

Evidencia de desempeño: GA8-220501096-AA2-EV02 - APK (desarrollar módulos
móvil según requerimientos del proyecto)

Presentado por: Maria Ximena Avila Payares
Magangué, Bolívar

Instructor: Milton Iván Barbosa Gaona

Centro de la productividad del diseño y la productividad empresarial.

Girardot, Cundinamarca.

Análisis y Desarrollo de Software

Ficha: (27583347)

SENA





Tabla de contenido

Introducción.....	2
Objetivos.....	3
Especificación de requisitos funcionales.....	4
Funcionamiento de Android Studio y su SDK	5
Diagrama de clases.	7
Diagrama de paquetes.....	8
Diagrama de componentes.....	8
Donde se ubican los componentes.....	9
Inicio de sesión	9
Metodología de Desarrollo de Software: Scrum	14
Mapa de navegación de la aplicación.	17
Análisis del Código XML y Determinación de Librerías y Frameworks.....	17
Conclusión	19

Introducción

En el contexto del desarrollo de aplicaciones móviles, la creación de módulos efectivos es fundamental para garantizar que el software cumpla con los requisitos del sistema y ofrezca una experiencia de usuario óptima. Este proyecto se centra en el desarrollo de módulos para una aplicación Android, siguiendo un enfoque metódico y



estructurado que contempla desde la codificación hasta la documentación y pruebas de cada componente. La evidencia de producto GA8-220501096-AA2-EV02 se orienta a desarrollar habilidades prácticas en el uso de herramientas como Android Studio y su SDK, además de aplicar conceptos de diseño de software, asegurando que cada módulo esté alineado con los requisitos funcionales y no funcionales del sistema.

Objetivos

Objetivo General: Desarrollar módulos de software para dispositivos móviles en la plataforma Android, asegurando que cumplan con los requerimientos del sistema y estén estructurados de acuerdo con las mejores prácticas de desarrollo.

Objetivos Específicos:

Estudio de Requerimientos:

Analizar y comprender los requerimientos del sistema, asegurando que cada módulo se alinee con ellos.

Familiarización con Herramientas:

Dominar el uso de Android Studio y su SDK, facilitando el desarrollo y la implementación de módulos.


Diseño de Arquitectura:

Crear y analizar diagramas de clases, paquetes y componentes que orienten la estructura de la aplicación.

Metodología de Desarrollo:

Aplicar una metodología de desarrollo de software adecuada, siguiendo un enfoque ágil o iterativo.

Codificación Eficiente:



Codificar cada módulo utilizando un lenguaje adecuado, dividiendo el código en componentes reutilizables y siguiendo buenas prácticas.

Control de Versiones:

Implementar un repositorio de control de versiones (GIT, SVN, etc.) para gestionar cambios y colaboraciones efectivamente.

Uso de Librerías y Frameworks:

Identificar y utilizar librerías y frameworks necesarios para la capa de presentación, optimizando el desarrollo.

Pruebas Unitarias:

Realizar pruebas unitarias para cada módulo, asegurando la funcionalidad y estabilidad del código.

Documentación:

Documentar adecuadamente los ambientes de desarrollo y pruebas, facilitando el entendimiento y mantenimiento del sistema.


Especificación de requisitos funcionales

El software SOF-INTCOR permitirá realizar las siguientes funciones:

1. Registro de clientes: los clientes o beneficiarios podrán registrarse ingresando; (nombre de iglesia, dirección, número de teléfono, correo).

2. Ingreso de datos: El cliente o usuario ingresara allí la información contable concerniente.

3. Gestión de administrador: El administrador del sistema gestionará los roles de usuarios del sistema, de igual forma aceptará y solicitará información concerniente con la plataforma. (datos personales, contabilidad, etc.).



4. Gestión de clientes: El cliente o beneficiario hará las siguientes gestiones mediante la configuración ingresar contabilidad, enviar y recibir mensajes al administrador o solicitante de la información. 8 de marzo de 2024 Conceptos básicos programación orientada a objetos.

5. Firma digital: El solicitante y cliente firmaran la información contable mediante códigos tanto la recibida como la enviada.

6. Almacenamiento de información: Proceso que permitirá a una persona aceptar la contabilidad ingresada por un usuario o cliente para un cálculo total.

7. Actualización de datos: Proceso por el cual cada usuario podrá actualizar sus datos tanto personales como contables dentro de su perfil o base de datos que le corresponde según el usuario, (cliente, solicitante o administrador).

Funcionamiento de Android Studio y su SDK

Android Studio es el entorno de desarrollo integrado (IDE) oficial para el desarrollo de aplicaciones en la plataforma Android. Proporciona herramientas completas para crear, probar y depurar aplicaciones, optimizando el flujo de trabajo de los desarrolladores. El **SDK de Android** (Software Development Kit) es un conjunto de herramientas y bibliotecas que permite a los desarrolladores crear aplicaciones que funcionen en dispositivos Android.

Componentes Clave de Android Studio

1. Editor de Código:

- Ofrece características como autocompletado, resaltado de sintaxis y refactorización, que facilitan la escritura de código eficiente.



2. Interfaz de Diseño:

- Permite diseñar interfaces de usuario de forma visual, utilizando el sistema de diseño de Android. Se puede alternar entre el modo de diseño y el modo de texto XML.

3. Gestor de Proyecto:

- Facilita la organización de archivos y recursos, permitiendo una estructura de proyecto clara y fácil de navegar.

4. Herramientas de Pruebas:

- Incluye emuladores y herramientas para realizar pruebas en diferentes versiones de Android y dispositivos.

