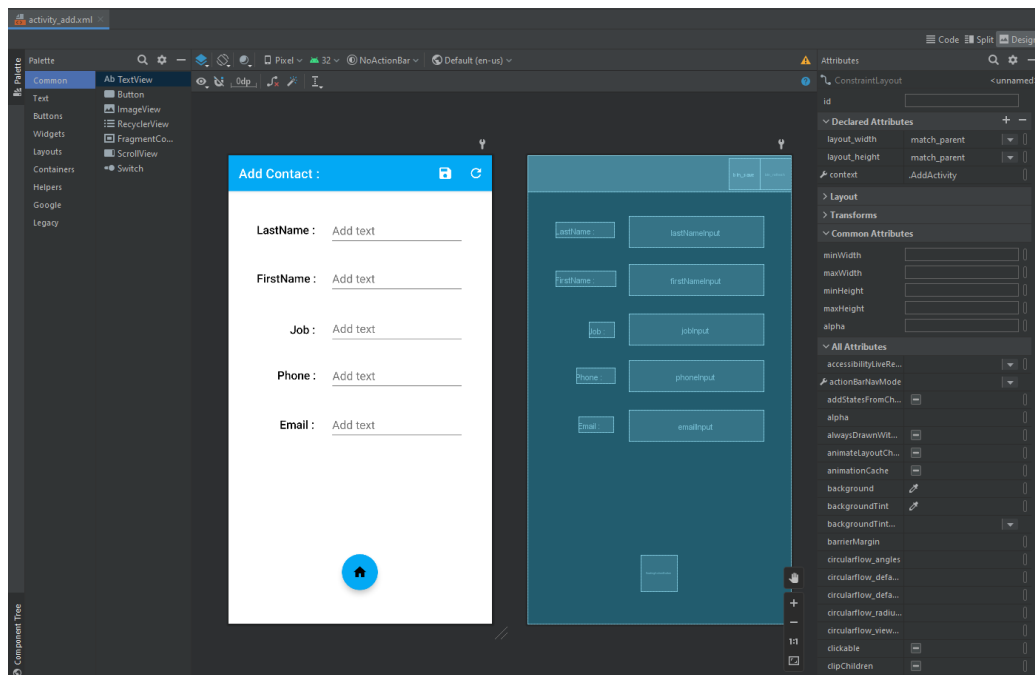


Compte Rendu

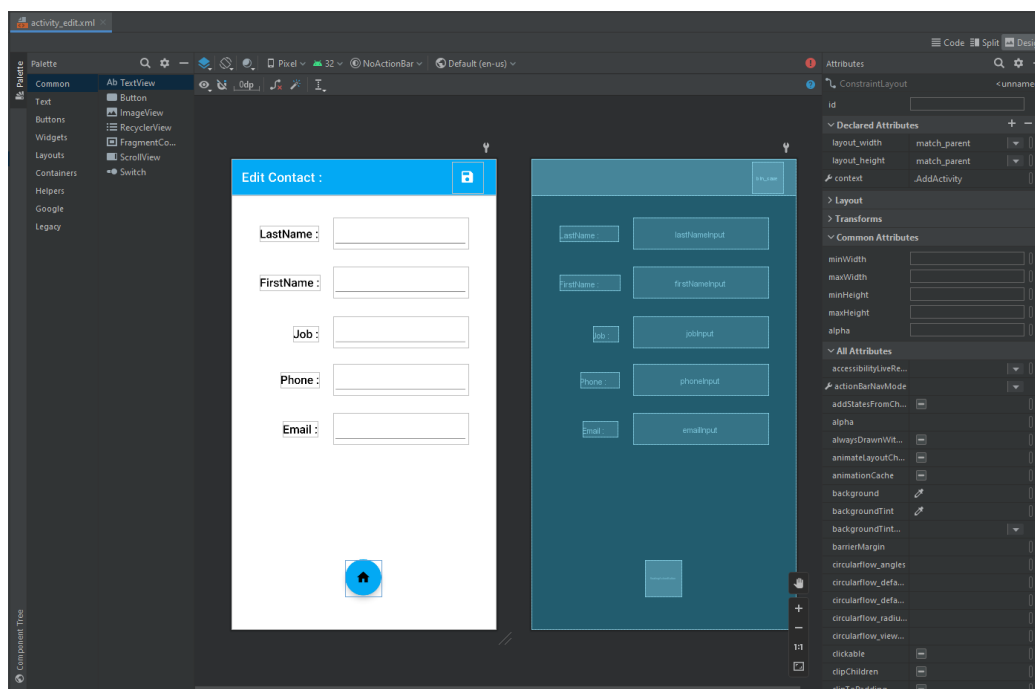
