

Erfahrungsbericht

Robin Enderle

Sprint 1:

Ziel: Objekte anwählen, verschieben und rotieren

Im Sprint 1 ging es für mich zuerst darum, Unity, C# wie auch das MOSIM-Framework kennen zu lernen. „Learning by doing“ war hier die beste Möglichkeit sich die wichtigsten Funktionen bei zu bringen. Nach einer Besprechung mit meinen Teamkollegen wählte ich die Aufgabe Objekte verschiebbar zu machen. Dafür musste ich zuerst Objekte anwählbar machen. Als Rückgabeparameter sollte man das ausgewählte Objekt bekommen. Nachdem ich dies geschafft hatte musste ich herausfinden, wie ich die Mauskoordinaten in Weltkoordinaten konvertieren kann. Diese neuen Koordinaten konnte ich dann dem Objekt zuweisen. Schwierigkeiten gab es allerdings, wenn das Objekt noch Collider hatte. Beim Verschieben konnte es daher sein, dass das Objekt mit anderen Objekten kollidiert. Eine weitere Schwierigkeit war die Schwerkraft. Diese wirkte kontinuierlich gegen die Mausposition, wenn diese oberhalb des Bodens war. Diese zwei Probleme konnten durch temporäres Deaktivieren von Collidern und der auf dieses Objekt wirkenden Schwerkraft gelöst werden.

Als zweite Aufgabe setzte ich mich an die Funktion Objekte rotieren zu können. Durch Recherche im Internet gelang mir dies relativ schnell. Auch hier müssen Collider wie auch die Schwerkraft temporär deaktiviert werden.

Sprint 2:

Ziel: Objekte rotieren mit sichtbaren Rotationsachsen

Für die Rotationsfunktion sollten jetzt noch Achsen eingeblendet werden, damit der Benutzer besser feststellen kann um welche Achse er das Objekt drehen kann. Dies erwies sich allerdings als schwerer als gedacht. Die Achsen sollten sich je nach Größe des Objekts in der Länge anpassen und zum richtigen Zeitpunkt ein- bzw. ausgeblendet werden. Nach verschiedenen Versuchen mit Codebeispielen aus dem Internet gelang es mir dann die Achsen dynamisch anzuzeigen.

Sprint 3:

Ziel: Avatar-Bewegungen steuern

In diesem Sprint arbeite ich mit Jonas Schoeler an Bewegungsabläufen des MOSIM Avatars. Zuerst haben wir den vorhandenen Code mit den dazu mitgelieferten Avatar-Funktionen analysiert. Diese Bewegungen haben wir dann mit eigenen Codeteilen versucht selbst anzusteuern. Nach wenigen Versuchen gelang uns dies auch. Dabei stellten wir fest, dass uns noch ein wichtiges Funktionselement fehlte. Und zwar das Erstellen des Zielobjekts beim verschieben von Objekten. Solange sich Jonas weiterhin mit den Abläufen beschäftigt,

entwickelte ich die Funktion Zielobjekte zu erstellen. Zur späteren Vereinfachung sind dies Kind-Objekte des zu verschiebenden Objekts. Dabei ergab sich allerdings das Problem, dass sich Kind-Objekte beim Verschieben des Elternobjekts mitbewegen. Dies wurde durch ein weiteres Skript verhindert. Nun musste nur noch die Farbe des Kind-Objekts zur besseren Verständlichkeit verändert werden. Danach habe ich mich mit Jonas über den aktuellen Stand ausgetauscht und wir haben wieder zusammen an dynamischen Bewegungsabläufen gearbeitet. Die Schwierigkeit hierbei sind die Startbedingungen jeder Bewegung, da diese voneinander abhängig gemacht werden müssen. Alle Abhängigkeiten müssen vor dem Abspielen der Simulation feststehen, da der Ablauf dann vom Framework selbst übernommen wird.

Sprint 4:

Ziel: Avatar-Bewegungen automatisch nacheinander anlaufen

Weiterentwicklung der dynamischen Bewegungsabläufe. Dabei mussten viele Tests durchgeführt und Probleme gelöst werden. Diese dynamischen Bewegungsabläufe waren für mich eines der schwierigsten Dinge im Projekt. Durch Rücksprachen mit Herrn Gaisbauer konnten wir jedoch unser Ziel erreichen.

Sprint 5:

Ziel: Präsentation vorbereiten, Dokumentation vervollständigen

Zur Präsentation des Projekts haben wir uns ein Simulationsablauf mit einem Bus vorgestellt. Dieser soll von dem Avatar mit ein paar wenigen Bauteilen bestückt werden. Diese Bauteile wie auch der Bus wurde von mir im Autodesk Inventor konstruiert und mit Jonas Schoeler zusammen für die Verwendung in Unity zum Prefab verarbeitet. Zusätzlich wurde die Dokumentation vervollständigt. Zur Präsentation wurde ein Video des Simulationsablaufs erstellt.

Lessons Learned:

- Learning by doing ist in so einem Fall für mich die beste Variante mit der Materie vertraut zu werden. Dies erfordert zwar hohes Durchhaltevermögen doch das Verständnis ist nachher dafür umso höher.
- Codebeispiele im Internet sind nicht immer die Beste Variante auch wenn sie dort so dargestellt werden.
- Manchmal muss viel probiert werden. Mit vielen Versuchen und Änderungen den Überblick zu halten ist schwer. Immer wieder kurze Pausen zu machen, um wieder von einem anderen Blickwinkel darauf zu haben ist deshalb sehr hilfreich.
- Auch wenn es Dokumentationen über Frameworks gibt, ist es manchmal sehr hilfreich direkt mit den Entwicklern des Frameworks zu kommunizieren.

- Dokumentation früher auf Vollständigkeit prüfen und einer höheren Priorität zuordnen.
- Probleme im Team besprechen ist oft sehr hilfreich. Andere Personen, die noch nichts mit dem Problem zu tun hatten, haben oft eine neue und hilfreiche Herangehensweise