



## Grupo de Estudo de Sistemas de Informação e Telecomunicação para Sistemas Elétricos - GTL

### Uso da Tecnologia Móvel para o Acesso a Dados de Tempo Real do Sistema Elétrico

Rafael Luiz Rezende Bastos (\*)  
UFRJ

Marcelo Rosado da Costa  
CEPEL

Carmen Lucia Tancredo Borges  
COPPE /UFRJ

Felipe Vieira Duval  
CEPEL

## RESUMO

Este trabalho tem por objetivo apresentar o desenvolvimento de um aplicativo de celular para o sistema operacional *Android* que permite acessar de maneira móvel as informações de tempo real disponibilizadas pelo Sistema Aberto de Gerenciamento de Energia (SAGE), que é um sistema utilizado para gerenciar e supervisionar redes de energia elétrica. A criação de um aplicativo móvel para o SAGE se tornou viável, pois as novas atualizações do sistema de gerenciamento passaram a ter entre seus protocolos de comunicação o protocolo OPC UA (*Open Platform Communications Unified Architecture*), e assim cada SAGE poderá a ser também um servidor de dados do sistema elétrico sob sua supervisão.

## PALAVRAS-CHAVE

SAGE, OPC UA, Android, Indústria 4.0, Mobilidade

### 1.0 - INTRODUÇÃO

O mundo está adentrando a era da Indústria 4.0, na qual uma densa malha de informações está se criando permitindo a comunicação entre pessoas e pessoas, entre pessoas e máquinas e entre máquinas e máquinas [1]. Essa malha de informações surge a partir do momento no qual tanto pessoas quanto máquinas e sistemas passam a ser equipados com ferramentas capazes de trocar informação de maneira remota e móvel.

Atualmente, uma das principais ferramentas que confere às pessoas esse caráter de mobilidade e conectividade são os aparelhos *smartphones*, pois eles são capazes de acessar redes de *internet* e são dispositivos portáteis, ou seja, são capazes de integrar uma rede de informações de praticamente qualquer lugar que tenha cobertura de *internet*.

Mas, afinal, porque não trazer este conceito de conectividade e mobilidade para as áreas de operação e manutenção do sistema elétrico de potência? Qual seria a utilidade de desenvolver um aplicativo para *smartphone* capaz de acessar dados de tempo real de um sistema elétrico? A quem interessaria ser capaz de acessar dados elétricos pelo celular?

Bom, algumas situações ilustram a utilidade de um aplicativo com essa funcionalidade como, por exemplo, o caso de uma equipe de manutenção que precisa de um meio para acessar localmente informações de tempo real de um determinado equipamento, identificando dados como informações acerca da corrente elétrica atual, da corrente elétrica máxima e da corrente elétrica mínima do equipamento. Ou ainda, a situação de um operador que precisaria acompanhar de fora da sala de operação o nível de tensão em uma determinada barra caso acontecesse algum problema no sistema elétrico em um momento no qual ele estivesse longe de sua estação de trabalho.

Pensando em casos assim, percebe-se que um aplicativo para celular capaz de prover o acesso a informações de tempo real seria então uma ferramenta muito útil para o setor elétrico, provendo mais conectividade e mobilidade na relação usuário x sistema, qualidades fundamentais da Indústria 4.0.

Este trabalho tem por objetivo apresentar o uso da tecnologia móvel através do desenvolvimento de um aplicativo de celular para o sistema operacional Android e que acesse as informações de tempo real disponibilizadas em um servidor OPC UA (*Open Platform Communications Unified Architecture*) do Sistema SAGE (Sistema Aberto de Gerenciamento de Energia), permitindo a mobilidade de acesso seguro, confiável e contínuo à informação ao usuário a qualquer momento e em qualquer lugar.

## 2.0 - O SAGE E O PROTOCOLO OPC UA

O Sistema Aberto de Gerenciamento de Energia (SAGE) é um sistema computacional desenvolvido pelo Centro de Pesquisas de Energia Elétrica (CEPEL) com o objetivo de executar funções de gerenciamento de energia em sistemas elétricos de potência. O sistema SAGE é composto por um sistema SCADA (*Supervisory Control and Data Acquisition*) e pelas funções de Análise de Redes (Configurador de Redes, Estimação de Estado, Análise de Contingência, Fluxo de Potência do Operador).

As funções de Análise Estática de Redes do SAGE EMS (*Energy Management System*) têm por objetivo monitorar a condição operativa corrente do sistema elétrico, fornecendo ao operador uma estimativa confiável do estado do sistema, informando quando da ocorrência de condições operativas não desejáveis e produzindo estratégias de controle que permitam alterar o ponto de operação para uma condição operativa normal. [2].

