FACULDADE DE INFORMÁTICA E ADMINISTRAÇÃO PAULISTA
FIAP

RELATÓRIO TÉCNICO DO PROJETO DE INTEGRAÇÃO COM O TWITTER

Trabalho final para o módulo de
Fundamentos da Tecnologia Java e
Modelagem UML 2.0 para o curso de MBA
em Desenvolvimento de Aplicações Java,
SOA e Internet das coisas

Professor: Michel Pereira Fernandes

MARCOS ANTONIO SOUZA PINHEIRO
RM: 30366
github: https://github.com/marcos-pinheiro/27SCJ_AtividadeFinalTwitter.git

SÃO PAULO
2016

Sumário

1. Bibliotecas.....3
2. Organização do sistema.....4
3. Diagramas de classe.....6
4. Diagramas de sequencia.....7
5. Screenshots.....9

SÃO PAULO
2016

1 Bibliotecas

1. Twitter4J

   Neste projeto foi utilizado a biblioteca Twitter4j. Esta biblioteca possui classes que abstrai funcionalidades nos quais executam chamadas HTTP a API do Twitter. Podemos citar funcionalidades tais como: “Tweetar”, obter tweets em um determinado período, obter retweets em um determinado período, etc.

1. Stream API

   Neste projeto também foi utilizado a API de stream do Java 8. Esta API que cria um fluxo em coleções de dados aplicando funcionalidades tais como: filtros, transformações, coletas, redução e etc.

2. Java Concurrent API

   A API de concorrência do Java fornece uma série de classes e métodos para manipular, executar e reciclar threads em aplicações concorrentes.

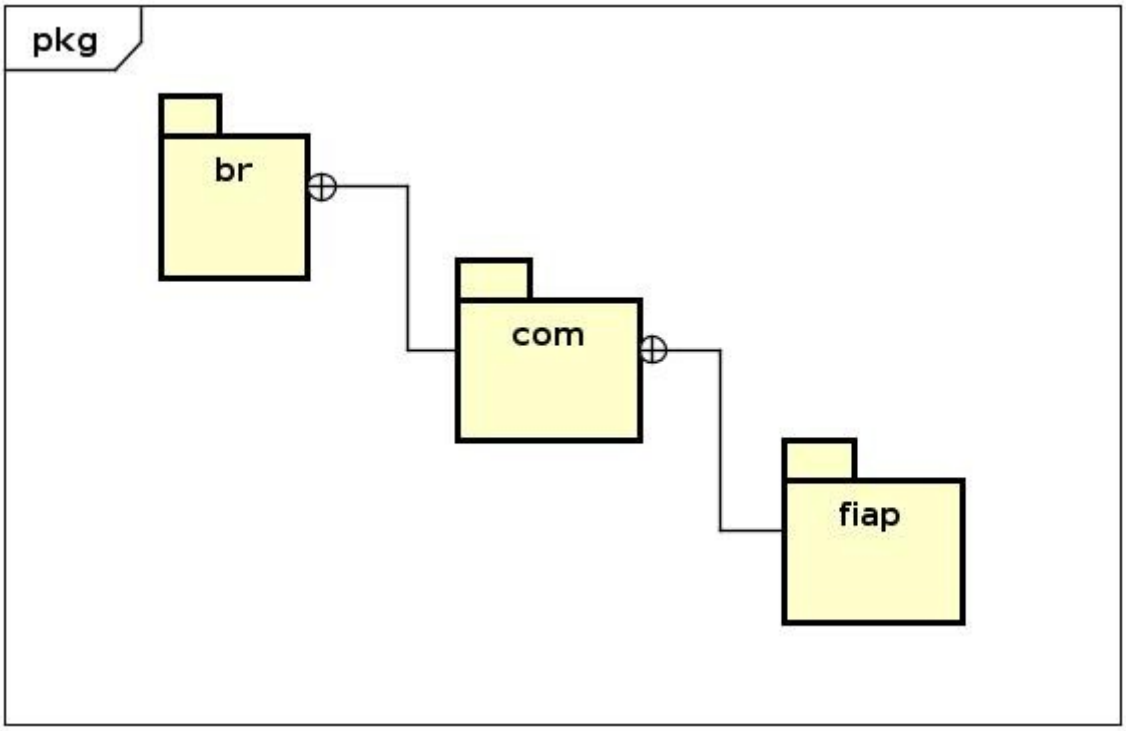
3. Collections API

   Esta API fornece uma série de classes e interfaces para os mais variados tipos de estruturas de dados. Desde estruturas como Listas, Filas, Pilhas, coleções de conjuntos e chave-valor, threads-safe e non-theads-safe, entre outros.

