



Defensa de la práctica final

Desarrollo de Sistemas Software basados
en Componentes y Servicios

Juan Carlos Serrano Pérez

Pedro Manuel Gómez-Portillo López

Introducción



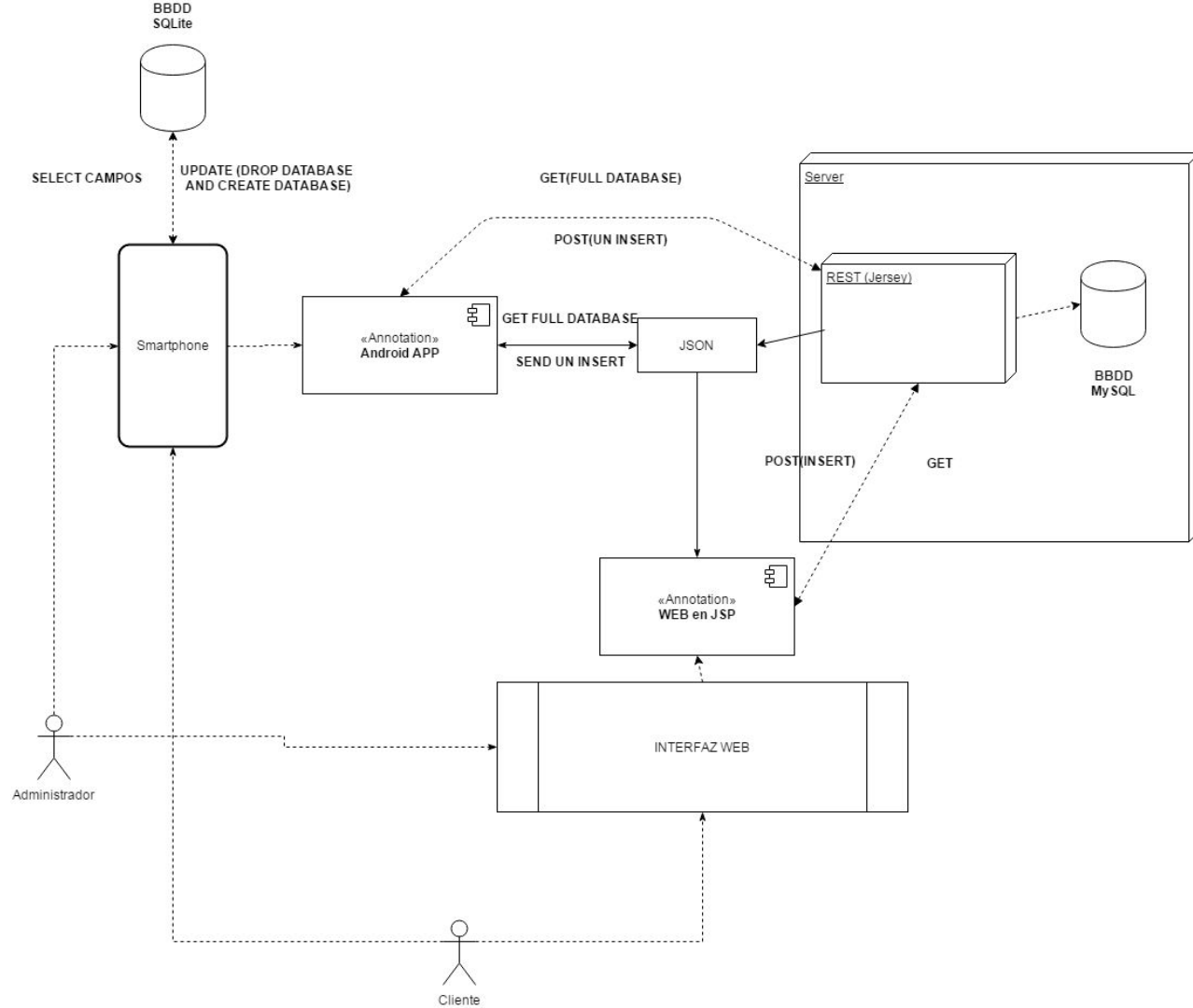
División de trabajo

- Aplicación Android - Juan Carlos Serrano
- Servidor y páginas web - Pedro Manuel Gómez-Portillo



Roles

- El usuario de la **página web** es un **administrador** que tiene permisos para realizar todas las acciones. Puede crear y borrar farmacias, usuarios, productos y realizar pedidos en nombre de otro usuario.
- El usuario de la **aplicación móvil** es un **cliente**, que tras iniciar sesión puede ver las farmacias, realizar pedidos y ver su historial.



