

Pflichtenheft

**„Entwicklung einer PC Software für Kinder als
Produktbeilage“**

Stand: 23.01.2020

Auftraggeber: Hochschule für Technik und Wirtschaft Dresden
Friedrich-List-Platz 1
D-01069 Dresden
Ansprechpartner: Herr Lionel Burg

Auftragnehmer: LDJ Software Hochschulstraße 1
D-01069 Dresden
Ansprechpartner: Herr Jannik Schmied

Inhaltsverzeichnis

1 Zielbestimmung	3
1.1 Muss-Kriterien	3
1.2 Kann-Kriterien	3
1.3 Abgrenzungskriterien	3
2 Produkteinsatz	4
2.1 Anwendungsbereich	4
2.2 Zielgruppen	4
2.3 Produktumgebung	4
2.3.1 Architektur	4
2.3.2 Technologie	4
2.3.3 Komponenten	4
2.3.4 Schnittstellen	4
2.4 Betriebsbedingungen	4
3 Produktfunktionen/Anforderungen	5
3.1 Funktionale Anforderungen	5
3.1.1 Beschreibung der FA mit Rollen innerhalb der Geschäftsprozesse	5
3.1.2 Aktivitäten mit Benutzerschnittstelle (UI)	6
3.1.3 Fachliches Klassendiagramm (domain model) / Produktdaten	7
3.2 Nichtfunktionale Anforderungen	7
3.2.1 Benutzbarkeit	7
3.2.2 Zuverlässigkeit	7
3.2.3 Effizienz	7
3.2.4 Softwarewartung	7
3.2.5 Sicherheit	7
3.2.6 Normen	7
4 Testung	7
5 Monitoring/ Support bei Übergabe oder ähnliche Leistungen	7
6 Dokumentation	8
6.1 Anwenderdokumentation	8
6.2 Administratordokumentation	8
6.3 Entwicklerdokumentation	8
6.4 Weitere referenzierte Dokumente	8
7 Vorgehen	8
8 Entwicklungsumgebung	10
9 Glossar	10

1 Zielbestimmung

Für die Firma „Chipmunks Choice“ soll eine PC-Software für Kinder als Produktbeilage zu deren Frühstücksflocken entwickelt werden.

Das zu realisierende Programm soll über eine grafische Oberfläche verfügen, über die der Benutzer mit dem Firmenmaskottchen „Cherry Chipmunk“ interagieren kann. Zudem sollen periodisch Informationstexte angezeigt werden, während sich der Zustand des Maskottchens fortwährend ändert. Die Nahrung des Streifenhörnchens ist limitiert und kann durch Aktivierung eines neu erworbenen Codes aufgefüllt werden. Aufgrund der angestrebten Zielgruppe soll auf eine kindgerechte Umsetzung geachtet werden.

1.1 Muss-Kriterien

MK-IO-01	STD-Output	Implementierung von F1: Die interaktive Benutzeroberfläche ist in einem GUI-Toolkit realisiert.
MK-IO-02	STD-Input	Implementierung von F1, F2: Die grafische Oberfläche muss auf verschiedene Mausereignisse reagieren.
MK-IO-03	STD-Output	Implementierung von F5: In periodischen Zeitabständen werden Label mit Informationen zu Streifenhörnchen angezeigt.
MK-SYS-01	OO-Analyse	Implementierung von F3: Es wird ein Textfeld für die Eingabe von Erweiterungs-codes bereitgestellt. Codes werden nach dem Klicken eines „Einlösen“-Buttons verifiziert
MK-SYS-02	UML2	Implementierung von F4: Die Sicherung des aktuellen Spielstandes erfolgt beim ordnungsgemäßen beenden automatisch in eine Datei. Beim erneuten starten wird dieser aus der Datei geladen.
MK-IMPL-01	Java Code	Die Implementierung muss in Java erfolgen.
MK-IMPL-02	Code Style	Der Java-Code ist nach den Vorgaben des Allman-Style zu implementieren. (https://de.wikipedia.org/wiki/Einr%C3%BCckungsstil#Allman_/_BSD)

1.2 Kann-Kriterien

KK-IO-01	Erzwungene Bewegung	Beim Drücken der „F“ Taste bewegt sich das Maskottchen
----------	---------------------	--

1.3 Abgrenzungskriterien

AK-T-01	Testung	Das System soll keinen Usability-Test durchlaufen
---------	---------	---

2 Produkteinsatz



2.1 Anwendungsbereich

Die Software dient der temporären Beschäftigung der kleinen Kunden der Firma „Cherry Chipmunks“ beziehungsweise ihrem Produkt „Cherry Chipmunks Cereal Choice“.

