

# DESIGNKONZEPT HAPPY TIGER

Timon Dehmer, Online Medien, 256322.

Happy Tiger ist ein 2D Jump&Run Spiel für eine jüngere Zielgruppe. Der Happy Tiger muss eine gewisse Anzahl an generierten Münzen sammeln um das Spiel zu gewinnen. Währenddessen muss er sich vor Gegnern in Acht nehmen.

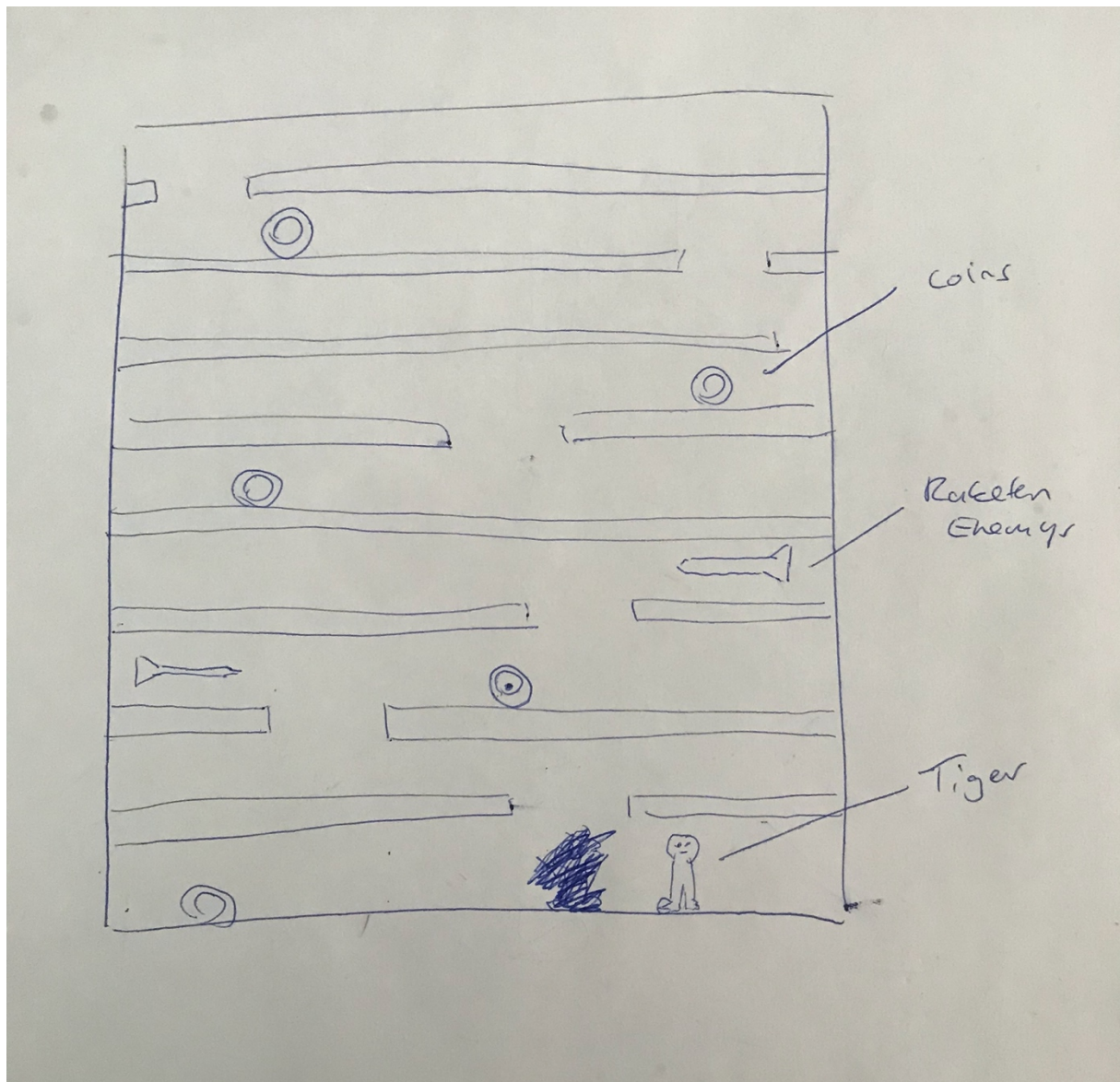
Berührt er eine Rakete verliert er das Spiel und muss von vorne beginnen. Die Münzen befinden sich auf verschiedenen Floors, die auf unterschiedlichen Höhen generiert werden.