Aplicación web

Servidor



- **Python + Flask** por ser el equivalente a **Jersey**
- Se ha evitado delegar funcionalidad de alto nivel en librerías
 - Serialización de clases en XML y JSON
 - Parseo de URIs
 - Gestión de la base de datos
- **Repositorio** - <https://github.com/gomezportillo/dss-pharmacy>

Justificación del lenguaje de programación



- Muy parecido a práctica 3 → Aprender nuevas tecnologías
- Configurar IDE vs programar a bajo nivel
- Preferencia personal por **Python**

PaaS² utilizados I



- Buena parte del desarrollo en Navidad
- No podíamos quedar en persona
- Necesidad de desplegar el servidor localmente



Servicios
PaaS

²Platform as a Service

PaaS² utilizados II

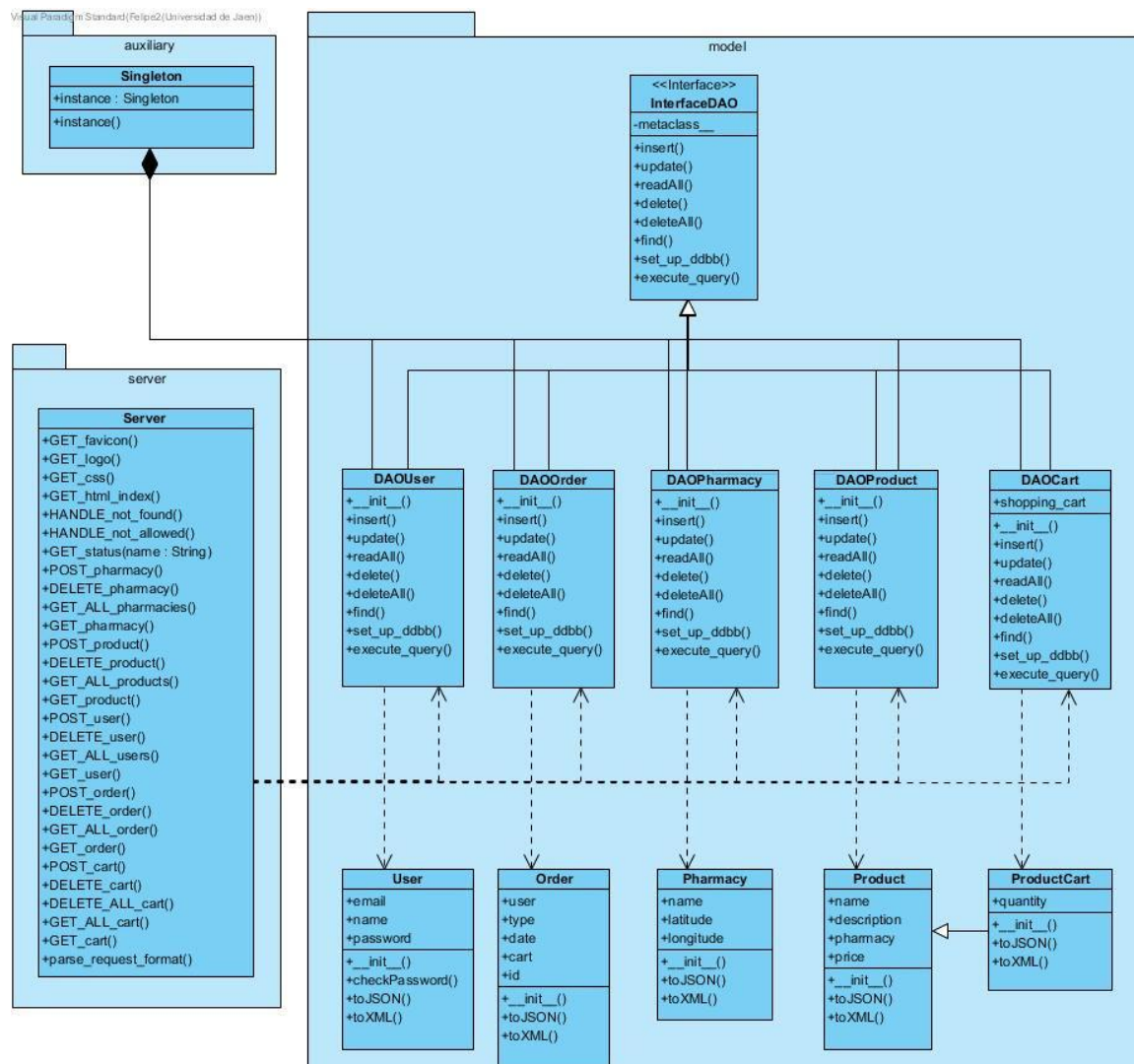
Heroku



ClearDB



Diagrama de clases



Rutas



- **\$BASE_URL/<servicio>** devuelve la página web del servicio especificado
 - *BASE_URL/pharmacies*
 - *BASE_URL/orders*
 - *BASE_URL/products*
 - *BASE_URL/cart*
 - *BASE_URL/users*
- **\$BASE_URL/rest/<servicio>** es la interfaz a la API REST del servidor
 - Acepta y responde peticiones HTTP usando **URIs**

Uso de URIs¹

URL/rest/users

```
[
  {
    "email": "admin",
    "name": "Adminstrator",
    "password": "admin"
  },
  {
    "email": "gomezportillo@dss.com",
    "name": "Pedro Manuel Gomez-Portillo",
    "password": "1234"
  }
]
```

URL/rest/users/admin

```
{
  "email": "admin",
  "name": "Adminstrator",
  "password": "admin"
}
```

¹Uniform Resource Identifier

Uso de URIs II



```
@app.route('/rest/pharmacies/<string:name>', methods=['GET'])
def GET_pharmacy(name):

    pharmacy = DAOPharmacy.instance().find( name )

    if pharmacy is not None:
        if format == 'xml':
            response = Response(pharmacy.toXML(),
                                status=201,
                                mimetype='text/xml')
```

XML y JSON

[URL/rest/products?format=xml](https://dss-pharmacy.herokuapp.com/rest/products?format=xml)

← → ↻ 🏠 🔒 <https://dss-pharmacy.herokuapp.com/rest/products?format=xml>

This XML file does not appear to have any style information associated with it. The document

