

TRABAJO DE FIN DE GRADO

Grado en Ingeniería Informática

MEJORA DEL PANEL DE METRICAS DE UNA STARTUP REAL

Julián Tanausú Villar Vázquez



D. **Isabel Sánchez Berreil**, con N.I.F. **42.885.838-S** profesora Contratada Doctora adscrita al Departamento de Ingeniería Informática de la Universidad de La Laguna, como tutora.

C E R T I F I C A (N)

Que la presente memoria titulada: Mejora del panel de métricas de una startup real, ha sido realizada bajo su dirección por D. Julián Tanausú Villar Vázquez, con N.I.F. 78854067Q

Y para que así conste, en cumplimiento de la legislación vigente y a los efectos oportunos firman la presente en

La Laguna a 4 de septiembre de 2017

Licencia

© Esta obra está bajo una licencia de Creative Commons Reconocimiento-CompartirIgual 4.0 Internacional.

RESUMEN

El presente Trabajo de Fin de Grado ha sido propuesto por la empresa Turitop en la oferta del curso 2016/2017 en el Grado en Ingeniería Informática.

TuriTop proporciona todas las herramientas necesarias para que sus clientes puedan vender sus servicios turísticos a través de su página web, además de poder gestionar todas sus reservas. Su solución dispone de una versión beta para la visualización de distintas estadísticas que muestran al cliente la evolución de las ventas de los productos. Sin embargo, al estar en fase preliminar no se ha cuidado su presentación, por lo que no son sencillas de interpretar para los clientes. El objetivo de este trabajo ha sido elaborar métricas y su representación mediante gráficas que faciliten su comprensión en un entorno web para los clientes de la empresa Turitop.

INDICE

1. INTRODUCCIÓN	4
1.1. OBJETIVOS	4
1.2. ALCANCE.....	5
1.3. ANTECEDENTES.....	5
1.4. DESTINATARIOS.....	6
2. ANALÍTICA, GESTIÓN DEL CONOCIMIENTO DE LA EMPRESA . 7	
3. TECNOLOGÍAS USADAS.....	11
3.1. WAMP SERVER	11
3.2. FORMATO JSON.....	11
3.3. API REST.....	13
<i>Características de REST</i>	<i>13</i>
3.4. JAVASCRIPT	15
3.5. AJAX(<i>ASYNCHRONOUS JAVASCRIPT AND XML</i>).....	15
3.6. JQUERY.....	16
3.7. BOOTSTRAP	17
3.8. CHART.JS.....	18
4. IMPLEMENTACIÓN	19
4.1. ARQUITECTURA.....	19
4.2. BASE DE DATOS	20
4.3. MÉTRICAS SELECCIONADAS.....	22
4.4. INTERFAZ	22
4.5. ASPECTOS A DESTACAR	26
5. CONCLUSIONS	28
6. TRABAJO FUTURO.....	29
7. REFERENCIAS.....	30

1. Introducción

TuriTop es una empresa tecnológica dedicada especialmente a ofrecer un sistema de reservas en la nube. Proporciona todas las herramientas necesarias para que los clientes puedan vender todos sus servicios en su página web, en la página web de terceros y en redes sociales. Además se da soporte para la gestión de todas las reservas sin necesidad de hardware, instalación de software o soporte informático.

Turitop dispone de una versión beta de visualización de distintas estadísticas para mostrar al cliente el estado de su empresa y la evolución de las ventas de los productos. Los objetivos específicos de este trabajo se han definido para mejorar la forma en que se le muestran los datos relativos a su empresa a los usuarios.

1.1. Objetivos

Los objetivos propuestos para este Trabajo de Fin de Grado han sido:

- Mejorar el panel de métricas de una startup real, la empresa Turitop.
- La empresa determina como requisito de la aplicación web la transferencia de los datos en formato JSON.
- La implementación del servicio web debe realizarse utilizando php.
- Aprendizaje de tecnologías de desarrollo web.

1.2. Alcance

Se pretende disponer de una aplicación para visualizar el panel de métricas de la empresa Turitop. Para ello se debe consultar datos de la base de datos puesta a nuestra disposición y que da soporte a los servicios que ofertan. En la aplicación se manejan solamente los datos relevantes de cara al cálculo de las métricas, para lo que se almacenan en formato JSON. Se implementará un servicio web de tipo REST que proporcione estos datos. Se implementará una aplicación web para generar las distintas gráficas.

No se creará ninguna tabla adicional a la base de datos. Se debe tener en cuenta, que por motivos de confidencialidad, para la elaboración del TFG sólo se han facilitado ciertas tablas de la base de datos y con campos limitados.

1.3. Antecedentes

Turitop pone a disposición de los clientes una versión beta de un panel de control con métricas globales. Las métricas existentes no son intuitivas para el usuario, cuesta comprender su información, ya que se muestran de forma numérica en tablas. Se puede apreciar en la *Figura 01* cómo se presentan los datos a los usuarios en la versión actual. Se plantea la alternativa de resolver este problema haciendo uso de gráficas que faciliten su comprensión.

