Analizando a la competencia

Joel Sánchez López

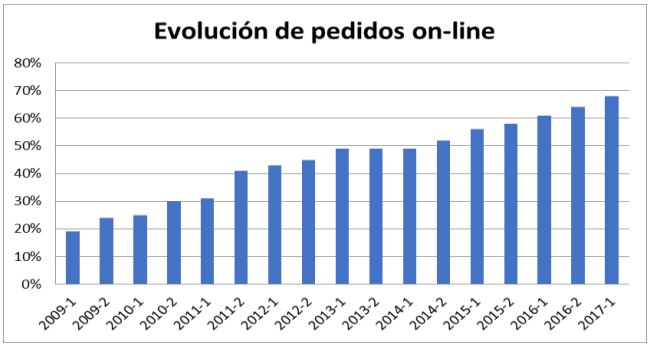
Grado en Ingeniería Informática

Resumen

El proyecto presentado a continuación tiene como objetivo la mejora del plan comercial de la empresa Venca, ayudando así a la toma de decisiones con la finalidad de mejorar la estrategia de mercado y conseguir llegar a los objetivos marcados.
Para ello crearemos un nuevo sistema de análisis de la competencia en el que incluiremos nuevas bases de datos, diversos informes y el programario necesario para rellenarlos.
Las herramientas principales en este proyecto serán Python (utilizando el entorno de desarrollo Spyder de Anaconda), SQL Server Management y Excel (en el cual también se incluye programación en VBA para macros). Más secundarias serán Filezilla y Pentaho, las cuales utilizaremos para hacer llegar los informes de manera automática a cualquier ordenador que los necesite.

1. Introducción a la empresa: Venca

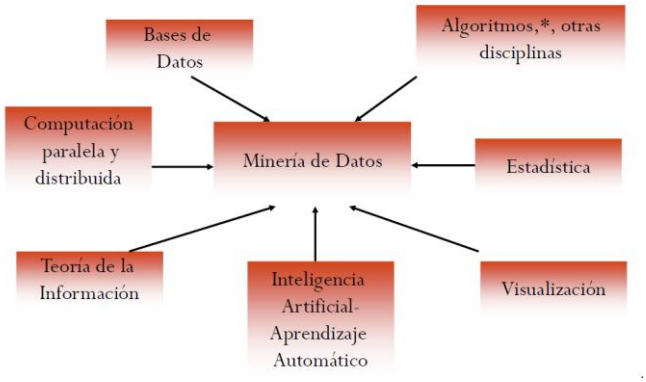
Venca es una empresa multinacional del sector de la moda especializada en la venta a distancia y con una trayectoria de 48 años de experiencia en el sector.
Venca se ha caracterizado por la venta a través del catálogo desde hace más de 30 años. Este tipo de venta a distancia se denomina Off-line, ya que el cliente recibe en su casa un catálogo y para realizar un pedido envía su hoja de petición por correo o fax, o llama por teléfono.
En los últimos seis años esto ha cambiado. Venca, a la vez que toda la sociedad, está cada vez más ligada al mundo on-line. Esto ha hecho que la evolución de la venta a distancia online dentro de Venca haya evolucionado considerablemente. Tanto es así, que actualmente más del 60% de los pedidos generados en Venca son hechos on-line.
En la siguiente gráfica podemos ver la evolución semestral de pedidos on-line desde el año 2009 hasta la actualidad:



A raíz de este cambio dentro de la empresa, Venca necesita una nueva visión y forma de analizar a la competencia, ya que todo está on-line. Una herramienta importante y necesaria es un analizador para poder tratar todos los datos que obtienen a través de diferentes marcas y conseguir una mayor estrategia de mercado para la buena conversión dentro de la empresa.

2. Aspectos teóricos: Minería de datos

En la parte más baja de la pirámide, tendremos quizás miles o millones de datos a los cuales habrá que hacer un procesamiento para extraer solo lo más importante y transformarlos a una estructura que sea comprensible para poder leerlos más tarde. Esto es exactamente para lo que utilizaremos minería de datos.
La Minería de Datos es un proceso que se tiene que centrar en las acciones derivadas del descubrimiento de conocimiento, no en el mecanismo de descubrimiento en sí mismo. La solución es más que un conjunto de técnicas y herramientas.
Este proceso está formado por diferentes ramas que hacen que llegues al resultado esperado:



La finalidad de este proceso es, principalmente, obtener los siguientes casos dentro la empresa:
- Conocimiento de los clientes
- Conocimiento de la competencia
- Detección de segmentos (tanto dentro como fuera)
- Cálculo de perfiles
- Cross-selling
- Detección de buenos clientes
- Mejora de respuesta de mailings
- Campañas de adquisición de clientes.

