

1.主题概述

1.

修改E 3中的LoginOrRegister代码， 当导航到该页面时， 使用SharedPreferences保存或读取用户登录状态。

2.

修改LoginOrRegister代码， 当用户点击登录&注册时， 按E3中登录规则判断是否通过， 如果登录成功则将当前登录名输入框的值写入到SharedPreferences中的userName记录中， 另外再写入loginState值为1。当用户点击返回时， 判断SharedPreferences中是否有用户名记录， 如无， 则保持原功能不变；否则在SharedPreferences写入loginState值为0。

3.

用户是否登录的判断逻辑改成使用现SharedPreferences包中， es.source.code.activity的loginState值进行功能实。

4.

新建SCOSHelper.java类， 设置RelativeLayout布局， 显示， 用户使用协议， 关于系统， 电话人工帮助， 短信帮助， 邮件帮助， 当用户点击“电话人工帮助”时， 使用 ImplicitIntent 调用系统 app 实现自动拨号功能， 拨打目标号码为“5554”；当用户点击“短信帮助”时， 使用 SmsManager 自动发送短信只目 标号码“5554”， 内容为“test scos helper”。当短信发送完毕后， 使用 Toast 提示“求助短信发送成功”；当用户点击“邮件帮助”时， 启动自定义多线程 MailSender 完成 邮件发送， 邮件内容和标题自定义。当邮件发送完毕后， 使用 Handler 接收消息， 当消息值为 1 时， 使用 Toast 提示“求助邮件已发送成功”。

5.

修改E3中FoodOrderView用户点击结账按钮时， 启动代码， 使用AsyncTask模拟结账功能。当AsyncTask， 并使用ProgressBar实时显示结账进度， 要求AsyncTask运行6秒后模拟完成结账功能。

2.步骤

修改LigionOrRegister->修改MainScreen和SCOSEntry->添加SCOSHelper->修改FoodOrderView。

3.实现

1.添加SharedPreferences代码

```
1  //LoginOrRegister
2  /*
3   使用SharedPreferences进行用户登录状态判断，当状态为0，设置登录按钮不可见，当为
4   1，设置注册按钮不可见。
5   */
6  protected void onCreate(Bundle savedInstanceState) {
7      super.onCreate(savedInstanceState);
8      setContentView(R.layout.loginorregister);
9      mButton_Enter = (Button)findViewById(R.id.enterButton);
10     mButton_Login = (Button)findViewById(R.id.loginButton);
11     mButton_Return = (Button)findViewById(R.id.returnButton);
12     etName = (EditText)findViewById(R.id.userName);
13     etPassword = (EditText)findViewById(R.id.userPassword);
14     setListeners();
15
16     SharedPreferences sharedPreferences =
17         this.getSharedPreferences("User",MODE_PRIVATE);
18     int loginState = sharedPreferences.getInt("loginState",0);
19     if(loginState == 0){
20         mButton_Enter.setVisibility(View.INVISIBLE);
21     }
22     else{
23         mButton_Login.setVisibility(View.VISIBLE);
24     }
25 }
```

2.修改LoginOrRegister代码

当用户点击登录时，如果登录&注册成功则将当前登录名输入框的值写入到SharedPreferences中的userName记录中，另外再写入loginState值为1。当用户点击返回时，判断SharedPreferences中是否有用户名记录，如无，则保持原功能不变；否则在SharedPreferences写入loginState值为0。

```
1  //LoginOrRegister
```

```

2 private class OnClick implements View.OnClickListener{
3
4     @Override
5     public void onClick(View view){
6
7         final User loginUser = new User();
8         Intent intent = null;
9         final SharedPreferences.Editor editor =
10         getSharedPreferences("User",MODE_PRIVATE).edit();
11         Thread thread = null;
12
13         switch (view.getId()){
14
15             case R.id.loginButton:
16                 showProgress(R.string.registering);
17
18                 thread = new Thread(new Runnable() {
19                     @Override
20                     public void run() {
21
22                         //sendJson();
23                         String url =
24                         MyPOSTHttpURLConnection.BASE_URL + "/LoginValidator";
25                         Map<String,String > parms = new
26                         HashMap<String, String>();
27                         String name =
28                         etName.getText().toString();
29                         String password =
30                         etPassword.getText().toString();
31                         parms.put("name",name);
32                         parms.put("password",password);
33
34                         String result =
35                         MyPOSTHttpURLConnection.getContextByHttp(url,parms);
36                         Log.d("result",result);
37
38                         Message msg = new Message();
39                         msg.what = 20;
40                         Bundle data = new Bundle();
41
42                         data.putString("result",result);
43                         msg.setData(data);
44
45                     }
46                 });
47                 thread.start();
48             }
49     }
50 }

```

```

41         handler.sendMessage(msg);
42     }
43
44
45     Handler handler = new Handler(){
46         @Override
47         public void handleMessage(Message msg) {
48             if(msg.what==20){
49                 Bundle data = msg.getData();
50                 String key =
data.getString("result");
51                 Log.d("key",key);
52                 try{
53                     JSONObject json = new
JSONObject(key);
54                     String result =
(String)json.get("result");
55                     if(result.equals("success")){
56
57                         mButton_Enter.setVisibility(View.VISIBLE);
58
59                         mButton_Login.setVisibility(View.INVISIBLE);
60
61                         loginUser.setOldUser(false);
62
63                         loginUser.setUserName(etName.getText().toString());
64
65                         loginUser.setPassword(etPassword.getText().toString());
66
67                         Toast.makeText(LoginOrRegister.this,"欢迎您成为 SCOS 新用户",Toast.LENGTH_SHORT).show();
68
69                                     //设置SharedPreferences
70                                     String string =
mButton_Enter.getText().toString();
71
72                                     editor.putString("userName",string);
73
74                                     editor.putInt("loginState",1);
75
76                                     editor.commit();
77
78                                     Toast.makeText(LoginOrRegister.this,"注册成功",Toast.LENGTH_LONG).show();
79
80                                     }
81                                 else{

```

```

71     Toast.makeText(LoginOrRegister.this, "输入内容不符合规
72     则", Toast.LENGTH_SHORT).show();
73
74
75
76     }
77     catch (Exception e){
78         e.printStackTrace();
79     }
80 }
81 }
82 };
83
84
85 });
86
87     thread.start();
88
89     dismissProgress();
90     break;
91     case R.id.enterButton:
92         showProgress(R.string.logining);
93
94         thread = new Thread(new Runnable() {
95             @Override
96             public void run() {
97
98                 //sendJson();
99                 String url =
100 MyPOSTHttpURLConnection.BASE_URL + "/LoginValidator";
101                 Map<String,String> parms = new
102 HashMap<String, String>();
103                 String name =
104 etName.getText().toString();
105                 String password =
106 etPassword.getText().toString();
107                 parms.put("name",name);
108                 parms.put("password",password);
109                 String result =
110 MyPOSTHttpURLConnection.getContextByHttp(url,parms);
111                 Log.d("result",result);
112                 Message msg = new Message();
113                 msg.what = 21;

