

strange galactic friends

Von Konstantin Hirschberger, Jonas Hoffmann,
Luca Köster und Jannis Gollwitzer



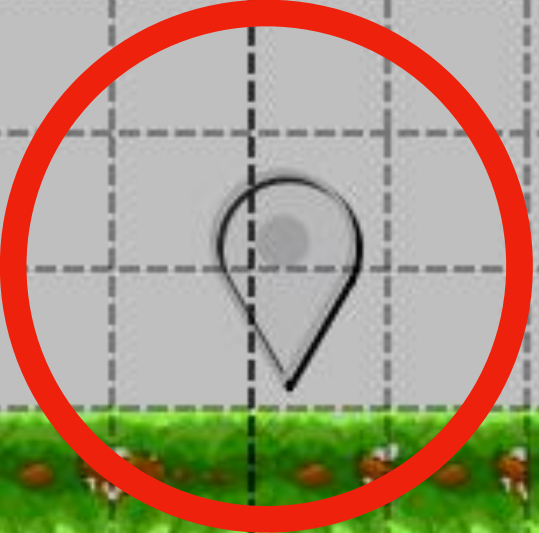
2,5 Minuten

World Design

- Pixel-Arts / Tiles erstellt mit Aseprite
- Level-Design erstellt mit Tiled
 - direkt-Import in Phaser aus Tiled JSON



