



UNIVERSIDAD DE BURGOS
ESCUELA POLITÉCNICA SUPERIOR
Grado en Ingeniería en Informática



TFG del Grado en Ingeniería Informática
Predicción de apuestas deportivas



Presentado por Nuño Basurto Hornillos
en Universidad de Burgos — 23/01/2017
Tutor: Alvar Arnaiz González
Cristobal José Carmona del Jesús

Índice general

Índice general	I
Índice de figuras	III
Apéndice A Plan de Proyecto Software	1
A.1. Introducción	1
A.2. Planificación temporal	1
A.3. Estudio de viabilidad	4
Apéndice B Especificación de Requisitos	8
B.1. Introducción	8
B.2. Objetivos generales	8
B.3. Catalogo de requisitos	9
B.4. Especificación de requisitos	9
Apéndice C Especificación de diseño	17
C.1. Introducción	17
C.2. Diseño de datos	17
C.3. Diseño procedimental	20
C.4. Diseño arquitectónico	20
Apéndice D Documentación técnica de programación	23
D.1. Introducción	23
D.2. Estructura de directorios	23
D.3. Manual del programador	24
D.4. Compilación, instalación y ejecución del proyecto	26
D.5. Pruebas del sistema	30
Apéndice E Documentación de usuario	32
E.1. Introducción	32

<i>ÍNDICE GENERAL</i>	II
-----------------------	----

E.2. Requisitos de usuarios	32
E.3. Instalación	33
E.4. Manual del usuario	34

Bibliografía	38
---------------------	-----------

Índice de figuras

A.1. Distribución de los commits a lo largo del tiempo.	4
A.2. Distribución de los commits en días y horas.	4
B.1. Diagrama completo de los casos de uso.	10
C.1. Modelo relacional	18
C.2. Tabla de partidos	19
C.3. Diagrama de flujo	21
C.4. Arquitectura de tres niveles Cliente servidor	22
D.1. Loguearse en Drupal	24
D.2. Administración de apuestas	25
D.3. Atajos en Drupal	25
D.4. Editar contenidos	25
D.5. Apache en funcionamiento	27
D.6. Instalación PHP	29
E.1. Importación de servicio virtual.	33
E.2. Preferencias del servicio virtual.	34
E.3. Menú con el que puede interactuar el usuario.	35
E.4. Vista del usuario de Inicio.	35
E.5. Vista del usuario del Informe de la Jornada.	36
E.6. Vista del usuario del Informe Jornada.	36
E.7. Vista del usuario del Informe de la Jornada con la URL diferente.	37
E.8. Vista del usuario del Informe del pronóstico.	37

Apéndice A

Plan de Proyecto Software

A.1. Introducción

Vamos a ver cómo ha ido evolucionando el proyecto durante su realización, así como un estudio de viabilidad donde podemos ver si podemos llevarlo a cabo en el futuro.^[4]

A.2. Planificación temporal

El proyecto se ha llevado a cabo utilizando metodologías ágiles, más concretamente SCRUM, realizando un numero de reuniones donde concretábamos los objetivos que íbamos a llevar acabo en cada sprint, no solo nos comunicábamos en estas reuniones, también utilizábamos el gestor de proyectos, Trello para resolver aquellas dudas que iban surgiendo durante la realización del sprint, una vez se acababa en el sprint en la reunión observábamos el trabajo realizado. Tanto en Trello como en GitHub podemos ver como se han ido desarrollando cada uno de los objetivos de los sprints.

Sprint 1 (20/09/2016 – 04/10/2016)

Es una toma de contacto con la gran cantidad de herramientas con las que vamos a trabajar. Lo primero y fundamental es la creación de una máquina virtual Ubuntu 14.04, sobre ella se va a instalar un servidor Apache, se va a seguir un tutorial que ha sido cedido por el tutor. Tras analizar algunas herramientas de gestión de proyectos como Git, Bitbucket o GitHub, se decide trabajar con GitHub como gestor de versiones y con Trello como gestor de proyectos. Tras ello vamos a montar Drupal sobre el servidor Apache, durante la instalación de Drupal se han ido encontrado algunos problemas para enlazarlo correctamente con la base de datos que se ha creado durante la instalación.

de Apache. Por último se empieza a investigar sobre el scraping y se crea un sencillo script con el que extraer algún dato.

Sprint 2 (04/10/2016 – 21/10/2016)

Lo primero es crear una estructura en la base de datos en la cual poder almacenar la información, que vamos a ir recopilando mediante los algoritmos de web scraping. El primer algoritmo de scraping que vamos a crear es el que extraiga los resultados de los partidos y con ello todas las estadísticas. Es importante la selección de la página web sobre la que realizar el scraping ya que se necesitan una uniformidad en la URL. La primera página web que elegimos es la de Marca.com ya que la URL de los partidos es válida, el problema surge al extraer varios partidos ya que en algunos partidos directamente no hay estadísticas. Tras una larga búsqueda se encuentra otra página web, resultados-futbol en ella no nos encontramos con los problemas de Marca y la URL es válida. Todos los datos que extraemos mediante el algoritmo de scraping hay que cargarlos en la base de datos, esto lo vamos a hacer utilizando una sintaxis de PHP especial para el acceso a MySQL a través de PHP. Inicialmente está habiendo muchos problemas para la utilización de algunas funciones de carga de datos. Finalmente, se intentan añadir algunos temas a Drupal como por ejemplo el de Trello para poder trabajar directamente sobre Drupal. Pero nos encontramos con un error de autenticación, que indagando por foros vemos que no tiene solución por lo cual descartamos estos y trabajamos con Trello independientemente desde el navegador.

Sprint 3 (21/10/2016 – 08/11/2016)

En este sprint tenemos que implementar el algoritmo de backpropagation, esto en un principio es un problema ya que mi conocimiento de PHP no es demasiado grande y el tiempo que se va a tardar en llevarlo a cabo puede extenderse. Valoramos diferentes algoritmos que vamos encontrando por internet, para tomarlos como base para la creación de nuestro algoritmo, finalmente nos quedamos con uno en Python y procedemos a su traducción a PHP, resulta costosa dado que este algoritmo en Python tiene clases y es así como lo vamos implementando en PHP. Creamos un algoritmo de scraping que extrae, de la misma página utilizada anteriormente para la extracción de resultados, las cuotas de las casas de apuestas, lo óptimo es la ejecución de este script con la menor anterioridad a la jornada, ya que no todas las cuotas se encuentran en cualquier momento de la semana y estas van variando.

Sprint 4 (08/11/2016 – 15/11/2016)

Dado que el algoritmo no ha sido terminado correctamente decidimos prolongar su implementación a este sprint y tratamos de ver su respuesta ante una base de datos sencilla como Iris y más adelante probarle con otras bases

de datos como balance, pima y wine. Tras varios ajustes en el código conseguimos una buena ejecución con Iris. Para ir preparando la entrada de datos a la red neuronal del algoritmo de backpropagation, se lleva a cabo un script que calcule las rachas de los equipos, además se realiza un tercer algoritmo de scraping para la recolección de los datos de cada equipo en la clasificación, este cambio conlleva un cambio en la base de datos. Creamos las instancias para darlas de entrada en el algoritmo de backpropagation pero aún hay que hacer algunos ajustes ya que los datos no son del todo correctos.

Sprint 5 (15/11/2016 – 23/11/2016)

Lo primero en este Sprint es terminar las instancias que vamos a insertar en el algoritmo de backpropagation, en el anterior sprint se cargó la clasificación del equipo y la idea en un primer momento era hacer un update por cada jornada, en vez de esto vamos a cargar las clasificación de cada equipo en cada jornada, las rachas y la clasificación se unirán en la base de datos, teniendo la información de cada equipo en cada una de las jornadas. Tras varias semanas copiando y pegando el código de Sublime Text en el nodo de Drupal se encuentra una manera de vincularlo y así ahorrar bastante tiempo.

Sprint 6 (23/11/2016 – 06/12/2016)

Con las instancias ya definidas vamos a intentar que la ejecución nos de valores con los que poder empezar a trabajar, estos valores cuestan obtenerlos ya que hay que hacer varios ajustes en la red neuronal, como el control del número de iteraciones o la cantidad de neuronas que deseamos utilizar, las pruebas las vamos almacenando poco a poco para luego compararlas tranquilamente.

Sprint 7 (06/12/2016 - 22/12/2016)

En este sprint vamos a dejar terminada la interfaz, los informes que muestran resultados van a quedar acabados de cara al usuario, distinguiendo tres informes, el de toda la temporada, el que muestre lo obtenido en pasadas jornadas y el de la próxima jornada. También vamos a normalizar el nombre de los nodos de Drupal y de los scripts, el objetivo es una mayor homogeneidad. Las direcciones URL también van a ser normalizadas y se va a poner una pantalla de Inicio para el usuario.

