



Universidad de Valladolid

**ESCUELA DE INGENIERÍA
INFORMÁTICA DE SEGOVIA**

**Grado en Ingeniería
Informática de Servicios y
Aplicaciones**

**OVERNEWS: Aplicación de noticias
multilingüe**

**Javier Calle Olivier
Tutores:
FERNANDO DÍAZ GÓMEZ
JOSE IGNACIO FARRÁN MARTÍN**

Índice general

Índice de figuras	6
Índice de tablas	9
Agradecimientos	13
Resumen	15
Abstract.....	17
Capítulo 1 Descripción del proyecto	19
1.1. Introducción	19
1.2. Objetivos del trabajo	20
1.3. Estado del arte.....	20
1.3.1 Solución propuesta	20
1.3.2 Entorno de la aplicación.....	21
1.4 Alcance (Árbol de características)	23
1.5 Entorno y tecnologías utilizadas	25
1.5.1. <i>Backend</i>	25
1.5.2. <i>Frontend</i>	27
1.5.3 Sistema de control de versiones	30
1.6 Estructura de la memoria	31
Capítulo 2 Planificación	33
2.1. Metodología de trabajo	33
2.2. Planificación del proyecto	34
2.2.1. Planificación temporal inicial.....	34
2.2.2. Análisis de riesgos.....	40
2.2.3. Presupuesto.....	42
2.2.4 Seguimiento Real	52
Capítulo 3 Análisis	55
3.1 Requisitos de Usuario	55
3.2 Casos de Uso.....	56
3.2.1 Especificación de los Actores	56
3.2.2 Listado de Casos de Uso	57
3.2.3 Diagrama de Casos de Uso	60
3.2.4 Especificación de Casos de Uso.....	60

3.3 Requisitos funcionales	69
3.3.1 Requisitos de información.....	72
3.4 Requisitos no funcionales	73
Capítulo 4 Diseño	75
4.1 Arquitectura Lógica	75
4.2 Arquitectura Física.....	77
4.3 Diagramas (secuencia y actividad)	78
4.3.1 Diagrama de secuencia iniciar sesión.....	78
4.3.2 Diagrama de secuencia obtención y traducción de noticias.....	79
4.3.3 Diagrama de secuencia mostrar noticias	79
4.4 Modelo de datos	81
4.4.1 Modelo Entidad-Relación	81
4.4.2 Diseño lógico.....	82
4.4.3 Diseño físico.....	82
4.5 Diseño de las Interfaces de Usuario.....	86
Capítulo 5 Implementación	99
5.1 Estructura del proyecto	99
5.1.1 <i>Backend</i>	99
5.1.2 <i>Frontend</i>	109
5.1.3 API Traducción	112
5.2 Cuestiones de implementación reseñables.....	114
5.2.1 Manejo de datos persistentes y temporales	114
5.2.2 Obtención y traducción de las noticias.....	116
5.2.3 Gestión de Autenticación y Autorización	119
Capítulo 6 Pruebas.....	121
6.1 Pruebas de caja blanca	121
6.2 Pruebas de caja negra.....	122
Capítulo 7 Conclusiones.....	131
7.1 Conclusiones	131
7.2 Líneas de trabajo futuras.....	131
Webgrafía	133
Anexo I Manual de despliegue	135
Manual para el <i>Backend</i> :.....	135
Manual para el <i>Frontend</i> :	137

Manual para la API de traducción en local:	137
Anexo II Manual de usuario	139
Anexo III Manual de administrador	155

Índice de figuras

Figura 1. Google News.....	21
Figura 2. Flipboard	21
Figura 3. Árbol de características	24
Figura 4. PostgreSQL	25
Figura 5. FastAPI.....	26
Figura 6. Node.js	27
Figura 7. ReactJS.....	28
Figura 8. Material-UI.....	29
Figura 9. Git.....	30
Figura 10. GitHub.....	30
Figura 11. Gráfico Modelo en Espiral	33
Figura 12. Diagrama de Gantt (Incremento I)	36
Figura 13. Diagrama de Gantt (Incremento II).....	37
Figura 14. Diagrama de Gantt (Incremento III)	38
Figura 15. Diagrama de Gantt (Incremento IV)	39
Figura 16. Diagrama de Casos de Uso	60
Figura 17. Arquitectura lógica.....	76
Figura 18. Arquitectura Física.....	77
Figura 19. Diagrama de Secuencia Iniciar Sesión	78
Figura 20. Diagrama de secuencia obtención y traducción de noticias.....	79
Figura 21. Diagrama de secuencia mostrar noticias	80
Figura 22. Modelo Entidad-Relación	81
Figura 23. Diccionario de datos "USER" (Usuario).....	83
Figura 24. Diccionario de datos "SUGGESTION" (Sugerencias)	84
Figura 25. Diccionario de datos "NEW" (Noticia).....	84
Figura 26. Diccionario de datos "CATEGORY" (Categoría)	84
Figura 27. Diccionario de datos "SOURCE" (Fuente)	85
Figura 28. Diccionario de datos "PREF_CATEGORY" (Preferencia_Categoría).....	85
Figura 29. Diccionario de datos "PREF_SOURCE" (Preferencia_Fuente)	85
Figura 30. Diccionario de datos "COMMENT" (Comentario)	85
Figura 31. Diccionario de datos "SAVEDNEW" (Noticia_Guardada).....	86
Figura 32. Boceto Iniciar Sesión	87
Figura 33. Boceto Registro.....	88
Figura 34. Boceto Página Principal (Usuarios No Registrados)	89
Figura 35. Boceto Página Principal (Usuarios Registrados)	90
Figura 36. Boceto Solicitar Fuente	91
Figura 37. Boceto Mis Noticias	92
Figura 38. Boceto "Mis Comentarios"	93
Figura 39. Boceto "Sobre Nosotros"	94
Figura 40. Boceto "Mostrar Noticia"	95
Figura 41. Boceto "Gestionar Perfil"	96
Figura 42. Boceto "Panel del Administrador"	97
Figura 43. Estructura del Backend	99

Figura 44. Contenido subdirectorio /dtos	100
Figura 45. Contenido subdirectorio /models	101
Figura 46. Contenido subdirectorio /routers.....	102
Figura 47. Contenido subdirectorio /scripts	104
Figura 48. Contenido subdirectorio /obtain_news.....	104
Figura 49. Contenido subdirectorio /services	106
Figura 50. Aclaración de los códigos de idioma	108
Figura 51. Ejemplo Contenido news_ar	108
Figura 52. Contenido subdirectorio /utils	109
Figura 53. Contenido frontend.....	110
Figura 54. Contenido /public	110
Figura 55. Contenido /src	111
Figura 56. Componentes de React.....	111
Figura 57. Contenido /pages.....	112
Figura 58. Contenido Api Traducción.....	113
Figura 59. Contenido ejemplo de news_en_1	114
Figura 60. Rotación de archivos JSON	115
Figura 61. Script.bat	115
Figura 62. Fichero "rotate_news.py".....	116
Figura 63. Función increment_number_for_news().....	116
Figura 64. Ejecución del planificador en "main.py"	117
Figura 65. Función start_sheduler().....	117
Figura 66. Secuencia obtención de noticias.....	117
Figura 67. Función job_fuction()	118
Figura 68. Secuencia JWT.....	119
Figura 69. Creación BD desde PGAdmin 4.....	136
Figura 70. Variables globales de la base de datos	136
Figura 71. Base de datos creada	137
Figura 72. Página Principal OverNews Visitante.....	139
Figura 73. Pantalla Registro de Usuario.....	139
Figura 74. Correo activación de cuenta	140
Figura 75. Activación completada.....	140
Figura 76. Error en la activación	140
Figura 77. Pantalla Inicia de Sesión	141
Figura 78. Error al Iniciar Sesión	141
Figura 79. Página Principal OverNews Usuario.....	142
Figura 80. Página Principal (Noticias más vistas).....	142
Figura 81. Página Principal (Todas las noticias)	143
Figura 82. Página Mis Noticias	143
Figura 83. Página Mis Noticias al pulsar "Buscar Noticias"	144
Figura 84. Página Mis Noticias al "Guardar Ajustes"	144
Figura 85. Pantalla Solicitar Fuente	145
Figura 86. Pantalla Solicitar Fuente al "Añadir Sugerencia"	145
Figura 87. Pantalla Mis Comentarios	146
Figura 88. Pantalla Sobre Nosotros	147
Figura 89. ReCAPTCHA al "Contáctanos".....	147
Figura 90. Error por no rellenar recaptcha	148

Figura 91. Correo electrónico "Contáctanos"	148
Figura 92. Pantalla Ver Noticia	149
Figura 93. Añadir Comentario a una noticia	150
Figura 94. Pantalla Gestionar Mi Perfil.....	150
Figura 95. PopUp Aviso al "Eliminar Cuenta"	151
Figura 96. Pantalla Recuperar Contraseña	151
Figura 97. Correo Enviado al "Recuperar Contraseña"	152
Figura 98. Correo de Recuperación de Contraseña	152
Figura 99. Página para Recuperar la Contraseña.....	152
Figura 100. Página 404 Not Found.....	153
Figura 101. Pantalla Panel del Administrador.....	156
Figura 102. Panel del Administrador con la Gráfica Modificada.....	156
Figura 103. Sugerencia Aceptada por el Administrador	157
Figura 104. Sugerencia eliminada por el Administrador.....	157

Índice de tablas

Tabla 1. Comparativa Entorno de la Aplicación	22
Tabla 2. Escala de probabilidad.....	40
Tabla 3. Escala de impacto.....	40
Tabla 4. Análisis de Riesgos	41
Tabla 5. Plan de Mitigación y Reducción de Riesgos	42
Tabla 6. Complejidad entradas externas PFNA.....	43
Tabla 7. Entradas externas.....	44
Tabla 8. Complejidades salidas y consultas externas PFNA.....	44
Tabla 9. Salidas externas	45
Tabla 10. Consultas externas.....	45
Tabla 11. Complejidad ficheros lógicos internos y externos PFNA	46
Tabla 12. Ficheros lógicos internos.....	46
Tabla 13. Tabla Ponderaciones PFNA	47
Tabla 14. Factores de Ajuste de Valor (FAV).....	47
Tabla 15. Presupuesto Recursos Técnicos (Método Albrecht).....	49
Tabla 16. Estimación del Presupuesto Total (Método Albrecht)	50
Tabla 17. Presupuesto Recursos Técnicos (Planificación Inicial).....	51
Tabla 18. Estimación del Presupuesto Total (Planificación Inicial)	51
Tabla 19. Comparativa entre estimaciones.....	52
Tabla 20. Presupuesto ajustado (Planificación Final)	53
Tabla 21. Coste Real.....	53
Tabla 22. Comparativa con la Planificación Inicial (Planificación Final)	54
Tabla 23. Requisitos de Usuario.....	56
Tabla 24. Especificación Actor Usuario.....	56
Tabla 25. Especificación Actor Visitante	57
Tabla 26. Especificación Actor Administrador	57
Tabla 27. Casos de Uso	58
Tabla 28. Asociación de Casos de Uso con Requisitos de Usuario	59
Tabla 29. Especificación del CU-01	61
Tabla 30. Especificación del CU-02.....	61
Tabla 31. Especificación del CU-03.....	62
Tabla 32. Especificación del CU-04.....	62
Tabla 33. Especificación del CU-05.....	63
Tabla 34. Especificación del CU-06.....	63
Tabla 35. Especificación del CU-07.....	64
Tabla 36. Especificación del CU-08.....	64
Tabla 37. Especificación del CU-09.....	65
Tabla 38. Especificación del CU-10.....	65
Tabla 39. Especificación del CU-11	66
Tabla 40. Especificación del CU-12.....	66
Tabla 41. Especificación del CU-13.....	67
Tabla 42. Especificación del CU-14.....	67
Tabla 43. Especificación del CU-15.....	68

Tabla 44. Especificación del CU-16.....	68
Tabla 45. Especificación del CU-17.....	69
Tabla 46. Requisitos Funcionales.....	72
Tabla 47. Requisitos de información.....	73
Tabla 48. Requisitos no funcionales.....	74
Tabla 49. Diseño Interfaz de Iniciar Sesión	87
Tabla 50. Diseño Interfaz de Registro	88
Tabla 51. Diseño Interfaz de Página Principal (Usuarios No Registrados).....	89
Tabla 52. Diseño Interfaz de Página Principal (Usuarios Registrados)	90
Tabla 53. Diseño Interfaz de Solicitar Fuent.....	91
Tabla 54. Diseño Interfaz de Mis Noticias.....	92
Tabla 55. Diseño de Interfaz Mis Comentarios.....	93
Tabla 56. Diseño de Interfaz Sobre Nosotros.....	94
Tabla 57. Diseño de Interfaz "Mostrar Noticia "	96
Tabla 58. Diseño de Interfaz "Gestionar Perfil"	97
Tabla 59. Diseño Interfaz "Panel del Administrador"	98
Tabla 60. Endpoints.....	103
Tabla 61. Descripción de los servicios (services).....	107
Tabla 62. PCN-01: Registrar un nuevo usuario.....	122
Tabla 63. PCN-02: Iniciar sesión	122
Tabla 64. PCN-03: Verificar el token actual	122
Tabla 65. PCN-04: Obtener una lista de todos los usuarios	122
Tabla 66. PCN-05: Actualizar la información del usuario	123
Tabla 67. PCN-06: Eliminar un usuario por su id	123
Tabla 68. PCN-07: Eliminar la cuenta del usuario actual	123
Tabla 69. PCN-08: Activar un usuario con el token dado.....	123
Tabla 70. PCN-09: Cambiar la contraseña del usuario	123
Tabla 71. PCN-10: Recuperar contraseña	124
Tabla 72. PCN-11: Actualizar preferencias de usuario	124
Tabla 73. PCN-12: Añadir un nuevo comentario	124
Tabla 74. PCN-13: Eliminar un comentario.....	124
Tabla 75. PCN-14: Actualizar un comentario	125
Tabla 76. PCN-15: Ver comentarios de una noticia.....	125
Tabla 77. PCN-16: Obtener todos los comentarios del usuario actual.....	125
Tabla 78. PCN-17: Obtener una categoría	125
Tabla 79. PCN-18: Añadir una nueva categoría.....	125
Tabla 80. PCN-19: Eliminar una categoría	126
Tabla 81. PCN-20: Obtener una fuente	126
Tabla 82. PCN-21: Añadir una nueva fuente	126
Tabla 83. PCN-22: Eliminar una fuente	126
Tabla 84. PCN-23: Contactar con el soporte.....	126
Tabla 85. PCN-24: Obtener todas las noticias.....	127
Tabla 86. PCN-25 Obtener una noticia multilingüe en específico	127
Tabla 87. PCN-26 Obtener estadísticas de las noticias	127
Tabla 88. PCN-27: Guardar una noticia como favorita.....	127
Tabla 89. PCN-28: Eliminar una noticia guardada como favorita	127
Tabla 90. PCN-29: Obtener las noticias guardadas del usuario actual.....	128

Tabla 91. PCN-30: Añadir una nueva sugerencia	128
Tabla 92. PCN-31: Eliminar una sugerencia por su id.....	128
Tabla 93. PCN-32: Obtener las sugerencias del usuario actual.....	128
Tabla 94. PCN-33: Obtener todas las sugerencias	128
Tabla 95. PCN-34: Aceptar una sugerencia	129

Agradecimientos

Quiero expresar mi sincero agradecimiento a todas las personas que contribuyeron de manera significativa al éxito de este Trabajo de Fin de Grado, **OVERNEWS**. Este proyecto no habría sido posible sin su apoyo, orientación y colaboración.

En primer lugar, quiero agradecer a mis compañeros de clase, quienes han sido una fuente constante de inspiración y motivación a lo largo de este viaje académico. Sus ideas, debates y apoyo mutuo fueron invaluableles en cada etapa del desarrollo de este proyecto. Juntos hemos enfrentado desafíos y alcanzados metas, y estoy agradecido por su amistad y apoyo.

También deseo expresar mi profundo agradecimiento a mis profesores y asesores académicos en la Universidad de Valladolid (UVA). Su experiencia, orientación y valiosos consejos desempeñaron un papel fundamental en la formulación y ejecución de este TFG. Su compromiso con la educación y la excelencia académica ha sido una fuente constante de inspiración para mí.

Resumen

OVERNEWS es una aplicación empresarial que aborda la barrera del idioma para los visitantes de España al proporcionar noticias traducidas en tiempo real, lo que hace que la información sea más accesible y valiosa para una audiencia internacional. El objetivo principal de este proyecto es desarrollar una aplicación web multilingüe que resuelva esta problemática de manera eficiente y efectiva.

Esta aplicación ofrece una solución pragmática a aquellos usuarios que desean acceder a noticias en su lengua materna mientras visitan España. Además, automatiza de manera integral la recolección de noticias actualizadas y su traducción, proporcionando así una experiencia informativa más accesible.

El proyecto aborda desafíos tecnológicos y lingüísticos, demostrando la aplicación de conocimientos en el campo de la Ingeniería Informática. Además, presenta una oportunidad para explorar el desarrollo de aplicaciones web modernas y el uso de tecnologías populares en el desarrollo web.

Palabras claves: Noticias, multilingüe, automatización, actualización, eficiencia, aplicación web, traducción automática.

Abstract

OVERNEWS is a business application that addresses the language barrier for visitors to Spain by providing translated news in real time, making information more accessible and valuable to an international audience. The main objective of this project is to develop a multilingual web application that solves this problem efficiently and effectively.

This application offers a pragmatic solution to those users who want to access news in their native language while visiting Spain. In addition, it comprehensively automates the collection of updated news and its translation, thus providing a more accessible information experience.

The project addresses technological and linguistic challenges, demonstrating the application of knowledge in the field of computer engineering. Additionally, it presents an opportunity to explore modern web application development and the use of popular technologies in web development.

Keywords: News, multilingual, automation, updating, efficiency, web application, automatic translation.

Capítulo 1

Descripción del proyecto

1.1. Introducción

España, conocida por su historia, hermosos paisajes y vibrante cultura, es elegido por millones de personas como uno de los destinos turísticos más prominentes y visitados en el mundo entero. Cada año, millones de viajeros de diversos rincones del planeta llegan a España en busca de experiencias únicas y enriquecedoras. Sin embargo, detrás de la magnífica experiencia que España ofrece, se esconde una barrera que puede obstaculizar la conexión de estos visitantes con la realidad local, el idioma.

La diversidad de lenguas y dialectos que se hablan en el territorio español puede convertirse en un desafío para quienes no tienen fluidez en el idioma local. Esto se manifiesta de manera especialmente marcada en la lectura de noticias locales, donde el acceso a información precisa y actualizada puede verse limitado por la barrera lingüística.

Es precisamente en este contexto, dónde el Trabajo de Fin de Grado (TFG), titulado **OVERNEWS**, emerge como una solución integral y automatizada. Esta aplicación web se concibe con el propósito fundamental de brindar a los visitantes de España la capacidad de acceder a noticias en su lengua materna, sin importar la distancia lingüística que pueda existir.

Una característica esencial de **OVERNEWS** es su funcionamiento completamente automatizado una vez que se encuentra en línea. Esto significa que los usuarios pueden disfrutar de noticias actualizadas y su correspondiente traducción de manera inmediata, sin intervención manual, lo que facilita la obtención de información relevante de forma sencilla y eficiente.

Este TFG representa la aplicación de los diferentes conocimientos obtenidos en el campo de la Ingeniería Informática para abordar una necesidad real y recurrente en el contexto turístico de España. A lo largo de este documento, exploraremos en detalle el desarrollo y funcionamiento de **OVERNEWS**, así como los desafíos tecnológicos y lingüísticos que se superaron en su creación.

A medida que avanzamos en esta exploración, se revelará cómo esta aplicación contribuye de manera significativa a la accesibilidad de la información y a la mejora de la experiencia de aquellos que eligen España como su destino de viaje.

1.2. Objetivos del trabajo

El propósito central de este proyecto es el diseño y la implementación de **OVERNEWS**, una aplicación pionera que busca revolucionar el acceso a la información para los visitantes extranjeros en España. **OVERNEWS** permitirá a los usuarios disfrutar de las noticias locales en su idioma preferido. Los objetivos específicos para alcanzar esta meta son:

- Desarrollo de un software capaz de recolectar y traducir automáticamente diversos artículos o noticias.
- Compatibilidad con una amplia variedad de dispositivos, incluyendo tanto teléfonos móviles como ordenadores.
- Creación de una interfaz de usuario sencilla e intuitiva que permita a los usuarios explorar las noticias, efectuar búsquedas detalladas y acceder a las traducciones de forma ágil y eficaz.

1.3. Estado del arte

En la sección de estado del arte se explicará la solución propuesta por **OVERNEWS** y el entorno de la aplicación. Se revisarán otras aplicaciones similares existentes, comparando su funcionalidad y haciendo hincapié en los aspectos innovadores de **OVERNEWS**.

1.3.1 Solución propuesta

OVERNEWS se encuentra en el cruce de varios ámbitos innovadores como lo son la publicación automatizada de noticias, la traducción automática y el desarrollo de interfaces de usuario multilingües. A pesar de los avances importantes en estas áreas de manera individual, su integración de todos estos elementos es escasa.

La iniciativa de **OVERNEWS** sobresale por su aproximación vanguardista basada en la fusión de estas tecnologías, creando un servicio excepcional que facilita el acceso a noticias frescas y traducidas en diversos idiomas mediante una interfaz intuitiva y adaptable.

Este enfoque responde específicamente a la demanda de facilitar el acceso a la información para públicos de todo el mundo, presentando un avance considerable en comparación con las alternativas anteriores. En este proceso, **OVERNEWS** no solamente satisface esta demanda en el ámbito de los servicios informativos multilingües, sino que también mejora la experiencia del usuario evidenciando su compromiso con la innovación y la excelencia en el campo de la tecnología de la información.

1.3.2 Entorno de la aplicación

Para entender mejor la posición de OVERNEWS en el mercado, es esencial comparar sus características con las de otras aplicaciones similares. A continuación, se presentan dos ejemplos relevantes: Google News y Flipboard.

1.3.2.1 Google News



Figura 1. Google News

Google News es un agregador de noticias creado por Google que actúa como un índice dinámico de información. En esta página, puedes ver las últimas noticias de diferentes medios sobre los temas que más te interesan. Gracias a su algoritmo avanzado, Google News te ofrece contenido relevante y personalizado basado en tus intereses. [\[1\]](#)

La información se actualiza cada cinco minutos, asegurando que siempre tengas acceso a las noticias más recientes y actualizadas. Aunque Google News está disponible en múltiples idiomas (exactamente 35), no ofrece traducción automática en tiempo real de las noticias.

Google News te permite:

- Acceder a eventos de actualidad, noticias internacionales y contenido variado de distintos editores.
- Suscribirte a temas y proveedores de noticias específicos.
- Compartir artículos y añadirlos a marcadores.

1.3.2.2 Flipboard



Figura 2. Flipboard

Flipboard es una aplicación de agregación de noticias y contenido que presenta los artículos en formato de revista digital. Los usuarios pueden seleccionar temas de interés y Flipboard muestra contenido relevante de diversas fuentes. Similar a Google News, Flipboard está disponible en un total de total de 21 idiomas, lo que deja claro la magnitud de su éxito, pero no proporciona traducción automática en tiempo real. Esta aplicación posee las siguientes características:

- Lector de noticias: permite visualizar contenido de distintas fuentes.
- Organizador de contenido: además de poder leer las noticias o el contenido de interés, también es posible organizarlo y agruparlo en secciones.

- Creador de contenido: puede ser utilizado para compartir contenido con otros usuarios.
- Una plataforma o aplicación social: se concibe de esta forma debido a que permite agrupar contenido de distintas fuentes, comentarlos y compartirlos con los usuarios suscritos. [2]

1.3.2.3 Comparativa

A continuación, se muestra una tabla comparativa de las dos aplicaciones mencionadas en relación con OVERNEWS.

Característica	OVERNEWS	Google News	Flipboard
Agregación de Noticias	Sí	Sí	Sí
Personalización por Intereses	No	Sí	Sí
Traducción Automática	Sí (cada hora)	No	No
Disponibilidad Multilingüe	Sí (traducción a 7 idiomas)	Sí (35 idiomas)	Sí (21 idiomas)
Fuentes de Noticias	Múltiples medios españoles	Múltiples medios globales	Múltiples medios globales
Actualización en Tiempo Real	No (actualización cada hora)	Sí (cada cinco minutos)	Sí
Plataformas Disponibles	Web, Aplicaciones Móviles	Web, Aplicaciones Móviles	Web, Aplicaciones Móviles
Marcado de Noticias Favoritas	Sí	Sí	Sí
Comentarios sobre Noticias	Sí	No	Sí
Búsqueda por Fuentes y Categorías	Sí	Sí	Sí

Tabla 1. Comparativa Entorno de la Aplicación

OVERNEWS es la única aplicación que traduce noticias automáticamente a siete idiomas preestablecidos cada hora. No existe esta función en Google News o en Flipboard. Estas dos aplicaciones permiten a los usuarios leer noticias en muchos idiomas, pero no traducen automáticamente las noticias.

De este modo, OVERNEWS puede proporcionar a los usuarios datos confiables de las fuentes de medios españolas, en cualquier idioma que hablen, de una manera que realmente permita superar las barreras del idioma.

Además, OVERNEWS posee características adicionales que hacen que la experiencia del usuario sea aún mejor. Las noticias que son de especial interés para el usuario pueden ser marcadas para que el usuario las archive y luego, vaya directamente a estas noticias de mayor

interés. Tanto Google News, como Flipboard tienen esta característica de marcar las noticias como favoritas. Otra característica interesante de OVERNEWS es que, según lo previsto, permitirá a los usuarios comentar las noticias. Esto no solo es un medio de interacción que proporciona a las personas la oportunidad de conversar y compartir sus opiniones entre sí, sino que también es una comunidad relativamente más activa y abierta. Google News no tiene esta característica, por lo que, en este sentido, OVERNEWS sobresale por encima de la competencia.

En la función de búsqueda y filtrado, OVERNEWS también permite la búsqueda de noticias específicas por fuente y categoría, como Google News y Flipboard. Otro punto interesante es la interfaz de usuario de OVERNEWS, que está especialmente diseñada para tener una interfaz multilingüe intuitiva. Esto hace factible que cualquier persona, con o sin dominio del castellano, lea las noticias locales, una funcionalidad que Google News y Flipboard no soportan. Mientras que el primero se centra más en la personalización, y el segundo es más visual, OVERNEWS está comprometido con ofrecer una experiencia de usuario que supere las posibles barreras lingüísticas.

1.4 Alcance (Árbol de características)

Este proyecto está diseñado para automatizar la recopilación de noticias y su posterior traducción. Es por ello por lo que he seleccionado cinco de los medios digitales más relevantes en España. La elección de estas fuentes se ha realizado con el objetivo de ofrecer una visión equilibrada y diversa de la actualidad española.

Adicionalmente, la aplicación se dedica a convertir el contenido de estas noticias de español a siete lenguas distintas, elegidas conforme a su importancia para el público objetivo y las necesidades lingüísticas de los turistas en España. Esta selección inicial garantiza que la plataforma sea accesible y útil para la mayoría de los interesados.

Además, se ha diseñado OVERNEWS con una arquitectura y tecnología que permite una fácil escalabilidad, tanto en la adición de nuevas fuentes de noticias como en la expansión a más idiomas, en el futuro.

A continuación, se muestra el árbol de características, que describe más detalladamente el alcance del proyecto.

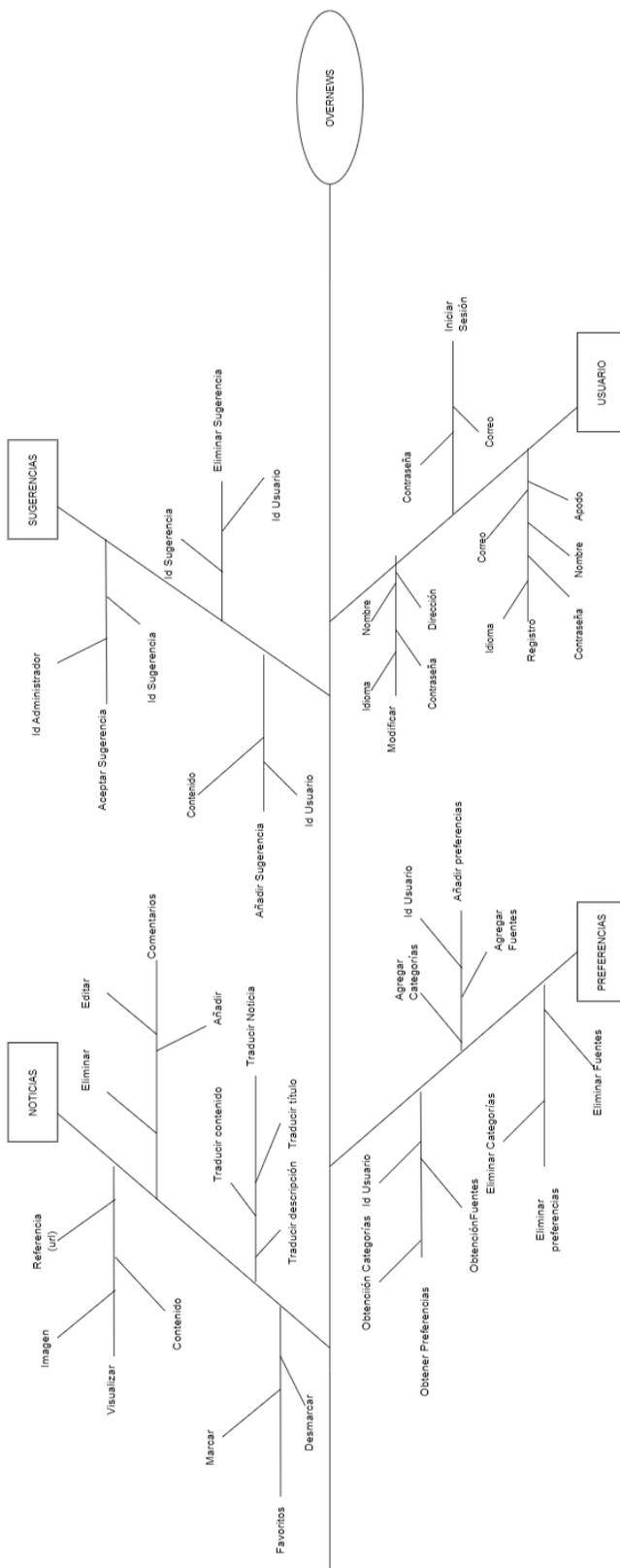


Figura 3. Árbol de características

1.5 Entorno y tecnologías utilizadas

El desarrollo de OVERNEWS se fundamenta en una arquitectura claramente diferenciada entre el backend y el frontend, con el backend funcionando como una API REST.

Las API (*Application Programming Interfaces*) permiten que servicios se comuniquen con otros, sin necesidad de saber cómo están implementados. REST (*Representational State Transfer*) es un conjunto de principios que definen la forma en que se deben crear, leer, actualizar y eliminar los datos.

Esta separación garantiza una mayor flexibilidad y escalabilidad, permitiendo que tanto el *backend* como el *frontend* evolucionen de forma independiente según sus propias necesidades y tecnologías. A continuación, se presenta un desglose de las tecnologías clave seleccionadas para el proyecto

1.5.1. Backend

1.5.1.1 POSTGRESQL



Figura 4. PostgreSQL

POSTGRESQL es un gestor de bases de datos altamente competente y de código abierto, bajo una licencia BSD que permite su uso tanto en aplicaciones personales como comerciales. Considerado un sistema de nivel empresarial, POSTGRESQL puede rivalizar en funcionalidad y rendimiento con sistemas de bases de datos comerciales, incluso con gigantes como Oracle. Aunque no es el más popular, POSTGRESQL posee ventajas significativas a nivel de funcionalidad, destacando su habilidad para manejar grandes volúmenes de datos, como en su modelo de licencia. A diferencia de MySQL, POSTGRESQL ofrece una licencia completamente abierta para cualquier tipo de uso. [3]

Características de POSTGRESQL:

- Alta disponibilidad: POSTGRESQL opera de forma estable y continua en el servidor, destacándose por su robustez, consistencia y tolerancia a fallos.
- Concurrencia elevada: Maneja múltiples solicitudes de clientes simultáneamente, garantizando acceso a datos sin bloqueos.
- Diversidad de tipos de datos: Soporta tipos de datos comunes y otros más especializados como direcciones IP, direcciones MAC, *arrays*, números decimales

- de precisión configurable, figuras geométricas, entre otros.
- Soporte a *triggers* y vistas: Permite la creación de eventos y acciones automáticas, así como la consulta de datos de formas variadas.
- Modelo objeto-relacional: Facilita el manejo de datos como objetos e incluye características de orientación a objetos como la herencia de tablas.
- Bases de datos distribuidas: Asegura la integridad de las transacciones distribuidas a través de múltiples sistemas.
- Compatibilidad con múltiples lenguajes de programación: Funciona con funciones internas en el servidor en lenguajes como C, C++, Java, PHP, Python o Ruby, y proporciona interfaces para ODBC, JDBC y una variedad de lenguajes de programación.

Este conjunto de características hace que PostgreSQL sea una opción formidable para que OVERNEWS tenga un sistema de gestión de bases de datos fiable y versátil. [\[4\]](#)

1.5.1.2 FASTAPI



Figura 5. FastAPI

FASTAPI es un *framework* moderno para el desarrollo de APIs en Python, reconocido por su velocidad y eficacia. La generación automática de documentación y el soporte para operaciones asincrónicas, apoyadas por un sistema de tipado fuerte, hacen que sea un *framework* de fácil usabilidad y eficiente.

Las principales características de FASTAPI son:

- Alto rendimiento: FASTAPI se compara favorablemente con tecnologías consolidadas como Node.js, gracias al uso de STARLETTE para la gestión asincrónica y PYDANTIC para la validación de datos. Estas herramientas posicionan a FASTAPI como uno de los *frameworks* más rápidos para Python.
- Eficiencia en el Desarrollo: Este *framework* aumenta significativamente la productividad, permitiendo a los desarrolladores enfocarse en la lógica del negocio más que en la infraestructura subyacente del código. Se estima que usando este *framework* se puede duplicar e incluso triplicar la velocidad de desarrollo.
- Reducción de errores de programación: FASTAPI reduce los errores de programación hasta en un 40%, lo que reduce el tiempo de desarrollo de la aplicación.
- Minimización de código duplicado: Promueve la reutilización de código, disminuyendo la redundancia y los posibles errores mediante un diseño que maximiza la eficacia de cada declaración de parámetros.
- Documentación y preparación para la producción: Incluye herramientas para la creación de documentación automática interactiva, lo que facilita la colaboración y mejora el mantenimiento a largo plazo.

- Adhesión a Estándares: FastAPI es completamente compatible con estándares abiertos para APIs como OpenAPI y JSON *Schema*, asegurando una integración y extensibilidad óptimas en diversos entornos. [5]

1.5.2. Frontend

1.5.2.1 Node.js



Figura 6. Node.js

Node.js es un entorno de ejecución para JavaScript que permite a los desarrolladores escribir y ejecutar código en el servidor, no solo en el navegador. Esto ha revolucionado la forma en que se desarrollan las aplicaciones web, al permitir un enfoque unificado para el desarrollo tanto del *frontend* como del *backend* utilizando un solo lenguaje de programación.

Las principales características de Node.js son:

- Reducción del tiempo de comercialización: Node.js es reconocido por su ligereza y eficiencia, lo que permite a los desarrolladores transformar ideas en productos funcionales rápidamente. Su capacidad para mantener un alto rendimiento mientras reduce los costos y los tiempos de desarrollo lo convierte en una opción atractiva para startups y empresas que desean lanzar sus productos al mercado de manera eficiente.
- Escalabilidad: Su estructura permite adaptarse fácilmente a mayores demandas, agregando más módulos o servicios según sea necesario, sin comprometer la reactividad ni el rendimiento.
- Versatilidad en el desarrollo: Node.js es aplicable tanto para el desarrollo del *frontend* como del *backend*. Esta flexibilidad es beneficiosa para los proyectos que utilizan marcos como React.js, ya que simplifica la gestión de recursos humanos al permitir que los desarrolladores trabajen a lo largo de toda la pila tecnológica.
- Ecosistema: Con NPM (*Node Package Manager*), Node.js ofrece el repositorio de software de más rápido crecimiento en el mundo. NPM alberga una vasta biblioteca de paquetes reutilizables que facilitan el desarrollo rápido y eficiente.

La integración de Node.js en proyectos que utilizan React.js es particularmente poderosa, ya que permite un flujo de desarrollo coherente y optimizado. Esta combinación potencia la eficiencia en la construcción de aplicaciones interactivas y dinámicas, haciendo uso completo de las capacidades modernas de desarrollo web. [6]

1.5.2.2 React.js



Figura 7. ReactJS

React.js es una biblioteca de JavaScript diseñada para crear interfaces de usuario de forma eficiente y con un enfoque declarativo. A continuación, se describen las diferentes características que facilitan el desarrollo rápido y escalable de aplicaciones web complejas y dinámicas. [7]

- **Reutilización de componentes:** React.js fomenta una estructura donde los componentes pueden ser utilizados repetidamente en diferentes secciones de una aplicación. Este diseño no solo simplifica y agiliza el proceso de desarrollo al disminuir la cantidad de código que necesita ser escrito, sino que también mejora la mantenibilidad y escalabilidad del software. Con React.js, los desarrolladores pueden construir aplicaciones robustas con mayor eficiencia y coherencia.
- **Eficiencia de rendimiento:** la biblioteca de JavaScript adopta un enfoque declarativo para la programación, lo que simplifica el proceso de desarrollo y mejora la predictibilidad del código. React.js maneja el flujo de datos y la actualización de la interfaz de usuario de manera eficiente gracias a su algoritmo de reconciliación, conocido como *Virtual DOM (Document Object Model)*. Gracias a este enfoque se minimizan las operaciones costosas en el DOM real, proporcionando una mejora significativa en el rendimiento.
- **Flexibilidad:** otro aspecto para tener en cuenta es su compatibilidad, con una amplia gama de bibliotecas y marcos de JavaScript, lo que le permite integrarse sin problemas en proyectos ya existentes.
- **Comunidad activa:** React.js se beneficia de una comunidad extensa y dinámica de desarrolladores, que continuamente aportan al ecosistema a través de nuevas bibliotecas, herramientas y plugins. Esta riqueza de recursos y soporte otorga una base sólida para el desarrollo profesional y la innovación continua en el campo del desarrollo web.

