

Estudo de Usabilidade de Assinaturas Digitais em Objetos Metrológicos

Thaís Bardini Idalino, Martin Vigil, Gabriel Estevam de Oliveira, Paola de Oliveira Abel, Erik Gabriel Na, Carlos Eduardo Vitorino Gomes, Victor Henrique Hardt, Marcos Tomaszewsk, Gabriel Bristot Loli

Relatório Técnico INE 004/2022





Universidade Federal de Santa Catarina
Laboratório de Segurança em Computação

Estudo de Usabilidade de Assinaturas Digitais em Objetos Metrológicos

Relatório Final

Versão 1.0

Florianópolis
2022

Sumário

1 Introdução	4
1.1 Objetivos	4
1.2 Documentos complementares	4
2 Arquitetura do Sistema	5
2.1 Requisitos do Sistema	5
2.2 Modelagem dos Componentes	6
2.3 Casos de Uso	9
2.3.1 Verificar Abastecimento	10
2.3.2 Realizar Manifestação	12
2.3.3 Visualizar leitura anterior - Consumidor	13
2.3.4 Enviar dados para o INMETRO	13
2.3.5 Obter Dados da Bomba de Combustível	13
2.3.6 Visualizar leitura anterior - Fiscal	15
2.3.7 Enviar Leitura de Dados para o Servidor	15
2.3.8 Consultar fiscalização	16
2.3.9 Consultar Certificados	17
2.3.10 Consultar Estabelecimentos	17
2.3.11 Consultar Bombas de Combustíveis	18
2.3.12 Consultar Abastecimentos	18
2.3.13 Consultar Manifestação	19
2.4 Diagramas de Sequência	19
2.4.1 Aplicativo Consumidor	20
2.4.2 Aplicativo Fiscal	33
3 Modelagem do Banco de Dados	48
3.1 Tabelas	48
3.1.1 Tabela de Estabelecimentos	48
3.1.2 Tabela de Bombas Medidoras de Combustível (BMCs)	49
3.1.3 Tabela de Dispositivos Indicadores	50
3.1.4 Tabela de Dispositivos Transdutores	50
3.1.5 Tabela de Registros Metrológicos	51
3.1.6 Tabela de Pacotes de Dados dos Abastecimentos	52
3.1.7 Tabela de Certificados Digitais	53
3.1.8 Tabela de Fiscais	54
3.1.9 Tabela de Eventos de Fiscalização de BMCs	55
3.1.10 Tabela de Manifestações	55
3.1.11 Tabela de Registros de Interrupção	56
3.1.12 Tabela de Carga de Software	57
3.1.13 Tabela de Registro de Evento de Manutenção	57
3.1.14 Tabela de Alteração de Parâmetros	58
3.1.15 Tabela de Ensaio de Bicos	59
3.1.16 Tabela de Verificação de Dispositivos Indicadores	59

3.1.17 Tabela de Verificação de Dispositivos Transdutores	60
3.1.18 Tabela de Modelos de Bombas Medidoras de Combustíveis	61
3.1.19 Tabela de Tabelas de Referência de Software	62
3.1.20 Tabela de Hash de Intervalo de Memória	62
3.2 Diagrama do Banco de Dados	63
4 Tecnologias Utilizadas	65
5 Aspectos de Segurança	66
5.1 Aplicativos móveis	66
5.1.1 Medidas adotadas	66
5.1.2 Possíveis vulnerabilidades	66
5.2 Servidor	66
5.2.1 Medidas adotadas	67
5.2.2 Possíveis vulnerabilidades	67
6 Considerações Finais	69

1 Introdução

O sistema Medida Inteligente - INMETRO foi desenvolvido pelo Laboratório de Segurança em Computação (LabSEC) da Universidade Federal de Santa Catarina para promover a comunicação com as novas bombas medidoras de combustíveis líquidos. O sistema é composto por dois aplicativos mobile e um servidor. Um dos aplicativos mobile é destinado a usuários consumidores finais e o outro a usuários fiscais. Enquanto o servidor é responsável por armazenar os dados que são enviados por ambos os aplicativos.

1.1 Objetivos

Este documento tem como objetivo:

- a) Apresentar a arquitetura do software, requisitos funcionais e não-funcionais.
- b) Apresentar os casos de uso e diagramas de sequência.
- c) Apresentar a modelagem do banco de dados.
- d) Apresentar especificações técnicas do software.
- e) Apresentar aspectos relevantes sobre segurança do software.