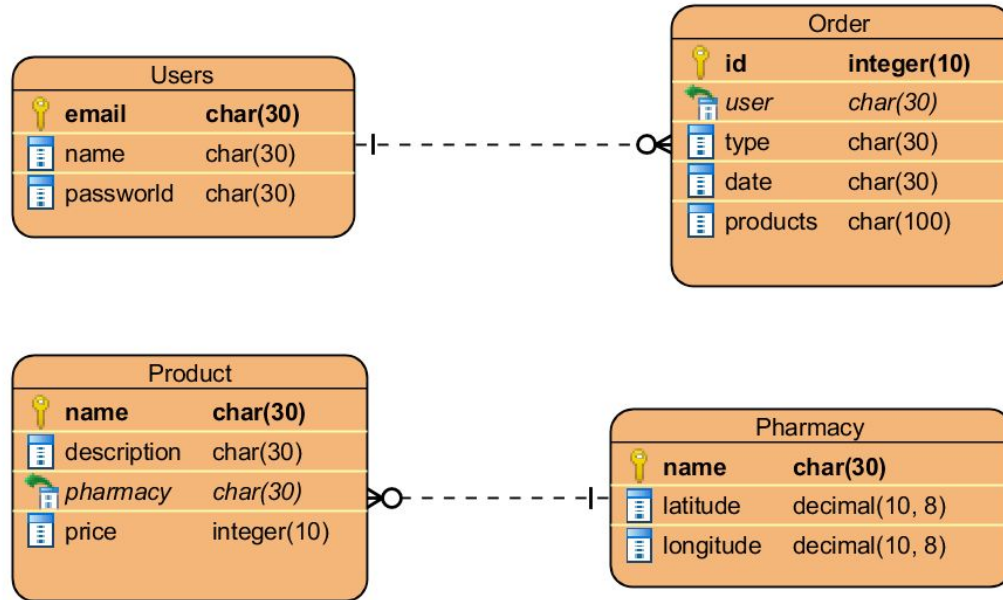
```
▼<products>
  ▼<product>
    <name>Bandage</name>
    <description>Cures wounds</description>
    <pharmacy>Pharmacy 1</pharmacy>
    <price>10</price>
  </product>
  ▼<product>
    <name>Frenadol</name>
    <description>Cures flu</description>
    <pharmacy>Pharmacy 2</pharmacy>
    <price>12</price>
  </product>
  ▼<product>
    <name>Ibuprofen</name>
    <description>Cures headache</description>
    <pharmacy>Pharmacy 1</pharmacy>
    <price>7</price>
  </product>
</products>
```

[URL/rest/products?format=json](https://dss-pharmacy.herokuapp.com/rest/products?format=json)

← → ↻ 🏠 🔒 <https://dss-pharmacy.herokuapp.com/rest/products?format=json>

```
[
  {
    "name": "Bandage",
    "description": "Cures wounds",
    "pharmacy": "Pharmacy 1",
    "price": 10
  },
  {
    "name": "Frenadol",
    "description": "Cures flu",
    "pharmacy": "Pharmacy 2",
    "price": 12
  },
  {
    "name": "Ibuprofen",
    "description": "Cures headache",
    "pharmacy": "Pharmacy 1",
    "price": 7
  }
]
```

Diagrama de Entidad-Relación de la BBDD




Páginas web

Bienvenida



Pharmacy manager

 Pharmacies

 Products

 Users

 Orders

 Cart

Welcome to the pharmacy management application.



Server version: 1.2

Server GitHub repository: <https://github.com/gomezportillo/dss-pharmacy>

Server developer: Pedro Manuel Gómez-Portillo

Android app developer: Juan Carlos Serrano

Farmacias

Pharmacy manager

[+ Pharmacies](#)

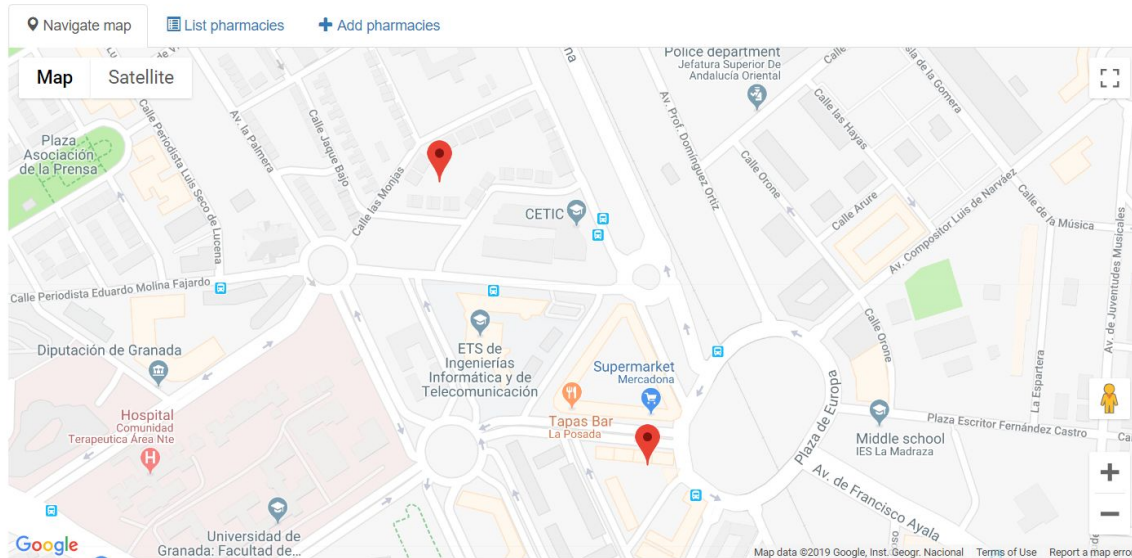
[Products](#)

