

Backend-Dokumentation «Point Destroyer»

Technologie-Stack

Frontend	<p>Natives JavaScript, keine weiteren Libraries</p> <p>Das Frontend wird mit nativem JavaScript umgesetzt, um die grösstmögliche Flexibilität zu wahren und gleichzeitig den Code klein zu halten um so wenig Dateien wie möglich dem Client ausliefern zu müssen.</p>
Backend	<p>Java-Applikation</p> <p>Für das Spiel wird HTTP verwendet. Java bietet hier eine einfache und gute Möglichkeit HTTP umzusetzen.</p>
Datenspeicher	<p>MongoDB</p> <p>Als Performance-Starke Datenbank wird die objekt-orientierte Datenbank MongoDB verwendet. Die einzelnen «Player Records», d.h. die Information zu einem Spiel und möglicherweise, ob es geownt wurde, können dabei als unabhängige Objekte angesehen werden.</p>

Frontend-API

POST /api/player-record {id: string, playername: string, initialMessage: string, onwedMessage: string, initialPlaytime: string, fasterTime: string, emoji: string}

GET /api/player-record {id: string, playername: string, initialMessage: string, onwedMessage: string, initialPlaytime: string, fasterTime: string, emoji: string}

PUT /api/player-record {id: string, playername: string, initialMessage: string, onwedMessage: string, initialPlaytime: string, fasterTime: string, emoji: string}

POST /api/emoji-list {id: string, emojis: [string]}

GET /api/emoji-list {id: string, emojis: [string]}

PUT /api/emoji-list {id: string, emojis: [string]}

Datenbank

Die Datenbank umfasst folgende Tabellen:

Tabelle	Inhalt
Player-Record	id playername initialMessage onwedMessage initialPlaytime fasterTime emoji
Emoji-List	id emojis

Backend

Player-Records

Über die API */api/player-records* kann ein fertiges Spiel gespeichert werden. Ebenso ist es möglich, die bisherigen Spielergebnisse abzurufen. Um Fehler zu beheben, ist es ebenso möglich, einen Eintrag anzupassen.

In der Datenbank wird ein Eintrag mit folgenden Werten erstellt:

<i>id</i>	Wird im Frontend generiert.
<i>playername</i>	übergebener <i>playername</i> .
<i>initialMessage</i>	übergebene <i>initalMessage</i> .
<i>ownedMessage</i>	«null»
<i>initalPlaytime</i>	übergebene <i>initialPlaytime</i> .
<i>fasterPlaytime</i>	«null»
<i>emoji</i>	übergebenes <i>emoji</i> .

Die automatisch generierte *id* ist zusammengesetzt aus einer randomisierten Zahlenfolge und dem gewählten *playername*.

Emoji-List

Über die API */api/emoji-list* können die verschiedene Emojis ausgelesen werden. Diese werden für den Player-Record benutzt. Ebenso ermöglicht die API, diese Liste anzupassen oder neue Listen zu erstellen.

In der Datenbank wird ein Eintrag mit folgenden Werten erstellt:

id Wird im Frontend generiert.

emojis übergebener *emojis*.