



Asignatura: Application Development for Mobile Devices.

Tarea 37: Envío de Correo.

La siguiente aplicación permite compartir un mensaje de correo electrónico; inclusive, el correo se permite redactarlo incluyendo archivos adjuntos. Se puede utilizar una de las acciones que se indican en la tabla 1 y además incluir los datos del destinatario y del asunto.

Acción	Esquema de URI de datos	Tipo MIME
ACTION_SENDTO (sin archivos adjuntos)	Ninguno	"text/plain"
ACTION_SEND (un archivo adjunto)		"/"
ACTION_SEND_MULTIPLE (varios archivos adjuntos)		

Tabla 1. Opciones de acciones.

La tabla 2 muestra las constantes que representan las opciones de la clase Intent para configurar el envío de un correo.

Extras	
Intent.EXTRA_EMAIL	Una matriz de cadenas con todas las direcciones de correo electrónico Para de los destinatarios.
Intent.EXTRA_CC	Una matriz de cadenas con todas las direcciones de correo electrónico CC de los destinatarios.
Intent.EXTRA_BCC	Una matriz de cadenas con todas las direcciones de correo electrónico CCO de los destinatarios.
Intent.EXTRA_SUBJECT	Una cadena con el asunto del mensaje de correo electrónico.
Intent.EXTRA_TEXT	Una cadena con el cuerpo del mensaje de correo electrónico.
Intent.EXTRA_STREAM	Una Uri que apunta a los archivos adjuntos. Si se usa la acción ACTION_SEND_MULTIPLE, en su lugar debe ser un ArrayList que contenga varios objetos Uri.

Tabla 2. Opciones de extras.

La aplicación básica de envío de un correo en Android sería similar a la siguiente.

```
Intent intent = new Intent(Intent.ACTION_SENDTO);
intent.setType("message/rfc822");
intent.setData(Uri.parse("mailto:"));
intent.putExtra(Intent.EXTRA_EMAIL, new String[]{"usuario@gmail.com"});
intent.putExtra(Intent.EXTRA_SUBJECT, "asunto del mensaje");
intent.putExtra(Intent.EXTRA_TEXT, "cuerpo del mensaje");

try {
    startActivity(Intent.createChooser(intent, "send mail"));
} catch (ActivityNotFoundException ex) {
    Toast.makeText(this, "No mail app found!!!", Toast.LENGTH_SHORT);
} catch (Exception ex) {
    Toast.makeText(this, "Unexpected Error!!!", Toast.LENGTH_SHORT);
}
```

El RFC822

Los Requests for Comments RFC son documentos que describen algunos protocolos de Internet. Otros protocolos se revisan y estandarizan formalmente y son promovidos por otras organizaciones como CCITT y ANSI. El RFC822 es el formato estándar de Internet para encabezados de mensajes de correo electrónico. El nombre del RFC822, contiene la especificación (STD 11, RFC 822). El formato 822 es la actualización del formato 733.

DESARROLLO.

1. Crear un nuevo proyecto en Android Studio. Abrir el archivo MainActivity.java y agregar el siguiente código.

Dr. Alejandro S. Cifuentes A.



```
import android.net.Uri;
import android.os.Bundle;
import android.app.Activity;
import android.content.Intent;
import android.util.Log;
import android.view.*;
import android.view.View.*;
import android.widget.*;

public class MainActivity extends Activity implements OnClickListener {
    @Override
    protected void onCreate(Bundle b) {
        super.onCreate(b);
        setContentView(R.layout.activity_main);
        Button jbn = (Button) findViewById(R.id.xbn);
        jbn.setOnClickListener(this);
    }
    public void onClick(View v){
        enviarEmail();
    }
    protected void enviarEmail() {
        Log.i("Enviar email", "");
        String[] TO = {""};
        String[] CC = {""};
        Intent emailIntent = new Intent(Intent.ACTION_SEND);
        emailIntent.setData(Uri.parse("mailto:"));
        emailIntent.setType("text/plain");
        emailIntent.putExtra(Intent.EXTRA_EMAIL, TO);
        emailIntent.putExtra(Intent.EXTRA_CC, CC);
        emailIntent.putExtra(Intent.EXTRA_SUBJECT, "De ESCOM");
        emailIntent.putExtra(Intent.EXTRA_TEXT, "Correo desde Android Studio");
        try {
            startActivity(Intent.createChooser(emailIntent, "Send mail..."));
            finish();
            Log.i("Fin envío de correo...", ""); // Máximo 23 caracteres
        } catch (android.content.ActivityNotFoundException ex) {
            Toast.makeText(MainActivity.this, "No hay cliente de correo instalado.",
            Toast.LENGTH_SHORT).show();
        }
    }
}
```

