

# Manual tècnic del projecte final QA i testing

04.2024

---

Specialisterne

Raimon Targa

Daniel Urios

Marc Auladell

## Índex

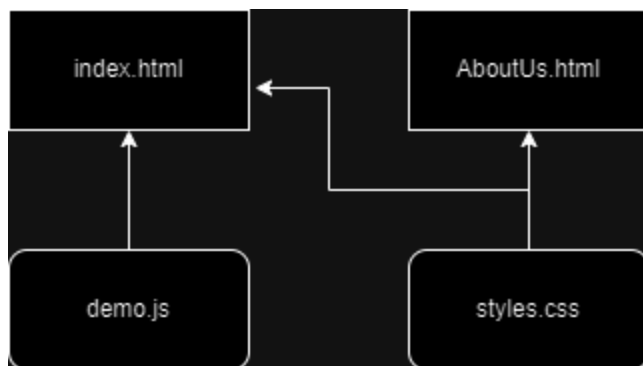
<b>Definició i especificació dels requeriments.....</b>	<b>2</b>
<b>Arquitectura del sistema.....</b>	<b>2</b>
<b>Disseny del model de dades.....</b>	<b>3</b>
API_URL.....	3
results ... banner.....	3
PAGE_SIZE.....	3
MAX_PAGES.....	3
MAX_ANIMATED_SPRITES.....	4
currentPage.....	4
queryResults.....	4
filteredResults.....	4
<b>Descripció de funcions del sistema.....</b>	<b>4</b>
filterElements().....	4
queryPage(pageNumber).....	4
displayElement(element).....	4
showElement(element).....	4
clearPage().....	4
nextPage().....	5
previousPage().....	5
goBack().....	5
updateResults().....	5

## Definició i especificació dels requeriments

Els requeriments inicials del projecte eren els següents:

- Una pàgina principal que contingui una llista dels pokemons. Aquesta llista mostra un número limitat de pokemons per cada pàgina, i ha de ser possible canviar la pàgina. S'ha de poder buscar pokemons pel seu nom utilitzant un camp de filtre i s'ha de poder seleccionar un pokemon amb un clic per veure la seva fitxa en la pàgina de detall.
- Una pàgina de detall on es puguin veure els detalls d'un pokemon específic. Inicialment només s'ha planejat implementar una quantitat limitada de detalls però es podrien afegir més.
- Una pàgina d'About Us que tingui alguna informació sobre nosaltres.

## Arquitectura del sistema



- La pàgina principal i la pàgina de detall estan dintre de index.html
- La pàgina d'About Us està dintre de AboutUs.html
- La lògica de la llista i la fitxa està dintre de demo.js, que s'executa desde index.html
- L'arxiu styles.css conté la informació de l'estil de les dues pàgines html

## Disseny del model de dades

```
const API_URL = "https://pokeapi.co/api/v2/pokemon"
const results = document.getElementById("results")
const previousButton = document.getElementById("previousButton")
const nextButton = document.getElementById("nextButton")
const searchField = document.getElementById('pokemonInput')
const backButton = document.getElementById("backButton")
const fichaPokemon = document.getElementById("fichaPokemon")
const nombrePokemon = document.getElementById("nombrePokemon")
const tipoPokemon = document.getElementById("numeroPokemon")
const alturaPokemon = document.getElementById("alturaPokemon")
const pesoPokemon = document.getElementById("pesoPokemon")
const imgPokemon = document.getElementById("imgPokemon")
const cryPokemon = document.getElementById("cryPokemon")
const banner = document.getElementById("banner")

const PAGE_SIZE = 25
const MAX_PAGES = 52
const MAX_ANIMATED_SPRITES = 649
let currentPage = 0
let queryResults = []
let filteredResults = []
```

### API\_URL

Conté la URL base sobre la qual fem crides.

### results ... banner

Contenen les referències a tots els elements html que necessitem interactuar per codi.

### PAGE\_SIZE

Conté el número d'elements màxims en una pàgina.

### MAX\_PAGES

Conté el número màxim de pàgines, que està marcat pel número total de pokemons que hi ha.

## MAX\_ANIMATED\_SPRITES

Conté el número d'sprites que estan animats (només existeixen sprites animats fins a la generació 5).

## currentPage

Conté el número de la pàgina actual (comença des de 0).

## queryResults

Conté tots els pokemons que s'han trobat realitzant la consulta a la API.

## filteredResults

Conté tots els pokemons que s'han filtrat utilitzant el camp de cerca.

## Descripció de funcions del sistema

### filterElements()

Filtra els elements de la llista utilitzant el camp de cerca

### queryPage(pageNumber)

Cerca una pàgina (de 25 elements) de l'API. És una funció recursiva que es crida fins que arriba a la última pàgina, i és asíncrona, però només es crida quan s'hagin recollit tots els resultats de la cerca anterior.

### displayElement(element)

Afegeix un element visualment a la llista.

### showElement(element)

Mostra la fitxa en detall d'un element seleccionat.

### clearPage()

Neteja la pàgina, eliminant tots els elements.

### nextPage()

Canvia la pàgina a la següent.

### previousPage()

Canvia la pàgina a l'anterior.

### goBack()

Torna a la pàgina principal, des de la pàgina de detall.

### updateResults()

Actualitza la llista que es mostra a pàgina principal en base a la pàgina actual i si hi ha un filtre o no.