

# MOODIFY

“Aplicación de Registro y Conexión Basada en Estados de Ánimo”

Sistemas Web I, Grupo 101



Yolanda Llop Pellisa

Karen Cebreros López

Myriam Lizeth Figueroa Morales

Marta Canino Romero

Xavier Alexander Mora Peraza



CEU

| *Universidad  
San Pablo*

# Índice de Contenidos

Histórico de versiones .....	3
Descripción .....	4
Motivación.....	4
Objetivos.....	4
Requisitos funcionales .....	7
Justificación de los requisitos .....	9
Documentos de diseño .....	10
Sitemap .....	10
Wireframe.....	11
Arquitectura.....	15
<b>Frontend (Interfaz de Usuario)</b> .....	15
<b>Backend (Servidor)</b> .....	15
<b>Base de Datos (MongoDB)</b> .....	16
<b>AWS Amazon Rekognition</b> .....	16
<b>OpenAI API</b> .....	16
<b>Despliegue de la aplicación en Azure</b> .....	17
<b>APK para Aplicación de Móvil</b> .....	18
Manual de usuario .....	22

## Histórico de versiones

<b>Versión</b>	<b>Fecha</b>	<b>Descripción</b>
<b>1.0.0</b>	23/09/2024	Versión inicial de la aplicación a modo de maqueta, con todas las interfaces solo del lado del cliente
<b>2.0.0</b>	8/10/2024	Segunda versión de la aplicación con el lado del servidor implementado, back-end funcional, conexión y sincronización con la base de datos en MongoDB
<b>3.0.0</b>	9/10/2024	Despliegue de la aplicación en Azure

# Moodify

## Descripción

Moodify es una aplicación web que permite a los usuarios registrar sus emociones diarias, conectar con amigos, compartir su estado de ánimo acompañado de listas de canciones, y obtener recomendaciones musicales basadas en análisis emocional y reconocimiento facial. La plataforma ofrece funcionalidades en tiempo real para que los usuarios puedan interactuar instantáneamente y visualizar los estados de ánimo de sus amigos.

En Moodify, la **gestión de usuarios** estará centralizada en una **base de datos MongoDB**, que nos permitirá manejar de manera eficiente datos no relacionales y estructuras dinámicas propias de la interacción social. Cada usuario tendrá un documento individual que almacenará información como su perfil, historial de emociones, listas de canciones, conexiones con amigos, y preferencias personalizadas.

## Motivación

La motivación detrás de Moodify surge de la creciente conexión entre nuestras emociones y la música como una forma de expresión y alivio emocional. En un mundo cada vez más digital, donde las interacciones sociales a veces pierden autenticidad, Moodify busca crear un espacio donde las personas puedan compartir sus estados de ánimo de manera genuina, conectar con amigos a través de sus emociones y descubrir música que resuene con sus sentimientos. Al integrar tecnologías de análisis emocional y recomendaciones musicales personalizadas, la aplicación pretende mejorar el bienestar emocional y fortalecer las relaciones entre usuarios, fomentando una comunidad segura, empática y dinámica.

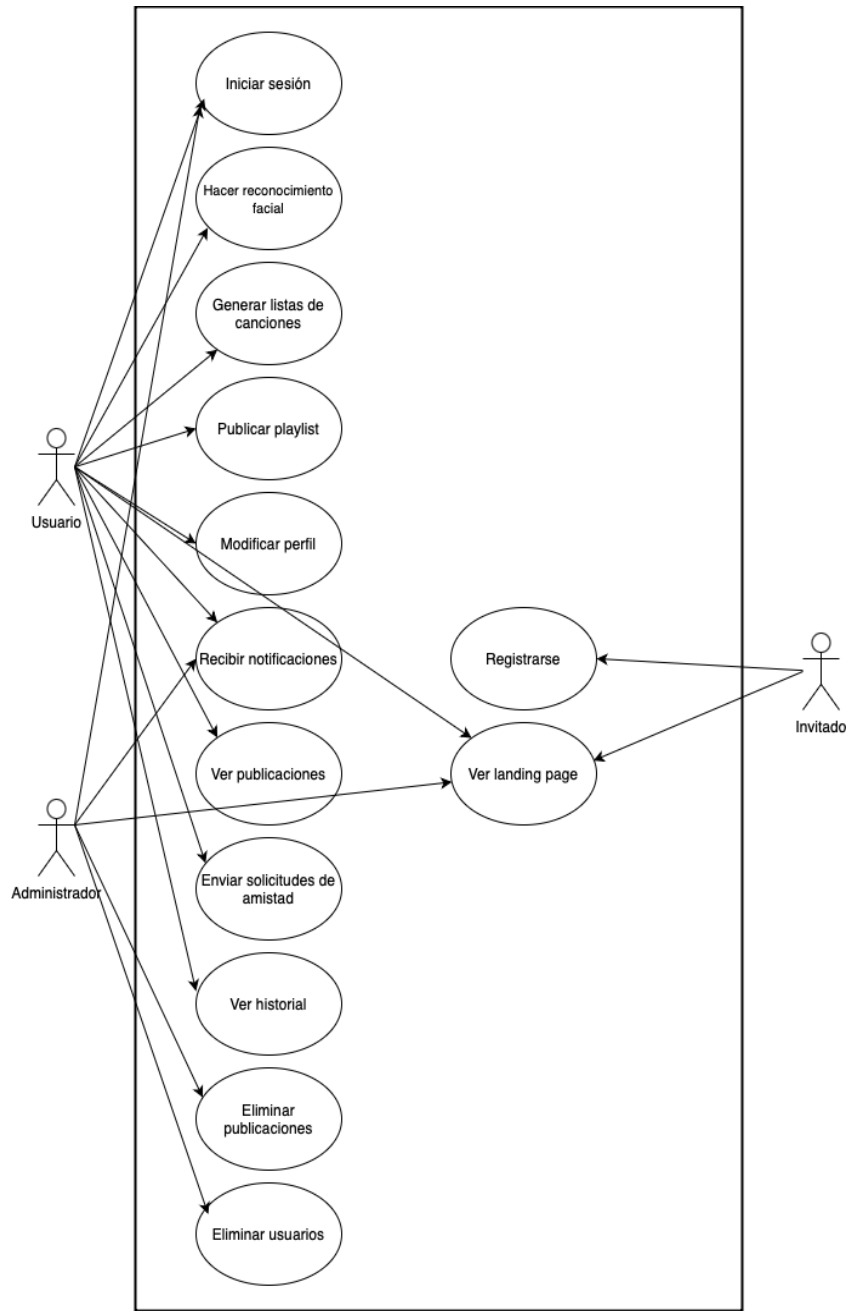
## Objetivos

1. Facilitar el registro diario de emociones: Permitir a los usuarios guardar sus estados de ánimo y canciones asociadas de manera rápida y sencilla.
2. Generar conexiones emocionales: Crear una comunidad donde los usuarios puedan compartir y conocer los estados de ánimo de sus amigos.
3. Ofrecer música personalizada: Generar listas de reproducción basadas en las emociones de los usuarios mediante análisis de emociones y reconocimiento facial.
4. Proporcionar una experiencia de usuario en tiempo real: Permitir actualizaciones y notificaciones instantáneas sobre los estados de ánimo y actividades de los amigos.
5. Asegurar un entorno moderado: Proporcionar al administrador herramientas para gestionar y moderar las publicaciones y comentarios inadecuados.

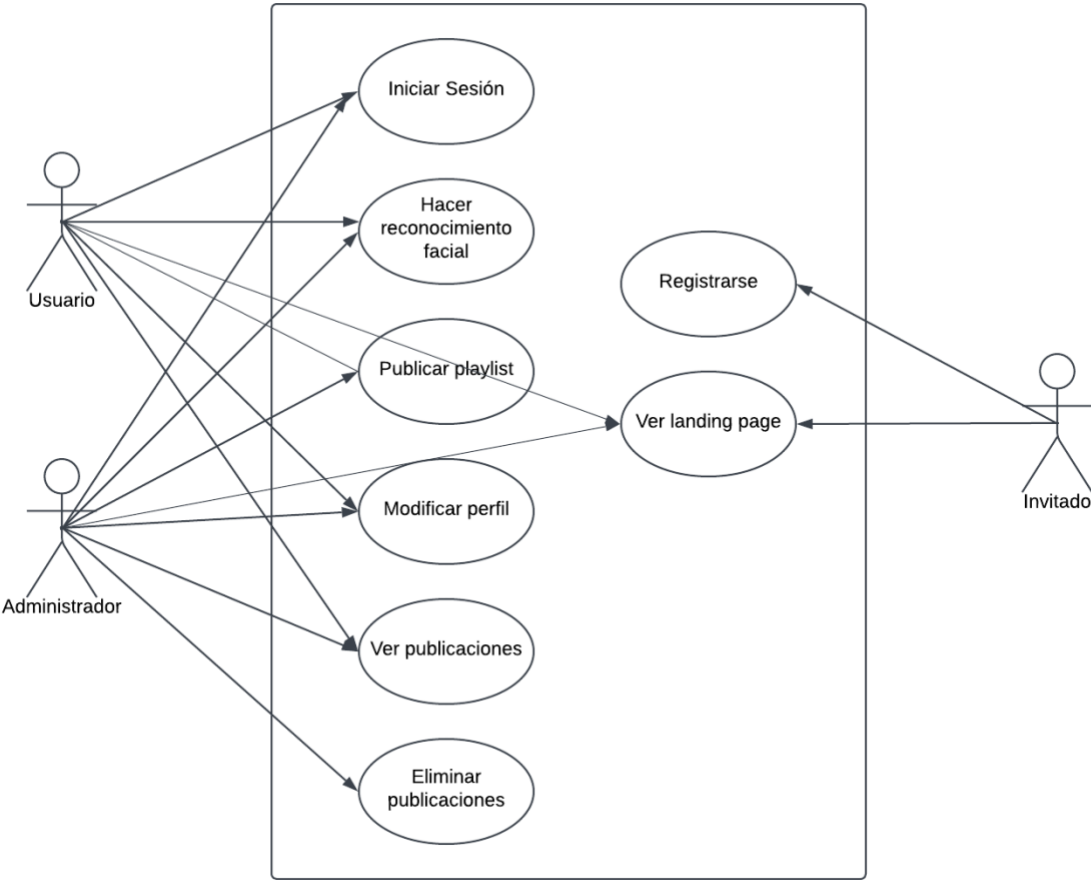
6. Mejorar la interacción social a través de recomendaciones: Sugerir amigos según patrones emocionales y gustos musicales compartidos.

Diagrama de casos de uso

Versión inicial



Versión final



## Requisitos funcionales

En un inicio, se consideraron y definieron los siguientes requisitos funcionales:

**RF1:** Iniciar sesión como usuario registrado.

Los usuarios registrados deben poder iniciar sesión introduciendo sus credenciales válidas.

**RF2:** Registrar un nuevo usuario.

La aplicación debe permitir la creación de cuentas utilizando un correo electrónico válido.

**RF3:** Generar automáticamente una lista de canciones según la emoción detectada.

El sistema debe generar una lista de canciones automáticamente basada en las emociones detectadas por reconocimiento facial.

**RF4:** Registrar una emoción y asociarla a una lista de canciones.

El usuario registrado debe poder almacenar su emoción detectada junto con una lista de canciones en la base de datos.

**RF5:** Consultar el historial de emociones y listas de canciones.

El usuario registrado debe poder acceder a un registro histórico de sus emociones y canciones generadas.

**RF6:** Publicar una emoción diaria y asociarla a una lista de canciones.

El usuario registrado debe poder realizar publicaciones que incluyan su emoción diaria y la lista de canciones generada.

**RF7:** Visualizar publicaciones de amigos en un feed.

El usuario registrado debe poder ver publicaciones de amigos en tiempo real en un feed.

**RF8:** Gestionar conexiones de amistad.

El usuario registrado debe poder enviar, aceptar o rechazar solicitudes de amistad.

**RF9:** Interactuar con publicaciones de amigos.

El usuario registrado debe poder comentar y reaccionar a las publicaciones de amigos.

**RF10:** Crear playlists personalizadas basadas en emociones seleccionadas.

El usuario registrado debe poder seleccionar emociones específicas y generar playlists personalizadas.

**RF11:** Implementar actualizaciones en tiempo real en el feed y las notificaciones.

El sistema debe utilizar tecnología en tiempo real para mostrar actualizaciones de publicaciones y notificaciones.

**RF12:** Moderar publicaciones como administrador.

El administrador debe poder editar o eliminar publicaciones ofensivas.

**RF13: Gestionar cuentas de usuarios como administrador.**

El administrador debe poder suspender, activar o eliminar cuentas de usuarios.

**RF14: Visualizar una landing page como invitado.**

Los invitados deben ser redirigidos a una página de inicio que describa las funcionalidades principales de la aplicación.

**RF15: Acceder y modificar el perfil del usuario registrado.**

El usuario registrado debe poder visualizar y actualizar su información personal.

**RF16: Ganar puntos por registrar emociones.**

El usuario registrado debe recibir puntos por registrar emociones y desbloquear logros al cumplir acciones específicas.

**RF17: Recibir notificaciones en tiempo real.**

El usuario registrado debe recibir notificaciones cuando un amigo publique una emoción o actualice una lista de canciones.

**RF18: Recibir recordatorios para registrar emociones.**

El usuario registrado debe recibir recordatorios diarios para registrar emociones y resúmenes semanales de su actividad emocional.



