
TECHNICAL DESIGN DOCUMENT



Team: Lena Spitz (Teamleiterin), Lars Haider, Mareen Allgaier

Project Overview

Team

Das Spiel Volamus wird von einem Team von drei Personen entwickelt: Lena Spitz (Teamleiterin), Lars Haider und Mareen Allgaier

Game Concept

Einige Gelehrte der Acagamics haben sich das Ziel gesetzt die legendäre goldene Wolke zu finden, um sie zu studieren. Für dieses gewagte Unternehmen suchen sie aus der Tierwelt die Besten der Besten. Diese sollen mithilfe eines Ballspielturniers ermittelt werden.

In dem Spiel spielen zwei Spieler gegeneinander. Dabei spielen sie die jeweiligen Charaktere des Turniers.

Technical Goals

Ziel ist, dass das fertige Spiel eine vollständige und ästhetische 3D Graphik enthält. Auch während dem Spielen sollen die Animationen der Bewegungen und Schläge flüssig und passend sein. Die Motivation der Spieler wird währenddessen sowohl durch visuelle als auch auditive Feedbacks gesteigert.

Um den Spielern das Treffen des Balls zu vereinfachen, ist am Boden der Schatten des Balls zu sehen. Wichtig beim Schlagen ist, dass es gelingt, eine realistische Flugbahn des Balls zu simulieren, indem die Gravitation und Begrenzungen berücksichtigt werden.

Weiterhin muss das Spiel ein selbst-updatendes Interface beinhalten.

Technical Risks

Da kein Teammitglied über Vorwissen in der Spieleprogrammierung verfügt, können während der Entwicklung häufiger Schwierigkeiten auftreten.

Schwierig wird es, die technischen Abläufe genau zu simulieren, sodass das Spiel realistisch und korrekt läuft. Dazu zählen insbesondere die Kollisionen der Spielcharaktere mit dem Ball, sowie richtig berechnete Flugbahnen des Balls. Außerdem muss darauf geachtet werden, dass die Bewegungen richtig skaliert sind. Hier ist es wichtig den richtigen Mittelweg zwischen zu schnell und zu langsam zu finden, aber auch die Schrittgröße passend einzustellen.

Schwierigkeiten bei der Graphik und dem Feedback werden vor allem bei den Animationen (insbesondere des Wassers) auftreten und bei der zeitlich genauen Zuweisung des entsprechenden Sounds zu den jeweiligen (Bewegungs-)Animationen.

Development Environment

Die Implementierung des Spiels findet in der Entwicklungsumgebung Visual Studio mit dem Framework Monogame statt. Es wird sowohl Windows 7 als auch Windows 10 genutzt.

System

Das Spiel soll eine Windows Applikation werden.

Da es mit Monogame als Framework programmiert wird und im Rahmen des Projekts nicht zu umfangreich wird, gelten als Systemanforderungen die Anforderungen von Monogame. Diese sind DirectX11 oder höher beziehungsweise OpenGL 3.0, Windows XP oder eine spätere Windows Version, mindestens Visual C# 2010 und eine Grafikkarte.

Game Play

Game Structure/Game Objects

Im Endspiel sollen die Spieler die Möglichkeit haben, bis zu drei verschiedene Charaktere zu spielen. Der Fokus liegt zunächst bei der Implementierung des Pinguins, der über den größten Schlagradius verfügt. Weiterhin können dann ein schneller Delfin und eine Hummel, die am höchsten springen kann, hinzugefügt werden.

Weitere Personen, wie der Schiedsrichter und die Zuschauer, werden als Acagamics-Männchen dargestellt.

Zusätzlich werden Modelle für den Ball, das Spielfeld und seine Begrenzung, sowie das Netz benötigt. Da das Spiel mit zusätzlichen Features ausgestattet werden soll, werden weitere Modelle benötigt. Eine Fahne wird den Spielern immer die aktuelle Windrichtung anzeigen. Pfeilartige Modelle dienen als Gegenstände, die von den Spielern eingesammelt werden können.

Physics

Bei der Implementierung des Spiels müssen drei physikalische Eigenschaften berücksichtigt werden.

Die Gravitation spielt bei der Flugbahn des Balls eine große Rolle. Hierfür wird die Wurfparabel, jedoch mit Vernachlässigung des Luftwiderstandes, verwendet. Auch beim Springen der Charaktere muss Gravitation beachtet werden. Hierbei wird das Springen auf einen senkrechten Sprung auf der Stelle beschränkt.

