

Donnerstag, 6. Juli 2023

Projektarbeit

Runner 96 – Computerspiel in Visual Basic

Linus von Maltzan, Erik Siegel, Philipp Brocher

Studiengang TWE22

Dualen Hochschule Baden-Württemberg Ravensburg

Erklärung zum Programmentwurf

gem. § 5(2) der „Studien- und Prüfungsordnung DHBW Technik“ vom 18. Mai 2009.

Hiermit erkläre ich, dass ich den vorgelegten Programmentwurf selbständig angefertigt, nicht anderweitig zu Prüfungszwecken vorgelegt und keine anderen als die angegebenen Hilfsmittel verwendet habe.

_____ Ort	_____ Datum	_____ Mat.-Nr.	_____ Unterschrift
--------------	----------------	-------------------	-----------------------

_____ Ort	_____ Datum	_____ Mat.-Nr.	_____ Unterschrift
--------------	----------------	-------------------	-----------------------

_____ Ort	_____ Datum	_____ Mat.-Nr.	_____ Unterschrift
--------------	----------------	-------------------	-----------------------

Inhaltsverzeichnis

1	Einleitung.....	1
1.1	Aufgabe	1
1.2	Spielidee	1
2	Anforderungskatalog	2
2.1	Start der Software	2
2.2	Hauptmenü	3
2.2.1	„Play/Play Again“ Untermenü	3
2.2.2	“High Scores” Untermenü	4
2.2.3	“Change Name” Untermenü	5
2.2.4	„Music Disabled/Enabled“ Menüpunkt	5
2.2.5	„Credits“ Untermenü	5
2.3	Grundfunktionen des Spiels	5
2.3.1	Spieloberfläche	6
2.3.2	Spielfigur.....	7
2.3.3	Spielablauf.....	8
2.4	Sonstige Erklärungen	9
2.4.1	Bedienung der Software.....	9
2.4.2	Spielerlebnis.....	9
2.4.3	Animation der Spielfigur	10
2.4.4	Längeneinheiten	10
3	Struktogramme des Spielablaufs.....	11
3.1	Main.....	11

3.2	SoundManager	11
3.3	KeyInput.....	12
3.4	MenuLoop.....	13
3.5	RoundStart.....	13
3.6	ArraySetUp	13
3.7	PlayerAnimator	13
3.8	GroundAnimator	14
3.9	CollisionManager.....	14
3.10	GameOver.....	14
4	Programmierung	15
4.1	Informationsquellen	15
4.2	Tools	16
5	Programmtest	16
5.1	Testpläne	16
5.2	Testergebnis	23
6	Fazit und Ausblick	23
7	Literaturverzeichnis	25

1 Einleitung

Diese Arbeit beschäftigt sich mit dem Entwicklungsprozess eines konsolenbasierten Spiels in Visual Basic, das sich von den beliebten Arcade-Spielen der 70er Jahre inspirieren lässt.

1.1 Aufgabe

Die Spieler sollen sich wie für dieses Genre typisch durch eine Vielzahl von Hindernissen manövrieren und dabei versuchen, einen hohen Punktestand zu erreichen, während die Level zunehmend anspruchsvoller werden. Um dieses Projekt erfolgreich umzusetzen, soll zuerst ein umfassender Anforderungskatalog erstellt werden auf dessen Basis später das Spiel programmiert wird. Zusätzlich sollen Struktogrammen angelegt werden, um den Ablauf des Spiels zu veranschaulichen. Um die Zuverlässigkeit und Qualität des Spiels zu gewährleisten, werden wir im letzten Schritt umfangreiche Testpläne erstellen und den Programcode auf seine Robustheit prüfen.

1.2 Spielidee

In "Runner 96" besteht das Hauptziel des Spielers darin, so weit wie möglich zu laufen. Allerdings ist er einer tückischen Welt voller Hindernissen ausgesetzt. Der Spieler kann nicht anhalten, ihm bleibt daher nur eine Möglichkeit: Springen und den Hindernissen ausweichen. Mit zunehmendem Fortschritt im Spiel werden immer mehr Hindernisse auf den Spieler zukommen und sein Reaktionsvermögen auf die Probe stellen.

2 Anforderungskatalog

2.1 Start der Software

Beim ersten Start der Software erscheint der Startbildschirm mit dem Namen des Spiels.

Mit einem Klick auf die Leertaste wird der Startbildschirm geschlossen und das erste von vier Startmenüs wird geöffnet.

Die einzelnen Startmenüs werden über einen Klick auf die Space/Return-Taste aufgerufen.

Die 4 Startmenüs sind chronologisch hintereinander angeordnet (von 1 bis 4).

Erfolgt bei geöffnetem Startmenü die Eingabe der Space/Return-Taste durch den Benutzer, so wird das aktuelle Startmenü geschlossen und das chronologisch nächste Startmenü wird gestartet.

Wenn das 4. Startmenü aufgerufen ist und die Eingabe der Space/Return-Taste erfolgt, wird das Hauptmenü geöffnet.

Inhaltlicher Aufbau der 4 Startmenüs

Startmenü 1:	Begrüßung des Benutzers Was ist das Ziel des Spiels? Motivationssatz
Startmenü 2:	Erklärung der Benutzeroberfläche Wie Spiele ich? Wie bediene ich das Menü?
Startmenü 3:	Eingabe des Namens für den Highscore
Startmenü 4:	Motivation für den Benutzer vor dem endgültigen Start des Spiels

2.2 Hauptmenü

Das Hauptmenü ist der zentrale Bestandteil der Software.