1.2 Documentos complementares

Além desse relatório, existem outros documentos complementares com instruções para execução e utilização do sistema. Os documentos são listados a seguir:

- a) Descrição da API
- b) Manual do Aplicativo do Consumidor
- c) Manual do Aplicativo do Fiscal

2 Arquitetura do Sistema

Esta seção apresenta uma visão geral da arquitetura do sistema, incluindo os requisitos do sistema, a modelagem dos componentes, os casos de uso e os diagramas de sequência.

2.1 Requisitos do Sistema

O sistema desenvolvido propõe-se a atender os requisitos levantados pela equipe a partir do problema apresentado pelo INMETRO. Nesta seção os requisitos serão apresentados em alto-nível, podendo representar mais de uma funcionalidade do sistema.

O sistema possui os seguintes requisitos funcionais:

- **RF1:** O sistema deve permitir ao consumidor consultar dados referentes a abastecimentos realizados em bombas de combustíveis.
- **RF2:** O sistema deve permitir ao consumidor reportar reclamações e/ou denúncias referentes a abastecimentos realizados.
- **RF3:** O sistema deve permitir ao fiscal consultar dados referentes a parâmetros metrológicos das bombas de combustíveis e dispositivo associados.
- **RF4:** O sistema deve permitir inserir dados de identificação de dispositivos e tabelas de referência de software.
- **RF5:** O sistema deve permitir consultar certificados, estabelecimentos, objetos metrológicos, registros metrológicos e reclamações/denúncias.

Além desses, o sistema possui os seguintes requisitos não-funcionais:

- **RNF1:** O aplicativo móvel deve ser capaz de obter dados das bombas de combustíveis através de comunicação bluetooth.
- **RNF2:** O aplicativo móvel deve ser capaz de verificar as assinaturas digitais feitas pelas bombas de combustíveis.
- **RNF3:** O sistema deve ser capaz de manter uma base de dados atualizada com os certificados OM-BR válidos das bombas de combustíveis.
- **RNF4:** O sistema deve ser capaz de armazenar informações dos objetos metrológicos.

- **RNF5:** O sistema deve ser capaz de armazenar registros metrológicos.

Por fim, elencou-se os seguintes requisitos desejáveis:

- **RD1:** É desejável que os aplicativos realizem a verificação de assinatura digital e certificação digital de forma offline.
- **RD2:** É desejável que o aplicativo do fiscal realize a verificação de integridade de software de forma offline.
- **RD3:** É desejável que os aplicativos armazenem localmente os dados obtidos offline para serem posteriormente enviados ao servidor quando obtiverem conexão com Internet.

2.2 Modelagem dos Componentes

O sistema é composto por três componentes: um servidor e dois aplicativos móveis. Os componentes são partes independentes que atuam de forma integrada para cumprir os requisitos do sistema. Além dos componentes internos do sistema, existem três componentes externos que interagem com o sistema: a bomba de combustível, o verificador de conformidades e o SGI-INMETRO. A bomba de combustível é o componente que gera os dados que alimentam o sistema. O verificador de conformidade é responsável por fazer a verificação das assinaturas digitais. Enquanto o SGI-INMETRO fornece o sistema de autenticação para os usuários fiscais.

Os aplicativos móveis permitem conectar-se à bombas de combustíveis e obter dados dos objetos metrológicos e de abastecimentos. Através dos aplicativos móveis também é possível enviar os dados para serem registrados no servidor. O servidor, por sua vez, mantém uma base de dados para os certificados digitais, as informações dos objetos metrológicos e os registros metrológicos. Por fim, estes dados podem ser consultados através de uma API que é disponibilizada pelo servidor.

A Figura 1 mostra o diagrama de componentes do sistema. Os componentes em cinza claro são externos ao sistema e estão ilustrados apenas para indicação da comunicação. Os componentes em branco são internos do sistema. Os componentes são detalhados a seguir:

Bombas de combustíveis: fornecem as informações sobre os objetos metrológicos e abastecimentos. Cada dispositivo transdutor da bomba de combustível possui um Certificado OM-BR e emite pacotes de dados assinados digitalmente.

Servidor: armazena a base de dados e fornece um serviço de consulta e registros de dados.

Aplicativos Móveis (Consumidor e Fiscal): fornecem uma interface com o usuário consumidor e o com o usuário fiscal. Obtêm dados das bombas de combustíveis e comunicam-se com o servidor, consultando e registrando dados.

