Pokédex

The challenge.

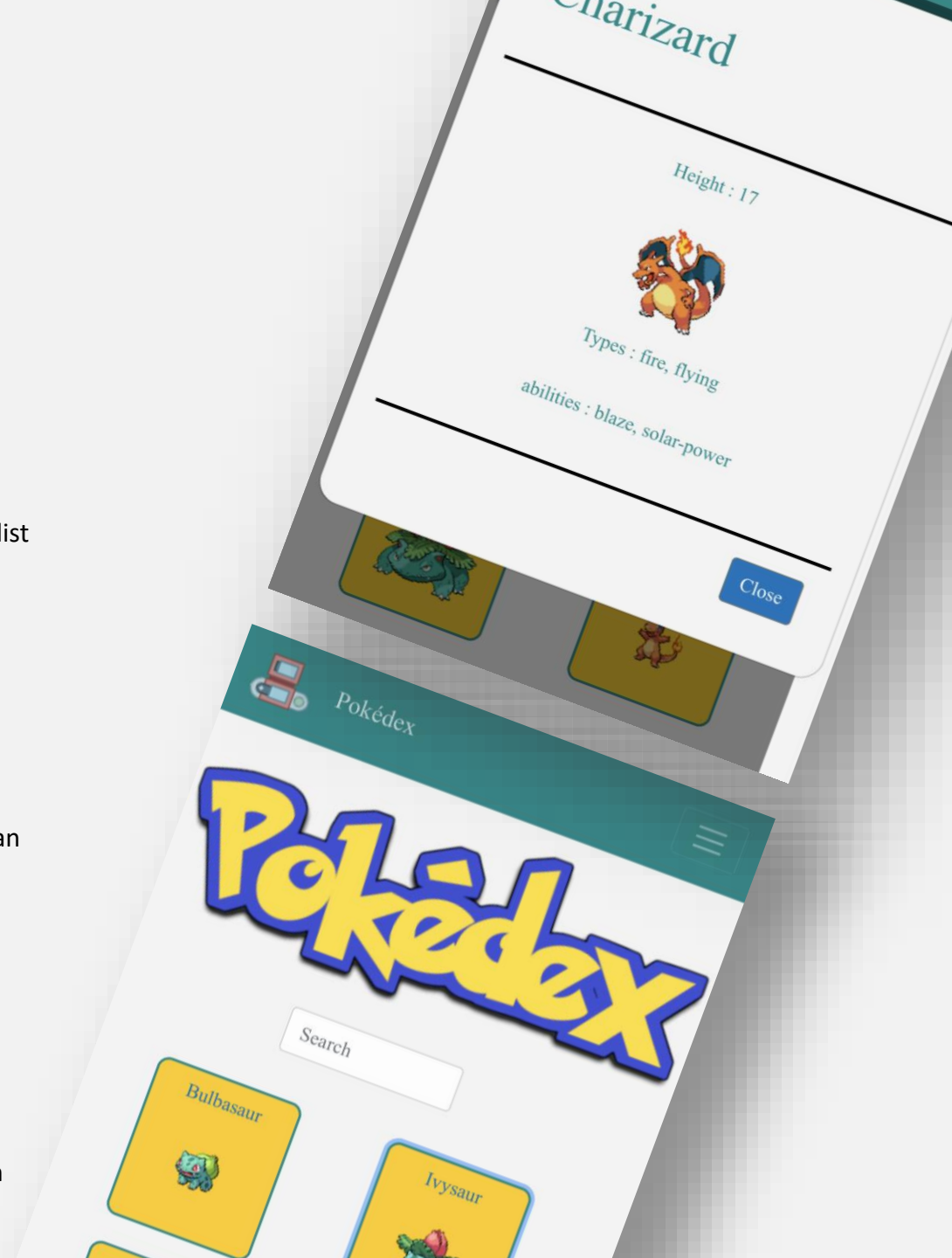
To create a web application that could fetch aggregate data from an external API and allow a user to view a list of data and see more details for a given data on demand.

The process.

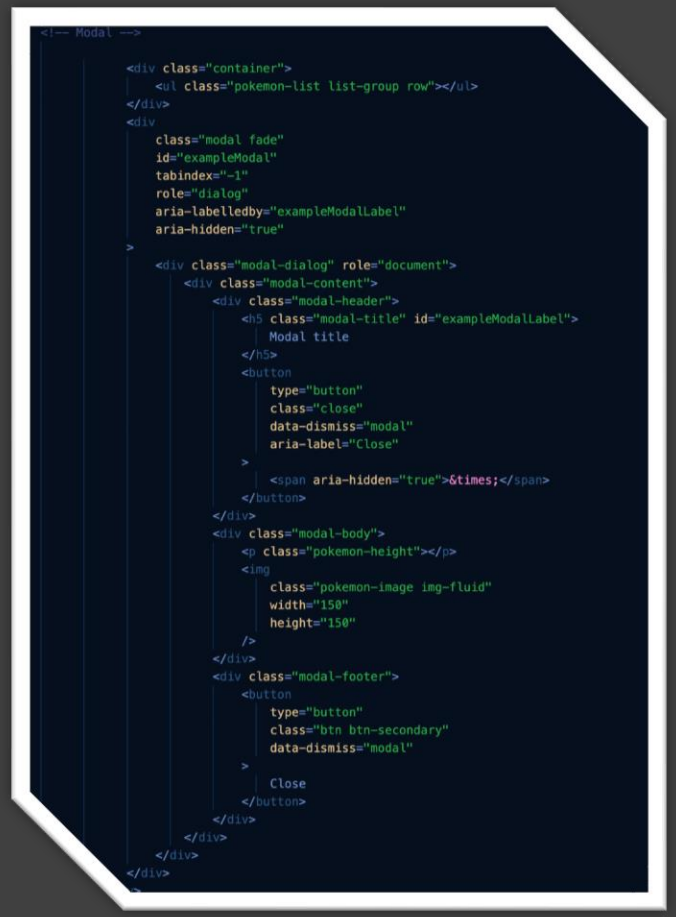
The process was to create a framework for the web app using HTML. Then using JavaScript, load data from an external API and allow the viewing of more details using modals. Then using CSS, apply styling to the web application to allow for a more user-friendly experience.

