



**UNIVERSIDAD
DE BURGOS**

TFG del Grado en Ingeniería Informática

**GII_O_MA_20.04 Mejora de la
herramienta web dinámica para la
captación, tratamiento y
presentación de datos relacionados
con la coyuntura económica de
Burgos**

Presentado por Raquel Sancha Sánchez
en Universidad de Burgos — 18 de febrero
de 2021

Tutores: Bruno Baruque Zanon
Santiago Porras Alfonso
Carlos López Nozal



UNIVERSIDAD DE BURGOS

D. Bruno Baruque Zanon, profesor del departamento de Lenguajes y Sistemas Informáticos, área de Ingeniería Informática.

Expone:

Que el alumno D. Raquel Sancha Sánchez, con DNI 71567050C, ha realizado el Trabajo final de Grado en Ingeniería Informática titulado Mejora de la herramienta web dinámica para la captación, tratamiento y presentación de datos relacionados con la coyuntura económica de Burgos.

Y que dicho trabajo ha sido realizado por el alumno bajo la dirección del que suscribe, en virtud de lo cual se autoriza su presentación y defensa.

En Burgos, 18 de febrero de 2021

Vº. Bº. del Tutor:

Vº. Bº. del co-tutor:

D. nombre tutor

D. nombre co-tutor

Resumen

Este trabajo es una ampliación de la primera versión del TFG llamado *Herramienta web dinámica para la captación, tratamiento y presentación de datos relacionados con la coyuntura económica de Burgos* [Link al proyecto en Github](#).

Este proyecto fue creado para facilitar el trabajo del equipo que todos los años realiza unos boletines de coyuntura económica para informar de la situación de la economía burgalesa.

Esta aplicación permite la entrada de datos y su clasificación.

El objetivo del presente proyecto es realizar mejoras en el diseño de la aplicación y la corrección de errores o malas prácticas del código así como añadir nuevas funcionalidades como la introducción de datos directamente de la página del INE o la capacidad de exportar las tablas a Excel.

Descriptores

Economía, coyuntura económica, boletín, MySQL, Laravel, PHP, variables, aplicación web, Instituto Nacional de Estadística, predicción

...

Abstract

This work is an extension of the first version of the TFG called *Dynamic web tool for the collection, treatment and presentation of data related to the economic situation of Burgos* [Link to the project on Github](#).

This project was created to facilitate the work of the team that every year produces economic newsletters to report on the situation of the Burgos economy.

This application allows data entry and classification.

The objective of this project is to make improvements in the design of the application and the correction of errors or bad practices in the code as well as to add new functionalities such as entering data directly from the INE page or the ability to export the tables to Excel.

Keywords

Economy, economic situation, newsletter, MySQL, Laravel, PHP, variables, web application, National Institute of Statistics, prediction

Índice general

Índice general	III
Índice de figuras	V
Índice de tablas	VI
Introducción	1
Objetivos del proyecto	3
2.1. Objetivos generales	3
2.2. Objetivos técnicos	3
2.3. Objetivos personales	4
Conceptos teóricos	5
3.1. Encriptado de contraseñas	5
3.2. Datos abiertos	7
3.3. Algoritmos de predicción de datos	8
Técnicas y herramientas	11
4.1. Técnicas	11
4.2. Herramientas	12
4.3. Librerías	15
Aspectos relevantes del desarrollo del proyecto	19
5.1. Inicio del proyecto: Aprendizaje	19
5.2. Gestión de los errores	20
5.3. Ampliación de funcionalidades	22
5.4. Problemas	34

Trabajos relacionados	37
6.1. Herramienta web dinámica para la captación, tratamiento y presentación de datos relacionados con la coyuntura económica de Burgos	37
6.2. JSON-Stat	38
Conclusiones y Líneas de trabajo futuras	39
7.1. Conclusiones técnicas	39
7.2. Conclusiones personales	39
7.3. Líneas de trabajo futuras	40
Bibliografía	41

Índice de figuras

3.1. Ejemplo de la utilización de password_hash()	6
3.2. Encriptación de contraseñas en mi aplicación	7
4.3. Opciones que ofrece codacy para el análisis de nuestro código	15
4.4. DebugBar para Laravel	17
5.5. Sección Administración de usuarios	21
5.6. Editar datos de un usuario	21
5.7. Ejemplo de una tabla del INE	23
5.8. Datos de una tabla del INE	23
5.9. Salida de los datos en formato JSON	24
5.10. Estructura de la url	24
5.11. Estructura de una petición REST de Eurostat	25
5.12. Conjuntos de datos del Eurostat	26
5.13. Búsqueda de datos desde la API	27
5.14. Algunos conjuntos de datos de la aplicación	28
5.15. Gráfico de la comparación de la predicción con los valores reales	29
5.16. Gráfico de la comparación de la predicción con los valores reales	30
5.17. Gráfico de la comparación de la predicción con los valores reales	30
5.18. Gráfico de la comparación de la predicción con los valores reales	31
5.19. Gráfico de la comparación de la predicción con los valores reales	31
5.20. Gráfico de la comparación de la predicción con los valores reales	32
5.21. Captura de un problema típico al intentar instalar un paquete nuevo	34
5.22. Configuración de los hosts virtuales	35
5.23. Archivo de configuración .env de laradock	36
6.24. Variables de la aplicación	37
6.25. Tabla de una de las variables de la aplicación	38

Índice de tablas

5.1. Comparativa de las plataformas de datos abiertos	28
5.2. Comparativa de los algoritmos de predicción de datos	33

Introducción

Todos los años, el Departamento de Economía Aplicada de la Universidad de Burgos realiza un boletín exhaustivo sobre la coyuntura económica en el ámbito burgalés ([Enlace a algunos boletines](#)).

Estos boletines los crea el Equipo multidisciplinar de Coyuntura radicado en la Facultad de Ciencias Económicas y Empresariales de la Universidad de Burgos. En virtud del Convenio Marco de Colaboración firmado por la Universidad de Burgos y la actual Caja Viva Caja Rural.

Este equipo multidisciplinar integrado por 16 profesores, de los Departamentos de Economía Aplicada, Economía y Administración de Empresas y Derecho analiza la evolución económica coyuntural de la provincia de Burgos.

Para ello recopila datos de distintas fuentes: Instituto Nacional de estadística (INE), Banco Nacional de España, Eurostat...

El objetivo de la realización de los boletines de coyuntura económica es conocer el desarrollo de la economía del ámbito estudiado, mediante el análisis de información económica y su divulgación a un amplio público de empresas, profesionales y particulares.

Aunque el cliente final de la aplicación es este equipo, también se permitirá el acceso a invitados para que puedan ver la información recogida por lo que existe un sistema de roles con distintos permisos que se explicarán más adelante.

La aplicación se creó en 2017 para usarse como una herramienta que permita organizar y tratar dichos datos, así como almacenarlos y facilitar el trabajo al departamento que los recoge.

La aplicación proporciona una interfaz web que permite la entrada y el almacenamiento de variables e incluye herramientas para la visualización de datos entre otras funcionalidades.

A partir de estos datos se generan tablas que podrán ser filtradas y mostradas

al usuario de la manera que a éste le resulte mas cómoda y utilizarlas para realizar el boletín.

La aplicación también genera gráficos que muestran los datos de una forma sencilla e intuitiva.

Los invitados que accedan a la aplicación deberán tener un rol solo de lectura, por lo que sólo podrán filtrar y visualizar las tablas y gráficos de las variables económicas que están almacenadas en la base de datos de la aplicación.

Los encargados de introducir datos estadísticos, modificarlos o borrar los que estén obsoletos o erróneos, serán los administradores de la aplicación.

Este trabajo consiste en mejorar dicha aplicación y realizar un proceso como es el mantenimiento de software en un entorno profesional.

La aportación que hace este trabajo a la aplicación se basa en realizar algunas mejoras, corregir errores y añadir nuevas funcionalidades como la de introducir datos de forma automatizada desde el INE o la predicción de datos.

Objetivos del proyecto

2.1. Objetivos generales

- Identificar los errores existentes en la aplicación.
- Corregir los errores.
- Ampliar la funcionalidad de la aplicación permitiendo introducir datos desde el INE.
- Realizar predicciones de los datos y mostrarlas en gráficos.
- Permitir exportar los datos de las tablas a Excel.
- Mejorar la ayuda a los usuarios.
- Implementar la edición de usuarios.
- Mejorar la seguridad de la aplicación.
- Publicar la aplicación en su entorno de explotación final que es el servidor de la UBU.
- Crear una aplicación web funcional y operativa que funcione sin errores.

2.2. Objetivos técnicos

- Aprovechar las ventajas de la arquitectura de Laravel que consiste en modelos, vistas y controladores.

- Hacer uso de visual studio code para el desarrollo del código.
- Conseguir transferir los datos del INE en formato JSON a las tablas de la base de datos.
- Exportar correctamente los datos de las tablas a Excel.
- Encriptar las contraseñas usando un algoritmo seguro y eficiente.
- Mejorar la administración de usuarios.