5. Control de Versiones:

- Integración con sistemas de control de versiones como Git, lo que permite gestionar cambios en el código de manera efectiva.

6. Depurador:

- Herramientas que permiten identificar y solucionar problemas en el código, facilitando el seguimiento de errores y la optimización del rendimiento.

SDK de Android

El SDK de Android proporciona:

1. API de Android:

- Un conjunto de bibliotecas que permiten acceder a funcionalidades del sistema operativo Android, como cámaras, sensores, almacenamiento y más.

2. Herramientas de Construcción:

- Incluye herramientas para compilar y empaquetar aplicaciones, asegurando que sean compatibles con diferentes versiones de Android.

3. Documentación y Ejemplos:

- Recursos que ayudan a los desarrolladores a entender cómo utilizar las distintas APIs y funcionalidades.

4. Emulador:

- Permite simular un dispositivo Android en una computadora para probar aplicaciones sin necesidad de un dispositivo físico.

Diagrama de clases.

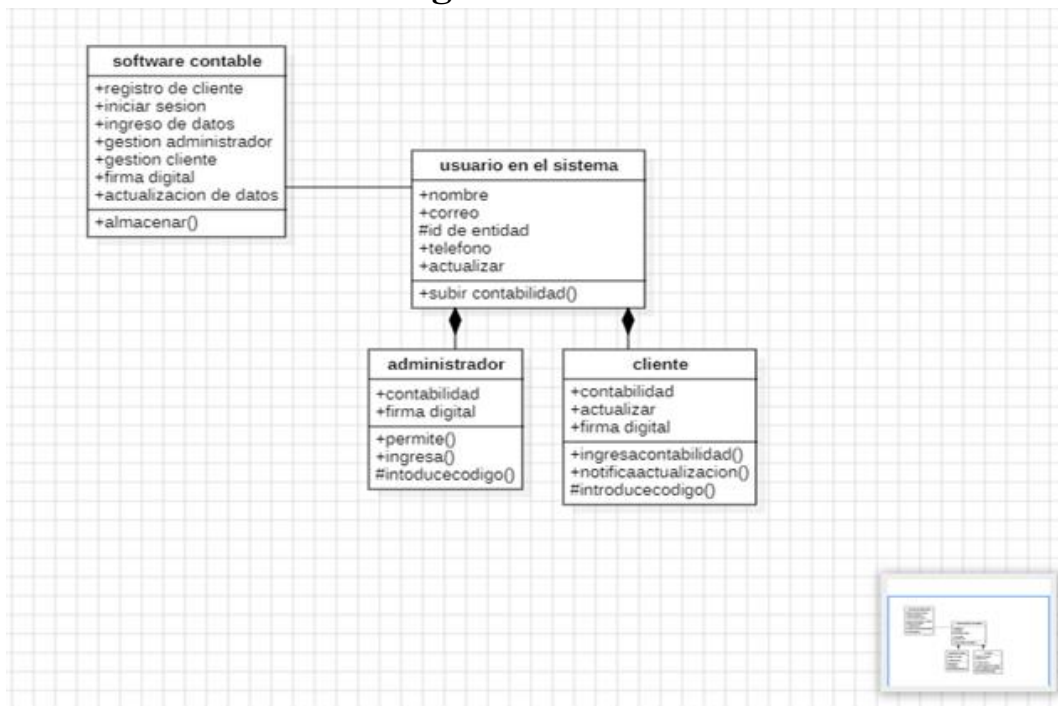


Diagrama de paquetes.

