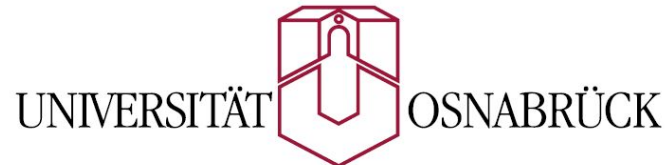


Computergrafik Praktikum 2016

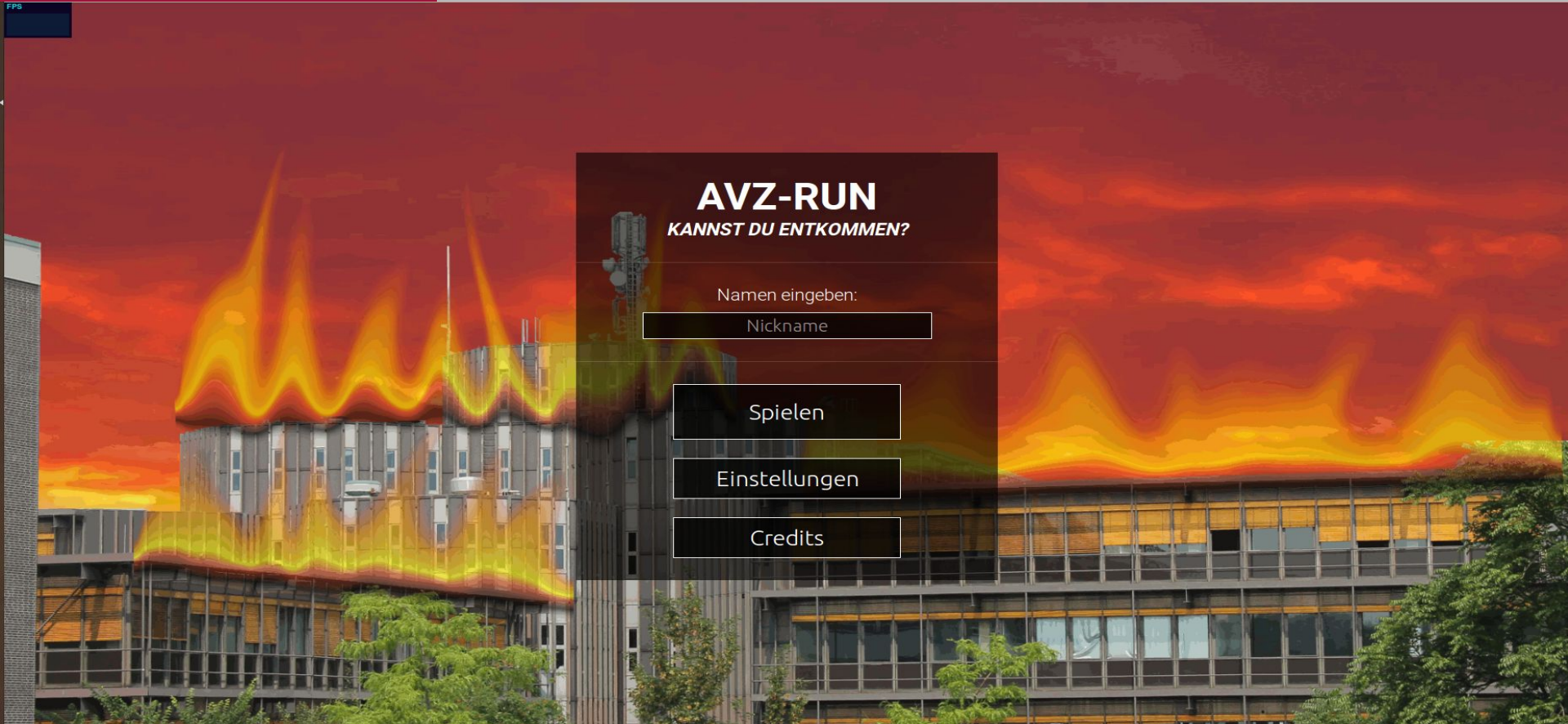
Gruppe A

AVZ-Run

Kannst du entkommen?



FPS



AVZ-RUN

KANNST DU ENTKOMMEN?

Namen eingeben:







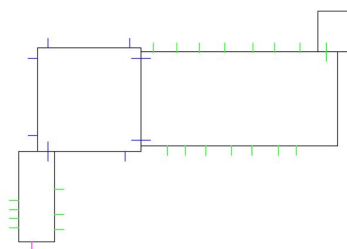








LevelEditor



```

<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<level>
  <rooms>
    <room index="0" name="center" rotation="0.0" x="400.0" y="320.0">room</room>
    <room index="1" name="buero" rotation="0.0" x="628.0" y="284.0">room</room>
    <room index="2" name="klogang_solo" rotation="0.0" x="334.0" y="431.0">room</room>
    <door father="0" normx="0.0" normy="1.0" type="glass" x="404.77" y="334.0">door</door>
    <door father="0" normx="1.0" normy="0.0" type="glass" x="408.0" y="331.77">door</door>
    <door father="0" normx="1.0" normy="0.0" type="glass" x="408.0" y="314.23">door</door>
    <door father="0" normx="0.0" normy="-1.0" type="glass" x="405.77" y="312.0">door</door>
    <door father="0" normx="0.0" normy="-1.0" type="glass" x="388.23" y="312.0">door</door>
    <door father="0" normx="-1.0" normy="0.0" type="glass" x="386.0" y="314.23">door</door>
    <door father="0" normx="-1.0" normy="0.0" type="glass" x="386.0" y="330.77">door</door>
    <door father="0" normx="0.0" normy="1.0" type="glass" x="388.23" y="334.0">door</door>
  </rooms>
  <spawns/>
  <lights/>
  <fires/>
</level>

```

Toilettenflur

Clear

Speichern & Beenden

Rückgängig

- Level besteht aus Räumen und Wegen
- Ein Raum besitzt Position, Winkel und Wege
- Bei Klick im Editor werden Wege verglichen
 - Raum wird ggf versetzt
 - Raum und seine Wege werden ins Level geschrieben
- Räume werden aus einer XML geladen
 - Wir speichern nur Name, Position und Winkel