Laut des physikalischen Gesetzes, kann an einer Stelle, an der ein Körper ist, kein zweiter sein. Deshalb gibt es einige Kollisionen, die beachtet werden müssen. Für das Spiel relevant sind jegliche Kollisionen der Spieler mit Ball, Feld und Netz. Der Ball wiederum kann mit Spielern, dem Netz, und dem Boden kollidieren.

Zusätzlich haben die jeweils unterschiedlichen Charaktere und der Ball individuelle Geschwindigkeiten, die so aufeinander abgestimmt werden müssen, dass der Spaß an dem Spiel langfristig erhalten bleibt.

Player Actions

Beide Spieler haben die Möglichkeit ihren Charakter nach rechts-links und vorne-hinten zu bewegen. Darüber hinaus haben sie die Fähigkeit zu springen. Zur Vereinfachung der Steuerung ist für einen der Spieler die Tastatur und für den anderen Spieler ein Controller vorgesehen.

Der Spieler, der seinen Charakter per Tastatur steuert, kann über W/A/S/D laufen und über Space springen. Der Spieler mit dem Controller steuert seinen Charakter über den linken Stick und springt mit A (bei Xbox Controllern) oder X (bei Playstation Controllern).

Den Spielern stehen zwei unterschiedliche Schläge zu Verfügung, die über die Tastatur mit Q und E ausgeführt werden. Soll der Ball leicht geschlagen werden erfolgt dies über Q, während über E ein starker Schlag ausgeführt wird. Die Richtung der Flugbahn wird entweder über die Pfeiltasten gesteuert. Mit dem Controller kann mit dem Left Bumper LB (LB) leicht und mit dem Right Bumper (RB) stark geschlagen werden. Die Richtungssteuerung des Schlagens erfolgt über den rechten Stick. Um beiden Spielern die Steuerung der Flugrichtung zu erleichtern, gibt es bei beiden Charakteren ein Vektorpfeil, der diese anzeigt.

Victory Conditions

Startet man das Spiel Volamus zum ersten Mal kann zunächst nur die Eiswelt gespielt werden. Vor dem Matchbeginn können die zu erreichende Punkteanzahl und die zusätzlichen Features eingestellt werden.

Wird es später mehrere Welten und einen Story Modus geben, müssen die Welten nacheinander erfolgreich absolviert werden. Bei erfolgreichem Abschluss einer Welt werden diese und der dazugehörige Charakter freigeschaltet.

Sind alle drei Welten freigeschaltet, ist der Story Modus abgeschlossen und es erscheint ein Endscreen.

Während eines Matches müssen beide Spieler versuchen, Punkte zu erreichen. Ein Spieler erhält einen Punkt, wenn der Ball den Boden im gegnerischen Spielfeld berührt, der Gegner den Ball aus dem Feld raus schlägt oder der Gegner mehr als drei Ballkontakte direkt nacheinander hatte.

User Interface

Game Menus

Wird das Spiel gestartet, muss ein Spieler Enter auf der Tastatur oder A auf dem Controller drücken, um den Startbildschirm zu verlassen. Anschließend wird die Geschichte des Spiels eingeblendet. Diese kann übersprungen werden, wodurch man direkt in das Hauptmenü gelangt.