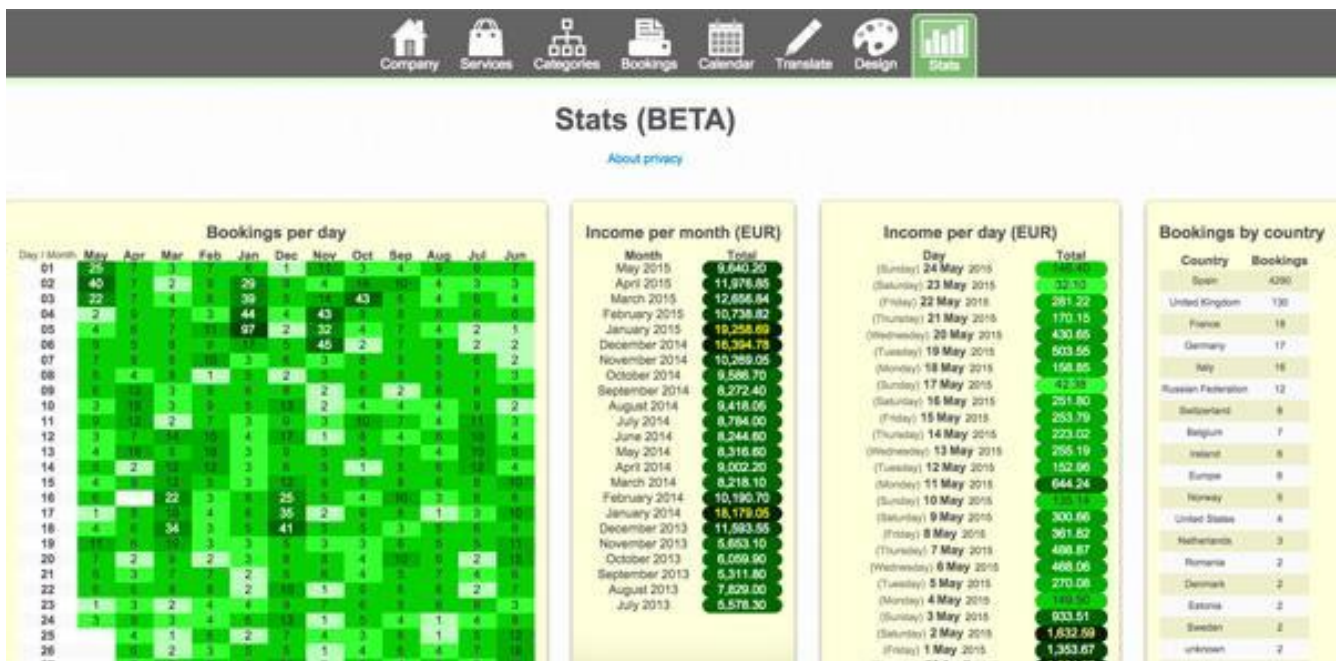


Figura 01: Diseño actual, de las estadísticas mostradas por TURITOP

1.4. Destinatarios

La plataforma está dirigida a toda empresa, por pequeña que sea, cliente de Turitop, para facilitarles la gestión de las reservas y obtener datos relevantes sobre el éxito de sus productos. También a la propia Turitop, ya que está diseñada para facilitar el desarrollo futuro de cuadros de mando en su producto.

2. Analítica, gestión del conocimiento de la empresa

En la actualidad numerosas empresas han tomado conciencia de la importancia de los datos que manejan sobre su negocio. En especial es fundamental identificar las preferencias de los usuarios, conocer qué productos nos están generando clientes, qué es lo más popular, o qué rendimiento se obtiene de algún producto. En este contexto se desarrolla la analítica empresarial, que permite detectar y realizar predicciones sobre el negocio a partir de los datos que se van registrando en la empresa. El analista se encargará de proporcionar datos y resultados de análisis para que los responsables tomen decisiones.

Existen diferentes niveles de análisis que se pueden realizar en la empresa:

- *Analítica Descriptiva*: Se analizan datos históricos para identificar patrones y tendencias. En este nivel se conoce la empresa, los productos, los clientes. Abarca técnicas como la visualización, regresión, etc.
- *Analítica Predictiva*: Se basa en modelos predictivos a partir de los datos que se disponen. Permite anticipar el futuro con una cierta probabilidad de acierto. Corresponde a técnicas más sofisticadas como la minería de datos.
- *Analítica Prescriptiva*: Se utiliza los datos para establecer las estrategias y acciones que permitan mejorar los resultados. Generalmente requieren el uso de algún modelo predictivo previo. Se utilizan técnicas como la optimización y simulación.

Los beneficios que se obtienen al utilizar análisis de este tipo son múltiples, pero dos son los más destacables:

- Genera una visión detallada para el seguimiento del rumbo del negocio y de diferentes variables decisivas en el buen desarrollo de la empresa.
- Permite planificar estrategias a medio y largo plazo, así como generar información necesaria para tomar decisiones rápidas.

En todo proyecto de analítica se deben identificar una serie de objetivos que deben estar alineados con los intereses de la empresa. Por ejemplo: adquisición de clientes, venta de productos, contratación de servicio, etc. En consonancia con los objetivos marcados es preciso definir los indicadores claves de rendimiento (KPI por las siglas en inglés), que permitan monitorizar el desarrollo del proceso y finalmente evaluar los resultados de las distintas acciones.

Las principales características de los KPI son:

- medibles en unidades y por tanto cuantificables,
- específicos: deben referirse a un aspecto concreto
- temporales: se puede establecer la periodicidad de la medición

Por ejemplo:

- Indicadores respecto a ingresos y costes pueden ser: ventas total, ventas por producto, ventas por delegaciones, coste de adquisición del cliente, etc.
- Indicadores respecto a la a la fidelización de los clientes: podemos usar rentabilidad de cliente, tasa de retención de clientes, NPS (medida de la lealtad a una marca), índice de satisfacción del cliente, churn, número de clientes fidelizados, etc

Existen determinados KPIs que son fundamentales según el tipo de

negocio de la empresa (Gemma Muñoz Vera, 2011). por ejemplo:

- *Comercio Electrónico*: Ratio de conversión de compradores nuevos y compradores repetidores, conversiones por campañas de marketing, porcentaje de pedidos respecto a los que iniciaron el proceso de compra, etc.
- *Turismo*: Promedio de visitantes que realizan reservas, Promedio de usuarios que solicitan reserva, porcentaje de ingresos que viene de ofertas.