[Users](#)

[Orders](#)


[Cart](#)


Pharmacies



Productos I

Pharmacy manager

 Pharmacies


 Products

 Users

 Orders

 Cart

Products

 See products

[+ Add products](#)

Name	Description	Price (€)	Pharmacy	Add to cart	Delete
Bandage	Cures wounds	10	Pharmacy 1		
Frenadol	Cures flu	12	Pharmacy 2		
Ibuprofen	Cures headache	7	Pharmacy 1		

Productos II



Pharmacy where is sold


Pharmacy 1

Pharmacy 1

Pharmacy 2

Usuarios

Pharmacy manager

 Pharmacies

 Products

 Users




 Orders

 Cart

Users


 See users

[+ Add user](#)

Email	Name	Password	Delete
admin	Administrator	admin	
gomezportillo@dss.com	Pedro Manuel Gomez-Portillo	1234	
juan.carlos.wow.95@gmail.com	Juan Carlos Serrano	secretpassworkd	


Carrito

Pharmacy manager

 Pharmacies

 Products

 Users

 Orders

 Cart

Cart

Name	Description	Price (€)	Pharmacy	Quantity
Frenadol	Cures flu	12	Pharmacy 2	1
Bandage	Cures wounds	10	Pharmacy 1	1

Total price: 22€

Email:

Enter a registered email for reserving or purchasing your cart



Empty cart



Reserve products



Buy products

Pedidos



Pharmacy manager

 Pharmacies

 Products

 Users

 Orders

 Cart

Orders

ID	Client	Type	Date	List of products
1	juan.carlos.wow@gmail.com	Reserve	1-1-2019 11:24:44	<ul style="list-style-type: none">Ibuprofen. Cures headache. Pharmacy 1. 7EUR x 2u
11	gomezportillo@dss.com	Purchase	2019-01-06 20:22:07	<ul style="list-style-type: none">Frenadol. Cures flu. Pharmacy 2. 12EUR x 1uBandage. Cures wounds. Pharmacy 1. 10EUR x 1u

Video demo de la parte web

Welcome to the pharmacy management application.



Server version: 1.3

Server Github repository: <https://github.com/gonzopomilio/ida-pharmacy>

Server developer: Pedro Manuel Gómez (Pomilio)

Android app developer: Juan Carlos Becerra

Aplicación Android

```

+ DetallesReservaActivity extends AppCompatActivity
{
    fields
    - textViewTit... :TextView
    - textViewM... :TextView
    - textViewDet... :TextView
    - btnVol... :Button
    - carr... :Carrito
    - fin... WIDTH :int
    - fin... HEIGHT :int
    - STR :String
    constructors
    methods
    # onCreate(savedInstanceState : Bundle) :void
    - selectBB... (fec... String) :void
    - crearQR() :void
    - encodeAsBit... (str :String) :Bitmap
}

```

```

+ HacerReservaActivity extends AppCompatActivity
implements GoogleApiClient.OnConnectionFailedList...
{
    fields
    - fin... WIDTH :int
    - fin... HEIGHT :int
    - STR :String
    - googleApCli... :GoogleApiClient
    - carr... :Carrito
    constructors
    methods
    # onCreate(savedInstanceState : Bundle) :void
    - encodeAsBit... (str :String) :Bitmap
    - onConnectionFa... (connectionRes... ConnectionRe... ) :void
    - realizarPO... () :void
    - handleSignInRe... (result :GoogleSignInRe... ) :void
    - insertarCarritoBDD() :void
}

