The website that we created is based around creating a search engine for Magic: The Gathering cards. Magic is the oldest and most successful trading card game on the market, and due to its age and size, finding the cards, their prices, and other information pertaining to cards can be quite difficult. This website attempts to ease the process of searching for cards by allowing any user to search for a word, phrase, or card name, and the website returns all cards that have that text within the name of the card. We did this by having an API connection to a website called Scryfall, which hosts all the relevant information that we would need for our cards in a bulk database. We pull this database, and create a local copy of it. From there, the user is able to use a search bar to pull up the name, mana cost, type, rarity, cost, and set of the card that they are searching for.

Our website is serviced by Flask. This allows quick and easy access to endpoints required to invoke data from our database to our web server. In addition to this, we are using a SQLite database, and a HTML template page to host the data.

As a team, Hayden primarily worked on setting up the api call to Scryfall to retrieve the data, working on populating the webpage with cleaned up price data, and mana symbols, whereas Morgan worked on the frontend of the website, as well as the search engine. Daiquon worked on the word report for this assignment. Mya unfortunately did not contribute to the website nor the word report for this assignment. Our collaboration was done over a discord server.

Our website deserved to be deemed the best out of the competition due to our seamless integration of a database and search engine into a website. This allows for a fluid and quick access to card data in a enticing, visual way, allowing users to quickly and efficiently find data that they are searching for about any specific card or cards.

We were unable to get the website to host its self on a webpage. As a result, you will find that this website is delivered in a zip folder. Hayden has ensured that this dynamically populates, so in order to properly view this webpage, all that you will need to do is the following:

1. Download, and unzip the Scryfall\_Project.zip file.
2. Download Flask. This can be done by opening up a command prompt, and typing ‘pip install Flask’ if you have python installed. In addition to this, you will need to download cors. This can be done by typing ‘pip install flask\_cors’ in your command prompt.
3. After both flask and flask\_cors are installed, navigate to the ../Scryfall\_Project/backend directory.
4. Once you are in the backend directory, run ‘Python -w flask run’. This will create a webpage local to your machine that is being fed data by the attached database.

And you’re done! The website is now complete, and available for use.