**MissionFet: Plataforma Integral para la Gestión de Proyectos de Grado**

**Manual De Instalación Y Configuración Del Software**

**Autores: Jesus David Mendez Mendez, Andrés Felipe Puentes Rivera, Derek Agmeth Quevedo Santofimio**

**Tutor del Proyecto: Carlos Alberto Lugo Puentes**

**Fundación Escuela Tecnológica de Neiva “Jesús Oviedo Pérez” – FET**

**2025**

**Manual De Instalación Y Configuración Del Software**

**1. Introducción**

* **Nombre del Software**: MissionFet: Plataforma Integral para la Gestión de Proyectos de Grado
* **Versión**: 1.0 (Primera versión oficial)
* **Descripción**: MissionFet es una plataforma web desarrollada para optimizar y digitalizar la gestión de los proyectos de grado dentro de una universidad. Este sistema permite a los estudiantes registrar sus proyectos, subir avances, recibir correcciones, retroalimentación y evaluaciones por parte de los docentes, todo dentro de un entorno virtual accesible y centralizado. Su uso facilita el seguimiento académico, reduce la carga administrativa y mejora la comunicación entre los actores del proceso académico.
* **Objetivo del Manual**: Este manual tiene como propósito guiar paso a paso al usuario en la instalación y configuración inicial del sistema MissionFet en un entorno local. Se ha diseñado para ser comprensible por personas con conocimientos básicos de computación, pero también útil para usuarios con más experiencia técnica.
* **Conocimientos Previos**: Para realizar correctamente la instalación y configuración de MissionFet: *Plataforma Integral para la Gestión de Proyectos de Grado*, es importante contar con algunos conocimientos técnicos básicos. No se requiere experiencia avanzada en informática, pero sí es necesario tener una comprensión mínima de los siguientes aspectos:
  + **Uso básico del sistema operativo Windows**: Navegar por carpetas, copiar y mover archivos, y comprender rutas de archivos (por ejemplo, C:\xampp\htdocs).
  + **Navegación web y descarga de archivos**: Descargar archivos desde Internet, descomprimir archivos ZIP y ejecutar instaladores.
  + **Instalación de software**: Saber cómo instalar programas en Windows, como XAMPP o editores de código.
  + **Manejo básico de Git (opcional pero recomendado)**: Aunque el manual incluirá los pasos para descargar el proyecto desde GitHub, es útil tener conocimientos sobre cómo clonar repositorios utilizando Git, especialmente si se desea actualizar el sistema con futuras versiones o colaborar con el desarrollo.
  + **Conceptos generales de desarrollo web (nivel usuario)**: Comprender que el software funciona sobre tecnologías como:
    - **PHP** (para el procesamiento del lado del servidor)
    - **HTML, CSS y JavaScript** (para la estructura, estilo y funcionalidad del lado del cliente)
    - **XAMPP** (para crear un entorno local con servidor web y base de datos)
    - **Base de datos MySQL** (para almacenar la información del sistema)
    - **Twilio** (opcional, solo si se desea configurar el envío de notificaciones por SMS)
    - **Visual Studio Code (VSCode)** u otro editor de código para posibles ajustes o revisiones del sistema
* **Alcance**: Este manual está diseñado para cubrir la instalación y configuración básica del software *MissionFet* en un entorno local, es decir, dentro del computador del usuario, sin necesidad de servidores externos.
  + **Sistema operativo recomendado:**  
    Este manual está orientado principalmente al sistema Windows, ya que fue la plataforma utilizada durante el desarrollo y pruebas del software. Todos los pasos están explicados con base en este sistema operativo, incluyendo rutas de instalación, configuraciones y herramientas compatibles como XAMPP.  
    Aunque el sistema podría funcionar en otros entornos como Linux o macOS, no se garantiza su funcionamiento completo ni se incluyen instrucciones específicas para esos sistemas, ya que no se realizaron pruebas en ellos.
  + **Tipo de instalación cubierta:**  
    Este documento guía únicamente la instalación básica y configuración inicial del sistema. No se incluyen configuraciones avanzadas como despliegue en servidores web públicos, implementación de medidas de seguridad para producción, personalización del código fuente, ni la integración completa con servicios externos.
  + **Sobre Twilio y WhatsApp:**  
    *MissionFet* incorpora una integración con Twilio, la cual permite el envío de mensajes a través de WhatsApp como medio de notificación para usuarios.  
    Sin embargo, es importante tener en cuenta que esta funcionalidad ha sido implementada y probada únicamente con la versión gratuita de Twilio. El comportamiento del sistema con la versión de pago o cuentas empresariales aún no ha sido verificado, por lo que podrían presentarse diferencias en el funcionamiento, especialmente en términos de entrega de mensajes, límites, formatos o restricciones.
  + **Audiencia del manual:**  
    Este manual está dirigido a cualquier persona que desee instalar y usar la plataforma, incluyendo administradores, personal académico, estudiantes, desarrolladores o encargados del área tecnológica de una institución educativa. El enfoque es didáctico y paso a paso, pensado para que incluso personas con conocimientos mínimos en computación puedan completar exitosamente la instalación.