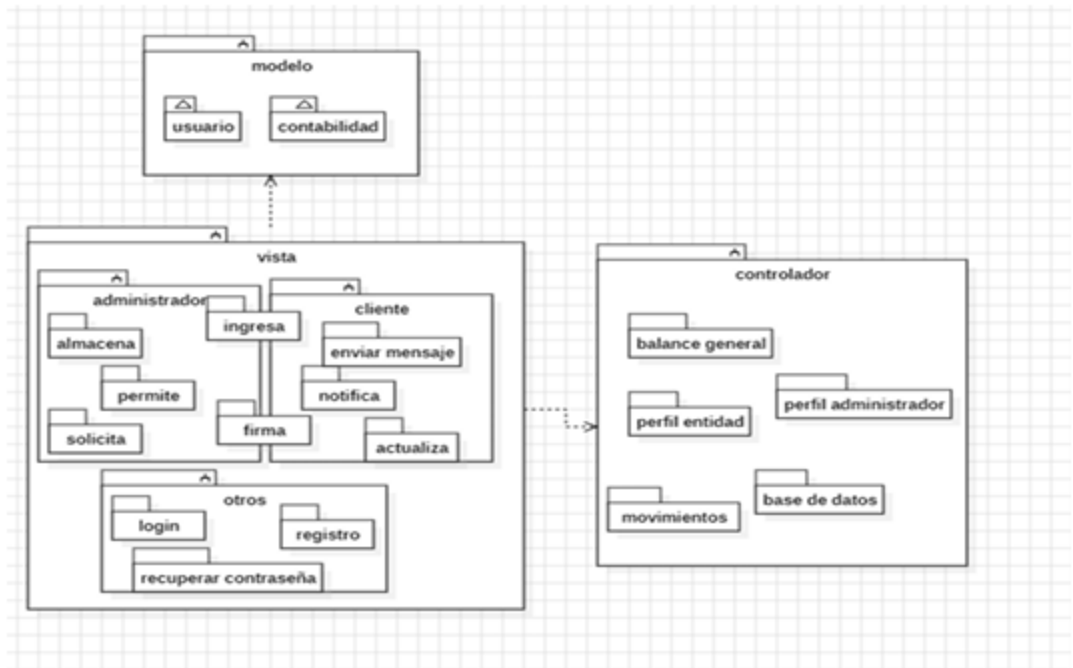
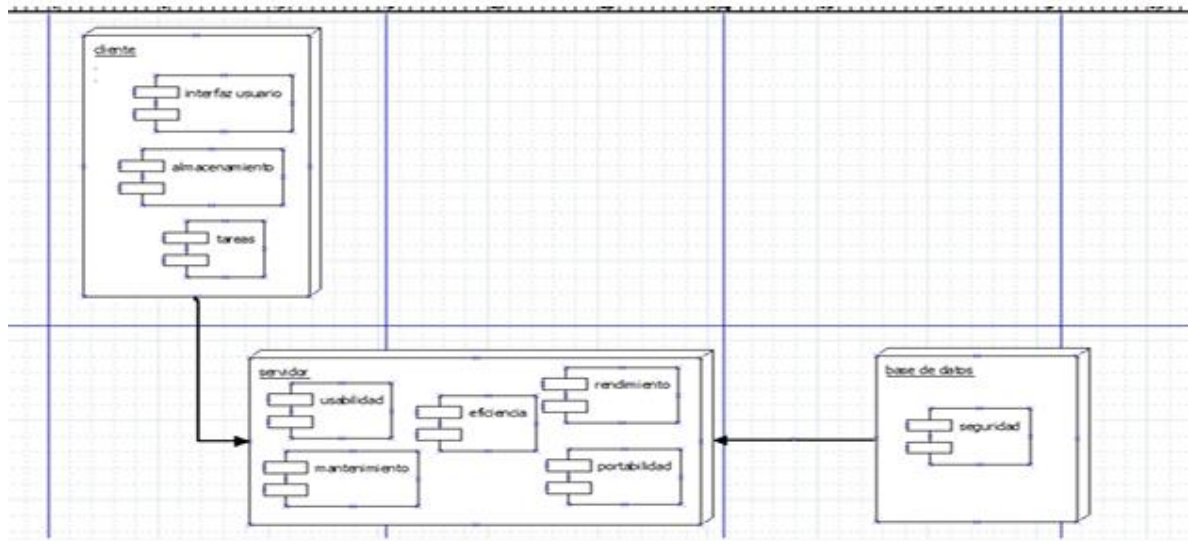
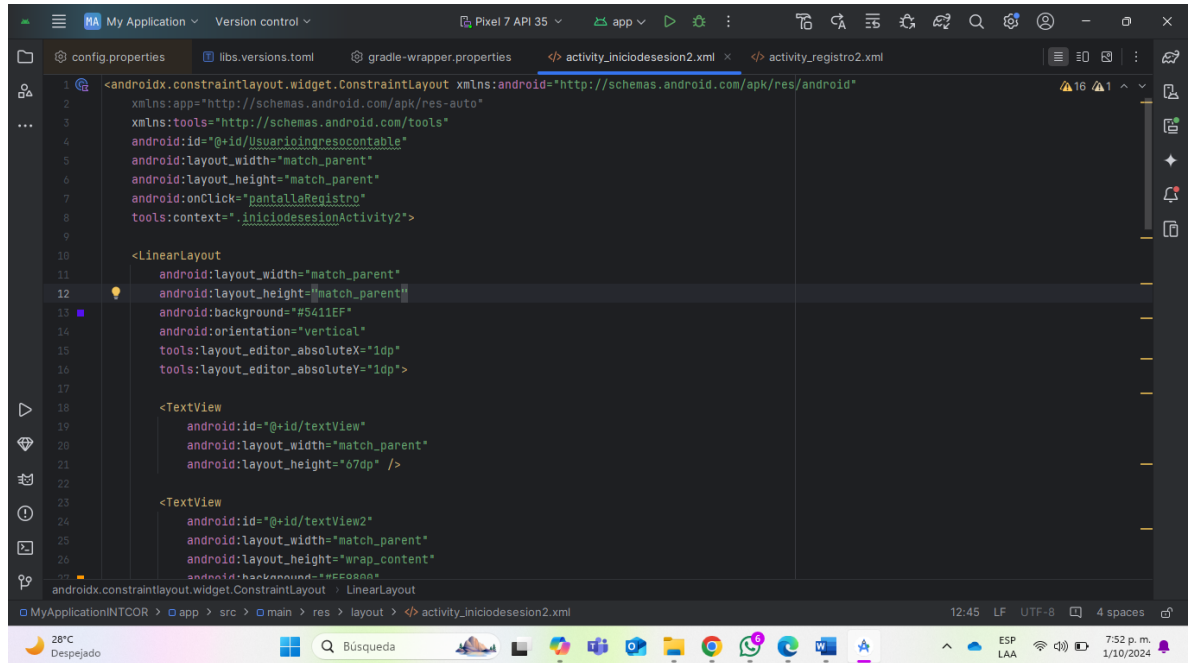


Diagrama de componentes.

