



UNIVERSITAT^{DE}
BARCELONA

Trabajo de final de grado

**GRADO DE INGENIERIA
INFORMÁTICA**

**Facultad de Matemáticas e Informática
Universitat de Barcelona**

Portal web SIDBRINT

Autor: Albert Lázaro de Lara

Director: Dr. Simone Balocco
Responsable: Lourdes Prades Artigas
Realizado en: Departament Matemàtica
Aplicada i Anàlisi

Barcelona, 22 de junio de 2018

Abstracto (Castellano)

Sidbrint es un portal de investigación de la Universidad de Barcelona que recopila información sobre las Brigadas Internacionales de la Guerra Civil Española. Este portal es una plataforma online colaborativa que permite a investigadores de toda Europa consultar e incorporar nuevos datos.

En este portal se han reportado problemas por parte de usuarios que lo usan, y además, el portal es antiguo y no aprovecha las nuevas tecnologías para presentar datos y resultados, por lo que la intención del proyecto es analizar el portal tanto a nivel de usabilidad como a nivel de estructura y mejorar tanto las tecnologías usadas en él, como la visualización de la interfaz para que los usuarios puedan acceder con facilidad a un entorno digital complejo.

Se plantean dos objetivos: el primero trata de arreglar los problemas de la plataforma actual y el segundo se trata de plantear una nueva arquitectura para el portal, con el consecuente cambio de tecnologías y sus consecuencias, a partir del análisis que se realiza.

Este proyecto permite tocar diversos ámbitos como el análisis de una interfaz de usuario, el análisis de requerimientos por parte de un cliente o la puesta en marcha de un proyecto de software, tanto de front-end como de back-end.

Abstracto (Català)

Sidbrint és un portal de investigació de la Universitat de Barcelona que recopila informació sobre las Brigades Internacionals de la Guerra Civil Espanyola. Aquest portal és una plataforma online col·laborativa que permet a investigadors de tota Europa consultar i incorporar noves dades.

En aquest portal s'han reportat problemes de usabilitat per part dels usuaris que el consulten, a part que el portal és antic i no aprofita les noves tecnologies per presentar dades i resultats, per tant la intenció del projecte és analitzar el portal tant a nivell de usabilitat com a nivell d'estructura i millorar tant les tecnologies utilitzades en ell, com la visualització de la interfase per a que els usuaris puguin accedir amb facilitat a un entorno digital complex.

Es plantegen dos objectius: el primer tracta d'arreglar els problemes de la plataforma actual i el segon tracta de plantejar una nova arquitectura per al portal, amb el conseqüent canvi de tecnologies i les seves conseqüències, a partir de l'anàlisi que es realitza.

Aquest projecte permet tocar diversos àmbits com l'anàlisi de una interfase de usuari, l'anàlisi de requeriments per part de un client o la posta en marxa de un projecte de software, tant de front-end com de back-end.

Abstracto (English)

Sidbrint is an investigation portal of Barcelona University that collects information about International Brigades of the Spanish Civil War. This portal is a collaborative online platform that allow to researchers of all Europe to consult and incorporate new data.

Usability problems have been reported on this portal by users who consult it, and also, this site is old and it doesn't take advantage of new technologies to present data and results, so the intention of this project is to analyze the portal both at usability level and at structure level and improve the technologies used and the visualization of the user interface so that users can access easily to a complex digital environment.

Two objectives are proposed: the first tries to fix the problems of the actual platform and the second tries to propose a new architecture for the site, with the consequent change of technologies and his consequences, from the analysis made.

This project allows to work on diverse fields like user interface analysis, the analysis of requirements of the client ideas or the startup of a software project, both frontend and backend.

Agradecimientos

Quiero agradecer a todas las personas que me han apoyado en los estudios del grado de ingeniería informática durante estos años. A mi familia que me ha apoyado desde el principio del grado y en los momentos más difíciles de este, a los compañeros que he ido conociendo durante el grado y que tantos momentos hemos compartido en las clases del grado, en la biblioteca y salas de estudio de la facultad para preparar los trabajos y exámenes que hemos hecho a lo largo del grado y a los profesores por transmitir los conocimientos necesarios y por resolver las dudas que iban surgiendo a lo largo de los cursos.

Índice

1. Introducción	1
1.1. Objetivo	1
1.2. Motivación	2
2. Antecedentes	3
2.1. Inicio del proyecto	3
2.2. Desarrollo del proyecto	3
2.3. Colaboraciones y repercusión mediática	4
3. Planificación	6
3.1. Previsión	6
3.2. Real	7
3.3. Tareas a realizar	7
3.3.1. Desarrollo actual	8
3.3.2. Desarrollo futuro	8
3.4. Reuniones	9
4. Sistema actual	11
4.1. Drupal	11
4.1.1. Nodo	11
4.1.2. Tipos de entidad	12
4.1.3. Comentario	12
4.1.4. Taxonomía	12
4.1.5. Usuario	12
4.1.6. Módulo	12
4.1.7. Regiones y bloques	13
4.1.8. Menús	15
4.1.9. Temas	15
4.1.10. Vistas	15
4.1.11. Diseño de vistas	16
4.1.12. Base de datos	16
4.1.13. Ruta	16
4.1.14. Bootstrap	16
4.1.15. Permisos	16

4.1.16. Caché	16
4.2. Bootstrap framework	17
4.3. Drush	17
4.4. Base de Datos	17
4.5. Cambios realizados	20
4.5.1. Creación de un subtema	21
4.5.2. Configuración de bloques	21
4.5.3. Creación de bloques	21
4.6. Resultados	22
4.6.1. Página principal	22
4.6.2. Página resultados	25
4.6.3. Página ficha brigadista	28
4.6.4. Página ficha fuente documental	30
5. Sistema futuro	32
5.1. Tecnologías escogidas	32
5.1.1. Servidor LAMP - Linux / Apache / Mysql / PHP	32
5.1.2. Laravel	32
5.2. Análisis de requerimientos	33
5.2.1. Requisitos	33
5.2.2. Casos de uso	34
5.3. Diseño de la base de datos	38
5.3.1. Tablas Principales	41
5.3.2. tabla usuario	42
5.3.3. Tablas imagen y archivo	43
5.3.4. Tabla configuración	44
5.3.5. Script migración	44
5.4. Seguridad	45
5.4.1. Autenticación	45
5.4.2. Autorización	46
5.4.3. Encriptación	46
5.4.4. Hashing	47
6. Conclusiones	48
7. Anexo I. Reuniones	51

8. Anexo II. Base de datos Drupal	59
9. Anexo III. Casos de uso en detalle	66

1. Introducción

Con el paso del tiempo las tecnologías para crear portales web han ido evolucionando, los estándares de diseño web han cambiado y han aparecido nuevos dispositivos con una amplia variedad de tamaños de pantalla.

Sidbrint¹ es un portal sobre la Guerra Civil Española, que recopila grandes cantidades de información sobre brigadistas, fuentes documentales y encuadramientos militares. Estos datos son añadidos por los responsables del proyecto y por estudiantes u otros colaboradores a través de una parte del portal exclusiva para usuarios registrados.

Los datos almacenados pueden ser consultados por cualquier usuario a través de la búsqueda y filtros que ofrecen el portal.

Sidbrint se creó con Drupal², un gestor de contenido pensado para que personas con pocos conocimientos de informática puedan crear su página web en él, a través de plugins que realizan diferentes funciones y sin necesidad de tener conocimientos de programación o maquetación web. Este portal ha ido cambiando con la ayuda de diferentes empresas, en un trabajo de mantenimiento, pero no es suficiente ya que el portal necesita hacer cambios más notables.

A parte de problemas de funcionamiento que se han reportado, se ha hecho un estudio de interacción del portal a varios usuarios, con el resultado de que este es poco intuitivo al uso de las personas.

1.1. Objetivo

El objetivo de este proyecto es el de llevar a cabo un análisis de las peticiones de los investigadores de Sidbrint, que dividirán el proyecto en dos partes: La primera petición consiste en implementar una serie de mejoras en el portal web, tales como un rediseño de la interfaz del sitio y la corrección de ciertos errores reportados. La segunda, consiste en plantear una nueva arquitectura, utilizando tecnologías mas modernas.

Para ello se actualizarán las tecnologías que se han estado usando en esta aplicación, sustituyéndolas por tecnologías más actuales. Este cambio conllevará la re-estructuración del diseño de la base de datos usada, la migración de los datos a la nueva base de datos, el análisis de seguridad para proteger la aplicación, etc.

El hosting del portal está en manos de una empresa que gestiona el servidor que contiene Sidbrint. Esta empresa es la que crea una imagen del servidor para poder trabajar con una copia de las tecnologías y configuraciones que usa la plataforma actual.

El análisis consiste en definir el proyecto, creando una planificación donde se estimarán las tareas a realizar durante los meses en los que se trabajará en este proyecto. Estas tareas, serán consensuadas tanto con los investigadores de Sidbrint

¹www.sidbrint.ub.edu

²www.drupal.org

como con la empresa que se encarga del Drupal y se les asignarán unos tiempos para ser llevadas a cabo.

Los resultados del estudio de interacción serán analizados para crear una web más intuitiva tanto para los usuarios que visitan el portal como para los usuarios que suben contenido nuevo y se dedican a su gestión.

1.2. Motivación

El hecho de trabajar para una organización (en este caso los investigadores de Sidbrint), con un problema tecnológico real, fue la motivación para elegir este proyecto, ya que se puede enfocar desde un punto de vista profesional, donde una empresa sin los conocimientos tecnológicos necesarios recurre a otra empresa con los conocimientos y aptitudes para llevar a cabo la tarea requerida.

La organización del trabajo es una aptitud muy valorada en el mundo laboral. El hecho de organizar el proyecto desde cero, llevando a cabo la planificación de éste, haciendo reuniones con los clientes para evaluar el proyecto, crear las tareas, ver cuales se podrán llevar a cabo y cuales no por falta de recursos como el tiempo y hacer un seguimiento de la evolución del proyecto es una buena manera de entender el trabajo que realizan los *project managers* y ponerse en la piel de uno.

La programación web sigue siendo a día de hoy un sector de la informática con mucha demanda, desde la programación *front-end*, encargada del funcionamiento de la parte cliente de una aplicación web al *back-end*, que se encarga del funcionamiento del servidor de la aplicación. Además, el software distribuido, aquel que se comunica con diferentes máquinas, es el más vulnerable en cuanto a ataques informáticos y requiere de una buena programación para evitar agujeros de seguridad o la extracción de datos confidenciales y sensibles.

2. Antecedentes

En este apartado se hace una explicación de los antecedentes del proyecto, para entender la historia que hay detrás del proyecto y la evolución que ha sufrido.

2.1. Inicio del proyecto

El proyecto se inicia en el año 2011 con el propósito de crear el primer Sistema Digital sobre las Brigadas Internacionales y los Brigadistas de la Guerra Civil española (SIDBRINT).

El primer objetivo planteado es el diseño de un portal web para recuperar los brigadistas a través de las fuentes documentales.

Seguidamente se plantea el objetivo de diferenciar los diferentes tipos de datos que se quieren almacenar en la base de datos: Brigadistas, Fuentes, Encuadramiento militar e Información digital.

El tercer objetivo fue la investigación de cuáles son los sistemas de organización y clasificación del conocimiento, creación de una taxonomía histórica, un tesoro de historia de la Guerra Civil y las Brigadas Internacionales, y una clasificación automática que asegurasen en su conjunto la accesibilidad total al contenido de las fuentes documentales que tratan las Brigadas Internacionales y los Brigadistas de la Guerra Civil española.

El último objetivo consistía en que el sistema SIDBRINT organizara y posibilitara la accesibilidad al Patrimonio Histórico, y potenciara un nuevo espacio de investigación y de conocimiento sobre la Memoria Histórica de la Guerra Civil española. Espacio digital que no existía aún en España.

2.2. Desarrollo del proyecto

El desarrollo de Sidbrint prácticamente es realizado en el año 2012 siguiendo el plan de trabajo propuesto en el proyecto. El diseño del portal se desarrolló en todos sus procesos y requerimientos a partir del orden siguiente:

1. Portal web:

- CMS (Gestión contenidos digitales). Se realiza la migración a Drupal 7.
- Diseño Web: se cambia de estilo a uno más limpio. Se hace uso de fondos en blanco para mejorar la legibilidad y usabilidad del portal.
- Accesibilidad: se varían los colaboradores en estático para facilitar la lectura.
- Navegabilidad: se mejora la navegabilidad del portal con una nueva taxonomía corporativa, y con motores de búsqueda internos. Se introduce mejora de las búsquedas y creación de buscadores específicos por tipo de contenido. Se programan las búsquedas con Motor Apache SolR.

- Estándares: se realiza la adaptación de CSS y HTML a los estándares web (W3C).
- Formatos de la información (salida): se desarrollan los formatos de salida de presentación para las fichas (Brigadistas).
- Formatos de la información (entrada): se desarrollan los formatos de entrada para la introducción de datos en las fichas de todas y cada una de las fuentes y de los datos.

2. Bases de datos:

- Revisión e integración de las bases de datos existentes (Brigadistas, Fuentes Documentales y Encuadre Militar). Importación y exportación de los registros. Refinamiento de los datos para su interconexión entre las diversas bases y se resuelven bugs de categorías.

3. Administración y operatividad de portal:

- Creación de áreas y menús personalizados según el rol de cada usuario.
- Menú adaptado según los perfiles de usuarios: según normas de usabilidad y según los targets de los mismos.
- Asignación de permisos para la correcta utilización del portal.
- Compatibilidad con los diferentes navegadores.
- Panel de edición de taxonomías para usuarios avanzados.

2.3. Colaboraciones y repercusión mediática

En el proyecto Sidbrint colaboran activamente particulares, asociaciones, archivos, bibliotecas y centros de investigación.

Destaca sobretodo el trabajo de muchos voluntarios con diferentes disciplinas como bibliotecarios, archivadores, médicos, antropólogos, historiadores, especialistas en arte, lingüistas, expertos en redes sociales y tecnólogos, tanto del ámbito español como del ámbito internacional.

La participación de mas de 35000 voluntarios extranjeros en la guerra, provenientes de más de 50 países, asegura el interés de Sidbrint en el ámbito internacional. Ésto está demostrado ya que:

- Sidbrint recibe respuestas desde otros países a través de emails, entradas en el portal, donaciones de información por parte de familiares de brigadistas i especialistas en la Guerra Civil.
- Existe una respuesta internacional por parte de instituciones científicas que han contactado con Sidbrint, han ofrecido su colaboración y han manifestado su interés por el proyecto.

- Se ha manifestado interés por parte de los especialistas más importantes en Brigadas Internacionales a título individual de todo el mundo.
- Existe una vinculación emocional de numerosos familiares de brigadistas que han encontrado en Sidbrint la posibilidad de conocer aspectos de las biografías de sus antepasados que desconocían.
- Se han recibido invitaciones desde diversas instituciones internacionales para presentar Sidbrint.
- Los investigadores de Sidbrint han participado en congresos internacionales.

Desde el año 2014 Sidbrint está financiado por el Vicerrectorado de Investigación de la Universidad de Barcelona, además de ser un portal institucional. Gracias a este acuerdo los estudiantes de la Universidad pueden hacer sus Trabajos de Fin de Grado en relación a Sidbrint.

3. Planificación

Este proyecto cuenta con muchas tareas a realizar en un período de seis meses de actividad. Desde la primera reunión han ido apareciendo nuevas tareas, por lo que se priorizan las tareas en función de su importancia y su urgencia. Para ello se ha realizado una planificación del proyecto y se han escogido las tareas con mas prioridad, dejando a un lado las menos prioritarias.

Para reflejar las planificaciones se opta por usar los diagramas de *Gantt*, que se verán en los siguientes apartados. Estos diagramas son una herramienta gráfica para poder exponer el tiempo de dedicación previsto para las tareas del proyecto a lo largo del tiempo definido y ver el tiempo real que se ha tardado en realizarse.

3.1. Previsión

Para el proyecto, se hizo una previsión, que consiste en predecir cuantos días se tardará en hacer cada tarea seleccionada dentro del tiempo estimado, de tal manera que se puede hacer una estimación de que tareas se realizarán.

Al principio, se planteó que el proyecto estaría enfocado en la creación de la página desde cero utilizando tecnologías web mas modernas, el diseño de una nueva estructura de la base de datos empezando a estructurar el nuevo sitio, tal y como refleja la figura 1 (la formación en las tecnologías se hizo previamente al planteamiento de la previsión).

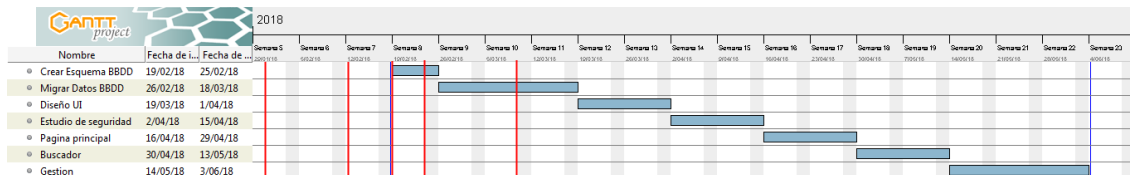


Figura 1: Diagrama de Gantt de la previsión inicial.

Las líneas verticales rojas que aparecen en el Diagrama indican los días en que se produjo una reunión. El desarrollo del proyecto se ha realizado con ciclos de SCRUM, donde cada sprint es el tiempo entre dos reuniones.

Poco antes de la quinta reunión se decide, junto a los responsables de Sidbrint y del servidor, cambiar las tareas para dar prioridad a la resolución de los problemas del sistema actual. Se toma esta decisión para que el proyecto Sidbrint pueda mantener la inversión que recibe, dado que se necesita observar cambios visuales en el portal. Para ello se hace una nueva planificación con las nuevas tarea (figura 2).

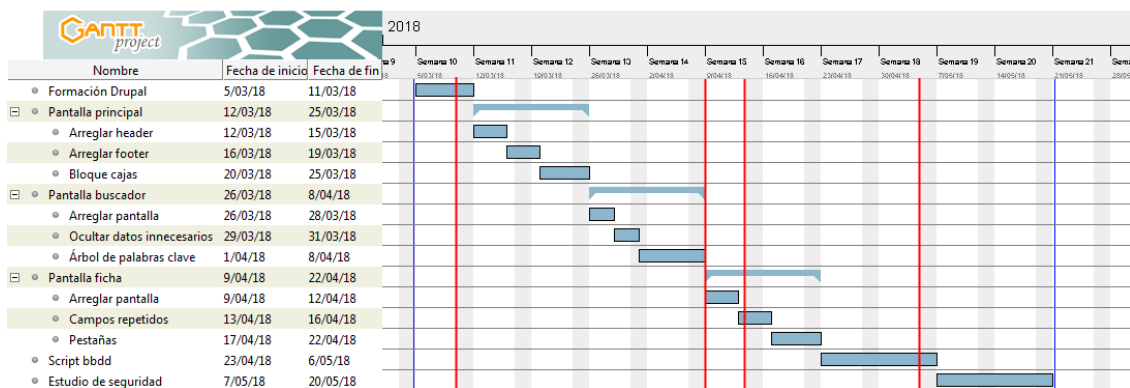


Figura 2: Diagrama de Gantt de la siguiente previsión.

Este cambio añade las tareas de cambios visuales en Drupal, corrección de errores y además añade una semana de formación en Drupal.

3.2. Real

A continuación se muestra el diagrama de la realidad del trabajo realizado durante el proyecto una vez terminado.

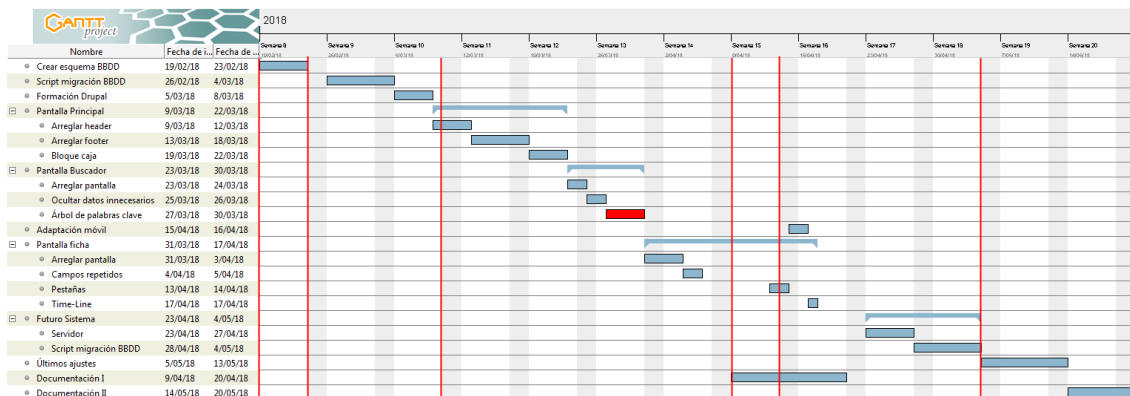


Figura 3: Diagrama de Gantt del trabajo real.

