

Daniel Rebouças de Queiroz - 421751

1) Interface:

The image displays two side-by-side mobile application screens. Both screens have a status bar at the top showing the time 11:26, a gear icon, a signal icon, and a battery icon. The left screen, titled 'Contatos de Emergência', features a light purple header. Below it, a white box contains the text 'Bem vindo' in bold. Underneath are input fields for 'Login' and 'Senha', followed by a purple 'ENTRAR' button. At the bottom of the white box, there is a purple 'NOVO' button and a link 'Esqueceu a senha?'. The right screen, titled 'Novo Usuário', also has a light purple header. It contains a white box with the title 'Novo Usuário' in bold. Below the title are input fields for 'Nome', 'Usuário', 'Senha', and 'Email'. There are two toggle switches: 'Manter Logado?' (which is turned on) and 'Tema escuro?' (which is turned off). At the bottom of the white box is a purple 'CRIAR' button. Both screens have a black navigation bar at the bottom with standard Android navigation icons.

11:26 11:26

Contatos de Emergência Novo Usuário

Bem vindo **Novo Usuário**

Login Nome

Senha Usuário

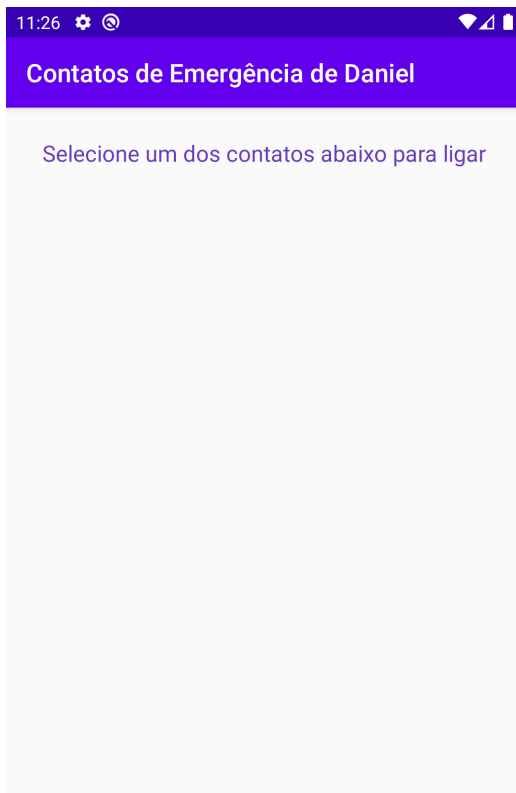
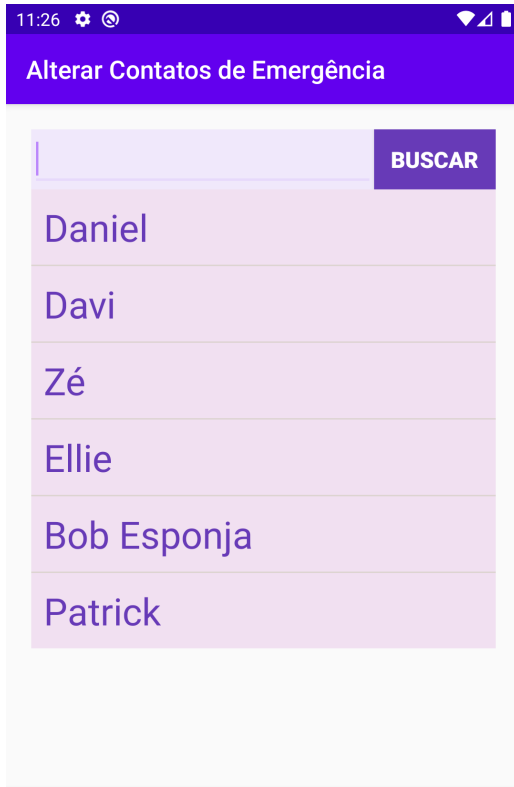
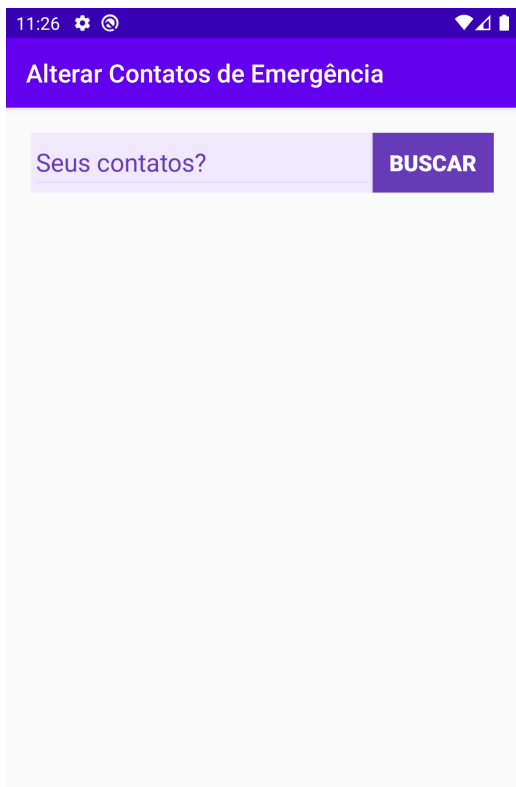
Senha Senha