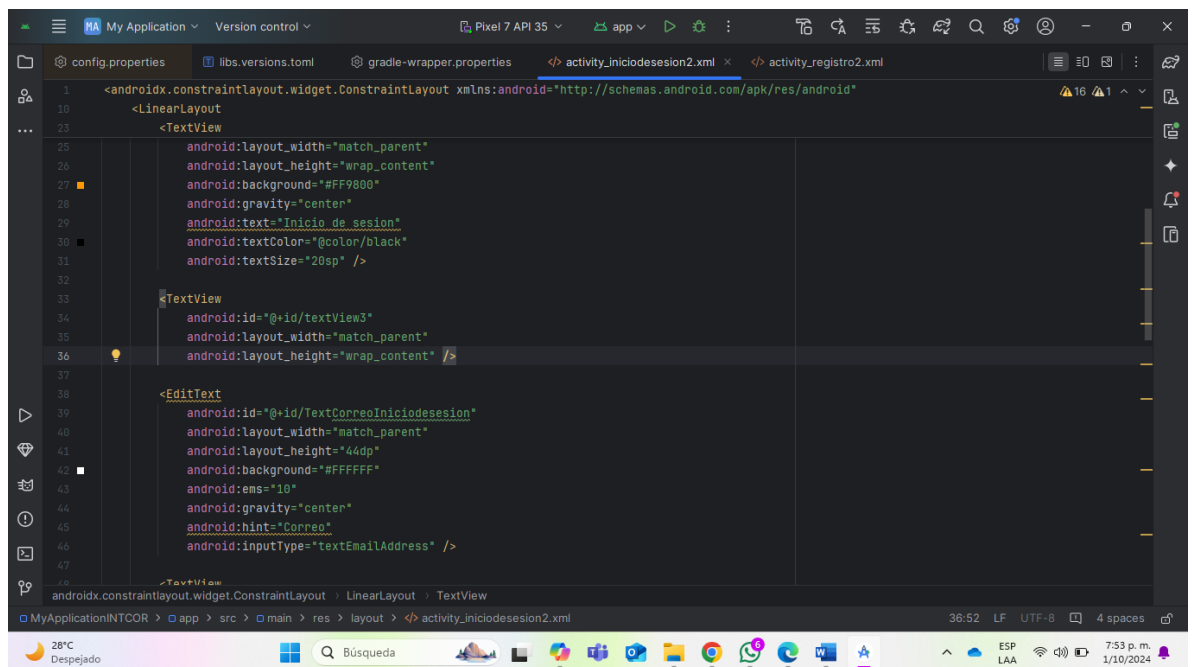


Donde se ubican los componentes.