3.3. Tareas a realizar

Este proyecto está compuesto de una serie de tareas acordadas con los responsables de Sidbrint y de Drupal a través de varias reuniones realizadas (estas reuniones son descritas en la siguiente sección).

Al estar el proyecto dividido en dos partes, se procede a describir las tareas en dos secciones diferentes: la sección para el desarrollo actual y la sección para el desarrollo futuro.

3.3.1. Desarrollo actual

El desarrollo actual es la parte del proyecto en la que más tareas existen. Este está formado por un conjunto de tareas centradas en Drupal:

- **Instalación tema nuevo:** la primera tarea consiste en instalar un nuevo tema de Drupal basado en Bootstrap para facilitar la implementación de los cambios deseados.
- **Página principal:** esta tarea consiste en implementar una nueva portada para el sitio, de forma que los problemas de la anterior portada sean eliminados. Para ello se eliminarán elementos existentes y se añadirán otros elementos.
- **Página buscador:** se desea arreglar los problemas de interfaz de esta página, con la idea de poder mostrar los resultados de búsqueda con los datos mínimos. Para mejorar la experiencia de usuario, este ha de poder elegir el número de resultados a mostrar.
- **Página ficha resultado brigadista:** en esta página se requiere mostrar la información de manera distribuida, por lo que se desea instalar un sistema de pestañas y así poder distribuir mejor la información por la página.
- **Página ficha resultado fuente documental:** la página que muestra los datos de una fuente documental muestra el nombre de la fuente de manera repetida por la página, se requiere eliminar duplicados y maquetar la página con la ayuda de Bootstrap.

3.3.2. Desarrollo futuro

El desarrollo futuro recibe menos tiempo de desarrollo que el actual. Para esta parte se asignan las siguientes tareas:

- **Crear máquina virtual:** crear y configurar una máquina virtual que funcione como máquina de desarrollo. Esta máquina debe poseer un entorno LAMP (Linux, Apache, MySQL y PHP), para ello el sistema operativo debe ser Ubuntu Server. También se debe configurar la conectividad con la máquina anfitriona, instalar el framework de desarrollo Laravel y montar un servidor FTP.
- **Creación de un nuevo esquema de base de datos:** se requiere un nuevo esquema de base de datos para almacenar toda la información de la anterior base de datos de Sidbrint, para ello se debe pensar en un nuevo esquema priorizando la organización de los datos en tablas.
- **Script de migración:** se requiere de un script capaz de conectarse a la base de datos de Drupal y a la nueva base de datos para traspasar la información de Sidbrint, teniendo en cuenta el nuevo esquema. Los datos de conexión a las bases de datos han de ser externas al script para su manipulación.

3.4. Reuniones

Para hacer el seguimiento del proyecto y resolver dudas sobre los cambios deseados en la plataforma, se realiza una serie de reuniones con los responsables de Sidbrint, con la empresa que gestiona los servidores que contienen la plataforma y la empresa que mantiene la plataforma.

Estas reuniones se dividen en dos partes: la primera parte trata sobre las peticiones para resolver los problemas de la plataforma actual y la segunda parte trata de planificar un nuevo entorno para mostrar los datos usando tecnologías mas modernas.

Los puntos claves discutidos para la primera parte son:

Mejorar y simplificar el aspecto del portal, con la idea de que éste sea más intuitivo de cara a los usuarios. Para ello, se decide realizar los siguientes cambios:

- **Eliminación del acceso de usuario:** muchos usuarios que entran por primera vez en el portal creen que han de registrarse en él para acceder al contenido, ya que el botón de acceso de usuario es uno de los primeros elementos que se visualizan.
- **Destacar el buscador:** la principal función del portal es la de realizar búsquedas. Para hacer entender al usuario el objetivo de la web, se desea una barra de búsqueda más grande y que destaque sobre los otros elementos del portal.
- **Eliminación del control deslizante:** el cliente desea eliminarlo a causa de que no se quiere en el nuevo diseño y ocupa bastante espacio.
- **Edición del menú principal:** El portal cuenta con una barra de búsqueda para realizar las búsquedas, por lo que se decide eliminar la opción *Buscador* del menú.
- **Eliminación de una barra en la página de resultados:** al haber dos barras de búsqueda en la página de resultados, se puede malinterpretar y entender que existen dos métodos de búsqueda, por lo que se ha de eliminar una barra.
- **Arreglar el diseño del listado de resultados:** el portal requiere el arreglo de varios errores visuales, la mejora del listado y la posibilidad de poder elegir el número de resultados a mostrar por pantalla.
- **Creación de pestañas en la ficha de un brigadista:** la página aparece saturada de información. Se requiere implementar un sistema de dos pestañas para repartir la información. La información general se mostrará en la primera pestaña y las referencias de documentos en la segunda pestaña.
- **Eliminación migas de pan:** el cliente desea eliminar las migas de pan ya que no son necesarias. Estas llegan a tres páginas de profundidad, en la página

de resultados las migas son erróneas y en la ficha de fuentes documentales aparece el título completo tres veces: una en las migas de pan.

- **Menú central de cajas con enlaces:** el cliente pide una cuadrícula de enlaces a brigadistas, fuentes documentales, encuadramientos militares y otros contenidos en la cual pueda seleccionar que cuadros serán visibles y cuales se ocultarán desde las preferencias de Drupal. Esta tarea será substituida por la siguientes tareas en la última reunión realizada.
- **Mostrar últimas noticias:** la página principal debe mostrar las n últimas noticias, donde n debe ser un número entre 3 y 10. Éste apartado permite tener contenido actualizado y dinámico en la portada, facilitando el posicionamiento SEO en navegadores como Google.
- **Adición de una descripción del portal:** los usuarios que entran por primera vez al portal pueden desconocer el objetivo del portal. Para solventar éste problema se añade una breve descripción del portal y su objetivo en la portada. Éste texto ha de poder editarse desde la configuración de Drupal.
- **Filtros en la barra de búsqueda:** el cliente pide tener un filtro de contenido en la barra de búsqueda, con el objetivo de mostrar al usuario que tipo de datos puede encontrar en el buscador. Para ello, se añaden tres botones de radio (radio buttons) debajo de la barra: brigadistas, fuentes documentales y encuadramientos militares.

En cuanto a la segunda parte del proyecto se plantean los siguientes aspectos:

- **Roles y acciones:** se concreta que acciones puede realizar cada tipo de usuario. Se mantienen los roles y sus funcionalidades de la versión original.
- **Base de datos:** se ha de rediseñar el esquema de la base de datos. Esta base de datos no contará con tablas exclusivas de Drupal, por lo que no se almacenarán datos cacheados de Drupal, bloques, vistas, registros, etc, los cuales no tienen ninguna utilidad en la nueva base de datos. De esta manera se facilitará el mantenimiento de la base de datos, será más ligera en cuanto a cantidad de datos y será más fácil de consultar.

La información detallada sobre las reuniones se encuentra en el Anexo I de este documento.

4. Sistema actual

En este capítulo se detallan las tecnologías usadas en el sistema actual y otras que se añadirán para mejorar la visualización de éste. El sistema actual, está montado con un sistema de gestión de contenidos, Drupal. Para hacer más fácil la maquetación y la adaptación a otras pantallas, se cambiará el tema actual de Drupal por un tema que incluye Bootstrap 3.

Éste sistema corre bajo una arquitectura LAMP (Linux, Apache, MySQL, PHP). El sistema operativo Linux usa la distribución Ubuntu Server 16.04.3 LTS (Long Term Support), que tiene soporte garantizado hasta Abril del 2021.

4.1. Drupal

Drupal es un sistema de gestión de contenidos o CMS³ libre, modular, multi-propósito y muy configurable, con el que se puede publicar artículos, imágenes, archivos y permite la administración de usuarios.

Drupal también es un sistema dinámico: en lugar de almacenar los contenidos en archivos estáticos en el sistema de archivos del servidor de forma fija, el contenido de las páginas y configuraciones son almacenados en una base de datos y se editan a través de un entorno web. Este contenido se define como nodos, donde estos pueden representar las páginas, posts de un blog, o nuevos objetos, que son almacenados de igual manera.

Este sistema esta diseñado para ser una solución CMS para usuarios no técnicos que necesitan simplicidad y flexibilidad. Por ejemplo, a diferencia de otros CMS, Drupal permite la creación de páginas o plugins a través de Vistas, por lo que no hace falta saber programar, aunque requiere de un aprendizaje del sistema por parte del usuario.

Para usar Drupal, se debe entender los conceptos generales que forman parte de él.

4.1.1. Nodo

Un *nodo* es un término genérico para cualquier pieza de contenido del sitio web. El *tipo de contenido* del nodo indicará que tipo de información mostrará el nodo. Por ejemplo, una *página básica* tiene mínimo dos campos: título y cuerpo. Otro ejemplo sería un producto, que tendría campos como nombre, precio, unidades o modelo).

En el caso de Sidbrint, las fuentes documentales, los brigadistas, los encuadramientos militares y las noticias son nodos.

³Content Management System

4.1.2. Tipos de entidad

Un *tipo de entidad* es una abstracción para agrupar campos. Los tipos de entidades son usados para mostrar datos, ya sean nodos, comentarios, taxonomías, términos, perfiles de usuario o cualquier cosa desarrollada.

4.1.3. Comentario

Los comentarios son otro tipo de contenido para un sitio desarrollado en Drupal. Cada comentario es una pequeña pieza de contenido que un usuario envía, ligado a un nodo en particular.

4.1.4. Taxonomía

Drupal tienen un sistema de clasificación de contenido conocido como *taxonomía*. Se pueden definir vocabularios (grupos de términos de taxonomía) y añadir términos a cada vocabulario. Cada vocabulario puede ser unido a uno o más tipos de contenidos y, de esta manera, los nodos pueden ser agrupados en categorías, etiquetados o clasificados.

4.1.5. Usuario

Un usuario es un tipo de entidad que representa a un usuario del sitio web del mundo real. Por defecto un usuario tiene un conjunto de propiedades incluyendo su nombre de usuario, su email, su contraseña para acceder al sitio y su rol. Pueden haber otras propiedades proveídas por otros módulos y pueden extenderse a nuevos campos. Un ejemplo de una propiedad proveída por otro módulo es un enlace a una cuenta de un perfil del usuario en una red social.

4.1.6. Módulo

Un módulo es un código de software que extiende la funcionalidad de Drupal. Los módulos pueden pertenecer a tres categorías:

- **Core:** son aquellos módulos incluidos en la descarga de Drupal. Pueden ser habilitados o deshabilitados sin la necesidad de descargar componentes adicionales.
- **Contributed:** estos módulos son descargados en la sección Módulos de la página de Drupal e instalados junto a la instalación de Drupal.
- **Custom:** módulos creados por el dueño del portal de Drupal. Estos requieren un completo conocimiento de Drupal, el lenguaje de programación PHP y la API de Drupal.

Uno ejemplo de módulo es el módulo *Devel*. Éste módulo muestra las variables disponibles en cada página del portal y además muestra las consultas realizadas a la base de datos que se realizan cuando se procesa una petición a una url del sitio web.

Con los módulos, es posible crear nuevos bloques, usando funciones de la API de módulos:

- **hook_block_info:** se define la información del bloque.
- **hook_block_view:** se establece que título tendrá el bloque, el diseño de este y que archivos CSS y Javascript tiene enlazados.
- **hook_block_configure:** permite la adición de nuevos campos en el formulario de configuración del bloque.
- **hook_block_save:** permite guardar las configuraciones adicionales en la base de datos.

4.1.7. Regiones y bloques

Las páginas pertenecientes a un sitio desarrollado con Drupal son presentados con *regiones*. Estas pueden incluir la cabecera, contenido principal, barras laterales, pies de página... Un tema puede definir regiones adicionales. En la figura 4 se muestra un modelo de una página de Drupal, incluyendo las principales regiones.

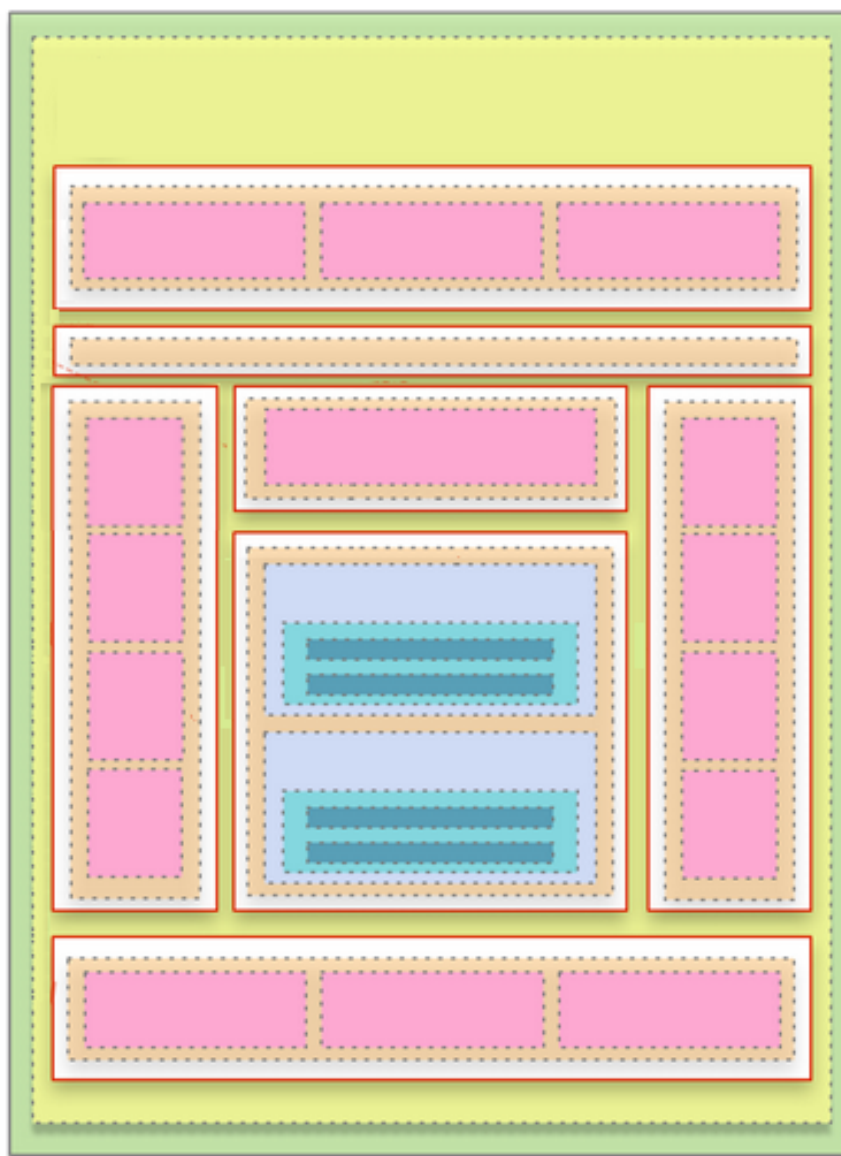


Figura 4: Diseño de una página en Drupal.

Los *bloques* son piezas de información mostradas en las regiones de la página web. Estos pueden ser piezas de HTML o texto, menús, la salida de módulos (por ejemplo temas del foro) o listas dinámicas (por ejemplo últimas noticias).

La administración de Drupal dispone de un apartado *Bloques* en el cual se asignan los bloques que se quieren mostrar a las distintas regiones del diseño. Dentro de la configuración de cada bloque se asigna las páginas donde debe ser mostrado, mediante el uso de rutas. Esta configuración también se puede realizar a través de código fuente en los módulos, pero no se recomienda su uso, dado que estas preferencias se programan en un script de PHP, por lo cual quedan ocultas para el administrador de Drupal y tienen prioridad sobre la configuración hecha a través de la administración.

BLOQUE	REGIÓN	OPERACIONES
Navegación		
+ Alternador de idioma (Texto de la interfaz del usuario)	Navegación ▼	configurar
+ Menú principal	Navegación ▼	configurar
Top Bar		
+ Filter Search Form	Top Bar ▼	configurar
Destacado		
No hay bloques en esta región		
Ayuda		
No hay bloques en esta región		
Contenido		
+ Custom pager	Contenido ▼	configurar
+ Sidbriint About	Contenido ▼	configurar
+ Vista: Recent blog entries Block	Contenido ▼	configurar

Figura 5: Ejemplo de la configuración de bloques.

4.1.8. Menús

Existen cuatro tipos de menús en Drupal 7:

- El menú **principal** es construido por los administradores del sitio y es mostrado automáticamente en la cabecera de la página en muchos temas.
- El menú de **administración** permite acceder a los diferentes apartados de la administración del sitio web.
- El menú de **navegación** es un menú general que contiene enlaces proveídos por los módulos del sitio.
- El menú de **usuario** contiene enlaces a la cuenta de usuario y el enlace de cerrar sesión.

Se pueden crear menús personalizados y mostrarlos, habilitando sus bloques correspondientes.

4.1.9. Temas

Los temas controlan la apariencia del sitio web, como el sitio es mostrado, incluyendo el aspecto gráfico, el diseño y los colores. Un tema consiste en uno o varios archivos de plantillas PHP que definen la salida de HTML, junto uno o mas archivos CSS que definen los diseños, fuentes, colores y otros estilos.

4.1.10. Vistas

Las vistas permiten presentar los nodos o otras entidades como páginas, bloques, RSS feeds u otros formatos. El principal objetivo de las vistas es crear listas de contenido dinámicas, por ejemplo una lista de últimas noticias, en este caso ordenadas por fecha de publicación.

4.1.11. Diseño de vistas

Las vistas pueden ser creadas como páginas o bloques. Se puede cambiar el diseño de los datos mostrados creando plantillas personalizadas. Las plantillas pueden ser creadas para cambiar el aspecto y el estilo de la salida.

4.1.12. Base de datos

Drupal almacena la información en una base de datos. Dentro de la base de datos, cada tipo de información tiene su propia tabla. Por ejemplo la información básica de los nodos se almacena en la tabla *Node* y cada campo guarda su información en una tabla separada, creada automáticamente por Drupal.

La base de datos más común para Drupal es MySQL.

4.1.13. Ruta

Al visitar una URL del sitio de Drupal, la parte de la URL después de la dirección del sitio base se conoce como ruta.

Al visitar una ruta en Drupal, este descubre que información ha de enviar al navegador comprobando su lista de ítems de menú y rutas.

Generalmente Drupal permite a cada módulo definir rutas de las cuales el módulo será responsable y, cuando se visite una ruta en particular, Drupal preguntará al módulo si debe mostrarse o no en la página.

4.1.14. Bootstrap

Bootstrap es la Unidad Central de Procesamiento (CPU) de Drupal. Esta espera a una solicitud de ruta, cuando llega empieza a procesar la solicitud.

4.1.15. Permisos

Los permisos son usados para controlar que pueden visualizar los usuarios y que áreas particulares de la página pueden editar. Estos permisos pertenecen a usuarios registrados (administradores, editores de contenido, miembros del sitio) y usuarios no registrados. Los permisos pueden configurarse para ser muy específicos.

4.1.16. Caché

Drupal guarda mucha información de bloques, páginas, configuraciones... en la base de datos. Para evitar el uso de muchas consultas a base de datos, Drupal hace uso de la caché, para servir páginas guardadas en ella y evitar el ralentizamiento del sistema.

4.2. Bootstrap framework

Bootstrap es un framework de código abierto para diseñar sitios y aplicaciones web con HTML, CSS y el lenguaje Javascript. Este framework contiene plantillas de diseño con tipografía, formularios, botones, cuadros, menús de navegación y otros elementos de diseño web para el desarrollo *front-end*.

Éste framework utiliza un sistema de cuadrilla que utiliza una serie de contenedores, filas y columnas para diseñar y alinear el contenido. La cuadrilla tiene un ancho estándar de 940 píxeles. Alternativamente, el desarrollador puede usar un diseño de ancho-variable.

Para ambos casos, la herramienta tiene cuatro variaciones para hacer uso de distintas resoluciones y tipos de dispositivos: teléfonos móviles, formato de retrato y paisaje, tabletas y computadoras con baja y alta resolución (pantalla ancha). Esto ajusta el ancho de las columnas automáticamente.

4.3. Drush

Drush es una herramienta de consola para sistemas UNIX para hacer de interfaz de Drupal. Este contiene una amplia variedad de comandos para interactuar con los módulos, temas, perfiles de un sistema Drupal.

Entre sus comandos cabe destacar el comando **drush cc all**, que permite eliminar la caché del sitio sin tener que navegar por las páginas de configuración de Drupal y el comando **drush en module_name**, que descarga, instala y habilita los módulos, evitando tener que descargarlo a través de un navegador para luego tener que subirlo al servidor, instalarlo y habilitarlo en la página de módulos de Drupal.