Recentemente, o CEPEL desenvolveu nativamente o protocolo padrão OPC UA no sistema SAGE, o qual é um dos padrões que viabilizam a integração deste sistema à Indústria 4.0, ou IloT (*Industrial Internet of Things*). Este protocolo de comunicação industrial viabiliza a interoperabilidade para troca de dados de forma confiável e segura entre diferentes dispositivos da automação, permitindo o acesso remoto aos dados de tempo real do sistema supervisionado por meio da rede de computadores.

### 2.1 As estruturas básicas do protocolo OPC UA

As unidades fundamentais que organizam e armazenam as informações disponibilizadas em um servidor OPC UA são os chamados nós. Os nós são estruturas armazenadoras de dados que se conectam entre si por meio de estruturas chamadas referências. Essas referências são os *links* que interligam os diversos nós de informação presentes no servidor e que permitem a organização de uma rede complexa composta por muitos nós [3].

Os nós podem ser de diferentes classes, sendo as principais as classes chamadas *Object*, *Variable* e *Method*. Os nós de classe *Object* são nós com função de agrupar outros nós que se referem a algo em comum. Nós do tipo *Object* funcionam analogamente às pastas de um sistema que agrupam arquivos relacionados a um tema comum. Nesse sentido de criar um agrupamento organizado de informações, os nós *Objects* não tem a finalidade de armazenar valores, mas sim de aglutinar, por meio das referências, outros nós que sejam relacionados a algo comum. Os nós de classe *Variable*, por sua vez, têm como função o armazenamento de valores, como já sugere o seu nome. Já os nós de classe *Method* tem a função de acionar alguma função, executada sobre algum outro nó [3].

Para ilustrar a implementação desses conceitos, um exemplo simplificado pode ser apresentado para o caso da modelagem de um disjuntor. Para modelar as informações de um disjuntor do sistema elétrico por meio do OPC UA, uma abordagem seria criar um nó de classe *Object* para representar o equipamento disjuntor e agrupar as informações relacionadas a esse disjuntor. Para armazenar o valor do estado desse disjuntor (aberto ou fechado) seria criado um nó de classe *Variable* responsável por armazenar o estado atual do disjuntor. E, para disponibilizar a função de enviar comandos de abertura e fechamento para o disjuntor, seria criado um nó de classe *Method* responsável por executar a função de abertura ou fechamento sobre o disjuntor representado. Tendo esses três nós e nomeando-os, por exemplo, por Disjuntor-1 (*Object*), Estado-Disjuntor-1 (*Variable*) e Controle-Disjuntor-1 (*Method*), seriam criadas referências para fazer o *link* entre esses nós, de forma a se criar um modelo estruturado e organizado do disjuntor de interesse. A Figura 1 traz uma ilustração para representar essa modelagem simplificada de um disjuntor do sistema elétrico.

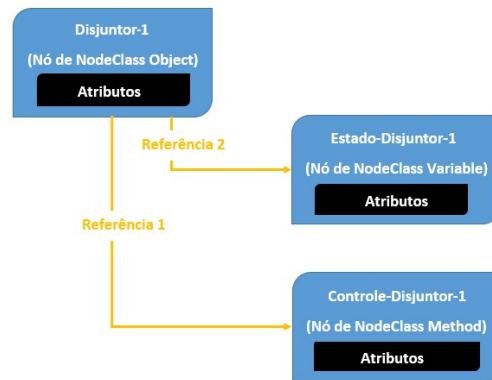


FIGURA 1 – Modelagem simplificada de um disjuntor usando a estrutura do protocolo OPC UA.

Os nós possuem diferentes atributos, que representam características próprias desses nós. Os atributos variam conforme a classe dos nós, mas alguns existem em todos eles, dentre os quais cabe destacar o atributo *NodeId*. Esse atributo representa a identificação do nó, que é única em todo o servidor e sem a qual não se pode realizar nenhuma ação sobre os nós. Leitura de dados, escrita de dados, acesso às referências dos nós, para tudo isso é necessário saber o identificador do nó de interesse.

A interação com os nós, as ações de leitura, de navegação acessando as referências entre os nós, a execução de um método em um nó, tudo isso é possível por meio dos chamados Serviços. O OPC UA é baseado em uma arquitetura orientada a serviços então as principais ações executadas para interação com o servidor são realizadas por meio de serviços, que podem ser entendidos como funções criadas para realizar determinadas tarefas de uma forma padronizada. Dessa forma, o servidor dispõe de serviços para conexão, para leitura de valores, para navegação pelos nós do servidor, etc.

Conhecer esses conceitos básicos é fundamental para o desenvolvimento de um aplicativo capaz de interagir com um servidor OPC UA. A função do aplicativo é atuar como um cliente capaz de se comunicar com o servidor utilizando os serviços disponibilizados por ele para navegar pelos seus nós, ler valores de interesse, etc. Para o aplicativo cliente o interesse não é o de modelar dados, mas sim de, conhecendo a estrutura e a organização do padrão de comunicação, e sabendo utilizar os serviços disponíveis, poder acessar as informações de interesse modeladas e organizadas pelo servidor.

