

MIS NOTAS ONLINE

Descripción

Esta ha sido mi primera App, y es un poco más sencilla que las demás. La he realizado un poco más por mi utilidad, ya que he guardado muchas veces mis contraseñas y mis notas en un dispositivo y luego las he perdido si me han quitado el móvil o por otras circunstancias, y con esto pretendía guardar mis notas en la nube, y así no perderlas nunca.

Se trata de una App donde puedes guardar tus notas online, modificarlas, borrarlas o verlas y que estén asociadas a una cuenta.

Clases

- C actualizarNota
- C addNote
- C borrarNota
- C editarNota
- C globalClass
- C LoginRequest
- C MainActivity
- C MenuPrincipal
- C MisNotas
- C Notas
- C RegisterRequest
- C Registrarse
- C Registro

Las clases creadas son las que vemos a la izquierda.

Algunas son para conectar y compartir datos entre la App y la BD, y otras Clases para crear los ArrayList de objetos, por ejemplo Comida que constaría de los siguientes parámetros:

```
public class Notas {  
    public Notas(int idNota, String contenido) {  
        this.idNota = idNota;  
        this.contenido = contenido;  
    }  
  
    public int getPertenece() { return pertenece; }  
    public void setPertenece(int pertenece) { this.pertenece = pertenece; }  
    public int getIdNota() { return idNota; }  
    public void setIdNota(int idNota) { this.idNota = idNota; }  
    public String getContenido() { return contenido; }  
    public void setContenido(String contenido) { this.contenido = contenido; }  
  
    private int pertenece;  
    private int idNota;  
    private String contenido;  
  
    @Override  
    public String toString() { return idNota + ": " + " " + contenido; }  
}
```

CLASE GLOBAL:

En mis aplicaciones, he usado una clase global para que puedan acceder a ella desde el resto de intent. En ella, he guardado algunos parámetros básicos como el ArrayList de Notas que se carga desde la base de datos, o mis datos para obtener dichas notas.

```
public class GlobalClass extends Application implements Serializable {
    public GlobalClass() {
        this.MisNotas = new ArrayList<>();
        this.idNota = 0;
        this.idUser = 0;
    }

    public int getIdUser() { return idUser; }

    public void setIdUser(int idUser) { this.idUser = idUser; }

    public int getIdNota() { return idNota; }

    public void setIdNota(int idNota) { this.idNota = idNota; }
    private int idUser;
    private int idNota;

    public int idNotaActual;
    public String contenidoActual;

    ArrayList<Notas> MisNotas;

    public void añadirNota(int IdUser, String contenido) {

        Notas n = new Notas(IdUser, contenido);
        MisNotas.add(n);

    }

}
```



LOGIN:

Para el Login simplemente es como las demás: dos EditText que obtendrá tus datos y los cotejara con la BD, y si son correctos cargara todos los demás en tu dispositivo.

Además de la opción de Registrarse, que serían varios EditText para guardar tus datos en la BD y darte de alta.



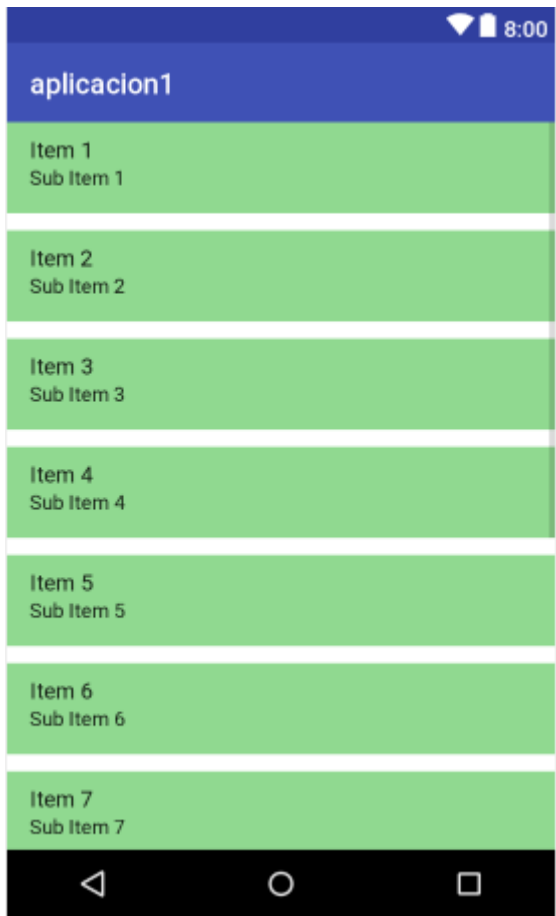
CLASE MENU PRINCIPAL:

Una vez hemos logueado y cargado los parámetros , nos permitirá escribir notas nuevas, o ver las antiguas notas.

Si hacemos click en guardar nota, nos la guardará en la nube en nuestra cuenta.

CLASE LISTA DE NOTAS:

Aquí simplemente saco el contenido de las notas por pantalla, y permito, al hacer click en ellas, llevarlas a una nueva actividad donde poder editarla, borrarla y guardarla.



CLASE EDITAR NOTA



La clase editar nota, es parecida a la de menú principal, sólo que con un botón más de editar nota, y si guardas la nota, la modifica en la base de datos y no crea una nueva.

CLASES DE CONEXIÓN CON BD

Como en otras aplicaciones, he añadido una clase para conectar con la base de datos y obtener o ingresar valores. Estas clases son por ejemplo como ésta :

```
public class AddNote extends StringRequest {
    private static final String REGISTER_REQUEST_URL="http://www.pgongorasoria.com/Notas/añadirNotas.php";
    private Map<String, String> params;

    public AddNote(int id, String contenido, Response.Listener<String> listener) {
        super(Request.Method.POST, REGISTER_REQUEST_URL, listener, errorListener: null);
        params= new HashMap<String, String>();
        params.put("id", id+"");
        params.put("contenido", contenido);
    }

    @Override
    public Map<String, String> getParams() { return params; }
}
```

En la que se especifica que valores vamos a pasar por post a un script php alojado en mi servidor.

```
public class RegisterRequest extends StringRequest {
    private static final String REGISTER_REQUEST_URL="http://www.pgongorasoria.com/Notas/Register.php";
    private Map<String, String> params;

    public RegisterRequest(String name, String password, Response.Listener<String> listener) {
        super(Method.POST, REGISTER_REQUEST_URL, listener, errorListener: null);
        params= new HashMap<String, String>();
        params.put("username", name);
        params.put("password", password);
    }

    @Override
    public Map<String, String> getParams() { return params; }
}
```