Proyecto computación móvil:

Mockups con funcionalidad: https://www.figma.com/design/oWMqA6Ojw6LbJP9QdxDNmP/Mockups-pagina?node-id=0-1&t=XkGS93EDoORMEDJn-1

Aplicación por hacer: Red Social de anime

Se trabajara con Supabase.

Ideas claves: X (Twitter), Instagram, Tik Tok, Facebook.

Temática usada: Anime

Requerimientos obligatorios: JwT (Json web tocken) Código no superior a 150 líneas

Aplicaciones para desarrollar: Android Studio y visual studio

Metodología: PWA arquitectura de aplicaciones web progresiva

Alcance: Global

Objetivos: Conectar personas con gustos similares al anime con contenido de otras personas, conocer gente y hacer conexiones mediante gustos, publicaciones y contenido relacionado al anime.

Crear una app que permita filtrar las conexiones para lograr hacer una experiencia grata.

Ventanas para realizar: Login, página de contenido, página de perfil, conexiones, chats, modificador de publicaciones.

Implementación de IA para creación de imagenes y filtraciones.

Lenguajes mínimos a usar: Html, Css, Javascript, Java, kotlin.

Requerimientos funcionales:

Login: El login tendrá 3 botones 2 los cuales son importantes uno de registro para que un usuario nuevo pueda registrarse y crear una cuenta en la aplicación modificando desde la app la base de datos, el botón de ingreso para usuarios con cuentas ya creadas que usan la base de datos para consultar usuarios existentes.

El botón de login hace un despliegue a una vista dentro del Javascript que contendrá un input de correo y un input de contraseña, con un botón el cual hace una consulta a la base de datos para traer el usuario por validaciones de contraseña y correo (ambos deben de estar llenos o con algo ingresado), al haber una confirmación alojará a la ventana principal, un botón de volver que lo que hace es devolver a la vista principal.

El botón de registro envía a una vista externa la cual tiene 3 inputs uno para nombre de usuario (el cual no puede estar vacío o “null”), uno para correo (que evalué el formato de correo valido y con un correo existente como confirmación) y uno para contraseña, un botón de registrar el cual hace un registro (modificación a la base de datos) y un botón de volver el cual retorna a la vista principal

El botón de noticias de anime abre una vista externa la cuál tiene una card central con una conexión a Api de animes que cargas fotos de anime y sus nombres, al hundir a esa información despliega una vista pequeña con una noticia relacionada, también tiene un botón de volver que retorna a la vista principal

Pestaña inicial: La pestaña inicial tiene un panel ocultable que contiene para las pestañas de perfil, publicación, búsqueda, chats e inicio, tiene un botón para reacciones, un texto para comentarios acerca de publicaciones, al lado derecho una barra pegada que contiene perfiles sugeridos y sus historias.

Tiene un menú ocultable en la zona superior el cual al darle clic revelará o esconderá el menú, al tendrá un menú flotante ubicado verticalmente en la zona derecha al darle clic a cada perfil que no tenga historia te llevará a la pestaña del perfil de ese usuario o si tiene una historia mostrará la historia posteada, se tendrá un tipo de cards ubicadas centralmente con la foto o post, una zona del perfil o usuario que al darle clic a su icono te llevara a su perfil y una zona de likes que aumenta o decrece según su interacción y para hacer comentarios o ver comentarios la cual despliega una vista externa para comentar con un input de texto y un botón para postear el cual hace una modificación a una base de datos con los comentarios o para ver comentarios muestra una vista que carga de la base de datos comentarios que se han hecho de esa publicación.

(Requerimientos y validaciones ya hechas)

Chats: La pestaña de chats cuenta con el panel ocultable, un selector tipo deslizador de selección para chats comunes o destacados, la burbuja de chats con el icono de perfil y el mensaje abajo tendrá los perfiles seguidos o destacados con animes recomendados u historias.

La pestaña de chat tiene el menú ocultable, seguido de un Slider que alterna vistas, al lado derecho el slider mostraría una vista de chats generales o nuevos, donde al darle clic a la zona de texto entraras a una nueva vista con una card que tiene un input para textos, una zona de visualización de mensajes y un globo de icono con el nombre de perfil del usuario, al darle clic al icono de usuario puedes ir a ver su perfil o ver su historia si la tiene, y al lado izquierdo el Slider mostrará una vista de chats recurrentes o importantes que es igual que la vista de chat comunes solo que cambian los perfiles, abajo tendrá una zona con iconos y perfiles de usuarios seguidos/compartidos y al darle clic mostrar una vista con su historia o sino tiene mostrar una vista con el perfil.