Ursprung

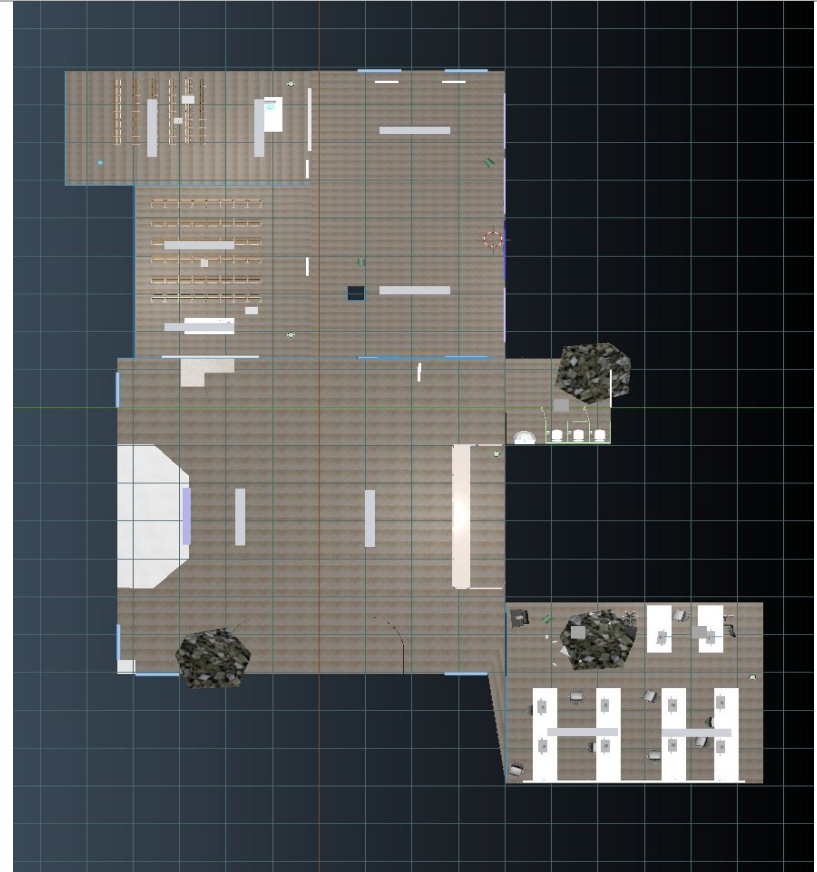
- >Notwendigkeit eines spielbaren Levels
- >Unabhängigkeit von Editor

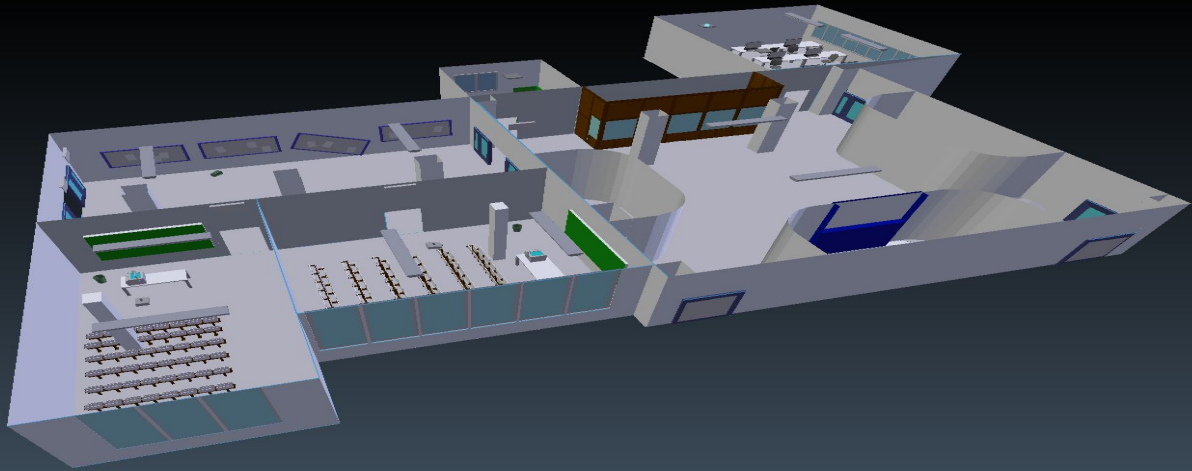
Aufbau des Grundgerüsts

- >Anpassung von Modellen
- >Einbindung von nicht interaktiven Objekten
- >Vereinigen zu einem großen Objekt

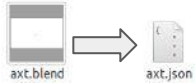
Spielablauf

- > Mitgestaltung und Umsetzung des Leveldesign
- >Übergang zu einer spielbaren Demo

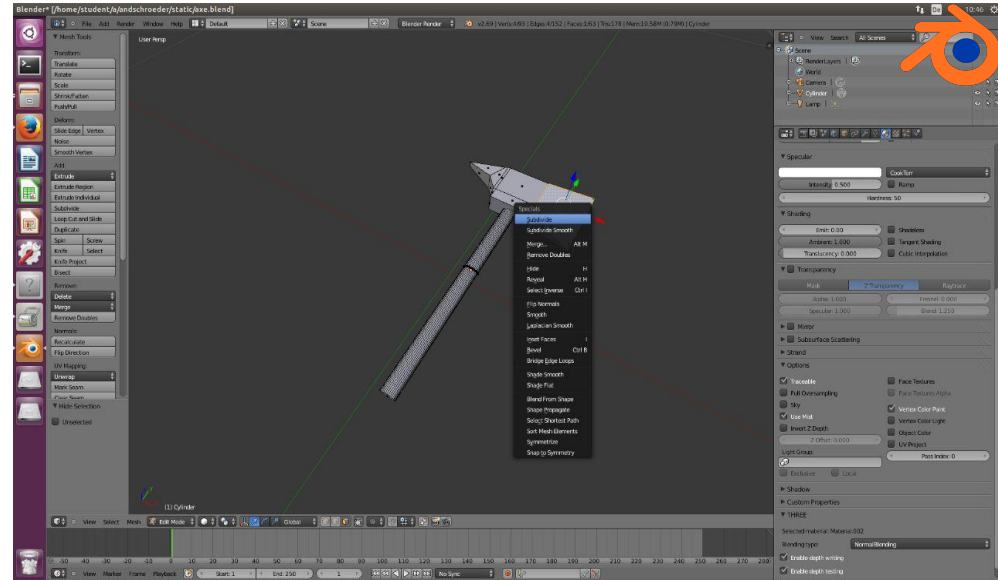




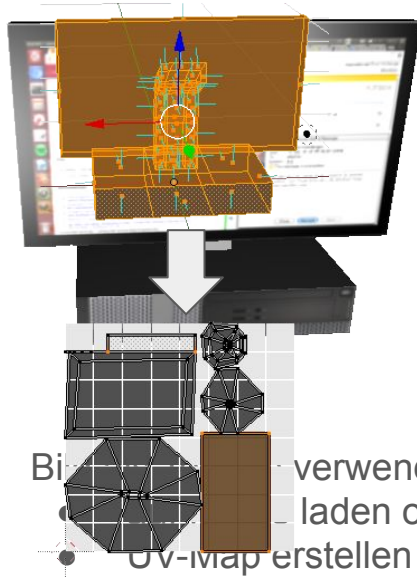
- Alle Modelle mit Blender erstellt
 - Vorteil: exportieren in three.js kompatible json-Dateien möglich



- einige Beispiele:
 - Item
 - Axt, Feuerlöscher, etc.
 - interaktive Objekte
 - PCs, Türen, etc
 - statische Objekte
 - Lichter, Plakate, etc
 - Räume
 - Flure, Büros, etc

