



Sistema de Recogida de Datos Cuantitativos en Operativos de Cruz Roja Navarra.

Nombre Estudiante: **Francisco Javier Martínez Rodríguez.**

Plan de estudios: Grado de Ingeniería Informática.

Área trabajo final: Desarrollo Web.

Nombre Consultor/a: Gregorio Robles Martínez

Nombre Profesor/a responsable de la asignatura: Santi Caballe Llobet.

Fecha Entrega: 08 ene 2020.



Esta obra está sujeta a una licencia de Reconocimiento-NoComercial-SinObraDerivada. [3.0 España de Creative Commons.](https://creativecommons.org/licenses/by-nc-nd/3.0/es/)



FICHA DEL TRABAJO FINAL

Título del trabajo:	<i>Sistema de Recogida de Datos Cuantitativos en Operativos de Cruz Roja Navarra.</i>
Nombre del autor:	<i>Francisco Javier Martínez Rodríguez.</i>
Nombre del consultor/a:	<i>Gregorio Robles Martínez.</i>
Nombre del PRA:	<i>Santi Caballe Llobet.</i>
Fecha de entrega (mm/aaaa):	01/2020
Titulación:	<i>Grado de Ingeniería Informática.</i>
Área del Trabajo Final:	<i>Desarrollo Web.</i>
Idioma del trabajo:	<i>Castellano.</i>
Palabras clave:	<i>Desarrollo Web, recogida valores, datos cuantitativos.</i>
<p>Resumen del Trabajo (máximo 250 palabras): <i>Con la finalidad, contexto de aplicación, metodología, resultados y conclusiones del trabajo.</i></p>	
<p>SIREDACU es un sistema creado para el área de Comunicaciones de Cruz Roja Navarra cuya finalidad es la recogida de datos a través de una aplicación que sustituye la actual, que ha quedado obsoleta por la nueva realidad del voluntariado.</p> <p>Para realizar la aplicación se ha realizado una planificación basada en cuatro puntos intermedios de validación, que han marcado el proyecto. I Definición y planificación. II Diseño del sistema. III Implementación de la aplicación. IV Memoria y presentación del proyecto.</p> <p>Se ha utilizado una metodología de programación basada en el ciclo de vida de sistemas ágiles y XP, desarrollando la aplicación en modo local y subiendo partes finalizadas al hosting del proyecto para ser probadas por usuarios reales.</p> <p>Se han marcado los objetivos, resultados y los riesgos del proyecto. Realizados diseños de diagramas, base de datos, iteración y pantalla que muestran el funcionamiento de la aplicación.</p> <p>Para el proyecto se ha recurrido a los lenguajes y <i>frameworks</i>: Bootstrap, Vue, JavaScript, PHP, SQL, JQuery, CSS y HTML; de las siguientes aplicaciones para su desarrollo y gestión: phpMyAdmin, MariaDB, Visual Studio Code, y, GitHub para almacenar los ficheros. La infraestructura de internet para soportar la aplicación debe usar LAMP (Linux, Apache, MySQL/MariaDB y PHP).</p> <p>Se han planteado una serie de pruebas de funcionamiento de las pantallas y mostrado un ejemplo de ejecución para la aceptación del proyecto.</p>	



La aplicación es completamente funcional, aunque se debe profundizar en algunas tecnologías usadas para mejorar la calidad del producto y facilitar la implantación de nuevas utilidades.

Abstract (in English, 250 words or less):

SIREDACU is a system created for the Communications Department of Navarre Red Cross whose purpose is the collection of data through an application that replaces the current one that has become obsolete by the new reality of volunteering.

To carry out the application, it has carried out a planning based on four intermediate validation points that have marked the project. I Definition and planning. II System design. III Implementation of the application. IV Report and presentation of the project.

It has been used a programming methodology based on the life cycle of agile and XP technology, developing the application in local mode and uploading finished parts to the project hosting to be tested by real users.

The objectives, results and risks of the project have been marked. Made diagram designs, database, iteration and screen that show the operation of the application.

For the project, I have used the languages and *frameworks* Bootstrap, Vue, JavaScript, PHP, SQL, JQuery, CSS and HTML; and the following applications for its development and management phpMyAdmin, MariaDB, Visual Studio Code, and, GitHub to store the files. The internet infrastructure to support the application must use LAMP (Linux, Apache, MySQL / MariaDB and PHP).

Have been proposed a series of screen operation tests have been proposed and an example of execution for project acceptance.

The application is fully functional, although must be explored some technologies used to improve product quality and facilitate the implementation of new utilities.



Contenido

1. INTRODUCCIÓN.....	9
1.1. Contexto y justificación del trabajo.	9
1.2. Objetivos del trabajo.	10
1.2.1. Objetivo principal del proyecto.	11
1.2.2. Objetivos generales del proyecto.....	11
1.2.3. Objetivos específicos del proyecto.	12
1.2.4. Resultados esperados.	13
1.2.5. Análisis de riesgos.	14
1.3 Enfoque y método seguido.	15
1.3.1 Estado del arte.....	15
1.3.2 Enfoque y método seguido.	18
1.4. Planificación del trabajo.....	20
1.4.1. Hitos.....	20
1.4.2. Temporización.....	22
1.4.3. Recursos utilizados.	23
1.4.4. Gestión de incidencias.....	25
1.5. Informes de seguimiento.	26
1.6. Breve resumen de productos obtenidos.	35
1.7. Breve descripción del resto de capítulos de la memoria.	38
2. DISEÑOS.....	40
2.1. Diseño de diagramas.....	40
2.2. Diseño de la base de datos.	41
2.3. Flujos de iteración.	46
2.4. Representación visual de pantallas a alto nivel.	49
3. IMPLEMENTACIÓN	51
3.1. Servicios Web.	51
3.2. Base de datos.....	52
3.3. Dispositivos utilizados.	53
4. PRUEBAS.....	55
5. ACEPTACIÓN DEL PROYECTO.....	59
6. CONCLUSIONES.....	64
7. ANEXOS.....	66
7.1. Infografía.	66
7.2. Mejoras en la aplicación.....	67
7.3. Instalación aplicación.	68
8. GLOSARIO	70
9. BIBLIOGRAFÍA.....	71



Lista de figuras.

Ilustración 1 - Tecnologías más populares.....	15
Ilustración 2 - Principales Web <i>Frameworks</i>	15
Ilustración 3 - Principales bases de datos para el desarrollo web.	16
Ilustración 4 - Entornos de desarrollo más populares.....	16
Ilustración 5 - Interconexión entre tecnologías y su uso.....	17
Ilustración 6 - Ciclo de vida XP.	19
Ilustración 7 - Diagrama de Gantt del proyecto.	22
Ilustración 8 - Caso de uso general de la aplicación.	40
Ilustración 9 - Iteración general programa.....	46
Ilustración 10 - Iteración creación usuario.....	46
Ilustración 11 - Iteración creación plantilla.	47
Ilustración 12 - Iteración creación operación.....	47
Ilustración 13 - Iteración creación formulario.....	48
Ilustración 14 - Iteración generación Informe.	48
Ilustración 15 - Evolución diseño pantalla inicial.	49
Ilustración 16 - Evolución diseño pantalla usuario.....	49
Ilustración 17 - Evolución diseño pantalla plantilla.	49
Ilustración 18 - Evolución diseño pantalla formularios.....	50
Ilustración 19 - Evolución diseño pantalla operación.....	50
Ilustración 20 - Pantalla gestión dominio siredacu.crnavarra.org.....	51
Ilustración 21 - Ejemplo enlace a librerías en CDN.	51
Ilustración 22 - Diseño Entidad Relación base de datos.	52
Ilustración 23 - Ejemplo uso Bootstrap.....	53
Ilustración 24 - Ejemplo uso Vue.....	53
Ilustración 25 - Distribución ficheros aplicación web.	53
Ilustración 26 - Depuración de código en Google Chrome	54
Ilustración 27 - Datos generales de una plantilla.....	59
Ilustración 28 - Introducción de ítems y definición de fragmentación de datos.	59
Ilustración 29 - Introducción de etiquetas.....	59
Ilustración 30 - Activación de plantilla.	60
Ilustración 31 - Selección plantilla para operación.	60
Ilustración 32 - Introducción de datos generales a la operación.....	60
Ilustración 33 - Introducción de valores para cada campo creado.....	60
Ilustración 34 - Introducción de etiquetas para la operación.....	61
Ilustración 34 - Activación de la operación.	61
Ilustración 36 - Selección de la operación para el formulario.	61
Ilustración 37 - Identificar unívocamente un formulario.	61
Ilustración 38 - Introducción de datos en la matriz.	61
Ilustración 39 - Validación del envío del formulario.	62
Ilustración 40 - Control de formularios introducidos.	62
Ilustración 41 - Selección de operación / plantilla para realizar el informe.....	62
Ilustración 42 - Selección acotación de datos para el informe.....	62
Ilustración 43 - Ejemplo resultado del informe antes de imprimir.....	63
Ilustración 44 - Infografía del uso de SIREDACU.....	66
Ilustración 45 – Ejemplo en phpMyAdmin para dar de alta la base de datos SIREDACU.....	68
Ilustración 46 – Ejemplo en phpMyAdmin para importar la definición de tablas, campos, relaciones.....	68



Lista de tablas.

Tabla 1 - Temporización del proyecto.	22
Tabla 2 - Gestión de incidencias.	25
Tabla 3 - Definición de valores que componen la tabla usuarios.	41
Tabla 4 - Definición de valores que componen la tabla plantillas.	42
Tabla 5 - Definición de valores que componen la tabla plan_items.	42
Tabla 6 - Definición de valores que componen la tabla plan_tags.	43
Tabla 7 - Definición de valores que componen la tabla operaciones.	43
Tabla 8 - Definición de valores que componen la tabla ope_campo.	44
Tabla 9 - Definición de valores que componen la tabla campo_valor.	44
Tabla 10 - Definición de valores que componen la tabla ope_tags.	44
Tabla 11 - Definición de valores que componen la tabla formularios.	45
Tabla 12 - Definición de valores que componen la tabla form_ope.	45
Tabla 13 - Definición de valores que componen la tabla form_plan.	45



Prólogo.

Al ser este trabajo el último en el que me embarco dentro del Grado de Ingeniería Informática, quería que sirviera para poner en valor la mayor parte del conocimiento que he ido obteniendo durante todos los semestres cursados.

La oportunidad de dar uso en la vida real era un extra de motivación para crear una aplicación que llevaba tiempo queriendo abordar, pero para la cual no tenía los conocimientos suficientes hasta realizar el 'minor' de desarrollo web. Lo aprendido en el Itinerario de Sistemas de Información ha sido de gran ayuda para un correcto seguimiento del proyecto. Por último, los conocimientos adquiridos en muchas de las asignaturas me han servido para hacer unos diseños y una planificación que han concluido con una versión inicial completamente funcional de la aplicación.

Este trabajo, que es el compendio de muchos años de esfuerzo, quisiera que fuera un agradecimiento, por un lado, a mi madre que ha tenido que soportar las noches que andaba por casa, a horas intempestivas, mientras trabajaba en la realización de las Pruebas de Evaluación Continua y otros menesteres de la carrera.

Por otro lado, a todas las personas que han creído en mi trabajo, me han animado, dado su apoyo y me han soportado durante toda esta década. En especial a Iñaki, Iker, Rafael, Raul, Martín, el 'consejo de sabios', mi esposa 'Rachel' y otros muchos que seguro que me dejo en el tintero.

Un proyecto personal que ha durado casi doce años y que, por desgracia, no voy a poder compartir con buenas personas que en este tiempo se han ido. Por eso, quisiera que quedará un recuerdo hacia ellos y, de manera especial nombrar a Jesús Iso y María Pilar Rodríguez, quienes me han marcado de forma especial y a quienes echo muchísimo de menos.

Por último, quisiera agradecer a Víctor Padilla, una gran persona, quien me ha acogido en su casa siempre con gratitud y pendiente en todo momento de mi bienestar, de manera que su hogar se convertía en el centro de operaciones en cada viaje que hacía a Madrid para realizar los exámenes. El hecho de que siempre me haya podido llevar hasta la facultad el día del examen es algo impagable, al darme mucha tranquilidad para afrontar las pruebas. Ni decir el regalazo que suponía la noche después de realizar un examen, los DVD que tenía preparados para estar hasta altas horas viendo películas para celebrar que el semestre había finalizado.



1. INTRODUCCIÓN.

1.1. Contexto y justificación del trabajo.

El presente proyecto busca dar una solución actual a las aplicaciones propias con las que cuenta el área de Sistemas de Información de Cruz Roja Navarra, para la recopilación de datos cuantitativos que hace en determinadas operaciones específicas que no tienen una solución deseada en el ecosistema de aplicaciones existentes a nivel nacional.

El nuevo sistema se adapta a las nuevas realidades tecnológicas y a la metodología de trabajo de las áreas que intervienen, para consignarle un propósito más genérico y un destinatario menos específico, que permite ofrecer una flexibilidad y adaptabilidad para escoger y personalizar los datos que se recogen y su organización.

En la actualidad, la recopilación de datos se realiza con formularios en papel que, una vez se transmiten a un centro, son introducidos por éste en una aplicación, creada en Access, para más tarde generar informes que son enviados al área de Marca y Comunicación, que realiza su difusión en medios de comunicación y redes sociales. De manera manual, se sube a una página web HTML estática un informe pdf que se genera a voluntad. La aplicación es rígida, no permite ninguna flexibilidad, administración, adaptabilidad...

La presente aplicación quiere lograr la introducción de los datos en el origen donde se producen, de manera que se evita la repetición de trabajos que se encadenan hasta que queden almacenados en una base de datos. A su vez, este sistema permitirá lograr la disponibilidad inmediata de los datos en la realización de informes e incluso su difusión por distintos medios sin demora.



1.2. Objetivos del trabajo.

Lo que se pretende en este TFG es la creación de una aplicación web que permita la creación de plantillas por parte de **administradores** para la recogida de datos en la que se **definen** los ítems que debe contener y cómo se fragmentan estos ítems. Estas **plantillas** admiten 'tags' que ayudan a agrupar los datos de manera que facilite más tarde los cálculos de cara a la realización de informes.

Los **organizadores crean operaciones** a partir de las plantillas y las personalizan para cada operación añadiendo nuevos 'tags' que, más tarde, se usan para agrupar o discriminar datos para la realización de informes. Para personalizar la introducción de datos y que se identifique cada formulario de manera inequívoca (cada una de las identidades que se generan a partir de una plantilla), se definen determinados campos, cuya combinación es única dentro de la Operación. Los formularios que se generan para las operaciones son administrados por los organizadores.

Los **operarios**, que serán **anónimos**, son los **encargados de la introducción de datos** y de seleccionar los valores de los campos de identificación, creado por los organizadores, para que **cada formulario** que introduzcan sea único. Deben conocer la clave que hace que la base de datos admita el formulario que rellenen cuando procedan a su envío al servidor.

Cada uno los intervinientes (administradores, organizadores y operarios) puede añadir, leer, modificar y eliminar datos en los formularios. Los administradores son los encargados de crear, leer, modificar y eliminar plantillas, y, tanto **los administradores como los organizadores**, tendrán potestad para **decidir cuándo un formulario se convierte en un elemento de consulta**, impidiendo la alteración de datos por parte de los operarios.

La aplicación tendrá un **espacio que genera informes** de manera ágil, al permitir seleccionar operaciones o plantillas de forma que se acote la búsqueda por campos, 'tags'... Algunos de estos informes serán **accesibles** para que sean visibles por medios de comunicación sin intermediarios. Estos informes se nutren de los datos recopilados que hay en cada momento.



1.2.1. Objetivo principal del proyecto.

Crear una Aplicación web para la recopilación de datos cuantitativos que se realizan en distintas actividades desde Cruz Roja Navarra para la realización y publicación de datos de forma dinámica.

1.2.2. Objetivos generales del proyecto.

Del estudio de lo descrito en el punto 1.2 ‘Objetivos del trabajo’ se han desarrollado una serie de objetivos, que permiten dotar a la aplicación las funcionalidades necesarias para lograr cumplir con los requisitos necesarios para la recogida de datos y su posterior tratamiento.

- ❖ **O1.** Generar un entorno de trabajo que soporte las funcionalidades que el sistema de recogida de datos cuantitativos necesita para gestionar con garantías todas sus partes.

- ❖ **O2.** Crear la interface que permita gestionar los distintos roles de los usuarios que participan en la aplicación.

- ❖ **O3.** Generar el espacio de trabajo para crear las plantillas que contienen la descripción de los datos que se desean obtener y cómo están fragmentados.

- ❖ **O4.** Generar el espacio de trabajo para las operaciones, donde se personalice una plantilla añadiendo campos que luego identificarán un formulario de forma inequívoca y el control de los datos que hayan introducido los usuarios. Se deben establecer las fechas en que se un usuario trabajará con los formularios.

- ❖ **O5.** Generar formularios basados en la plantilla con los valores de selección definidos en la operación, para introducir los datos y enviarlos con los valores deseados.

- ❖ **O6.** Generar la interface que permita seleccionar y acotar los datos para obtener informes.



1.2.3. Objetivos específicos del proyecto.

Se presenta de forma desglosada una relación de trabajos que hay que realizar para alcanzar con éxito las metas de los objetivos generales planteados para este proyecto.

- ❖ **OE1.1** Definir la relación de tablas y campos para almacenar la información necesaria en el sistema planteado.
- ❖ **OE1.2** Seleccionar el servidor de base de datos que soporte las operaciones que se requieran.
- ❖ **OE1.3** Instalar un servidor web que soporte las tecnologías planteadas para el desarrollo del sistema, facilitando a los usuarios su uso.
- ❖ **OE1.4** Lograr la puesta en marcha del entorno web y el funcionamiento de los sistemas que se deben integrar en el servidor para soportar el desarrollo de la aplicación.

- ❖ **OE2.1** Crear usuarios para la aplicación, así como definir sus roles dentro del sistema y su contraseña de acceso.

- ❖ **OE3.1** Crear una pantalla para trabajar plantillas y su estado.
- ❖ **OE3.2** Definir los ítems y cómo están fragmentados en subítems.
- ❖ **OE3.3** Asignar 'tags' a la plantilla.

- ❖ **OE4.1** Crear una pantalla que permite trabajar con operaciones a partir de las plantillas activas.
- ❖ **OE4.2** Implementar campos identificadores que permiten la individualización de los formularios.
- ❖ **OE4.3** Definir las fechas en las que los formularios de la operación estarán disponibles para los usuarios.
- ❖ **OE4.4** Personalizar la relación de 'tags' que usa la operación basados en los definidos en la plantilla.
- ❖ **OE4.5** Comprobar la relación de formularios introducidos para la operación.

- ❖ **OE5.1** Crear una pantalla para introducir datos recopilados a través de formularios que se puede sobrescribir.
- ❖ **OE5.2** Enviar formularios introduciendo una palabra de aceptación definida para la operación.

- ❖ **OE6.1** Seleccionar y realizar acotación de datos para obtener informes.



1.2.4. Resultados esperados.

Al finalizar el proyecto se espera que el sistema propuesto realice una recogida de datos organizada para su tratamiento en tiempo real, de forma que los datos estén disponibles para realizar informes por parte del personal del área de Marca y Comunicación.

- ❖ **RO1.1** Existe una base de datos que soporta la información necesaria para el funcionamiento de la herramienta, la recopilación de datos, su tratamiento y puesta a disposición para consultas, informes...
- ❖ **RO1.2** Existe un servidor web que soporta los requerimientos del proyecto, así como la explotación de forma fluida de la aplicación.

- ❖ **RO2.1** Se controlan los usuarios que se registran en la aplicación y los roles que adoptan.

- ❖ **RO3.1** Se crean y controlan plantillas.
- ❖ **RO3.2** Los usuarios son capaces de asignar cuantos ítems consideren y decidir cómo el conjunto de ítems está fragmentado en subítems.
- ❖ **RO3.3** Se asignan 'tags' a la plantilla.

