|  |
| --- |
|  |
| FlightSimulator |
| Universal Windows App mit Sensornutzung |

|  |
| --- |
| Patrick Müller  20.6.2016 |

Inhaltsverzeichnis

[Zielsetzung 2](#_Toc455974712)

[Sensoren 2](#_Toc455974713)

[Accelerometer 2](#_Toc455974714)

[Gyrometer 2](#_Toc455974715)

[Inclinometer 2](#_Toc455974716)

[OrientationSensor 2](#_Toc455974717)

[Fazit 2](#_Toc455974718)

[Sensoren Tests 3](#_Toc455974719)

[Accelerometer 3](#_Toc455974720)

[Inclinometer 3](#_Toc455974721)

[OrientationSensor 3](#_Toc455974722)

[Fazit 3](#_Toc455974723)

[Applikation 4](#_Toc455974724)

[Allgemeines 4](#_Toc455974725)

[Hauptmenü 4](#_Toc455974726)

[Im Flug 5](#_Toc455974727)

[Flughafen 6](#_Toc455974728)

[Einstellungen 7](#_Toc455974729)

[Sound 7](#_Toc455974730)

[DebugInfo 7](#_Toc455974731)

# Zielsetzung

Ziel dieses Projekts ist es, eine Applikation zu entwickeln, welche mit Hilfe der in Mobilgeräten verfügbaren Sensoren ein Flugzeug auf dem Bildschirm neigen bzw. drehen kann, während sich die Umgebung bewegt.

Hierfür werden zunächst die verfügbaren Sensoren aufgelistet und in Tests verglichen, um die beste Lösung für die Aufgabenstellung zu ermitteln. Anschließend wird ein Prototyp entwickelt in dem grundlegende Funktionen vorhanden sind.

# Sensoren

## Accelerometer

Das Accelerometer, auch Beschleunigungssensor, misst die auf ihn wirkende Beschleunigung in 3 Achsen. Dadurch lässt sich jede Bewegung des Gerätes nachvollziehen, z.B., wenn man es schüttelt. Außerdem kann durch die auf das Gerät wirkende Schwerkraft die Ausrichtung des Geräts im Dreidimensionalen Raum bestimmt werden.

## Gyrometer

Das Gyrometer misst die Winkelgeschwindigkeiten entlang der 3 Achsen des Gerätes. Mit Hilfe dieses Sensors kann bestimmt werde, wir schnell das Gerät in die jeweilige Richtung gedreht wird.

## Inclinometer

Das Inclinometer bestimmt den Schwenk-, Neige- und Rollwinkel des Gerätes. Diese Winkel werden aus den kombinierten Daten des Accelerometers, Gyrometers und Magnetometers errechnet. Mit Hilfe dieser Winkel lässt sich die relative Ausrichtung des Gerätes sehr genau bestimmen.

## OrientationSensor

Der OrientationSensor ist quasi eine extrem präzise Vorstufe des Inclinometers, der die Ausrichtung des Gerätes aus den Daten des Accelerometers, Gyrometers und Magnetometers errechnet und in Form einer Drehmatrix oder eines Quaternions ausgibt.

## Fazit

Am besten geeignet scheint das Inclinometer zu sein, da es speziell die Winkel der einzelnen Achsen des Gerätes ausgibt.

Aus den Daten des OrientationSensor lassen sich diese Winkel zwar auch berechnen, allerdings bedeutet das einen sehr großen Rechenaufwand, da man eine Rotationsmatrix in die entsprechenden Winkel umrechnen müsste.

Das Accelerometer scheint grundsätzlich für die Aufgabe geeignet zu sein, doch auch hier muss aus den Koordinaten mit Hilfe von trigonometrischen Funktionen der Winkel berechnet werden.

Das Gyrometer erscheint ungeeignet, da die Winkelgeschwindigkeit allein nicht ausreicht, um ein Objekt relativ zu Geräteausrichtung zu bewegen.

# Sensoren Tests

Alle Tests sind bisher nur auf einem Software-Emulator durchgeführt worden.

## Accelerometer

Das Accelerometer liefert Daten welche direkt die Orientierung des Gerätes widergeben, wodurch sich im Simulator eine sehr direkte Steuerung des Objektes ergibt.

Der größte Nachteil ist, dass man das Gerät tatsächlich um 90 Grad drehen muss um eine solche Neigung im Programm zu erzielen.

## Inclinometer

Da das Inclinometer auch die Winkelgeschwindigkeit mit einbezieht, ergibt sich eine indirektere Steuerung als mit dem Accelerometer. So lassen sich im Programm relativ starke Neigungen erzielen, während das Gerät nur leicht geneigt ist.

