

Maria del Roser Sagrera Perelló

**Desarrollo de una herramienta interactiva para mejorar
la visualización de datos en el ámbito clínico**

Trabajo de Fin de Grado

Dirigido por Joana Cañellas Santos

dirigido por la Dra Maria Ferré Bergadà

Grado en Ingeniería Biomédica



UNIVERSITAT ROVIRA I VIRGILI

Tarragona 2023

Resumen

En los últimos años, se ha trabajado a nivel mundial para crear una mayor consciencia sobre la gestión adecuada de los eventos adversos (EA) en el sistema sanitario, con el objetivo de mejorar la seguridad del paciente. Tal y como afirman diversos estudios, existe un panorama preocupante en términos de seguridad del paciente, por lo que se está trabajando para hacer frente a esta situación.

Una de las medidas fundamentales para abordar esta problemática es contar con un sólido sistema de notificación de incidentes que permita registrar y gestionar eficientemente las incidencias, así como para asignar los recursos necesarios para encararlas de manera adecuada.

En esta conjetura, este trabajo se centra en el desarrollo de una plataforma web que recopila y visualiza algunos de los datos del Sistema de Notificación de Incidentes de Seguridad de los Pacientes de Cataluña (SNiSP Cat), con el objetivo de proporcionar una herramienta visual e intuitiva para los profesionales de la sanidad, ayudando así a mejorar su toma de decisiones como también a implementar nuevas medidas para mejorar la seguridad del paciente.

La plataforma web se ha desarrollado utilizando la infraestructura de la plataforma de Seguridad de los Pacientes de Cataluña (PSP Cat), que está formada por el Sistema de Notificación de Incidentes y Seguridad de los Pacientes de Cataluña (SNiSP Cat), la aplicación Proactiva de Seguridad de los Pacientes de Cataluña (PROSP Cat) y el Cuadro de Mando de Seguridad de los Pacientes en Cataluña (QCSP Cat). Este proyecto forma parte de una iniciativa más amplia que se está llevando a cabo por el Hospital Universitario Joan XXIII de Tarragona.

Palabras clave: seguridad del paciente; notificación de incidencias; SNiSP; plataforma web; mapa visual.

Abstract

In recent years, global efforts have been made to raise awareness about the proper management of adverse events (AE) in the healthcare system, with the aim of improving patient safety. As various studies affirm, there is a worrisome landscape regarding patient safety, and concerted efforts are being made to address this situation.

One crucial measure to tackle this issue is to establish a robust incident notification system that enables the efficient registration and management of incidents, as well as the allocation of necessary resources to address them appropriately.

In this context, this work focuses on the development of a web-based platform that collects and visualizes selected data from the Patient Safety Incident Notification System of Catalonia (SNiSP Cat). The goal is to provide a visual and intuitive tool for healthcare professionals, thereby enhancing their decision-making processes and facilitating the implementation of new measures to improve patient safety.

The web platform has been developed using the infrastructure of the Patient Safety of Catalonia Platform (PSP Cat), which comprises the Patient Safety Incident Notification System of Catalonia (SNiSP Cat), the Proactive Patient Safety Application of Catalonia (PROSP Cat), and the Patient Safety Dashboard in Catalonia (QCSP Cat). This project is part of a broader initiative carried out by the Joan XXIII University Hospital in Tarragona.

Keywords: patient safety; incident reporting; SNiSP; web platform; visual map.

Resum

En els últims anys, s'ha treballat a nivell mundial per a crear una major consciència sobre la gestió adequada dels esdeveniments adversos (EA) en el sistema sanitari, amb l'objectiu de millorar la seguretat del pacient. Tal com afirmen diversos estudis, existeix un panorama preocupant en termes de seguretat del pacient, per la qual cosa s'està treballant per a fer front a aquesta situació.

Una de les mesures fonamentals per a abordar aquesta problemàtica és comptar amb un sòlid sistema de notificació d'incidents que permeti registrar i gestionar eficientment les incidències, així com per a assignar els recursos necessaris per a encarar-les de manera adequada.

En aquesta conjectura, aquest treball se centra en el desenvolupament d'una plataforma web que recopila i visualitza alguns de les dades del Sistema de Notificació d'Incidents de Seguretat dels Pacients de Catalunya (SNiSP Cat), amb l'objectiu de proporcionar una eina visual i intuïtiva per als professionals de la sanitat, ajudant així a millorar la seva presa de decisions com també a implementar noves mesures per a millorar la seguretat del pacient.

La plataforma web s'ha desenvolupat utilitzant la infraestructura de la Plataforma de Seguretat dels Pacients de Catalunya (PSP Cat), que està formada pel Sistema de Notificació d'Incidents i Seguretat dels Pacients de Catalunya (SNiSP Cat), l'aplicació Proactiva de Seguretat dels Pacients de Catalunya (PROSP Cat) i el Quadre de Comandament de Seguretat dels Pacients a Catalunya (QCSP Cat). Aquest projecte forma part d'una iniciativa més àmplia que s'està duent a terme per l'Hospital Universitari Joan XXIII de Tarragona.

Paraules clau: seguretat del pacient; notificació d'incidències; SNiSP; plataforma web; mapa visual.

Agradecimientos

Este proyecto no hubiera sido posible sin la ayuda de mi tutora, Joana Cañellas, a quien le agradezco la paciencia y dedicación durante el proceso. También ha sido indispensable la ayuda y acompañamiento y consejos de mi tutora del TFG, Maria Ferré.

Además, agradecer a la Unidad de Calidad y Seguridad del Paciente del Hospital Universitario Joan XXIII de Tarragona por darme la oportunidad de poder aportar mi granito de arena en un proyecto así.

Y, por último, pero no menos importante, gracias a mis compañeras y amigas, a Cris y a mi familia por estar siempre a mi lado y apoyarme en todo lo que me propongo.

Índice de figuras

1. Evolución (número) de las notificaciones recibidas en el sistema de notificación del Departamento de Salud des de mayo de 2013 hasta diciembre de 2016 [11].	5
2. Evolución del número de notificaciones en AP (extraídas el 13 de julio de 2021) [13].	6
3. Plataforma de Seguridad de los Pacientes de Cataluña [16].	7
4. Archivos y directorios del proyecto Django "SNiSP_Project". Captura realizada desde Visual Studio Code.	12
5. Archivos y directorios de la aplicación "snisp" del proyecto Django "SNiSP_Project". Captura realizada desde Visual Studio Code.	13
6. Archivos y directorios de la aplicación "auth_app" del proyecto Django "SNiSP_Project". Captura realizada desde Visual Studio Code.	14
7. Esquema del funcionamiento de Django. Captura extraída del curso Udemy "Python and Django Full Stack Web Developer Bootcamp".	16
8. Diagrama del modelo entidad-relación de la base de datos snisp.	17
9. Captura de pantalla del favicon de la web.	18
10. Captura de pantalla de la navbar principal con el logotipo corporativo de Gencat.	18
11. Captura de pantalla de la navbar secundaria con el logotipo de la aplicación.	19
12. Captura de pantalla del login para acceder a la plataforma.	19
13. Captura de pantalla de la página principal de la web.	20
14. Captura de pantalla de la página principal de la web. Solamente a la gráfica interactiva "Implatació EAP". Seleccionando los implantados, situando el ratón encima de la región sanitaria de Barcelona.	21
15. Captura de pantalla de la página principal de la web. En el mapa visual donde se ha seleccionado la región sanitaria de "Lleida".	21
16. Captura de pantalla de la página de la región sanitaria de Tarragona.	22
17. Captura de pantalla de la página de la región sanitaria de Tarragona. (Zum de la parte derecha).	23
18. Captura de pantalla de la página de la región sanitaria de Tarragona. En el mapa visual donde se ha pasado por encima del EAP de Falset.	23
19. Captura de pantalla de la página de la región sanitaria de Tarragona. Solamente a la gráfica interactiva. Situando el ratón encima de la parte de los EAP implantados.	24
20. Captura de pantalla de la página de la región sanitaria de Barcelona.	29
21. Captura de pantalla de la página de la región sanitaria de "Terres de l'Ebre".	29
22. Captura de pantalla de la página de la región sanitaria de "Alt Pirineu i Aran".	29
23. Captura de pantalla de la página de la región sanitaria de "Lleida".	29
24. Captura de pantalla de la página de la región sanitaria de "Catalunya Central".	29
25. Captura de pantalla de la página de la región sanitaria de "Girona".	29

Índice de cuadros

1. Ejemplo del dataframe "df_implantats"..... 17
2. Ejemplo del dataframe "df_merged_implantats". Unión de los dos dataframe mediante el nombre del centro..... 17

Índice de abreviaturas

AP Atención Primaria.

APEAS En España contamos con el estudio sobre la Seguridad de los Pacientes en Atención Primaria (APEAS) como referencia de estudio epidemiológico en AP.

CAI Centros de Atención Intermedia.

CSS *Cascading Style Sheets.*

EA Efecto Adverso.

EAP Equipo de Atención Primaria.

ENEAS Estudio Navional sobre los Efectos Adversos

HA Hospital de Agudos.

HTML *HyperText Markup Language.*

HTTP *Hypertext Transfer Protocol.*

IBEAS Estudio de prevalencia de efectos adversos en hospitales de Latinoamérica.

ICS Instituto Catalán de la Salud.

IOM *Institute of Medicine.*

OMS Organización Mundial de la Salud.

PROSP Cat Aplicación Proactiva de Seguridad de los Pacientes de Cataluña.

PSP Plataforma de Seguridad del Paciente.

QCSP Cat Cuadro de Mando de Seguridad de los Pacientes en Cataluña.

SEM Servicios de Emergencias Médicas.

SNiSP Cat Sistema de Notificación y Seguridad del Paciente de Cataluña.

SP Seguridad del Paciente.

SQL *Structured Query Language.*

TSPC Cloud Aplicación de notificaciones genérico para todo tipo de incidentes relacionados con la salud del paciente.

UQSP *Unitat de Qualitat i Seguretat del Pacient*

VIH Virus de Inmunodeficiencia Humana.