- ❖ **RO4.1** Los usuarios crean operaciones usando plantillas.
- ❖ **RO4.2** Se generan los campos necesarios para identificar los formularios de forma unívoca y se adaptan los 'tags' para la operación.
- ❖ **RO4.3** Se revisan los datos de los formularios introducidos, bloqueando, modificando y eliminando registros.

- ❖ **RO5.1** Usuarios anónimos son capaces de rellenar un formulario, seleccionar los campos de identificación y enviarlos introduciendo una palabra de aceptación.

- ❖ **RO6.1** Los datos son usados conforme se reciben en informes que se generan seleccionando y acotando valores.



1.2.5. Análisis de riesgos.

El reto planteado en este proyecto no es ajeno a problemas durante su desarrollo. Por un lado, una temporalización exigente para lograr una versión funcional del aplicativo. Por otro, al utilizar varias tecnologías, para algunas de ellas existe un limitado conocimiento que genera incertidumbre sobre la correcta aplicación de las mismas. Se presenta una relación de riesgos analizados para los distintos objetivos, así como una posible solución para los mismos.

- ❖ **ARO1.1 ARO1.2** La base de datos o el servidor web del hosting no soporta una o varias tecnologías. En caso de no soportar las tecnologías habrá que valorar la posibilidad de montar un servidor propio que soporte todas las funcionalidades de la aplicación o la viabilidad de cambiar aquellas tecnologías no soportadas.
- ❖ **ARO2.1** No es suficiente la funcionalidad de la pantalla para la gestión de los roles dentro de la aplicación. Al comienzo y finalización del desarrollo de cada fase se revisará si es necesario modificar el funcionamiento.
- ❖ **ARO3.1 ARO3.2 AR3.3** Los *stakeholders* solicitan ampliación de niveles de implementación de subítems o nuevas funcionalidades. Si las solicitudes no se ajustan a las necesidades para cumplir con la aceptación inicial de la aplicación propuesta, se añadirán a un documento de mejoras.
- ❖ **ARO4.1 RO4.2 RO4.3** Los *stakeholders* solicitan ampliar el tipo de campos que se pueden introducir para la identificación unívoca de los formularios. A pesar de estar fuera de los requerimientos de aceptación inicial, se analizará la complejidad de su implantación. Si no es posible su adaptación, se añadirá al documento de mejoras.
- ❖ **ARO5.1** El sistema de introducción de una contraseña por operación es demasiado sencillo. Se estudiarán otros sistemas que permitan que solo la persona que deba enviar un formulario pueda hacerlo. Dependiendo de la complejidad del sistema entrará en la lista de mejoras.
- ❖ **ARO6.1** Los informes que se generan no son aceptados por el área de Marca y Comunicación. Desde el primer momento que se realiza el informe se estará en contacto con el citado departamento para diseñar cómo se formatea la visualización de datos. Para el desarrollo inicial el objetivo es obtener el mismo formato de informe que se obtiene en la aplicación de escritorio.



1.3 Enfoque y método seguido.

1.3.1 Estado del arte¹.

Para la realización del proyecto se han visitado páginas web y se han visualizado vídeos que daban una imagen del estado de los lenguajes, tecnologías relacionadas con la web más usadas y su evolución durante el presente año. El estudio más completo y formal provenía de stackoverflow [26], el cual estaba referenciado en muchos de los artículos y vídeos revisados, por lo tanto, se considera que es una buena guía para este apartado, ya que nace de una encuesta a 90.000 profesionales de todo el mundo. Los gráficos aportados en este apartado pertenecen a ese mismo informe.

El informe de stackoverflow trata multitud de aspectos, que van desde el perfil del desarrollador hasta cuál es la dedicación de los desarrolladores al negocio del software. Uno de los aspectos a los que se dedica una sección es a las tecnologías de desarrollo web. En muchas de las referencias se observa si se suman los porcentajes que sobrepasan el cien por cien, puesto que la encuesta lo que refleja es el porcentaje de los encuestados que utilizan una o más tecnologías.

Most Popular Technologies

Programming, Scripting, and Markup Languages



Ilustración 1 - Tecnologías más populares

conlleve su adquisición por parte de Oracle que está licenciando ciertos usos de la máquina virtual [35].

En el informe se hace referencia a los principales *Frameworks* utilizados y casi la mitad de los desarrolladores usan JQuery. Otros marcos muy utilizados gracias al apoyo que tienen detrás por la industria son React.js que es utilizado por Google para sus productos y Angular cuyo uno de los grandes valedores es FaceBook [30]. Indicar que cada

Se visualizan las cinco tecnologías más usadas por los encuestados de un listado que incluye más de una veintena de tecnologías distintas. JavaScript, HTML/CSS y Python están ligados con la programación Web, y SQL es un lenguaje muy popular para interrogar a bases de datos. Según el informe, Python está ganando popularidad, aunque es un lenguaje que tiene veteranía, sobrepasa a Java por el lastre que

Web Frameworks

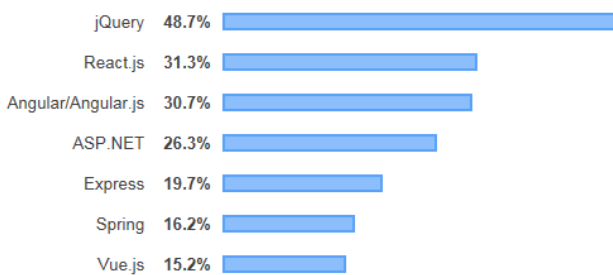


Ilustración 2 - Principales Web Frameworks

vez hay más desarrolladores de React. En el listado podemos observar que Vue, un *framework* con una filosofía que se asemeja a la de React.js [28], está en el séptimo puesto, debido a su juventud, pero va ganando adeptos por su facilidad de aprendizaje.

¹ Fuentes (incluidos gráficos) para este punto referenciadas en la Bibliografía [25] al [35].



Databases

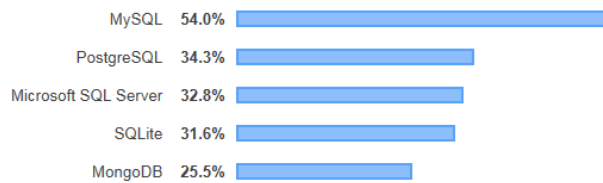


Ilustración 3 - Principales Bases de datos para el desarrollo web.

Otro punto muy importante era la elección de la base de datos. Al día de hoy las bases de datos relacionales siguen estando entre las favoritas de los desarrolladores. MariaDB es una base de datos que está emergiendo, pero aún está en séptimo lugar del ranking con un 16.5%. La compra de MySQL por parte de Oracle [35] hace prever que más desarrolladores vayan usando MariaDB que es un desarrollo en la que participan parte de la comunidad que antes desarrollaba MySQL [35].

Most Popular Development Environments



Ilustración 4 - Entornos de desarrollo más populares.

En cuanto al entorno de desarrollo Visual Studio Code, es usado por casi la mitad de los desarrolladores gracias a la gran comunidad de programadores que están dando soporte de manera constante aportando nuevos *plugins* que ayudan a las tareas de desarrollo.

Aunque el informe trata muchos más aspectos, se ha considerado que estos son los aspectos más relevantes que podían ayudar en la búsqueda de los diferentes apartados que son candidatos a integrar el proyecto y, por tanto, era necesario una visión global de cuáles son las tendencias actuales en el mundo del desarrollo web.

Se han observado otras tendencias que se están abriendo paso como el uso de contenedores (Docker, Open Container Initiative...), Blockchain... pero hay que considerar la falta de madurez suficiente por mi parte, en el campo del desarrollo web, para incorporarlas en esta fase del proyecto dejando estas tecnologías para futuras ampliaciones de la aplicación.

Remarcar que en la mayoría de los documentos y vídeos que tratan sobre las tecnologías de desarrollo web, sus autores, remarcan que no se trata de escoger la más usada o intentar trabajar con todos los *frameworks*, sino que se utilice aquel que mejor se ajuste a las necesidades del proyecto e intentar especializarse en el mismo [32] [33] [34].

Para mi proyecto tenía claro que debía utilizar tecnologías con las que tuviera una experiencia previa, aunque fuera mínima, me permitía establecer una temporización más realista, a pesar de las dificultades que pudiera encontrarme con lenguajes en los que no soy experto.

En el informe podemos observar que las tecnologías principales a usar para el proyecto se encuentran entre las más usadas por los programadores, aunque hay alguna de ellas que ha perdido popularidad como PHP, que es usada solo por un 26.4%, pero cuya principal utilización está pensada como interfaz en el lado del servidor entre la base de datos y la aplicación web.

En cuando a *Frameworks* señalados en el informe me decanté por el uso de Vue por varios motivos [27] [28]. El principal se debe a que su curva de aprendizaje es la menos pronunciada de los tres (Vue, Angular y React) [25]. Vue no necesita para su uso más que una etiqueta en HTML que haga referencia a la localización de la librería y se integra bien si se usa de forma híbrida con otras tecnologías.



1.3.2 Enfoque y método seguido.

Por parte del área de sistemas de información se tenía claro el camino hacia las tecnologías web de manera que se abandonara el uso de aplicaciones de escritorio, lo que permitiría adaptar nuestra forma de trabajo a la nueva realidad de la institución, donde cada vez más existe una descentralización dentro de la provincia en la que existen puntos de trabajo que no están dentro de la red informática. Para la realización de este camino aproveché toda la experiencia invertida en las aplicaciones de escritorio.

La dificultad de contar con presupuesto conllevó la utilización de recursos disponibles como el *hosting* actual, el uso de *frameworks*, lenguajes de programación, aplicaciones... *open source* o sin coste, junto con el uso de estándares abiertos.

A la hora de afrontar el proyecto se ha tenido en cuenta la exigencia en los plazos para ofrecer un sistema funcional de manera que no se quede en una mera prueba de concepto, esto obliga a darle al proyecto un enfoque lo más práctico posible que permita -en el momento de la entrega de esta memoria- conseguir un nivel funcional completo, y, por este motivo, el uso de metodologías ágiles [1] centrándonos en el uso de XP Programación Extrema [2] [3] [4] es una buena decisión. Se ha priorizado que la aplicación funcione, frente a la excelencia en la implementación de algunas tecnologías, dejando esta evolución del código una vez lograda la funcionalidad mínima comprometida.

Hay que ser consciente de que las metodologías ágiles están pensadas para el desarrollo en equipos con un *feedback* continuo, por lo que tenemos claro que no me puedo ajustar a todos los preceptos, seleccionando aquellas partes que se adapten a nuestra particularidad para lograr el éxito del proyecto.

A pesar de usar metodologías ágiles no debemos dejar de lado la documentación propia del proyecto, documentos y trabajos específicos para un trabajo final de grado que requieren un esfuerzo extra en su redacción por la dificultad que supone siempre el lanzamiento inicial de un proyecto y la decisión de las distintas tecnologías que se desean usar.

Se ha realizado una descripción a alto nivel del propósito de la aplicación y los objetivos a nivel general, específicos, los resultados esperados para cada objetivo y sus riesgos detectados. Se ha diseñado una temporización que permita lograr los hitos marcados para el desarrollo de la aplicación y se han expuesto los recursos necesarios para realizar los distintos trabajos que lleven al cumplimiento de los objetivos.

Se ha creado una tabla que permita conocer las distintas incidencias que aparecen durante el desarrollo, de forma que se establecen las acciones necesarias para conocer su estado en todo momento y su resolución. Se han establecido los riesgos que podemos encontrar en la realización de la aplicación para tomarlos en consideración en las distintas fases del desarrollo.

Se ha acordado con el consultor llevar a cabo seguimientos intermedios entre los distintos hitos de entrega para que esté informado de los avances y recibir *feedback* sobre aquellos aspectos en los que se estén produciendo desviaciones por exceso o defecto, además de las observaciones sobre mejoras y posibles riesgos que impidan la buena consecución del proyecto.



Para realizar la programación, pruebas y desarrollo se ha establecido trabajar en modo local y subir al hosting versiones que sean funcionales para depurarlas y ponerlas en funcionamiento. Se ha seguido el modelo de vida de un desarrollo bajo la premisa comentada del uso de XP.



Ilustración 6 - Ciclo de vida XP.

Como elemento central de desarrollo se ha utilizado la herramienta Visual Studio Code, que es una de las herramientas de edición de texto más utilizadas en la actualidad para el desarrollo, gracias a su constante mantenimiento y una enorme cantidad de extensiones, para una gran variedad de lenguajes que le dan una enorme versatilidad. El editor soporta el trabajo con todos los lenguajes, *frameworks*... que he utilizado durante el desarrollo y permite con una de sus extensiones actualizar el contenido visual del navegador de forma automatizada conforme se guardan los cambios.

La aplicación se ha pensado en principio para utilizarse con Google Chrome en sus últimas versiones y, al utilizar *frameworks* como Bootstrap o Vue, no debería existir problema en el uso en otros navegadores como Opera o Firefox, pero no se garantiza al 100% la funcionalidad. La elección de Google Chrome se ha realizado porque es uno de los dos navegadores 'oficiales' dentro de Cruz Roja. Internet Explorer se usa para las aplicaciones -pero ya no es considerado por su fabricante como un navegador sino como una "solución de compatibilidad" [23] sin soporte para nuevos estándares- Google Chrome se usa para trabajar todo el entorno 365 de Microsoft. sí tiene constantes actualizaciones por parte de su fabricante y ofrece herramientas para la depuración del código que se genera.

Para el desarrollo también se ha instalado en modo local XAMPP, que es un servidor que ofrece servicios de web, base de datos, php... sin necesidad de realizar configuración y siendo operativo desde el primer momento como una réplica del *hosting* que precisaba para desarrollar y probar de forma rápida el código de la aplicación.

El documento de memoria se ha constituido como el documento de trabajo central donde reflejar todos los aspectos relacionado con el proyecto. Este documento se ha pensado como un documento de consulta y/o resumen del trabajo, cuyo uso ha experimentado las mismas evoluciones que el propio proyecto transformando, adaptando y actualizando su contenido de manera simultánea al desarrollo de la aplicación.

Se ha facilitado a varios usuarios la dirección donde se subían partes de la aplicación para probar su funcionamiento y donde recibir un *feedback* sobre fallas y posibles mejoras que se realicen en un futuro si no entran dentro de las funcionalidades iniciales planteadas para el proyecto.

Mientras los usuarios probaban la aplicación y señalaban los fallos o posibles mejoras que se iban encontrando, se decidió abordar la memoria y la presentación al completar en gran medida el ciclo de producción. Con esto podía abordar ciertos apartados de la memoria que hasta este momento había eludido e ir preparando un guion con los asuntos a tratar en la exposición del trabajo.

Para la realización del vídeo se estudiaron varios programas de edición y se decidió ir grabando la presentación con el guion preparado, creando a continuación presentaciones que apoyarán la exposición de forma que clarificará ciertos contenidos difíciles de entender sin un apoyo visual.



1.4. Planificación del trabajo.

El proyecto se ha dividido en cuatro fases consecutivas, por lo que es necesario que todos los trabajos de una fase estén terminados para poder afrontar la siguiente. Existe una fase que es transversal a todo el proyecto que es la redacción de la memoria, que aglutina la documentación de las distintas partes que componen el proceso de creación de la aplicación. De esta forma la propia memoria se convierte en un documento de referencia para el desarrollo.

1.4.1. Hitos.

Hito 1: Definición inicial del proyecto.

Se define el proyecto y se expone para su aprobación. Se realiza el estudio de las necesidades revisando las aplicaciones existentes y cuáles eran las necesidades actuales del proyecto. Se realiza una descripción de la funcionalidad del sistema y bajo qué premisas se acepta su funcionamiento. Se establecen los objetivos generales y específicos que había que cumplir y una temporalización mostrada en el punto 1.4.2. Tabla 1.

Hito 2: Funcionalidad creación de usuario y plantillas finalizada.

Se realizan los diseños generales que afectaban a toda la aplicación y los específicos para cumplir con el segundo hito. Entre estos diseños cabe destacar la descripción de las tablas de la base de datos con sus campos y el diseño de entidad-relación, los flujos de iteración en situación ideal para las funcionalidades de este hito, el diseño a mano alzada de todas las pantallas y el diseño formal de las pantallas afectadas por este hito.

Se crean los documentos de seguimiento del proyecto, gestión de incidencias y gestión de riesgos. Se genera en la memoria los apartados donde aglutinar la información que se genera referida al proyecto. Se añade parte del documento que conforma el hito 1, y se completa los apartados restantes que forman parte del diseño del proyecto.

Se crea un subdominio en el proveedor del hosting donde se da de alta la base de datos con el diseño realizado. A lo largo de la creación de las primeras pantallas de la aplicación se observa la necesidad de realizar algún rediseño que mejora el funcionamiento interno del código.

Se instalan los entornos y programas necesarios para la creación de la aplicación, se programa las primeras dos pantallas diseñadas y se deja para un momento posterior la autenticación de usuarios para permitir que se prueben las funcionalidades de la aplicación por colaboradores externos.

Se sube el código fuente a un repositorio compartido en la nube para la inspección, por parte del tutor, en cualquier momento a la vez que se sube al hosting pantallas funcionales para ser comprobadas y obtener un *feedback* sobre posibles fallas y mejoras en futuras versiones.



Hito 3: Funcionalidad creación de operaciones, formulario...

En este tercer hito me centré en finalizar una versión operativa de la aplicación de manera que cubriera todo el ciclo. Para cerrar dicho ciclo se genera la pantalla que controla la operación, hasta la visualización de los formularios rellenos, donde se esperó hasta que estuviera operativa la pantalla donde se introducen los formularios para finalizar esta parte.

Una vez realizadas estas pantallas se genera la realización de informes de muestra que parte de la selección de una operación o una plantilla. Para ambos informes se muestran campos para acotar la búsqueda de información.

Una vez finalizada esta parte se sube la versión al hosting donde se realizan más pruebas y se pasa un perfil con permiso completo de acceso a la aplicación a varios usuarios para que realicen pruebas, detecten problemas, mejoras... Se crea una infografía que ayude a los usuarios a dar los pasos lógicos en el uso de la aplicación.

Se actualiza la memoria incluyendo las partes del proyecto que se han implementado durante esta fase como son los algoritmos de iteración y la representación visual de operaciones, formularios e informes, además de realizar las modificaciones y actualizaciones necesarias que afectaban a otras partes que se rediseñan como, por ejemplo, la definición de la base de datos para ajustar cambios de diseños internos del producto.

Para la presentación del trabajo se inicia la creación de una presentación que sirva de guía a la grabación de la exposición.

Hito 4: Presentación proyecto, memoria, últimos flecos.

Para preparar la presentación del proyecto se aprende el manejo de HitFilm Express² que permite, en su versión gratuita, una gran cantidad de opciones para la edición de vídeo y audio.

Se graban, por un lado, los vídeos donde se explica cada apartado del trabajo realizado y, por otro lado, presentaciones en Microsoft Power Point que acompañen e ilustren las distintas exposiciones de forma que ayudan al entendimiento de la propia exposición. Se utiliza monosnap³ para grabar las partes de la presentación que mostraba un ejemplo de uso de la aplicación a través del ordenador.

Se finaliza la memoria rellorando todos los apartados que se habían obviado, se realizan correcciones ortográficas, léxicas...

Por último, se organiza y presenta toda la información que correspondía al proyecto para que estuviera a disposición del tribunal evaluador.

A pesar de haberlo intentado ha sido imposible seguir avanzando con los informes básicos antes de la entrega del proyecto, por lo que aparecen en la relación de trabajos a realizar una vez finalizada la presentación del proyecto.

² <https://fxhome.com/hitfilm-express>

³ <https://monosnap.com/>



1.4.2. Temporización.

Se muestra una temporización del proyecto planificada y aceptada en el hito1.