2.3. Objetivos personales

- Aprender a programar en PHP.
- Aprender la metodología de Laravel y de las aplicaciones web.
- Entender el código escrito por otra persona y modificarlo según las necesidades para realizar procedimientos comunes en el desarrollo de software profesional, como el mantenimiento del software, refactorización de código, etc.
- Aprender a desplegar una aplicación en un servidor.

Conceptos teóricos

3.1. Encriptado de contraseñas

Para cifrar las contraseñas se estudiaron las posibles funciones de encriptación para este proyecto.

password_hash

Esta función de PHP crea un hash de contraseña usando un algoritmo de hash fuerte de único sentido.

Una función criptográfica hash es un algoritmo matemático que transforma cualquier bloque arbitrario de datos en una nueva serie de caracteres con una longitud fija. Independientemente de la longitud de los datos de entrada, el valor hash de salida tendrá siempre la misma longitud.^[5] Password_hash() se emplea para crear un hash con una cadena dada como primer argumento utilizando el algoritmo pasado como segundo argumento. Se puede elegir entre estos dos algoritmos para el encriptado:

PASSWORD_DEFAULT Usa el algoritmo bcrypt. Esta constante está diseñada para cambiar siempre que se añada un algoritmo nuevo y más fuerte a PHP. Por esto la longitud de la encriptación puede variar según cambia el algoritmo.

PASSWORD_BCRYPT Usa el algoritmo CRYPT_BLOWFISH para crear el hash. Produce un hash usando el identificador "\$2y\$". El resultado siempre será un string de 60 caracteres, o FALSE en caso de error.

Para comprobar que una contraseña introducida en el login coincide con el hash guardado en la base de datos se usa la función `password_verify()`.^[16]

```
<?php

// Almacenar el hash de la contraseña
$consulta = sprintf("INSERT INTO users(name,pwd) VALUES('%s','%s');",
    pg_escape_string($nombre_usuario),
    password_hash($contraseña, PASSWORD_DEFAULT));
$resultado = pg_query($conexión, $consulta);

// Consultar si el usuario envió la contraseña correcta
$consulta = sprintf("SELECT pwd FROM users WHERE name='%s';",
    pg_escape_string($nombre_usuario));
$fila = pg_fetch_assoc(pg_query($conexión, $consulta));

if ($fila && password_verify($contraseña, $fila['pwd'])) {
    echo 'Bienvenido, ' . htmlspecialchars($nombre_usuario) . '!';
} else {
    echo 'La autenticación ha fallado para ' . htmlspecialchars($nombre_usuario) . '.';
}

?>
```

Figura 3.1: Ejemplo de la utilización de `password_hash()`

crypt

Esta función tiene un funcionamiento muy similar que `password_hash()`. Se le pasan dos parámetros, el string que queremos encriptar y un string de salt para la base del hash.^[15] A continuación se define lo que es un string salt.

En criptografía, la sal o salt en inglés comprende bits aleatorios que se usan como argumento en una función que crea contraseñas. El otro argumento es el string que queremos codificar. El string salt también puede usarse como parte de una clave en un cifrado u otro algoritmo criptográfico. A veces se usa como salt el vector de inicialización, un valor generado previamente. Las claves con sal complican los ataques de diccionario que cifran cada una de las entradas del mismo: cada bit de sal duplica la cantidad de almacenamiento y computación requeridas.^[26]

Hash de Laravel

Laravel tiene una clase Hash que permite el encriptado de contraseñas usando los algoritmos Bcrypt y Argon2.

Bcrypt es una excelente opción para hacer hash de contraseñas porque su factor de trabajo es ajustable, lo que significa que el tiempo que lleva generar un hash puede incrementarse a medida que aumenta la potencia del hardware. Cuando se procesan contraseñas, la lentitud es buena. Cuanto más tiempo tarda un algoritmo en codificar una contraseña, más tardan los usuarios malintencionados en generar todas las cadenas posibles que pueden utilizarse en ataques de fuerza bruta contra aplicaciones.[12]

```
public function entrustPasswordHash()  
{  
    $this->password = Hash::make($this->password);  
    $this->save();  
}
```

Figura 3.2: Encriptación de contraseñas en mi aplicación

Después de informarme de todas las posibilidades que había para el cifrado, elegí la clase hash de Laravel usando el algoritmo bcrypt. Escogí esta porque me pareció la más actual y documentada y, además pertenece a Laravel por lo que pensé que sería mas adecuada para mi trabajo. En cuanto al algoritmo bcrypt me pareció bastante fiable y eficaz al utilizarse en otros sistema de elevada relevancia como OpenBSD y algunas distribuciones de Linux.

3.2. Datos abiertos

"Los datos abiertos son datos que pueden ser utilizados, reutilizados y redistribuidos libremente por cualquier persona, y que se encuentran sujetos, cuando más, al requerimiento de atribución y de compartirse de la misma manera en que aparecen."[8]

Los datos abiertos:

- Deben ser abiertos jurídicamente hablando, es decir, que deben estar en un sitio de acceso público y que sus condiciones de uso sean libres y sin restricciones.
- Tienen que ser publicados en formatos que puedan ser leídos por dispositivos electrónicos. Además deben permitir que su acceso sea universal y gratuito para todos los usuarios, sin el uso de contraseñas, restricciones o *firewalls*.

Existen dos conceptos importantes respecto a los datos abiertos:

Catálogo de datos Lista de conjuntos de datos disponibles en la plataforma de datos abiertos a la que estamos accediendo. Suele componerse de metadatos, información de la licencia de uso y datos. Es el elemento más importante en la plataforma de distribución de datos abiertos. Pueden ofrecerse en varios formatos como JSON, XML, CSV, etc.

Plataforma Servicio que da acceso a los usuarios al catálogo de datos. Además suele ofrecer un foro en línea para posibles preguntas, apoyo técnico, etc.

Algunas plataformas de datos abiertos poseen APIs (Application Programming Interfaces). Estas APIs nos permiten comunicarnos con el catálogo de datos de una forma más dinámica y precisa.

3.3. Algoritmos de predicción de datos

Uno de los requisitos que se propuso al iniciar el proyecto fue el incluir algún tipo de predicción o proyección de datos a futuro, empleando para ello algún algoritmo básico de Machine Learning. A continuación se describen algunos conceptos de la minería de datos. Los algoritmos de predicción de datos se distinguen en dos grandes grupos: Clasificadores y regresores.

Clasificadores

Estos algoritmos predicen datos categóricos basados en la información con la que han sido entrenados. No entraré en más detalle ya que no se usan en el proyecto.

Regresores

Los estimadores de tipo regresor se usan para predecir datos continuos. Son los siguientes:

Adaline Es un tipo de red neuronal artificial caracterizada por tener múltiples nodos los cuales aceptan muchos inputs para generar un solo output [20]. Las variables que utiliza son:

x Es el vector que contiene los datos de entrada.

w Vector que indica la fuerza de conexión entre los valores de entrada y la neurona.

n Número de inputs.

θ . La constante.

y Datos de salida.

Regression Tree Consiste en un árbol de decisión que se va bifurcando según los posibles resultados de una serie de decisiones relacionadas. Los parámetros son los siguientes:

maxHeight Altura máxima del árbol.

maxLeafSize Máximo número de decisiones que una hoja puede tener.

maxFeatures Máximo número de columnas a considerar al elegir una decisión.

minPurityIncrease Aumento mínimo de pureza necesario para que un nodo no se pode durante el crecimiento del árbol.

Extra Tree Regressor Es igual que el anterior pero se diferencian en que este modelo escoge la siguiente decisión de forma aleatoria en vez de buscar el mejor valor en una serie de datos. Son muy rápidos de construir pero sus resultados son muy variables. Los parámetros son los mismos.

KNN K Nearest Neighbors (KNN) es un algoritmo de fuerza bruta que localiza el k más cercano de los valores de entrada con los que se ha entrenado para hacer su predicción. Tiene tres parámetros:

k El número de vecinos a considerar al hacer la predicción.

weighted Booleano que indica si se debe considerar la distancia de los vecinos más cercanos al hacer la predicción.

kernel El kernel que se va a usar para computar la distancia entre los puntos de referencia.

K-d Neighbors Regressor Es una implementación rápida del algoritmo anterior. Usa un árbol binario con reconocimiento espacial para la búsqueda de vecinos más cercanos. K-d Neighbors Regressor funciona localizando la vecindad de una muestra mediante una búsqueda binaria y luego realiza una búsqueda de fuerza bruta solo en las muestras cercanas o dentro de la vecindad de la muestra desconocida. La principal

ventaja de K-d Neighbors sobre la fuerza bruta KNN es la velocidad de inferencia. Parámetros:

k El número de vecinos a considerar al hacer la predicción.

weighted Booleano que indica si se debe considerar la distancia de los vecinos más cercanos al hacer la predicción.

tree Árbol usado para ejecutar las búsquedas de los vecinos cercanos.

MLP Regressor Es una red neuronal multicapa con un output continuo adecuado para problemas de regresión. El regresor de perceptrón multicapa (MLP) es capaz de manejar problemas complejos de regresión no lineal formando representaciones de orden superior de las características de entrada utilizando capas ocultas intermedias definidas por el usuario.

El MLP también tiene instantáneas de red y monitoreo del progreso.