2 Organização do sistema

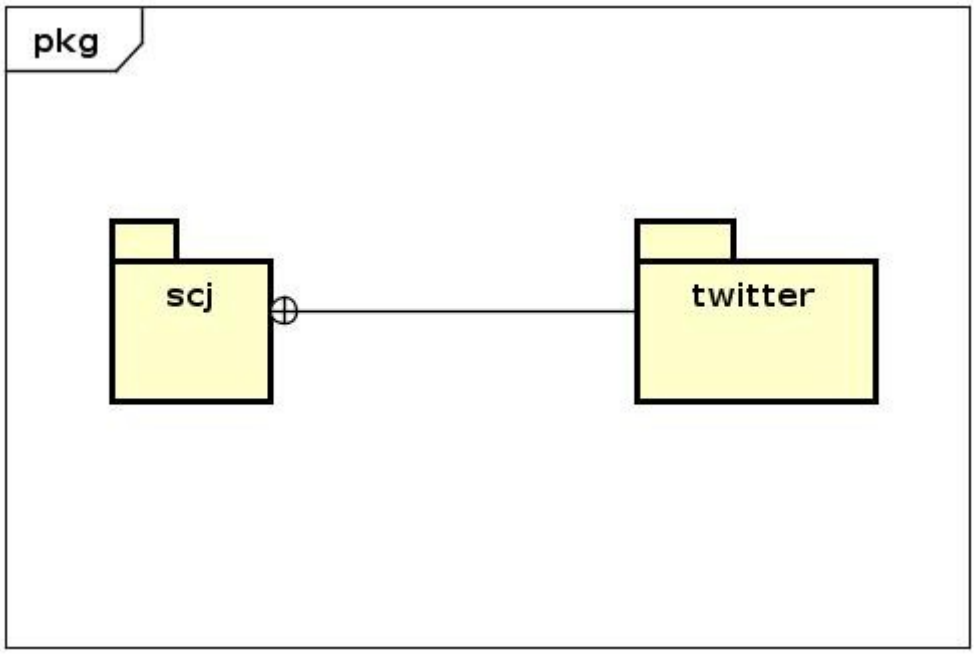
Para uma melhor organização das classes do projeto, em java utilizamos os pacotes/packages. Com os pacotes podemos agrupar um conjunto de classes que pertençam a um mesmo domínio ou objetivo.

Neste projeto organizamos nossos pacotes da seguinte maneira:

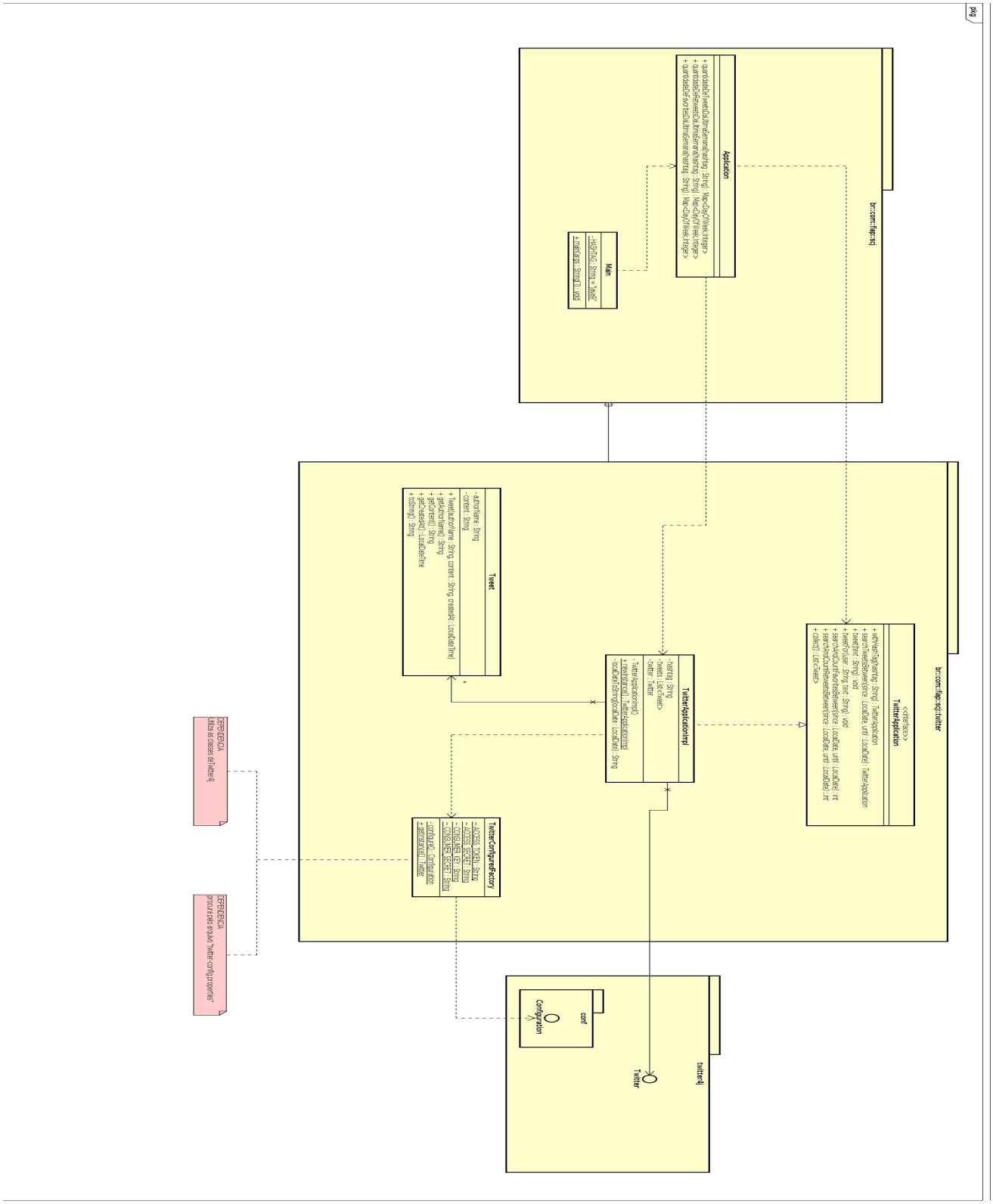


Por convenção, criamos nossos pacotes inicialmente com o domínio da aplicação ao contrario. No caso br.com.fiap

Após esta estrutura é definido o pacote “scj” onde ficarão as classes relacionadas a um contexto geral da aplicação, como por exemplo a classe do método main. E dentro deste pacote existe outro pacote chamado “twitter” no qual possui classes e interfaces que trabalham diretamente com a API do Twitter4J encapsulando as funcionalidades solicitadas neste projeto.



