



UNIVERSIDAD DE BURGOS
ESCUELA POLITÉCNICA SUPERIOR
Grado en Ingeniería Informática



**TFG del Grado en Ingeniería
Informática**
ARBUBU



Presentado por Félix Movilla Alonso
en Universidad de Burgos — 9 de octubre
de 2019

Tutores: Pedro Renedo Fernández y Antonio
Jesús Canepa Oneto



UNIVERSIDAD DE BURGOS
ESCUELA POLITÉCNICA SUPERIOR
Grado en Ingeniería Informática



D. Pedro Renedo Fernández, profesor del departamento de , área de Lenguajes y Sistemas Informáticos.

Expone:

Que el alumno D. Félix Movilla Alonso, con DNI 71294724Z, ha realizado el Trabajo final de Grado en Ingeniería Informática titulado ARBUBU.

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, 9 de octubre de 2019

Vº. Bº. del Tutor:

Vº. Bº. del co-tutor:

D. Pedro Renedo Fernández

D. Antonio Jesús Canepa Oneto

Resumen

A medida que va pasando el tiempo vemos que es más importante nuestra concienciación con el medio ambiente. Esto es lo que nuestros amigos de UBUVerde tratan de inculcarnos.

Con este proyecto hemos tratado de que todos los alumnos de la Universidad de Burgos tengan acceso a la localización de los árboles singulares de Burgos, así como una descripción de los aspectos más importantes de cada uno de ellos, mediante un diseño web.

Para realizar este proyecto hemos trabajado con Python, que es un lenguaje de programación perfecto para lo que hemos realizado. Python tiene una gran comunidad a sus espaldas y está en constante evolución.

Además hemos necesitado de un framework de desarrollo web, el elegido ha sido Django, ya que está enteramente escrito en Python y es de código abierto.

Por último, hemos necesitado una base de datos donde guardar las características de nuestros arboles, hemos elegido sqlite3, que es la base de datos que viene ya incorporada con nuestro framework.

Descriptores

UBUVerde, Python, Django, Sqlite3, diseño web . . .

Abstract

As time goes by we can see the importance about the environment and how we can help it. This is that UBUVerde's people try to teach us.

With this project we want that all the UBU's students have access to singular trees location in Burgos and the descriptions about their most important details through a web design system.

To work to this project we have used Python, because it uses the perfectly programming language for what we want to do. Python has many supports in the community and experience. It is always in constant evolution.

Futhermore we have needed a framework of web development. The chosen one has been Django because it is completely written by Python, and it has a code opened.

Finally, we have needed a database where we can save the tree's details. We have chosen Sqlite3, because it's the database we have in our framework.

Keywords

UBUVerde, Python, Django, Sqlite3, web design . . .

Índice general

| | |
|--|------------|
| Índice general | III |
| Índice de figuras | V |
| Índice de tablas | VI |
| Introducción | 1 |
| 1.1. Descripción del contenido del trabajo | 1 |
| 1.2. Estructura de la memoria | 1 |
| 1.3. Estructura de los anexos | 2 |
| 1.4. Contenido del Cd | 2 |
| Objetivos del proyecto | 3 |
| 2.1. Objetivos Generales | 3 |
| 2.2. Objetivos Técnicos | 3 |
| Conceptos teóricos | 5 |
| 3.1. Estructura de datos | 5 |
| 3.2. Leaflet | 6 |
| Técnicas y herramientas | 7 |
| 4.1. Técnicas Metodológicas | 7 |
| 4.2. Herramientas de Desarrollo | 8 |
| Aspectos relevantes del desarrollo del proyecto | 15 |
| Trabajos relacionados | 17 |

| | |
|---|-----------|
| Conclusiones y Líneas de trabajo futuras | 19 |
| Bibliografía | 21 |

Índice de figuras

| | |
|---|----|
| 3.1. Ejemplo mapa interactivo | 6 |
| 4.2. Logo Python | 8 |
| 4.3. Logo Django | 9 |
| 4.4. Logo SQLite3 | 10 |
| 4.5. Logo Atom | 11 |
| 4.6. Logo GitHub | 12 |
| 4.7. Logo LaTeX | 12 |
| 4.8. Logo StarUml | 13 |

Índice de tablas

Introducción

1.1. Descripción del contenido del trabajo

Ante la creciente demanda de las personas por tener identificado cada cosa que le rodea, surgió la idea de Arbubu.

