Raytracing on Android

Kagerer Miriam

Baldinger Andreas

Shehata Abdu

08.05.2021

Version 1.0



# **1 Einleitung**

## **1.1 Zweck des Dokuments**

Das Dokument dient dazu den Verlaufs des Projekts zu dokumentieren und auf den Source Code genauer einzugehen. Weiters werden mit dem Dokument die Bedingungen, Ziele und Vorgaben festgelegt. Zusätzlich sollen die Endresultate, Messungen/Benchmarks, tatsächliche Kosten und tatsächliche Dauer zur Evaluierung enthalten sein.

## **1.2 Gültigkeit des Dokuments**

Der Rahmen dieses Dokuments umfasst das gesamte Projekt “Raytracing on Android”.

Änderungen können und werden von den drei Hauptverantwortlichen vorgenommen.

Die Aktualität des Dokuments ist anhand der Versionsnummer und des Versionsdatums prüfbar. Die aktuellste Version nimmt immer Bezug auf die aktuellste Version des Projekt-Codes und der Projekt-Dokumentation.

## **1.3 Begriffsbestimmungen und Abkürzungen**

OpenGL - Open Graphics Library - [Spezifikation](https://de.wikipedia.org/wiki/Spezifikation) einer plattform- und programmiersprachen übergreifenden [Programmierschnittstelle](https://de.wikipedia.org/wiki/Programmierschnittstelle) zur Entwicklung von 2D- und 3D-[Computergrafik Anwendungen](https://de.wikipedia.org/wiki/Computergrafik).

ES - Embedded Systems - Die Version von OpenGL die unter anderem für mobile Endgeräte entwickelt wurde und auf Android nutzbar ist.

VCS - Version Control System (Beispiel folgt)

Git - Eine [freie Software](https://de.wikipedia.org/wiki/Freie_Software) zur [verteilten Versionsverwaltung](https://de.wikipedia.org/wiki/Versionsverwaltung#Verteilte_Versionsverwaltung) von [Dateien](https://de.wikipedia.org/wiki/Datei)

FPS - Frames Per Second - Maß für die [Bildfrequenz](https://de.wikipedia.org/wiki/Bildfrequenz)

API - Application Programming Interface - Programmteil, der von einem [Softwaresystem](https://de.wikipedia.org/wiki/Software) anderen [Programmen](https://de.wikipedia.org/wiki/Computerprogramm) zur Anbindung an das System zur Verfügung gestellt wird.

GUI - Graphical User Interface - bezeichnet eine Form von [Benutzerschnittstelle](https://de.wikipedia.org/wiki/Benutzerschnittstelle) eines [Computers](https://de.wikipedia.org/wiki/Computer).

IDE - integrated development environment - eine Sammlung von Computerprogrammen, mit denen die Aufgaben der Softwareentwicklung möglichst ohne Medienbrüche bearbeitet werden können.

## **1.4 Zusammenhang mit anderen Dokumenten**

Es liegt weder ein Angebot, noch ein Lastenheft vor.

TODO - Lugstein fragen

## **1.5 Überblick über das Dokument**

Der folgende Teil des Dokumentes befasst sich mit den unten angeführten Punkten:

2) Allgemeinen Beschreibung der Produkts

3) Detaillierte Beschreibung der Produktmerkmale

4) Vorgaben an die Projektabwicklung

5) Verpflichtungen des Auftraggebers

6) Literaturverweise

7) Anhang

# **2 Allgemeine Beschreibung des Produkts**

Diese allgemeine Beschreibung soll ein „Management Summary“ sein, also noch keine spezifischen Produktmerkmale beschreiben.

## **2.1 Zusammenhang mit bereits laufenden Projekten**

Es gibt keine Zusammenhänge mit bereits laufenden Projekten.

TODO - Lugstein fragen

## **2.2 Zusammenhang mit Vorgänger- und Nachfolgeprojekten**

Es gibt keinen Zusammenhänge mit Vorgänger - oder Nachfolgeprojekten.