Otras métricas más complejas y de interés para las empresas pueden ser:

- *Coste de adquisición de cliente (CAC)*: El costo de adquisición de clientes es el precio que se paga por adquirir un nuevo cliente. En su forma más simple, se puede calcular de la siguiente forma:

$$CAC = \text{Costes totales campaña} / \text{nuevos clientes adquiridos en ese periodo concreto.}$$

Este cálculo requiere el registro de datos a lo largo de un periodo de tiempo, pero son muy recomendables, para darse cuenta de qué campañas son las rentables y cuáles no.

- *LifeTime Value (LTV)*: El Lifetime Value es el valor neto de los ingresos que nos genera un cliente durante el tiempo que es cliente de nuestra organización. Se trata de una previsión, ya que no tenemos manera de saber cuánto tiempo permanecerá un cliente, ni su frecuencia de compra, así como el gasto en cada una de las que vaya a hacer. Para realizar el cálculo se evalúa el gasto medio del cliente en sus compras multiplicado por la frecuencia con la que

realiza compras de nuestros productos durante un año y multiplicado por el número de años que es nuestro cliente.

$LTV = \text{Valor medio compra} * \text{Frecuencia Compra} * \text{Duracion}$
Relación

Por lo tanto, un objetivo en la empresa debe ser que el coste de adquisición de un cliente (CAC) sea menor que el lifetime value (LTV), ya que nos estaría costando menos captar un cliente que los beneficios obtenidos por él.

- *Churn rate*: El Churn Rate o Tasa de Cancelación de clientes es una métrica que mide el número de clientes que han dejado de seguir a una compañía en un largo período de tiempo. Con ello obtenemos el porcentaje de abandono de la empresa a partir de las personas que dejan de ser clientes.

3. Tecnologías usadas

En este capítulo se describe el conjunto de tecnologías usadas en el desarrollo del proyecto, hay que destacar que en la mayor parte de los casos su elección ha venido dada por los requisitos marcados por la empresa para una posterior integración del producto en su sistema.

3.1. WAMP Server

WampServer es un entorno de desarrollo web para Windows que permite crear aplicaciones web con Apache 2, PHP y bases de datos MySQL. Además cuenta con la herramienta PhpMyAdmin que facilita la administración de las bases de datos.

3.2. Formato JSON

JSON (JavaScript Object Notation) es un formato para la notación de objetos en JavaScript, cuyo uso en la web para el intercambio de datos se ha extendido por su sencillez frente a XML. Otra de las ventajas de este formato es su facilidad de integración con javascript, si bien es verdad que existen multitud de herramientas para la lectura e interpretación en cualquier lenguaje de programación. Por lo tanto, puede ser usado para el intercambio de información entre distintas tecnologías.

Los elementos de un objeto JSON pueden ser almacenados en dos tipos de estructuras:

- Una colección de pares de nombre/valor. En varios lenguajes esto es conocido como un *objeto*, registro, estructura, diccionario, tabla hash, lista de claves o un array asociativo.
- Una lista ordenada de valores. En la mayoría de los lenguajes, esto se implementa como arrays, vectores, listas o secuencias.

Estas son estructuras universales. En la sintaxis de JSON un objeto se especifica mediante llaves y un array mediante corchetes. En la *Figura 02* se presenta un objeto json, con un único elemento, *marcadores*, que corresponde a un array de objetos.

Ejemplo:

```
{
  "marcadores": [
    {
      "latitude": 40.416875,
      "longitude": -3.703308,
      "city": "Madrid",
      "description": "Puerta del Sol"
    },
    {
      "latitude": 40.417438,
      "longitude": -3.693363,
      "city": "Madrid",
      "description": "Paseo del Prado"
    },
    {
      "latitude": 40.407015,
      "longitude": -3.691163,
      "city": "Madrid",
      "description": "Estación de Atocha"
    }
  ]
}
```

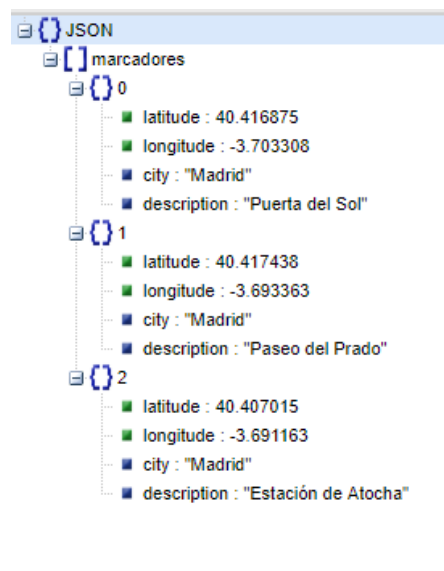


Figura 03: Resultado

Figura 02: Estructura JSON

3.3. API REST

REST es cualquier interfaz entre sistemas que use HTTP para obtener datos o generar operaciones sobre esos datos. Si bien REST surgió como una arquitectura software, en la actualidad también se utiliza el término para referirse a cualquier interfaz que transmite datos específicos de un dominio sobre HTTP sin capas adicionales. Por otro lado, aunque no es un estándar, si está basado en estándares: HTTP, URL, XML, HTML, etc.