Para ello, tal y como se menciona antes, se necesitan varios procesos antes de obtener el dato a estudiar.
Como primer paso es necesario un preproceso de datos, ya que los datos reales suelen estar "sucios", es decir:
- Incompletos: se han perdido valores de atributos, atributos de interés o los datos están resumidos.
- Ruido: errores y "outliers" (p.e. precios desorbitados al concluir el crawling).

- Inconsistentes: hay discrepancias en los nombres y/o valores.

Las principales tareas del preprocesamiento son:

- Limpieza de datos: Rellenar los valores nulos, identificar y/o eliminar los outliers, resolver las inconsistencias, tratar los valores con ruido…
- Integración de datos: Combinar datos de diferentes fuentes.
- Transformación de los datos: Normalización y agregación.

3. Tecnologías utilizadas

Para el correcto desarrollo del proyecto son necesarias una serie de herramientas con las cuales se trabaja en la empresa y que, si bien no han sido impuestas, he decidido utilizar por una mera cuestión de comodidad a la hora de utilizar todos las mismas.

- **ANACONDA:**

Con más de 4.5 millones de usuarios, Anaconda es la plataforma de ciencia de datos Python más popular del mundo. Dentro de su inmenso universo, hemos elegido Spyder, como herramienta principal para programar los scripts en Python.

Spyder: (anteriormente Pydee) es un entorno de desarrollo integrado y multiplataforma de código abierto (IDE) para programación científica en el lenguaje Python. Spyder integra NumPy, SciPy, Matplotlib e IPython, así como otro software de código abierto.

- **MICROSOFT SQL SERVER:**

Microsoft SQL Server es un sistema de manejo de bases de datos del modelo relacional desarrollado por la empresa Microsoft.

El lenguaje de desarrollo utilizado (por la línea de comandos o mediante la interfaz gráfica de management studio) es Transact-SQL (TSQL), una implementación del estándar ANSI del lenguaje SQL, utilizado para manipular y recuperar datos (DML), crear tablas y definir relaciones entre ellas (DDL).

- **EXCEL:**

Microsoft Excel es una aplicación de hojas de cálculo que forma parte de la suite de oficina Microsoft Office. Permite realizar tareas contables y financieras gracias a sus funciones, desarrolladas específicamente para ayudar a crear y trabajar con hojas de cálculo.

Esta aplicación nos permitirá dar forma a nuestros datos, organizándolos ya sea en números o texto y en diferentes hojas de cálculo.

También nos permitirá su análisis, gracias a gráficos, tablas dinámicas e infinidad de herramientas que nos ofrece.

- **PENTAHO:**

Pentaho se define a sí mismo como una plataforma de BI "orientada a la solución" y "centrada en procesos" que incluyen todos los principales componentes requeridos para implementar soluciones basadas en procesos y ha sido concebido desde el principio para estar basada en procesos.

- **Spoon / Kettle**: Spoon es la herramienta gráfica que nos permite el diseño de las transformaciones y trabajos del sistema de ETTL's de Pentaho Data Integration (PDI), también conocido como Kettle (acrónimo recursivo: "Kettle Extraction, Transformation, Transportation, and Load Environment).

- **FILEZILLA:**

FileZilla Client es un cliente multiplataforma rápido y fiable de FTP, FTPS y SFTP con muchas funcionalidades útiles y una intuitiva interfaz gráfica de usuario.

4. Implementación del proyecto

Venca, al igual que todas las empresas dedicadas a la venta online, está sufriendo la evolución de los e-commerce, la forma de comprar de sus clientes y la forma de vender de sus competidores. Es por ello por lo que necesita de herramientas para analizar todo este cambio en la medida de lo posible.

Para poder llevar a cabo este proyecto se ha decidido dividirlo en 4 fases claves:

• Programación de los scripts: Aprender Python y conseguir desarrollar programas que automáticamente entren en X páginas web y logren extraer los datos solicitados.

• Incorporación de los datos: Incorporar los datos extraídos en las bases de datos de la empresa.

• Generación de informes: Lograr generar informes entendibles y con los datos solicitados de manera casi automática para el posterior análisis por parte del departamento que los solicite, hacer llegar estos análisis mediante emails o automatizarlos gracias a tareas diarias de Windows, Pentaho y Filezilla.

• Depuración y mantenimiento: Tanto los scripts como las bases de datos y los informes, necesitarán de una supervisión diaria por cualquier cambio o error que pueda suceder en su ejecución o en el entorno web de la competencia.

Programación de los scripts:

Una de las tareas quizás más difíciles, era aprender un nuevo lenguaje de programación no dado en la carrera o visto muy por encima.

A partir de aquí todo fue buscar y buscar información sobre como crawlear webs, ver ejemplos, y ponerse a probar y probar código, hasta dar finalmente con las herramientas y programación correcta.

