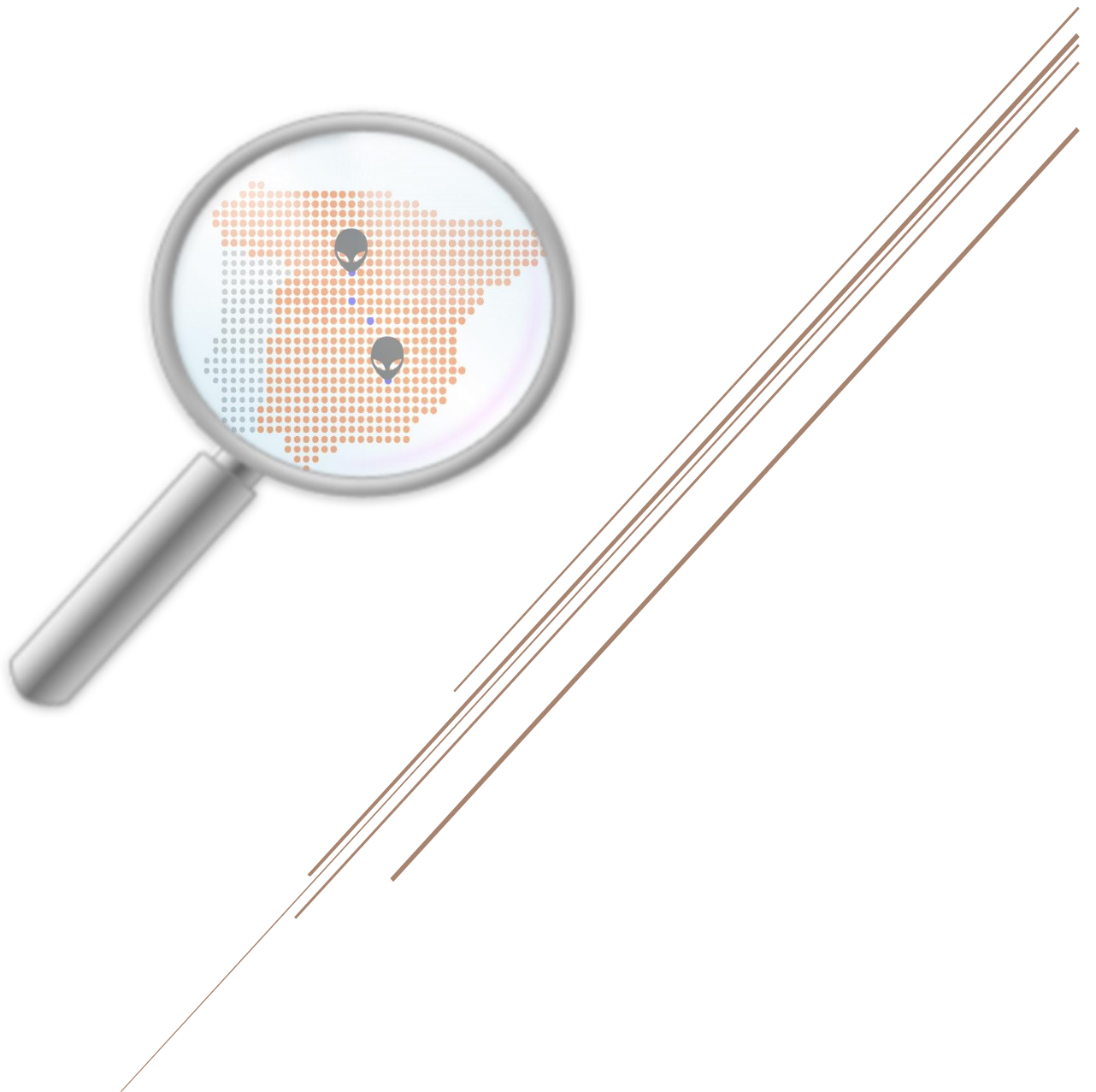




MISTERY TRACKS SPAIN

Proyecto DAW 18/19



Iván Casado Gutierrez
72141582N



TABLA CONTENIDOS

TABLA ILUSTRACIONES	4
RESUMEN.....	6
INTRODUCCIÓN.....	7
OBJETIVOS	8
Objetivo General	8
Objetivos Específicos	8
1. ANÁLISIS DEL CONTEXTO Y ESTADO DEL ARTE	9
1.1 Análisis del contexto.....	9
1.2 Análisis de tendencias	13
1.3 Competencia	14
2. ESTADO DEL ARTE.....	15
2.1 Estudio	15
2.2 Conclusiones.....	20
3 INNOVACIÓN.....	21
3.1 Aportación.....	21
3.2 Soluciones Innovadoras Aplicadas	21
4. REQUISITOS.....	22
4.1 Requisitos Funcionales	22
4.2 Requisitos No Funcionales	24
5. DISEÑO	25
5. Colores	25
5.2 Diagrama lógico de red	26
5.3 Diagrama Entidad / Relación	28
5.3 Diagrama relacional	30
5.4 Diagrama de clases	31
5.5 Diagrama de interfaces.....	43
5.5.1 Mapa navegación	43
5.5.2 Interfaces	44
5.6 Diseño de algoritmos	56
6. PLANIFICACIÓN.....	61
6.1 Fases.....	61



6.2 Cronograma	62
6.3 Recursos	63
7. DESARROLLO	64
Explicación.....	64
7.1 BACK-END	64
7.1.1 Despachador	65
7.1.2 Estructura del proyecto	69
7.2 FRONT-END	72
7.2.1 Estructura.....	72
7.2.2 Vistas.....	73
7.2.3 Formularios.....	74
7.2.4 Validar Formularios	75
8. PUESTA EN MARCHA.....	77
9. PRUEBAS Y CONTROL DE CALIDAD.....	79
10. PLAN EMPRESARIAL.....	83
10.1 Localización	83
10.2 Inmovilizado intangible.....	84
10.3 Propiedad industrial	84
10.4 Aplicaciones informáticas	84
10.5 Inmovilizado material	86
10.5.1 Terrenos.....	86
10.5.2 Edificaciones y Construcciones	86
10.5.3. Instalaciones	86
10.5.4. Maquinaria y utillaje	87
10.5.5. Mobiliario	88
10.5.6. Equipos proceso de la información.....	88
10.5.7. Elementos de transporte	88
10.5.8. Otro inmovilizado material.....	88
10.6 Presupuesto detallado.....	89
10.6.1 Inversión	89
10.6.2 Gastos	89
10.6.3 Umbral de rentabilidad.....	89
11. PLAN ECONÓMICO FINANCIERO	91
11.1. Plan Económico	91



11.2. Ingresos	92
11.3. Compras y gastos.....	93
11.4. Gastos de personal	93
11.5 Gastos Financieros	94
11.6 Cuenta Explotación Previsional.....	95
11.7 Balance previsional	96
11.8 Análisis Económico	97
11.8.1 Ratios de rentabilidad.....	97
11.9 Análisis Financiero	98
11.9.1 Ratios liquidez	98
11.10 Necesidades Forma Jurídica	98
11.10.1 Necesidades	98
11.10.2 Financiación.....	99
11.10.3 Forma jurídica.....	99
12. Bibliografía.....	100
Conclusiones	101



TABLA ILUSTRACIONES

Ilustración I Perdida Población	10
Ilustración II Ranking Podcast Misterio	12
Ilustración III Evolución visitantes Auschwitz	16
Ilustración IV Llegada visitantes a Medellín	16
Ilustración V Municipios con más extranjeros que españoles	18
Ilustración VI Estadísticas lectores revista más allá	19
Ilustración VII Diagrama red	27
Ilustración VIII Diagrama E/R.....	29
Ilustración IX Diagrama relacional	30
Ilustración X Diagrama de clases.....	31
Ilustración XI Mapa navegación	43
Ilustración XII Pantalla inicio INTF01 Prototipo	44
Ilustración XIII Pantalla inicio INTF01 Resultado Final	44
Ilustración XIV Pantalla Relacionados/Noticias INTF02 Prototipo	45
Ilustración XV Pantalla Relacionados/Noticias INTF02 Resultado Final	45
Ilustración XVI Pantalla Relacionados/Podcast INTF03 Prototipo	46
Ilustración XVII Pantalla Relacionados/Podcast INTF03 Resultado Final	46
Ilustración XVIII Pantalla Relacionados/Videos INTF04 Prototipo	47
Ilustración XIX Pantalla Relacionados/Videos INTF04 Resultado Final	47
Ilustración XX Pestaña Rutas INTF05 Prototipo.....	48
Ilustración XXI Pestaña contacto INTF06 Prototipo	49
Ilustración XXII Pestaña Contacto INTF06 Resultado Final	49
Ilustración XXIII Pestaña foro INTF07 Prototipo	50
Ilustración XXIV Pantalla Login INTF08 Prototipo.....	51
Ilustración XXV Pantalla Login INTF08 Resultado Final	51
Ilustración XXVI Pantalla Registro INTF09	52
Ilustración XXVII Pantalla Registro INTF09 Resultado Final.....	52
Ilustración XXVIII Pantalla subir ruta INTF10.....	53
Ilustración XXIX Cambiar contraseña INTF11	54
Ilustración XXX Pantalla Perfil Usuario INTF12.....	55
Ilustración XXXI Diagrama de flujo Login.....	56
Ilustración XXXII Diagrama flujo Nuevo Usuario	57
Ilustración XXXIII Diagrama Flujo Emitir Voto	58
Ilustración XXXIV Diagrama Flujo Subir Contenido	59
Ilustración XXXV Diagrama Flujo Búsqueda	60
Ilustración XXXVI Gestor de dependenciaspom.xml.....	64
Ilustración XXXVII Anotaciones utilizadas por hibernate	65
Ilustración XXXVIII imagen del package principal	65
Ilustración XXXIX Imagen del mapeo de carpetas	66
Ilustración XL Imagen ViewResolver.....	66
Ilustración XLI Ejemplo ViewResolver	67
Ilustración XLII Ejemplo de salida de datos clase contacto	67



Ilustración XLIII origen de los datos de la BBDD	68
Ilustración XLIV Message properties	68
Ilustración XLV Muestra mensaje properties	68
Ilustración XLVI Manager de entidades	68
Ilustración XLVII Ejemplo anotación @Entity	69
Ilustración XLVIII Estructura de paquetes	69
Ilustración XLIX Controladores de clases	69
Ilustración L Ejemplo clase controladora	70
Ilustración LI Ejemplo interfaz Repository	71
Ilustración LII Inyección de repositorios con @Autowired	71
Ilustración LIII Ejemplo de consultas JPA	71
Ilustración LIV Métodos de acceso a datos de JPA	72
Ilustración LV Estructura Web	72
Ilustración LVI Ejemplo url css	73
Ilustración LVIII Ejemplo inserción imagen	73
Ilustración LIX uso de href	73
Ilustración LX Acciones de los formularios	74
Ilustración LXI Anotaciones validación formularios	75
Ilustración LXII Demostración validación	75
Ilustración LXIII Error contraseñas personalizado	76
Ilustración LXIV errores de contraseñas distintas	76
Ilustración LXV Inversión	91
Ilustración LXVI Fórmula Google Adsense	92
Ilustración LXVII Ventas e ingresos	92
Ilustración LXVIII Gastos y Compras	93
Ilustración LXIX Gastos Personal	93
Ilustración LXX Gastos financieros	94
Ilustración LXXI Explotación previsional	95
Ilustración LXXII Balance previsional	96
Ilustración LXXIII Datos de la empresa	99



RESUMEN

La propuesta del proyecto es un sitio web en el que los usuarios interesados en lugares con misterio en España puedan encontrar mediante geolocalización todos los lugares nacionales con historia y catalogados con misterio.

Conocer las historias de esos lugares y sus leyendas.

Incluyendo secciones con enlaces interesantes sobre la temática y un calendario de eventos.

El objetivo es agilizar las búsquedas de un público creciente que ha perdido el miedo al qué dirán y que no teme contar sus experiencias.

Acercar a la gente lugares nuevos que visitar con una temática poco habitual pudiendo crear rutas personalizadas.

Palabras clave: Misterio, Mistery, Rutas, Tracks, Turismo alternativo, Turismo de experiencia, Paranormal, Sobrenatural, Desconocido, Turismo Oscuro.



INTRODUCCIÓN

Como aficionado a los temas de misterio me he decantado por un proyecto que consiste en la creación de un portal web de misterio que se llamará Mystery Tracks Spain.

Se escoge este tema porque se considera una forma diferente de viajar, muy distinta a la de la mayoría de la gente, y que ofrece experiencias alternativas y únicas que la mayoría de la población no entiende porque piensan que estos temas están solo destinados para los “locos”.

Como usuario de este tipo de páginas en ocasiones me he visto con problemas para encontrar información de sitios que visitar.

La propuesta del proyecto es un sitio web en el que los usuarios interesados por lugares con misterio en España puedan encontrar mediante geolocalización todos los lugares nacionales con historia y catalogados con misterio. Incluyendo secciones con enlaces interesantes sobre la temática y un calendario de eventos.

Refiriéndose a lugares con misterio como:

- Casas encantadas
- Apariciones O.V.N.I
- Zonas marcadas por la guerra
- Acontecimientos trágicos
- Pueblos abandonados...
- Etc.

El portal servirá también como foro gracias a su sistema de comentarios y las rutas creadas por los usuarios a su vez guardarán las mejores puntuaciones de usuarios y rutas, éstas serán mostradas en la web.

Para interactuar con la página habrá varios tipos de usuarios, los usuarios anónimos solo podrán realizar búsquedas en el mapa.

Los usuarios registrados podrán postear en los foros lugares nuevos y sus propias rutas. Para evitar engaños solo los usuarios con puntuaciones más altas podrán colaborar sin supervisión de los administradores.

El propósito final es generar en torno a la web un punto de encuentro para la comunidad de amantes de lo paranormal, en el que puedan compartir con otros usuarios con las mismas inquietudes. Además de agilizar las búsquedas reuniendo toda la información en un mismo lugar.



OBJETIVOS

Objetivo General

- Facilitar una plataforma colaborativa en la que los usuarios puedan buscar rutas con temática “*Paranormal*” además de subir las suyas propias.
- Servir como nexo de unión facilitando a los usuarios comunicarse entre sí.
- Favorecer el turismo rural haciendo crecer la economía de los pueblos que se están quedando despoblados.

Objetivos Específicos

- OBJ1 Crear diferentes tipos de usuarios.
- OBJ2 Crear una interfaz sencilla y fácil de utilizar para el usuario.
- OBJ3 Integrar un mapa interactivo para ayudar al usuario a realizar las búsquedas.
- OBJ4 Ofrecer un sistema de puntuaciones de usuarios y rutas.
- OBJ5 Ofrecer al usuario un apartado con enlaces relacionados a la temática de la plataforma.
- OBJ6 Integrar un calendario con información sobre próximos eventos.
- OBJ7 Permitir al usuario comprar entradas o hacer reservas sin tener que abandonar la página.
- OBJ8 Permitir a los usuarios añadir nuevos contenidos.
- OBJ9 Crear un foro para que se comuniquen los usuarios.
- OBJ10 Permitir a los usuarios cambiar la contraseña.
- OBJ11 Permitir a los usuarios consultar sus datos.



1. ANÁLISIS DEL CONTEXTO Y ESTADO DEL ARTE

1.1 Análisis del contexto

En la actualidad, el turismo se ha convertido en una de las actividades económicas de mayor crecimiento a nivel mundial. A pesar de las reiteradas crisis y del detrimento social en muchos lugares del mundo, los datos estadísticos del sector dan una imagen positiva en términos de crecimiento económico.

La población española ha aumentado alrededor de un 36% desde 1975: se ha pasado de tener 34,2 millones de habitantes a cerca de 46,7 millones, pero este aumento no es en todas las zonas por igual.

Durante los últimos años, en los que el país ha sufrido una revolución económica, amplias regiones del país se han visto afectadas por movimientos migratorios de gran importancia desde las zonas rurales hasta las grandes ciudades.

Provincias como Soria han visto como su población se reducía en este periodo más de un 23%, según las cifras de población que maneja el Instituto Nacional de Estadística (INE), mientras que otras como Madrid han crecido un 73% impulsadas por el crecimiento de la capital y las ciudades colindantes.



Porcentaje de municipios de cada CCAA que ha perdido población desde 1998

Entre 1998 y 2018

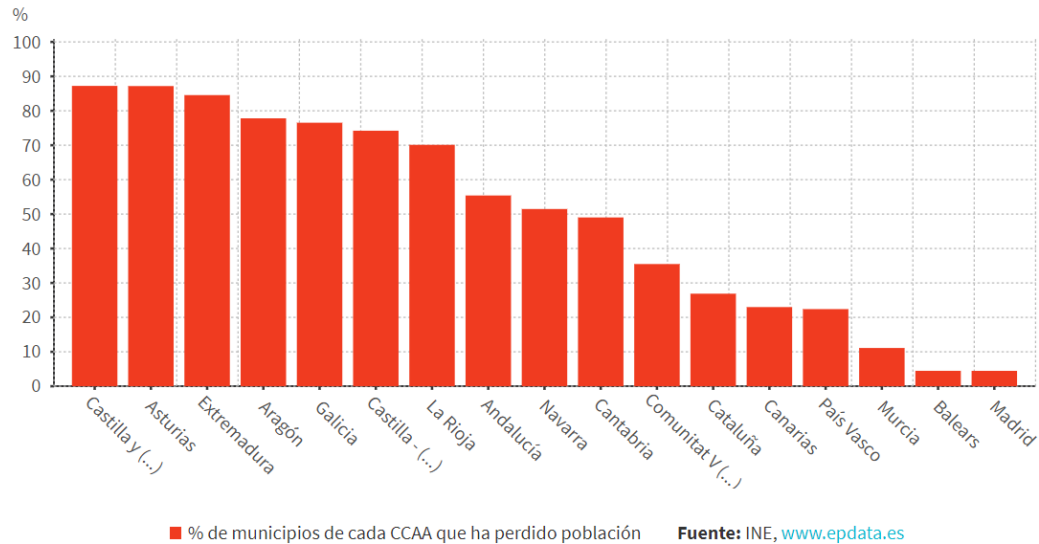


Ilustración I Perdida Población

Sin firmante (2017/2018). *La España vacía: despoblación en España, datos y estadísticas.*

La Datos sobre despoblación: ¿Qué regiones están más afectadas por la despoblación?

Recuperado de <https://www.epdata.es/datos/despoblacion-espana-datos-estadisticas/282>

En la era de la televisión por internet, los libros electrónicos y las películas en ‘streaming’, la radio también se ha adaptado a los nuevos hábitos de la población. Así nace el ‘podcast’ una palabra que aparece de la combinación de iPod y ‘broadcast’ (emisión) – es un archivo de sonido que se puede descargar de internet y reproducir en dispositivos como ordenadores, tabletas, móviles o iPods.

Los datos que tenemos a día de hoy, procedentes del EGM (Estudio General de Medios), dicen que “*consumen podcast el 1,6% de los oyentes globales de radio en España. Es decir, unos 360.000 oyentes (alrededor de un 20% de los que consumen radio por streaming a través de internet). Siempre según los datos del EGM, que es la única encuesta que se realiza en España para medir las audiencias de radio y que desde comienzos de 2017 incluye el consumo de podcast en sus mediciones*”.



De todos modos, se debe tener en cuenta que las cifras del EGM hacen referencia al consumo de podcast ligados a las emisoras de radio, por lo que el consumo real de podcast no radiofónicos es mayor. Tanto es así, que si se repasa la lista de los podcast más escuchados durante la semana en España según la plataforma ivoox.com, son varios entre los contenidos más consumidos que no tienen origen radiofónico.

Como se muestra en la imagen siguiente actualizada a 9/04/19, de los podcast más escuchados en la plataforma ivoox entre los 15 primeros puestos encontramos 7 relacionados con el misterio.

1	=		Nadie sabe nada, con Buenafuente y Romero Con Andreu Buenafuente y Berto Romero
2	=		Cuarto Milenio (Oficial) Cuarto Milenio es un programa de televisión, producido por Mediaset España, dirigido y presentado por los periodistas Iker Jiménez y Carmen Porter. Este programa se emite semanalmente en la cadena Cuatro, desde noviembre de 2005. El...
3	↑ +3		DIAS EXTRAÑOS con Santiago Camacho Vivimos en una época convulsa, el exceso de información genera desinformación y, a veces, las noticias más jugosas y significativas se quedan en el tintero de los grandes medios. La mayor diferencia entre nosotros y los medios de...
4	↓ -1		Humor en la Cadena SER El podcast con todo el humor de la Cadena SER
★			Promocionado ERA Magazine El podcast de la música independiente española, con entrevistas a grupos, discográficas, salas de conciertos, promotoras... Todos tienen cabida en este podcast que podréis escuchar desde la web, eramagazine.fm y desde las plataformas...
5	↓ -1		El Partidazo de COPE De lunes a viernes con Juanma Castaño de 23.30 a 1:30h el mejor deporte en COPE. Resume la actualidad deportiva, reuniendo a los protagonistas de la noticia.



6	-1		La Rosa de los Vientos Podcast de Historia, misterio, Investigación, relatos. Sábados 1:00 a 4:00 Domingos 1:30 a 04:00 con Bruno Cardeñosa y Silvia Casasola
7	=		Documentales Sonoros Este podcast está pensado para las personas como yo, que nuestros ojos ya no nos permiten ver lo que uno desea o le interesa y tienen curiosidad de aprender y conocer, también para aquellas que constantemente están fuera de casa por su...
8	=		Es la Mañana de Federico Información y opinión con Federico Jiménez Losantos y sus colaboradores habituales para arrancar con fuerza el día.
9	=		Espacio en Blanco 'Espacio en blanco' es el programa de misterio más legendario de la radio española. Comenzó su andadura en Radio Cadena Española en 1987. Durante 17 años ininterrumpidos, ha llenado las ondas de temas únicos que han abierto a la...
10	+4		La Vida Moderna El programa que ha cambiado las noches de la radio, con David Broncano, Ignatius Farray y Quequé
11	=		LA RESISTENCIA de David Broncano David Broncano se pone al frente de un nuevo programa diario en #0. Bajo el nombre de "La Resistencia" se trata del primer late-late de la televisión nacional y se emite en el canal exclusivo de Movistar+ justo después de "Late Motiv"....
12	-2		La Escómbula de la Brújula Historia, leyendas, misterios y lugares mágicos. TCon Jesus Callejo, Carlos Canales, Juan Ignacio Cuesta, David Sentinella, Marcos Carrasco y muchos más colaboradores.
13	-1		Misterios Canal que engloba podcasts del ámbito y temática del misterio: programas de radio, audiodocumentales, ... En facebook: https://www.facebook.com/pages/Canal-misterios-de-ivoox/507827489292233 Todo mi trabajo está realizado de manera...
14	-1		Momentos con Luis Rodríguez La noche, momentos para las confidencias de madrugada... Momentos, un programa de radio que presenta y dirige Luis Rodríguez en Cadena Cope Cataluña y Andorra, todos los días de 1,30 a 6 horas
★			Promocionado El Vuelo de Yorch Programa semanal musical temático de la parrilla de Jucal Radio de La Puebla de Alfindén, Zaragoza, y compartido con Rebote FM, de San Adrián, Navarra, Turia78.com de Paterna, Valencia, Rock FM de Costa Rica y con www.radioviajera.com
15	=		SER Historia La historia universal y sus más destacados acontecimientos son el hilo conductor de este programa presentado y dirigido por Nacho Ares. www.nachoares.com / www.serhistoria.es

Ilustración II Ranking Podcast Misterio

Ivoox.com. (2019). Ranking semanal de podcasts en iVoox

. https://www.ivoox.com/top100_hb.html



1.2 Análisis de tendencias

El disfrute de la experiencia es cada vez más importante. Incluso en las encuestas el 62% de usuarios señala que vivirlas es más importante que tener posesiones materiales.

En este contexto, a tenor de los resultados de su estudio, Booking.com indica que: *“en 2019 veremos viajeros disfrutando de experiencias auténticas y diferentes, incluso alojamientos con toques infantiles y divertidos, con piscinas de bolas o castillos hinchables para adultos de la generación millennial o Z, ya que 46% de los encuestados tiene pensado visitar un destino que les haga sentirse de nuevo como niños”*.

Sin firmante. (octubre 2018). Las tendencias de viajes que los españoles seguirán en 2019. Experiencias

Recuperado de https://www.hosteltur.com/109383_las-tendencias-de-viajes-que-los-epanoles-seguiran-en-2019.html



1.3 Competencia

1. Donde Viajamos

Es una plataforma que ofrece varios tipos de viajes o de rutas, pero en cuanto a misterio ofrece pocas opciones.

