



UNIVERSIDAD CATÓLICA LOS ÁNGELES
CHIMBOTE

ESCUELA PROFESIONAL INGENIERIA DE SISTEMA
FACULTAD DE INGENIERIA DE SISTEMA



DOCENTE

HEBER GÓMEZ HURTADO

ALUMNA

YADIRA RODRIGUEZ GARCIA

CURSO

TECNOLOGIA MOVIL

AÑO

2018

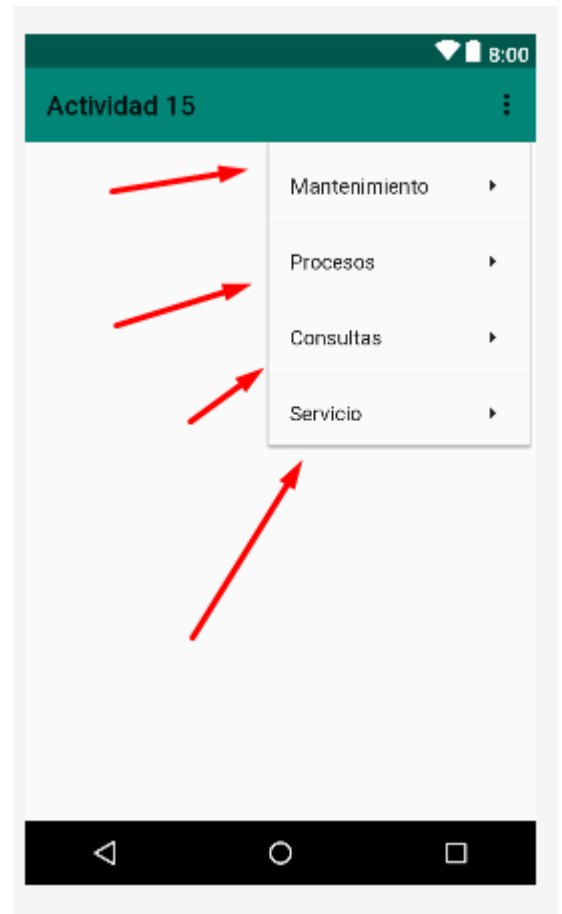
CHIMBOTE-PERU

Actividad N° 15

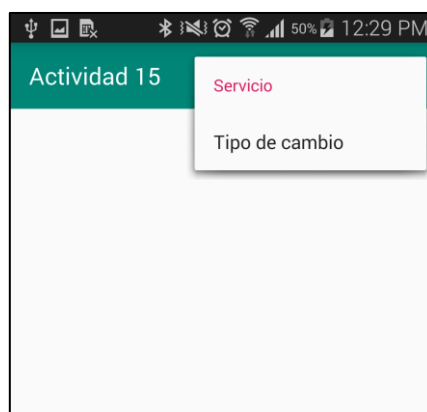
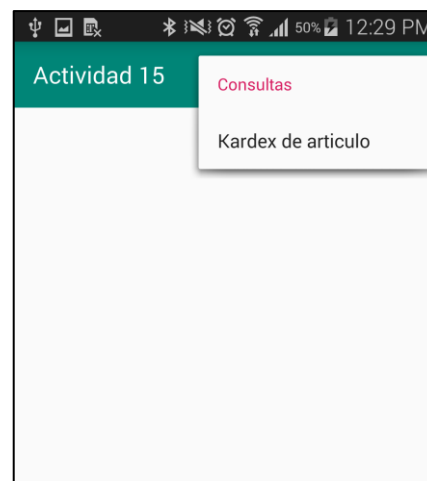
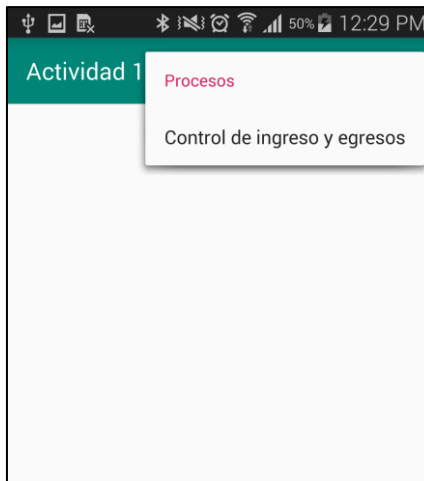
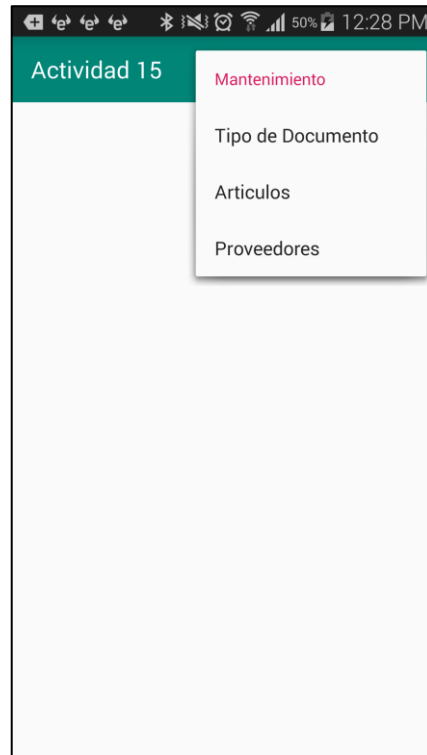
Menú

