1. **Admin Dashboard:**

Contains:

1. num of subscription and number of active users for (fona, faxa, and porta services)
2. total number of customers, and active customers.
3. Num of Issue requests and cheque book requests.
4. Total number of users and them details (name, id, role, branch code, and status) in a table.

So, the initialization service for admin dashboard will return an object like this:

{

“fona”: {

“subscriptions” : number,

“activeUsers” : number

},

“faxa”: {

“subscriptions” : number,

“activeUsers” : number

},

“porta”: {

“subscriptions” : number,

“activeUsers” : number

},

“customers”: {

“total” : number,

“active” : number

},

“ChequeBookRequest”: number,

“issueRequest”: number,

“users”:

{ “totalCount”:number,

“users”: [

{  
 “username” : ”str”,

“userID”: “number”,

“role” : “str”,

“branchCode” : “str”,

“status” : “str”

},

{..}

]

}

Users must be paginated and cached (by logic)

And all of the counter fields you can retrieve it from the DB by a count query.

Where I’ll find this information? Guess what! in DB

1. For num of subscriptions and number of active users:

SELECT

COUNT(\*) AS total\_count,

SUM(activeIND) AS active\_count

FROM subscription

WHERE serviceId = 'serviceID';

1. For total number of customers and active customers:

SELECT

COUNT(\*) AS total\_count,

SUM(activeIND) AS active\_count

FROM client

Note: In the old MW.client table there is no activeIND!

1. For Requests: as I understand: in the old DB, there are 2 tables: request , and useCase for each request there is a useCase id, I think the usecase means the request type, issue ? cheque ? and so on.

So if I’m right, the query will be:

SELECT

COUNT(\*) AS total\_count,

FROM request

WHERE usecaseID = ‘usecaseID’

1. Users data:

SELECT \* from USER

Note: we can replace the role id column by a role Enum, and store its int value in DB, when retrieve it the JVM will map this num to its appropriate value.

In the programming side: handle the pagination and caching using spring data JPA.