Para el correcto funcionamiento de los scripts y poder instalar las herramientas que se han utilizado en ellos, bastaría con escribir en el cmd (habiendo previamente instalado Anaconda y por lo tanto Python):

"pip install [selenium/bs4/urllib/pyodbc/time/datetime]".

Todos los scripts constan del mismo formato:

- Importamos librerías.

- Conectamos a la base de datos.
- Definimos variables y extraemos los datos a la base de datos.
- Cerramos conexiones.

La gestión de errores en los scripts es casi imposible, ya que las webs están en constante cambio y ellas mismas tienen links caídos que se pueden rastrear y que dan un error al entrar a ellos. Aun así, se ha hecho una pequeña gestión de ellos, en la cual cada script por si solo genera un informe .txt y cada vez que hay un error escribe el link que ha fallado y el porqué. En caso de detectar un error grave, por ejemplo, en número de artículos rastreados en un día, este archivo nos ayudara a saber que ha pasado.

El proyecto inicial constaba de un total de 7 tiendas (+Venca) a analizar: C&A, H&M, Kiabi, Mango, Shana, Stradivarius y Zara. Al obtener los primeros esbozos y resultados, no se tardó en pedir más tiendas hasta hacer un total de 11; Bershka, Blanco y Lefties.

Todos los scripts se adjuntan en el apartado 9. Anexos de la memoria.

Y muchos más que no se adjuntan por no sobrecargar.

Incorporación de los datos:

Los scripts van a generar una cantidad inmensa de datos cada vez que se ejecuten, para los cuales tenemos que crear una muy buena estructura de base de datos para poder analizarlos y consultarlos de una manera rápida e intuitiva.

Gracias a esto, podemos obtener datos de manera diaria/semanal/mensual, que ayudaran al departamento de compras y a la empresa a enriquecer sus bases de datos y a tomar decisiones basándose en la competencia, ayudándonos a definir mejor nuestras campañas, nuestros precios y a tener una visión general de Venca en comparación al resto.

Al fin y al cabo, Venca, al igual que cualquier otra empresa, intenta llegar a todos sus clientes por Internet intentando darles la mejor oferta posible y el mejor producto posible; un producto que se lleve en este momento y un precio competitivo en el mercado, y qué mejor manera de lograr esto que pudiendo analizar cómo lo hacen grandes y pequeñas empresas con las cuales competimos directamente.

Antes de ponernos con ello, tenemos que hacer un análisis y previsión de los datos que vamos a tener, volumen, información que nos van a aportar y de qué manera los queremos extraer para analizar finalmente, al igual que las veces que rellenaremos las tablas de nuestra futura base de datos en la cual acabaran los datos finalmente tras la ejecución de los scripts.

Para enfrentarnos a este volumen de datos, al principio del proyecto cree una nueva base de datos en Venca para dedicarla únicamente al análisis de la competencia y a los datos que los scripts nos iban a generar, junto con una red de tablas estructuradas dentro de esta base, para ello. Opté por crear una tabla principal donde añadí el objetivo principal del proyecto, que era el análisis de las 11 tiendas pedidas, y puse también una variable país y añadí Francia. Para el resto (niño, niña, bebes, perfumería, tallas grandes, etc.) creé una tabla independiente para cada uno, igualmente con la variable país por si se daba el caso que analizábamos estas ramas también fuera de España, todas ellas, dentro de la misma base de datos previamente creada.

Pros:

- Tenía toda la información principal en una tabla, incluida Francia, lo que me permitía atacar solo una tabla (consultas, procesos, scripts…) y poder hacer análisis de precios entre países directamente desde la misma.
- Tener separado mujer, hombre, niño, niña y bebes y, por lo tanto, poder distribuir el espacio en varias tablas, lo que equivale a consultas más rápidas y precisas.
- Informes y scripts más directos, al atacar cada uno a la tabla que le toca.

Contras:

- En el caso de juntar España y Francia, gran volumen de datos, lo que ralentizaba las consultas.
- Información más dispersa, por lo que había que atacar más de una tabla en el caso de querer datos diferentes.
- Posibilidad de generar datos erróneos si, por ejemplo, no se filtra por país en la tabla que tiene datos de varios de éstos.

Al final, debido al contra del gran volumen de datos, creé una estructura con su tabla y sus procesos para Francia y lo separé de la tabla de España, ya que ralentizaban considerablemente las consultas.

En la decisión de la estructura final también influyó, por no decir que decidió todo, el número de ejecuciones de los scripts, optando finalmente y tras probar la ejecución diaria, por una ejecución semanal. Al ser la que nos daba un volumen de datos más moderado y una velocidad más que aceptable para las consultas.