## Justificación de los requisitos

### ○ Cumplidos

De los 18 requisitos de nuestra aplicación web, cumplimos con 10, dentro de los cuales se encuentran los estrictamente necesarios según las especificaciones iniciales, así como algunos adicionales relacionados con funcionalidades extras que decidimos incorporar para mejorar la experiencia del usuario.

Dichos requisitos que sí se cumplieron, son:

- RF1: Iniciar sesión como usuario registrado
- RF2: Registrar un nuevo usuario
- RF3: Generar automáticamente una lista de canciones según la emoción detectada
- RF4: Registrar una emoción y asociarla a una lista de canciones
- RF6: Publicar una emoción diaria y asociarla a una lista de canciones
- RF7: Visualizar publicaciones de amigos en un feed
- RF11: Implementar actualizaciones en tiempo real en el feed y las notificaciones
- RF12: Moderar publicaciones como administrador
- RF14: Visualizar una landing page como invitado
- RF15: Acceder y modificar el perfil del usuario registrado

### ○ No cumplidos

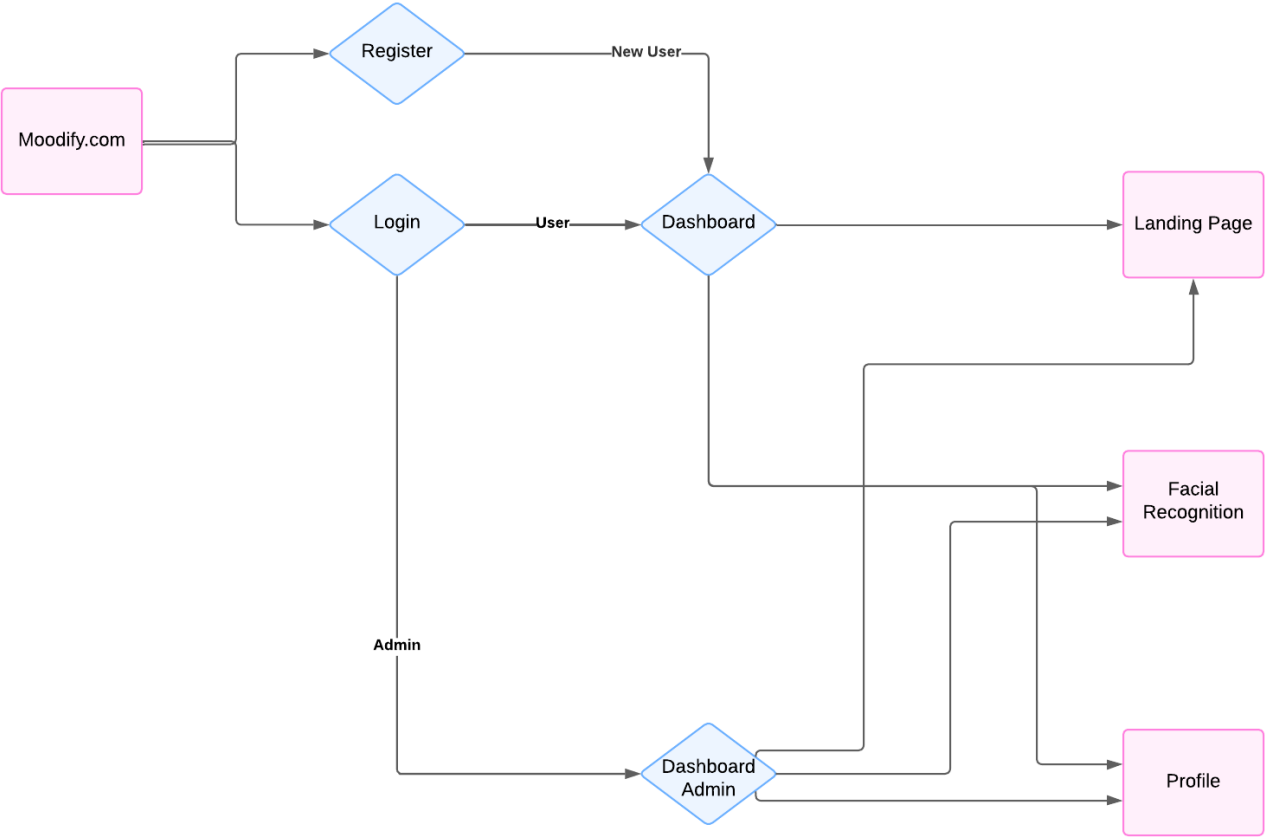
Abajo podemos encontrar una tabla que contiene los requisitos no cumplidos de nuestra aplicación web, con su respectiva justificación.

RF5: Consultar el historial de emociones y listas de canciones	Debido a la falta de tiempo, este requisito no pudo ser implementado.
RF8: Gestionar conexiones de amistad	Este requisito faltó de implementar debido a que resultó ser más compleja de lo previsto inicialmente.
RF9: Interactuar con publicaciones de amigos	Debido que no pudimos cumplir con el requisito RF8, este requisito tampoco pudo ser implementado.
RF10: Crear playlists personalizadas basadas en emociones seleccionadas	Este requisito demandaba más tiempo del disponible, debido al uso extra de APIs que se necesitaban para su implementación.
RF13: Gestionar cuentas de usuarios como administrador	Se priorizó el requisito RF12 y nuevamente por falta de tiempo, no se pudo terminar de implementar éste.
RF16: Ganar puntos por registrar emociones	Debido a su baja prioridad en nuestro plan de desarrollo, se decidió dejar este requisito para el final en caso de que sobrara tiempo disponible para implementarlo.
RF17: Recibir notificaciones en tiempo real	Implementar el servicio de notificaciones en tiempo real resultó ser más complejo de lo previsto en un inicio y excedió el alcance inicial del proyecto.
RF18: Recibir recordatorios para registrar emociones	Al igual que con otros requisitos, debido a la falta de tiempo, éste tampoco pudo ser implementado.

## Documentos de diseño

### Sitemap

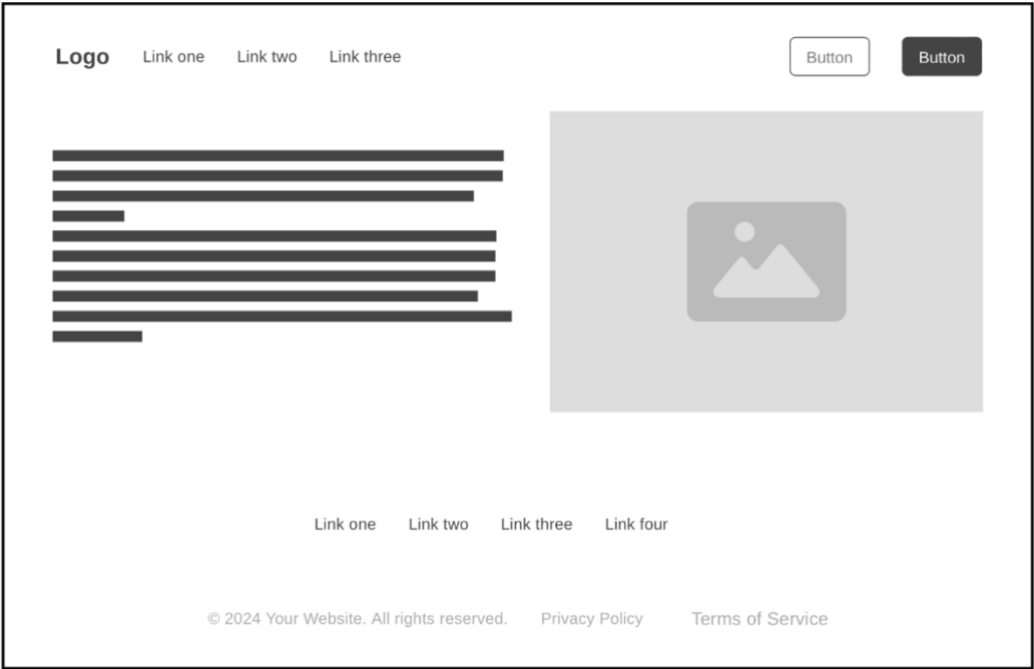
Representación estructurada de las páginas y secciones de la aplicación web. Define la jerarquía y las relaciones entre las distintas partes del sistema:



# Wireframe

Esquema visual de baja fidelidad que muestra la disposición y estructura básica de la aplicación web, diseñado inicialmente como base previa a implementar y desarrollar la aplicación.

## Landing Page:



## Login:



## Register:

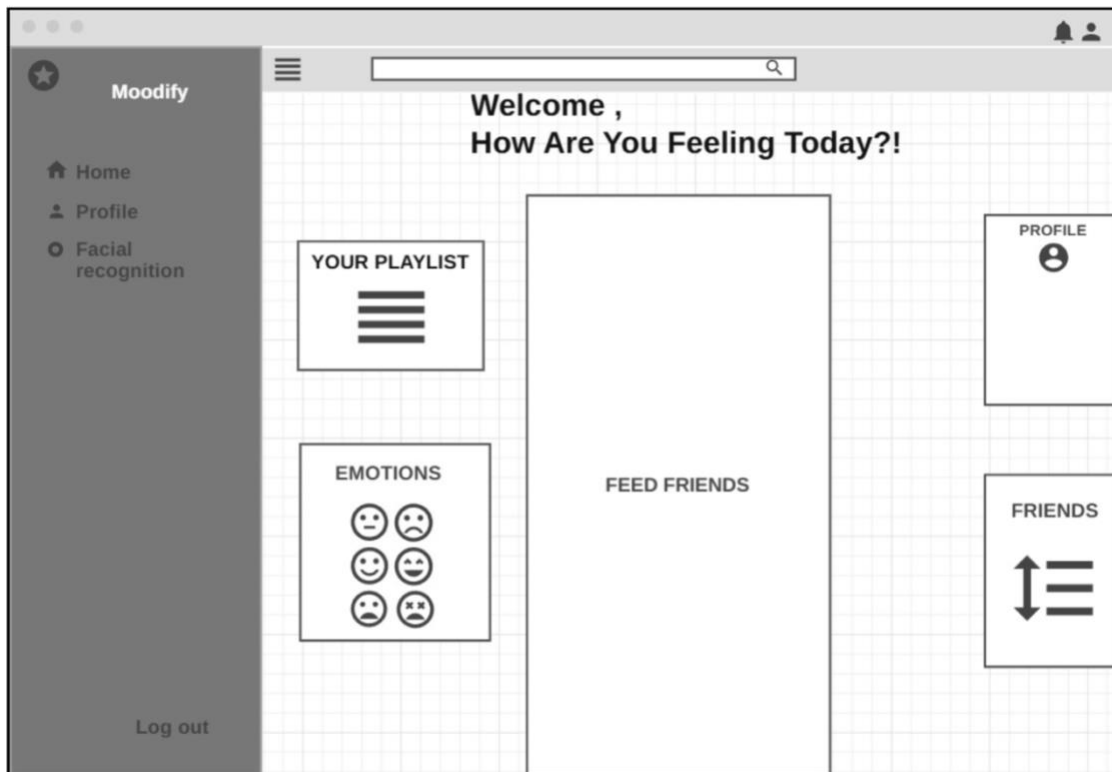
Registrarse

Iniciar Sesión

Registro

Al registrarse, acepta nuestros [Términos de servicio](#) y [Política de privacidad](#).

## Home Page:



Profile:

★

⌵

🔔

👤

← Perfil

Nombre Apellido

Información Personal

Cancelar

Guardar cambios

Facial Recognition:

★

⌵

🔔

👤

Camera

Detected emotion:

Happy

Recommended songs:

• Song 1 - Album - Artist

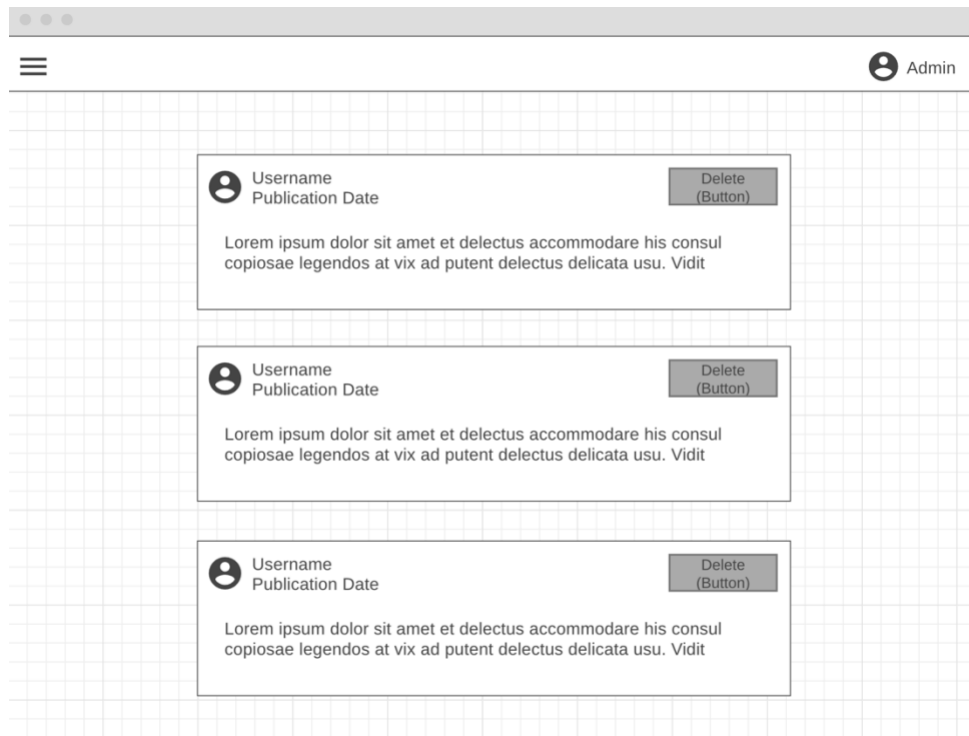
• Song 2 - Album - Artist

• Song 3 - Album - Artist

• Song 4 - Album - Artist

• Song 5 - Album - Artist

Button  
(Generate playlist)

**Dashboard Admin:****Publicaciones:**

## Arquitectura

Estructura técnica y lógica de la aplicación, que incluye el diseño del backend, frontend, bases de datos y flujos de comunicación. Describe cómo interactúan los diferentes componentes del sistema para garantizar un funcionamiento eficiente y escalable.