Verificador de conformidades: responsável pela verificação das assinaturas e certificados digitais.

SGI-INMETRO: fornece o sistema de autenticação para os usuários fiscais.

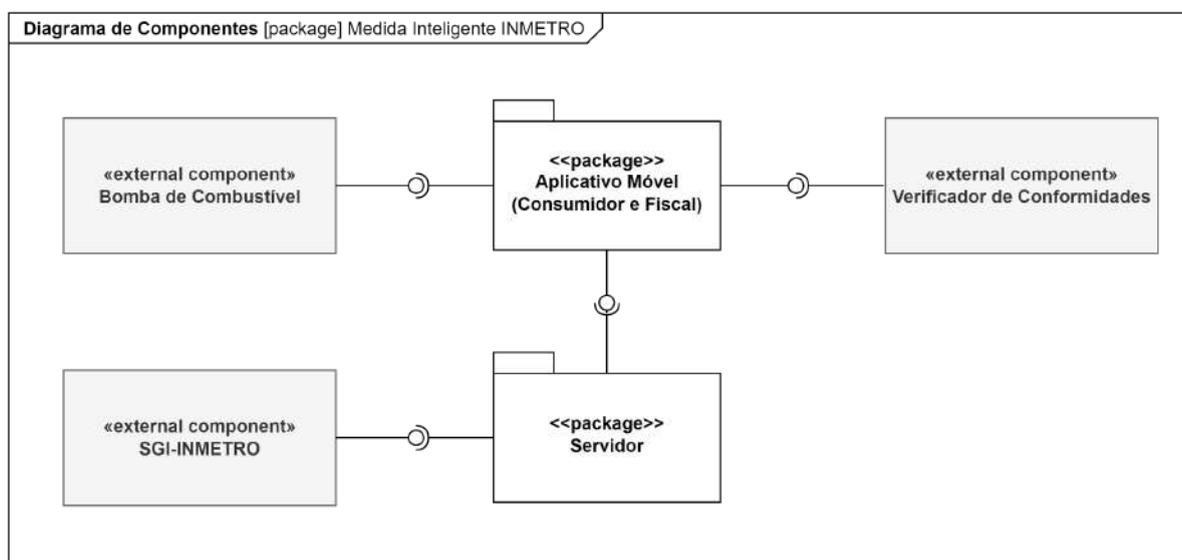


Figura 1: Diagrama de componentes do sistema.

A Figura 2 ilustra os componentes internos do pacote servidor. Os componentes são detalhados a seguir:

Banco de Certificados OM-BR: armazena os Certificados OM-BR das bombas de combustíveis fornecidos pela Autoridade Certificadora.

Banco de Objetos Metrológicos: armazena os dados dos postos de combustíveis e vinculação das bombas de combustíveis.

Banco de Registros Metrológicos: armazena dados de abastecimentos, reclamações e/ou denúncias.

API Externa: fornece uma interface para consulta e registro de dados às bases de dados pela Aplicação Web e Aplicativo Móvel.

