

# Um Sistema de Gestão de Identidades Federadas e Centrado no Usuário alinhado ao Programa de Governo Eletrônico Brasileiro

PIBIC

Bolsista: **André Luiz de Oliveira**  
Orientadora: **Michelle Silva Wingham**

XV Seminário de Iniciação Científica







# Roteiro

---

- ▶ Introdução
- ▶ Objetivos
- ▶ Aplicações E-GOV
- ▶ Material e Métodos
- ▶ Protótipo
- ▶ Experimentos
- ▶ Resultados
- ▶ Considerações finais

# Introdução

---

- ▶ Abertura e transparência dos governos
- ▶ *Open Government*: visa o aumento da participação do cidadão e o envolvimento destes no governo (THIBEAU e REED, 2009)
- ▶ Programas e-Gov:     
- ▶ Heterogeneidade dos procedimentos e dos dados existentes entre admin. central e locais;  
 57º posição do ranking mundial e-Gov  
24º posição do ranking e-Participação

# Introdução

---

- ▶ Programa de Governo Eletrônico:
  - ▶ Democratizar o acesso à informação;
  - ▶ Ampliar discussões;
  - ▶ Dinamizar a prestação de serviços públicos.
- ▶ Criação de um sistema de **identificação**, de **autenticação** e de **autorização** de usuários;
- ▶ Sistemas de Gestão de Identidades:
  - ▶ Ferramentas para o gerenciamento de identidades no mundo digital.

# Sistemas de Gestão de Identidades (GId)

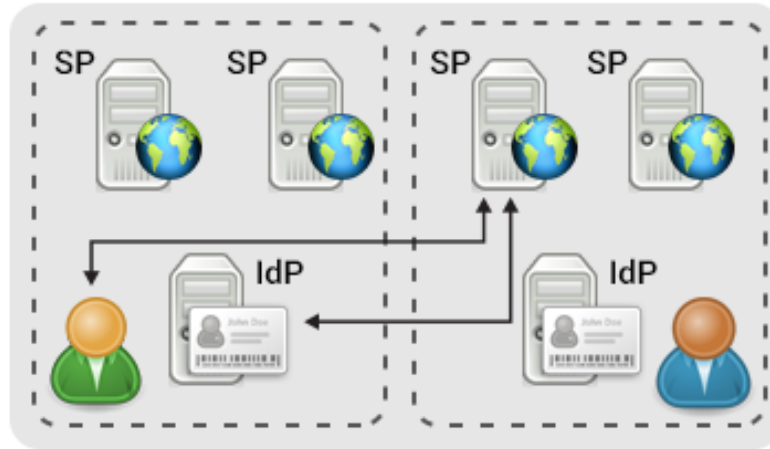
---

- **Usuário:** aquele que deseja acessar algum serviço;
- **Identidade:** conjunto de atributos de um usuário, que pode ser seu nome, endereço, filiação, data de nascimento, etc;
- **Provedor de Identidades (*Identity Provider* – IdP):** responsável por fazer a gestão da identidade de um usuário. Após o usuário passar por um processo de autenticação, este recebe uma credencial, dita identidade, que é reconhecida como válida pelos provedores de serviço;
- **Provedor de Serviços (*Service Provider* – SP)** oferece recursos a usuários autorizados, após verificar a autenticidade de sua identidade e após comprovar que a mesma carrega todos os atributos necessários para o acesso.