1.5.2.3 MUI (Material-UI)



Figura 8. Material-UI

Material-UI es una herramienta poderosa para diseñar interfaces de usuario en proyectos React. Inspirado en *Material Design* de Google, facilita la creación de aplicaciones con un diseño cohesivo y atractivo, ofreciendo amplias opciones de personalización para adaptarse a las necesidades específicas del proyecto. [8]

- Estética unificada y atractiva: *Material-UI* trae consigo un conjunto de componentes visualmente atractivos que siguen los principios de *Material Design*. Esto no solo asegura que tu aplicación tenga un aspecto moderno y profesional, sino que también garantiza la coherencia visual a lo largo de toda la interfaz, mejorando la experiencia del usuario.
- Personalización a medida: A pesar de que *Material-UI* ofrece estilos predeterminados, brinda una gran flexibilidad para adaptar cada componente a tus necesidades específicas. Puedes modificar colores, tipografías y otros elementos para que se alineen perfectamente con la identidad de tu marca o para responder a requisitos particulares del proyecto.
- Eficiencia en el desarrollo: Con su extenso catálogo de componentes listos para integrar, *Material-UI* te permite agilizar el desarrollo de tu interfaz. Esta biblioteca reduce el tiempo que normalmente invertirías en diseñar y codificar elementos comunes de la UI (*User Interface*), permitiéndote concentrarte en desarrollar la lógica y funcionalidades únicas de tu aplicación.
- Adaptabilidad automática: Los componentes de *Material-UI* están diseñados para responder y adaptarse a diferentes tamaños de pantalla, lo que asegura que tu aplicación funcione de manera óptima tanto en dispositivos móviles como en ordenadores. Esta responsividad automática es crucial para alcanzar y mantener una buena usabilidad en el amplio espectro de dispositivos utilizados por los usuarios hoy en día.

1.5.3 Sistema de control de versiones

1.5.3.1 *Git*



Figura 9. *Git*

Git es un sistema de control de versiones distribuido que ha transformado la forma en que gestionamos el código fuente. Permite llevar un registro completo de todas las modificaciones, facilitando el trabajo individual y colaborativo. [9]

Las principales características de *Git* son:

- Colaboración eficiente: *Git* me ha permitido trabajar en distintas partes del proyecto de forma independiente, creando ramas para cada nueva funcionalidad y fusionándolas sin problemas cuando estaban listas.
- Gestión de versiones: Mantener un historial detallado de los cambios ha sido crucial para revertir errores y mejorar el código progresivamente.
- Rendimiento rápido: *Git* maneja operaciones como *commits* y *merges* de manera rápida y eficiente, lo que ha acelerado significativamente mi flujo de trabajo.

1.5.3.2 GitHub



Figura 10. *GitHub*

Para alojar y gestionar el repositorio de **OVERNEWS**, he utilizado *GitHub*, una plataforma web que complementa a *Git* que facilita el desarrollo y la colaboración, incluso cuando se trabaja solo. [10]

Las principales características de *GitHub* son:

- Repositorio centralizado: *GitHub* ha servido como un lugar seguro y accesible para almacenar todo el código del proyecto. De esta forma siempre tengo acceso a las versiones anteriores y actuales del código.
- Revisiones y automatización: Con los *pull requests* y *GitHub Actions*, puedo revisar cambios de manera estructurada, automatizar tareas repetitivas y revertir cambios si al desarrollar tengo errores.
- Documentación del proyecto: *GitHub* facilita la creación y mantenimiento de

documentación a través de su sistema de *wikis* y archivos README, lo que ha sido esencial para organizar y explicar las funcionalidades del proyecto de manera clara y accesible.

La combinación de Git y GitHub ha sido fundamental para el desarrollo de OVERNEWS, permitiéndome gestionar el código de manera eficiente, realizando un seguimiento claro de las diferentes tareas a implementar.

1.6 Estructura de la memoria

En esta sección, se presentará cómo va a ser la estructura de la memoria.

- **Capítulo 1. Descripción del Proyecto:** Introducción al proyecto OVERNEWS, sus objetivos, el estado del arte en la tecnología de agregadores de noticias, el entorno de la aplicación y una visión general de las características y tecnologías utilizadas.
- **Capítulo 2. Planificación:** Explicación de la metodología de trabajo utilizada, la planificación temporal del proyecto, el análisis de riesgos, el presupuesto estimado y el seguimiento real del desarrollo.
- **Capítulo 3. Análisis:** Descripción de los requisitos del usuario y del sistema, los casos de uso detallados, y los requisitos funcionales y no funcionales necesarios para el correcto funcionamiento de OVERNEWS.
- **Capítulo 4. Diseño:** Presentación de la arquitectura lógica y física de la aplicación, junto con diagramas de secuencia y actividad, y el modelo de datos, incluyendo el modelo entidad-relación, el diseño lógico y físico, y el diseño de las interfaces de usuario.
- **Capítulo 5. Implementación:** Detalle de la estructura del proyecto, explicando la implementación del backend, frontend y la API de traducción, junto con las cuestiones de implementación reseñables, como la gestión de datos persistentes y temporales, y la autenticación y autorización.
- **Capítulo 6. Pruebas:** Descripción de las pruebas de caja blanca y caja negra realizadas para asegurar la funcionalidad de OVERNEWS, junto con los resultados obtenidos.
- **Capítulo 7. Conclusiones:** Reflexión sobre el desarrollo del proyecto, los logros alcanzados y las dificultades superadas, junto con las líneas de trabajo futuras para seguir mejorando y ampliando las funcionalidades de OVERNEWS.

- **Webgrafía:** Lista de las fuentes y referencias utilizadas durante el desarrollo del proyecto.
- **Anexos:** Presentación de los manuales de uso y de despliegue de la aplicación.

Capítulo 2

Planificación

2.1. Metodología de trabajo

La selección del modelo en espiral para el desarrollo de OVERNEWS fue motivada por la naturaleza compleja y dinámica del proyecto, así como por la necesidad de abordar adecuadamente los riesgos y cambios en los requisitos a lo largo del tiempo. Este modelo es ideal para este tipo de proyectos, donde las innovaciones tecnológicas y las fluctuaciones influir significativamente en la dirección del desarrollo.

¿Qué es el modelo en espiral?

El modelo en espiral de Boehm es un modelo de desarrollo de software tradicional que integra elementos de los modelos en cascada, incrementales y ágiles, destacándose por su enfoque en el análisis y gestión de riesgos. Este modelo evolutivo organiza el desarrollo de software en cuatro cuadrantes: planificación, análisis de riesgos, implementación y evaluación. Cada cuadrante constituye un ciclo de actividades que se repite continuamente hasta que el software cumple con los requisitos y expectativas de las partes interesadas. [11]

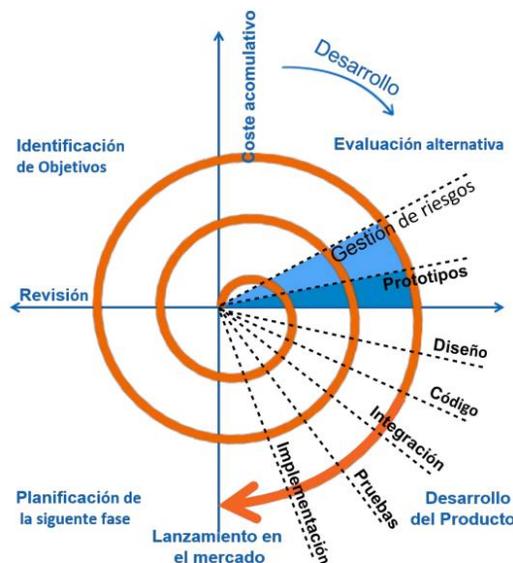


Figura 11. Gráfico Modelo en Espiral

A medida que estos ciclos se completan, se desarrollan prototipos sucesivos que se van refinando y mejorando con cada iteración. Esto permite que el producto vaya ganando madurez y funcionalidad de forma controlada y estructurada.

El modelo en espiral fue seleccionado para este proyecto debido a los riesgos significativos asociados con su naturaleza compleja y de gran escala. Este enfoque permite abordar los riesgos de manera cíclica y continua a lo largo del ciclo de vida del proyecto, lo que contribuye a una mayor madurez y refinamiento del producto final.

El modelo en espiral es incremental y está diseñado para iterar únicamente cuando existen riesgos, conflictos o errores que amenazan el desarrollo del proyecto (Smith, 2020). Cada iteración del modelo de Boehm se estructura en torno a una espiral, con cada vuelta representando un conjunto de actividades específicas. Estas actividades se dividen en las siguientes fases:

- **Planeación o planificación:** En esta fase se definen los objetivos del proyecto, así como las restricciones y alternativas disponibles. Este ciclo comienza con una investigación preliminar que establece una base sólida para el desarrollo posterior.
- **Análisis de riesgo:** Aquí se evalúan todos los posibles riesgos que puedan afectar el proyecto. Esta evaluación incluye el diseño y validación de prototipos para identificar y mitigar riesgos durante el ciclo de desarrollo.
- **Implementación:** Esta fase abarca el desarrollo del software, incluyendo la codificación y las pruebas necesarias para validar su correcto funcionamiento. Es crucial para asegurar que el software cumple con los requisitos establecidos y está libre de errores significativos.
- **Evaluación:** La última fase del ciclo se centra en analizar los riesgos identificados y verificar si se han encontrado soluciones viables. Esta fase es esencial para determinar el progreso del proyecto y para guiar el enfoque de la próxima iteración, asegurando que el desarrollo se mantenga en el camino correcto.

2.2. Planificación del proyecto

En esta sección se presentará la planificación inicial del proyecto **OVERNEWS**, así como la distribución final del tiempo dedicado a cada fase del desarrollo.

2.2.1. Planificación temporal inicial

En esta sección se detalla la planificación organizada para la realización del Trabajo Fin de Grado, cuya carga es de 12 ECTS. A continuación, se presenta la distribución del trabajo a lo largo de los diferentes incrementos, tomando en cuenta los posibles riesgos que podrían extender la duración de las iteraciones según las necesidades y contratiempos que surjan durante su desarrollo.

Dado el carácter complejo del proyecto **OVERNEWS** con su robusta arquitectura, la necesidad de traducir noticias automáticamente a múltiples idiomas, y la gestión segura y eficiente de usuarios y datos, se han planificado cuatro incrementos. Cada uno de estos incrementos incluirá las iteraciones necesarias para asegurar el desarrollo exitoso del Trabajo Fin de Grado.

Este TFG está programado para comenzar a mediados de noviembre, y desde entonces, se avanzará en cada uno de los incrementos hasta la fecha límite en abril. La carga de trabajo inicial estimada es de 300 horas, según lo estipulado en el plan de estudios. Sin embargo, esta podría incrementarse a 380 horas debido a complicaciones y riesgos anticipados. Este margen adicional de 80 horas permitirá gestionar imprevistos y asegurará que cualquier posible retraso no afecte la entrega final del proyecto.

La distribución de horas entre los cuatro incrementos se planifica de la siguiente manera: 111 horas para el primer incremento, 90 horas para el segundo incremento, 90 horas para el tercer incremento y 90 horas para el último incremento. Un total de 381 horas, que se asemejan con lo estipulado en el plan de estudios, y que se desarrollará en un periodo de 3 meses.

A continuación, se detallan las tareas específicas de cada incremento, proporcionando una guía clara y estructurada para la creación del diagrama de Gantt en Microsoft Project 2010. Hay que tener en cuenta que 1 día en el diagrama de Gantt va a equivaler a 6 horas, ya que es el tiempo que le voy a emplear cada día.

2.2.1.1. Incremento I: Fundamentos del Proyecto

El primer incremento se enfoca en establecer las bases sólidas del proyecto, con una duración total de 19 días (114 horas). Las tareas incluyen la definición de objetivos y alcance, análisis de requisitos, configuración inicial del entorno de desarrollo, y diseño de la arquitectura del sistema. También se configuran los repositorios en GitHub y se crea la documentación inicial. Este incremento es crucial que se desarrolle de forma adecuada ya que establecerá los pilares para los siguientes incrementos.

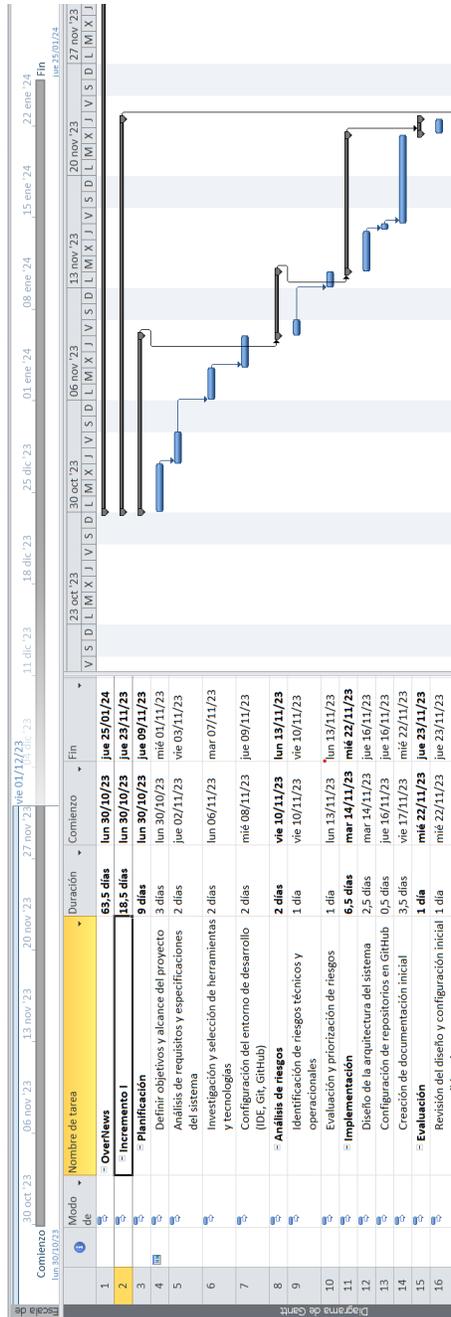


Figura 12. Diagrama de Gantt (Incremento I)

2.2.1.2. Incremento II: *Web Scraping* y Almacenamiento

El segundo incremento aborda la implementación de la funcionalidad de *web scraping* y el almacenamiento de noticias. En esta fase, se desarrollan los scripts necesarios para extraer datos de diferentes fuentes de noticias utilizando *BeautifulSoup* y se configura la base de datos *PostgreSQL* para almacenar estos datos. La integración entre el *scraping* y el almacenamiento es fundamental para asegurar que las noticias se recojan y almacenen de manera eficiente y segura.



Figura 13. Diagrama de Gantt (Incremento II)

2.2.1.3. Incremento III: Traducción Automática y Gestión de Usuarios

El tercer incremento se implementarán la funcionalidad para traducir las noticias a varios idiomas utilizando una API de traducción y se desarrolla el sistema de registro y autenticación de usuarios. Además, se incluye la funcionalidad para la selección de idioma,

permitiendo a los usuarios personalizar su experiencia de acuerdo con sus preferencias lingüísticas.

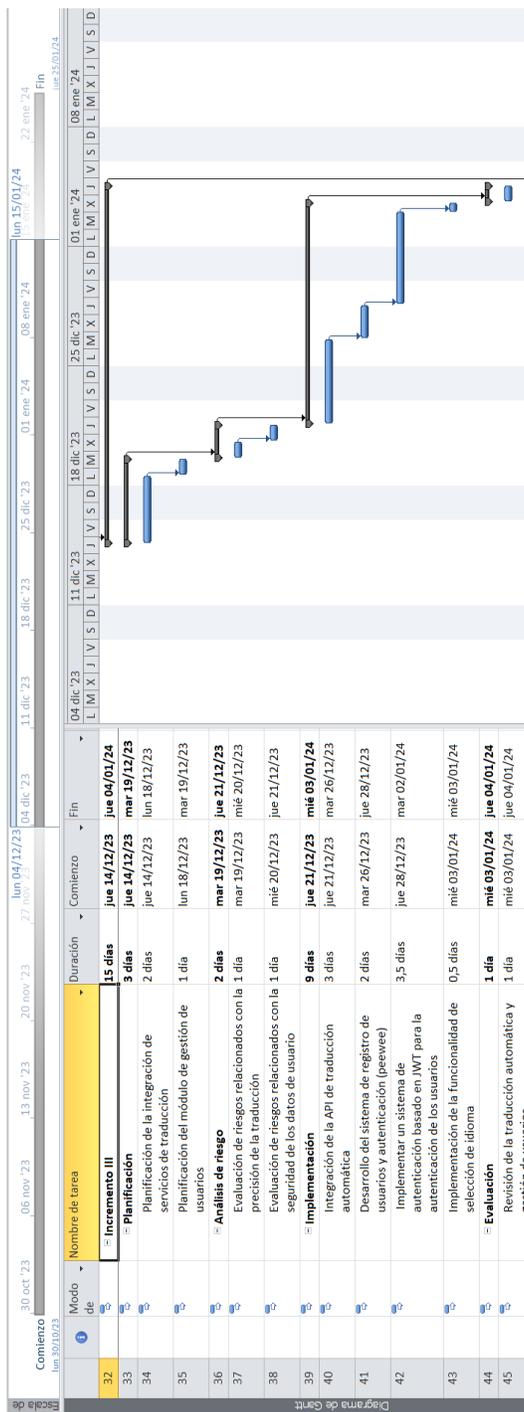


Figura 14. Diagrama de Gantt (Incremento III)

2.2.1.4. Incremento IV: Interfaz de Usuario

El cuarto y último incremento se dedica al desarrollo de la interfaz de usuario y la

implementación de funcionalidades de personalización. Utilizando React.js y MUI, se diseña una interfaz intuitiva y adaptativa (responsive) que permite a los usuarios navegar fácilmente por las noticias. También se agregan funciones para que los usuarios puedan marcar noticias como favoritas y comentar sobre ellas, así como personalizar el contenido según su idioma preferido.



Figura 15. Diagrama de Gantt (Incremento IV)

2.2.2. Análisis de riesgos

En la planificación inicial del proyecto **OVERNEWS**, es fundamental considerar que pueden surgir diversos problemas que provoquen retrasos en la primera ejecución del proyecto, e incluso que imposibiliten su finalización en el plazo establecido. Por ello, es necesario identificar los posibles riesgos, evaluar la probabilidad de su ocurrencia y determinar el impacto que podrían tener en el proyecto.

El costo del proyecto se medirá en términos de tiempo, calculando el número de días de retraso que podrían causar en caso de materializarse. Este análisis de riesgos se realizará para todas las iteraciones, como se mencionó anteriormente en la sección 2.1.

En las diferentes iteraciones, se procederá a identificar los riesgos potenciales, la probabilidad de su ocurrencia y el impacto en días que podrían ocasionar en el proyecto. Partiendo del peor de los escenarios posibles, se realizará un análisis detallado para prever y mitigar cualquier eventualidad que pueda comprometer el progreso y la finalización exitosa de **OVERNEWS**.

Primero, se determinarán los posibles riesgos que podrían presentarse durante el desarrollo del proyecto. Luego, se evaluará la probabilidad de que ocurran y el impacto en días que cada uno podría generar. Este enfoque nos ayudará a tomar acciones preventivas para reducir los retrasos y garantizar que el proyecto avance según lo previsto, finalizando dentro del tiempo estipulado.

Para llevar a cabo la evaluación del riesgo, se han considerado las siguientes escalas de probabilidad e impacto:

Probabilidad			
		\geq	$<$
5	Casi cierto	0.80	1.00
4	Probable	0.50	0.80
3	Posible	0.20	0.50
2	Improbable	0.10	0.20
1	Raro	0.01	0.10

Tabla 2. Escala de probabilidad

Impacto		
1	Insignificante	valor $< 1\%$
2	Menor	$1\% \leq \text{valor} < 5\%$
3	Moderado	$5\% \leq \text{valor} < 10\%$
4	Mayor	$10\% \leq \text{valor} < 20\%$
5	Catastrófico	$20\% \leq \text{valor}$

Tabla 3. Escala de impacto

A continuación, se muestra una tabla en dónde se identifican y evalúan los diferentes riesgos a tener en cuenta a la hora de elaborar el proyecto de **OVERNEWS**.

Riesgo	Descripción	Probabilidad	Impacto	Clasificación Probabilidad	Clasificación Impacto	Costo en días
1	Las páginas de las cuales se extraen las noticias pueden cambiar su estructura o dejar de estar disponibles, lo que requeriría modificaciones en los <i>scripts</i> de <i>web Scraping</i>	80%	5%	Casi cierto (5)	Moderado (3)	3
2	La API de traducción utilizada puede experimentar problemas técnicos, afectando la traducción automática de noticias.	30%	4%	Posible (3)	Menor (2)	4
3	Pueden ocurrir errores en la configuración o en las consultas de la base de datos, lo que afectaría el almacenamiento y recuperación de noticias.	30%	3%	Posible (3)	Menor (2)	3
4	Podrían surgir vulnerabilidades de seguridad en el sistema de gestión de usuarios, comprometiendo la protección de datos personales.	40%	3%	Posible (3)	Menor (2)	4
5	Los requisitos del proyecto pueden cambiar durante su desarrollo, requiriendo ajustes en el plan inicial y en las funcionalidades implementadas	90%	10%	Casi cierto (5)	Mayor (4)	7
6	La integración entre los diferentes componentes del sistema (<i>web Scraping</i> , traducción, almacenamiento y <i>frontend</i>) podría presentar problemas de compatibilidad	50%	6%	Probable (3)	Moderado (3)	3
7	Podría haber alteraciones en las circunstancias personales que dificulten la dedicación prioritaria al avance del proyecto	50%	15%	Probable (3)	Mayor (4)	10

Tabla 4. Análisis de Riesgos

Para solventar estos riesgos de manera efectiva, se muestran los planes de mitigación y

reducción diseñados para cada riesgo. Estas medidas permiten anticipar posibles problemas y establecer estrategias para minimizar su impacto en el proyecto.

Riesgo	Exposición	Plan de Reducción	Plan de Mitigación
1	15	Monitoreo continuo de cambios en las páginas de noticias	Actualizar y ajustar <i>scripts</i> de <i>Scraping</i> de la forma más rápida y eficiente
2	6	Utilizar una API de traducción secundaria	Implementar una lógica de respaldo en caso de fallo
3	6	Realizar pruebas exhaustivas y seguir buenas prácticas de configuración	Restaurar desde la última copia de seguridad disponible
4	6	Implementar autenticación JWT y cifrado de datos	Resolver problemas de compatibilidad rápidamente mediante pruebas iterativas
5	20	Mantener documentación clara y realizar revisiones periódicas de requisitos	Ajustar el plan del proyecto y redistribuir tareas rápidamente
6	9	Realizar pruebas de integración frecuentes	Identificar y resolver problemas de compatibilidad rápidamente, ajustando el código según sea necesario
7	12	Crear un calendario flexible que permita ajustes según sea necesario	Priorizar tareas y ajustar plazos según las circunstancias personales

Tabla 5. Plan de Mitigación y Reducción de Riesgos

2.2.3. Presupuesto

En esta sección se detallará el presupuesto económico estimado para el proyecto OVERNEWS, teniendo en cuenta tanto los recursos humanos como los recursos técnicos necesarios. Para estimar el presupuesto del proyecto de la manera más precisa posible, se emplearán dos métodos complementarios: el método de Albrecht y la estimación de esfuerzos basada en la planificación temporal.

2.2.3.1 Estimación mediante método Albrecht

El método de Albrecht se utiliza para cuantificar las funcionalidades de un sistema a través de puntos de función. Estos puntos de función proporcionan una medida cuantitativa y objetiva del tamaño de una aplicación. Este enfoque es muy popular en la industria del software debido a su simplicidad y la fiabilidad de sus resultados. La fiabilidad se deriva del hecho de que los puntos de función son valores cuantificables basados en un conjunto de métricas que permiten gestionar la productividad, el coste, la calidad y la complejidad del software. [12]

El método de Albrecht divide el sistema en los siguientes componentes:

- **Entradas externas:** Estas son todas las funcionalidades que permiten introducir datos en el sistema. Esto incluye tanto los datos ingresados por los usuarios como aquellos provenientes de otros sistemas, como podría ser un proceso de aprovisionamiento.
- **Salidas externas:** Estas comprenden todos los datos que salen del sistema, principalmente como resultado de la ejecución de algoritmos. Los datos de salida suelen generar archivos que pueden ser utilizados como entradas para otras aplicaciones.
- **Consultas externas:** Las consultas incluyen tanto entradas como salidas. Las entradas se utilizan para seleccionar datos de archivos lógicos, ya sean internos o externos, y las salidas son los datos recuperados que fueron solicitados.
- **Ficheros lógicos internos:** Estos son conjuntos de datos relacionados lógicamente que se almacenan dentro de la aplicación misma.
- **Ficheros lógicos externos:** Estos son conjuntos de datos relacionados lógicamente que se almacenan fuera de los límites de la aplicación.

El uso del método de Albrecht y sus puntos de función permite una gestión eficaz del desarrollo de software, asegurando que se puedan medir de manera precisa las diferentes dimensiones del proyecto y facilitar una planificación adecuada.

Cálculo de los puntos de función no ajustados (PFNA)

Para realizar el cálculo de los puntos de función no ajustados (PFNA), se deben evaluar los componentes mencionados anteriormente en términos de su complejidad. Cada componente se clasifica como simple, medio o complejo, y se asigna un valor de puntos de función basado en esta clasificación.

❖ Entradas externas:

Tipos de ficheros	Datos elementales		
	1-4	5-15	>16
0-1	Baja	Baja	Media
2-3	Baja	Media	Alta
>3	Media	Alta	Alta

Tabla 6. Complejidad entradas externas PFNA

A continuación, se presenta una tabla que detalla las diferentes entradas externas de la aplicación, junto con sus correspondientes niveles de complejidad.

Entradas externas	Tipos de ficheros	Datos elementales	Complejidad
Iniciar Sesión	0-1	1-4	Baja
Registro Usuario	0-1	5-15	Baja
Escribir un	0-1	1-4	Baja

comentario			
Editar un comentario	0-1	1-4	Baja
Modificar preferencias de selección de noticias	>3	1-4	Media
Modificar datos usuarios	0-1	5-15	Baja
Solicitar fuente al administrador	0-1	1-4	Baja
Marcado de noticias como favoritas	0-1	1-4	Baja
Contactar a través de Gmail	0-1	1-4	Baja
Filtrar noticias según sus preferencias	0-1	5-15	Baja
Obtener noticias (web Scraping)	0-1	>16	Media

Tabla 7. Entradas externas

❖ Salidas externas:

Tipos de ficheros	Datos elementales		
	1-4	5-15	>16
0-1	Baja	Baja	Media
2-3	Baja	Media	Alta
>3	Media	Alta	Alta

Tabla 8. Complejidades salidas y consultas externas PFNA

A continuación, se presenta una tabla que detalla las diferentes salidas externas de la aplicación, junto con sus correspondientes niveles de complejidad.

Salidas externas	Tipos de ficheros	Datos elementales	Complejidad
Número de usuarios	>3	1-4	Baja
Número de noticias traducidas	0-1	1-4	Baja
Número de visitas de una noticia	0-1	1-4	Baja

Número total de visitas	0-1	1-4	Baja
Número de noticias guardadas	0-1	1-4	Baja
Número de comentarios de un usuario	0-1	1-4	Baja
Contenido de las noticias solicitadas	>3	5-15	Alta

Tabla 9. Salidas externas

❖ Consultas externas:

Para calcular la complejidad de las consultas externas se usa las mismas reglas que en la Tabla de Salidas externas PFNA.

En la siguiente tabla se muestran las diferentes consultas externas que se deberán de desarrollar clasificadas según su complejidad.

Consultas externas	Tipos de ficheros	Datos elementales	Complejidad
Obtener noticias traducidas almacenadas	>3	5-15	Alta
Obtener todo el contenido de una noticia	0-1	5-15	Baja
Obtener noticias favoritas	>3	5-15	Alta
Obtener preferencias del usuario	2-3	1-4	Baja
Obtener comentarios de un usuario	0-1	5-15	Baja
Obtener comentarios de una noticia	0-1	1-4	Baja
Obtener solicitudes de fuentes	0-1	1-4	Baja
Obtener usuario	0-1	6	Baja
Obtener sugerencias	0-1	1-4	Baja

Tabla 10. Consultas externas

❖ Ficheros lógicos internos:

Tipos de ficheros	Datos elementales		
	1-19	20-50	>51
1	Baja	Baja	Media
2-5	Baja	Media	Alta
>6	Media	Alta	Alta

Tabla 11. Complejidad ficheros lógicos internos y externos PFNA

En la siguiente tabla se muestran los diferentes ficheros lógicos internos catalogados, según su complejidad, que se deberán de desarrollar para desarrollar correctamente la aplicación.

FLI	Tipos de ficheros	Datos elementales	Complejidad
Tabla usuarios	1	1-19	Baja
Tabla de noticias	>6	1-19	Media
Tabla categorías	1	1-19	Baja
Tabla fuentes	1	1-19	Baja
Tabla comentarios	1	1-19	Baja
Tabla noticias favoritas	1	1-19	Baja
Tabla preferencias de las noticias	1	1-19	Baja
Tabla sugerencias	1	1-19	Baja

Tabla 12. Ficheros lógicos internos

❖ Ficheros lógicos externos

Para clasificar los diferentes ficheros lógicos externos que tiene la aplicación se haría uso de la tabla de Complejidad ficheros lógicos internos y externos PFNA.

En esta versión no se contemplan ficheros lógicos externos, por lo tanto, no lo tendremos en cuenta en el cálculo de los PFNA.

Componente	Complejidad	Ponderación	Total por Tipo	Total x Complejidad
Entradas externas	Baja	x3	9	27
	Media	x4	2	8
	Alta	x6	0	0
Salidas externas	Baja	x4	6	24
	Media	x5	0	0
	Alta	x7	1	7
Consultas externas	Baja	x3	7	21
	Media	x4	0	0
	Alta	x6	2	12
Ficheros lógicos internos	Baja	x7	7	49
	Media	x10	1	10
	Alta	x15	0	0

Ficheros lógicos externos	Baja	x5	0	0
	Media	x7	0	0
	Alta	x10	0	0
Total PFNA				158

Tabla 13. Tabla Ponderaciones PFNA

Cálculo del factor de ajuste (FA)

Para estimar con precisión el esfuerzo necesario para desarrollar OVERNEWS, se han evaluado diversos factores de ajuste que influyen en la complejidad del sistema.

Atributos	Influencia
Comunicación de datos	4
Funciones distribuidas	4
Prestaciones	3
Gran uso de la configuración	2
Velocidad de las transacciones	3
Entrada on-line de datos	5
Diseño para la eficiencia del usuario final	3
Actualización de datos on-line	4
Complejidad del proceso lógico interno de la aplicación	4
Reusabilidad del código	2
Facilidad de instalación	2
Facilidad de operación	2
Localizaciones múltiples	1
Facilidad de cambios	2
Total	42

Tabla 14. Factores de Ajuste de Valor (FAV)

A continuación, se destacan los atributos con mayor influencia y se explica brevemente el motivo de su relevancia.

- **Comunicación de datos:** La aplicación requiere una intensa comunicación de datos entre sus componentes y servicios, incluyendo la recopilación, traducción y presentación de noticias.
- **Funciones distribuidas:** El proyecto utiliza múltiples servicios distribuidos para manejar datos de diversas fuentes de noticias, lo que aumenta significativamente la complejidad del sistema.
- **Entrada on-line de datos:** OVERNEWS permite la entrada de datos en tiempo real para el registro de usuarios, configuración de preferencias y marcación de noticias favoritas, lo que es fundamental para su funcionalidad.
- **Actualización de datos on-line:** La necesidad de actualizar noticias y traducciones en tiempo real añade una considerable complejidad técnica al proyecto.
- **Complejidad del proceso lógico interno de la aplicación:** La lógica interna es compleja debido al procesamiento y traducción de datos de múltiples fuentes, así como la interacción con diversas APIs y con la base de datos.

Para determinar los factores de ajuste de valor (FAV) de manera precisa, se utiliza la siguiente fórmula:

$$FA = 0,65 + (0,01 \times \sum FAV)$$

donde $\sum FAV$ representa la suma de los valores de influencia de los atributos, calculada en la tabla anterior. Teniendo en cuenta los valores obtenidos nos quedaría que:

$$FA = 0,65 + (0,01 \times 42) = 1,07$$

Este factor de ajuste nos permite refinar la estimación inicial, teniendo en cuenta los diversos factores de complejidad del sistema.

Cálculo de los puntos de función ajustados (PFA)

Para calcular los puntos de función ajustados, se utiliza la fórmula siguiente:

$$PFA = PFNA \times FA$$

Donde PFNA representa los puntos de función no ajustados y FA es el factor de ajuste. Considerando los resultados obtenidos:

$$PFA = 158 \times 1,07 = 169 \text{ (aproximado)}$$

Cálculo del tiempo en días de esfuerzo

Para estimar la duración del proyecto, se puede asignar un valor de esfuerzo inicial basado en puntos de función. Una aproximación basándonos en valores promedio de la industria informática, se recomienda utilizar la siguiente referencia: 1 mes de esfuerzo (aproximadamente 21 días laborales) equivale a 13 Puntos de Función.

Dado que este proyecto tiene un total de 169 Puntos de Función, podemos calcular la duración estimada del proyecto utilizando esta referencia. Dividiendo el total de Puntos de Función entre la cantidad de Puntos de Función que se pueden completar en un mes, obtenemos:

$$\text{Duración estimada} = 169 / 13 = 13 \text{ meses}$$

Por lo tanto, se estima que el proyecto requerirá aproximadamente 13 meses de esfuerzo para completarse.

A continuación, se detallará el presupuesto económico del proyecto, teniendo en cuenta tanto los recursos técnicos como los recursos humanos necesarios para su desarrollo basándome en la estimación obtenida a través del método Albrecht. Esta doble aproximación nos proporcionará una visión completa y detallada del coste total del proyecto, asegurando que todos los aspectos relevantes sean considerados.

Recursos técnicos

Para la realización del proyecto **OVERNEWS** se han utilizado programas de uso gratuito, y la licencia de Office es la que nos proporciona la UVA, por lo que no ha sido necesario realizar ninguna inversión en software. En cuanto al hardware, el único dispositivo necesario ha sido un ordenador junto con sus periféricos (teclado, ratón, etc.).

El ordenador y sus complementos tuvieron un coste de 1.300€, con una vida útil estimada de entre 3-5 años. Es por esto por lo que tomaremos que la vida útil es de 4 años. Durante los 13 meses de desarrollo del proyecto, el ordenador se utilizó intensivamente, por lo tanto, se le sumará un porcentaje extra de gasto. A pesar de esto, se ha considerado que aproximadamente el 33,95% (29,2% + 3,75%) de la vida útil del dispositivo se ha destinado a este proyecto, debido a las altas demandas técnicas del mismo. Por lo tanto, la inversión en hardware para el desarrollo del proyecto se estima en 442,00 €.

Además del coste del hardware y software, es importante considerar los gastos operativos, como la luz y el wifi, que son esenciales para el desarrollo del proyecto. El gasto mensual medio de luz es de 55,83€ y el de internet es de 41,99€. Para los 13 meses de desarrollo, esto supone un coste adicional de 725,79 € en luz y 545,87€ en internet, sumando un total de 586,92€.

Recurso técnico	Coste
Hardware	442,00€
Software	0,00€
Luz	725,79 €
Wifi	545,87€
Total	1.713,66€

Tabla 15. Presupuesto Recursos Técnicos (Método Albrecht)

El gasto total de todos los recursos técnicos sería de 1.713,66€.

Recursos humanos

En este apartado se realiza una estimación detallada del coste del proyecto, teniendo en cuenta el número total de horas dedicadas y el salario medio de un desarrollador en España. El salario medio de un desarrollador se sitúa en 2.500€ mensuales.

El proyecto, con los datos obtenidos usando el método Albrecht, requiere de una dedicación total de 13 meses de trabajo. Al multiplicar el número de meses necesarios por el salario mensual de un desarrollador, se obtiene un coste total de 32.500€.

Estimación del Presupuesto Total

Este apartado proporciona un resumen del presupuesto del proyecto, utilizando el Método de Albrecht para estimar los esfuerzos necesarios. A partir de los puntos de función calculados con este método, se ha determinado un tiempo de desarrollo estimado de 13 meses. Con esta información, se ha calculado el presupuesto necesario para los recursos humanos y técnicos, alcanzando un total de 34.213,66€. A continuación, se detalla el resumen del presupuesto para este proyecto.

Recurso	Presupuesto Estimado
Recursos técnicos	1.713,66€
Recursos humanos	32.500,00€
Total	34.213,66€

Tabla 16. Estimación del Presupuesto Total (Método Albrecht)

2.2.3.2 Estimación de Costes Basada en la Planificación Inicial

En este apartado se llevará a cabo la estimación de costes, teniendo en cuenta los datos obtenidos en la planificación temporal inicial (sección 2.2.1) del proyecto OVERNEWS. En esta sección se estimó que el tiempo requerido para desarrollar este proyecto era de 3 meses. Este plazo asegura una ejecución eficiente del proyecto.

