**Bachelorarbeit - Planung**

|  |  |  |  |
| --- | --- | --- | --- |
| *Student\*in:* | Luisa Pätzold | *Matr.Nr.* | 1382147 |
| *Studienfach:* | Mediendesigninformatik | *Betreuer:* | Jan Löcher |

**- Titel -**

Test Driven Game Development

**- Beschreibung -**

*Kurze Zusammenfassung der Aufgabe: Problem/Hintergrund von allgemein bis spezifisch, konkrete Ziele, Herangehensweise/Methode, Herausforderungen und Chancen, optionale Verläufe, erwartetes Ergebnis. Tipp: Schreiben Sie erst diese Zusammenfassung, wenn alle anderen Kapitel feststehen.*

Im Rahmen dieser Arbeit soll das Prinzip des Test-Driven Developments auf die Entwicklung von Videospielen angewendet werden. Der Begriff Test-Driven Development (TDD) beschreibt Softwareentwicklung, die auf automatischen Tests basiert. Als Einstieg in die Thematik wird das Prinzip des Test-Driven Developments sowie dessen aktuelle Verwendungsbereiche und -umstände erläutert. Daraufhin werden Überlegungen getroffen, wie dieser Ansatz auf die Entwicklung von Videospielen in Unity3D sowie der Unreal Engine 4 angewendet werden könnte. Aufbauend darauf soll mithilfe eines minimalistischen Spiels in Unity3D der Versuch gewagt werden, die getroffenen Überlegungen in die Praxis umzusetzen. Hierbei wird dokumentiert, welche Herangehensweisen sich speziell bei Videospielen empfehlen, welche Probleme auftreten und wie sie sich gegebenenfalls lösen lassen. Sollte sich der Versuch der Umsetzung insgesamt als erfolgreich und sinnvoll erweisen, so werden anschließend die gesammelten Erkenntnisse nach Möglichkeit auf ein umfangreicheres Spiel angewendet, entweder in Unity3D oder der Unreal Engine 4. Sollte die Umsetzung hingegen scheitern, werden in einer Evaluation die Gründe dafür dargelegt und abschließend bewertet. Unabhängig vom Ausgang des Versuchs soll am Ende eine Aussage über den Nutzen sowie die Grenzen von Test-Driven Development für die Spieleentwicklung getroffen werden.

**- Motivation -**

*Wie ist der Kontext der Aufgabe. Ist diese Teil von einer großen Aufgabe? Was ist der Sinn dahinter? Welche Probleme will man lösen? Warum ist das wichtig? Warum ist der Lösungsansatz vielversprechend?*

Test Driven Development gilt als Best Practice und wird unter anderem in der Webentwicklung erfolgreich eingesetzt. Bei der Spieleentwicklung ist dieser Ansatz dagegen kaum verbreitet. In der Praxis gestaltet es sich oft schwierig, automatisierte Unit Test Frameworks in Game Engines einzubinden. Zudem ist es meist nicht klar, wie man für 3D- und Echtzeitszenarien Tests schreiben kann, oder wie man Spielarchitekturen so modular aufbauen kann, dass sie automatisch testbar bleiben. Im Rahmen dieser Arbeit soll daher das Prinzip des Test Driven Developments in der Videospielentwicklung modellhaft versucht und anschließend auf Tauglichkeit bewertet werden.

**- Ziele –**

*Welche messbaren Ziele/Deliverables gibt es im Verlauf der Bearbeitung?*

* Prototyp eines minimalistischen Spiels, entwickelt mit TDD in Unity3D
* darauf aufbauende Sammlung von Richtlinien für TDD in der Spieleentwicklung
* ggf. Anwendung von TDD auf ein komplexeres Spiel (Unity3D oder Unreal Engine 4) oder ausführliche Bewertung der Untauglichkeit von TDD

**- Ansatz, Methodik, Vorgehensweise und Aufgaben –**

*Wie gehen Sie konkret vor? Was ist der grundlegende Ansatz? Welche konkreten Aufgaben ergeben sich daraus? Welchen Hintergrund erarbeiten Sie? Wie machen Sie das? Konzipieren Sie, Implementieren Sie, evaluieren Sie, jeweils was und wie?*

Der Ansatz ist, zunächst grundlegend das Verfahren von Test Driven Development darzulegen und praktische Übung zu erlangen, indem es auf die Entwicklung eines einfachen Spiels angewendet wird. Die daraus resultierenden Erfahrungen fließen in Richtlinien für die Verwendung von TDD bei der Spieleentwicklung ein. Bei erfolgreicher Verwendung werden diese Richtlinien anschließend auf ein umfangreicheres Spiel angewendet, andernfalls deren Untauglichkeit begründet.