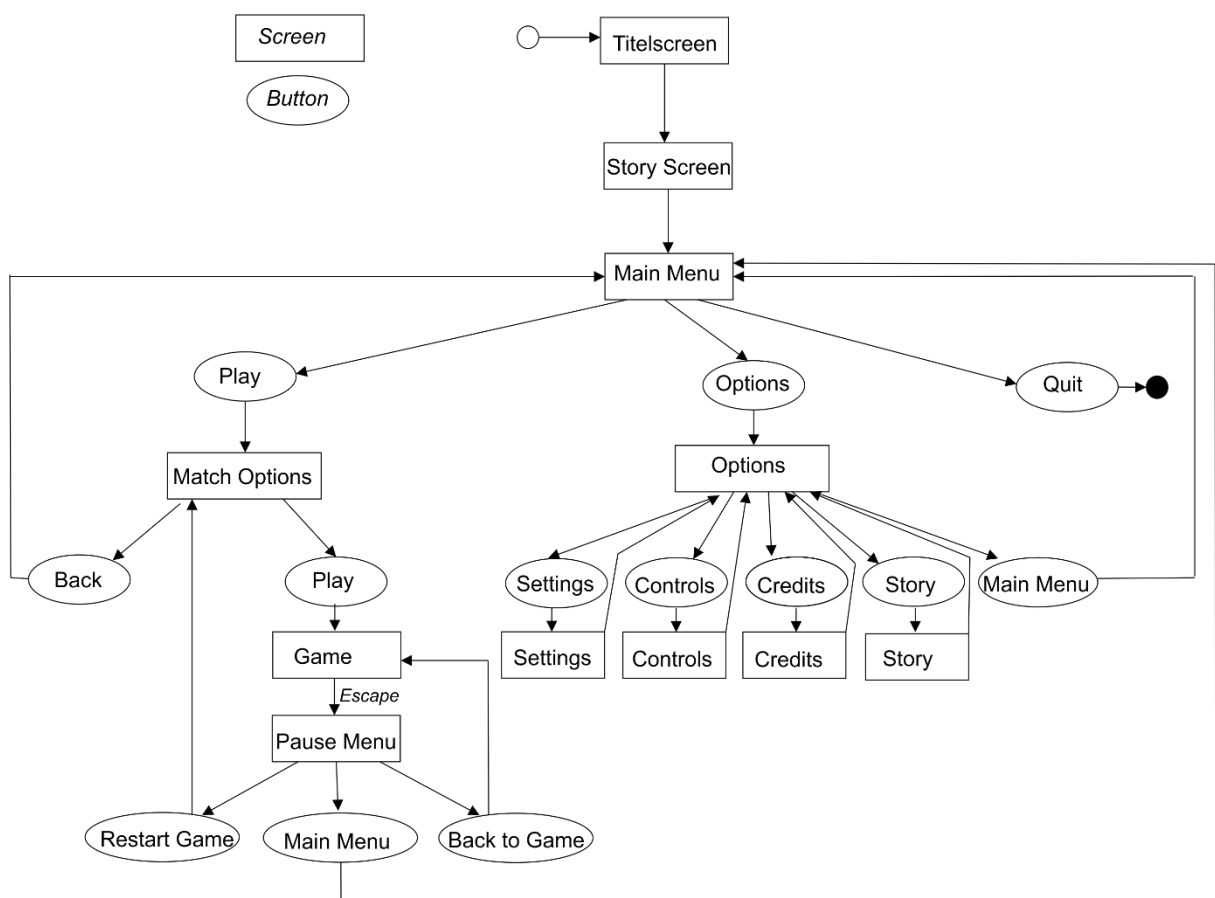
Im Hauptmenü gibt es drei Möglichkeiten: Das Spiel zu verlassen, ein Spiel zu beginnen oder auf die Optionen zu gehen.

In den Optionen kann die Geschichte noch einmal angeschaut werden. Unter „Credits“ befinden sich die Quellen der verwendeten Töne und Texturen, sowie die Angabe der Entwickler. Bei „Settings“ kann die Musiklautstärke und der Vollbildmodus ein- beziehungsweise ausgeschaltet werden. „Controls“ bietet den Spielern die Möglichkeit, die Tastenbelegung für die Steuerung mit der Tastatur und dem Controller nachzuschauen.

Über „Main Menu“ gelangt man zurück in das Hauptmenü. Startet man von dort ein Spiel, können Matchoptionen eingestellt werden. Dazu zählen die zu erreichenden Punkte, der Wind und die einzusammelnden Gegenstände.

Das Spiel kann durch Drücken der „Escape“-Taste angehalten werden. Jederzeit kann man zurück in das aktuelle Spiel gelangen, alternativ ein neues Spiel starten oder auch das Hauptmenü aufrufen.

Verknüpfungsübersicht der Menüs und Screens:



Graphics

Features

Die Modelle, zu denen überwiegend die Spielfiguren, das Feld, der Ball und das Netz zählen, werden mit der freien 3D Graphiksoftware „Blender“ modelliert. Um Animationen nachzuahmen müssen manche Modelle in einzelne Modelle zerlegt werden.

Eine wichtige Eigenschaft des Balls ist seine physikalisch korrekte Flugbahn. Weiterhin soll der Schatten des Balls auf dem Boden des Spielfelds dargestellt werden. Anhand dessen soll es den Spielern erleichtert werden, die Position des Balls abzuschätzen.

Das Logo und „Winner“ und „Loser“ Banner werden mit der Graphik-Software Corel Draw Essentials X5 erstellt.

