

POKEMON APPLICATION

1. Pokémon API

<http://www.json-generator.com/api/json/get/cfxSFgvWxu?indent=2>

2. Resources

- <https://github.com/AsynctaskCoffee/PokeApi-pokedex>
- <https://raw.githubusercontent.com/Biuni/PokemonGO-Pokedex/master/pokedex.json>
- <https://github.com/alvareztech/Pokedex>
- <https://www.youtube.com/watch?v=r2ALBYx1JuY>

3. Introduction

Pokémon, also known as Pocket Monsters in Japan was developed by the franchise “The Pokémon Company” which is famous in developing frictional characters. The application contains the list of Pokémon characters and users can easily search their favourite characters from the application. User can also find the full detail about their favourite character. The REST API is used to fetch characters which are 151 in numbers. Each character is unique in nature according to their type and abilities.

4. Purpose

The goal of the application is to gather all Pokémon characters in one application so that user can easily access their favourite characters and their details

5. System Features

5.1. Functional Requirements

Criticality Scale	
<i>Non-Critical</i>	Non
<i>Less Critical</i>	Less
<i>Critical</i>	Normal
<i>Very Critical</i>	Very
<i>Extremely Critical</i>	Extreme

5.1.1. The application should fetch data from JSON file

- Description** – The application will be able to fetch data from the JSON file in which all the Pokémon characters are listed and also have details about each character.
- Criticality** - **Extreme**
- Technical issues** - The app must have Retrofit file to add url for fetching the data. Also the RESTApi is used to fetch the data from the list.
- Risks** - Low risk – the implementation of this is very straightforward.

- e. **Dependencies** – The device must have internet connection for accessing the JSON file.

5.1.2. The application should show the list of characters

- a. **Description** - The application will store a list of all the Pokémon characters from the JSON file in the list. This list should be accessible, editable, and persistent between application lifetimes.
- b. **Criticality** - **Extreme**
- c. **Technical issues** - The app must use JSON file for displaying the list.
- d. **Risks** - High risk - Technical implementation is relatively straightforward but JSON file must have all the data.
- e. **Dependencies** – JSON File

5.1.3. The list contains the picture and name of the Pokémon character.

- a. **Description** - The application can get the name and picture of Pokémon character from the JSON file.
- b. **Criticality** - **Extreme**
- c. **Technical issues** - The app must use list position to get the name and picture.
- d. **Risks** - Low risk - technical implementation is relatively straightforward.
- e. **Dependencies** – List of Pokémon

5.1.4. The application should be able to search the character from the list

- a. **Description** - The application can able to search the character from the list.
- b. **Criticality** - **Normal**
- c. **Technical issues** - The app must have list of Pokémon characters.
- d. **Risks** - Low risk - technical implementation is relatively straightforward.
- e. **Dependencies** – List of Pokémon

5.1.5. The application should show the details of the Pokémon character

- a. **Description** - The application can able to show all the details of Pokémon character.
- b. **Criticality** - **Normal**
- c. **Technical issues** - The app must have list of Pokémon characters.
- d. **Risks** - Low risk - technical implementation is relatively straightforward.
- e. **Dependency** – List of Pokémon

5.2. Hardware Interfaces

- Android Device
- Android Emulator

5.3. User Interfaces

- Front-end software: Android UI
- Back-end software: RESTAPI

Screenshots



5.4. Performance Requirements

5.4.1. System

The application will run on all Android devices API level greater than 28.

5.4.2. Response Time

- The application should take less than 4 seconds when running on an Android phone and less than 8 second when on an emulator or tablet.
- The application will run fine until the user begins to multi-task between 3 or more processes.
- The application should give response within 5 seconds.
- The application can fetch data within 2 seconds.

5.4.3. Scalability

The application will be able to scale to the size of android device according to their screen.