```

```

+ InfoFarmaciaCustom
implements GoogleMap.InfoWindowAda...
{
    fields
    - fin... mWind... :View
    - mCont... :Context
    constructors
    + InfoFarmaciaCust... (conte... Context)
    methods
    - redrawWindowT... (marker :Marker, view :View) :void
    + getInfoWind... (marker :Marker) :View
    + getInfoConte... (marker :Marker) :View
}

```

```

+ ListaProductosCarritoActivity extends AppCompatActivity
{
    fields
    - carr... :Carrito
    - listView
    - btnVol... :Button
    - btnComp... :Button
    constructors
    methods
    # onCreate(savedInstanceState : Bundle) :void
    + dialogEliminarProd... (posic... int) :void
    - crearLI... () :void
}

```

```

+ HistorialActivity extends AppCompatActivity
{
    fields
    - btnVol... :Button
    - listView
    - carr... :Carrito
    constructors
    methods
    # onCreate(savedInstanceState : Bundle) :void
    - selectBB... () :void
}

```

```

+ Farmacia
{
    fields
    - name :String
    - latitud... :double
    - longitud... :double
    constructors
    methods
    + getNa... () :String
    + setName(na... String) :void
    + getLatit... () :double
    + setLatit... (latit... double) :void
    + getLongit... () :double
    + setLongit... (longitud... double) :void
}

```

```

+ MapsActivity extends FragmentActi...
implements OnMapReadyCallb...
{
    fields
    - mM... :GoogleMap
    - fin... FINE_LOCATI... :String
    - fin... COURSE_LOCATI... :String
    - fin... LOCATION_PERMISSION_REQUEST_C... :int
    - mLocationPermissionsGran... :Boolean
    - farmaciasNomb... :String
    - carr... :Carrito
    constructors
    methods
    # onCreate(savedInstanceState : Bundle) :void
    - getLocationPermiss... () :void
    - onMapRea... (googleMap... GoogleMap...) :void
}

```

```

+ Respuesta
{
    fields
    - status :String
    - status :String
    - message :String
    constructors
    methods
    + getStatus() :String
    + setStatus(status :String) :void
    + getMessage() :String
    + setMessage(message :String) :void
    + toStr... () :String
}

```

```

+ MainActivity extends AppCompatActivity
implements GoogleApiClient.OnConnectionFailedList...
{
    fields
    - googleApCli... :GoogleApiClient
    - signInButt... :SignInButton
    - carr... :Carrito
    constructors
    methods
    # onCreate(savedInstanceState : Bundle) :void
    - onConnectionFa... (connectionRes... ConnectionRe... ) :void
    - onActiviRes... (requestCode :int, resultCode :int, data :Intent) :void
}

```

```

+ GetPostService
{
    fields
    methods
    + getAPIPha... (Call<List<Farmacia>...
    + getAPIProd... (Call<List<Product...
    + crearPedi... (emal :String, type :String, date :String, products :String) :Call<Respu...
}

```

```

+ ListaProductosActivity extends Activity
implements View.OnClickListener
{
    fields
    - listView
    - btnShowCheckedit... :Button
    - mAdapter... MultiSelectionAdapter<Produ...
    - farm... :String
    - product... ArrayList<Product...
    - carr... :Carrito
    constructors
    methods
    # onCreate(savedInstanceState : Bundle) :void
    - cargarProdu... () :void
    - onCLi... (v :View) :void
}

```

```

+ MultiSelectionAdapter<T> extends BaseAdap...
{
    fields
    - mCont... :Context
    - mInfra... :LayoutInflater
    - mList :ArrayList<T>
    - mSparseBooleanAr... :SparseBooleanArr...
    - mCheckedChangeListe... :OnCheckedChangeListener
    constructors
    + MultiSelectionAda... (conte... Context, list :ArrayList<T>)
    methods
    + getCheckedIta... () :ArrayList<T>
    + getCo... () :int
    + getIt... (posic... int) :Obj...
    + getIt... (posic... int) :lo...
    + getVI... (posic... int, convertView : View, parent :ViewGro... ) :View
}

