
PROYECTO #2

PokeBattle TEC

Instituto Tecnológico de Costa Rica

Escuela de Ingeniería en Computación

Desarrollo de Aplicaciones para Dispositivos Móviles

Profesor: Andréi Fuentes

Alumnos:

Madriz Masis Gabriel Eduardo

Saenz Rojas Francisco Javier

Salazar Agüero Emmanuel Rodolfo

Documentación Externa Proyecto #2

PokeBattle TEC

Version 1.0

Lunes 13 de Octubre de 2014

CONTENIDO

Descripción del app.....	3
funcionalidades.....	3
User Experience and User Interface.....	4
Descripción de diseño de alto nivel.....	16
Descripción detallada	17
Descripción de los web services	19
interacción con sistemas externos.....	20

DESCRIPCIÓN DEL APP

La aplicación denominada PokeBattle TEC es una app estilo juego que utiliza el componente de hardware llamado Global Positioning System (GPS) como eje principal del mismo, PokeBattle TEC se conecta con tres Application Programming Interface los cuales son: Facebook API, Twitter API y un Pokemon API.

La aplicación con ayuda del GPS le mostrara distintos pokemons según la zona en la cual se encuentre ubicado brindando la posibilidad de luchar con dicho pokemon, cada pokemon aparecera de forma aleatoria en las distintas coordenadas geograficas y con distintos niveles. El usuario cuando inicie la app podra escoger cuatro pokemon entre los cuales podra escoger cual sera su acompañante durante cada batalla.

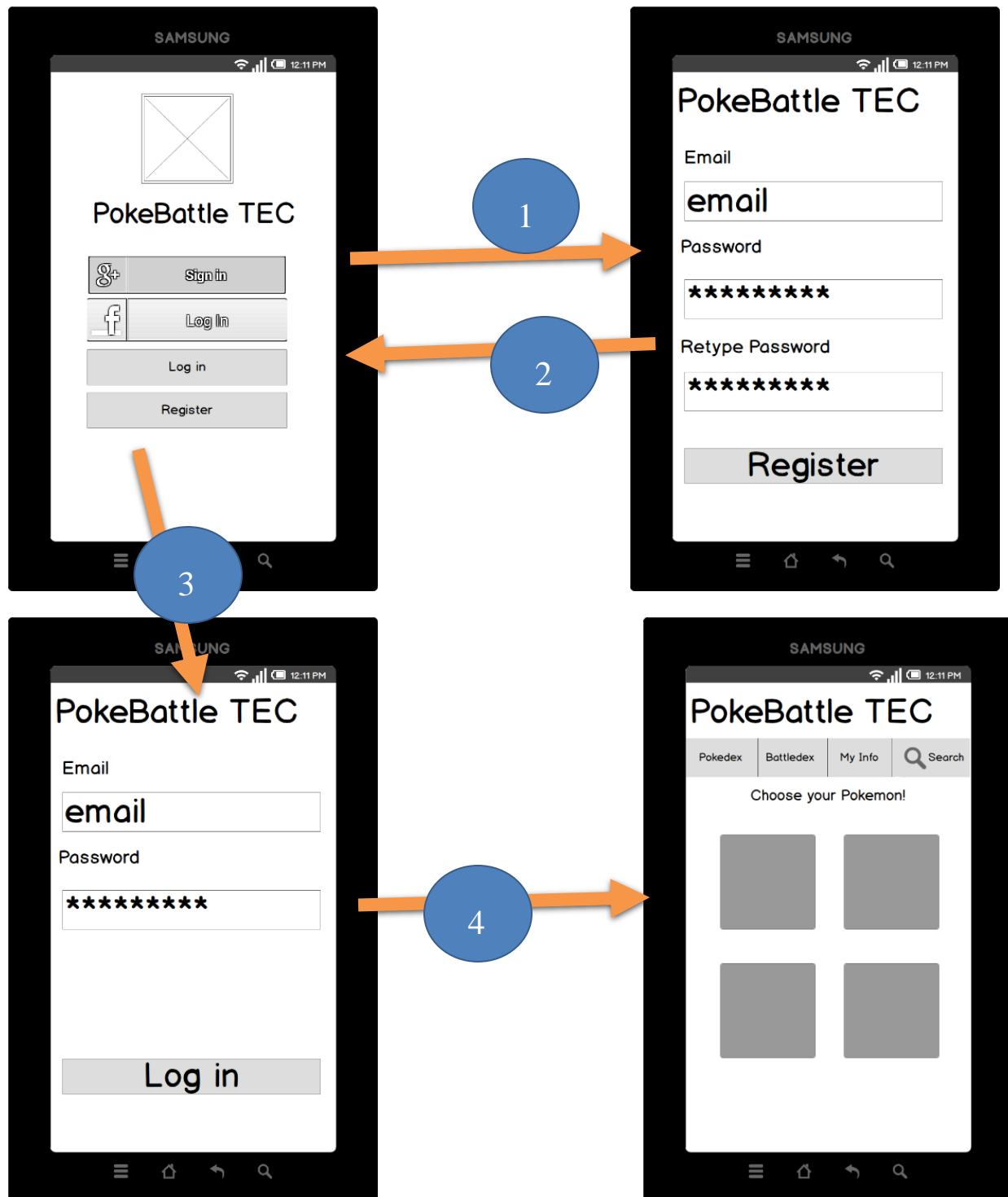
FUNCIONALIDADES

La aplicación tiene varias funcionalidades las cuales hacen que sea una app bastante atractiva para un público juvenil y adulto dado a la temática que se encuentra cubriendo. Las funcionalidades de la aplicación son:

- Luchar con Pokemon generados con nivel aleatorio encontrados según la ubicación del usuario en el GPS. Es decir que el usuario se encontrara en distintos lugares del Tecnológico de Costa Rica y según el lugar, puede ser que aparezca o no un Pokemon, en caso de aparecer se realizar un cálculo aleatorio para calcular el nivel de dicho Pokemon y seguido se procede a luchar con el Pokemon.
- Mover de forma aleatoria cada cierto tiempo (sin definir el tiempo) la posición de determinado Pokemon en determinada posición geográfica dentro del Tecnológico de Costa Rica.
- Llevar un control de cuantos Pokemon ha vencido el usuario y sus respectiva duración y fecha.
- Permitir al usuario escoger un Pokemon inicial.
- Visualizar una Pokedex, similar a los Pokedex del juego de Nintendo.
- Hacer búsquedas según el tipo del Pokemon.