**2. Requisitos del Sistema**

* **Hardware**:

Para ejecutar MissionFet en un entorno local, se recomienda contar con los siguientes recursos mínimos en el computador:

* + **CPU (Procesador):** Intel Core i3 de 6ª generación o superior / AMD equivalente
  + **RAM (Memoria):** Mínimo 4 GB de RAM (8 GB recomendados para un mejor rendimiento si se trabaja con otras aplicaciones al mismo tiempo)
  + **Espacio en Disco:** Al menos 100 MB disponibles para instalar el software y guardar archivos asociados (el sistema actualmente ocupa aproximadamente 52 MB)
* **Software**:
  + **Sistema Operativo:** Compatible con Windows 10 o superior (64 bits). También puede funcionar en versiones anteriores (como Windows 8.1) o sistemas operativos basados en Linux, siempre y cuando puedan ejecutar PHP, XAMPP o servidores equivalentes. Este manual está enfocado principalmente en la instalación para usuarios de Windows.
  + **Dependencias adicionales:** Asegúrese de contar con lo siguiente:
    - XAMPP: para ejecutar el servidor Apache y MySQL.
    - Composer: obligatorio para instalar y gestionar el SDK de Twilio.
    - SDK de Twilio: instalado mediante Composer (se crea automáticamente la carpeta vendor/ en el proyecto).
    - Acceso a internet: necesario para el uso de la API oficial de Twilio, que permite la autenticación (usando SID y token) y el envío de mensajes de WhatsApp desde la plataforma.
    - Navegador web moderno (Google Chrome, Mozilla Firefox, Microsoft Edge, etc.)
    - Opcional pero recomendado: Visual Studio Code como editor de código.
* **Entorno**:
  + **Lenguaje de programación:** 
    - PHP (para la lógica del servidor)
    - HTML, CSS puro y JavaScript (para la estructura, estilos y funcionalidades de la interfaz)
  + **Servidor web o de aplicaciones:** 
    - Apache (incluido en XAMPP)
    - También puede utilizarse el servidor embebido de PHP mediante el siguiente comando, desde la terminal: “php -S localhost:8000 to\_aplicacion”. Este comando permite ejecutar el sistema desde cualquier carpeta, sin necesidad de mover los archivos al directorio de instalación de XAMPP.

**3. Descarga del Software**

* **Repositorio:** El software MissionFet: Plataforma Integral para la Gestión de Proyectos de Grado se encuentra alojado en la plataforma GitHub, y puede ser accedido mediante el siguiente enlace: <https://github.com/FelipePuentes7/Proyecto_aplicacion.git>

Este repositorio es público, lo que significa que cualquier persona puede acceder a él sin necesidad de iniciar sesión o tener una cuenta en GitHub.

* **Versión recomendada**: Actualmente, la plataforma se encuentra en su primera versión, por lo tanto, esta es la única versión disponible y recomendada para su instalación.
* **Instrucciones para la descarga**: A continuación, se explican dos métodos para descargar el software: uno para usuarios principiantes (sin usar comandos) y otro para usuarios con conocimientos intermedios (usando Git).
  + **Opción 1:** Descarga directa desde el navegador (recomendado para principiantes)
    - Abre tu navegador de preferencia (Google Chrome, Microsoft Edge, Firefox, etc.).
    - Ingresa a la siguiente dirección: <https://github.com/FelipePuentes7/Proyecto_aplicacion>
    - Una vez dentro del repositorio, haz clic en el botón verde que dice "Code".



* + - En el menú desplegable, selecciona "Download ZIP".