The goal.

Learn and display the basics of front-end development through the creation of a responsive web application that provides the user with information on Pokémon.



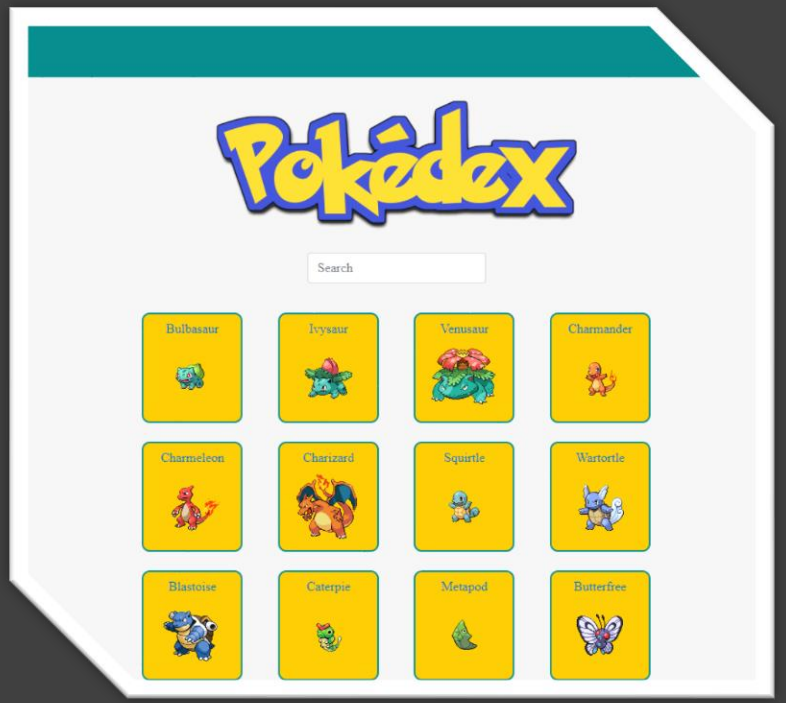
<HTML>



The HyperText Markup Language or HTML is the standard markup language for documents designed to be displayed in a web browser. It is in this portion of the project that I structured the layout of the web page using HTML elements. To the right is actual HTML code I used to structure the modal.

```
<div class="modal-dialog" role="document">
  <div class="modal-content">
    <div class="modal-header">
      <h5
        class="modal-title“
        id="exampleModalLabel">
          Modal title
      </h5>
      <button
        type="button"
        class="close"
        data-dismiss="modal"
        aria-label="Close"
      >
        <span
          aria-hidden="true">&times;
        </span>
      </button>
```

{CSS}



GitHub

CSS or Cascading Style Sheets is a protocol used to instruct the web browser how to display HTML content. This is where I started adding styling to the HTML. In the code to the right, you can see how I used CSS to present the modals I structured using HTML. I used design elements to determine the borders and padding as well as the color of the text and background.

```
.modal-content {
    border-radius: 20px;
    padding: 20px;
    color: var(--primary-color);
    background-color: var(--background-color);
}

.modal-footer {
    border-top: 3px solid #000;
}

.modal-header {
    border-bottom: 3px solid #000;
}
```

{JavaScript}

GitHub

JavaScript is a high-level, interpreted programming language that is primarily used for adding interactivity and dynamic behavior to websites. In this project I used JavaScript to enhance the user experience by enabling interactive features like this example of the modal function. The code to the right determines how the data from the external API will be presented to the user when interreacting with the modal.

```
function showDetails(pokemon) {
  loadDetails(pokemon).then(function () {
    function showModal(pokemon) {
      let modalTitle = $('.modal-title')
      let modalBody = $('.modal-body')
        modalTitle.empty()
        modalBody.empty()
          let nameElement = $(
'<h1>' +
            pokemon.name.charAt(0).toUpperCase() +
            pokemon.name.slice(1) +
'</h1>'
)
    let heightElement = $(
      '<p>' + 'Height : ' + pokemon.height + '</p>'
)
    let imageElement =
      document.createElement('img')
    imageElement.src = pokemon.imageUrl
}
```

Retrospective.

When I originally set out to design this web app, I was looking to create something that was intuitive and enjoyable to use, and I think I have accomplished that goal. The project was an enjoyable journey, but it was not without its complications. At times working to integrate data from an external API into my web app could be difficult but after multiple iterations I was able to accurately fetch and display the data I wanted. In the end I am left with an app that allows the user to browse through or search for 150 Pokémon, and on interacting with the badge, open a modal with more detailed information for that Pokémon.

Moving forward, I hope to apply what I have learned from this project to create projects that are user-focused and enjoyable to use. Bringing my unique personality to the table and using my skillset to help tackle difficult problems with creative solutions.