major_dom

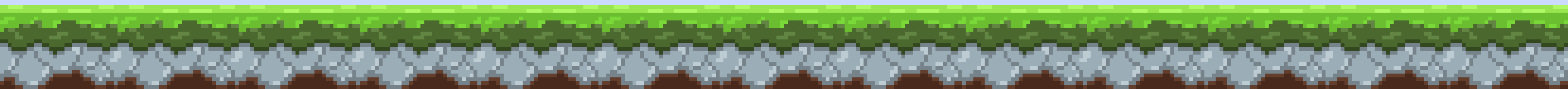


moot

2,5 Minuten

World Design

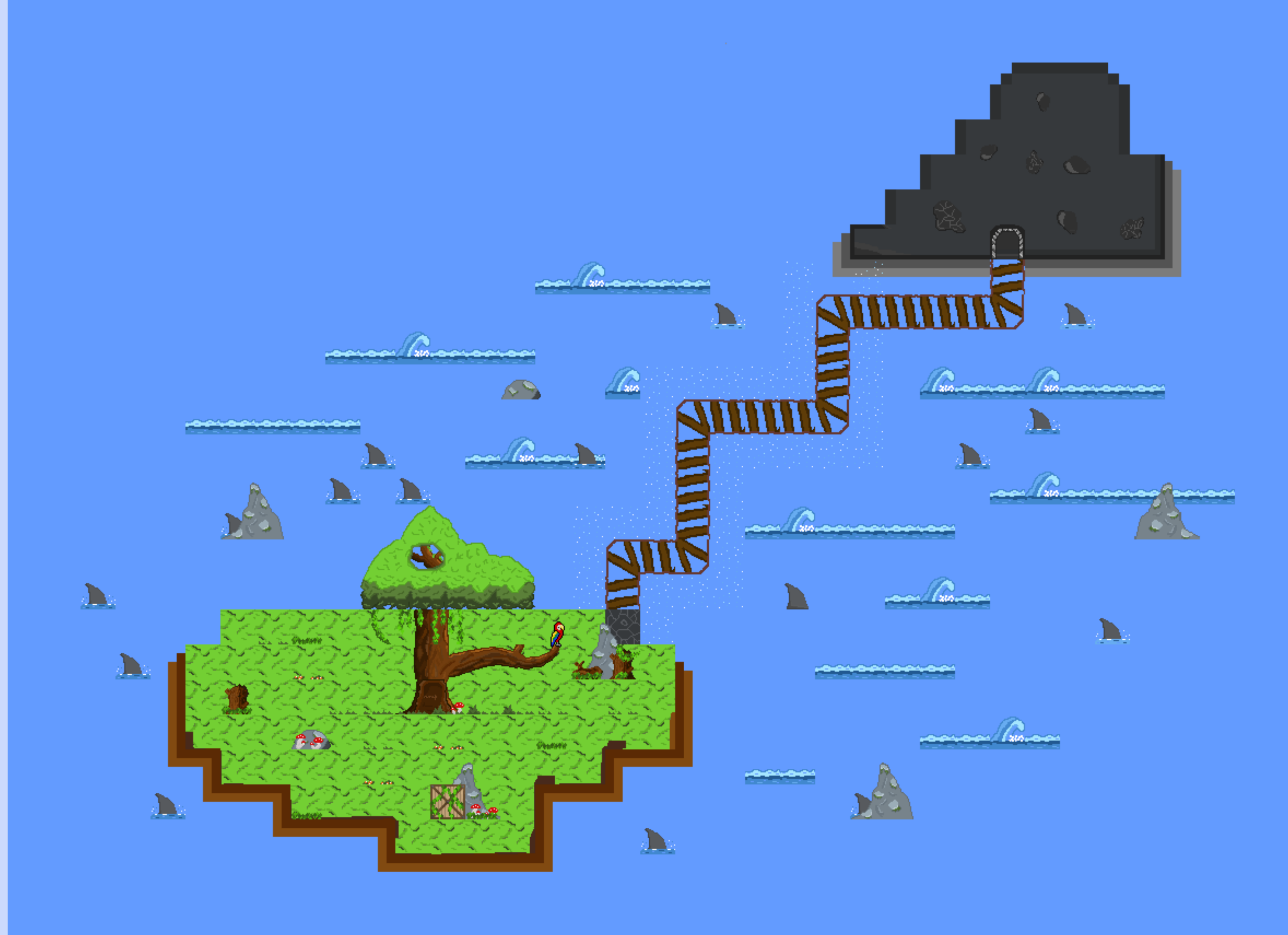
- **Pixel-Arts / Tiles** erstellt mit Aseprite
- **Level-Design** erstellt mit Tiled
 - **direkt-Import** in Phaser aus Tiled JSON
 - **Start-Punkte** der Charaktere
 - **Platzierung** von Spiel-Elementen
 - **Druckplatten**
 - **Boxen**
 - **Collectibles**



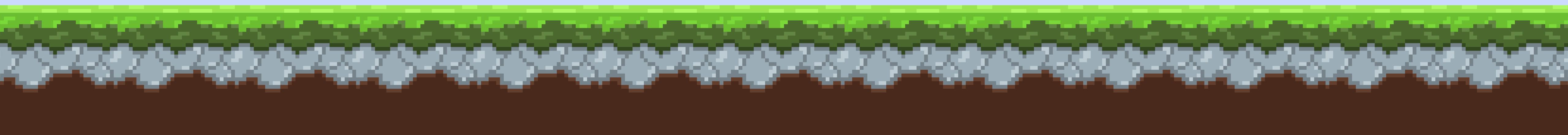
2,5 Minuten

World Map

- mehrere Inseln auf einem Planeten
- jede Insel stellt ein Level dar
- jedes Level Theme schon hier erkennbar



Character Design



2,5 Minuten

Character Design

Moot

- traurig
- mysteriös
- scheint sauer (ist es nicht)



Major Dom

- selbstbewusst
- freundlich
- klar, strukturiert
- clever, schlau



2,5 Minuten

Character Design

- In-Game: Pixel-Art
- Art-Driven Character Design
--> Charaktere wandeln sich nicht während Spielverlauf
- Humanoid Character Design: physisch ähnlich zu Menschen
- Zero-Dimensional Characters
nur diskrete Emotionalität: Character Emotionen unwichtig für Spielverlauf



Major Dom v1

- inkonsistente Linienstärken und Schatten
- Body (Stärke): dickere Linien
- Kopf (klare Struktur): dünnere Linien
- 2. Auge rechts macht MD zu "Standard" Alien -> entfernen