* + - Se descargará un archivo comprimido con el nombre Proyecto\_aplicacion-main.zip (puede variar ligeramente).
    - Una vez descargado, extrae el contenido del archivo ZIP haciendo clic derecho sobre él y seleccionando la opción "Extraer todo" o "Descomprimir", dependiendo de tu sistema operativo.
  + **Opción 2:** Clonar el repositorio usando Git (recomendado para usuarios con conocimientos técnicos)
    - Para este método debes tener instalado Git en tu computadora.
    - Abre el programa Git Bash o cualquier terminal que tengas instalada.
    - Navega a la carpeta donde deseas guardar el proyecto. Por ejemplo: “cd C:\Users\TuUsuario\Documents”
    - Ejecuta el siguiente comando: “git clone <https://github.com/FelipePuentes7/Proyecto_aplicacion.git>”
    - Este comando descargará todos los archivos del repositorio en una nueva carpeta llamada Proyecto\_aplicacion.
  + **Nota importante:** Después de descargar el software (por cualquiera de los métodos), asegúrate de tener instalado XAMPP y de seguir los pasos de instalación y configuración que se indican en las siguientes secciones del manual.

**4. Instalación del Software**

* Antes de comenzar, asegúrate de tener lo siguiente:
  + Conexión a internet (para descargar dependencias y acceder a la API de Twilio).
  + Acceso a la terminal o consola de comandos del sistema.
  + Un editor de texto recomendado: Visual Studio Code (opcional, pero útil).
  + Tener configurado correctamente el token de Twilio, SID, AUTH TOKEN y número de teléfono verificado en los archivos correspondientes del proyecto (esto se explica en el manual de configuración más adelante).

**4.1 Instalación en Windows**

* **Paso 1: Instalar XAMPP (incluye PHP y MySQL)**
  + Ve a: <https://www.apachefriends.org/es/index.html>
  + Descarga la versión más reciente de **XAMPP para Windows** (que ya incluye PHP y MySQL).
  + Instálalo siguiendo los pasos, seleccionando los módulos Apache, MySQL y PHP.
  + **Abre el Panel de Control de XAMPP y haz clic en “Start” en los módulos Apache y MySQL.**
* **Paso 2: Instalar Git**
  + Descarga desde: <https://git-scm.com/download/win>
  + Instálalo con las opciones predeterminadas.
  + Abre la terminal y escribe: “git --version”. Para verificar que Git esté funcionando.
* **Paso 3: Instalar Composer (Gestor de dependencias de PHP)**
  + Ve a: <https://getcomposer.org/download/>
  + Descarga el instalador para Windows.
  + Durante la instalación, Composer detectará automáticamente PHP desde XAMPP (si no, puedes indicarle manualmente la ruta, normalmente: C:\xampp\php\php.exe).
  + Abre una terminal y verifica: “composer --version”
* **Paso 4: Clonar el proyecto**
  + Abre la terminal (CMD o Git Bash).
  + Navega a una carpeta donde quieras guardar el proyecto: “cd C:\Users\TuUsuario\Proyectos”
  + Clona el repositorio: “git clone <https://github.com/FelipePuentes7/Proyecto_aplicacion.git>”
* **Paso 5: Instalar dependencias del proyecto (Twilio)**
  + Entra en la carpeta del proyecto: “cd Proyecto\_aplicacion”
  + Instala Twilio usando Composer: “composer require twilio/sdk”. Esto creará automáticamente las carpetas vendor/ y los archivos composer.lock y composer.json.
* **Paso 6: Ejecutar el servidor local**
  + Desde la misma carpeta: “php -S localhost:8000 -t to\_aplicacion”
  + Abre tu navegador y visita: “http://localhost:8000”

**4.2 Instalación en Linux (Ubuntu recomendado)**

* **Paso 1: Instalar PHP, Git y Composer**
  + En la terminal, ejecuta: “sudo apt update

sudo apt install php php-cli git curl unzip php-mbstring php-xml php-curl -y”

* + Luego instala Composer: “sudo apt install composer -y”
  + Verifica ambas herramientas: “php -v

composer --version”

* **Paso 2: Clonar el proyecto**
  + git clone https://github.com/FelipePuentes7/Proyecto\_aplicacion.git
  + cd Proyecto\_aplicacion
* **Paso 3: Instalar el SDK de Twilio**
  + composer require twilio/sdk
* **Paso 4: Ejecutar el servidor local**
  + php -S localhost:8000 -t to\_aplicacion
  + En el navegador, ingresa a: “http://localhost:8000”

