JUST DO IT
HÖHERE ABTEILUNG FÜR INFORMATIONSTECHNOLOGIE

tgm
Die Schule der Technik

Laborprotokoll

SolarSystem

SEW
5BHITT 2015/16

Burkhard Hampl &
Sebastian Steinkellner

Version 0.1
Begonnen am 03. November 2015
Beendet am 08. Dezember 2015

Inhaltsverzeichnis

1 Projektbeschreibung

Erstelle eine einfache Animation unseres Sonnensystems!

In einem Team (2) sind folgende Anforderungen zu erfüllen.

- Ein zentraler Stern
- Zumindest 2 Planeten, die sich um die eigene Achse und in elliptischen Bahnen um den Zentralstern drehen
- Ein Planet hat zumindest einen Mond, der sich zusätzlich um seinen Planeten bewegt
- Kreativität ist gefragt: Weitere Planeten, Asteroiden, Galaxien,...
- Zumindest ein Planet wird mit einer Textur belegt (Erde, Mars,... sind im Netz verfügbar)

Events:

- Mittels Maus kann die Kameraposition angepasst werden: Zumindest eine Überkopf-Sicht und parallel der Planentenbahnen
- Da es sich um eine Animation handelt, kann diese auch gestoppt werden. Mittels Tasten kann die Geschwindigkeit gedrosselt und beschleunigt werden.
- Mittels Mausklick kann eine Punktlichtquelle und die Textierung ein- und ausgeschaltet werden.
- Schatten: Auch Monde und Planeten werfen Schatten.

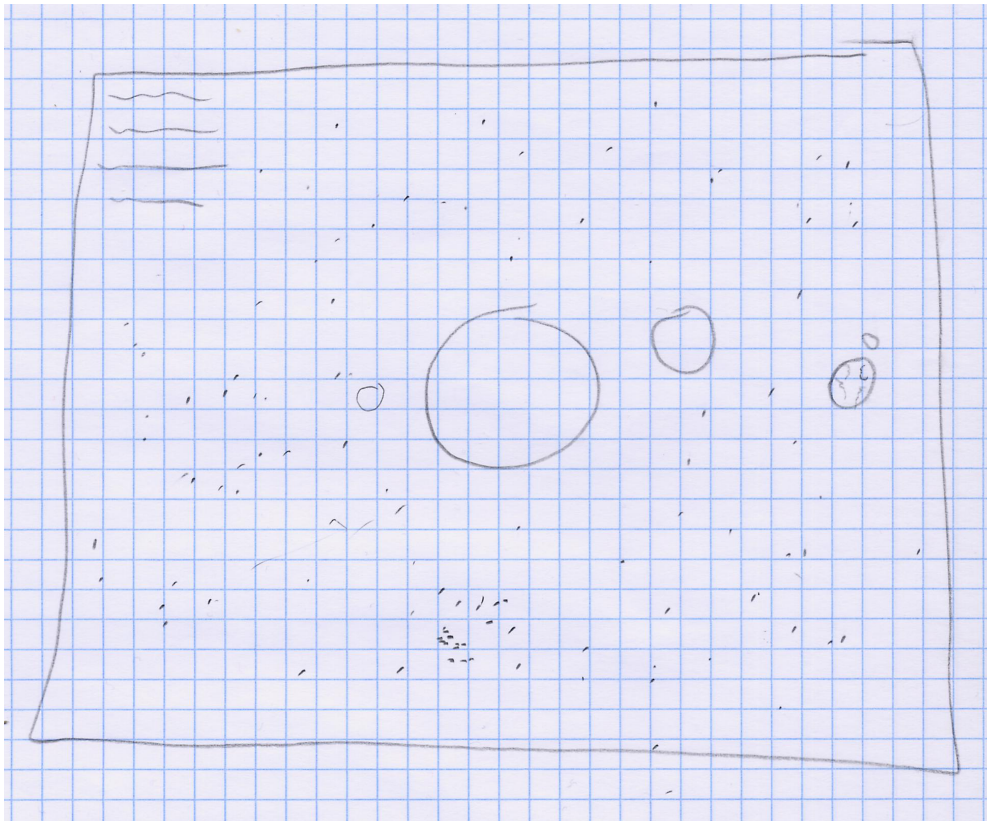
Wählt ein geeignetes 3D-Framework für Python (Liste unter https://wiki.python.org/moin/PythonGameLibraries) und implementiert die Applikation unter Verwendung dieses Frameworks.