XHUP *Xarxa d'Hospitals d'Utilització Pública.*

Índice

1	Introducción.....	1
1.1	Objetivos del proyecto	2
1.2	Estructura del documento	2
2	Seguridad del paciente	3
2.1	Marco teórico.....	3
2.2	Sistema de Notificación de Incidentes	4
3	Análisis de requerimientos	8
4	Metodología	9
4.1	Entorno de programación.....	9
4.1.1	Herramientas de programación	9
4.1.1.1	<i>Frontend</i>	9
4.1.1.2	<i>Backend</i>	10
4.1.1.3	Librerías.....	11
4.1.2	<i>Framework Django</i>	11
4.2	Bases de datos.....	16
4.2.1	Servidor Snisp	16
5	Resultados	18
5.1	Apartados de la plataforma web	18
6	Limitaciones	25
7	Futuras mejoras.....	26
8	Conclusión	27
9	Referencias	28
	Anexo 1. <i>Templates</i> de cada región sanitaria	29
	Anexo 2. Código de programación	30

1 Introducción

En los últimos años, se han llevado a cabo diversos estudios relacionados con la seguridad del paciente (SP), entre ellos destacan APEAS (Estudio sobre la seguridad de los Pacientes en Atención Primaria), ENEAS (Estudio Nacional de Efectos Adversos ligados a la Hospitalización) e IBEAS (Prevalencia de Efectos Adversos en Hospitales de Latinoamérica), entre otros. Estas investigaciones han desempeñado un papel fundamental al proporcionar un mayor conocimiento y arrojar luz sobre la incidencia de los eventos adversos (EA) en el sistema sanitario español. Los resultados de estos estudios han revelado un panorama preocupante en términos de seguridad del paciente, generando una conciencia renovada en nuestro país y situando la gestión de los eventos adversos como una prioridad de gran importancia en el ámbito de la salud.

El creciente número de mejoras relacionadas con la SP destaca la importancia de contar con un sólido sistema de notificación de incidentes. Un sistema que permita registrar y gestionar las incidencias de forma eficiente, con el objetivo de mejorar en todos los aspectos y asignar adecuadamente los recursos necesarios.

En este contexto, la Unidad de Calidad y Seguridad del Paciente (UQSP) del Hospital Universitario Joan XXIII de Tarragona está desarrollando una plataforma web innovadora. Con el objetivo principal de mostrar los lugares donde se ha implementado el sistema actual de notificación y gestión de incidentes a través de un mapa interactivo.

La web se está llevando a cabo a partir de la plataforma de Seguridad de los Pacientes de Cataluña (PSP Cat) que está formada por el Sistema de Notificación de Incidentes de Seguridad de los Pacientes de Cataluña (SNiSP Cat), la aplicación Proactiva de Seguridad de los Pacientes de Cataluña (PROSP Cat) y el Cuadro de Mando de Seguridad de los Pacientes en Cataluña (QCSP Cat).

El presente trabajo final de grado se llevó a cabo en el Hospital Universitario Joan XXIII de Tarragona, concretamente en UQSP, como resultado de mi participación en prácticas en dicha institución. El objetivo principal del trabajo ha sido el desarrollo de la plataforma web para SNiSP Cat, el cual surge de un proyecto ya iniciado que se está llevando a cabo por mi tutora de TFG, Joana Cañellas. Éste trata de implementar y desarrollar una plataforma web para la aplicación PROSP Cat.

El objetivo del proyecto es combinar las dos plataformas, SNiSP y PROSP, con el fin de consolidar una única plataforma independiente y completa.

1.1 Objetivos del proyecto

El objetivo principal del proyecto es realizar una plataforma web para la aplicación SNiSP Cat. Respecto a los objetivos específicos, consisten en adquirir conocimientos relacionados con el desarrollo de aplicaciones web, como, por ejemplo:

- Conocer los lenguajes de programación Python, CSS, HTML, JavaScript. Además de otras librerías como Bootstrap, JQuery.
- Familiarizarse con el *framework* Django, una herramienta ampliamente usada en el desarrollo web.
- Comprender y distinguir los conceptos de *backend* y *frontend*. Aprender a diseñar y desarrollar tanto el *backend* como el *frontend* de una plataforma web.
- Aprender los conocimientos relacionados con bases de datos mediante el uso del lenguaje SQL (*Structured Query Language*).
- Utilizar herramientas y tecnologías relacionadas con el desarrollo web, como la terminal prompt de Anaconda y GitHub.

1.2 Estructura del documento

El proyecto se estructura en nueve capítulos, que se describen a continuación:

- En el presente capítulo se realiza una breve introducción que explica la temática del proyecto relacionado con la seguridad del paciente.
- En el capítulo dos se realiza la descripción del marco teórico y contexto relacionado con la seguridad del paciente y los sistemas de notificación de incidentes en los últimos años.
- El tercer capítulo se centra en el análisis de requerimientos que abarca nuestra plataforma, detallando las necesidades y funcionalidades requeridas.
- En el cuarto capítulo se describe la metodología utilizada, incluyendo las herramientas de programación y la base de datos usada en el desarrollo del proyecto.
- En el quinto capítulo se presentan los resultados obtenidos en relación con la plataforma web, explicando el proceso de desarrollo paso a paso.
- En el capítulo seis se detallan las limitaciones que surgieron durante la realización del proyecto y como se abordaron.
- En el capítulo siete se comentan algunas de las posibles mejoras que podrían llevarse a cabo para mejorar el proyecto en el futuro.
- En el capítulo ocho se presenta la conclusión obtenida después de finalizar el proyecto, resumiendo los hallazgos y logros alcanzados.
- En el último capítulo, el nueve, se listan las referencias bibliográficas y fuentes utilizadas en el trabajo.

2 Seguridad del paciente

Según la definición de la Organización Mundial de la Salud (OMS) la seguridad del paciente es "la ausencia de daño no intencional para el paciente causado por la atención sanitaria que se brinda o por la falta de esta atención" [1].

La seguridad del paciente (SP) es un tema de gran importancia en el ámbito de la atención médica y la prestación de servicios de salud, además de ser uno de los cimientos en los que se ha apoyado la práctica clínica en todos sus escenarios de aplicación (hospitales, centro de salud...) [2]. El movimiento de SP comenzó con el lanzamiento del informe del *Institute of Medicine* (IOM) "To Err Is Human" a finales de 1999 [3]. Basándose en las conclusiones de un trabajo realizado una década antes, el IOM adoptó una posición clara, "el nivel de daño no intencionado en la medicina es inaceptablemente alto y se debe actuar al respecto". A partir de ahí se planteó el desafío de lanzar un gran esfuerzo para cambiar los sistemas mal diseñados y dañinos para hacerlos más seguros [4]. El punto de partida para promover ese cambio fue plantear la SP como un punto muy necesario para conseguir mejoras en la sanidad.

2.1 Marco teórico

Actualmente podemos decir que la SP es fundamental para prestar servicios sanitarios esenciales de calidad. De hecho, existe un claro consenso de que los servicios de salud de calidad en todo el mundo deben ser eficaces y seguros y estar centrados en las personas. Además, para que los beneficios de una atención sanitaria de calidad sean efectivos, los servicios de salud deben prestarse de manera oportuna, equitativa, integrada y eficiente [5].

En cuanto a las causas de la SP se destacan los problemas de seguridad, los cuales se producen como consecuencia de múltiples sucesos, evitando una interpretación simplista en la que el error humano es la única explicación posible. Como alternativa se propone un enfoque más sistémico, dónde frecuentemente encontramos que después del análisis en profundidad del problema de seguridad existen causas relacionadas con la comunicación, la ergonomía, la fatiga, la dotación de personal, la supervisión o la formación inadecuadas [6].

Los incidentes y accidentes son precedidos por algún tipo de "acción insegura" que suele conllevar a que alguien cometa un error [6] [7], este debe ser analizado para saber cómo ocurrió y examinar las "condiciones latentes" del sistema que ha dado lugar a esa acción insegura. Se pretende destacar también que existen factores contribuyentes que forman parte del análisis de cualquier evento adverso, por ejemplo: factores humanos, en los que encontramos los turnos, la fatiga, la capacitación; factores del proceso, se trata de un análisis de los diferentes fallos en los pasos del proceso; equipamiento, dónde encontramos el mantenimiento reactivo y proactivo de la maquinaria; factores ambientales, ya son el ruido, espacio, mobiliario, etc.; gestión de la información, que trata la comunicación intra y extra del equipo; el liderazgo, y finalmente, la supervisión adecuada. Por lo tanto, después de haber nombrado esta serie de factores que contribuyen a una inseguridad en un entorno sanitario podríamos decir que muchos eventos adversos se deben a situaciones relacionadas con el propio sistema y no a negligencias o imprudencias del personal [6].

En cuanto a las consecuencias de esta inseguridad en la sanidad, una gran cantidad de pacientes sufren lesiones o mueren debido a una atención sanitaria poco segura y de mala calidad. Algunas de las prácticas o situaciones médicas que más preocupan en relación con la SP son los errores de medicación, las infecciones relacionadas con la atención sanitaria (afectan a 7 de cada 100 pacientes hospitalizados en países de ingresos altos y 10 de cada 100 pacientes hospitalizados en países de ingresos bajos y medios); los procedimientos quirúrgicos poco seguros que provocan complicaciones en hasta el 25% de los pacientes; las prácticas de inyección poco seguras en lugares de atención sanitaria, las cuales pueden llegar

a transmitir infecciones, como el VIH y los virus de la hepatitis B y C, y que también plantean un peligro directo para los pacientes y los profesionales sanitarios; los errores diagnósticos que afectan aproximadamente al 5% de los adultos que reciben atención ambulatoria; las prácticas de transfusión poco seguras; errores de irradiación que abarcan los casos de sobreexposición a la radiación y aquellos en los que se irradia al paciente incorrecto o la zona equivocada [5].

Otra práctica preocupante es que la septicemia no sea diagnosticada a tiempo y, en el caso que si lo sea, se podrían evitar daños mayores como la muerte y la tromboembolias venosas, que son una de las causas más frecuentes y previsibles de daños a los pacientes [5].

2.2 Sistema de Notificación de Incidentes

El error humano es inevitable, especialmente en un sistema tan complejo como el de la atención médica. Simplemente luchar por la perfección o castigar a las personas que cometen errores no mejorará significativamente la seguridad, porque no es realista esperar la perfección de las personas que trabajan en entornos complejos y de alta presión [8].

Por tal de evitar posibles incidentes y reducir el riesgo de deslices es necesario prestar atención al diseño de protocolos, equipos y entornos de trabajo. Un ejemplo, sería el uso de listas de verificación, que se trata de listas algorítmicas de acciones que se deben realizar en un entorno clínico determinado, con el objetivo de garantizar que no se olvide ningún paso [9], para evitar omitir pasos críticos y puntos clave de los procesos, así como la implementación de funciones que sean obligatorias para minimizar las soluciones alternativas, eliminando las distracciones de las áreas de trabajo que requieran concentración e implementando otras técnicas de rediseño [8].

