UNIVERSIDAD PRIVADA DE TACNA

FACULTAD DE INGENIERÍA

Escuela Profesional de Ingeniería de Sistemas

Sistema Gestor de contraseñas: ChargePass

Curso: *Patrones de Software*

Docente: *Mag. Patrick Cuadros Quiroga*

Integrantes:

Nina Vargas, Luigui Augusto **(2019065166)**
Chambe Torres, Edgard Reynaldo **(2019064917)**
Condori Vargas,Tomas Yoel **(2018000487)**

Tacna – Perú
2025

Sistema Gestor de contraseñas: ChargePass

Versión *{1.0}*

ÍNDICE GENERAL

1. Introducción

1.1 Propósito

El propósito de este aplicativo es ofrecer una plataforma digital segura y eficiente para la generación y gestión de contraseñas personalizadas, destinada a usuarios previamente verificados mediante correo electrónico. Este sistema permitirá a los usuarios registrar sus cuentas, verificar su identidad mediante un código enviado a su correo electrónico, iniciar sesión y generar contraseñas seguras mediante dos mecanismos: un sistema automático con validaciones internas y una opción manual mediante token. El enfoque principal es garantizar la seguridad y autenticidad en el acceso de los usuarios, evitando registros fraudulentos y optimizando la generación de claves seguras.

Este sistema ha sido desarrollado utilizando Flutter para la interfaz del usuario y Firebase como backend para el manejo de autenticaciones, base de datos y lógica de verificación,incorporación de validación de dominios de correo electrónico para limitar el registro a usuarios reales y legítimos, además de la posible integración con extensiones o herramientas externas.

1.2 Alcance

La solución se presenta como una aplicación móvil (y eventualmente web) que cubre el siguiente conjunto de funcionalidades clave:

- Registro de usuarios con verificación por correo electrónico.
- Inicio de sesión únicamente después de la verificación exitosa del correo.
- Generación de contraseñas mediante:
- Opción automática con criterios de seguridad establecidos (longitud, caracteres especiales, etc.).
- Opción manual mediante el ingreso de un token personalizado.
- Almacenamiento seguro de los registros de contraseña.

Control de acceso y autenticación mediante Firebase Auth.

Posibilidad futura de:

- Limitar registros según dominios de correo válidos.
- Implementar extensiones de seguridad (ej. autenticación multifactor, validaciones de dispositivo).

1.3 Definiciones, Siglas y Abreviaturas

- **Firebase:** Conjunto de dispositivos interconectados que recopilan y comparten datos a través de internet, permitiendo una comunicación eficiente entre ellos.
- **Token:** Medida de acidez o alcalinidad de una solución, crucial para evaluar la calidad del agua.
- **Check automático:**Grado de claridad del agua, influenciado por la presencia de partículas suspendidas.
- **Validación de correo**: Sólidos disueltos totales en el agua, un indicador importante de la calidad del agua, que mide la cantidad total de minerales, sales, metales y otros contaminantes disueltos.

1.4 Referencias

Este proyecto se basa en la documentación técnica:

Documento Técnico de Desarrollo en Flutter y Firebase
Firebase Authentication Documentation
RFC 4122 – GUID/UUID standards (para tokens únicos)
OWASP Guidelines for Password Security

1.5 Visión General

Este proyecto busca brindar una solución robusta para la generación segura de contraseñas desde dispositivos móviles o web, validando primero la identidad del usuario mediante verificación por correo. La aplicación se enfocará en ofrecer una experiencia de usuario intuitiva, segura y orientada a la protección de datos. La incorporación futura de restricciones por dominios de correo electrónico asegurará que solo usuarios autorizados puedan registrarse, fortaleciendo aún más la autenticidad de los perfiles en la plataforma.

2. Posicionamiento

2.1 Oportunidad de negocio

En un entorno digital donde las amenazas a la seguridad y la autenticidad de usuarios crecen constantemente, ofrecer una herramienta de generación de contraseñas segura y con control estricto de acceso se vuelve una necesidad urgente. Este sistema, desarrollado de forma modular, podrá evolucionar para ser integrado en entornos empresariales, académicos o personales.

Las fases del desarrollo permitirán implementar:

- **Release 1:** Registro, verificación de correo, login y generación básica de contraseñas.
- **Release 2:** Gestión avanzada de tokens y contraseñas, historial de claves.
- **Release 3:** Validación de dominios de correo, integración con gestores de contraseñas y medidas avanzadas de seguridad.

