Individuelle Abschlussarbeit BLJ

Migros Tycoon

Finn Domeisen

27. Juni 2025

Der Migros-Genossenschafts-Bund





# Inhaltsverzeichnis

[Inhaltsverzeichnis 2](#_Toc201825415)

[1 Änderungstabelle 4](#_Toc201825416)

[2 Einleitung 6](#_Toc201825417)

[2.1 Aufgabenstellung und Ziel des Projekts 6](#_Toc201825418)

[1. Projektbeschreibung 6](#_Toc201825419)

[2. Bekannte Risiken zu Projektbeginn 6](#_Toc201825420)

[3 Planung 7](#_Toc201825421)

[3. Geplante Projektstruktur 7](#_Toc201825422)

[Tatsächlicher Projektverlauf 8](#_Toc201825423)

[3.1 Planung 9](#_Toc201825424)

[3.2 Terminplan (Gantt-Diagramm) 9](#_Toc201825425)

[3.3 Welche Wege gehe ich? Welche Möglichkeiten habe ich? 9](#_Toc201825426)

[3.4 Entscheidungsmatrix 10](#_Toc201825427)

[3.5 Fazit 10](#_Toc201825428)

[4 Hauptteil 11](#_Toc201825429)

[4.1 Vorgehen 11](#_Toc201825430)

[Milestone 1 14](#_Toc201825431)

[Milestone 2 15](#_Toc201825432)

[Milestone 3 15](#_Toc201825433)

[4.2 Ergebnis der Arbeit 15](#_Toc201825434)

[4.3 Funktionsbeschreibung 16](#_Toc201825435)

[ Dropper & Förderband 16](#_Toc201825436)

[ Collector 16](#_Toc201825437)

[ Kaufen-Buttons 16](#_Toc201825438)

[ Upgrader 16](#_Toc201825439)

[4.4 Arbeitsjournal 16](#_Toc201825440)

[4.5 Testplan 16](#_Toc201825441)

[ Kauf-Funktion 16](#_Toc201825442)

[ Dropper-Logik 16](#_Toc201825443)

[ Soundauslösung 16](#_Toc201825444)

[ GUI-Anzeige 16](#_Toc201825445)

[ Tutorial 17](#_Toc201825446)

[5 Arbeitsjournale 18](#_Toc201825447)

[5.1 Tag 1 – 4. Juni 2025 18](#_Toc201825448)

[5.2 Tag 2 - 5. Juni 2025 18](#_Toc201825449)

[5.3 Tag 3 - 6. Juni 2025 18](#_Toc201825450)

[5.4 Tag 4 - 11. Juni 2025 18](#_Toc201825451)

[5.5 Tag 5 - 12. Juni 2025 19](#_Toc201825452)

[5.6 Tag 6 - 13. Juni 2025 19](#_Toc201825453)

[5.7 Tag 7 - 18. Juni 2025 19](#_Toc201825454)

[5.8 Tag 8 - 19. Juni 2025 19](#_Toc201825455)

[5.9 Tag 9 - 20. Juni 2025 20](#_Toc201825456)

[5.10 Tag 10 - 25. Juni 2025 20](#_Toc201825457)

[5.11 Tag 11 - 26. Juni 2025 20](#_Toc201825458)

[5.12 Tag 12 - 27. Juni 2025 20](#_Toc201825459)

[6 Anhang 21](#_Toc201825460)

# Änderungstabelle

|  |  |  |  |
| --- | --- | --- | --- |
| **Version** | **Datum** | **Änderung** | **Beschreibung** |
|  |  |  |  |
| 0.1 | 4.6.2025 | Repository erstellt | Milestones und Issues auf GitHub erstellt. |
| 0.2 | 5.6.2025 | Dokumentation erstellt und Theorie erarbeitet | Word-Datei für die Dokumentation erstellt und Issues eingefügt. Schwerpunkt lag auf Theorie: Videos geschaut, Artikel zu Avatar, UI, UserInput, Objekten und Scripting gelesen. Wichtige Punkte notiert. |
| 0.3 | 6.6.2025 | Weltgestaltung abgeschlossen und Spawnbereich aufgebaut | Die Spielfläche wurde erstellt: Bodenplatte mit Gras, Umrandung aus Hochhäusern und Bergen, Himmelmodell aus der Toolbox. In der Mitte ist der Start-Spawn mit Migros-Logo. Von dort führen vier Wege zu den Tycoon-Plätzen. Unsichtbare Wand hinzugefügt und dekorative Bäume platziert. |
| 0.4 | 11.6.2025 | Erster Dropper, Förderband und Geldanzeige umgesetzt | Im Homeoffice wurden Floor, Wände, Decke und Eingang des Tycoons mit Parts modelliert. Anschliessend entstand der erste Dropper, der mithilfe von Tutorials und ChatGPT pro Sekunde einen Block erzeugt. Ein erstes Förderband mit Script wurde eingebaut, ebenso eine funktionierende Geldanzeige via ScreenGUI und LocalScript. Der Collector zur Umwandlung der Blöcke in Geld wurde begonnen, funktioniert aber noch nicht wie gewünscht. |
| 1.0 | 12.6.2025 | Gameplay und Kauflogik verbessert, neues Design umgesetzt | Der Fokus lag auf Gameplay und Lua-Skripting. Die Kauflogik wurde verbessert: Geld wird korrekt abgezogen, Käufe sind nur bei ausreichendem Betrag möglich. Das Haus erhielt ein neues Design mit realistischen Materialien. Die Sliding Door wurde optimiert. Zudem wurde begonnen, die Kaufreihenfolge zu begrenzen. Ein realistisches 3D-Modell eines Migros Ice Tea wurde mithilfe einer KI erstellt. Die Dokumentation wurde heute kaum erweitert. |
| 1.3 | 13.6.2025 | Weitere Dropper, Produkte-Modelle und GitHub-Issues ergänzt | Zwei Dropper und ein Upgrader wurden gebaut. Mehrere Migros-Produkte (Glace, Energy, Wackelpudding) wurden als 3D-Modelle via AI generiert und ins Regal eingefügt. Ein Titelbild wurde ebenfalls mit AI erstellt. GitHub-Issues wurden aktualisiert. Die Dokumentation blieb unbearbeitet. |
| 1.7 | 18.6.2025 |  |  |
| 1.9 | 19.6.2025 |  |  |
| 2.0 | 20.6.2025 |  |  |
| 2.1 | 25.6.2025 |  |  |
| 2.2 | 26.6.2025 |  |  |
| 2.2 | 27.6.2025 |  |  |
| 2.2 |  |  |  |

