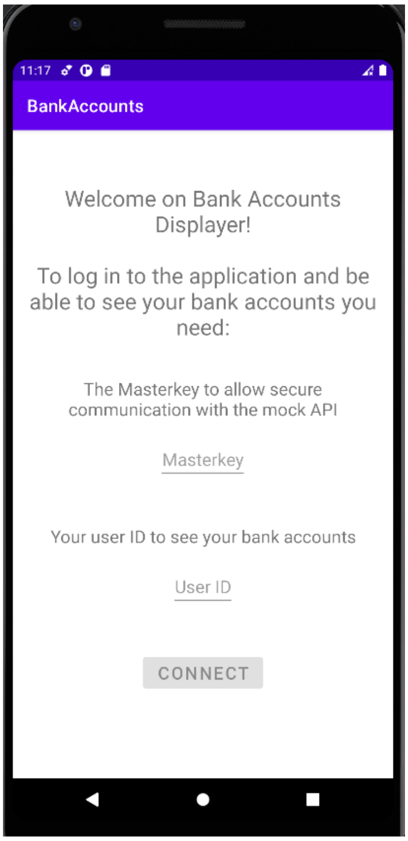
User experience



Welcome on Bank Accounts Displayer!

To log in to the application and be able to see your bank accounts you need:

The Masterkey to allow secure communication with the mock API

Masterkey

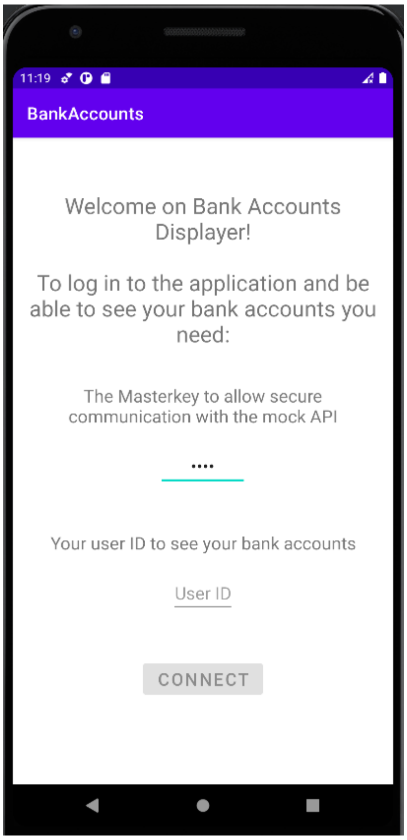
Your user ID to see your bank accounts

User ID

When we launch the application, we start by disconnecting the device from any network (WIFI and LTE).

The first of the two layouts of the application presents like this :

- Some text to introduce the application name and steps to access the main part of the application, i.e. the second layout where users can see their accounts.
- A Masterkey text field. This Masterkey is what helps us determine if the user is who he pretends to be and thus securing the application and the account information from any non client user.
- A User ID text field. This user ID replaces the classic authentication process. At the moment a user has the Masterkey, he can then access every other user's profile. But this isn't relevant for us since all users have the same accounts.
- A Connect button which is enabled only if both text fields above are not empty.



Welcome on Bank Accounts Displayer!

To log in to the application and be able to see your bank accounts you need:

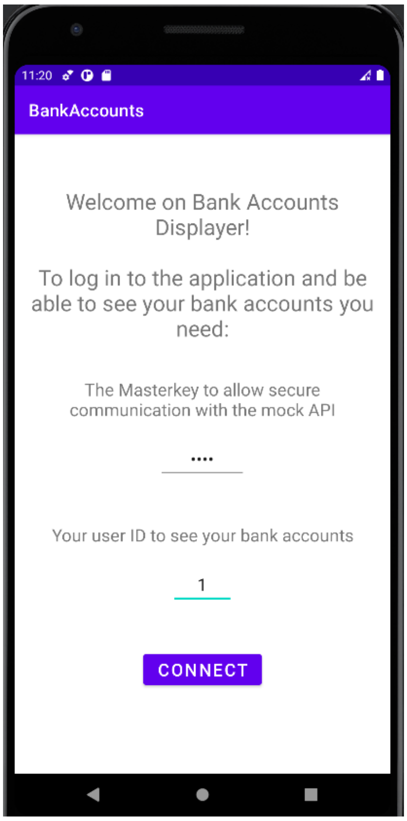
The Masterkey to allow secure communication with the mock API