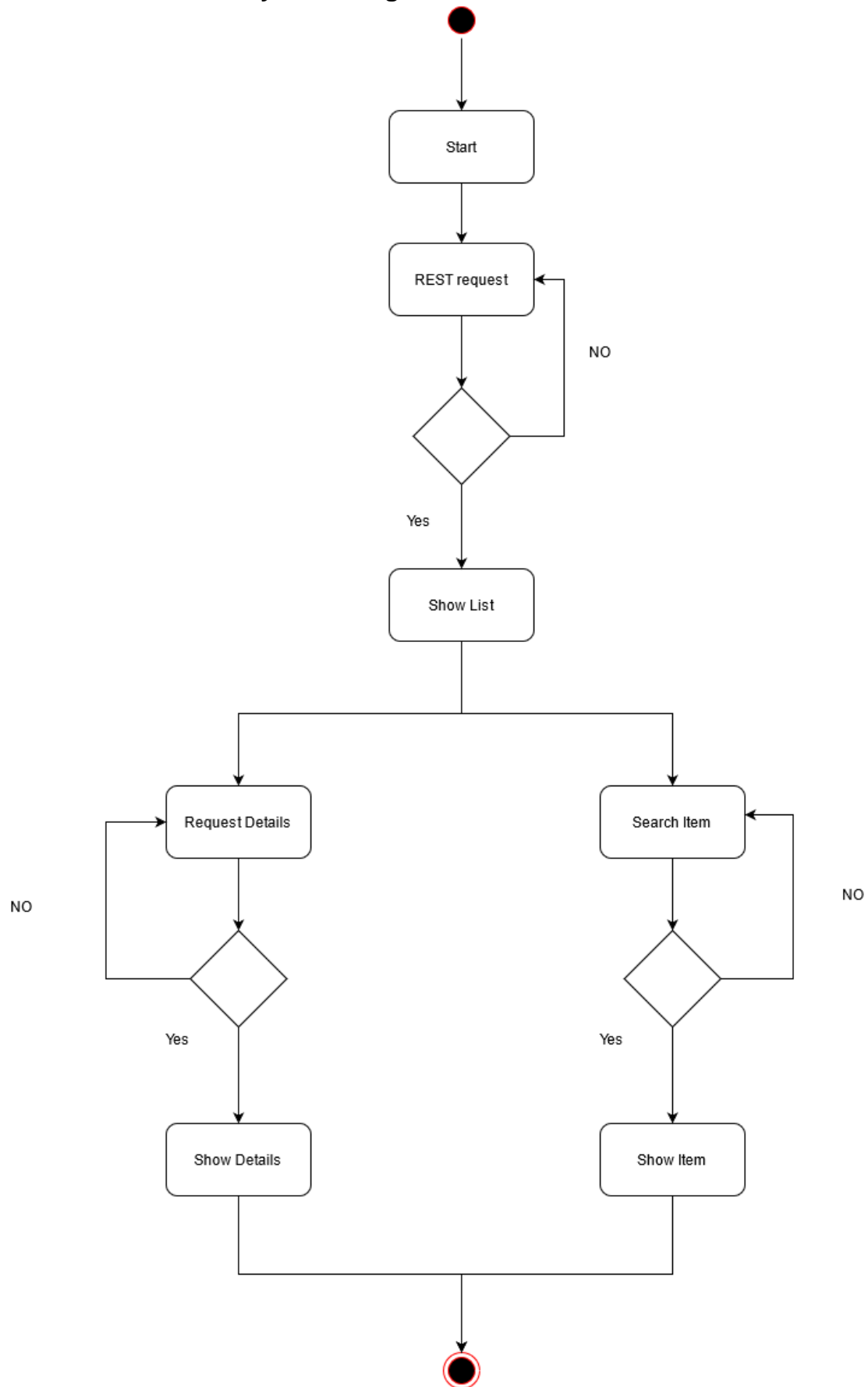
5.5. Security Requirements

- All the data which is used for the REST API must be secure.
- Only this request can access the JSON file.