**4.3 Instalación en macOS**

* **Paso 1: Instalar Homebrew (si no lo tienes)**
  + /bin/bash -c "$(curl -fsSL <https://raw.githubusercontent.com/Homebrew/install/HEAD/install.sh>)"
* **Paso 2: Instalar PHP, Git y Composer**
  + brew install php git composer
* **Paso 3: Clonar el repositorio**
  + git clone https://github.com/FelipePuentes7/Proyecto\_aplicacion.git
  + cd Proyecto\_aplicacion
* **Paso 4: Instalar dependencias del proyecto**
  + composer require twilio/sdk
* **Paso 5: Ejecutar servidor local**
  + php -S localhost:8000 -t to\_aplicacion
  + Y accede desde tu navegador a: “http://localhost:8000”

**5. Configuración Inicial**

* **Configuración de la base de datos**:
  + **Paso 1: Iniciar el servidor MySQL en XAMPP**
    - Abre XAMPP.
    - Da clic en el botón "Start" del módulo MySQL (y también Apache, si no está iniciado).
    - Asegúrate de que ambos tengan el fondo verde.
  + **Paso 2: Acceder a phpMyAdmin**
    - Abre tu navegador web.
    - Escribe en la barra de direcciones: “<http://localhost/phpmyadmin>”
    - Presiona Enter.
  + **Paso 3: Crear la base de datos**
    - Dentro de phpMyAdmin, da clic en el botón "Nueva" en el panel izquierdo.
    - En el campo Nombre de la base de datos, escribe: “proyecto”
    - Selecciona el cotejamiento utf8\_general\_ci (si no está seleccionado).
    - Da clic en Crear.
  + **Paso 4: Importar la base de datos**
    - Una vez creada la base de datos, da clic sobre su nombre en el panel izquierdo.
    - Luego da clic en la pestaña Importar.
    - Presiona el botón Elegir archivo y selecciona el archivo que se encuentra en el proyecto en la siguiente ruta: “database/proyecto\_.sql”
    - Asegúrate de que el formato seleccionado sea SQL.
    - Da clic en el botón Continuar para importar la base de datos.
* **Configuración de las credenciales de conexión**
  + El archivo que gestiona la conexión con la base de datos se encuentra en: “config/conexion.php”
  + Revisa que las siguientes líneas estén configuradas correctamente, según los parámetros de tu servidor MySQL:  
    “<?php

$host = 'localhost';

$dbname = 'proyecto';

$username = 'root'; // Cambiar si tu usuario es diferente

$password = ''; // Coloca tu contraseña si configuraste una

try {

$conn = new PDO("mysql:host=$host;dbname=$dbname", $username, $password);

$conn->setAttribute(PDO::ATTR\_ERRMODE, PDO::ERRMODE\_EXCEPTION);

} catch (PDOException $e) {

die("Error en la conexión: " . $e->getMessage());

}

?>”

* + Si tu servidor MySQL tiene un usuario y contraseña diferente, reemplázalos en las variables $username y $password.
* **Configuración del servidor**:
  + **Opción A: Usar la terminal de Visual Studio Code**
    - Abre Visual Studio Code.
    - Abre la carpeta del proyecto (Archivo > Abrir carpeta).
    - Abre una nueva terminal dentro de VS Code (Terminal > Nueva Terminal).
    - Ejecuta el siguiente comando: “php -S localhost:8000 -t to\_aplicacion” Asegúrate de que “to\_aplicacion” sea el nombre de la carpeta raíz de tu aplicación.
  + **Opción B: Mover el proyecto a htdocs y usar navegador**
    - Copia la carpeta del proyecto en: “C:\xampp\htdocs\”
    - Abre tu navegador y entra a: “http://localhost/NombreCarpeta”

**6. Ejecución del Software**

* **Opción A: Ejecución desde terminal con servidor embebido de PHP**
  + **Paso 1: Abrir la terminal**
    - Abre Visual Studio Code o cualquier terminal de tu sistema operativo.
    - Accede a la carpeta raíz del proyecto usando el comando cd. Por ejemplo: “cd C:\ruta\del\proyecto”
  + **Paso 2: Ejecutar servidor local de PHP**
    - En la terminal, ejecuta el siguiente comando (ajusta to\_aplicacion si tu carpeta tiene otro nombre): “php -S localhost:8000 -t to\_aplicacion”
  + **Paso 3: Acceder desde el navegador**
    - Abre tu navegador web y visita: <http://localhost:8000>
  + **Comprobación**
    - Si todo está correctamente configurado, deberías ver la interfaz principal de la aplicación.
    - En caso de error, verifica que la base de datos esté corriendo y que el archivo conexion.php tenga los datos correctos.
* **Opción B: Ejecución desde XAMPP y navegador**
  + **Paso 1: Mover la carpeta del proyecto**
    - Copia la carpeta del proyecto a la ruta: “C:\xampp\htdocs\”
  + **Paso 2: Iniciar servicios en XAMPP**
    - Abre el panel de control de XAMPP.
    - Inicia los módulos de Apache y MySQL.
  + **Paso 3: Acceder desde el navegador**
    - Abre tu navegador web.
    - Escribe la siguiente URL, ajustando NombreCarpeta al nombre real del proyecto: “<http://localhost/NombreCarpeta>”
  + **Comprobación**
    - Deberías visualizar la página de inicio de la aplicación.
    - Si aparece algún error, asegúrate de que los servicios de XAMPP están activos y que la base de datos fue correctamente importada.

