**Is 302, Moso As,**

**Task : Service Design**

SEGMENT 1

TASK MAP

Moso is using AWS. Create a lambda code which

* ~~Checks database for user who didn’t login for 1 month~~
* Warn them with their registered emails that if they do not login within a week their user profiles would be deleted.
* ~~And do so, delete user and save deleted user profile logs.~~
* ~~User should be informed on deletion~~

TASK SUBS

1. Create WarnedUser with warningDate
   1. ~~Id:PK : auto-increment~~
   2. ~~UUID : VarChar~~
   3. ~~Warning\_Date : DateTime~~
   4. ~~isWarned: Boolean~~
   5. ~~isDeleted: Boolean~~
   6. ~~Deletion\_Date : DateTime~~
   7. ~~Last\_login~~
   8. ~~Email~~
2. Select expiring users
   1. ~~Needed Columns from master table:~~
      1. ~~Uuid : VarChar~~
      2. ~~Email : VarChar~~
      3. ~~Last\_login : DateTime~~
   2. ~~Meet condition : Last\_login <= DATE\_SUB(NOW(), INTERVAL 11 Months)~~
3. ~~Get data from master table~~
   1. ~~Response = list of user objects~~

4-For each expiry in expiry\_user\_objects;

Check if

~~expiry exists in the warned\_users table in the moso database,~~

~~And isMailed true a week ago,~~

then,

~~Delete user from the users table in the moso database~~,

Delete user from aws servers,

And,

~~Set isDeleted true in the warned\_users table,~~

#isMailed:false|empty, then,

Send an email to the user,

use the registered mail in the warned\_users table,

and say that “your moso account is passive for a time please take action an log into your account in a week. Otherwise your account will be deleted permanently”,

And,

~~set isMailed to true,~~

And,

~~register the mail sending date to the mail\_date column I in the warned\_users table in the moso database,~~

On the other hand, expiry is mailed but it was less then a week ago,

do nothing, just skip,

If expiry does not exist in the warned\_users table,

~~Save non existence element to warned\_users table as a new row~~

And, Then do as in ‘If #isMailed:false|empty’.

SEGMENT 2

10/10/2023

BEST PRACTICES

First, Go manual and create an environment (framework, dependancy, mysql/database)

Source is separate,

This business logic is uavhengig,

TASK MAP

manual.. 2 rds on aws

Database=mysql

Create a lambda of local

Load dependancy packs..

Check db connections..

Test lambda done, then migrate local code,

Run the code on..serverless--- lambda

Do as sam/cdk version..

TASK SUB 10/10

~~Source rds.. on aws:~~

~~Migrate local data~~

~~Used rds service on aws to create db,~~

~~Mysqlworkbench~~

~~/server/export databse (with structure and data)~~

~~/new connection(aws endpoint, user, pass)~~

~~/creare new schema (for coming aws db)~~

~~/server/import database~~

passive\_users db on aws:

Manual (comply certain dayatypes), or

Pycharm (redefine host and other connectioin rules)

db on aws;

migrate or fill after filtering.