Über das Hauptmenü können alle Funktionen und Inhalte der Software über Untermenüs aufgerufen werden.

Bedienung:

Drücken der Pfeiltaste nach oben	Navigation durch das Menü nach oben
Drücken der Pfeiltaste nach unten	Navigation durch das Menü nach unten
Drücken der Space/Return-Taste	Ausgewählte Aktion (blau hinterlegt) wird ausgeführt

Allgemeine Bedienung zum Öffnen/Schließen von Untermenüs:

Gilt für die Untermenüs: „High-Scores“; „Change-Name“; „Credits“

Bei geöffnetem Hauptmenü: Durch die Eingabe der Space/Return-Taste wird das ausgewählte Untermenü gestartet.

Innerhalb der Untermenüs wählt der Spieler die Aktion „zurück“ aus, um zum Startbildschirm zurückzukehren.

Über die Space/Return-Taste wird das geöffnete Untermenü beendet und das Hauptmenü wird gestartet.

Das Untermenü „Play/Play-Again“ wird beendet, wenn die Spielfigur mit einem Hindernis kollidiert und der Spieler anschließend die Space/Return-Taste drückt.

Das Untermenü „Music Disabled/Enabled“ hat zwei Zustände, welche durch die Space/Return-Taste abwechselnd aktiviert werden.

2.2.1 „Play/Play Again“ Untermenü

Wird das Untermenü „Play Again“ aufgerufen, so startet das Spiel.

Der genaue Ablauf des Spiels ist im Gliederungspunkt „3. Grundfunktionen des Spiels“ genauer beschrieben.

Der Spieler spielt so lange, bis die Spielfigur mit einem Hindernis kollidiert.

Wenn die Spielfigur mit einem Hindernis kollidiert, wird das Untermenü „Play/Play Again“ über die Space/Return-Taste beendet und das Hauptmenü wird gestartet.

Beim ersten Aufruf des Hauptmenüs, nach dem Start der Software, heißt dieses Untermenü „Play“.

Das Untermenü „Play“ wird nach der ersten gespielten Runde in „Play Again“ umbenannt.

2.2.2 “High Scores” Untermenü

Alle Highscores werden auf Firebase gespeichert.

Was wird gespeichert?

Die ID der gespielten Runde, der Name des aktuellen Spielers, der Zeitpunkt des Rundenendes und der Highscore der gespielten Runde.

Den Highscore sortiert die Datenbank absteigend.

Der Highscore zeigt die 8 besten gespielten Runden (Highscores) an.

Der Highscore wird nur ausgegeben, wenn der Spieler auf seinem mobilen Endgerät eine Internetverbindung sicherstellt.

Kann der Spieler auf seinem mobilen Endgerät keine Internetverbindung sicherstellen, wird im Untermenü „High-Scores“ eine Fehlermeldung ausgegeben („UNABLE TO CONNECT TO DATABASE“).

2.2.3 “Change Name” Untermenü

Startet der Spieler das “Change Name” Untermenü, wird er aufgefordert einen Namen einzugeben.

Die maximale Länge des Namens beträgt 15 Zeichen.

Das Eingabefeld zur Eingabe eines Namens wird im Untermenü „Change-Name“ mittig visualisiert.

!!!

2.2.4 „Music Disabled/Enabled“ Menüpunkt

Der Menüpunkt „Music Disabled/Enabled“ ist kein reines Untermenü, hebt sich optisch aber nicht von der Untermenüs ab.

Der Menüpunkt hat zwei Zustände: Musik ein oder Musik aus

Über die Space/Return-Taste werden die zwei Zustände abwechselnd aktiviert.

Beim ersten Aufruf des Hauptmenüs ist die Musik aktiviert.

2.2.5 „Credits“ Untermenü

Im „Credits“ Untermenü werden die Ersteller der Software und der Soundtracks namentlich benannt.

Die jeweiligen Personen werden zentriert untereinander aufgelistet.

2.3 Grundfunktionen des Spiels

Das grundlegende Spiel besteht aus einer Spieloberfläche, der Spielfigur und einem festgelegten Spielablauf.

Der 3. Gliederungspunkt Grundfunktionen des Spiels bezieht sich auf das Untermenü „Play/Play Again“.

2.3.1 Spieloberfläche

Die Spieloberfläche besteht aus dem Untergrund, den Hindernissen und den Wolken.

2.3.1.1 Hindernisse

Die Hindernisse erscheinen in zufälligem Abstand auf dem Untergrund und kommen von rechts auf die Spielfigur zu.

Die Hindernisse sind verschiedene Hürden bestehend aus einer Kombination aus folgenden Zeichen ``[,], |`` (2 Längeneinheiten hoch)

Auf dem Untergrund wird beim Start kein Hindernis angezeigt.

Unmittelbar nach Spielbeginn erscheinen die ersten Hindernisse

Die Hindernisdichte auf der Spieloberfläche nimmt im Verlauf des Spiels zu (von 1 bis 50 Hindernissen).

Maximal werden 50 Hindernisse auf der Spieloberfläche angezeigt.

Die Spielfigur kann nicht durch die Hindernisse durchlaufen, sondern sie muss diese durch Springen überwinden.

2.3.1.2 Position der Spielfigur, Untergrund und Wolken

Nach dem Start des Spiels wird auf der Spieloberfläche die Spielfigur angezeigt. Diese wird gemäß dem Algorithmus animiert (laufen, springen+landen).

Die Spielfigur wird am linken Bildschirmrand oberhalb des Untergrunds visualisiert.