Análisis que detallo a continuación:

- Diario:
    - Pros:
        - Capacidad de detectar novedades, ofertas, descuentos y promociones diarias de cualquier marca.
        - Capacidad de reacción a las pocas horas en caso de gran cambio en la competencia.
        - Tener una vista general del mercado al día.
    - Contras:
        - Volumen incontrolable de datos.
        - Ralentización en todos los procesos de consulta debido al gran volumen de datos.
        - Hay que disponer de un tiempo diario para generar los reportes y comprobar que todo está correcto.
- Semanal:
    - Pros:
        - Generación de datos moderada y velocidad de las consultas más que aceptable.
        - Visión general del mercado semanal.
        - Capacidad de analizar novedades, ofertas, etc. Semanalmente.
        - Solo necesitas dedicar un día a la semana para la generación de los informes.

- Contras:
  - Pierdes la opción del análisis y respuesta diaria.

- Mensual:
  - Pros:
    - Dispondría de las consultas más rápidas ya que el volumen de datos sería el menor de los 3.
    - En caso de que a la empresa así le interese, disponer de un análisis de mercado mensual.
    - Solo debes preocuparte una vez al mes de la generación de informes.
  - Contras:
    - En el caso del análisis, es el "menos preciso" y el que peor opción a respuesta nos da.
    - Posible pérdida de mucha información durante un mes sobre la competencia.
    - Posible cambio en los códigos fuentes de las webs y, por lo tanto, fallo en los scripts asegurado. (Este estaría siempre, pero a diario lo detectas y lo corriges. Semanalmente también lo detectas y lo corriges de una semana para otra, y das menos margen en ambos casos a las páginas a cambiar su código).

En este caso, el servidor nos viene dado por la empresa, aun así, pudiendo ser un proyecto aplicado a cualquier empresa o particular. En el apartado 6.2 Coste económico de la memoria adjunto también un estudio para la decisión a tomar por tal de elegir un tipo de servidor u otro.

Para crear la base de datos, una vez abierta la interfaz de usuario de SQL y logueados en nuestro servidor, vamos a *Databases* > *New Database…*

Lo único que haremos es ponerle un nombre y darle a "OK". La base de datos irá escalando automáticamente conforme entren los datos, para esto obviamente tenemos que tener conciencia de los datos que vamos a escalar y, asegurar que vamos a tener tanto espacio como poder de procesamiento para ellos. En este caso, lo asegura Venca conjuntamente conmigo, tras ver lo que iban ocupando las ejecuciones.

Tras haber generado la base de datos, nos quedaría generar los usuarios. Podríamos pensar en principio dos; el owner y el de usuario. Uno con acceso a todo y con todo tipo de permisos para poder escribir desde los scripts, leer, crear, borrar, etc. Y otro para el uso simplemente de lectura, ya sea tanto en desde Excel como desde el propio SQL para los usuarios de Venca.

Para crear un usuario nos vamos a *Security* > *Logins* > *New Login…*

Definimos un nombre, una contraseña y una base de datos por defecto en caso de que solo queramos que tenga acceso a una.

Y, por último, en "User mapping", definimos el rol que le queremos dar y a que bases de datos se lo queremos dar.

Ejemplo de creación de la tabla principal:

```
USE [MKT_CRAWLER]
GO

/****** Object:  Table [dbo].[crawlervenca]
******/
SET ANSI_NULLS ON
GO

SET QUOTED_IDENTIFIER ON
GO

CREATE TABLE [dbo].[crawlervenca](
        [Client] [varchar](50) NULL,
        [Category] [varchar](500) NULL,
        [Brand] [varchar](500) NULL,
        [SubBrand] [varchar](500) NULL,
        [URL] [varchar](1000) NULL,
        [Link] [varchar](1000) NULL,
        [NEW] [varchar](1000) NULL,
        [Description] [varchar](1000) NULL,
        [Comment] [varchar](1000) NULL,
        [PriceBeforeDiscount] [float] NULL,
        [ActualPrice] [decimal](16, 2) NULL,
        [Top20] [varchar](500) NULL,
        [Crawl_Day] [datetime] NULL,
        [Crawl_Date] [datetime] NULL,
        [ID_Presentation] [bigint] NULL,
        [#Colors] [int] NULL,
        [CodeProd] [bigint] NULL,
        [MinSize] [nvarchar](1000) NULL,
        [MaxSize] [nvarchar](1000) NULL,
        [Execution_Time] [nvarchar](1000) NULL,
        [linea_2] [varchar](50) NULL,
        [cuartil] [bigint] NULL,
        [fecha] [smalldatetime] NULL,
        [dia] [int] NULL,
        [mes] [int] NULL,
        [año] [int] NULL,
        [sem] [int] NULL,
        [dia_sem] [int] NULL,
        [id_num] [int] IDENTITY(1,1) NOT NULL
) ON [PRIMARY]

GO
```

Después a través de procedures hacemos toda la gestión de los datos:

- procedure_update_linea: Todos los datos extraídos no nos vendrán con unas categorías de producto definidas o al menos, no definidas por nosotros. Para esto estará este procedure. Cogerá los datos de cada día/semana y analizando los nombres de los productos, sus links y las urls de sus escaparates les pondrá un nombre de categoría definido por Venca. Por ejemplo, "Camiseta de tirantes", la podrá en la categoría de "Camisetas", y así con todos los productos que vaya encontrando analizados en la última ejecución.
- procedure_novedades: Como su propio nombre indica, este será el procedure encargado de encontrar las novedades de la última ejecución de los scripts, comparando los productos con todo el historial que hay de ejecuciones y extrayendo solo los nuevos que no ha encontrado previamente.
- resumen_superofertas: Procedure encargado de detectar cualquier producto con descuento y dividirlo en diversas franjas puestas por Venca, siendo >70% la mayor y entre 0,1% y 10% la menor.
  - procedure_resumen_ofertas: Con los datos extraídos previamente, hacemos un pequeño resumen para

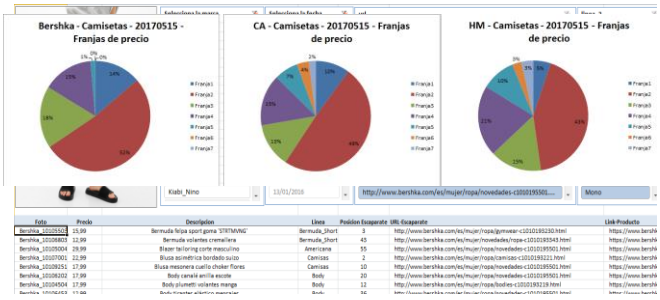
saber en cada categoría cuantos productos tenemos divididos por tipo descuento. Nos da una visión general de la oferta de la competencia, pudiendo llegar a saber el número de Camisetas en oferta y con qué tipo de descuento.
- procedure_resumen_semanal: Este procedure se encargará de hacer el resumen final con el cual elaboraremos los Excels. Se encarga de hacer un resumen para el top 20 de los productos (20 primeras posiciones de los escaparates) y otro para el total de productos. Primero, por categoría de producto, pone los precios a cada franja y finalmente con estas categorías y franjas llama a un subprocedure que se encargará de realizar el resumen final:

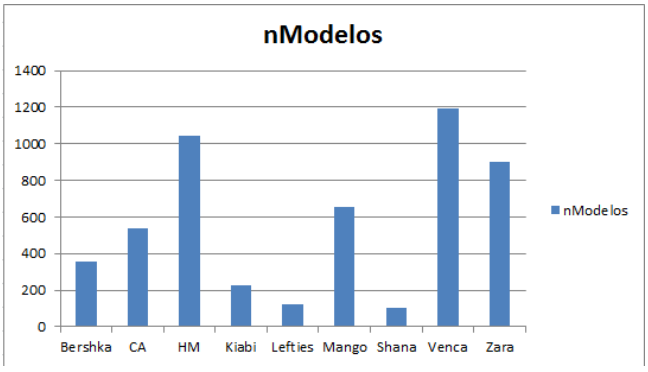
o resumen_diario: Procedure que recibe como parámetros una fecha, una categoría y 7 franjas de precios; con ellas sacara por marca y categoría, el número de modelos por cada categoría, precio medio, precio mínimo, cuartiles, mediana, precio máximo y numero de artículos por franjas de precio.

Informes:

Gracias a estos informes podremos saber cómo evoluciona nuestro plan de campaña online frente al de otras marcas y seguidamente tomar decisiones de como continuar con él.
Mediante VBA automatizaremos la creación de los mismos.

brand	Categoria	Fecha	nModelos	PrecioMedio	PrecioMinimo	Q1	Q2	Mediana	Q3	Q4	PrecioMax
Bershka	Camisetas	20170515	357	10,14	3,99	5,44	8,64	9,99	11,19	15,31	19,99
CA	Camisetas	20170515	538	10,97	3	5,17	7,96	9,9	11,29	19,5	29,9
HM	Camisetas	20170515	1045	13,43	1,99	7,04	10,36	12,99	14,36	21,99	69,99
Kiabi	Camisetas	20170515	223	10,72	2	4,43	8,59	10	11,68	18,2	45
Lefties	Camisetas	20170515	123	8,53	3	5,01	7,36	8,25	9,48	12,26	17
Mango	Camisetas	20170515	653	17,51	4,99	8,84	14,92	15,99	18,88	27,45	69,99
Shana	Camisetas	20170515	106	7,76	2,99	3,62	6,24	7,99	9,33	11,85	15,99
Venca	Camisetas	20170515	1194	10,99	2,99	5,21	7,77	8,99	10,42	20,56	59,99
Zara	Camisetas	20170515	900	13,6	2,95	6,47	11,91	12,95	16,08	19,96	29,95

