

WBD5100.S1 PHP Custom CMS

Julien Ramirez

WDBP 0920

SAE Genève

11 décembre 2021



"Votre collection de cartes à portée de clique"



Présentation du projet

“Am i Lucky ?” vous permet d’enregistrer votre collection de cartes préférée de manière numérique. Accessible partout depuis votre ordinateur ou votre smartphone, l’application deviendra l’outil ultime pour vous permettre de compléter vos collections de cartes, de les partager avec vos amis, ou le monde entier.

L’application propose déjà les jeux de cartes Pokémon et Magic the gathering. D’autres jeux tels que “Flesh and Blood” ou encore “Yu-Gi-Oh” viendront prochainement s’ajouter au catalogue de cartes disponibles.

Fonctionnement général de l’application

Pour pouvoir profiter au maximum de l’application et commencer à créer vos collections, il vous suffit de vous enregistrer sur le site.

Une fois cela fait, vous pourrez alors créer votre première collection grâce au bouton “Mes collections”. Choisissez le nom de votre collection, le jeu auquel elle appartient, puis commencez à rechercher les cartes que vous souhaitez ajouter. La recherche fonctionne de deux façon :

- La recherche par set de carte qui se base sur les noms ou codes de set afin de vous afficher toutes les cartes appartenant à ce set.
- La recherche par nom de carte qui vous permet d’entrer le nom de la carte que vous recherchez. Toutes les cartes comprenant le mot que vous avez entré s’afficheront alors.

Une fois que vous aurez coché toutes les cartes que vous voulez ajouter à votre collection, cliquez sur “ajouter les cartes à ma collection”, et le tour est joué ! Vos cartes sont désormais enregistrées dans votre collection.



Enfin, dans le panneau de contrôle de votre collection, vous pourrez voir vos cartes de manière plus précise, les supprimer de votre collection, ou même supprimer la collection entière si vous le désirez.

Documentation technique

1. Langages

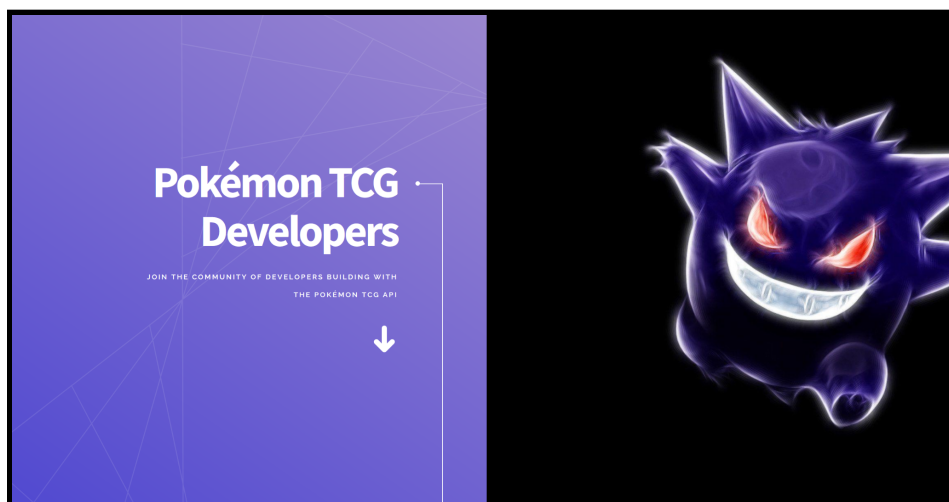
L'application utilise les langages HTML, CSS, SQL et PHP et est développée uniquement pour les formats desktop.

2. API :

L'acquisition des différentes cartes de l'application se fait par l'utilisation de deux API.

Pour les cartes Pokémon :