En una API REST no se invocan métodos remotos, sino que se accede a recursos a través de sus identificadores, sobre los recursos se pueden llevar a cabo acciones a través de los comandos del protocolo HTTP.

Las ventajas de desarrollar una API REST son la portabilidad a diferentes plataformas, la escalabilidad, visibilidad, y fiabilidad. La principal razón de estas características es la separación entre la interfaz del usuario del servidor y el almacenamiento de datos y la separación entre cliente y servidor. Lo único indispensable es que las respuestas se hagan siempre en el lenguaje de intercambio de información.

Características de REST

Protocolo cliente/servidor sin estado: cada petición HTTP contiene toda la información necesaria para ejecutarla, lo que permite que ni cliente ni servidor necesiten recordar ningún estado previo para satisfacerla.

Las operaciones más importantes relacionadas con los datos en

cualquier sistema REST y la especificación HTTP son cuatro: POST (crear), GET (leer y consultar), PUT (editar) y DELETE (eliminar).

Para la transferencia de datos en un sistema REST, este aplica acciones concretas (POST, GET, PUT y DELETE) sobre los recursos, siempre y cuando estén identificados con una URI.

La URI de un servicio REST, debe ser como la de una página web, intuitiva, de tal forma que el usuario pueda incluso adivinarla. Para lograrlo, se debe diseñar URIS jerárquicas, con una única ruta raíz, abriendo ramas a través de las subrutas. Por ejemplo: Un servicio de hilos de graficas que tiene varios tipos de gráficas.

http://www.TFG_Julian.org/graficas/reservasMes/{mes}

Otras recomendaciones para crear URIS en un servicio REST:

- Ocultar la extensión de archivos en el servidor(.jsp.php.asp), así se puede portar la solución a otra tecnología, sin cambiar las URIS
- Mantener todo en minúsculas
- No dejar espacios en blanco, sustituirlos por guiones
- No se necesita incluir datos para identificar el recurso en la petición, se mapean así todos los datos que la aplicación maneja.

3.4. Javascript

A través de la manipulación del árbol DOM, podremos renderizar nuevo contenido en el navegador como respuesta a las acciones llevadas a cabo por el usuario sobre la página web que actúa como interfaz de usuario. Para ello se usa javascript, lenguaje interpretado que provee los métodos necesarios. El navegador del cliente es el encargado de interpretar las instrucciones javascript, como con cualquier tecnología web es importante ajustarse a los estándares para poder obtener los resultados esperados independientemente del navegador que se utilice. El estándar actual corresponde a ECMAScript 6.

Como lenguaje de programación tiene las siguientes características: interpretado, orientado a objetos, débilmente tipado, dinámico. La sintaxis comparte características de C y Java.

3.5. AJAX(*Asynchronous JavaScript And XML*)

Mediante el uso de AJAX se logra que el intercambio de información con el servidor se produzca de forma asíncrona y envíe sólo la información que es necesario actualizar, no hay que esperar a que el servidor genere toda la página HTML. Los datos adicionales que se solicitan al servidor son cargados en segundo plano, sin variar el comportamiento de la web. Su uso se extiende a múltiples plataformas, sistemas operativos y navegadores. Por lo tanto permite realizar cambios en las páginas sin necesidad de recargarlas, mejorando la velocidad y la usabilidad. En realidad es una combinación de tecnologías:

- JavaScript
- XML, JSON, XLST

- XMLHttpRequest
- DOM
- XMLHttpRequest, CSS

El objeto XMLHttpRequest da acceso al motor de ajax, que es quien se encarga de establecer la comunicación con el servidor en segundo plano, sin necesidad de recargar las páginas (*Figura 04*).

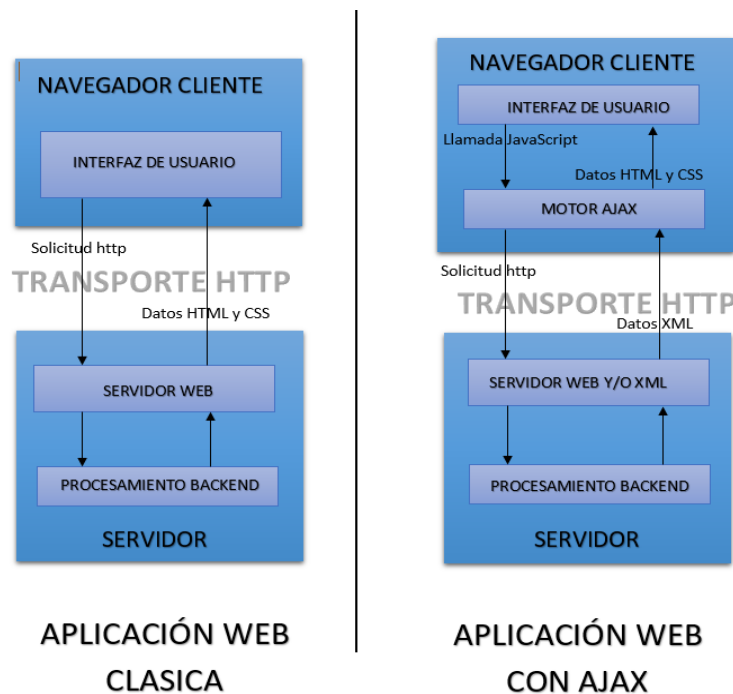


Figura 04: Comparativa

3.6. JQuery

JQuery es una librería de código abierto que facilita la tarea de programar en JavaScript, especialmente la manipulación del árbol DOM, el uso de AJAX y el manejo de eventos. La sintaxis para acceder a los elementos HTML con JQuery se basa en los selectores CSS, lo que también facilita la programación.