4.4. Base de Datos

La base de datos de Drupal 7 funciona con MySQL 5.7.20. Para esta versión de Drupal, es necesario que el host permita a las cuentas de la base de datos usar los siguientes derechos:

- SELECT: permite seleccionar datos a través de una consulta.
- INSERT: permite insertar datos a través de una consulta.
- UPDATE: permite modificar datos existente a través de una consulta.
- DELETE: permite eliminar datos a través de una consulta.
- CREATE: permite la creación de bases de datos, tablas, índices, vistas...
- DROP: permite eliminar bases de datos tablas, columnas, índices...
- INDEX: permite la creación o eliminación de índices.

- ALTER: permite la modificación de la estructura de una tabla (añadir o eliminar columnas e índices, cambiar el tipo de las columnas, el nombre de las columnas o incluso el nombre de la tabla).

También se requiere que la variable de sistema *max_allowed_packet* tenga un valor al menos de 16M (el valor por defecto de MySQL es de 1MB). Esta variable hace referencia al tamaño del paquete que contiene las consultas. En caso de que la variable no tenga este valor es posible su modificación a través de la consola de MySQL:

```
shell> mysqld --max_allowed_packet=16M
```

Cuando se instala Drupal, se crean 74 tablas dentro de la base de datos. En la figura 6 se muestra el diagrama de entidades con las tablas agrupadas según su objetivo.

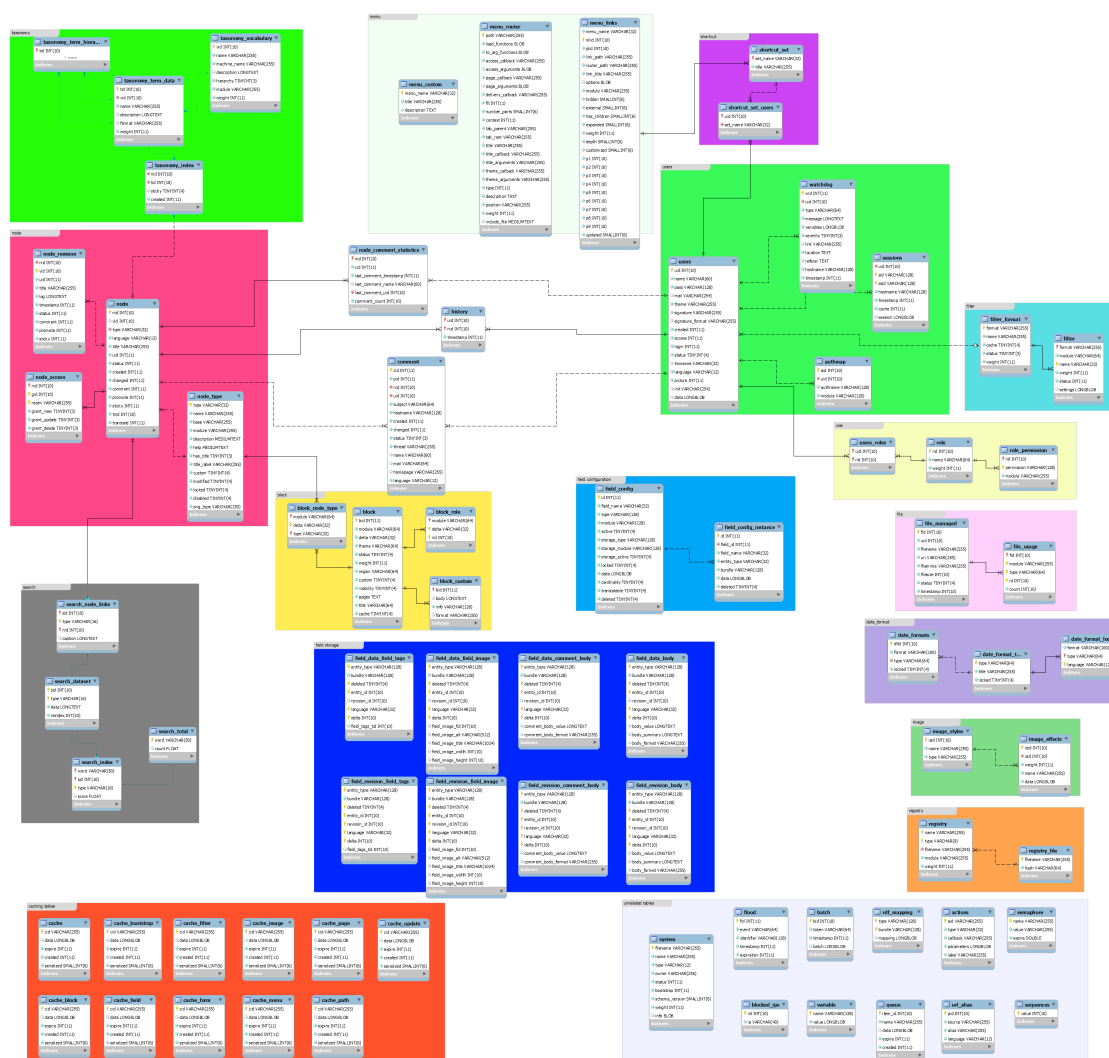


Figura 6: Base de datos de Drupal.

Los grupos de tablas más importantes de la base de datos son:

- Grupo bloc (amarillo): se encuentran las tablas que guardan la información de los bloques de Drupal.
- Grupo cache (salmón): las 11 tablas que forman este grupo son las tablas de caché, que guardan la caché de los bloques, imágenes, menús, páginas...
- Grupo field storage (azul marino): estas tablas guardan los valores y revisiones de los campos que hay disponibles por defecto en una instalación de Drupal. Estos campos pueden ser de comentarios, tags, etc.
- Grupo node (rosa): guarda la información de los nodos.
- Grupo search (gris oscuro): tablas relacionadas con búsquedas. Guardan información que se buscará, información que enlaza a otros nodos, el total de palabras que se usan en las búsquedas...
- Grupo taxonomía (verde lima): estas tablas guardan los datos de taxonomías, términos y vocabulario.

El detalle del esquema de la base de datos de Drupal (figura 6) es presentado en el Anexo II.

Por cada campo dentro de los tipos de contenidos adicionales (en este caso Brigadista, Fuente y Encuadramiento) Drupal crea una tabla de datos y una de revisión, con la consecuencia de si el tipo de contenido brigadista tiene 46 campos, se creará el mismo número de tablas para datos y para revisiones. Esto se debe a que la base de datos de Drupal trabaja con punteros, de forma que la información de un elemento no es guardada en una fila de una sola tabla, sino que está en diferentes tablas, cada una guardando un dato en concreto. Estas tablas tienen un campo que apunta a la tabla *node*, para enlazar la información.

Para la extracción de los datos de un tipo de contenido, Drupal ha de lanzar una consulta con varias uniones (*Joins*), uno por cada campo del contenido, ya que cada campo está en una tabla de manera individual.

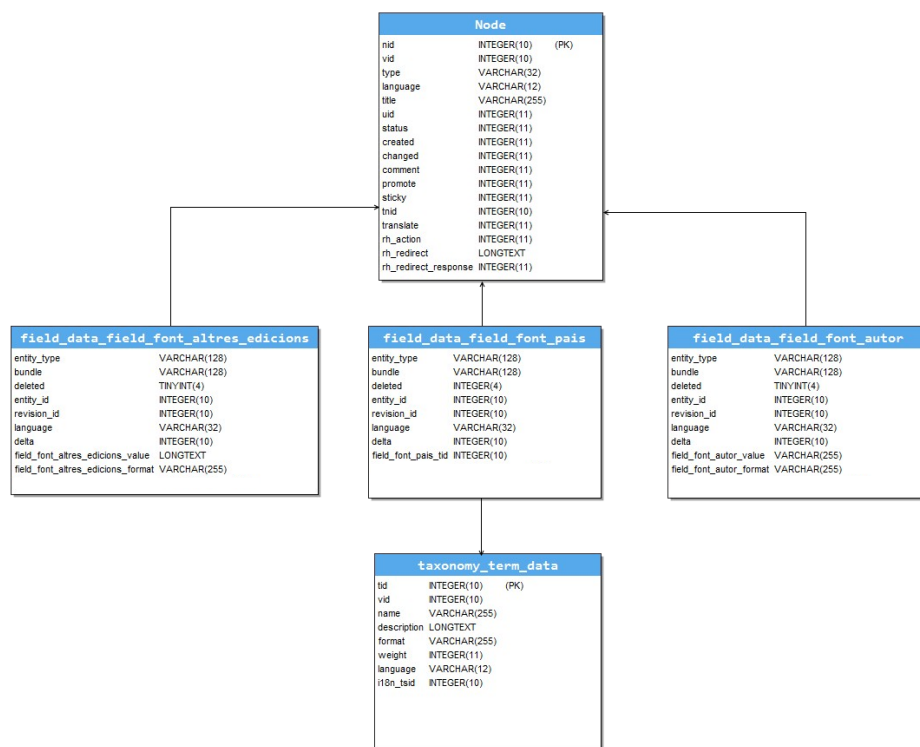


Figura 7: Contenido fuente documental de Drupal.

En la figura 7 se puede observar como sería la obtención de la información de las fuentes documentales. La tabla *Node* contiene el título de las fuentes y es la tabla principal de los contenidos. Las tablas *field_data_field_...* contienen la información que se guarda en los diferentes campos del formulario que sirve para crear fuentes, siempre que los campos sean de texto o botones radiales (*radio button*). En caso de que los campos sean desplegable o de múltiples opciones, en la tabla no se guardará el valor, sino un puntero a la tabla *taxonomy_term_data* donde se guardan las taxonomías como los idiomas o países de las fuentes.

4.5. Cambios realizados

Los cambios son realizados a través del administrador de Drupal y a través de bloques (tanto hechos con el módulo Vistas como hechos por módulos nuevos a través de scripts de PHP). Esta decisión se toma siguiendo los criterios de buenas prácticas de Drupal, ya que al ser un gestor de contenidos, ha de ser personalizable para los usuarios y administradores del sitio.

También se corrigen los errores de programación que se han encontrado en la página. Estos errores dificultan el buen uso de Drupal, evitando el correcto funcionamiento de la configuración del administrador.

4.5.1. Creación de un subtema

A partir del tema basado en Bootstrap, se crea un subtema `Sidbrint_bootstrap`. El tema original contiene unas plantillas para la creación de subtemas, donde se puede escoger una plantilla según que tecnología de estilos se prefiera usar (CSS puro, LESS o SASS).

El subtema escogido para el proyecto es LESS, ya que facilita la estructuración de la hoja de estilos, permite crear variables dentro de la hoja de estilos y debe ser compilada para usarla, por lo cual facilita la detección de errores en los estilos.

El uso de Bootstrap facilita la maquetación de la web y su visualización en diferentes tamaños de pantalla, agilizando el trabajo de la construcción de la nueva interfaz y sin haber de invertir tiempo en profundizar en tecnologías de maquetación HTML y CSS.

4.5.2. Configuración de bloques

Los bloques del portal se muestran independientemente de su configuración en la administración de regiones y bloques. Este hecho niega la posibilidad de poder administrar el diseño del portal para un usuario administrador y viola las buenas prácticas de Drupal.

Este hecho se debe a que la configuración de los bloques se encuentra junto al código fuente de creación de estos en su módulo correspondiente. Con la eliminación del fragmento de código fuente que contiene la configuración se consigue que los bloques obedezcan a la configuración del administrador de Drupal.

Finalmente, toda la configuración de los bloques se lleva a través de la administración de Drupal, siguiendo las buenas prácticas y solucionando los errores mencionados anteriormente.

4.5.3. Creación de bloques

Para la creación de los bloques se han añadido nuevos módulos creados a partir de código fuente. Un módulo consiste de mínimo dos archivos:

- ***.info:** define el nombre, la descripción, la categoría del módulo y a que versión de Drupal va dirigido. En caso de usar archivos Javascript o CSS se puede indicar su ubicación para ser usados por el módulo.
- ***.module** define el funcionamiento del módulo y, en caso de crear uno o más bloques, define el diseño de estos, su comportamiento y su configuración a través de la API de módulos de Drupal y del lenguaje de programación PHP.

También se ha hecho uso del módulo Vistas para crear el bloque *Últimas noticias*, dado que esta opción es una manera fácil y rápida de crear bloques para listados de nodos.

4.6. Resultados

En éste apartado se analizarán los resultados obtenidos a través de los cambios especificados en el apartado anterior. Por cada sección, se mostrará la imagen original de una página, una breve descripción de los cambios y finalmente la imagen que refleja los cambios en la página.

4.6.1. Página principal



Figura 8: Diseño inicial de la página principal.

Al principio se había desarrollado un diseño de una cuadrícula de enlaces configurable, de manera que un usuario administrador pudiera escoger los cuadros que se muestran en la cuadrícula. Pero el diseño no convenció por la falta de elementos como una breve definición de los objetivos de Sidbrint o la muestra de noticias recientes en la portada, para tener una página principal más dinámica. Esto se puede apreciar en la figura 9.



Figura 9: Diseño de cuadrícula de la página principal.

En la figura 10 se destaca la barra de búsqueda y se destaca los diferentes tipos de datos que se pueden encontrar. Se aprecian las últimas noticias, mostrando el título, la fecha de publicación y una imagen de esta. Cabe destacar que también aparece la breve descripción de los objetivos del sitio, siendo esta editable en el administrador de Drupal y con formato HTML.

Con los cambios realizados, es posible conocer cual es la misión de Sidbrint, que tipo de información se puede encontrar y se puede comprobar que el sitio se va actualizando, a través de las últimas noticias que se muestran.

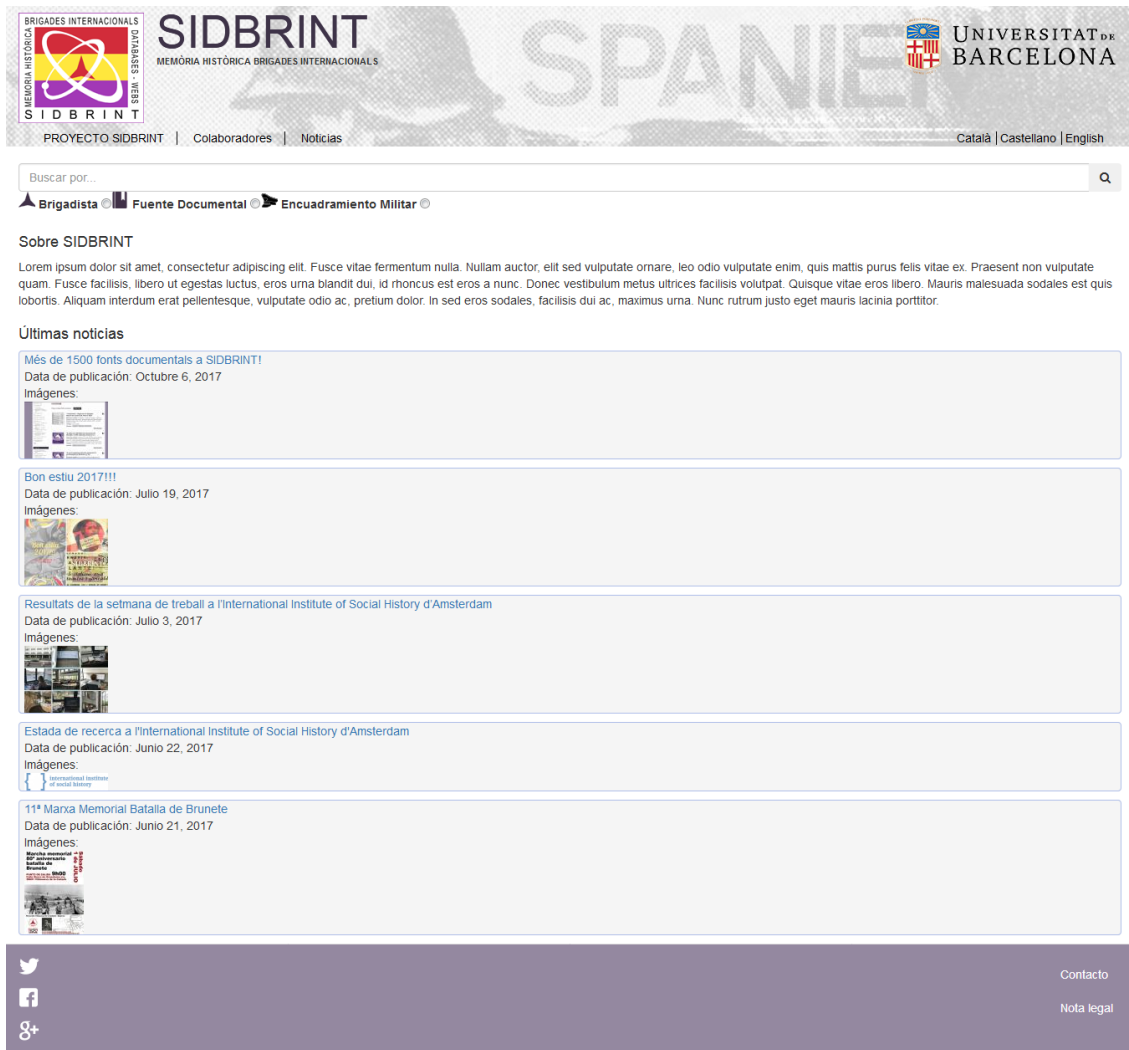


Figura 10: Diseño final de la página principal.

4.6.2. Página resultados

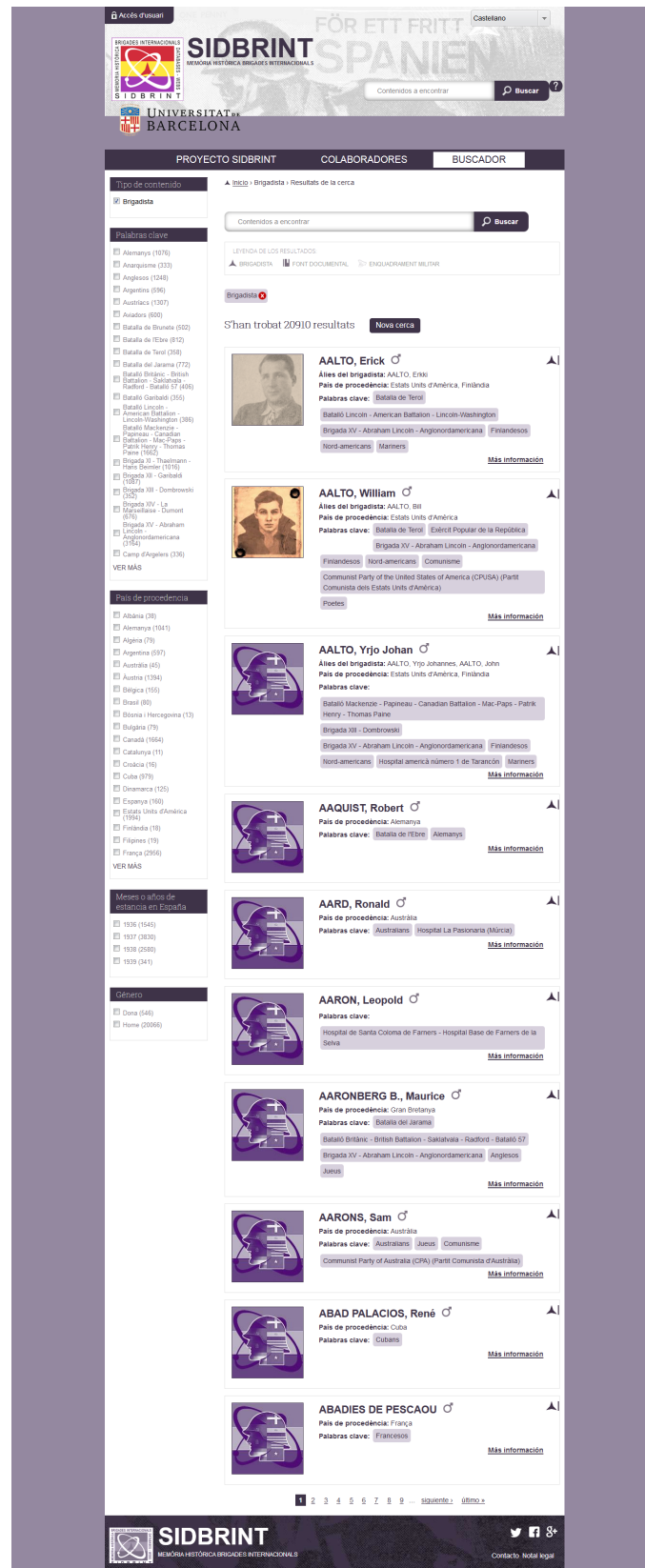


Figura 11: Diseño inicial de la página de resultados.

