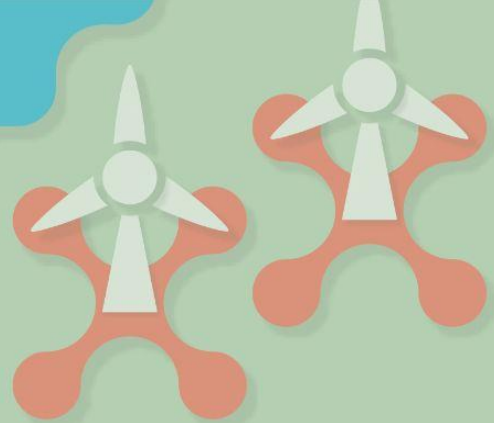
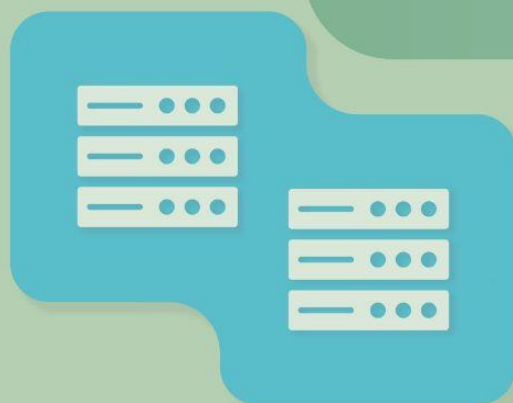




WEBLAB:



WebLab – Entwickler:innen

Dokumentation

Was ist WebLab?

WebLab ist ein digitales Puzzle-Spiel über Internet-Infrastruktur und deren materiellen Ressourcenverbrauch. Spieler:innen bauen ein funktionierendes Netzwerk, indem sie Häuser und Infrastrukturkomponenten miteinander verbinden und gleichzeitig Energiebedarf sowie Ressourcenverbrauch berücksichtigen.

Das Spiel wurde als Teil der Dauerausstellung „**Materialwelten**“ im **Technischen Museum Wien** entwickelt. Ziel ist es, die materiellen Grundlagen digitaler Infrastruktur sichtbar zu machen und die ökologischen Auswirkungen von Internetnutzung verständlich zu vermitteln.

WebLab wurde mit der **Unity Engine** umgesetzt und ist als interaktive Anwendung für Touchscreens und Webbrowser konzipiert. Die Unity-Engine übernimmt grundlegende Funktionen wie Input Handling, Rendering, Kollisionsprüfung und Szenenverwaltung.

Dokumentation

Das Spiel kombiniert Elemente aus Puzzle-, Strategie- und Lernspielen und verfolgt einen **Serious-Game-Ansatz**: Spielerische Interaktion soll Verständnis für Ressourcenverbrauch, Energieversorgung und nachhaltige Infrastruktur schaffen.

Für wen ist WebLab? Wem hilft es wodurch?

WebLab richtet sich primär an:

Museumsbesucher:innen

- Besucher:innen der Ausstellung „Materialwelten“
- Jugendliche und Erwachsene mit Interesse an Technologie und Nachhaltigkeit

Bildungskontext

- Lehrkräfte und Vermittler:innen im Bereich MINT, Medienbildung und Nachhaltigkeit
- Schulklassen und informelle Lernsettings

Spielinteressierte Nutzer:innen

- Menschen, die sich spielerisch mit Infrastruktur, Ressourcenverbrauch und Klimawirkung digitaler Technologien beschäftigen möchten

Das Spiel hilft dabei:

- **digitale Infrastruktur verständlich zu machen**
- den Zusammenhang zwischen **Internetnutzung und Ressourcenverbrauch** sichtbar zu machen
- **systemisches Denken über Technologie und Nachhaltigkeit** zu fördern

Durch die Visualisierung von Netzwerken, Energieflüssen und Ressourcenlimits werden komplexe Zusammenhänge in einer interaktiven Simulation erfahrbar.

Diese Entwickler:innen-Dokumentation richtet sich speziell an Personen, die:

- das Spiel technisch erweitern möchten
- neue Level oder Spielmechaniken entwickeln wollen
- die Softwarearchitektur nachvollziehen möchten

Ein Ziel des Projektes war es daher, die technische Struktur möglichst modular zu halten.

Wie funktioniert WebLab?

Grundprinzip des Spiels

Spieler:innen bauen ein Netzwerk, indem sie verschiedene Infrastrukturkomponenten auf einem Spielfeld platzieren.

Die zentrale Aufgabe besteht darin:

- Häuser miteinander zu verbinden
- Infrastrukturkomponenten zu platzieren
- ausreichende Energieversorgung sicherzustellen
- Ressourcenlimits einzuhalten

Je nach Effizienz erhalten Spieler:innen **bis zu drei Sterne pro Level**.

