package com.twitter.product\_mixer.component\_library.candidate\_source.earlybird

import com.twitter.search.earlybird.{thriftscala => t}

import com.twitter.inject.Logging

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

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

import com.twitter.stitch.Stitch

import javax.inject.Inject

import javax.inject.Singleton

@Singleton

class EarlybirdTweetCandidateSource @Inject() (

earlybirdService: t.EarlybirdService.MethodPerEndpoint)

extends CandidateSource[t.EarlybirdRequest, t.ThriftSearchResult]

with Logging {

override val identifier: CandidateSourceIdentifier = CandidateSourceIdentifier("EarlybirdTweets")

override def apply(request: t.EarlybirdRequest): Stitch[Seq[t.ThriftSearchResult]] = {

Stitch

.callFuture(earlybirdService.search(request))

.map { response: t.EarlybirdResponse =>

response.searchResults.map(\_.results).getOrElse(Seq.empty)

}

}

}