

GESTURN

Sistema de consulta remota de turnos

Memoria

19 de enero de 2023

Trabajo de Final de Grado en Ingeniería Informática

Itinerario en Tecnologías de la información

Área de Sistemas de Gestión del conocimiento

Miguel García González

Dirección académica: Javier Martí Pintanel

Responsable académico: Atanasi Daradoumis Haralabus



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 consulta remota de turnos</i>
Nombre del autor:	<i>Miguel García González</i>
Nombre del consultor/a:	<i>Javier Martí Pintanel</i>
Nombre del PRA:	<i>Atanasi Daradoumis Haralabus</i>
Fecha de entrega:	Enero de 2023
Titulación:	<i>Grado en Ingeniería Informática</i>
Área del Trabajo Final:	<i>Área de Sistemas de Gestión del Conocimiento</i>
Idioma del trabajo:	<i>Castellano</i>
Palabras clave	<i>Gestión del Conocimiento, Página web, Single Page Application (SPA), Blazor</i>
Resumen del Trabajo	
<p>Con el avance de las nuevas tecnologías, podemos reinventar o evolucionar soluciones ya existentes para poder ser más eficientes y además contribuir con el medio ambiente.</p> <p>Esta aplicación tiene como objetivo la realización de un prototipo funcional que mejore la gestión del conocimiento, el cual permita a las personas obtener un turno rápidamente en un establecimiento y poder realizar el seguimiento de este, de forma no presencial.</p> <p>Esta solución puede substituir los dispositivos comúnmente llamados “avisadores”, provocando un impacto positivo en el medio ambiente.</p>	

Work Sheet

Work title:	<i>Remote shift enquiry system</i>
Author's name:	<i>Miguel García González</i>
Name of consultant:	<i>Javier Martí Pintanel</i>
Name of the PRA:	<i>Atanasi Daradoumis Haralabus</i>
Delivery date:	January 2023
Degree:	<i>Degree in Computer Engineering</i>
Final Work Area:	<i>Knowledge Management Systems Area</i>
Language of the work:	Spanish
Key words	<i>Knowledge Management, Website, Single Page Application (SPA), Blazor</i>
Summary of the work:	
<p>With the advance of new technologies, we can reinvent or evolve existing solutions in order to be more efficient and also contribute to the environment.</p> <p>The objective of this application is to create a functional prototype that improves knowledge management, allowing people to obtain a shift quickly in an establishment and to be able to follow it up in a non-presential way.</p> <p>This solution can replace the devices commonly known as "pager paging device", having a positive impact on the environment.</p>	

Dedicatoria

El resultado de este trabajo está dedicado a toda mi familia. Principalmente, a mis padres que me han apoyado durante toda mi vida en los momentos más difíciles, pero también en los buenos. Gracias por todos los valores y principios que me habéis inculcado.

También quiero dedicarlo a mi pareja Sali. Por la paciencia que ha tenido conmigo y el tiempo que no he podido disfrutar a su lado por estar ausente durante este tiempo.

Agradecimientos

Muchas gracias a todo el personal y organización de la Universitat Oberta de Catalunya por permitirme realizar este grado, sin este modelo educativo no hubiese tenido la oportunidad de compaginar el trabajo con los estudios y poder finalizar el grado.

Especialmente, mi más sincero agradecimiento al tutor de este trabajo de final de grado, Javier Martí Pintanel. Gracias a sus sabios consejos y observaciones ha contribuido enormemente en la calidad del proyecto, haciendo que me sienta acompañado durante todo el proceso.

Índice

1. Introducción	1
1.1 Contexto del problema a resolver	1
1.2 Exposición del problema	1
1.3 Propuesta de solución.....	1
1.4 Alcance	2
1.5 Objetivos	2
1.6 Impacto en sostenibilidad, aspectos ético-sociales y de diversidad	3
1.7 Enfoque y método	3
1.8 Motivación personal	3
1.9 Conocimiento previo.....	4
2. Planificación	5
2.1 Introducción	5
2.2 Relación de hitos.....	5
2.3 Grupos de tareas.....	6
2.4 Temporalización/esfuerzo.....	6
2.5 Valoración económica.....	9
2.6 Análisis de riesgos.....	10
3. Análisis	12
3.1 Situación actual.....	12
3.2 Análisis del dominio	15
3.2.1 Roles.....	15
3.3 Análisis funcional	15
3.3.1 Análisis de requisitos.....	15
3.3.1.1 Requisitos funcionales.....	15
3.3.1.2 Requisitos no funcionales.....	16
3.4 Stakeholders	17
3.5 Casos de uso	17
3.6 Definición de encuestas de satisfacción	27
3.7 Estudio de las tecnologías web actuales	27
4. Diseño	29
4.1 Diseño interfaz.....	29
4.2 Diseño de base de datos	36
4.2.1 Diseño de datos conceptual	36
4.2.2 Diseño de datos lógico	37

4.2.3	Diseño de datos físico	38
5.	Implementación del prototipo	39
5.1	Entorno de desarrollo	39
5.2	Dependencias	39
5.3	Base de datos.....	40
5.4	Estructura del proyecto.....	41
5.5	Descripción de la interfaz (páginas)	43
5.5.1	Pantalla de carga.....	43
5.5.2	Pantalla de identificación (login).....	45
5.5.3	Menú Registro / Entrar.....	45
5.5.4	Pantalla de registro	46
5.5.5	Pantalla inicial.....	47
5.5.6	Menús del aplicativo	48
5.5.7	Pantalla “Nueva sala”	49
5.5.8	Pantalla “Mis salas”	49
5.5.9	Pantalla visualización de turnos en sala	51
5.5.10	Pantalla visualización código QR sala	52
5.5.11	Pantallas de encuestas	53
5.5.12	Pantallas de gestión de la cuenta.....	54
5.6	Pruebas	56
5.6.1	Conclusiones.....	57
5.7	Proyectos derivados.....	57
5.8	Observaciones	58
6.	Implantación del prototipo	59
6.1	Creación de la infraestructura	59
6.1.1	Creación de la máquina virtual	59
6.1.2	Instalación del sistema operativo	61
6.1.3	Instalación de los requisitos	68
6.1.3.1	Instalación de la base de datos	68
6.1.3.1.1	Configuración de SQL Server Express	72
6.1.3.1.2	Instalación del Internet Information Services (IIS)	73
6.1.3.1.3	Instalación de ASP CORE RUNTIME	74
6.2	Configuración de la aplicación	75
6.2.1	Creación del certificado auto-firmado	77
6.3	Instalación/configuración de la plataforma web en IIS.....	77
6.4	Observaciones adicionales	80
7.	Conclusiones	82
8.	Glosario.....	83

9. Bibliografía	84
10. Anexo I: Informe ejecutivo	85
11. Anexo II: Informe de seguimiento	88
12. Anexo III: Segundo Informe de seguimiento	92

1. Introducción

1.1 Contexto del problema a resolver

En innumerables ocasiones de la vida cotidiana nos encontramos que debemos esperarnos presencialmente a ser atendidos, ya sea en entidades públicas o privadas. Un ejemplo puede ser la cola para esperar mesa en cualquier restaurante concurrido.

Esto implica estar físicamente en el lugar durante toda la espera previa a nuestro turno, obligándonos a no poder realizar cualquier otra tarea.

En la sociedad actual las personas buscan poder realizar varias tareas simultáneamente siempre que sea posible. En el transporte público se puede aprovechar para revisar el correo electrónico, escuchar música, revisar las redes sociales o simplemente jugar con tu teléfono inteligente. Gracias a la continua evolución de la ciencia y la tecnología esto es factible ya que muchas de estas tareas podemos realizarlas con un smartphone.

De igual manera, la pandemia que hemos vivido recientemente ha marcado un antes y un después sobre todo en las zonas públicas en las que se suelen congregarse bastantes personas, controlando más estrictamente la cantidad de personas que pueden estar dentro de los establecimientos o incluso limitándose a un número mucho menor.

1.2 Exposición del problema

En la mayoría de los lugares en los que se debe realizar cola para poder ser atendido ya implementa algún sistema de turnos, normalmente un sistema de tiquetes impresos con tu turno asignado. El problema es que debemos estar físicamente presentes para poder ver los avances en los turnos.

Otra alternativa existente al tradicional sistema de tiquetes impresos es la utilización de unos dispositivos físicos bastante costosos llamados coloquialmente “avisadores”.

Estos dispositivos disponen únicamente de la función de advertir a su portador que su turno es el siguiente para que se aproxime de nuevo al establecimiento. La función de estos dispositivos es muy limitada, deben ser recargados con frecuencia, disponen de un corto alcance y además pueden extraviarse.

1.3 Propuesta de solución

Con el avance de las nuevas tecnologías y con los *smartphones* presentes en nuestro día a día podemos presentar una evolución a este sistema de turnos, permitiéndonos ver en tiempo real y desde cualquier parte el estado de nuestro turno.

La propuesta de solución es la realización de un prototipo funcional de sistema informático que permita mejorar la gestión del conocimiento, el cual permita a las personas obtener un turno rápidamente en un establecimiento y poder realizar el seguimiento de este de forma no presencial. Los usuarios podrán consultar en todo

momento cuantos turnos hay por delante y una aproximación del tiempo restante hasta que su turno.

Este sistema tiene como objetivo principal que los usuarios puedan aprovechar el tiempo limitado del que disponen de la mejor manera posible.

1.4 Alcance

El alcance se centra en el desarrollo de un prototipo funcional de sistema informático que permita la mejora en la gestión del conocimiento de los turnos en cada negocio o establecimiento, el enfoque del sistema se limitará en este proyecto a la hostelería ya que se utilizan mayoritariamente sistemas de colas simples en los cuales toda persona se añade al final de la misma cola. Otros tipos de negocios disponen de varios tipos de colas, según la función para la que necesitas ser atendido, se descartan estos tipos de negocios del alcance, pero podrán tratarse específicamente como proyectos derivados a partir del resultado de este trabajo, o bien, generalizar el resultado de este proyecto para aumentar su alcance.

El prototipo final deberá ser capaz de permitir crear salas virtuales en las que otros usuarios podrán unirse para hacer la cola e interactuar entre ellos. Una vez el usuario sea atendido tendrá la opción de responder una pequeña encuesta con la finalidad de mejorar el servicio.

1.5 Objetivos

El objetivo es poder permitir a los usuarios disfrutar libremente del tiempo que se dispone durante el periodo de espera hasta su turno. El sistema debe de ser fácil de usar, rápido e intuitivo para que pueda ser utilizado por la gran mayoría de la población.

Por lo tanto, este objetivo se cubrirá y concretará mediante los siguientes resultados:

- Diseño e implementación del prototipo funcional.
- Ofrecer al usuario una experiencia positiva y agradable.

El prototipo funcional deberá cumplir los siguientes requisitos:

- Ofrecer una interfaz clara e intuitiva, apta para el mayor público posible.
- Mostrar claramente la situación del turno consultado.
- Generar informes mediante el uso de encuestas con el objetivo de mejorar aspectos del sistema y del propio establecimiento.

1.6 Impacto en sostenibilidad, aspectos ético-sociales y de diversidad

El impacto en la sostenibilidad es positivo si el sistema llega a reemplazar alguno de los actuales sistemas de turnos que se han descrito anteriormente. Con este sistema no hace falta la impresión de ningún tique por lo que se estaría contribuyendo con el medio ambiente al no desperdiciar papel, de la misma manera, también substituiría a los dispositivos físicos llamados “avisadores” haciendo innecesaria su producción.

Se debe tener en cuenta a toda la población, muchas personas no disponen de un *smartphone* o no tienen las habilidades o capacidades suficientes como para utilizarlo, se debe ofrecer facilidad de uso, pensando en las personas con menos capacidades digitales como son los ancianos.

1.7 Enfoque y método

Si la realización de esta aplicación hubiese sido encargada por una persona ajena, la metodología que muy posiblemente utilizaría sería la agile, buscando la satisfacción del cliente y mejorando la calidad del producto final mediante diversos *sprints*.

En este caso, por estar circunscrito en un TFG y desarrollado íntegramente por una única persona, he optado por la metodología clásica en cascada, por lo que las tareas estarán claramente definidas y estructuradas.

Se va a dedicar un tiempo adicional en el estudio de la tecnología web a utilizar ya que la elección de una mala tecnología para el proyecto podría ser catastrófico en una etapa avanzada del desarrollo.

Durante la fase de construcción del sistema se va a utilizar una metodología iterativa en la que cada iteración resultará en un producto funcional el cual añade valor al producto final, por lo tanto, se utilizará una metodología mixta, en cascada pero con diversas iteraciones en las tareas de construcción y prueba.

Una vez obtenidas todas las funcionalidades deseadas tendremos el producto terminado.

1.8 Motivación personal

Durante los últimos años he tenido especial interés en la automatización de procesos. Poder automatizar un proceso manual para que sea completamente autónomo o automático y requiera la interacción mínima de una persona me produce satisfacción y llega a ser gratificante.

1.9 Conocimiento previo

En mi actual trabajo se desarrollan terminales los cuales incluyen varios dispositivos para poder automatizar al máximo los procesos. Estos terminales incluyen ordenadores con pantalla táctil, lectores de tarjetas RFID, lectores de códigos de barras, interfonos, cámaras de reconocimiento de matrículas e impresoras de tiques de diferentes tamaños.

Las aplicaciones que desarrollamos son de escritorio por lo que no tengo experiencia previa en el entorno web.

2. Planificación

2.1 Introducción

En la planificación vamos a detallar todas las tareas relacionadas con el desarrollo del proyecto, remarcando los hitos y las fechas claves. Con la temporalización completa de las tareas abstraemos el esfuerzo y el coste económico necesario para la realización de estas.

Finalmente se han evaluado los posibles riesgos que pueden acontecer durante el transcurso de vida del proyecto, estableciendo acciones para prevenir o corregir estos riesgos.

2.2 Relación de hitos

A continuación, vamos a visualizar la tabla con los hitos alcanzados durante toda la etapa del proyecto. Existen una serie de hitos ya predefinidos en el proyecto los cuales corresponden a los entregables de las actividades para la realización del TFG: Entrega Informe Ejecutivo (PEC1), Informe de seguimiento (PEC2) y Finalización desarrollo entorno web (PEC 3).

Hito	Fecha
Planteamiento del proyecto	19/10/2022
Planificación del proyecto	21/10/2022
Entrega Informe Ejecutivo (PEC 1)	23/10/2022
Finalización del diseño de la interfaz	06/11/2022
Informe de seguimiento (PEC 2)	20/11/2022
Finalización de pruebas	29/12/2022
Finalización desarrollo entorno web (PEC 3)	05/01/2023

Los dos primeros hitos se realizan en la etapa inicial del proyecto y se entregan juntamente con el informe ejecutivo.

Una vez superado el plazo de entrega del informe ejecutivo se inicia la ejecución del proyecto, dicha ejecución comienza por el análisis del diseño de interfaz, punto muy importante para tener clara toda la estructura visual que va a tener la aplicación. Será de gran ayuda para el programador que debe construir la aplicación posteriormente. Al finalizar el diseño se debe entregar al jefe de proyectos para que compruebe la calidad.

El hito más importante del proyecto es el de finalizar el desarrollo del entorno web, ya con las pruebas de *testing* finalizadas satisfactoriamente. Este punto se considera el más importante porque es el producto principal del proyecto.

2.3 Grupos de tareas

Para alcanzar los objetivos del proyecto se aplicará la metodología PMBOK, utilizando la estructura de desglose de trabajo (EDT) facilitando de esta manera la estimación de los recursos y el cálculo del tiempo y el coste.

Tras el análisis se establecen los siguientes grupos de tareas:

Gestión del proyecto: analizar el proyecto en su totalidad para realizar una valoración de la viabilidad, creación de las tareas, temporalización de cada fase e informes de seguimiento. Realización del plan de riesgos, valoración económica del esfuerzo total y guía de implantación del prototipo funcional.

Este grupo es el único que permanecerá abierto durante la vida del proyecto.

Análisis: determinar los requisitos funcionales y no funcionales, definición de los *stakeholders*, casos de uso, diagrama de entidades y modelo de datos.

Diseño: arquitectura del sistema, diseño de toda la interfaz de la aplicación, diseño de datos, diseño funcional y diseño de pruebas.

Construcción: construcción del entorno de desarrollo del prototipo y creación de los manuales necesarios para su utilización.

Pruebas: diseñar y ejecutar diferentes pruebas para comprobar el correcto funcionamiento del sistema.

2.4 Temporalización/esfuerzo

El esfuerzo total es de aproximadamente 230 horas de trabajo, se estima que el proyecto tenga una duración de 83 días naturales (55 días hábiles) teniendo como fecha de inicio el 15/10/2022 y como fecha final el 05/01/2023.

A continuación, vamos a detallar cada una de las tareas que componen el proyecto, especificando el responsable, la descripción de la tarea y su esfuerzo:

- Propuesta del proyecto: especificación formal del problema y su propuesta de solución, definiendo unos objetivos y resultados claros y bien determinados.

Responsable: director de proyectos. Tiempo destinado: 4 horas.

- Planificar tareas: definición de los hitos relevantes del proyecto, grupos de tareas y descripción detallada de cada tarea indicando el rol al que está asignada. Se han destinado más horas a esta tarea por la importancia y el impacto que puede tener la planificación en el transcurso del proyecto.

Responsable: director de proyectos. Tiempo destinado: 12 horas.

- Valoración económica: incluye principalmente el esfuerzo (horas) destinado para la realización completa del proyecto.

Responsable: director de proyectos. Tiempo destinado: 8 horas.

- Análisis de riesgos: analizar los diferentes riesgos que puedan surgir, el impacto que supone cada uno, la probabilidad de producirse y las acciones preventivas y/o correctivas a realizar en caso.

Responsable: director de proyectos. Tiempo destinado: 4 horas.

- Definir requisitos funcionales: identificar todas las funcionalidades que debe cumplir la plataforma.

Responsable: analista de software. Tiempo destinado: 4 horas.

- Definir requisitos no funcionales: identificar todos los requisitos no funcionales que debe tener el sistema.

Responsable: analista de software. Tiempo destinado: 4 horas.

- Definición de stakeholders: identificar las diferentes partes interesadas que van a utilizar la aplicación.

Responsable: analista de software. Tiempo destinado: 4 horas.

- Definición de casos de uso: descripción detallada de los diferentes procesos que se realizarán utilizando la plataforma.

Responsable: analista de software. Tiempo destinado: 4 horas.

- Definición de encuestas: definición de las encuestas que se van a realizar a los usuarios una vez utilicen la plataforma.

Responsable: analista de software. Tiempo destinado: 4 horas.

- Estudio tecnologías web: estudio en profundidad de las actuales tecnologías web para escoger con cual se realizará la construcción del prototipo Esta tarea es de vital importancia ya que la mala elección de la tecnología podría hacer fracasar el proyecto.

Responsable: analista de software. Tiempo destinado: 28 horas.

- Diseñar interfaz: realizar el diseño gráfico de todas las pantallas que formarán la plataforma.

Responsable: desarrollador UI/UX. Tiempo destinado: 12 horas.

- Diseñar base de datos: realizar el diseño lógico y físico de toda la base de datos.

Responsable: desarrollador web. Tiempo destinado: 8 horas.

- Diseñar pruebas: definir las diferentes pruebas que se van a realizar para validar el funcionamiento del prototipo.

Responsable: desarrollador web. Tiempo destinado: 4 horas.

- **Desarrollo de la aplicación web:** construcción completa de la plataforma web hasta tener finalizado el prototipo funcional.

Responsable: desarrollador web. Tiempo destinado: 100 horas.

- **Manual de instalación:** elaboración del manual de instalación detallando todos los pasos para instalar la plataforma en un entorno nuevo.

Responsable: técnico de sistemas. Tiempo destinado: 4 horas.

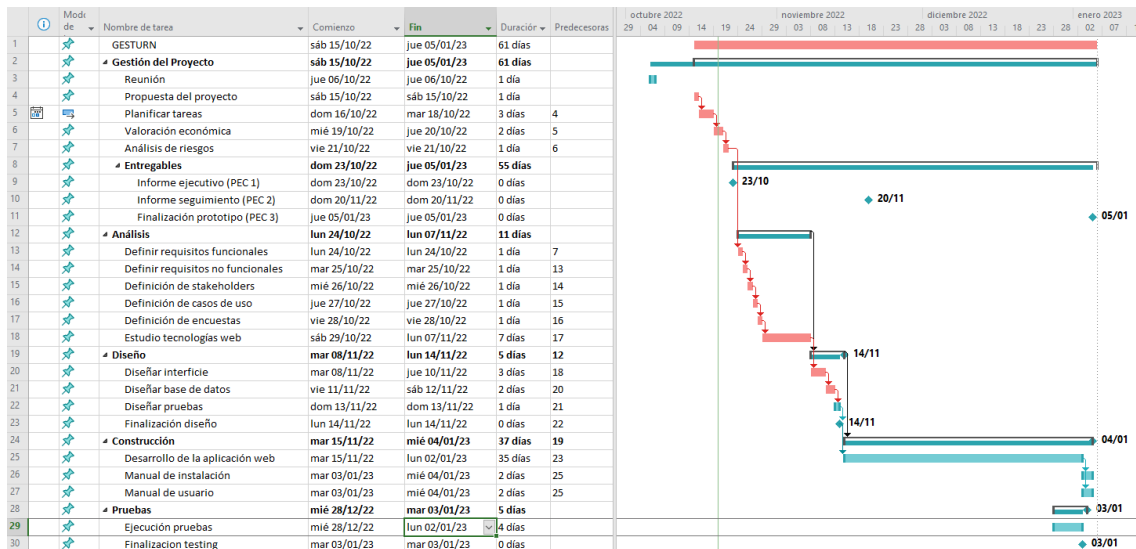
- **Manual de usuario:** elaboración del manual de usuario detallando todos los pasos que debe seguir para la utilización de la plataforma.

Responsable: desarrollador web. Tiempo destinado: 2 horas.

- **Ejecución y evaluación de pruebas:** ejecución de todas las pruebas diseñadas para validar el correcto funcionamiento del sistema.

Responsable: desarrollador web. Tiempo destinado: 14 horas.

Podemos ver representada de forma clara la temporalización de cada tarea en el diagrama de Gantt:



En la siguiente tabla podemos comparar el esfuerzo y duración cada grupo de tareas:

Grupo de tarea	Fecha inicio	Fecha fin	Duración	Esfuerzo
Gestión de proyecto	15/10/2022	05/01/2023	83 días	28 horas
Análisis	24/10/2022	07/11/2022	15 días	48 horas
Diseño	08/11/2022	14/11/2022	7 días	24 horas
Construcción	15/11/2022	04/01/2023	51 días	100 horas
Pruebas	28/12/2022	03/01/2026	7 días	20 horas

Si sumamos el esfuerzo de cada grupo obtendremos un total de 220 horas, se ha previsto inicialmente que la duración del proyecto es de 230 horas, por lo tanto, hemos dejado un pequeño margen de 10 horas para contingencias en el caso de que surja cualquier problema o incluso para mejoras de calidad, suponiendo que el proyecto avanza según lo planificado.

2.5 Valoración económica

Para el cálculo de la valoración económica se ha tenido en cuenta exclusivamente el coste del personal involucrado en el proyecto, no se ha previsto contratar ningún servicio adicional proveniente de una empresa externa. Tampoco se ha contemplado el alojamiento del prototipo en un servidor de terceros.

Los **roles** identificados para la realización del proyecto son los siguientes:

- **Director de proyectos:** seguimiento de la correcta planificación del proyecto, elaborar informes, reuniones, gestión de riesgos...
- **Analista de software:** análisis de casos, encuestas y elección de la mejor tecnología para el desarrollo del proyecto, entre otras funciones.
- **Desarrollador web:** encargado de la construcción de la página web.
- **Desarrollador UI/UX:** encargado de implementar una interfaz atractiva, intuitiva y que ofrezca una buena experiencia al usuario.
- **Técnico de sistemas:** encargado en la implementación / configuración del sistema ya sea en servidores privados o públicos, también se encarga de la seguridad del servidor.

Tras analizar ofertas de trabajo para cada rol en diversos portales de empleo¹, se ha determinado el salario/año. Para obtener el coste anual se ha tenido en cuenta la siguiente fórmula:

Coste anual² = 70% sueldo bruto, 20% seguridad social y 10% indemnización y prestaciones sociales.

Considerando 1800 horas efectivas anuales (BOE, 22 de febrero de 2018³) de trabajo se ha determinado el coste/hora para cada rol.

Para facilitar el cálculo del coste de cada rol profesional vamos a utilizar la siguiente tabla:

Rol	Coste hora	Horas trabajadas	Total
Director de proyectos	50€	28 horas	1400€
Analista de software	40€	48 horas	1920€
Desarrollador web	35€	116* horas	4060€
Desarrollador UI/UX	35€	24 horas	840€

¹ LinkedIn: Red social orientada al uso empresarial, permitiendo la búsqueda de empleo. Sitio web: <https://es.linkedin.com/>

² Fórmula coste/sueldo: <https://www.personio.es/glosario/coste-de-trabajador-para-empresa/>

³ BOE: https://www.boe.es/diario_boe/txt.php?id=BOE-A-2018-3156

Técnico de sistemas	30€	4 horas	120€
Contingencia	50€	10 horas	500€
		230 horas	8840€

**el desarrollador web también se encargará de realizar las pruebas de funcionamiento.*

Si sumamos el total de horas trabajadas podemos comprobar que efectivamente obtenemos las 220 horas previstas, el coste total del personal asciende a 8340€.

Al total obtenido se deben sumar 10 horas para el plan de contingencia y así tener margen de maniobra en el caso de suceder cualquier imprevisto. En este caso el plan de contingencia tendría un coste de 500€, para el cálculo se ha utilizado el coste más elevado de los diferentes roles. Si sumamos esta cantidad al total inicial obtenemos un **coste final estimado de 8840€.**

2.6 Análisis de riesgos

Se han evaluado los posibles riesgos que pueden acontecer durante el transcurso de vida del proyecto, estableciendo acciones para prevenir o corregir estos riesgos.

A continuación, vamos a ver en detalle cada uno de los riesgos que se han detectado:

Código	Riesgo	Descripción	Probabilidad de aparición	Impacto
R1	Desconocimiento tecnologías	Desconocimiento de las diferentes tecnologías para el desarrollo web.	Baja	Alto
R2	Inexperiencia planificación	Falta de conocimientos o de experiencia en la planificación de proyectos.	Media	Medio
R3	Alcance excesivo	Es complicado desarrollar aplicaciones que tengan como objetivo la mayoría de la población.	Baja	Medio
R4	Problemas personales	El desarrollo completo del proyecto se realiza por una única persona que adoptará los distintos roles, si enferma o le surge algún problema puede afectar seriamente a la entrega.	Baja	Alto
R5	Problemas técnicos	Durante la construcción del prototipo pueden surgir errores inesperados.	Alta	Alto
R6	Experiencia negativa para el usuario	Si la utilización del prototipo no es intuitiva o incluso lenta, puede ocasionar descontento para el usuario.	Baja	Alto

R7	Pérdida de información	Por un error es posible que se pierda una parte o la totalidad del trabajo realizado.	Baja	Alto
----	------------------------	---	------	------

A continuación, vamos a definir una serie de **acciones preventivas (AP)** para reducir la probabilidad que estos riesgos lleguen a suceder, para reducir el impacto vamos a definir también unas **acciones correctivas (AC)**:

Código	Acción	Descripción	Riesgo asociado
AP-1	Formación	Realizar una formación previa al personal para que conozcan las tecnologías actuales.	R1, R2
AP-2	Estudio	Destinar más tiempo en la elección de la tecnología para asegurarse que es la correcta.	R1
AP-3	Definir alcance	Definir con más detalle el alcance puede marcar la diferencia ya que a veces tendemos a generalizar demasiado.	R3
AP-4	Crear margen de horas	En la planificación se contemplan algunos días adicionales que pueden ser utilizados en caso de emergencia.	R4, R5
AP-5	Revisar cada versión de la aplicación	Durante el proceso de construcción analizar si cada mejora penaliza el rendimiento o la utilización de la aplicación.	R6
AP-6	Restaurar copia de seguridad	Si realizamos copias de seguridad periódicamente podemos recuperar la totalidad o la mayor parte de los datos.	R7
AP-7	Trabajar en la nube	Existen herramientas que permiten realizar copias exactas de todo nuestro trabajo y almacenarlo en la nube.	R7
AC-1	Reducir alcance	Limitar o descartar alguna funcionalidad del proyecto	R3
AC-2	Incrementar horario laboral	Aumentar la carga de trabajo para poder finalizar la tarea dentro del plazo establecido.	R4, R5

3. Análisis

3.1 Situación actual

Para realizar el análisis de la situación actual se ha hecho previamente una búsqueda extensa de los sistemas actuales en funcionamiento que cubren funciones similares a la que pretende cubrir este proyecto.

