



SYSTEMENTWURF: TETRIS

PFLICHTENHEFT

Andrej, Jan, Xiaojie, Zhenya



8. NOVEMBER 2022

AIF31

Inhaltsverzeichnis

1 Einleitung	2
2 Ausgangssituation und Zielsetzung	3
3 Dekomposition des Gesamtsystems	4
4 Schnittstellenübersicht.....	5
5 Lebenszyklusanalyse.....	6
6 Funktionale Anforderungen	7
7 Nicht-Funktionale Anforderungen	8
8 ANFORDERUNGSVERFOLGUNG ZUM LASTENHEFT.....	9
9 ANFORDERUNGSVERFOLGUNG ZU DEN SPEZIFIKATIONEN	10
10 ABNAHMEKRITERIEN UND VORGEHEN ZUR AUSGANGSPRÜFUNG	11
11 Lieferumfang	12

1 Einleitung

Im ersten Halbjahr der Klasse 13 haben wir als kleine vierer Gruppe im Zusammenhang unseres Softwareunterrichtes gemeinsam vor, an einer Softwareentwicklung zu arbeiten.

Zuerst wurden im Unterricht die Ideen zu einer Softwareentwicklung zusammengetragen. Im Anschluss wurde die Gruppen- / Themeneinteilung von unserem Softwarelehrer bekannt gegeben.

Wir wollen als kleine Gruppe das Spiel Tetris versuchen zu Programmieren.

Im Laufe des Unterrichtes einigten wir uns darauf, ein Pflichtenheft zu erstellen, in dem das Lastenheft beinhaltet werden soll.

Wie im letzten Satz schon erwähnt, soll das vorliegende Dokument unser gemeinsames Ergebnis der noch kommenden Softwareentwicklung darstellen.

2 Ausgangssituation und Zielsetzung

Die Ausgangssituation bildet wie oben schon beschrieben unser Softwareunterricht. Grob zusammengefasst fand dort die Themen und Gruppeneinteilung statt.

Unser Hauptziel besteht darin, uns mit den Arbeitsmethoden auseinander zu setzen. Darunter sollen wir auch lernen wie die Zusammenarbeit an einem Projekt in einem kleineren Team auszusehen, das beinhaltet sowohl die Arbeitsteilung untereinander als auch die Absprache und Zeiteinteilung während des gesamten Projektentwurfes.

Zudem gehört auch der eigentliche Softwareentwurf, d.h. wir einigen uns untereinander wer welchen Teil der Software programmiert und führen das am Ende unserer Zusammenarbeit als gemeinsames laufendes Programm unserer Klasse im Softwareunterricht vor.

Das Programm soll auf einen PC mit Windowsbetriebssystem laufen können. Wir werden die Programmiersprache C# zur eigentlichen Softwareentwicklung benutzen. Es wird vorausgesetzt das auf dem PC, auf dem das Programm laufen soll, Visual Studios 2022 Community Edition installiert ist.

Während des Projektes werden wir Kanban für den Austausch untereinander benutzen.

Am Ende des Projektes sollen wir eine exe Datei, den Quellcode und die Projektdokumentation abgeben.

3 Dekomposition des Gesamtsystems

Das Programm Tetris beinhaltet:

1. Ein Start Menü mit 4 Buttons zur Navigation
 - 1.1. Start: führt das Spiel aus
 - 1.2. Einstellungen: Lautstärke und Schwierigkeitsgrad ändern und speichern
 - 1.3. Steuerung: Tasten werden angezeigt, welche Du für das Spiel brauchst
 - 1.4. Highscore: Sehe dich auf der Rangliste
2. Eine SQL-Datenbank, um den Highscore und den Namen zu speichern
3. Folgende UML-Diagramme wurden erstellt
 - a. Anwendungsfalldiagramm
 - b. Klassendiagramm
 - c. Sequenzdiagramm
 - d. Projektablaufplan

4 Schnittstellenübersicht

Das Spiel soll über eine Windows-Forms GUI gespielt werden.

Die Highscores sollen auf einer online Datenbank gespeichert werden.

5 Lebenszyklusanalyse

Das Projekt gilt nach der Produktabnahme als beendet.

6 Funktionale Anforderungen

Anmeldung

1. Optionale Anmeldung mit beliebigem, unbenutztem Namen
2. Optionales synchronisieren von Rekorden

Game

1. 7 Verschiedene Blöcke
2. Ein zufälliger Block wird am oberen Rand erstellt
3. Block kann bewegt werden
4. Wird automatisch nach unten bewegt
5. Bei voll ausgefüllter Reihe wird die Reihe entfernt
6. Nach x entfernten Reihen wird die Spielgeschwindigkeit erhöht

Post-Game

1. Score wird synchronisiert
2. Rangliste wird angezeigt

7 Nicht-Funktionale Anforderungen

Performance

Die Highscore Synchronisation soll von 30 Benutzern gleichzeitig benutzt werden können.

Benutzbarkeit

Die Software soll durch Schüler ohne Erklärung bedient werden können.

8 ANFORDERUNGSVERFOLGUNG ZUM LASTENHEFT

9 ANFORDERUNGSVERFOLGUNG ZU DEN SPEZIFIKATIONEN

10 ABNAHMEKRITERIEN UND VORGEHEN ZUR AUSGANGSPRÜFUNG

1. Die UML-Diagramme liegen vor
2. Der Quellcode liegt vor
3. Das Programm ist ausführbar und stürzt nicht beim Starten ab
4. Das Pflichtenheft liegt vor
5. Alle Mitglieder des Projektes können auf alle zugehörigen Sachen auf GitHub zugreifen

11 Lieferumfang

1. .EXE Datei die Auf einem aktuellem Windows-PC lauffähig ist
2. Project Quellcode über GitHub bereitgestellt
3. MySQL Datenbank
4. Planungsdokumente