Técnicas y herramientas

4.1. Técnicas

Scrum

En Scrum un proyecto se ejecuta en ciclos temporales cortos y de duración fija (iteraciones o sprints).[18]

A cada sprint se le asigna unas tareas, generalmente con relación entre sí y en torno a un mismo tema.

Al finalizar la iteración se revisan las tareas en una reunión con el resto de miembros del equipo.

Integración continua

La integración continua es una práctica de ingeniería de software que consiste en hacer integraciones automáticas de un proyecto lo más a menudo posible para así poder detectar fallos cuanto antes. Entendemos por integración la compilación y ejecución de pruebas de todo un proyecto.[23]

Ventajas de usar la integración continua

- Capacidad de detección de problemas temprana, lo que facilita su solución.
- Disponibilidad en cualquier momento de distintas versiones.
- Ejecución inmediata de las pruebas unitarias.
- Monitorización continua de las métricas de calidad del código del proyecto.

4.2. Herramientas

Voy a clasificar las herramientas según las fases en las que han sido utilizadas.

Desarrollo

Laravel

El framework usado para el desarrollo de la aplicación ha sido Laravel. Laravel incluye Eloquent, un mapeador relacional de objetos (ORM) que simplifica la interacción con la base de datos. Cuando se usa Eloquent, cada tabla de la base de datos tiene un modelo correspondiente que se usa para interactuar con esa tabla. Además de recuperar registros de la tabla de la base de datos, los modelos Eloquent también le permiten insertar, actualizar y eliminar registros de la tabla.[13]

Laravel también se caracteriza por el uso de controladores para agrupar la funcionalidad de un determinado recurso.[1]

Posee una estructura de directorios predefinidos que ayuda a organizar nuestro proyecto. Los más importantes son *resources* que contiene las vistas de la aplicación, *routes* que tiene el archivo que guarda la definición de las rutas y *app* que guarda los modelos y los controladores entre otros archivos de configuración del proyecto.

Composer

Composer es un sistema de gestión de paquetes para programar en PHP el cual provee los formatos estándar necesarios para manejar dependencias y librerías de PHP. [21]

Es un gestor de dependencias para proyectos escritos en el lenguaje de programación PHP. Eso quiere decir que nos permite gestionar (declarar, descargar y mantener actualizados) los paquetes de software en los que se basa nuestro proyecto PHP.

Cuando se empieza un proyecto en PHP, ya de cierta complejidad, no vale solo con la librería de funciones nativa de PHP. Generalmente se usa alguna que otra librería de terceros desarrolladores, que permite evitar empezar todo desde cero. Ya sea un framework o algo más específico como un sistema para debug o envío de email, registro de usuarios, exportación de datos, etc., cualquier cosa que se pueda necesitar ya puede estar creada por otros desarrolladores.

Para usar composer se debe tener un archivo JSON en el que se deberá

escribir los paquetes que queramos instalar, el nombre del proyecto, la descripción, algunos comandos que se quiera que se ejecuten en el momento de la instalación o de la actualización (como por ejemplo generar la clave del proyecto u optimizarlo), etc. Puede haber muchas opciones de configuración posibles en este archivo. Este archivo debe llamarse `composer.json`.

Al usar el comando `composer install`, se crea automáticamente un archivo llamado `composer.lock` donde aparece una información más detallada de las dependencias instaladas además de ir a los repositorios de paquetes de software y descargar aquellas librerías mencionadas, copiándolas en la carpeta del proyecto.

Estas dependencias se instalan en la carpeta *vendor* de la aplicación.^[4]

Visual Studio Code

Es un entorno de desarrollo que permite crear sitios y aplicaciones web. Es compatible con múltiples lenguajes de programación, tales como C++, C#, Visual Basic .NET, F#, Java, Python, Ruby y PHP (Éste último es el que se ha usado para el proyecto).^[24] Esta herramienta detecta el lenguaje, da sugerencias de escritura, detecta errores y proporciona posibles soluciones. También se puede sincronizar con el proyecto en Github facilitando el seguimiento del proyecto.

Git

Se ha usado Git como herramienta para el control de las versiones del proyecto. Además se puede usar junto con Github desde Visual Studio Code al descargar su extensión.

Xampp

XAMPP es un paquete de software libre, que consiste principalmente en una pila de herramientas de desarrollo que incluye un sistema de gestión de bases de datos MySQL, el servidor web Apache y los intérpretes para lenguajes de script PHP y Perl. El nombre es en realidad un acrónimo: X (para cualquiera de los diferentes sistemas operativos), Apache, MariaDB/MySQL, PHP, Perl.^[25] Se ha usado esta aplicación para las pruebas locales de mi proyecto.

Despliegue

Docker

Docker sirve para automatizar el despliegue de aplicaciones dentro de contenedores software de forma que se pueda probar la aplicación en distintos equipos.

[22] Docker tiene como ventaja que aísla solo los recursos del sistema operativo que necesita y no una cantidad de recursos fija como hace una máquina virtual.

Para el despliegue de la aplicación se ha utilizado un contenedor especial para aplicaciones de Laravel llamado Laradock.

Laradock contiene multitud de imágenes conectadas entre ellas que se usan para el despliegue de la app. Sin embargo, solo se han usado Apache, PHP, PhpMyAdmin, MySQL y Selenium.

Heroku

Heroku es una plataforma en la nube que permite subir aplicaciones para probarlas. La principal ventaja que tiene Heroku es que su uso a un nivel básico es gratuito. Como desventaja diría que la aplicación va muy lenta en este servidor web.

Calidad del código

Codacy

Para evaluar la calidad del código se ha usado la herramienta Codacy. Permite realizar un análisis del proyecto y reportar los posibles problemas que haya. En el apartado *Issues* se pueden ver clasificados por varios criterios:

- Lenguaje de programación.
- Categorías: Estilo del código, seguridad y código sin usar.
- Nivel de peligrosidad: Distingue entre errores y warnings.
- Patrones no recomendados: Accesos estáticos, nombres de variables cortos, etc.

En el apartado *Files* aparecen los archivos del proyecto calificados con una nota en función del número de *Issues* que tenga, el grado de duplicación

y la complejidad. También permite ver la calidad de los cambios desde el apartado *Commits*.

Esta aplicación es muy fácil de usar y se puede conectar a un repositorio de Github de manera que al hacer algún cambio en éste el análisis se actualice.

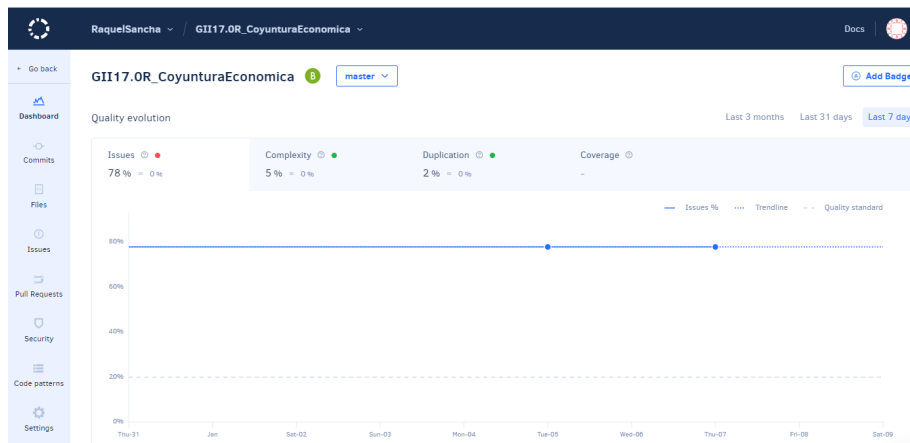


Figura 4.3: Opciones que ofrece codacy para el análisis de nuestro código

Documentación

Latex

L^AT_EX es una herramienta para la composición de texto con una serie de comandos que permiten formar documentos a gusto del usuario.

Está formada por un gran conjunto de macros de TeX, con la intención de facilitar el uso del lenguaje de composición tipográfica.[27]

Su documentación es muy amplia y detallada con multitud de ejemplos.

4.3. Librerías

Laravel Excel

Para exportar los datos de las tablas a Excel se ha usado la biblioteca Laravel Excel. Esta biblioteca está basada en PhpSpreadsheet que es un recurso escrito en PHP puro el cual proporciona un conjunto de clases que permiten leer y escribir en diferentes formatos de archivo de hoja de cálculo, como Excel y LibreOffice Calc.[7]

Chart.js

Esta librería en JavaScript es gratuita y de código abierto y se usa para la visualización de datos en forma de gráfico. Tipos de gráficos que permite crear:

- Gráfico de barras.
- Gráfico de líneas.
- Área.
- Burbuja.
- Radar.
- Polar.
- Dispersión.

Laravel DebugBar

Este proyecto, realizado por [un usuario de github](#), consiste en una DebugBar elaborada exclusivamente para proyectos en Laravel. Ayuda a identificar errores y a recopilar información de ejecución de la aplicación. Algunos de los datos que recopila son:

- Resultado de las consultas realizadas a la base de datos.
- Información de la ruta en la que estamos.
- Información de las vistas cargadas.
- Eventos.
- Información de la versión de Laravel y del entorno de desarrollo.
- Datos de los usuarios.
- Valores de los archivos de configuración.