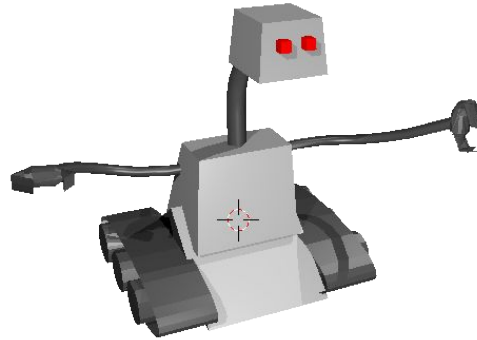


Verschiedene Möglichkeiten für Texturen:



Bilder verwenden
 laden oder
 UV-Map erstellen

Kombinationen möglich



Materials einfärben

- Material besitzt u. A. Farbe

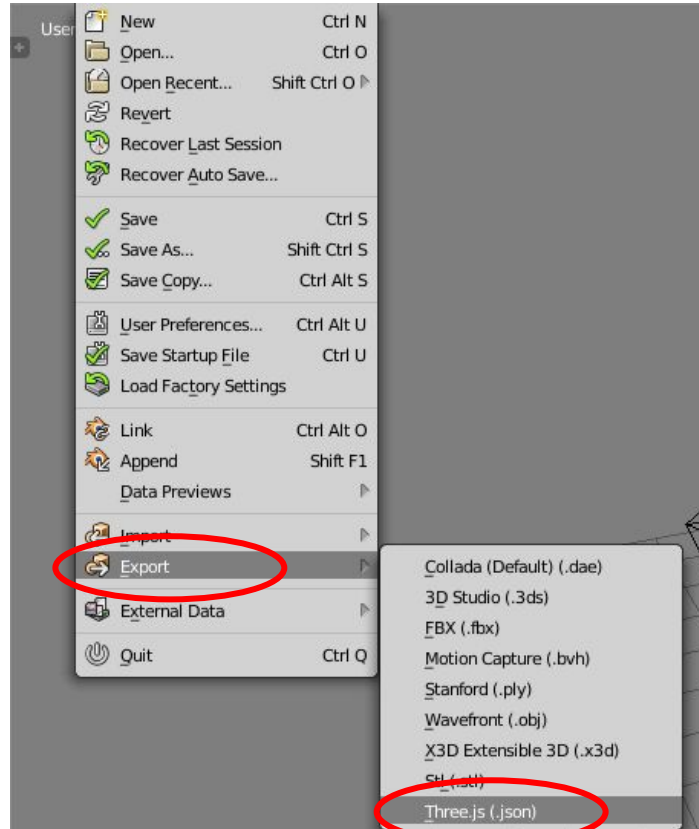
Objekte mit individuellen Texturen



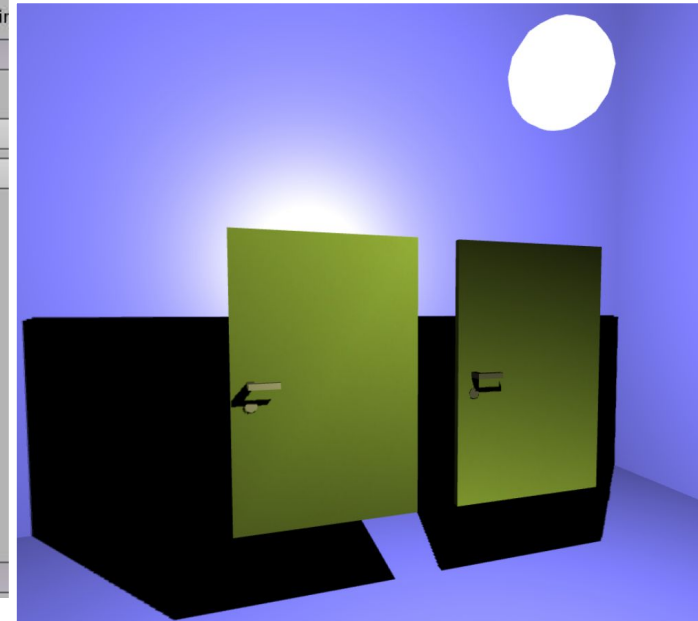
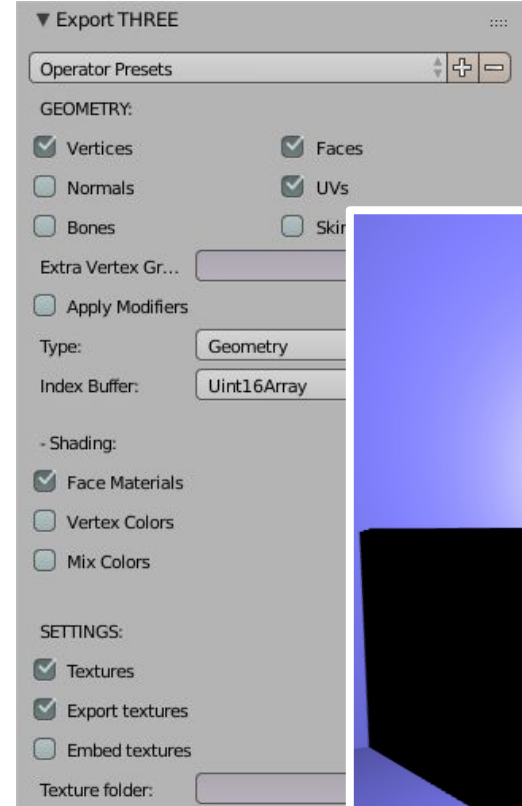
Vertices Färben

- Face-Farben werden interpoliert

- Modelle im JSON Format
- Export Plug-ins auf GitHub
 - Blender
 - 3D Studio Max
 - Maya
 - Autodesk Revit
- Objekt in Szene auswählen
 - Immer nur 1 Objekt
 - ggf. mergen



- Face Materials
 - verschieden farbige Materialien
- Vertex Colors
 - wenn Vertex Painter genutzt wurde
- Textures, Export Textures
 - wenn Texturen vorhanden
- Normals
 - nicht mit exportiert
 - Beleuchtungsprobleme

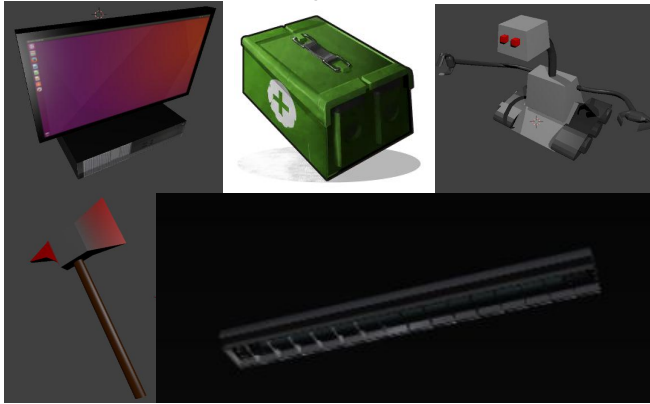


- Vertices, Faces, UVs
- Materials
 - shading
 - basic
 - lambert
 - phong
 - opacity
 - mapDiffuse: texture
 - mapDiffuseRepeat
 - clamb
 - repeat