```

```

109         Bundle data = new Bundle();
110         data.putString("result",result);
111         msg.setData(data);
112
113         handler.sendMessage(msg);
114     }
115     //使用Handler处理消息
116     Handler handler = new Handler(){
117         @Override
118         public void handleMessage(Message msg) {
119             if(msg.what==21){
120                 Bundle data = msg.getData();
121                 String key =
data.getString("result");
122                 Log.d("key",key);
123                 try{
124                     JSONObject json = new
JSONObject(key);
125                     String result =
(String)json.get("result");
126                     if(result.equals("success")){
127
128                         loginUser.setOldUser(true);
129
130                         loginUser.setUserName(etName.getText().toString());
131                         loginUser.setPassword(etPassword.getText().toString());
132
133                         Toast.makeText(LoginOrRegister.this,"已登
录",Toast.LENGTH_SHORT).show();
134
135                         //设置SharedPreferences
136                         String string =
mButton_Enter.getText().toString();
137                         editor.putString("userName",string);
138                         editor.putInt("loginState",1);
139                         editor.commit();
140
141                         Intent intent = new
Intent(LoginOrRegister.this,MainScreen.class);
142
143                         startActivity(intent);

```

```

140         Toast.makeText(LoginOrRegister.this,"登录成
功",Toast.LENGTH_LONG).show();
141     }
142     else{
143         Toast.makeText(LoginOrRegister.this,"输入内容不符合规
则",Toast.LENGTH_SHORT).show();
144     }
145     }
146     catch (Exception e){
147         e.printStackTrace();
148     }
149     }
150     }
151     };
152     });
153     thread.start();
154
155     dismissProgress();
156     break;
157
158     case R.id.returnButton:
159         String string =
mButton_Enter.getText().toString();
160         editor.putString("userName",string);
161         editor.putInt("loginState",0);
162         editor.commit();
163         intent = new
Intent(LoginOrRegister.this,MainScreen.class);
164         startActivity(intent);
165         break;
166
167     }
168 }
169 }

```

3.修改MainScreen和SCOSEEntry代码

```

1  //SCOSEEntry.java
2  @Override
3  public boolean onTouchEvent(MotionEvent event) {
4      return mGestureDetector.onTouchEvent(event);
5  }

```

```

6
7  GestureDetector.SimpleOnGestureListener mGestureListener = new
  GestureDetector.SimpleOnGestureListener() {
8      public boolean onFling(MotionEvent e1, MotionEvent e2, float
        velocityX, float velocityY) {
9          float x = e1.getX() - e2.getX();
10
11          if (x > MIN_DISTANCE && Math.abs(velocityX) >
            MIN_VELOCITY) {
12
13              showToast(getResources().getString(R.string.welcome)); //显示欢迎
14
15              SharedPreferences.Editor editor =
                getSharedPreferences("User",MODE_PRIVATE).edit();
16              editor.putInt("loginState",0);
17              editor.commit();
18
19              Intent intent = new
                Intent(SCOSEEntry.this,MainScreen.class);
20              intent.putExtra("from","FromEntry");
21              startActivity(intent);
22
23              finish();
24          }
25          return false;
26      }
  };//

```

```

1  //MainScreen
2  @Override
3      protected void onCreate(Bundle savedInstanceState) {
4          super.onCreate(savedInstanceState);
5          setContentView(R.layout.mainscreen);
6          mLoginButton = (Button) findViewById(R.id.HomeLoginButton);
7          mOrderButton = (Button) findViewById(R.id.HomeOrderButton);
8          mMenuButton = (Button) findViewById(R.id.HomeMenuButton);
9          mHelpButton = (Button) findViewById(R.id.HomeHelpButton);
10         mHomeImage = (ImageView) findViewById(R.id.HomeLogo);
11
12         Intent intent = getIntent();
13         String string = intent.getStringExtra("from");
14         SharedPreferences sharedPreferences =
            this.getSharedPreferences("User",MODE_PRIVATE);
15         int loginState = sharedPreferences.getInt("loginState",0);

```



```

16         if(loginState == 0){
17             mOrderButton.setVisibility(View.INVISIBLE);
18             mMenuButton.setVisibility(View.INVISIBLE);
19         }
20         if(loginState == 1){
21             if (mOrderButton.getVisibility() == View.INVISIBLE) {
22                 mOrderButton.setVisibility(View.VISIBLE);
23             }
24             if (mMenuButton.getVisibility() == View.INVISIBLE) {
25                 mMenuButton.setVisibility(View.VISIBLE);
26             }
27         }
28
29         setListeners();
30

```

4.新建SCOSHelper类

实现用户使用协议，关于系统， 电话人工帮助， 短信帮助， 邮件帮助。

```

1  //Help
2  /*
3  使用隐式消息打电话，发送短信，发送邮件时使用SMTP协议
4  */
5  public class Help extends AppCompatActivity {
6      private GridView gridView;
7      private List<Map<String, Object>> dataList;
8      private SimpleAdapter adapter;
9      @Override
10     protected void onCreate(Bundle savedInstanceState) {
11         super.onCreate(savedInstanceState);
12         setContentView(R.layout.help);
13
14         gridView = (GridView) findViewById(R.id.gridview);
15         initData();
16
17         String[] from={"img", "text"};
18
19         int[] to={R.id.grid_view_image,R.id.grid_view_textview};
20
21         adapter=new SimpleAdapter(this, dataList,
22             R.layout.layout_grid_view_item, from, to);
23

