

Asignatura	Datos del alumno	Fecha
Ingeniería del <i>software</i> avanzada	Apellidos: Jiménez Acosta	
	Nombre: Ronaldo	

Trabajo: Diseño de una aplicación con patrones

1. Introducción

La aplicación que estamos diseñando es un cliente de Twitter, una plataforma social que permite a los usuarios compartir mensajes cortos conocidos como "tweets". El propósito de este diseño es aplicar patrones de arquitectura y diseño de software para estructurar la funcionalidad de la aplicación, enfocándonos en las siguientes características principales:

- **Registro de usuario e inicio de sesión:** permitir a los usuarios crear una cuenta y acceder a la plataforma.
- **Publicación de tweets y gestión de interacciones sociales:** los usuarios podrán escribir, publicar, retuitear y dar "me gusta" a los tweets.
- **Línea de tiempo y búsqueda de usuarios:** los usuarios podrán ver una línea de tiempo personalizada con tweets de las personas que siguen y buscar otros usuarios por nombre o palabra clave.

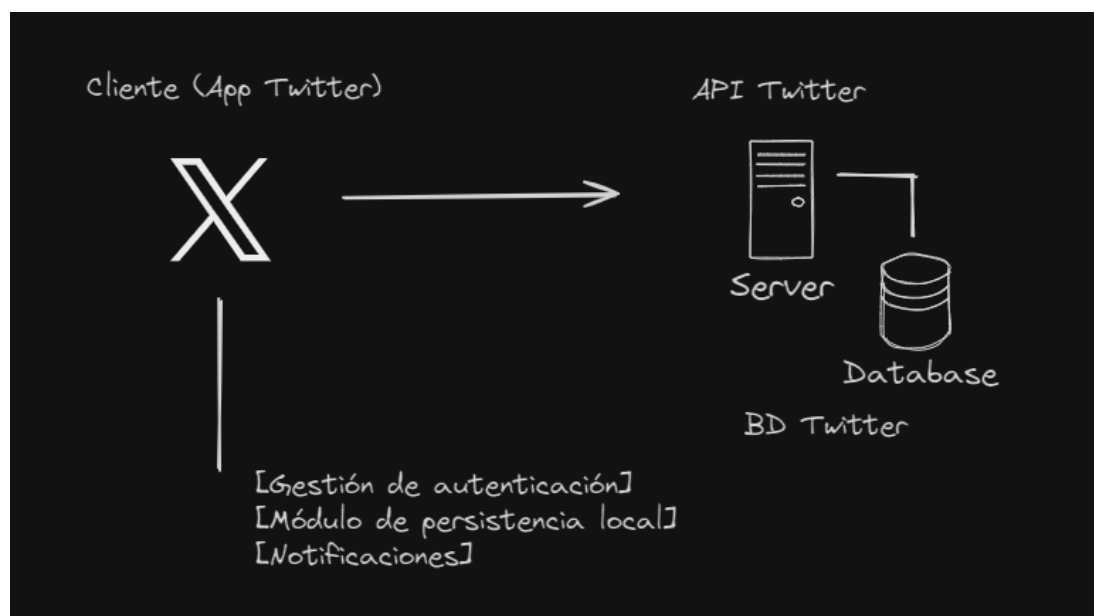
Este documento se centrará en el diseño de estos aspectos clave, aplicando patrones de diseño para resolver problemas específicos en cada una de las funcionalidades mencionadas.

2. Diseño de la arquitectura del sistema

La aplicación de Twitter debe seguir una arquitectura cliente-servidor, donde el cliente (la aplicación que estamos diseñando) interactúa con la API de Twitter para acceder y gestionar datos como tweets, usuarios y notificaciones.

