**Sistema Apoyo Móvil - S.A.M.**

**Fase 2 - Capstone**

**Integrantes:**

* **Pierre Asencio - 21.351.129-7**
* **Claudio Castro - 20.834.339-4**
* **Bruno Lobos - 21.539.763-7**
* **Víctor Marambio - 18.865.404-5**

**Docente: Reginaldo Salinas**

**Sección: 708D**

Índice

[**1. Abstract 2**](#_heading=h.wbo4usv0oijm)

[**2. Desarrollo de Ingeniería 3**](#_heading=h.x5zoyic0hy6d)

[2.1 - Ajustes a la Propuesta considerando dificultades, facilitadores y retroalimentación 3](#_heading=h.jkw05ttj8psc)

[2.2 - Metodología de Trabajo y Estructura 4](#_heading=h.vj8pg65vln6e)

[2.3 - Avance del Proyecto y Gestión de Retrasos 4](#_heading=h.m15ljxjqkhni)

[**3. Evidencias que dan Cuenta del Avance del Proyecto 5**](#_heading=h.gkkpcv9kc6hi)

[3.1 - Diagramas UML 6](#_heading=h.sgiii1kfguqh)

[3.2 - Prototipo del Proyecto 16](#_heading=h.eh6esg2jbj58)

[3.3 - Base de Datos 18](#_heading=h.7qwojd9f2wb2)

[3.4 - Producto Mínimo Viable 23](#_heading=h.itk0dxwhyp1t)

[**4. Conclusiones 24**](#_heading=h.5d9xnsjs3whr)

# 1. Abstract

El presente informe técnico evalúa el progreso del **Proyecto APT: Sistema Asistencia Móvil**, correspondiente a la décima semana de la asignatura. El objetivo del proyecto es optimizar los tiempos de respuesta ante emergencias dentro del campus mediante una aplicación móvil que notifica inmediatamente al personal de asistencia. A partir de la retroalimentación recibida, la propuesta fue ajustada para incluir una **separación arquitectónica estricta** entre la **Plataforma Móvil (Usuario)** y el **Sistema Administrativo (Mantenedor)**. El modelo de datos se solidifica en cuatro entidades principales (**USUARIO, ATENCIÓN, SERVICIO, ADMINISTRADOR**). Se detalla la **división estratégica del equipo** en roles Backend y Frontend como respuesta a un factor externo, lo que, si bien causó un retraso en la implementación del Backend, ha fortalecido la arquitectura modular. Se presenta el Diagrama de Actividad final como evidencia de la lógica funcional validada, enfocada en la **Identificación de Personal y Notificación Push**, que constituye el corazón del Mínimo Producto Viable (MVP).

# 2. Desarrollo de Ingeniería

## 2.1 - Ajustes a la Propuesta considerando dificultades, facilitadores y retroalimentación

La retroalimentación y la dinámica interna del equipo impulsaron ajustes fundamentales en la estructura y el modelo del proyecto, garantizando el cumplimiento de los requerimientos de separación de roles y gestión de datos.

### A. Ajustes a la Propuesta por Retroalimentación Docente

El proyecto fue ajustado para formalizar la **separación de sistemas** y la **normalización de datos**:

* **Arquitectura Dual (Sistemas de Administración y de Usuario):** Se separó la lógica en dos plataformas funcionales. El **Sistema de Usuario** (App Móvil) se dedica al consumo de servicios (login, QR, solicitud), mientras que el **Sistema de Administración** (Software Mantenedor) opera para la gestión total de usuarios, servicios y registro de atenciones, garantizando que la base de datos no tenga datos poblados manualmente.
* **Modelo de Datos (Normalización):** Se consolidó el modelo en **tres tablas** principales: **USUARIO, ATENCIÓN, SERVICIO**. La tabla **ATENCIÓN** fue definida como la tabla intermedia para establecer una relación de **Muchos a Muchos** entre un **USUARIO** que solicita un servicio y el **SERVICIO** que es solicitado múltiples veces.

### B. Factores que han facilitado y/o dificultado el desarrollo

El desarrollo del proyecto fue **facilitado** por la **buena disposición y actitud proactiva** de los integrantes del equipo, lo que nos permitió mantener el avance a pesar de las limitaciones logísticas.

El principal **obstáculo** fue la solicitud externa de **reducir el número de integrantes del equipo** tras la primera entrega. Este impedimento inicial forzó un **ajuste estructural** significativo en la distribución de la carga. No obstante, este obstáculo se transformó en un factor **facilitador** al permitir una **división estructural clara de los roles** en **Backend** y **Frontend**, lo cual **optimizó la ejecución técnica** y alineó el esfuerzo a la arquitectura cliente-servidor del proyecto.

## 2.2 - Metodología de Trabajo y Estructura

La metodología de trabajo implementada es **Scrum**, adaptada a la nueva estructura del equipo. Los roles técnicos surgieron de la división obligada, asegurando la pertinencia con los requerimientos disciplinares de la Ingeniería:

* **Rol Frontend:** Responsable de la **Plataforma Móvil**, la **interfaz de usuario (UI/UX)** y la **integración del lector QR**.
* **Rol Backend:** Responsable del **Diseño de la API REST**, la **conexión con Firebase**, la lógica de **Notificación Push** y la **gestión de la BBDD**.

## 2.3 - Avance del Proyecto y Gestión de Retrasos

### Actividades Ajustadas o Eliminadas

No se eliminó ninguna actividad crítica del Plan de Trabajo. Se realizó el siguiente **ajuste estructural** para compensar la falta de integrantes: las actividades de **Modelado UML, Pruebas de Integración y Documentación Técnica** fueron **fusionadas y re-asignadas** a los roles de **Desarrollo (Backend y Frontend)**. Cada desarrollador es ahora responsable de probar y documentar su propio módulo, garantizando que la calidad sea mantenida a pesar de no contar con un rol de apoyo dedicado.

### Actividades que no han iniciado o están retrasadas

La actividad **Implementación del Backend (API y Lógica de Negocio)** está experimentando un **retraso** respecto a la planificación inicial.

