UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO

FACULTAD DE INGENIERÍA

EL USO DE WEB SCRAPING PARA LA EXTRACCIÓN DE DATOS

INFORME DE ACTIVIDADES PROFESIONALES

Que para obtener el título de

Ingeniera en computación

P R E S E N T A

Pineda Leal Viridiana

ASESORA DE INFORME

M.I. Norma Elva Chávez Rodríguez

Ciudad Universitaria, Cd. Mx., 2017

ÍNDICE

LISTA DE FIGURAS

INTRODUCCIÓN

Vivimos en una era digital donde las necesidades de las personas no se limitan por el lugar donde viven, esto da lugar a varias propuestas innovadoras. Cada día la tecnología se va abriendo paso para reemplazar tareas cotidianas, así como el trabajo monótono. Hoy en día, un ingeniero en computación es necesario para desarrollar soluciones que reemplace el trabajo monótono o que no puede ser pagado, ya sea porque no se encuentra el personal adecuado o porque la empresa no tiene los recursos necesarios para cubrir las necesidades.

La ingeniería en computación es una carrera con alta demanda que necesita actualizarse constantemente debido a que cada día, cada minuto, cada segundo, se descubren nuevas tecnologías que evolucionan sin límites.

¿Qué pasa si la empresa donde laboras necesita personal para una tarea específica pero no tiene los recursos para pagarle? La empresa debe buscar alternativas. Es aquí donde un ingeniero interviene con varias soluciones, una de ellas es emplear la tecnología.

Gracias a mi formación profesional pude auxiliar a la empresa con mis conocimientos en programación y el dominio de un software llamado: Screen-Scraper. También es importante mencionar que actualmente tenemos acceso a la información desde cualquier parte del mundo con tan sólo tener una computadora. El internet se ha vuelto fundamental en mi trabajo.

En este informe redacto mi trabajo como ingeniera en computación en la extracción de datos con WebScraping para el análisis de precios en el mercado, así como las dificultades que se me presentaron.

CAPÍTULO 1: LA EMPRESA

En este capítulo se describirá brevemente las características de la empresa en la cual se prestaron los servicios profesionales.

Descripción

Se laboró en una empresa familiar ubicada en Zapopan, Jalisco. Esta empresa se fundó en la Ciudad de México y, por oportunidades de negocio, cambió de domicilio a Jalisco. Actualmente, se encuentra legalmente constituida y registrada como una Sociedad de Responsabilidad Limitada de Capital Variable.

El representante legal interviene directamente en la administración y la dirige. También se encarga de elaborar proyectos relacionados con Ciencia de Datos y Tecnologías de la Información. El área de Recursos Humanos se encarga de la selección y reclutamiento del personal, además de colaborar en temas de administración, contabilidad y procesos de negocio. Ambos fundadores son socios capitalistas.

La toma de decisiones es compartida ya que ambos socios poseen la misma cantidad de acciones y se reúnen para acordar el rumbo de la empresa y de sus actividades.

Actualmente, la empresa cuenta con nueve empleados contratados por servicios profesionales, algunos laboran remotamente y hacen entrega de sus avances por medios electrónicos, mientras que otros asisten físicamente a las instalaciones.

Misión

Habilitar el talento joven para brindar a las MiPyME cultura organizacional y soluciones tecnológicas que contribuyan a prolongar su permanencia indefinidamente.

Visión

Ser la empresa de referencia en mejores prácticas y metodologías para contribuir a generar una sociedad en armonía a través de la transformación de las MiPyME en empresas enfocadas en la generación de valor, así como en el desarrollo de las personas en seres humanos plenos.

Valores

Responsabilidad, compromiso, diversidad, innovación, pasión, ética.

Objetivos organizacionales

- ✓ Habilitar a los jóvenes en cuanto a conocimientos tecnológicos y competencias para que puedan ofrecer valor a las empresas y a la sociedad.
- ✓ Otorgar a los empleados condiciones adecuadas y flexibles de trabajo, que les permita organizar su tiempo para actividades profesionales y personales.
- ✓ Generar un ambiente de trabajo colaborativo y en armonía.
- ✓ Desarrollar soluciones tecnológicas que satisfagan las necesidades específicas de las MiPyME y les abran la posibilidad de crecimiento y desarrollo.
- ✓ Desarrollar soluciones enfocadas a mejorar la productividad a nivel empresarial y personal.

- ✓ Dar a los socios la seguridad de trabajar en una empresa comprometida, honesta, ética y responsable.
- ✓ Ayudar a desarrollar soluciones que sean amigables con el planeta.
- ✓ Generar rendimientos atractivos para los accionistas, cuidando que se cumplan los valores de la empresa.

Historia

La empresa nace el 24 de febrero de 2016 como respuesta a la necesidad de dos emprendedores que deseaban otorgar valor a su sociedad a través de sus conocimientos tecnológicos y humanos.

En la Figura 1 se muestra el organigrama de la empresa

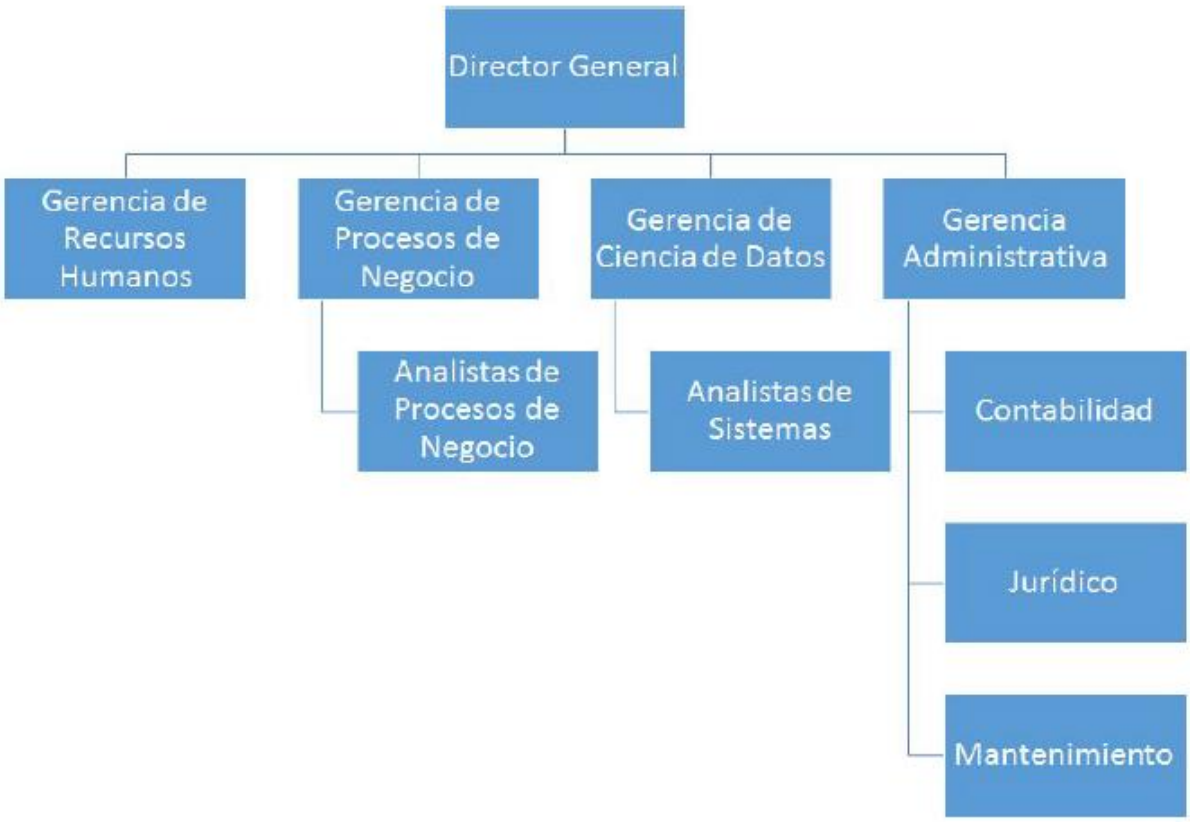

Figura 1: Organigrama de la empresa

CAPÍTULO 2: EL PUESTO DE TRABAJO

En este capítulo se describirá la experiencia profesional adquirida.

Descripción

El puesto que se desempeñó en la empresa es el de "Analista de sistemas" con el propósito de desarrollar programas que mejorarán los procesos cotidianos de la empresa, así como brindar apoyo en proyectos internos.

Este puesto es relevante dentro de la empresa recién fundada. El perfil es ser un profesional especializado del área de la informática, encargado del desarrollo de aplicaciones en lo que respecta a su diseño y obtención de los algoritmos, así como de analizar las posibles utilidades y modificaciones necesarias de los sistemas operativos para una mayor eficacia de un sistema informático, así como dar apoyo técnico a los usuarios de las aplicaciones existentes.

Los objetivos que se deben cumplir cumplir son:

- ✓ Desarrollo de programas ágiles
- ✓ Programas óptimos
- ✓ Resolución de problemas en el menor tiempo posible
- ✓ Proyectos finalizados con éxito

Trabajo por honorarios

Las principales diferencias entre ganar por honorarios y por nómina son las obligaciones fiscales: declaración mensual/anual, facturación, descuento de impuestos directo al sueldo y pagos de impuestos.

Cabe mencionar que trabajé como *freelance*, es decir, trabajador independiente. Esto fue un reto porque había que adaptarse a las necesidades de la empresa y tener el equipo necesario para cumplir las obligaciones como analista de sistemas.

Lo más difícil de trabajar así, no sólo es la distancia sino la comunicación limitada que se tuvo con el jefe, debido a que sólo se podía tratar personalmente con él una vez a la semana y posteriormente sólo por videollamadas en la noche.

CAPÍTULO 3: HERRAMIENTAS UTILIZADAS

En este capítulo se mencionarán las herramientas y tecnologías utilizadas para el desempeño laboral en la empresa.

SCRUM

Scrum es un proceso en el que se aplican de manera regular un conjunto de buenas prácticas para trabajar colaborativamente, en equipo, y obtener el mejor resultado posible de un proyecto.

Figura 2: Flujo de SCRUM

En Scrum se realizan entregas parciales y regulares del producto final, jerarquizadas por el beneficio que aportan al receptor del proyecto. Un proyecto se ejecuta en bloques temporales cortos y fijos. Cada iteración tiene que proporcionar un resultado completo, un incremento de producto final que sea susceptible de ser entregado con el mínimo esfuerzo al cliente cuando lo solicite.

CA Agile Central

Se denomina así a la plataforma empresarial específicamente concebida para ampliar las prácticas de desarrollo ágil. CA Agile Central está diseñada para ampliar las prácticas de desarrollo de la metodología Scrum. Le permite proporcionar un núcleo para que los equipos planifiquen el trabajo, lo clasifiquen según sus prioridades y realicen seguimientos de este a un ritmo sincronizado. Además, permite medir su productividad, su previsibilidad, su calidad y su capacidad de respuesta con estadísticas de rendimiento en tiempo real.

Mediante esta plataforma se registraban los avances de los proyectos asignados. Debido a que era la única persona en el desarrollo de los proyectos, se ocupó para que el jefe supervisara trabajo desarrollado.

Control de versiones

El control de versiones es un sistema que registra los cambios realizados sobre un archivo o conjunto de archivos a lo largo del tiempo, de modo que puedas recuperar versiones específicas más adelante.

Este sistema te permite revertir archivos a un estado anterior, revertir el proyecto entero a un estado anterior, comparar cambios a lo largo del tiempo, ver quién modificó por última vez y mucho más. Usar un VCS (Version Control System) también significa generalmente que se puede recuperar archivos fácilmente. Además, se obtiene todos estos beneficios a un costo muy bajo.

Beanshell

BeanShell es un lenguaje de scripting basado en la sintaxis de Java. Al igual que ocurre con Python o Perl podemos crear aplicaciones completas, puesto que BeanShell usa directamente una máquina virtual de Java y puede usar todas las librerías de Java disponibles.

Inspeccionador de elementos de Chrome.

Se trata de una consola donde nos muestra los elementos que hay en una página web, tales como imágenes, diseño en css y hasta los códigos fuentes que utiliza para desplegarse.

WebScraping

Es una técnica utilizada mediante programas de software para extraer información de sitios web. Usualmente, estos programas simulan la navegación de un humano en la World Wide Web ya sea utilizando el protocolo HTTP manualmente, o incrustando un navegador en una aplicación.

El web scraping está muy relacionado con el registro de la web, la cual indexa la información de la web utilizando un robot y es una técnica universal adoptada por la mayoría de los motores de búsqueda. Sin embargo, el web scraping se enfoca más en la transformación de datos sin estructura en la web (como el formato HTML) en datos estructurados que pueden ser almacenados y analizados en una base de datos central, en una hoja de cálculo o en alguna otra fuente de almacenamiento. El término web scraping también está relacionado con la automatización de tareas en la Web, la cual simula la navegación de un humano utilizando un software de computadora. Algunos de los usos del web scraping son: la comparación de precios en tiendas, la monitorización de datos relacionados con el clima de cierta región, la detección de cambios en sitios webs y la integración de datos en sitios webs. También es utilizado para obtener información relevante de un sitio a través de los rich snippets.

Screen-Scraper Enterprise Edition versión 6.0

Screen-Scraper es un software de propósito general que permite extraer datos de una página a través de un servidor proxy. Ha estado en continuo desarrollo desde el año 2002. Entre sus principales funciones destacan: copiar texto de una página web, introducir datos en formularios y enviarlos, iteración a través de buscadores, descarga de archivos, integración con lenguajes de programación o aplicaciones externas a través de API, entre otras.

Para poder manejar Screen-Scraper es necesario tener conocimientos de programación y de desarrollo web.

Screen-Scraper utiliza un servidor proxy que se encuentra entre el navegador y el servidor web. Una vez configurado, cada vez que haga clic en un enlace o envíe un formulario desde su navegador éstos son registrados por Screen-Scraper para después retransmitir la información al servidor web de destino. El servidor web podrá enviar respuestas correspondientes, Screen-Scraper registra estas respuestas y las pasa a su navegador. En el siguiente diagrama, se muestra la interacción de Screen-Scraper con el usuario y la red.

Figura 3: Intervención de Screen-Scraper entre la red y el usuario

Esta característica nos limita a sólo visitar páginas con ciertos protocolos debido a que el proxy sólo muestra páginas http.

Dentro de Screen-Scraper, se puede programar en varios lenguajes como: Java, JavaScript y Python. También es posible usar Screen-Scraper de manera externa, por medio de archivos como son bat o Shell, o lenguajes de programación como Ruby, Java, Python, etcétera, por medio de la API. Gracias a esto podemos solucionar las limitantes del software.

Gracias a que se puede invocar Screen-Scraper externamente desde un archivo bat, se pudo automatizar la descarga de los archivos todos los días a la media noche.

Scripts

Son archivos dentro de Screen-Scraper donde se programa código. Cada script puede ser llamadado en sesiones y archivos escrapeables. Este no necesita declaración de variables, pero sí indicarle en qué lenguaje se está interpretando.

Archivos escrapeables

Son archivos de Screen-Scraper encargados de la interacción con la página. Es ahí donde se mandan variables y se obtiene información. Éstos necesitan de una sesión o un script para ser llamados y ejecutados, en el caso de ser una sesión quien lo llama, debe tener un orden.

Envío de datos

Dentro del archivo Scrapeable, hay una sección donde se le indica el envío de datos a la página. En esta pestaña se muestra lo que se envía a la página. En el campo “key” muestra el nombre del parámetro que recibe la página, en “value” el valor, en “sequences” el orden en que se envía y en “type” el tipo de envío de los datos. Abajo se ilustra la zona donde el software hace el envío de parámetros.

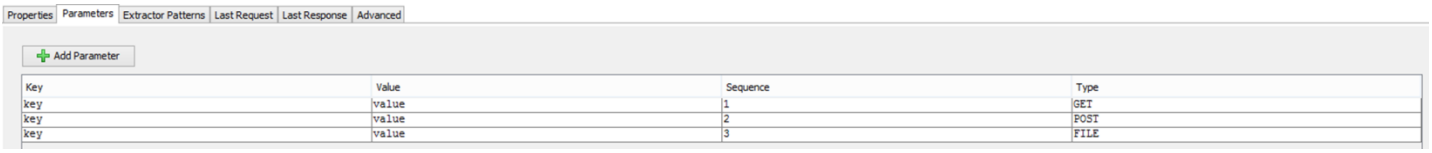

Figura 4: Pestaña de parámetros de un archivo Escrapeable

Envio y recibo de parámetros en Screen-Scraper

Screen-Scraper maneja una nomenclatura para indicar qué se guarda y qué se envía de la página con la que se está interactuando.

Para el envío de variables se utiliza ~##~, mientras que para el recibo es ~@@~. La siguiente imagen ilustra un ejemplo.

Figura 5: Ejemplo de envío y recibo de parámetros

No es necesario declarar las variables, si no existe, el software por defecto lo interpreta como “null”. Para el caso de recibo, crea la variable con su respectivo valor.

Patrones extractores

Es una herramienta de los archivos Escrapeables para obtener la información de la página una vez se haya obtenido el html. Se usa con expresiones regulares y se puede configurar el número de veces se ejecutará y en qué momento. Los patrones extractores pueden llamar scripts.

Sesiones

Es un archivo que unifica a todo el proceso de navegación, es decir, contiene todos los archivos escrapeables y scripts necesarios para explorar la página como si fuera un humano.

Variables

En Screen-Scraper se manejan 3 tipos de variables (Figura 6):

1. Variables de script/programación que existen mientras el script se ejecuta.
2. Variables de archivo scrapeable/datos que existen mientras el archivo scrapeable se ejecuta. Sólo los scripts que fueron llamados por el archivo scrapeable tienen acceso a esas variables.
3. Variables de sesión, que son el equivalente a las variables globales. Cualquier archivo dentro de la sesión puede acceder a ellas.