* Darlegung des Prinzips von Test Driven Development
* Entwicklung eines minimalistischen Spiels unter Verwendung von TDD in der Unity3D Engine
* Aufstellung von Richtlinien für die Anwendung von TDD in der Spieleentwicklung
* Bei Erfolg: Anwendung dieser Richtlinien auf ein umfangreicheres Spiel, um ihre Tauglichkeit zu beweisen. Hierbei wird entweder ein Spiel in Unity3D aus einem früheren Semester oder ein aktuelles Spiel der Firma Limbic Entertainment in der Unreal Engine 4 verwendet
* Andernfalls: Begründung der Untauglichkeit in Form einer Evaluation

**- Zusätzliche Anforderungen, Abhängigkeiten, Risiken -**

*Wovon hängt Ihre Arbeit ab, was benötigen Sie? Müssen Sie Ergebnisse von anderen verarbeiten, die noch nicht fertig sind? Benötigen Sie Hardware, die die Abteilung noch nicht angeschafft hat? Wie sicher ist die rechtzeitige Verfügbarkeit? Welche Alternativen ergeben Sich?*

Für die Entwicklung dieser Arbeit bedarf es keines besonderen Equipments, die Unity3D Engine sowie ausreichend leistungsstarke Workstations stehen dauerhaft zur Verfügung und können problemlos verwendet werden. Für die Anwendung von TDD auf ein umfangreicheres Spiel ist es von Vorteil, wenn dieses bereits größtenteils fertig vorliegt, da es zeitlich nicht möglich sein wird, mehr als ein sehr einfaches Spiel für diese Arbeit zu entwickeln. Ein Tower Defense Game aus einem früheren Semester bietet sich hierfür bereits als Kandidat an, andernfalls sollte ein passendes Game vom betreuenden Dozenten zur Verfügung gestellt werden, was bereits zugesichert wurde. Sollte die Verwendung von TDD in der Unreal Engine realistisch erscheinen und ausreichend Zeit zur Verfügung stehen, so kann hierfür auch ein aktuelles Spiel von Limbic Entertainment in UE4 verwendet werden.

**- Erfolgskontrolle -**

*Wann haben Sie das Problem gelöst und wie kann man das messen? Was sind Mindestanforderungen? Was wäre eine besonders herausragende Leistung?*

Der Erfolg dieser Arbeit wird am ausführlichen Versuch der Verwendung von TDD sowie der anschließenden Bewertung und Sammlung an Richtlinien gemessen. Hierbei ist es nicht zwingend notwendig, dass dieser Versuch erfolgreich endet.

**- Zeitplan –**

*Wie verteilen Sie Ihre Aufgaben auf den zeitlichen Verlauf der Bearbeitung? Berücksichtigen Sie, wann Sie ggf. abwesend sind, andere Verpflichtungen haben, wo viele Feiertage sind. Wo bauen Sie Pufferzeiten ein, was sind feste Meilensteine und Deadlines?*

| ***Schritt*** | ***Abschluss*** |
| --- | --- |
| - Einarbeitung in Test Driven Development + Vorgehensweise übersichtlich aufstellen  - Recherche, wie Unity & Unreal TDD unterstützen | 11.11.2018 |
| - Anmeldung der Bachelorarbeit | 12.11.2018 |
| - Darlegung von Test Driven Development  - vorläufige Überlegung, wie es in der Spieleentwicklung funktionieren könnte  - Einarbeitung in den Unity Test Runner  - grobe Einarbeitung in Unreal Automation System  - vorläufige Planung des minimalistischen Spiels  - GitHub Set Up | 01.12.2018 |
| Versuch der Verwendung von TDD im Rahmen der Entwicklung des minimalistischen Spiels mithilfe des Unity Test Runners | 21.12.2018 |
| Sammlung und Formulierung der bei der Entwicklung gefundenen Richtlinien/Best Practices/Probleme | 07.01.2018 |
| Anwendung der Richtlinien auf das umfangreichere Spiel: Lässt sich TDD auch auf bereits begonnene Spieleentwicklung anwenden?  - falls möglich, dann in Unreal mit einem Spiel von Limbic  - andernfalls eigenes Spiel in Unity | 04.02.2019 |
| Puffer und finales Schreiben | 11.02.2019 |
| Abgabe der Arbeit | 12.02.2019 |