View Modes

Da es sich bei Volamus um ein Zwei-Spieler-Spiel handelt, ist es wichtig, dass beide Spieler die gleichen Spielbedingungen haben, damit ein faires Match gewährleistet werden kann. Um beiden eine gleich gute Sicht auf das Geschehen zu ermöglichen, gibt es einen Split Screen. Die jeweilige Kamera ist hinter und schräg über (ca. 55°) dem jeweiligen Spieler platziert. So haben beide eine optimale Sicht auf ihren Spieler und den heranfliegenden Ball.

Animationen

Während dem Spiel sorgen Animationen für ein visuelles Feedback. Nach vorne rotierende Flügel dienen dabei als Schlaganimation. Wird der Spielcharakter bewegt, sorgt eine minimale Rotation nach links und rechts dafür, dass der Charakter einen watschelnden Eindruck macht. Auch der Ball ist durch eine Rotation um sich selbst animiert. Fahnen an der Mittellinie verdeutlichen dem Spieler den Einfluss des Windes. Dabei drehen sie sich in die jeweilige Windrichtung.

Audio

Für einen erhöhten Spielspaß ist es wichtig, viele verschiedene Feedbacks einzubauen. Da diese nicht nur visuell sein sollen, müssen verschiedene Geräusche und Musiken eingebaut werden. Die Musik, welche während eines Matches abgespielt wird, wird mit einer freien Musiksoftware komponiert.

Soundeffekte, wie das Jubeln der Zuschauer, das Pfeifen des Schiedsrichters, das Schlagen des Balls sowie die Tastenbetätigung im Menü werden aus einer freien Soundbibliothek ausgewählt.

Zur Einbindung der Sound wird Irrklang verwendet.

Data Management

Wird das Spiel um einen Story Modus erweitert, ist es nötig den jeweiligen Spielfortschritt, in der Collection zu speichern.

Für den Storymodus sollen Gewinne der einzelnen Spieler gespeichert und gegebenenfalls auch auf den Zwischen-Match-Screens angezeigt werden – es wird also angezeigt, ob Delfin 1 oder Delfin 2 gewonnen hat und sich auf die Suche nach der Goldenen Wolke machen wird.

Volamus wird über ein Pause System verfügen. In Anbetracht dessen, ist es auch hier wichtig die Positionen der Spieler und des Balls einzufrieren und die Zeit zu stoppen.

