

UNIRVERSIDAD AUTONOMA JUAN MISAEL SARACHO

FACULTAD DE CIENCIAS Y TECNOLOGIA

CARRERA INGENIERIA INFORMATICA



APLICACIÓN MOVIL PARA PROCESAR PEDIDOS DE UNA VETERINARIA

MATERIA: TECNOLOGIAS EMERGENTES

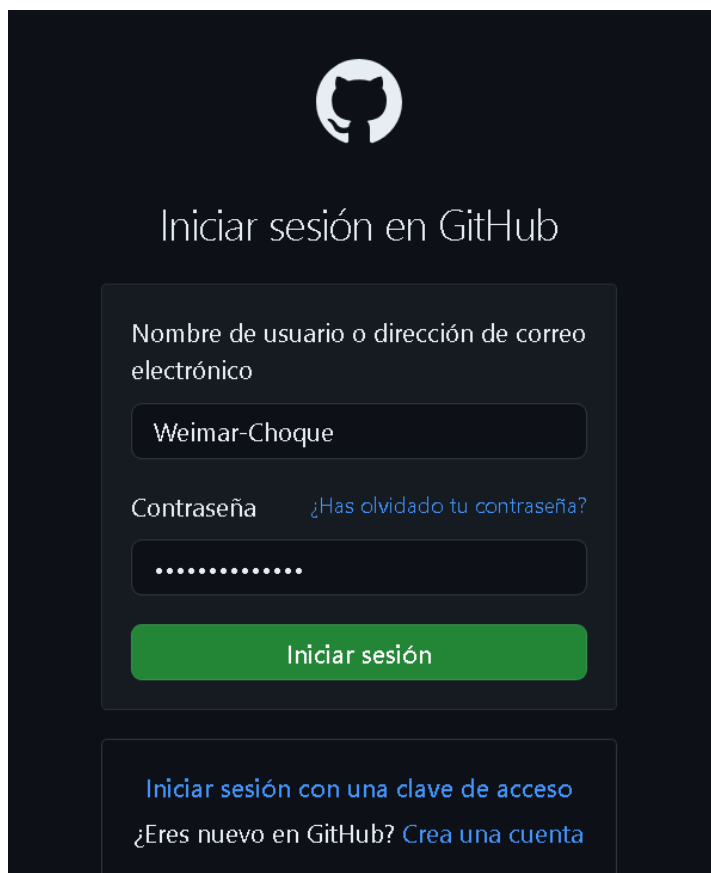
DOCENTE COPA FERNANDEZ TEOFILO

Tarija - Bolivia

Para crear un repositorio en GitHub que contenga el código fuente completo de tu aplicación Android, Firebase, y los esquemas de bases de datos, sigue estos pasos generales:

1. Crear un Repositorio en GitHub

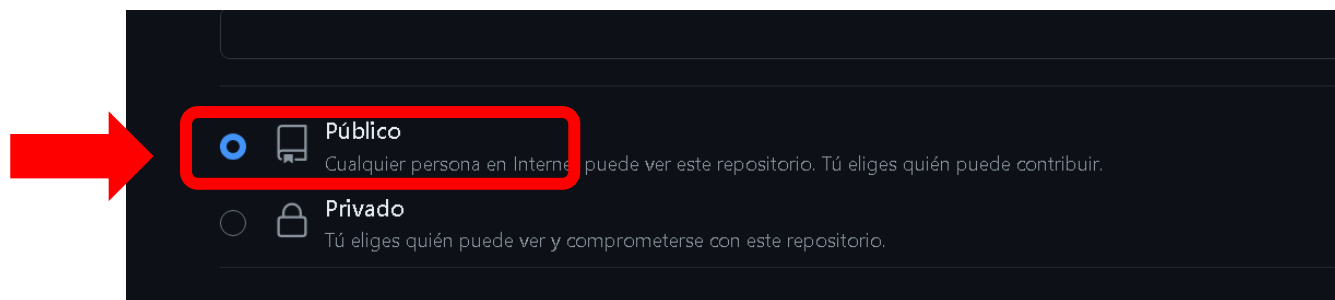
1. **Inicia sesión** en tu cuenta de GitHub.