### 3.0 - O APLICATIVO DE CELULAR DO SAGE

A ideia de se desenvolver um aplicativo para *smartphone* no próprio CEPEL com capacidade de se comunicar com o SAGE surgiu com o intuito de fazer o SAGE caminhar em direção à tendência global de acessibilidade remota e móvel da Indústria 4.0. O desenvolvimento de um aplicativo nativo viabiliza a implementação de funções mais específicas, que possam atender às demandas de um usuário SAGE, pois atualmente os aplicativos desse segmento que existem no mercado são de uso genérico. Assim, desenvolver um aplicativo próprio permitiria focar em funcionalidades que facilitassem a vida de um usuário SAGE.

As ferramentas utilizadas no desenvolvimento do aplicativo de celular para o sistema operacional Android foram:

- Android Studio como ambiente de desenvolvimento do aplicativo;
- SDK Android para dispor de todas as ferramentas necessárias à infraestrutura da aplicação;
- Biblioteca disponibilizada pela OPC Foundation, a OPC UA Java Stack para as funções relacionadas ao protocolo de comunicação;
- Pacotes de APIs do Spongy Castle, para os recursos de criptografia;
- SLF4J Android para funções relacionadas ao logging.

#### 3.1 Arquitetura de comunicação

A arquitetura de comunicação do SAGE através do OPC UA contempla o acesso as informações da função SCADA e EMS do SAGE. A função SCADA obtém os dados dos equipamentos de campo do sistema elétrico e a função EMS usa esses dados do SCADA para produzir seus resultados. As informações do sistema elétrico obtidas pelo SCADA e processadas pelo EMS ficam disponíveis para acesso por meio do servidor OPC UA presente no SAGE. O cliente OPC UA (o aplicativo) pode então acessar os dados do sistema elétrico armazenados no servidor e também solicitar o envio de comandos aos equipamentos de campo por meio do servidor OPC UA presente no SAGE. A partir da solicitação de envio de comando, o servidor utiliza então a função SCADA para emitir esse comando aos equipamentos de campo do sistema elétrico. Essa comunicação cliente-servidor se dá por meio do protocolo OPC UA, utilizando os serviços disponibilizados pelo servidor para realizar a conexão, a navegação pelos nós do sistema, a leitura de valores, o envio de comando por meios dos métodos destinados a essa função, etc. A Figura 2 ilustra essa arquitetura de comunicação

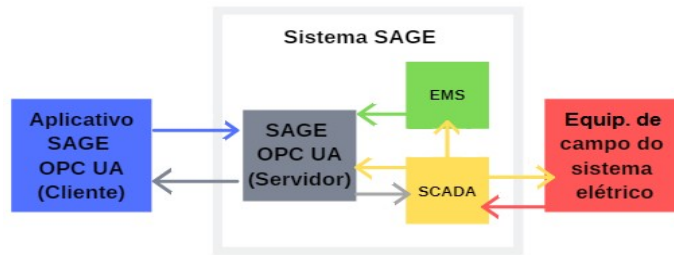


FIGURA 2 – Arquitetura de comunicação.

### 3.2 Segurança

Para garantir a questão da segurança de acesso, o aplicativo implementa um modo de conexão credenciado. Somente com uma conexão credenciada, utilizando login e senha criados pela pessoa que ficar responsável por gerenciar o servidor, é possível interagir com funções mais sensíveis, como o envio de comandos de controle supervisorio. Além do fato de precisar ser credenciado, outra ferramenta de manutenção da segurança de acesso é a existência de permissões de usuário que existe no próprio SAGE, ou seja, não basta o cliente que acessar o servidor ser um cliente cadastrado no servidor, ele precisa ter recebido as permissões e privilégios para executar determinadas ações.

A segurança do tráfego de informações também é outro requisito atendido pelo aplicativo na comunicação cliente-servidor. O fluxo de informações se dá de forma assinada e criptografada, fazendo uso de certificados digitais e de chaves privadas. Assim, só o cliente e o servidor conseguem ler as informações trocadas, pois durante o trajeto entre cliente e servidor as informações fluem de forma criptografada, ilegível, evitando o acesso de terceiros que venham a tentar interceptar a informação nesse percurso intermediário.

### 3.3 Telas e funções do aplicativo

A estrutura do aplicativo foi planejada para promover uma navegação intuitiva e direta, com o design da aplicação pensado de forma que os valores fossem apresentados de maneira clara e com ênfase na informação. Para atingir esse objetivo, as funcionalidades da aplicação foram listadas em uma gaveta de navegação lateral permitindo que o acesso à informação fosse feito de maneira prática a partir de qualquer tela do aplicativo. Outro acesso poderá ser através do código QR (*Quick Response*).