XLM Menu

```
<menu xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    tools:context="com.example.yadira.actividad15.MainActivity">
    <item
        android:id="@+id/action_settings"
        android:orderInCategory="100"
        android:title="Mantenimiento"
        app:showAsAction="never" >
        <menu>
            <item
                android:id="@+id/manteni_td"
                android:title="Tipo de Documento"
            ></item>
            <item
                android:id="@+id/manteni_ar"
                android:title="Articulos"
            ></item>
            <item
                android:id="@+id/manteni_pro"
                android:title="Proveedores"
            ></item>
        </menu>
    </item>
    <item
        android:id="@+id/action_procesos"
        android:orderInCategory="100"
        android:title="Procesos"
        app:showAsAction="never" >
        <menu>
            <item
                android:id="@+id/proceso_cont"
                android:title="Control de ingreso y egresos"
            ></item>
        </menu>
    </item>
    <item
        android:id="@+id/action_consultas"
        android:orderInCategory="100"
        android:title="Consultas"
        app:showAsAction="never" >
        <menu>
            <item
                android:id="@+id/consultas_kardex"
                android:title="Kardex de articulo"
            ></item>
        </menu>
    </item>
    <item
        android:id="@+id/action_servicio"
        android:orderInCategory="100"
        android:title="Servicio"
        app:showAsAction="never" >
        <menu>
            <item
                android:id="@+id/servicio_tc"
                android:title="Tipo de cambio"
            ></item>
        </menu>
    </item>
</menu>
```



Resultado del menú



Implementar las GUI necesarias que permitan mostrar, registrar, modificar y eliminar datos de la tabla **TipoDocumento**

Creación de la Base de Datos SQLite.

Clase **BaseDatosSQLiteHelper**, donde se maneja la conexión a la base de datos SQLite, y poder hacer CUD.

