

Erweiterte Raumfunktionen

Lernergebnisse:

Nach Abschluss der Übung sind die Studierenden in der Lage, verschiedene Werkzeuge der Raumgestaltung zu verwenden, um eine interaktive, multimediale Umgebung in Form eines WorkAdventure Raums zu gestalten. Die Studierenden sind in der Lage, die API von WA zu navigieren und Skripte zu entwerfen.

1. Mediengestaltung und Raumgestaltung

(a) Laden Sie das Buch „Grundlagen der Mediengestaltung : Konzeption, Ideenfindung, Bildaufbau, Farbe, Typografie, Interface Design“, 2021 von C. Fries aus der Hochschulbibliothek herunter.

(b) Verschaffen Sie sich einen Überblick über Kapitel 3 – Visuelle Grunderfahrungen und S.190-192 .

Diskutieren Sie mit der Seminargruppe Möglichkeiten, wie diese Informationen in die Gestaltung eines WA Raumes integriert werden können.

Empfohlen: Suchen Sie im Buch weitere Informationen und entwickeln Sie Verbindungen zur Gestaltung von WA Räumen. Suchen Sie auch außerhalb des Buchs nach Gestaltungsgesetzen/-elementen.

Beispielgedanke:

Durch die Leserichtung von links nach rechts im Deutschen ist eine Gestaltung der angedachten Bewegung im Raum in diese Richtung empfehlenswert.

(c) Hilfestellung zur Projektarbeit – zur selbstständigen Bearbeitung

Verschaffen Sie sich einen Überblick zu S. 136-141 zum Thema Konzeption. Sie können diese Informationen als mögliche Grundlage für Ihre Raumkonzeption betrachten, indem Sie Verbindungen zu Ihrem Fachthema und Inhalten der Vorlesung schaffen. Integrieren Sie außerdem Sound, Text und andere Medien und Funktionen in Ihre Raumgestaltung.

Sie übernehmen die Rolle von Lehrenden, Ihre Zielgruppe sind Lernende. Genauere Überlegungen sind Ihnen überlassen.

2. Überblick und Übung Skripte

Öffnen Sie map.tmj des Map Starter Kits. Öffnen Sie main.ts im Ordner „src“.

WorkAdventure stellt die Möglichkeit zur Verwendung von Skripten mittels Typeskript/Javaskript zur Verfügung.

Sie können eigene, simple Funktionen integrieren oder auf eine API von zusätzlichen Funktionen zugreifen.