Componentes principales:

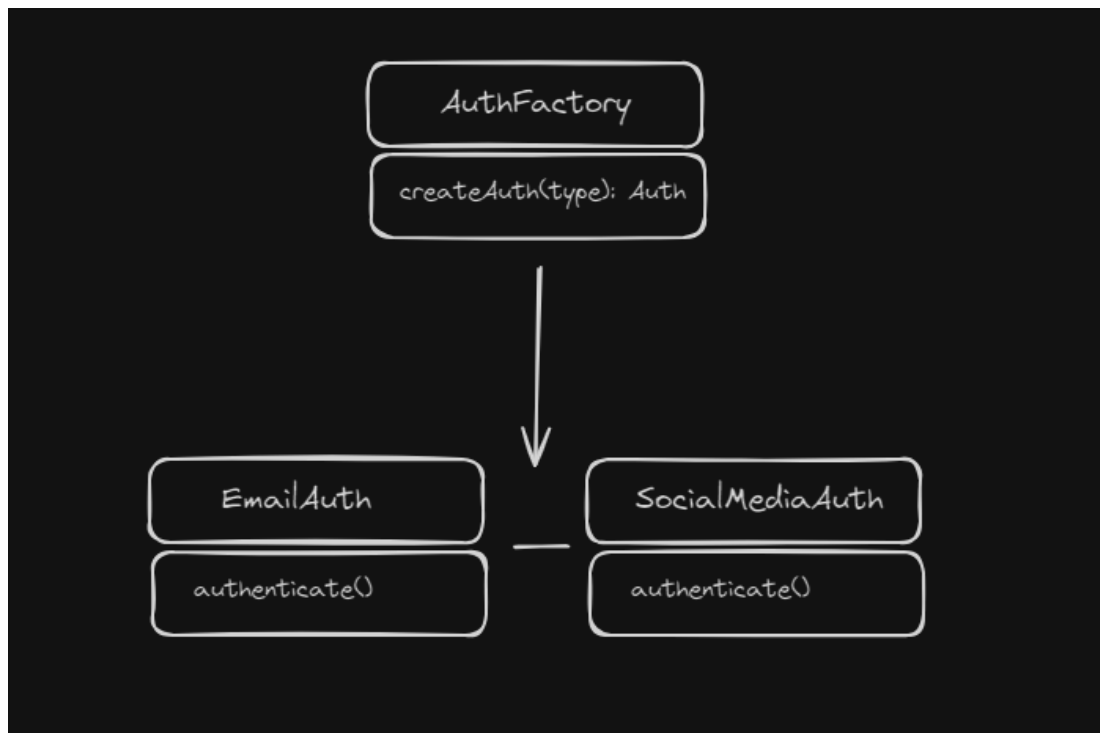
- **Cliente (App Twitter):** Este componente incluye la interfaz gráfica (UI) y la lógica de negocio. Los usuarios interactúan con este cliente para realizar acciones como publicar tweets, ver la línea de tiempo, seguir usuarios, etc.
- **API de Twitter:** Este sistema externo permite a la aplicación interactuar con los datos en Twitter. A través de peticiones REST, el cliente puede obtener información, publicar tweets y gestionar las interacciones sociales.
- **Gestión de autenticación:** Un componente dentro del cliente que se conecta con la API para autenticar usuarios, manejar sesiones, y gestionar los tokens de seguridad necesarios para la interacción con Twitter.
- **Módulo de persistencia local:** Un componente que guarda datos temporalmente en el dispositivo del usuario para mejorar la experiencia offline, como guardar la línea de tiempo o el historial de búsqueda.
- **Notificaciones:** Un sistema que se conecta al servicio de notificaciones de Twitter para recibir alertas sobre menciones, retweets, y mensajes directos.



