

说明文档

1. 软件名称

注册登录界面

2. 完成人

彭翊桐 20192131018

3. 完成时间


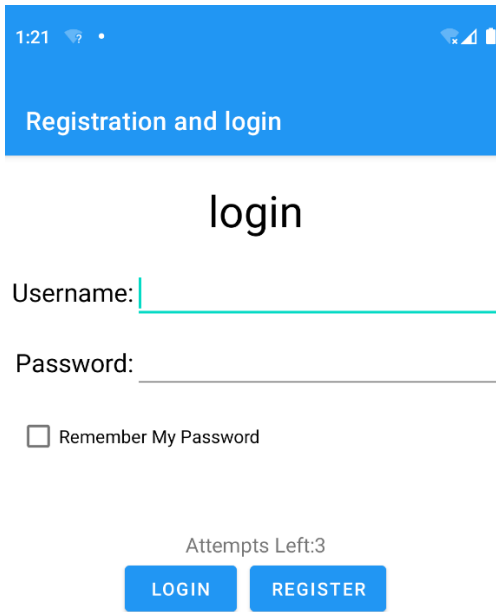
2021 年 11 月 20 日

4. 内容简介

本软件实现了用户注册、登录、记住密码、限制密码输错次数的功能并支持中英双语。

5. 界面设计：

登录界面：

	
--	---

The image displays two side-by-side mobile app interface mockups for a login and registration system. The left mockup is in Chinese, titled '登录与注册' (Login and Registration), and the right is in English, titled 'Registration and login'. Both screens feature a blue header bar with the time '1:21' and status icons. The main content area is white. The Chinese version has fields for '用户名:' (Username) and '密码:' (Password), a '记住密码' (Remember Password) checkbox, and buttons for '登录' (Login) and '注册' (Register). It also shows '可尝试的剩余次数: 3' (Attempts Left: 3). The English version has fields for 'Username:' and 'Password:', a 'Remember My Password' checkbox, and buttons for 'LOGIN' and 'REGISTER'. It also shows 'Attempts Left:3'. Both screens have a black Android navigation bar at the bottom.

注册界面：

12:57

Registration and login

register

Username:

Password:

REGISTER

CANCEL

12:57

登录与注册

注册

用户名:

密码:

注册

取消

未注册时点击登录：

12:56

登录与注册

登录

用户名:

密码:

☒ 记住密码

可尝试的剩余次数: 3

登录

注册

请先注册

12:56

Registration and login

login

Username:

Password:

☒ Remember My Password

Attempts Left:3

LOGIN

REGISTER

Please register first

注册时信息为空：

12:57 登录与注册	12:57 Registration and login
注册	register
用户名：_____	Username: _____
密码：_____	Password: _____
<div>注册</div> <div>取消</div>	<div>REGISTER</div> <div>CANCEL</div>
<div>用户名和密码不能为空</div> <div>User name and password cannot be empty</div>	

登录成功：

1:22 登录与注册	1:22 Registration and login
登录	login
用户名：PYT_____	Username: PYT_____
密码：.... _____	Password: _____
<input type="checkbox"/> 记住密码	<input type="checkbox"/> Remember My Password
<div>可尝试的剩余次数：3</div> <div>登录</div> <div>注册</div>	<div>Attempts Left:3</div> <div>LOGIN</div> <div>REGISTER</div>
<div>登陆成功</div> <div>Login successful</div>	

登录时信息为空：

1:26

登录与注册

登录

用户名：PYT

密码：

☒ 记住密码

可尝试的剩余次数：3

登录

注册

请输入用户名密码

1:26

Registration and login

login

Username: PYT

Password:

☒ Remember My Password

Attempts Left:3

LOGIN

REGISTER

Please enter user name and password

无剩余尝试次数：

1:27

登录与注册

登录

用户名：PYT

密码：...

☒ 记住密码

可尝试的剩余次数：0

登录

注册

无剩余尝试次数

1:27

Registration and login

login

Username: PYT

Password: ...