2.2 Definición del problema

El uso de contraseñas débiles, la reutilización de claves y la creación de cuentas falsas son problemas comunes en los sistemas actuales. Muchos usuarios no cuentan con mecanismos seguros ni automatizados para generar y almacenar sus contraseñas, lo que expone sus datos personales y plataformas a riesgos importantes. Además, la falta de validación de correos válidos permite el registro de cuentas falsas o bots.

Este sistema resuelve estas problemáticas al ofrecer:

- Verificación obligatoria por correo electrónico antes de iniciar sesión.
- Generación de contraseñas robustas, tanto automáticas como personalizadas.
- Futuras restricciones de dominios para aumentar la autenticidad de los usuarios.

3. Descripción de los interesados y usuarios

3.1 Resumen de los interesados

Los interesados en este proyecto incluyen:

Los interesados en esta aplicación móvil de generación y gestión segura de contraseñas forman parte de distintos grupos que intervienen tanto en el uso cotidiano como en el desarrollo y la administración del sistema. Cada grupo tiene funciones específicas que contribuyen al correcto funcionamiento y evolución de la app. Los principales interesados son:

- **Usuarios finales:** Son las personas que utilizan la app para registrarse, verificar su correo electrónico y generar contraseñas de manera segura, ya sea automáticamente mediante validaciones internas o manualmente mediante un token. Buscan una experiencia fluida, confiable y fácil de usar desde sus dispositivos móviles.

- **Equipo de desarrollo:** Responsable de la implementación técnica de la aplicación utilizando Flutter como framework de desarrollo y Firebase como plataforma backend. Se encargan de mantener el sistema actualizado, corregir errores, implementar nuevas funcionalidades y garantizar un alto nivel de seguridad.

- **Administradores del sistema:** Supervisan el correcto funcionamiento de la app desde una perspectiva más estratégica. Están encargados de revisar la actividad general de usuarios, gestionar los dominios de correo válidos para registro, y monitorear estadísticas de uso, seguridad y rendimiento del sistema.

3.2 Resumen de los usuarios

Los usuarios finales de este sistema son:

Los usuarios de la aplicación son quienes interactúan directamente con ella a través de sus dispositivos móviles. Utilizan el sistema principalmente para generar contraseñas seguras y gestionar su acceso personal. Cada grupo de usuarios tiene necesidades y niveles de acceso distintos:

- **Usuarios finales:** Pueden registrarse mediante un formulario simple, validar su cuenta con un código enviado por correo electrónico y acceder al generador de contraseñas. La app les permite generar contraseñas automáticas o ingresar tokens de forma manual para mayor control. En el futuro, se incluirán validaciones que restringirán el registro a ciertos dominios de correo válidos.

- **Equipo de desarrollo:** Si bien no usan la app en el sentido tradicional, sí la prueban constantemente durante el proceso de desarrollo y mantenimiento. Sus necesidades se centran en la usabilidad, seguridad, escalabilidad y facilidad de integración con nuevas funcionalidades.

- **Administradores del sistema:** Utilizan herramientas de backend y dashboards conectados a Firebase para monitorear el uso del sistema, validar registros, analizar estadísticas y controlar dominios de correo permitidos. También pueden gestionar reportes y eventos de seguridad.

3.3 Entorno de usuario

La aplicación está diseñada para funcionar en dispositivos móviles, tanto Android como iOS, y está desarrollada con Flutter para asegurar una experiencia uniforme en ambas plataformas. Los usuarios acceden a la app a través de una interfaz intuitiva y moderna, con pantallas optimizadas para facilitar el registro, validación, inicio de sesión y generación de contraseñas.

La autenticación se realiza mediante correo electrónico, con envío automático de un código de verificación. Una vez validados, los usuarios pueden utilizar las funciones principales de la app. A futuro, se planea incorporar extensiones o filtros para restringir dominios de correo electrónico, asegurando que los usuarios provengan de fuentes válidas o instituciones verificadas.

3.4 Perfiles de los interesados

Usuarios finales:

- Necesitan una aplicación sencilla, rápida y segura que les permita crear y almacenar contraseñas sin complicaciones.
- Buscan funciones intuitivas para gestionar su seguridad personal, ya sea a través de contraseñas automáticas o tokens manuales.
- Valoran la privacidad y la protección de sus datos.

Equipo de desarrollo:

- Necesita mantener un entorno de desarrollo estable, seguro y fácil de escalar.
- Requiere herramientas de depuración y testing, además de integración continua para actualizaciones constantes.
- Trabaja en la implementación de nuevas características, como el filtro de dominios de correo.