* **Motivo del Retraso:** Se invirtió tiempo adicional en la **definición robusta del modelo de datos** (las cuatro tablas) para asegurar la solidez del sistema de información, además de la complejidad inherente a la configuración inicial de la integración con **Firebase** y el servicio de **Notificación Push**.
* **Estrategia para Avanzar:** Se aplica una estrategia de **priorización del Mínimo Producto Viable (MVP)** del Backend, enfocando el esfuerzo en el *core* funcional: **Validación de Token**, **Identificación de Personal** y **Notificación**. Las horas asignadas a la fase de **Documentación** serán utilizadas temporalmente para completar esta codificación esencial, asegurando que el producto funcional esté operativo antes de proceder con el informe final.

# 3. Evidencias que dan Cuenta del Avance del Proyecto

La principal evidencia del avance del diseño y la lógica funcional es la validación del Diagrama de Actividad UML Final. Este diagrama demuestra la implementación de la lógica central del negocio:

* Lógica Central del Backend: El diagrama confirma que el camino de procesamiento exitoso del API Backend realiza las acciones más importantes del sistema: Identificar Personal Asignado y Enviar Notificación Push al Personal antes de enviar la confirmación a la Interfaz, cumpliendo con el objetivo de optimizar los tiempos de respuesta.
* Modelo de Integración: El diagrama valida la correcta separación de responsabilidades entre la App, el registro histórico en BBD Firebase, y la lógica de negocio en el API Backend.

## 3.1 - Diagramas UML

| Diagrama Caso de Uso - Caso General |
| --- |
|  |

| Diagrama Caso de Uso - Caso extendido Administrar Usuarios |
| --- |
|  |

| Diagrama Caso de Uso - Caso extendido Administrar Servicios |
| --- |
|  |

| Diagrama Caso de Uso - Caso extendido Administrar Solicitudes |
| --- |
|  |

| Diagrama Base de Datos - Modelo Entidad-Relación |
| --- |
|  |

| Diagrama Base de Datos - Modelo Lógico |
| --- |
|  |

| Diagrama de Clases |
| --- |
|  |

| Diagrama de Actividad |
| --- |
|  |

| Diagrama de Secuencia |
| --- |
|  |

| Diagrama de Comunicación |
| --- |
|  |

| Diagrama de Infraestructura |
| --- |
|  |

## 3.2 - Prototipo del Proyecto

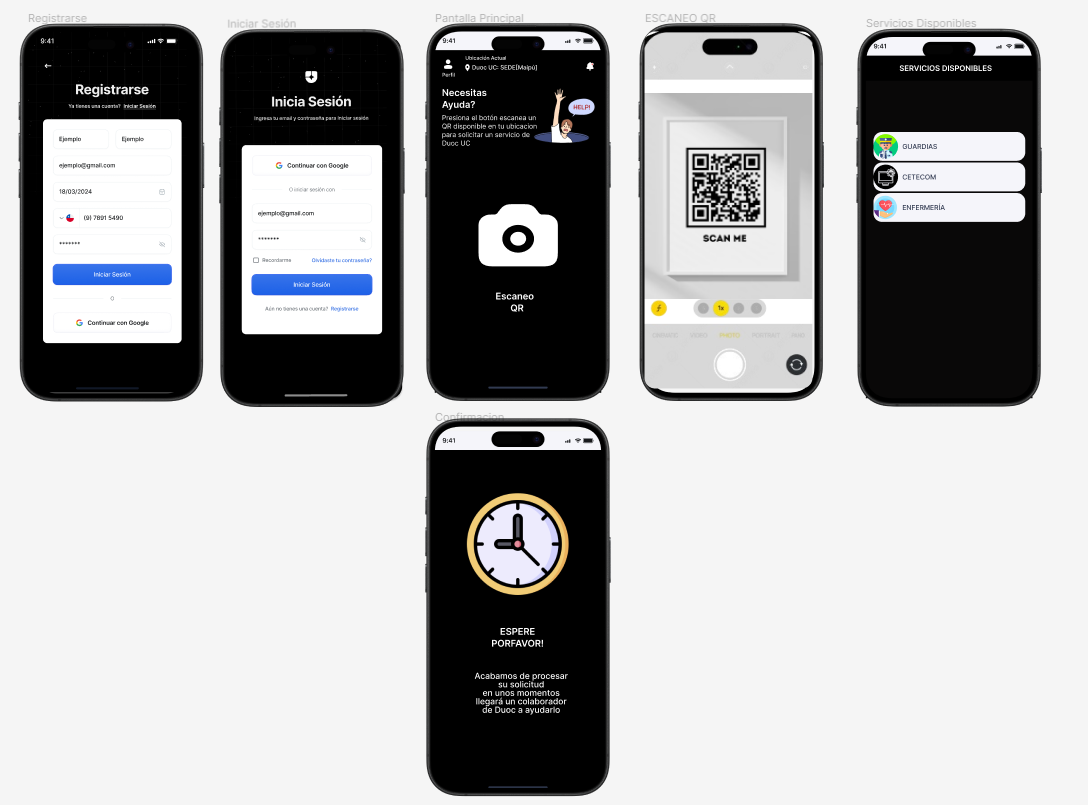
### 3.2.1 Creación del primer prototipo

### 

Partimos con esta pequeña y simple base de idea del proyecto, un google forms conectada a un google sheets.

(<https://docs.google.com/forms/d/e/1FAIpQLSfdQsQAJ4_f9oSCoLx2GxAPbskf_Vogvo-3sLIPLSzvSy3blg/viewform?usp=dialog>)

