Capstone 1 Proposal:
Dungeons & Dragons Content Repository

Goal:

The goal of this project is to allow a comprehensive place for Dungeons and Dragons (D&D) players to view official content, create their own (homebrew), and to discuss items with each other by means of a comment section for every item.

Targeted Users:

The primary targeted users of this application would be players and dungeon masters of Dungeons and Dragons.

Data and Data Source:

There are two sources that I am looking at using: https://open5e.com/ and http://www.dnd5eapi.co/. Of the two, I am leaning towards the former, just because it appears to contain a larger set of documents. However, both of these APIs are similar and the specific documents (spells, classes, weapons, armor) have their own specific endpoints as well as basic querying enabled on each of those endpoints.

Approach Outline:

The database schema will be different for each category of document, and even then, there will be variation among different documents within the same category. For example, one spell may cause damage to a group of enemy monsters and another spell may be used to heal a single allied player, which means that one spell will have damage and range statistics while the other one will have healing numbers. This will mean that schemas will contain a number of qualities, but many will be nullable. As far as simply fetching documents through the API, this is largely only a concern for handling returned JSON for display, but since custom items will need to mimic these schema, I will perform initial pulls of all documents for each category to ensure that I have cataloged all potential and mandatory fields. Additionally, some of the schema are nested, which will require intermediate tables in some places.

As for the database schema that will be unique to the app (and separate from an approximation of any part of the API), there will be comments (id, document index, title, body text, poster id, datetime posted), and users (id, username, email address, hashed password). Additional fields or tables may be added to implement minor feature upgrades (favorited weapons, monsters, etc.), but comments and users will otherwise be fairly straightforward.

Possible concerns are generally related to any documents from the API being modified while comments on that document would be otherwise "fixed". It may be necessary to take some precaution in such case, such as saving directly to the database any document that has a comment on it, so that it can be retrieved by the commenter or anyone else who wants to see the item in the proper context.

The only user information that would need to be kept secure would be the password, which would be stored on the database as a hash. As of now I do not have any plans to employ another API for password retrieval, but this would be a very likely additional feature if there is sufficient time to allow for it.

The functionality of the app is essentially to get official documents (spells, classes, weapons, armor), to post unofficial documents of those same categories, and to add and view comments on those documents.

A user should be able to log in and be presented with options to search for one of the supported document categories, or to simply go to the list of documents for all of these. On a document list or a search, a list of matching documents should populate on that page. Each item in that list should show the name of the document, the origin (which source book, or whether it is custom content and by which user), a button to expand the document, and a link to its dedicated page. The button to expand the document would cause this list item to expand and show the full detail of the document. Clicking on the link to its dedicated page would take it to a similar view with no other documents, but would show whatever comments there are for that document.

As far as stretch goals, there are many more categories of documents that could be added, as well as automated email for password resets, which would be the highest priority for the project. The app could also be further developed to allow for upvoting and downvoting documents for logged-in users, as well as notifying users if there has been additional comment activity on a document that they have created or commented on. These notifications, as well as a quick link to a trending document, could all show up on a user's home page directly after logging in.