Además, las opciones que ofrece son las más conocidas.

2. Rutas Misteriosas

Ofrece rutas de misterio a nivel nacional con guías expertos en el mundo del misterio.

Pero ofrece muy pocas comunidades.

3. Gluubo

Integra un mapa de google con ubicaciones, pero solo por la zona de las comunidades catalana, valenciana y aragonesa.

La mayor competencia de este proyecto sería Gluubo porque es la idea que más se parece a Mistery Tracks Spain, aunque la más potente es la de rutas misteriosas porque es la más atractiva al ofrecer guías reconocidos en el mundo del misterio porque muchos de estos guías tienen libros, programas de radio de televisión etc....

Así que mezclando estas dos propuestas se obtendría el resultado ideal.



2. ESTADO DEL ARTE

El nicho de mercado en el que estará enfocado este proyecto será en el sector del turismo, más concretamente en el turismo alternativo como puede ser el turismo de experiencias u oscuro.

2.1 Estudio

El turismo ha sido uno de los fenómenos que más ha evolucionado especialmente en los últimos tiempos, han ido apareciendo nuevas tipologías, algunas de ellas tan diferentes como lo es el “*turismo oscuro*”. El turismo oscuro es un concepto relativamente nuevo, planteado por primera vez en 1996 por John Lennon y Malcolm Foley para poner nombre a un tipo de turismo cada vez más creciente y con una oferta muy dispersa a lo largo de todo el mundo.

Este tipo de turismo engloba diferentes lugares tales como: Cárceles, campos de concentración, lugares que han sufrido catástrofes naturales, escenarios de guerra, cementerios, edificios históricos y otros lugares que recogen dichos infortunios.

El crecimiento del “*turismo oscuro*” cada vez está siendo más notorio y Sharpley y Stone (1996) hablan de él como:

“Igualmente, no puede haber duda de que, en el último medio siglo, y en consonancia con el notable crecimiento en turismo, el Turismo Oscuro se ha generalizado y diversificado. En cuanto a la oferta, se ha producido un rápido crecimiento en la prestación de atracciones o experiencias... Por otra parte, el Turismo oscuro se ha vuelto ampliamente reconocido tanto como forma de Turismo, como herramienta de promoción...”

Traducción sacada de un artículo de IDTR (Insitute For Dark Turism Research).

[https://www.uclan.ac.uk/research/explore/groups/institute for dark tourism research.php](https://www.uclan.ac.uk/research/explore/groups/institute%20for%20dark%20tourism%20research.php)

Gráfica que muestra el crecimiento de visitantes al campo de concentración de Auschwitz situado en Polonia, y conocido por las muertes de judíos a manos de los nazis que se produjeron en la II Guerra Mundial.



Evolución número visitantes en Auschwitz desde 1946

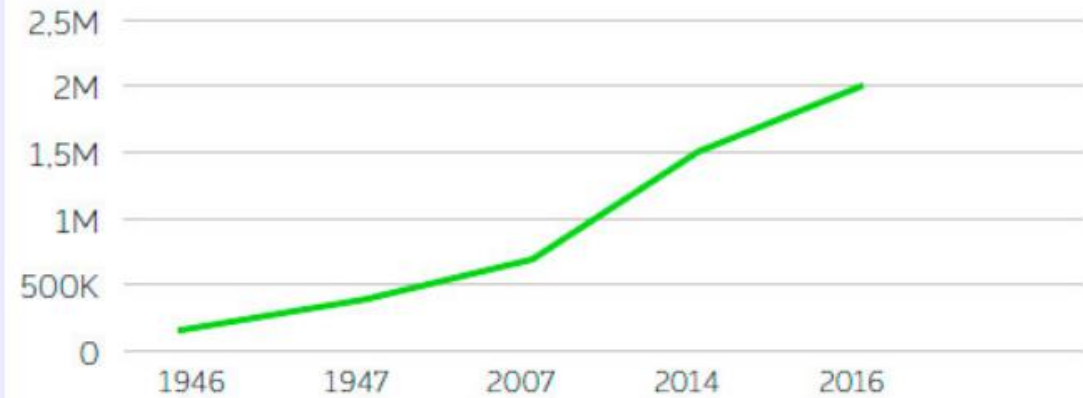


Ilustración III Evolución visitantes Auschwitz

Por poner otro ejemplo en la gráfica siguiente podemos ver el crecimiento de turistas en la ciudad de Medellín (Colombia) gracias a que ofrecen visitas ligadas al famoso narcotraficante Pablo Escobar.

Llegada de visitantes a Medellín desde 2008



Ilustración IV Llegada visitantes a Medellín

AEPT. (octubre 2017). **Turismo oscuro: la demanda de nuevas emociones**

. **Acontecimientos trágicos y el cine**

Recuperado de <http://www.aept.org/informe-ostealea-turismo-oscuro>



HALLOWEEN, impulsando el turismo oscuro.

La Fiesta de Halloween originaria de Estados Unidos, en España, no sólo ha aumentado el gasto en ocio, sino que también registra un significativo aumento en la promoción de destinos relacionados con turismo oscuro en numerosos blogs de viajes presentes en Internet y también en las agencias de viajes tradicionales.

AEPT. (octubre 2017). **Turismo oscuro: la demanda de nuevas emociones**

. **Halloween, motor del turismo negro**

Recuperado de <http://www.aept.org/informe-ostelea-turismo-oscuro>

El turismo rural

Francisco Mestre, fundador de Pueblos más Bonitos de España, cree que el *"turismo es una de las soluciones que puede evitar la despoblación"*.

Los objetivos de su organización son la promoción del patrimonio natural, cultural y rural para incrementar el turismo. Es el caso del municipio alcarreño de Hita, que pertenece a dicha organización, y que ha registrado un crecimiento de 83,7% en número de visitantes. Provocando un aumento del empleo en los sectores de la hostelería y restauración.

el Economista Yago Gantes(noviembre 2018). **La despoblación rural, un problema que tienen las ciudades**

El turismo como forma de vida

. **Halloween, motor del turismo negro**

Recuperado de <https://www.eleconomista.es/evasion/noticias/9483314/10/18/La-despoblacion-rural-un-problema-que-tienen-las-ciudades.html>

La despoblación

Como dato positivo a la despoblación rural tenemos la inmigración de ciudadanos de otros países que han visto en la despoblación de zonas rurales la oportunidad para labrarse una vida en España o de extranjeros que se asientan en este tipo de pueblos buscando tranquilidad. Esta llegada de migrantes ha cambiado la vida en muchos pueblos españoles hasta el punto de que en algunos municipios en 2018 ya había registrados más extranjeros que españoles.



En estos municipios hay empadronados más extranjeros que españoles

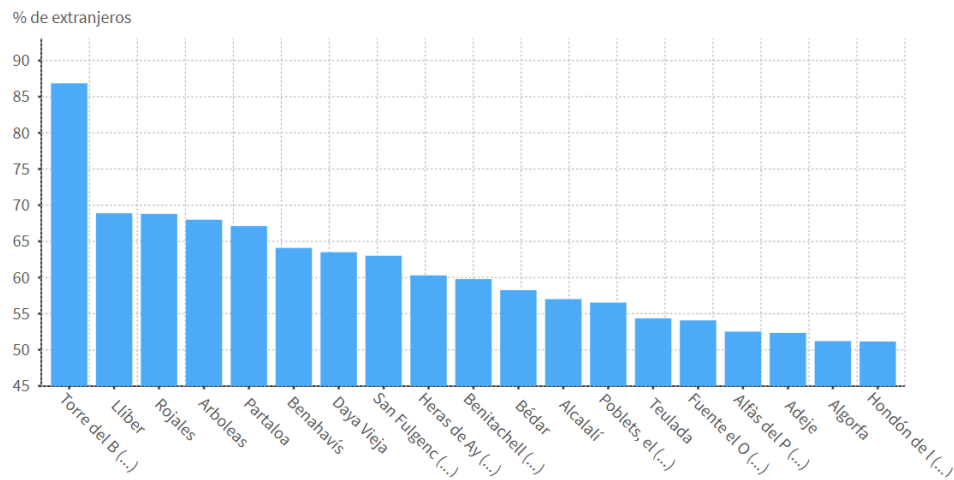


Ilustración V Municipios con más extranjeros que españoles

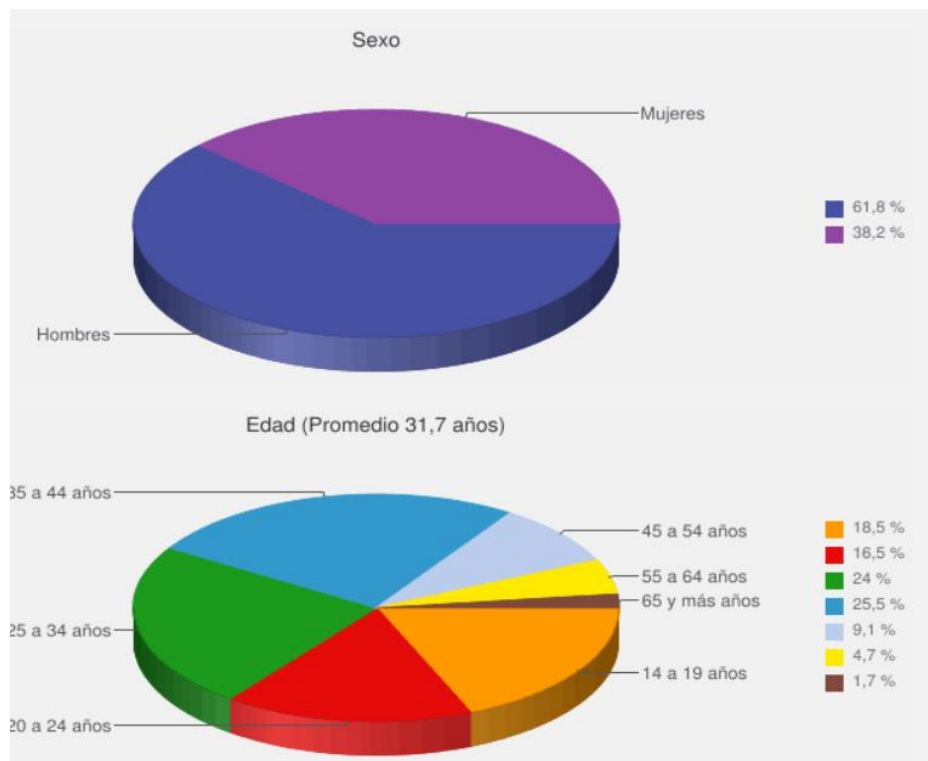
Sin firmante (2017/2018). *La España vacía: despoblación en España, datos y estadísticas.*

La Datos sobre despoblación: ¿Qué regiones están más afectadas por la despoblación?

Recuperado de <https://www.epdata.es/datos/despoblacion-espana-datos-estadisticas/282>

Perfil del consumidor de misterio

Según un antiguo estudio realizado por el escritor Javier Sierra cuando dirigía la revista MÁS ALLÁ las características de sus lectores en España se resumieron en los siguientes datos:



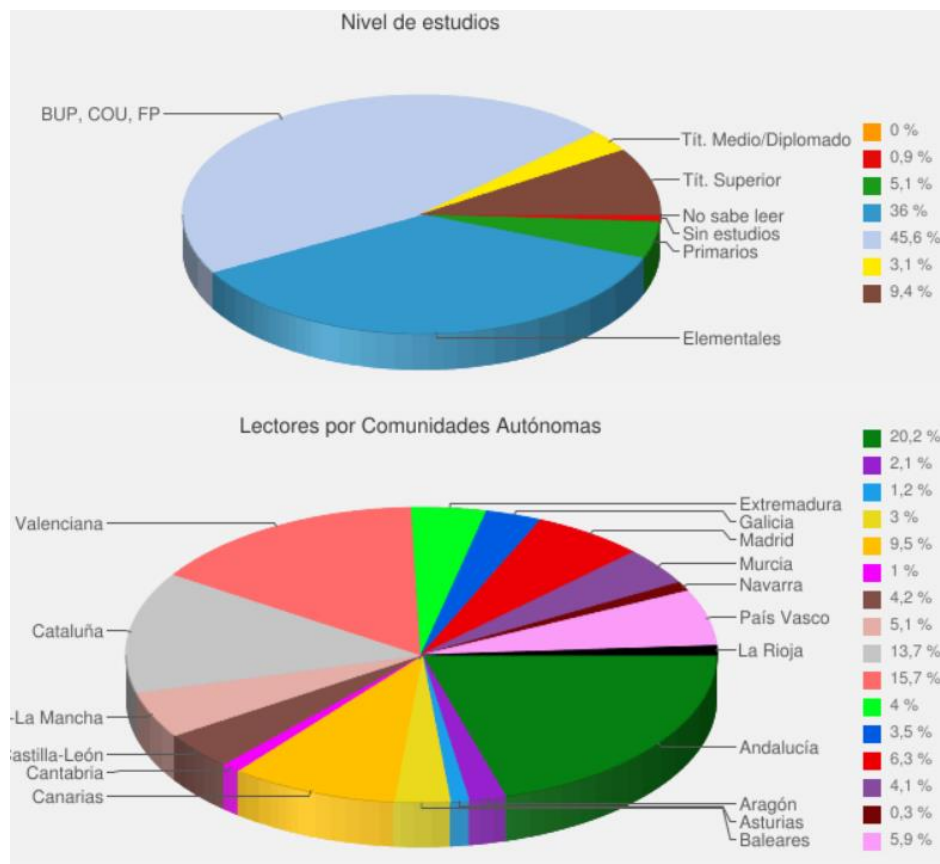


Ilustración VI Estadísticas lectores revista más allá

Javier Sierra (1999). Título del artículo. Perfil del lector Mas Alla

Recuperado de <https://www.javiersierra.com/biografia/periodismo/mas-alla/>



2.2 Conclusiones

- Los blogs de viajes, son un indicador de que el turismo oscuro es tendencia
- La denominación “turismo oscuro” comprende prácticas y actividades muy diferentes. Como por ejemplo el turismo de patrimonio, el turismo educativo, el turismo cinematográfico o el turismo bélico.
- El éxito turístico de los sitios “oscuros” se vincula a la capacidad de las infraestructuras turísticas de crear entornos emocionales que brinden al turista una experiencia única.
- El turismo oscuro, pese a ser un nicho de mercado en pleno desarrollo presenta un problema de comercialización debido a que quedan resistencias de tipo ético en la mercantilización de algunos lugares relacionados con acontecimientos trágicos.
- Promocionar la función educativa de estos lugares puede facilitar la comercialización de muchos sitios de turismo oscuro.
- El turismo oscuro es un factor de regeneración de territorios y destinos afectados por catástrofes.
- El rol activo de las comunidades locales en el nuevo relato de destinos que han sufrido catástrofes naturales o provocadas por el hombre es una condición imprescindible para asegurar una regeneración económica, social y cultural de los territorios.



3 INNOVACIÓN

3.1 Aportación

Lo que se pretende aportar con este proyecto es integrar ideas que ya existen porque se considera que lo que ya hay está incompleto. Actualmente ya existen webs dedicadas al misterio, pero no ofrecen la posibilidad de trazar rutas, o si lo permiten hay muy pocas posibilidades para escoger.

También existen páginas dedicadas a viajes en general que ofrecen un apartado de misterio, pero con pocas opciones o los lugares más reconocidos.

Con Mystery Tracks Spain se pretende resaltar el tema paranormal por encima del resto para darlo a conocer y que pueda dejar de ser un tema tabú.

Al proveer de un foro también se fomenta la comunicación y participación de los usuarios interesados en el misterio.

3.2 Soluciones Innovadoras Aplicadas

Como innovaciones se van a ofrecer:

- Mapa nacional completo con los lugares relacionados con el misterio
- Sistema de puntuaciones de sitios y usuarios para que puedan añadir nuevos lugares a la base de datos de la página.
- Foro de discusión para conocer gente con las mismas preocupaciones.
- Enlaces de interés: Podcast, videos, noticias....
- Calendario de eventos para poder consultar visitas guiadas, charlas, etc...



4. REQUISITOS

Los servicios que va a proporcionar esta herramienta son:

4.1 Requisitos Funcionales

Nº de requisito	RF01
Nombre de requisito	Tipos de usuario
Tipo	Restricción
Objetivo	OBJ1
Descripción	Habrà 3 tipos de usuarios: • <u>Anónimo</u>: solo podrán acceder al mapa y a realizar búsquedas. • <u>Registrado</u>: además de las mismas opciones que un usuario anónimo podrá acceder a los foros y subir rutas. Si tiene una puntuación mínima de 4 sobre 5 podrá subir contenidos sin supervisión. • <u>Administrador</u>: para el desarrollador de la página y su correcto funcionamiento.
Prioridad	Alta/Esencial
Nº de requisito	RF02
Nombre de requisito	Login
Tipo	Restricción
Objetivo	OBJ1
Descripción	La herramienta contará con un sistema de login que permita acceder a los usuarios registrados.
Prioridad	Alta/Esencial
Nº de requisito	RF03
Nombre de requisito	Acceso a recursos
Tipo	Funcional
Objetivo	OBJ2
Descripción	La herramienta tendrá una vista principal con un menú que permita el acceso a todos los recursos
Prioridad	Alta/Esencial
Nº de requisito	RF04
Nombre de requisito	Mapa de navegación
Tipo	Funcional
Objetivo	OBJ3
Descripción	La herramienta tendrá un mapa de navegación
Prioridad	Alta/Esencial
Nº de requisito	RF05
Nombre de requisito	Puntuaciones



Tipo	Funcional
Objetivo	OBJ4 OBJ1
Descripción	Se puntuarán los lugares subidos por los usuarios para evitar fraudes. Los usuarios con puntuación alta podrán añadir contenidos sin supervisión.
Prioridad	Baja/Opcional
Nº de requisito	RF06
Nombre de requisito	Enlaces relacionados
Tipo	Funcional
Objetivo	OBJ5
Descripción	La herramienta contará con un apartado para enlaces relacionados (Noticias, podcast, video.....)
Prioridad	Baja/Opcional
Nº de requisito	RF07
Nombre de requisito	Calendario
Tipo	Funcional
Objetivo	OBJ6
Descripción	La herramienta contará con un apartado para calendario de eventos.
Prioridad	Baja/Opcional
Nº de requisito	RF08
Nombre de requisito	Reservas
Tipo	Funcional
Objetivo	OBJ7
Descripción	La herramienta contará con un servicio de enlaces a compras y reservas.
Prioridad	Media/Deseado
Nº de requisito	RF09
Nombre de requisito	Añadir contenidos
Tipo	Funcional
Objetivo	OBJ8
Descripción	La página tendrá una vista que permita añadir contenidos los usuarios registrados.
Prioridad	Media/Deseado
Nº de requisito	RF10
Nombre de requisito	Foro
Tipo	Funcional
Objetivo	OBJ9
Descripción	Habrà un foro para que los usuarios puedan interactuar.
Prioridad	Alta/Esencial
Nº de requisito	RF11
Nombre de requisito	Cambio Contraseña
Tipo	Funcional
Objetivo	OBJ10



Descripción	Habrá posibilidad de cambiar la contraseña para los usuarios registrados
Prioridad	Alta/Esencial
Nº de requisito	RF12
Nombre de requisito	Perfil de usuario
Tipo	Funcional
Objetivo	OBJ11
Descripción	El usuario será capaz de consultar los datos de su perfil
Prioridad	Media/Deseado

4.2 Requisitos No Funcionales

Nº Requisito	Requisito
RNF01	El desarrollo se hará con el IDE Eclipse
RNF02	El servidor de aplicaciones será tomcat y estará integrado en Eclipse
RNF03	Se utilizará el framework Spring integrado en el IDE
RNF04	La Base de datos será H2 y se accederá a ella mediante Hibernate
RNF05	El desarrollo MVC de la aplicación se hará con J2EE
RNF06	El front-end se desarrollará con css,html y jsp
RNF07	Se aplicará la Ley Orgánica de Protección de Datos (LOPD)
RNF08	Se aplicará la Ley de Servicios de la Sociedad de la Información y Comercio Electrónico (LSSICE)



5. DISEÑO

La estructura que va a seguir esta herramienta es:

5. Colores

Para el **fondo**, necesitamos tonos claros como blanco, para que los textos sean legibles.

FFFFFFFF

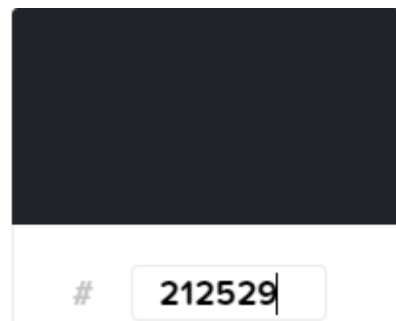
Para el menú, botones y otros elementos que se querían destacar como los artículos se ha elegido una gama de naranja.

Porque representa entusiasmo, creatividad y ánimo y se pretende eso precisamente que la gente pierda el miedo a contar sus experiencias.



Para el calendario y la cabecera se ha escogido tonalidades negras y grises

Porque representa misterio, sobriedad y formalidad.





5.2 Diagrama lógico de red

Los clientes se conectarán a la aplicación a través de un servicio de DynDNS proporcionado por noip.com la dirección en cuestión será www.mysterytracksspain.ddns.net

Hostname	IP/URL
ddns.net	
mysterytracksspain.ddns.net	90.77.137.191

Dicho servicio se configurará en el router propio y este se encargará de hacer el reenvío de puertos hasta la dirección del servidor.

En la imagen se muestran 2 mapeos uno para http y otro para https.

No.	LAN IP	Tipo de Protocolo	Puerto en la LAN	Puerto público
1	192.168.1.20	TCP	8080	80
2	192.168.1.20	TCP	8081	443

La estructura de red será la siguiente:

- Un router mikrotik que hará las labores de enrutador y de firewall. Tendrá como IP la 192.168.1.1 y repartirá IPs de la 100 a la 254. El resto estarán reservadas para los elementos de la red y las máquinas virtuales. El router también hará las veces de servidor DDNS para los accesos a los servidores web y de VPN en caso de querer conectarse desde fuera de la red local.
- Un rack en el que se alojarán y conectarán el resto de los equipos que conforman la red.
 - Un switch conectado al router principal para repartir la salida a internet.
 - Un servidor citrix que alojará las máquinas virtuales con IP estática 192.168.1.10/24.
 - Máquina Virtual sobre la que se desplegará la aplicación con IP estática 192.168.1.20/24. El servidor de aplicaciones será tomcat y estará integrado en el IDE Eclipse. Se utilizará el lenguaje Java J2EE y se desarrollará mediante el framework Spring, el acceso a datos será con Hibernate a una BBDD H2 embebida en el proyecto.



- NAS dedicado a realizar copias de seguridad de las MVs con IP estática 192.168.1.30/24
- Ordenador portátil conectado por DHCP en el que a través de la red se hará el desarrollo y la gestión tanto de la aplicación web como del servidor citrix.