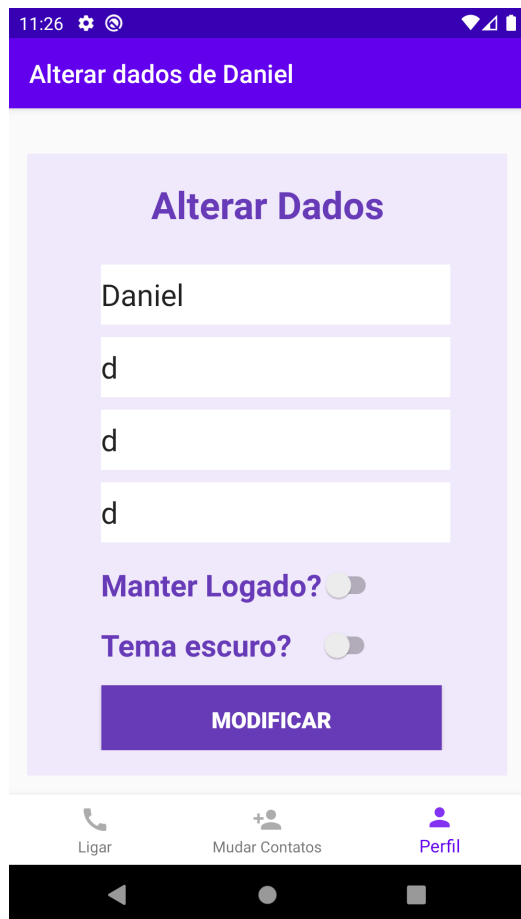
Email

ENTRAR **Manter Logado?** ☒

NOVO [Esqueceu a senha?](#) **Tema escuro?** ☐

CRIAR





2) Armazenamento Local (usuário e contatos)

Salva o usuário localmente:

```
public void salvarModificacoes(User user){
    SharedPreferences salvaUser = getSharedPreferences( name: "usuarioPadrao", Activity.MODE_PRIVATE);
    SharedPreferences.Editor escritor = salvaUser.edit();

    escritor.putString( s: "nome",user.getNome());
    escritor.putString( s: "senha",user.getSenha());
    escritor.putString( s: "login",user.getLogin());

    //Escrever no SharedPreferences
    escritor.putString( s: "email",user.getEmail());
    escritor.putBoolean( s: "manterLogado",user.isManterLogado());
    escritor.putBoolean( s: "tema",user.isTema_escuro());

    //Falta Salvar o E-mail

    escritor.commit(); //Salva em Disco

    Toast.makeText( context: PerfilUsuario_Activity.this, text: "Modificações Salvas",Toast.LENGTH_LONG).show() ;

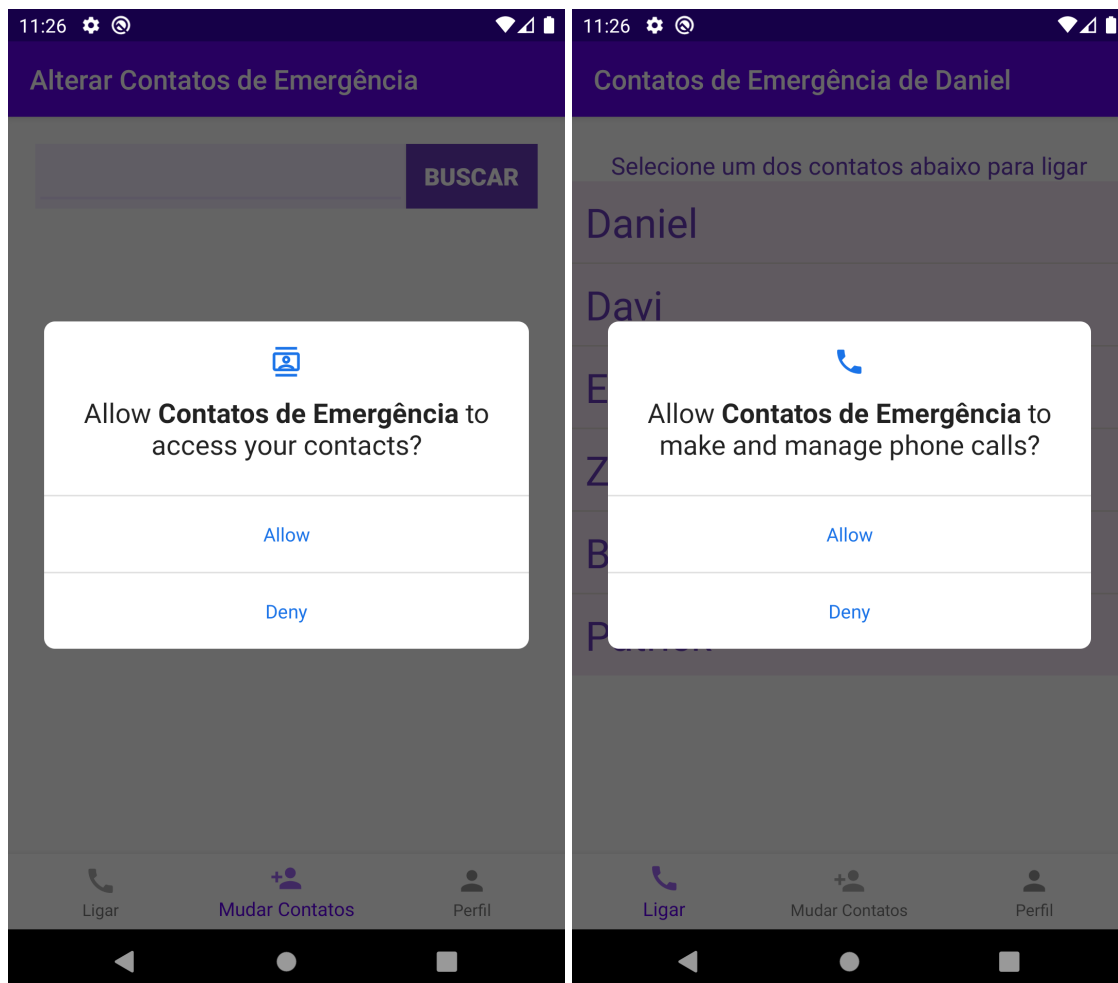
    finish();
}
```