Arbubu trata de poner en conocimiento de las personas los diferentes árboles singulares que se encuentran en las zonas universitarias, en un futuro podría expandirse a toda la ciudad de Burgos o a otras ciudades o incluso a otro tipo de ámbitos, pero de ello hablaremos más adelante en Conclusiones Lineas de trabajo futuras(añadir hiperenlace, cuando se cree).

Con todo esto tratamos de que los alumnos universitarios y las personas de todos los ámbitos y edades tengan a un solo click la información de los árboles singulares que les rodean.

Para facilitar la búsqueda de dichos árboles hemos incorporado un mapa interactivo que nos muestra en tiempo real donde están situados cada uno de ellos, con una ventana de información de la especie y del propio individuo.

Además para los más metidos en el tema de la naturaleza y los árboles se incorpora unos filtros de búsqueda, para que busquen un árbol en particular por el que están interesados.

1.2. Estructura de la memoria

La memoria se ha estructurado siguiendo los siguientes apartados:

- **Introducción:** descripción del contenido del trabajo.

- **Objetivos del Proyecto:** explicación de los objetivos generales y técnicos del proyecto.
- **Conceptos teóricos:** explicación de los principales conceptos teóricos.
- **Técnicas y herramientas:** descripción breve y concisa de las técnicas y herramientas utilizadas en el proyecto.
- **Aspectos relevantes del desarrollo del proyecto:** explicación y desarrollo de los aspectos más relevantes del proyecto.
- **Trabajos relacionados:** descripción de trabajos que tengan cierto parecido al nuestro.
- **Conclusiones líneas de trabajo futuras:** descripción de las posibles líneas de trabajo futuras y conclusiones del proyecto.

1.3. Estructura de los anexos

- **Plan de proyecto:** planificación temporal y estudio de la viabilidad económica y legal del proyecto.
- **Requisitos:** especificación de los requisitos que se establecen al principio del proyecto.
- **Diseño:** muestra la información relacionada con el diseño de la interfaz además del diseño de clases.
- **Manual del programador:** recoge la instalación de herramientas, la compilación, ejecución del proyecto y pruebas.
- **Manual del usuario:** guía de usuario con instrucciones que puedan facilitar el correcto manejo de la aplicación.

1.4. Contenido del Cd

- **Memoria:** contenido de la memoria en formato pdf.
- **Anexos:** contenido de los anexos en formato pdf.
- **Vídeo explicativo:** vídeo explicando el funcionamiento básico de la aplicación web.
- **Código:** versión del código más reciente de la aplicación web.

Objetivos del proyecto

El principal objetivo del proyecto es realizar un diseño web, en el cual se puedan ver los árboles singulares de las zonas universitarias de Burgos, con sus principales características.

A través de un mapa podremos ver donde están ubicados los árboles.

2.1. Objetivos Generales

- Observar en un mapa los árboles singulares de las zonas universitarias de Burgos y ver sus características.
- Filtrar a través de la familia, nombre científico y nombre común de la especie, autóctona y motivo singular donde se sitúan los árboles buscados y sus características.
- Loguearnos como usuario y ser capaces de importar y descargar datos de los árboles.
- Realizar una primera toma de contacto con la búsqueda de árboles, que en una futura mejora no solo busquemos arboles, es decir, que seamos capaces de buscar monumentos, lugares importantes ...
- Poder compartir y dar me gusta a la página de facebook de UbuVerde e interactuar con ellos a través de twitter.

2.2. Objetivos Técnicos

- Ser capaz de introducir datos en la base de datos Sqlite3.

- Ser capaz de descargar datos de la base de datos Sqlite3.
- Plasmar en el mapa esos datos introducidos en la base de datos.
- Programar en Python y html el diseño web que va a tener nuestro proyecto.
- Guardar en un repositorio de GitHub los cambios que hemos ido realizando.
- Utilizar el framework Django para realizar correctamente nuestro diseño web.

Conceptos teóricos

En esta sección vamos a hablar de la forma en el cual vamos a clasificar nuestros modelos de datos, así como los conceptos teóricos relacionados con el proyecto.

3.1. Estructura de datos

Todos los seres vivos podemos clasificarnos según unas categorías taxonómicas [21], pero en este caso nos vamos a centrar en lo va el proyecto, que no es otro que los árboles.

Hemos elegido esta estructura de los datos porque con ellos somos capaces de englobar todos los aspectos que más se adecuan a nuestro proyecto.