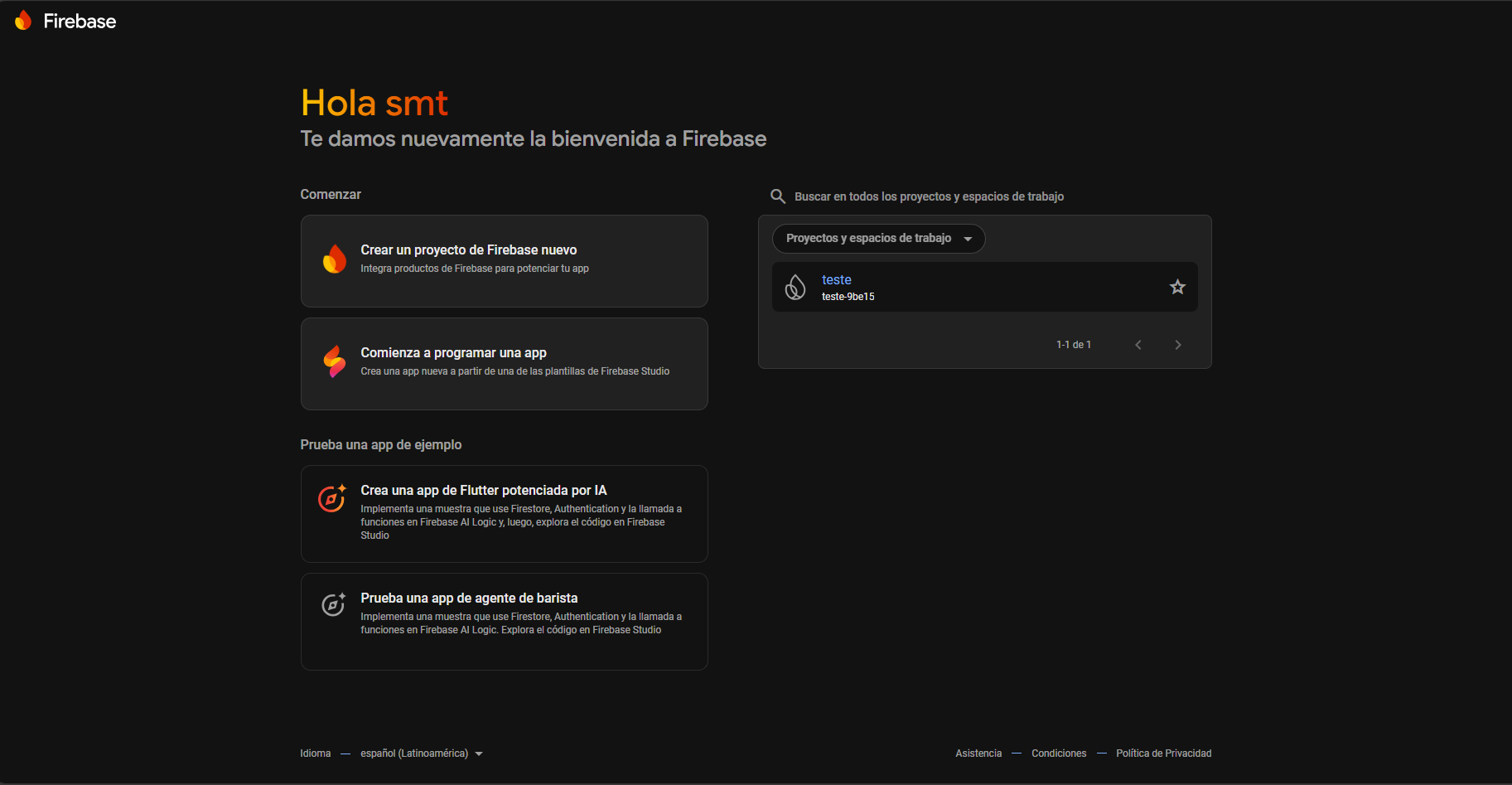
### 3.2.2 Creación del Mock Up

****

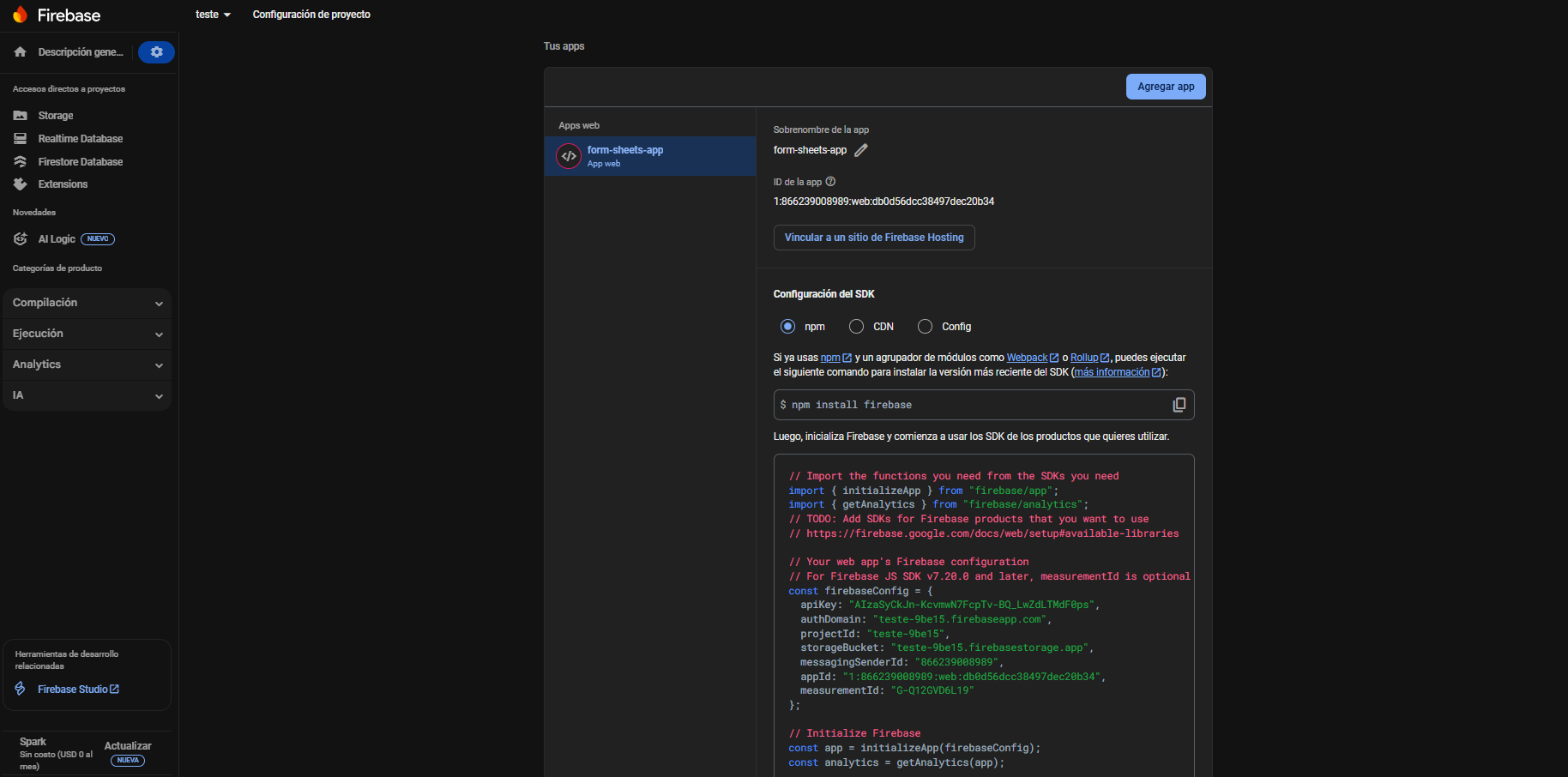
Armamos el Mock up para definir bien las matrices,diseño y parámetros que abarcará el proyecto, gracias a la metodología que usamos podemos ir modificando mientras avanzamos en él.