# Einleitung

## Aufgabenstellung und Ziel des Projekts

Im Rahmen meines Abschlussprojekts habe ich mich entschieden, ein eigenes Spiel in Roblox zu entwickeln. Die Plattform Roblox Studio bietet die Möglichkeit, Spiele mit einer Kombination aus 3D-Modellierung, Benutzeroberflächen und der Programmiersprache Lua zu erstellen. Mein Ziel war es, ein Tycoon Spiel zu realisieren, bei dem man durch verdientes Geld eine Migros Filiale Schritt für Schritt aufbauen kann. Dabei beginnt der Spieler mit einem leeren Grundstück und verdient Geld durch Button Klicks oder automatische Dropper. Dieses Geld kann genutzt werden, um verschiedene Bauelemente wie Wände, Däcker, Beleuchtung und Migros Produkte zu kaufen.

## Projektbeschreibung

Die Aufgabenstellung bestand darin, ein funktionierendes und visuell ansprechendes Spiel zu gestalten, das eine klare Logik und ein motivierendes Belohnungssystem hat. Zusätzlich sollte das Spiel die Möglichkeit bieten, den Spielfortschritt nicht zu speichern, damit Spieler das Spiel mehrmals spielen kann. Neben dem technischen Aspekt war es mir auch wichtig, die Migros als Marke im Spiel wiederzuerkennen. Deshalb integrierte ich typische Migros Produkte, das Logo und das bekannte orange Farbdesign.

## Bekannte Risiken zu Projektbeginn

Zu Beginn des Projekts waren mir wenige Risiken bewusst. Einer der grössten Unsicherheitsfaktoren war meine fehlende Erfahrung mit der Programmiersprache Lua. Ich wusste, dass es einige Zeit brauchen würde, um die Grundlagen zu verstehen und sie korrekt anzuwenden. Auch der komplexe Aufbau eines Tycoon Spiels mit mehreren Droppern, einem Förderband, einem Sammler, einem Geldsystem, Benutzeroberflächen und automatisierten Abläufen stellte eine grosse Herausforderung dar. Zudem war unklar, ob alle Modelle und Grafikelemente technisch sauber funktionieren und wie viel Zeit das Modellieren und das Testing beanspruchen würde.

Trotz dieser bekannten Risiken habe ich mich bewusst für ein Projekt entschieden, das mich sowohl gestalterisch als auch technisch fordert. Mein Ziel war es, nicht nur ein Spiel zu bauen, sondern dabei auch viel über die Zusammenhänge zwischen Design, Logik und Code zu lernen.

# Planung

## Geplante Projektstruktur

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | 4.6 | | | 5.6 | | | 6.6 | | | 11.6 | | | 12.6 | | | 13.6 | | | 18.6 | | | 19.6 | | | 20.6 | | | 25.6 | | | 26.6 | | | 27.6 | | |
| Milestone 1 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Projektbewilligung |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Tutorial Videos |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Tutorial 1. lesen |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Tutorial 2. lesen |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Tutorial 3. lesen |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Roblox Account |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Tutorial Einführung |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Plattform erstellen |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Spawnpunkt erstellen |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Spawnpunkt dekorieren |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Tycoon Plätze |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Tycoon Auswahl |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Geld UI |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Map Wände |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 1. Dropper |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Blender research |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Blender installieren |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Milestone 2 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Alles kaufbar machen |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Migros Modelle |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Sounds |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Ladebildschirm |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Milestone 3 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Kontrollieren |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Testen |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Optional: Veröffentlichen |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |

## Tatsächlicher Projektverlauf

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | 4.6 | | | 5.6 | | | 6.6 | | | 11.6 | | | | 12.6 | | | | 13.6 | | | | 18.6 | | | | 19.6 | | | | 20.6 | | | | 25.6 | | | | 26.6 | | | | 27.6 | | | |
| Milestone 1 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Projektbewiligung |  |  |  |  |  |  |  |  |  | |  |  |  | |  |  |  | |  |  |  | |  |  |  | |  |  |  | |  |  |  | |  |  |  | |  |  |  | |  |  |  | |
| Tutorial Videos |  |  |  |  |  |  |  |  |  | |  |  |  | |  |  |  | |  |  |  | |  |  |  | |  |  |  | |  |  |  | |  |  |  | |  |  |  | |  |  |  | |
| Tutorial 1. lesen |  |  |  |  |  |  |  |  |  | |  |  |  | |  |  |  | |  |  |  | |  |  |  | |  |  |  | |  |  |  | |  |  |  | |  |  |  | |  |  |  | |
| Tutorial 2. lesen |  |  |  |  |  |  |  |  |  | |  |  |  | |  |  |  | |  |  |  | |  |  |  | |  |  |  | |  |  |  | |  |  |  | |  |  |  | |  |  |  | |
| Tutorial 3. lesen |  |  |  |  |  |  |  |  |  | |  |  |  | |  |  |  | |  |  |  | |  |  |  | |  |  |  | |  |  |  | |  |  |  | |  |  |  | |  |  |  | |
| Roblox Account |  |  |  |  |  |  |  |  |  | |  |  |  | |  |  |  | |  |  |  | |  |  |  | |  |  |  | |  |  |  | |  |  |  | |  |  |  | |  |  |  | |
| Tutorial Einführung |  |  |  |  |  |  |  |  |  | |  |  |  | |  |  |  | |  |  |  | |  |  |  | |  |  |  | |  |  |  | |  |  |  | |  |  |  | |  |  |  | |
| Plattform erstellen |  |  |  |  |  |  |  |  |  | |  |  |  | |  |  |  | |  |  |  | |  |  |  | |  |  |  | |  |  |  | |  |  |  | |  |  |  | |  |  |  | |
| Spawnpunkt erstellen |  |  |  |  |  |  |  |  |  | |  |  |  | |  |  |  | |  |  |  | |  |  |  | |  |  |  | |  |  |  | |  |  |  | |  |  |  | |  |  |  | |
| Spawnpunkt dekorieren |  |  |  |  |  |  |  |  |  | |  |  |  | |  |  |  | |  |  |  | |  |  |  | |  |  |  | |  |  |  | |  |  |  | |  |  |  | |  |  |  | |
| Tycoon Plätze |  |  |  |  |  |  |  |  |  | |  |  |  | |  |  |  | |  |  |  | |  |  |  | |  |  |  | |  |  |  | |  |  |  | |  |  |  | |  |  |  | |
| Tycoon Auswahl |  |  |  |  |  |  |  |  |  | |  |  |  | |  |  |  | |  |  |  | |  |  |  | |  |  |  | |  |  |  | |  |  |  | |  |  |  | |  |  |  | |
| Geld UI |  |  |  |  |  |  |  |  |  | |  |  |  | |  |  |  | |  |  |  | |  |  |  | |  |  |  | |  |  |  | |  |  |  | |  |  |  | |  |  |  | |
| Map Wände |  |  |  |  |  |  |  |  |  | |  |  |  | |  |  |  | |  |  |  | |  |  |  | |  |  |  | |  |  |  | |  |  |  | |  |  |  | |  |  |  | |
| 1. Dropper |  |  |  |  |  |  |  |  |  | |  |  |  | |  |  |  | |  |  |  | |  |  |  | |  |  |  | |  |  |  | |  |  |  | |  |  |  | |  |  |  | |
| Blender research |  |  |  |  |  |  |  |  |  | |  |  |  | |  |  |  | |  |  |  | |  |  |  | |  |  |  | |  |  |  | |  |  |  | |  |  |  | |  |  |  | |
| Blender installieren |  |  |  |  |  |  |  |  |  | |  |  |  | |  |  |  | |  |  |  | |  |  |  | |  |  |  | |  |  |  | |  |  |  | |  |  |  | |  |  |  | |
|  |  |  |  |  |  |  |  |  |  | |  |  |  | |  |  |  | |  |  |  | |  |  |  | |  |  |  | |  |  |  | |  |  |  | |  |  |  | |  |  |  | |
| Milestone 2 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Alles kaufbar machen |  |  |  |  |  |  |  |  |  | |  |  |  | |  |  |  | |  |  |  | |  |  |  | |  |  |  | |  |  |  | |  |  |  | |  |  |  | |  |  |  | |
| Migros Modelle |  |  |  |  |  |  |  |  |  | |  |  |  | |  |  |  | |  |  |  | |  |  |  | |  |  |  | |  |  |  | |  |  |  | |  |  |  | |  |  |  | |
| Sounds |  |  |  |  |  |  |  |  |  | |  |  |  | |  |  |  | |  |  |  | |  |  |  | |  |  |  | |  |  |  | |  |  |  | |  |  |  | |  |  |  | |
| Ladebildschirm |  |  |  |  |  |  |  |  |  | |  |  |  | |  |  |  | |  |  |  | |  |  |  | |  |  |  | |  |  |  | |  |  |  | |  |  |  | |  |  |  | |
|  |  |  |  |  |  |  |  |  |  | |  |  |  | |  |  |  | |  |  |  | |  |  |  | |  |  |  | |  |  |  | |  |  |  | |  |  |  | |  |  |  | |
| Milestone 3 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Kontrollieren |  |  |  |  |  |  |  |  |  | |  |  |  | |  |  |  | |  |  |  | |  |  |  | |  |  |  | |  |  |  | |  |  |  | |  |  |  | |  |  |  | |
| Testen |  |  |  |  |  |  |  |  |  | |  |  |  | |  |  |  | |  |  |  | |  |  |  | |  |  |  | |  |  |  | |  |  |  | |  |  |  | |  |  |  | |
| Optional: Veröffentlichen |  |  |  |  |  |  |  |  |  | |  |  |  | |  |  |  | |  |  |  | |  |  |  | |  |  |  | |  |  |  | |  |  |  | |  |  |  | |  |  |  | |
|  |  |  |  |  |  |  |  |  |  | |  |  |  | |  |  |  | |  |  |  | |  |  |  | |  |  |  | |  |  |  | |  |  |  | |  |  |  | |  |  |  | |

