package com.twitter.frigate.pushservice.send\_handler.generator

import com.twitter.frigate.common.base.MagicFanoutProductLaunchCandidate

import com.twitter.frigate.magic\_events.thriftscala.MagicEventsReason

import com.twitter.frigate.magic\_events.thriftscala.ProductType

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

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

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

import com.twitter.frigate.thriftscala.CommonRecommendationType

import com.twitter.frigate.thriftscala.FrigateNotification

import com.twitter.util.Future

object MagicFanoutProductLaunchCandidateGenerator extends CandidateGenerator {

override def getCandidate(

targetUser: PushTypes.Target,

notification: FrigateNotification

): Future[PushTypes.RawCandidate] = {

require(

notification.commonRecommendationType == CommonRecommendationType.MagicFanoutProductLaunch,

"MagicFanoutProductLaunch: unexpected CRT " + notification.commonRecommendationType

)

require(

notification.magicFanoutProductLaunchNotification.isDefined,

"MagicFanoutProductLaunch: magicFanoutProductLaunchNotification is not defined")

require(

notification.magicFanoutProductLaunchNotification.exists(\_.magicFanoutPushId.isDefined),

"MagicFanoutProductLaunch: magicFanoutPushId is not defined")

require(

notification.magicFanoutProductLaunchNotification.exists(\_.fanoutReasons.isDefined),

"MagicFanoutProductLaunch: fanoutReasons is not defined")

val magicFanoutProductLaunchNotification = notification.magicFanoutProductLaunchNotification.get

val candidate = new RawCandidate with MagicFanoutProductLaunchCandidate {

override val target: Target = targetUser

override val pushId: Long =

magicFanoutProductLaunchNotification.magicFanoutPushId.get

override val candidateMagicEventsReasons: Seq[MagicEventsReason] =

magicFanoutProductLaunchNotification.fanoutReasons.get

override val productLaunchType: ProductType =

magicFanoutProductLaunchNotification.productLaunchType

override val frigateNotification: FrigateNotification = notification

}

Future.value(candidate)

}

}