La librería es un archivo .js que se puede descargar desde el sitio oficial: <https://jquery.com/> colocar en una carpeta js y luego vincular con una etiqueta <script>, por ejemplo:

```
<script src="js/jquery-3.2.1.min.js"></script>
```

Otra forma de vincular la librería es utilizando el servidor de Google. De esta manera no necesitamos descargarla ni subirla a nuestro servidor.

```
<script  
src="https://ajax.googleapis.com/ajax/libs/jquery/  
3.2.1/jquery.min.js"></script>
```

3.7. Bootstrap

Bootstrap es un framework CSS desarrollado por Twitter que permite dar estilos a una página web mediante librerías CSS. La ventaja de usar este tipo de frameworks radica en la facilidad que aportan a la hora de crear interfaces limpias y que se adapten a cualquier dispositivo, sea cual sea el tamaño de pantalla que se requiera (responsive design).

Con Bootstrap 3 el framework se ha vuelto bastante más compatible con el desarrollo web responsive, entre otras características destacan las siguientes:

- Soporte (casi completo) con HTML5 y CSS3
- Se ha añadido un sistema GRID de 12 columnas donde insertar todo el contenido. Gracias a ello es más fácil la tarea de diseñar una web adaptable e intuitiva.
- Establece Media Queries para 4 tamaños diferentes de dispositivos.
- Nos permite insertar imágenes, que se adaptarán al tamaño de

pantalla.

Es una excelente alternativa para desarrolladores web compatible con la mayoría de navegadores del mercado:

- Google Chrome (en todas las plataformas).
- Safari (tanto en iOS como en Mac).
- Mozilla Firefox (en Mac y en Windows).
- Internet Explorer (en Windows y Windows Phone).
- Opera (en Windows y Mac).

3.8. Chart.js

El desarrollo de las gráficas se realiza en el lado del cliente utilizando javascript, y aprovechando la librería Chart.js

Chart.js es una solución para mostrar gráficos de forma sencilla, similar a la API de Google Charts, pero más sencilla y además es software libre. Una de las principales ventajas de la librería es que su tiempo de carga es bajo. Dispone de 6 tipos de gráficos diferentes, es responsive y basada en HTML5 y CSS3.

4. Implementación

4.1. Arquitectura

El cliente, utilizando AJAX realiza peticiones a la API REST en el lado del servidor y programada en PHP. Esta es la encargada de acceder a la base de datos y almacenar en un fichero JSON los datos relevantes a la petición. A través de la API se transfiere al cliente el fichero JSON generado, y en él se visualizan los datos de forma gráfica, utilizando la librería Chart.js. A partir de ahí, no se realizan más conexiones a la base de datos, todos es leído a través del fichero JSON.

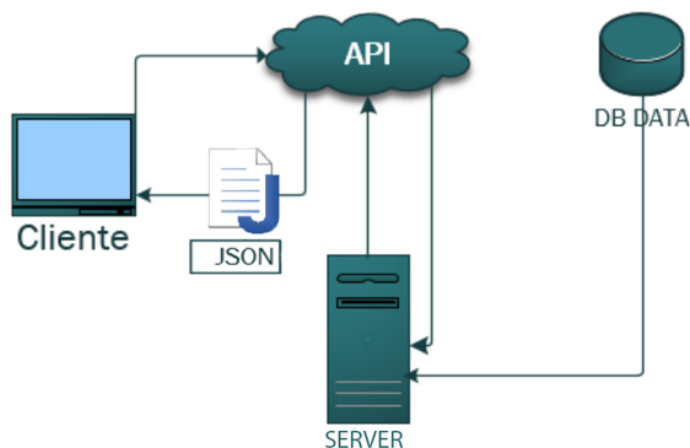


Figura 08

4.2. Base de datos

La base de datos proporcionada por Turitop es solo una pequeña parte de la base de datos original. Por la Ley de protección de datos, solo se dispone de las tablas relevantes para la realización de las gráficas que no violan la privacidad de los usuarios que realizan reservas. También se ha debido respetar restricciones motivadas por los intereses de la empresa. Por estas razones sólo se han incluido los datos de uno de los usuarios cliente de Turitop.

Se dispone de un total de 5 tablas enlazadas entre si: `client_booking`, `company`, `language`, `product`, `product_lang`. Se describe a continuación lo que representa cada una de ellas.

client_booking

Tabla de la cual se ha recopilado la información de cada una de las reservas que esta empresa ha tenido por parte de sus clientes. Lo datos relevantes en esta tabla corresponden a: fecha, dinero facturado y tipo del pago. Aunque la tabla disponga de más datos, estos han sido los escogidos como importantes a la hora de realizar las estadísticas.

Company

Recoge la información de los clientes de Turitop, es decir las empresas que utilizan la plataforma. La información útil de esta tabla ha sido el nombre de la empresa y el identificador, que, como se vio anteriormente, juega el papel de contraseña al loguear.

language

Almacena el idioma de los productos registrados.

product

En product se registran los datos del producto que oferta un determinado cliente de Turitop.

product_lang

Recoge los nombres de los productos que ofertan las empresas usuario del servicio de Turitop.

