



Universidad Nacional Autónoma de México

Facultad de Ingeniería

Cómputo Móvil

Profesor:

Ing. Marduk Perez de Lara Dominguez

Grupo: 03 | Semestre: 2024-2

Proyecto Final:

PumaVoice

Equipo 4:

Leyva Mercado Christian Alejandro

López Becerra Ricardo

Rivas Solís Carlos Eduardo

Fecha de entrega: 24/05/2024

Índice

Índice.....	1
Introducción.....	1
Breve descripción de la app.....	2
Características y análisis de viabilidad.....	3
Características del producto mínimamente viable.....	3
Alcance de la implementación.....	7
Wireframes y descripción detallada.....	9
Pantalla de inicio.....	9
Proceso de registro.....	10
Pantalla de inicio de sesión.....	13
Pantalla principal.....	14
Función de búsqueda.....	15
Mensajería.....	16
Flujo de datos.....	18
Stack de desarrollo.....	20
Frontend.....	20
Flutter.....	20
Backend.....	21
Firebase.....	21
Publicación en las tiendas de aplicaciones.....	22
Conclusiones.....	23
Referencias.....	24

Introducción

La libertad de expresión es uno de los derechos fundamentales que toda persona debería gozar, ya que esta forma nuestros sistemas políticos y sociales, y aunque los avances tecnológicos como las redes sociales nos dan la gran facilidad de divulgarlas a una mayor cantidad de personas, esta presenta un gran problema, y es sobre la divulgación de enormes cantidades de información falsa e incluso el miedo a la censura por parte de las demás personas, como represalias a los que opinan en contra de ciertas ideologías.

Este problema se incrementa en el contexto de los estudiantes universitarios, una etapa muy importante para estos, donde desarrollan sus formas de relacionarse en un ambiente profesional, y aunque existan plataformas o grupos en los que los estudiantes pueden opinar, comentar o publicar cosas relacionadas a su entorno, siguen sufriendo el grave problema del miedo a la censura, lo que hace que varios desistan de dar su opinión y en otros casos, por la falta de sistemas de filtro de quienes ingresan, existen muchos casos de personas ajenas a las instituciones que publican información falsa con varios motivos.

En este contexto, nuestro objetivo es dar una herramienta que solucione todos estos problemas a los estudiantes, un ambiente en el que puedan expresarse libremente sin miedo a las represalias, y teniendo la seguridad de que todos los que participen sean miembros de la comunidad estudiantil, reduciendo en gran medida la información falsa por parte de terceros.

Breve descripción de la app

Nuestra aplicación que mostramos **PumaVoice**, busca resolver este problema para los estudiantes, al ser una aplicación que busca que los estudiantes universitarios, por medio de “comunidades”, puedan interactuar entre ellos, expresándose y publicando cosas en común de acuerdo a las comunidades que formen adentro, teniendo la ventaja de que pueden publicar y comentar de manera anónima, lo que les da mayor libertad de expresarse sin miedo a sufrir represalias en caso de que sus comentarios no sean del agrado de ciertos grupos.

Esto puede parecer parecido a otras aplicaciones de comunidades, como lo es **Reddit**, pero en realidad tiene más parecido una aplicación llamada **Blind**, la cual también funciona por medio de comunidades que se pueden formar, pero con la mayor característica es que esta aplicación es exclusiva para la gente dedicada a la industria de la tecnología de Estados Unidos, esto por medio de un filtro que solo

permite ingresar a aquellas personas que tengan un correo empresarial como lo es el de @microsoft.com, evitando así que gente ajena a la comunidad ingrese.

Nuestra aplicación funcionará de la misma manera que estas aplicaciones, especialmente con **Blind**, ya que nuestra aplicación usará un sistema de filtrado que permitirá únicamente a los estudiantes con correos de universidades, como por ejemplo @comunidad.unam.mx o @alumno.ipn.mx, asegurándose que únicamente gente de la comunidad hagan uso de este y se desarrolle en un ambiente sano.