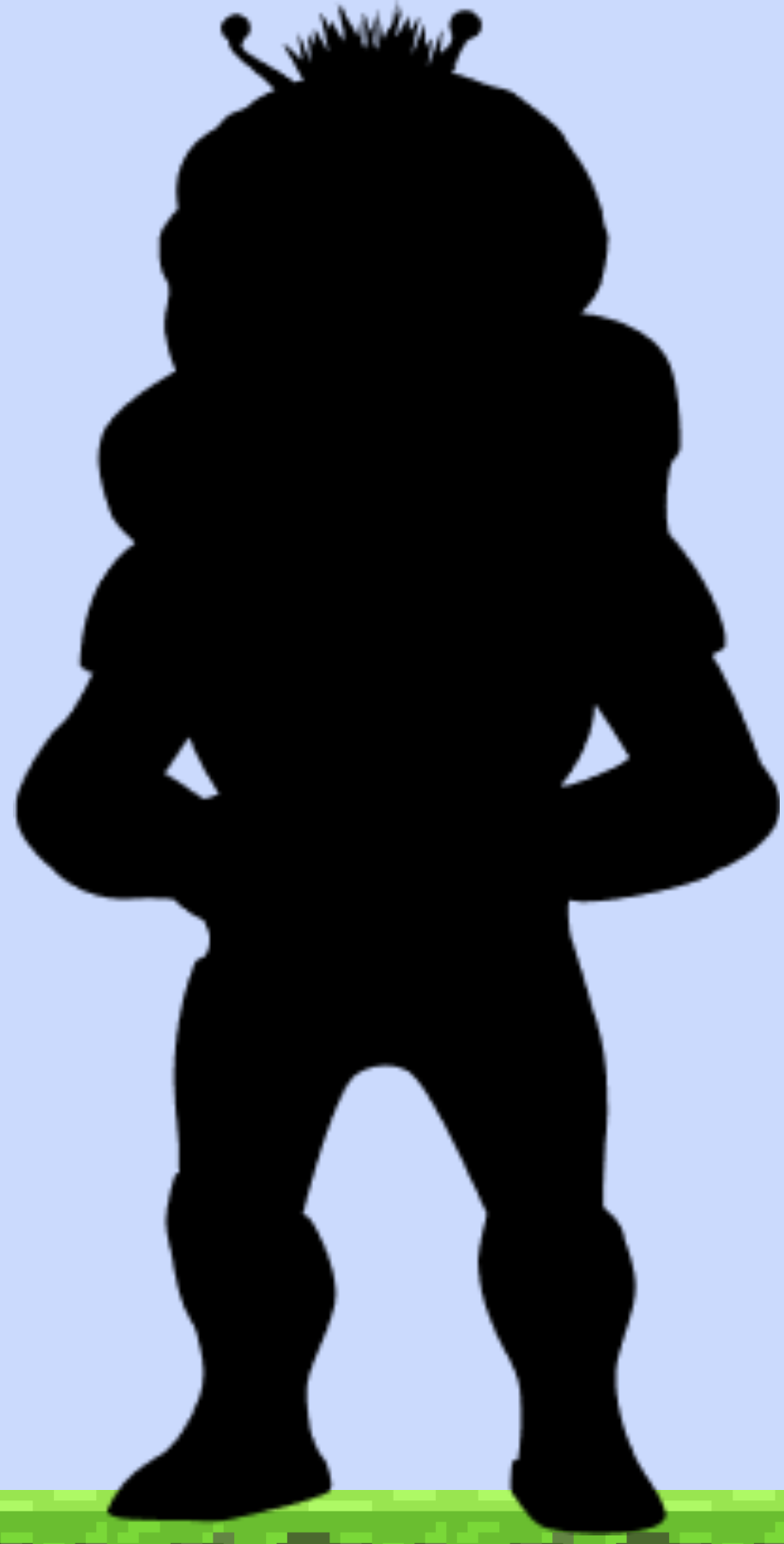


Major Dom v2

- **Space Suit:**
dunklerer Anzug, hellere Akzente, konsistente Schatten, mehr (Komplementär-) Kontrast, Akzente "leuchten"
- zweites Auge entfernt
- Grundform (männlich) Dreieck --> Oberkörper (wirkt erwachsen)