# Modelos de GId



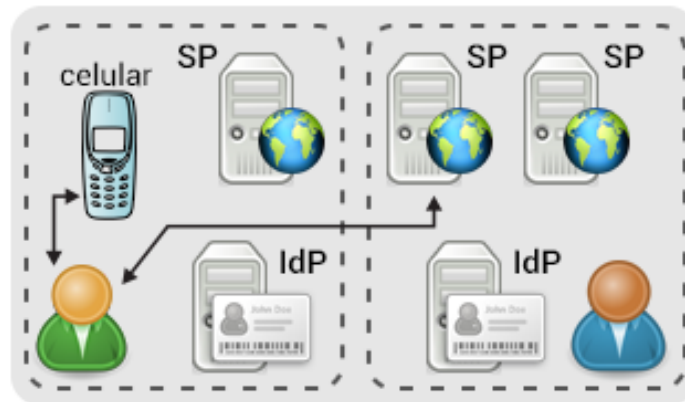
(a) Tradicional



(b) Federado

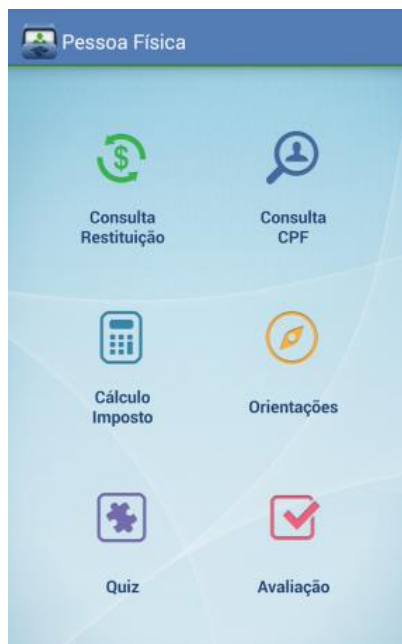


(c) Centralizado



(d) Centrado no usuário

# Aplicações E-GOV



# Objetivo Geral

---

- ▶ Prover a **gestão de identidades** adequada ao programa GOV.BR, por meio do desenvolvimento de um **protótipo de um sistema** de gestão identidades **federadas e centrado** no usuário, baseado no padrão **OpenId Connect**;