••••

Your user ID to see your bank accounts

User ID

We can see that when we write inside the Masterkey field that characters are hidden like classic password fields and the connect button isn't enabled yet.



BankAccounts

Welcome on Bank Accounts Displayer!

To log in to the application and be able to see your bank accounts you need:

The Masterkey to allow secure communication with the mock API

••••

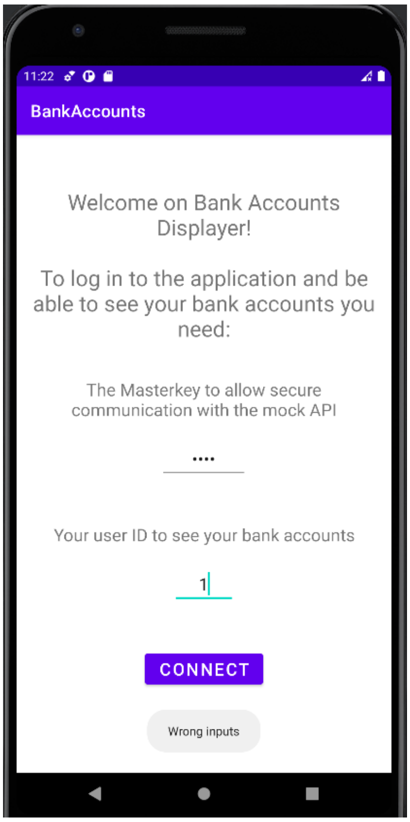
Your user ID to see your bank accounts

1

But when we complete both Masterkey field and User ID field, the connect button is enabled.

However, this is not enough to access the second layout: Masterkey and User ID have to be correct inputs.

For the ID, it is not complicated since it just has to be an integer between 1 and 73. Plus, the inputField is set to numerical so we can't write something else than numbers inside this field.



BankAccounts

Welcome on Bank Accounts Displayer!

To log in to the application and be able to see your bank accounts you need:

The Masterkey to allow secure communication with the mock API

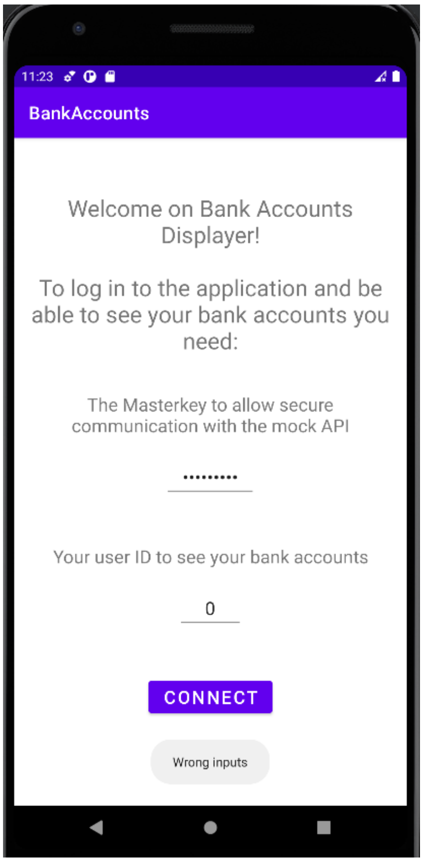
••••

Your user ID to see your bank accounts

1

Wrong inputs

And when we click on connect, we can see that inputs were incorrect (indicating by the toast). The Masterkey was wrong but the ID was correct.