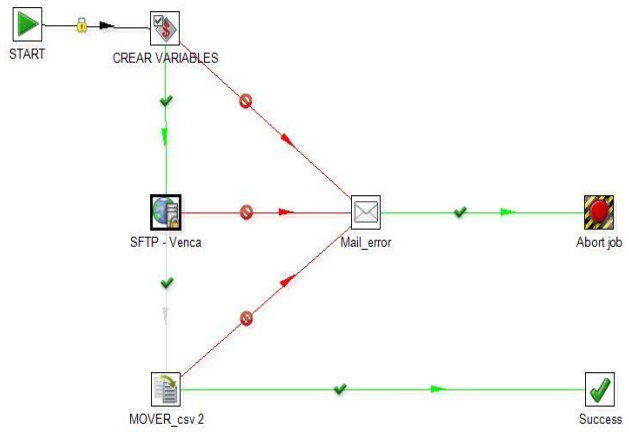




Ejecución y distribución:

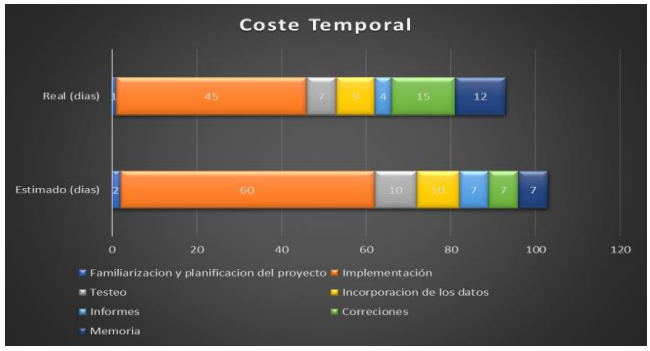
- Ejecución manual: Siempre tenemos la posibilidad de abrir Spyder y ejecutar desde el propio programa cualquier script. La ventaja de esto es que podemos ejecutarlo en cualquier momento que nos lo pidan, e incluso modificar el script para analizar solo un escaparate en concreto o cualquier información que quieran obtener al momento.

- Ejecución automática: Mediante el programador de tareas de Windows. Para ello abrimos “Programador de tareas”, y creamos una tarea básica.
Tras esto, debemos elegir un nombre para la tarea y si queremos una descripción. A continuación, nos pedirá cuando se va a ejecutar esta tarea, si diariamente, semanalmente, mensualmente, una sola vez, etc.
Por último, en acción seleccionamos “Iniciar un programa”, y aquí buscamos el archivo .bat de los scripts, podemos tener todos los programas en un .bat o generar tantos .bat y tareas como scripts queramos ejecutar y añadirlos de uno en uno.
Para la distribución tendremos dos formas:
Una vez cada lunes tenemos los informes hechos, los copiamos auna carpeta compartida por los departamentos que los vayan a utilizar, y finalmente mandamos un e-mail con las direcciones a estas carpetas para que los puedan abrir con tan solo un click.
Y con Pentaho:



1. Crearé las variables con las rutas de salida y destino para los archivos.
2. Conexión al SFTP para dejar los archivos.
3. Mover los ficheros a la carpeta “Tratado”
4. En caso de cualquier fallo en los pasos anteriores, se mandará un email.

5. Planificación temporal y coste económico

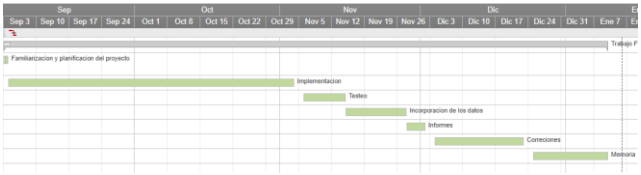


Como podemos ver en la tabla, el proyecto se finalizó antes de lo estimado. Esto es debido mayormente a la dificultad que se le dio a la implementación del proyecto, ya que era programar unos scripts que nadie antes en la empresa se había puesto a programar, y en mi caso, era aprender un lenguaje nuevo de programación para poder realizarlos.

Esta programación se acabó en 1 mes y medio, y se pensaron 2 meses.
Por otro lado, la otra gran diferencia de tiempo viene en las correcciones, ya que se pensó que una vez realizado el proyecto al completo, scripts, test, la incorporación en bases de datos, gestión de las bases con SQL y la generación de informes, corregir posibles fallos iba a ser trivial y no iba a requerir más de una semana. La variación de 8 días viene dada a que conforme pasaban los días y se iban ejecutando los scripts, encontramos que las webs podían cambiar su código fuente, tenía que llenar los scripts de "try catch" para poder dar distintas posibilidades para extraer un dato y prever el cambio de la web, también vimos de la manera que crecían las bases de datos y como ralentizaban las consultas, tuve que dividir en tablas más pequeñas los datos. Y todo esto fue lo que hizo estar más tiempo del previsto corrigiendo errores.
En total el proyecto se realizó en 3,1 meses frente a los 3,4 previstos, que son 744 horas reales contra 824 previstas, un 10% más rápido de lo que se preveía. 10 días antes exactamente.