Recursos técnicos

Como ya he comentado al obtener los recursos técnicos al estimar a través del método Albrecht, no ha sido necesario realizar ninguna inversión en software. En cuanto al hardware, se requiere de los mismos dispositivos que mencioné anteriormente, que suponía un coste de 1300€.

En este caso, al analizar los dispositivos requeridos, hay que tener en cuenta que se utilizaría durante 3 meses. Además, el ordenador se utilizó intensivamente. Es por esto, que se ha considerado que aproximadamente el 10% (6,25% + 3,75%) de la vida útil del dispositivo se ha destinado a este proyecto, debido a las altas demandas técnicas del mismo. Por lo tanto, la inversión en hardware para el desarrollo del proyecto se estima en 130,00€.

Para calcular los gastos operativos, como la luz y el wifi, tendré en cuenta que el gasto mensual medio de luz es de 55,83€ y el de internet es de 41,99€. Para los 3 meses de desarrollo, esto supone un coste adicional de 167,49€ en luz y 125,97€ en internet, sumando un total de 586,92€.

Recurso técnico	Coste
Hardware	130,00€
Software	0,00€
Luz	167,49€
Wifi	125,97€
Total	423,46€

Tabla 17. Presupuesto Recursos Técnicos (Planificación Inicial)

Por lo tanto, el gasto que se tendría que hacer para los recursos técnicos sería de 423,46€

Recursos humanos

En esta sección se realiza una estimación detallada del coste del proyecto, teniendo en cuenta el número total de horas dedicadas según los datos obtenidos en la planificación inicial. Como ya comenté anteriormente el salario medio de un desarrollador se sitúa en 2.500€ mensuales, basado en una jornada laboral estándar de 8 horas diarias.

El proyecto requiere de una dedicación total de 381 horas de trabajo. Al considerar una jornada completa de 8 horas, esto equivale a un total de 48 jornadas de trabajo. Teniendo en cuenta que un mes laboral típico comprende alrededor de 21 jornadas laborales, el tiempo necesario para completar el proyecto se estima en aproximadamente 2,3 meses.

Al multiplicar el número de meses necesarios por el salario mensual de un desarrollador, se obtiene un coste total de la parte de recursos humanos, siendo de 5.750€.

Estimación del Presupuesto Total

Este apartado proporciona un resumen del presupuesto del proyecto basado en la planificación inicial. Según los datos obtenidos, se ha estimado que el proyecto tendrá una duración de 3 meses. Con esta planificación, se ha calculado el presupuesto necesario para los recursos humanos y técnicos, resultando en un total estimado de 6.173,46€. A continuación, se presenta el resumen detallado del presupuesto para este proyecto.

Recurso	Presupuesto Estimado
Recursos técnicos	423,46€
Recursos humanos	5.750,00€
Total	6.173,46€

Tabla 18. Estimación del Presupuesto Total (Planificación Inicial)

2.2.3.3 Comparación de las estimaciones

En este apartado se realizará una comparación entre las estimaciones presupuestarias del proyecto OVERNEWS obtenidas mediante el método de Albrecht y las basadas en los valores obtenidos durante la planificación inicial. Esta comparación permitirá evaluar la precisión y la eficacia de ambos métodos de estimación, proporcionando una visión más completa del esfuerzo y los recursos financieros necesarios para el desarrollo del proyecto.

Estimación	Presupuesto
Método Albrecht	34.213,66€
Basada en la Planificación Inicial	6.173,46€

Tabla 19. Comparativa entre estimaciones

La estimación del método de Albrecht es considerablemente mayor (34.213,66€) que la basada en la planificación inicial (6.173,46€), sugiriendo una anticipación de mayor esfuerzo y recursos.

El método de Albrecht, basado en puntos de función, proporciona una estimación más detallada y conservadora, reflejando una visión más realista del esfuerzo requerido. En contraste, la planificación inicial parece subestimar los recursos necesarios, presentando una visión más optimista. Utilizar ambos enfoques puede ofrecer una visión más completa y equilibrada del presupuesto necesario para el desarrollo del proyecto.

2.2.4 Seguimiento Real

Como he mencionado en la sección anterior (Sección 2.2.3) durante la planificación inicial del proyecto OVERNEWS, se estimó que la duración de este sería de entre dos y tres meses, con la intención de finalizar a finales de enero o principios de febrero. Sin embargo, la realidad del desarrollo presentó algunos desafíos inesperados. La complejidad del proyecto fue mayor de lo anticipado, lo que resultó en demoras mientras resolvía problemas técnicos y de implementación. Además, a mediados de enero, comencé a trabajar a tiempo completo, lo que redujo significativamente el tiempo disponible para dedicar al TFG.

Estas circunstancias hicieron que el proyecto se extendiera hasta junio, duplicando prácticamente el tiempo inicialmente previsto. A pesar de estos contratiempos, el avance constante y el esfuerzo dedicado fuera del horario laboral permitieron completar el proyecto con éxito. La prolongación refleja también la capacidad de adaptarse a nuevas circunstancias y mantener el foco en los objetivos del proyecto.

2.2.4.1 Coste real del proyecto

El incremento en el tiempo de desarrollo del proyecto ha resultado en un aumento significativo del coste total respecto a lo inicialmente presupuestado. La extensión del plazo de tres meses a siete meses ha elevado de forma considerable el coste los recursos técnicos y humanos necesarios. A continuación, se analizará más detenidamente este incremento.

Recursos técnicos

En la planificación inicial, se estimó un uso intensivo del hardware durante tres meses, calculando que aproximadamente el 10% de la vida útil del equipo se utilizaría para el

proyecto, con un coste de 130€. Con la extensión del proyecto a siete meses, este uso se ha incrementado proporcionalmente. Además, los gastos operativos como luz y wifi también han aumentado.

Teniendo en cuenta los valores usados en la planificación inicial, se obtienen el coste real de los recursos técnicos:

- **Hardware:**
 - Uso inicial (3 meses): 130€
 - Uso ajustado (7 meses): $130€ * (7/3) = 303,33€$
- **Gastos operativos (Luz y Wifi):**
 - Luz (55,83€/mes * 7 meses): 390,81€
 - Wifi (41,99€/mes * 7 meses): 293,93€
 - Total de los gastos operativos: $390,81€ + 293,93€ = 684,74€$

Recurso Técnico	Coste Ajustado
Hardware	303,33€
Software	0,00€
Luz	390,81€
Wifi	293,93€
Total	988,07€

Tabla 20. Presupuesto ajustado (Planificación Final)

Recursos Humanos

Inicialmente, se estimó un coste de 5.750€ basado en 381 horas de trabajo distribuidas en 2,3 meses. Dado que el proyecto se extendió a siete meses, hay que tener en cuenta que, el sobrecoste que esto supone. Durante estos últimos 4 meses, debido a que no he podido dedicarle todo el tiempo posible, se pasaría de tener un contrato de jornada completa a jornada parcial. Por lo tanto, en estos cuatro meses, al haberle empleado una tercera parte del tiempo requerido, equivaldría a 1,33 meses con jornada completa ($4 \times 0,33=1,33$).

Por lo tanto, suponiendo que un programador en el sector informático cobra 2500€, el coste ajustado sería de 10.825€

Coste real

El presupuesto total se ajusta en función de los incrementos en recursos técnicos y humanos, reflejando la extensión del proyecto.

Recurso	Coste Real
Recursos técnicos	988,07€
Recursos humanos	10.825€
Total	11.813,07€

Tabla 21. Coste Real

2.2.4.2 Comparación con la Planificación Inicial

En este apartado se realiza una comparación detallada entre la planificación inicial y la ejecución real del proyecto OVERNEWS. Esta comparación revela varias conclusiones importantes en términos de costes.

Balance económico	Total
Presupuesto Estimado (Inicial)	6.173,46€
Presupuesto Ajustado (Final)	11.813,07€
Diferencia	5.639,61€

Tabla 22. Comparativa con la Planificación Inicial (Planificación Final)

El presupuesto inicial estimado era de 6.173,46€, mientras que el coste final ajustado alcanzó los 11.813,07€ representando un incremento de 5.639,61€. Esta diferencia significativa subraya la importancia de llevar a cabo una planificación flexible y la capacidad de adaptarse a cambios y desafíos imprevistos durante el desarrollo de un proyecto.

Capítulo 3

Análisis

En este capítulo se realiza un análisis exhaustivo de los requisitos y casos de uso del proyecto OVERNEWS, proporcionando una base sólida para su diseño y desarrollo. Los puntos que tratar son: Requisitos de Usuario (Sección 3.1), Casos de Uso (Sección 3.2), Requisitos Funcionales (Sección 3.3) y Requisitos No Funcionales (Sección 3.4).

3.1 Requisitos de Usuario

En esta primera sección del Análisis se definirán los requisitos de usuario de la aplicación. Los requisitos de usuario se definen como afirmaciones sobre los objetivos de los usuarios o las tareas de negocio que estos deben desempeñar utilizando el nuevo el sistema.[\[13\]](#)

A continuación, se presenta la tabla de requisitos de usuario para OVERNEWS.

ID	Descripción
RU-01	El usuario debe poder registrarse en la aplicación.
RU-02	El usuario debe poder iniciar sesión en la aplicación una vez registrado.
RU-03	El usuario registrado debe poder recuperar su contraseña en caso de olvido.
RU-04	El usuario registrado debe poder cerrar sesión en la aplicación.
RU-05	El usuario no registrado debe de poder leer las noticias en castellano.
RU-06	El usuario registrado debe poder ver y leer noticias en su idioma preferido.
RU-07	El usuario registrado debe poder buscar noticias por categoría o por fuente.
RU-08	El usuario registrado debe poder marcar noticias como favoritas.
RU-09	El usuario registrado debe poder ver una lista de sus noticias favoritas.
RU-10	El usuario registrado debe poder comentar en las noticias.
RU-11	El usuario registrado debe poder editar y eliminar sus propios comentarios, si así lo desea.
RU-12	El usuario debe poder ver los comentarios de otros usuarios en las noticias.
RU-13	El usuario registrado debe poder ver los comentarios que ha realizado.
RU-14	El usuario registrado debe poder gestionar su perfil, incluyendo la selección de su idioma preferido para las noticias.
RU-15	El usuario registrado debe poder eliminar su cuenta, si así lo desea.
RU-16	El usuario registrado debe poder sugerir nuevas fuentes al administrador.

RU-17	El usuario registrado debe poder contactar con el soporte a través de email.
RU-18	El usuario administrador debe poder eliminar noticias.
RU-19	El usuario administrador debe poder eliminar usuarios.
RU-20	El usuario administrador debe poder eliminar sugerencias de nuevas fuentes.
RU-21	El usuario administrador debe poder aceptar las sugerencias de los usuarios
RU-22	El usuario administrador debe poder eliminar comentarios.
RU-23	El usuario administrador debe poder ver analíticas sobre las noticias.

Tabla 23. Requisitos de Usuario

3.2 Casos de Uso

En esta sección se definirán los diferentes casos de uso que se desarrollarán en la aplicación OVERNEWS. Los casos de uso son descripciones de las interacciones entre un usuario (actor) y el sistema, destinadas a alcanzar un objetivo específico. Los casos de uso comprenden cómo los usuarios interactuarán con el sistema y qué funcionalidades se deben implementar para satisfacer sus necesidades.

Para obtener una información completa de los casos de uso, se desarrollarán los siguientes puntos: Especificación de Actores (Sección 3.2.1), Diagramas de Casos de Uso (Sección 3.2.2) y Especificación de Casos de Uso (Sección 3.2.3).

3.2.1 Especificación de los Actores

En esta sección se proporcionará información detallada sobre los actores involucrados en los modelos de procesos a implementar en la aplicación OVERNEWS. Los actores pueden ser organizaciones, roles o responsabilidades que interactúan con el sistema. La especificación se llevará a cabo utilizando plantillas que describen las características y funciones de cada actor, asegurando una comprensión clara de sus interacciones y contribuciones al sistema.

ACT-01	Usuario
Versión	V1.0 (20/06/2024)
Dependencias	
Descripción	Este actor representa al usuario registrado en OVERNEWS. Los usuarios tienen acceso completo a las funcionalidades de la aplicación, incluyendo la capacidad de ver y leer noticias en su idioma preferido, marcar noticias como favoritas, comentar en noticias, ver comentarios de otros usuarios, gestionar su perfil y buscar noticias por categoría o fuente. También pueden sugerir nuevas fuentes, eliminar su cuenta y contactar con soporte a través de email.
Comentarios	Se trata de un usuario que ha completado el proceso de registro en la aplicación y ha iniciado sesión.

Tabla 24. Especificación Actor Usuario

ACT-02	Visitante
Versión	V1.0 (20/06/2024)
Dependencias	
Descripción	Este actor representa a un usuario que no se ha registrado en OVERNEWS. Aunque no requieren haberse dado de alta, sus funcionalidades están limitadas. Los visitantes pueden consultar y leer noticias en español, pero no pueden acceder a funciones avanzadas como comentar, marcar noticias como favoritas, gestionar un perfil personal.
Comentarios	El visitante tiene acceso a la visualización de noticias y a la información general de la aplicación, pero no puede interactuar plenamente con el sistema sin registrarse.

Tabla 25. Especificación Actor Visitante

ACT-03	Administrador
Versión	V1.0 (20/06/2024)
Dependencias	
Descripción	Este actor representa al administrador del sistema OVERNEWS. Los administradores tienen privilegios especiales que les permiten gestionar la aplicación, incluyendo la capacidad de eliminar noticias, eliminar usuarios, eliminar sugerencias de nuevas fuentes, eliminar comentarios y ver analíticas sobre las noticias.
Comentarios	El administrador es responsable de mantener el orden y la calidad del contenido en la plataforma, asegurando que las noticias y los usuarios cumplan con las políticas establecidas.

Tabla 26. Especificación Actor Administrador

3.2.2 Listado de Casos de Uso

Antes de visualizar los diagramas, es fundamental comprender los diferentes casos de uso definidos para la aplicación.

ID	Caso de Uso	Descripción	Actores
CU-01	Registrarse en la aplicación	Permite al visitante registrarse para obtener una cuenta de usuario.	Visitante
CU-02	Iniciar sesión	Permite al usuario o visitante iniciar sesión en la aplicación con sus credenciales.	Usuario, Visitante
CU-03	Recuperar contraseña	Permite al usuario recuperar su contraseña en caso de olvido.	Usuario
CU-04	Cerrar sesión	Permite al usuario cerrar sesión en la aplicación.	Usuario

CU-05	Ver y leer en noticias español	Permite al visitante ver y leer noticias solo en español.	Visitante
CU-06	Ver y leer en noticias en varios idiomas	Permite al usuario ver y leer noticias en su idioma preferido.	Usuario
CU-07	Buscar noticias	Permite al usuario buscar noticias por categoría o por fuente.	Usuario
CU-08	Marcar noticias como favoritas	Permite al usuario marcar noticias como favoritas.	Usuario
CU-09	Ver noticias favoritas	Permite al usuario ver una lista de sus noticias favoritas.	Usuario
CU-10	Comentar en noticias	Permite al usuario comentar en las noticias.	Usuario
CU-11	Editar y eliminar comentarios	Permite al usuario editar y eliminar sus propios comentarios.	Usuario
CU-12	Ver comentarios de otros usuarios	Permite al usuario ver los comentarios de otros usuarios en las noticias.	Usuario, Visitante
CU-13	Ver propios comentarios	Permite al usuario ver los comentarios que ha realizado.	Usuario
CU-14	Gestionar perfil	Permite al usuario gestionar su perfil, incluyendo la selección de su idioma preferido y editar o eliminar su cuenta.	Usuario
CU-15	Sugerir nuevas fuentes	Permite al usuario sugerir nuevas fuentes de noticias al administrador.	Usuario
CU-16	Contactar con soporte	Permite al usuario contactar con el soporte a través de email.	Usuario
CU-17	Gestionar contenidos y usuarios	Permite al administrador eliminar noticias, usuarios, comentarios y sugerencias, aceptar sugerencias y ver analíticas sobre las noticias.	Administrador

Tabla 27. Casos de Uso

A continuación, se muestra la tabla en la que se asocia los requisitos de usuario con los casos de uso.

ID	Requisito de Usuario	Caso de Uso	ID de CU
RU-01	El usuario debe poder registrarse en la aplicación.	Registrarse en la aplicación	CU-01
RU-02	El usuario debe poder iniciar sesión en la aplicación una vez registrado.	Iniciar sesión	CU-02
RU-03	El usuario registrado debe poder recuperar su	Recuperar	CU-03

	contraseña en caso de olvido.	contraseña	
RU-04	El usuario registrado debe poder cerrar sesión en la aplicación.	Cerrar sesión	CU-04
RU-05	El usuario no registrado debe de poder leer las noticias en castellano.	Ver y leer noticias en español	CU-05
RU-05	El usuario registrado debe poder ver y leer noticias en su idioma preferido.	Ver y leer noticias en varios idiomas	CU-05
RU-06	El usuario registrado debe poder buscar noticias por categoría o por fuente.	Buscar noticias	CU-07
RU-07	El usuario registrado debe poder marcar noticias como favoritas.	Marcar noticias como favoritas	CU-08
RU-08	El usuario registrado debe poder ver una lista de sus noticias favoritas.	Ver noticias favoritas	CU-09
RU-09	El usuario registrado debe poder comentar en las noticias.	Comentar en noticias	CU-10
RU-10	El usuario registrado debe poder editar y eliminar sus propios comentarios si así lo desea.	Editar y eliminar comentarios	CU-11
RU-11	El usuario debe poder ver los comentarios de otros usuarios en las noticias.	Ver comentarios de otros usuarios	CU-12
RU-12	El usuario registrado debe poder ver los comentarios que ha realizado.	Ver propios comentarios	CU-13
RU-13	El usuario registrado debe poder gestionar su perfil, incluyendo la selección de su idioma preferido para las noticias.	Gestionar perfil	CU-14
RU-14	El usuario registrado debe poder eliminar su cuenta si así lo desea.		
RU-15	El usuario registrado debe poder sugerir nuevas fuentes al administrador.	Sugerir nuevas fuentes	CU-15
RU-16	El usuario registrado debe poder contactar con el soporte a través de email.	Contactar con soporte	CU-16
RU-17	El usuario administrador debe poder eliminar noticias.	Gestionar contenidos y usuarios	CU-17
RU-18	El usuario administrador debe poder eliminar usuarios.		
RU-19	El usuario administrador debe poder eliminar sugerencias de nuevas fuentes.		
RU-20	El usuario administrador debe poder aceptar las sugerencias de los usuarios.		
RU-21	El usuario administrador debe poder eliminar comentarios.		
RU-22	El usuario administrador debe poder ver analíticas sobre las noticias.		

Tabla 28. Asociación de Casos de Uso con Requisitos de Usuario

3.2.3 Diagrama de Casos de Uso

En esta sección, se presenta el diagrama de casos de uso.

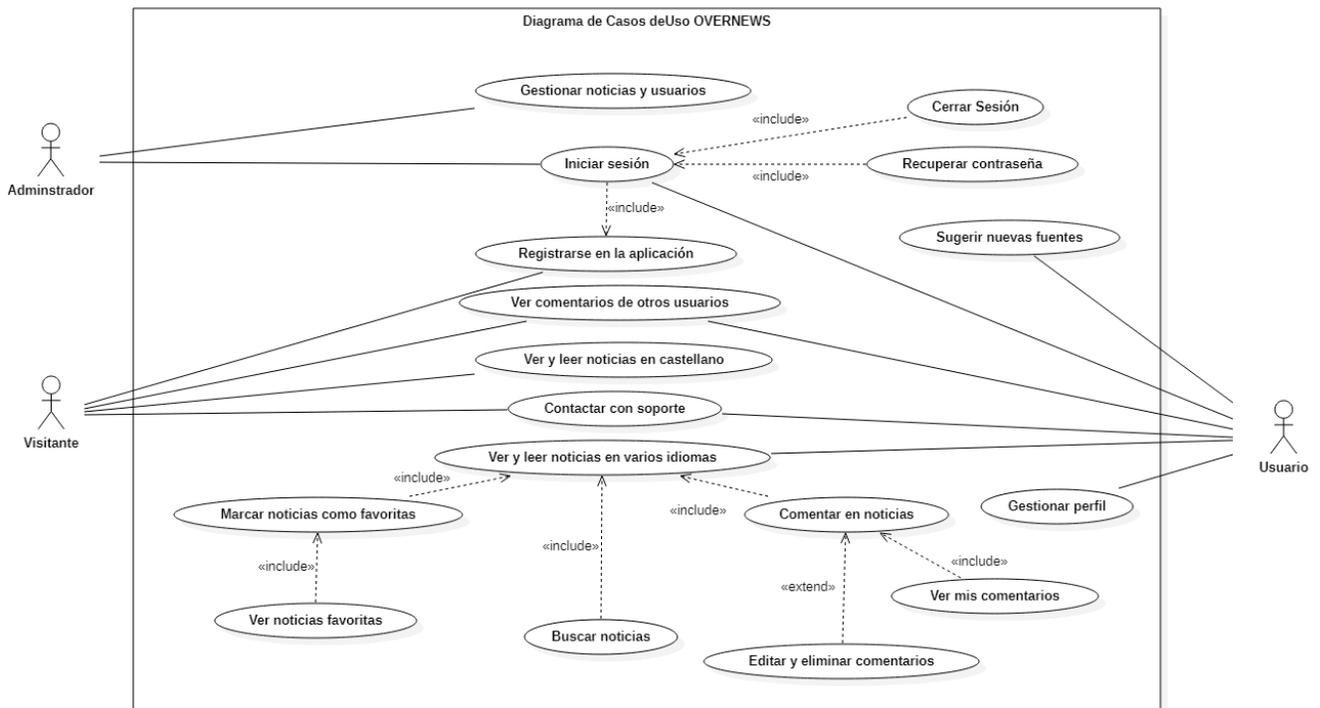


Figura 16. Diagrama de Casos de Uso

3.2.4 Especificación de Casos de Uso

En esta sección se presentan las especificaciones detalladas de los casos de uso identificados para la aplicación OVERNEWS.

CU-01	Registrarse en la aplicación
Versión	1.0 (20/06/2024)
Actor Principal	Visitante
Actor Secundario	-
Dependencias	• RU-01 El usuario debe poder registrarse en la aplicación.
Precondición	El visitante no debe tener una cuenta existente en la aplicación.
Descripción	Permite al visitante registrarse para obtener una cuenta de usuario.
Secuencia Normal	<ol style="list-style-type: none"> 1. El visitante selecciona la opción de registrarse en la aplicación. 2. El sistema le redirige a la página de Registro. 3. El visitante completa el formulario con sus datos y lo envía. 4. El sistema valida los datos y crea una nueva cuenta de usuario. 5. Si la validación es exitosa, el sistema obtiene el token (JWT) y envía un correo de validación.

	<ol style="list-style-type: none"> 6. El visitante recibe el correo y confirma su registro. 7. El sistema redirige al usuario a la página principal.
Postcondición	El visitante tendrá una cuenta creada y podrá iniciar sesión en la aplicación.
Excepciones	Si los datos no son válidos, se muestra un mensaje de error y se solicita corregir los datos.
Importancia	Alta

Tabla 29. Especificación del CU-01

CU-02	Iniciar sesión
Versión	1.0 (20/06/2024)
Actor Principal	Usuario, Visitante
Actor Secundario	-
Dependencias	<ul style="list-style-type: none"> • RU-02 El usuario debe poder iniciar sesión en la aplicación una vez registrado.
Precondición	El usuario o visitante debe tener una cuenta registrada en la aplicación.
Descripción	Permite al usuario o visitante iniciar sesión en la aplicación con sus credenciales.
Secuencia Normal	<ol style="list-style-type: none"> 1. El usuario o visitante selecciona la opción de iniciar sesión. 2. El sistema le redirige a la página de inicio de sesión. 3. El usuario o visitante ingresa sus credenciales y las envía. 4. El sistema valida las credenciales. 5. Si las credenciales son válidas, el sistema permite el acceso a la cuenta.
Postcondición	El usuario o visitante tendrá acceso a las funcionalidades de la aplicación según su rol.
Excepciones	Si las credenciales no son válidas, se muestra un mensaje de error y se solicita reintentar.
Importancia	Alta

Tabla 30. Especificación del CU-02

CU-03	Recuperar contraseña
Versión	1.0 (20/06/2024)
Actor Principal	Usuario
Actor Secundario	-
Dependencias	<ul style="list-style-type: none"> • RU-03 El usuario registrado debe poder recuperar su contraseña en caso de olvido.
Precondición	El usuario ha olvidado su contraseña.

Descripción	Permite al usuario recuperar su contraseña en caso de olvido.
Secuencia Normal	<ol style="list-style-type: none"> 1. El usuario selecciona la opción de recuperar contraseña. 2. El sistema muestra un formulario para ingresar el correo electrónico. 3. El usuario ingresa su correo electrónico y lo envía. 4. El sistema valida el correo y envía un enlace de recuperación. 5. El usuario recibe el enlace y lo utiliza para restablecer su contraseña. 6. El usuario rellena el formulario dónde confirma su nueva contraseña. 7. Si las contraseñas coinciden, se cambia la contraseña.
Postcondición	El usuario recibirá un enlace para restablecer su contraseña y podrá crear una nueva.
Excepciones	<p>Si el correo no es válido, se muestra un mensaje de error y se solicita reintentar.</p> <p>Si las contraseñas no coinciden, se le muestra un mensaje de error.</p>
Importancia	Media

Tabla 31. Especificación del CU-03

CU-04	Cerrar sesión
Versión	1.0 (20/06/2024)
Actor Principal	Usuario
Actor Secundario	-
Dependencias	<ul style="list-style-type: none"> • RU-04 El usuario registrado debe poder cerrar sesión en la aplicación.
Precondición	El usuario debe estar autenticado en la aplicación.
Descripción	Permite al usuario cerrar sesión en la aplicación.
Secuencia Normal	<ol style="list-style-type: none"> 1. El usuario selecciona la opción de cerrar sesión. 2. El sistema cierra la sesión del usuario y lo redirige a la página de inicio.
Postcondición	El usuario será desconectado de la aplicación y se finalizará su sesión activa.
Excepciones	-
Importancia	Alta

Tabla 32. Especificación del CU-04

CU-05	Ver y leer noticias en español
Versión	1.0 (20/06/2024)
Actor Principal	Visitante
Actor Secundario	-
Dependencias	<ul style="list-style-type: none"> • RU-05 El visitante debe poder ver y leer noticias en español.
Precondición	El visitante debe entrar en la aplicación.

Descripción	El sistema deberá comportarse como se describe en el siguiente caso de uso cuando un visitante desee ver y leer noticias en español.
Secuencia Normal	<ol style="list-style-type: none"> 1. El visitante selecciona la opción de ver noticias. 2. El sistema muestra una lista de noticias en español. 3. El visitante selecciona una noticia para leer. 4. El sistema muestra el contenido completo de la noticia en español.
Postcondición	El visitante podrá ver y leer noticias en español.
Excepciones	-
Importancia	Alta

Tabla 33. Especificación del CU-05

CU-06	Ver y leer noticias en varios idiomas
Versión	1.0 (20/06/2024)
Actor Principal	Usuario
Actor Secundario	-
Dependencias	• RU-05 El usuario registrado debe poder ver y leer noticias en su idioma preferido.
Precondición	El usuario debe estar autenticado en la aplicación.
Descripción	El sistema deberá comportarse como se describe en el siguiente caso de uso cuando un usuario desee ver y leer noticias en su idioma preferido.
Secuencia Normal	<ol style="list-style-type: none"> 1. El usuario selecciona la opción de ver noticias. 2. El sistema muestra una lista de noticias en el idioma preferido del usuario. 3. El usuario selecciona una noticia para leer. 4. El sistema muestra el contenido completo de la noticia en el idioma seleccionado.
Postcondición	El usuario podrá ver y leer noticias en el idioma seleccionado.
Excepciones	-
Importancia	Alta

Tabla 34. Especificación del CU-06

CU-07	Buscar noticias
Versión	1.0 (20/06/2024)
Actor Principal	Usuario
Actor Secundario	-
Dependencias	• RU-06 El usuario registrado debe poder buscar noticias por categoría o por fuente.
Precondición	El usuario debe estar autenticado en la aplicación.
Descripción	Permite al usuario buscar noticias por categoría o por fuente.

Secuencia Normal	<ol style="list-style-type: none"> 1. El usuario selecciona la opción de buscar noticias. 2. El sistema muestra un formulario de búsqueda con opciones de categoría y fuente. 3. El usuario ingresa los criterios de búsqueda y envía el formulario. 4. El sistema procesa la búsqueda y muestra los resultados. 5. El usuario selecciona una noticia de los resultados para leer. 6. El sistema muestra el contenido de la noticia seleccionada.
Postcondición	El usuario podrá ver los resultados de la búsqueda de noticias según los criterios seleccionados.
Excepciones	-
Importancia	Alta

Tabla 35. Especificación del CU-07

CU-08	Marcar noticias como favoritas
Versión	1.0 (20/06/2024)
Actor Principal	Usuario
Actor Secundario	-
Dependencias	<ul style="list-style-type: none"> • RU-07 El usuario registrado debe poder marcar noticias como favoritas.
Precondición	El usuario debe estar autenticado en la aplicación.
Descripción	Permite al usuario marcar noticias como favoritas.
Secuencia Normal	<ol style="list-style-type: none"> 1. El usuario selecciona una noticia para ver. 2. El sistema muestra el contenido de la noticia. 3. El usuario selecciona la opción de marcar como favorita. 4. El sistema marca la noticia como favorita y la guarda en la tabla de noticias favoritas.
Postcondición	La noticia será marcada como favorita y almacenada en la lista de favoritas del usuario.
Excepciones	-
Importancia	Media

Tabla 36. Especificación del CU-08

CU-09	Ver noticias favoritas
Versión	1.0 (20/06/2024)
Actor Principal	Usuario
Actor Secundario	-
Dependencias	<ul style="list-style-type: none"> • RU-08 El usuario registrado debe poder ver una lista de sus noticias favoritas.
Precondición	El usuario debe estar autenticado en la aplicación y tener noticias marcadas como favoritas.
Descripción	Permite al usuario ver una lista de sus noticias favoritas.

Secuencia Normal	<ol style="list-style-type: none"> 1. El usuario selecciona la opción de ver noticias favoritas. 2. El sistema muestra una lista de noticias marcadas como favoritas por el usuario. 3. El usuario selecciona una noticia de la lista para leer. 4. El sistema muestra el contenido de la noticia seleccionada.
Postcondición	El usuario podrá ver la lista de sus noticias favoritas.
Excepciones	-
Importancia	Media

Tabla 37. Especificación del CU-09

CU-10	Comentar en noticias
Versión	1.0 (20/06/2024)
Actor Principal	Usuario
Actor Secundario	-
Dependencias	• RU-09 El usuario registrado debe poder comentar en las noticias.
Precondición	El usuario debe estar autenticado en la aplicación.
Descripción	Permite al usuario comentar en las noticias.
Secuencia Normal	<ol style="list-style-type: none"> 1. El usuario selecciona una noticia para ver. 2. El sistema muestra el contenido de la noticia. 3. El usuario selecciona la opción de comentar. 4. El sistema muestra un formulario para ingresar el comentario. 5. El usuario ingresa su comentario y lo envía. 6. El sistema guarda el comentario y lo muestra junto con la noticia.
Postcondición	El comentario será almacenado y visible para otros usuarios.
Excepciones	-
Importancia	Alta

Tabla 38. Especificación del CU-10

CU-11	Editar y eliminar comentarios
Versión	1.0 (20/06/2024)
Actor Principal	Usuario
Actor Secundario	-
Dependencias	• RU-10 El usuario registrado debe poder editar y eliminar sus propios comentarios si así lo desea.
Precondición	El usuario debe estar autenticado en la aplicación y tener comentarios existentes.
Descripción	Permite al usuario editar y eliminar sus propios comentarios.

Secuencia Normal	<ol style="list-style-type: none"> 1. El usuario selecciona una noticia con su comentario. 2. El sistema muestra el contenido de la noticia y el comentario del usuario. 3. El usuario selecciona la opción de editar o eliminar el comentario. 4. El sistema permite la edición o confirma la eliminación del comentario.
Postcondición	Los cambios en los comentarios serán guardados o eliminados según la acción realizada.
Excepciones	-
Importancia	Media

Tabla 39. Especificación del CU-11

CU-12	Ver comentarios de otros usuarios
Versión	1.0 (20/06/2024)
Actor Principal	Usuario, Visitante
Actor Secundario	-
Dependencias	<ul style="list-style-type: none"> • RU-11 El usuario debe poder ver los comentarios de otros usuarios en las noticias.
Precondición	La noticia debe tener comentarios de otros usuarios.
Descripción	Permite al usuario ver los comentarios de otros usuarios en las noticias.
Secuencia Normal	<ol style="list-style-type: none"> 1. El usuario selecciona una noticia para ver. 2. El sistema muestra el contenido de la noticia y los comentarios de otros usuarios.
Postcondición	Los comentarios de otros usuarios serán visibles para el usuario.
Excepciones	-
Importancia	Alta

Tabla 40. Especificación del CU-12

CU-13	Ver propios comentarios
Versión	1.0 (20/06/2024)
Actor Principal	Usuario
Actor Secundario	-
Dependencias	<ul style="list-style-type: none"> • RU-12 El usuario registrado debe poder ver los comentarios que ha realizado.
Precondición	El usuario debe estar autenticado en la aplicación y tener comentarios existentes.
Descripción	Permite al usuario ver los comentarios que ha realizado.

Secuencia Normal	<ol style="list-style-type: none"> 1. El usuario selecciona la opción de ver sus comentarios. 2. El sistema muestra una lista de comentarios realizados por el usuario. 3. El usuario selecciona un comentario para ver más detalles. 4. El sistema muestra el contenido del comentario y la noticia asociada.
Postcondición	Los comentarios realizados por el usuario serán visibles para él.
Excepciones	-
Importancia	Media

Tabla 41. Especificación del CU-13

CU-14	Gestionar perfil
Versión	1.0 (20/06/2024)
Actor Principal	Usuario
Actor Secundario	-
Dependencias	• RU-13 El usuario registrado debe poder gestionar su perfil, incluyendo la selección de su idioma preferido y eliminar su cuenta.
Precondición	El usuario debe estar autenticado en la aplicación.
Descripción	Permite al usuario gestionar su perfil, incluyendo la selección de su idioma preferido y editar o eliminar su cuenta.
Secuencia Normal	<ol style="list-style-type: none"> 1. El usuario selecciona la opción de gestionar su perfil. 2. El sistema muestra el formulario de perfil del usuario. 3. El usuario edita su información de perfil, selecciona su idioma preferido o elimina su cuenta. 4. El sistema guarda los cambios realizados por el usuario.
Postcondición	Los cambios en el perfil del usuario serán guardados y actualizados.
Excepciones	-
Importancia	Alta

Tabla 42. Especificación del CU-14

CU-15	Sugerir nuevas fuentes
Versión	1.0 (20/06/2024)
Actor Principal	Usuario
Actor Secundario	Administrador
Dependencias	• RU-15 El usuario registrado debe poder sugerir nuevas fuentes al administrador.
Precondición	El usuario debe estar autenticado en la aplicación.
Descripción	Permite al usuario sugerir nuevas fuentes de noticias al administrador.
Secuencia Normal	<ol style="list-style-type: none"> 1. El usuario selecciona la opción de sugerir nuevas fuentes. 2. El sistema muestra un formulario para ingresar la sugerencia de nuevas fuentes. 3. El usuario completa el formulario y lo envía.

	4. El sistema envía la sugerencia al administrador para su revisión.
Postcondición	La sugerencia será enviada al administrador para su revisión y posible inclusión.
Excepciones	-
Importancia	Media

Tabla 43. Especificación del CU-15

CU-16	Contactar con soporte
Versión	1.0 (20/06/2024)
Actor Principal	Usuario
Actor Secundario	Administrador
Dependencias	<ul style="list-style-type: none"> • RU-16 El usuario registrado debe poder contactar con el soporte a través de email.
Precondición	El usuario debe estar autenticado en la aplicación.
Descripción	1. Permite al usuario contactar con el soporte a través de email.
Secuencia Normal	<ol style="list-style-type: none"> 1. El usuario selecciona la opción de contactar con soporte. 2. El sistema muestra un formulario para ingresar el mensaje de soporte. 3. El usuario completa el formulario y lo envía. 4. El sistema envía el mensaje al equipo de soporte vía email.
Postcondición	El mensaje será enviado al equipo de soporte.
Excepciones	-
Importancia	Media

Tabla 44. Especificación del CU-16

CU-17	Gestionar contenidos y usuarios
Versión	1.0 (20/06/2024)
Actor Principal	Administrador
Actor Secundario	Usuario
Dependencias	<ul style="list-style-type: none"> • RU-17 El usuario administrador debe poder eliminar noticias. • RU-18 El usuario administrador debe poder eliminar usuarios. • RU-19 El usuario administrador debe poder eliminar sugerencias de nuevas fuentes. • RU-20 El usuario administrador debe poder aceptar las sugerencias de los usuarios. • RU-21 El usuario administrador debe poder eliminar comentarios. • RU-22 El usuario administrador debe poder ver analíticas sobre las noticias.