As funções tradicionais de navegação dos clientes OPC UA genéricos foram implementadas no aplicativo por meio de uma árvore hierárquica de nós de informações, onde pode-se obter a leitura dos valores, das descrições das entidades e dos atributos presentes no sistema SAGE. Além disso, algumas funções personalizadas foram implementadas para a utilização em conjunto com o SAGE e serão descritas a seguir.

As telas descritas a seguir apresentarão dados simulados na base de demonstração de um SAGE executado num computador nas instalações do Cepel e com acesso à internet. O servidor OPC UA está ativo nesta máquina.

#### 3.3.1 Conexão

A tela de conexão é o meio a partir do qual o usuário vai dizer ao aplicativo qual é o servidor SAGE OPC UA o qual ele deseja se conectar. O aplicativo permite dois modos de tentativa de conexão, a conexão anônima (somente leitura) ou a conexão identificada. A Figura 3 ilustra o preenchimento da tela para esses dois modos de conexão.



FIGURA 3 – Conexão Identificada x Conexão Anônima.

Após preencher os dados de conexão e apertar o botão de conectar, caso a conexão seja bem sucedida, aparecerá a mensagem "Conectado" na tela. Caso contrário, aparecerá a mensagem "Não conectado".

### 3.3.2 Favoritos

Durante a navegação pelo sistema, o usuário é capaz de seleccionar quais os pontos são principais para ele e pode então adicionar esses pontos à sua lista de Favoritos. A tela Favoritos exibe essa lista de pontos de interesse com seus respectivos valores. Esses valores são atualizados por padrão a cada segundo, mas é possível aumentar esse tempo de atualização por meio da opção "Tempo de Refresh" que é disponibilizada quando o usuário clica no ícone de menu da tela de Favoritos, localizado no canto superior direito da tela. A Figura 4-(a) mostra a tela de Favoritos e o menu dessa tela apresentando a opção de mudar o tempo de atualização de valores, respectivamente.

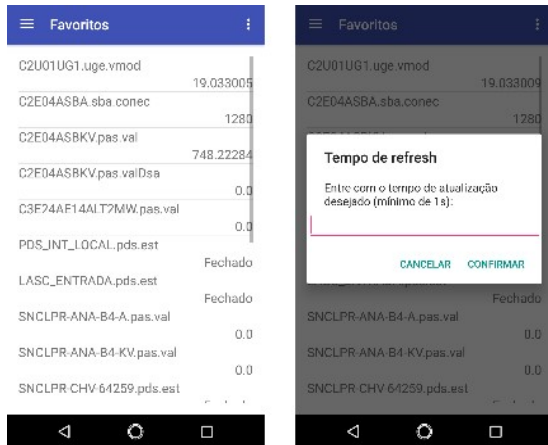


FIGURA 4 - (a) – Favoritos x Tempo de refresh.



FIGURA 4 – (b) – Edição de Favoritos x gráfico

A partir da tela de favoritos é possível ver detalhes dos pontos que estão nessa lista ao clicar em algum ponto de interesse. Ao fazer isso, o usuário é direcionado à tela de edição de favoritos, conforme notado na Figura 4-(b). Essa tela de edição permite:

- Ler a descrição do ponto;
- Inserir, alterar ou remover as coordenadas geográficas do ponto para georreferenciamento;
- Remover o ponto da lista de Favoritos;
- Visualizar um gráfico de tempo real da medida;
- Enviar comando de controle supervisão caso o usuário tenha as permissões para tal.

### 3.3.3 Pesquisar Equipamentos

A tela Pesquisar Equipamentos é por onde o usuário consegue acessar as listas de equipamentos geradas e fazer pesquisas nominais dentro dessas listas. No topo da tela há um filtro para seleccionar qual lista de equipamentos o usuário deseja acessar. O usuário dispõe também de um filtro nominal dinâmico que fica logo abaixo do filtro de tipos de equipamentos. Conforme o usuário vai escrevendo o nome do identificador do ponto de interesse no filtro, o programa vai filtrando dinamicamente a lista de equipamentos selecionada, apresentando apenas os nomes que contém os caracteres que estão sendo digitados. Assim, torna-se muito mais fácil achar pontos de interesse no sistema. A Figura 5 mostra o filtro de equipamentos com as opções de todos os tipos de equipamentos disponíveis para busca, e mostra um exemplo de uso do filtro dinâmico fazendo a seleção apenas dos geradores que têm "g1" em seu identificador.