Durch die Verzögerte Wirkung auf das Objekt ergibt sich ein realistischeres Gefühl wie etwa bei einer Flugsimulation.

Leider kehrt sich das Vorzeichen des Rollwinkels um, sobald man das Gerät über bzw. unter den virtuellen Horizont neigt. Da dieser Vorgang mit extremen Winkelsprüngen einhergeht, ist das Inclinometer leider nicht verlässlich genug.

## OrientationSensor

Hierbei wird aus der vom Sensor ausgegebenen Rotationsmatrix der Neigungswinkel berechnet.

Vom Verhalten her erscheint der OrientationSensor fast identisch mit den Ergebnissen des Inclinometers, allerdings etwas langsamer bzw. schwächer. Möglicherweise wird das Gyrometer bei diesem Sensor anders Berücksichtigt als beim Inclinometer.

Auch hier muss das Gerät, wie beim Accelerometer, bis zu 90 Grad geneigt werden um im Programm den gleichen Winkel zu erzeugen.

## Fazit

Obwohl das Inclinometer am geeignetsten erschien, habe ich mich nun doch entschieden, das Accelerometer für die Applikation zu benutzen. Die Sprünge im Rollwinkel des Inclinometers konnte ich leider nicht so ausgleichen, dass ein gutes Spielgefühl entsteht.

Für das Accelerometer wäre zwar eine relativ starke Geräteneigung nötig, dies lässt sich allerdings durch Anpassung der Geschwindigkeitsfunktionen im Code ausgleichen. So ist es nicht nötig, das Gerät im 90° Winkel zu halten um die maximale Geschwindigkeit zu erreichen, es reicht schon ein Winkel von ca. 60° (dieser Wert ist im Code anpassbar falls nötig).

# Applikation

Nachfolgend werden die implementierten Features der Applikation dargestellt.

## Allgemeines

Zur Steuerung wird das Accelerometer des Gerätes benutzt.

Die Applikation ist in 3 Grundlegende Abschnitte aufgeteilt:

* Hauptmenü
* Flugmodus
* Bodenmodus/Flughafen

Ein Einstellungsmenü ist vom Hauptmenü und vom Bodenmodus aus erreichbar.

Die Assets der Applikation werden im ApplicationData-Folder des Gerätes gespeichert.

## Hauptmenü



Das Hauptmenü besteht aus einem Grid mit einer ImageBrush als Hintergrund. In einem StackPanel sind ein TextBlock für den Titel und eine weiteres StackPanel, welches 3 Buttons enthält.

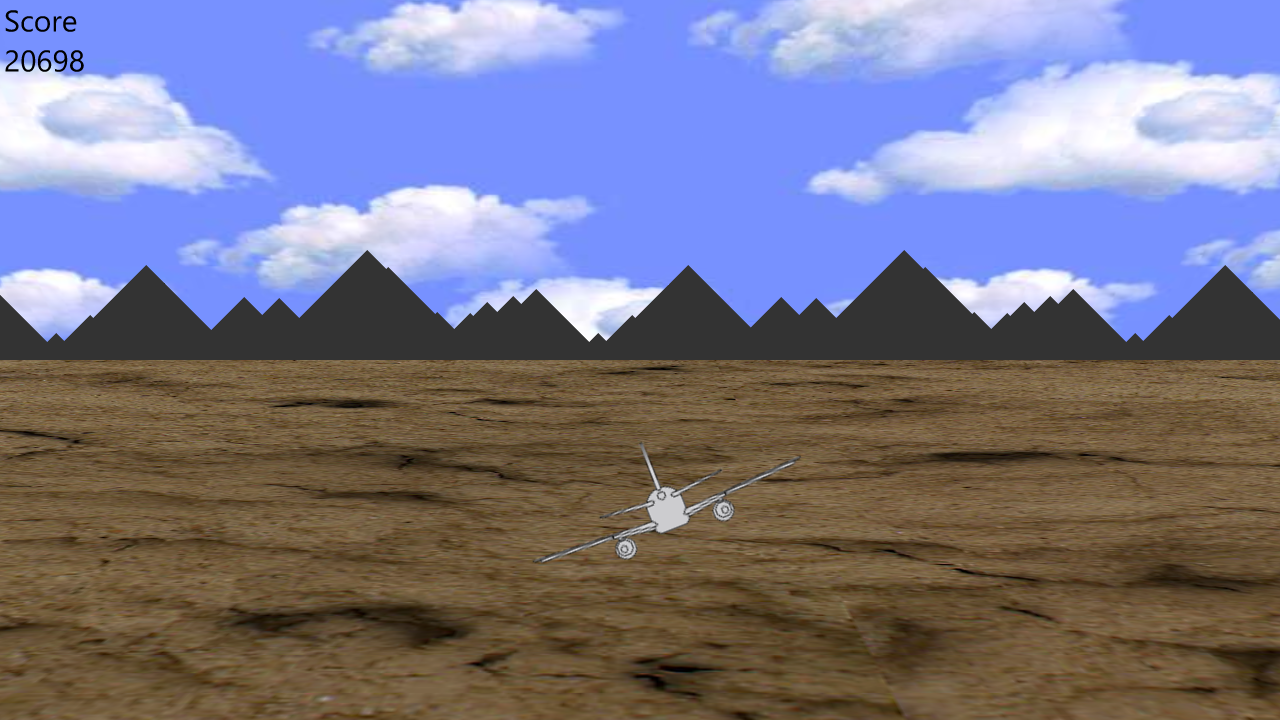
Als Hintergrundbild wird ein Graustufen-Screenshot aus dem Flugmodus verwendet.