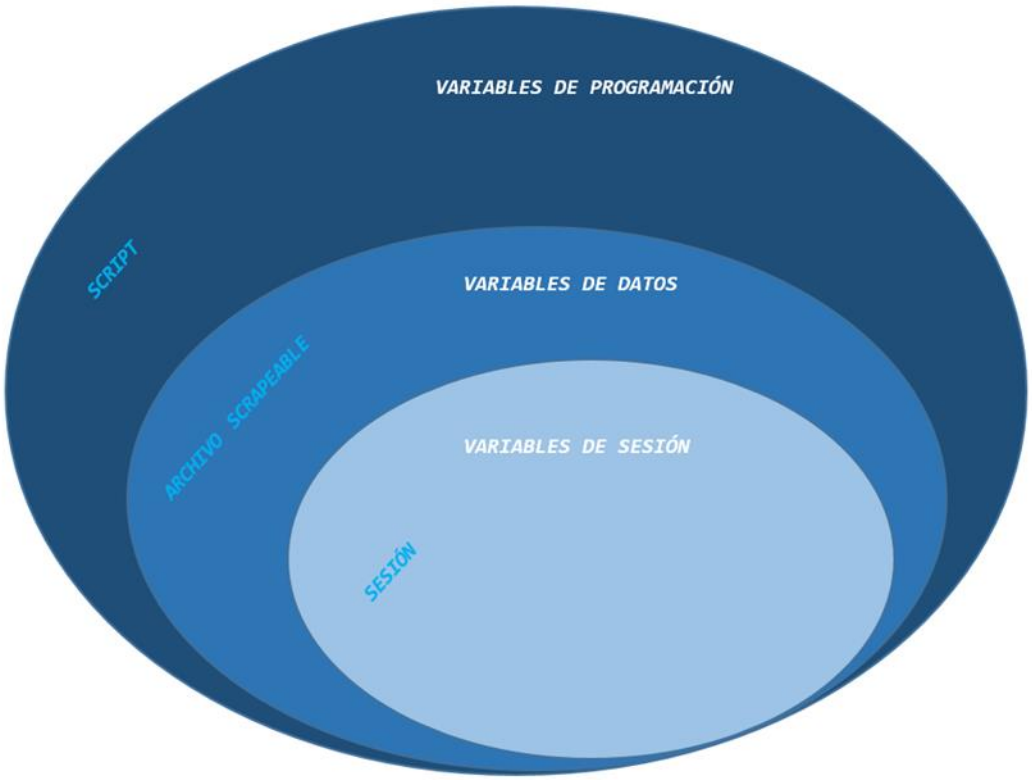

Figura 6: Variables y acceso a ellas en Screen-Scraper

Bibliotecas adicionales

Además de utilizar la API de Screen-Scraper (ver anexo API Screen-Scraper), agregué bibliotecas. Screen-Scraper lee todas las bibliotecas que tiene en su carpeta de instalación al iniciar. Gracias a esta caracteristica se puede extender la funcionalidad del software.

Agregué la biblioteca Javacsv para lectura y escritura de archivos csv porque Screen-Scraper no tiene una utilidad que me permitiera la lectura de un archivo csv, sólo había para escritura.

GIT

GIT es un software de control de versiones diseñado por Linus Torvalds, pensando en la eficiencia y la confiabilidad del mantenimiento de versiones de aplicaciones cuando éstas tienen un gran número de archivos de código fuente. Es compatible con Windows y Linux.

Con el fin de tener un control de versiones de cada cambio, así como tener los archivos en la nube para que los empleados tuvieran acceso a ellos, en el proyecto se utilizó GIT sincronizado con bitbucket.

Bitbucket es un servicio de alojamiento basado en web, para los proyectos que utilizan el sistema de control de revisiones Mercurial y Git.

Gracias a que GIT funciona en línea de comandos, se pudo automatizar la actualización de archivos a través de bats. Esto se programó para cuando Screen-Scraper terminara de bajar los archivos del día.

CAPÍTULO 4: PARTICIPACIÓN DEL ALUMNO EN LA EMPRESA

En este capítulo se explicará con detalle las actividades que se realizaron durante el tiempo que se laboró profesionalmente en la empresa.

La empresa decidió iniciar un proyecto que requeria de la extracción de datos para realizar análisis estadísticos de precios.

Este proyecto consistía en hacer estadísticas sobre el comportamiento de los precios en tiempos determinados. Para eso, se necesitaba extraer datos de productos, siendo los principales los de la canasta básica. De cada producto se necesitaba el precio, la marca, el contenido, la descripción, así como la fecha en que se obtuvo dicha información.

Sin embargo, la empresa se enfrentaba a un problema: al ser una empresa recién fundada, no contaba con los recursos suficientes para pagarle al personal que estuviera extrayendo los datos de diferentes páginas y pasarlos en un archivo Excel. Es en este punto donde apoyo con mis conocimientos en “web scraping” y herramientas que hacen uso de esta tecnología.

Utilicé un software llamado “Screen-Scraper” que usa web scraping con interpretación en java, java script y python. Este software fusiona dichos lenguajes de programación con beanshell, esto hace que la programación sea más sencilla debido a que no es necesario declarar variables, ni hacer métodos y objetos. Gracias a su compatibilidad con varios lenguajes de programación, fue posible mandarlo a ejecutar en ciertos días a ciertas horas con los parámetros necesarios.

La función de Screen-Scraper es simular que es un ser humano, para esto se estudió el código fuente de cada zona a la cual se necesitaba acceder, de tal manera que la seguridad de las páginas web no bloquearan al software por sus acciones debido a toman una petición repetitiva como una amenaza. Posteriormente, a través de los scripts, se indicaba a Screen-Scraper qué hacer, qué enviar (usuario/contraseña/fechas), de dónde leer los parámetros, a qué páginas acceder y en qué orden para llegar a la zona de interés; una vez dentro de la página deseada, se indicaron patrones extractores que le indicaban qué debe buscar y qué extraer; finalmente se le indicaba qué hacer con la información, limpiarla de basura (como etiquetas html) y pasarla a un archivo de texto, en este caso csv.

CAPÍTULO 5: DESARROLLO DEL PROYECTO

En este capítulo se explicarán las diversas etapas que implicó el desarrollo del proyecto.

Durante mi participación en la empresa se aprendió que un proyecto se realiza en dos etapas:

Desarrollo: La etapa de desarrollo es donde se construye la solución operacional del proyecto en curso, es decir la etapa donde se diseña, se programa, se prueban y modifican los componentes del proyecto.

Producción: Una vez listas las soluciones de la etapa de desarrollo, se hace paso a la producción. En esta etapa es lo que se va a quedar del proyecto y a lo que posteriormente se le dará mantenimiento en caso de necesitarse.

Estuve participando en este proyecto del 10 de diciembre de 2015 al 28 de marzo de 2016. A continuación detallo lo que realicé dentro del proyecto.

Análisis

La siguiente figura muestra el plan general del proyecto:

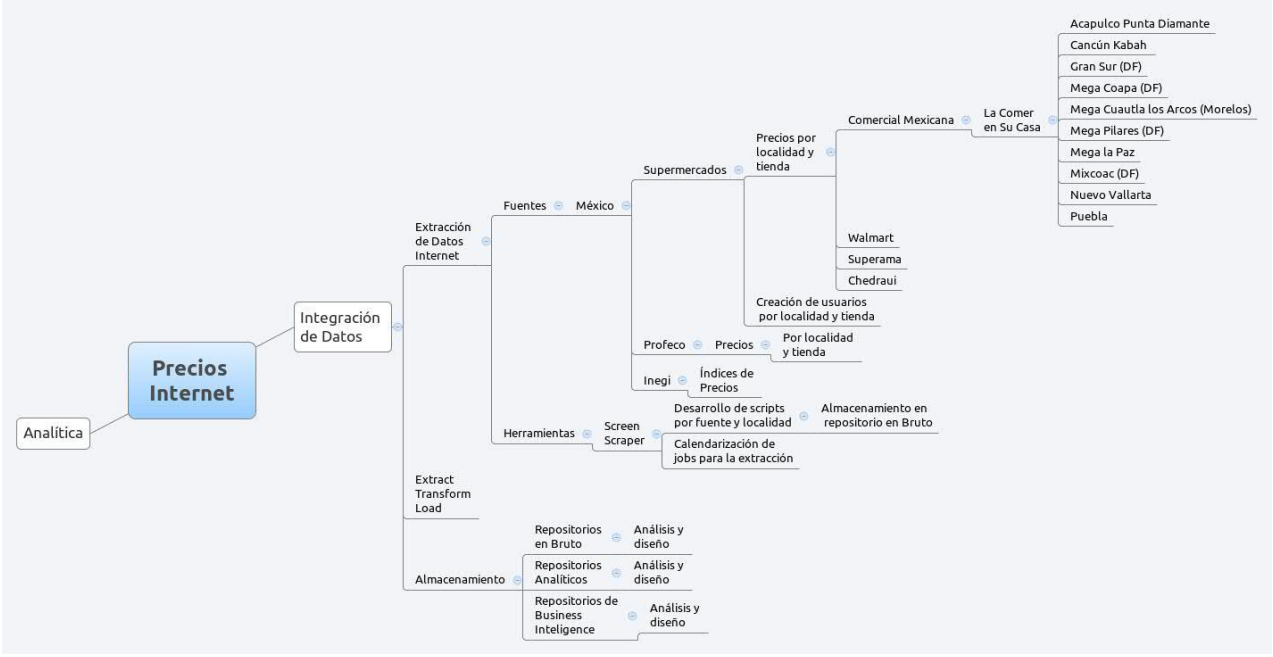

Figura 7: Plan general del proyecto

Como se puede ver en la figura, se pensó en extraer los precios de páginas de supermercados comerciales como son Walmart®, Comercial Mexicana®, SAMS club®, entre otras.

Se empezó a explorar cada una de las páginas para saber cómo estaba dividida la página y posteriormente analizarlo a nivel código con Screen-Scraper y el inspeccionador de elementos del navegador. Gracias a eso se pudieron descartar varias páginas debido a ciertas restricciones que a continuación menciono:

- Se necesita de una cuenta para visualizar contenido
- En ciertos casos, como SAMS club®, se requería una membresía
- Algunos servidores no eran estables y el tiempo de respuesta era muy largo o no se obtenía respuesta a ciertos departamentos.
- La página funcionaba con un protocolo https lo cual daba un problema de compatibilidad con Screen-Scraper.

Para que todos tuvieran acceso a los archivos, se usó GIT para actualizar en un repositorio en bitbucket. Para tener un orden los archivos se guardaban con la siguiente estrucuta:

```
AAAAMMDD_Pagina.csv
```

Donde AAAA es el año a cuatro dígitos, MM es el mes a dos dígitos y DD es el día a dos dígitos.

Comercial mexicana®

Estructura de la página

Al explorar la página, se pudo mostrar que mostraba productos por sucursales, departamentos y si estaban en existencia, esta página no necesitaba de un usuario y contraseña para mostrar su contenido en cuanto a productos.

Al analizarlo a fondo el código html, se descubrió que sólo manejaban 10 sucursales cada uno con su respectivo ID.

SUCURSAL	ID
GranSur	288
Mixcoac	16
Mega Coapa	174
Mega Pilares	5
Cancun Kabah	158
Mega Cuautla Los Arcos	337
Nuevo Vallarta	345
Acapulco Punta Diamante	329
Mega La Paz	378
Puebla	164

Tabla 1 Sucursales en la página de Comercial Mexicana®

Posteriormente la página desglosa sus departamentos.

DEPARTAMENTO	SUBDEPARTAMENTO
Abarrotes comestibles	Abarrotes orgánicos

Abarrotes comestibles	Aceites comestibles oliva y vinagre
Abarrotes comestibles	Alimentos enlatados
Abarrotes comestibles	Atunes y mariscos
Abarrotes comestibles	Azúcar y endulzantes
Abarrotes comestibles	Cereal
Abarrotes comestibles	Chocolate y saborizantes
Abarrotes comestibles	Consomes pure y caldillos
Abarrotes comestibles	Galletas y barras
Abarrotes comestibles	Gelatinas postres y frutas en almíbar
Abarrotes comestibles	Harina arroz y frijol
Abarrotes comestibles	Leche y leche en polvo
Abarrotes comestibles	Mayonesa mostaza y aderezos
Abarrotes comestibles	Miel cajeta mermelada
Abarrotes comestibles	Pan de caja tostadas y tortillas
Abarrotes comestibles	Productos para diabéticos
Abarrotes comestibles	Sal semillas especias y sazonadores
Abarrotes comestibles	Salsas chiles y moles
Abarrotes comestibles	Sopas y pastas
Abarrotes comestibles	TeCafes y sustitutos de crema
Alimentos preparados	Pintxos
Alimentos preparados	Platillos
Alimentos preparados	Rostizados
Bebidas	Aguas
Bebidas	Bebidas de sabor
Bebidas	Bebidas de soya
Bebidas	Cocteles sin alcohol
Bebidas	Deportivas y energéticas
Bebidas	Jugos y néctares
Bebidas	Refrescos
Bebidas	Saborizantes y concentrados
Bebidas	Accesorios
Bebidas	Alimento infantil
Bebidas	Higiene y perfumería
Bebidas	Pañales
Carne pescado y salchichoneria	Carne (Res)
Carne pescado y salchichoneria	Carne cerdo y aves orgánico
Carne pescado y salchichoneria	Carnes frías y embutidos
Carne pescado y salchichoneria	Cerdo
Carne pescado y salchichoneria	Organicos
Carne pescado y salchichoneria	Pescado y mariscos
Carne pescado y salchichoneria	Pollo

Carne pescado y salchichoneria	Preparados a granel
Desechables	Bolsas para basura
Desechables	Papel higiénico y facial
Desechables	Platos vasos y cubiertos desechables
Desechables	Protección de alimentos
Desechables	Servilletas y servitoallas
Dulces	Dulces típicos y a granel
Dulces	Dulces y chocolates
Dulces	Fruta seca
Farmacia	Alimento para bebé
Farmacia	Analgésicos y sueros
Farmacia	Antimicrobianos y sistema inmunologico
Farmacia	Aparato digestivo
Farmacia	Cardiovascular
Farmacia	Dermatológicos
Farmacia	Especialidad
Farmacia	FARMACOM
Farmacia	Higiene bucal
Farmacia	Hormonales
Farmacia	Material para curación
Farmacia	Oftálmicos
Farmacia	Planeación familiar e higiene íntima
Farmacia	Sistema nervioso
Farmacia	Sistema respiratorio
Farmacia	Sistema urologicos
Farmacia	Vitamínicos y complementos
Ferretería y jarcería	Accesorios de jarciería
Ferretería y jarcería	Carbon y velas
Ferretería y jarcería	Cintas y Pegamento
Ferretería y jarcería	Electricidad
Ferretería y jarcería	Escobas y cepillos
Ferretería y jarcería	Fibras esponjas y guantes de limpieza
Ferretería y jarcería	Focos
Ferretería y jarcería	Pilas
Ferretería y jarcería	Planchado
Ferretería y jarcería	Telas para limpieza
Ferretería y jarcería	Trapeadores y mops
Frutas y verduras	Empacados y preparados
Frutas y verduras	Frutas
Frutas y verduras	Frutas cítricas
Frutas y verduras	Frutas y verduras orgánicas

Frutas y verduras	Hojas tallos manojos hongos y hierbas
Frutas y verduras	Secos y semillas a granel
Frutas y verduras	Verduras
Higiene personal	Accesorios para pies y calzado
Higiene personal	Afeitado
Higiene personal	Cremas y cuidado facial
Higiene personal	Cuidado del cuerpo
Higiene personal	Cuidado e higiene oral
Higiene personal	Desodorantes y talcos
Higiene personal	Fijadores y modeladores
Higiene personal	Incontinencia
Higiene personal	Jabón de tocador
Higiene personal	Orgánicos
Higiene personal	Protección femenina
Higiene personal	Shampoo acondicionadores y tratamientos
Higiene personal	Tinte y tratamiento capilar
Limpieza	Aromatizantes
Limpieza	Detergente y suavizantes
Limpieza	Insecticidas y plaguicidas
Limpieza	Jabón blanqueadores y comp. lavandería
Limpieza	Lavatrastes
Limpieza	Limpiadores
Lácteos y congelados	Alimentos congelados
Lácteos y congelados	Bebidas lácteas
Lácteos y congelados	Cremas
Lácteos y congelados	Frutas y verduras congeladas
Lácteos y congelados	Hielo
Lácteos y congelados	Huevo
Lácteos y congelados	Lácteos orgánicos
Lácteos y congelados	Mantequilla y margarina
Lácteos y congelados	Postres nieves y helados
Lácteos y congelados	Yoghurt líquido
Lácteos y congelados	Yoghurt sólido
Mascotas	Alimento para mascotas
Mascotas	Limpieza y accesorios
Panadería y tortillería	Panadería
Panadería y tortillería	Pasteles y repostería
Panadería y tortillería	Tortillería
Quesos	Quesos importados
Quesos	Quesos nacionales
Vegetales en conserva	Enlatados

Vegetales en conserva	Vegetales en Conserva
Vinos licores y cigarros	Aguardientes y Mezcal
Vinos licores y cigarros	Brandy
Vinos licores y cigarros	Cervezas
Vinos licores y cigarros	Cognac
Vinos licores y cigarros	Coolers cocteles y rompopes
Vinos licores y cigarros	Cremas jerez y vinos generosos
Vinos licores y cigarros	Orgánicos
Vinos licores y cigarros	Ron
Vinos licores y cigarros	Tabacos
Vinos licores y cigarros	Tequila
Vinos licores y cigarros	Vinos y licores
Vinos licores y cigarros	Vodka y ginebra
Vinos licores y cigarros	Whisky
Botanas	
Gourmet	

Tabla 2 Departamentos y subdepartamentos de Comercial Mexicana®

Finalmente la página muestra los productos y los va actualizando conforme uno va bajando el scroll del mouse.

Desarrollo de solución

Una vez entendida la estructura de la página, se usó Screen-Scraper para crear una sesión que bajara los precios en un archivo de texto.

Se empezó por crear los archivos escrapeables volviendo a hacer la interacción con la página pero esta vez usé Screen-Scraper como proxy para guardar todas las transacciones. Se eligieron las transacciones que son necesarias como son las que envían y reciben los datos de las sucursales, departamentos, subdepartamentos y productos. Cabe mencionar que el software captura hasta las imágenes, por eso es necesario hacer la depuración.

La falta de conocimiento en las funciones y herramientas de Screen-Scraper llevó a hacer algo poco óptimo para la lectura de todos los ID: se creó un archivo de texto separado por comas para, posteriormente con programación, indicarle a Screen-Scraper lo que debía mandar a través de sus archivos escrapeables.

¿Por qué poco óptimo? Porque si las sucursales cambiaban o se añadían más, debía hacerse otra programación para actualizar el txt o en el peor de los casos: actualizarlo manualmente. También se corría el riesgo de que cometiera algún error al escribir el txt manualmente o que se guardara en una codificación incompatible por Screen-Scraper.

