

Pen and PapAR

—Martin Miller Projektbericht —

March 14, 2025

1 Aller Anfang ist schwer...

Das Projekt selbst war meine eigene Vision, allerdings war der Start des Projekts aus mehreren Gründen doch sehr holprig. Der erste Plan war es mit der Unreal Engine 5 zu arbeiten, da ich mit dieser insgesamt mehr Erfahrung hatte. Nach mehreren Ansätzen und mehreren Versuchen, ein HoloLens Projekt aufzusetzen, stiessen wir auf mehrere Probleme. Zum einen stiessen wir, damals nur Cassandra Lenk und ich, auf einen Artikel, in welchem Unreal den Support für die HoloLens 2 beendet hatte. Zum Anderen hatten wir Versuch eine andere ältere Version zu verwenden das Problem, das diese nicht auf den Computern in der HTWK verfügbar war.

Somit mussten wir nochmal neu starten und haben das Projekt kurz vor Weihnachten auf Unity begonnen. Zu dieser Zeit kamen zwei weitere Personen ins Team. Salem Zin Ider Nasser und Jannis Kerz stiessen zu uns und mit Jannis hatten wir jemanden an Bord, der sich mit Unity auskannte. Er erstellte ein kleines Tutorial, um einen Einstieg in Unity zu erleichtern. Darauf hin begann die Arbeit in Unity an dem Projekt PenAndPapAR mit reichlich Verzögerung zum neuen Jahr.

2 Die Arbeit am Projekt

Mit der neuen Teamgröße war nun auch ein größerer Scope möglich, allerdings war nun auch eine genauere Teamstruktur vonnöten. Als der Urheber der Idee und einer der erfahrensten DnD Spieler im Team wurde mir die Rolle als Teamlead zugeteilt. Bei der Einteilung für die ersten Elemente wurde mir und Jannis das Implementieren der generellen Statistiken zugeteilt. Dabei habe ich Layer 1 und 3 übernommen und Jannis hat Layer 2 übernommen. Ich habe anschliessend das grobe Design der aller 3 Layer übernommen.

Dazu habe ich Anfangs im wöchentlichen Meeting Protokoll geführt, bis etwa Februar, wo ich damit aufhörte, weil diese Protokolle sowieso nie gelesen wurden und ich mich mehr auf das Nacharbeiten der noch fehlenden Features in unserem Projekt fokussieren wollte. Auch habe ich am Anfang des Projekts das Projekt eingerichtet und den Editor für das Implementieren einer HoloLens 2 Anwendung eingerichtet, so wie das Aufsetzen der Szene und des Gits. Letzteres geschah in Zusammenarbeit mit Jannis Kerz, nachdem das von mir zuvor aufgesetzte Unity Version Control System sich als unbrauchbar für unser Projekt und unseren Workflow herausstellte. Dies erforderte auch erstmal das Herausfiltern von dem riesigen Overhead den der Unity Editor 6 mit in die Projektdateien brachte.

Nachdem die allgemeinen *Stats* fertig implementiert waren, habe ich mich darum gekümmert den Charakter in einer Datenbank speichern zu können und wieder abrufen zu können und vorerst einen MOCK-Up Charakter in der Schnittstelle erstellen lassen, um die Funktionalität der Stats überprüfen zu kommen. Allerdings wurde schnell klar, dass das Interface, wie es bei uns aktuell noch erstellt wurde, nicht anklickbar war. Es fehlten Button Objekte, wodurch ich mich dann an die Implementierung der Clickables kümmerte und meinen Plan der Datenbank anbindung erstmal zur Seite schob. Danach begann ich die Einrichtung der HoloLens und musste feststellen, dass das Projekt aus irgendeinem Grund nicht sichtbar war in der HoloLens. Ich richtete sie ein, aber übergab die Aufgabe, diesen Bug zu beheben dann an Salem Zin Ider Nasser und widmete mich stattdessen einem weiteren Problem, das sich abzeichnete.

Denn es wurde deutlich, dass das mergen in Unity komplexer war als erwartet mit Git. So geschah es zum Beispiel bei unserem ersten Merge Versuch, das plötzlich die Szenen unbrauchbar waren und Teilweise sogar das Projekt selbst im Unity Editor nicht mehr geöffnet werden konnte. Somit übernahm ich auch das Beheben etlicher Merge Konflikte und das Einführen eines Workflows zum mergen, von zwei Branches aus Unity um zukünftig zu verhindern, das ein Merge die Settings und Szenen, die bereits implementiert waren, erneut zerstört. Dies geschah dann mit einer Django DB, weil ich mit Django bereits erste berührungspunkte hatte und so einen einfacheren Einstieg hatte, um diese Datenbank anschliessend über eine *REST-Schnittstelle* mit dem Unity-Projekt zu Verbinden.

Danach erweiterte ich noch die Sichtbarkeiten der Stats und erweiterte sie um die Funktion auch das dazugehörige 3D-Objekt aufzurufen.

Dies nahm einen großen Teil meiner Zeit in den letzten Februar Wochen ein zusammen mit dem Anpassen der Schnittstelle in Unity. Nachdem beides fertig war, ging es für mich ab März dann zum Thema implementation der

REST Schnittstelle selbst in Kommunikation mit Jannis Kerz, damit dieser seine *DnD-Beyond* API an meine REST-API anschliessen konnte.

Zwischendurch hatte ich auch bereits vorarbeit geleistet und mich schlau gemacht über eine mögliche 3D-Map Generierung, basierend auf gängigen Dungeon Builder Anwendungen. Diese boten nämlich das Exportieren ihrer Maps an, um sie in Virtuellen Tabletop Spielen zu nutzen, wie zum Beispiel Forge oder Roll20 es anbietet. Die Idee war simpel: Ein 5 foot Quadrat auf der Karte wird dargestellt durch einen 3D-Würfel, welcher auf der Oberseite als Textur einen Teil der Map enthielt. Die Informationen wären in einer JSON enthalten und so würde man dann eine Map zusammen basteln können. Leider fehlten zum Stand jetzt zwei Dinge, um diese Funktion wirklich zu implementieren. Die ausgegebenen JSON Daten boten zwar Informationen für *Fog of War* und die angabe, wie gross ein 5 Feet Kästchen in Pixeln sein soll auf dem ausgegebenen Image. Allerdings waren die 3D-Informationen entweder ganz verloren, oder für uns zwar auffindbar, aber nicht lesbar.