## Planung

Für ein erfolgreiches Projekt war eine klare und realistische Planung wichtig. Deshalb habe ich mir bereits zu Beginn überlegt, wie ich meine Zeit aufteilen möchte, welche Ziele ich erreichen will und welche verschiedenen Wege zur Auswahl stehen. Mein Ziel war nicht nur ein funktionierendes, spannendes Roblox Spiel, sondern auch ein Projekt, bei dem ich Neues lernen und mich weiterentwickeln kann.

## Terminplan (Gantt-Diagramm)

Das Projekt dauerte insgesamt zwölf Tage. Um die Zeit sinnvoll zu nutzen, habe ich mein Projekt in mehrere Meilensteine in GitHub unterteilt. Diese Meilensteine halfen mir, die Arbeitsetappen logisch zu strukturieren.

In einem Gantt-Diagramm habe ich diese Schritte zeitlich visualisiert. Dabei habe ich auch Pufferzeiten eingeplant, zum Beispiel für Fehler oder längeres Debugging. In diesen freien Zeiten habe ich ausserdem an der Dokumentation gearbeitet oder mein GitHub-Projekt beziehungsweise das Repository aktualisiert, um den Fortschritt sauber festzuhalten und die Übersicht zu behalten.

## Welche Wege gehe ich? Welche Möglichkeiten habe ich?

Bereits zu Beginn habe ich verschiedene Projektideen gesammelt und ihre Vor- und Nachteile miteinander verglichen. Ich wollte ein Projekt umsetzen, das sowohl kreativ als auch technisch anspruchsvoll ist. Es sollte mir die Möglichkeit geben, eine Programmiersprache zu lernen, visuell zu gestalten und gleichzeitig ein funktionierendes Produkt abzugeben.

## Entscheidungsmatrix

|  |  |  |  |
| --- | --- | --- | --- |
| **Möglichkeit** | **Vorteile** | **Nachteile** | **Entscheidung** |
|  |  |  |  |
| **Roblox Tycoon** | Kreativ, frei gestaltbar, Lua lernen, Community-Support | Kein Vorwissen mit Programmiersprache, Einarbeitung nötig | Gewählt |
| **ABAP-Tool** | Bezug zur zukünftigen Arbeit (Migros SAP), realistisch | Kein Spiel, weniger kreative Gestaltung, schwierige Fehlerbehebung | Abgelehnt |
| **Unity Mini Game** | Moderne Engine, viele Möglichkeiten, gute Grafik | Hohe Komplexität, schon mal gemacht | Abgelehnt |

Die Wahl fiel nicht direk klar auf das Roblox-Tycoon-Spiel. Jedoch kombiniert es Spiellogik, Gestaltung, Animation, Programmierung und Benutzerinteraktion. Das sind genau die Bereiche, die mich interessieren. Ausserdem konnte ich mit Lua eine neue Sprache lernen und hatte dank der Roblox Community Zugriff auf viele Hilfestellungen, Vorlagen und Hilfeforen.

Das ABAP-Projekt hätte zwar gut zu meiner künftigen Abteilung gepasst, wäre aber deutlich technischer und weniger kreativ gewesen. Die Umsetzung eines Spiels mit Unity habe ich mich dagegen entschieden, weil ich schon ein Escape Room als Sportferien Projekt gemacht habe.