3. Diseño de componentes y clases (diagrama)

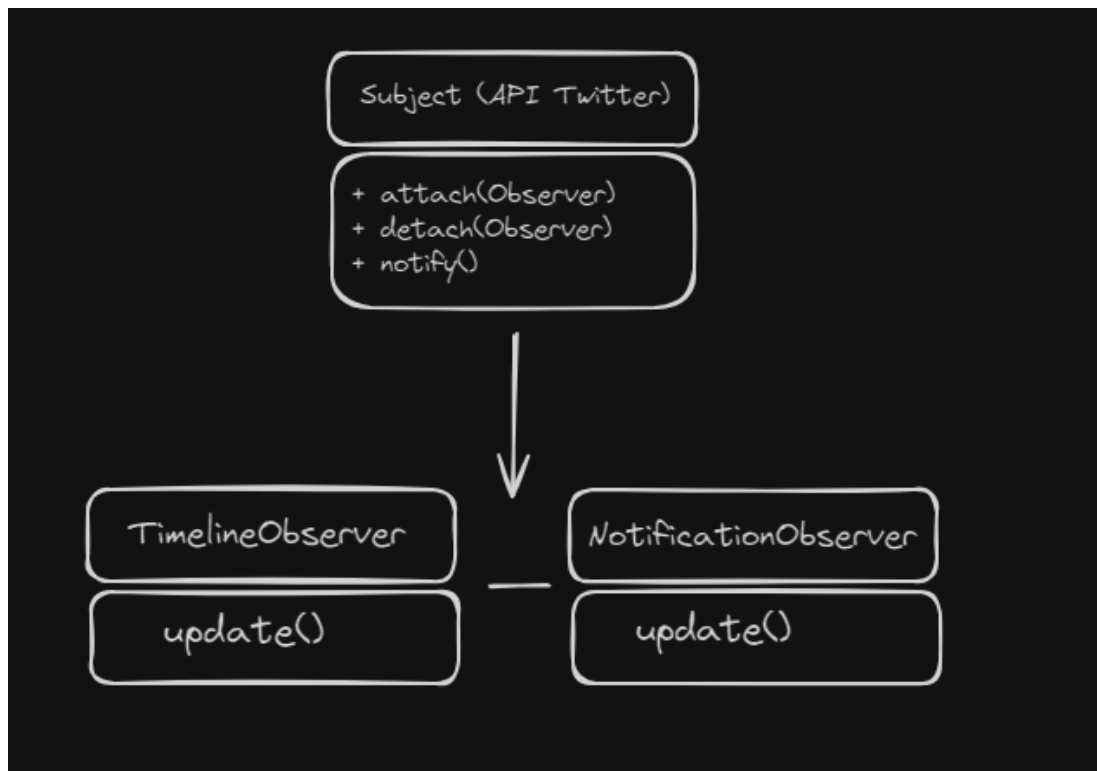
Patrón 1: Factory Method para el registro e inicio de sesión

- **Problema:** Necesitamos instanciar diferentes tipos de autenticaciones (registro con email, inicio de sesión con email o redes sociales).
- **Funcionalidad asociada:** Registro de usuario e Inicio de sesión
- **Solución:** Aplicar el patrón Factory Method para crear diferentes métodos de autenticación. El Factory Method permite instanciar la clase de autenticación requerida en tiempo de ejecución.