Dokumentation: <https://workadventu.re/map-building/scripting.md> 30.03.2023

Dokumentation: <https://workadventu.re/map-building-extra/about.md> 30.03.2023

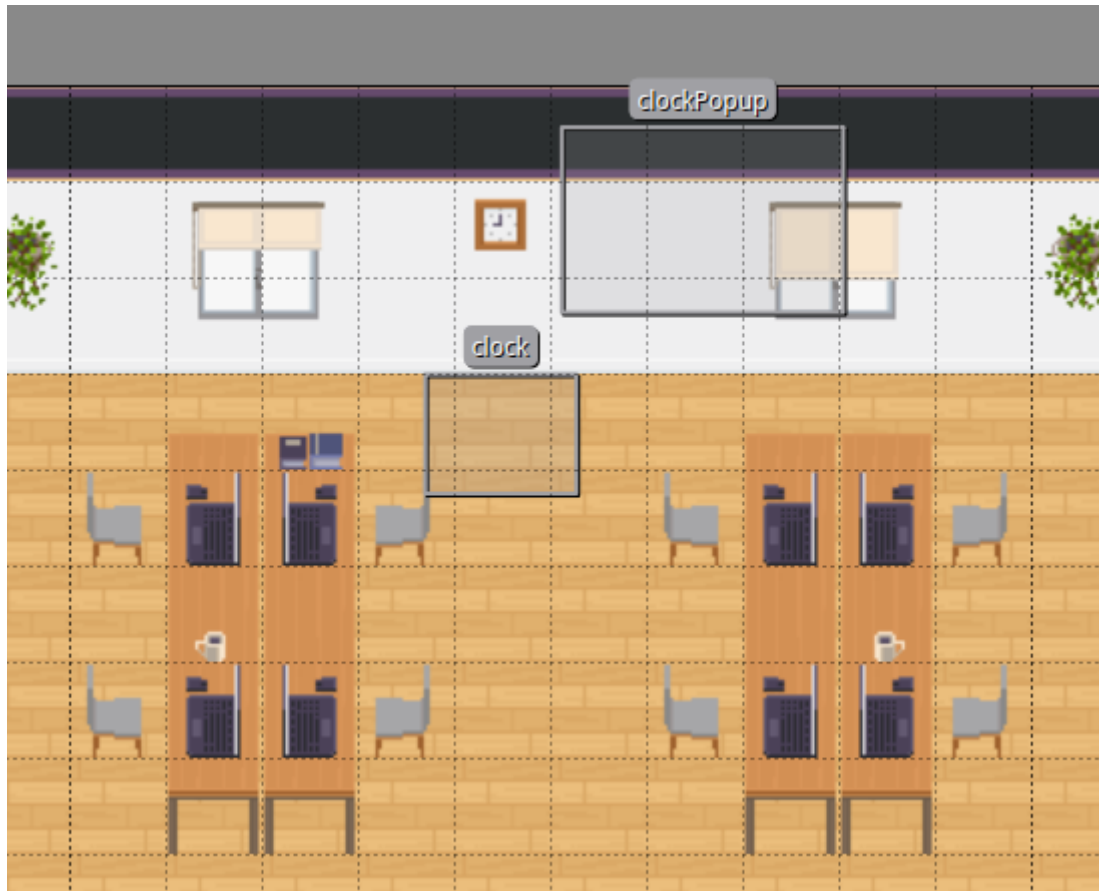
Für die Nutzung der API steht eine Dokumentation zur Verfügung:

Dokumentation: <https://workadventu.re/map-building/api-reference> 30.03.2023

Einige Funktionen, auf die in den Empfehlungen für KW 20 eingegangen wurde, können alternativ über Funktionen der API implementiert werden.

Ein simples Skript, mit dessen Hilfe die Grundlagen der Nutzung der API erlernt werden können, ist im Map Starter Kit implementiert.

In map.tmj wurden zwei Objekte angelegt: „clock“ und „clockPopup“.



Betreten TeilnehmerInnen das Areal „clock“ öffnet sich im Areal „clockPopup“ ein Popup mit der aktuellen Uhrzeit.



Beide Objekte verfügen über keine Eigenschaften.

mapCopyright	Credits: WorkAdventure (https://W...)
mapDescription	A perfect virtual office to get start...
mapImage	map.png
mapName	Office
script	src/main.ts

Als Eigenschaft wurde für die Mapdatei „script“ mit einem Verweis auf main.ts gesetzt.

```

/// <reference types="@workadventure/iframe-api-typings" />

// import auf diese Weise bei Verwendung von bundler (wie bei Starter Kit)

// bootstrapExtra initialisiert custom properties
import { bootstrapExtra } from "@workadventure/scripting-api-extra";

console.log('Script started successfully!');

let currentPopup: any = undefined;

// Waiting for the API to be ready
WA.onInit().then(() => {
  console.log('Scripting API ready');
  console.log('Player tags: ', WA.player.tags)

  WA.room.area.onEnter('clock').subscribe(() => {
    const today = new Date();
    const time = today.getHours() + ":" + today.getMinutes();
    currentPopup = WA.ui.openPopup("clockPopup", "It's " + time, []);
  })

  WA.room.area.onLeave('clock').subscribe(closePopup)

  // The line below bootstraps the Scripting API Extra library
  //that adds a number of advanced properties/features to WorkAdventure
  bootstrapExtra().then(() => {
    console.log('Scripting API Extra ready');
  }).catch(e => console.error(e));
}).catch(e => console.error(e));

function closePopup(){
  if (currentPopup !== undefined) {
    currentPopup.close();
    currentPopup = undefined;
  }
}

export {};

```

Die Zeilen 1, 6, 27-29 stellen den Zugriff auf die API her.