TP Annuaire Professionnel

Interfaces Graphiques :

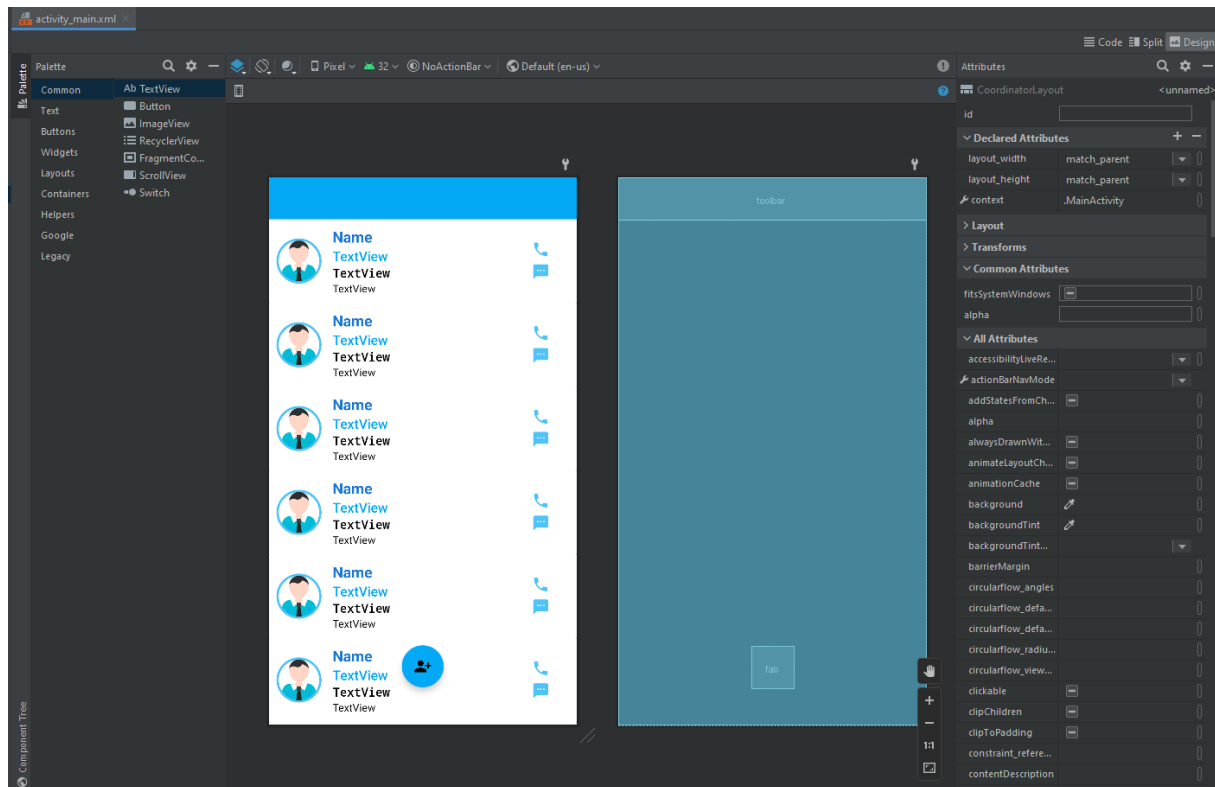
➤ Layout **activity_add** :