En la página de resultados se mejora la visibilidad de las cajas de resultado, quitando las palabras clave de ellas. El nuevo tema aprovecha toda la longitud horizontal, a consecuencia las cajas tienen mas espacio horizontal y los títulos de las fuentes documentales tienen más espacio y no se cortan.

Se da la posibilidad al usuario de poder elegir el número de resultados que se mostrarán por cada página.

Con estos cambios, se pueden visualizar mas resultados por pantalla, con poca información, facilitando la visualización de la página.

Los cambios son reflejados en la figura 12.

- 10 items
- 20 items
- 30 items



Tipologia: Llibre o opuscle



Tipologia: Documentació d'arxiu (correspondència, carnets, fulls volants)



Tipologia: Documentació d'arxiu (correspondència, carnets, fulls volants)



Tipologia: Documentació d'arxiu (correspondència, carnets, fulls volants)



Tipologia: Documentació d'arxiu (correspondència, carnets, fulls volants)



Tipologia: Article de publicació periòdica (diari o revista)



Tipologia: Article de publicació periòdica (diari o revista)



Tipologia: Article de publicació periòdica (diari o revista)



Tipologia: llibre o opuscle



Tipologia: Article de publicació periòdica (diari o revista)

27

4.6.3. Página ficha brigadista

The screenshot shows the initial design of the brigadista profile page on the SIDBRINT website. The header includes the SIDBRINT logo, the text 'MEMORIA HISTÓRICA BRIGADES INTERNACIONALES', and a search bar. The main navigation bar has three tabs: 'PROYECTO SIDBRINT', 'COLABORADORES', and 'BUSCADOR'. The profile page for William Aalto is displayed, featuring a portrait, a list of keywords (Palabras clave), and a list of references (Bibliografía). The keywords include 'Batalla de Terol', 'Ejército Popular de la República', 'Brigada XV - Abraham Lincoln - Anglonordamericana', 'Finlandeses', 'Nord-americanos', 'Comunismo', 'Communist Party of the United States of America (CPUSA) (Partit Comunista dels Estats Units d'Amèrica)', and 'Poetes'. The references list works by Castell, Carrol, Celada, and Centro Documental de la Memoria Histórica. The page also includes social media icons and a footer with the SIDBRINT logo and contact information.

Figura 13: Diseño inicial de la página ficha brigadista.

Para simplificar la visualización de los datos de los brigadistas, de manera que no se muestre una gran densidad de texto, se han creado dos pestañas de información: la primera muestra los datos básicos de un brigadista y la segunda muestra sus referencias bibliográficas. el resultado se puede apreciar en la figura 14.



Palabras clave:

Batalla de Terol Batalló Lincoln - American Battalion - Lincoln-Washington Brigada XV - Abraham Lincoln - Anglonordamericana

Finlandesos Nord-americans Mariners

Àlies del brigadista:

AALTO, Erkki

País de procedència:

Estats Units d'Amèrica

Finlàndia

Lloc de naixement d'aquest brigadista:

Finlàndia

Data de naixement d'aquest brigadista:

1913

Cos de l'exèrcit:

Terra

Brigada:

Brigada XV - Abraham Lincoln - Anglonordamericana

Batallo:

Batailló Lincoln - American Battalion - Lincoln-Washington

Anys/mesos de la seva estada a Espanya:

1936

1937

1938

1939

Més i

Arribà a bord de

Ferit a la cama a Terol

No va poder tornar als

Mor durant la Segona Guerra Mundial

Access URL:

Access URI:
<http://193.169>

<http://192.168.56.101/es/node/24906>

29

4.6.4. Página ficha fuente documental

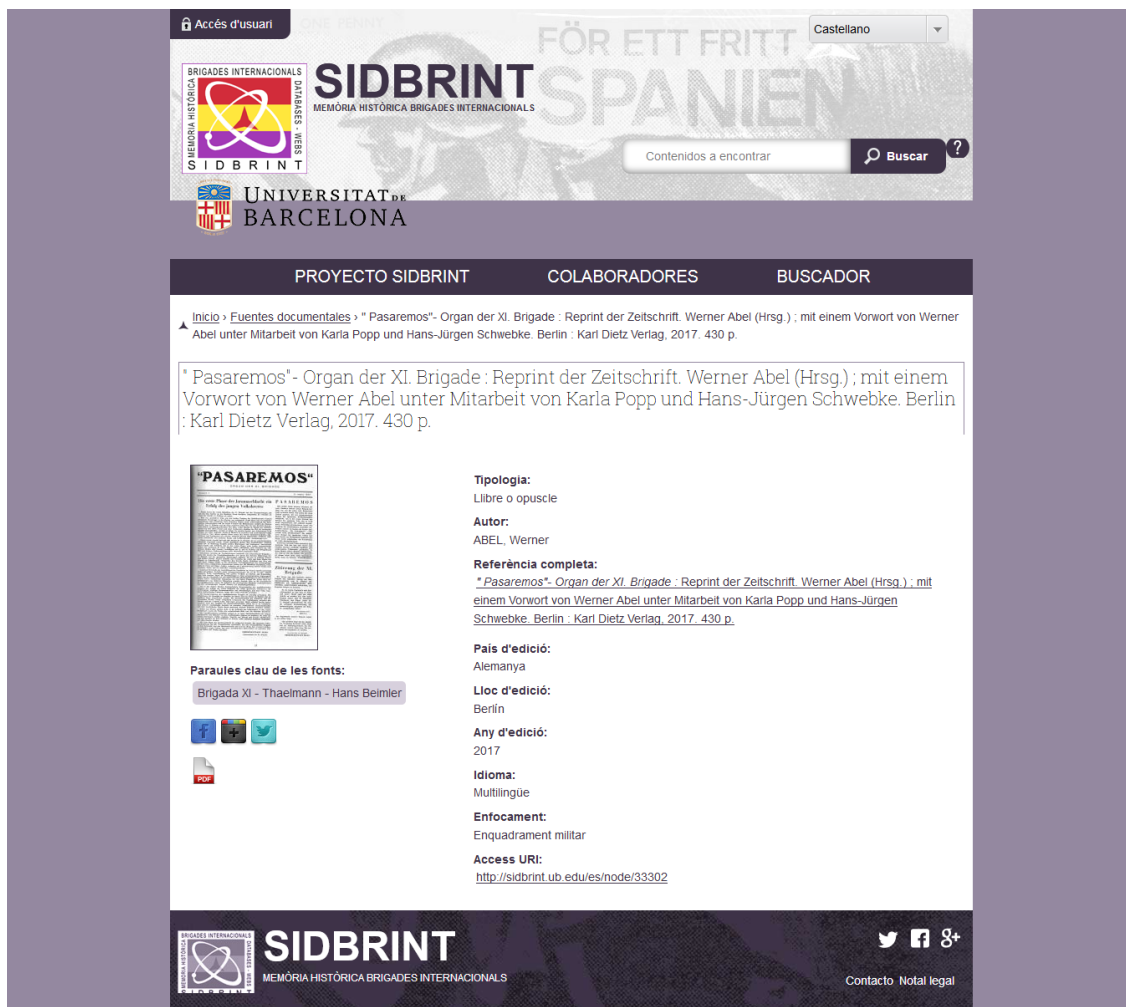


Figura 15: Diseño inicial de la página ficha fuente documental.

Con el fin de eliminar información duplicada (en este caso el título), se decide quitar las migas de pan de la página, que solo mostraban tres niveles de profundidad:

1. Index: página principal.
2. Búsqueda: página de resultados.
3. Título de la fuente documental de la ficha actual.

"Pasaremos"- Organ der XI. Brigade : Reprint der Zeitschrift. Werner Abel (Hrsg.) ; mit einem Vorwort von Werner Abel unter Mitarbeit von Karla Popp und Hans-Jürgen Schwebke. Berlin : Karl Dietz Verlag, 2017. 430 p.



Paraules clau de les fonts:

Brigada XI - Thaelmann - Hans Beimier



Enquadrament militar

Access URI:

<http://192.168.56.101/es/node/33302>

Tipologia:

Libre o opuscle

Autor:

ABEL, Werner

Referència completa:

"Pasaremos"- Organ der XI. Brigade : Reprint der Zeitschrift. Werner Abel (Hrsg.) ; mit einem Vorwort von Werner Abel unter Mitarbeit von Karla Popp und Hans-Jürgen Schwebke. Berlin : Karl Dietz Verlag, 2017. 430 p.

País d'edició:

Alemanya

Lloc d'edició:

Berlin

Any d'edició:

2017

Idioma:

Multilingüe

Enfocament:



[Contacto](#)

[Notat legal](#)

Figura 16: Diseño final de la ficha de fuente documental.

5. Sistema futuro

En este capítulo se procede a detallar los requisitos del futuro sistema, las tecnologías que usará, el nuevo diseño de la base de datos y la seguridad en la aplicación.

Cabe destacar que el desarrollo de esta parte ha estado breve, en ese poco tiempo ha sido posible montar la máquina virtual, configurar el entorno de desarrollo y crear la nueva base de datos para Laravel, migrando y adaptando los datos de Drupal, a través de un script escrito en PHP. Sin embargo se espera que esta primera planificación pueda servir de ayuda a futuros desarrolladores del portal.

5.1. Tecnologías escogidas

5.1.1. Servidor LAMP - Linux / Apache / Mysql / PHP

Se ha construido el entorno del sistema futuro bajo una arquitectura LAMP (Linux, Apache, Mysql y PHP). Para ello, se ha elegido la distribución Ubuntu Server 16.04 LTS, dado que esta tiene soporte garantizado hasta Abril de 2021, asegurando así un amplio de ciclo de vida de los servicios de los que dependerá el portal.

El 27 de abril de 2018 se publica una nueva versión de Ubuntu Server (18.04 LTS), como la máquina virtual ya está configurada, se mantendrá la versión actual de Ubuntu hasta junio, donde se publicará una actualización que cambiará la versión actual por la nueva.

Los servicios básicos instalados son:

- Apache 2.4.18
- Mysql 5.7.21
- PHP 7.0.25

Para la implementación del servidor, se ha usado una máquina virtual de Oracle VM VirtualBox, simulando así un entorno donde el programador desarrolla con su máquina y el sistema está aislado en otra máquina que hace de servidor. El intercambio de archivos se hace a través del protocolo FTP (File Transfer Protocol) y de Filezilla: un entorno gráfico que facilita el intercambio de archivos.

5.1.2. Laravel

Laravel es un framework de código abierto para desarrollar aplicaciones y servicios web con PHP 5 y PHP 7, siguiendo el patrón de arquitectura Modelo-Vista-Controlador y basado en Symfony. Su filosofía es el desarrollo de código PHP elegante, simple y fácil de leer, permitiendo una multitud de funcionalidades.

Éste framework dispone de varias herramientas para facilitar el desarrollo:

- **Composer:** herramienta de gestión de dependencias para PHP. Permite declarar que librerías depende un proyecto. A diferencia de otros gestores de dependencias, Composer gestiona las librerías por proyectos, instalándolas en un directorio dentro del proyecto (*vendor* por defecto).
- **Blade:** sistema de plantillas para crear vistas en Laravel. Permite extender plantillas creadas y secciones en otras vistas en las que también se tienen accesibles las variables. A diferencia de otros sistemas de plantillas Blade permite el uso de código PHP en las vistas. De hecho, todas las vistas de Blade son compiladas a código PHP y cacheadas hasta que no se modifiquen.
- **Eloquent:** es el ORM⁴ Cada tabla de la base de datos tiene su modelo correspondiente, el cual se utiliza para interactuar con la tabla. Los modelos permiten la obtención de los datos de las tablas o insertar nuevos datos en la tabla, entre otras operaciones CRUD (Create, Read, Update y Delete).
- **Routing:** Las rutas de Laravel se definen en el archivo de rutas, localizado en el directorio *routes*. Estos archivos son cargados automáticamente por el framework. Cabe destacar dos archivos importantes:
 - *routes/web.php*: en este archivo se definen las rutas que son para la interfaz web (generalmente se usan para cargar las vistas creadas con Blade). Estas rutas están asignadas al grupo middleware⁵ web que proporciona características como estado de sesiones y protección CSRF (Cross-site request forgery).
 - *routes/api.php*: en este archivo se definen las rutas que forman parte de una API. Estas rutas no usan estados y están asignadas al grupo middleware api.
- **Artisan:** interfaz de línea de comandos incluida con Laravel que contiene una serie de comandos que asisten al desarrollo de una aplicación. Con estos comandos se pueden crear modelos, migraciones de bases de datos, controladores, etc. Laravel permite crear comandos personalizados.

Para este proyecto, se escoge la versión 5.5 LTS de Laravel, publicada el 30 de Agosto de 2017. Esta versión seguirá recibiendo actualizaciones de corrección de errores durante dos años y recibirá actualizaciones de seguridad durante tres años.

5.2. Análisis de requerimientos

5.2.1. Requisitos

Los propietarios de Sidbrint piden una serie de requisitos que el portal debe cumplir y unas funcionalidades. La funcionalidad principal del sitio es la de poder

⁴Mapeo objeto-relacional o Object-Relational mapping en inglés.

⁵lógica de intercambio de información entre aplicaciones

consultar datos sobre brigadistas de las Brigadas Internacionales de la Guerra Civil Española, fuentes documentales que reflejan datos sobre los brigadistas y de hechos históricos y encuadramientos militares.

Estos datos deben ser introducidos en la plataforma a través de formularios y solo los usuarios que tengan los permisos adecuados podrán añadir, editar y eliminar dichos datos.

5.2.2. Casos de uso

Los usuarios que navegan por la plataforma tienen diferentes objetivos:

- Buscar información a través del buscador y consultar las fichas de los brigadistas, fuentes documentales o encuadramientos militares.
- Escribir nuevas fichas a través de los formularios disponibles en el apartado de edición del sitio.
- Revisar los datos entrados por otros usuarios con perfil de colaboradores.
- Gestionar los usuarios y la configuración de Drupal.

Estos objetivos no pueden ser desempeñados por cualquier usuario, ya que supondría un riesgo para la integridad de los datos (estos podrían ser modificados o eliminados por un usuario que no debería tener este derecho). Por lo tanto se definen unos roles con el fin de limitar las acciones que un usuario puede llevar a cabo, dependiendo del perfil que tenga.

Usuario anónimo

Los usuarios anónimos pueden navegar por los diferentes sitios de la web, para realizar consultas y ver información, pero no les está permitido crear, modificar o eliminar contenido del sitio así como gestionarlo. En caso de querer hacer una de las acciones anteriores, el usuario debe identificarse en el sitio siempre que esté registrado en él. En tal caso podrá acceder a través de una dirección de correo electrónico y de una contraseña que debe conocer.

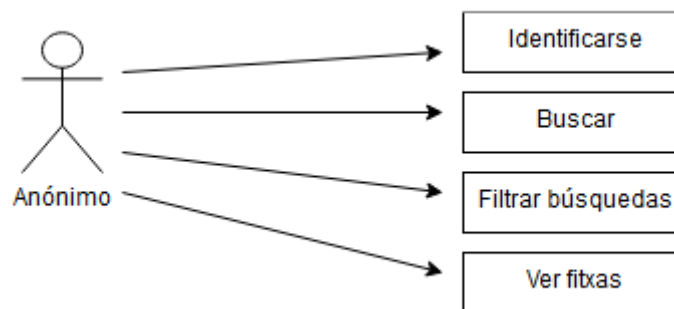


Figura 17: Caso de uso del usuario anónimo

Usuario editor

Los usuarios editores pueden realizar las acciones del usuario anónimo. Cuando estos se identifican en el portal, pueden añadir contenidos (excepto fuentes documentales y noticias) y pueden modificar su contraseña de acceso al sitio. Al estar identificados adquieren la acción de desconectarse (cerrar la sesión).

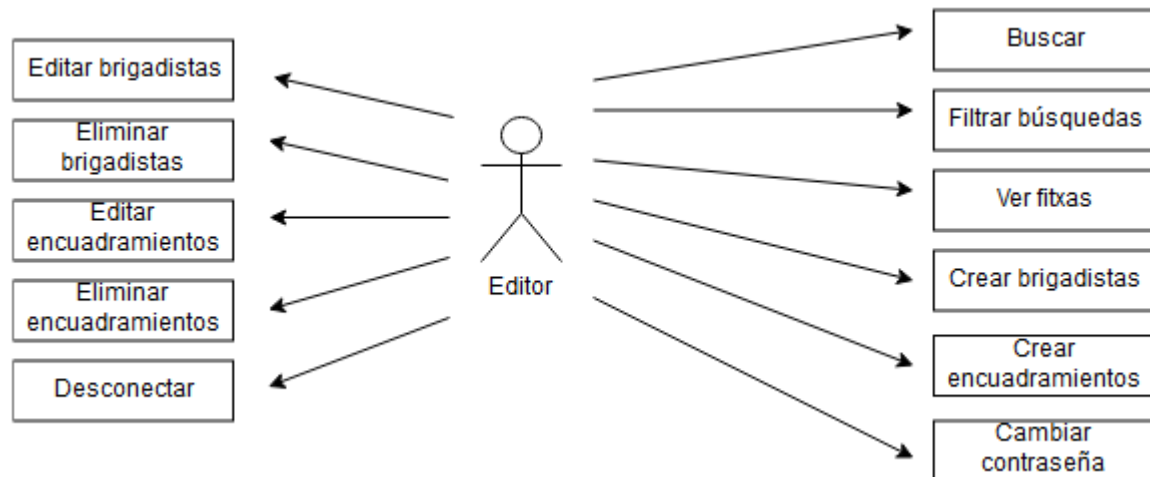


Figura 18: Caso de uso del usuario editor

Usuario editor de fuentes

Este tipo de usuarios poseen las mismas acciones que los usuarios editores, con el añadido de que pueden crear, editar o eliminar fuentes documentales.

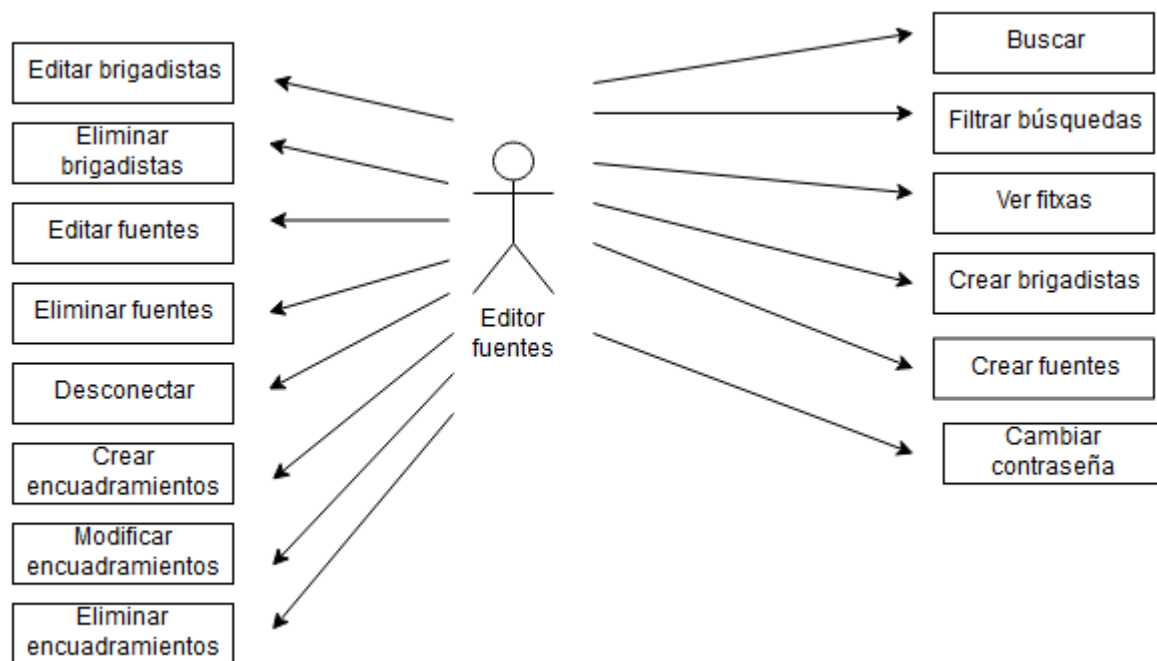


Figura 19: Caso de uso del usuario editor de fuentes

Super-editor

Los usuarios con rol super-editor son trabajadores de Sidbrint. Pueden añadir, editar y eliminar cualquier tipo de contenido y pueden revisar y aprobar el contenido escrito por los usuarios con roles de editor y editor-fuentes.