## Fazit

Insgesamt war die Planung für mich ein sehr wichtiger Teil des Projekts. Sie hat mir geholfen, realistisch zu bleiben, klare Prioritäten zu setzen und meine Zeit sinnvoll einzuteilen. Auch wenn ich nicht jeden einzelnen Punkt exakt nach Plan erledigen konnte, war die grobe Struktur für meinen Fortschritt sehr hilfreich.

# Hauptteil

* Vorgehen, Zwischenschritte
* Ergebnis der Arbeit (z.B. Konstruktionszeichnungen, Zusammenstellungen)
* Funktionsbeschreibung (mit Querverweis zum Anhang)
* Arbeitsjournal
* Testplan

## Vorgehen

### Projektbewilligung und Zielsetzung

Zuerst musste ich ein selber gewähltes Projekt planen und bewilligen lassen. Dabei hatte ich drei Ideen, die mir am besten gefielen und über welche ich mir viel Gedanken gemacht habe. Mit Hilfe von dem vorbereiteten Chat GPT prompt von Frank, konnte mir viele Ideen geben, die ich möglicherweise als Projekt machen könnte.Leider war ich mit keiner richtig glücklich und habe überlegt, was mich in der freizeit interessiert, oder ob ich ein problem hat, für das ein programm das problem lösen würde oder etwas mit migros zu tun hat.

Zur Auswahl waren: Ein Spiel mit Unity, ein Tool mit Abap und ein Roblox Spiel. Da ich in den Sportferien schon ein Spiel mit Unity erstellt habe, fand ich es wäre langweilig, wenn ich nochmal ein Unity Spiel machen würde. Jedoch hat mir das Spiel erstellen sehr spass gemacht und ich fand es sehr spannend.

Das zweite; ein ABAP Tool war zum einen toll, da es mit Migros, beziehungsweise meiner zukünftigen Abteilung zu tun hat. Jedoch finde ich ein Tool zu machen nicht sehr ansprechend, wenn man jetzt andere Möglichkeiten hätte.

Selber spiele ich ab und zu Roblox und weiss deshalb, dass sie eine Spiele Engine gratis anbieten. Dann dachte ich mir ich könnte doch ein Roblox Spiel erstellen. Die Idee das ich etwas mit Migros verbinde habe ich nicht vergessen, deswegen habe ich ein Tycoon-Spiel in Roblox erstellt, bei dem man eine Migros-Filiale aufbauen kann.

Nach dem ich meine Idee Reto vorgetragen, musste ich ihm noch erklären was Roblox überhaupt ist. Als Reto mein Projekt bewilligt hat, fing ich direkt an.

Als erstes habe ich das GitHub Repository erstellt und das ReadME geschrieben. Dazu habe ich noch das GitHub Project erstellt und mit dem Repository verknüpft.

Im Project habe ich die 3 Meilensteine erstellt und habe die ersten Issues geschrieben. Dies war am Anfang schwer, weil man nicht weiss was noch dazwischen kommt oder ob etwas fehlt.

### Tutorials & Einführung in Roblox

Ich startete mit mehrern Roblox video tutorials und learning artikel gelesen die allgemein um Roblox gehen, aber auch um spezifische Themen wie Avatar, Animation, UI, Userinput, Objekte, Software, Scripting.

Es gibt von dem offizielen roblox account sehr viele hilf-, erklär und tutorial videos. Diese habe ich auf Youtube geschaut oder habe auf [https://create.roblox.com/docs/](https://create.roblox.com/docs/de-de) nachgelesen

Ich startete mit mehreren Roblox- und Lua-Tutorials, um die Grundlagen zu verstehen:

* Benutzeroberfläche von Roblox Studio
* Wie Lua funktioniert (Variablen, Bedingungen, Events, Funktionen)
* Wie man GUIs, Buttons und Trigger erstellt
* Was ein Tycoon-Modul ist und wie man es anpasst

Ich las mehrere Tutorials (1–3), die ich mir auch notierte.

* Roblox Account erstellen und vorbereiten

Ich erstellte einen Roblox-Account und richtete die Roblox Studio Umgebung ein. Dann begann ich mit einem leeren Template, in das ich mein eigenes Tycoon-System einbauen wollte.

* Plattform & Map vorbereiten

Ich baute eine eigene Map:

* Eine große Plattform mit Grastextur
* Darauf erstellte ich vier Plätze für Tycoon-Spieler
* Ich gestaltete einen dekorierten Spawnpunkt mit Migros-Logo
* Ich fügte Berge, Steine und Landschaftselemente hinzu
* Tycoon-System und erstes Gameplay

Dann begann ich mit der Umsetzung des eigentlichen Spiels:

* Ich platzierte Tycoon-Plätze und richtete die Ownership-Systeme ein (Touch-to-Claim)
* Ich programmierte das Giver- und Collector-System
* Ich fügte eine Cash-Variable hinzu, die beim Sammeln von Blöcken erhöht wird
* Ich gestaltete eine Geld-UI zur Anzeige des aktuellen Guthabens
* Dropper, Förderband & Upgrader

Ich entwickelte die ersten Dropper, die regelmäßig graue Blöcke erzeugen. Diese bewegen sich über ein Förderband zum Collector.  
Später fügte ich Upgrader hinzu, welche die Blöcke (z. B. zu Migros Ice Tea) umwandeln, damit sie mehr Geld wert sind.