No hemos decidido introducir más modelos de entidades ya que los comunes son los especificados y todo lo que fuera añadir alguna entidad más no nos permitiría desarrollar el proyecto de la mejor manera posible.

En nuestro proyecto vamos a clasificar los árboles de esta manera:

- **Familia:** Es la agrupación de árboles que se encuentran en un orden, por características comunes entre ellos.
- **Género:** De las familias provienen los géneros, conjuntos de especies relacionadas entre sí por características comunes.
- **Especie:** Es un grupo de individuos con las mismas características.
- **Individuo:** Son cada uno de los árboles, con unas características particulares.

3.2. Leaflet

Leaflet [16] es una biblioteca JavaScript de código abierto utilizada para plasmar en nuestra página web mapas interactivos, los cuales son compatibles con dispositivos móviles.

Se caracteriza por la sencillez, simplicidad, rendimiento y usabilidad, lo que hace que sea una herramienta perfecta para nuestro proyecto.

También es posible añadirle infinidad de plugins que lo hace todavía más completo.

Para verlo con más claridad incluyo una imagen 3.1 en la cual podemos observar un mapa con los países por los que ha pasado o vivido el titular del blog Andy Maloney.



Figura 3.1: Ejemplo mapa interactivo [3]

Técnicas y herramientas

En esta sección vamos a hablar de las técnicas metodológicas y de las herramientas de desarrollo seguidas durante el proyecto.

4.1. Técnicas Metodológicas

No hemos seguido una metodología pura, es decir, no me he basado simplemente en una sola metodología si no que he ido eligiendo aspectos de varias de ellas.

En primer lugar, tomamos aspectos de la **metodología en cascada** [17], ya que hemos partido de unos requisitos iniciales, que posteriormente hemos ido adaptando según los cambios y necesidades del cliente.

Más adelante una vez que teníamos claros los requisitos iniciales, nos hemos centrado más en la **metodología scrum** [30]. Decimos que utilizamos esta metodología porque a lo largo del desarrollo del proyecto hemos tenido reuniones semanales con los tutores, además de una reunión inicial con el encargado de Ubu Verde.

En todo momento hemos ido mezclando ambas técnicas ya que a las reuniones semanales con los tutores se fueron añadiendo a continuación el diseño e implementación de la aplicación.

Una vez que el diseño e implementación estaban realizados fuimos añadiendo las distintas pruebas para comprobar que no dejábamos cabos sueltos.

A medida que íbamos realizando las distintas partes del proyecto se iban subiendo al repositorio de Github.

4.2. Herramientas de Desarrollo

Lenguaje de Programación

Lo primero antes de elegir las herramientas que vamos a utilizar es elegir el lenguaje de programación, tenemos infinidad de lenguajes para desarrollar nuestro proyecto, pero los más viables para ello creí que eran **Php** [25] y **Python** [29] Ver figura. 4.2

Me decanté por Python ya que es un lenguaje que me entra más por la vista y es más intuitivo, es un lenguaje más nuevo que Php por lo que en un futuro cuando trabaje habrá menos gente que conozca este lenguaje, es decir, menos competencia, existe una gran comunidad con gran cantidad de tutoriales ...



Figura 4.2: Logo Python [12]

Framework

A la hora de decidir entre que framework elegir, me planteé dos posibles opciones que fueron **Flask** [10] y **Django** [33] Ver figura. 4.3

Finalmente me decidí por Django porque me pareció un framework más avanzado, te facilita mucho su desarrollo, ya que gran parte del código viene implementada y no es necesario programarlo, es el más utilizado por lo tanto es el que más comunidad tendrá a sus espaldas en caso de fallo o duda, es seguro ya que implementa medidas de seguridad por defecto y evita fallos como el SQL Injection, incluye una interfaz para acceder a la base de datos ...



Figura 4.3: Logo Django [15]

Bases de Datos

Existen 4 posibles opciones de bases de datos para elegir con Django:

- **PostgreSQL** [26] es un sistema de gestión de bases de datos relacional orientado a objetos y de código abierto.
- **SQLite 3** [31] es un sistema de gestión de bases de datos relacional, el conjunto de la base de datos (definiciones, tablas, índices, y los propios datos), son guardados como un solo fichero estándar en la máquina host. Permite bases de datos de hasta 2 Terabytes de tamaño, y también permite la inclusión de campos tipo BLOB. Ver figura. 4.4
- **MySQL** [23] es un sistema de gestión de bases de datos relacional desarrollado bajo licencia dual: Licencia pública general/Licencia comercial por Oracle Corporation y está considerada como la base de datos de código abierto más popular del mundo.
- **Oracle** [24] es un sistema de gestión de base de datos de tipo objeto-relacional, su dominio en el mercado de servidores empresariales había sido casi total hasta que recientemente tiene la competencia del Microsoft SQL Server y de la oferta de otros RDBMS con licencia libre como PostgreSQL, MySQL o Firebird.