Salva os contatos localmente:

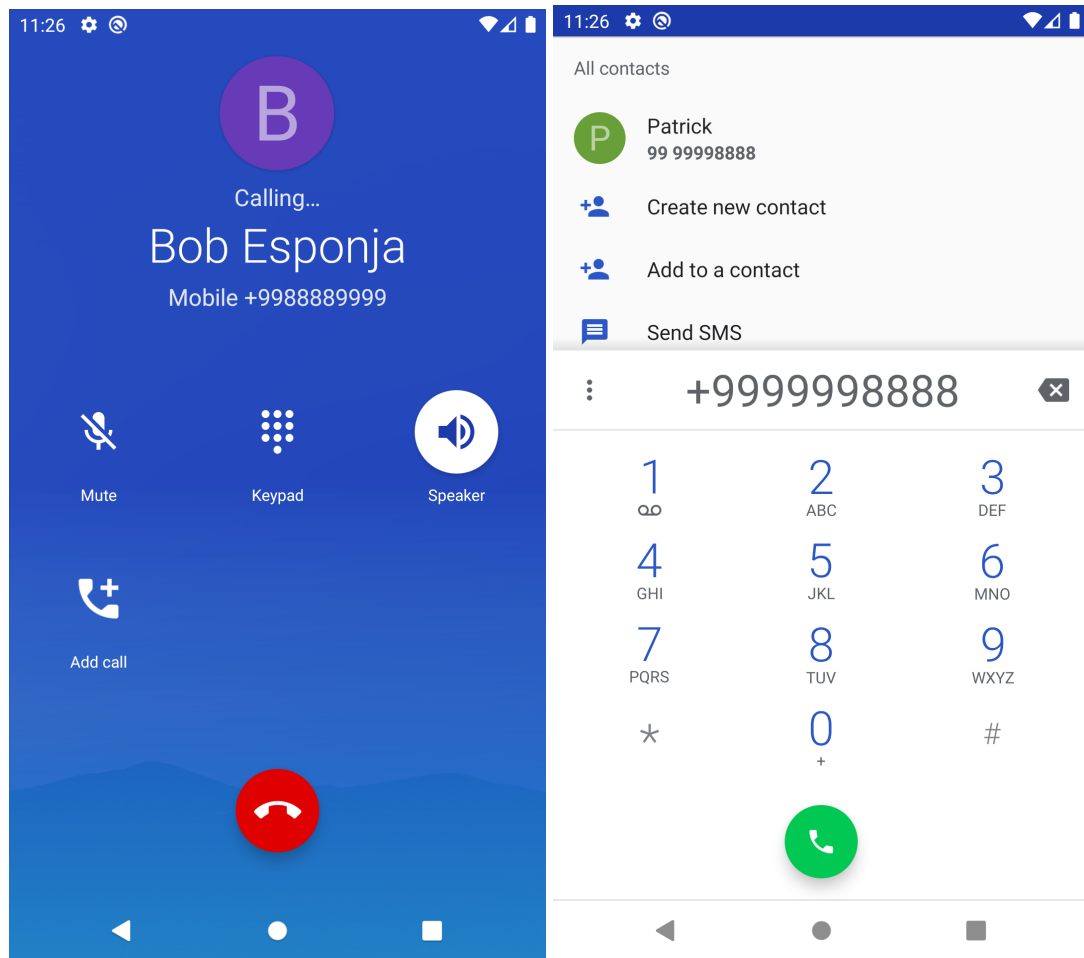
```
public void salvarContato (Contato w){
    SharedPreferences salvaContatos = getSharedPreferences( name: "contatos",Activity.MODE_PRIVATE);

    int num = salvaContatos.getInt( s: "numContatos", i: 0); //checando quantos contatos já tem
    SharedPreferences.Editor editor = salvaContatos.edit();
    try {
        ByteArrayOutputStream dt = new ByteArrayOutputStream();
        ObjectOutputStream oos = new ObjectOutputStream(dt);
        dt = new ByteArrayOutputStream();
        oos = new ObjectOutputStream(dt);
        oos.writeObject(w);
        String contatoSerializado= dt.toString(StandardCharsets.ISO_8859_1.name());
        editor.putString( s: "contato"+(num+1), contatoSerializado);
        editor.putInt( s: "numContatos", i: num+1);
    }catch(Exception e){
        e.printStackTrace();
    }
    editor.commit();
    user.getContatos().add(w);
}
```