Sprint 8 (22/12/2016 - 09/01/2017)

En este último sprint vamos a dejar toda la documentación cerrada y algún pequeño retoque en la web. Los scripts van a ser reforzados con control de errores fuera de ejecución para cerrar completamente el proyecto.

Vamos a observar dos gráficos extraídos de GitHub donde se ve el trabajo realizado, y la distribución del mismo. En la primera figura [A.1](#), observamos

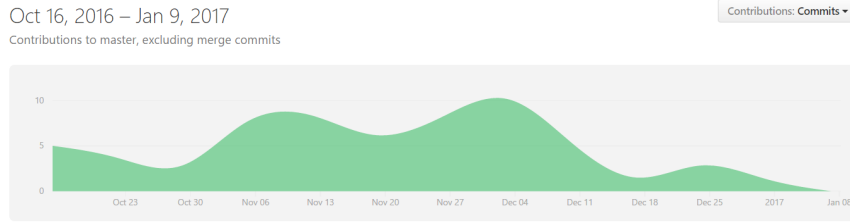


Figura A.1: Distribución de los commits a lo largo del tiempo.

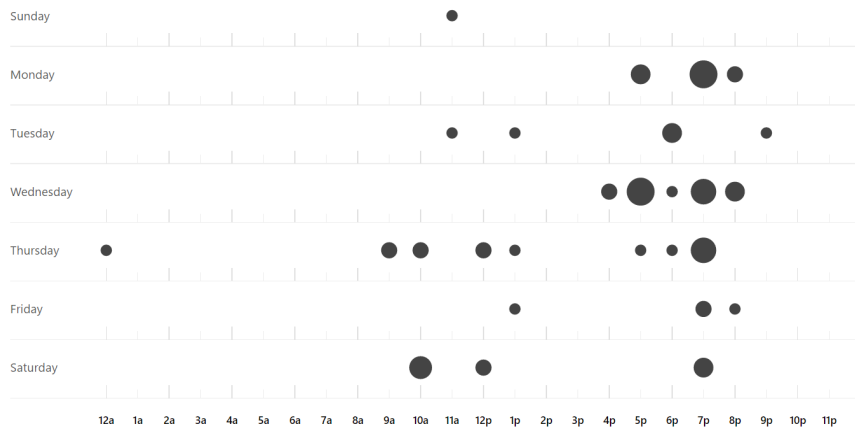


Figura A.2: Distribución de los commits en días y horas.

como un incremento de commits durante la realización del cuarto Sprint, ya que en este se terminó de implementar el algoritmo de backpropagation. También vemos un aumento de commits durante el sexto sprint, donde se analizó la configuración de la red neuronal y el correcto preprocesamiento de los datos en entrada a la red neuronal.

En este segundo gráfico [A.2](#) podemos observar las horas más fructíferas de trabajo. Dado que por las mañana trabajaba el número de commits es significativamente más bajo, en cambio por las tardes en torno las 19:00 es donde vemos un mayor número.

A.3. Estudio de viabilidad

En esta sección vamos a analizar si se puede apostar o no por este proyecto en un futuro, vamos a tener en cuenta diferentes puntos de vista, como la viabilidad económica y la legal. Sin duda ambos factores son determinantes para seguir con el proyecto o al menos para que podamos lucrarnos de ello.

Viabilidad económica

Vamos a ver una visión económica del coste que supondría realizar este proyecto, para un único trabajador durante 4 meses, trabajando 2 horas diarias.

Es necesario contratar a un trabajador con experiencia en Drupal y PHP, que tenga cierta soltura para trabajar con bases de datos, estimamos que el sueldo del trabajador será de 20€ /hora netos, por lo que el 32,1% de cotización a la Seguridad Social [3] le corresponde a la empresa. Teniendo en cuenta 22 días laborables al mes:

$22 \text{ días/mes} * 2 \text{ horas/día} * 20\text{€/hora} = 880\text{€/mes}$ a pagar al trabajador. Teniendo en cuenta el pago de la SS: $880 * 1.321 = 1168.42\text{€}$ al mes hay que invertir en el trabajador.

Esto supone que los 4 meses de duración del proyecto suponen 4649.92€.

Ahora hay que analizar el coste que supone el hardware y el software, como todas las herramientas utilizadas o tenían licencia GNU o una version gratuita suficiente, no ha sido necesario ningún coste en lo referente a software. En cuanto al hardware, el equipo tiene como especificaciones: Intel Core i7, 8GB de RAM y un disco duro de 1TB, el precio aproximado del equipo es de 850€. El plazo de amortización es de 4 años.

- Gastos de amortización anuales: 212.5 €
- Coste mensual: $212.5 / 12 = 17.71 \text{ €}$
- Como el proyecto es de 4 meses el gasto de amortización es: $17.71\text{€} * 4 = 70.83\text{€}$

Los gastos mensuales de Internet son de 50€ mensuales, suponiendo 200€ al final del proyecto.

Los costes totales ascienden a: $4649.92 + 70.83 + 200 = 4920.75 \text{ €}$

La viabilidad económica del proyecto una vez realizado, en un futuro podemos enfocarla de dos formas, ya que depende del uso del programa o de la venta a una empresa del mismo.

- Uso personal: Los beneficios y pérdidas que se han ido logrando, podemos observarlos en el balance general, lo normal es que cuanto más información posea la red neuronal, mejores resultados nos de.Imagen
- Venta a empresas: Sin duda esta no es la finalidad del proyecto, es difícil que una casa de apuestas opte por un proyecto como este dado que hay servicios de multinacionales que nos ofrecen una mayor capacidad de

computación que el nuestro. Este proyecto lo veo algo más destinado al pequeño consumidor que quiera consultar resultados y opciones de apuestas en un momento dado.

No obstante dado que la licencia utilizada en GNU/GPL no se han realizado cálculos de licencias u otras formas de generar dinero.

Viabilidad legal

En el aspecto legal la única técnica cuestionable es el web scraping. En cuanto a su uso, se considera ilegal si puede generar un riesgo de asociación o comporte un aprovechamiento indebido de la reputación o el esfuerzo ajeno, es decir, es desleal sin supone un obstáculo a la afirmación de esa empresa en el mercado. El Tribunal Supremo dicta que las técnicas de scraping constituyen una técnica legal si se cumplen determinadas condiciones y supuestos.[referencia]

En resumidas cuentas, no depende del scraping en sí mismo, sino del uso que se le dan a los datos extraídos con este mismo. Si el uso de estos datos resultan competencias desleal nos encontraremos ante un acto ilegal.

Las licencias de los programas utilizados tienen las siguientes licencias:

- **Drupal:** GPL v2 y GPL v3.
- **VirtualBox:** Privativa y GPL.
- **Ubuntu:** GPL y otras licencias libres.
- **Apache:** Licencia Apache 2.0.
- **MySQL:** GPL o uso comercial.
- **PHP:** Licencia PHP.

Como vamos a nombrar copyleft, es necesario explicar brevemente de lo que trata. Consiste en permitir una libre distribución de copias y versiones modificadas, donde los derechos de autor persisten en las versiones modificadas[6].

La licencia GNU GPL es la licencia más utilizada en cuanto a software libre. Tiene como propósito decir que el software es libre, pero se encuentra protegido mediante *copyleft*, de apropiación externa[7].

La licencia de PHP es una licencia de software libre pero que no permite copyleft a diferencia de la GNU GPL. La licencia PHP no es compatible con la licencia GPL[8].

En cuanto a la licencia Apache, se trata también de una licencia de software libre permisiva, requiere la conservación de los derechos de autor pero no es copyleft [5].

Una vez realizado este análisis de las licencias utilizadas, hemos decidido poner una licencia GNU GPL al proyecto dado que queremos mantener los derechos de autor, además gracias al *copyleft* mantenemos los derechos si se llevan a cabo nuevas versiones.

Apéndice B

Especificación de Requisitos

B.1. Introducción

Vamos a analizar profundamente todo lo referente a los objetivos del proyecto, si las metas, marcadas en un principio han sido alcanzadas y los requisitos que hemos establecido para guiarnos a lo largo del proyecto.

B.2. Objetivos generales

Desde el comienzo del proyecto se propusieron unos objetivos sobre los cuales construir el proyecto.

- Como bien vemos en el nombre del proyecto, predicción de apuestas deportivas, este es sin duda la piedra angular del proyecto sobre la que elaborar el resto. Hay que obtener el resultado final de partidos de fútbol, esto lo lograremos utilizando redes neuronales.
- Una vez tenemos los resultados de los partidos, estos hay que mostrarlos a los usuarios, lo realizaremos utilizando una interfaz sencilla y atractiva, la cual pueda utilizar el usuario, dado que estamos tratando con las apuestas deportivas se van a mostrar algunas cuotas de las casas de apuestas.
- Dado que no se puede estar a diario pesando que algoritmos de scraping ejecutar o que datos hay que mostrar es necesario automatizar todo, de esta manera el usuario tendrá toda la información actualizada y no esperar la intervención humana.