2. Abrir el archivo `activity_main.xml` y agregar el siguiente código.

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical" >
    <TextView
        android:id="@+id/xtv1"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Ejemplo de envío de correo"
        android:layout_alignParentTop="true"
        android:layout_centerHorizontal="true" />
    <TextView
        android:id="@+id/xtv2"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
```



```
        android:text="ESCOM" />
<ImageButton
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:id="@+id/xib"
    android:src="@drawable/robot" />
<Button
    android:id="@+id/xbn"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:text="@string/compose_email"/>
```

3. En el archivo **strings.xml**, agregar las siguientes etiquetas **<string>**:

```
<?xml version="1.0" encoding="utf-8"?>
<resources>
    <string name="app_name">Mi Email</string>
    <string name="compose_email">Escribir Email</string>
</resources>
```

4. Agregar una imagen a la carpeta **drawable**; en este caso, el archivo **robot.png** con la imagen.

5. Ejecutar la aplicación y en la pantalla mostrada digitar el botón **ESCRIBIR EMAIL**. En la figura 2, en la parte inferior, se muestran las opciones hacia las cuales se desea enviar el correo. Seleccionar la opción más conveniente; por ejemplo, en este caso **Yahoo Mail**.



Figura 1. Pantalla de inicio.



Figura 2. Selección de correo.

6. Al abrir Yahoo, se ingresan los datos del destinatario en el campo **Para** y los datos que se consideren necesarios, como se indica en la figura 3.
7. En el móvil receptor, se indica la notificación de recepción del correo, como se indica en la figura 4.
8. Por último, se abre el correo enviado, como se indica en la figura 5.

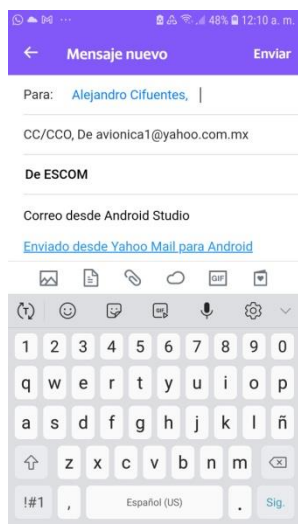


Figura 3. Envío de datos.

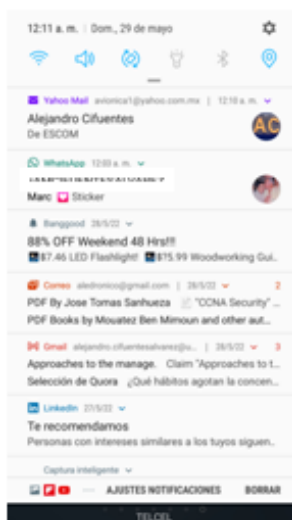


Figura 4. Notificación de correo.

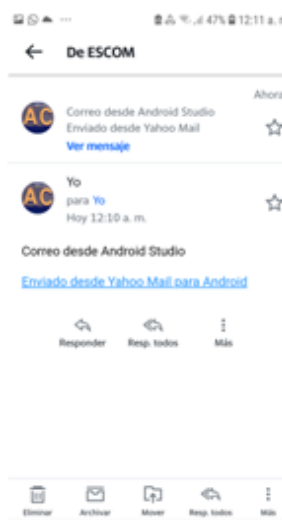


Figura 5. Lectura del correo.

EJERCICIO 1.

Modificar la aplicación anterior, agregando una base de datos **Agenda** que contenga los datos de contactos. Utilizar la agenda para seleccionar el nombre y el correo.

EJERCICIO 2.

Desarrollar la aplicación indicada en la página siguiente.

<https://soloelectronicos.com/2017/01/05/envio-de-e-mail-sin-intervencion-del-usuario-en-android-studio/>

Esta aplicación permite el envío de correo sin que se lo indique al usuario directamente, ya que la aplicación se encargará de ello. Documentar y ejecutar la aplicación mencionada en la página indicada.

EJERCICIO 3.

Utilizar la aplicación NFC para escribir en una etiqueta la información de un correo y, al leerla, envíe un mensaje al correo indicado en la etiqueta.

NOTA. Generar un reporte con todas las imágenes que documenten el desarrollo de las aplicaciones móviles. Guardar los archivos y el reporte en una carpeta comprimida con la sintaxis **AlumnoTarea37Grupo.zip** y enviarla al sitio indicado por el profesor.