Durante los últimos años se han ido haciendo mejoras relacionadas en la SP y cada vez se le da más importancia al hecho de registrar estos incidentes para poder llevar un control más exhaustivo de los mismos, que luego permitirá estudiar donde se podría mejorar y así reducir el riesgo. Establecer un sistema de notificación y registro de incidentes es el único modo de poder identificar el número total de incidentes, que es un requisito fundamental para mejorar la salud del paciente. Sin esta información, las organizaciones locales no pueden enfocar y asignar de manera adecuada sus recursos para garantizar el cambio [10]. Según el informe mencionado anteriormente del IOM, establecer los sistemas de notificación contribuyen a una estrategia clave para aprender de los errores y evitar su recurrencia [11].

Los sistemas de notificación y gestión de incidentes están diseñados para mejorar la seguridad y minimizar la posibilidad de que se repita un incidente. A partir de las notificaciones recibidas, es posible conocer cuáles son las debilidades del sistema y los factores que provocan esta falta de seguridad, además de formular medidas de mejora concretas. Los incidentes que tienen relación con la seguridad de los pacientes son aquellos acontecimientos o circunstancias que ocasionan, o podrían ocasionar, un daño innecesario relacionada con la atención sanitaria del paciente y que podría haberse evitado. En Catalunya, durante el año 2012, en el marco del actual Servicio de Promoción de la Calidad y la Bioética de la Dirección General de Ordenación Profesional y Regulación Sanitaria del Departamento de Salud, se impulsó un modelo de seguridad del paciente, inicialmente en hospitales de agudos y Equipos de Atención Primaria (EAP). Este modelo está diseñado para alcanzar la excelencia en la seguridad del paciente, especialmente en áreas críticas de los centros sanitarios, y minimizar lo máximo posible el riesgo al daño innecesario asociado a la atención prestada [11].

En aquel momento, siguiendo las directrices de los principales organismos nacionales e internacionales que velan por la seguridad del paciente, dentro de ese modelo de seguridad del Departamento de Salud, se consideró necesario la implantación de un sistema común de notificación y gestión de incidentes que fuese común en todos los centros. El sistema utilizado es el llamado TPSC Cloud™, que se trata de un sistema de notificación genérico para todo

tipo de incidentes relacionados con la Seguridad del paciente, voluntario, confidencial, no punitivo, anónimo, con orientación sistémica y con análisis local de los casos, que permite el aprendizaje y la implementación de mejoras en el ámbito del centro sanitario. Fue implantado en todos los hospitales de la red hospitalaria de uso público (XHUP) y consorcios, y se empezó a desplegar de manera progresiva en el ámbito de la atención primaria en Cataluña, hasta ahora [11].

La plataforma TPSC Cloud™ fue implantada con el objetivo de conocer los tipos de incidentes, saber quién, y de dónde se notifican, el grado de riesgo, a quien involucró y si se llegó a afectar al paciente [12]. Las principales funciones de esta eran: la notificación, que permitía notificar de una manera estructurada a partir de diferentes tipos de cuestionarios según el tipo de incidente (medicación, caídas, etc.); gestión, que pretendía realizar una gestión proactiva mediante herramientas (matriz de riesgos, análisis de causa raíz, análisis de causa efecto, análisis de barreras, análisis de procesos, etc.) de forma íntegra en la plataforma; análisis e informes, que permite analizar los incidentes y prevenir los errores, y finalmente, posibles acciones de mejora, para poder definir, planificar y hacer un seguimiento de las acciones de mejora, medidas de prevención o cambios en la organización [11].

Se fueron haciendo estudios y revisando los incidentes que se iban notificando mediante esta plataforma con el objetivo de mejorar y obtener los datos más reales posibles para así, posteriormente, poder realizar mejoras. Se observó que durante su puesta en marcha en mayo de 2013 hasta el 3 de diciembre de 2016 se obtuvieron un total 21.222 notificaciones de incidentes relacionados con la seguridad del paciente. Los incidentes fueron clasificados según los ámbitos donde se registraban las notificaciones y también según su evolución en años. En la siguiente figura 1 observamos la evolución de las notificaciones recibidas desde mayo del 2013, cuando fue implantado, hasta diciembre de 2016.

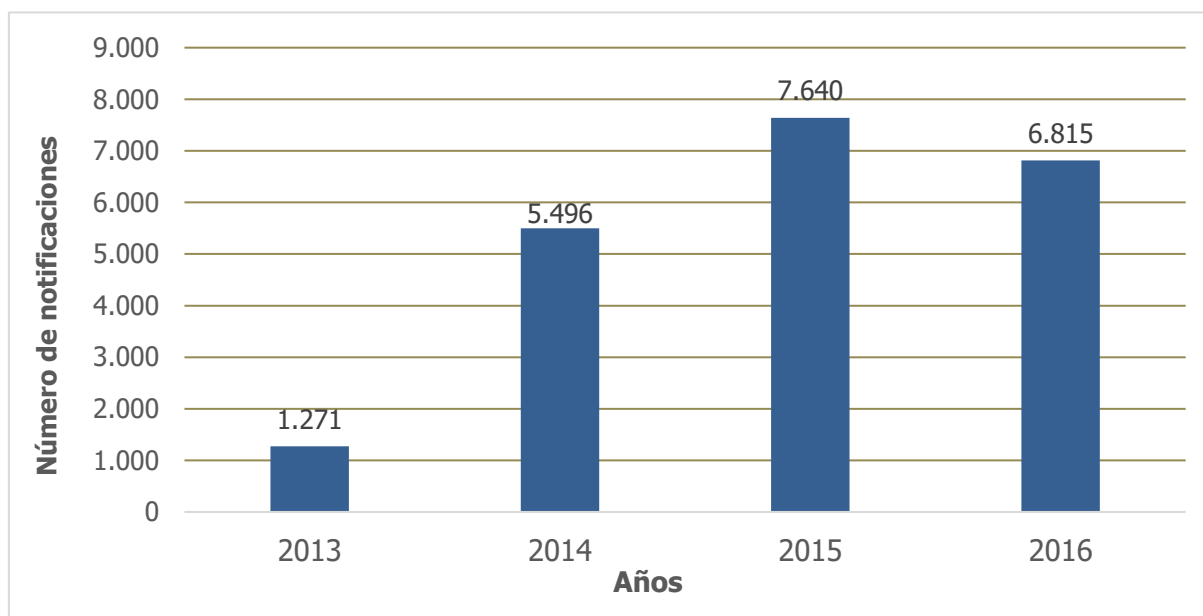


Figura 1. Evolución (número) de las notificaciones recibidas en el sistema de notificación del Departamento de Salud desde mayo de 2013 hasta diciembre de 2016 [11].

Las notificaciones provienen mayormente de hospitales, sin embargo, se consideró de gran importancia llevar a cabo una vigilancia activa del aumento de las incidencias notificadas que se estaban presentando en el ámbito de la atención primaria [11].

La SP es un tema crítico en la atención sanitaria y es especialmente importante en AP, por eso se empezaron a ver algunos cambios a partir del 2015, es decir, el año que se diseñó un formulario específico para notificar los incidentes de forma más efectiva y más real en la AP [13].

AP es el nivel asistencial más próximo y accesible, donde se establece el primer contacto con el sistema sanitario y se presta una atención continuada a lo largo de toda la vida de la persona, de manera longitudinal e integral [14]. El adecuado funcionamiento de AP garantiza un sistema sanitario de mayor calidad, más seguro, y más eficiente. Se trata de un nivel sanitario muy accesible y por lo tanto más propenso al colapso, situación que conlleva a que los pacientes acudan a otros niveles y se haga un mal uso de los servicios de urgencias hospitalarios, ineficientes y con más riesgos. O, por otra parte, que este colapso lleve a un retraso en la atención, que puede suponer retrasos en la evaluación, diagnóstico y tratamiento [14].

Durante la pandemia, se tomó conciencia acerca de la necesidad imperante de invertir en estrategias que fortalezcan y consoliden la AP como un elemento prioritario de la agenda política, debido a los múltiples beneficios que ofrece a la población en general, así como por su papel protector en los grupos más vulnerables [14]. Se redactó un informe en julio de 2021 y se estudió la evolución del número de notificaciones en AP, como se muestra en la figura 2.

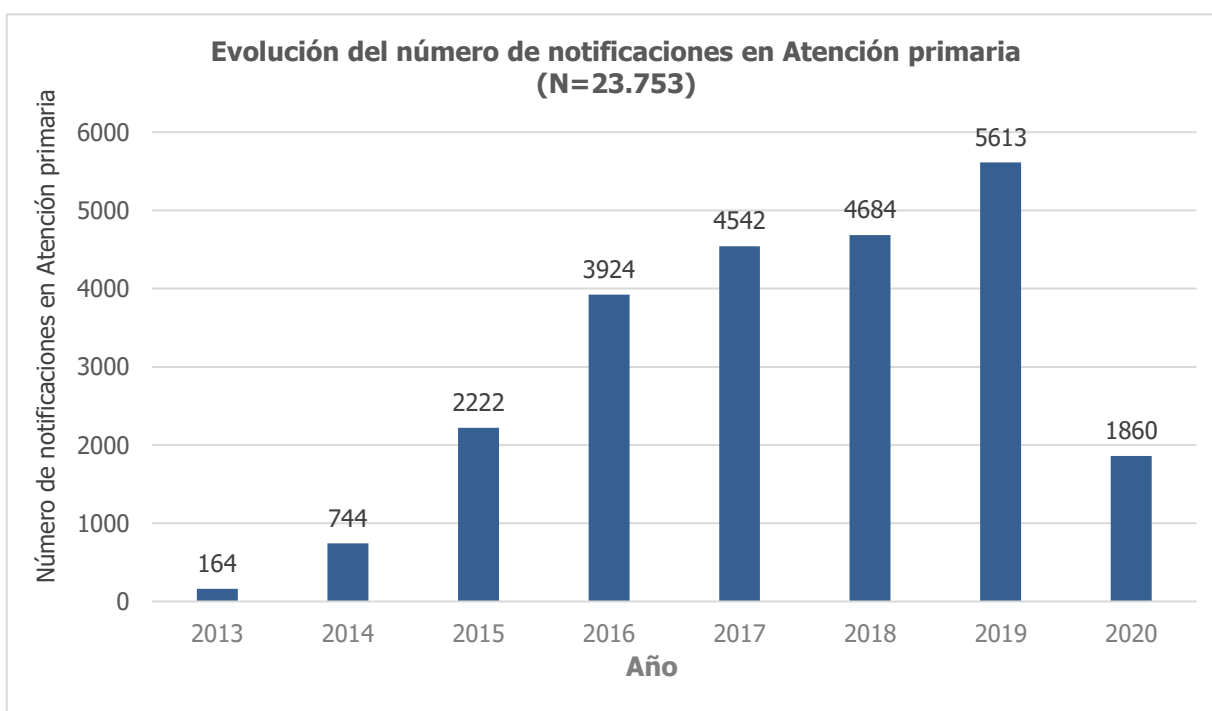


Figura 2. Evolución del número de notificaciones en AP (extraídas el 13 de julio de 2021) [13].