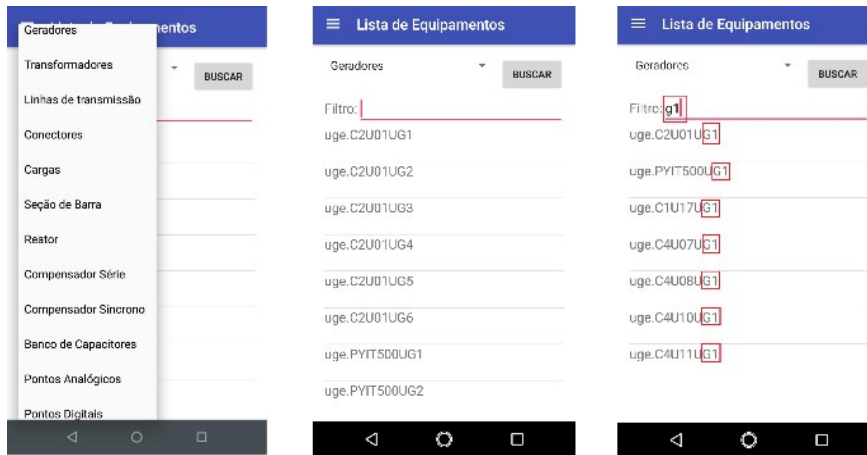


FIGURA 5 – Filtro de equipamentos e filtro nominal

Após localizar o ponto de interesse e clicar sobre ele, o usuário é direcionado a uma tela contendo os atributos desse equipamento e seus respectivos valores, que são atualizados a cada segundo. Caso seja um ponto digital com função de controle supervisor, surge também na parte inferior da tela um botão para envio de comando de controle supervisor, função disponível apenas para usuários autorizados. Caso o usuário clique em algum dos atributos, surge uma nova tela apresentando o valor do ponto, sua descrição, um botão para adicionar esse ponto à lista de Favoritos, e um botão exibir seu gráfico de tempo real. A Figura 6 mostra a sequência de telas que surge ao navegar pela pesquisa de equipamentos.



FIGURA 6 – Navegação pela pesquisa de equipamentos.

### 3.3.4 Navegador

A função chamada de Navegador realiza percursos pela árvore hierárquica de nós gerada e organizada pelo servidor. Esse é o método de busca dos aplicativos genéricos OPC UA. A tela do Navegador começa exibindo o nó raiz, ao clicar nele surge uma lista contendo seus nós filhos, clicando em algum desses nós, surge uma lista contendo seus nós filhos, e assim sucede a navegação recursivamente. A navegação inversa, por sua vez, é realizada pressionando o botão voltar sucessivamente. Como há diferentes classes de nós, a lógica da navegação implementada foi: se o nó selecionado for um nó do tipo *Object* (nós organizadores do espaço de endereçamento), exibe-se uma lista contendo os outros nós agrupados por esse nó *Object*; se o nó selecionado for do tipo *Variable* (nós que armazenam valores), exibe-se uma tela contendo o nome do ponto, seu valor, sua descrição, um botão para adicionar à lista de Favoritos, um botão para exibir seu gráfico de tempo real, e, se for um ponto digital com função de comando habilitada, um botão para envio de comando de controle supervisor para usuários autorizados. A Figura 7 mostra uma sequência de telas exibidas em um processo de navegação.

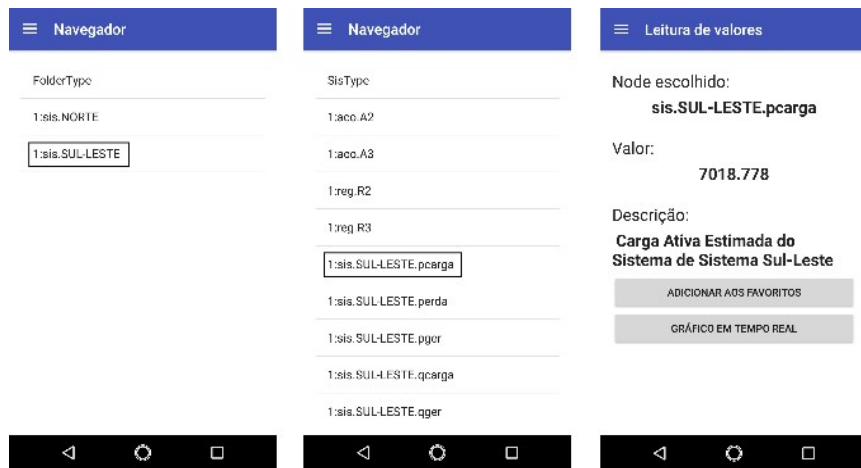


FIGURA 7 – Exemplo de navegação pela árvore.

### 3.3.5 Leitor de código QR

Um modo alternativo desenvolvido para leitura de atributos e valores de dado equipamento é através de um leitor de código QR. O modo de navegação com auxílio do leitor de código QR é semelhante aos modos de navegação mostrados anteriormente, com a diferença que o leitor localizará diretamente o ponto no sistema. A partir daí surge uma lista com os atributos do equipamento e um botão de controle supervisor se for um ponto habilitado para tal. Caso o usuário selecione algum dos atributos, surge uma tela contendo o nome do atributo, seu valor, sua descrição, e um botão para adicioná-lo à lista de Favoritos. A Figura 8 mostra uma sequência de telas exibidas em uma navegação usando o leitor de código QR.