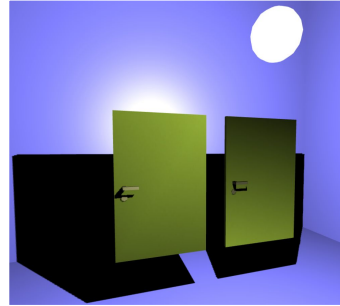
```
{
  "name": "Wing.005Geometry",
  "vertices": [-1.41501, 1.275, 0.0349969, -1.41502, 1.225, 0.134997, -1.41501, 1.225, 0.0349969, -1.41502, 1.225, 0.134997, -1.41501, 1.225, 0.0349969, -1.41502, 1.225, 0.134997],
  "uvs": [[0.0412482, 0.685709, 0.0704116, 0.713862, 0.144611, 0.754689, 0.149458, 0.718161, 0.0412482, 0.685709, 0.0704116, 0.713862, 0.144611, 0.754689, 0.149458, 0.718161],
  "faces": [10, 0, 5, 4, 0, 0, 1, 2, 10, 5, 6, 4, 0, 1, 3, 2, 10, 7, 1, 6, 0, 4, 5, 3, 10, 8, 9, 10, 0, 6, 7, 8, 10],
  "materials": [{
    "mapDiffuseRepeat": [1.001, 1.001],
    "transparent": false,
    "opacity": 1,
    "blending": "NormalBlending",
    "mapDiffuse": "door_tex2.jpg",
    "DbgIndex": 1,
    "DbgColor": 15597568,
    "colorDiffuse": [0.64, 0.64, 0.64],
    "colorEmissive": [0, 0, 0],
    "mapDiffuseAnisotropy": 1,
    "DbgName": "door_mat",
    "mapDiffuseWrap": ["RepeatWrapping", "RepeatWrapping"],
    "depthTest": true,
    "depthWrite": true,
    "visible": true,
    "shading": "phong",
    "wireframe": false
  }],
  "metadata": {"generator": "io_three"...}
}
```

Wir haben:

Objekte



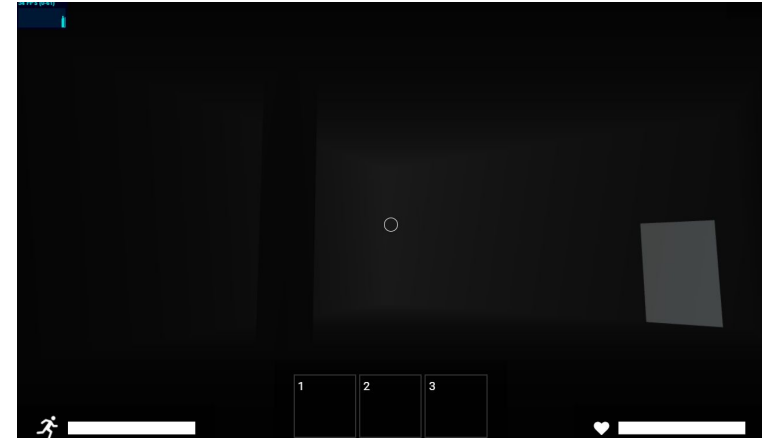
Türen



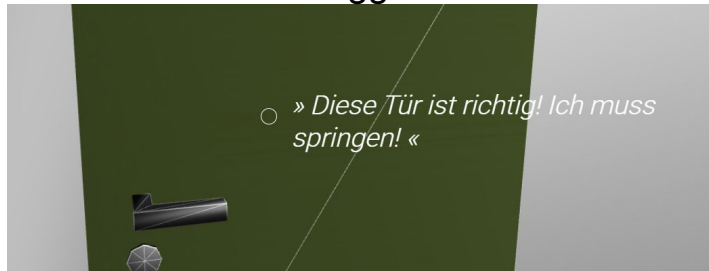
Feuer



Räume



Trigger





Bauen wir einen Raum...





```

<room name="lecturerroomleer" filename="lecturerroom1.json" smooth="0" fogdensityincrease="1">
  <origin x="0" y="0"/>
  <size xmin="-2" ymin="-2" xmax="12" ymax="8" />
  <doors>
    <door type="norm" index="1" position="(12,-0.75)" normal="(1,0)" act="codeopenable" stretch="1" />
  </doors>
  <spawns>
    <spawn index="1" position="(10, 4, 2.2)" object="axt" normaltowall="(1,0)" scale="3" oninteract="pickUpItem"/>
    <spawn index="2" position="(10, 5, 0)" object="transponder" normaltowall="(1,0)" scale="2" oninteract="pickUpItem"/>
    <spawn index="2" position="(10, 7, 1)" object="computer" normaltowall="(0,-1)" scale="1" oninteract="enterCH"/>
    <spawn index="2" position="(11.8, 0.3, 1.8)" object="pinpad" normaltowall="(0,-1)" scale="1" oninteract="enterPin"/>
    <spawn index="stuhl" position="(4 to 8 step 1, 1 to 7 step 0.6, 0)" object="pult" normaltowall="(0,1)" scale="1"/>
  </spawns>
  <fires>
    <fire index="11" position="(9,0,0)" size="(1,1,2)" val="10" />
  </fires>
  <lights>
    <light index="1" kind="pointlight" objectname="deckenlicht" position="(9,3,5)" intensity="0.7" color="0xFFFFFFFF" objectscale="1" normal="(0,1)" visiblewidth="14"/>
    <light index="2" kind="pointlight" objectname="deckenlicht" position="(1,3,5)" intensity="0.7" color="0xFFFFFFFF" objectscale="1" normal="(0,1)" visiblewidth="14"/>
  </lights>
  <triggers>
    <trigger index="exit" xpos="12" ypos="0" size="3" functionname="showThoughts" fparam1="Hinter dieser Tür ist die Treppe" fparam2="5000"/>
    <trigger index="exit2" xpos="13.5" ypos="-1" size="2" functionname="nextLevel"/>
    <trigger index="startgedanke" xpos="2" ypos="2" size="2" functionname="showThoughts" fparam1="Wie bin ich denn hier gelandet?" fparam2="5000"/>
  </triggers>
</room>
  