Durante la búsqueda se ha reflejado un alto número de resultados de aplicaciones funcionales para ámbitos similares como por ejemplo aplicaciones de citas previas o aplicaciones de turnos laborales. Estos ámbitos pueden confundirse con el tema de este proyecto, pero son totalmente distintos.

El proyecto nace para cubrir la necesidad de poder realizar una espera de forma remota que hasta día de hoy se tenía que hacer de forma tradicional *in-situ*.

Solo se ha encontrado un sistema que realiza la misma función que este proyecto, la utilización de unos dispositivos que ya se pueden considerar arcaicos. Estos dispositivos se denominan avisadores, vamos a realizar una pequeña tabla con las ventajas y desventajas de la utilización de estos dispositivos:

Ventajas	Desventajas
<ul style="list-style-type: none"> - Evitan aglomeraciones en las colas. - Aumenta la satisfacción de los clientes. - Fácil de utilizar. 	<ul style="list-style-type: none"> - Dispositivos muy costosos. - Posibilidad de extraviarse. - Batería limitada. - Rango de alcance corto (menos de un kilómetro). - Dimensiones de los dispositivos. - Peso. - Requieren de mantenimiento. - Difícil de reparar.

Como se puede observar en la tabla, las desventajas de estos dispositivos son múltiples. Tenía sentido utilizarlos hace unas décadas, pero hoy en día se pueden desarrollar sistemas con muchas menos carencias.

Para poder hacerse una idea de cómo son estos dispositivos se adjuntan algunos ejemplos reales con imágenes y una pequeña descripción:

Ejemplo 1

Avisadores marca VEVOR.

Parte delantera:



Fuente: <https://www.vevor.es/>

Parte trasera:



Fuente: <https://www.vevor.es/>

Precio: 106,99€.

Características:

- 16 dispositivos.
- Dos modos de llamada (vibración, señales luminosas y señales sonoras o vibración y señales luminosas).
- Alcance máximo de 300-500 metros.
- Batería de 200 mAh⁴ con autonomía de 24 horas de funcionamiento.

⁴ Miliamperios o mAh.

Ejemplo 2

Avisadores marca TG.

Parte frontal y trasera:



Fuente: <https://www.tpvgalicia.com/>

Precio: 387€.

Características:

- 10 dispositivos, ampliable hasta 500.
- Alcance máximo de 500-900 metros en exterior.
- Alcance máximo de 100 metros en interior.
- Autonomía de 48 horas de funcionamiento.

No se ha encontrado ninguna aplicación móvil o página web que cubra esta funcionalidad ni que permita realizar y gestionar turnos de forma remota.

A continuación, vamos a realizar otra tabla con las ventajas y desventajas de la utilización del sistema planteado para poder compararlo fácilmente con la única tecnología que existe actualmente:

Ventajas	Desventajas
<ul style="list-style-type: none"> - Sin coste. - Alcance ilimitado. - No requiere mantenimiento. - Visualización de los turnos existentes previos al nuestro. - Evitan aglomeraciones en las colas. - Aumenta la satisfacción de los 	<ul style="list-style-type: none"> - Requieren un smartphone con conexión a internet. - Puede no ser accesible para toda la población.

clientes. - Fácil de utilizar. - Estadísticas.	
--	--

La utilización de este tipo de solución es mucho más lógica gracias a la evolución de la tecnología.

3.2 Análisis del dominio

3.2.1 Roles

En el aplicativo existen únicamente tres roles:

- Cliente de una cola: utilizará el aplicativo para realizar la cola virtual y opcionalmente contestar las encuestas una vez finalizado el uso. El cliente tiene la opción de eliminar su turno de la cola.
- Gestor de una cola: usuario con permisos para poder gestionar los turnos de una sala. Tienen la capacidad de eliminar cualquier turno ya finalizado de la sala y crear de nuevos.
- Propietario de la cola: usuario creador de la cola. Tiene los mismos permisos que los gestores, pero además puede invitar a otros usuarios a ser gerentes de su sala.

Estos tres roles relacionan a los usuarios con las colas virtuales (salas). Esto no significa que un usuario solo pueda tener un rol, es muy posible que un mismo usuario sea cliente de diversas colas y gerente o dueño de otras.

3.3 Análisis funcional

3.3.1 Análisis de requisitos

En este apartado se exponen los requisitos que han surgido durante el análisis del dominio. Los requisitos funcionales definen funciones que debe realizar el sistema, ya sean cálculos, detalles técnicos, manipulación de datos u otras funcionalidades. Por otra parte, los requisitos no funcionales representan las características de funcionamiento y restricciones de la aplicación.

3.3.1.1 Requisitos funcionales

La identificación y gestión de estos requisitos es una actividad crítica, se debe realizar con mucho rigor ya que una definición deficiente de estos requisitos puede llevar al fracaso del proyecto.

A continuación, vamos a visualizar una tabla con todos los **requisitos funcionales (RF)**:

Código	Descripción
--------	-------------

RF1	Los usuarios deben registrarse y autenticarse para utilizar el sistema.
RF2	Los usuarios solo podrán tener un turno activo en cada sala.
RF3	Los gestores podrán crear nuevos turnos en las salas que dispongan de permiso.
RF4	Los gestores podrán finalizar turnos de usuarios de la sala que gestionan.
RF5	Los propietarios de la sala poseen los mismos permisos que los gestores, pero además pueden invitar a otros usuarios a ser gestores de sus salas.
RF6	Los propietarios pueden revocar los permisos de los gestores.
RF7	Los usuarios propietarios pueden crear más de una sala.
RF8	El sistema ofrecerá la posibilidad de contestar una encuesta a los usuarios que finalicen su turno.
RF9	El sistema creará un código QR para facilitar unirse a cada sala.
RF10	Los usuarios deben poder visualizar los turnos pendientes de la sala previos al suyo.

3.3.1.2 Requisitos no funcionales

La identificación y definición de este tipo de requisitos puede presentar algunas dificultades ya que la conformidad de estos puede ser sujeta a libre interpretación, por eso, es importante definirlos con el máximo detalle posible.

A continuación, vamos a visualizar una tabla con todos los **requisitos no funcionales (RNF)**:

Código	Descripción
RNF1	La interfaz debe ser intuitiva.
RNF2	El sistema debe ser fácil de utilizar.
RNF3	El sistema debe ser rápido y dar una respuesta casi inmediata.
RNF4	Debe ser responsiva y compatible con la mayoría de los dispositivos.
RNF5	Los permisos de una cola solo pueden ser editados por su propietario.
RNF6	El sistema debe implementar protocolos de comunicación seguros.
RNF7	El sistema estará disponible en castellano.
RNF8	Se utilizará una base de datos relacional.
RNF9	El sistema debe mostrar mensajes orientativos o informativos para guiar al usuario.

3.4 Stakeholders

A continuación, se identifican los diferentes perfiles interesados que utilizan la aplicación. Principalmente son tres grupos:

- Usuarios o clientes: Persona que quiere utilizar el sistema para realizar una cola de forma remota y poder visualizar e incluso finalizar su turno una vez sea atendido.
- Gestores o trabajadores: Persona que tiene permisos para visualizar los turnos de una sala, crearlos, modificarlos o eliminarlos.
- Propietario de sala: Persona que dispone de los mismos permisos que los gestores, pero además puede añadir o eliminar gestores de sus salas.

3.5 Casos de uso

Descripción detallada de los diferentes procesos que se realizan en la plataforma por los distintos tipos de usuarios.

Los usuarios con rol de gestor heredan las funciones de los usuarios o clientes y amplían sus funcionalidades con la capacidad de crear turnos para otros clientes y gestionar cualquier turno de la sala.

Los usuarios con rol de propietario heredan las funciones de los gestores y extienden sus funcionalidades a poder invitar usuarios a ser gestores de sus salas, eliminarlos o modificar información de la sala.

Código o identificador:	CU01
Descripción:	Registro en la aplicación.
Actor primario:	Usuario no registrado.
<i>Stakeholders</i> interesados:	Clientes, gestores o propietarios.
Precondición:	El usuario no debe estar registrado en el sistema y debe tener correo electrónico.
Garantías en caso de éxito:	Se guarda la información y el usuario queda registrado.
Escenario principal de éxito o operaciones:	<ol style="list-style-type: none"> 1. El usuario accede a la página principal. 2. El usuario accede al registro de usuarios. 3. El sistema muestra la pantalla de registro. 4. El usuario rellena la información solicitada. 5. El sistema valida los datos y crea el usuario.
Extensiones:	<ol style="list-style-type: none"> 4a. El usuario introduce algún dato con formato incorrecto. 4b. El sistema informa al usuario del dato con formato

	<p>incorrecto.</p> <p>4c. Volvemos al paso 4.</p> <p>5a. El sistema detecta que ya existe el usuario.</p> <p>5b. El sistema informa al usuario de que ya existe esa cuenta.</p> <p>5c. Volvemos al paso 4.</p>
--	--

Código o identificador:	CU02
Descripción:	Identificarse en el sistema.
Actor primario:	Usuario registrado.
<i>Stakeholders</i> interesados:	Clientes, gestores o propietarios.
Precondición:	El usuario debe estar registrado en el sistema.
Garantías en caso de éxito:	El usuario puede acceder a la aplicación con la cuenta deseada.
Escenario principal de éxito o operaciones:	<ol style="list-style-type: none"> 1. El usuario accede a la página principal. 2. El usuario accede a la pantalla de <i>login</i>. 3. El usuario introduce sus datos. 4. El sistema valida e identifica al usuario y lo redirige a la ventana principal.
Extensiones:	<p>4a. El usuario o contraseña no es correcto.</p> <p>4b. El sistema muestra un aviso indicando el problema.</p> <p>4c. Volvemos al paso 3.</p>

Código o identificador:	CU03
Descripción:	Salir del sistema.
Actor primario:	Usuario registrado.
<i>Stakeholders</i> interesados:	Clientes, gestores o propietarios.
Precondición:	El usuario debe estar identificado en el sistema.
Garantías en caso de	El usuario no estará identificado en el sistema.

éxito:	
Escenario principal de éxito o operaciones:	<ol style="list-style-type: none"> 1. El usuario presiona el botón para salir del sistema. 2. El sistema verifica si el usuario realmente quiere salir de la aplicación. 3. El usuario confirma la petición de salir. 4. El sistema cierra la sesión del usuario.
Extensiones:	

Código o identificador:	CU04
Descripción:	Consultar o modificar datos personales de la cuenta.
Actor primario:	Usuario registrado.
<i>Stakeholders</i> interesados:	Clientes, gestores o propietarios.
Precondición:	El usuario debe estar identificado en el sistema.
Garantías en caso de éxito:	El usuario puede visualizar o modificar su información personal.
Escenario principal de éxito o operaciones:	<ol style="list-style-type: none"> 1. El usuario accede a la ventana de datos personales. 2. El sistema muestra la pantalla de datos personales. 3. El usuario visualiza o modifica sus datos. 4. El sistema valida el formato de los datos introducidos. 5. El sistema guarda la modificación de los datos.
Extensiones:	<p>4b. El sistema informa al usuario del dato con formato incorrecto.</p> <p>4c. Volvemos al paso 3.</p> <p>5a. El sistema detecta si existe algún problema con la modificación.</p> <p>5b. El sistema informa al usuario del problema encontrado.</p> <p>5c. Volvemos al paso 4.</p>

Código o identificador:	CU05
Descripción:	Crear una sala de turnos.

Actor primario:	Usuario registrado.
<i>Stakeholders</i> interesados:	Propietarios.
Precondición:	El usuario debe estar identificado en el sistema.
Garantías en caso de éxito:	Se creará una nueva sala y el usuario será su propietario.
Escenario principal de éxito o operaciones:	<ol style="list-style-type: none"> 1. El usuario pulsa el botón de crear una nueva sala. 2. El sistema muestra la pantalla de creación de sala. 3. El usuario rellena los datos de la sala. 4. El sistema valida los datos y crea la sala.
Extensiones:	<p>4a. El sistema detecta que el nombre de la sala ya está en uso.</p> <p>4b. El sistema informa al usuario que el nombre de la sala está en uso.</p> <p>4c. Volvemos al paso 3.</p>

Código o identificador:	CU06
Descripción:	Buscar sala de turnos.
Actor primario:	Usuario registrado.
<i>Stakeholders</i> interesados:	Clientes, gestores o propietarios.
Precondición:	El usuario debe estar identificado en el sistema.
Garantías en caso de éxito:	El usuario visualizará la sala de turnos en la que está interesado.
Escenario principal de éxito o operaciones:	<ol style="list-style-type: none"> 1. El usuario escribe el nombre de la sala que quiere buscar y pulsa en buscar. 2. El sistema muestra la pantalla de la sala con los turnos e información de esta.
Extensiones:	<p>2a. El sistema detecta que el nombre de la sala buscada no existe.</p> <p>2b. El sistema informa al usuario de que el nombre de la sala no existe.</p> <p>2c. Volvemos al paso 1.</p>

Código o identificador:	CU07
Descripción:	Creación de turno en sala.
Actor primario:	Usuario registrado.
<i>Stakeholders</i> interesados:	Clientes, gestores o propietarios.
Precondición:	El usuario debe estar identificado en el sistema. El usuario debe estar visualizando la pantalla de la sala con los turnos. El usuario no debe tener ningún turno existente en la sala.
Garantías en caso de éxito:	El usuario creará un turno para él.
Escenario principal de éxito o operaciones:	<ol style="list-style-type: none"> 1. El usuario pulsa en el botón de crear turno nuevo. 2. El sistema muestra la pantalla de la sala con su turno creado.
Extensiones:	

Código o identificador:	CU08
Descripción:	Visualizar el estado del turno.
Actor primario:	Usuario registrado.
<i>Stakeholders</i> interesados:	Clientes, gestores o propietarios.
Precondición:	El usuario debe estar identificado en el sistema. El usuario debe tener un turno existente en la sala.
Garantías en caso de éxito:	El usuario obtendrá información del estado de su turno.
Escenario principal de éxito o operaciones:	<ol style="list-style-type: none"> 1. El usuario escribe el nombre de la sala que quiere buscar y pulsa en buscar. 2. El sistema muestra la pantalla de la sala con los turnos e información de esta.
Extensiones:	

Código o identificador:	CU09
Descripción:	Finalizar turno existente.
Actor primario:	Usuario registrado.
<i>Stakeholders</i> interesados:	Clientes, gestores o propietarios.
Precondición:	El usuario debe estar identificado en el sistema. El usuario debe estar visualizando la pantalla de la sala con los turnos. El usuario debe tener un turno existente en la sala.
Garantías en caso de éxito:	El usuario finalizará su turno existente.
Escenario principal de éxito o operaciones:	<ol style="list-style-type: none"> 1. El usuario pulsará el botón de finalizar turno. 2. El sistema finalizará el turno del usuario. 3. El sistema le redirigirá a la pantalla principal de la aplicación.
Extensiones:	

Código o identificador:	CU10
Descripción:	Responder encuestas de satisfacción.
Actor primario:	Usuario registrado.
<i>Stakeholders</i> interesados:	Clientes.
Precondición:	El usuario debe estar identificado en el sistema. El usuario debe finalizar su turno.
Garantías en caso de éxito:	El sistema almacenará las respuestas de la encuesta del usuario.
Escenario principal de éxito o operaciones:	<ol style="list-style-type: none"> 1. El sistema mostrará la encuesta al usuario. 2. El usuario rellenará los datos de la encuesta. 3. El sistema informará al usuario de que la encuesta ha sido guardada correctamente. 4. El sistema redirigirá al usuario a la pantalla principal.
Extensiones:	

Código o identificador:	CU11
Descripción:	Crear turno para usuario ajeno a la aplicación.
Actor primario:	Gestor o propietario.
<i>Stakeholders</i> interesados:	Clientes, gestores o propietarios.
Precondición:	El usuario debe estar identificado en el sistema. El usuario debe tener permisos para crear turnos a usuarios ajenos. El usuario debe estar visualizando la pantalla de la sala con los turnos.
Garantías en caso de éxito:	El usuario creará un turno para persona ajena.
Escenario principal de éxito o operaciones:	<ol style="list-style-type: none"> 1. El gestor o propietario pulsa en el botón de crear turno nuevo. 2. El sistema muestra la pantalla de la sala con el nuevo turno creado.
Extensiones:	

Código o identificador:	CU12
Descripción:	Finalizar cualquier turno de la sala.
Actor primario:	Gestor o propietario.
<i>Stakeholders</i> interesados:	Clientes, gestores o propietarios.
Precondición:	El usuario debe estar identificado en el sistema. El usuario debe tener permisos para eliminar turnos a usuarios ajenos. El usuario debe estar visualizando la pantalla de la sala con los turnos.
Garantías en caso de éxito:	El turno ajeno será eliminado.
Escenario principal de	<ol style="list-style-type: none"> 1. El gestor o propietario pulsa sobre el botón de

éxito o operaciones:	<p>finalizar en el turno que desea finalizar.</p> <p>2. El sistema muestra de nuevo la pantalla de la sala de turnos.</p>
Extensiones:	

Código o identificador:	CU13
Descripción:	Modificar información de la sala de turnos.
Actor primario:	Propietario.
<i>Stakeholders</i> interesados:	Propietario.
Precondición:	<p>El usuario debe estar identificado en el sistema.</p> <p>El usuario debe ser el propietario de la sala a modificar.</p> <p>El usuario debe estar visualizando la pantalla de la sala con los turnos.</p>
Garantías en caso de éxito:	El propietario modifica la información de la sala.
Escenario principal de éxito o operaciones:	<ol style="list-style-type: none"> 1. El propietario pulsa sobre el botón de Modificar sala. 2. El sistema muestra la pantalla de edición de la sala con la información. 3. El usuario modifica los datos de la sala. 4. El sistema valida los datos y guarda las modificaciones.
Extensiones:	<p>4a. El sistema detecta que el nombre de la sala ya está en uso.</p> <p>4b. El sistema informa al usuario que el nombre de la sala está en uso.</p> <p>4c. Volvemos al paso 3.</p>

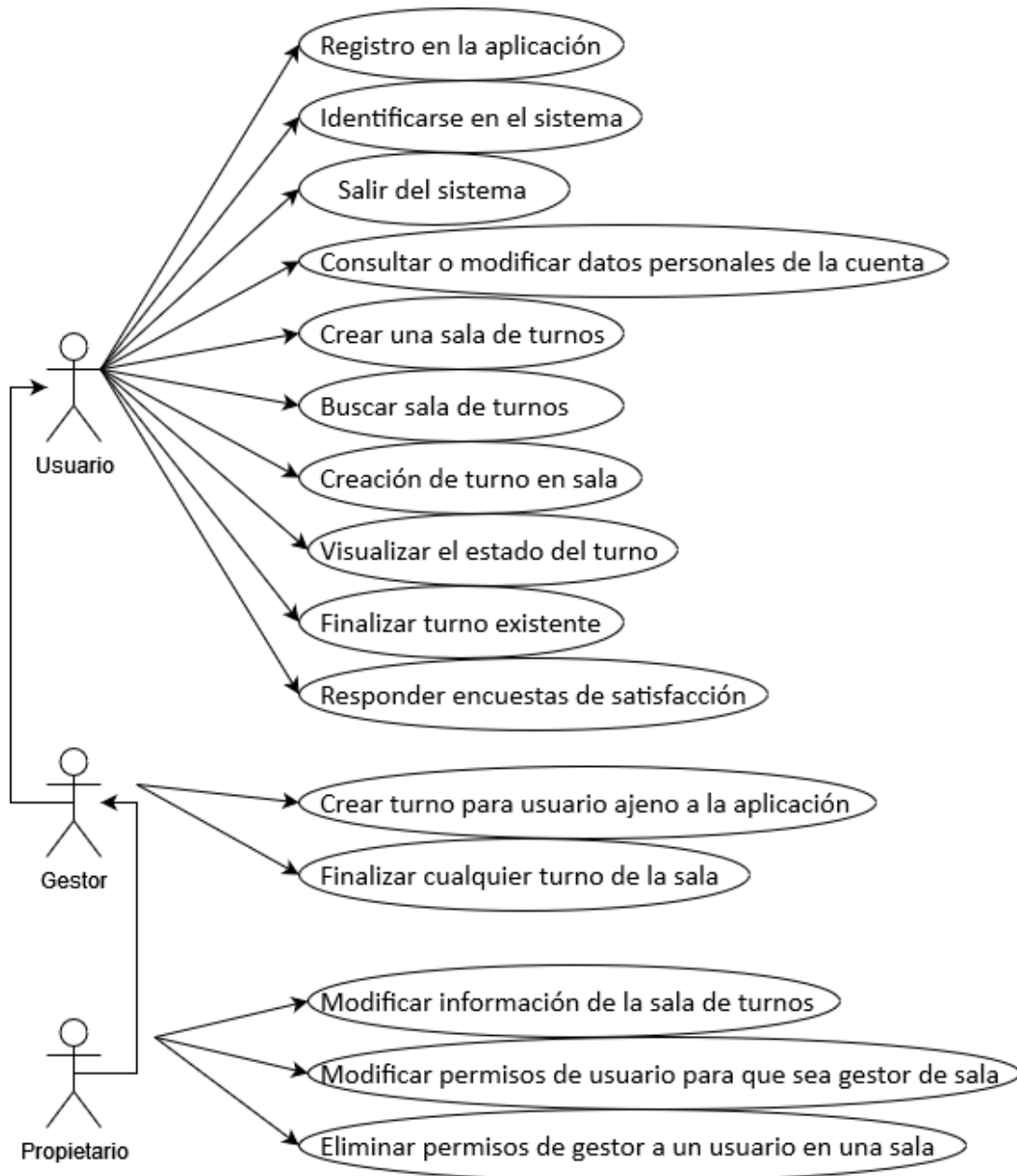
Código o identificador:	CU14
Descripción:	Modificar permisos de usuario para que sea gestor de sala.
Actor primario:	Usuario propietario de sala.
<i>Stakeholders</i>	Gestores y propietarios.

interesados:	
Precondición:	<p>El usuario propietario debe estar identificado en el sistema.</p> <p>El usuario propietario de la sala debe tener creada la sala a la que quiere dar permisos al otro usuario.</p> <p>El usuario que desea ser gestor de la sala debe estar registrado en el sistema.</p>
Garantías en caso de éxito:	El usuario gestor obtendrá los permisos de gestor para la sala correspondiente.
Escenario principal de éxito o operaciones:	<ol style="list-style-type: none"> 1. El propietario de la sala accede a su sala de turnos. 2. El propietario pulsa el botón de añadir gestores a la sala. 3. El propietario de la sala escribe el correo electrónico del usuario al que quiere darle permisos de gestor.
Extensiones:	<p>3a. El sistema verifica si el correo introducido está dado de alta en el sistema.</p> <p>3b. El sistema informa al propietario si se ha concedido satisfactoriamente los permisos al usuario o si hay algún error.</p>

Código o identificador:	CU15
Descripción:	Eliminar permisos de gestor a un usuario en una sala.
Actor primario:	Usuario propietario de sala.
<i>Stakeholders</i> interesados:	Propietarios.
Precondición:	El usuario propietario debe estar identificado en el sistema.
Garantías en caso de éxito:	El usuario gestor perderá sus permisos de gestor en la sala correspondiente.
Escenario principal de éxito o operaciones:	<ol style="list-style-type: none"> 1. El propietario de la sala accede a su sala de turnos 2. El propietario de la sala pulsa el botón de eliminar gestores de la sala. 3. El sistema muestra la lista de usuarios con permisos de gestor en su sala. 4. El propietario de la sala elige el usuario que quiere

	<p>eliminar los permisos de gestor.</p> <p>5. El sistema verifica si el propietario realmente quiere eliminar los permisos a ese gestor.</p> <p>6. El propietario confirma la petición.</p>
Extensiones:	

Como vemos, los casos de uso (CU) de un sistema contienen los requisitos funcionales. Vamos a representar también un esquema de los casos de uso de forma simplificada:



3.6 Definición de encuestas de satisfacción

En este apartado se han definido las encuestas que se realizan a los usuarios una vez utilicen la plataforma, estas encuestas son opcionales por lo que el usuario tendrá la libre elección de contestarlas o no.

Se han definido cinco preguntas específicas de diferentes aspectos de la aplicación para poder tener un *feedback* de la satisfacción del cliente con el uso del aplicativo:

1. ¿Cuán satisfecho se encuentra con el uso general de la aplicación? (1-10)
2. ¿Está satisfecho con el diseño de la aplicación? (1-10)
3. ¿El proceso de crear/visualizar su turno le resultó fácil? (1-10)
4. ¿Cuán probable es que recomiende la aplicación a otros lugares? (1-10)
5. Sugerencias. (campo de texto)

Con los resultados de estas preguntas se puede calcular el índice de satisfacción, un valor numérico que mide hasta qué punto se muestra satisfecho el cliente con nuestro sistema. Además, se puede analizar cuáles son los puntos fuertes y débiles del sistema para posteriormente realizar un análisis y mejorarlo.

3.7 Estudio de las tecnologías web actuales

Estudio en profundidad de las actuales tecnologías web para escoger con cual se realizará la construcción del prototipo Esta tarea es de vital importancia ya que la mala elección de la tecnología podría hacer fracasar el proyecto.

Blazor y Angular son dos *frameworks*⁵ populares para construir aplicaciones web. Ambos tienen sus propias características y capacidades únicas, los dos son adecuados para diferentes tipos de proyectos. Algunas diferencias clave entre los dos frameworks serían:

1. **Lenguaje:** Blazor utiliza C# y sintaxis Razor, mientras que Angular utiliza TypeScript.
2. **Arquitectura:** Blazor es un framework de aplicaciones de una sola página (SPA), mientras que Angular es un framework con todas las funciones que se puede utilizar para crear tanto SPA como aplicaciones tradicionales renderizadas en servidor.
3. **Vinculación de datos:** Blazor utiliza un modelo de programación reactiva para la vinculación de datos, mientras que Angular utiliza un modelo basado en plantillas.

