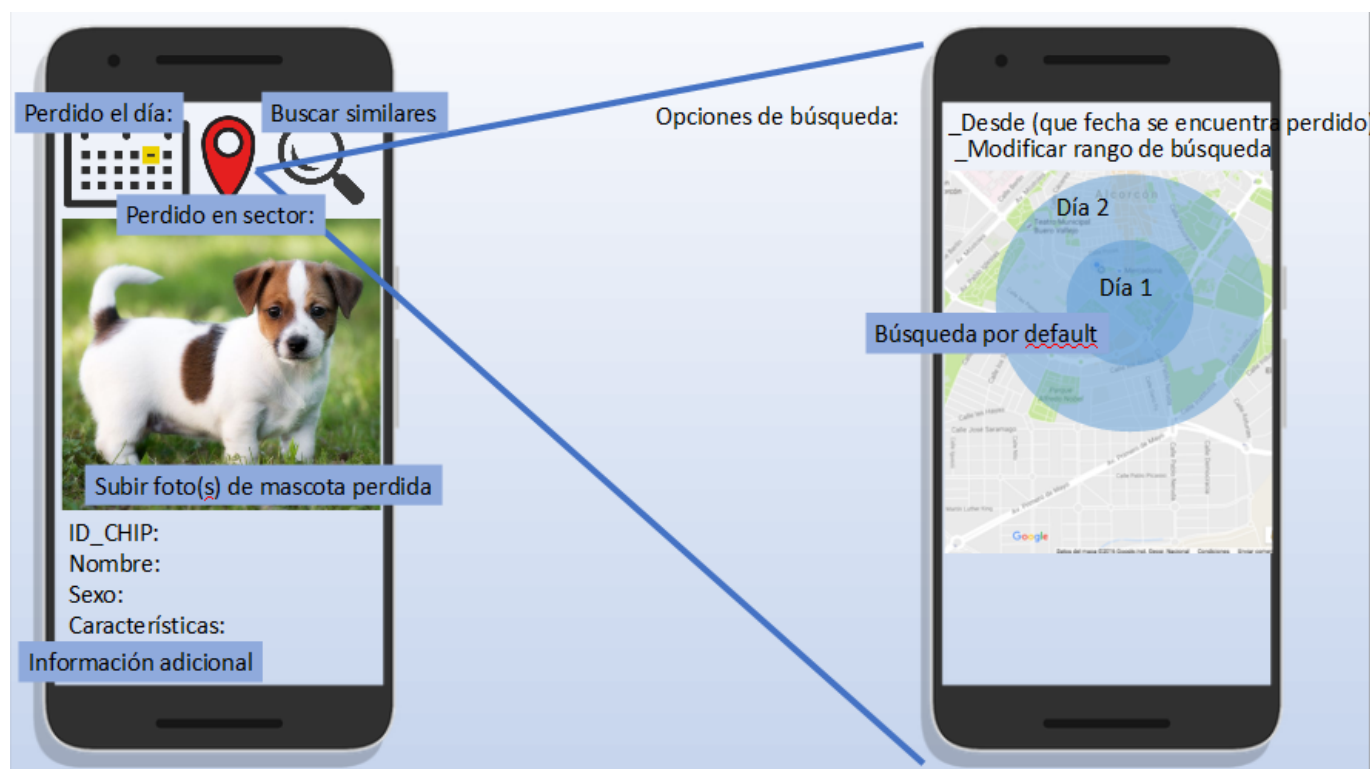


Proyecto Quilterrier

El avance a presentar el día de hoy corresponde a una aplicación llamada Quilterrier la cual tiene por objetivo, ser un centro de ayuda para la búsqueda de mascotas perdidas, adopción de animales y tenencia responsable de los mismos. Por esta razón se pretende incluir en la misma, las siguientes características (escalables y con el objetivo de incentivar la búsqueda de información y el desarrollo de nuevos conocimientos y habilidades entre los integrantes del grupo).





