



Proyecto Urbex

Catalina Requena Ferrario
Yiyo Zapata Martínez
Agustín Tuduri Marín
CIFP Francesc de Borja Moll
1º DAW 2021-2022

Resumen	3
Introducción	3
Objetivos	3
Alcance	4
Diagrama inicial	4
Mínimo exigible	4
Secciones de la aplicación	5
Ampliación	6
Exclusiones	6
Metodología	7
Fases de desarrollo	7
Estudio de Viabilidad	7
Análisis	7
Diseño	7
Desarrollo	8
Implantación	8
Metodología en cascada	8
Por qué usar metodología en cascada	9
Planificación	10
Diagrama de Gantt	10
Análisis	10
Antecedentes	10
Partes interesadas	10
Requisitos de la aplicación	10
Requisitos funcionales	10
Requisitos no funcionales	11
Requisitos de sistema	11
Diagrama de casos de uso	12
Documento detallado de requisitos.	13
Pruebas de aceptación de requisitos(alfa, beta).	13
Listado de posibles tecnologías a utilizar para cumplir con todos los requisitos del proyecto.	13
Selección de tecnologías mediante la matriz Requisitos/Tecnologías.	13
Modelo Entidad Relación (MER).	13
Diseño	13
Modelo conceptual del proyecto (ampliar el de la introducción con las tecnologías seleccionadas en la fase de Análisis).	13
Diagrama de clases	14
Diagrama de secuencia	14
Paso a relacional del MER y normalización.	14
Planificación y diseño de los casos de pruebas. Tipos de pruebas, ...	15
Diseño del esquema de la web	

Indice de diagramas

D1- Diagrama inicial.	4
D2 - Diagrama de Gantt.	10
D3- Diagrama de casos de uso.	12
D4- Modelo Entidad Relación (MER).	13
D5- Diagrama de clases.	14
D6- Diagrama de secuencia.	

Resumen

En este documento se encontrarán todos los puntos que hacen referencia al correcto desarrollo del proyecto como el objetivo principal del desarrollo, requisitos, análisis, arquitectura del sistema, diseño, implementación, pruebas, la gestión y la conclusión final.

La aplicación permitirá categorizar los lugares, comentarios, el conocimiento de su ubicación, la accesibilidad, el nivel de seguridad y deterioro de estos sitios, así como una guía y reglas sobre la realización de este tipo de actividad.

Introducción

Objetivos

El objetivo de Urbex es el desarrollo de una aplicación web para la exploración en común de lugares abandonados en la que los usuarios tendrán que registrarse para poder acceder a la información, reglas, comentarios, imágenes y propias publicaciones de lugares a explorar, y otros usuarios que harán de moderador a las publicaciones del resto de la comunidad.

Para ello se necesitará un sistema de cuentas con formulario en nuestra app web donde los usuarios puedan registrarse en ella.

En cada publicación los usuarios podrán acceder a la localización, imágenes y las reglas para la visita de estos lugares.

Si un usuario desea publicar un lugar o eliminarlo, enviará una petición a moderación; estas peticiones tendrán que pasar primero por el moderador y comprobar si se ajusta a las reglas de la aplicación.

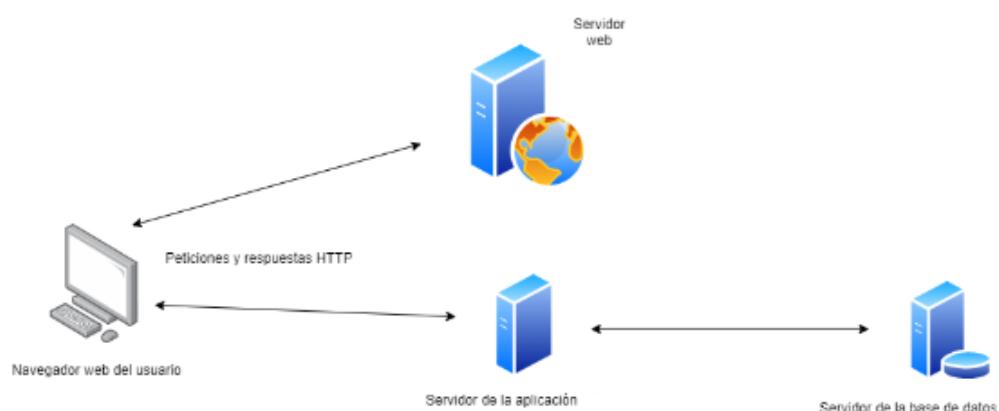
La aplicación contará con un número máximo de usuarios y en el momento en el que se llegue a este límite, no se permitirá la creación de nuevos usuarios.