Desde que se implantó el sistema de notificación y gestión de incidentes en el año 2013 hasta el año 2019, se observa un aumento notable de notificaciones de incidentes en AP, pero, el año 2020 se observó un número muy menor de notificaciones. El número total de centros

que reportaron incidentes el año 2020 fue de 211 respecto a los 336 centros del año 2019, y los 273 del 2018 [13].

El año 2021, el Departamento de Salud desarrolló un sistema de incidentes propio y continuista para no perder los datos que fueron notificados desde el año 2013 con el anterior sistema de notificación de una empresa externa. Esto permitió adaptar los formularios de notificación a las necesidades de los diferentes ámbitos sanitarios (hospital, atención primaria, intermedia y salud mental) y a la vez mejorar las funciones de éstos, tanto en la gestión de incidentes, en el seguimiento de acciones de mejora, así como en el análisis de la información [15].

La nueva plataforma de Seguridad de los Pacientes de Cataluña (PSP Cat), se llevó a cabo en el año 2021 gracias a un convenio entre el Departamento de Salud y el Instituto Catalán de la Salud. PSP Cat está formada por el Sistema de Notificación de Incidentes de Seguridad de los Pacientes de Cataluña (SNiSP Cat), la aplicación Proactiva de Seguridad de los Pacientes de Cataluña (PROSP Cat) y el Cuadro de Mando de Seguridad de los Pacientes en Cataluña (QCSP Cat). (Observe la figura 3).

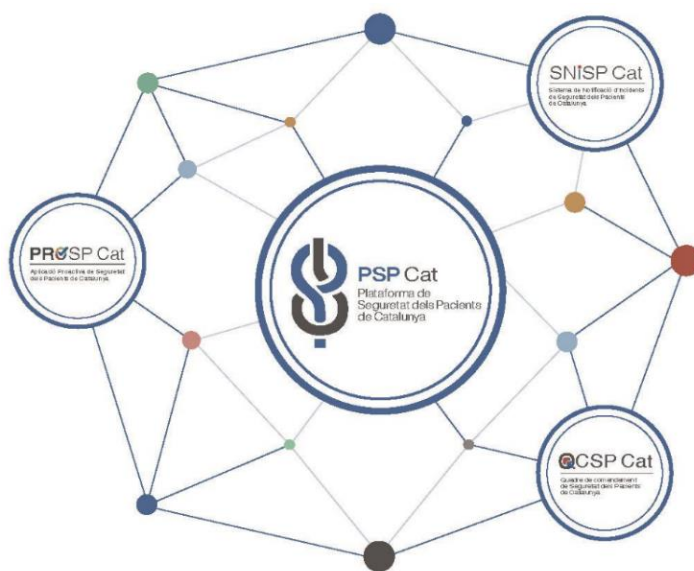


Figura 3. Plataforma de Seguridad de los Pacientes de Cataluña [16].

Esta plataforma aborda la gestión del riesgo de manera integral. Por un lado, adopta una estrategia proactiva que tiene como objetivo prevenir los riesgos mediante la implementación de la PROSP Cat junto a protocolos, procedimientos, manuales de SP y prácticas seguras. Por otro lado, cuenta con una estrategia reactiva a través de la SNiSP Cat que se enfoca en la detección, análisis y aprendizaje de los incidentes. De esta forma, la plataforma gestiona eficazmente el riesgo en todas sus etapas [16].

El objetivo principal que se pretende conseguir con estos sistemas de notificación no es cuantificar el número de incidentes que se producen en los centros sino detectar los puntos débiles del sistema para poder aplicar mejoras [16]. Esta plataforma trata de facilitar la notificación, el análisis, la gestión y el aprendizaje de los incidentes de seguridad de los pacientes que afectan a los pacientes atendidos y propone medidas correctoras para reducir al máximo su repetición. En definitiva, es una estrategia retroactiva que fomenta la cultura de la seguridad de los pacientes en las organizaciones que la usan [15].

3 Análisis de requerimientos

El Departamento de Calidad del Hospital Joan XXIII pedía que la plataforma web siguiese una serie de requisitos, explicados seguidamente en este apartado. Pero, de manera indispensable, se exigía que la aplicación a desarrollar permitiera mostrar los datos de cada una de las regiones sanitarias de manera muy visual y mediante un mapa interactivo. La aplicación debía mostrar los EAP que tenían implantado el SNI SP y los que no, de manera que se pudiese hacer un análisis visual mostrando gráficas y elementos interactivos para poder hacer, con un simple vistazo, una comparación entre cada una de las regiones sanitarias de toda Cataluña.

Los requisitos para la elaboración de la plataforma web eran los siguientes:

- Tener una página con el mapa global interactivo.
- Tener una página para cada región sanitaria donde aparece el mapa interactivo.
- Mostrar en color rojo las regiones del mapa donde no está implantado el sistema de notificación.
- Mostrar en color verde las regiones del mapa donde está implantado el sistema de notificación.
- Tener un menú que te dirija a cada región sanitaria.
- Tener una "navbar" principal corporativa de Gencat.
- Tener una "navbar" secundaria con el logotipo del aplicativo SNI SP Cat.
- Tener un inicio de sesión con usuario y contraseña.
- Tener un botón de cierre de sesión.
- Mostrar una gráfica interactiva con el número de implantados y de no implantados.
- Mostrar los datos totales y los implantados en cada una de las páginas.
- Insertar en cada una de las páginas (de las regiones sanitarias) un listado de los nombres de los equipos por implantar.
- Insertar un modal que, al desplazar el ratón por encima de cada comarca cada región sanitaria, aparezca el nombre del equipo.
- Distinguir en el mapa global cada una de las regiones sanitarias.

4 Metodología

Para crear la plataforma web, que permite la visualización de los equipos implantados en toda Cataluña a partir de los datos obtenidos del Sistema de Notificación de Incidentes de la Seguridad de Cataluña (SNiSP Cat), se utilizó el *framework* Django y, como entorno de desarrollo, Visual Studio Code. Es importante destacar que este proyecto es solo una parte de un proyecto más amplio, del cual se ha tomado inspiración para el diseño y la estética.

Para la creación de este proyecto, cabe remarcar que se realizó el curso de la plataforma Udemy "Python and Django Full Stack Web Developer Bootcamp", ya que no se tenían conocimientos previos en ninguna de las herramientas de este entorno de programación, ni en la creación de aplicaciones webs.

4.1 Entorno de programación

Como se ha mencionado anteriormente, el entorno de desarrollo utilizado ha sido Visual Studio Code, que es una de las herramientas más empleadas por los desarrolladores para crear aplicaciones web. Visual Studio Code proporciona un amplio conjunto de herramientas de programación y de depuración, facilitando así el desarrollo de proyectos similares a la plataforma web creada.

Una página web consta de dos partes principales: el *frontend* y el *backend*. El *frontend* es la parte de una página web que los usuarios ven e interactúan. Por otro lado, *backend* es la tecnología utilizada para procesar y administrar información, administrar bases de datos, autenticar usuarios y ejecutar la lógica comercial de la aplicación.

El *frontend* se basa en tres tecnologías principales: HTML (lenguaje de marcado de hipertexto), CSS (hojas de estilo en cascada) y JavaScript. Estas tecnologías se utilizan para crear la estructura, el estilo y la interactividad de la aplicación web.

Por otro lado, existen múltiples opciones en el *backend* donde se pueden utilizar diferentes tecnologías como Php, Node, Java, Python, etc. El *backend* consta de tres partes principales: lenguaje de programación, marco (*framework*) y base de datos. Para crear la plataforma, se utilizó Python como lenguaje de programación, Django como marco y MySQL como base de datos.

4.1.1 Herramientas de programación

El entorno de desarrollo de la plataforma web se llevó a cabo en un ordenador portátil del Hospital Universitario Joan XXIII de Tarragona con el sistema operativo de Windows 10, ya que es el que me facilitaron para poder realizar el proyecto.

A continuación, haré un breve resumen de las herramientas usadas, diferenciando entre *frontend* y *backend*:

4.1.1.1 *Frontend*

HTML. Se trata del lenguaje por excelencia para crear y estructurar el contenido de una página web. Se basa principalmente en proporcionar etiquetas o elementos que describen la semántica y la estructura de los diferentes componentes que forman una página web. Cada una de estas etiquetas o elementos, pretenden formar un compartimento para definir una función específica y así, poder presentar diferentes tipos de funciones, ya sean párrafos de texto, imágenes, encabezados, tablas, formularios, etc.

Este lenguaje se basa en la estructura de árbol, es decir, cada uno de los elementos está definido dentro de un compartimento, el cual se puede anidar dentro de otros compartimentos,

y estos, definir otros elementos, creando así una estructura jerárquica. Esto permite organizarlos y dar forma al contenido de manera útil y coherente.

Una vez formada y estructurada la plantilla HTML, podemos hacer uso de dos lenguajes fundamentales, que explicaremos a continuación: CSS y JavaScript. Estos, permiten dar estilo e interactividad a la página de manera respectiva.

CSS. Se trata de un lenguaje de programación que se encarga principalmente de describir la apariencia y el formato de un documento HTML. CSS se emplea para aplicar estilos y presentar la parte más visual de una página web.

Su estructura trata de un selector y de un conjunto de propiedades que se aplicarán sobre ese selector. El selector indica a qué determinado elemento de nuestra plantilla HTML se le van a aplicar las propiedades como, por ejemplo, el color de fondo, el tamaño de la fuente, el espaciado, etc.

Además, este lenguaje también permite la aplicación de estilos de forma jerárquica y en cascada. Esto permite mayor flexibilidad y personalización en el diseño de la página web. En la creación de la página web, ha ayudado y aportado el estilo, la estructura y las posiciones de cada uno de los elementos que forman la web, ya sean las barras de navegación, *navbar*, tanto la principal como la secundaria, hasta la posición, color y fuente del botón para el cierre de sesión *logout*. Otras de las utilidades las mostraré en el apartado de resultados.