```

```
23         gridView.setAdapter(adapter);
24         gridView.setNumColumns(2);
25
26         gridView.setOnItemClickListener(new
AdapterView.OnItemClickListener() {
27             Intent intent = null;
28             @Override
29             public void onItemClick(AdapterView<?> parent, View view,
int position, long arg3) {
30                 switch(position){
31                     case 0:
32                         //协议
33                         Toast.makeText(Help.this, "by冰水鉴
心", Toast.LENGTH_SHORT).show();
34                         break;
35
36                     case 1:
37                         //关于
38                         Toast.makeText(Help.this, "SCOS点餐系
统", Toast.LENGTH_SHORT).show();
39                         break;
40
41                     case 2:
42                         //电话
43                         intent = new
Intent(Intent.ACTION_DIAL, Uri.parse("tel:"+5554));
44                         startActivity(intent);
45                         break;
46
47                     case 3:
48                         //短信
49                         String tel = "5554";
50                         String content = "test scos help";
51                         if
(PhoneNumberUtils.isGlobalPhoneNumber(tel)) {
52
53                             Intent intent = new
Intent(Intent.ACTION_SENDTO, Uri.parse("smsto:" + tel));
54                             intent.putExtra("sms_body", content);
55                             startActivity(intent);
56                         }
57                         Toast.makeText(getApplicationContext(), "求助
短信发送成功", Toast.LENGTH_SHORT).show();
58                         break;
59
```

```

60             case 4:
61                 //邮件
62                 new Thread(new Runnable() {
63                     @Override
64                     public void run() {
65                         sendEmail();
66                     }
67
68                 }).start();
69                 Toast.makeText(Help.this,"求助邮件已发送成
70 功",Toast.LENGTH_SHORT).show();
71                 break;
72             }
73         }
74     });
75
76 }
77
78
79 public void helpToMainscreen(View view){
80     Intent intent = new Intent(this,MainScreen.class);
81     startActivity(intent);
82 }
83
84
85 void initData() {
86     //图标
87     int icno[] = {
88         R.drawable.free04,R.drawable.free08,R.drawable.free21,R.drawable.free
89         28 ,R.drawable.free38};
90     //图标下的文字
91     String name[]={"用户使用协议","关于系统", "电话人工帮助","短信帮
92 助","邮件帮助"};
93     dataList = new ArrayList<Map<String, Object>>();
94     for (int i = 0; i <icno.length; i++) {
95         Map<String, Object> map=new HashMap<String, Object>();
96         map.put("img", icno[i]);
97         map.put("text",name[i]);
98         dataList.add(map);
99     }
100 }
101
102 private void sendEmail() {
103     try {

```

```

101         MailSenderInfo mailInfo = new MailSenderInfo();
102         mailInfo.setMailServerHost("smtp.163.com");
103         mailInfo.setMailServerPort("25");
104         mailInfo.setValidate(true);
105         mailInfo.setUserName("18207162619@163.com"); //邮箱地址
106         mailInfo.setPassword("xxxxxxx"); //邮箱密码
107         mailInfo.setFromAddress("18207162619@163.com"); //和
        username的邮箱地址一致
108         mailInfo.setToAddress("773939719@qq.com");
109         mailInfo.setSubject("邮件主题");
110         mailInfo.setContent("邮件内容");
111
112         //这个类主要来发送邮件
113         SimpleMailSender sms = new SimpleMailSender();
114         boolean b = sms.sendTextMail(mailInfo); //发送文本格式, 返回是
        否发送成功的boolean类型
115
116         Log.e("MainActivity", "MainActivity
        sendEmail()" + b); //sms.sendHtmlMail(mailInfo); //发送html格式
117
118     } catch (Exception e) {
119         Log.e("SendMail", e.getMessage(), e);
120     }
121 }
122 }

```

5.修改FoodOrderView，实现模拟结账功能，结账功能放进了片段中

```

1 //OrderedViewFragment
2 public class OrderedViewFragment extends Fragment {
3     private ListView mListView;
4     private static OrderedListViewAdapter orderedListViewAdapter;
5     private Activity activity;
6     private List<Food> foods = new ArrayList<>();
7     private ProgressDialog progressDialog;
8     private Button button = null;
9     private Button button_cancle = null;
10    private ProgressBar progressBar = null;
11    private TextView textView = null;
12    private View view;
13    private int count=0;
14    private final static int STOP=0x10000;
15    private final static int NEXT=0x10001;
16

```

```
17
18     public OrderedViewFragment(){
19
20     }
21
22     public static OrderedListViewAdapter getOrderListViewAdapter(){
23         return orderedListViewAdapter;
24
25     }
26     @Override
27     public View onCreateView(LayoutInflater inflater, @Nullable
ViewGroup container, Bundle savedInstanceState) {
28         view =
inflater.inflate(R.layout.layout_fragment_ordered_view,
container,false);
29
30         listView =
(ListView)view.findViewById(R.id.ordered_view_fragment);
31         orderedListViewAdapter = new
OrderedListViewAdapter(getActivity(),foods);
32         listView.setAdapter(orderedListViewAdapter);
33         button = (Button)
view.findViewById(R.id.ordered_view_button);
34         button.setOnClickListener(new View.OnClickListener() {
35             @Override
36             public void onClick(View v) {
37                 //Toast.makeText(getContext(),"请结
账",Toast.LENGTH_SHORT);
38                 MyAsyncTask myAsyncTask = new MyAsyncTask();
39                 myAsyncTask.execute();
40             }
41         });
42         button_cancle = (Button)
view.findViewById(R.id.ordered_view_button_cancle);
43         button_cancle.setOnClickListener(new View.OnClickListener() {
44             @Override
45             public void onClick(View v) {
46                 //
47                 for(int i = 0; i < ordered.size();i++){
48                     ordered.get(i).setInlist(false);
49                     ordered.get(i).setCount(0);
50                     ordered.get(i).setState(true);
51                 }
52                 ordered.clear();
```

```

53         Toast.makeText(getActivity(),"已取
    消",Toast.LENGTH_SHORT).show();
54         orderedListViewAdapter.notifyDataSetChanged();
55         //Intent intent = new
    Intent(getActivity(),SCOSEEntry.class);
56         //startActivity(intent);
57     }
58 });
59     return view;
60 }
61
62     public void initFoods(List<Food> foods){
63         this.foods = foods;
64         //foods.add(new Food("",0,R.drawable.wine));
65     }
66     private class MyAsyncTask extends AsyncTask<String, Object,
    Boolean>{
67         @Override
68         protected void onPreExecute() {
69             //showProgress("结账中...");
70             progressBar = view.findViewById(R.id.progressBar);
71             progressBar.setVisibility(View.VISIBLE);
72             progressBar.setMax(100);
73             progressBar.setProgress(0);
74             textView = view.findViewById(R.id.progress_text);
75             textView.setVisibility(View.VISIBLE);
76         }
77
78         @Override
79         protected Boolean doInBackground(String... strings) {
80             Thread t = new Thread(new Runnable(){
81                 @Override
82                 public void run() {
83                     // TODO Auto-generated method stub
84                     for(int i=0;i<20;i++){
85                         try {
86                             count=(i+1)*5;
87                             Thread.sleep(300);
88                             textView.setText(count+"%");
89                             if(count==100){
90                                 Message msg=new Message();
91                                 msg.what=STOP;
92
93                                 handler.sendMessage(msg);
94                                 break;