* Kaufen-Logik und Buttons

Ich programmierte die Kauf-Logik:

* Spieler sehen rote Flächen mit einem Preis
* Nur wenn sie genug Geld haben, können sie kaufen
* Nach dem Kauf wird das Objekt sichtbar
* Die Buttons verschwinden danach und geben ggf. den nächsten frei (Reihenfolge)

Außerdem: Ein Sound wird beim Kauf abgespielt.

* Migros-Modelle mit AI erstellen

Ich wollte Migros-Produkte im Regal. Da ich nicht alles selbst modellieren konnte, nutzte ich eine AI, die aus Bildern 3D-Modelle generierte. So entstanden:

* Migros Ice Tea
* Energy Drink
* Glace
* Wackelpudding

Diese stellte ich im Spiel in einem Regal dar.

* Sounds hinzufügen

Ich baute passende Sounds ein, z. B.:

* Beim Kauf eines Objekts
* Beim Einsammeln von Geld
* Beim Beanspruchen eines Tycoon-Platzes

Diese Sounds liegen im ReplicatedStorage und werden zur Laufzeit abgespielt.

* Ladebildschirm (Loading Screen)

Ich programmierte ein GUI, das beim Start des Spiels erscheint:

* „Migros Tycoon“ Schrift mit Leucht-Effekt
* Orangefarbener Ladebalken
* Prozentanzeige  
  Das GUI blendet sich automatisch aus, sobald das Spiel geladen ist.
* Slide-Tutorial erstellen

Ich wollte neuen Spielern helfen, die Spielfunktionen zu verstehen. Deshalb erstellte ich ein Tutorial mit Slideshow:

* GUI mit „Next“-Button
* Zeigt Texte + Bilder (Shift = Rennen, Rote Fläche = Kaufen, Dropper etc.)
* Optional ein animiertes GIF als letztes Bild
* Testphase

Ich testete das Spiel ausführlich:

* Funktionieren alle Buttons?
* Werden Objekte korrekt freigeschaltet?
* Spielt der Sound zur richtigen Zeit?
* Läuft das Tutorial sauber durch?
* Funktionieren Dropper und Upgrader zuverlässig?

Fehler dokumentierte ich in meinem GitHub-Issue-Board und bearbeitete sie dort.

### Was ist ein Tycoon?

### Was ist Roblox?

Mein Projekt „Migros Tycoon“ startete mit einer strengen Einarbeitungsphase in die Grundlagen von Roblox Studio und der Programmiersprache Lua. In den ersten Tagen arbeitete ich mit Online-Tutorials, Dokumentationen und Videos, um die Grundlogik von Tycoon Spielen zu verstehen. In dieser Phase war mein Hauptziel, die Komplexität des Vorhabens realistisch einzuschätzen und gleichzeitig die Projektstruktur zu planen.

Ich habe mein Projekt in drei zentrale Meilensteine eingeteilt, die den Ablauf strukturieren und den Fortschritt sichtbar machen:

* **Milestone 1: Vorbereitung & Grundlagen**
* **Milestone 2: Spielmechanik und Inhalte**
* **Milestone 3: Testing & Kontrolle (Dokumentation)**

Jeder Meilenstein war mit konkreten Aufgaben verknüpft, zum Beispiel „Roblox Account erstellen“, „Spawnpunkt dekorieren“, „Erster Dropper“, „Sounds“ oder „Ladebildschirm“. Diese Planung hat mir sehr geholfen, den Überblick zu behalten und mich jeden Tag auf einzelne Schritte zu konzentrieren.

Milestone 1 beinhaltete die theoretische Einführung. Ich lernte viel über die Funktionsweise von GUIs, UserInput, Animationen und Objekte in Roblox.

Ausserdem bereitete ich die Spielwelt vor: Ich gestaltete die Hauptplattform, fügte eine Grasplatte darüber hinzu, modellierte Gebäude und Berge und legte den zentralen Spawnpunkt fest. Dieser wurde mit einem Migros Logo dekoriert. Anschliessend baute ich vier Wege, die zu den jeweiligen Tycoon Plätzen führen.

Im Verlauf dieses Abschnitts entwickelte ich auch erste einfache Tycoon Mechaniken: Ich programmierte einen Dropper, der regelmässig Blöcke an einem bestimmten Ort erzeugt, die sich über ein programmiertes Förderband zur Collector Platte bewegen, welche die Blöcke löscht und den Geldstand hochzählt. Hier kam Lua intensiv zum Einsatz. Ich lernte, wie man „Touched“-Events nutzt, GUI-Werte aktualisiert und Variablen auswertet. Dies war eine grosse Herausforderung, da ich vorher keine Lua Erfahrung hatte. Ich habe manchmal die Unterstützung von ChatGPT genutzt und dessen Vorschläge in mein Projekt eingebaut.

Milestone 2 fokussierte sich auf die spielerische Interaktivität.