2.2 Zielgruppen

Benutzt wird die Software vordergründig von Kunden der Altersgruppe 5 – 10 Jahren

2.3 Produktumgebung

Das System benötigt mindestens eine installierte Java Runtime ab Java-Version 1.0. Um Java einfach starten zu können, sollte die Pfad-Variable auf den bin-Ordner der Javaumgebung gesetzt sein. Hardwareanforderungen bestehen keine.

2.3.1 Architektur

Es wird keine spezielle Architektur verwendet. Der Aufbau entspricht einer typischen Java-Anwendung. Die Darstellung wird von der Logik getrennt.

2.3.2 Technologie

Es handelt sich um eine Java-Anwendung, wobei Swing zur Darstellung der grafischen Oberfläche eingesetzt wird.

2.3.3 Komponenten

Aufgrund des Projektumfangs ist es nicht nötig das Projekt in Komponenten zu zerlegen. Eine inhaltliche Trennung wird bei Logik und Darstellung durchgeführt.

2.3.4 Schnittstellen

Bei diesem Projekt gibt es nur die grafische Schnittstelle mit welcher der Nutzer interagiert. Andere Schnittstellen sind nicht vorgesehen.

24 Betriebsbedingungen

Die Software ist für Heimrechner aller Art sowie jegliche Betriebssysteme konzipiert, Voraussetzung hierbei ist lediglich eine Installation von Java. Für die Einrichtung ist einmalig eine Internetverbindung erforderlich.




3.1 Funktionale Anforderungen

3.1.1 Beschreibung der FA mit Rollen innerhalb der Geschäftsprozesse

Das zu entwickelnde Programm hat 5 Anwendungsfälle: den Aufbau der GUI, das Füttern, die Code Eingabe, das Speichern des aktuellen Spielstandes und das Einblenden von Infotexten. Das Programm wird durch den Nutzer über die GUI bedient.

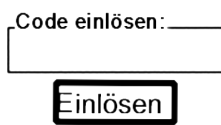
F1	Aufbau des GUI's beim Start	Die Benutzeroberfläche und die Grafiken werden geladen
F2	Füttern des Maskottchens	Durch Mausklick auf die Cerealien soll das Maskottchen gefüttert werden und eine Packung verbraucht werden
F3	Code Eingabe zum Auffüllen der Müslipackungen	Es soll ein Feld geben in den ein Schlüssel eingetragen werden kann. Wenn dieser Korrekt ist soll die Anzahl der zum Füttern zur Verfügung stehenden Müslipackungen erhöht werden.
F4	Speichern und beenden	Beim beenden des Programmes soll der Zustand des Spiels gespeichert werden.
F5	Informationen zu Streifenhörnchen	Nach Ablauf einer gewissen Zeitspanne erscheinen ständig alternierende Informationen zu Streifenhörnchen.

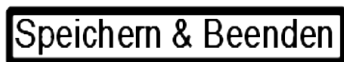
3.1.2 Aktivitäten mit Benutzerschnittstelle (UI)

Anwendungsfall ID	AF - 01
AF Name	Aufbau der GUI beim Start
Akteur	am Betriebssystem angemeldeter Nutzer
Vorbedingung	Die EXE Datei wurde gestartet.
Auslösendes Ereignis	Starten der Anwendung
Nachbedingung Erfolg	Die Grafiken werden geladen und Korrekt dargestellt.
Nachbedingung Fehlschlag	Programm konnte nicht gestartet werden
Ablauf	<ul style="list-style-type: none"> - Doppelklick auf die Anwendung - Grafiken werden geladen - Grafiken werden angezeigt
Benutzerschnittstelle	