## 3.3 - Base de Datos

### **3.3.1 - Creación de Base de Datos**

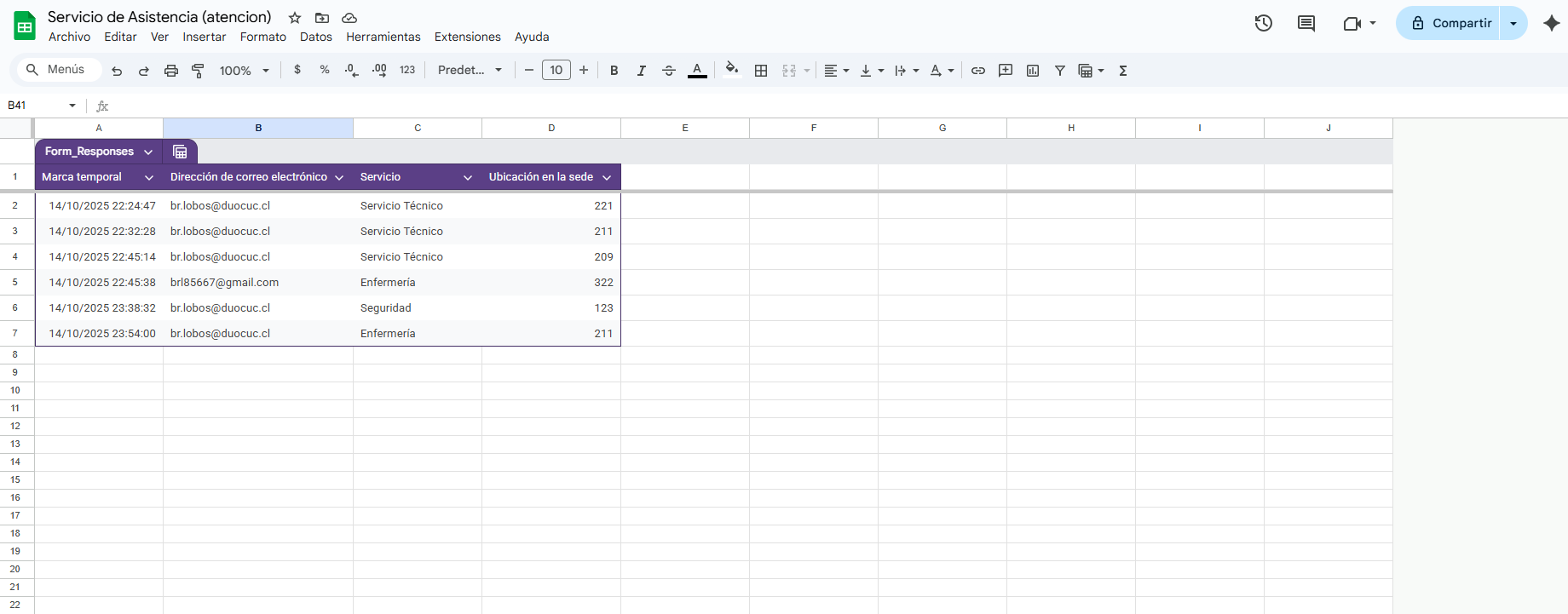
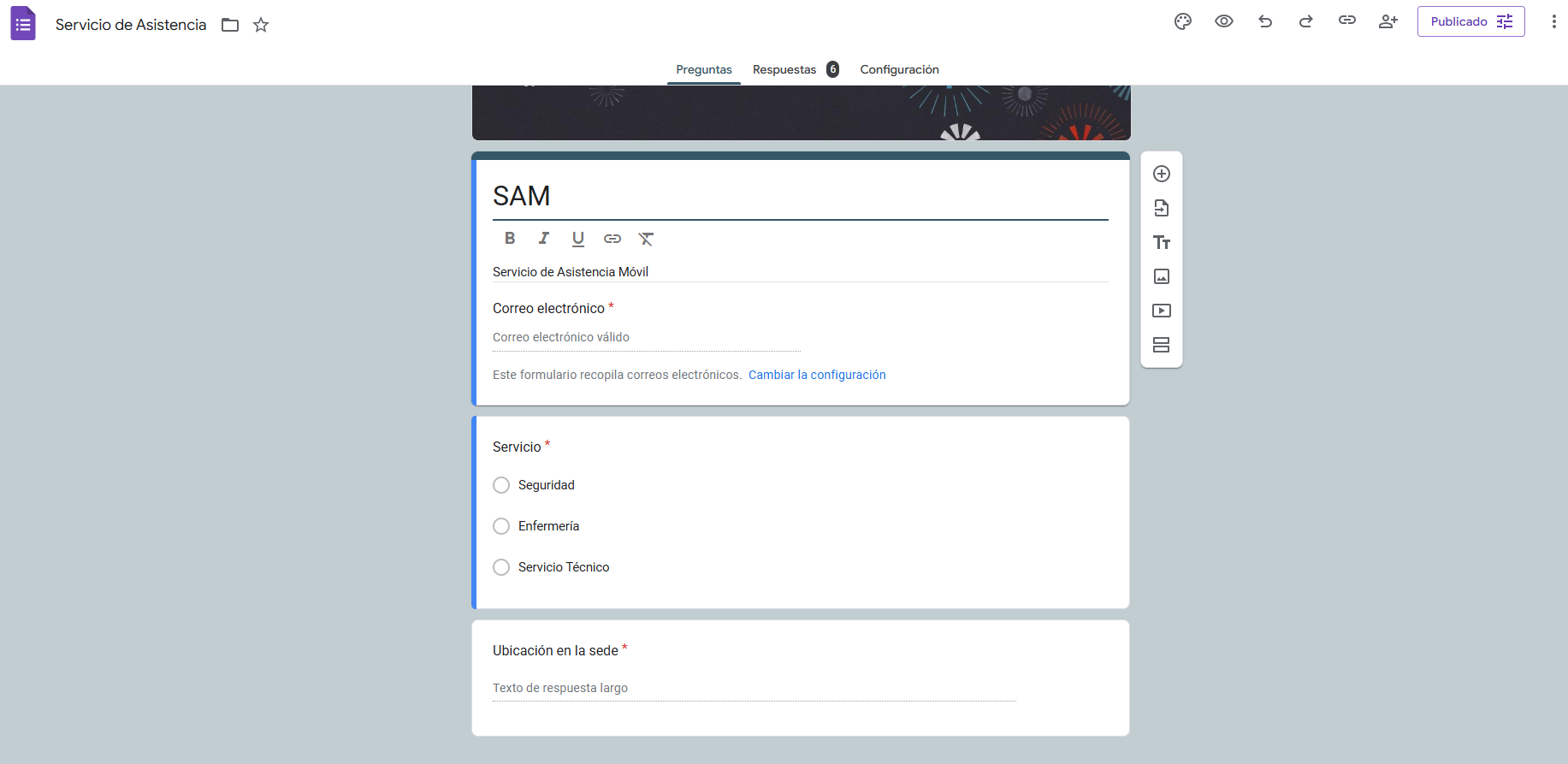