Der Untergrund auf dem sich die Spielfigur bewegt ist ein horizontaler Strich am unteren Rand der Spielfläche.

Im oberen Bereich der Spielfläche werden Wolken visualisiert.

Es gibt zwei verschiedene Arten von Wolken, die visualisiert sind.

Die Wolken werden in einem begrenzten Intervall (oberer Bereich der Spielfläche) mit zufälligen y-Parametern erzeugt.

Die Wolken bewegen sich vom rechten Bildschirmrand zum linken Bildschirmrand, bis sie wieder ausgeblendet werden.

Die Wolken werden grafisch durch folgende Symbole visualisiert:

"_"; "("; ")"

2.3.2 Spielfigur

-Die Spielfigur ist notwendig, um das Spiel zu spielen.

-Die Spielfigur bildet einen Menschen grafisch durch eine Kombination durch folgende Symbole ab:

"_"; "@"; "|"; "/" ; "\" ; "." ; "," ; "`" ; "-"; ":" ; "L"; ""

- Die Spielfigur ist 8 Längeneinheiten hoch und 5 Längeneinheiten breit.

-Die Spielfigur wird auf einer Stelle animiert und die Hindernisse kommen von rechts auf die Spielfigur zu.

-Die Spielfigur hat eine Laufanimation bestehend aus 8 Schritten (Case0-Case7).

-Mit dem Start des Spiels fängt die Spielfigur an zu laufen.

-Die Spielfigur springt 5 Längeneinheiten hoch

- Ein Sprung besteht aus der Fähigkeit zu springen und wieder zu landen.

- Die Bewegung der Spielfigur wird durch den Spieler gesteuert. Klickt er auf die Leertaste, so springt die Spielfigur ab und landet wieder. Andernfalls läuft die Spielfigur auf dem Untergrund weiter.
- Die Spielfigur bewegt sich so lange bis sie ein Hindernis berührt.
- Bei Berührung mit einem Hindernis verliert die Spielfigur ihr einziges Leben.
- Die Animation stoppt und das Spiel ist zu Ende (Game Over).

2.3.3 Spielablauf

- Bei Spielbeginn hat der Spieler ein Leben.
- Im Spielverlauf nehmen Hindernisdichte und -geschwindigkeit zu.
- Eine gespielte Runde endet, wenn alle Leben verbraucht sind, dann Anzeige „Game Over“.
- Nach dem Ende des Spiels wird der Benutzer zum Hauptmenü weitergeleitet.
- Über das Hauptmenü kann das Spiel neu gestartet werden.
- Wenn das Untermenü „Play/Play Again“ aktiv ist, wird der Score, der Timer und die FPS am linken oberen Bildschirmrand untereinander aufgelistet.

Score: Der Score steigt mit jedem überwundenem Hindernis. Der Score ist eine Zahl mit dem Wertebereich $Z=\{0, \infty\}$. Der Score wird nach jeder gespielten Runde zurück gesetzt auf 0.

Timer: Der Timer gibt die gespielte Zeit pro gespielter Runde an. Nach jeder gespielten Runde wird der Timer auf 0 zurück gesetzt. Der Timer startet mit jeder neu gestarteten Runde bei 0 und wird mit jeder Sekunde um eine Zahl erhöht.

FPS: FPS=Bilder pro Sekunde. Die FPS werden in Echtzeit vom System ausgegeben.

2.4 Sonstige Erklärungen

2.4.1 Bedienung der Software

Mobiles Endgerät: Um die Software visuell aufzurufen und zu bedienen, benötigt der Spieler ein mobiles Endgerät, welches mit der Software kompatibel ist. Ein mobiles Endgerät ist beispielsweise ein Laptop.

Aktion (im Bezug auf die Bedienung des Hauptmenüs): Aktionen sind alle möglichen Funktionen der Software, die durch den Spieler über die Pfeiltasten ausgewählt werden und anschließend durch die Space/Return-Taste ausgeführt werden.

2.4.2 Spielerlebnis

Spieler: Der Spieler ist der Endanwender der Codes, welcher das Spiel am Ende spielt. Er ist nicht zu verwechseln mit der digitalen Spielfigur, welche bereits genauer definiert wurde. Der Spieler steuert die Spielfigur über die Leertaste.

Spiel: Das Spiel ist der zentrale Bestandteil der Software und wird durch das „Play/Play Again“ Untermenü aufgerufen. Der Spieler versucht die Spielfigur über die Hindernisse zu steuern und so lange wie möglich nicht mit einem Hindernis zu kollidieren.

Gespielte Runde: Eine gespielte Runde beginnt mit dem Aufruf des Untermenüs „Play/Play Again“, wenn das Hauptmenü geschlossen wird. Kollidiert die Spielfigur mit einem Hindernis, wird das Untermenü „Play/Play Again“ über Tasteneingabe geschlossen und das Hauptmenü gestartet. Hiermit ist das Kriterium einer gespielten Runde erfüllt.

Spieloberfläche: Ist der Bereich, der für den Anwender des Programms auf seinem mobilen Endgerät zu sehen ist.

2.4.3 Animation der Spielfigur

Laufen: „Laufen ist eine Methode der terrestrischen Fortbewegung, die es Menschen und anderen Tieren ermöglicht, sich schnell zu Fuß fortzubewegen. Laufen ist eine Gangart, die durch eine Luftphase gekennzeichnet ist, in der sich alle Füße über dem Boden befinden.“ [1] Die Spielfigur schwingt ihre Arme gegenläufig nach vorne und hinten. Dasselbe passiert gegenläufig mit den Beinen.