Der „New Game“-Button startet ein neues Spiel und navigiert den User in den Flugmodus.

Der „Load Game“-Button versucht ein gespeichertes Spiel aus dem ApplicationData-Folder zu laden und navigiert den User dann in den Flugmodus im entsprechenden Spielzustand. Wenn kein gespeicherter Spielstand vorhanden ist, wird ein neues Spiel gestartet.

Der „Settings“-Button öffnet einen ContentDialog. In diesem Dialog ist ein StackPanel mit zwei CheckBoxen darin. Zum einen lassen sich die DebugInformation im Flugmodus ein- oder ausblenden, zum anderen lässt sich die Soundausgabe der Applikation aktivieren bzw. deaktivieren.

## Im Flug



Die Oberfläche des Flugmodus besteht im Wesentlichen aus 4 Elementen:

* Airplane
* GroundArea
* SkyArea
* MountainArea

Außerdem sind im oberen linken Eck eine Punkteanzeige und im oberen rechten Eck die Debug Informationen, falls diese in den Settings aktiviert sind.

Das Airplane ist ein Image das auf einen transparenten Canvas gezeichnet wird.

Die GroundArea ist ein transparenter Canvas, auf dem 4 identische Images gezeichnet werden, die bündig aneinander angeordnet sind. Der Canvas ist mit Hilfe einer PlaneProjection um 86° geneigt, so dass ein dreidimensionaler Eindruck entsteht.

Die SkyArea ist genauso wie die GroundArea aufgebaut, allerdings um einen schwächeren Winkel in die entgegengesetzte Richtung geneigt.

Auch die MountainArea ist ein transparenter Canvas auf dem zwei identische Gebirgszüge nebeneinander gezeichnet sind. Diese sind entlang des virtuellen Horizonts, an dem SkyArea und GroundArea aufeinandertreffen, angeordnet.

Alle 16 Millisekunden werden vom Accelerometer neue Werte ausgelesen und anhand dieser die vertikale und horizontale Geschwindigkeit des Flugzeugs berechnet. Mit Hilfe dieser Geschwindigkeiten werden die Images entsprechende bewegt. Die verschiedenen Ebenen werden in unterschiedlichen Geschwindigkeiten bewegt um einen Parallax-Effekt zu erzeugen und so den Eindruck von tiefe zu vermitteln. Wenn ein Image den sichtbaren Bereich verlässt, wird es auf die andere Seite des Bildschirms versetzt, um ein endloses Scrolling zu ermöglichen.

Durch Neigen des Gerätes nach vorne bzw. hinten, beschleunigt bzw. bremst man das Flugzeug. Wenn die Geschwindigkeit unter einen gewissen Schwellenwert fällt, stoppen die Flug-Animationen und ein FadeIn zu einem schwarzen Bild beginnt. Sobald der Bildschirm schwarz ist, beginnt ein FadeOut und der Flughafen wird angezeigt.

## Flughafen



Die Flughafenansicht besteht aus einem Grid mit einer ImageBrush als Hintergrund, das den aktuellen Ort darstellt. In einem StackPanel sind der Name des aktuellen Ortes als Textblock und ein weiteres StackPanel, das 3 Buttons enthält.