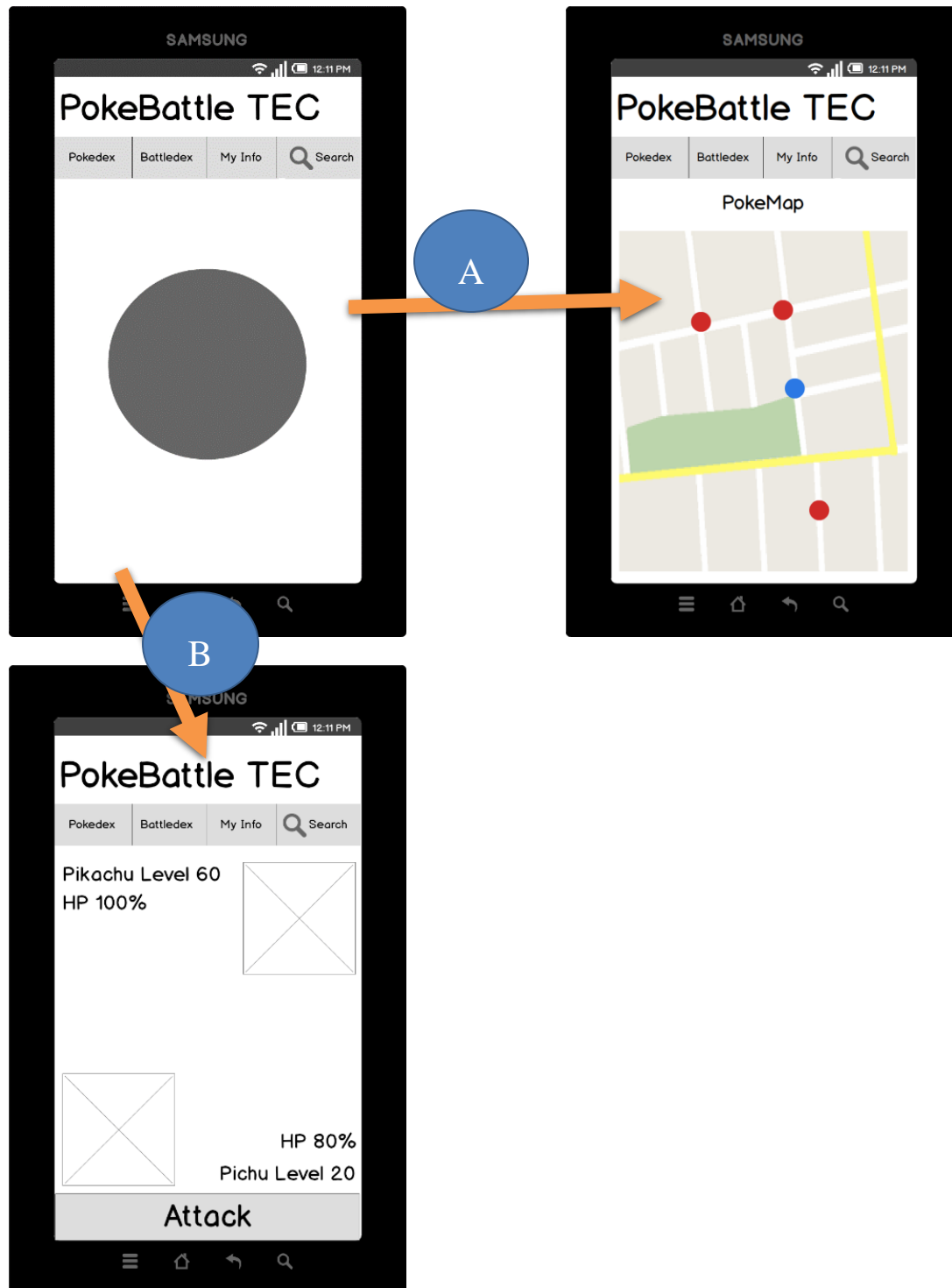
USER EXPERIENCE

INGRESAR SIN REGISTRO



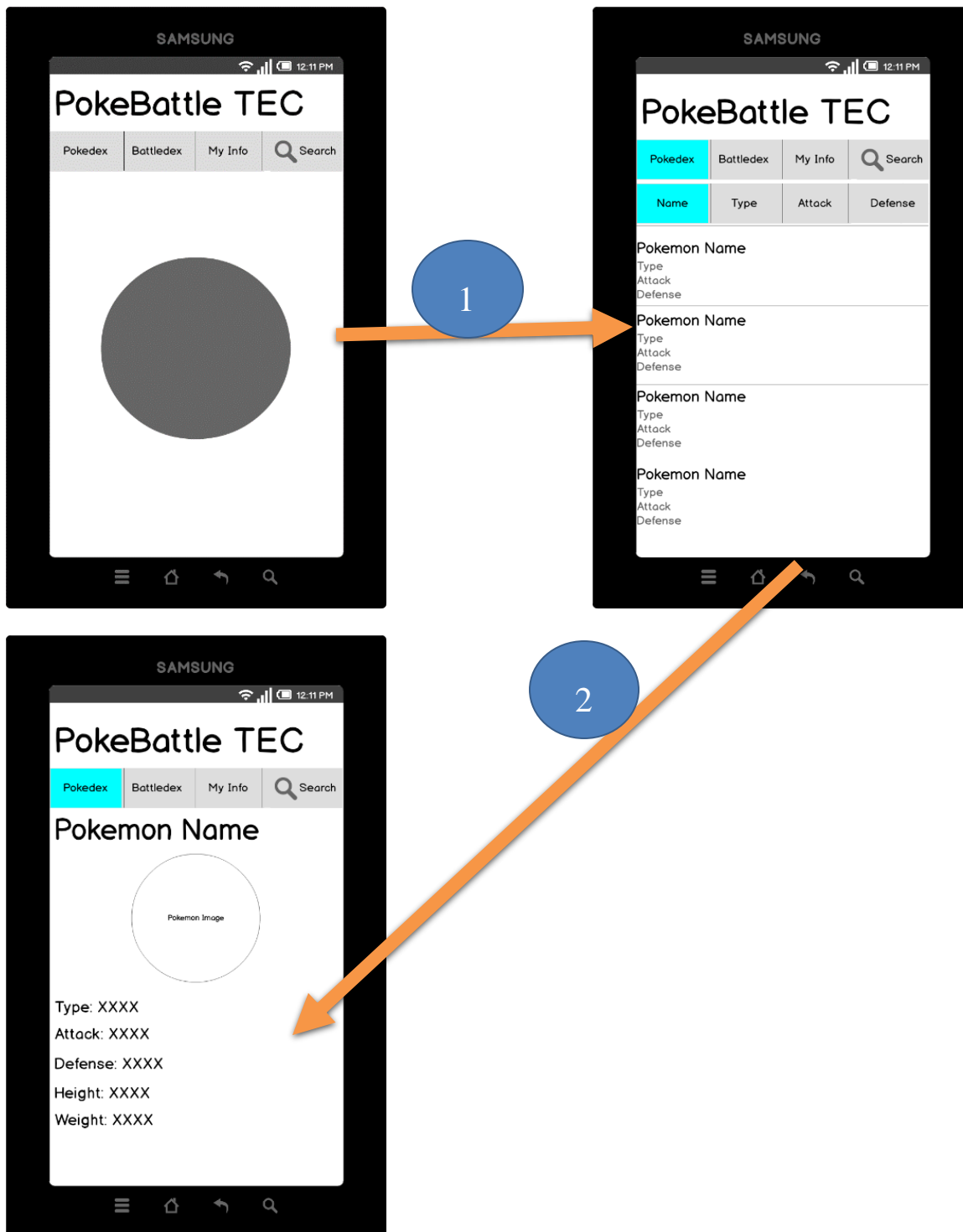
1. En caso de no contar con una cuenta de Facebook o Twitter o no querer usarlas, el sistema brindara la opción de registro donde únicamente se le pide un email y la contraseña.
2. Luego del registro se envía a la pantalla inicial para que el usuario ingrese.
3. El usuario ingresa al sistema.
4. Por ser la primare vez se le pedirá que escoge un Pokemon Inicial.

BUSCAR POKÉMON



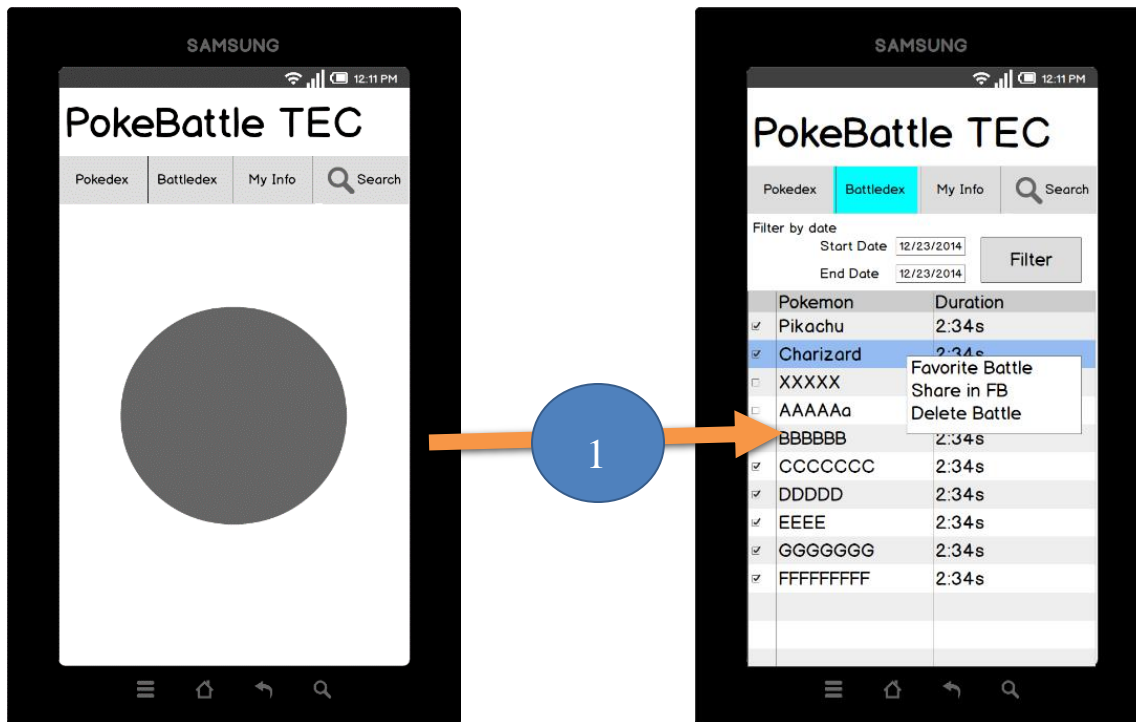
- A. En la pantalla principal tendrá un botón con la imagen de una Pokebola en la cual cuando se le toca se procede a hacer dos opciones, si no hay un Pokemon cerca procede a mostrar un mapa de los mismos.
- B. En caso de que el Pokemon se encuentre cerca se procede a mostrar la interfaz de batalla.

