package com.twitter.frigate.pushservice.model.ntab

import com.twitter.frigate.common.base.TweetAuthorDetails

import com.twitter.frigate.common.base.TweetCandidate

import com.twitter.frigate.pushservice.exception.TweetNTabRequestHydratorException

import com.twitter.frigate.pushservice.model.PushTypes.PushCandidate

import com.twitter.frigate.pushservice.params.PushFeatureSwitchParams

import com.twitter.notificationservice.thriftscala.InlineCard

import com.twitter.notificationservice.thriftscala.StoryContext

import com.twitter.notificationservice.thriftscala.StoryContextValue

import com.twitter.frigate.pushservice.util.EmailLandingPageExperimentUtil

import com.twitter.notificationservice.thriftscala.\_

import com.twitter.util.Future

trait TweetNTabRequestHydrator extends NTabRequestHydrator {

self: PushCandidate with TweetCandidate with TweetAuthorDetails =>

override def senderIdFut: Future[Long] =

tweetAuthor.map {

case Some(author) => author.id

case \_ =>

throw new TweetNTabRequestHydratorException(

s"Unable to obtain Author ID for: $commonRecType")

}

override def storyContext: Option[StoryContext] = Some(

StoryContext(

altText = "",

value = Some(StoryContextValue.Tweets(Seq(tweetId))),

details = None

))

override def inlineCard: Option[InlineCard] = Some(InlineCard.TweetCard(TweetCard(tweetId)))

override lazy val tapThroughFut: Future[String] = {

Future.join(tweetAuthor, target.deviceInfo).map {

case (Some(author), Some(deviceInfo)) =>

val enableRuxLandingPage = deviceInfo.isRuxLandingPageEligible && target.params(

PushFeatureSwitchParams.EnableNTabRuxLandingPage)

val authorProfile = author.profile.getOrElse(

throw new TweetNTabRequestHydratorException(

s"Unable to obtain author profile for: ${author.id}"))

if (enableRuxLandingPage) {

EmailLandingPageExperimentUtil.createNTabRuxLandingURI(authorProfile.screenName, tweetId)

} else {

s"${authorProfile.screenName}/status/${tweetId.toString}"

}

case \_ =>

throw new TweetNTabRequestHydratorException(

s"Unable to obtain author and target details to generate tap through for Tweet: $tweetId")

}

}

override def socialProofDisplayText: Option[DisplayText] = None

}