
Especificación de requisitos de software

Proyecto: ASPRA Móvil





Proyecto Integrador 2
Especificación de requisitos de software

Ficha del documento

Fecha	Revisión	Autor	Verificado dep. Calidad.
2023	31-08	ASPRA Móvil	Verificado
2023	09-09	ASPRA Móvil	Verificado
2023	27-09	ASPRA Móvil	Verificado
2023	01-10	ASPRA Móvil	Verificado
2023	06-10	ASPRA Móvil	Verificado
2023	07-10	ASPRA Móvil	Verificado
2023	08-10	ASPRA Móvil	Verificado

Contenido

FICHA DEL DOCUMENTO	3
CONTENIDO	4
1 INTRODUCCIÓN	5
1.1 Propósito	5
1.2 Alcance	5
1.3 Personal involucrado	5
1.4 Definiciones, acrónimos y abreviaturas	6
1.5 Referencias	7
1.6 Resumen	7
2 DESCRIPCIÓN GENERAL	7
2.1 Perspectiva del producto	8
2.2 Características de los usuarios	8
2.3 Restricciones	8
3 REQUISITOS ESPECÍFICOS	9
3.1 Requerimientos Funcionales y No Funcionales	9
3.1.1 Requerimientos Funcionales (RF)	9
3.1.2 Requerimientos No Funcionales (RNF)	9
3.2 Product Backlog	10
3.2.1 Tasks (Tareas)	10
3.2.2 User Stories (Historias de Usuario)	12
3.3 Sprints	13
3.3.1 Sprint 0	13
3.3.2 Sprint 1	13
3.3.3 Sprint 2	14

3.4	Anexo II: Capturas de las Activitys	15
3.4.1	Pantallas	15
3.4.1.1	Sprint 0	15
3.4.1.2	Inicio de Sesión	16
3.4.1.3	Registrarse	17
3.4.1.4	Reportar	18
3.4.1.5	Reporte Exitoso	19
3.4.1.6	Contacto	20
3.4.2	Código	21
3.4.2.1	Pantalla Principal	21
3.4.2.2	Inicio de Sesión	22
3.4.2.3	Registrarse	23
3.4.2.4	Reportar	24
3.4.2.5	Reporte Exitoso	26
3.4.2.6	Contacto	27
3.4.3	Diseño (Layout)	28
3.4.3.1	Pantalla Principal	28
3.4.3.2	Inicio de Sesión	29
3.4.3.3	Registrarse	30
3.4.3.4	Reportar	31
3.4.3.5	Reporte Exitoso	32
3.4.3.6	Contacto	33
3.5	Video Demo (Sprint 1)	34

1 Introducción

Este documento es una Especificación de Requisitos Software (ERS) para el Sistema de información para la gestión de procesos y control basándose en las directrices dadas por el estándar IEEE para IRS ANSI/IEEE 830, 1998. Desarrollando formalmente el documento necesario para fomentar el aprendizaje, desarrollo y organización del grupo para poder crear la aplicación ("ASPRA Móvil") en Android la cual sea innovadora y funcional enriqueciendo nuestra experiencia como desarrolladores.

1.1 Propósito

El presente documento tiene como propósito dar a conocer el funcionamiento general o, más específicamente, definir las especificaciones funcionales, para el desarrollo de un sistema de información móvil destinado al proyecto ASPRA Móvil que está dirigido tanto a los administradores del negocio (y sus respectivos empleados), como a los clientes del mismo.

1.2 Alcance

Esta especificación de requisitos está dirigida al usuario del sistema, que posea conocimientos y el acceso a navegar en Internet, que cumpla con los requisitos de seguridad y esté de acuerdo con los términos y condiciones del Sistema.

1.3 Personal involucrado

Nombre	Juan Diego Gonzalez Antoniazzi
Rol	Scrum Master - Tester
Categoría Profesional	Junior
Responsabilidad	Probar funcionalidades y ejecutar pruebas
Información de contacto	Mail: juandi19972008@gmail.com

Nombre	Federico Fabián Bellon
Rol	Desarrollador
Categoría Profesional	Junior
Responsabilidad	Desarrollar colaborativamente el producto
Información de contacto	Mail: federicobellon@gmail.com

Nombre	Alejandro Moreno Cabrero
Rol	Desarrollador
Categoría Profesional	Junior/Trainee
Responsabilidad	Desarrollar colaborativamente el producto
Información de contacto	Mail: Morenocabreroalejandro@gmail.com

Nombre	Melanie N. Reyes M.
---------------	---------------------

Rol	Desarrolladora
Categoría Profesional	Junior
Responsabilidad	Desarrollar colaborativamente el producto
Información de contacto	Mail: melanie.reyes.tecnicos2017@gmail.com

Nombre	Fabiana Jazmin Amato
Rol	Desarrollador
Categoría Profesional	Junior
Responsabilidad	Desarrollar colaborativamente el producto
Información de contacto	Mail: proyectos.amato@gmail.com

Nombre	Axel Ezequiel Montivero
Rol	Desarrollador
Categoría Profesional	Trainee
Responsabilidad	Desarrollar colaborativamente el producto
Información de contacto	Mail: ezmant14@gmail.com

Nombre	Nicolas Alberto Mattos
Rol	Desarrollador
Categoría Profesional	Junior
Responsabilidad	Desarrollar colaborativamente el producto
Información de contacto	Mail: nicomattos2014@gmail.com

1.4 Definiciones, acrónimos y abreviaturas

Nombre	Descripción
Usuario	Persona que usará el sistema
SIS-I	Sistema de Información Web para la Gestión de Procesos Administrativos y Académicos
ERS	Especificación de Requisitos Software
RF	Requerimiento Funcional
RNF	Requerimiento No Funcional
FTP	Protocolo de Transferencia de Archivos
Moodle	Aula Virtual
ISPC	Instituto Superior Politécnico Córdoba
TSDWAD	Tecnicatura Superior en Desarrollo Web y Aplicaciones Digitales
BE	Back-End
FE	Front-End
BBDD	Bases de Datos
GitHub	Donde se aloja proyecto de Git para su trabajo de manera remota
Git	Programa de control de versiones, utilizado para trabajo en grupo de manera eficiente
Header	Parte superior de la estructura de una página web
Login	Lugar del sitio web donde el usuario usa sus credenciales para iniciar sesión
US	Historia de Usuario

TK	Tarea
DB	Base de Datos
DER	Modelo Entidad-Relación
MVC	Modelo-Vista-Controlador
CRUD	Crear-Leer-Actualizar-Borrar
DAO	Organización Autónoma Descentralizada
Java	Lenguaje de programación
Android Studio	Entorno de desarrollo integrado de apps para Android
Django	Framework de desarrollo de código (escrito en Python) que respeta el patrón de diseño conocido como MVC

1.5 Referencias

Título del Documento	Referencia
Standard IEEE 830 - 1998	IEEE

1.6 Resumen

En este documento informativo sobre el proyecto se detalla de qué se trata el mismo, como se va llevar a cabo, qué lenguajes se van a utilizar, qué rol y quienes son sus integrantes.

En la introducción se abordan puntos que hacen a la comprensión general del proyecto, su propósito, su objetivo, su área de acción, y conceptos necesarios para la lectura.

A continuación se describen las características y componentes del sitio, así como los roles de desarrolladores y participantes.

Y finalmente, en la sección de requisitos se presenta de forma organizada la serie de pasos técnicos y metodológicos necesarios para cumplir con el objetivo.

Este documento cuenta con 3 (tres) secciones:

- La primera sección describe al producto, su alcance y su funcionalidad.
- La segunda sección se centra en el producto, describiendo su propósito, características de los diferentes usuarios involucrados, lenguajes de codificación utilizados y restricciones del sistema.
- La tercera parte define los requerimientos para el desarrollo del producto y las instancias en las que éstos se irán completando

2 Descripción general

2.1 Perspectiva del producto

La aplicación “ASPRA Móvil” será un producto diseñado para trabajar en el entorno móvil de Android lo que permitirá su utilización de forma rápida y eficaz, además de ofrecer seguridad y confiabilidad a la hora de realizar adopciones, seguimientos, donaciones y contactar veterinarios. Para este proyecto, nos enfocaremos principalmente en el sistema de adopción.

2.2 Características de los usuarios

Tipo de usuario	Entidad protectora de animales
Formación	Cuidado y atención de animales en situación de vulnerabilidad
Actividades	Brindar información, fotos y/o videos sobre los animales rescatados. Realizar seguimiento de reportes de animales extraviados.

Tipo de usuario	Visitante
Formación	Manejo de entorno móvil en Android
Actividades	Ver animales rescatados por la organización. Contactar veterinarios.

Tipo de usuario	Visitante registrado
Formación	Manejo de entorno móvil en Android
Actividades	Realizar seguimiento de los animales rescatados para adoptar y/o realizar una donación a la entidad. Reportar animales perdidos, abandonados o maltratados. Contactar veterinarios.

2.3 Restricciones

- Interfaz para ser usada con internet (puede ser desde el celular y/o computadora).
- Interfaces de usuario intuitivas.
- Funcional en las versiones de Android más comunes.
- Lenguajes y tecnologías en uso: Java, Django, Base de Datos, Git y GitHub.
- La comunicación entre cliente y servidor deberá establecerse a través de protocolos HTTP.
- El uso de la aplicación requiere de conexión a internet
- Consideraciones de seguridad.
- Políticas de la empresa
- Protocolos de comunicación con soporte técnico.
- Los servidores deberán atender consultas concurrentemente.
- Disponibilidad de dispositivo móvil para el acceso a la aplicación.

