**Code Challenge Time Log**

Explain Marvel API and the complexity of the md5 hash

Create app and install dependencies – 10 minutes

* Used react create-app Pokédex
* Installed:
* Axios for await async
* Emotion (explain if they don’t know) JSX

Create route and switch functions – 20 minutes

* Learnt mainly for navbar usage
* Inside app and not reloading full pages

Research [PokeApi](https://pokeapi.co/) & skim documentation – 45 minutes

* Support is good for PokéApi
* There is a second API for images
* Wanted the retro feel so used the original images
* Getting to know the API naming conventions

Create the main Pokédex page to retrieve Pokémon API information – 1 hour

* Started with pulling the API information to check I understand and I’m pulling correct info
* Used res console.log to check for data
* Created the state variables to be easy to reference
* Hooks and function components
* Worked out a way to generate a random number which equates to the Pokémon ID
* Put RanNum into the URL
* Created a window refresh button
* Set up the return structure using divs & names

Create the search Pokémon page – 1 hour

* Started with pulling the API information to check I understand and I’m pulling correct info
* Used res console.log to check for data
* Created the state variables to be easy to reference
* Googled search bar best methods
* Google how the PokéApi data is converted to English metrics and added via math.round (googled)
* Set up the return structure using div & class names for a table design like the old school Gameboy

Styling both pages – 1 hour

* Searched for the Pokémon colours
* Background like a poke ball using gradient
* Wanted the retro vibe, used old pixelated image
* Found an old Gameboy image

Final touches, favicon, background gradient and button styling – 15 minutes