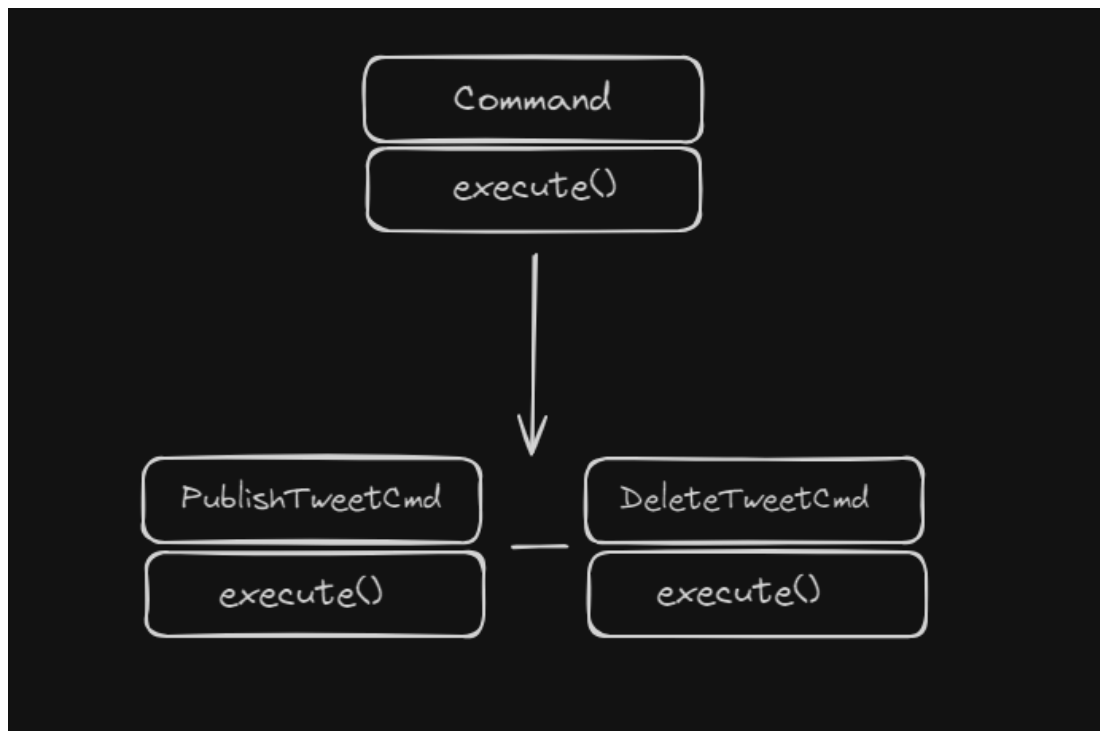
Patrón 2: Observer para la línea de tiempo y notificaciones

- **Problema:** La línea de tiempo debe actualizarse automáticamente cuando los usuarios a los que seguimos publican nuevos tweets, o cuando se recibe una nueva notificación.
- **Funcionalidad asociada:** Línea de tiempo y Notificaciones .
- **Solución:** El patrón Observer permite que los usuarios "observen" a otros usuarios y a la API de Twitter para recibir actualizaciones en tiempo real.



Patrón 3: Command para la publicación de tweets

- **Problema:** Se necesita ejecutar comandos como publicar, editar o eliminar tweets de manera que se puedan manejar de forma uniforme y, eventualmente, deshacer operaciones.
- **Funcionalidad asociada:** Publicación de tweets .
- **Solución:** El patrón Command encapsula las operaciones como objetos, permitiendo ejecutar o deshacer comandos sin necesidad de saber los detalles de la operación.



4. Diseño de la interfaz de usuario

Patrón 1: MVC (Modelo-Vista-Controlador)

- **Problema:** Separar la lógica de negocio (publicación de tweets, manejo de la línea de tiempo) de la interfaz gráfica.
- **Solución:** El patrón MVC divide la aplicación en tres componentes: Modelo (datos), Vista (interfaz), y Controlador (manejo de eventos de usuario). Esto permite mantener la UI desacoplada de la lógica.