3 Requisitos específicos

3.1 Requerimientos Funcionales y No Funcionales

3.1.1 Requerimientos Funcionales (RF)

Identificación Requerimiento	Nombre del Requerimiento	Características	Descripción	Prioridad
RF#01	Registrar Usuario	Permite a los usuarios registrarse en la aplicación	Los usuarios deben poder crear una cuenta proporcionando su información personal	Alta
RF#02	Iniciar Sesión de Usuario	Los usuarios deben poder iniciar sesión con sus cuentas registradas	Los usuarios deben poder acceder a sus perfiles y realizar acciones personalizadas después de iniciar sesión	Alta
RF#03	Cerrar Sesión de Usuario	Los usuarios deben poder cerrar sesión en la aplicación	Los usuarios deben poder proteger su privacidad y seguridad al cerrar sesión en la aplicación	Media
RF#04	Reportar animales abandonados o maltratados	El usuario puede reportar animales abandonados o maltratados a través de la aplicación	Los usuarios se les debe permitir contribuir al rescate de animales informando sobre mascotas abandonadas o maltratadas y proporcionando detalles como ubicación, descripción y fotografías	Alta
RF#05	Instalación en Dispositivos Android	La aplicación debe estar disponible para instalación en dispositivos Android	La aplicación debe ser accesible para usuarios de dispositivos Android a través de la tienda de aplicaciones	Alta
RF#06	Contactar Refugio/Albergue	Los usuarios deben poder contactar al refugio o albergue donde se encuentra la mascota	Los usuarios deben tener la capacidad de comunicarse con el refugio o albergue para obtener más información sobre una mascota	Media

3.1.2 Requerimientos No Funcionales (RNF)

Identificación Requerimiento	Nombre del Requerimiento	Características	Descripción	Prioridad
<i>RNF#01</i>	<i>Compatibilidad con Android</i>	<i>La aplicación debe ser compatible con una amplia variedad de dispositivos Android.</i>	<i>La aplicación debe funcionar de manera consistente en diferentes modelos y versiones de dispositivos Android</i>	<i>Alta</i>
<i>RNF#02</i>	<i>Seguridad de Datos</i>	<i>Los datos de los usuarios y de los animales deben estar protegidos contra accesos no autorizados</i>	<i>La seguridad de los datos es esencial para proteger la información confidencial de los usuarios y los detalles de los animales</i>	<i>Alta</i>
<i>RNF#03</i>	<i>Disponibilidad</i>	<i>La aplicación debe estar disponible las 24 horas y los 7 días de la semana</i>	<i>La aplicación debe estar disponible en todo momento para garantizar que los usuarios puedan acceder cuando lo necesiten</i>	<i>Alta</i>
<i>RNF#04</i>	<i>Usabilidad</i>	<i>La aplicación debe ser intuitiva y fácil de usar</i>	<i>La usabilidad es clave para asegurar que los usuarios puedan utilizar la aplicación de manera eficiente</i>	<i>Alta</i>
<i>RNF#05</i>	<i>Escalabilidad</i>	<i>La aplicación debe ser escalable para acomodar un crecimiento en el número de usuarios y datos</i>	<i>La aplicación debe estar preparada para manejar un aumento en la demanda y el volumen de información</i>	<i>Alta</i>

3.2 Product Backlog

3.2.1 Tasks (Tareas)

- ☐ TK#01 Designar al / a la Scrum Master inicial
- ☐ TK#02 Registrar a cada Miembro en el Repositorio
- ☐ TK#03 Crear documentación IEEE830
- ☐ TK#04 Registrar Reuniones (Meetings)
- ☐ TK#05 Crear el Project estilo Kanban
- ☐ TK#06 Crear Wiki del Proyecto

- ☐ TK#07 Elaborar lista de pendientes de producto (Product Backlog)
- ☐ TK#08 Verificar Script SQL de la Base de Datos
- ☐ TK#09 Diseñar mapa de sitio de la aplicación
- ☐ TK#10 DER (Diagrama de Entidad-Relación)
- ☐ TK#11 Modelo Relacional
- ☐ TK#12 Diagrama de Caso de Uso
- ☐ TK#13 Diagrama de Clases
- ☐ TK#14 Diseñar pantallas de la app en Figma
- ☐ TK#15 Implementación navegabilidad en Android Studio
- ☐ TK#16 Diseño de pantalla principal (Home) en Android Studio
- ☐ TK#17 Diseño de pantalla de "Contacto" en Android Studio
- ☐ TK#18 Diseño de pantalla de inicio de sesión (Login) en Android Studio
- ☐ TK#19 Diseño de pantalla de registro de usuario (Registro) en Android Studio
- ☐ TK#20 Diseño de pantalla de reportes (Reporte) en Android Studio
- ☐ TK#21 Integración de Pantallas en Android Studio
- ☐ TK#22 Planificación de Casos de Pruebas
- ☐ TK#23 Diseñar interfaz de usuario para la pantalla de Productos/Servicios
- ☐ TK#24 Desarrollar funcionalidad de CRUD básico para Productos/Servicios
- ☐ TK#25 Diseñar interfaz de usuario para la pantalla de Clientes/Usuarios
- ☐ TK#26 Desarrollar funcionalidad de CRUD básico para Clientes/Usuarios
- ☐ TK#27 Actualizar documentación IEEE830
- ☐ TK#28 Actualizar Tablero Kanban
- ☐ TK#29 Actualizar Wiki
- ☐ TK#30 Realizar respaldo del proyecto en GitHub
- ☐ TK#31 Preparar Demo Final en video
- ☐ TK#32 Exportar aplicación y subirla a GitHub

- ☐ TK#33 Grabar video grupal de todas las funcionalidades
- ☐ TK#34 Publicar la aplicación

3.2.2 User Stories (Historias de Usuario)

- ☐ US#01 Como usuario, quiero reportar animales perdidos o encontrados para contribuir a su rescate.
- ☐ US#02 Como usuario, quiero ver animales perdidos o encontrados.
- ☐ US#03 Como usuario, quiero poder ver la información de una mascota encontrada.
- ☐ US#04 Como usuario, quiero contactarme con el refugio o albergue donde se encuentra la mascota.
- ☐ US#05 Como usuario, quiero registrarme en la aplicación.
- ☐ US#06 Como usuario, quiero poder filtrar las mascotas por especie, edad y ubicación para encontrar una que se ajuste a mis preferencias.
- ☐ US#07 Como usuario, quiero poder ver los detalles de una mascota seleccionada.
- ☐ US#08 Como usuario, quiero seguir el progreso de los animales rescatados mediante la aplicación para estar informado sobre su situación.
- ☐ US#09 Como usuario, quiero poder compartir fotos o videos de los animales rescatados.
- ☐ US#10 Como usuario, quiero crear una cuenta en la aplicación para acceder a todas las funcionalidades.
- ☐ US#11 Como usuario, quiero poder iniciar sesión en la aplicación con mi cuenta registrada para acceder a mi perfil y realizar acciones personalizadas.
- ☐ US#12 Como usuario, quiero poder reportar un animal perdido o abandonado a través de la aplicación, proporcionando detalles como la ubicación, descripción y fotos del animal.
- ☐ US#13 Como usuario, quiero poder marcar un animal como rescatado en la aplicación, una vez que haya sido encontrado.
- ☐ US#14 Como usuario, quiero recibir notificaciones en la aplicación cuando haya actualizaciones sobre los animales que he reportado y/o rescatado.
- ☐ US#15 Como usuario, quiero poder editar mi perfil en la aplicación, para actualizar mi información personal y preferencias de notificación.
- ☐ US#16 Como usuario, quiero cerrar sesión en la aplicación para proteger mi privacidad y seguridad.
- ☐ US#17 Como usuario, quiero poder instalar la aplicación en mi dispositivo Android para

acceder a sus funcionalidades.

- ☐ US#18 Como administrador, quiero tener acceso a un panel de administración en la aplicación para gestionar los reportes de animales perdidos y abandonados.
- ☐ US#19 Como usuario, quiero acceder a un enlace que redirija al WhatsApp del dueño del animal desde la aplicación para facilitar el contacto.

3.3 Sprints

3.2.1 Sprint 0

N° de sprint	00
Sprint Backlog	<ul style="list-style-type: none"> Definir requerimientos funcionales para la app a desarrollar (colocarlos en el Product Backlog del Project), a su vez revisar si han cumplimentado todos los requerimientos previos, realizando mejoras del mismo. Definir tareas dentro de las Historias de Usuario (GITHUB). Plantear Historias de Usuarios y Tareas dependientes de las US para incorporarlas en el repositorio remoto gitHub. (Issues y Milestones).
Responsabilidades	Todo el equipo
Calendario	Fecha Inicio = 28/08/2023 - Fecha de Fin = 05/09/2023
Inconvenientes:	Sin inconvenientes para la realización de dicho Sprint.

3.2.2 Sprint 1

N° de sprint	01
Sprint Backlog	<ul style="list-style-type: none"> Crear su propio DER y Modelo relacional para documentar las tablas en la DB. Crear un Diagrama de Clases y Casos de Uso para facilitar el modelado en POO. Script SQL de la base de datos actualizada con las tablas nuevas. Actualizar el repositorio grupal, creando una carpeta dentro de la branch. <p><u>Diseño de pantallas de Activity</u></p> <ul style="list-style-type: none"> Se puede presentar maquetación en Figma y principalmente navegabilidad el Proyecto de Android Studio (paso previo sin funcionalidad).