Springen: „Springen ist das Abschnellen des Körpers vom Boden, wobei ein oder mehr Füße eingesetzt werden, um Höhe, Weite oder Tiefe zu überwinden.“ [2] Die Spielfigur wird aus ihrer Bewegung heraus nach oben verschoben, bis sie ihren Hochpunkt erreicht hat. Dabei bewegt sie Arme und Beine ganz normal weiter. (Animation für das Laufen)

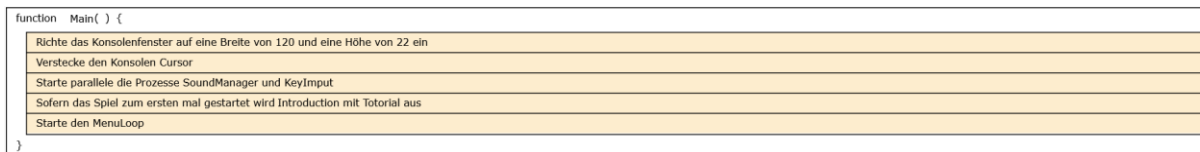
Landen: Von oben herab auf dem Boden aufsetzen.; Nachdem die Spielfigur ihren Hochpunkt erreicht hat, wird sie wieder schrittweise in Richtung Weg (Anfangshöhe vor dem Sprung) bewegt, bis sie endgültig wieder in die Position des Laufens zurückkehrt und weiterläuft. Dabei bewegt sie Arme und Beine ganz normal weiter (Animation für das Laufen).

2.4.4 Längeneinheiten

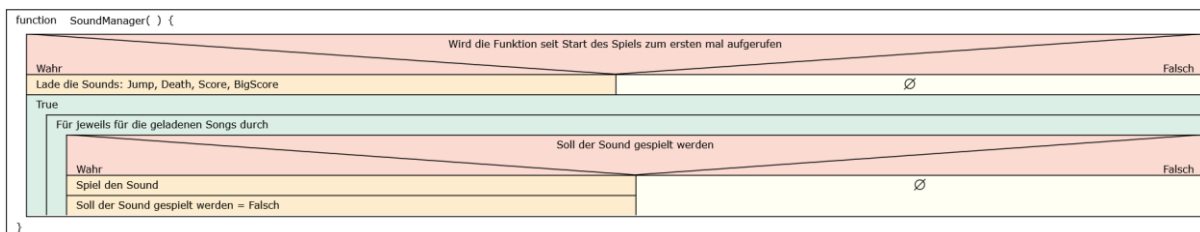
Eine Spalte bzw. eine Zeile des Konsolenfensters.