3 Diagramas de classe



OBS: Imagens e o projeto do diagrama acompanha o projeto no Github

4 Diagramas de sequencia

Diagrama de sequencia relacionado ao método quantidade de tweetsTest

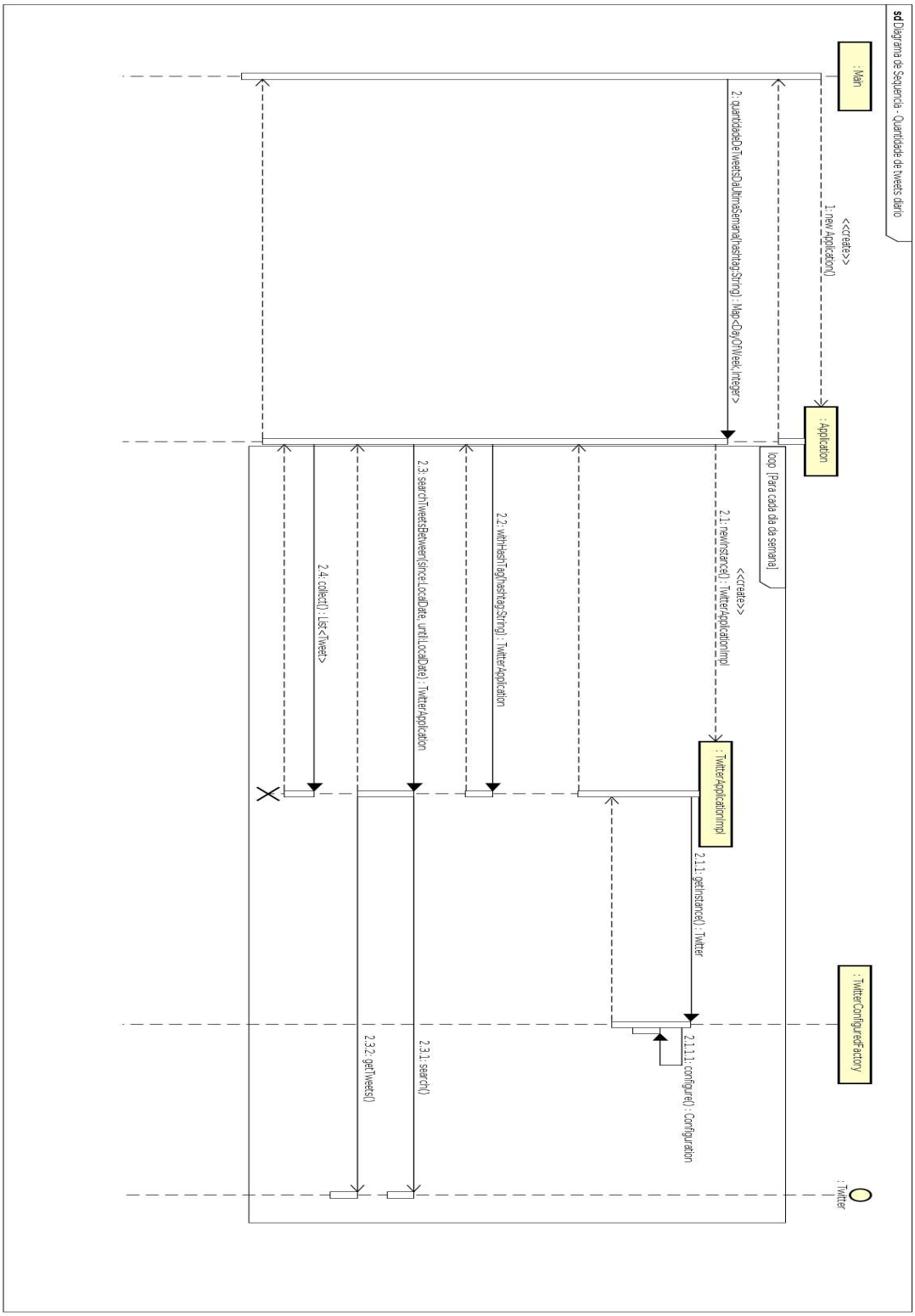
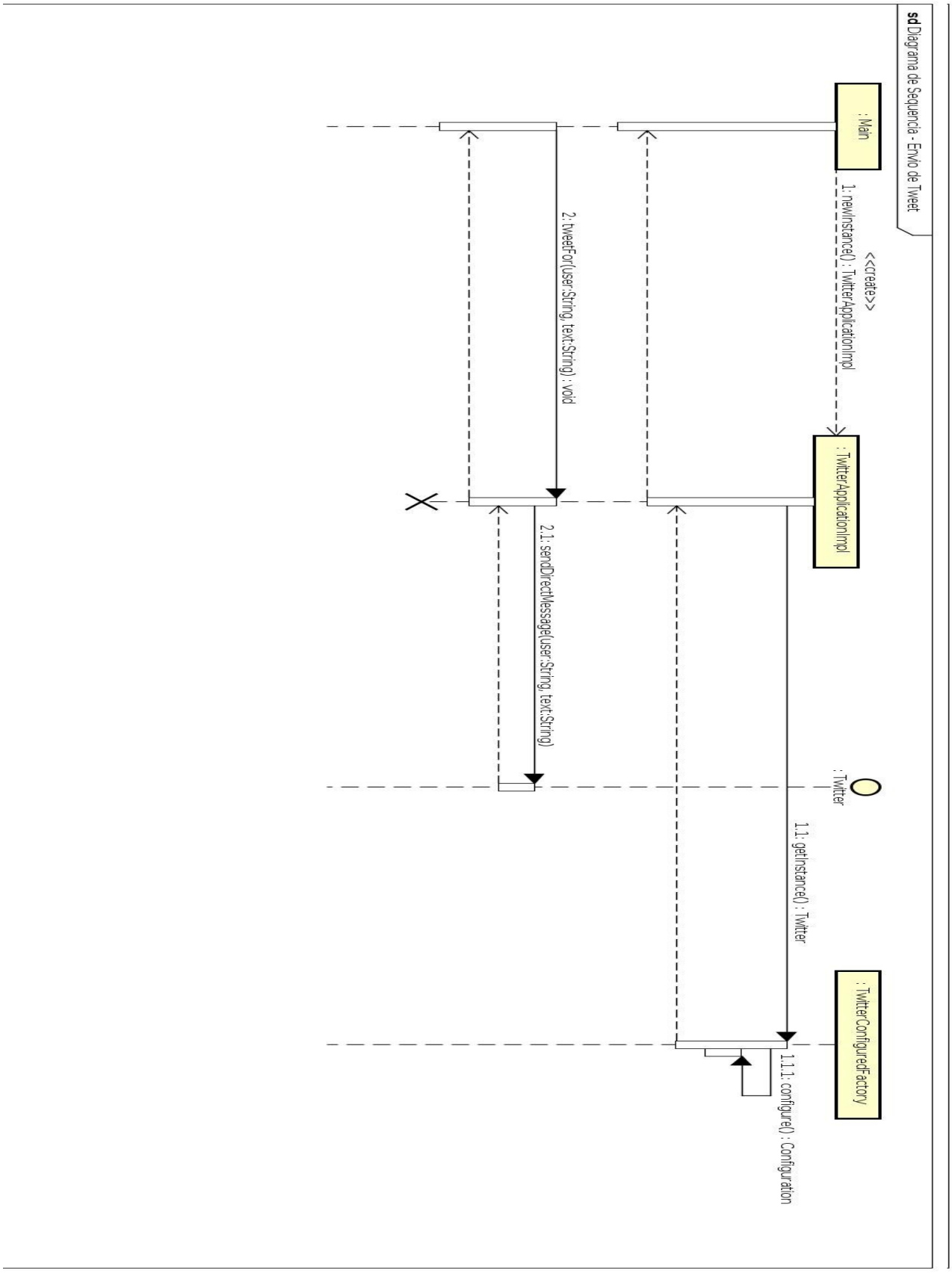