Nombre.	Fecha Inicio.	Fecha Fin.
HITO 1.	18/09/2019	01/10/2019
Definición y aceptación del proyecto.	18/09/2019	30/09/2019
HITO 2	01/10/2019	05/11/2019
Definición base de datos.	01/10/2019	09/10/2019
Selección plataforma web y base de datos.	01/10/2019	14/10/2019
Crear página manejo roles usuarios.	10/10/2019	21/10/2019
Crear página plantillas.	22/10/2019	04/11/2019
HITO 3	06/11/2019	20/12/2019
Página operaciones.	06/11/2019	26/11/2019
Página formularios.	27/11/2019	10/12/2019
Página informes.	11/12/2019	01/01/2020
HITO 4	01/10/2019	09/01/2020
Generar memoria.	01/10/2019	08/01/2020

Tabla 1 - Temporización del proyecto.

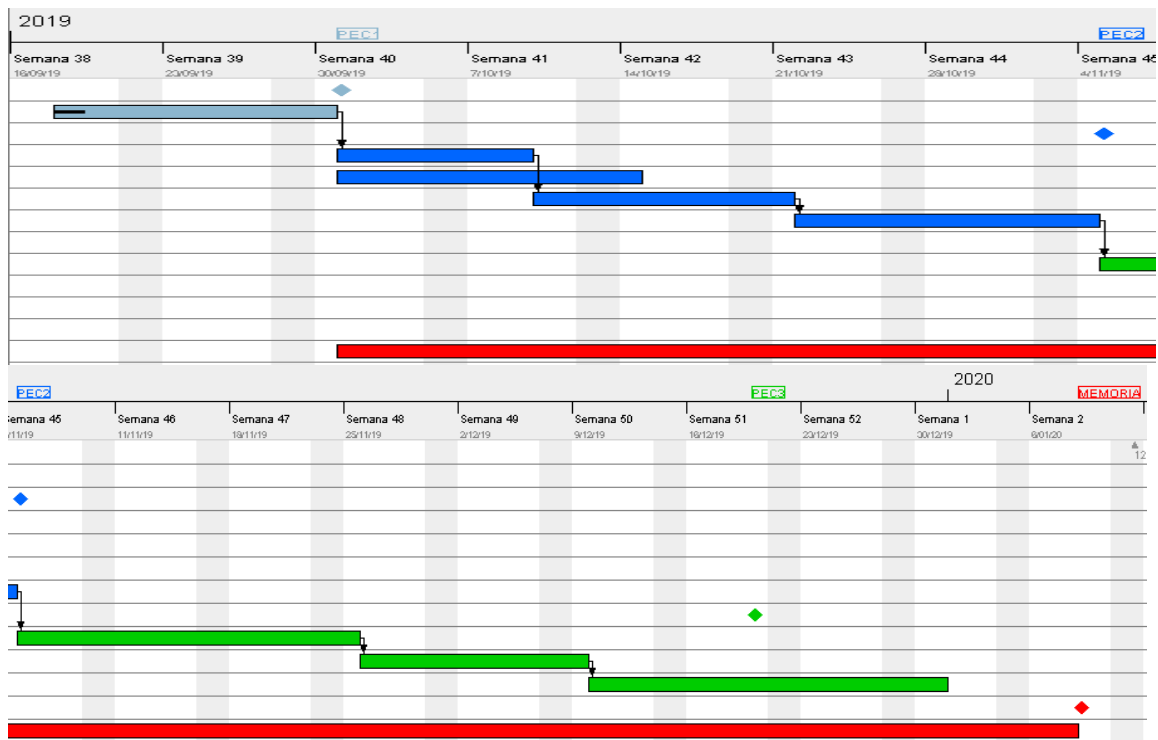



Ilustración 7 - Diagrama de Gantt del proyecto.



1.4.3. Recursos utilizados.

- **Servidores Web.**

Se decidió utilizar los servicios del proveedor de servicios web⁴ que tiene Cruz Roja Navarra al que se le explicó las necesidades del proyecto. El proveedor facilitó una consola de administración del *hosting* para implementar los recursos que se necesitarán. Esta decisión me permitió centrar en la creación de las páginas web y ahorrar un tiempo esencial en la instalación, configuración y mantenimiento de un servidor web.

 Para configurar el subdominio se utilizó la herramienta 'plesk⁵ web host edition' que es un panel de control centralizado para controlar el subdominio (aplicaciones, monitorización, actualizaciones...), además contiene enlaces a las aplicaciones utilizadas como el servidor de base de datos MySQL/MariaDB y la consola de administración phpMyAdmin.

- **Aplicaciones.**



Se ha utilizado la herramienta draw.io⁶ que es una herramienta online y/o escritorio para el diseño de distintos diagramas permitiendo utilizar toda la simbología que se usa en UML. La herramienta la he usado para mostrar los flujos de iteración, el diseño formal de las pantallas y el caso general de uso.



Se utilizó phpMyAdmin⁷ que es un software gratuito escrito en PHP para administrar y manejar bases de datos que funcionan bajo MySQL y MariaDB. Permite realizar operaciones bajo la estructura de la base de datos (tablas, columnas, relaciones, permisos...) y ejecutar instrucciones SQL. He utilizado de manera complementaria la utilidad que tiene para mostrar el diseño de la base de datos para la imagen de Entidad-Relación, que muestra de manera gráfica toda la estructura de las tablas, la relación entre sus valores, claves...



Como servidor de base de datos se ha utilizado MariaDB⁸ que está soportado por el proveedor de servicios de Internet y da soporte para toda la funcionalidad necesaria para la aplicación.



Para la programación del código web he decidido utilizar el entorno de desarrollo Visual Studio Code⁹ que es un editor de código fuente desarrollado por Microsoft para Windows, compatible con varios lenguajes de programación y se adapta a las necesidades del proyecto.



GitHub¹⁰ como repositorio en la nube permite el control de versiones para guardar los avances que se van realizando en el proyecto y consultar las diferencias entre distintas versiones. Ofrece en estos momentos, de forma privada, un lugar donde el consultor tenga libre acceso a los ficheros del desarrollo del proyecto para su inspección.

⁴ <https://www.masbytes.es>

⁵ <https://www.plesk.com>

⁶ <https://www.draw.io/?splash=0>

⁷ <https://www.phpmyadmin.net>

⁸ <https://mariadb.org>

⁹ <https://code.visualstudio.com>

¹⁰ <https://github.com>



XAMPP¹¹ como servidor local para probar todo el entorno web que entre otros servicios ofrece un servidor web, servidor de base de datos MySQL, la consola de administración phpMyAdmin, PHP....



Google Chrome¹² para las pruebas de funcionamiento, seguimiento y depuración de código, ya que el propio navegador trae una gran cantidad de herramientas que apoyan el desarrollo web.

- **Lenguajes de Programación y Frameworks.**



Se ha decidido el uso del lenguaje de marcado HTML5¹³ para conformar el esqueleto de las páginas que componen la aplicación al marcar cómo se debe formatear el documento para Internet.



Bootstrap 4¹⁴ es un *framework* CSS que permite controlar el aspecto visual y la distribución de los distintos elementos en la página web. Bootstrap permite al programador simplificar el trabajo sin poseer conocimientos específicos de CSS, lo que ahorra tiempo en la creación de las aplicaciones.



CSS3¹⁵ independiza el contenido de una página web de su presentación visual y se utilizó para aquellos aspectos visuales que no se controlaba a través de Bootstrap.



Se ha utilizado Vue¹⁵ que es un *framework open source* basado en JavaScript para construir los interfaces de usuario, hace que la aplicación sea reactiva, permite crear componentes para encapsular código, facilitar su reutilización...



JavaScript⁹ es un lenguaje de programación utilizado para crear páginas web dinámicas de manera que estas incorporen animaciones, efectos y otras características. Es un lenguaje que se usa en el lado del cliente de manera que los navegadores interpretan sus instrucciones para ofrecer funcionalidades.



PHP¹⁶ es un lenguaje libre, de propósito general, que se utiliza en el lado del servidor encargado en nuestra aplicación para la comunicación con la base de datos. El lenguaje admite incrustar HTML y es utilizado para la creación de páginas web dinámicas.



SQL¹⁷ como lenguaje standard de interrogación para base de datos relacionales como MySQL y MariaDB. SQL permite desde crear la estructura de una base de datos hasta la consulta, actualización, eliminación o inserción de datos de una forma sencilla.



JQuery¹⁸ como lenguaje de programación dinámico para el DOM en la parte relacionada con las matrices en formularios, operaciones e informes. JQuery nos da completo control sobre todos los aspectos de presentación y edición de las mismas, que no obtenemos con vue.

¹¹ <https://www.apachefriends.org/es/index.html>

¹² https://www.google.com/intl/es_es/chrome/

¹³ <https://www.w3c.es>

¹⁴ <https://getbootstrap.com/>

¹⁵ <https://vuejs.org>

¹⁶ <https://www.php.net>

¹⁷ <https://www.iso.org/home.html>, <https://www.ansi.org/>

¹⁸ <https://jquery.com>



1.4.4. Gestión de incidencias.

Durante la realización del proyecto se produjeron situaciones no previstas que se debieron afrontar, para lo que se estableció un seguimiento hasta su solución. [10] [11] [12]

Id.	Descripción	F. Ini.	Prio. ¹	Est. ²	Acciones	F. Fin.
1	Base datos da error con tablas.	07/10/19	Media	Cer.	Salvar la información de las tablas no corrompidas. Eliminar y crear de nuevo la base de datos. Importar base de datos. Salvar nueva definición base de datos.	08/10/19
2	El subdominio creado no es accesible vía web.	08/10/19	Media	Cer.	Envío e-mail al proveedor que modificó sus servicios DNS para que resolviera la dirección.	09/10/19
3	Hay problemas para empezar con la programación web.	11/10/10	Alta	Cer	Se revisan varios cursos <i>on-line</i> para afianzar conceptos de programación.	18/10/19
4	Al subir las primeras versiones de la página no funciona la conexión con la base de datos.	18/10/19	Alta	Cer.	Se comunica con el proveedor el cual indica los ficheros a revisar. Se encuentra problemas con contraseñas.	19/10/19
5	Pruebas realizadas con usuario, la diferencia de azul en los botones no es suficiente para intuir su estado.	21/10/19	Baja	Cer.	Algunos botones que no deban interactuar solo se visualizan cuando se necesitan.	06/12/19
6	Dificultades con valores con acentos.	25/10/19	Alta	Cer.	Se modifica el conjunto de caracteres en la web.	28/10/19
7	Complejidad en el uso en elementos que componían ítems y niveles.	28/10/19	Alta	Cer.	Se redefine la base de datos y se unifica el código simplificando su uso.	04/11/19
8	Búsqueda de datos en subformularios para informes.	15/11/19	Media	Cer.	Se estudia la consulta y se redefine su uso.	03/12/19

Tabla 2 - Gestión de incidencias.

1 Alta, Media, Baja. 2 Pendiente, En curso, Cerrado.



1.5. Informes de seguimiento.

Durante el desarrollo del proyecto una vez finalizado el hito 1, se acordó con el consultor el envío periódico del estado. Para dejar constancia de estos envíos se ha rellenado los siguientes informes [11] [12].

Proyecto: Sistema de Recogida de Datos Cuantitativos en Operativos de Cruz Roja Navarra.	
Versión del documento: V1.0.	Fecha: 13 oct 2019.

1. Periodo del informe.
Del 1 de octubre al 13 de octubre de 2019.

2. Situación del proyecto.
<p>● Se ha aceptado la propuesta de proyecto con lo que se han definido y comenzado a rellenar los primeros apartados que compondrán la memoria.</p> <p>● Se han incluido diseños preliminares que afectan al segundo hito.</p> <p>● Se ha definido la base de datos, seleccionado la plataforma web y el servidor de base de datos.</p> <p>● Se ha creado el 'site' para la aplicación (https://siredacu.crnavarra.org) y subida la definición de la base de datos al servidor.</p>

3. Hitos conseguidos.		
Hito.	F. Prev.	F. Real.
Definición BD.	09/10/2019	09/10/2019
Sel. Plataforma Web y Base de datos.	14/10/2019	12/10/2019

4. Próximos hitos.	
Hito.	F. Prev.
Pantalla Roles.	21/10/2019
Pantalla Plantillas.	04/11/2019

5. Temas pendientes.	
Fecha.	Tema.



Proyecto: Sistema de Recogida de Datos Cuantitativos en Operativos de Cruz Roja Navarra.	
Versión del documento: V1.0.	Fecha: 27 oct 2019.

1. Periodo del informe.

Del 14 de octubre al 27 de octubre de 2019.

●	2. Situación del proyecto.
●	Se ha generado la pantalla para la creación y asignación de roles a usuarios. Se está generando la pantalla de trabajo con plantillas.
●	Se crear un repositorio en línea con GIT (https://github.com/fmartinezrod/siredacu.git).

3. Hitos conseguidos.

Hito.	F. Prev.	F. Real.
Crear pág. Usuario.	21/10/19	20/10/19

4. Próximos hitos.

Hito.	F. Prev.
Crear Pág. Plantillas.	04/11/19
Entrega Pec 2.	05/11/19

5. Temas pendientes.

Fecha.	Tema.
20/10/19	Generar apartado estado del arte.
20/10/19	Crear Glosario de términos propios de la aplicación y de la memoria.

Proyecto: Sistema de Recogida de Datos Cuantitativos en Operativos de Cruz Roja Navarra.	
Versión del documento: V1.0.	Fecha: 05 nov 2019.

1. Periodo del informe.

Del 28 de octubre al 05 de noviembre de 2019.

●	2. Situación del proyecto.
●	Se ha generado completamente la pantalla de trabajo con las plantillas.
●	Se ha incluido nuevo contenido en la memoria y ampliado secciones existentes con información actualizada.
	Se han agrupado dos tablas de la base de datos ya que se ha observado que el fin de las mismas era bastante similar, permitiendo la reutilización de código.

3. Hitos conseguidos.

Hito.	F. Prev.	F. Real.
Crear pág. Plantilla.	04/11/19	03/11/19
Entrega PEC 2.	05/11/19	05/11/19

4. Próximos hitos.

Hito.	F. Prev.
Crear Pág. Operaciones.	26/11/19

5. Temas pendientes.

Fecha.	Tema.
20/10/19	Generar apartado estado del arte.



PEC 2	Proyecto: Sistema de Recogida de Datos Cuantitativos en Operativos de Cruz Roja Navarra.	
	Versión del documento: V1.0.	Fecha: 05 nov 2019.

1. Periodo del informe.
Del 01 de octubre de 2019 al 05 de noviembre de 2019.

2. Situación del proyecto.
<p>Los avances en el desarrollo del proyecto van conforme lo previsto en la planificación. La base de datos ya está implementada en el sitio web creado dentro del <i>hosting</i> 'crnavarra.org'.</p> <p>Se han consultado los <i>frameworks</i> deseados para el proyecto y comprobado que pueden ofrecer todas las funcionalidades necesarias para el desarrollo del proyecto.</p> <p>Se ha comenzado con la memoria que contiene toda la información necesaria previa al comienzo del proyecto (objetivos, enfoque, planificación, gestión incidencias...), y, contiene los diseños necesarios para hacer frente a todos los trabajos hasta el hito de control.</p> <p>Se ha programado la pantalla de acceso, el control de los usuarios y plantillas conforme la planificación. Se ha dejado para un momento posterior la autenticación de usuarios de forma que se prueben todas las partes de la aplicación libremente.</p> <p>He habilitado almacenamiento privado en Git (https://github.com/fmartinezrod/siredacu.git) donde se sube constantemente las modificaciones y nuevas funcionalidades sobre el código que se va desarrollando, con el objetivo de ser consultado por el consultor, sirva como copia de seguridad y control de versiones.</p> <p>Se sube nuevas versiones de la aplicación a la dirección http://siredacu.crnavarra.org con la misión para varios usuarios de dar un <i>feedback</i> sobre la misma.</p>

3. Hitos conseguidos.		
Hito.	F. Prev.	F. Real.
Definición DB.	09/10/19	09/10/19
Selección plataforma.	14/10/19	12/10/19
Crear página usuarios.	21/10/19	20/10/19
Crear página plantillas.	04/11/19	03/11/19

4. Próximos hitos.	
Hito.	F. Prev.
Crear Página Operaciones.	26/11/19
Crear Página Formularios.	10/12/19
Crear Formularios.	01/01/20

5. Temas pendientes.	
Fecha.	Tema.
20/10/19	Crear apartado estado del arte.



Proyecto: Sistema de Recogida de Datos Cuantitativos en Operativos de Cruz Roja Navarra.	
Versión del documento: V1.0.	Fecha: 24 nov 2019.

1. Periodo del informe.

Del 06 de noviembre al 24 de noviembre de 2019.

2. Situación del proyecto.

Se ha generado la pantalla de operación exceptuando la parte de visualización de los formularios realizados ya que se ha decidido dejar esta parte hasta la finalización de la pantalla de la realización de formularios que aportará la información necesaria para este punto.

Se ha completado casi en su totalidad la pantalla de formularios en la que se ha finalizado la introducción de datos cuando el formulario no se han duplicado los datos. Falta el control cuando se duplica un formulario con los mismos datos seleccionadores indicados.

Se reformula el hito de finalización de las pantallas de operaciones por el cambio en la programación de ambas pantallas que ha pasado de ser secuencial a ser en paralelo, retrasando la fecha de finalización de operaciones a la misma que los formularios sin que este cambio afecte a la ejecución del proyecto.

3. Hitos conseguidos.

Hito.	F. Prev.	F. Real.

4. Próximos hitos.

Hito.	F. Prev.
Crear Pág. Operaciones.	26/11/19
Crear Pág. Formulario.	10/12/19

5. Temas pendientes.

Fecha.	Tema.
20/10/19	Generar apartado estado del arte.



Proyecto: Sistema de Recogida de Datos Cuantitativos en Operativos de Cruz Roja Navarra.	
Versión del documento: V1.0.	Fecha: 08 dic 2019.

1. Periodo del informe.
Del 25 de noviembre al 8 de diciembre de 2019.

2. Situación del proyecto.
<p>Se ha finalizado la parte que quedaba sobre la pantalla de operaciones con la posibilidad de controlar el estado de los formularios rellenados.</p> <p>Se ha finalizado igualmente la pantalla de introducción de formularios con las opciones para evitar duplicados y otros elementos.</p> <p>Se han generado informes de ejemplo que funcionan tanto introduciendo operaciones como plantillas, permitiendo la acotación de datos con los campos seleccionadores en el caso de las operaciones y la relación de 'tags' y operaciones en el caso de las plantillas.</p> <p>Se ha creado una infografía que explica de forma breve el orden lógico de trabajo con la aplicación.</p>

3. Hitos conseguidos.		
Hito.	F. Prev.	F. Real.
Crear Pág. Operaciones.	26/11/19	28/11/19
Crear Pág. Formularios.	10/12/19	05/12/19

4. Próximos hitos.	
Hito.	F. Prev.
Crear Pág. Informes.	01/01/20

5. Temas pendientes.	
Fecha.	Tema.
20/10/19	Generar apartado estado del arte.



Proyecto: Sistema de Recogida de Datos Cuantitativos en Operativos de Cruz Roja Navarra.	
Versión del documento: V1.0.	Fecha: 20 dic 2019.

1. Periodo del informe.
Del 9 de diciembre al 20 de diciembre de 2019.

2. Situación del proyecto.
<p>Se han corregido fallos detectados en la aplicación y se ha avanzado en la generación de la memoria. Se ha creado una presentación en PowerPoint que actuará de guion en la creación del video de presentación del proyecto.</p> <p>No se cierra el hito de informes por si tengo posibilidad de ampliarlo antes de la entrega final del proyecto.</p>

3. Hitos conseguidos.		
Hito.	F. Prev.	F. Real
Entrega PEC 3.	20/12/2019	20/12/2019
Generar informes.	01/01/2020	

4. Próximos hitos.	
Hito.	F. Prev.
Entrega memoria.	08/01/20
Entrega presentación.	08/01/20

5. Temas pendientes.	
Fecha.	Tema.



PEC 3	Proyecto: Sistema de Recogida de Datos Cuantitativos en Operativos de Cruz Roja Navarra.	
	Versión del documento: V1.0.	Fecha: 20 dic 2019.