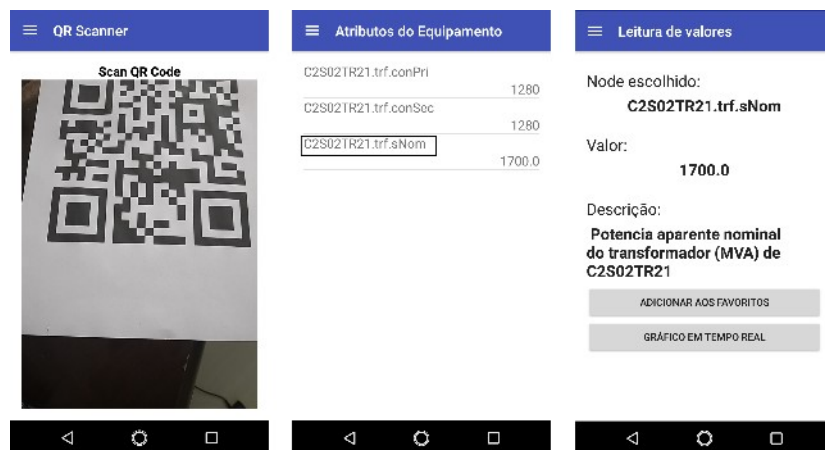


FIGURA 8 – Exemplo de navegação por código QR.

### 3.3.6 Mapa Geral

Por meio da tela de edição de Favoritos é possível inserir, alterar ou remover dados de coordenadas geográficas. Esses dados são usados pelas telas de mapas do aplicativo, como é o caso da tela Mapa Geral. A função mapa geral acessa a lista de favoritos e verifica quais dos favoritos têm coordenadas geográficas inseridas. A partir daí o programa localiza os pontos favoritos com coordenadas no mapa. Para gerar o mapa e utilizar funções para manipular informações no mapa, utilizou-se a API do Google Maps [4]. Todos os pontos que são marcados no mapa ao serem clicados exibem uma janela contendo o identificador do ponto e seu valor, que é atualizado a cada segundo. Caso o ponto selecionado seja um ponto digital de atributo de estado, exibe-se além da janela de informações botões de envio de comando de controle supervisor para usuários cadastrados. Além disso, diferentes cores foram usadas como recurso de auxílio visual de informações. Esse recurso foi utilizado para pontos do tipo Pontos Analógicos ou Pontos Digitais. Os pontos analógicos têm atributos de valores de limites inferiores e superiores, além do atributo de valor. Assim, para auxiliar o usuário na detecção de pontos analógicos cujo valor esteja violando seu limite inferior de advertência ou seu limite superior de advertência, o marcador de pontos analógicos fica vermelho, caso o valor do ponto esteja violando algum limite, ou fica verde, caso o ponto esteja dentro dos limites. Para os pontos digitais, a cor muda conforme o estado de seu atributo. Caso o atributo do ponto digital esteja indicando estado aberto, o seu marcador no mapa fica amarelo, caso o ponto digital esteja fechado, o seu marcador fica azul. A Figura 9-(a) mostra duas telas do mapa geral, a primeira mostrando a janela de informações e botões de controle supervisor para os pontos digitais, e a outra mostrando os diferentes marcadores coloridos no mapa, usados para indicação visual do estado dos pontos analógicos e digitais para o usuário.



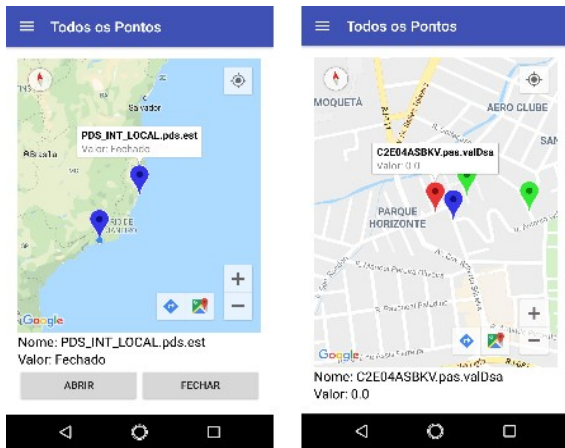


FIGURA 9-(a) – Mapa Geral.