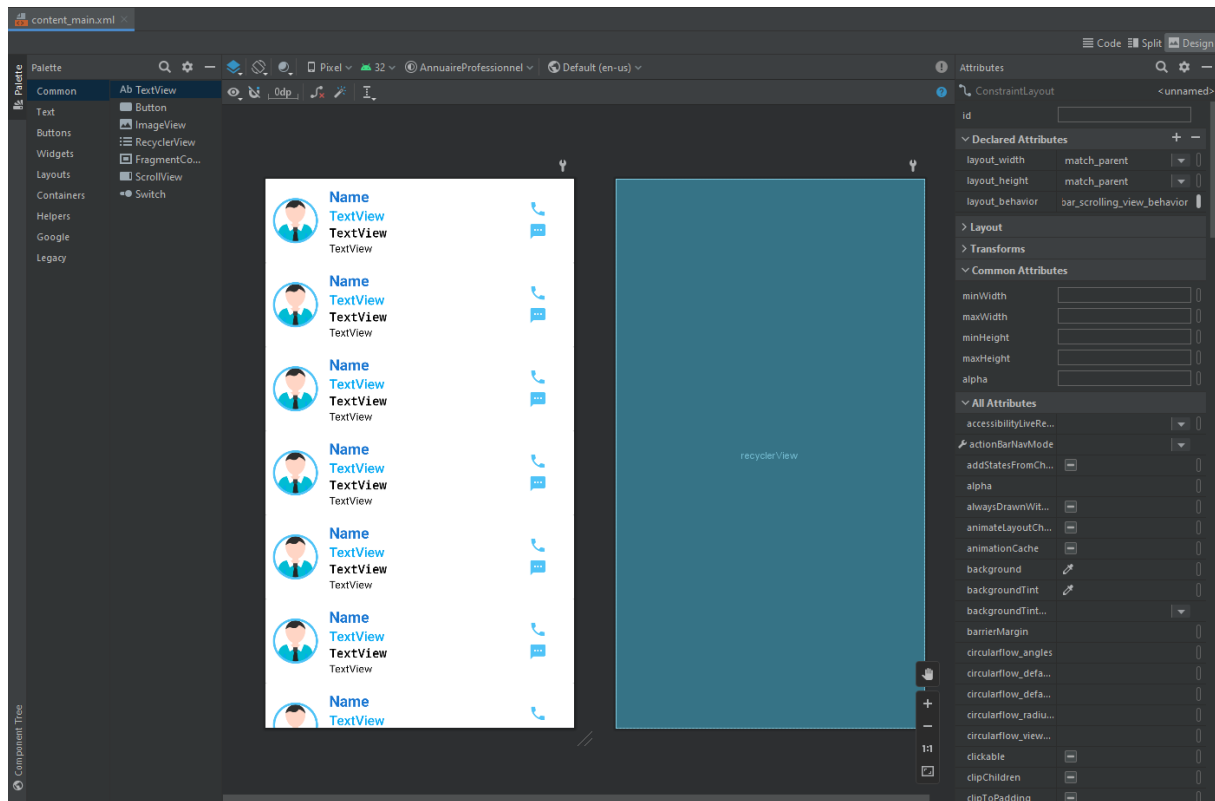
➤ Layout **activity_edit** :