Technische Architektur

Weblab wurde in der Unity Engine entwickelt und verwendet Unity's eigene Lösungen für Input-Handling, Rendering, Collision Detection und so weiter. Aufbauend darauf wurde die Grundfunktionalität des Spieles entwickelt:

Tile-System

Die grundlegenden Bausteine des Spiels sind sogenannte **Tiles**.

Ein Tile besteht aus mehreren quadratischen Unterkomponenten, die zusammen unterschiedliche Formen bilden können.

Tile Spawner

Die Tile Spawner existieren auf der UI Ebene des Spiels. Werden sie mit dem Finger angefasst oder mit dem Cursor geklickt, wird das entsprechende Tile Prefab instanziiert und Drag and Drop aktiviert. Der Tile Spawner behält außerdem den Überblick darüber, wie viele Tiles noch platziert werden können und deaktiviert sich wenn bereits alle verfügbaren Tiles dieser Sorte auf dem Spielfeld vorhanden sind.

Drag-and-Drop System

Wird ein Tile von den Tile Spawnern instanziiert oder vom Spielfeld aufgehoben, aktiviert sich das Drag and Drop und kümmert sich darum, dass das Tile auf die Position des Fingers oder des Cursors gesetzt wird, bis es losgelassen wird. Gleichzeitig wird die Position des Tiles auf den Bildschirm beschränkt.

Wird das Tile losgelassen, wird es zuerst so verschoben, dass es genau mit dem Raster übereinstimmt. Danach wird mithilfe von 2D Box Collider überprüft, ob das Tile innerhalb des Spielfeldes liegt und nicht mit anderen Tiles überlappt. Sollte es falsch platziert sein, wird es zurück zum Tile Spawner gesendet und gelöscht.

Tile Grid

Das Tile Grid basiert auf Unity's Grid Component, womit World-Space Koordinaten auf ein Raster abgebildet werden können. Wird ein Tile korrekt auf dem Spielfeld platziert, werden die X und Y Position jeder Unterkomponente im Raster-Koordinatensystem abgespeichert.

Tile Connection

Sobald Tiles auf dem Spielfeld platziert wurden, werden die umliegenden Felder des Rasters nach Nachbar-Tiles abgesucht. Wird ein Nachbar gefunden, so wird dies abgespeichert. Nachdem alle neuen Nachbarn vermerkt wurden, wird ein Algorithmus gestartet, der vom Startpunkt ausgehend einen Weg über alle verbundenen Tiles zu den Endpunkten sucht. Verschiedene Pfade werden abgegangen und gespeichert. Am Ende werden sich überlappende Pfade aussortiert und die Wege, die übrig bleiben, werden mithilfe eines Unity Line-Renderers auf das Spielfeld gezeichnet. All dies geschieht jedes mal, wenn sich das Spielfeld ändert - also ein neues Tile platziert oder ein bereits platziertes Tile entfernt oder verschoben wird.

Objectives

Jedes Level hat verschiedene Objectives - Aufgaben, die gelöst werden müssen um ein bis drei Sterne zu erreichen und das Level abzuschließen. Jede Aufgabe zählt für sich, zu wie viel Prozent sie vollendet ist - diese Prozente werden zu einem übergreifenden Prozentsatz zusammen gezählt. Aufgaben von höheren Sternen können jedoch erst freigeschaltet werden, wenn die Aufgaben des niedrigeren Sternes vollendet sind.

Zu den Aufgaben, die nachverfolgt werden, zählt zum Beispiel: Welche Endpunkte verbunden oder nicht verbunden sind, ob zu viele Ressourcen verwendet werden, oder ob genug Stromversorgung vorhanden ist.

Erweiterbarkeit des Systems

Die Architektur erlaubt mehrere Erweiterungen:

Neue Level

- zusätzliche Karten
- neue Netzwerkstrukturen

Neue Tile-Typen

- Infrastrukturkomponenten
- Energiequellen
- Spezialelemente

Neue Objectives

- CO₂-Grenzwerte
- Energieeffizienz
- Ressourcenlimits

Didaktische Erweiterungen

- zusätzliche Visualisierungen
 - erklärende Inhalte
 - Daten zu Rohstoffen oder Energieverbrauch
-

Weiterentwicklung

WebLab wurde so entwickelt, dass zukünftige Erweiterungen möglich sind.

Dazu gehören beispielsweise:

- neue Spielmechaniken
- zusätzliche Infrastrukturtypen
- neue Szenarien oder Level
- weitere Visualisierungen von Ressourcenflüssen

Die modulare Struktur der Gameplay-Systeme erleichtert Anpassungen und Erweiterungen.