☒ Remember My Password

Attempts Left:0

LOGIN

REGISTER

No attempts left

6. 代码设计:

登录界面:

```
package com.example.registrationandlogin

import ...

class MainActivity : AppCompatActivity() {
    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        setContentView(R.layout.activity_main)

        /*点击注册按钮，跳转到注册界面*/
        button.setOnClickListener { it: View!
            val intent = Intent( packageContext: this, LoginActivity::class.java)
            startActivity(intent)
        }
    }
}
```

```
/*设定并显示原始剩余密码输入次数*/
var left = 3
textView6.text = left.toString()

/*默认记住用户名*/
val prefs1 = getSharedPreferences( name: "userdata", Context.MODE_PRIVATE)
editTextTextPersonName.setText(prefs1.getString("username", ""))
/*若记住了密码则显示密码并勾选记住密码*/
if (prefs1.getBoolean("check", false))
{
    editTextTextPersonName2.setText(prefs1.getString("password", ""))
    checkBox.isChecked = true
}
/*否则不勾选记住密码*/
else
    checkBox.isChecked = false
```

```

/*设置登录按钮*/
button2.setOnClickListener{ it: View!
    if (left == 0) //若无剩余输入次数则提示并不允许登录
        Toast.makeText( context: this, "无剩余尝试次数", Toast.LENGTH_SHORT).show()
    else //否则
    {
        val prefs = getSharedPreferences( name: "userdata", Context.MODE_PRIVATE)
        val username = prefs.getString("username", "")
        val password = prefs.getString("password", "")
        if (username == "") //未保存有用户名信息表示为注册，提示需要先注册
            Toast.makeText( context: this, "请先注册", Toast.LENGTH_SHORT).show()
        else if (editTextTextPersonName.text.toString() == "" || editTextTextPersonName2.text.toString() == "") //用户名或密码为输入，提示输入
            Toast.makeText( context: this, "请输入用户名密码", Toast.LENGTH_SHORT).show()
        /*验证输入*/
        else if (username.equals(editTextTextPersonName.text.toString()) && password.equals(editTextTextPersonName2.text.toString())) //
        {
            Toast.makeText( context: this, "登陆成功", Toast.LENGTH_SHORT).show()
            val editor = getSharedPreferences( name: "userdata", Context.MODE_PRIVATE).edit()
            editor.putBoolean("check", checkBox.isChecked)
            editor.apply()
        }
        else //错误
        {
            Toast.makeText( context: this, "用户名密码错误", Toast.LENGTH_SHORT).show()
            left-- //剩余输入次数减一
            textView6.text = left.toString()
        }
    }
}
}

```

注册界面：

```

package com.example.registrationandLogin

```

```

import ...