Pokemon TCG.io

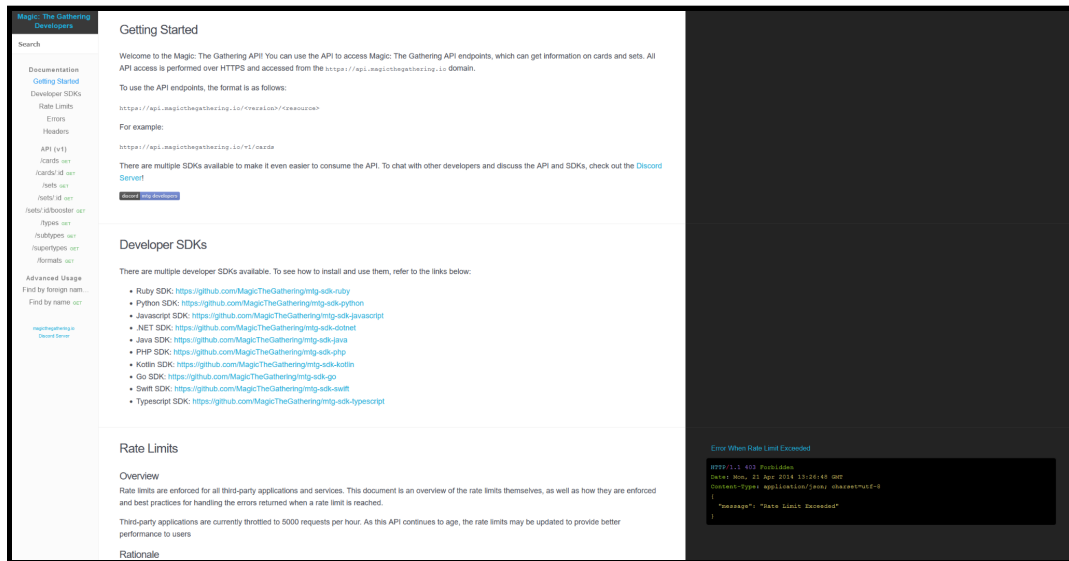


<https://pokemontcg.io/> (nécessite une clé API via inscription)



Pour les cartes Magic :

MagicTheGathering.io



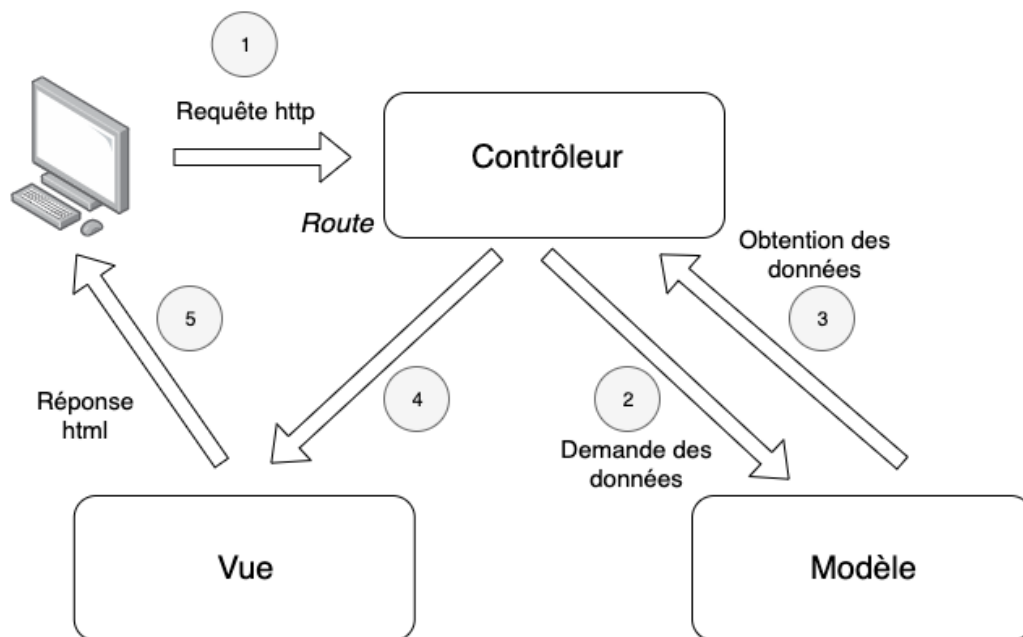
<https://docs.magicthegathering.io/> (nécessite aucune clé API)

Les deux API proposent des versions SDK compatibles avec PHP, ce qui permet d'exécuter les différentes requêtes directement dans des modèles php dédiés. Les requêtes permettent notamment d'obtenir le nom des sets de cartes, les cartes liées aux différents sets ainsi que toutes les informations des cartes elles-même, images comprises.



3. Architecture MVC

L'application est construite sur le modèle MVC (models views controllers), ci-dessous, un schéma explicatif de cette architecture :



Ce modèle a été choisi dans le but d'améliorer la maintenabilité du code et de l'application. En effet, ce dernier offre une meilleure visibilité du code et permet de structurer proprement les différentes couches de l'application.



4. Framework

Afin d'obtenir un visuel propre, l'application utilise le framework Bulma.