⁵ Framework: Un *framework* es un conjunto de herramientas y librerías que se utilizan para facilitar el desarrollo de aplicaciones

4. **Inyección de dependencias:** Blazor utiliza un contenedor de inyección de dependencias (DI) integrado, mientras que Angular utiliza su propio sistema DI.
5. **Rendimiento:** Blazor utiliza WebAssembly para ejecutar código C# en el navegador, lo que puede dar lugar a tiempos de renderizado más rápidos en comparación con Angular, que depende del tiempo de ejecución de JavaScript.
6. **Comunidad:** Angular es un framework más maduro con una mayor comunidad de desarrolladores, lo que significa que es más probable que encuentres respuestas a tus preguntas y recursos para el aprendizaje y el desarrollo. Blazor, en cambio, es un framework más reciente con una comunidad en crecimiento, pero no está tan bien establecido como Angular.
7. **Pruebas:** Tanto Blazor como Angular tienen sólidos marcos y herramientas de prueba, pero el enfoque de las pruebas puede ser muy diferente. Blazor utiliza una combinación de pruebas unitarias, pruebas de integración y pruebas de extremo a extremo (E2E), mientras que Angular utiliza principalmente pruebas E2E con la herramienta Protractor.
8. **Despliegue:** Tanto Blazor como Angular pueden desplegarse en diversos entornos de alojamiento, incluidas plataformas en la nube como Azure y AWS. Las aplicaciones Blazor pueden desplegarse como SPA independientes o como parte de una aplicación .NET tradicional renderizada en servidor, mientras que las aplicaciones Angular pueden desplegarse como SPA independientes o como parte de una aplicación renderizada en servidor utilizando una herramienta como Angular Universal.

La elección entre Blazor y Angular dependerá de las necesidades y preferencias específicas de cada proyecto. Si el programador se siente cómodo utilizando C# y está buscando un framework que pueda ser utilizado para construir SPAs, entonces Blazor es una buena opción. Por otro lado, si está más familiarizado con TypeScript y quiere un framework con una amplia gama de características y capacidades, entonces Angular podría ser una mejor opción.

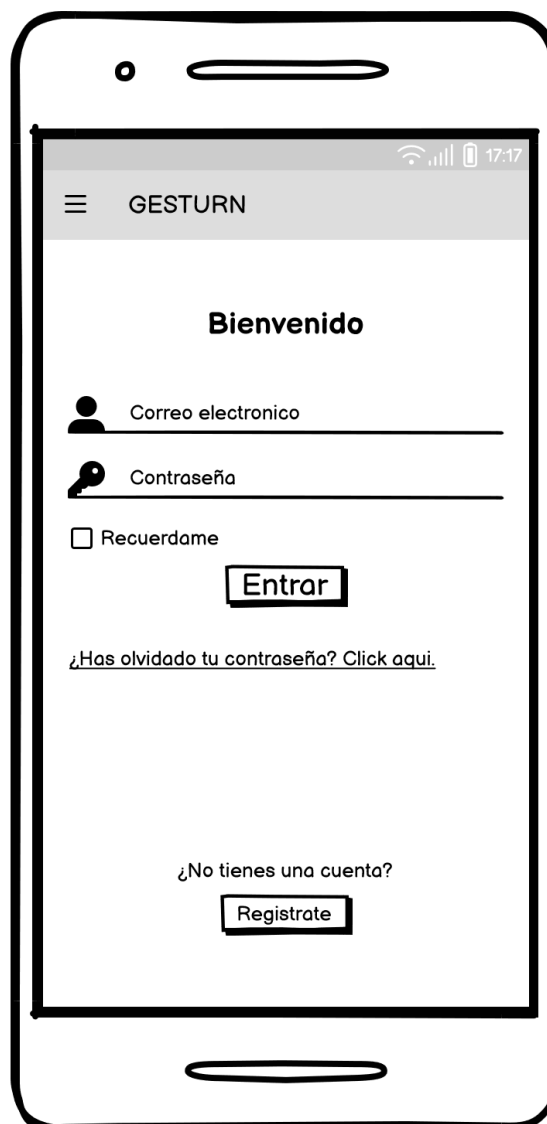
4. Diseño

4.1 Diseño interfaz

A continuación, vamos a visualizar una representación gráfica realizada con *wireframes*⁶ de las diferentes pantallas que contendrá la aplicación. Con la ayuda de esta representación obtendremos una visión más tangible y fácil de entender.

Esta representación no obliga a que el desarrollo de la aplicación se haga exactamente igual, pero sí que sirve como eje central.

Pantalla principal del sistema cuando el usuario no está identificado:

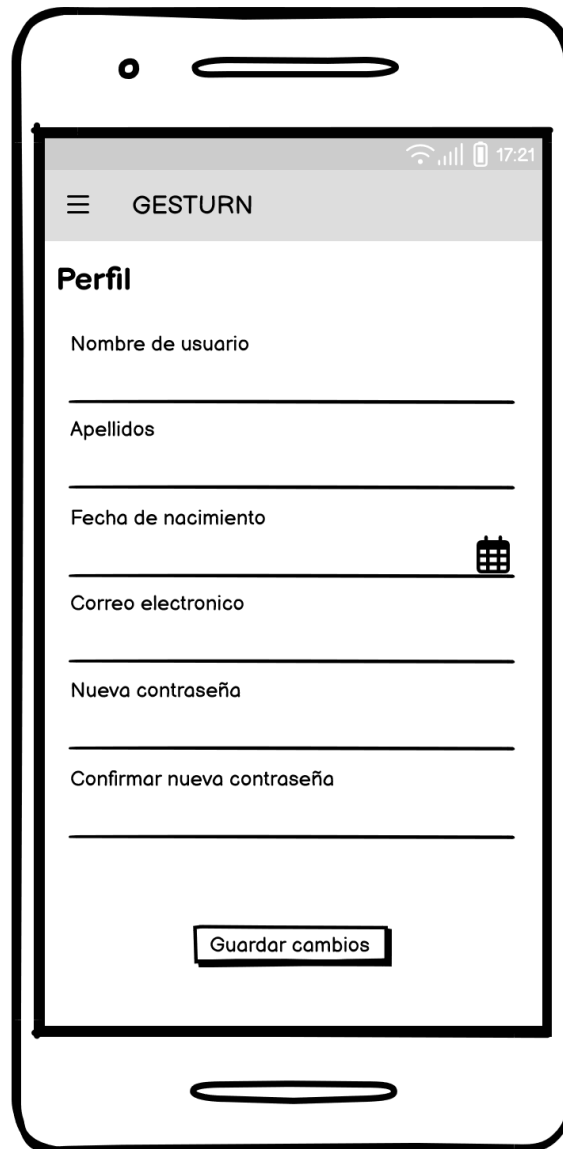


⁶ Wireframe, también conocido como un esquema de página o plano de pantalla, es una guía visual que representa el esqueleto o estructura visual de un sitio web.

Pantalla de registro para nuevo usuario:

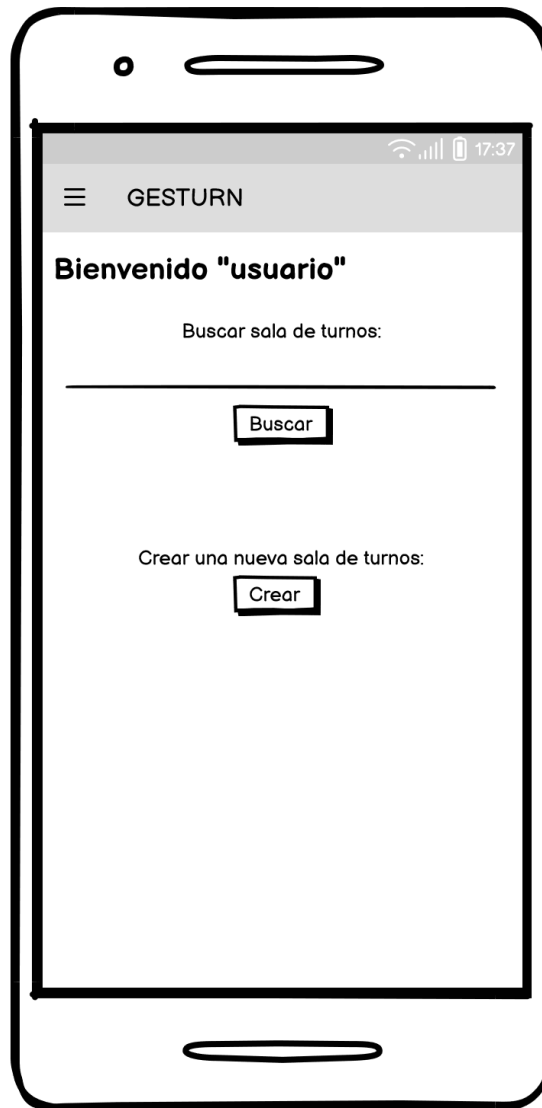
The image shows a wireframe of a mobile application registration screen. At the top, there is a status bar with signal strength, Wi-Fi, and battery icons, and the time 17:19. Below the status bar is a header with a hamburger menu icon and the text 'GESTURN'. The main content area is titled 'Registrar nuevo usuario'. It contains three input fields: 'Correo electronico' with a person icon, 'Contraseña' with a key icon, and 'Confirmar contraseña' with a key icon. Below these fields is a 'Regístrate' button. At the bottom, there is a link '¿Ya tienes una cuenta?' with an 'Iniciar sesión' button.

Pantalla de modificación de datos personales:

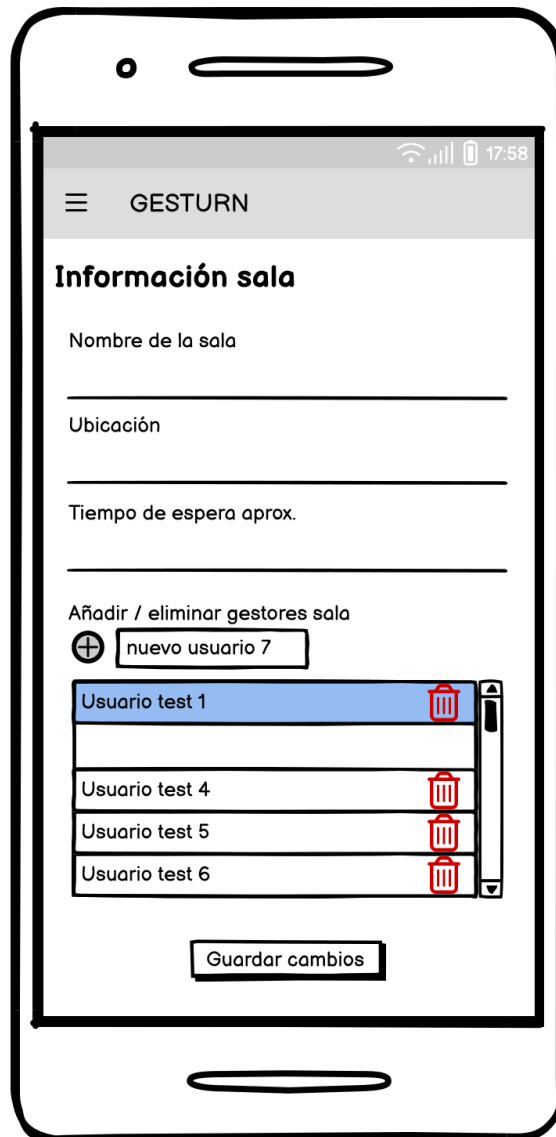


The image shows a mobile application interface for editing a user profile. At the top, there is a status bar with signal strength, battery, and the time 17:21. Below that is a header with a hamburger menu icon and the text 'GESTURN'. The main content area is titled 'Perfil' and contains several input fields: 'Nombre de usuario', 'Apellidos', 'Fecha de nacimiento' (with a calendar icon), 'Correo electrónico', 'Nueva contraseña', and 'Confirmar nueva contraseña'. At the bottom of the form is a button labeled 'Guardar cambios'.

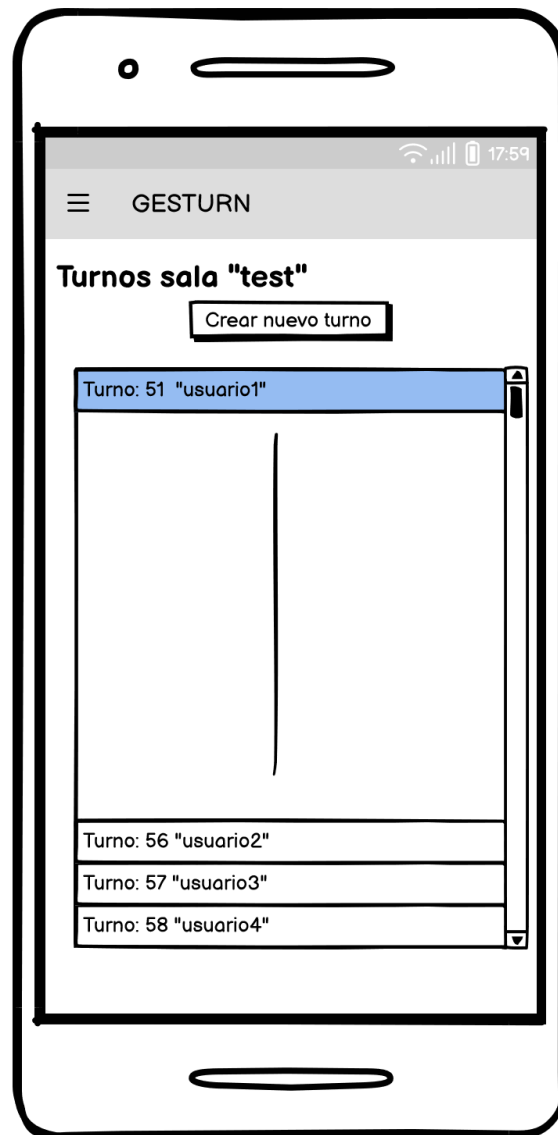
Pantalla de inicio una vez el usuario está identificado:



Pantalla de creación / edición sala de turnos:

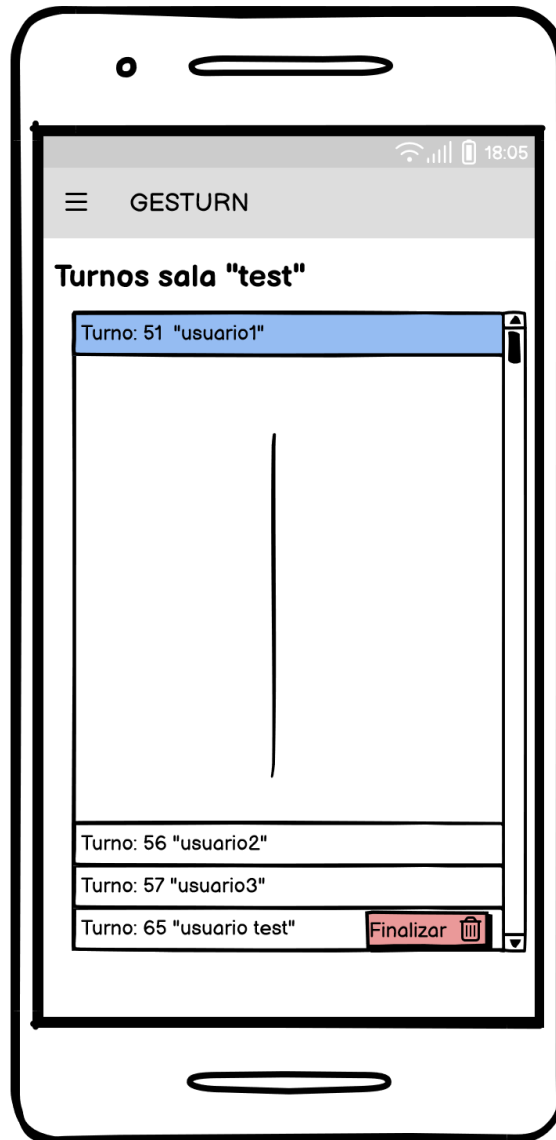


Pantalla de visualización turnos sala:



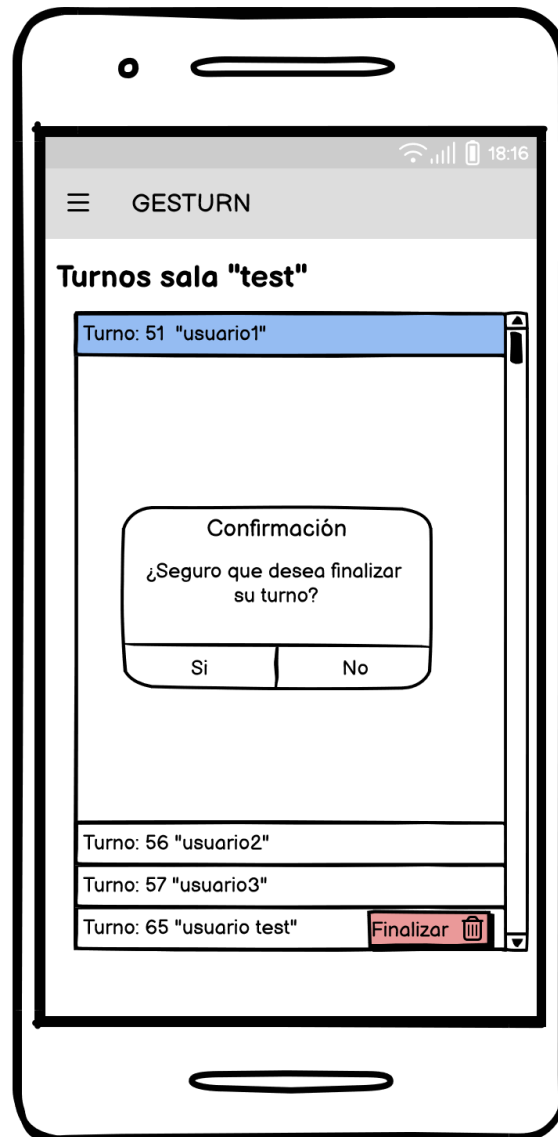
Siempre que el usuario no tenga un turno activo en esa sala aparecerá el botón de “Crear nuevo turno”.

Pantalla de visualización de turnos sala con turno ya creado:



El usuario podrá finalizar su turno pulsando en el botón de “Finalizar”.

Pantalla de confirmación de finalizar turno:



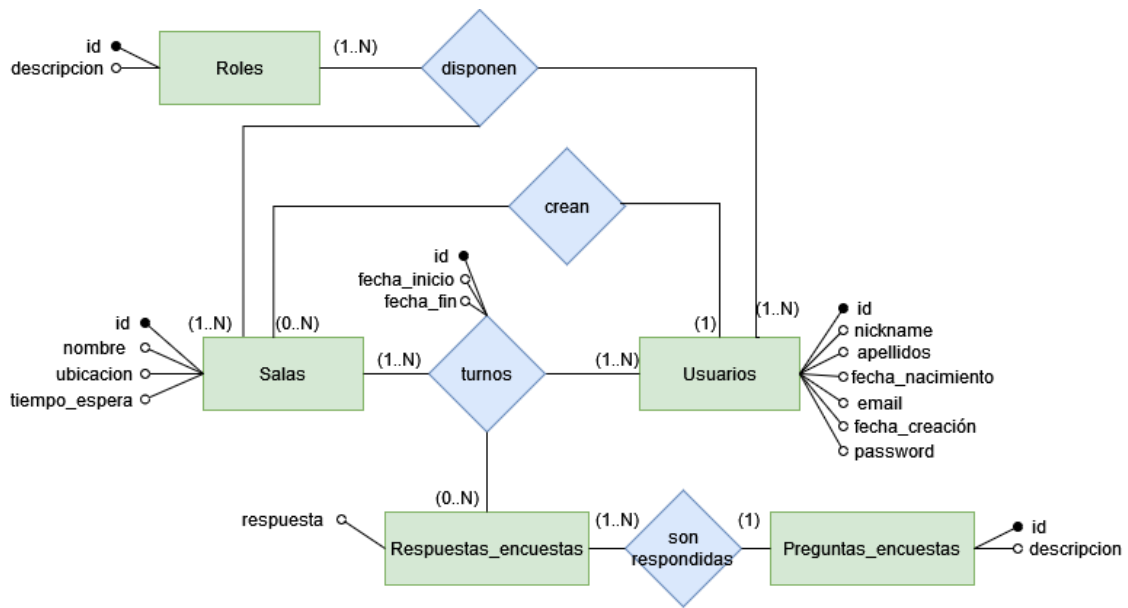
4.2 Diseño de base de datos

El diseño de la base de datos no es un proceso sencillo, la complejidad de la información y la cantidad de los requisitos del sistema pueden añadir dificultad. Por este motivo, vamos a dividir el proceso del diseño de la base de datos en tres fases: diseño de datos conceptual, diseño de datos lógico y por último diseño de datos físico.

4.2.1 Diseño de datos conceptual

El modelo de datos conceptual se utiliza para traducir los problemas del mundo real a un nivel muy alto, facilitando la comprensión de las entidades y así puede entenderse por cualquier persona.

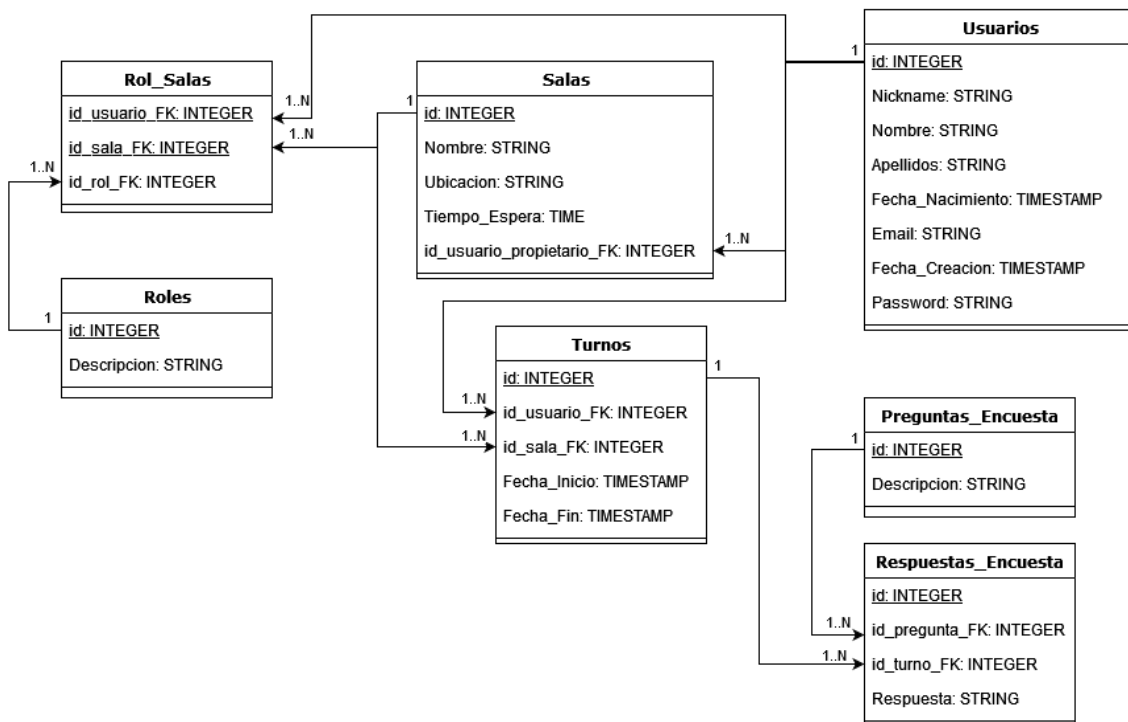
Para ello vamos a representar la estructura general de los datos requeridos en un modelo entidad-relación:



4.2.2 Diseño de datos lógico

El diseño lógico de los datos es el siguiente paso para construir realmente toda la arquitectura de la aplicación. A diferencia del modelo conceptual, este diseño utiliza índices y claves (primarias o externas) para representar la relación entre los datos. La representación del modelo lógico de los datos es independiente de cualquier implementación del DBMS.

A continuación, vamos a representar el modelo lógico de la base de datos mediante un esquema relacional:



La clave primaria de cada tabla es la propiedad que está subrayada. Como podemos observar todas las tablas utilizan de clave primaria la propiedad “id” menos la tabla “Rol_Salas”, tiene sentido que la clave primaria de esta tabla sea una clave conjunta de las propiedades “id_usuario_FK” y “id_sala_FK” ya que solo se puede tener un rol por usuario y sala.

La propiedad “Nombre” de la tabla “Salas” debe ser única, ya que interesa que no exista más de un registro con el mismo nombre.

4.2.3 Diseño de datos físico

El diseño físico de la base de datos es el proceso de traducción del modelo lógico abstracto a un diseño técnico específico para el DBMS seleccionado.

Este punto se deja pendiente hasta que la construcción del prototipo esté empezada. Con la tecnología que se va a utilizar se autogenerará el diseño físico de la base de datos utilizando como base las clases creadas en la programación del prototipo.

5. Implementación del prototipo



En este apartado se implementará el prototipo de la web cumpliendo con el análisis y el diseño realizado previamente.

5.1 Entorno de desarrollo


Para el desarrollo del prototipo se ha utilizado el IDE⁷ Microsoft Visual Studio Community 2022. Utilizando Git⁸ para el control de versiones del código fuente.

5.2 Dependencias


Las dependencias que se han utilizado para la implementación de la plataforma han sido las siguientes:

 Microsoft.EntityFrameworkCore.SqlServer por Microsoft	6.0.12
Microsoft SQL Server database provider for Entity Framework Core.	7.0.1
 Microsoft.EntityFrameworkCore.Tools por Microsoft	6.0.12
Entity Framework Core Tools for the NuGet Package Manager Console in Visual Studio.	7.0.1

Estas dos dependencias añaden compatibilidad con la base de datos Microsoft SQL Server, son las encargadas de comunicar la aplicación con la base de datos.

 MudBlazor por Garderober, Henon and Contributors	6.1.2
MudBlazor is an ambitious Material Design component framework for Blazor with an emphasis on ease of use and clear structure. It is perfect for .NET developers who want to rapidly build web applications without having to struggle with CSS and Javascript. MudBlazor, being written entirely in C#, empow...	6.1.7

MudBlazor⁹ es un potente *framework* que facilita y agiliza el desarrollo de la interfaz de la aplicación, ahorrándonos la elaboración de código HTML y CSS.

 QRCode por Raffael Herrmann	1.4.3
QRCode is a simple library, written in C#.NET, which enables you to create QR codes.	

QRCode¹⁰ es una conocida librería escrita en C# para facilitar la generación de códigos QR.

Las dependencias que vienen incluidas por defecto en un proyecto Blazor no se han especificado ya que son las mismas para todos los proyectos.

⁷ IDE: Entorno de desarrollo integrado (IDE) es una aplicación de software que ayuda a los programadores a desarrollar software de manera eficiente.