Diagrama con el resultado final:

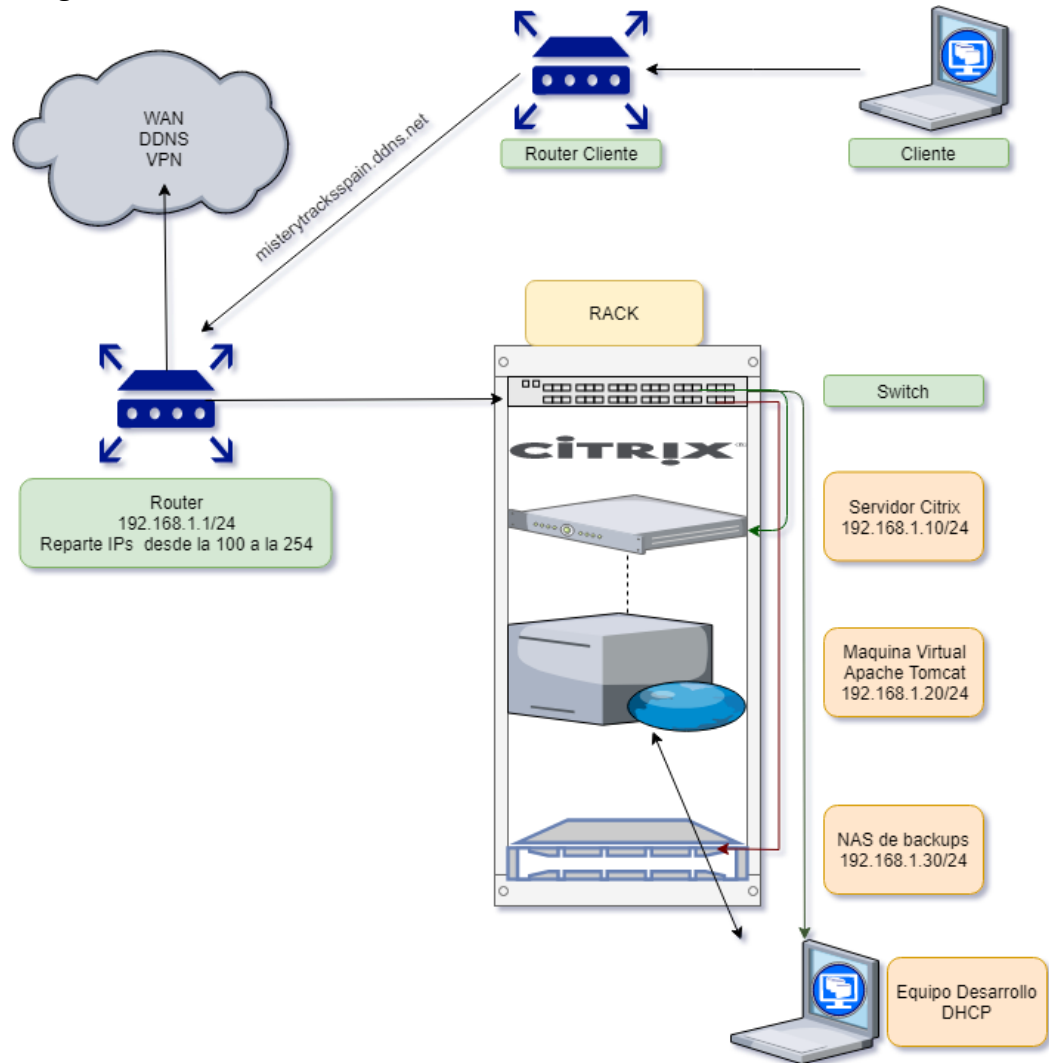


Ilustración VII Diagrama red



5.3 Diagrama Entidad / Relación

Para cubrir las necesidades de la aplicación y su correcto funcionamiento el sistema tendrá una base de datos que permita la siguiente funcionalidad:

- Tener un registro de los lugares con misterio, de cada lugar se quiere saber el identificador, nombre, la ciudad en la que se encuentra, una descripción de lo que sucede en ese lugar y la comunidad autónoma a la que pertenece.
- Guardar los datos de los usuarios que se registren, de estos se quiere saber el nombre, usuario, contraseña, email y fecha de nacimiento.
- Cada usuario podrá recibir votos además de poder votar lugares, usuarios, y rutas.
También tienen la posibilidad de escribir mensajes en el foro.
- De los mensajes interesa saber el tema al que pertenece, identificador, usuario que lo escribió, la fecha y hora en la que ha sido escrito.
El mensaje solo puede pertenecer a un tema.
- Cada ruta será creada por un usuario, de estas se conocerá el usuario que la subió, y los lugares que la conforman.
- En las comunidades autónomas se sabrá el nombre, código numérico y código de 3 caracteres.
- Los temas tendrán un identificador, un nombre y una descripción.

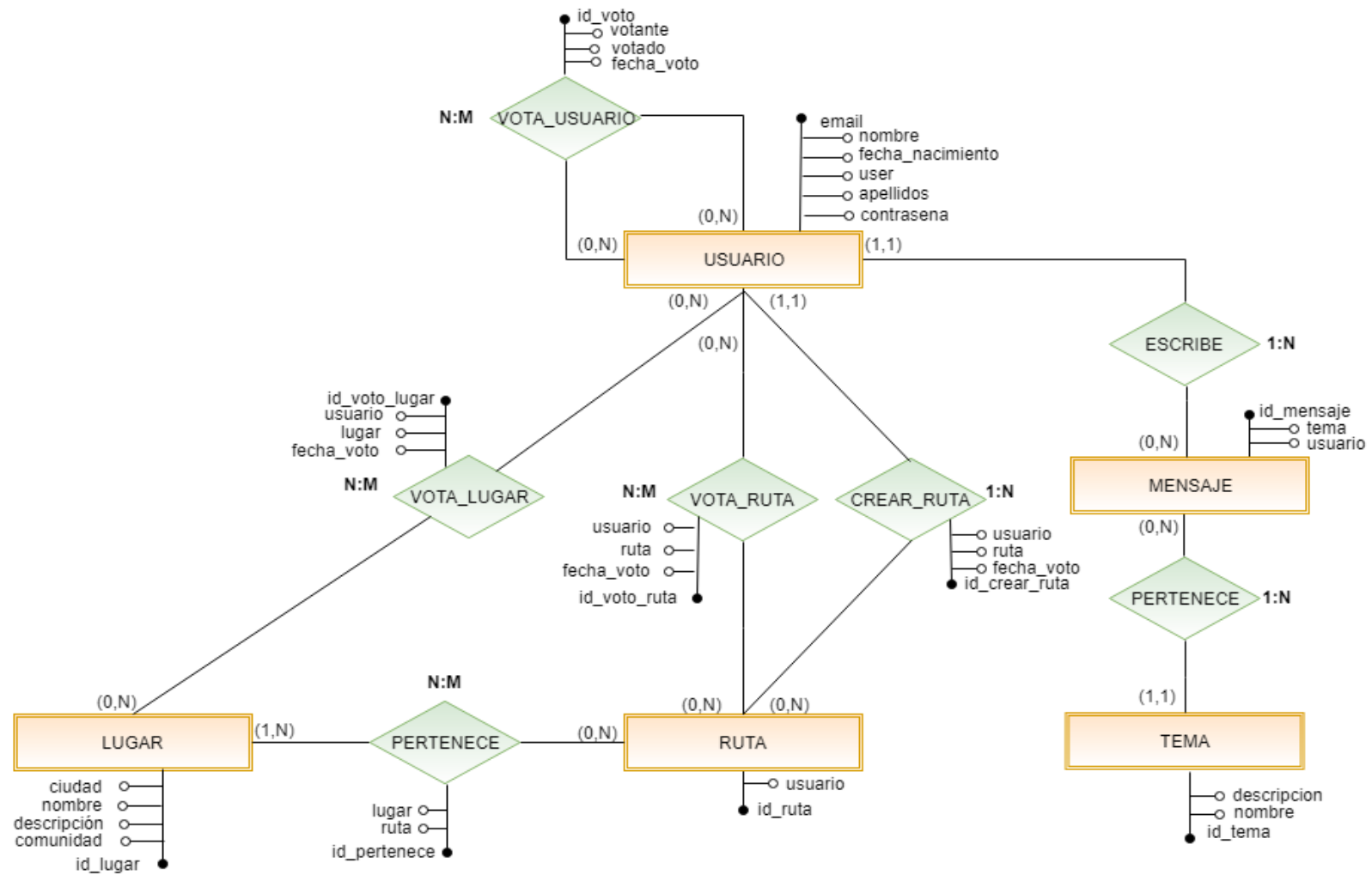


Ilustración VIII Diagrama E/R



5.3 Diagrama relacional

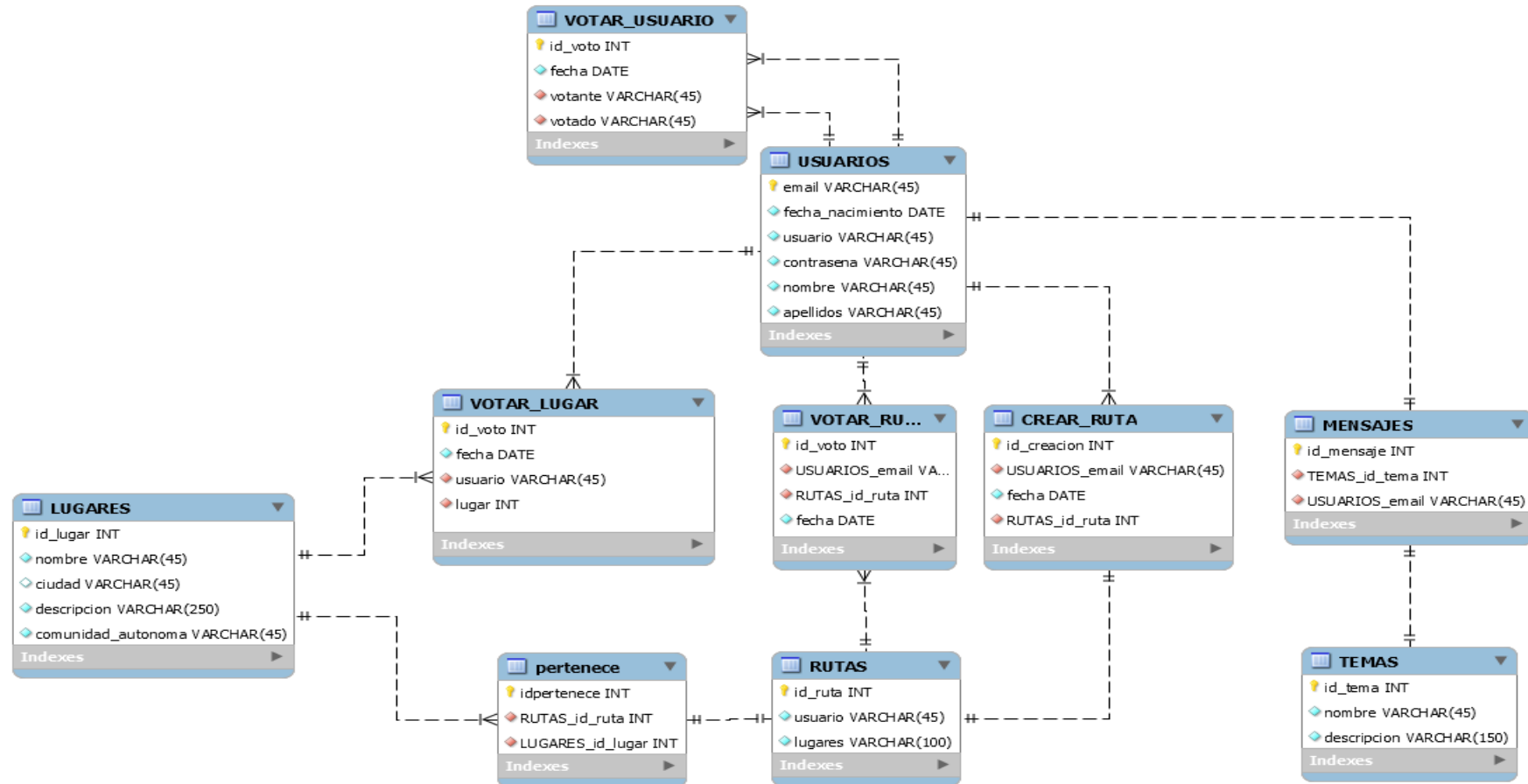


Ilustración IX Diagrama relacional



5.4 Diagrama de clases

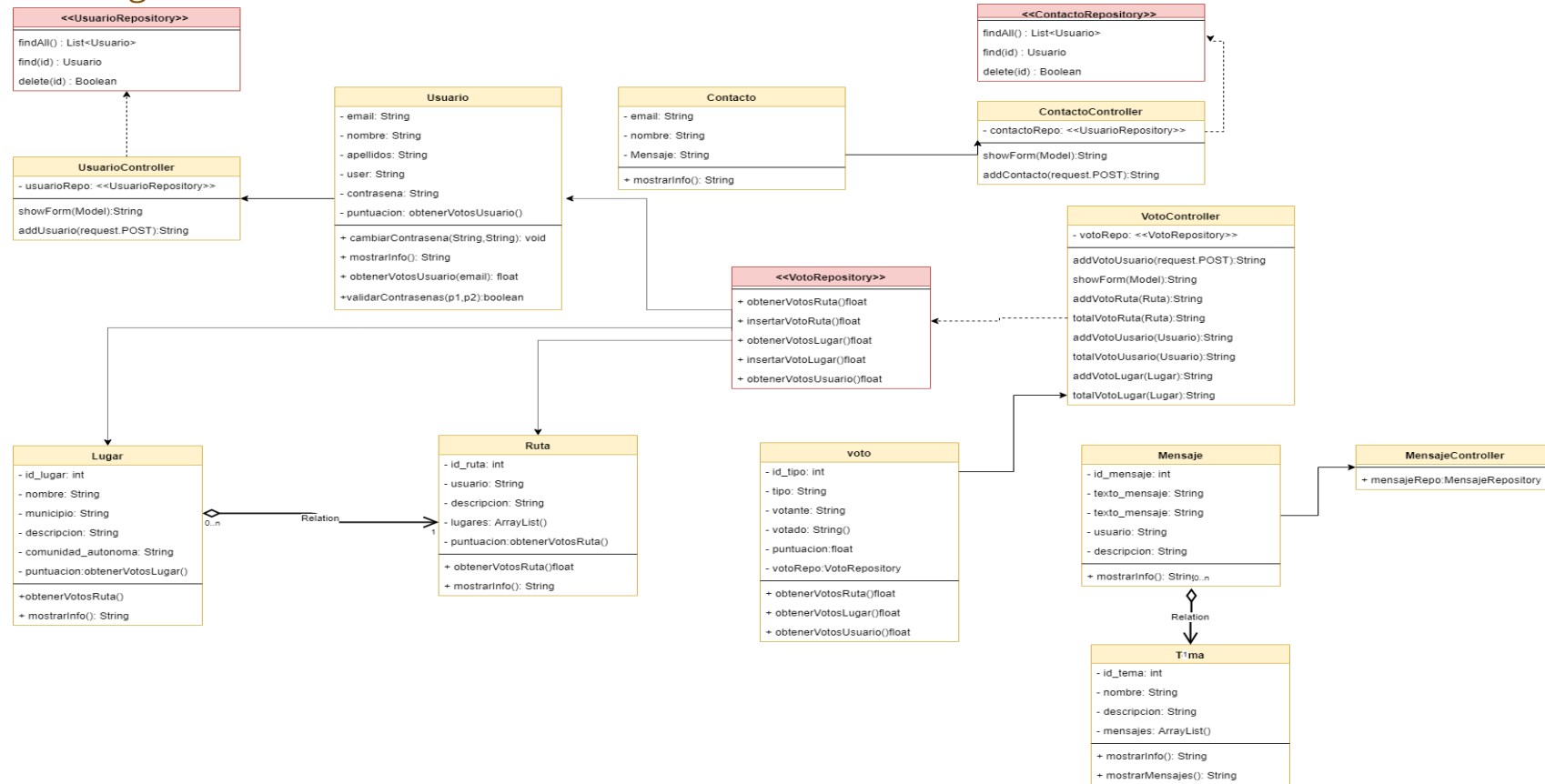


Ilustración X Diagrama de clases



A continuación, se explican más detalladamente las clases e interfaces, ya que no se han puesto todas en el diagrama porque las clases Controller son casi todas iguales

Y las interfaces Repository también, estas lo que hacen de intermediario entre la clase y su tabla en base de datos pudiendo así consultar e insertar datos.

Clase	Usuario
Descripción	Clase utilizada para guardar los datos de los usuarios registrados en la aplicación.
Atributos	• <u>email</u>: Atributo de tipo String que almacena el correo del <i>Usuario</i>. • <u>nombre</u>: Atributo de tipo String que almacena el nombre real del <i>Usuario</i>. • <u>apellidos</u>: Atributo de tipo String que almacena los apellidos del <i>Usuario</i>. • <u>contraseña</u>: Atributo de tipo String que almacena la contraseña del <i>Usuario</i>. • <u>contraseña2</u>: Atributo de tipo String que almacena la contraseña del <i>Usuario</i>. • <u>Puntuación</u>: Atributo de tipo float obtenido del retorno del método obtenerVotosUsuario()
Métodos	• <u>cambiarContrasena (String, String)</u>: método que recibe 2 parámetros que serán contraseñaNueva y contraseñaVieja y los insertará en la base de datos. • <u>obtenerVotosUsuario (email)</u>: método que sobrescribe el método de la interfaz y saca la media de puntos obtenidos por el <i>Usuario</i> que coincida con el email introducido. • <u>mostrarInfo ()</u>: devuelve todos los datos del <i>Usuario</i>.
Relación	Se relaciona con la clase <i>UsuarioController</i> Implementa las interfaces siguientes: • <<usuarioRepository>>



Clase	Lugar
Descripción	Clase utilizada para guardar los datos de los <i>Lugares</i> que saldrán en el mapa.
Atributos	• <u>id</u>: Atributo de tipo Integer que almacena el identificador del <i>Lugar</i>. • <u>descripción</u>: Atributo de tipo String que guarda la descripción del <i>Lugar</i>. • <u>municipio</u>: Atributo de tipo String que almacena el municipio en el que se encuentra el <i>Lugar</i>. • <u>comunidad autónoma</u>: Atributo de tipo String que almacena la comunidad autónoma en la que se encuentra el <i>Lugar</i>. • <u>latitud</u>: Atributo de tipo String que almacena la latitud en la que se encuentra el <i>Lugar</i>. • <u>longitud</u>: Atributo de tipo String que almacena la longitud en la que se encuentra el <i>Lugar</i>. • <u>puntuación</u>: Atributo de tipo float obtenido del retorno del método obtenerVotosLugar()
Métodos	• <u>obtenerVotosLugar (id lugar)</u>: método que sobrescribe el método de la interfaz y saca la media de puntos obtenidos por el <i>Lugar</i> que coincide con el id introducido. • <u>mostrarInfo ()</u>: devuelve todos los datos del <i>Lugar</i>.
Relación	Se relaciona con la clase <i>Ruta</i> a través de un ArrayList() de lugares y LugarController implementa las interfaces siguientes: • <<votoRepository>> • <<lugarRepository>>



Clase	Mensaje
Descripción	Clase utilizada para guardar los <i>Mensajes</i> escritos por los <i>Usuarios</i> .
Atributos	• <u>id mensaje</u>: Atributo de tipo Integer que almacena el identificador del <i>Mensaje</i>. • <u>descripción</u>: Atributo de tipo String que guarda la descripción del <i>Mensaje</i>. • <u>usuario</u>: Atributo de tipo String que almacena el identificador del usuario que escribió el <i>Mensaje</i>. • <u>texto mensaje</u>: Atributo de tipo String que guarda el texto del <i>Mensaje</i>.
Métodos	• <u>mostrarInfo ()</u>: devuelve todos los datos del <i>Mensaje</i>.
Relación	Se relaciona con la clase <i>Tema</i> a través de un ArrayList() de <i>Mensajes</i> . Y con <i>UsuarioController</i>

Clase	Ruta
Descripción	Clase utilizada para guardar los datos de las <i>Rutas</i> subidas por los <i>Usuarios</i> .
Atributos	• <u>id ruta</u>: Atributo de tipo Integer que almacena el identificador de la <i>Ruta</i>. • <u>descripción</u>: Atributo de tipo String que guarda la descripción de la <i>Ruta</i>. • <u>usuario</u>: Atributo de tipo String que almacena el identificador del usuario que subió la <i>Ruta</i>. • <u>lugares</u>: Atributo ArrayList () que guarda una lista de los lugares que componen la <i>Ruta</i>. • <u>puntuación</u>: Atributo de tipo float obtenido del retorno del método obtenerVotosRuta()
Métodos	• <u>obtenerVotosRuta (id ruta)</u>: método que sobrescribe el método de la interfaz y saca la media de puntos obtenidos por el <i>Lugar</i> que coincide con el id introducido. • <u>mostrarInfo ()</u>: devuelve todos los datos del <i>Lugar</i>.
Relación	Se relaciona con la clase <i>Lugar</i> a través de un ArrayList() de lugares y <i>LugarController</i> implementa las interfaces siguientes: • <<rutaRepository>> • <<votoRepository>>



Clase	Tema
Descripción	Clase utilizada para guardar los mensajes escritos por los usuarios.
Atributos	• <u>id tema</u>: Atributo de tipo Integer que almacena el identificador del <i>Tema</i>. • <u>descripción</u>: Atributo de tipo String que guarda la descripción del <i>Tema</i>. • <u>nombre</u>: Atributo de tipo String que almacena el nombre del <i>Tema</i>. • <u>mensajes</u>: Atributo de tipo ArrayList() que almacena los <i>Mensajes</i> correspondientes al <i>Tema</i>.
Métodos	• <u>mostrarInfo ()</u>: devuelve todos los datos del <i>Tema</i>. • <u>mostrarMensajes ()</u>: devuelve todos los <i>Mensajes</i> del <i>Tema</i>.
Relación	Se relaciona con la clase <i>Mensaje</i> a través de un ArrayList() de <i>Mensajes</i> . Y con <u><i>TemaController</i></u>

Clase	Contacto
Descripción	Clase utilizada para guardar los mensajes escritos por los usuarios en la pestaña de contacto.
Atributos	• <u>id</u>: Atributo de tipo Integer y autoincremental que almacena el identificador del <i>Contacto</i>. • <u>mensaje</u>: Atributo de tipo String que guarda el mensaje. • <u>email</u>: Atributo de tipo String que almacena el email de <i>Contacto</i>.
Métodos	• <u>mostrarInfo ()</u>: devuelve todos los datos del <i>Contacto</i>.
Relación	Se relaciona con la clase <i>ContactoController</i>



Clase	UsuarioController
Descripción	Clase utilizada para conectarse a la tabla mediante una interfaz y gestionar el intercambio de datos entre el front-end y el back-end .
Atributos	• <u>usuarioRepository</u> : instancia del tipo UsuarioRepository usada para el intercambio de datos con la BBDD funciona con la anotación @Autowired y Spring gestiona los datos
Métodos	• showForm(RequestMethod.GET) captura las peticiones GET que llegan a /registro , instancia un objeto Usuario y se lo manda al formulario • submit(Request.Method.POST) captura la action del formulario , hace las validaciones comprobando las anotaciones puestas en la clase y si no hay errores inserta en BBDD y si hay se los envía a la vista que los recoge con la etiqueta <form:errors>
Relación	Se relaciona con la clase <i>Usuario</i> y la interfaz <u>UsuarioRepository</u> .

Clase	<<interface>> usuarioRepository
Descripción	Interfaz utilizada para la interacción con la tabla USUARIOS .
Atributos	No tiene atributos, pero debe recibir objeto de tipo Usuario y el tipo de dato de la anotación @Id de la clase Usuario
Métodos	• no tiene métodos propios • utiliza de la clase que implementa JPARepository: • insert() • delete() • find(id) • findAll() • deleteAll() • etc...
Relación	Extiende la clase <i>JPARepository</i> La utiliza la clase <i>UsuarioController</i> .



Clase	ContactoController
Descripción	Clase utilizada para conectarse a la tabla mediante una interfaz y gestionar el intercambio de datos entre el front-end y el back-end .
Atributos	• <u>contactoRepository</u> : instancia del tipo ContactoRepository usada para el intercambio de datos con la BBDD funciona con la anotación @Autowired y Spring gestiona los datos
Métodos	• showForm(RequestMethod.GET) captura las peticiones GET que llegan a /contacto, instancia un objeto Contacto y se lo manda al formulario • submit(Request.Method.POST) captura la action /addContacto del formulario , hace las validaciones comprobando las anotaciones puestas en la clase y si no hay errores inserta en BBDD y si hay se los envía a la vista que las recoge con la etiqueta <form:errors>
Relación	Se relaciona con la clase <i>Usuario</i> y la interfaz <u>UsuarioRepository</u> .

Clase	<<interface>> contactoRepository
Descripción	Interfaz utilizada para la interacción con la tabla CONTACTO .
Atributos	No tiene atributos, pero debe recibir objeto de tipo <i>Contacto</i> y el tipo de dato de la anotación @Id de la clase <i>Contacto</i> .
Métodos	• no tiene métodos propios • utiliza de la clase que implementa JPAREpository: • insert() • delete() • find(id) • findAll() • deleteAll() • etc...
Relación	Extiende la clase <i>JPARepository</i> La utiliza la clase <i>ContactoController</i> .



Clase	LugarController
Descripción	Clase utilizada para conectarse a la tabla mediante una interfaz y gestionar el intercambio de datos entre el front-end y el back-end .
Atributos	• <u>clugarRepository</u> : instancia del tipo LugarRepository usada para el intercambio de datos con la BBDD funciona con la anotación @Autowired y Spring gestiona los datos
Métodos	• showForm(RequestMethod.GET) captura las peticiones GET que llegan a /lugar, instancia un objeto <i>Lugar</i> y se lo manda al formulario en el que se hará la subida de lugares. • submit(Request.Method.POST) captura la action /addLugar del formulario , hace las validaciones comprobando las anotaciones puestas en la clase y si no hay errores inserta en BBDD y si hay se los envía a la vista que las recoge con la etiqueta <form:errors>
Relación	Se relaciona con la clase <i>Lugar</i> y la interfaz <u>LugarRepository</u> .