	<ul style="list-style-type: none"> • Incluir una pantalla para contacto. • Incorporar la navegabilidad de la aplicación completa. • Subirla al repo grupal, con GIT , en una branch por desarrollador para que cada uno tenga una copia. Luego crear una branch feature, para realizar nuestros cambios que no se encuentran aún en condiciones de incorporarse a la branch develop. • La branch "MAIN" o "MASTER" es la que suele usarse para mantener las versiones estables (productivas) de nuestros proyectos. Listas para salir a producción (Aquí deben dejar las versiones al cierre del SPRINT). • Pueden activar en GitHub la opción WIKI para reflejar avances individuales y de equipo.
Responsabilidades	Todo el equipo
Calendario	Fecha Inicio = 25/09/2023 - Fecha de Fin = 01/10/2023
Inconvenientes:	Sin inconvenientes para la realización de dicho Sprint.

3.2.3 Sprint 2

N° de sprint	02
Sprint Backlog	<p><i>Requerimientos del Incremento del producto</i></p> <ul style="list-style-type: none"> • Funcionalidades completas (diseño y desarrollo integrado). • CRUD básico funcional, uno para productos/servicios y otro para clientes/usuarios. • Actualizar documentación IEEE830 - tablero Kanban (project) y Wiki. <ul style="list-style-type: none"> • Actualizado todo el proyecto en GitHub como respaldo. • Demo final en video con participación de todo el equipo describiendo todas las funcionalidades. • Exportar la aplicación en un archivo para poder probarla y subirla a GitHub. <ul style="list-style-type: none"> • Video grupal de todas las funcionalidades de la aplicación. • OPCIONALES: aplicación publicada.
Responsabilidades	Todo el equipo
Calendario	Fecha Inicio = 23/10/2023 - Fecha de Fin = 27/10/2023
Inconvenientes:	

3.4 Anexo II: Capturas de las Activitys



3.4.1 Pantallas


3.4.1.1 Pantalla Principal



3.4.1.2 Inicio de Sesión

11:11

 50 %

 Inicio

Inicio de Sesión

Correo

Contraseña

¿Has olvidado tu contraseña?

INGRESAR

¿Desea registrarse?

REGISTRARSE

3.4.1.3 Registrarse

11:11

  50 %

 Inicio de Sesión

Nombre de usuario

Correo electrónico

Teléfono

Contraseña

Confirme la contraseña

CONFIRMAR DATOS

3.4.1.4 Reportar



Reportar

Direccion

Por favor, ingrese una dirección

Motivo

Maltrato
Abandono
Otro

Descripcion

Por favor, ingrese una descripción

ENVIAR REPORTE

3.4.1.5 Reporte Exitoso



**Su reporte fue enviado
exitosamente!**

SALIR

3.4.1.6 Contacto



Contacto

Telefono:

+549 000 0000000

Correo electronico:

aspra_contacto@aspra.com

Direccion:

Avenida Siempre Viva 1234

**Para mas
informacion,ingrese a
nuestro sitio web**

Gracias por su colaboracion



3.4.2 Código

3.4.2.1 Pantalla Principal

```
1 package com.example.aspra_app;
2
3 import androidx.appcompat.app.AppCompatActivity;
4
5 import android.content.Intent;
6 import android.os.Bundle;
7 import android.view.View;
8
9
10 9 usages
11 </> public class MainActivity extends AppCompatActivity {
12
13     @Override
14     protected void onCreate(Bundle savedInstanceState) {
15         super.onCreate(savedInstanceState);
16         setContentView(R.layout.activity_main);
17     }
18
19     1 usage
20     public void irLogin(View view) {
21         Intent intent = new Intent( packageContext: this, LoginActivity.class);
22         startActivity(intent);
23     }
24
25     2 usages
26     public void irReportar(View view) {
27         Intent intent = new Intent( packageContext: this, ReportarActivity.class);
28         startActivity(intent);
29     }
30
31     1 usage
32     public void irContacto(View view) {
33         Intent intent = new Intent( packageContext: this, ContactoActivity.class);
34         startActivity(intent);
35     }
36 }
```

Running Devices Pixel 2 XL API 30

11:11 75%

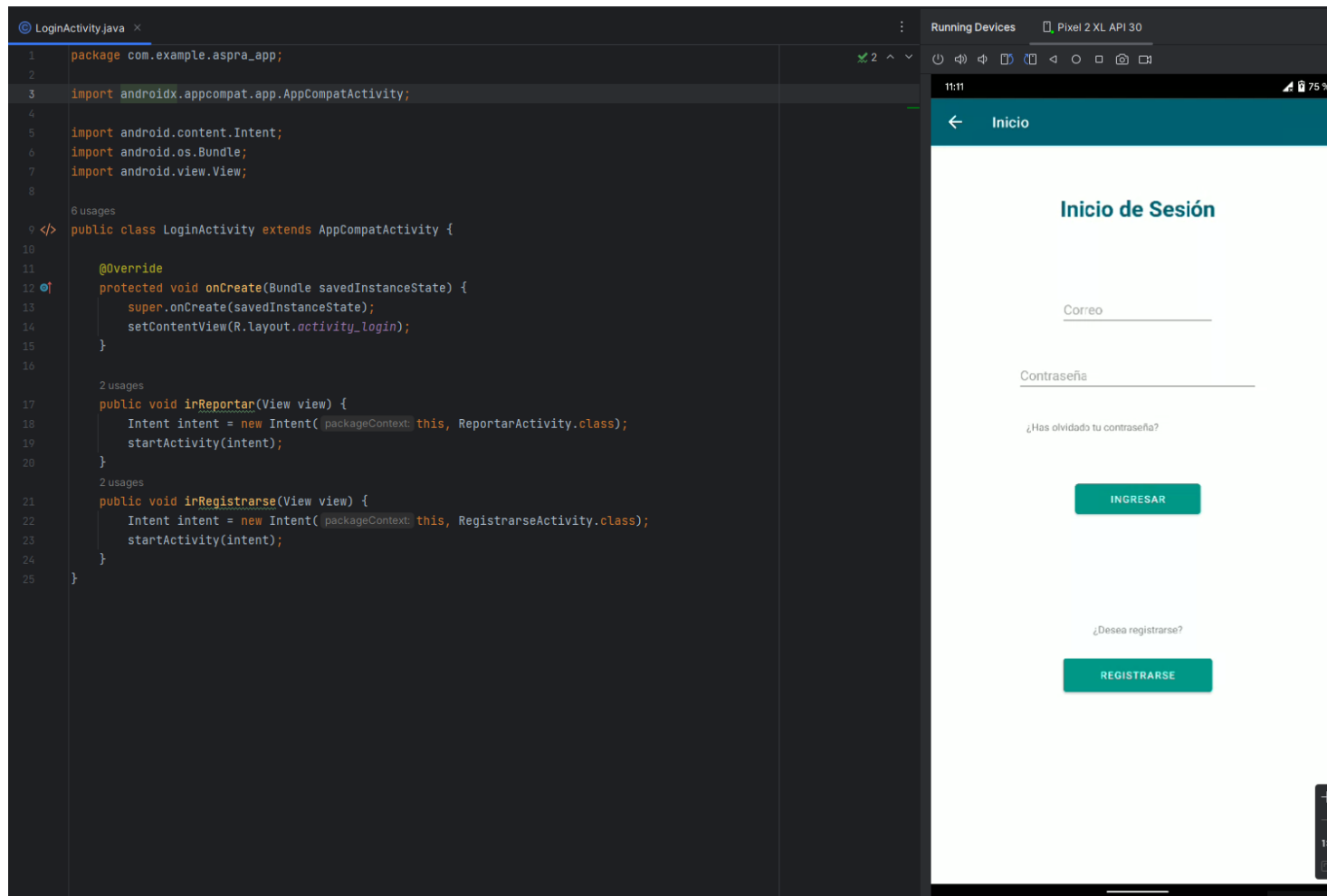
AS.PR.A
Asociacion Protectora de Animales

