

 | Integrantes - RM :

Augusto Barcelos Barros - 98078

Gabriel Gribl de Carvalho - 96270

Gabriel Souza de Queiroz -98570

Gabriela Zanotto Alves Rodrigues-551629

 | Repositório no GitHub:

<https://github.com/FiapChallenge/Gs/tree/main/Java>

Solução Proposta e Funcionalidades:

O sistema proposto é um aplicativo abrangente para agricultores, que fornece informações e recursos essenciais para melhorar a eficiência e a produtividade nas lavouras. Como informações meteorológicas personalizadas para cidade, e acesso a posts para criar uma comunidade na Agricultura onde todos podem se ajudar de forma mais eficiente.

As funcionalidades presentes são:

- Visualização da temperatura atual, temperatura mínima, umidade, clima, velocidade do vento, hora do nascer e do pôr do Sol com a utilização da API do OpenWeather
- Sistema de cadastro e login
- Seção para visualização, remoção e criação de Posts
- FAQ para dúvidas frequentes
- Seção para sugestão de melhorias no aplicativo

Acreditamos que com uma comunidade mais unida e especializada em dicas na agricultura trará bastante benefícios ao cultivo agrícola, e esse por sua vez, aumentando sua produção, reduzirá drasticamente a fome no mundo.

Funcionalidades a implementar:

- Edição de posts
- Tradução do aplicativo para vários idiomas

Requisitos:

- ☐ Três métodos operacionais que recebam algum parâmetro e retorne algum valor:

1)

```
// Método para obter informações do clima de uma cidade através da API do OpenWeather
You, 2 hours a
public static List<String> getInfo(String city) {
    String apiKey = "3d6dbc169a239f24eada6faa74fc9dc4";
    String encodedCity = city;
    try {
        encodedCity = URLEncoder.encode(city, enc:"UTF-8");
    } catch (Exception e) {
        System.out.println(x:"Erro ao codificar a String cidade");
    }
    String apiUrl = "http://api.openweathermap.org/data/2.5/weather?q=" + encodedCity + "&appid=" + apiKey
        + "&units=metric" + "&lang=pt_br";
    try {
        URL url = new URL(apiUrl);
        HttpURLConnection connection = (HttpURLConnection) url.openConnection();
        connection.setRequestMethod(method:"GET");

        BufferedReader reader = new BufferedReader(
            new InputStreamReader(connection.getInputStream(), StandardCharsets.UTF_8));
        String line;
        StringBuilder response = new StringBuilder();

        while ((line = reader.readLine()) != null) {
            response.append(line);
        }
        reader.close();
        connection.disconnect();

        JSONObject jsonResponse = new JSONObject(response.toString());

        double temperatura = jsonResponse.getJSONObject(key:"main").getDouble(key:"temp");
        double tempMaxima = jsonResponse.getJSONObject(key:"main").getDouble(key:"temp_max");
        double tempMinima = jsonResponse.getJSONObject(key:"main").getDouble(key:"temp_min");
        int umidade = jsonResponse.getJSONObject(key:"main").getInt(key:"humidity");
    }
}
```

2)

```
// Método para buscar um usuário pelo email
public Usuario buscarUsuario(String email) {
    for (Usuario usuario : usuarios) {
        if (usuario.getEmail().equals(email)) {
            return usuario;
        }
    }
    return null;
}
```

3)

```
// Método para exibir o menu principal
public static int menu(Usuario usuario, String menu) {
    String info = "";
    int opcao = JOptionPane.showOptionDialog(parentComponent:null, (info + menu), "AgroSolution - " + usuario.getNome(), JOptionPane.DEFAULT_OPTION,
        JOptionPane.QUESTION_MESSAGE, usuario.getFoto(),
        new String[] { "0", "1", "2", "3", "4", "5", "6", "7" }, initialValue:"1");
    return opcao;
}
```

☐ Um método com sobrecarga e outro com sobrescrita:

Sobrecarga:

```
// Sobrecarga de método
public void addUsuario(String nome, String email, String senha) {
    Usuario usuario = new Usuario(nome, email, senha);
    usuarios.add(usuario);
}

// Sobrecarga de método
public void addUsuario(Usuario usuario) {
    usuarios.add(usuario);
}
```

Sobrescrita:

```
// Sobrescrita de método toString para exibir informações do usuário
@Override
public String toString() {
    return "Usuario [email=" + email + ", nome=" + nome + "]\n";
}
```

