package com.twitter.follow\_recommendations.common.clients.email\_storage\_service

import com.twitter.cds.contact\_consent\_state.thriftscala.PurposeOfProcessing

import com.twitter.emailstorage.api.thriftscala.EmailStorageService

import com.twitter.emailstorage.api.thriftscala.GetUsersEmailsRequest

import com.twitter.stitch.Stitch

import javax.inject.Inject

import javax.inject.Singleton

@Singleton

class EmailStorageServiceClient @Inject() (

val emailStorageService: EmailStorageService.MethodPerEndpoint) {

def getVerifiedEmail(

userId: Long,

purposeOfProcessing: PurposeOfProcessing

): Stitch[Option[String]] = {

val req = GetUsersEmailsRequest(

userIds = Seq(userId),

clientIdentifier = Some("follow-recommendations-service"),

purposesOfProcessing = Some(Seq(purposeOfProcessing))

)

Stitch.callFuture(emailStorageService.getUsersEmails(req)) map {

\_.usersEmails.map(\_.confirmedEmail.map(\_.email)).head

}

}

}