1. Periodo del informe.
Del 06 de noviembre de 2019 al 20 de diciembre de 2019.

2. Situación del proyecto.
<p>● Se han generado las pantallas pendientes pertenecientes a la aplicación y un grupo básico de informes. Se ha implementado la parte de autenticación de usuario y adecuado el resto de pantallas para su aceptación.</p> <p>● Se han reformulado los hitos de operaciones y plantillas para hacer una implementación en paralelo ya que había dependencias entre ambas pantallas. La fecha de finalización de la segunda pantalla se ha respetado como fecha final para ambas pantallas.</p> <p>● Se ha avanzado con la memoria en todos sus apartados dejando para el siguiente hito una revisión de lo aportado, además de la corrección de otros aspectos como la gramática, la ortografía y la distribución visual.</p> <p>● Se ha actualizado el repositorio Git con las últimas modificaciones y puesto a disposición de usuarios la aplicación para realizar pruebas facilitando una infografía para explicar su uso.</p> <p>● A pesar de existir un conjunto básico de informes, no se da por finalizar el hito por si es posible la ampliación de los mismos antes de la entrega final, además de la implantación de algunas de las mejoras que se han detectado.</p> <p>● Se ha creado una presentación básica en PowerPoint a modo de guion para la realización del video.</p>

3. Hitos conseguidos.		
Hito.	F. Prev.	F. Real.
Crear Pág. Operaciones.	26/11/19	28/12/19
Crear Pág. Formularios.	10/12/20	05/12/19
Generar Informes.	01/01/20	

4. Próximos hitos.	
Hito.	F. Prev.
Entrega Memoria.	08/01/20
Presentación Proyecto.	08/01/20

5. Temas pendientes.	
Fecha.	Tema.



Proyecto: Sistema de Recogida de Datos Cuantitativos en Operativos de Cruz Roja Navarra.	
Versión del documento: V1.0.	Fecha: 31 dic 2019.

1. Periodo del informe.
Del 21 de diciembre al 31 de diciembre de 2019.

2. Situación del proyecto.
<p>● Se ha creado el video de presentación del proyecto.</p> <p>● Se ha finalizado la memoria, quedando una última revisión de aspectos formales como ortografía, gramática...</p>

3. Hitos conseguidos.		
Hito.	F. Prev.	F. Real.

4. Próximos hitos.	
Hito.	F. Prev.
Entrega memoria.	08/01/20
Entrega presentación	08/01/20

5. Temas pendientes.	
Fecha.	Tema.

Proyecto: Sistema de Recogida de Datos Cuantitativos en Operativos de Cruz Roja Navarra.	
Versión del documento: V1.0.	Fecha: 8 ene 2020.

1. Periodo del informe.
Del 1 de enero al 8 de enero de 2020.

2. Situación del proyecto.
<p>● Se finaliza la memoria.</p> <p>● Se prepara el contenido a presentar y se suben a los repositorios indicados los distintos contenidos para que estén disponibles para su evaluación.</p> <p>A pesar de intentar ampliar los informes básicos no ha existido tiempo material disponible por lo que se pasa al listado de trabajos futuros.</p>

3. Hitos conseguidos.		
Hito.	F. Prev.	F. Real.
Entrega memoria.	08/01/20	08/01/20
Entrega presentación.	08/01/20	08/01/20

4. Próximos hitos.	
Hito.	F. Prev.

5. Temas pendientes.	
Fecha.	Tema.



PEC 4	Proyecto: Sistema de Recogida de Datos Cuantitativos en Operativos de Cruz Roja Navarra.	
	Versión del documento: V1.0.	Fecha:08 ene 2020.

1. Periodo del informe.
Del 21 de diciembre de 2019 al 08 de enero de 2020.

●	2. Situación del proyecto.
●	Se crea el video de presentación del proyecto.
●	Se finalizan todos los apartados de la memoria.
	Se prepara el contenido a presentar y se suben a los repositorios indicados los distintos contenidos para que estén disponibles para su evaluación.
	A pesar de intentar ampliar los informes básicos no ha existido tiempo material disponible por lo que se pasa al listado de trabajos futuros.

3. Hitos conseguidos.		
Hito.	F. Prev.	F. Real.
Entrega memoria.	08/01/20	08/01/20
Entrega presentación.	08/01/20	08/01/20

4. Próximos hitos.	
Hito.	F. Prev.

5. Temas pendientes.	
Fecha.	Tema.



1.6. Breve resumen de productos obtenidos.

En esta sección se muestra una relación de las partes de la aplicación que se han obtenido.

- **Index.html:** Pantalla encargada de dar acceso a las distintas partes de la aplicación dependiendo del rol del usuario.
 - Inicio.js Contiene la lógica de la pantalla de inicio.
 - comprobarUsuarioPass.php Devuelve el nivel de acceso para un usuario.
 - comprobarUsuariold.php Mira si el usuario con dicho identificador existe.
- **Usuarios.html:** Permite definir qué roles va a abarcar un usuario y controlar su estado dentro de la aplicación.
 - usuarios.js. Contiene la lógica que controla los usuarios.
 - searchUsuarios.php Localización del usuario.
 - insertUsuario.php Creación del usuario.
 - deleteUsuario.php Eliminación del usuario.
 - updateUsuario.php Actualización de los datos del usuario.
- **Plantillas.html:** Definimos los datos para recoger y como están repartidos. Se asignan términos que ayuden a realizar agrupaciones de estas plantillas cuando se realicen informes.
 - plantillas.js Contiene la lógica que controla las plantilla.
 - searchPlantilla.php Búsqueda de la plantilla.
 - insertPlantilla.php Creación de la plantilla.
 - deltePlantilla.php Eliminación de la plantilla.
 - updatePlantilla.php Actualización de datos de la plantilla.
 - searchPlanForm.php Devolvemos los formularios de una operación.
 - getItems.php Obtener lo ítems de la plantilla.
 - insertItem.php Inserción de un ítem en la base de datos.
 - deleteItem.php Eliminación de un ítem en la base de datos.
 - subirltem.php Reordenar los ítems. Elevación de posición de un ítem.
 - bajarItem.php Reordenar los ítems. Descenso de posición de un ítem.
 - getPlanTags.php Obtener palabras de agrupación adjuntadas a la plantilla.
 - insertPlanTag.php Insertar 'tags' a la plantilla.
 - deletePlanTag.php Eliminar 'tags' de la plantilla.



- **Operaciones.html:** Crear una operación que contiene una plantilla y decidir que valores van a identificar de forma inequívoca a cada formulario para la operación.
 - Operaciones.js Contiene la lógica de las operaciones.
 - searchOperacion.php Búsqueda de operaciones.
 - insertOperacion.php Creación de una operación.
 - selectPlantillasActivas.php Selecciona las plantillas activas.
 - deleteOperacion.php Eliminación de una operación.
 - updateOperacion.php Actualización de los datos de la operación.
 - getOpeCampos.php Obtenemos los campos que identificarán un formulario.
 - insertCampo.php Crea un campo que identificará un formulario.
 - deleteCampo.php Elimina un campo que identificará un formulario.
 - subirCampo.php Sube el elemento seleccionado en el orden de los campos.
 - bajarCampo.php Baja el elemento seleccionado en el orden de los campos.
 - getOpeValores.php Recoge los valores que componen cada campo de selección.
 - insertValor.php Inserta un valor que corresponde a un campo de selección.
 - deleteValor.php Elimina un valor que corresponde a un campo de selección.
 - subirValor.php Reordena el orden en el que se deben visualizar los valores dentro de un campo de selección aumentando su posición.
 - bajarValor.php Reordena el orden en el que se deben visualizar los valores dentro de un campo de selección disminuyendo su posición.
 - getOpeTags.php Recoge los 'tags' que se utilizarán para una operación.
 - insertOpeTag.php Guarda los 'tags' que se utilizarán para la operación.
 - deleteOpeTag.php Elimina un 'tags' que se utilizarán para la operación.
 - copyTags.php Copia los 'tags' predefinidos en una plantilla a la operación.
 - getFormularios.php Obtiene la relación de formularios rellenados para la operación.
 - getFormValores.php Obtenemos los valores introducido en los formularios.
 - getItemsPlan.php Obtenemos los ítems que conforman la plantilla seleccionada para la operación.
 - UpdateEstadoFormulario.php Activar / desactivar la edición de un formulario.
 - deleteFormulario.php Marcar un formulario rellenado como eliminado.
 - insertValorFormulario.php. Inserta un valor en un formulario existente.



- **Formularios.html:** Rellenar los formularios de recogida de datos para una determinada operación.
 - Formularios.js Contiene la lógica para rellenar formularios.
 - selectOperacionesActivas.php Muestra las operaciones que están activas.
 - getOpeCampos.php Obtenemos los campos que identificará un formulario.
 - getOpeValores.php Recoge los datos que componen cada campo de selección.
 - getPlantilla.php Obtiene los datos de la plantilla para el formulario.
 - comprobarPass.php Comprueba la palabra de aceptación del formulario.
 - searchValCamFor.php Obtenemos los valores del formulario.
 - getPermisoModificar.php Indica si se puede modificar el contenido del formulario.
 - insertFormulario.php Guarda el formulario.
 - insertValorCampo.php Guardan los campos seleccionados.
 - insertValorFormulario.php Guarda los valores introducidos en el formulario.
 - deleteTablaFormulario.php Elimina los datos rellenados en el formulario.

- **Informes.html:** Se obtienen informes referidos a las operaciones realizadas o las operaciones que usan una plantilla determinada.
 - Informes.js Contiene la lógica para mostrar informes.
 - selectOperaciones.php Muestra la relación de operaciones para los informes.
 - selectPlantillas.php Muestra la relación de plantillas para los informes.
 - getOpeCampos.php Obtenemos campos de selección de operaciones.
 - getOpeValores.php Obtenemos los valores que componen cada campo para acotar la búsqueda de datos para el informe.
 - getPlantilla.php Obtenemos los datos de la plantilla.
 - getOpePlan.php Obtenemos las operaciones que usan una determinada plantilla.
 - getOpeTagsPlan.php Obtenemos los 'tags' que están en las operaciones seleccionadas.
 - infOperaciones Obtenemos los formularios de las operaciones seleccionadas.
 - infPlanOperaciones.php Obtenemos los valores almacenados para los formularios seleccionados.



1.7. Breve descripción del resto de capítulos de la memoria.

En los siguientes apartados se muestra una descripción en profundidad de los trabajos realizados en la creación de la aplicación que abarcan el diseño, las herramientas, medios utilizados, las pruebas que tienen que pasar las distintas partes de la aplicación para aceptarlas, las conclusiones sobre los trabajos realizados, dos apartados técnicos como son el glosario de términos técnicos y los referidos a los términos propios de la aplicación y una bibliografía con las principales fuentes de consulta utilizadas para el desarrollo de la aplicación.

Se realiza una descripción rápida de cada uno de los apartados con sus principales componentes.

- **Diseños.**

En este apartado se abordan los distintos diseños empleados para plasmar los modelos necesarios para desarrollar la aplicación. El uso de diseños ha facilitado la comprobación de las distintas funcionalidades para encontrar en una fase inicial posibles fallas e incongruencias que en fases avanzadas del proyecto hubieran puesto en riesgo el éxito del mismo.

- Diseño de casos de uso.

Se ha insertado en primer lugar un diseño de diagramas de uso a través del lenguaje de modelado unificado (UML) [5] [6] [7] que muestra a golpe de vista cuáles son los principales roles que intervienen en la aplicación junto con sus acciones.

- Diseño de la base de datos.

Se describe con precisión la estructura que soporta la información que se requiere para la aplicación. Se han definido las tablas, sus campos, su tipología y la descripción de la información que recoge. Se ha procedido a la normalización de los datos, definido campos, claves principales, claves foráneas...

- Flujos de Interacción.

Haciendo uso nuevamente del Lenguaje de Modelado Unificado (UML) [5] [6] se utilizan los flujos de iteración para mostrar en detalle cómo sería la creación de cada uno de los procesos que se realizan en la aplicación y que se corresponden a su vez con las pantallas principales de las mismas, dando una visión general de cómo funciona el sistema.

- Representación visual de pantalla a alto nivel.

Se Muestra el proceso de diseño de las pantallas, desde el primer boceto hasta su implementación final, para evidenciar el proceso creativo de las pantallas.



- **Implementación.**

En este apartado se aborda cómo se ha realizado la implementación de los componentes de la aplicación, justificando las decisiones de uso para cada una de ellos.

- Servicios Web.

Explicación de cuál ha sido la solución utilizada para implementar y configurar los servicios web que dan soportes al desarrollo y el proceso para su puesta en marcha.

- Bases de datos.

Se explica cómo están descritos los principales datos que componen la base de datos, cómo se comportan dentro de las tablas y se muestra un diseño de la Entidad-Relación de la base de datos.

- Dispositivos Utilizados.

Explicación de los dispositivos donde se ha desarrollado y probado el software y la estructura de ficheros generada.

- **Pruebas.**

En este apartado se muestran las pruebas realizadas para que cada pantalla de la aplicación se pudiera dar por válida.

- **Aceptación del proyecto.**

Al inicio del proyecto se estableció que la aplicación sería aceptada si servía para la introducción de las plantillas de atenciones que se realizaban durante el encierro en las fiestas de San Fermín y el posterior envío del resultado al área de Marca y Comunicación para su difusión. En este apartado se muestran los pasos que hay que realizar en la aplicación para hacer el citado trabajo.

- **Conclusiones.**

Se muestra una evaluación de la experiencia en este semestre en el que se abordan los principales resultados del trabajo, junto con las dificultades y los próximos pasos que debería dar para hacer crecer el desarrollo realizado.

- **Anexos.**

Se aportan varios documentos que completan la memoria del trabajo con la infografía, una tabla con los futuros trabajos y una descripción de los ficheros que componen el Trabajo Final de Grado y cómo deben ser interpretados o utilizados.



2. DISEÑOS.

2.1. Diseño de diagramas.

A través de los casos de uso [5] [6] [7] se muestra a alto nivel cómo interactúan los intervinientes con el sistema para ofrecer una visión inicial de cómo funciona.

El diagrama inicial del sistema que afecta a la recogida de datos ofrece una perspectiva general de los diferentes apartados que deben comprender el desarrollo y los roles de los usuarios con su iteración con la aplicación.

Se ha optado por la realización de un caso de uso general frente a varios específicos porque más adelante se detallan con precisión cómo funcionan los distintos apartados de la aplicación.

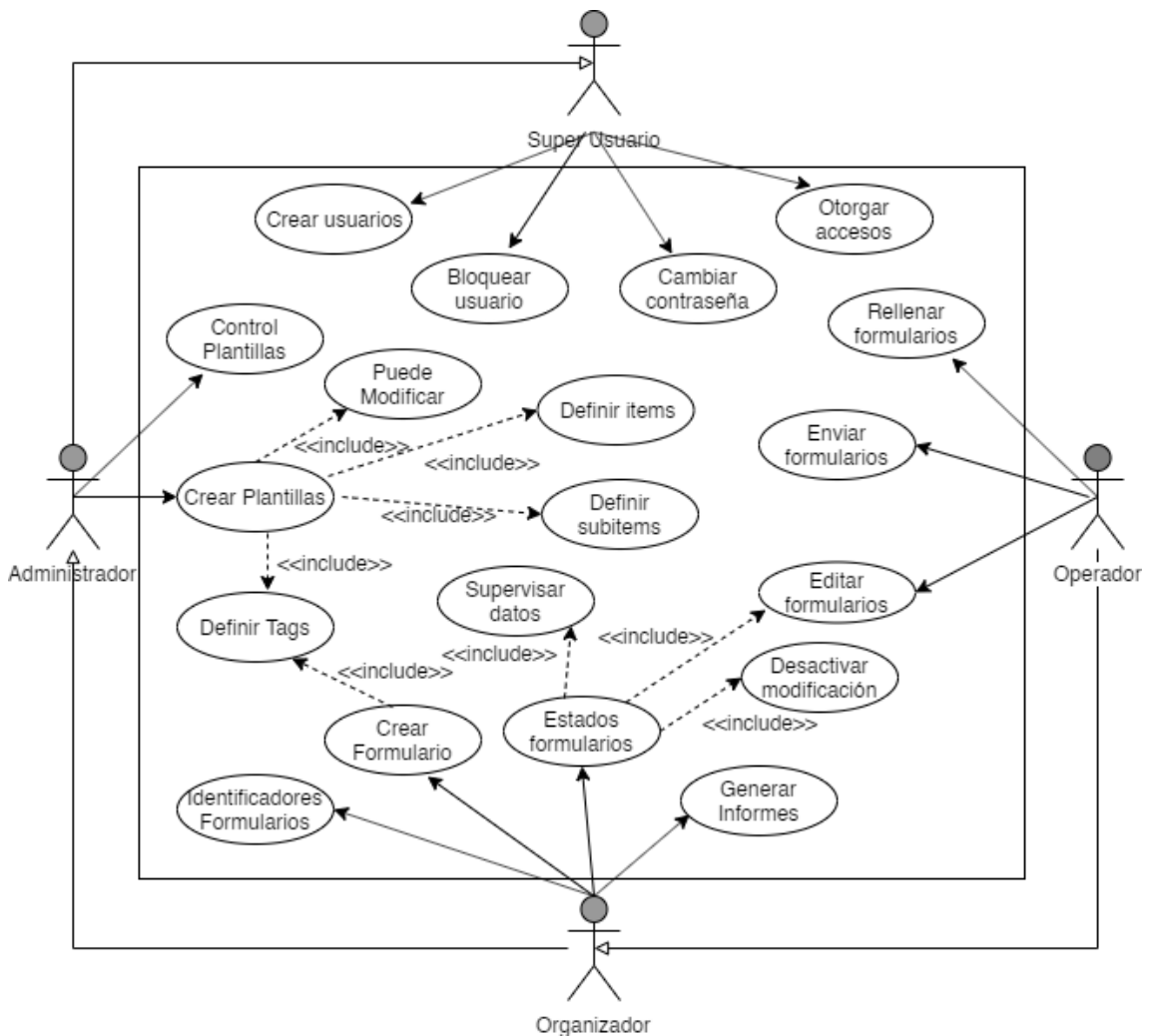


Ilustración 8 - Caso de uso general de la aplicación.



2.2. Diseño de la base de datos.

En este apartado, se discute la estructura que componen las tablas y sus principales características [9].

En las tablas que mantienen los valores principales de los usuarios, plantillas, operaciones y formularios se ha considerado que en caso de eliminación es más correcto marcar la información como eliminada, en vez de su eliminación real, para prevenir destrucción fortuita de información.

Se establece un campo extra de fecha de eliminación para que en posteriores procesos de mantenimiento de información se destruya la información que no se necesita. Se establecen campos que identifican a los creadores de elementos y la fecha de la misma para futuros uso que se añadan al sistema.

Tabla 1. Tabla **Usuarios**.

K	Columna	Tipo	Tam	Nulos	Descripción
*	id_user	INTEGER		No	Número interno de usuario (autoinc).
	user	CHAR	25	No	Identificador de usuario.
	pass	CHAR	25	No	Contraseña de usuario.
	super	CHAR	1	No	S / N Acceso total a la aplicación.
	uplan	CHAR	1	No	S / N Puede trabajar con plantillas.
	uope	CHAR	1	No	S / N Puede trabajar con operaciones.
	uactiva	CHAR	1	No	S / N Perfil activo.
	fuser	DATE		No	Fecha creación del usuario.
	del_user	CHAR	1	No	S / N Fila eliminada.
	fdel_user	DATE		Si	Fecha eliminación usuario.