- Integrar a la app cosas para que sea de uso diario de dueño responsable, cosa de que no solamente se use cuando se haya extraviado su mascota
- Ofrecer incentivos para buscar mascotas perdidas, como sacos de comida, accesorios, descuentos en alguna veterinaria, etc.
- Sería ideal que el chip pudiera vincularse a la aplicación
- Activity con los extravíos más recientes por ubicación (cosa de que yo pueda activar mi ubicación y vea los extravíos acorde al sector)
- Alerta de robo
- Crear vínculo con datos ingresados para postear directamente en algún otro sitio (facebook, wsapp, Instagram, etc).

Dentro de estas funcionalidades, para esta primera entrega decidimos centrarnos en cosas nuevas no vistas en clases, por lo que se trabajó principalmete 5 areas:

1. Sistema de registro de cuentas, Login y **Firestore**.
2. Recycle View.
3. Incorporación de mapas a la aplicación.
4. Interfaz gráfica
5. Github

Estos puntos se detallan a continuación:

Login:

Para la aplicación se tiene pensado utilizar un sistema de Login (soportado en Google Firebase), con el fin de diferenciar entre usuarios que puedan solamente ver el contenido de la app y los que puedan ver y realizar comentarios o solicitudes de búsqueda o avisar sobre mascotas encontradas.

Así, con esto en mente, el primer paso fue la implementación inicial del sistema de Login como se presenta a continuación (imagenes ilustrativas no representan la apariencia final de la aplicación, son solo un modo de preestar visualmente el avance y el código de la misma):



Estas 4 activitys fueron codificadas y se dejaron en funcionamiento, permitiendo crear cuentas, loguearse con ellas, realizar restablecimiento de contraseña y desloguearse si así se necesita.

El código y el readme en detalle de esta puede encontrarse los siguientes enlaces:

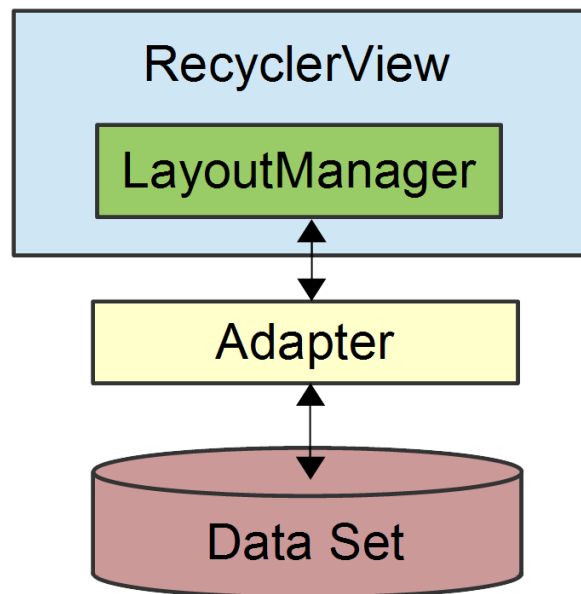
- [Código](#)
- [Readme en detalle](#)

Importante el señalar que a futuro se pretende incluir funcionalidades como login con Facebook, Twitter y Google, pero por ahora solo está disponible el login por correo y contraseña.

Recycle View:

Mediante esta parte de la app se busca mostrar en orden a las macotas que serán dadas en adopción y aquellas que estan perdidas, de forma que los usuario registrados puedan verlas y publicar (a futuro) los comentarios y avisar sobre mascotas encontradas. La forma en que esto se llevó a cabo se explica a continuación:

- **Main Activity:** Se dispuso realizar un landing que incluyera un RecylcerView con los últimas mascotas perdidas o en adopción.



Como podemos observar, se deben Implementar:

- DataSet
- Adapter
- RecyclerView

Data Set

Al ser un mock up, se implementó de la siguiente manera:

Agregamos algunos con los diferentes perros/gatos, su clasificación, su ubicación y si es perro o gato.

```
<resources>
  <string name="app_name">QuilTerrier</string>

  <string-array name="nombre_perro">
    <item>Jhon</item>
    <item>Gerald</item>
    <item>Goku</item>
    <item>Ruben</item>
    <item>Mara</item>
  </string-array>

  <string-array name="tipo">
    <item>Perdido!</item>
    <item>Adopción</item>
    <item>Perdido!</item>
    <item>Perdido!</item>
    <item>Adopción</item>
  </string-array>

  <string-array name="ubicacion">
    <item>Caldera</item>
    <item>Los Nogales</item>
```

```

        <item>Independencia</item>
        <item>Maipu</item>
        <item>Valparaíso</item>
    </string-array>
    <string-array name="especie">
        <item>Perro</item>
        <item>Perro</item>
        <item>Gato</item>
        <item>Perro</item>
        <item>Perro</item>
    </string-array>

</resources>

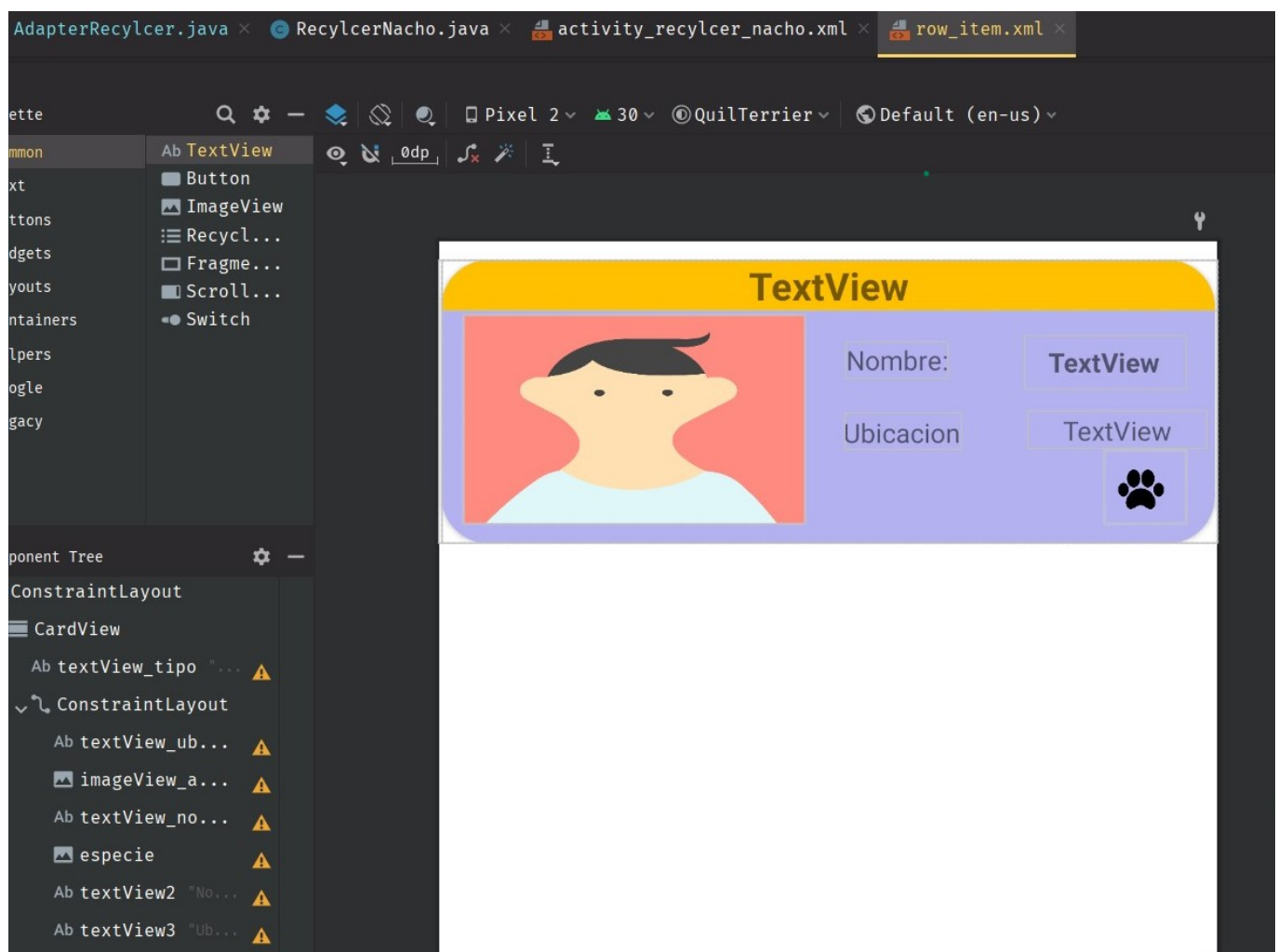
```

Adapter

Se creó una clase que extiende de RecyclerView.Adapter y necesita nuestro propio viewHolder, en este caso lo llamamos MyViewHolder. A este se le asocian los elementos de la UI, como botones, textview, etc.