JavaScript. Se trata de un lenguaje de programación de alto nivel, interpretado y orientado a objetos. Su principal función es agregar interactividad y funcionalidad a la página web. Éste permite interactuar con la plantilla HTML previamente creada y manipular su contenido y apariencia. Sobre todo, en el proyecto se ha centrado en el uso de JavaScript, para responder a eventos como clics, desplazamientos, eventos de *mouse*, estando este último muy presente en la interactividad que se ha usado en la página web. Por ejemplo: cuando el puntero del ratón se mueve sobre un elemento, se dispara el evento de *mouseover*, lo que permite realizar acciones específicas en respuesta. Otro ejemplo común es cuando el puntero del ratón se desplaza sobre un elemento y este hace activar la aparición de un modal.

4.1.1.2 **Backend**

Python. Es un lenguaje de alto nivel, interpretado y de propósito general que destaca por su sintaxis clara y legible. Es un gran lenguaje para aprender, es simple, potente y tiene muchas herramientas que nos pueden ser de gran utilidad a la hora de crear una aplicación web. Python cuenta con una biblioteca estándar extensa y poderosa, que nos proporciona una amplia gama de módulos y funciones que nos facilitan el desarrollo de aplicaciones, por lo que ha sido el lenguaje elegido para la creación de la plataforma.

Django. Django es un *framework* de desarrollo web gratuito y de código abierto (open source) de alto nivel que permite crear aplicaciones web de manera rápida y sencilla. Entre las posibles ventajas que ofrece se encuentran su gran cantidad de funcionalidades integradas, su capacidad para gestionar bases de datos y la eficiencia que proporciona en el manejo de solicitudes HTTP [17].

En cuanto a la productividad, Django proporciona una arquitectura y patrones bien definidos e incluye muchas características integradas, como autenticación de usuarios, manejo de formularios, administración, entre otros, lo que permite acelerar el proceso de desarrollo. Por lo que Django cumple todos los requisitos como *framework* poderoso y eficiente para la creación y el manejo de cualquier aplicación web.

MySQL. Es un sistema de administración de bases de datos relacionales conocido por su eficiencia y velocidad, incluso en entornos con grandes cantidades de datos. Este software

de código abierto destaca por su estabilidad y confiabilidad en la gestión de bases de datos, así como por su amplia compatibilidad con diferentes sistemas operativos.

4.1.1.3 Librerías

En este apartado explicaré algunas de las librerías más importantes que se han usado para la realización del proyecto. Muchas de ellas eran desconocidas para mi conocimiento y por esa razón he tenido que familiarizarme con ellas durante la creación de la plataforma.

No son librerías específicas de Django, sino que son librerías más generales utilizadas en el desarrollo web y en análisis de datos.

JQuery: es una biblioteca de JavaScript de código abierto y multiplataforma que simplifica la interacción y manipulación de documentos HTML, manejo de eventos y animaciones. Esta herramienta ha facilitado la creación de interacciones dinámicas y efectos más visuales de la web.

Bootstrap: es un conjunto de archivos CSS, JavaScript y fuentes que proporciona estilos prediseñados y componentes interactivos para facilitar el desarrollo de interfaces web.

Lo que ha sido de mayor utilidad de la herramienta es el sistema de cuadrícula (*Grid*) de Bootstrap que permite dividir el contenido de las páginas en columnas, dónde se indica cuantas columnas debería tener el diseño según la medida del dispositivo de donde se va a visualizar la página.

Pandas: se trata de una biblioteca de análisis de datos para Python. En mi caso se ha usado para el tratamiento de los datos obtenidos de la base de datos de hospital Mysql.

D3: es una biblioteca de JavaScript para la visualización de datos en la web. D3 ayuda a dar vida a los datos mediante HTML, SVG y CSS. Se ha utilizado para crear y mostrar el mapa visual e interactivo.

Openpyxl: es una biblioteca de Python especializada en el manejo de archivos de Excel (.xlsx). Ha resultado útil para cargar y trabajar con los datos que estaban en formato Excel.

SQLAlchemy: es una biblioteca de Python que facilita la interacción con bases de datos relacionales. Me ha sido útil para conectar mi código con la base de datos del hospital mediante la función "create_engine()" que produce un objeto *engine* basado en una URL.

Highcharts JS: es una biblioteca de JavaScript diseñada para la visualización interactiva de datos. Ha facilitado enormemente la creación de gráficos y visualizaciones en la plataforma, añadiendo un elemento visual más atractivo y fácil de entender.

4.1.2 Framework Django

En este apartado explicaré la estructura y el funcionamiento del *framework* Django que he utilizado para desarrollar la plataforma web.

4.1.2.1 Estructura del Framework

Para comenzar, es recomendable crear un entorno virtual que permita tener una instalación virtual de Python y los paquetes necesarios en el ordenador. Al crear un entorno virtual, podemos probar nuevas funciones sin dañar la aplicación web, ya que contiene la versión actualizada de los paquetes. Es muy recomendable para proyectos, ya que los mantiene autónomos y evita problemas al actualizar los paquetes. Para ello, se siguen una serie de comandos.

En mi caso, se ha utilizado la terminal de Anaconda (Anaconda Prompt) para llevar a cabo todo el proceso de ejecución del proyecto, la cual ya estaba instalada en mi ordenador.

Una vez creado y activado el entorno virtual, se puede crear nuestro proyecto Django utilizando comandos en la terminal (`django-admin startproject SNiSP_Project`). Esto generará una serie de archivos y directorios predeterminados, como se muestra en la figura 4.

Esta parte se ha eliminado por tener contenido confidencial.

Figura 4. Archivos y directorios del proyecto Django "SNiSP_Project". Captura realizada desde Visual Studio Code.

Los archivos que se generan son los siguientes:

- `_init_.py`: un archivo en blanco que indica que este directorio se puede tratar como un paquete de Python.
- `settings.py`: aquí se almacenan todas las configuraciones del proyecto.
- `urls.py`: un script de Python que contiene los patrones de URL para el proyecto, es decir, las diferentes plantillas o páginas web que tendrá.
- `wsgi.py`: otro script de Python que actúa como interfaz de puerta de enlace web del servidor, utilizado para implementar la aplicación web (snisp) en producción.
- `manage.py`: un script de Python que se utiliza para ejecutar comandos relacionados con la construcción de nuestra web. Durante el desarrollo de la web, utilizaremos el siguiente comando en la terminal: `python manage.py runserver`, que nos proporcionará un enlace para ver los cambios en el navegador.

A diferencia de los archivos generados automáticamente, se ha creado de manera manual las siguientes carpetas con el objetivo de mejorar la eficiencia y la optimización:

- Directorio "Templates": aquí se encuentran todas las plantillas HTML de mi página web, especialmente las páginas principales. Más adelante, se creará una aplicación interna en el proyecto para agregar funcionalidades y páginas adicionales en la plataforma, en la cual, también se incluirá otro directorio "templates".
- Directorio "Static": en esta carpeta almaceno los archivos CSS y JavaScript utilizados para mejorar la interacción y estética de las páginas principales de la web. Este directorio también estará presente en la aplicación al igual que los "templates".

Django es una colección de aplicaciones y configuraciones que, cuando se combinan, forman la aplicación web. Por esta razón, se ha creado la aplicación "snisp" utilizando

comandos en la terminal, lo que genera archivos similares a los del proyecto principal, como se muestra en la figura 5.

Esta parte se ha eliminado por tener contenido confidencial.

Figura 5. Archivos y directorios de la aplicación "snisp" del proyecto Django "SNiSP_Project". Captura realizada desde Visual Studio Code.

Estos archivos son:

- `_init_.py`: similar al del proyecto base.
- `admin.py`: permite registrar los modelos en la interfaz de administración de Django.
- `apps.py`: aquí se pueden realizar configuraciones específicas de la aplicación.
- `models.py`: almacena los modelos de datos de la aplicación. En mi caso, no se ha utilizado, ya que los datos provienen directamente de la conexión directa con la base de datos.
- `tests.py`: puede contener funciones de prueba para el código, pero en mi caso no se ha utilizado.
- `views.py`: aquí se encuentran las funciones encargadas de manejar las solicitudes y respuestas de las plantillas HTML de la aplicación.
- `urls.py`: se trata del mismo archivo al de nuestro proyecto principal. Este archivo no venía predeterminado al generar los comandos en la terminal para la creación de la aplicación, sino que lo creé yo misma.

Además, he añadido la carpeta "templates" (como se muestra en la figura 5) y la carpeta "static". Estas carpetas almacenan, respectivamente, los archivos HTML y los recursos estáticos utilizados en la aplicación.

- En la carpeta "templates", se encuentran las páginas HTML correspondientes a cada una de las regiones sanitarias del mapa principal de mi plataforma. Cada región sanitaria tiene su propia página HTML.
- La carpeta "static" se ha subdividido en dos carpetas. La primera, "CSS", donde se han guardado los archivos CSS relacionados con la apariencia de las plantillas HTML de la aplicación "snisp", y, la segunda, "JS", donde se han almacenado los archivos JavaScript utilizados para la interacción en las plantillas. Sin embargo, en mi caso, se ha integrado gran parte de la interacción

directamente en las plantillas HTML utilizando etiquetas `<script>`, ya que resultaba más sencillo.

A parte de todos los archivos y directorios que se han creado de forma predeterminada o de forma manual, existe un directorio llamado "media" (se muestra en la figura 5) que se encarga de almacenar el mapa en un archivo en formato .svg y el archivo Excel dónde encontraremos los códigos 'BR' que nos permitirá relacionar nuestra base de datos con el mapa interactivo.

Además, como uno de los requisitos de la plataforma web era que hubiese una página principal para iniciar sesión (página "login") se ha añadido al proyecto una nueva aplicación llamada "auth_app" siguiendo la misma estructura que para la aplicación principal "snisp", de tal manera que han aparecido los mismos archivos y directorios. (se muestra en la figura 6).

Esta parte se ha eliminado por tener contenido confidencial.

Figura 6. Archivos y directorios de la aplicación "auth_app" del proyecto Django "SNIISP_Project". Captura realizada desde Visual Studio Code.

Para llevar a cabo la página de inicio de sesión el "template" principal es "login.html", todo el proceso se ha realizado mediante el sistema de autenticación y autorización que proporciona Django, simplemente como ejemplo se ha creado un usuario aleatorio para entrar en la plataforma web y la plantilla en HTML en este caso, no ha sido necesario crear una vista para mostrar la información de inicio de sesión.

En un futuro, cuando se implemente la aplicación web definitiva unida con el PROSP se pretende establecer una asociación con otra página web en la que los usuarios ya están registrados y suelen iniciar sesión con habitualidad. Una vez iniciado sesión en esa página web más genérica se generará un token único que se asociará con tu sesión activa. Este, servirá como una forma segura de identificar y autenticar tu sesión cada vez que se interactúe con la plataforma web de los mapas visuales.

Esta integración entre ambas plataformas permite aprovechar la base de usuarios existente.