Características y análisis de viabilidad

Características del producto mínimamente viable

Creemos que **PumaVoice** tiene el potencial de expandirse y tener un universo de características, pero la idea está fundamentada en un pequeño conjunto de características que garanticen el cumplimiento de la visión de nuestra aplicación. Las características que definirán a **PumaVoice** son las siguientes:

- 1. Verificación de pertenencia a la comunidad universitaria:** Los usuarios deben verificar su pertenencia a la comunidad universitaria a través de un correo electrónico institucional. Esto asegura que solo los miembros auténticos de la comunidad universitaria pueden participar en las discusiones.

Viabilidad: **PumaVoice** depende de esta característica para su funcionamiento y para diferenciarse de su competencia, pero involucra depender en parte de la infraestructura de la Universidad. Un riesgo es sufrir de censura por parte de la universidad al bloquear los correos de verificación de la app. A pesar de esto, sería posible continuar verificando cuentas cambiando la dirección de donde se envía el código o link de verificación.

En el lado técnico, se deberá implementar un sistema de hashing del correo electrónico del usuario para no tener que almacenar el correo en si, pero si detectar el intento de crear varias cuentas con el mismo correo. Esto se realizaría de manera similar a como se crean hashes de contraseñas en la mayoría de sistemas informáticos seguros.

2. **Foros temáticos:** Los usuarios pueden participar en una variedad de foros temáticos, que abarcan una amplia gama de temas. Estos foros son creados, manejados y moderados por la propia comunidad, lo que ayuda a mantener un ambiente de respeto y colaboración.

Viabilidad: El uso de las comunidades puede ser un excelente atractivo para los usuarios al unirse a nuestra aplicación, teniendo la oportunidad de crear comunidades e interactuar con gente con intereses en común, creando también ambientes de interacción más sanos entre las personas. Un riesgo con respecto a esto es que también se formen comunidades con ideales bastante tóxicos o incluso de índoles ilegales. Aún así con la creación de un algoritmo que detecte palabras clave que usen las comunidades más tóxicas y un constante monitoreo de cómo cambian sus conceptos, para poder detectar estos grupos y darlos de baja, con esto se volvería viable el uso de comunidades entre los estudiantes, aunque la propia aplicación de este método haga más costoso el desarrollo y mantenimiento de la aplicación.

Otro riesgo con respecto a esta característica es que no todas las facultades otorgan correos institucionales específicos para esa facultad, por lo que algunas comunidades no podrían acceder a la característica de foros específicos de sus facultades o al menos no con las mismas características de seguridad del resto de la aplicación.

3. **Publicación de contenido anónimamente:** Los usuarios pueden publicar texto, imágenes y videos en los foros. Esto permite una amplia variedad de formas de expresión y facilita discusiones más ricas y variadas.

Viabilidad: Esta característica es una de las que presentará un mayor riesgo técnico ya que involucraría almacenar grandes volúmenes de información y procesarla a grandes velocidades. El reto técnico de diseñar la infraestructura del sistema para soportar algo así sería importante y es posible que para tener un producto viable rápidamente se tenga que prescindir de algunas características. Una aplicación de estas características probablemente también requerirá una gran inversión económica en servidores o proveedores en la nube.

4. **Avatar personalizable y sistema de insignias:** Los usuarios pueden personalizar su avatar y ganar insignias especiales a través de su

participación en la plataforma. Esto añade un elemento de gamificación que puede incentivar la participación.

Viabilidad: Esta característica motivará a los usuarios a hacer un mayor uso de la aplicación, siendo más participativos y expresivos, en las comunidades, con tal de ir obteniendo insignias o elementos de decoración para sus avatares. Un riesgo que esto presentaría sería de que las comunidades se llenen con participaciones “vacías”, con tal de ir juntando los suficiente para obtener las insignias. La forma de evitar esto sea con la implementación de un sistema que detecte las participaciones vacías como repetir las mismas palabras en todos lados; otra alternativa de esto sea el uso de un sistema de “likes” en las publicaciones y comentarios, para que la propia comunidad diferencie los comentarios efectivos de los vacíos.