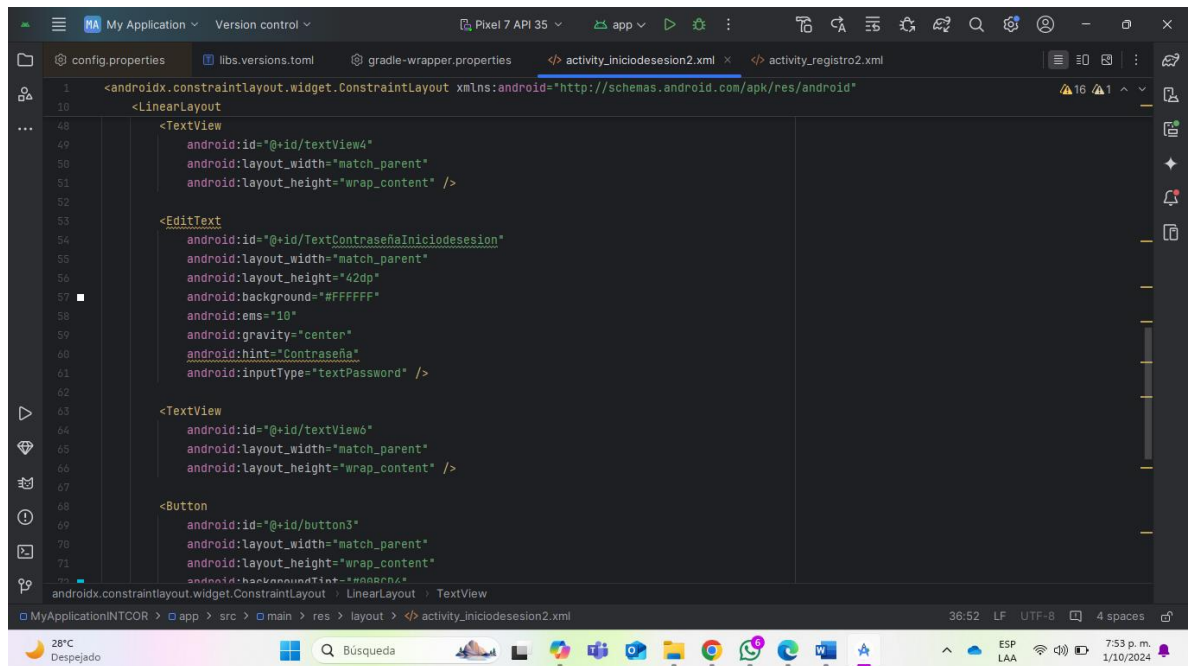
Inicio de sesión



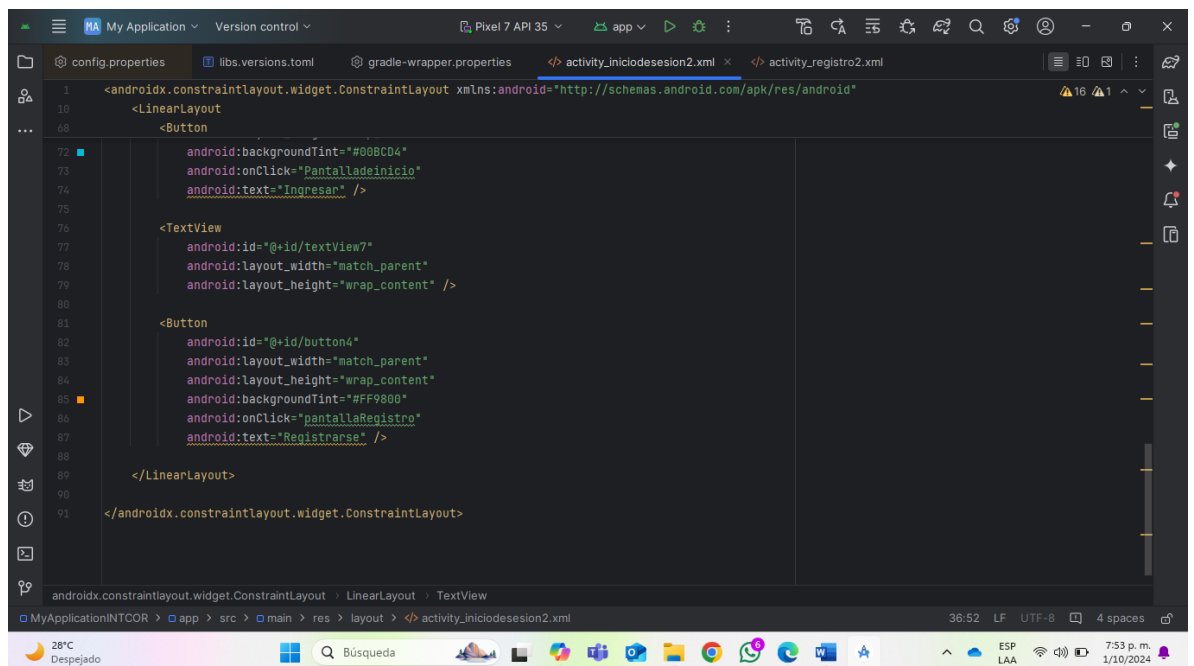
```
1 <androidx.constraintlayout.widget.ConstraintLayout xmlns:android="http://schemas.android.com/apk/res/android"
2     xmlns:app="http://schemas.android.com/apk/res-auto"
3     xmlns:tools="http://schemas.android.com/tools"
4     android:id="@+id/UsuarioIngresoContable"
5     android:layout_width="match_parent"
6     android:layout_height="match_parent"
7     android:onClick="pantallaRegistro"
8     tools:context=".IniciodesesionActivity2">
9
10
11     <LinearLayout
12         android:layout_width="match_parent"
13         android:layout_height="match_parent"
14         android:background="#5411EF"
15         android:orientation="vertical"
16         tools:layout_editor_absoluteX="1dp"
17         tools:layout_editor_absoluteY="1dp">
18
19         <TextView
20             android:id="@+id/textView"
21             android:layout_width="match_parent"
22             android:layout_height="67dp" />
23
24         <TextView
25             android:id="@+id/textView2"
26             android:layout_width="match_parent"
27             android:layout_height="wrap_content"
28             android:background="#E8E8E8"
29             android:text="Inicio de sesión"
30             android:textColor="@color/black"
31             android:textSize="20sp" />
32
33         <TextView
34             android:id="@+id/textView3"
35             android:layout_width="match_parent"
36             android:layout_height="wrap_content" />
37
38         <EditText
39             android:id="@+id/TextCorreoIniciodesesion"
40             android:layout_width="match_parent"
41             android:layout_height="44dp"
42             android:background="#FFFFFF"
43             android:ems="10"
44             android:gravity="center"
45             android:hint="Correo"
46             android:inputType="textEmailAddress" />
47
48     </LinearLayout>
49 </androidx.constraintlayout.widget.ConstraintLayout>
```



```
1 <androidx.constraintlayout.widget.ConstraintLayout xmlns:android="http://schemas.android.com/apk/res/android"
2     xmlns:app="http://schemas.android.com/apk/res-auto"
3     xmlns:tools="http://schemas.android.com/tools"
4     android:id="@+id/UsuarioIngresoContable"
5     android:layout_width="match_parent"
6     android:layout_height="match_parent"
7     android:onClick="pantallaRegistro"
8     tools:context=".IniciodesesionActivity2">
9
10
11     <LinearLayout
12         android:layout_width="match_parent"
13         android:layout_height="match_parent"
14         android:background="#FF9800"
15         android:orientation="vertical"
16         tools:layout_editor_absoluteX="1dp"
17         tools:layout_editor_absoluteY="1dp">
18
19         <TextView
20             android:id="@+id/textView"
21             android:layout_width="match_parent"
22             android:layout_height="67dp" />
23
24         <TextView
25             android:id="@+id/textView2"
26             android:layout_width="match_parent"
27             android:layout_height="wrap_content"
28             android:background="#E8E8E8"
29             android:text="Inicio de sesión"
30             android:textColor="@color/black"
31             android:textSize="20sp" />
32
33         <TextView
34             android:id="@+id/textView3"
35             android:layout_width="match_parent"
36             android:layout_height="wrap_content" />
37
38         <EditText
39             android:id="@+id/TextCorreoIniciodesesion"
40             android:layout_width="match_parent"
41             android:layout_height="44dp"
42             android:background="#FFFFFF"
43             android:ems="10"
44             android:gravity="center"
45             android:hint="Correo"
46             android:inputType="textEmailAddress" />
47
48     </LinearLayout>
49 </androidx.constraintlayout.widget.ConstraintLayout>
```



```
1 <androidx.constraintlayout.widget.ConstraintLayout xmlns:android="http://schemas.android.com/apk/res/android"
10 <LinearLayout
11     <TextView
12         android:id="@+id/textView4"
13         android:layout_width="match_parent"
14         android:layout_height="wrap_content" />
15
16     <EditText
17         android:id="@+id/TextContraseñaIniciodesesion"
18         android:layout_width="match_parent"
19         android:layout_height="42dp"
20         android:background="#FFFFFF"
21         android:ems="10"
22         android:gravity="center"
23         android:hint="Contraseña"
24         android:inputType="textPassword" />
25
26     <TextView
27         android:id="@+id/textView6"
28         android:layout_width="match_parent"
29         android:layout_height="wrap_content" />
30
31     <Button
32         android:id="@+id/button3"
33         android:layout_width="match_parent"
34         android:layout_height="wrap_content"
35         android:backgroundTint="#000000"
36         android:text="Ingresar" />
37 />
38 />
```



```
1 <androidx.constraintlayout.widget.ConstraintLayout xmlns:android="http://schemas.android.com/apk/res/android"
10 <LinearLayout
11     <Button
12         android:backgroundTint="#008BCD4"
13         android:onClick="PantalladeInicio"
14         android:text="Ingresar" />
15
16     <TextView
17         android:id="@+id/textView7"
18         android:layout_width="match_parent"
19         android:layout_height="wrap_content" />
20
21     <Button
22         android:id="@+id/button4"
23         android:layout_width="match_parent"
24         android:layout_height="wrap_content"
25         android:backgroundTint="#FF9800"
26         android:onClick="pantallaRegistro"
27         android:text="Registrarse" />
28 />
29 />
```