TODO - Lugstein fragen

## **2.3 Zweck des Produkts**

Der Zweck der App ist es zu zeigen, dass Raytracing auf Android implementiert werden kann und dass aktuelle, mobile Hardware performant genug ist Realtime-Raytracing zu betreiben.

Der entscheidende Vorteil gegenüber der etablierten Render-Methode “Rasterization” ist, dass sich mit Raytracing naturgetreue Lichtberechnungen durchführen lassen, die in einem realistischer aussehenden Bild resultieren.

In der Filmindustrie profitiert man schon lange von den Vorzügen die Raytracing mit sich bringt. In Realtime-Computer-Grafikanwendungen wie Spielen (Bilder können nicht zu einem früheren Zeitpunkt vor-gerendert werden) war dies aber mit der bisherigen Hardware-Leistung nicht möglich, auf mobilen Endgeräten nahezu undenkbar.

## **2.4 Abgrenzung und Einbettung des Produkts**

Festlegen der Systemgrenzen. Welche wesentlichen Leistungen sollen umfasst werden, was gehört nicht dazu, sondern zur Umgebung? Blockdiagramm, das zeigt, wie das zu erstellende Produkt mit anderen Systemen in Beziehung steht.

TODO - Lugstein fragen

## **2.5 Überblick über die geforderte Funktionalität**

Hier soll nur ein Überblick vermittelt werden – die detaillierte Beschreibung folgt in Kapitel 3

## **2.6 Allgemeine Einschränkungen**

Vorgaben hinsichtlich Schnittstellen, Standards, Methoden, die eingehalten werden müssen.

## **2.7 Vorgaben zu Hardware und Software**

Die App soll auf Android Geräten mit OpenGL ES 3.1 und aufwärts laufen. Die API wird ist erst ab Android 5 (API Level 21) verfügbar und läuft somit nicht auf älteren Geräten. Die App wird in der Entwicklungsumgebung Android Studio entwickelt. Als Version Control System wird git benutzt.

## **2.8 Benutzer des Produkts**

Die Zielgruppe des Projektes sind hauptsächlich Technikinteressierte, es benötigt aber keine besondere Ausbildung oder Vorkenntnisse, da es ein möglichst benutzerfreundliches Design haben soll.

# **3 Detaillierte Beschreibung der geforderten Produktmerkmale**

Hier werden die vom Auftraggeber geforderten und von der Entwicklung zugesagten Produkteigenschaften definiert.

Jedes geforderte Merkmal sollte

· eindeutig bezeichnet sein,

· bei Bedarf mit einer Priorität versehen werden und

· die Herkunft der Anforderung zeigen (auch Bezüge zum Lastenheft).

Die Beschreibung sollte so sein, dass

· möglichst geringer Interpretationsspielraum bleibt,

· eine Überprüfbarkeit gegenüber dem fertig gestellten Produkt gegeben ist.

## **3.1 Lieferumfang**

Produkt und dessen Teilkomponenten, Schulungsunterlagen, Handbücher, …

Das Produkt ist eine App und enthält somit keine Lieferbaren Produkte.

## **3.2 Abläufe (Szenarien) von Interaktionen mit der Umgebung**

Use Case Nummer: 1

Der Benutzer startet das Programm und steuert die Kamera der Szene mithilfe der in 3.4.1 beschriebenen Funktion.

## **3.3 Ziele des Benutzers**

In Use Case Nummer 1 kann der Benutzer unter Verwendung der in 3.4.1 beschriebenen Funktion die gesamte Szene überblicken. Er kann verdeckte Objekte sichtbar machen und die durch den Betrachtungswinkel verursachte Echtzeit-Veränderung des Lichteinfalls betrachten.