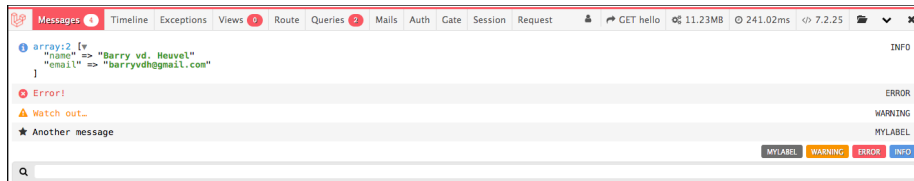


Figura 4.4: DebugBar para Laravel

Rubix ML

Esta biblioteca ha sido usada para la predicción de los datos. Incluye múltiples algoritmos de aprendizaje tanto para datos categóricos como continuos. En el caso de mi aplicación he usado los algoritmos para datos continuos.

Las fases por las que tienen que pasar los datos para realizar la predicción son:

- Extracción.
- Transformación.
- Carga.
- Entrenamiento.
- Predicción.

Para realizar la estimación, hay que precargar los datos que se tengan, adaptarlos y pasárselos al estimador para entrenarlo. Los posibles algoritmos que se usan para la estimación han sido explicados en el apartado de conceptos teóricos.

Aspectos relevantes del desarrollo del proyecto

En este apartado se van a recoger los aspectos más importantes del desarrollo del proyecto. Desde cuestiones del desarrollo del proyecto, hasta los numerosos problemas a los que hubo que enfrentarse y cómo se solucionaron.

5.1. Inicio del proyecto: Aprendizaje

En esta fase, probablemente la más costosa a nivel personal, se detallarán los conceptos de los que hizo falta un aprendizaje para empezar el proyecto.

Aprendizaje de Laravel

Laravel tiene un sistema de carpetas y archivos de configuración muy extenso que al principio cuesta entender pero por suerte existen multitud de tutoriales en forma de vídeos o de artículos que explican como iniciarse en Laravel.

Si no se tiene ningún conocimiento previo sobre este framework no recomendaría mirar la documentación de la página oficial ya que, aunque es muy completa, no considero que ofrezca una visión general sobre el funcionamiento del mismo sino información muy específica que no sirve al usuario si no conoce primero la dinámica general de la herramienta.

Tutoriales que seguí:

- *Laravel PHP Framework Tutorial*(freeCodeCamp)[9]

- *Curso Laravel*(pildorasinformaticas)[17]
- *Ejemplo de un proyecto en laravel*(Julio Yáñez Novo)[14]

Aprendizaje de PHP

Este lenguaje no se enseña en la carrera y tampoco había trabajado con él antes así que tuve que aprenderlo desde cero. No me resultó tan difícil como aprender a entender proyectos con Laravel así que esta parte me supuso menos trabajo.

Las herramientas que usé para familiarizarme con este lenguaje fueron:

- *Tutorial de PHP para principiantes*(ionos)[11]
- *Curso php*(tutorialesya)[19]

5.2. Gestión de los errores

En la aplicación original existían varios errores que hubo que solucionar.

Exportación de datos a excel . Existía un botón para exportar los datos de la tabla elegida a Excel pero no funcionaba. Para conseguir que el botón cumpliera la funcionalidad prometida he usado la biblioteca Laravel Excel (explicada anteriormente en el apartado "Técnicas y herramientas")

Administración de usuarios Esta sección daba un error cuando se intentaba acceder a ella.

Para entender esta sección hace falta conocer el sistema de usuarios que usa la aplicación:

Súper administrador Las personas con este rol pueden controlar los roles de los demás usuarios y modificarlos. Además se les permite aceptar o declinar las peticiones de registro en la aplicación a nuevos usuarios. También tienen acceso al resto las funcionalidades de la aplicación.

Administrador Este tipo de usuarios tienen acceso a todas las partes de la aplicación menos a la de la administración de usuarios, es decir, no pueden aceptar nuevos usuarios ni cambiar roles.

Invitado Este tipo de usuarios no está registrado en la aplicación. Sólo tiene acceso a las tablas para su visualización. Para poder acceder a las demás funcionalidades deberá realizar una petición de registro.

Usuarios del sistema	
Email	Acciones
SuperAdmin@superadmin.es	<button>Editar</button> <button>Borrar</button>
Admin@admin.es	<button>Editar</button> <button>Borrar</button>
Solicitudes de creación de cuenta	
Email	Acciones
raquelsanchar12@gmail.com	<button>Aceptar</button> <button>Declinar</button>

Figura 5.5: Sección Administración de usuarios

Desde esta pantalla, la cual solo se puede acceder si tienes privilegios de súper administrador, se permite aceptar las solicitudes pendientes de creación de una cuenta así como editar los roles y borrar otros perfiles.

Editar usuario "SuperAdmin"	
Nombre	<input type="text"/>
Email	<input type="text"/>
Contraseña	<input type="password"/>
<button>Modificar</button>	

Figura 5.6: Editar datos de un usuario

También se creó esta pantalla para permitir al usuario editar sus datos.

5.3. Ampliación de funcionalidades

Después de subsanar los errores que existían en la aplicación, se decidió implementar más funcionalidades.

Introducción de datos desde una base de datos externa

Para realizar el informe de la coyuntura económica de Burgos se usan múltiples fuentes así que se pensó en una forma de sacar la información de una manera más sencilla para el usuario.

Las bases de datos abiertos más utilizadas son: el Instituto Nacional de Estadística, Eurostat y los datos abiertos de la Junta de Castilla y León. En esta sección se hablará de cada una de ellas y se explicará su manera de extraer datos concluyendo con la justificación de la plataforma elegida.

Instituto nacional de estadística

Una fuente de la que suelen sacar los datos para el análisis de la coyuntura económica es el Instituto Nacional de Estadística (INE). Por ello, para facilitar la extracción de los datos, se pensó en la posibilidad de permitir seleccionar la información de esta página y que la aplicación la insertara automáticamente en la base de datos.

El catálogo de datos de esta plataforma proviene de dos fuentes: la base de datos de difusión Tempus3 y el repositorio de ficheros PC-Axis.

La información se distribuye en tablas en las que se permite filtrar los datos según sus características.

INE
Instituto Nacional de Estadística

INEbase / Encuesta sobre la estructura de las explotaciones agrícolas

TOTAL NACIONAL. Encuesta sobre la estructura de las explotaciones agrícolas año 2016
Clasificación según superficie agrícola utilizada

Número, superficie total y superficie agrícola utilizada (SAU) de las explotaciones
Unidades: definidas en los valores de las variables

Seleccione valores a consultar

Tamaño de las explotaciones según SAU (Ha.)	Características básicas
Explotaciones con tierras	Explotaciones. Número
Explotaciones sin SAU	Explotaciones. %
Explotaciones con SAU	Superficie total. Ha.
< 1	Superficie total. %
1 a < 2	SAU. Ha.
2 a < 5	SAU. %
5 a < 10	
Seleccionados: 12	Total: 12

Tamaño de las explotaciones según SAU (Ha.)	Características básicas
Explotaciones con tierras	Explotaciones. Número
Explotaciones sin SAU	Explotaciones. %
Explotaciones con SAU	Superficie total. Ha.
< 1	Superficie total. %
1 a < 2	SAU. Ha.
2 a < 5	SAU. %
5 a < 10	
Seleccionados: 6	Total: 6

Figura 5.7: Ejemplo de una tabla del INE

Una vez se seleccionen los valores que se quieran consultar, los datos aparecen en forma de tablas o distribuidos en un gráfico.

Tabla	Gráfico					
	Explotaciones. Número	Explotaciones. %	Superficie total. Ha.	Superficie total. %	SAU. Ha.	SAU. %
Explotaciones con tierras	933.059 ¹	100,00 ¹	30.012.082 ¹	100,00 ¹	23.229.753 ¹	100,00 ¹
Explotaciones sin SAU	6.130 ¹	0,66 ¹	46.903 ¹	0,16 ¹	0 ¹	0,00 ¹
Explotaciones con SAU	926.929 ¹	99,34 ¹	29.965.179 ¹	99,84 ¹	23.229.753 ¹	100,00 ¹
< 1	72.009 ¹	7,72 ¹	70.476 ¹	0,23 ¹	38.147 ¹	0,16 ¹
1 a < 2	166.915 ¹	17,89 ¹	380.188 ¹	1,27 ¹	227.875 ¹	0,98 ¹
2 a < 5	230.338 ¹	24,69 ¹	1.008.581 ¹	3,36 ¹	734.801 ¹	3,16 ¹
5 a < 10	140.561 ¹	15,06 ¹	1.225.957 ¹	4,08 ¹	999.039 ¹	4,30 ¹
10 a < 20	112.284 ¹	12,03 ¹	1.848.460 ¹	6,16 ¹	1.562.656 ¹	6,73 ¹
20 a < 30	50.194 ¹	5,38 ¹	1.452.387 ¹	4,84 ¹	1.227.462 ¹	5,28 ¹
30 a < 50	52.201 ¹	5,59 ¹	2.543.820 ¹	8,48 ¹	2.004.089 ¹	8,63 ¹
50 a < 100	50.485 ¹	5,41 ¹	4.358.896 ¹	14,52 ¹	3.540.418 ¹	15,24 ¹
>=100	51.942 ¹	5,57 ¹	17.076.413 ¹	56,90 ¹	12.895.266 ¹	55,51 ¹

Figura 5.8: Datos de una tabla del INE

Además de esta manera de visualizar los datos, el Instituto Nacional de Estadística proporciona un servicio que permite descargar la información en formato JSON. Para ello debemos crear una petición en forma de url.

```
{
  "COD": "IPC206449", "Nombre": "Total Nacional. Índice general. Variación mensual. ", "FK_Unidad": 135, "FK_Escala": 1, "Data": [
    {
      "Fecha": 1606777200000, "FK_TipoDato": 3, "FK_Periodo": 12, "Anyo": 2020, "Valor": 0.2, "Secreto": false
    }
  ]
}
```

Figura 5.9: Salida de los datos en formato JSON

La url para la petición de los datos tiene el formato que aparece en la imagen.