```

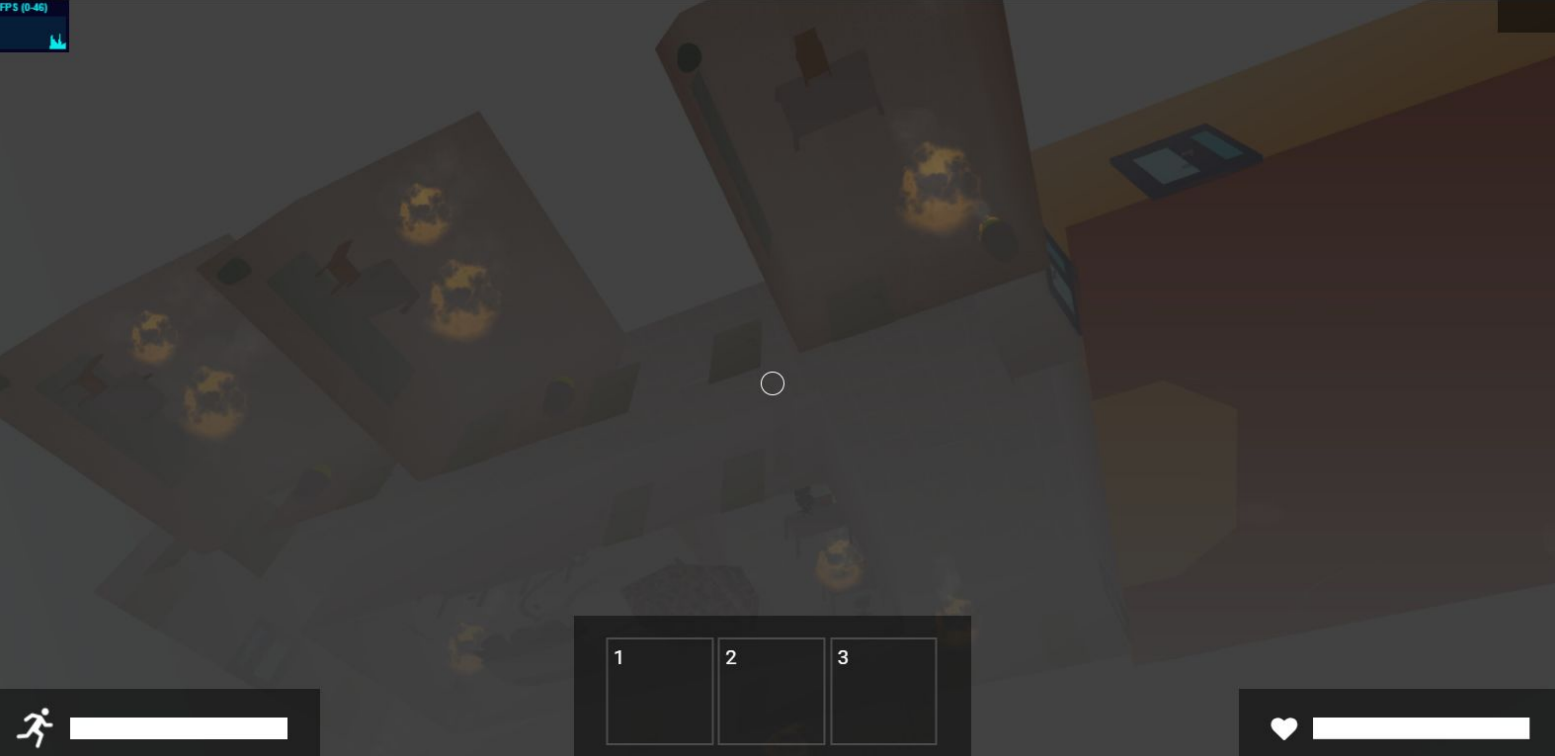
selbe Datei, verschiedene Räume!

```

<door type="norm" index="18" position="(-8.98,8.6)" normal=(-1, 0) act="openable"/>
<door type="norm" index="19" position="(-8.98,10.1)" normal=(-1, 0) act="openable"/>
<door type="norm" index="20" position="(-8.98,15.6)" normal=(-1, 0) act="openable"/>
<door type="norm" index="17" position="(-8.98,19.87)" normal=(-1, 0) act="openable"/>
<door type="norm" index="18" position="(-7.18,23.24)" normal=(0, 1) act="autopenable"/>
<door type="glass" index="19" position="(-1.72,23.24)" normal=(0, 1) act="openable"/>
<door type="norm" index="20" position="(3.18,23.24)" normal=(0, 1) act="openable"/>
<door type="norm" index="21" position="(-3.78,2.88)" normal=(1, 0) act="autopenable"/>
<door type="norm" index="22" position="(-3.78, -1.88)" normal=(1, 0) act="closed"/>
</doors>
<lights>
<light index="1" kind="pointlight" objectname="deckenlicht" position="(6.12,5.8)" intensity="0.8" color="0xFFFFFFFF" objectscale="1" normal=(0,1) viewlinewidth="33"/>
<light index="2" kind="pointlight" objectname="deckenlicht" position="(8.12,5.8)" intensity="0.8" color="0xFFFFFFFF" objectscale="1" normal=(0,1) viewlinewidth="34"/>
<light index="3" kind="pointlight" objectname="deckenlicht" position="(6,-1.5)" intensity="0.5" color="0xFFFFFFFF" objectscale="1" normal=(0,1) viewlinewidth="34"/>
<light index="4" kind="pointlight" objectname="deckenlicht" position="(8,-1.5)" intensity="0.5" color="0xFFFFFFFF" objectscale="1" normal=(0,1) viewlinewidth="34"/>
<light index="5" kind="pointlight" objectname="deckenlicht" position="(1,20.5)" intensity="0.8" color="0xFFFFFFFF" objectscale="1" normal=(0,1) viewlinewidth="34"/>
<light index="6" kind="pointlight" objectname="deckenlicht" position="(1,20.5)" intensity="0.8" color="0xFFFFFFFF" objectscale="1" normal=(0,1) viewlinewidth="34"/>
</lights>
<files>
<file index="all" position="(7.5, -7, 0)" size="(3,3,3)" val="6"/>
<file index="cl2" position="(1, 14, 0)" size="(3,3,3)" val="6"/>
<file index="cl3" position="(1, -3.8, 0)" size="(3,2,2)" val="6"/>
</files>
<spawns>
<spawn index="1" position="(1.03, 0.3, 0)" object="wand_kaputt" normaltowell="(0,1)" scale="2.18"/>
<spawn index="2" position="(-1, 16, 0)" object="stuhll" normaltowell="(1,0)" scale="1.8"/>
<spawn index="3" position="(2, -3.8, 0)" object="tisch" normaltowell="(0,1)" scale="1"/>
<spawn index="4" position="(-1, -3.8, 0)" object="tisch" normaltowell="(0,1)" scale="1"/>
<spawn index="5" position="(1, 2, 0)" object="stuhll" normaltowell="(1,0)" scale="2"/>
<spawn index="6" position="(0, 2, 0)" object="stuhll" normaltowell="(0,0)" scale="1"/>
<spawn index="7" position="(5.8, 8, 1.8)" object="tafel" normaltowell="(0,1)" scale="2"/>
<spawn index="8" position="(3, -3.8, 1)" object="computer" normaltowell="(1,0)" scale="1" orientat="0"/>
<spawn index="9" position="(4, -3.8, 0)" object="computer" normaltowell="(1,0)" scale="1" orientat="0"/>
<spawn index="10" position="(2, 12, 0)" object="evil_robot" normaltowell="(0,1)" scale="2" orientat="0"/>
<spawn index="11" position="(0.3, 6.4, 0)" object="akt" normaltowell="(1,0)" scale="1" orientat="pickUpItem"/>
<spawn index="12" position="(4, -3.8, 0)" object="bucacatuhl" normaltowell="(1,0)" scale="1"/>
<spawn index="13" position="(2.8, -3.8, 0)" object="bucacatuhl" normaltowell="(1,0)" scale="1"/>
<spawn index="14" position="(3, 7, 0)" object="tisch" normaltowell="(0,1)" scale="1"/>
<spawn index="15" position="(3, 8, 0)" object="tisch" normaltowell="(0,1)" scale="1"/>
<spawn index="16" position="(3, 11, 0)" object="tisch" normaltowell="(0,1)" scale="1"/>
