datum	Bemerkung
Skript mds, 2 Modellierungstechniken, S. 8	T. Petermann	Beispiel Schiefer Wurf mit Luftwiderstand
Skript mech, Dynamik des Massenpunktes, S.9	R. Lamprecht	Beispiel Schiefer Wurf
Skript matl,	N. Hofmann P. Coray	gesamtes Skript aus der Vorlesung

2.6 Verteiler und Freigabe

Der Verteiler des Lastenhefts ist wie folgt:

- Ablage im Git- Repository
- `norbert.hofmann@fhnw.ch`
- `patrick.coray@fhnw.ch`
- `joel.koch@students.fhnw.ch`
- `raphael.waltenspuel@students.fhnw.ch`

Die Freigabe des letzten Heftes erfolgt durch mindestens einen der folgenden Dozenten:

- Prof. Dr. Norbert Hofmann
- Dr. Patrick Coray

3 Ausgangslage

Ausgangslage ist der Auftrag, im Zuge des Moduls Matlab Workshop eine Software zu entwickeln.

3.1 Ist-Konzept

Zum jetzigen Zeitpunkt besteht keine Software. Die Software wird neu geschrieben.

3.2 Projektidee

Die Idee des Projektes Artillery entstand dadurch, dass nach einer Software gesucht wurde welche in der Komplexität und Menge der Funktionen stark skalierbar ist, die uns einen Bezug zu anderen Modulen gibt (wie Mechanik oder modellieren dynamischer Systeme) und zu guter Letzt auch Spass macht.

3.3 Langfristige Strategie

Die Langfristige Strategie besteht darin, Wissen bezüglich dem Einsatz von Matlab zu erwerben und dieses in weiteren Modulen einzusetzen. Langfristig soll dies den Aufwand pro ECTS verringern.

4 Konzept und Rahmenbedingungen

4.1 Zielsetzung

Es wird ein Artilleriesimulator entwickelt. In diesem Simulator müssen die Spieler über die Eingabe von Parametern versuchen, sich gegenseitig zu treffen. Dabei findet das Geschehen in einer zufällig generierten Umgebung statt, in welcher auch Parameter wie Wind veränderlich sind. Die Flugparabel wird unter Berücksichtigung des Luftwiderstands berechnet und gezeichnet. Ziel ist es, eine stabil funktionierende Anwendung in Matlab zu schreiben, welche mindestens alle Musskriterien erfüllt.

4.2 Must Kriterien

4.2.1 *Systemelemente*

- Zwei Spieler können gegeneinander spielen.
- Die Spieler spielen rundenbasiert an einem Rechner.
- Das Spiel ist 2-dimensional.

4.2.2 *Spielelemente*

- Die Landschaft wird pro Spielrunde neu berechnet.
- Die Landschaft wird zufällig generiert.
- Der Wind weht und ändert in Abhängigkeit der Zeit die Stärke.
- Die Spieler werden zufällig platziert.
- Jeder Spieler hat einen Tank.
- Der Spieler kann die Parameter der Kanone ändern. Diese sind:
 - Stärke der Treibladung (Skalare Grösse)
 - Winkel des Kanonenrohrs (Winkel in Grad)
- oder der Spieler kann diese über eine Mauseingabe per Geschicklichkeit steuern.
- Der Spieler gibt den Feuerbefehl

- Das Spiel zeichnet die exakte Flugparabel unter Berücksichtigung des Luftwiderstandes. Dies in einer 2 dimensional Ebene
- Trifft das Geschoss die Umgebung, so bildet sich dort einen Krater.
- Trifft das Geschoss den Gegenspieler so wird dieser zerstört. Das Spiel ist zu ende.

4.3 Need Kriterien

4.3.1 Spielelemente

- Das Spiel geht über mehrere Runden.
- Trifft der Spieler erhält einen Punkt und eine neue Spielrunde beginnt.
- Der Spieler der hinten liegt fängt an. In der ersten Runde wird gelost.
- Die Kanonen haben eine Toleranz. Die Schuss- Parabel unterliegt einer Streuung

4.4 Nice to Have Kriterien

4.4.1 Systemelemente

- Das Spiel kann über das Netzwerk / Internet gespielt werden
- Es können mehr als zwei Spieler Teilnehmen.
- Das Spiel kann sowohl Rundenbasiert nacheinander, als auch Rundenbasiert gleichzeitig gespielt werden. (Gleichzeitig heisst, wenn alle Spieler ihren Zug eingegeben haben wird dieser ausgeführt.
- Das Spiel bietet Soundeffekte
- Das Spiel ist 3-dimensional

4.4.2 Spielelemente

- Der Spieler kann verschiedene Geschosse wählen. Diese können sein:
 - A. Grösser Sprengkraft, dafür Schwerer.
 - B. Streumunition
 - C. ...