Hay dos cosas importantes a destacar:

- Utilizamos el `LayoutInflater.from(parent.getContext()).inflate(R.layout.row_item, this, false)`. Este marca que se debe *inflar* el layout Row item, que solo tiene una Cardview, y esta será reproducida n veces siendo n la cantidad de datos.



- `onBindViewHolder(@NonNull MyViewHolder holder, int position)` implica que vamos a unir(bind), el holder(MyViewHolder) con la posición de la "lista". Ahí explícitamente le estamos informando que en función al holder, defina, texto, color, e incluso imagenes.

```
public class AdapterRecylcer extends
RecyclerView.Adapter<AdapterRecylcer.MyViewHolder> {
    String name[], type[], location[], especieAnimal[];
    int images[];
    Context context;

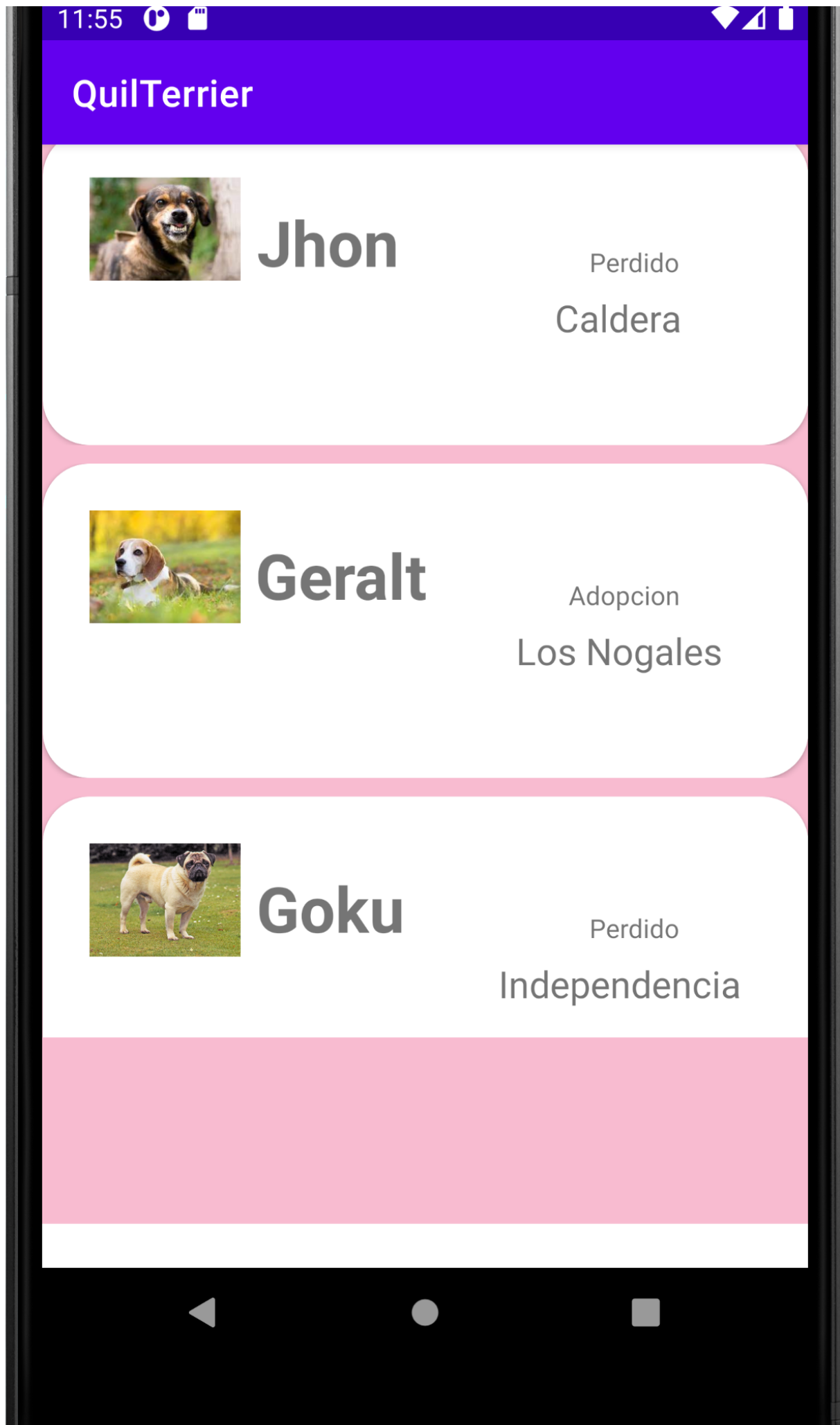
    public AdapterRecylcer(Context ctx, String nombre[], String tipo[], String
ubicacion[], String especieArr[],
                           int img[]) {
        context = ctx;
        name = nombre;
        type = tipo;
        location = ubicacion;
        images = img;
        especieAnimal = especieArr;
    }

    public class MyViewHolder extends RecyclerView.ViewHolder {
        TextView nombre, tipo, ubicacion;
        ImageView imageView_animal, especie;

        public MyViewHolder(@NonNull View itemView) {
            super(itemView);
            imageView_animal = itemView.findViewById(R.id.imageView_animal);
            nombre = itemView.findViewById(R.id.textView_nombre);
            tipo = itemView.findViewById(R.id.textView_tipo);
            ubicacion = itemView.findViewById(R.id.textView_ubicacion);
            especie = itemView.findViewById(R.id.especie);
        }
    }
}
```