<spawn index="17" position="(3, 6.8, 0.88)" object="robot" normaltowell="(1,0)" scale="0.4" orientat="0"/>
<spawn index="18" position="(3, 2, 8, 1)" object="computer" normaltowell="(1,0)" scale="1" orientat="0"/>
<spawn index="19" position="(3, 10, 1)" object="cbp" normaltowell="(0,1)" scale="2"/>
<spawn index="20" position="(3, 11.8, 0.88)" object="robot" normaltowell="(1,0)" scale="0.4" orientat="0"/>
<spawn index="21" position="(1.8, 1.8, 0)" object="riegel" normaltowell="(0,1)" scale="1.8"/>
<spawn index="22" position="(3, 10, 0)" object="riegel" normaltowell="(0,1)" scale="1"/>
<spawn index="23" position="(1.8, 8, 0)" object="riegel" normaltowell="(0,1)" scale="1"/>
<spawn index="24" position="(3, 18, 0)" object="riegel" normaltowell="(0,1)" scale="1"/>
<spawn index="25" position="(3, 9, 1)" object="riegel" normaltowell="(0,1)" scale="1"/>
<spawn index="26" position="(3, 9, 1.4)" object="pingad" normaltowell="(0,1)" scale="1" orientat="entertain"/>
<spawn index="27" position="(4.8, 17.2, 0)" object="tisch" normaltowell="(1,0)" scale="1"/>
<spawn index="28" position="(4.8, 17.2, 0)" object="computer" normaltowell="(0,-1)" scale="1" orientat="robotControl"/>
</spawns>
<lights>
<light index="1" kind="pointlight" objectname="deckenlicht" position="(6.12,5.8)" intensity="0.5" color="0xFFFFFFFF" objectscale="1" normal=(0,1) viewlinewidth="33"/>
<light index="2" kind="pointlight" objectname="deckenlicht" position="(8.12,5.8)" intensity="0.5" color="0xFFFFFFFF" objectscale="1" normal=(0,1) viewlinewidth="34"/>
<light index="3" kind="pointlight" objectname="deckenlicht" position="(6,-1.5)" intensity="0.5" color="0xFFFFFFFF" objectscale="1" normal=(0,1) viewlinewidth="34"/>
<light index="4" kind="pointlight" objectname="deckenlicht" position="(8,-1.5)" intensity="0.5" color="0xFFFFFFFF" objectscale="1" normal=(0,1) viewlinewidth="34"/>
<light index="5" kind="pointlight" objectname="deckenlicht" position="(1,20.5)" intensity="0.5" color="0xFFFFFFFF" objectscale="1" normal=(0,1) viewlinewidth="34"/>
<light index="6" kind="pointlight" objectname="deckenlicht" position="(1,20.5)" intensity="0.5" color="0xFFFFFFFF" objectscale="1" normal=(0,1) viewlinewidth="34"/>
<light index="7" kind="pointlight" objectname="deckenlicht" position="(1,20.5)" intensity="0.5" color="0xFFFFFFFF" objectscale="1" normal=(0,1) viewlinewidth="34"/>
<light index="8" kind="pointlight" objectname="deckenlicht" position="(1,5.5)" intensity="0.5" color="0xFFFFFFFF" objectscale="1" normal=(0,1) viewlinewidth="34"/>
</lights>
<files>
<file index="all" position="(7.5, -7, 0)" size="(3,3,3)" val="6"/>
<file index="cl2" position="(2.5, 4, 0)" size="(4,7,3)" val="12"/>
</files>
<spawns>
<spawn index="1" position="(-2.2, -4, 0)" object="tisch" normaltowell="(1,0)" scale="1.1"/>
<spawn index="2" position="(1, -4.5, 2)" object="tafel" normaltowell="(1,0)" scale="2"/>
<spawn index="3" position="(1, -1.2, 0)" object="tisch" normaltowell="(1,0)" scale="1"/>
<spawn index="4" position="(1, -1.2, 0)" object="cbp" normaltowell="(1,0)" scale="2"/>
<spawn index="5" position="(-2, -2, 0)" object="muelleimer" normaltowell="(1,0)" scale="2"/>
<spawn index="6" position="(1, -2.5, 0)" object="stuhll" normaltowell="(0,-1)" scale="2"/>
<spawn index="7" position="(1, -3.5, 0)" object="loscher" normaltowell="(0,-1)" scale="1" orientat="pickUpItem"/>
<spawn index="8" position="(0, 15, 0)" object="akt" normaltowell="(1,0)" scale="1.5" orientat="pickUpItem"/>
<spawn index="stuhll" position="(-1.2 to 5.4 step 0.6, 2 to 14 step 1, 0)" object="pult" normaltowell="(1,0)" scale="1"/>
</spawns>
</room>

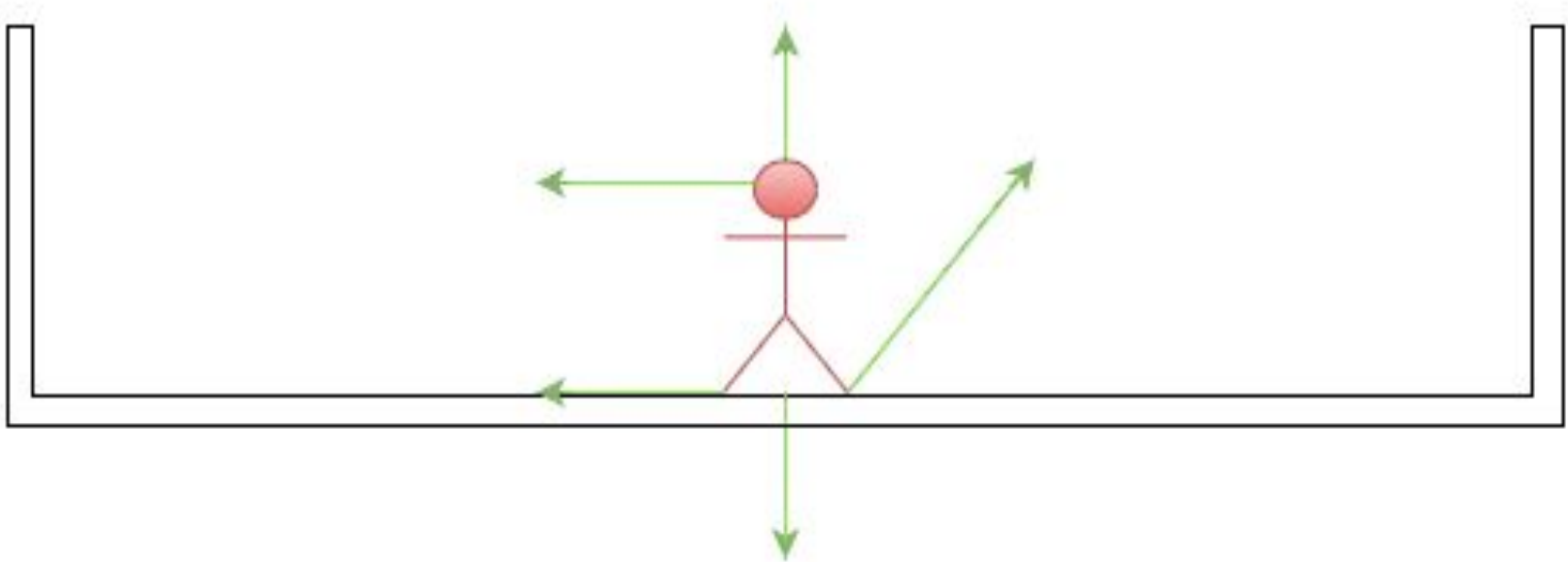
```