Alcance

El proyecto se pretende desarrollar en un plazo de 3 meses con fecha de entrega el día 06/06/2022 con la finalidad de satisfacer las necesidades y cumplir las expectativas ofrecidas.

Lo que se quiere proyectar es una aplicación web donde existen usuarios que pueden visitar, comentar y compartir lugares de su interés. Los posibles riesgos que hay en el desarrollo del proyecto pueden ser el corto plazo de tiempo , y la poca experiencia de los desarrolladores.

Diagrama inicial



Metodología

Fases de desarrollo

Todo desarrollo software está dividido en unas fases. Estos son: Estudio de viabilidad, Análisis, Diseño y Desarrollo e implantación del sistema.

Estudio de Viabilidad

En esta fase analizaremos las necesidades y propondremos una solución a corto plazo. Los criterios que se harán en esta propuesta no serán estratégicos sino tácticos, y relacionados con aspectos económicos, técnicos, legales y operativos.

Los resultados derivados de esta fase constituyen la base para tomar la decisión de seguir adelante o abandonar el proyecto.

Analisis

Conseguiremos una especificación detallada del sistema de información, por medio de un catálogo de requisitos y modelos (casos de uso y de clases).

- 1- Descripción del sistema de información a partir del estudio de viabilidad.
- 2- Se recogen de forma detallada los requisitos funcionales y no funcionales (restricciones, seguridad...)
- 3-Definición de interfaces de usuario.

Diseño

Definición de la arquitectura del sistema y del entorno tecnológico, junto con la especificación detallada de los componentes del sistema de información.

- Especificación técnica del plan de pruebas
 - Definición de requisitos de implantación.
 - Diseño de procedimientos de migración y carga inicial.
-
- Los componentes del sistema (módulos o clases, según el caso) y de las estructuras de datos.
 - Los procedimientos de migración y sus componentes asociados.
 - La definición y revisión del plan de pruebas, y el diseño de las verificaciones de los niveles de prueba establecidos.
 - El catálogo de excepciones, que permite establecer un conjunto de verificaciones relacionadas con el propio diseño o con la arquitectura del sistema.
 - La especificación de los requisitos de implantación.

Desarrollo

Construcción y prueba de los diferentes componentes del sistema de información a partir de su conjunto de especificaciones lógicas y físicas obtenidas en la fase de diseño.

- Se desarrollan procedimientos de operación y de seguridad.
- Elaboración de manuales de usuario final y explotación.

Implantación

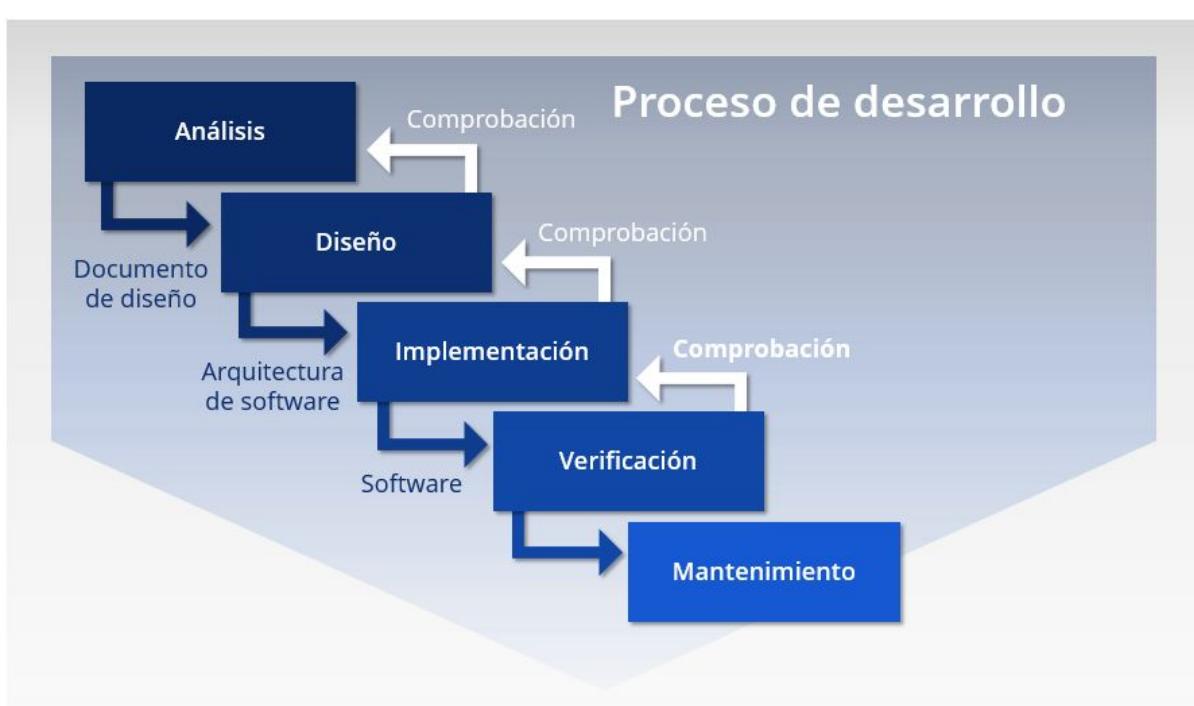
Este proceso tiene como objetivo principal la entrega y la aceptación del sistema en su totalidad.