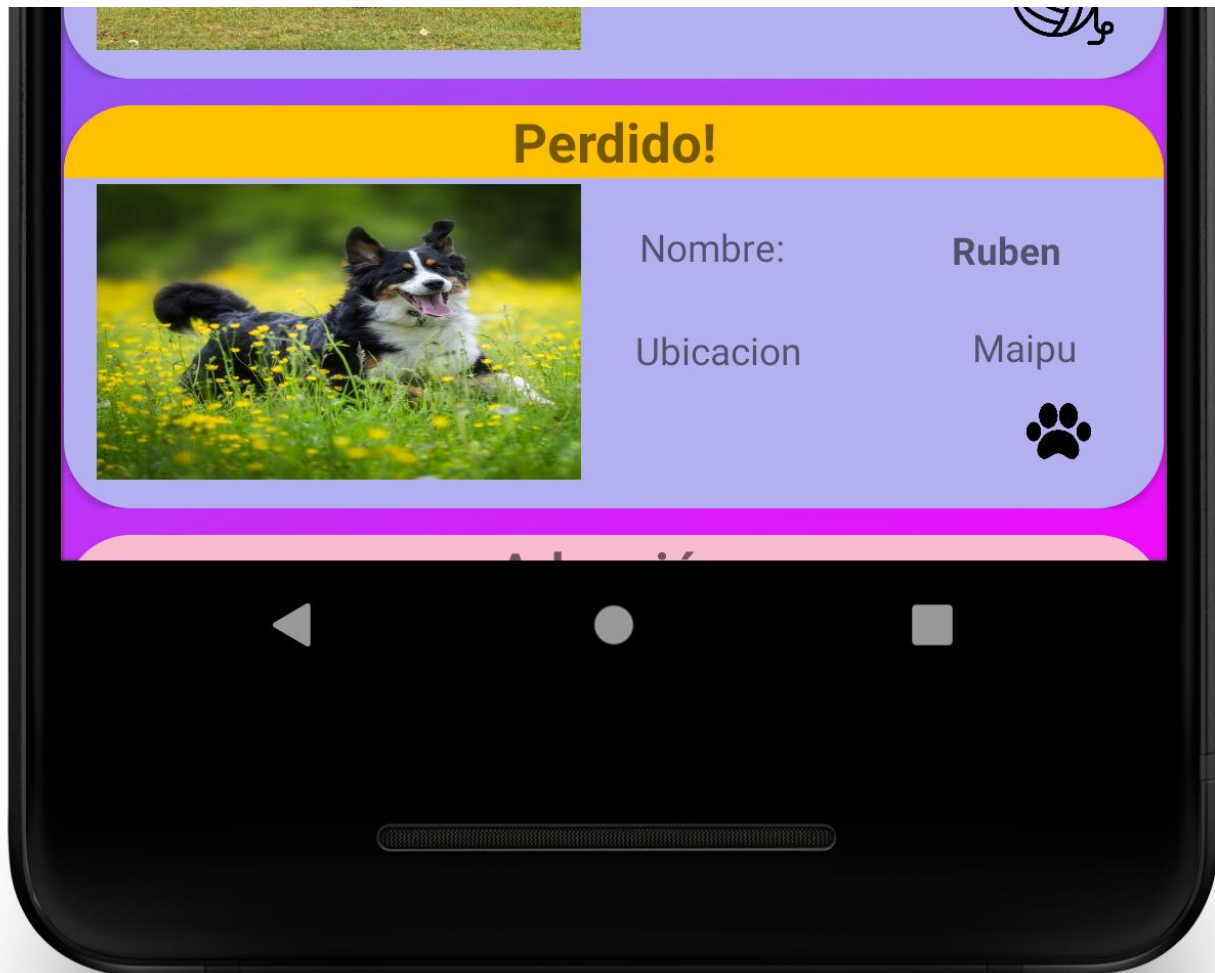
Esto traducido en las vistas de Android se expresa de la siguiente manera para el usuario:





Primeros intentos (así comenzo)





Version de prueba (así termino)

Incorporación de mapas a la aplicación:

En esta version de la aplicacion, se planeo implementar los mapas donde hubo extravio de las mascotas, por lo cual se estudio el tema con respecto a la adherencia de ubicación en la aplicación. Para esto si bien no esta del todo implementado esta parte en la presente entrega, si podemos decir que los mapas fueron utilizados pero solucionar el problema de la localización fina con el GPS

Interfaz gráfica:

Se trabajo en la interfaz que verá el usuario final de cara a la aplicación, en esta se busca que sea facil de utilizar, llamativa (reflejado en el color de los botones), amena (ya que se condireando la posible perdida de la mascota, que para gran parte de las personas es parte de la familia), para esto utilizamos una paleta de tonalidades suaves para darle a entender al usuario que será una herramienta de apoyo en esos tiempo dificiles. La tipografía, sencilla y clara (Quicksand light y Quicksand bold fueron las ocupadas), todo esto acompañado de imagenes simples que muestren el uso de la aplicación en pos de ayudar a las mascotas.

Github:

Para poder tener un uso eficiente de Github asi como un mayor entendimiento de la plataforma creamos un solo repositorio y de ese repositorio fuimos creando ramas en base al desarrollo de las tareas asignadas a cada participante del proyecto. Esto nos permitio mejor comprension al momento de unir las ramas dentro de la rama principal y así trabajar como un grupo profesional con la plataforma.

[Enlace a nuestro repositorio](#)

nrubioh / quilterrierApp

<> Code

Issues

Pull requests

Actions

Projects

Wiki

Security

Insights

Settings

main 7 branches 0 tags

Go to file

Add file

Code

Yayson Olivares added todo		0e0d15e 13 hours ago	32 commits
Markdown Login	Login con fondos y Markdown		15 hours ago
QuilTerrier_Java	added todo		13 hours ago
imagenes	merged data		14 hours ago
.gitignore	Initial commit		7 days ago
LICENSE	Initial commit		7 days ago
README.md	TODO		6 days ago
bosquejoEstructura.pptx	Add files via upload		7 days ago
diagrama casos de usos borrador.png	Diagrama casos de uso editable		6 days ago
proyecto app cachorros.pptx	ppt inicial		6 days ago