Finalmente me decanté por SQLite 3 ya que es la que viene implementada por defecto con Django, es la más sencilla de las cuatro, pero para el proyecto que estamos desarrollando es más que suficiente.



Figura 4.4: Logo SQLite3 [1]

IDE

Existen infinidad de IDEs para trabajar con Python, decidí buscar información para decantarme por uno u otro:

- **PyDev** [28] es un IDE open source que se ejecuta en Eclipse, incluye modo depuración de Django, multilingüe, análisis de código, marcado de errores ...

No fue la elegida ya que he trabajado varias veces con Eclipse y no me termina de gustar.

- **PyCharm** [27] es un IDE con dos versiones, la open source (bastante limitada a mi modo de ver) y la profesional. Incluye autocompletado de código, navegación intuitiva, depurador gráfico ...

No resultó tampoco la elegida debido a su limitación y falta de prestaciones en su versión open source.

- **VIM** [32] es un IDE open source con licencia GPL, es ligero y rápido. Su configuración resulta un poco costosa debido a que necesita varios complementos para que funcione en su máximo esplendor, por lo que por esto tampoco resultó elegido.

- **Atom** [19] es un IDE open source, desarrollado por GitHub, lo que lo hace ideal para el control de versiones del proyecto, incluye infinidad de plugins que hacen que lo podamos personalizar a nuestro antojo. Ver figura. 4.5

Es el que más me llamó la atención y el que más cómodo me resultó trabajar, así que por eso elegí este IDE.



Figura 4.5: Logo Atom [7]

Herramientas de Gestión

He observado que existen varias posibilidades para ir subiendo nuestros avances a un repositorio.

Las opciones que me han parecido mejores son las siguientes:

- **BitBucket** [20] es un servicio de alojamiento basado en web, ofrece la posibilidad de cuentas gratuitas pero limitando el número de repositorios.
- **GitHub** [22] es una plataforma de diseño colaborativo de software para alojar proyectos utilizando el sistema de control de versiones Git. Ver figura. 4.6

El elegido ha sido GitHub ya que es una herramienta que hemos utilizado en varias asignaturas y estoy más familiarizado con ella.



Figura 4.6: Logo GitHub [4]

Documentación

A la hora de realizar la documentación de nuestro proyecto existen varias opciones para ello:

- **Microsoft Word** [9] es una aplicación orientada para el procesamiento de textos y permite crear, editar y compartir documentos de Word. Puedes trabajar con otras personas en tiempo real. No es gratuito, aunque existen versiones de prueba, para los estudiantes de la universidad nos ofrecen licencias.
- **Writer** [18] es un procesador de texto multiplataforma, es de código abierto y cada vez está más extendido entre los usuarios.
- **LaTeX** [14] es una herramienta utilizada para realizar documentos de ámbito científico y técnicos, es de software libre. Para su utilización es necesario otra herramienta: **TeXstudio** [6] es un editor de documentos escritos en LaTeX. Ver figura. 4.7

Finalmente la elegida ha sido LaTeX ya que aunque pueda parecer difícil en un primer momento una vez que te acostumbras a ella es la más completa y funcional.



Figura 4.7: Logo LaTeX [8]

Programas para la creación de diagramas

Existe una gran variedad de programas para la realización de diagramas, los que más interesantes me han parecido son los siguientes:

- **Modelio** [2] es una herramienta para realizar modelos de código abierto, es fácil e intuitiva permitiendo añadir nuevas funcionalidades.
- **ArgoUml** [11] es una herramienta de código libre sencilla de utilizar, soporta Uml y es utilizada tanto para ingeniería inversa como para ingeniería de software.
- **StarUml** [13] es una herramienta de código abierto para desarrollar proyectos Uml. Ver figura. 4.8

Finalmente la elegida ha sido StarUml ya que es la herramienta que más me ha gustado y la que más intuitiva me ha parecido de las que he utilizado, además ya había trabajado con ella anteriormente.