INICIO DE SESIÓN

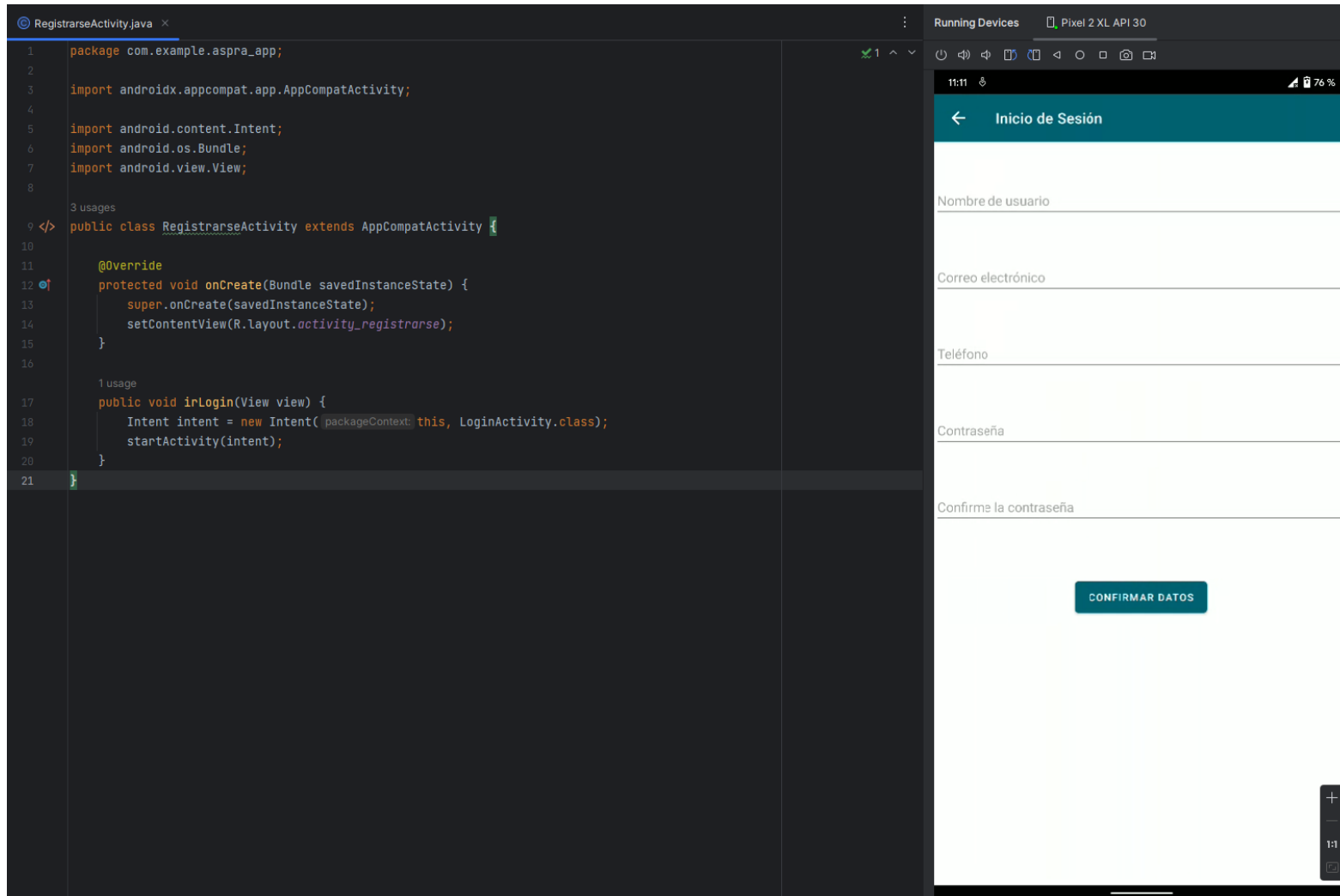
REPORTAR

CONTACTO

3.4.2.2 Inicio de Sesión



3.4.2.3 Registrarse



The screenshot displays the Android Studio environment. On the left, the 'RegistrarseActivity.java' file is open, showing the following code:

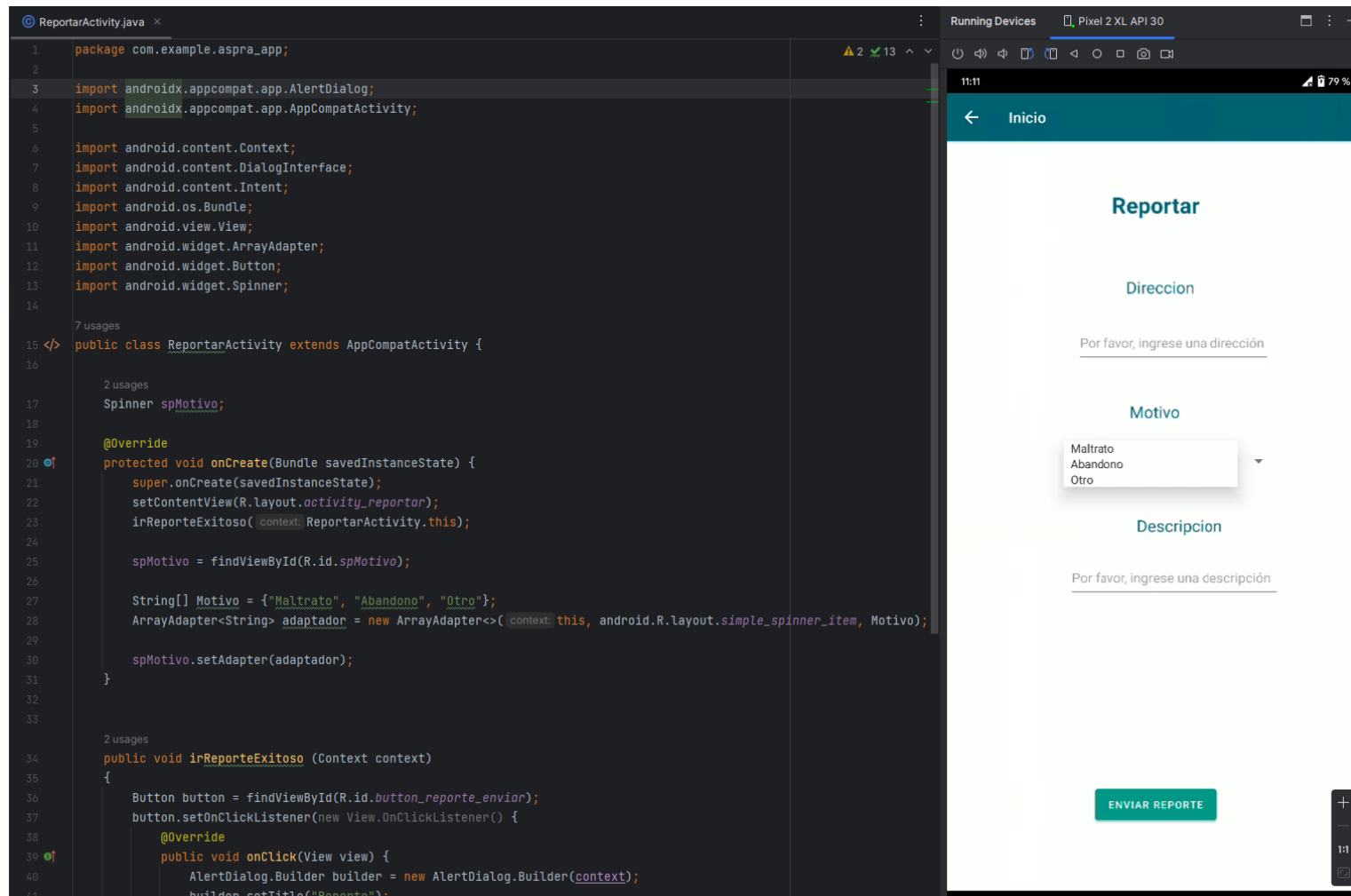
```
1 package com.example.aspra_app;
2
3 import androidx.appcompat.app.AppCompatActivity;
4
5 import android.content.Intent;
6 import android.os.Bundle;
7 import android.view.View;
8
9 <?> public class RegistrarseActivity extends AppCompatActivity {
10
11     @Override
12     protected void onCreate(Bundle savedInstanceState) {
13         super.onCreate(savedInstanceState);
14         setContentView(R.layout.activity_registrarse);
15     }
16
17     1 usage
18     public void irLogin(View view) {
19         Intent intent = new Intent( packageContext: this, LoginActivity.class);
20         startActivity(intent);
21     }
22 }
```

On the right, the 'Running Devices' tab shows a virtual device named 'Pixel 2 XL API 30'. The emulator screen displays a login form titled 'Inicio de Sesión' with the following fields:

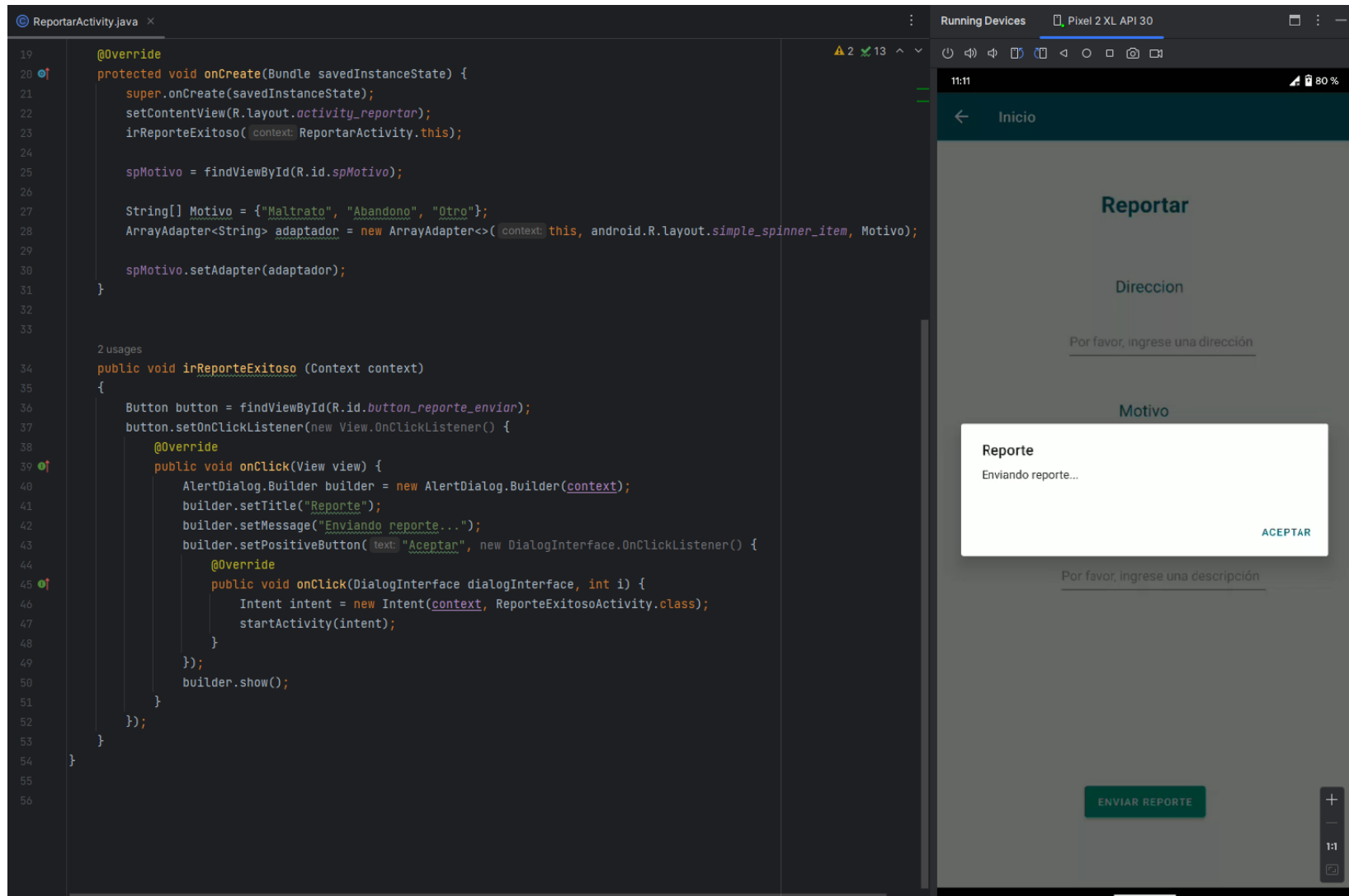
- Nombre de usuario
- Correo electrónico
- Teléfono
- Contraseña
- Confirme la contraseña

A teal button labeled 'CONFIRMAR DATOS' is positioned below the password fields. The emulator's status bar at the top shows the time as 11:11 and a battery level of 76%.