Bulma: the modern CSS framework that just works.

Bulma is a free, open source framework that provides ready-to-use frontend components that you can easily combine to build responsive web interfaces.

✓ No CSS knowledge required.

Star 44,725 downloads 823k/month

```
npm install bulma
```

Getting started

Download v0.9.3

<https://bulma.io/>

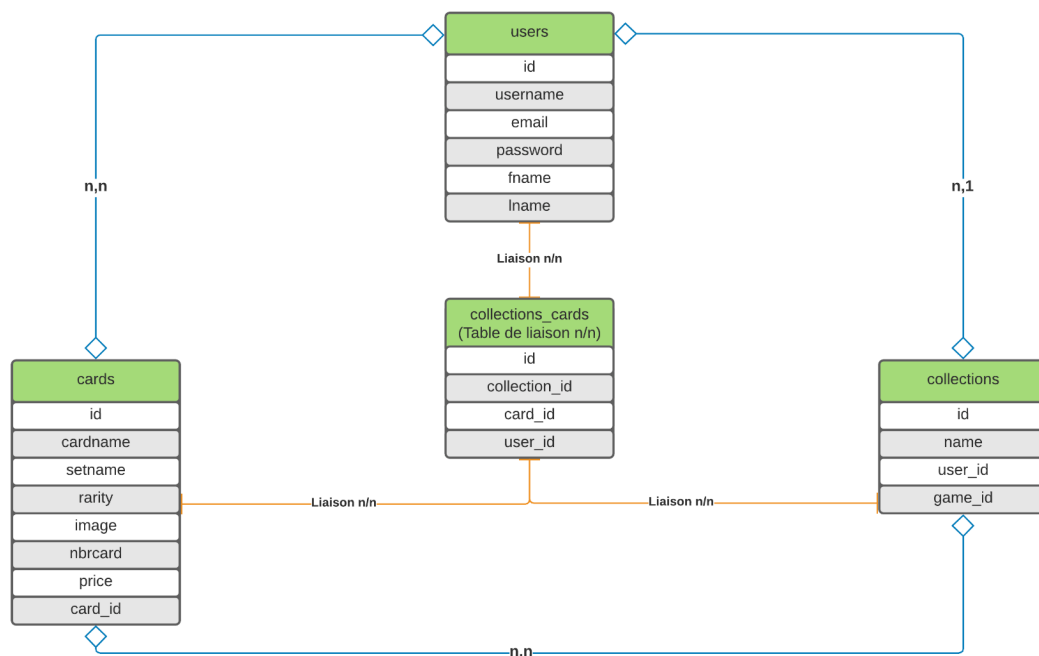
Bulma offre, par l'intermédiaire de ses classes CSS, un panel de style propre et simple à mettre en place afin de créer un site au rendu moderne et design. Il permet également un accès simple pour des messages d'erreur, pour des fonctions de login ou d'inscription, et une barre de navigation soignée. Toutes ces raisons ont facilité le choix de ce framework pour le projet.



5. Base de données

La base de données a été créée sur HeidiSQL. Elle contient 4 tables ainsi que plusieurs relations entre elles. Ci-dessous, vous trouverez un schéma de cette base de données :

Diagramme BDD Am I Lucky



Une table de liaison a dû être créée afin de pouvoir gérer les relations N/N, car, par exemple, une collection peut posséder plusieurs cartes et une carte peut appartenir à plusieurs collection.

Toutes les requêtes liées à la base de données se font en langage SQL depuis le modèle qui gère la requête en question.



Difficultés rencontrées et solutions

Durant le développement de l'application, plusieurs difficultés sont apparues. Afin de pouvoir proposer un produit fonctionnel, il a donc été nécessaire de surmonter ces difficultés et de trouver des solutions adaptées. Ci-dessous, vous trouverez une liste de quelques une de ces difficultés et des solutions proposées pour les surmonter :

1. **Problème :**

L'API Pokémon est passé de la V1 à la V2 en plein développement du site, ce qui a engendré des erreurs car les requêtes d'accès aux cartes étaient complètement différentes.

Solution :

La v2 étant quand même plus rapide, la solution à été de mettre à jour le code du site, d'apprendre la forme des nouvelles requêtes et de les implémenter dans le site.

2. **Problème :**

Sur la première version de l'application, le système de routage était différent de l'actuel. En effet, le premier système de routage ne permettait pas de passer des paramètres dans l'url.