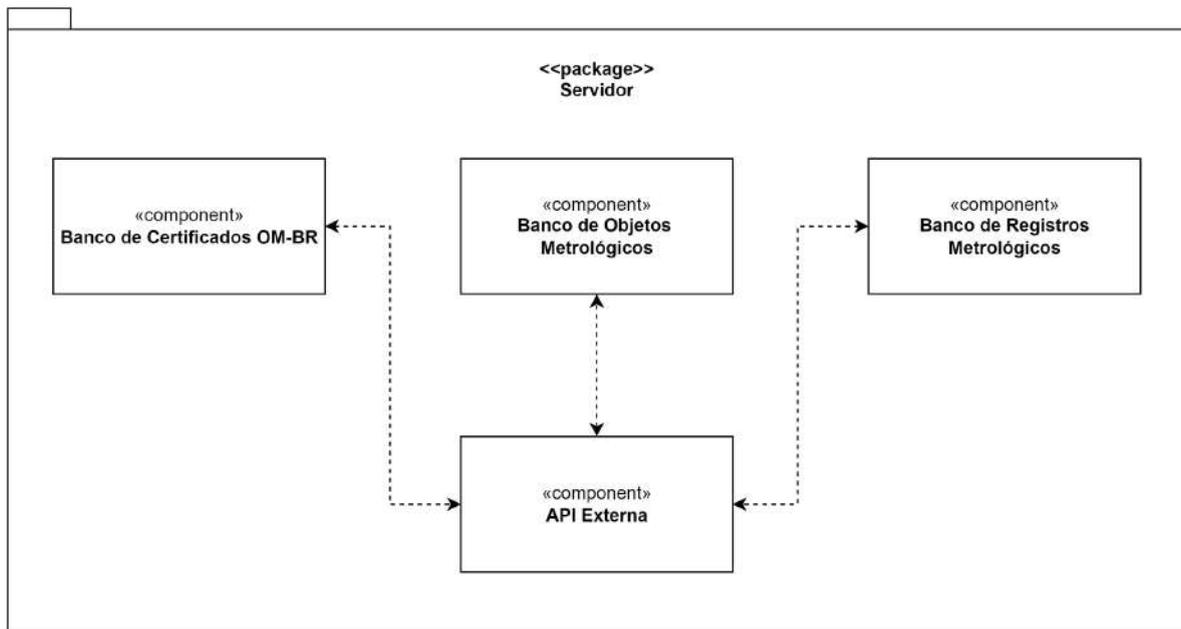


Figura 2: Detalhamento do pacote servidor.

A Figura 3 ilustra os componentes internos do pacote Aplicativo Móvel. Os componentes são detalhados a seguir:

Módulo Bluetooth: solicita dados para a bomba de combustível e decodifica os dados.

Módulo de Assinatura Digital: verifica a assinatura digital emitida pela bomba de combustível referente aos abastecimentos. A verificação pode ser encaminhada para o verificador de conformidades ou realizada localmente, caso não haja conexão com a Internet.

Módulo de Consulta à API: busca dados dos objetos metroológicos no servidor.

Módulo de Reclamações e/ou Denúncias: envia dados ao servidor para serem registrados na base de registros metroológicos.

Módulo de Banco de Dados: responsável por armazenar os dados das leituras localmente.

Interface com Usuário: obtém e fornece dados ao usuário, interconecta os outros componentes.

São previstos dois tipos de usuários que utilizarão o aplicativo móvel: o usuário consumidor e o usuário fiscal. Para cada um deles existe um aplicativo dedicado, ambos com os mesmos componentes, apenas com funcionalidades diferentes que serão vistas mais a frente.

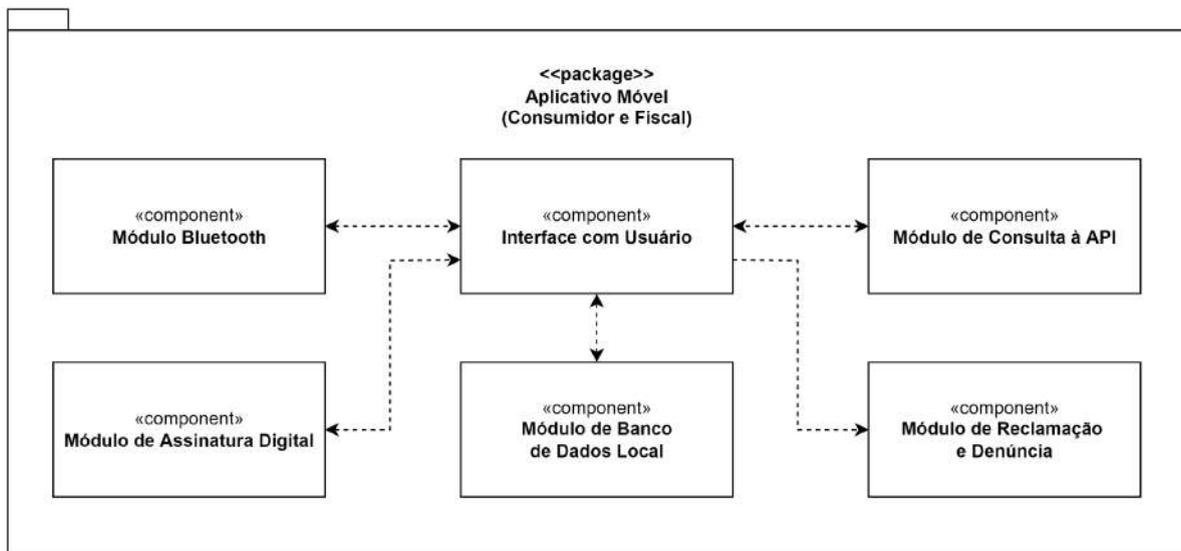


Figura 3: Detalhamento do pacote Aplicativo Móvel.

