Manual de Mantenimiento — Braille Express

**Fecha**: 26/06/2025

**Autor**: Ylia Jamile Ochoa Gutierrez

**Versión**: 1.0

# 1. Introducción

Este manual describe de manera detallada los procedimientos de mantenimiento preventivo y correctivo del sistema Braille Express, una solución tecnológica compuesta por tres componentes principales: la aplicación móvil Android, la base de datos en Firebase y el módulo de hardware físico controlado por Arduino Nano. El objetivo principal de este documento es proporcionar una guía estructurada al equipo técnico encargado del soporte y operación del sistema, con el fin de garantizar la continuidad del servicio, la integridad de los datos y el correcto funcionamiento del hardware electromecánico.

El mantenimiento adecuado de Braille Express es esencial para asegurar que la plataforma siga cumpliendo con su propósito educativo: facilitar la comprensión lectora de estudiantes con discapacidad visual mediante la traducción automática de texto digital a Braille. Dado que el sistema involucra tanto software como componentes físicos, el enfoque del mantenimiento es integral y considera aspectos como:

* Prevención de fallos en la aplicación móvil, conectividad Bluetooth y módulo Arduino.
* Monitoreo de rendimiento, errores y estabilidad en el entorno de Firebase.
* Limpieza periódica de datos innecesarios o temporales.
* Actualización de firmware, bibliotecas y librerías utilizadas en el sistema.
* Verificación del estado de los servos, la comunicación HC-06 y las respuestas del hardware.

Este manual está dirigido a desarrolladores, técnicos de soporte, personal de mantenimiento escolar o cualquier miembro del equipo responsable de mantener operativa y segura la solución Braille Express. Su correcta aplicación garantiza una experiencia de uso continua, confiable y accesible para los estudiantes y docentes beneficiarios.

# 2. Objetivos del Mantenimiento

* Prevenir fallos antes de que afecten la experiencia del usuario.
* Corregir errores en la app, base de datos o hardware de manera eficiente.
* Asegurar la integridad de la información almacenada en Firebase.
* Mantener la conectividad Bluetooth y la funcionalidad del dispositivo Braille.

# 3. Mantenimiento Preventivo

## 3.1 Tareas Semanales

* Verificar registros de errores y mensajes Logcat en Android Studio.
* Confirmar conectividad y respuesta del módulo Bluetooth (HC-06).
* Validar sincronización correcta entre Firebase y la app.
* Revisar funcionamiento físico de los 6 servomotores del sistema Braille.

## 3.2 Tareas Mensuales

* Limpieza manual de entradas duplicadas o erróneas en TbHistorial o TbConfiguracion.
* Probar funcionalidades críticas: login, registro, traducción y envío al Arduino.
* Revisar el estado de almacenamiento local (SharedPreferences) y sincronización por ID de usuario.
* Realizar pruebas en múltiples dispositivos Android con diferentes versiones del sistema operativo.

## 3.3 Tareas Trimestrales

* Verificar que el firmware cargado en el Arduino Nano sigue siendo compatible con la versión de la app.
* Realizar pruebas de desempeño del Bluetooth bajo diferentes distancias y obstáculos.
* Evaluar la estructura de datos en Firebase y optimizar reglas de seguridad si es necesario.
* Auditar el comportamiento de los ViewModels y su interacción con Firebase para evitar fugas de memoria.

# 4. Mantenimiento Correctivo

## 4.1 Gestión de Errores Comunes

* **La app no se conecta por Bluetooth:** Confirmar que el módulo HC-06 esté emparejado y encendido.
* **Firebase no guarda datos:** Revisar conexión a internet, reglas de acceso o posibles errores de permisos.
* **Traducciones no se registran en historial:** Verificar que el método de envío también invoque la función de almacenamiento.
* **Preferencias no se aplican:** Corregir posibles problemas de lectura desde SharedPreferences o de sincronización con TbConfiguracion.

## 4.2 Procedimiento de Corrección

1. Identificar el error (pantalla afectada, logs, dispositivo).
2. Replicar el error en un entorno de prueba.
3. Corregir el código o revisar la configuración (Bluetooth/Firebase).
4. Validar la solución en varios dispositivos.
5. Documentar la causa raíz y la solución aplicada.

# 5. Respaldo de Información

## 5.1 Respaldo Manual

* Desde la consola de Firebase, exportar los datos de:
* /TbUsuarios
* /TbHistorial
* /TbConfiguracion
* Descargar archivo .json y almacenarlo en una unidad segura.

## 5.2 Respaldo Programado (CRON)

* Usar funciones de exportación automática desde Firebase CLI.
* Programar tareas (CRON o scripts) en entorno de servidor para guardar copias de seguridad en Google Drive o almacenamiento local.

# 6. Actualización del Sistema

**6.1 Aplicación Android**

* Compilar nueva versión APK desde Android Studio.
* Firmar digitalmente el APK para distribución.
* Instalar la nueva versión en los dispositivos o distribuir vía Google Play o enlaces directos.