Solution :

La fonction de routage a dû être complètement réécrite afin de pouvoir passer au moins un paramètre. Cette nouvelle fonction s'est avérée plus efficace et mieux structurée.



3. Problème :

La base de données imaginée lors de la création de l'application ne comprenait pas les bonnes relations. En effet, comme expliqué dans le chapitre "Documentation technique" au point numéro 5, la base de données nécessite des relations N/N pour fonctionner.

Solution :

Afin de palier au problème de relation, il a fallu créer une table supplémentaire afin de créer des liens, par exemple, entre les cartes et les collections.

4. Problème :

La structure MVC ne faisait pas partie du projet lors de sa création. Cette structure était inconnue du développeur et semblait être une source de perte de temps lors du développement.

Solution :

La structure MVC a été apprise puis implémentée dans le projet. Cela a finalement permis de structurer le code proprement, de travailler avec des objets et de faciliter les interactions entre les différentes couches.



Futur du projet et amélioration

La version présentée à ce jour est un pilote et n'est pas une version définitive du projet. L'idée générale est posée, et le design de base définit. Il y a cependant des points d'amélioration possibles ainsi que de nouvelles fonctions implémentables pour rendre le projet plus attrayant.

Voici quelques points qui seront à traiter à l'avenir :

1. Ajout de nouveaux jeux dans l'application tels que Flesh and blood ou encore Yu-Gi-Ho.
2. Décliner le site en application mobile.
3. Implémenter une reconnaissance de cartes par le biais d'une photo prise sur un smartphone.
4. Améliorer le design et l'ergonomie générale.
5. Permettre un partage de collection avec les autres utilisateurs par le biais des pseudonymes.
6. Enregistrer les cartes sur une base de données propre afin d'améliorer la vitesse de chargement du site.
7. Fonctionnalités liées à l'enregistrement de l'utilisateur et à la gestion du compte.



Remerciements

Ce projet a été un réel plaisir mais aussi un défi à réaliser pour moi. Il m'a permis d'acquérir de nouvelles compétences en développement et une vision plus large de la puissance du développement web.

Je remercie mes professeurs à la SAE de Genève pour leurs enseignements et leurs disponibilités.

Je remercie les collectionneurs de cartes que j'ai rencontré au fil du développement du site, pour leurs précieux conseils concernant les créations de collections et la recherche de cartes.

Merci pour le temps passé à lire ce document et à tester / juger mon travail.

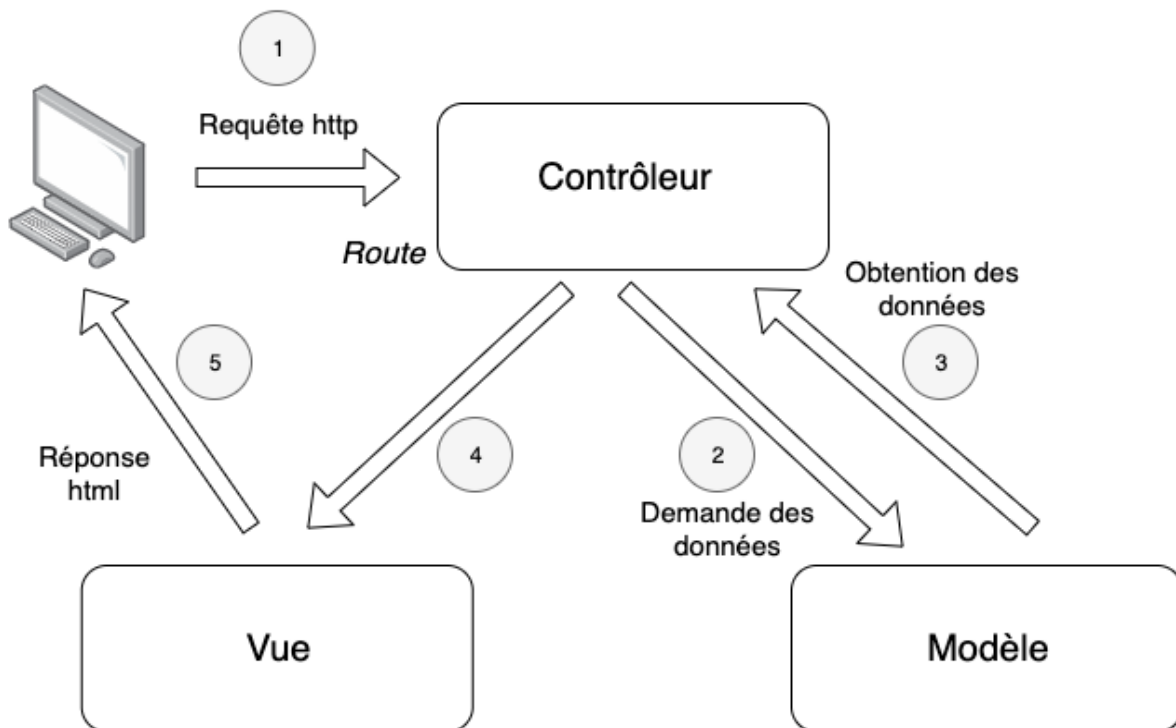
Et lorsque vous ouvrirez vos prochains booster de cartes, n'oubliez pas de vous demander ...