2.3 Casos de Uso

Nesta seção são descritos os casos de uso do sistema. Os casos de uso descrevem comportamentos do sistema e estão relacionados aos requisitos funcionais. O sistema possui quatro tipos de atores: Consumidor, Fiscal e Observador. Os atores Consumidor e Fiscal irão acessar o sistema através de aplicativos mobile, um aplicativo para cada um dos dois atores. Enquanto o ator Observador poderá consultar a base de dados através da API. A Figura 4 atribui casos de usos a cada um dos atores. Os casos de usos são detalhados nas seções seguintes.

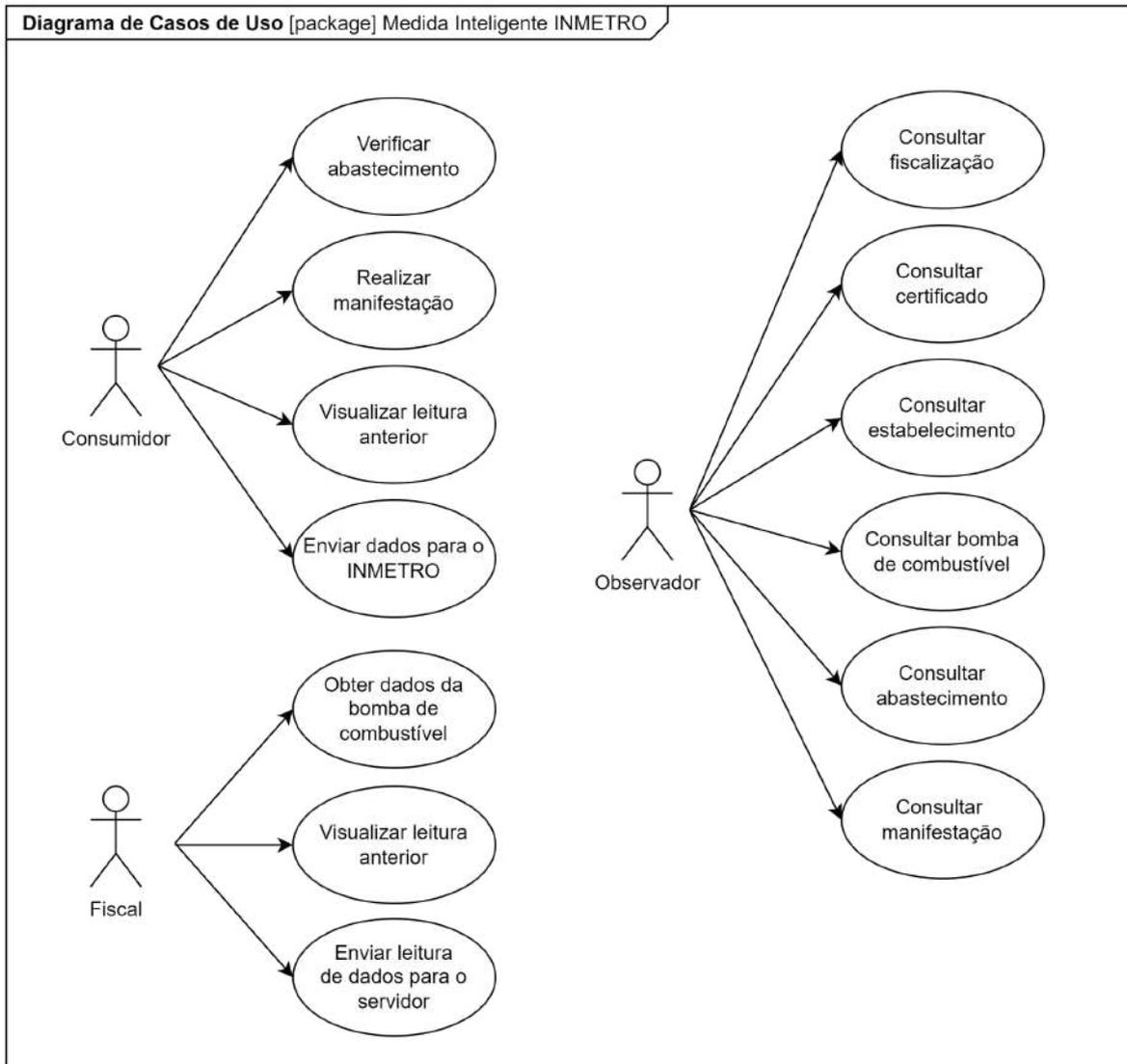


Figura 4 - Atores e Casos de Usos.