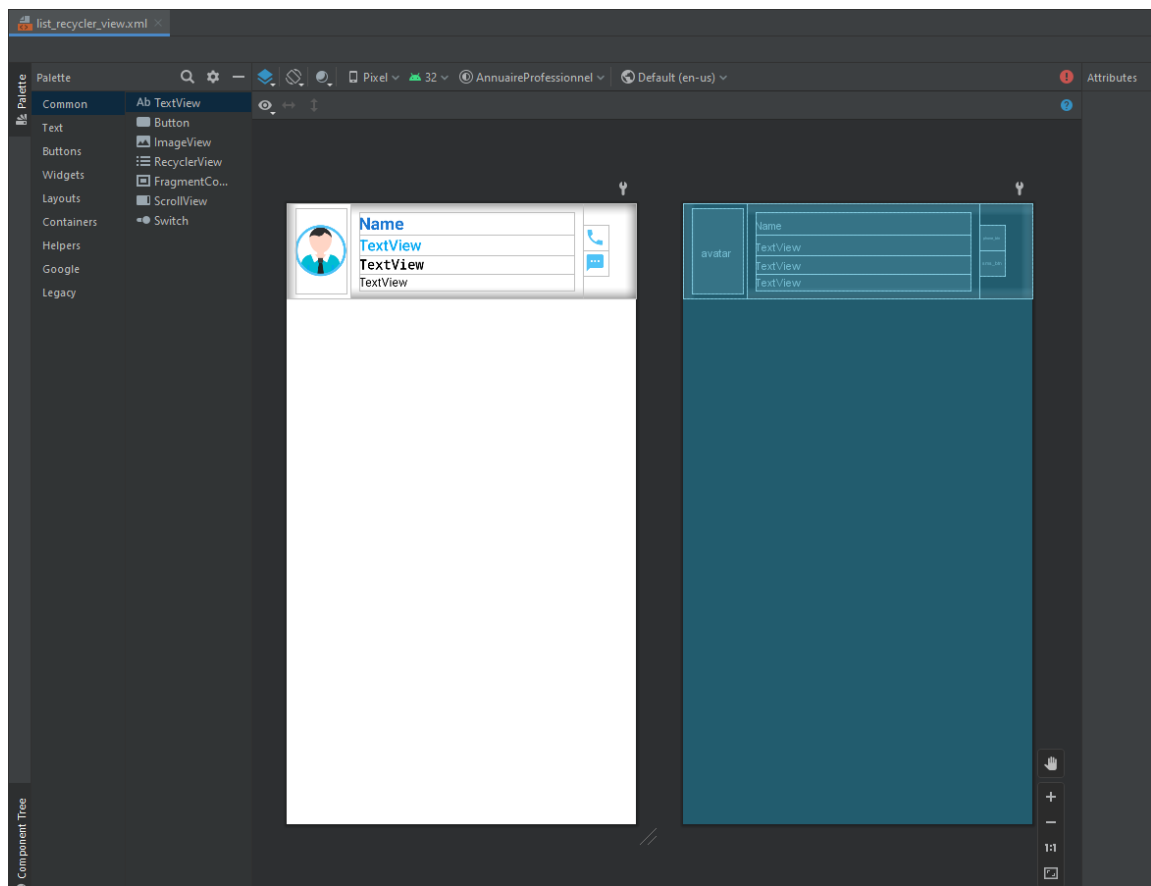
► Layout **activity_main** :



► Layout **activity_detail** :



➤ Layout `list_recycler_view` :



Code :

➤ AppDataBase :

```
AppDataBase.java
1 package ma.enset.annuaire;
2
3
4 import android.content.Context;
5 import androidx.room.Database;
6 import androidx.room.Room;
7 import androidx.room.RoomDatabase;
8
9 @Database(entities = {Contact.class}, version = 1, exportSchema = false)
10 public abstract class AppDataBase extends RoomDatabase {
11     private static AppDataBase database;
12
13     private static String DATABASE_NAME = "db";
14
15     public synchronized static AppDataBase getInstance(Context context) {
16         if(database == null) {
17             database = Room.databaseBuilder(context.getApplicationContext(), AppDataBase.class, DATABASE_NAME)
18                 .allowMainThreadQueries().fallbackToDestructiveMigration().build();
19         }
20         return database;
21     }
22
23     public abstract ContactDAO contactDAO();
24 }
25
```