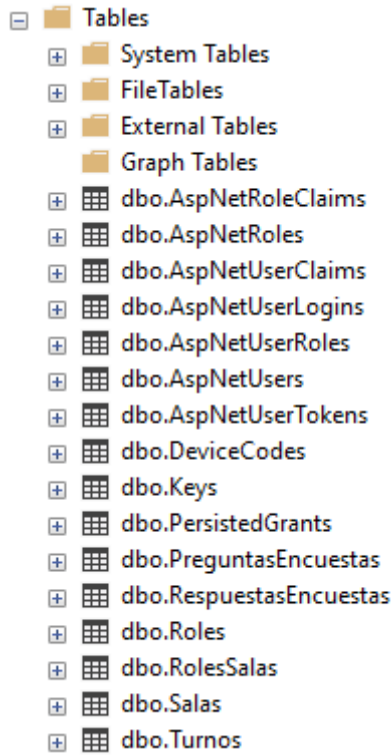
⁸ Sitio web: <https://www.atlassian.com/es/git/tutorials/what-is-git>

⁹ Sitio web: <https://mudblazor.com/>

¹⁰ Sitio web: <https://github.com/codebude/QRCode/>

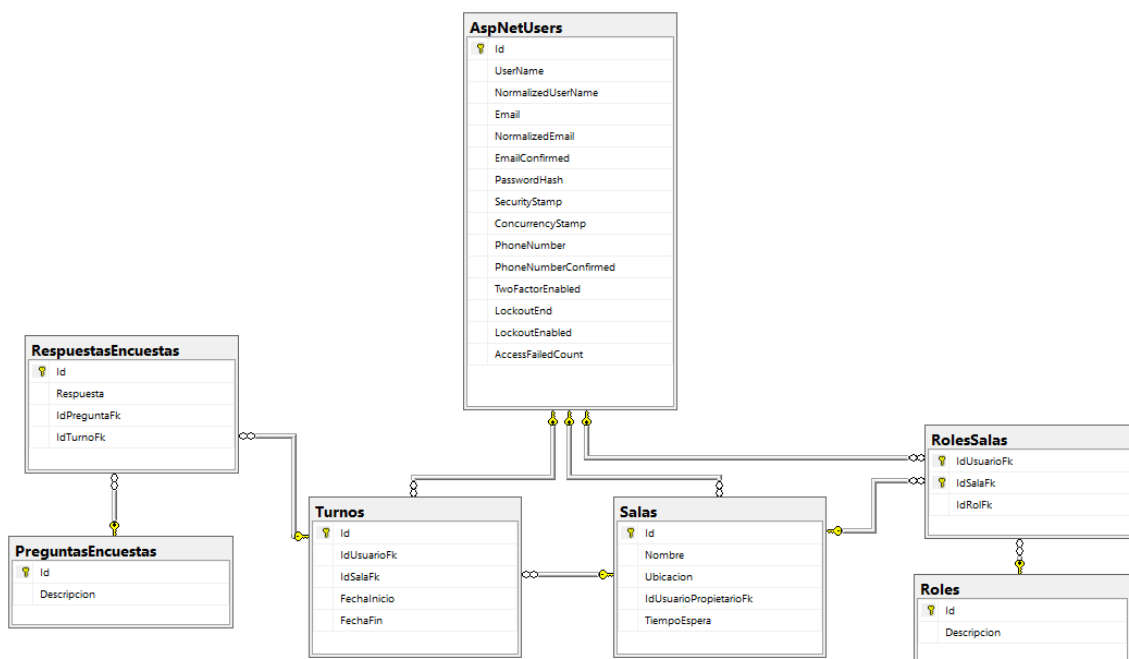
5.3 Base de datos

La construcción de la base de datos ha sido realizada fielmente a los requisitos y al diseño establecido. Al utilizar Blazor, este crea algunas tablas adicionales para la gestión de usuarios. A continuación, podemos observar todas las tablas creadas:



Todas las tablas que tienen el prefijo “AspNet” son las que se han creado de forma automática al utilizar Blazor, la única tabla que utilizamos y relacionamos con las otras tablas es “AspNetUsers”.

A continuación, podemos observar el diagrama de las relaciones:



Se ha utilizado *Entity Framework*¹¹ para la creación de las tablas necesarias para la aplicación, por lo que el desarrollo se ha centrado en la correcta creación de los modelos de datos en clases y sus relaciones. Una vez se han creado estas clases se han aplicado las migraciones correspondientes a la base de datos para su construcción.

5.4 Estructura del proyecto

El modelo de hospedaje que se ha elegido para el desarrollo ha sido **Blazor WebAssembly**, ofreciéndonos la capacidad de compartir código entre las aplicaciones cliente y servidor, integración con MVC¹² y *Razor Pages*¹³.

Este modelo de hospedaje nos ofrece varias ventajas:

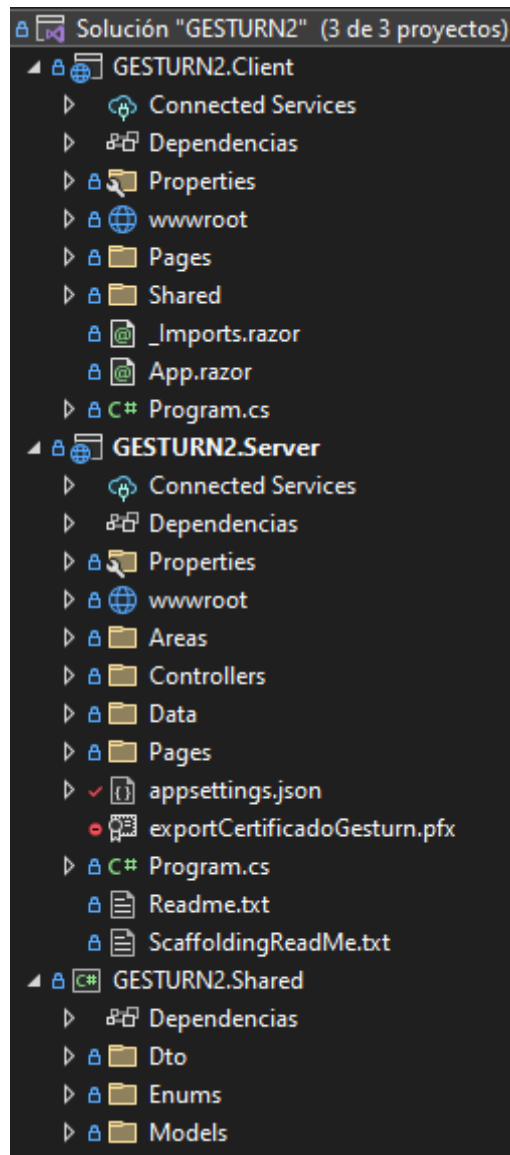
- El trabajo se descarga del servidor al cliente (la primera vez que se accede a la página puede ser más lenta ya que debe descargar los recursos).
- No hay ninguna dependencia del lado servidor de .NET por lo que la aplicación puede seguir funcionando si el servidor se queda sin conexión.
- Los recursos y capacidades del cliente se aprovechan completamente.

A continuación, podemos ver la organización de carpeta y archivos del proyecto:

¹¹ Sitio web: <https://learn.microsoft.com/es-es/dotnet/framework/data/adonet/ef/overview>

¹² Patrón de arquitectura de software Modelo-Vista-Controlador (MVC)

¹³ Razor es un lenguaje de plantillas basado en HTML y C# que se usa para generar dinámicamente la interfaz de usuario web



Podemos separar la aplicación en tres grupos principales: el lado del cliente, el lado del servidor y la parte común "shared".

En el cliente disponemos de la siguiente estructura:

- Directorio "wwwroot": es la carpeta raíz web de la aplicación que contiene los recursos estáticos públicos.
- Directorio "Pages" y "Shared": contienen los componentes o páginas enrutables que componen la aplicación. La ruta de cada página se especifica utilizando la directiva "@page".
- Archivo "_Imports.razor": Engloba las directivas de Razor comunes que se van a incluir en los componentes de la aplicación.
- Archivo "App.razor": Componente raíz que configura el enrutamiento del lado cliente.
- Archivo "Program.cs": es el punto de entrada de la aplicación que configura el host de WebAssembly.

En el servidor disponemos de la siguiente estructura:

- Directorio “Areas”: contiene las páginas que se van a utilizar para la creación y autenticación de los usuarios en la plataforma.
- Directorio “Controllers”: contiene los controladores de la aplicación, actuando como intermediarios entre los modelos y las vistas, gestionando y transformando los datos según las necesidades.
- Directorio “Data”: contiene todos los archivos de las migraciones que se han realizado en el transcurso del desarrollo. También contiene el archivo que utilizamos directamente para acceder a la base de datos (hereda de DbContext) y otra clase que utilizamos para la carga inicial de datos en el momento de crear la base de datos.
- Archivo “appsettings.json”: contiene
- Archivo “Program.cs”:

En la parte común “shared” disponemos de la siguiente estructura:

- Directorio “Dto”: este directorio contiene todas las clases DTO utilizadas para transferir datos desde la parte del servidor al cliente o viceversa.
- Directorio “Enums”: contiene las clases tipo “enum”.
- Directorio “Models”: contiene las clases que representan la base de datos.

5.5 Descripción de la interfaz (páginas)

En este punto se muestra el resultado final de cada una de las pantallas diseñadas en el aplicativo. Las capturas se han realizado a una resolución de 390x844 (tamaño aprox. de un teléfono de 6,7 pulgadas), ya que la web está pensada para funcionar sobre todo en dispositivos móviles.

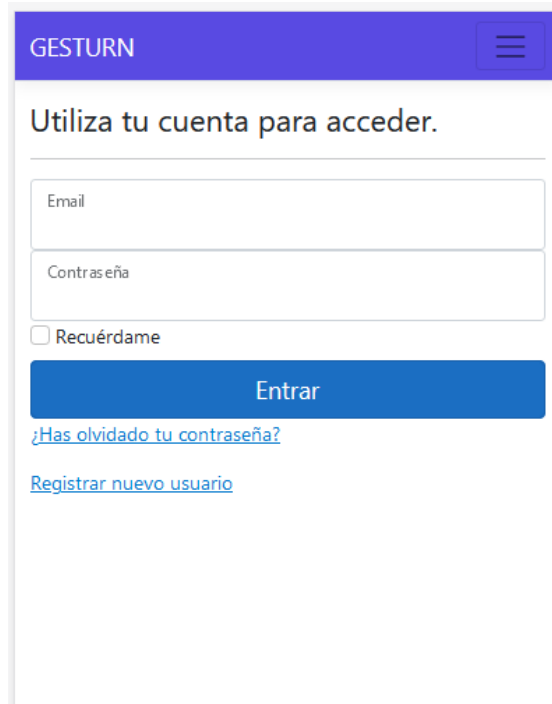
5.5.1 Pantalla de carga

Durante las cargas de la web se ha diseñado una pantalla específica para ello:



Como podemos observar en el centro se visualiza el nombre de la aplicación, justo debajo hay una barra de progreso y por último abajo a la derecha e incluido el logo de la universidad. Los colores del fondo de pantalla simulan tener movimiento moviéndose de un lado a otro.

5.5.2 Pantalla de identificación (login)



The screenshot shows the login interface of the GESTURN application. At the top, there is a purple header with the text 'GESTURN' on the left and a hamburger menu icon on the right. Below the header, the text 'Utiliza tu cuenta para acceder.' is displayed. There are two input fields: 'Email' and 'Contraseña'. Below these fields is a checkbox labeled 'Recuérdame'. A large blue button labeled 'Entrar' is positioned below the checkbox. Underneath the button, there are two links: '¿Has olvidado tu contraseña?' and 'Registrar nuevo usuario'.

La pantalla de identificación es sencilla, los usuarios deben utilizar la cuenta de correo electrónico que han utilizado y la contraseña. Si se pulsa en la opción de “Recuérdame”, las credenciales se guardarán aunque se cierre la página web o el navegador.

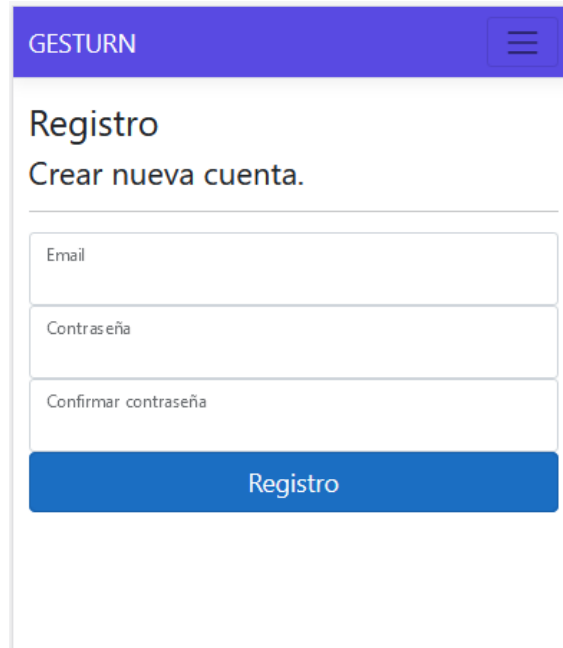
5.5.3 Menú Registro / Entrar



This screenshot is similar to the previous one, showing the login form. However, a purple menu overlay is visible at the top left, below the header. The menu contains two options: 'Registro' and 'Entrar'. The rest of the page content, including the input fields, checkbox, and buttons, remains the same as in the previous screenshot.

Si el usuario no está identificado en la aplicación, el menú que visualizará arriba a la derecha tendrá únicamente esas dos opciones: Registro y Entrar.

5.5.4 Pantalla de registro



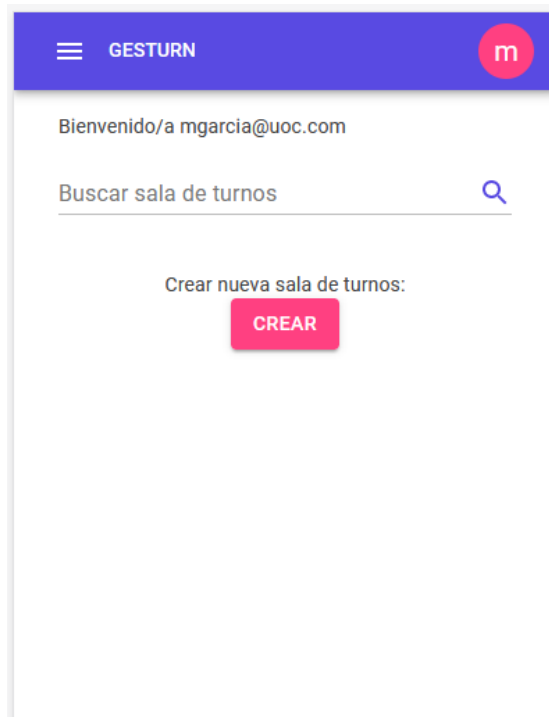
The screenshot shows a mobile application interface for registration. At the top, there is a purple header bar with the text 'GESTURN' on the left and a hamburger menu icon on the right. Below the header, the title 'Registro' is displayed in a large font, followed by the subtitle 'Crear nueva cuenta.' in a smaller font. The registration form consists of three input fields: 'Email', 'Contraseña', and 'Confirmar contraseña'. Each field is a simple white box with a thin grey border. At the bottom of the form is a prominent blue button with the white text 'Registro'.

Esta pantalla es la que debe rellenar el usuario si desea crear una nueva cuenta.

Si el correo electrónico ya está siendo usado para otra cuenta o si la contraseña no reúne los requisitos mínimos de seguridad la cuenta no se creará y la aplicación informará al usuario del problema.

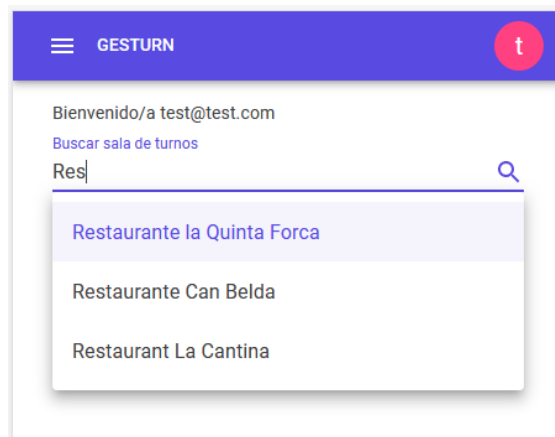
Para el desarrollo de este prototipo no se ha activado el envío de correos electrónicos por lo que el usuario no deberá confirmar ningún correo para confirmar la cuenta.

5.5.5 Pantalla inicial



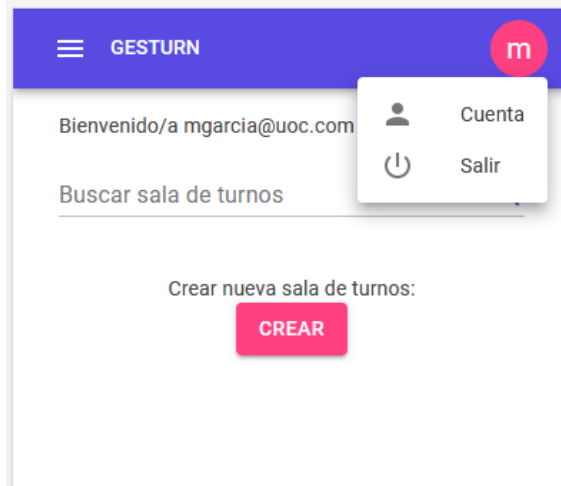
Una vez nos hemos identificado en el aplicativo, se nos redirigirá a la pantalla principal. Aquí podremos observar la cuenta con la que nos hemos identificado y a continuación un buscador autocompletable para buscar la sala del negocio que nos interese.

Para acceder a una sala debemos buscarla con el buscador y hacer clic en la que nos interesa:



Por último, tenemos un botón por si queremos crear nuestra propia sala.

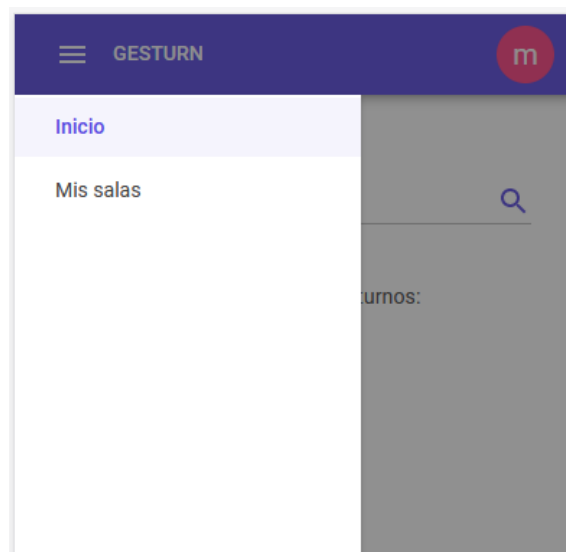
5.5.6 Menús del aplicativo



Una vez estamos identificados podremos observar la inicial de nuestra cuenta en la parte superior derecha, este menú contiene la opción de “Cuenta” y de “Salir”.

Si pulsamos en la primera opción podremos modificar la información de nuestra cuenta y la segunda opción nos borrará la sesión y redirigirá a la pantalla de identificación.

El otro menú que tiene la aplicación es pulsando en las tres rayas de la parte superior izquierda:



Este menú únicamente contiene la opción de “Inicio” que es donde nos encontramos actualmente y la opción de “Mis salas”.

5.5.7 Pantalla “Nueva sala”

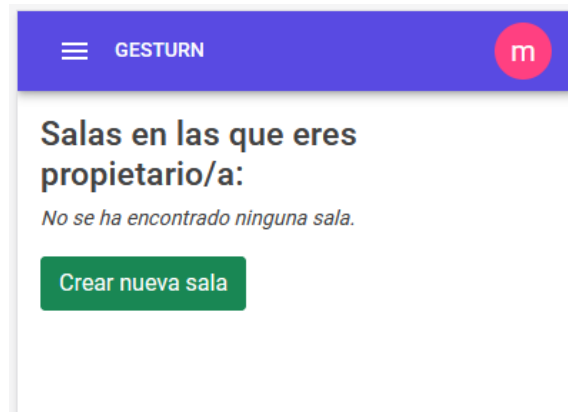


The screenshot shows the 'Nueva Sala' (New Room) form in the GESTURN app. The form is titled 'Nueva Sala' and has three input fields: 'Nombre sala' (Room name) with the value 'Restaurante la Quinta Forca', 'Ubicación' (Location) with the value 'Carrer de Santa Llogaia, 58, 17740, Girona', and 'Tiempo espera aprox. turno' (Approximate waiting time per shift) with the value '4'. Below the input fields are two buttons: 'Guardar' (Save) and 'Cancelar' (Cancel). The app's header is purple with the 'GESTURN' logo and a user profile icon.

En esta pantalla simplemente rellenamos la información de la sala con el nombre, la ubicación y el tiempo de espera aproximado de cada turno.

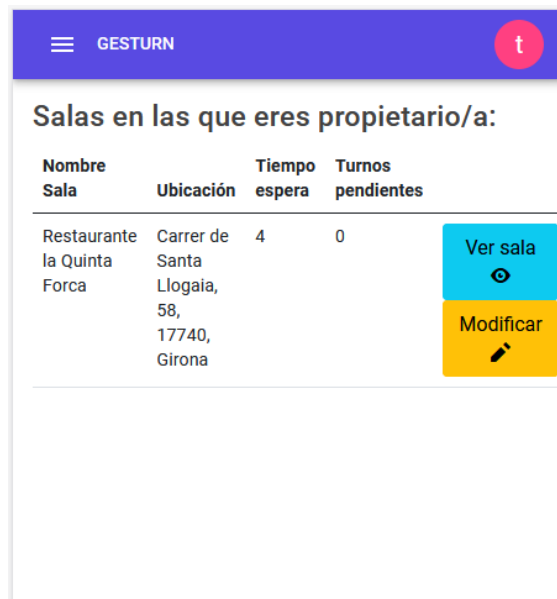
5.5.8 Pantalla “Mis salas”

Si el usuario con el que estamos identificados no es el propietario de ninguna sala visualizamos la pantalla de la siguiente manera:



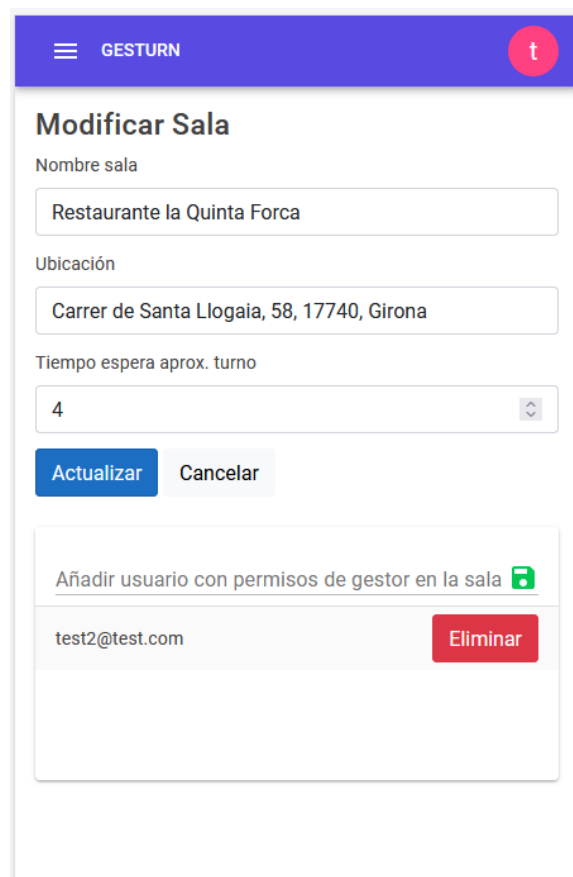
The screenshot shows the 'Mis salas' (My rooms) screen in the GESTURN app. The screen is titled 'Salas en las que eres propietario/a:' (Rooms in which you are the owner/a). Below the title, it says 'No se ha encontrado ninguna sala.' (No rooms were found). At the bottom, there is a green button labeled 'Crear nueva sala' (Create new room). The app's header is purple with the 'GESTURN' logo and a user profile icon.

En cambio, si el usuario tiene alguna sala creada visualizamos la información de estas:



Podemos ver en la tabla información correspondiente a cada sala, los turnos pendientes que hay en ese momento y dos botones para acceder directamente a la sala o modificar su información.

Si pulsamos sobre el botón de “Modificar”, accederemos a la misma pantalla que “Nueva sala” pero en este caso para modificar su información e incluso añadir a otros usuarios para que tengan permisos de gestores en esta sala:



Por ejemplo, en esta sala se ha añadido al usuario test2@test.com para que disponga de permisos adicionales para gestionar los turnos.

Para añadir un usuario simplemente hay que escribir el correo electrónico y pulsar sobre el botón verde de guardar.

Para eliminar los permisos de algún usuario simplemente hay que pulsar sobre el botón de “Eliminar” de dicho usuario.

5.5.9 Pantalla visualización de turnos en sala

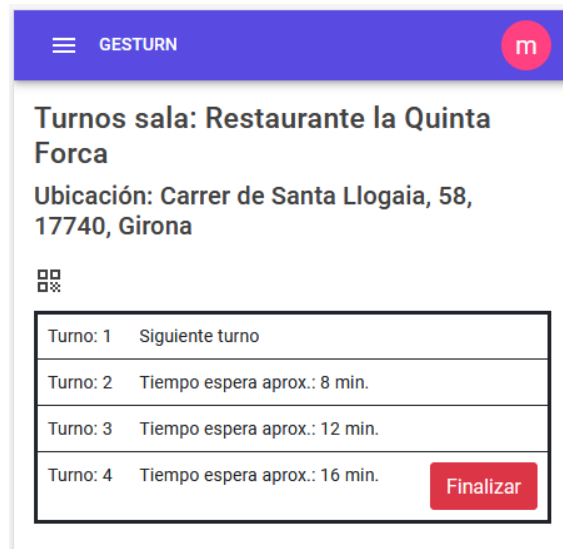


Esta pantalla es la que se utiliza para ver los turnos que hay actualmente pendientes a la espera de ser atendidos.

En la parte superior podemos ver el nombre de la sala, la ubicación y un botón para visualizar el código QR por si queremos compartir esta sala con otro usuario.

En la parte central visualizamos la lista de los turnos pendientes y el botón de “Nuevo turno” en el caso de que no tengamos ningún turno creado.

Si creamos un turno la pantalla pasará a visualizarse de la siguiente manera:



Como podemos ver el usuario únicamente puede finalizar su propio turno.

Los usuarios con permisos de gestor o los propietarios de la sala pueden finalizar cualquier turno de la sala.

Cuando nuestro turno sea el siguiente en ser atendido (sea el primero de la fila), **sonará una notificación** para que el usuario se percate de que ha llegado su turno.

Los turnos **se actualizan de forma automática** para que el usuario no tenga que actualizar la página web.

5.5.10 Pantalla visualización código QR sala

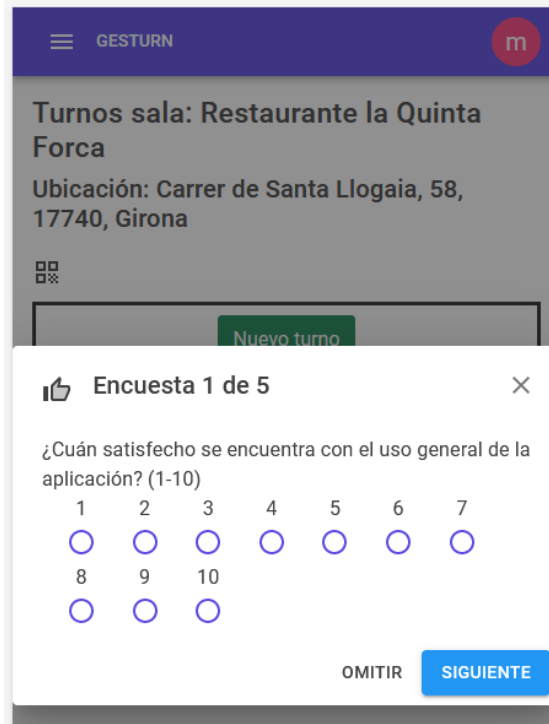


Al pulsar sobre el botón del código QR en la pantalla de visualización de turnos de la sala, se nos abrirá una ventana en la que podemos visualizarlo.

5.5.11 Pantallas de encuestas

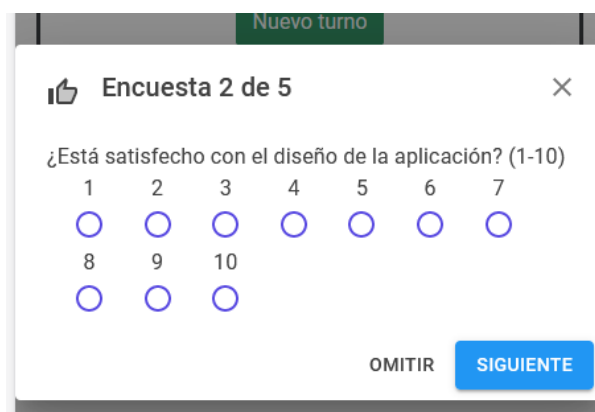
Una vez finalicemos nuestro turno se nos abrirá una pantalla en la que podremos contestar la encuesta, esta encuesta se puede omitir si el usuario lo desea.