## **3.4 Geforderte Funktionen des Produkts**

TODO Lugstein

### Die App soll fähig sein, eine simple 3D-Szene darzustellen, welche durch Steuerung der Kamera betrachtet werden kann.

### 3.4.1 Kamerasteuerung

#### 3.4.1.1 Wirkungsweise der Kamera-Steuerung

Durch Streichen über den Bildschirm lässt sich die Kamera und damit die Ansicht/Szene rotieren.

#### 3.4.1.2 Abhängigkeiten / Randbedingungen

Die Geschwindigkeit der Kamera-Rotation wird in einer zentralen Kamera-Klasse einstellbar sein und kann sich zum Beispiel szenen-spezifisch ändern. Dies gilt auch für weitere in Verbindung stehende Kamera-Eigenschaften wie zum Beispiel “Field-of-View” und “View Frustum” (beschreiben Größe und Tiefe der Ansicht).

## **3.5 Externe Schnittstellen des Produkts**

Das Verhalten im Sinne einer Interaktion des zu erstellenden Produkts sollte bereits oben in Form von Szenarien beschrieben sein. Hier geht es um die Beschreibung, wie die konkreten Schnittstellen aussehen und funktionieren.

### 3.5.1 Benutzerschnittstellen (User Interfaces)

Wie sieht das GUI aus, Verweise auf Prototypen.

TODO Screenshot des fertigen GUI einfügen.

### 3.5.2 Systemschnittstellen

Beschreibung der SW- und HW-Schnittstellen, mit denen das zu erstellende System kommuniziert. Beschreibung der Kommunikationsart, des Datenformats. Verfeinerung des Überblicks in Kapitel 2.

#### 3.5.2.1 Kamera-Steuerung TODO

· Syntax/Semantik

· Datenformat

· Protokolle

· Datenraten

## **3.6 Sonstige geforderte Produktmerkmale**

Es folgt eine Beschreibung der nicht funktionalen Anforderungen:

### 3.6.1 Geschwindigkeitsmerkmale (performance)

· Reaktionszeiten/Inputlag (auf zum Beispiel die in 3.4.1 beschriebene Kamerasteuerung/Fingerbewegung)

· Anlaufzeiten (App-Start, Szenen-Ladezeit)

· Durchsatzrate/Bildwiederholrate (FPS, Logic-Updates/Sekunde)

### 3.6.2 Ressourcenmerkmale (resources)

Die folgenden Punkte fallen alle unter den Überpunkt bzw. die Resource “Smartphone)

· Datenmengen

· CPU-Bedarf

· CPU-Auslastung (möglicherweise Per-Core-Auslastung bei Multithreading)

· GPU-Bedarf

· GPU-Auslastung

· Speicher

· Arbeitsspeicher

### 3.6.3 Schutzmerkmale (security)

Schutz des Produkts gegen Eingriffe von außen (unberechtigter Zugriff, Virenschutz, …)

Es bestehen keine besonderen Notwendigkeiten von zusätzlichen Schutz des Produktes, weswegen die App nur die Standart Schutzfunktion von Android-Studio verwendet. ??

### 3.6.4 Sicherheitsmerkmale (safety)

Smartphones, die OpenGL nicht verwenden, können diese App nicht herunterladen, da dies im Manifest der App festgelegt wurde.

Beim Beenden der App werden sämtliche Resourcen wieder freigegeben, um nicht unnötig Speicherplatz zu verwenden.

### 3.6.5 Portabilitätsmerkmale (portability)

Die Portabilität des Produktes liegt in seiner Natur als Smartphone-App. Es ist allerdings auf Android begrenzt und nicht auf andere mobile Betriebssysteme übertragbar.

Die App kann einen Eintrag im Google Play Store erhalten, um dort als Download zur Verfügung gestellt zu werden bzw. auf diversen Websites erhältlich sein.