`https://servicios.ine.es/wstempus/js/{idioma}/{función}/{input}[?parámetros]`

Figura 5.10: Estructura de la url

Los campos que aparecen entre llaves, { }, son obligatorios.^[10]
 Los campos que aparecen entre corchetes, [], son opcionales y cambian en relación a la función considerada.
 Descripción de cada uno de ellos

idioma ES para español e EN para inglés. Por defecto está puesto el español.

función Funciones implementadas en el sistema para poder realizar diferentes tipos de consulta en función del tipo de fuente, Tempus3 o PC-Axis, y del elemento que se quiere obtener.

inputs Identificadores de los elementos de entrada de las funciones. Estos inputs varían en base al repositorio del que se extraen los datos.

parámetros Los parámetros en la URL se establecen a partir del símbolo ?.

Cuando haya más de un parámetro, el símbolo & se utiliza como separador.

No todas las funciones admiten todos los parámetros posibles. Por ello haremos una clasificación para explicarlos mejor:

1. Parámetros comunes a todas las funciones

page Si hay más de 500 elementos, la consulta se divide en páginas. Esta opción nos permite seleccionar la página que queremos visualizar.

download Para descargar el fichero JSON.

det Este parámetro da más detalles de la información mostrada.

tip Cambia la forma de mostrar la información.

2. Parámetros para la petición de datos

date Filtra los datos por fecha; fecha concreta, lista o rango de fechas.

nult Devuelve los últimos n datos. Ejemplo: nult=4 devuelve los 4 últimos datos.

3. Parámetros para la obtención de datos y metadatos en base al ámbito geográfico

geo Con geo = 1 para provincias, municipios u otras desagregaciones y geo = 0 para datos nacionales.

Eurostat

Eurostat es una base de datos que recoge información a nivel europeo. Tiene un servicio de datos en JSON y en Unicode mediante el cual permite descargar los archivos disponibles en estos dos formatos a través de una petición REST.

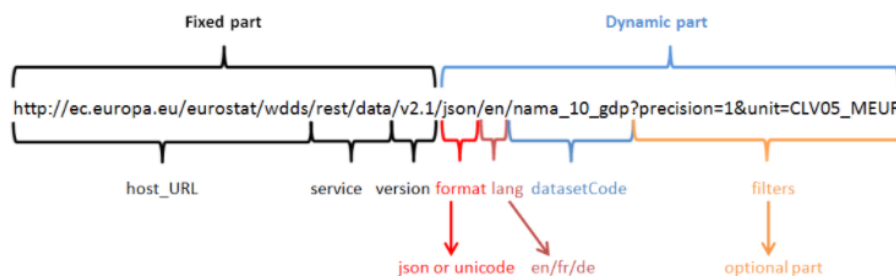


Figura 5.11: Estructura de una petición REST de Eurostat

[6] Como se puede ver la estructura es muy parecida a la petición de datos del INE, con una parte fija, otra parte que indica el formato, una que indica el lenguaje y finalmente el código que identifica los datos. También permite filtrar la información.

El inconveniente de este servicio es la forma de conseguir el identificador de los datos (*datasetCode* en la imagen). Los códigos de los conjuntos de datos se pueden encontrar ordenados por temas en la página de Eurostat

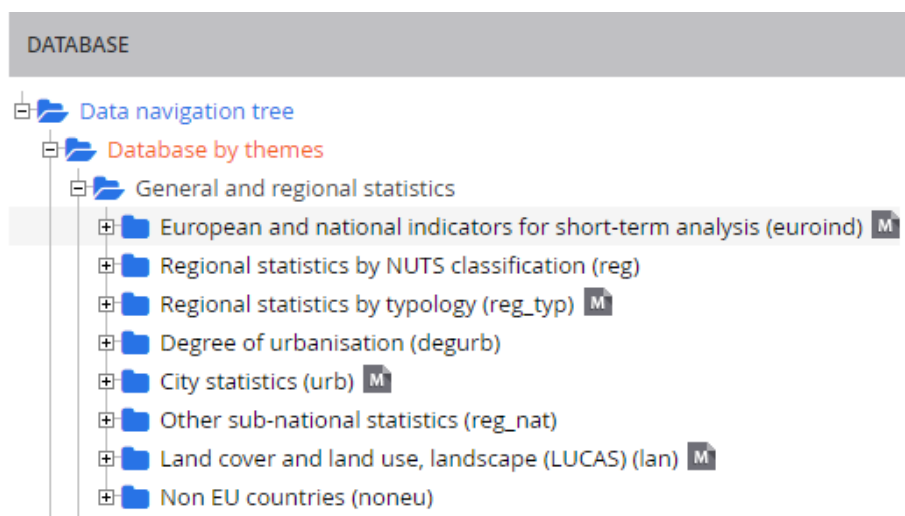


Figura 5.12: Conjuntos de datos del Eurostat

Datos abiertos de la Junta de Castilla y León

El portal de Datos Abiertos se enmarca en el proyecto de Gobierno Abierto de la Junta de Castilla y León por el que se pone en marcha el Modelo de Gobierno Abierto de la Junta de Castilla y León junto con la información de transparencia, el espacio de participación ciudadana y la presencia en redes sociales entre otras actuaciones.

Este portal tiene como objetivos aumentar la transparencia, proporcionando mayor información sobre la actividad de la Junta de Castilla y León y conseguir la participación y colaboración de los ciudadanos y empresas, a través de la interlocución con los mismos, de manera que el intercambio de conocimiento y experiencias permita el avance conjunto de la iniciativa pública y privada.^[3]

Búsqueda de conjuntos de datos

q	<input type="text"/>	Consulta de texto completo		
rows	<input type="text" value="10"/>	Número de filas del resultado (valor predeterminado: 10)		
start	<input type="text"/>	Índice del primer resultado para devolver (usar para paginación)		
sort	<input type="text"/>	Expresión de ordenación (campo o -campo)		
facet	<input type="text"/> +	Nombre de facetas para activar en los resultados		
refine	<table><tr><td><input type="text" value="Nombre de faceta"/></td><td><input type="text" value="Valor"/></td></tr></table>	<input type="text" value="Nombre de faceta"/>	<input type="text" value="Valor"/>	Refinamientos para aplicar
<input type="text" value="Nombre de faceta"/>	<input type="text" value="Valor"/>			
exclude	<table><tr><td><input type="text" value="Nombre"/></td><td><input type="text" value="Valor"/></td></tr></table>	<input type="text" value="Nombre"/>	<input type="text" value="Valor"/>	Exclusiones para aplicar
<input type="text" value="Nombre"/>	<input type="text" value="Valor"/>			

Enviar

Figura 5.13: Búsqueda de datos desde la API

Esta plataforma también permite descargar la información en formato JSON, CSV o Excel. Su funcionamiento es muy simple pero no permite automatizar la extracción de los datos como las bases de datos anteriores.



Figura 5.14: Algunos conjuntos de datos de la aplicación

Conclusión

Para elegir qué plataforma usar para la extracción de datos se hizo una evaluación de sus características.

	INE	Eurostat	JCYL
Posee un sistema de petición de datos por url	X	X	
La petición de los datos puede automatizarse	X		
Posee un gran catálogo de datos	X	X	X
Facilidad de uso	X		X
Facilidad de extracción de datos	X		

Tabla 5.1: Comparativa de las plataformas de datos abiertos

Habiendo sopesado las tres opciones, se eligió la plataforma del Instituto Nacional de Estadística por ser la más completa y sencilla de automatizar.

Predicción de datos

Otra funcionalidad interesante que se decidió añadir fue la posibilidad de predecir los datos de una variable.

Para ello, se utilizaron una serie de algoritmos de predicción de datos. Para elegir cuál usar para la predicción se hizo un estudio de todos los posibles de la biblioteca de machine learning usada.

Para seleccionar el algoritmo, hubo que tener en cuenta la dimensión de los datos de la aplicación, es decir, que el volumen de información para la predicción no va a ser muy grande así que el algoritmo usado tiene que ser muy flexible y con unos resultados óptimos para conjuntos de datos pequeños. Para la prueba de los métodos, se seleccionó un conjunto de datos reales y se predijo un año que ya se conocía para poner las predicciones en comparación con los valores reales. Se pueden ver los resultados en forma de gráfico: en azul las predicciones producidas por los estimadores y en rojo los valores reales.

