package com.twitter.follow\_recommendations.products.home\_timeline

import com.twitter.follow\_recommendations.assembler.models.ActionConfig

import com.twitter.follow\_recommendations.assembler.models.FollowedByUsersProof

import com.twitter.follow\_recommendations.assembler.models.FooterConfig

import com.twitter.follow\_recommendations.assembler.models.GeoContextProof

import com.twitter.follow\_recommendations.assembler.models.HeaderConfig

import com.twitter.follow\_recommendations.assembler.models.Layout

import com.twitter.follow\_recommendations.assembler.models.TitleConfig

import com.twitter.follow\_recommendations.assembler.models.UserListLayout

import com.twitter.follow\_recommendations.assembler.models.UserListOptions

import com.twitter.follow\_recommendations.common.base.BaseRecommendationFlow

import com.twitter.follow\_recommendations.common.base.IdentityTransform

import com.twitter.follow\_recommendations.common.base.Transform

import com.twitter.follow\_recommendations.flows.ads.PromotedAccountsFlow

import com.twitter.follow\_recommendations.flows.ads.PromotedAccountsFlowRequest

import com.twitter.follow\_recommendations.blenders.PromotedAccountsBlender

import com.twitter.follow\_recommendations.common.models.DisplayLocation

import com.twitter.follow\_recommendations.common.models.Recommendation

import com.twitter.follow\_recommendations.flows.post\_nux\_ml.PostNuxMlFlow

import com.twitter.follow\_recommendations.flows.post\_nux\_ml.PostNuxMlRequestBuilder

import com.twitter.follow\_recommendations.products.common.Product

import com.twitter.follow\_recommendations.products.common.ProductRequest

import com.twitter.follow\_recommendations.products.home\_timeline.configapi.HomeTimelineParams.\_

import com.twitter.inject.Injector

import com.twitter.product\_mixer.core.model.marshalling.request

import com.twitter.product\_mixer.core.product.guice.ProductScope

import com.twitter.stitch.Stitch

import javax.inject.Inject

import javax.inject.Singleton

@Singleton

class HomeTimelineProduct @Inject() (

postNuxMlFlow: PostNuxMlFlow,

postNuxMlRequestBuilder: PostNuxMlRequestBuilder,

promotedAccountsFlow: PromotedAccountsFlow,

promotedAccountsBlender: PromotedAccountsBlender,

productScope: ProductScope,

injector: Injector,

) extends Product {

override val name: String = "Home Timeline"

override val identifier: String = "home-timeline"

override val displayLocation: DisplayLocation = DisplayLocation.HomeTimeline

override def selectWorkflows(

request: ProductRequest

): Stitch[Seq[BaseRecommendationFlow[ProductRequest, \_ <: Recommendation]]] = {

postNuxMlRequestBuilder.build(request).map { postNuxMlRequest =>

Seq(

postNuxMlFlow.mapKey({ request: ProductRequest => postNuxMlRequest }),

promotedAccountsFlow.mapKey(mkPromotedAccountsRequest))

}

}

override val blender: Transform[ProductRequest, Recommendation] = {

promotedAccountsBlender.mapTarget[ProductRequest](getMaxResults)

}

private val identityTransform = new IdentityTransform[ProductRequest, Recommendation]

override def resultsTransformer(

request: ProductRequest

): Stitch[Transform[ProductRequest, Recommendation]] = Stitch.value(identityTransform)

override def enabled(request: ProductRequest): Stitch[Boolean] =

Stitch.value(request.params(EnableProduct))

override def layout: Option[Layout] = {

productMixerProduct.map { product =>

val homeTimelineStrings = productScope.let(product) {

injector.instance[HomeTimelineStrings]

}

UserListLayout(

header = Some(HeaderConfig(TitleConfig(homeTimelineStrings.whoToFollowModuleTitle))),

userListOptions = UserListOptions(userBioEnabled = true, userBioTruncated = true, None),

socialProofs = Some(

Seq(

FollowedByUsersProof(

homeTimelineStrings.whoToFollowFollowedByManyUserSingleString,

homeTimelineStrings.whoToFollowFollowedByManyUserDoubleString,

homeTimelineStrings.whoToFollowFollowedByManyUserMultipleString

),

GeoContextProof(homeTimelineStrings.whoToFollowPopularInCountryKey)

)),

footer = Some(

FooterConfig(

Some(ActionConfig(homeTimelineStrings.whoToFollowModuleFooter, "http://twitter.com"))))

)

}

}

override def productMixerProduct: Option[request.Product] = Some(HTLProductMixer)

private[home\_timeline] def mkPromotedAccountsRequest(

req: ProductRequest

): PromotedAccountsFlowRequest = {

PromotedAccountsFlowRequest(

req.recommendationRequest.clientContext,

req.params,

req.recommendationRequest.displayLocation,

None,

req.recommendationRequest.excludedIds.getOrElse(Nil)

)

}

private[home\_timeline] def getMaxResults(req: ProductRequest): Int = {

req.recommendationRequest.maxResults.getOrElse(

req.params(DefaultMaxResults)

)

}

}