➤ Contact :

```
1 package ma.enset.annuaire;
2
3 import ...
4
5
6
7
8
9
10 @Entity
11 public class Contact implements Serializable {
12
13     @PrimaryKey(autoGenerate = true)
14     private int ID;
15     @ColumnInfo(name = "LastName")
16     String lastName;
17     @ColumnInfo(name = "FirstName")
18     String firstName;
19     @ColumnInfo(name = "Job")
20     String job;
21     @ColumnInfo(name = "Phone")
22     String phone;
23     @ColumnInfo(name = "Email")
24     String email;
25     @Ignore
26     public Contact() {}
27
28
29     public Contact(Integer ID, String lastName, String firstName, String job, String phone, String email) {
30         this.ID = ID;
31         this.lastName = lastName;
32         this.firstName = firstName;
33         this.job = job;
34         this.phone = phone;
35         this.email = email;
36     }
37     @Ignore
38     public Contact(String lastName, String firstName, String job, String phone, String email) {
39         this.lastName = lastName;
40         this.firstName = firstName;
41         this.job = job;
42         this.phone = phone;
43         this.email = email;
44     }
45
46     public int getID() { return ID; }
47     public void setID(int ID) { this.ID = ID; }
```

➤ ContactDAO :

```
1 package ma.enset.annuaire;
2
3 import ...
4
5
6
7
8
9
10 @Dao
11 public interface ContactDAO {
12
13     @Query("SELECT * FROM Contact")
14     List<Contact> getAll();
15
16     @Insert
17     void insert(Contact contact);
18
19     @Query("SELECT * FROM Contact WHERE FirstName LIKE :keyword OR LastName LIKE :keyword")
20     List<Contact> findByName(String keyword);
21
22     @Query("SELECT * FROM Contact WHERE ID= :ID")
23     Contact findById(int ID);
24
25     @Query("DELETE FROM Contact WHERE ID= :ID")
26     void delete(int ID);
27
28     @Update
29     void update(Contact contact);
30
31     @Query("DELETE FROM Contact")
32     void clear();
33 }
```

➤ ContactsAdapter :

```
ContactsAdapter.java
1 package ma.enset.annuaire;
2
3 import ...
4
17
18 public class ContactsAdapter extends RecyclerView.Adapter<ContactsAdapter.ViewHolder> {
19
20     private List<Contact> contactsList;
21     private Context context;
22     private AppDataBase database;
23     private SelectedContact selectedContact;
24     private MainActivity mainActivity;
25     private Context ctx;
26
27     public ContactsAdapter(List<Contact> contactsList, Activity context, SelectedContact selectedContact, MainActivity mainActivity) {
28         this.contactsList = contactsList;
29         this.mainActivity=mainActivity;
30         this.selectedContact=selectedContact;
31         this.context = context;
32         notifyDataSetChanged();
33     }
34
35     @NonNull
36     @Override
37     public ViewHolder onCreateViewHolder(@NonNull ViewGroup parent, int viewType) {
38         ctx=parent.getContext();
39         View view= LayoutInflater.from(parent.getContext()).inflate(R.layout.list_recycler_view,parent, attachToRoot: false);
40         return new ViewHolder(view,ctx);
41     }
42
43     @NonNull
44     @Override
45     public void onBindViewHolder(@NonNull ViewHolder holder, int position) {
46         Contact contact=contactsList.get(position);
47         database=AppDataBase.getInstance(context);
48
49         holder.id.setText(String.valueOf(contact.getID()));
50         holder.name.setText(contact.getFirstName()+" "+contact.getLastName());
51         holder.job.setText(contact.getJob());
52         holder.phone.setText(contact.getPhone());
53         holder.email.setText(contact.getEmail());
54     }
55 }
```

```
56
57 @Override
58 public int getItemCount() { return contactsList.size(); }
59
60
61
62
63 public class ViewHolder extends RecyclerView.ViewHolder implements View.OnLongClickListener {
64     TextView id, name, email, job, phone;
65     ImageButton btCall;
66     Context context;
67     public ViewHolder(@NonNull View itemView, Context context) {
68         super(itemView);
69         id=itemView.findViewById(R.id.ID);
70         name=itemView.findViewById(R.id.Name);
71         job=itemView.findViewById(R.id.Job);
72         phone=itemView.findViewById(R.id.Tel);
73         email=itemView.findViewById(R.id.Email);
74         btCall=itemView.findViewById(R.id.phone_btn);
75         this.context=context;
76         itemView.setOnLongClickListener(this::onLongClick);
77         itemView.setOnClickListener(new View.OnClickListener() {
78             @Override
79             public void onClick(View v) {
80                 selectedContact.selectedContact(contactsList.get(getAdapterPosition()));
81             }
82         });
83     }
84
85     @Override
86     public boolean onLongClick(View view) {
87         AlertDialog.Builder builder = new AlertDialog.Builder(ctx);
88         String contactName = contactsList.get(getAdapterPosition()).getFirstName()
89             + " "
90             + contactsList.get(getAdapterPosition()).getLastName();
91         builder.setTitle("Confirmation :")
92             .setMessage("You want to delete "+contactName+ " from contacts ?")
93             .setCancelable(false)
94             .setPositiveButton(text "Yes", new DialogInterface.OnClickListener() {
95                 @Override
96                 public void onClick(DialogInterface dialog, int which) {
97                     int idContact = Integer.parseInt(id.getText().toString());
98                     Contact contactToDelete = new Contact();
99                     int contactPos = getAdapterPosition();
100                     contactToDelete.setID(idContact);
```