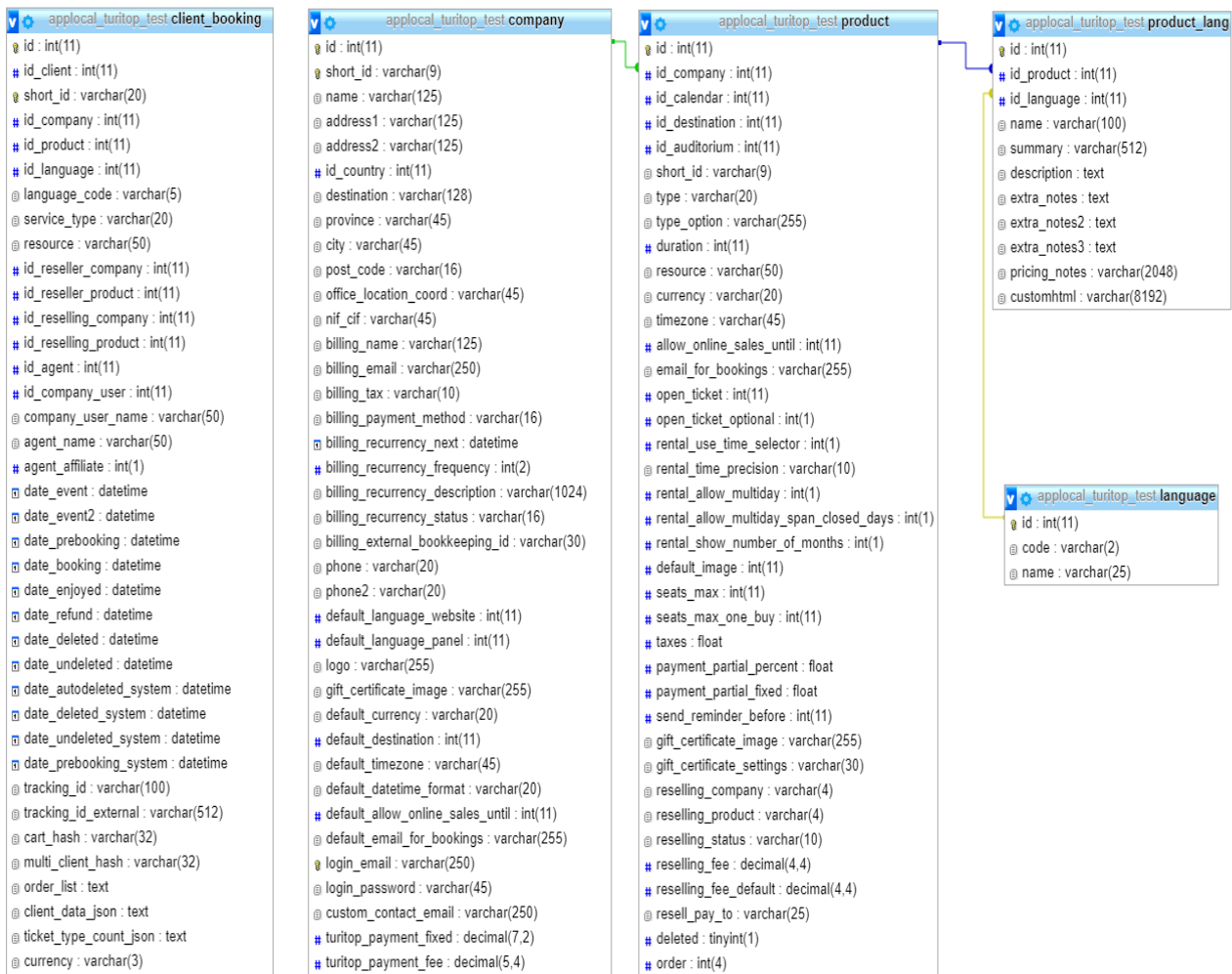


Figura 09: Extracto de la base de datos

4.3. Métricas Seleccionadas

A partir del estudio previo se valoró calcular algunas métricas que pudiesen ser de interés como el Coste de Adquisición del Cliente o el Life Time Value. Sin embargo, para ninguno de estos dos caso se disponía de los datos necesarios, por lo que no se podían calcular. Por tanto, se han considerado exclusivamente las métricas solicitadas por Turitop y que corresponden a:

- Cantidad de reservas en un día, el último mes
- Cantidad de reservas al mes, en el último año.
- Porcentaje ingresos con cada producto

4.4. Interfaz

Se trata de una página web sencilla e intuitiva, donde los clientes de Turitop pueden ver de forma clara la evolución de las métricas de su empresa a través de los gráficos diseñados.

Las métricas calculadas no son de relevancia para la empresa Turitop, están destinadas a quienes contraten sus servicios, facilitando información útil sobre las ofertas que publican en esta plataforma. El acceso se realiza a través del login. El usuario corresponde al nombre de la empresa. Como contraseña se ha considerado el id ya que no se dispone de las contraseñas reales de los diferentes usuarios.

