**Виконавець:** Момот Роман КН-919а

**ЛАБОРАТОРНА РОБОТА 9**

**Надання послуг іншим застосункам**

***Мета:*** Освоїти навички роботи з локалізації для Андроїд програм.

***Варіант:*** 16

TaskNumber = (16 mod 10) + 1 = 7

***Індивідуальне завдання:*** Відобразити тип номеру телефону поряд з ним в списку контактів.

Створити новий проект з використанням шаблону Empty Activity, використати типові налаштування, задавши ім’я пакету проекту за правилами попередніх робіт. 2. Обробити відсутність з’єднання з інтернетом з відображенням Snackbar з можливістю відкриття системних налаштувань.

В класі MainActivity створити клас ConnectivityChangeReceiver та успадкувати його від BroadcastReceiver. Додати функцію по обробці зміни стану підключеності до інтернету. Обробити відповідне широкомовне повідомлення в створеному класі-приймачі.

Визначити функцію по створенню Snackbar з однією дією, який буде відображати інформацію про відсутність з’єднання з інтернетом та надавати можливість користувачеві відкрити системні налаштування.

Реалізувати функцію відображення та приховування об’єкту Snackbar, що було оголошено раніше. Додати до класу MainActivity поле для зберігання цього об’єкту.

Реалізувати функцію відкриття системних налаштувань.

Відобразити список контактів з номерами телефонів з можливістю зателефонувати обраному контакту.

Реалізувати функцію отримання списку контактів.

Створити адаптер для ListView, заповнити його прочитаними контактами.

Додати відповідний дозвіл для доступу до контактів.

Обробити натиснення на елемент списку. При натисненні викликати системний екран набору номеру, після чого користувач самостійно натисне на кнопку виклику.

**Хід виконнаня роботи**

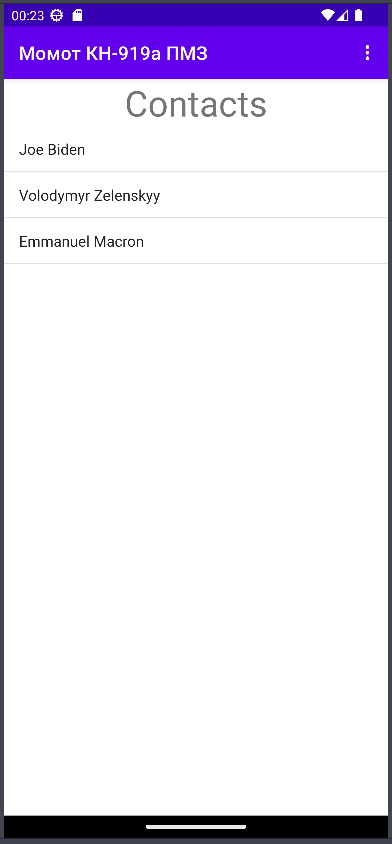
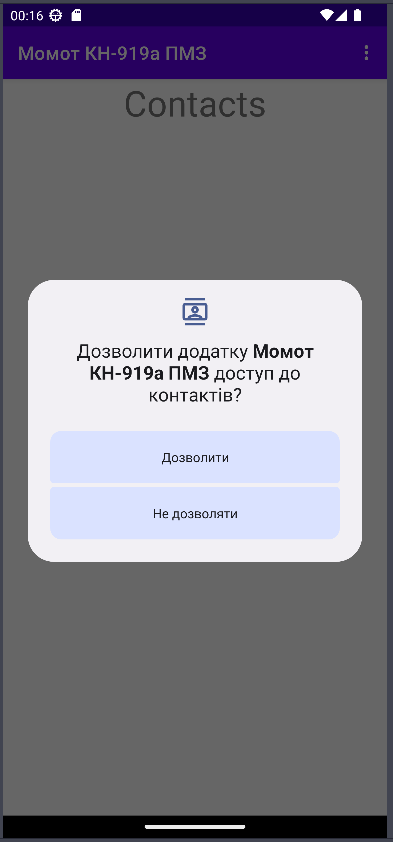
1. Створюємо клас, якій виводить на екран список контактів та їх номери телефонів. При активації цього класу додаток запрошує дозвіл на перевірку списку контактів, що свідчить про те, що додаток дійсно працює зі списком контактів (рис. 9.1).