Esta fue la estructura del archivo de texto:

```
sucursales=288:GranSur;16:Mixcoac;174:MegaCoapa;5:MegaPilares;158:CancunKabah;337:Mega
CuautlaLosArcos;345:NuevoVallarta;329:AcapulcoPuntaDiamante;378:MegaLaPaz;164:Puebla
departamentos=1:6,8,154,2,152,11,156,157,10,153,7,5,9,151,3,155,4;13:17,14,18,20,16,19
,15;21:30,27,25,29,31,26,32,24,28,23,22;33:35,34;36:38,41,42,39,44,37,40,43;45:46,48,4
7;51:158,159;53:54,56,55;57:77,69,75,73,60,64,62,74,65,58,67,63,70,59,66,76,71,72,68,6
1;78:85,81,86,80,82,84,79,83;87:89,92,91,90,88;93:98,97,99,96,95,94;100:12,105,111,104
,107,110,109,102,101,112,106,108,103;113:123,118,116,119,117,125,126,121,114,122,115,1
24,120;127:129,128;130:132,134,131,133;135:145,146,139,138,140,143,137,136,144,142,141
;147:150,148,149
depEspeciales=50;52
```

Screen-Scraper lee los archivos de texto para cargar variables de la siguiente forma:

```
Variable=valor
Variable1=valor1
```

En mi archivo le indico 3 variables: “sucursales” para el id y el nombre de las sucursales; “departamentos” para el id de los departamentos y el id de los subdepartamentos; finalmente “depEspeciales” para el id de los departamentos que no tienen subdepartamentos.

A continuación se explica la construcción de cada variable. Se usó “;“para separar los conjuntos de valores.

sucursales: se indicó que el primer valor antes del signo “:” es el id de la sucursal, lo que sigue después es el nombre.

departamentos: el primer valor es el id del departamento, después de los “:” viene el conjunto de subdepartamentos asociados al valor separados por comas.

depEspeciales: sólo separé los únicos valores que tomará.

A través de un script, le indiqué al software la lectura y separación de los valores. Este es el fragmento del código:

```
session.loadVariables(session.getv("ruta"));
session.logVariables();

ConjuntoSucursales= session.getVariable("sucursales");
ConjuntoDepartamentos= session.getVariable("departamentos");
ConjuntoEspeciales= session.getVariable("depEspeciales");

String[] Sucursales = ConjuntoSucursales.split(";");
String[] Departamento = ConjuntoDepartamentos.split(";");
String[] Especial = ConjuntoEspeciales.split(";");
```

Una vez cargadas las variables al software, a través de programación a Screen-Scraper se le indicó a dónde y en qué orden enviar las variables para obtener la información, así como qué hacer con ella.

Para la parte de guardado de información, se utilizaron los patrones extractores del software para indicarle a través de expresiones regulares qué información me interesa extraer. Posteriormente se le

indiqué que escribiera un archivo csv para plasmar el contenido. En la Figura 8 se muestra el archivo creado.

marca	producto	contenido	precio
ABBOTT	Isomil 1	900grs	$458.35
ABBOTT	Similac 11 85	1pza	$351.60
ABBOTT	Similac total	1pza	$364.75
ABBOTT	Similac total	1pza	$356.85
BAYER	Novamil ac 2	400grs	$189.50
BAYER	Novamil ae 1	400grs	$188.50
BAYER	Novamil ae 2	400grs	$188.50
BAYER	Novamil s 2	400grs	$168.00
BAYER	Novamil ac 1	400grs	$189.50
BAYER	Novamil 2	400grs	$133.15
BAYER	Formula lact	400grs	$130.90
DANONE BAI	Aptamil a.r. I	1pza	$187.00
DANONE BAI	Aptamil com	1pza	$209.10
DANONE BAI	Aptamil pep	1pza	$310.10
DANONE BAI	Aptamil prer	1pza	$310.00
ABBOTT	Pedialyte 30	500ml.	$23.00
ABBOTT	Pedialyte 30	500ml.	$23.00
ABBOTT	Pedialyte 30	500ml.	$23.00
ABBOTT	Pedialyte 60	500ml.	$23.00
ABBOTT	Pedialyte 45	500ml.	$23.00
ABBOTT	Pedialyte 30	500ml.	$23.00
ABBOTT	Pedialyte 60	500ml.	$23.00
ALCON	Nevanac 1m	1pza	$404.10
ALLERGAN E	Blefamide sf	10ml.	$393.50

Figura 8: Archivo de precios de Comercial Mexicana®

Resultados

Se ejecutó esta sesión unos días, porque el tiempo de extracción era de aproximadamente 48 horas para un total de 51680 datos. Gracias a la versión de Screen-Scraper se logró ejecutar sesiones paralelas para descargar el contenido de todas las sucursales al mismo tiempo y sin embargo el tiempo de ejecución seguía siendo excesiva (poco más de 18 horas). Esto es debido al tiempo de respuesta de los servidores de Comercial Mexicana®. Esta página fue descartada tiempo después para el proyecto.

Walmart®

Estructura de página

En esta página no se necesita un usuario para ver los precios. La página está dividida en departamentos y subdepartamentos.

Al analizar el código fuente, fue sorprenderte ver que no había id asociado a un departamento.

DEPARTAMENTO
Despensa
Lacteos-y-Huevo
Bebidas
Farmacia
Bebes
Carnes-y-Pescados
Limpieza-y-Mascotas
Frutas-y-Verduras
Higiene-y-Belleza
Congelados
Salchichoneria
Vinos-y-Licores
Panaderia-y-tortilleria

Tabla 3 Departamentos en Walmart®

Para los productos se encontró que manejaban una Unidad de Pago por Capitación (UPC), pero no había un id asociado. Más adelante se descubrió de que no se necesitaba el id de cada producto como en las demás páginas.

Desarrollo de solución

Esta página fue de las más sencillas, debido a que sólo se le mandaba el nombre del departamento como se extrajo del código fuente y te daba todos los productos. No se necesitaba de nada adicional. En la Figura 9 se muestra el archivo generado.

upc	marca	producto	descripcion	precio
8.585E+11	Maggi	MAGGI JUGC	hojas sazona	$15.00
8.585E+11	Maggi	MAGGI JUGC	hojas sazona	$15.00
8.585E+11	Maggi	MAGGI JUGC	hojas sazona	$15.00
8.41007E+11	Gallo	GALLO TULIP	tulipanes gal	$40.00
8.41007E+11	Gallo	GALLO PASTA	pajaritas gall	$40.00
8.41007E+11	Gallo	GALLO MARG	margaritas ga	$40.00
8.41007E+11	Gallo	GALLO PLUM	plumas gallo	$40.00
8.07681E+11	Barilla	BARILLA PAS	fettuccine ba	$15.90
8.07681E+11	Barilla	BARILLA PAS	spaghetti rig	$15.90
8.07681E+11	Barilla	BARILLA PAS	spaghetti ba	$15.90
8.07681E+11	Barilla	BARILLA PAS	spaghetti cap	$15.90
8.07681E+11	Barilla	BARILLA PAS	tortiglioni ba	$15.90
8.07681E+11	Barilla	BARILLA PAS	sopa de codo	$15.90
8.07681E+11	Barilla	BARILLA PAS	penne rigate	$15.90
8.07681E+11	Barilla	BARILLA PAS	fusilli barilla	$15.90
8.07681E+11	Barilla	BARILLA PAS	spaghetti ba	$15.90
8.07681E+11	Barilla	BARILLA SPA	spaghetti ba	$15.90
8.07681E+11	Barilla	BARILLA PAS	sopa de mo?	$7.10
8.07681E+11	Barilla	BARILLA PAS	sopa de engr	$7.10
8.07681E+11	Barilla	BARILLA PAS	sopa de letra	$7.10
8.07681E+11	Barilla	BARILLA PAS	sopa de codo	$7.10
8.07681E+11	Barilla	BARILLA PAS	spaghetti ba	$7.10

Figura 9: Archivo de precios de Walmart®

Resultados

Para esta página hubo un tiempo de extracción de aproxiamademente 1 hora para un total aproximado de 18318 productos.

PROFECO

Después de tener varios archivos de productos de diferentes lugares, nos cuestionamos si todas tendrían los mismos productos, además de que estaríamos omitiendo productos de mercados y locales que no tienen página de internet donde exhiban sus productos. La herramienta de PROFECO "QUIEN ES QUIEN EN LOS PRECIOS" fue un complemento a las demás páginas.

Las ventajas de esta herramienta es que los precios se actualizaban diario.

Estructura de la página.

Al explorar la página, se te daba la opción de entrar como usuario registrado o visitante. Esta herramienta contiene todos los estados y municipios de México. Sus productos estaban divididos en varias categorías y subcategorías.

Al analizar su código fuente encontré que manejaba un id para cada estado y se concatenaba con el id de un municipio, también valores de un botón que está fuertemente relacionado con cada estado y que por desgracia no se encuentra en el código fuente, es algo que Screen-Scraper detectó una sola vez durante la sesión proxi. La tabla que se muestra a continuación la hice manualmente en un csv para que Screen-Scraper lo leyera.

ID_CIUDAD	CIUDAD	IMAGEBUTTON1.X	IMAGEBUTTON1.Y
121201	Acapulco	36	15
262603	Agua Prieta	50	15
010101	Aguascalientes	54	7
292902	Apizaco	53	13
040401	Campeche	45	6
232301	Cancún	32	9
040402	Cd. del Carmen	34	12
232302	Chetumal	32	2
080801	Chihuahua	27	12
150901	Ciudad de México y área metropolitana	53	8
080802	Ciudad Juárez	40	10
060601	Colima	46	11
232304	Cozumel	54	7
171701	Cuernavaca	21	8
252501	Culiacán	26	14
101001	Durango	44	10
20203	Ensenada	31	13
141401	Guadalajara	62	11
252502	Guasave	52	7
262601	Hermosillo	32	15
272702	Heroica Cárdenas	39	11

030302	La Cabos	34	8
030301	La Paz	56	4
111101	León	19	15
282801	Matamoros	47	3
313101	Mérida	50	11
020202	Mexicali	48	9
050502	Monclova	56	6
191901	Monterrey	28	13
161601	Morelia	52	5
262602	Nogales	38	10
282802	Nuevo Laredo	33	3
202001	Oaxaca	36	5
080803	Ojinaga	54	0
131301	Pachuca	56	14
050503	Piedras Negras	42	5
232303	Playa del Carmen	24	10
212101	Puebla	45	9
222201	Querétaro	50	6
282803	Reynosa	28	12
050501	Saltillo	35	8
242401	San Luis Potosí	66	9
262607	San Luis Río Colorado	49	5
070702	Tapachula	44	7
181801	Tepic	62	12
020201	Tijuana	32	9
292901	Tlaxcala	31	10
151501	Toluca	44	14
050504	Torreón y Gómez Palacio	38	5
070701	Tuxtla Gutiérrez	29	14
303001	Veracruz	40	4
272701	Villahermosa	55	8
323201	Zacatecas	38	6

Tabla 4 Estados en PROFECO

Mientras tanto en los productos, las categorías y subcategorías no tenían un id, sólo el producto, eso facilitó mucho las cosas, pues ya no tenía que navegar hasta el producto.

ID_PRODUCTO	PRODUCTO
0166	ACEITE
0864	ELOTE
0931	FRIJOLES

0971	MAYONESA
0972	MOLE ROJO EN PASTA
0977	SALSA PICANTE
0137	SARDINA
0965	AGUA SIN GAS
3061	DICCIONARIO
1241	HISTIACIL NF
1725	OMEPRAZOL

Tabla 5 Algunos productos de PROFECO

Desarrollo de solución

Al hacerlo con Screen-Scraper se generaba muchos archivos, la página usaba AJAX. Esta sesión costó mucho trabajo debido a que, si no se enviaba los parámetros que me generaba AJAX, la sesión se perdía y redireccionaba a la página de inicio. Se hicieron muchas pruebas, pues no se podía descartar todos los archivos, en alguno de ellos se perdían los parámetros y la página redireccionaba. Investigando la API de Screen-Scraper se encontró una función donde pude obtener los valores de AJAX y enviarlos en cada página que visitaba.

Una vez encontrado la secuencia que daba los resultados deseados, se empezó a desarrollar la solución y ahí se descubrió que los productos cambiaban de acuerdo a la época del año, por ejemplo en diciembre se agregaban árboles de navidad y en enero lo quitaban. Así se decidió diario actualizar la lista de productos disponibles. Para eso se hizo una sesión de Screen-Scraper donde extraía todos los productos y los escribía en un csv para posteriormente leerlos en otra sesión que se dedicara a extraer los precios de los productos.

```
Archivo="Productos.csv";
CsvWriter file = new CsvWriter(Ruta+Archivo);
String[] header = {"ID_PRODUCTO", "PRODUCTO"};
file.setHeader(header);
HashMap hm = new HashMap();
hm.put("ID_PRODUCTO", dataRecord.get("ID_PRODUCTO"));
hm.put("PRODUCTO", dataRecord.get("PRODUCTO"));
file.write(hm);
file.flush();
file.close();
```

Para poder hacer la sesión que bajaría los precios, debía leer los csv de Ciudades y Productos. Este es el fragmento de código que lee un csv:

```
String[] parseCSVLine(String line, int index, int columnsToGet){
    int START_STATE = 0;
    int FIRST_QUOTE = 1;
    int SECOND_QUOTE = 2;
    int IN_WORD = 3;
    int IN_WORD_WITHOUT_QUOTES = 4;
```

```
int state = START_STATE;
String word = "";
ArrayList lines = new ArrayList();
char[] chars = line.toCharArray();

 for (int i = 0; i &lt; chars.length; i++){
    char c = chars[i];

    if (c == '"'){
        if (state == START_STATE){
            state = FIRST_QUOTE;
        }
        else if ((state == FIRST_QUOTE) || (state == IN_WORD)){
            state = SECOND_QUOTE;
        }
        else if (state == SECOND_QUOTE){
            word += ("" + c);
            state = IN_WORD;
        }
    }
    else if (c == ','){
        if ((state == SECOND_QUOTE) || (state == IN_WORD_WITHOUT_QUOTES)){
            state = START_STATE;

            lines.add(word);
            if (lines.size() == columnsToGet) break;
                word = "";
        }
        else if (state == START_STATE){
            state = START_STATE;
            lines.add(word.replaceAll("\"\"", "\""));
        }
        else{
            word += ("" + c);
            state = IN_WORD;
        }
    }
    else{
        if (state == START_STATE) state = IN_WORD_WITHOUT_QUOTES;
        else if (state != IN_WORD_WITHOUT_QUOTES){
             state = IN_WORD;
            word += ("" + c);
        }
    }
}
if (lines.size() &lt; columnsToGet){
    if ((state == SECOND_QUOTE) || (state == IN_WORD_WITHOUT_QUOTES))
         lines.add(word.replaceAll("\"\"", "\""));
}
String[] linesArray = new String[lines.size()];

for (int i = 0; i &lt; lines.size(); i++){
    linesArray[i] = (String) lines.get(i);
```

```
    }

    return linesArray;
}
```

Durante el análisis se halló que si uno elegía “todos los municipios” el id correspondiente a municipio tenía un 0 al final, así que se concatenó un 0 al final de cada id de ciudad al mandarlo a la página.

```
 session.setVariable("ID_MUNICIPIO", idC+"0");
```

id producto	producto	id marca	marca	descripcion	precio	establecimiento	fecha de observació	ciudad
166	ACEITE	166058	123	BOTELLA 1 L	$17.90	CHEDRAUI SUCURSA	18/03/2016	Acapulco
166	ACEITE	166058	123	BOTELLA 1 L	$18.50	WALMART SUCURSA	17/03/2016	Acapulco
166	ACEITE	166058	123	BOTELLA 1 L	$20.25	SORIANA HIPER SUC	22/03/2016	Acapulco
166	ACEITE	166058	123	BOTELLA 1 L	$22.80	FARMACIA GUADAL	18/03/2016	Acapulco
166	ACEITE	166058	123	BOTELLA 1 L	$23.37	MEGA COMERCIAL M	16/03/2016	Acapulco
166	ACEITE	166058	123	BOTELLA 1 L	$23.50	EXTRA SUCURSAL 5	22/03/2016	Acapulco
9000	ACEITE DE OL	9000026	BORGES	BOTELLA 500	$64.90	MEGA COMERCIAL M	16/03/2016	Acapulco
9000	ACEITE DE OL	9000026	BORGES	BOTELLA 500	$75.00	WALMART SUCURSA	17/03/2016	Acapulco
9002	ALCAPARRA	9002003	EL SERPIS	FRASCO 100	$30.01	MEGA COMERCIAL M	16/03/2016	Acapulco
170	GRASA COMI	170002	INCA. CLASI	PAQUETE 1 K	$29.90	SORIANA HIPER SUC	22/03/2016	Acapulco
170	GRASA COMI	170002	INCA. CLASI	PAQUETE 1 K	$33.00	MEGA COMERCIAL M	16/03/2016	Acapulco
170	GRASA COMI	170002	INCA. CLASI	PAQUETE 1 K	$37.50	CHEDRAUI SUCURSA	18/03/2016	Acapulco
170	GRASA COMI	170002	INCA. CLASI	PAQUETE 1 K	$39.10	WALMART SUCURSA	17/03/2016	Acapulco
172	MANTECA DE	172002		PAQUETE 50	$22.00	MEGA COMERCIAL M	16/03/2016	Acapulco
172	MANTECA DE	172002		PAQUETE 50	$22.90	SORIANA HIPER SUC	22/03/2016	Acapulco
172	MANTECA DE	172002		PAQUETE 50	$28.00	WALMART SUCURSA	17/03/2016	Acapulco
147	MANTEQUILL	147002	CHIPILO	BARRA 225 G	$26.00	SORIANA HIPER SUC	22/03/2016	Acapulco
147	MANTEQUILL	147002	CHIPILO	BARRA 225 G	$30.40	MEGA COMERCIAL M	16/03/2016	Acapulco
171	MARGARINA	171027	I CANT BELIE	BOTE 454 GR	$54.00	WALMART SUCURSA	17/03/2016	Acapulco
171	MARGARINA	171027	I CANT BELIE	BOTE 454 GR	$54.00	CHEDRAUI SUCURSA	18/03/2016	Acapulco
171	MARGARINA	171027	I CANT BELIE	BOTE 454 GR	$59.20	SORIANA HIPER SUC	22/03/2016	Acapulco
171	MARGARINA	171027	I CANT BELIE	BOTE 454 GR	$59.50	MEGA COMERCIAL M	16/03/2016	Acapulco

Figura 10: Archivo de precios de PROFECO

Resultados

La pagina de PROFECO resulto ser una de las más completas en cuanto a análisis de precios.Los servidores de PROFECO respondían eficientemente, los primeros días se tardó 8 horas para bajar un total aproximado de 77537 productos. Después comenzó a bajar el tiempo hasta mantenerse en aproximadamente 5 horas.

Como se puede ver en la Figura 10, PROFECO tiene información de las páginas anteriormente mencionadas. Así fue como se decidió sólo bajar precios de esta página, que para nosotros fue la más completa.

Automatización

Una vez hechas las sesiones, se necesitaban ejecutar todos los días para obtener nuevos datos. Se determinó que es conveniente bajarlos durante la madrugada para no saturar los servidores de cada página durante el día.

También se requería que al finalizar la descarga de precios, se actualizara en el repositorio de bitbucket para que los encargados del análisis los tuvieran listos.

Sin embargo era muy tedioso estar hasta tarde o despertar durante la madrugada para lograr ese fin. Se pensó en una solución para automatizar diario la descarga y actualización de archivos.