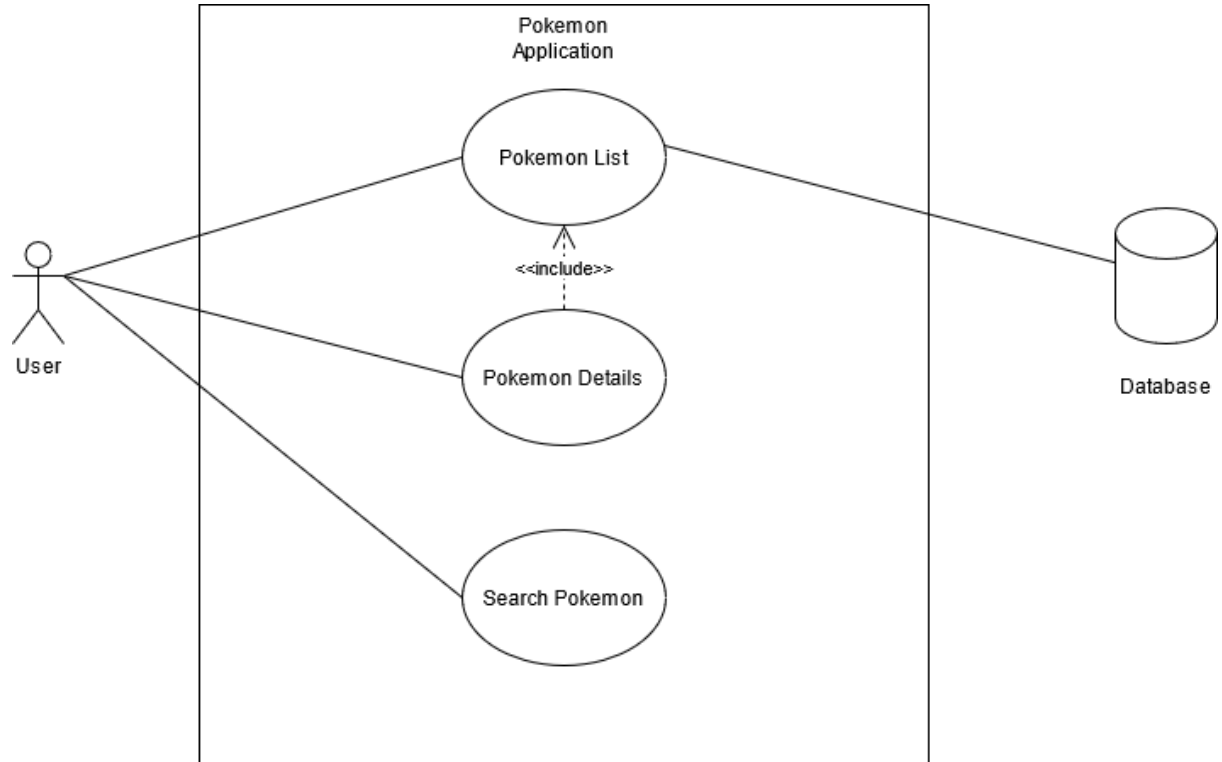
5.6. Software Quality Attributes

- Availability: The application and online API must be available 24/7.
- Correctness: The commands and user need must be fulfils in the correct manner.
- Maintainability
- Usability
- Adaptability
- Portability: Use anywhere

5.7. State/ Activity Flow Diagram



5.8. Use Case



Use Case Name	Pokémon List
Actor	User
Description	All actors will be able to access the list screen
Pre-Condition	Request data from JSON file.
Post-Condition	Get data from the JSON file.
Flow of events	Request using RESTApi and get data from the file.

Use Case Name	Pokémon Details
Actor	User
Description	All actors will be able to see the details screen.
Pre-Condition	Get data of Pokémon character from the list.
Post-Condition	Show details of each Pokémon.
Flow of events	Get data of Pokémon character and show in the details page.

Use Case Name	Search Pokémon
Actor	User
Description	All actors will be able to search Pokémon from the list.
Pre-Condition	Get data of Pokémon character from the list.
Post-Condition	Show search result of Pokémon requested.
Flow of events	Get data of Pokémon character from the list.

5.9. Sequence Diagram