Primera pregunta:



The screenshot shows the GESTURN app interface. At the top, there is a purple header with a menu icon, the text 'GESTURN', and a user profile icon 'm'. Below the header, the app displays the restaurant name 'Turnos sala: Restaurante la Quinta Forca' and its location 'Ubicación: Carrer de Santa Llogaia, 58, 17740, Girona'. A QR code icon is visible. A green button labeled 'Nuevo turno' is partially visible. Overlaid on this is a white survey modal titled 'Encuesta 1 de 5'. The question is '¿Cuán satisfecho se encuentra con el uso general de la aplicación? (1-10)'. There are ten radio buttons arranged in two rows of five, numbered 1 to 10. At the bottom of the modal are two buttons: 'OMITIR' and 'SIGUIENTE'.

Segunda pregunta:



The screenshot shows the same app interface as above, but with a different survey modal overlaid. The modal is titled 'Encuesta 2 de 5'. The question is '¿Está satisfecho con el diseño de la aplicación? (1-10)'. It features ten radio buttons numbered 1 to 10, arranged in two rows of five. At the bottom, there are 'OMITIR' and 'SIGUIENTE' buttons.

Tercera pregunta:

Nuevo turno

Encuesta 3 de 5

¿El proceso de crear/visualizar su turno le resultó fácil? (1-10)

1 2 3 4 5 6 7

8 9 10

OMITIR **SIGUIENTE**

Cuarta pregunta:

Encuesta 4 de 5

¿Cuán probable es que recomiende la aplicación a otros lugares? (1-10)

1 2 3 4 5 6 7

8 9 10

OMITIR **SIGUIENTE**

Quinta y última pregunta:

Encuesta 5 de 5

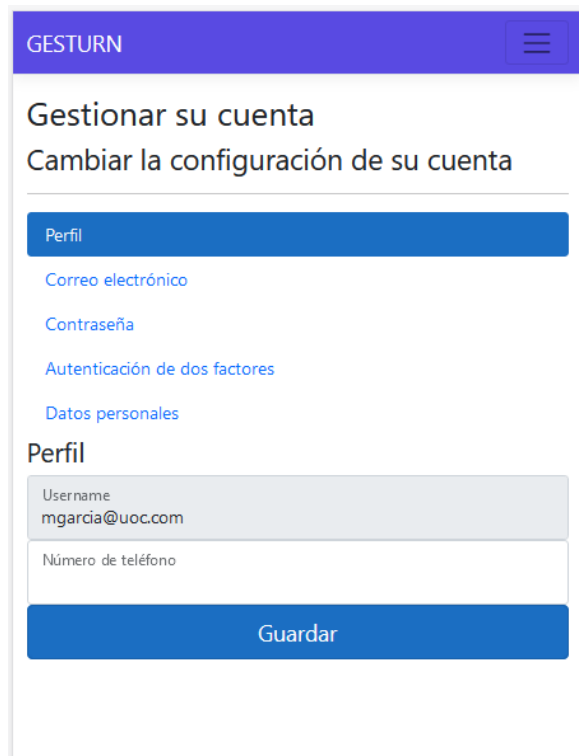
¿Tiene alguna sugerencia para mejorar la aplicación?

Observaciones

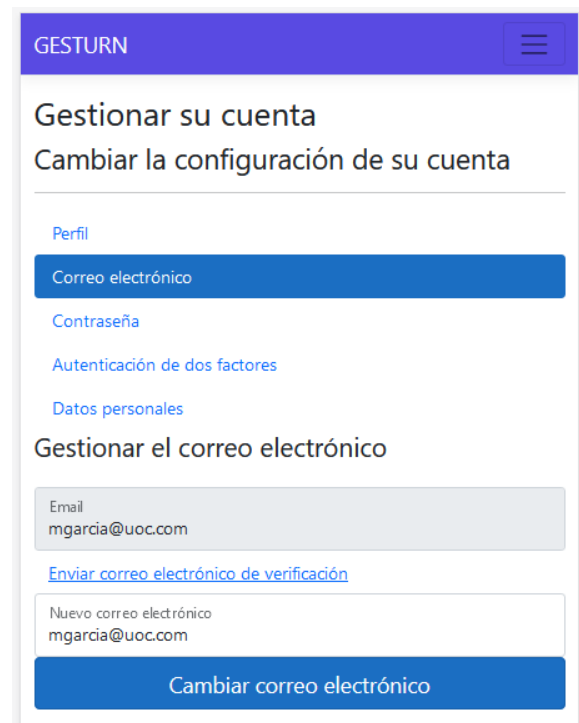
OMITIR **FINALIZAR**

5.5.12 Pantallas de gestión de la cuenta

Pulsando la opción de “Cuenta” del menú accederemos a los datos de nuestra cuenta:



Opción de modificar correo electrónico:



Opción de modificar contraseña:

La opción de “Datos personales” nos da opción a borrar definitivamente nuestra cuenta.

5.6 Pruebas

Recuperamos los requisitos funcionales y no funcionales establecidos en el análisis del proyecto e indicamos si se ha cumplido con éxito o no el requisito:

Código	Descripción	Resultado
RF1	Los usuarios deben registrarse y autenticarse para utilizar el sistema.	Correcto
RF2	Los usuarios solo podrán tener un turno activo en cada sala.	Correcto
RF3	Los gestores podrán crear nuevos turnos en las salas que dispongan de permiso.	Correcto
RF4	Los gestores podrán finalizar turnos de usuarios de la sala que gestionan.	Correcto
RF5	Los propietarios de la sala poseen los mismos permisos que los gestores, pero además pueden invitar a otros usuarios a ser gestores de sus salas.	Correcto
RF6	Los propietarios pueden revocar los permisos de los gestores.	Correcto
RF7	Los usuarios propietarios pueden crear más de una sala.	Correcto
RF8	El sistema ofrecerá la posibilidad de contestar una encuesta a los usuarios que finalicen su turno.	Correcto
RF9	El sistema creará un código QR para facilitar unirse a cada sala.	Correcto
RF10	Los usuarios deben poder visualizar los turnos pendientes de la sala previos al suyo.	Correcto

Código	Descripción	Resultado
RNF1	La interfaz debe ser intuitiva.	Correcto
RNF2	El sistema debe ser fácil de utilizar.	Correcto
RNF3	El sistema debe ser rápido y dar una respuesta casi inmediata.	Correcto
RNF4	Debe ser responsiva y compatible con la mayoría de los dispositivos.	Correcto
RNF5	Los permisos de una cola solo pueden ser editados por su propietario.	Correcto
RNF6	El sistema debe implementar protocolos de comunicación seguros.	Correcto
RNF7	El sistema estará disponible en castellano.	Correcto
RNF8	Se utilizará una base de datos relacional.	Correcto
RNF9	El sistema debe mostrar mensajes orientativos o informativos para guiar al usuario.	Correcto

5.6.1 Conclusiones

El RF1 funciona correctamente. Una mejora sería habilitar el envío de correo para confirmar que la dirección es correcta.

El RF2 está implementado tal como se planteó inicialmente. En una etapa avanzada del desarrollo surgió la idea de que los clientes solo pudieran tener un turno activo en todas las salas, ya que es lo más cercano a la realidad.

El RNF1 se ha intentado realizar de la forma más simple posible, la intuición es subjetiva y puede variar de persona a persona por lo que una persona puede verlo fácil e intuitivo, pero quizás otra persona no tanto.

El RNF3 dependerá, entre otras cosas, de la conexión a internet que disponga el dispositivo a la hora de consultar la web y también de la potencia del servidor que aloje la web.

5.7 Proyectos derivados

Algunos de los proyectos derivados que han surgido serían los siguientes:

- Añadir autenticación de doble factor y añadir la posibilidad de entrar con la cuenta de Gmail.
- Añadir traducciones en diversos idiomas para que el alcance de la plataforma sea mayor.
- Añadir una mayor personalización de las salas, permitiendo a los propietarios realizar sus propias encuestas para los clientes de sus salas. También sería interesante poder personalizar el numerador del turno de la sala, permitiendo elegir su longitud e incluso algún prefijo (p.e. "A-0001").
- Añadir en la pantalla principal sugerencias al usuario de las salas cercanas utilizando la geolocalización.

- Implementar un proceso en que la aplicación auto-calcule el tiempo aproximado de cada turno según el tiempo medio de los turnos que se han cerrado anteriormente en esa sala.
- No permitir que todos los usuarios tengan la libertad de crear nuevas salas y restringir esa posibilidad para usuarios “premium”.

5.8 Observaciones

En la carpeta compartida de Google Drive se ha publicado el archivo “.ova” de la máquina virtual utilizada para la elaboración de este documento, enlace al archivo: <https://drive.google.com/file/d/1kH0pv425iTlcrAeH3WrKnFLq-4JIK3bP/view?usp=sharing>

De igual forma, se ha publicado el paquete de la publicación de la solución:

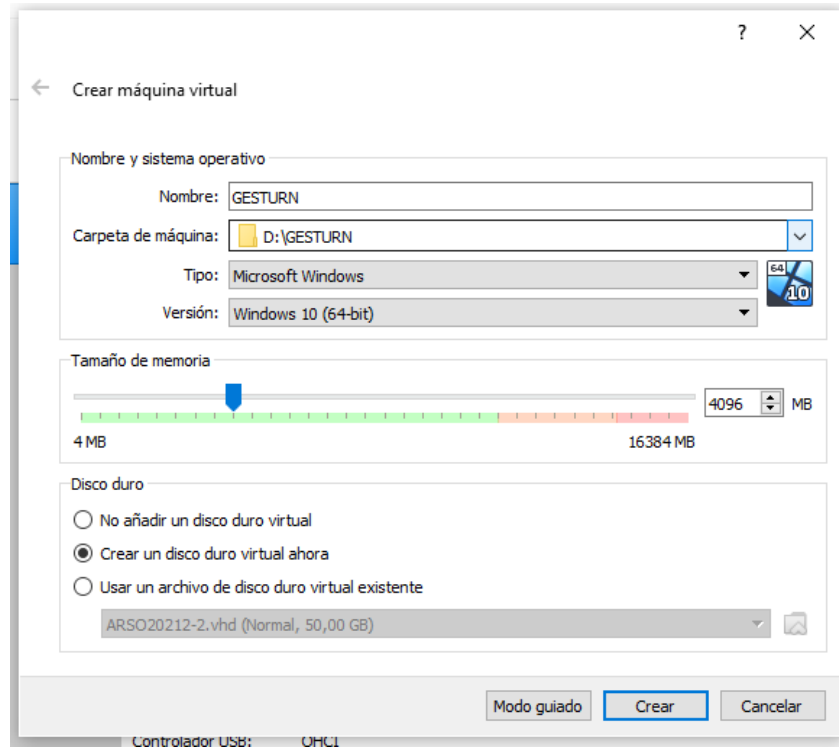
https://drive.google.com/file/d/1DsTBVX2Rx71NkPYPnGySKtrE39IAymGH/view?usp=share_link

6. Implantación del prototipo

6.1 Creación de la infraestructura

6.1.1 Creación de la máquina virtual

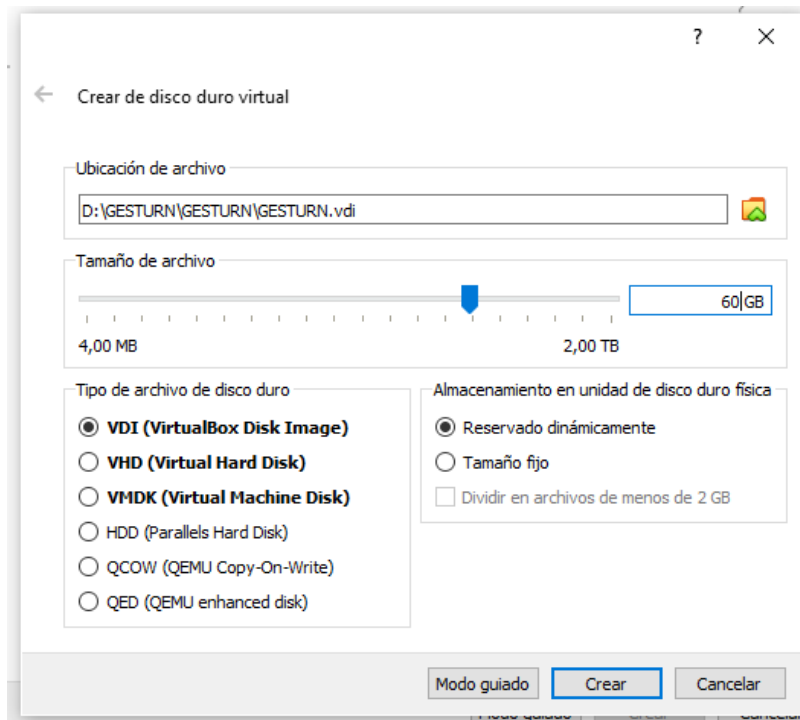
Para la creación de la máquina virtual vamos a utilizar el programa VirtualBox¹⁴. Una vez tenemos instalado el software podemos proceder a la creación de la máquina:



En nuestro caso vamos a utilizar Microsoft Windows como sistema operativo, le asignamos 4 Gb de memoria RAM¹⁵ y seleccionamos la opción de “Crear un disco duro virtual ahora”.

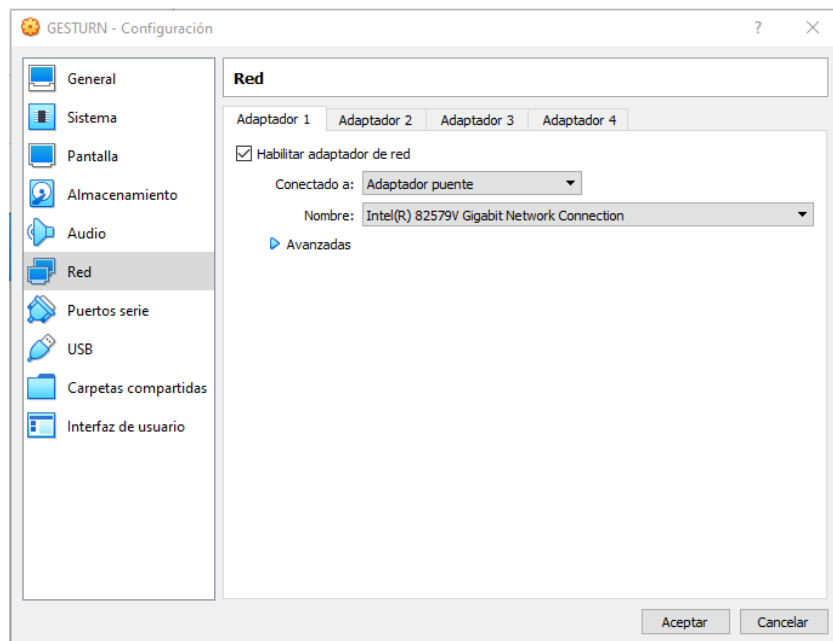
¹⁴ VirtualBox: <https://www.virtualbox.org/wiki/Downloads>

¹⁵ Memoria de acceso aleatorio (*Random Access Memory, RAM*)



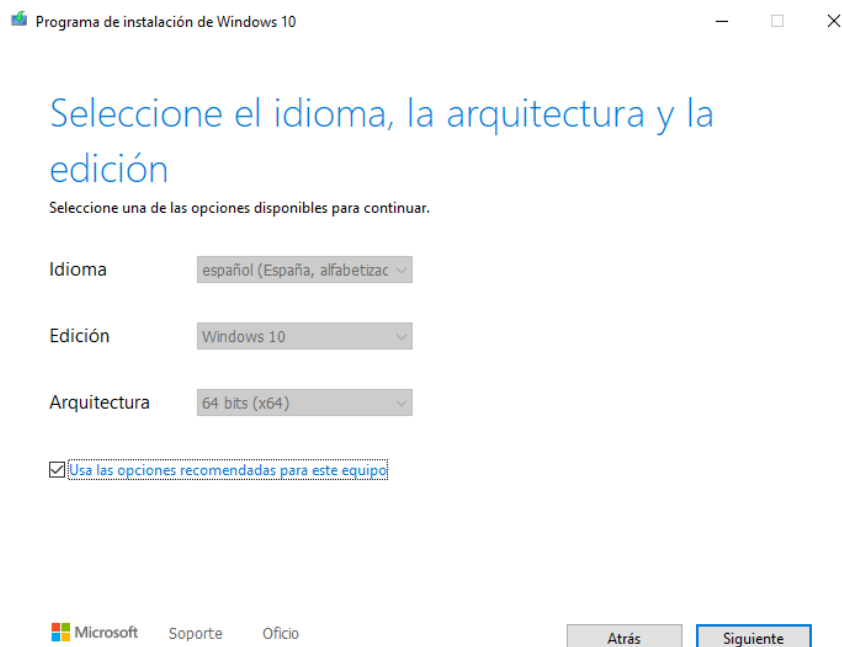
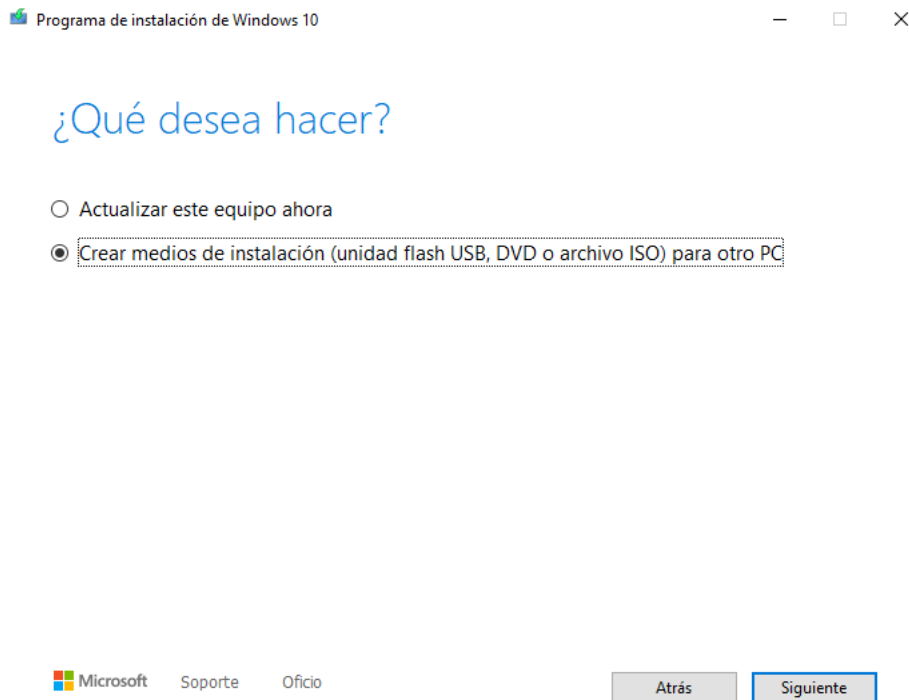
Elegimos el tamaño máximo del disco duro y la opción de “Reservado dinámicamente”, para que el archivo que contiene la máquina virtual vaya creciendo a medida que vamos requiriendo de más espacio en disco duro.

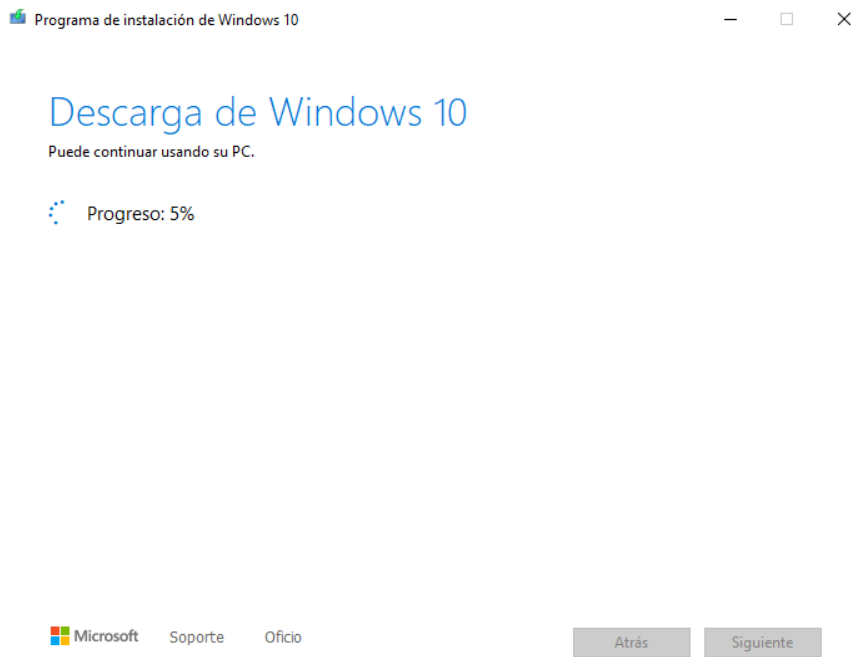
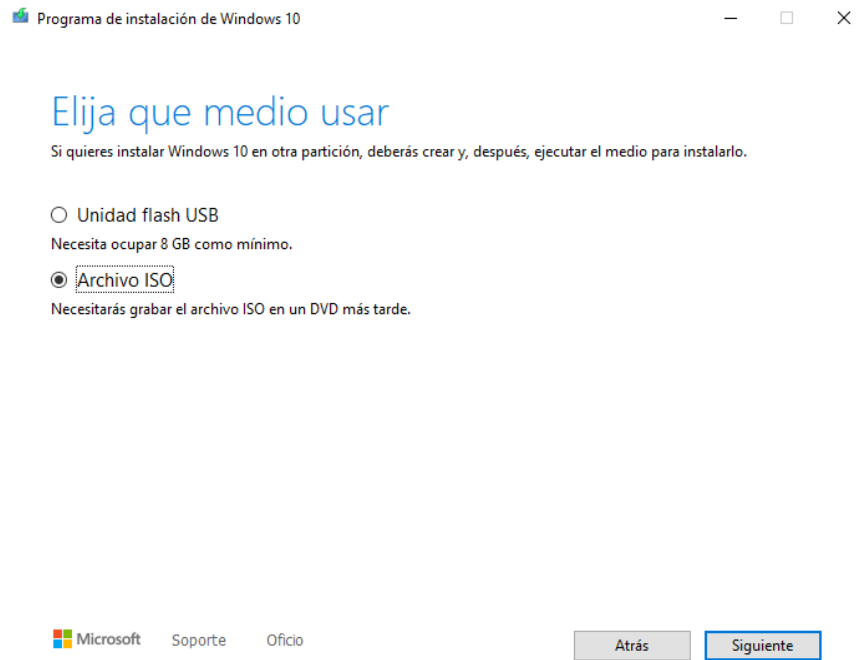
Una vez creada vamos a la configuración para modificar el adaptador de red de la siguiente manera:



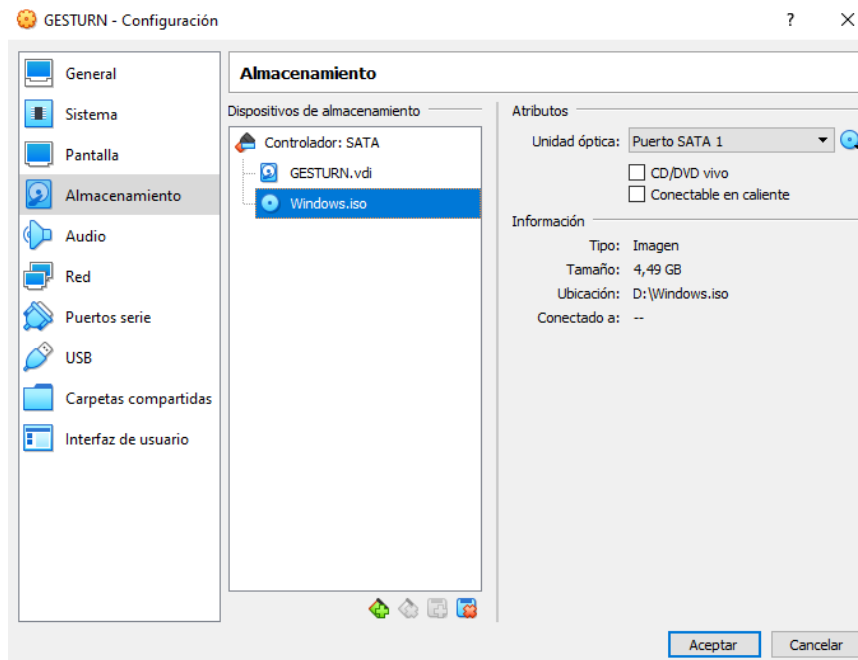
6.1.2 Instalación del sistema operativo

Podemos descargar la ISO de Windows 10 directamente desde su página oficial (<https://www.microsoft.com/es-es/software-download/windows10>)

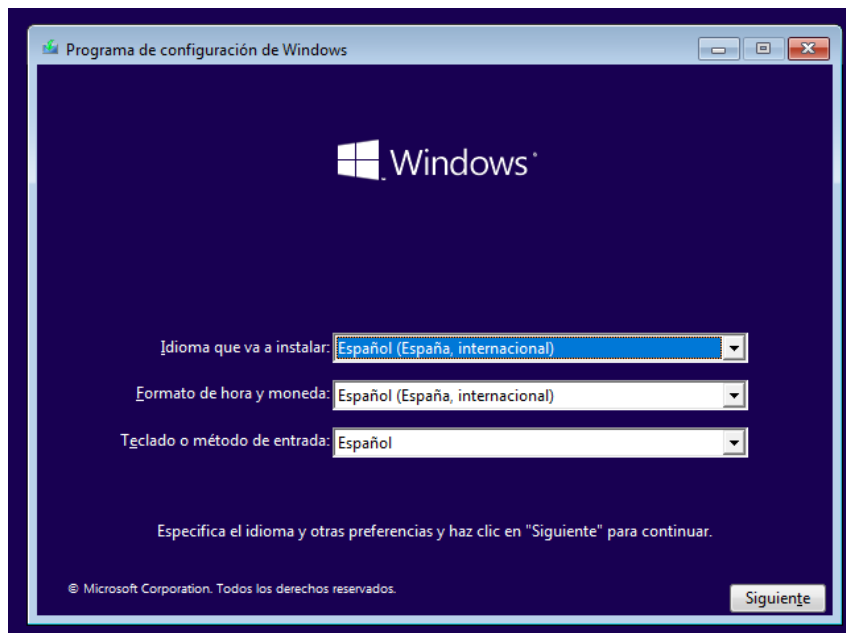


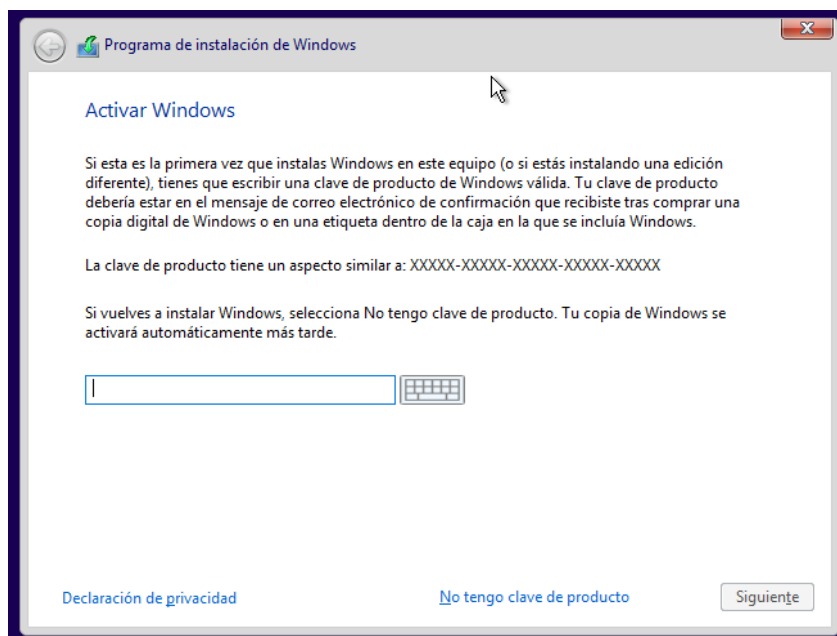
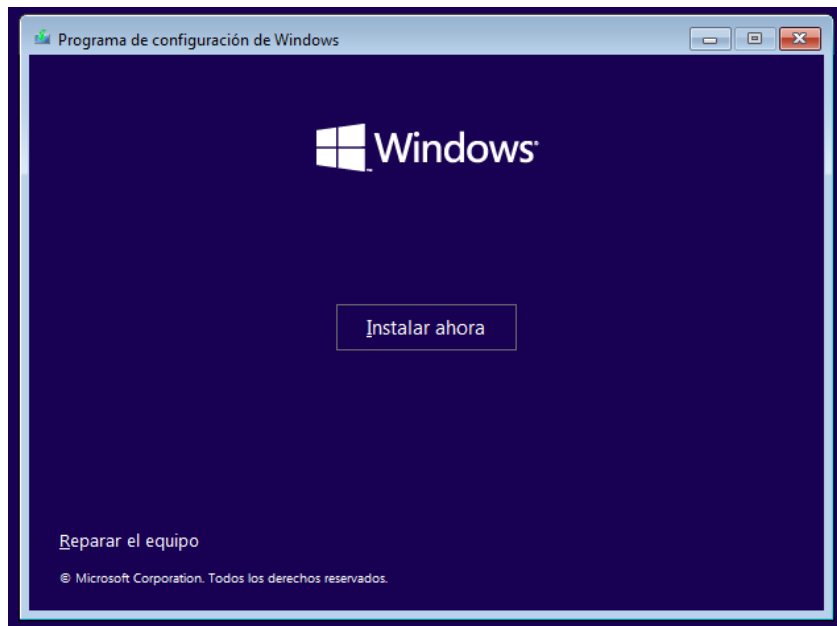