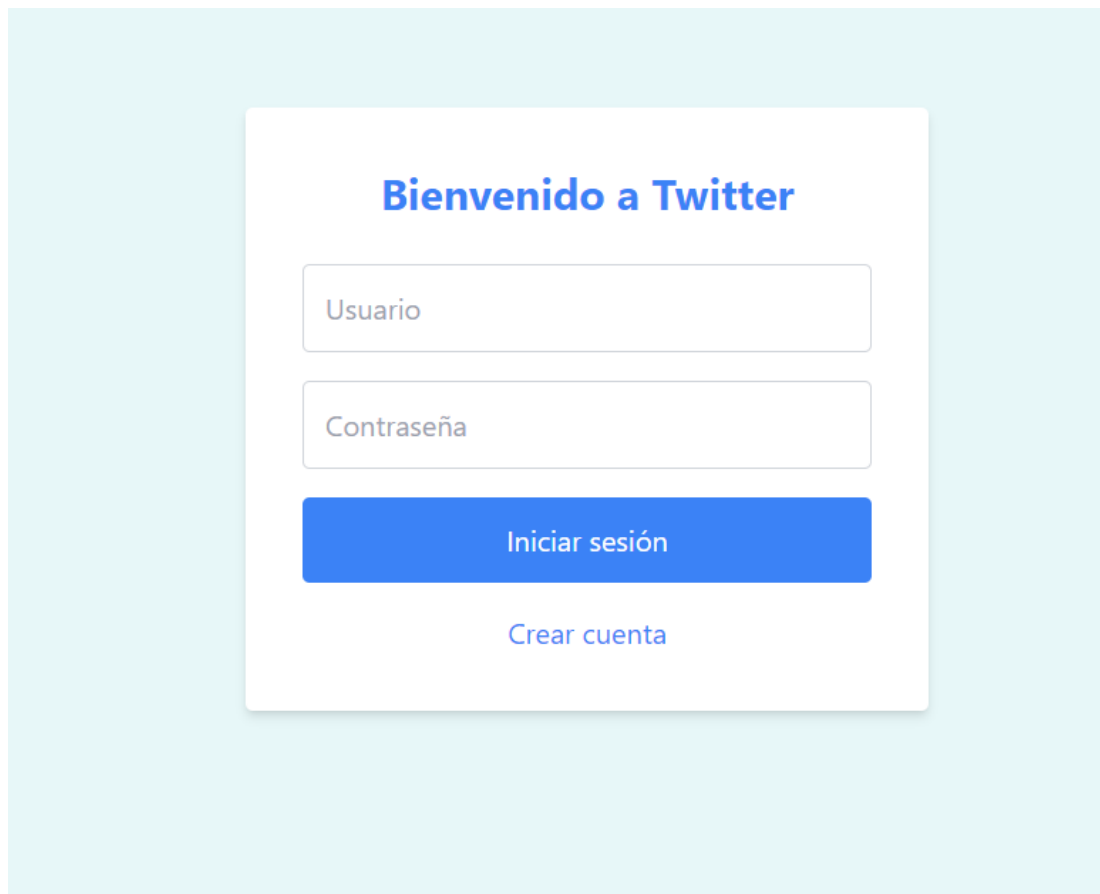
Patrón 2: Observer en la UI

- **Problema:** La UI debe actualizarse automáticamente cuando cambian los datos (nuevos tweets o notificaciones).
- **Solución:** Aplicar Observer para que la Vista observe el Modelo, permitiendo actualizaciones en tiempo real de la línea de tiempo.

Patrón 3: Singleton para la gestión de sesión del usuario

- **Problema:** La aplicación debe garantizar que solo exista una instancia de la sesión de usuario durante la ejecución de la app.

- **Solución:** Utilizar el patrón Singleton para la clase que maneja la sesión, asegurando que exista una única instancia compartida en toda la aplicación.



A login screen for Twitter. It features a light blue background. In the center is a white card with a blue header that says "Bienvenido a Twitter". Below the header are two input fields: "Usuario" and "Contraseña". Below these fields is a blue button labeled "Iniciar sesión". At the bottom of the card is a blue link that says "Crear cuenta".



A Twitter feed interface. At the top left is the "Twitter" logo, and at the top right is a blue button labeled "Perfil". Below the header is a list of tweets. The first tweet is from "Usuario1" (@usuario1) and says "Este es un tweet de ejemplo." with options "Me gusta", "Retweet", and "Responder". The second tweet is from "Usuario2" (@usuario2) and says "Aquí va otro tweet de ejemplo." with the same options.

Nuevo Tweet

Publicar

¿Qué estás pensando?

Añadir Imagen Añadir Video

Nota: Para esta actividad, he creado un diseño utilizando **Astro y TailwindCSS**. Opté por estas herramientas porque me facilitan el proceso de creación de interfaces de usuario, permitiéndome desarrollar de manera rápida y eficiente. Puedes ver el diseño en el siguiente enlace:

<https://astro-twitter.vercel.app/>