

# ANTEPROYECTO

APLICACIÓN WEB DE MENSAJES EFÍMEROS CON SISTEMA DE  
REACCIONES Y MODERACIÓN DE CONTENIDO

FÉLIX SAN JOSÉ HERNÁN  
CFGS DESARROLLO DE APLICACIONES WEB

## **ÍNDICE**

<b>INTRODUCCIÓN .....</b>	<b>2</b>
<b>CASOS DE USO.....</b>	<b>3</b>
<b>OBJETIVOS.....</b>	<b>5</b>
<b>FASES DEL PROYECTO .....</b>	<b>6</b>
<b>HERRAMIENTAS DEL PROYECTO.....</b>	<b>7</b>
<b>TIEMPO ESTIMADO PARA SU REALIZACIÓN.....</b>	<b>8</b>

## INTRODUCCIÓN

En este proyecto se quiere desarrollar una aplicación web en la que los usuarios puedan dejar mensajes escritos para que otros usuarios puedan visualizarlos de manera aleatoria e interactúen con ellos.

Al acceder a la web, el usuario podrá realizar un clic para que se le muestre un mensaje seleccionado aleatoriamente de entre los almacenados en la base de datos. Dicho mensaje podrá ser reaccionado mediante un “me gusta” o con emoticonos, una vez se cierre ese mensaje no podrá volver a verlo, ya que los mensajes aparecen de manera aleatoria y con una gran cantidad de entradas en la base de datos la probabilidad de ver un mensaje dos veces será muy reducida.

Esta decisión de diseño se inspira en la idea de representar los mensajes como estrellas fugaces, estableciendo así una temática visual y conceptual basada en el espacio y el universo.

Cualquier usuario registrado podrá escribir nuevos mensajes y gestionarlos desde su perfil, donde verán las reacciones de otros usuarios a su mensaje, eliminarlo o editarlo en cualquier momento.

Asimismo, los mensajes podrán ser reportados por otros usuarios si se consideran ofensivos o inapropiados, una vez un mensaje llegue a un número determinado de reportes se notificará a la cuenta del administrador por correo para que lo revise manualmente y decida si mantenerlo o eliminarlo. En caso de eliminarlo y si el autor acumula varios mensajes eliminados por infracciones, se suspenderá su cuenta.

La aplicación se encontrará alojada en un servidor propio y se podrá acceder a ella fuera de la red local, evitando el uso de hosting en línea, por preferencias personales y por ejemplo garantizar un mayor control sobre los recursos y la estructura de este, evitando posibles errores al desplegar la aplicación debido a las configuraciones específicas de cada hosting.

## **CASOS DE USO**

### **Caso de Uso 1: Ver e interactuar con un mensaje**

Actor: Usuario sin registrar, Usuario registrado

Descripción: Permite al usuario ver un mensaje aleatorio e interactuar con él

#### **Flujo Principal:**

1. El usuario accede a la aplicación.
2. Hace clic en cualquier parte de la ventana de la aplicación.
3. Se le muestra un mensaje aleatorio de la base de datos.
4. El usuario puede responder al mensaje con una lista de emoticonos y reportar el mensaje como inapropiado.
5. Al cerrar el mensaje, el usuario no podrá volver atrás para volver a verlo.

### **Caso de Uso 2: Registrarse o Iniciar Sesión**

Actor: Usuario sin registrar

Descripción: Permite a los usuarios registrarse en la aplicación o iniciar sesión, realizando al finalizar una doble autenticación.

#### **Flujo Principal:**

1. El usuario elige entre registrarse o iniciar sesión.
2. Si elige registrarse, ingresa los datos requeridos y el sistema crea una cuenta.
3. Si elige iniciar sesión con su cuenta de la aplicación, ingresa sus credenciales.
4. Se le pide al usuario que ingrese el código para la doble autenticación.
5. Una vez autenticado, el usuario regresa a la aplicación con acceso a su cuenta.

### **Caso de Uso 3: Un usuario gestiona sus propios mensajes**

Actor: Usuario registrado

Descripción: Permite a un usuario ver sus mensajes, las reacciones de otros usuarios a ellos, y poder eliminarlos o editarlos.

#### Flujo Principal:

1. El usuario accede al panel de control de su perfil.
2. Se le muestra una lista con todos sus mensajes, las reacciones de los usuarios y botones para proceder a eliminar o editar.
3. En caso de editar, podrá editar todo el mensaje desde esa propia página, confirmar los cambios o cancelarlos.
4. En caso de eliminar, se le mostrará una notificación para que confirme que quiere eliminarlo.