```

```

class LoginActivity : AppCompatActivity() {
    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        setContentView(R.layout.activity_login)

        button.setOnClickListener{ it: View!
            this.finish()
        }
    }
}

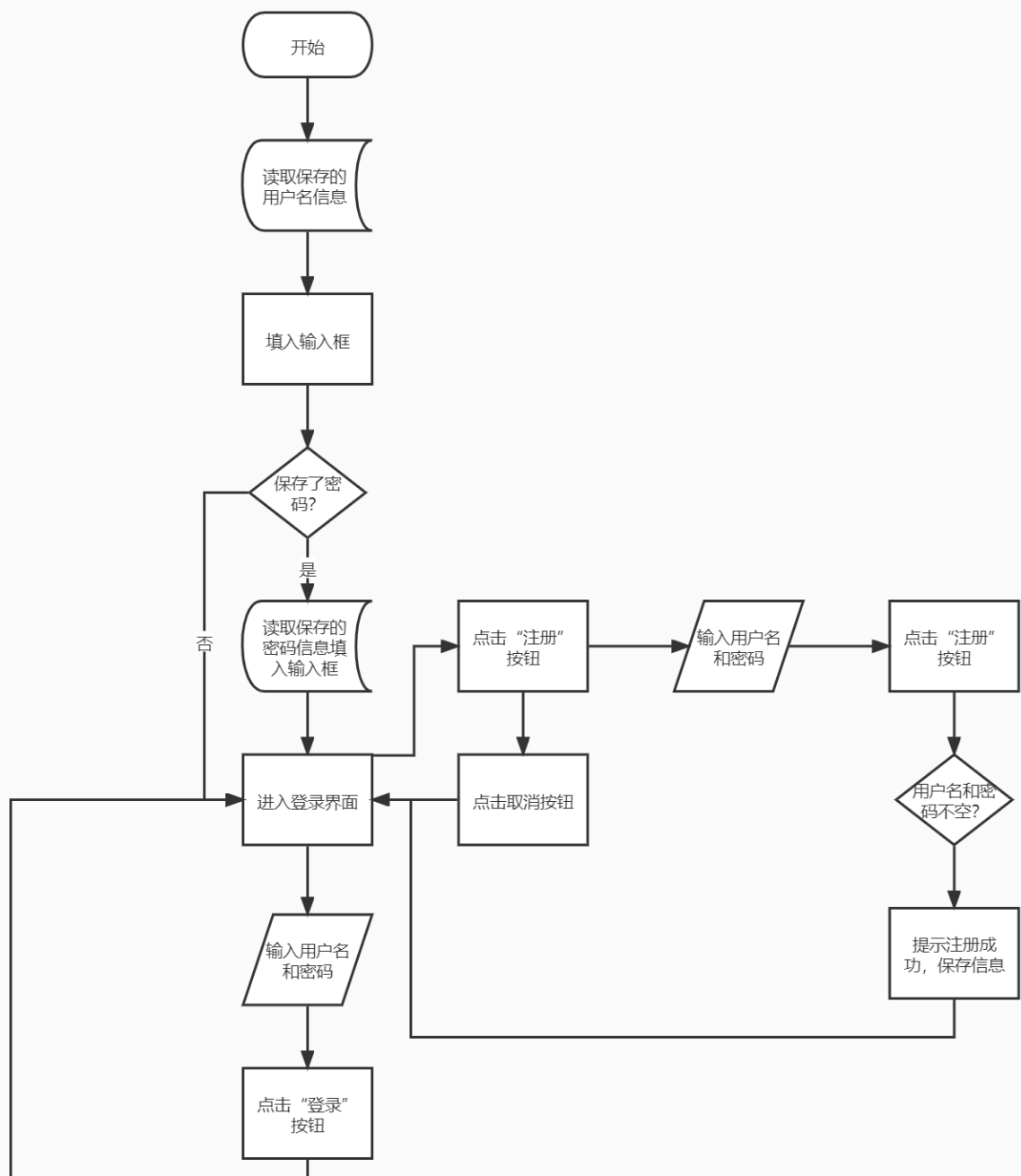
```