3 Struktogramme des Spielablaufs

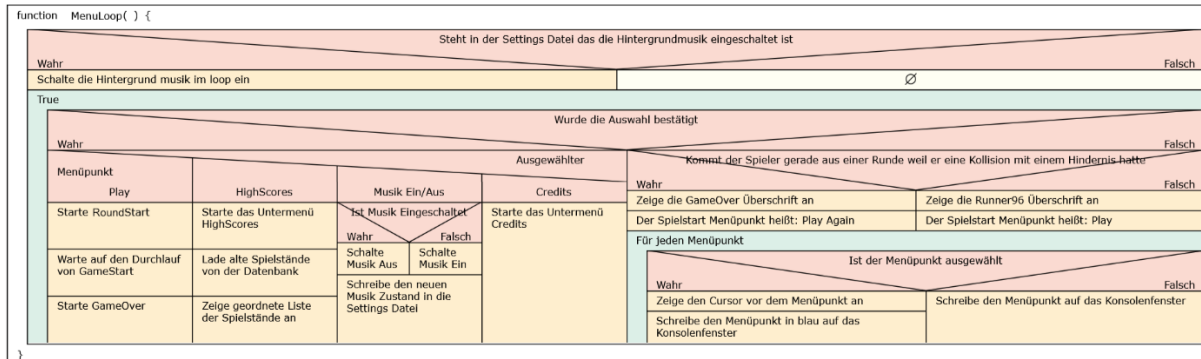
3.1 Main



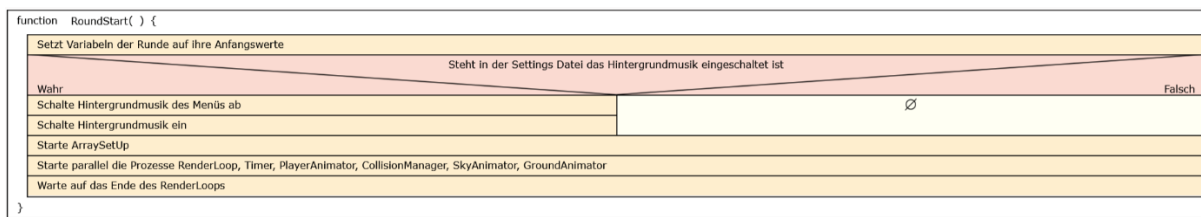
3.2 SoundManager



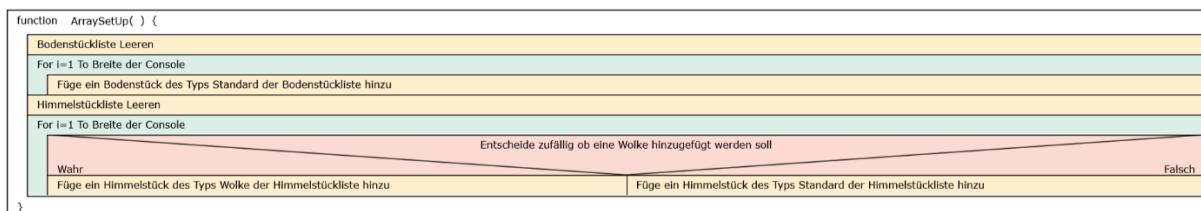
3.4 MenuLoop



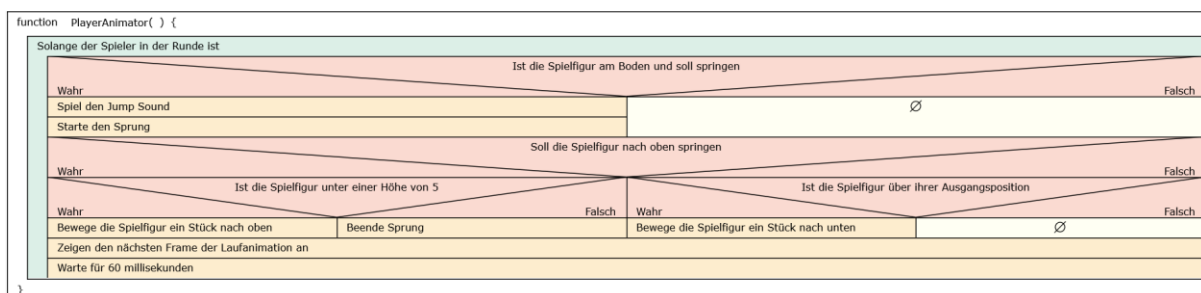
3.5 RoundStart



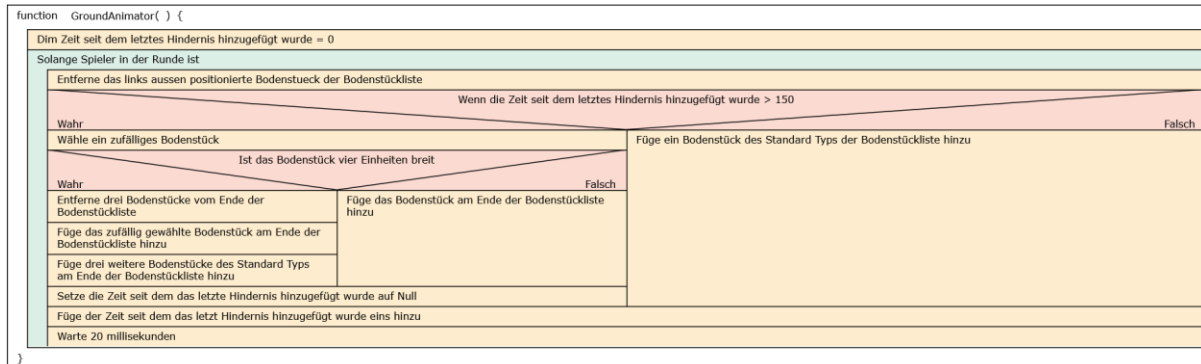
3.6 ArraySetUp



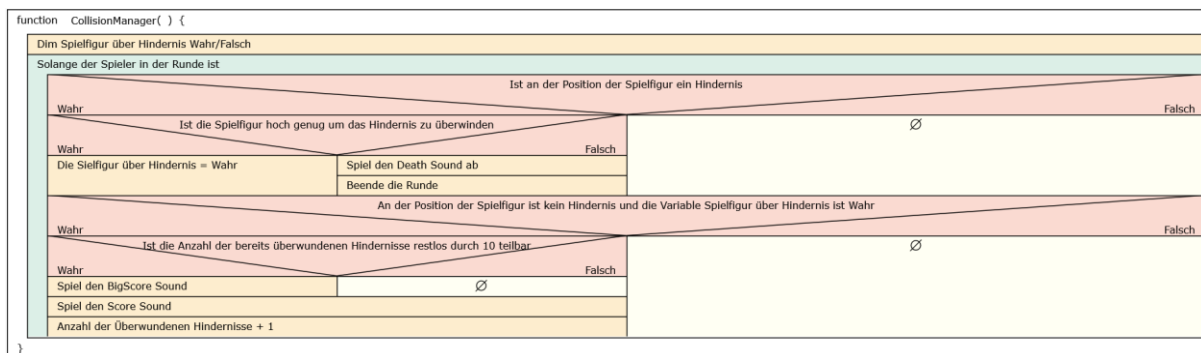
3.7 PlayerAnimator



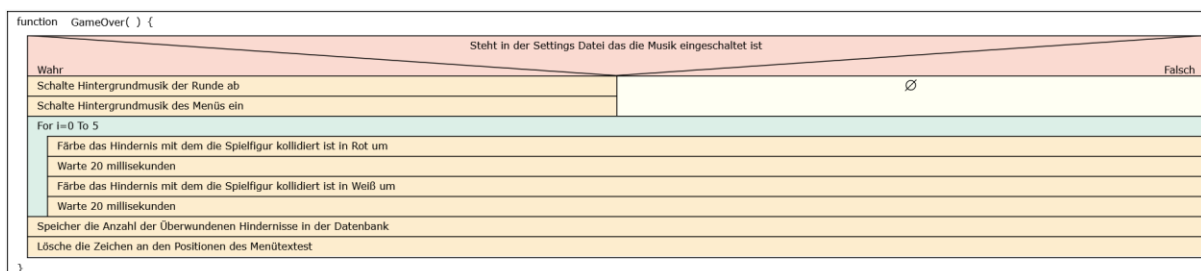
3.8 GroundAnimator



3.9 CollisionManager



3.10 GameOver



4 Programmierung

4.1 Informationsquellen

Visual Basic ist eine schon etwas in die Jahre gekommene Programmiersprache. Doch obwohl keiner von uns VB zuvor genutzt hatte, war uns die Programmiersprache von Anfang an vertraut. Was daran liegt, dass sie wie die uns bereits bekannte Programmiersprache C# auf dem robusten .NET Framework basiert. Das heißt jedoch nicht, dass wir ohne zu Googeln ausgekommen sind. Die offizielle Microsoft-Dokumentation [3] stellte sich als sehr gute Informationsquelle für Visual Basics heraus. Aber auch Stack Overflow, der beste Freund eines jeden Programmierers, half uns oft weiter, wenn wir nach Lösungen suchten.

Ein Feature, das uns sehr wichtig war, ist die Möglichkeit, mehrere Sounds parallel abspielen zu können, beispielsweise einen Sprungsound, ohne die Hintergrundmusik zu unterbrechen. Da diese Funktion nicht nativ in Visual Basic enthalten ist, griffen wir auf die Windows Multimedia API zurück, deren Implementierung Yang Kok Wah in seinem Artikel [4] sehr gut erklärt hat.

Da die Highscore-Funktion ein wichtiger Bestandteil unserer Spielidee war, planten wir von Anfang an die Möglichkeit ein, den eigenen Highscore mit dem anderer Spieler vergleichen zu können. Firebase ist ein sehr leistungsfähiges Tool für solche Aufgaben, da man mit nur wenigen Zeilen Code vollwertige Datenbanklösungen kostenlos implementieren kann. Ein sehr hilfreiches Tutorial zur Integration mit FireSharp [5] war das Video [6] von Hassaan Raza von The Amazing Codeverse.

4.2 Tools

Der Großteil der Entwicklung erfolgte in Visual Studio [7]. Diese leistungsstarke IDE bot uns effiziente Tools und Funktionen zum Schreiben, Debuggen und Testen des Codes.