Anwendungsfall ID	AF-02
AF Name	Füttern des Maskottchens
Akteur	am Betriebssystem angemeldeter Nutzer
Vorbedingung	Programm ist geöffnet und korrekt gestartet.
Auslösendes Ereignis	Auswahl der Füttern Schaltfläche
Nachbedingung Erfolg	Energie des Maskottchens steigt
Nachbedingung Fehlschlag	Nichts passiert

Ablauf	<ul style="list-style-type: none"> - Schaltfläche auswählen - Energie steigt
Benutzerschnittstelle	

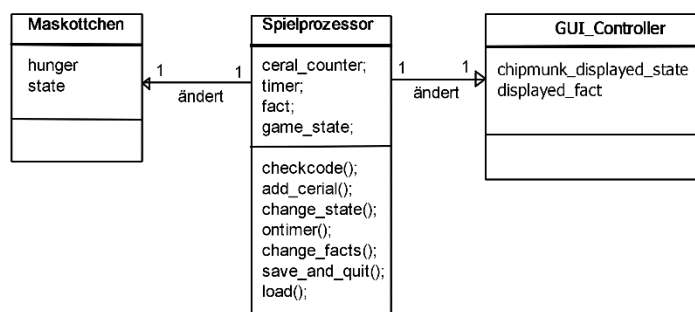
Anwendungsfall ID	AF-03
AF Name	Codeeingabe
Akteur	am Betriebssystem angemeldeter Nutzer
Vorbedingung	Programm ist geöffnet und korrekt gestartet.
Auslösendes Ereignis	Auswahl der Code Schaltfläche
Nachbedingung Erfolg	Verfügbares Essen wird wieder aufgefüllt
Nachbedingung Fehlschlag	Nichts passiert
Ablauf	<ul style="list-style-type: none"> - Schaltfläche auswählen - Essen wird wiederhergestellt
Benutzerschnittstelle	

Anwendungsfall ID	AF-04
AF Name	Speichern und beenden
Akteur	am Betriebssystem angemeldeter Nutzer
Vorbedingung	Programm ist geöffnet und korrekt gestartet.
Auslösendes Ereignis	Auswahl der beenden Schaltfläche
Nachbedingung Erfolg	Der aktuelle Stand wird gespeichert und das Programm wird beendet.
Nachbedingung Fehlschlag	Nichts passiert
Ablauf	<ul style="list-style-type: none"> - Schaltfläche auswählen - Stand in eine Datei speichern - Programm wird beendet
Benutzerschnittstelle	

Anwendungsfall ID	AF-05
AF Name	Informationen zu Streifenhörnchen

Akteur	am Betriebssystem angemeldeter Nutzer
Vorbedingung	Programm ist geöffnet und korrekt gestartet.
Auslösendes Ereignis	Timer
Nachbedingung Erfolg	Eine Information zu Streifenhörnchen wird angezeigt
Nachbedingung Fehlschlag	Nichts passiert
Ablauf	- Nach Ablauf eines Timers wird ein Label angezeigt
Benutzerschnittstelle	<div> Streifenhörnchen Fakten: Es gibt 23 Unterarten von Streifenhörnchen. </div>

3.1.3 Fachliches Klassendiagramm (domain model) / Produktdaten



3.2 Nichtfunktionale Anforderungen

3.2.1 Benutzbarkeit

Das Programm sollte so einfach und simpel wie möglich gestaltet sein damit jedes Kind intuitiv damit arbeiten kann.

3.2.2 Zuverlässigkeit

Die Anwendung muss immer zuverlässig funktionieren da es sich um einen integralen Teil des täglichen Arbeitsprozesses handelt.

3.2.3 Effizienz

Effizienz muss hier nicht explizit betrachtet werden, da durch einfache Benutzung der höchste Grad an Effizienz erreicht wird.

3.2.4 Softwarewartung

Gewartet wird das Programm durch den Hersteller.

