Lastenheft Matlab

Beispiel Projektname (z.B. Schiffe Versenken, XY-Berechnung, …)

Diese Textfelder werden zum Kommentieren verwendet (für Hinweise etc).  
In einem „richtigen“ Lastenheft kommen die Textfelder natürlich nicht vor.

In diesem Beispiel werden Elemente von realen Lastenheften vergangener Matlab-Projektarbeiten verwendet. Aufgrund der starken Vermischung kann nicht jedes Wort referenziert werden. Es sei den Autoren jedoch ausdrücklich gedankt.

**Teammitglieder:** Kermit the Frog

Lisa Simpson

**Dozent:** Professor Balduin Bienlein

**Fach** Matlab

**Studiengang:** Studiengang XY-Technik

Fachhochschule Nordwestschweiz, Hochschule für Technik

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Version** | **Autor** | **Datum** | **Status** | **Kommentar** |
| 0.9 | Frog / Simpson | 03.04.2015 | provisorisch |  |
| 1.0 | Frog / Simpson | 23.04.2015 | definitiv | von Prof Bienlein genehmigt |

**Inhaltsverzeichnis**

Bitte die Nummerierung der Kapitel beibehalten!

(dies ist u.a. für das Überprüfen Ihres Lastenheftes nötig)

[1 Zielbestimmung 3](#_Toc408915591)

[1.1 Muss 3](#_Toc408915592)

[1.2 Wunsch 3](#_Toc408915593)

[1.3 Abgrenzungskriterien 3](#_Toc408915594)

[2 Produkteinsatz 3](#_Toc408915595)

[2.1 Anwendungsbereich 3](#_Toc408915596)

[2.2 Zielgruppe 3](#_Toc408915597)

[3 Produkteumgebung 4](#_Toc408915598)

[3.1 Software 4](#_Toc408915599)

[3.2 Betriebssystem 4](#_Toc408915600)

[3.3 Hardware 4](#_Toc408915601)

[4 Produktfunktionen 4](#_Toc408915602)

[5 Produktdaten 5](#_Toc408915603)

[6 Produkt-Leistungen 5](#_Toc408915604)

[7 Benutzungsschnittstelle 5](#_Toc408915605)

[8 Qualitätsbestimmung 6](#_Toc408915606)

[9 Testfälle 6](#_Toc408915607)

[10 Anhang 6](#_Toc408915608)

# Zielbestimmung

An dieser Stelle steht ein Beschrieb worum es geht (z.B. Spiel xy, Berechnung für eine Semesterarbeit, …). Ferner wird umschrieben was in Matlab getan werden soll

Auch „Das Programm ist zur Anwendung im Rahmen des Matlab Workshops im XY-Semester 2033 gedacht“ ist gültig.

## Muss

1. Einlesen von Daten
2. Berechnung XY
3. Grafische Benutzer Oberfläche für die Bedienung

Folgende Soll-Schwerpunkte ‘müssen’\* vorkommen!

● Dateien lesen und schreiben (ASCII oder Excel)

● Daten manipulieren und darstellen

● Benutzereingaben (optional GUI)

\*Falls sich etwas nicht integrieren lässt bitte mit den Dozenten absprechen.

1. Parameter 1 einstellbar
2. Parameter 2 einstellbar
3. Spiel Spieler gegen Spieler
4. 2D Plot von …
5. Abspeichern von Erstellten
6. Abspeichern von Daten im ASCII und Excel Format

## Wunsch

1. 3D Plot von …
2. Bibliothek
3. Spiel gegen Computer
4. Eingaben speicherbar

## Abgrenzungskriterien

Das Programm wird nur für eine bestimmte Vereinfachung geschrieben. Gespielt wird ohne Hintergrundmusik. Es können maximal drei Bauelemente ausgewählt und kombiniert werden.

# Produkteinsatz

Auch „Das Programm ist zur Anwendung im Rahmen des Matlab Workshops im XY-Semester 2033 gedacht“ ist gültig.

Produkteinsatz = Wo wird das Produkt (d.h. Ihre Software) eingesetzt?