Precondición	El administrador debe estar autenticado en la aplicación.
Descripción	Permite al administrador eliminar noticias, usuarios, comentarios y sugerencias, aceptar sugerencias y ver analíticas sobre las noticias.
Secuencia Normal	<ol style="list-style-type: none"> 1. El administrador selecciona la opción de gestionar contenidos y usuarios. 2. El sistema muestra una interfaz con opciones para eliminar noticias, usuarios, comentarios y sugerencias. 3. El administrador selecciona la acción a realizar y confirma la operación. 4. El sistema realiza la operación seleccionada y actualiza la interfaz.
Postcondición	Las acciones realizadas por el administrador se reflejarán en el sistema, manteniendo la calidad y el orden del contenido.
Excepciones	-
Importancia	Alta

Tabla 45. Especificación del CU-17

3.3 Requisitos funcionales

En esta sección se introducirán los diferentes requisitos funcionales que definirán el comportamiento del software. Los requisitos funcionales describen con precisión el comportamiento del software, las funciones que se deben incluir para cumplir y completar las actividades de los usuarios sistema. [13]

RU-ID	RF-ID	Descripción del Requisito Funcional
RU-01	RF-01	El sistema pedirá al visitante que rellene el formulario de registro para obtener una cuenta de usuario.
	RF-02	El sistema debe de comprobar que tanto el correo electrónico como el apodo del usuario son únicos.
	RF-03	El sistema debe de comprobar que las contraseña y la confirmación de esta tengan el mismo valor,
	RF-04	El sistema enviará un email a los usuarios dados de alta en la aplicación para que confirmen su cuenta.
RU-02	RF-05	El sistema pedirá al usuario introducir su correo electrónico y contraseña para poder iniciar sesión en la aplicación.
	RF-06	En caso de que el formulario enviado no corresponda con los datos almacenados en la plataforma, el sistema solicitará al usuario que vuelva a intentarlo, indicando el error de autenticación.
	RF-07	El sistema devolverá un mensaje de error si el usuario se ha registrado, pero no ha validado la cuenta a través del correo que se le ha mandado.
	RF-08	Una vez iniciada la sesión, el sistema se conectará con el protocolo de autorización de APIs OAuth 2.0 para proporcionar acceso a los recursos requeridos por el usuario usando un token (JWT).
RU-03	RF-09	El sistema permitirá al usuario recuperar su contraseña proporcionando su correo electrónico para recibir un enlace de restablecimiento.
	RF-10	El sistema debe solicitar al usuario que ingrese la nueva contraseña dos

		veces al seguir el enlace de recuperación, para asegurar que no haya errores de entrada y prevenir olvidos.
RU-04	RF-11	El sistema permitirá al usuario cerrar sesión en cualquier momento a través de un botón que se encontrará al seleccionar el perfil.
RU-05	RF-12	El sistema permitirá a los usuarios no registrados (visitantes) ver y leer noticias solo en español.
RU-06	RF-13	El sistema permitirá al usuario ver y leer noticias en su idioma preferido.
	RF-14	El sistema debe traducir automáticamente las noticias al inglés, alemán, ruso, árabe, italiano, francés y portugués al momento de su almacenamiento.
	RF-15	El sistema utilizará dos APIs gratuitas de traducción para traducir las noticias.
	RF-16	Si se alcanza el máximo de peticiones diarias a las APIs gratuitas de traducción, el sistema debe usar una API de traducción local para continuar traduciendo las noticias.
RU-07	RF-17	El sistema permitirá al usuario buscar noticias por categoría o por fuente.
	RF-18	El sistema deberá proporcionar un formulario para filtrar noticias por categoría y fuente, utilizando switches para que los usuarios puedan seleccionar fácilmente las opciones deseadas.
RU-08	RF-19	El sistema deberá permitir al usuario registrado marcar una noticia como favorita mediante un ícono de un corazón ubicado junto a cada noticia.
	RF-20	El sistema deberá proporcionar una confirmación visual (como un cambio de color en el corazón) cuando una noticia sea marcada como favorita.
RU-09	RF-21	El sistema deberá proporcionar una sección en la aplicación donde se liste todas las noticias marcadas como favoritas por el usuario.
	RF-22	El sistema deberá permitir al usuario registrado acceder a la lista de noticias favoritas desde el menú principal.
	RF-23	El sistema deberá permitir al usuario eliminar noticias de su lista de favoritas mediante el mismo ícono del corazón, que se encontrará junto a cada noticia en la lista.
RU-10	RF-24	El sistema permitirá que el usuario registrado pueda escribir comentarios en las noticias mediante un campo de texto disponible debajo de cada noticia.
	RF-25	El sistema deberá almacenar los comentarios realizados por los usuarios registrados en la base de datos, asociados con la noticia correspondiente y el usuario que lo realizó.
RU-11	RF-26	El sistema deberá permitir al usuario registrado editar sus comentarios mediante un botón de edición junto a cada comentario que haya realizado.
	RF-27	El sistema deberá permitir al usuario registrado eliminar sus comentarios mediante un botón de eliminación junto a cada comentario que haya realizado.
	RF-28	El sistema deberá actualizar o eliminar el comentario editado en la base de datos y reflejar los cambios en la interfaz de usuario inmediatamente.
RU-12	RF-29	El sistema deberá mostrar todos los comentarios ordenados cronológicamente. realizados por los usuarios debajo de cada noticia.

	RF-30	El sistema deberá mostrar el apodo del usuario y la fecha de creación junto a cada comentario.
RU-13	RF-31	El sistema deberá proporcionar una sección en el perfil del usuario donde se liste todos los comentarios realizados por ese usuario.
	RF-32	El sistema deberá permitir al usuario acceder a la lista de comentarios realizados desde el menú principal.
	RF-33	El sistema deberá mostrar cada comentario realizado por el usuario junto con la noticia correspondiente y la fecha de creación.
RU-14	RF-34	El sistema deberá permitir al usuario registrado acceder a la sección de gestión de perfil desde el menú principal.
	RF-35	El sistema deberá proporcionar un formulario en la sección de Mi Perfil donde el usuario podrá actualizar su información personal.
	RF-36	El sistema deberá almacenar las actualizaciones del perfil en la base de datos y reflejar los cambios inmediatamente en la interfaz de usuario
RU-15	RF-37	El sistema proporcionará una opción en la sección de Mi Perfil para que el usuario registrado pueda eliminar su cuenta.
	RF-38	El sistema deberá solicitar confirmación al usuario antes de eliminar su cuenta a través de un <i>popup</i> .
	RF-39	El sistema permitirá al usuario sugerir nuevas fuentes de noticias al administrador.
	RF-40	Cuando se elimine la cuenta de un usuario, también se deberá de eliminar todos los comentarios y noticias favoritas asociados al usuario.
RU-16	RF-41	El sistema deberá proporcionar un formulario donde el usuario registrado pueda sugerir nuevas fuentes de noticias.
	RF-42	El sistema deberá permitir al usuario registrado acceder a la sección de sugerir fuentes desde el menú principal
	RF-43	El sistema deberá mostrar un listado de las diferentes sugerencias que haya realizado junto con el estado en el que se encuentra (Enviado/Confirmado).
RU-17	RF-44	El sistema deberá proporcionar un formulario de contacto en el que se le solicitará al usuario que rellene el campo asunto y mensaje.
	RF-45	El sistema enviará el mensaje del usuario registrado al correo electrónico del equipo de soporte.
	RF-46	El sistema deberá notificar al usuario registrado que su mensaje ha sido enviado correctamente.
RU-18	RF-47	El sistema proporcionará una interfaz para que el administrador pueda ver la lista de noticias publicadas.
	RF-48	El sistema deberá permitir al administrador seleccionar y eliminar noticias específicas de la lista de noticias.
	RF-49	El sistema deberá eliminar las noticias seleccionadas, junto con todos los comentarios y favoritos asociados a estas.
RU-19	RF-50	El sistema deberá proporcionar una interfaz para que el administrador pueda ver la lista de usuarios registrados.
	RF-51	El sistema deberá permitir al administrador seleccionar y eliminar usuarios específicos de la lista.
	RF-52	El sistema deberá eliminar los usuarios seleccionados, junto con todos los

		comentarios y noticias favoritas asociados a dichos usuarios.
RU-20	RF-53	El sistema deberá proporcionar una interfaz para que el administrador pueda ver la lista de sugerencias de nuevas fuentes enviadas por los usuarios.
	RF-54	El sistema deberá permitir al administrador seleccionar y eliminar sugerencias específicas de la lista.
RU-21	RF-55	El sistema deberá permitir al administrador seleccionar y eliminar comentarios específicos de la lista.
RU-22	RF-56	El sistema deberá proporcionar una interfaz para que el administrador pueda ver la lista de comentarios realizados en las noticias.
	RF-57	El sistema deberá permitir al administrador seleccionar y eliminar comentarios específicos de la lista.
RU-23	RF-58	El sistema deberá proporcionar una interfaz de analíticas donde el administrador pueda ver datos sobre las noticias, tales como el número de vistas y comentarios.
	RF-59	El sistema deberá permitir al administrador filtrar las analíticas por rangos de fechas, como la última semana (días) o el último año (meses).

Tabla 46. Requisitos Funcionales

3.3.1 Requisitos de información

Esta sección contiene los requisitos de almacenamiento de información (requisitos de información) que se han identificado para OVERNEWS. Estos requisitos especifican qué información debe almacenar el sistema para poder ofrecer la funcionalidad descrita en los casos de uso del sistema o en otros requisitos.

ID	Descripción
RI-01	Los datos que se le solicitará en el formulario de registro son el nombre completo, correo electrónico, contraseña, domicilio y el idioma.
RI-02	En la sección gestión del perfil, el usuario podrá modificar su nombre, contraseña, domicilio o el idioma.
RI-03	El sistema generará y almacenará el ID de las noticias encriptando el título de la noticia utilizando el algoritmo SHA-256.
RI-04	El sistema almacenará por cada una de las noticias el id generado, el título de la noticia, el contenido, la categoría, la fuente de origen, la url de la noticia, la url de la imagen y la fecha de publicación.
RI-05	Cuando el usuario mande una sugerencia de nuevas fuentes de noticias al administrador, el sistema debe almacenar la fuente sugerida, el usuario que realizó la sugerencia y la fecha de la sugerencia.
RI-06	De los comentarios realizados en las noticias, se almacenará el texto del comentario, autor del comentario, la noticia en la que se ha comentado y la fecha de creación.
RI-07	Cuando un usuario marque como favorita una noticia, se almacenará el id de la noticia y el id del usuario que ha realizado la acción

RI-08	El sistema debe almacenar las traducciones de las noticias en archivos JSON temporales dentro de la carpeta /static/news, organizados en diferentes carpetas por idioma (news_en, news_de, news_it, etc.)
RI-09	A las 00:00h, el sistema debe rotar los archivos de traducción diariamente, moviendo news_<idioma>1 a news<idioma>2, y así sucesivamente hasta news<idioma>_7. De esta forma las traducciones de las noticias solo están en la aplicación durante 7 días.

Tabla 47. Requisitos de información

3.4 Requisitos no funcionales

Los requisitos no funcionales determinan cómo de bien el sistema realiza sus funciones. Estos requisitos abordan características esenciales como disponibilidad, usabilidad, seguridad y rendimiento, garantizando una experiencia de alta calidad para el usuario. Además, describen el entorno en el que el sistema opera, asegurando su compatibilidad con diferentes plataformas, su portabilidad, y su conformidad con idiomas, monedas, leyes y terminología específicos. [13]

ID	Tipo	Descripción
RNF-01	Rendimiento	El sistema debe ser capaz de manejar al menos 1000 solicitudes concurrentes sin degradar el rendimiento.
RNF-02	Rendimiento	El tiempo de respuesta para cargar una página de noticias debe ser inferior a 2 segundos.
RNF-03	Rendimiento	El sistema debe realizar la traducción de todas las noticias en menos de 1 hora.
RNF-04	Seguridad	El sistema debe asegurar que todos los datos de usuario (como contraseñas) estén encriptados utilizando un algoritmo seguro (SHA-256).
RNF-05	Seguridad	El sistema debe implementar autenticación y autorización basadas en JWT para todas las operaciones que requieran acceso de usuario registrado.
RNF-06	Seguridad	Las conexiones al sistema deben ser seguras y encriptadas utilizando HTTPS.
RNF-07	Usabilidad	La interfaz de usuario debe ser intuitiva y fácil de usar, permitiendo a los usuarios acceder a las funcionalidades principales en no más de tres clics.
RNF-08	Usabilidad	El sistema debe ser accesible y usable en dispositivos móviles y de escritorio.
RNF-09	Usabilidad	El sistema debe proporcionar mensajes de error claros y útiles para ayudar a los usuarios a resolver problemas.
RNF-10	Escalabilidad	El sistema debe ser escalable para soportar el aumento de usuarios y volumen de datos sin necesidad de rediseñar la arquitectura.

RNF-11	Mantenibilidad	El código fuente del sistema debe estar bien documentado para facilitar futuras actualizaciones y mantenimiento.
RNF-12	Fiabilidad	El sistema debe tener una disponibilidad del 99.9%, garantizando que esté operativo casi todo el tiempo.
RNF-13	Compatibilidad	El sistema debe ser compatible con los navegadores web más utilizados, incluyendo Chrome, Firefox, Safari y Edge.
RNF-14	Localización e Internacionalización	El sistema debe soportar la visualización de noticias en al menos siete idiomas: inglés, alemán, ruso, árabe, italiano, francés y portugués.
RNF-15	Localización e Internacionalización	El sistema debe permitir la adición de nuevos idiomas de manera sencilla y sin requerir cambios significativos en el código.
RNF-16	Seguridad	Todos los accesos al servidor de recursos deberán ser estrictamente validados por el servidor de autorización siguiendo el protocolo de autorización OAuth2.0.
RNF-17	Restricción	El <i>Backend</i> será escrito en el lenguaje de programación Python utilizando FastAPI.
RNF-18	Restricción	El <i>Frontend</i> llevará React JS por debajo utilizando Material UI como librería de componentes.

Tabla 48. Requisitos no funcionales

Capítulo 4

Diseño

4.1 Arquitectura Lógica

En esta sección se definirá la arquitectura lógica del proyecto. Elegir la arquitectura adecuada es uno de los puntos más críticos y relevantes del sistema. Tras haber analizado las características y requisitos en la parte de Análisis (Sección 3), se concluye que la arquitectura más adecuada es la arquitectura Modelo-Vista-Controlador (MVC).

Esta arquitectura se caracteriza por componerse de tres capas, la capa de presentación (Vista), la capa de negocio (Controlador) y la capa de datos (Modelo). De esta forma la aplicación se estructura en tres capas principales: presentación, negocio y datos.

La capa de presentación se encarga de la interacción con el usuario final. Los componentes de esta capa incluyen la aplicación frontend, desarrollada en *ReactJS*, y los *DTOs* (*Data Transfer Objects*), que se utilizan para transferir datos entre el *frontend* y los *routers*. Es decir, que en esta capa se encuentra toda la información que se muestra al usuario.

La capa de negocio contiene la lógica de la aplicación. En esta capa se van a encontrar los *routers*, que actúan como controladores, y todos los servicios que se llaman desde estos. Los servicios contienen la lógica de negocio y, cuando es necesario, interactúan con la capa de datos para obtener o almacenar información.

La capa de datos es la encargada de manejar tanto el acceso a la base de datos, como las diferentes interacciones de la aplicación con los servicios externos. Es por esto que, en esta capa se van a encontrar los diferentes modelos de datos (*models*) o entidades y el ORM (*Peewee*), que se encarga de gestionar la interacción con la base de datos. El último componente de esta capa, son los diferentes *scripts* encargados de adquirir las noticias de las diferentes fuentes, y de traducir las noticias a los 7 idiomas definidos en esta primera versión. El *script* que llama tanto a la obtención de las noticias como la traducción es `update-news.py`.

La decisión de que los scripts de traducción y obtención de noticias se encuentren en la capa de datos se basa en que la capa de negocio agrupe únicamente la lógica de negocio y las reglas del sistema, sin preocuparse por los detalles de la interacción con APIs externas o el manejo de datos.

A continuación, se muestra la arquitectura lógica.

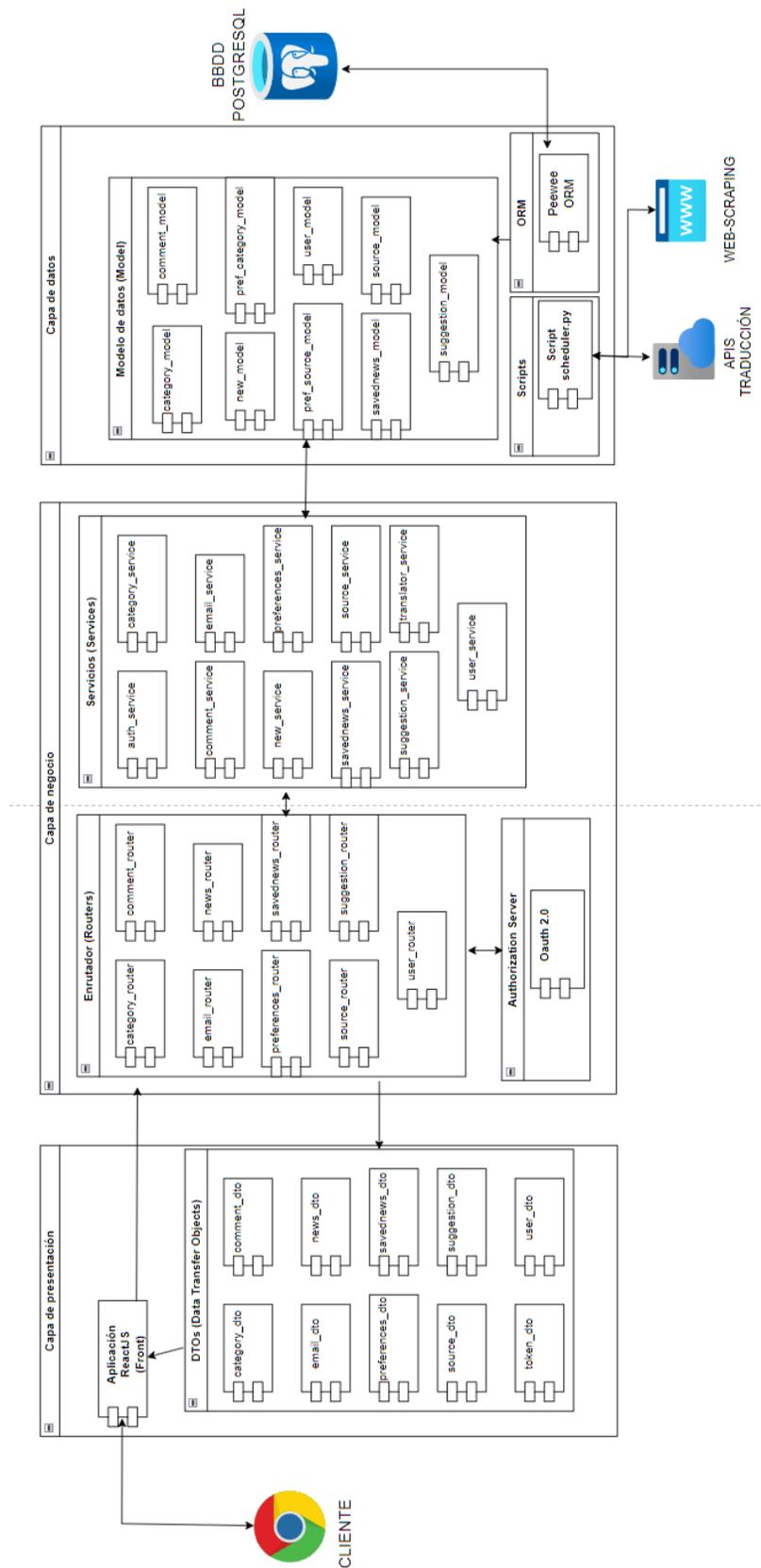


Figura 17. Arquitectura lógica

4.2 Arquitectura Física

En esta sección, se mostrará la arquitectura física que caracteriza a **OVERNEWS**. La arquitectura física expresa cuáles son los componentes físicos (cliente, servidor, servidores web, bases de datos, etc.) que participan en el funcionamiento de la aplicación, así como la relación entre ellos. [14]

En la siguiente figura, se observarán todos estos componentes físicos.

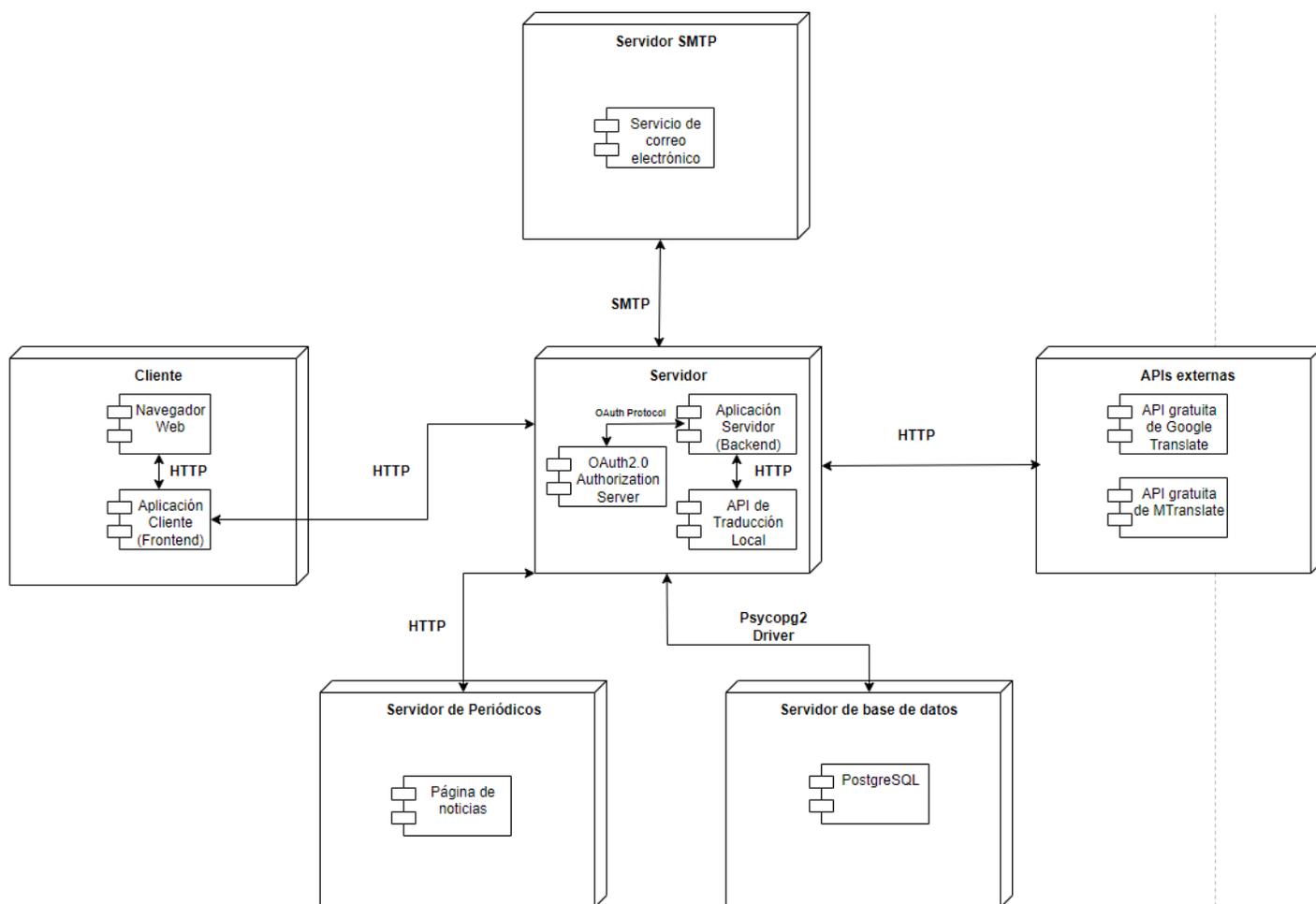


Figura 18. Arquitectura Física

4.3 Diagramas (secuencia y actividad)

En esta sección se van a mostrar algunos de los diagramas de secuencia que son más relevantes.

Los diagramas de secuencia son diagramas que modelan la interacción entre los objetos de nuestro sistema y en dónde también se identifica la comunicación y las operaciones entre los mismos. [15]

Los diagramas de secuencia que pueden aportar mayor valor para comprender el funcionamiento de la aplicación son el de iniciar sesión (Sección 4.3.1), el de obtención y traducción de las noticias (Sección 4.3.2) y el de mostrar noticias (Sección 4.3.3).

4.3.1 Diagrama de secuencia iniciar sesión.

A la hora de iniciar sesión, es importante aclarar que se usa la autenticación JWT (*JSON Web Token*). De esta forma si existe un token válido en el Local Storage, el usuario ya estaría iniciado, y por lo tanto es redirigido directamente a la página principal.

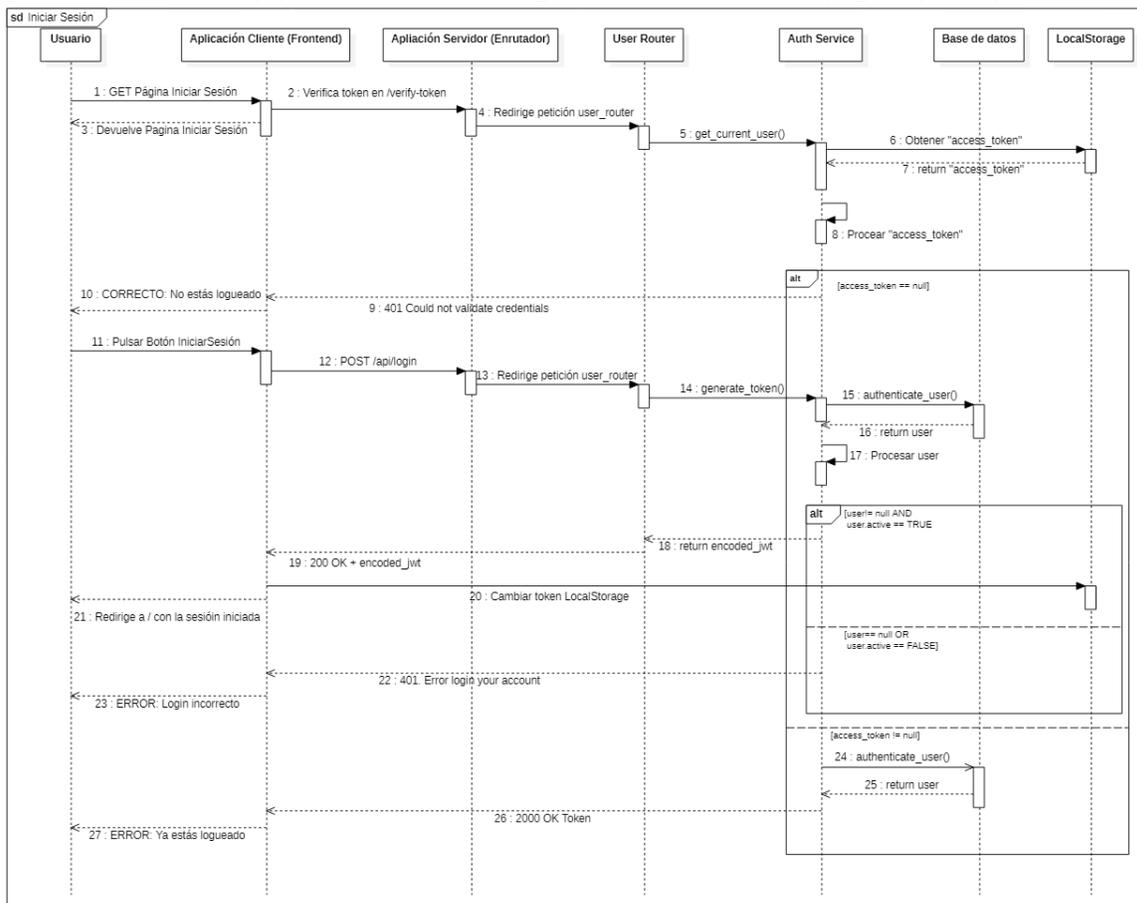


Figura 19. Diagrama de Secuencia Iniciar Sesión

En la imagen anterior, se muestra el diagrama de secuencia desde el momento en el que el usuario pulsa el botón de `Iniciar Sesión` desde el menú principal.

4.3.2 Diagrama de secuencia obtención y traducción de noticias

Es importante tener en cuenta que el proceso de obtención y traducción de noticias se ejecuta automáticamente cada hora mediante un planificador (`scheduler.py`), garantizando que las noticias estén siempre actualizadas. Las noticias se obtienen mediante un *script* de *web scraping* y se almacenan únicamente en la base de datos las noticias en español. Las traducciones se almacenan en archivos JSON organizados por idioma y según el día en el que se ha obtenido. Además, la traducción a los diferentes idiomas se realiza de forma paralela, permitiendo una mayor eficiencia al traducir simultáneamente en varios idiomas.

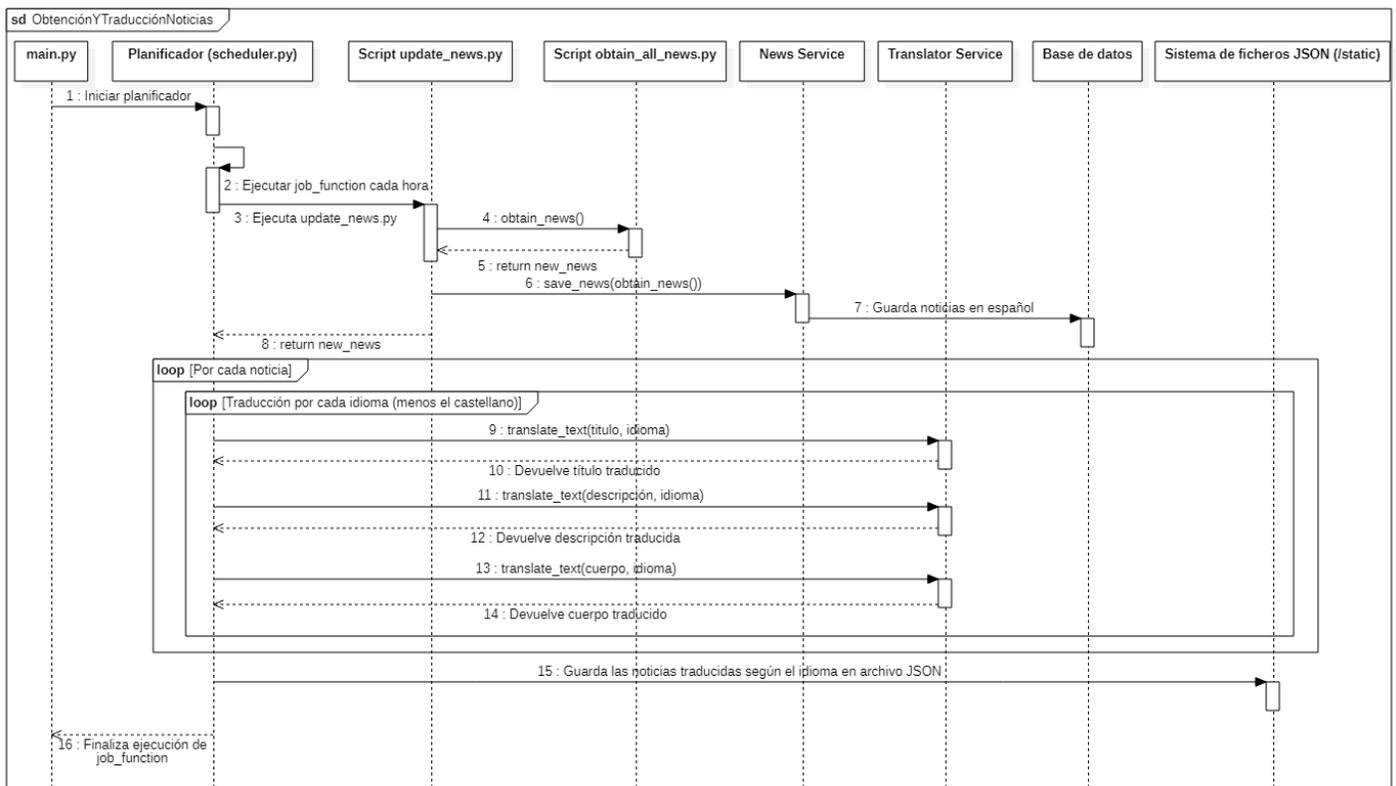


Figura 20. Diagrama de secuencia obtención y traducción de noticias

4.3.3 Diagrama de secuencia mostrar noticias

En el siguiente diagrama de secuencia se describe el proceso completo para obtener y mostrar noticias en la aplicación.

A la hora de obtener y mostrar las noticias es importante comprender cómo se almacenan las diferentes noticias en los diferentes idiomas dentro de la aplicación. Toda la información acerca de las noticias se va a almacenar en diferentes carpetas según su idioma como he comentado anteriormente en la Sección 4.3.2. Cada una de estas carpetas contiene 7 ficheros JSON nombrados de la siguiente forma: `news_{idioma}_{number}`. El valor de `number` se va a almacenar en la base de datos y nos va a indicar en qué número de fichero se encuentra la noticia traducida. Téngase en cuenta que las noticias se rotan diariamente con un horizonte de 7 días, y el valor de `number` indica en qué archivo de rotación se encuentra dicha noticia. Este valor se actualiza cuando sea pertinente.

Es por esto, que lo primero que se hace es obtener el `id` de las noticias requeridas de la base de datos (ya que en la base de datos no se va a guardar toda la información de las noticias).

Estos `ids` se agruparán en función del `number` al que pertenezcan, para obtener las noticias de la forma más eficiente. De esta forma se consigue que se buscan todas las noticias de un determinado fichero JSON a la vez, en vez de obtener las noticias de uno en uno.

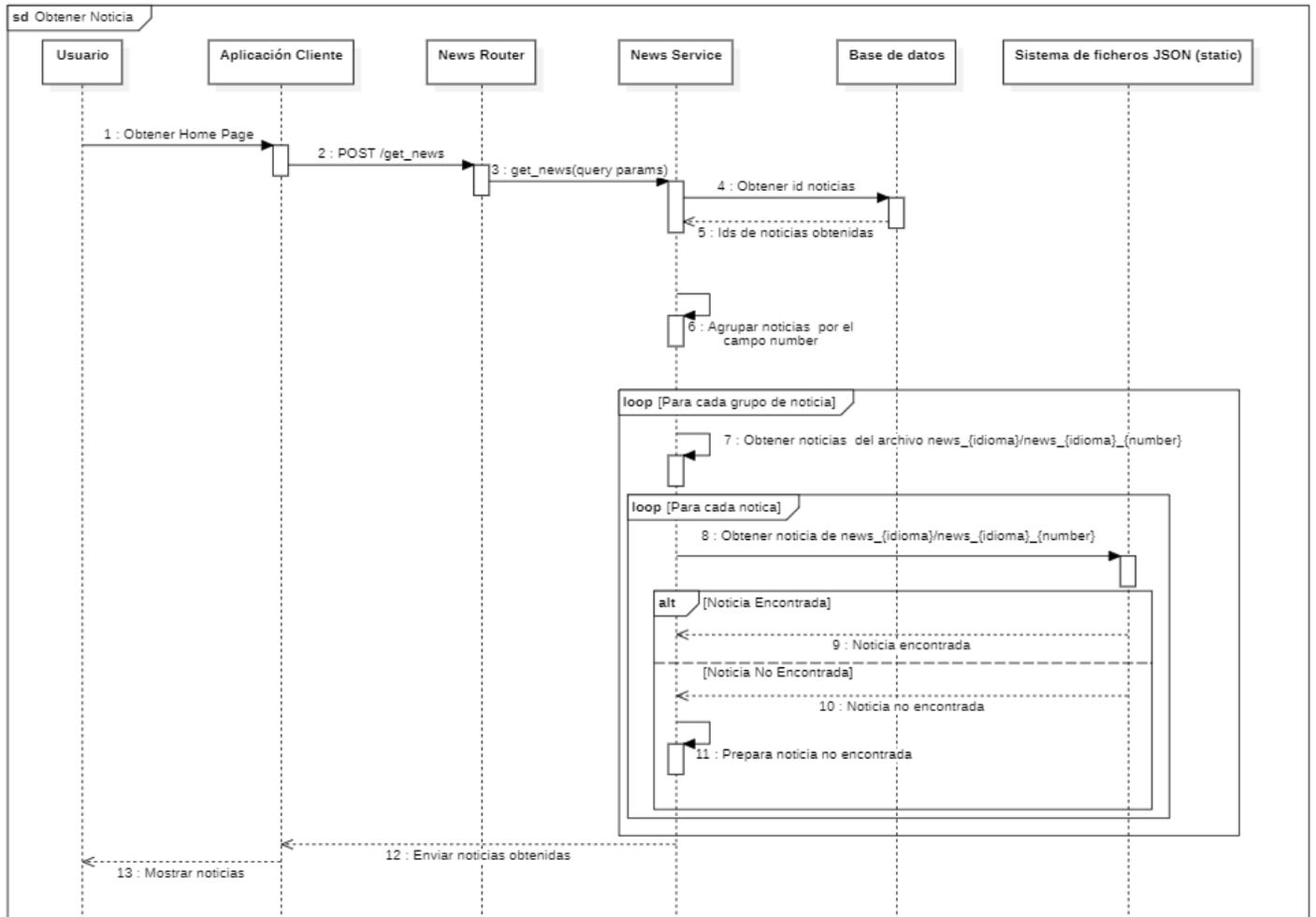


Figura 21. Diagrama de secuencia mostrar noticias

4.4 Modelo de datos

En esta sección, se detalla el modelo de datos de OVERNEWS, para obtener una visión clara de cómo se gestionan y almacenan los datos dentro de la aplicación.

El modelo de datos físicos es un componente esencial del modelado de datos, ya que representa la estructura del almacenamiento y las diferentes relaciones entre los objetos de la base de datos.