```

```

95         }
96         else{
97             Message msg=new Message();
98             msg.what=NEXT;
99             handler.sendMessage(msg);
100        }
101        } catch (InterruptedException e) {
102            // TODO Auto-generated catch block
103            e.printStackTrace();
104        }
105    }
106
107    }
108    });
109    t.start();
110
111    return null;
112    }
113
114    @Override
115    protected void onPostExecute(Boolean aBoolean) {
116        ordered.clear();
117        orderedListViewAdapter.notifyDataSetChanged();
118        button.setClickable(false);
119        super.onPostExecute(aBoolean);
120
121    }
122    @Override
123    protected void onProgressUpdate(Object... values) {
124        super.onProgressUpdate(values);
125    }
126    }
127
128    public void showProgress(String messageRes) {
129        progressDialog = new ProgressDialog(getActivity());
130
131        progressDialog.setTitle(getResources().getString(R.string.app_name))
132        ;
133        progressDialog.setMessage(messageRes);
134        progressDialog.setCancelable(false);
135        progressDialog.show();
136    }
137    private Handler handler=new Handler(){
138        @SuppressWarnings("static-access")

```

```

138         public void handleMessage(Message msg){
139             switch(msg.what){
140                 case STOP:
141                     progressBar.setVisibility(View.GONE);
142                     Thread.currentThread().interrupt();
143                     progressBar =
view.findViewById(R.id.progressBar);
144                     progressBar.setVisibility(View.INVISIBLE);
145                     textView = view.findViewById(R.id.progress_text);
146                     textView.setVisibility(View.INVISIBLE);
147                     Toast.makeText(getApplicationContext(),"已结
账",Toast.LENGTH_SHORT).show();
148                     break;
149                 case NEXT:
150                     if(!Thread.currentThread().interrupted()){
151                         progressBar.setProgress(count);
152                     }
153                     break;
154             }
155         }
156     };
157 }

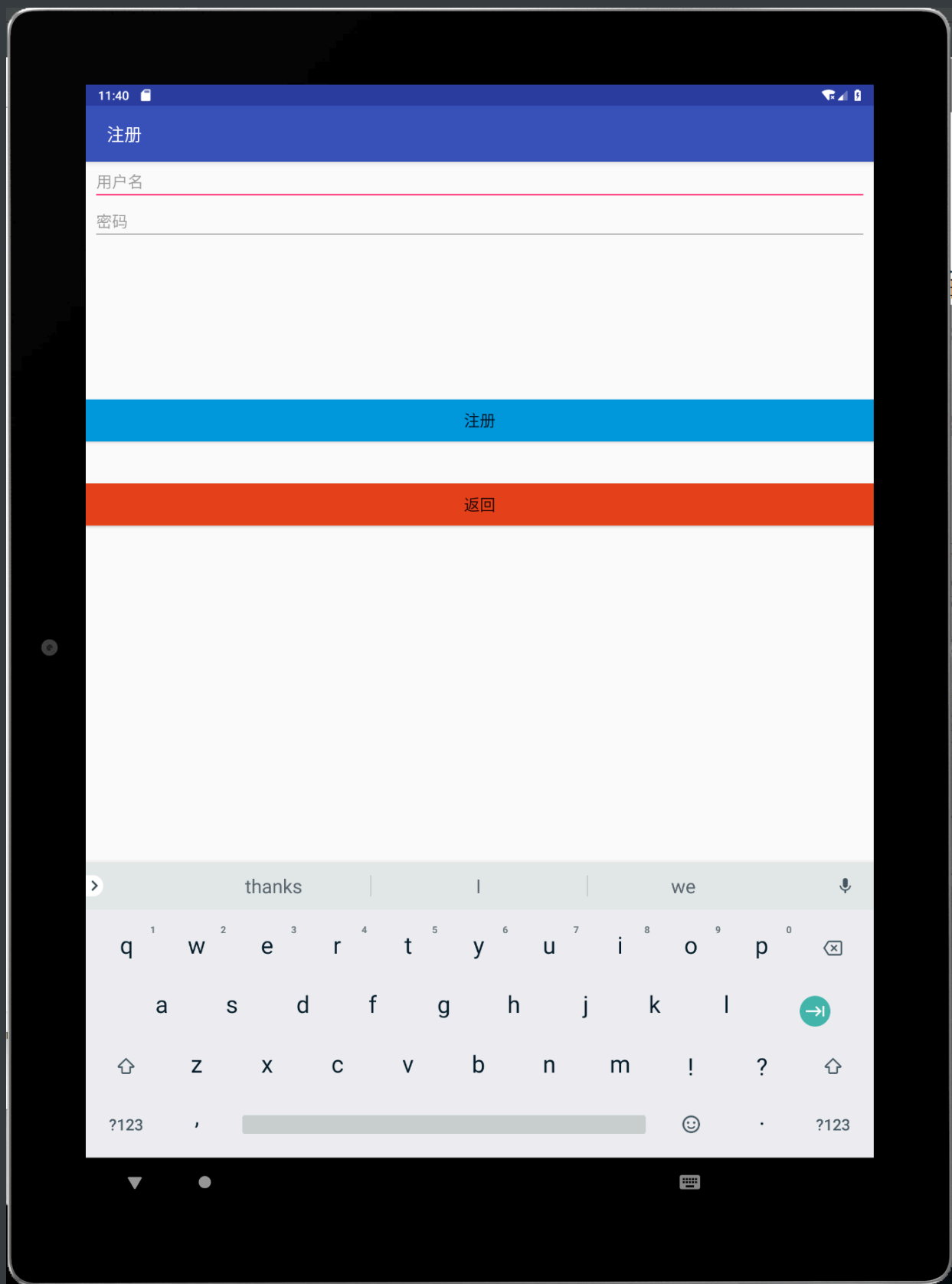
```