3.4.2.4 Reportar



```
1 package com.example.aspra_app;
2
3 import androidx.appcompat.app.AlertDialog;
4 import androidx.appcompat.app.AppCompatActivity;
5
6 import android.content.Context;
7 import android.content.DialogInterface;
8 import android.content.Intent;
9 import android.os.Bundle;
10 import android.view.View;
11 import android.widget.ArrayAdapter;
12 import android.widget.Button;
13 import android.widget.Spinner;
14
15 7 usages
16 </> public class ReportarActivity extends AppCompatActivity {
17
18     2 usages
19     Spinner spMotivo;
20
21     @Override
22     protected void onCreate(Bundle savedInstanceState) {
23         super.onCreate(savedInstanceState);
24         setContentView(R.layout.activity_reportar);
25         irReporteExitoso(context: ReportarActivity.this);
26
27         spMotivo = findViewById(R.id.spMotivo);
28
29         String[] Motivo = {"Maltrato", "Abandono", "Otro"};
30         ArrayAdapter<String> adaptador = new ArrayAdapter<>(context: this, android.R.layout.simple_spinner_item, Motivo);
31
32         spMotivo.setAdapter(adaptador);
33     }
34
35     2 usages
36     public void irReporteExitoso (Context context)
37     {
38         Button button = findViewById(R.id.button_reporte_enviar);
39         button.setOnClickListener(new View.OnClickListener() {
40             @Override
41             public void onClick(View view) {
42                 AlertDialog.Builder builder = new AlertDialog.Builder(context);
43                 builder.setTitle("Reporte");
```

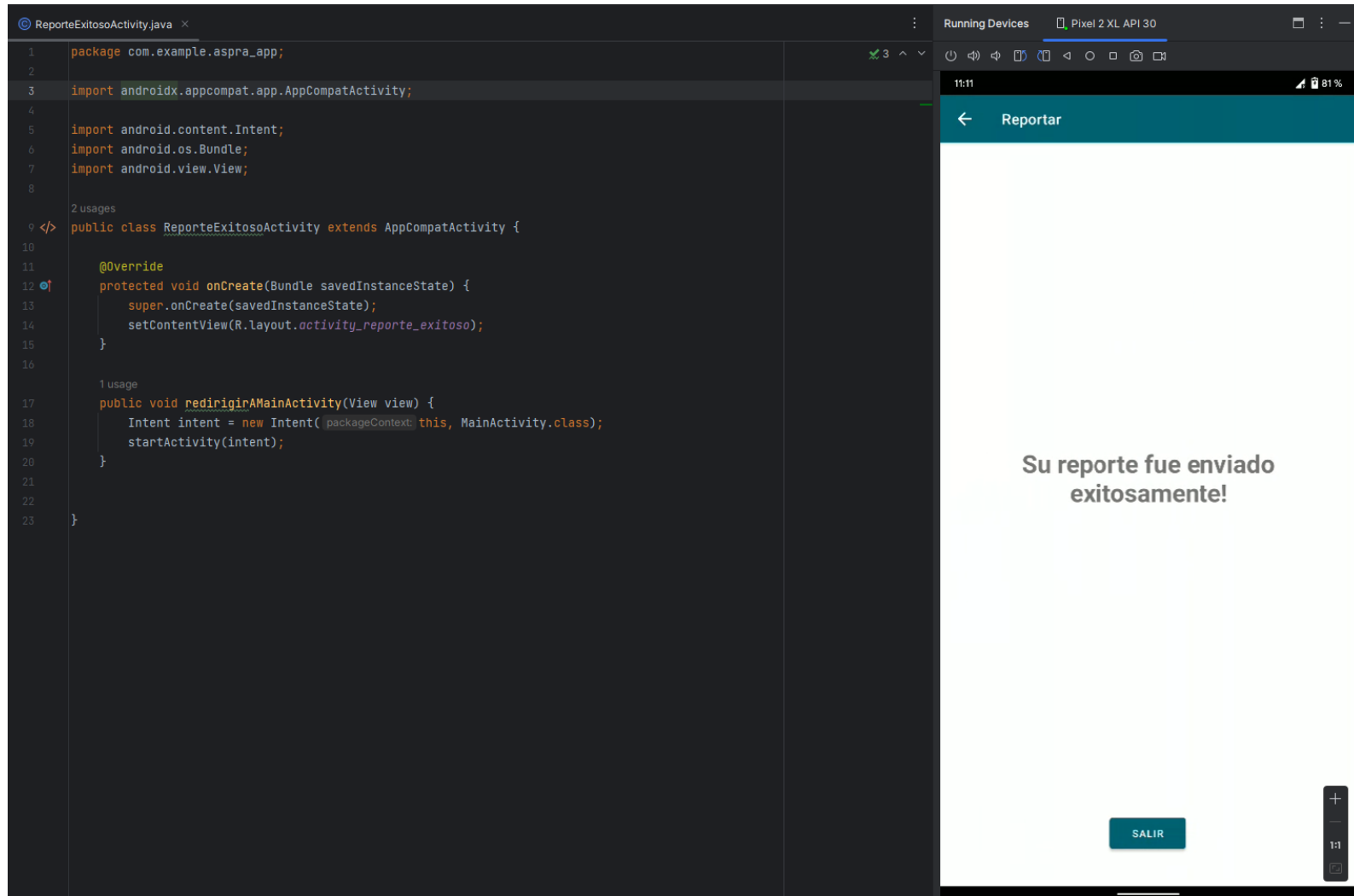
The image shows a screenshot of an IDE (IntelliJ IDEA) with the `ReportarActivity.java` file open. The code implements the `onCreate` and `irReporteExitoso` methods. The `onCreate` method initializes the spinner with a list of motives: "Maltrato", "Abandono", and "Otro". The `irReporteExitoso` method handles the button click by showing an alert dialog with the message "Enviando reporte..." and an "ACEPTAR" button. The alert dialog is currently displayed over the Android emulator.

```
19  @Override
20  protected void onCreate(Bundle savedInstanceState) {
21      super.onCreate(savedInstanceState);
22      setContentView(R.layout.activity_reportar);
23      irReporteExitoso(context, ReportarActivity.this);
24
25      spMotivo = findViewById(R.id.spMotivo);
26
27      String[] Motivo = {"Maltrato", "Abandono", "Otro"};
28      ArrayAdapter<String> adaptador = new ArrayAdapter<>(context, this, android.R.layout.simple_spinner_item, Motivo);
29
30      spMotivo.setAdapter(adaptador);
31  }
32
33  2 usages
34  public void irReporteExitoso (Context context)
35  {
36      Button button = findViewById(R.id.button_reporte_enviar);
37      button.setOnClickListener(new View.OnClickListener() {
38
39          @Override
40          public void onClick(View view) {
41              AlertDialog.Builder builder = new AlertDialog.Builder(context);
42              builder.setTitle("Reporte");
43              builder.setMessage("Enviando reporte...");
44              builder.setPositiveButton(text: "Aceptar", new DialogInterface.OnClickListener() {
45
46                  @Override
47                  public void onClick(DialogInterface dialogInterface, int i) {
48                      Intent intent = new Intent(context, ReporteExitosoActivity.class);
49                      startActivity(intent);
50                  }
51              });
52              builder.show();
53          }
54      });
55  }
56  }
```

The Android emulator shows the "Reportar" screen with the following elements:

- Header: Inicio
- Section: Reportar
- Form: Direccion (Por favor, ingrese una dirección)
- Form: Motivo (Por favor, ingrese una descripción)
- Button: ENVIAR REPORTE
- Alert Dialog: Reporte (Enviando reporte... ACEPTAR)

3.4.2.5 Reporte Exitoso



The screenshot displays the Android Studio environment. On the left, the Java code for `ReporteExitosoActivity` is visible, showing imports for `AppCompatActivity`, `Intent`, `Bundle`, and `View`. The `onCreate` method sets the content view to `R.layout.activity_reporte_exitoso`. The `redirigirAMainActivity` method creates an intent to start the `MainActivity`. On the right, the 'Running Devices' tab shows a virtual Pixel 2 XL with API level 30. The app's UI is displayed, featuring a teal header with a back arrow and the text 'Reportar'. The main content area shows the message 'Su reporte fue enviado exitosamente!' in a large, bold font. At the bottom, there is a teal button labeled 'SALIR'.

```
1 package com.example.aspra_app;
2
3 import androidx.appcompat.app.AppCompatActivity;
4
5 import android.content.Intent;
6 import android.os.Bundle;
7 import android.view.View;
8
9 </> public class ReporteExitosoActivity extends AppCompatActivity {
10
11     @Override
12     protected void onCreate(Bundle savedInstanceState) {
13         super.onCreate(savedInstanceState);
14         setContentView(R.layout.activity_reporte_exitoso);
15     }
16
17     public void redirigirAMainActivity(View view) {
18         Intent intent = new Intent( packageContext, this, MainActivity.class);
19         startActivity(intent);
20     }
21
22 }
23 }
```