Una vez obtenida la ISO de Windows 10 procedemos a configurar la máquina virtual para que arranque con dicha ISO:



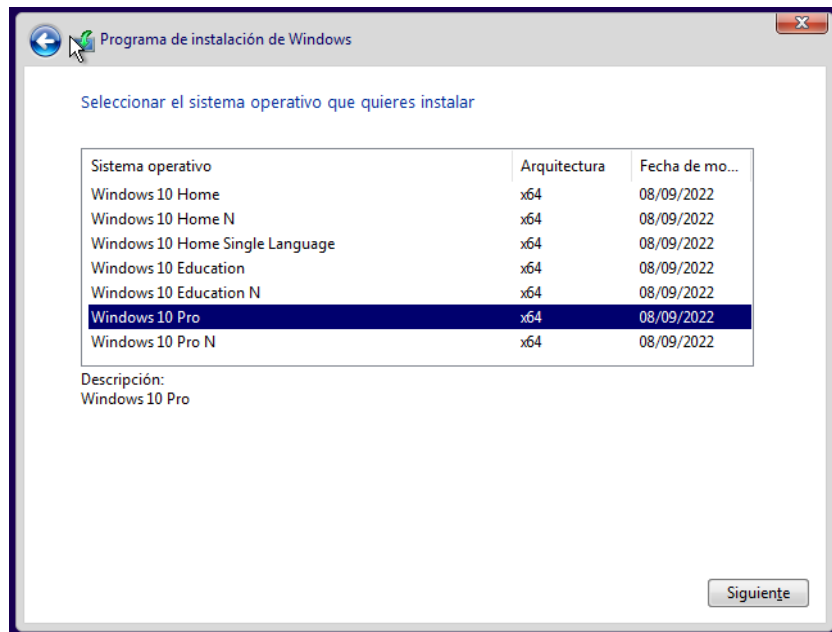
Iniciamos la máquina virtual y configuramos Windows 10:



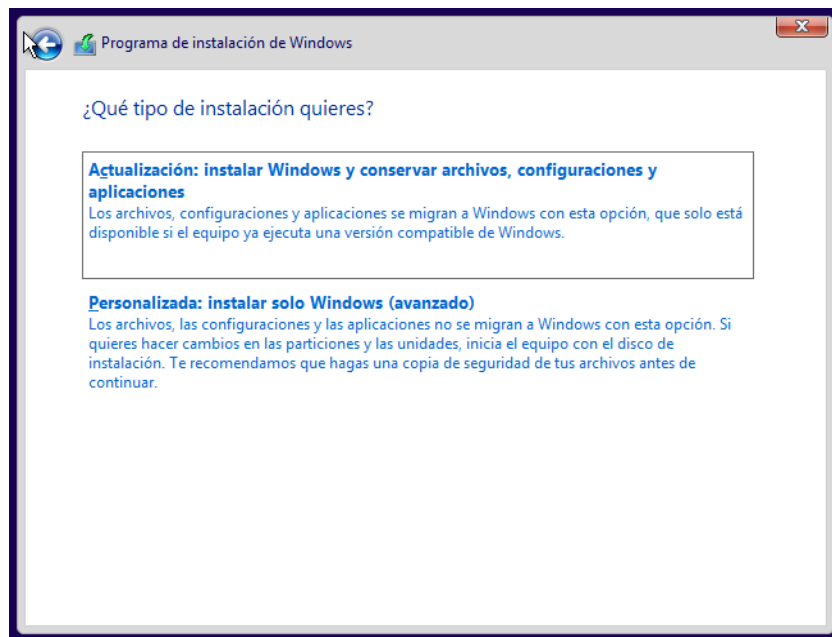


En la pantalla de activación de Windows pulsamos en “No tengo clave de producto”.

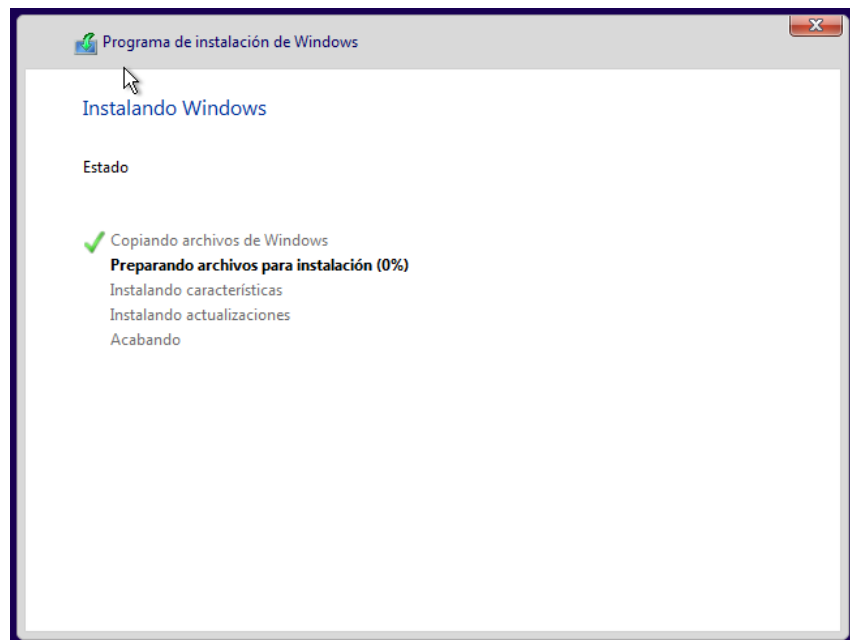
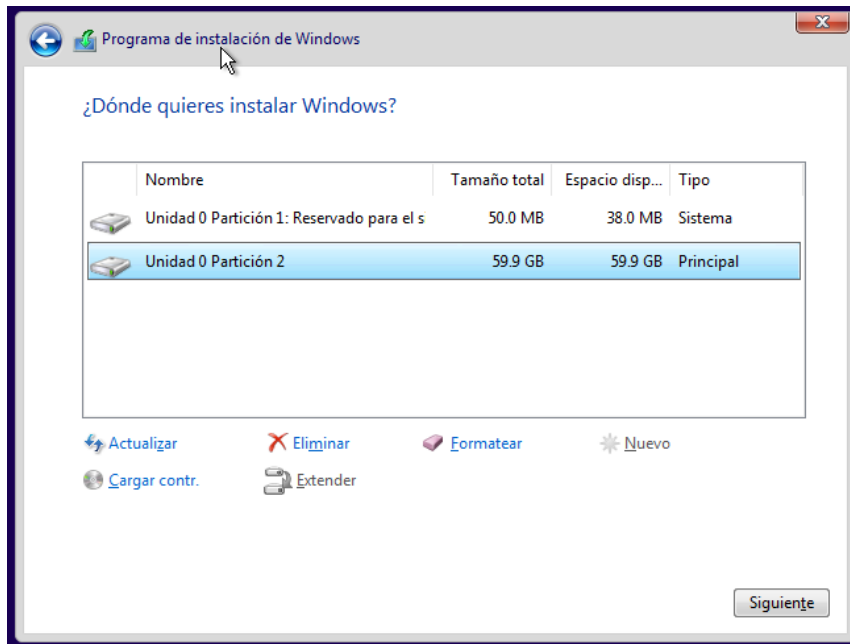
Elegimos el tipo Windows 10 Pro:



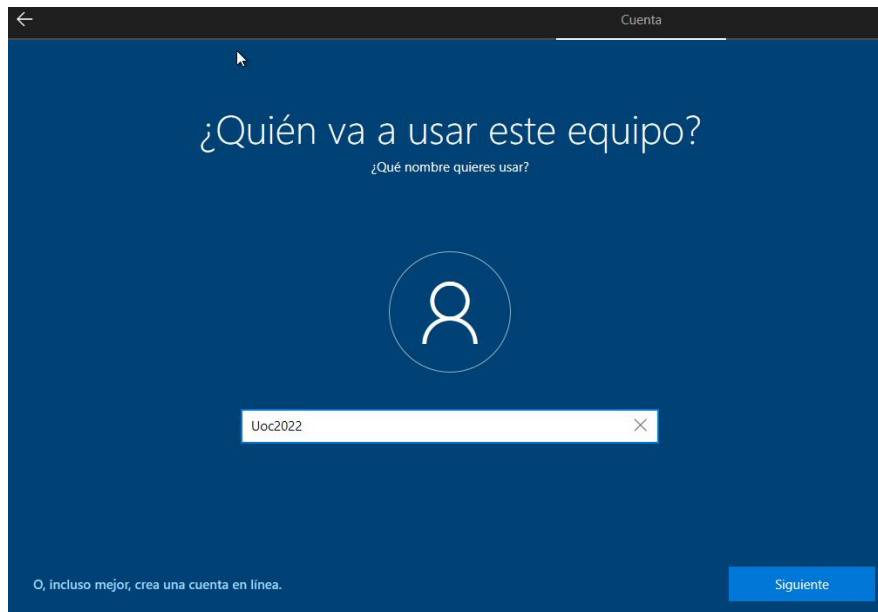
En el tipo de instalación elegimos Personalizada.



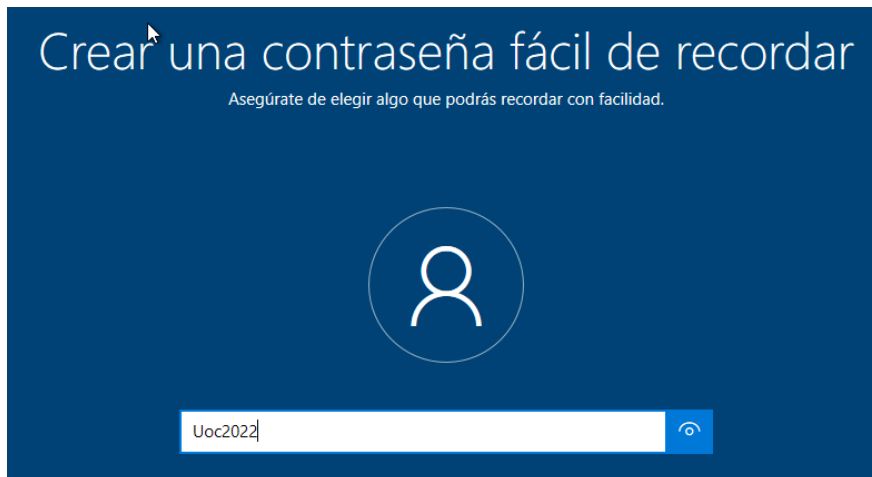
A continuación, debemos asignar el disco duro en el que queremos instalar el sistema operativo:



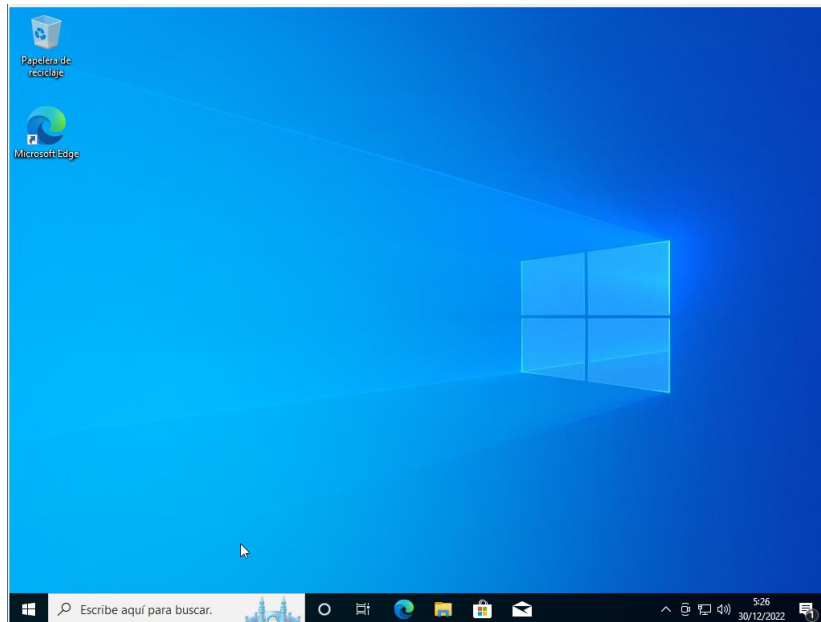
Una vez finaliza la instalación elegimos la región, la distribución de teclado y el nombre del usuario del equipo:



La contraseña que utilizamos es **Uoc2022**



La configuración tardará unos minutos más en finalizar y ya podremos visualizar el escritorio de nuestra máquina virtual:

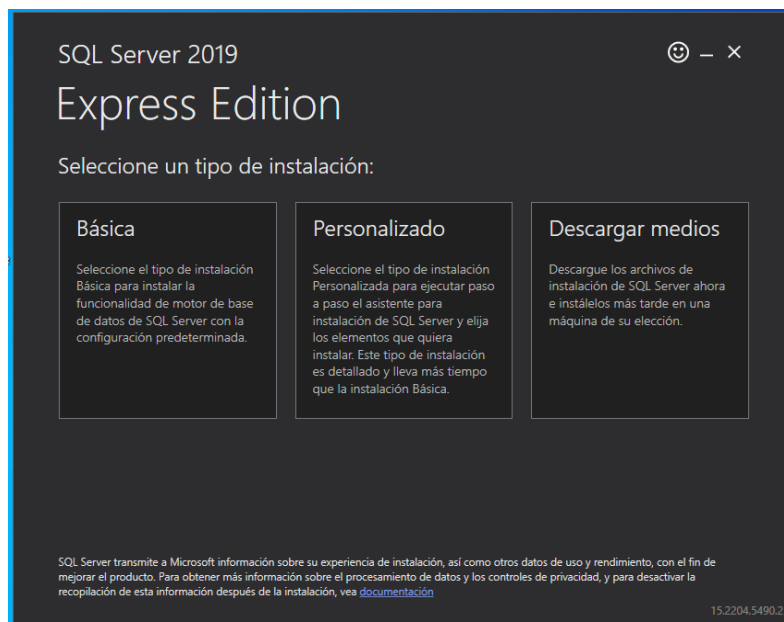


6.1.3 Instalación de los requisitos

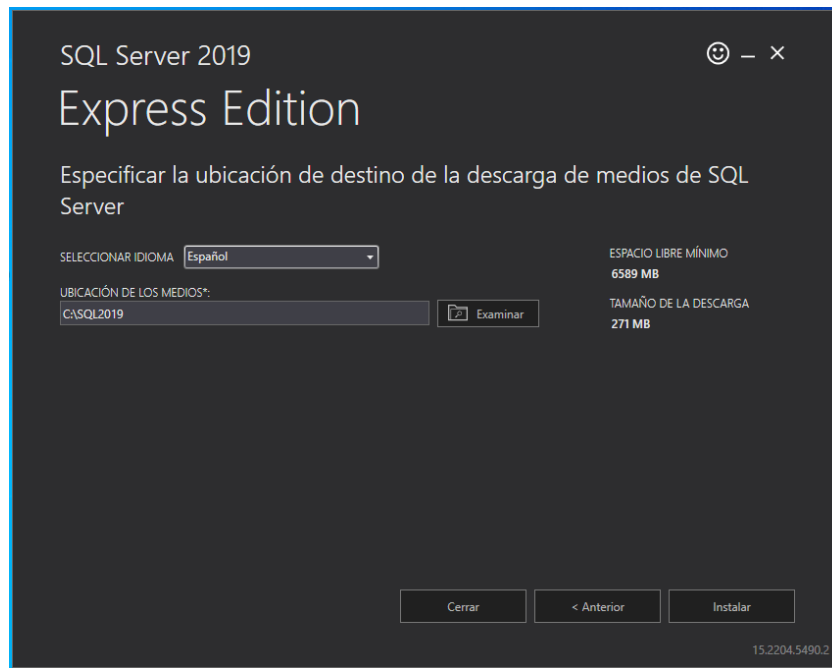
6.1.3.1 Instalación de la base de datos

Se ha decidido utilizar el motor de base de datos Microsoft SQL Server Express, para descargarlo, accedemos a la página web oficial (<https://www.microsoft.com/es-es/sql-server/sql-server-downloads>).

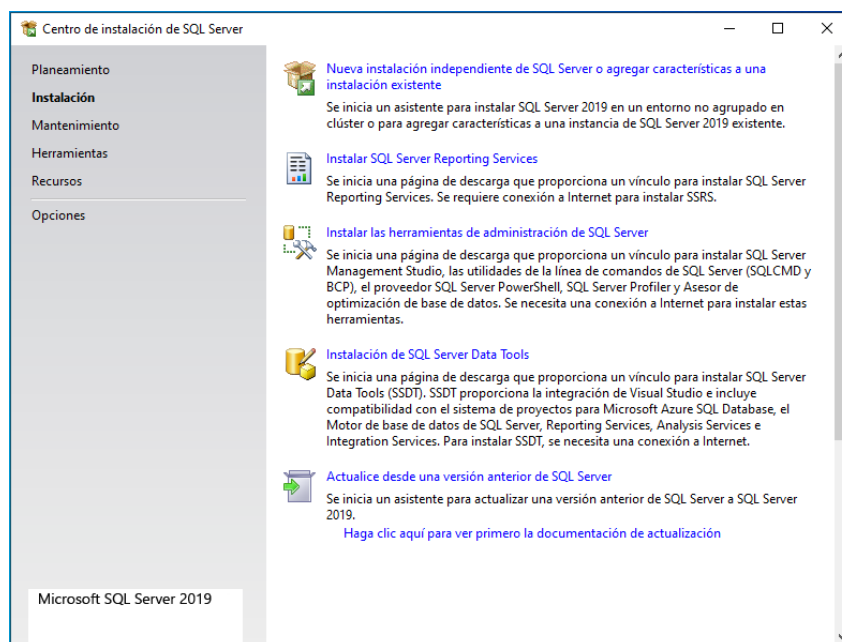
Una vez descargado ejecutamos el archivo y elegimos la instalación personalizada:

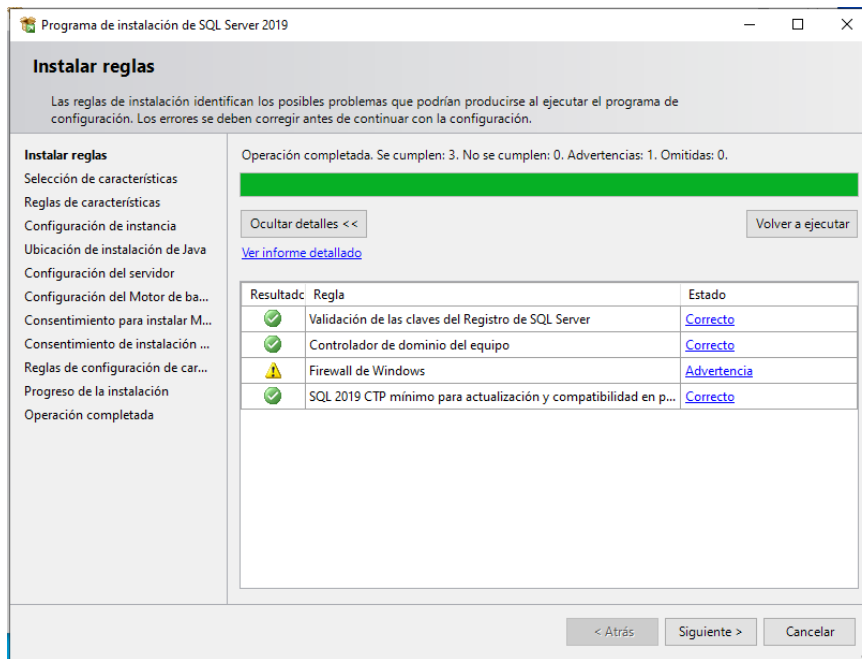


Elegimos la ruta de instalación y pulsamos Instalar:

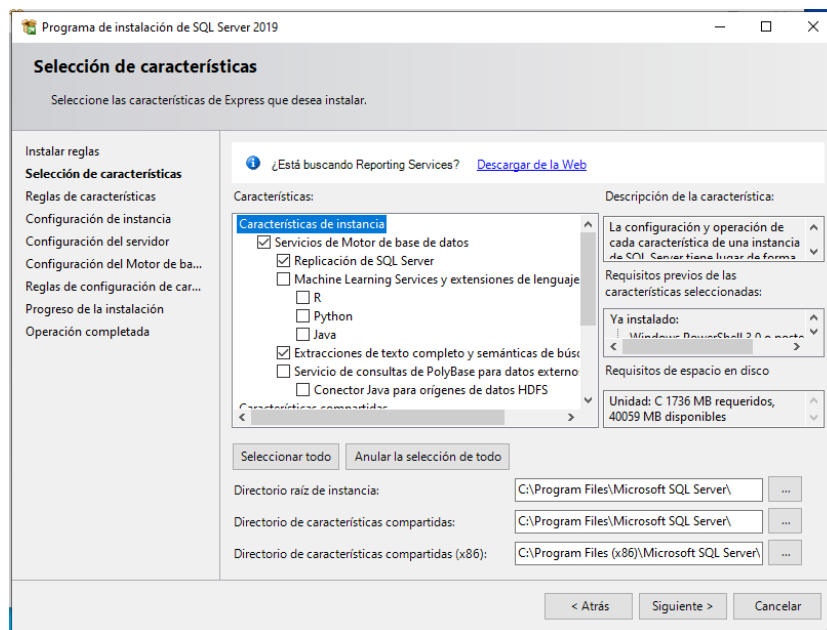


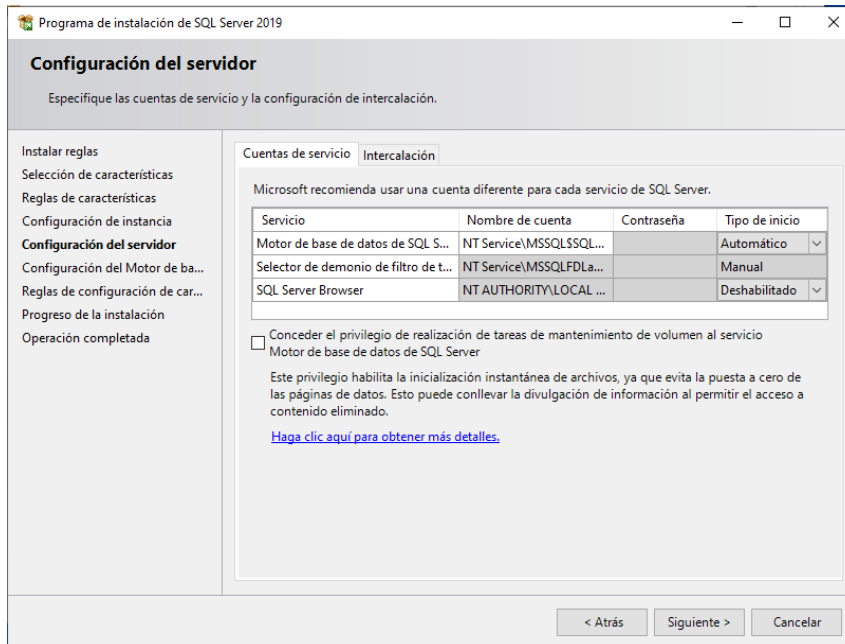
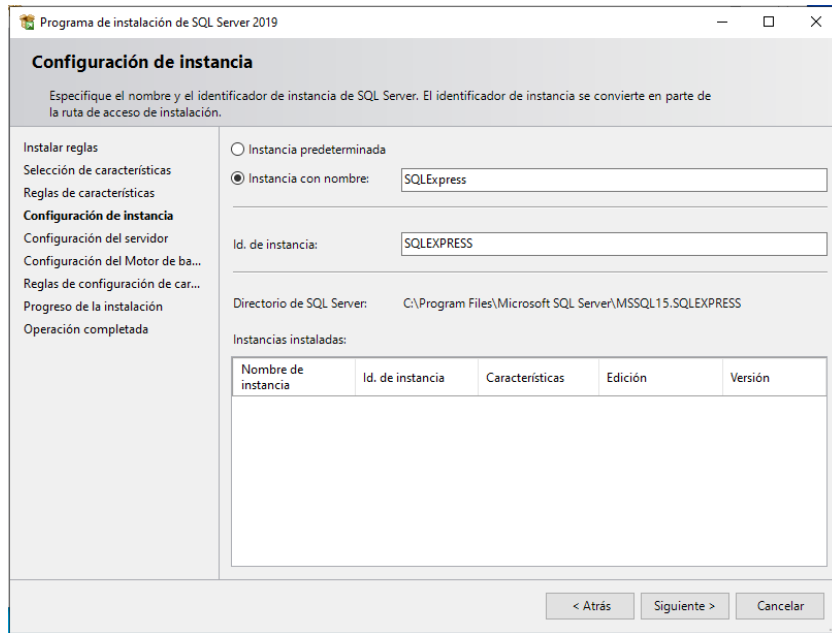
Pulsamos en “Nueva instalación independiente de SQL Server”:



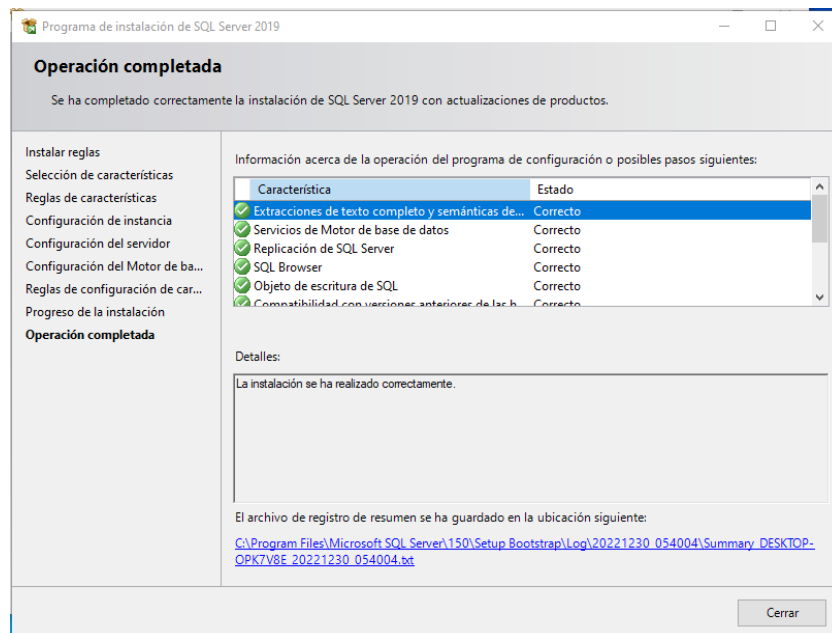
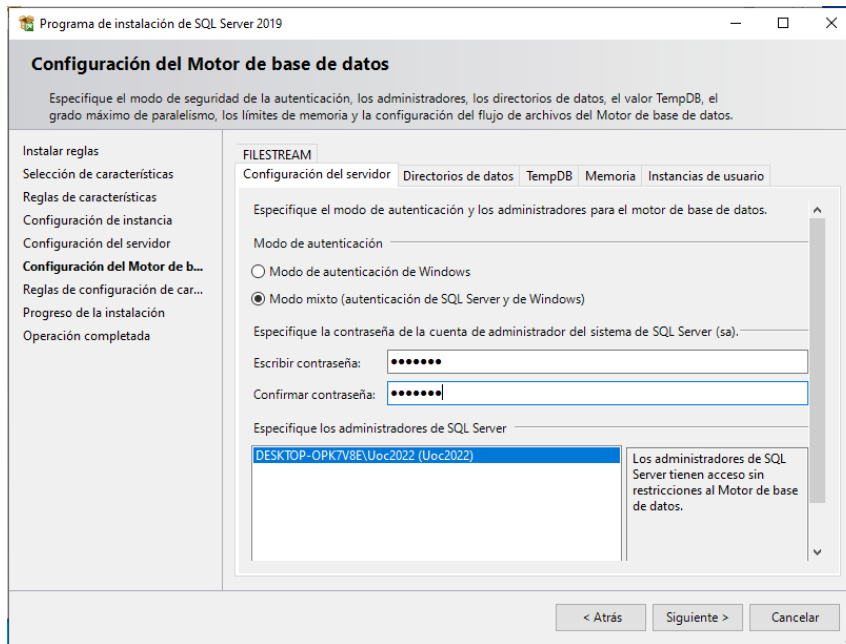


Desmarcamos la opción de “Machine Learning Services”, no es necesario y también ahorramos espacio en disco duro:





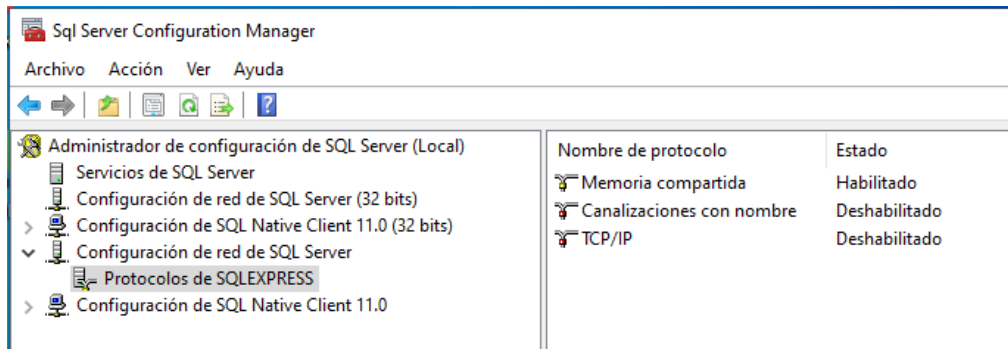
Elegimos la opción de Modo mixto e introducimos una contraseña, en este caso utilizamos “Uoc2022”.