5. **Chat de conversaciones directas:** Los usuarios pueden tener conversaciones directas con otros miembros de la plataforma. Esto puede facilitar discusiones más profundas y personales.

Viabilidad: Una vez más el mayor reto de esta característica sería procesar toda la información que se generaría y hacer que el sistema se mantenga responsive. El diseño del servicio, así como la infraestructura serían un reto importante.

Con respecto a la seguridad y tomando en cuenta las limitaciones de una app joven, la moderación de contenido en los chats de mensajes directos quedaría relegada a los usuarios los cuales contarían con la herramienta de bloquear y reportar a otros usuarios que manden contenido inapropiado.

6. **Motor de búsqueda de comunidades:** Los usuarios pueden buscar y unirse a nuevas comunidades dentro de la plataforma. Esto facilita la exploración de nuevos temas y la interacción con diferentes grupos dentro de la comunidad universitaria.

Viabilidad: Teniendo implementadas las características para almacenar grandes volúmenes de post, un sistema de indexado de comunidades no debería presentar un mayor reto y probablemente sería parte de la implementación del sistema anterior.

7. **Interfaz de usuario accesible:** La aplicación **PumaVoice** cuenta con una interfaz de usuario sencilla e intuitiva que facilita la navegación y el uso de la plataforma. Los usuarios pueden acceder fácilmente a todas las funcionalidades de la aplicación, independientemente de su nivel de experiencia tecnológica y limitaciones.

Viabilidad: Una interfaz gráfica intuitiva y atractiva es fundamental para la atracción y retención del público. Afortunadamente los frameworks de desarrollo móvil actuales nos permiten hacer interfaces atractivas de manera rápida a comparación de tecnologías más antiguas. Por lo mismo, consideramos que esta característica es no solo viable, sino necesaria.

Además de estas características que consideramos esenciales para cumplir con la visión de **PumaVoice**, estas otras características son consideradas, pero no se esperaría a terminar su implementación para lanzar la aplicación.

Varias de las características mostradas a continuación están relacionadas con la monetización de la aplicación, pero para que esta sea viable en primer lugar se requeriría de una gran base de usuarios que pueda comprar los productos, lo que no se lograría hasta que la aplicación madure un poco más. De esta manera, no consideramos que la monetización tenga que estar presente en un primer lanzamiento de la aplicación y para cumplir con la visión de la misma.

1. **Tienda de cosméticos para el avatar y de insignias especiales:** Los usuarios pueden adquirir diferentes elementos para personalizar su avatar y comprar insignias especiales que destacan la participación y contribuciones de otros usuarios en la plataforma. Esta característica añade un elemento adicional de gamificación que puede incentivar la participación de los usuarios. Además añade un elemento de exclusividad que ayudará a que más personas quieran comprar los productos.

Viabilidad: Teniendo implementada la característica de personalización de avatar la venta de contenido exclusivo adicional sería trivial. El mayor riesgo sería tener la capacidad de diseñar y vender contenido atractivo para los

usuarios que sea motivación suficiente para obtenerlo. Para esto, los diseñadores podrían ser una adición necesaria al equipo por sus mejores habilidades generando contenido atractivo.

2. **Plataforma de publicidad:** Los usuarios y empresas pueden promocionar sus propias publicaciones o anuncios dentro de la plataforma. Esta característica permite a los usuarios llegar a un público más amplio dentro de la comunidad universitaria.

Viabilidad: La viabilidad de esta característica dependerá de si la comunidad universitaria se convierte en un mercado atractivo para alguna empresa ya que ellos son los que comprarán el servicio de publicidad.

3. **Comunidad de apoyo:** PumaVoice cuenta con una comunidad de apoyo en la que los usuarios pueden solicitar ayuda, resolver dudas y compartir sus experiencias con otros miembros de la comunidad. Este soporte comunitario contribuye a la creación de un ambiente de respeto y colaboración dentro de la plataforma.

Alcance de la implementación

El objetivo de nuestro proyecto, es desarrollar una aplicación que sirva como medio para crear foros de discusión para los estudiantes universitarios, donde podrán participar y comentar de manera anónima.