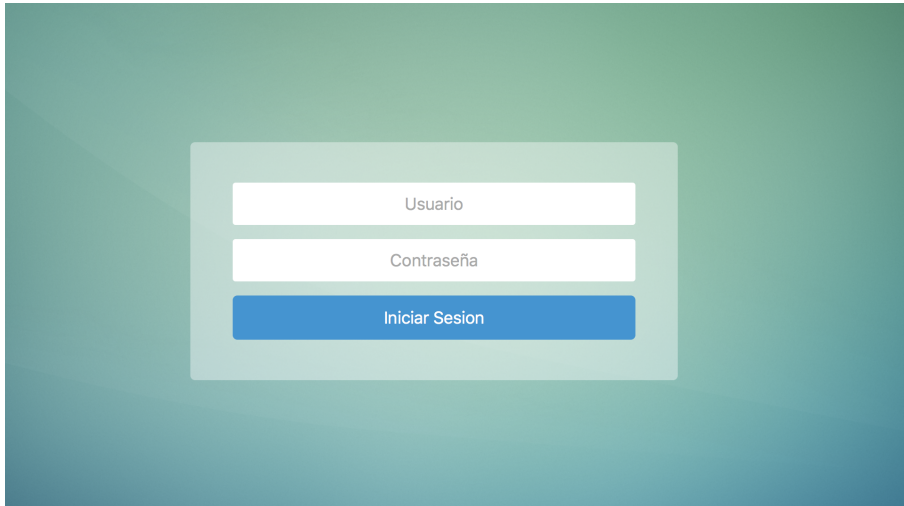


Figura 10: Login

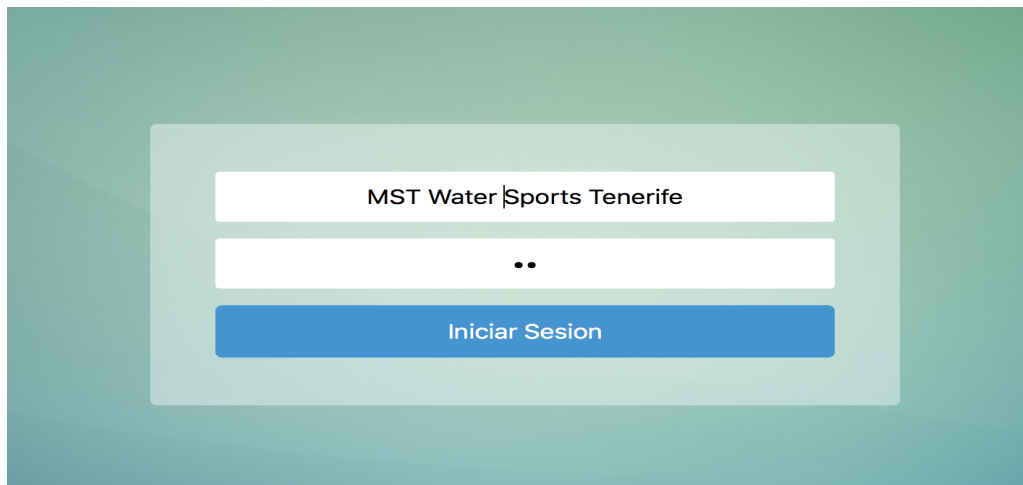


Figura 11: Registro en la aplicación web.

Al hacer login, ya aparece en la esquina derecha superior, el nombre de la empresa logueada, donde se puede cerrar la sesión. Automáticamente se generan las gráficas con los datos de dicha empresa, en la primera observamos la cantidad de reservas realizadas en un día, el último mes. Posicionando el cursor encima

de cada barra, se nos indica la cantidad exacta. Para la siguiente gráfica, el porcentaje de ingresos por producto, debemos de insertar, una fecha inicial, desde la que empezará la búsqueda, hasta una fecha como tope. Una vez establecida la fecha, se nos muestra la gráfica de sectores, cada color un producto distinto. En la que, deslizando el cursor por cada sector, se nos mostrará el porcentaje exacto. Ejemplos de las métricas implementadas se pueden consultar en las Figuras: 12, 13 y 14.



Figura 12: Cantidad de reservas en un día, el último mes



Figura 13: Cantidad de reservas al mes, en el último año.