Implantación de la
plataforma.

Implantación en la red.
Realización de las operaciones.
Visualización de la plataforma.

Metodología en cascada

En este proyecto se utilizará la metodología cascada, un tipo de metodología secuencial que gestiona el proyecto dividiéndolo en fases. Cada fase se comenzará cuando la previa haya finalizado.



Consiste en ordenar de forma lineal las diferentes etapas que debe seguir un equipo a la hora de desarrollar un software. Es un proceso que permite que puedas relacionar cada una de las etapas del modelo en cascada con la anterior, de esta manera, considera los elementos que debes quitar o añadir en la fase siguiente.

1- Definición del problema:

Especificación de requisitos.

Análisis

Recopilación y documentación de los requisitos.

En esta etapa recopilaremos información integral para definir qué requiere el proyecto.

Esta información se puede recoger de formas diferentes. En este caso hemos usado entrevistas y lluvia de ideas, entre otros.

Al final de esta fase, los requisitos del proyecto deben quedar claros.

2- Desarrollo:

Diseño

Una vez definidos los requisitos, el equipo diseña el sistema. No hay ninguna codificación durante esta frase, pero el equipo establece especificaciones (lenguaje de programación, requisitos hardware...)

Implementación.

En esta fase se codifica. Se toma toda la información de la etapa anterior y se crea un producto funcional.

Se implementa el código en pequeñas porciones, las cuales se integrarán al final de esta fase o al principio de la siguiente.

Pruebas

Una vez que está listo todo el código, pueden empezar las pruebas del producto. Los encargados de las pruebas encuentran los problemas y los informan metódicamente. Si surgen problemas graves, tu proyecto tal vez deba regresar a la fase uno para ser evaluado nuevamente.

3- Mantenimiento:

Instalación y mantenimiento

En esta fase, el producto está completo y tu equipo presenta los entregables que deberán instalarse o lanzarse

El producto se ha entregado al cliente y está en uso. A medida que surjan problemas, tal vez tu equipo necesitará crear parches y actualizaciones para solucionarlos. Nuevamente, si surgen problemas graves, es posible que debas regresar a la fase uno.

Por qué usar metodología en cascada

Usa una estructura clara.

Determina el objetivo final de forma clara.

Transmite bien la información.

Muestra el estado del avance.

Hace que el proyecto sea sencillo de gestionar.

Ahorra tiempo y dinero.

Diagrama de Gantt

	INICIO	FINAL	08/04	11/04	12/04	13/04	14/04	15/04	18/04	19/04	20/04	21/04	22/04	25/04	26/04	27/04	28/04	29/04	02/05	03/05	05/05	11/05	13/05	17/05	27/05
ANALISIS:																									
Partes interesadas	08/04/2022	12/04/2022																							
Requisitos funcionales	13/04/2022	15/04/2022																							
Req. no funcionales	13/04/2022	15/04/2022																							
Req. de sistema	15/04/2022	18/04/2022																							
Diagrama casos de uso	19/04/2022	19/04/2022																							
Detalles de requisitos	20/04/2022	22/04/2022																							
Modelo entidad relación	22/04/2022	22/04/2022																							
Lista de tecnologías	25/04/2022	25/04/2022																							
Lista de tecnologías (matriz)	25/04/2022	25/04/2022																							
DISEÑO:																									
Modelo conceptual	26/04/2022	26/04/2022																							
Diagrama de clases	27/04/2022	29/04/2022																							
Diagrama de Secuencias	28/04/2022	02/05/2022																							
Relacional a normalización	02/05/2022	02/05/2022																							
Planificación, diseño y pruebas	27/04/2022	03/05																							
DESARROLLO:																									
Construcción del SI	05/05/2022	10/05/2022																							
Proc. de operaciones	11/05/2022	13/05/2022																							
Proc. de seguridad	11/05/2022	13/05/2022																							
Elaboración de manuales UF	13/05/2022	17/05/2022																							
IMPLEMENTACIÓN:																									
Codificación	20/05/2022	27/05/2022																							
Implementación del código	24/05/2022	27/05/2022																							

Análisis

Antecedentes

Hoy en día son muchos los jóvenes que buscan lugares especiales donde realizar actividades varias, y si bien la información va de boca en boca, vemos necesaria una aplicación donde poder almacenarla y compartirla.

