Java - Poke Deck

Architecture

Logique

I wanted a sort of MVC CLI, with a clear separation between logic, actions, and data.

Classes:

Pokedeck

```
variables:
    - ArrayList<Card>
methodes:
    - getCards
    - addCard
    - removeCard
    - searchCards
    - save
```

Card

```
variables
    -
methodes:
    - setDescription
```

```
    - getDescription
```

Card subclasses:

- Pokemon
  ++ HP
  ++ Type
- Trainer
- Energy
  ++ Type

Actions

```
methodes:
    - displayCards
    - addCard
    - editCard
    - removeCard
    - searchCard
    - save
```

Menu

```
methodes:
    -displayMenu
    -doSomething (calls an action)
```

Analyse

This project's goal is to make a Pokedeck in Java. First, I had to learn a bit about the basic rules of Pokemon.
There are 3 types of cards:

- Pokemon cards
- Trainer cards
- Energy cards
  The logic for these cards is held in one Card superclass, so that I can store them in one ArrayList in the Pokedeck object.

For my Interface, I chose to have actions chosen by typing an associated number, instead of typing the action literally. I think that this way the navigation is clearer and simpler.

I had no trouble adding functionalities to my project so I think I had a good architecture overall.