Entregables:

- Análisis de los requisitos de la aplicación.
- Diseño de wireframes de la aplicación.
- Aplicación híbrida completamente funcional para sistemas operativos Android, iOS y Web.

- Integración de la aplicación con los sistemas de correos institucionales de la UNAM.
- Documentación de funcionamiento de la app.

Requisitos:

- El sistema debe permitir registrarse únicamente a personas que tengan un correo institucional de la UNAM (posteriormente se buscará expandirlo a otras universidades).
- El sistema debe permitir a los usuarios crear comunidades con temáticas libres y administradas por los mismos.
- Debe permitir a los usuarios la opción de publicar y comentar de manera anónima.
- Debe de tener un sistema de personalización de avatares y de insignias que los usuarios puedan usar para decorar.
- La aplicación debe permitir un sistema de monetización por medio de una tienda virtual y publicidad interna.
- Debe permitir el uso de chats privados entre estudiantes.

Límites:

- El proyecto solo puede ser usado por la comunidad universitaria.
- No incluye el desarrollo de hardware, únicamente de software.
- Inicialmente solo se aplicará para los correos institucionales de la UNAM.

Restricciones:

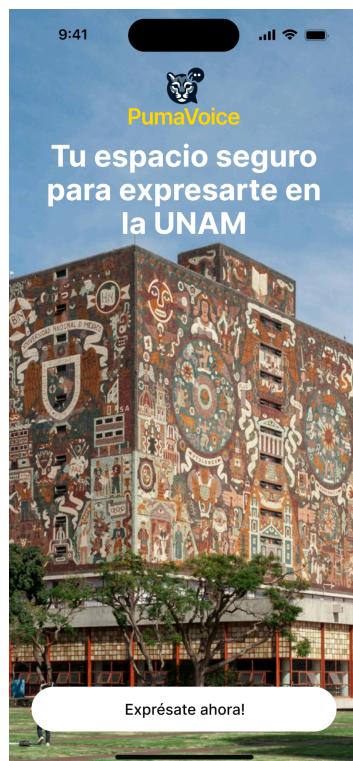
- El presupuesto disponible es de quinientos mil pesos mexicanos (MXN \$500,000.00).
- El proyecto deberá completarse en un plazo de 16 meses.
- Se asume que la institución educativa participará para poder hacer uso del sistema de correo para los estudiantes.

Criterios de aceptación:

- El sistema deberá entregarse completamente funcional y sin errores críticos.
- Todo entregable debe de estar documentado a detalle.

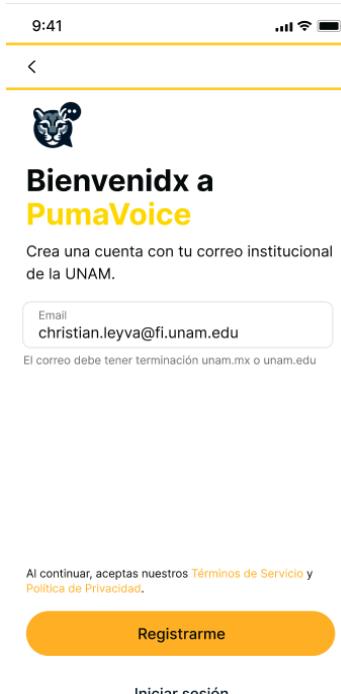
Wireframes y descripción detallada

Pantalla de inicio

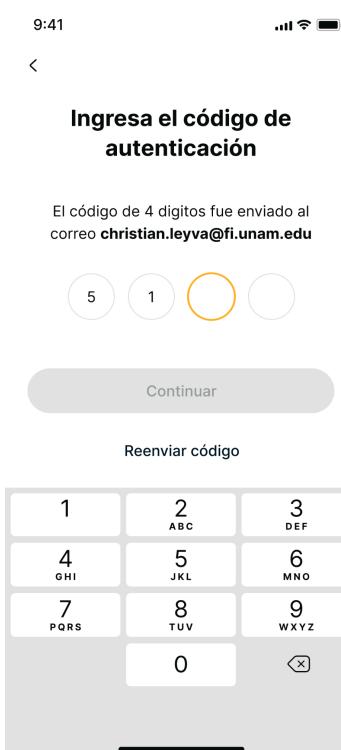


Primera pantalla que carga al abrir la app por primera vez donde se muestra una imagen decorativa, el nombre de la app y una invitación a usar la app.