La aplicación X tiene como objetivo facilitar la comunicación y la transacción de información entre los usuarios interesados en encontrar lugares donde llevar a cabo actividades urbanas como grafitti, exploración, fotografía, grabación...

Partes interesadas

Los actores de estos proyectos son:

- Profesorado, que irá evaluando la evolución del proyecto.
- Alumnos/técnicos que planterán y desarrollarán la aplicación web.
- IES Borja Moll donde se desarrollará.
- Usuarios finales. El público.

Requisitos de la aplicación

Requisitos funcionales

Usuario no registrado

- RF01: Registrarse
- RF02: Acceder al inicio

Usuario registrado

- RF03: Iniciar sesión
- RF04: Actualizar perfil
- RF05: Cambiar contraseña
- RF06: Publicar lugares
- RF07: Actualizar lugar subido
- RF08: Eliminar lugar subido
- RF09: Ver lista pública de lugares
- RF10: Comentar lugares

Moderador

- RF03: Iniciar sesión
- RF04: Actualizar perfil
- RF05: Cambiar contraseña
- RF06: Publicar lugares
- RF07: Actualizar lugar subido
- RF08: Eliminar lugar subido
- RF09: Ver lista pública de lugares
- RF10: Comentar lugares
- RF11: Aprobar publicación lugar
- RF12: Moderar comentarios

Administrador

- RF02: Acceder al inicio
- RF03: Iniciar sesión
- RF04: Actualizar perfil
- RF05: Cambiar contraseña
- RF06: Publicar lugares
- RF07: Actualizar lugar subido
- RF08: Eliminar lugar subido
- RF09: Ver lista pública de lugares
- RF10: Comentar lugares
- RF11: Aprobar publicación lugar
- RF12: Moderar comentarios
- RF13: Elegir moderador
- RF14: Administrar cuentas
- RF15: Aprobar peticiones

- RF16: Crear secciones

Requisitos no funcionales

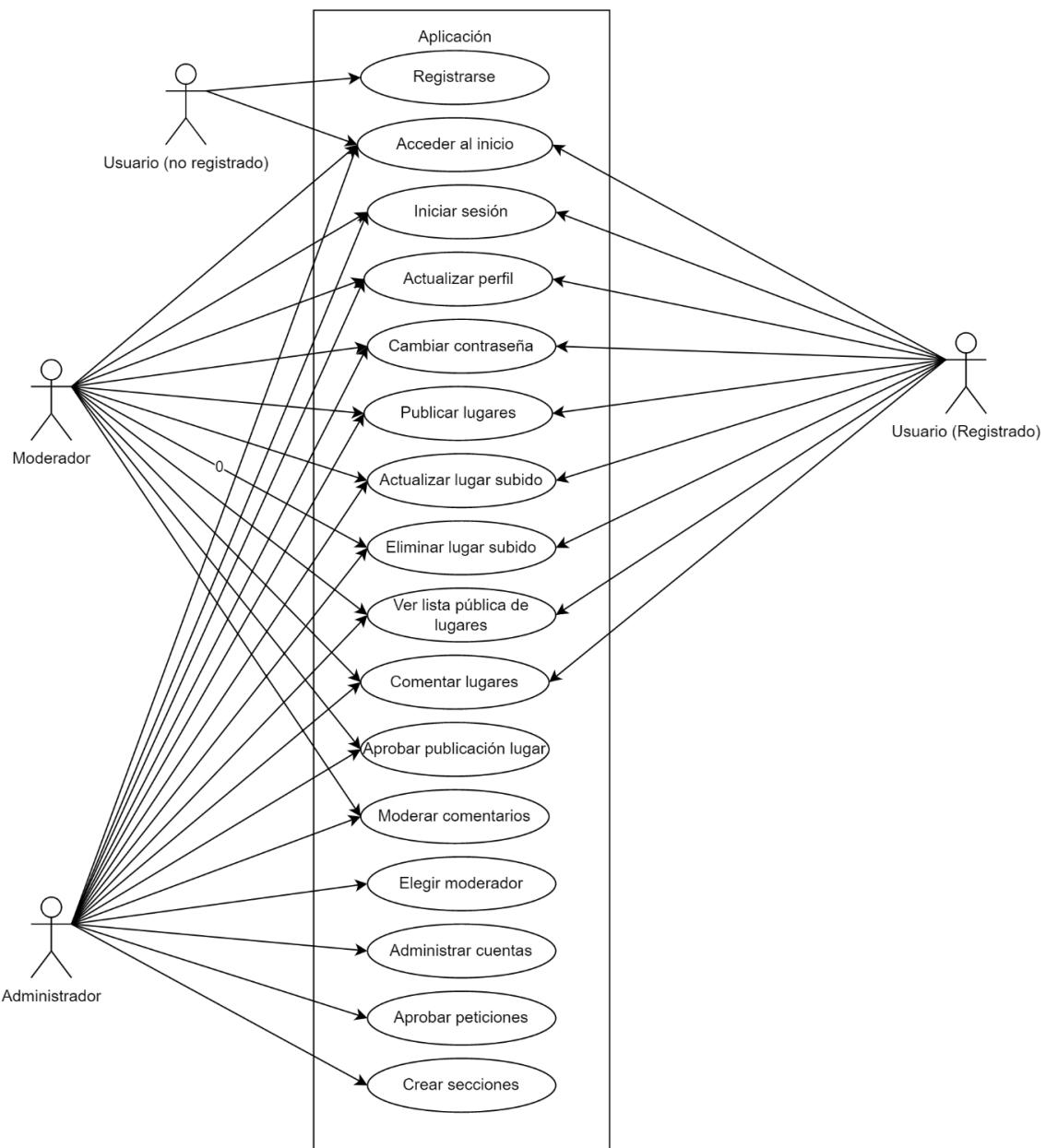
- RnF01: Escalabilidad.
- RnF02: Elasticidad.
- RnF03: Reusabilidad.
- RnF04: Funcionamiento de la aplicación sin internet.
- RnF05: Funcionamiento en app para android, solo funcionamiento web.
- RnF06: Implementación de un sistema de seguridad.
- RnF06: Registro de menores de edad.
-