...bauen wir nun Stockwerke!



```
LevelEditor  
  
<?xml version="1.0" encoding="UTF-8" standalone="no"?>  
<level>  
<rooms>  
<room index="0" name="center" rotation="0.0" x="400.0" y="320.0">room</room>  
<room index="1" name="buero" rotation="0.0" x="628.0" y="284.0">room</room>  
<room index="2" name="Klogang_solo" rotation="0.0" x="334.0" y="311.0">room</room>  
<door father="0" normx="0.0" normy="1.0" type="glass" x="466.77" y="314.0">door</door>  
<door father="0" normx="1.0" normy="0.0" type="glass" x="408.0" y="331.77">door</door>  
<door father="0" normx="1.0" normy="0.0" type="glass" x="468.0" y="314.23">door</door>  
<door father="0" normx="0.0" normy="-1.0" type="glass" x="405.77" y="312.0">door</door>  
<door father="0" normx="0.0" normy="-1.0" type="glass" x="388.23" y="312.0">door</door>  
<door father="0" normx="-1.0" normy="0.0" type="glass" x="388.0" y="314.23">door</door>  
<door father="0" normx="-1.0" normy="0.0" type="glass" x="388.0" y="330.77">door</door>  
<door father="0" normx="0.0" normy="1.0" type="glass" x="388.23" y="314.0">door</door>  
</>  
</>  
</>
```

```
="0.012" correcttransponder="43">  
  
0.015" correctpin="0042">
```



4 Bestandteile:

Flamme:

- Kein Partikelsystem sondern Sampling eines Bildes
- Flammen selber Größe bestehen aus geklonten meshes einer einzelnen Feuerinstanz
- Gute Performance da nur diese Instanz Berechnungen durchführt

Rauch :

- Partikelsystem, da geringere Komplexität als bei Flamme

Pointlight:

- Veränderung der Intensität bringt das Feuer zum flackern

Kollisionsbox:

- Benötigt für die Interaktion mit Feuerlöscher und Schaden bei Berührung



Aspekt:

- Möglichst realistische Darstellung einer Rauchentwicklung
- Beeinflussung der Spieldynamik

Umsetzung: Einbindung mit Three.Fog

```
scene.fog = new THREE.FogExp2(0x424242, 0.00002 + myfog);
```

Erweiterung:

- Der Nebel wird mit fortlaufender Zeit zunehmen und die Sicht sehr einschränken.
- Jeder Raum wird unterschiedlich schnell mit Nebel gefüllt

Nebel (Rauchentwicklung)



Raum anfangs ohne Rauch

Raum halb mit Rauch gefüllt

Raum komplett mit Rauch gefüllt

(Nullsicht)

- interagierbare Objekte werden durch Raycasting erkannt und mit Wireframe-Mesh markiert
- wie können wir interagierbare Objekte von Terrain unterscheiden?:

```
GameObject = function(mesh, interaction, type, name)
```

- `mesh`: ursprünglicher Mesh, um den das GameObject gewrappt wurde
- `interaction`: Übergabe beliebiger Funktionen (z.B. Türen öffnen, Objekte aufsammeln, Feuer löschen, mit Computer interagieren)
- `type`: Unterscheidung verschiedener Typen (normale Interactables, Feuer, Ausgang oder Trigger)
- `name`: zur Abfrage von speziellen Objekttypen, die selten vorkommen (z.B. Transponder, Schwamm, ...)

- z.B. *Pin Pad & Transponder Hack*
- JavaScript spricht CSS an
→ kann HTML-Elemente einblenden/wieder ausblenden
- unsichtbare HTML-Elemente werden über die Textur gelegt und mit entsprechenden interact-Funktionen verknüpft
- diverse Teile des Haupt-Loops des Spiels werden temporär deaktiviert, während andere weiterlaufen müssen



Wichtige Aktionen, welche mit Sounds unterstützt werden müssen:

- Laufen / Sprinten
- Feuer / Rauch
- Öffnen und Schließen von Türen
- Tiefes Einatmen nach zu langem Sprinten
- Husten im Rauch
- und viele weitere Aktionen

→ Möglichst **“realistischer”** Klang und Einsatz von Sounds

→ Synchronität z.B. bei Fußschritten sehr wichtig

Einfache Einbindung von Sounds mit ThreeJS:

```
var audioListener = new THREE.AudioListener();  
camera.add(audioListener);  
var audioLoader = new THREE.AudioLoader();  
var sound = new THREE.Audio(audioListener);
```