4.结论

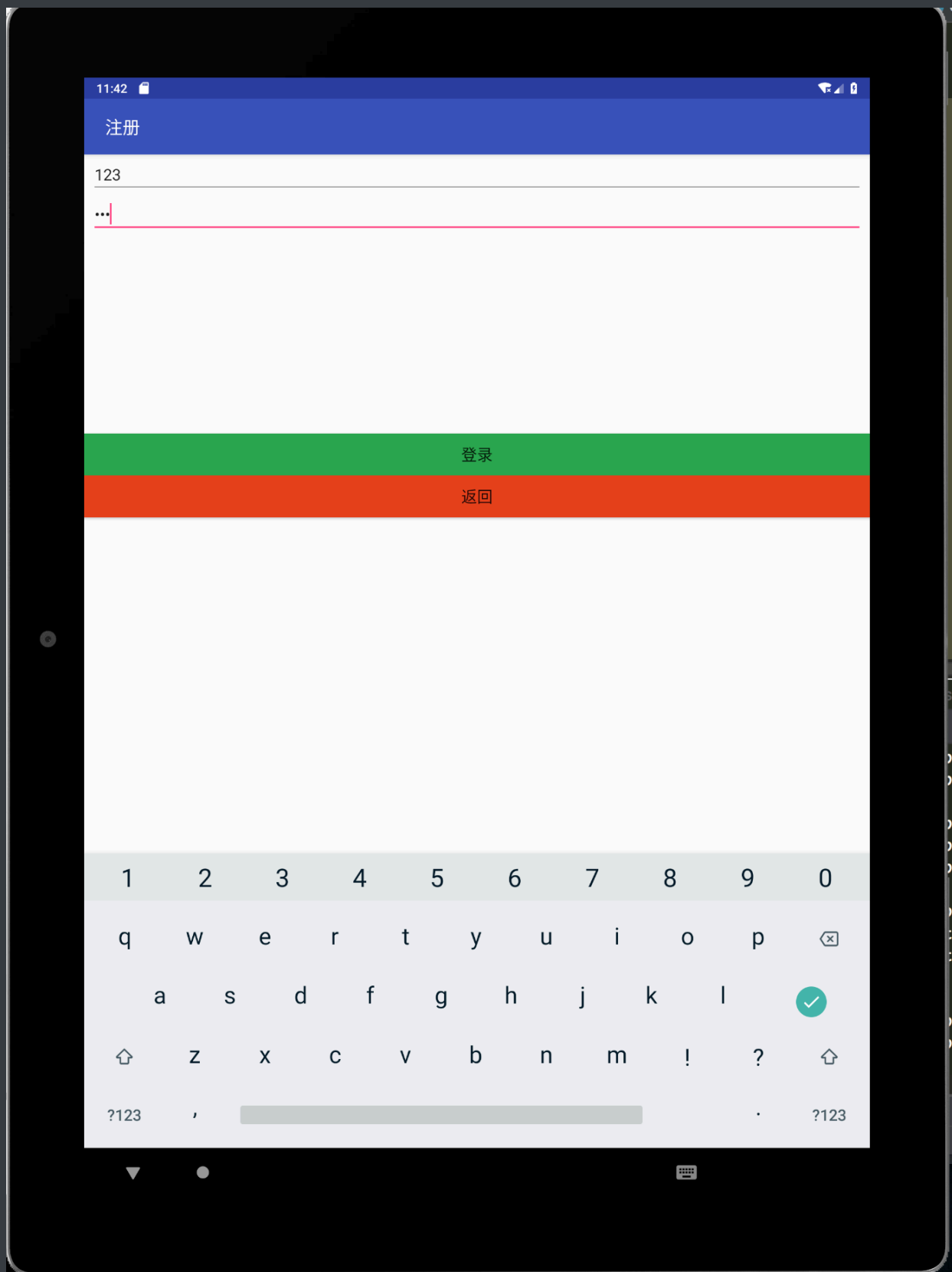
本次练习：

1. 使用SharedPreferences来实现数据读写，熟悉了这个轻量级存储工具的使用。
2. 再次使用了GridView，熟练掌握adapter view的用法。
3. 同时掌握了多线程处理方法。