Von besonderem Interesse sind für das Verstehen der Implementation der Uhr die Zeilen 17-21 und 23.

In Zeile 17-21 wird, sobald ein Avatar das Areal mit dem identifier „clock“ betritt ein subscribe für die anonyme Funktion in den Zeilen 18-20 ausgeführt.

Dort wird die aktuelle Zeit als String zusammengefügt und ein Popup im Areal des Objekts „clockPopup“ geöffnet.

Zeile 23 umfasst ein subscribe auf die Funktion closePopup, die das generierte Popup schließt.

(a) Implementieren Sie folgendes Verhalten:

Hinweis: Nutzen Sie das Map Starter Kit als Basis, um Probleme bei der Integration von Skripten zu minimieren. Integrieren Sie Ihre Funktionen in die vorhandene main.ts

TeilnehmerInnen sollen einen Bereich Ihrer Map betreten und verlassen. Bei jeder dieser beiden Aktionen soll eine Chatnachricht erscheinen. In der Nachricht soll der Name des Avatars vorkommen.

Hinweis: Sie benötigen eine benannte Ebene, die in dem Bereich, in dem das Verhalten stattfinden soll, definiert ist (e.g. Tiles enthält).

- I. Erstellen Sie eine Ebene, platzieren Sie einige Tiles. Benennen Sie die Ebene erkennbar.
- II. Nutzen Sie die room Funktionen der API, um eine Funktion zu schreiben, die aufgerufen wird, sobald ein Avatar die Ebene (e.g. die Tiles auf der Ebene) betritt. Nutzen Sie die chat und player Funktionen, um eine Nachricht in den Chat zu schreiben.
- III. Schreiben Sie analog eine Funktion, die beim Verlassen der Ebene aufgerufen wird.

Dokumentation: <https://workadventu.re/map-building/api-room.md> 30.03.2023

Dokumentation: <https://workadventu.re/map-building/api-player.md> 30.03.2023

Dokumentation: <https://workadventu.re/map-building/api-chat.md> 30.03.2023

Dokumentation: <https://rxjs.dev/guide/observable> 30.03.2023

(b) Implementieren Sie folgendes Verhalten, modifizieren Sie Ihr Ergebnis aus Teilaufgabe (a):

Ein Charakter soll beim Betreten eines Bereichs der Map erscheinen.
Suchen Sie ein Charakterasset und ein Soundasset.
Geben Sie dem Charakter einen Namen, falls dieser noch keinen ausgehend von der Quelle besitzt.

Modifizieren Sie das Tileset so, dass im WA Menü im Browser Copyrightinformationen angezeigt werden. Legen Sie eine Ebene an und setzen Sie die Eigenschaften so, dass Copyrightinformationen angezeigt werden.

Hinweis: Vgl. Übung KW 19 und <https://workadventu.re/map-building/wa-maps.md> 30.03.2023

Implementieren Sie das Verhalten:
TeilnehmerInnen betreten einen Bereich Ihrer Map.

Platzieren Sie das Tile des ausgewählten Charakters in der Nähe des Avatars.

Spielen Sie für wenige Sekunden das ausgewählte Soundasset ab.

Schreiben Sie eine Nachricht in den Chat. Sie soll von dem Charakter stammen (Name) und den Namen des Avatars enthalten.

Beim Verlassen des Bereichs soll der Bereich zurückgesetzt und eine weitere Nachricht in den Chat geschrieben werden.

Dokumentation: <https://workadventu.re/map-building/api-room.md> 30.03.2023

Dokumentation: <https://workadventu.re/map-building/api-sound.md> 30.03.2023

Dokumentation: <https://workadventu.re/map-building/api-chat.md> 30.03.2023