Inhaltsverzeichnis

1. Requirementsanaylse 2

# Requirementsanaylse

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Anforderungen | Datum | Geschätzte Zeit | Benötigte Zeit | Zuständig |
| Zentraler Stern ( Sonne) | 22.11.15 |  | 30min | Adler |
| 2 Planeten | 22.11.15 |  | 30min | Karic |
| Planettextur | 22.11.15 |  | 30min | Karic |
| Planet um eigene Achse | 22.11.15 |  | 60min | Adler |
| Elliptische Bahn um Zentralstern | 22.11.15 |  | 90min | Adler |
| Weitere Planeten | 22.11.15 |  | 80min | Karic |
| Textur Asteroiden |  |  |  |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Events | Datum | Geschätzte Zeit | Zuständig | Erledigt |
| Kamerposition anpassen |  |  | 30min |  |
| Start/Stop der Animation |  |  |  |  |
| Textierung Ein/Aus |  |  |  |  |
| Schatten von Planeten |  |  |  |  |

Zu evaluierende Frameworks:

Pygame

(<http://www.pygame.org/>) is a set of Python modules designed for writing games. It is written on top of the excellent SDL library. This allows you to create fully featured games and multimedia programs in the python language. It is the most popular, and portable game library for python, with over 1000 free and open source projects that use pygame to look at.

Panda3D

(<https://www.panda3d.org>) Panda3D is a game engine, a framework for 3D rendering and game development for Python and C++ programs. Panda3D is Open Source and free for any purpose, including commercial ventures, thanks to its liberal license. Go ahead and grab it here. To learn more about Panda3D's capabilities, visit the gallery and the feature list. To learn how to use Panda3D, check the documentation resources. If you get stuck, ask for help from our community, which is very active.

Planet(Elternklasse)

* Koordinaten(x,y)
* Rotationsspeed
* Translationspeed
* Textur, Farbe
* Schatten
* Größe
* Abhängigkeit(Planet)

Planeten(Kind)

* Koordinaten(x,y)
* Textur, Farbe
* Schatten
* Größe
* Laufbahn

Mond(Kind-Kind)

* Planet
* Koordinaten(x,y)
* Textur, Farbe
* Schatten
* Größe
* Laufbahn

Move(Algorithmen)

* Rotation des Planeten
* Translation(Laufbahn) um abhängigen Planeten

Teammitglieder

Zur Realisierung der Software ist natürlich ein ausreichend qualifiziertes und motiviertes Team nötig. Dieses Team besteht aus Philipp Adler und Adin Karic. Beide derzeit Schüler am TGM mit Fachrichtung Informationstechnologie. Beide Teammitglieder nehmen im Team dieselben gleichberechtigten Rollen ein (Entwickler/Projektmanager)

Tools

Als Entwicklungsumgebung wird PyCharm verwendet. Zur Versionskontrolle wird Git benutzt.

Evaluierung der zu benutzenden Frameworks

|  |  |  |
| --- | --- | --- |
|  | **PyGame** | **Panda3D** |
| **Installation** | 8/10  Nach einiger Recherche erfolgreich | 10/10  Gut beschrieben & ohne Probleme |
| **Komplexität/**  **Handhabung** | 8/10  Man verliert am Anfang ein wenig den Überblick | 8/10  Nach einigem Einlesen akzeptabel |
| **Dokumentation** | 6/10  Viele Tutorials,  Doku gibt’s auch (wenn auch nicht so gut) | 8/10  Sehr ausführliches Manual vorhanden |
| **(Lizenz-)kosten** | 10/10  Freie Nutzung | 10/10  Freie Nutzung |
| **Community** | 7/10  Ist vorhanden, aber nicht so gut wie bei Panda3D | 9/10  Forum, IRC-Channel und Blog sind vorhanden |
| **Prototyp (Example)** | 10/10  Ufo-Spiel als Example  Funktionierte einwandfrei | 10/10  SolarSystem als Example  Funktionierte einwandfrei |
| **Gesamtpunktzahl** | **49/60** | **55/60** |

Aufgrund der obigen Bewertung der beiden Frameworks (mit den gewählten Kriterien) ist die Benutzung von Panda3D für unser Vorhaben zu präferieren.