11:23

BankAccounts

Welcome on Bank Accounts Displayer!

To log in to the application and be able to see your bank accounts you need:

The Masterkey to allow secure communication with the mock API

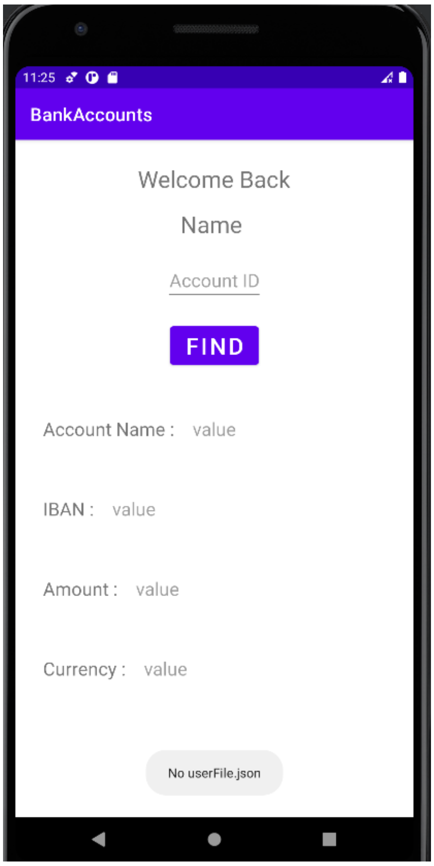
·········

Your user ID to see your bank accounts

0

CONNECT

Wrong inputs


11:25

BankAccounts

Welcome Back

Name

Account ID

FIND

Account Name : value

IBAN : value

Amount : value

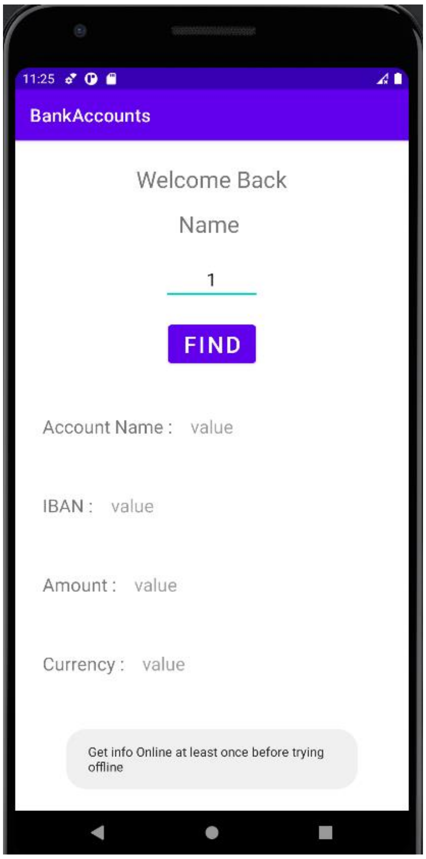
Currency : value

No userFile.json

Now, the Masterkey was correct but the ID wasn't and same result as before (error toast)

Afterwards, I put the same Masterkey as before with a valid user ID and got connected to the application. Normally, I would have the name of the user displayed instead of "name". But since the client is not connected to any network and it is the first time he connects to the application, no info is stored on his device, hence the error toasted: no userFile.json.

Even if not connected to internet and not having access to any information, I give him access to the second layout to show him that he got the right Masterkey since he is a registered confirmed user.