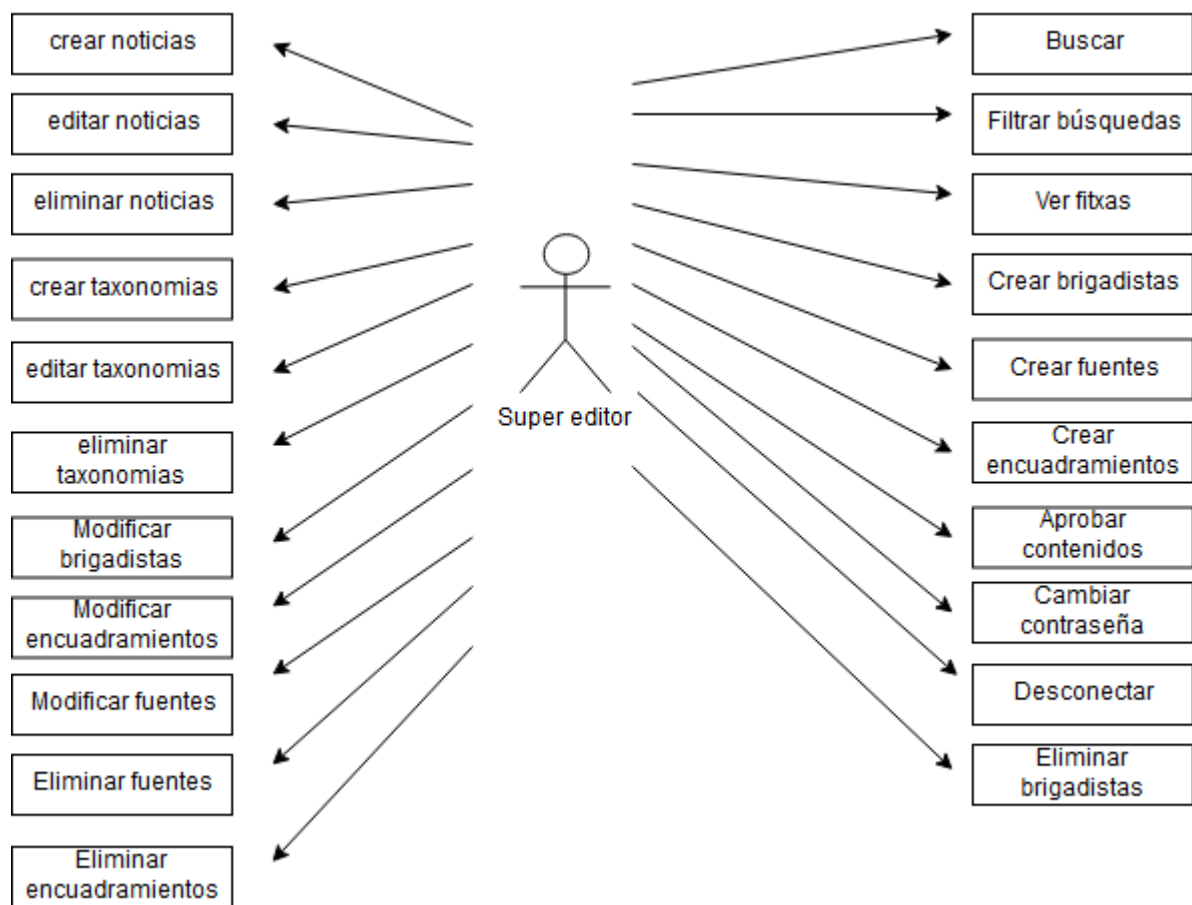


Figura 20: Caso de uso del usuario super-editor

Usuario administrador

En último lugar, el usuario administrador se encarga de administrar el sitio, además de poder crear usuarios y editar su información, como su rol, email o contraseña. También puede añadir, modificar o eliminar contenido del portal.

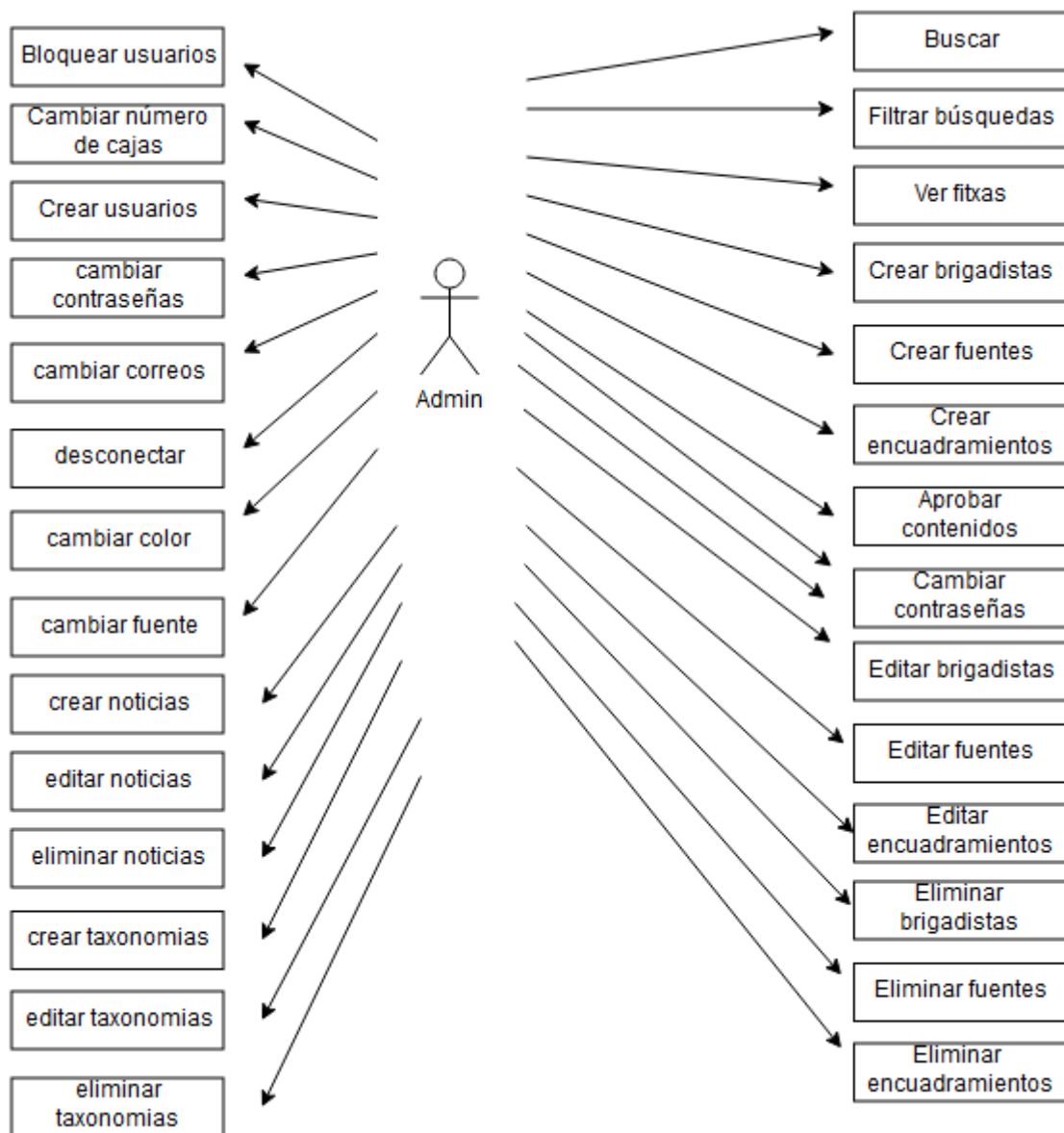


Figura 21: Caso de uso del usuario administrador

En el Anexo II se describen los casos de uso en detalle.

5.3. Diseño de la base de datos

El esquema de la base de datos se ha construido usando las migraciones de Laravel. Por cada migración se define los atributos de una tabla y sus correspondientes tipos de datos.

```
public function up()
{
    Schema::create('users', function (Blueprint $table) {
        $table->increments('id');
```

```

$table->string('nombre');
$table->string('email')->unique();
$table->string('password');
$table->boolean('bloqueado');
$table->integer('rol_id')->unsigned();
$table->dateTime('ultimo_acceso')->nullable();
$table->rememberToken();
$table->timestamps();

$table->foreign('rol_id')->references('id')->on('rols');
});
}

```

Para exportar las migraciones a la base de datos, primero debe definirse a que base de datos ha de conectarse Laravel, configurando la conexión en el archivo *.env*. Seguidamente se ha de ejecutar el comando *php artisan migrate*, creando así las tablas en la base de datos, con sus respectivos campos (siempre que no haya errores).

A continuación se detalla el esquema de la base de datos que usa Laravel (Figura 22).

Éste nuevo diseño gira entorno a los datos de brigadistas, fuentes documentales, encuadramientos y noticias (tablas de color verde). El objetivo del diseño es tener los datos mas centrados en las tablas principales, evitando el uso de uniones. Aún así, aparecen otras tablas, ya que almacenar toda la información en las tablas principales conllevaría la duplicación de datos. Las relaciones con estas tablas son:

- Uno o varios: la tabla de claves primaria sólo contiene un registro que se relaciona con ninguno, uno o varios registros en la tabla relacionada. Un ejemplo seria la relación entre cuerpos y brigadistas, donde un brigadista puede pertenecer a un solo cuerpo (tierra, mar o aire) o a ninguno, pero un cuerpo tiene varios brigadistas.
- Varios a varios: cada registro en ambas tablas puede estar relacionado con varios registros (o con ninguno) en la otra tabla. Estas relaciones requieren una tercera tabla, denominada tabla de enlace o asociación, porque los sistemas relacionales no pueden alojar directamente la relación. Un ejemplo claro es la relación entre idiomas y fuentes documentales: el idioma inglés puede tener varias fuentes documentales y una fuente documental puede estar disponible en más de un idioma.

5.3.1. Tablas Principales

Fuente	Brigadista	Encuadramiento	Noticia
fuente_ID INTEGER(10) (PK)	brigadista_ID INTEGER(10) (PK)	encuadramiento_ID INTEGER(10) (PK)	noticia_ID INTEGER(10) (PK)
autor VARCHAR(191)	nombre VARCHAR(191)	unidad VARCHAR(191)	título VARCHAR(191)
título VARCHAR(255)	imagen_id INTEGER(10)	Bibliografia INTEGER(10)	cuerpo TEXT
imagen_ID INTEGER(10) (FK)	lugar_nacimiento VARCHAR(191)	completo TINYINT(1)	usuario_ID INTEGER(10) (FK)
tipologia_ID INTEGER(10) (FK)	dia_nacimiento INTEGER(11)	publicado TINYINT(1)	eliminado TINYINT(1)
referencia_completa TEXT	mes_nacimiento INTEGER(11)	eliminado TINYINT(1)	publicado TINYINT(1)
lugar_edicion VARCHAR(191)	ano_nacimiento INTEGER(11)	usuario_ID INTEGER(10) (FK)	created_at TIMESTAMP(1)
autor_brigadista TINYINT(1)	lugar_defuncion VARCHAR(191)	created_at TIMESTAMP(1)	updated_at TIMESTAMP(1)
ano_edicion VARCHAR(191)	dia_defuncion INTEGER(11)	updated_at TIMESTAMP(1)	deleted_at TIMESTAMP(1)
otras_ediciones TEXT	mes_defuncion INTEGER(11)	imagen_ID INTEGER(10) (FK)	
completo TINYINT(1)	ano_defuncion INTEGER(11)	designacion_ID INTEGER(10) (FK)	
eliminado TINYINT(1)	genero TINYINT(1)	deleted_at TIMESTAMP(1)	
publicado TINYINT(1)	comentari TEXT		
usuario_ID INTEGER(10) (FK)	mapa VARCHAR(191)		
created_at TIMESTAMP(1)	usuario_ID INTEGER(10) (FK)		
updated_at TIMESTAMP(1)	completo TINYINT(1)		
deleted_at TIMESTAMP(1)	eliminado TINYINT(1)		
	publicado TINYINT(3)		
	created_at TIMESTAMP(1)		
	updated_at TIMESTAMP(1)		
	blindado_ID INTEGER(10) (FK)		
	cuerpo_ID INTEGER(10) (FK)		
	deleted_at TIMESTAMP(1)		

Figura 23: Tablas principales de Sidbrint.

Todas las tablas principales cuentan con los siguientes campos, excepto *Noticia*, que no contiene ni *imagen_id* ni *completo*:

- **imagen_id**: clave foránea que apunta a la fila correspondiente en la tabla imagen, usando las claves primarias. A cada brigadista, fuente o encuadramiento le corresponde ninguna o solo una imagen.
- **completo**: campo que indica si un registro está completo o está pendiente de finalizar.

- **eliminado:** indica si un registro se ha “eliminado”. De esta forma se hace un borrado seguro, con la consecuencia de que no aparece en los listados ni aparece en las búsquedas, pero sigue existiendo en la base de datos.
- **publicado:** este campo permite ocultar o mostrar los registros en los listados y búsquedas. Su funcionalidad es la de no mostrar contenido que deba ser revisado anteriormente. Una vez revisado correctamente se puede cambiar el estado para mostrar el registro pertinente.
- **usuario_ID:** clave foránea que apunta a la fila correspondiente en la tabla usuario, indicando que usuario a creado el registro.
- **temporales:** los campos temporales indican cuando se crea un registro, se modifica o es eliminado de manera segura. Estos campos son *created_at*, *updated_at* y *deleted_at* respectivamente.

5.3.2. tabla usuario

Usuario		
usuario_ID	INTEGER(10)	(PK)
nombre	VARCHAR(191)	
email	VARCHAR(191)	
password	VARCHAR(191)	
rol_ID	INTEGER(10)	(FK)
ultimo_acceso	DATETIME	
remember_token	VARCHAR(100)	
bloqueado	TINYINT(1)	
created_at	TIMESTAMP(1)	
updated_at	TIMESTAMP(1)	

Figura 24: Tabla usuario de Sidbrint.

La tabla usuario contiene los registros de todos los usuarios que pueden acceder a la administración del sitio. Al contener la información de los usuarios, cabe destacar que debe protegerse la información de identificación de los usuarios para evitar el robo de credenciales, con la consecuencia de que un atacante pueda suplantar a un usuario y pueda realizar acciones como borrar información de la base de datos. Por ello Laravel encripta el campo *password* (se explica con más detalle en el apartado Seguridad).

Por defecto, Laravel crea el campo *email* como único, por lo que no pueden existir dos o más usuarios con el mismo email y además añade el campo *remember_token* usado por el framework, para protegerse contra el pirateo de la cookie que contiene la información de “recordarme” al iniciar sesión.

El campo *rol* es una clave foránea que apunta a la tabla *Rol*. De esta manera, a los usuarios se les asigna un rol, otorgándoles una serie de acciones y privilegios según su rol.

Otro campo a tener en cuenta es *bloqueado*. Existen usuarios que son colaboradores temporales: suben información al portal durante un periodo de tiempo y cuando éste finaliza dichos usuarios dejan de subir contenido. Para proteger la información y al mismo tiempo dejar constancia de los autores, existe la posibilidad de bloquear el acceso a los usuarios usando éste campo.

5.3.3. Tablas imagen y archivo

Imagen		
iimagen_ID	INTEGER(10)	(PK)
url	VARCHAR(191)	
alt	VARCHAR(400)	
campo	VARCHAR(191)	
filename	VARCHAR(200)	
title	VARCHAR(400)	
width	INTEGER(10)	
height	INTEGER(10)	

Imagen_noticia		
imagen_ID	INTEGER(10)	(PK)
url	VARCHAR(191)	
alt	VARCHAR(400)	
filename	VARCHAR(200)	
width	INTEGER(10)	
height	INTEGER(10)	
noticia_ID	INTEGER(10)	(FK)

Figura 25: Tablas imagen e imagen_noticia.

Las imágenes se guardan en una carpeta específica del proyecto, en consecuencia en la tabla imágenes se guarda la ruta y el nombre del archivo, entre otros datos. Con la columna *campo*, podemos saber con que tabla se relaciona y, de esta manera, se guardan las imágenes en una tabla y no quedan repartidas en múltiples tablas.

Las imágenes de las noticias tienen su propia tabla, ya que una noticia puede tener mas de una imagen, a diferencia de las tablas principales. En consecuencia la clave foránea de la relación está en la tabla *imagen_noticia*, apuntando a la tabla *noticia*.

Todos los archivos son almacenados en el directorio `storege/app/public`, y son referenciados en la base de datos guardando su ubicación.

Archivo		
archivo_ID	INTEGER(10)	(PK)
nombre	VARCHAR(191)	
url	VARCHAR(191)	
tipo	VARCHAR(191)	
tabla	VARCHAR(191)	
campo_ID	INTEGER(10)	
Eliminado	TINYINT(1)	
created_at	TIMESTAMP(1)	
updated_at	TIMESTAMP(1)	
deleted_at	TIMESTAMP(1)	

Figura 26: Tabla archivo

La tabla *Archivo* funciona igual que la tabla *imagen*: en el campo *tipo* se especifica a que tabla pertenecen y, como un registro principal puede tener más de un archivo relacionado, la tabla *archivo* se encarga de guardar la clave foránea de la relación.

5.3.4. Tabla configuración

Configuracion		
configuracion_ID	INTEGER(10)	(PK)
clave	VARCHAR(191)	
valor	VARCHAR(191)	

Figura 27: Tabla configuración

La tabla configuración actúa como un diccionario: contiene parejas clave-valor, en dónde se busca por la clave para obtener el valor de esta. Por ejemplo se puede crear la pareja para configurar el número de noticias a mostrar en la página principal. El nombre de la clave es “num_noticias_portada” y su valor puede ser una cadena de texto representando un número n , donde n es un número natural con valor superior a cero.

5.3.5. Script migración

Para traspasar los datos de la base de datos de Drupal a la nueva base de datos, se ha creado un script que realiza dicha tarea.

Este script dispone de un archivo de configuración del cual se obtienen los datos de conexión tanto de la base de datos original como de la nueva base de datos. Una vez ejecutado el script, este irá haciendo consultas al origen y adaptará los datos a la nueva estructura de la base de datos de Laravel.

Las imágenes de los brigadistas, fuentes documentales y noticias y los archivos relacionados han de ser copiados manualmente al directorio `storage/app/public`. Para acceder a ellos a través de urls, se ejecuta el comando `php artisan storage:link`. Esto permite acceder a los ficheros a través de una url como `http://dominio.com/storage/image.jpg`.

Para comprobar la carga de información y de imágenes se han creado cuatro vistas. Las urls de las vistas se encuentran en el archivo `web.php` del directorio `routes` dentro del proyecto Laravel.

5.4. Seguridad

Este capítulo detalla aspectos sobre la seguridad del sistema, la parte más difícil de implementar correctamente. En la actualidad es importante proteger las aplicaciones y sitios web de posibles ataques maliciosos como la obtención de información sensible, la suplantación de usuarios, la explotación de agujeros de seguridad o la caída de servicios.

Laravel cuenta con un componente de seguridad para implementar de manera fácil la autenticación y la autorización, para evitar que un usuario acceda a un recurso al cual no debería tener acceso.

5.4.1. Autenticación

La autenticación es el proceso por el cual la aplicación trata de confirmar que alguien es quien dice ser, a través de las credenciales de acceso. Después la aplicación decide si ese alguien debe tener acceso a un determinado recurso.

Los servicios de autenticación de Laravel están contruidos con *guardias* y *proveedores*. Los *guardias* definen cómo los usuarios se autentifican por cada solicitud (por ejemplo Laravel usa una sesión para mantener el estado usando el almacenamiento de las sesiones y cookies).

Los *proveedores* definen cómo se recuperan los usuarios de su almacenamiento persistente. Laravel posee un soporte para recuperar los usuarios usando Eloquent y el generador de consultas de base de datos.

Laravel dispone de unos controladores de autenticación localizados en el directorio `App/Http/Controllers/Auth`. Estos controladores son:

- **RegisterController:** controla el registro de nuevos usuarios.
- **LoginController:** maneja el registro de nuevos usuarios.

- **ForgotPasswordController:** se encarga de enviar por correo electrónico los links para resetear las contraseñas.
- **ResetPasswordController:** contiene la lógica para resetear contraseñas.

Cada uno de estos controladores incluye sus métodos necesarios para cumplir su objetivo, por lo que no es necesario modificarlos.

Usando el comando de consola `php artisan make:auth`, Laravel creará todas las rutas y vistas necesarias para la autenticación.

Una vez autenticado el usuario, se puede obtener el usuario a través de la fachada⁶ `Auth` con dos métodos: `Auth::user()` devuelve el usuario actualmente autenticado y `Auth::id()` devuelve el id del usuario autenticado.

Con la fachada `Auth` se puede comprobar si un usuario aun está autenticado a través de `Auth::check()`.

5.4.2. Autorización

En caso de querer proteger las rutas para que solo sean accedidas por usuarios identificados, Laravel dispone del middleware `Auth`. Este se puede usar en la definición de las rutas o en el constructor del controlador.

```
Route::get('profile', function () {
    // Only authenticated users may enter...
})->middleware('auth');
```

Figura 28: Uso del middleware `Auth` en la definición de una ruta.

```
public function __construct()
{
    $this->middleware('auth');
}
```

Figura 29: Uso del middleware `Auth` en el constructor de un controlador.