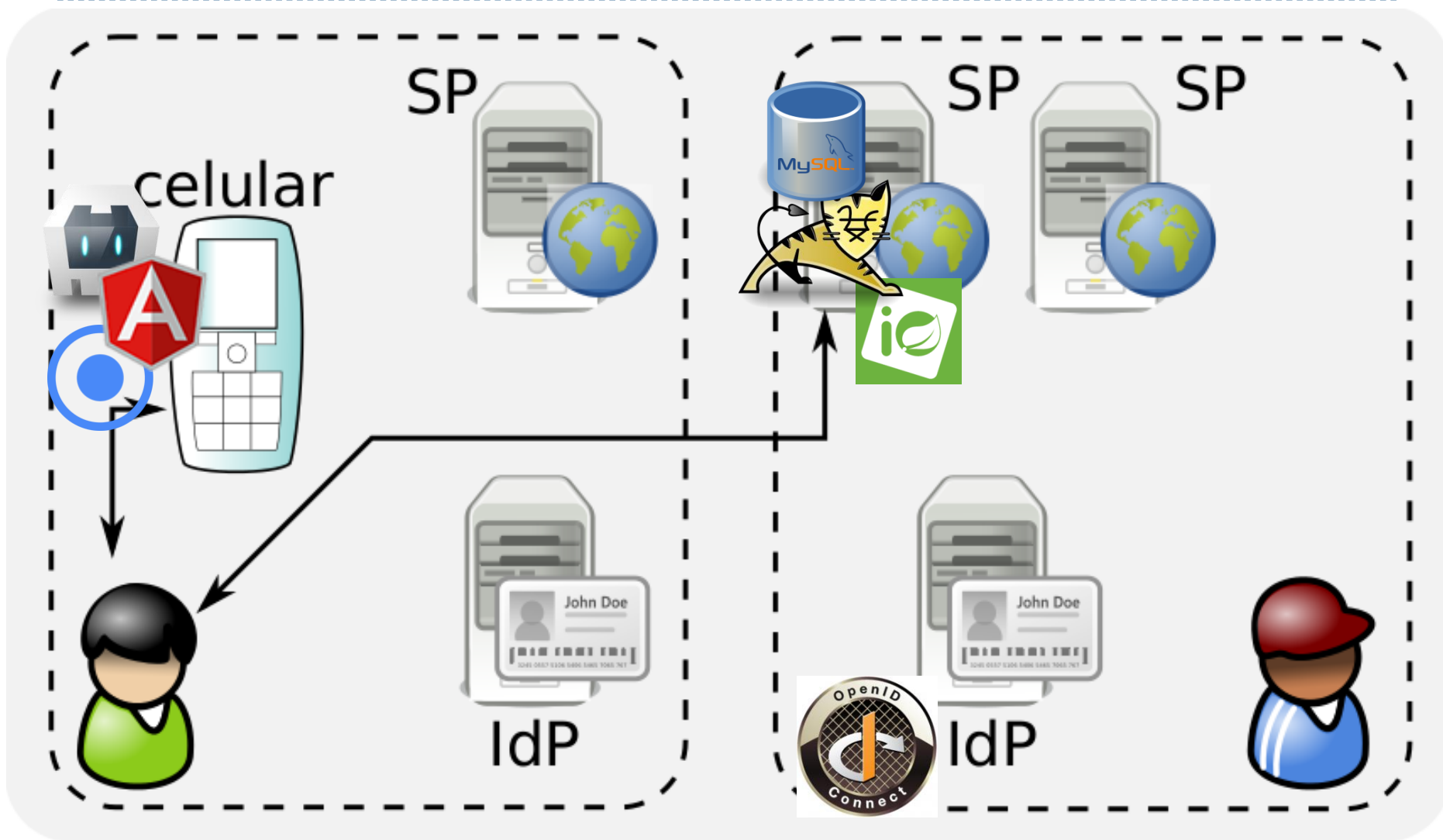


# Materiais e Métodos

---

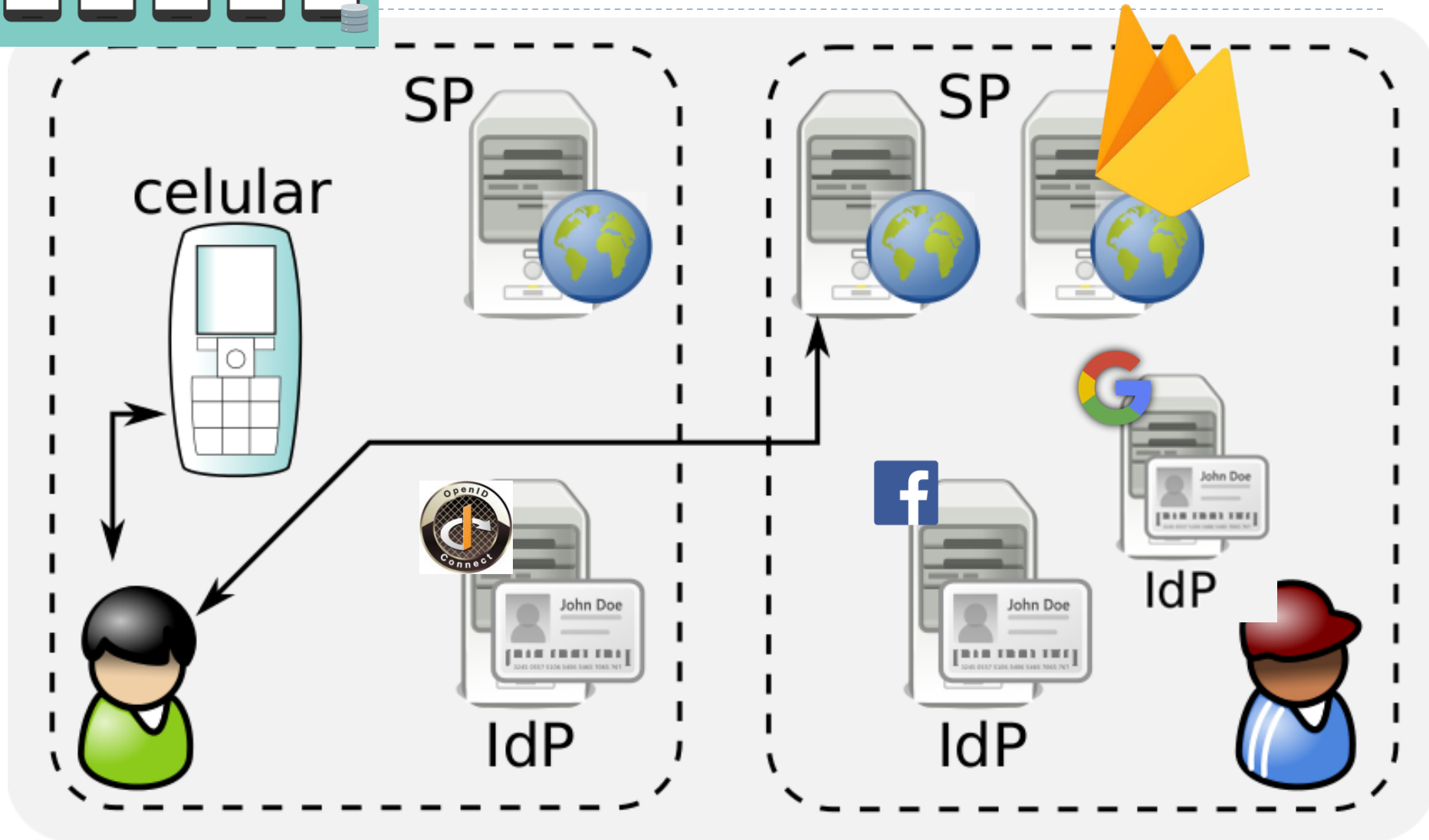
- ▶ Estudo bibliográfico:
  - ▶ análise das estratégias nacionais de Gld adotadas em outros países
  - ▶ identificar as soluções tecnológicas amplamente aceitas nestes países
- ▶ Modelagem do sistema de gestão de identidade centrado no usuário, com a elaboração de diagramas e especificações com a linguagem UML
- ▶ Implementação de um protótipo com prova de conceitos.
- ▶ Avaliação do protótipo da solução (integrado a uma aplicação de eGov).