Um effektiver im Team zusammenarbeiten zu können, haben wir von Anfang an unseren Programmcode mit GitHub [8] verbunden. Diese Integration ermöglichte es uns, gemeinsam an dem Projekt zu arbeiten und Änderungen nachzuverfolgen.

Bei der Erstellung der Struktogramme haben wir den Open-Source-Editor Struktog [9] genutzt, der von der TU Dresden entwickelt wurde. In Struktog lassen sich Diagramme nicht nur als Bild speichern, sondern auch als JSON-Datei. Dadurch konnten die Struktogramme im späteren Verlauf des Projekts problemlos angepasst werden, was ein großer Vorteil war.

5 Programmtest

5.1 Testpläne

Nr.	Bezeichnung	Nr.	Durchführung	Erwartetes Ergebnis	X
1	Start der Software	1	Software starten	Software startet	X
2	Startmenüs	1	Drücken der Enter Taste im Startbildschirm	Startmenü 1 öffnet sich	X
		2	Im Startmenü 1 die Enter/Leertaste drücken	Startmenü 1 schließt sich und Startmenü 2 öffnet sich	X

		3	Im Startmenü 2 die Enter/Leertaste drücken	Startmenü 2 schließt sich und Startmenü 3 öffnet sich	X
		4	Startmenü 3 öffnet sich	Startmenü 3 fordert Spieler auf einen Namen einzugeben	X
		5	Eingabe des Namens im Startmenü 3	Eingegebener Name wird Textfeld angezeigt	X
		6	Kein Name wird eingegeben und Enter/Leertaste wird gedrückt	Spieler wird automatisch „PLAYER 1“ genannt	X
		7	Drücken der Enter Taste	Name wird gespeichert und „Continue“ links oben blau hinterlegt	X
		8	Drücken der Enter Taste im Startmenü 3	Startmenü 3 wird geschlossen und Startmenü 4/ Hauptmenü öffnet sich	X
		9	Menüpunkt „Play“ ist im Startmenü 4/ Hauptmenü vorausgewählt	Play wird in Blau angezeigt	X
3	Spielstart	1	Drücken der Enter Taste mit vorausgewähltem Menüpunkt „Play“	Spiel wird gestartet	X
4	Bewegung Spielfigur	1	Spiel starten ohne drücken jeglicher Tasten	Spielfigur läuft unverändert auf derselben Stelle weiter	X
		2	Kurzes Drücken der Leertaste (ca. 1s)	Spielfigur „springt“ einmalig um 5 Zeilen nach oben	X
		3	Loslassen der Leertaste nach dem Sprung	Spielfigur landet und läuft weiter	X
		4	Gedrückt halten der Leertaste über	Spielfigur „springt“ bei Bodenkontakt (0,1 Zeile vor dem Boden) erneut	X