Prueba con el algoritmo Adaline

Como se puede observar, este estimador no produce buenas predicciones para la aplicación ya que necesita más volumen de datos.

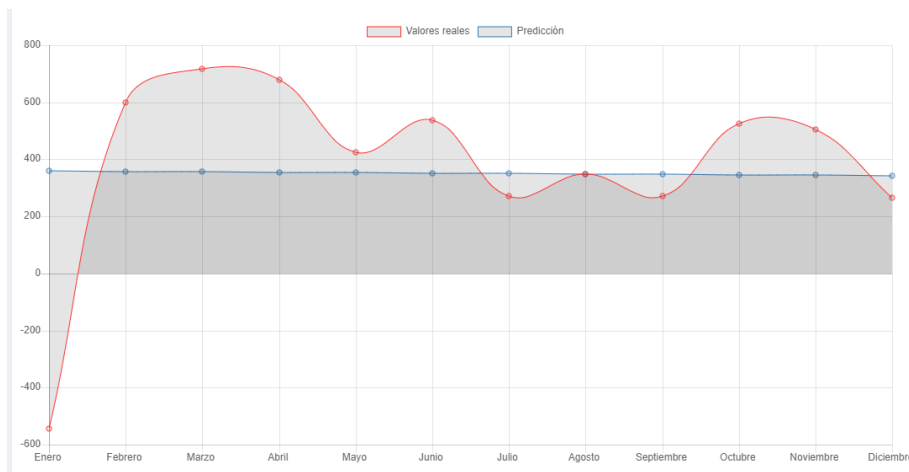


Figura 5.15: Gráfico de la comparación de la predicción con los valores reales

Prueba con el algoritmo MLP Regressor

Con este estimador ocurre lo mismo que con el anterior, al ser los dos redes neuronales necesitan mayor cantidad de datos para sus predicciones.

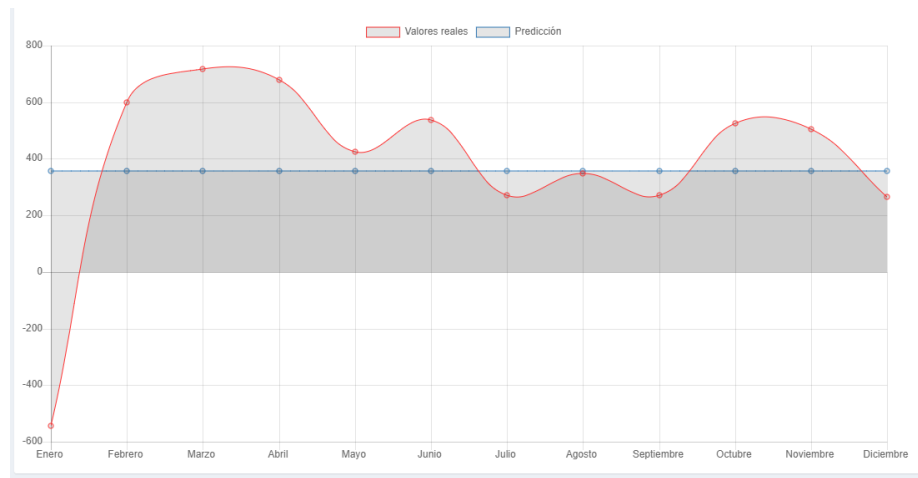


Figura 5.16: Gráfico de la comparación de la predicción con los valores reales

Prueba con el algoritmo K-d Neighbours Regressor

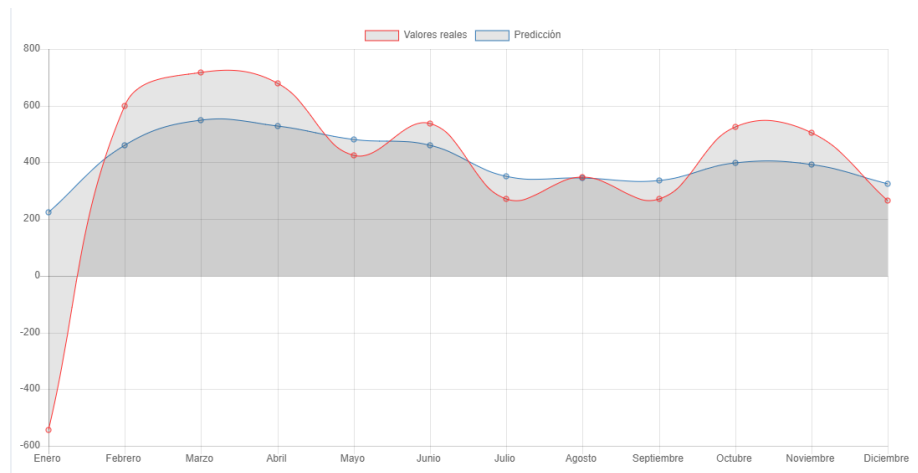


Figura 5.17: Gráfico de la comparación de la predicción con los valores reales

Prueba con el algoritmo KNN Regressor

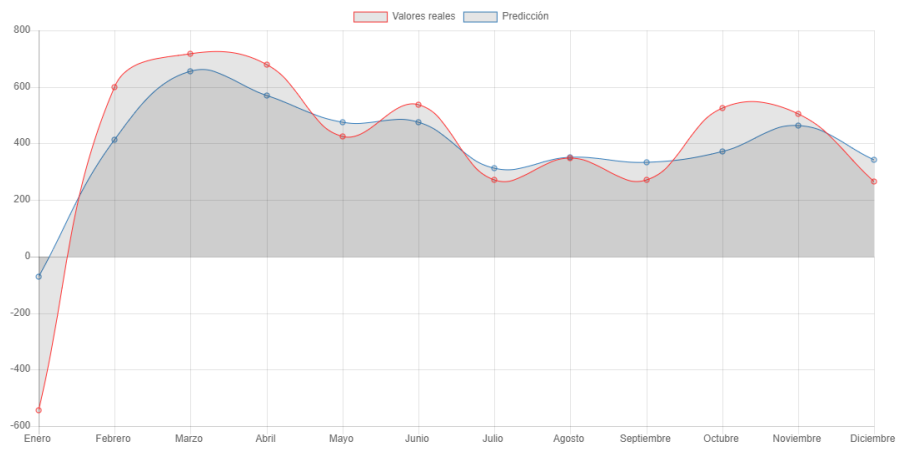


Figura 5.18: Gráfico de la comparación de la predicción con los valores reales

Prueba con el algoritmo Regression Tree

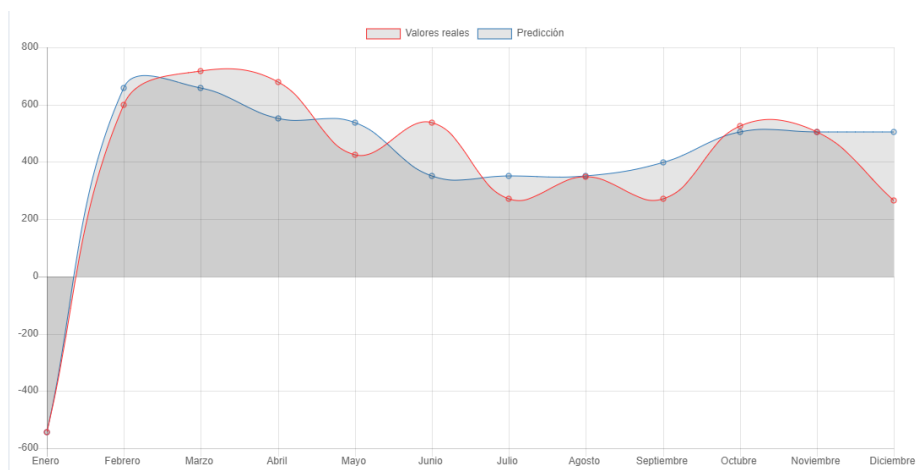


Figura 5.19: Gráfico de la comparación de la predicción con los valores reales

Prueba con el algoritmo Extra Regression Tree

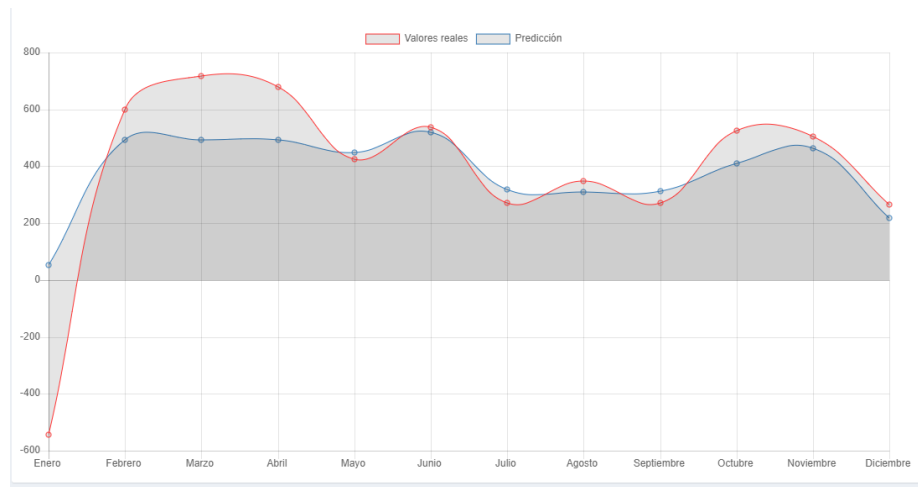


Figura 5.20: Gráfico de la comparación de la predicción con los valores reales

Conclusiones

Para elegir qué algoritmo usar para la predicción de datos se hizo una evaluación de los resultados de sus pruebas.

