

Backend Engineer Test

The popular collectible card game hearthstone releases constantly new expansions, showcasing thousands of new cards. Since it is overwhelming, we are in the need to create an intelligent deck builder to reach legendary rank.

To achieve this, it is necessary to create a JSON REST API endpoint in Django, that retrieves 30 cards from the card set of the latest expansion “Rastakhan’s Rumble” given a player class (eg. Mage, Hunter, etc).

Card information can be retrieved from the following source.

```
curl --get --include 'https://omgvamp-hearthstone-v1.p.mashape.com/cards/sets/Rastakhan%27s%20Rumble' \
-H 'X-Mashape-Key: ZTMJtzbYvXmshPTFEZI4ztIy3I68p1nPwgHjsnIGukKZeJxGcs '
```

Each item in the array represents a card. Eg:

```
{
  "cardId": "TRL_500e",
  "dbfId": "50371",
  "name": "Absolutely Mad!",
  "cardSet": "Rastakhan's Rumble",
  "type": "Enchantment",
  "text": "+2/+2.",
  "playerClass": "Priest",
  "locale": "enUS"
}
```

There are 10 possible values for the “playerClass” attribute:

Druid, Hunter, Mage, Paladin, Priest, Rogue, Shaman, Warlock, Warrior, Neutral

Objectives

Build an endpoint that given a `player_class`, return 30 cards picked randomly within the following conditions:

- The deck must contain only cards matching the `player_class` or "Neutral".
- There can only be a maximum of two of the same cards.

Sample response JSON response:

```
[
  {
    "dbf_id": "50371",
    "name": "Absolutely Mad!",
    "player_class": "Priest"
  },
  {
    "dbf_id": "50378",
    "name": "Avenger",
    "player_class": "Neutral"
  },
  .....
]
```

The app must be deployed in heroku.com and the database should already be populated with the card information.

Evaluation

We will consider the following criteria for evaluation:

- Git usage, frequent commits and useful descriptions.
- Code style (PEP08).
- Class and methods documentation .