Dokumentation – Blockstack

# Anforderungen

Bei BLOCKSTACK geht es darum einen Turm aus Blöcken aufzubauen, je mehr Blöcke desto besser. Dabei kommt nach jedem erfolgreichem platzieren eines Blocks ein neuer Block am Seil eines Krans. Je ungenauer man die Blöcke stapelt, desto mehr wackelt der Turm, das Seil an dem die Blöcke ins Spiel kommen schwingt immer. Wenn der Turm aus Blöcken sich zu weit auf eine Seite neigt, so kippt dieser und das Spiel ist zu ende. Der Highscore beim Spiel wird durch die Anzahl gestapelter Blöcke definiert. Zu Beginn des Spiels findet man sich im Hauptmenü, es existieren die Optionen "Hilfe", "Spiel starten" und "Highscores". Wenn man das Spiel spielt besteht die Möglichkeit, das Spiel zu pausieren. Es erscheint ein Pausenmenü mit den drei Optionen "Fortfahren", "Neustart" oder "Hauptmenü". Wenn der Turm aus Blöcken einstürzt, sieht man seine erreichte Punktezahl und kann von dort aus zu der Highscore-Liste navigieren; wenn die erreichte Punktezahl unter den Top 10 liegt so kann man einen Namen angeben und diesen Speichern. Danach kehrt man ins Hauptmenü zurück oder das Spiel direkt neu starten.

# Konvention und Guidelines

## Ordnerstruktur

Documentation - Beinhaltet die Dokumentation  
Libraries - Beinhaltet die externen Libraries und Frameworks  
View - Beinhaltet die Views  
Controller - Beinhaltet die Controllers  
Game - Beinhaltet die Game Logik, da es mehrere Files sind wurde ein Ordner erstellt.

## Guidelines

Wir halten uns an die folgenden Guidelines: http://javascript.crockford.com/code.html

## Libraries und Frameworks

Wir haben uns entschieden mit Matter.js zu arbeiten; eine Library die uns Phsiycs zur verfügung stellt.  
http://brm.io/matter-js/  
Ausserdem arbeiten wir mit dem responisve Framework Foundation  
<http://foundation.zurb.com/>

Ausserdem arbeiten wir mit dem Framework „jquery“ und „jquery-mobile“   
<https://jquery.com/>  
<https://jquerymobile.com/>

# UML

# Database

# Test