B.3. Catalogo de requisitos

En este apartado vamos a ver todas las características que posee nuestro proyecto y que se han implementado. Podremos ver todos los requisitos funcionales.

- **RF-1:** Se deben obtener todos los datos mediante técnicas de web scraping y almacenarlos en la base de datos.
 - **RF-1.1:** Se tienen que obtener los resultados de los partidos disputados.
 - **RF-1.2:** Se debe obtener cada jornada los datos de cada equipo en la clasificación.
 - **RF-1.3:** Se deben conseguir las cuotas de diferentes casas de apuestas en los partidos de cada jornada.
 - **RF-1.4:** los datos del web scraping se deben almacenar en la base de datos.
- **RF-2:** Disponer de una red neuronal capaz de predecir resultados de futuras jornadas.
 - **RF-2.1:** Hacer un cálculo de las rachas para las instancias de entrada en la red neuronal.
 - **RF-2.2:** Preprocesar los datos de entrada para acomodarlos a lo que espera la red neuronal.
 - **RF-2.3:** Realizar el reentrenamiento de la red con los datos de la BBDD.
 - **RF-2.4:** Realizar la predicción de los partidos de las siguientes jornadas y almacenarla en la base de datos.
- **RF-3:** Diseño de la página web intuitiva permitiendo una buena interacción con el usuario.
 - **RF-3.1:** Realizar la administración de Bootstrap.
 - **RF-3.2:** Se deberá poder observar lo logrado en el global de jornadas previas.
 - **RF-3.3:** Se deberá poder observar detalladamente cada jornada.
 - **RF-3.4:** Se podrá observar el pronóstico realizada para la próxima jornada.

B.4. Especificación de requisitos

En esta sección vamos a apreciar más detalladamente lo referente a los casos de uso.

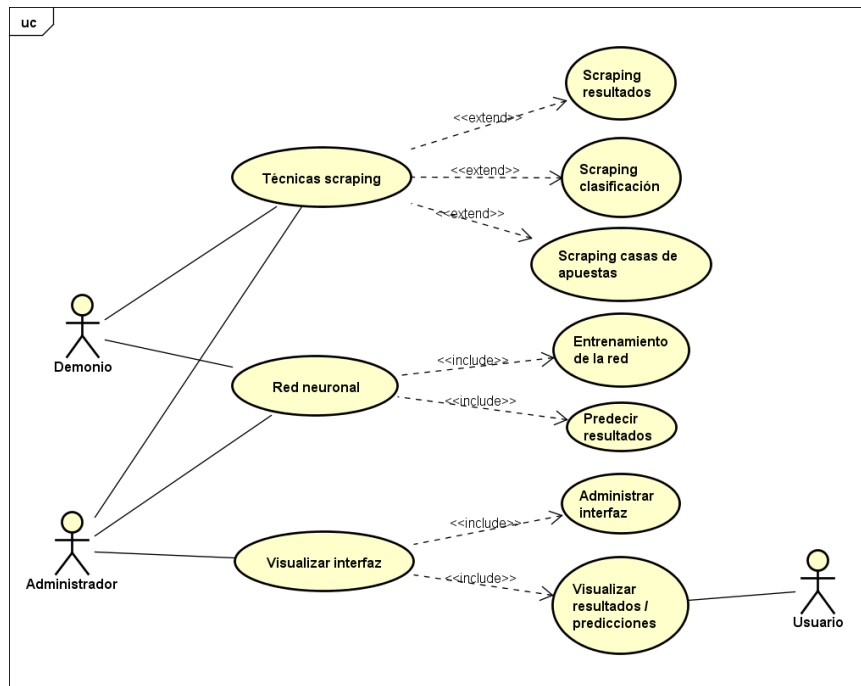


Figura B.1: Diagrama completo de los casos de uso.

Casos de uso

En la figura B.1, podemos observar el diagrama de casos de uso del programa, a continuación vamos a ver más detalladamente cada caso de uso. Son tres los actores que podemos encontrar en el diagrama de casos de uso y es que el demonio se encarga de la ejecución automática, pero el administrador si lo desea también puede realizar las diferentes ejecuciones. Por último el usuario únicamente puede visualizar la interfaz.

Caso de uso	Técnicas scraping
Requisitos	RF-1 RF-1.1 RF-1.2 RF-1.3 RF-1.4
Descripción	El sistema obtiene todos los datos necesarios para poder utilizar la red neuronal y para poner en la interfaz.
Precondiciones	Los datos no han sido recopilados previamente.
Acciones	1. El administrador ejecuta los algoritmos de scraping desde Drupal, o bien espera la ejecución automática del demonio. 2. La aplicación realiza la ejecución. 2.1 Se recopilan los datos en la base de datos. 2.2 Se actualiza la interfaz web.
Postcondiciones	Ninguna
Excepciones	Si se intentan introducir los datos de nuevo se produce un error en el sistema.
Importancia	Alta

Cuadro B.1: Caso de uso “Técnicas scraping”.

Caso de uso	Red neuronal
Requisitos	RF-2 RF-2.1 RF-2.2 RF-2.3 RF-2.4
Descripción	El demonio o el administrador lanza la ejecución del algoritmo de backpropagation sobre la red neuronal para conseguir el pronóstico de los partidos.
Precondiciones	Los datos han sido recopilados previamente.
Acciones	1. Se realiza el cálculo de las rachas de los equipos. 2. La aplicación realiza el preprocesamiento previo a la entrada de las instancias en la red neuronal. 3. Se cargan los datos en la red neuronal y se realiza el entrenamiento. 4. Se lleva a cabo el test y se almacenan los datos obtenidos en la base de datos.
Postcondiciones	Ninguna
Excepciones	Si ya se ha realizado ya la carga de datos de los pronósticos en la base de datos no se lleva a cabo de nuevo.
Importancia	Alta

Cuadro B.2: Caso de uso “Red neuronal”.

Caso de uso	Visualizar interfaz
Requisitos	RF-3 RF-3.1 RF-3.2 RF-3.3 RF-3.4
Descripción	El sitio web recoge los datos proporcionados por la red neuronal y el algoritmo de scraping de las casas de apuestas.
Precondiciones	Los datos han sido recopilados previamente. Se han llevado a cabo el correspondiente cálculo de las jornadas.
Acciones	1. Se podrá observar un resumen global con los valores de cada jornada. 2. De manera individual cada jornada se puede observar más detalladamente. 3. El pronóstico de la siguiente jornada estará disponible para orientar al usuario. 4. Se podrá realizar una gestión de la interfaz visual, así como modificar textos o tablas.
Postcondiciones	Ninguna
Excepciones	Ninguna
Importancia	Media

Cuadro B.3: Caso de uso “Visualizar interfaz”.

Caso de uso	Scraping resultados
Requisitos	RF-1.1 RF-1.4
Descripción	Extraemos los datos de los partidos de fútbol ya disputados mediante web scraping.
Precondiciones	Los partidos deben haberse disputado.
Acciones	1. Se accede uno a uno a cada partido mediante la url. 2. Se extraen los datos de ambos equipos. 3. Se almacenan los datos en la base de datos.
Postcondiciones	Ninguna
Excepciones	No se recogen si se han recogido previamente.
Importancia	Alta

Cuadro B.4: Caso de uso “Scraping resultados”.

Caso de uso	Scraping clasificación
Requisitos	RF-1.2 RF-1.4
Descripción	Extraemos los datos de la clasificación para cada uno de los equipos mediante web scraping.
Precondiciones	Los partidos deben haberse disputado.
Acciones	1. Se accede a la clasificación donde se encuentran todos los datos de los equipos. 2. Se extraen los datos de la clasificación de cada equipo. 3. Se almacenan los datos en la base de datos.
Postcondiciones	Ninguna
Excepciones	No se recogen si se han recogido previamente.
Importancia	Alta

Cuadro B.5: Caso de uso “Scraping clasificación”.

Caso de uso	Scraping casas de apuestas
Requisitos	RF-1.3 RF-1.4
Descripción	Extraemos los datos de la clasificación para cada uno de los equipos mediante web scraping.
Precondiciones	Los partidos no deben haberse disputado. La recopilación de datos hay que hacerla en la fecha más cercana posible para tener los últimos datos.
Acciones	1. Accedemos uno a uno a los partidos de la jornada que está por disputar. 2. Se extraen las cuotas de las cuatro casas de apuestas que se encuentran en la web. 3. Se almacenan los datos en la base de datos.
Postcondiciones	Ninguna
Excepciones	No se recogen si se han recogido previamente.
Importancia	Baja

Cuadro B.6: Caso de uso “Scraping casas de apuestas”.