Running Devices: Pixel 2 XL API 30

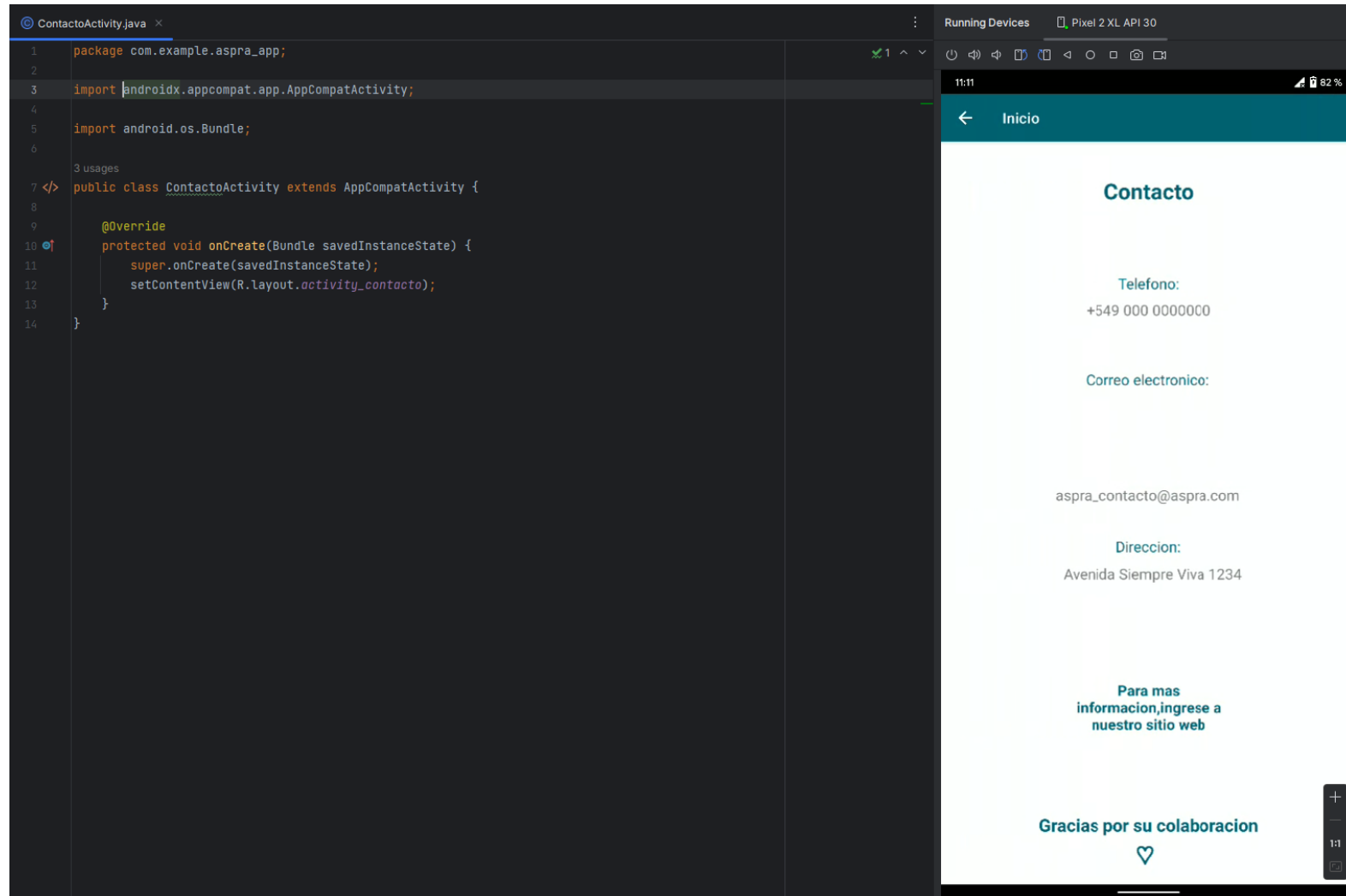
11:11 81%

Reportar

Su reporte fue enviado exitosamente!

SALIR

3.4.2.6 Contacto



The screenshot displays the Android Studio environment. On the left, the `ContactoActivity.java` file is open, showing the following code:

```
1 package com.example.aspra_app;
2
3 import androidx.appcompat.app.AppCompatActivity;
4
5 import android.os.Bundle;
6
7 3 usages
8 </> public class ContactoActivity extends AppCompatActivity {
9
10     @Override
11     protected void onCreate(Bundle savedInstanceState) {
12         super.onCreate(savedInstanceState);
13         setContentView(R.layout.activity_contacto);
14     }
15 }
```

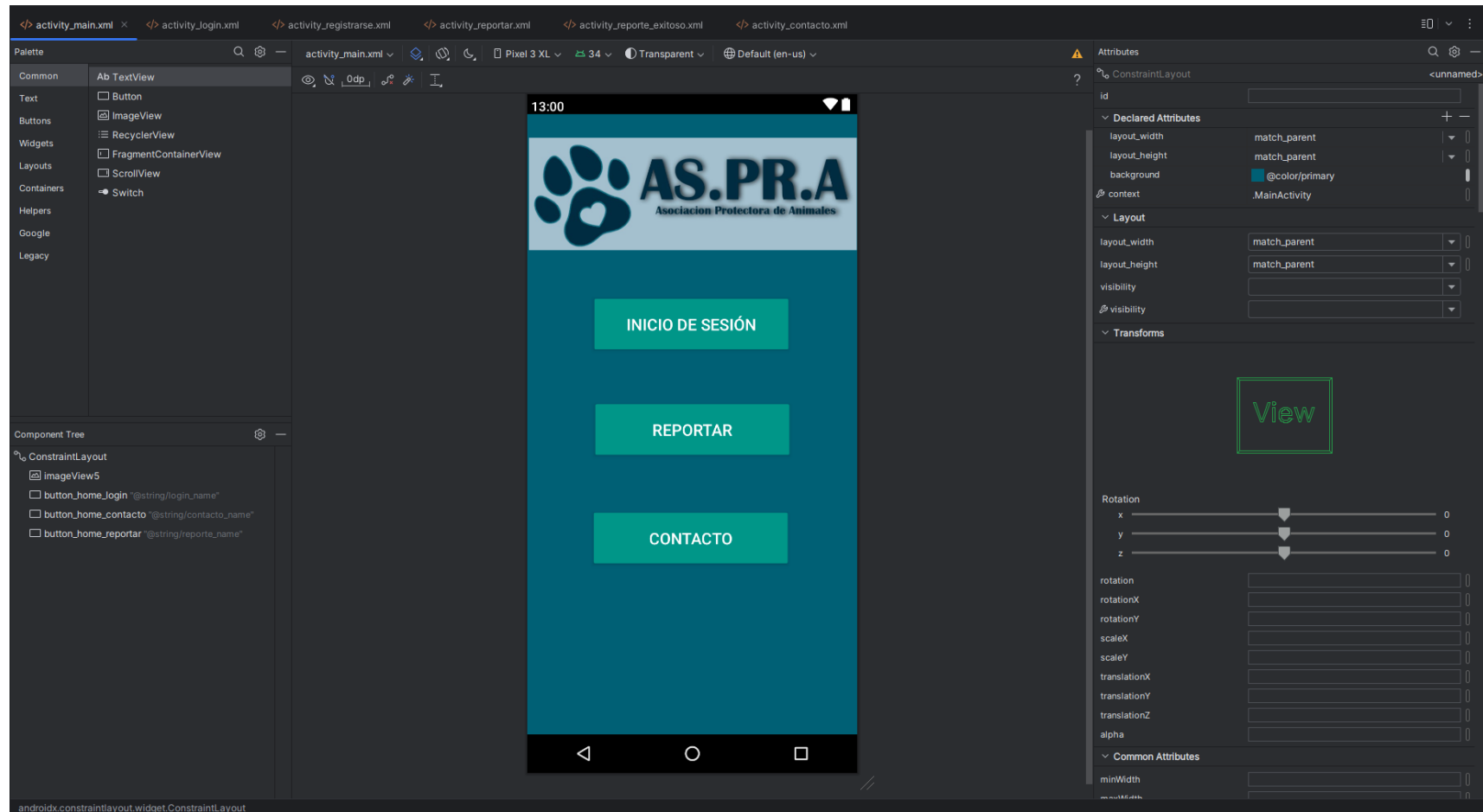
On the right, the 'Running Devices' tab shows a virtual device named 'Pixel 2 XL API 30'. The emulator screen displays the 'Inicio' (Home) screen with a teal header. The main content area is titled 'Contacto' and contains the following information:

- Telefono:** +549 000 0000000
- Correo electronico:** aspra_contacto@aspra.com
- Direccion:** Avenida Siempre Viva 1234
- Para mas informacion,ingrese a nuestro sitio web**
- Gracias por su colaboracion** with a heart icon below it.