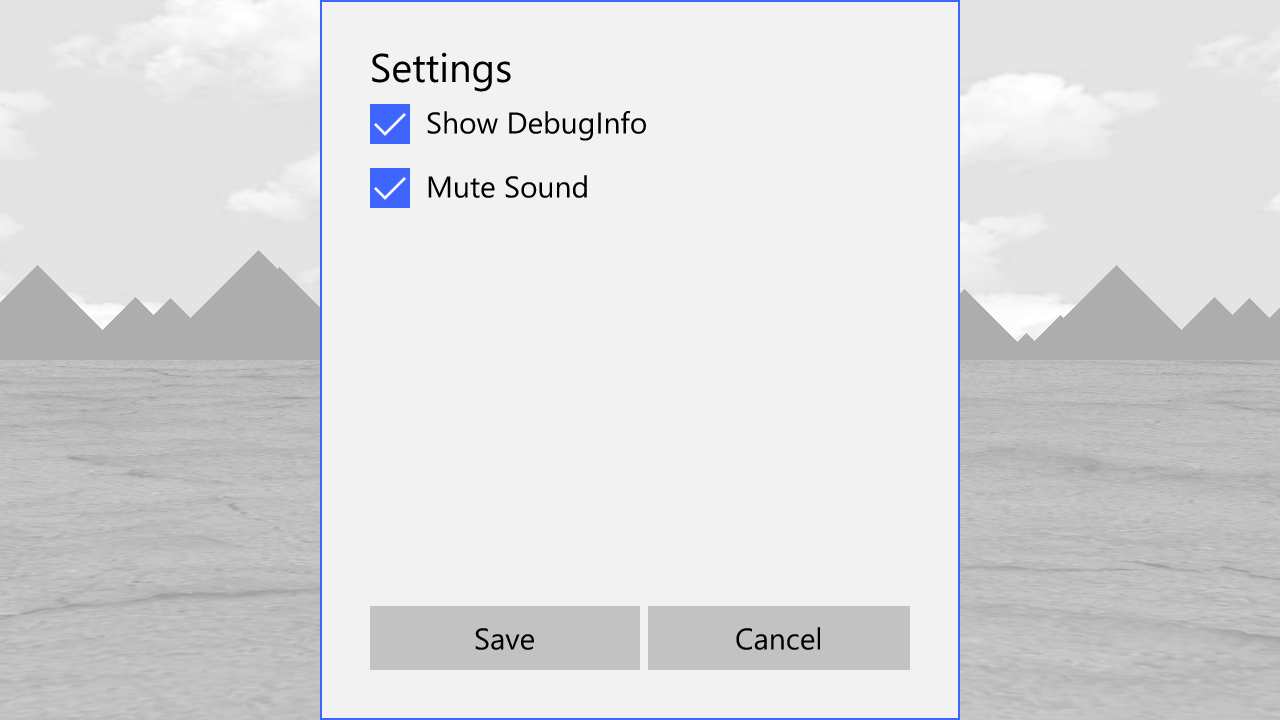
Mit Hilfe des „Lift Off!“-Button verlässt man den Flughafen und gelangt wieder in den Flugmodus.

Der „Save Game“-Button speichert den aktuellen Spielzustand in einer Datei auf dem Gerät.

Der „Settings“-Button öffnet, wie im Hauptmenü, einen ContentDialog, in dem die selben Einstellungen geändert werden können.

Verschiedene Orte sind in Form eines Dictionaries mit Ortsname als Key und ImagePath als Value gespeichert. Aus dieser Liste werden nacheinander die Orte geladen, wenn der Nutzer das Flugzeug landet und die entsprechenden Namen und Bilder Orte angezeigt.

## Einstellungen



Das Einstellungsmenü besteht aus einem ContentDialog und verfügt momentan über zwei CheckBoxen:

* Show DebugInfo; Zeigt die Debug Informationen im Flugmodus an
* Mute Sound; Stellt die Soundausgabe an bzw. aus

Klickt man den Save-Button, werden die Einstellungen übernommen und in einer Datei auf dem Gerät gespeichert. Bei klicken des Cancel-Buttons oder klicken des Hardwarebuttons Zurück, werden die vorherigen Einstellungen wiederhergestellt und keine Änderungen gespeichert.

## Sound

Mit Hilfe von MediaElement werden im Flugmodus Geräusche eines Düsentriebwerks und Hintergrundmusik abgespielt. Beim Landen werden diese Sounds angehalten und beim Anzeigen des Flughafens werden Umgebungsgeräusche eines Flughafens abgespielt.

Diese Sounds lassen sich über das Einstellungsmenü Stumm schalten.

## DebugInfo

Wenn in den Einstellungen die DebugInfos aktiviert sind, werden im Flugmodus im oberen, rechten Eck des Bildschirms zusätzliche Informationen angezeigt.

Zum einen werden die Winkel um die das Gerät geneigt ist angezeigt. Außerdem auch die Geschwindigkeiten in horizontaler und vertikaler Richtung und die relative Position des Flugzeugs zum Startpunkt in einem neuen Spiel.