Рисунок 9.1. Перше відкриття додатку

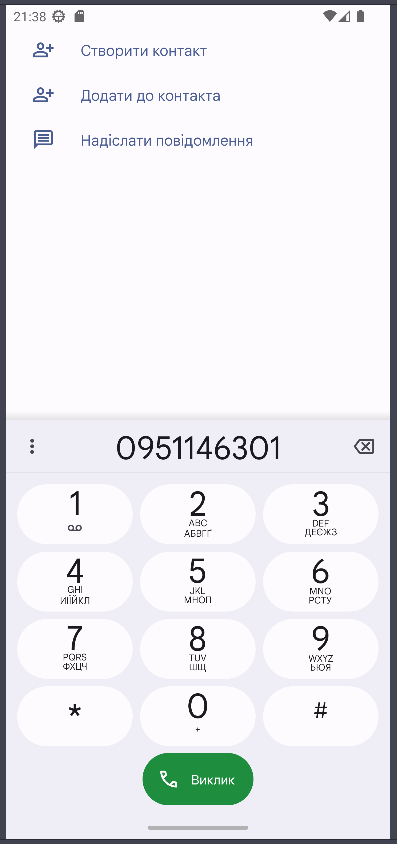
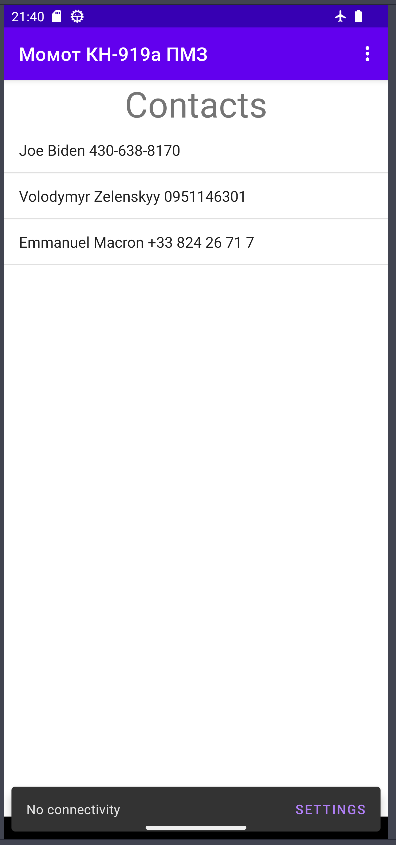
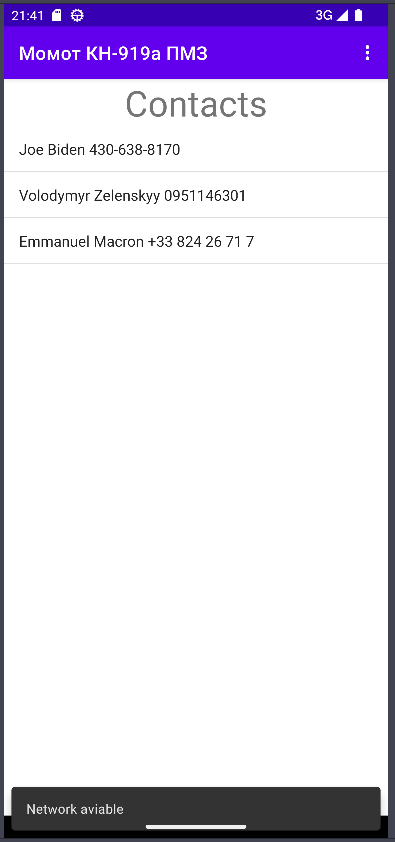
1. При натисканні на контакт відкривається меню набору номеру телефона. Також клас реагує на зміну статусу мережі (рис. 9.2).