5.4.3. Encriptación

La encriptación de Laravel utiliza OpenSSL para aportar la encriptación de tipo AES-256 y AES-128.

⁶Las fachadas proporcionan una interfaz “estática” para las clases que están disponibles en el contenedor de servicios de la aplicación.

Para empezar se debe configurar una *key* (clave) en el archivo de configuración `config/app.php`, aunque se recomienda usar el comando `php artisan key:generate`, que usará el generador de bytes aleatorio de PHP para generar la clave. En el siguiente ejemplo se hace una demostración de como encriptar un valor:

```
public function storeSecret(Request $request, $id)
{
    $user = User::findOrFail($id);

    $user->fill([
        'secret' => encrypt($request->secret)
    ]->save());
}
```

5.4.4. Hashing

Los hash o funciones de resumen son algoritmos que consiguen crear a partir de una entrada (ya sea un texto, una contraseña o un archivo, por ejemplo) una salida alfanumérica de longitud normalmente fija que representa un resumen de toda la información que se le ha dado (a partir de los datos de la entrada crea una cadena que solo puede volverse a crear con esos mismos datos).

Este método tiene varias funciones: está asegurar que no se ha modificado un archivo en una transmisión, hacer ilegible una contraseña o firmar digitalmente un documento.

La fachada Hash que posee Laravel proporciona Bcrypt y Argon2 como métodos de Hashing para almacenar las contraseñas de usuario. A continuación, se muestra un ejemplo de como emplearlo:

```
public function update(Request $request)
{
    // Validate the new password length...

    $request->user()->fill([
        'password' => Hash::make($request->newPassword)
    ]->save());
}
```

6. Conclusiones

Sidbrint necesitaba una nueva imagen para su página web, una imagen moderna y adaptada a los nuevos tiempos con una interfaz sencilla, sin elementos que confundan al usuario e indicando de manera sencilla el objetivo del portal y su funcionalidad. Esto es lo que se ha pretendido realizar a través de los diferentes cambios que se han ido realizando a lo largo del proyecto.

En el desarrollo de portales y aplicaciones web el hecho de que tengan una buena disposición de los elementos del diseño es tan importante como un correcto funcionamiento. Estos elementos no solo han de mostrar funcionalidades o información, si no que además deben estar bien organizados y bien posicionados en función del objetivo que se ha establecido para el portal: si este sirve para realizar búsquedas, es importante transmitir al usuario este objetivo a través de una barra de búsqueda que destaque del resto de elementos de un portal.

Se ha comprendido que al trabajar con clientes y con un tiempo limitado es muy importante planificar bien el trabajo, saber escuchar las necesidades del cliente y ayudarle en la toma de decisiones importantes, saber decir que tareas son posibles y cuales no se pueden llevar a cabo por complejidad o por inversión de tiempo.

A nivel técnico, se ha aprendido como funciona un gestor de contenidos como Drupal y cual es la necesidad que satisfacen, que es la de dar herramientas a gente con pocos conocimientos de informática para poder desarrollar y añadir contenido en sus blogs o portales web de manera sencilla. El hecho de configurar la máquina virtual para tener acceso a la base de datos desde la máquina anfitriona a través de SSH, configurar las conexiones y un servidor FTP para el traspaso de archivos ha dado una pequeña visión de como se crea un servidor de desarrollo con el cual desarrollar una solución web.

Se ha conseguido desarrollar las tareas siguientes tareas:

- **Instalación tema nuevo:** se ha conseguido instalar un nuevo tema, partiendo del tema principal Bootstrap.
- **Página principal:** se han realizado los cambios pertinentes a través del uso de bloques.
- **Página buscador:** se ha mejorado la presentación de los resultados de búsqueda y es posible elegir el número de resultados a visualizar. No se ha conseguido resolver la visualización de palabras clave en una estructura de árbol.
- **Página ficha resultado brigadista:** se ha conseguido crear un sistema de pestañas para distribuir la información. También se ha mejorado la visualización usando Bootstrap.
- **Página ficha resultado fuente documental:** se ha conseguido eliminar información duplicada y se ha maquetado con Bootstrap.

En cuanto a la segunda parte se ha aprendido la importancia de diseñar un buen esquema de base de datos para almacenar cantidades grandes de información,

donde almacenar los datos de manera organizada y con sentido en diferentes tablas ayuda al mantenimiento y crecimiento de esta. También se ha visto una parte del framework Laravel a través de un tutorial y de la instalación de este en una nueva máquina virtual y trabajando en esta.

Se han podido resolver las siguientes tareas:

- **Crear máquina virtual:** se ha creado y configurado la máquina virtual que contiene Laravel.
- **Creación de un nuevo esquema de base de datos:** se ha analizado la base de datos de Drupal. A partir de este análisis, se ha diseñado un esquema de la nueva base de datos para Laravel. Una vez diseñado el esquema, se han escrito las diversas migraciones de las tablas en el framework Laravel, con la posibilidad de poder exportar la base de datos en un motor MySQL.
- **Script migración:** se ha desarrollado un script que conecta la base de datos de Drupal. Por cada tabla, descarga los datos y los sube al nuevo esquema de base de datos.
- **Comprobación archivos:** se hace una comprobación de las imágenes de la base de datos, mostrándolas en vistas creadas en Laravel. Esta tarea cuenta como una tarea extra.

Valoración personal

El desarrollo de este proyecto ha sido interesante para simular una posición como desarrollador con la responsabilidad de contactar con los clientes y organizar reuniones para seguir el progreso del desarrollo, analizar las sugerencias y propuestas y gestionando el tiempo establecido para el proyecto.

Durante el proyecto se optó por dar prioridad al portal en funcionamiento con Drupal y dejar para el final algunas tareas del futuro portal, lo cual inicialmente no era una decisión de agrado ya que no era el planteamiento inicial del proyecto. Resolver los problemas que tenía Drupal fue una tarea un tanto frustrante al principio (se debe tener en cuenta que dichos problemas eran causados por el incumplimiento de las buenas prácticas de Drupal).

El hecho de haber trabajado con un sistema de gestión de contenidos ha ayudado a entender que estas herramientas pueden ser una alternativa al desarrollo de sitios web con frameworks, permitiendo un desarrollo mas rápido y facilidad de gestión por parte de usuarios.

Finalmente cabe mencionar que la realización del proyecto ha aportado mas conocimiento para realizar proyectos web a nivel profesional. Ha sido una buena experiencia trabajar para un proyecto de investigación como Sidbrint debido a su misión de digitalizar la memoria histórica de la Guerra Civil Española.

Trabajos futuros

El desarrollo de este proyecto lleva a pie a otros futuros proyectos gracias a la disposición de grandes cantidades de información sobre las brigadas internacionales en la guerra civil:

- Creación de una API abierta: para la exportación de los datos y la explotación para la investigación es posible crear una API abierta dónde investigadores de todo el mundo puedan obtener datos por separado (un brigadista en concreto por ejemplo) o datos de una consulta (obtención de los datos de todos los brigadistas que participaron en una batalla concreta y que su origen era un país en concreto).
- Proyectos de BigData: una vez creada la API abierta y obtenido los datos, se pueden realizar diferentes estudios sobre los brigadistas en la guerra civil a través de los diferentes parámetros de los brigadistas que se tienen almacenados en la base de datos.
- Proyectos de visualización de datos: usando las tecnologías de visualización de datos que existen a día de hoy, se pueden representar los datos con gráficos de diversos tipos o incluso representar los datos sobre un mapa (por ejemplo mostrar un mapa de calor sobre el número de muertes de brigadistas en las diferentes batallas que hubo en la península Ibérica).
- Continuidad del proyecto actual: partiendo del trabajo realizado en Drupal, se puede seguir desarrollando nuevas características, como por ejemplo añadir el nodo batallas en el sistema y mostrar más información sobre estas.
- Continuidad del futuro proyecto: partiendo del servidor montado con Laravel y la base de datos traspasada a través del script que se encarga de migrar los datos de Drupal, se puede continuar en el desarrollo del portal, decidiendo que tecnología de Front-End es la más idónea para el proyecto.

7. Anexo I. Reuniones

A continuación se detallan las nueve reuniones realizadas a través del transcurso del proyecto, indicando en que fecha sucedieron, quienes participaron y que se discutió.

1º Reunión

La primera reunión se lleva a cabo el 13 de Diciembre del 2017. Los asistentes son los responsables de Sidbrint para discutir sobre los cambios a realizar, teniendo en mente un nuevo sistema que usará tecnologías modernas. En esta reunión se designan las tareas deseadas, a partir de los problemas que se mencionan a continuación:

- Interfaz de usuario:

- Se necesita cambiar la página principal debido a que no es intuitiva para el uso de los usuarios. Para ello se desea eliminar el enlace a la pantalla de login, destacar el buscador y eliminar elementos tales como el slide de debajo de la cabecera.
- Se requiere una nueva tipografía, ya que la usada es Arial y no convence a los responsables.
- Crear el diseño con la posibilidad de su manipulación: los componentes de la página han de poder añadirse o quitarse de la vista en un menú de configuración, solo accesible para usuarios que tengan el rol con privilegios para hacerlo (en este caso sería el usuario administrador).
- Los resultados de búsqueda deben mostrar menos información debido a que cada resultado tiene una altura bastante grande para la pantalla del navegador. Para ello se ha de ocultar información, como las palabras claves de cada resultado, que ocupan una gran parte de la caja que contiene estos.
- Los títulos de los resultados de búsqueda de las fuentes documentales, al ser largos, quedan cortados en las cajas de resultados.
- En la pantalla de resultados, el filtro de palabras clave se quiere mostrar en forma de árbol, ya que las palabras claves están organizadas en categorías.
- En las fichas de brigadistas, si un brigadista tiene muchas referencias bibliográficas, estas ocupan una gran parte de la página. Una solución es crear dos pestañas: una con la información del brigadista y otra pestaña con las referencias bibliográficas.
- En las fichas de fuentes documentales, el título de estas se muestra tres veces: una como título de la página, otro en los breadcrumbs (semillas de pan) y otro para el enlace a la referencia. Una posible solución sería eliminar los breadcrumbs.

- En las fichas de fuentes documentales, se ha de poder diferenciar si la referencia enlaza a un archivo completo o si hace referencia a un documento en una biblioteca.
- Administración:
 - Se desea poder cambiar el color de fondo a través de la configuración del sitio, solo accesible para el usuario administrador.
 - Se desea poder elegir que cajas se verán en el futuro menú que se mostrará en la página principal del portal. Este menú consistirá en una cuadrícula de cajas con enlaces a diferentes secciones del portal.
 - El usuario administrador deberá poder cambiar el logo de la página a través de la configuración del portal.
 - En el nuevo portal se desea que el usuario administrador pueda añadir nuevos usuarios y pueda gestionarlos.

Además de los requerimientos de la interfaz de usuario y de administración, se discuten los siguientes requerimientos:

- El nuevo portal ha de ser multi-idioma, con los idiomas Castellano, Catalán e Inglés.
- Se han de crear los roles que ya existían en la anterior plataforma. Estos son:
 - **Administrador:** administra el sitio. Tiene todos los privilegios.
 - **Super-editor:** puede crear, editar, eliminar y aprobar cualquier tipo de contenido.
 - **Editor de fuentes:** puede crear, editar o eliminar fuentes, brigadistas y encuadramientos militares.
 - **Editor** solo puede crear, editar o eliminar brigadistas y encuadramientos militares.
- Las fichas de brigadistas, fuentes documentales y encuadramientos militares se han de poder exportar en formato CSV, XML y Mendeley, para poder trabajar con los datos.

2º Reunión

En la segunda reunión realizada el día 31 de Enero del 2018, se hace entrega de la máquina virtual que contiene una copia del servidor Ubuntu Server que incluye todo el sistema de Sidbrint (Drupal, la base de datos, etc). El mismo día, se contacta con el responsable de los servidores con la idea de cambiar la contraseña de acceso a la cuenta de administrador de Ubuntu para poder acceder a los datos de Drupal y configurar el acceso a ellos.

3º Reunión

Se realiza una tercera reunión, llevada a cabo el día 12 de Febrero del 2018 y con la participación de un responsable de Sidbrint, para aclarar dudas sobre el futuro sistema:

- En el sistema de Drupal existen apartados de los que no se habían hablado. Estos son: colecciones digitales, especialistas, recursos de la web, etc. Estos apartados no se requieren en el nuevo sistema, los únicos apartados que se requieren son brigadistas, fuentes documentales y encuadramientos militares.
- En el formulario de creación de brigadistas de Drupal existen unos campos compuestos por un botón de radio (radiobutton) y un campo de texto cada uno. En estos campos se pregunta si se conoce cierta información del brigadista, por ejemplo si se sabe el lugar donde éste nació. Para ello se marca o se desmarca el botón de radio, que contiene las opciones *Sí* o *No*. Si se marca *Sí*, se muestra la caja de texto para rellenar con la información pertinente. Se ha decidido que en el nuevo sistema se prescindirán de los botones de radio y se mostrarán siempre las cajas de texto, que podrán quedar vacías si no se conocen los datos pertinentes.
- En la base de datos, existen brigadistas que no tienen género. Esto se debe a que se desconoce cual es su género ya que hay nombres que en unos países son de hombre y en otros países son de mujer. Se decide actualizar los géneros no asignados a hombre, ya que la gran mayoría de brigadistas eran hombres y estadísticamente hay muchas más probabilidades de que sean hombres a que sean mujeres.
- En la base de datos, existen brigadistas que no pertenecen a ningún cuerpo. Un brigadista puede pertenecer al cuerpo de tierra, aire o mar en caso de que fuera soldado y no médico (estos no luchaban). Se decide que los brigadistas que no tienen asignado un cuerpo se les asigne el cuerpo de tierra, ya que éste es el cuerpo que tenía mas soldados.

En el período de tiempo que se produjo esta reunión se estaba investigando como estaba estructurada la base de datos de Sidbrint y se estaba analizando la gestión del sitio con el uso de un usuario con rol administrador.

4º Reunión

La cuarta reunión se concertó para seguir resolviendo dudas sobre el sistema de Drupal. Esta reunión se llevó a cabo el día 19 de Febrero de 2018 con el responsable de Sidbrint que administraba el portal. Se aclararon las siguientes dudas:

- En el formulario de brigadistas, hay un campo de texto, llamado *metadatos* que no parece tener un uso y no es un campo obligatorio. Este campo no se

usaba, pero se dejó para un futuro desarrollo en el cual se recogen ontologías. Este desarrollo no se realiza, ya que tiene un coste de tiempo elevado y no es un requisito prioritario.

- Se desconocen las funciones que desempeña cada rol y que personas de Sidbrint tienen asignado cada rol.
 - **Editor:** Son estudiantes o investigadores que se ofrecen voluntarios para crear nuevas fichas de brigadistas y encuadramientos militares, editarlas o eliminarlas.
 - **Editor fuentes:** personal de Sidbrint que puede añadir nuevas fichas de fuentes documentales, editarlas o eliminarlas.
 - **Super-editor:** personal de sidbrint concreto que puede aprobar contenidos.
 - **Admin:** personal de sidbrint concreto que gestiona el portal. Entre sus funciones está la de gestionar los usuarios.
- En el portal de Sidbrint existe un apartado que corresponde a un mapa de Google Maps, donde se muestra el lugar de origen de los brigadistas, las batallas en España y la muerte de los brigadistas. La cuestión era saber si se quiere mantener el mapa en el nuevo sistema y de dónde salen los datos que usa el mapa. El mapa se quiere mantener, pero este es un desarrollo externo al Drupal del portal así que no interfiere en el proyecto. Los datos usados no son extraídos de la base de datos que usa Drupal, sino que son obtenidos de un archivo CSV.

Cuando se realizó esta reunión, había en marcha el proceso de diseñar un nuevo esquema de la base de datos para la nueva plataforma, teniendo en cuenta como esta estructurada la base de datos de Drupal, para no perder información.

5º Reunión

La quinta reunión se realiza el 23 de Febrero del 2018 con la participación del responsable de Sidbrint. En esta reunión se presenta la estructura de la nueva base de datos, sus ventajas y sus inconvenientes a través de una breve presentación de diapositivas (esta información se muestra en el sub apartado **Base de Datos** del apartado **Sistema futuro**). Esta estructura es bien recibida y aceptada, en consecuencia se designan las tareas para crear la estructura de la base de datos y de migrar los datos del sistema Drupal a la nueva estructura.

Se identifican las tareas que son importantes para el nuevo portal:

- Creación de la nueva estructura de base de datos.
- **Migración de los datos:** es necesario extraer los datos de la base de datos de Drupal y añadirlos a la nueva base de datos sin perder información y que esté bien relacionada.

- **Página principal:** debe tener en cuenta los requisitos de diseño de la primera reunión.
- **Buscador:** se ha de poder buscar información y se ha de poder filtrar a través de unos campos.
- **Gestión de datos:** se ha de poder añadir, editar y eliminar información teniendo en cuenta el rol del usuario que está realizando dichas acciones.

También se analizaron futuros desarrollos que, por falta de tiempo, no se pueden llevar a cabo en este proyecto:

- **Campo metadatos:** se han de poder gestionar las ontologías de los brigadistas y luego usarlas.
- **Mapa:** se ha de integrar el mapa al sistema, recoger los datos de la base de datos y mostrarlos en él.
- **Sección de batallas:** se quiere añadir una nueva sección con información sobre las batallas que ocurrieron durante la Guerra Civil Española.

6º Reunión

Reunión transcurrida el día 9 de Marzo del 2018 y realizada con el responsable de la empresa que lleva el Drupal de Sidbrint. Se discuten los cambios que hay que hacer en el Drupal actual (está reunión se realiza después de acordar que primero se harán cambios en el Drupal).

Se llega a las siguientes conclusiones:

- Se decide usar un nuevo tema de Drupal que incluye Bootstrap, para facilitar el trabajo de maquetación. Esto se decide así porque es mas fácil empezar con un nuevo tema que modificar el tema anterior, además de que se puede usar Bootstrap.
- Se ha de analizar el desarrollo la empresa anterior que estuvo trabajando en este Drupal, ya que no siguieron las buenas prácticas de Drupal, complicando así algunas tareas, como la de ocultar ciertos elementos en la pantalla principal.
- Se han de añadir dos ramas en un repositorio de BitBucket: una contendrá el tema actual con los cambios que hace la empresa y otra contendrá el tema de Bootstrap con los cambios nuevos.
- se creará un servidor para poder ver los cambios y hacer un seguimiento.

En cuanto al nuevo sistema, se discute como hacer las búsquedas, ya que hay muchos datos y se pueden realizar varias consultas simultáneas, por lo que hay que evitar que el servicio se colapse. Lo bueno del nuevo diseño de la Base de Datos es que evita los múltiples joins que hace para mostrar los datos de brigadistas, encuadramientos y fuentes. Por otra parte, la base de datos de Drupal tiene cacheadas las búsquedas, por lo que devuelve búsquedas realizadas y es más rápido al devolver resultados, pero la caché ocupa mucho espacio.

7º Reunión

Reunión con el responsable de Sidbrint con fecha de 9 de Abril del 2018. Se enseñan los cambios realizados en el portal de Drupal.

Portada

En la portada está visible la cuadrícula con seis enlaces, tal como muestra la figura 9 (subapartado *Cambios realizados* del capítulo *Sistema Actual*):

- Brigadistas: enlaza a la página de resultados de búsqueda con elementos de tipo Brigadista.
- Fuentes documentales: enlaza a la página de resultados de búsqueda con elementos de tipo Fuentes Documentales.
- Encuadramientos Militares: enlaza a la página de resultados de búsqueda con elementos de tipo Encuadramiento Militar.
- Proyecto Sidbrint: enlaza a la página que contiene información sobre el proyecto.
- Colaboradores: enlaza a la página con información sobre los colaboradores e investigadores del proyecto Sidbrint.
- Noticias: enlaza a la página que contiene las noticias del portal, mostradas en lista.

Se han quitado los bloques no deseados, cambiado los iconos de las redes sociales y dejado solo los idiomas Catalán, Castellano e Inglés. También se pide quitar los contenidos mostrados, que aún no se han podido quitar por falta de conocimiento.

Al responsable le interesa añadir filas de cajas con enlaces, donde la primera corresponde a búsquedas para realizar, la segunda fila contiene imágenes, noticias y temas relacionados y la tercera fila contiene información del proyecto, colaboradores y los avisos legales. Esta decisión se toma teniendo en cuenta las ideas que tiene el responsable basadas en otras portadas de otros sitios web, tales como Wikipedia⁷.