**7. Configuración Avanzada**

* **Roles de usuario y permisos**: La plataforma MissionFet está diseñada para manejar tres tipos de usuarios:
  + **Administrador:** Encargado de aprobar o rechazar registros, gestionar usuarios y supervisar el funcionamiento general.
  + **Tutor:** Realiza seguimiento, evaluación y retroalimentación de los proyectos.
  + **Estudiante:** Registra su proyecto y sube avances para revisión.
  + **Cómo se configuran los roles:**
    - Al momento de registrarse, el usuario debe indicar su rol (estudiante o tutor).
    - El administrador recibirá la solicitud de registro y, desde el panel de administración, podrá aprobar o rechazar la cuenta.
    - Los roles están definidos tanto en la base de datos (tabla de usuarios) como en el código fuente, y se utilizan para mostrar distintas interfaces y permisos según el tipo de usuario.
    - Asegúrate de tener una cuenta de tipo administrador creada desde la base de datos o configurada manualmente en caso de que no exista aún.
* **Integraciones externas**: MissionFet utiliza la API de Twilio de forma gratuita para el envío de mensajes de WhatsApp cuando un usuario es aprobado. Esta funcionalidad permite enviar una notificación automática para avisar que la cuenta ha sido habilitada.
  + **Pasos ya realizados en la configuración estándar:**
    - Se instaló el SDK de Twilio mediante Composer.
    - Se configuraron las credenciales (SID, Auth Token y número verificado) dentro de los archivos PHP correspondientes.
    - La integración se realiza a través de una petición API desde el backend, utilizando la librería oficial de Twilio.
  + **Nota:** Esta integración ha sido probada únicamente con la versión gratuita de Twilio. La compatibilidad con la versión paga puede variar, especialmente en cuanto a:
    - Límite de mensajes.
    - Validación de números no verificados.
    - Restricciones geográficas o de contenido.

Para mayor seguridad, revisa las políticas de uso de Twilio antes de hacer la transición a un plan pago.

* **Ajustes adicionales**: Aunque el sistema está diseñado para ser funcional desde su instalación básica, hay algunas recomendaciones para mejorar su seguridad y rendimiento:
  + **Seguridad:**
    - **Contraseña de usuario:** Se solicita en el inicio de sesión, pero se recomienda implementar un algoritmo de cifrado como password\_hash() en futuras versiones.
    - **Validación de formularios:** Se debe asegurar que todos los datos enviados por el usuario sean validados en el servidor para evitar inyecciones SQL o XSS.
    - **Control de sesiones:** Las sesiones PHP (session\_start()) ya están implementadas en los archivos de autenticación. Asegúrate de cerrarlas correctamente con session\_destroy() en el logout.
  + **Rendimiento:** Actualmente no se usan técnicas avanzadas de optimización (como caché o paginación), pero si el número de usuarios crece, se recomienda:
    - Aplicar paginación en tablas que muestren muchos registros.
    - Optimizar las consultas SQL con índices adecuados en campos de búsqueda frecuentes.
    - Utilizar archivos comprimidos (CSS/JS) para reducir tiempos de carga.

**8. Actualizaciones y Mantenimiento**

Para mantener la plataforma MissionFet actualizada, estable y segura, es importante seguir una serie de pasos antes y después de aplicar cambios en el sistema. Esta sección detalla cómo actualizar el software, realizar copias de seguridad y mantener el sistema en óptimas condiciones.