Team

Burkhard Hampl und Sebastian Steinkellner

2 Ergebnisse

2.1 GUI-Skizzen und Bedienkonzept



Tastaturbelegung

- „ESCAPE“: Beenden
- „W“: Kamera nach vorn
- „S“: Kamera nach hinten
- „E“: Kamera nach oben
- „Q“: Kamera nach unten
- „A“: Kamera nach links
- „D“: Kamera nach rechts
- „SPACE“: Simulation anhalten und weiterführen
- „B“: Reset Simulation
- „F“:
- „-“: Simulation verlangsamen

- „+“: Simulation verschnellern

2.2 Evaluierung der Frameworks

PyGame

PyGame ist ein Framework welches es recht einfach macht simple Spiele zu implementieren. Das Grundkonzept liegt dabei, dass eine sogenannte „game-loop“ immer wieder aufgerufen wird und man darin die Berechnungen und Darstellungen erledigt.

Das Problem dass wir damit hatten, war das wir den Aufwand um das Solarsystem umzusetzen recht hoch einschätzten.

http://www.nerdparadise.com/tech/python/pygame/basics/part1/

PyWavefront

PyWavefront ist eine Bibliothek welche das Einlesen von 3D Objekt Dateien (.obj und .mtl) ermöglicht. Praktischerweise beinhaltet diese Bibliothek ein Beispiel welches eine Erde darstellt. PyWavefront ist pygame recht ähnlich, es besitzt genauso eine Gameloop, dass es da eine Methode ist, welche mittels einer Annotation markiert wird.