Annexes

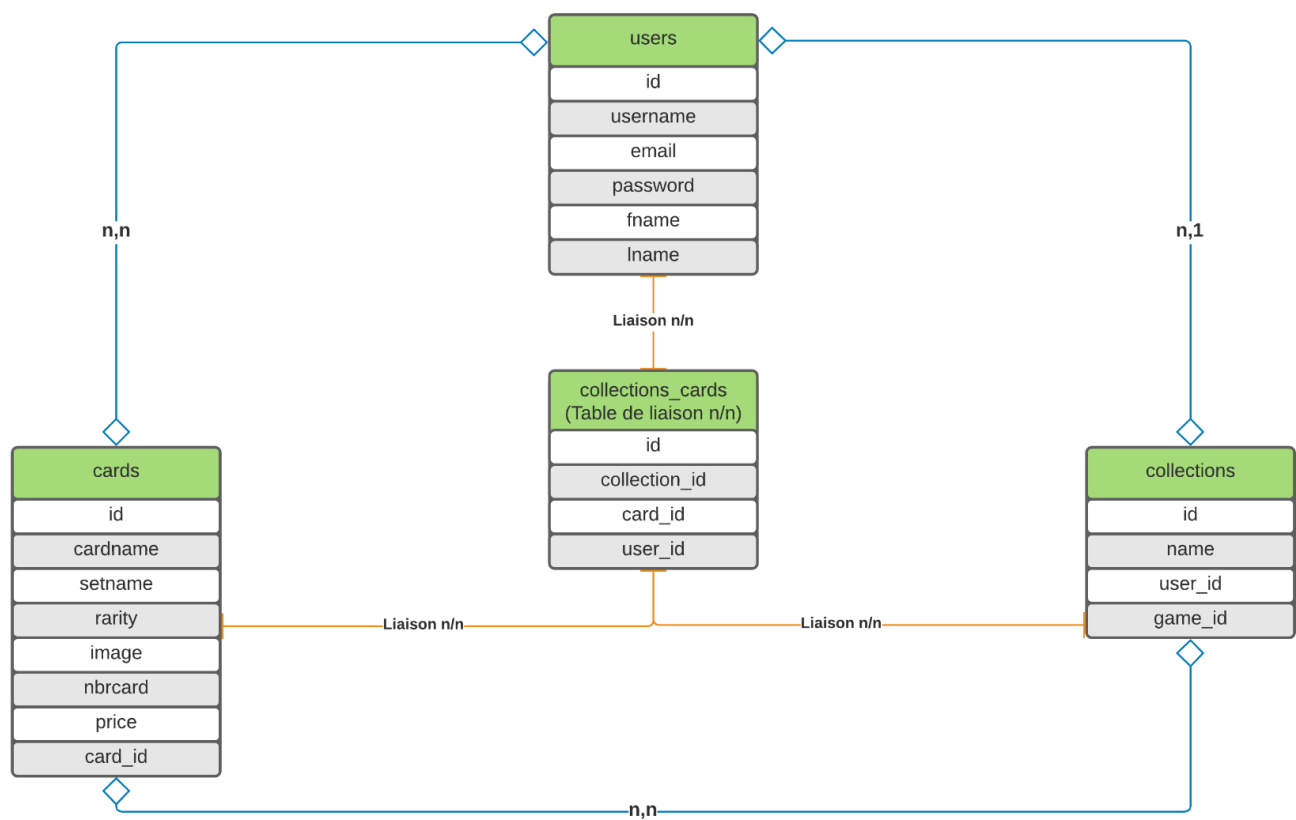
1. Schéma du modèle MVC





2. Schéma de la base de données

Diagramme BDD Am I Lucky



3. Site map