Caso de uso	Entrenamiento de la red
Requisitos	RF-2.1 RF-2.2 RF-2.3
Descripción	Obtenemos los datos de la BBDD , generamos las instancias y preprocesamos los datos antes de introducirlos a la red neuronal, finalmente se introducen para el entrenamiento.
Precondiciones	Los datos de partidos y la clasificación deben estar en la BBDD Se debe haber hecho el cálculo de las rachas.
Acciones	1. Accedemos a la BBDD y extraemos los datos necesarios. 2. Generamos las instancias. 3. Preprocesamos las instancias. 4. Introducimos los datos en la red neuronal.
Postcondiciones	Llevar acabo la predicción.
Excepciones	Ninguna.
Importancia	Alta

Cuadro B.7: Caso de uso “Entrenamiento de la red”.

Caso de uso	Predecir resultados
Requisitos	RF-2.4 RF-1.4
Descripción	Obtenemos los datos de la BBDD , generamos las instancias y preprocesamos los datos antes de introducirlos a la red neuronal, finalmente se introducen para el entrenamiento.
Precondiciones	Tener la red neuronal entrenada.
Acciones	1. Realizamos la predicción tres veces y nos quedamos con la media. 2. Almacenamos los datos obtenidos en la base de datos.
Postcondiciones	Ninguna.
Excepciones	Si se han almacenado los datos previamente no se vuelve a almacenar.
Importancia	Alta

Cuadro B.8: Caso de uso “Predecir resultados”.

Caso de uso	Administrar interfaz
Requisitos	RF-3.1
Descripción	El administrador puede realizar diferentes modificaciones respecto a la interfaz, utilizando las herramientas que Drupal proporciona y realizando cambios en los scripts.
Precondiciones	Tener permisos de administrador en Drupal.
Acciones	1. Modificar temas, módulos o bloques. 2. Se pueden editar los scripts.
Postcondiciones	Ninguna.
Excepciones	Ninguna.
Importancia	Media

Cuadro B.9: Caso de uso “Administrar interfaz”.

Caso de uso	Visualizar resultados/predicciones
Requisitos	RF-3.2 RF-3.3 RF-3.4
Descripción	El usuario puede observar aquellos datos que le resulten de interés como lo referente a jornadas pasadas y el pronóstico de la próxima jornada.
Precondiciones	Datos presentes en la base de datos. Se ha debido realizar las ejecuciones correspondientes del algoritmo de backpropagation.
Acciones	1. Visualizar el balance global de todas las jornadas. 2. Visualizar el balance detallado de cada jornada. 3. Visualizar la predicción de la próxima jornada.
Postcondiciones	Ninguna.
Excepciones	Ninguna.
Importancia	Media

Cuadro B.10: Caso de uso “Visualizar resultados/predicciones”.

Especificación de diseño

C.1. Introducción

En este apéndice vamos a tratar de todo aquello relacionado a la organización del proyecto.

C.2. Diseño de datos

A lo largo de este proyecto la cantidad de datos con la que hemos trabajado es muy grande, dada la necesidad de datos de una red neuronal para un buen pronóstico. En la sección vamos a ver como hemos almacenado estos datos y que uso hemos hecho de estos.

Almacenamiento de datos

Los datos han sido almacenados en el sistema de gestión de bases de datos relacionales MySQL, esta la hemos administrado utilizando PHPMyAdmin. Dada la gran cantidad de datos y su diversidad ha sido necesario trabajar con diferentes tablas, como podemos observar en el modelo relacional [C.1](#), aquí podemos observar todas las tablas y sus relaciones con la excepción de la tabla de partidos, que podemos observar en esta figura [C.2](#), dado que esta posee una enorme cantidad de campos.

Como se puede observar en el modelo relacional la tabla *balance_general* se encuentra apartada del resto, y es que esta tabla ha sido creada para evitar tener que hacer todos los cálculos de cada jornada al cargar la ventana *Informe general*.

A continuación se van a explicar cada una de las tablas utilizadas:

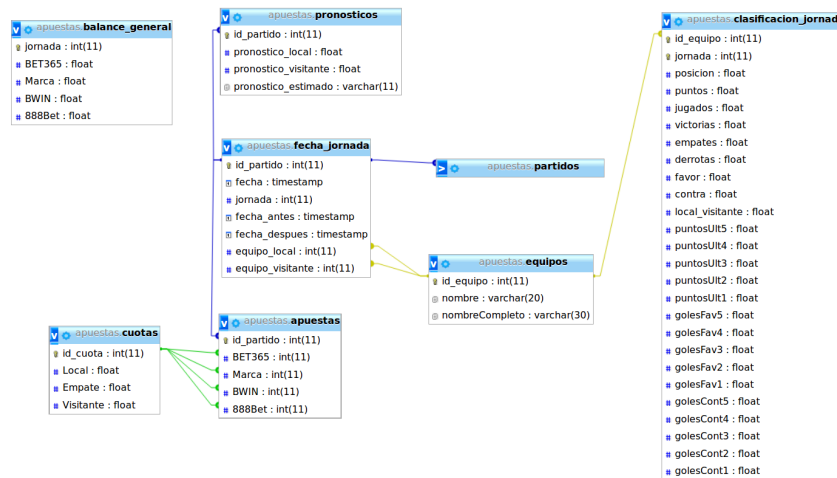


Figura C.1: Modelo relacional

- **equipos** En esta tabla encontramos los 20 equipos de primera división, se han dispuesto alfabéticamente y esta tabla posee tres campos, entre ellos está el nombre, este campo es muy importante a la hora de realizar el scrapping, ya que es así como identifica la web a cada equipo, el tercer y último campo es «nombreCompleto», utilizado para mostrar el nombre de dicho equipo en la interfaz.
- **fecha_jornada** Esta tabla contiene cada partido de cada jornada de la liga, con algunos campos como la fecha, la jornada o los equipos contendientes.
- **clasificacion_jornada** Aquí podemos observar la posición en la clasificación de cada uno de los equipos en cada jornada, así como las últimas rachas de los equipos. Esta tabla resulta fundamental a la hora de insertar datos en la red neuronal.
- **partidos** Mediante las técnicas de scraping extraemos datos de cada partido, estos datos son almacenados aquí y contiene todas las estadísticas de ambos equipos. Los datos son utilizados para la entrada en la red neuronal.
- **pronosticos** Esta tabla contiene la salida de datos de la red neuronal y el resultado final obtenido por esta para cada uno de los partidos.
- **apuestas** Organiza todas las cuotas de las casas de apuestas para cada partido, registrando todos los identificadores.
- **cuotas** Posee las cuotas y sus valores dependiendo de la victoria local, empate o victoria visitante.

  apuestas. partidos
 id_partido : int(11)
posesion_local : float
posesion_visitante : float
goles_local : int(11)
goles_visitante : int(11)
remates3p_local : int(11)
remates3p_visitante : int(11)
rematesfuera_local : int(11)
rematesfuera_visitante : int(11)
remates_local : int(11)
remates_visitante : int(11)
paradas_local : int(11)
paradas_visitante : int(11)
corners_local : int(11)
corners_visitante : int(11)
outsides_local : int(11)
outsides_visitante : int(11)
amarillas_local : int(11)
amarillas_visitante : int(11)
rojas_local : int(11)
rojas_visitante : int(11)
asistencias_local : int(11)
asistencias_visitante : int(11)
palos_visitante : int(11)
lesiones_local : int(11)
lesiones_visitante : int(11)
sustituciones_local : int(11)
sustituciones_visitante : int(11)
faltas_local : int(11)
faltas_visitante : int(11)

Figura C.2: Tabla de partidos

- **balance_general** Almacena el balance de cada casa de apuestas en cada jornada, esta tabla se utiliza posteriormente para mostrarla en la interfaz.

C.3. Diseño procedimental

En esta sección se va a poder observar el funcionamiento del proyecto.

Diagrama de flujo

En esta imagen [C.3](#) podemos ver la ejecución del programa y de su dependencia del día en el que nos encontremos. Los campos *fecha_antes* y *fecha_despues* son el día previo y posterior a una jornada. Usualmente es trata del viernes a las 00:00 y de Martes a las 00:00, esto es así por que el primer partido de la jornada suele disputarse el viernes a las 20:45 y el último el lunes a las 20:45.

Una vez ejecutados los algoritmos de backpropagation y del scraping de casas de apuestas, el usuario ya podrá acceder al informe pronóstico donde gozará de toda la información disponible.

En cuanto se ejecutan los algoritmos de scraping de resultados y clasificación y cálculo de rachas, el usuario ya podrá acceder sin problemas a la información actualizada de como ha ido la jornada, ya que se actualizará la ventana de informe de jornada y a su vez el informe global.

C.4. Diseño arquitectónico

Se ha utilizado una arquitectura cliente servidor, esta consiste en un cliente que realiza peticiones a otro «servidor» el cual le proporciona la respuesta. Actualmente esta es la forma de trabajo más extendida en la comunicación por redes.