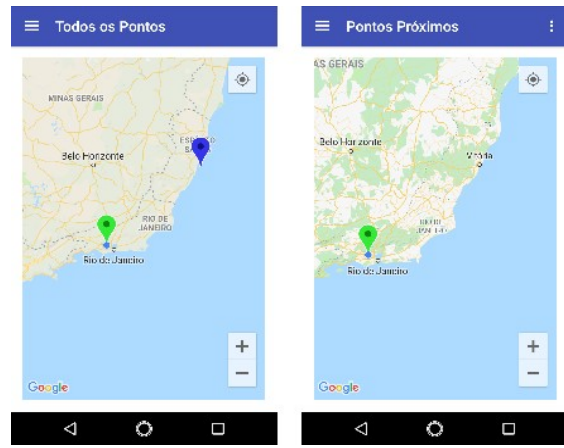


FIGURA 9-(b) – Mapa Geral x Mapa Proximidades.

### 3.3.7 Mapa Proximidades

A tela do mapa de proximidades segue os mesmos princípios da tela do mapa geral. O que diferencia a função Mapa Proximidades da função Mapa Geral é que na tela mapa proximidades só são exibidos os pontos favoritos que estejam fisicamente próximos ao usuário, dentro de um raio de alcance centrado na posição geográfica do usuário. O valor de raio de varredura inicial é de 50 metros, porém esse valor pode ser configurado utilizando a opção Raio de Varredura no menu da tela localizado no canto superior direito. A função usa uma implementação da Fórmula de Haversine [5] para calcular a distância entre o usuário e cada ponto Favorito cujas coordenadas foram inseridas. Se a distância calculada for menor que o raio de varredura, então o marcador é inserido no mapa. Caso o usuário se distancie de algum marcador que estava no mapa por uma distância maior que o raio de varredura, o marcador é removido do mapa. A verificação de pontos próximos é feita a cada três segundos e a atualização dos valores dos marcadores que estão no mapa é feita a cada um segundo. A função da tela Mapa Proximidades é útil principalmente ao usuário de campo que precisa localizar determinado equipamento em suas proximidades. A Figura 9-(b) mostra a diferença de resultados de uma tela de Mapa Geral para uma tela de Mapa Proximidades.

## 4.0 - EXEMPLOS DE APLICAÇÃO

O uso do aplicativo de celular do SAGE pode ajudar a agilizar o trabalho em diversas áreas do setor elétrico. O acesso ao SAGE de forma portátil e remota pode ser de grande utilidade tanto para gestores preocupados com informações gerenciais do sistema elétrico, quanto para equipes de campo, sejam equipes de operação de redes de distribuição ou equipes de manutenção de instalações.

### 4.1 Informações gerenciais

Com o uso do aplicativo, os profissionais responsáveis pelo gerenciamento de instalações, etc, não ficam mais limitados às estações de trabalho para ter acesso aos seus dados de interesse. Por exemplo, o responsável por uma instalação de eletricidade que tenha o sistema SAGE pode acessar a qualquer hora e lugar, que tenha acesso à internet, as informações de tensão elétrica no barramento ou corrente nos alimentadores de carga, etc. Assim, no caso da ocorrência de uma emergência nessa instalação em um momento no qual o profissional responsável pelo gerenciamento dessa instalação está longe da estação de trabalho, ele pode utilizar o aplicativo para acessar os dados do SAGE e, de onde ele estiver, avaliar de forma preliminar as informações elétricas dessa instalação. Isso representa, além de mais facilidade de acesso à informação, mais agilidade na solução de problemas

### 4.2 Manutenção de instalações

O aplicativo móvel do SAGE também apresenta ferramentas úteis na manutenção de instalações que sejam supervisionadas pelo SAGE. No caso de usinas geradoras, ou subestações, por exemplo, os responsáveis pela instalação podem adotar um sistema de identificação dos principais equipamentos por meio de códigos QR que fiquem visíveis e acessíveis aos profissionais responsáveis pelas inspeções e manutenções. Dessa forma, esses profissionais poderiam usar o aplicativo móvel do SAGE para escanear os códigos QR dos equipamentos sob inspeção, e poderiam ler as informações elétricas em tempo real desses equipamentos, avaliando se as grandezas estão conforme o esperado. Isso geraria manutenções mais rápidas e mais eficientes, pois os dados obtidos pelo aplicativo ajudariam a fazer uma análise preliminar de quais equipamentos necessitam de intervenção. Além disso, caso alguns desses equipamentos sejam submetidos a reparos, seria possível verificar, logo após a reenergização do equipamento, se ele está operando bem após seu reparo.