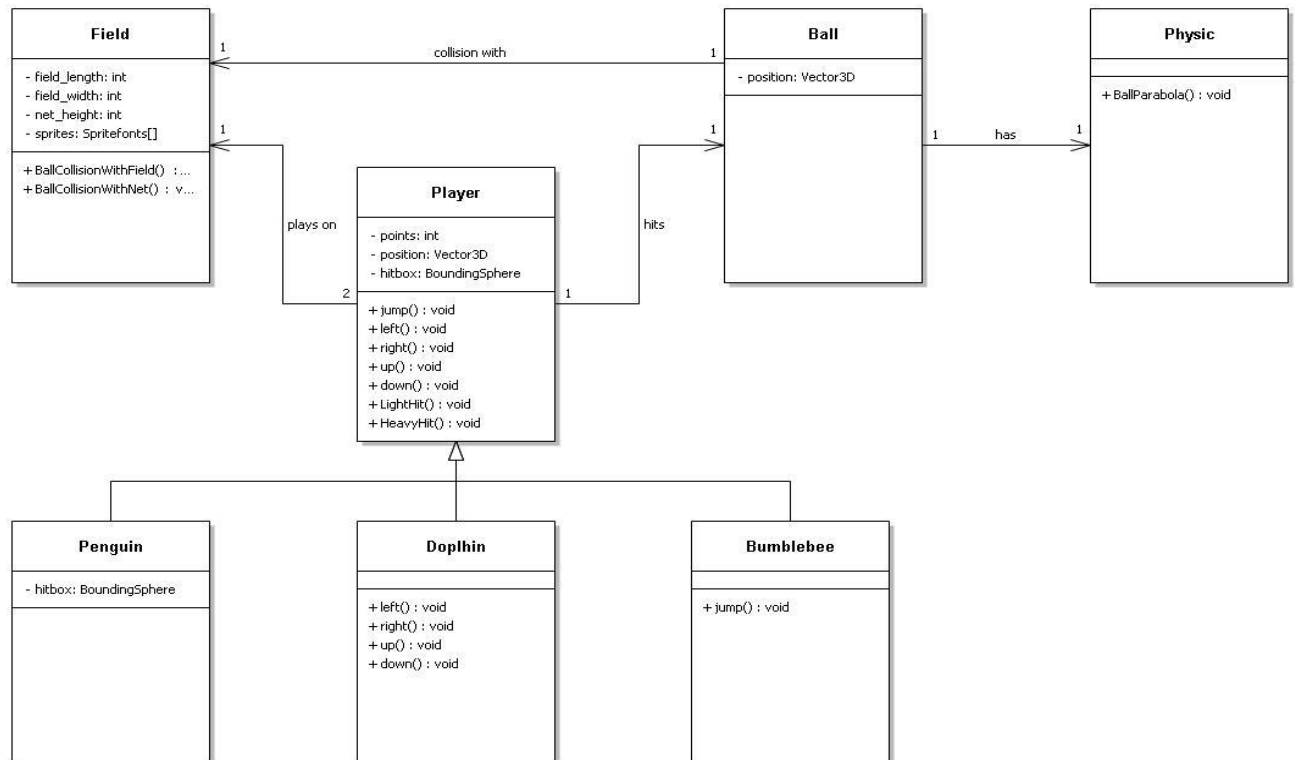
Klassen

Die Implementierung des gesamten Spiels wird in verschiedenen Klassen erfolgen.

Dabei gibt es voraussichtlich die Klassen Game, Spielfeld, Ball, Physics und die abstrakte Klasse Player von der Pinguin, Delfin und Hummel erben.

In der Klasse Game befindet sich der Standart Game loop mit den Funktionen initialize(), load content() und draw().

Die restlichen Klassen, deren Attribute, Funktionen und Unterteilung sowie grobe Zusammenhänge ist in folgendem Klassendiagramm zu sehen.



Milestones

Bis zum zweiten Meilenstein, der am 3.Juni sein wird, soll das Spiel über die groben Bewegungsabläufe und Handlungen verfügt. Das heißt es soll zwei Spieler geben, bei denen es sich aber noch nicht um fertige Modelle handeln muss, die sich innerhalb ihres Feldes bewegen können. Dabei steht auch noch nicht die Optik des Feldes oder gar der Umgebung im Vordergrund. Ziel ist es auch eine funktionierende Flugbahnberechnung des Balls zu haben und diese einzubauen. Kollisionen sollen so weit funktionieren, dass die grobe Spielhandlung zu erkennen ist. Weiterhin soll bereits ein kleines Menü existieren.

Für die Verwirklichung des zweiten Milestones, wurden die jeweiligen Teilziele grob auf die Teammitglieder verteilt:

Zum Meilenstein III am 01.07. sollen die Gamestates komplett implementiert sein und die Übergänge zwischen diesen funktionieren. Außerdem sollen Kollisionen und Physics flüssiger sein.

Je nach verbliebener Zeit besteht danach die Möglichkeit Animation und Grafik detailliert auszuarbeiten und das Spiel schöner zu gestalten. Dadurch werden diese erst gegen Ende des Projekts komplett fertig gestellt sein.

Das gesamte Projekt wird bis Mitte/Ende September abgeschlossen sein.

Voraussichtlicher Zeitplan und Aufteilung der Arbeitspakete:

	Meilenstein II (3.Juni)	Meilenstein III (1.Juli)	Semesterferien (spätestens ab 8.August)
Lena Spitz	Modelle, Kollision	Modelle, Optimierung	Modelle + Animation
Lars Haider	(Game States), Bewegungen, Kollision	Game States, Optimierung	Sounds
Mareen Allgaier	Flugbahn des Balls, Kollision	Physics, Optimierung	Menüdesign

Project Management

Um das Projekt erfolgreich abzuschließen, ist eine gute Planung erforderlich. Dabei steht bei uns die Kommunikation im Vordergrund. Es ist wichtig, dass jeder jederzeit erreichbar ist, um sicherzustellen, dass alle Mitglieder auf dem aktuellen Stand sind. Bei regelmäßigen Treffen können Probleme und Erfolge besprochen werden. Darüber hinaus dienen die Treffen dazu, das weitere Vorgehen zu planen und Arbeitspakete auf die einzelnen Teammitglieder zu verteilen.

Die Treffen werden auch dazu genutzt, die zu Beginn festgelegten Erfolgskriterien zu prüfen.