package com.twitter.unified\_user\_actions.adapter

import com.twitter.inject.Test

import com.twitter.unified\_user\_actions.adapter.TestFixtures.EmailNotificationEventFixture

import com.twitter.unified\_user\_actions.adapter.email\_notification\_event.EmailNotificationEventUtils

class EmailNotificationEventUtilsSpec extends Test {

test("Extract TweetId from pageUrl") {

new EmailNotificationEventFixture {

val invalidUrls: Seq[String] =

List("", "abc.com/what/not?x=y", "?abc=def", "12345/", "12345/?")

val invalidDomain = "https://twitter.app.link/addressbook/"

val numericHandle =

"https://twitter.com/1234/status/12345?cxt=HBwWgsDTgY3tp&cn=ZmxleGl&refsrc=email)"

assert(EmailNotificationEventUtils.extractTweetId(pageUrlStatus).contains(tweetIdStatus))

assert(EmailNotificationEventUtils.extractTweetId(pageUrlEvent).contains(tweetIdEvent))

assert(EmailNotificationEventUtils.extractTweetId(pageUrlNoArgs).contains(tweetIdNoArgs))

assert(EmailNotificationEventUtils.extractTweetId(invalidDomain).isEmpty)

assert(EmailNotificationEventUtils.extractTweetId(numericHandle).contains(12345L))

invalidUrls.foreach(url => assert(EmailNotificationEventUtils.extractTweetId(url).isEmpty))

}

}

test("Extract TweetId from LogBase") {

new EmailNotificationEventFixture {

assert(EmailNotificationEventUtils.extractTweetId(logBase1).contains(tweetIdStatus))

}

}

}