## Anwendungsbereich

Das Programm soll im Rahmen des Projekts XY des Instituts für Z zur Anwendung kommen. Das integrierte Spiel ist zur Auflockerung während der Arbeit gedacht. Weiter soll es helfen die gemessenen Daten besser zu verstehen

Zielgruppe: Im einfachsten Fall sind dies die Dozenten, sowie die Programmierer selbst.

## Zielgruppe

Die primäre Zielgruppe ist Herr B vom Projekt XY und die Firma Q bei denen das Programm zur Anwendung kommt. Des Weiteren ist der Spielmodus zur Überbrückung von Wartezeit gedacht.

# Produkteumgebung

Umgebung = Was für Hardware, Treiber, Bibliotheken, … benötigt Ihre Software um zu funktionieren?

## Software

Matlab Version R2014b

Mit den Modulen „Image Processing Toolbox 9.1“ und „Statistics Toolbox 9.1“

## Betriebssystem

Windows 7 und nachfolgende

Ohne Gewähr: OS X, Linux

## Hardware

* PC / Laptop mit den Mindestanforderungen von Matlab Version R2014b
* Bildschirm mit einer Auflösung von mindestens 1366 × 768.
* Maus mit zwei Knöpfen (rechte Maustaste).

# Produktfunktionen

Funktion = Wie verhält sich ihr Programm? Was kann ein User tun?

/F10/ Konfiguration des Programms mit den Parametern Länge, Breite, Höhe, …

/F20/ Beim Start des Programms werden zunächst Standardwerte verwendet („Grundeinstellungen“).

/F25/ Wird ein ungültiger Wert eingegeben, erscheint eine Fehlermeldung.  
 Wichtig: /F25/ wurde nachträglich, d.h. nach der ersten Abgabe, dazwischengefügt.  
 Nummerieren Sie zuerst immer in 10er Schritten.

/F30/ Berechnet wird die Temperatur als Funktion der Zeit.

/F40/ Das Spiel kann gestartet, pausiert und abgebrochen werden.

/F50/ Die Berechnung kann wahlweise mit Algorithmus a oder Algorithmus b erfolgen.

Der Grund für die Gliederung in /F10/, /F20/, … /D10/, /D20/, … Punkte, ist dass Sie und wir während Ihrer Arbeit den Stand überblicken können und nach Abgabe evaluieren können was erreicht worden ist und was nicht. Da es sich um ein Lastenheft handelt ist ein nicht erreichen eines Punktes idR. nicht besonders schlimm – d.h. Sie müssen dann nicht zwingend alles erfüllen, solange sie begründen können warum (idR. läuft nicht immer alles wie erwartet).

Würden die Produktfunktionen, Produktdaten, … nur umschreiben und nicht in nummerierte Punkte gliedert werden, so ergäbe sich tendenziell eine “neblige” und nur schwer durchschaubare Spezifikation der zu erledigenden und auszuführenden Arbeit.

Nummerieren Sie zunächst in zehner Schritten: /F10/, /F20/, /F30/, …

Wenn nach der ersten Freigabe des Lastenhefts ein Punkt hinzukommt, z.B. zwischen /F10/ und /F20/, dann kann dieser mit /F15/ hinzugefügt werden.

Fällt nachträglich ein Punkt weg, so streichen Sie diesen durch (aber nicht löschen). Z.B.

~~/F50/ Hier stand was veraltetes~~

# Produktdaten

Daten = Was für Daten erhalten Sie? → Textdateien? Bilder? Plots? …

/D10/ Ausgegeben wird ein 2D Plot der Temperatur als Funktion der Zeit.

/D20/ Die numerischen Resultate werden als csv ASCII Datei abgespeichert.

/D30/ Plot Visualisierungen können als png-Datei abgespeichert werden

Wie Leistungsfähig ist ihr Produkt – Denken Sie z.B. an „von 0 auf 100 in 3 Sekunden; 300 PS; maximal 280 km/h; … ; Ferner: Einschränkungen, d.h. was es nicht leistet.

# Produkt-Leistungen

/L10/ Die Berechnung darf nicht länger als drei Sekunden benötigen.

/L20/ Das Spiel soll flüssig laufen.

/L30/ Das Programm löst die vollständigen Maxwell Gleichungen und nicht nur eine Approximation.

/L40/ Die Genauigkeit beträgt drei signifikante Ziffern.

