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

import com.twitter.product\_mixer.component\_library.decorator.urt.builder.contextual\_ref.ContextualTweetRefBuilder

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

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

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

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

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

import com.twitter.product\_mixer.core.functional\_component.decorator.urt.builder.item.tweet.BaseEntryIdToReplaceBuilder

import com.twitter.product\_mixer.core.functional\_component.decorator.urt.builder.item.tweet.BaseTimelinesScoreInfoBuilder

import com.twitter.product\_mixer.core.functional\_component.decorator.urt.builder.item.tweet.BaseTweetHighlightsBuilder

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

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

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

import com.twitter.product\_mixer.core.functional\_component.decorator.urt.builder.social\_context.BaseSocialContextBuilder

import com.twitter.product\_mixer.core.model.marshalling.response.urt.item.conversation\_annotation.ConversationAnnotation

import com.twitter.product\_mixer.core.model.marshalling.response.urt.item.forward\_pivot.ForwardPivot

import com.twitter.product\_mixer.core.model.marshalling.response.urt.item.tombstone.TombstoneInfo

import com.twitter.product\_mixer.core.model.marshalling.response.urt.item.tweet.Tweet

import com.twitter.product\_mixer.core.model.marshalling.response.urt.item.tweet.TweetDisplayType

import com.twitter.product\_mixer.core.model.marshalling.response.urt.item.tweet.TweetItem

import com.twitter.product\_mixer.core.model.marshalling.response.urt.metadata.Badge

import com.twitter.product\_mixer.core.model.marshalling.response.urt.promoted.PrerollMetadata

import com.twitter.product\_mixer.core.model.marshalling.response.urt.promoted.PromotedMetadata

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

case object TweetCandidateUrtItemBuilder {

val TweetClientEventInfoElement = "tweet"

}

case class TweetCandidateUrtItemBuilder[Query <: PipelineQuery, Candidate <: BaseTweetCandidate](

clientEventInfoBuilder: BaseClientEventInfoBuilder[Query, Candidate],

displayType: TweetDisplayType = Tweet,

entryIdToReplaceBuilder: Option[BaseEntryIdToReplaceBuilder[Query, Candidate]] = None,

socialContextBuilder: Option[BaseSocialContextBuilder[Query, Candidate]] = None,

highlightsBuilder: Option[BaseTweetHighlightsBuilder[Query, Candidate]] = None,

innerTombstoneInfo: Option[TombstoneInfo] = None,

timelinesScoreInfoBuilder: Option[BaseTimelinesScoreInfoBuilder[Query, Candidate]] = None,

hasModeratedReplies: Option[Boolean] = None,

forwardPivot: Option[ForwardPivot] = None,

innerForwardPivot: Option[ForwardPivot] = None,

feedbackActionInfoBuilder: Option[BaseFeedbackActionInfoBuilder[Query, Candidate]] = None,

promotedMetadata: Option[PromotedMetadata] = None,

conversationAnnotation: Option[ConversationAnnotation] = None,

contextualTweetRefBuilder: Option[ContextualTweetRefBuilder[Candidate]] = None,

prerollMetadata: Option[PrerollMetadata] = None,

replyBadge: Option[Badge] = None,

destinationBuilder: Option[BaseUrlBuilder[Query, Candidate]] = None)

extends CandidateUrtEntryBuilder[Query, Candidate, TweetItem] {

override def apply(

pipelineQuery: Query,

tweetCandidate: Candidate,

candidateFeatures: FeatureMap

): TweetItem = {

val isPinned = candidateFeatures.getTry(IsPinnedFeature).toOption

TweetItem(

id = tweetCandidate.id,

entryNamespace = TweetItem.TweetEntryNamespace,

sortIndex = None, // Sort indexes are automatically set in the domain marshaller phase

clientEventInfo = clientEventInfoBuilder(

pipelineQuery,

tweetCandidate,

candidateFeatures,

Some(TweetClientEventInfoElement)),

feedbackActionInfo = feedbackActionInfoBuilder.flatMap(

\_.apply(pipelineQuery, tweetCandidate, candidateFeatures)),

isPinned = isPinned,

entryIdToReplace =

entryIdToReplaceBuilder.flatMap(\_.apply(pipelineQuery, tweetCandidate, candidateFeatures)),

socialContext =

socialContextBuilder.flatMap(\_.apply(pipelineQuery, tweetCandidate, candidateFeatures)),

highlights =

highlightsBuilder.flatMap(\_.apply(pipelineQuery, tweetCandidate, candidateFeatures)),

displayType = displayType,

innerTombstoneInfo = innerTombstoneInfo,

timelinesScoreInfo = timelinesScoreInfoBuilder

.flatMap(\_.apply(pipelineQuery, tweetCandidate, candidateFeatures)),

hasModeratedReplies = hasModeratedReplies,

forwardPivot = forwardPivot,

innerForwardPivot = innerForwardPivot,

promotedMetadata = promotedMetadata,

conversationAnnotation = conversationAnnotation,

contextualTweetRef = contextualTweetRefBuilder.flatMap(\_.apply(tweetCandidate)),

prerollMetadata = prerollMetadata,

replyBadge = replyBadge,

destination =

destinationBuilder.map(\_.apply(pipelineQuery, tweetCandidate, candidateFeatures))

)

}

}