VER POKEDEX



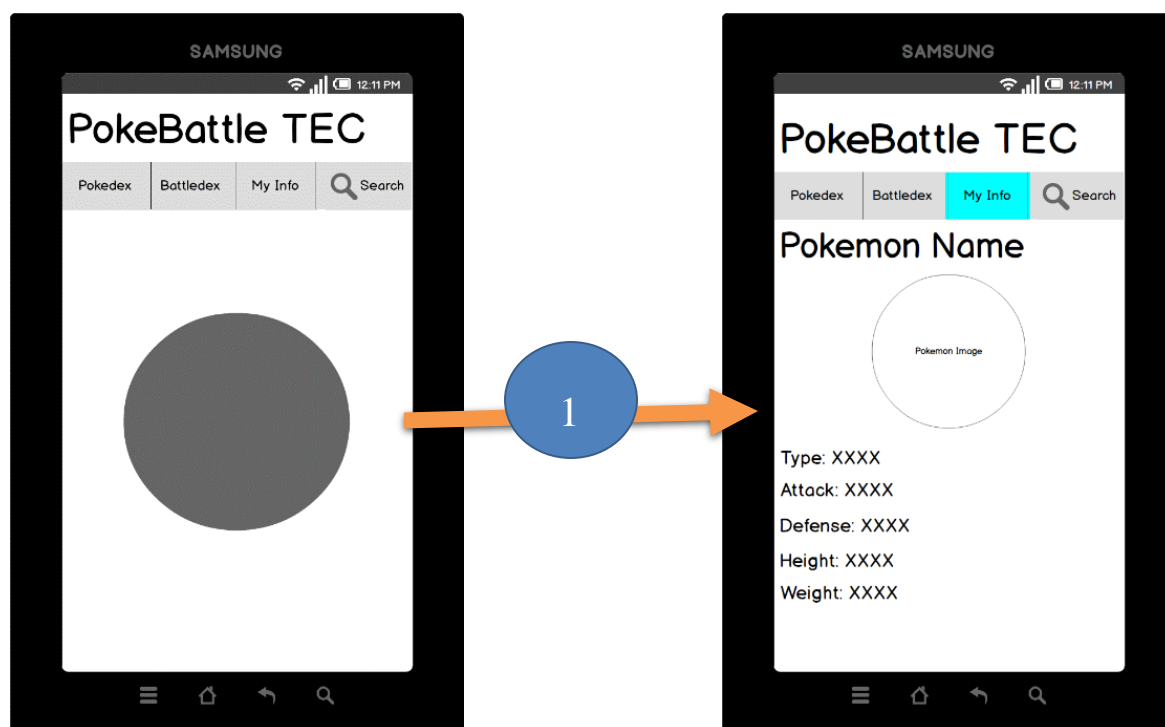
1. En el menú principal seleccione la pestaña llamada Pokedex, seguidamente se abre una lista con todos los Pokemon de la primera generación, solo se mostraran los Pokemon que el usuario haya visto, ya sea ganando o perdiendo.
2. Si se selecciona un Pokemon se desplegara toda la información referente a dicho Pokemon.

VER REGISTRO DE BATALLAS



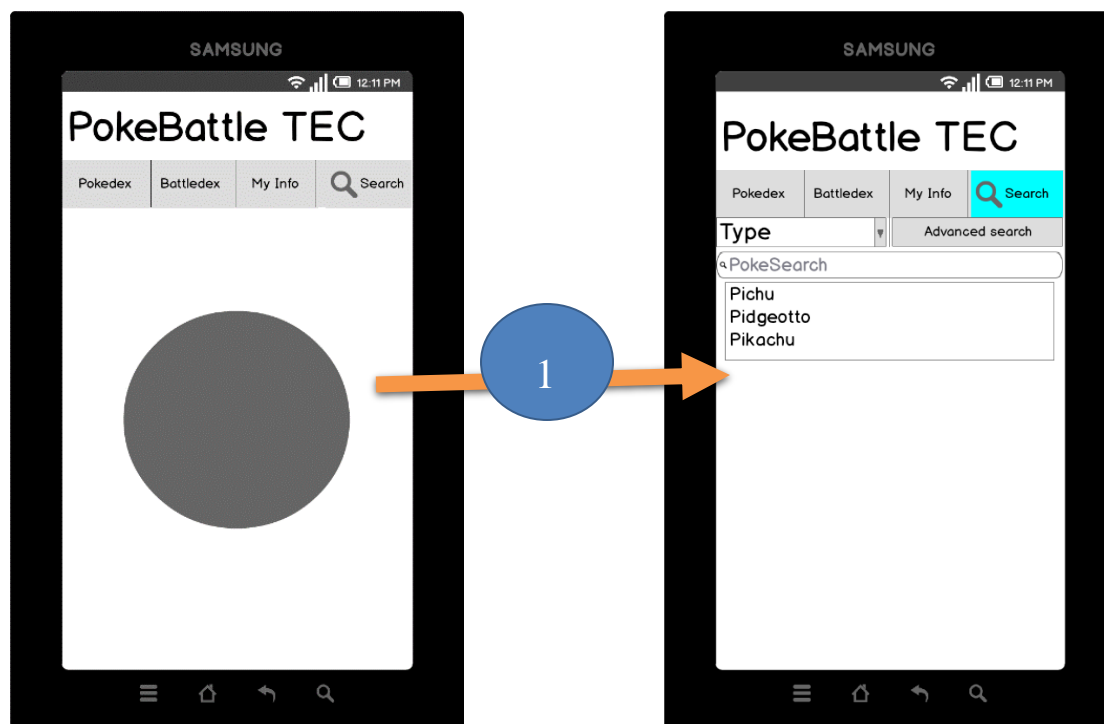
1. En el menú principal se selecciona la pestaña Battledex en la cual se desplegara la información sobre las batallas que ha tenido el usuario, datos como si lo derroto o perdió, nombre y la duración, los datos pueden ser filtrados por fecha.

VER DATOS DE MI POKEMON Y PERSONALES



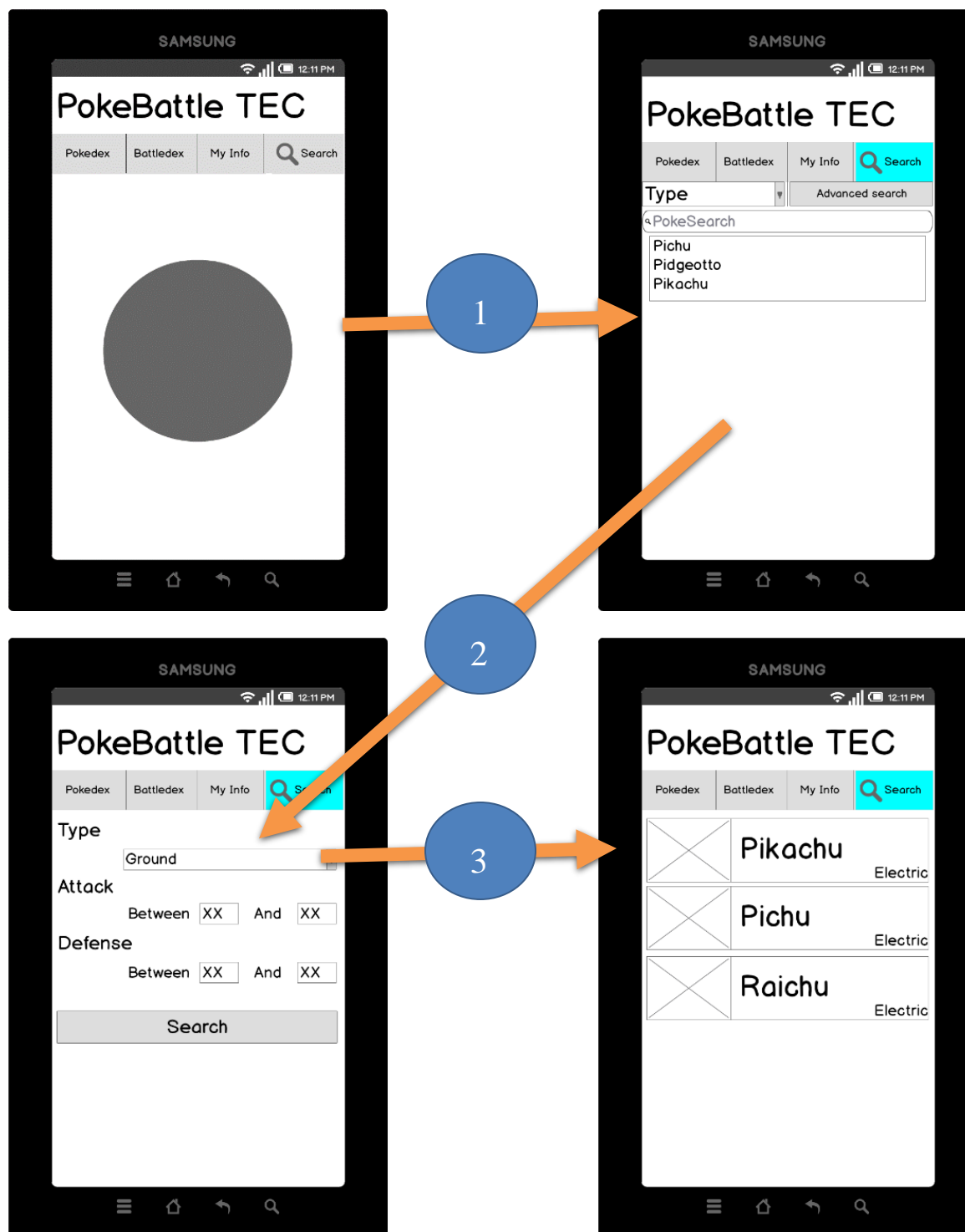
1. En el menú principal seleccione la pestaña My Info y se mostrara la información del usuario e información del Pokemon escogido como el tipo, ataque, entre otros.