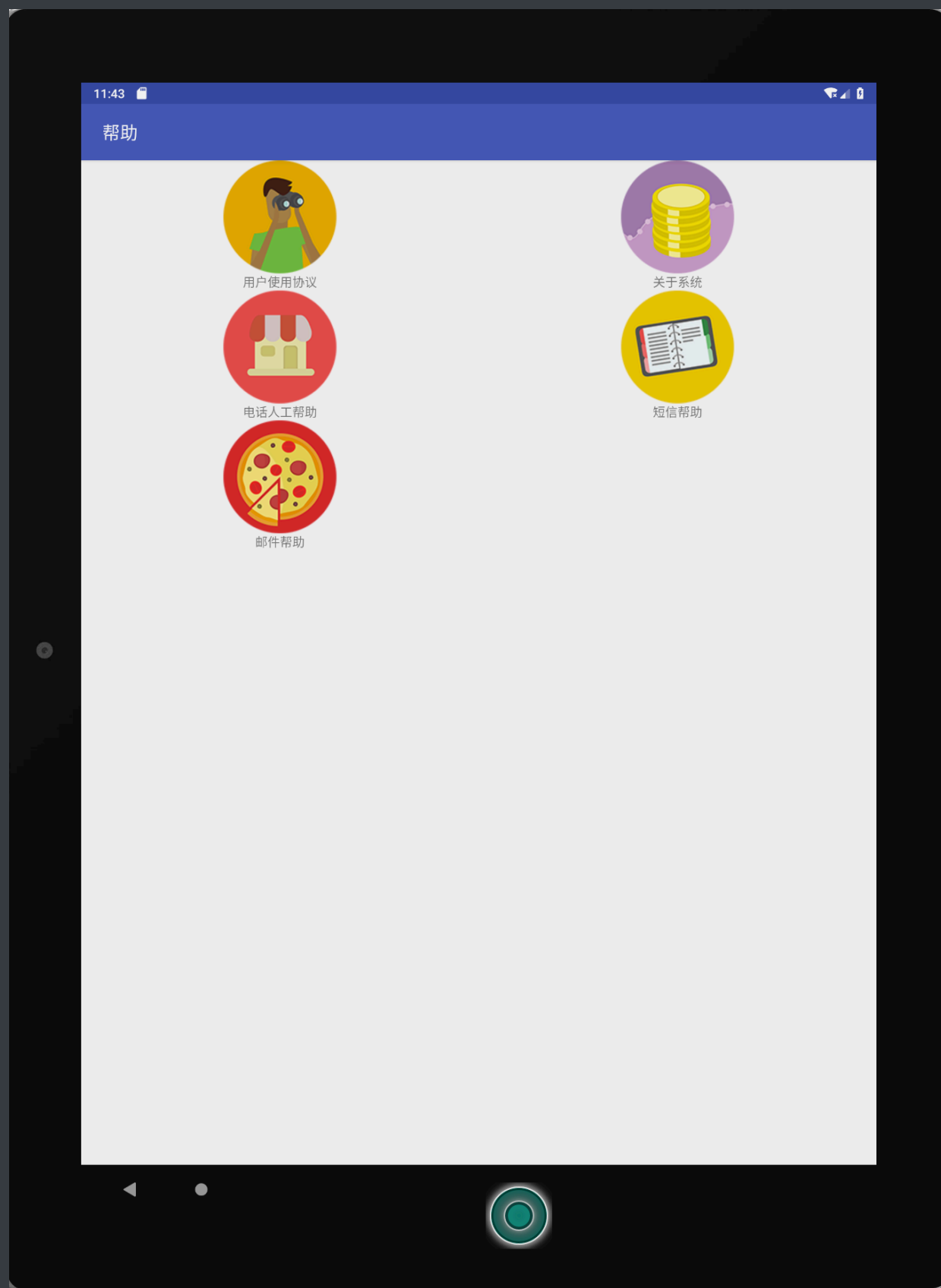
1.效果图



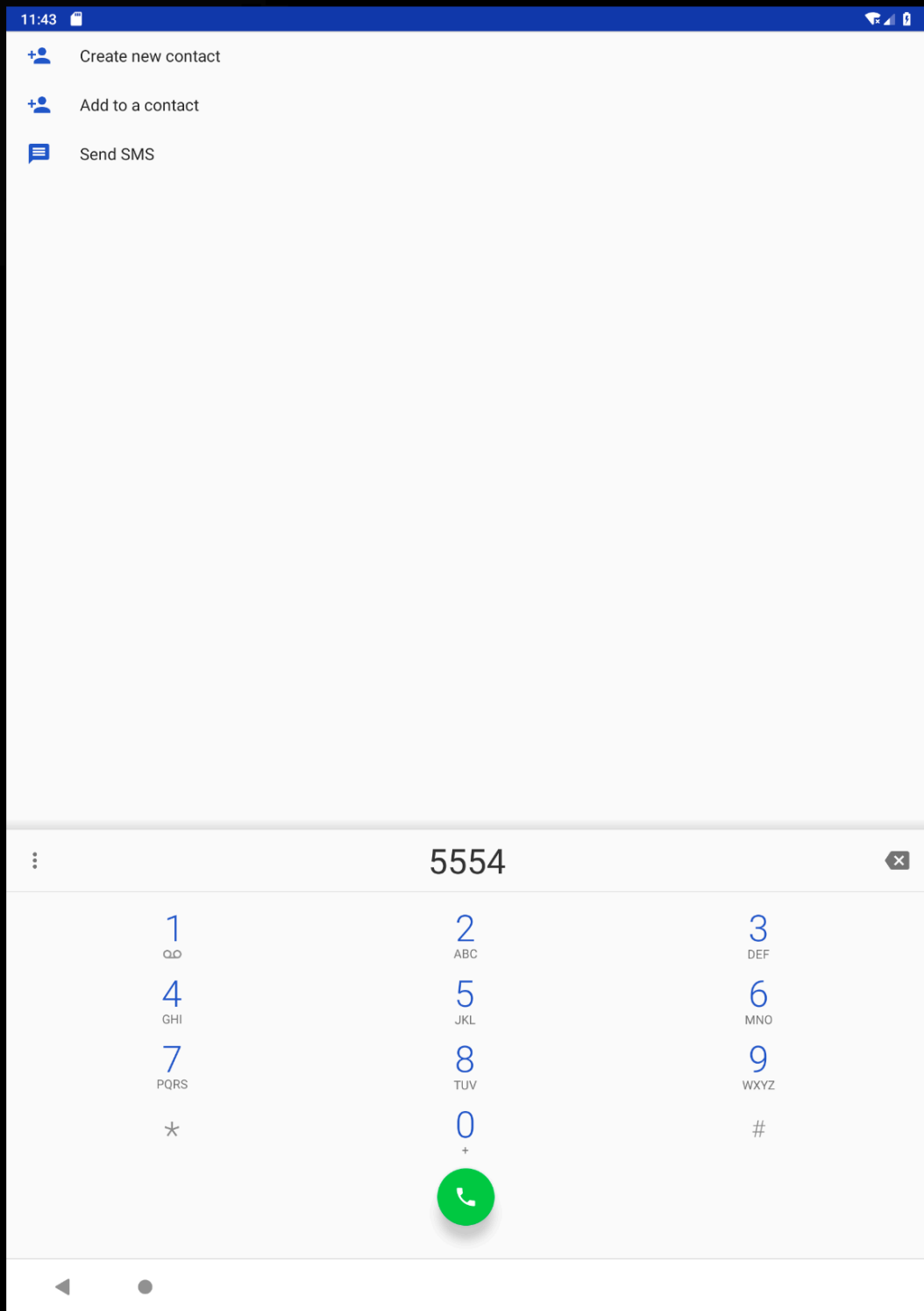
注册页面



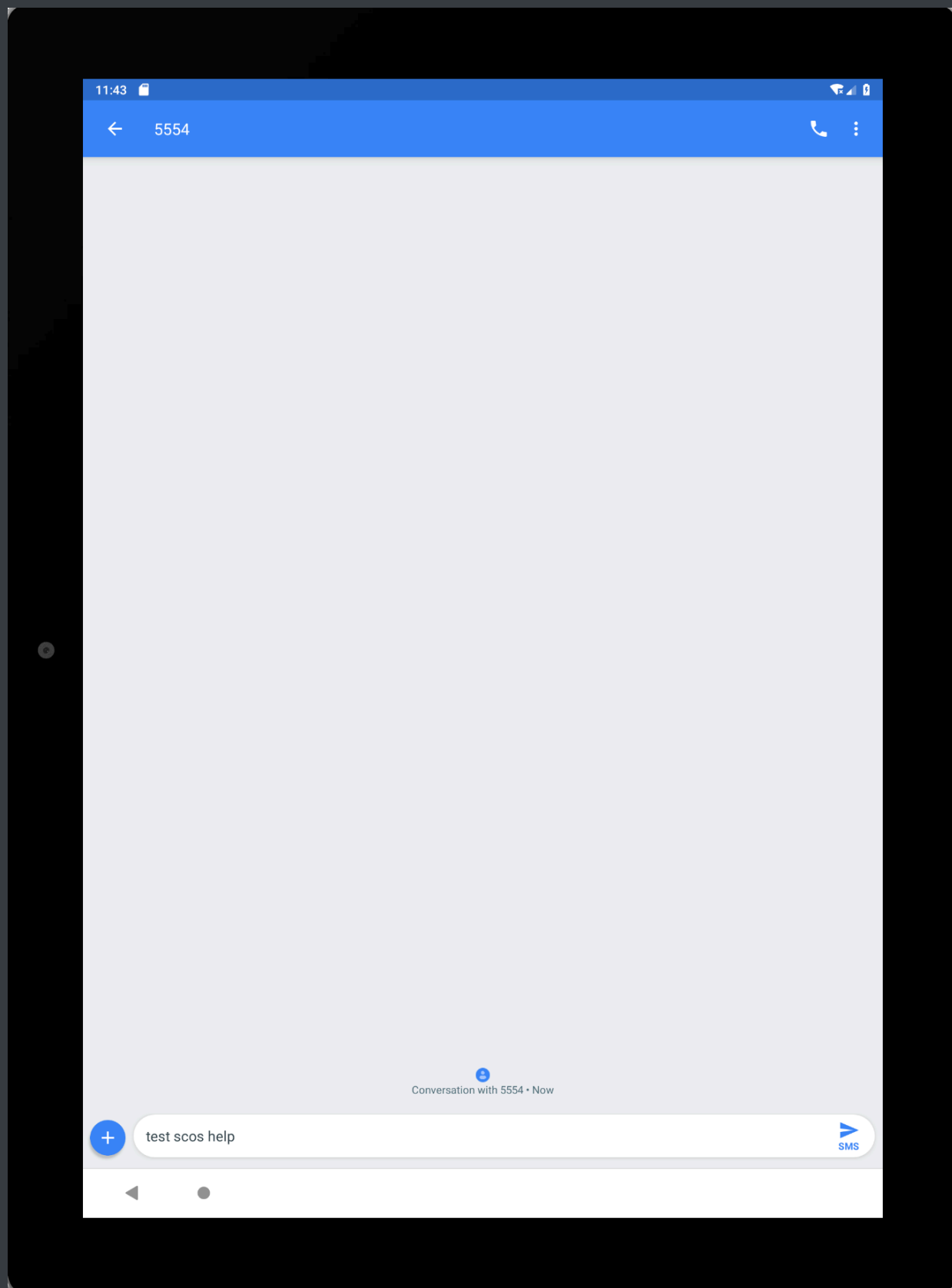
注册成功后显示登录



帮助页面



拨打求助电话



短信页面



18207162619

11月4日 10:02

发给：冰水鉴心

📄

18207162619 将 冰水鉴心 加入此次会话

📄 📄 📄 📄

邮件详情 

请勿轻信密保、汇款、中奖信息，勿轻易拨打陌生电话。🔥 举报垃圾邮件

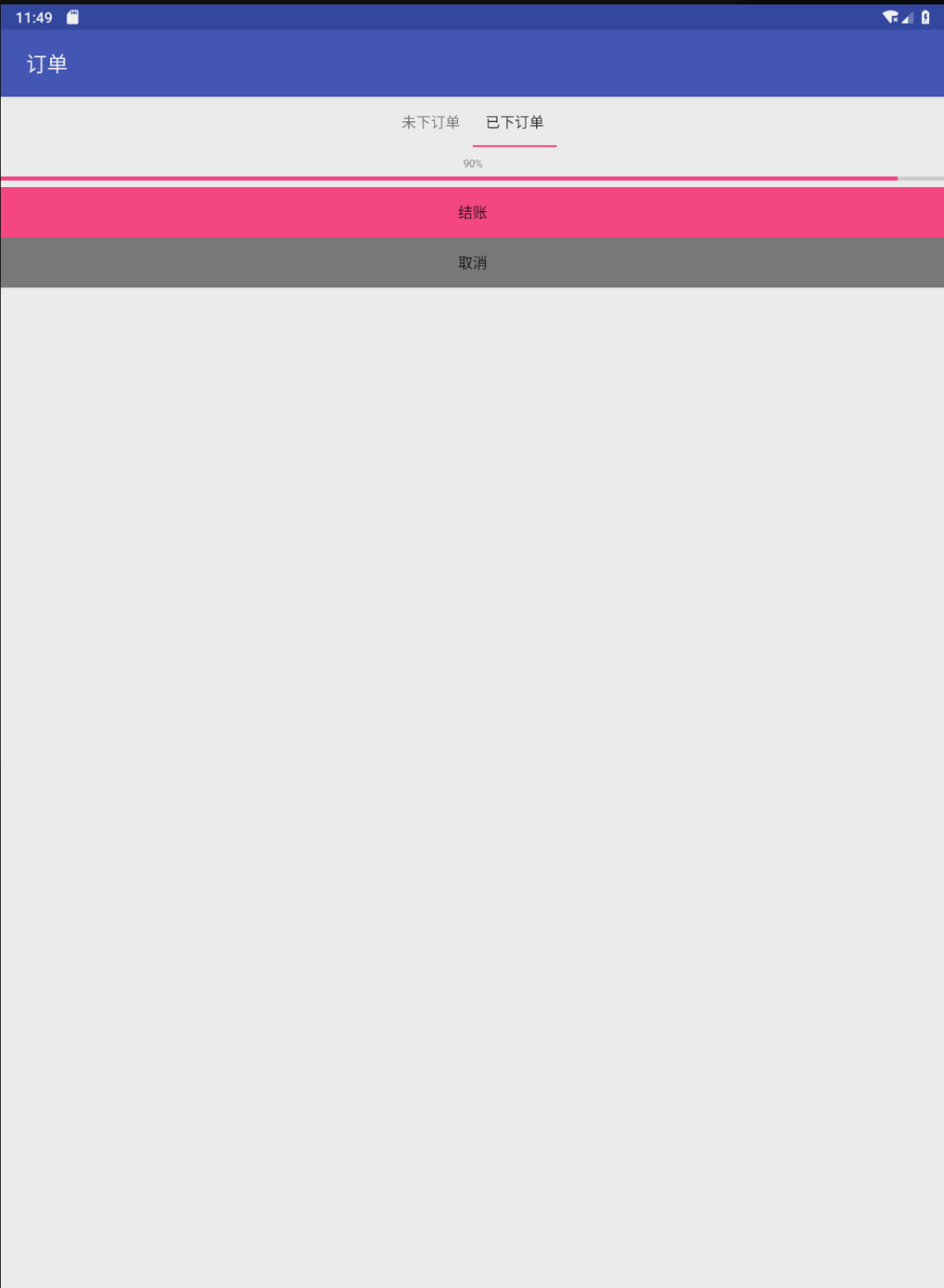
网站安全云检测

邮件内容

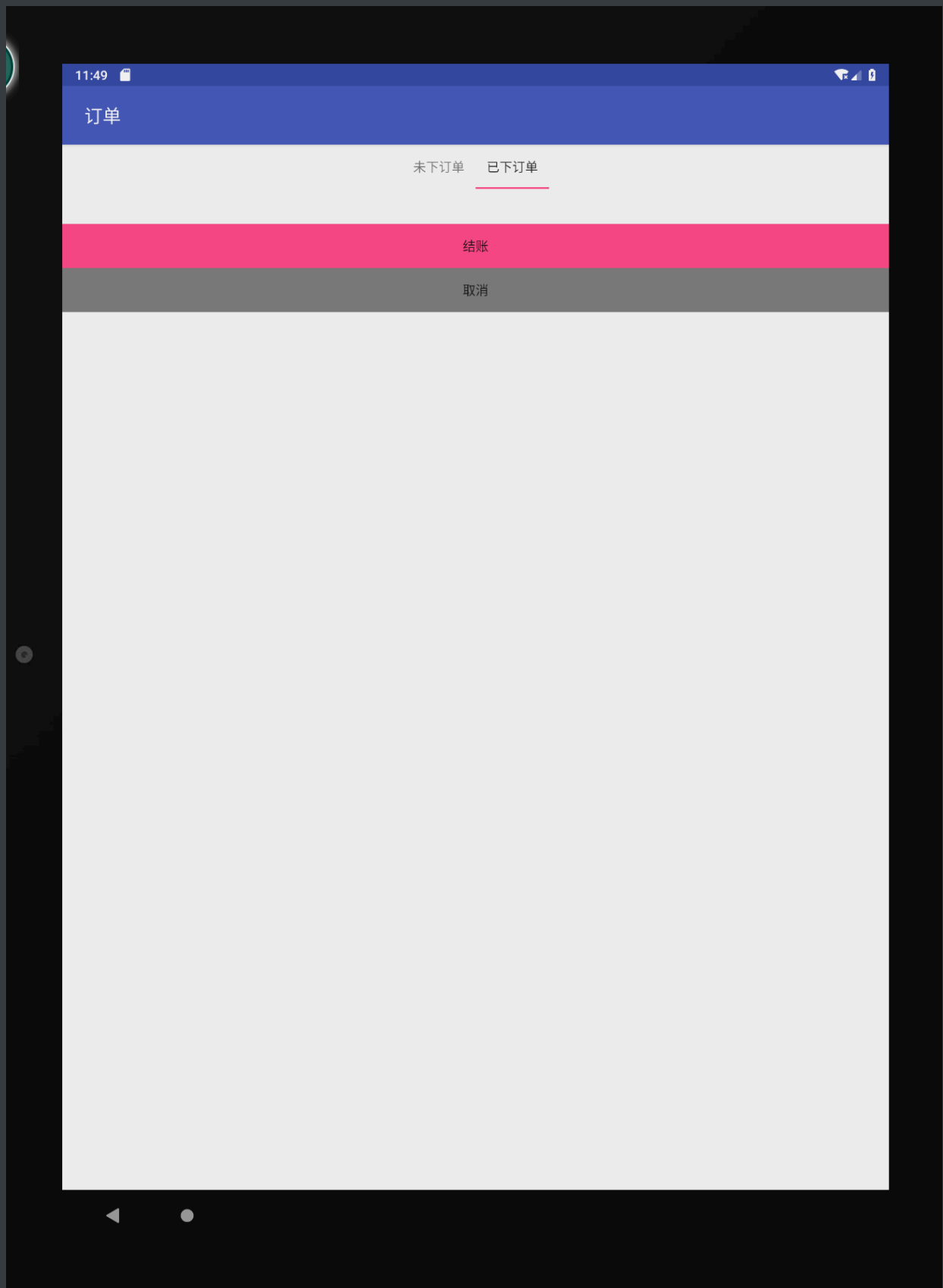
快捷回复给：18207162619

 回复  回复全部  转发  删除 更多...

邮件接收



订单结账进度



结账完毕后，结账按钮不可点击

2.如何提高程序性能和避免ANR异常的出现？

Android程序不可能无限制地使用内存和CPU资源，过多地使用内存会导致程序内存溢出，即OOM(Out of Memory)；而过多地使用CPU资源，一般是指做大量的耗时任务，会导致手机变得卡顿甚至程序无法响应的情况，即ANR(Application not Responding)。

性能优化：

1. 线程优化：线程优化的思想是采用线程池，避免程序中存在大量的Thread。线程池可以重用内部的线程，从而避免了线程的创建和销毁所带来的性能开销，同时线程池还能有效地控制线程池的最大并发数，避免大量的线程因互相抢占系统资源而导致阻塞现象的发生。因此，在实际开发中，我们要尽量采用线程池，而不是每次都要创建一个Thread对象。

