Magic-The-Gathering: Main folder.

↳css:

↳index.css: Web app UI implementation.

↳landing.css: Login screen UI implementation

↳site.min.css: Web app implementation.

↳toasty.min.css: Pop up “toasts” implementation.

↳util.css: Utility implementation.

↳documentation: Folder containing all documentation for the project.

↳documentation.pdf: This file. Documents GitHub repository structure.

↳implementationPlan.pdf: Documents our implementation plan for this project.

↳fonts: Folder containing all font data for out web app.

↳html: Folder containing html files for the web app.

↳dashboard: Folder containing HTML file for the main interface.

↳index.html: HTML file that builds and displays the web app in a browser.

↳landing.html: HTML file that builds and displays the login screen in a browser.

↳images: Folder containing all images used in web app.

↳icons: Folder containing all icon images used in web app.

↳js: Folder containing main classes and functions for web app functionality.

↳database: Folder containing main .js file for calling our database.

↳firebase.js: Initializes firebase with a configuration key, authenticates the user’s account, and send the user back to the login screen if they are not logged in properly. Contains many functions related to accessing the user’s saved decks in our database and retrieving Card data from our database.

↳libraries: Folder containing main functionality for pop up “toasts”.

↳vendor: Folder containing .css files for “toasts” implementation.

↳toasty.min.js: Contains functionality to pop up “toasts” to the user.

↳logic: Folder containing main class logic for our web app’s functionality.

↳Card.js: Object class to construct a *Card* and retrieve values from it. This is constructed from the card’s name, mana cost, converted mana cost, type line, Oracle text, power, toughness, color, color identity, legality, quantity, and cost.

↳Deck.js: Object class to construct a *Deck* of *Cards*, retrieve *Cards* from it, and calculate statistics of the *Deck*. A *Deck* is constructed with a name, a format, and the array of *Cards*. Also contains a function to calculate the odds of drawing a specific card from the deck a specific number of times in a specific number of draws, which is a feature of our web app.

↳Pseudocode for Advanced Card chances: Details how the pseudocode to calculate *Deck* statistics works.

↳index.js: Contains many functions for web app functionality. This main screen has three sections. The first section is “Input Cards”, which allows the user to input cards into a deck. The user inputs the name of the deck, the name of the card, and the number of that card to add to the deck. The second section is “View Decks”, which allows the user to create or select a deck and view the cards in that deck. The third section is “Statistics”, which allows the user to view statistics about the selected deck, such as the number of lands or the average converted mana cost. This section also allows the user to calculate the odds of drawing a specified card from the selected deck a specified number of times in a specified number of draws, in the “Odds of a Card” subsection.

↳landing.js: Contains many functions for log-in functionality. Validates the user through Google login and creates a user file in our database for them. Only Google login currently works. Facebook and normal logins were a reach goal.

↳site.min.js: Contains website functionality.

↳.gitignore: Plugin ignore functionality.

↳card_data.json: Main JSON file that the firebase Database organizes.

↳README.md: Project title.