Código Java

```
public class BaseDatosSQLiteHelper extends SQLiteOpenHelper {

    final String CREAR_TABLA_ARTICULO = "CREATE TABLE Articulo (codigo TEXT PRIMARY KEY, descripcion TEXT, " +
        "cantidad INTEGER, precio REAL, unidadMedia TEXT, estado INTEGER DEFAULT 1)";
    final String CREAR_TABLA_PROVEEDOR = "CREATE TABLE Proveedor (codigo INTEGER PRIMARY KEY, nombre TEXT, " +
        "ruc TEXT, direccion TEXT, rubro TEXT)";
    final String CREAR_TABLA_TIPODOCUMENTO = "CREATE TABLE TipoDocumento (codigo TEXT PRIMARY KEY, descripcion TEXT)";
    final String CREAR_TABLA_MOVIMIENTO = "CREATE TABLE Movimiento (codigo INTEGER PRIMARY KEY, fecha TEXT, " +
        "tipo TEXT, cantidad INTEGER, tipoDocumento TEXT, documento TEXT, articulo " +
        "TEXT, FOREIGN KEY(tipoDocumento) REFERENCES TipoDocumento(codigo))";

    public BaseDatosSQLiteHelper(Context context, String name, SQLiteDatabase.CursorFactory factory, int version) {
        super(context, name, factory, version);
    }

    @Override
    public void onCreate(SQLiteDatabase db) {
        db.execSQL(CREAR_TABLA_ARTICULO);
        db.execSQL(CREAR_TABLA_PROVEEDOR);
        db.execSQL(CREAR_TABLA_TIPODOCUMENTO);
        db.execSQL(CREAR_TABLA_MOVIMIENTO);
    }

    @Override
    public void onUpgrade(SQLiteDatabase db, int oldVersion, int newVersion) {
        db.execSQL("DROP TABLE IF EXISTS Proveedor");
        db.execSQL("DROP TABLE IF EXISTS UnidadMedida");
        db.execSQL("DROP TABLE IF EXISTS Movimiento");
        db.execSQL("DROP TABLE IF EXISTS Articulo");

        db.execSQL(CREAR_TABLA_ARTICULO);
        db.execSQL(CREAR_TABLA_PROVEEDOR);
        db.execSQL(CREAR_TABLA_TIPODOCUMENTO);
        db.execSQL(CREAR_TABLA_MOVIMIENTO);
    }
}
```

