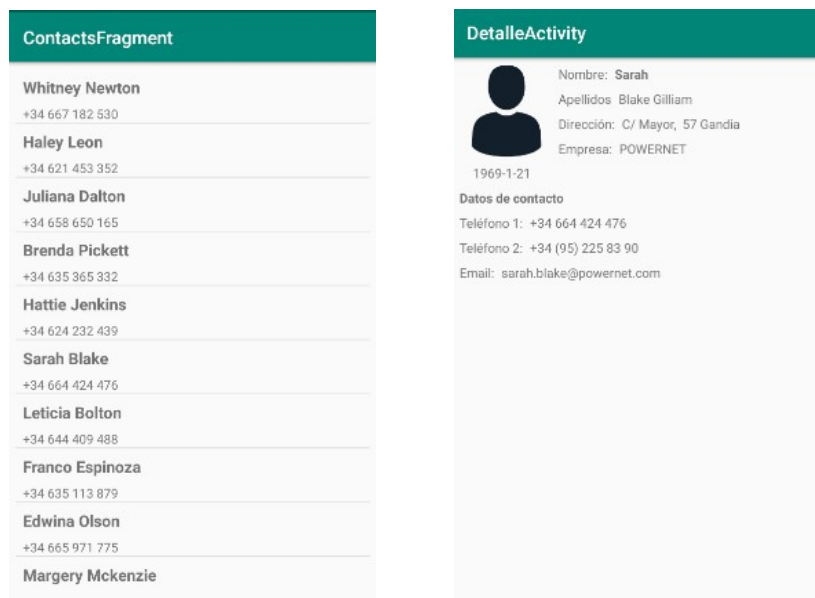


# EXERCICI

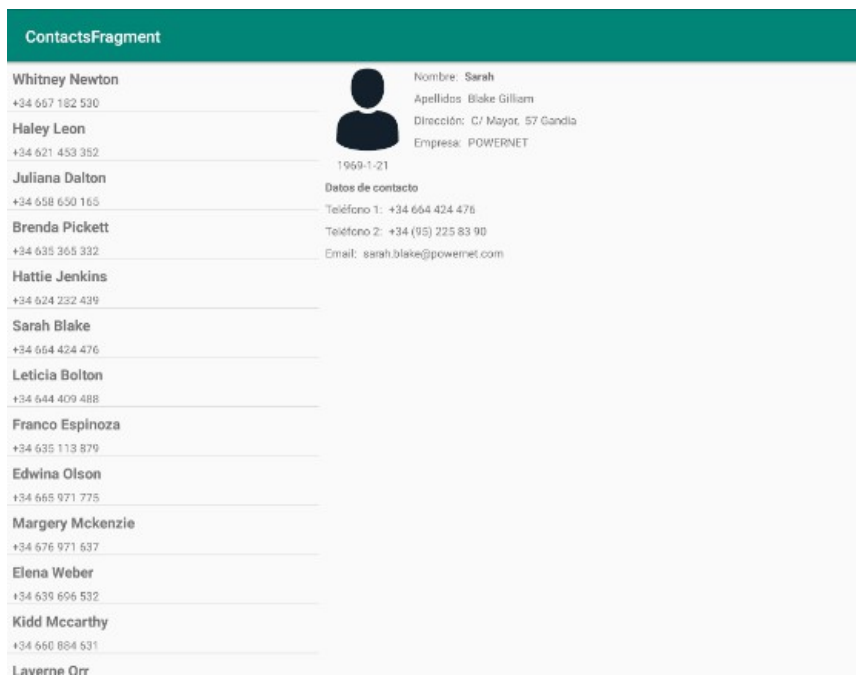
## FRAGMENTS

L'objectiu d'aquest exercici és mostrar una llista de contactes, procedent d'un arxiu amb format **JSON**, en una interfície gràfica que s'adapte bé a l'orientació i la grandària de la pantalla del dispositiu reaprofitant, en la mesura del possible, codi i layouts.


Per dur a terme aquesta tasca una de les millors opcions que tenim és l'ús de **Fragment**. Recordem que un **Fragment** és una porció de la interfície d'usuari que pot afegir-se o eliminar-se de forma independent a la resta d'elements d'una **Activity**.



En **smartphones** tenim dos Activities.



Visualització en **tablets** (horizontal)

ContactsFragment	
+34 621 453 352	
Juliana Dalton	
+34 658 650 165	
Brenda Pickett	
+34 635 365 332	
Hattie Jenkins	
+34 624 232 439	
Sarah Blake	
+34 664 424 476	
Leticia Bolton	
+34 644 409 488	
Franco Espinoza	
+34 635 113 879	
Edwina Olson	
+34 665 971 775	
	Nombre: Sarah Apellidos: Blake Gilliam Direcció: C/ Mayor, 57 Gandia Empresa: POWERNET 1989-1-21
Datos de contacto	
Teléfono 1: +34 664 424 476	
Teléfono 2: +34 (95) 225 83 90	
Email: sarah.blake@powernet.com	

Visualització en **tablets** (vertical)

Com podem apreciar a les imatges, depenen de l'orientació i la grandària de la pantalla del dispositiu mostrarem un o dos **fragments** amb disposicions horitzontals o verticals.