Ich machte die bereits erstellten Gegenstände kaufbar, indem ich ein Button System programmierte. Spieler können Objekte nur dann kaufen, wenn sie genug Geld haben. Ausserdem arbeitete ich an einem System, das eine Reihenfolge beim Freischalten von Objekten erzwingt. Also dass man erst das vorherige Objekt kaufen muss, damit man das nächste überhaupt sieht.

Parallel dazu erstellte ich Migros Produkte mithilfe von AI-generierten 3D-Modellen (Ice Tea, Glace, Wackelpudding etc.), die in einem Regal ausgestellt werden. Das Erlebnis wurde durch passende Sounds ergänzt, die beim Kaufen oder Einsammeln von Geld abgespielt werden. Auch der Ladebildschirm habe ich in dieser Phase programmiert. Dieser zeigt animiert „Migros Tycoon“ sowie eine Prozentanzeige und einen Ladebalken.

Milestone 3 konzentrierte sich auf Kontrolle und Test.

Ich überprüfte die Funktionalität aller Scripts, testete die Logik der Dropper, überprüfte UI-Anzeigen, kaufbare Objekte und die Animationen. Um neuen Spielern die wichtigsten Funktionen verständlich zu machen, habe ich eine Slide-Show erstellt, die zentrale Spielmechaniken wie das Rennen mit Shift oder das Kaufen über rote Flächen erklärt. Ich plante keine Speicherung des Spielfortschritts, da der Tycoon nicht sehr lange dauert und man ihn deshalb mehrmals machen können sollte.

## Ergebnis der Arbeit

Das Ergebnis ist ein vollständiges, fehlerfreies Tycoon-Spiel, das es Spielern ermöglicht, ihre eigene Migros Filiale aufzubauen. Es beginnt mit der Auswahl eines Platzes und führt über den Aufbau von Dropppern, Upgradern und Deko-Objekten zur Simulation eines funktionierenden Geschäfts. Alle Objekte werden visuell durch Produkte und Logos dargestellt. Die Benutzeroberfläche ist benutzerfreundlich und wurde komplett mit GUIs, Sounds und Ladebildschirm ausgestattet. Das Spiel bietet einsteigerfreundliches Gameplay und eine positive, markentreue Atmosphäre.

## Funktionsbeschreibung

Die Hauptfunktionen lassen sich in vier Gruppen einteilen:

* Dropper & Förderband: Generieren regelmässig Blöcke, die automatisch in Richtung Collector transportiert werden. Lua steuert dabei die Position und Geschwindigkeit.
* Collector: Erkennt per Script, wenn ein Block mit „Cash“-Wert ihn berührt, zerstört ihn und erhöht den Spielerkontostand.
* Kaufen-Buttons: Diese befinden sich auf roten Flächen. Sobald ein Spieler genug Geld hat und die Fläche berührt, wird der nächste Gegenstand freigeschaltet und ein Sound abgespielt.
* Upgrader: Verwandeln einfache graue Blöcke in farbige Produkte mit höherem Wert (z. B. Migros Ice Tea).

## Arbeitsjournal

Ich habe täglich ein Journal geführt, dass meine Fortschritte, Herausforderungen und Lösungswege dokumentiert. Diese Einträge zeigen den realen Verlauf im Vergleich zur geplanten Gantt-Übersicht. Besonders deutlich wird dabei, dass ich zwischendurch mehr Zeit mit Coding verbrachte als ursprünglich angenommen. Dafür konnte ich einige Abschnitte, wie z. B. Map-Dekoration, schneller abschließen.

## Testplan

Um mein Spiel auf Funktionalität zu testen, habe ich folgende Tests durchgeführt:

* Kauf-Funktion: Spieler können nur mit ausreichend Geld kaufen, Reihenfolge wird eingehalten.
* Dropper-Logik: Blöcke werden korrekt erzeugt, transportiert und eingesammelt.
* Soundauslösung: Alle geplanten Sounds (Kauf, Einsammeln, Ownership) spielen zur richtigen Zeit.
* GUI-Anzeige: Geldstand, Ladebalken, Texte und Buttons funktionieren korrekt.
* Tutorial: Slide-Wechsel, Bilder und Texte passen und sind logisch.

Fehler oder Bugs habe ich in meinem GitHub-Repository in den Issues dokumentiert und dort auch Lösungen ergänzt.

# Arbeitsjournale

## Tag 1 – 4. Juni 2025

Ich habe mein Abschlussprojekt offiziell gestartet. Anfangs war ich noch unsicher über das Thema. Mit Hilfe eines KI-Prompts suchte ich nach Ideen, aber keine hat mich wirklich überzeugt. Ich dachte über sinnvolle Tools oder Spiele nach und kam schließlich auf die Idee, ein eigenes Spiel zu machen. Da ich privat manchmal Roblox spiele, entschied ich mich für ein **Tycoon-Spiel, bei dem man eine Migros-Filiale aufbaut**. Das passte auch zu meinem Lehrbetrieb. Ich richtete mein GitHub Repository ein, erstellte das Kanban-Board und definierte erste Issues.

