Lucidy Scrap

Maylon Felix de Brito - 09/0011171*
Kamilla Holanda - 09/0038363†

September 25, 2012

Abstract

Esse relatório apresenta um sistema desenvolvido em linguagem Python que utiliza um sistema concorrente para realizar a tarefa web scraping das variáveis em um período de tempo regular. As variáveis extraídas são armazenadas em um hashset desenvolvido em linguagem C. Desse modo, são exploradas as características de Programação Concorrente, Programação Imperativa e Programação Orientada a Objetos.

1 Introdução

Um dos maiores objetivos da moderna Administração Pública é estímulo à transparência pública. A ampliação da divulgação das ações governamentais a maioria dos brasileiros contribui para o fortalecimento da democracia e prestigia e desenvolve as noções de cidadania.

As Páginas de Transparência Pública dão continuidade às ações de governo voltadas para o incremento da transparência e do controle social, com objetivo de divulgar as despesas realizadas pelos órgãos e entidades da Administração Pública Federal, informando sobre execução orçamentária, licitações, contratações, convênios, diárias e passagens.

Dessa forma, conforme dispõe a Portaria Interministerial nº 140, de 16 de março de 2006, cada órgão e entidade deve ter sua própria Página de Transparência com informações detalhadas. Seguindo o que dispõe essa portaria o governo do Governo do Distrito Federal (GDF) apresenta os seus gasto no Portal da Transparência do Distrito Federal [1].

Dada a importância que a transparência pública assumiu nos dias atuais, esse projeto tem por objetivo facilitar o acompanhamento e a visualização dos gastos do GDF por meio de um desenvolvimento de um sistema de monitoramento Web.

O sistema de Monitoramento consiste em extrair dados específicos a um determinado contexto na web, rastrear esse dado a uma dada frequência, e inferir

*e-mail:maylon.felix@gmail.com
†e-mail:holanda.kamilla@gmail.com
[1]Portal da Transparência do Distrito Federal:`http://www.transparencia.df.gov.br`.

informações sobre essa massa de dados. Nesse projeto esses dados serão apresentados em gráficos detalhados que facilitarão sua interpretação pelos cidadãos.

A disciplina de Paradigmas de Programação se propões a ensinar os fundamentos de linguagem de programação e proporcionar a prática de programação com os principais paradigmas de programação. O objetivo é proporcionar ao aluno uma visão geral dos conceitos envolvidos no projeto e no uso de paradigmas de programação para que ele tenha condições de selecionar a linguagem correta para a implementação de um sistema. Esse projeto de implementação de um sistema de monitoramento em tempo real se encaixa perfeitamente dentro do contexto da disciplina. Uma vez que está sendo desenvolvido em linguagem Python um sistema concorrente para realizar a tarefa web scraping das variáveis em período de tempo regular. As variáveis extraídas são armazenadas em um hashset desenvolvido em linguagem C. Desse modo, são exploradas as características de Programação Concorrente, Programação Imperativa e Programação Orientada a Objetos.

As entradas desse programa são basicamente as receitas e as despesas do GDF, sendo esses dados atualizadas diariamente. Nessa lista as variáveis extraídas são:

- Receita: Recursos auferidos na gestão, a serem computados na apuração do resultado do exercício, desdobrados nas categorias econômicas de correntes e de capital.

- Receita Corrente: No sentido amplo, consiste na soma de valores recebidos durante um determinado período de tempo. No setor público, é a soma de ingressos, impostos, taxas, contribuições e outras fontes de recursos, arrecadados para atender às despesas públicas.

- Receita de Capital: São os ingressos de recursos financeiros oriundos de atividades operacionais ou não operacionais para aplicação em despesas operacionais, correntes ou de capital, visando ao atingimento dos objetivos traçados nos programas e ações de governo. São denominados receita de capital porque são derivados da obtenção de recursos mediante a constituição de dívidas, amortização de empréstimos e financiamentos e/ou alienação de componentes do ativo permanente.

- Receitas intra orçamentarias correntes: São receitas correntes de órgãos, autarquias, fundações, empresas dependentes e de outras entidades integrantes dos orçamentos fiscal e da seguridade social, quando o fato que originar a receita decorrer de despesa de órgão, autarquia, fundação, empresa dependente ou de outra entidade constante desses orçamentos, no âmbito da mesma esfera de governo.

- Receitas intra Orçamentarias capital:Receitas de capital de órgãos, fundos, autarquias, fundações, empresas estatais dependentes e outras entidades integrantes dos orçamentos fiscal e da seguridade social derivadas da obtenção de recursos mediante a constituição dedívidas, amortização de

empréstimos e financiamentos ou alienação decomponentes do ativo permanente, quando o fato que originar a receita decorrer dedespesa de órgão, fundo, autarquia, fundação, empresa estatal dependente ou outraentidade constante desses orçamentos, no âmbito da mesma esfera de governo.

- Deduções Restituições receitas: São as deduções de restituições de receitas.
- Outros dados de entrada: completar.

As saídas desse programa são basicamente gráficos detalhados das receitas e das despesas do GDF. Os tipos de gráficos utilizados são conhecidos como gráficos de colunas com detalhamentos.

2 Desenvolvimento

Nessa seção desenvolvimento é apresentada a estrutura/arquitetura do Programa, o seu procedimento para a execução, os problemas técnicos enfrentados no desenvolvimento e as bibliotecas externas necessárias para a solução do problema.

