Database Systems Project 1

Jjjj222

1 Overview

This application [1, 2] is a deck collector for Hearthstone [3], an online collectible card game developed by Blizzard [4]. In Hearthstone, a player can collect cards, build decks from these cards, and battle with other players by his or her decks.

The purpose of this application is to support players building their decks. Users can manage their decks by adding, deleting, or updating it in the database. It also supports other functionalities such as card searching and deck comparison.

The link of this application is as follows:
`http://projects.cse.tamu.edu/jjjj222/hsinfo/index.php`

2 Data Collection and Database Establishment

To build this application, 2 resources are required: the first one is the informaiton of every card, and the second one is a collection of decks. Although basically decks can be added by user manually, it is helpful to pre-include some basic ones.

2.1 Cards

I found a json file with the latest information of cards on Hearthstone JSON [5]. It's not in a relational format, so I made some modification while parsing it by php. All the information is stored into the table "Card" after parsing.

```
CREATE TABLE Card
(
    setType         CHAR(40),
    setSeq          INT,
    id              CHAR(20)        PRIMARY KEY,
    name            VARCHAR(255),
    type            CHAR(20),
    faction         CHAR(20),
    rarity          CHAR(20),
    cost            INT,
    attack          INT,
    durability      INT,
    health          INT,
    elite           BOOLEAN,
    race            CHAR(20),
    text            VARCHAR(255),
    inPlayText      CHAR(40),
    flavor          VARCHAR(255),
    artist          VARCHAR(255),
```

```
    collectible     BOOLEAN,
    playerClass     CHAR(20),
    howToGet        VARCHAR(255),
    howToGetGold    VARCHAR(255),
    mechanics       VARCHAR(255)
);
```

2.2 Decks

I used the decks on HearthHead [6] to build the default decks. The raw data looks like as follow:

```
Beast Druid [Kolento] by BooYaBob
http://www.hearthhead.com/deck=125114/beast-druid
Class: Druid

2 Innervate
2 Living Roots
2 Power of the Wild
2 Darnassus Aspirant
...
```

The application will automatically parse every such files under given folder and store them into Deck table, as well as building a seperate table for each deck.

```
CREATE TABLE Deck
(
    id              CHAR(20)        PRIMARY KEY,
    name            VARCHAR(255),
    class           CHAR(20),
    creator         CHAR(40),
    num             INT,
    link            VARCHAR(255),
    comment         VARCHAR(255)
);

CREATE TABLE xxx
(
    id              CHAR(20)        PRIMARY KEY,
    num             INT,
);
```

3 User Manual

In this application, as demonstrated in Figure 1, there are 6 tabs with different functionalities:

3.1 Home

The entrance of this application, for publishing news and information.

3.2 Card

Users can search for cards here.

For example, if an user wants to look for “Epic” cards in “Mage” class, then he o she can click on the “rarity” and choose “Epic”; then on “playerClass” and “Mage”, and application will automatically filter out the cards that meet the requirements.

In addition, users can type in partial name of a given card in “name” field to get cards. For example, if an user types in “fire”, then the cards with “fire” in their name will be displayed in the table below.

3.3 Deck

Users can review and delete decks here.

This tab supports the similar functionalities as in “Card”, but with an additional mouse over support: moving the mouse over the row of given deck will display the detail information of the deck on the right part of the screen.

If an user wants to delete a given deck, he can just click the “delete” button beside it.

