package com.twitter.home\_mixer.product.for\_you

import com.twitter.home\_mixer.functional\_component.decorator.builder.HomeClientEventInfoBuilder

import com.twitter.home\_mixer.model.HomeFeatures.SuggestTypeFeature

import com.twitter.home\_mixer.product.for\_you.functional\_component.gate.PushToHomeRequestGate

import com.twitter.home\_mixer.product.for\_you.model.ForYouQuery

import com.twitter.product\_mixer.component\_library.decorator.urt.UrtItemCandidateDecorator

import com.twitter.product\_mixer.component\_library.decorator.urt.builder.item.tweet.TweetCandidateUrtItemBuilder

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

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

import com.twitter.product\_mixer.core.feature.featuremap.FeatureMap

import com.twitter.product\_mixer.core.feature.featuremap.FeatureMapBuilder

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

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

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

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

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.model.common.identifier.CandidateSourceIdentifier

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

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

import com.twitter.timelineservice.suggests.{thriftscala => st}

import javax.inject.Inject

import javax.inject.Singleton

@Singleton

class ForYouPushToHomeTweetCandidatePipelineConfig @Inject() ()

extends CandidatePipelineConfig[

ForYouQuery,

ForYouQuery,

TweetCandidate,

TweetCandidate

] {

override val identifier: CandidatePipelineIdentifier =

CandidatePipelineIdentifier("ForYouPushToHomeTweet")

override val gates: Seq[Gate[ForYouQuery]] = Seq(PushToHomeRequestGate)

override val queryTransformer: CandidatePipelineQueryTransformer[

ForYouQuery,

ForYouQuery

] = identity

override val featuresFromCandidateSourceTransformers: Seq[

CandidateFeatureTransformer[TweetCandidate]

] = Seq(new CandidateFeatureTransformer[TweetCandidate] {

override def features: Set[Feature[\_, \_]] = Set(SuggestTypeFeature)

override val identifier: TransformerIdentifier =

TransformerIdentifier("ForYouPushToHomeTweet")

override def transform(input: TweetCandidate): FeatureMap =

FeatureMapBuilder().add(SuggestTypeFeature, Some(st.SuggestType.Magicrec)).build()

})

override val resultTransformer: CandidatePipelineResultsTransformer[

TweetCandidate,

TweetCandidate

] = identity

override val candidateSource: CandidateSource[

ForYouQuery,

TweetCandidate

] = PassthroughCandidateSource(

CandidateSourceIdentifier("PushToHomeTweet"),

{ query => query.pushToHomeTweetId.toSeq.map(TweetCandidate(\_)) }

)

override val decorator: Option[

CandidateDecorator[ForYouQuery, TweetCandidate]

] = {

val tweetItemBuilder = TweetCandidateUrtItemBuilder(

clientEventInfoBuilder = HomeClientEventInfoBuilder()

)

Some(UrtItemCandidateDecorator(tweetItemBuilder))

}

}