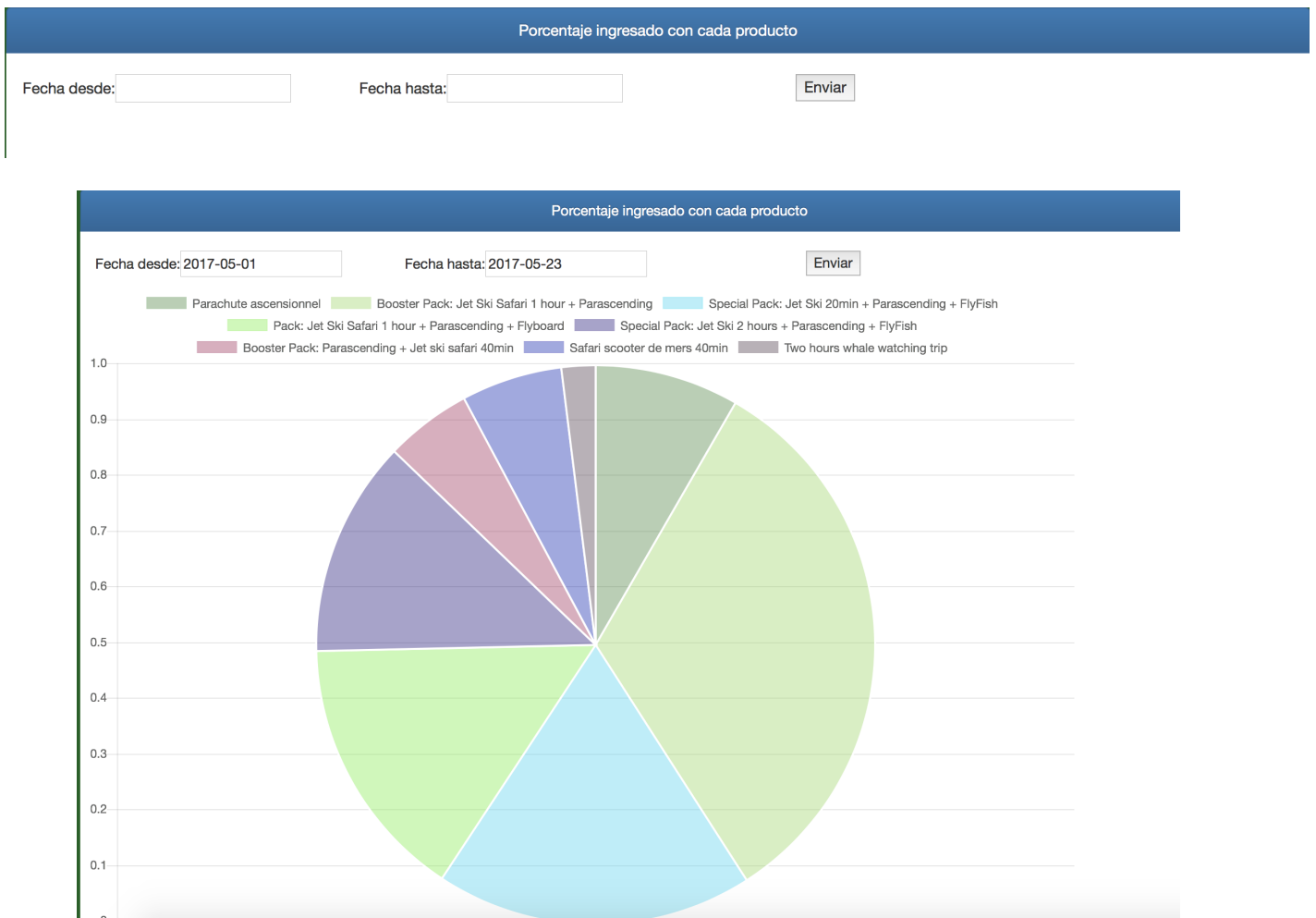


Figura 15: Porcentaje ingresos con cada producto

4.5. Aspectos a destacar

En cuanto al trabajo realizado ha habido varios aspectos que se deben destacar. En primer lugar, para la programación del servicio REST fue necesario un periodo de aprendizaje de esta tecnología. Su uso fue propuesto por la tutora y no tenía experiencia previa en el campo. Por esta razón hubo un tiempo de investigación en el que se realizaron pequeños proyectos de pruebas. Esta fase requirió una parte importante del tiempo dedicado al proyecto.

Otro de los problemas principales en el que se tuvo que invertir una gran cantidad de tiempo, fue el análisis y comprensión de la base de datos facilitada por Turitop. Se requirieron varias reuniones con la empresa. Al comenzar el trabajo se disponía de la versión completa de esta, después de un tiempo de trabajo y una vez concretadas las métricas que se decidió programar, se simplifica esta, conteniendo exclusivamente las tablas más relevantes para diseñar las gráficas.

Finalmente, en la fase inicial del trabajo los estilos fueron programados en CSS sin utilizar ningún framework de diseño por desconocimiento de estas tecnologías. En la revisión de esta parte del trabajo la tutora marca como requisito el uso de bootstrap, para obtener un diseño moderno, adaptable a cualquier tamaño de pantalla y de una forma eficiente. El tiempo empleado en el aprendizaje de esta tecnología fue relativamente corto, a lo que ayudó la excelente documentación de su página web.

Para finalizar, se ha implementado tanto seguridad en el cliente, como en el servidor, con el objetivo de evitar inyección SQL.

Sobre el cliente:

- No se admiten campos vacíos.
- Uso de expresiones regulares, para no aceptar caracteres especiales.

sobre el servidor:

- Utilización de las declaraciones preparadas que nos proporciona la API de MYSQL para PHP
- Aseguramos que las peticiones que llega al login, es una petición AJAX.
- Se escapa cualquier carácter especial, que pueda tener el login.

5. Conclusions

In conclusion, this work has not been as complicated as I expected, the great advantage is that I already knew of php, html, css, javascript, which made the task much easier.

The most laborious has been to understand the Turitop database, in order to know which tables you would have to use and make the appropriate queries.

On the other hand, it also took me a while to understand the REST API, I had to practice doing a little project about it.

All the goals set by the company have been achieved, to achieve them have been raised the use of modern technologies for the programming of web applications. All this has also been achieved to meet the goal of learning some unknown web technologies before starting work such as AJAX, CSS frameworks, REST services, etc.

6. Trabajo futuro

Respecto a la herramienta que se ha programado la principal tarea que se debería acometer como trabajo futuro es su integración en el sistema de la empresa. Sin embargo, de la forma en que se ha programado no debería encontrarse ninguna dificultad importante.

Por otra parte, se recomienda hacer pruebas con la totalidad de la base de datos, a la cual nos ha sido imposible acceder.

Del mismo modo será fácil de implementar nuevas métricas, que puedan resultar de interés. También sería de utilidad configurar cuadros de mando que integren varios gráficos y que la aplicación disponga de una versión móvil, con la que los clientes puedan acceder en cualquier momento y solo dándose de alta la primera vez que se arranca la aplicación, a todos los datos y estadísticas de su empresa.

7. Referencias

- JC Mouse. (FEBRERO 18, 2016). Crea un servicio web REST con PHP y MYSQL. FEBRERO 18, 2016, de JC-MOUSE Sitio web: <http://www.jc-mouse.net/proyectos/crea-un-servicio-web-rest-con-php-y-mysql>
- Mark Otto. Bootstrap, de Bootstrap Sitio web: <http://getbootstrap.com/>
- Chart.js. Mar 17, 2013, de GitHub Sitio web: <http://www.chartjs.org/>
- Geeky Theory. JSON, de Github Sitio web: <https://geekytheory.com/json-iii-gestionar-json-en-php>
- Gemma Muñoz Vera. (2011). El arte de medir: Manual de analítica Web.: Profit Editorial.
- Dave Methvin. (August 22nd, 2005). JQUERY. August 22nd, 2005 , de JQUERY Sitio web: <https://jquery.com/>