Clase	<<interface>> LugarRepository
Descripción	Interfaz utilizada para la interacción con la tabla Lugares .
Atributos	No tiene atributos, pero debe recibir objeto de tipo <i>Lugar</i> y el tipo de dato de la anotación @Id de la clase <i>Lugar</i> .
Métodos	• no tiene métodos propios • utiliza de la clase que implementa JpaRepository: • insert() • delete() • find(id) • findAll() • deleteAll() • etc...
Relación	Extiende la clase <i>JpaRepository</i> La utiliza la clase <i>LugarController</i> .



Clase	TemaController
Descripción	Clase utilizada para conectarse a la tabla mediante una interfaz y gestionar el intercambio de datos entre el front-end y el back-end .
Atributos	• <u>temaRepository</u> : instancia del tipo <i>TemaRepository</i> usada para el intercambio de datos con la BBDD funciona con la anotación <i>@Autowired</i> y Spring gestiona los datos.
Métodos	• <code>showForm(RequestMethod.GET)</code> captura las peticiones GET que llegan como <code>/tema</code>, instancia un objeto <i>Tema</i> y se lo manda al formulario.. • <code>submit(Request.Method.POST)</code> captura la action del formulario , hace las validaciones comprobando las anotaciones puestas en la clase y si no hay errores inserta en BBDD y si hay se los envía a la vista que las recoge con la etiqueta <code><form:errors></code>
Relación	Se relaciona con la clase <i>Tema</i> y la interfaz <u><i>TemaRepository</i></u> .

Clase	<code><<interface>></code> TemaRepository
Descripción	Interfaz utilizada para la interacción con la tabla Temas .
Atributos	No tiene atributos, pero debe recibir objeto de tipo <i>Tema</i> y el tipo de dato de la anotación <i>@Id</i> de la clase <i>Tema</i> .
Métodos	• no tiene métodos propios • utiliza de la clase que implementa <i>JpaRepository</i>: • <code>insert()</code> • <code>delete()</code> • <code>find(id)</code> • <code>findAll()</code> • <code>deleteAll()</code> • etc...
Relación	Extiende la clase <i>JpaRepository</i> La utiliza la clase <i>TemaController</i> .



Clase	<u>RutaController</u>
Descripción	Clase utilizada para conectarse a la tabla mediante una interfaz y gestionar el intercambio de datos entre el front-end y el back-end .
Atributos	• <u>rutaRepository</u> : instancia del tipo <i>RutaRepository</i> usada para el intercambio de datos con la BBDD funciona con la anotación <i>@Autowired</i> y Spring gestiona los datos.
Métodos	• <code>showForm(RequestMethod.GET)</code> captura las peticiones GET que llegan como /tema, instancia un objeto <i>Ruta</i> y se lo manda al formulario.. • <code>submit(Request.Method.POST)</code> captura la action /addRuta del formulario , hace las validaciones comprobando las anotaciones puestas en la clase y si no hay errores inserta en BBDD y si hay se los envía a la vista que las recoge con la etiqueta <code><form:errors></code>
Relación	Se relaciona con la clase <i>Ruta</i> y la interfaz <i>RutaRepository</i> .

Clase	<code><<interface>></code> <i>RutaRepository</i>
Descripción	Interfaz utilizada para la interacción con la tabla RUTAS .
Atributos	No tiene atributos, pero debe recibir objeto de tipo <i>Ruta</i> y el tipo de dato de la anotación <i>@Id</i> de la clase <i>Ruta</i> .
Métodos	• no tiene métodos propios • utiliza de la clase que implementa <i>JpaRepository</i>: • <code>insert()</code> • <code>delete()</code> • <code>find(id)</code> • <code>findAll()</code> • <code>deleteAll()</code> • etc...
Relación	Extiende la clase <i>JpaRepository</i> La utiliza la clase <i>RutaController</i> .



Clase	MensajeController
Descripción	Clase utilizada para conectarse a la tabla mediante una interfaz y gestionar el intercambio de datos entre el front-end y el back-end .
Atributos	• <u>mensajeRepository</u> : instancia del tipo <i>MensajeRepository</i> usada para el intercambio de datos con la BBDD funciona con la anotación <i>@Autowired</i> y Spring gestiona los datos.
Métodos	• showForm(RequestMethod.GET) captura las peticiones GET que llegan como /mensaje, instancia un objeto <i>Mensaje</i> y se lo manda al formulario. • submit(Request.Method.POST) captura la action addMensaje del formulario , hace las validaciones comprobando las anotaciones puestas en la clase y si no hay errores inserta en BBDD y si hay se los envía a la vista que las recoge con la etiqueta <form:errors>
Relación	Se relaciona con la clases <i>Tema</i> , <i>Mensaje</i> y la interfaz <u><i>MensajeRepository</i></u> .

Clase	<<interface>> <i>MensajeRepository</i>
Descripción	Interfaz utilizada para la interacción con la tabla MENSAJE .
Atributos	No tiene atributos, pero debe recibir objeto de tipo <i>Mensaje</i> y el tipo de dato de la anotación <i>@Id</i> de la clase <i>Mensaje</i> .
Métodos	• no tiene métodos propios • utiliza de la clase que implementa <i>JpaRepository</i>: • insert() • delete() • find(id) • findAll() • deleteAll() • etc...
Relación	Extiende la clase <i>JpaRepository</i> La utiliza la clase <u><i>MensajeController</i></u> .



Clase	VotosController
Descripción	Clase utilizada para conectarse a la tabla <i>Votos</i> mediante una interfaz y gestionar el intercambio de datos entre el front-end y el back-end .
Atributos	• <u>votosRepository</u> : instancia del tipo <i>VotosRepository</i> usada para el intercambio de datos con la BBDD funciona con la anotación <i>@Autowired</i> y Spring gestiona los datos.
Métodos	• showForm(RequestMethod.GET) captura las peticiones GET que llegan como /voto, instancia un objeto <i>Voto</i> y se lo manda al formulario. • submitVotoRuta(Request.Method.POST) captura la action addVotoRuta del formulario , hace las validaciones comprobando las anotaciones puestas en la clase y si no hay errores inserta en BBDD y si hay se los envía a la vista que las recoge con la etiqueta <form:errors> • submitVotoUsuario(Request.Method.POST) captura la action addVotoUsuario del formulario , hace las validaciones comprobando las anotaciones puestas en la clase y si no hay errores inserta en BBDD y si hay se los envía a la vista que las recoge con la etiqueta <form:errors> • submitVotoLugar(Request.Method.POST) captura la action addVotoLugar del formulario , hace las validaciones comprobando las anotaciones puestas en la clase y si no hay errores inserta en BBDD y si hay se los envía a la vista que las recoge con la etiqueta <form:errors>
Relación	Se relaciona con la clase <i>Votos</i> y la interfaz <i>VotosRepository</i> .

Clase	<<interface>> VotosRepository
Descripción	Interfaz utilizada para la interacción con la tabla VOTOS .
Atributos	No tiene atributos, pero debe recibir objeto de tipo <i>Voto</i> y el tipo de dato de la anotación <i>@Id</i> de la clase <i>Voto</i> .
Métodos	• insertVotoRuta(id_ruta,votante,puntos) • insertVotoUsuario(mail_votado,mail_votante,puntos) • insertVotoLugar(id_lugar,votante,puntos) • utiliza de la clase que implementa JpaRepository: • insert() • delete() • find(id) • findAll() • deleteAll() • etc...
Relación	Extiende la clase <i>JpaRepository</i> La utilizan las clases <i>VotosController</i> <i>UsuarioController</i> , <i>LugarController</i> y <i>RutaController</i>



5.5 Diagrama de interfaces

5.5.1 Mapa navegación

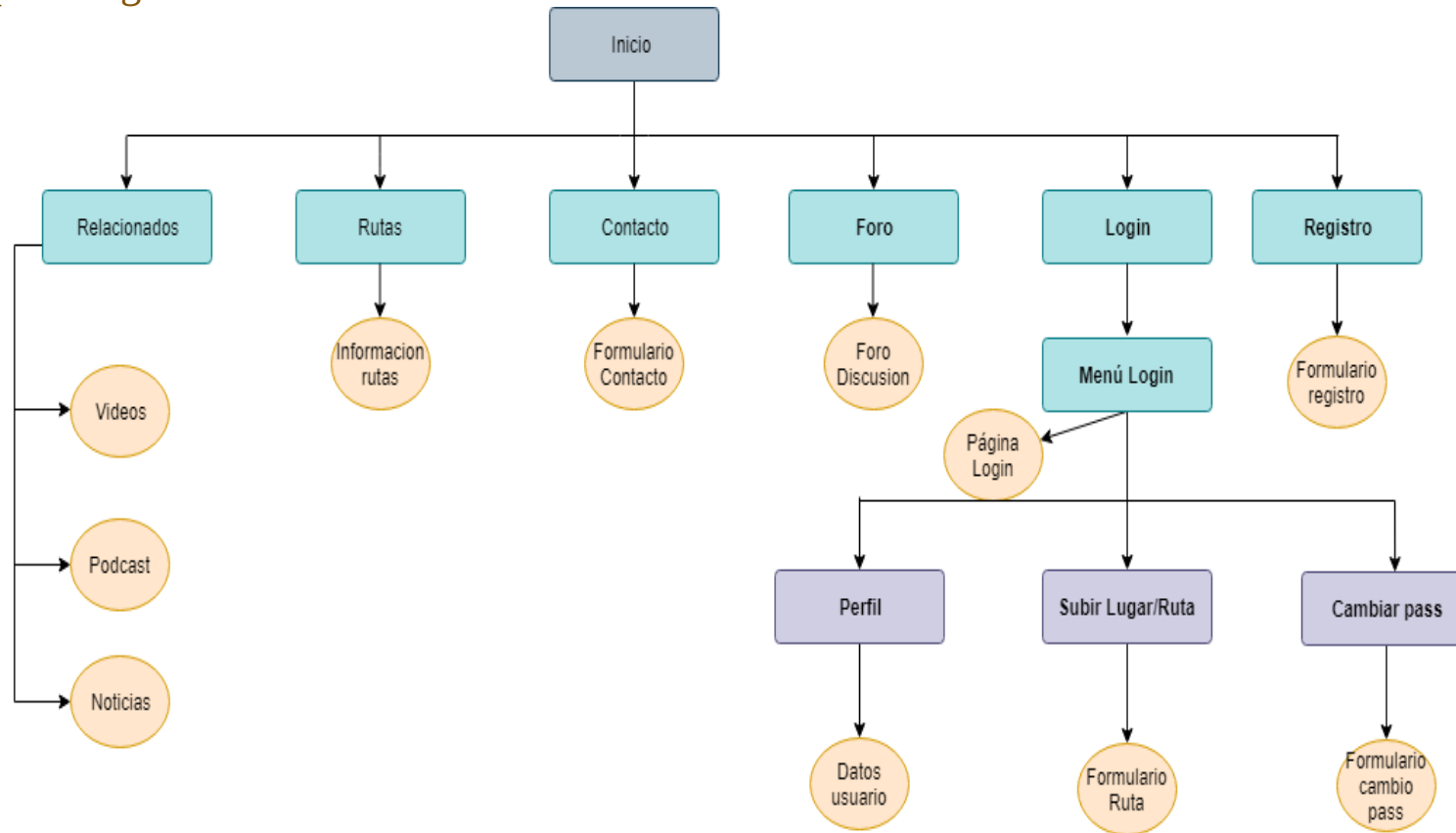


Ilustración XI Mapa navegación



5.5.2 Interfaces

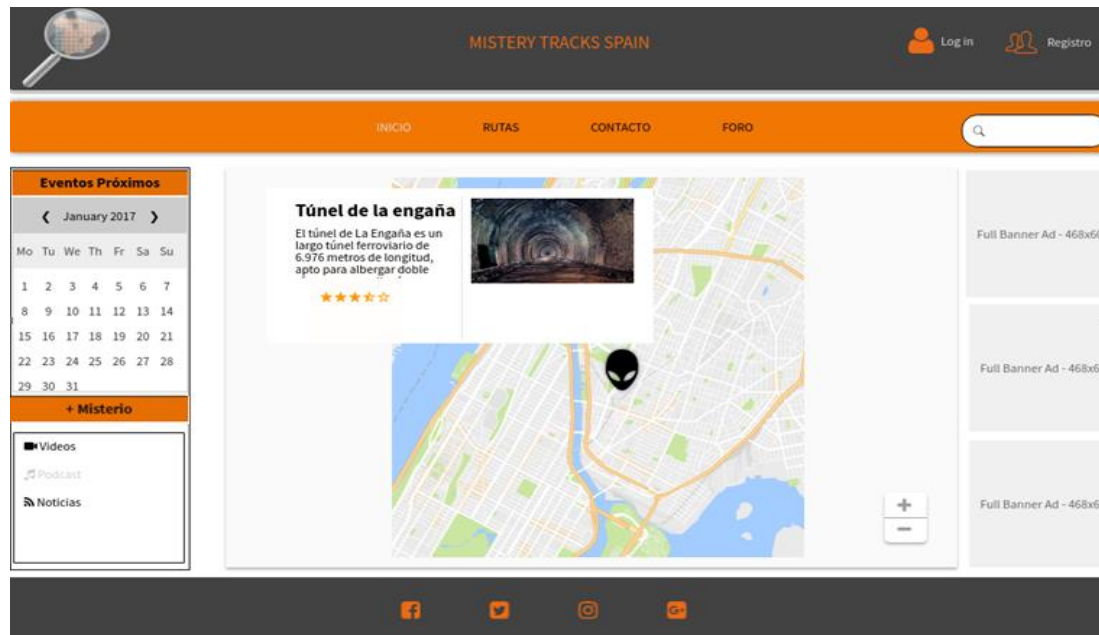
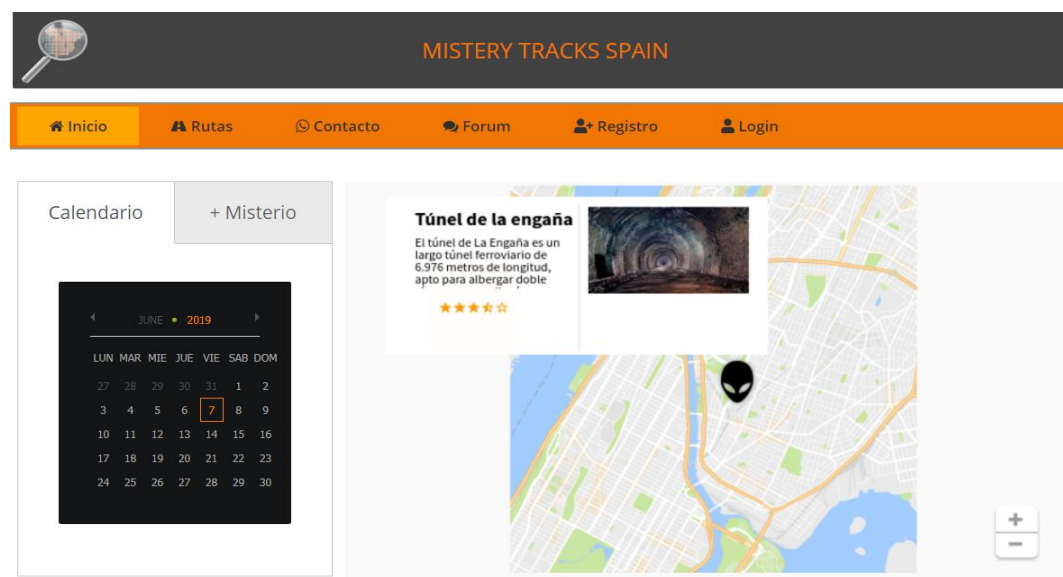


Ilustración XII Pantalla inicio INTF01 Prototipo



Copyright © 2019 Mystery Tracks - Design Iván Casado

Ilustración XIII Pantalla inicio INTF01 Resultado Final

Identificación	INTF01
Tipo	Salida
Formato	Impresa
Contenido	Pantalla principal desde la que se accede al resto de recursos de la herramienta. En ella se muestra el mapa y permite acceder a los enlaces relacionados.
Frecuencia	A petición del usuario
Requisitos	RF03 RF04

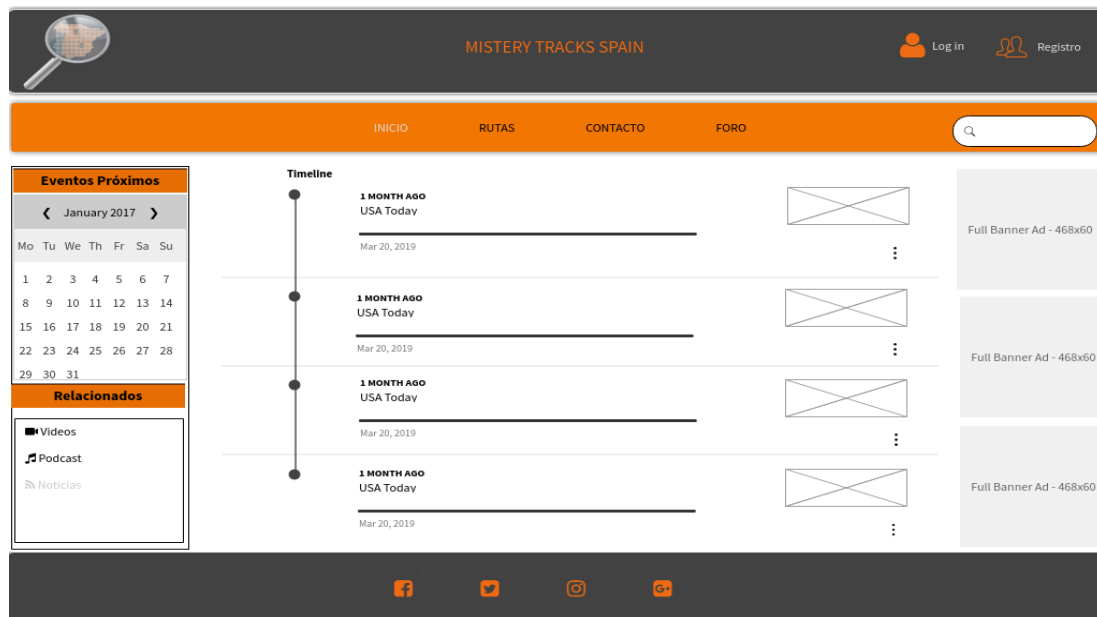


Ilustración XIV Pantalla Relacionados/Noticias INTF02 Prototipo

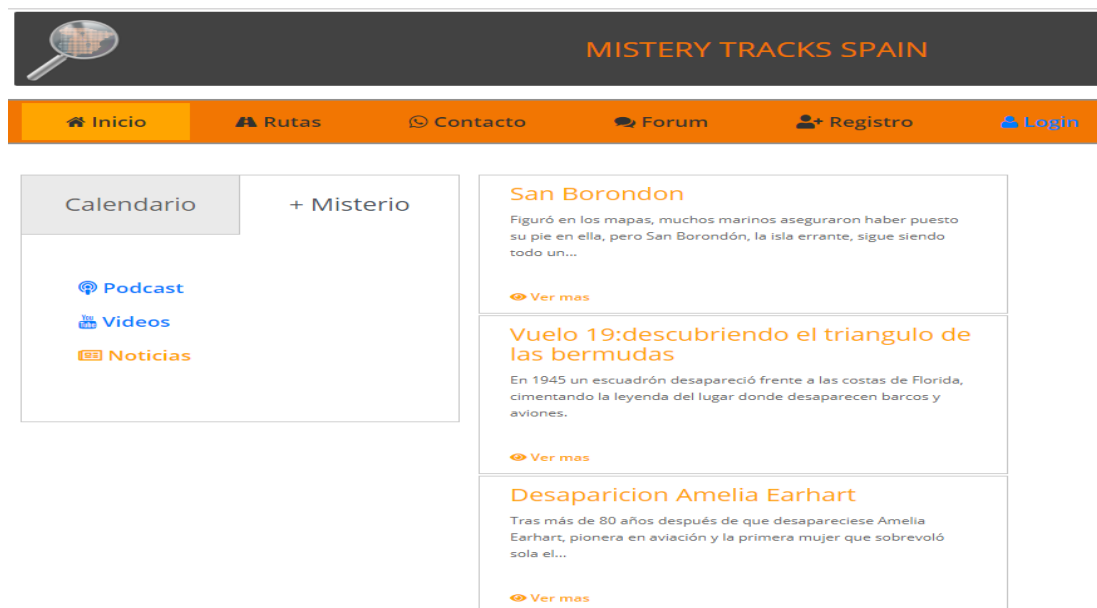


Ilustración XV Pantalla Relacionados/Noticias INTF02 Resultado Final

Identificación	INTF02
Tipo	Salida
Formato	Impresa
Contenido	Pantalla correspondiente a la sección de relacionados que muestra noticias de misterio.
Frecuencia	A petición del usuario
Requisitos	RF04 RF06

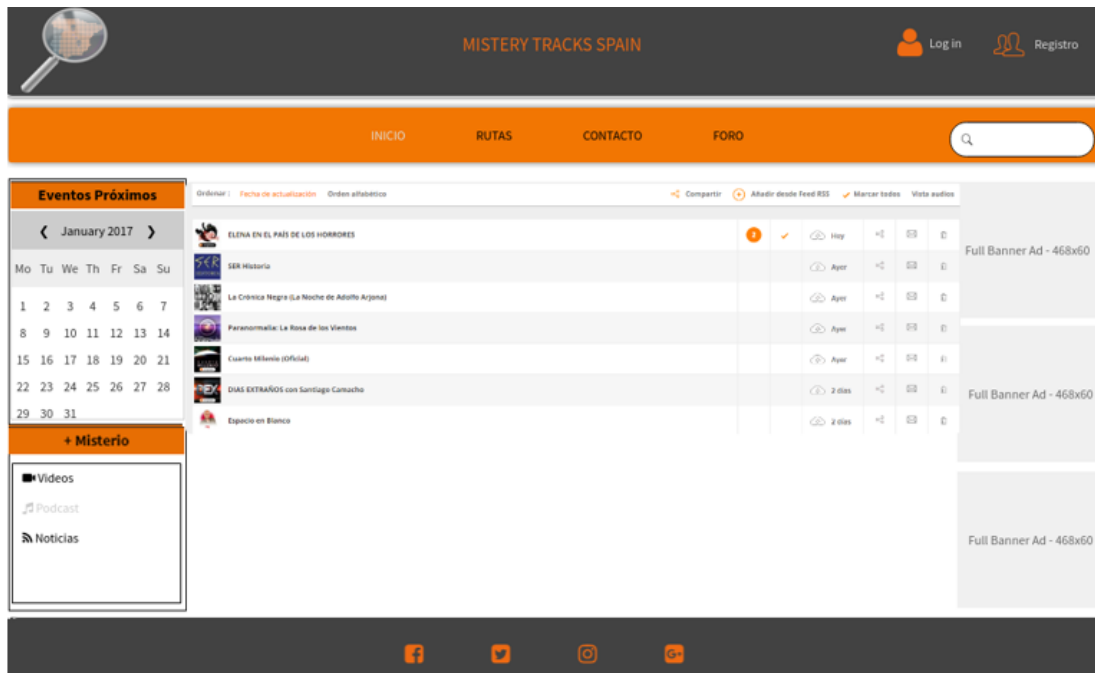


Ilustración XVI Pantalla Relacionados/Podcast INTF03 Prototipo

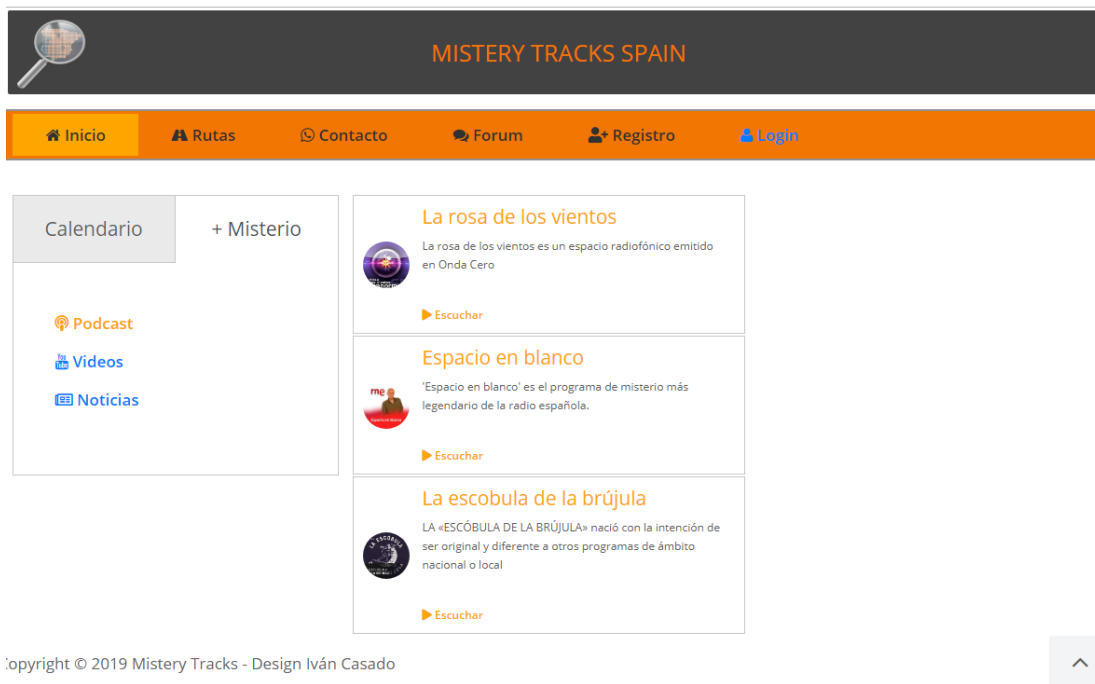


Ilustración XVII Pantalla Relacionados/Podcast INTF03 Resultado Final

Identificación	INTF03
Tipo	Salida
Formato	Impresa
Contenido	Pantalla correspondiente a la sección de relacionados que muestra podcast de misterio.
Frecuencia	A petición del usuario
Requisitos	RF04 RF06