Administradores del sistema:

- Necesitan acceso a métricas de uso, logs de actividad y configuraciones de seguridad.
- Requieren interfaces administrativas para modificar dominios válidos, auditar cuentas y responder ante eventos sospechosos.

3.5 Perfiles de los Usuarios

Usuarios finales:

- **Acceso:** Acceden mediante validación por correo electrónico.

- **Funcionalidades:** Registro, verificación de cuenta, inicio de sesión, generación automática de contraseñas y uso de token manual.

Administradores del sistema:

- **Acceso:** Acceso privilegiado al backend (por ejemplo, consola Firebase).
- **Funcionalidades:** Monitoreo de actividad, gestión de usuarios, análisis de estadísticas, control de dominios válidos y revisión de seguridad.

3.6 Necesidades de los interesados y usuarios

Usuarios finales:

- Necesitan una app funcional, rápida y segura para generar contraseñas.
- Desean facilidad para verificar su cuenta y recibir retroalimentación clara en cada paso del proceso.
- Buscan flexibilidad con opciones como el token manual.

Equipo de desarrollo:

- Necesita un entorno que facilite el desarrollo multiplataforma y permita escalar el sistema fácilmente.
- Requiere buena documentación, acceso a errores reportados y retroalimentación de usuarios para mejoras constantes.

Administradores del sistema:

Requieren visibilidad sobre el comportamiento del sistema y la actividad de los usuarios.

Necesitan herramientas para prevenir el abuso del sistema y garantizar que solo usuarios válidos accedan a las funcionalidades.

Están interesados en implementar métricas para evaluar el uso y efectividad del sistema.

4. Vista General del Producto

4.1 Perspectiva del producto

La aplicación está diseñada para ofrecer una solución segura, moderna y fácil de usar para la generación de contraseñas, dirigida principalmente a usuarios que requieren mejorar su gestión de seguridad personal o profesional. Está construida sobre Firebase para el backend y Flutter como framework multiplataforma.

Público objetivo: Usuarios en general, especialmente aquellos con conocimientos básicos/intermedios de ciberseguridad, instituciones que deseen restringir el uso del sistema a dominios de correo específicos, y personas que busquen reforzar su gestión de contraseñas.

Ventajas competitivas:

Verificación por correo electrónico que agrega una capa de seguridad.

Posibilidad de generar contraseñas seguras con un solo clic.

Opción para insertar tokens manuales, brindando flexibilidad.

Interfaz simple y amigable, enfocada en una experiencia fluida.

Futuro filtro de dominios válidos, ideal para entornos controlados (como empresas o universidades).

Escalabilidad: Aunque inicialmente orientada a un público general, la app puede personalizarse fácilmente para organizaciones que necesiten control sobre el acceso, mediante validaciones por dominio y manejo centralizado.

4.2 Resumen de capacidades

A continuación, se detallan las principales capacidades que ofrecerá el sistema:

- Consulta de disponibilidad de espacios
- Realización de reservas
- Gestión de reservas
- Notificaciones y alertas
- Métricas y reportes
- Interfaz intuitiva y amigable
- Seguridad y privacidad de datos

4.3 Suposiciones y dependencias

Suposiciones:

- Los usuarios tendrán conexión a internet estable.
- Los correos electrónicos proporcionados por los usuarios serán válidos y accesibles.
- El servicio de correo electrónico de Firebase funcionará correctamente para el envío de tokens de validación.
- Los usuarios comprenderán el proceso de verificación por correo.

Dependencias:

- Firebase Authentication para gestionar usuarios.
- Firebase Firestore (o Realtime Database) para guardar tokens si se requiere persistencia.

- Flutter como framework de desarrollo móvil.
- Servicios de terceros para análisis o estadísticas (opcional).
- Disponibilidad de la app en tiendas (Google Play / App Store en el futuro).

4.4 Costos y precios

Actualmente el desarrollo se realiza como un proyecto gratuito. No se espera monetización en las primeras fases.

Infraestructura:

- Firebase (plan gratuito): $0
- Flutter (open source): $0
- GitHub para control de versiones y despliegue: $0

Desarrollo:

- Registro, validación y autenticación en Firebase: $0
- Lógica de generación de contraseñas y tokens en Flutter: $0

4.5 Licenciamiento e instalación

- El código fuente estará disponible públicamente en GitHub.
- Contendrá una guía paso a paso en la Wiki del repositorio para compilarlo o adaptarlo.
- La app se podrá instalar directamente desde la tienda o mediante el archivo APK (dependiendo del despliegue).

5. Características del Producto

Autenticación y Seguridad

Validación del usuario mediante código enviado por correo electrónico.

Restricción futura por dominios de correo válidos.

Acceso seguro con manejo de sesiones mediante Firebase.

Generación de Contraseñas

Contraseñas automáticas generadas de forma segura.

Generación mediante tokens personalizados ingresados por el usuario.

Interfaz Amigable

Diseño minimalista, claro y centrado en la experiencia de usuario.

Adaptado a dispositivos móviles (Android/iOS).

Gestión del Acceso

Restricciones futuras para correos fuera de dominios autorizados.

Control administrativo del dominio permitido para el registro.

Notificaciones y Seguridad

Mensajes informativos en pantalla tras el registro, validación y errores.

Posibilidad de agregar validaciones por vencimiento del token.

6. Restricciones

- Uso del plan gratuito de Firebase (sujeto a límites de uso mensual).
- App desarrollada exclusivamente en Flutter.
- Dependencia del servicio de autenticación de Firebase.
- Proyecto open source alojado en GitHub.
- No incluye actualmente almacenamiento local persistente de contraseñas.

7. Rangos de Calidad

Para garantizar un funcionamiento eficiente y confiable se establecen los siguientes rangos de calidad que deben cumplirse:

Usabilidad: Diseño pensado para usuarios no técnicos. Flujo guiado paso a paso.

Fiabilidad: Firebase garantizar operaciones seguras y consistentes.

Seguridad: Uso de tokens por correo, validación del dominio (en futuro), y uso de Firebase Auth respaldado por Google.

8. Precedencia y Prioridad

El desarrollo de la app se organiza por entregas iterativas:

Release 1 - MVP (Semana 2-3):

Registro y validación por correo electrónico.

Generación de contraseñas automáticas.

Token manual básico.

Release 2 - Validaciones y Filtros (Semana 4-5):

Restricción por dominios válidos.

Mejoras en la gestión de tokens.

Validación de errores y retroalimentación visual.

Release 3 - Versión Final (Semana 6-7):

Optimización UI/UX.

Publicación en GitHub con documentación.

Pruebas en dispositivos reales.

9. Otros requerimientos del producto

a) Estándares legales

- Uso de HTTPS (Firebase lo proporciona por defecto).
- Políticas claras de privacidad en futuras versiones públicas.
- Código abierto bajo licencia MIT o similar.

b) Estándares de comunicación

- Notificaciones internas mediante alertas visuales.
- Envío de correos de validación automáticos con mensajes personalizados.

c) Estándares de cumplimiento de plataforma

- Compatible con Android desde la versión 8.0 (mínimo).
- Escalable hacia App Store con cambios mínimos.

d) Estándares de calidad y seguridad

- Prácticas seguras en el manejo de datos del usuario.
- Verificación de identidad mediante token único y limitado en tiempo.

CONCLUSIONES

La app de generación de contraseñas busca ofrecer una solución práctica, segura y accesible para todo tipo de usuarios, con posibilidades de expansión hacia entornos organizacionales más exigentes. Al aprovechar Firebase y Flutter, se logra rapidez de desarrollo, escalabilidad y una interfaz moderna.

La arquitectura modular permite incorporar nuevas funcionalidades y adaptarse a diferentes públicos sin rehacer el núcleo de la aplicación.

RECOMENDACIONES

Basado en la implementación y evaluación del app se ofrecen las siguientes recomendaciones para optimizar y expandir su funcionalidad:

1. Probar con diferentes tipos de usuarios para asegurar una experiencia fluida.
2. Implementar pruebas unitarias y funcionales para garantizar la estabilidad.
3. Considerar la implementación futura de almacenamiento seguro de contraseñas.
4. Establecer una estrategia de despliegue clara para Play Store/App Store.

BIBLIOGRAFÍA

Flutter. (s.f.). *Flutter - Beautiful native apps in record time*. https://flutter.dev

Firebase. (s.f.). *Firebase - App success made simple*. https://firebase.google.com

Dart. (s.f.). *Dart programming language*. https://dart.dev

GitHub Docs. (s.f.). *GitHub Documentation*. https://docs.github.com

Stack Overflow. (s.f.). *Stack Overflow - Where Developers Learn, Share, & Build Careers*. https://stackoverflow.com