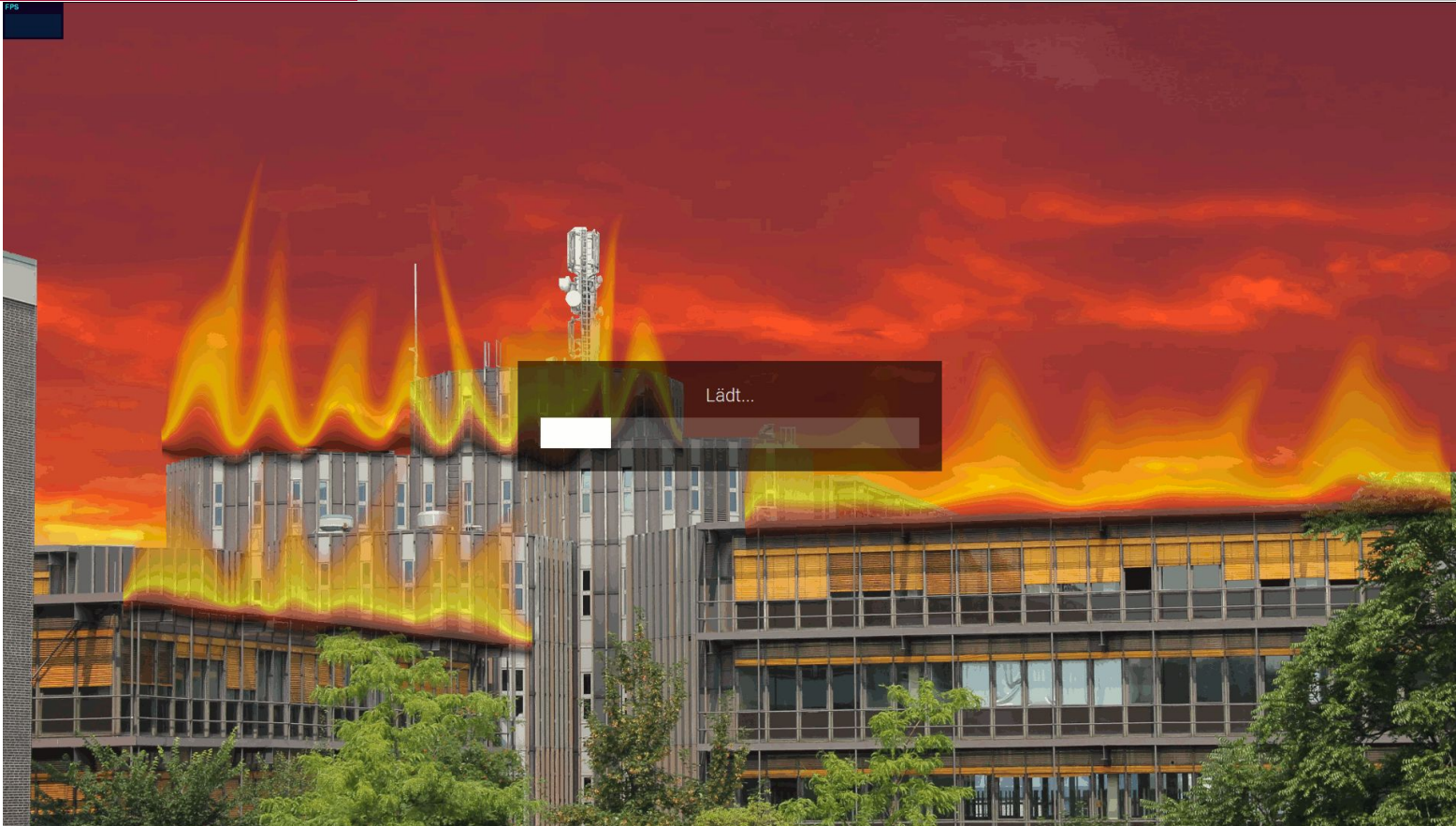
- Audio-Support basierend auf der [“Web Audio API”](#)
- Jeder Sound ist ein eigenes Audio-Objekt
- Unterstützung vom MP3-Format in Firefox, Chrome und Microsoft Edge

3D-Audio-Positionierung

- Mit **THREE.PositionalAudio()** sehr einfach umsetzbar
- Audio-Objekte werden Meshs hinzugefügt und haben dann **eine feste Position im Raum**

Beispiel **Feuer**:

- Das Audio-Objekt wird direkt beim Erstellen des Feuers generiert
- Je weiter man sich vom Feuer entfernt, desto leiser wird der Sound

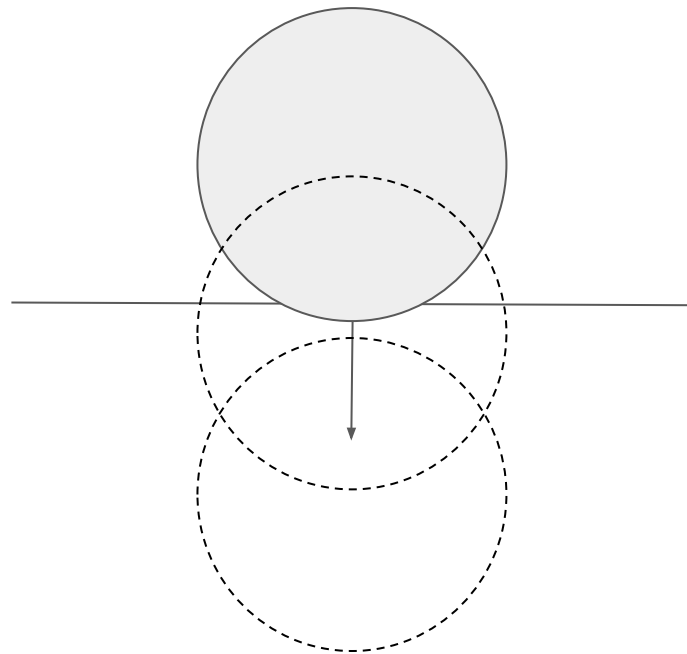
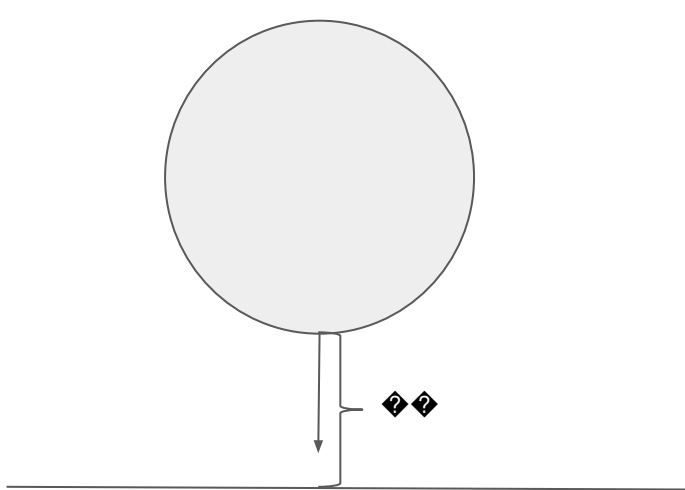




Animation durch moveObject-Methode realisiert, folgende Übergaben werden gebraucht:

- Mesh
- StartPosition
- Strecke
- Geschwindigkeit
- Zeitdifferenz





Problem:	Lösung:
<ul style="list-style-type: none"> - alle Objekte werden in jedem Frame auf Kollisionen geprüft 	<ul style="list-style-type: none"> - speichere Objekte in Octree - Datenstruktur
<ul style="list-style-type: none"> - Feuer enthalten Pointlights - bei benutztem Phong-Material sehr rechenintensiv - Beleuchtung: Ursprünglich viele Pointlights geplant 	<ul style="list-style-type: none"> - Ändern der Materialien in Lambert-Material (wo möglich) - leider: Verzicht auf komplexe Beleuchtung
<ul style="list-style-type: none"> - initiale Ladezeit beim starten des Spiels 	<ul style="list-style-type: none"> - alle Models vor dem Rendern vorladen - Laden der Models in den Grafikspeicher leider nicht erfolgreich