* **Paso 1: Actualizar el Software:** Si se han realizado cambios en el repositorio de GitHub (como nuevas funcionalidades o correcciones de errores), se pueden aplicar en el entorno local con el siguiente comando: “git pull origin master”. Asegúrate de estar dentro del directorio del proyecto antes de ejecutar este comando.
* **Paso 2: Realizar Copia de Seguridad de la Base de Datos:** Antes de realizar cualquier actualización, es fundamental hacer una copia de seguridad de la base de datos. Esto evita pérdida de información ante posibles fallos.
  + **Instrucciones en XAMPP:**
    - Abre phpMyAdmin desde el panel de control de XAMPP.
    - Selecciona la base de datos llamada proyecto.
    - Haz clic en la pestaña Exportar.
    - Elige el método Rápido y formato SQL.
    - Haz clic en Continuar para descargar el archivo .sql.
* **Paso 3: Instalar Dependencias y Ejecutar Cambios:** Si el proyecto ha recibido actualizaciones relacionadas con librerías externas (como Twilio o futuras dependencias), asegúrate de ejecutar el siguiente comando para instalar o actualizar: “composer update”

**9. Desinstalación**

* **Paso 1: Detener el servidor o servicio en ejecución:** Dependiendo de cómo se haya iniciado el sistema, existen dos formas principales de detener el servidor:
  + **Si usaste la terminal con PHP embebido:** Si ejecutaste el servidor local con el siguiente comando: “php -S localhost:8000 -t to\_aplicacion”
    - Dirígete a la ventana de la terminal que está ejecutando el servidor.
    - Presiona Ctrl + C para interrumpir el proceso y cerrar el servidor web local.
    - Verás un mensaje indicando que el proceso ha sido terminado exitosamente.
  + **Si usaste XAMPP:**
    - Abre el panel de control de XAMPP.
    - Detén los módulos **Apache** y **MySQL** pulsando los botones "Stop".
    - Asegúrate de que ambos servicios se marquen como "Detenido".
* **Paso 2: Eliminar todos los archivos y carpetas relacionados:** Una vez que el servidor ha sido detenido, procede a eliminar los archivos del proyecto y sus dependencias.
  + **Eliminar el directorio del proyecto:**
    - **Ubicación típica**: El proyecto puede encontrarse en una ruta personalizada, por ejemplo:  
      C:\Users\TuUsuario\Documents\Proyecto\_aplicacion en Windows  
      o /home/usuario/Documentos/Proyecto\_aplicacion en Linux/macOS.
    - **Acción**:
      * En Windows: haz clic derecho sobre la carpeta Proyecto\_aplicacion y selecciona Eliminar o presiona Shift + Supr para borrado definitivo.
      * En Linux/macOS, puedes usar el siguiente comando en la terminal: “rm -rf /ruta/del/proyecto/Proyecto\_aplicacion”
  + **Eliminar dependencias de Composer:** Si utilizaste Composer para instalar el SDK de Twilio, dentro del proyecto se generó una carpeta llamada vendor/ y un archivo composer.lock. Asegúrate de eliminar también:
    - vendor/
    - composer.lock
    - composer.json (si ya no planeas reutilizarlo)

Si tienes Composer instalado de forma global, no es necesario desinstalarlo ya que puede servir para otros proyectos.

* **Paso 3: Eliminar la base de datos del sistema:** Eliminar la base de datos es esencial si deseas liberar espacio o evitar datos obsoletos.
  + **Usando phpMyAdmin (XAMPP)**
    - Abre el navegador y entra a: <http://localhost/phpmyadmin>
    - En la columna izquierda, localiza la base de datos llamada proyecto.
    - Haz clic sobre ella y selecciona la pestaña Operaciones.
    - Al final de la página, verás la opción Eliminar la base de datos (DROP).
    - Confirma la eliminación. Esto eliminará permanentemente todas las tablas y registros.
  + **Usando la línea de comandos de MySQL:**
    - Abre la terminal o consola.
    - Conéctate al cliente MySQL: “mysql -u root -p”
    - Ingresa tu contraseña (si no tienes, solo presiona Enter).
    - Ejecuta el siguiente comando para eliminar la base de datos: “DROP DATABASE proyecto;”
    - Escribe exit para salir del cliente de MySQL.