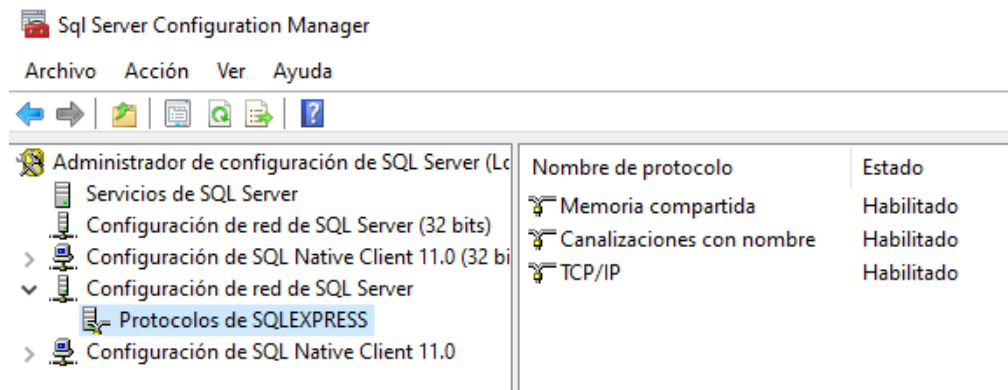
Opcionalmente podemos instalar también SQL Server Management Studio (SSMS), para comprobar fácilmente la creación de la base de datos y poder realizar consultas directamente. Para ello accedemos a la página oficial ([Descarga de SQL Server Management Studio \(SSMS\) - SQL Server Management Studio \(SSMS\) | Microsoft Learn](#)).

6.1.3.1.1 Configuración de SQL Server Express

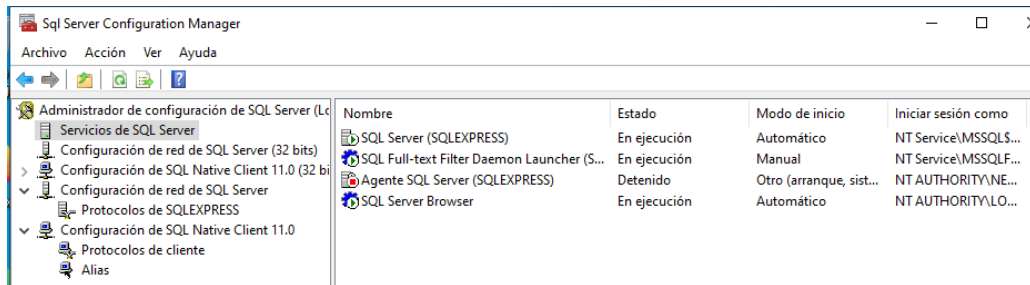
Para permitir el acceso de nuestra aplicación a la base de datos debemos habilitar el protocolo correspondiente, para ello abrimos el “Sql Server Configuration Manager” y habilitamos la “Canalización con nombre y por TCP/IP”:



Debe quedar de la siguiente manera:



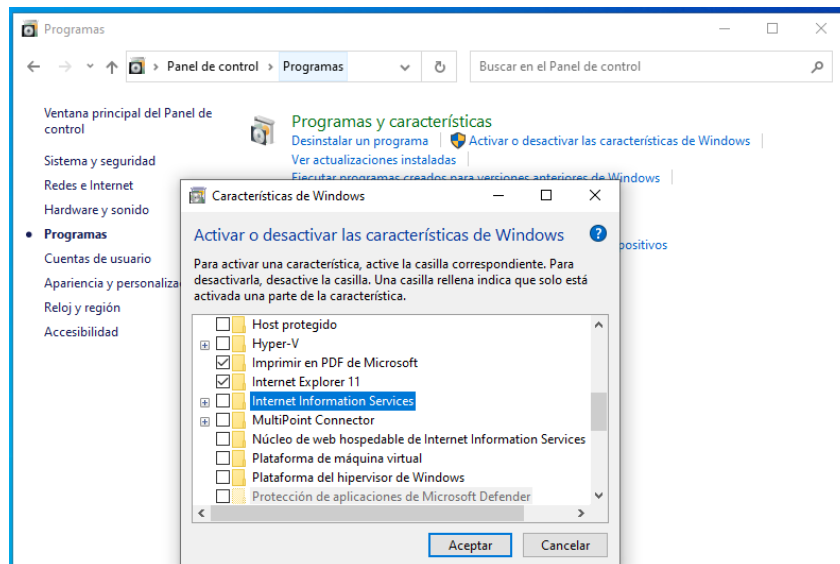
Por último, debemos habilitar también el SQL Server Browser:



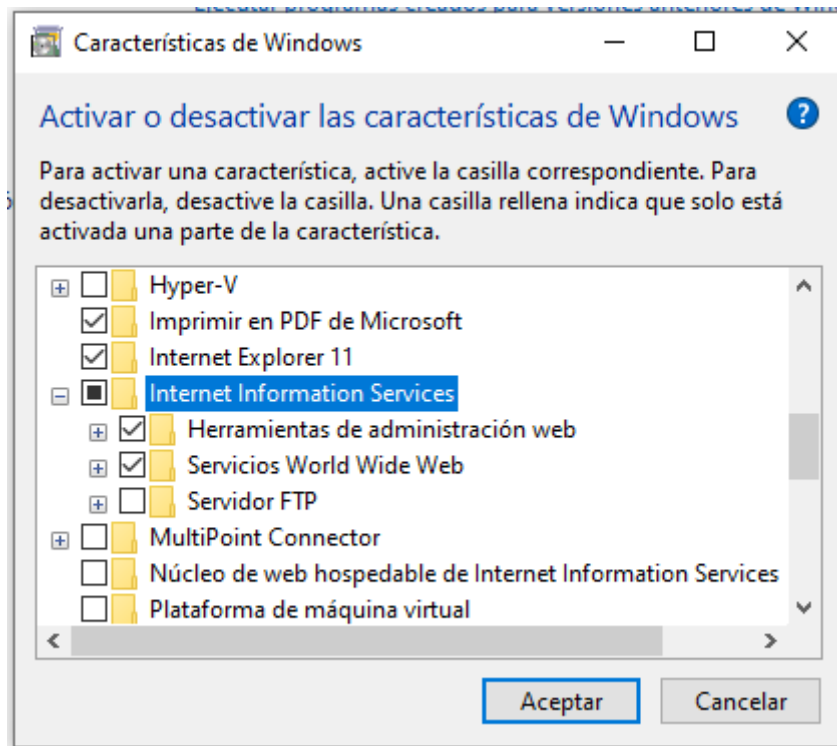
Realizando esta configuración permitimos que la aplicación pueda establecer conexión con la base de datos.

6.1.3.2 Instalación del Internet Information Services (IIS)

Para configurar el IIS debemos abrir el Panel de control -> Activar o desactivar las características de Windows:



Debemos marcar las opciones para que finalmente quede como en la siguiente figura:



6.1.3.3 Instalación de ASP CORE RUNTIME

Para poder ejecutar la aplicación es necesario tener instalado el “ASP NetCore Runtime”.

También necesitaremos el “Hosting Bundle” para que la web funcione correctamente en IIS, podemos descargarlo en el siguiente enlace: <https://dotnet.microsoft.com/en-us/download/dotnet/6.0>

Run apps - Runtime ⓘ

ASP.NET Core Runtime 6.0.0

The ASP.NET Core Runtime enables you to run existing web/server applications. **On Windows, we recommend installing the Hosting Bundle, which includes the .NET Runtime and IIS support.**

IIS runtime support (ASP.NET Core Module v2)
16.0.21299.0

OS	Installers	Binaries
Linux	Package manager instructions	Arm32 Arm32 Alpine Arm64 Arm64 Alpine x64 x64 Alpine
macOS		Arm64 x64
Windows	Hosting Bundle x64 x86 winget instructions	Arm64 x64 x86

Es muy importante escoger bien la versión, debe ser específicamente la de la figura anterior (v. 16.0.21299). Otras versiones pueden causar incompatibilidades y provocar errores en el funcionamiento.

6.2 Configuración de la aplicación

Debemos configurar la web para que se comuniquen con la base de datos que hemos configurado, para eso debemos acceder al directorio raíz de la aplicación y modificar el archivo "appsettings.production". El contenido de este archivo es, por defecto, el siguiente:

```
{
  "ConnectionStrings": {
    "DefaultConnection":
    "Server=127.0.0.1\\SQLEXPRESS;Database=GESTURN_BBDD;MultipleActiveResultSets=true;
    User Id=sa;Password=Uoc2022;"
  },
  "Logging": {
    "LogLevel": {
      "Default": "Information",
      "Microsoft.AspNetCore": "Warning"
    }
  },
  "IdentityServer": {
    "Key": {
      "Type": "File",
```

```

    "FilePath": "exportCertificadoGesturn.pfx",
    "Password": "password"
  },
  "Clients": {
    "GESTURN2.Client": {
      "Profile": "IdentityServerSPA"
    }
  }
},
"AllowedHosts": "*"
}

```

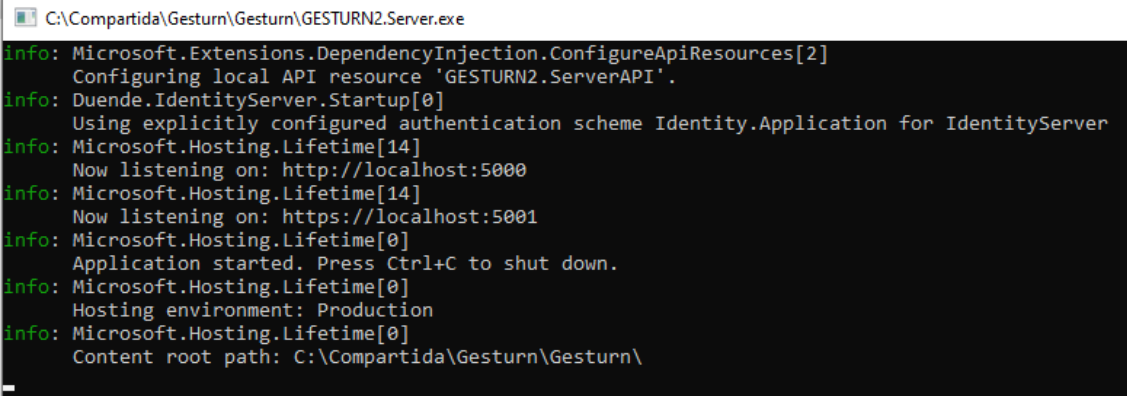
Como podemos observar en “DefaultConnection” hemos configurado el servidor local con la instancia a la base de datos “127.0.0.1\\SQLEXPRESS”, si quisiéramos que la aplicación se comuniqué con otra instancia u otro servidor deberíamos modificar esto.

A continuación, se le indica el nombre de la base de datos “GESTURN_BBDD”, la web la creará en el primer arranque si no existe.

Por último, en “User Id=sa” se configura el nombre del usuario con el que accedemos al servidor de base de datos y con “Password=Uoc2022” configuramos la contraseña.

También es importante la ruta del certificado y la contraseña que este tiene, ya que la web necesita de un certificado válido para su correcto funcionamiento. Por defecto ya viene configurado un certificado válido, aunque en el siguiente apartado de la implantación mostramos como crear un nuevo certificado auto-afirmado válido.

Una vez tenemos la aplicación configurada podemos comprobar su correcto funcionamiento de forma manual, para ello nos situamos en el directorio raíz de la plataforma web y ejecutamos el archivo “GESTURN2.SERVER”. Si la configuración no es correcta podremos observar el error por pantalla, si por lo contrario, todo está correctamente configurado visualizaremos esta pantalla:



```

C:\Compartida\Gesturn\Gesturn\GESTURN2.Server.exe
info: Microsoft.Extensions.DependencyInjection.ConfigureApiResources[2]
      Configuring local API resource 'GESTURN2.ServerAPI'.
info: Duende.IdentityServer.Startup[0]
      Using explicitly configured authentication scheme Identity.Application for IdentityServer
info: Microsoft.Hosting.Lifetime[14]
      Now listening on: http://localhost:5000
info: Microsoft.Hosting.Lifetime[14]
      Now listening on: https://localhost:5001
info: Microsoft.Hosting.Lifetime[0]
      Application started. Press Ctrl+C to shut down.
info: Microsoft.Hosting.Lifetime[0]
      Hosting environment: Production
info: Microsoft.Hosting.Lifetime[0]
      Content root path: C:\Compartida\Gesturn\Gesturn\

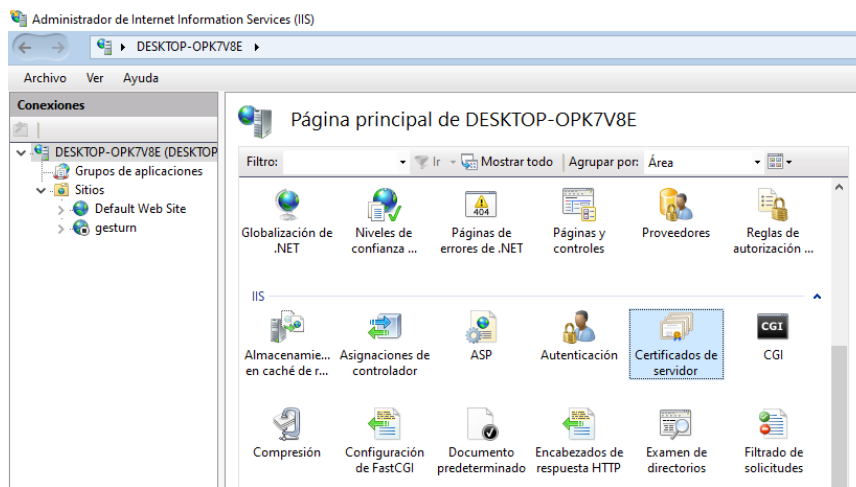
```

6.2.1 Creación del certificado auto-firmado

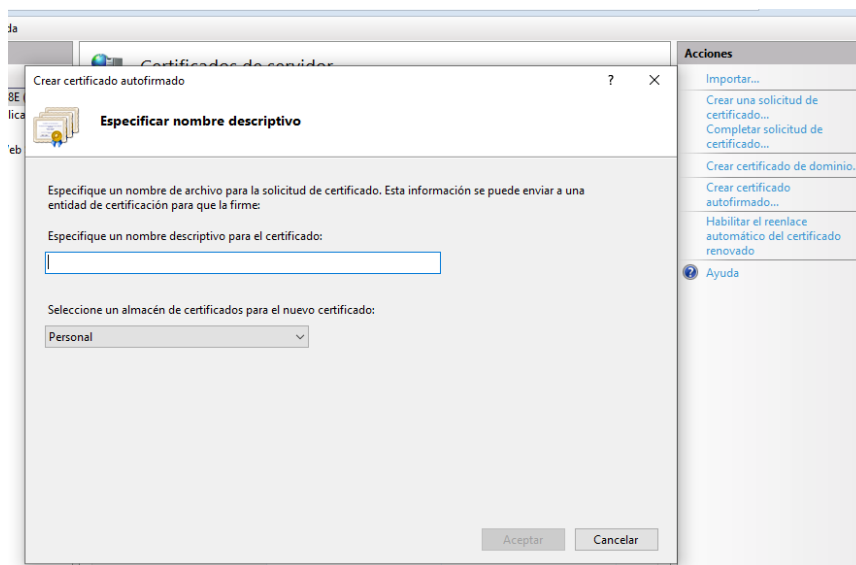
Existen multitud de formas de crear un certificado auto-firmado válido, una manera rápida y sencilla es utilizando la herramienta web (<https://getacert.com/>). Con esta web podemos crear nuestro certificado y sustituirlo por el existente de nuestra aplicación, como hemos comentado en anteriormente, se deberá reconfigurar el archivo “appsettings.production” para que la aplicación funcione con el nuevo certificado.

6.3 Instalación/configuración de la plataforma web en IIS

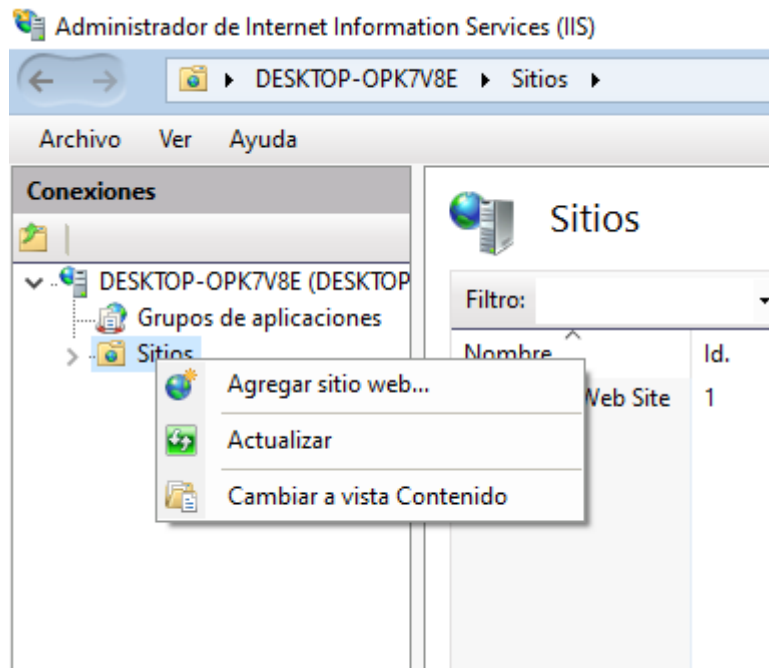
Para poder poner en funcionamiento la aplicación en el IIS primero necesitamos crear un certificado de servidor:



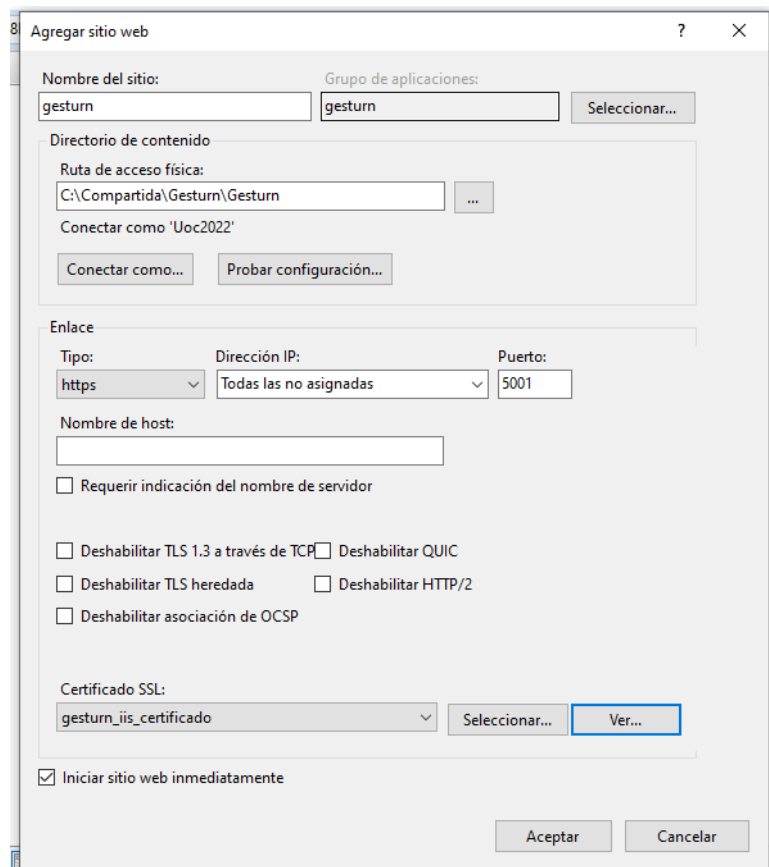
En este caso vamos a utilizar un certificado autofirmado, indicamos el nombre que deseemos, por ejemplo: “gesturn_iis_certificado” y lo creamos:



A continuación, agregamos el nuevo sitio web:



Configuramos el nombre del sitio, la ruta física del directorio raíz de la aplicación, un usuario que tenga permisos de lectura/escritura en ese directorio, el enlace https a la aplicación con el puerto 5001 y por último escogemos el certificado SSL creado previamente:



Ahora podemos comprobar que el sitio web está funcionando, accediendo directamente con el navegador, para ello debemos previamente tener conocimiento de la IP de la máquina virtual.

Si abrimos una consola de comandos y escribimos “ipconfig” nos mostrará la IP que tiene asignada esta máquina:

```
C:\Windows\system32\cmd.exe
C:\Users\Uoc2022>ipconfig

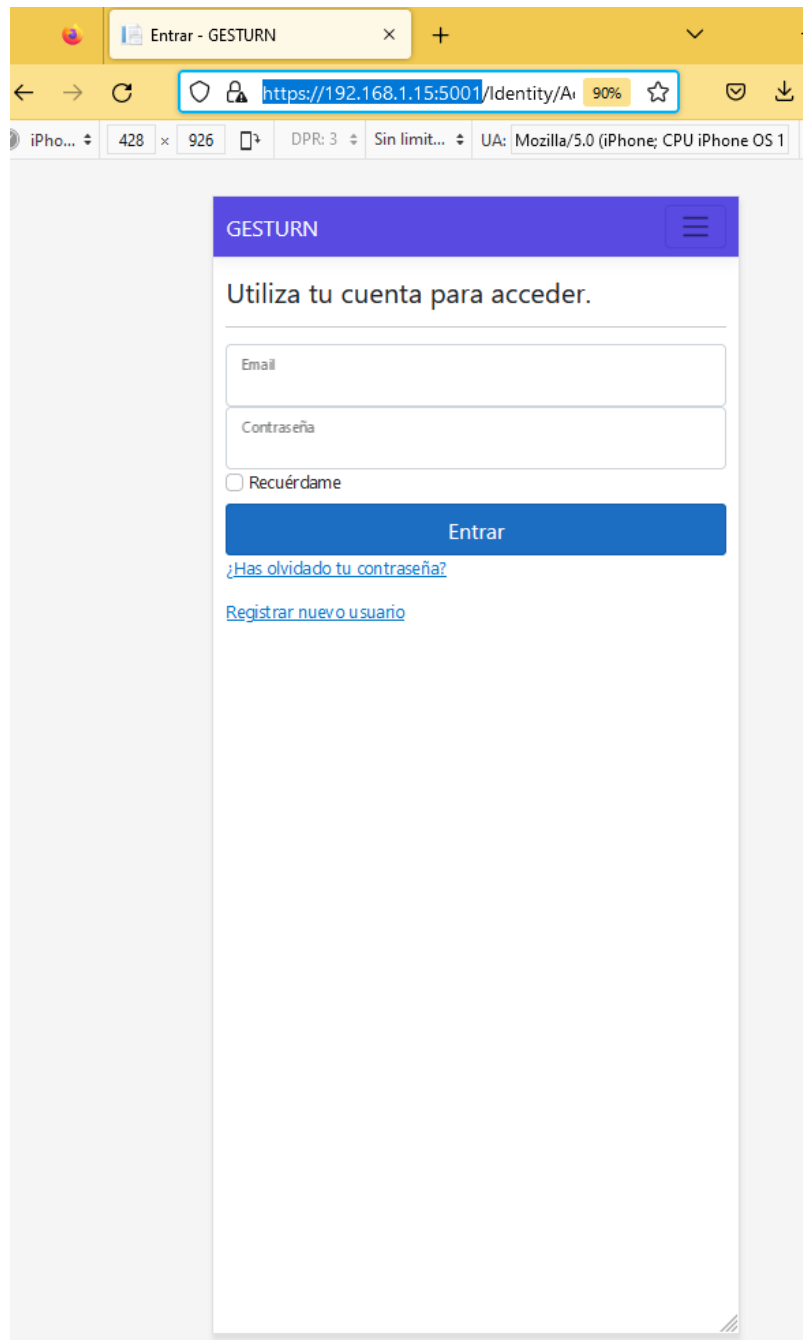
Configuración IP de Windows

Adaptador de Ethernet Ethernet:

    Sufijo DNS específico para la conexión. . . : home
    Vínculo: dirección IPv6 local. . . . . : fe80::6ea9:e9e8:2fde:d3b9%3
    Dirección IPv4. . . . . : 192.168.1.15
    Máscara de subred . . . . . : 255.255.255.0
    Puerta de enlace predeterminada . . . . . : fe80::f830:5cff:fe4b:18a5%3
                                                192.168.1.1

C:\Users\Uoc2022>
```

En mi caso es la 192.168.1.15, por lo tanto, podremos acceder desde cualquier navegador que esté en la misma red que la máquina virtual a la URL: <https://192.168.1.15:5001>



6.4 Observaciones adicionales

La implantación de este prototipo se ha realizado en un sistema operativo Windows aunque es viable realizar la implantación en un sistema operativo Unix, como por ejemplo Linux. Todas las dependencias son compatibles.