El servidor espera solicitudes en un puerto, el cliente es quien realiza estas solicitudes, a su vez el cliente reserva un puerto desde el cual realizar las solicitudes. Utilizando esta arquitectura el cliente tiene la libertad de obtener la información que desea cuando quiera y de diferentes orígenes.[\[1\]](#)

El modelo utilizado en este proyecto ha sido el de tres niveles, similar al que podemos observar en esta figura [C.4](#), incluimos una base de datos donde incluye y extrae datos el servidor. Nuestro servidor es un Apache y la base de datos elegida es MySQL. En la arquitectura de tres niveles, las aplicaciones a nivel de servidor son descentralizadas, permitiendo un mayor grado de flexibilidad, mayor seguridad y mejor rendimiento.

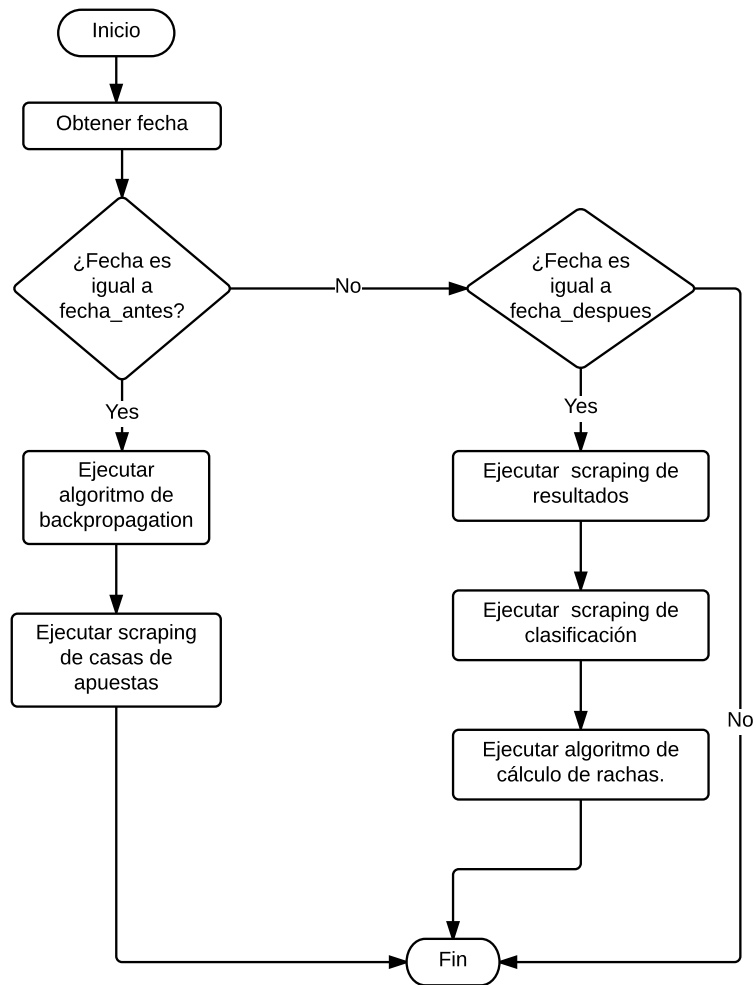


Figura C.3: Diagrama de flujo

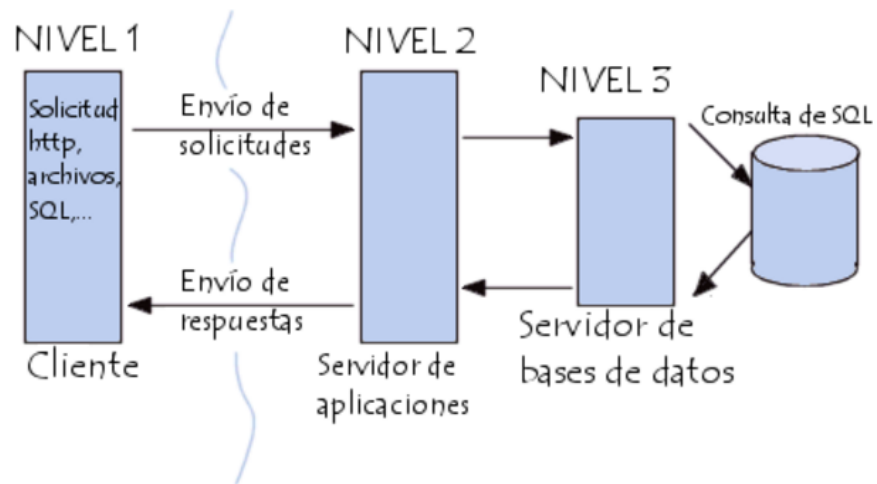


Figura C.4: Arquitectura de tres niveles Cliente servidor

Documentación técnica de programación

D.1. Introducción

En este apéndice se va a realizar una descripción más avanzada del proyecto, para aquella persona que desee continuar su desarrollo o conocer el funcionamiento más a fondo.

D.2. Estructura de directorios

En esta sección vamos a dar un sentido a la organización de las carpetas del trabajo.

- **algorithms:** En este directorio vamos a encontrar todos los algoritmos realizados a lo largo del proyecto, dos ficheros son el algoritmo de back-propagation y el encargado de hacer el cálculo de rachas, el resto tienen una nomenclatura particular:
 - Scraping: Los ficheros que empiezan así, son los ficheros utilizados para el web scraping.
 - Informe: Estos scripts son los encargados de mostrarnos por la interfaz las tablas correspondientes.
- **database:** Contiene la última versión de la base de datos.
- **images:** Este directorio tiene las imágenes que se muestran en la interfaz.

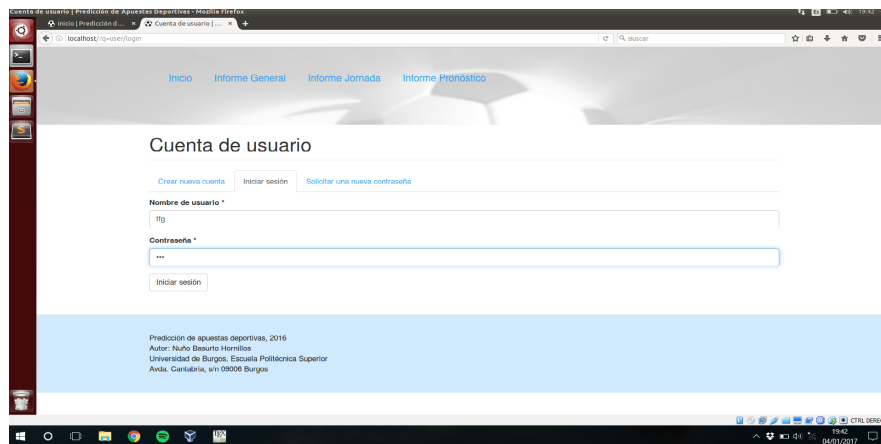


Figura D.1: Loguearse en Drupal

- **otros:** El resto de directorios son propios de Drupal, apenas se han hecho cambios en estos ya que no ha sido necesario. En el directorio `sites/all/themes`, se ha añadido el tema de bootstrap que podemos observar en la interfaz

D.3. Manual del programador

En esta sección vamos a ver diversas acciones que nos van a permitir un uso más avanzado del trabajo desarrollado.

Para el manejo de la interfaz web, es necesario un cierto conocimiento de uso del gestor Drupal.

Administración interfaz web

Para la administración de la misma, lo primero es logearse como administrador en Drupal, para ello hay que acceder a la página de login:

`http://localhost/?q=user/login`

Introducimos el usuario *tfg* y la contraseña *tfg*. Una vez dentro si queremos ejecutar algún algoritmo, por que por ejemplo en el momento de ejecución del demonio el servidor no estaba activo, accederemos al panel de la izquierda llamado Navegación y pulsaremos en Administrar apuestas, aquí nos mostrará los diferentes posibilidades de ejecución

En la parte de arriba en un color grisáceo oscuro tenemos los atajos **D.3**, el primero «Agregar contenido» nos permite añadir nuevas páginas donde ejecu-

APÉNDICE D. DOCUMENTACIÓN TÉCNICA DE PROGRAMACIÓN²⁵



Figura D.2: Administración de apuestas



Figura D.3: Atajos en Drupal

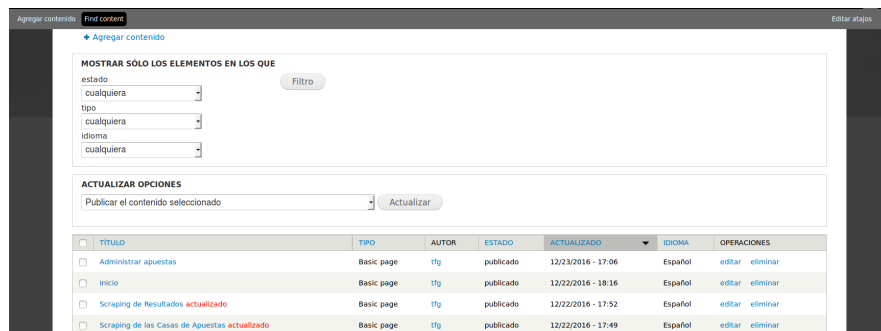


Figura D.4: Editar contenidos

tar algoritmos, find content nos permite realizar ejecuciones, modificaciones o eliminar los contenidos ya existentes^{D.4}. Si por ejemplo deseamos editar un fichero, en este caso *Scraping de resultados*, nos encontramos con este código:

```
<?php
include_once 'algorithms/Scraping_resultados.php'
include_once 'algorithms/Scraping_clasificacion.php'
```

```
include_once 'algorithms/Calculo_rachas.php'
?>
```