	Adaline	MLP	K-dN	KNN	RT	ERT
Realiza predicciones óptimas con pocos datos			X	X	X	X
Es rápido			X	X		X
Es intuitivo y sencillo de utilizar					X	X
Sus predicciones son muy variables						X
Tiene funciones que calculan desviaciones	X	X				X

Tabla 5.2: Comparativa de los algoritmos de predicción de datos

Habiendo visto los resultados de las predicciones y evaluándolos, se llegó a la conclusión de que el mejor algoritmo para la estimación de datos para esta aplicación es el Extra Tree Regressor por su rapidez y facilidad de uso así como la validez de sus predicciones ante conjuntos de datos pequeños.

5.4. Problemas

A lo largo del desarrollo del proyecto se encontraron problemas que condicionaron el trabajo. En esta sección se expondrán y se explicarán cómo se solucionaron.

Problemas de compatibilidad

Este tipo de problemas tuvieron que ver con Composer. Como ya se ha explicado, Composer es un gestor de dependencias del proyecto. El problema de usar este gestor es que al instalar nuevas librerías era muy frecuente que aparecieran errores de compatibilidad.

```
# composer require maatwebsite/excel
Using version ^3.1 for maatwebsite/excel
./composer.json has been updated
Running composer update maatwebsite/excel
Loading composer repositories with package information
Updating dependencies
Your requirements could not be resolved to an installable set of packages.

Problem 1
- maatwebsite/excel 3.1.0 requires illuminate/support ^5.5 -> found illuminate/support[v5.5.0, ..., v5.8.36] but it conflicts with another require.
- maatwebsite/excel[3.1.1, ..., 3.1.6] require illuminate/support 5.5.* || 5.6.* || 5.7.* -> found illuminate/support[v5.5.0, ..., v5.7.28] but it conflicts with another require.
- maatwebsite/excel[3.1.7, ..., 3.1.15] require illuminate/support 5.5.*[5.6.*[5.7.*[5.8.* -> found illuminate/support[v5.5.0, ..., v5.8.36] but it conflicts with another require.
- maatwebsite/excel[3.1.16, ..., 3.1.18] require illuminate/support 5.5.*[5.6.*[5.7.*[5.8.*]^6.0 -> found illuminate/support[v5.5.0, ..., v5.8.36, v6.0.0, ..., v6.20.12] but it conflicts with another require.
- maatwebsite/excel[3.1.19, ..., 3.1.21] require illuminate/support 5.5.*[5.6.*[5.7.*[5.8.*]^6.0|^7.0 -> found illuminate/support[v5.5.0, ..., v5.8.36, v6.0.0, ..., v6.20.12, v7.0.0, ..., v7.30.3] but it conflicts with another require.
- maatwebsite/excel[3.1.22, ..., 3.1.25] require illuminate/support 5.5.*[5.6.*[5.7.*[5.8.*]^6.0|^7.0|^8.0 -> found illuminate/support[v5.5.0, ..., v5.8.36, v6.0.0, ..., v6.20.12, v7.0.0, ..., v7.30.3, v8.0.0, ..., v8.22.1] but it conflicts with another require.
- maatwebsite/excel 3.1.26 requires illuminate/support 5.8.*|^6.0|^7.0|^8.0 -> found illuminate/support[v5.8.0, ..., v5.8.36, v6.0.0, ..., v6.20.12, v7.0.0, ..., v7.30.3, v8.0.0, ..., v8.22.1] but it conflicts with another require.
- Root composer.json requires maatwebsite/excel ^3.1 -> satisfiable by maatwebsite/excel[3.1.0, ..., 3.1.26].

Installation failed, reverting ./composer.json and ./composer.lock to their original content.
```

Figura 5.21: Captura de un problema típico al intentar instalar un paquete nuevo

Para la resolución de estos problemas se han tenido que actualizar, borrar e incluso instalar versiones viejas de las dependencias problemáticas.

Problemas en el despliegue del proyecto

Para la producción del trabajo se han usado muchos servidores locales y on-line como Xampp, Wampp, Docker, etc y todos ellos han supuesto un

reto en su correcta configuración.

Xampp

Xampp ha sido la herramienta más usada para el desarrollo del proyecto. La mayoría de los problemas se tuvieron a la hora de configurar el host virtual.

```
NameVirtualHost *
<VirtualHost *>
    DocumentRoot "C:\xampp\htdocs"
    ServerName localhost
</VirtualHost>
<VirtualHost *>
    DocumentRoot "C:\miproyecto\htdocs"
    ServerName proyecto.test
    <Directory "C:\miproyecto\htdocs">
        Require all granted
    </Directory>
</VirtualHost>
```

Figura 5.22: Configuración de los hosts virtuales

Al principio, la ruta no era resuelta correctamente por el navegador. Para la solución a este problema hizo falta cambiar el archivo de configuración del servidor de Xampp, y habilitar los hosts virtuales.

Además hubo que borrar la caché del navegador ya que cargaba la anterior configuración.

Docker

Para usar esta herramienta se usó el contenedor *laradock* que tiene multitud de imágenes para proyectos de Laravel. El principal problema fue el servidor que utilizaba ya que en un principio se usó *nginx* y no funcionaba correctamente. Después se cambió a *apache*.

```
### Paths #####  
  
# Point to the path of your applications code on your host  
APP_CODE_PATH_HOST=../  
  
# Point to where the `APP_CODE_PATH_HOST` should be in the container  
APP_CODE_PATH_CONTAINER=/var/www/
```

Figura 5.23: Archivo de configuración .env de laradock

El direccionamiento del proyecto se realiza desde el archivo *.env*. Laradock usa un direccionamiento virtual de los archivos así que esta parte también fue problemática.

Heroku

Para la presentación de la aplicación se decidió utilizar en Heroku en un principio, que es un hosting gratuito donde se puede alojar tu aplicación web. Hubo varios problemas al instalarlo ya que el proyecto tiene dos partes, la parte del código y la base de datos. Heroku ofrece una base de datos propia pero es de pago, además está en PostgreSQL y la del proyecto está en MySQL.

Para solucionar esto se consiguió conectar la aplicación a una base de datos externa totalmente gratuita llamada db4free.net.

Cuando se consiguió que todo funcionara correctamente, se notó la lentitud de la aplicación, pero al ser un hosting gratuito es normal ya que no nos ofrecen mucha memoria.

La aplicación funciona, pero al tener tantos servicios con peticiones agotamos la memoria RAM que nos ofrece Heroku y el rendimiento de la aplicación no es el esperado.

Para poder tener un buen rendimiento, sería necesario usar la versión de pago de Heroku, mucho más potente.

Trabajos relacionados

6.1. Herramienta web dinámica para la captación, tratamiento y presentación de datos relacionados con la coyuntura económica de Burgos

Esta herramienta es en la que se ha basado este trabajo, ya que es una ampliación y mejora de ésta.

Esta aplicación web permite la introducción de datos en tablas separadas por variables.

Nombre	Descripción	Tipo	Fuente	
Indicadores de Demanda	Los municipios con población inferior a los 1.000 habitantes, presentan evoluciones y tendencias muy dispares y en algunos casos muy variables, posiblemente por circunstancias muy particulares. En general, se acentúa la regresión poblacional, en una situación que se puede calificar de muy difícil retorno por el elevado número de núcleos de población y la escasa dimensión de los mismos, lo que dificulta la implantación y dotación de unos servicios mínimos, que garanticen el bienestar de la población con unos baremos aceptables en nuestra sociedad actual. De nuevo encontramos una evolución anómala en el conjunto de los municipios de entre 250 y 500 habitantes, donde el conjunto ha descendido muy levemente y hay núcleos de población en esta franja poblacional con incrementos significativos de población (Villamanzo casi un 12%, Mecerreyes supera el 10%), sin embargo la mayoría mantiene significativas pérdidas de población.	Tasa de variación	IPÍ Base 2007	Modificar Borrar
Indicadores de Oferta	En el caso de las economías de mercados emergentes y en desarrollo, se ha continuado con un elevado ritmo de crecimiento. La reducción de la demanda externa en muchas economías emergentes, como las de Asia, se ha compensado con la mejora de la demanda interna. El desafío actual se centra en afrontar los nuevos riesgos de recalentamiento derivados de las limitaciones de capacidad productiva y de la gran afluencia de flujos de capital. Las economías en desarrollo –en especial, las economías de África subsahariana– también han reanudado un elevado crecimiento en 2010, aunque amenazado por la fuerte subida del precio de los alimentos y de la energía y la incertidumbre suscitada por las tensiones políticas y sociales en los países de Oriente Medio y Norte de África.	Tasa de variación	Elaboración Propia	Modificar Borrar
Perspectivas de la economía: Proyecciones	La evolución negativa en Burgos se ha confirmado en el año 2010 con los datos del padrón, que recoge una disminución del -0,2%, aunque el avance del padrón a 1 de enero de 2011 refleja una pequeña recuperación del 0,16%. Sin embargo, los datos de población estimada, también elaborados por el INE, indican una pérdida de población mucho más preocupante a partir del otoño de 2009, que en tanto anual superan el -0,6%, situación que se mantiene e incluso se agrava ligeramente durante el año 2010 y 2011, lo que implica una pérdida superior de más de 5.000 habitantes desde octubre de 2009. Además, esta disminución, en porcentaje, es muy superior a la que experimenta el conjunto de la Comunidad Autónoma,	Variación de población	INE y Elaboración propia	Modificar Borrar

Figura 6.24: Variables de la aplicación

Cada tabla guarda los datos asociados a la variable que representa. Puede haber distintos ámbitos o categorías pertenecientes a la variable.

