

Inhaltsverzeichnis

1	Anforderungen und Ziele	1
1.1	Motivation	1
1.2	Ziele der Implementierung	3
1.3	Verwendete Software	4
2	Produktionsprozess: Projektplanung und Analyse	5
2.1	Entwicklungsprozesse in Interaktiver 3D Software und Games .	5
2.1.1	Projektmanagement Modelle	5
2.1.2	Internes Tool Developing oder Tool licencing	8
2.2	Mitglieder eines Entwicklerteams	9
2.2.1	Artists	10
2.2.2	Designer	10
2.2.3	Engineer	12
2.2.4	Weitere für diese Arbeit nicht relevante Teammitglieder .	13
2.3	Stakeholderanalyse intern	13
2.4	Ein Arbeitsprozess wird entwickelt	13
2.4.1	Game Authoring / Game Development	13
2.4.2	Tool Development - der organisatorische Ablauf	14
3	Entwicklung eines Konzeptes	17
3.1	Use Cases der verschiedenen Entwickler	19
3.1.1	Was möchten Artists?	19
3.1.2	Was möchten Designer?	19
3.1.3	Was möchten Entwickler?	19
3.1.4	Prozess bezogen	19
3.1.5	Projekt bezogen	19
3.2	Aktuelle Engines und deren Arbeitsprozesse	19
3.2.1	Prozesse in Game Engines und/oder Frameworks	19
3.2.2	Unreal Engine 4	19
3.2.3	Unity 3D	19

3.2.4	idTech X	19
3.2.5	Weitere	19
3.3	Konzeptentwurf	19
3.3.1	Systemdesign für ein Plugin	19
3.3.2	Systemdesign für einen Project-Handler	19
3.3.3	Entfernen von Abhängigkeiten	19
3.3.4	Zeitersparnis durch bekannte Tools	19
3.3.5	Warum Fusee und Cinema 4D?	19
3.4	Game Assets, Asset Pipelines und Asset Management Tools . .	19
3.4.1	Asset Pipeline und Feedback	20
3.4.2	Asset Pipelines in Fusee	20
3.4.3	Asset Management in Fusee	20
3.4.4	Warum eine Trennung von Code und Content?	20
3.5	Der Weg der Cinema 4D C++ API bis zur Verwendung dieser im Projekt	20
3.5.1	In der Implementierung verwendete Softwareprojekte / Ausgangssituation	20
3.5.2	Cinema 4D Plugin API und SDK	21
3.5.3	Uniplug	25
3.5.4	Fusee	30
3.6	Das eigentliche Plugin	30
3.6.1	Visualisierung der Systemarchitektur	30
3.6.2	Generieren eines Fusee Projektes	31
3.6.3	Code Generation und die Vermeidung von Roundtrips (nicht so ganz roundtrips, generierung um generierung etc.)	31
3.6.4	Partial Classes in .NET	31
3.6.5	XPresso Schaltungen - Visual Programming / Program- mieren ohne Programmieren	31
4	Ergebnisse und Erkenntnisse	32
4.1	Game Authoring Entwicklungsprozesse jetzt und in Zukunft . .	32
4.2	Wie weit ist die Implementierung fortgeschritten?	32
4.3	Welcher Mehrwert wurde erreicht?	32
4.4	Integration des Systems in den weiteren Projektverlauf von FU- SEE	32
	Verzeichnis der Sourcecode Beispiele	34
	Tabellenverzeichnis	35

Abbildungsverzeichnis	36
UML Diagramme	37
Literaturverzeichnis	37