Proyecto Final
Open Bank

Android

Openbank

Objetivo

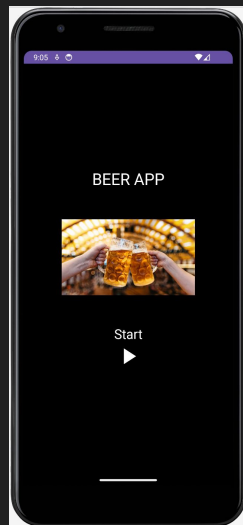- Implementar en una APP de Android los conocimientos adquiridos en la formación

Especificaciones Técnicas

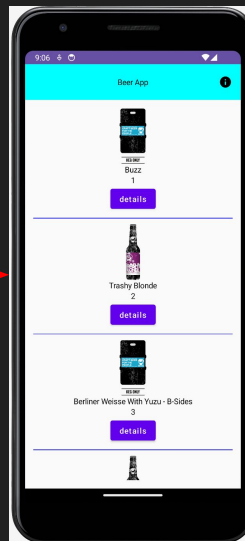- Android Studio
- Kotlin
- Retrofit
- Koin

Puntos Clave

- Nav builder
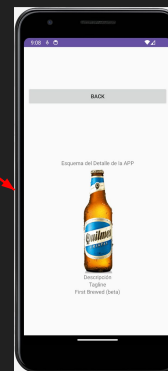- API
- Vistas y funcionalidad APP

1. Ingreso a la APP

2. Ingreso al Detalle

3. Ingreso a info

BEER APP

Start

Beer App

Buzz
1
details

Trashy Blonde
2
details

Berliner Weisse With Yuzu - B-Sides
3
details

back

CRAFT BEER FOR THE PEOPLE

KEG ONLY

A light, crisp and bitter IPA brewed with English and American hops. A small batch brewed only once.

BACK

Esquema del Detalle de la AFP

Descripción
Tagline
First Brewed (beta)

Sobre la Api…

Example Response

All beer endpoints return a json array with a number of beer objects inside.

```
[
  {
    "id": 192,
    "name": "Punk IPA 2007 - 2010",
    "tagline": "Post Modern Classic. Spiky. Tropical. Hoppy.",
    "first_brewed": "04/2007",
    "description": "Our flagship beer that kick started the
craft beer revolution. This is James and Martin's original
take on an American IPA, subverted with punchy New Zealand
hops. Layered with new world hops to create an all-out riot of
grapefruit, pineapple and lychee before a spiky, mouth-
puckering bitter finish.",
    "image_url": "https://images.punkapi.com/v2/192.png",
    "abv": 6.0,
    "ibu": 60.0,
    "target_fg": 1010.0,
    "target_og": 1056.0,
    "ebc": 17.0,
    "srm": 8.5,
    "ph": 4.4,
    "attenuation_level": 82.14,
    "volume": {
      "value": 20,
      "unit": "liters"
    },
    "boil_volume": {
      "value": 25,
      "unit": "liters"
    },
```

```kotlin
n158384
data class BeerDto (
    @Json(name="id") val id: String?,
    @Json(name="name") val name: String?,
    @Json(name="tagline") val tagline: String?,
    @Json(name="first_brewed") val firstBrewed: String?,
    @Json(name="description") val description: String?,
    @SerializedName("image_url") val imageUrl: String?,
)
```

the

from

Root Endpoint

The root endpoint should prefix all resources and is only accessible over HTTPS. CORS is also enabled.

```
https://api.punkapi.com/v2/
```

Get Beers

Gets beers from the api, you can apply several filters using url paramaters, the available options are listed below.

```
$ curl https://api.punkapi.com/v2/beers
```

Get a Single Beer

Gets a beer from the api using the beers id.

```
$ curl https://api.punkapi.com/v2/beers/1
```

```kotlin
± n158384
interface BeerApi {

    ± n158384
    @GET("beers")
    suspend fun getBeerList(): List<BeerDto>

    ± n158384
    @GET("beers/{id}")
    suspend fun getBeerById(@Path("id") id: String): List<BeerDto>


    ± n158384
    companion object{
        const val BASE_URL = "https://api.punkapi.com/v2/"
    }
}
```

```kotlin
± n158384
class RemoteDataSourceImpl (
    private val beerApi: BeerApi
    ) : RemoteDataSourceInterface {
    ± n158384
    override suspend fun getBeerList(): List<BeerDto> =
        beerApi.getBeerList()

    ± n158384
    override suspend fun getBeerById(id: String): BeerDto =
        beerApi.getBeerById(id)[0]

}
```

Testing

- Android Test

- Unit Test

      1- Models

      2- Mappers

      3- ViewModel

      4- Repository

      5- UseCases

Problemas

-   retrofit Builder (gson y luego moshi, no al revés)

Mejoras

- Nav Bar

- 3 Stars

- Pagination

- Caching

Pagination

Requests that return multiple items will be limited to 25 results by default. You can access other pages using the `?page` paramater, you can also increase the amount of beers returned in each request by changing the `?per_page` paramater.

```
$ curl https://api.punkapi.com/v2/beers?page=2&per_page=80
```

Gracias