```

```

+ ListafarmaciasActivity extends AppCompatActivity
{
    fields
    - listView
    - carr... :Carrito
    - btnVol... :Button
    constructors
    methods
    # onCreate(savedInstanceState : Bundle) :void
}

```

```

+ AdminSQLite extends SQLiteOpenHelper...
{
    fields
    constructors
    + AdminSQ... (conte... Context, nomb... String, fecto... CursorFactory, version :int)
    methods
    + onCreate(db :SQLiteDatab... ) :void
    + onUpgra... (db :SQLiteDatab... , oldVersi... int, newVersion :int) :void
}

```

```

+ RetrofitClient
{
    fields
    - retrofit :Retrofit
    constructors
    methods
    + getCLi... (baseUrl :String) :Retrofit
}

```

```

+ ApiUtils
{
    fields
    - fin... BASE_URL :String
    constructors
    methods
    + getAPISev... () :GetPostServ...
}

```

```

+ Producto
implements Serializable
{
    fields
    - descript... :String
    - name :String
    - pharma... :String
    - pri... :String
    - unid... :int
    constructors
    methods
    + getUnid... () :int
    + setUnid... (unid... int) :void
    + incrementarUn... () :void
    + getDescript... () :String
    + setDescript... (descript... String) :void
    + getNa... () :String
    + setName(na... String) :void
    + getPharm... () :String
    + setPharma... (pharma... String) :void
    + getPri... () :String
    + setPrie... (pri... String) :void
    + toStr... () :String
}

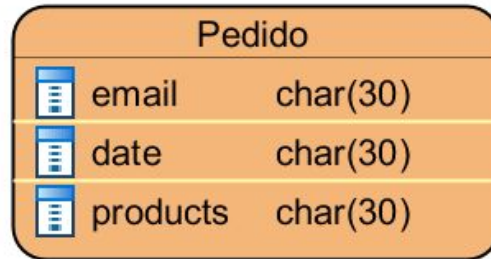
```

```

+ Carrito
implements Serializable
{
    fields
    - em... :String
    - product... List<Product...
    - fin... tipo :String
    - pr :String
    constructors
    + Carrito()
    methods
    + addProdu... (pr :Produ... ) :void
    + getEm... () :String
    + setEm... (em... String) :void
    + getProdu... () :List<Product...
    + eliminarProd... (n :int) :void
    + yaEsta(pr :Produ... ) :boolean
    + incrementarUn... (pr :Produ... ) :void
    + generarJSON() :String
    + hacerPO... (fec... String, pro :String) :void
    + toStr... () :String
}

```

Diagrama de Entidad-Relación de la BBDD



Conexión REST



- Uso de la librería *Retrofit* de Android
- Otras opciones probadas
 - *HttpURLConnection*
 - *Volley*

Auth0¹



Auth0

¹<https://auth0.com/>

Código QR



Uso de recursos



	CPU	Memoria	Red	Energía
MainActivity	1%	40 MB	0.3 KB/s	Light
MapsActivity	30%	70 MB	5 KB/s	Medium
ListaFarmaciasActivity	2%	50 MB	0 KB/s	Ligh
ListaProductosActivity	3%	61 MB	0.4 KB/s	Ligh
ListaProductosCarritoActivity	3%	55 MB	0 KB/s	Ligh
HacerReservaActivity	5%	72 MB	0.6 KB/s	Ligh
HistorialActivity	3%	55 MB	0 KB/s	Ligh
DetallesReservaActivity	5%	69 MB	0 KB/s	Medium

Video demo de la aplicación Android



Mayores problemas encontrados

Servidor + web

- Refactorizar el servidor para evitar usar funciones de alto nivel
- ~~ORM JPA~~ → Poblar las tablas HTML con los datos recibidos del servidor manualmente con JavaScript

Android

- Aprender Android
- Conexión REST con el servidor

Gracias por su atención
¿Preguntas?