## Tag 2 - 5. Juni 2025

Heute war ein reiner Theorie-Tag. Ich war vom vielen Lesen und Aufnehmen ziemlich erschöpft. Ich habe mir verschiedene Lerninhalte angeschaut: Tutorials, Artikel, Dokumentationen zu Roblox Studio und Lua. Themen waren unter anderem: Avatare, Animationen, UI, Userinput, Objekte, Scripting. Ich machte mir Notizen zu den wichtigsten Punkten und füllte meine Dokumentationsstruktur weiter.

## Tag 3 - 6. Juni 2025

Ich habe heute die Spielwelt gebaut. Ich begann mit einer Grundplatte, darüber eine Grasfläche. Der Rand bestand auf einer Seite aus Hochhäusern, auf der anderen aus Bergen. Den Himmel habe ich aus der Toolbox importiert. Ich erstellte vier Wege von einem zentralen Punkt aus, wo auch der Spawnpunkt liegt – dekoriert mit einem Migros-Logo. Um die Welt realistischer zu machen, baute ich noch Bäume und eine unsichtbare Wand, damit Spieler nicht aus der Welt herauslaufen können.

## Tag 4 - 11. Juni 2025

Ich hatte Homeoffice und arbeitete intensiv am Spiel. Ich erstellte Floor, Walls, Decke und Eingang eines Tycoon-Gebäudes. Dann begann ich mit dem ersten Dropper: Zwei Parts und ein Skript erzeugen regelmässig einen Block. Mit Tutorials und Hilfe von ChatGPT klappte das. Danach baute ich ein Förderband, das noch nicht ganz flüssig läuft. Ich gestaltete auch die Geldanzeige mit ScreenGUI, TextLabel und LocalScript. Am Schluss arbeitete ich am Collector, welcher Geld geben soll, aber noch nicht korrekt funktioniert.

## Tag 5 - 12. Juni 2025

Ich machte große Fortschritte im Bereich Gameplay und Logik. Die Hausstruktur überarbeitete ich mit besseren Materialien, es sieht jetzt realistischer aus. Ich optimierte die Sliding Door und programmierte die Kauf-Logik: Man kann nur kaufen, wenn man genug Geld hat. Auch die Reihenfolge von Upgrades wird nun geprüft. Für das Förderband suchte ich nach besseren Lösungen. Mit AI erstellte ich ein 3D-Modell vom Migros Ice Tea, das hervorragend gelang.

## Tag 6 - 13. Juni 2025

Ich arbeitete weiter an der Spielmechanik. Ich generierte ein cooles Titelbild mit AI, das die Stimmung des Spiels gut einfängt. Ich pflegte mein GitHub-Projekt mit neuen Issues und schloss erledigte Aufgaben ab. Ich erstellte weitere Migros-Modelle: Energy Drink, Glace, Wackelpudding. Im Regal sehen sie sehr schön aus. Ich baute zwei neue Dropper und begann mit dem Upgrader, der graue Blöcke in wertvollere Migros-Produkte verwandelt.

## Tag 7 - 18. Juni 2025

Ich arbeitete heute ausschließlich an der Dokumentation. Ich erstellte eine vollständige Änderungstabelle und schrieb die Einleitung. Anschließend begann ich mit der Planung: Ich baute ein Gantt-Diagramm auf, was durch viele Aufgaben recht aufwändig war. Zusätzlich erstellte ich eine Entscheidungsmatrix zur Projektwahl und schrieb den zugehörigen Textabschnitt.

## Tag 8 - 19. Juni 2025

Heute hatte ich Homeoffice. Am Vormittag schrieb ich an der Dokumentation, schloss die Planung und das Gantt-Diagramm ab. Am Nachmittag arbeitete ich im Spiel: Ich suchte passende Sounds, konvertierte sie und importierte sie ins Roblox Creator Hub. Statt jeden Sound mehrfach einzubauen, lagerte ich sie im ReplicatedStorage. Ich reparierte auch fehlerhafte Modelle, die in der Luft schwebten. Als nächstes plante ich den Ladebildschirm.

## Tag 9 - 20. Juni 2025

Ich konnte heute den Ladebildschirm fertigstellen – animiert, farbig, mit Ladebalken und Prozentanzeige. Zuerst hatte ich ein Problem mit der Darstellung über den ganzen Bildschirm, dass ich durch eine Änderung im LocalScript lösen konnte. Danach begann ich mit dem Tutorial-GUI, das beim Spielstart erscheint. Es erklärt in mehreren Slides die Grundfunktionen des Spiels. Ich räumte den Code noch etwas auf. Zum Schluss schrieb ich weiter an der Dokumentation.

## Tag 10 - 25. Juni 2025

## Tag 11 - 26. Juni 2025

## Tag 12 - 27. Juni 2025

# Anhang

* Quellenangaben, Literaturverzeichnis (verbindliche Angaben)
* Begriffserklärungen/Glossar (verbindliche Angaben)
* Programm-Code
* Scripts
* Foto-Dokumentation
* KI-Chat-Auszüge