Requisitos de sistema

RS01: Debe estar disponible las 24h

RS02: Podrá filtrarse la búsqueda de publicaciones

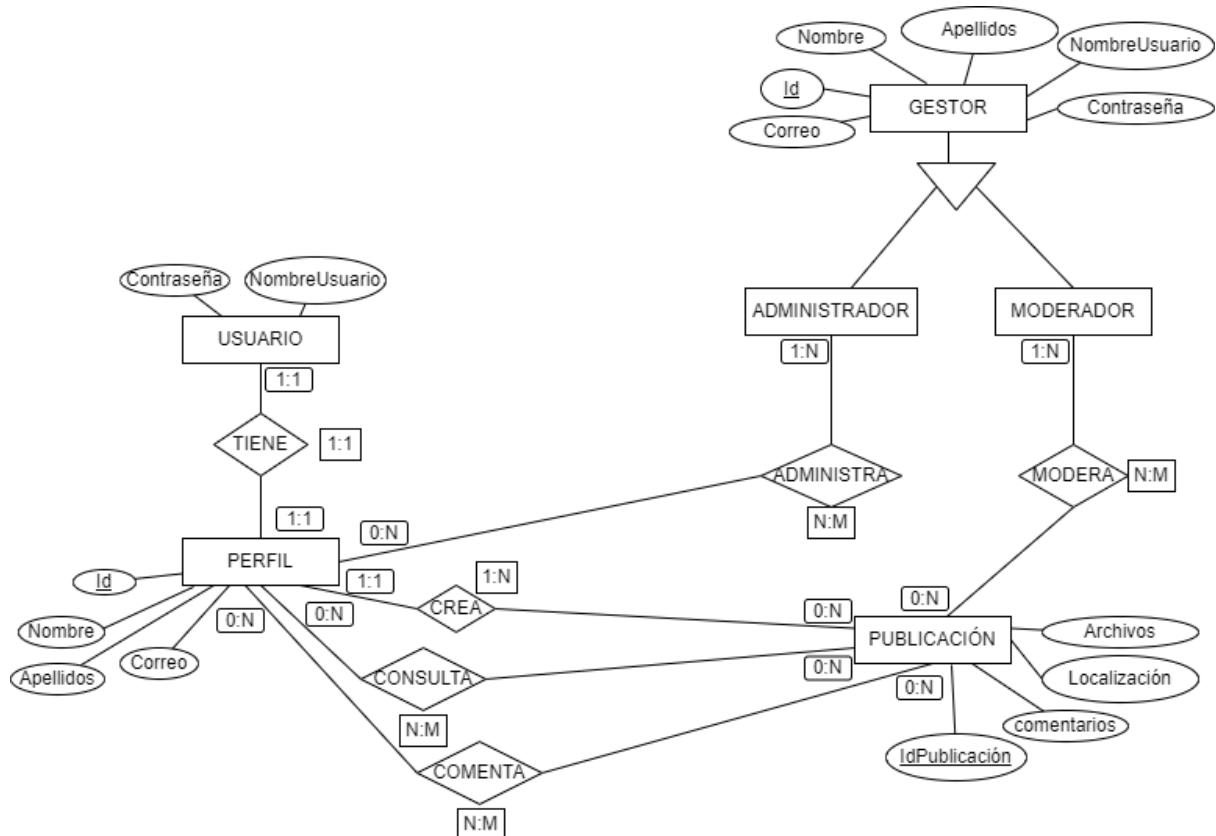
Diagrama de casos de uso



Selección de tecnologías mediante la matriz Requisitos/Tecnologías.

