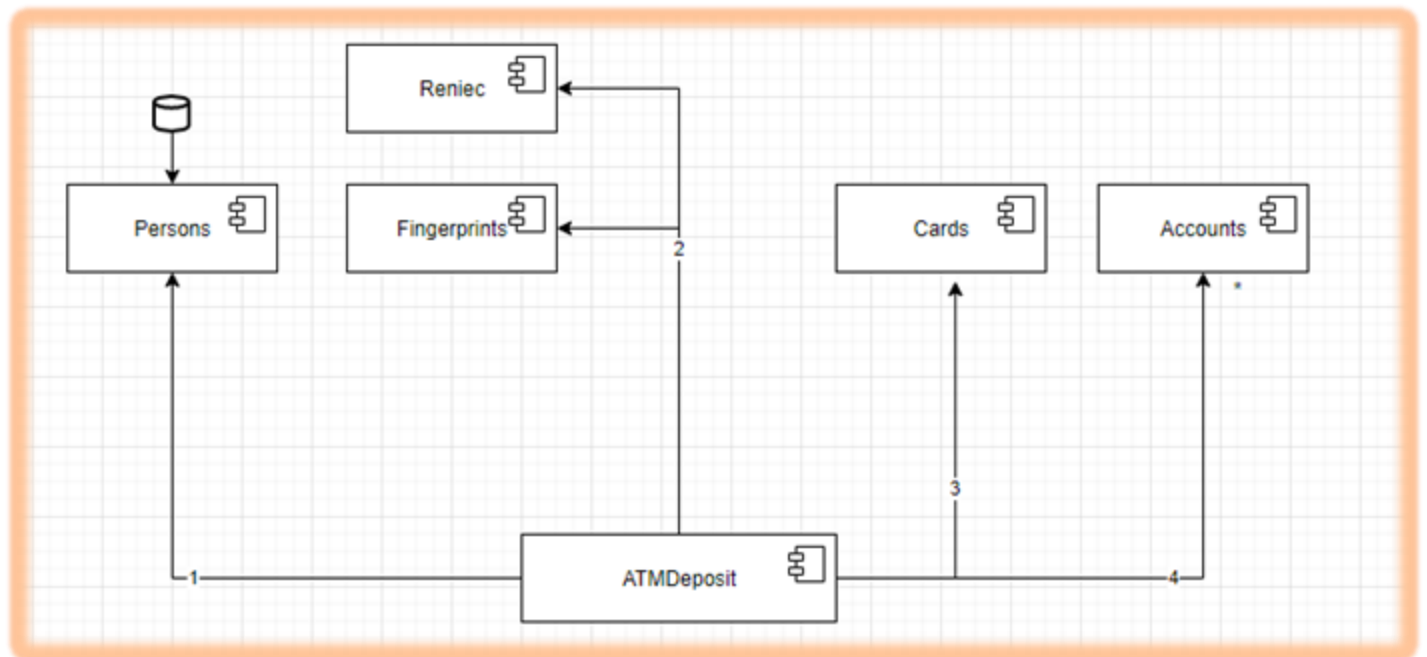


## ATMDeposit (Single<AtmDepositResponse>)

Api orquestadora que recibe DNI y una cantidad



Método	Uri
POST	/atm/deposits/
Request	<pre>{   "documentNumber": "10000000"   "amount": 100 }</pre>
Response	<pre>{   "fingerprintEntityName": "Core", //Reniec o Core. Del "entityName"   "validAccounts": [{     "accountNumber": "1111222233334441XXX"   }, {     "accountNumber": "1111222233334442XXX"   }, {     "accountNumber": "1111222233334443XXX"   }   ],   "customerAmount": 3100 //suma de montos de cada cuenta + la cantidad depositada }</pre>

## Persons (Single<PersonResponse>)

Api que obtiene los datos de un cliente por DNI

Método	Uri
GET	/core/persons?documentNumber="10000000"
Response	{ "id": "1", "document": "10000000", "fingerprint": true, "blacklist": true }

### BD Embebida.

Columna	Descripcion	Ejemplo1	Ejemplo2	Ejemplo3
Id	Nro identificador autogen	1	2	3
Document	Nro de documento dni	10000000	10000001	10000002
Fingerprint	Flag indica si el core tiene su huella registrada. <b>True = Api de Fingerprints. False = Api de Reniec</b>	True	False	True
Blacklist	Flag indica si esta en lista negra. Tirar Excepcion = true	False	False	True

## Fingerprints (Single<FingerprintResponse>)

Api que evalua en base a un DNI, **siempre devolverá success TRUE.**

Método	Uri
POST	/core/fingerprints/validate
Request	{ "document": "10000000" }
Response	{ "entityName": "Core", "success": true }

## Reniec (Single<ReniecResponse>)

Api que evalua en base a un DNI, **siempre devolverá success TRUE**.

Método	Uri
POST	/external/reniec/validate
Request	{ "document": "I" }
Response	{ "entityName": "Reniec", "success": true }

## Cards (Single<CardResponse>)

Api que obtiene las tarjetas de un cliente. **Siempre responde lo mismo**

Método	Uri
GET	/core/cards?documentNumber="10000000"
Response	{ "cards": [{ "cardNumber": "111122223333444 <b>1</b> ", "active": true },{ "cardNumber": "111122223333444 <b>2</b> ", "active": true },{ "cardNumber": "111122223333444 <b>3</b> ", "active": true },{ "cardNumber": "111122223333444 <b>4</b> ", "active": false },{ "cardNumber": "111122223333444 <b>5</b> ", "active": false },{ "cardNumber": "111122223333444 <b>6</b> ", "active": false } ] } Solo operar en base a las activas,

## Accounts (Single<AccountResponse>)

Api que obtiene la cuenta de una tarjeta. La cuenta devuelta será una concatenación de la tarjeta + "XXX"

**Obligatorio: Colocar Thread.sleep() de 5 segundos.**

Método	Uri
GET	/core/accounts?cardNumber="111122223333444?"
Response	{ "accountNumber": "1111222233334441XXX", "amount": 1000 }
Response	{ "accountNumber": "1111222233334442XXX", "amount": 500 }
Response	{ "accountNumber": "1111222233334443XXX", "amount": 1500 }

//pruebas unitarias, cobertura al 100% del api ATM

//H2

Serializable

<http://www.sc.ehu.es/sbweb/fisica/cursolJava/fundamentos/archivos/objetos.htm>

<https://stackoverflow.com/questions/43468637/feign-client-spring-boot-application-and-rx-observable-class-not-found-error>