**FIAP**

**MBA EM DESENVOLVIMENTO DE APLICAÇÕES JAVA – SOA E INTERNET DAS COISAS**

**Twitter Small Analytics**

**#java8**

**FUNDAMENTOS DA TECNOLOGIA JAVA E UML**

**FIAP**

**MBA EM DESENVOLVIMENTO DE APLICAÇÕES JAVA – SOA E INTERNET DAS COISAS**

**Twitter Small Analytics**

**#java8**

HELENA STRADA FRANCO DE SOUZA – RM31321

JULIO OLIVEIRA DA SILVA – RM31524

VINICIUS RIGO – RM31667

**FUNDAMENTOS DA TECNOLOGIA JAVA E UML**

Sumário

[Considerações Iniciais 4](#_Toc468963061)

[Manual De Utilização 5](#_Toc468963062)

[Importar Projeto no Eclipse 5](#_Toc468963063)

[Configurar Chaves de Acesso 5](#_Toc468963064)

[Componentes, Bibliotecas e Frameworks 7](#_Toc468963065)

[Pacotes, classes e métodos 8](#_Toc468963066)

[Diagrama de classes 10](#_Toc468963067)

[Diagrama de sequência 11](#_Toc468963068)

[Capturas de tela e aplicação 15](#_Toc468963069)

[GitHub 16](#_Toc468963070)

# Considerações Iniciais

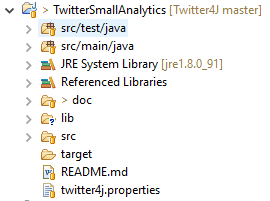
O Twitter4J é uma biblioteca não oficial que permite realizar chamadas para a API do Twitter através da linguagem Java, podendo realizar a contagem e o monitoramento de tweets em tempo real.

# Manual De Utilização

## Importar Projeto no Eclipse

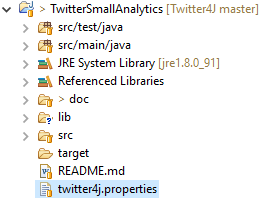
Após realizar o download do projeto disponível no GitHub, e descompactar o zip em sua pasta de preferência, realize os passos a seguir para importar o projeto no eclipse.

1. Abrir o eclipse;
2. Clicar em File > Import;
3. Escolha General > Projects from Folder or Archive:
   1. Clique em ‘Next’;
4. Clique em ‘Directory’:
   1. Selecione a pasta desejada;
5. Clique em Finish.



## Configurar Chaves de Acesso

Para utilizar o Twitter Small Analytics é necessário configurar as chaves de acesso no arquivo twitter4j.properties.



debug=false

oauth.consumerKey=\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

oauth.consumerSecret=\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

oauth.accessToken=\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

oauth.accessTokenSecret=\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Com a chave e o token de acesso configurados é necessário criar uma instância da classe *TwitterSmallAnalytics.java* passando como parâmetro a hashtag que desejar, para que a contagem dos tweets ocorra. Após isso deve-se escolher uma estratégia de contagem de dados dos tweets definidos na classe enum *Action.java*.

# Componentes, Bibliotecas e Frameworks

# Pacotes, classes e métodos

A disposição dos itens a seguir, possuem as seguintes características:

**Caminho/do/pacote**

Definição do pacote

**NomeDaClasse.java**

Definição da classe

**Métodos**

Definição dos métodos

**src/test/java/scj28/java/atividadefinal**

Pacote que contém a classe principal para acesso à API do twitter.

**TwitterSmallAnalyticsTestDrive.java**

Classe de teste que executa os principais métodos de sistema para acesso a API do twitter.

**src/main/java/scj28/java/atividadefinal**

Pacote que contém todas as classes da aplicação

**TwitterSmallAnalytics.java**

Classe principal do sistema responsável por executar as consultas à base do Twitter.

**src/main/java/scj28/java/atividadefinal/enumeration**

Pacote que contém as classes Enum.

**Action.java**

Classe enum que define os tipos de ordenação disponíveis no sistema.

**TipoDeOrdenacao.java**

Classe enum que define os tipos de ordenação disponíveis do sistema (Por autor ou data do tweet).

**src/main/java/scj28/java/atividadefinal/strategy**

Pacote onde ficam definidas as estratégias que mudam o comportamento do sistema em tempo de execução

**Actionable.java**

Interface que define a ação que será executada utilizando o padrão de projetos strategy.