El plan fue el siguiente: que se encendiera la computadora sola a las 12:00 de la noche, iniciar sesión automáticamente, abrir Screen-Scraper para ejecutar las sesiones y al finalizar actualizar todos los archivos en el repositorio. También se debía decidir en qué momento apagarse, debía hacer un cálculo para que no se apagara durante el proceso o para que no estuviera mucho tiempo prendida (ahorro de enrgía eléctrica).

Se investigó cómo lograr cada acción para aplicarlo y que funcionara como se deseaba.

Encendido

Para que la computadora se encendiera automáticamente, se encontró que es posible desde la BIOS.

La computadora era una HP que se característiza por tener una interacción con el mouse, esto facilitó seleccionar los días que debía prenderse la computadora automáticamente. En esa pantalla se programó para que de lunes a viernes se encendiera a las 12:00 pm. Se logró el resultado deseado.

Inicio de sesión

Se estaba trabajando bajo un Windows 7 con 2 usuarios. Se logró decirle al sistema operativo en qué cuenta iniciara sesión automáticamente. Estos son los pasos:

1. Volver administrador la cuenta en la que se inicie sesión.
2. Pulsar Windows + R.
3. Ejecutar “netplwiz”.

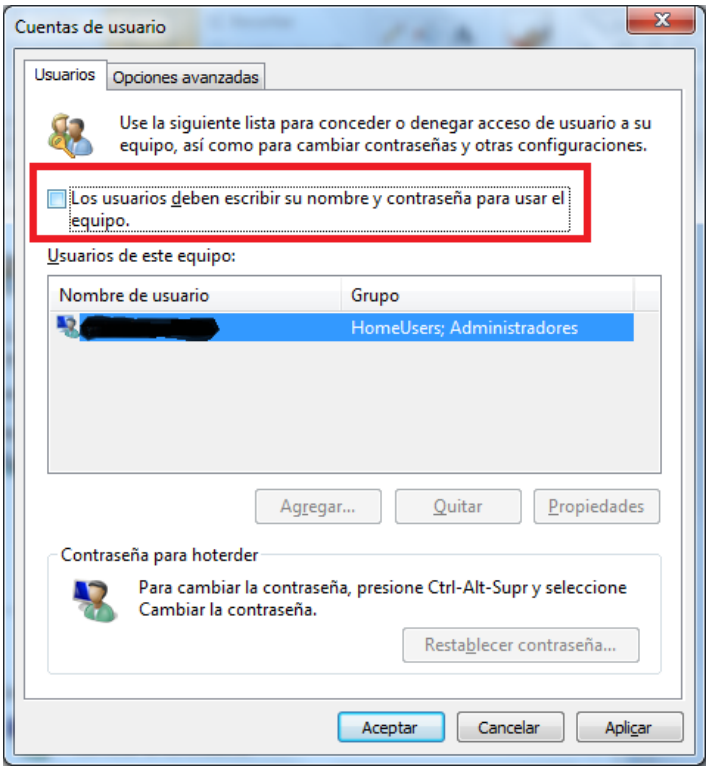

Figura 11: Cambiar opciones de usuario

4. En la ventana seleccionar el usuario y desmarcar la opción que dice "los usuarios deben colocar nombre de usuario y contraseña".
5. Aceptar y reiniciar.

Ejecución de tareas

Se Usó el "Programador de tareas" de Windows. Investigando se encontró que no se necesita tener abierto Screen-Scraper para ejecutar una sesión, se pueden mandar a llamar desde cualquier programa externo. Para el caso de programación en Java, se necesitaba exportar las bibliotecas necesarias que se podían encontrar en la carpeta de instalación de Screen-Scraper, pero se utilizaron bats para no estar recompilando con cada cambio nuevo.

Para ejecutar una sesión en una línea de comandos se sigue la estructura:

```
"Ruta de los archivos de java de Screen-Scraper" -jar screen-scraper.jar -s "nombre de la sesión" -p "ruta=ruta del txt que contiene los parametros"
```

Se decicidió que se debía imprimir, en un archivo de texto, todo lo que pasaba en consola en caso de error. Se le agregó ">" para que la orden se plasmara en un archivo de texto:

```
"Ruta de los archivos de java de Screen-Scraper" -jar screen-scraper.jar -s "nombre de la sesión" -p "ruta=ruta del txt que contiene los parametros">ruta y nombre del archivo log.log
```

Se hicieron bats adicionales para mover los archivos a la carpeta de GIT y actualizarlos en el repositorio. Todos los bats se encuentran en la sección "Códigos de bats" de los ANEXOS.

Una vez teniendo todos los bats, se usó el programador de tareas de Windows, se ajustó que se ejecutara al iniciar sesión y en la sección "Acciones" se agregaron los bats por orden de ejecución, así como la orden para apagar el equipo al finalizar todo.

En la siguiente figura se muestra cómo quedó programadas las tareas:

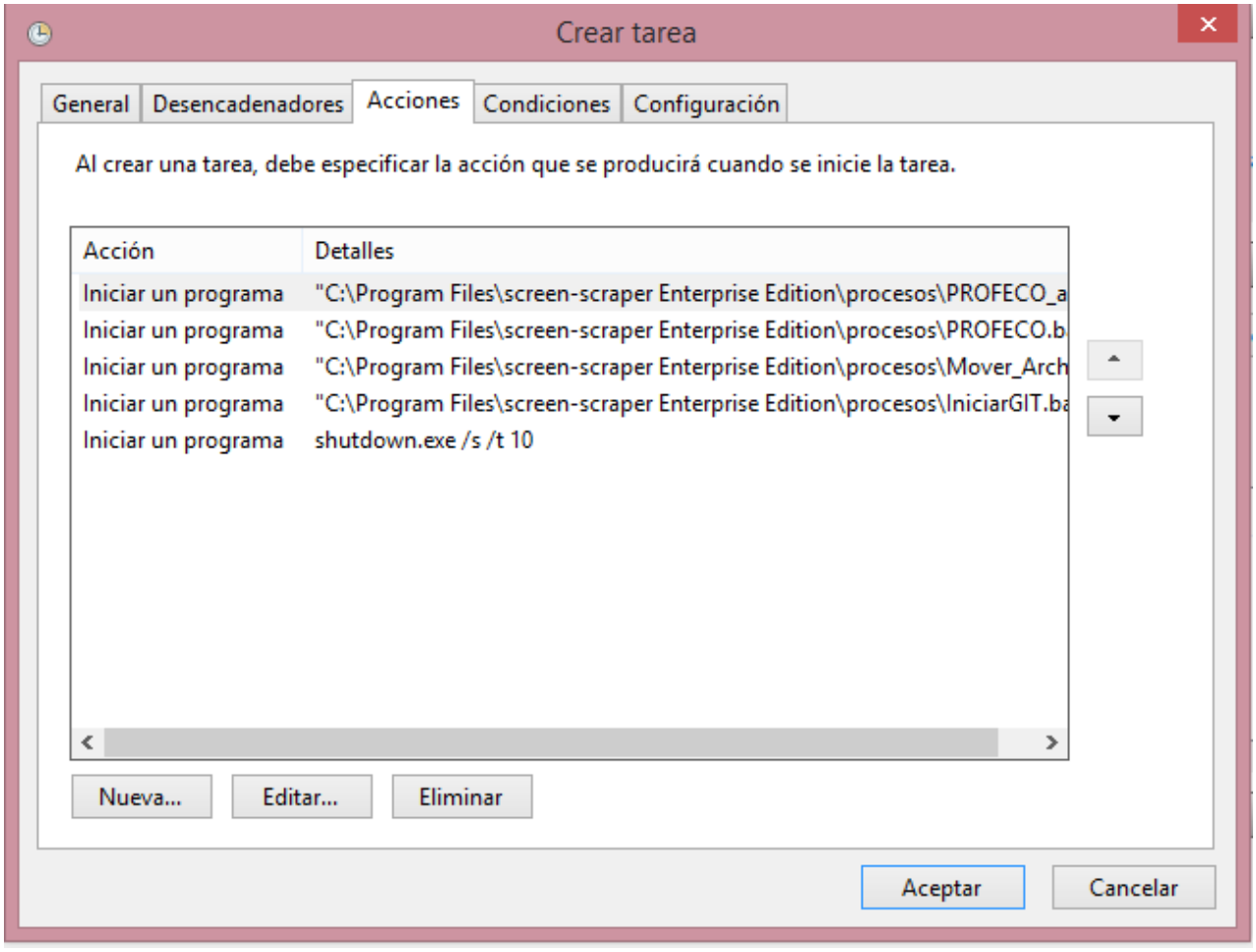

Figura 12: Orden de ejecución de bats en tarea programada

Máquina virtual

Hubo un periodo de tiempo en que se usó una maquina virtual, la cual también tenia Windows 7.

Mientras se tuvo el software en la máquina virtual, se estuvo buscando cómo programar Virtual Box para que iniciara la máquina específica. No se halló solución en internet sin embargo se encontró. Al abrir las propiedades del acceso directo, se visualizan los comandos que se ejecutaban para iniciar la máquina deseada (Figura 13).

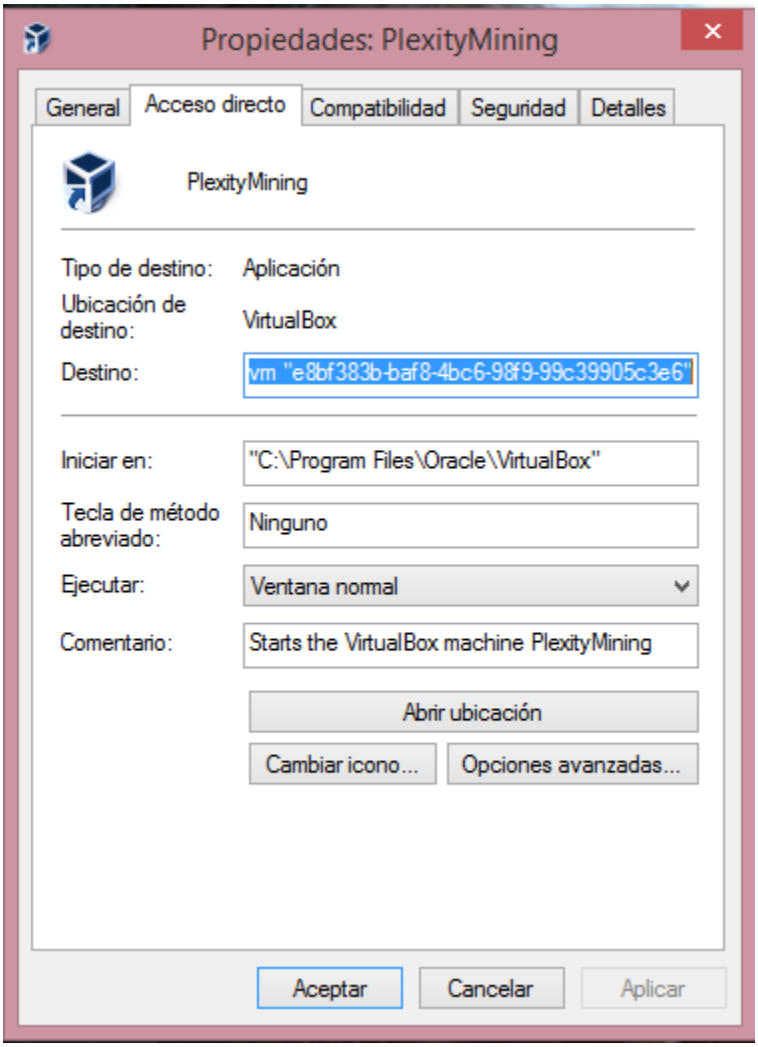

Figura 13: Propiedades de acceso directo a máquina virtual

Se copiaron las propiedades y se pegaron en una tarea programada de Windows, configurada para que se ejecute al iniciar sesión.

CONCLUSIONES

La mayoría de los estudiantes y recién egresados no se sienten preparados para el mundo labroal, sin embargo nuestra formación en la carrera y la facultad nos preparó lo suficiente para enfrentarlo. Tal vez no se ensñearon todas las herramientas y tecnología que utilizan “allá afuera”, pero tenemos una formación profesional para seguir aprendiendo. También se nos enseñó las bases para entender lo nuevo, la mentalidad para enfrentar y solucionar problemas, así como la actitud para saber trabajar en equipo y bajo presión.

El objetivo del proyecto “Análisis de precios” fue reducir costos, tiempo y recursos humanos, lo cual se logró mediante un previo análisis de las páginas donde se encontraba la información requerida de las entidades que contienen catálogos de productos necesarios para la empresa.

El resultado del análisis llevó a utilizar la herramienta “Screen-scraper” debido a que resultó ser la más completa y felxible en cuanto a la extracción de datos ya que se puede reforzar con otros lenguajes de programación para solucionar limitantes propias del software como son: el orden de navegación en la página, envío de datos variables, entre otras.

El desarrollo de este proyecto tuvo las siguientes aportaciones a la empresa: archivos estructurados en Excel para que el personal que no tiene conocimientos en programación pueda utilizar los datos extraídos de la página para su posterior análisis estadístico. Gracias a la automatización proporcionada en este proyecto se hicieron más eficientes los procesos dentro de la empresa, con lo cual se cumplieron los objetivos.

Se debe tomar en cuenta que se debe dar constante mantenimiento al proyecto debido a que las estructuras de las páginas pueden cambiar en el futuro lo cual requerirá cambios en el código fuente.

Los beneficios, en cuanto a crecimiento profesional, que se obtuvieron con el desarrollo de este proyecto fueron: trabajo en equipo, aprendizaje de lenguajes de programación propio de un software, comprensión de metodologías ágiles, análisis de solución, implementación de código fuente, comprensión de patrones y expresiones regulares.

GLOSARIO

Expresiones regulares: es una secuencia de caracteres que forma un patrón de búsqueda, principalmente utilizada para la búsqueda de patrones de cadenas de caracteres u operaciones de sustituciones.

MiPyME: acrónimo de "micro, pequeña y mediana empresa".

Rich snippets: son una convención a través de schema.org de los tres principales buscadores, Google, Bing y Yahoo, para etiquetar el contenido de las webs y facilitar que los resultados de búsqueda puedan mostrarse con información más avanzada a la tradicional.

CSV: del inglés *comma-separated values (valores separados por comas)*, es un archivo de texto que se caracteriza por tener sus valores ordenados y separados por comas.

AJAX: del inglés Asynchronous JavaScript And XML (JavaScript asíncrono y XML), es una técnica de desarrollo web para crear aplicaciones interactivas.

BIOS: del inglés Basic Input Output System (sistema básico de entrada y salida). Es la zona donde se configura el hardware que se usa al iniciar la computadora.

REFERENCIAS

Anónimo (12 de noviembre de 2016). Qué es SCRUM [Página de internet] Recuperado de: https://proyectosagiles.org/que-es-scrum/

Anónimo (13 de noviembre de 2016). Web scraping [Página de internet] Recuperado de: https://es.wikipedia.org/wiki/Web_scraping

Screen-Scraper (13 de noviembre de 2016). Proxy Server Setup [Imagen] Recuperado de: http://community.screen-scraper.com/files/media/tutorials/tutorial1/how_the_proxy_server_works.png

Anónimo (13 de noviembre de 2016). Empezando - Acerca del control de versiones [Página de internet] Recuperado de: https://git-scm.com/book/es/v1/Empezando-Acerca-del-control-de-versiones

Anónimo (17 de diciembre de 2016). Beanshell [Página de internet] Recuperado de: http://www.guia-ubuntu.com/index.php/BeanShell

Redaccion on sábado (19 de febrero de 2017). Iniciar sesión automaticamente con un usuario en windows 7 [Página de internet] Recuperado de: http://www.torresoft.com/2013/02/iniciar-sesion-automaticamente-con-un.html

Anónimo (19 de febrero de 2017). Pequeña y mediana empresa [Página de internet] Recuperado de: https://es.wikipedia.org/wiki/Peque%C3%B1a_y_mediana_empresa

ANEXOS

API Screen-Scraper

Se enlistan y describen las funciones que se usaron. Si se necesita consultar la API completa, se encuentra en: http://community.screen-scraper.com/documentation/api

Scraping Engine API

FUNCIÓN	DESCRIPCIÓN
Object dataRecord.get (Object key)	Obtiene la información extraída de los patrones extractores.
void log.log (Object message)	Muestra un mensaje. Dentro de Screen-Scraper lo muestra en negro.
void log.logError (Object message)	Muestra un mensaje. Dentro de Screen-Scraper lo muestra en color rojo.
void log.logInfo (Object message)	Muestra un mensaje. Dentro de Screen-Scraper lo muestra en color azul.
void log.logVariables ( )	Muestra todas las variables cargadas.
DataRecord scrapeableFile.getASPXValues (boolean onlyStandard)	*Obtiene los valores de ASPX .NET: __VIEWSTATE, __EVENTTARGET, __EVENTVALIDATION y __EVENTARGUMENT.*
void session.clearCookies ( )	Limpia las cookies de la sesión.
void session.executeScriptWithContext (String scriptName)	Ejecuta el script indicado.
Object session.getv (String identifier)	Obtiene el valor de una variable de sesión.
Object session.getVariable (String identifier)	
void session.log (Object message)	Muestra un mensaje. Dentro de Screen-Scraper lo muestra en negro.
void session.loadVariables (String fileToReadFrom)	Carga las variables de un archivo.
void session.logVariables ( )	Muestra todas las variables cargadas en un formato especial.
void session.logInfo (Object message)	Muestra un mensaje. Dentro de Screen-Scraper lo muestra en color azul.
void session.logError (Object message)	Muestra un mensaje. Dentro de Screen-Scraper lo muestra en color rojo.
void session.scrapeFile (String scrapeableFileIdentifier)	Ejecuta el archivo scrapeable indicado.
void session.setv (String identifier, Object value)	Crea una variable de sesión. Si ya existe, sobreescribe el valor.
void session.setVariable (String identifier, Object value)	
String sutil.getCurrentDate (String format) MM mes a dos dígitos dd día a dos dígitos yyyy año a tres dígitos HH hora a dos dígitos	*Obtiene fecha y/o hora del sistema en el formato que se le indique.*

mm minutos a dos dígitos ss segundos a dos dígitos SS décimas de segundo zzz zona horaria	
void sutil.randomPause (long min, long max)	*Hace una pausa random dentro de un rango indicado.*

Utilities API

FUNCIÓN	DESCRIPCIÓN
CsvWriter CsvWriter (String filePath) *CsvWriter CsvWriter (String filePath, boolean addTimeStamp)* *CsvWriter CsvWriter (String filePath, char separator)* *CsvWriter CsvWriter (String filePath, char separator, boolean addTimeStamp)* *CsvWriter CsvWriter (String filePath, char separator, char quotechar)* *CsvWriter CsvWriter (String filePath, char separator, char quotechar, char escapechar)* *CsvWriter CsvWriter (String filePath, char separator, char quotechar, String lineEnd)* *CsvWriter CsvWriter (String filePath, char separator, char quotechar, char escapechar, String lineEnd)*	Crea un archivo csv
void csvWriter.close ( )	Cierra el archivo csv
void csvWriter.flush ( )	*Limpia el buffer de escritura.*
void csvWriter.setHeader (String[ ] header)	Si no existen, crea los encabezados del archivo csv.
void csvWriter.write (DataRecord dataRecord)	*Escribe los valores indicados al archivo.*

Códigos de Screen-Scraper

Estos son los códigos que se hicieron y posteriormente “Screen-Scraper 6.0 Enterprise edition” generó al ser exportados.