Perfil:

El perfil tendrá la barra perfil menú, la zona de publicaciones del usuario, una zona de logros que se otorgan en la aplicación, zona de gustos, hobbies y aficiones del usuario en un botón desplegable, el nombre y los apodos de cada usuario, más un fondo personalizable, un ícono de perfil que se puede modificar, y un botón para enviar a la zona de ajustes.

A parte de la zona de menú superior el perfil contendrá una zona para publicaciones posteadas dividida en una sección de 2 por dispositivo para celular y 4 para pc posterior a eso tendrá otra sección con 2 botones los cuales será uno de logros que al darle clic mostrara una vista pequeña de que logros tiene el usuario otorgados por la app los cuales estarán en cards con fecha de obtención, puntos y nombre del logro, otro botón que despliega una vista con los gustos que tiene el usuario (también pensé en un botón que se despliega hacia abajo) que muestre únicamente 5-7 gustos si se llegan a pasar el programa negaría otra (etiqueta) (los gustos son establecidos y elegidos en la app no auto ingresados) del usuario, y la sección final seria un fondo personalizable del usuario (la imagen será mirada para no ver contenido sexualizable) y en el centro un icono del usuario (misma validación que el fondo) ambos al darle clic pedirá acceso interno a la galería para colocar imágenes internas o propias.

Búsquedas:

Este contendrá el menú de la app, una zona central donde habrá publicaciones, videos o frases publicadas por usuarios que al hacer clic o hundir llevará a esta misma, en la zona izquierda perfiles sugeridos o conexiones de seguidos, en la zona inferior una caja de texto de búsqueda donde mostrará una vista que tendrá un deslizador para filtrar contenido entre publicaciones, hashtags o gente, una caja central que mostrará los resultados más similares y al final el cuadro de texto donde se buscará.

Estará organizado con una sección principal de contenido con las publicaciones hechas por usuarios, al darle clic a una publicación mostrará una vista a parte con la imagen o el post, un titulo y el logo o el perfil del usuario, si le das al icono te lleva a su perfil o si tiene historia abrirá una vista externa con la historia.

Un menú orientado verticalmente con perfiles de usuarios sugeridos que al darle clic muestra o te lleva a su perfil

Y Un input que al darle clic te muestra una vista con un tipo de card grande que contiene un slider para alternar entre contenido, persona o hashtags lo cual filtra el tipo de contenido buscado, en el centro irán las respuestas si son post el contenido ubicado en el centro con el icono del usuario y su título (al darle clic al icono hace lo que ya dije), y un input de tipo texto al fondo que lo que hace es consultar dependiendo los filtros en la base de datos para mostrar los resultados, también un botón de cerrar que devuelve a la vista principal.

Ajustes:

Tendrá un deslizador ocultable para la selección entre configuraciones generales y de perfil, para configuraciones personales se podrá modificar las publicaciones hechas, los gustos, el nombre o apodo, el fondo y la foto de perfil, para la zona de ajustes generales tendrá un botón para el seguimiento que mostrará la cantidad de horas en la app, eliminar la cuenta que modificara la base de datos, una zona para reportes, una para modificar la apariencia del sistema (colores app, o icono de la app) y una zona al final para cambiar de cuentas, que cerrará sesión y llevará al login.

El módulo de ajustes tendrá un deslizador ocultable para alternar entre configuraciones personales y generales; en las personales el usuario podrá modificar o eliminar publicaciones, cambiar sus gustos, editar nombre y apodo, personalizar fondo y foto de perfil; en las generales podrá consultar el tiempo de uso en la app, eliminar su cuenta (con confirmación y modificación en la base de datos), enviar reportes, cambiar la apariencia (colores o ícono de la app) y cerrar sesión o cambiar de cuenta redirigiéndose al login, siempre contando con un botón de volver a la vista principal.