			beliebigen Zeitraum >1s	und wiederholt dies so lang bis Leertaste losgelassen wird	
5	Hindernisse	1	Spiel starten	Hindernisse (maximal 2 Spieleinheiten 2 breit und hoch) erscheinen zufällig am rechten Spielfeldrand	X
		2	Spiel starten/laufen lassen	Erschienenene Hindernisse bewegen sich von rechts nach links auf Spielfigur zu	X
		3	Spiel starten/laufen lassen	Überwundene Hindernisse verschwinden hinter dem linken Spielfeldrand	X
		4	Spiel lang laufen lassen	Hindernisdichte nimmt zu (Maximal 50 Hindernisse auf der Spieloberfläche)	X
		5	Spiel lang laufen lassen	Hindernisgeschwindigkeit nimmt zu	X
		6	Spiel laufen lassen	Hindernisse werden ausschließlich am rechten Spielfeldrand erzeugt	X
		7	Drücken der Leertaste vor einem Hindernis	Überspringen des Hindernisses	X
6	Kollisionserkennung	1	Spiel laufen lassen ohne drücken jeglicher Tasten	Spielfigur kollidiert mit Hindernis (nur wenn Spielfigur am Boden ist, während sich ein Hindernis in Spalte 6 befindet) → Hindernis das Kollision verursacht hat wird rot markiert	X
		2	Spiel laufen lassen ohne drücken jeglicher Tasten	Nach Kollision wechselt Spiel in Game-Over Bildschirm/Hauptmenü	X

7	Hauptmenü	1	Drücken der Pfeiltaste nach unten	Menüpunkt unterhalb derzeit ausgewähltem Menüpunkt wird ausgewählt und in blau angezeigt	X
		2	Drücken der Pfeiltaste nach unten	Menüpunkt oberhalb derzeit ausgewähltem Menüpunkt wird ausgewählt und in blau angezeigt	X
		3	Menüpunkt „Play Again“ auswählen und mit Enter/Leertaste bestätigen	Spiel wird erneut gestartet	X
		4	Menüpunkt „High Scores“ auswählen und mit Enter/Leertaste bestätigen (mit Internet)	High-Score Liste öffnet sich und zeigt die bisher höchst erzielten Ergebnisse (sofern man mit dem Internet verbunden ist)	X
		5	Menüpunkt „High Scores“ auswählen und mit Enter/Leertaste bestätigen (ohne Internet)	High Score Listen Menü öffnet sich und zeigt Fehlermeldung „- UNABLE TO CONNECT TO DATABASE -“, an	X
		6	In High-Score Liste Enter/Leertaste drücken	High-Score Liste wird geschlossen und es öffnet sich das Hauptmenü	X
		7	Menüpunkt „Change Name“ auswählen	Menüpunkt wird in blu angezeigt	X
		8	Menüpunkt Change Name mit Enter/Leertaste bestätigen	Change Name Menü wird aufgerufen	X
		9	Aufrufen des Change Name Menüs	Bisher verwendeter Name erscheint im Textfeld	X

		10	Eingeben des neuen Namens	Alter Name kann durch einfache Eingabe des neuen Namens überschrieben werden	X
		11	Drücken der Rücktaste im Change Game Menü	Einzelnes Zeichen des bisherigen Namens wird entfernt und der Cursor bewegt sich um eine Spalte nach links	X
		12	Erneutes Drücken der Rücktaste im Change Game Menü	Weiteres Zeichen des bisherigen Namens wird entfernt und Cursor bewegt sich um eine Spalte nach Links	X
		13	Drücken der Entfernen Taste	Der ausgewählte Buchstabe wird entfernt und der Cursor bleibt an der gleichen Stelle	X
		14	Drücken der Enter/Leertaste	Name wird gespeichert, das Change Name Menü beendet und das Hauptmenü geöffnet	X
		15	Bestätigen des leeren Textfeldes im Change Name Menü, ohne einen neuen Namen eingegeben zu haben	Ungültige Eingabe → alter Name wird weiterhin verwendet	X
		16	Auswählen des Menüpunktes „Music Enabled/Disabled“	Menüpunkt wird in blau angezeigt	X
		17	Drücken der Enter/Leertaste bei ausgewähltem Menüpunkt „Music Enabled“	Spielhintergrundmusik wird ausgeschaltet und Menüpunkt wechselt von „Music Enabled“ zu „Music Disabled“	X
		18	Drücken der Enter/Leertaste bei ausgewähltem	Spielhintergrundmusik wird eingeschaltet und Menüpunkt wechselt von	X

			Menüpunkt „Music Disabled“	„Music Disabled“ zu „Music Enabled“	
		19	Auswählen des Menüpunktes „Credits“	Ausgewählter Menüpunkt wird in blau angezeigt	X
		20	Drücken der Enter/Leertaste bei ausgewähltem Menüpunkt „Credits“	„Credits Menü“ wird aufgerufen und zeigt Ersteller und Mitwirkende Personen	X
		21	Drücken der Enter/Leertaste	„Credits Menü“ wird geschlossen und Hauptmenü aufgerufen	X
8	Sounds	1	Aufrufen der Software	Ein kurzer Sound wird beim Starten des Spiels abgespielt	X
		2	Aufrufen der Software	Die Spielmusik beginnt direkt im Anschluss zum Startsound zu spielen und läuft dauerhaft weiter	X
		3	Drücken der Leertaste im Spielverlauf	Ein spezifisch auf den Sprung abgestimmter Sound wird abgespielt	X
		4	Überspringen von 10 Hindernissen	Ein Sound wird alle 10 „Scorepunkte“ zusätzlich zum Sprung-Sound abgespielt	X
		5	Kollision mit einem Hindernis	Verzerrter Sound wird abgespielt	X
9	Sonstige Funktionen	1	Spiel starten	Score wird in der oberen linken Ecke zum Spielbeginn als „0“ angezeigt	X
		2	Spiel starten	Timer wird in der linken oberen Ecke unterhalb des Scores zum Spielbeginn als „0“ angezeigt und zählt die Sekunden hoch	X