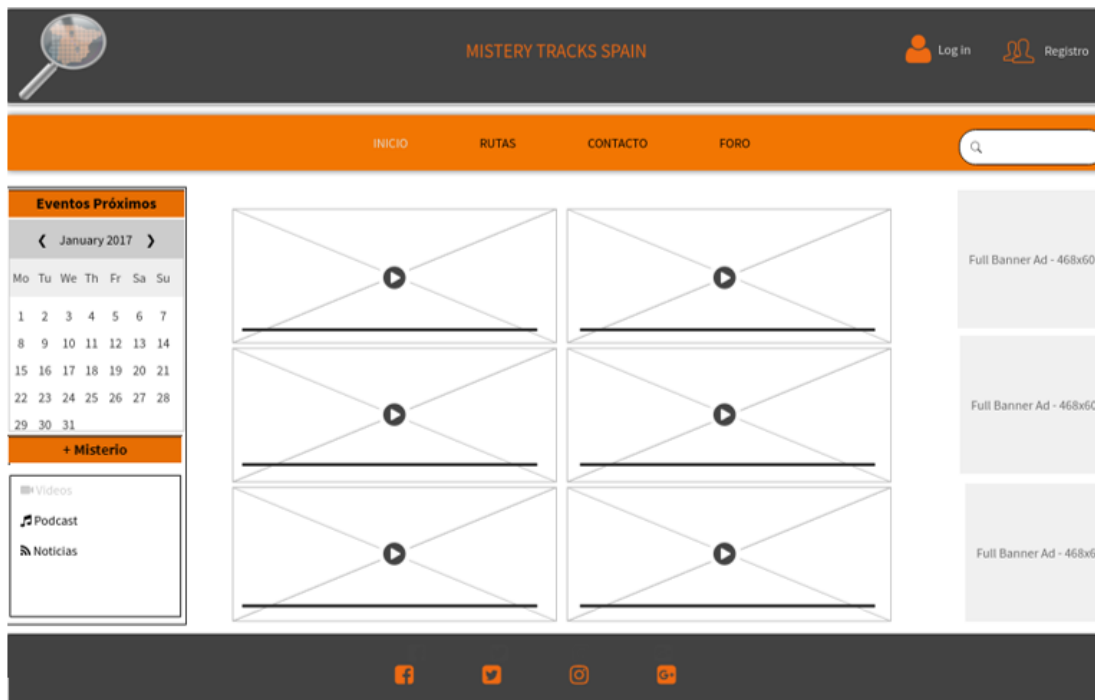


Ilustración XVIII Pantalla Relacionados/Videos INTF04 Prototipo

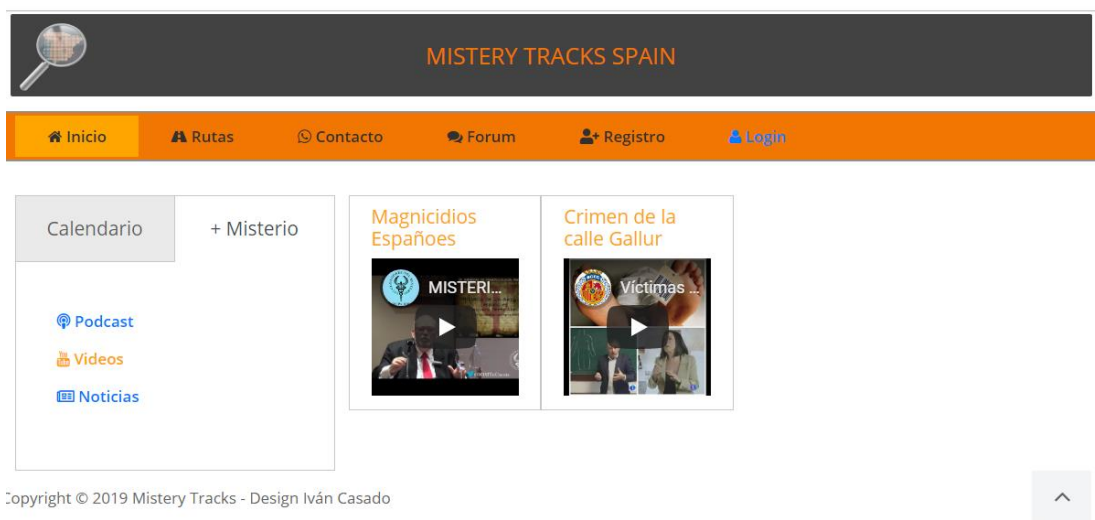
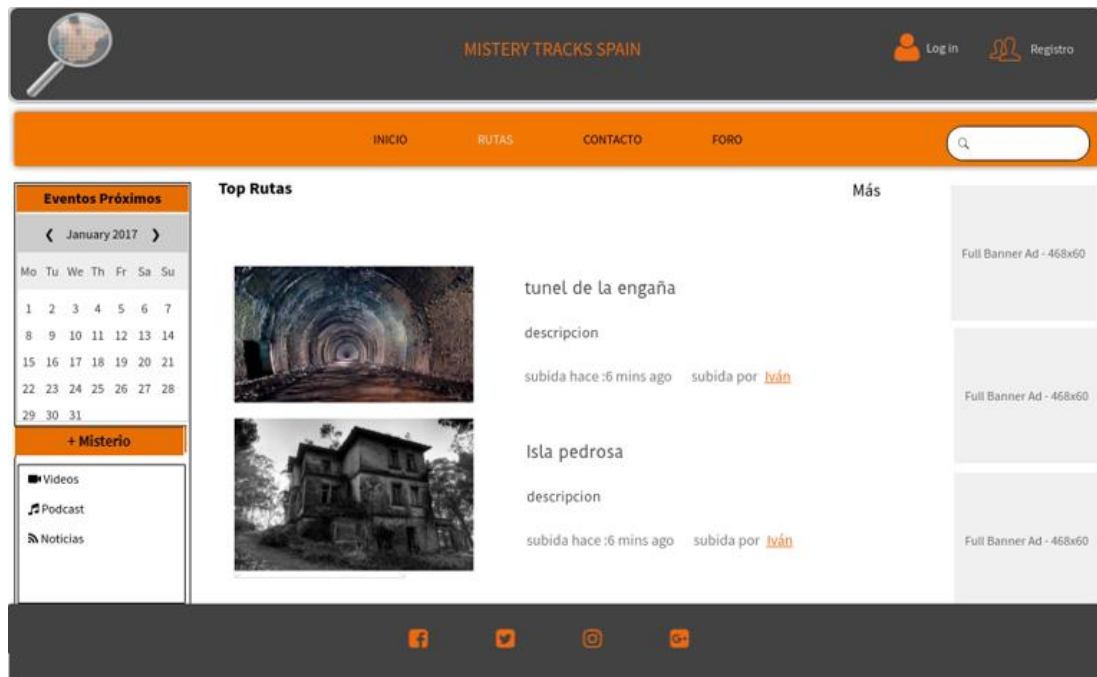


Ilustración XIX Pantalla Relacionados/Videos INTF04 Resultado Final

Identificación	INTF04
Tipo	Salida
Formato	Impresa
Contenido	Pantalla correspondiente a la sección de relacionados que muestra videos de misterio.
Frecuencia	A petición del usuario
Requisitos	RF04 RF06

*Ilustración XX Pestaña Rutas INTF05 Prototipo*

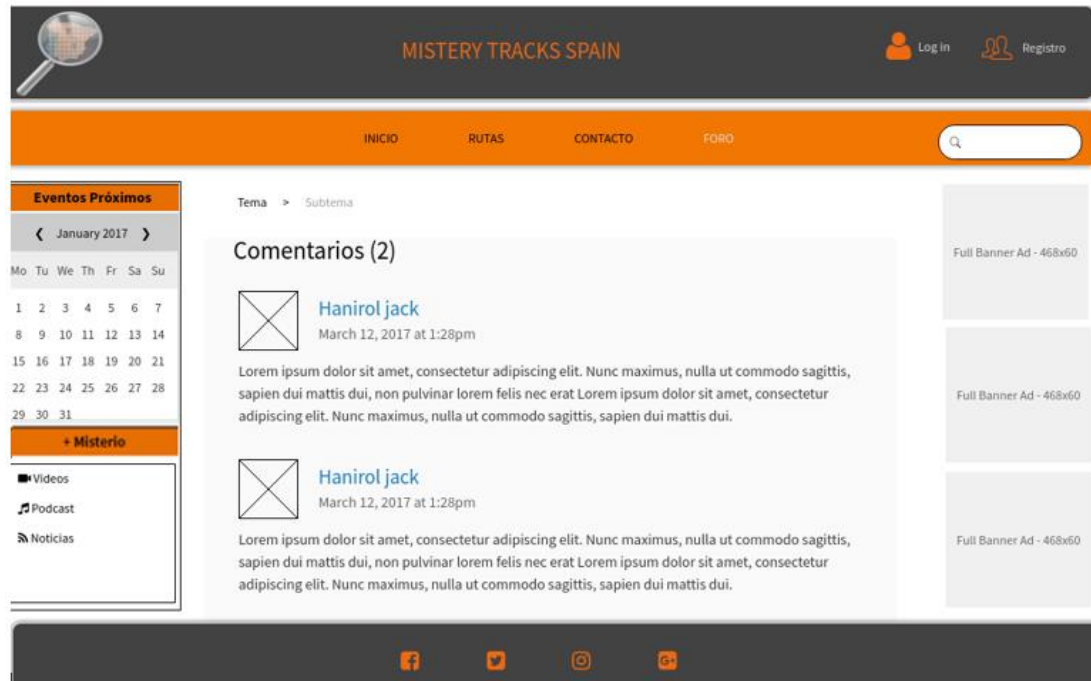
Identificación	INTF05
Tipo	Salida
Formato	Impresa
Contenido	Pestaña rutas que muestra información de rutas creadas por los usuarios
Frecuencia	A petición del usuario
Requisitos	RF03 RF04



Ilustración XXI Pestaña contacto INTF06 Prototipo

Ilustración XXII Pestaña Contacto INTF06 Resultado Final

Identificación	INTF06
Tipo	Entrada
Formato	Impresa
Contenido	Pestaña contacto que muestra un formulario para que el usuario pueda escribir sus dudas. Solicita un nombre, un mail y un mensaje. Y lo inserta en base de datos.
Frecuencia	A petición del usuario
Requisitos	RF03 RF04

*Ilustración XXIII Pestaña foro INTF07 Prototipo*

Identificación	INTF07
Tipo	Salida / Entrada
Formato	Impresa
Contenido	Pestaña foro que muestra un foro para interacción de los usuarios. Muestra los temas y permite añadir mensajes
Frecuencia	A petición del usuario
Requisitos	RF10



MISTERY TRACKS SPAIN

Log in Registro

Full Banner Ad - 468x60

Full Banner Ad - 468x60

Full Banner Ad - 468x60

USUARIO Su user

CONTRASEÑA Su contraseña

Entrar Reset

Volver

f t i g +

Ilustración XXIV Pantalla Login INTF08 Prototipo

MISTERY TRACKS SPAIN

Inicio Rutas Contacto Forum Registro Login

Iniciar Sesión

Usuario

Contraseña

Nuevo aquí? [Unete](#)

Enviar

Copyright © 2019 MysteryTracksSpain - Diseño: Iván Tooplate

^

Ilustración XXV Pantalla Login INTF08 Resultado Final

Identificación	INTF08
Tipo	Entrada
Formato	Impresa
Contenido	Pantalla login que permite acceder a los usuarios registrados Solicita el usuario y la contraseña, y lo comprueba en la base de datos.
Frecuencia	A petición del usuario
Requisitos	RF01 RF02

*Ilustración XXVI Pantalla Registro INTF09**Ilustración XXVII Pantalla Registro INTF09 Resultado Final*

Identificación	INTF09
Tipo	Entrada
Formato	Impresa
Contenido	Pantalla login que permite registrar un nuevo usuario. Solicita nombre, apellido, user, email, contraseña y fecha nacimiento. Y lo inserta en la base de datos.
Frecuencia	A petición del usuario
Requisitos	RF01 RF02



Las siguientes pantallas son accesibles después de iniciar sesión.

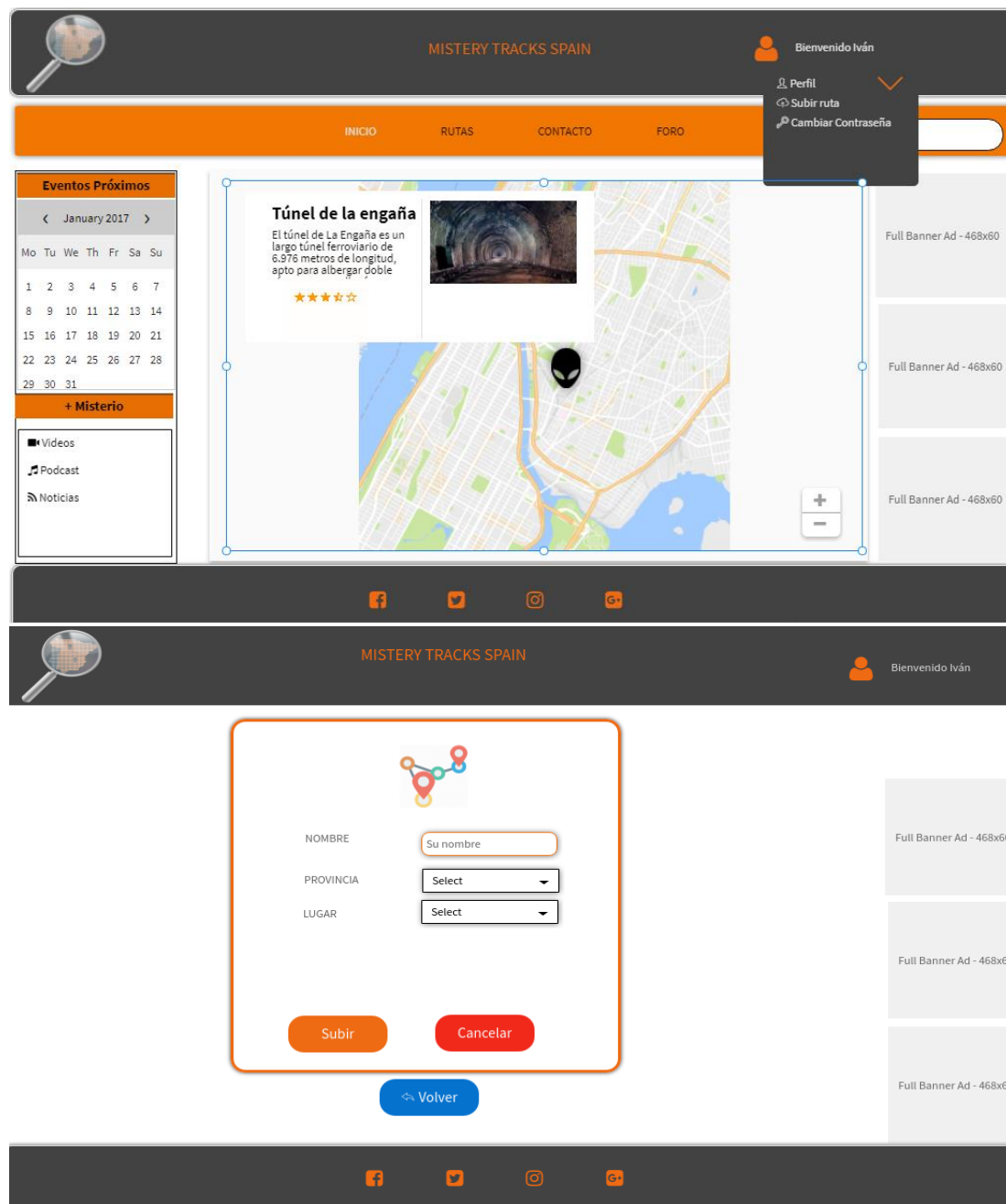


Ilustración XXVIII Pantalla subir ruta INTF10

Identificación	INTF10
Tipo	Entrada
Formato	Impresa
Contenido	Pantalla subir ruta que permite registrar una nueva ruta o un lugar. Solicita nombre usuario, provincia y lugar. Y lo inserta en la base de datos.
Frecuencia	A petición del usuario
Requisitos	RF09



MISTERY TRACKS SPAIN

Bienvenido Iván

Full Banner Ad - 468x60

Full Banner Ad - 468x60

Full Banner Ad - 468x60

Facebook Twitter Instagram Google+

Ilustración XXIX Cambiar contraseña INTF11

Identificación	INTF10
Tipo	Entrada
Formato	Impresa
Contenido	Pantalla que permite al usuario cambiar su contraseña. Solicita contraseña nueva y antigua. Y lo inserta en la base de datos.
Frecuencia	A petición del usuario
Requisitos	RF11

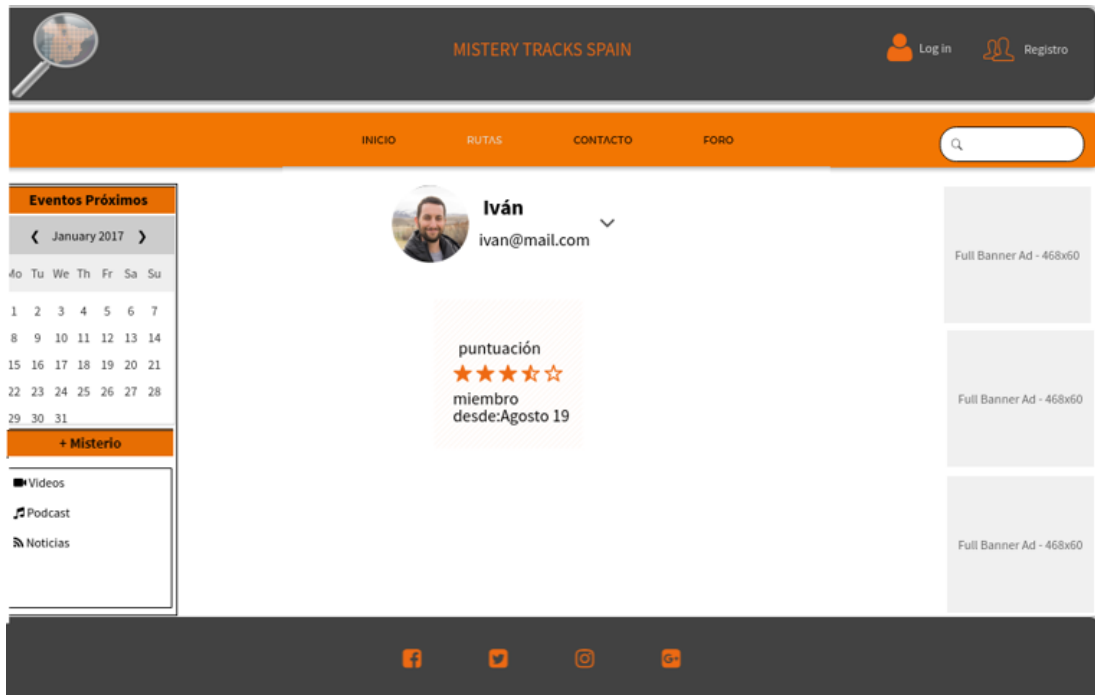


Ilustración XXX Pantalla Perfil Usuario INTF12

Identificación	INTF11
Tipo	Salida
Formato	Impresa
Contenido	Pantalla que permite al usuario ver su información.
Frecuencia	A petición del usuario
Requisitos	RF12



5.6 Diseño de algoritmos

Proceso login

Diagrama que explica el proceso de un usuario cuando intenta iniciar sesión.

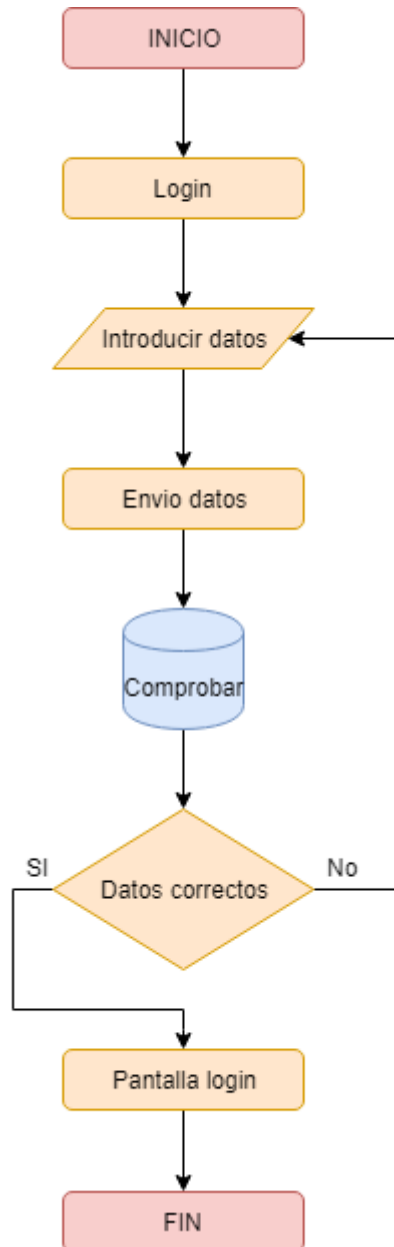


Ilustración XXXI Diagrama de flujo Login



Proceso registro

Diagrama que explica el proceso cuando un usuario intenta registrarse en la aplicación.

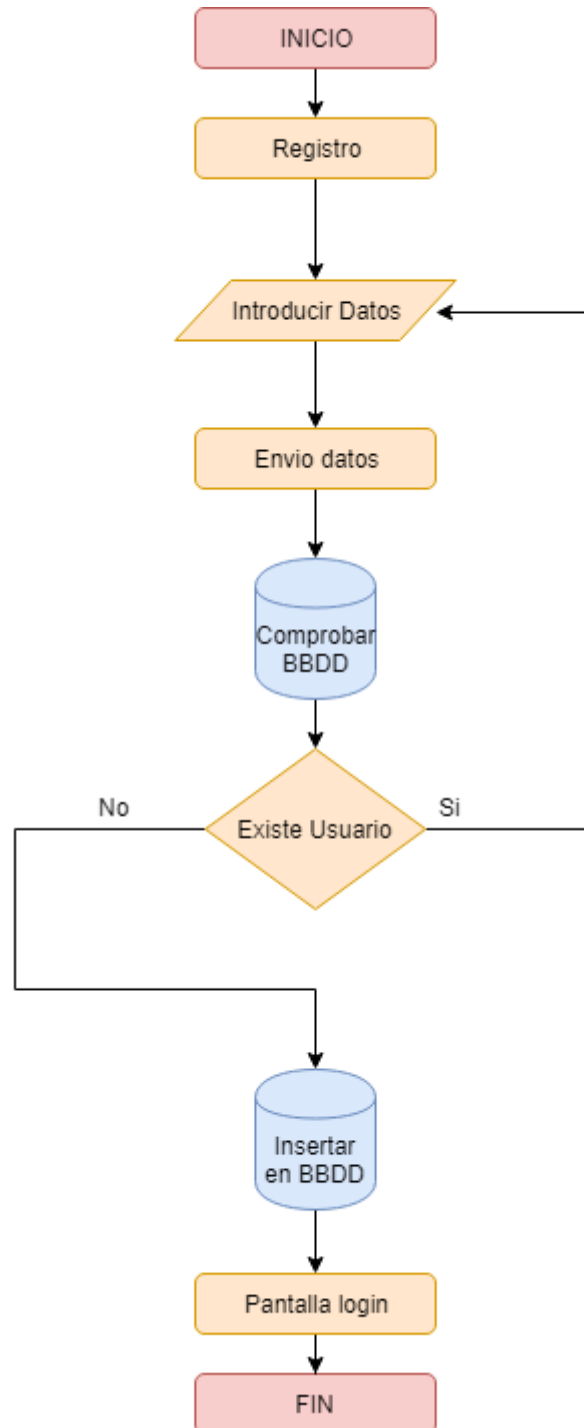


Ilustración XXXII Diagrama flujo Nuevo Usuario



Proceso emitir voto

Diagrama que explica el proceso cuando usuario intenta añadir un nuevo voto.