Figura 4.8: Logo StarUml [5]

Aspectos relevantes del desarrollo del proyecto

Este apartado pretende recoger los aspectos más interesantes del desarrollo del proyecto, comentados por los autores del mismo. Debe incluir desde la exposición del ciclo de vida utilizado, hasta los detalles de mayor relevancia de las fases de análisis, diseño e implementación. Se busca que no sea una mera operación de copiar y pegar diagramas y extractos del código fuente, sino que realmente se justifiquen los caminos de solución que se han tomado, especialmente aquellos que no sean triviales. Puede ser el lugar más adecuado para documentar los aspectos más interesantes del diseño y de la implementación, con un mayor hincapié en aspectos tales como el tipo de arquitectura elegido, los índices de las tablas de la base de datos, normalización y desnormalización, distribución en ficheros³, reglas de negocio dentro de las bases de datos (EDVHV GH GDWRV DFWLYDV), aspectos de desarrollo relacionados con el WWW... Este apartado, debe convertirse en el resumen de la experiencia práctica del proyecto, y por sí mismo justifica que la memoria se convierta en un documento útil, fuente de referencia para los autores, los tutores y futuros alumnos.

Trabajos relacionados

Este apartado sería parecido a un estado del arte de una tesis o tesina. En un trabajo final grado no parece obligada su presencia, aunque se puede dejar a juicio del tutor el incluir un pequeño resumen comentado de los trabajos y proyectos ya realizados en el campo del proyecto en curso.

Conclusiones y Líneas de trabajo futuras

Todo proyecto debe incluir las conclusiones que se derivan de su desarrollo. Éstas pueden ser de diferente índole, dependiendo de la tipología del proyecto, pero normalmente van a estar presentes un conjunto de conclusiones relacionadas con los resultados del proyecto y un conjunto de conclusiones técnicas. Además, resulta muy útil realizar un informe crítico indicando cómo se puede mejorar el proyecto, o cómo se puede continuar trabajando en la línea del proyecto realizado.

Bibliografía

- [1] Alejandro Suárez Lamadrid y Antonio Suárez Jiménez. Guía rápida de sqlite3. https://python-para-impacientes.blogspot.com/p/blog-page_27.html, 2019. [Online; accessed 07-Oct-2019].
- [2] AlternativeTo. Modelio. <https://alternativeto.net/software/modelio-open/>, Agosto 2019. [Online; accessed 8-Oct-2019].
- [3] Andy Maloney. Creating an interactive map with leaflet and openstreetmap. <https://asmaloney.com/2014/01/code/creating-an-interactive-map-with-leaflet-and-openstreetmap/>, 2014. [Online; accessed 07-Oct-2019].
- [4] Asfo. Subiendo miles y miles de archivos a github fácilmente con git+bash. <https://medium.com/@asfo/subiendo-miles-y-miles-de-archivos-a-github-f%C3%A1cilmente-con-git-bash-e493d4a94fe0>, Junio 2018. [Online; accessed 07-Oct-2019].
- [5] Azure-Aws-google-Cloud. Staruml on cloud. <https://secureanyccloud.com/staruml-on-cloud-technical-support-cloud-help-azure-aws-opensource-cognosys/>, Noviembre 2016. [Online; accessed 07-Oct-2019].
- [6] Damián Amoedo. Textstudio, un entorno de escritura para crear documentos latex. <https://ubunlog.com/textstudio-crear-documentos-latex/>, Septiembre 2019. [Online; accessed 8-Oct-2019].
- [7] davr59. Editor de texto atom y sincronización de su configuración en la nube. <https://dispuestoaaprender.com/2015/11/15/>