Inicio de sesion

Correo

Contraseña

Ingresar

Registrarse

Registro



Registro

Nombre de entidad

NIT

Fecha de vigencia

Direccion

Nombre de usuario

Correo

Contraseña

Confirmar contraseña

Continuar

My Application Version control Pixel 7 API 35 app

Gradle project sync failed. Basic functionality (e.g. editing, debugging) will not work properly. Try Again Open 'Build' View Show Log in Explorer

```

1 <?xml version="1.0" encoding="utf-8"?>
2 <androidx.constraintlayout.widget.ConstraintLayout xmlns:android="http://schemas.android.com/apk/res/android"
3     xmlns:app="http://schemas.android.com/apk/res-auto"
4     xmlns:tools="http://schemas.android.com/tools"
5     android:id="@+id/UsuarioIngresoConatable"
6     android:layout_width="match_parent"
7     android:layout_height="match_parent"
8     android:layout_gravity="center"
9     android:background="#3888E6"
10    tools:context=".RegistroActivity2">
11
12    <TextView
13        android:id="@+id/textRegistro"
14        android:layout_width="346dp"
15        android:layout_height="25dp"
16        android:background="#FF9800"
17        android:gravity="center"
18        android:text="Registro"
19        android:textColor="@color/black"
20        android:textSize="20sp"
21        app:layout_constraintBottom_toBottomOf="parent"
22        app:layout_constraintEnd_toEndOf="parent"
23        app:layout_constraintHorizontal_bias="0.498"
24        app:layout_constraintStart_toStartOf="parent"
25        app:layout_constraintTop_toTopOf="parent"