**10. Solución de Problemas (Troubleshooting)**

* **Problema 1: El navegador muestra "No se puede acceder a este sitio" al ingresar** [**http://localhost**](http://localhost)**:**
  + **Posible causa:**
    - El servidor Apache de XAMPP no está iniciado.
    - El proyecto no está en la carpeta correcta (htdocs de XAMPP).
    - Puerto incorrecto.
  + **Solución:**
    - Abrir el Panel de Control de XAMPP.
    - Asegurarse de que el módulo Apache esté en estado "Running".
    - Verificar que el proyecto Proyecto\_aplicacion esté dentro de la ruta: “C:\xampp\htdocs\”
    - Acceder desde el navegador “<http://localhost/Proyecto_aplicacion>”
    - Si aún falla, ingresar a: “<http://localhost:8080/Proyecto_aplicacion>” (solo si cambiaste el puerto en el archivo httpd.conf de Apache).
* **Problema 2: Error de conexión a la base de datos ("Access denied" o "No se pudo conectar al servidor MySQL"):**
  + **Posible causa:**
    - El archivo conexion.php no tiene las credenciales correctas.
    - MySQL no está corriendo.
    - La base de datos proyecto no fue importada.
  + **Solución:**
    - Abrir el archivo: “Proyecto\_aplicacion/config/conexion.php”
    - Verificar que los valores coincidan con los de tu entorno local (XAMPP por defecto):

$host = 'localhost';

$dbname = 'proyecto';

$username = 'root';

$password = '';

* + - Confirmar que MySQL está activo desde el panel de XAMPP.
    - Entrar a phpMyAdmin y asegurarse de haber importado correctamente el archivo: “Proyecto\_aplicacion/database/proyecto\_.sql”
* **Problema 3: Al registrar un usuario, no llega el mensaje de aprobación por SMS:**
  + **Posible causa:**
    - El SDK de Twilio no está bien instalado o configurado.
    - Las credenciales de Twilio en el archivo .env o de configuración están incompletas o incorrectas.
    - El número ingresado en el formulario no está en formato internacional.
  + **Solución:**
    - Asegúrate de haber ejecutado en la raíz del proyecto: “composer install”
    - Verifica que el archivo donde se inicializa Twilio contenga correctamente:

$sid = 'TU\_SID';

$token = 'TU\_AUTH\_TOKEN';

$twilio = new Client($sid, $token);

* + - Confirma que el número de teléfono al que se envía el SMS tenga el prefijo internacional, por ejemplo: +573001234567.
    - Si estás usando la versión gratuita de Twilio, asegúrate de haber verificado ese número desde la plataforma.
* **Problema 4: La aplicación no guarda los datos después de registrar un nuevo usuario:**
  + **Posible causa:**
    - Permisos de escritura restringidos en archivos o base de datos.
    - Error en la consulta SQL de inserción.
    - Campos obligatorios no completados correctamente.
  + **Solución:**
    - Revisar el código PHP del formulario para ver si hay errores de validación.
    - Activar display\_errors en PHP para ver los errores exactos:
    - En php.ini, asegurarse de tener: “display\_errors = On”
    - Verificar en la consola de navegador o en las herramientas de desarrollador que no haya errores de red (HTTP 500 o 404).
    - Confirmar desde phpMyAdmin que los registros realmente no se están guardando (revisar la tabla usuarios u otras relacionadas).
* **Problema 5: Composer no reconoce los comandos o no se instala correctamente:**
  + **Posible causa:**
    - Composer no está agregado al PATH del sistema.
    - Composer no fue instalado correctamente.
  + **Solución:**
    - Para comprobar si Composer está instalado: “composer --version”
    - Si no lo reconoce, agregar Composer manualmente al PATH (en Windows).
    - Reinstalar Composer desde https://getcomposer.org/ asegurándote de seleccionar la opción para incluirlo en la variable de entorno.

**11. Soporte y Contacto**

* **Correo de soporte**: Puedes ponerte en contacto directo con los creadores del proyecto a través de sus correos institucionales:
  + Jesús Méndez: jesus\_mendezme@fet.edu.co
  + Andrés Puentes: andres\_puentesri@fet.edu.co
  + Derek Quevedo: [derek\_quevedosa@fet.edu.co](mailto:derek_quevedosa@fet.edu.co)

Estos correos están habilitados para responder inquietudes sobre la instalación, configuración, uso y mantenimiento de la plataforma.