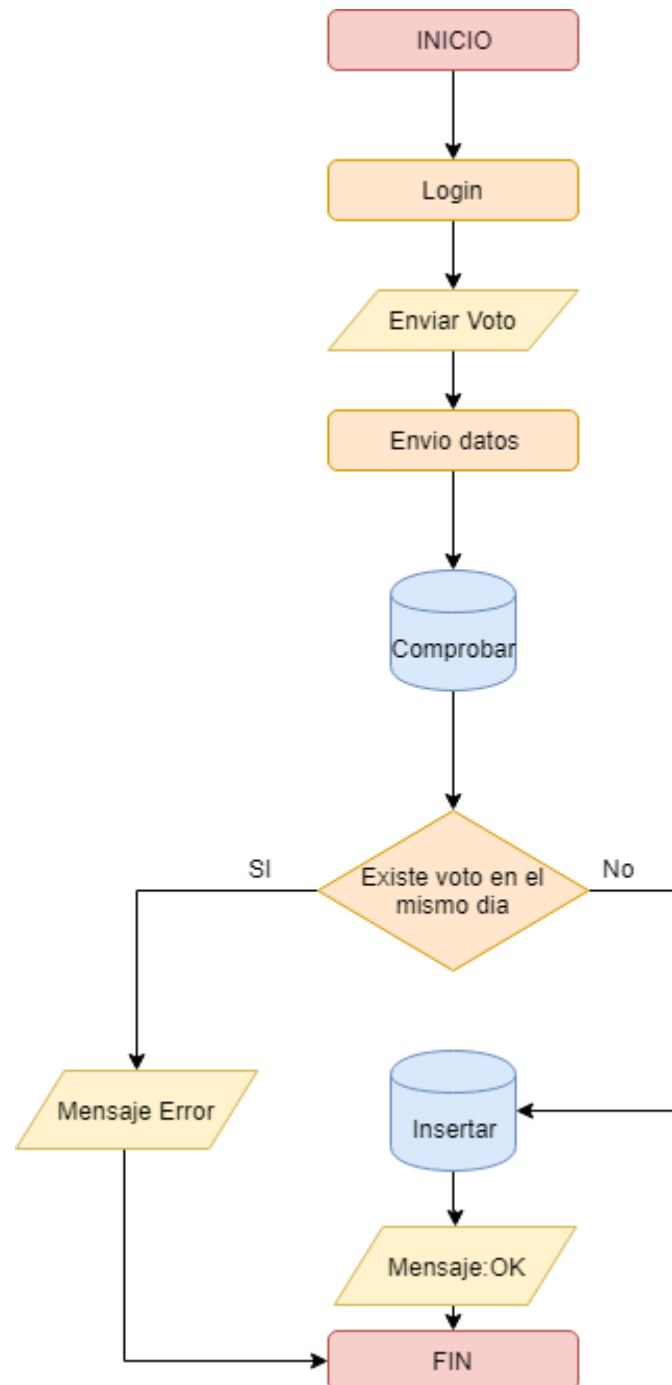


Ilustración XXXIII Diagrama Flujo Emitir Voto



Proceso subir contenido

Diagrama que explica el proceso cuando un usuario intenta añadir nuevo contenido.

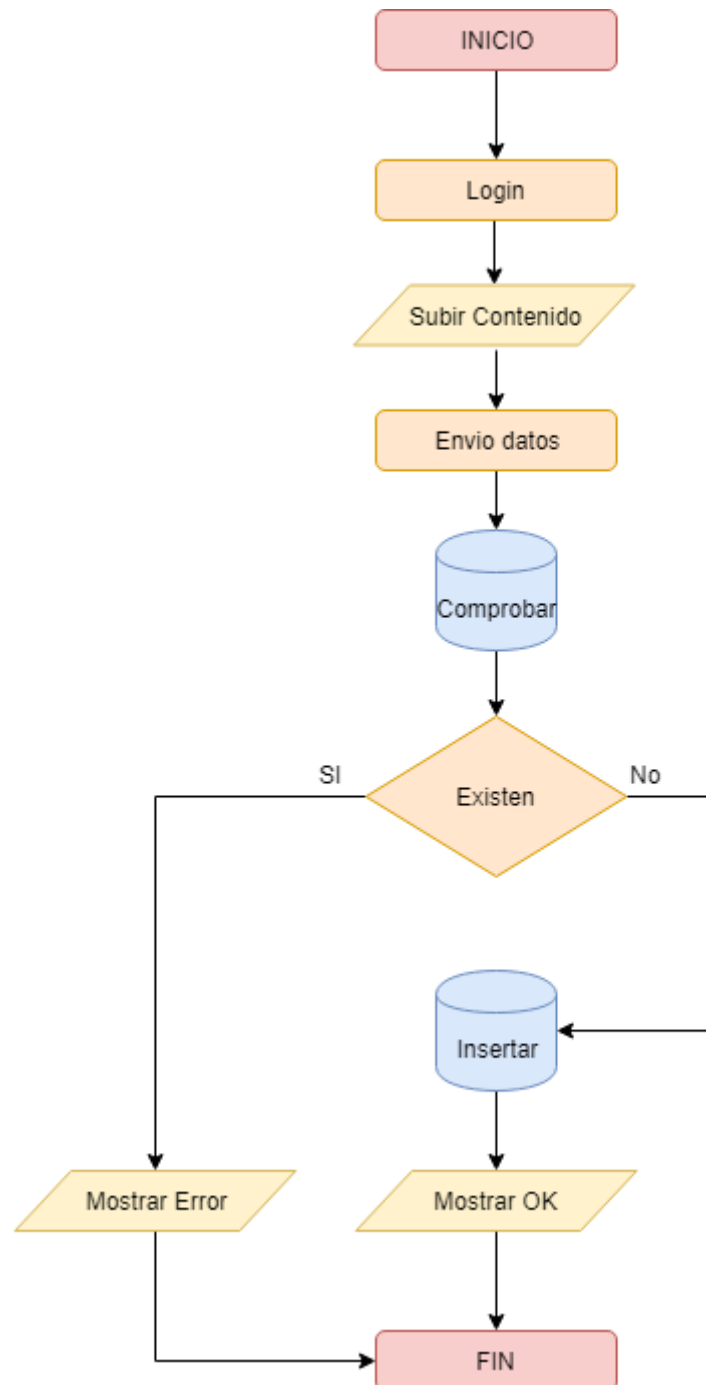


Ilustración XXXIV Diagrama Flujo Subir Contenido



Proceso Buscar

Diagrama que explica el proceso cuando un usuario hace una búsqueda.

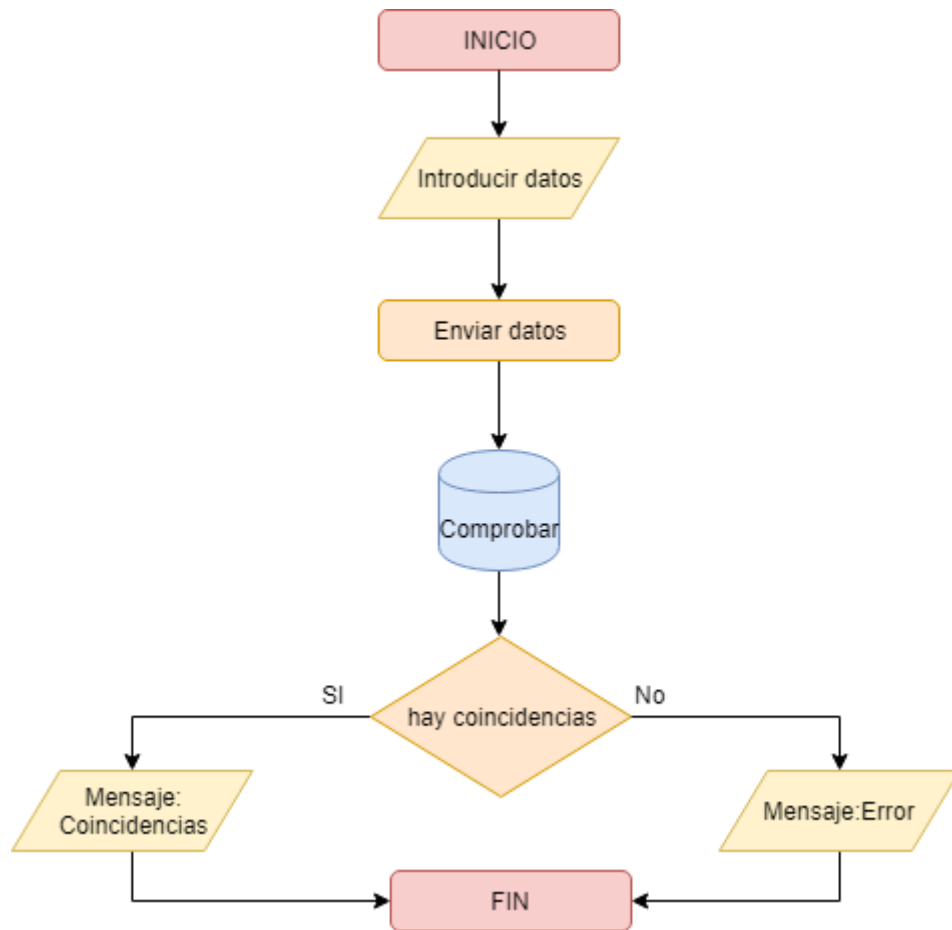


Ilustración XXXV Diagrama Flujo Búsqueda



6. PLANIFICACIÓN

6.1 Fases

Fase1: Información

Una vez que se tiene la idea comienza a recabarse información al respecto recogiendo ideas y comprobando si pudiera llegar a ser factible. Además, se comienza la elaboración del documento que recoja todos estos datos.

En esta primera etapa se hará la introducción sobre la propuesta escogida, se declararán los objetivos y se hará el análisis de requisitos y de contexto.

Fase2: Diseño

Cuando la idea estaba clara empieza la fase de diseño. Los diseños realizados son los siguientes:

- Distribución de la red local y forma de conexión remota.
- Diagrama Entidad/Relación de la base de datos a utilizar.
- Diagrama de clases.
- Prototipado y mapa de navegación de las interfaces.

Fase3: Codificación

En esta fase se dedica un tiempo a formación al tratarse de un lenguaje y un framework que no se dominan.

Teniendo todos los diseños hechos, se comienzan las labores de desarrollo a la vez que se va aprendiendo sobre la tecnología elegida:

- Máquina virtual en la que instalar el IDE el servidor de la base de datos y el servidor de aplicaciones.
- Crear la BBDD con sus tablas y relaciones.
- Crear las clases necesarias para la aplicación.
- Crear las pantallas basadas en el prototipo.

Una vez que los aspectos básicos de la aplicación están creados, se harán los test de funcionalidad.

Fase4: Documentación

En esta etapa se realizarán los manuales técnicos para los usuarios y se terminará la documentación que ya se ha ido haciendo durante el resto de fases.



6.2 Cronograma

ACTIVIDADES	MESES															
	1(Marzo)				2(Abril)				3(Mayo)				4(Junio)			
	SEMANAS															
	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4
FASE 1 Información																
Resumen																
Introducción																
Objetivos																
Análisis del contexto y estado del arte																
Análisis de requisitos																
FASE 2 Diseño																
Diagrama de red																
Diagrama entidad/relación																
Diagrama clases																
Diagrama de interfaces																
Diseño de procedimientos																
Planificación																
Implementación																
FASE 3 Codificación																
Puesta en marcha																
Explotación																
Pruebas																
Control calidad																
Gestión económica																
FASE 4 Documentación																
Manuales y maquetación																



6.3 Recursos

Materiales

Se instalará un S.O específico para virtualización de servidores: Citrix xenserver 7.6.El servidor será 1 módulo DELL PowerEdge R440.

La aplicación estará instalada sobre tomcat como servidor de aplicaciones y apache2 como servidor web en una máquina virtual con Ubuntu Server 18.04 en la que también se instalará el servidor de base de datos.

Con la intención de evitar caídas y ofrecer el servicio el mayor tiempo posible, el servidor DELL tendrá doble salida de corriente alterna y 2 ISP distintos ofreciendo internet.

La salida a internet, el balance de carga y las tareas de firewall serán mediante un Mikrotik RB4011UiAS-RM.

Las tareas de desarrollo se harán con un MSI GL62M 7RDX.

Como sistema de backup un NAS Synology DiskStation DS1819+.

Humanos

Las áreas que compondrán la empresa estarán a cargo de una única persona y son las siguientes:

- Administración: Se encarga de la administración interna de la empresa, de los contratos con los clientes y de las acciones de RRHH.
- Desarrollo: Se encarga del mantenimiento y soporte de la aplicación, así como de la investigación y el desarrollo de mejoras y nuevas funcionalidades.
- Marketing: Se encarga de la atención al cliente, del mantenimiento de la página web y de las redes sociales. Tendrá en cuenta las opiniones de los clientes y sus valoraciones.
- Sistemas: Se encarga del mantenimiento y soporte de los servidores e incidencias.

Logística

No hay recursos logísticos previstos



7. DESARROLLO

Explicación

Para el desarrollo se ha elegido el lenguaje JAVA en concreto J2EE. Por lo cual el front-end estará en jsp y html.

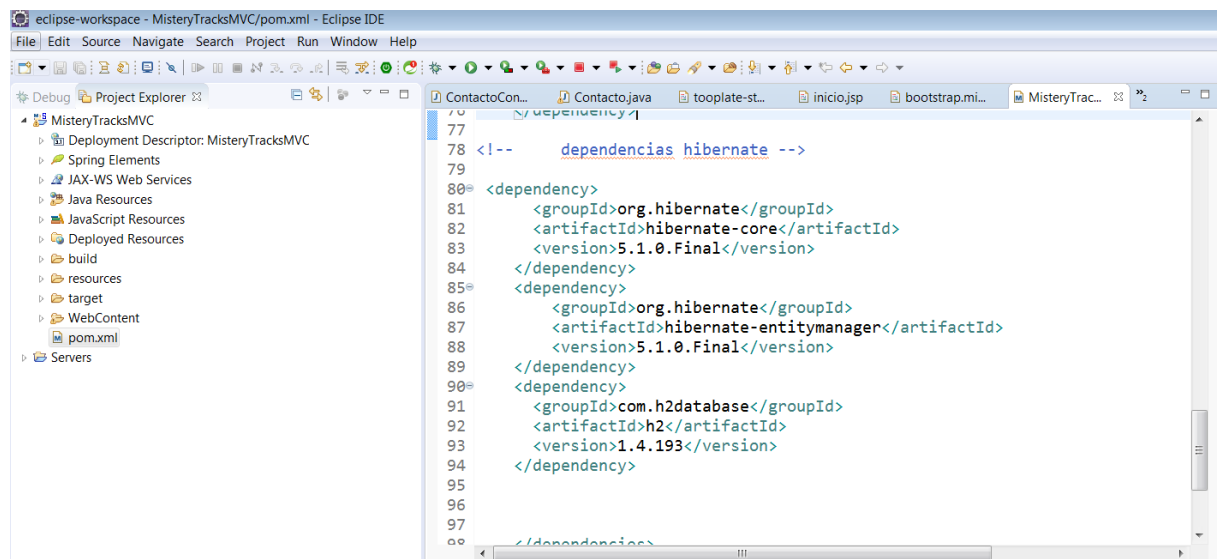
Para que todo esto funcione se utiliza el modelo vista controlador SpringMVC que es un framework muy completo y que se puede integrar fácilmente en el IDE eclipse.

La arquitectura del proyecto será Dynamic Web Project y se transformará a Apache MAVEN que es una herramienta para gestión de proyectos en JAVA, será la encargada de gestionar todas las dependencias, drivers, librerías y repositorios necesarios para el funcionamiento del proyecto.

7.1 BACK-END

Aquí se va a explicar la parte del proyecto que “no se ve” y que enseña la forma en que el proyecto gestiona los datos.

A continuación, se muestra una parte del fichero pom.xml que utiliza MAVEN, en el que se ven como se añaden las dependencias.



```
77
78 <!-- dependencias hibernate -->
79
80 <dependency>
81   <groupId>org.hibernate</groupId>
82   <artifactId>hibernate-core</artifactId>
83   <version>5.1.0.Final</version>
84 </dependency>
85 <dependency>
86   <groupId>org.hibernate</groupId>
87   <artifactId>hibernate-entitymanager</artifactId>
88   <version>5.1.0.Final</version>
89 </dependency>
90 <dependency>
91   <groupId>com.h2database</groupId>
92   <artifactId>h2</artifactId>
93   <version>1.4.193</version>
94 </dependency>
95
96
97
98 </dependencies>
```

Ilustración XXXVI Gestor de dependenciaspom.xml

Lo que se puede ver son 2 dependencias

com.h2database que es la utilizada para la base de datos H2 que es la utilizada para la construcción proyecto.

Hibernate-entitymanager se encargará de identificar mediante @anotaciones las clases y sus atributos, insertando en BBDD el nombre de la clase como tabla y los atributos como campos.



Ilustración XXXVII Anotaciones utilizadas por hibernate

Después de configurar las dependencias se configuraría el despachador **dispatcher-servlet.xml** que lo que hace es indicar a la aplicación donde esta cada cosa.

indica cual es el paquete en el que está el controlador principal





Resources

Hace el mapeo de carpetas para que el .jsp sepa dónde buscar los css imágenes etc..

```
<!-- se añade la carpeta de imagenes para poder usarlas en el jsp -->
<mvc:resources location="/resources/img/"
mapping="/img/**"></mvc:resources>

<!-- se añade la carpeta de css para poder usarlas en el jsp -->
<mvc:resources location="/resources/css/"
mapping="/css/**"></mvc:resources>

<!-- se añade la carpeta de css para poder usarlas en el jsp -->
<mvc:resources location="/resources/webfonts/"
mapping="/webfonts/**"></mvc:resources>

<!-- se añade la carpeta de videos para poder usarlas en el jsp -->
<mvc:resources location="/resources/videos/"
mapping="/videos/**"></mvc:resources>

<!-- se añade la carpeta de js para poder usarlas en el jsp -->
<mvc:resources location="/resources/js/" mapping="/js/**"></mvc:resources>
```

Ilustración XXXIX Imagen del mapeo de carpetas

ViewResolver

Recoge los datos del controlador para mostrar la vista deseada.

```
<bean id="viewResolver"
class="org.springframework.web.servlet.view.InternalResourceViewResol
<property name="viewClass"
value="org.springframework.web.servlet.view.JstlView" />
<property name="prefix" value="/WEB-INF/jsp/" />
<property name="suffix" value=".jsp" />
</bean>
```

Ilustración XL Imagen ViewResolver



Ejemplo ViewResolver:

```
import org.springframework.stereotype.Controller;

@Controller
public class MysteryTracksSpainController {

    // acciones si en la url no aparece algo del estilo /rutas
    @RequestMapping("/")
    public String sayRaiz(Model model) {

        // atributos que manda
        model.addAttribute("saludo", "Hola Mundo!!!");
        model.addAttribute("mensaje",
            "Me llena de orgullo y satisfacción saludaros en este primer
            // como se llama la pagina a la que te redirige
            return "inicio";
    }
}
```

Ilustración XLI Ejemplo ViewResolver

Cuando el usuario hace una petición a la IP de la página la recoge este controlador con el `@RequestMapping("/")` crea unos atributos que se mandarían a la vista y utiliza el return para ir a la página inicio , como se puede ver en la ilustración 33

El despachador recibe una cadena de texto con el valor inicio y concatena la extensión (prefix) .jsp y nos envía a dicha página.

Si se quisiera mostrar el “saludo” bastaría con poner `{ $saludo }` en la vista.

Ejemplo del formulario de contacto, que recibe un objeto y muestra sus atributos

```
1 <@ page language="java" contentType="text/html; charset=utf-8"
2 pageEncoding="utf-8"%>
3 <html>
4 <head>
5 </head>
6 <body>
7 <h2>Información de contacto enviada</h2>
8 <table>
9 <tr>
10 <td>Nombre :</td>
11 <td>${contacto.nombre}</td>
12 </tr>
13 <tr>
14 <td>Mensaje :</td>
15 <td>${contacto.mensaje}</td>
16 </tr>
17 <tr>
18 <td>Email :</td>
19 <td>${contacto.email}</td>
20 </tr>
21 <tr>
22 <td>id :</td>
23 <td>${contacto.id}</td>
24 </tr>
25 </table>
26 </body>
27 </html>
28
29
```

Ilustración XLII Ejemplo de salida de datos clase contacto



dataSource

indica el nombre de la base de datos y las claves de acceso

```
<!-- base datos -->
<bean id="dataSource"
      class="org.springframework.jdbc.datasource.DriverManagerDataSource">
    <property name="driverClassName" value="org.h2.Driver" />
    <property name="url" value="jdbc:h2:./mystery" />
    <property name="username" value="sa" />
    <property name="password" value="" />
</bean>
```

Ilustración XLIII origen de los datos de la BBDD

messageProperties

indica donde está el archivo con todos los mensajes de error, que luego se usaran para la validación de formularios

```
<!-- Este bean nos permitirá usar message properties -->
<bean id="messageSource"
      class="org.springframework.context.support.ReloadableResourceBundleMe

    <property name="basename" value="/WEB-INF/messages" />
</bean>
```

Ilustración XLIV Message properties

1NotEmpty.registroForm.nombre=Por favor introduzca el nombre
2Size.registroForm.nombre=El nombre debe tener entre 4 y 20 caracteres
3Size.registroForm.apellido=El apellido debe tener entre 4 y 20 caracteres
4Email.registroForm.email=Introduzca un mail valido
5Size.registroForm.contrasena=la contrasena debe tener entre 4 y 10 caracteres

Ilustración XLV Muestra mensaje properties

Entitymanager

Indica la ruta en la que el programa tiene que scanear en busca de clases con la anotación @Entity y así saber que esa clase se insertará en BBDD.

```
<!-- entity manager (las clases para la base de datos) -->
<bean id="entityManagerFactory"
      class="org.springframework.orm.jpa.LocalContainerEntityManagerFactory"
    <property name="dataSource" ref="dataSource" />
    <property name="jpaVendorAdapter" ref="jpaVendorAdapter" />
    <!-- spring based scanning for entity classes -->
    <property name="packagesToScan" value="com.mysterytracksspain.model"
</bean>
```

Ilustración XLVI Manager de entidades



```
2
3 /*Anotaciones Utilizadas por entity manager*/
4 @Entity
5 /*Tabla correspondiente en Base de datos*/
6 @Table(name="CONTACTO")
7 public class Contacto {
8
```

Ilustración XLVII Ejemplo anotación @Entity

Hasta aquí llegan los datos utilizados por el despachador, a continuación, se creó la estructura de paquetes añadiendo 3 más que son controller , model y repo:

7.1.2 Estructura del proyecto

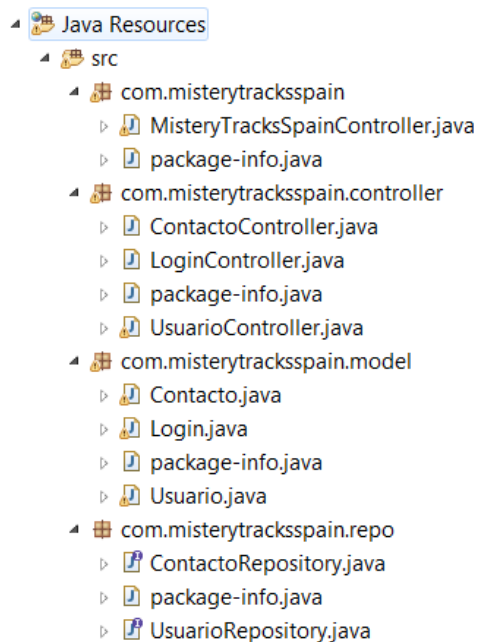


Ilustración XLVIII Estructura de paquetes

Controller

En este paquete se alojarán los controladores de cada clase, las clases que usan formulario deberán tener un controlador para atender las peticiones

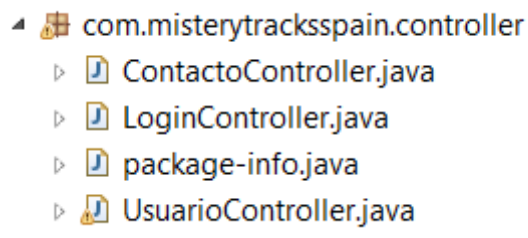


Ilustración XLIX Controladores de clases



```
@Controller
public class UsuarioController {

    /*
     * inyeccion del repositorio de usuarios para el uso de la base de datos
     */
    @Autowired
    private UsuarioRepository usuarioRepository;

    // accion cuando en la url detecta /registro
    @RequestMapping(value = "/registro", method = RequestMethod.GET)
    public String showForm(Model model) {
        Usuario registro = new Usuario();
        model.addAttribute("usuarioForm", registro);
        return "registro";
    }

    // accion cuando en la url detecta addUsuario
    // la peticion viene del formulario de registro
    @RequestMapping(value = "/addUsuario", method = RequestMethod.POST)
    public String submit(@Valid @ModelAttribute("usuarioForm") Usuario usuario, BindingResult result, Model model) {

        // comprobacion de si la direccion de correo ya existe
        if (usuarioRepository.exists(usuario.getEmail())) {
            // envio de mensaje
            model.addAttribute("errorusuario", "el usuario ya existe");
            // retorno a la pagina de registro
            return "registro";
        } else {
            // comprobacion de que sean iguales
            if (usuario.validaContraseña()) {
                // comprobacion del resto de campos mediante anotaciones
                if (result.hasErrors()) {
                    // vuelta al formu registro
                    return "registro";
                } else {
                    // si no existe lo guarda en BBDD
                    usuarioRepository.save(usuario);
                    // envia a la vista de registro un objeto usuario llamado registro
                    model.addAttribute("registro", usuario);
                    return "viewRegistro";
                }
            } else {
                // si las contraseñas no coinciden crea un mensaje y retorna al registro
                model.addAttribute("errorcontrasena", "las contraseñas no coinciden");
                return "registro";
            }
        }
    }
}
```

Ilustración L Ejemplo clase controladora

Como se ve en la imagen precedente, el usuario al pulsar el botón enviar en el formulario de registro envía un valor **/addRegistro** que lo que hace recibir los datos por POST y validar los campos (utilizando el result) y que las contraseñas sean iguales, si todo es correcto comprueba que el usuario no existe en el repositorio UsuarioRepository (utilizando el mail que es la primary key) si tampoco da error entonces lo añade a la base de datos.