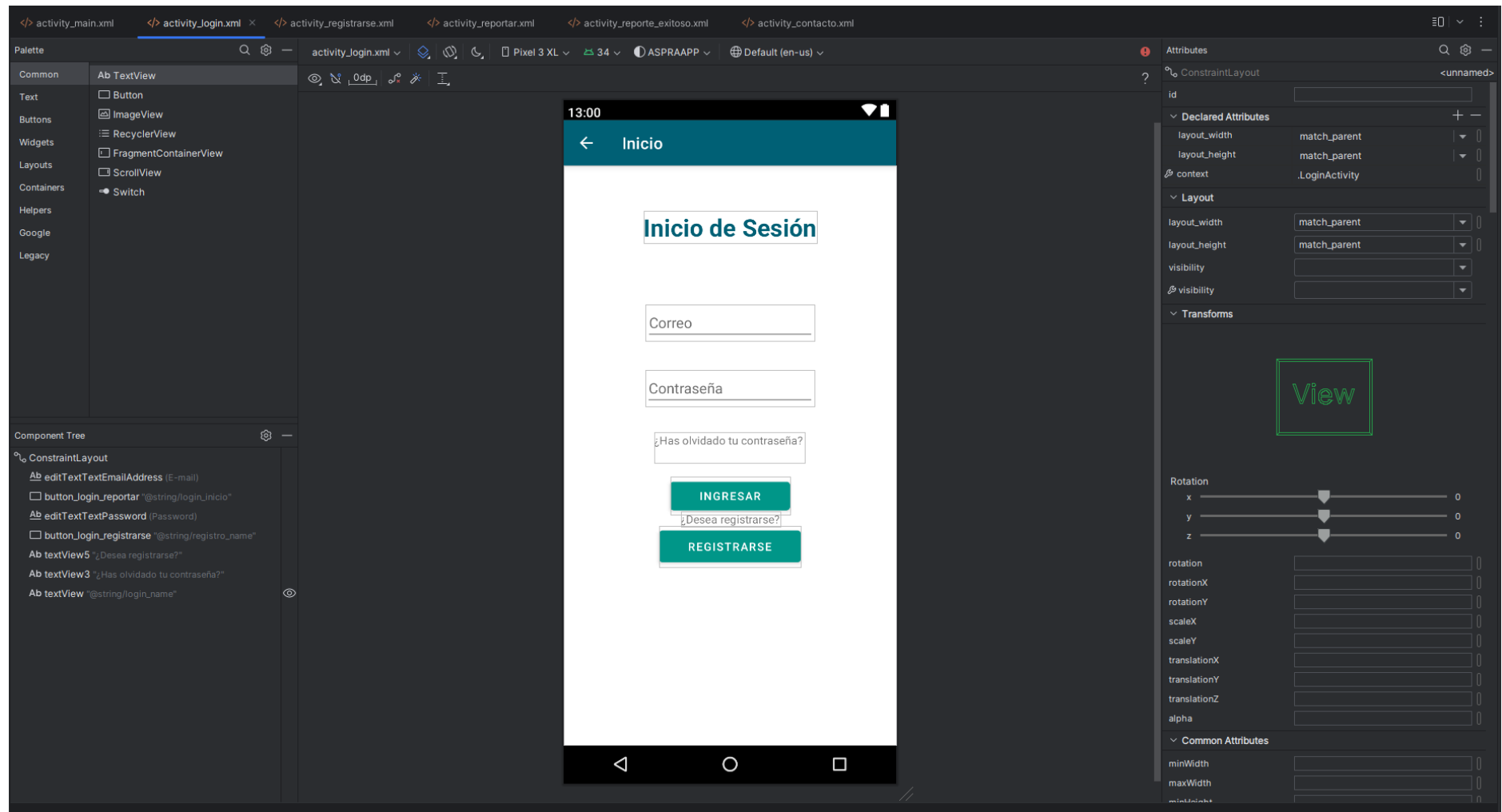
The emulator's status bar at the top shows the time as 11:11 and battery level at 82%.