Moodify es una aplicación web diseñada con una arquitectura de cliente-servidor que combina tecnologías modernas para ofrecer funcionalidades avanzadas de reconocimiento facial y otros servicios adicionales. A continuación, se describen los componentes principales y cómo interactúan entre sí:

### Frontend (Interfaz de Usuario)

El frontend es la parte de la aplicación con la que los usuarios interactúan directamente. Su objetivo es captar las solicitudes del usuario y mostrar las respuestas de manera clara y visual. Para esto, utilizamos:

- **HTML y CSS:** Para estructurar y estilizar la interfaz.
- **Frameworks como Bootstrap:** Para crear un diseño responsivo y atractivo de forma eficiente.
- **SweetAlert:** Para mostrar mensajes y alertas visuales de manera moderna y profesional.
- **JavaScript:** Para manejar la interacción con el usuario en el navegador, enviar solicitudes al backend y procesar las respuestas de manera dinámica.

El frontend se encarga de **recopilar datos** del usuario, como imágenes o entradas, y de **mostrar las respuestas** proporcionadas por el backend, asegurando una experiencia intuitiva.

### Backend (Servidor)

El backend está construido con **Node.js** y utiliza el framework **Express.js** para manejar las solicitudes y coordinar las diferentes funcionalidades. Este componente actúa como el núcleo de la aplicación, cumpliendo las siguientes funciones:

1. **Recepción de solicitudes:** El servidor recibe las solicitudes HTTP enviadas por el frontend.
2. **Procesamiento de datos:** Valida las entradas y coordina las funcionalidades necesarias.
3. **Comunicación con servicios externos:**
  - a. **AWS:** Para realizar análisis avanzados como reconocimiento facial.
  - b. **OpenAI:** Para generar la lista de canciones.
4. **Generación de respuestas:** Procesa los resultados obtenidos y los envía de vuelta al frontend.

Este diseño permite mantener una estructura organizada y escalable en el backend, facilitando la integración de nuevas funcionalidades en el futuro.

### Base de Datos (MongoDB)

Para la persistencia de datos, utilizamos **MongoDB**, una base de datos NoSQL que nos permite almacenar información estructurada y no estructurada. En nuestra aplicación, MongoDB es utilizada para:

- I. Guardar datos de usuarios.
- II. Almacenar resultados procesados, como registros relacionados con el análisis facial.
- III. Registrar configuraciones o logs de interacciones.

Su flexibilidad y escalabilidad hacen que sea ideal para manejar los datos de nuestra aplicación.

### AWS Amazon Rekognition

Amazon Rekognition es un servicio de AWS que permite analizar imágenes mediante aprendizaje automático. En el caso de nuestra aplicación:

- I. **Subida de la imagen:** La aplicación envía la imagen capturada al servicio de Amazon Rekognition a través de la API.
- II. **Detección de emociones:** Se utiliza el método DetectFaces de Rekognition, que analiza los rostros en la imagen y devuelve datos sobre características faciales, incluyendo las emociones predominantes (por ejemplo, alegría, tristeza, enojo, etc.).
- III. **Respuesta de la API:** El servicio responde con un objeto JSON que incluye las emociones detectadas, cada una con un nivel de confianza. Este JSON se envía al servidor de la aplicación para su procesamiento.

### OpenAI API

OpenAI ofrece modelos de lenguaje como GPT que permiten generar contenido dinámico en función de las entradas proporcionadas. En el caso de nuestra aplicación:

- I. **Preparación de la solicitud:** Después de obtener la emoción detectada, se envía una solicitud a la API de OpenAI especificando el contexto. Por ejemplo, se puede incluir un prompt como: "Genera una lista de canciones relacionadas con emociones de alegría".
- II. **Uso de la API:** Se realiza una llamada a la API de OpenAI utilizando el endpoint de chat/completions (para GPT-4 o GPT-3.5). Se proporciona el prompt en el cuerpo de la solicitud, junto con los parámetros necesarios (modelo a usar, temperatura, y otros ajustes).
- III. **Respuesta de la API:** OpenAI responde con una lista de canciones generada dinámicamente, que se devuelve al servidor de la aplicación para su presentación al usuario.

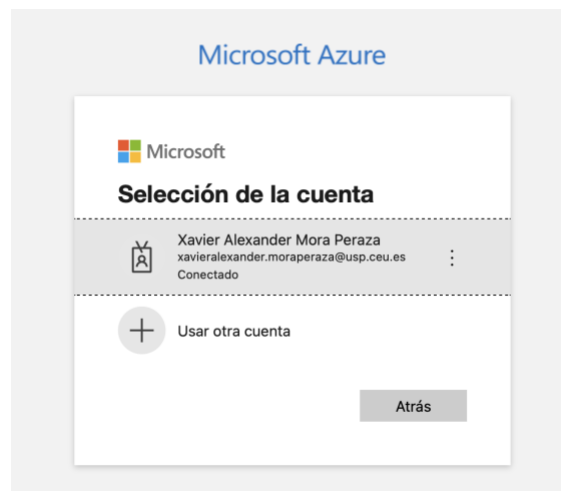


## Despliegue de la aplicación en Azure

Azure ofrece una capa gratuita para realizar despliegue de aplicaciones web. En el caso de nuestra aplicación:

- I. Hemos descargado e instalado la línea de comandos de Azure (Azure CLI).
- II. Iniciamos sesión en la plataforma de Azure a través de la CLI.

```
xaviermoraperaza@LAPTOP-16Q0DCUF ~ % az login
A web browser has been opened at https://login.microsoftonline.com/organizations/oauth2/v2.0/authorize. Please continue
the login in the web browser. If no web browser is available or if the web browser fails to open, use device code flow
with 'az login --use-device-code'.
[
  {
    "cloudName": "AzureCloud",
```



- III. Entramos al directorio donde se encuentra la aplicación y lanzamos el comando de despliegue.

Parámetros del comando:

- Webapp up: Para indicarle que se va a desplegar una aplicación.
- --sku F1: para indicarle que sea de la capa gratuita.
- --name: El nombre de la aplicación.

```
xaviermoraperaza@MacBook-Pro-de-Xavier src % az webapp up --sku F1 --name moodify-demo
```

- IV. El sistema verifica si el nombre de la aplicación ya existe.

**The webapp 'moodify-demo' doesn't exist**

Si no existe continúa.

- V. Si todo va bien comienza a crear los recursos.

```

Creating Resource group 'xavieralexander.moraperaza_rg_5260' ...
Resource group creation complete
Creating AppServicePlan 'xavieralexander.moraperaza_asp_5985' or Updating if already exists
Creating webapp 'moodify-demo' ...
Configuring default logging for the app, if not already enabled
Creating zip with contents of dir /Users/xaviermoraperaza/Desktop/swi/moodify/src ...
Getting scm site credentials for zip deployment
Starting zip deployment. This operation can take a while to complete ...
Deployment endpoint responded with status code 202
You can launch the app at http://moodify-demo.azurewebsites.net
Setting 'az webapp up' default arguments for current directory. Manage defaults with 'az configure --scope local'
--resource-group/-g default: xavieralexander.moraperaza_rg_5260
--sku default: F1
--plan/-p default: xavieralexander.moraperaza_asp_5985
--location/-l default: canadacentral
--name/-n default: moodify-demo

```

VI. Finalmente, si todo ha ido correcto, levanta la aplicación y devuelve la información.

```

{
  "URL": "http://moodify-demo.azurewebsites.net",
  "appserviceplan": "xavieralexander.moraperaza_asp_5985",
  "location": "canadacentral",
  "name": "moodify-demo",
  "os": "Linux",
  "resourcegroup": "xavieralexander.moraperaza_rg_5260",
  "runtime_version": "node|16-LTS",
  "runtime_version_detected": "0.0",
  "sku": "FREE",
  "src_path": "//Users//xaviermoraperaza//Desktop//swi//moodify//src"
}

```

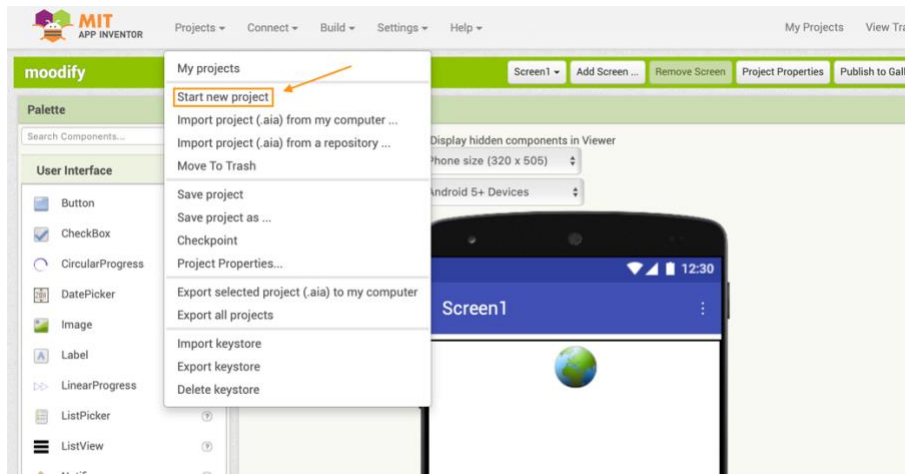
Enlace de la aplicación: <https://moodify-demo.azurewebsites.net>

## **APK para Aplicación de Móvil**

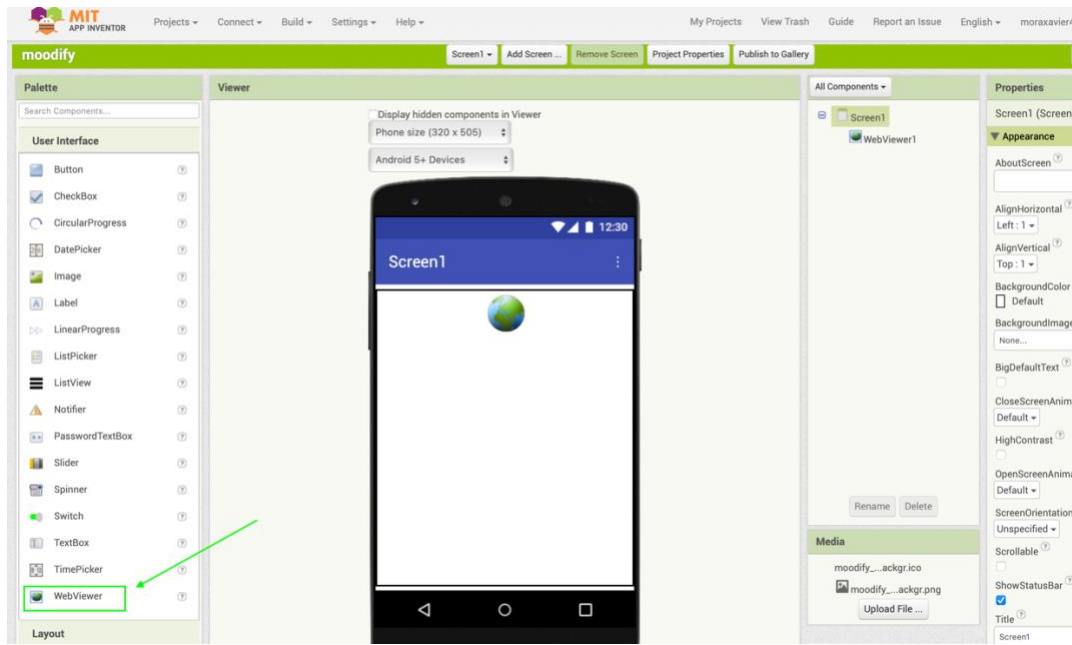
A parte de la aplicación web tambien hemos generado un apk para móvil (Android) a partir de la aplicación web.

Para ello hemos utilizado la plataforma de [MIT-App-Inventor](#), a continuación, se muestra el proceso de conversión.