Tabla 3 - Definición de valores que componen la tabla usuarios.

La tabla '**usuarios**' mantiene los usuarios de la aplicación que trabajan con plantillas y operaciones. El campo '**user**' será único entre las *tuplas* que no tengan el *flag* '**del_user**' a 'S'. Cuando se elimina un usuario no se borra de la base de datos, pero se indica la fecha en la que se realizó la acción. El valor '**uactivo**' permite indicar cuando un usuario puede trabajar con la aplicación.

Tabla 2. Tabla **Plantillas**.

K	Columna	Tipo	Tam	Nulos	Descripción
*	id_plan	INTEGER		No	Número interno id. plantilla (autoinc).
	ref_plan	CHAR	25	No	Referencia de la plantilla.
	des_plan	CHAR	100	No	Descripción de la plantilla.
	nitem	INTEGER		No	Indica el número de filas de la plantilla.
	nsubitem	INTEGER		No	Indica el número de divisiones de cada fila. Valor entre 1 y 2.
	pactiva	CHAR	1	No	S / N Plantilla activa.
	pfija	CHAR	1	No	S / N Plantilla definitiva.
FK	id_user	INTEGER		No	Usuario que crea la plantilla.
	fplan	DATE			Fecha creación de la plantilla.
	del_plan	CHAR	1	No	S / N Plantilla eliminada.
	fdel_plan	DATE		Si	Fecha eliminación plantilla.

Tabla 4 - Definición de valores que componen la tabla plantillas.

La tabla '**plantillas**' mantiene las plantillas. El valor '**nitem**' y '**nsubitem**' indica el número de datos que queremos recoger y el nivel de fragmentación de estos. El campo '**activo**' indica cuando la plantilla estará disponible para una operación. El campo '**pfija**' se usa para señalar que la plantilla no debe modificarse, y este campo debe ser puesto a 'S' la primera vez que es utilizada en una operación.

Tabla 3. Tabla **plan_items**.

K	Columna	Tipo	Tam	Nulos	Descripción
*FK	id_plan	INTEGER		No	Id plantilla.
*	nivel	INTEGER		No	Nivel de subítem.
*	ítem	CHAR	25	No	Valor del ítem.
	oitem	INTEGER		No	Orden del ítem.

Tabla 5 - Definición de valores que componen la tabla plan_items.

La tabla '**plan_items**' mantiene la relación de ítems que se quiere recoger y el orden en el que debe mostrarse. Si la columna '**nivel**' tiene un valor de '0' almacena los ítems. Los siguientes niveles muestran cómo se fragmenta cada dato. En la actualidad solo permite un máximo de 2 niveles. Cada nivel de disgregación de datos debe contener al menos un elemento. '**osubitem**' marca el orden en que mostrar los elementos de cada nivel.

Tabla 4. **plan_tags**.

K	Columna	Tipo	Tam	Nulos	Descripción
*FK	id_plan	INTEGER		No	Id plantilla.
*	tag	CHAR	25	No	Valor.

Tabla 6 - Definición de valores que componen la tabla plan_tags

La tabla **plan_tags** mantiene la relación de 'tags' pertenecientes a la plantilla.

Tabla 5. **Operaciones**.

K	Columna	Tipo	Tam	Nulos	Descripción
*	id_ope	INTEGER		No	Id operación. (autoinc)
	ref_ope	CHAR	25	No	Referencia operación.
	des_ope	CHAR	60	No	Descripción operación.
	oactiva	CHAR	1	No	S / N Operación activa.
	finiope	DATE		No	Fecha Inicio el operario ve la operación.
	ffinope	DATE		No	Fecha Final el operario ve la operación.
FK	id_plan	INTEGER		No	Plantilla en la que está basado.
FK	id_user	INTEGER		No	Usuario que ha creado la operación.
	pwd	CHAR	25	No	Clave aceptación formularios.
	fope	DATE		No	Fecha creación operación.
	del_ope	CHAR	1	No	S / N Operación eliminada.
	fdel_ope	DATE		Si	Fecha eliminación.

Tabla 7 - Definición de valores que componen la tabla operaciones.

La tabla **operaciones** mantiene las operaciones que se crean a partir de una plantilla. El campo '**ref_ope**' de las operaciones que no tengan el campo '**delete_ope**' a 'S' no pueden repetirse. '**oactivo**' marca si la operación está en modo lectura. La opción '**oactivo**' prevalece sobre las fechas de inicio '**finiope**' y '**ffinope**' introducidas. El campo '**pwd**' contiene la palabra de aceptación que los usuarios deben introducir en los formularios para que estos sean aceptados por el sistema.

Tabla 6. **ope_campo**.

K	Columna	Tipo	Tam	Nulos	Descripción
*FK	id_ope	INTEGER		No	Id. interno de la operación.
*	ref_campo	CHAR	5	No	Referencia al nombre de campo.
	campo	CHAR	25	No	Nombre del campo selector.
	tipo	CHAR	1	No	Como interpretar los valores para 'campo'. T -> Texto. D -> Fecha
	oselect	INTEGER		No	Orden que debe aparecer el valor.

Tabla 8 - Definición de valores que componen la tabla *ope_campo*.

La tabla **ope_campo** mantiene la relación de los campos de selección que individualizarán la identidad de cada formulario y el orden de su selección. El campo '**tipo**' indica cómo interpretar la información guardada en el campo '**valor**' de la tabla '*campo_valor*'.

Tabla 7. **campo_valor**.

K	Columna	Tipo	Tam	Nulos	Descripción
*FK	id_ope	INTEGER		No	Id. interno de la operación.
*FK	ref_campo	CHAR	5	No	Referencia al nombre del campo.
*	valor	CHAR	25	No	Valor del campo.
	ovalor	INTEGER		No	Orden en que debe aparecer el valor.

Tabla 9 - Definición de valores que componen la tabla *campo_valor*.

La tabla **campo_valor** mantiene la relación de los valores que corresponden a un campo de selección que individualizarán la identidad de cada formulario y el orden en que deben aparecer.

Tabla 8. **ope_tags**.

K	Columna	Tipo	Tam	Nulos	Descripción
*FK	id_ope	INTEGER		No	Id. interno de la operación.
*	tag	CHAR	25	No	Valor.

Tabla 10 - Definición de valores que componen la tabla *ope_tags*.

La tabla **ope_tags** mantiene la relación de los '**tags**' que se asocian a la operación.



Tabla 9. Formularios.

K	Columna	Tipo	Tam	Nulos	Descripción
*	id_form	INTEGER		No	Id. del formulario. (autoinc).
FK	id_ope	INTEGER		No	Id. Operación a la que pertenece.
FK	id_plan	INTEGER		No	Id. Plantilla de la que procede.
	factiva	CHAR	1	No	S / N Formulario activo.
	fform	DATE		No	Fecha creación del formulario.
	del_form	CHAR	1	No	S / N Formulario eliminado.
	fdel_form	DATE		SI	Fecha eliminación.

Tabla 11 - Definición de valores que componen la tabla formularios.

La tabla **'formularios'** mantiene los datos del formulario introducido. El valor del campo **'factiva'** marca si es posible editar y modificar el formulario o se encuentra en modo solo lectura.

Tabla 10. form_ope.

	Columna	Tipo	Tam	Nulos	Descripción
*FK	id_form	INTEGER		No	Id. interno del formulario.
*FK	id_ope	INTEGER		No	Id. interno de la operación.
*FK	ref_campo	CHAR	5	No	Campo seleccionado.
	valor	CHAR	60	No	Valor del campo.

Tabla 12 - Definición de valores que componen la tabla form_ope.

La tabla **'form_ope'** contiene los valores que van a identificar de forma inequívoca a cada formulario introducido.

Tabla 11. form_plan.

	Columna	Tipo	Tam	Nulos	Descripción
FK	id_form	INTEGER		No	Id. interno del formulario.
FK	id_plan	INTEGER		No	Id. interno de la plantilla.
	Ítem	CHAR	25	No	Referencia del ítem.
	sub1	CHAR	10	No	Referencia del 1 subítem.
	sub2	CHAR	10	Si	Referencia del 2 subítem. Si se ha declarado un segundo nivel no puede estar vacío.
	valor	INTEGER		No	Valor que queremos almacenar.

Tabla 13 - Definición de valores que componen la tabla form_plan.

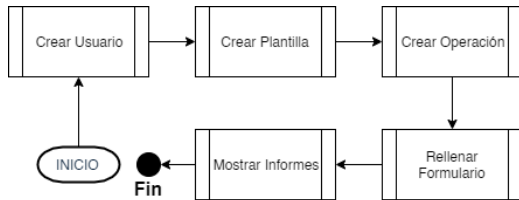
La tabla **'form_plan'** contiene los valores que se introducen en el formulario. Dependiendo del nivel del subítem hay elementos que no podrán ser nulos. Solo debe existir un valor para un mismo **'ítem'** & **'sub1'** & **'sub...'**....



2.3. Flujos de iteración.

En este apartado se muestran los principales flujos de iteración para operaciones básicas de creación en las distintas pantallas, de forma que ofrece una visión inicial de cómo debería trabajar el sistema [5] [6].

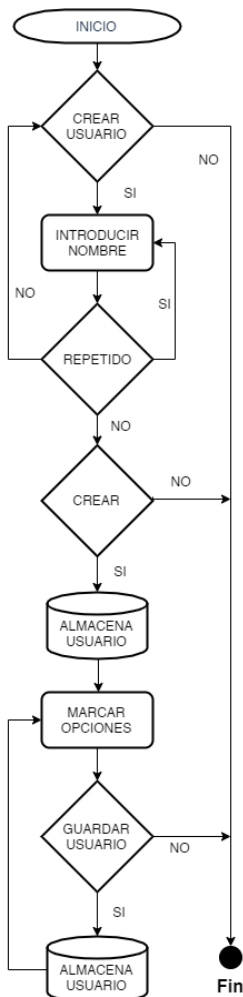
- Flujo general de funcionamiento.



Se da de alta usuarios a los que se asignan roles. Dependiendo de los roles, los usuarios crean plantillas que se usan en los formularios que recopilan datos para las operaciones, y así, generar informes sobre la actividad desarrollada.

Ilustración 9 - Iteración general programa.

- Creación de usuarios.



Se introduce un nombre y se pulsa 'Buscar', si existe se traen sus datos, si no existe permite la opción 'Añadir'.

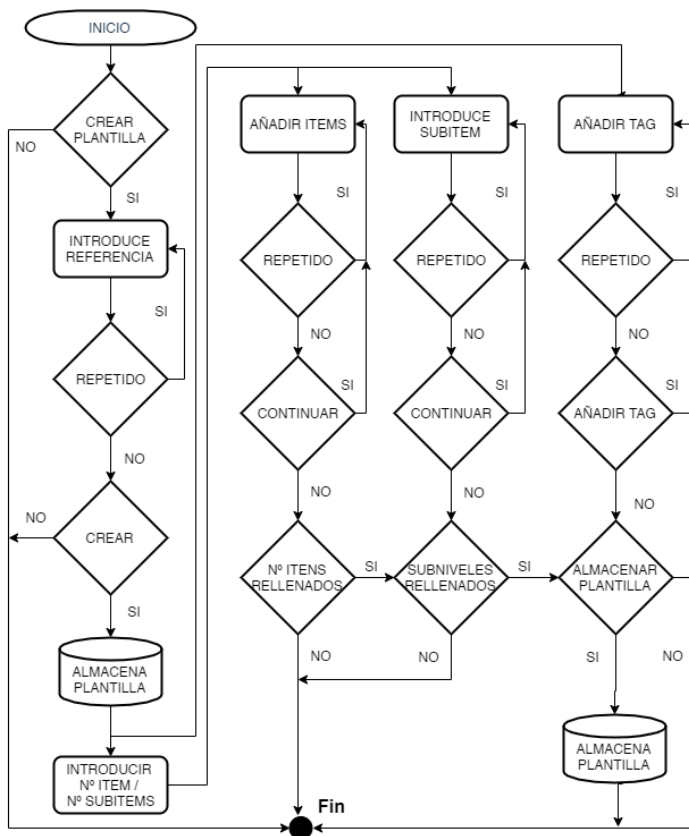
Una vez creado el usuario en el sistema se señalan opciones que definen sus roles dentro de la aplicación y su contraseña de acceso.

Para que el usuario use la aplicación es necesario señalar que está activo su perfil y no está en blanco la casilla de la contraseña.

Ilustración 10 - Iteración creación usuario.



• Creación de plantillas.



Se introduce una referencia de plantilla, si no existe permite dar su alta, si existe se traen sus datos.

Se indica el número de ítems y subniveles que existen.

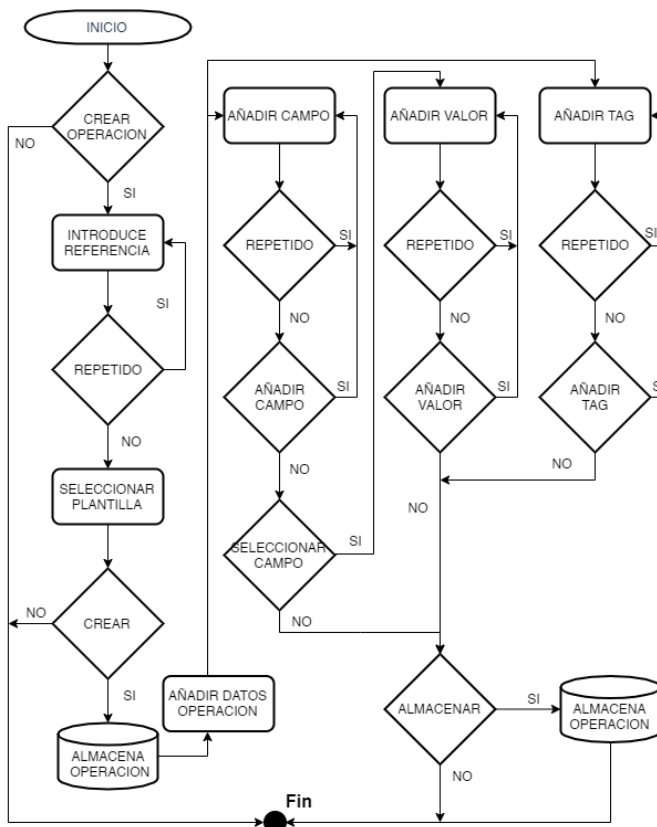
Para almacenar las modificaciones debe introducir el número de ítems y valores en cada uno de los subniveles indicados.

No se exige la introducción de 'tags' para almacenar la plantilla.

Para que la plantilla esté disponible debe señalarse la opción 'Plantilla activa'.

Ilustración 11 - Iteración creación plantilla.

• Creación de operaciones.



Se introduce una referencia de operación, si no existe permite dar su alta seleccionando antes una plantilla. Al dar de alta se copian los 'tags' introducidos en la plantilla. Si existe trae los datos de la operación y la relación de formularios rellenados.

Mientras no se haya rellenado un formulario se permite cambiar de plantilla.

Cuando se introduce un campo se indica el tipo de campo y deja introducir valores para dicho campo.

Para que el formulario esté activo se señala la opción de 'Operación activa'.

Ilustración 12 - Iteración creación operación.



• Rellenar formulario.

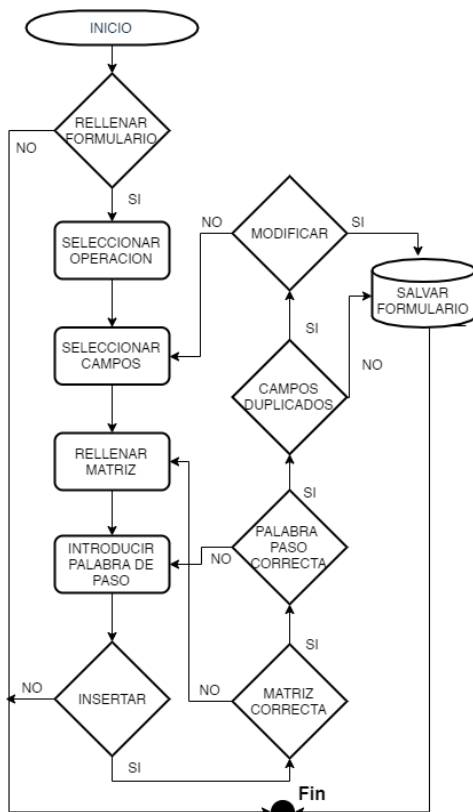


Ilustración 13 - Iteración creación formulario.

Se selecciona una operación cuyo momento en el que se va a rellenar el formulario se encuentra entre sus fechas de inicio y finalización con un estado de operación activa.

El conjunto de los campos seleccionados debe ser único para una operación. Si se ha enviado antes el formulario con dichos campos rellenados, debe indicarse si quiere sustituir el anterior introducido.

La palabra de aceptación del formulario se ha definido a la hora de crear la operación.

El sistema admite valores con decimales separados por punto.

• Mostrar Informe.

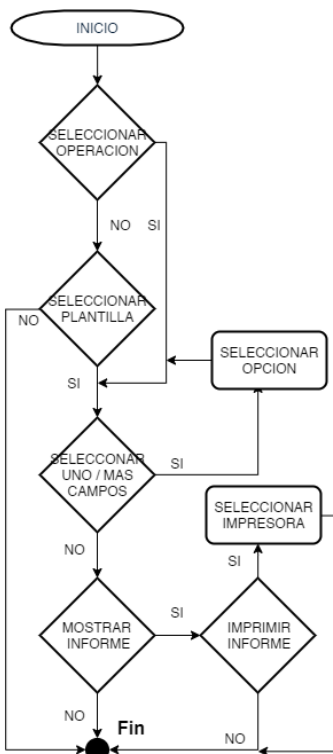


Ilustración 14 - Iteración generación Informe.

Se selecciona una operación o una plantilla. Las operaciones y las plantillas deben estar activas.

Cada una de las dos opciones muestra una serie de valores que sirven para acotar la búsqueda. Si no se señala se hace el acumulado del conjunto de valores. Si se señala algún valor se realiza el acumulado por el/los valor/es señalados.

Para visualizar el resultado se pulsa sobre el botón 'Informe', y, si queremos imprimir o guardar el resultado se pulsa el botón 'Imprimir'.



2.4. Representación visual de pantallas a alto nivel.

Muestra una relación de pantallas que componen el sistema [6].

- Pantalla entrada a la aplicación.

Diseño inicial. Pantalla final.

The 'Diseño inicial' shows a browser window with a menu of buttons: Formularios, Operaciones, Plantillas, Informes, Usuarios, and Salir. The 'Pantalla final' shows a more refined interface titled 'SIREDACU.' with the same menu items as blue buttons, a login form with fields for 'Usuario:' (containing 'user') and 'Contraseña:' (containing '****'), and an 'Enviar' button.

Ilustración 15 - Evolución diseño pantalla inicial.

- Pantalla usuarios.

1^{er} boceto. Diseño inicial. Pantalla final.

The '1^{er} boceto' is a handwritten sketch. The 'Diseño inicial' shows a browser window titled 'Ctrl. Usuarios' with a dropdown menu for 'Usuario:' (set to 'Option 1'), plus and minus buttons, an 'Activo' checkbox, checkboxes for 'Operaciones:', 'Plantillas:', and 'Super usuario:', a 'Clave:' field, and 'Guardar' and 'Salir' buttons. The 'Pantalla final' shows a polished interface titled 'USUARIOS.' with a 'Usuario:' field, 'Buscar' and 'Quitar' buttons, a 'Usuario activo:' section with checkboxes for 'Operaciones.', 'Plantillas.', and 'Super Usuario.', a 'Contraseña:' field, and 'Modificar' and 'Salida' buttons.

Ilustración 16 - Evolución diseño pantalla usuario..

- Pantalla Creación de plantillas.

1^{er} boceto. Diseño inicial. Pantalla final.