## DESIGN HAPPY TIGER

### Spritesheet



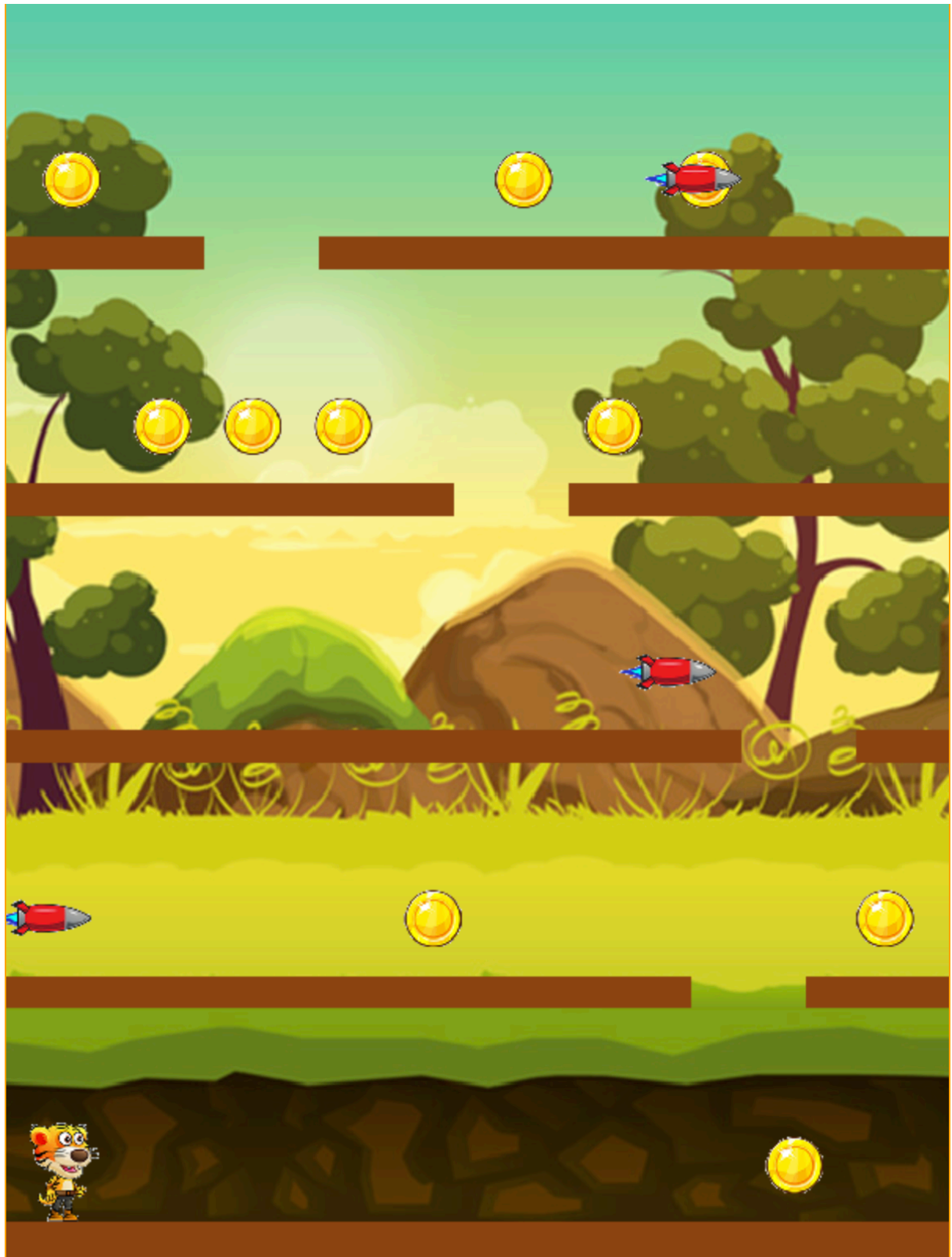
# Erste Skizze



# Hintergrund



# Spiel





# Ausarbeitung

1	Nutzerinteraktion	Mit „D“ läuft der Tiger nach rechts, mit „A“ nach links. Mit „Space“ springt er und auf „S“ duckt er sich. Mit „F“ rennt der Tiger in die Richtung, die gerade anvisiert ist.
2	Objektinteraktion	Der Tiger sammelt die Münzen auf und stirbt, wenn er von einer Rakete getroffen wird.
3	Objektanzahl variabel	Beim Start des Spieles wird eine variable Anzahl an Floors generiert. Die Anzahl der Raketen passt sich zudem der Anzahl der Floors an.
4	Szenenhierarchie	Level ist Child von Game. Der Tiger, die Raketen, Münzen, Floors und der Hintergrund sind Kinder von Level. Somit ist gewährleistet, dass zum Beispiel verschiedene Level erstellt werden könnten.
5	Sound	Es ist eine Hintergrundmusik eingebunden. Zudem werden Soundeffekte beim Einsammeln von Münzen abgespielt und, wenn man von einer Rakete getroffen wird.
6	GUI	Der User kann durch das GUI am Beginn folgende Optionen wählen. <ol style="list-style-type: none"> <li>1. Er startet direkt das Spiel über „Start“.</li> <li>2. Er wählt „Optionen“ und kann die Lautstärke der Musik und Soundeffekte regulieren.</li> <li>3. Er geht auf Anleitung um zu erfahren, wie er den Tiger steuert.</li> </ol> Sobald er verliert oder gewinnt erhält er die Option über „Restart“ das Level neu zu laden und noch einmal zu spielen.
7	Externe Daten	Über die „data.json“ kann extern ein Spielparameter verändert werden. So kann die Anzahl der Coins in der Datei definiert werden. Die Anzahl an Coins die angegeben ist wird im Spiel generiert.
8	Verhaltensklassen	Jedes Objekt im Spiel hat eine eigene Klasse. So ist das Verhalten des Tigers in „Tiger.ts“ definiert. Andere Verhaltensklassen: <ul style="list-style-type: none"> <li>• Rakete -&gt; Rocket.ts</li> <li>• Münzen -&gt; Coin.ts</li> <li>• ...</li> </ul>
9	Subklassen	In der Item.ts sind einige Eigenschaften beschrieben, die aufgrund von Überschneidungen von der Rakete in Rocket.ts und von den Münzen in Coin.ts geerbt werden.
10	Maße & Positionen	Die Maße, Skala und Positionen sind alle im Größen-Verhältnis dem Tiger angepasst. Das Spiel ist im

		Hochformat, da der Tiger von unten nach oben springen kann.
11	Event-System	<p>Das Event-System wird verwendet. Zum Beispiel für:</p> <ol style="list-style-type: none"> <li>1. Keydown Event bei der Steuerung des Charakters</li> <li>2. On-Click Events beim Klicken der Buttons im Menü</li> <li>3. Der Sound kann mit einem Regler gesteuert werden</li> </ol>