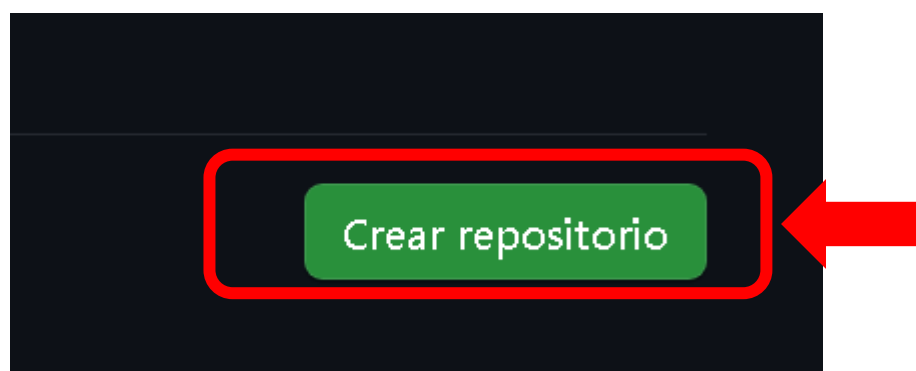
2. En la esquina superior derecha, haz clic en + y selecciona **Nuevo repositorio**.



3. Completa el nombre del repositorio, la descripción (opcional) y elige si será público o privado "En nuestro caso lo definimos **Público**".

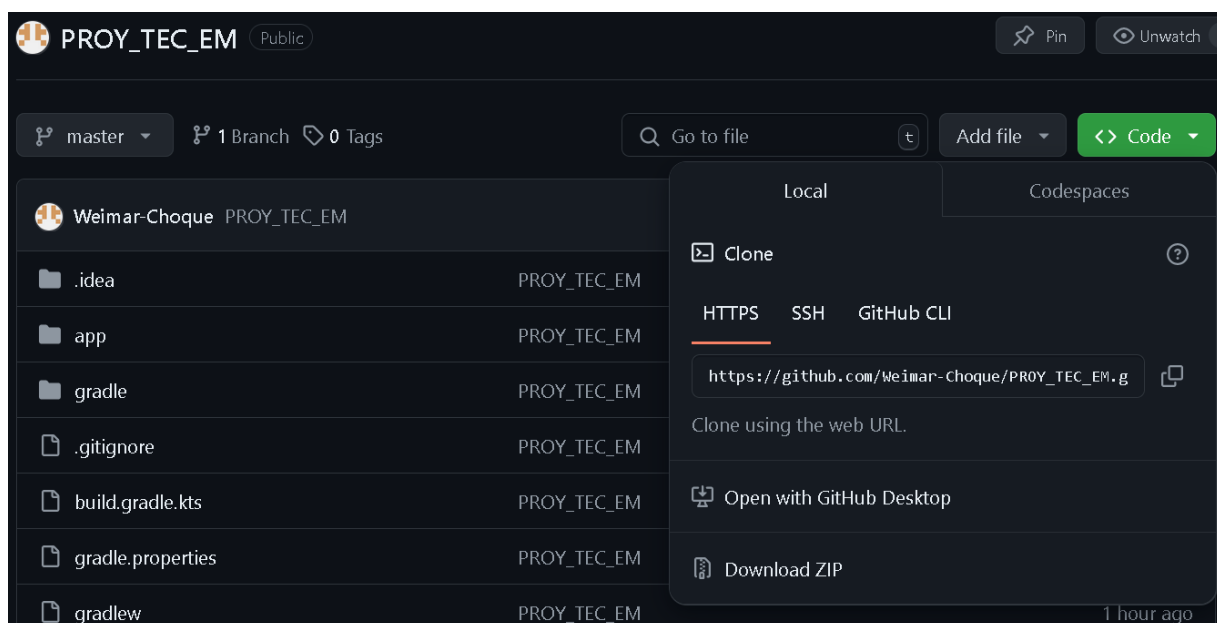


4. Haz clic en **Crear repositorio**.



2. Configurar el Repositorio Local

1. **Clona** el repositorio recién creado en tu máquina local usando Git Bash o la terminal de tu sistema operativo:



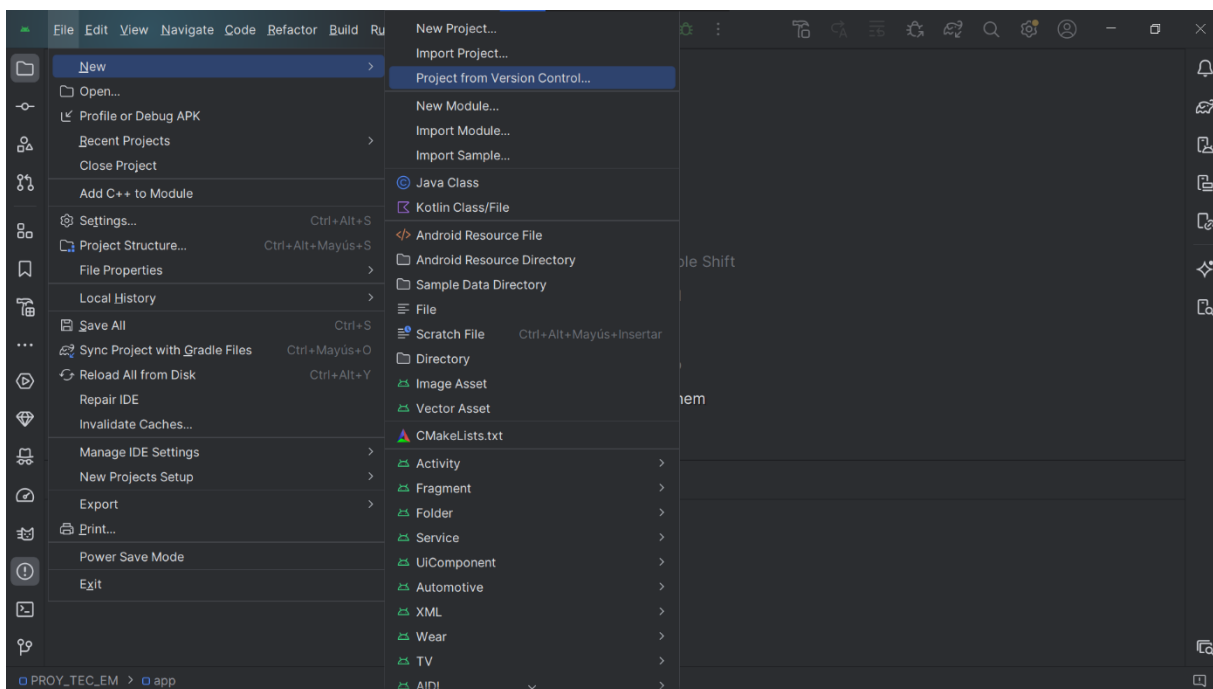
git clone https://github.com/tu_usuario/tu_repositorio.git

- Reemplaza `tu_usuario` y `tu_repositorio` con tu nombre de usuario y el nombre del repositorio en GitHub.

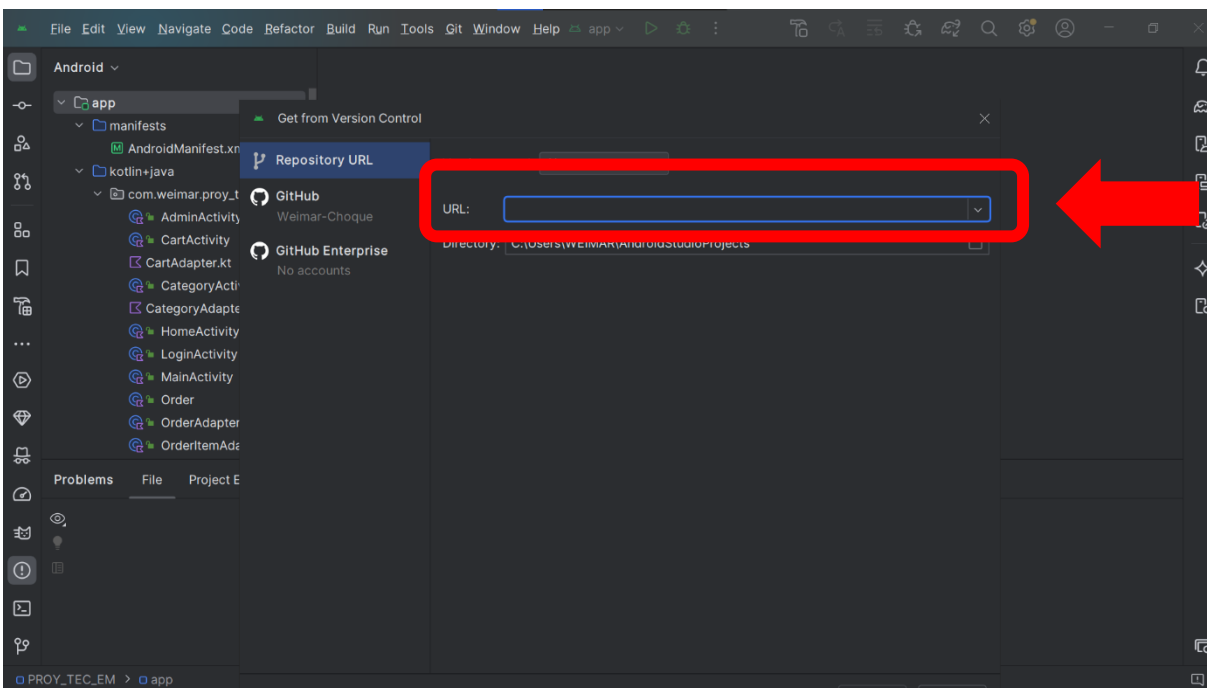
2. Configura el proyecto **localmente**:

- Coloca el código fuente de la aplicación Android en el directorio adecuado.

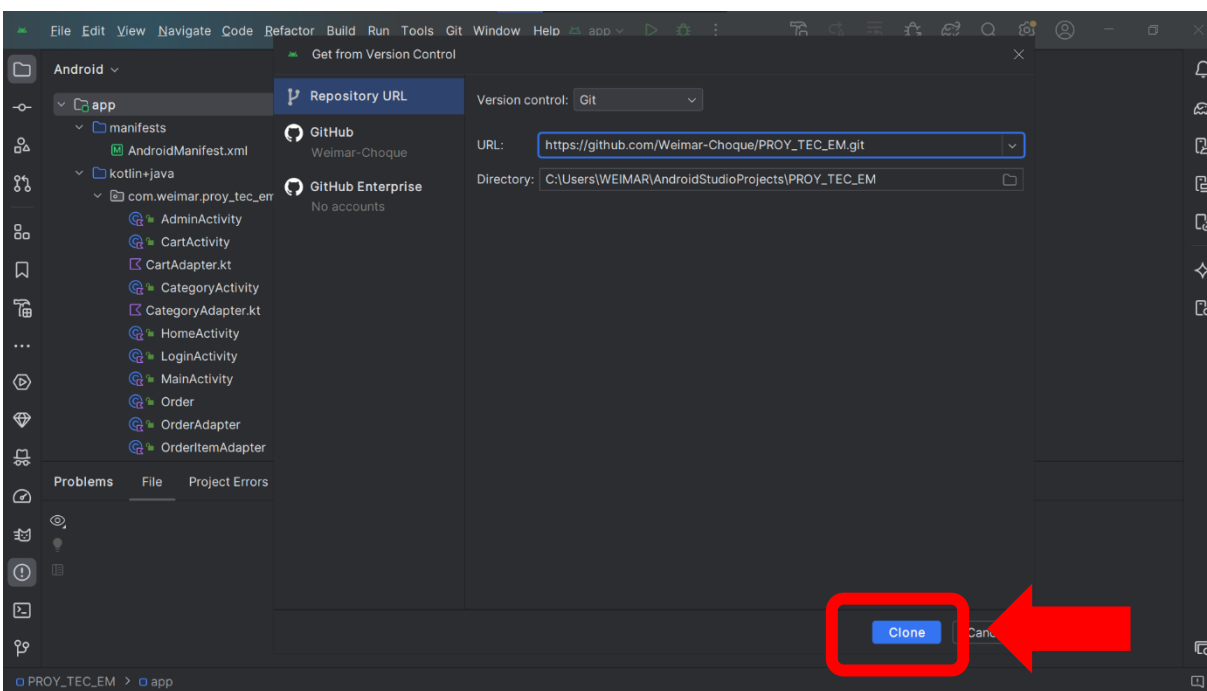
En Android Studio ve a **File > New > Project from Version Control**



Se abrirá la siguiente ventana en la cual debes de ingresar el enlace generado por GitHub para clonar el repositorio.