- [editor-de-texto-atom-y-sincronizacion-en-la-nube-de-su-configuracion/](#), Noviembre 2015. [Online; accessed 07-Oct-2019].
- [8] GUILLERMO JULIÁN. Cómo instalar y usar latex en windows. <https://www.xatakawindows.com/aplicaciones-windows/como-instalar-y-usar-latex-en-windows>, Abril 2014. [Online; accessed 07-Oct-2019].
- [9] live.com. Word online. <https://chrome.google.com/webstore/detail/word-online/fiombgjlkfdpdkbhfoofeeinbehmajg?hl=es>, Octubre 2016. [Online; accessed 8-Oct-2019].
- [10] Matt Makai. Flask. <https://www.fullstackpython.com/flask.html>, 2019. [Online; accessed 4-Oct-2019].
- [11] MediaWiki. Argouml. <https://www.ecured.cu/ArgoUML>, Julio 2019. [Online; accessed 8-Oct-2019].
- [12] Python Software Foundation. Python. <https://www.python.org/>, 2019. [Online; accessed 07-Oct-2019].
- [13] Source Forge. Staruml. <https://sourceforge.net/projects/staruml/>, Septiembre 2019. [Online; accessed 8-Oct-2019].
- [14] The Latex Project. Latex – a document preparation system. <https://www.latex-project.org/>, Agosto 2019. [Online; accessed 8-Oct-2019].
- [15] Tobias McNulty. Django is boring, or why tech startups (should) use django. <https://www.caktusgroup.com/blog/2016/12/14/django-boring-or-why-tech-startups-should-use-django/>, Diciembre 2016. [Online; accessed 07-Oct-2019].
- [16] Vladimir Agafonkin. Leaflet - an open-source javascript library for mobile-friendly interactive maps. <https://leafletjs.com/>, 2019. [Online; accessed 07-Oct-2019].
- [17] Desarrollo web. El modelo en cascada: desarrollo secuencial de software. <https://www.ionos.es/digitalguide/paginas-web/desarrollo-web/el-modelo-en-cascada/>, Marzo 2019. [Online; accessed 18-Sep-2019].
- [18] Wikipedia contributors. Apache openoffice writer. https://es.wikipedia.org/wiki/Apache_OpenOffice_Writer, Agosto 2019. [Online; accessed 8-Oct-2019].

- [19] Wikipedia contributors. Atom. [https://es.wikipedia.org/wiki/Atom_\(software\)](https://es.wikipedia.org/wiki/Atom_(software)), Agosto 2019. [Online; accessed 5-Oct-2019].
- [20] Wikipedia contributors. Bitbucket. <https://es.wikipedia.org/wiki/Bitbucket>, Septiembre 2019. [Online; accessed 8-Oct-2019].
- [21] Wikipedia contributors. Categoría taxonómica — Wikipedia, the free encyclopedia. https://es.wikipedia.org/wiki/Categoría_taxonómica, 2019. [Online; accessed 02-Jun-2019].
- [22] Wikipedia contributors. Github. <https://es.wikipedia.org/wiki/GitHub>, Septiembre 2019. [Online; accessed 8-Oct-2019].
- [23] Wikipedia contributors. Mysql. <https://es.wikipedia.org/wiki/MySQL>, Agosto 2019. [Online; accessed 30-Sep-2019].
- [24] Wikipedia contributors. Oracle database. https://es.wikipedia.org/wiki/Oracle_Database, Septiembre 2019. [Online; accessed 30-Sep-2019].
- [25] Wikipedia contributors. Php. <https://es.wikipedia.org/wiki/PHP>, Septiembre 2019. [Online; accessed 2-Oct-2019].
- [26] Wikipedia contributors. Postgresql. <https://es.wikipedia.org/wiki/PostgreSQL>, Septiembre 2019. [Online; accessed 1-Oct-2019].
- [27] Wikipedia contributors. Pycharm. <https://en.wikipedia.org/wiki/PyCharm>, Octubre 2019. [Online; accessed 5-Oct-2019].
- [28] Wikipedia contributors. Pydev. <https://en.wikipedia.org/wiki/PyDev>, Julio 2019. [Online; accessed 5-Oct-2019].
- [29] Wikipedia contributors. Python. <https://es.wikipedia.org/wiki/Python>, Octubre 2019. [Online; accessed 2-Oct-2019].
- [30] Wikipedia contributors. Scrum (desarrollo de software). [https://es.wikipedia.org/wiki/Scrum_\(desarrollo_de_software\)](https://es.wikipedia.org/wiki/Scrum_(desarrollo_de_software)), Septiembre 2019. [Online; accessed 18-Sep-2019].
- [31] Wikipedia contributors. Sqlite. <https://es.wikipedia.org/wiki/SQLite>, Septiembre 2019. [Online; accessed 1-Oct-2019].
- [32] Wikipedia contributors. Vim. <https://es.wikipedia.org/wiki/Vim>, Octubre 2019. [Online; accessed 5-Oct-2019].
- [33] Threespot y Andrevv. Meet django. <https://www.djangoproject.com/>, 2019. [Online; accessed 4-Oct-2019].