Major Dom – Silhouette



Major Dom v1

- Körper in Dreiecksform
- großer Kopf
- einzelne Teile wenig erkennbar

Major Dom v2

- Antennen größer
- Arm ausgebreitet
- Haare = Detail

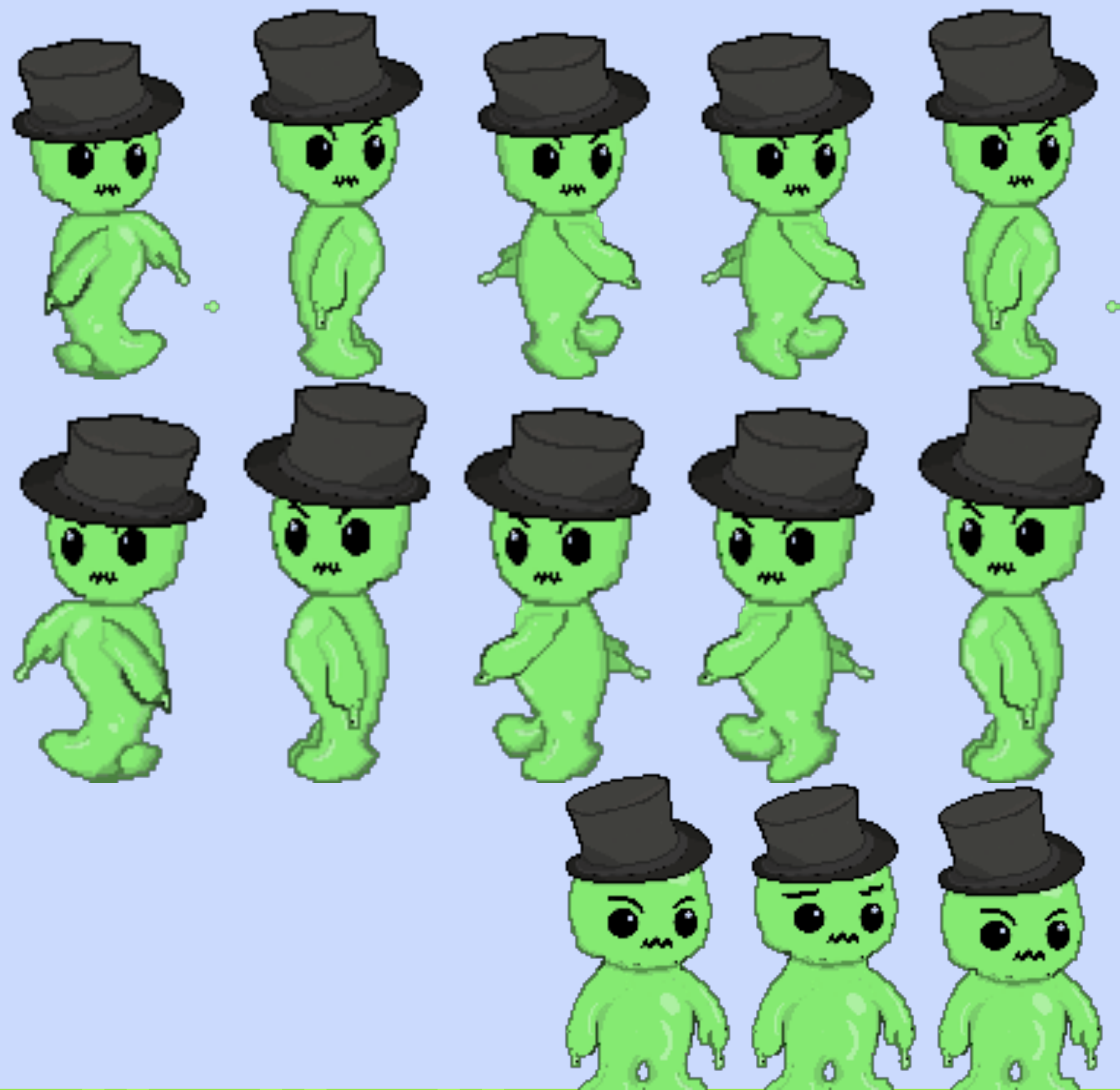


Moot v1



- **dünne Linien:**
Schleim-Körper glänzt mehr, mehr
Bewegungsfreiheit
- **dicke Linien:**
Hut, bleibt in Comic Stil
(differenziert von Körper)
- **dickere Hut-Linien bleiben für Comic-Stil**