Para obtener esta visión, se detallará cómo se ha diseñado la base de datos paso a paso. Por lo tanto, primero se mostrará un análisis detallado del modelo entidad-relación (Sección 4.4.1), después se llevará a cabo el diseño lógico (Sección 4.4.2) y por último se detallarán los diferentes diccionarios de datos (Sección 4.4.3).

4.4.1 Modelo Entidad-Relación

En esta sección se verá en detalle el modelo entidad-relación o modelo conceptual de OVERNEWS. El modelo entidad relación describe de forma sencilla y libre ambigüedad la naturaleza de los datos corporativos. En este, además de identificar las diferentes entidades de datos más relevantes y sus relaciones, también se detallan los atributos y restricciones. [16]

En la siguiente imagen, se detalla el modelo entidad-relación.

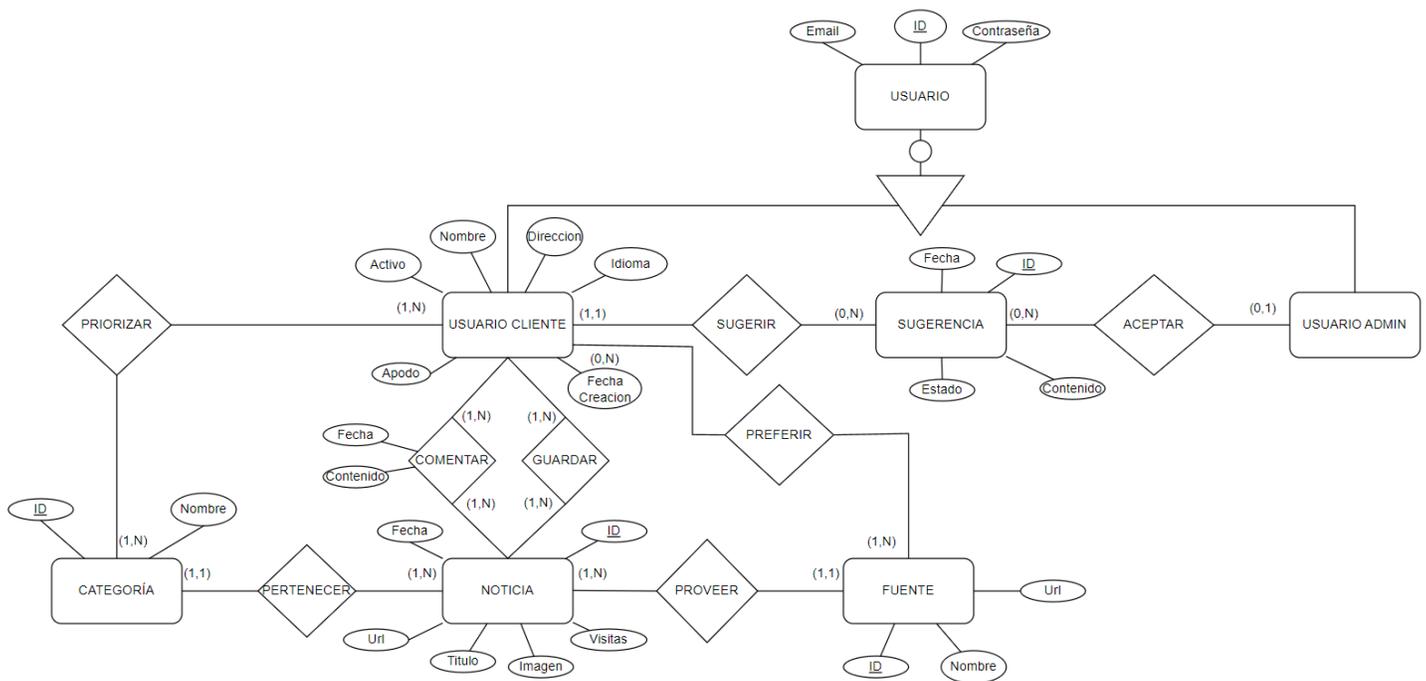


Figura 22. Modelo Entidad-Relación

4.4.2 Diseño lógico

En esta sección, se llevará a cabo el diseño lógico de la base de datos. El diseño lógico se centra en transformar el modelo conceptual de la base de datos en un modelo lógico detallado. Este proceso no solo garantiza que la estructura sea correcta, sino también que la base de datos pueda manejar eficientemente todas las transacciones necesarias. [17]

A continuación, se muestra el modelo lógico detallado:

USUARIO (Id, Email, Contraseña, Apodo, Nombre, Dirección, Idioma, Fecha_creación, Activado, Administrador)

SUGERENCIA (Id, Contenido, Fecha, Estado, Usuario)
FK: Usuario -> USUARIO(Id)

NOTICIA (Id, Título, Fecha Url, Imagen, Visitas, Número, Categoría, Fuente)
FK: Categoría -> CATEGORÍA(Id)
FK: Fuente -> FUENTE(Id)

CATEGORÍA (Id, Nombre)

FUENTE (Id, Nombre, Url)

PREFERENCIA_CATEGORÍA (Categoría, Usuario)
FK: Categoría -> CATEGORÍA(Id)
FK: Usuario -> USUARIO(Id)

PREFERENCIA_FUENTE (Fuente, Usuario)
FK: Fuente -> FUENTE (Id)
FK: Usuario -> USUARIO(Id)

COMENTARIO (Usuario, Noticia, Contenido, Fecha_publicación)
FK: Usuario -> USUARIO(Id)
FK: Noticia -> NOTICIA (Id)

NOTICIAS_GUARDADAS (Usuario, Noticia)
FK: Usuario -> USUARIO(Id)
FK: Noticia -> NOTICIA (Id)

4.4.3 Diseño físico

Hasta este momento, ya tenemos conocimiento del modelo conceptual (Sección 4.4.1) y del

modelo lógico de los datos en la aplicación OVERNEWS. Lo que faltaría mostrar el diseño físico de la base de datos.

Para mostrar el diseño físico de los datos, utilizaré los diccionarios de datos. Los diccionarios de datos proporcionan una descripción detallada de los elementos de datos (entidades) dentro de una base de datos.

4.4.3.1 Diccionario de datos

A continuación, se muestran los diccionarios de datos que componen la base de datos:

Name	Data type	Length/Precision	Scale	Not NULL?	Primary key?
id	integer v			<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
email	character varying v	255		<input checked="" type="checkbox"/>	<input type="checkbox"/>
username	character varying v	255		<input checked="" type="checkbox"/>	<input type="checkbox"/>
password	character varying v	255		<input checked="" type="checkbox"/>	<input type="checkbox"/>
full_name	character varying v	255		<input checked="" type="checkbox"/>	<input type="checkbox"/>
address	character varying v	255		<input type="checkbox"/>	<input type="checkbox"/>
language	character varying v	255		<input checked="" type="checkbox"/>	<input type="checkbox"/>
isAdmin	boolean v			<input checked="" type="checkbox"/>	<input type="checkbox"/>
is_active	boolean v			<input checked="" type="checkbox"/>	<input type="checkbox"/>
created_at	timestamp without t... v			<input checked="" type="checkbox"/>	<input type="checkbox"/>

Figura 23. Diccionario de datos "USER" (Usuario)

Name	Data type	Length/Precision	Scale	Not NULL?	Primary key?
id	integer v			<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
content	text v			<input checked="" type="checkbox"/>	<input type="checkbox"/>
date	date v			<input checked="" type="checkbox"/>	<input type="checkbox"/>
client_id	integer v			<input checked="" type="checkbox"/>	<input type="checkbox"/>
state	boolean v			<input checked="" type="checkbox"/>	<input type="checkbox"/>

Figura 24. Diccionario de datos "SUGGESTION" (Sugerencias)

Name	Data type	Length/Precision	Scale	Not NULL?	Primary key?
id	character varying v	255		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
title	character varying v	255		<input checked="" type="checkbox"/>	<input type="checkbox"/>
category_id	integer v			<input checked="" type="checkbox"/>	<input type="checkbox"/>
source_id	integer v			<input checked="" type="checkbox"/>	<input type="checkbox"/>
url	character varying v	255		<input checked="" type="checkbox"/>	<input type="checkbox"/>
image	character varying v	255		<input checked="" type="checkbox"/>	<input type="checkbox"/>
date	date v			<input checked="" type="checkbox"/>	<input type="checkbox"/>
view	integer v			<input checked="" type="checkbox"/>	<input type="checkbox"/>
number	integer v			<input checked="" type="checkbox"/>	<input type="checkbox"/>

Figura 25. Diccionario de datos "NEW" (Noticia)

Name	Data type	Length/Precision	Scale	Not NULL?	Primary key?
id	integer v			<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
name	character varying v	255		<input checked="" type="checkbox"/>	<input type="checkbox"/>

Figura 26. Diccionario de datos "CATEGORY" (Categoría)

Name	Data type	Length/Precision	Scale	Not NULL?	Primary key?
id	integer			<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
name	character varying	255		<input checked="" type="checkbox"/>	<input type="checkbox"/>
url	character varying	255		<input checked="" type="checkbox"/>	<input type="checkbox"/>

Figura 27. Diccionario de datos "SOURCE" (Fuente)

Name	Data type	Length/Precision	Scale	Not NULL?	Primary key?
id	integer			<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
category_id	integer			<input checked="" type="checkbox"/>	<input type="checkbox"/>
user_id	integer			<input checked="" type="checkbox"/>	<input type="checkbox"/>

Figura 28. Diccionario de datos "PREF_CATEGORY" (Preferencia_Categoría)

Name	Data type	Length/Precision	Scale	Not NULL?	Primary key?
id	integer			<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
source_id	integer			<input checked="" type="checkbox"/>	<input type="checkbox"/>
user_id	integer			<input checked="" type="checkbox"/>	<input type="checkbox"/>

Figura 29. Diccionario de datos "PREF_SOURCE" (Preferencia_Fuente)

Name	Data type	Length/Precision	Scale	Not NULL?	Primary key?
id	integer			<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
content	text			<input checked="" type="checkbox"/>	<input type="checkbox"/>
publication_date	date			<input checked="" type="checkbox"/>	<input type="checkbox"/>
user_id	integer			<input checked="" type="checkbox"/>	<input type="checkbox"/>
news_id	character varying	255		<input checked="" type="checkbox"/>	<input type="checkbox"/>

Figura 30. Diccionario de datos "COMMENT" (Comentario)

Name	Data type	Length/Precision	Scale	Not NULL?	Primary key?
id	integer v			<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
user_id	integer v			<input checked="" type="checkbox"/>	<input type="checkbox"/>
news_id	character varying v	255		<input checked="" type="checkbox"/>	<input type="checkbox"/>

Figura 31. Diccionario de datos "SAVEDNEW" (Noticia_Guardada)

4.4.3.2 Consideraciones

Al diseñar el modelo físico de datos para OVERNEWS, se han tomado en cuenta varias consideraciones importantes.

La primera consideración es que las entidades **PREF_CATEGORY**, **PREF_SOURCE**, **COMMENT**, y **SAVEDNEWS**, he decidido agregarles un identificador propio (Id) ya que facilita la gestión de las transacciones con la base de datos. Este se debe a que permite filtrar y operar sobre estas entidades utilizando un solo atributo en lugar de dos atributos (claves primarias compuestas).

También resalta que en el modelo físico únicamente se han mostrado cómo se almacenan los datos persistentes. Las noticias traducidas no son consideradas datos persistentes, ya que el ciclo de vida de una noticia es de 7 días. En las siguientes secciones, se detallará cómo se gestionan y almacenan tanto las noticias completas como las noticias traducidas.

Por último, resaltar que tanto las entidades como los atributos en el modelo físico están en inglés. Esta elección no es aleatoria, sino que quiero con esto es facilitar la comprensión y colaboración a nivel internacional.

4.5 Diseño de las Interfaces de Usuario

En esta última sección del Diseño (Sección 4) de la aplicación, se mostrará el diseño de las diferentes interfaces de usuario que tendrá OVERNEWS. Las interfaces se han diseñado cuidadosamente para que el usuario tenga en todo momento una experiencia intuitiva y agradable.

Para mostrar el diseño de las diferentes interfaces, se utilizarán tablas específicas para cada interfaz.

Estas tablas tienen los siguientes atributos:

- Descripción: Una breve explicación de la funcionalidad y el propósito de la interfaz.
- Activación: Las acciones que desencadenan la visualización de la interfaz.
- Boceto: Una representación visual de la interfaz.
- Eventos: Las diferentes interacciones que se pueden realizar en la interfaz.
- Elementos: Una lista de los componentes principales de la interfaz.

A continuación, se muestran estas tablas para cada interfaz de la aplicación.

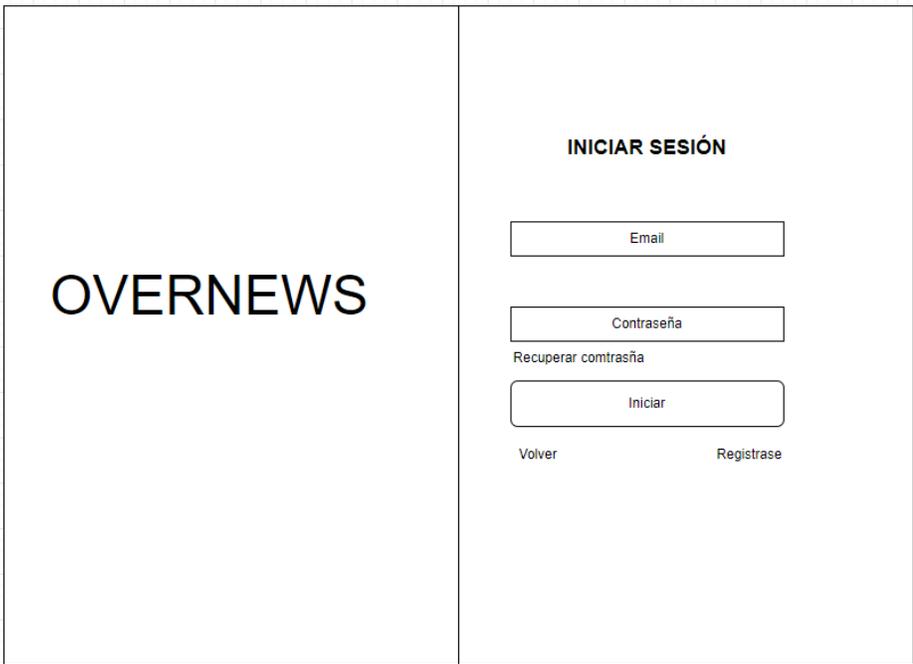
Iniciar Sesión	
Descripción	Interfaz donde los usuarios pueden iniciar sesión en OVERNEWS .
Activación	Cuando el usuario hace click en el botón "Iniciar Sesión" desde la página principal o la página de registro.
Boceto	
<i>Figura 32. Boceto Iniciar Sesión</i>	
Eventos	<ul style="list-style-type: none"> • Click en Iniciar Sesión: Envía la información del formulario y comprueba autenticación. • Click en "Recuperar contraseña": Redirige a la página de recuperación de contraseña. • Click en "Regístrate": Redirige a la página de registro • Click en "Volver": Redirige a la página anterior.
Elementos	<ol style="list-style-type: none"> 1) Campo de email del formulario. 2) Campo de contraseña del formulario. 3) Botón "Iniciar Sesión". 4) Enlace "Recuperar contraseña". 5) Enlace "Regístrate". 6) Enlace "Volver".

Tabla 49. Diseño Interfaz de Iniciar Sesión

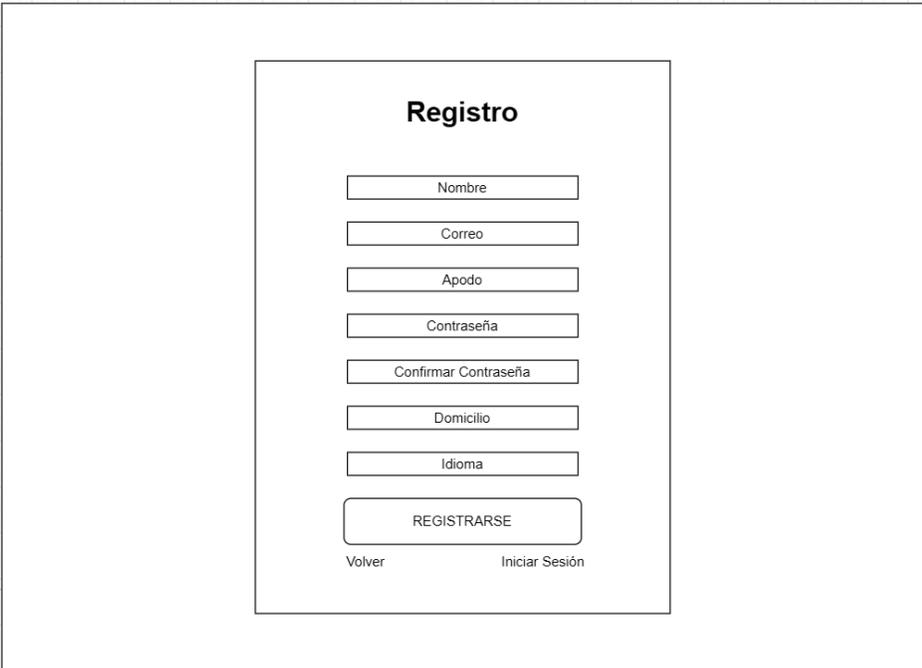
Registro	
Descripción	Interfaz donde los nuevos usuarios pueden registrarse en OVERNEWS .
Activación	La página se activa cuando el usuario hace click en el enlace "Regístrate" desde la página de inicio de sesión.
Boceto	 <p style="text-align: center;"><i>Figura 33. Boceto Registro</i></p>
Eventos	<ul style="list-style-type: none"> • Click en el botón "Registrarse": Inicia el proceso de registro. • Click en "Inicia Sesión": Redirige a la página de Iniciar Sesión • Click en "Volver": Redirige a la página anterior.
Elementos	<ol style="list-style-type: none"> 1) Campo de nombre del formulario de registro. 2) Campo de email del formulario de registro. 3) Campo de apodo. 4) Campo de contraseña. 5) Campo de confirmar contraseña. 6) Campo de domicilio. 7) Campo de idioma. 8) Botón "Registrarse. 9) Enlace "Inicia Sesión". 10) Enlace "Volver".

Tabla 50. Diseño Interfaz de Registro

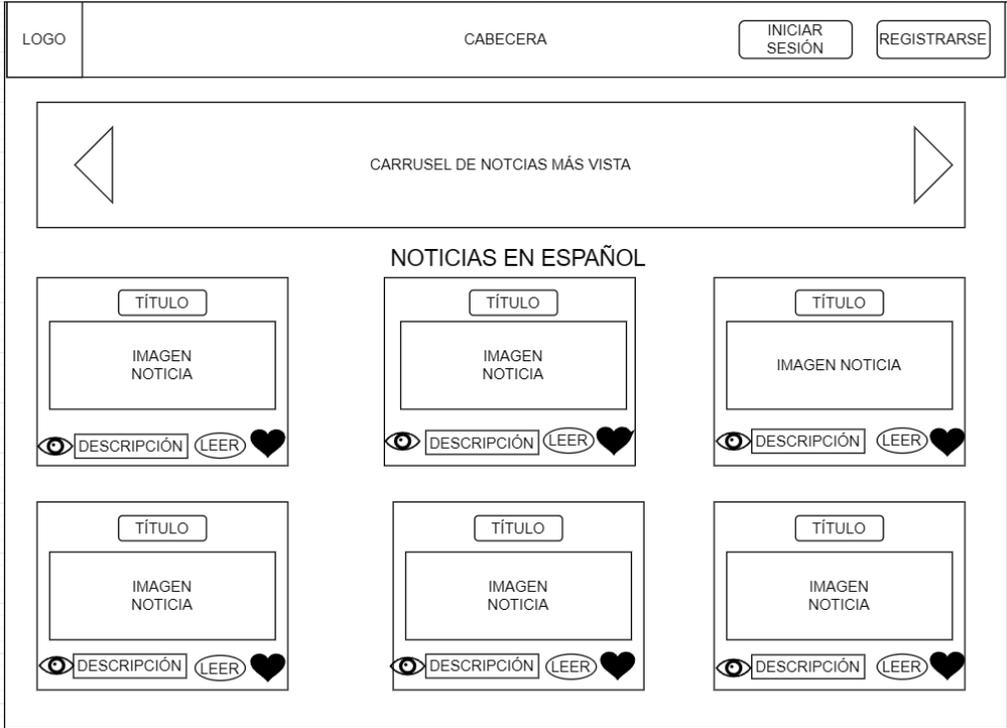
Página Principal (Usuarios No Registrados)	
Descripción	Interfaz donde los usuarios no registrados pueden ver y leer noticias en español.
Activación	Cuando un usuario no autenticado accede a OVERNEWS .
Boceto	 <p style="text-align: center;"><i>Figura 34. Boceto Página Principal (Usuarios No Registrados)</i></p>
Eventos	<ul style="list-style-type: none"> • Click en "Iniciar Sesión": Redirige a la página de inicio de sesión. • Click en "Registrarse": Redirige a la página de registro. • Click en las flechas del carrusel: Navega por las noticias más vistas. • Click en "Leer": Redirige a la página con el contenido completo de la noticia. • Click en el Icono de corazón: Muestra un mensaje para registrarse o iniciar sesión para poder guardar la noticia.
Elementos	<ol style="list-style-type: none"> 1) Cabecera. 2) Logo de OVERNEWS. 3) Carrusel de noticias más vistas. 4) Noticias en español. 5) Botón "Iniciar Sesión". 6) Botón "Registrarse" 7) Botón "Leer". 8) Icono de corazón. 9) Icono del ojo donde se muestran el número de visitas de la noticia.

Tabla 51. Diseño Interfaz de Página Principal (Usuarios No Registrados)

Página Principal (Usuarios Registrados)	
Descripción	Interfaz donde los usuarios registrados pueden ver y leer noticias en su idioma preferido.
Activación	La página se activa cuando un usuario autenticado accede a OVERNEWS.
Boceto	 <p style="text-align: center;"><i>Figura 35. Boceto Página Principal (Usuarios Registrados)</i></p>
Eventos	<ul style="list-style-type: none"> • Click en el botón "Perfil": Redirige a la página de perfil del usuario. • Click en las flechas del carrusel: Navega por las noticias más vistas. • Click en "Leer": Redirige a la página con el contenido completo de la noticia. • Click en el icono de corazón: Guarda la noticia como favorita.
Elementos Clave	<ol style="list-style-type: none"> 1) Cabecera. 2) Carrusel de noticias más vistas. 3) Noticias en idioma del usuario 4) Botón "Perfil" 5) Botones "Leer". 6) Icono de corazón. 7) Icono del ojo donde se muestran el número de visitas de la noticia.

Tabla 52. Diseño Interfaz de Página Principal (Usuarios Registrados)

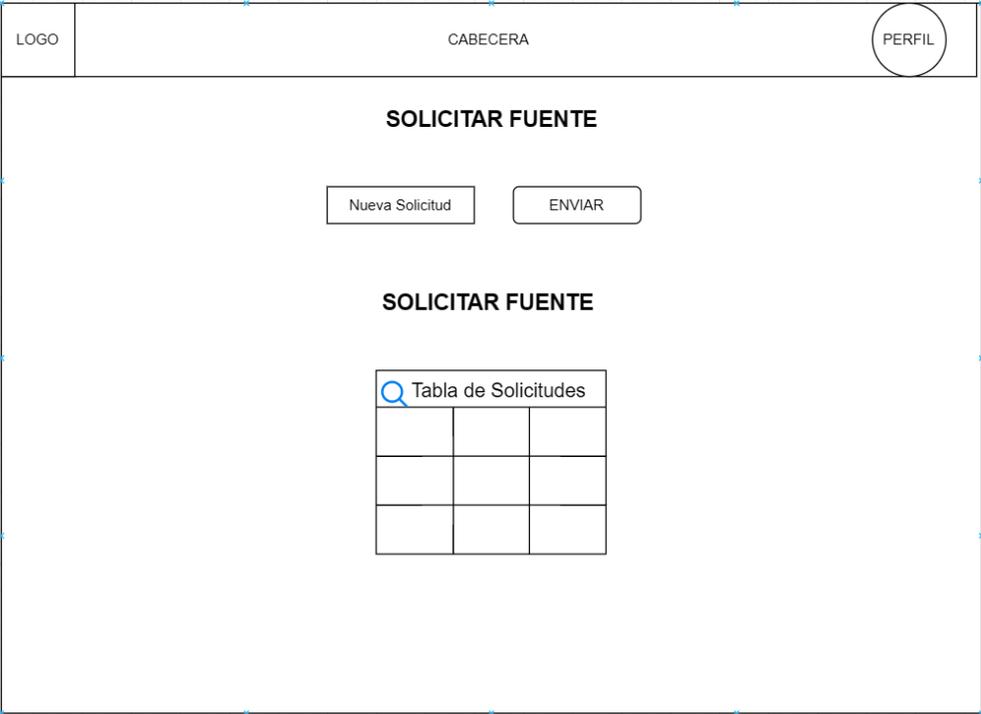
Solicitar Fuente	
Descripción	Interfaz donde los usuarios pueden solicitar nuevas fuentes y ver el estado de sus solicitudes anteriores.
Activación	La página se activa cuando el usuario autenticado selecciona la opción "Solicitar Fuente" en el menú de navegación.
Boceto	 <p style="text-align: center;"><i>Figura 36. Boceto Solicitar Fuente</i></p>
Eventos	<ul style="list-style-type: none"> • Click en el botón "Perfil": Redirige a la página de perfil del usuario. • Click en el botón "Enviar Solicitud": Envía la solicitud de nueva fuente al administrador. • Click en la lupa de búsqueda: Filtra las solicitudes basadas en el texto ingresado.
Elementos	<ol style="list-style-type: none"> 1) Cabecera. 2) Campo de la nueva solicitud. 3) Botón "Enviar Solicitud". 4) Botón "Perfil". 5) Tabla de solicitudes. 6) Lupa de búsqueda.

Tabla 53. Diseño Interfaz de Solicitar Fuente

Mis Noticias	
Descripción	Interfaz donde los usuarios pueden gestionar sus preferencias de noticias y ver sus noticias favoritas.
Activación	Cuando el usuario autenticado selecciona la opción "Mis Noticias" en el menú de navegación.
Boceto	<p style="text-align: center;"><i>Figura 37. Boceto Mis Noticias</i></p>
Eventos	<ul style="list-style-type: none"> • Click en el botón "Perfil": Redirige a la página de perfil del usuario. • Click en el botón "Guardar Ajustes": Guarda las preferencias de noticias del usuario. • Click en "Filtrar Noticias": Filtra las noticias según las preferencias.
Elementos	<ol style="list-style-type: none"> 1) Cabecera. 2) Botón "Perfil". 3) <i>Checkboxes</i> de categorías. 4) <i>Checkboxes</i> de fuentes. 5) Botón "Guardar Ajustes". 6) Botón "Filtrar Noticias". 7) Sección "Mis Favoritas".

Tabla 54. Diseño Interfaz de Mis Noticias

Mis Comentarios	
Descripción	Interfaz donde los usuarios pueden ver todos los comentarios que han realizado e ir a las noticias correspondientes.
Activación	La página se activa cuando el usuario autenticado selecciona la opción "Mis Comentarios" en el menú de navegación.
Boceto	<p style="text-align: center;"><i>Figura 38. Boceto "Mis Comentarios"</i></p>
Eventos	<ul style="list-style-type: none"> • Click en "Ir a la noticia": Redirige a la página de la noticia correspondiente. • Click en el botón "Perfil": Redirige a la página de perfil del usuario.
Elementos	<ol style="list-style-type: none"> 1) Cabecera. 2) Botón "Perfil". 3) Lista de comentarios. 4) Enlace "Ir a la noticia".

Tabla 55. Diseño de Interfaz Mis Comentarios

Sobre Nosotros	
Descripción	Interfaz donde se muestra información sobre OVERNEWS y donde los usuarios se pueden poner en contacto vía email.
Activación	La página se activa cuando el usuario selecciona la opción "Sobre Nosotros" en el menú de navegación.
Boceto	 <p style="text-align: center;"><i>Figura 39. Boceto "Sobre Nosotros"</i></p>
Eventos	<ul style="list-style-type: none"> • Click en "Enviar": Envía el formulario de contacto al email de OVERNEWS. • Click en el botón "Perfil": Redirige a la página de perfil del usuario.
Elementos	<ol style="list-style-type: none"> 1) Cabecera. 2) Botón "Perfil". 3) Sección de información sobre OVERNEWS. 4) Campo asunto del formulario de contacto. 5) Campo contenido del formulario de contacto. 6) Botón "Enviar"

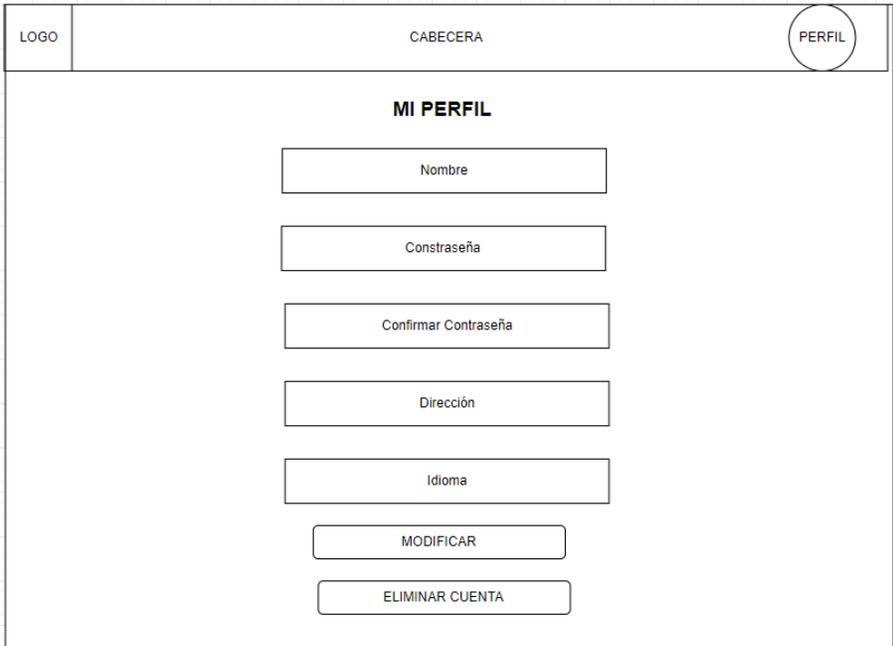
Tabla 56. Diseño de Interfaz Sobre Nosotros

Mostrar Noticia	
Descripción	Página donde se muestra el contenido completo de una noticia específica, incluyendo, la fuente, la categoría y los comentarios. Además, los usuarios pueden añadir, editar o eliminar sus comentarios.
Activación	La página se activa cuando el usuario selecciona una noticia específica desde cualquier sección de OVERNEWS.
Boceto	
Eventos	<ul style="list-style-type: none"> • Click en "Ir a la noticia": Redirige a la página de la noticia original. • Click en "Añadir": Añade un nuevo comentario a la noticia. • Click en "Editar": Permite editar un comentario propio. • Click en "Eliminar": Permite eliminar un comentario propio. • Click en "Volver": Permite volver a la página anterior.
Elementos	<ol style="list-style-type: none"> 1) Cabecera. 2) Botón "Perfil". 3) Imagen de la noticia. 4) Contenido completo de la noticia.

Figura 40. Boceto "Mostrar Noticia"

	<ol style="list-style-type: none"> 5) Fuente y categoría. 6) Botón "Ir a la noticia". 7) Comentarios de noticia. 8) Campo para añadir el comentario. 9) Botón "Añadir" el comentario. 10) Botón "Editar" el comentario. 11) Botón "Eliminar" el comentario. 12) Botón "Volver".
--	---

Tabla 57. Diseño de Interfaz "Mostrar Noticia "

Gestionar Perfil	
Descripción	Interfaz donde los usuarios pueden editar diferentes campos de su perfil (nombre, contraseña, dirección e idioma) y también pueden eliminar su cuenta si lo desean.
Activación	La página se activa cuando el usuario autenticado selecciona la opción "Perfil" en el menú de navegación.
Boceto	 <p style="text-align: center;"><i>Figura 41. Boceto "Gestionar Perfil"</i></p>
Eventos	<ul style="list-style-type: none"> • Click en "Modificar": Guarda los cambios realizados en el perfil del usuario. • Click en "Eliminar Cuenta": Elimina la cuenta del usuario.
Elementos Clave	<ol style="list-style-type: none"> 1) Cabecera. 2) Campo de nombre 3) Campo de contraseña 4) Campo de repetir contraseña. 5) Campo de dirección. 6) Campo de idioma. 7) Botón " Modificar ".

8) Botón " Eliminar Cuenta ".

Tabla 58. Diseño de Interfaz "Gestionar Perfil"

Panel del Administrador	
Descripción	Interfaz que permite al administrador gestionar de manera eficiente las noticias, comentarios, usuarios y sugerencias recibidas de los usuarios. El administrador cuenta con opciones para eliminar cualquier noticia, comentario o usuario que considere inapropiado o innecesario. Además, tiene la capacidad de aceptar o rechazar sugerencias enviadas por los usuarios,
Activación	La página se activa cuando un usuario con permisos de administrador selecciona la opción "Panel del Administrador" en el menú de navegación.
Boceto	
Eventos	<ul style="list-style-type: none"> • Click en “Eliminar” (Noticias): Elimina la noticia seleccionada. • Click en “Eliminar” (Comentarios): Elimina el comentario

Figura 42. Boceto "Panel del Administrador"

	<p>seleccionado.</p> <ul style="list-style-type: none"> • Click en “Eliminar” (Usuarios): Elimina el usuario seleccionado. • Click en "Eliminar” (Sugerencias): Elimina la sugerencia seleccionada. • Click en “Aceptar” (Sugerencias): Acepta la sugerencia seleccionada.
<p>Elementos</p>	<ol style="list-style-type: none"> 1) Cabecera. 2) Botón “Perfil”. 3) Tabla de Noticias. 4) Tabla de Comentarios. 5) Tabla de Usuarios. 6) Tabla de Sugerencias: 7) Botón “Eliminar” para cada una de las tablas. 8) Botón “Aceptar” para la Tabla de Sugerencias.

Tabla 59. Diseño Interfaz "Panel del Administrador"

Capítulo 5

Implementación

En este capítulo se detallará el proceso de implementación del proyecto **OVERNEWS**, describiendo la estructura del proyecto (Sección 5.1) y las cuestiones de implementación más relevantes (Sección 5.2).

5.1 Estructura del proyecto

En esta sección, se presentará la estructura general del proyecto, desglosada en tres componentes principales: *Backend* (Sección 5.1.1), *Frontend* (Sección 5.1.2) y API de Traducción (Sección 5.1.3). Esto se debe a que estos tres componentes pueden funcionar de forma totalmente independiente. En cada una de estas se explicará en detalle la estructura y el propósito de cada componente.

5.1.1 *Backend*

Como ya mencioné anteriormente en el apartado de entorno y tecnologías utilizadas (Sección 1.4), en el *backend* he decidido usar Python como lenguaje de programación. Es por esto por lo que he utilizado *FastAPI* como *framework*.

Este componente que actúa como una API REST tiene la siguiente estructura:

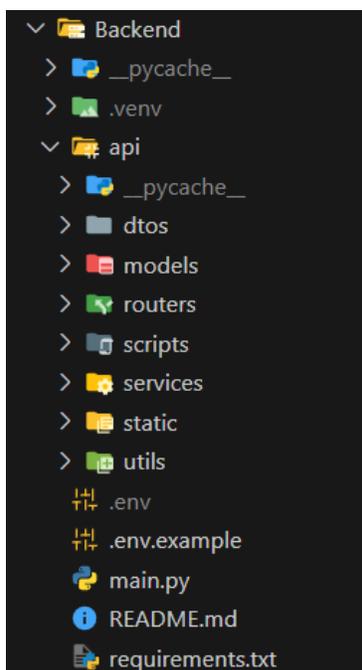


Figura 43. Estructura del Backend

Como se puede observar en la imagen anterior, el *backend* contiene 3 directorios:

1. `/_pycache_`: Directorio generado automáticamente por Python que almacena archivos compilados.
2. `/.venv`: Entorno virtual de Python que aísla las dependencias del proyecto, asegurando que no haya conflictos con otras aplicaciones o proyectos en el mismo sistema.
3. `/api`: es el directorio principal del *backend*.

El directorio `api` es el núcleo del *backend* y contiene los componentes principales que gestionan la lógica del negocio, las rutas de la API, los modelos de datos y las utilidades. A continuación, se detallan los subdirectorios de `/api` y sus funciones:

5.1.1.1 DTOs

Este subdirectorio contiene las clases DTO (*Data Transfer Objects*), cuya función es definir las diferentes estructuras de datos que se envían y reciben a través de los *endpoints*. Así los datos se validan y transforman adecuadamente.

A continuación, se muestran los diferentes ficheros que contiene el subdirectorio `dtos`.

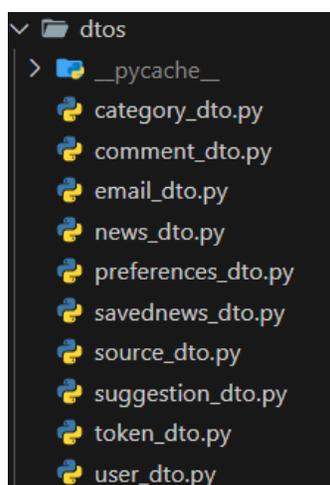


Figura 44. Contenido subdirectorio `/dtos`

Usando los DTOs se consigue encapsular los datos que se transfieren entre las distintas capas, asegurando que solo la información necesaria y permitida sea accesible. De esta forma, se reduce el riesgo de exponer datos sensibles.