Proceso de registro



Pantalla donde se invita al usuario a registrarse con su correo institucional (solo dominios unam) o al iniciar sesión si ya cuenta con una cuenta.



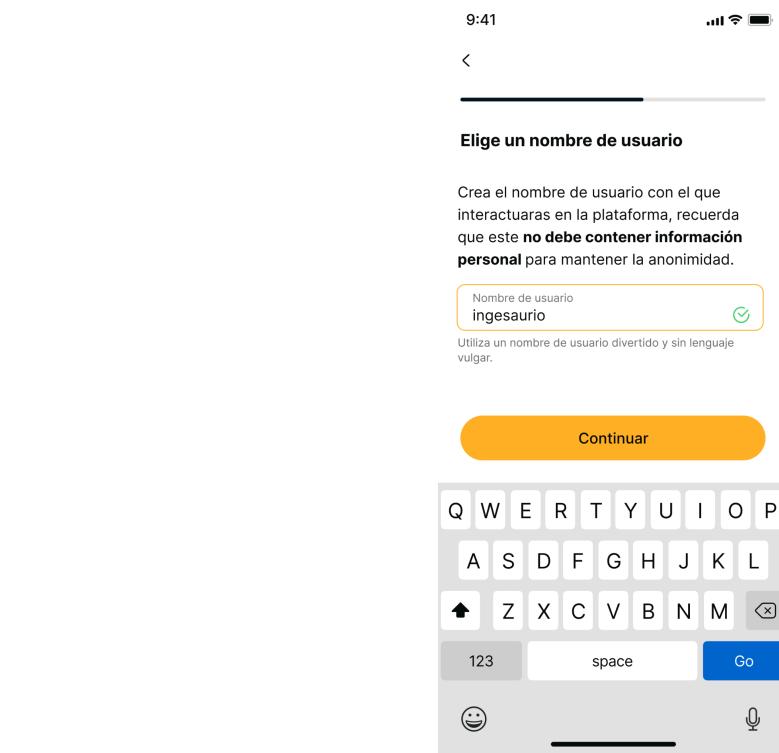
A partir del correo que el usuario ingreso, se debe colocar el código de verificación para confirmar que el correo sí le pertenece al usuario y continuar con el proceso de registro



Después de confirmar que se está haciendo uso de un correo institucional se debe crear la contraseña de 8 a 20 caracteres.



Una vez pasado los demás paso del registro el usuario debe seleccionar la escuela de la unam a la que pertenece



En esta pantalla el usuario debe crear un nombre de usuario, pero teniendo en cuenta que resguarde la privacidad del mismo (no nombres reales, etc) para así mantener el anonimato.

Pantalla de inicio de sesión



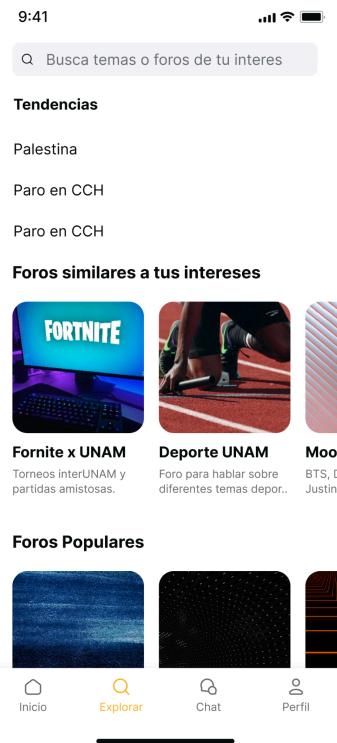
Si el usuario ya está registrado, puede ingresar a la plataforma ingresando su correo institucional y contraseña.