2.1 Arquitetura

O programa possui três grandes módulos, um módulo em Python, um módulo em C e um módulo em html/JavaScript. Cada módulo possui um comportamento próprio e funções bem definidas. A Figura **??** apresenta uma visão da arquitetura do programa.

Cada um das funções de cada módulo são detalhadas abaixo:

- Gerenciamento de threads e manipulação de Strings:

  O programa está centrado em python. O módulo python cria e gerencia as threads que são utilizadas para realizar a extração de dados. Nesse ponto são implementados os conceitos de programação concorrente vistos em sala de aula, uma vez que a utilização de threads paralelisa a extração de dados tornando o processo de webscrapping mais rápido.

  Além de gerenciamento dos threads o módulo Python é responsável pela manipulação das variáveis extraídas em formato Strings, pois possui uma enorme quantidade de recursos para manipular Strings e Expressões Regulares.

  O módulo permite o armazenamento das variáveis em arquivos no formato CSV. Esse é o formato de arquivos interpretado pelos demais módulos.

- Integração entre Python e C:

  O Python gerencia a integração entre os dois módulos. Ele chama e utiliza as funções implementadas em C utilizando um recurso chamado Ctypes

[2]. Ctypes é uma biblioteca funcional externa para Python que prove compatibilidade entre os tipos de dados C/Python e permite a chamada de funções implementadas em C.

- Definição da estrutura de hash e manipulações matemáticas:

  No módulo C estão implementadas as estruturas de hashset da lista 4.

  São realizadas manipulações matemáticas nesse módulo.

- Camada de apresentação:

  Esse módulo do sistema é responsável pela apresentação ao usuário final das variáveis manipuladas. Ele foi desenvolvido utilizando o Twitter Bootstrap [3] que é uma coleção de ferramentas grátis para criação de websites e aplicações web. Ele permite uma melhor utilização dos recursos disponíveis em HTML e CSS, pois possui diversos templates e outros componentes de interface.

  O portal reune uma serie de informações referentes a documentação do sistema, ferramentas utilizadas e permite o download de todo o código do projeto, entretanto sua principal função é apresentar graficamente os dados obtidos. Os gráficos são gerados utilizando uma biblioteca JavaScript chamada HighCharts [4] . Os dados de entrada para criação dos gráficos são arquivos no formato CSV.

2.2 Execucao

Os passos juntamente com as bibliotecas necessárias para execução do programa são apresentados nessa seção. O software deve ser executado em sistema operacional linux e no browser google chromium. A IDE de desenvolvimento é o Netbeans 6.5, a versão do Python é a 2.7.2, para a documentação esta sendo usado o Doxygen 1.8.2 e para controle de versão e compartilhamento de código o Git distributed version control system 1.7.12.

O programa está centrado em Python e para que ele consiga controlar todos os módulos e realizar suas funções são necessárias algumas bibliotecas adicionais listadas abaixo. A instalação das bibliotecas pode ser facilitada utilizando easy install [5]

- ctypes;
- mechanize;
- cookielib;
- threading;

[2]Ctypes:`http://docs.python.org/library/ctypes.html`.
[3]Homepage Twitter Bootstrap:`http://twitter.github.com/bootstrap/`.
[4]Homepage HighCharts:`http://www.highcharts.com/`.
[5]Biblioteca easy install:`http://packages.python.org/distribute/easy_install.html`.

- Queue;
- errno;
- sys;
- time;
- ast;
- BeautifulSoup;
- os;
- re;

Todo o código produzido pode ser baixado no seguinte site: `http://kamillaaah.github.com/LucidScrapy/`. Esse é o site do projeto produzido automaticamente pelo github.

2.3 Problemas Técnicos

Os principais problemas enfrentados nessas lista foram na hora de manipulação dos threads e da criação dos gráficos com HighCharts. A seguir cada problema é detalhado:

- São criadas 6 threads que devem salvar 6 arquivos no formato CSV, porém apenas três arquivos são salvos apóso a execução do programa. Esse problema de gerenciamento das threads ainda não foi resolvido nessa lista.
- A criação de gráficos com HighCharts a partir dos arquivos no formato CSV foi bem complicada. Mesmo toda com toda a lógica correta os gráficos simplesmente não eram gerados. Após muita pesquisa descobriu-se que o JavaScript não estava conseguindo ler os arquivos no formato CSV no disco local. Esse é um dos padrões de segurança implementados pelos browsers. Para resolver esse problema descobriu-se a existencia de flags que podem ser utilizados no chromium. Essa flags dão ao browser funcionalidades especiais, pois habilitam ou desabilitam certos recursos padrões. Uma dessas flags é "–allow-file-access-from-files" que permite que os arquivos indicados a seguir consigam ler arquivos em diretórios locais. Desse modo essa flag resolveu o problema da geração dos graficos.

3 Conclusão

Essa lista resultou no desenvolvimento de grande parte do trabalho. Grande avanços foram alcançados nela como a integração entre os módulos C, Python e JavaScript. As principais dificuldades concentraram-se no gerenciamento de threads e na criação dos gráficos. Grande parte deles foi resolvido sobrando apenas alguns para serem concertados em entregas futuras.

Na próxima lista pretende-se realizar alguma manipulação estatística com esses dados utilizando pra isso Scheme e aprimorar o módulo de apresentação dos dados.