5.1.1.2 Models

El segundo subdirectorio los modelos que representan las entidades de la base de datos. Estos modelos definen la estructura de las tablas y las relaciones entre ellas, utilizando *Peewee* como ORM (*Object-Relational Mapping*) para interactuar con la base de datos.

Por lo tanto, existirá una *model* por cada tabla en la base de datos. En la siguiente imagen, se muestran los diferentes modelos que conforman el sistema.

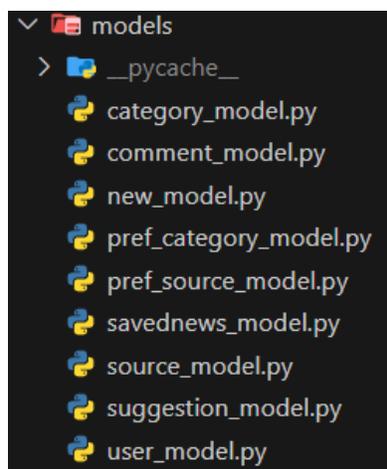


Figura 45. Contenido subdirectorio /models

5.1.1.3 Routers

Este subdirectorio alberga los distintos enrutadores (*routers*) o controladores que gestionan las solicitudes HTTP y que van a dirigir el tráfico a los servicios apropiados. Cada enrutador está encargado de un conjunto específico de rutas (la ruta es la parte del *path* de la url asociada al recurso solicitado por vía HTTP y que constituye la forma concreta de solicitar un servicio determinado al servidor bajo el paradigma de una API Rest). También son conocidos como *endpoints*, que están relacionadas con una funcionalidad particular del sistema.

En la siguiente imagen se muestran los diferentes ficheros que se encuentran dentro de este subdirectorio:

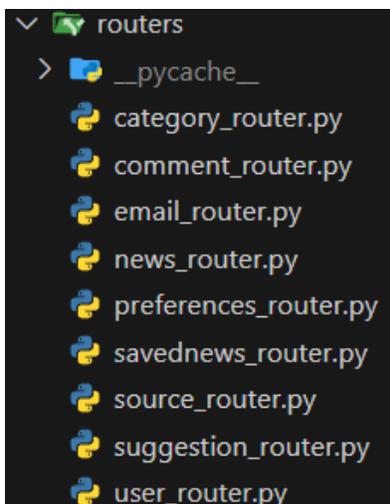


Figura 46. Contenido subdirectorio /routers

En la siguiente tabla, se enumerarán todos los *endpoints* que contiene la aplicación, proporcionando una visión detallada de las rutas disponibles y sus respectivas funcionalidades, es decir, la definición de la API Rest soportada por OVERNEWS.

Router	Ruta	Método HTTP	Descripción
Category	/api/get_category/{category_id}	GET	Obtiene información detallada sobre una categoría específica por su id.
	/api/add_category	POST	Añade una nueva categoría
	/api/delete_category/{category_id}	DELETE	Elimina una categoría específica por su id.
Comment	/api/add_comment	POST	Añade un nuevo comentario
	/api/delete_comment/{comment_id}	DELETE	Elimina un comentario específico por su id.
	/api/update_comment/{comment_id}	PUT	Actualiza un comentario específico por su id.
	/api/user_comments	GET	Obtiene todos los comentarios del usuario actual.
	/api/news_comments/{news_id}	GET	Obtiene los comentarios de una noticia específica por su id.
Email	/api/contact_us	POST	Envía el email que el usuario desea al correo de soporte de OVERNEWS.
	/api/send_password_email	POST	Envía un email para que un usuario pueda recuperar su contraseña
News	/api/get_news	POST	Obtiene una lista de noticias según los parámetros que recibe.
	/api/get_new/	POST	Obtiene una noticia específica por su id y el idioma del usuario actual.

Router	Ruta	Método HTTP	Descripción
News	/api/get_statistics	GET	Obtiene estadísticas de las noticias.
	/api/new/{new_id}	DELETE	Elimina una noticia específica por su id.
	/api/get_all_news	GET	Obtiene todas las noticias.
Preferences	/api/get_preferences	GET	Obtiene las preferencias del usuario actual
	/api/update_preferences	PUT	Actualiza las preferencias del usuario actual.
SavedNews	/api/save_news/{news_id}	POST	Guarda una noticia específica por su id como favorita.
	/api/delete_saved_news/{news_id}	DELETE	Elimina una noticia por su id guardada como favorita.
	/api/get_saved_news	GET	Obtiene las noticias guardadas del usuario actual.
	/api/get_saved_news_ids	POST	Obtiene el id de todas las noticias guardadas del usuario actual.
Source	/api/get_source/{source_id}	GET	Obtiene una fuente específica por su id.
	/api/add_source	POST	Añade una nueva fuente.
	/api/delete_source/{source_id}	DELETE	Elimina una fuente específica por su id.
Suggestion	/api/add_suggestion	POST	Añade una nueva sugerencia.
	/api/delete_suggestion/{suggestion_id}	DELETE	Elimina una sugerencia específica por su id.
	/api/user_suggestions	GET	Obtiene las sugerencias del usuario actual
	/api/update_suggestion_state/{suggestion_id}	PUT	Actualiza el estado de una sugerencia específica por su id.
	/api/get_all_suggestions	GET	Obtiene todas las sugerencias.
User	/api/create_user	POST	Crea un nuevo usuario.
	/api/login	POST	Inicia sesión y obtiene un token de acceso.
	/api/logout	GET	Cierra sesión e invalida el token actual.
	/api/verify-token	GET	Verifica el token actual.
	/api/users	GET	Obtiene una lista de todos los usuarios.
	/api/user	PUT	Actualiza la información del usuario actual.
	/api/user/{user_id}	DELETE	Elimina un usuario específico por su id.
	/api/user	DELETE	Elimina la cuenta del usuario actual.
	/api/activate/{token}	GET	Activa un usuario con el token dado.
/api/change-password	POST	Cambia la contraseña del usuario.	

Tabla 60. Endpoints

5.1.1.4 *Scripts*

Este subfichero contienen todos los *scripts* o ficheros que se van a encargar de obtener las noticias y llevar a cabo la traducción de forma automática. A continuación, se muestra el contenido de este subfichero:

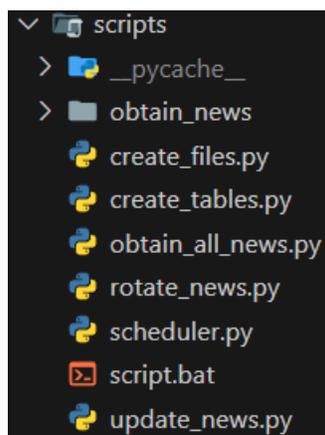


Figura 47. Contenido subdirectorio /scripts

- Subdirectorio `obtain_news`: En este subdirectorio se encuentran los archivos que contienen la lógica de cómo obtener las noticias haciendo *web-scraping*. Realmente estos ficheros no se usan a la hora de obtener las noticias, pero viene bien para hacer pruebas de una fuente y una categoría en concreto. En la siguiente imagen se muestra el contenido de este subdirectorio.

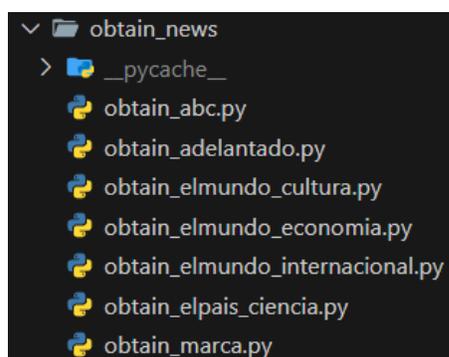


Figura 48. Contenido subdirectorio /obtain_news

- Fichero `create_files.py`: Este *script* se va a encargar de crear los ficheros y carpetas en donde se van almacenar las noticias traducidas de manera temporal.
- Fichero `create_tables.py`: Este *script* se va a utilizar para crear e inicializar las tablas en la base de datos. Es útil durante la configuración inicial del proyecto o cuando se necesita reconstruir la base de datos desde cero.

- Fichero `obtain_all_news.py`: Este archivo es el encargado de obtener todas las noticias de las diferentes fuentes y categorías. Este archivo obtiene las noticias usando la librería *BeautifulSoup*, que permite hacer *web-scraping* de forma eficiente.
- Fichero `rotate_news.py`: Este *script* se va a encargar de que todos los días a las 00:00, roten los archivos temporales en los que se encuentra toda la información de las noticias traducidas. Estos datos se encuentran en el directorio `/api/static`. En la siguiente sección se dará más detalles sobre cómo se almacenan y cómo se rotan los ficheros que almacenan las noticias traducidas.
- Fichero `sheduler.py`: Este fichero es el encargado de automatizar la actualización y traducción de noticias utilizando *APScheduler* para ejecutar tareas en el intervalo de tiempo que se desee. Por lo tanto, este fichero va a actuar de planificador, haciendo que cada hora obtengan las noticias nuevas (ejecutando el fichero `update_news.py`) y se traduzcan en segundo plano. En la Sección 5.2, se darán más matices sobre el funcionamiento de este *script*.
- `Script.bat`: Este archivo `script.bat` está diseñado para ser programado y ejecutado automáticamente utilizando la opción de Programador de Tareas en Windows. Su propósito es que el `script rotate_news.py` se ejecute sin problemas todos los días a la 00:00. A la hora de subirlo a un servidor usando Unix/Linux, se usaría *Crontab* para automatiza la tarea.
- Fichero `update_news.py`: este fichero va a implementar la funcionalidad desarrollada en el fichero `obtain_news.py` y va a almacenar los datos persistentes de las noticias en la base de datos (en español).

5.1.1.5 Services

El subdirectorío `services` es fundamental, ya que contiene casi toda la lógica de negocio de la aplicación. Este se encarga de implementar todas las funcionalidades que requieren todos los endpoints mencionados anteriormente en las Sección 5.1.1.3.

Es por ello, que cada uno de los ficheros que se encuentran en `/services` están orientado a cubrir una funcionalidad en concreto. El contenido de este subdirectorío se puede apreciar en la siguiente imagen:

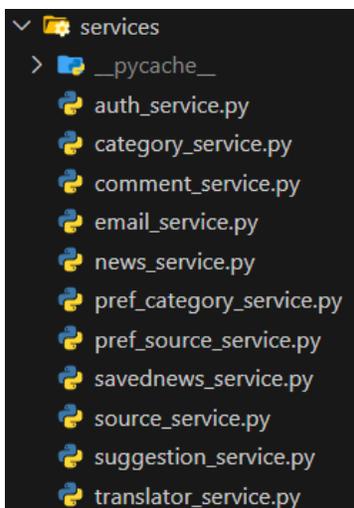


Figura 49. Contenido subdirectorío /services

En la siguiente tabla, se muestran las diferentes funcionalidades que se implementan en cada uno de los ficheros de services.

Archivo	Descripción
auth_service.py	Gestiona la autenticación y autorización de usuarios, incluyendo la verificación de tokens y la encriptación de las contraseñas.
category_service.py	Maneja la lógica relacionada con las categorías de noticias, como la creación, eliminación y consulta de categorías.
comment_service.py	Gestiona las operaciones sobre los comentarios de las noticias, incluyendo la adición, actualización, eliminación y recuperación de comentarios.
email_service.py	Encargado de las funcionalidades de envío de correos electrónicos, como el contacto y la recuperación de contraseñas.
news_service.py	Gestiona el almacenamiento, actualización y eliminación de noticias.
pref_category_service.py	Maneja las preferencias de categorías de noticias para los usuarios, permitiendo la actualización y recuperación de estas preferencias.
pref_source_service.py	Gestiona las preferencias de fuentes de noticias para los usuarios, permitiendo la actualización y recuperación de estas preferencias.
savednews_service.py	Encargado de la lógica para guardar y eliminar noticias favoritas o guardadas por los usuarios.
source_service.py	Maneja la lógica relacionada con las fuentes de noticias, incluyendo la creación, eliminación y consulta de fuentes.
suggestion_service.py	Gestiona las sugerencias de los usuarios, permitiendo la creación, actualización del estado y eliminación de sugerencias.

Archivo	Descripción
translator_service.py	Encargado de la traducción de las noticias a los diferentes idiomas que se requiera.
user_service.py	Gestiona la lógica relacionada con los usuarios, incluyendo la creación, actualización y eliminación de usuarios.

Tabla 61. Descripción de los servicios (services)

5.1.1.6 Static

Este subdirectorio, es muy relevante para OVERNEWS, ya que almacena temporalmente las noticias en archivos JSON organizados según su idioma y día. Así se consigue un acceso eficiente y organizado a las noticias, facilitando su manejo y traducción.

En la siguiente imagen se muestra el contenido del subdirectorio `/static`.



Figura. Contenido subdirectorio `/static`

El directorio contiene 8 carpetas con las noticias completas en los diferentes idiomas y 2 archivos JSON. Cabe resaltar, que las carpetas donde se almacenan las diferentes noticias no se crearán hasta que se lleve a cabo la primera ejecución.

Directorios `news_{idioma}`

Cada uno de estos 8 directorios, almacena las noticias completas en un idioma en completo. Estas carpetas están nombradas siguiendo el formato `news_{idioma}`, donde `{idioma}` representa el código del idioma. A continuación, se muestra el significado de los diferentes códigos de idioma:

Código de Idioma	Idioma
es	Español
en	Inglés
fr	Francés
de	Alemán
it	Italiano
pt	Portugués
ru	Ruso
ar	Árabe

Figura 50. Aclaración de los códigos de idioma

Cada carpeta contiene 7 archivos JSON que almacenan las noticias de los últimos 7 días. Cada día las noticias rotan asegurando que las noticias sean accesibles temporalmente durante una semana.

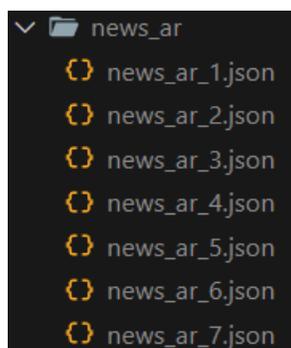


Figura 51. Ejemplo Contenido news_ar

`info.json`

El primer archivo JSON proporciona información la información necesaria para poder inicializar los usuarios, las categorías y las fuentes en la base de datos de forma automática. Además, también almacena las *urls* necesarias para realizar web scraping y obtener nuevas noticias.

`new_news.json`

El segundo archivo JSON almacena temporalmente las noticias recién obtenidas. De esta forma actúa como fichero intermediario mientras las noticias se traducen.

5.1.1.7 Utils

El subdirectorio `utils` contiene utilidades esenciales que facilitan la configuración y gestión de la base de datos, así como otras configuraciones importantes del sistema. Este directorio incluye dos siguientes archivos.

- Fichero `database.py`: Este fichero va a gestionar la conexión y el estado de la base de datos utilizando *Peewee* como ORM.
- Fichero `settings.py`: Se va a encargar de manejar la configuración global del utilizando *Pydantic* y variables de entorno. En este se usa *dotenv* para cargar las configuraciones desde el archivo `.env`, que es dónde se van a almacenar las variables globales.

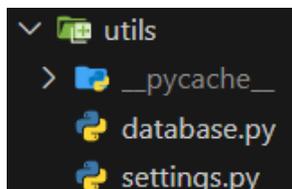


Figura 52. Contenido subdirectorio `/utils`

5.1.1.9 Archivos principales

Además de los diferentes subdirectorios comentados anteriormente, colgando del directorio de `/api` hay archivos que tienen gran relevancia:

- Fichero `.env`: En este se almacenan las variables globales o de entorno utilizadas en el proyecto.
- Fichero `.env.example`: Sirve como plantilla para el archivo `".env"`. Este fichero facilita la configuración inicial del proyecto para nuevos desarrolladores, indicando qué variables de entorno son necesarias.
- Fichero `main.py`: Este es el fichero principal del *backend*. En este se definen las rutas, la conexión a la base de datos y la inicialización del planificador (`sheduler.py`) para que cada hora se obtengan las noticias y se traduzcan correctamente.
- Fichero `README.md`: Este fichero incluye una descripción del proyecto, instrucciones de instalación y de configuración.
- Fichero `requirements.txt`: En este fichero se listan todas las dependencias de Python que deben ser instalados para que la aplicación funcione correctamente.

5.1.2 Frontend

En esta sección se verá cómo se estructura el *frontend*. El *frontend* del proyecto OVERNEWS está diseñado de manera organizada, para facilitar tanto el desarrollo como el

mantenimiento de la aplicación. A continuación, se van a explicar los componentes y archivos principales.

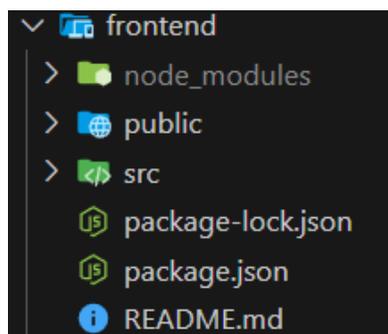


Figura 53. Contenido frontend

En los siguientes puntos se van a describir los archivos y directorios principales.

- `/node_modules`: Directorio donde se almacenan todas las dependencias de *Node.js* necesarios para el proyecto. Este directorio se va a generar directamente cuando instalemos las dependencias usando `npm install`.
- `/public`: EL directorio `public` contiene el `index.html` que es el archivo de entrada HTML sobre el que *React* construye toda la aplicación. Además, en este también van a haber recursos estáticos como imágenes, que, al encontrarse en este directorio, no son procesados por el empaquetador y se sirven tal cual.

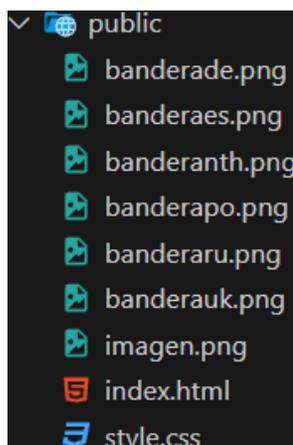


Figura 54. Contenido /public

- `/src`: Este directorio es el más relevante y que contiene el código fuente principal de la aplicación *frontend* desarrollada con *React.js*. A continuación, se entra en detalle sobre el contenido de este directorio:

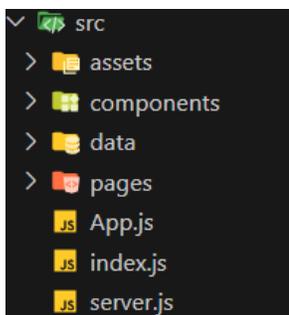


Figura 55. Contenido /src

1. /assets: Subdirectorio donde se almacenan recursos como imágenes y estilos.
2. /components: Subdirectorio donde se encuentran todos los componentes que van a formar parte de las interfaces del usuario. Los diferentes componentes se almacenan en diferentes subdirectorios.

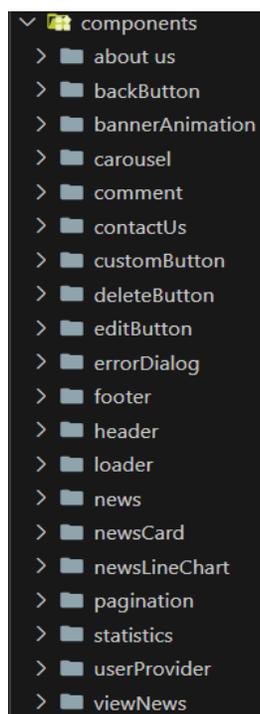


Figura 56. Componentes de React

3. /data: En este directorio se encuentra el archivo `data.json`, que se usa a modo de diccionario ya que en el se encuentra el texto de las interfaces en los diferentes idiomas. De esta forma se facilita el soporte multilingüe de la aplicación OVERNEWS.
4. /pages: Contiene las diferentes páginas o interfaces de la aplicación. Las páginas van a estar compuestas por los diferentes componentes de *React* que se han creado.

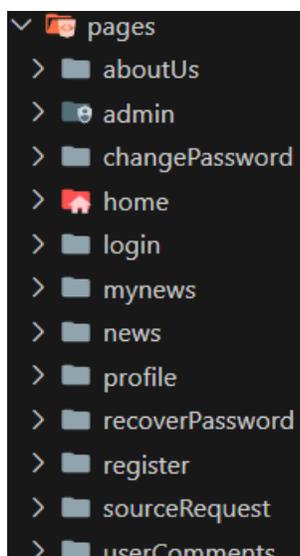


Figura 57. Contenido /pages

5. Fichero `App.js`: Archivo principal de la aplicación *React* donde se declara el componente principal `App`. En este se definen las rutas (enrutamiento) y se declara el `UserProvider` (encargado de la autenticación y la gestión del usuario). En la siguiente sección se verán más detalles.
 6. Fichero `index.js`: Es el punto de entrada de la aplicación donde se renderiza el componente principal (`App`).
 7. Fichero `server.js`: En este archivo se declara cómo va a ser la url del servidor al que se le van a hacer las diferentes peticiones.
- Fichero `package-lock.json`: Este archivo es el que asegura la instalación consistente de dependencias.
 - Fichero `package.json`: Este es el archivo de configuración principal para *Node.js*.
 - Fichero `README.md`: Al igual que en el *backend*, va a existir este archivo a modo de guía rápida del *frontend*.

5.1.3 API Traducción

El último componente en la arquitectura del proyecto **OVERNEWS** es una API de traducción que hemos extraído de un repositorio en GitHub. Dado que el alcance del proyecto no permitía desarrollar una API de traducción desde cero, se decidió usar y modificar una ya existente. La url de la API de traducción original es: <https://github.com/kylegallatin/machine-translation-service>.

La API se despliega en un entorno local y se ocupa de todas las solicitudes de traducción en el caso de que los límites de las dos APIs gratuitas que estamos empleando se excedan.

Esta API de traducción ha sido personalizada para trabajar sin problemas con **OVERNEWS** para que las traducciones se hagan de manera efectiva y fluida.

Aunque esté fuera del alcance del proyecto, voy a explicar brevemente su estructura:

- `/data`: Este directorio contiene los modelos de inteligencia artificial entrenados. En este caso, hay siete modelos que traducen del español a los otros siete idiomas soportados por **OVERNEWS**.
- Fichero `app.py`: Esta es la clase principal de este componente y en esta se declara el enrutador junto con las diferentes rutas a las que se pueden llamar.
- Fichero `config.py`: En este fichero se almacenan las variables de configuración necesarias.
- Fichero `download_models.py`: Este fichero permite que se puedan descargar los diferentes modelos Helsinki-NLP, que son modelos desarrollados por la Universidad de Helsinki.
- Fichero `translate.py`: Fichero en el que se encuentran los servicios de traducción que hacen uso de los diferentes modelos de inteligencia artificial.
- Fichero `package-lock.json`: En este se manejan las versiones exactas de las dependencias instaladas.
- Fichero `README.md`: Documentación del proyecto,
- Fichero `requirements.txt`: Contiene una lista todas las dependencias de Python necesarias para que la API funcione correctamente

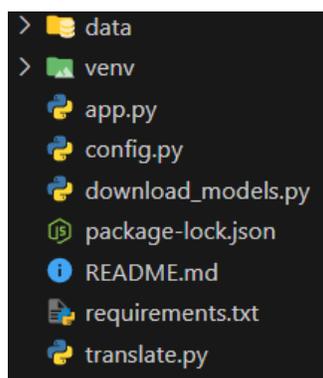


Figura 58. Contenido Api Traducción

5.2 Cuestiones de implementación reseñables

Una vez explicada la estructura del proyecto de OVERNEWS (Sección 5.1), se entra en detalle en las cuestiones de implementación que merecen ser destacadas por su relevancia e impacto en el proyecto. A continuación, se presentan las más significativas.

5.2.1 Manejo de datos persistentes y temporales

Como ya se ha mencionado anteriormente, únicamente se va a almacenar en la base de datos las noticias en español. Además, en relación con las noticias almacenadas en base de datos, es importante resaltar que no se tendrán los campos “descripción” y “cuerpo” de la noticia, ya que, por lo general, son los campos más pesados. Esto se hace para aligerar el espacio que ocupa las noticias en base de datos y que constituye el estado persistente asociado a las noticias gestionadas por OVERNEWS.

Cabe destacar que el objetivo principal de la aplicación es que se tengan noticias **actualizadas** en diferentes idiomas. Es por esto, que, a la hora de desarrollar el proyecto, decidí que el ciclo de vida de las noticias en la aplicación fuese de 7 días. De esta forma, se consigue ahorrar espacio, además de tener una página actualizada sin datos duplicados.

Por lo tanto, la información de las noticias traducidas se almacena en archivos JSON organizados en directorios por idioma y día (Sección 5.1.1.6), y esta información constituye el estado temporal asociado a las noticias gestionadas por OVERNEWS. Un ejemplo del contenido de los archivos JSON mencionados se muestra en la siguiente figura.

```

{
  "News": [
    {
      "id": "53db3c1e98653f6160c27e64ad8fdb64c610ab5a77439a1630b9df1beec90dfe",
      "title": "Tourism of Segovia summons the Dulce Isabel dessert contest",
      "description": "This activity is part of the acts organized by the 550th anniversary of the proclamation of Queen Elizabeth",
      "info": "The proclamation acts of Queen Elizabeth I of Castilla that took place in Segovia 550 years ago will have their swe
    },
    {
      "id": "44d358019c9fb5f9c2e972ac37215b27b0fb2b4708c8cbd52a0b78463bb8a134",
      "title": "The Segovia Museum exhibits 32 awarded works in 'Young Art'",
      "description": "The sample includes works of painting, sculpture, fashion design, graphic or audiovisual design",
      "info": "The Segovia Museum has inaugurated the traveling exhibition 'Art on the Road', in which 32 works by young winners a
    },
    {
      "id": "246f519c367c04e0069527b2d1ceeb436ff9df498a453aba89e330a58f2dc43b",
      "title": "Civil Guard and Firefighters share fire training",
      "description": "The command hosts an information days against first interventions in incidents in which propulsion vehicles",
      "info": "within the framework of the training actions that are developed in the Traffic subsector of the Civil Guard of Segov
    },
    {
      "id": "2ebf0550ecb9c330b2a1a2bc6ddd8f9c6692bc3468eae30e2c0b4b3992b5485c",
      "title": "Detained a person for breaching a restraining order",
      "description": "The long feast weekend has resulted in more than thirty complaints of the Local Police, 17 for urinating on",
      "info": "The Local Police of Segovia has reported the arrest this weekend of a person for brokenness of a removal order.The
    },
    {
      "id": "bfdf354a984769b84d0d3ab2c25fe7d58682f10e3dd5b53bdb25962ba382b250",
      "title": "The party challenge for drinking distributors",
      "description": "The companies in the sector trigger their activity these days with beer and white wine as main claims .",
      "info": "Alvaro Gil has been working on the distribution of drinks 25 years.He has known from within the fairs and festivit
    }
  ]
}

```

Figura 59. Contenido ejemplo de news_en_1

Como se puede apreciar en la imagen, se almacena el `id` que la noticia en español tiene en la base de datos. De esta forma cuando se solicite una noticia, primero se va a buscar en base de datos, y posteriormente se va a buscar en los archivos JSON según el día y el idioma.

Buscar por idioma es una tarea sencilla ya que es ir a la carpeta `news_{idioma}` del directorio `/static`, pero lo difícil es saber en qué archivo JSON se encuentra la noticia que se solicita.

En este punto, nació la necesidad de incluir un campo en la tabla de `news` que hiciese referencia al número del fichero JSON en el que se encuentra la noticia. Esta referencia se almacena en el campo `number`.

Para conseguir que el ciclo de vida de las noticias fuese de 7 días, decidí implementar algo muy similar a lo que implementan en los sistemas operativos con la rotación de los archivos de los `logs`. Estos cada día a las 00:00h, para que los logs estén actualizados, sobrescriben los archivos unos sobre otros para que rote la información en función del tiempo.

En la siguiente imagen se puede observar cómo se implementa en la aplicación para las noticias en español que se encuentran en el directorio `news_es`.

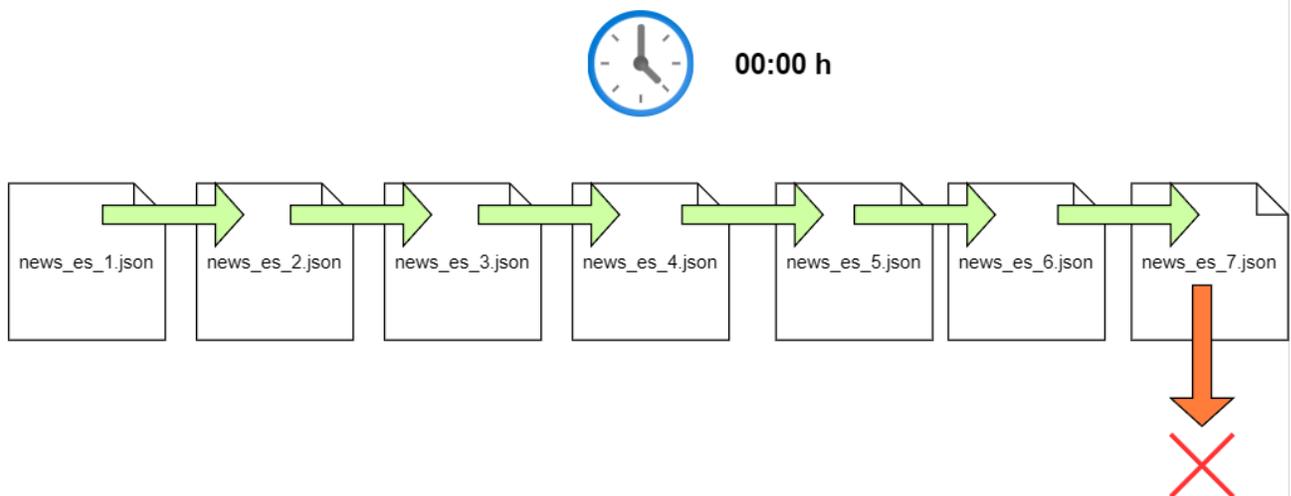


Figura 60. Rotación de archivos JSON

Esto se ejecuta en todas las carpetas `news_{idiomas}` gracias a que todos los días a las 00:00h se ejecuta el `script.bat`, que a su vez ejecuta el fichero `rotate_news.py`.

```
@echo off
cd C:\Users\jcall\Desktop\TFG\Backend\api
set PYTHONPATH=%PYTHONPATH%;C:\Users\jcall\OneDrive\Escritorio\TFG\Backend
python -m scripts.rotate_news You, 6 months ago • FIX: script.bat
```

Figura 61. Script.bat

Por lo tanto, el fichero `rotate_news.py` va a ser el encargado de rotar los ficheros dentro de las diferentes carpetas, y de incrementar el valor `number` en la base de datos para que no

se pierda la referencia (si el número es mayor que 7, significa que ya no existe en los archivos temporales).

```

from api.services.news_service import increment_number_for_news
import os
import json
import shutil

def rotate_news_files():
    folder_path = os.path.join(os.path.dirname(__file__), '..', 'static', 'news')
    elements = os.listdir(folder_path)
    folders = [element for element in elements if os.path.isdir(os.path.join(folder_path, element))]
    for folder in folders:
        for i in range(6, -1, -1):
            file_name = f"{folder}_{i}.json"
            new_file_name = f"{folder}_{i + 1}.json"

            file_path = os.path.join(os.path.dirname(__file__), '..', 'static', 'news', folder, file_name)
            new_file_path = os.path.join(os.path.dirname(__file__), '..', 'static', 'news', folder, new_file_name)

            if os.path.exists(file_path):
                shutil.move(file_path, new_file_path)
            if i == 0:
                with open(new_file_path, "w") as new_file:
                    json.dump({"News": []}, new_file)

rotate_news_files()
increment_number_for_news()

```

Figura 62. Fichero "rotate_news.py"

```

def increment_number_for_news():
    try:
        query = New.update(number=New.number + 1).where(New.number < 8)
        query.execute()
    except Exception as e:
        raise HTTPException(status_code=500, detail=str(e))

```

Figura 63. Función increment_number_for_news()

De esa forma se consigue rotar las noticias y actualizar la referencia de manera correcta, de modo que, no se pierde la relación entre los datos persistentes y datos temporales.

5.2.2 Obtención y traducción de las noticias

Obtención de las noticias

Como ya se ha comentado con anterioridad a lo largo de la Sección 5.1, el fichero que se encarga de automatizar el proceso que permite obtener y traducir las noticias cada hora, es `scheduler.py`. Este va a actuar como planificador y se va a ejecutar directamente desde el `main.py`, que es el archivo principal del *backend*.

```
if __name__ == "__main__":
    print("Antes de llamar a start_scheduler()")
    start_scheduler()
    print("Después de llamar a start_scheduler()")
    uvicorn.run("main:app",port=8000)
```

Figura 64. Ejecución del planificador en "main.py"

Desde el main.py se invoca a la función start_sheduler(), que se va a encargar de iniciar el planificador.

```
def start_scheduler():
    print("Iniciando planificador")
    scheduler = BackgroundScheduler()
    job_function()
    scheduler.add_job(job_function, 'interval', hours=1)
    scheduler.start()
```

Figura 65. Función start_sheduler()

En esta función, se indica que se debe de ejecutar la tarea job_function() cada hora. Esta función va a ser la encargada de ejecutar el fichero update_news.py, que obtiene las noticias y las almacena tanto en base de datos, como en el fichero new_news.json. Este fichero va a servir para saber las noticias que se tiene que traducir.

En la siguiente imagen se observa un esquema de la secuencia de este proceso.

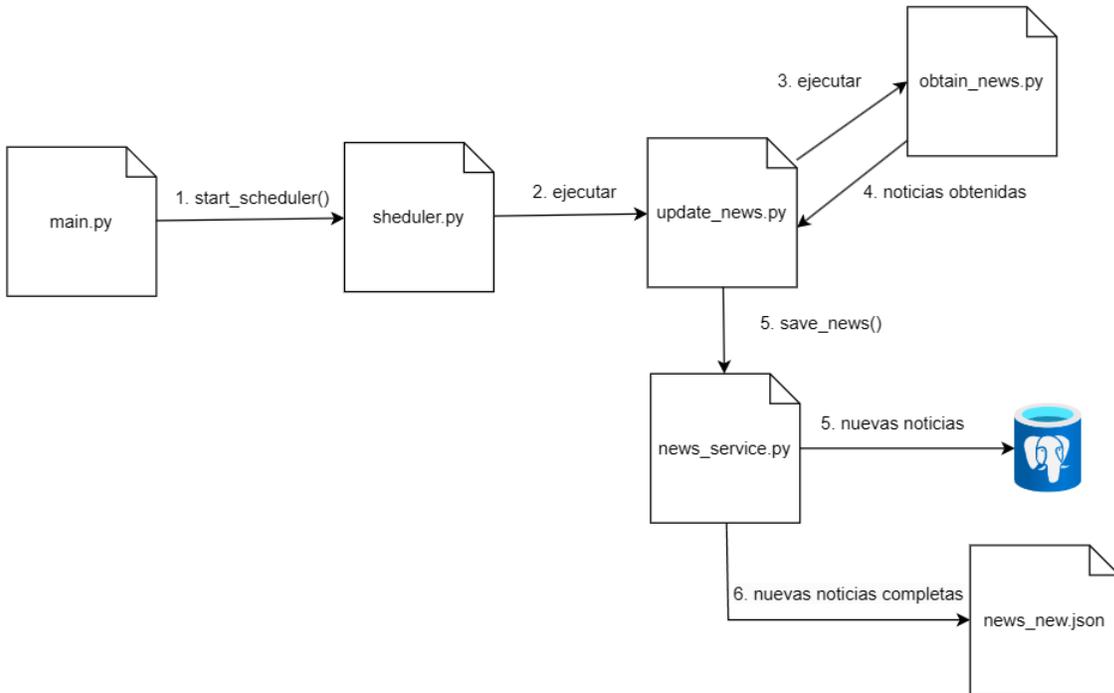


Figura 66. Secuencia obtención de noticias

Traducción de las noticias

Una vez obtenidas las noticias, la función `job_function()` va a crear un hilo de ejecución por cada idioma que esté disponible en la aplicación, para traducir las noticias (llamando a la función `run_additional_process()`).

```
def job_function():
    print("Ejecutando job_function")
    create_folder_and_files("es")
    result = subprocess.run(["python", "api/scripts/update_news.py"], capture_output=True, text=True)
    print(result.stdout)
    print(result.stderr)
    languages = get_languages()
    if "es" in languages:
        languages.remove("es")
    for language in languages:
        create_folder_and_files(language)
        threading.Thread(target=run_additional_process, args=(language,)).start()
```

Figura 67. Función `job_fuction()`

Para entender mejor este proceso, se debe de entender lo que hace cada una de las siguientes funciones:

- `create_folder_and_files()`: Este método tiene como objetivo crear la estructura de directorios y archivos específica para almacenar noticias traducidas en un idioma dado, en caso de que no exista.
- `get_languages()`: Función encargada de obtener los idiomas a los que tiene que traducir las noticias. Para obtener los idiomas, se comprueba qué idiomas han seleccionado los diferentes usuarios para que sea más eficiente la aplicación.
- `run_additional_process()`: Método que va a traducir a llamar a la función `transalate_text` de `translate_service.py`, y que va a almacenar las noticias traducidas en los archivos `news_{idioma}_1` (ya que se almacena en el día de hoy).

Lo último que me gustaría remarcar acerca de la traducción de noticias, es que el orden de las llamadas a las APIs de traducción es el siguiente:

1. API de *Google Trasnlate* gratuita.
2. API de *MTranslate* gratuita.
3. API de traducción en local.

Realizando pruebas, me he dado cuenta de que realmente con las dos primeras APIs, ya se traducen correctamente las noticias, pero por si acaso, he tomado la decisión de incluir una API de traducción en local para prevenir futuras sobrecargas.

5.2.3 Gestión de Autenticación y Autorización

Es importante destacar que, para la autenticación de usuarios, se utiliza JWT, que se genera encriptando el nombre de usuario (apodo) utilizando el algoritmo HS256 y una clave secreta.