HACER BÚSQUEDA SIMPLE



1. En el menú principal se selecciona la pestaña Search para hacer búsquedas, se muestra una pantalla donde se puede buscar por el tipo y el nombre.

HACER BÚSQUEDA AVANZADA

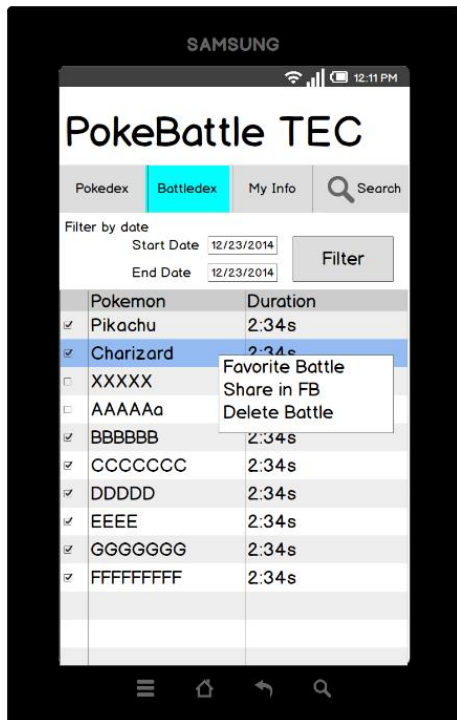


1. En el menú principal escoger la pestaña Search
2. Dentro de la búsqueda simple tocar el botón de búsqueda avanzada y llenar los parámetros
3. Se muestran los resultados de la búsqueda.

USER INTERFACE

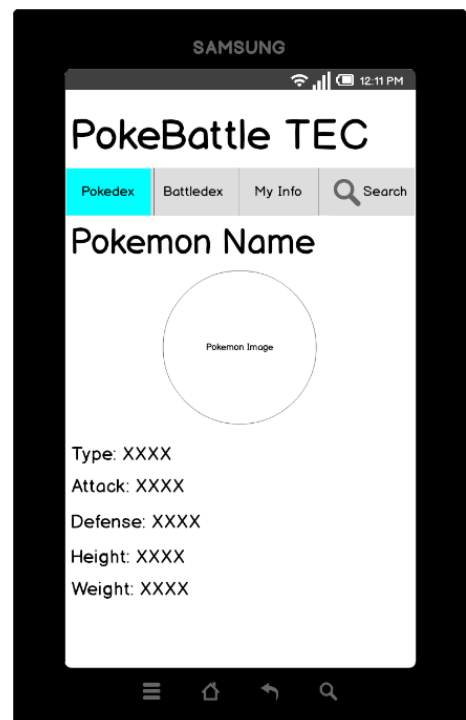
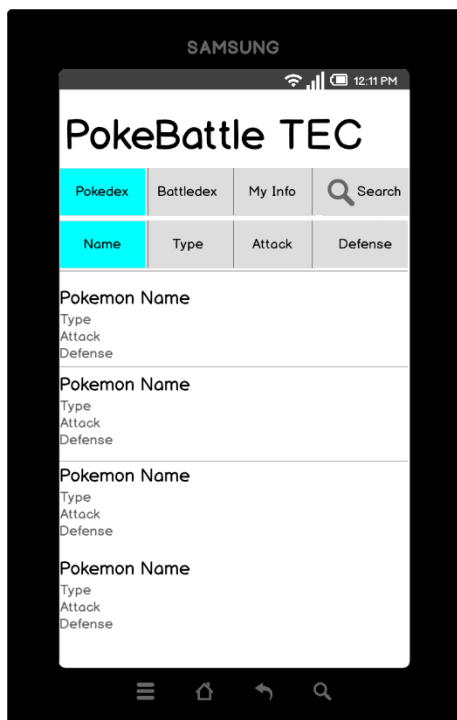
PATRÓN DE DESPLIEGUE DE DATOS

Tablas sencillas



Se considera que es tablas sencillas dado que utiliza encabezados fijos, tiene colores para cada fila y alinea el texto a la izquierda

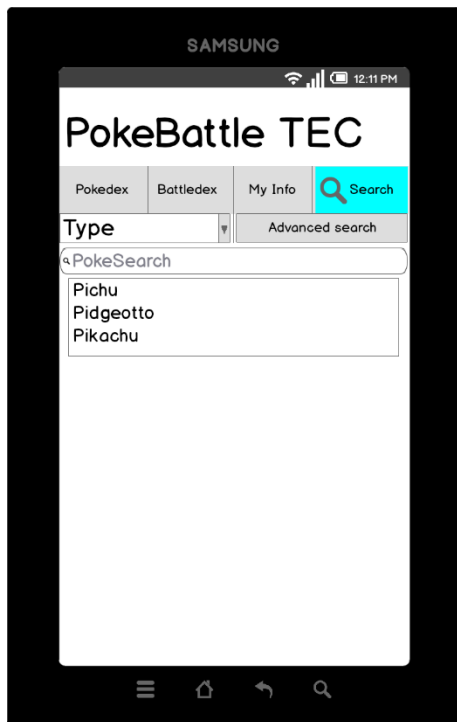
Master – Detail



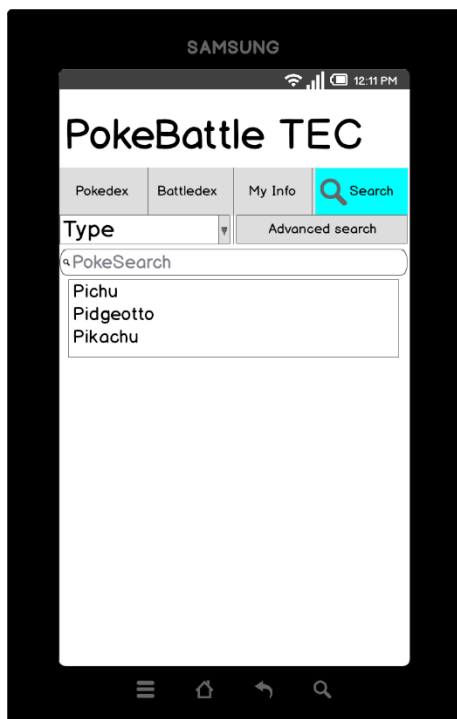
Se muestra una lista donde cuando el usuario ingresa en alguna opción se despliega el detalle.