- Die Kanone kann sich im Verlauf des Spiels erhitzen, dies erhöht die Ungenauigkeit und kann bis zum Ausfall der Kanone führen.
- Es können sich Schockeffekte bilden. Trifft ein Geschoss in der unmittelbaren Umgebung so muss der Gegenspieler einen Spielzug aussetzen.
- Die Spieler können mehrere Tanks haben.
- Die Tanks haben individuelle Beschädigungen
- Das Spiel bietet weitere Aktionen wie
 - D. Fahren
 - E. Reparieren
- Weitere Gadgets sind immer möglich...

4.5 Bewertung der Zielerreichung

Die Bewertung der Zielerreichung erfolgt durch die Dozenten:

- Prof. Dr. Norbert Hofmann
- Dr. Patrick Coray

Als Grundlage zur Bewertung gilt die Liste der Funktionen und der Umsetzung dieser.

5 Produkteinsatz

Das Programm ist zu Lehrzwecken im Zuge des Matlab Workshops gedacht. Der stetig steigende Schwierigkeitsgrad im Aufbau der Funktionen soll uns erlauben unsere Fähigkeiten kontinuierlich zu steigern, anzuwenden und frühe Erfolge zu erzielen. Weiter soll es stets ein Ansporn sein möglichst viele schwierige Nice to Have Funktionen zu implementieren.

Das fertige Produkt soll dann als Lückenfüller in nicht allzu anspruchsvollen Vorlesungen¹ dienen.

5.1 Benutzer / Zielgruppe

Die Benutzer sind primär die Dozenten welche die Software testen und wir, die Studenten. Weiter soll das Programm dem Studiengang 3Sbb zur Verfügung gestellt werden.

¹ Steht in keinem Bezug zu dem Modul matl.

6 Produktumgebung

6.1 Systemvoraussetzung

6.1.1 Windows

Operating Systems	Processors	Disk Space	RAM	Graphics
Windows 10	Any Intel or AMD x86 processor supporting SSE2 instruction set*	1 GB for MATLAB only, 3–4 GB for a typical installation	2 GB; With Simulink Coder, 4 GB is recommended	No specific graphics card is required. Hardware accelerated graphics card supporting OpenGL 3.3 with 1GB GPU memory recommended.
Windows 8.1				
Windows 8				
Windows 7 Service Pack 1				
Windows Server 2012				
Windows Server 2008 R2 Service Pack 1				
Windows Server 2008 Service Pack 2				

6.1.2 Mac OS

Mac Os Systeme ohne Gewähr

6.1.3 Linux

Linux Systeme ohne Gewähr

7 Anforderungen

Die Nice to Have Elemente werden nicht beschrieben. Diese werden im Falle des Implementierens der Listen als Ergänzung hinzugefügt.

7.1 Funktionale Anforderungen

Nachfolgend sind die funktionalen Anforderungen der Must Elemente beschrieben.

/F10/ Der User startet das Programm in Matlab über den Befehl `>> ARTILLERY`

/F20/ Das Programm öffnet das Main Menue.

/F30/ Im Main Menue können die grundlegenden Parameter eingestellt werden.

Die grundlegenden Parameter sind: Spielmodi (Geschicklichkeit; Taktik); Windstärke (low, medium, high); Einstellung der Berghöhe. (low, medium, high);

/F40/ Starten des Spiels über einen Button.

/F50/ Eine zweidimensionale Landschaft wird zufällig generiert. Dabei wird die gewünschte Berghöhe berücksichtigt.

/F60/ Der Wind weht mit einer Stärke. Der Wind wird im Diagramm als Zahl und Richtungsvektor angezeigt. Der die Windstärke hängt von den eingestellten Parametern Windstärke ab.

/F70/ Die generierte Landschaft entspricht in der x Achse zwischen 500m und 2000m

/F80/ Der Panzer jedes Spielers wird zufällig platziert. Dies aber auf der Landschaftsoberfläche und auf einem flachen Plateau.

/F90/ Im Spielmodi Geschicklichkeit, erscheint ein Powerbalken in der Mitte des Feldes.

/F100/Im Spielmodi Taktik, erscheint ein Eingabefeld für Treibladungsstärke und Winkel.

/F110/Im Spielmodi Geschicklichkeit, kann der Spieler welcher am Zug ist, die Parameter der Kanone ändern. Für den Winkel geschieht dies über die Lage des Mauszeigers in Bezug zum Standort der Kanone. Für die Stärke geschieht dies über die Zeit welche der Spieler die Taste drückt.

/F120/Im Spielmodi Taktik, kann der Spieler welcher am Zug ist, die Parameter der Kanone ändern. Dies geschieht über die Eingabe der Werte für Treibladung in Joule und Winkel in Grad in einem Eingabefeld. Den Feuerbefehl gibt der Spieler durch drücken einer Taste.

/F130/Das Spiel zeichnet nun ausgehend von dem schießenden Spieler die exakte Flugparabel des Projektils unter Berücksichtigung des Luftwiderstandes. Es soll dafür eine möglichst exakte Flugbahn gezeichnet werden. Es werden die Methoden aus den Modulen Mechanik und Modellieren dynamischer Systeme verwendet

/F140/Trifft das Geschoss die Umgebung, so bildet sich dort ein Krater. Dies geschieht durch das ändern des Landschaftsvektor an dieser Koordinate.

