package com.twitter.product\_mixer.component\_library.pipeline.candidate.who\_to\_follow\_module

import com.twitter.account\_recommendations\_mixer.{thriftscala => t}

import com.twitter.product\_mixer.component\_library.candidate\_source.account\_recommendations\_mixer.AccountRecommendationsMixerCandidateSource

import com.twitter.product\_mixer.component\_library.model.candidate.UserCandidate

import com.twitter.product\_mixer.core.feature.Feature

import com.twitter.product\_mixer.core.functional\_component.candidate\_source.BaseCandidateSource

import com.twitter.product\_mixer.core.functional\_component.common.alert.Alert

import com.twitter.product\_mixer.core.functional\_component.decorator.CandidateDecorator

import com.twitter.product\_mixer.core.functional\_component.decorator.urt.builder.metadata.BaseFeedbackActionInfoBuilder

import com.twitter.product\_mixer.core.functional\_component.decorator.urt.builder.timeline\_module.BaseModuleDisplayTypeBuilder

import com.twitter.product\_mixer.core.functional\_component.filter.Filter

import com.twitter.product\_mixer.core.functional\_component.gate.BaseGate

import com.twitter.product\_mixer.core.functional\_component.transformer.CandidateFeatureTransformer

import com.twitter.product\_mixer.core.functional\_component.transformer.CandidatePipelineQueryTransformer

import com.twitter.product\_mixer.core.functional\_component.transformer.CandidatePipelineResultsTransformer

import com.twitter.product\_mixer.core.model.common.identifier.CandidatePipelineIdentifier

import com.twitter.product\_mixer.core.pipeline.PipelineQuery

import com.twitter.product\_mixer.core.pipeline.candidate.DependentCandidatePipelineConfig

import com.twitter.timelines.configapi.FSParam

import com.twitter.timelines.configapi.Param

import com.twitter.timelines.configapi.decider.DeciderParam

class WhoToFollowArmDependentCandidatePipelineConfig[Query <: PipelineQuery](

override val identifier: CandidatePipelineIdentifier,

override val enabledDeciderParam: Option[DeciderParam[Boolean]],

override val supportedClientParam: Option[FSParam[Boolean]],

override val alerts: Seq[Alert],

override val gates: Seq[BaseGate[Query]],

accountRecommendationsMixerCandidateSource: AccountRecommendationsMixerCandidateSource,

override val filters: Seq[Filter[Query, UserCandidate]],

moduleDisplayTypeBuilder: BaseModuleDisplayTypeBuilder[Query, UserCandidate],

feedbackActionInfoBuilder: Option[

BaseFeedbackActionInfoBuilder[PipelineQuery, UserCandidate]

],

displayLocationParam: Param[String],

excludedUserIdsFeature: Option[Feature[PipelineQuery, Seq[Long]]],

profileUserIdFeature: Option[Feature[PipelineQuery, Long]])

extends DependentCandidatePipelineConfig[

Query,

t.AccountRecommendationsMixerRequest,

t.RecommendedUser,

UserCandidate

] {

override val candidateSource: BaseCandidateSource[

t.AccountRecommendationsMixerRequest,

t.RecommendedUser

] =

accountRecommendationsMixerCandidateSource

override val queryTransformer: CandidatePipelineQueryTransformer[

PipelineQuery,

t.AccountRecommendationsMixerRequest

] = WhoToFollowArmCandidatePipelineQueryTransformer(

displayLocationParam = displayLocationParam,

excludedUserIdsFeature = excludedUserIdsFeature,

profileUserIdFeature = profileUserIdFeature

)

override val featuresFromCandidateSourceTransformers: Seq[

CandidateFeatureTransformer[t.RecommendedUser]

] = Seq(WhoToFollowArmResponseFeatureTransformer)

override val resultTransformer: CandidatePipelineResultsTransformer[

t.RecommendedUser,

UserCandidate

] = { user => UserCandidate(user.userId) }

override val decorator: Option[CandidateDecorator[Query, UserCandidate]] =

Some(

WhoToFollowArmCandidateDecorator(

moduleDisplayTypeBuilder,

feedbackActionInfoBuilder

))

}