PATRÓN DE BÚSQUEDA

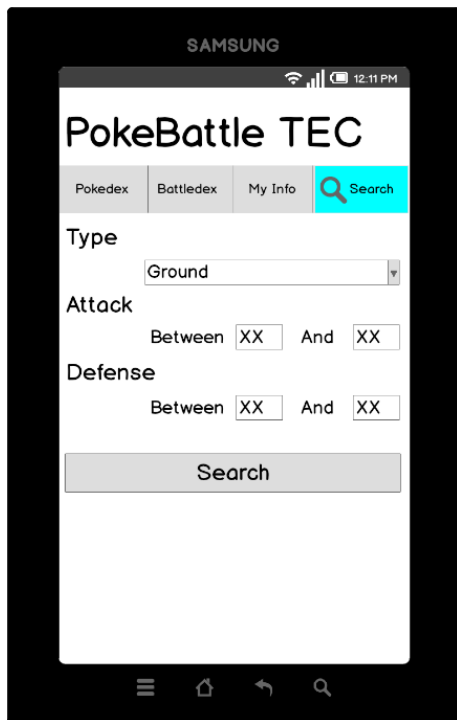
Autocompletado, dado que le sugiere al usuario la palabra que busca



Alcance, se puede buscar por determinado tipo

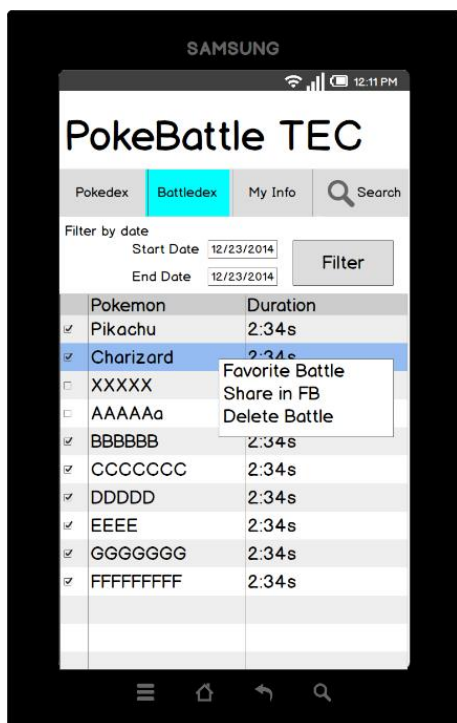


Formularios de Búsqueda, dado que se puede buscar por el nivel de ataque y defensa



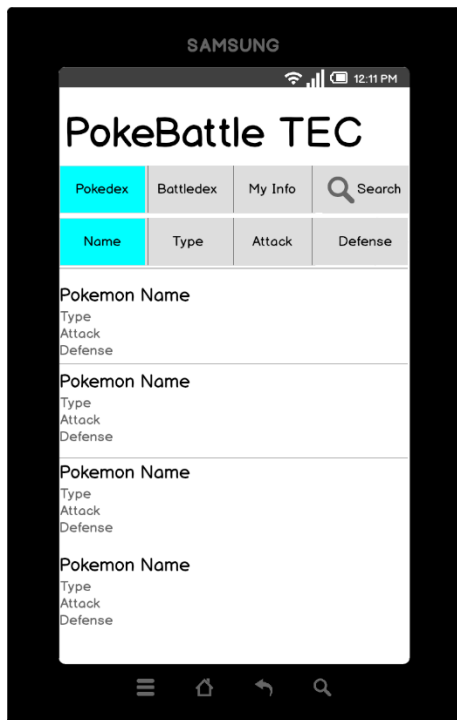
PATRÓN DE FILTRO

Filtro, dado que se puede filtrar las batallas por un intervalo de fechas.



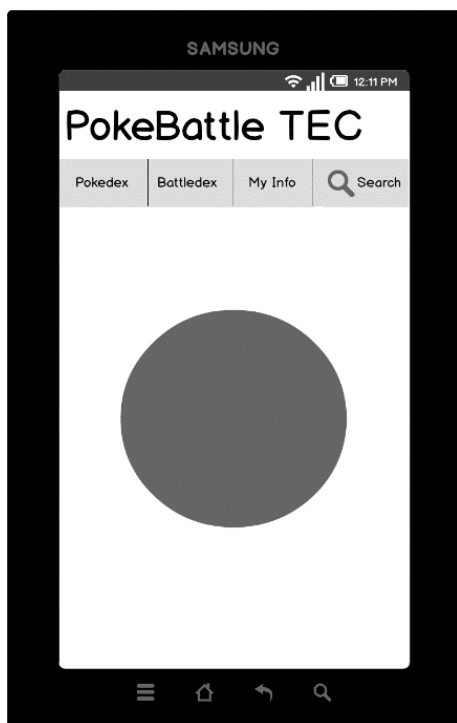
PATRÓN DE ORDENAMIENTO

Ordenamiento, se puede ordenar por nombre, tipo, ataque, y nivel de defensa

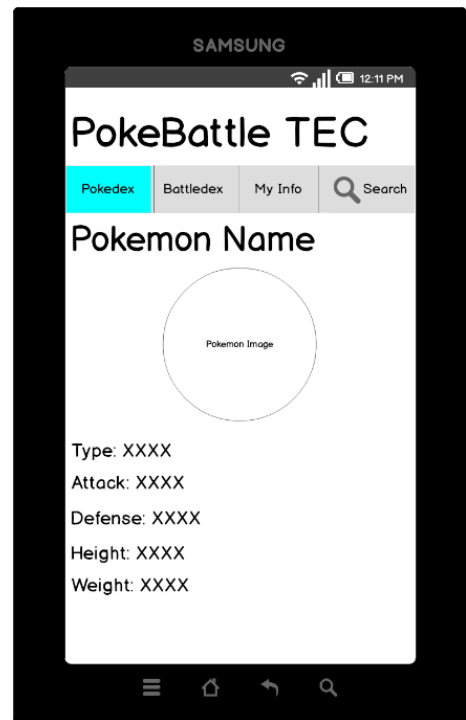
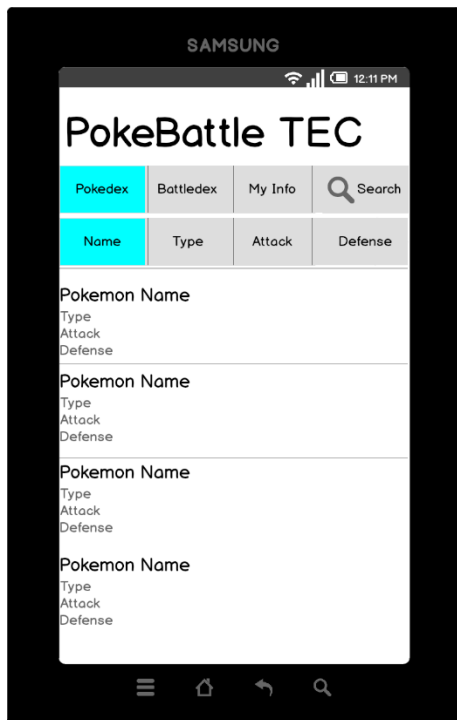


PATRÓN DE NAVEGACIÓN PRIMARIA

Pestañas, dado que se usan para mostrar, Pokedex, Battledex, My Info y Search



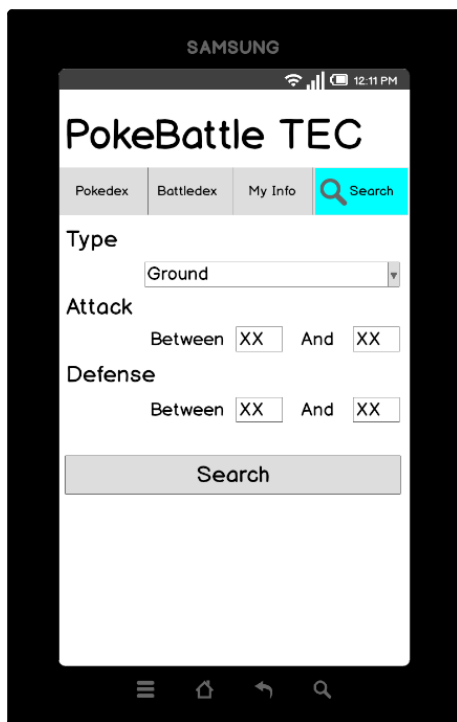
PATRÓN DE NAVEGACIÓN SECUNDARIA



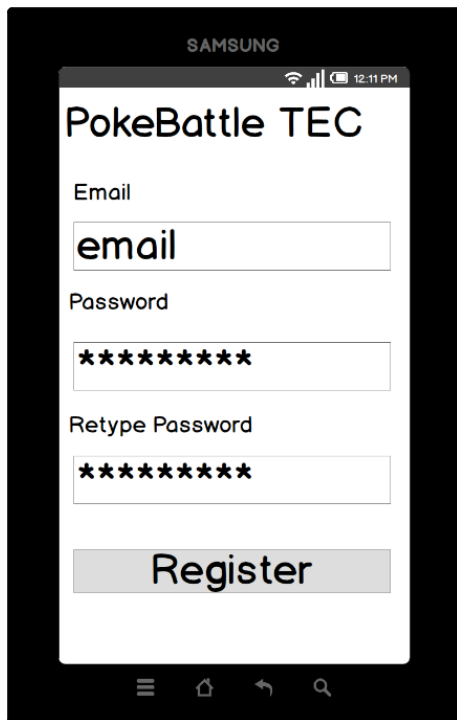
Se muestra una lista donde cuando el usuario ingresa en alguna opción se despliega el detalle.