### 4.3 Operação de redes de distribuição

O aplicativo possui funções muito úteis para equipes de campo de operação de redes de distribuição, dentre as quais as funções de georreferenciamento com o auxílio de mapas e a função de envio de comandos de controle supervisão se destacam. As telas do mapa de proximidades permitem um acompanhamento “*in-loco*” da rede de distribuição pelas equipes de campo que atuam na operação de redes de distribuição que utilizam o SAGE, como é o caso de algumas concessionárias que utilizam o módulo SCADA do SAGE em suas redes de distribuição. Unindo funções de georreferenciamento com o auxílio de mapas e a função de envio de comandos de controle supervisão é possível ao profissional de campo localizar dentro de um raio de varredura da sua posição as chaves com acionamento remoto que ele precisa manobrar, por exemplo, na situação de transferência de carga entre circuitos por meio de chaves de recurso. Enquanto as equipes estiverem transitando pelas ruas do bairro da cidade, elas podem visualizar automaticamente informações dos religadores supervisionados e controlados pelo centro de operação próximos a eles. Além disso, as equipes podem verificar o estado dos equipamentos e se as medições de tensão ou corrente estão dentro dos limites.

### 5.0 - CONSIDERAÇÕES FINAIS

Esse aplicativo é o primeiro passo no sentido de prover ao SAGE a qualidade de ser um sistema móvel. Essa ferramenta poderá ser muito útil para equipes de campo, que poderão realizar seus trabalhos e acompanhar os resultados pelo próprio aplicativo. Espera-se que futuramente o aplicativo esteja disponível em outras plataformas de dispositivos móveis, e que novas funções importantes sejam implementadas, como a questão de trabalhar com alarmes e eventos, que são ferramentas bastante úteis para os usuários SAGE.

### 6.0 - REFERÊNCIAS BIBLIOGRÁFICAS

- (1) MAGRANI, E. A Internet das Coisas. 1 ed. Rio de Janeiro, Brasil, FGV Editora, 2018.
- (2) CEPEL. “SAGE – Sistema Aberto de Gerenciamento de Energia”. Disponível em: [http://www.cepel.br/pt\\_br/produtos/sage-sistema-aberto-de-gerenciamento-de-energia.htm](http://www.cepel.br/pt_br/produtos/sage-sistema-aberto-de-gerenciamento-de-energia.htm). Acesso em: 17/12/2018.
- (3) MAHNKE, W., LEITNER, S.-H., DAMM, M. OPC Unified Architecture. 1 ed. Germany, Springer-Verlag Berlin Heidelberg, 2009.
- (4) DEVELOPERS, A. “Maps SDK for Android”. Disponível em: <https://developers.google.com/maps/documentation/android-sdk/intro>. Acesso em: 02/01/2019.
- (5) VENESS, C. “Calculate distance, bearing and more between Latitude/Longitude points”. Disponível em: <https://www.movable-type.co.uk/scripts/latlong.html>. Acesso em: 09/07/2018.

### 7.0 - DADOS BIOGRÁFICOS



Rafael Luiz Rezende Bastos graduou-se em Engenharia Elétrica pela Universidade Federal do Rio de Janeiro (UFRJ) em 2019. Atuou como estagiário do Departamento de Automação de Sistemas do Centro de Pesquisas de Energia Elétrica – CEPEL de 2016 a 2018, onde começou o estudo contido neste artigo. Suas áreas de interesse incluem algoritmos de análise de redes e sistemas de SCADA.



Carmen Lucia Tancredo Borges possui graduação em Engenharia Elétrica pela UERJ (1984), mestrado em Engenharia Elétrica pela COPPE/UFRJ (1991) e doutorado em Engenharia de Sistemas e Computação pela COPPE/UFRJ (1998). É Professor de Engenharia Elétrica da UFRJ desde 1996, atuando na pós-graduação da COPPE e na graduação da Poli. Tem experiência de ensino, pesquisa e desenvolvimento de projetos na área de Sistemas de Potência, atuando principalmente em: Análise, Simulação e Otimização, Confiabilidade, Geração Distribuída, Fontes Alternativas, Sistemas Inteligentes e Processamento de Alto Desempenho. Foi Presidente do Capítulo PES-Power & Energy Systems da Seção Rio do IEEE de 2012 a 2014.



Felipe Vieira Duval, possui graduação em Ciência da Computação pela UFF (2008), com Mestrado em Biologia Computacional e Sistema com tese nas áreas Mineração de Dados e Farmacovigilância (2016). Atua há 14 anos como Analista de Sistemas, sendo 4 no Centro de Pesquisas de Energia Elétrica – CEPEL onde está atualmente. Sua experiência inclui administração de banco de dados, desenvolvimento de softwares em diversas linguagens e Automatização de processos.



Marcelo Rosado da Costa graduou-se em Engenharia Elétrica pela UFRJ (1993), com mestrado em Engenharia Elétrica pela COPPE/UFRJ (1998). É pesquisador do Departamento de Automação de Sistemas do Centro de Pesquisas de Energia Elétrica – CEPEL - desde 1996, atuando no desenvolvimento de aplicações de análise de redes do SAGE, desenvolvido pelo CEPEL. Suas áreas de interesse para pesquisa incluem algoritmos de análise de redes, sistemas de supervisão e controle, simulador para treinamento de operadores e sistema de gerenciamento de energia.