Comercial Mexicana®

```
<?xml version="1.0" encoding="UTF-8"?>
<scraping-session use-strict-mode="true"><script-instances><script-instances when-to-run="10" sequence="1" enabled="true"><script><script-text>session.setVariable("Fecha", sutil.getCurrentDate("yyyyMMdd"));
session.logInfo( "Cargando datos..." );

//session.loadVariables("C:/Users/Viridiana/Documents/Datos/ComercialMexicana.txt");
session.loadVariables(session.getv("ruta"));
session.logVariables();
```

```
ConjuntoSucursales= session.getVariable("sucursales");
ConjuntoDepartamentos= session.getVariable("departamentos");
ConjuntoEspeciales= session.getVariable("depEspeciales");

String[] Sucursales = ConjuntoSucursales.split(";");
String[] Departamento = ConjuntoDepartamentos.split(";");
String[] Especial = ConjuntoEspeciales.split(";");

for (int i = 0; i &lt; Sucursales.length; i++)
{
	String[] Sucursal = Sucursales[i].split(":");
	session.setVariable("succId", Sucursal[0]);
	session.setVariable("NombreSucursal", Sucursal[1]);

	for (int j = 0; j &lt; Departamento.length; j++)
	{
		String[] Subdepartamento = Departamento[j].split(":");
		String[] Seccion=Subdepartamento[1].split(",");
		for (int k = 0; k &lt; Seccion.length; k++)
		{

			session.setVariable("pasId", Subdepartamento[0]);
			session.setVariable("padreId", Seccion[k]);
			sutil.randomPause(1000, 47000);
			session.scrapeFile( "Obtener precios I" );
		}
	}

	for (int l = 0; l &lt; Especial.length; l++)
	{
		session.setVariable("padreId", Especial[l]);
		sutil.randomPause(1000, 47000);
		session.scrapeFile( "Obtener precios II" );
	}
}
```

```
</script-text><name>ComercialMexicana_CargaDatos</name><language>Interpreted
Java</language></script></script-instances><owner-type>ScrapingSession</owner-
type><owner-name>Comercial_Mexicana</owner-name></script-
instances><name>Comercial_Mexicana</name><notes>Precios de la comercial mexicana.
</notes><cookiePolicy>0</cookiePolicy><maxHTTPRequests>1</maxHTTPRequests><external_pr
oxy_username></external_proxy_username><external_proxy_password></external_proxy_passw
ord><external_proxy_host></external_proxy_host><external_proxy_port></external_proxy_p
ort><external_nt_proxy_username></external_nt_proxy_username><external_nt_proxy_passwo
rd></external_nt_proxy_password><external_nt_proxy_domain></external_nt_proxy_domain><
external_nt_proxy_host></external_nt_proxy_host><anonymize>false</anonymize><terminate
_proxies_on_completion>false</terminate_proxies_on_completion><number_of_required_prox
```

```
ies>5</number_of_required_proxies><originator_edition>2</originator_edition><logging_l
evel>1</logging_level><date_exported>enero 08, 2016
22:01:40</date_exported><character_set>UTF-
8</character_set><created_by_version>6.0</created_by_version><scrapeable-files
sequence="-1" will-be-invoked-manually="true" tidy-html="dont"><last-scraped-
data></last-scraped-
data><URL>http://www.superensucasa.com/lacomer/doHome.action</URL><last-
request></last-request><name>Obtener precios II</name><extractor-patterns sequence="1"
automatically-save-in-session-variable="false" if-saved-in-session-variable="0"
filter-duplicates="false" cache-data-set="false" will-be-invoked-
manually="false"><pattern-text>&lt;h4&gt;

                                        ~@marca@~
                                        &lt;/h4&gt;
                                        &lt;div class="parrafo"
style="width:250px;"&gt;
                                        &lt;p&gt;
                                            &lt;a href=~@zelda@~&gt;
                                             ~@producto@~
                                             &lt;/a&gt;

&amp;nbsp~@extra@~&amp;nbsp;~@contenido@~&amp;nbsp;~@medida@~&amp;nbsp;
                                       &lt;/p&gt;
                                       &lt;/div&gt;
                                       &lt;div class="price"&gt;
                                           &lt;span&gt;

                                                  &lt;a
class="txt_naranja_productos"&gt;~@precio@~&lt;/a&gt;</pattern-
text><identifier>Untitled Extractor Pattern</identifier><extractor-pattern-tokens
optional="false" save-in-session-variable="false" compound-key="true" strip-
html="false" resolve-relative-url="false" replace-html-entities="false" trim-white-
space="false" exclude-from-data="true" null-session-variable="false"
sequence="2"><regular-expression>[^&lt;&gt;]*</regular-
expression><identifier>zelda</identifier></extractor-pattern-tokens><extractor-
pattern-tokens optional="false" save-in-session-variable="false" compound-key="true"
strip-html="false" resolve-relative-url="false" replace-html-entities="false" trim-
white-space="false" exclude-from-data="true" null-session-variable="false"
sequence="4"><regular-expression>[^&lt;&gt;]*</regular-
expression><identifier>extra</identifier></extractor-pattern-tokens><extractor-
pattern-tokens optional="false" save-in-session-variable="false" compound-key="true"
strip-html="false" resolve-relative-url="false" replace-html-entities="false" trim-
white-space="false" exclude-from-data="false" null-session-variable="false"
sequence="7"><regular-expression></regular-
expression><identifier>precio</identifier></extractor-pattern-tokens><extractor-
pattern-tokens optional="false" save-in-session-variable="false" compound-key="true"
strip-html="false" resolve-relative-url="false" replace-html-entities="false" trim-
white-space="false" exclude-from-data="false" null-session-variable="false"
sequence="1"><regular-expression>[^&lt;&gt;]*</regular-
expression><identifier>marca</identifier></extractor-pattern-tokens><extractor-
pattern-tokens optional="false" save-in-session-variable="false" compound-key="true"
strip-html="false" resolve-relative-url="false" replace-html-entities="false" trim-
white-space="false" exclude-from-data="false" null-session-variable="false"
```

```
sequence="3"><regular-expression>[^&lt;&gt;]*</regular-
expression><identifier>producto</identifier></extractor-pattern-tokens><extractor-
pattern-tokens optional="false" save-in-session-variable="false" compound-key="true"
strip-html="false" resolve-relative-url="false" replace-html-entities="false" trim-
white-space="false" exclude-from-data="false" null-session-variable="false"
sequence="6"><regular-expression>[^&lt;&gt;]*</regular-
expression><identifier>medida</identifier></extractor-pattern-tokens><extractor-
pattern-tokens optional="false" save-in-session-variable="false" compound-key="true"
strip-html="false" resolve-relative-url="false" replace-html-entities="false" trim-
white-space="false" exclude-from-data="false" null-session-variable="false"
sequence="5"><regular-expression>[\d,]+</regular-
expression><identifier>contenido</identifier></extractor-pattern-tokens><script-
instances><script-instances when-to-run="80" sequence="1"
enabled="true"><script><script-text>session.setVariable("Fecha",
sutil.getCurrentDate("yyyyMMdd"));
session.logInfo( "Cargando datos..." );

//session.loadVariables("C:/Users/Viridiana/Documents/Datos/ComercialMexicana.txt");
session.loadVariables(session.getv("ruta"));
session.logVariables();

ConjuntoSucursales= session.getVariable("sucursales");
ConjuntoDepartamentos= session.getVariable("departamentos");
ConjuntoEspeciales= session.getVariable("depEspeciales");

String[] Sucursales = ConjuntoSucursales.split(";");
String[] Departamento = ConjuntoDepartamentos.split(";");
String[] Especial = ConjuntoEspeciales.split(";");

for (int i = 0; i &lt; Sucursales.length; i++)
{
      String[] Sucursal = Sucursales[i].split(":");
      session.setVariable("succId", Sucursal[0]);
      session.setVariable("NombreSucursal", Sucursal[1]);

      for (int j = 0; j &lt; Departamento.length; j++)
      {
            String[] Subdepartamento = Departamento[j].split(":");
            String[] Seccion=Subdepartamento[1].split(",");
            for (int k = 0; k &lt; Seccion.length; k++)
            {

                  session.setVariable("pasId", Subdepartamento[0]);
                  session.setVariable("padreId", Seccion[k]);
                  sutil.randomPause(1000, 47000);
                  session.scrapeFile( "Obtener precios I" );
            }
      }

      for (int l = 0; l &lt; Especial.length; l++)
      {
            session.setVariable("padreId", Especial[l]);
            sutil.randomPause(1000, 47000);
```

```
            session.scrapeFile( "Obtener precios II" );
      }
}
```

```
</script-text><name>ComercialMexicana_CargaDatos</name><language>Interpreted
Java</language></script></script-instances><owner-type>ExtractorPattern</owner-
type><owner-name>Untitled Extractor Pattern</owner-name></script-
instances></extractor-patterns><HTTPParameters
sequence="9"><key>jsp</key><type>GET</type><value>PasilloPadre.jsp</value></HTTPParame
ters><HTTPParameters
sequence="3"><key>succId</key><type>GET</type><value>~#succId#~</value></HTTPParameter
s><HTTPParameters
sequence="5"><key>path</key><type>GET</type><value>,52</value></HTTPParameters><HTTPPa
rameters
sequence="7"><key>mov</key><type>GET</type><value>1</value></HTTPParameters><HTTPParam
eters
sequence="6"><key>pathPadre</key><type>GET</type><value></value></HTTPParameters><HTTP
Parameters
sequence="8"><key>subOpc</key><type>GET</type><value>0</value></HTTPParameters><HTTPPa
rameters
sequence="4"><key>opcion</key><type>GET</type><value>listaproductos</value></HTTPParam
eters><HTTPParameters
sequence="10"><key>succFmt</key><type>GET</type><value>100</value></HTTPParameters><HT
TPParameters
sequence="2"><key>pasId</key><type>GET</type><value>~#padreId#~</value></HTTPParameter
s><HTTPParameters
sequence="1"><key>padreId</key><type>GET</type><value>~#padreId#~</value></HTTPParamet
ers><script-instances><owner-type>ScrapeableFile</owner-type><owner-name>Obtener
precios II</owner-name></script-instances></scrapeable-files><scrapeable-files
sequence="-1" will-be-invoked-manually="true" tidy-html="dont"><last-scraped-
data></last-scraped-
data><URL>http://www.superensucasa.com/lacomer/doHome.action</URL><last-
request></last-request><name>Obtener precios I</name><extractor-patterns sequence="1"
automatically-save-in-session-variable="false" if-saved-in-session-variable="0"
filter-duplicates="false" cache-data-set="false" will-be-invoked-
manually="false"><pattern-text>&lt;h4&gt;

                                        ~@marca@~
                                        &lt;/h4&gt;
                                        &lt;div class="parrafo"
style="width:250px;"&gt;
                                        &lt;p&gt;
                                            &lt;a href=~@zelda@~&gt;
                                             ~@producto@~
                                             &lt;/a&gt;

&amp;nbsp~@extra@~&amp;nbsp;~@contenido@~&amp;nbsp;~@medida@~&amp;nbsp;
                                       &lt;/p&gt;
```

```
                                        &lt;/div&gt;
                                        &lt;div class="price"&gt;
                                            &lt;span&gt;

                                                &lt;a
class="txt_naranja_productos"&gt;~@precio@~&lt;/a&gt;</pattern-
text><identifier>Untitled Extractor Pattern</identifier><extractor-pattern-tokens
optional="false" save-in-session-variable="false" compound-key="true" strip-
html="false" resolve-relative-url="false" replace-html-entities="false" trim-white-
space="false" exclude-from-data="false" null-session-variable="false"
sequence="1"><regular-expression>[^&lt;&gt;]*</regular-
expression><identifier>marca</identifier></extractor-pattern-tokens><extractor-
pattern-tokens optional="false" save-in-session-variable="false" compound-key="true"
strip-html="false" resolve-relative-url="false" replace-html-entities="false" trim-
white-space="false" exclude-from-data="false" null-session-variable="false"
sequence="7"><regular-expression></regular-
expression><identifier>precio</identifier></extractor-pattern-tokens><extractor-
pattern-tokens optional="false" save-in-session-variable="false" compound-key="true"
strip-html="false" resolve-relative-url="false" replace-html-entities="false" trim-
white-space="false" exclude-from-data="false" null-session-variable="false"
sequence="3"><regular-expression>[^&lt;&gt;]*</regular-
expression><identifier>producto</identifier></extractor-pattern-tokens><extractor-
pattern-tokens optional="false" save-in-session-variable="false" compound-key="true"
strip-html="false" resolve-relative-url="false" replace-html-entities="false" trim-
white-space="false" exclude-from-data="true" null-session-variable="false"
sequence="4"><regular-expression>[^&lt;&gt;]*</regular-
expression><identifier>extra</identifier></extractor-pattern-tokens><extractor-
pattern-tokens optional="false" save-in-session-variable="false" compound-key="true"
strip-html="false" resolve-relative-url="false" replace-html-entities="false" trim-
white-space="false" exclude-from-data="false" null-session-variable="false"
sequence="6"><regular-expression>[^&lt;&gt;]*</regular-
expression><identifier>medida</identifier></extractor-pattern-tokens><extractor-
pattern-tokens optional="false" save-in-session-variable="false" compound-key="true"
strip-html="false" resolve-relative-url="false" replace-html-entities="false" trim-
white-space="false" exclude-from-data="true" null-session-variable="false"
sequence="2"><regular-expression>[^&lt;&gt;]*</regular-
expression><identifier>zelda</identifier></extractor-pattern-tokens><extractor-
pattern-tokens optional="false" save-in-session-variable="false" compound-key="true"
strip-html="false" resolve-relative-url="false" replace-html-entities="false" trim-
white-space="false" exclude-from-data="false" null-session-variable="false"
sequence="5"><regular-expression>[\d,]+</regular-
expression><identifier>contenido</identifier></extractor-pattern-tokens><script-
instances><script-instances when-to-run="80" sequence="1"
enabled="true"><script><script-text>session.logInfo( "Obteniendo precios..." );

Ruta="C:/Users/Viridiana/Documents/Precios/Comercial Mexicana/";
//Archivo=sutil.getCurrentDate("yyyyMMdd")+session.getVariable("NombreSucursal")+".csv
";
Archivo=session.getVariable("Fecha")+session.getVariable("NombreSucursal")+".csv";


CsvWriter head = new CsvWriter(Ruta+Archivo);
```

```
String[] header = {"marca", "producto", "contenido", "precio", "fuente", "entidad",
"municipio", "diaSemana", "anio"};
head.setHeader(header);
head.close();

precio=dataRecord.get("precio").replace(',', ' ');
contenido=dataRecord.get("contenido").replace(',', '
')+dataRecord.get("medida").replace(',', ' ');
producto=dataRecord.get("producto").replace(',', ' ');
marca=dataRecord.get("marca").replace(',', ' ');

FileWriter out = null;
try
{
	out = new FileWriter(Ruta+Archivo,true);

	out.write( marca+",");
	out.write( producto+",");
	out.write( contenido+",");
	out.write( precio+",");
	out.write( "\n" );


	out.close();
}
catch( Exception e )
{
	log.logError("Error al crear archivo: " + e.getMessage() );
}</script-text><name>ComercialMexicana_EscribePrecios</name><language>Interpreted
Java</language></script></script-instances><owner-type>ExtractorPattern</owner-
type><owner-name>Untitled Extractor Pattern</owner-name></script-
instances></extractor-patterns><HTTPParameters
sequence="10"><key>agruId</key><type>GET</type><value>~#pasId#~</value></HTTPParameter
s><HTTPParameters
sequence="6"><key>mov</key><type>GET</type><value>1</value></HTTPParameters><HTTPParam
eters
sequence="2"><key>pasId</key><type>GET</type><value>~#pasId#~</value></HTTPParameters>
<HTTPParameters
sequence="11"><key>succFmt</key><type>GET</type><value>100</value></HTTPParameters><HT
TPParameters
sequence="7"><key>subOpc</key><type>GET</type><value>0</value></HTTPParameters><HTTPPa
rameters
sequence="8"><key>jsp</key><type>GET</type><value>PasilloPadre.jsp</value></HTTPParame
ters><HTTPParameters
sequence="5"><key>pathPadre</key><type>GET</type><value></value></HTTPParameters><HTTP
Parameters
sequence="1"><key>padreId</key><type>GET</type><value>~#padreId#~</value></HTTPParamet
ers><HTTPParameters
sequence="4"><key>path</key><type>GET</type><value>,77,77</value></HTTPParameters><HTT
PParameters
sequence="3"><key>opcion</key><type>GET</type><value>listaproductos</value></HTTPParam
eters><HTTPParameters
sequence="9"><key>succId</key><type>GET</type><value>~#succId#~</value></HTTPParameter
```

```
s><script-instances><owner-type>ScrapeableFile</owner-type><owner-name>Obtener precios
I</owner-name></script-instances></scrapeable-files></scraping-session>
```