PATRÓN DE DISEÑO DE FORMULARIOS

Búsqueda

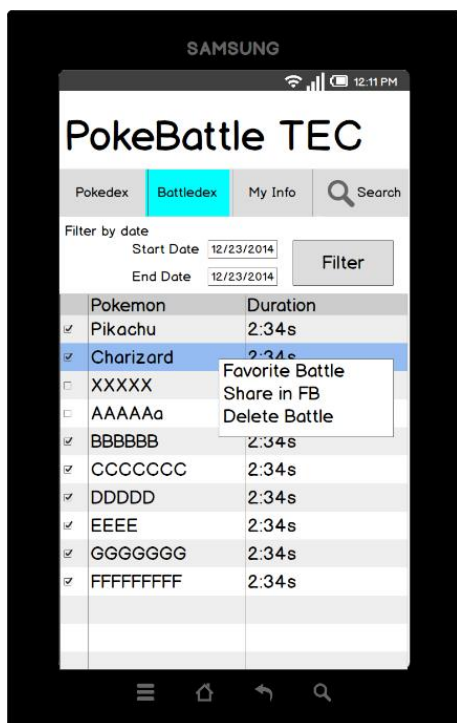


Registro



PATRÓN DE HERRAMIENTAS

Menú contextual, seleccionando una fila, se le desplegara un menú con funciones para ejecutar.



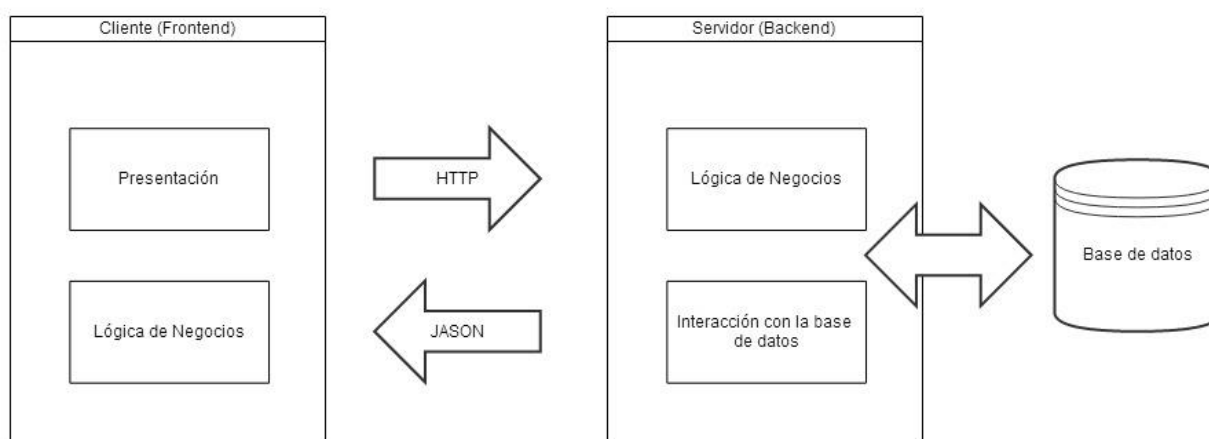
DESCRIPCIÓN DE DISEÑO DE ALTO NIVEL

La arquitectura en la que se desarrollará la aplicación es Mobile, en la cual se tendrá un cliente con dos capas una de presentación y otra con una pequeña lógica de negocios, por otro lado se tendrá un servidor donde estará una lógica de negocios más robusta y la capa de datos. El medio de

comunicación entre el cliente y el servidor será por medio de un API que será desarrollado por el equipo de trabajo.

Además se tendrá un par de API's externos, los cuales son el de Facebook para registrarse a la aplicación y postear resultados de las batallas, así como del API de Pokemon, para poder acceder a la información detallada de cada uno de los pokemon a utilizar en as batallas y por último Google Maps para la ubicación de los pokemon y de las personas.

A continuación se presenta el diagrama de la arquitectura que se usará:



CLIENTE

Al lado del cliente se cuenta con una capa de presentación y una de lógica de negocios. En el caso de la de presentación es meramente visual, es lo que el usuario podrá ver con lo que podrá interactuar como ya se vio en la sección de Wireframes, ahora bien aquí los datos son introducidos por éste y en pasan a la lógica de negocios del cliente donde hacen la función de captura de parámetros para su envío por medio de HTTP al backend.

SERVIDOR

En el caso del servidor posee una capa de lógica de negocios en la que se procesará toda la información procedente del cliente, además tiene una capa de enlace de datos con la base de datos en la cual se establecerá la interacción con la base. En ambos casos se manejan niveles de seguridad para la seguridad tanto del backend en sí mismo como de todos los datos que éste maneja. Posteriormente la información procesada y ordenes serán enviada por medio de un JSON al cliente.

DESCRIPCIÓN DETALLADA

Diagrama de Clases

Para el diagrama de clases se pensó en forma que fuese útil el manejo de los datos se están manejando constantemente, se crearon cuatro clases las cuales serán encargadas de manejar los datos del usuario como la información del Pokemon y email, también se encuentra la información de la Pokedex donde se puede ver cuales son los Pokémon que ya se han visualizado. Y se agrega una battledex la cual se encargara de manejar los datos sobre las batallas realizadas.

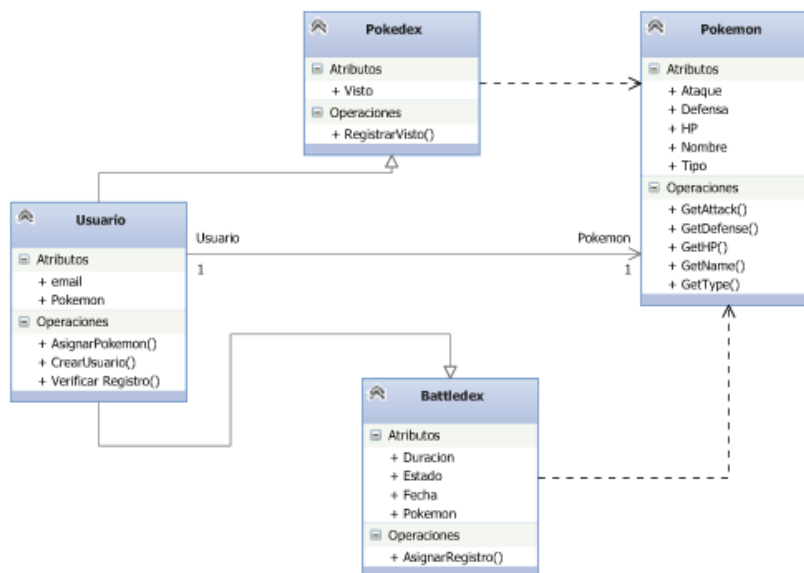
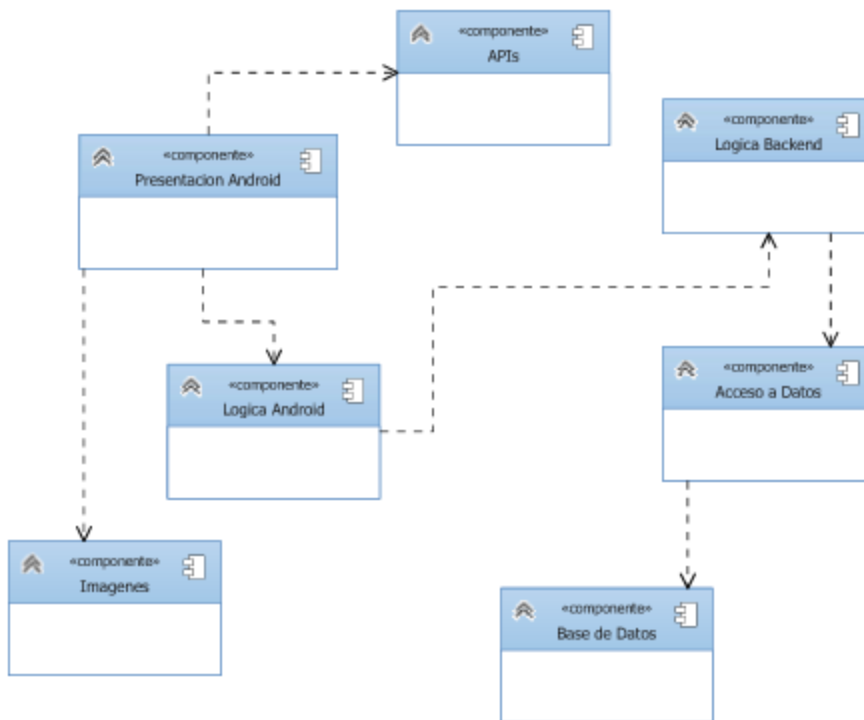