De esta sencilla manera llama a los algoritmos que deseamos, ahora toca añadir unas líneas especiales dentro de los scripts, para que Drupal pueda acceder a ellos:

```
define('DRUPALROOT', getcwd());
require_once DRUPALROOT . '/includes/bootstrap.inc';
drupal_bootstrap(DRUPAL_BOOTSTRAP_FULL);
```

Está hecho de esta forma para no tener que copiar todo el código del script, en el contenido de Drupal. De esta forma se puede utilizar un buen editor de código fuente, como Sublime Text, en vez de usar el editor de texto de Drupal que complica mucho y ralentiza la programación.

Este ha sido un pequeño resumen de como realizar las modificaciones básicas en Drupal, si se desea aprender más a fondo las modificaciones que se pueden realizar, es recomendable el libro [DRUPAL 7 - Bibliografía] y ante dudas acudir a la comunidad de Drupal, que expone sus dudas en <https://www.drupal.org/forum>.

D.4. Compilación, instalación y ejecución del proyecto

En esta sección se va a hablar de aquellos pasos que hay que llevar a cabo para poder trabajar sobre el el proyecto y la implementación del código.

La instalación va a realizarse sobre la distribución de Linux Ubuntu, en concreto sobre su versión 14.04. Bien puede realizarse sobre una máquina virtual o como anfitrión.

LAMP

LAMP es el acrónimo de la infraestructura de internet que trabaja con las siguientes herramintas:

- Linux: Como sistema operativo.
- Apache: El servidor web.
- MySQL: Como gestor de base de datos, aunque también puede instalarse MariaDB
- PHP: Lenguaje de programación, aunque también esta aceptado el uso de Perl y Python. [Referencia LAMP Wikipedia]

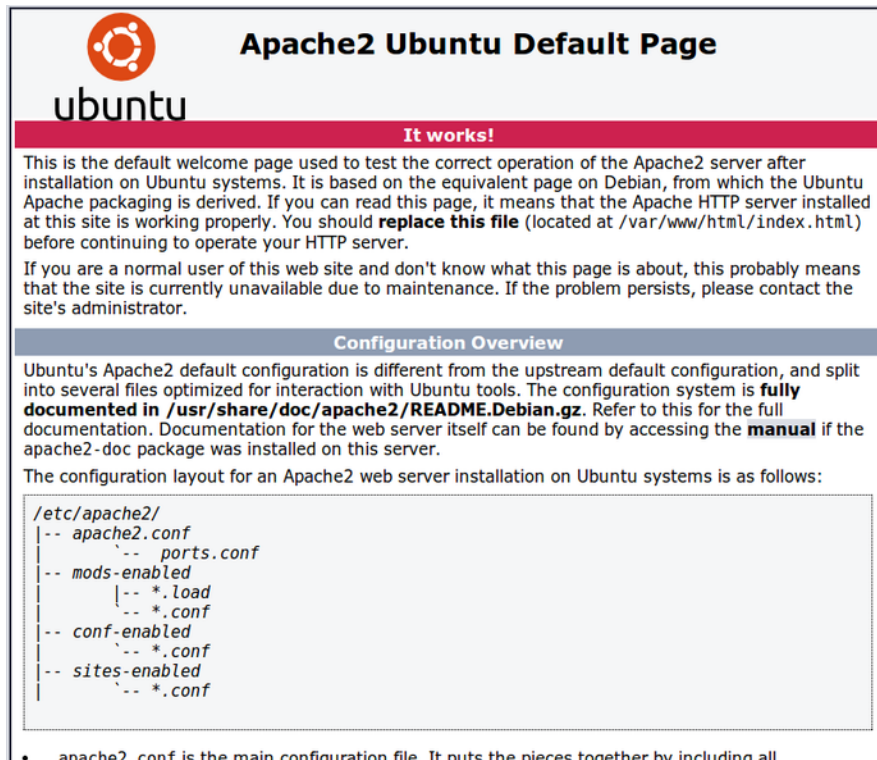


Figura D.5: Apache en funcionamiento

Apache

El primer paso va a ser la instalación del servidor web Apache en nuestro sistema. El servidor puede ser instalado perfectamente a través de la shell de Linux:

```
$ sudo apt-get update
$ sudo apt-get install apache2
```

Una vez realizado esto ya tenemos el servidor instalado, para comprobar que funciona correctamente entramos en el navegador y ponemos:

`http://localhost`

Tras ello deberíamos ver:

MySQL

Ahora vamos a proceder a instalar el gestor de bases de datos MySQL, para ello escribimos en el terminal el siguiente comando:

```
$ sudo apt-get install mysql-server mysql-client
```

Es una instalación sencilla, debemos recordar la contraseña y el usuario utilizados. Ahora vamos a proceder a crear la base de datos en MySQL y seguidamente un script de seguridad.

```
$ sudo mysql_install_db
$ sudo mysql_secure_installation
```

PHP

Lo primero es instalar PHP y para ello vamos a ejecutar el siguiente comando en la terminal:

```
$ sudo apt-get install libapache2-mod-php5 php5 php5-mcrypt
```

Tras ello procedemos a instalar algunos módulos de PHP:

```
$ sudo apt-get install php5-cgi php5-cli php5-common php5-curl php5-dbg ph
```

Podemos comprobar que se ha instalado correctamente creando el fichero:

```
$ sudo nano /var/www/html/info.php
```

Insertándole el código:

```
<?php
    phpinfo();
?>
```

Listing D.1: Código par comprobar el funcionamiento de PHP”.

Finalmente en el navegador ponemos:

```
http://localhost/info.php
```

La imagen que nos encontremos debería ser similar a esta:

Ahora tenemos que activar el módulo de PHP para Apache, para ello en el fichero

```
$sudo nano /etc/apache2/mods-enabled/php2.conf
```

Hay que cambiar la línea a On.

```
php_admin_value engine On
```

Tras el cambio es muy importante reinicia el servicio de apache así:

```
$ sudo service apache2 restart
```



<div> <div>PHP Version 5.5.9-1ubuntu4.20</div>  </div>	
System	Linux nuno-VirtualBox 4.4.0-57-generic #78~14.04.1-Ubuntu SMP Sat Dec 10 00:14:47 UTC 2016 x86_64
Build Date	Oct 3 2016 13:00:15
Server API	Apache 2.0 Handler
Virtual Directory Support	disabled
Configuration File (php.ini) Path	/etc/php5/apache2
Loaded Configuration File	/etc/php5/apache2/php.ini
Scan this dir for additional .ini files	/etc/php5/apache2/conf.d
Additional .ini files parsed	/etc/php5/apache2/conf.d/05-opcache.ini, /etc/php5/apache2/conf.d/10-pdo.ini, /etc/php5/apache2/conf.d/20-curl.ini, /etc/php5/apache2/conf.d/20-gd.ini, /etc/php5/apache2/conf.d/20-json.ini, /etc/php5/apache2/conf.d/20-mcrypt.ini, /etc/php5/apache2/conf.d/20-mysql.ini, /etc/php5/apache2/conf.d/20-mysqli.ini, /etc/php5/apache2/conf.d/20-pdo_mysql.ini, /etc/php5/apache2/conf.d/20-readline.ini

Figura D.6: Instalación PHP

phpMyAdmin

Una vez instalado LAMP, vamos a instalar el gestor de MySQL phpMyAdmin, que nos va a ayudar a trabajar en la base de datos. Lo primero es instalar su paquete:

```
$ sudo apt-get install phpmyadmin
```

En el paso de la instalación que pregunta por la utilización de dbconfig-common hay que indicar «yes». Tras ello habilitamos la extensión php5-mcrypt:

```
$ sudo php5enmod mcrypt
```

Una vez realizados estos pasos reiniciamos el servicio apache como hicimos anteriormente:

```
$ sudo service apache2 restart
```

Tras ello ya podemos acceder a través del navegador con:

```
http://localhost/phpmyadmin
```

Drupal 7

Lo primero es descargarse la última versión de Drupal 7 de la página <http://www.drupal.org>, descargamos sobre la raíz del sistema. Una vez esté este

fichero en la raíz al poner la url en el navegador: <http://localhost/admin>, nos encontraremos con una interfaz con la cual la instalación de Drupal nos resultará bastante sencilla. En la primera pantalla escogemos que queremos un perfil Standard y continuamos. Más adelante nos pedirá el nombre de la base de datos, es necesario que esta exista ya en nuestro servidor para que Drupal pueda ser instalado. Y de esta manera sencilla nos encontramos con Drupal 7 instalado.