2. 布局优化：布局优化的思想很简单，就是尽量减少布局文件的层级。有选择地使用性能较低的ViewGroup，比如RelativeLayout。如果布局中既可以使用LinearLayout也可以使用RelativeLayout，那么就使用LinearLayout，因为RelativeLayout比较复杂，它的布局过程需要花费更多的CPU时间。FrameLayout和LinearLayout一样都是一种简单高效的ViewGroup，因此也可以多使用FrameLayout。

避免ANR出现：

1. 避免在主线程上进行复杂耗时的操作，比如说发送接收网络数据/进行大量计算/操作数据库/读写文件等。这个可以通过使用AsyncTask或者使用多线程来实现。

2. BroadcastReceiver 要进行复杂操作的的时候，可以在onReceive()方法中启动一个Service来处理。

3. 在设计及代码编写阶段避免出现出现同步/死锁或者错误处理不恰当等情况。

3.在多线程编程中，怎样保证线程安全？

保证线程安全：

1. 竞争与原子操作 多个线程同时访问和修改一个数据，可能造成很严重的后果。出现严重后果的原因是很多操作被操作系统编译为汇编代码之后不止一条指令，因此在执行的时候可能执行了一半就被调度系统打断了而去执行别的代码了。一般将单指令的操作称为原子的(Atomic)，因为不管怎样，单条指令的执行是不会被打断的。因此，为了避免出现多线程操作数据的出现异常，Linux系统提供了一些常用操作的原子指令，确保了线程的安全。但是，它们只适用于比较简单的场合，在复杂的情况下就要选用其他的方法了。