Diagrama de Componentes

Los componentes que componen el proyecto se basan en la presentación y logica en el lado del cliente y las imagenes, así como la conexión a las APIs, mientras que en el otro extremo se encuentra la logica y el acceso de datos en el backend. El backend será en NodeJS



DESCRIPCIÓN DE LOS WEB SERVICES

Pokedex de Usuario

Obtiene la información de los pokemon que el usuario ha enfrentado y ganado. Aquellos pokemon que no fueron vencidos, no se mostraran en el pokedex.

URL: pokebattleTEC.com/pokedex/id_user

Parámetros:

- id_user: el id del usuario especifico del cual se requiere el pokedex.

Método: GET

Actualizar zonas de Batallas

Actualiza las zonas de batallas en intervalos de tiempo predeterminados, para establecer las zonas donde hay pokemon en el mapa

URL: pokebattleTEC.com/zones

Método: GET

Registro de Usuario

Registra un nuevo usuario, que tendrá la capacidad de batallar contra pokemon.

URL: pokebattleTEC.com/user/new

Parámetros:

- JSON con datos del usuario.

Método: POST

Información de Usuario

Obtiene la información relevante del usuario.

URL: pokebattleTEC.com/user/id_user

Parámetros:

- id_user: El identificador del usuario específico.

Método: GET

Información de Batalla

Obtiene información pertinente a la batalla que va a iniciar, el pokemon que se va a enfrentar y los ataques de la batalla.

URL: pokebattleTEC.com/battle/new

Método: GET

Vencer Pokemon

Registra el resultado de la batalla, actualiza el pokedex del usuario y el estatus del usuario con respecto a la batalla.

URL: pokebattleTEC.com/battle/result

Parámetros:

- JSON con datos de la batalla

Método: POST

INTERACCIÓN CON SISTEMAS EXTERNOS

La aplicación PokeBattle TEC tendrá contacto con tres Application Programming Interface (API) las cuales son útiles para el registro del usuario y para conocer la información de los Pokemon.

FACEBOOK API

El Facebook API estará encargado de hacer ingreso a la aplicación por medio del sistema de validación de Facebook, además será el encargado de permitir compartir información sobre la batalla favorita.

La API no tiene limitaciones de uso que restrinjan su usabilidad a excepción de algunos datos de importancia que requieran que la aplicación deba ser evaluada por los sistemas de Facebook, los demás datos, los cuales son los que se proceden a utilizar son públicos y no requieren de permisos.

En relación a realizar publicación en el perfil de personal, se deberá pedir los respectivos permisos.

TWITTER API

El Twitter API cumplirá la misma función de hacer ingreso a la aplicación por medio del sistema de validación de Twitter para los usuarios que cuenten con una cuenta de Twitter. A excepción de la API de Facebook, no se utilizará el Twitter API para hacer publicaciones.

POKEMON API

El Pokémon API es un RESTful API que en un solo lugar permite acceder a toda la información sobre los Pokemon por medio de RESTful, el API será utilizado para conocer información sobre los Pokemon y demás información que sea importante conocer.

Un ejemplo de un llamado es: <http://pokeapi.co/api/v1/pokemon/1/>

```
{
  "abilities": [
    {
      "name": "chlorophyll",
      "resource_uri": "/api/v1/ability/34/"
    },
    {
      "name": "overgrow",
      "resource_uri": "/api/v1/ability/65/"
    }
  ],
  "attack": 49,
  "catch_rate": 0,
  "created": "2013-11-03T15:05:41.260678",
  "defense": 49,
  "descriptions": [
    {
      "name": "bulbasaur_gen_1",
      "resource_uri": "/api/v1/description/4/"
    },
    {
      "name": "bulbasaur_gen_1",
      "resource_uri": "/api/v1/description/5/"
    },
    {
      "name": "bulbasaur_gen_1",
```

```
        "resource_uri": "/api/v1/description/6/"
    },
    {
        "name": "bulbasaur_gen_2",
        "resource_uri": "/api/v1/description/7/"
    },
    {
        "name": "bulbasaur_gen_2",
        "resource_uri": "/api/v1/description/8/"
    },
    {
        "name": "bulbasaur_gen_3",
        "resource_uri": "/api/v1/description/9/"
    },
    {
        "name": "bulbasaur_gen_3",
        "resource_uri": "/api/v1/description/10/"
    },
    {
        "name": "bulbasaur_gen_3",
        "resource_uri": "/api/v1/description/11/"
    },
    {
        "name": "bulbasaur_gen_4",
        "resource_uri": "/api/v1/description/12/"
    },
    {
        "name": "bulbasaur_gen_4",
        "resource_uri": "/api/v1/description/13/"
    },
    {
        "name": "bulbasaur_gen_4",
        "resource_uri": "/api/v1/description/14/"
    },
    {
```

```

        "name": "bulbasaur_gen_5",
        "resource_uri": "/api/v1/description/15/"
    },
    {
        "name": "bulbasaur_gen_1",
        "resource_uri": "/api/v1/description/2/"
    },
    {
        "name": "bulbasaur_gen_1",
        "resource_uri": "/api/v1/description/3/"
    }
],
"egg_cycles": 0,
"egg_groups": [
    {
        "name": "Plant",
        "resource_uri": "/api/v1/egg/7/"
    },
    {
        "name": "Monster",
        "resource_uri": "/api/v1/egg/1/"
    }
],
"ev_yield": "1 special attack",
"evolutions": [
    {
        "level": 16,
        "method": "level_up",
        "resource_uri": "/api/v1/pokemon/2/",
        "to": "Ivysaur"
    }
],
"exp": 64,
"growth_rate": "medium slow",
"happiness": 0,

```

```

"height": "7",
"hp": 45,
"male_female_ratio": "87.5/12.5",
"modified": "2013-11-30T13:59:47.261100",
"moves": [
  {
    "learn_type": "tutor",
    "name": "Bind",
    "resource_uri": "/api/v1/move/20/"
  },
  {
    "learn_type": "tutor",
    "name": "Grass-pledge",
    "resource_uri": "/api/v1/move/520/"
  },
  {
    "learn_type": "machine",
    "name": "Echoed-voice",
    "resource_uri": "/api/v1/move/497/"
  },
  {
    "learn_type": "machine",
    "name": "Round",
    "resource_uri": "/api/v1/move/496/"
  },
  {
    "learn_type": "machine",
    "name": "Venoshock",
    "resource_uri": "/api/v1/move/474/"
  },
  {
    "learn_type": "egg move",
    "name": "Power-whip",
    "resource_uri": "/api/v1/move/438/"
  },

```



```

{
    "learn_type": "egg move",
    "name": "Sludge",
    "resource_uri": "/api/v1/move/124/"
},
{
    "learn_type": "tutor",
    "name": "String-shot",
    "resource_uri": "/api/v1/move/81/"
},
{
    "learn_type": "tutor",
    "name": "Knock-off",
    "resource_uri": "/api/v1/move/282/"
},
{
    "learn_type": "machine",
    "name": "Grass-knot",
    "resource_uri": "/api/v1/move/447/"
},
{
    "learn_type": "machine",
    "name": "Captivate",
    "resource_uri": "/api/v1/move/445/"
},
{
    "learn_type": "egg move",
    "name": "Leaf-storm",
    "resource_uri": "/api/v1/move/437/"
},
{
    "learn_type": "machine",
    "name": "Energy-ball",
    "resource_uri": "/api/v1/move/412/"
},