Repo

En este paquete es donde se guardarán las interfaces encargadas de las operaciones que tengan que ver con las bases de datos, cada clase que tenga tabla deberá tener un repositorio e implementar su interfaz NombreclaseRepository para poder operar con ella.

Ejemplo

```

8
9 @Repository
10 /*
11  * extender la interfaz, hay que incluir dos clases:
12  * 1-Usuario, el tipo de entidad
13  * que va a ser gestionada
14  * 2-El tipo de dato del atributo que esté marcado
15  * como @Id en dicha clase.
16  */
17
18 //esta interfaz se encargara de interactuar con la tabla USUARIO
19 public interface UsuarioRepository extends JpaRepository<Usuario,String> {
20
21 }
22

```

Ilustración LI Ejemplo interfaz Repository

Para que funcione solo debe crearse como tipo interfaz y extender JpaRepository a la cual indicamos el tipo de objeto y el tipo de @id que tiene la clase.

Para usar los métodos de acceso a datos debemos iniciar la interfaz e inyectar los datos con la anotación `@Autowired` que utiliza Spring MVC.

```
@Controller
public class UsuarioController {

    /*
     * inyeccion del repositorio de usuarios para el uso de la base de datos
     */
    @Autowired
    private UsuarioRepository usuarioRepository;
```

Ilustración LII Inyección de repositorios con @Autowired

Así ya se podrían usar los métodos propios de la interfaz o incluso se pueden crear consultas personalizadas, equivalen a las queries en otros tipos de BBDD como Oracle o Mysql.

```
// accion cuando en la url detecta addUsuario
// la peticion viene del formulario de registro
@RequestMapping(value = "/addUsuario", method = RequestMethod.POST)
public String submit(@Valid @ModelAttribute("usuarioForm") Usuario usuario, BindingResult result) {

    // comprobacion de si la direccion de correo ya existe
    if (usuarioRepository.exists(usuario.getEmail())) {
        // envio de mensaje
        model.addAttribute("errorusuario", "el usuario ya existe");
        retorno a la pagina de registro
        return "registro";
    }
}
```

Ilustración LIII Ejemplo de consultas JPA

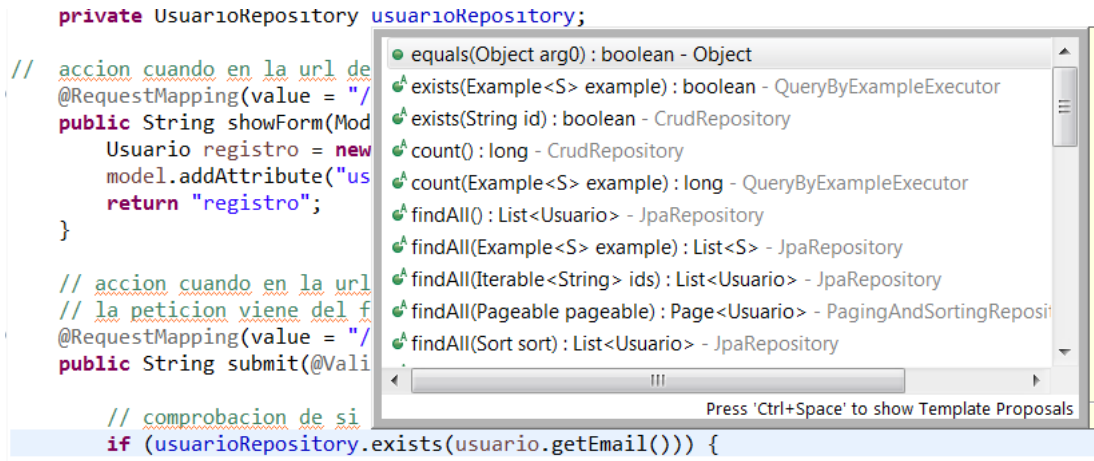


Ilustración LIV Métodos de acceso a datos de JPA

7.2 FRONT-END

7.2.1 Estructura

Esta parte es la que se comunica con los controladores para enviar/recibir los datos de los controladores.

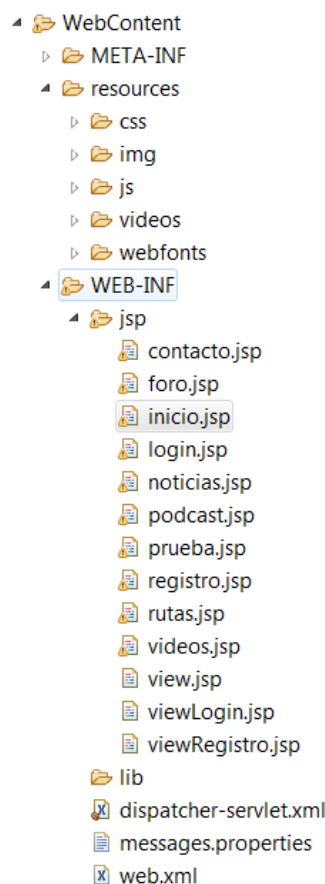


Ilustración LV Estructura Web



7.2.2 Vistas

Las vistas estarán todas almacenadas en la misma carpeta al igual que las “resources”, cuando queramos acceder a scripts imágenes ... deberemos indicarlo el jsp de la siguiente manera:

Primero: importando el jsp/core añadiendo el prefijo “c”

```
<!-- tag para poder usar mapping resources con css, img, video creado en dispatcher -->
<%@taglib prefix="c" uri="http://java.sun.com/jsp/jstl/core"%>
</taglib>
```

Segundo: indicando el prefijo la ruta del recurso deseando

```
l6 <link rel="stylesheet"
l7     href='<c:url value="css/tooplate-style.css"></c:url>'>
l8 <link rel="stylesheet" href='<c:url value="css/calendario.css"></c:url>'>
l9 </link>
```

Ilustración LVI Ejemplo url css

```
<div class="tm-company-img-container">
    aqui va el mapa -->
    <img src='<c:url value="/img/mapa.png"></c:url>' alt="Image" />
</div>
```

Ilustración LVII Ejemplo inserción imagen

Para las redirecciones se usarán etiquetas <a> cuyo value enviara al controlador la página a la que quiere ir

```
<ul class="navbar-nav mx-auto tm-navbar-nav">
  <li class="nav-item activo"><a class="nav-link" href="#"><i
    class="fa fa-1x fa-home"></i><b> Inicio</b><span
    class="sr-only">(current)</span></a></li>
  <li class="nav-item"><a class="nav-link" href="rutas"><i
    class="fa fa-1x fa-road"></i><b> Rutas</b></a></li>

  <li class="nav-item"><a class="nav-link " href="contacto"><i
    class="fab fa-whatsapp"></i><b> Contacto</b></a></li>

  <li class="nav-item"><a class="nav-link" href="foro"><i
    class="fa fa-1x fa-comments"></i><b> Forum</b></a></li>
```

Ilustración LVIII uso de href



7.2.3 Formularios

Para el uso de formularios deberemos usar unas etiquetas específicas

Como son `<form:form>` para indicar 2 atributos 1 `modelAttribute` que es lo que envía el controlador de controlador correspondiente y `action` que indica el método que va a usar el formulario para trabajar con los datos.

```
-----  
<form:form modelAttribute="usuarioForm" action="addUsuario"  
           class="tm-contact-form" method="POST">  
  
accion cuando en la url detecta /registro  
@RequestMapping(value = "/registro", method = RequestMethod.GET)  
public String showForm(Model model) {  
    Usuario registro = new Usuario();  
    model.addAttribute("usuarioForm", registro);  
    return "registro";  
}  
  
// accion cuando en la url detecta addUsuario  
// la peticion viene del formulario de registro  
@RequestMapping(value = "/addUsuario", method = RequestMethod.POST)  
public String submit(@Valid @ModelAttribute("usuarioForm") Usuario usuario, BindingResult result, Model model) {  
  
    // comprobacion de si la direccion de correo ya existe  
    if (usuarioRepository.exists(usuario.getEmail())) {  
        // envío de mensaje  
    }  
}
```

Ilustración LIX Acciones de los formularios

`<form:label>` que indica los nombres de la cabecera de la caja de texto

`<form:input>` funciona igual que el `input` de HTML pero con una diferencia, que aquí hay que indicar un `path`, que no es otra cosa que el nombre del atributo de la clase a la que pertenece el campo, esto se usará luego en las validaciones.

```
-----  
<form:input type="password" path="contrasena" id="contact_pass"  
           class="form-control" placeholder="Contraseña" required="true"></form:input>  
-----  
  
/*anotaciones validacion formulario registro*/  
@Size(min = 4, max = 20)  
@NotBlank  
/*anotaciones acceso a datos*/  
@Column(name="nombre")  
private String nombre;
```

Y por último `<form:errors>` que recoge errores en caso de haberlos, los errores que muestra son los que se han añadido en el `message.properties` que se muestra en el punto 7.1.1



7.2.4 Validar Formularios

Como se indica en el punto anterior, se dispone de una etiqueta específica para recoger los errores enviados por el controller.

Tras pulsar el botón de enviar el controlador recoge los datos enviados mediante post por el formulario, para validar se hace uso de anotaciones

```
/*anotaciones validacion formulario registro*/
@NotBlank
@Size(min = 4, max = 10)
/*anotaciones acceso a datos*/
@Column(name="contrasena")
private String contrasena;
```

Ilustración LX Anotaciones validación formularios

En la imagen anterior se usa `@NotBlank` que indica que el campo no puede estar vacío y de `@Size` que no permite que la contraseña se menor de 4 caracteres ni mayor de 10.

```
@RequestMapping(value = "/addContacto", method = RequestMethod.POST)
public String submit(@Valid @ModelAttribute("contactoForm") Contacto contacto, BindingResult result, Model model) {
    if (result.hasErrors()) {
        return "contacto";
    } else {
        model.addAttribute("contacto", contacto);
        contactin.save(contacto);
        return "view";
    }
}
```

Se hace mediante `BindingResult` que devuelve un objeto `result` que si tienes errores nos devuelve a la página del formulario y mostrando los mensajes de error , en caso de no tener errores introduce el objeto en su repositorio y este a la base de datos.

Registro

as

el tamaño tiene que estar entre 4 y 20

as

el tamaño tiene que estar entre 4 y 20

Ilustración LXI Demostración validación



También se pueden crear reglas de validación propias como el utilizado para comparar las 2 contraseñas introducidas en el registro, que en caso de ser distintas nos muestra un mensaje personalizado.

El único problema es que para recogerlo debemos embeber código Java dentro del código html.

```
// comprobacion de que sean iguales
if (usuario.validaContrasena()) {

    // comprobacion del resto de campos mediante anotaciones
    if (result.hasErrors()) {
        // vuelta al formu registro
        return "registro";
    } else {
        // si no existe lo guarda en BBDD
        usuarioRepository.save(usuario);
        // envia a la vista de registro un objeto usuario llamado registro
        model.addAttribute("registro", usuario);
        return "viewRegistro";
    }
} else {
    // si las contraseñas no coinciden crea un mensaje y retorna al registro
    model.addAttribute("errorcontrasena", "las contraseñas no coinciden");
    return "registro";
}

<form:errors path="contrasena" cssClass="formuAlerta formuError"></form:errors>
error contraseñas distintas -->
<%
    if (request.getAttribute("errorcontrasena") != null) {
%>
<div class="formuAlerta formuError">${errorcontrasena}</div>
<%
```

Ilustración LXII Error contraseñas personalizado

Demostración:

[Inicio](#) [Rutas](#) [Contacto](#) [Forum](#) [Registro](#)

el tamaño tiene que estar entre 4 y 20

el tamaño tiene que estar entre 4 y 20

no es una dirección de correo bien formada

las contraseñas no coinciden

Ilustración LXIII errores de contraseñas distintas



8. PUESTA EN MARCHA

Una vez que la aplicación entra en fase alfa cubriendo los requisitos funcionales más importantes, se empezaría con las labores de instalación.

Fase1: Servidor Citrix

Se empezaría con la puesta en marcha del servidor de virtualización en este paso es necesario que el equipo tenga instalado el Sistema Operativo Xenserver 7.6.

Desde un equipo cliente en la misma red y a través del software XenCenter se harán las instalaciones de las diferentes máquinas virtuales.

Para una mayor eficacia en las labores de instalación de nuevas VMs el servidor cuenta con un segundo disco duro en el que se guardarán las ISO de los diferentes sistemas operativos. Además, contará con sistema de archivos NTFS y se compartirá mediante el protocolo SMB(Samba) esto ayudará a la hora de realizar las copias de seguridad que se guardarán en el NAS dedicado para tal fin.

Fase2: Instalación Servidor Tomcat / SQL

En la segunda fase se instala la máquina dedicada a servidor de aplicaciones irá montada sobre el S.O UbuntuServer18.04 a la que se le dedicarán 4GB de memoria RAM y 100Gb de disco, para facilitar su gestión y mantenimiento se activará lo primero el protocolo SSH para acceder a la consola de comandos de forma remota ya que por defecto viene deshabilitado.

Una vez se tiene el acceso remoto se empezaría con la instalación de tomcat8.5 y el servidor de SQL.

Fase3: Configuración Servidor Tomcat / SQL

Una vez instalados el servidor de aplicaciones y el de base de datos se crean usuarios nuevos con contraseñas seguras, y se retiran todos los accesos por defecto como pueden ser Admin:Admin.

En este punto también se reconfigurarán todos los puertos por defecto que se necesiten por motivos de seguridad (80,8080,443,22,23)

Fase4: Configuración Aplicación

En cuanto están configurados los puertos y los usuarios es el turno de hacer la configuración para que la aplicación interactúe con la base de datos.

Al tratarse de proyecto que implementa MAVEN estas gestiones se harán a través de un fichero llamado pom.xml que se encarga de la gestión de las dependencias.

Con las dependencias se configuran los drivers para la conexión a BBDD y los repositorios necesarios para las clases JAVA.



Fase5: Despliegue

Una vez funciona la interacción con la base de datos y todas las dependencias están correctamente se despliega la aplicación en el servidor para ello será necesario un fichero `environment.properties` en el que se introducirán los datos del servidor al que se quiere desplegar la aplicación del resto se ocupará el plugins antes citado MAVEN.

Fase6: Comprobación

Una vez el punto 5 está funcionando se hará un reenvío de puertos en el router a la IP y el puerto interno de la aplicación, si accediendo a través de la IP publica del router y un puerto configurado con anterioridad accedemos a la página la prueba será positiva.

Fase7: Publicación

Si los test realizados en el punto 6 son positivos entonces se configura el servicio DynDNS en el router, esto nos evitará tener que mirar la IP pública del router ya que esta cambia cada cierto tiempo al no tener contratada un IP publica estática.



9. PRUEBAS Y CONTROL DE CALIDAD

Nº CASO DE PRUEBA: CP-01		TIPO: RESTRICCIÓN	REQUISITO TESTEADO: RF-01
DESCRIPCIÓN	Comprobar distintos tipos de usuario		
PRERREQUISITOS	debe haber usuarios registrados en la página y un formulario de login		
PROCEDIMIENTO	- Acceder a la página - Revisión contenido - Acceder al login - Comprobar contenido		
RESULTADO ESPERADO	Los usuarios registrados ven opciones que los anónimos no		
RESULTADO OBTENIDO	Se obtiene el resultado esperado		

Nº CASO DE PRUEBA: CP-02		TIPO: RESTRICCIÓN	REQUISITO TESTEADO: RF-02
DESCRIPCIÓN	La herramienta contará con un sistema de login que permita acceder a los usuarios registrados.		
PRERREQUISITOS	debe haber usuarios registrados en la página y un formulario de login		
PROCEDIMIENTO	- Acceder a la página - Acceder al login - Comprobar contenido		
RESULTADO ESPERADO	Los usuarios registrados consiguen acceder y ven opciones que los anónimos no		
RESULTADO OBTENIDO	Se obtiene el resultado esperado		

Nº CASO DE PRUEBA: CP-03		TIPO: FUNCIONAL	REQUISITO TESTEADO: RF-03
DESCRIPCIÓN	La herramienta tendrá una vista principal con un menú que permita el acceso a todos los recursos		
PRERREQUISITOS	debe haber usuarios registrados en la página y un formulario de login		
PROCEDIMIENTO	- Acceder a la página - Comprobar contenido		
RESULTADO ESPERADO	Los usuarios registrados consiguen acceder y ven opciones que los anónimos no		
RESULTADO OBTENIDO	Se obtiene el resultado esperado		



Nº CASO DE PRUEBA: CP-04		TIPO: FUNCIONAL	REQUISITO TESTEADO: RF-04
DESCRIPCIÓN	La herramienta tendrá un mapa de navegación		
PRERREQUISITOS	Debe estar integrado el mapa de google		
PROCEDIMIENTO	a. Acceder a la página b. Comprobar funcionamiento del mapa		
RESULTADO ESPERADO	El mapa permite realizar búsquedas y ver lugares resaltados		
RESULTADO OBTENIDO	Sin resultado , falta de implementar		

Nº CASO DE PRUEBA: CP-05		TIPO: FUNCIONAL	REQUISITO TESTEADO: RF-05
DESCRIPCIÓN	La herramienta tendrá un sistema de puntuaciones que permita votar lugares		
PRERREQUISITOS	Los usuarios deben estar registrados en la página para poder votar Deben de haber rutas subidas por los usuarios		
PROCEDIMIENTO	a. Acceder a la página b. Login c. Acceder al apartado rutas d. Votar una ruta		
RESULTADO ESPERADO	Un usuario puede emitir votos a una ruta subida por otro usuario		
RESULTADO OBTENIDO	Sin resultado , falta de implementar		

Nº CASO DE PRUEBA: CP-06		TIPO: FUNCIONAL	REQUISITO TESTEADO: RF-06
DESCRIPCIÓN	La herramienta tendrá un apartado con enlaces relacionados		
PRERREQUISITOS	Debe existir un menú principal y un apartado relacionados		
PROCEDIMIENTO	a. Acceder a la página b. Comprobar funcionamiento de los menús y el apartado + Misterio c. Revisión de hipervínculos		
RESULTADO ESPERADO	Desde este menú se accede a todos los contenidos		
RESULTADO OBTENIDO	Se obtiene el resultado esperado		



Nº CASO DE PRUEBA: CP-07		TIPO: FUNCIONAL	REQUISITO TESTEADO: RF-07
DESCRIPCIÓN	La herramienta contará con un apartado para calendario de eventos.		
PRERREQUISITOS	En la pantalla inicial debe haber un calendario		
PROCEDIMIENTO	- Acceder a la página - Ir a la página principal - Pulsar calendario - Comprobar el calendario		
RESULTADO ESPERADO	Se debe mostrar el calendario y sus eventos		
RESULTADO OBTENIDO	Se accede al calendario pero no tiene implementado eventos		

Nº CASO DE PRUEBA: CP-08		TIPO: FUNCIONAL	REQUISITO TESTEADO: RF-08
DESCRIPCIÓN	La herramienta contará con un servicio de enlaces a compras y reservas.		
PRERREQUISITOS	La página debe tener un apartado rutas funcionales Estas deben incorporar un enlace a hoteles cercanos		
PROCEDIMIENTO	- Acceder a la página - Ir a la página principal - Hacer una búsqueda en el mapa - En un lugar obtenido apretar el hipervínculo		
RESULTADO ESPERADO	La página nos redirige a la página de contratación del hotel		
RESULTADO OBTENIDO	Sin resultados. No implementado		

Nº CASO DE PRUEBA: CP-09		TIPO: FUNCIONAL	REQUISITO TESTEADO: RF-09
DESCRIPCIÓN	La página tendrá una vista que permita añadir contenidos los usuarios registrados.		
PRERREQUISITOS	El usuario debe estar logeado		
PROCEDIMIENTO	- Acceder a la página - Login - Pestaña subir contenido - Añadir el contenido		
RESULTADO ESPERADO	El usuario puede añadir contenidos correctamente		
RESULTADO OBTENIDO	Sin resultados. No implementado		



Nº CASO DE PRUEBA: CP-10		TIPO: FUNCIONAL	REQUISITO TESTEADO: RF-10
DESCRIPCIÓN	La página tendrá un foro para que interactúen los usuarios.		
PRERREQUISITOS	El usuario debe estar logeado		
PROCEDIMIENTO	a. Acceder a la página b. Login c. Pestaña foro d. comprobar contenido		
RESULTADO ESPERADO	El usuario ve los contenidos de foros correctamente		
RESULTADO OBTENIDO	Sin resultados. No implementado		

Nº CASO DE PRUEBA: CP-11		TIPO: FUNCIONAL	REQUISITO TESTEADO: RF-11
DESCRIPCIÓN	Habrá posibilidad de cambiar la contraseña para los usuarios registrados		
PRERREQUISITOS	El usuario debe estar logeado		
PROCEDIMIENTO	a. Acceder a la página b. Login c. Pestaña perfil d. Opción cambiar contraseña		
RESULTADO ESPERADO	El usuario cambia la contraseña correctamente		
RESULTADO OBTENIDO	Sin resultados. No implementado		

Nº CASO DE PRUEBA: CP-12		TIPO: FUNCIONAL	REQUISITO TESTEADO: RF-12
DESCRIPCIÓN	El usuario será capaz de consultar los datos de su perfil		
PRERREQUISITOS	El usuario debe estar logeado		
PROCEDIMIENTO	a. Acceder a la página b. Login c. Pestaña perfil		
RESULTADO ESPERADO	El usuario ve sus datos correctamente		
RESULTADO OBTENIDO	Sin resultados. No implementado		

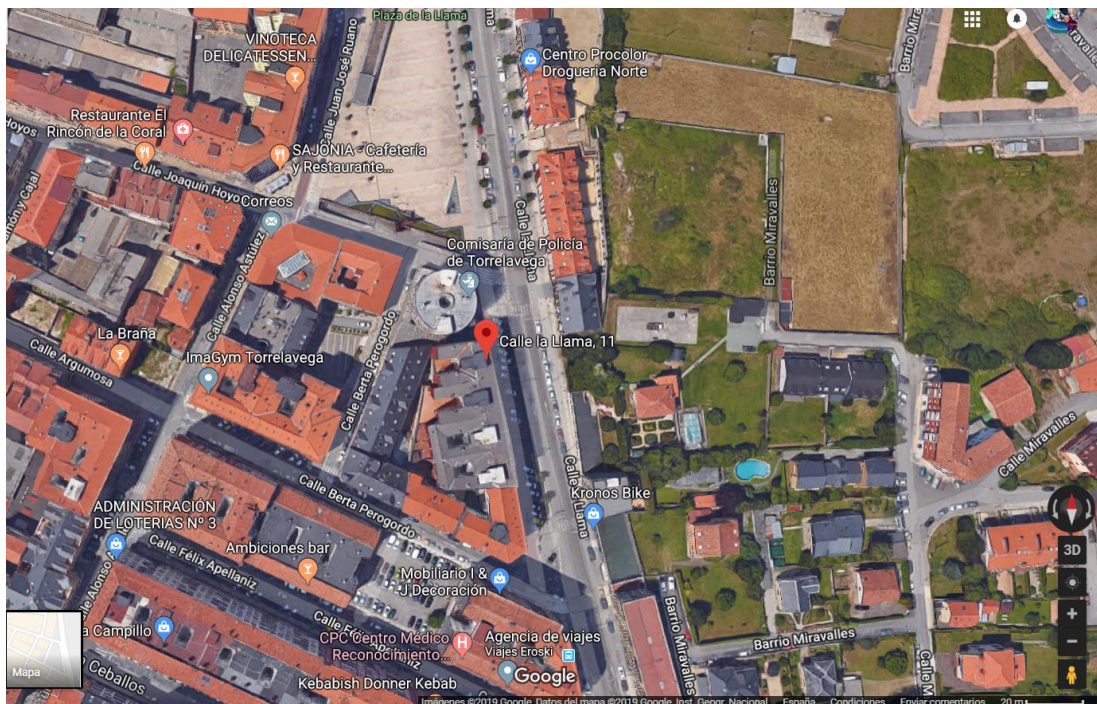


10. PLAN EMPRESARIAL

10.1 Localización

La situación inicial será en el propio domicilio en Torrelavega.