2. 同步与锁 为了避免多个线程同时读写一个数据而产生不可预料的后果，开发人员要将各个线程对同一个数据的访问同步，也就是说，在一个线程访问数据未结束的时候，其他线程不得对同一个数据进行访问。同步的最常用的方法是使用锁(Lock)，它是一种非强制机制，每个线程在访问数据或资源之前首先试图获取锁，并在访问结束之后释放锁；在锁已经被占用的时候试图获取锁时，线程会等待，直到锁重新可用。二元信号量是最简单的一种锁，它只有两种状态：占用与非占用，它适合只能被唯一一个线程独占访问的资源。对于允许多个线程并发访问的资源，要使用多元信号量(简称信号量)。

3.可重入 一个函数被重入，表示这个函数没有执行完成，但由于外部因素或内部因素，又一次进入该函数执行。一个函数称为可重入的，表明该函数被重入之后不会产生任何不良后果。可重入是并发安全的强力保障，一个可重入的函数可以在多线程环境下放心使用。

4.防止过度优化 在很多情况下，即使我们合理地使用了锁，也不一定能够保证线程安全，因此，我们可能对代码进行防止过度的优化以确保线程安全。我们可以使用volatile关键字试图阻止过度优化，它可以做两件事：第一，阻止编译器为了提高速度将一个变量缓存到寄存器而不写回；第二，阻止编译器调整操作volatile变量的指令顺序。

5.参考

参考书籍

- 1 【1】 《Head First Android开发》
- 2 【2】 《Android第一行代码》
- 3 【3】 《Head First Java》
- 4 【4】 《Java编程思想》
- 5 【5】 《Java核心技术》
- 6 【6】 《Android编程权威指南》

参考链接

- 【1】 [Android的视频教程](#)
- 【2】 [提供帮助的社区](#)
- 【3】 [技术博客CSDN](#)
- 【4】 [技术博客开源中国](#)
- 【5】 [张新强的CSDN博客](#)
- 【6】 [Hongyang](#)