Grand Archive API

Optionen zum suchen //Da sehe ich auch die Namen die Slugs der Sets

<https://api.gatcg.com/option/search>

Alle Karten einer Edition am Beispiel Alchemical Revolution

<https://api.gatcg.com/cards/search?prefix=ALC>

Für Seite 2

<https://api.gatcg.com/cards/search?page=2&prefix=ALC>

# Notizen für die Entwicklung

## React – Frontend

https://react.dev/

## Express - Backend

https://expressjs.com/

## Socket.IO

<https://socket.io/docs/v4/tutorial/introduction>

Wichtig für die Chat-Implementierung und der Aufrechterhaltung des Gamestates in Echtzeit

# Datenbanken

## cardDatabase

Eine Datenbank, die die API komplett abgreift und die gesamten Karten aller Editionen speichert.

## StateofTheGame

Zweite Datenbank, die den Status der aktiven Spiele dokumentiert, siehe stateoftheGame.DRAWIO-Datei

# Pull Rates

## Mercurial Heart

191 Cards

62C 45U 42R 32SR 10 UR 14CSR 1CUR

## Mortal Ambition

229 Cards

75c 59U 55R 32SR 8UR 11CSR 1CUR

## Alchemical Revolution

214 cards

81C 54U 42R 29SR 8UR 16CSR

## Dawn of Ashes

280 cards

114C 75U 54R 32SR 5UR 4CSR

## Fractured Crown

90 cards

28C 18U 21R 16SR 7UR

**Zusammenfassung (Prozentchancen)**

| **Edition** | **C (%)** | **U (%)** | **R (%)** | **SR (%)** | **UR (%)** |
| --- | --- | --- | --- | --- | --- |
| **Mercurial Heart** | 32.46 | 23.56 | 21.99 | 16.75 | 5.24 |
| **Mortal Ambition** | 32.75 | 25.76 | 24.02 | 13.97 | 3.49 |
| **Alchemical Revolution** | 37.85 | 25.23 | 19.63 | 13.55 | 3.74 |
| **Dawn of Ashes** | 40.71 | 26.79 | 19.29 | 11.43 | 1.79 |
| **Fractured Crown** | 31.11 | 20.00 | 23.33 | 17.78 | 7.78 |

Diese Tabelle zeigt die Verteilung der Seltenheiten pro Edition in Prozent.

# Booster Weitergabe

Überlegungen Booster an den nächsten Spieler weiter reichen.  
  
Wenn ein Spieler eine Card picked, dann steht der User auf „picked“.

User mit „picked“ können ein neues Booster erhalten und wenn dies geschieht, dann stehen sie auf not\_picked

Booster können sich bei einem Spieler stauen / Zeitlimit in der Zukunft.

Wenn sich Booster stauen, wird die **Reihenfolge** dieser relevant.

In der „draft\_pick“ Funktion könnte ich eine Abfrage alle User der Lobby machen und schauen welche Booster benötigen und diese dem Spieler zusenden.

Dann wird die Datenbank immer nur abgefragt, wenn ein User einen Pick macht, statt durch regelmäßige Zeitintervalle.

Booster haben nun eine „position“ INT = 1.  
Wenn ein Spieler bereits ein Booster auf 1 hat und ein weiteres Booster bekommt, dann bekommt das folgende Booster x+1.  
Wenn der Spieler wieder eine card picked, dann müssen all diese Werte -1 gerechnet werden.