**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](#_bookmark0)
   1. [Muss-Kriterien 3](#_bookmark1)
   2. [Kann-Kriterien 3](#_bookmark2)
   3. [Abgrenzungskriterien 3](#_bookmark3)
2. [Produkteinsatz 4](#_bookmark4)
   1. [Anwendungsbereich 4](#_bookmark5)
   2. [Zielgruppen 4](#_bookmark6)
   3. [Produktumgebung 4](#_bookmark7)
      1. [Architektur 4](#_bookmark8)
      2. [Technologie 4](#_bookmark9)
      3. [Komponenten 4](#_bookmark10)
      4. [Schnittstellen 4](#_bookmark11)
   4. [Betriebsbedingungen 4](#_bookmark12)
3. [Produktfunktionen/Anforderungen 5](#_bookmark13)
   1. [Funktionale Anforderungen 5](#_bookmark14)
      1. [Beschreibung der FA mit Rollen innerhalb der Geschäftsprozesse 5](#_bookmark15)
      2. [Aktivitäten mit Benutzerschnittstelle (UI) 6](#_bookmark16)
      3. [Fachliches Klassendiagramm (domain model) / Produktdaten 7](#_bookmark17)
   2. [Nichtfunktionale Anforderungen 7](#_bookmark18)
      1. [Benutzbarkeit 7](#_bookmark19)
      2. [Zuverlässigkeit 7](#_bookmark20)
      3. [Effizienz 7](#_bookmark21)
      4. [Softwarewartung 7](#_bookmark22)
      5. [Sicherheit 7](#_bookmark23)
      6. [Normen 7](#_bookmark24)
4. [Testung 7](#_bookmark25)
5. [Monitoring/ Support bei Übergabe oder ähnliche Leistungen 7](#_bookmark26)
6. [Dokumentation 8](#_bookmark27)
   1. [Anwenderdokumentation 8](#_bookmark28)
   2. [Administratorendokumentation 8](#_bookmark29)
   3. [Entwicklerdokumentation 8](#_bookmark30)
   4. [Weitere referenzierte Dokumente 8](#_bookmark31)
7. [Vorgehen 8](#_bookmark32)
8. [Entwicklungsumgebung 10](#_bookmark33)
9. [Glossar 10](#_bookmark34)

# 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.

# 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 Erweiterungscodes 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) |

# Kann-Kriterien

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

# Abgrenzungskriterien

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

# Produkteinsatz

# Anwendungsbereich

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

# Zielgruppen

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

# 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.

# Architektur

Es wird keine spezielle Architektur verwendet. Der Aufbau entspricht einer typischen Java-Anwendung.

Die Darstellung wird von der Logik getrennt.

# Technologie

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

# Komponenten

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

# Schnittstellen

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

# 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.

# Funktionale Anforderungen

# 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. |

# 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 | * Doppelklick auf die Anwendung * Grafiken werden geladen * Grafiken werden angezeigt |
| Benutzerschnittstelle |  |