4.1.2.2 Funcionamiento del *Framework*

Después de describir todos los archivos y directorios que existen en el proyecto, trataré de explicar el funcionamiento de Django de una manera más clara. Es decir, cada una de las conexiones básicas que he tenido que hacer para llegar a construir la plataforma.

Para empezar, es necesario conectar nuestra aplicación "snisp" al proyecto a través del archivo "settings.py". Luego, se asignan las vistas ("views.py") a las URLs ("urls.py"). Para esta tarea, se ha utilizado algunas de las herramientas que ofrece Django, como la función "include()". Esta función permite buscar coincidencias con expresiones regulares y vincularlas al archivo urls.py de nuestra aplicación.

Para vincular la aplicación "auth_app" al proyecto ha sido más sencillo ya que, como he dicho anteriormente, cuenta con un sistema de autenticación y solamente se ha realizado la conexión mediante las URLs (urls.py).

Las vistas (*views*) son responsables de manejar la lógica de interacción con el usuario. En Django, las vistas pueden ser funciones o clases que procesan las solicitudes HTTP y generan respuestas. Además, tienen la capacidad de acceder a los datos de las bases de datos. En mi caso, se ha creado un archivo llamado utils_snisp.py que establece las conexiones necesarias para obtener los datos requeridos. Las vistas también pueden utilizar plantillas para generar respuestas dinámicas. En el apartado de bases de datos, explicaré con más detalle cómo se realizan estas conexiones y obtención de datos.

Las plantillas (*templates*) son elementos fundamentales para comprender cómo Django interactúa y funciona en un sitio web, además de representar la capa de presentación en Django. Una plantilla contiene las partes estáticas de una página HTML, es decir, las secciones que permanecen iguales en todas las visualizaciones. Además, las plantillas contienen etiquetas de plantilla con una sintaxis especial que permite insertar contenido dinámico generado por las vistas de la aplicación Django. Para ello, se han creado directorios de plantillas en el proyecto principal y, posteriormente, un subdirectorio específico para las plantillas de la aplicación "snisp" y la de inicio de sesión.

Es necesario informar a Django sobre la ubicación de las plantillas editando la clave "DIR" dentro del diccionario "TEMPLATES" en el archivo settings.py del proyecto. Una vez realizado este proceso, se han podido crear archivos HTML como base.html, index.html, entre otros.

Para mejorar el aspecto estético y la dinámica del texto, se ha creado el directorio "static", tanto en el proyecto principal como en la aplicación "snisp". Además, se debe agregar la ruta de este directorio en el archivo settings.py.

En el directorio "static" del proyecto, se hace referencia al archivo master.css para la apariencia de los archivos en el directorio "template" del mismo. También se han utilizado los archivos mapa_snisp.css y snisp.css para la apariencia de estos en el directorio "template" de la aplicación.

Adjunto una figura 7 que muestra de manera clara y visual el funcionamiento de Django.

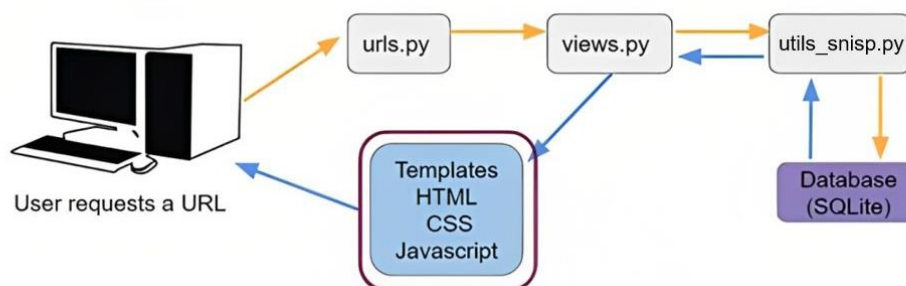


Figura 7. Esquema del funcionamiento de Django. Captura extraída del curso Udemy "Python and Django Full Stack Web Developer Bootcamp".

4.2 Bases de datos

La plataforma web que se está desarrollando en Cataluña busca abordar las necesidades y realidades importantes en el ámbito de la sanidad, tanto a nivel regional como en relación con los avances en seguridad del paciente. Además, con el creciente volumen de datos generados en medicina, es crucial implementar nuevas herramientas para aprovechar al máximo esa información y mejorar nuestro sistema de salud.

Los datos para la plataforma se han obtenido de la base de datos del aplicativo SNI SP Cat. Y, como se menciona anteriormente, este aplicativo se centra en la detección, análisis y aprendizaje de los incidentes. Estos datos no solamente son del Instituto Catalán de la Salud (ICS) sino que también forman parte otros servicios sanitarios.

4.2.1 Servidor Snisp

Para llevar a cabo la plataforma web, se ha usado un conjunto de datos que fueron obtenidos del servidor de pruebas snisp, el cual contiene datos provisionales que simulan el entorno de producción. Eso es debido a que aún se están realizando pruebas y análisis preliminares para tratar de validar la viabilidad de las técnicas utilizadas y ajustar al máximo los parámetros necesarios antes de trabajar con datos reales.

Se trata de una base de datos relacional, es decir, consiste en una base de datos tradicional basada en el modelo relacional. Este tipo de bases de datos utiliza tablas con filas y columnas para organizar y almacenar los datos de manera estructurada. Para interactuar con esta base de datos, se ha empleado el lenguaje SQL (Structured Query Language) y el sistema de gestión de bases de datos MySQL. Con SQL se han podido realizar consultas avanzadas, como la selección de datos específicos y la combinación de información de diferentes tablas y filtrados.

Se ha usado MySQL debido a su amplia robustez y escalabilidad, aparte de ser el sistema que ha proporcionado el hospital.

Se ha establecido conexión entre la base de datos y mi proyecto Django utilizando la biblioteca SQLAlchemy. Gracias a un archivo que se proporcionó llamado "connections.py" (se muestra en la figura 4) que contenía la configuración necesaria, como la función "create_engine()", para establecer una conexión segura a través de la VPN proporcionada por el hospital. Esto ha hecho posible trabajar de una manera más cómoda en un portátil proporcionado por el hospital y acceder a los datos necesarios.

Tabla 1. Ejemplo del dataframe "df_implantats".

Esta parte se ha eliminado por tener contenido confidencial.

Tabla 2. Ejemplo del dataframe "df_merged_implantats". Unión de los dos dataframe mediante el nombre del centro.

Esta parte se ha eliminado por tener contenido confidencial.

En la figura 8 podemos observar el diagrama del modelo E-R (Entidad-Relación), donde se representa la relación entre las dos tablas (entorns/centres) de la base de datos snisp, como se ha detallado anteriormente.

Figura 8. Diagrama del modelo entidad-relación de la base de datos snisp.

Esta parte se ha eliminado por tener contenido confidencial.

5 Resultados

En este apartado mostraremos el resultado del diseño de la plataforma web, teniendo en cuenta los requisitos detallados en el trabajo. Cabe añadir que, como ya se ha mencionado en la introducción del trabajo, la plataforma forma parte de otro proyecto anterior, por lo tanto, se han seguido unas cuestiones predeterminadas de obligada implementación.

A la hora de implementar la plataforma web, se ha priorizado la usabilidad, la simpleza y la accesibilidad. Por esa razón, se ha optado por transformar la información del hospital en gráficos intuitivos que permiten interpretar los datos de forma rápida y sin necesidad de tener instrucciones previas. De esta manera se garantiza que no haya errores de comprensión por parte de los usuarios.

5.1 Apartados de la plataforma web

Favicon: es un pequeño icono que aparece en la barra de pestañas de los navegadores, junto al lado del título de la página. En este caso, se ha insertado el logotipo de la aplicación.

Esta parte se ha eliminado por tener contenido confidencial.

Figura 9. Captura de pantalla del favicon de la web.

Este pequeño icono se ha puesto a partir del *template* base.html, que tiene la función principal de estructurar la base de toda la web. Se ha realizado un `<link>` con el atributo *rel/icon*, que permite colocar el *favicon* correctamente.

Navbar principal: un *navbar* o barra de navegación es una sección de una interfaz gráfica destinada a ayudar a los visitantes a acceder a la información. En este caso, se ha optado por dedicar el "navbar principal" a un uso visual y corporativo.

Esta parte se ha eliminado por tener contenido confidencial.

Figura 10. Captura de pantalla de la navbar principal con el logotipo corporativo de Gencat.

La navbar principal se ha incorporado a la base.html, al igual que el favicon y la navbar secundaria. También se ha creado un archivo CSS, "master.css", en la carpeta "static", que ha sido adjuntada a la plantilla base. De esta manera se ha generado la estética de la navbar.

Navbar secundaria: esta navbar hace la misma función que la "navbar principal". En ella, se encuentra información corporativa de la aplicación y, además, el botón de cierre de sesión, cuando el usuario ya está autenticado.

Esta parte se ha eliminado por tener contenido confidencial.

Figura 11. Captura de pantalla de la navbar secundaria con el logotipo de la aplicación.

Para crear la *navbar* secundaria se ha seguido el mismo procedimiento que en la *navbar* principal. Se ha incorporado a la *base.html*, al igual que el *favicon*, y también se ha usado el mismo archivo CSS, "*master.css*", que en el *navbar* principal.

Login: se ha realizado un *login*, ya que uno de los requisitos indispensables para la creación de la plataforma.

Esta parte se ha eliminado por tener contenido confidencial.

Figura 12. Captura de pantalla del login para acceder a la plataforma.

Para conseguir crear la plantilla de inicio de sesión he tenido en cuenta lo que se ha mencionado en el apartado de metodología que, básicamente, se resume en la creación de una aplicación aparte llamada "*auth_app*", donde se ha creado un directorio "*templates*" con el archivo "*login.html*", que contiene la estructura propia de nuestro inicio de sesión.

Para la creación y autenticación de los usuarios se ha usado el sistema que viene ya incorporado en Django para la autenticación de usuarios, por lo que no ha sido necesario crear una "*view*" para llevarlo a cabo. Simplemente, se ha vinculado la aplicación "*auth_app*" a nuestro proyecto mediante el archivo "*settings.py*" y se ha enlazado la plantilla con los urls de la aplicación, archivo "*urls.py*".

Para que ningún elemento de la plataforma se pudiese visualizar sin estar autenticado, se ha usado el decorador "*login_required*", que permite restringir el acceso a las vistas creadas para la plataforma. Este decorador verifica si el usuario ha iniciado sesión y, de lo contrario, redirige al usuario a la página de inicio de sesión.

De la misma manera, se ha usado el condicional "*if user.is_authenticated*" en las plantillas como "*base.html*", siguiendo el mismo procedimiento que el decorador explicado anteriormente.