The '1^{er} boceto' is a handwritten sketch. The 'Diseño inicial' shows a browser window titled 'Ctrl. Plantillas' with a dropdown for 'Plantilla:' (set to 'Option 1'), an 'Activa.' checkbox, a 'Descripción:' field, and a complex grid for adding items (L1 and L2) with plus and minus buttons, and a 'Tag' field with a list of tags. The 'Pantalla final' shows a polished interface titled 'PLANTILLAS.' with a 'Plantilla:' dropdown, 'Buscar' and 'Quitar' buttons, a table with columns for 'Nº Items', 'Nº Niveles', and 'Nº Modos', and 'Modificar' and 'Salida' buttons.

Ilustración 17 - Evolución diseño pantalla plantilla.



- Pantalla Creación de operación.

1^{er} boceto. Diseño inicial. Pantalla final.

The illustration shows the design evolution for the 'OPERACIONES' screen. It starts with a hand-drawn sketch on the left, followed by a wireframe labeled 'Diseño inicial' in the middle, and finally the polished 'Pantalla final' on the right. The final screen features a title 'OPERACIONES.', a search bar for 'Operación', a description field, and a table with columns for 'Fecha', 'Punto', 'Hombres', and 'Mujeres'. The table contains data for three dates: 2019-07-07, 2019-07-08, and 2019-07-09, with values for 'Hombres' and 'Mujeres' in different age groups.

Ilustración 19 - Evolución diseño pantalla operación.

- Pantalla Creación de formulario.

1^{er} boceto. Diseño inicial. Pantalla final.

The illustration shows the design evolution for the 'FORMULARIOS' screen. It starts with a hand-drawn sketch on the left, followed by a wireframe labeled 'Diseño inicial' in the middle, and finally the polished 'Pantalla final' on the right. The final screen features a title 'FORMULARIOS.', a search bar for 'Operación', a description field, and a table with columns for 'Hombres' and 'Mujeres' in different age groups. The table contains data for three dates: 2019-07-07, 2019-07-08, and 2019-07-09, with values for 'Hombres' and 'Mujeres' in different age groups.

Ilustración 18 - Evolución diseño pantalla formularios.



3. IMPLEMENTACIÓN.

3.1. Servicios Web.

Para disponer de los servicios web se ha decidido recurrir al proveedor donde se alojará la versión definitiva, para conseguir un *feedback* continuo mientras se construye la aplicación y ahorrar los recursos y tiempo necesario que harían falta para la puesta en marcha de un servidor web real, con todos los requerimientos necesarios para la aplicación (instalación y mantenimientos de los diferentes servicios, aplicaciones, securización del equipo, reconfiguración de la red...). En caso de crear un servidor web propio, se recomienda realizar una instalación de una distribución que incluya LAMP¹⁹ (Linux, Apache, MySQL/MariaDB y PHP).

Se ha creado un subdominio para la ocasión, de manera que se aísla el proyecto del resto del dominio para no interferir con otros trabajos. Esto permite personalizar la versión del servidor de base de datos y el PHP utilizado y, en caso de necesidad, modificar la configuración actual del hosting.

Para configurar el subdominio 'siredacu' se ha usado la herramienta *plesk web host edition*, que es un panel de control centralizado para controlar los distintos servicios que se personalizan para cada subdominio. En este caso he podido crear una base de datos personalizada de manera que se protege al resto de aplicaciones frente a desastres fortuitos por error de programación. El dominio utiliza la versión php 7.2.25 que es suficiente para el propósito seguido.

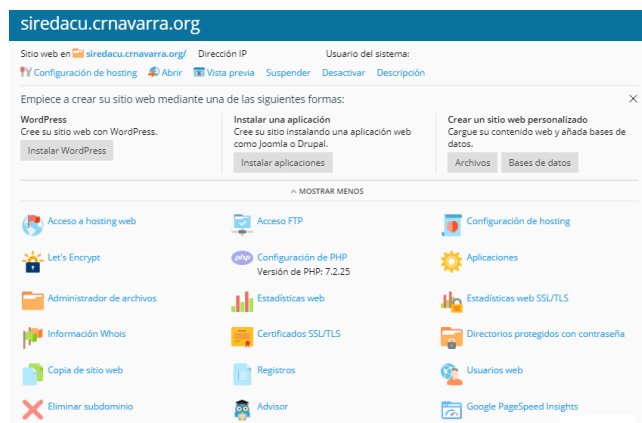


Ilustración 20 - Pantalla gestión dominio siredacu.crnavarra.org.

Aunque se han descargado al directorio local librerías de JavaScript de terceros para que en un momento dado continuara desarrollando sin conexión a internet, las páginas web recurren a los distintos CDN (*Content Delivery Network*) de los productos de terceros para acceder a las librerías publicadas.

```
<link rel="stylesheet" href=" ../css/bootstrap.min.css" integrity="sha384-ggOyR0iXCbMQv3Xipma34MD+dH/1fQ784/j6cY/iJTQU0hcWr7x9JVoRXT2MZw1T" crossorigin="anonymous">
```

Ilustración 21 - Ejemplo enlace a librerías en CDN.

¹⁹ <https://es.wikipedia.org/wiki/LAMP>



3.2. Base de datos.

Para la base de datos se ha utilizado MariaDB que es un servidor de base de datos *open source* en la que se ha implementado el diseño de tablas mostrado en el apartado ‘2.2 Diseño de la base de datos.’.

Se han generado **campos autoincrementales** para las tablas ‘formularios’, ‘operaciones’, ‘plantillas’ y ‘usuarios’ de forma que permite un mejor control de los datos aislando las partes manipulables y editables que se muestran, de la forma en que se trabajan los datos en la base de datos.

Se han declarado los siguientes valores como **claves foráneas**. ‘id_user’ en las tablas ‘plantillas’ y ‘operaciones’; ‘id_plan’ en las tablas ‘plan_items’, ‘plan_tags’ y ‘operaciones’; ‘id_ope’ en las tablas ‘ope_campo’, ‘ope_tags’, y ‘formularios’; ‘id_ope’-‘ref_campo’ en las tablas ‘campo_valor’ y ‘form_ope’; ‘id_form’ en las tablas ‘form_ope’ y ‘form_plan’. Las claves foráneas indican que los valores son una referencia a un valor que existe en otra tabla como valor principal y, por otro lado, se ha indicado a la base de datos que se usen como elemento de restricción y no se eliminen de la tabla principal hasta que deje de existir en todas las tablas donde estás referenciados.

En la programación de la aplicación se ha adoptado la medida de no eliminar los datos cuando se eliminan si estos han sido utilizados, sino marcarlos a través de un campo como eliminados. Con esta medida se recuperarán datos eliminados de manera accidental creando, cuando se considere necesario, procesos específicos de mantenimiento para su eliminación controlada.

Se muestra el diseño de entidad relación que aporta la estructura de las tablas y cómo están relacionados los datos que se almacenan en la base de datos de manera gráfica.

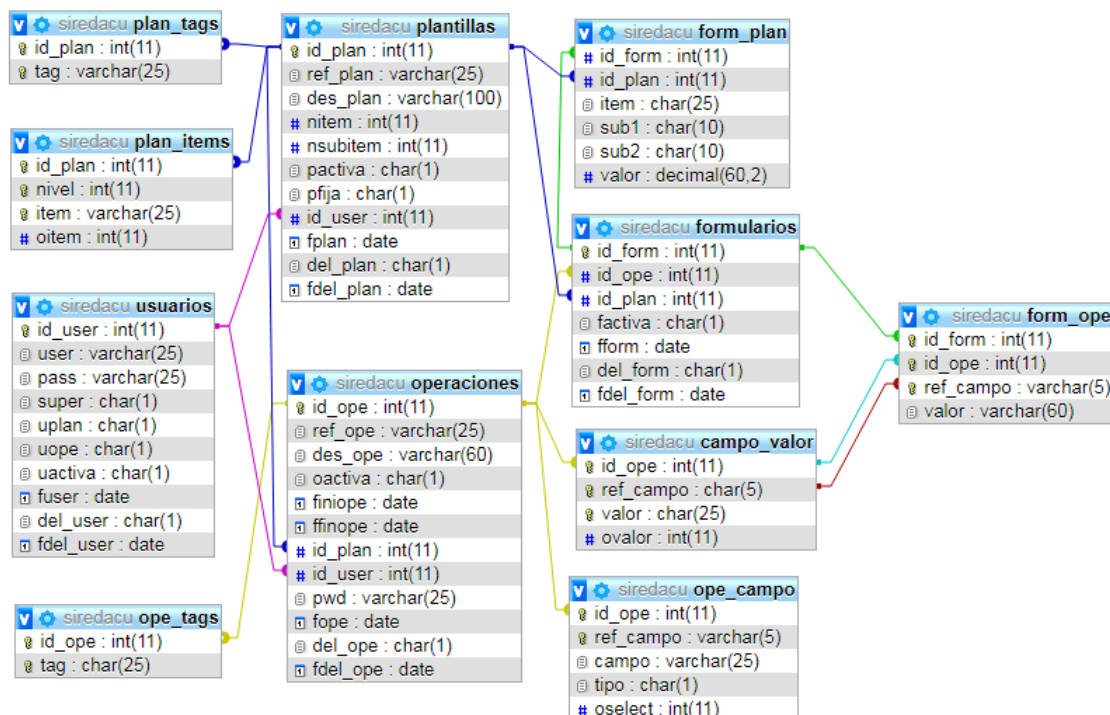


Ilustración 22 - Diseño Entidad Relación base de datos.



3.3. Dispositivos utilizados.

En la creación de la aplicación se han utilizado dos dispositivos para comprobar el desarrollo de las distintas pantallas. Por un lado, la propia computadora donde se programaba que ejecuta un Windows 7 profesional con Google Chrome y un móvil con sistema operativo Android versión 9.

La programación se hacía a la vez que se observaba el comportamiento con diferentes tamaños de pantalla utilizando Bootstrap, que es una librería especializada en la representación de elementos en el DOM. La librería es *responsive* que significa que el contenido se adapta a los diferentes tamaños de pantalla. En el ejemplo podemos comprobar que dependiendo del tamaño de la pantalla ((lg) larga, (md) media, (sm) pequeña), los botones utilizaran una 2/12, 3/12 o 6/12 parte de la anchura de la misma, además las clases (btn btn-primary btn-lg btn-block m-1) indican cómo debe mostrarse el botón en la pantalla.

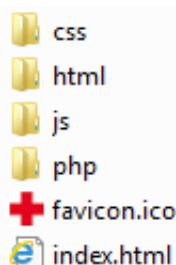
```
<div class="col-lg-2 col-md-3 col-sm-6" v-if="btnFormularios == false">
  <button type="button" class="btn btn-primary btn-lg btn-block m-1"
    :disabled="btnFormularios"
    @click="openVentana('./html/formularios.html')">Formularios
  </button>
</div>
```

Ilustración 23 - Ejemplo uso bootstrap.

Si con Bootstrap 4 se controla cómo mostrar los diferentes elementos, con Vue se controla de forma automatizada la visualización de datos, que permite ahorrar la introducción de código, como se muestra con el siguiente código en el que cuando se señala un campo de selección de forma automática redibuja el DOM mostrando los valores que corresponde al campo que se señala.

```
<!-- Mostramos los campos de selección -->
<ul class="list-inline ">
  <li class="pl-2 list-inline-item"
    v-for="(campo, cnt) in campos" :key="cnt">
    <label for="ref_ope" class="col-form-label">
      {{campo.campo}}:
    </label>
    <select v-model="currentValorCnt[cnt]"
      :class="{ 'error':err.ref_campo[cnt]}" class="custom-select">
      <option v-for="(valor, cnt) in valores[cnt]"
        :key='cnt' v-bind:value="valor.valor" > {{valor.valor}}
      </option>
    </select>
  </li>
</ul>
```

Ilustración 24 - Ejemplo uso vue.js



Los ficheros generados que conforman la aplicación web se han distribuido en una estructura de fácil reconocimiento, agrupados por el tipo de fichero que representan. Aunque en el directorio JavaScript se han incluido algunas de las librerías externas para poder trabajar *offline* en los ficheros '.html' se llaman por defecto a las librerías externas.

Ilustración 25 - Distribución ficheros aplicación web.



Se ha usado Google Chrome para probar la aplicación, no solo porque es la plataforma seleccionada para que se ejecute el desarrollo en su puesta en marcha, sino por las distintas herramientas que aporta a los desarrolladores.

A través de estas herramientas se realiza el seguimiento de la ejecución de código, inserción de puntos de ruptura, seguimiento específico de variables, consulta del estado del entorno, depuración del código, corrección de errores...

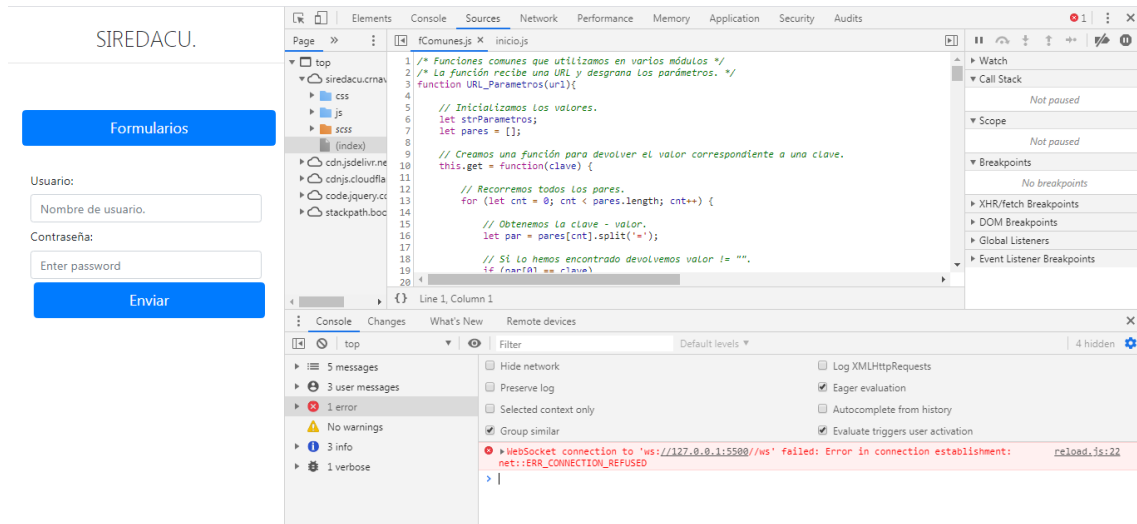


Ilustración 26 – Depuración de código en Google Chrome



4. PRUEBAS.

Al seguir la metodología de programación extrema (XP) trabajo su ciclo de vida (Análisis, Diseño, Desarrollo y Pruebas) [2]. En este apartado me centro en las distintas pruebas que he realizado para comprobar los diseños en cada fase del desarrollo.

Pantalla inicial.

- ✓ Los botones enlazan con las distintas pantallas.
- ✓ Cualquier usuario incluido los anónimos tiene activo el botón 'Formulario'.
- ✓ La activación del resto de los botones depende del rol del usuario tras comprobar la contraseña introducida.

Pantalla usuarios.

- ✓ Al entrar en la pantalla solo se visualiza la zona de introducción del usuario, los botones 'Añadir' y 'Quitar' están deshabilitados, y, los botones 'Buscar' y 'Salir' están habilitados.
- ✓ Se comprueba que el valor del nombre cumple con los valores admitidos en la base de datos.
- ✓ Tras pulsar el botón 'Buscar', si el usuario no existe se activa el botón 'Añadir', si existe se activa el botón 'Quitar', la zona de asignación de roles, la introducción de contraseña y el botón 'Modificar'.
- ✓ Al pulsar el botón 'Añadir' se activa la zona de introducción de roles, la contraseña, se desactiva el botón 'Añadir' y se activan los botones 'Quitar' y 'Modificar'. Se crea el usuario en la base de datos, se asigna una fecha y se devuelve un identificador.
- ✓ Se comprueba que los 'checkbox' se corresponden con los valores que se almacenan en la base de datos y la contraseña se muestra con caracteres '*'.
- ✓ Al pulsar el botón 'Modificar' se actualizan los datos en la base de datos.
- ✓ Al pulsar el botón 'Quitar' se pone el *flag* de eliminado en la base de datos y se inserta la fecha de eliminación.
- ✓ Se prueban situaciones de error para comprobar que salen las notificaciones de error y se señalan los campos erróneos.
- ✓ Las ventanas emergentes se muestran conforme se espera.



Pantalla plantillas.

- ✓ Mientras no exista una plantilla creada no se muestran ningún otro campo distinto a la referencia de la plantilla.
- ✓ Al entrar en la pantalla solo se visualiza la zona de introducción de la plantilla, los botones 'Añadir' y 'Modificar' están deshabilitados, y, los botones 'Buscar' y 'Salir' están habilitados.
- ✓ Cuando se han consultado los valores de la plantilla si se escribe un dato en la referencia de plantilla se ocultan los datos puesto que es una nueva búsqueda.
- ✓ No permite realizar nuevas búsquedas o salir de la pantalla hasta que no almacene las modificaciones.
- ✓ Si se indica un número de ítems al pulsar el botón 'Modificar' este valor debe coincidir con la relación de ítems introducidos.
- ✓ Si se indica número de niveles debe aparecer un conjunto de introducción de datos para cada nivel.
- ✓ Cuando se pulse el botón 'Modificar' deben existir al menos un valor en cada uno de los niveles indicados.
- ✓ No se repite el valor dentro de cada grupo de nivel o ítems.
- ✓ Si se ha utilizado la plantilla en alguna operación no se modificar el número de niveles.
- ✓ Si se ha utilizado la plantilla en alguna operación no se podrá eliminar un valor tanto sea un ítem como un valor de nivel que haya sido utilizado.
- ✓ No puede haber dos referencias de plantillas no eliminadas iguales.
- ✓ La relación de ítems y valores de cada nivel aparecen en el orden correcto a como se describen al crearlos.
- ✓ Cuando se selecciona un ítem o un valor de un nivel este se muestra en la pantalla de selección.
- ✓ Cuando solo hay un ítem o un valor en cada nivel no se activa los botones de subida y bajada, pero si el de eliminación.
- ✓ Si se eliminan todos los ítems o valores de un nivel se desactivan los botones 'sub', 'baj', '+' y '-' en el que se encontraba.
- ✓ Cuando se señala el primer valor de un ítem o nivel no se activa el botón 'sub'.
- ✓ Cuando se señala el último valor de un ítem o nivel no se activa el botón 'baj'.
- ✓ No se pueden repetir valores de 'tags'.
- ✓ Se prueban situaciones de error para comprobar que salen las notificaciones de error y se señalan los campos erróneos.



Pantalla Operaciones.

- ✓ Cuando se introduce una operación, si no existe muestra la relación de plantillas activas para seleccionar una plantilla.
- ✓ Mientras no se seleccione una plantilla no permite crear una operación.
- ✓ No se activa una operación mientras no exista una fecha de inicio y finalización correcta.
- ✓ Si se introduce un valor en el campo de entrada de operación se borran los datos.
- ✓ El campo contraseña se muestra con caracteres '*'.
✓ La fecha de inicio y finalización de la operación son válidas.
- ✓ La fecha de finalización es posterior a la de inicio.
- ✓ Si la plantilla se ha utilizado en un formulario no permite escoger otra para la operación.
- ✓ No se crea un campo cuyo nombre o referencia coincida con otro campo.
- ✓ Se respetan el número de caracteres máximos descritos para campos, referencias o valores descritos en la base de datos.
- ✓ Un campo no se elimina mientras contenga valores.
- ✓ Un campo no se elimina si se ha utilizado en un formulario.
- ✓ Un valor de un campo no se elimina si se ha utilizado en un formulario.
- ✓ No pueden coexistir dos 'tags' que tengan el mismo nombre.
- ✓ El primer elemento de un campo o valor no se puede subir de orden.
- ✓ El último elemento de un campo o valor no se puede bajar de orden.
- ✓ Se modifica un dato de un formulario pese a que esté marcado como solo lectura.
- ✓ Solo lectura está marcada cuando un formulario no se permite que lo modifiquen en la pantalla 'Formularios'.
- ✓ Si un valor no es numérico no debe aceptarse en un formulario.
- ✓ Cuando se pulsa sobre el botón 'eliminar' en un formulario se pide confirmación.
- ✓ Cuando se pulsa sobre el botón 'eliminar' en un formulario se marca como eliminado y se indica una fecha de eliminación en la base de datos.
- ✓ Se prueban situaciones de error para comprobar que salen las notificaciones de error y se señalan los campos erróneos.