Requisitos/ Tecnologías	MySQL	HTML/CSS	JS	Java
RF01	x	x	x	x
RF02		x	x	
RF03-16	x	x	x	x

Modelo Entidad Relación (MER).



Diseño

Modelo conceptual del proyecto (ampliar el de la introducción con las tecnologías seleccionadas en la fase de Análisis).

Diagrama de clases

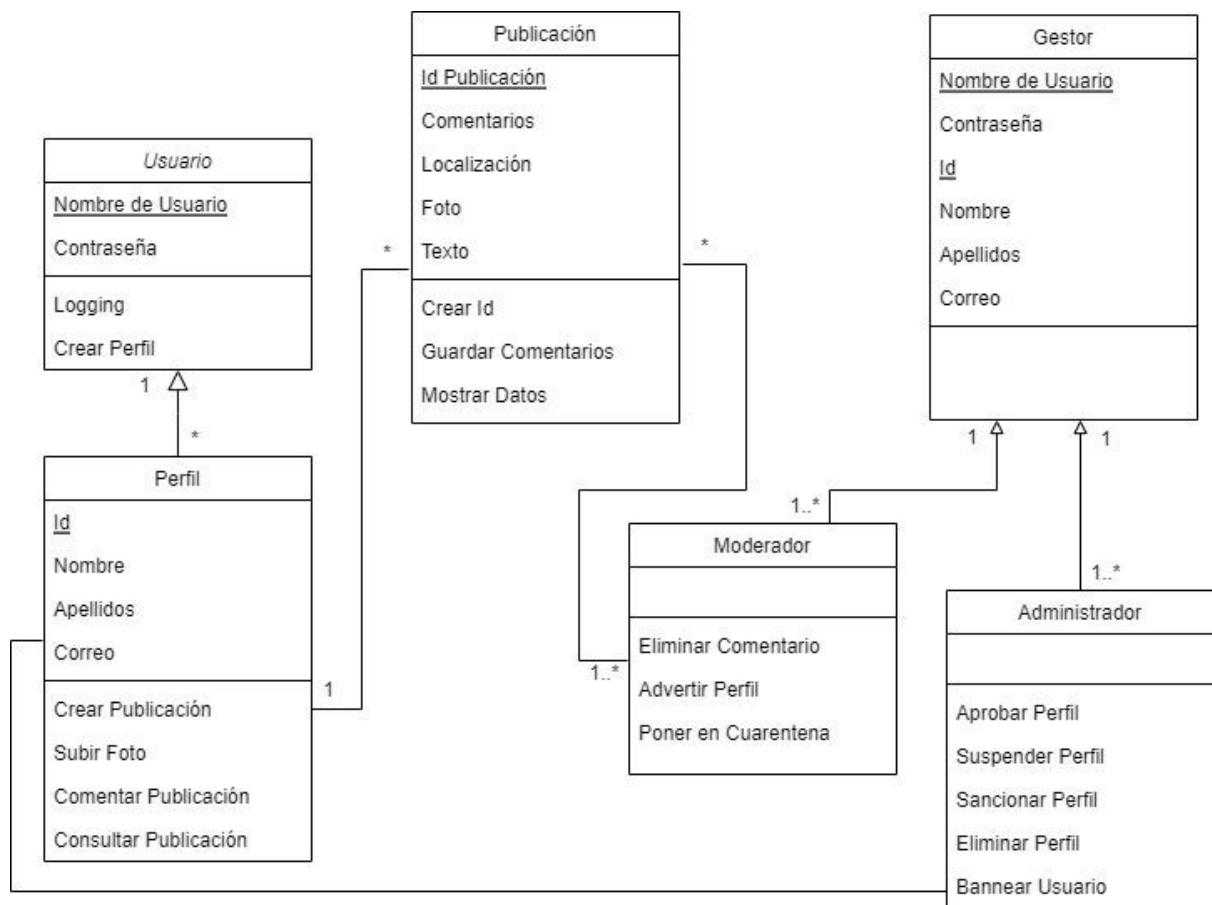
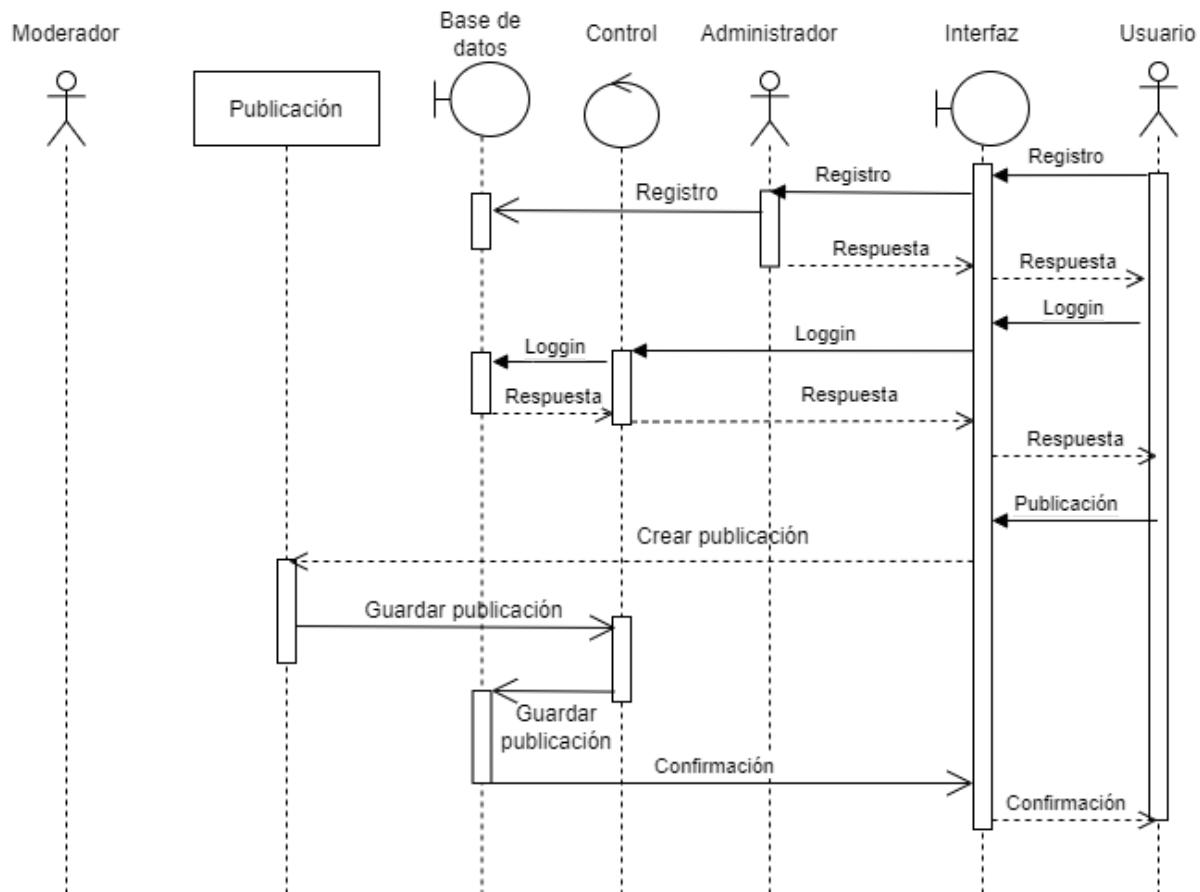