También se ha probado el despliegue en la nube de **Azure**, utilizando los servicios gratuitos y el saldo que ofrecen para nuevas cuentas. Rápidamente la plataforma estaba funcionando en la nube, adjunto captura con los servicios utilizados:

<input type="checkbox"/> Nombre ↑↓	Tipo ↑↓	Grupo de recursos ↑↓	Ubicación ↑↓	Suscripción ↑↓
<input type="checkbox"/>  ClaveGesturn	Almacén de claves	GESTURN2Server	West Europe	Azure subscription 1
<input type="checkbox"/>  GESTURN	App Service	GESTURN2Server	West Europe	Azure subscription 1
<input type="checkbox"/>  GESTURN2	Plan de App Service	GESTURN2Server	West Europe	Azure subscription 1
<input type="checkbox"/>  gesturnserver	SQL Server	GESTURN2Server	West Europe	Azure subscription 1

7. Conclusiones

A modo de cierre de este trabajo, podemos evidenciar que el objetivo del proyecto ha sido superado. Con el uso del aplicativo construido podemos visualizar/gestionar de forma remota los turnos de la cola deseada y aprovechar el tiempo de la mejor manera posible.

Durante la realización de este Trabajo de Final de Grado el aprendizaje ha sido constante, la poca experiencia en la planificación de proyectos ha hecho que sea un reto para mí en el que he podido aprender mucho de cada una de las fases.

He podido comprobar la importancia de una buena planificación, siendo vital para el correcto desarrollo del proyecto y del resultado final. Hasta la fecha, los pequeños proyectos que he desarrollado no disponían de fase de análisis ni diseño, gracias a este proyecto, en el que sí que se han realizado esas fases, he comprobado que facilitan enormemente la fase de desarrollo del producto.

En general estoy muy satisfecho con el resultado final del prototipo, he tenido la oportunidad de aprender nuevas tecnologías como Blazor y repasar el uso de otras tantas como MSSQLS, HTML, CSS, así como poner en práctica el conocimiento que hemos obtenido en otras asignaturas del grado.

Se han cumplido todos los requisitos funcionales y no funcionales propuestos, por lo que considero que en este aspecto ha sido un éxito.

La metodología utilizada ha sido mediante fases en cascada, a excepción de la fase de desarrollo del prototipo en la que se ha optado por una metodología iterativa en la que cada iteración ha resultado en un producto funcional que añade valor al producto final. Esta elección de metodología ha sido un acierto, logrando completar el prototipo de manera satisfactoria dentro de la planificación establecida.

El prototipo obtenido no deja de ser una primera aproximación, el prototipo es completamente funcional, pero hay mucho margen de mejora. Algunas de las líneas de futuro que no se han desarrollado en este proyecto son: la implementación de diversos idiomas y así aumentar el alcance, añadir una mayor personalización de las salas y encuestas, modificar la pantalla principal para que aparezcan las salas cercanas al usuario, crear un proceso que calculen automáticamente el tiempo medio aproximado de turno por sala y por último no permitir la creación de salas a todos los usuarios para que solo unos pocos puedan hacerlo.

8. Glosario

LinkedIn: Red social orientada al uso empresarial, permitiendo la búsqueda de empleo.
Sitio web: <https://es.linkedin.com/>

Fórmula coste/sueldo: Fórmula utilizada para el cálculo. Sitio web:
<https://www.personio.es/glosario/coste-de-trabajador-para-empresa/>

BOE: Boletín Oficial del Estado. Sitio web:
https://www.boe.es/diario_boe/txt.php?id=BOE-A-2018-3156

mAh: Miliamperios o mAh.

Framework: Un *framework* es un conjunto de herramientas y librerías que se utilizan para facilitar el desarrollo de aplicaciones.

Wireframe: también conocido como un esquema de página o plano de pantalla, es una guía visual que representa el esqueleto o estructura visual de un sitio web.

IDE: Entorno de desarrollo integrado (IDE) es una aplicación de software que ayuda a los programadores a desarrollar software de manera eficiente.

Git: Sistema de control de revisiones de código abierto. Sitio web:
<https://www.atlassian.com/es/git/tutorials/what-is-git>

Mudblazor: Librería de programación. Sitio web: <https://mudblazor.com/>

QRCode: Librería de programación. Sitio web:
<https://github.com/codebude/QRCode/>

Entity Framework: Conjunto de tecnologías de ADO .NET que facilitan el desarrollo de aplicaciones. Sitio web: <https://learn.microsoft.com/es-es/dotnet/framework/data/adonet/ef/overview>

MVC: Patrón de arquitectura de software Modelo-Vista-Controlador (MVC)

Razor: lenguaje de plantillas basado en HTML y C# que se usa para generar dinámicamente la interfaz de usuario web.

VirtualBox: Software de virtualización de máquinas virtuales, propiedad de Oracle.
Sitio web: <https://www.virtualbox.org/wiki/Downloads>

RAM: Memoria de acceso aleatorio (*Random Access Memory, RAM*)

9. Bibliografía

[1] *Avisadores remotos | UnGatoNipón*. (n.d.). Recuperado el 12 de enero, 2023, desde <https://www.ungatonipon.com/746/avisadores-remotos>

[2] *Sistemas de Llamada Para Restauración. Restaurantes >> Touch It Global*. (n.d.). Recuperado el 12 de enero, 2023, desde https://www.touchitglobal.com/sector/restauracion/?gclid=CjwKCAjw-rOaBhA9EiwAUkLV4o FqYDDKnKYRbCjxbukrunzqxpG5jFZ38o5k706rpsjA2GN95sU_xoC KEYQAvD BwE

[3] *5.4. Crear la EDT/WBS - Project Management | Gladys Gbegnedji*. (n.d.). Recuperado el 12 de enero, 2023, desde <https://www.gladysgbegnedji.com/crear-la-edt-wbs/>

[4] *Días laborables calculadora en España*. (n.d.). Recuperado el 12 de enero, 2023, desde <https://www.dias-laborables.es/>

[5] *Coste del trabajador para la empresa: Guía + Fórmula | Personio*. (n.d.). Recuperado el 12 de enero, 2023, desde <https://www.personio.es/glosario/coste-de-trabajador-para-empresa/>

[6] *Free Project Status Report Template - Project Management Docs*. (n.d.). Recuperado el 12 de enero, 2023, desde <https://www.projectmanagementdocs.com/template/project-monitoring-and-controlling/project-status-report/>

[7] *Qué es un requerimiento funcional? - La Oficina de Proyectos de Informática*. (n.d.). Recuperado el 12 de enero, 2023, desde <http://www.pmoinformatica.com/2018/05/que-es-requerimiento-funcional.html>

[8] *StarUML*. (n.d.). Recuperado el 12 de enero, 2023, desde <https://staruml.io/>

[9] *La diferencia entre el modelo de datos conceptual y lógico | Talent Garden*. (n.d.). Recuperado el 12 de enero, 2023, desde <https://talentgarden.org/es/data/the-difference-between-conceptual-and-logical-data-model/>

[10] *¿Qué es un archivo ISO? Definición y aplicación de archivos ISO*. (n.d.). Recuperado el 12 de enero, 2023, desde <https://www.ionos.es/digitalguide/servidores/know-how/que-es-un-archivo-iso/>

[11] *Qué es Git | Atlassian Git Tutorial*. (n.d.). Recuperado el 12 de enero, 2023, desde <https://www.atlassian.com/es/git/tutorials/what-is-git>

[12] *IdentityUser Class (Microsoft.AspNetCore.Identity) | Microsoft Learn*. (n.d.). Recuperado el 12 de enero, 2023, desde <https://learn.microsoft.com/en-us/dotnet/api/microsoft.aspnetcore.identity.identityuser?view=aspnetcore-6.0>

[13] *Modelos de hospedaje Blazor en ASP.NET Core | Microsoft Learn*. (n.d.). Recuperado el 12 de enero, 2023, desde <https://learn.microsoft.com/es-es/aspnet/core/blazor/hosting-models?view=aspnetcore-7.0>

10. Anexo I: Informe ejecutivo



Universitat
Oberta
de Catalunya

GESTURN

Sistema de consulta remota de turnos

Informe ejecutivo

23 de octubre de 2022

Trabajo de Final de Grado en Ingeniería Informática

Itinerario en Tecnologías de la información

Área de Sistemas de Gestión del conocimiento

Miguel García González

Dirección académica: Javier Martí Pintanel

Responsable académico: Atanasi Daradoumis Haralabus

En innumerables ocasiones de la vida cotidiana nos encontramos con que debemos esperar a ser atendidos, ya sea físicamente o incluso virtualmente, obligándonos a permanecer “in-situ” o constantemente pendientes a que llegue nuestro turno.

Actualmente la única solución conocida para poder remediar este problema es la utilización de unos dispositivos físicos bastante costosos llamados coloquialmente “avisadores”. Estos dispositivos disponen únicamente de la función de advertir a su portador que su turno es el siguiente para que se aproxime de nuevo al establecimiento. La función de estos dispositivos es muy limitada, deben ser recargados con frecuencia y además pueden extraviarse.

Entonces, para dar solución a este problema y poder amenizar las esperas en los turnos el proyecto GESTURN (Sistema de consulta remota de turnos) tiene como objetivo diseñar un sistema de tecnología web de gestión de contenido, gratuito y disponible desde cualquier smartphone con conexión a internet.

Por estos motivos, el **alcance** se centra en el desarrollo de un prototipo funcional de sistema informático que permita la mejora en la gestión del conocimiento de los turnos en cada negocio o establecimiento.

Para alcanzar los objetivos del proyecto se aplicará la metodología PMBOK, utilizando la estructura de desglose de trabajo (EDT) facilitando de esta manera la estimación de los recursos y el cálculo del tiempo y el coste.

Tras el análisis se establecen los siguientes grupos de tareas:

- **Gestión del proyecto:** analizar el proyecto en su totalidad para realizar una valoración de la viabilidad, creación de las tareas, temporalización de cada fase e informes de seguimiento. Realización del plan de riesgos, valoración económica del esfuerzo total y guía de implantación del prototipo funcional.
- **Análisis:** determinar los requisitos funcionales y no funcionales, definición de los *stakeholders* y casos de uso.
- **Diseño:** diseño de toda la interfaz de la aplicación, modelo de base de datos, funcionalidades y arquitectura.
- **Construcción:** construcción del entorno de desarrollo del prototipo y creación de los manuales necesarios para su utilización.
- **Pruebas:** diseñar y ejecutar diferentes pruebas para comprobar el correcto funcionamiento del sistema.

Se han evaluado los posibles **riesgos** que pueden acontecer durante el transcurso de vida del proyecto, estableciendo acciones para **prevenir o corregir** estos riesgos.

Algunos de estos riesgos podrían ser el desconocimiento de las tecnologías punteras para el desarrollo web, inexperiencia en la planificación de proyectos, un alcance excesivo e incluso problemas personales que pudieran afectar al desarrollo del proyecto.

Las medidas de prevención o corrección de estos ejemplos están detalladas en la planificación del proyecto, podemos destacar algunas de estas medidas: formación al personal, reducción del alcance o utilización de copias de seguridad de respaldo.

Los **roles** identificados para la realización del proyecto son los siguientes:

- **Director de proyectos:** seguimiento de la correcta planificación del proyecto, elaborar informes, reuniones, gestión de riesgos...
- **Analista de software:** análisis de casos, encuestas y elección de la mejor tecnología para el desarrollo del proyecto, entre otras funciones.
- **Desarrollador web:** encargado de la construcción de la página web.
- **Desarrollador UI/UX:** implementar una interfaz atractiva, intuitiva y que ofrezca una buena experiencia al usuario.
- **Técnico de sistemas:** encargado en la implementación / configuración del sistema ya sea en servidores privados o públicos, también se encarga de la seguridad del servidor.

Para la ejecución total del proyecto se invierten un total de **230 horas de trabajo**, con un **coste aproximado de 8840€** incluyendo un plan de contingencias de 500€.

Se estima que el proyecto tenga una **duración de 83 días naturales** (55 días hábiles) teniendo como fecha de inicio el 15/10/2022 y como fecha final el 05/01/2023. A continuación, podemos visualizar los hitos de la planificación:

Hito	Fecha
Planteamiento del proyecto	15/10/2022
Planificación del proyecto	18/10/2022
Entrega Informe Ejecutivo (PEC 1)	23/10/2022
Finalización del diseño de la interfaz	14/11/2022
Informe de seguimiento (PEC 2)	20/11/2022
Finalización de pruebas	03/01/2022
Finalización desarrollo entorno web (PEC 3)	05/01/2023

11. Anexo II: Informe de seguimiento



Universitat
Oberta
de Catalunya

GESTURN

Sistema de consulta remota de turnos

Primer Informe de seguimiento

20 de noviembre de 2022

Trabajo de Final de Grado en Ingeniería Informática

Itinerario en Tecnologías de la información

Área de Sistemas de Gestión del conocimiento

Miguel García González

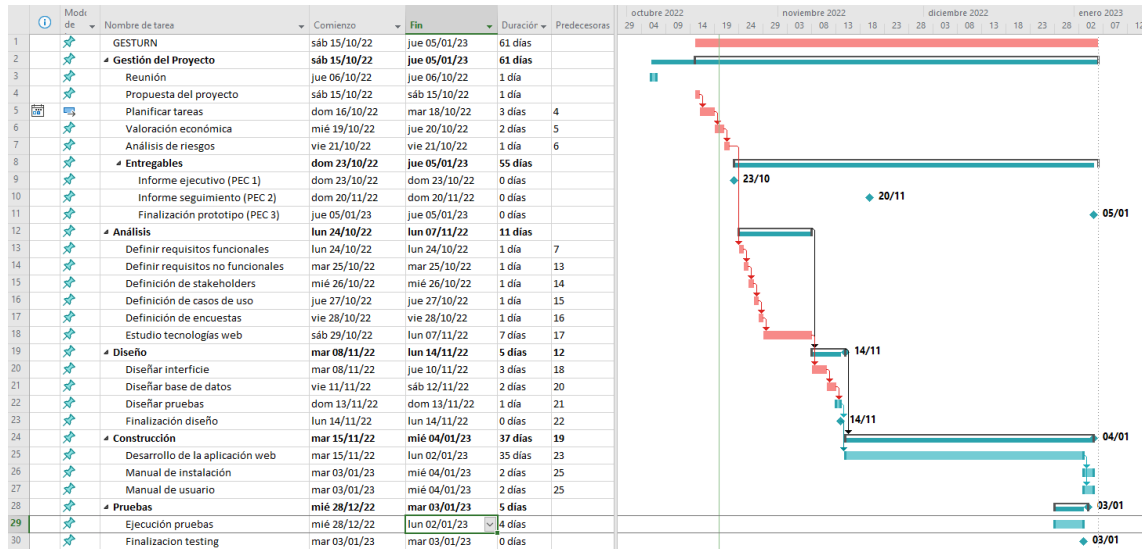
Profesor director del TFG: Javier Martí Pintanel

Profesor director del área: Atanasi Daradoumis Haralabus

1. Estado del proyecto

Este informe resume de manera general el estado del desarrollo del proyecto hasta la fecha, evaluando los principales indicadores, indicando los hitos completados y las tareas (completadas, en ejecución y pendientes).

Recuperamos el diagrama de Gantt de la fase de planificación para poder observar en qué punto nos encontramos del desarrollo:



2. Entregables

Se adjunta el documento con el análisis del dominio y análisis funcional del proyecto y otro documento con el diseño realizado hasta la fecha.

3. Indicadores

Con el fin de tener una idea general del estado del proyecto, vamos a utilizar una serie de indicadores para ver el nivel de desviación que hay actualmente con cada uno de ellos:

Indicador	Estado	Descripción
Alcance	Correcto	No se ha modificado el alcance inicial del proyecto.
Planificación	Buena	Se han producido algunos retrasos con la finalización de tareas específicas, pero se han mitigado.
Coste	Correcto	Se mantiene el coste inicial ya que el esfuerzo no ha sufrido cambios.
Riesgos	Correctos	Se descartan riesgos y se mantienen los riesgos iniciales restantes sin contemplar de nuevos.
Calidad	Correcta	La calidad del trabajo realizado es correcta.

El único indicador que se ha visto afectado ha sido el de planificación, pudiendo mitigar estos retrasos con acciones correctivas.

4. Hitos

En la siguiente tabla vamos a visualizar los hitos alcanzados y los pendientes con la fecha prevista de realización:

Hito	Fecha	¿Finalizado?
Planteamiento del proyecto	19/10/2022	<input checked="" type="checkbox"/>
Planificación del proyecto	21/10/2022	<input checked="" type="checkbox"/>
Entrega Informe Ejecutivo (PEC 1)	23/10/2022	<input checked="" type="checkbox"/>
Finalización del diseño de la interfaz	06/11/2022	<input checked="" type="checkbox"/>
Informe de seguimiento (PEC 2)	20/11/2022	<input checked="" type="checkbox"/>
Finalización de pruebas	29/12/2022	<input type="checkbox"/>
Finalización desarrollo entorno web (PEC 3)	05/01/2023	<input type="checkbox"/>

5. Gestión de riesgos

Vamos a re-evaluar los riesgos en esta etapa del proyecto, indicando los riesgos que han podido aumentar o disminuir la probabilidad de aparición o su impacto. Para ello, vamos a recuperar la tabla de los riesgos detectados en la planificación del proyecto:

Código	Riesgo	Descripción	Probabilidad de aparición inicial	Impacto inicial
R1	Desconocimiento tecnologías	Desconocimiento de las diferentes tecnologías para el desarrollo web.	Baja	Alto
R2	Inexperiencia planificación	Falta de conocimientos o de experiencia en la planificación de proyectos.	Media	Medio
R3	Alcance excesivo	Es complicado desarrollar aplicaciones que tengan como objetivo la mayoría de la población.	Baja	Medio
R4	Problemas personales	El desarrollo completo del proyecto se realiza por una única persona que adoptará los distintos roles, si enferma o le surge algún problema puede afectar seriamente a la entrega.	Baja	Alto
R5	Problemas	Durante la construcción del	Alta	Alto

	técnicos	prototipo pueden surgir errores inesperados.		
R6	Experiencia negativa para el usuario	Si la utilización del prototipo no es intuitiva o incluso lenta, puede ocasionar descontento para el usuario.	Baja	Alto
R7	Pérdida de información	Por un error es posible que se pierda una parte o la totalidad del trabajo realizado.	Baja	Alto

- El riesgo R1 desaparece, después de realizar el estudio de las tecnologías web durante la fase de análisis ya se obtiene el conocimiento mínimo para poder elegir una tecnología adecuada.
- El riesgo R2 reduce su probabilidad de aparición ya que se ha progresado en la planificación del proyecto hasta la fecha de hoy.
- El riesgo R3 también reduce la probabilidad de aparición al haber definido con detalle los requisitos funcionales, no funcionales, casos de uso y roles durante el análisis.
- El riesgo R4 se mantiene, no se puede descartar ni reducir su probabilidad de aparición ya que puede ser inesperado en cualquier fase del proyecto.
- Los riesgos R5 y R6 se mantienen de igual manera ya que aún no se ha empezado la construcción del prototipo.
- El riesgo R7 se reduce en menor medida ya que ya han sido entregados algunos entregables del proyecto, igualmente es un riesgo que no puede ser eliminado hasta la finalización del proyecto.

6. Re-planificación y re-análisis

Durante la fase de diseño se ha tomado la decisión de alternar el orden de las tareas de diseño de la base de datos con la de diseño de interfaz, realizando en primer lugar la de diseño de la base de datos.

Este cambio no afecta a la temporalidad de las entregas ni al esfuerzo, se ha decidido realizar el cambio porque es más sencillo poder diseñar una interfaz de usuario cuando ya tenemos todos los campos de la base de datos definidos.

Durante la fase de análisis se han destinado más horas de las previstas a la tarea de definición de casos de uso. Se ha podido compensar este incremento de las horas en esa tarea con otras que se han finalizado en menos tiempo, como por ejemplo, las tareas de definición de *stakeholders* y definición de encuestas.

12. Anexo III: Segundo Informe de seguimiento



Universitat
Oberta
de Catalunya

GESTURN

Sistema de consulta remota de turnos

Segundo Informe de seguimiento

5 de enero de 2023

Trabajo de Final de Grado en Ingeniería Informática

Itinerario en Tecnologías de la información

Área de Sistemas de Gestión del conocimiento

Miguel García González

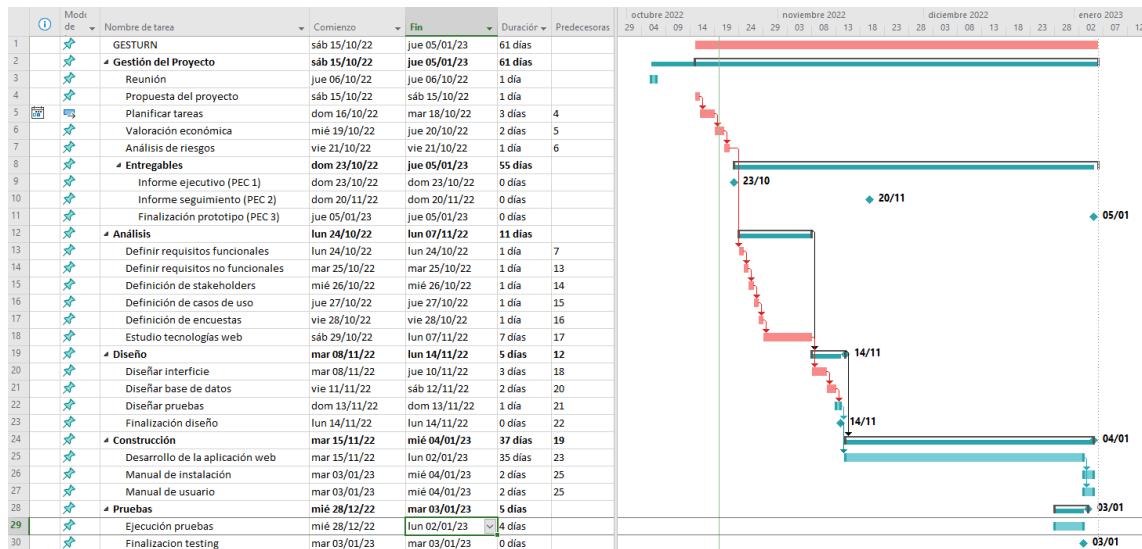
Profesor director del TFG: Javier Martí Pintanel

Profesor director del área: Atanasi Daradoumis Haralabus

1. Estado del proyecto

Este informe resume el estado actual del proyecto a modo de cierre. Se ha finalizado la etapa de desarrollo del proyecto por lo que se va a evaluar los principales indicadores, indicando los hitos completados y las tareas realizadas.

Recuperamos el diagrama de Gantt de la fase de planificación para poder observar las fechas en las que se finaliza cada tarea y la fecha de finalización prevista:



2. Entregables

Junto a este informe se adjuntan los siguientes entregables:

- **Documento de implementación del prototipo.** Se adjunta el documento de implementación con la totalidad de las tareas realizadas y cumpliendo con el análisis y diseño establecido previamente.
- **Documento de implantación del prototipo.** Se adjunta el documento de implantación en el que se detalla paso a paso cómo poner en marcha el sistema en un nuevo entorno.
- **Paquete de la publicación del aplicativo.** Se adjunta el comprimido con los archivos necesarios para montar la plataforma en un servidor web.
- **Fichero de la máquina virtual.** Se adjunta el enlace para la descarga de la máquina virtual completa en el último punto del documento de implementación.
- **Segundo informe de seguimiento.** Documento actual donde podemos comprobar el estado final del proyecto.

3. Indicadores

Con el fin de tener una idea general del estado del proyecto, vamos a utilizar una serie de indicadores para ver el nivel de desviación que hay actualmente con cada uno de ellos:

Indicador	Estado	Descripción
Alcance	Correcto	No se ha modificado el alcance inicial del proyecto.
Planificación	Correcta	No se ha producido ninguna modificación en la planificación.
Coste	Correcto	Se mantiene el coste inicial ya que el esfuerzo no ha sufrido cambios.
Riesgos	Correctos	Se descartan los riesgos al finalizar el proyecto.
Calidad	Correcta	La calidad del trabajo realizado es correcta.

4. Hitos

En la siguiente tabla vamos a visualizar los hitos alcanzados y la fecha en la que se han realizado:

Hito	Fecha	¿Finalizado?
Planteamiento del proyecto	19/10/2022	<input checked="" type="checkbox"/>
Planificación del proyecto	21/10/2022	<input checked="" type="checkbox"/>
Entrega Informe Ejecutivo (PEC 1)	23/10/2022	<input checked="" type="checkbox"/>
Finalización del diseño de la interfaz	06/11/2022	<input checked="" type="checkbox"/>
Informe de seguimiento (PEC 2)	20/11/2022	<input checked="" type="checkbox"/>
Finalización de pruebas	29/12/2022	<input checked="" type="checkbox"/>
Finalización desarrollo entorno web (PEC 3)	05/01/2023	<input checked="" type="checkbox"/>

5. Gestión de riesgos

Se ha finalizado el proyecto en su totalidad por lo que los riesgos ya no pueden tener ningún efecto.

Recuperamos los riesgos detectados en la planificación del proyecto para actualizar su estado en la probabilidad e impacto:

Código	Riesgo	Descripción	Probabilidad de aparición / Impacto
R1	Desconocimiento tecnologías	Desconocimiento de las diferentes tecnologías para el desarrollo web.	Inexistente
R2	Inexperiencia planificación	Falta de conocimientos o de experiencia en la planificación de proyectos.	Inexistente
R3	Alcance excesivo	Es complicado desarrollar aplicaciones que tengan como	Inexistente

		objetivo la mayoría de la población.	
R4	Problemas personales	El desarrollo completo del proyecto se realiza por una única persona que adoptará los distintos roles, si enferma o le surge algún problema puede afectar seriamente a la entrega.	Inexistente
R5	Problemas técnicos	Durante la construcción del prototipo pueden surgir errores inesperados.	Inexistente
R6	Experiencia negativa para el usuario	Si la utilización del prototipo no es intuitiva o incluso lenta, puede ocasionar descontento para el usuario.	Inexistente
R7	Pérdida de información	Por un error es posible que se pierda una parte o la totalidad del trabajo realizado.	Inexistente

6. Re-planificación y re-análisis

Durante la fase de construcción y pruebas no se ha alterado la temporalización de las tareas, por lo que no se ha visto afectado el esfuerzo total.

La fase de construcción se componía principalmente por una única tarea, la construcción de la plataforma web. Durante esta fase se ha detectado que realmente esta tarea se podía haber dividido en dos: una tarea de formación y otra de desarrollo. Igualmente se ha cumplido con la fecha prevista para la finalización de la tarea.