**6.2 Configuración de Firebase**

* Revisar y ajustar reglas de seguridad.
* Confirmar integridad de estructuras JSON después de actualizaciones.
* Validar conectividad con Realtime Database y autenticación.
* Usar Arduino IDE para cargar el archivo actualizado Braille.ino.
* Confirmar asignación correcta de pines y lógica de impresión Braille.

**6.3 Firmware del Arduino**

* Usar Arduino IDE para cargar el archivo actualizado Braille.ino.
* Confirmar asignación correcta de pines y lógica de impresión Braille.

# 7. Documentación de Cambios

Se registrará cada cambio en el archivo Cambios.doc, que estará en un archivo Drive Braille Express, este estará presentado bajo el siguiente formato:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Versión | Fecha | Tipo de Cambio | Descripción Detallada | Responsable |
| 1.0.0 | 15/06/2025 | Publicación inicial | Primera versión estable de la aplicación Braille Express. Incluye traducción de texto a Braille y Bluetooth. | Equipo de desarrollo |
|  |  | Agregado | - Funcionalidad de traducción texto → Braille con vista en pantalla. - Conexión a Arduino Nano vía HC-06. |  |
|  |  | Agregado | - Sistema de historial persistente. - Configuración de tema y color de botones con sincronización en Firebase. |  |
| 1.1.0 | 01/07/2025 | Mejora | Menú contextual en historial: opciones para copiar texto y eliminar entradas. | Y. Ochoa / DevTeam |
|  |  | Corrección | Ajustes en el cambio dinámico de tema claro/oscuro y aplicación del color de botones. |  |
|  |  | Mejora | Separación de lógica con MVVM y mejoras en la accesibilidad visual de la interfaz. |  |
| 1.2.0 | En desarrollo | Planificado | Implementación futura de OCR para imágenes, soporte de múltiples idiomas y panel estadístico para docentes. |  |

Leyenda:

* **Agregado**: Nueva funcionalidad o componente incorporado al sistema.
* **Corrección**: Solución aplicada a errores o problemas detectados.
* **Mejora**: Optimización de funcionalidades existentes o del rendimiento.
* **Eliminado**: Funcionalidad retirada por obsolescencia o rediseño.
* **Planificado**: Cambios en curso o pendientes de implementación.

# 8. Herramientas Recomendadas

* Desarrollo y depuración: Android Studio, Arduino IDE, Firebase Console.
* Monitoreo de errores: Logcat, Firebase Crashlytics.
* Control de versiones: Git + GitHub.
* Respaldo de datos: Firebase CLI, Google Drive, JSON export.
* Documentación técnica: Google Docs, Notion o GitHub Wiki.

# 9. Contacto de Soporte

Para incidencias mayores o errores de funcionamiento, contactar al equipo técnico en:

[soporteBrailleExpress@gmail.com](mailto:soporteBrailleExpress@gmail.com)

# 10. Conclusión

El mantenimiento de Braille Express no solo implica la corrección de fallos técnicos o la actualización de componentes, sino que representa un compromiso continuo con la calidad, la accesibilidad y la inclusión educativa. Este sistema, que integra una aplicación móvil Android, una plataforma de almacenamiento en la nube mediante Firebase, y un módulo físico con Arduino y servomotores, requiere un enfoque integral de soporte técnico para asegurar que funcione de manera óptima en entornos reales con usuarios con discapacidad visual.

La aplicación de buenas prácticas en el mantenimiento preventivo, como la verificación de logs, la sincronización de datos y las pruebas de conectividad con hardware, permite anticipar problemas antes de que afecten a los usuarios finales. Asimismo, el mantenimiento correctivo documentado y estructurado garantiza que, ante cualquier eventualidad, se pueda responder de manera rápida, segura y eficiente.

Además, la creación de respaldos periódicos, la documentación sistemática de cambios, y la revisión periódica de las configuraciones y preferencias del usuario, aseguran la continuidad del servicio, incluso en casos de fallos inesperados o cambios de dispositivo.

En el marco de un entorno educativo, donde estudiantes con discapacidad visual dependen de Braille Express para desarrollar habilidades de lectura y comprensión a través del sistema Braille, la responsabilidad del mantenimiento técnico adquiere una dimensión ética y social. Por eso, este manual debe ser entendido como una guía viva, que evoluciona junto con la tecnología, las necesidades del usuario y las mejoras del sistema.

Finalmente, es importante que el equipo técnico encargado de Braille Express mantenga una actitud proactiva, colaborativa y centrada en el usuario, promoviendo la accesibilidad, la mejora continua y la sostenibilidad del proyecto a largo plazo. Solo así se garantizará que esta solución siga siendo una herramienta efectiva, confiable y transformadora para las personas que más la necesitan.