Diagrama de secuencia



Paso a relacional del MER y normalización.

RELACIONAL

```

usuario(nombreUsuario, contraseña, idPerfil);
USUARIO_PK(nombreUsuario);
USUARIO_FK(idPerfil) REFERENCES (perfil.id);
  
```

```

perfil(id,nombre,apellidos,correo);
PERFIL_PK(id);
  
```

```

publicacion(idPublicacion, comentarios, ubicacion,archivos,idPerfil);
PUBLICACION_PK(idPublicacion);
PUBLICACION_FK(idPerfil) REFERENCES (perfil.id);
  
```

```

comenta(idPerfil, idPublicacion);
COMENTA_PK(idPerfil, idPublicacion);
COMENTA_FK(idPerfil) REFERENCES (perfil.id);
COMENTA_FK(idPublicacion) REFERENCES (publicacion.id);
  
```

```

consulta(idPerfil, idPublicacion);
CONSULTA_PK(idPerfil, idPublicacion);
CONSULTA_FK(idPerfil) REFERENCES (perfil.id);
CONSULTA_FK(idPublicacion) REFERENCES (publicacion.id);

gestor(id, nombre, apellidos, nombreUsuario, contraseña, correo);
GESTOR_PK(id);

administrador(idGestor);
ADMINISTRADOR_PK(idGestor);
ADMINISTRADOR_FK(idGestor) REFERENCES (gestor.id);

moderador(idGestor);
MODERADOR_PK(idGestor);
MODERADOR_FK(idGestor) REFERENCES (gestor.id);

administra(idPerfil,idGestor);
ADMINISTRA_PK(idPerfil,idGestor);
ADMINISTRA_FK(idPerfil) REFERENCES (perfil.id);
ADMINISTRA_FK(idGestor) REFERENCES (gestor.id);

modera(idPublicacion,idGestor);
MODERA_PK(idPublicacion,idGestor);
MODERA_FK(idPublicacion) REFERENCES (publicacion.idPublicacion);
MODERA_FK(idGestor) REFERENCES (gestor.id);

```

Normalización

usuario(nombreUsuario, contraseña, idPerfil)

perfil(id, nombre, apellidos, correo)

publicacion(idPublicacion, comentarios, ubicacion, archivos, idPerfil)

comentarios(idPerfil, idPublicacion)

consultas(idPerfil, idPublicacion)

gestor(id, nombre, apellidos, nombreUsuario, contraseña, correo)

administrador(idGestor)

moderador(idGestor)

administra(idPerfil,idGestor)

modera(idPublicacion,idGestor)

Diseño del esquema de la web

Secciones de la aplicación

- Página principal

Sin iniciar sesión

Contenido:

- Nombre y logo de la aplicación
- Texto de bienvenida.
- Información general para inducir al usuario a registrarse.
- Alerta de cookies.

Acceso a (Vía botón, link...):

- Loggin y register.
 - Imagen de perfil.
 - Nombre de usuario.
 - Nombre.
 - Apellidos.
 - Email.
 - Fecha de nacimiento.
 - Política de protección de datos.
- Barra de navegación (Menú principal)
- Sobre nosotros
- Guía de uso
 - Guía de uso de la aplicación
 - Guía de uso de comentarios y publicaciones.
 - Guía de exploración urbana.
- Contacto.
- Política de cookies.

Sesión iniciada

- Nombre de usuario y acceso a su perfil.
- Acceso a publicaciones.
- Acceso a creación de publicaciones.

USUARIOS:

- Usuario no registrado
 - Acceder a la página de inicio de la aplicación.
 - Registrarse
- El usuario registrado podrá:

- Actualizar perfil (cambios aprobados por el administrador)
 - Cambiar contraseña (cambios aprobados por el administrador)
 - *Publicar lugares (pendientes de moderación)*
 - *Eliminar publicaciones (pendiente de moderación)*
 - Ver listado de publicaciones, pudiendo filtrarlas por zona, tipo de construcción, nivel de deterioro, nivel de rareza, nivel de accesibilidad.
 - Comentar publicaciones
- Moderadores
 - Aprobar publicaciones que suben los usuarios.
 - Moderar comentarios
- Administradores:
 - Elegir moderadores.
 - Aprobar publicaciones que suben los usuarios.
 - Moderar comentarios
 - Administrar las cuentas
 - Leer y aprobar las peticiones de eliminación de lugares publicados.
 - Eliminar cuentas
 - Aceptar cambios en los perfiles de los usuarios.
 - Restaurar contraseñas.
 - Crear nuevas secciones.

Listado de tecnologías.

Por parte del frontend:

- Javascript
 -HTML
 -CSS

Backend:

-Java
 -Tomcat

Base de datos:

-Mysql

Otras tecnologías:

- Bootstrap
- Mariadb

Implementación

En este apartado veremos la implementación de una prueba funcional a partir del modelo conceptual de nuestro proyecto.

Instalación de plataforma→implementación en la red→operaciones→visualización de la plataforma.

Planificación y diseño de los casos de pruebas. Tipos de pruebas

El diseño de los casos de prueba serán unitarios (caja negra) donde se irán probando las pequeñas funciones y código del programa.

Los casos de prueba:

-Función registrar un usuario → salida → deberá registrar al usuario correctamente y enviarlo a la base de datos a través del servidor y comprobar que el usuario no esté registrado ya, una vez este todo bien redirigir al login.

-función login → salida → la función debe comprobar que el usuario esta registrado obteniendo los datos de contraseña de la base de datos, en caso contrario te redirige a la función registro.

-Función subir un post → salida → antes de insertar el post se debe comprobar que cumple con los términos de la app.

Pruebas de integración:

Se han puesto a prueba la conexión y las interfaces de los servidores a la base de datos y por supuesto con el frontend.

Pruebas de carga:

La app se ha probado en en un navegador web de otros ordenadores para comprobar si resiste otros entornos.

Pruebas de aceptación:

Alfa. Se lleva a cabo en IES Borja Moll donde estará supervisado por los creadores de la app

Beta. Las realizan los profesores desde su entorno sin ninguno de los creadores para que les guíe.