De esta forma, con este *token* se puede garantizar que solo los usuarios autenticados pueden acceder a ciertas funcionalidades de la aplicación.

Este token se va a almacenar en el `LocalStorage` del navegador, permitiendo mantener la sesión del usuario activa entre las diferentes visitas a la aplicación.

Para saber si un usuario está autenticado o no, se obtiene el *token* desde el `LocalStorage` y se decodifica. Si al decodificar el *token*, no se encuentra ningún usuario con el apodo especificado, se considera que el *token* es inválido.

De esta forma, es como si se tuviese *un Authorization Server* de forma local:

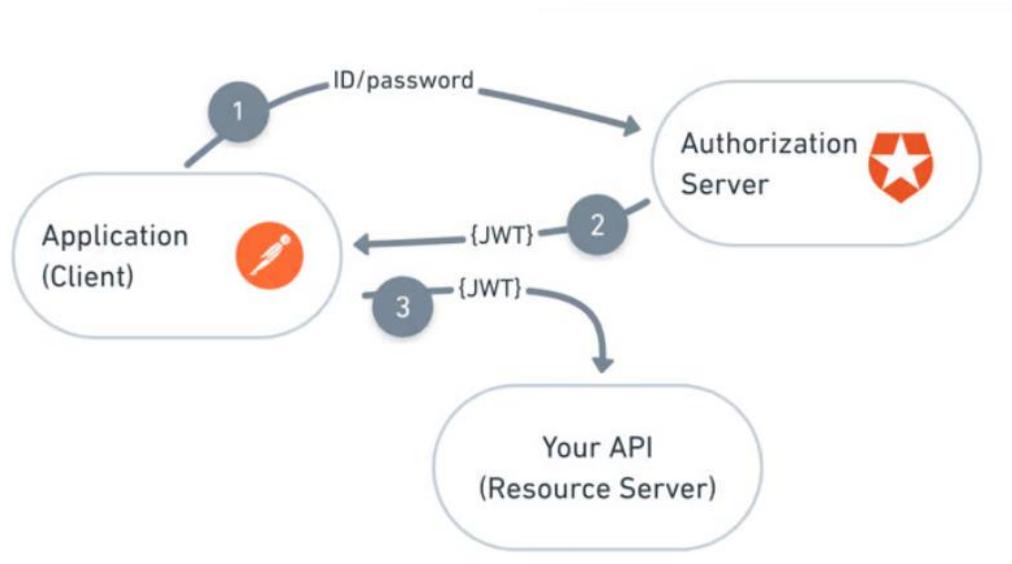


Figura 68. Secuencia JWT

Capítulo 6

Pruebas

En este capítulo, se van a realizar las diferentes pruebas o test que verifican que el software desarrollado función de manera correcta. Para comprobar el correcto funcionamiento del sistema se van a llevar a cabo pruebas de caja blanca (Sección 6.1) y pruebas de caja negra (Sección 6.2).

6.1 Pruebas de caja blanca

Las pruebas de caja blanca se centran en examinar el código del programa. En esta versión de la aplicación no se han llevado a cabo tests de los diferentes servicios, pero, aun así, se ha llevado a cabo un análisis manual del código para comprobar su correcto funcionamiento.

A continuación, se describen las pruebas realizadas:

1. Cobertura de Código: Revisión manual del código, ejecutando todas las funcionalidades de la aplicación.
2. Prueba de Conexión con la Base de Datos: Establecer la conexión con la base de datos y verificar que no se producen errores inesperados
3. Prueba de Inserción de Datos. Insertar datos y verificar su correcto almacenamiento.
4. Prueba de Eliminación de Datos: Realizar eliminaciones desde la aplicación y verificar que los datos se eliminan correctamente.
5. Prueba de Actualización de Datos: Comprobar que actualizando los datos desde la aplicación se realizan correctamente los cambios en la base de datos.
6. Pruebas de Seguridad de las APIs: Comprobar que las APIs solo reciben y devuelven los datos que deben.
7. Pruebas de Autenticación: Comprobar que tanto el registro como el login funcionan correctamente.
8. Pruebas de Manejo de Errores: Comprobar que los errores de la aplicación se muestren de una manera correcta, mostrando una alerta del error al usuario.

6.2 Pruebas de caja negra

Las pruebas de caja negra son una de las técnicas más comunes para verificar un determinado software, donde se evalúan las funcionalidades de una aplicación, sin tener en cuenta su estructura interna. De esta forma, este tipo de pruebas se centran únicamente en las entradas y salidas del software de acuerdo con sus especificaciones y requisitos. [18]

En las siguientes tablas, se muestran las diferentes pruebas de caja negra que se han llevado a cabo.

PCN-01: Registrar un nuevo usuario	
Propósito	Crear una nueva cuenta de usuario
Prerrequisitos	Ninguno
Datos de entrada	Email, nombre de usuario, contraseña, nombre completo
Resultado esperado	Mensaje de registro exitoso (200)
Resultado obtenido	Correcto

Tabla 62. PCN-01: Registrar un nuevo usuario

PCN-02: Iniciar sesión	
Propósito	Crear una nueva cuenta de usuario
Prerrequisitos	El usuario debe estar registrado en el sistema
Datos de entrada	Email, nombre de usuario, contraseña, nombre completo, idioma, dirección
Resultado esperado	Token de autenticación (200)
Resultado obtenido	Correcto

Tabla 63. PCN-02: Iniciar sesión

PCN-03: Verificar el token actual	
Propósito	Verificar la validez del token actual
Prerrequisitos	El usuario debe estar registrado y con sesión iniciada
Datos de entrada	Cabecera de autorización.
Resultado esperado	Información del usuario (200)
Resultado obtenido	Correcto

Tabla 64. PCN-03: Verificar el token actual

PCN-04: Obtener una lista de todos los usuarios	
Propósito	Obtener una lista de todos los usuarios registrados
Prerrequisitos	El usuario debe tener privilegios de administrador.
Datos de entrada	Cabecera de autorización.
Resultado esperado	Lista de usuarios (200)
Resultado obtenido	Correcto

Tabla 65. PCN-04: Obtener una lista de todos los usuarios

PCN-05: Actualizar la información del usuario	
Propósito	Actualizar la información del usuario actual
Prerrequisitos	El usuario debe estar registrado y con sesión iniciada.
Datos de entrada	Información actualizada del usuario y la cabecera de autorización
Resultado esperado	Mensaje de actualización exitosa (200)
Resultado obtenido	Correcto

Tabla 66. PCN-05: Actualizar la información del usuario

PCN-06: Eliminar un usuario por su id	
Propósito	Eliminar un usuario específico por su id
Prerrequisitos	El usuario debe tener privilegios de administrador.
Datos de entrada	Id del usuario, cabecera de autorización
Resultado esperado	Mensaje de eliminación exitosa (200)
Resultado obtenido	Correcto

Tabla 67. PCN-06: Eliminar un usuario por su id

PCN-07: Eliminar la cuenta del usuario actual	
Propósito	Eliminar la cuenta del usuario actual
Prerrequisitos	El usuario debe estar registrado y con sesión iniciada.
Datos de entrada	Cabecera de autorización
Resultado esperado	Mensaje de eliminación exitosa (200)
Resultado obtenido	Correcto

Tabla 68. PCN-07: Eliminar la cuenta del usuario actual

PCN-08: Activar un usuario con el token dado	
Propósito	Activar una cuenta de usuario utilizando un token
Prerrequisitos	El usuario debe haberse registrado y recibir el token de activación.
Datos de entrada	Token de activación
Resultado esperado	Mensaje de activación exitosa (200)
Resultado obtenido	Correcto

Tabla 69. PCN-08: Activar un usuario con el token dado

PCN-09: Cambiar la contraseña del usuario	
Propósito	Cambiar la contraseña del usuario actual
Prerrequisitos	El usuario debe estar registrado y con sesión iniciada.
Datos de entrada	Contraseña nueva, confirmación de la contraseña y la cabecera de autorización
Resultado esperado	Mensaje de cambio exitoso (200)
Resultado obtenido	Correcto

Tabla 70. PCN-09: Cambiar la contraseña del usuario

PCN-10: Recuperar contraseña	
Propósito	Enviar un email para recuperar la contraseña
Prerrequisitos	El usuario debe estar registrado y proporcionar su email.
Datos de entrada	Email del usuario
Resultado esperado	Mensaje de correo enviado (200)
Resultado obtenido	Correcto

Tabla 71. PCN-10: Recuperar contraseña

PCN-11: Actualizar preferencias de usuario	
Propósito	Actualizar las preferencias de noticias del usuario
Prerrequisitos	El usuario debe estar registrado y con sesión iniciada.
Datos de entrada	Lista de categorías y fuentes preferidas y la cabecera de autorización
Resultado esperado	Mensaje de actualización exitosa (200)
Resultado obtenido	Correcto

Tabla 72. PCN-11: Actualizar preferencias de usuario

PCN-12: Añadir un nuevo comentario	
Propósito	Añadir un nuevo comentario a una noticia
Prerrequisitos	El usuario con la sesión iniciada tiene que añadir un comentario a una noticia.
Datos de entrada	Id de la noticia, contenido del comentario y la cabecera de autorización
Resultado esperado	Mensaje de creación exitosa (200)
Resultado obtenido	Correcto

Tabla 73. PCN-12: Añadir un nuevo comentario

PCN-13: Eliminar un comentario	
Propósito	Eliminar un comentario de una noticia
Prerrequisitos	El usuario con la sesión iniciada debe haber escrito el comentario que desea eliminar.
Datos de entrada	Id del comentario y cabecera de autorización
Resultado esperado	Mensaje de éxito (200)
Resultado obtenido	Correcto

Tabla 74. PCN-13: Eliminar un comentario

PCN-14: Actualizar un comentario	
Propósito	Actualizar un comentario específico por su id
Prerrequisitos	El usuario con la sesión iniciada tiene que actualizar un comentario que haya escrito él.
Datos de entrada	Id del comentario, contenido actualizado y la cabecera de autorización
Resultado esperado	Mensaje de actualización exitosa (200)
Resultado obtenido	Correcto

Tabla 75. PCN-14: Actualizar un comentario

PCN-15: Ver comentarios de una noticia	
Propósito	Obtener los comentarios de una noticia específica
Prerrequisitos	La noticia debe existir y tener comentarios asociados.
Datos de entrada	Id de la noticia, cabecera de autorización
Resultado esperado	Lista de comentarios (200)
Resultado obtenido	Correcto

Tabla 76. PCN-15: Ver comentarios de una noticia

PCN-16: Obtener todos los comentarios del usuario actual	
Propósito	Obtener todos los comentarios realizados por el usuario actual
Prerrequisitos	El usuario debe estar registrado y con sesión iniciada.
Datos de entrada	Cabecera de autorización
Resultado esperado	Lista de comentarios (200)
Resultado obtenido	Correcto

Tabla 77. PCN-16: Obtener todos los comentarios del usuario actual

PCN-17: Obtener una categoría	
Propósito	Obtener información detallada sobre una categoría específica por su id
Prerrequisitos	Ninguno
Datos de entrada	Id de la categoría
Resultado esperado	Detalles de la categoría (200)
Resultado obtenido	Correcto

Tabla 78. PCN-17: Obtener una categoría

PCN-18: Añadir una nueva categoría	
Propósito	Añadir una nueva categoría
Prerrequisitos	El usuario debe tener privilegios de administrador.
Datos de entrada	Nombre de la categoría
Resultado esperado	Mensaje de creación exitosa (200)
Resultado obtenido	Correcto

Tabla 79. PCN-18: Añadir una nueva categoría

PCN-19: Eliminar una categoría	
Propósito	Eliminar una categoría por su id
Prerrequisitos	El usuario debe tener privilegios de administrador.
Datos de entrada	Id de la categoría
Resultado esperado	Mensaje de eliminación exitosa (200)
Resultado obtenido	Correcto

Tabla 80. PCN-19: Eliminar una categoría

PCN-20: Obtener una fuente	
Propósito	Obtener información detallada sobre una fuente específica por su id
Prerrequisitos	Ninguno
Datos de entrada	Id de la fuente
Resultado esperado	Detalles de la fuente (200)
Resultado obtenido	Correcto

Tabla 81. PCN-20: Obtener una fuente

PCN-21: Añadir una nueva fuente	
Propósito	Añadir una nueva fuente
Prerrequisitos	El usuario debe tener privilegios de administrador.
Datos de entrada	Nombre de la fuente
Resultado esperado	Mensaje de creación exitosa (200)
Resultado obtenido	Correcto

Tabla 82. PCN-21: Añadir una nueva fuente

PCN-22: Eliminar una fuente	
Propósito	Eliminar una fuente por su id
Prerrequisitos	El usuario debe tener privilegios de administrador.
Datos de entrada	Id de la fuente
Resultado esperado	Mensaje de eliminación exitosa (200)
Resultado obtenido	Correcto

Tabla 83. PCN-22: Eliminar una fuente

PCN-23: Contactar con el soporte	
Propósito	Enviar un email al soporte de OverNews
Prerrequisitos	El usuario debe estar registrado y con sesión iniciada.
Datos de entrada	Mensaje de contacto y la cabecera de autorización
Resultado esperado	Mensaje de envío exitoso (200)
Resultado obtenido	Correcto

Tabla 84. PCN-23: Contactar con el soporte

PCN-24: Obtener todas las noticias	
Propósito	Obtener una lista de todas las noticias
Prerrequisitos	El usuario debe estar registrado y con sesión iniciada.
Datos de entrada	Cabecera de autorización
Resultado esperado	Lista de noticias (200)
Resultado obtenido	Correcto

Tabla 85. PCN-24: Obtener todas las noticias

PCN-25 Obtener una noticia multilingüe en específico	
Propósito	Obtener información completa de una noticia multilingüe por su id.
Prerrequisitos	El usuario debe estar registrado y con sesión iniciada.
Datos de entrada	Id de la noticia y la cabecera de autorización
Resultado esperado	Información completa de la noticia (200)
Resultado obtenido	Correcto

Tabla 86. PCN-25 Obtener una noticia multilingüe en específico

PCN-26 Obtener estadísticas de las noticias	
Propósito	Obtener estadísticas de las noticias
Prerrequisitos	El usuario debe tener privilegios de administrador.
Datos de entrada	Cabecera de autorización
Resultado esperado	Estadísticas de las noticias (200)
Resultado obtenido	Correcto

Tabla 87. PCN-26 Obtener estadísticas de las noticias

PCN-27: Guardar una noticia como favorita	
Propósito	Guardar una noticia específica como favorita
Prerrequisitos	El usuario con la sesión iniciada tiene que guardar una noticia.
Datos de entrada	Id de la noticia y la cabecera de autorización
Resultado esperado	Mensaje de éxito (200)
Resultado obtenido	Correcto

Tabla 88. PCN-27: Guardar una noticia como favorita

PCN-28: Eliminar una noticia guardada como favorita	
Propósito	Eliminar una noticia específica como favorita
Prerrequisitos	El usuario con la sesión iniciada tiene que eliminar una noticia guardada.
Datos de entrada	Id de la noticia y la cabecera de autorización
Resultado esperado	Mensaje de éxito (200)
Resultado obtenido	Correcto

Tabla 89. PCN-28: Eliminar una noticia guardada como favorita

PCN-29: Obtener las noticias guardadas del usuario actual	
Propósito	Obtener todas las noticias guardadas por el usuario actual
Prerrequisitos	El usuario debe estar registrado y con sesión iniciada.
Datos de entrada	Cabecera de autorización
Resultado esperado	Lista de noticias guardadas (200)
Resultado obtenido	Correcto

Tabla 90. PCN-29: Obtener las noticias guardadas del usuario actual

PCN-30: Añadir una nueva sugerencia	
Propósito	Añadir una nueva sugerencia
Prerrequisitos	El usuario con la sesión iniciada tiene que añadir una sugerencia.
Datos de entrada	Contenido de la sugerencia y la cabecera de autorización
Resultado esperado	Mensaje de creación exitosa (200)
Resultado obtenido	Correcto

Tabla 91. PCN-30: Añadir una nueva sugerencia

PCN-31: Eliminar una sugerencia por su id	
Propósito	Eliminar una sugerencia específica
Prerrequisitos	El usuario con la sesión iniciada tiene que eliminar una sugerencia que haya escrito él.
Datos de entrada	Id de la sugerencia y la cabecera de autorización
Resultado esperado	Mensaje de éxito (200)
Resultado obtenido	Correcto

Tabla 92. PCN-31: Eliminar una sugerencia por su id

PCN-32: Obtener las sugerencias del usuario actual	
Propósito	Obtener todas las sugerencias realizadas por el usuario actual
Prerrequisitos	El usuario debe estar registrado y con sesión iniciada.
Datos de entrada	Cabecera de autorización
Resultado esperado	Lista de sugerencias (200)
Resultado obtenido	Correcto

Tabla 93. PCN-32: Obtener las sugerencias del usuario actual

PCN-33: Obtener todas las sugerencias	
Propósito	Obtener una lista de todas las sugerencias
Prerrequisitos	El usuario debe tener privilegios de administrador.
Datos de entrada	Cabecera de autorización
Resultado esperado	Lista de sugerencias (200)
Resultado obtenido	Correcto

Tabla 94. PCN-33: Obtener todas las sugerencias

PCN-34: Aceptar una sugerencia	
Propósito	Actualizar el estado de una sugerencia específica por su id
Prerrequisitos	El usuario debe tener privilegios de administrador.
Datos de entrada	Id de la sugerencia y la cabecera de autorización
Resultado esperado	Mensaje de éxito (200)
Resultado obtenido	Correcto

Tabla 95. PCN-34: Aceptar una sugerencia

Capítulo 7

Conclusiones

En esta sección se detallan las conclusiones tras haber desarrollado el proyecto de OVERNEWS (Sección 7.1) y también se definen las siguientes mejoras y ampliaciones a realizar (Sección 7.2).

7.1 Conclusiones

Llevar a cabo este proyecto ha sido una experiencia impresionante y emocionante. Al diseñar OVERNEWS, pude poner en práctica todo lo aprendido a lo largo de la carrera en estos últimos cuatro años. Desde la planificación y el diseño del sistema hasta la implementación y la depuración, cada etapa ha ayudado a mejorar mis habilidades y conocimientos en el campo de la Ingeniería Informática.

A pesar de los problemas que han ido surgiendo y de que me ha llevado más tiempo de lo que me habría gustado, estoy muy contento con el resultado. En términos de aprendizaje, lo que he conseguido con este proyecto, ha sido a enfrentar y resolver problemas complejos, colaborar con una amplia gama de tecnología y herramientas diversas, y la importancia de una planificación meticulosa y una ejecución cuidadosa.

Además, el haber seleccionado *frameworks* con los que nunca había trabajado como por ejemplo *FastApi*, ha hecho que se incremente de forma exponencial el aprendizaje a lo largo del desarrollo del proyecto.

El placer de ver cómo una idea aparentemente compleja se ha convertido en una aplicación útil y funcional, es muy satisfactorio.

7.2 Líneas de trabajo futuras

En esta sección, se definirán las siguientes funcionalidades a implementar si se deseara seguir con este proyecto, ya que siempre hay espacio para mejoras y ampliaciones. Algunas de las líneas de trabajo a futuro que considero son las siguientes:

- **Implementación de Pruebas Unitarias:** Esto supone integrar pruebas unitarias tanto en el *frontend* como en el *backend*. Para el *backend* se implementaría *PyTest* y para el *frontend* se utilizaría *Jest* o *Mocha*.
- **Notificaciones Personalizadas:** Esta funcionalidad permitiría que los usuarios pudieran recibir noticias en su correo electrónico según las fuentes y categorías de su

preferencia.

- **Compartir noticias:** Esta nueva funcionalidad permitiría que los usuarios pudiesen compartir las noticias a través de las redes sociales como WhatsApp, Instagram y X.
- **Chat de Usuarios:** Desarrollar una funcionalidad de *chat* que permita a los usuarios interactuar entre ellos en directo. Así los usuarios podrían compartir opiniones y conectarse con otros usuarios en tiempo real.

Estas líneas de trabajo a futuro harían que se reforzase la comunidad de OVERNEWS.

Webgrafía

- [1] Xataka. (2023, enero 15). *Google News: Qué es y cómo configurarlo para que te muestre las noticias que te interesan*. <https://www.xataka.com/basics/google-news-que-como-configurarlo-te-muestre-noticias-que-te-interesan> Última vez visitado: 05/2024.
- [2] Codina, L. (2023, junio 20). *Flipboard: Cómo utilizar el agregador de información en su máximo potencial*. <https://www.lluiscodina.com/flipboard-agregador-de-informacion/> Última vez visitado: 05/2024.
- [3] Arsys. (2023, marzo 10). *PostgreSQL: Qué es y para qué se utilizan estos servidores*. <https://www.arsys.es/blog/postgresql-servidores> Última vez visitado: 07/2024
- [4] Platzi. (2023, mayo 5). *¿Qué es PostgreSQL?* <https://platzi.com/blog/que-es-postgresql/> Última vez visitado: 05/2024.
- [5] Tiangolo. (2023). *Características de FastAPI*. <https://fastapi.tiangolo.com/es/features/> Última vez visitado: 05/2024.
- [6] Profile. (2023, abril 18). *¿Qué es Node.js?* <https://profile.es/blog/que-es-nodejs/> Última vez visitado: 05/2024.
- [7] React. (2023). *Guía de inicio de React*. <https://es.reactjs.org/docs/getting-started.html> Última vez visitado: 05/2024.
- [8] 611 Digital. (2023). *¿Qué es Material UI y por qué deberíamos usarlo?* <https://611digital.com/que-es-material-ui-y-por-que-deberiamos-usarlo/#:~:text=Material%20UI%20es%20una%20librer%C3%ADa,web%20en%20muy%20poco%20tiempo> Última vez visitado: 05/2024.
- [9] Git. (2023). *Inicio - Sobre el Control de Versiones: Acerca del Control de Versiones*. <https://git-scm.com/book/es/v2/Inicio---Sobre-el-Control-de-Versiones-Acerca-del-Control-de-Versiones> Última vez visitado: 05/2024.
- [10] Xataka. (2023, marzo 8). *¿Qué es GitHub y qué le ofrece a los desarrolladores?* <https://www.xataka.com/basics/que-github-que-le-ofrece-a-desarrolladores> Última vez visitado: 05/2024.
- [11] Guevara, A. (2023, abril 12). *Modelo Espiral*. Medium. <https://anaguevara560.medium.com/modelo-espiral-20d49551f565> Última vez visitado: 05/2024.

- [12] Cillero, M. (2023). *Método Albrecht*. <https://manuel.cillero.es/doc/metodologia/metrica-3/tecnicas/tecnicas-de-estimacion/metodo-albrecht/> Última vez visitado: 06/2024.
- [13] González Cabrero, F. J. (2022). *Elicitación de Requisitos*. Universidad de Valladolid.
- [14] Encinas Plaza, S. (2023). *Arquitectura de una aplicación empresarial*. Universidad de Valladolid.
- [15] González Cabrero, F. J. (2020). *PDS Diagramas de Interacción*. Universidad de Valladolid.
- [16] Martínez Prieto, M. Á. (2020). *Modelo Entidad-Relación*. Universidad de Valladolid.
- [17] Martínez Prieto, M. Á. (2020). *Metodología de Diseño Lógico*. Universidad de Valladolid.
- [18] KeepCoding. (2021) *Qué son las pruebas de caja negra*. <https://keepcoding.io/blog/que-son-las-pruebas-de-caja-negra/>

Anexo I Manual de despliegue

En este primer anexo, se informará en detalle cómo poder desplegar correctamente la aplicación. Es importante resaltar que actualmente, para ejecutar esta aplicación, se necesita una máquina que tenga instalado *Windows* como sistema operativo. Además, recomiendo encarecidamente usar *Visual Studio Code* como IDE (*Integrated Development Environment*).

Como ya se ha visto a lo largo de la memoria, esta aplicación la forman 3 componentes que son independientes entre sí.

Manual para el *Backend*:

Para poder desplegar el *backend*, lo primero de todo necesitarás tener la carpeta de Backend localizada en el sistema y deberás de tener instalado Python en la máquina, preferiblemente la versión más reciente (<https://www.python.org/downloads/>). Es por esto por lo que desde la terminal deberás de ejecutar los siguientes comandos:

1. `cd ./ruta_Backend`
2. `python -m venv .venv`: Para inicializar en entorno virtual.
3. `Set-ExecutionPolicy -Scope Process -ExecutionPolicy Bypass`: Para poder ejecutar el script que active el entorno virtual.
4. `./venv/Scripts/activate.ps1`: Para activar el entorno virtual.
5. `pip install -r .\requirements.txt`: Para descargar todas las librerías requeridas
6. `python main.py`: Para ejecutar el *backend*. Hay que tener en cuenta que siempre que se ejecute el *backend*, se llevará a cabo la obtención de noticias.

Entre el comando 5 y 6, deberás de configurar las variables globales que se encuentran en el archivo `.env.example`. Al terminar de configurarlo, debes de renombrar el fichero a `.env`.

Paso preliminar:

Para arrancar la base de datos se debe de instalar PGADMIN4 (<https://www.pgadmin.org/download/pgadmin-4-windows/>) y se debe de configurar las variables de entorno en el fichero `.env`. A continuación, debes de crear una base de datos desde PGADMIN4 dando clic derecho sobre Database >> Create .

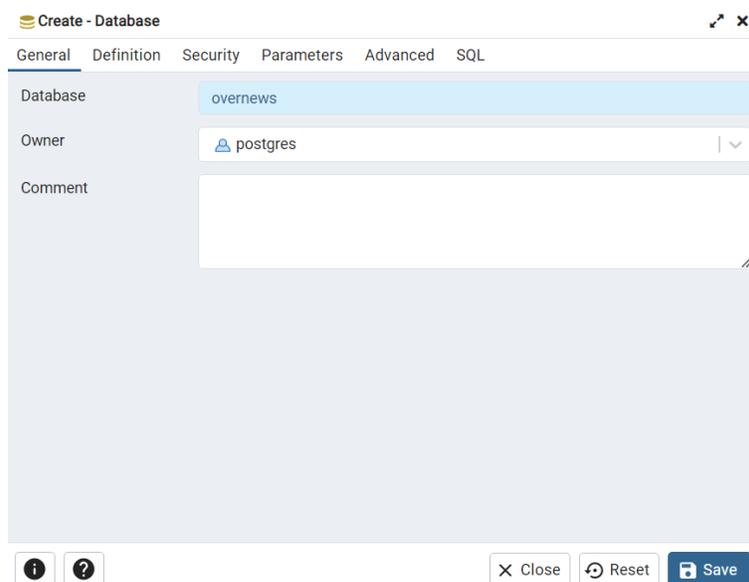


Figura 69. Creación BD desde PGAdmin 4

Después, tiene que configurar las variables globales de la base de datos con el nombre, usuario, puesto, contraseña y host de la nueva base de datos.

```
# Database connection
DB_NAME=
DB_USER=
DB_PASS=
DB_HOST=
DB_PORT=
```

Figura 70. Variables globales de la base de datos

Una vez conectada la base de datos con el *backend* se deben de ejecutar los siguientes comandos para crear la misma base de datos con datos ya inicializados:

1. `cd ./ruta_Backend`
2. `python` : Abrimos la consola de Python.
3. `from api.scripts.create_tables import create_tables :`
 Importamos el método `create_tables()` .
4. `create_tables()` : Ejecutamos el método `create_tables()` .

Y de esta forma, ya se ha creado e inicilializado la base de datos correctamente como se observa en la siguiente imagen.

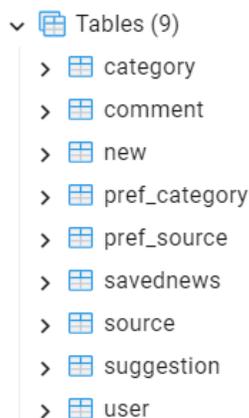


Figura 71. Base de datos creada

Manual para el *Frontend*:

Para desplegar el *frontend*, es necesario instalarse Node.js la máquina (<https://nodejs.org/en/download/prebuilt-installer>) . Una vez instalado Node.js, debemos de conocer a ruta en la que se encuentra la carpeta `ruta_Frontend`, abrir una terminal y ejecutar los siguientes comandos:

1. `cd ./ruta_Frontend`
2. `npm install` : Instala las dependencias requeridas del *frontend*.
3. `npm run start` : Despliegue el *frontend*.

De esta forma, ya arranca el *frontend* de forma correcta.

Manual para la API de traducción en local:

Para poder arrancar correctamente la API, deberás de tener instalado PYTHON. Debido a que los modelos de IA (Inteligencia Artificial) entrenados ocupan 2 GB, no puedo compartirlo. Por ello, se deberá bajar desde *Git*.

Para poder bajarlo se deben seguir los siguientes pasos:

1. `git clone https://github.com/kylegallatin/machine-translation-service.git` : Clona el repositorio.
2. `cd cd ./ruta_API`
3. `mkdir data` : Crea la carpeta donde se van a almacenar los modelos de IA

entrenados.

4. `python download_models.py --source es --target {idiomas}`: Ejecuta este comando para cada uno de los diferentes idiomas a los que se traducirán las noticias (Figura 50).

Teniendo los modelos instalados, ya podrá ejecutar la API de traducción:

1. `cd ./ruta_API`
2. `python -m venv .venv`: Para inicializar en entorno virtual.
3. `Set-ExecutionPolicy -Scope Process -ExecutionPolicy Bypass`: Para poder ejecutar el script que active el entorno virtual.
4. `./venv/Scripts/activate.ps1`: Para activar el entorno virtual
5. `pip install -r .\requirements.txt`: Para descargar todas las dependencias requeridas.
6. `python app.py`: Para ejecutar la api de traducción.

Anexo II Manual de usuario

En este segundo anexo se proporcionará el manual de uso para los usuarios sin permisos de administrador, detallando cómo debe utilizar la aplicación.

Página principal (usuario visitante):

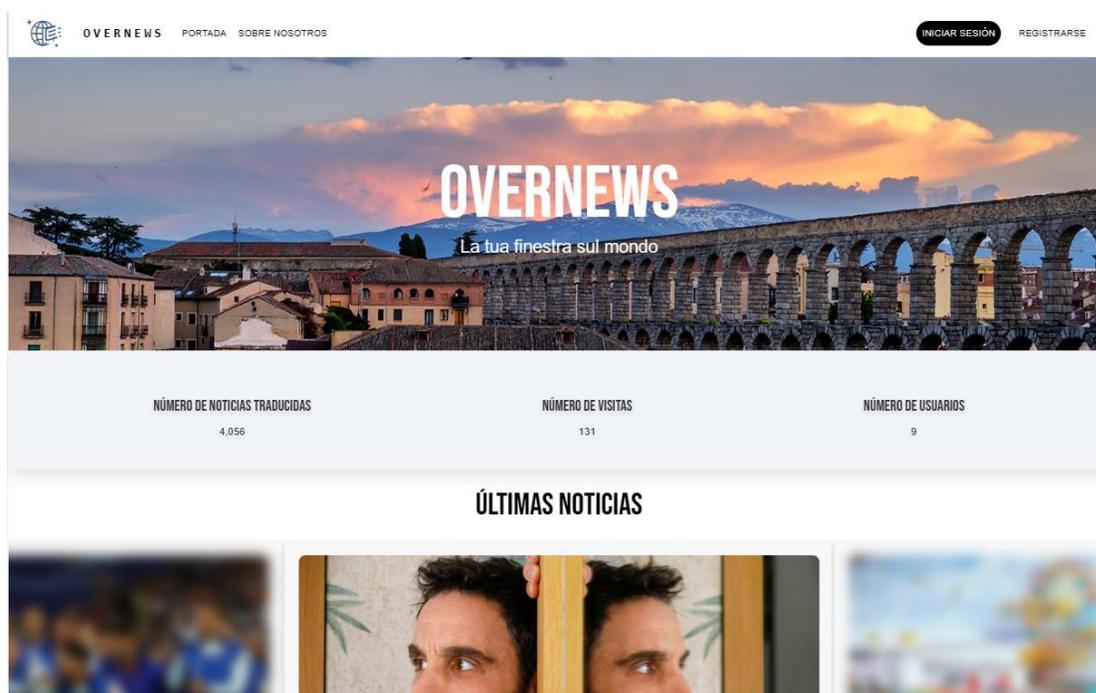


Figura 72. Página Principal OverNews Visitante

Registro:

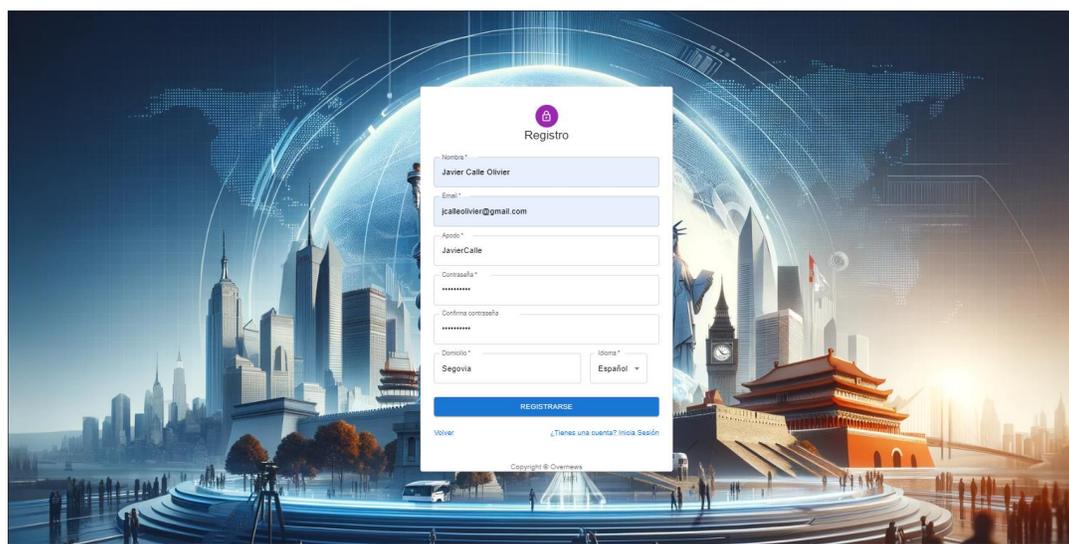


Figura 73. Pantalla Registro de Usuario

Al registrarse, se le manda un correo electrónico para que valide su cuenta:

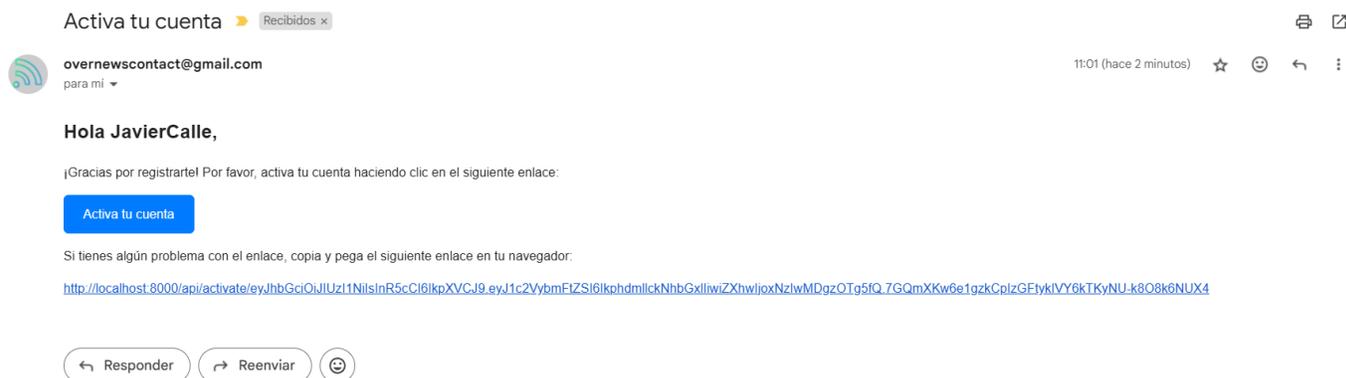


Figura 74. Correo activación de cuenta

Cuando pulse el botón de “Activar tu cuenta”, le aparece la siguiente ventana:



Figura 75. Activación completada

Si vuelve a pulsar el botón de “Activar tu cuenta” una vez que ya ha activado su cuenta previamente, le aparecerá la siguiente pantalla:



Figura 76. Error en la activación

Inicio de sesión:



Figura 77. Pantalla Inicia de Sesión

INICIAR SESIÓN

[Recuperar contraseña](#)

INICIAR SESIÓN

[Volver](#)

[¿No tienes cuenta? Regístrate](#)

Copyright © Overnews

Si el usuario inicia sesión correctamente, se le redirigirá a la página principal. Si no, se le indicará que ha fallado el inicio de sesión:



Figura 78. Error al Iniciar Sesión

INICIAR SESIÓN

❌ Error al iniciar sesión

[Recuperar contraseña](#)

INICIAR SESIÓN

[Volver](#)

[¿No tienes cuenta? Regístrate](#)

Copyright © Overnews

Página principal (Usuario con sesión iniciada):

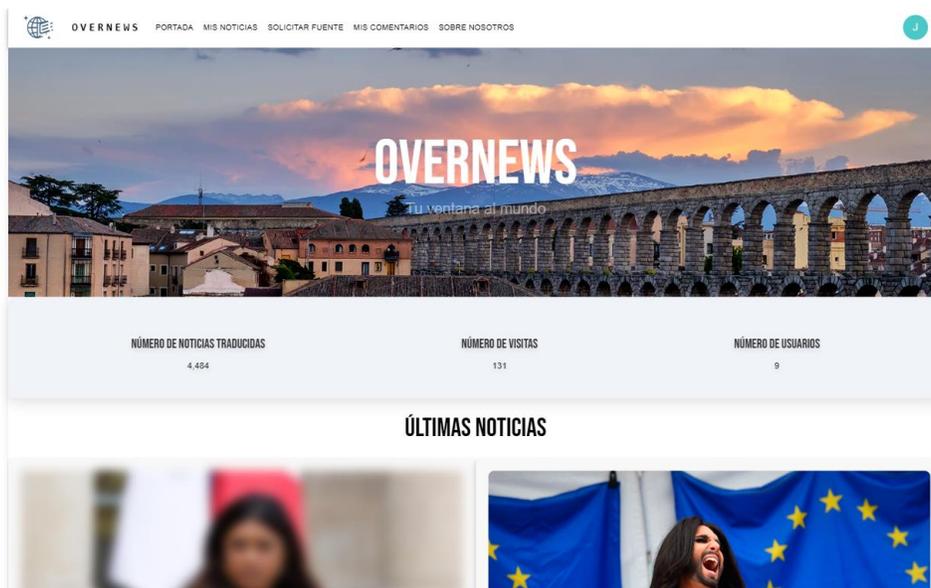


Figura 79. Página Principal OverNews Usuario

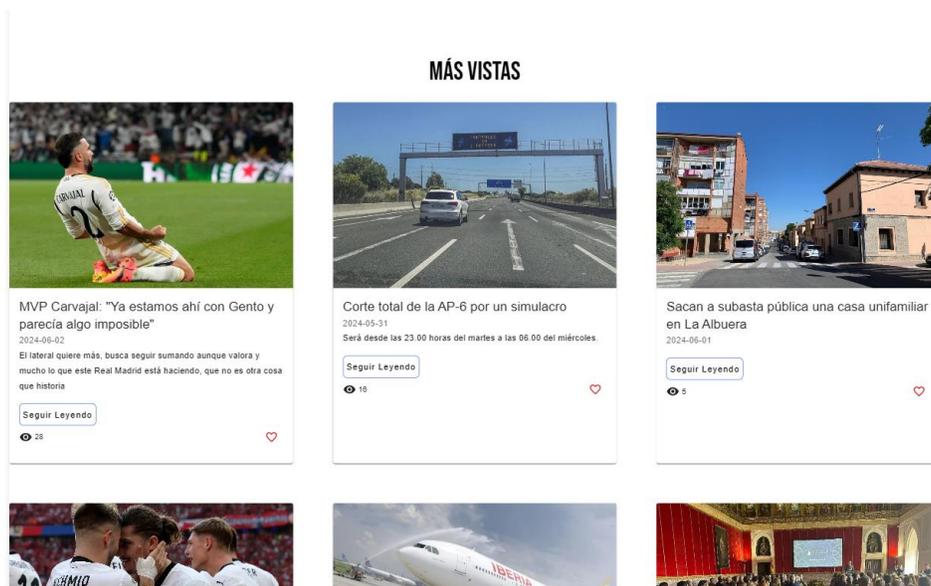


Figura 80. Página Principal (Noticias más vistas)

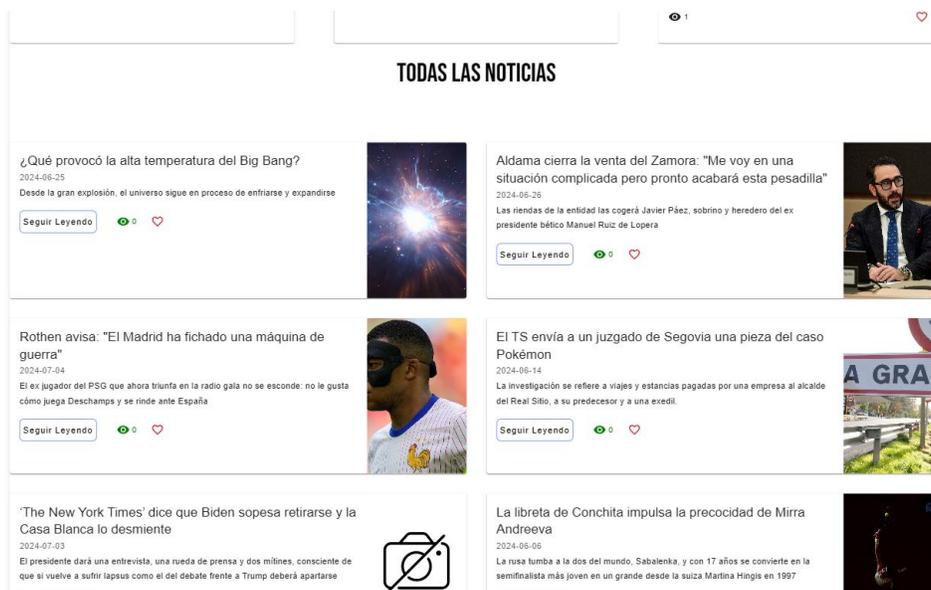


Figura 81. Página Principal (Todas las noticias)

Página Mis Noticias:

En esta página aparecen sus preferencias de las noticias y sus noticias favoritas.