Pantalla principal

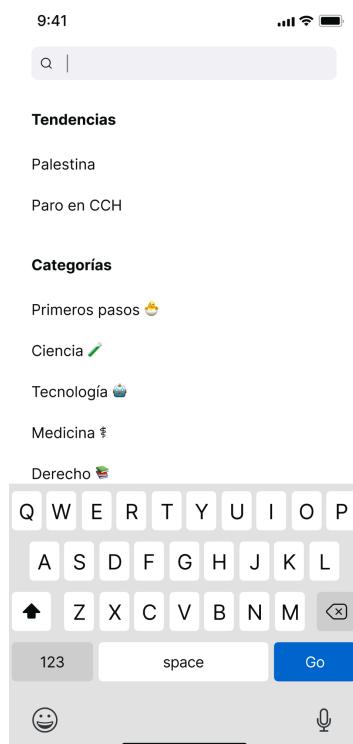


Esta es la pantalla principal de la aplicación, aquí el usuario va a interactuar con las publicaciones recientes de otros usuarios o publicar una nueva.

Función de búsqueda



En esta pantalla el usuario puede buscar el nombre de algún tema o foro que le interese, además de ver sugerencias de foros que le pueden interesar al usuario basado en su actividad.



Vista de la pantalla al tocar la barra de búsqueda, mostrando las tendencias de búsqueda y distintas categorías

Mensajería



Pantalla donde se pueden revisar los mensajes con otros usuarios, así como enviar uno nuevo a otros usuarios. Primero se muestran los mensajes sin leer.



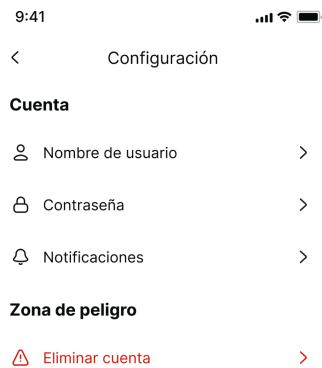
Pantalla del chat con un usuario en específico.



Esta pantalla muestra el avatar actual del usuario, además de permitirle entrar a distintas configuraciones de la aplicación, editar su perfil y hasta cerrar su sesión de la aplicación.



En esta pantalla le permite al usuario cambiar su avatar y escuela de manera sencilla.



Esta pantalla muestra las distintas configuraciones que se pueden hacer en la cuenta y hasta eliminar la cuenta de la aplicación si así el usuario lo decide.

Flujo de datos

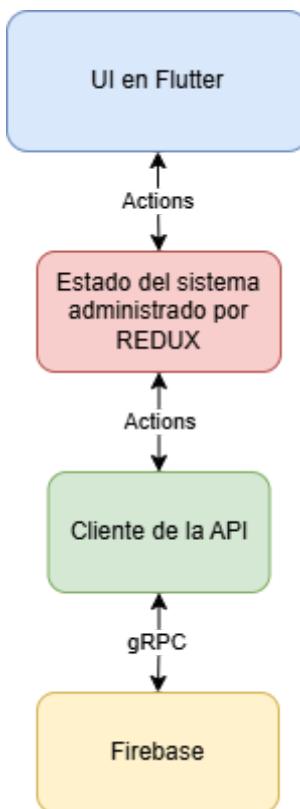
La app al ser muy similar a una red social deberá procesar grandes cantidades de información y compartirla entre sus diferentes pantallas manteniendo la fluidez. Al tener que mantener un estado muy extenso, proponemos utilizar el patrón REDUX para mantener el estado de toda la aplicación. Esto lo que nos permitirá separar la lógica de los datos de la aplicación de la interfaz gráfica declarativa. El estado será manipulado desde dos frentes, la interfaz de usuario y el cliente de la app de la API alojada en Firebase.

El cliente de la aplicación actualizará el estado utilizando Redux para asegurar que el usuario tenga siempre información actualizada y correcta. El cliente de la aplicación manejaría todas las peticiones a la API la cual dada el alcance del proyecto sería compleja y extensa.