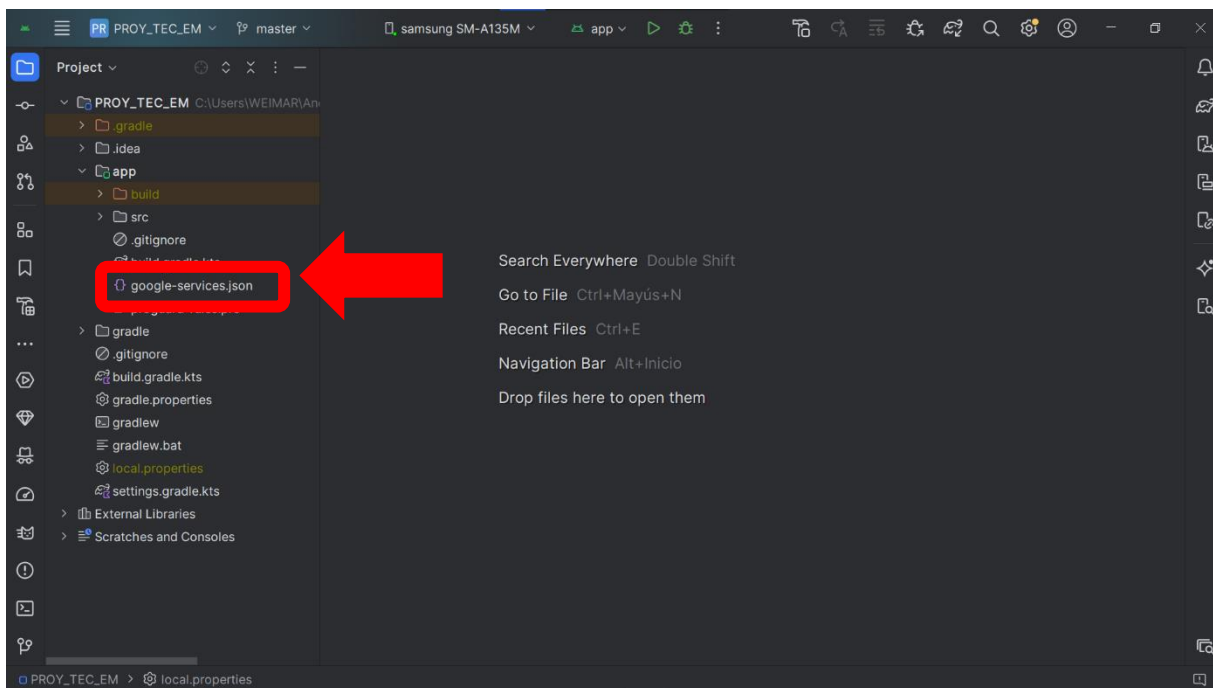


Ingresas el enlace y presionas Clone



Finalmente conecta un dispositivo Android o utiliza el emulador integrado en Android Studio para compilar y ejecutar la aplicación Android.

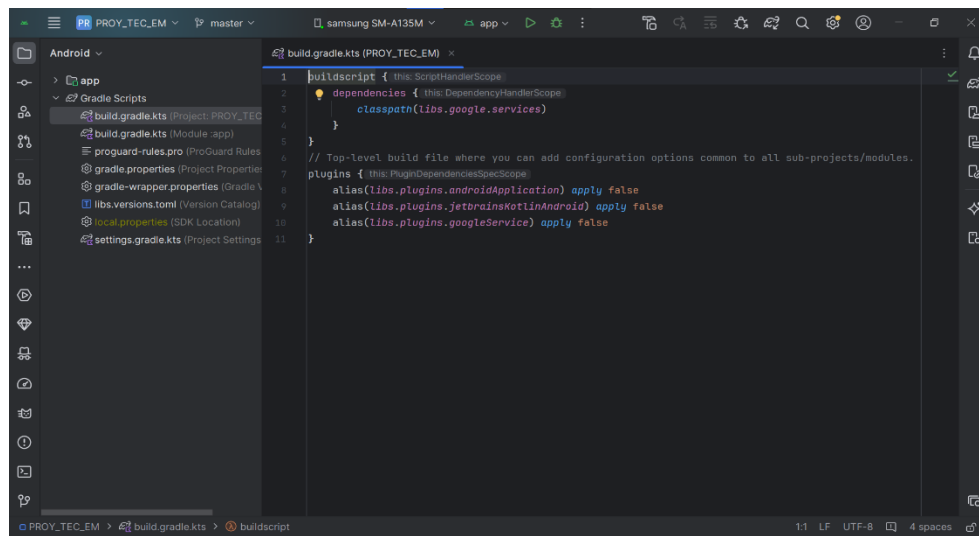
Asegúrate de que en el proyecto clonado se encuentre el archivo **google-services.json** que es para agregar Firebase al proyecto de Android.