Moot v1



Moot v2



- **Kindlichkeit**
 - A. Kopf und Körper fast gleich hoch
 - B. Grundform: Kreis
- **Grüner Schleim und Schleim Pocken assoziieren traurige Hintergrundgeschichte**
- **Hell-Dunkel Kontrast: melancholisch, mystisch**
- **Linienstärke**
 - A. Hut: dick für Comic Art
 - B. Körper: dünn für Schleimfluss und Erkennbarkeit
- **Kanten und Rundung smooth und ruhig**

Moot - Silhouette

- Silhouette hat direkt gepasst
- Schleimstrukturen und Tropfen eindeutig erkennbar
- Schleim fließt "frei" herum
- Kanten wirken dynamisch und interessant



So hätten **Moot** und **Major Dom** auch
aussehen können.

"zu Alien-Standard"

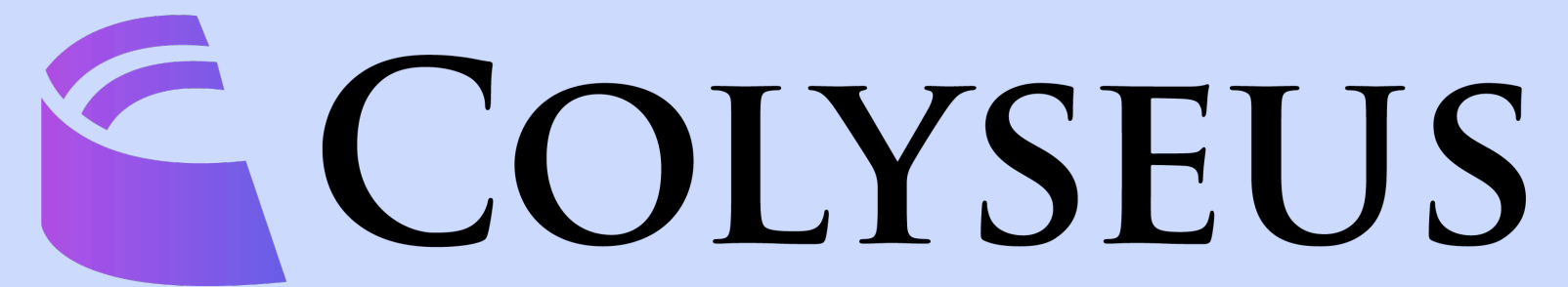


Projekt Management

Architektur

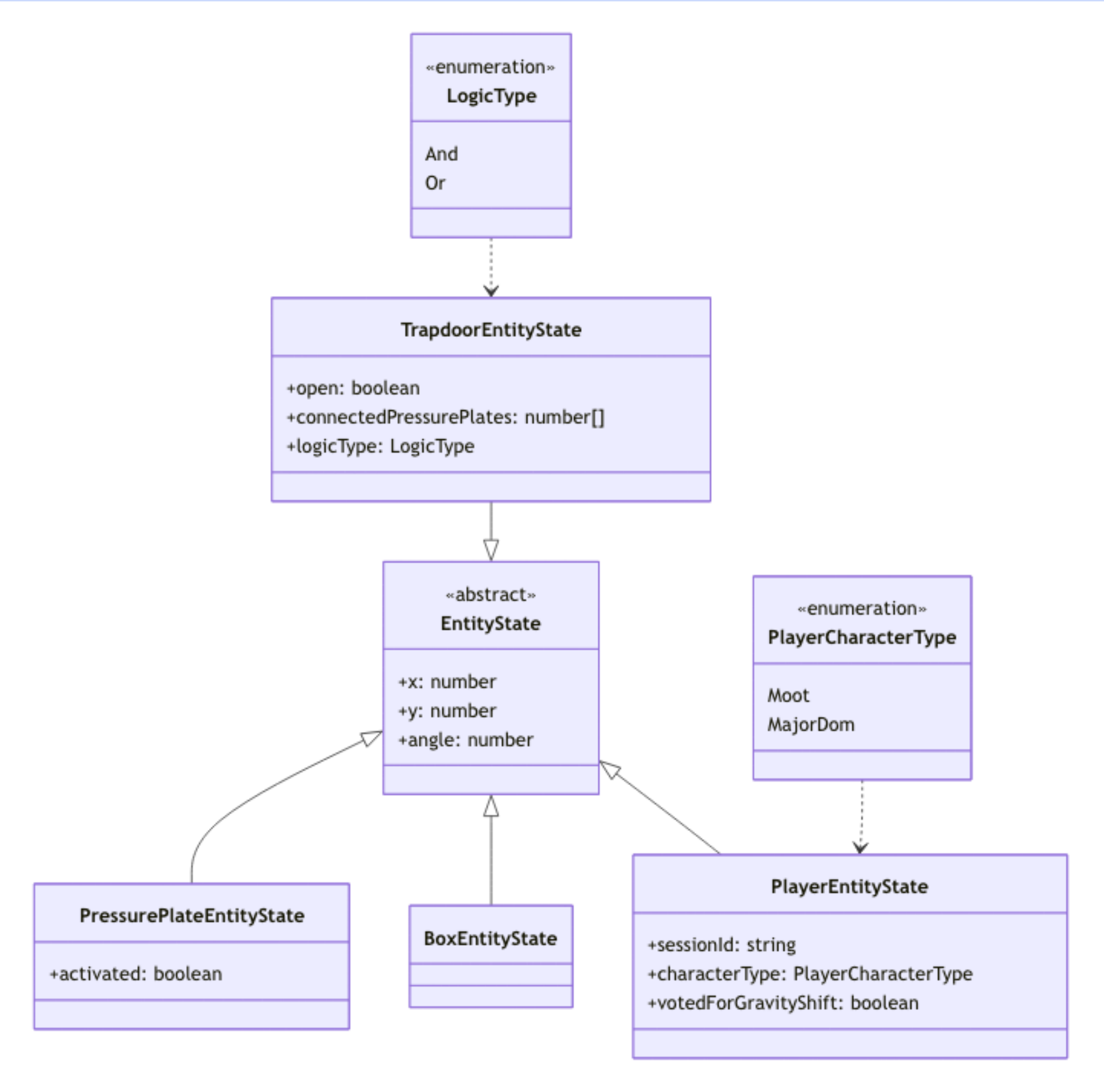
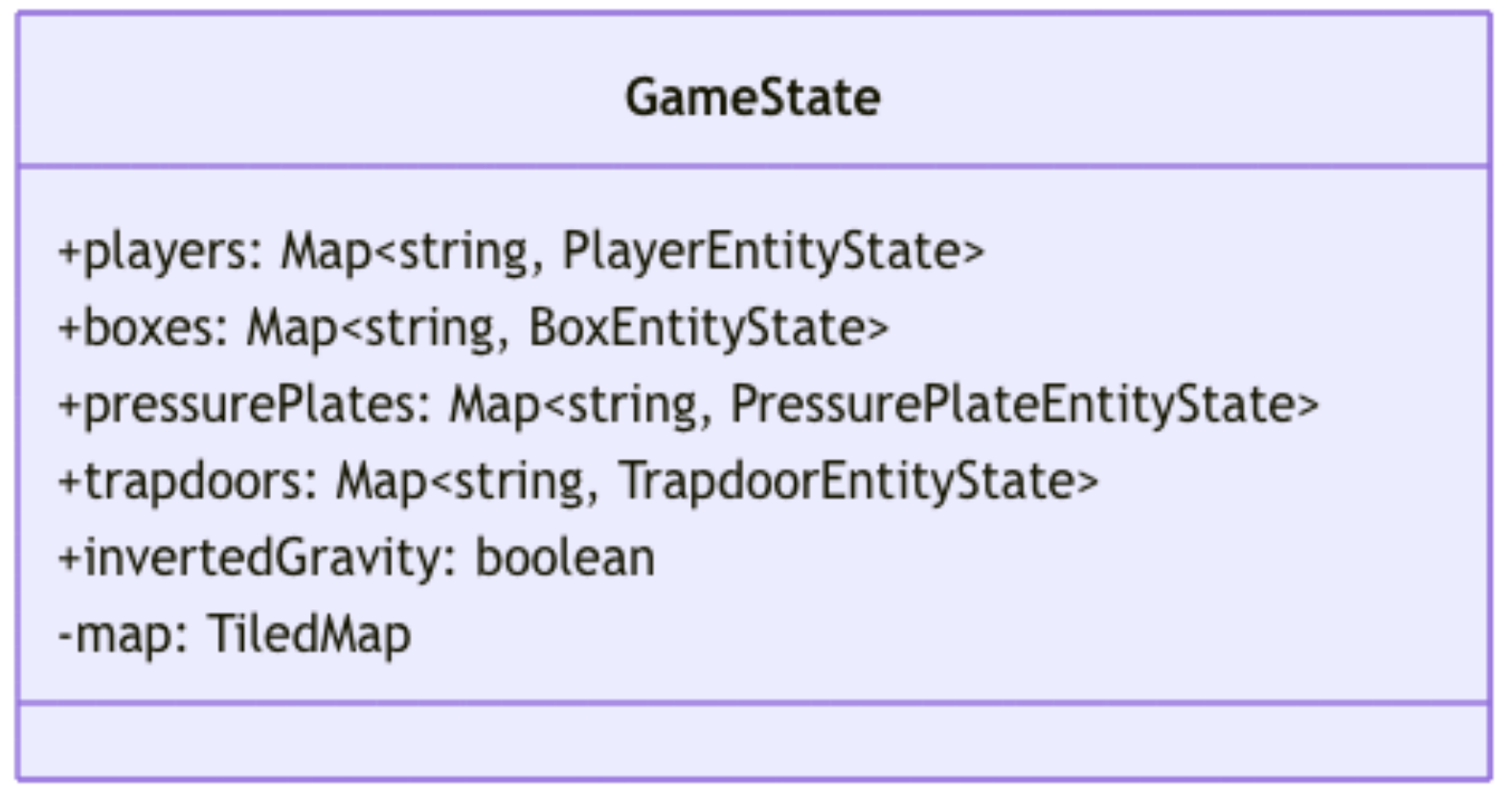