Walmart®

Actualización de departamentos

```
<?xml version="1.0" encoding="UTF-8"?>
<scraping-session use-strict-mode="true"><script-instances><owner-
type>ScrapingSession</owner-type><owner-name>Walmart_actualizaciones</owner-
name></script-
instances><name>Walmart_actualizaciones</name><notes></notes><cookiePolicy>0</cookiePo
licy><maxHTTPRequests>1</maxHTTPRequests><external_proxy_username></external_proxy_use
rname><external_proxy_password></external_proxy_password><external_proxy_host></extern
al_proxy_host><external_proxy_port></external_proxy_port><external_nt_proxy_username><
/external_nt_proxy_username><external_nt_proxy_password></external_nt_proxy_password><
external_nt_proxy_domain></external_nt_proxy_domain><external_nt_proxy_host></external
_nt_proxy_host><anonymize>false</anonymize><terminate_proxies_on_completion>false</ter
minate_proxies_on_completion><number_of_required_proxies>5</number_of_required_proxies
><originator_edition>2</originator_edition><logging_level>1</logging_level><date_expor
ted>febrero 11, 2016 15:52:57</date_exported><character_set>UTF-
8</character_set><created_by_version>6.0</created_by_version><scrapeable-files
sequence="1" will-be-invoked-manually="false" tidy-html="jtidy"><last-scraped-
data></last-scraped-data><URL>http://www.walmart.com.mx/super/</URL><last-
request></last-request><name>Departamento</name><extractor-patterns sequence="1"
automatically-save-in-session-variable="false" if-saved-in-session-variable="0"
filter-duplicates="false" cache-data-set="false" will-be-invoked-
manually="false"><pattern-text>href="/super/Categoria.aspx?Departamento=d-
~@DEPARTAMENTO@~"&#xd;
</pattern-text><identifier>Untitled Extractor Pattern</identifier><extractor-pattern-
tokens optional="false" save-in-session-variable="false" compound-key="true" strip-
html="false" resolve-relative-url="false" replace-html-entities="false" trim-white-
space="false" exclude-from-data="false" null-session-variable="false"
sequence="1"><regular-expression></regular-
expression><identifier>DEPARTAMENTO</identifier></extractor-pattern-tokens><script-
instances><script-instances when-to-run="80" sequence="1"
enabled="true"><script><script-text>Ruta="C:/Users/Viridiana/Documents/Screen-
Scraper/";
//Ruta= session.getVariable("GuardarEn");
Archivo="WalmartDepartamentos.csv";


CsvWriter file = new CsvWriter(Ruta+Archivo);

String[] header = {"DEPARTAMENTO"};
file.setHeader(header);
HashMap hm = new HashMap();
hm.put("DEPARTAMENTO", dataRecord.get("DEPARTAMENTO"));
file.write(hm);
file.flush();
```

```
file.close();</script-
text><name>Walmart_actualizaciones_super</name><language>Interpreted
Java</language></script></script-instances><owner-type>ExtractorPattern</owner-
type><owner-name>Untitled Extractor Pattern</owner-name></script-
instances></extractor-patterns><script-instances><owner-type>ScrapeableFile</owner-
type><owner-name>Departamento</owner-name></script-instances></scrapeable-
files></scraping-session>
```

Obtención y escritura de precios

```
<?xml version="1.0" encoding="UTF-8"?>
<scraping-session use-strict-mode="true"><script-instances><script-instances when-to-
run="10" sequence="1" enabled="true"><script><script-text>session.setVariable("Fecha",
sutil.getCurrentDate("yyyyMMdd"));</script-
text><name>Fecha</name><language>Interpreted Java</language></script></script-
instances><script-instances when-to-run="10" sequence="2"
enabled="true"><script><script-text>session.logInfo( "Cargando datos..." );

//session.loadVariables("C:/Users/Viridiana/Documents/Screen-
Scraper/Datos/PROFECO_Archivos.txt");
//session.loadVariables("C:/Users/Viridiana/Documents/Screen-
Scraper/Datos/Walmart_Archivos.txt");
session.loadVariables(session.getv("ruta"));
session.logVariables();</script-text><name>CargaDatos</name><language>Interpreted
Java</language></script></script-instances><script-instances when-to-run="10"
sequence="3" enabled="true"><script><script-text>String[] parseCSVLine(String line,
int index, int columnsToGet){
    int START_STATE = 0;
    int FIRST_QUOTE = 1;
    int SECOND_QUOTE = 2;
    int IN_WORD = 3;
    int IN_WORD_WITHOUT_QUOTES = 4;
    int state = START_STATE;
    String word = "";
    ArrayList lines = new ArrayList();
    char[] chars = line.toCharArray();

     for (int i = 0; i &lt; chars.length; i++){
        char c = chars[i];

        if (c == '"'){
            if (state == START_STATE){
                state = FIRST_QUOTE;
            }
            else if ((state == FIRST_QUOTE) || (state == IN_WORD)){
                state = SECOND_QUOTE;
            }
            else if (state == SECOND_QUOTE){
                word += ("" + c);
                state = IN_WORD;
            }
```

```
        }
        else if (c == ','){
            if ((state == SECOND_QUOTE) || (state == IN_WORD_WITHOUT_QUOTES)){
                state = START_STATE;

                lines.add(word);
                if (lines.size() == columnsToGet) break;
                    word = "";
            }
            else if (state == START_STATE){
                state = START_STATE;
                lines.add(word.replaceAll("\"\"", "\""));
            }
            else{
                word += ("" + c);
                state = IN_WORD;
            }
        }
        else{
            if (state == START_STATE) state = IN_WORD_WITHOUT_QUOTES;
            else if (state != IN_WORD_WITHOUT_QUOTES){
                 state = IN_WORD;
                word += ("" + c);
            }
        }
    }
    if (lines.size() &lt; columnsToGet){
        if ((state == SECOND_QUOTE) || (state == IN_WORD_WITHOUT_QUOTES))
             lines.add(word.replaceAll("\"\"", "\""));
    }
    String[] linesArray = new String[lines.size()];

    for (int i = 0; i &lt; lines.size(); i++){
        linesArray[i] = (String) lines.get(i);
    }

    return linesArray;
}


// File from which to read.
depto=session.getVariable("Departamentos");
File inputFile = new File(depto);

FileReader in = new FileReader( inputFile );
BufferedReader buffRead = new BufferedReader( in );

// Read the file in line-by-line.
int index = 0;
while( ( searchTerm = buffRead.readLine() )!=null){
    // Don't read header row
    if (index&gt;0){
        // Parse the line into an array
```

```
        line = parseCSVLine(searchTerm, index, 1);

        // Get the values
        dep = line[0];



        // Set the needed values as session vaiables
        session.setVariable("DEPARTAMENTO", dep);


        // Scrape for those values
        session.scrapeFile("Obtener_precio");

    }
    index++;
}

// Close up the file.
in.close();
buffRead.close();</script-
text><name>Walmart_cargaDepartamentos</name><language>Interpreted
Java</language></script></script-instances><owner-type>ScrapingSession</owner-
type><owner-name>Walmart</owner-name></script-
instances><name>Walmart</name><notes></notes><cookiePolicy>0</cookiePolicy><maxHTTPReq
uests>1</maxHTTPRequests><external_proxy_username></external_proxy_username><external_
proxy_password></external_proxy_password><external_proxy_host></external_proxy_host><e
xternal_proxy_port></external_proxy_port><external_nt_proxy_username></external_nt_pro
xy_username><external_nt_proxy_password></external_nt_proxy_password><external_nt_prox
y_domain></external_nt_proxy_domain><external_nt_proxy_host></external_nt_proxy_host><
anonymize>false</anonymize><terminate_proxies_on_completion>false</terminate_proxies_o
n_completion><number_of_required_proxies>5</number_of_required_proxies><originator_edi
tion>2</originator_edition><logging_level>1</logging_level><date_exported>febrero 11,
2016 15:52:57</date_exported><character_set>UTF-
8</character_set><created_by_version>6.0</created_by_version><scrapeable-files
sequence="-1" will-be-invoked-manually="true" tidy-html="jtidy"><last-scraped-
data></last-scraped-
data><URL>http://www.walmart.com.mx/super/WebControls/hlSearch.ashx</URL><last-
request></last-request><name>Obtener_precio</name><extractor-patterns sequence="1"
automatically-save-in-session-variable="false" if-saved-in-session-variable="0"
filter-duplicates="false" cache-data-set="false" will-be-invoked-
manually="false"><pattern-
text>{"Cantidad":"~@EXTRA@~","Description":"~@descripcion@~"~@EXTRA@~,"LargeDescriptio
n":"~@producto@~","Precio":"~@precio@~",~@EXTRA@~,"upc":"~@upc@~",~@EXTRA@~,"Brand":"~
@marca@~","FamilyName":"~@familia@~","DepartmentName":"d-
~@departamento@~","LineName":"~@linea@~",~@EXTRA@~}</pattern-text><identifier>Untitled
Extractor Pattern</identifier><extractor-pattern-tokens optional="false" save-in-
session-variable="false" compound-key="true" strip-html="false" resolve-relative-
url="false" replace-html-entities="false" trim-white-space="false" exclude-from-
data="false" null-session-variable="false" sequence="4"><regular-
expression>[^"]*</regular-expression><identifier>producto</identifier></extractor-
pattern-tokens><extractor-pattern-tokens optional="false" save-in-session-
variable="false" compound-key="true" strip-html="false" resolve-relative-url="false"
```

```
replace-html-entities="false" trim-white-space="false" exclude-from-data="false" null-
session-variable="false" sequence="9"><regular-expression>[^"]*</regular-
expression><identifier>marca</identifier></extractor-pattern-tokens><extractor-
pattern-tokens optional="false" save-in-session-variable="false" compound-key="true"
strip-html="false" resolve-relative-url="false" replace-html-entities="false" trim-
white-space="false" exclude-from-data="false" null-session-variable="false"
sequence="5"><regular-expression>[^"]*</regular-
expression><identifier>precio</identifier></extractor-pattern-tokens><extractor-
pattern-tokens optional="false" save-in-session-variable="false" compound-key="true"
strip-html="false" resolve-relative-url="false" replace-html-entities="false" trim-
white-space="false" exclude-from-data="true" null-session-variable="false"
sequence="11"><regular-expression>[^"]*</regular-
expression><identifier>departamento</identifier></extractor-pattern-tokens><extractor-
pattern-tokens optional="false" save-in-session-variable="false" compound-key="true"
strip-html="false" resolve-relative-url="false" replace-html-entities="false" trim-
white-space="false" exclude-from-data="false" null-session-variable="false"
sequence="7"><regular-expression>[^"]*</regular-
expression><identifier>upc</identifier></extractor-pattern-tokens><extractor-pattern-
tokens optional="false" save-in-session-variable="false" compound-key="true" strip-
html="false" resolve-relative-url="false" replace-html-entities="false" trim-white-
space="false" exclude-from-data="true" null-session-variable="false"
sequence="10"><regular-expression>[^"]*</regular-
expression><identifier>familia</identifier></extractor-pattern-tokens><extractor-
pattern-tokens optional="false" save-in-session-variable="false" compound-key="true"
strip-html="false" resolve-relative-url="false" replace-html-entities="false" trim-
white-space="false" exclude-from-data="true" null-session-variable="false"
sequence="12"><regular-expression>[^"]*</regular-
expression><identifier>linea</identifier></extractor-pattern-tokens><extractor-
pattern-tokens optional="false" save-in-session-variable="false" compound-key="true"
strip-html="false" resolve-relative-url="false" replace-html-entities="false" trim-
white-space="false" exclude-from-data="true" null-session-variable="false"
sequence="13"><regular-expression></regular-
expression><identifier>EXTRA</identifier></extractor-pattern-tokens><extractor-
pattern-tokens optional="false" save-in-session-variable="false" compound-key="true"
strip-html="true" resolve-relative-url="false" replace-html-entities="true" trim-
white-space="false" exclude-from-data="false" null-session-variable="false"
sequence="2"><regular-expression>[^"]*</regular-
expression><identifier>descripcion</identifier></extractor-pattern-tokens><script-
instances><script-instances when-to-run="80" sequence="1"
enabled="true"><script><script-text>session.logInfo( "Obteniendo precios..." );

Ruta="C:/Users/Viridiana/Documents/Screen-Scraper/Precios/Walmart/";
//Ruta= session.getVariable("GuardarEn");
Archivo=session.getVariable("Fecha")+"Walmart.csv";


CsvWriter head = new CsvWriter(Ruta+Archivo);

String[] header = {"upc", "marca", "producto", "descripcion", "precio"};
head.setHeader(header);
head.close();
String[] c= {
"\""+dataRecord.get("upc")+"\"",
```

```
"\""+dataRecord.get("marca")+"\"",
"\""+dataRecord.get("producto")+"\"",
"\""+dataRecord.get("descripcion")+"\"",
"\""+dataRecord.get("precio")+"\"",
};
FileWriter out = null;
try
{
	out = new FileWriter(Ruta+Archivo,true);

	for (int k = 0; k &lt; c.length; k++)
	{
		out.write( c[k]+",");
	}
	out.write( "\n" );


	out.close();
}
catch( Exception e )
{
	log.logError("Error al crear archivo: " + e.getMessage() );
}</script-text><name>Walmart_EscribePrecios</name><language>Interpreted
Java</language></script></script-instances><owner-type>ExtractorPattern</owner-
type><owner-name>Untitled Extractor Pattern</owner-name></script-
instances></extractor-patterns><HTTPParameters
sequence="4"><key>Linea</key><type>GET</type><value></value></HTTPParameters><HTTPPara
meters
sequence="5"><key>marca</key><type>GET</type><value></value></HTTPParameters><HTTPPara
meters
sequence="1"><key>Text</key><type>GET</type><value></value></HTTPParameters><HTTPParam
eters
sequence="2"><key>Departamento</key><type>GET</type><value>~#DEPARTAMENTO#~</value></H
TTPParameters><HTTPParameters
sequence="3"><key>Familia</key><type>GET</type><value></value></HTTPParameters><script
-instances><owner-type>ScrapeableFile</owner-type><owner-name>Obtener_precio</owner-
name></script-instances></scrapeable-files></scraping-session>
```