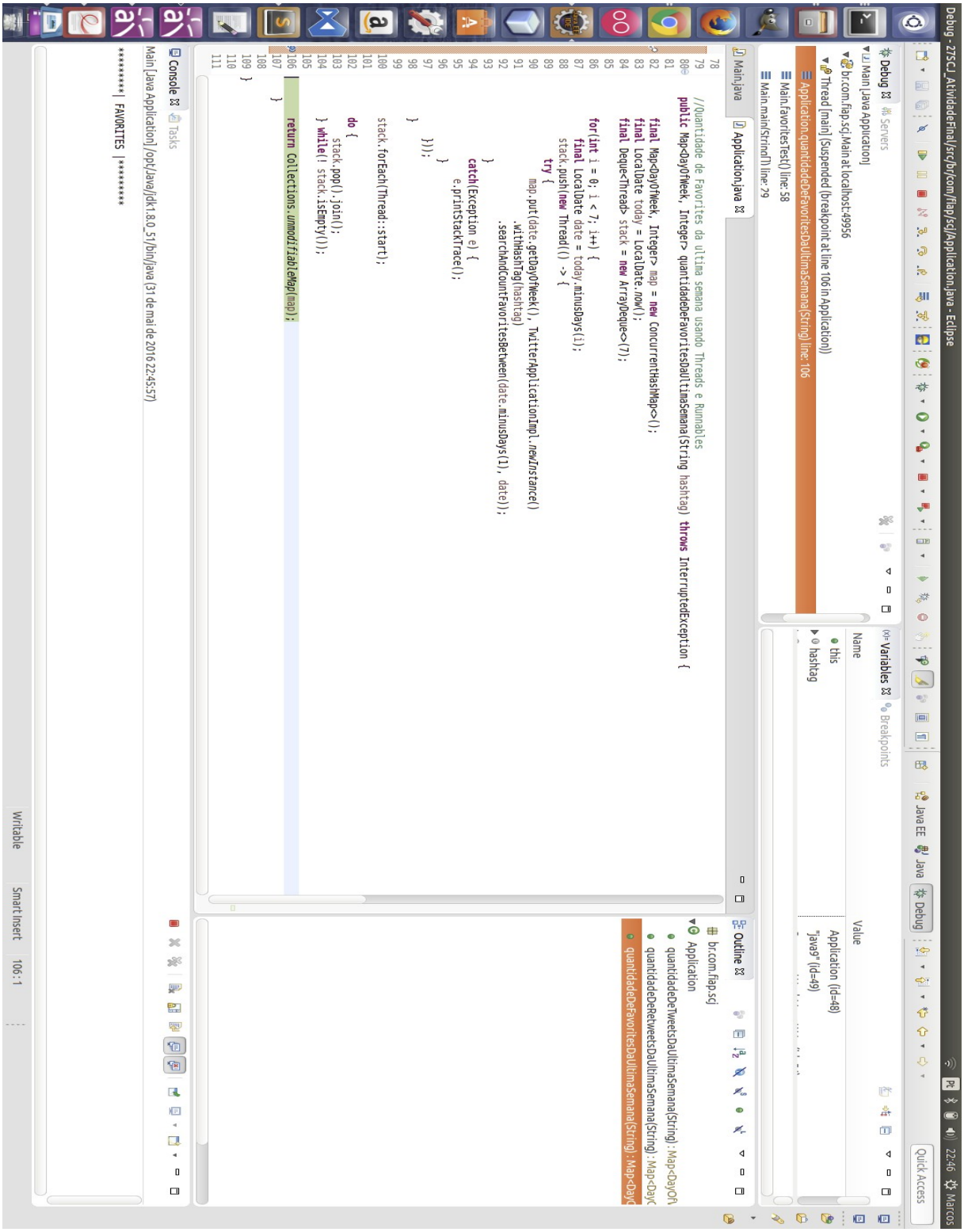
Diagrama de sequencia relacionado ao método tweetForTest