2.3.1 Verificar Abastecimento

Caso de Uso: Verificar abastecimento.

Ator: Consumidor.

Cenário Principal: O consumidor chega no estabelecimento e realiza um abastecimento. Após o abastecimento ser concluído, inicia o aplicativo móvel e solicita realizar nova leitura. O aplicativo então realiza o escaneamento bluetooth e mostra as bombas de combustíveis que estão na área de alcance. O consumidor seleciona a bomba que foi utilizada para realizar o abastecimento. Após o consumidor informar a bomba de combustível, ocorre o pareamento entre o dispositivo e a bomba de combustível. Após o pareamento, o aplicativo recebe alguns dados gerais da bomba de combustível e solicita que o usuário informe o

dispositivo indicador utilizado no abastecimento. Após a escolha do dispositivo indicador, o aplicativo recebe os dados associados ao último abastecimento do dispositivo indicador. Além disso, o aplicativo consulta na API do servidor o estabelecimento associado à bomba de combustível. Após obter todos os dados, o aplicativo envia os dados para o servidor e mostra na tela os dados do estabelecimento (nome e endereço) e dados referentes ao abastecimento (nº da bomba, data, hora, volume abastecido, preço/litro e total a pagar). Após isso, o aplicativo solicita para a API do Verificador de Conformidades¹ a verificação das assinaturas digitais dos pacotes de dados de medição. O aplicativo então mostra a opção de obter mais informações, caso o consumidor escolha essa opção são informados a identificação do fabricante e modelo da bomba de combustível, o número de dispositivos transdutores e o resultado da verificação das assinaturas digitais. Por fim, o aplicativo mostra as opções de realizar nova leitura e relatar problemas. Ao escolher a opção de realizar download, o aplicativo gera um arquivo com os dados do abastecimento e descarrega no dispositivo. Ao escolher a opção nova leitura, o aplicativo retorna à tela inicial. Ao escolher a opção relatar problema, o aplicativo é direcionado a área de manifestação.

Cenário Alternativo 1 - O consumidor inicia o aplicativo sem realizar nenhum abastecimento: o aplicativo mostra os dados do último abastecimento, caso esteja disponível.

Cenário Alternativo 2 - Bluetooth e/ou Localização desligado(a): caso o bluetooth e/ou a localização do dispositivo esteja desligado(a), o aplicativo solicita a ativação para o usuário e aguarda até que o procedimento seja realizado.

Cenário Alternativo 3 - Falha no pareamento: caso não seja possível realizar o pareamento, o aplicativo mostra uma mensagem de falha e sugere a opção de tentar novamente.

Cenário Alternativo 4 - Assinatura digital inválida: caso a assinatura digital esteja inválida, o aplicativo mostra uma mensagem ao consumidor e sugere que ele reporte o erro.

Cenário Alternativo 5 - Sem conexão de rede: caso não seja possível estabelecer conexão de rede, as verificações de assinaturas digitais serão feitas localmente. Nesse caso, não é feita a verificação de certificados revogados. Além disso, não será possível obter os dados do estabelecimento e essa seção ficará oculta. Também não será possível enviar os dados do abastecimento e de manifestação para o servidor. Nesse caso, os dados são armazenados localmente até que seja possível estabelecer conexão de rede. Se em um

¹ Serviço disponibilizado pelo ITI para verificar assinaturas digitais no contexto da ICP-Brasil. Na versão de testes, utilizará-se o servidor de homologação disponibilizado pelo LabSEC.

momento posterior, o consumidor abrir o aplicativo e for possível estabelecer conexão de rede, então os dados pendentes são enviados para o servidor.

Cenário Alternativo 6 - Bomba de combustível não cadastrada no servidor: é possível que a bomba de combustível utilizada ainda não esteja cadastrada no servidor do Sistema Medida Inteligente. Nesse caso, não será possível obter os dados do estabelecimento da bomba de combustível. Por conta disso, a seção de “Dados do estabelecimento” ficará oculta.