Búsqueda

En la pantalla de búsqueda, se mencionan los siguientes problemas:

⁷<https://es.wikipedia.org/wiki/Wikipedia:Portada>

- Para solucionar el problema de la altura de cada caja de resultados, se pide reducir el tamaño de las imágenes.
- Los alias de los brigadistas no aparecen separados (un brigadista puede tener más de un alias), por lo cual hay que añadir una separación para que se visualicen bien.
- La búsqueda de encuadramientos no tiene ningún filtro, se requiere añadir un filtro de *designación específica*.
- El filtro de palabras clave sigue sin estar en forma de árbol. Esta tarea se ha investigado, pero no se ha avanzado debido a las dificultades que conlleva, por lo que se priorizan otras tareas.
- Se reporta un problema en el buscador: cuando se realiza una búsqueda y se usa un filtro, el filtro funciona sobre la búsqueda realizada, pero si hay filtros activos y se realiza una búsqueda, esta búsqueda no contempla los filtros activados anteriormente, por lo que buscará en todos los datos. Interesa que cuando se haga una búsqueda, la haga con los datos filtrados anteriormente.
- Las imágenes de las fuentes documentales salen cortadas verticalmente.

Ficha resultados

La ficha de brigadista necesita pestañas para repartir mejor la información y tener una vista menos saturada de datos. Para ello se harán dos pestañas: una con la información general y la otra con las referencias del brigadista.

Se propone una reunión con el encargado del Drupal de Sidbrint.

8º Reunión

La octava reunión se produce el 13 de Abril del 2018 con el responsable del Drupal de Sidbrint. En esta reunión se muestra los cambios realizados en el portal de Sidbrint.

Con la ayuda del responsable se elimina la visualización del contenido (noticias, brigadistas, fuentes y encuadres) que se mostraba en portada y se crean las pestañas de la ficha de brigadista.

Se menciona la propuesta de diseñar la portada con las cajas de enlaces y se plantea otra solución.

El diseño de las cajas ocupa la totalidad de la portada para mostrar enlaces, sin ninguna información que explique que es Sidbrint o que muestre contenido como las últimas noticias (para llegar a este contenido, se debe ir a los enlaces correspondientes). Este diseño está orientado más a hacer búsquedas.

Es recomendable mostrar en la portada una descripción sobre que es el proyecto Sidbrint, para que sirva y cuáles son sus virtudes, para que los nuevos usuarios tengan esta información disponible en la portada sin tener que navegar por el portal.

Tener las últimas noticias en portada permite tener actualizada a la comunidad de usuarios que visitan la web y permite una mayor indexación (SEO) en los resultados de buscadores, ya que el hecho de que una portada se actualice es uno de los criterios prioritarios para los indexadores de Google y otros buscadores.

Por otra parte, se plantea un diseño de buscador “ágil”, inspirado en Airbnb⁸, empresa donde una gran parte de su negocio es el proceso de búsqueda. Para ello se sugiere la creación de una barra de búsqueda mucho más grande y centrada para focalizar la atención de los usuarios, además de facilitar el proceso de búsqueda a los usuarios que realizan muchas búsquedas. Se añadirán unos botones bien integrados para dejar claro que tipo de búsquedas se pueden hacer en la web (brigadistas, fuentes y encuadramientos) de manera que no haría falta usar tres cajas grandes si se dispone bien la información y se integra en un sistema de búsqueda que se explique por el solo.

9º Reunión

La novena y última reunión realizada se produce el 5 de mayo de 2018. Junto al responsable de Drupal y el responsable de Sidbrint se hace una comparativa del diseño de cajas con enlace y el diseño que destaca la barra de búsqueda, muestra noticias nuevas y explica la misión de Sidbrint en portada.

Se convence al responsable de Sidbrint de que el diseño actual no es el correcto, que es preferible tener una barra de búsqueda con filtros integrados y una sección de últimas noticias en la página principal. También se habla de arreglar los últimos detalles de la página de resultados y de futuros desarrollos después de la finalización del proyecto.

⁸<https://www.airbnb.es/>

8. Anexo II. Base de datos Drupal

En este anexo se procede a describir los grupos más relevantes del esquema de bases de datos de Drupal mostrado en la figura 6 (apartado *Base de datos* en el capítulo *Sistema actual*).

Tablas de bloques

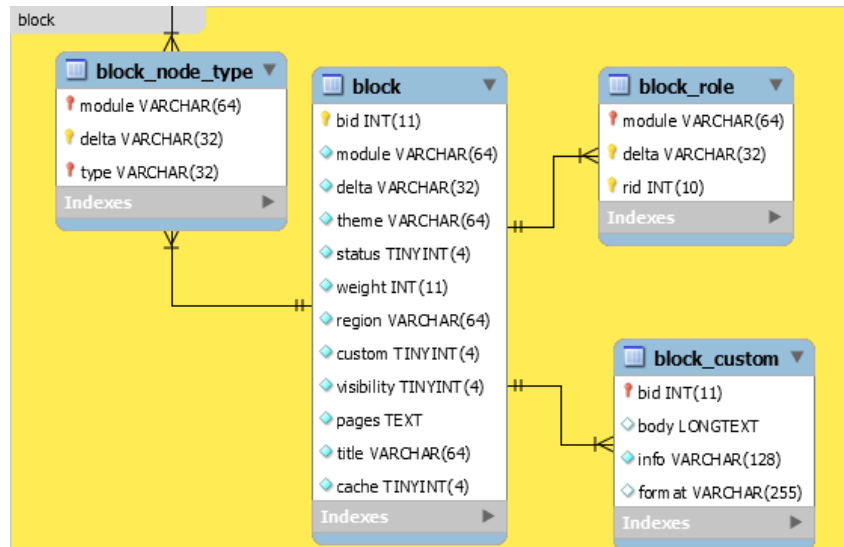


Figura 30: Grupo de tablas bloque

Estas cuatro tablas almacenan la información sobre los bloques:

- **block**: almacena las configuraciones de los bloques.
- **block_custom**: almacena el contenido de los bloques creados por el usuario de Drupal.
- **block_node_type**: almacena la información que configura los criterios de visualización para bloques según el tipo de contenido.
- **block_role**: guarda los permisos de acceso para los bloques basados en los roles de usuarios.

Tablas de caché

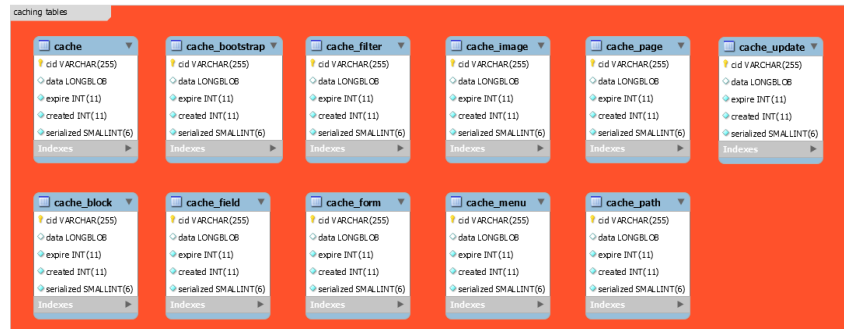


Figura 31: Grupo de tablas caché

Once de las setenta y cuatro tablas que conforman la base de datos existen para dar soporte a la caché. Estas tablas son:

- **cache:** almacena en caché los elementos no separados en sus propias tablas de caché.
- **cache_block:** los módulos de bloque pueden almacenar en memoria caché bloques ya construidos aquí.
- **cache_bootstrap:** los datos requeridos durante el proceso de arranque pueden almacenarse en caché en esta tabla.
- **cache_field:** almacena valores de campos en caché.
- **cache_filter:** almacena piezas de texto filtrado.
- **cache_form:** almacena formularios recientemente construidos y sus datos almacenados.
- **cache_image:** guarda en caché la información sobre la manipulación de imágenes que están en progreso.
- **cache_menu:** guarda información de menús y del árboles de enlaces generados.
- **cache_page:** almacena páginas comprimidas para servir las a usuarios anónimos.
- **cache_path:** guarda en caché los alias de las rutas.
- **cache_update:** guarda en caché la información de las actualizaciones del proyecto.

Tablas de campo

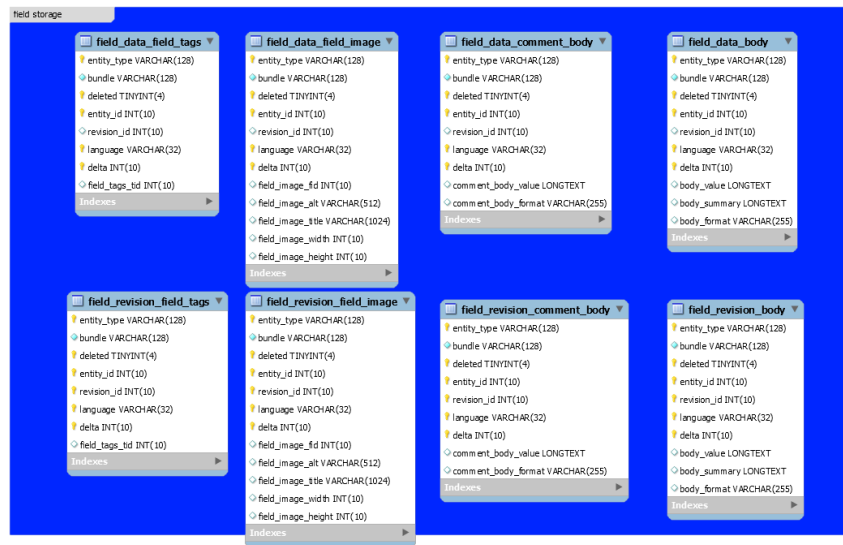


Figura 32: Grupo de tablas campo

Estas tablas son usadas para almacenar valores y revisiones para los campos que están disponibles en una instalación inicial de Drupal 7.

- **field_data.body:** almacena detalles sobre el campo del cuerpo de una entidad.
- **field_revision.body:** almacena información sobre las revisiones de los campos de cuerpo.
- **field_data.comment.body:** almacena información sobre los comentarios asociados a una entidad.
- **field_revision.comment.body:** almacena detalles sobre las revisiones de los comentarios.
- **field_data.field.image:** almacena información sobre las imágenes asociadas a una entidad.
- **field_revision.field.image:** almacena la información sobre las revisiones de imágenes.
- **field_data.field.tags:** almacena información sobre los tags asociados a una entidad.
- **field_revision.field.tags:** almacena información sobre las revisiones de los tags y/o términos de una taxonomía asociada a una entidad.

Tablas independientes

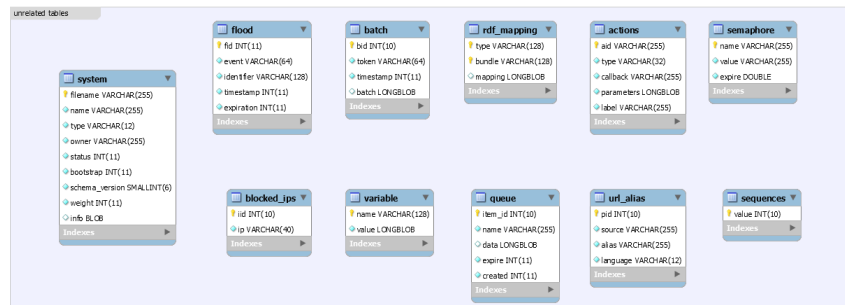


Figura 33: Grupo de tablas independientes

Estas tablas no pertenecen a ningún grupo de tablas:

- **actions**: guarda información de las acciones.
- **batch**: almacena los detalles sobre lotes (procesos que se ejecutan en múltiples solicitudes HTTP).
- **blocked_ips**: almacena una lista de direcciones IP bloqueadas.
- **flood**: controla el umbral de eventos, como el número de intentos de contacto.
- **queues**: almacena ítems en una cola.
- **rdf_mapping**: almacena mapeos RDF personalizados para tipos de contenido definidos por el usuario.
- **semaphore**: almacena semáforos, bloqueos y banderas.
- **sequences**: almacena IDs.
- **system**: contiene una lista de todos los módulos, temas y motores de temas que están o han sido instalados.
- **url_alias**: contiene una lista de alias de URL para las rutas de Drupal.
- **variable**: almacena pares variable/valor creados por el núcleo de Drupal o por un módulo o tema.

Tablas de nodos

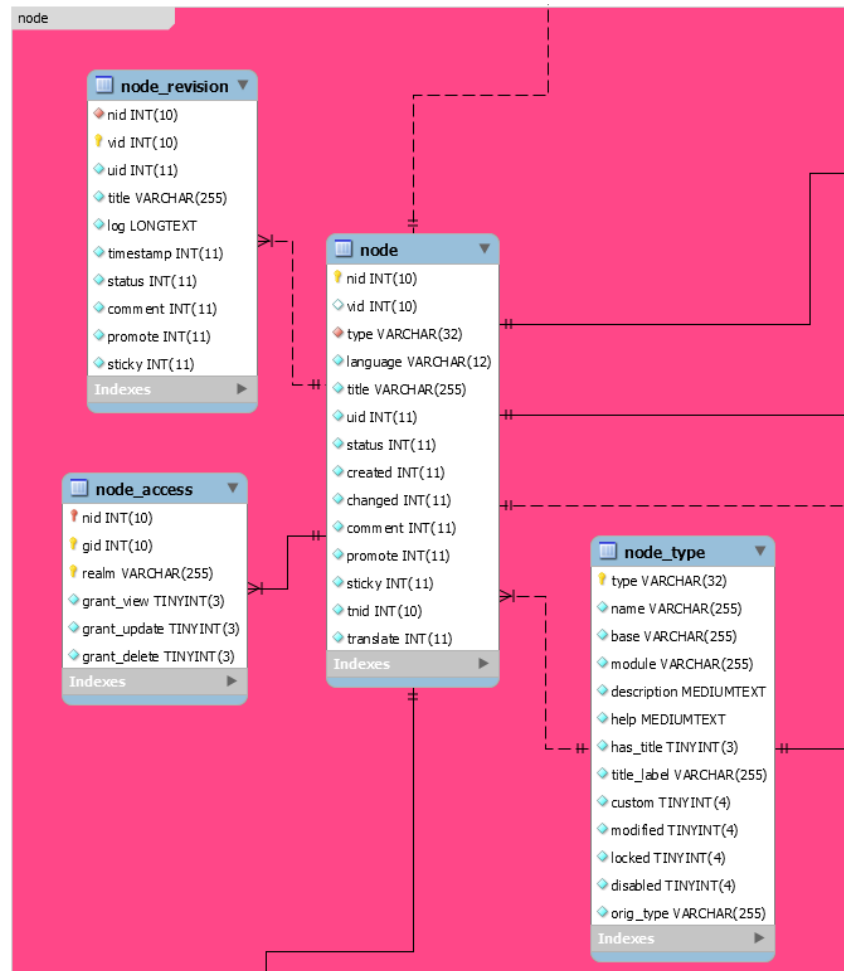


Figura 34: Grupo de tablas nodo

El grupo de tablas de nodos almacena información sobre estos.

- **node**: almacena la información más básica de cualquier tipo de nodo en la base de datos. Esta tabla tiene cinco relaciones con tablas no específicas de nodos:
 - history
 - comment
 - node_comment_statistics
 - search_node_links
 - taxonomy_index
- **node_access**: identifica qué pares de reino/concesión debe poseer un usuario para ver, actualizar o eliminar nodos específicos.

- **node_revision:** guarda información acerca de cada versión guardada de un nodo.
- **node_type:** guarda información sobre todos los tipos de nodos definidos.

Tablas de búsqueda

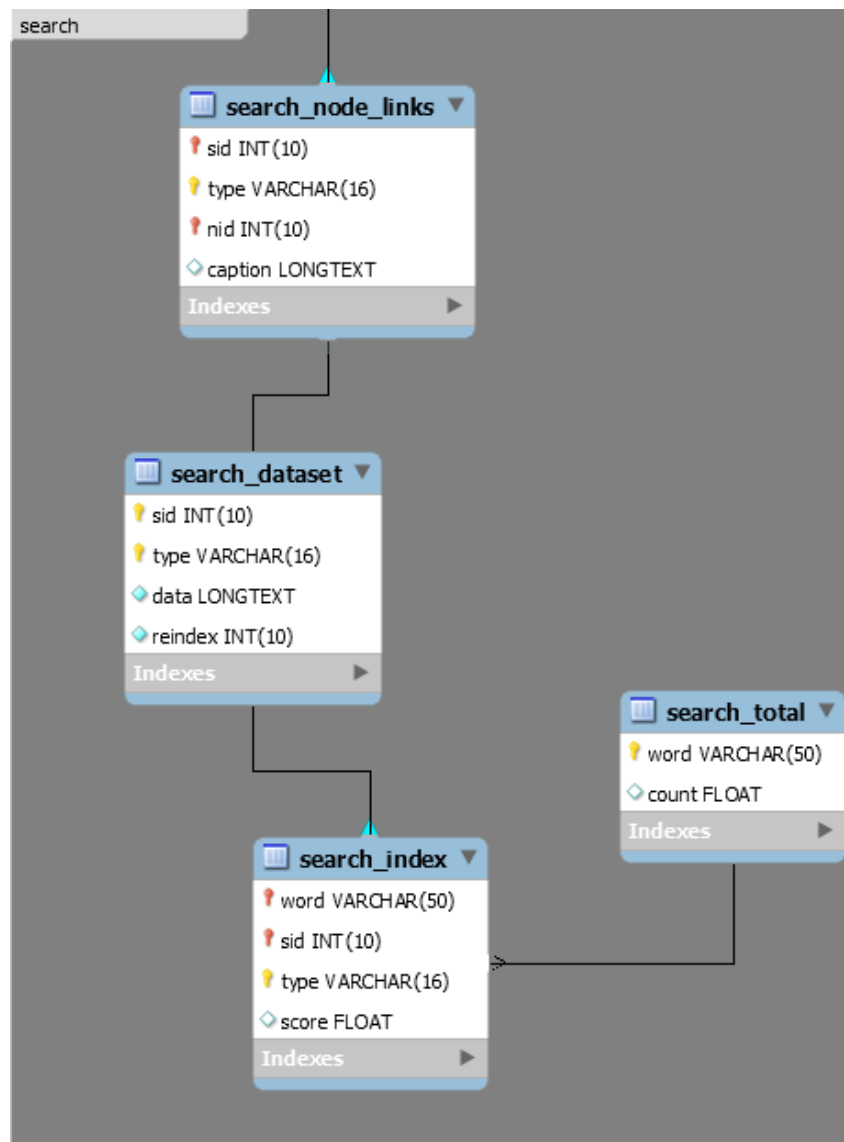


Figura 35: Grupo de tablas búsqueda

Este grupo está formado por las siguientes tablas:

- **search_dataset:** almacena ítems que serán buscados.
- **search_index:** almacena el índice de búsqueda y palabras, ítems y puntuaciones asociados.

- **search_node_links:** guarda ítems que enlazan a otros nodos.
- **search_total:** almacena totales de búsqueda de palabras.

Tablas de taxonomía

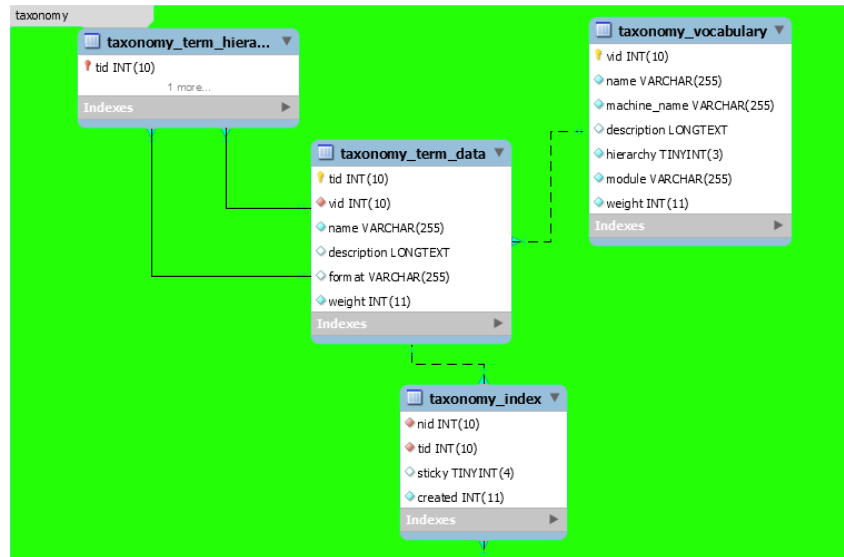


Figura 36: Grupo de tablas taxonomía

En este grupo de tablas se guardan las taxonomías, términos y vocabulario de Drupal. Existen cuatro tablas de taxonomías:

- **taxonomy_index:** mantiene la información desnormalizada sobre la relaciones nodo-término.
- **taxonomy_term_data:** almacena la información del término.
- **taxonomy_term_hierarchy:** almacena las relaciones jerárquicas entre términos.
- **taxonomy_vocabulary:** almacena información del vocabulario.