# Arquitetura do Protótipo





# Experimentos Firebase Auth



# Protótipo: Telas

The image displays three mobile application screen prototypes, each with a status bar at the top showing various icons and a time.

**Screen 1 (Left):** Titled "Login" with a "Fechar" button. It features input fields for "Email" (containing "andre") and "Senha" (containing "\*\*\*\*\*"). A blue "Entrar" button is positioned below the fields. At the bottom, it shows "URL Servidor" as "https://100.65.72.236:8443".

**Screen 2 (Middle):** Also titled "Login". It has "Email" and "Senha" input fields, followed by a blue "Entrar" button. Below this is a section labeled "OU" with three blue buttons for social media login: "g" (Google), "fok" (Facebook), and a Twitter icon. The "URL Servidor" at the bottom is "https://127.0.0.1:8443".

**Screen 3 (Right):** Titled "Ocorrências" with a menu icon on the left. It includes a blue "Adicionar" button. Below is a list of incidents, each with a pencil icon for editing and a trash icon for deletion:

- Padaria roubada
- Tiroteio
- Fugitivo na rua
- Briga de comerciantes

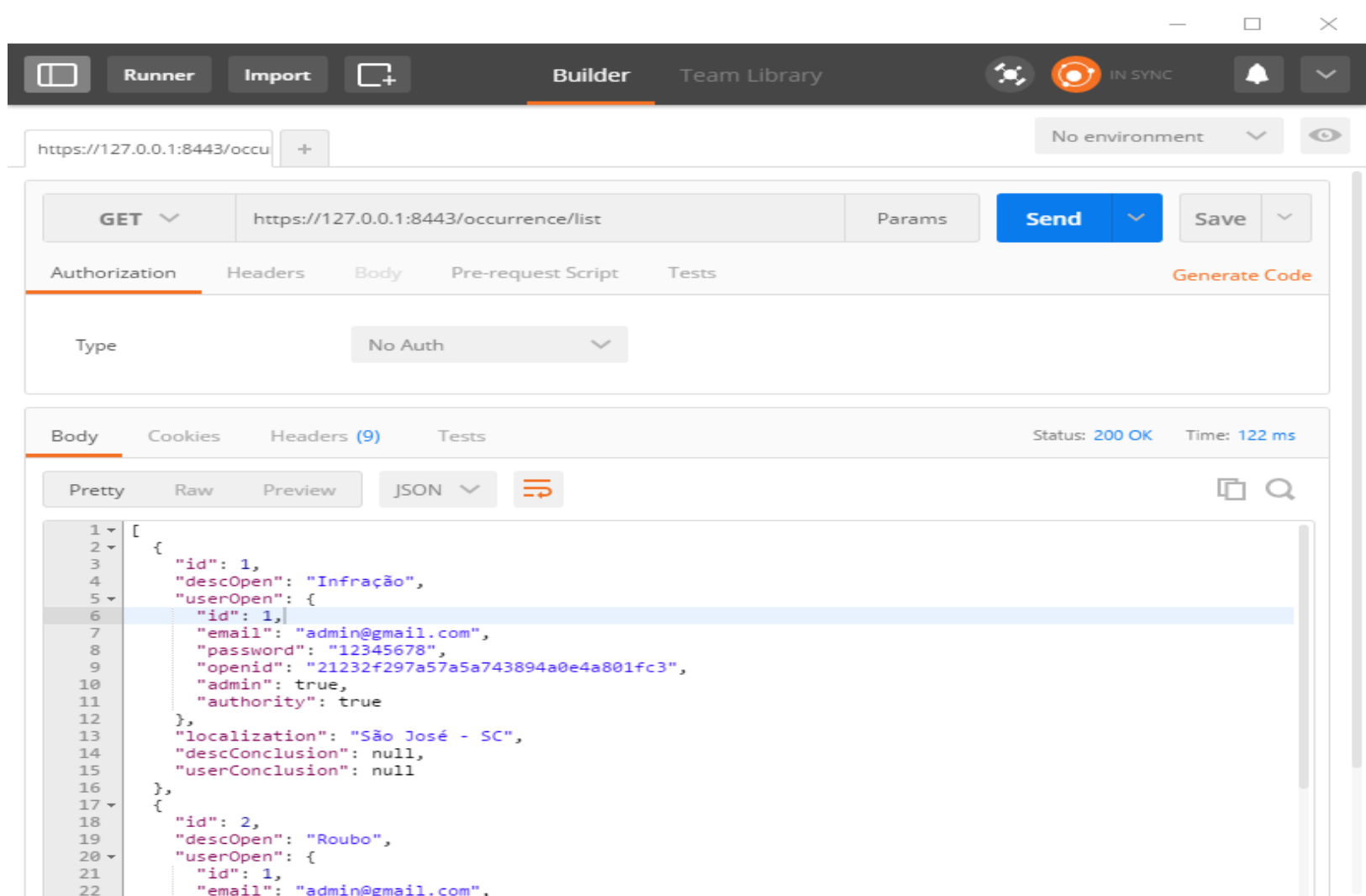
# Protótipo: Serviço

---

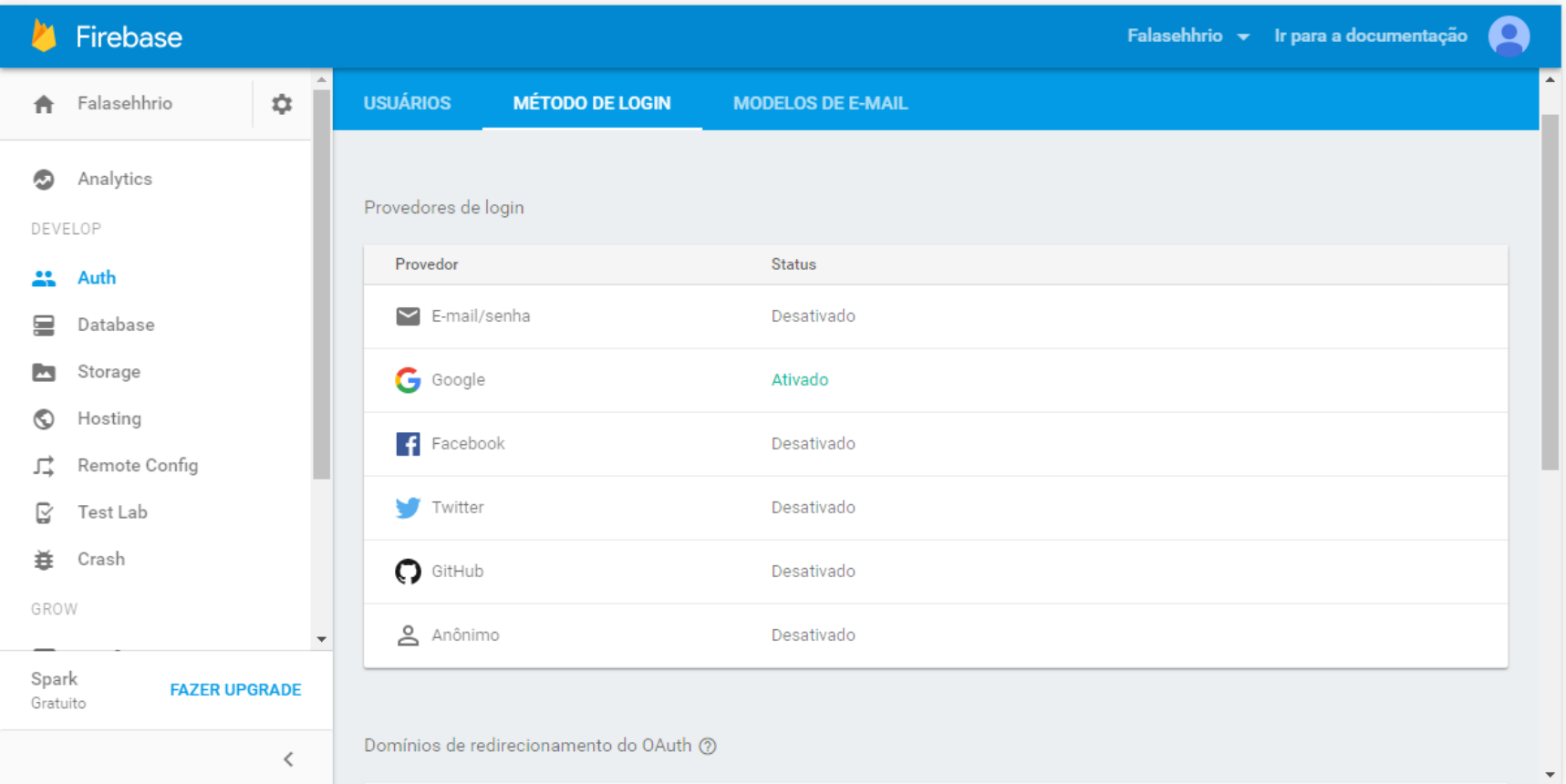
```
@RequestMapping(value="/save",method=RequestMethod.POST)
public ResponseEntity<Void> save(@RequestBody Occurrence m) {
    try {
        log.info("init");
        db.save(m);
        return new ResponseEntity<>(HttpStatus.OK);
    } catch (ModelException e) {
        log.error("Error: " + e.getMessage());
        return new ResponseEntity<>(HttpStatus.INTERNAL_SERVER_ERROR);
    }
}

@RequestMapping("/list")
public ResponseEntity<List<Occurrence>> list() {
    try {
        log.info("init");
        List<Occurrence> list = db.list();
        if(list != null && list.size()>0){
            return new ResponseEntity<List<Occurrence>>(list, HttpStatus.OK);
        } else {
            return new ResponseEntity<List<Occurrence>>(HttpStatus.NOT_FOUND);
        }
    } catch (ModelException e) {
        log.error("Error: " + e.getMessage());
        return new ResponseEntity<>(HttpStatus.INTERNAL_SERVER_ERROR);
    }
}
```