Página principal: una vez iniciado sesión en la plataforma, entraremos en la página principal de la web, donde ya podremos observar datos y el mapa interactivo que nos dará una serie de información.

Esta parte se ha eliminado por tener contenido confidencial.

Figura 13. Captura de pantalla de la página principal de la web.

- **Menú lateral:** este menú situado a la izquierda de la pantalla llamado "Implantació Equips", se encarga de dirigir (a las páginas de las regiones sanitarias respectivas: "Tarragona", "Barcelona", "Terres de l'Ebre", "Alt Pirineu i Aran", "Lleida", "Catalunya Central" y "Girona") y redirigir (a cualquiera de las páginas de la aplicación). Este menú es similar en todos los *templates* de la web por lo que está ubicado en la plantilla base de nuestro proyecto "base.html" (contiene el condicional "if user.is_authenticated" de tal manera que solo aparecerá si el usuario esta autenticado). La plantilla base está conectada a todas las subplantillas de la plataforma, tanto la estructura "{% extends 'base.html' %}" como la estética "{% load static %}".
- **Mapa Visual:** el mapa, situado en el centro de nuestra plantilla, nos da información sobre los equipos de AP que han implantado el sistema de notificación y gestión de incidentes (en color verde) y los equipos que no lo han hecho (en color rojo). El mapa se ha introducido como un archivo formato .svg, el cual nos proporcionaron desde el hospital. Asimismo, el mapa completo se ha cargado en la plantilla "index.html" y, cada una de las regiones sanitarias, en cada una de las plantillas correspondientes. Todo ello se ha programado mediante el lenguaje de JavaScript. Para pintar el mapa se han usado los datos pertenecientes a la base de datos nombrada y explicada en el apartado de metodología (bases de datos). El DataFrame respectivo se ha creado dentro de una función, en el archivo llamado "utils_snisp.py", definida como "equips_totCat()", la cual devuelve el DataFrame con los datos tratados de todos los equipos de AP implantados en todo Cataluña con su respectivo código "BR" al que igualamos a los códigos "BR" del mapa. De esta manera, para pintar el mapa se ha usado el método de JavaScript "forEach(item)" que hace iteración sobre cada elemento "BR" y así, todos los "BR" que coincidan con los del mapa se les aplica la operación de pintar con color verde su zona. (Eso significará que están implantados). Para poder visualizar y usar estos datos en el respectivo mapa se ha creado, en el archivo "views.py" del proyecto "SNiSP_Project", una vista llamada "IndexView", que renderiza la platilla "index.html" y agrega toda la información necesaria que observaremos en el mapa y en la página principal al contexto de la plantilla "index.html".
- **Gráfica y elemento contenedor:** en la parte derecha de la plantilla se puede observar una gráfica con los porcentajes (de cada una de las regiones sanitarias) respectivos a la implantación del sistema SNiSP en AP. En la parte superior de la gráfica se observa un elemento contenedor que simplemente hace el recuento de los equipos totales y de los equipos implantados. La gráfica se ha realizado mediante la librería de JavaScript nombrada anteriormente "Highcharts". Con ella se ha realizado un gráfico llamado "Implantació Equips" que,

aparte de mostrar a simple vista los porcentajes de cada una de las regiones, es interactivo, de tal manera que al posicionar el ratón por encima del gráfico podemos observar solamente el porcentaje de los implantados o de los que no. También, si hacemos "click" encima de los elementos de la leyenda "Per Implantar" o "Implantats" podemos visualizar las cifras correspondientes en solitario. (Se muestra en la figura 14)

Esta parte se ha eliminado por tener contenido confidencial.

Figura 14. Captura de pantalla de la página principal de la web. Solamente a la gráfica interactiva "Implatació EAP". Seleccionando los implantados, situando el ratón encima de la región sanitaria de Barcelona.

- **Botón cierre de sesión:** el botón de cierre de sesión se ha situado en la barra de navegación secundaria, por lo que se ha implementado en la plantilla "base.html". Este botón "Sortir" se ha hecho con la estética corporativa respectiva de Gentcat, en el archivo "master.css".
El botón está programado para que solamente aparezca cuando el usuario ha iniciado sesión, al igual que el menú lateral explicado anteriormente.

Otras de las partes interactivas de la página principal son:

- Cambia de color la región sanitaria al desplazar el ratón por encima de ellas, esto se ha hecho mediante la librería d3. (Se muestra en la figura 15).
- Al hacer "click" encima de las regiones sanitarias del mapa te redirige a la región sanitaria respectiva.

Esta parte se ha eliminado por tener contenido confidencial.

Figura 15. Captura de pantalla de la página principal de la web. En el mapa visual donde se ha seleccionado la región sanitaria de "Lleida".

Región Sanitaria de Tarragona: una vez hagamos “click” en el menú lateral de la página principal o encima de la región sanitaria de Tarragona nos redirigirá a la página respectiva de la región sanitaria de Tarragona (Se muestra en la figura 16).

Esta parte se ha eliminado por tener contenido confidencial.

Figura 16. Captura de pantalla de la página de la región sanitaria de Tarragona.

Como se puede observar en la figura 16, las barras de navegación (principal y secundaria), el menú lateral “Implantació Equips” y el botón de cierre de sesión “Sortir” mantienen su estructura, función y estética.

- **Mapa Visual “Tarragona”:** el mapa, situado en el centro de nuestra plantilla, nos da información sobre los equipos de AP que han implantado el sistema de notificación y gestión de incidentes (en color verde) y los equipos que no lo han hecho (en color rojo), al igual que el mapa visual global.

El mapa de la región sanitaria se ha introducido de la misma manera que el mapa visual global, pero en una plantilla distinta, dentro de la aplicación “snisp”, llamada “RS_Tarragona.html”. De la misma manera, todo se ha programado mediante el lenguaje de JavaScript.

En este caso, para pintar el mapa, se han usado los mismos datos que para pintar el mapa global, pero acotando el rango usando otra función específica (también creada en el archivo “utils_snisp.py”) definida como “equips_regio(regio_sanitaria)”. Esta función devuelve el dataframe con los datos tratados de todos los equipos de AP implantados en la región que indiquemos, en este caso, Tarragona. Esto ha sido posible gracias a que dentro de la función “equips_regio(regio_sanitaria)” se ha usado la misma “query” del mapa global (situada en la función “equips_totCat()”) pero, en este caso añadiendo una nueva condición que limita los datos según la región sanitaria que se indique. Eso irá en función de la región sanitaria a la que nos dirijamos.

La operación de pintar el mapa se ha realizado de la misma manera que para el mapa global, usando el método de JavaScript “forEach(item)”, iterando sobre cada elemento “BR” de la región sanitaria de Tarragona.

De la misma manera, para poder visualizar los datos en el respectivo mapa de la región se ha creado, en el archivo views.py (en este caso dentro de la aplicación “snisp”), una vista llamada “RSTarragonaView”, que permite agregar y enviar toda la información necesaria que observaremos en el mapa de Tarragona y en la página de la región sanitaria al contexto de la plantilla “RS_Tarragona.html”.

- **Gráfica y elemento contenedor:** En la parte derecha de la plantilla (se puede observar en la figura 16) tenemos el mismo elemento contenedor visualizado en la plantilla global pero, en este caso, solo referente a la región sanitaria de Tarragona. Asimismo, podemos observar de manera más clara el recuento de este en la figura 17 (32 EAP totales en Tarragona de los cuales 15 han sido implantados). En la parte inferior del contenedor (figura 17), se observa una gráfica muy similar a la de la plantilla global, pero en este caso haciendo referencia a la región sanitaria de

Tarragona. La gráfica se ha realizado de manera similar, pero al contrario de la plantilla principal, este se ha realizado solamente con el número exacto, no con el porcentaje.

Esta parte se ha eliminado por tener contenido confidencial.

Figura 17. Captura de pantalla de la página de la región sanitaria de Tarragona. (Zum de la parte derecha).

- **Lista de EAP para implantar:** una de las modificaciones existentes respecto a la plantilla inicial es que se muestra una lista de los EAP que no han sido implantados en la región sanitaria (Por ejemplo: en el "EAP SANT PERE" no han implantado el sistema SNiSP Cat). (Se observa en la figura 17).

Para realizar esta tarea se ha creado un diccionario (en el archivo de "views.py" en la aplicación "snisp") llamado "dict_equips" a partir de un DataFrame (generado en la función "equips_totCat_NomMapa()" del archivo "utils_snisp.py" de la aplicación "snisp").

Seguidamente, se ha convertido ese diccionario en una cadena "JSON" para pasarlo al contexto de la plantilla (en este caso, "RS_Tarragona.html").

Una vez dentro de la plantilla "RS_Tarragona.html" se ha usado la función "JSON.parse()" para convertir una cadena de texto (del diccionario "dict_equips") "JSON" en un objeto JavaScript. De esta manera se han podido mostrar los nombres de cada uno de los EAP.

Las partes interactivas de la página de la región sanitaria son:

- Se ha incorporado un elemento modal para mostrar al pasar el ratón por encima del mapa de la región sanitaria (en este caso de Tarragona) aparezca un modal con el nombre del EAP en el cual estamos encima. (Se puede observar en la figura 18).

Esta parte interactiva se ha realizado siguiendo el mismo mecanismo que para generar la lista de los EAP de la figura 17. (Usando el diccionario creado "dict_equips").

Esta parte se ha eliminado por tener contenido confidencial.

Figura 18. Captura de pantalla de la página de la región sanitaria de Tarragona. En el mapa visual donde se ha pasado por encima del EAP de Falset.

- En la gráfica que se muestra el recuento de los EAP se ha realizado el mismo sistema de interactividad que en la gráfica de la plantilla principal. Al posicionar el ratón por

encima se puede visualizar tanto el porcentaje como el número de EAP. (Se puede observar en la figura 19). También, se ha añadido la función interactiva de poder hacer "click" encima de la leyenda "Per Implantar" o "Implantats para poder visualizar las cifras correspondientes en solitario.

Observamos en la figura 19 un número de 15 EAP implantados en Tarragona, con un porcentaje del 46.9%.

Esta parte se ha eliminado por tener contenido confidencial.

Figura 19. Captura de pantalla de la página de la región sanitaria de Tarragona. Solamente a la gráfica interactiva. Situando el ratón encima de la parte de los EAP implantados.

Cabe destacar que para hacer las páginas referentes a cada una de las regiones sanitarias ("Barcelona", "Lleida", "Girona", "Catalunya central", "Alt Pirineu i Aran", "Terre de l'Ebre") se ha realizado el mismo proceso que para hacer la región sanitaria de Tarragona. En cada una de las plantillas .html se ha puesto la región del mapa respectiva y se ha pintado de la misma manera que para el mapa global y para la región de Tarragona.