Рисунок 9.2. Відкриття меню набору номеру

1. При включенні або відключенні пристрою від мережі виводиться повідомлення на екран про це, а також є кнопка для відкриття налаштувань мережі (рис 9.3).

 Рисунок 9.3. Виведення повідомлення про зміну статусу мережі

**Код програми**

**MainActivity.java**

package com.ntukhpi.otp.momot.fourth\_course\_dma.lab9;

import android.Manifest;

import android.annotation.SuppressLint;

import android.content.ContentResolver;

import android.content.Context;

import android.content.Intent;

import android.content.pm.PackageManager;

import android.database.Cursor;

import android.net.ConnectivityManager;

import android.net.Network;

import android.net.Uri;

import android.os.Bundle;

import android.provider.ContactsContract;

import static android.provider.ContactsContract.Contacts.DISPLAY\_NAME\_PRIMARY;

import android.provider.Settings;

import android.widget.ArrayAdapter;

import android.widget.ListView;

import android.widget.Toast;

import androidx.core.app.ActivityCompat;

import androidx.core.content.ContextCompat;

import com.google.android.material.snackbar.Snackbar;

import com.ntukhpi.otp.momot.fourth\_course\_dma.Menu;

import com.ntukhpi.otp.momot.fourth\_course\_dma.lab1.R;

import java.util.ArrayList;

public class MainActivity extends Menu {

private final ArrayList<Contact> contacts = new ArrayList<>();

private boolean isReadingContactsAllowed = false;

@Override

protected void onCreate(Bundle savedInstanceState) {

super.onCreate(savedInstanceState);

setContentView(R.layout.lab9\_activity\_main);

createInternetConnectionListener();

readContactsTask();

}

public void openNetworkActivity() {

Intent intent = new Intent(Settings.ACTION\_WIRELESS\_SETTINGS);

startActivity(intent);

}

private void createInternetConnectionListener() {

ConnectivityManager.NetworkCallback networkCallback = new ConnectivityManager.NetworkCallback() {

@Override

public void onAvailable(Network network) {

Snackbar.make(getWindow().getDecorView().getRootView(), "Network aviable", Snackbar.LENGTH\_LONG)

.show();

}

@Override

public void onLost(Network network) {

Snackbar.make(getWindow().getDecorView(), "No connectivity", Snackbar.LENGTH\_INDEFINITE)

.setAction("SETTINGS", v -> openNetworkActivity())

.show();

}

};

ConnectivityManager connectivityManager =

(ConnectivityManager) this.getSystemService(Context.CONNECTIVITY\_SERVICE);

connectivityManager.registerDefaultNetworkCallback(networkCallback);

}

private void readContactsTask() {

int hasReadContactPermission = ContextCompat.checkSelfPermission(this, Manifest.permission.READ\_CONTACTS);

if (hasReadContactPermission == PackageManager.PERMISSION\_GRANTED) {

isReadingContactsAllowed = true;

} else {

ActivityCompat.requestPermissions(this, new String[]{Manifest.permission.READ\_CONTACTS}, 1);

}

if (isReadingContactsAllowed) {

getContacts();

}

}

@Override

public void onRequestPermissionsResult(int requestCode, String[] permissions, int[] grantResults) {

super.onRequestPermissionsResult(requestCode, permissions, grantResults);

if (requestCode == 1 && (grantResults.length > 0 && grantResults[0] == PackageManager.PERMISSION\_GRANTED)) {

isReadingContactsAllowed = true;

}

if (isReadingContactsAllowed) {

getContacts();

} else {

Toast.makeText(this, "You need to allow reading contacts", Toast.LENGTH\_LONG).show();

}

}

@SuppressLint("Range")

private void getContacts() {

ContentResolver contentResolver = getContentResolver();