Pantalla Formularios.

- ✓ La pantalla permite seleccionar una operación activa, cuyo momento en el que se rellene el formulario, se encuentra entre la fecha de inicio y finalización indicadas en la operación.
- ✓ La descripción de la operación no se modifica.
- ✓ Aparecen los campos en el orden creado.
- ✓ Cada campo contiene los valores que se muestran ordenados como se ha indicado a la hora de crear la operación.
- ✓ Se genera una tabla para la introducción de datos.
- ✓ Debe señalarse todos los campos de selección para poder insertar un formulario en la base de datos.
- ✓ Cuando se inserta los datos en la base de datos se comprueba que la palabra de aceptación sea correcta.
- ✓ Cuando existe un formulario previo con los mismos campos señalados permite cancelar la inserción o sobrescribir los datos del anterior formulario.
- ✓ Si existe un formulario previo con los mismos campos señalados y marcado como solo lectura no permite su actualización.



Informes.

- ✓ Cuando se empieza a escribir una operación se acota una búsqueda de operaciones activas y desaparece la caja de entrada de plantillas.
- ✓ Cuando se selecciona una operación se muestran los campos con los valores creados para cada uno de ellos.
- ✓ Si se señala un valor en cualquiera de los campos, la búsqueda de datos se acotará por aquellos formularios en los que coincida los datos.
- ✓ Cuando se empieza a escribir una plantilla se acota una búsqueda de plantillas activas y desaparece la caja de entrada de operaciones.
- ✓ Cuando se señala una plantilla se muestran las operaciones que tienen la plantilla y están activas. Se muestra la relación de 'tags' que tienen insertadas las operaciones.
- ✓ Cuando se pulsa el botón de 'Informe' se muestra el resultado y se oculta la parte de selección.
- ✓ Cuando se pulsa el botón 'Volver a seleccionar' se oculta el resultado y vuelve los controles de selección de la operación o plantilla seleccionada.
- ✓ Cuando se pulsa sobre el botón 'Imprimir' se lanza la pantalla de impresión donde se ocultan los botones y solo aparece el informe.



5. ACEPTACIÓN DEL PROYECTO.

Para la aceptación del proyecto es importante que el programa permita realizar la recogida a pie de calle, por los operadores, el resumen de atenciones que se realizan en los encierros de las fiestas de San Fermín y se introduzcan en sus equipos dichos datos para que estén a disposición del área de Marca y Comunicación, como se indicaba en los antecedentes de la propuesta del trabajo.

En este apartado, demostraremos que el desarrollo web realizado admite la preparación y ejecución del trabajo descrito y que supone el punto inicial de partida para la explotación de la aplicación. Para ello se usa un usuario con todos los permisos.

Primero se genera una plantilla a través del formulario que existe en la pantalla con el mismo nombre. Se da un nombre y se pulsa la acción ‘Añadir’ para darlo de alta en la base de datos, pudiendo a continuación insertar un comentario.

Ilustración 27 - Datos generales de una plantilla.

A continuación, se indica el número de ítems (10) y en cuántos niveles están fragmentados los datos (2) para que permita introducir en cada nivel la relación de valores.

Ilustración 28 - Introducción de ítems y definición de fragmentación de datos.

Por último, se asignan palabras claves que sirvan para agrupar formularios rellenos que correspondan a distintas operaciones.

Ilustración 29 - Introducción de etiquetas.



Para utilizar la plantilla en operaciones es necesario activar la misma marcando la casilla reservada a tal efecto.

Ilustración 30 - Activación de plantilla.

Una vez generada la plantilla su utiliza dentro de una operación. Para ello en la pantalla ‘Operaciones’ introducimos el nombre que queremos asignar. Si no existe, muestra un cuadro combinado donde seleccionar la plantilla que he creado y dar de alta la operación. Esta acción copia los ‘tags’ definidos en la plantilla.

Ilustración 31 - Selección plantilla para operación.

Ya podemos dar una descripción a la operación e insertar una fecha de inicio y final de la operación. Estas fechas son importantes porque es la ventana de tiempo en la que se introducirán los formularios. Definimos la palabra de paso de aceptación de los formularios rellenados.

Ilustración 32 - Introducción de datos generales a la operación.

Ya podemos crear los campos de selección que identificarán de forma inequívoca cada formulario. Se dan de alta dos campos, uno contendrá el día y otro el puesto. Para el campo ‘día’ se da de alta los días que hay encierro, y en el campo ‘puesto’ se asigna un valor para cada uno de los puestos donde se realizan atenciones.

Ilustración 33 - Introducción de valores para cada campo creado.



Por último, añadimos nuevos ‘tags’ a los que se han copiado de la plantilla de trabajo seleccionada.

Ilustración 34 - Introducción de etiquetas para la operación.

Para introducir formularios, hay que marcar que la operación está activa.

Ilustración 35 - Activación de la operación.

Ahora tenemos listas las condiciones para que los operadores del servicio del encierro envíen la relación de atenciones realizadas en sus puestos. Por ejemplo, el operador de radio que va al puesto de atención en el ayuntamiento el día 7 de Julio, debe entrar en la pantalla de formularios, para la que no necesita ningún permiso, y seleccionar entre las operaciones activas la que se refiere a encierros.

Ilustración 36 - Selección de la operación para el formulario.

Al seleccionar la plantilla el sistema genera la relación de campos para seleccionar los valores para su identificación. Escogemos la fecha del 7 de julio de 2019 y el puesto del Ayuntamiento.

Ilustración 37 - Identificar unívocamente un formulario.

Podemos introducir las atenciones realizadas en la matriz que se forma con los valores que definen la plantilla seleccionada.

	HOMBRES			MUJERES		
	< 18	18 - 25	> 25	< 18	18 - 25	> 25
Pisotones	1		3		3	
H. Asta						
Lipotimia						
Curas, Heridas		4		1		3
Esguinces						
At. Médicas						
Luxaciones						
Traslados		1				
Obitos						
Otros			2			1

Ilustración 38 - Introducción de datos en la matriz.



Si quiere editar los datos introducidos el operario no tiene más que enviar de nuevo el formulario y aceptar la solicitud de modificación del formulario.

Como medida de seguridad para enviar la matriz, debe introducir la palabra de aceptación indicada cuando se generó la operación.

Contraseña:

Ilustración 39 - Validación del envío del formulario.

Aquellos usuarios que gestionan las operaciones verán la relación de formularios que se han rellenado, donde podrán bloquear su edición para que nadie los modifique por error, eliminar un formulario o cambiar los datos que considere.

Solo lectura	Eliminar	Fecha	Puesto	Písotones						H. Asta								
				HOMBRES			MUJERES			HOMBRES			MUJERES			HON		
				< 18	18 - 25	> 25	< 18	18 - 25	> 25	< 18	18 - 25	> 25	< 18	18 - 25	> 25	< 18	18	
<input type="checkbox"/>	Eliminar	2019-07-07T00:00	Ayuntamiento	1.00		3.00		3.00										

Ilustración 40 - Control de formularios introducidos.

Por último, entrando en la página de informes podemos realizar dos tipos de informe dependiendo de si se quiere sobre una operación o plantilla.

Seleccione una operación o una plantilla:

Ilustración 41 - Selección de operación / plantilla para realizar el informe.

Dependiendo de la opción que seleccionemos tendremos unos campos de selección u otros para acotar el resultado de los informes. Cuando se selecciona una operación muestra los campos de selección creados para la operación. Cuando se selecciona una plantilla se muestran la relación de operaciones que usan esa plantilla y el conjunto de 'tags' definidos para esas operaciones, de forma que el informe acotará bien por las operaciones señaladas o los 'tags' que se hayan definido en las operaciones.

Seleccione una operación o una plantilla:

Seleccione una operación o una plantilla:

Fecha

- 2019-07-07 00:00
- 2019-07-08 00:00
- 2019-07-09 00:00
- 2019-07-10 00:00

Puesto

- Santo Domingo
- Ayuntamiento
- Mercaderes
- Estafeta

Operaciones:

Tags:

- 2019
- Encierro
- SF
- SF2019

Ilustración 42 - Selección acotación de datos para el informe.



Al pulsar sobre 'Informes' la aplicación mostrará el resultado. Si queremos cambiar la selección de elementos no tenemos más que pulsar sobre el botón 'Volver a seleccionar'. Si queremos imprimir el resultado se pulsa sobre el botón 'Imprimir' que mostrará la misma hoja ocultando los botones.

CRUZ ROJA NAVARRA - Encierros SF 2019

Resumen de datos introducidos

	HOMBRES			MUJERES			Total
	< 18	18 - 25	> 25	< 18	18 - 25	> 25	
Pisotones	1	0	3	0	3	0	7
H. Asta	0	0	0	0	0	0	0
Lipotimia	0	0	0	0	0	0	0
Curas, Heridas	0	4	0	0	1	3	8
Esguinces	0	0	0	0	0	0	0
At. Médicas	0	0	0	0	0	0	0
Luxaciones	0	0	0	0	0	0	0
Traslados	0	1	0	0	0	0	1
Obitos	0	0	0	0	0	0	0
Otros	0	0	2	0	0	1	3
Totales	1	5	5	0	4	4	19

	HOMBRES	MUJERES	Total		< 18	18 - 25	> 25	Total
Pisotones	4	3	7	Pisotones	1	3	3	7
H. Asta	0	0	0	H. Asta	0	0	0	0
Lipotimia	0	0	0	Lipotimia	0	0	0	0
Curas, Heridas	4	4	8	Curas, Heridas	0	5	3	8
Esguinces	0	0	0	Esguinces	0	0	0	0
At. Médicas	0	0	0	At. Médicas	0	0	0	0
Luxaciones	0	0	0	Luxaciones	0	0	0	0
Traslados	1	0	1	Traslados	0	1	0	1
Obitos	0	0	0	Obitos	0	0	0	0
Otros	2	1	3	Otros	0	0	3	3
Totales	11	8	19	Totales	1	9	9	19

Imprimir.

Volver a seleccionar.

Salida.

Ilustración 43 - Ejemplo resultado del informe antes de imprimir.



6. CONCLUSIONES

En primer lugar, resaltar que el presente trabajo ha sido un reto académico por la exigencia de lograr un sistema funcional en todos los apartados definidos y un reto personal para crear un producto que responde a una necesidad real y que, una vez finalice la presentación del mismo, se pondrá en producción.

El trabajo me ha dado una perspectiva básica de cómo funciona un desarrollo web y experiencia en el uso de algunas de sus tecnologías, permitiendo un primer contacto real con algunas de ellas, de manera que pueda decidir en cuáles quiero profundizar y en cuáles debería buscar otras alternativas.

Para la planificación del proyecto tenía en cuenta el conocimiento limitado sobre las tecnologías de desarrollo web, por lo que a la hora de planificar intenté estipular unos objetivos mínimos por el nivel de incertidumbre de lograr dominar los aspectos mínimos necesarios para la creación de la aplicación.

En general la programación ha sido bastante correcta, resaltando como incidencia que hubo que mezclar los desarrollos de la pantalla de operaciones y formularios, por aportar esta última información necesaria para completar la pantalla anterior. Aun así, se pudo respetar la fecha de finalización conjunta del hito.

Si bien la aplicación es funcional, la obligación de ceñirnos a unos determinados plazos ha ocasionado que haya algunos aspectos que he tenido que posponer hasta la entrega del proyecto, algunos de ellos están relacionados con mejoras en la funcionalidad detectados durante el desarrollo, otros en limitar de manera temporal una utilidad y algunos aspectos con la calidad del código producido.

En cuanto a mejoras de funcionalidad, he comprobado que hay una serie de medidas que podrían facilitar el uso de la aplicación de los usuarios, por ejemplo, una vez que un usuario ha creado una operación o plantilla si no introduce su nombre de forma correcta el sistema no las va a saber encontrar. En vez de esto, habría que modificar para que, al igual que ocurre cuando se hace un informe, conforme se escribe el nombre deseado el sistema proponga resultados para escoger el valor buscado.

Si nos referimos a las limitaciones de la funcionalidad, una de las ideas era que los formularios rellenos que se muestran en la operación se fueran actualizando conforme los usuarios los modifican o los envían a la base de datos, aunque por limitación de tiempo de momento solo muestran los que se encuentran en el momento de consultar la operación. Otra funcionalidad con los listados es ofrecer la agrupación de informes por valores de selección de forma que muestre totales por el valor seleccionado.

Una vez finalizada la memoria y el vídeo de presentación del trabajo dedicaremos tiempo para intentar avanzar lo más posibles hasta el momento de la entrega final del proyecto de las funcionalidades programadas y no implementadas comentadas. No he incluido aquellas mejoras que he visto interesantes surgidas al realizar las pruebas, ya que el proyecto corría el riesgo de entrar en una espiral en la cual nunca se da por validado el punto final inicial.



En cuanto a la calidad del código considero que es correcto, pero hay puntos interesantes en los que hay que profundizar como el uso de Vue para generar componentes de manera que, aunque se pierda tiempo ahora en su aprendizaje, se gane en el futuro al permitir su reutilización y la reducción del código introducido. Me gustaría estudiar si hay alguna alternativa mejor a Bootstrap o generar partes con CSS, puesto que reconozco que, si bien me ha facilitado el desarrollo, no me termina de convencer su funcionamiento, hay momentos en los que es muy complejo lograr que se comporte la presentación de los objetos como me gustaría. Creo que hay bastante margen para optimizar algunas partes recurrentes a lo largo de la aplicación (ej. llamadas a la función \$.Ajax) que merece la pena estudiar.

El producto obtenido es un buen punto de partida, pero hay una serie de trabajos que comienzan una vez finalizado la entrega del mismo para ampliar funcionalidades y mejorar su código. Es el momento de la apertura de un periodo de reflexión con el grupo de compañeros del área de comunicaciones para debatir cuáles son las líneas de trabajo que tengo que seguir en la evolución de la aplicación.

Ya he mencionado la necesidad de profundizar en el uso de Vue.js a través de componentes, pero también hay que evolucionar en otras mejoras como un segundo paquete de informes más evolucionados con la posibilidad de uso de gráficos, conexión con redes sociales, envío de correos... Hay que mejorar aspectos de la aplicación como el control de sesión de usuarios entre las diferentes pantallas, crear nuevos roles, algunos aspectos de visualización en las pantallas y el comportamiento de algunas consultas cuando exista un volumen considerable de datos.

Hay aspectos que me gustaría investigar y utilizarlos en la aplicación como el uso de Blockchain, la tecnología PWA (*Progressive Web Apps*) o el trabajo con sistemas para BigData cuando haya un volumen considerable de datos para la realización de estudios. La posibilidad de enlazar sistemas de adquisición de datos a través de hardware específico o *web services* para la obtención de datos.



7. ANEXOS.

7.1. Infografía.

Se adjunta al documento un .pdf con la infografía que se envió a los usuarios para tener información básica de cómo funcionaba la aplicación.

Cualquier usuario puede introducir formularios sin contraseña.

MENU SIREDACU.

Formularios

Usuario: Nombre de usuario.

Contraseña: Enter password

Enviar

Usuario y contraseña inicial: 'user'.

Niveles de acceso:

- * Super usuario.-> Acceso completo.
- * Plantillas.-> Crea Plantillas.
- * Operaciones.-> Crea Operaciones, Listar Informes
- * Anónimos -> Introducir formularios.

MENU PLANTILLAS.

Las plantillas permite declarar la matriz de introducción de datos.

Plantilla: Nombre plantilla.

Buscar

Añadir

Se introduce un nombre, pulsar 'Buscar', si no existe pulsar 'Añadir'.

Plantilla: Encierros Plantilla activa.

Descripción: Datos recogida en encierros.

Buscar

Quitar

Para que la plantilla esté disponible señalar 'Plantilla activa'.

Nº Ítems: 10 Indica los conceptos.

Nº Niveles: 2 Niveles división conceptos.

Ítem: Pisotones Valor.

+

-

sub

baj

Añadir. Quitar. Cambiar posición.

Pisotones

H. Asta

Mientras un ítem no se haya utilizado nunca, se puede eliminar.

tags:

Tag: SF2019

+

-

Etiquetas para plantillas.

'SF';SF2019';

Las etiquetas se usan para poder agrupar distintas operaciones con la misma plantillas en informes.

MENU OPERACIONES.

Asigna una plantilla a la operación.

Operación: Encierros SF 2019 Plantilla: Encierros

Buscar

Añadir

Descripción: Atenciones realizadas durante los encierros.

Fecha inicio: 07/07/2019 Fecha fin: 14/07/2019 Contraseña:

Se indica las fechas en la que se puede rellenar el formulario.

Se define la palabra de aceptación de formularios.

valor: Valor

+

-

sub

baj

Santo Domingo

Ayuntamiento

Mercaderes

Estafeta

Se crean campos para selección y se define su tipo entre: Texto o Fecha.

Para cada campo se asignan sus valores.

Solo lectura	Eliminar	Fecha.	Puesto.	Pisotones				
				Hombres		Mujere		
				< 18	18 - 25	> 25	< 18	18 - 25
<input type="checkbox"/>	Eliminar	2019-07-07T00:00	Santo Domingo					
<input type="checkbox"/>	Eliminar	2019-07-07T00:00	Ayuntamiento					

Controlar los formularios introducidos para modificar, eliminar o impedir su modificación.

FORMULARIOS.

Se selecciona una operación activa entre las fecha de inicio y fin definidas.

Se señala un valor para cada campo definido.

Introducir valores en la matriz diseñada en la plantilla seleccionada en la operación.

Colocar la palabra de aceptación indicada cuando se creó la operación.

Operación: Encierros SF 2019

Descripción: Atenciones realizadas durante los encierros.

Fecha: 2019-07-07T00:00 Puesto: Espoz y Mina

Pisotones	Hombres		Mujere		
	< 18	18 - 25	> 25	< 18	18 - 25
Plutonias					
H. Asta					
Sipontanias					
Cares, her.	1	2			
Españoles					
At. Médicas					
Susaciones					
Trasados					
Globos					
Otros					

Contraseña: Palabra de paso.

MENU INFORMES.

Los informes se realizan sobre operaciones o plantillas

Para acotar la búsqueda se señalan los valores que muestre.

Se pueden imprimir, o modificar los datos de selección.

Cuando se imprime los botones se ocultan.

Seleccione una operación o una plantilla:

Referencia Operación: Encierros SF 2019 Referencia Plantilla: Encierros

Seleccione una operación o una plantilla: Encierros SF 2019 Seleccione una operación o una plantilla: Encierros SF 2019

Fecha: 2019-07-08 00:00 Puesto: Santo Domingo Operaciones: Encierros SF 2019 Tag: Encierros SF SF2019

CRUZ ROJA NAVARRA - Encierros SF 2019

Operación	Fecha	Puesto	Valor	Valor	Valor
Encierros SF 2019	2019-07-07 00:00	Santo Domingo			
Encierros SF 2019	2019-07-07 00:00	Ayuntamiento			
Encierros SF 2019	2019-07-09 00:00	Mercaderes			
Encierros SF 2019	2019-07-10 00:00	Estafeta			

Ilustración 44 - Infografía del uso de SIREDACU.