```

MyApplicationINTCOR > app > src > main > res > layout > activity_registro2.xml

27°C Despejado Búsqueda 8:02 p.m. 1/10/2024

```
<?xml version="1.0" encoding="utf-8" ?>
<androidx.constraintlayout.widget.ConstraintLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_constraintVertical_bias="0.12" />

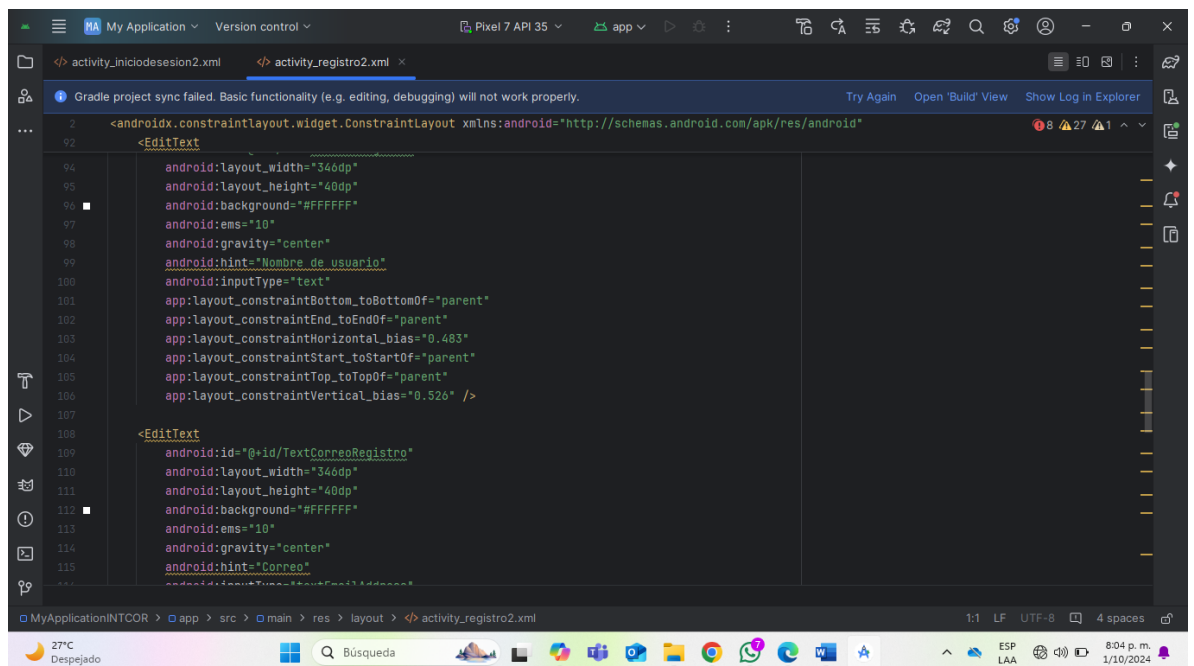
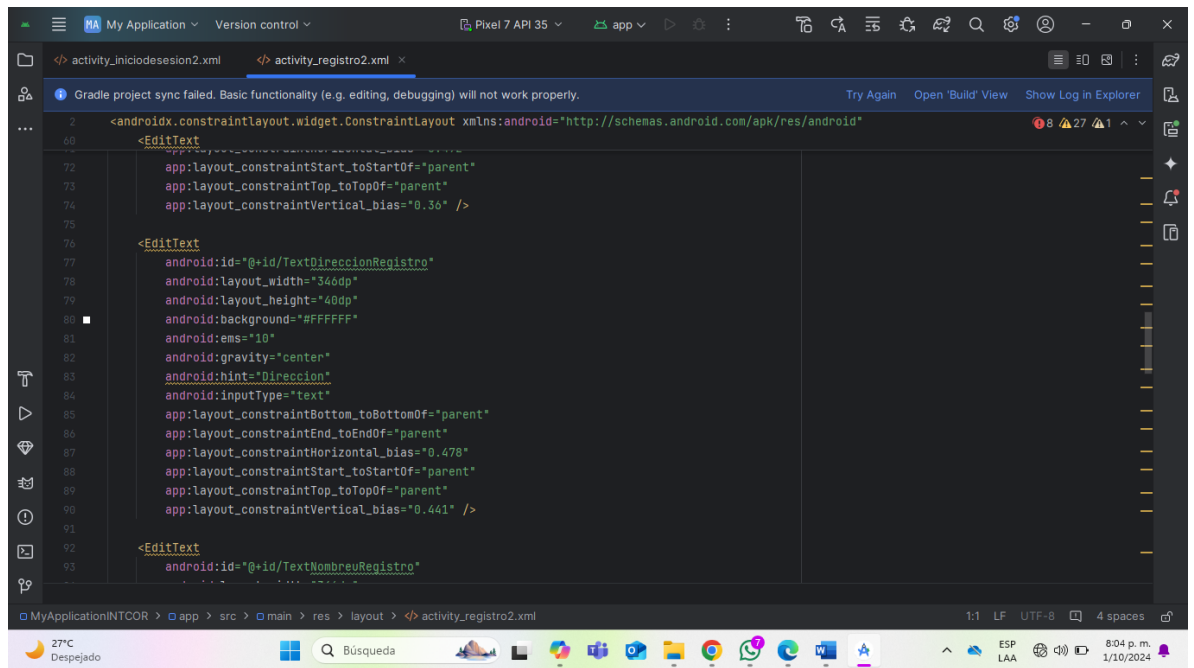
    <EditText
        android:id="@+id/TextNITRegistro"
        android:layout_width="340dp"
        android:layout_height="40dp"
        android:background="#FFFFFF"
        android:ems="10"
        android:gravity="center"
        android:hint="NIT"
        android:inputType="text"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.497"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent"
        app:layout_constraintVertical_bias="0.271" />


    <EditText
        android:id="@+id/TextNombreRegistro"
        android:layout_width="340dp"
        android:layout_height="40dp"
        android:background="#FFFFFF"
        android:ems="10"
        android:gravity="center"
        android:hint="Nombre Registro"
        android:inputType="text"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.497"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent"
        app:layout_constraintVertical_bias="0.271" />
```

```
<?xml version="1.0" encoding="utf-8" ?>
<androidx.constraintlayout.widget.ConstraintLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_constraintVertical_bias="0.12" />

    <EditText
        android:id="@+id/TextNITRegistro"
        android:layout_width="340dp"
        android:layout_height="40dp"
        android:background="#FFFFFF"
        android:ems="10"
        android:gravity="center"
        android:hint="NIT"
        android:inputType="text"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.497"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent"
        app:layout_constraintVertical_bias="0.271" />

    <EditText
        android:id="@+id/TextFechaVigenciaRegistro"
        android:layout_width="340dp"
        android:layout_height="40dp"
        android:background="#FFFFFF"
        android:ems="10"
        android:gravity="center"
        android:hint="Fecha de vigencia"
        android:inputType="date"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.492"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent"
        app:layout_constraintVertical_bias="0.271" />
```





adaptabilidad y la mejora continua, permitiendo a los equipos responder rápidamente a los cambios en los requisitos y en el entorno del proyecto.

Componentes Clave de Scrum

1. Roles:


- **Scrum Master:** Facilita el proceso Scrum, elimina impedimentos y ayuda al equipo a seguir las prácticas de Scrum.
- **Product Owner:** Representa los intereses de los stakeholders, prioriza el backlog del producto y asegura que el equipo esté trabajando en las tareas más importantes.
- **Equipo de Desarrollo:** Grupo multidisciplinario responsable de crear el producto. Se autoorganiza y trabaja de manera colaborativa.

2. Artefactos:

- **Product Backlog:** Lista priorizada de funcionalidades, mejoras y correcciones que se requieren en el producto. Es gestionada por el Product Owner.
- **Sprint Backlog:** Conjunto de tareas seleccionadas del Product Backlog que el equipo se compromete a completar en un sprint específico.
- **Incremento:** La suma de todos los elementos del Product Backlog completados durante un sprint y los anteriores, representando un producto potencialmente entregable.