Empezó con la creación de el proyecto bajo nombre “teste”

****

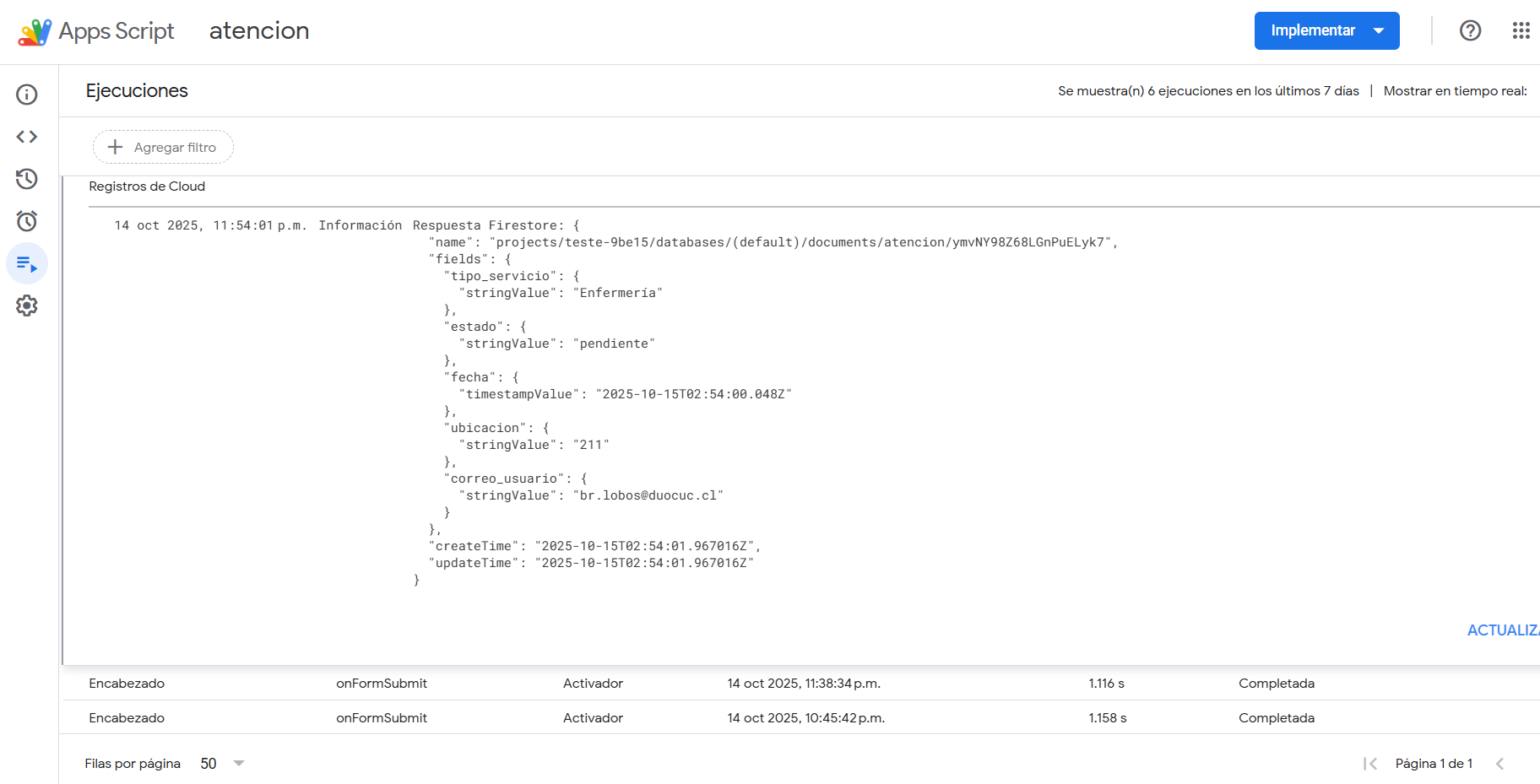
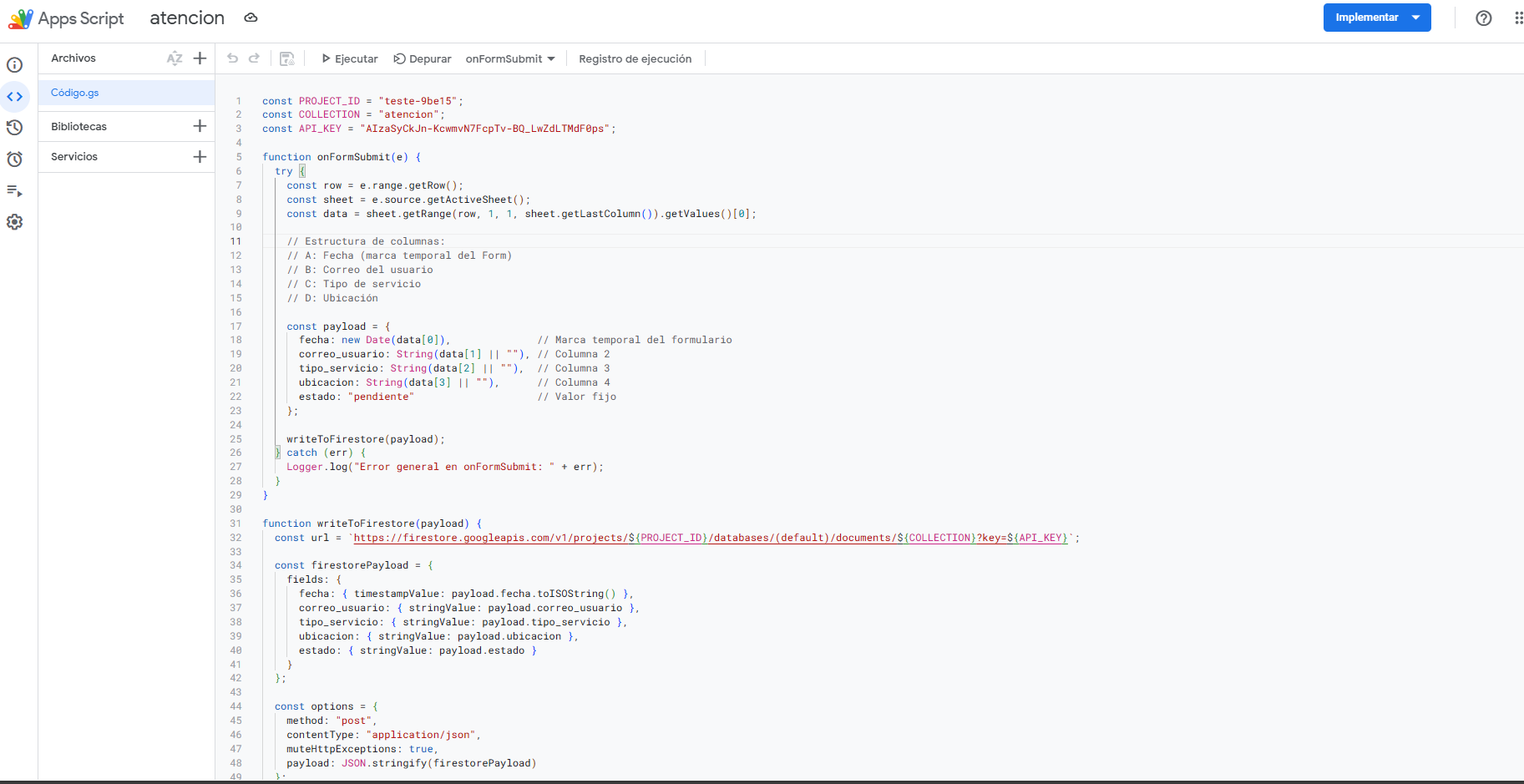
Luego se agregó una app web para crear la llave API de la BD