		3	Spiel starten	Anzahl der „FPS“ wird angezeigt	X
		4	Überwinden eines Hindernis	Der Score wird um eins erhöht	X
		5	Spiel laufen lassen	Der Timer zählt die Sekunden aufwärts	X
		6	Überwinden mehrerer Hindernisse im Spielverlauf	Der Score zählt aufwärts hoch	X
		7	Game Over durch Kollision mit einem Hindernis	Timer und Highscore hören auf zu zählen	X
		8	Spiel starten	Im oberen Bereich des Spielfeldes wird eine zufällige Anzahl an Wolken eingebendet	X
		9	Spiel laufen lassen	Neue Wolken erscheinen zufällig auf der rechten Bildschirmseite	X
		10	Spiel laufen lassen	Wolken bewegen sich im Verlauf des Spiels nach links und verschwinden aus dem linken Spielfeldrand	X

5.2 Testergebnis

Die Testergebnisse haben wertvolle Erkenntnisse geliefert und Bereiche aufgezeigt, in denen weitere Verbesserungen vorgenommen werden können. Das Programm läuft trotz seiner Komplexität sehr stabil, insbesondere in Bezug auf die Tastatureingabe. Die Hinderniserkennung funktioniert in der Regel zuverlässig, könnte jedoch durch zusätzliche Feinabstimmungen weiter verbessert werden, insbesondere bei der Erkennung von Hindernissen von oben. Außerdem bietet die Interaktion mit dem Konsolenfenster, Optimierungsspielraum. Die Anpassung der Fenstergröße kann zu visuellen Unstimmigkeiten führen, und nach minimieren und maximieren des Fensters wird der Cursor sichtbar. Des Weiteren können bei der Ausführung des Programms als eigenständige Executable außerhalb der IDE, Anzeigeprobleme auftreten. Die Renderfähigkeiten des Programms sind im Allgemeinen zufriedenstellend, können jedoch bei schnellen Bewegungen, zu Anzeigefehlern führen. Darüber hinaus kann es auf bestimmten Geräten Einschränkungen bei der parallelen Soundwiedergabe geben, und die Wiedergabeflüssigkeit des Spiels kann durch unzureichende Rechenressourcen Verzögerungen erfahren.

6 Fazit und Ausblick

Abschließend kann man sagen, dass die Zusammenarbeit in der Gruppe gut funktioniert hat. Wir hatten Spaß bei der Ausarbeitung und sind mit dem Resultat dieser Projektarbeit sehr zufrieden.

Obwohl wir unsere Ziele mit Visual Basic erreicht haben, wäre es interessant gewesen, die Möglichkeiten von fortschrittlicheren Programmierumgebungen wie Unity oder Unreal Engine zu erkunden. Mit solchen Game Engines lassen sich oft in kürzester Zeit anspruchsvolle Spiele entwickeln, da sie arbeitsintensive

Aufgaben übernehmen und auch dem Leihen das Erstellen komplexer Anwendungen ermöglichen.

Mit Blick in die Zukunft eröffnen sich zahlreiche Möglichkeiten, um das Spiel weiter auszubauen. Es ist naheliegend, die in den Testergebnissen aufgeführten Probleme zu beheben und somit die Stabilität und Zuverlässigkeit des Spiels weiter zu verbessern. Zusätzlich könnte man Features wie ein Münzsystem einführen, bei dem die Spieler durch das Sammeln von Coins, Upgrades und Power-Ups erwerben können. Oder die Implementierung von benutzerspezifischen Profilen, welche den Spielern ermöglichen würden, ihren Spielfortschritt zu speichern und personalisierte Einstellungen beizubehalten.

Zum Schluss bleibt nur noch zu sagen: Schnür' deine Laufschuhe und renne, als gäbe es kein Morgen! Hol dir den Highscore und lass die Hindernisse alt aussehen.

7 Literaturverzeichnis

- [1] Wikipedia, „Wikipedia,“ 22 Mai 2023. [Online]. Available: <https://en.wikipedia.org/wiki/Running>. [Zugriff am 24 Mai 2023].
- [2] Wikipedia, „Wikipedia,“ 2 Oktober 2021. [Online]. Available: <https://de.wikipedia.org/wiki/Springen>. [Zugriff am 23 Mai 2023].
- [3] Microsoft, „Microsoft,“ [Online]. Available: <https://learn.microsoft.com/en-us/dotnet/visual-basic/>. [Zugriff am 24 Mai 2023].
- [4] Y. K. Wah, „Codeproject,“ 4 Juni 2014. [Online]. Available: <https://www.codeproject.com/Articles/781792/Fun-with-Sound>. [Zugriff am 8 Juni 2023].
- [5] lucasratske, „<https://github.com/ziyasal/FireSharp>,“ 26 Januar 2023. [Online]. Available: <https://github.com/ziyasal/FireSharp>. [Zugriff am 8 Juni 2023].
- [6] H. Raza, „Youtube,“ 9 Januar 2020. [Online]. Available: <https://youtu.be/mK2MErhxIO0>. [Zugriff am 7 Juni 2023].
- [7] Microsoft, „Visualstudio,“ [Online]. Available: <https://visualstudio.microsoft.com/>. [Zugriff am 19 April 2023].
- [8] GitHub, „GitHub,“ [Online]. Available: <https://desktop.github.com/>. [Zugriff am 19 April 2023].
- [9] T. Leonhardt, „GitLab,“ 23 Mai 2023. [Online]. Available: <https://gitlab.com/ddi-tu-dresden/cs-school-tools/struktog>. [Zugriff am 9 Juni 2023].