Una vez instalado Drupal 7 podemos seleccionar diferentes módulos y temas para añadir al gestor de contenido. En el desarrollo de este proyecto se intentó trabajar con algunos módulos como el de Trello pero este tenía problemas con la autenticación. En cambio si que se instaló satisfactoriamente el tema de Bootstrap con el cual nos encontramos en la interfaz. Para instalar un tema es aconsejable descargarlo desde http://www.drupal.org/project/project_theme y tras ello descomprimirlo en la carpeta `/sites/all/themes` tras ello si accedemos en nuestro navegador a la dirección: <http://localhost/admin/appearance> nos lo encontraremos así, para los módulos es similar, solo que hay que almacenar el módulo descargado y descomprimido en `/sites/all/modules` y tras ello acceder a la página <http://localhost/admin/modules>.

D.5. Pruebas del sistema

Dado que hemos realizado un diseño web centrado en el usuario es necesario realizar una serie de pruebas centradas en el usuario. Se ha realizado a un total de 10 usuarios con la edad comprendida entre los 22 y 56 años.

Las pruebas se realizaron con fecha de 27 de Diciembre de 2016. La última jornada disputada era la 16 y la que estaba pronosticada era la 17. Se propusieron una serie de cuestiones en un orden determinado y se observaron las acciones de los usuarios.

- Acceder al pronóstico de la próxima jornada. **9/10**
- Acceder al balance global de todas las jornadas. **10/10**
- Obtener el balance global de la jornada 14 de BET365. **10/10**
- Acceso al balance de la última jornada. **7/10**
- Acceder al balance de la jornada 14. **5/10**

Con este sencillo test nos hemos encontrado con la dificultad de algunos usuarios para acceder al balance de una jornada particular y es que, para este acceso los usuarios debían acceder a *Informe Global* y desde ahí picar en la jornada correspondiente, dado que los usuarios estaban en la página *Informe*

Jornada, buscaban algún enlace desde la jornada que ahí teníamos y la jornada 14.

Las conclusiones extraídas del test nos indican la necesidad de poner enlaces en *Informe Jornada* al resto de jornadas, para evitar volver a *Informe Global*. En cuanto al resto del sitio web al ser sencillo apenas genera problemas al usuario.

Documentación de usuario

E.1. Introducción

En este apéndice vamos a ver como instalar y realizar todas las acciones necesarias para poder ejecutar el proyecto finalmente.

E.2. Requisitos de usuarios

El objetivo de esta sección es indicar aquellos requisitos necesarios para que el usuario pueda utilizar el proyecto.

Requisitos software

Es necesario tener instalado el gestor de máquinas virtuales Oracle VM Virtual Box. Este gestor puede ser utilizado prácticamente en cualquier sistema operativo, soportando la mayoría de sistemas Windows, Linux, Mac OS X y Solaris. Dado que la máquina es de 64 bits, en ocasiones nos encontramos con que Virtual Box no nos permite crearla, por ello es necesario tener activadas las opciones de virtualización desde la BIOS.

Requisitos hardware

En lo concerniente al hardware necesitamos un sistema que pueda correr sin problemas la máquina virtual. Como requisito mínimo es necesario un sistema cuya memoria RAM disponible para la máquina sea de 512 MB de RAM. En cuanto al espacio libre de disco es necesario al menos 20 GB. El procesador debe ser un x86, cualquier Intel o AMD reciente debe valer[2].

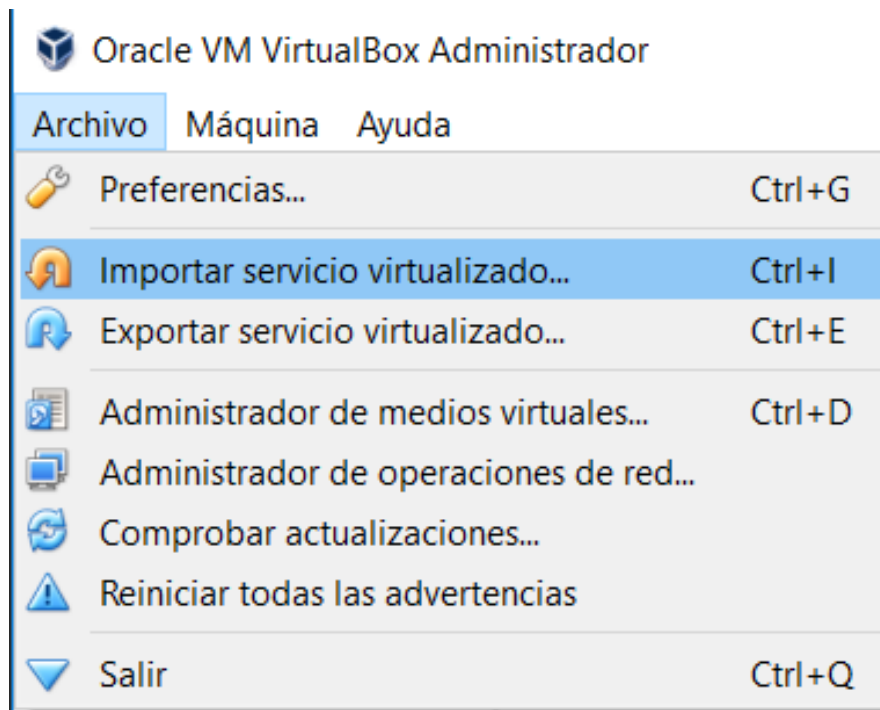


Figura E.1: Importación de servicio virtual.

E.3. Instalación

Se va a proporcionar un disco que contiene la máquina en el formato *Open Virtualization Format* «OVF», que es un servicio virtualizado de la máquina virtual. En Oracle VM VirtualBox en la parte superior pulsamos *Archivo* e *Importar servicio virtualizado* como vemos en la figura E.1, se abrirá una ventana donde debemos seleccionar la ubicación del servicio virtualizado. Automáticamente se pasará a una ventana en las preferencias de la máquina por defecto como podemos ver en la figura E.2, y pulsamos *Importar*. Por defecto tenemos 1 GB de RAM, que resulta suficiente para que funcione la máquina, si aumentamos la RAM veremos como aumenta la fluidez del servicio virtualizado.

Una vez iniciado es tan sencillo como abrir el Mozilla Firefox, cuyo icono encontramos a la izquierda en la barra de navegación de Ubuntu. Automáticamente debería abrirse en la página de inicio de nuestro sitio web, encaso de que no lo haga tecleamos en la barra de búsqueda <http://localhsot/inicio>. Una vez ya dentro, podremos utilizar el sitio web destinado.

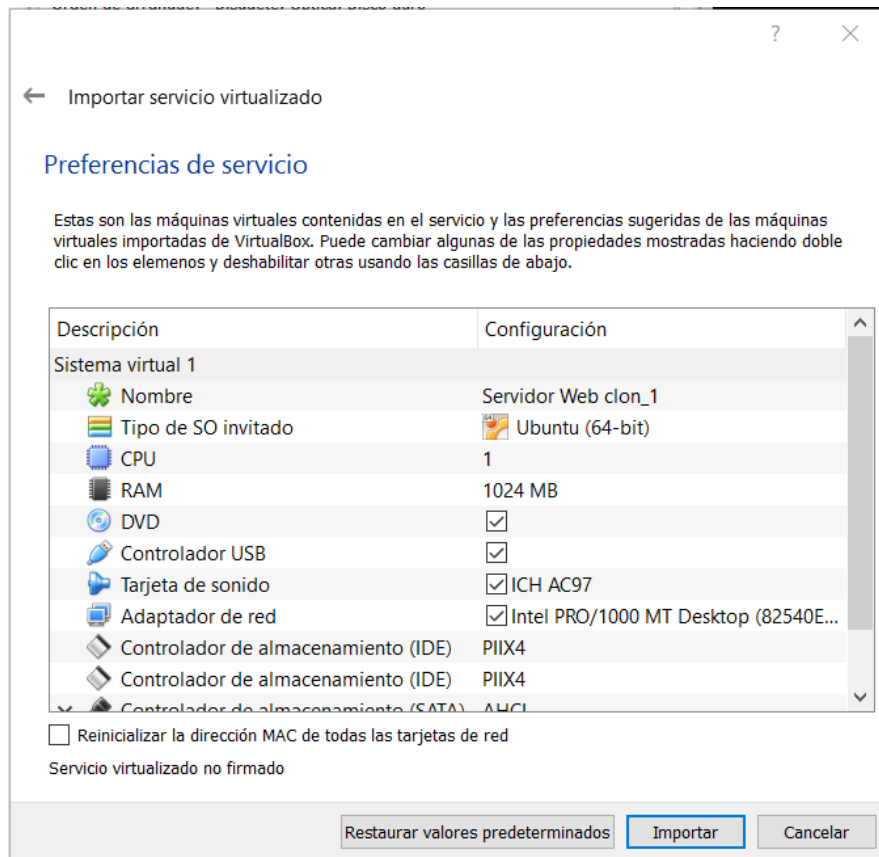


Figura E.2: Preferencias del servicio virtual.