	2011												2012											
	Ene	Feb	Mar	Abr	May	Jun	Jul	Ago	Sep	Oct	Nov	Dic	Ene	Feb	Mar	Abr	May	Jun	Jul	Ago	Sep	Oct	Nov	Dic
España																								
Consumo																								
IPI de bienes de consumo	7.9	7.9	7.9	-8.2	-8.2	-8.2	-3.5	-3.5	-3.5	-3.6	-3.6	-3.6	-4.2	-4.2	-4.2	-7.1	-7.1	-7.1	-6.9	-6.9	-6.9	-0.7	-0.7	-0.7
Inversión																								
Importación de bienes de equipo	-7.6	-7.6	-7.6	-31.1	-31.1	-31.1	-26.7	-26.7	-26.7	-15.5	-15.5	-15.5	-18.5	-18.5	-18.5	-26.6	-26.6	-26.6	-27.9	-27.9	-27.9	-15	-15	-15
Matriculación de vehículos industriales	10.8	10.8	10.8	2	2	2	-19.3	-19.3	-19.3	-22.5	-22.5	-22.5	-27.5	-27.5	-27.5	-14.3	-14.3	-14.3	-22.1	-22.1	-22.1	-4.9	-4.9	-4.9
Sector Exterior																								
Importaciones	18.1	18.1	18.1	9.7	9.7	9.7	-2.9	-2.9	-2.9	1.4	1.4	1.4	-4.8	-4.8	-4.8	1.3	1.3	1.3	5	5	5	1	1	1

Fuente: "IPI Base 2007"

Figura 6.25: Tabla de una de las variables de la aplicación

6.2. JSON-Stat

Esta plataforma recoge herramientas para el procesado de datos en formato JSON.^[2] Algunas de ellas son:

JSON-stat Viewer Permite ver el contenido de una consulta JSON.

JSON-stat Table Browser Permite ver el contenido de una consulta JSON en forma de tablas.

JSON-stat Input Table Browser Permite pegar un texto en formato JSON y verlo en forma de tabla.

Exportador CSV Convierte a formato CSV un JSON.

Importador CSV Convierte a formato JSON un CSV.

Conclusiones y Líneas de trabajo futuras

7.1. Conclusiones técnicas

Considero que casi todos los objetivos fijados inicialmente se han cumplido satisfactoriamente. Los errores de la aplicación se han solucionado y además se han conseguido añadir las nuevas funcionalidades propuestas:

- Extracción de datos desde el INE.
- Predicción de datos y representación gráfica.
- Edición de usuarios.
- Extracción de las tablas a formato Excel.

Además se ha cumplido una de las principales exigencias por la parte del cliente que fue la subida de la aplicación al entorno de explotación de pruebas en un servicio web a cargo del Servicio de Informática de la UBU.

7.2. Conclusiones personales

Considero que realizar este trabajo me ha enseñado importantes conocimientos en el desarrollo y despliegue de aplicaciones web. También me ha enseñado a cómo afrontar posibles futuros proyectos en solitario llevando una planificación correcta y organizada.

Además he ganado experiencia en la resolución de los problemas que puedan surgir en el transcurso de un futuro proyecto.

7.3. Líneas de trabajo futuras

Algunos conceptos sobre posibles cambios o mejoras respecto al proyecto pueden ser:

- Mejorar la extracción de datos desde el INE ya que hay algunos conjuntos de datos que no es capaz de reconocer.
- Migrar la aplicación a una app móvil.
- Importar datos desde otras plataformas mencionadas anteriormente como Eurostat o los datos abiertos de la Junta de Castilla y León.
- Sería interesante poder exportar todos los datos de las tablas y de los gráficos a otro tipo de ficheros como por ejemplo “.pdf”, “.doc”, etc.

Bibliografía

- [1] ajgallego. Controladores de laravel. URL https://ajgallego.gitbook.io/laravel-5/capitulo_2/capitulo_2_controladores.
- [2] XAVIER BADOSA. Json-stat. URL <https://json-stat.org/tools/>.
- [3] datosabiertos jcyl. Datos abiertos de la junta de castilla y león: El proyecto. URL <https://datosabiertos.jcyl.es/web/es/iniciativa-datos-abiertos/proyecto-junta-castilla-leon.html>.
- [4] desarrolloweb. Explicación del uso de composer. URL <https://desarrolloweb.com/articulos/composer-gestor-dependencias-para-php.html>.
- [5] Brian Donohue. Definición de hash. URL <https://latam.kaspersky.com/blog/que-es-un-hash-y-como-funciona/2806/#:~:text=10%20Abr%202014-,Una%20funci%C3%B3n%20criptogr%C3%A1fica%20hash%2D%20usualmente%20conocida%20como%20%E2%80%9Chash%E2%80%9D%2D,tendr%C3%A1%20siempre%20la%20misma%20longitud>.
- [6] Eurostat. Eurostat. URL <https://ec.europa.eu/eurostat/data/database>.
- [7] Laravel excel. Laravel excel. URL <https://laravel-excel.com/>.
- [8] Open Knowledge Foundation. Datos abiertos. URL <http://opendatahandbook.org/guide/es/what-is-open-data/>.
- [9] freeCodeCamp.org. Tutorial laravel básico. URL <https://www.youtube.com/watch?v=ImtZ5yENzgE>.

- [10] INE. Api json - definición de urls. URL <https://www.ine.es/dyngs/DataLab/manual.html?cid=47>.
- [11] ionos. Tutorial php para principiantes. URL <https://www.ionos.es/digitalguide/paginas-web/creacion-de-paginas-web/tutorial-de-php-fundamentos-basicos-para-principiantes/>.
- [12] Laravel. Hash de laravel, . URL <https://laravel.com/docs/8.x/hashing>.
- [13] Laravel. Modelos de laravel, . URL <https://laravel.com/docs/8.x/eloquent>.
- [14] Julio Yáñez Novo. Primer proyecto en laravel. URL <https://codigoxules.org/primeros-pasos-para-la-creacion-de-una-aplicacion-web-con-laravel/>.
- [15] php.net. Función crypt, . URL <https://www.php.net/manual/es/function.crypt.php>.
- [16] php.net. password_hash(), . URL <https://www.php.net/manual/es/function.password-hash.php>.
- [17] pildorasinformaticas. Tutorial laravel 2. URL https://www.youtube.com/watch?v=0sHSrqrZCnM&list=PLU8oAlHdN5Bk-qkvjER90g2c_jVmpAHBh.
- [18] proyectos ágiles. Definición de scrum. URL <https://proyectosagiles.org/que-es-scrum/>.
- [19] tutorialesya. Curso de php. URL <https://www.tutorialesprogramacionya.com/phpya/>.
- [20] Wikipedia. Adaline, . URL <https://en.wikipedia.org/wiki/ADALINE>.
- [21] Wikipedia. Composer, . URL <https://es.wikipedia.org/wiki/Composer>.
- [22] Wikipedia. Docker, . URL [https://es.wikipedia.org/wiki/Docker_\(software\)](https://es.wikipedia.org/wiki/Docker_(software)).
- [23] Wikipedia. Integración continua, . URL [https://es.wikipedia.org/wiki/Integraci%C3%B3n_continua#:~:text=La%20integraci%C3%B3n%20continua%20\(continuous%20integration,pruebas%20de%20todo%20un%20proyecto](https://es.wikipedia.org/wiki/Integraci%C3%B3n_continua#:~:text=La%20integraci%C3%B3n%20continua%20(continuous%20integration,pruebas%20de%20todo%20un%20proyecto).

- [24] Wikipedia. Visual studio, . URL https://es.wikipedia.org/wiki/Microsoft_Visual_Studio.
- [25] Wikipedia. Xampp, . URL <https://es.wikipedia.org/wiki/XAMPP>.
- [26] wikipedia. salt. URL [https://es.wikipedia.org/wiki/Sal_\(criptograf%C3%ADa\)](https://es.wikipedia.org/wiki/Sal_(criptograf%C3%ADa)).
- [27] Wikipedia. Latex — wikipedia la enciclopedia libre, 2015. URL <https://es.wikipedia.org/w/index.php?title=LaTeX&oldid=84209252>. [Internet; descargado 30-septiembre-2015].