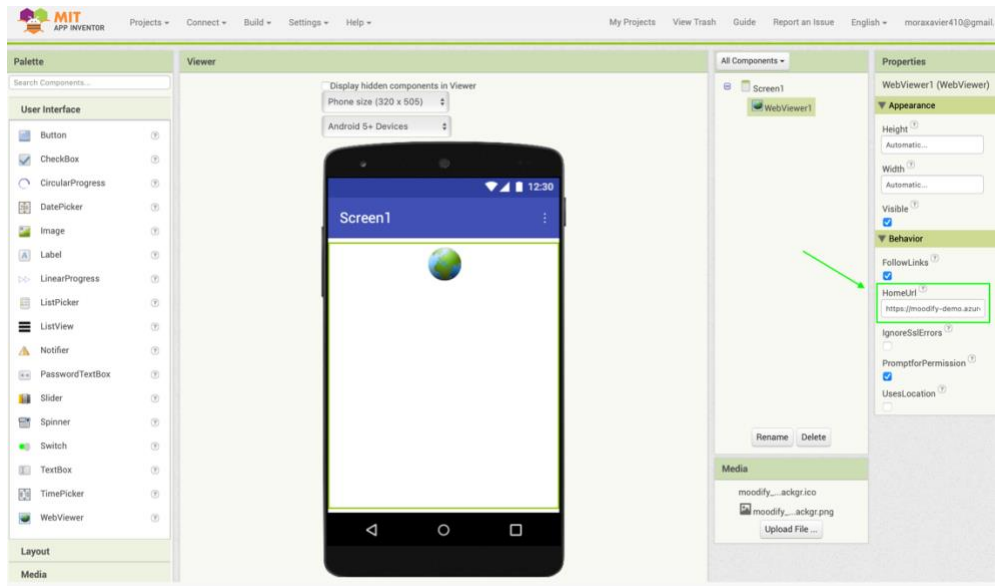
I. Una vez que se ha iniciado sesión en la plataforma, creamos un nuevo proyecto.



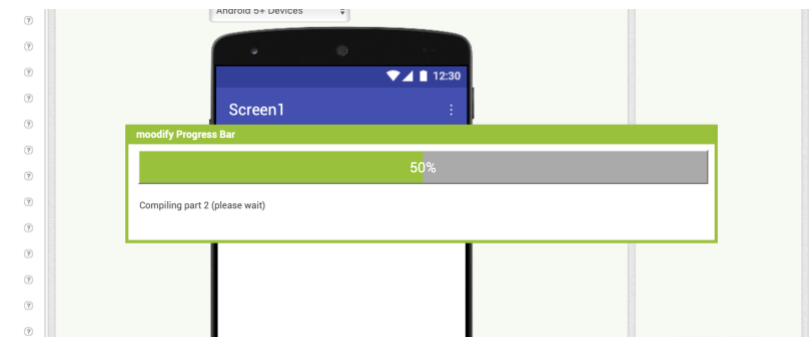
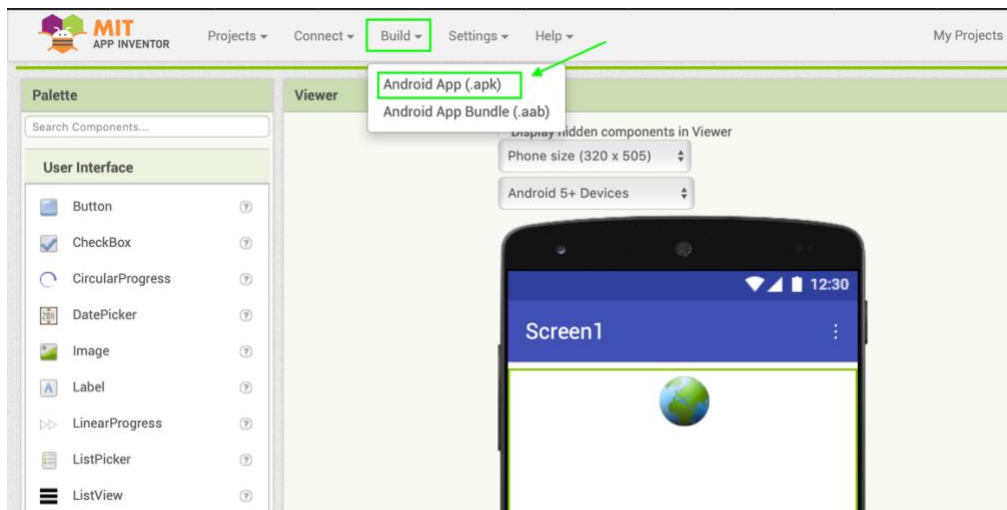
## II. Añadimos el elemento WebView.



## III. Agregamos la URL de nuestra aplicación web.



#### IV. Generamos el apk.



V. Finalmente descargamos el apk.

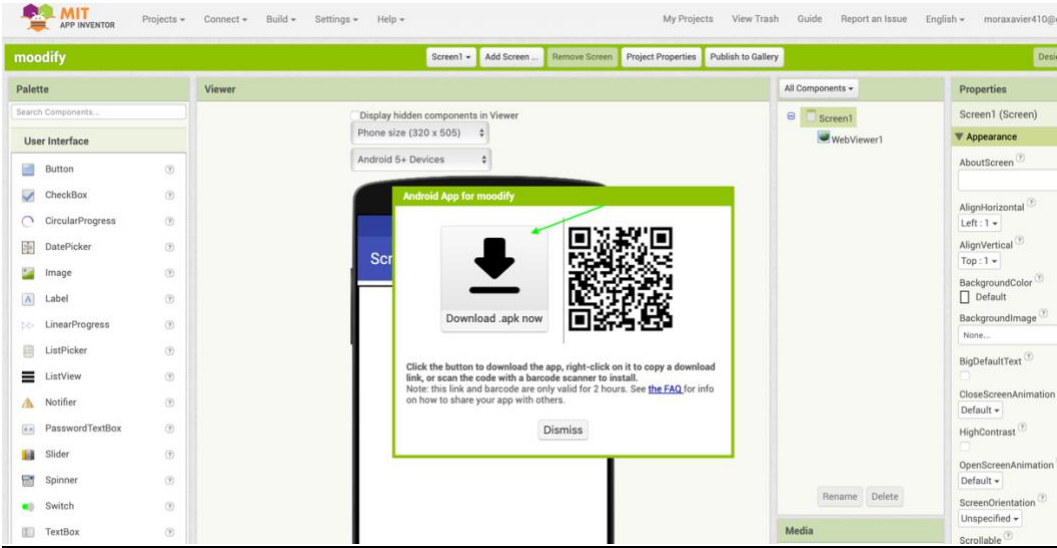
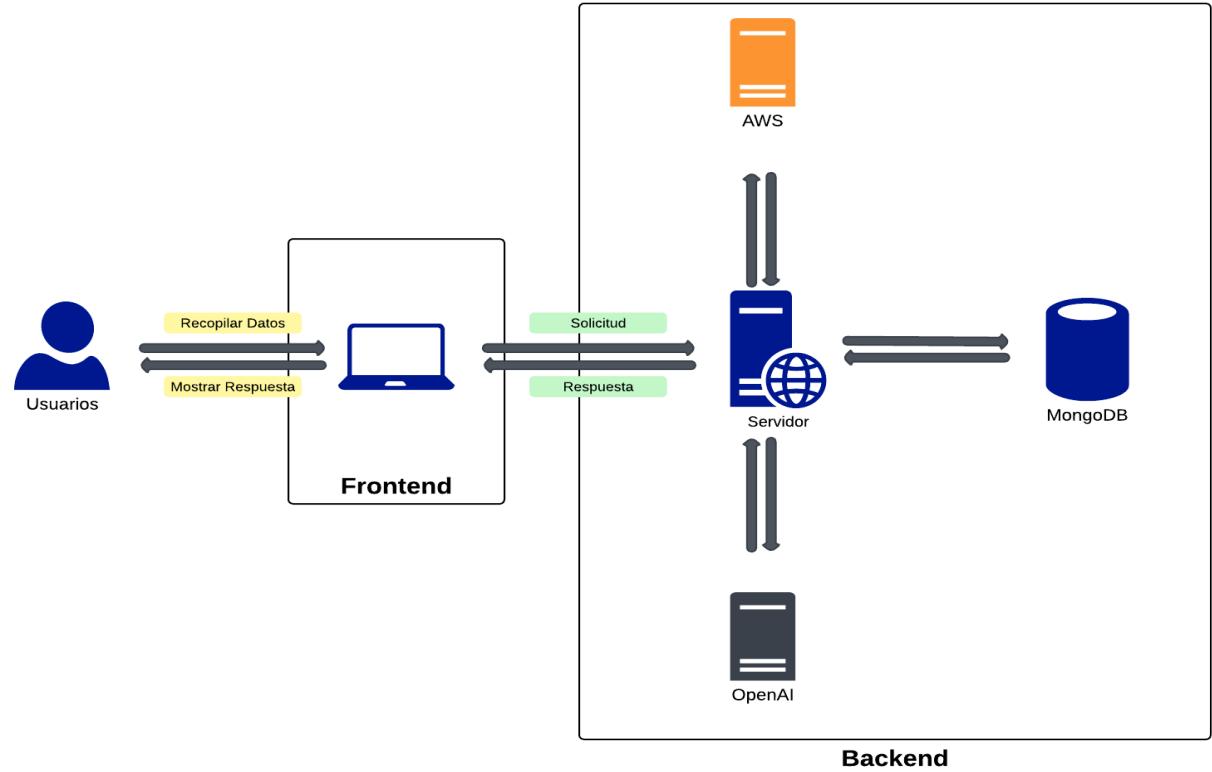


Diagrama de la arquitectura



# Manual de usuario

Apartado que explica cómo utilizar la aplicación, describiendo las funcionalidades principales, pasos para realizar tareas específicas y solución de problemas comunes. Está diseñado para guiar al usuario final en el uso eficaz del sistema.

Al entrar a la [aplicación web](#), accedemos en primer lugar a la **landing page**, aquí el usuario se encuentra con una breve descripción, y la posibilidad de registrarse si no dispone de una cuenta, o de iniciar sesión en el caso de que ya sea usuario de Moodify:

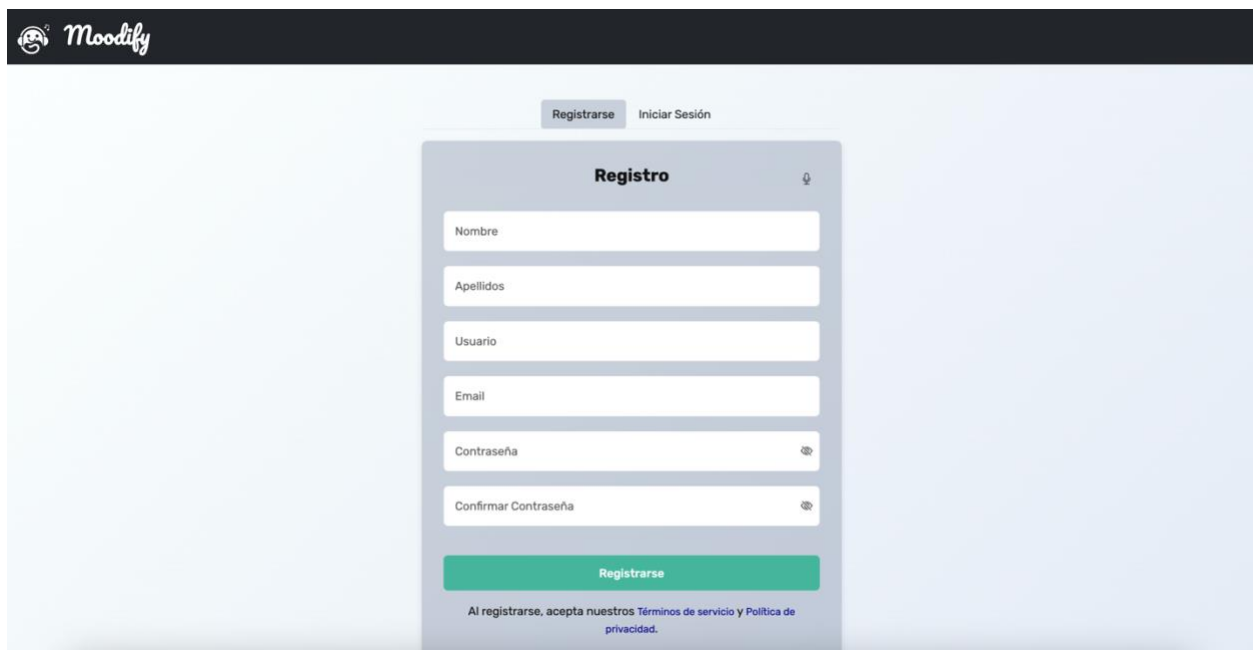


## I- Registro de usuarios

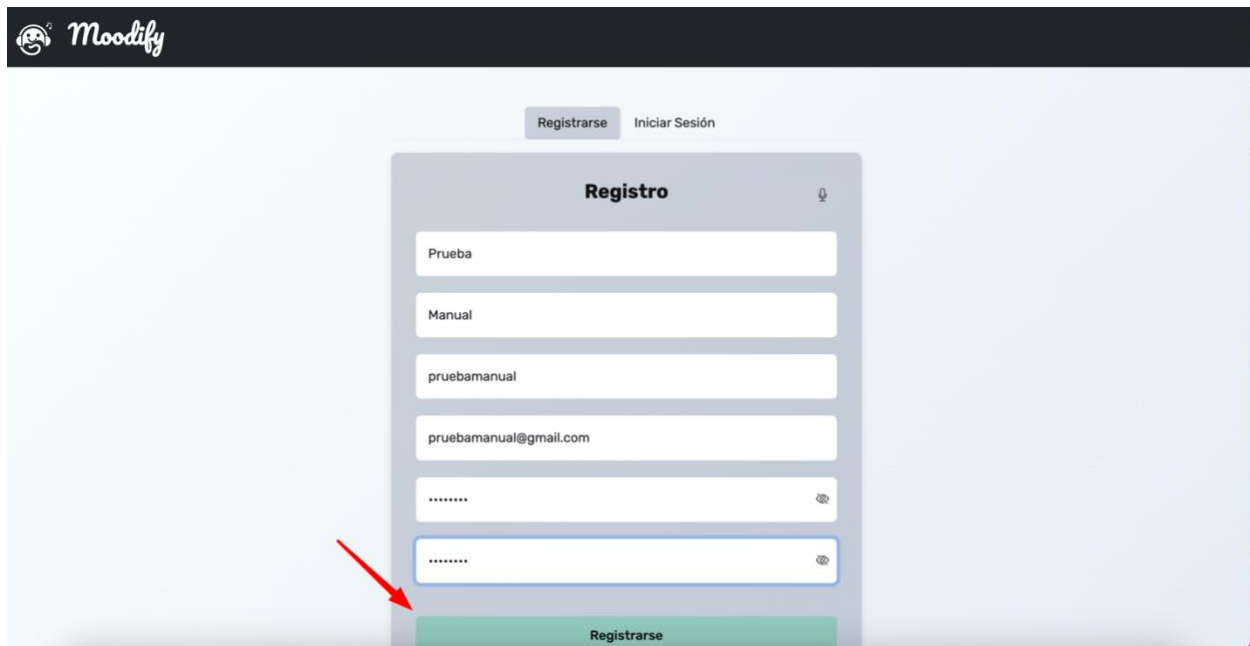
Haciendo click en 'Registrarse':