Diagrama de Gantt:

Nombre de la tarea	Fecha de inicio	Fecha de finaliza…	Duración
Trabajo Final de Grado	**03/09/17**	**09/01/18**	**93d**
Familiarizacion y planificacion del proyecto	03/09/17	03/09/17	1d
Implementacion	04/09/17	03/11/17	45d
Testeo	06/11/17	14/11/17	7d
Incorporacion de los datos	15/11/17	27/11/17	9d
Informes	28/11/17	01/12/17	4d
Correciones	04/12/17	22/12/17	15d
Memoria	25/12/17	09/01/18	12d



En el coste económico vamos a considerar por un lado el coste del personal, en esta casi 1 persona y con lo que sería sueldo de programador, que está actualmente entorno a unos 35 euros, y por otro lado de infraestructura, ya que necesitaremos por un lado un buen ordenador donde programar y gestionar las bases de datos y los informes, y, por otro lado, necesitaremos unos servidores donde almacenar todos los datos. Tenemos un coste estimado de 824 h x 35 €/h = 28 840 € y un coste real de 744 h x 35 €/h = 26 040 €. Aquí podemos ver como finalmente he acabado saliendo más barato de lo estimado, ya que realicé el proyecto en un total de 10 días menos del tiempo estimado por la empresa. Suponiendo este adelanto, un ahorro de 2 800 € y consiguiente ganancia de confianza y demostración a la empresa que ellos están por delante de cualquier bien económico personal.
A la hora del coste de infraestructura para un proyecto de este calibre, tenemos que analizar detenidamente los pros y contras de elegir un servidor u otro, ya que es una inversión muy importante. A continuación, analizamos las 3 opciones disponibles:

- On-premises: Sería la opción en físico, ya sea particular o en la empresa. En el caso de que la empresa ya disponga de un servidor con espacio y poder de procesamiento suficiente para correr el proyecto, no habría problema. En caso de ser particular o no disponer de ello, deberíamos de empezar a analizar que necesitamos y el coste que tendría.
    - o Pros:
        - Precio: Pagas exactamente por lo que quieres y vas a utilizar.
        - Puedes aprovechar infraestructura y caudal.
        - Seguridad personalizada.
    - o Contras:
        - Necesitas una dimensión mínima
          → Tienes que ir redimensionando en caso de que tus bases de datos vayan creciendo más de lo esperado o necesiten de más poder de procesamiento (+RAM, +CPU, +HDD).
        - Tienes que llevar tú el control de back ups y seguridad.
        - Mantenimiento.
    - o Precio: ± 6 000 €.
        - Servidor + Licencias (Windows + SQL)
- Cloud – IaaS: Infraestructura como servicio, es decir, contratamos el servidor entero en Cloud.
    - o Pros:
        - Conectividad.
        - Pagas por uso.
        - Redimensionamiento on demand.
        - Licencias incluidas.
    - o Contras:
        - Precio elevado para un uso 24/7.
        - Back up y mantenimiento se pagan a parte.
    - o Precio: ± 2 400 € anuales.
- Cloud – SaaS: Software como servicio, en esta opción contratamos el servicio de SQL, por un lado, y por otro la máquina virtual encargada de apuntar a este servidor SQL. La mejor opción para SQL es Azure, ya que es la opción oficial de Microsoft, la cual nos garantiza back ups y tener el producto siempre actualizado con sus últimas versiones. Para la opción de máquina virtual, tras analizar el mercado tenemos a DigitalOcean con precios y variantes bastante más asequibles que cualquier otro.
    - o Pros:
        - Ídem IaaS.
        - Mantenimiento, actualizaciones y back ups asegurados.
    - o Contras:
        - Sigue teniendo un precio elevado para un uso 24/7.
    - o Precio: ± 940 € anuales.

En total tendríamos un coste de proyecto de 33 668,58 €*.