### **Caso de Uso 4: El administrador gestiona los reportes**

Actor: Usuario administrador

Descripción: El administrador gestiona manualmente si un mensaje reportado debe ser eliminado o se puede mantener.

#### Flujo Principal:

1. El administrador recibe una notificación por correo con un link al mensaje reportado o accede a todos los mensajes desde una lista en el panel de la web.
2. Se muestra el mensaje, el número de reportes y las acciones a realizar: eliminar el mensaje o dejarlo.
3. En caso de eliminar el mensaje, y si el autor acumula varios mensajes eliminados por infracciones, se bloquea su cuenta.

## **OBJETIVOS**

- **Implementar un sistema de visualización de mensajes aleatorios**
  - ✓ Almacenar correctamente los nuevos mensajes y mostrarlos a los usuarios de manera satisfactoria.
- **Permitir las interacciones con los mensajes**
  - ✓ Registrar las interacciones de los usuarios con un mensaje para poder mostrarlos al resto de usuarios y al autor, además de la capacidad de reportar un mensaje.
- **Sistema de registro e inicio de sesión con doble autenticación**
  - ✓ Garantizar la seguridad de las cuentas de los usuarios con doble autenticación y cifrado de sus datos.
- **Panel de control para los usuarios**
  - ✓ Lugar en el que los usuarios registrados puedan gestionar todos sus mensajes escritos y ver las reacciones de otros usuarios.
- **Sistema de reportes y gestión de un administrador**
  - ✓ Notificar por correo a un administrador cuando se supere un número de reportes en un mensaje, y permitir la gestión del mismo desde la web.
- **Diseño responsive, visualmente atractivo y cómodo de manejar**
- **Desplegar la aplicación en servidor**

## **FASES DEL PROYECTO**

### **1. Planificación**

- Definir que realizará la aplicación y cómo
- Decidir que acciones podrá realizar el usuario

### **2. Diseño y funcionamiento**

- Bocetos de cómo se verá la web
- Que ocurrirá al hacer clic en las distintas partes de la web

### **3. Desarrollo de la aplicación**

- Llenar la base de datos con mensajes previamente generados para poder realizar mejores pruebas
- Mostrar los mensajes aleatorios al usuario
- Permitir añadir reacciones a los mensajes
- Inicio de sesión y registro con doble autenticación
- Preparar zonas para gestionar los mensajes propios y los reportados con los administradores

### **4. Pruebas, depuración y optimización**

### **5. Despliegue y alojamiento en equipo local**

A lo largo de todo el proyecto y en todas sus fases se irá realizando la documentación de este, indicando todos los pasos realizados detalladamente.

## **HERRAMIENTAS DEL PROYECTO**

- **Lenguajes y tecnologías:**
  - ✓ HTML y CSS: creación y diseño visual de la interfaz de usuario.
  - ✓ JavaScript: manejo de la interacción dinámica con la interfaz, integración de librerías para un mejor desarrollo y llamadas asíncronas al backend.
  - ✓ PHP: gestionar la lógica del proyecto en el servidor y la gestión de peticiones.
  - ✓ SQL: almacenamiento y gestión de datos en la base de datos.
  
- **Entorno de desarrollo y herramientas auxiliares:**
  - ✓ Visual Studio Code: editor de código principal para desarrollar tanto el frontend como el backend, gracias al soporte ofrecido por extensiones de los distintos tipos de lenguajes de programación.
  - ✓ Git y GitHub: para el control de versiones y respaldo del código.
  - ✓ MySQL Workbench: gestión visual de la base de datos durante el desarrollo.
  - ✓ PHPMailer: para el envío de códigos de doble autenticación y notificar al administrador de reportes.
  
- **Entorno de despliegue:**
  - ✓ Debian: equipo personal con el sistema operativo Debian donde se alojará el proyecto, configurado como servidor web y de bases de datos.
  - ✓ Apache: actuará como servidor web para servir el frontend de la aplicación.



## **TIEMPO ESTIMADO PARA SU REALIZACIÓN**

El proyecto se realizará a lo largo de 2-3 meses, durante los cuales se irán trabajando todas las fases del proyecto al mismo tiempo que se realiza la documentación, tratando de dejar el máximo tiempo posible para realizar test de rendimiento de la aplicación.