9. Anexo III. Casos de uso en detalle

En este anexo se procede a describir cada caso de uso descrito en el apartado *Análisis de requerimientos* del capítulo sistema futuro.

UC1. Identificarse	
Objetivo:	Identificarse en la plataforma
Actor:	Usuario anónimo
Precondiciones:	-
Procedimiento:	1. Usuario: entra al formulario de login de Sidbrint a través de http://sidbrint.ub.edu/es/user. 2. Sistema: procesa la petición y muestra el formulario. 3. Usuario: entra su usuario y contraseña y envía el formulario. 4. Sistema: valida los datos. 5. Sistema: da acceso al usuario. 6. Sistema: redirige al usuario a la página principal.
Extensiones:	4.1 Los datos del usuario no están bien introducidos o el usuario no existe. Volver al paso 3. 4.2 El sistema muestra un mensaje de error.

UC2. Buscar	
Objetivo:	buscar información sobre brigadistas, encuadramientos o fuentes
Actor:	Usuario anónimo, editor, editor de fuentes, super-editor, administrador
Precondiciones:	-
Procedimiento:	1. Usuario: entra al portal Sidbrint. 2. Sistema: procesa la petición y muestra la página principal. 3. Usuario: escribe una o varias palabras en el cuadro de búsqueda y presiona el botón de buscar. 4. Sistema: filtra los datos usando las palabras escritas por el usuario y devuelve los resultados paginados en la página de resultados.
Extensiones:	4.1 El usuario no entra ninguna palabra y presiona el botón de búsqueda. 4.2 El sistema devuelve todos los resultados, paginados en la página de resultados.

UC3. Filtrar búsquedas	
Objetivo:	buscar información sobre brigadistas, encuadramientos o fuentes
Actor:	Usuario anónimo, editor, editor de fuentes, super-editor, administrador
Precondiciones:	Haber realizado una búsqueda
Procedimiento:	1. Usuario: realiza una búsqueda. 2. Sistema: procesa la petición y muestra los resultados en la página de resultados. 3. Usuario: selecciona un filtro de búsqueda. 4. Sistema: filtra los datos usando los filtros seleccionados por el usuario y devuelve los resultados paginados en la página de resultados.
Extensiones:	4.1 El usuario no entra ninguna palabra y presiona el botón de búsqueda. 4.2 El sistema devuelve todos los resultados, paginados en la página de resultados.

UC4. Ver fichas	
Objetivo:	buscar información
Actor:	Usuario anónimo, editor, editor de fuentes, super-editor, administrador
Precondiciones:	Haber realizado una búsqueda
Procedimiento:	1. Usuario: realiza una búsqueda. 2. Sistema: procesa la petición y muestra los resultados en la página de resultados. 3. Usuario: selecciona un resultado a través de un enlace. 4. Sistema: devuelve la información de la ficha en una página específica.
Extensiones:	-

UC5. Crear contenido (brigadista o encuadramientos)	
Objetivo:	Crear contenido (brigadista o encuadramientos) y guardarlo en la base de datos
Actor:	Usuario editor, editor de fuentes, super-editor, administrador
Precondiciones:	Estar logueado
Procedimiento:	1. Usuario: visita Sidbrint. 2. Sistema: procesa la petición solicitada. 3. Usuario: accede a la administración del sitio → nuevo (brigadista, encuadramiento militar). 4. Sistema: presenta el formulario para crear un nuevo contenido. 5. Usuario: rellena los campos y envía el formulario. 6. Sistema: valida el formulario. 7. Sistema: guarda los datos en la Base de datos. 8. Sistema: notifica al usuario el estado de la acción (ok/error).
Extensiones:	6.1 Hay campos obligatorios vacíos. Vuelve al paso 5. 6.2 El sistema devuelve un mensaje de error y remarca los campos necesarios.
UC6. Editar contenido (brigadista o encuadramientos)	
Objetivo:	Editar contenido (brigadista o encuadramientos) y guardarlo en la base de datos
Actor:	Usuario editor, editor de fuentes, super-editor, administrador
Precondiciones:	Estar logueado
Procedimiento:	1. Usuario: visita Sidbrint 2. Sistema: procesa la petición solicitada. 3. Usuario: accede a la administración del sitio → contenido (brigadistas o encuadramientos). 4. Usuario: hace clic en el botón editar del contenido deseado dentro de un listado. 5. Sistema: presenta el formulario para editar el contenido, con los campos llenados con la información previa. 6. Usuario: edita los campos deseados y envía el formulario. 7. Sistema: valida el formulario. 8. Sistema: guarda los datos en la Base de datos. 9. Sistema: notifica al usuario el estado de la acción (ok/error).
Extensiones:	7.1 Hay campos obligatorios vacíos. Vuelve al paso 6. 7.2 El sistema devuelve un mensaje de error y remarca los campos necesarios.

UC7. Eliminar contenido (brigadista o encuadramientos)	
Objetivo:	Eliminar contenido (brigadista o encuadramientos) y guardar cambio en la base de datos
Actor:	Usuario editor, editor de fuentes, super-editor, administrador
Precondiciones:	Estar logueado
Procedimiento:	1. Usuario: visita Sidbrint 2. Sistema: procesa la petición solicitada. 3. Usuario: accede a la administración del sitio → contenido (brigadistas o encuadramientos). 4. Usuario: hace clic en el botón eliminar del contenido deseado dentro de un listado. 5. Sistema: cambia el campo eliminado del contenido a True en la base de datos. 6. Sistema: notifica al usuario el estado de la acción (ok/error).
Extensiones:	-
UC8. Desconectar	
Objetivo:	Cerrar la sesión del sitio
Actor:	Usuario editor, editor de fuentes, super-editor, administrador
Precondiciones:	Estar logueado
Procedimiento:	1. Usuario: visita Sidbrint 2. Sistema: procesa la petición solicitada. 3. Usuario: presiona el link de Logout. 4. Sistema: procesa la petición y des-autentica al usuario. 5. Sistema: Redirecciona al usuario a la página principal.
Extensiones:	-

UC9. Cambiar contraseña	
Objetivo:	cambiar la contraseña del usuario por la nueva introducida, siguiendo un criterio mínimo de seguridad, en la base de datos.
Actor:	Usuario editor, editor de fuentes, super-editor, administrador
Precondiciones:	Estar logueado
Procedimiento:	1. Usuario: visita Sidbrint 2. Sistema: procesa la petición solicitada. 3. Usuario: accede a la administración del sitio → perfil → cambiar contraseña. 4. Sistema: muestra el formulario de cambio de contraseña. 5. Usuario: escribe se contraseña actual y escribe dos veces la nueva contraseña. Después envía los datos. 6. Sistema: valida los datos. 7. Sistema: realiza el cambio de contraseña en la base de datos. 8. Sistema: notifica al usuario el estado de la acción (ok/error).
Extensiones:	6.1 Hay campos vacíos, la contraseña actual es incorrecta, los dos campos de contraseña nueva no concuerdan o no satisface los requerimientos mínimos. Vuelve al paso 5. 6.2 El sistema devuelve un mensaje de error y remarca los campos incorrectos.
UC10. Crear fuente documental	
Objetivo:	Crear fuente documental y guardarlo en la base de datos
Actor:	Usuario editor de fuentes, super-editor, administrador
Precondiciones:	Estar logueado con rol editor de fuentes
Procedimiento:	1. Usuario: visita Sidbrint. 2. Sistema: procesa la petición solicitada. 3. Usuario: accede a la administración del sitio → nueva fuente documental. 4. Sistema: presenta el formulario para crear una nueva fuente. 5. Usuario: rellena los campos y envía el formulario. 6. Sistema: valida el formulario. 7. Sistema: guarda los datos en la Base de datos. 8. Sistema: notifica al usuario el estado de la acción (ok/error).
Extensiones:	6.1 Hay campos obligatorios vacíos. Vuelve al paso 5. 6.2 El sistema devuelve un mensaje de error y remarca los campos necesarios.

UC11. Editar fuente documental	
Objetivo:	Editar fuente documental y guardarla en la base de datos
Actor:	Usuario editor de fuentes, super-editor, administrador
Precondiciones:	Estar logueado con rol editor de fuentes
Procedimiento:	1. Usuario: visita Sidbrint. 2. Sistema: procesa la petición solicitada. 3. Usuario: accede a la administración del sitio → fuentes documentales. 4. Usuario: hace clic en el botón editar de la fuente deseada dentro de un listado. 5. Sistema: presenta el formulario para editar la fuente, con los campos llenados con la información previa. 6. Usuario: edita los campos deseados y envía el formulario. 7. Sistema: valida el formulario. 8. Sistema: guarda los datos en la Base de datos. 9. Sistema: notifica al usuario el estado de la acción (ok/error).
Extensiones:	7.1 Hay campos obligatorios vacíos. Vuelve al paso 6. 7.2 El sistema devuelve un mensaje de error y remarca los campos necesarios.
UC12. Eliminar fuente documental	
Objetivo:	Eliminar fuente documental y guardar cambio en la base de datos
Actor:	Usuario editor de fuentes, super-editor, administrador
Precondiciones:	Estar logueado con rol editor de fuentes
Procedimiento:	1. Usuario: visita Sidbrint. 2. Sistema: procesa la petición solicitada. 3. Usuario: accede a la administración del sitio → fuentes documentales. 4. Usuario: hace clic en el botón eliminar de la fuente deseada dentro de un listado. 5. Sistema: cambia el campo eliminado de la fuente a True en la base de datos. 6. Sistema: notifica al usuario el estado de la acción (ok/error).
Extensiones:	-

UC13. Crear noticia	
Objetivo:	Crear una noticia y guardarla en la base de datos
Actor:	super-editor, administrador
Precondiciones:	Estar logueado con rol super-editor o administrador
Procedimiento:	1. Usuario: visita Sidbrint. 2. Sistema: procesa la petición solicitada. 3. Usuario: accede a la administración del sitio → nueva noticia. 4. sistema: presenta el formulario para crear una nueva noticia. 5. Usuario: rellena los campos y envia el formulario. 6. Sistema: valida el formulario. 7. Sistema: guarda los datos en la Base de datos. 8. Sistema: notifica al usuario el estado de la acción (ok/error).
Extensiones:	6.1 Hay campos obligatorios vacíos. Vuelve al paso 5. 6.2 El sistema devuelve un mensaje de error y remarca los campos necesarios.
UC14. Editar noticia	
Objetivo:	Editar noticia y guardarla en la base de datos
Actor:	Usuario super-editor, administrador
Precondiciones:	Estar logueado con rol super-editor o administrador
Procedimiento:	1. Usuario: visita Sidbrint. 2. Sistema: procesa la petición solicitada. 3. Usuario: accede a la administración del sitio → noticias. 4. Usuario: hace clic en el botón editar de la noticia deseada dentro de un listado. 5. Sistema: presenta el formulario para editar la noticia, con los campos llenados con la información previa. 6. Usuario: edita los campos deseados y envia el formulario. 7. Sistema: valida el formulario. 8. Sistema: guarda los datos en la Base de datos. 9. Sistema: notifica al usuario el estado de la acción (ok/error).
Extensiones:	7.1 Hay campos obligatorios vacíos. Vuelve al paso 6. 7.2 El sistema devuelve un mensaje de error y remarca los campos necesarios.

UC15. Eliminar noticia	
Objetivo:	Eliminar noticia y guardar cambio en la base de datos
Actor:	Usuario super-editor, administrador
Precondiciones:	Estar logueado con rol super-editor o administrador
Procedimiento:	1. Usuario: visita Sidbrint. 2. Sistema: procesa la petición solicitada. 3. Usuario: accede a la administración del sitio → noticias. 4. Usuario: hace clic en el botón eliminar de la noticia deseada dentro de un listado. 5. Sistema: cambia el campo eliminado de la noticia a True en la base de datos. 6. Sistema: notifica al usuario el estado de la acción (ok/error).
Extensiones:	-
UC16. Crear taxonomía	
Objetivo:	Crear una taxonomía y guardarla en la base de datos
Actor:	super-editor, administrador
Precondiciones:	Estar logueado con rol super-editor o administrador
Procedimiento:	1. Usuario: visita Sidbrint. 2. Sistema: procesa la petición solicitada. 3. Usuario: accede a la administración del sitio → nueva taxonomía. 4. Sistema: presenta el formulario para crear una nueva taxonomía. 5. Usuario: rellena los campos y envía el formulario. 6. Sistema: valida el formulario. 7. Sistema: guarda los datos en la Base de datos. 8. Sistema: notifica al usuario el estado de la acción (ok/error).
Extensiones:	6.1 Hay campos obligatorios vacíos. Vuelve al paso 5. 6.2 El sistema devuelve un mensaje de error y remarca los campos necesarios.

UC17. Editar taxonomía	
Objetivo:	Editar taxonomía y guardarla en la base de datos
Actor:	Usuario super-editor, administrador
Precondiciones:	Estar logueado con rol super-editor o administrador
Procedimiento:	1. Usuario: visita Sidbrint. 2. Sistema: procesa la petición solicitada. 3. Usuario: accede a la administración del sitio → taxonomías. 4. Usuario: hace clic en el botón editar de la taxonomía deseada dentro de un listado. 5. Sistema: presenta el formulario para editar la taxonomía, con los campos llenados con la información previa. 6. Usuario: edita los campos deseados y envía el formulario. 7. Sistema: valida el formulario. 8. Sistema: guarda los datos en la Base de datos. 9. Sistema: notifica al usuario el estado de la acción (ok/error).
Extensiones:	7.1 Hay campos obligatorios vacíos. Vuelve al paso 6. 7.2 El sistema devuelve un mensaje de error y remarca los campos necesarios.
UC18. Eliminar taxonomía	
Objetivo:	Eliminar taxonomía y guardar cambio en la base de datos
Actor:	Usuario super-editor, administrador
Precondiciones:	Estar logueado con rol super-editor o administrador
Procedimiento:	1. Usuario: visita Sidbrint. 2. Sistema: procesa la petición solicitada. 3. Usuario: accede a la administración del sitio → taxonomías. 4. Usuario: hace clic en el botón eliminar de la taxonomía deseada dentro de un listado. 5. Sistema: cambia el campo eliminado de la taxonomía a True en la base de datos. 6. Sistema: notifica al usuario el estado de la acción (ok/error).
Extensiones:	-

UC19. Aprobar contenidos	
Objetivo:	Aprobar contenido (brigadista, encuadramiento o fuente) y guardar cambio en la base de datos
Actor:	Usuario super-editor, administrador
Precondiciones:	Estar logueado con rol super-editor o administrador
Procedimiento:	1. Usuario: visita Sidbrint. 2. Sistema: procesa la petición solicitada. 3. Usuario: accede a la administración del sitio → contenido (brigadista, encuadramiento o fuente). 4. Usuario: hace clic en el botón aprobar del contenido deseado dentro de un listado. 5. Sistema: cambia el campo aprobado del contenido a True en la base de datos. 6. Sistema: notifica al usuario el estado de la acción (ok/error).
Extensiones:	-
UC20. Bloquear usuarios	
Objetivo:	Bloquear usuarios para evitar la creación, edición o eliminación de contenido
Actor:	Usuario administrador
Precondiciones:	Estar logueado con rol administrador
Procedimiento:	1. Usuario: visita Sidbrint. 2. Sistema: procesa la petición solicitada. 3. Usuario: accede a la administración del sitio → usuarios. 4. Usuario: hace clic en el botón bloquear del usuario deseado dentro de un listado. 5. Sistema: cambia el campo bloqueado del usuario a True en la base de datos. 6. Sistema: notifica al usuario el estado de la acción (ok/error).
Extensiones:	-

UC21. Cambiar número de cajas

Objetivo:	mostrar u ocultar cajas de la cuadrícula mostrada en la pantalla principal
Actor:	Usuario administrador
Precondiciones:	Estar logueado con rol administrador
Procedimiento:	1. Usuario: visita Sidbrint. 2. Sistema: procesa la petición solicitada. 3. Usuario: accede a la administración del sitio → visualización. 4. Usuario: selecciona los checkboxes de los cuadros que quiere mostrar en el apartado Cuadros. 5. Usuario: hace clic en el botón Guardar, enviando el estado de los checkboxes. 6. Sistema: actualiza las preferencias en la base de datos. 7. Sistema: notifica al usuario el estado de la acción (ok/error).
Extensiones:	-

UC22. Crear usuario

Objetivo:	crear un nuevo usuario para acceder al backend
Actor:	Usuario administrador
Precondiciones:	Estar logueado con rol administrador
Procedimiento:	1. Usuario: visita Sidbrint. 2. Sistema: procesa la petición solicitada. 3. Usuario: accede a la administración del sitio → usuarios → nuevo. 4. Sistema: presenta el formulario para crear un usuario nuevo. 5. Usuario: rellena los campos y envía el formulario. 6. Sistema: valida el formulario. 7. Sistema: guarda los datos en la base de datos. 8. Sistema: notifica al usuario el estado de la acción (ok/error).
Extensiones:	6.1 Hay campos obligatorios vacíos. Vuelve al paso 5. 6.2 El sistema devuelve un mensaje de error y remarca los campos necesarios.

UC23. Cambiar contraseñas

Objetivo:	cambiar la contraseña de otro usuario
Actor:	Usuario administrador
Precondiciones:	Estar logueado con rol administrador
Procedimiento:	1. Usuario: visita Sidbrint. 2. Sistema: procesa la petición solicitada. 3. Usuario: accede a la administración del sitio → usuarios. 4. Sistema: presenta el listado de usuarios. 5. Usuario: selecciona el usuario deseado. 6. Sistema: muestra los datos del usuario. 7. Usuario: selecciona cambiar contraseña actual. 8. Sistema: devuelve un formulario con dos cajas de texto. 9. Usuario: entra la contraseña dos veces y la envía. 10. Sistema: valida el formulario. 11. Sistema: modifica la contraseña en la base de datos. 12. Sistema: notifica al usuario el estado de la acción (ok/error).
Extensiones:	10.1 Hay campos obligatorios vacíos, la contraseña es demasiado débil o las dos contraseñas no concuerdan. Vuelve al paso 9. 10.2 El sistema devuelve un mensaje de error y remarca los campos necesarios.

UC24. Cambiar email	
Objetivo	cambiar el email de otro usuario
Actor	Usuario administrador
Precondiciones:	Estar logueado con rol administrador
Procedimiento:	1. Usuario: visita Sidbrint. 2. Sistema: procesa la petición solicitada. 3. Usuario: accede a la administración del sitio → usuarios. 4. Sistema: presenta el listado de usuarios. 5. Usuario: selecciona el usuario deseado. 6. Sistema: muestra los datos del usuario. 7. Usuario: selecciona cambiar email actual. 8. Sistema: devuelve un formulario con una caja de texto. 9. Usuario: entra el email nuevo y lo envía. 10. Sistema: valida el formulario. 11. Sistema: modifica el email en la base de datos. 12. Sistema: notifica al usuario el estado de la acción (ok/error).
Extensiones:	10.1 El campo está vacío o no se ha escrito un email. Vuelve al paso 9 10.2 El sistema devuelve un mensaje de error y remarca los campos necesarios.
UC25. Cambiar color	
Objetivo:	cambiar los colores del portal
Actor:	Usuario administrador
Precondiciones:	Estar logueado con rol administrador
Procedimiento:	1. Usuario: visita Sidbrint. 2. Sistema: procesa la petición solicitada. 3. Usuario: accede a la administración del sitio → visualización. 4. Usuario: selecciona el color a mostrar en el apartado colores. 5. Usuario: hace clic en el botón Guardar, enviando el color deseado. 6. Sistema: actualiza las preferencias en la base de datos. 7. Sistema: notifica al usuario el estado de la acción (ok/error).
Extensiones:	-

UC26. Cambiar fuente

Objetivo:	cambiar la fuente de los textos del portal
Actor:	Usuario administrador
Precondiciones:	Estar logueado con rol administrador
Procedimiento:	1. Usuario: visita Sidbrint. 2. Sistema: procesa la petición solicitada. 3. Usuario: accede a la administración del sitio → visualización. 4. Usuario: selecciona la fuente a mostrar en el apartado fuente. 5. Usuario: hace clic en el botón Guardar, enviando la fuente deseada. 6. Sistema: actualiza las preferencias en la base de datos. 7. Sistema: notifica al usuario el estado de la acción (ok/error).
Extensiones:	-

Referencias

- [1] Drupal 7 documentación
<https://www.drupal.org/docs/7>
- [2] Bootstrap 3 documentación
<https://getbootstrap.com/docs/3.3/>
- [3] W3schools CSS tutorial
<https://www.w3schools.com/css/default.asp>
- [4] Laravel 5.5 documentación
<https://laravel.com/docs/5.5/>
- [5] Laracast tutorial *Laravel from scratch*
<https://laracasts.com/series/laravel-from-scratch-2017/>
- [6] PHP documentación <http://php.net/manual/en/index.php>