### 3.6.6 Zuverlässigkeit (reliability)

TODO Lugstein

Maßzahlen z. B. Ausfallszeiten in Minuten / Jahr und MTBF (Mean Time Between Failure)

### 3.6.7 Wartungsmerkmale (maintenance)

TODO Lugstein

Angaben zur Wartungsfreundlichkeit, welche Tools können verwendet werden?

### 3.6.8 Wiederverwendbarkeitsmerkmale (reuse)

Ist eine spätere Wiederverwendung von Produktteilen gewünscht, werden die Anforderungen daran festgelegt.

### 3.6.9 Benutzbarkeitsmerkmale (usability)

TODO Lugstein

Z. B. Angabe, wie viele Bedienungsfehler nach einer vierstündigen Einschulung bei einem Usability-Test noch gemacht werden dürfen.

Aufgrund des benutzerfreundlichen Designs dürften keine Bedienungsfehler auftreten.

# **4 Vorgaben an die Projektabwicklung**

## **4.1 Anforderungen an die Realisierung**

· Hardware

* Smartphone

· Software

* Android Betriebssystem
* OpenGL
* Java als Programmiersprache

## **4.2 Fertige und zugekaufte Komponenten**

Beschreiben der Komponenten, die zugekauft werden oder schon fertig vorhanden sind.

Für die Test sind bereits private PCs vorhanden auf denen gratis Software (Android Studio als IDE) und von der FH zur Verfügung gestellte Software (Microsoft Office) installiert wurde.

Es sind außerdem private Smartphones vorhanden, die für die Applikations-Testungen auf echten Endgeräten verwendet werden. Dies ist manchmal notwendig weil der in Android Studio integrierte Emulator OpenGL ES aber Version 3 nicht unterstützt.

## **4.3 Unterauftragnehmer**

Es sind keine Unterauftragnehmer vorhanden.

## **4.4 Abnahmebedingungen** !!!-----------------------------------------------------------

· Rahmenbedingungen TODO Lugstein

o Wogegen wird abgenommen (Pflichtenheft)?

* Die Projektabgabe erfolgt über GIT und wird von einem Professor bzw. dem Betreuer durchgeführt.
* Die Testdaten werden von den Entwicklern selbst bereitgestellt.

o Wer unterzeichnet das Abnahmeprotokoll.

· Abnahmekriterien

o Festlegung des Abnahmetests

o Wann ist die Abnahme erfolgreich (Resfehlerquote)

o Werden die nicht funktionalen Anforderungen erfüllt?

· Abnahmenunterlagen

o Testprotokolle

· Gutachten, Sicherheitsnachweise

o Sind Gutachten oder Sicherheitsnachweise beizubringen?

· Erfüllung von Vorschriften und Normen

o Hält das Produkt die vorgeschriebenen Normen und Vorschriften ein?

## **4.5 Lieferbedingungen**

· Wann werden die einzelnen Komponenten geliefert?

· Wie wird geliefert (elektronisch, auf CD, Source Code, nur Executable)?

## **4.6 Gewährleistung**

Vereinbarung über die Gewährleistungsdauer, Umfang der Gewährleistung, das Fehlermeldungsverfahren.

# **5 Verpflichtungen des Auftraggebers**

Auflistung, wozu der Auftraggeber im Rahmen des Projekts verpflichtet ist, z. B.

· Bereitstellung von HW

· Bereitstellung von SW

· Zur Verfügung stellen von Dokumentation

· Schulungsmaßnahmen

· Zuständigkeiten und Ansprechpartner bekannt geben

· Testdaten zur Verfügung stellen, Zugang zu Testanlagen ermöglichen

· Vorgehen, falls Änderungen gewünscht sind

· …

# **6 Literaturverweise**

# **7 Anhang**

Falls erforderlich, können hier Materialien eingefügt werden, die nicht in die hier definierte Gesamtstruktur passen und dennoch zum Pflichtenheft gehören. Beispiele dafür sind Modelle (z. B. SDL, UML).