# Protótipo: Postman Rest



# Experimentos: Firebase Auth

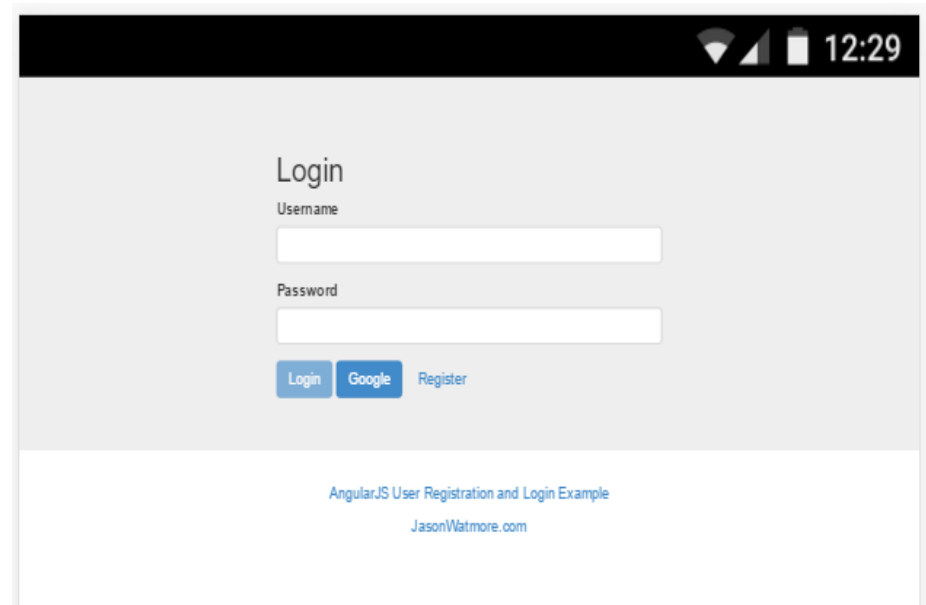
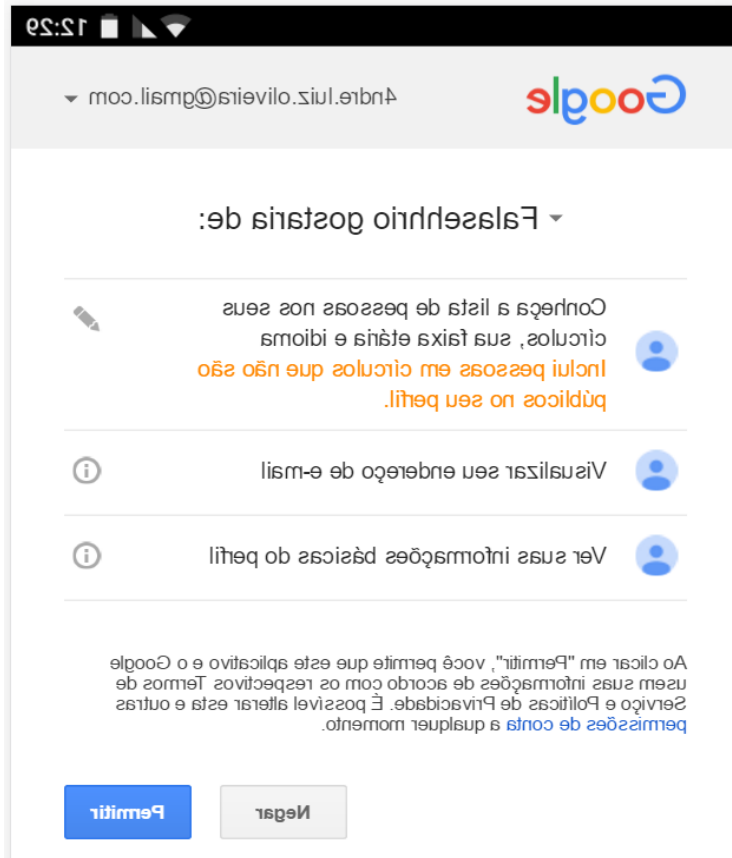


The screenshot shows the Firebase console interface. The top navigation bar includes the Firebase logo, the user name 'Falasehhrio', a dropdown arrow, a link to 'Ir para a documentação', and a user profile icon. The left sidebar contains a list of services: Analytics, DEVELOP (with a sub-item 'Auth' highlighted), Database, Storage, Hosting, Remote Config, Test Lab, Crash, and GROW. At the bottom of the sidebar, it says 'Spark Gratuito' and 'FAZER UPGRADE'. The main content area has three tabs: 'USUÁRIOS', 'MÉTODO DE LOGIN' (selected), and 'MODELOS DE E-MAIL'. Under the 'MÉTODO DE LOGIN' tab, there is a section titled 'Provedores de login' containing a table of login providers.

Provedor	Status
E-mail/senha	Desativado
Google	Ativado
Facebook	Desativado
Twitter	Desativado
GitHub	Desativado
Anônimo	Desativado

Below the table, there is a section titled 'Domínios de redirecionamento do OAuth' with a help icon.

# Experimentos: Autenticação Google





# Protótipo: Provedor de Identidade MITREid

The screenshot displays the MITREid OpenID Connect Server web interface. The top navigation bar includes links for Home, About, Statistics, and Contact, along with a user profile dropdown for 'admin'. A left sidebar categorizes navigation options into Administrative (Manage Clients, Whitelisted Clients, Blacklisted Clients, System Scopes), Personal (Manage Approved Sites, Manage Active Tokens, View Profile Information), and Developer (Self-service client registration, Self-service protected resource registration). The main content area, titled 'Home / Manage Clients', features a 'Refresh' button, a '+ New Client' button, and a search bar. Below this, a table lists registered clients. The first entry is 'Simple Web App', which is highlighted with a blue icon and a '1' in a blue square. Its information includes the URL 'http://localhost:8080/simple-web-app/openid\_connect\_login' and a registration time of '6 hours ago'. Action buttons for 'address', 'phone', 'openid', 'email', and 'profile' are provided for this client. To the right of the client entry are buttons for 'Edit', 'Whitelist', and 'Delete'. The interface is clean and modern, with a light gray background and blue accents.