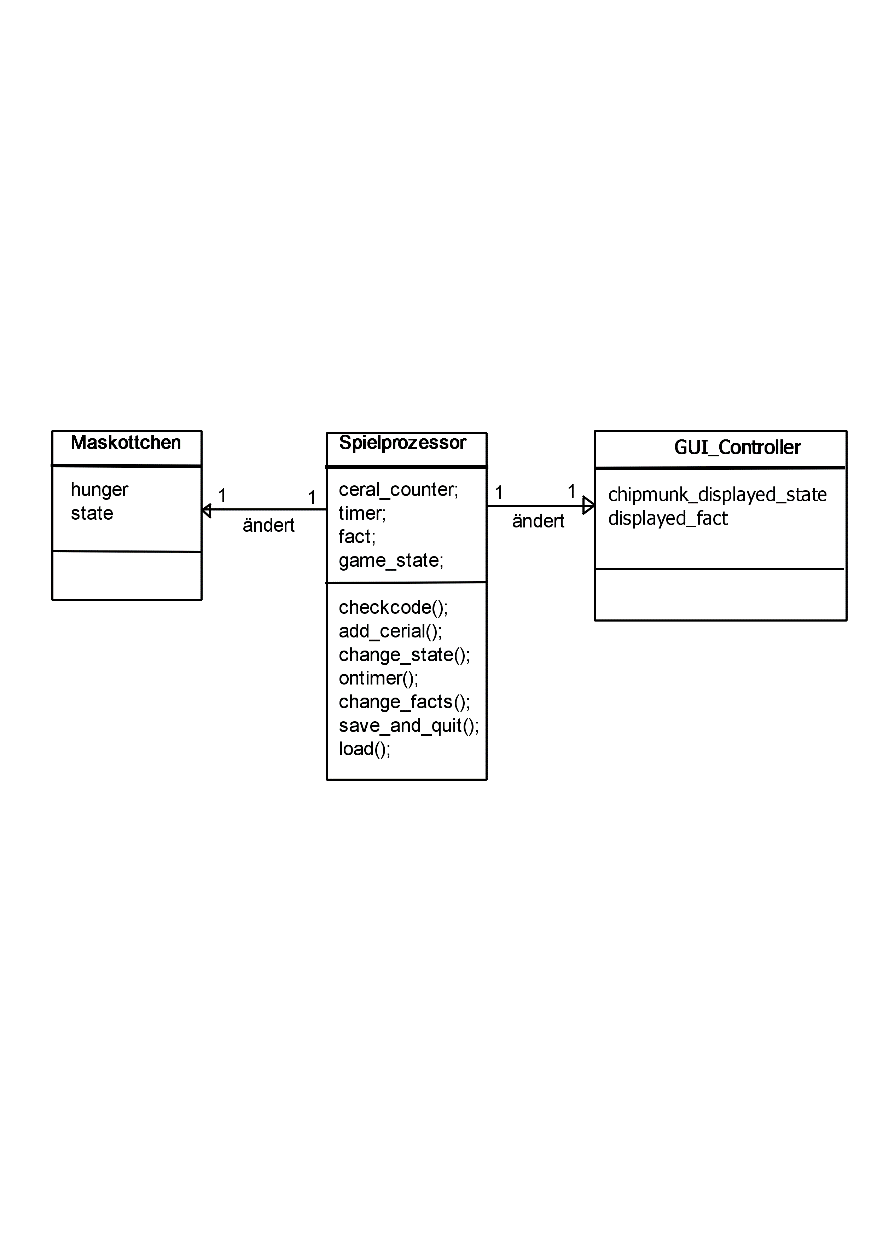
|  |  |
| --- | --- |
| Anwendungsfall ID | AF-02 |
| AF Name | Füttern des Maskottchens |
| Akteur | am Betriebssystem angemeldeter Nutzer |
| Vorbedingung | Programm ist geöffnet und korrekt gestartet. |
| Auslösendes Ereignis | Auswahl der Füttern Schaltfläche |
| Nachbedingung Erfolg | Energie des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

* + 1. **Fachliches Klassendiagramm (domain model) / Produktdaten**



# Nichtfunktionale Anforderungen

* + 1. **Benutzbarkeit**

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

* + 1. **Zuverlässigkeit**

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

* + 1. **Effizienz**

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

* + 1. **Softwarewartung**

Gewartet wird das Programm durch den Hersteller.

* + 1. **Sicherheit**

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

* + 1. **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 |

1. **Testung**

* Interaktion
* Code einlösen
* Anzeige von Informationstexten
* Spielabläufe

Es wird ein einfacher Funktionstest für alle Anwendungsfälle durchgeführt.

# 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

# Dokumentation

* 1. **Anwenderdokumentation**

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

# Administratorendokumentation

Eine Administratorendokumentation ist nicht vorgesehen.

# Entwicklerdokumentation

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

# 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.

# 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 |
| 20.05.2020 | 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 [% erledigte Gliederungs-  Punkte] | Soll |  | 100 |  |  |  |  |  |
| Ist |  |  |  |  |  |  |  |
| Umgebung [Tools] | Soll |  |  | 2 |  |  |  |  |
| Ist |  |  | 2 |  |  |  |  |
| Diagramme [Anzahl] | Soll |  | 2 |  |  |  |  |  |
| Ist |  | 1 |  |  |  |  |  |
| Quellcode [LOC] | Soll |  |  |  | 300 |  |  |  |
| Ist |  |  |  | 50 |  |  |  |
| Verhältnis [LOC/  Kommentare] | Soll |  |  |  | 5:1 |  |  |  |
| Ist |  |  |  |  |  |  |  |
| Tests [Fälle/  Methoden] | Soll |  |  |  |  | 1 |  |  |
| Ist |  |  |  |  | 0 |  |  |
| Anwenderdoku [Wörter] | Soll |  |  |  |  | 50 |  |  |
| Ist |  |  |  |  | 140 |  |  |
| Entwicklerdoku [Wörter] | Soll |  |  |  |  | 50 |  |  |
| Ist |  |  |  |  | 60 |  |  |
| Release [Artefakte] | Soll |  |  |  |  |  | 5 | 5 |
| Ist |  |  |  |  |  |  |  |

# 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.

# Glossar

AG: Auftraggeber

AN: Auftragnehmer

FA: Funktionale Anforderungen

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