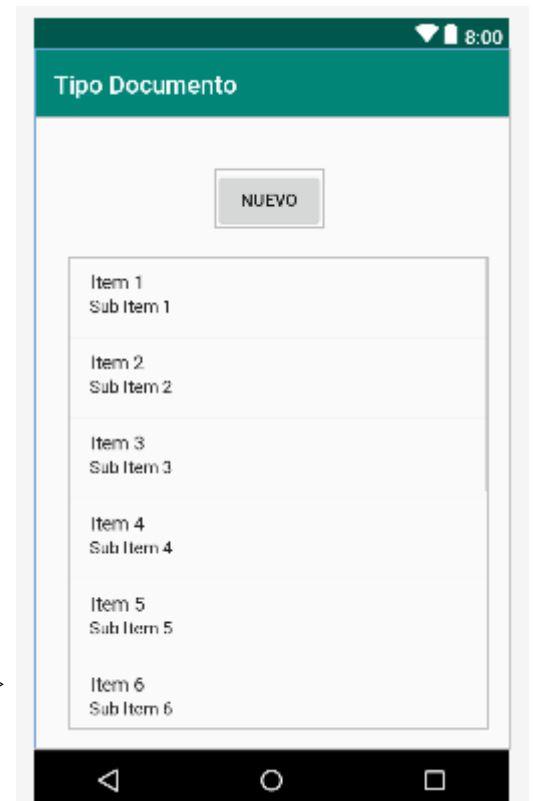
Mantenimiento a la tabla TipoDocumento

Diseño

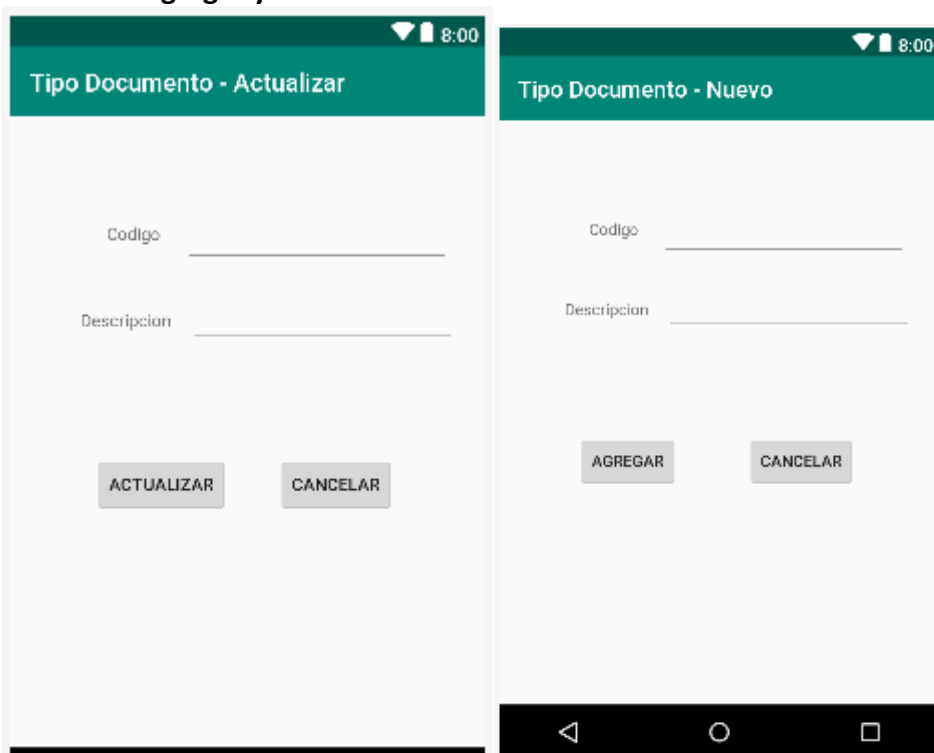
```
<AbsoluteLayout
    android:layout_width="match_parent"
    android:layout_height="match_parent">

    <ListView
        android:id="@+id/listaTipoDoc"
        android:layout_width="339dp"
        android:layout_height="382dp"
        android:layout_x="29dp"
        android:layout_y="169dp"
        tools:layout_editor_absoluteX="8dp"
        tools:layout_editor_absoluteY="8dp" />

    <Button
        android:id="@+id/button"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_x="146dp"
        android:layout_y="98dp"
        android:onClick="nuevoTipoDoc"
        android:text="Nuevo"
        tools:layout_editor_absoluteX="145dp"
        tools:layout_editor_absoluteY="85dp" />
</AbsoluteLayout>
```



Diseño de Agregar y editar



Código java

Vista principal

```
public class MantenimientoTipoDoc extends AppCompatActivity {
    ListView listView;
    ArrayList<String> lista=new ArrayList<>();
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_mantenimiento_tipo_doc);
        Toolbar toolbar = (Toolbar) findViewById(R.id.toolbar);
        setSupportActionBar(toolbar);
        listView=(ListView) findViewById(R.id.ListaTipoDoc);
        listView.setOnItemClickListener((parent, view, position, id) -> {
            seleccion(position);
        });
        cargarDatos();
    }
    private void seleccion(final int index){
        final String[] items = {"Editar", "Eliminar"};
        AlertDialog.Builder builder =
            new AlertDialog.Builder( context: this);
        builder.setTitle("Selección")
            .setItems(items, (dialog, item) -> {
                if (item==0){
                    Intent intent = new Intent( packageContext: MantenimientoTipoDoc.this,TipoDocActualizar.class);
                    String datos[]=lista.get(index).split( regex: " ");
                    intent.putExtra( name: "cod",datos[0]);
                    intent.putExtra( name: "descrip",datos[1]);
                    startActivity(intent);
                }else if (item==1){
                    eliminarRegistro(lista.get(index).split( regex: " ")[0]);
                }
            });
        builder.create();
        builder.show();
    }
    private void eliminarRegistro(String id){
        BaseDatosSQLiteHelper con = new BaseDatosSQLiteHelper( context: this, name: "DB_registros", factory: null, version: 1);
        SQLiteDatabase db = con.getReadableDatabase();
        db.execSQL("DELETE FROM TipoDocumento WHERE codigo='"+id+"'");
        db.close();
        cargarDatos();
    }
    public void nuevoTipoDoc(View view){
        Intent intent =new Intent( packageContext: this,TipoDocNuevo.class);
        startActivity(intent);
    }
    private void cargarDatos(){
        lista.clear();
        ArrayAdapter<String> adapter= new ArrayAdapter<>( context: this,android.R.layout.simple_list_item_1,lista);
        listView.setAdapter(adapter);
        BaseDatosSQLiteHelper con = new BaseDatosSQLiteHelper( context: this, name: "DB_registros", factory: null, version: 1);
        SQLiteDatabase db = con.getReadableDatabase();
        Cursor c=db.rawQuery( sql: "SELECT * FROM TipoDocumento", selectionArgs: null);
        if (c.moveToFirst()){
            do {
                String item= c.getString( 0)+" "+c.getString( 1);
                lista.add(item);
            }while(c.moveToNext());
        }
    }
    @Override
    protected void onResume() {
        super.onResume();
        cargarDatos();
    }
}
```

