

Datum	Aktivität	Stunden
29.10.2015	Einrichten der Programmierungsumgebung Xcode Verstehen von GLFW und erstellen eines Beispielprogramms zur Darstellung eines Dreiecks	2
30.10.2015	Implementieren eines ShaderManagers zum Laden und Debuggen von Vertex- und Fragmentshadern Erweitern des vorhandenen Vertex- und Fragmentshader zur Interpolierung der Farben des Dreiecks	2
31.10.2015	Implementieren einer Klasse Camera zur Bewegung im Raum Implementieren einer Klasse Keyboard, die es mir erlaubt abzufragen, ob eine Taste gedrückt ist oder nicht Implementieren einer Funktion mouse_callback die es mir erlaubt Mausbewegungen auf die Kamera umzusetzen	3
01.11.2015	Implementieren eines eigenen MeshLoaders zum Laden von .obj Dateien. Erfolgreiches Laden eines Würfels.	2
02.11.2015	Implementieren eines Blinn-Phong Shading Models zur schöneren Darstellung	1
05.11.2015	Erweitern des MeshLoaders. Umbauen auf indiziertes Laden von Objekten. Implementierung der Berechnung von Normals.	3
06.11.2015	Unterscheidung zwischen flat-shaded und smooth-shaded Models und der Berechnung von Normals. Implementierung von Vererbung.	2
07.11.2015	Code-refactoring und vertraut machen wie man GitHub mit Xcode verwendet. Erster commit.	2
08.11.2015	Neuimplementierung des MeshLoaders, da meine eigene Implementierung nicht mit großen Objekten zurechtkam. Verwendung des Librarys Assimp zum Laden von .obj Dateien. Unterscheidung zwischen Flat- und Smoothshaded Models ist nun nicht mehr notwendig. Als Test wurde ein Stanford Bunny geladen. Es hat einwandfrei funktioniert. Andern des Vertex- und Fragmentshader damit Konstanten dynamisch über Uniforms geändert werden können. Erstellen einer Klasse Light die es ermöglicht das Licht dynamisch zu bewegen. Erstellen einer Klasse Material die es ermöglicht das Material dynamisch zu ändern.	5
09.11.2015	Implementieren einer Klasse Heightmap die es ermöglicht eine beliebig große flache Fläche, bestehend aus Dreiecken zu erstellen Die Ecken der Dreiecke sind indiziert und ermöglicht somit auch das effiziente Laden großer Heightmaps.	3
10.11.2015	Erweitern der Klasse Heightmap um es zu ermöglichen die Höhe beliebiger Vertices zu verändern. Vertrautmachen mit dem Headerfile stb_image.h zum Laden von Bilddateien (.png). Verwenden der schwarz-weiß Bildinformationen zum erfolgreichen Laden einer Heightmap.	3
13.11.2015	Überlegen des Aufbaus der Initialpräsentation. Recherche.	2
14.11.2015	Erweitern des Aufbaus der Initialpräsentation. Recherche.	2
15.11.2015	Erweitern des Aufbaus der Initialpräsentation. Erstellen und Suchen von geeigneten Grafiken.	3
19.11.2015	Umsetzen der Initialpräsentation in Latex	3
20.11.2015	Umsetzen der Initialpräsentation in Latex	2
20.12.2015	Implementierung des Diamond-Square-Algorithmus abgeschlossen	7
25.12.2015	Implementierung des Fault-Algorithmus abgeschlossen	3