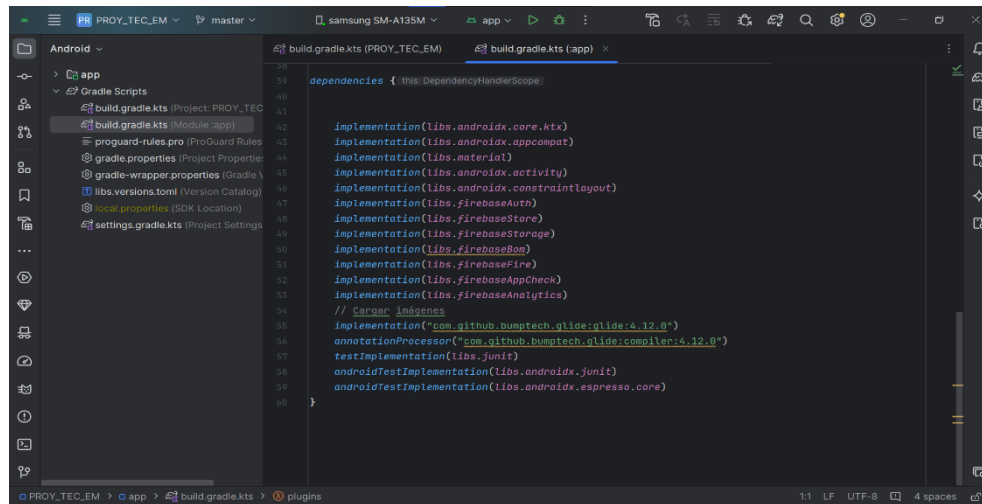
En tu archivo `build.gradle` a nivel de módulo (normalmente `app/build.gradle`). Aquí están las dependencias necesarias para Firebase Authentication, Firestore, y Glide para el manejo de imágenes.

- ✓ **Firebase BoM (Bill of Materials):** Firebase BoM te permite gestionar todas las versiones de tus dependencias de Firebase de una manera centralizada y asegurarte de que todas las bibliotecas de Firebase sean compatibles entre sí.
- ✓ **Firebase Authentication:** Para autenticar a los usuarios.
- ✓ **Firebase Firestore:** Para manejar la base de datos NoSQL.
- ✓ **Glide:** Para cargar y mostrar imágenes de manera eficiente.

A nivel de Proyecto



A nivel de Modulo



Esquemas de Bases de Datos

Firestore

Colección: users

Campo	Tipo	Descripción
uid	String	ID único del usuario (auto-generado por Firebase Auth)
name	String	Nombre del usuario
email	String	Correo electrónico del usuario
role	String	Rol del usuario (e.g., admin, user)

Colección: carts

Cada usuario tiene su propia subcolección `cart` dentro de su documento en la colección `users`.

Subcolección: `cart` (dentro de `users/{userId}`)

Campo	Tipo	Descripción
<code>productId</code>	String	ID único del producto
<code>name</code>	String	Nombre del producto
<code>price</code>	Number	Precio del producto
<code>imageUrl</code>	String	URL de la imagen del producto
<code>quantity</code>	Number	Cantidad del producto en el carrito

Colección: products

Campo	Tipo	Descripción
<code>id</code>	String	ID único del producto
<code>name</code>	String	Nombre del producto
<code>description</code>	String	Descripción del producto
<code>price</code>	Number	Precio del producto
<code>imageUrl</code>	String	URL de la imagen del producto

Colección: orders

Campo	Tipo	Descripción
<code>orderId</code>	String	ID único de la orden (auto-generado)
<code>userId</code>	String	ID del usuario que realizó la orden
<code>products</code>	Array	Lista de productos en la orden
<code>totalPrice</code>	Number	Precio total de la orden
<code>createdAt</code>	Timestamp	Fecha de creación del documento