**Favorite.java**

Ação que busca a quantidade de favoritações por dia na última semana.

**Organizer.java**

Ação que busca e imprime os tweets ordenados pelo nome do autor.

**Retweet.java**

Ação que busca a quantidade de retweets da última semana.

**Tweet.java**

Ação que busca a quantidade de tweets da última semana.

**scj28.java.atividadefinal.vo**

Pacote onde são colocados os Vos.

**TweetVO.java**

POJO criado para tragefar os dados necessários para executar as ações seguindo o padrão Value Object.

# Diagrama de classes

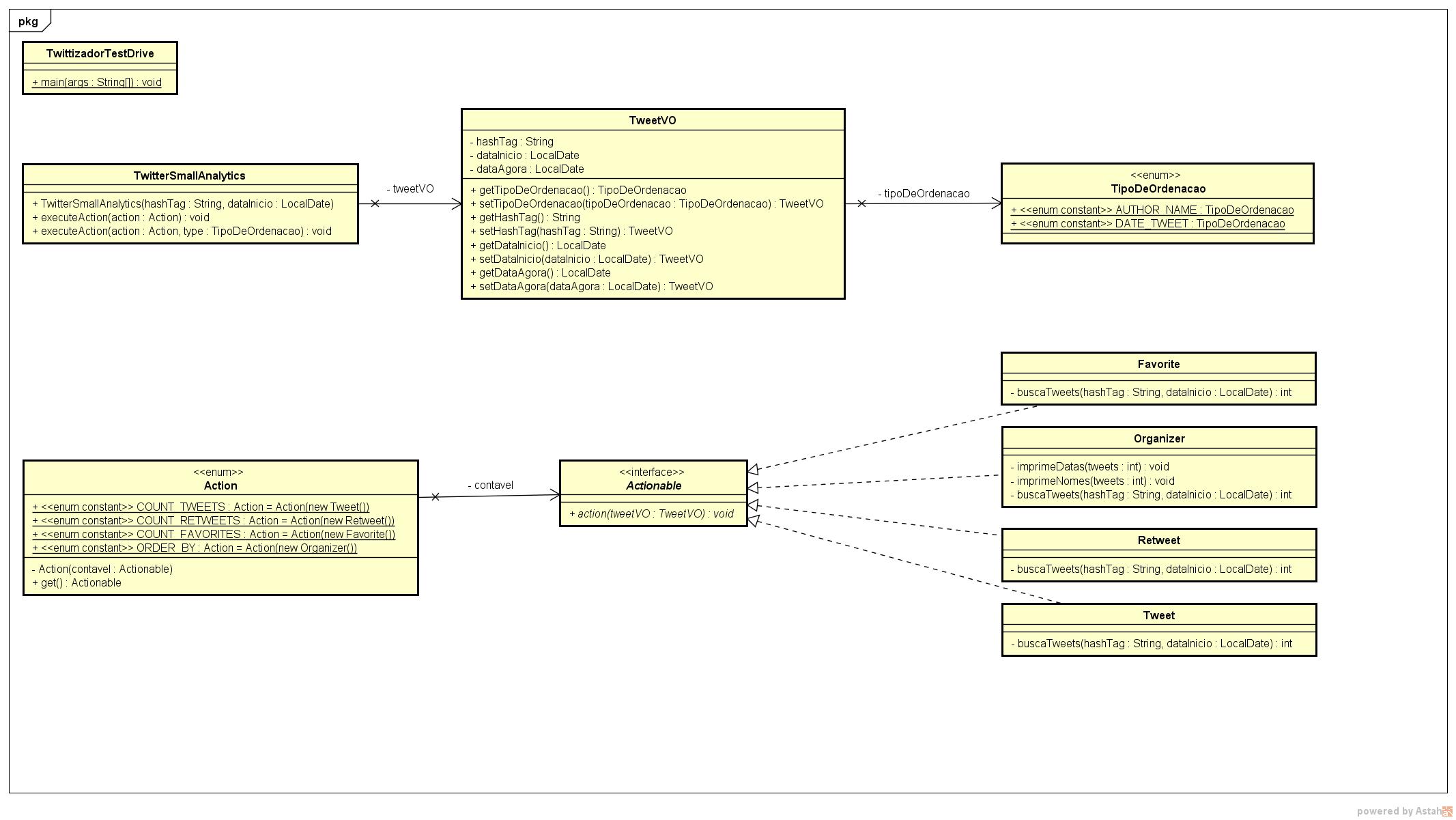


Figura 1 - Diagrama de Classes

# Diagrama de sequência

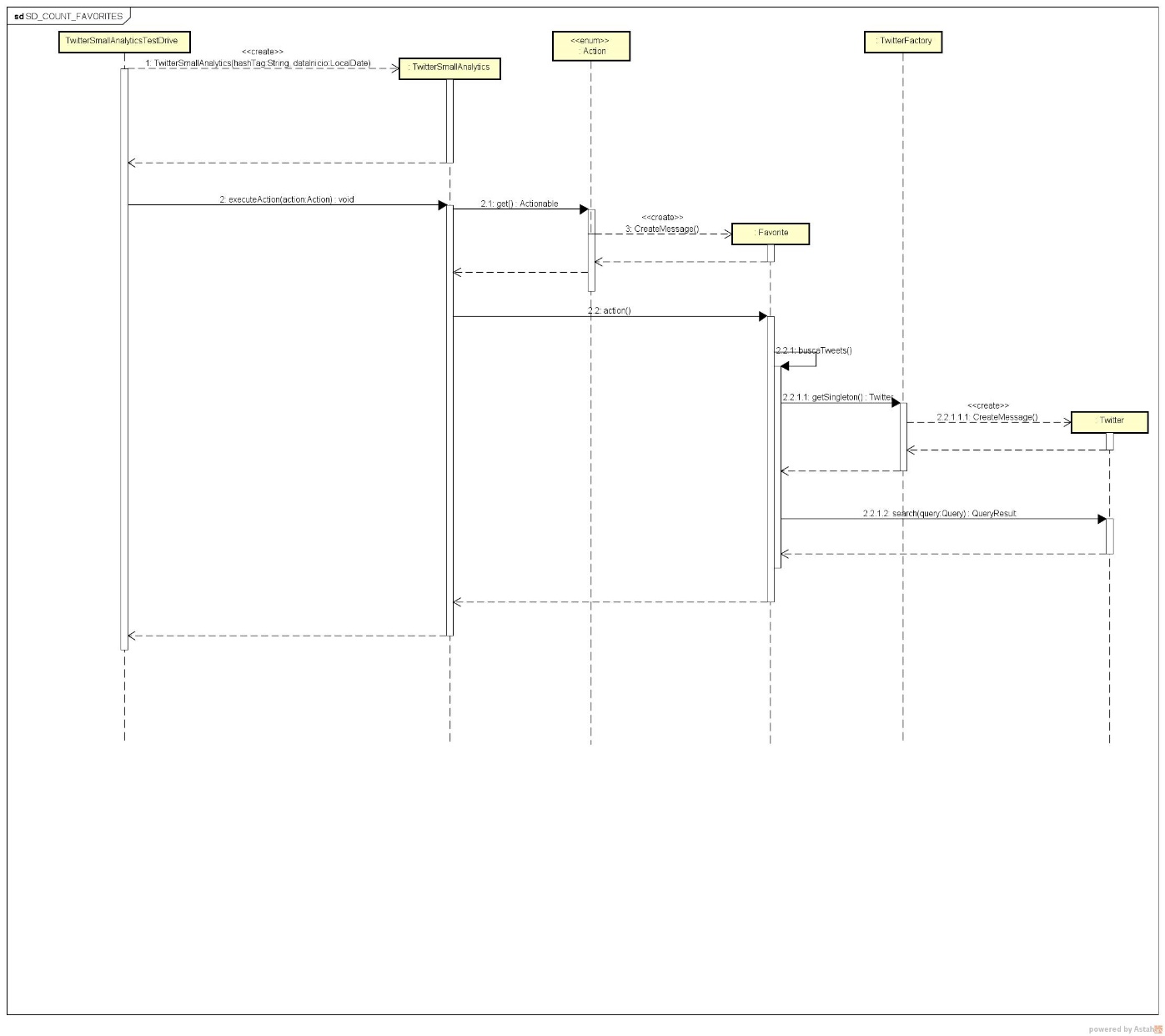


Figura 2 - Diagrama de Sequência - Contar Favoritos

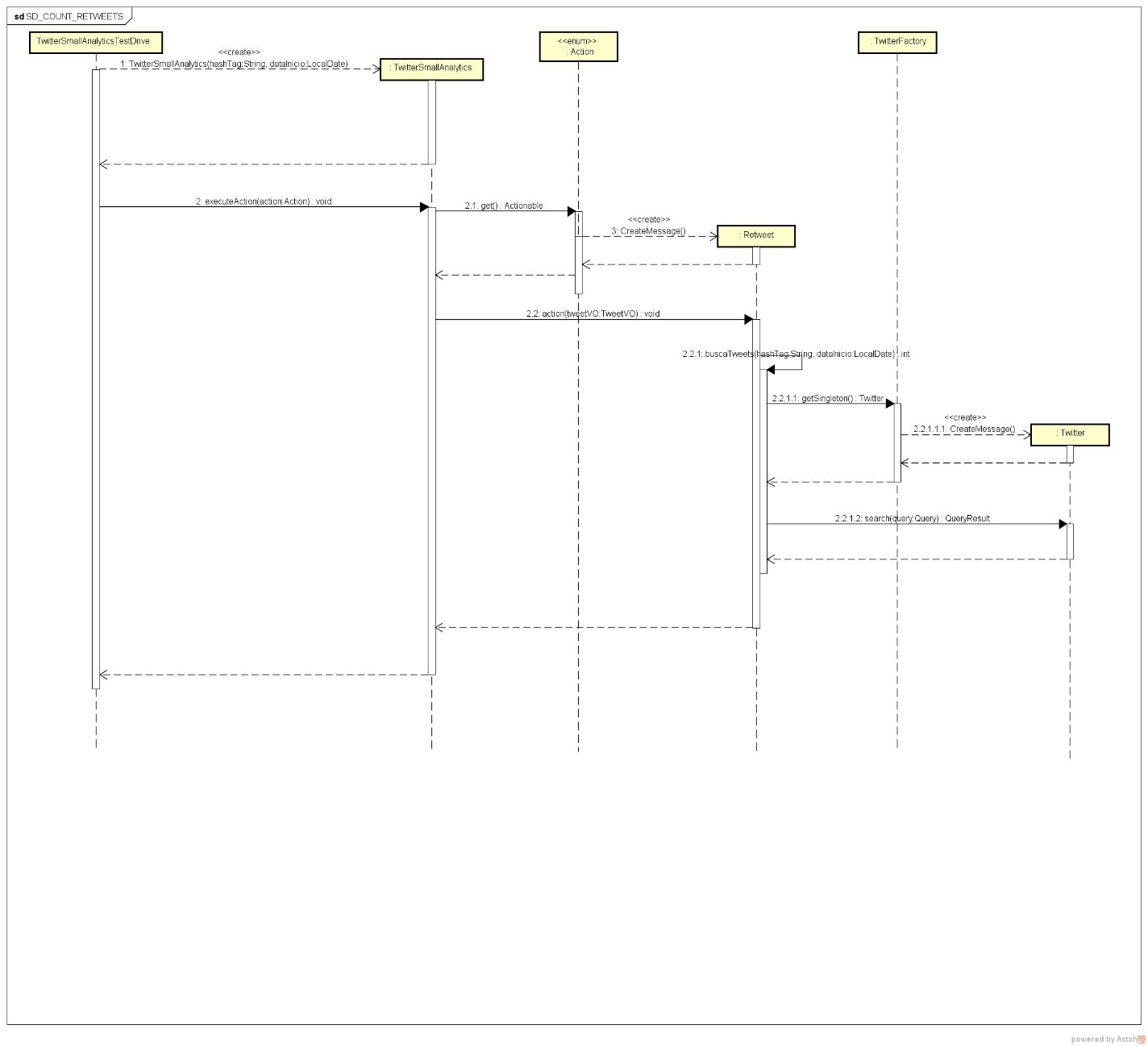


Figura 3 - Diagrama de Sequências - Contar Retweets

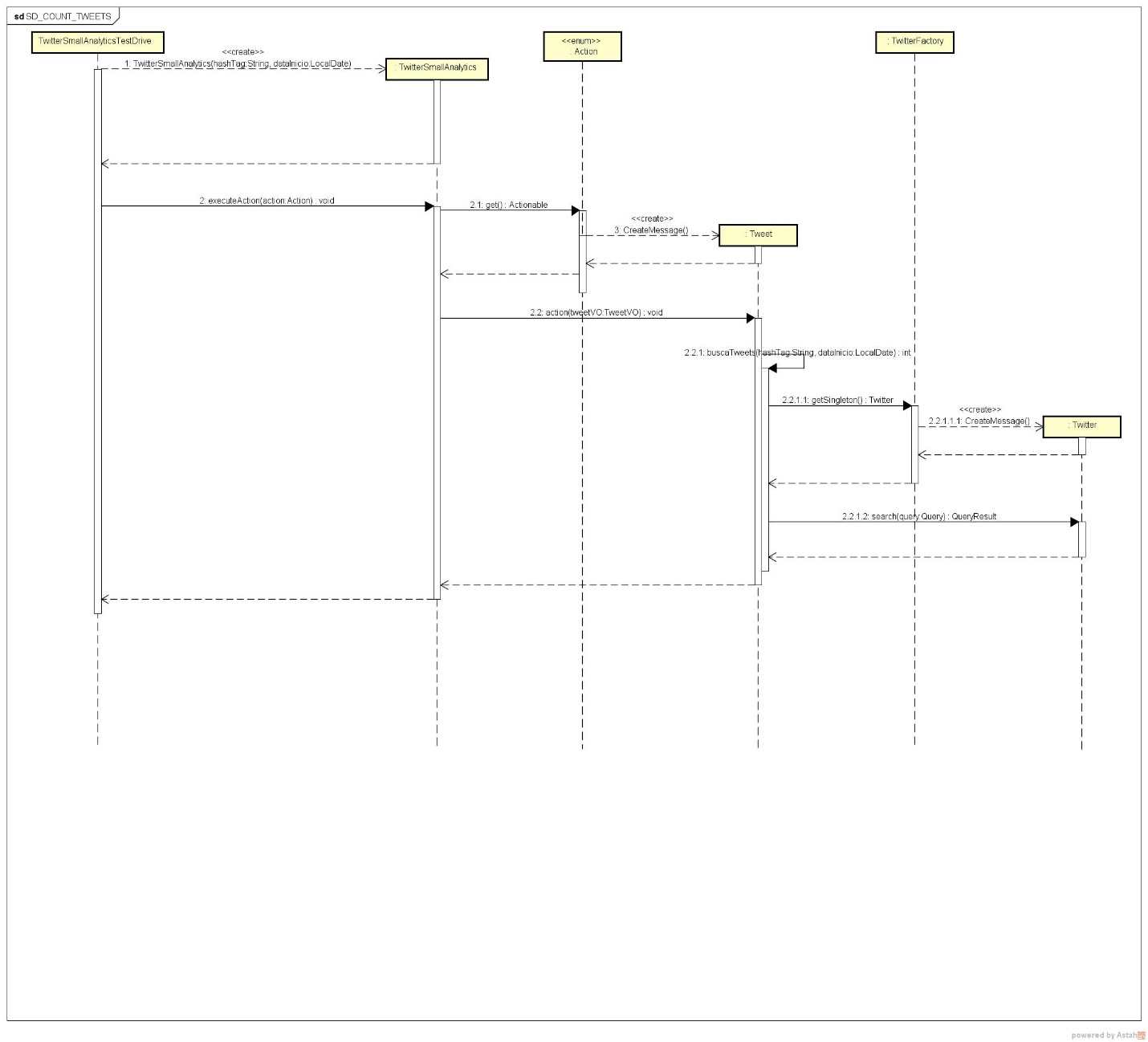