3.2.5 Sicherheit

Das Programm muss keinen besonderen Sicherheitsstandards genügen da es keinen Zugriff auf das Netzwerk oder sensible Firmendaten hat.

3.2.6 Normen

NF-B1	Benutzung	Die Anwendung soll nur in Form einer grafischen Oberfläche genutzt werden.
NF-E1	Effizienz	Die Ausgabe des Ereignisses soll unmittelbar erfolgen.
NF-S1	Sicherheit	Diesbezüglich liegen keine Anforderungen vor.
NF-N1	Normen	Die eingeblendeten Labels können andere Grafikelemente überdecken

4 Testung

- Interaktion
- Code einlösen
- Anzeige von Informationstexten
- Spielabläufe

Es wird ein einfacher Funktionstest für alle Anwendungsfälle durchgeführt.

5 Monitoring/ Support bei Übergabe oder ähnliche Leistungen

Im Rahmen des einjährigen Supports für die Software werden folgende Leistungen erbracht:

- Kritische Patches,
- Support,
- Rufbereitschaft 8x5 per E-Mail

6 Dokumentation

6.1 Anwenderdokumentation

Die Anwenderdokumentation wird als typische readme.md Datei in deutscher und englischer Sprache im Repository zur Verfügung gestellt.

6.2 Administratordokumentation

Eine Administratordokumentation ist nicht vorgesehen.

6.3 Entwicklerdokumentation

Als Entwicklerdokumentation werden die mit Javadoc generierten HTML-Dokumente im Repository zur Verfügung gestellt.

6.4 Weitere referenzierte Dokumente

Das Pflichtenheft wurde mit Bezug auf das „Lastenheft - Entwicklung einer PC Software für Kinder als Produktbeilage“ erstellt. Lastenheft, Pflichtenheft und die Anwender-, Administrator- und Entwicklerdokumentation befinden sich im Repository Software Engineering.

7 Vorgehen

Für die Anwendungsfälle AF-01, AF-02 und AF-03 wird ein Prototyp erstellt, der gemäß den nicht funktionalen Anforderungen inkrementell erweitert wird. Danach erfolgt der Funktionstest. Diese letzte Testversion gilt als Release Candidate auf deren Basis auch die Dokumentation abgeschlossen wird (code freeze). Anschließend erfolgt die Übergabe.

Meilensteine sind:

Datum	Meilenstein
14.01.2020	Auftakt
14.01.2020	Projektplan
23.01.2020	Pflichtenheft
28.01.2020	Angebot
TbA	Prototyp
TbA	Release Candidate
TbA	Übergabe

Die Fortschrittskontrolle erfolgt anhand folgender Indikatoren:



Indikator		Kick-Off	Projektplan und Pflichtenheft	Erstellen der Entwicklungs- umgebung	Prototyp	Funktions- test	Release Candidate	Übergabe
Pflichtenheft	Soll		100					
[% erledigte Gliederungs- Punkte]	Ist							
Umgebung	Soll			2				
[Tools]	Ist			2				
Diagramme	Soll		2					
[Anzahl]	Ist		1					
Quellcode	Soll				300			
[LOC]	Ist				50			
Verhältnis	Soll				5:1			
[LOC/ Kommentare]	Ist							
Tests	Soll					1		
[Fälle/ Methoden]	Ist					0		
Anwenderdoku	Soll					50		
[Wörter]	Ist							
Entwicklerdoku	Soll					50		
[Wörter]	Ist							
Release	Soll						5	5
[Artefakte]	Ist							

8 Entwicklungsumgebung

Für die Entwicklung der Software werden wir JetBrains' IntelliJ IDEA in der Version 2019.3.2 benutzen.

Die Entwicklerdokumentation wurde mit Javadoc erstellt, der Quellcode ist entsprechend kommentiert.

An die Hardware und Orgware bestehen keine besonderen Anforderungen.

9 Glossar

AG: Auftraggeber

AN: Auftragnehmer

FA: Funktionale Anforderungen

UI: User Interface (Benutzerschnittstelle), hier Ein- und Ausgabe von Text in der Konsole