OpenID Connect Server Home About Statistics Contact admin

ADMINISTRATIVE

- Manage Clients
- Whitelisted Clients
- Blacklisted Clients
- System Scopes

PERSONAL

- Manage Approved Sites
- Manage Active Tokens
- View Profile Information

DEVELOPER

- Self-service client registration
- Self-service protected resource registration

Home / Manage Clients

Refresh + New Client Search...

Client	Information	
1 Simple Web App	http://localhost:8080/simple-web-app/openid_connect_login address phone openid email profile Registered 6 hours ago	Edit Whitelist Delete

Refresh + New Client

Powered by MITREid Connect 1.5.5-01180805 © 2016 The MITRE Corporation and MIT Internal Test Corporation

# Protótipo: Provedor de Identidade MITREid

MITREid Connect: Simple Web App

## Log In


Use this page to log in by entering an **issuer URI** or a **webfinger identifier**. Use the buttons to pre-fill the form with a known identifier.

Local MITREid  
Connect Server  
(default setup)

mitre.org  
integration site  
demo user

Log In

# Protótipo: Provedor de Identidade MITREid

 OpenID Connect Server   Home   About   Statistics   Contact   admin

## Approval Required for *Simple Web App*

**Caution:**  
This client was dynamically registered .  
It has been approved 1 time previously.

You will be redirected to the following page if you click Approve:  
[http://localhost:8080/simple-web-app/openid\\_connect\\_login](http://localhost:8080/simple-web-app/openid_connect_login)

Access to:

- ☒ log in using your identity
- ☒ basic profile information
- ☒ email address
- ☒ physical address
- ☒ telephone number

Remember this decision:

- ☒ remember this decision until I revoke it
- ☐ remember this decision for one hour
- ☐ prompt me again next time

**Do you authorize "Simple Web App"?**

AuthorizeDeny

# Protótipo: Provedor de Identidade MITREid

MITREid Connect: Simple Web App

HomeUserAdminLogout

admin

## Hello Demo Admin

This page requires that the user be logged in with a valid account and the `ROLE_USER` Spring Security authority. If you are reading this page, you are currently logged in.

The authorization provider will create a Principal object based on the `iss` and `sub` claims associated with your ID token. This value can be used as a globally unique username within the application (though it's not meant to be human-readable). Your Principal is:

```
{sub=90342.ASDFJWFA, iss=http://localhost:8080/openid-connect-server-webapp/}
```

The authorization provider will assign your account a set of authorities depending on how it's configured. Your current login has the following Spring Security authorities:

- `ROLE_USER`
- `ROLE_ADMIN`
- `OIDC_90342.ASDFJWFA_http://localhost:8080/openid-connect-server-webapp/`

## ID Token

Your ID Token has the following set of claims:

Name	Value
sub	90342.ASDFJWFA
aud	5e9c0311-6463-454d-9dbf-45841aff1b66
kid	rsa1
iss	http://localhost:8080/openid-connect-server-webapp/
exp	Mon Aug 22 2016 01:44:01 GMT-0300 (Hora oficial do Brasil)
iat	Mon Aug 22 2016 01:34:01 GMT-0300 (Hora oficial do Brasil)

# Considerações finais

---

- ▶ Experimentos de integração do aplicativo mobile com servidor de autenticação não foi concluída;
- ▶ Aplicação mobile:
  - ▶ integrar com a câmera
  - ▶ localização do dispositivo
- ▶ Pesquisar sobre integração do MITREid à plataforma Firebase que respeite a política E-GOV

# Considerações finais

---

- ▶ Modelo centrado no usuário
  - ▶ Governo participativo;
  - ▶ Selecionar quais informações deseja liberar aos provedores de serviços;
- ▶ Mobile First
- ▶ Experimentos
  - ▶ Plataforma MITREid
  - ▶ Plataforma Firebase Auth
- ▶ Experimentos de integração do aplicativo mobile com servidor de autenticação não foi concluída
- ▶ Aplicação mobile dar continuidade e implementação de melhorias:
  - ▶ integrar com a câmera
  - ▶ localização do dispositivo
- ▶ Pesquisar sobre integração do MITREid à plataforma Firebase
- ▶ 20 que respeite a política E-GOV