Esta parte se ha eliminado por tener contenido confidencial.

6 Limitaciones

En este apartado haré énfasis en las limitaciones que se han tenido a lo largo del desarrollo de la plataforma web.

En primer lugar, una de las restricciones que tuve durante la ejecución de este proyecto fue mi falta de experiencia y conocimiento previo con el lenguaje de programación Django y el desarrollo de plataformas web, ya que durante la carrera no había tenido la oportunidad de trabajar en este tipo de proyectos, lo que implicó un desafío adicional.

Sin embargo, para superar esta limitación realicé un curso donde pude adquirir las habilidades y conocimientos que me han ayudado a tener una base para seguir aprendiendo en este entorno. Aunque este aprendizaje supuso un tiempo adicional en la planificación y desarrollo de la web, considero que realizar un proyecto de esta magnitud y de esta especialidad es también una oportunidad, ya que podría trabajar perfectamente en este ámbito en un futuro. Es por lo que, aunque se ha superado con creces mi objetivo, es posible que algunas características o funcionalidades más avanzadas no estén completamente implementadas debido a mi nivel de experiencia.

En segundo lugar, tuve que enfrentarme a otra limitación significativa como es la calidad de los datos disponibles. Es más, a lo largo de la elaboración del proyecto encontré que muchos de los nombres de EAP no coincidían exactamente entre los datos de la base de datos y el archivo Excel que me facilitaron para poder relacionar los nombres con cada uno de los códigos "BR", los cuales eran necesarios para representarlos y pintar el mapa. Por lo que muchos de los EAP que ahora mismo salen como no implantados, probablemente, en la futura normalización de los datos sean EAP que sí han implantado el SNI SP. Esta falta de coherencia en los nombres dificultó el proceso de vinculación y generó inexactitudes en los datos que se muestran en la plataforma.

En tercer lugar, se añade otra limitación que ralentizó el proceso, la burocracia. En este sentido, antes de poder iniciar el proyecto, necesité pedir permisos al Hospital Joan XXIII para poder acceder a las bases de datos, así como para obtener un ordenador portátil para implementar la web, ya que no se podía hacer con el mío personal por la protección de datos de los usuarios del centro sanitario.

7 Futuras mejoras

En cuanto a las futuras mejoras, o bien, a lo que se podría continuar haciendo si el proyecto durase más tiempo podría detallar los siguientes aspectos:

En primer lugar, hacer una normalización exhaustiva de los datos de todos los hospitales y centros de EAP de Catalunya con el objetivo de tener una mayor realidad de la plataforma web y que realmente aporte un enfoque de cambio para la seguridad del paciente.

En segundo lugar, se podrían mostrar los datos respecto al tipo de incidentes (que están predeterminadas en el sistema SNI SP Cat), como por ejemplo: infecciones, medicación, gestión de riesgos, miscelánea, higiene de manos, identificación de pacientes, procedimientos y caídas. Estos datos se podrían mostrar de manera más visual con formato de gráficas con porcentajes de, por ejemplo, tipo de incidente que se produce de manera más reiterada, para así poder controlar con mayor exactitud donde se debe aplicar la mejora.

En tercer lugar, aplicar esta plataforma para todos los departamentos sanitarios como Hospitales de Agudos (HA), Centros de Atención Intermedia (CAI), Servicio de Emergencias Médicas (SEM)...

Por último, se podría mejorar la plataforma añadiendo un filtro según el proveedor que al usuario le interesase mirar, por ejemplo, entre centros del ICS (Instituto Catalán de la Salud), Xarxa Santa Tecla o centros privados...

8 Conclusión

El objetivo principal del proyecto fue realizar una plataforma web para mostrar los datos de la aplicación SNiSP Cat. Este objetivo se pudo llevar a cabo gracias a bases de datos del servidor "snisp" y al *framework* Django. Así pues, hemos cumplido el objetivo principal, ya que es posible visualizar de manera sencilla los EAP que han implantado el SNiSP y los que no. De esta manera, esta plataforma, en un futuro no muy lejano, podrá ser muy útil para determinar con claridad todos los EAP de Cataluña que han implantado este sistema y, por lo tanto, que ofrecen una mayor seguridad del paciente delante de los centros sanitarios donde no lo han hecho. También, con las futuras mejoras que se pueden llegar a implementar, se podrá llevar un control más concreto y exhaustivo de los incidentes que se producen de manera reiterada, para poder frenarlos.

También se han conseguido otros objetivos secundarios relacionados con el aprendizaje de estas nuevas herramientas de programación y entornos web. Asimismo, he asimilado conceptos nuevos, como es, comprender y distinguir los términos *backend* y *frontend*, además de aprender a diseñarlos.

Por último, pero no menos importante, se ha reconocido la importancia de notificar y gestionar adecuadamente los incidentes para garantizar una mayor seguridad del paciente. En esta línea, he determinado que es crucial promover y formar a los profesionales sanitarios en relación con la cultura organizacional, superando así el temor al castigo y manteniendo la confidencialidad de la información. Debido a esto, se conseguirá una mayor cantidad de datos que, con la ayuda de aplicaciones o plataformas web como la creada en este proyecto, Mapa Visual del SNiSP Cat, aportaran un control más fiable de la información recogida y, por tanto, una mayor seguridad en el sistema sanitario actual.

9 Referencias

- [1] I. del director General, "Medidas mundiales en materia de seguridad del paciente." 2019. [Online]. Available: <https://www.who.int/es/news-room/fact-sheets/detail/patient-safety>
- [2] Carlos Aibar Remón, Jesús M. Aranz Andrés, Julian Vitaller Burillo, Yolanda Agra Varela, Teresa Giménez Júlvez, and Pilar Marcos Calvo, "Seguridad del paciente en atención primaria," 2003.
- [3] M. E. Stefl, "To Err is Human: Building a Safer Health System in 1999.," *Front. Health Serv. Manage.*, vol. 18, no. 1, pp. 1–2, 2001, doi: 10.1097/01974520-200107000-00001.
- [4] L. L. Leape, "Patient Safety in the Era of Healthcare Reform," *Clin. Orthop. Relat. Res.*, vol. 473, no. 5, pp. 1568–1573, May 2015, doi: 10.1007/s11999-014-3598-6.
- [5] OMS, "Seguridad del paciente," 2019. <https://www.who.int/es/news-room/fact-sheets/detail/patient-safety>
- [6] M. De Sanidad and Y. Consumo, "Sistemas de registro y notificación de incidentes y eventos adversos Calidad y Evaluación Sanitaria Calidad y Evaluación Sanitaria Planificación Sanitaria Planificación Sanitaria SECRETARÍA GENERAL DE SANIDAD AGENCIA DE CALIDAD DEL SISTEMA NACIONAL DE SAL."
- [7] C. Vincent, "Understanding and Responding to Adverse Events," *N. Engl. J. Med.*, vol. 348, no. 11, pp. 1051–1056, 2003, doi: 10.1056/nejmhpr020760.
- [8] "Systems Approach," *September 7*, 2019. <https://psnet.ahrq.gov/primer/systems-approach>
- [9] "Checklists," *September 7*, 2019. <https://psnet.ahrq.gov/primer/checklists>
- [10] B. Castro Matesanz, "TRABAJO DE FIN DE GRADO ENFERMERÍA Sistema de Notificación y Aprendizaje para la Seguridad del Paciente (SINASP): Situación actual en Atención Primaria de Ferrol," 2016.
- [11] M. Massanés *et al.*, "Roda contínua de formació i consens. SeguRetat del pacient Notificació i gestió d'incidents relacionats amb la seguretat dels pacients com a eina per a la millora contínua," 2017.
- [12] J. Brea Aymericha, J. Bladé Creixentib, N. Martín Vergarac, M.L. Forés Palaciosc, E. Granado Fontc, and A. TorresCreixenti, "TPSC-CLOUD, UNA HERRAMIENTA PARA LA MEJORA DE LA SEGURIDAD DE LOS PACIENTES," 2016.
- [13] D. de Salut, "Notificació d'incidents en la seguretat del pacient a l'atenció primària a Catalunya," 2021. [Online]. Available: <https://seguretatdelspacients.gencat.cat/web/.content/minisite/seguretatpacients/professionals/projectes/sist-notificacio/Notificacio-incidentes-seguretat-pacient-AP-2020.pdf>
- [14] M. P. Astier Peña, M. L. Torijano Casalengua, R. Añel Rodríguez, J. Palacio Lapuente, and C. Aibar Remón, "The past, present and future of patient safety in primary care," *Atencion Primaria*, vol. 53. Elsevier Doyma, Dec. 01, 2021. doi: 10.1016/j.aprim.2021.102221.
- [15] "Formació sobre les noves funcionalitats incorporades al Sistema de Notificació d'Incidents de Seguretat dels Pacients (SNI SP Cat) del Departament de Salut," *28 Abril*, 2023. <https://seguretatdelspacients.gencat.cat/ca/detalls/noticia/Formacio-sobre-les-noves-funcionalitats-incorporades-al-Sistema-de-Notificacio-dIncidents-de-Seguretat-dels-Pacients-SNI-SP-Cat-del-Departament-de-Salut>
- [16] D. de S. Generalitat de Catalunya, "Formació Gestors Sistema de Notificació d'Incidents de Seguretat dels Pacients (SNI SP Cat) març de 2023," 2023.
- [17] "Django tutorial." <https://tutorial.djangogirls.org/es/>

Anexo 1. *Template* de cada región sanitaria.

Imágenes de cada uno de los *templates* respectivos a cada una de las regiones sanitarias.

Esta parte se ha eliminado por tener contenido confidencial.

Figura 20. Captura de pantalla de la página de la región sanitaria de Barcelona.

Esta parte se ha eliminado por tener contenido confidencial.

Figura 21. Captura de pantalla de la página de la región sanitaria de "Terres de l'Ebre".

Esta parte se ha eliminado por tener contenido confidencial.

Figura 22. Captura de pantalla de la página de la región sanitaria de "Alt Pirineu i Aran".

Esta parte se ha eliminado por tener contenido confidencial.

Figura 23. Captura de pantalla de la página de la región sanitaria de "Lleida".

Esta parte se ha eliminado por tener contenido confidencial.

Figura 24. Captura de pantalla de la página de la región sanitaria de "Catalunya Central".

Esta parte se ha eliminado por tener contenido confidencial.

Figura 25. Captura de pantalla de la página de la región sanitaria de "Girona".

Anexo 2. Código de programación

Esta parte se ha eliminado por tener contenido confidencial.