### **3.3.2 - Creación de Formularios Google Form y almacenamiento en Google Sheets**

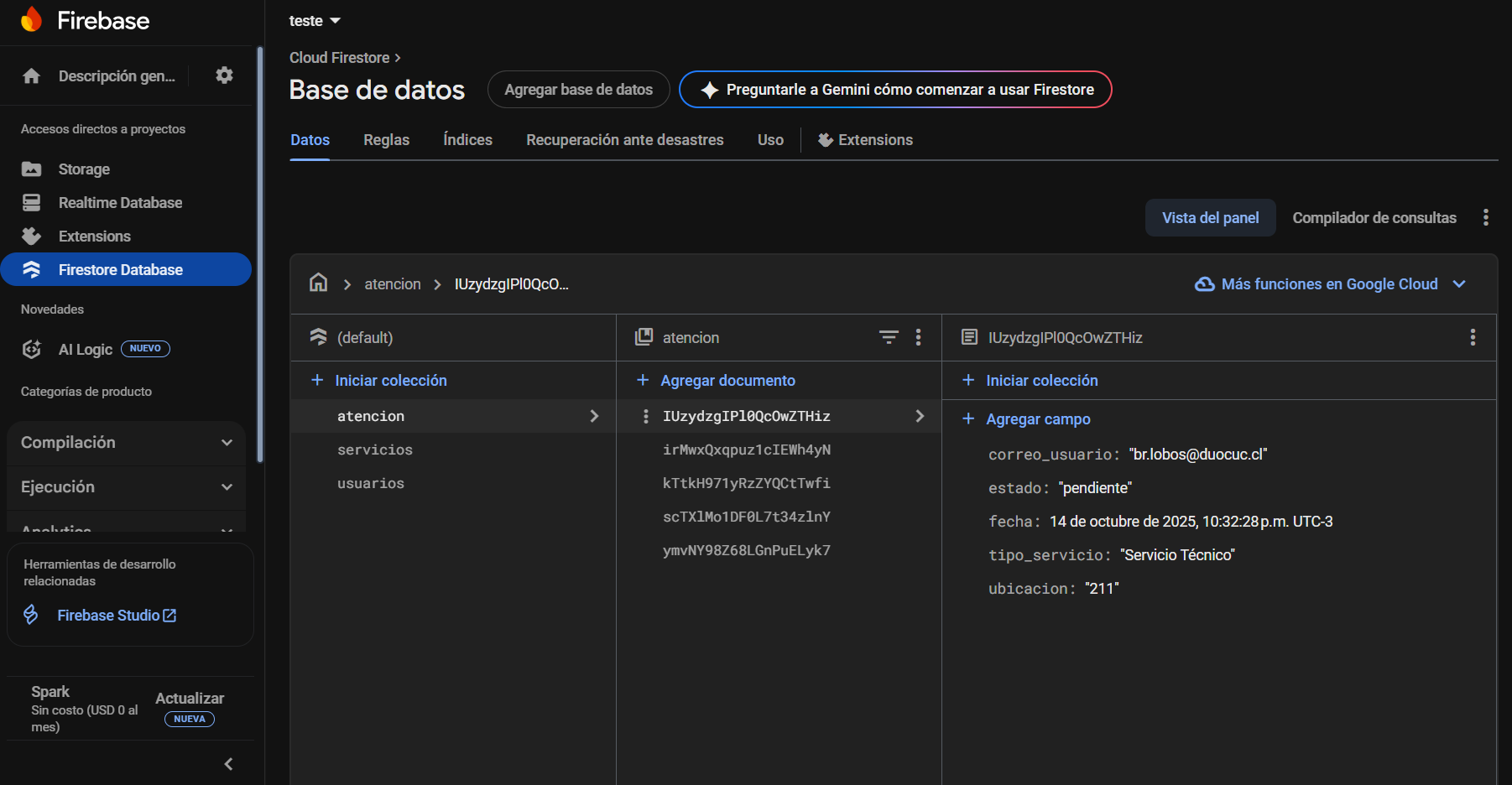


Para este avance se usó Google Forms para mostrar lo que nuestra App Móvil va a registrar a futuro, Google Forms tiene la habilidad de crear y vincular un archivo de Google Sheets fácilmente que guarda las respuestas

### **3.3.3 - Conexión Google Sheets - Firestore y poblado de Base de Datos**

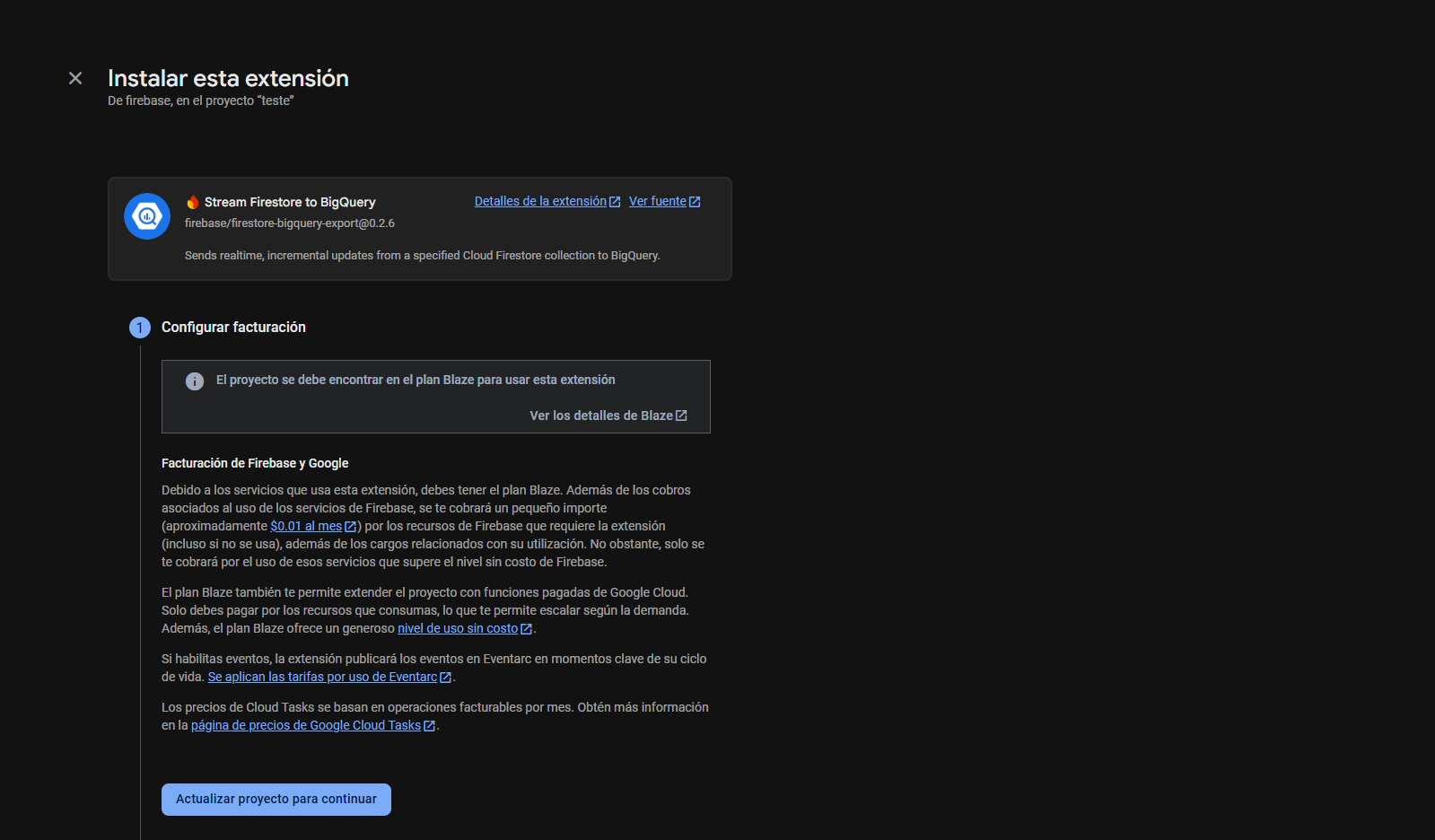


Con la extensión “Apps Script” se fabricó código que enviará los datos de los formularios almacenados en Google Sheets a Firestore, con la adición de un Trigger, los datos se envían el momento que un usuario responde el formulario de atención

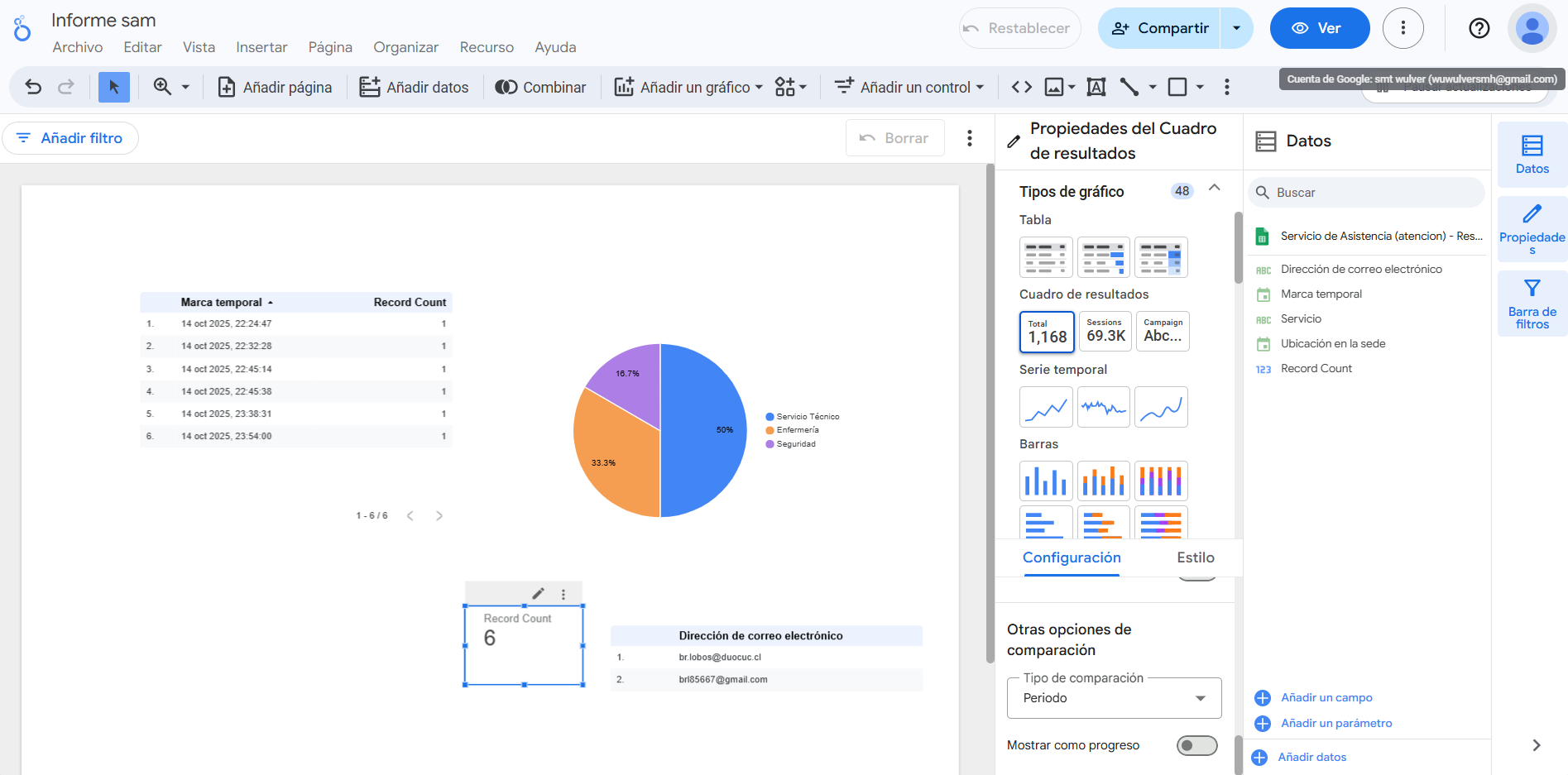


Base de datos de datos poblada mediante formularios, Firestore tiene integrada funcionalidades CRUD, uno tiene la habilidad de fácilmente Crear, Leer, Actualizar y Eliminar datos, las IDs únicas de las “Colecciones” y “Documentos” no se puede actualizar

### **3.3.4 - Extracción de datos a Looker Studio**

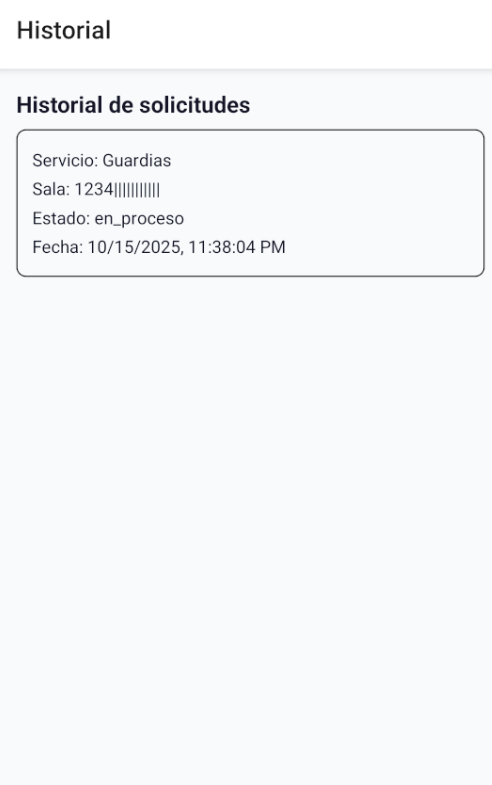


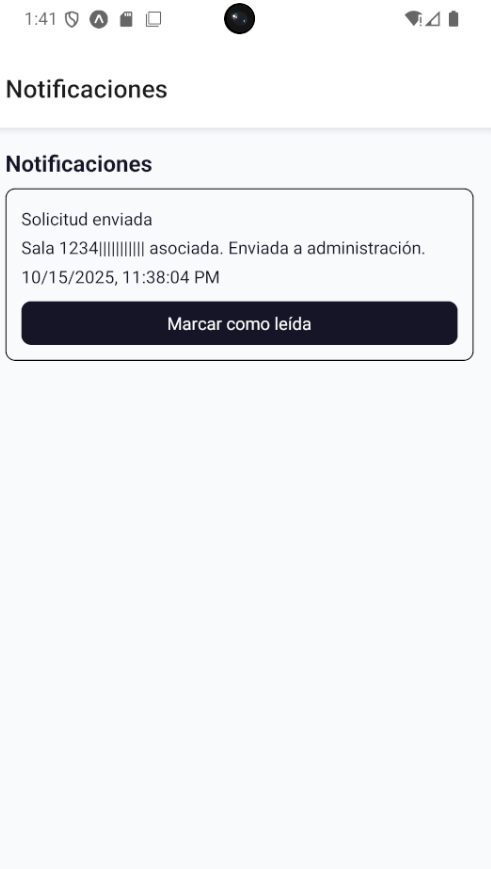
Se intentó usar BigQuery para la coneccion a Looker Studio, desafortunadamente, el proyecto, al no poseer capital, optó por otro medio para realizar el proceso de ETL



No existe conexión directa de Firestore a Looker Studio, pero si existe una conexión desde Sheets a Looker, teniendo desde ya los datos guardados en Google Sheets se creó la conexión, Looker Studio actualiza los datos automáticamente cada cierto tiempo, el operador también puede actualizar los datos cuando quiera

## 3.4 - Producto Mínimo Viable





Se realizó la primera versión de la app mobile

# 4. Conclusiones

La Fase 2 del proyecto concluye con un diseño de ingeniería validado y robusto, habiendo implementado los ajustes de arquitectura solicitados (separación de sistemas y normalización de datos). La adaptación del equipo a una estructura Backend/Frontend fue clave para transformar un obstáculo externo en un factor de optimización. Aunque la implementación del Backend sufre un retraso, la solidez del Modelo de Datos y la validación del Diagrama de Actividad aseguran que la codificación restante se enfocará de manera eficiente en el MVP, manteniendo la integridad del cronograma de entrega final.