Welcome Back

Name

1

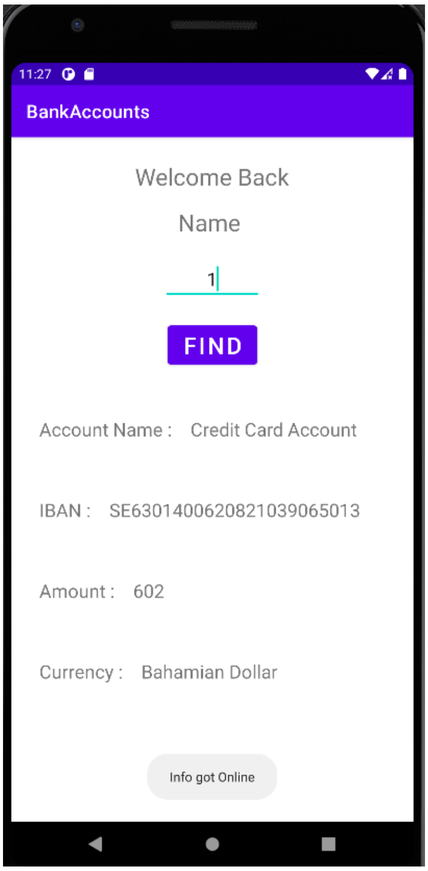
Account Name : value

IBAN : value

Amount : value

Currency : value

Get info Online at least once before trying offline



Welcome Back

Name

1

Account Name : Credit Card Account

IBAN : SE6301400620821039065013

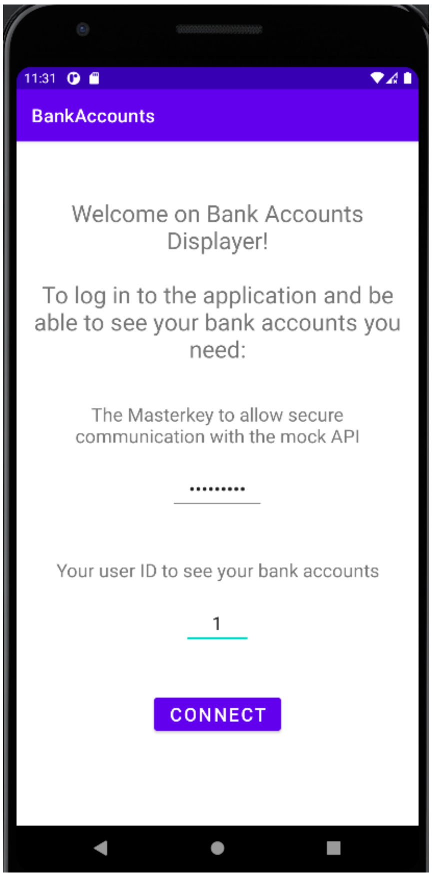
Amount : 602

Currency : Bahamian Dollar

Info got Online

So if we couldn't get the user information, then the user can't get the account information either. And we toast this error : "Get info online at least once before typing offline"

But if we connect the device to the network and click find for account id 1, we get info of this account online (as said by the toast).



Welcome on Bank Accounts Displayer!

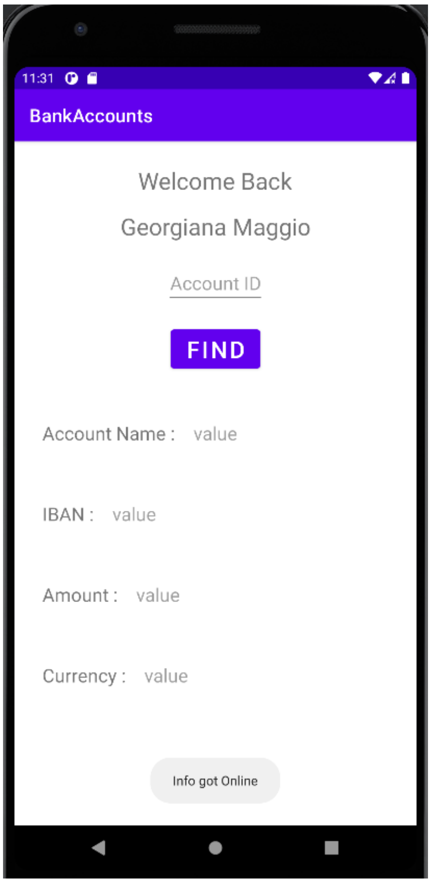
To log in to the application and be able to see your bank accounts you need:

The Masterkey to allow secure communication with the mock API

•••••••••

Your user ID to see your bank accounts

1



Welcome Back

Georgiana Maggio

Account ID

Account Name : value

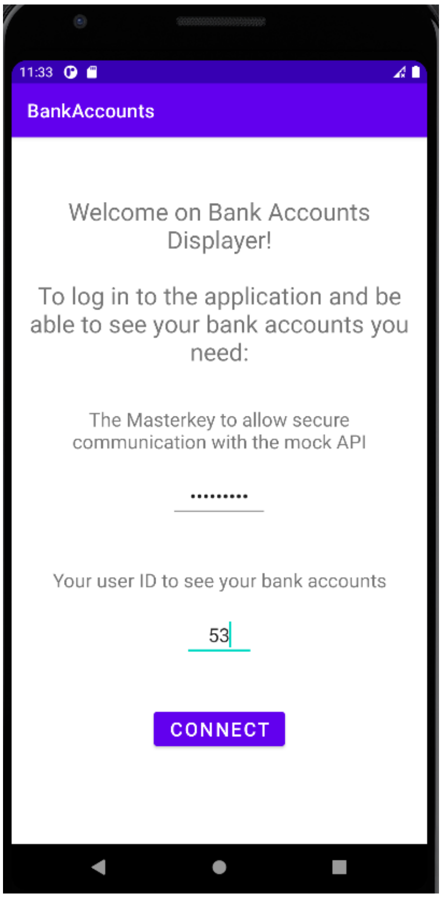
IBAN : value

Amount : value

Currency : value

Info got Online

So now if we go on the previous layout of the application and stay connected to the network, we are now going to get user information online and have first and last name of the user corresponding to the User ID.



Welcome on Bank Accounts Displayer!

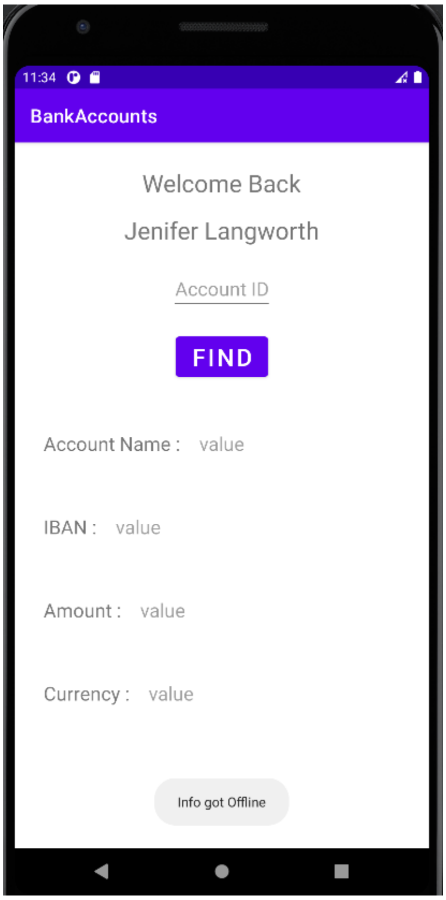
To log in to the application and be able to see your bank accounts you need:

The Masterkey to allow secure communication with the mock API

•••••••••

Your user ID to see your bank accounts

53



Welcome Back

Jenifer Langworth

Account ID

Account Name : value

IBAN : value

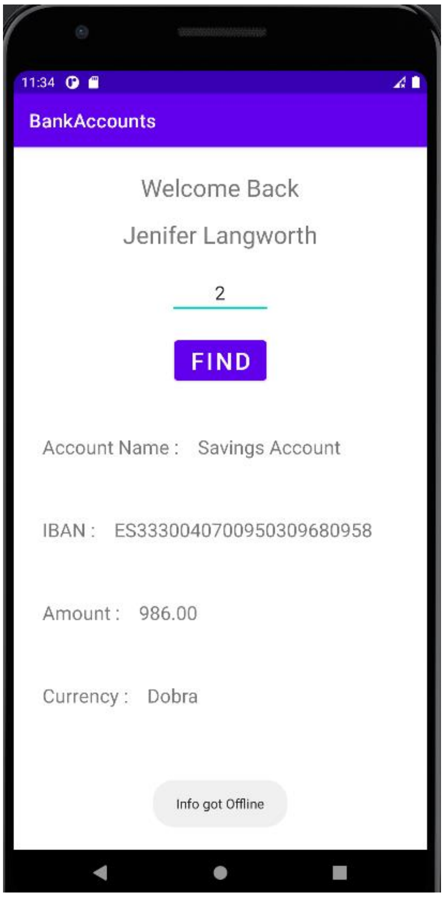
Amount : value

Currency : value

Info got Offline

If we get back on the first layout, disconnect the device from internet and type a user ID different from the previous one, we could expect to not have any information.

But when we got info online for the first time, we registered the info of all users. So when registering offline after that, we can get back any user's info.



Welcome Back

Jenifer Langworth

2

Account Name : Savings Account

IBAN : ES3330040700950309680958

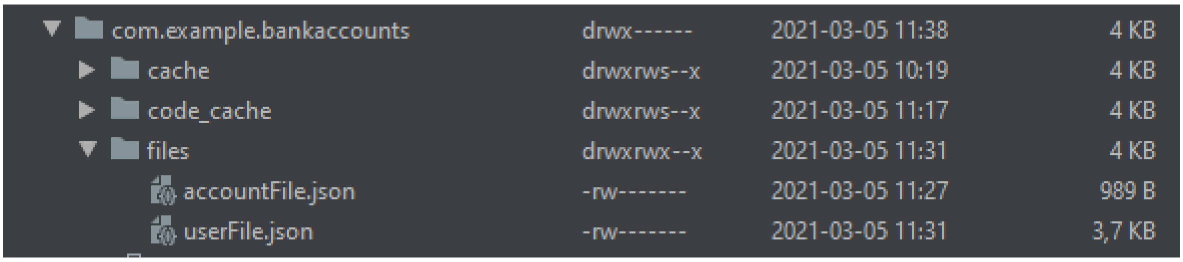
Amount : 986.00

Currency : Dobra

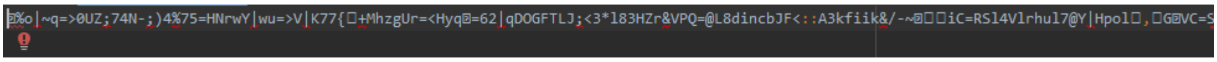
Info got Offline

Same thing for the account information : since we already got info online once for one account, we registered info of all accounts inside the device this one time.

Therefore, it is possible to get info of account offline even if user didn't get its information online directly.



```
▼ com.example.bankaccounts     drwx------    2021-03-05 11:38    4 KB
  ▶ cache                      drwxrws--x    2021-03-05 10:19    4 KB
  ▶ code_cache                 drwxrws--x    2021-03-05 11:17    4 KB
  ▼ files                      drwxrwx--x    2021-03-05 11:31    4 KB
      accountFile.json         -rw-------    2021-03-05 11:27   989 B
      userFile.json            -rw-------    2021-03-05 11:31  3,7 KB
```



```
▯%o|~q=>0UZ;74N-;)4%75=HNrwY|wu=>V|K77{□+MhzgUr=<Hyq▯=62|qDOGFTLJ;<3*l83HZr&VPQ=@L8dincbJF<::A3kfiik&/-~▯□□iC=RSl4Vlrhul7@Y|Hpol□,□G▯VC=S
```

As for the files containing those information on users and accounts, there are stored on the device of the client and encrypted with a special function.