Nuestra aplicación utilizaría una API basada en gRPC por dos razones principales:

- Desempeño: Las APIs basadas en gRPC se caracterizan por tener mejor desempeño que otros tipos de APIs. El desempeño en nuestra app es de vital importancia debido a la gran cantidad de información que se tendría que transmitir entre el cliente y el servidor.
- Integración con los servicios de Google: Google ofrece en sus servicios la posibilidad de utilizar APIs basadas en gRPC de manera muy cómoda ya que la compañía es una de las principales usuarias de esta tecnologías para sus propios sistemas.

El flujo de datos se puede resumir en el siguiente diagrama:



Flujo de datos de la aplicación resumido.

Stack de desarrollo

El stack de desarrollo a utilizar será conformado por tecnologías modernas que facilitan el desarrollo de aplicaciones móviles rápidamente y con escalabilidad. Para el frontend, es decir, la aplicación en sí, se utilizará Flutter. Para el backend se utilizarán las tecnologías ofrecidas por Firebase que permite crear backends fácilmente y escalarlos según las necesidades de crecimiento de la aplicación.

Frontend

El frontend será creado en Flutter. Su principal ventaja con respecto a otros productos similares es que Flutter ofrece un desempeño casi nativo en aplicaciones híbridas. Se decidió ir por un enfoque híbrido debido a que se necesita que la mayor cantidad de personas posibles utilicen y una de las maneras de lograrlo fue lanzar la aplicación tanto para Android como para iOS. Crear y mantener dos proyectos nativos sería muy tardado y costoso, por lo que un enfoque híbrido es ideal para lograr un mayor alcance.

Flutter

Flutter es un framework de interfaz de usuario móvil gratuito y de código abierto creado por Google. Fue lanzado en mayo de 2017 y se ha convertido en una herramienta popular para el desarrollo de aplicaciones móviles. Algunas de las características principales que ofrece son las siguientes:

1. Una sola base de código: Flutter permite crear aplicaciones móviles nativas para Android e iOS utilizando una única base de código. Esto significa que se necesita escribir dos versiones separadas de la aplicación para cada plataforma.

2. Interfaz de usuario fluida y personalizable: Flutter ofrece una interfaz de usuario que se adapta a cualquier tamaño de pantalla y orientación. Además, sigue las directrices de Material Design para aplicaciones de Android y Cupertino para apps de iOS.
3. Desarrollo rápido y iteración: Flutter incorpora herramientas como Hot Reload y Hot Restart, que permiten ver los cambios en el código casi al instante sin perder el estado de la aplicación. Esto facilita el desarrollo rápido y la iteración.
4. Integración con otros servicios de Google: Para el backend de nuestra aplicación consideramos que firebase sería la mejor opción. Firebase al ser un proyecto de Google al igual que flutter ofrece mejor interacción entre ambos proyectos.

Backend

Firebase

Firebase es una plataforma de desarrollo de backend para aplicaciones web y móviles, propiedad de Google. Lanzado inicialmente en 2011 pero posteriormente adquirida por Google en 2014, permitiendo añadir más funciones como el SDK de los productos de Google facilitando su uso. Entre sus principales funciones se encuentran:

1. Bases de datos en tiempo real: Con el uso de la “nube”, guarda los datos en un formato No SQL, siendo en este caso en formato JSON, permitiendo el poder consultar, modificar y disponer de los datos en tiempo real por parte del usuario, pudiendo actualizarlo constantemente sin participación del usuario por medio de eventos.

2. Autenticación de usuarios: Firebase ofrece un sistema de autenticación para el registro y acceso de los usuarios, parecido al de otras plataformas populares, facilitando bastante las tareas de protección y seguridad.
3. Almacenamiento en la nube: Se tiene un sistema de almacenamiento donde los desarrolladores pueden guardar el desarrollo de sus aplicaciones y sincronizarlos, siendo personalizable las reglas de estas.
4. Sistema de reporte de crasheos.
5. Laboratorio de pruebas de la aplicación.
6. Configuración remota: Este proceso ofrece la capacidad de modificar funciones o apariencia de la aplicación, sin necesidad de publicar alguna actualización o parche para la misma, ahorrando al usuario la labor de hacerlo.