Cursor cursor = contentResolver.query(ContactsContract.Contacts.CONTENT\_URI,

null, null, null, null);

if(cursor != null) {

while(cursor.moveToNext()) {

String id = cursor.getString(cursor.getColumnIndexOrThrow(ContactsContract.Contacts.\_ID));

String hasPhone =

cursor.getString(cursor.getColumnIndex(ContactsContract.Contacts.HAS\_PHONE\_NUMBER));

if (hasPhone.equalsIgnoreCase("1")) {

Cursor phones = getContentResolver().query(ContactsContract.CommonDataKinds.Phone.CONTENT\_URI, null,

ContactsContract.CommonDataKinds.Phone.CONTACT\_ID + " = " + id, null, null);

phones.moveToFirst();

String contactNumber = phones.getString(phones.getColumnIndex(ContactsContract.CommonDataKinds.Phone.NUMBER));

String contactName = cursor.getString(cursor.getColumnIndex(DISPLAY\_NAME\_PRIMARY));

Contact contact = new Contact(contactName, contactNumber);

contacts.add(contact);

}

}

}

outputDataToList();

}

private void outputDataToList() {

ArrayAdapter<Contact> adapter = new ArrayAdapter<>(this,

android.R.layout.simple\_list\_item\_1, contacts);

ListView contactList = findViewById(R.id.lab9\_contact\_list);

contactList.setAdapter(adapter);

contactList.setOnItemClickListener((parent, view, position, id) -> {

openPhoneDial(contacts.get(position).getPhone());

Snackbar.make(getWindow().getDecorView().getRootView(), contacts.get(position).getPhone(), Snackbar.LENGTH\_LONG)

.show();

});

}

private void openPhoneDial(String phoneNumber) {

Intent intent = new Intent(Intent.ACTION\_DIAL);

intent.setData(Uri.parse("tel:" + phoneNumber));

startActivity(intent);

}

}

**Contact.java**

package com.ntukhpi.otp.momot.fourth\_course\_dma.lab9;public class Contact {  
 private String name;  
 private String phone;  
  
 public Contact(String name, String phone) {  
 this.name = name;  
 this.phone = phone;  
 }  
  
 public String getName() {  
 return name;  
 }  
  
 public void setName(String name) {  
 this.name = name;  
 }  
  
 public String getPhone() {  
 return phone;  
 }  
  
 public void setPhone(String phone) {  
 this.phone = phone;  
 }  
  
 @Override  
 public String toString() {  
 return name + " " + phone;  
 }  
}

**Lab9\_activity\_man.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:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"  
 tools:context="com.ntukhpi.otp.momot.fourth\_course\_dma.lab9.MainActivity">  
  
 <TextView  
 android:id="@+id/lab9\_header"  
 android:layout\_width="0dp"  
 android:layout\_height="wrap\_content"  
 android:gravity="center"  
 android:text="@string/lab9\_contacts"  
 android:textSize="38sp"  
 app:layout\_constraintBottom\_toTopOf="@id/lab9\_contact\_list"  
 app:layout\_constraintHorizontal\_bias="0.0"  
 app:layout\_constraintLeft\_toLeftOf="parent"  
 app:layout\_constraintRight\_toRightOf="parent"  
 app:layout\_constraintTop\_toTopOf="parent" />  
  
 <ListView  
 android:id="@+id/lab9\_contact\_list"  
 android:layout\_width="0dp"  
 android:layout\_height="0dp"  
 app:layout\_constraintBottom\_toBottomOf="parent"  
 app:layout\_constraintLeft\_toLeftOf="parent"  
 app:layout\_constraintRight\_toRightOf="parent"  
 app:layout\_constraintTop\_toBottomOf="@+id/lab9\_header" />  
</androidx.constraintlayout.widget.ConstraintLayout>

**Висновок**

У результаті виконання лабораторної роботи було ознайомлено з роботою зі списком контактів в телефоні. Був створений додаток, який виводить назву контакту та його номер телефону на екран, реагує на зміну мережі, а також відкриває налаштування телефону.