E.4. Manual del usuario

El uso por parte del usuario de la aplicación web es muy sencilla, ya que las ejecuciones de los algoritmos están programadas por lo que no requiere intervención del usuario. El usuario se limita a observar los datos proporcionados por la interfaz, la interfaz es de sencillo uso y a continuación vamos a ver como utilizarla.

En la parte superior nos encontramos con una menú donde son 4 las opciones, como podemos observar en esta imagen [E.3](#), y que vamos a describir brevemente:

- **Inicio:** En esta opción nos encontramos con una breve descripción del proyecto y de las herramientas utilizadas.
- **Informe General:** En este caso podemos observar del beneficio o pérdida obtenido en cada jornada y el balance global.



Figura E.3: Menú con el que puede interactuar el usuario.



Figura E.4: Vista del usuario de Inicio.

- **Informe Jornada:** Si accedemos a través del menú podemos observar lo obtenido en una única jornada.
- **Informe Pronóstico:** Nos proporciona los resultados que se esperan y las cuotas de las casas de apuestas.

Como podemos observar en la imagen, en esta primera página nos encontramos con la descripción del proyecto [E.4](#).

En la ventana de informe general [E.5](#), podemos observar una tabla con el balance de cada una de las jornadas, en cada una de las casas de apuestas, pulsando en la jornada accedemos a los partidos de la jornada más detalladamente, los cuadros en verde son aquellos en los que se ha obtenido un balance positivo, en cambio encontramos en amarillo donde se pierde el dinero. Recordar que este es un balance por euro apostado. Por ejemplo en la jornada 14 observamos en la casa BET365 1.45, entonces si apostásemos 10 en total (1 por partido) acabaríamos el día con 11.45, el 1.45 es el beneficio. En cambio en la jornada 15 en la misma casa de apuestas observamos -2.60, en este caso acabaríamos la jornada con 8.4, perdiendo dicho dinero.

A la ventana de Informe Jornada tenemos dos maneras de acceder, a través del menú donde observaremos el informe de la última jornada, o bien a través de los hipervínculos que encontramos en el Informe General, donde podemos

Jornada	bet365	888sport	bwin	888sport
1	-1.20	-1.24	-1.13	-1.09
2	4.80	5.16	4.71	4.75
3	-0.15	-0.05	-0.21	-0.08
4	0.85	1.04	0.82	0.89
5	0.30	0.27	0.29	0.22
6	-3.75	-3.81	-3.74	-3.78
7	-5.10	-5.07	-4.92	-4.87
8	0.75	0.89	0.60	0.94
9	-3.39	-3.29	-3.14	-3.27
10	-1.40	-1.36	-1.31	-1.44
11	2.02	2.90	2.56	2.64
12	-1.50	-2.00	-1.00	-1.90
13	7.44	7.41	7.48	7.44
14	1.45	1.45	1.80	1.50
15	-2.60	-2.67	-2.76	-2.66
16	-2.03	-2.03	-2.15	-2.00
Balance final:	-3.51	-2.40	-2.10	-2.71

Figura E.5: Vista del usuario del Informe de la Jornada.

localhost/informe-jornada								Buscar	
Informe de la Jornada									
Jornada 16									
Partido	Resultado	Pronóstico	bet365	888sport	bwin	888sport			
Alavés-Real Betis	1-0	1	2.30	2.30	2.25	2.30			
Athletic-Celta	2-1	X	-1.00	-1.00	-1.00	-1.00			
Barcelona-Espanyol	4-1	X	-1.00	-1.00	-1.00	-1.00			
Deportivo-Osasuna	2-0	1	1.57	1.57	1.60	1.60			
Valencia-Real Madrid	0-0	2	-1.00	-1.00	-1.00	-1.00			
Sevilla-Málaga	4-1	X	-1.00	-1.00	-1.00	-1.00			
Atlético-Las Palmas	1-0	X	-1.00	-1.00	-1.00	-1.00			
Sporting-Villarreal	1-3	X	-1.00	-1.00	-1.00	-1.00			
Granada-R. Sociedad	0-2	1	-1.00	-1.00	-1.00	-1.00			
Leganes-Eibar	1-1	X	4.10	4.10	4.00	4.10			
Balance			-2.03	-2.03	-2.15	-2.00			

Figura E.6: Vista del usuario del Informe Jornada.

acceder al resto de jornadas. Podemos ver en negrita aquellos pronósticos que se han acertado, a su vez las casas de apuestas nos muestran lo predecido, mientras que aquellos partidos errados aparecen con -1 ya que el euro apostado en el partido se ha perdido. En la parte de abajo podemos observar el balance que veremos en amarillo si es negativo y en verde si es positivo.

En la imagen E.6, observamos como la url es limpia por lo que nos muestra la última jornada a la que se ha accedido por el menú superior, en cambio si se ha accedido por el informe general veremos la imagen ?? con la URL diferente.

En el informe del pronóstico E.8, podemos observar todas las cuotas de los partidos para el pronóstico realizado. Y las ganancias potenciales que tenemos

Informe de la Jornada

Jornada 15

Partido	Resultado	Pronóstico	bet365	BETDEJES	bwin	888
Celta-Sevilla	0-3	1	-1.00	-1.00	-1.00	-1.00
Espanyol-Sporting	2-1	1	1.73	1.70	1.72	1.72
Málaga-Granada	1-1	1	-1.00	-1.00	-1.00	-1.00
Osasuna-Barcelona	0-3	2	1.17	1.18	1.18	1.20
Real Madrid-Deportivo	3-2	1	1.10	1.10	1.09	1.12
R. Sociedad-Valencia	3-2	X	-1.00	-1.00	-1.00	-1.00
Villarreal-Atlético	3-0	2	-1.00	-1.00	-1.00	-1.00
Real Betis-Athletic	1-0	2	-1.00	-1.00	-1.00	-1.00
Elbar-Alavés	0-0	X	3.40	3.35	3.25	3.30
Las Palmas-Leganés	1-1	1	-1.00	-1.00	-1.00	-1.00
Balance			-2.60	-2.67	-2.76	-2.66

Figura E.7: Vista del usuario del Informe de la Jornada con la URL diferente.

Informe del Pronóstico

Jornada 17

Partido	Pronóstico	bet365	BETDEJES	bwin	888
Athletic - Alavés	1	1.62	1.57	1.57	1.60
Celta - Málaga	1	1.80	1.85	1.87	1.85
Espanyol - Deportivo	X	3.25	3.15	3.25	3.10
Osasuna - Valencia	1	3.75	3.80	3.90	3.80
Real Madrid - Granada	1	1.05	1.01	1.10	1.10
R. Sociedad - Sevilla	2	3.00	3.00	3.10	3.10
Villarreal - Barcelona	2	1.53	1.53	1.50	1.55
Real Betis - Leganés	X	3.25	3.20	3.30	3.20
Elbar - Atlético	X	3.50	3.50	3.60	3.75
Las Palmas - Sporting	1	1.53	1.55	1.57	1.55
Ganancias Potenciales:		24.28	24.16	24.76	24.60

Figura E.8: Vista del usuario del Informe del pronóstico.

una vez finalizada la jornada se realizará el recuento y esta jornada pasará al informe de la jornada, poniendo aquí los partidos de la nueva jornada a la espera del pronóstico y de las cuotas de las casas de apuestas.

Bibliografía

- [1] EcuRed. Arquitectura cliente servidor, 2016.
- [2] Oracle. Requisitos virtualbox, 2016.
- [3] Seguridad Social. Bases y tipos de cotización, 2016.
- [4] Wikipedia. Latex — wikipedia, la enciclopedia libre, 2015. [Internet; descargado 30-septiembre-2015].
- [5] Wikipedia. Apache license — wikipedia, la enciclopedia libre, 2016. [Internet; descargado 10-enero-2017].
- [6] Wikipedia. Copyleft — wikipedia, la enciclopedia libre, 2016. [Internet; descargado 10-enero-2017].
- [7] Wikipedia. Gnu general public license — wikipedia, la enciclopedia libre, 2016. [Internet; descargado 10-enero-2017].
- [8] Wikipedia. Licencia php — wikipedia, la enciclopedia libre, 2016. [Internet; descargado 10-enero-2017].