Figura 4 - Diagrama de Sequência - Contar Tweets

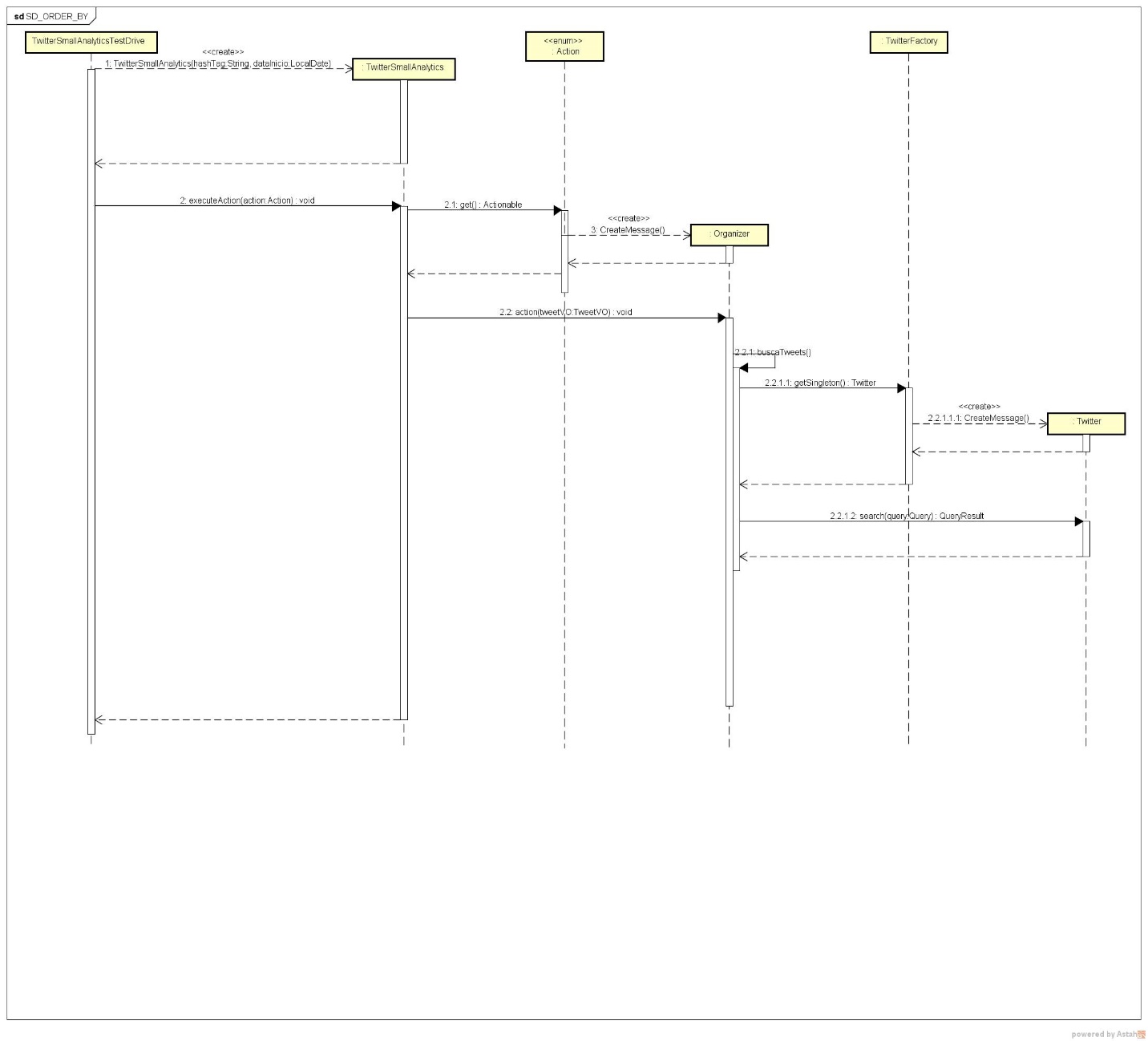


Figura 5 - Diagrama de Sequência - Ordenar

# Capturas de tela e aplicação

Esta versão de saída foi executada no dia 07 de novembro de 2016 com a hashtag **#java8**.

Data Inicial - 2016-10-31 | Data Final - 2016-11-07

Quantidade por dia de tweets da última semana:

2016-10-31 15 tweets.

2016-11-01 15 tweets.

2016-11-02 30 tweets.

2016-11-03 30 tweets.

2016-11-04 30 tweets.

2016-11-05 30 tweets.

2016-11-06 15 tweets.

<===========================================>

Quantidade por dia de retweets da última semana:

2016-10-31 81 retweets.

2016-11-01 27 retweets.

2016-11-02 123 retweets.

2016-11-03 113 retweets.

2016-11-04 28 retweets.

2016-11-05 34 retweets.

2016-11-06 24 retweets.

<===========================================>

Quantidade por dia de favoritações da última semana:

2016-10-31 31 favoritações.

2016-11-01 15 favoritações.

2016-11-02 13 favoritações.

2016-11-03 31 favoritações.

2016-11-04 16 favoritações.

2016-11-05 12 favoritações.

2016-11-06 3 favoritações.

<===========================================>

Primeiro Nome: @AAMECSTHA2

Ultimo Nome: @\_MATIWEB\_

Data mais antiga: 30/10/2016 08:30

Data mais recente: 05/11/2016 19:33

# GitHub

O código fonte da aplicação está disponível em: <https://github.com/hstrada/Twitter4J>.