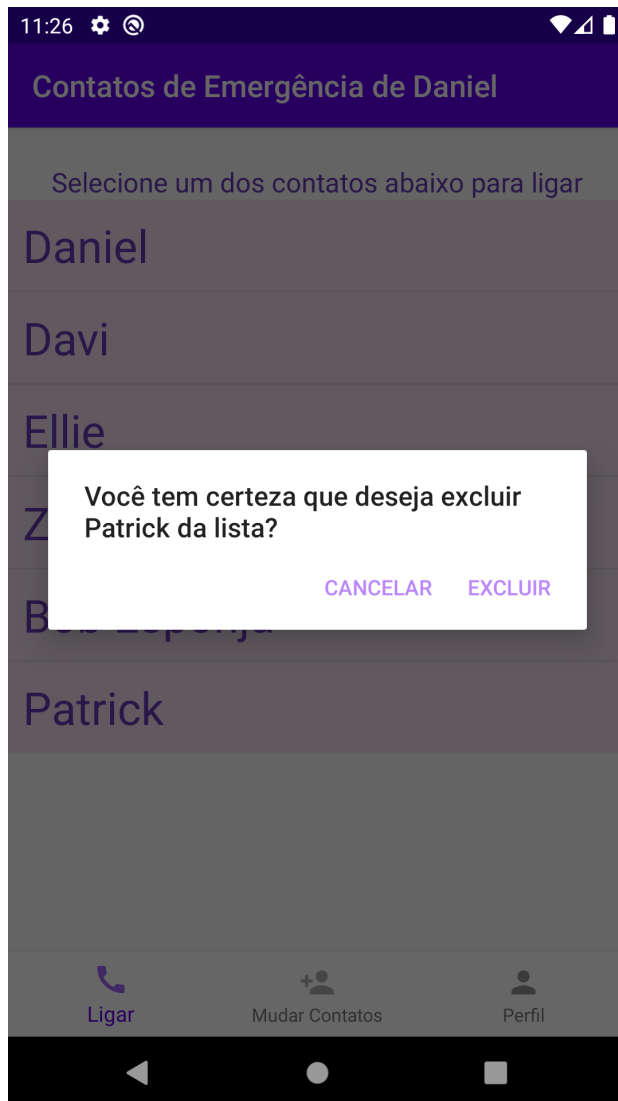
3) Lidar com permissões



Caso o usuário permita o aplicativo ligar diretamente para o contato, a ligação acontece, caso contrário o número do contato vai para o “Dialer”



4) Acesso/busca/alteração da lista de contatos



Repositório:

https://github.com/danielreboucas/Programacao_para_Dispositivos_Moveis/tree/main/T01%20-%20Contatos%20de%20emergencia/Contatos_de_Emergencia