/F150/Trifft das Geschoss den Gegenspieler so wird dieser zerstört, der Spieler welcher trifft, erhält einen Punkt und eine neue Runde beginnt.

7.1.1 Need Funktionen

Nachfolgend sind die funktionalen Anforderungen der Need Elemente beschrieben.

/F160/Ergänzend zu Punkt /F30/ Auswahl / Einstellen der Parameter: Spielmodi (Geschicklichkeit; Taktik); Windstärke (low, medium, high), Windböen (low, medium, high); Anzahl Runden (0 bis 100 St.); Einstellung der Berghöhe. (low, medium, high);

/F170/Ergänzend zu Punkt /F60/ Der Wind weht und ändert in Abhängigkeit der Zeit die Stärke. Der Wind wird im Diagramm als Zahl und Richtungsvektor angezeigt. Die Windstärke hängt von den eingestellten Parametern Windstärke ab. Die Änderung der Windstärke während einer Runde hängt von den eingestellten Parametern Windböen ab

/F180/In den Ecken des Spielfeldes werden die Punktestände angezeigt.

/F190/Befindet sich das Spiel in der ersten Runde, oder herrscht Gleichstand so ermittelt das Programm zufällig welcher Spieler startet.

/F200/Herrscht kein Gleichstand, so starten der Spieler der zurückliegt.

7.2 Daten Schnittstellen

Nachfolgend sind die Anforderungen und das Datenmanagement der Must Elemente beschrieben.

/D10/ Das Spiel hat feste Grundeinstellungen.

/D20/ Die Grundeinstellung der Daten sind in jeweiligen Klassen beschrieben. Dies soll das Handling mit grossen Mengen an Spielparametern erleichtern. Zum einen können diese dann einfach als Objekt übergeben werden zum andern können die Parameter einfach über Methoden eingestellt werden.

7.2.1 Need Daten Schnittstellen

Nachfolgend sind die Anforderungen und das Datenmanagement der Need Elemente beschrieben.

/D30/ Die Spielzüge werden als Datei gespeichert. (Z. Bsp. .txt;.csv;.xml)

7.3 Benutzer Schnittstellen

/B10/ Das Spiel wird in einer GUI Dargestellt

/B20/ Die Eingabe kann über die Maus erfolgen

/B30/ Die Eingabe kann über die Tastatur erfolgen.

7.4 Leistungsmerkmale

/L10/ Das Spiel muss flüssig laufen. Das bedeutet die Kometenflugbahn darf nicht stocken. Pausen in welchen nichts passiert und keine Eingabe möglich ist sind zu vermeiden.

/L20/ Die Berechnung der Parabel soll $< 3s$. dauern.

7.5 Testfälle

/E10/ Aufstarttest des Programms

/E20/ Eingabe aller Parameter aus /F30/. (Ohne gegenseitige Abhängigkeiten)

/E30/ Abbrechen nach Eingabe Parameter.

/E40/ Fehlhandlung. Neben Buttons klicken, neben Spielfeld klicken.
/E50/ Eingabe unrealistischer Parameter.
/E60/ Parabel mathematisch nachprüfen. Weite, maximale Höhe.
/E70/ Eingabe während Schussanimation.
/E80/ Punktezähler. Zählt dieser korrekt hoch.
/E90/ Abbruch / Neustart des Programmes
/E100/Nächstes Level starten.
/E110/Win/Loss Bedingungen testen.

7.6 Nichtfunktionale Anforderungen Benutzbarkeit

7.6.1 *Design*

Das Programm soll möglichst ansprechend gestaltet sein. Dabei ist speziell auf einem Benutzerfreundliche resp. selbsterklärende Interface zu achten. Die Grafik des Spielfeldes muss erkennen lassen was mit den einzelnen Elementen wie beispielsweise einem Panzer gemeint ist.

7.6.2 *Zuverlässigkeit*

Das Programm soll stabil Laufen. Abstürze dürfen durch normales Handeln nicht verursacht werden.

7.6.3 *Änderbarkeit; Übertragbarkeit; Wartbarkeit*

Die Software muss gut dokumentiert sein. Die Software muss jederzeit durch dritte gelesen und erweitert werden können.

Die Mathematischen Berechnungen müssen in der Dokumentation nachvollzogen werden können. (Bsp. Formeln in Latex)

Zur Versionierung soll ein git repository verwendet werden.

8 Lieferumfang

Der Lieferumfang beinhaltet die Software inklusiver der Dokumentation.

9 Abnahmekriterien

Die Abnahme erfolgt durch die Dozenten. Die Software inklusive der Dokumentation ist bis spätestens 14. Januar 2016 abzugeben.

Die Entlastung des Projektteams erfolgt durch eine genügende Benotung.

10 Unterschriften

Student

Herr Joel Koch

.....

Ort, Datum

Unterschrift

Student

Herr Raphael Waltenspül

.....

Ort, Datum

Unterschrift