7.2. Mejoras en la aplicación.

Fecha.	Tipo.	Descripción.	Solución.	Est.
15/11/19	M	En operaciones y plantillas cuando se quiere buscar y no se recuerda el nombre resulta difícil encontrar el elemento creado.	Conforme se escriba el nombre el sistema muestra aquellos que coincidan con lo escrito.	P
08/12/19	M	El identificador de usuario que se pasan entre las pantallas es fácil de suplantar.	Cambiar la forma en que se crean los identificadores de sesión de usuario.	P
08/12/19	M	Si un usuario no se acuerda de la contraseña no tiene un proceso de recuperación.	Crear un proceso de recuperación autónomo de contraseñas para los usuarios activos.	P
10/12/19	M	Poder copiar los datos de una operación o plantilla de forma que ahorre tiempo al copiar los valores de los campos.	Generar un botón en cada una de las pantallas que permita realizar dicha operación.	P
20/12/19	M	En los informes no se muestra un título para cada nivel de agrupación de datos.	Generar un campo cuando se crean los niveles en la plantilla que indique un título.	P
20/12/19	M	Cuando se eliminan los datos no se pide confirmación.	Crear diálogos de confirmación en las operaciones de eliminación de plantillas, operaciones, usuarios.	P
28/12/19	M	No se eliminan nunca los datos de la base de datos.	Crear procesos automatizados para eliminar al cabo de un determinado tiempo aquellos datos marcados como eliminados.	P
30/12/19	M	No se muestran en informes totales por los campos de selección.	Generar informes agrupados por los campos de selección.	P
01/01/20	M	Si se está en operaciones y hay un cambio en formularios no se refleja en el momento.	Comprobación periódica si hay cambios en formularios para mostrar los nuevos datos.	P
01/01/20	CI	Crear componentes en el código.	Revisar que implementaciones se pueden convertir en componentes.	P
01/01/20	CI	Unificar funciones AJAX.	Estudiar si se puede crear un componente o pequeña librería que simplifique su implementación en el programa.	P
01/01/20	CI	Autenticación de datos para que no se alteren.	Estudio de implementación de Blockchain.	P
01/01/20	M	Los informes solo son obtenidos por personal de la institución.	Uso de sistemas de envío automáticos a redes sociales tras la aprobación del contenido por parte de Marca y Comunicación.	P
05/01/20	CI	Se detecta alguna consulta a la base de datos repetida.	Revisar código para unificar consultas repetidas.	P
05/01/20	CI	Asegurar el cambio todo o nada en la base de datos.	Estudio de implementación de transacciones cuando hay varias consultas de modificación para un trabajo.	P

Tipo: M -> Mejora, Cambio Internos.

Estado: P -> Pendiente.



7.3. Instalación aplicación.

En este apartado se muestra la relación de ficheros que componen el Trabajo Final de Grado y como deben utilizarse.

- **Memoria.pdf** -> Contiene toda la documentación referente a la construcción del proyecto.
- **Siredacu.zip** -> Contiene los ficheros del desarrollo web que deben ser subidos a un hosting. En este caso los ficheros están subidos al subdominio <https://siredacu.crnavarra.org> tal y como vienen en el fichero comprimido. El punto de entrada en la aplicación se realiza a través del fichero que está en la raíz con el nombre index.html.

En la carpeta php existe un fichero llamado conector.php donde se pueden actualizar los parámetros de conexión a la base de datos como el usuario y el *password* del administrador/usuario del servidor de la base de datos donde se instala la aplicación.

- **DefinicionDB.sql** -> Contiene la definición de la base de datos para ser cargada en un servidor MaríaDB o MySQL.

Para ser cargada en la base de datos tan solo debemos entrar en phpMyAdmin y crear una base de datos llamada siredacu.

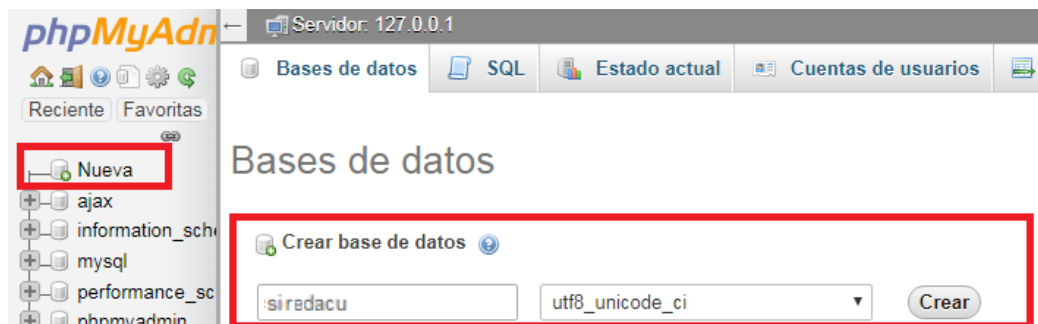


Ilustración 45 – Ejemplo en phpMyAdmin para dar de alta la base de datos SIREDACU.

Posteriormente en la pestaña importar seleccionar el fichero y pulsar el botón 'Continuar'.

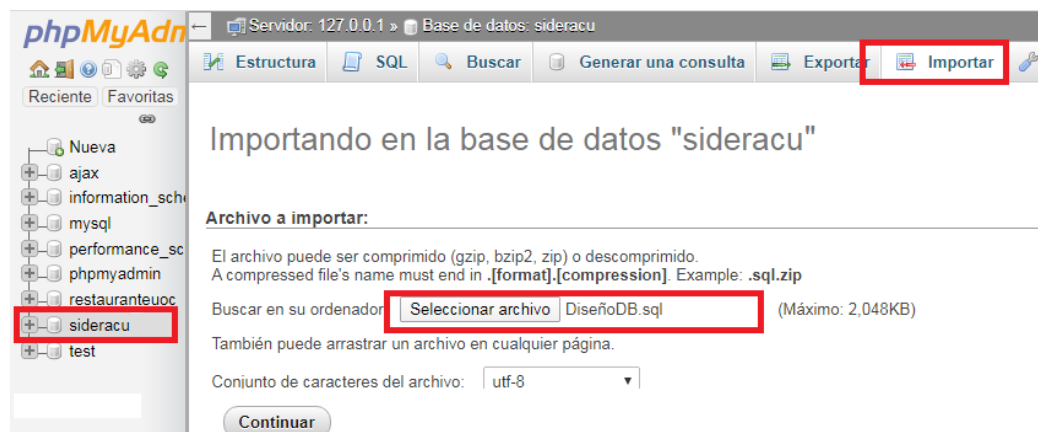


Ilustración 46 – Ejemplo en phpMyAdmin para importar la definición de tablas, campos, relaciones...

Automáticamente genera el administrador 'user' y contraseña 'user' para la aplicación.



- **PRESENTACION TFG OK COMPRIMIDO.mp4** -> Video con la presentación del Trabajo Final de Grado subido a la sección Presenta en el aula de la asignatura.

Dirección alternativa para revisarlo.

https://uoc0-my.sharepoint.com/:v:/g/personal/fmartinezrod_uoc_edu/EQVfPxoQIEIAvYyKnBgbAt4Bsd4LpYx5uKV1Dr3jir9nxQ?e=Sfjw2F

- **PRESENTACION TFG OK.mp4** -> Video con la presentación del Trabajo Final de Grado en alta definición.

Dirección para revisarlo.

https://uoc0-my.sharepoint.com/:v:/g/personal/fmartinezrod_uoc_edu/EVvcDRReEyK9CsrWVC1u6L5cBLg6le7B80TuFnNmXIV7loA?e=orTrZA

- **Infografía.pdf** -> Infografía enviada a los usuarios que hicieron pruebas donde se muestra de forma básica cómo usar la aplicación.



8. GLOSARIO

- Blockchain:** Cadena de bloques. Tecnología consistente en una base de datos distribuida y segura que verifica y valida registros de información.
- CDN:** *Content Delivery Network* (red de distribución de contenidos).
- CSS:** *Cascading Style Sheets* (hoja de estilos en cascada).
- DNS:** *Domain Name System* (sistema de nombres de dominio).
- DOM:** *Document Object Model* (Modelo de Objetos del Documento).
- Dominio:** Nombre que identifica de forma única un sitio en Internet.
- Feedback:** Retroalimentación a una acción realizada por parte del receptor.
- Hosting:** Espacio en Internet donde se aloja un sitio web.
- HTML:** *Hyper Text Markup Language* (lenguaje de marcas de hipertexto).
- LAMP:** Linux, Apache, MySQL/MariaDB, PHP. Acrónimo para indicar las herramientas que se ejecutan en un servidor web.
- Metodologías ágiles:** Enfoque metodológico para la toma de decisiones en proyectos de software basados en políticas de desarrollo iterativos e incremental.
- PHP:** *Hypertext Preprocessor* (procesador de hipertexto).
- PWA:** Progressive Web App. Aplicación web que se instala como si fuera nativa en un dispositivo.
- SQL:** *Structured Query Language* (lenguaje de consulta estructurada).
- Stakeholders:** Aquellos interesados en torno a la ejecución del proyecto.
- Subdominio:** Subclasificación del nombre del dominio para fines administrativos y organizativos.
- UML:** *Unified Modeling Language* (lenguaje modelado unificado).
- XP:** *Extreme Programming* (programación extrema).

TÉRMINOS PROPIOS DE LA APLICACIÓN

- Administradores:** Usuarios que administran la aplicación.
- Formulario:** Cada una de las tomas de datos que se realizan en una operación.
- Ítem:** Elemento sobre los que se quiere actuar recopilando datos sobre su uso.
- Organizadores:** Usuarios que crean operaciones.
- Operaciones:** Actividad en la que se desean recopilar datos sobre las acciones que se ejecutan utilizando una plantilla predefinida.
- Operarios:** Cada persona que rellena un formulario.
- Plantilla:** Definición de una relación de ítems y como se disgregan estos datos.
- SIREDACU:** Sistema de REcogida de DATos CUantitativos.
- Subítem:** Como están fragmentados los ítems.
- Tag:** Valor que se usa para identificar y agrupar plantillas y operaciones.



9. BIBLIOGRAFÍA

- Metodologías Ágil.

[1] Ágil & Scrum. Artículo del blog *Management Plaza*. Entrada: **Sabes cómo funciona XP**. Autor: *Juan Luis Vila Grau*. Fecha Publicación: 2017. <https://managementplaza.es/blog/sabes-como-funciona-xp/>

[2] Artículo de la enciclopedia on-line *Wikipedia*. Entrada: **Programación extrema**. Autores: varios, ver enlace página web. Última modificación: 30 ago 2019. https://es.wikipedia.org/wiki/Programaci%C3%B3n_extrema

[3] Libro: **Scrum y XP desde las trincheras**. Autor: *Henrik Kniberg*. Fecha Publicación: 2007. Editorial: *C4Media Inc*, ISBN: 978-1-4303-2262-1. Web Autor: <https://www.crisp.se/konsulter/henrik-kniberg>

[4] Sitio Web: Manifiesto for Agile Software Development. Entrada: **Manifiesto por el Desarrollo Ágil de Software**. Autor: *Varios autores*. Fecha Publicación: Web Publicación: <https://agilemanifesto.org/iso/es/manifesto.html>

- Desarrollo UML

[5] Libro: **UML Gota a Gota**. Autor: *Martin Fowler con Kendall Scott*. Fecha publicación: 1997. Editorial: *Addison Wesley Longman Inc*. ISBN: 0-201-32563-2. Web autor: <https://martinfowler.com>

[6] Libro: **Análisis UML**. Autor: *Jordi Pradel Miquel, Jose Raya Martos*. Fecha publicación: Editorial: *UOC. Universidad Oberta de Catalunya*.

[7] Video: **Diagrama de casos de uso**. *Universitat Politècnica de València - UPV* Autor: *Penades Gramage Maria Carmen*. Fecha: 04 oct 2017. Web Publicación: <https://www.youtube.com/watch?v=orvAkFFWo5o>

- Base de datos.

[8] Videos: **Curso de PHP/MySQL**. Autor: *Juan Diaz* – canal youtube: *pildorasinformáticas*. Fecha última actualización: 23 ene 2017. Web Publicación: <https://www.youtube.com/playlist?list=PLU8oAIHdN5BkinrODGXToK9oPAInJxmW>

[9] Artículo Sitio Web: *Diseño Web akus.net*. Entrada: **Tipos de datos en MySql para una base de datos SQL**. Autor: *Hugo Delgado*. Fecha Publicación: 31 ago 2019 Web Publicación: <https://disenowebakus.net/tipos-de-datos-mysql.php>

- Proyectos

[10] Libro: **Ejecución del proyecto**. Autor: *José Ramón Rodríguez*. Fecha publicación: Editorial: *UOC. Universidad Oberta de Catalunya*. Web autor:

[11] Libro: **Seguimiento y control del proyecto**. Autor: *José Ramón Rodríguez, Pere Mariné Jové*. Editorial: *UOC. Universidad Oberta de Catalunya*.

[12] Libro: **Guía de los fundamentos para la dirección de proyectos (guía del PMBOK)**. Autor: *Project Management Institute, Inc*. Fecha publicación: 2008. ISBN: 978-1-933890-72-2. Web: www.pmi.org



- Programación web.

[13] Videos: **Curso de Vue.js**. autor: *Ignacio Gutierrez* - Canal Youtube: *Bluuweb!*. Fecha: 8 nov 2018. Web Publicación:

<https://www.youtube.com/watch?v=GAQB7Y4X5fM&list=PLPI81lqbj-4J-gfAERGDCdOQtVgRhSvIT>

[14] Libro: **Vue.js in action**. Autor: *Erik Hanchett, Benjamin Listwon*. Fecha publicación: 2018. ISBN: 9781617294624. Editorial: *Manning Publications Co*. Web: www.pmi.org

[15] Videos: **Bootstrap 4! Curso Completo**. autor: *Ignacio Gutierrez* - Canal Youtube: *Bluuweb!*. Fecha última actualización: 7 jul 2017. Web Publicación:

<https://www.youtube.com/playlist?list=PLPI81lqbj-4lcaAluUICTmbYz0h9XQ8U1>

[16] Sitio Web: *Vue.js* Entrada: **El Framework JavaScript Progresivo**. Web Publicación:

<https://es-vuejs.github.io/vuejs.org/>

[17] Artículo Sitio Web: *Hackerthemes*. Entrada: **Bootstrap 4 Cheat Sheet**. Autor: *Alexander Rechsteiner*. Fecha Publicación: Web Publicación: <https://hackerthemes.com/bootstrap-cheatsheet>

[18] Videos: **Javascript Async Await, Promesas y Callbacks**. autor: - Canal Youtube: *Fazt!*. Fecha última actualización: 2 dic 2018. Web Publicación:

<https://www.youtube.com/watch?v=Q3HtXuDEy5s>

[19] Artículo Sitio Web: *todo JS*. Entrada: **Controlar la ejecución asíncrona**. Autor: . Fecha

Publicación: 27 sep 2016. Web Publicación: <https://www.todojs.com/controlar-la-ejecucion-asincrona/>

- Herramientas.

[20] Videos: **Git y Github | Curso Práctico de Git y Github Desde Cero**. autor: - Canal Youtube: *Fazt!*. Fecha última actualización: 1 ene 2018. Web Publicación:

<https://www.youtube.com/watch?v=HiXLkL42tMU>

[21] Videos: **Tutorial Como Utilizar Git en Visual Studio code [4/5]**. autor: - Canal Youtube: *FalconMasters*. Fecha última actualización: 25 mar 2018. Web Publicación:

<https://www.youtube.com/watch?v=jPQQISFOkRE>

[22] Videos: **Mejores Plugins de Visual Studio Code para desarrollo web - VS Code**. autor: *Carlos Azaustre* - Canal Youtube: *Carlos Azaustre*. Fecha última actualización: 5 sep 2017.

Web Publicación: <https://www.youtube.com/watch?v=CeAvbVpBWR4>

- Varios

[23] Artículo Sitio Web: *Xataka*. Entrada: **Microsoft quiere que dejemos de usar Internet Explorer cuanto antes: "no es un navegador, es una solución de compatibilidad"**. Autor: *Raúl Alvarez*. Fecha Publicación: 8 feb 2019. Web Publicación:

<https://www.xataka.com/aplicaciones/microsoft-quiere-que-dejemos-usar-internet-explorer-cuanto-antes-no-navegador-solucion-compatibilidad>

[24] Videos: **CURSO HITFILM EXPRESS**. autor: - Canal Youtube: *Lobinshare*. Fecha: 14 feb

2019. Web Publicación: https://www.youtube.com/watch?v=eQ-8OyBDPvo&list=PLu5LXaMUM_E2hUUmGzJYH6MFI0rd6FytV



- Estado del arte.

[25] Artículo Sitio Web: *código facilito*. Entrada: **Angular, React O Vue ¿Cuál Elegir?** Autor: *Uriel Hernandez*. Fecha Publicación: *11 mar 2019* Web Publicación: <https://codigofacilito.com/articulos/angular-react-vue>

[26] Artículo Sitio Web: *stackoverflow*. Entrada: **Developer Survey Results 2019**. Autor: Fecha Publicación: Web Publicación: <https://insights.stackoverflow.com/survey/2019>

[27] Artículo Sitio Web: *OpenWeibinars*. Entrada: **Vue JS vs Angular: comparación entre líder y aspirante**. Autor: *Daniel Ortego Delgado* Fecha Publicación: *22 ago 2017*. Web Publicación: <https://openwebinars.net/blog/vue-vs-angular/>

[28] Artículo Sitio Web: *OpenWeibinars*. Entrada: **Diferencias entre Vue.js y React**. Autor: *Daniel Ortego Delgado* Fecha Publicación: *19 sep 2017*. Web Publicación: <https://openwebinars.net/blog/diferencias-entre-vuejs-y-react/>

[29] Artículo Sitio Web: *Gartner*. Entrada: **Gartner Says the Future of App Development Is Multiexperience**. Autora: *Katie Costello* Fecha Publicación: *04 abr 2019*. Web Publicación: <https://www.gartner.com/en/newsroom/press-releases/2019-04-04-gartner-says-the-future-of-app-development-is-multiex>

[30] Artículo Sitio Web: *Gartner*. Entrada: **Gartner Top 10 Strategic Technology Trends for 2020**. Autora: *Kasey Panetta* Fecha Publicación: *21 oct 2019*. Web Publicación: <https://www.gartner.com/smarterwithgartner/gartner-top-10-strategic-technology-trends-for-2020/>

[31] Artículo Sitio Web: *codeburst.io* Entrada: **10 top Programming Languages in 2019 for Businesses**. Autora: *Ruchika Singh Aggarwa* Fecha Publicación: *12 jun 2019*. Web Publicación: <https://codeburst.io/10-top-programming-languages-in-2019-for-developers-a2921798d652>

[32] Artículo Sitio Web: *Youtube*. Entrada: **Desarrollo web en 2019: Futuro y Tendencias** 📌. Autor: *Victor Robles*. Fecha Publicación: *2 ene 2019*. Web Publicación: https://www.youtube.com/watch?v=sz_yzU2vLXI

[33] Artículo Sitio Web: *Youtube*. Entrada: **Desarrollo Web en el 2019**. Autor: *Fazt*. Fecha Publicación: *29 dic 2018*. Web Publicación: <https://www.youtube.com/watch?v=w2r2tTBAJIE>

[34] Artículo Sitio Web: *Youtube*. Entrada: **Lenguajes y tecnologías más rentables para 2019 | PlatziLive**. Autor: *Freddy Vega - Platzi*. Fecha Publicación: *27 dic 2018*. Web Publicación: <https://www.youtube.com/watch?v=FIOLc2DaSToI>

[35] Artículo Sitio Web: *Xataka*. Entrada: **Así es como Oracle ha sometido a Sun Microsystems a una muerte larga y agónica**. Autor: *Javier Pastor*. Fecha Publicación: *11 sep 2017*. Web Publicación: <https://www.xataka.com/historia-tecnologica/asi-es-como-oracle-ha-sometido-a-sun-microsystems-a-una-muerte-larga-y-agonica>