➤ AddActivity :

```
11 public class AddActivity extends AppCompatActivity {
12
13     AppDataBase database;
14     EditText lastName;
15     EditText firstName;
16     EditText job;
17     EditText phone;
18     EditText email;
19
20     @Override
21     protected void onCreate(Bundle savedInstanceState) {
22         database=AppDataBase.getInstance(this);
23         super.onCreate(savedInstanceState);
24         setContentView(R.layout.activity_add);
25         lastName = (EditText) findViewById(R.id.lastNameInput);
26         firstName = (EditText) findViewById(R.id.firstNameInput);
27         job = (EditText) findViewById(R.id.jobInput);
28         phone = (EditText) findViewById(R.id.phoneInput);
29         email = (EditText) findViewById(R.id.emailInput);
30         FloatingActionButton fab = findViewById(R.id.floatingActionButton);
31         fab.setOnClickListener(new View.OnClickListener() {
32             @Override
33             public void onClick(View view) {
34                 Intent myIntent = new Intent( packageContext AddActivity.this, MainActivity.class);
35                 startActivity(myIntent);
36             }
37         });
38     }
39     public void save(View view){
40
41         database.contactDAO().insert(new Contact(lastName.getText().toString(), firstName.getText().toString(), job.getText().toString(), phone.getText().toString(), email.getText().toString()));
42         Toast.makeText( context: AddActivity.this, text: "Contact added successfully!", Toast.LENGTH_SHORT).show();
43         Intent myIntent = new Intent( packageContext: AddActivity.this, MainActivity.class);
44         startActivity(myIntent);
45     }
46     public void clear(View view)
47     {
48         lastName.setText("");
49         firstName.setText("");
50         job.setText("");
51         phone.setText("");
52     }
53 }
```

➤ EditActivity :

```
25 protected void onCreate(Bundle savedInstanceState) {
26     super.onCreate(savedInstanceState);
27
28     database=AppDataBase.getInstance(this);
29
30     setContentView(R.layout.activity_edit);
31
32     lastName = (EditText) findViewById(R.id.lastNameInput);
33     firstName = (EditText) findViewById(R.id.firstNameInput);
34     job = (EditText) findViewById(R.id.jobInput);
35     phone = (EditText) findViewById(R.id.phoneInput);
36     email = (EditText) findViewById(R.id.emailInput);
37
38     Intent i = getIntent();
39     if(i.hasExtra( name: "ID")){
40         string = i.getStringExtra( name: "ID");
41     }
42     Contact contact = database.contactDAO().findById(Integer.parseInt(string));
43
44     lastName.setText(contact.getLastName());
45     firstName.setText(contact.getFirstName());
46     job.setText(contact.getJob());
47     phone.setText(contact.getPhone());
48     email.setText(contact.getEmail());
49     FloatingActionButton fab = findViewById(R.id.floatingActionButton);
50     fab.setOnClickListener(new View.OnClickListener() {
51         @Override
52         public void onClick(View view) {
53             Intent myIntent = new Intent( packageContext: EditActivity.this, MainActivity.class);
54             startActivity(myIntent);
55         }
56     });
57 }
58
59 public void save(View view){
60
61     database.contactDAO().update(new Contact(Integer.parseInt(string), lastName.getText().toString(), firstName.getText().toString(), job.getText().toString(), phone.getText().toString()));
62     Toast.makeText( context: EditActivity.this, text: "Contact updated successfully!", Toast.LENGTH_SHORT).show();
63 }
```