```

```

{
    "learn_type": "level up",
    "level": 37,
    "name": "Seed-bomb",
    "resource_uri": "/api/v1/move/402/"
},
{
    "learn_type": "level up",
    "level": 31,
    "name": "Worry-seed",
    "resource_uri": "/api/v1/move/388/"
},
{
    "learn_type": "machine",
    "name": "Natural-gift",
    "resource_uri": "/api/v1/move/363/"
},
{
    "learn_type": "egg move",
    "name": "Ingrain",
    "resource_uri": "/api/v1/move/275/"
},
{
    "learn_type": "egg move",
    "name": "Nature-power",
    "resource_uri": "/api/v1/move/267/"
},
{
    "learn_type": "egg move",
    "name": "Amnesia",
    "resource_uri": "/api/v1/move/133/"
},
{
    "learn_type": "egg move",
    "name": "Magical-leaf",

```

```

        "resource_uri": "/api/v1/move/345/"
    },
    {
        "learn_type": "machine",
        "name": "Bullet-seed",
        "resource_uri": "/api/v1/move/331/"
    },
    {
        "learn_type": "egg move",
        "name": "Grasswhistle",
        "resource_uri": "/api/v1/move/320/"
    },
    {
        "learn_type": "machine",
        "name": "Secret-power",
        "resource_uri": "/api/v1/move/290/"
    },
    {
        "learn_type": "machine",
        "name": "Facade",
        "resource_uri": "/api/v1/move/263/"
    },
    {
        "learn_type": "machine",
        "name": "Rock-smash",
        "resource_uri": "/api/v1/move/249/"
    },
    {
        "learn_type": "machine",
        "name": "Sludge-bomb",
        "resource_uri": "/api/v1/move/188/"
    },
    {
        "learn_type": "machine",
        "name": "Strength",

```

```

        "resource_uri": "/api/v1/move/70/"
    },
    {
        "learn_type": "machine",
        "name": "Sunny-day",
        "resource_uri": "/api/v1/move/241/"
    },
    {
        "learn_type": "machine",
        "name": "Hidden-power",
        "resource_uri": "/api/v1/move/237/"
    },
    {
        "learn_type": "level up",
        "level": 39,
        "name": "Synthesis",
        "resource_uri": "/api/v1/move/235/"
    },
    {
        "learn_type": "level up",
        "level": 25,
        "name": "Sweet-scent",
        "resource_uri": "/api/v1/move/230/"
    },
    {
        "learn_type": "egg move",
        "name": "Safeguard",
        "resource_uri": "/api/v1/move/219/"
    },
    {
        "learn_type": "machine",
        "name": "Frustration",
        "resource_uri": "/api/v1/move/218/"
    },
    {

```

```

        "learn_type": "machine",
        "name": "Return",
        "resource_uri": "/api/v1/move/216/"
    },
    {
        "learn_type": "machine",
        "name": "Sleep-talk",
        "resource_uri": "/api/v1/move/214/"
    },
    {
        "learn_type": "machine",
        "name": "Attract",
        "resource_uri": "/api/v1/move/213/"
    },
    {
        "learn_type": "machine",
        "name": "Fury-cutter",
        "resource_uri": "/api/v1/move/210/"
    },
    {
        "learn_type": "machine",
        "name": "Swagger",
        "resource_uri": "/api/v1/move/207/"
    },
    {
        "learn_type": "egg move",
        "name": "Charm",
        "resource_uri": "/api/v1/move/204/"
    },
    {
        "learn_type": "machine",
        "name": "Endure",
        "resource_uri": "/api/v1/move/203/"
    },
    {

```

```

        "learn_type": "machine",
        "name": "Giga-drain",
        "resource_uri": "/api/v1/move/202/"
    },
    {
        "learn_type": "machine",
        "name": "Mud-slap",
        "resource_uri": "/api/v1/move/189/"
    },
    {
        "learn_type": "machine",
        "name": "Protect",
        "resource_uri": "/api/v1/move/182/"
    },
    {
        "learn_type": "machine",
        "name": "Curse",
        "resource_uri": "/api/v1/move/174/"
    },
    {
        "learn_type": "machine",
        "name": "Snore",
        "resource_uri": "/api/v1/move/173/"
    },
    {
        "learn_type": "machine",
        "name": "Flash",
        "resource_uri": "/api/v1/move/148/"
    },
    {
        "learn_type": "egg move",
        "name": "Skull-bash",
        "resource_uri": "/api/v1/move/130/"
    },
    {

```

```

        "learn_type": "egg move",
        "name": "Light-screen",
        "resource_uri": "/api/v1/move/113/"
    },
    {
        "learn_type": "machine",
        "name": "Defense-curl",
        "resource_uri": "/api/v1/move/111/"
    },
    {
        "learn_type": "egg move",
        "name": "Petal-dance",
        "resource_uri": "/api/v1/move/80/"
    },
    {
        "learn_type": "machine",
        "name": "Headbutt",
        "resource_uri": "/api/v1/move/29/"
    },
    {
        "learn_type": "egg move",
        "name": "Razor-wind",
        "resource_uri": "/api/v1/move/13/"
    },
    {
        "learn_type": "machine",
        "name": "Substitute",
        "resource_uri": "/api/v1/move/164/"
    },
    {
        "learn_type": "machine",
        "name": "Rest",
        "resource_uri": "/api/v1/move/156/"
    },
    {

```

```

        "learn_type": "machine",
        "name": "Bide",
        "resource_uri": "/api/v1/move/117/"
    },
    {
        "learn_type": "machine",
        "name": "Reflect",
        "resource_uri": "/api/v1/move/115/"
    },
    {
        "learn_type": "machine",
        "name": "Double-team",
        "resource_uri": "/api/v1/move/104/"
    },
    {
        "learn_type": "machine",
        "name": "Mimic",
        "resource_uri": "/api/v1/move/102/"
    },
    {
        "learn_type": "machine",
        "name": "Rage",
        "resource_uri": "/api/v1/move/99/"
    },
    {
        "learn_type": "machine",
        "name": "Toxic",
        "resource_uri": "/api/v1/move/92/"
    },
    {
        "learn_type": "level up",
        "level": 41,
        "name": "Sleep-powder",
        "resource_uri": "/api/v1/move/79/"
    },
    },

```



```

{
    "learn_type": "level up",
    "level": 20,
    "name": "Poisonpowder",
    "resource_uri": "/api/v1/move/77/"
},
{
    "learn_type": "level up",
    "level": 48,
    "name": "Solarbeam",
    "resource_uri": "/api/v1/move/76/"
},
{
    "learn_type": "level up",
    "level": 27,
    "name": "Razor-leaf",
    "resource_uri": "/api/v1/move/75/"
},
{
    "learn_type": "level up",
    "level": 34,
    "name": "Growth",
    "resource_uri": "/api/v1/move/74/"
},
{
    "learn_type": "level up",
    "level": 7,
    "name": "Leech-seed",
    "resource_uri": "/api/v1/move/73/"
},
{
    "learn_type": "machine",
    "name": "Mega-drain",
    "resource_uri": "/api/v1/move/72/"
},

```

```

{
    "learn_type": "level up",
    "level": 1,
    "name": "Growl",
    "resource_uri": "/api/v1/move/45/"
},
{
    "learn_type": "machine",
    "name": "Double-edge",
    "resource_uri": "/api/v1/move/38/"
},
{
    "learn_type": "machine",
    "name": "Take-down",
    "resource_uri": "/api/v1/move/36/"
},
{
    "learn_type": "machine",
    "name": "Body-slam",
    "resource_uri": "/api/v1/move/34/"
},
{
    "learn_type": "level up",
    "level": 1,
    "name": "Tackle",
    "resource_uri": "/api/v1/move/33/"
},
{
    "learn_type": "level up",
    "level": 13,
    "name": "Vine-whip",
    "resource_uri": "/api/v1/move/22/"
},
{
    "learn_type": "machine",

```

```

        "name": "Cut",
        "resource_uri": "/api/v1/move/15/"
    },
    {
        "learn_type": "machine",
        "name": "Swords-dance",
        "resource_uri": "/api/v1/move/14/"
    }
],
"name": "Bulbasaur",
"national_id": 1,
"pkdx_id": 1,
"resource_uri": "/api/v1/pokemon/1/",
"sp_atk": 65,
"sp_def": 65,
"species": "seed pokemon",
"speed": 45,
"sprites": [
    {
        "name": "bulbasaur",
        "resource_uri": "/api/v1/sprite/2/"
    },
    {
        "name": "bulbasaur",
        "resource_uri": "/api/v1/sprite/1/"
    }
],
"total": 0,
"types": [
    {
        "name": "poison",
        "resource_uri": "/api/v1/type/4/"
    },
    {
        "name": "grass",

```

```
        "resource_uri": "/api/v1/type/12/"
    }
],
"weight": "69"
}
```