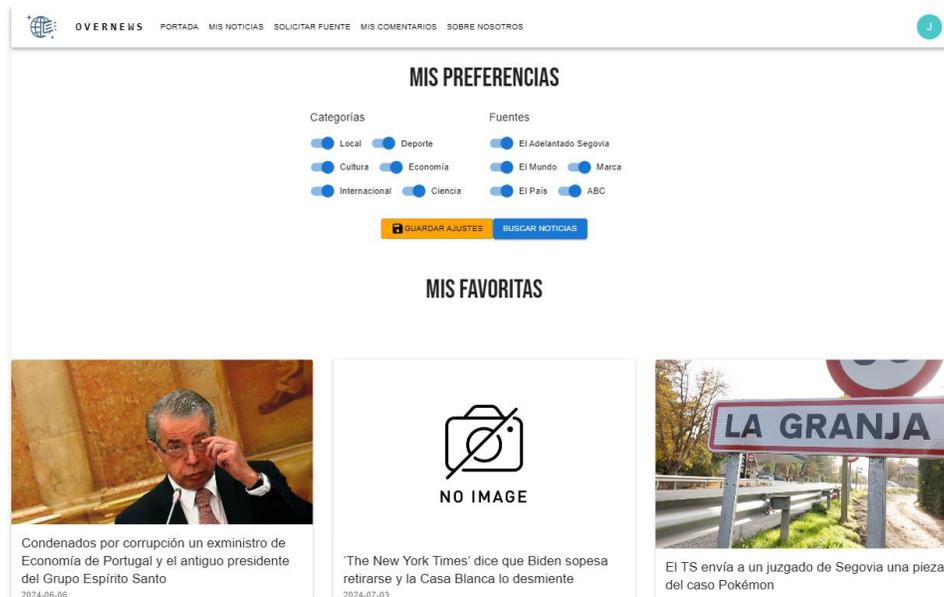


Figura 82. Página Mis Noticias

Si se pulsa el botón de “Buscar Noticias” se despliegan las noticias que cumplen con los filtros de categoría y fuente:

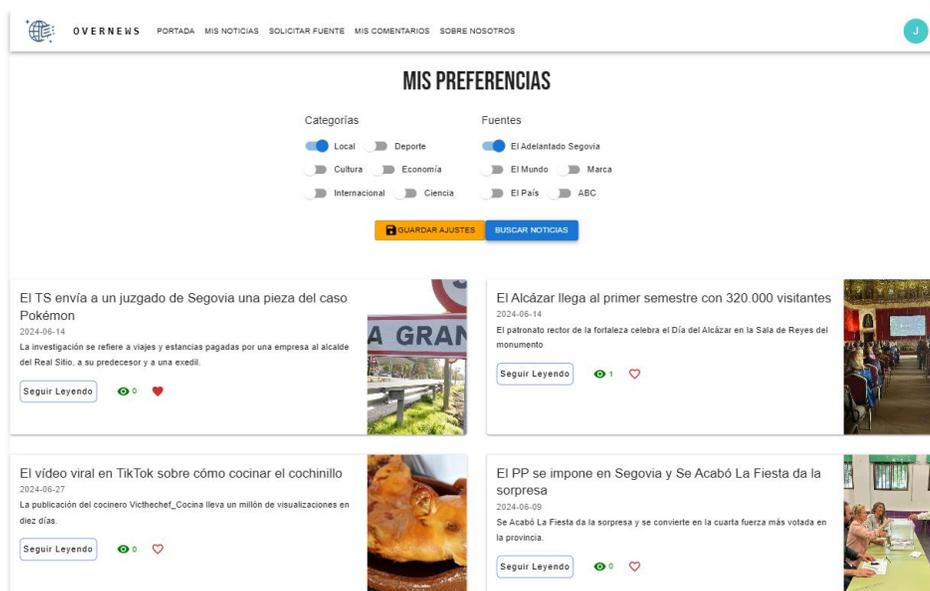


Figura 83. Página Mis Noticias al pulsar "Buscar Noticias"

Si se pulsa el botón de “Guardar Ajustes” se guardan las preferencias de forma que hasta que no vuelva a guardar otras preferencias, le aparecerán las indicadas en el momento que pulsó el botón.

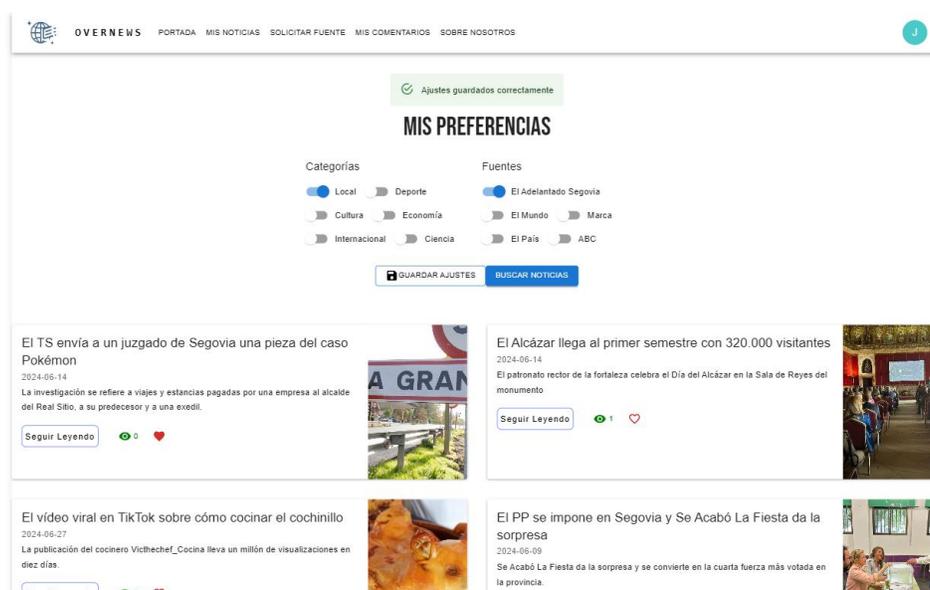


Figura 84. Página Mis Noticias al "Guardar Ajustes"

Pantalla Solicitar Fuente:

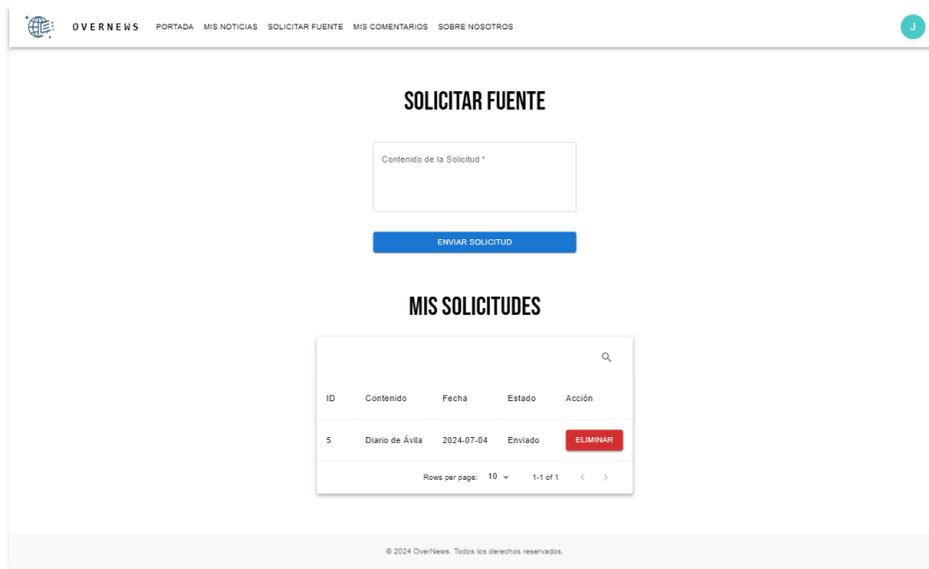


Figura 85. Pantalla Solicitar Fuente

Si pulsa el botón de “Solicitar Fuente”:

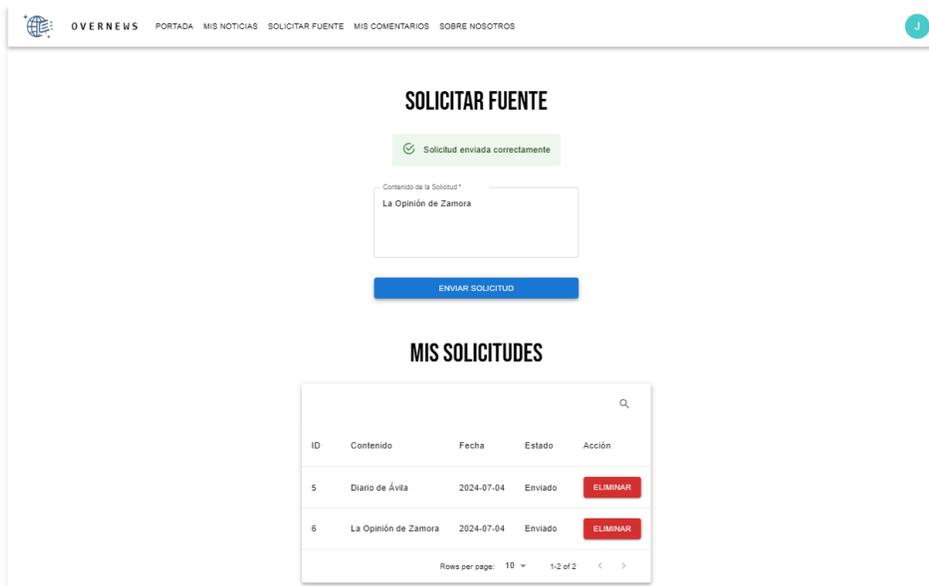
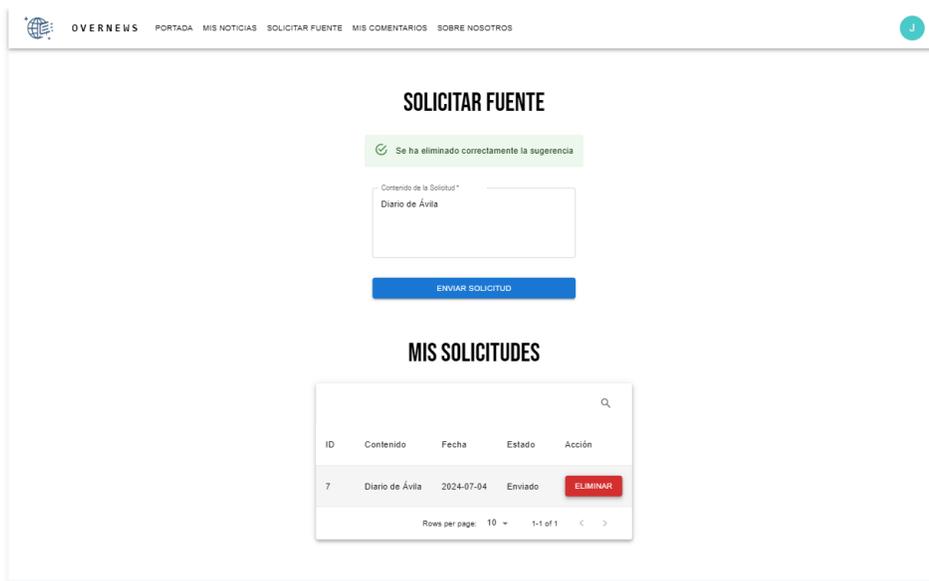


Figura 86. Pantalla Solicitar Fuente al "Añadir Sugerencia"

Si pulsa al botón de “Eliminar Sugerencia” se muestra el siguiente mensaje:



Pantalla Mis Comentarios:



Figura 87. Pantalla Mis Comentarios

Pantalla Sobre Nosotros:

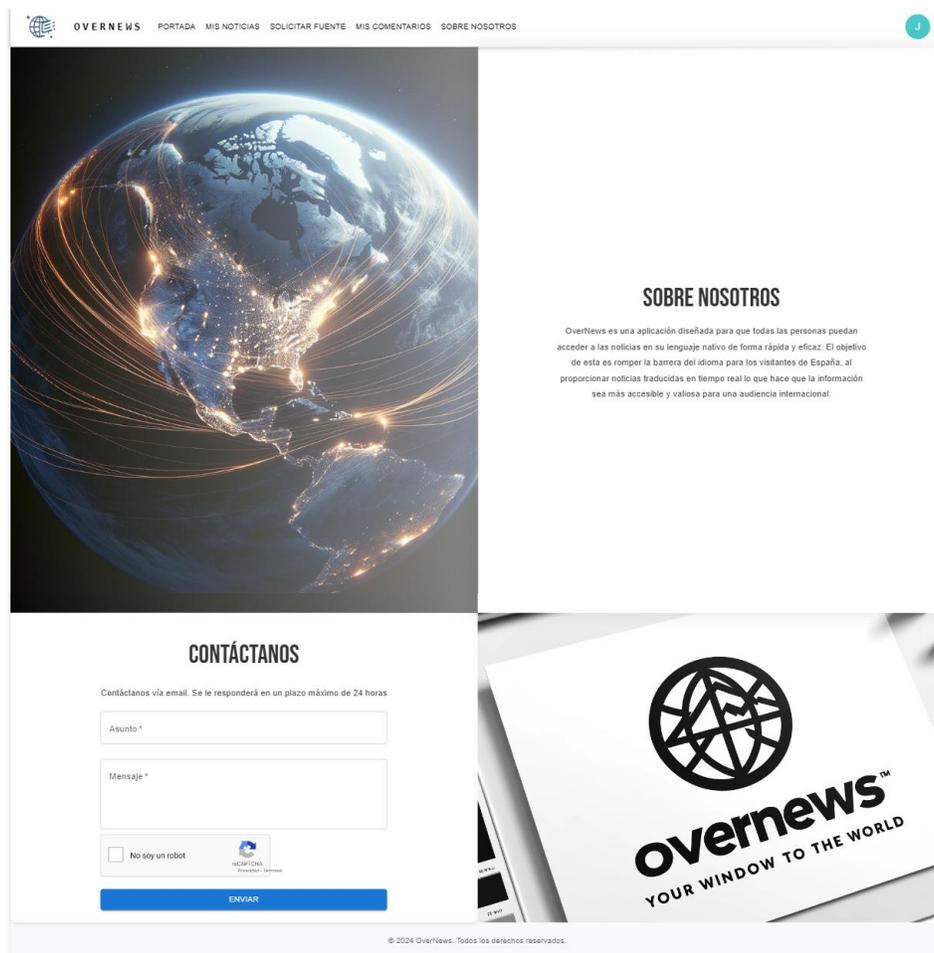


Figura 88. Pantalla Sobre Nosotros

Si el usuario quiere “Enviar” un mensaje a Contáctanos deberá de rellenar el reCAPTCHA

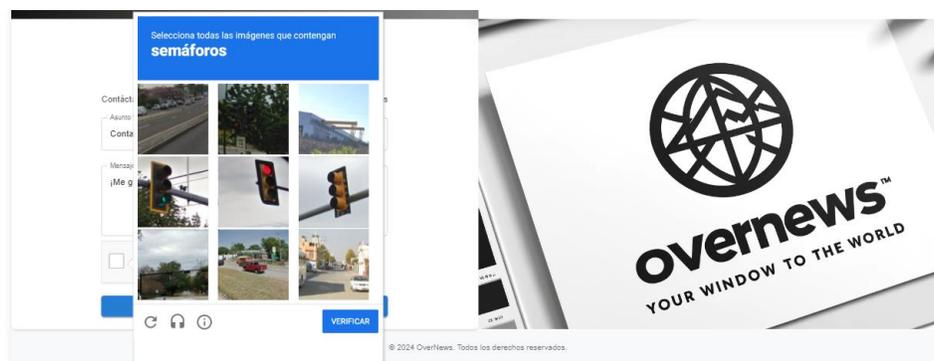


Figura 89. ReCAPTCHA al "Contáctanos"

Si el usuario quiere “Enviar” un mensaje, pero no completa el recaptcha, se le mostrará un error:

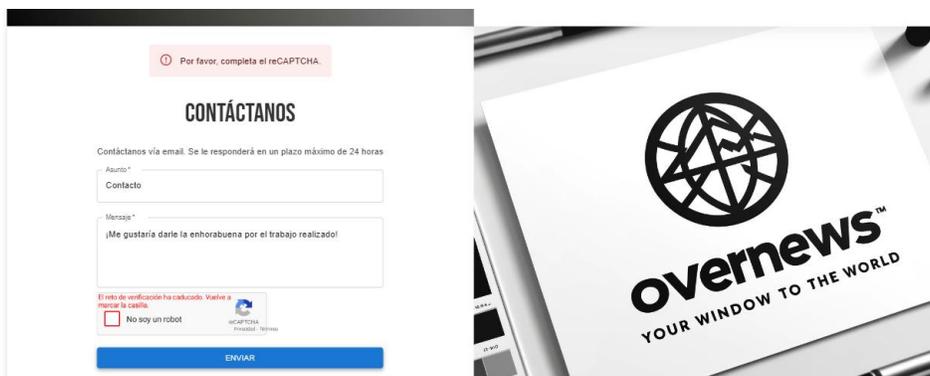
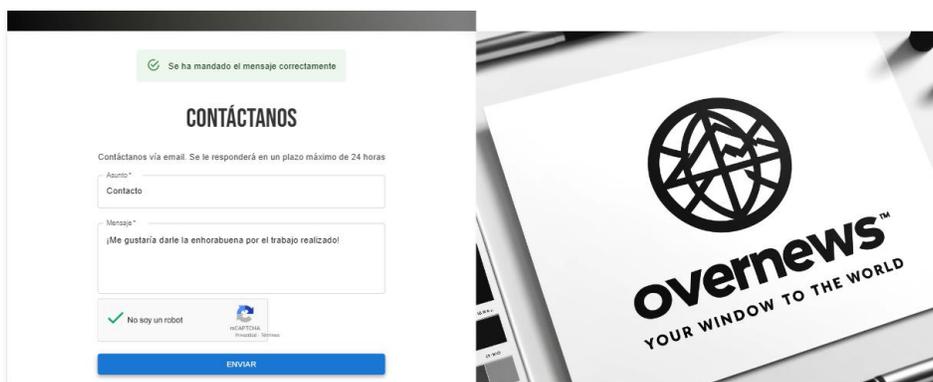


Figura 90. Error por no rellenar recaptcha

Si el usuario relleno correctamente el recaptcha y rellena los campos de Asunto y Contenido, se enviará correctamente el Gmail al correo de soporte.



Y soporte recibirá el siguiente mensaje:



Figura 91. Correo electrónico "Contáctanos"

Pantalla Ver Noticia:

OVERNEWS PORTADA MIS NOTICIAS SOLICITAR FUENTE MIS COMENTARIOS SOBRE NOSOTROS



Corte total de la AP-6 por un simulacro

2024-05-31

Será desde las 23.00 horas del martes a las 06.00 del miércoles.

El Ministerio de Transportes y Movilidad Sostenible ha anunciado que en la noche del 4 al 5 de junio se va a desarrollar un simulacro de incidente en la AP6, en el tramo de 'túneles de Guadarrama', que va a afectar al tráfico de esta vía que enlaza Madrid con Segovia. Para garantizar la seguridad de los trabajadores y usuarios, y en colaboración con la Sociedad Concesionaria Castellana Autopistas, desde las 23:00 horas del martes 4 hasta las 6:00 horas del miércoles habrá cortes totales de ambas calzadas. Los tramos con estos cortes completos son la calzada sentido A Coruña entre los kilómetros 52 y 61; y en sentido Madrid entre los kilómetros 61 y 47. -Corte total de la calzada sentido A Coruña entre los km 52 y 61. Se establecerá un desvío hacia la N-6 por la salida 50 de la autopista. Posteriormente, el tráfico se podrá reincorporar a la calzada sentido A Coruña a través del enlace de San Rafael en el km 64 de la N-603. -Corte total de la calzada sentido Madrid entre los km 61 y 47. En este caso, se habilitará un desvío por la salida 60 hacia la carretera N-603/N-6. Posteriormente, el tráfico se podrá reincorporar a la calzada sentido Madrid a través del enlace del Valle de Cuelgamuros, en el km 47. Mientras duren los desvíos, se ruega máxima precaución y atención a la señalización, junto con el estricto cumplimiento de las recomendaciones de la Dirección General de Tráfico (DGT), visibles en sus carteles luminosos de señalización variable.

EL ADELANTADO

Local

[Ir a la noticia original](#)

Comentarios

Escribe un comentario [Comentar](#)

A Acohamberlan
Vaya por dios, he pillado tráfico

J JavierCalle
Vayaa, a mi también me ha pillado :(

[Volver](#)

© 2024 OverNews. Todos los derechos reservados.

Figura 92. Pantalla Ver Noticia

Si pulsa el botón de “Ir a la noticia”, se le redirigirá a la url de la noticia original. Además, el usuario también podrá añadir comentarios:



Figura 93. Añadir Comentario a una noticia

Y también tendrá las opciones de eliminar y editar el comentario.



Y podrá guardarlo o cancelarlo.

Pantalla Gestionar mi Perfil:

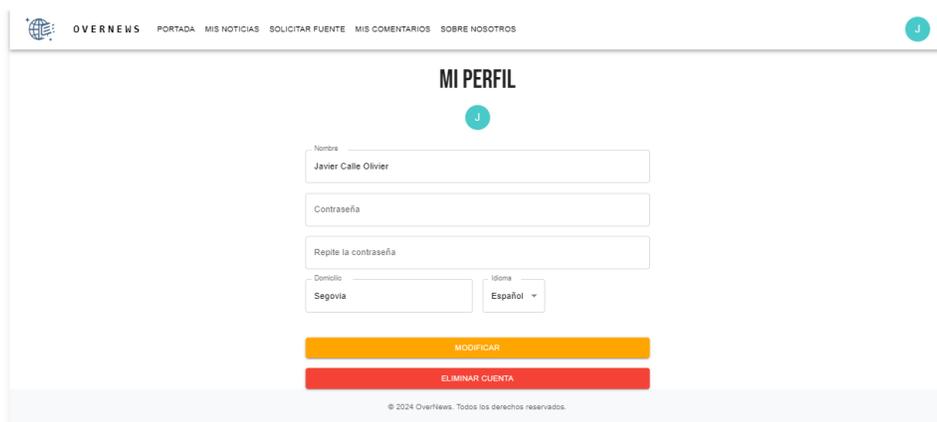


Figura 94. Pantalla Gestionar Mi Perfil

Si pulsa botón de “Modificar” y si que ha modificado algún campo, se le redirige a a página principal y se modifican los datos correctamente.

Y si pulsa el botón de “Eliminar Cuenta”, se le mostrará el siguiente *popup* indicándole que la acción es definitiva.

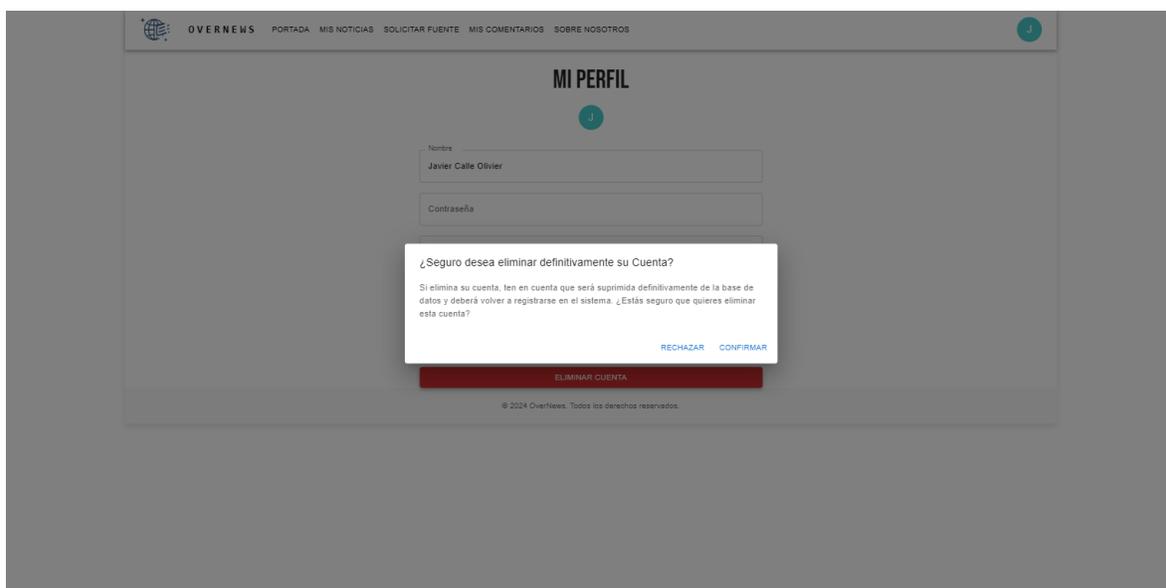


Figura 95. PopUp Aviso al "Eliminar Cuenta"

Recuperar Contraseña:

La única forma de acceder a esta opción es desde Iniciar Sesión pusando “Recuperar Contraseña”.



Figura 96. Pantalla Recuperar Contraseña

Así se queda la pantalla cuando se envía correctamente el correo:



Figura 97. Correo Enviado al "Recuperar Contraseña"

Y en su buzón le aparecerá el siguiente correo:



Figura 98. Correo de Recuperación de Contraseña

Y si cambias la contraseña:



Figura 99. Página para Recuperar la Contraseña

404 Not Found:

Por último, faltaría mencionar que si el usuario introduce una url incorrecta dentro la aplicación de se le redirigirá a la siguiente página indicándole que esa página no existe:

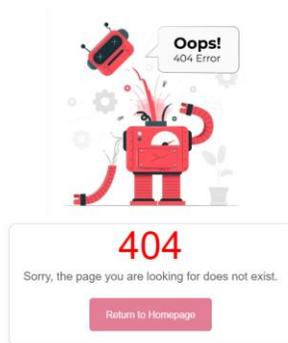


Figura 100. Página 404 Not Found

Dentro de esta página, hay un botón que permite al usuario redirigirse a la página principal.

Anexo III Manual de administrador

En el último anexo se proporcionará el manual de uso para los usuarios tienen permisos de administrador. Resaltar que estos usuarios van a iniciar sesión y les aparecerá directamente el Panel del Administrador:

PANEL DEL ADMINISTRADOR

GRÁFICA EVOLUTIVA DE LAS NOTICIAS

ÚLTIMA SEMANA ÚLTIMO AÑO

Nº de noticias

Fechas

GESTIONAR SUGERENCIAS

ID	Contenido	Fecha	Usuario	Estado	Actions
4	Besoccer	2024-06-30	2	Enviado	ELIMINAR ACEPTAR
7	Diario de Ávila	2024-07-04	20	Enviado	ELIMINAR ACEPTAR

Rows per page: 10 1-2 of 2

GESTIONAR USUARIOS

Email	Nombre de Usuario	Acción
caremencalle@gmail.com	Cco	ELIMINAR
caremenoliver@gmail.com	Coc	ELIMINAR
mcalle@gmail.com	MACF	ELIMINAR
davidpino@gmail.com	Dava	ELIMINAR
marioolmos@gmail.com	Rata Olmos	ELIMINAR
davidpinoalumno@gmail.com	Davaa	ELIMINAR

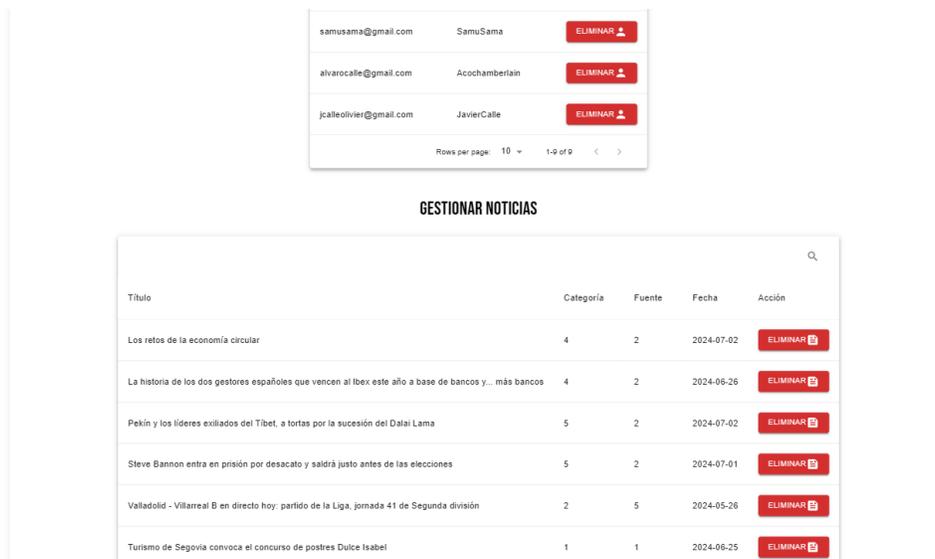


Figura 101. Pantalla Panel del Administrador

El administrador puede cambiarle el color y la escala (último año o última semana) a gráfica de la evolución de las noticias:

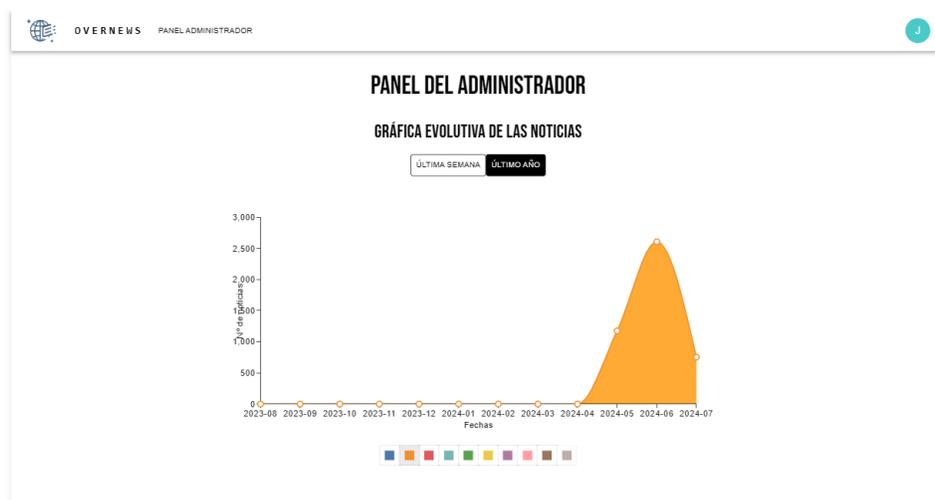


Figura 102. Panel del Administrador con la Gráfica Modificada

También va a poder aceptar las sugerencias que han llevado a cabo los usuarios:

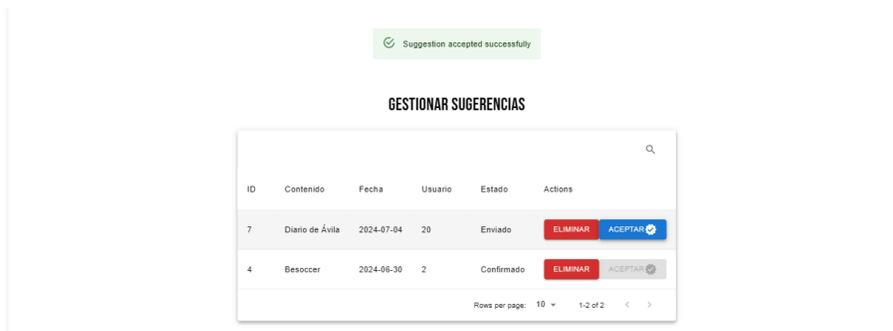


Figura 103. Sugerencia Aceptada por el Administrador

Y también va a poder eliminar sugerencias, noticias, usuarios y comentarios como en el ejemplo de la siguiente imagen:

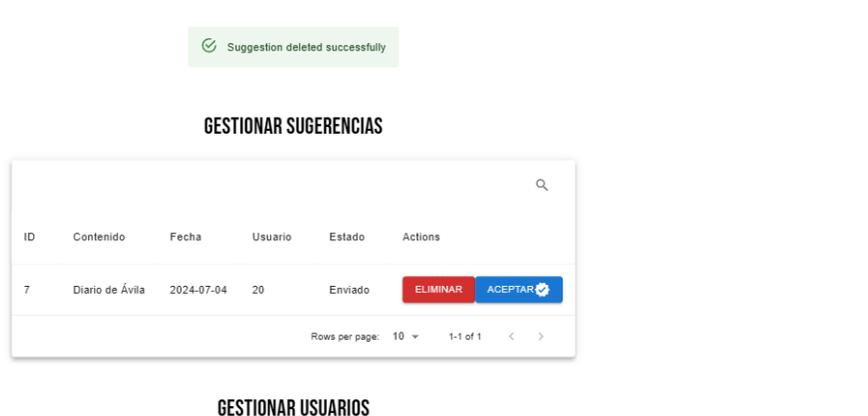


Figura 104. Sugerencia eliminada por el Administrador