➤ MainActivity :

```
MainActivity.java
1 package ma.enset.annuaire;
2
3 import ...
4
16
17 public class MainActivity extends AppCompatActivity implements ContactsAdapter.SelectedContact{
18     RecyclerView recyclerView;
19     AppDataBase db;
20     List<Contact> contacts;
21     LinearLayoutManager layoutManager;
22
23     Contact contact1=new Contact( lastName: "HACHMI", firstName: "Mohamed Amine", job: "Software engineer", phone: "0612345678", email: "aminehcm0@gmail.com");
24
25     @Override
26     protected void onCreate(Bundle savedInstanceState) {
27
28         super.onCreate(savedInstanceState);
29         setContentView(R.layout.activity_main);
30         Toolbar toolbar = findViewById(R.id.toolbar);
31         setSupportActionBar(toolbar);
32
33         db = AppDataBase.getInstance(this);
34         //db.contactDAO().insert(contact1);
35
36         contacts= db.contactDAO().getAll();
37         ContactsAdapter adapter = new ContactsAdapter(contacts, context: this, selectedContact: this, MainActivity: this);
38
39         recyclerView = (RecyclerView) findViewById(R.id.recyclerView);
40         layoutManager=new LinearLayoutManager( context: this);
41         recyclerView.setLayoutManager(layoutManager);
42         recyclerView.setAdapter(adapter);
43         contacts.clear();
44         contacts.addAll(db.contactDAO().getAll());
45         adapter.notifyDataSetChanged();
46
47         FloatingActionButton fab = findViewById(R.id.fab);
48         fab.setOnClickListener(new View.OnClickListener() {
49             @Override
50             public void onClick(View view) {
51                 Intent myIntent = new Intent( packageContext: MainActivity.this, AddActivity.class);
52                 startActivity(myIntent);
53             }
54         });
55     }
56 }
```

```
55
56
57 @Override
58 public boolean onCreateOptionsMenu(Menu menu) {
59     getMenuInflater().inflate(R.menu.menu_main, menu);
60     MenuItem searchItem = menu.findItem(R.id.search);
61
62     SearchView searchView = (SearchView) searchItem.getActionView();
63     searchView.setOnQueryTextListener(new SearchView.OnQueryTextListener() {
64         @Override
65         public boolean onQueryTextSubmit(String query) { return false; }
66
67         @Override
68         public boolean onQueryTextChange(String textQuery) {
69             RefreshListView((ArrayList<Contact>) db.contactDAO().findByName( keyword: "%"+textQuery+"%"));
70             return false;
71         }
72     });
73
74     return true;
75 }
76
77
78
79 @Override
80 public boolean onOptionsItemSelected(MenuItem item) {
81     int id = item.getItemId();
82     return super.onOptionsItemSelected(item);
83 }
84
85 @Override
86 public void selectedContact(Contact contact) {
87     Intent myIntent = new Intent( packageContext: MainActivity.this, EditActivity.class);
88     myIntent.putExtra( name: "ID",String.valueOf(contact.getID()));
89     startActivity(myIntent);
90 }
91
92 void RefreshListView(List<Contact> contacts){
93     ContactsAdapter adapter = new ContactsAdapter(contacts, context: this, selectedContact: this, MainActivity: this);
94     recyclerView.setAdapter(adapter);
95 }
96 }
```

Exécution :