/L50/ Berechnungen die länger als 10 Sekunden benötigen werden abgebrochen und lassen sich deshalb nicht berechnen.

/L60/ {Wunsch}: Während der Berechnung wird der Fortschritt angezeigt.

/L70/ {Wunsch}: Das GUI ist mit speziellen Ornamenten verziert.

# Benutzungsschnittstelle

Spezialfall: Wenn ihre Software nicht für die Interaktion mit Menschen gedacht ist, sondern z.B. eine Ausgabe für eine andere Software liefert.

Schnittstelle = Wie kommuniziert die Software mit dem User?

Z.B. über einen Plot, Kommandofenster, …

/B10/ Das Spiel wird in einem 2D Plot dargestellt

/B20/ Die 2D Plots werden in XY-Darstellung dargestellt.

/B30/ Hinweise und Fehlermeldungen werden im Command Fenster ausgegeben

/B40/ Die Spielfeldgrösse und weiteren Optionen werden über ein GUI eingegeben.

/B50/ Benutzer-Eingaben erfolgen über die Tastatur im Command Window.

/B60/ Die Steine im GUI können mit der Maus bewegt werden.

/B70/ {Wunsch} Die versteckten Optionen können in einer Textdatei manipuliert werden.

/B80/ {Wunsch}: Hinweise und Fehlermeldungen werden in einer Log-Datei gespeichert.

# Qualitätsbestimmung

Die Zielgruppe („Kunde“) der Software sind Laborratten welche Ihre Messungen möglichst schnell und einfach auswerten möchten ohne viel Zeit mit der Software zu verlieren. Die Wartung der Software erfolgt durch die Entwickler (Kermit und Lisa) selbst. Ein gewisser Wert wird auf die Korrektheit der Berechnung gelegt, wobei die Laborratten grobe Fehler aufgrund Ihrer Erfahrung sehr schnell erkennen würden.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Qualitätskriterium** | **sehr gut** | **gut** | **normal** | **nicht relevant** |
| Änderbarkeit (des Codes) |  |  | X |  |
| Überprüfbarkeit (des Codes) |  | X |  |  |
| Verständlichkeit (des Codes) |  | X |  |  |
| Wartbarkeit (des Codes) |  |  | X |  |
| Benutzungsfreundlichkeit (Programm) | X |  |  |  |
| Effizienz (Programm & Code) |  |  | X |  |
| Funktionale Korrektheit (Programm) |  | X |  |  |
| Funktionale Vollständigkeit (Programm) |  |  | X |  |
| Robustheit gegenüber dem Benutzer (Programm) |  | X |  |  |

Die Qualitätsbestimmung führt normalerweise zu einigen Fragen, hierzu folgendes:

Die Begriffe "gut", "normal", etc. sind schwammig. Es geht u.a. darum Schwerpunkte zu setzen und besonderheiten hervorzuheben.

Als Hilfe zum Ausfüllen:

Denken Sie an Ihre „Kunden“:

* Ist es ein Endprodukt für Leute mit wenig Computer Erfahrung?
  + Dann muss die Benutzungsfreundlichkeit und Robustheit sehr gut sein.
* Soll das Programm weiterentwickelt werden?
  + Dann muss die Änderbarkeit und Verständlichkeit des Codes sehr gut sein.
* Müssen die Resultate auf Biegen und Brechen stimmen?
  + Dies hat direkte Folgen für die funktionale Korrektheit.
* Müssen die Resultate auf Biegen und Brechen stimmen?
  + Dies hat direkte Folgen für die funktionale Korrektheit.
* Müssen alle Knöpfe, Textfelder, Berechnungsoptionen schon funktionieren oder sind die Benutzer flexibel genug um sich der Software anzupassen?
  + Dies hat direkte Folgen für die funktionale Vollständigkeit.

# Testfälle

Wie testen Sie ob Ihre Software tut was sie soll?

/E10/ Funktionstest von sämtlichen Bedienelementen.

/E20/ Falsche Werte eingeben

/E30/ Die berechneten Werte gegen bekannte Lösungen vergleichen.

/E40/ Das Programm auf verschiedenen Computern testen.

# Anhang

Weitere Informationen zum Lastenheft finden Sie in der Datei:

„Lastenheft - Zusammenfassung und weitere Infos.pdf“

Unterlagen zur Messung