3. Eventos:

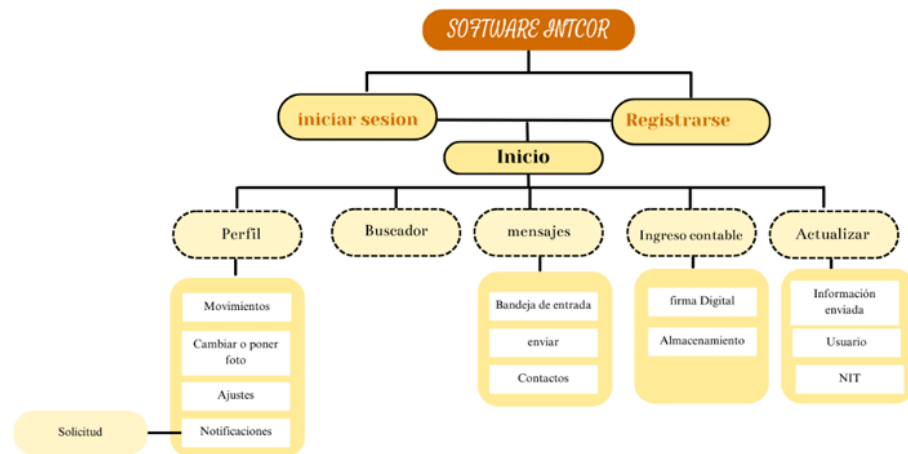
- **Sprint:** Ciclo de trabajo de duración fija (generalmente de 1 a 4 semanas) en el que se desarrolla un incremento del producto.

- 
- **Planificación del Sprint:** Reunión al inicio de cada sprint donde el equipo determina qué se va a trabajar y establece un objetivo de sprint.
 - **Daily Scrum:** Reuniones cortas (15 minutos) diarias donde el equipo revisa el progreso, planifica el trabajo del día y aborda impedimentos.
 - **Revisión del Sprint:** Reunión al final de cada sprint para demostrar el trabajo completado y recibir retroalimentación de los stakeholders.
 - **Retrospectiva del Sprint:** Reunión para reflexionar sobre el sprint pasado, identificando qué funcionó bien y qué se puede mejorar.

Beneficios de Scrum

- **Flexibilidad y Adaptabilidad:** Permite responder rápidamente a cambios en los requisitos y prioridades.
- **Mejora Continua:** Fomenta la reflexión y la adaptación, lo que lleva a un proceso de mejora continua.
- **Colaboración y Comunicación:** Promueve la comunicación constante entre los miembros del equipo y los stakeholders.
- **Entrega Incremental:** Facilita la entrega de versiones funcionales del producto a intervalos regulares, lo que proporciona valor temprano al cliente.

Mapa de navegación de la aplicación.



Análisis del Código XML y Determinación de Librerías y Frameworks

El código XML proporcionado define una interfaz de usuario para una pantalla de inicio de sesión en una aplicación Android. A continuación, se responde a las preguntas sobre las librerías necesarias y los frameworks a utilizar para la capa de presentación.

1. Librerías Necesarias para la Capa de Presentación

Para la implementación de la interfaz de usuario y las funcionalidades asociadas, se pueden considerar las siguientes librerías:

- **AndroidX:** La biblioteca androidx ya se está utilizando, lo cual es recomendable para garantizar compatibilidad y acceso a nuevas características. Asegúrate de tener las dependencias adecuadas en tu archivo build.gradle:

groovy

implementation 'androidx.appcompat:appcompat:1.4.0'



```
implementation 'androidx.constraintlayout:constraintlayout:2.1.3'
```

- **Material Components:** Si deseas mejorar la estética de los botones y otros elementos de la interfaz, puedes considerar usar las librerías de Material Design:

```
groovy
```

```
implementation 'com.google.android.material:material:1.5.0'
```

- **Retrofit o Volley (si se requiere comunicación con un backend):** Si planeas enviar datos a un servidor (como el correo y la contraseña), es posible que necesites una librería para realizar solicitudes HTTP:

```
groovy
```

```
implementation 'com.squareup.retrofit2:retrofit:2.9.0'
```

```
implementation 'com.android.volley:volley:1.2.0'
```

- **Glide o Picasso (para carga de imágenes):** Si tu interfaz necesita mostrar imágenes, estas librerías son útiles:


```
groovy
```

```
implementation 'com.github.bumptech.glide:glide:4.12.0'
```

```
implementation 'com.squareup.picasso:picasso:2.71828'
```

2. Frameworks a Utilizar

- **MVVM (Modelo-Vista-VistaModelo):** Aunque el código XML actual no implementa este patrón, se recomienda adoptarlo para una mejor separación de la lógica de negocio y la UI, facilitando pruebas y mantenimiento. Esto implica el uso de ViewModels y LiveData de la biblioteca AndroidX:



groovy

implementation 'androidx.lifecycle:lifecycle-viewmodel-ktx:2.4.1'

implementation 'androidx.lifecycle:lifecycle-livedata-ktx:2.4.1'

- **Retrofit:** Si decides implementar la comunicación con un servicio web, Retrofit es un framework ideal para la creación de API REST, facilitando las solicitudes y manejo de respuestas.
- **Dagger o Hilt:** Para inyección de dependencias, considera usar Hilt, que simplifica la inyección en aplicaciones Android:

groovy

Copiar código

implementation 'com.google.dagger:hilt-android:2.40.5'

kapt 'com.google.dagger:hilt-compiler:2.40.5'

Conclusión

El desarrollo de módulos para aplicaciones móviles en Android es un proceso que requiere un enfoque meticuloso y una comprensión profunda de los requerimientos del sistema, así como de las herramientas y metodologías de desarrollo. A través de esta evidencia de producto, se busca no solo crear un producto funcional, sino también fomentar buenas prácticas de codificación y documentación que aseguren la calidad y mantenibilidad del software. La culminación de este proyecto no solo contribuye a la formación del aprendiz en el campo del desarrollo móvil, sino que también prepara el terreno para futuros desafíos en el ámbito del desarrollo de software.