Se redirige al usuario a la pantalla de registro e inicio de sesión:

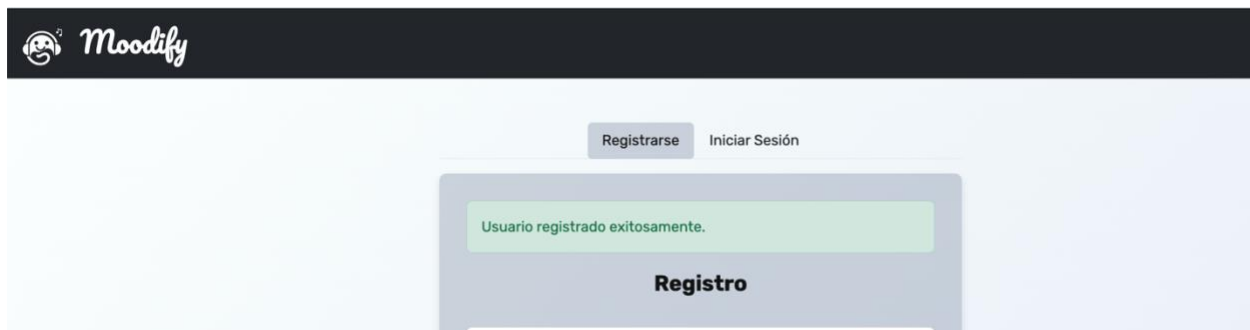


Para registrarse, el usuario debe completar el formulario con los datos solicitados y darle al botón de “Registrarse”:



The screenshot shows the Moodify website's registration page. At the top, there is a dark header with the Moodify logo. Below the header, there are two tabs: "Registrarse" (selected) and "Iniciar Sesión". The main content area features a light blue background with a central registration form titled "Registro". The form contains several input fields: "Prueba", "Manual", "pruebamanual", "pruebamanual@gmail.com", and two password fields represented by asterisks. A red arrow points to the green "Registrarse" button at the bottom of the form.

Si todos los campos introducidos son válidos, el usuario se registrará correctamente y mostrará un mensaje de éxito:



The screenshot shows the Moodify website's registration page after a successful registration. The "Registrarse" tab is still selected. A green message box at the top of the form area displays the text "Usuario registrado exitosamente." Below the message, the "Registro" form title is visible, and the input fields are partially obscured by a white box.

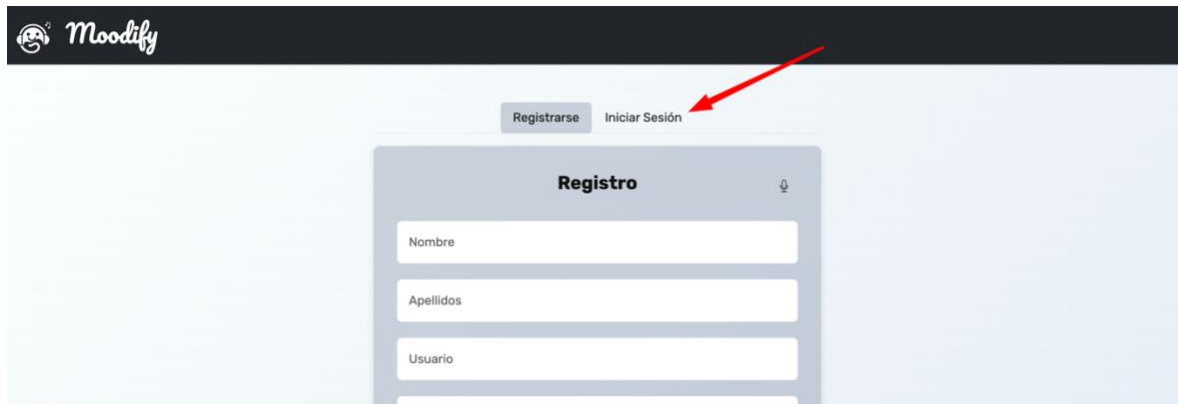


## 2- Inicio de sesión

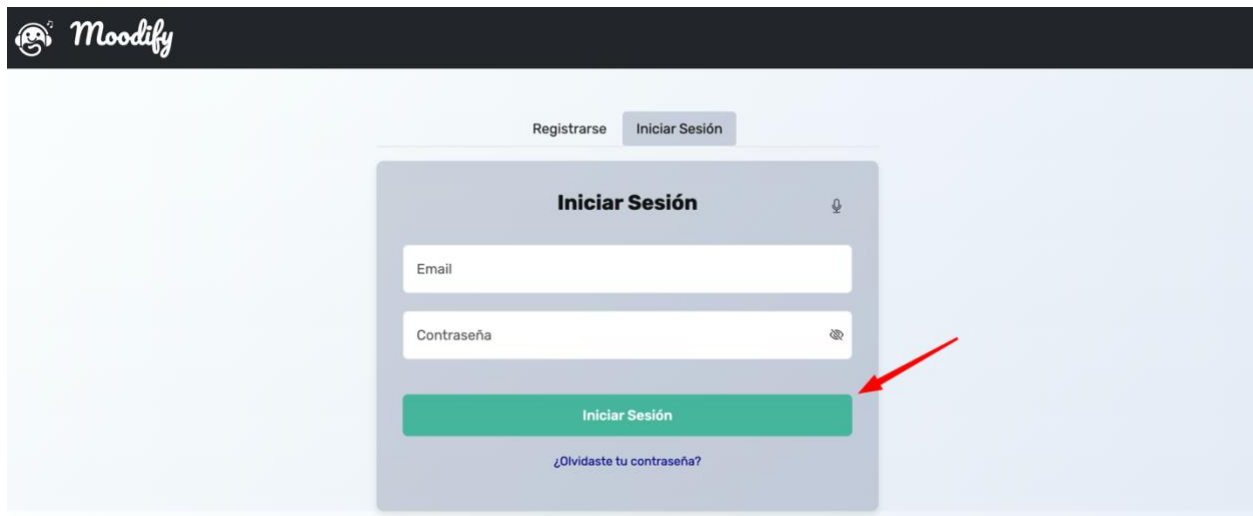
Haciendo click en ‘Iniciar Sesión’:



El usuario es redirigido a la siguiente pantalla, donde debe seleccionar la opción de la derecha:



El usuario accede al formulario de Inicio de Sesión, lo cumplimenta, y hace click sobre el botón de 'Iniciar Sesión'.

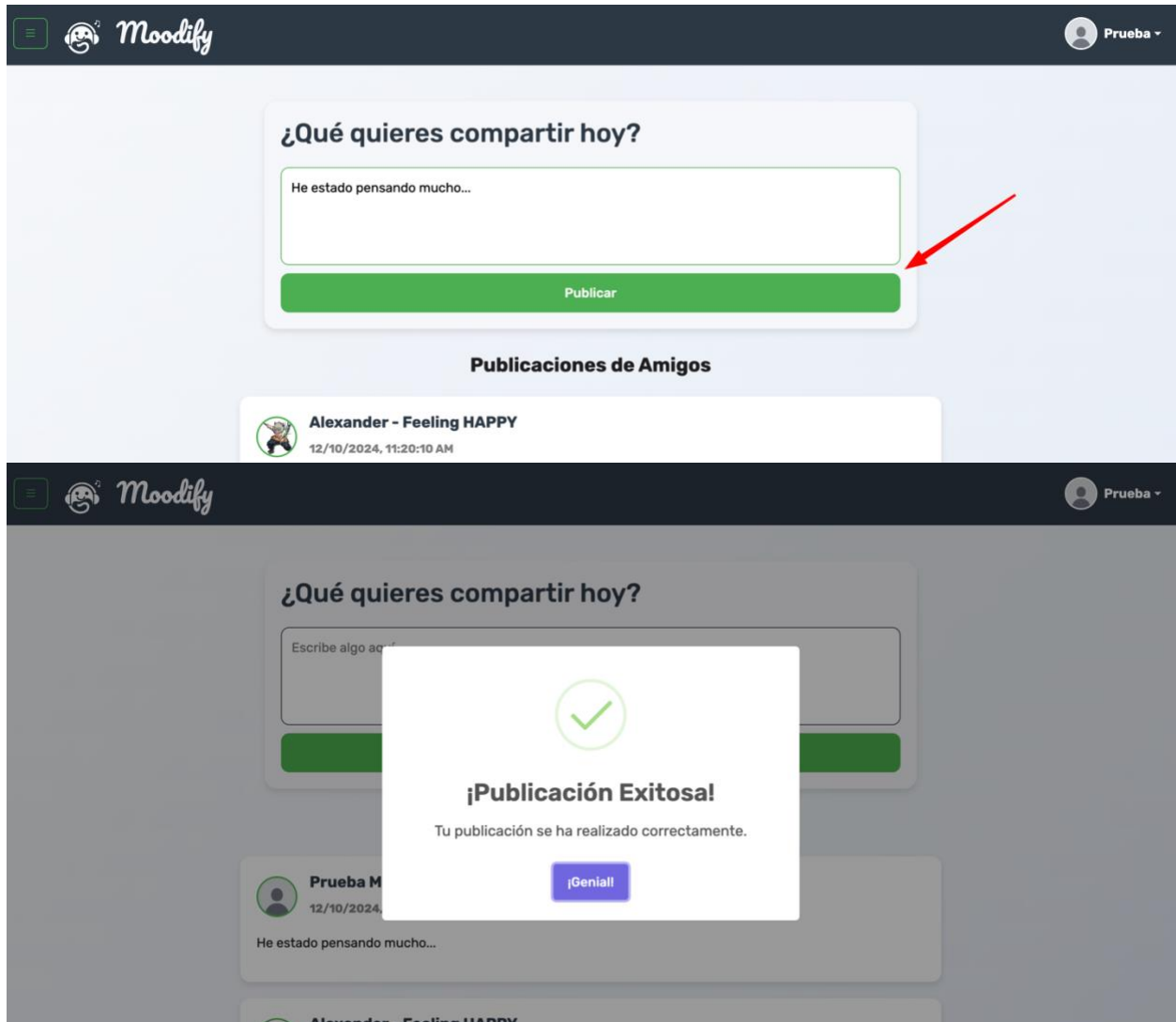


Si los datos son correctos, accederá directamente al **Dashboard** de la aplicación:



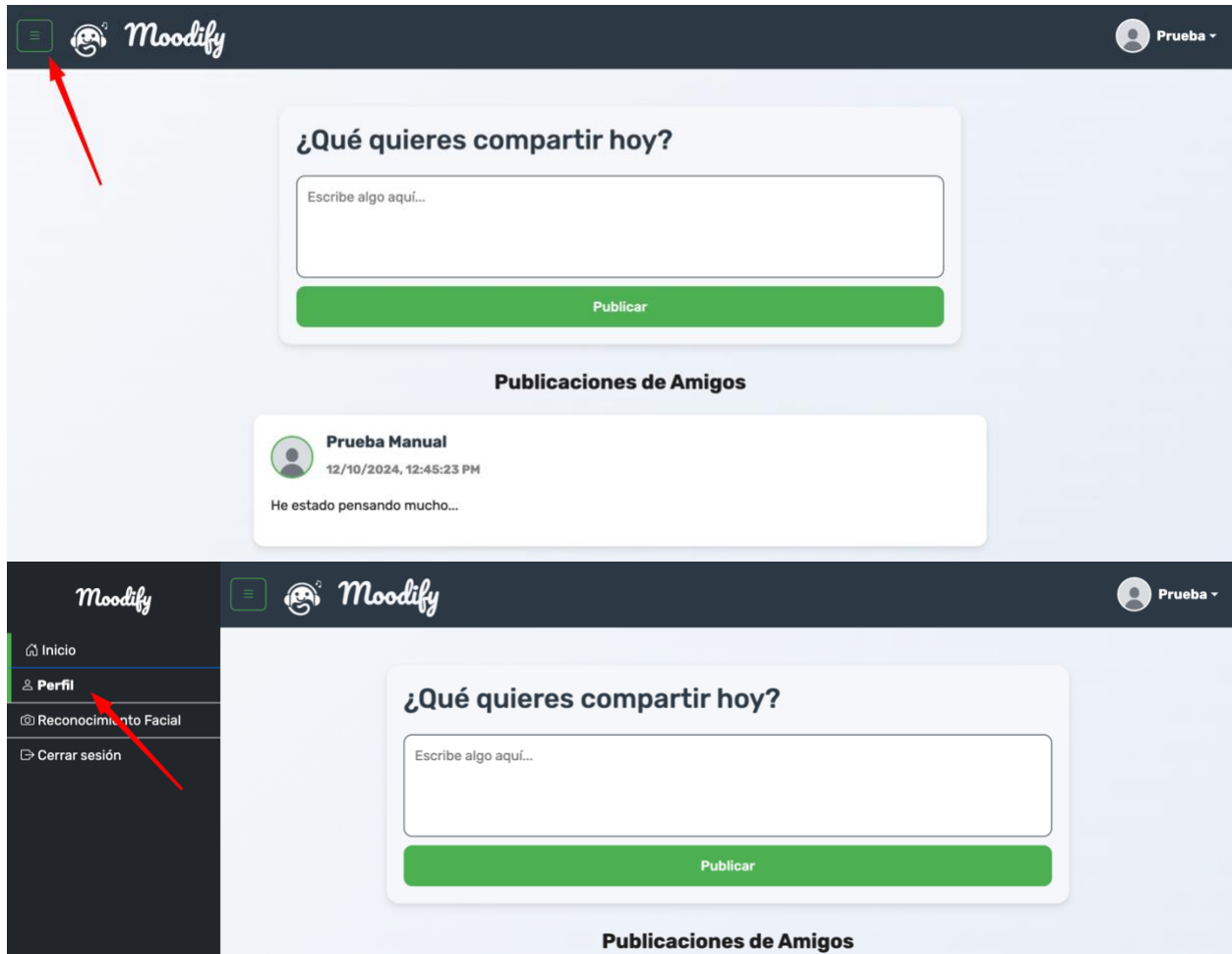
### 3- Dashboard

Desde el **Dashboard** el usuario puede ver las publicaciones de otros usuarios y hacer él mismo publicaciones de estados:

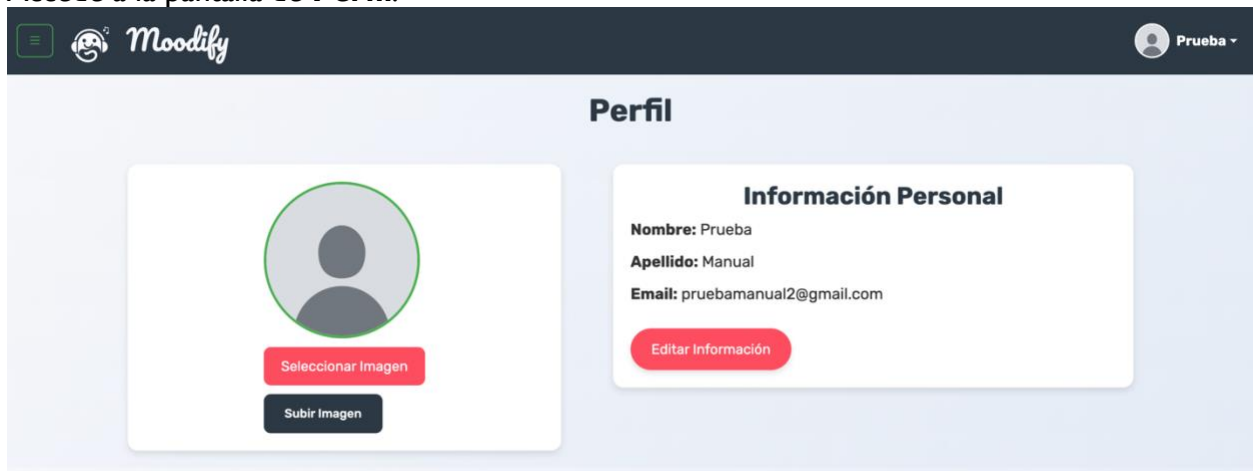


#### 4- Perfil

Desde el Dashboard y haciendo click sobre el icono del menú desplegable de la esquina superior izquierda, es posible también acceder al Perfil del usuario:

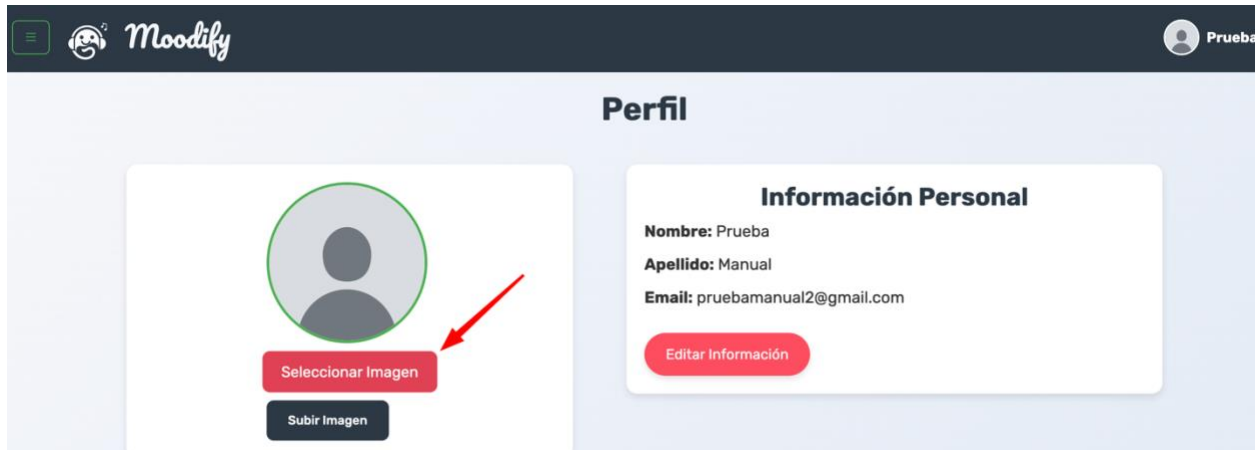


Accede a la pantalla de **Perfil**:

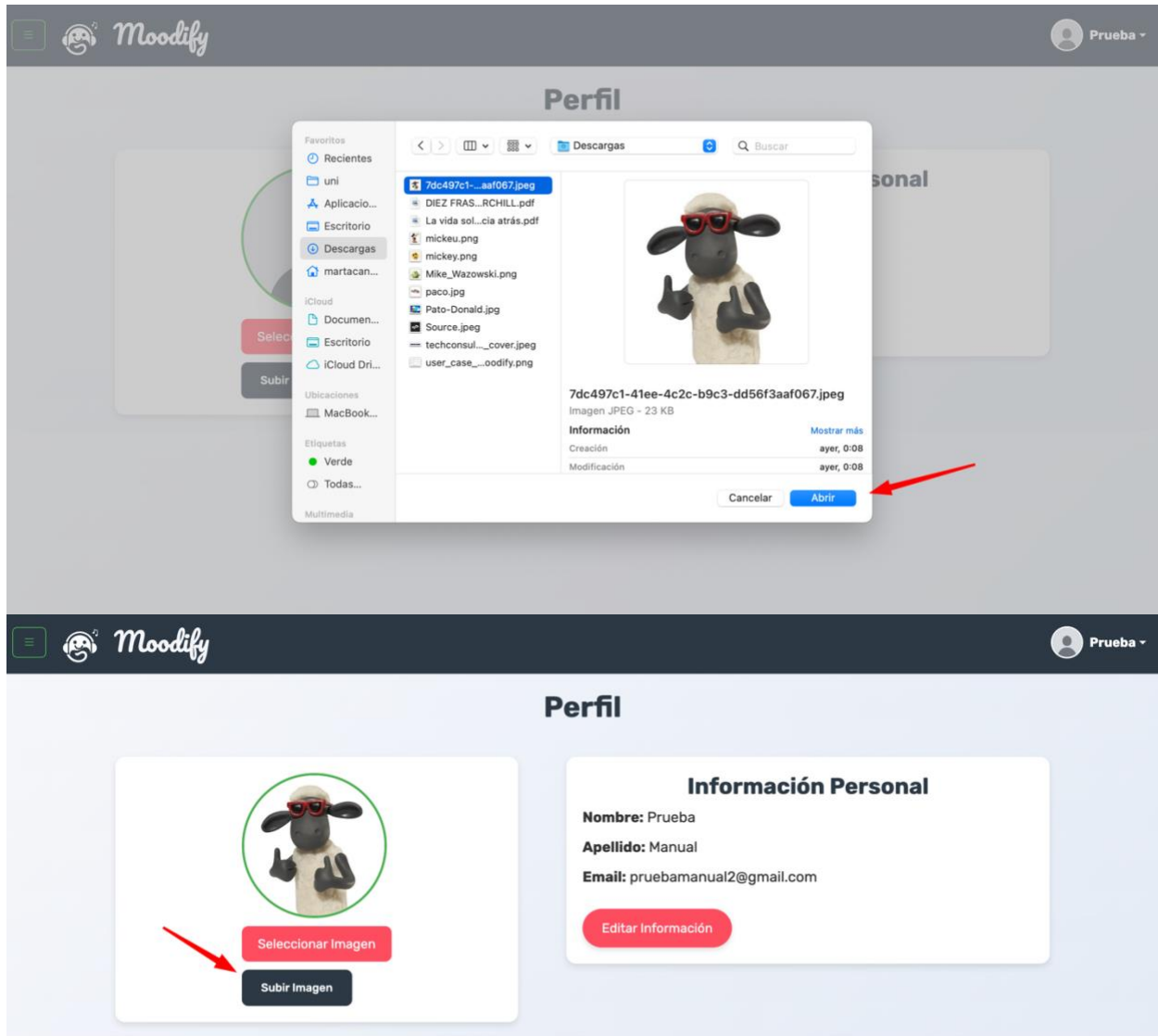


Desde la pantalla de Perfil el usuario puede tanto cambiar su foto de perfil, seleccionando y subiendo un archivo de su propio ordenador, como editar su información personal:

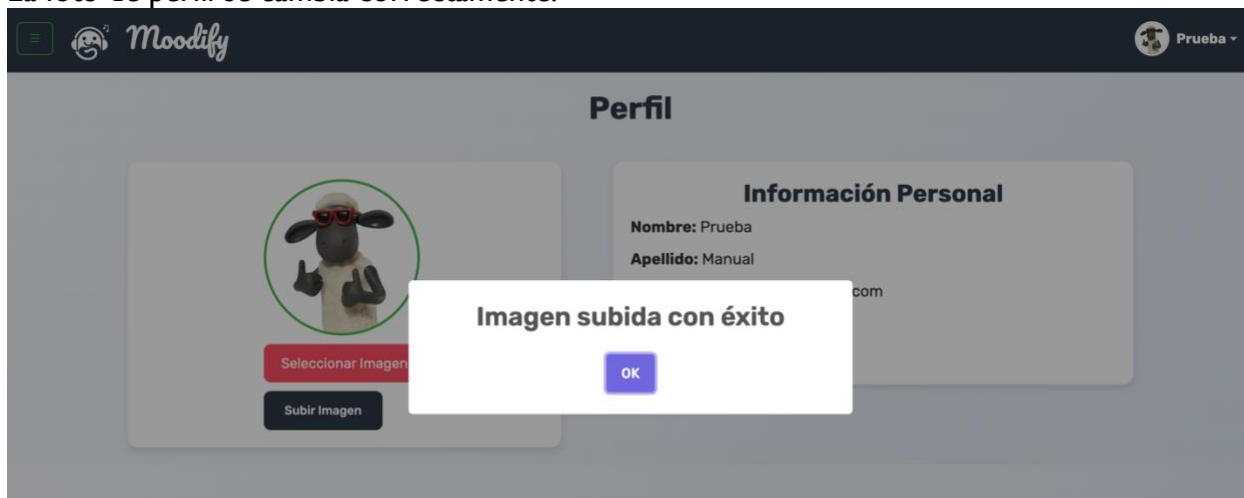
### Cambio de la foto de perfil



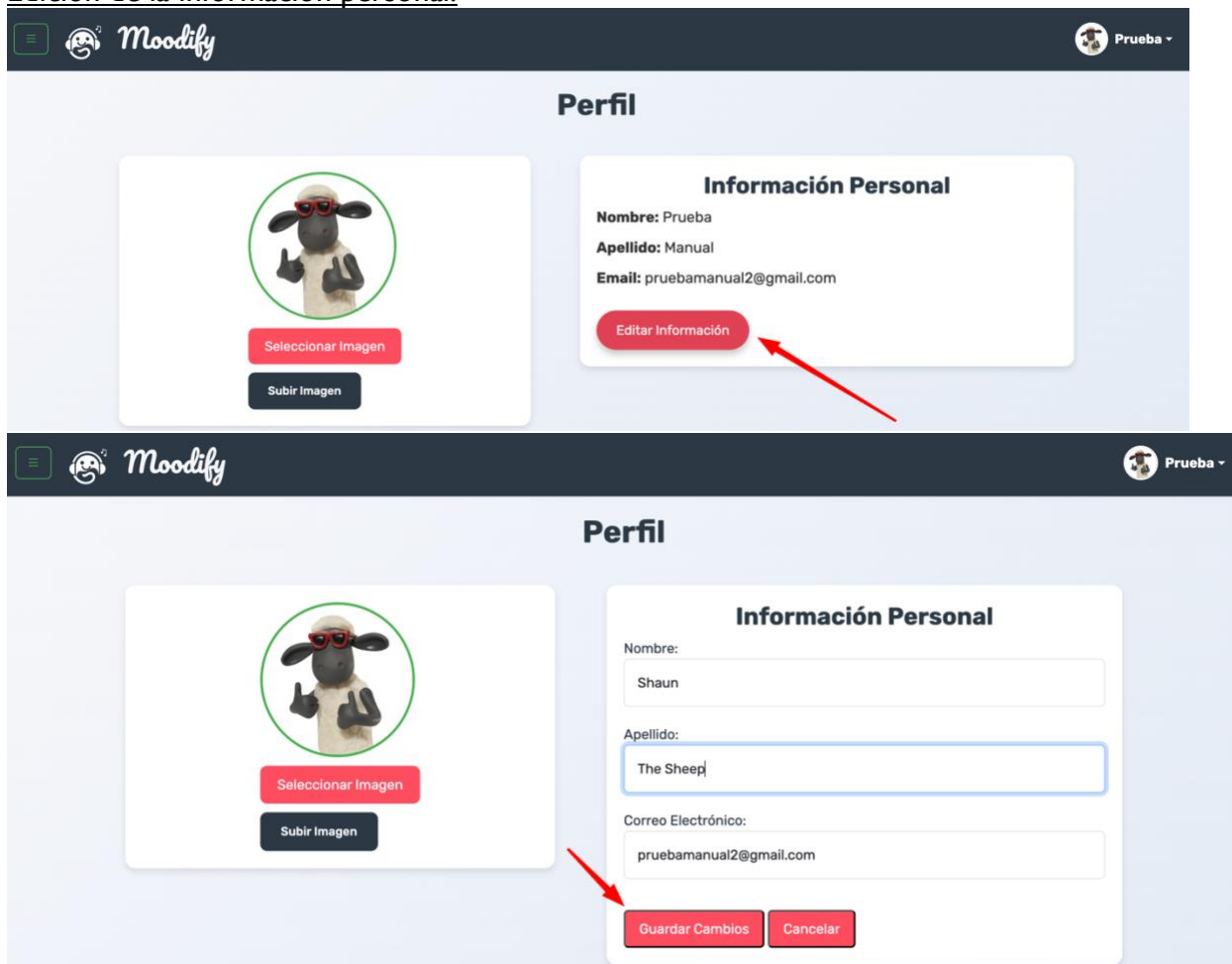
Seleccionamos la imagen de nuestros archivos:

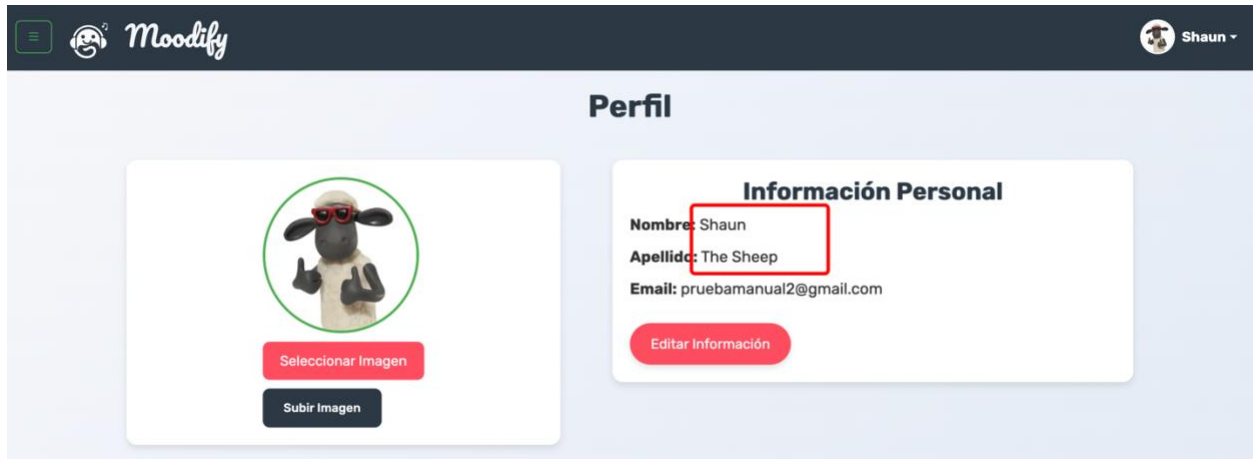


La foto de perfil se cambia correctamente:



Edición de la información personal:





Se muestra un mensaje de éxito:

**Información actualizada  
correctamente.**

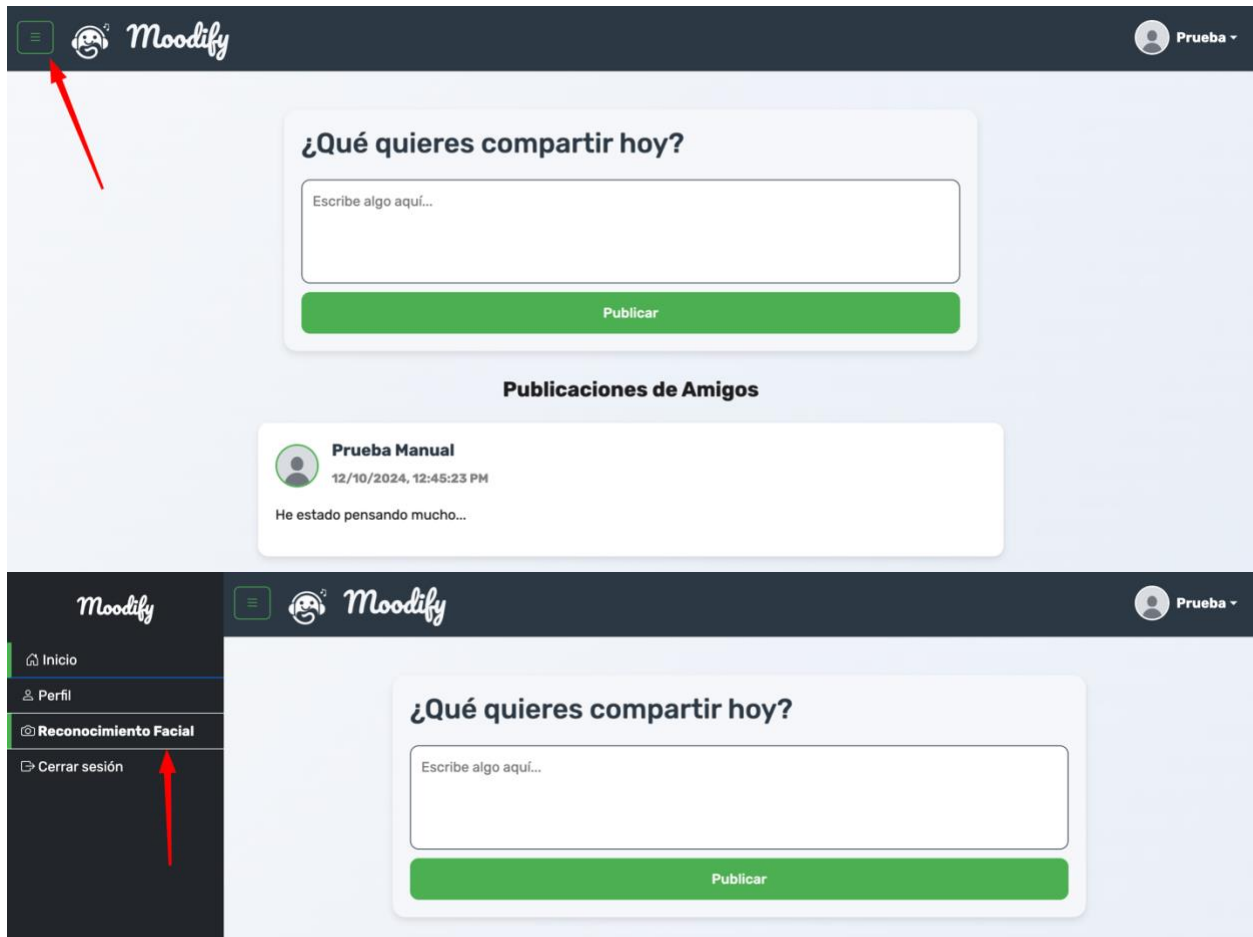
OK

Indica al usuario que su información personal ha sido modificada correctamente.



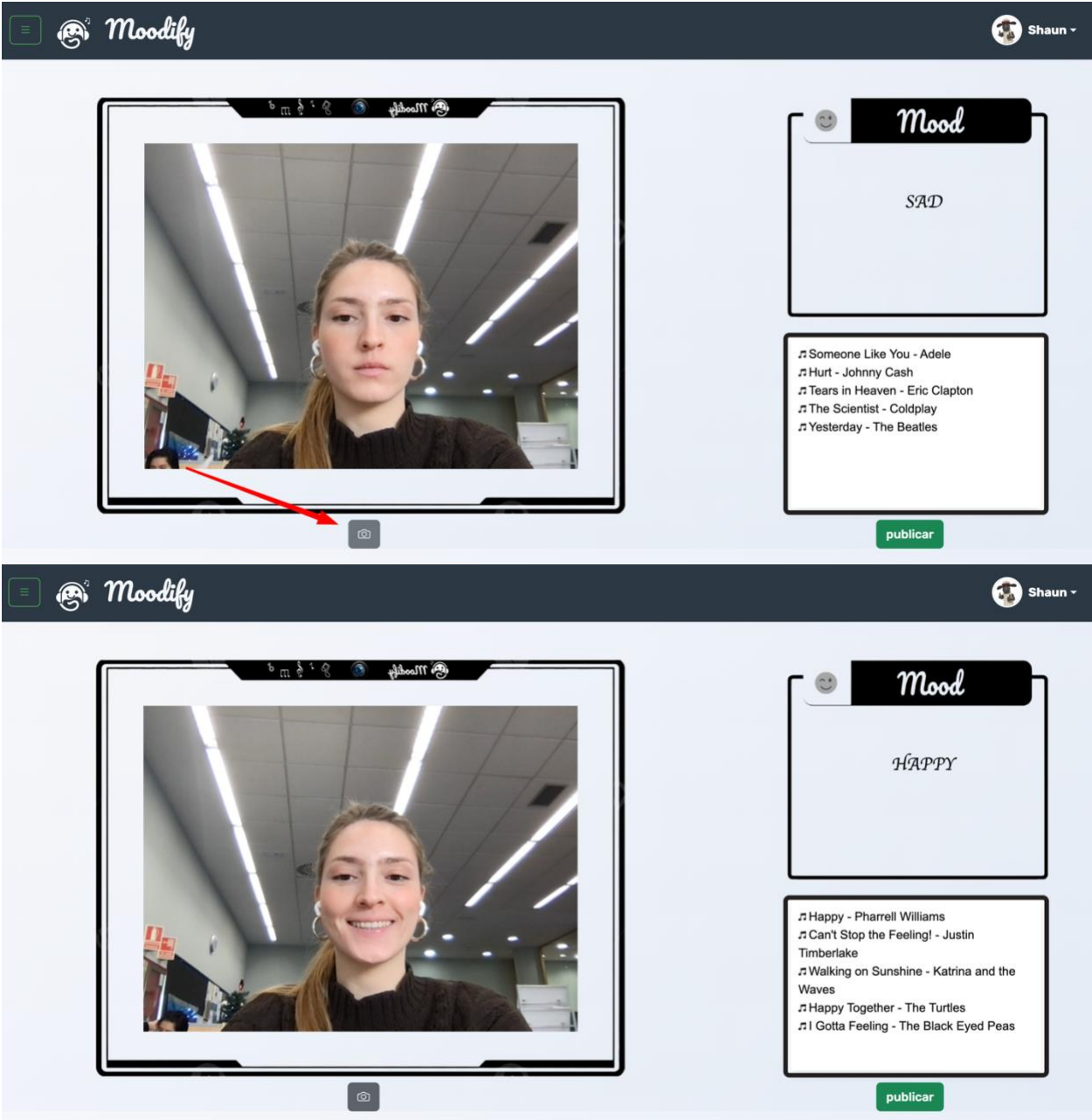
## 5- Reconocimiento facial

Desde el Dashboard y haciendo click sobre el icono del menú desplegable de la esquina superior izquierda, es posible también acceder a la funcionalidad de reconocimiento facial:

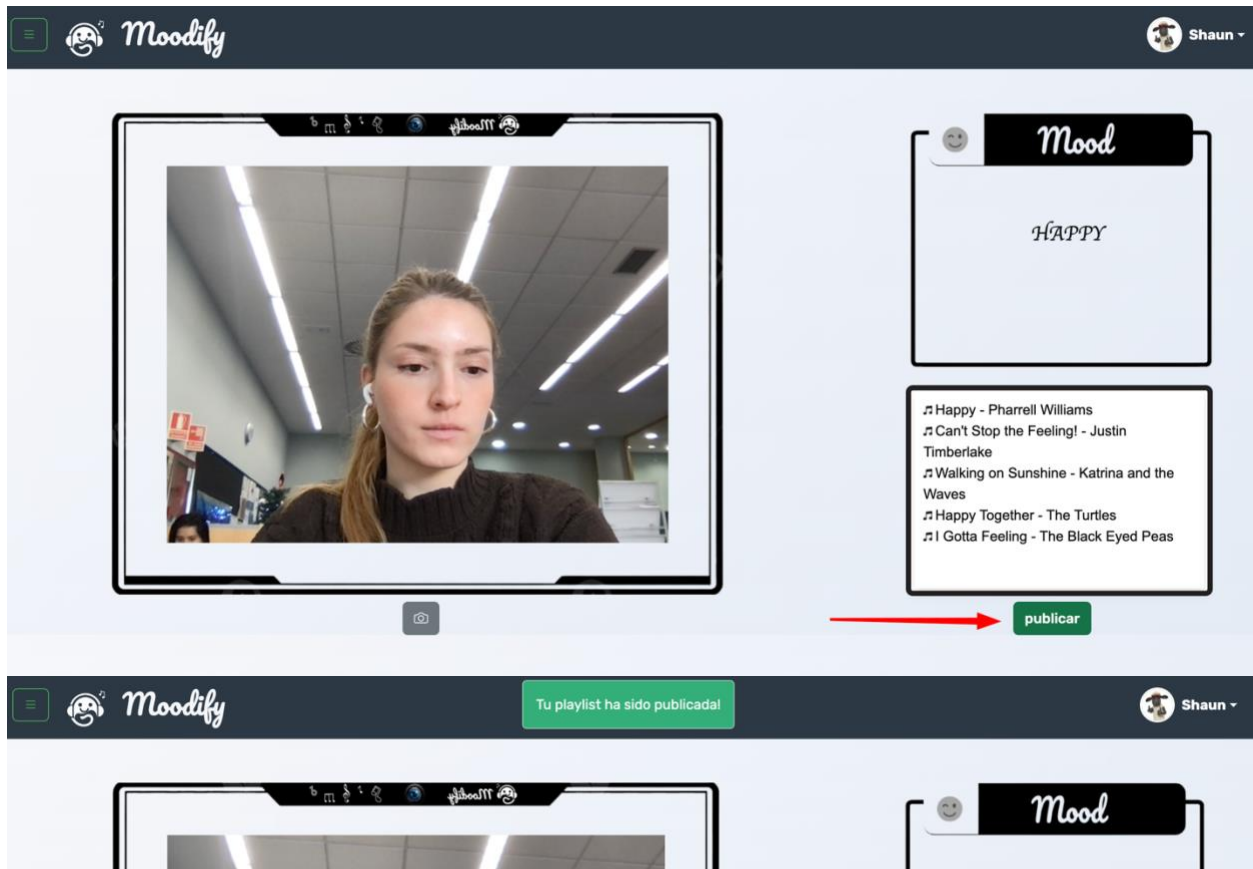


Accedemos a la pantalla de **Reconocimiento Facial**.

Si el usuario se hace una foto, en los paneles de la derecha se le muestra la emoción registrada en su rostro y una recomendación de 3 canciones acordes al estado de ánimo en el que se encuentre:



Para compartir en el **Dashboard** con todos los usuarios su estado de ánimo y la playlist recomendada, hacemos click sobre el botón de “Publicar”:

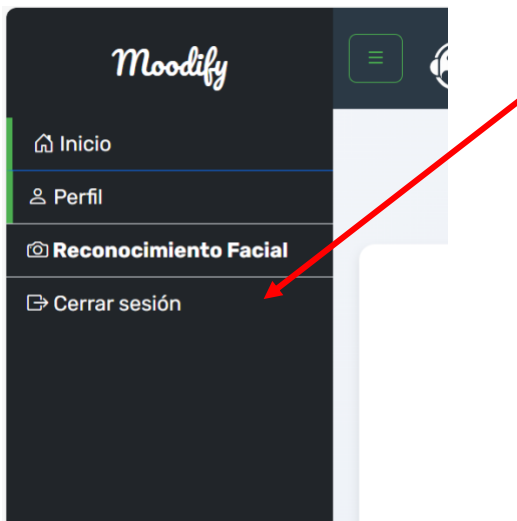


Comprobamos que se ha compartido en el Dashboard:



## 6- Cerrar sesión

Para cerrar sesión, es posible acceder tanto desde el menú desplegable lateral izquierdo, como desde el icono de la flecha en la esquina superior derecha.

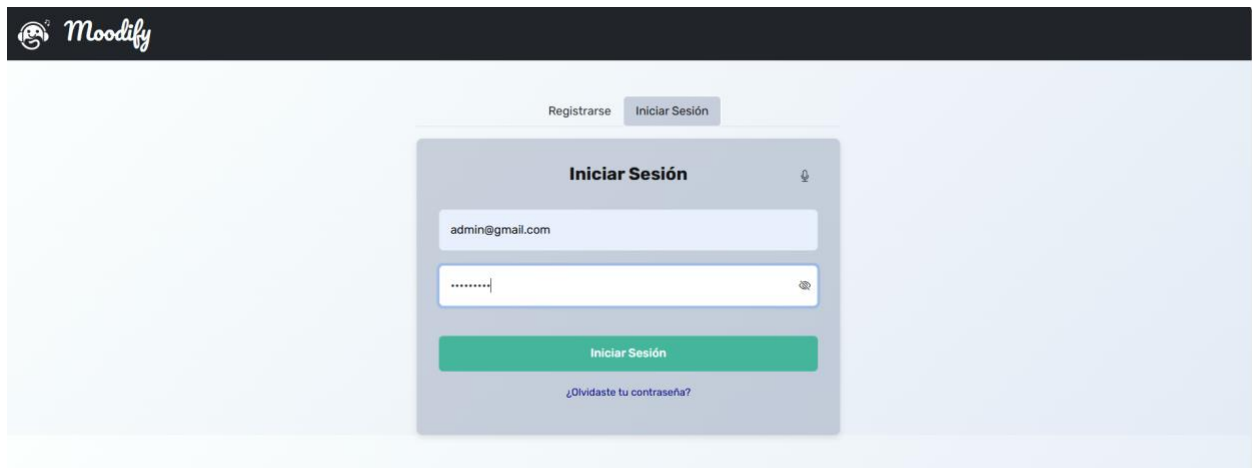


Una vez el usuario haga clic sobre la opción de ‘Cerrar sesión’ le devolverá a la página de inicio:



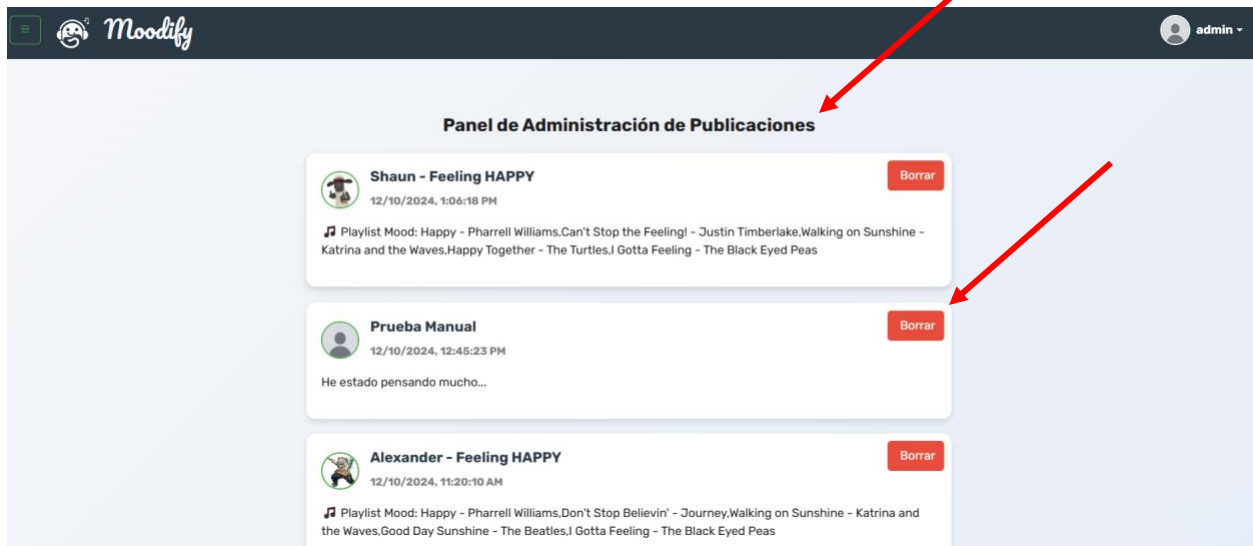
## 7- Administrador

Para acceder como administrador, es necesario introducir en el Inicio de Sesión las **credenciales especiales reservadas** para administradores:

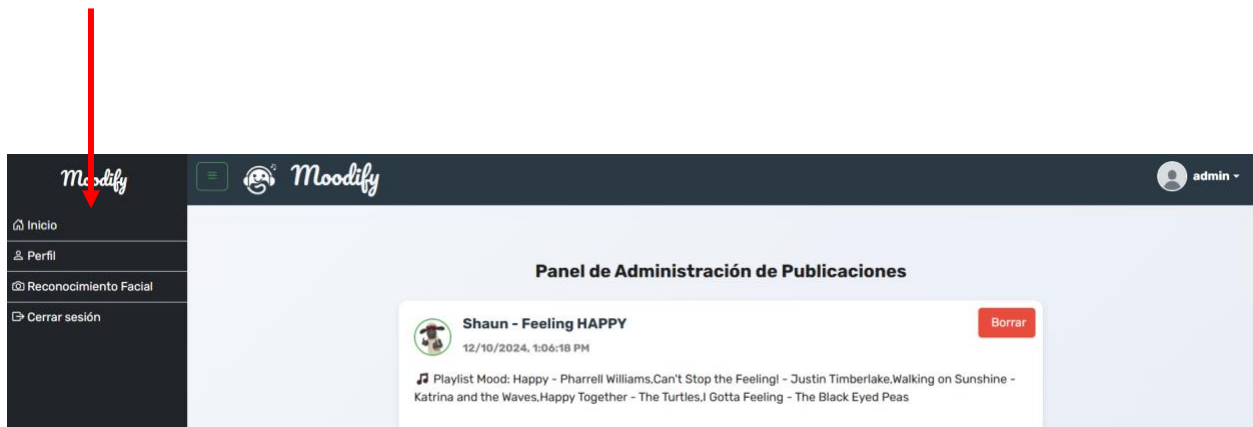


Al haber accedido con éxito, llevará al administrador a una página principal especial o **Dashboard de Administrador**. En el podrá observar:

- 1- El panel de publicaciones de usuarios
- 2- La posibilidad de **borrar/moderar** las publicaciones de los usuarios

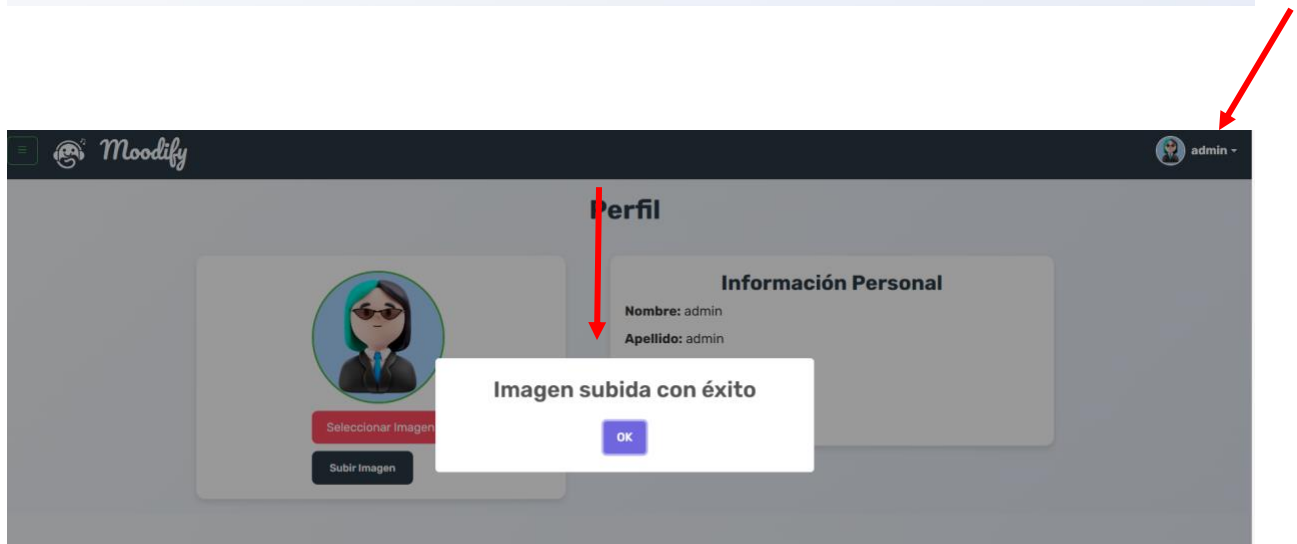


El administrador dispone también de acceso a funcionalidades como las de los usuarios comunes, el reconocimiento facial y la edición del perfil:



### Cambio de foto de perfil

El administrador puede también seleccionar una imagen y poder subirla y cambiarla cuando quiera:



### Modificación de la información personal:

Del mismo modo que un usuario normal, el administrador podría editar su información personal en el apartado de Información Personal del perfil:

