

Proyecto (Concurso mi Mascota)

El Problema:

Desde el año 2015 se lleva a cabo el concurso Mi Mascota para decidir 12 mascotas especiales que formarán parte del calendario. Para el concurso Mi Mascota, edición 2018 se les solicitará a los participantes que envíen un correo con sus datos, los de su mascota, una anécdota breve de la misma y 3 imágenes de dicho animal, de las cuales, se pueda elegir una para algún mes del calendario.

A medida que la cantidad de participantes ha aumentado, se ha generado un problema para los organizadores del concurso, pues resulta difícil responder y validar la información de todos los participantes dentro de los lapsos de tiempo apropiados. Además, existen dificultades relacionadas con la calidad de los datos enviados por los participantes; a saber: los participantes envían sus datos incompletos y no identifican correctamente la información consignada. Por otra parte, envían fotos demasiado grandes o con muy mala resolución. Debido a esto, se ha decidido desarrollar un sistema automatizado que ayude a los encargados del concurso a recibir la información, validarla y responder a los participantes dentro de lapsos de tiempos aceptables. La solución adoptada debe permitir establecer canales de comunicación entre los organizadores del concurso y los participantes, con el fin de poderles informar de manera rápida si califican o no. Además, la solución debe permitir a los participantes aceptados, modificar la información registrada, si así fuese solicitado por los organizadores del concurso.

Consideraciones preliminares:

1. Cada participante puede realizar una sola entrada en el concurso (se desea evitar repeticiones).
2. Cada entrada del concurso debe tener obligatoriamente la información del participante (nombre, apellido, número de teléfono, y número de cédula), y la información de la mascota (nombre, edad aproximada en meses y años, una historia entretenida de la mascota de no más de 400 caracteres, y 3 fotos de la mascota), opcionalmente se debe presentar la opción de aclarar si la mascota es adoptada, y en caso de ser así, dónde fue hallada.
3. El administrador debe poder aprobar, rechazar y solicitar modificaciones, del formulario.
 - a. Las modificaciones solo pueden ser realizadas si el administrador lo solicita.
 - b. El administrador es el único que debe tener acceso a los formularios resueltos.

- c. Se debe tener la opción de filtrar las entradas por cada categoría. (aprobado, rechazado, modificado y pendiente).
- d. El administrador debe poder eliminar imágenes subidas.

4. se debe incluir un módulo de notificación por correo que notifique a la persona del estado de su entrada cuando este cambie, en caso de que se le pida modificación la persona deberá recibir un vínculo para acceder a su entrada y modificar los datos requeridos (incluidas las imágenes).

Requerimientos tecnológicos:

1. Firebase: Plataforma que ofrece base de datos NoSQL del tipo documento de esquema flexible, la cual funciona utilizando operaciones rest (Get, post, put, patch, delete). Adicionalmente ofrece servicios de alojamiento de archivos, hosting de sitios web, manejador de usuarios. A continuación algunos enlaces de interés sobre Firebase:

- a. <https://firebase.google.com/>
- b. Uso en html: <https://firebase.google.com/docs/web/setup>
- c. Hosting: <https://firebase.google.com/docs/hosting/>
- d. Almacenamiento: <https://firebase.google.com/docs/storage/web/start>
- e. Usuarios: <https://firebase.google.com/docs/auth/web/manage-users>
- f. Base de datos:
 - i. <https://firebase.google.com/docs/database/>
 - ii. Uso de la base de datos: <https://firebase.google.com/docs/database/web/start>
 - iii. Seguridad: <https://firebase.google.com/docs/reference/security/database/>

2. Google Material: Patrón de diseño de interfaz gráfica, que ofrece google. Enlaces de interés:

- a. <https://getmdl.io/>
- b. Principios: <https://material.io/guidelines/>
- c. Fuentes: <https://fonts.google.com/>
- d. Iconos: <https://material.io/icons/>

3. Github: herramienta de versionamiento de código.

Metodología de trabajo:

1. Los proyectos deberán llevarse a cabo en sus grupos del proyecto principal, se hará especial énfasis en la **revisión de interfaces**. Al unísono, las **copias de código** no serán aceptadas, siendo causal suficiente para rechazar la solución diseñada por parte del preparador. De igual forma, es condición necesaria y excluyente para que las soluciones diseñadas sean aceptadas y evaluadas, **que las mismas funcionen** (corran).
2. El día de entrega será el **6 de julio** en la hora de preparadurías, donde se hará una revisión preliminar y los equipos darán a los preparadores el acceso adecuado a las aplicaciones diseñadas (incluyendo instrucciones de instalación).
3. El proyecto debe realizarse utilizando únicamente la plataforma de Firebase para su desarrollo, uso de base de datos y deploy. **No se aceptarán entregas** que corran locales o en otras plataformas en línea.
4. Los proyectos tienen que estar respaldados en github y deberán tener uso de versionamiento visible.
5. Se requiere documentación de arquitectura del código.

Revisado por Christian Guillén Drija.