Cenário Alternativo 7 - Erro ao enviar dados do abastecimento: caso não seja possível realizar o registro do abastecimento no servidor, o aplicativo mostra uma mensagem de falha e armazena os dados localmente, caso ainda não esteja armazenado. Nesse caso, o registro permanece pendente até a próxima tentativa de envio.

2.3.2 Realizar Manifestação

Caso de Uso: Realizar Manifestação.

Ator: Consumidor.

Cenário Principal: Ao ser redirecionado para a área de manifestação o consumidor informa o tipo de relato (reclamação ou denúncia), seleciona o assunto, o complemento e opcionalmente preenche um campo de texto livre. Logo abaixo, os dados pessoais do consumidor como: nome, CPF e email estarão auto preenchidos, buscados do login GOVBR realizado anteriormente. O consumidor então seleciona a opção “Enviar Manifestação”. O aplicativo é direcionado para outra tela onde é exibido o resumo da manifestação feita.

Cenário Alternativo 1 - Sem conexão de rede: caso não seja possível estabelecer conexão de rede para envio da manifestação, o aplicativo mostra uma mensagem de erro. Nesse caso, os dados são armazenados localmente até que seja possível estabelecer conexão de rede. Se em um momento posterior, o consumidor abrir o aplicativo e for possível estabelecer conexão de rede, então os dados pendentes são enviados para o servidor.

Cenário Alternativo 2 - Erro ao enviar manifestação : caso não seja possível realizar o registro da manifestação no servidor, o aplicativo mostra uma mensagem de falha e armazena os dados localmente, caso ainda não esteja. Nesse caso, o registro permanece pendente até a próxima tentativa de envio.

2.3.3 Visualizar leitura anterior - Consumidor

Caso de Uso: Visualizar leitura anterior.

Ator: Consumidor.

Cenário Principal: Na tela inicial do aplicativo consumidor é possível visualizar uma lista com as últimas leituras realizadas. Ao selecionar uma leitura é possível visualizar novamente os dados daquela leitura. Também será possível, caso ainda não se tenha feito, escolher a opção de relatar problema.

2.3.4 Enviar dados para o INMETRO

Caso de Uso: Enviar dados para o INMETRO.

Ator: Consumidor.

Cenário Principal: Após a realização de pelo menos uma leitura de abastecimento, será exibida na tela inicial do aplicativo o botão “Deseja enviar dados para o INMETRO?”. Esse botão ficará disponível sempre que existir pelo menos um abastecimento gravado no histórico que não tenha sido enviado ao servidor. Ao clicar no botão, o aplicativo mostrará um pop up, contendo mais informações acerca da importância do envio dos dados ao INMETRO. Ao selecionar o botão “Confirmar” no pop up, o aplicativo mostrará uma mensagem confirmando o envio dos dados.

Cenário Alternativo 1 - Caso o usuário não esteja logado no GOVBR: ao clicar no botão “Confirmar” dentro do pop up, o app abrirá o navegador para que o login seja concretizado. Ao final do processo de login, o navegador é fechado automaticamente e o aplicativo continua com o processo de envio dos dados.

2.3.5 Obter Dados da Bomba de Combustível

Caso de Uso: Obter Dados da Bomba de Combustível.

Ator: Fiscal.

Cenário Principal: Na tela inicial do aplicativo fiscal, o usuário irá encontrar a opção realizar nova leitura. Ao selecionar a opção, o aplicativo realizará o escaneamento bluetooth e irá mostrar as bombas de combustíveis que estão na área de alcance. O fiscal seleciona a bomba que deseja obter os dados e então ocorre o pareamento entre o dispositivo e a