SÃO PAULO
2016

OBS: Imagens e o projeto do diagrama acompanha o projeto no Github

5 Screenshots



Este método recupera a quantidade de favorites da ultima semana criando uma Thread para cada dia da semana para otimizar o tempo da busca.

Primeiramente se é definido um pilha do tipo Thread com sua capacidade igual a quantidade de dias da semana. Depois, em um loop com a quantidade de dias da semana, é armazenada na pilha instancias de Threads com classes anonimas de Runnable contendo nossa lógica de busca para um determinado dia em questão.

Após armazenar todos, finalmente se é executado o método start() percorrendo os elementos da pilha e após isso se é desempilhada os elementos da pilha de Thread invocando o método join() para aguardar o término da thread e a mesma se juntar a Thread principal. Quando todas as threads executarem se é retornado uma coleção chave-valor do tipo Map somente leitura contendo as quantidades e o dia da semana em questão.

SÃO PAULO
2016

Este método utiliza as classes da API Twitter4J para efetuar buscas através da classe Query. Com esta classe definimos as datas de inicio e fim e através do método search efetuamos a busca.

Este método retorna um QueryResult que deve ser iterado de forma página quando os resultados ultrapassam o valor padrão de resultados por página.

Enquanto iteramos os resultados estes são transformados, reduzidos e armazenados através da API Stream do Java 8