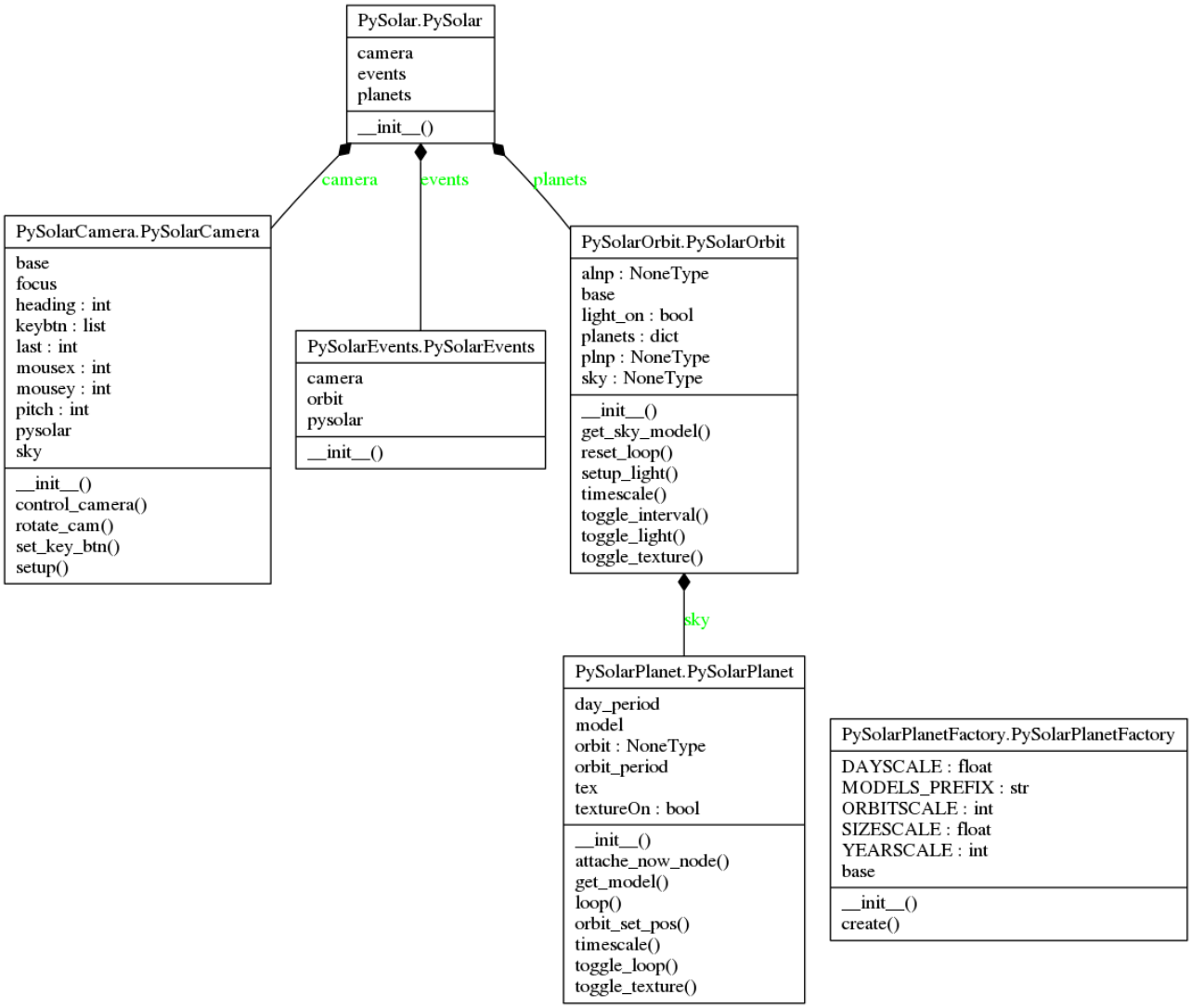
https://github.com/greenmoss/PyWavefront/blob/master/example/pyglet_demo2.py

Panda3D

Panda3D ist eine Spiele Engine für python, welches es einfach macht Spiele zu programmieren. Praktischerweise gibt es ein Beispiel, bei dem ein kleines Sonnensystems beinhaltet.

http://www.panda3d.org/manual/index.php/Sample_Programs:_Solar_System

2.3 Technische Dokumentation



Das Design von PySolar ist relative einfach gehalten. Die Hauptklasse PySolar initialisiert die Simulation und Klassen.

PySolarCamera ist die Kamera Klasse, welche die Steuerung der Kamera übernimmt.

PySolarEvents ist die Event Klasse, welche die Eingaben registriert und darauf reagiert.

PySolarOrbit ist die Orbit Klasse, welche alle Planeten beinhaltet und Verwaltet.

PySolarPlanet ist die Planeten Klasse, welche eine Planeten darstellt, verwaltet und steuert.

PySolarPlanetFactory ist die Factory für die Planeten, welche die Planeten

initialisiert.

Design Patterns

Python mach es einfach Design Principes umzusetzen, da alles eine schwach typisierte Programmiersprache ist, d.h. alles kann einfach ausgetauscht werden, d.h. aber auch, dass die keine „richtigen" Interfaces hat.

In unseren kleinen Beispiel PySolar, kann erstens einmal alles einfach Ausgetauscht werden. Weiteres wurde eine Factory umgesetzte, mit welches das initialisieren der einzelnen Planeten übernimmt, dadurch ist eine einfache Erweiterbarkeit gewährleistet.

2.4 Bedienungsanleitung

Um PySolar ausführe zu können muss Panda3d und Python2 installiert sein, dann kann man PySolar.py mittels Python2 ausführen. Nun Folgt man einfach den Anweisungen oder schaut unter den Punkt Tastaturbelegung nach. Die Steuerung sollte ziemlich Intuitiv und Selbsterklärend sein.