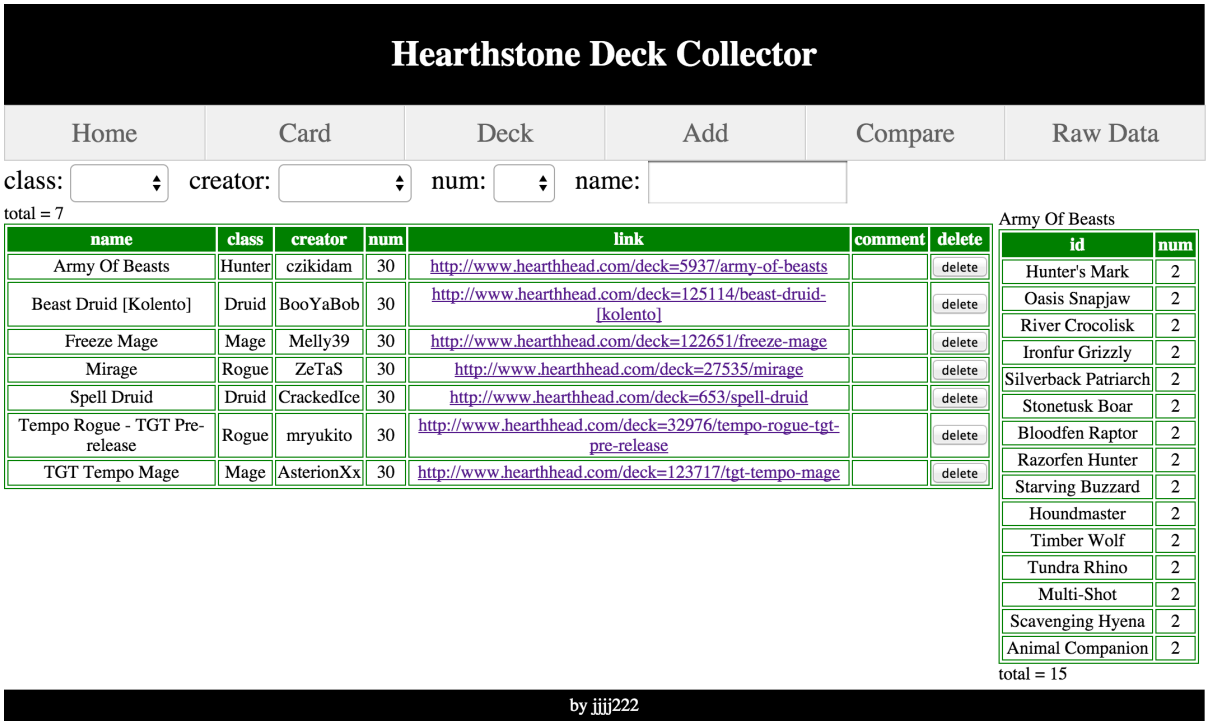

Figure 1: Screenshot of ”Deck” tab

3.4 Add

Users can add or modify decks here.

By the help of auto-recommendation, users can type in their card without difficulty. Simply pressing the "add" button after filling out all the required spaces, the user-specified deck will be added into the database.

If an user wants to update a deck, he can choose deck from "deck" and click "load" button, and the informaiton of the chosen deck will be loaded to this page. Users can press "update" while he or she is done the change, or press "add" to create a new deck based on current data.

3.5 Compare

Users can compare decks here.

Simply choose 2 decks to be comapred, and their informaiton will be listed below.

3.6 Raw Data

Basically it shouldn't appear in the application, but I put it here for convenience. This page provides the links for review and reset the raw data in database.

4 Thoughts and Conclusions

It took me a while to setup the development environment, because I planned to develop on my own computer and synchronize the work to the CSNET server only after reaching a certain point to release. To get everything done, I have to setup apache, php, mysql, and git on both environments, and it's a little tiresome.

There is no many problems in terms of using SQL to fetch data in database; however, it seems that every DBMS has it's own little differece. For example, MySQL doesn't support "LIKE", so I have to use "RLIKE" and the syntax is a little different. Besides, I can't use "FULL OUTER JOIN", so I have to use "LEFT JOIN" and "RIGHT JOIN" and combine the results myself.

One thing bothered me is that there is no natural join. I originally thought that everything will be joined easily with the same attribute names, but it turns out I have to work on "WHERE" and rename the attribute by "AS" myself. Plus, the "UNION" combine tuples by their positions instead of name, and it took me some time to try out the command by using the commend line provided by MySQL.

All in all, although setting up the connections between different components to build a web application is annoying, they turn out to be very efficient in terms of retrieving information. Even if I use many queries on a single operation, the MySQL can always finish the jobs in no time.

References

[1] Hearthstone Deck Collector
http://projects.cse.tamu.edu/jjjj222/hsinfo/index.php

[2] Source code of Hearthstone Deck Collector
https://github.com/jjjj222/hsinfo

[3] Hearthstone: Heroes of Warcraft
http://us.battle.net/hearthstone/en/

[4] Blizzard Entertainment
http://us.blizzard.com/en-us/

[5] Hearthstone JSON
http://hearthstonejson.com/

[6] HearthHead
http://www.hearthhead.com/