PROFECO

Actualización de ciudades y categorías

```
<?xml version="1.0" encoding="UTF-8"?>
<scraping-session use-strict-mode="true"><script-instances><owner-
type>ScrapingSession</owner-type><owner-name>PROFECO_Actualizaciones</owner-
name></script-instances><name>PROFECO_Actualizaciones</name><notes>Sesión para revisar
cambios en las categorías de productos
</notes><cookiePolicy>0</cookiePolicy><maxHTTPRequests>1</maxHTTPRequests><external_pr
oxy_username></external_proxy_username><external_proxy_password></external_proxy_passw
ord><external_proxy_host></external_proxy_host><external_proxy_port></external_proxy_p
ort><external_nt_proxy_username></external_nt_proxy_username><external_nt_proxy_passwo
rd></external_nt_proxy_password><external_nt_proxy_domain></external_nt_proxy_domain><
```

```
external_nt_proxy_host></external_nt_proxy_host><anonymize>false</anonymize><terminate
_proxies_on_completion>false</terminate_proxies_on_completion><number_of_required_prox
ies>5</number_of_required_proxies><originator_edition>2</originator_edition><logging_l
evel>1</logging_level><date_exported>febrero 11, 2016
15:52:56</date_exported><character_set>UTF-
8</character_set><created_by_version>6.0</created_by_version><scrapeable-files
sequence="4" will-be-invoked-manually="false" tidy-html="jtidy"><last-scraped-
data></last-scraped-
data><URL>http://www.profeco.gob.mx/precios/canasta/arbol_frame.aspx</URL><last-
request></last-request><name>Lista productos</name><extractor-patterns sequence="1"
automatically-save-in-session-variable="false" if-saved-in-session-variable="0"
filter-duplicates="false" cache-data-set="false" will-be-invoked-
manually="false"><pattern-text>&lt;a class="~@Arbol@~"
href="listaProdArbol.aspx?codigo=~@ID_PRODUCTO@~&amp;amp;~@Extra2@~"&gt;~@PRODUCTO@~&l
t;/a&gt;</pattern-text><identifier>Untitled Extractor Pattern</identifier><extractor-
pattern-tokens optional="false" save-in-session-variable="false" compound-key="true"
strip-html="false" resolve-relative-url="false" replace-html-entities="false" trim-
white-space="false" exclude-from-data="true" null-session-variable="false"
sequence="3"><regular-expression>[^&lt;&gt;]*</regular-
expression><identifier>Extra2</identifier></extractor-pattern-tokens><extractor-
pattern-tokens optional="false" save-in-session-variable="false" compound-key="true"
strip-html="true" resolve-relative-url="false" replace-html-entities="true" trim-
white-space="false" exclude-from-data="false" null-session-variable="false"
sequence="4"><regular-expression>[^&lt;&gt;]*</regular-
expression><identifier>PRODUCTO</identifier></extractor-pattern-tokens><extractor-
pattern-tokens optional="false" save-in-session-variable="false" compound-key="true"
strip-html="false" resolve-relative-url="false" replace-html-entities="false" trim-
white-space="false" exclude-from-data="false" null-session-variable="false"
sequence="2"><regular-expression>[\d,]+</regular-
expression><identifier>ID_PRODUCTO</identifier></extractor-pattern-tokens><extractor-
pattern-tokens optional="false" save-in-session-variable="false" compound-key="true"
strip-html="false" resolve-relative-url="false" replace-html-entities="false" trim-
white-space="false" exclude-from-data="true" null-session-variable="false"
sequence="1"><regular-expression>[^"]*</regular-
expression><identifier>Arbol</identifier></extractor-pattern-tokens><script-
instances><script-instances when-to-run="80" sequence="1"
enabled="true"><script><script-text>Ruta="C:/Users/Viridiana/Documents/Screen-
Scraper/";
//Ruta= session.getVariable("GuardarEn");
Archivo="Productos.csv";


CsvWriter file = new CsvWriter(Ruta+Archivo);

String[] header = {"ID_PRODUCTO", "PRODUCTO"};
file.setHeader(header);
HashMap hm = new HashMap();
hm.put("ID_PRODUCTO", dataRecord.get("ID_PRODUCTO"));
hm.put("PRODUCTO", dataRecord.get("PRODUCTO"));
file.write(hm);
file.flush();
file.close();</script-
text><name>PROFECO_Actualizaciones_Productos</name><language>Interpreted
```

```
Java</language></script></script-instances><owner-type>ExtractorPattern</owner-
type><owner-name>Untitled Extractor Pattern</owner-name></script-
instances></extractor-patterns><script-instances><owner-type>ScrapeableFile</owner-
type><owner-name>Lista productos</owner-name></script-instances></scrapeable-
files><scrapeable-files sequence="2" will-be-invoked-manually="false" tidy-
html="jtidy"><last-scraped-data></last-scraped-
data><URL>http://www.profeco.gob.mx/precios/canasta/homer.aspx</URL><last-
request></last-request><name>Elegir ciudad</name><extractor-patterns sequence="1"
automatically-save-in-session-variable="false" if-saved-in-session-variable="0"
filter-duplicates="false" cache-data-set="false" will-be-invoked-
manually="false"><pattern-text>name="__VIEWSTATEGENERATOR" id="__VIEWSTATEGENERATOR"
value="~@ASPX_VIEWSTATEGENERATOR@~"</pattern-text><identifier>Untitled Extractor
Pattern</identifier><extractor-pattern-tokens optional="false" save-in-session-
variable="true" compound-key="true" strip-html="false" resolve-relative-url="false"
replace-html-entities="false" trim-white-space="false" exclude-from-data="false" null-
session-variable="false" sequence="1"><regular-expression>[^"]*</regular-
expression><identifier>ASPX_VIEWSTATEGENERATOR</identifier></extractor-pattern-
tokens><script-instances><owner-type>ExtractorPattern</owner-type><owner-name>Untitled
Extractor Pattern</owner-name></script-instances></extractor-patterns><HTTPParameters
sequence="1"><key>__EVENTTARGET</key><type>POST</type><value>cmbCiudad</value></HTTPPa
rameters><HTTPParameters
sequence="2"><key>__EVENTARGUMENT</key><type>POST</type><value>~#ASPX_EVENTARGUMENT#~<
/value></HTTPParameters><HTTPParameters
sequence="3"><key>__LASTFOCUS</key><type>POST</type><value></value></HTTPParameters><H
TTPParameters
sequence="7"><key>cmbCiudad</key><type>POST</type><value>150901</value></HTTPParameter
s><HTTPParameters
sequence="6"><key>__EVENTVALIDATION</key><type>POST</type><value>~#ASPX_EVENTVALIDATIO
N#~</value></HTTPParameters><HTTPParameters
sequence="5"><key>__VIEWSTATEGENERATOR</key><type>POST</type><value>~#ASPX_VIEWSTATEGE
NERATOR#~</value></HTTPParameters><HTTPParameters
sequence="4"><key>__VIEWSTATE</key><type>POST</type><value>~#ASPX_VIEWSTATE#~</value><
/HTTPParameters><script-instances><script-instances when-to-run="40" sequence="1"
enabled="true"><script><script-text>DataRecord aspx =
scrapeableFile.getASPXValues(true);
session.setVariables(aspx);</script-text><name>ASPX</name><language>Interpreted
Java</language></script></script-instances><owner-type>ScrapeableFile</owner-
type><owner-name>Elegir ciudad</owner-name></script-instances></scrapeable-
files><scrapeable-files sequence="3" will-be-invoked-manually="false" tidy-
html="jtidy"><last-scraped-data></last-scraped-
data><URL>http://www.profeco.gob.mx/precios/canasta/homer.aspx</URL><last-
request></last-request><name>Elegir municipio</name><HTTPParameters
sequence="5"><key>__VIEWSTATEGENERATOR</key><type>POST</type><value>~#ASPX_VIEWSTATEGE
NERATOR#~</value></HTTPParameters><HTTPParameters
sequence="4"><key>__VIEWSTATE</key><type>POST</type><value>~#ASPX_VIEWSTATE#~</value><
/HTTPParameters><HTTPParameters
sequence="8"><key>listaMunicipios</key><type>POST</type><value>1509010</value></HTTPPa
rameters><HTTPParameters
sequence="9"><key>ImageButton1.x</key><type>POST</type><value>18</value></HTTPParamete
rs><HTTPParameters
sequence="3"><key>__LASTFOCUS</key><type>POST</type><value></value></HTTPParameters><H
TTPParameters
sequence="2"><key>__EVENTARGUMENT</key><type>POST</type><value>~#ASPX_EVENTARGUMENT#~<
```

```
/value></HTTPParameters><HTTPParameters
sequence="10"><key>ImageButton1.y</key><type>POST</type><value>9</value></HTTPParamete
rs><HTTPParameters
sequence="7"><key>cmbCiudad</key><type>POST</type><value>150901</value></HTTPParameter
s><HTTPParameters
sequence="6"><key>__EVENTVALIDATION</key><type>POST</type><value>~#ASPX_EVENTVALIDATIO
N#~</value></HTTPParameters><HTTPParameters
sequence="1"><key>__EVENTTARGET</key><type>POST</type><value>~#ASPX_EVENTTARGET#~</val
ue></HTTPParameters><script-instances><owner-type>ScrapeableFile</owner-type><owner-
name>Elegir municipio</owner-name></script-instances></scrapeable-files><scrapeable-
files sequence="1" will-be-invoked-manually="false" tidy-html="jtidy"><last-scraped-
data></last-scraped-
data><URL>http://www.profeco.gob.mx/precios/canasta/homer.aspx</URL><last-
request></last-request><name>Lista ciudades</name><extractor-patterns sequence="1"
automatically-save-in-session-variable="false" if-saved-in-session-variable="0"
filter-duplicates="false" cache-data-set="false" will-be-invoked-
manually="false"><pattern-text>&lt;option
value="~@ID_CIUDAD@~"&gt;~@CIUDAD@~&lt;/option&gt;&#xd;</pattern-
text><identifier>Untitled Extractor Pattern</identifier><extractor-pattern-tokens
optional="false" save-in-session-variable="false" compound-key="true" strip-
html="true" resolve-relative-url="false" replace-html-entities="true" trim-white-
space="false" exclude-from-data="false" null-session-variable="false"
sequence="2"><regular-expression>[^&lt;&gt;]*</regular-
expression><identifier>CIUDAD</identifier></extractor-pattern-tokens><extractor-
pattern-tokens optional="false" save-in-session-variable="false" compound-key="true"
strip-html="false" resolve-relative-url="false" replace-html-entities="false" trim-
white-space="false" exclude-from-data="false" null-session-variable="false"
sequence="1"><regular-expression>[^"]*</regular-
expression><identifier>ID_CIUDAD</identifier></extractor-pattern-tokens><script-
instances><script-instances when-to-run="80" sequence="1"
enabled="false"><script><script-text>Ruta="C:/Users/Viridiana/Documents/Screen-
Scraper/";
//Ruta= session.getVariable("GuardarEn");
Archivo="Ciudades.csv";


CsvWriter file = new CsvWriter(Ruta+Archivo);

String[] header = {"ID_CIUDAD", "CIUDAD","IMAGEBUTTON1.X","IMAGEBUTTON1.Y"};
file.setHeader(header);
HashMap hm = new HashMap();
hm.put("ID_CIUDAD", dataRecord.get("ID_CIUDAD"));
hm.put("CIUDAD", dataRecord.get("CIUDAD"));
hm.put("IMAGEBUTTON1.X", "0");
hm.put("IMAGEBUTTON1.Y", "0");
file.write(hm);
file.flush();
file.close();
</script-text><name>PROFECO_Actualizaciones_Ciudades</name><language>Interpreted
Java</language></script></script-instances><owner-type>ExtractorPattern</owner-
type><owner-name>Untitled Extractor Pattern</owner-name></script-
instances></extractor-patterns><extractor-patterns sequence="2" automatically-save-in-
session-variable="false" if-saved-in-session-variable="0" filter-duplicates="false"
```

```
cache-data-set="false" will-be-invoked-manually="false"><pattern-
text>name="__VIEWSTATEGENERATOR" id="__VIEWSTATEGENERATOR"
value="~@ASPX_VIEWSTATEGENERATOR@~"</pattern-text><identifier>Untitled Extractor
Pattern</identifier><extractor-pattern-tokens optional="false" save-in-session-
variable="true" compound-key="true" strip-html="false" resolve-relative-url="false"
replace-html-entities="false" trim-white-space="false" exclude-from-data="false" null-
session-variable="false" sequence="1"><regular-expression>[^"]*</regular-
expression><identifier>ASPX_VIEWSTATEGENERATOR</identifier></extractor-pattern-
tokens><script-instances><owner-type>ExtractorPattern</owner-type><owner-name>Untitled
Extractor Pattern</owner-name></script-instances></extractor-patterns><script-
instances><script-instances when-to-run="40" sequence="1"
enabled="true"><script><script-text>DataRecord aspx =
scrapeableFile.getASPXValues(true);
session.setVariables(aspx);</script-text><name>ASPX</name><language>Interpreted
Java</language></script></script-instances><owner-type>ScrapeableFile</owner-
type><owner-name>Lista ciudades</owner-name></script-instances></scrapeable-
files></scraping-session>
```