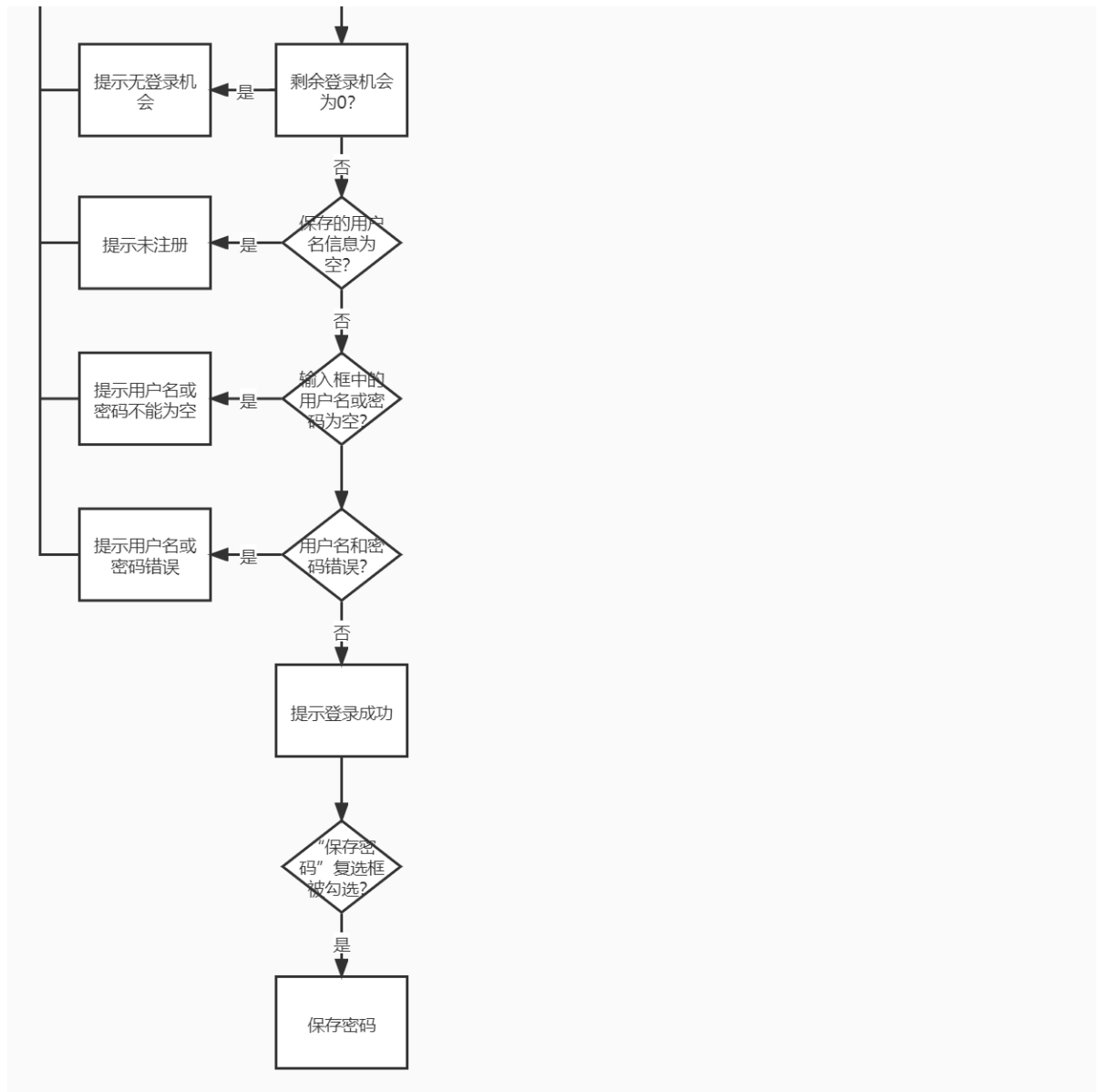
```

button2.setOnClickListener{ it: View!
    val editor = getSharedPreferences( name: "userdata", Context.MODE_PRIVATE).edit()
    if (editTextTextPersonName.text.toString() != "" && editTextTextPersonName2.text.toString() != "") //若用户名和密码均不空
    {
        //保存信息
        editor.putString("username", editTextTextPersonName.text.toString())
        editor.putString("password", editTextTextPersonName2.text.toString())
        editor.apply()
        Toast.makeText( context: this, R.string.signupsuccessfully, Toast.LENGTH_SHORT).show()
        this.finish()
    }
    else //否则提示不能为空
        Toast.makeText( context: this, R.string.notempty, Toast.LENGTH_SHORT).show()
}
}
}

```

7. 软件操作流程：





8. 难点和解决方案：

保存密码：解决方案是保存是否勾选保存密码复选框的布尔值，当程序启动时读取该布尔值，若为真则读取密码并填入编辑框，若为假则不读取。

9. 不足之处：

当剩余尝试次数为 0 时没有禁用登录按钮。

10. 今后的设想：

当剩余尝试次数为 0 时禁用登录按钮，不允许用户点击。