Publicación en las tiendas de aplicaciones

La publicación en las tiendas de aplicaciones no representará un reto significativo debido a que la aplicación no implementa algo que no se haya hecho antes y no utiliza características fuera de las permitidas por las tiendas de aplicaciones. Uno de los objetivos de la app es ser privada para sus usuarios, por lo que no se realizaría recolección de información de casi ningún tipo.

Para algunas características podrían ser requeridos permisos de los usuarios para algunas características de privacidad. Por ejemplo, como se tiene planeado que los usuarios puedan crear post con imágenes y videos, se requeriría como mínimo solicitar acceso a las galerías de imágenes del sistema y tal vez acceso a la cámara para captura de imágenes y videos bajo la responsabilidad de los usuarios

Nuestra app pretende incluir características de monetización que serían afectadas directamente por las reglas de las dos tiendas de aplicaciones principales. Estas reglas dictan un porcentaje que se les debe dar a los proveedores de las tiendas, entiéndase Apple y Google, por las compras de contenido digital en sus plataformas. Los precios de los productos vendidos tendrían que reflejar esta característica para evitar problemas con las compañías y permitir su publicación.

Conclusiones

Con el desarrollo de este proyecto final, pudimos desarrollar aún más nuestro proyecto del **PumaVoice**, llevándolo a un nivel de planeación mayor, aplicando todo lo aprendido en el semestre, principalmente el de analizar el entorno actual para ver las necesidades que podríamos cubrir con nuestros desarrollos. Por ejemplo, nuestro producto busca sobrepasar el problema de los estudiantes universitarios de poder comunicarse y expresarse sin límites, además de poder tener acceso a comentarios e información de estudiantes dentro de la propia institución educativa, problema que nosotros vemos con un gran potencial de impactar positivamente a los demás.

A través del desarrollo de este, aprendimos más cosas del desarrollo de proyectos a nivel profesional, desde el planteamiento de este, las ideas, restricciones, criterios de aceptación, las herramientas y softwares que necesitaremos para su desarrollo, además de los stack de desarrollo. También el desarrollar los diseños y prototipos de una aplicación, además del flujo de los datos de esta, para ir teniendo una idea de hacia dónde se va a dirigir el desarrollo. Por último tomar en cuenta ciertos aspectos pequeños pero importantes del proyecto, como por ejemplo su publicación en las tiendas de app store o algunas vulnerabilidades que tendría esta en su uso posterior.

Con todo esto, podemos llegar a la conclusión de que nuestra aplicación **PumaVoice** tiene un gran potencial de impactar en la vida de los demás, promoviendo la libertad de expresión universitaria y el acceso de información a la misma, teniendo en cuenta todos los problemas de vulnerabilidades que esto podría conllevar, planeando las contramedidas a estos problemas. Con todo esto desarrollado, se puede llegar a la conclusión de que el desarrollo de este proyecto final y de todo el semestre, fue un completo éxito.

Referencias

- colaboradores de Wikipedia. (2024, 2 enero). Firebase. Wikipedia, la Enciclopedia Libre. <https://es.wikipedia.org/wiki/Firebase>
- Mora, S. L. (2022, 4 octubre). Firebase: qué es, para qué sirve, funcionalidades y ventajas. DIGITAL55. <https://digital55.com/blog/que-es-firebase-funcionalidades-ventajas-conclusiones/>
- Bing. (n.d.). Crear imágenes. Recuperado de <https://www.bing.com/images/create>
- Flutter. (n.d.). Página principal. Recuperado de <https://flutter.dev/>
- DigiAuxilio. (n.d.). Flutter vs React Native 2024. Recuperado de <https://www.digiauxilio.com/blog/flutter-vs-react-native-2024/>
- Flutter. (n.d.). Firebase y backend. Recuperado de <https://docs.flutter.dev/data-and-backend/firebase>
- Firebase. (n.d.). Página principal. Recuperado de <https://firebase.google.com/?hl=es>
- Steam Community. (n.d.). Detalles del archivo compartido. Recuperado de <https://steamcommunity.com/sharedfiles/filedetails/?id=2804456563>