

TEMA. COMUNICACIÓN DE FRAGMENTS CON ACTIVITIES A TRAVÉS DE EVENTOS

1. Crear Interface propia para gestionar eventos propios del fragment.

```
public interface OnClickBoton {  
  
    public void pulsarBoton(String mensaje);  
}
```

2. Implementar el activity con ese Interface

1. `public class MainActivity extends AppCompatActivity implements OnClickBoton`

2. `public void pulsar(String texto){`

 `Toast.makeText(this,texto,Toast.LENGTH_LONG).show();`

 `}`

3. En el fragment creamos una referencia al interface, le llamamos por ejemplo listener.

```
OnClickBoton listener;
```

4. En el fragment, sobrescribimos el método `onAttach(Context context)` para referenciar el listener con dicho interface

```
@Override  
public void onAttach(Context context) {  
    super.onAttach(context);  
  
    try {  
        listener = (OnClickBoton) context;  
    } catch (ClassCastException e) {  
        throw new ClassCastException(context.toString() + " Se debe implementar  
        OnClickBotonDia");  
    }  
}
```

5. Dentro de `onCreateView`, llamaremos al listener con el método asociado cuando se genere el evento

```
View view=inflater.inflate(R.layout.fragment_modos_dia, container, false);
```

```
Button boton=view.findViewById(R.id.button);
```

```
boton.setOnClickListener(new View.OnClickListener() {  
    @Override  
    public void onClick(View v) {  
  
        listener.pulsar("hola");  
    }  
});
```