El nav será ocultable por medio de un botón arriba de él, el cual contendrá un icono con un A linkeado a cada pagina de la app 5 iconos, perfil (manda a la pagina de perfil), post (manda a la pagina para post), inicio (manda a la página principal), Búsqueda (manda a la página de búsquedas) y chats (manda a la pagina de chats) si no hay un usuario logueado al darle clic a cualquiera de los iconos o a cualquier post, icono, o historia automáticamente te llevara a la sección de login.

Se tiene planeado hacer un filtro de edad para los contenidos en el hecho de que si hay usuarios menores halla un bloqueo de acceso o no le aparezcan publicaciones subidas de tono.

Requerimientos No Funcionales

Rendimiento

El tiempo de respuesta de la aplicación no debe superar los 3 segundos en operaciones críticas (login, carga de publicaciones, envío de mensajes).

La app debe poder manejar al menos 10,000 usuarios concurrentes en su versión inicial.

Seguridad

Uso obligatorio de JWT (Json Web Token) para autenticación y autorización de usuarios.

Comunicación cifrada mediante HTTPS (TLS 1.2 o superior).

Los datos sensibles de los usuarios (contraseñas, correos) deben almacenarse cifrados.

Escalabilidad

La aplicación debe permitir la incorporación de nuevos módulos (IA de imágenes, filtros, nuevas secciones) sin necesidad de una reestructuración completa.

La arquitectura debe soportar crecimiento global.

Usabilidad

La interfaz debe ser intuitiva y consistente con los patrones de apps sociales conocidas (Instagram, TikTok, Twitter).

La aplicación debe estar disponible con un fondo de pantalla hecho por mi autoria

Accesibilidad: fuentes legibles, compatibilidad con lectores de pantalla básicos.

Los botones tendrán colores cálidos y atractivos a la vista de los usuarios, las imágenes serán suaves moderando un contenido

Compatibilidad

Compatible con Android 10 o superior.

Será hecha de manera responsiva con un enfoque tanto para computadores como celulares o dispositivos móviles

Desarrollada en Android Studio y Visual Studio, con soporte multiplataforma (móvil y eventualmente web).

Los lenguajes mínimos: HTML, CSS, JavaScript, Java, Kotlin.

Disponibilidad y Confiabilidad

Disponibilidad mínima del sistema: 99% mensual.

Manejo de fallos con mensajes de error claros sin afectar la experiencia del usuario.

Guardado automático de publicaciones en caso de fallo o cierre inesperado.

Mantenibilidad

El código no debe superar las 150 líneas por módulo, fomentando el modularidad.

Debe estar documentado y versionado en un repositorio colaborativo (Git).

Privacidad

Cumplimiento con normativas de protección de datos (ej. GDPR/CCPA si el alcance es global).

Los usuarios deben poder eliminar su cuenta y datos de manera permanente desde la app.

TABLA:

| **Categoría** | **Requerimiento** | **Descripción Específica** |
| --- | --- | --- |
| **Login** | Registro de usuario | Botón que abre vista con inputs de nombre (no nulo), correo (validación de formato y confirmación de existencia) y contraseña; botón para registrar (inserta en la BD) y botón de volver a inicio. |
|  | Ingreso de usuario | Botón que abre vista con inputs de correo y contraseña; al validar datos en la BD permite el acceso a la ventana principal; botón de volver que retorna al inicio. |
|  | Noticias de anime | Botón que abre vista con card central conectada a API de anime, mostrando imagen y nombre; al seleccionarlo muestra noticia relacionada; botón de volver al inicio. |
| **Pestaña Inicial** | Panel ocultable | Panel con pestañas de perfil, publicación, búsqueda, chats e inicio. |
|  | Publicaciones | Cards centrales con fotos o posts; zona de perfil (acceso a perfil del usuario), zona de likes (incrementa/decrementa según interacción), comentarios (vista externa con input y botón para postear en la BD). |
|  | Barra lateral derecha | Perfiles sugeridos e historias; clic en perfil con historia → muestra historia; sin historia → abre perfil. |
|  | Menú superior ocultable | Botón que despliega/oculta menú superior. |
| **Chats** | Panel ocultable | Incluye selector tipo deslizador para chats comunes o destacados. |
|  | Vista de chats | Slider con dos vistas: derecha (chats generales/nuevos), izquierda (chats recurrentes/ importantes). Cada chat abre vista con card de mensajes, input de texto y perfil. |
|  | Perfiles seguidos/compartidos | Íconos en la parte inferior; al clic mostrar historia o perfil. |
| **Perfil** | Información del perfil | Barra menú, publicaciones del usuario (2 columnas en móvil, 4 en PC), nombre, apodos, fondo e ícono de perfil personalizables. |
|  | Logros | Botón que abre vista con cards de logros (nombre, puntos, fecha de obtención). |
|  | Gustos | Botón que despliega 5-7 gustos seleccionables desde opciones de la app (no ingreso libre). Si se supera el límite, el sistema lo niega. |
|  | Fondo e ícono de perfil | Clic abre galería interna; validación para evitar imágenes sexualizables. |
|  | Botón de ajustes | Enlace directo a la pestaña de configuraciones. |
| **Búsquedas** | Zona principal | Publicaciones, videos o frases; al clic abre vista aparte con post, título y perfil (clic en perfil → abre perfil o historia). |
|  | Perfiles sugeridos | Listado lateral vertical; clic abre perfil. |
|  | Input de búsqueda | Caja inferior que abre vista con card grande + slider para filtrar entre publicaciones, hashtags o personas. Resultados centrales con ícono y título de usuario. Incluye botón de cerrar. |
| **Ajustes** | Configuraciones personales | Modificar/eliminar publicaciones, editar gustos, cambiar nombre/apodo, fondo y foto de perfil. |
|  | Configuraciones generales | Consultar horas de uso, eliminar cuenta (con confirmación en BD), enviar reportes, personalizar apariencia (colores/ícono app), cerrar sesión o cambiar cuenta (redirige al login). Siempre con botón de volver. |
| **Navegación (Nav)** | Menú ocultable | Botón superior despliega menú con 5 íconos: perfil, post, inicio, búsqueda y chats. |
|  | Restricción de acceso | Si no hay sesión iniciada, al clic en cualquier ícono, post o historia → redirige a login. |
| **Filtros de Edad** | Control de contenido | Si el usuario es menor, se bloquea o filtra contenido sensible/subido de tono. |
|  |  |  |
| **Categoría** | Requerimiento | Descripción Específica |
| **Rendimiento** | Tiempo de respuesta | ≤ 3 segundos en operaciones críticas (login, carga de publicaciones, mensajes). |
|  | Concurrencia | Soporte mínimo de 10,000 usuarios concurrentes en versión inicial. |
| **Seguridad** | Autenticación | Uso obligatorio de JWT para autenticación/autorización. |
|  | Comunicación | Todo el tráfico debe estar cifrado con HTTPS (TLS 1.2+). |
|  | Datos sensibles | Contraseñas y correos almacenados cifrados. |
| **Escalabilidad** | Nuevos módulos | Arquitectura que permita agregar funciones (IA, filtros, secciones) sin reestructuración completa. |
|  | Crecimiento global | Soporte para usuarios de diferentes regiones. |
| **Usabilidad** | Interfaz intuitiva | Estilo similar a apps sociales (Instagram, TikTok, Twitter). |
|  | Modos | Disponibilidad de modo oscuro. |
|  | Accesibilidad | Fuentes legibles, soporte para lectores de pantalla, botones con colores cálidos y amigables. |
| **Compatibilidad** | Dispositivos | Compatible con Android 10+, responsivo para PC y móviles. |
|  | Herramientas | Desarrollada en Android Studio y Visual Studio. Lenguajes mínimos: HTML, CSS, JavaScript, Java, Kotlin. |
| **Disponibilidad y Confiabilidad** | Uptime | Disponibilidad mínima del 99% mensual. |
|  | Manejo de fallos | Mensajes de error claros y guardado automático de publicaciones en caso de cierre inesperado. |
| **Mantenibilidad** | Código | Máx. 150 líneas por módulo, modularidad obligatoria. |
|  | Control de versiones | Código documentado y versionado en Git colaborativo. |
| **Privacidad** | Normativas | Cumplimiento con GDPR/CCPA (si es global). |
|  | Eliminación de datos | Posibilidad de eliminar cuenta y datos de manera permanente. |

Hotfix cambio a producción

Trabajamos el feature

Release importante (se reúnen todos los feature)

Develop qa

Pull cambios hechos por otra persona

Commit feat “mensaje”

Pull origin (más recomendado)

Clone (clonar la ruta para traer el proyecto)

Stash (guardar en memoria los cambios para ver si estos sirven o no (usar mucho la consola del visual studio code))