Obtención y escritura de precios

```
<?xml version="1.0" encoding="UTF-8"?>
<scraping-session use-strict-mode="true"><script-instances><script-instances when-to-
run="20" sequence="1" enabled="true"><script><script-text>session.setVariable("Fecha",
sutil.getCurrentDate("yyyyMMdd"));</script-
text><name>Fecha</name><language>Interpreted Java</language></script></script-
instances><script-instances when-to-run="20" sequence="2"
enabled="true"><script><script-text>session.logInfo( "Cargando datos..." );

//session.loadVariables("C:/Users/Viridiana/Documents/Screen-
Scraper/Datos/PROFECO_Archivos.txt");
//session.loadVariables("C:/Users/Viridiana/Documents/Screen-
Scraper/Datos/Walmart_Archivos.txt");
session.loadVariables(session.getv("ruta"));
session.logVariables();</script-text><name>CargaDatos</name><language>Interpreted
Java</language></script></script-instances><script-instances when-to-run="20"
sequence="3" enabled="true"><script><script-text>String[] parseCSVLine(String line,
int index, int columnsToGet){
    int START_STATE = 0;
    int FIRST_QUOTE = 1;
    int SECOND_QUOTE = 2;
    int IN_WORD = 3;
    int IN_WORD_WITHOUT_QUOTES = 4;
    int state = START_STATE;
    String word = "";
    ArrayList lines = new ArrayList();
    char[] chars = line.toCharArray();

     for (int i = 0; i &lt; chars.length; i++){
        char c = chars[i];

        if (c == '"'){
```

```
            if (state == START_STATE){
                state = FIRST_QUOTE;
            }
            else if ((state == FIRST_QUOTE) || (state == IN_WORD)){
                state = SECOND_QUOTE;
            }
            else if (state == SECOND_QUOTE){
                word += ("" + c);
                state = IN_WORD;
            }
        }
        else if (c == ','){
            if ((state == SECOND_QUOTE) || (state == IN_WORD_WITHOUT_QUOTES)){
                state = START_STATE;

                lines.add(word);
                if (lines.size() == columnsToGet) break;
                    word = "";
            }
            else if (state == START_STATE){
                state = START_STATE;
                lines.add(word.replaceAll("\"\"", "\""));
            }
            else{
                word += ("" + c);
                state = IN_WORD;
            }
        }
        else{
            if (state == START_STATE) state = IN_WORD_WITHOUT_QUOTES;
            else if (state != IN_WORD_WITHOUT_QUOTES){
                 state = IN_WORD;
                word += ("" + c);
            }
        }
    }
    if (lines.size() &lt; columnsToGet){
        if ((state == SECOND_QUOTE) || (state == IN_WORD_WITHOUT_QUOTES))
             lines.add(word.replaceAll("\"\"", "\""));
    }
    String[] linesArray = new String[lines.size()];

    for (int i = 0; i &lt; lines.size(); i++){
        linesArray[i] = (String) lines.get(i);
    }

    return linesArray;
}

ciudades = session.getVariable("Ciudades");
// File from which to read.
File inputFile = new File(ciudades);
```

```
FileReader in = new FileReader( inputFile );
BufferedReader buffRead = new BufferedReader( in );

// Read the file in line-by-line.
int index = 0;
while( ( searchTerm = buffRead.readLine() )!=null){
    // Don't read header row
    if (index&gt;0){
        // Parse the line into an array
        line = parseCSVLine(searchTerm, index, 4);

        // Get the values
        idC = line[0];
        nombreC = line[1];
        bx = line[2];
        by = line[3];


        // Set the needed values as session vaiables
        session.setVariable("ID_CIUDAD", idC);
        session.setVariable("ID_MUNICIPIO", idC+"0");
        session.setVariable("IMAGEBUTTON1.X", bx);
        session.setVariable("IMAGEBUTTON1.Y", by);
        session.setVariable("CIUDAD", nombreC);

        // Scrape for those values
        log.logVariables();
        //session.executeScriptWithContext("PROFECO_CargaProductos");

    }
    index++;
}

// Close up the file.
in.close();
buffRead.close();</script-text><name>PROFECO_CargaCiudades</name><language>Interpreted
Java</language></script></script-instances><owner-type>ScrapingSession</owner-
type><owner-name>PROFECO</owner-name></script-
instances><name>PROFECO</name><notes></notes><cookiePolicy>0</cookiePolicy><maxHTTPReq
uests>1</maxHTTPRequests><external_proxy_username></external_proxy_username><external_
proxy_password></external_proxy_password><external_proxy_host></external_proxy_host><e
xternal_proxy_port></external_proxy_port><external_nt_proxy_username></external_nt_pro
xy_username><external_nt_proxy_password></external_nt_proxy_password><external_nt_prox
y_domain></external_nt_proxy_domain><external_nt_proxy_host></external_nt_proxy_host><
anonymize>false</anonymize><terminate_proxies_on_completion>false</terminate_proxies_o
n_completion><number_of_required_proxies>5</number_of_required_proxies><originator_edi
tion>2</originator_edition><logging_level>1</logging_level><date_exported>febrero 11,
2016 15:52:57</date_exported><character_set>UTF-
8</character_set><created_by_version>6.0</created_by_version><scrapeable-files
sequence="-1" will-be-invoked-manually="true" tidy-html="jtidy"><last-scraped-
data></last-scraped-
data><URL>http://www.profeco.gob.mx/precios/canasta/homer.aspx</URL><last-
request></last-request><name>Obtener_ASPX</name><extractor-patterns sequence="1"
```

```
automatically-save-in-session-variable="false" if-saved-in-session-variable="0"
filter-duplicates="false" cache-data-set="false" will-be-invoked-
manually="false"><pattern-text>name="__VIEWSTATEGENERATOR" id="__VIEWSTATEGENERATOR"
value="~@ASPX_VIEWSTATEGENERATOR@~"</pattern-text><identifier>Untitled Extractor
Pattern</identifier><extractor-pattern-tokens optional="false" save-in-session-
variable="true" compound-key="true" strip-html="false" resolve-relative-url="false"
replace-html-entities="false" trim-white-space="false" exclude-from-data="false" null-
session-variable="false" sequence="1"><regular-expression>[^"]*</regular-
expression><identifier>ASPX_VIEWSTATEGENERATOR</identifier></extractor-pattern-
tokens><script-instances><owner-type>ExtractorPattern</owner-type><owner-name>Untitled
Extractor Pattern</owner-name></script-instances></extractor-patterns><script-
instances><script-instances when-to-run="40" sequence="1"
enabled="true"><script><script-text>DataRecord aspx =
scrapeableFile.getASPXValues(true);
session.setVariables(aspx);</script-text><name>ASPX</name><language>Interpreted
Java</language></script></script-instances><owner-type>ScrapeableFile</owner-
type><owner-name>Obtener_ASPX</owner-name></script-instances></scrapeable-
files><scrapeable-files sequence="-1" will-be-invoked-manually="true" tidy-
html="jtidy"><last-scraped-data></last-scraped-
data><URL>http://www.profeco.gob.mx/precios/canasta/homer.aspx</URL><last-
request></last-request><name>Municipio</name><HTTPParameters
sequence="7"><key>cmbCiudad</key><type>POST</type><value>~#ID_CIUDAD#~</value></HTTPPa
rameters><HTTPParameters
sequence="2"><key>__EVENTARGUMENT</key><type>POST</type><value>~#ASPX_EVENTARGUMENT#~<
/value></HTTPParameters><HTTPParameters
sequence="5"><key>__VIEWSTATEGENERATOR</key><type>POST</type><value>~#ASPX_VIEWSTATEGE
NERATOR#~</value></HTTPParameters><HTTPParameters
sequence="1"><key>__EVENTTARGET</key><type>POST</type><value>~#ASPX_EVENTTARGET#~</val
ue></HTTPParameters><HTTPParameters
sequence="4"><key>__VIEWSTATE</key><type>POST</type><value>~#ASPX_VIEWSTATE#~</value><
/HTTPParameters><HTTPParameters
sequence="8"><key>listaMunicipios</key><type>POST</type><value>~#ID_MUNICIPIO#~</value
></HTTPParameters><HTTPParameters
sequence="3"><key>__LASTFOCUS</key><type>POST</type><value></value></HTTPParameters><H
TTPParameters
sequence="6"><key>__EVENTVALIDATION</key><type>POST</type><value>~#ASPX_EVENTVALIDATIO
N#~</value></HTTPParameters><HTTPParameters
sequence="9"><key>ImageButton1.x</key><type>POST</type><value>~#IMAGEBUTTON1.X#~</valu
e></HTTPParameters><HTTPParameters
sequence="10"><key>ImageButton1.y</key><type>POST</type><value>~#IMAGEBUTTON1.Y#~</val
ue></HTTPParameters><script-instances><owner-type>ScrapeableFile</owner-type><owner-
name>Municipio</owner-name></script-instances></scrapeable-files><scrapeable-files
sequence="-1" will-be-invoked-manually="true" tidy-html="jtidy"><last-scraped-
data></last-scraped-
data><URL>http://www.profeco.gob.mx/precios/canasta/listaProdArbol.aspx</URL><last-
request></last-request><name>Obtener_precios</name><extractor-patterns sequence="1"
automatically-save-in-session-variable="false" if-saved-in-session-variable="0"
filter-duplicates="false" cache-data-set="false" will-be-invoked-
manually="false"><pattern-text>&lt;td align="left" valign="middle"&gt;&lt;a
id="~@id@~"
href="listaEstProd.aspx?cve_prodmarca=~@cve_prodmarca@~"&gt;~@producto@~,~@marca@~,~@d
escripcion@~&lt;/a&gt; &lt;/td&gt;&#xd;
&lt;td align="center" valign="middle"&gt;~@min@~&lt;/td&gt;&#xd;
```

```
&lt;td align="center" valign="middle"&gt;~@max@~&lt;/td&gt;&#xd;
&lt;td align="center" valign="middle"&gt;~@prom@~&lt;/td&gt;&#xd;
&lt;/tr&gt;&#xd;
</pattern-text><identifier>Extraer_precios</identifier><extractor-pattern-tokens
optional="false" save-in-session-variable="true" compound-key="true" strip-html="true"
resolve-relative-url="false" replace-html-entities="true" trim-white-space="false"
exclude-from-data="false" null-session-variable="false" sequence="4"><regular-
expression></regular-expression><identifier>marca</identifier></extractor-pattern-
tokens><extractor-pattern-tokens optional="false" save-in-session-variable="false"
compound-key="true" strip-html="false" resolve-relative-url="false" replace-html-
entities="false" trim-white-space="false" exclude-from-data="true" null-session-
variable="false" sequence="1"><regular-expression>[^"]*</regular-
expression><identifier>id</identifier></extractor-pattern-tokens><extractor-pattern-
tokens optional="false" save-in-session-variable="false" compound-key="true" strip-
html="false" resolve-relative-url="false" replace-html-entities="false" trim-white-
space="false" exclude-from-data="true" null-session-variable="false"
sequence="7"><regular-expression>[^&lt;&gt;]*</regular-
expression><identifier>max</identifier></extractor-pattern-tokens><extractor-pattern-
tokens optional="false" save-in-session-variable="true" compound-key="true" strip-
html="false" resolve-relative-url="false" replace-html-entities="false" trim-white-
space="false" exclude-from-data="false" null-session-variable="false"
sequence="2"><regular-expression>[^"]*</regular-
expression><identifier>cve_prodmarca</identifier></extractor-pattern-
tokens><extractor-pattern-tokens optional="false" save-in-session-variable="true"
compound-key="true" strip-html="true" resolve-relative-url="false" replace-html-
entities="true" trim-white-space="false" exclude-from-data="false" null-session-
variable="false" sequence="5"><regular-expression></regular-
expression><identifier>descripcion</identifier></extractor-pattern-tokens><extractor-
pattern-tokens optional="false" save-in-session-variable="true" compound-key="true"
strip-html="true" resolve-relative-url="false" replace-html-entities="true" trim-
white-space="false" exclude-from-data="false" null-session-variable="false"
sequence="3"><regular-expression></regular-
expression><identifier>producto</identifier></extractor-pattern-tokens><extractor-
pattern-tokens optional="false" save-in-session-variable="false" compound-key="true"
strip-html="false" resolve-relative-url="false" replace-html-entities="false" trim-
white-space="false" exclude-from-data="true" null-session-variable="false"
sequence="6"><regular-expression>[^&lt;&gt;]*</regular-
expression><identifier>min</identifier></extractor-pattern-tokens><extractor-pattern-
tokens optional="false" save-in-session-variable="false" compound-key="true" strip-
html="false" resolve-relative-url="false" replace-html-entities="false" trim-white-
space="false" exclude-from-data="true" null-session-variable="false"
sequence="8"><regular-expression>[^&lt;&gt;]*</regular-
expression><identifier>prom</identifier></extractor-pattern-tokens><script-
instances><script-instances when-to-run="80" sequence="1"
enabled="true"><script><script-text>sutil.randomPause(0, 500);
session.scrapeFile( "Obtener_fuente" );
session.clearCookies();</script-
text><name>PROFECO_ConsultaEstablecimiento</name><language>Interpreted
Java</language></script></script-instances><owner-type>ExtractorPattern</owner-
type><owner-name>Extraer_precios</owner-name></script-instances></extractor-
patterns><HTTPParameters
sequence="1"><key>codigo</key><type>GET</type><value>~#ID_PRODUCTO#~</value></HTTPPara
meters><script-instances><owner-type>ScrapeableFile</owner-type><owner-
```

```
name>Obtener_precios</owner-name></script-instances></scrapeable-files><scrapeable-
files sequence="-1" will-be-invoked-manually="true" tidy-html="jtidy"><last-scraped-
data></last-scraped-
data><URL>http://www.profeco.gob.mx/precios/canasta/homer.aspx</URL><last-
request></last-request><name>Ciudad</name><extractor-patterns sequence="1"
automatically-save-in-session-variable="false" if-saved-in-session-variable="0"
filter-duplicates="false" cache-data-set="false" will-be-invoked-
manually="false"><pattern-text>name="__VIEWSTATEGENERATOR" id="__VIEWSTATEGENERATOR"
value="~@ASPX_VIEWSTATEGENERATOR@~"</pattern-text><identifier>Untitled Extractor
Pattern</identifier><extractor-pattern-tokens optional="false" save-in-session-
variable="true" compound-key="true" strip-html="false" resolve-relative-url="false"
replace-html-entities="false" trim-white-space="false" exclude-from-data="false" null-
session-variable="false" sequence="1"><regular-expression>[^"]*</regular-
expression><identifier>ASPX_VIEWSTATEGENERATOR</identifier></extractor-pattern-
tokens><script-instances><owner-type>ExtractorPattern</owner-type><owner-name>Untitled
Extractor Pattern</owner-name></script-instances></extractor-patterns><HTTPParameters
sequence="7"><key>cmbCiudad</key><type>POST</type><value>~#ID_CIUDAD#~</value></HTTPPa
rameters><HTTPParameters
sequence="6"><key>__EVENTVALIDATION</key><type>POST</type><value>~#ASPX_EVENTVALIDATIO
N#~</value></HTTPParameters><HTTPParameters
sequence="5"><key>__VIEWSTATEGENERATOR</key><type>POST</type><value>~#ASPX_VIEWSTATEGE
NERATOR#~</value></HTTPParameters><HTTPParameters
sequence="1"><key>__EVENTTARGET</key><type>POST</type><value>cmbCiudad</value></HTTPPa
rameters><HTTPParameters
sequence="2"><key>__EVENTARGUMENT</key><type>POST</type><value>~#ASPX_EVENTARGUMENT#~<
/value></HTTPParameters><HTTPParameters
sequence="4"><key>__VIEWSTATE</key><type>POST</type><value>~#ASPX_VIEWSTATE#~</value><
/HTTPParameters><HTTPParameters
sequence="3"><key>__LASTFOCUS</key><type>POST</type><value></value></HTTPParameters><s
cript-instances><script-instances when-to-run="40" sequence="1"
enabled="true"><script><script-text>DataRecord aspx =
scrapeableFile.getASPXValues(true);
session.setVariables(aspx);</script-text><name>ASPX</name><language>Interpreted
Java</language></script></script-instances><owner-type>ScrapeableFile</owner-
type><owner-name>Ciudad</owner-name></script-instances></scrapeable-files><scrapeable-
files sequence="-1" will-be-invoked-manually="true" tidy-html="jtidy"><last-scraped-
data></last-scraped-
data><URL>http://www.profeco.gob.mx/precios/canasta/listaEstProd.aspx</URL><last-
request></last-request><name>Obtener_fuente</name><extractor-patterns sequence="1"
automatically-save-in-session-variable="false" if-saved-in-session-variable="0"
filter-duplicates="false" cache-data-set="false" will-be-invoked-
manually="false"><pattern-text>&lt;td align="left" valign="middle"&gt;&lt;a
href="fichaEst.aspx?est_folio=~@folio@~"&gt;~@establecimiento@~&lt;/a&gt;~@espacio@~&l
t;/td&gt;&#xd;
&lt;td align="center" valign="middle"&gt;~@precio@~&lt;/td&gt;&#xd;
&lt;td align="center" valign="middle"&gt;~@fechaObs@~&lt;/td&gt;&#xd;
</pattern-text><identifier>Untitled Extractor Pattern</identifier><extractor-pattern-
tokens optional="false" save-in-session-variable="true" compound-key="true" strip-
html="true" resolve-relative-url="false" replace-html-entities="true" trim-white-
space="false" exclude-from-data="false" null-session-variable="false"
sequence="2"><regular-expression>[^&lt;&gt;]*</regular-
expression><identifier>establecimiento</identifier></extractor-pattern-
tokens><extractor-pattern-tokens optional="false" save-in-session-variable="false"
```

```
compound-key="true" strip-html="false" resolve-relative-url="false" replace-html-
entities="false" trim-white-space="false" exclude-from-data="true" null-session-
variable="false" sequence="1"><regular-expression>[\d,]+</regular-
expression><identifier>folio</identifier></extractor-pattern-tokens><extractor-
pattern-tokens optional="false" save-in-session-variable="false" compound-key="true"
strip-html="false" resolve-relative-url="false" replace-html-entities="false" trim-
white-space="false" exclude-from-data="true" null-session-variable="false"
sequence="3"><regular-expression></regular-
expression><identifier>espacio</identifier></extractor-pattern-tokens><extractor-
pattern-tokens optional="false" save-in-session-variable="true" compound-key="true"
strip-html="true" resolve-relative-url="false" replace-html-entities="true" trim-
white-space="false" exclude-from-data="false" null-session-variable="false"
sequence="4"><regular-expression>[^&lt;&gt;]*</regular-
expression><identifier>precio</identifier></extractor-pattern-tokens><extractor-
pattern-tokens optional="false" save-in-session-variable="true" compound-key="true"
strip-html="true" resolve-relative-url="false" replace-html-entities="true" trim-
white-space="false" exclude-from-data="false" null-session-variable="false"
sequence="5"><regular-expression>[^&lt;&gt;]*</regular-
expression><identifier>fechaObs</identifier></extractor-pattern-tokens><script-
instances><script-instances when-to-run="80" sequence="1"
enabled="true"><script><script-text>session.logInfo( "Obteniendo precios..." );

Ruta="C:/Users/Viridiana/Documents/Screen-Scraper/Precios/PROFECO/";
//Ruta= session.getVariable("GuardarEn");
Archivo="Profeco"+session.getVariable("Fecha")+".csv";


CsvWriter head = new CsvWriter(Ruta+Archivo);

String[] header = {"id producto", "producto", "id marca", "marca", "descripcion",
"precio", "establecimiento", "fecha de observación", "ciudad"};
head.setHeader(header);
head.close();
String[] c= {
"\""+session.getv("ID_PRODUCTO")+"\"",
"\""+session.getv("producto")+"\"",
"\""+session.getv("cve_prodmarca")+"\"",
"\""+session.getv("marca")+"\"",
"\""+session.getv("descripcion")+"\"",
"\""+session.getv("precio")+"\"",
"\""+session.getv("establecimiento")+"\"",
"\""+session.getv("fechaObs")+"\"",
"\""+session.getv("CIUDAD")+"\"",
};
FileWriter out = null;
try
{
	out = new FileWriter(Ruta+Archivo,true);

	for (int k = 0; k &lt; c.length; k++)
	{
		out.write( c[k]+",");
	}
```

```
	out.write( "\n" );


	out.close();
}
catch( Exception e )
{
	log.logError("Error al crear archivo: " + e.getMessage() );
}</script-text><name>PROFECO_EscribePrecios</name><language>Interpreted
Java</language></script></script-instances><owner-type>ExtractorPattern</owner-
type><owner-name>Untitled Extractor Pattern</owner-name></script-
instances></extractor-patterns><HTTPParameters
sequence="1"><key>cve_prodmarca</key><type>GET</type><value>~#cve_prodmarca#~</value><
/HTTPParameters><script-instances><owner-type>ScrapeableFile</owner-type><owner-
name>Obtener_fuente</owner-name></script-instances></scrapeable-files><execute-
scripts-to-export><script-text>String[] parseCSVLine(String line, int index, int
columnsToGet){
    int START_STATE = 0;
    int FIRST_QUOTE = 1;
    int SECOND_QUOTE = 2;
    int IN_WORD = 3;
    int IN_WORD_WITHOUT_QUOTES = 4;
    int state = START_STATE;
    String word = "";
    ArrayList lines = new ArrayList();
    char[] chars = line.toCharArray();

     for (int i = 0; i &lt; chars.length; i++){
        char c = chars[i];

        if (c == '"'){
            if (state == START_STATE){
                state = FIRST_QUOTE;
            }
            else if ((state == FIRST_QUOTE) || (state == IN_WORD)){
                state = SECOND_QUOTE;
            }
            else if (state == SECOND_QUOTE){
                word += ("" + c);
                state = IN_WORD;
            }
        }
        else if (c == ','){
            if ((state == SECOND_QUOTE) || (state == IN_WORD_WITHOUT_QUOTES)){
                state = START_STATE;

                lines.add(word);
                if (lines.size() == columnsToGet) break;
                    word = "";
            }
            else if (state == START_STATE){
                state = START_STATE;
                lines.add(word.replaceAll("\"\"", "\""));
```

```
            }
            else{
                word += ("" + c);
                state = IN_WORD;
            }
        }
        else{
            if (state == START_STATE) state = IN_WORD_WITHOUT_QUOTES;
            else if (state != IN_WORD_WITHOUT_QUOTES){
                 state = IN_WORD;
                word += ("" + c);
            }
        }
    }
    if (lines.size() &lt; columnsToGet){
        if ((state == SECOND_QUOTE) || (state == IN_WORD_WITHOUT_QUOTES))
             lines.add(word.replaceAll("\"\"", "\""));
    }
    String[] linesArray = new String[lines.size()];

    for (int i = 0; i &lt; lines.size(); i++){
        linesArray[i] = (String) lines.get(i);
    }

    return linesArray;
}

// File from which to read.
productos=session.getVariable("Productos");
File inputFile = new File( productos);

FileReader in = new FileReader( inputFile );
BufferedReader buffRead = new BufferedReader( in );

// Read the file in line-by-line.
int index = 0;
while( ( searchTerm = buffRead.readLine() )!=null){
    // Don't read header row
    if (index&gt;0){
        // Parse the line into an array
        line = parseCSVLine(searchTerm, index, 1);

        // Get the values
        idP = line[0];


        // Set the needed values as session vaiables
        session.setVariable("ID_PRODUCTO", idP);

        // Scrape for those values
        session.scrapeFile("Obtener_ASPX");
        session.scrapeFile("Ciudad");
        session.scrapeFile("Municipio");
```

```
        session.scrapeFile("Obtener_Precios");
    }
    index++;
}

// Close up the file.
in.close();
buffRead.close();</script-
text><name>PROFECO_CargaProductos</name><language>Interpreted
Java</language></execute-scripts-to-export></scraping-session>
```

Códigos de bats

Comercial Mexicana®

Bajar precios

```
C:
cd "C:\Program Files\screen-scraper Enterprise Edition"
"C:\Program Files\screen-scraper Enterprise Edition\jre\bin\java" -jar screen-
scraper.jar -s "Comercial_Mexicana" -p
"ruta=C:/Users/Viridiana/Documents/Datos/ComercialMexicana_GranSur.txt"
>"log\ComercialMexicana_GranSur.log"
```

Walmart®

Actualizar departamentos y bajar precios

```
C:
cd "C:\Program Files\screen-scraper Enterprise Edition"
"C:\Program Files\screen-scraper Enterprise Edition\jre\bin\java" -jar screen-
scraper.jar -s "Walmart_actualizaciones">"log\Walmart_actualizaciones.log"
cd "C:\Users\Viridiana\Documents\Screen-Scraper"
copy WalmartDepartamentos.csv "C:\Users\Viridiana\Documents\Screen-
Scraper\Precios\Walmart"
copy WalmartDepartamentos.csv "C:\Users\Viridiana\Documents\Screen-Scraper\Datos"
del WalmartDepartamentos.csv
cd "C:\Program Files\screen-scraper Enterprise Edition"
"C:\Program Files\screen-scraper Enterprise Edition\jre\bin\java" -jar screen-
scraper.jar -s "Walmart" -p "ruta=C:/Users/Viridiana/Documents/Screen-
Scraper/Datos/Walmart_Archivos.txt" >"log\Walmart.log"
```

PROFECO

Actualizar la lista de productos.

```
C:
cd "C:\Program Files\screen-scraper Enterprise Edition"
"C:\Program Files\screen-scraper Enterprise Edition\jre\bin\java" -jar screen-
scraper.jar -s "PROFECO_Actualizaciones">"log\PROFECO_actualizaciones.log"
cd "C:\Users\Viridiana\Documents\Screen-Scraper"
```

```
copy Productos.csv "C:\Users\Viridiana\Documents\Screen-Scraper\Precios\PROFECO"
copy Productos.csv "C:\Users\Viridiana\Documents\Screen-Scraper\Datos"
del Productos.csv
```

Bajar precios

```
C:
cd "C:\Program Files\screen-scraper Enterprise Edition"
"C:\Program Files\screen-scraper Enterprise Edition\jre\bin\java" -jar screen-
scraper.jar -s "PROFECO" -p "ruta=C:/Users/Viridiana/Documents/Screen-
Scraper/Datos/PROFECO_Archivos.txt" >"log\PROFECO.log"
```

Actualizar en el repositorio

Mover los archivos descargados a la carpeta de GIT

```
cd "C:\Users\Viridiana\Documents\Screen-Scraper\Precios\PROFECO"
move *.csv "C:\Users\Viridiana\Documents\GIT\data\PROFECO"
cd "C:\Users\Viridiana\Documents\Screen-Scraper\Precios\Walmart"
move *.csv "C:\Users\Viridiana\Documents\GIT\data\Walmart"
```

Actualizar GIT

```
cd "C:\Users\Viridiana\Documents\GIT\data\PROFECO"
git add *
cd "C:\Users\Viridiana\Documents\GIT\data\Walmart"
git add *
git commit -m "Se agregan archivos nuevos"
git push origin master
exit
```

Iniciar la consola de GIT y ejecutar actualizaciones

```
"C:\Program Files\Git\git-cmd.exe" --cd-to-home "C:\Users\Viridiana\Documents\Screen-
Scraper\Bats\ActualizarGIT.bat"
```

Código de tareas programadas en Windows

Al exportar la tarea de Windows, se generó el siguiente código.

```
<?xml version="1.0" encoding="UTF-16"?>
<Task version="1.2" xmlns="http://schemas.microsoft.com/windows/2004/02/mit/task">
  <RegistrationInfo>
    <Date>2016-01-04T20:08:16.0582791</Date>
    <Author>thini7\Viridiana</Author>
```

```
    <Description>Proceso para obtener precios de la página de PROFECO.</Description>
  </RegistrationInfo>
  <Triggers>
    <CalendarTrigger>
      <StartBoundary>2016-01-05T06:07:00Z</StartBoundary>
      <Enabled>true</Enabled>
      <ScheduleByDay>
        <DaysInterval>1</DaysInterval>
      </ScheduleByDay>
    </CalendarTrigger>
  </Triggers>
  <Principals>
    <Principal id="Author">
      <UserId>thini7\Viridiana</UserId>
      <LogonType>InteractiveToken</LogonType>
      <RunLevel>HighestAvailable</RunLevel>
    </Principal>
  </Principals>
  <Settings>
    <MultipleInstancesPolicy>IgnoreNew</MultipleInstancesPolicy>
    <DisallowStartIfOnBatteries>false</DisallowStartIfOnBatteries>
    <StopIfGoingOnBatteries>false</StopIfGoingOnBatteries>
    <AllowHardTerminate>false</AllowHardTerminate>
    <StartWhenAvailable>false</StartWhenAvailable>
    <RunOnlyIfNetworkAvailable>false</RunOnlyIfNetworkAvailable>
    <IdleSettings>
      <StopOnIdleEnd>true</StopOnIdleEnd>
      <RestartOnIdle>false</RestartOnIdle>
    </IdleSettings>
    <AllowStartOnDemand>true</AllowStartOnDemand>
    <Enabled>true</Enabled>
    <Hidden>false</Hidden>
    <RunOnlyIfIdle>false</RunOnlyIfIdle>
    <WakeToRun>true</WakeToRun>
    <ExecutionTimeLimit>PT0S</ExecutionTimeLimit>
    <Priority>7</Priority>
    <RestartOnFailure>
      <Interval>PT1M</Interval>
      <Count>50</Count>
    </RestartOnFailure>
  </Settings>
  <Actions Context="Author">
    <Exec>
      <Command>"C:\Program Files\screen-scraper Enterprise
Edition\procesos\PROFECO_actualizaciones.bat"</Command>
    </Exec>
    <Exec>
      <Command>"C:\Program Files\screen-scraper Enterprise
Edition\procesos\PROFECO.bat"</Command>
    </Exec>
    <Exec>
      <Command>"C:\Program Files\screen-scraper Enterprise
Edition\procesos\Mover_Archivos.bat"</Command>
```

```
    </Exec>
    <Exec>
      <Command>"C:\Program Files\screen-scraper Enterprise
Edition\procesos\IniciarGIT.bat"</Command>
    </Exec>
    <Exec>
      <Command>shutdown.exe</Command>
      <Arguments>/s /t 10</Arguments>
    </Exec>
  </Actions>
</Task>
```