Server



- **Multiplayer Framework**
- **Kommunikation mit Client via Websockets**
- **Nachrichten delta-encoded**

Server: GameState



Client-Server-Communication

- **Clients render Physics**
- **Static Entities: können von jedem Client bearbeitet werden**
- **Dynamic Entities:**
 - **haben einen Physics Body**
 - **werden von einem Client berechnet**
 - **können nur mit dynamischen Entitäten von diesem Client und statischen Entitäten kollidieren**



Client: Manager

KeyboardManager
+leftIsDown: boolean +rightIsDown: boolean +jumpUpIsDown: boolean +jumpDownIsDown: boolean +gravityShiftIsDown: boolean
+InputManager(keyboardPlugin: Phaser.Input.Keyboard.KeyboardPlugin)

CameraManager
+CameraManager(entityManager: EntityManager, scene: Phaser.Scene) +update() : void

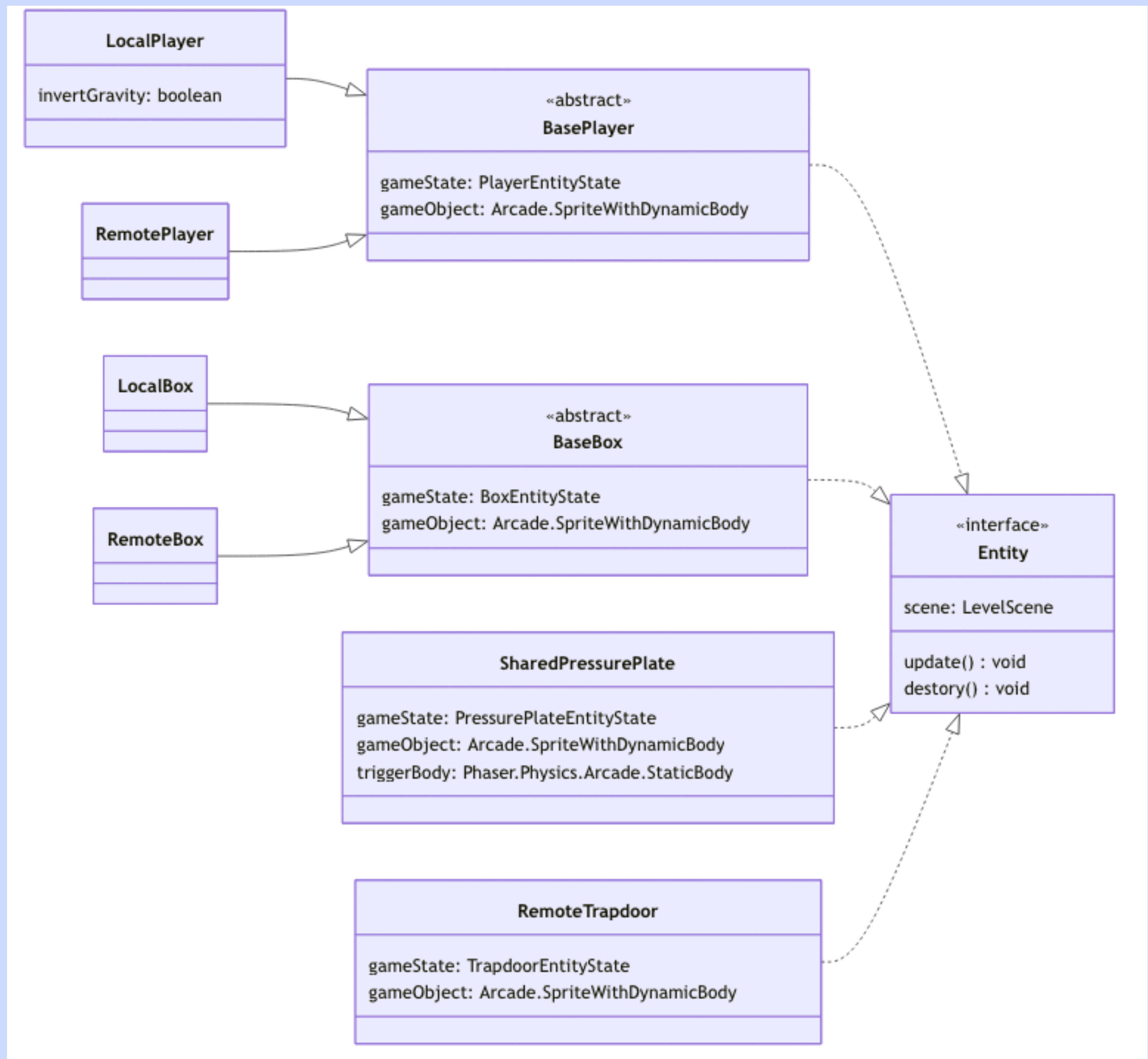
EntityManager
+EntityManager(scene: Phaser.Scene, room: Colyeus.Room<GameState>, keyboardManager: KeyboardManager) +getPlayers() : Map<String, BasePlayer> +update() : void



Client: Entitäten

- **Local Entities:** dynamische Entität die dem eigenen Client gehört
- **Remote Entities:** dynamische Entität die einem anderen Client gehört oder statische Entität, die vom Server berechnet wird
- **Shared Entities:** statische Entität
- **Base Entities:** standard Funktionalität die bei local und remote Entities gleich ist

Client: Entitäten



Client: Entitäten