### Recursos:

A la pàgina <http://moodle.germangascon.com> podeu trobar els recursos necessaris per fer aquest exercici.

- Arxiu amb les dades dels contactes, [contacts.json](#).
- Imatge per defecte del perfil dels contactes, [profile.png](#). Si vols pots utilitzar una altra imatge que t'agrada més.

### Passos a seguir:

- Crear un **nou projecte** que podem anomenar *ContactsFragment*.
- Crear els **layouts** dels dos **Fragment**, un per al llistat i altre per a la vista detall.
- Crear el **model** de dades **Contacto**.
- Crear el **layout** que contindrà els elements del llistat. Pots utilitzar **ListView** o **RecyclerView**, es deixa a elecció de l'alumne.
- Crear l'**adaptador** per a mostrar dels dades al llistat.
- Crear un **parser** per a importar les dades del arxiu JSON a un array del nostre model.
  - Recorda que mitjançant el **Context** podem obtenir els recursos de l'aplicació amb el mètode **getResources()**. Una vegada hem accedit als recursos podem obrir un recurs que es trobe en la carpeta **raw** fent ús del mètode **openRawResource(R.raw.contacts)**. Aquest mètode torna un **InputStream** per tant podem treballar com ho fem de forma habitual en Java.
  - El mètode **available()** d'un **InputStream** torna la grandària del arxiu.
  - El mètode **read(buffer)** llig l'arxiu i deixa el bytes llegits en el buffer.
  - Podem transformar un array de bytes a un **String**, creant un objecte de la classe **String** indicant-li en el constructor la codificació de caràcters. Ex:

- String json = new String(buffer, "UTF-8");
- Una vegada hem llegit l'arxiu i carregat a un String podem fer ús de les classes [JSONTokener](#), [JSONArray](#) i [JSONObject](#) per a interpretar cadascun dels contactes que hi ha a l'arxiu [contacts.json](#).
    - [JSONTokener](#) interpreta l'arxiu JSON i el deixa preparat per poder llegir els objectes.
      - Si l'objecte que conté es tracta d'un array hem d'interpretar-lo amb [JSONArray](#). Podem recorre-lo igual que un array normal.
      - Si l'objecte que conté es un objecte simple, podem llegir-lo amb [JSONObject](#). Aquesta classe té diferents mètodes per a llegir tipus bàsics:
        - [getInt\(nom\)](#)
        - [getString\(nom\)](#)
        - etc..
  - Crear la classe per al [Fragment](#) que mostra el **llistat** estenent la classe [Fragment](#). Recorda que has de redefinir els mètodes [onCreateView\(...\)](#) i [onActivityCreated\(...\)](#).
  - Crear la classe per al [Fragment](#) que mostra el **detall** estenent la classe [Fragment](#). Al igual que em vist a la teoria, crearem el mètode [mostrarDetalle\(\)](#) però en aquest cas rebrà un objecte de la classe [Contacto](#) com a paràmetre.
  - Crear els **layout** per a les **Activities**. La nostra aplicació tindrà dos Activities:
    - L'[Activity](#) principal en smartphones mostrarà només el llistat però en tablets mostrarà tot el contingut.
    - La segona [Activity](#) només serà utilitzada pels smartphones per mostrar el detall.
    - Recorda que els layouts s'anomenaran tots iguals, el que els distingirà serà la carpeta on els ubiquem.
      - [/res/layout](#) (dispositius mòbils)
      - [/res/layout-large](#) (tablets disposició horitzontal)
      - [/res/layout-large-port](#) (tablets disposició vertical)
  - Crear la interfície (**Interface**) [IContactoListener](#) amb el mètode [onContactoSeleccionado\(\)](#).
  - Implementar el [onItemClickListener](#) del [Fragment](#) que mostra el llistat de forma que faci una cridada al mètode de la interfície [onContactoSeleccionado\(\)](#) tal i com hem vist a la teoria.
  - Fer que l'[Activity](#) principal implemente la interfície [IContactoListener](#) i en el mètode [onContactoSeleccionado\(\)](#) fer que s'actualitzen les dades del [Fragment](#) detall (si està carregat) o bé que es llance l'[Activity](#) detall (en cas que no estiga carregat el [Fragment](#) detall).
    - Per poder passar dades complexes, en aquest cas un objecte [Contacto](#), mitjançant un Intent hem de fer que la nostra classe [Contacto](#) implemente la interfície [Serializable](#).
  - Per últim només ens queda implementar l'[Activity](#) que mostra el detall. Podem fer ús del mètode [getSerializableExtra\(\)](#) per obtenir les dades del contacte sobre el que s'ha fet click.

Documents a lliurar:

- **MainActivity.java** (Activity principal de l'aplicació)
- **DetalleActivity.java** (Activity detall)
- **Contacto.java** (El model de dades)
- **ContactParser.java** (Per importar contactes amb format JSON).
- **FragmentListado.java** (El Fragment que conté el llistat)
- **FragmentDetalle.java** (El Fragment que conté el detall)
- **IContactosListener.java** (La interfície per manejar esdeveniments onClick)

- **AdaptadorContactos.java** (Adaptador personalitzat)
- **Tots els layouts de l'aplicació.**