* **Documentación oficial**: Este manual técnico cumple la función de ser la documentación oficial de MissionFet, y está diseñado para guiar a los usuarios desde la descarga hasta la ejecución, configuración avanzada y solución de problemas. Se recomienda conservar una copia digital o impresa para futuras consultas.
* **Foros de soporte**: En caso de requerir soporte adicional sobre tecnologías utilizadas en el desarrollo de MissionFet, puedes acceder a las siguientes comunidades en línea:
  + **PHP:**
    - https://www.php.net/support.php
    - https://stackoverflow.com/questions/tagged/php
  + **JavaScript:**
    - https://developer.mozilla.org/es/docs/Web/JavaScript
    - https://stackoverflow.com/questions/tagged/javascript
  + **XAMPP (Apache, MySQL, PHP, Perl):**
    - https://community.apachefriends.org/f/
    - https://stackoverflow.com/questions/tagged/xampp
  + **Twilio (para integración de APIs):**
    - https://www.twilio.com/help
    - https://stackoverflow.com/questions/tagged/twilio

Estas plataformas ofrecen soluciones a problemas frecuentes, guías, documentación oficial y foros donde puedes hacer preguntas específicas y recibir respuestas de desarrolladores y expertos de todo el mundo.

**12. Glosario**

* **XAMPP:** Paquete que incluye Apache, MySQL, PHP y Perl, y permite montar un servidor local en tu computadora para probar aplicaciones web.
* **Apache:** Servidor web que interpreta y ejecuta archivos PHP, permitiendo que el sitio funcione correctamente en un entorno local.
* **MySQL:** Sistema de gestión de bases de datos utilizado para almacenar y organizar toda la información de la plataforma.
* **PHP:** Lenguaje de programación del lado del servidor usado en este proyecto para la lógica de la aplicación y la conexión con la base de datos.
* **HTML:** Lenguaje de marcado utilizado para estructurar el contenido de las páginas web.
* **CSS:** Lenguaje usado para dar estilo y diseño visual a los elementos HTML de la página.
* **JavaScript:** Lenguaje de programación del lado del cliente usado para interactividad y validaciones en la aplicación.
* **Twilio:** Plataforma que permite enviar mensajes SMS, utilizada en este proyecto para notificar al usuario cuando su cuenta ha sido aprobada.
* **API:** Conjunto de funciones y procedimientos que permiten la comunicación entre diferentes aplicaciones. En este caso, se usa para conectar con Twilio.
* **SDK:** Kit de desarrollo de software que facilita el uso de herramientas o plataformas externas, como el SDK de Twilio instalado con Composer.
* **Composer:** Gestor de dependencias para PHP que permite instalar bibliotecas o herramientas como el SDK de Twilio.
* **Repositorio:** Espacio en GitHub donde se almacena y gestiona el código fuente del proyecto.
* **Git:** Sistema de control de versiones que permite registrar cambios en el código y colaborar con otros desarrolladores.
* **GitHub:** Plataforma en línea para alojar repositorios Git, compartir proyectos y colaborar con otros usuarios.
* **Rama (branch):** Versión paralela del proyecto donde se pueden hacer modificaciones sin afectar la rama principal (master).
* **Base de datos:** Conjunto organizado de información que se almacena y consulta desde el servidor. En este proyecto, la base de datos se llama proyecto.
* **SQL:** Lenguaje para consultar y manipular datos en una base de datos. Se usa para crear, insertar, eliminar o consultar información.
* **Servidor local:** Entorno simulado en tu computador que permite correr aplicaciones web sin necesidad de estar en línea.
* **Terminal / Consola:** Interfaz donde se escriben comandos para ejecutar tareas como iniciar el servidor o instalar dependencias.
* **Roles de usuario:** Niveles de acceso definidos en el sistema (administrador, tutor y estudiante), cada uno con permisos específicos.

**13. Licencia y Créditos**

* **Licencia del software**: Este proyecto actualmente no cuenta con una licencia de software definida. Se recomienda a los desarrolladores establecer una licencia adecuada (por ejemplo, MIT, GPLv3 o Apache 2.0) si se planea distribuir, modificar o reutilizar este software públicamente. En ausencia de una licencia, el código se considera por defecto como "Todos los derechos reservados", lo que limita su uso por terceros.
* **Créditos**: Este software fue desarrollado como parte de un proyecto académico por los siguientes autores:
  + **Jesús Méndez**  
    Correo institucional: *Jesus\_mendezme@fet.edu.co*
  + **Andrés Puentes**  
    Correo institucional: *andres\_puentesri@fet.edu.co*
  + **Derek Quevedo**  
    Correo institucional: *Derek\_quevedosa@fet.edu.co*
* **Tutor del proyecto:**
  + **Carlos Alberto Lugo Puentes**

Rol: Docente y guía académico del desarrollo del software.

Agradecemos el apoyo brindado por la institución educativa y por el docente tutor que orientó el proceso de planificación, desarrollo y documentación del sistema.