**Calculado en base a escoger un servidor por ejemplo físico + ordenador personal. El presupuesto final puede variar según el acuerdo y análisis que se haga con la empresa/cliente tanto para la infraestructura como para el salario.*

Tipo	Coste
Personal **(744 h)** ***(coste x hora = 35€)***	26 040 €
Infraestructura **(ordenador personal + servidor)**	7 628,58 €
Total proyecto	**33 668,58 €**

6. Conclusiones

En primer lugar, me veo obligado a señalar como, a nivel individual, este proyecto me ha apasionado desde el primer momento y ha supuesto no sólo algo que he disfrutado hasta el final, sino también la base de nuevos aprendizajes, y es que durante la realización he aprendido un lenguaje de programación innovador. Durante la carrera opté por una especialización más centrada en bases de datos y SQL, y este proyecto me ha ayudado a profundizar todavía más dentro del mismo ámbito. Con todo, el nivel de SQL adquirido durante el proyecto ha sido realmente notable no solo a nivel de programación y escritura, sino también a nivel de administración de bases de datos gestionando la mía propia dentro de un servidor de Venca, generando y gestionando mis tablas, etc. Asimismo, he aprendido a manejar Excel de una manera que jamás habría imaginado posible; he descubierto trucos, fórmulas, su propio lenguaje de programación VBA y he hecho cosas que hasta el momento desconocía que se podían hacer con dicho programa. Por último, he aprendido una plataforma nueva que desconocía por completo, como es Pentaho, pero que para la empresa es importantísima ya que automatiza la mayoría de sus procesos; la cantidad de cosas que se pueden automatizar y hacer con dicho programa me dejaron totalmente anonadado.

En resumen, a nivel personal he adquirido muchísimo conocimiento de gran valor para mi desarrollo personal y que a su vez me ha servido para acabar de especializarme en aquello que, precisamente, ya empecé en la carrera: el SQL. He descubierto por completo mundos nuevos como son Python, Excel y Pentaho. Me ha hecho marcarme nuevas metas, como podría ser hacer un diseño web y app y, por qué no decirlo, me ha hecho ganarme un puesto en la empresa.

A nivel laboral, el proyecto ha servido para sustituir a un analizador que tenían contratado previamente a nivel externo, pagando X dinero cada mes. No sólo ha supuesto un ahorro a nivel económico, sino que, además, se ha podido disponer de los informes cuando y donde se han requerido, se ha ampliado el campo de análisis añadiendo más marcas de las que tenían contratadas, así como otros países, secciones, etc. Han pasado de analizar 4 marcas españolas en sección mujer, a 11 marcas en el mismo ámbito, marcas francesas, otras de hombre, niño, niña, bebes, marcas de perfumería, tallas grandes, etc.

Mirando al futuro, este proyecto me ha servido para ganarme la confianza de la empresa, entrando a formar parte de ella por tiempo indefinido. El proyecto seguirá creciendo, ya que añadiremos marcas nuevas u otras secciones. Asimismo, buscaré formas de rastrear cada vez más eficientes y, quizás, elabore una pequeña web donde poder mostrar los resultados en vivo y de manera mucho más visual y en la que cualquier persona en la empresa pueda navegar libremente y analizar los datos que ellos quieran en el momento que quieran. Por último, a nivel personal y con tal de evolucionar individualmente, me gustaría desarrollar una app en la cual se pudieran tener las mejores ofertas de las tiendas, llegasen notificaciones de un producto en concreto que se desee, se pudiese seguir su evolución de precios, etc., ya sea tanto a nivel de análisis de empresa como a nivel de uso personal.

7. Agradecimientos

A modo de cierre, no puedo acabar esta memoria sin agradecer a todo mi equipo, Marina Albaladejo (adjunta de Business Intelligence), Jordi Morató (Controller Marketing - BI) y especialmente a Luis Martínez (Responsable de BI), por haber confiado en mí y haber permitido que realice este proyecto y por haberme acogido en su equipo por lo que esperemos que sea mucho tiempo más. De verdad, muchísimas gracias.

Referencias

[1] Beaulieu, A. (2009). *Learning SQL*. 2nd ed. Beijing [etc.]: O'Reilly.

[2] Forta, B. (1999). *Sams teach yourself SQL in 10 minutes*. 4th ed. Indiana: Sams.

[3] Kouzis-Loukas, D. (2016). *Learning Scrapy*. 1st ed. Birmingham: Packt.

[4] Lutz, M. and Ascher, D. (1999). *Learning Python*. 5th ed. USA: O'Reilly & Associates

[5] Matthes, E. (2015). *Python crash course*: *A Hands-On, Project-Based Introduction to Programming.* 1st ed. San Francisco: No Starch Press.

[6] Mitchell, R. (2016). *Web Scraping with Python: Collecting Data from the Modern Web*. 4th ed. Beijin [etc.]: Oreilly & Associates Inc.

[7] Theregister.co.uk. (2018). *DigitalOcean cuts cloud server pricing to stop rivals eating its lunch*. [online] Available at: https://www.theregister.co.uk/2018/01/18/digitalocean_cuts_cloud_server_pricing_to_meet_and_beat_competitive_pressure/.

[8] Walkenbach, J. (2004). *Excel VBA Programming For Dummies*. 4th ed. Indiana: Wiley Publishing.