Calling implicit intent

Manifest:

<?*xml version*="1.0" *encoding*="utf-8"?>  
<manifest *xmlns:android*="http://schemas.android.com/apk/res/android"  
 *xmlns:tools*="http://schemas.android.com/tools">  
  
 <uses-feature  
 *android:name*="android.hardware.telephony"  
 *android:required*="true" />  
 <uses-permission *android:name*="android.permission.CALL\_PHONE"/>  
 <application  
 *android:allowBackup*="true"  
 *android:dataExtractionRules*="@xml/data\_extraction\_rules"  
 *android:fullBackupContent*="@xml/backup\_rules"  
 *android:icon*="@mipmap/ic\_launcher"  
 *android:label*="@string/app\_name"  
 *android:roundIcon*="@mipmap/ic\_launcher\_round"  
 *android:supportsRtl*="true"  
 *android:theme*="@style/Theme.Calling"  
 *tools:targetApi*="31">  
 <activity  
 *android:name*=".MainActivity"  
 *android:exported*="true">  
 <intent-filter>  
 <action *android:name*="android.intent.action.MAIN" />  
  
 <category *android:name*="android.intent.category.LAUNCHER" />  
 </intent-filter>  
 </activity>  
 </application>  
  
</manifest>

Java:

*package* com.example.calling;  
  
*import* android.content.Intent;  
*import* android.net.Uri;  
*import* android.os.Bundle;  
*import* android.view.View;  
*import* android.widget.EditText;  
  
*import* androidx.activity.EdgeToEdge;  
*import* androidx.appcompat.app.AppCompatActivity;  
*import* androidx.core.graphics.Insets;  
*import* androidx.core.view.ViewCompat;  
*import* androidx.core.view.WindowInsetsCompat;  
  
*public class* MainActivity *extends* AppCompatActivity {  
 EditText editText;  
 @Override  
 *protected void* onCreate(Bundle savedInstanceState) {  
 *super*.onCreate(savedInstanceState);  
 EdgeToEdge.*enable*(*this*);  
 setContentView(R.layout.*activity\_main*);  
 ViewCompat.*setOnApplyWindowInsetsListener*(findViewById(R.id.*main*), (v, insets) -> {  
 Insets systemBars = insets.getInsets(WindowInsetsCompat.Type.*systemBars*());  
 v.setPadding(systemBars.left, systemBars.top, systemBars.right, systemBars.bottom);  
 *return* insets;  
 });  
 editText=findViewById(R.id.*editTextText*);  
 findViewById(R.id.*button*).setOnClickListener(*new* View.OnClickListener() {  
 @Override  
 *public void* onClick(View v) {  
 String phone\_number = editText.getText().toString();  
  
 *// Check if the phone number is not empty  
 if* (!phone\_number.isEmpty()) {  
 *// Prepend "tel:" to create a valid URI* Uri uri = Uri.*parse*("tel:" + phone\_number);  
  
 Intent intent = *new* Intent(Intent.ACTION\_CALL,uri);  
 *// Start the intent* startActivity(intent);  
  
  
  
 }  
 }  
 });  
 }  
}

XML:

<?*xml version*="1.0" *encoding*="utf-8"?>  
<androidx.constraintlayout.widget.ConstraintLayout *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"  
 *android:id*="@+id/main"  
 *android:layout\_width*="match\_parent"  
 *android:layout\_height*="match\_parent"  
 *tools:context*=".MainActivity">  
  
 <EditText  
 *android:id*="@+id/editTextText"  
 *android:layout\_width*="252dp"  
 *android:layout\_height*="0dp"  
 *android:layout\_marginTop*="146dp"  
 *android:layout\_marginBottom*="90dp"  
 *android:ems*="10"  
 *android:hint*="mo no"  
 *android:inputType*="text"  
 *app:layout\_constraintBottom\_toTopOf*="@+id/button"  
 *app:layout\_constraintEnd\_toEndOf*="parent"  
 *app:layout\_constraintStart\_toStartOf*="parent"  
 *app:layout\_constraintTop\_toTopOf*="parent" />  
  
 <Button  
 *android:id*="@+id/button"  
 *android:layout\_width*="wrap\_content"  
 *android:layout\_height*="wrap\_content"  
 *android:layout\_marginBottom*="369dp"  
 *android:text*="Button"  
 *app:layout\_constraintBottom\_toBottomOf*="parent"  
 *app:layout\_constraintEnd\_toEndOf*="parent"  
 *app:layout\_constraintStart\_toStartOf*="parent"  
 *app:layout\_constraintTop\_toBottomOf*="@+id/editTextText" />  
  
</androidx.constraintlayout.widget.ConstraintLayout>