En cuanto el proyecto empiece a ser rentable o hubiera problemas de espacio se buscaría otra opción para instalar los servidores.





10.2 Inmovilizado intangible.

El artículo 101 LPI establece que “Los derechos sobre los programas de ordenador, así como sobre sus sucesivas versiones y los programas derivados, podrán ser objeto de inscripción en el Registro de la Propiedad Intelectual.

El procedimiento será por el registro normal. En este caso, tanto la autoría, como la titularidad, le corresponde a la empresa generadora del software. Solicitud de marca ... 125, 36 ...

Certificado ... 14,81....

TOTAL 140, 17

10.3 Propiedad industrial



SIGNOS DISTINTIVOS.		TRÁMITE O PAGO NO ELECTRÓNICOS		TRÁMITE Y PAGO ELECTRÓNICOS	
CONCEPTO		Clave	Importe	Clave	Importe
SOLICITUD DE MARCA O NOMBRE COMERCIAL	1ª Clase	MT01	147,49 €	ME01 (*)	125,36 €
	2ª Clase y cada sucesiva		95,55 €		81,21 €
Clase no pagada con la solicitud de marca o nombre comercial.		MT17	95,55 €	ME17 (*)	81,21 €
SOLICITUD DE MARCA COLECTIVA O DE GARANTÍA	1ª Clase	MT02	294,94 €	ME02 (*)	250,70 €
	2ª Clase y cada sucesiva		191,07 €		162,41 €
Clase no pagada con la solicitud de marca colectiva o de gar.		MT18	191,07 €	ME18 (*)	162,41 €
SOLICITUD DE REGISTRO INTERNACIONAL. Sólo comprende la tasa nacional, con independencia de las que hayan de abonarse a la Of. Int.		MT03	43,98 €	ME06	37,38 €
DIVISIÓN SOLICITUD. Por cada solicitud divisional resultante.		MT05	56,06 €	ME03	47,65 €
SOLICITUD DE RESOLUCIÓN URGENTE		MT07	54,26 €	ME08	46,12 €
PRIORIDAD UNIONISTA O EXPOSICIÓN. Por cada prior. reivindicada		MT08	20,25 €	ME09	17,21 €
INSCRIPCIÓN O CANCELACION DE CESIONES, LICENCIAS, DERECHOS REALES, OPCIONES COMPRA. Por registro afectado.		MT15	34,43 €	ME10	29,27 €
PAGO MÁXIMO.		MX15	7.184,75 €	XM10	6.107,04 €
CERTIFICACIONES		MT20	17,41 €	ME11	14,81 €



10.4 Aplicaciones informáticas

La mayoría de programas o sistemas utilizados serán libres o gratuitos
el resto se enseñan en la tabla más abajo mostrada.

BIEN	CARACTERISTICAS	AÑO	PRECIO	IVA
Avast Internet Security Premier	Antivirus	1	69.99	21%
Windows 10 pro	Sistema operativo	1	259	21%
Sublime text 3 bussines	Editor de texto avanzado	1	70.31	21%
TOTAL			399.3	21%



10.5 Inmovilizado material

10.5.1 Terrenos

No requeridos

10.5.2 Edificaciones y Construcciones

No se requiere el alquiler de locales para este proyecto, puesto que será desarrollado desde el propio domicilio.

10.5.3. Instalaciones

Se requerirá de la instalación de una segunda línea de internet con movistar



10.5.4. Maquinaria y utillaje

De la instalación de la red y los equipos que la componen se encargará Iván Casado y el material necesario será el siguiente.

BIEN	CARACTERISTICAS	AÑO	PRECIO	IVA
Crimpadora		1	9.59	21%
Tijera electricista		1	8.90	21%
pelacables		1	10.19	21%
100 m Cable UTP		1	38.52	21%
Tester Rj45		1	19.09	21%
Armario Rack 9U		1	77.99 €	21%
TOTAL			163.76€	21%



10.5.5. Mobiliario

Como material de oficina se va a necesitar:

BIEN	CARACTERISTICAS	AÑO	PRECIO	IVA
Mobiliario de oficina	1 x Mesa 1 x Silla 1 x TV	1	109.00 € 128.75 € 540.00 €	21%
TOTAL			777.75€	21%

10.5.6. Equipos proceso de la información

Los equipos que componen la red local son los siguientes:

BIEN	CARACTERISTICAS	AÑOS	PRECIO	IVA
PC Desarrollo	MSI GL62M 7RDX	1	0 €	21%
PC Servidor	PowerEdge R440 8x2.5" 4110 1x16GB 2x600GB	1	1.785,73 €	21%
Monitor	Acer Nitro VG220Q LED Display 54,6 cm (21.5")	1	129,66 €	21%
NAS de backups	Synology DS218PLAY Diskstation - NAS con 2 bahías	1	246,87	21%
Router	Mikrotik Routerboard RB4011iGS+RM	1	188,59	21%
Switch	Zyxel GS1100-16 16 puertos	1	57,45	21%
Disco duro	WD Blue 1 TB (7200 rpm, SATA a 6 Gb/s, 64 MB de caché, 3,5") azul	1	46,52 x 2 unidades	21%
TOTAL			2501,34 €	21%

10.5.7. Elementos de transporte

No requerido

10.5.8. Otro inmovilizado material

No requerido



10.6 Presupuesto detallado

10.6.1 Inversión

Para el desarrollo del proyecto se necesitará el uso de los equipos informáticos detallados en el punto 10.5.6 mostrando tanto los equipos servidor como los elementos que conforman la red local, además se necesitará hacer inversión en software para desarrollo que se detalla en el punto 10.4 de la memoria.

10.6.2 Gastos

Los gastos en servicios para el desarrollo del proyecto son los siguientes:

SERVICIO	MES	PRECIO	IVA
2ª salida de internet	3	99	21%
TOTAL		297	21%

Horas dedicadas al proyecto:

ESTIMACION DE TIEMPO	PERIODO 3 MESES
Días trabajados al mes	30
Días totales de proyecto	90
TOTAL HORAS (90días x 3h)	270

10.6.3 Umbral de rentabilidad

UMBRAL		3 MESES
Gastos		297€
Salario mínimo mensual		2700€
Salario mínimo interprofesional		2207,7€
Umbral	Gastos + salario mensual	2997€
Valor hora mínimo	Umbral/Horas totales	6,3€



Tiempo empleado	
Introducción	2h
Objetivos	2h
Análisis del contexto y estado del arte	8h
Análisis de requisitos	8h
Diagrama de red	1h
Diagrama entidad/relación	4h
Diagrama clases	4h
Diagrama de interfaces	20h
Diseño de procedimientos	8h
Planificación	4h
Formación	50h
Desarrollo	105h
Puesta en marcha	4h
Explotación	4h
Pruebas	5h
Control calidad	4h
Gestión económica	6h
Documentación	30h
TOTAL HORAS	270h
COSTE TOTAL (6,3€ * HORAS)	1701€

Por tanto, si el valor de hora mínimo es 6,3€ hora, un precio a partir de 1701€ sería rentable.



11. PLAN ECONÓMICO FINANCIERO

11.1. Plan Económico

INVERSIONES	% AMORT	AÑO 1	AÑO 2	AÑO 3
Inmovilizado Material				
Terrenos				
Instalaciones y Maquinaria	10%			
Mobiliario y Utillaje	10%	941,51		
Elementos de transporte	15%			
Equipo proceso de información	25%	2.501,34		
Otro inmovilizado material	10%			
Total Inmovilizado Material		3.442,85	0,00	0,00
Inmovilizado Intangible				
Investigación y Desarrollo	20%			
Patentes, licencias y marcas		92,81		
Aplicaciones informáticas	26%	399,30		
Derechos de traspaso				
Canon franquicia				
Total Inmovilizado Intangible		492,11	0,00	0,00
Inversiones Financieras		0,00	0,00	0,00
Amortización Inmovilizado Material		719,49	719,49	719,49
Amortización Inmovilizado Intangible		103,82	103,82	103,82
Fianzas y Depósitos			0,00	0,00

Ilustración LXIV Inversión



11.2. Ingresos

Publicidad

Mediante Google adsense

Formula 1

$\text{Ingresos} = \text{Páginas vistas} \times \text{Número de anuncios} \times 0.5 / 1000$

✓ Páginas vistas - Total de páginas leídas por los usuarios.

✓ Numero de anuncios - Mediante Adsense se pueden mostrar un máximo de tres bloques de anuncios en cada página.

Si el artículo tiene suficiente texto puedes agregar hasta tres bloques de enlaces adicionales (como puedes ver en esta página), que también generan ingresos aunque no de la misma cuantía que los anuncios.

✓ Promedio CPM - AdSense paga de acuerdo a los clics hechos por los lectores en los anuncios, sistema llamado PPC (Pago Por Clic) y a las impresiones o CPM (Costo por Mil), el valor tomado es un promedio.

Ilustración LXV Fórmula Google Adsense

$20\,000 \times 4 \times 0.5 / 1000 = 40\text{€}$ mes primer año

$40\,000 \times 4 \times 0.5 / 1000 = 80\text{€}$ mes segundo año

$80\,000 \times 4 \times 0.5 / 1000 = 160\text{€}$ mes tercer año

Reserva en hoteles

Un porcentaje de las ventas realizadas a través de la plataforma se obtendrá un beneficio del 5 %.

Se calcula con una media de 60 euros la noche y en 17 comunidades autónomas.

$15 \times 60 \times 0,05$ (comisión) $\times 12 \times 17 = 765\text{€}$ mes el primer año. 9180 anuales.

$30 \times 60 \times 0,05 \times 12$ meses $\times 17 = 1530\text{€}$ mes el segundo año. 18360 anuales.

$50 \times 60 \times 0,05 \times 12 \times 17$ comunidades $= 2550\text{€}$ mes el tercer año. 30600 anuales

VENTAS E INGRESOS	AÑO 1	AÑO 2	AÑO 3
Ventas (Productos - Servicios)			
Publicidad	480,00	960,00	1.920,00
Comision por venta turistica	9.180,00	18.360,00	30.600,00
Total ventas	9.660,00	19.320,00	32.520,00
Otros Ingresos			
Otros Ingresos de explotación			
Ingresos financieros			
Ingresos excepcionales			
Imputación de Subven. Inversión	0,00	0,00	0,00
Total Ingresos	0,00	0,00	0,00
TOTAL VENTAS E INGRESOS	9.660,00	19.320,00	32.520,00

Ilustración LXVI Ventas e ingresos



11.3. Compras y gastos

COMPRAS Y GASTOS	AÑO 1	AÑO 2	AÑO 3
Compras			
Total Compras	0,00	0,00	0,00
Servicios exteriores			
Arrendamientos			
Reparación y Conservación			
Servicios Profesionales			
Transportes			
Primas de Seguros			
Servicios bancarios y similares			
Publicidad y Promoción			
Suministros	1.188,00	1.188,00	1.188,00
Gastos de Viaje			
Tributos			
Gastos establecimiento	800,00		
Licencias y altas suministros			
Otros Gastos			
Total servicios exteriores	1.988,00	1.188,00	1.188,00
Gastos excepcionales			

Ilustración LXVII Gastos y Compras

11.4. Gastos de personal

PERSONAL	AÑO 1	AÑO 2	AÑO 3
Sueldos y Salarios	12.600,00	12.978,00	13.367,61
Seguridad Social	720,00	1.189,86	2.379,72
Otros gastos de personal			
Gastos de personal	13.320,00	14.167,86	15.747,33

Ilustración LXVIII Gastos Personal



11.5 Gastos Financieros

FINANCIACIÓN	AÑO 1	AÑO 2	AÑO 3
FINANCIACIÓN PROPIA			
<i>Aportaciones dinerarias</i>	5.000,00		
<i>Subvenciones no reintegrables</i>			
<i>Subvenciones a la inversión</i>			
FINANCIACIÓN AJENA			
<i>Préstamos Bancarios Largo Plazo</i>			
<i>Cuantía del préstamo</i>			
<i>Tipo de interés %</i>			
<i>Plazo en años</i>			
<i>Préstamos Bancarios Corto Plazo</i>			
<i>Cuantía del préstamo</i>			
<i>Tipo de interés %</i>			
<i>Plazo en meses</i>			
Gastos financieros	0,00	0,00	0,00

Ilustración LXIX Gastos financieros



11.6 Cuenta Explotación Previsional

PÉRDIDAS Y GANANCIAS		AÑO 1	AÑO 2	AÑO 3
		Mar 19 - Feb 20	Mar 20 - Feb 21	Mar 21 - Feb 22
Importe cifra de negocio				
	<i>Importe cifra de ventas</i>	9.660,00	19.320,00	32.520,00
Variación de existencias				
	<i>Variación de existencias</i>	0,00	0,00	0,00
Aprovisionamientos				
	<i>Consumo de mercaderías</i>	0,00	0,00	0,00
Otros ingresos de explotación				
	<i>Otros ingresos</i>	0,00	0,00	0,00
	<i>Subvenciones de explotación</i>	0,00	0,00	0,00
Gastos de personal				
	<i>Gastos de personal</i>	-13.320,00	-14.167,86	-15.747,33
Otros gastos de explotación				
	<i>Amendamientos</i>	0,00	0,00	0,00
	<i>Reparación y conservación</i>	0,00	0,00	0,00
	<i>Servicios profesionales</i>	0,00	0,00	0,00
	<i>Transportes</i>	0,00	0,00	0,00
	<i>Primas de seguros</i>	0,00	0,00	0,00
	<i>Servicios bancarios y similares</i>	0,00	0,00	0,00
	<i>Publicidad y promoción</i>	0,00	0,00	0,00
	<i>Suministros</i>	-1.188,00	-1.188,00	-1.188,00
	<i>Gastos de viaje</i>	0,00	0,00	0,00
	<i>Tributos</i>	0,00	0,00	0,00
	<i>Gastos establecimiento</i>	0,00	0,00	0,00
	<i>Licencias y otros suministros</i>	0,00	0,00	0,00
	<i>Otros gastos</i>	0,00	0,00	0,00
Amortización del inmovilizado				
	<i>Amortización del inmovilizado</i>	-761,79	-761,79	-761,79
Imputación de subvenciones de inmovilizado no finar				
	<i>Subvenciones a la inversión</i>	0,00	0,00	0,00
Otros resultados				
	<i>Ingresos Excepcionales</i>	0,00	0,00	0,00
	<i>Gastos excepcionales</i>	0,00	0,00	0,00
RESULTADO DE EXPLOTACIÓN		-5.609,79	3.202,35	14.822,88
	<i>Ingresos financieros</i>	0,00	0,00	0,00
	<i>Gastos financieros</i>	0,00	0,00	0,00
RESULTADO FINANCIERO		0,00	0,00	0,00
RESULTADO ANTES DE IMPUESTOS		-5.609,79	3.202,35	14.822,88
	<i>Impuesto sobre beneficios</i>	0,00	-960,70	-4.446,86
BENEFICIO O PERDIDA DESPUÉS DE IMPUESTOS		-5.609,79	2.241,64	10.376,01

Ilustración LXX Explotación previsional



11.7 Balance previsional

ACTIVO	AÑO 1	AÑO 2	AÑO 3
	Mar 19 - Feb 20	Mar 20 - Feb 21	Mar 21 - Feb 22
ACTIVO NO CORRIENTE	2.927,13	2.165,33	1.403,54
Inmovilizado Intangible			0,00
Investigación y Desarrollo	0,00	0,00	0,00
Patentes, licencias y marcas	92,81	92,81	92,81
Aplicaciones informáticas	399,30	399,30	399,30
Derechos de traspaso	0,00	0,00	0,00
Canon franquicia	0,00	0,00	0,00
Amortización acumulada Inmov. Intang.	-103,82	-207,64	-311,45
Inmovilizado Material			
Terrenos	0,00	0,00	0,00
Instalaciones y Maquinaria	0,00	0,00	0,00
Mobiliario y Utillaje	941,51	941,51	941,51
Elementos de transporte	0,00	0,00	0,00
Equipo proceso de información	2.255,30	2.255,30	2.255,30
Otro inmovilizado material	0,00	0,00	0,00
Amortización acumulada Inmov. Material	-657,98	-1.315,95	-1.973,93
Inversiones financieras	0,00	0,00	0,00
Fianzas - depósitos L/P	0,00	0,00	0,00
ACTIVO CORRIENTE	-2.753,31	1.964,98	17.395,71
Existencias	0,00	0,00	0,00
Clientes	0,00	0,00	0,00
Deudores y otros	0,00	0,00	0,00
Hacienda deudora	0,00	0,00	0,00
Tesorería	0,00	0,00	0,00
Otros activos corrientes/Recursos libre disp.	-2.753,31	1.964,98	17.395,71
TOTAL ACTIVO	173,82	4.130,31	18.799,25

Ilustración LXXI Balance previsional



11.8 Análisis Económico

11.8.1 Ratios de rentabilidad

$$\text{Rentabilidad Económica R.O.A (\%)} = \frac{\text{Resultado de explotación}}{\text{Activo total}} \times 100$$

AÑO 1: $\text{Rentabilidad económica R.O.A (\%)} = \frac{-5609,79}{173,82} \times 100 = -3227,27\%$

AÑO 2: $\text{Rentabilidad económica R.O.A (\%)} = \frac{3202,34}{4130,31} \times 100 = 77,53\%$

AÑO 3: $\text{Rentabilidad económica R.O.A (\%)} = \frac{14822,88}{18799,25} \times 100 = 78,84\%$

$$\text{Rentabilidad Financiera R.O.E. (\%)} = \frac{\text{Resultado antes de impuestos}}{\text{Fondos propios}} \times 100$$

AÑO 1: $\text{Rentabilidad financiera R.O.E. (\%)} = \frac{-5609,79}{-609,79} \times 100 = 919,95 \%$

AÑO 2: $\text{Rentabilidad financiera R.O.E. (\%)} = \frac{3202,35}{1631,85} \times 100 = 196,24\%$

AÑO 3: $\text{Rentabilidad financiera R.O.E. (\%)} = \frac{14822,88}{12007,86} \times 100 = 123,44 \%$

RENTABILIDAD	Año1	Año2	Año3
Resultado de explotación	-5609,79	3202,35	14822,01
Activo Total	173,82	4130,31	18799,25
Rentabilidad económica	-127,56%	54,27%	55,19%
Resultado antes de impuestos	-5609,79	3202,35	14822,01
Fondos propios	-609,79	1631,85	12007,86
Rentabilidad financiera	919,95%	196,95%	123,44 %



11.9 Análisis Financiero

11.9.1 Ratios liquidez

Fondo de Maniobra (1er Año)

Fondo de Maniobra = Activo Corriente – Pasivo Corriente =
-2.753,31 -609,79 = -3363,1

Fondo de Maniobra (2º Año)

Fondo de Maniobra = 1.964,98-1.631,85 =336,13

Fondo de Maniobra (3er Año)

Fondo de Maniobra = 17.395,71– 12.007,86= 5388,05

11.10 Necesidades Forma Jurídica

11.10.1 Necesidades

NECESIDADES	AÑO 1
INMOVILIZADO MATERIAL	3196,81 €
INMOVILIZADO INTANGIBLE	492,11 €
TOTAL COMPRAS	0€
TOTAL SERVICIOS EXTERIORES	1.188,00€
GASTOS DE PERSONAL	13.320 €
TOTAL	18196,81€



11.10.2 Financiación

No se requiere de financiación ajena.

11.10.3 Forma jurídica

Se elige como forma jurídica la Sociedad Limitada (S.L.) ya que puede ser unipersonal y se necesita un capital mínimo de 3.005 € que es desembolsado con la fundación.

Capital está dividido en participaciones, iguales, indivisibles y acumulables y el Impuesto de Sociedades es del 20% cuando los beneficios son inferiores a 120.000 €.

NOMBRE DE LA EMPRESA	MISTERY TRACKS SPAIN S.L.		
TIPO DE EMPRESA	SOCIEDAD LIMITADA		
ACTIVIDAD	DESARROLLO DE APLICACIONES WEB		
DIRECCIÓN	LA LLAMA 11 6G		
POBLACIÓN	TORRELAVEGA (CANTABRIA)		
INICIO DE ACTIVIDAD (Fecha)	13/03/2019		
TIPO MEDIO DE IVA (%)	21,00%	RESERVAS Y BENEFICIOS (%)	100,00%
TIPO MEDIO RETENCIÓN A EMPLEADOS (%)	15,00%	DÍAS DE COBRO VENTAS	
PORCENTAJE IMPUESTO (%)	30,00%	DÍAS DE PAGO	

Ilustración LXXII Datos de la empresa



12. Bibliografía

Fuentes Consultadas

Referencia	Descripción	Fuente
Ref 1	Despoblación en España	https://www.epdata.es/datos/despoblacion-espana-datos-estadisticas/282
Ref 2	Listas ivoox	https://www.ivoox.com/top100_hb.html
Ref 3	Análisis tendencias	https://www.hosteltur.com/109383_las-tendencias-de-viajes-que-los-epanoles-seguiran-en-2019.html
Ref 4	Competencia	https://www.dondeviajamos.com/rutas/ruta-misterio/ https://rutasmisteriosas.es/ https://blog.gluubo.com/ver/turismo-de-misterio/
Ref 5	Turismo oscuro	https://zaguan.unizar.es/record/32279/files/TAZ-TFG-2015-3144.pdf PÁGINA 5
Ref 6	Soluciones despoblación	https://www.eleconomista.es/evasion/noticias/9483314/10/18/La-despoblacion-rural-un-problema-que-tienen-las-ciudades.html
Ref 7	Perfiles de usuarios	https://www.javiersierra.com/biografia/periodismo/mas-alla/
Ref 8	IDTR	https://www.uclan.ac.uk/research/explore/groups/institute_for_dark_tourism_research.php
Ref 8	Leyes	http://www.ayalga.com/paginas-web-asturias/Las-leyes-que-afectan-a-las-paginas-Web-53.php



Conclusiones

Ha sido un camino muy duro y muy largo, me ha hecho pasar por todos los estados de ánimo conocidos: alegría, tristeza, enfado....

He aprendido muchas cosas nuevas y reforzar otras que ya sabía en la profesión que he elegido y que más me gusta.

Por lo tanto, solo me queda decir GRACIAS A TODOS mis profesores, incluso a aquellos con los que no tenía afinidad porque de todos ellos he sacado cosas positivas. De unos por lo en clase aprendido, de otros porque he aprendido a ser autosuficiente y a ganarme el pan con el sudor de mi frente.

Lo único que haría de forma distinta en el proyecto sería los frameworks y lenguajes escogidos, de haberlo hecho de otra forma podría mostrar un trabajo mucho más desarrollado, pero me quedo con la experiencia de haber sido capaz de poder mostrar algo elaborado con cosas totalmente nuevas para mí y que me reforzará de cara al futuro.

Salgo del curso orgulloso de mi mismo y cumpliendo las metas marcadas, y con empleo que era el motivo principal de hacer este grado.

GRACIAS!!!!!!

Iván Casado Gutierrez