Código java de crear nuevo Tipo Documento

```
public class TipoDocNuevo extends AppCompatActivity {

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_tipo_doc_nuevo);
        Toolbar toolbar = (Toolbar) findViewById(R.id.toolbar);
        setSupportActionBar(toolbar);
    }

    public void nuevoTipoDocumento(View view){
        EditText editText=(EditText)findViewById(R.id.tipoDocCodigo);
        EditText editText2=(EditText)findViewById(R.id.tipoDocDescrip);
        BaseDatosSQLiteHelper con = new BaseDatosSQLiteHelper( context: this, name: "DB_registros", factory: null, version: 1);
        SQLiteDatabase db = con.getWritableDatabase();
        ContentValues values = new ContentValues();
        values.put("codigo",editText.getText().toString());
        values.put("descripcion",editText2.getText().toString());
        db.insert( table: "TipoDocumento", nullColumnHack: null, values);
        db.close();
        finish();
    }

    public void salir(View view) { finish(); }
}
```

Código java de actualizar Tipo Documento

```
public class TipoDocActualizar extends AppCompatActivity {
    EditText id;
    EditText txt2;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_tipo_doc_actualizar);
        Toolbar toolbar = (Toolbar) findViewById(R.id.toolbar);
        setSupportActionBar(toolbar);
        txt2=(EditText) findViewById(R.id.tipoDocDescripAC);
        id=(EditText) findViewById(R.id.tipoDocCodigoAC);
        Bundle bundle = getIntent().getExtras();
        id.setText(bundle.getString( key: "cod"));
        txt2.setText(bundle.getString( key: "descrip"));
    }

    public void actualizar(View view){

        BaseDatosSQLiteHelper con = new BaseDatosSQLiteHelper( context: this, name: "DB_registros", factory: null, version: 1);
        SQLiteDatabase db = con.getReadableDatabase();
        db.execSQL("UPDATE TipoDocumento SET descripcion='"+txt2.getText().toString()+"' WHERE codigo='"+id.getText().toString()");
        db.close();
        onBackPressed();
    }

    public void salir(View view) { finish(); }
}
```

Código Eliminar este ya está incluido en el código de la vista principal

```
private void eliminarRegistro(String id){
    BaseDatosSQLiteHelper con = new BaseDatosSQLiteHelper( context: this, name: "DB_registros", factory: null, version: 1);
    SQLiteDatabase db = con.getReadableDatabase();
    db.execSQL("DELETE FROM TipoDocumento WHERE codigo='"+id+"'");
    db.close();
    cargarDatos();
}
```

Resultados

Tipo Documento

NUEVO

01 Factura

02 Boleta

Tipo Documento - Nuevo

Codigo 01

Descripcion Factura

AGREGAR CANCELAR

Tipo Documento

NUEVO

01 Factura

03 Boleta

Tipo Documento

NUEVO

01 Facturas2

Selección

Editar

Eliminar

Tipo Documento - Actualizar

Codigo 01

Descripcion Facturas2

ACTUALIZAR CANCELAR

Tipo Documento

NUEVO

01 Facturas2

03 Boleta

Tipo Documento

NUEVO

01 Facturas2

Selección

Editar

Eliminar

Tipo Documento

NUEVO

03 Boleta