3.4.3 Diseño (Layout)

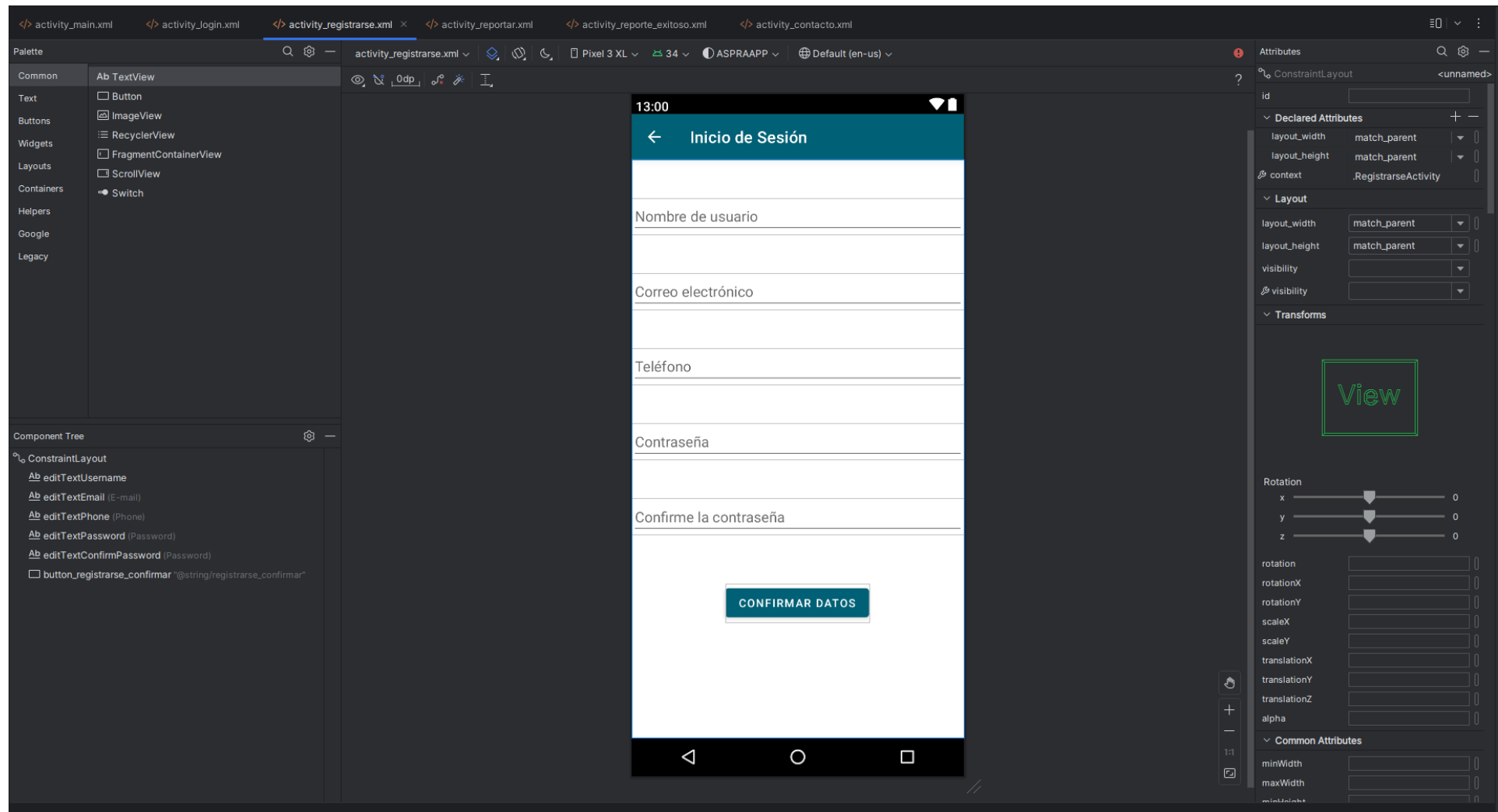
3.4.3.1 Pantalla Principal



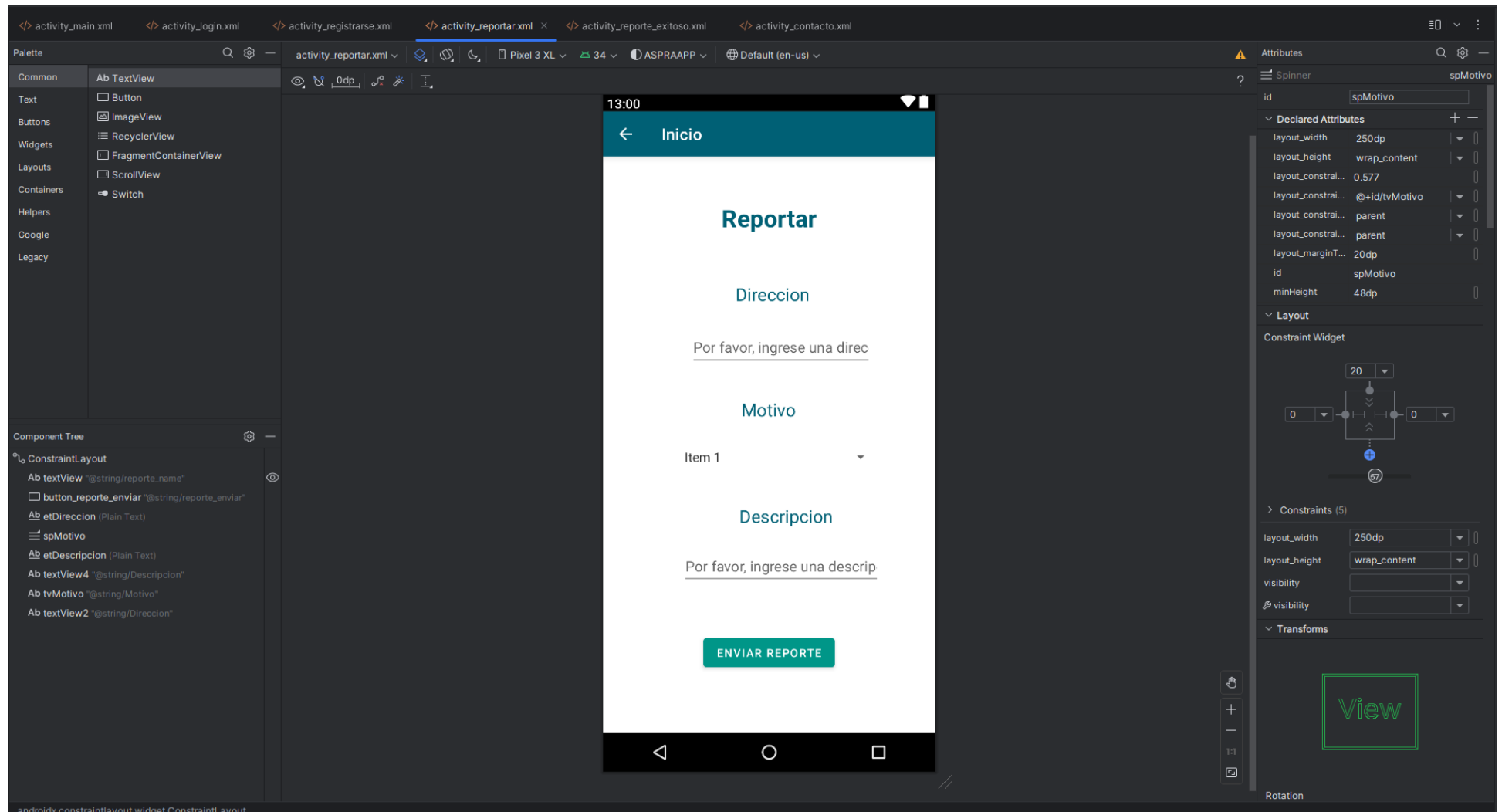
3.4.3.2 Inicio de Sesión



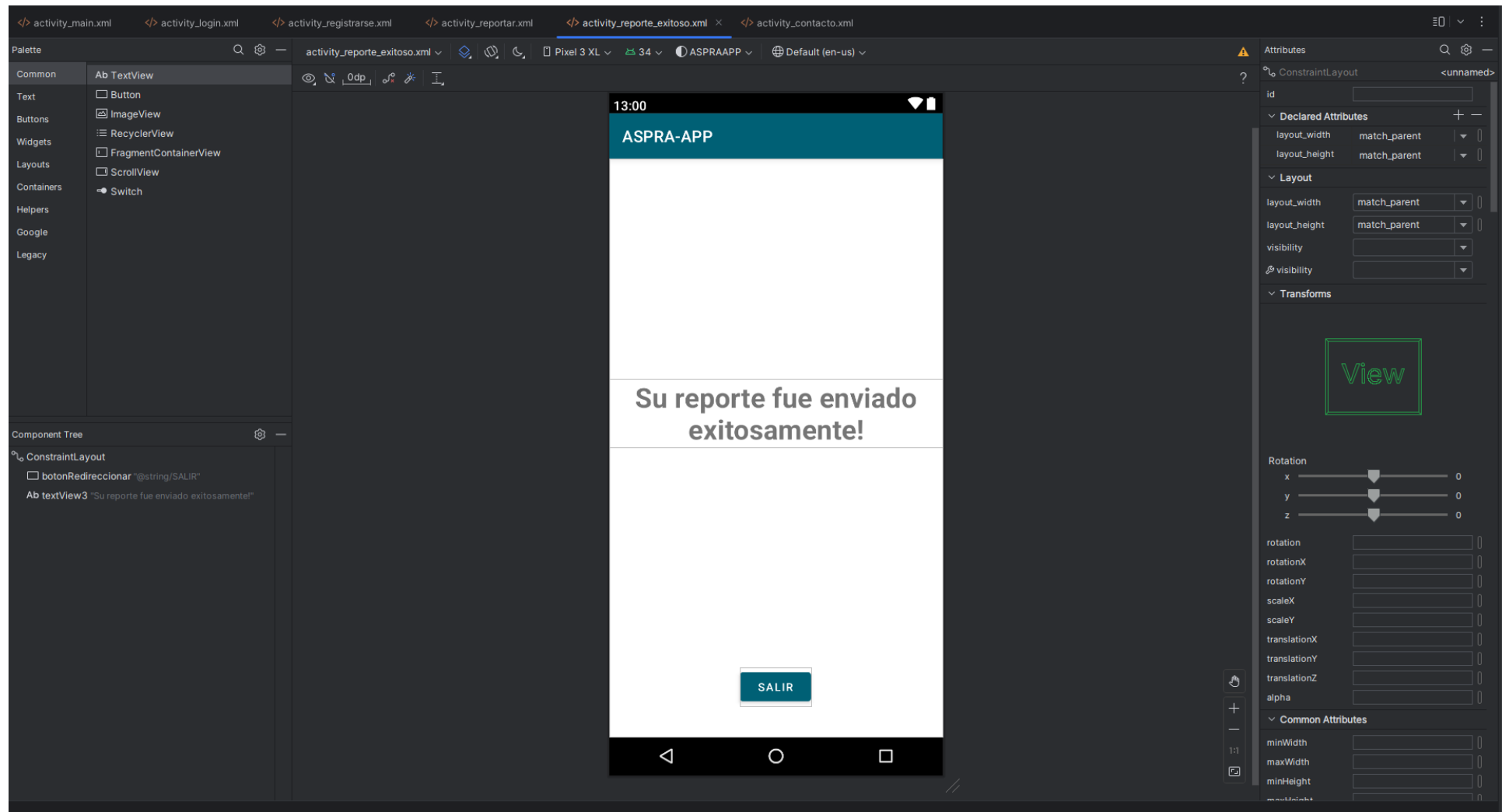
3.4.3.3 Registrarse



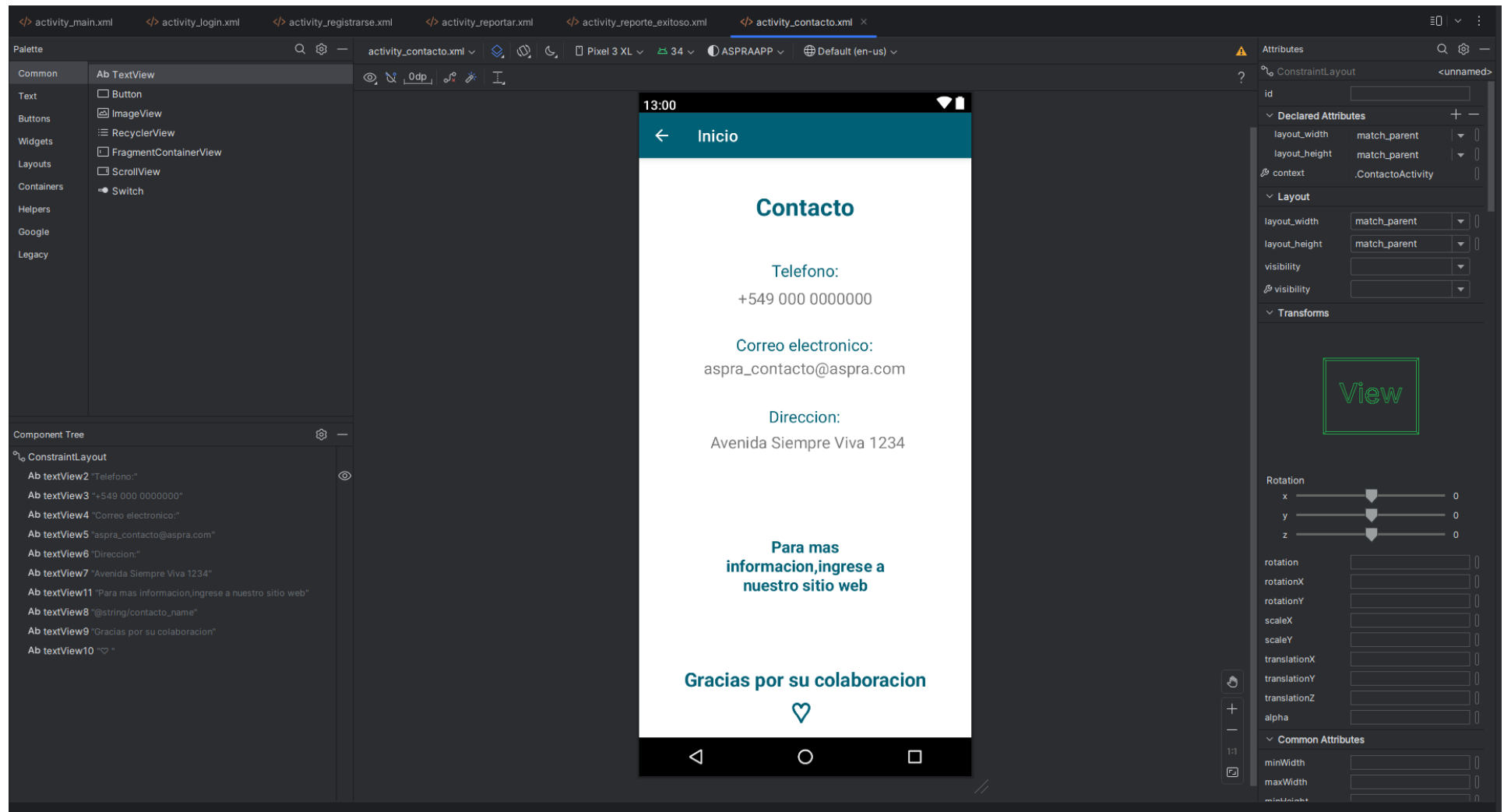
3.4.3.4 Reportar



3.4.3.5 Reporte Exitoso



3.4.3.6 Contacto





Proyecto Integrador 2
Especificación de requisitos de software

3.5 Video Demo (Sprint 1)

[Video Demo ASPRA Móvil Sprint 1](#)