bomba de combustível. Após o pareamento, o aplicativo solicita ao usuário o número do Inmetro relacionado ao equipamento fiscalizado. O número do Inmetro pode ser digitado manualmente ou escaneado por meio da câmera. Após isso, o aplicativo mostra as seguintes opções ao fiscal: Visualizar dados da Bomba, Dados do Estabelecimento, Verificação de Integridade de Software, Verificação da Certificação Digital, Abastecimento de ensaio, Registros e Reportar Leitura. A opção Visualizar Dados da Bomba redireciona para outra tela contendo as seguintes informações: Número de Série do Instrumento, Modelo do Instrumento, Número de Dispositivo Indicadores, Número de Dispositivos Transdutores, Identificação do Fabricante e número do Inmetro. A opção de Dados do Estabelecimento mostra a qual estabelecimento a bomba está vinculada, sendo que este dado é consultado na API do servidor. A opção Verificação de Integridade de Software mostra se o procedimento de verificação de integridade de software da bomba foi realizado com sucesso, é possível refazer a verificação com a opção 'Verificar'. A opção Verificação da Certificação Digital redireciona para uma tela com a lista de dispositivos transdutores, que pode ser expandida com um clique, revelando informações detalhadas dos certificados avaliados. Um botão na parte inferior da tela permite repetir a verificação. A opção Abastecimento de Ensaio redireciona para uma tela onde é possível visualizar os abastecimentos já registrados e realizar novas leituras de abastecimento. Ao selecionar o botão "Nova Leitura", o aplicativo voltará a se conectar com a bomba de combustível por meio do bluetooth. O usuário deverá selecionar o lado da bomba de combustível que contém o abastecimento para verificação. Após isso, o aplicativo busca os dados e redireciona para uma nova tela, onde são exibidas as informações básicas do abastecimento. O usuário pode selecionar a opção "Mais Informações", para visualizar detalhes do abastecimento. Há também o botão "Nova Leitura", que redireciona o usuário para a página inicial da seção Abastecimentos de Ensaio. Ao escolher a opção Registros tem-se acesso às alterações de parâmetros, cargas de software, eventos de interrupções e eventos de manutenções. Por fim, há a opção Reportar Leitura. O aplicativo irá pedir que o usuário realize autenticação com CPF, senha e UF. Se os dados informados forem válidos, o aplicativo mostra uma mensagem de registro enviado com sucesso.

Cenário Alternativo 1 - Bluetooth e/ou Localização desligado(a): caso o bluetooth e/ou a localização do dispositivo esteja desligado(a), o aplicativo solicita a ativação para o usuário e aguarda até que o procedimento seja realizado.

Cenário Alternativo 2 - Falha no pareamento: caso não seja possível realizar o pareamento, o aplicativo mostra uma mensagem de falha e sugere a opção de tentar novamente.

Cenário Alternativo 3 - Sem conexão de rede: caso não seja possível estabelecer conexão de rede, as verificações de assinaturas digitais serão feitas localmente. Nesse caso, não é feita a verificação de certificados revogados. Além disso, não será possível obter os dados do estabelecimento e estes aparecerão como indisponíveis. Por fim, não será possível enviar os dados ao servidor. Nesse caso, os dados são armazenados localmente até que seja possível estabelecer conexão de rede.

Cenário Alternativo 4 - Erro ao enviar relatório: caso não seja possível realizar o registro do relatório no servidor, o aplicativo mostra uma mensagem de falha e armazena os dados localmente, caso ainda não esteja armazenado. Nesse caso, o registro permanece pendente até a próxima tentativa de envio.

2.3.6 Visualizar leitura anterior - Fiscal

Caso de Uso: Visualizar leitura anterior.

Ator: Fiscal.

Cenário Principal: Na tela inicial do aplicativo fiscal é possível visualizar uma lista com as últimas fiscalizações realizadas. Cada fiscalização irá conter um status, que indicará 'Enviado' ou 'Não enviado'. Ao selecionar uma leitura é possível visualizar novamente os dados daquela leitura.

2.3.7 Enviar Leitura de Dados para o Servidor

Caso de Uso: Enviar leitura de dados para o servidor.

Ator: Fiscal.

Cenário Principal: Na tela inicial do aplicativo será possível visualizar as leituras com envio pendentes. Ao selecionar uma ou mais leituras, irá aparecer uma opção para enviar leitura. Ao escolher a opção, o aplicativo irá pedir que o usuário realize autenticação com CPF, senha e UF. Se os dados informados forem válidos, o aplicativo mostra uma mensagem de registro enviado com sucesso. É possível também realizar o envio de dados dentro da própria leitura. Basta selecionar a fiscalização desejada e rolar até a parte inferior da tela, onde há o botão "Reportar leitura".

Cenário Alternativo 1 - Sem conexão de rede: caso não seja possível estabelecer conexão de rede, o envio não é realizado e as leituras permanecem como pendentes.

Cenário Alternativo 2 - Erro ao enviar relatório: caso não seja possível realizar o registro do relatório no servidor, o aplicativo mostra uma mensagem de falha e as leituras permanecem como pendentes.

2.3.8 Consultar fiscalização

Caso de Uso: Consultar fiscalização.

Ator: Observador.

Cenário Principal: Existe na API do backend a opção de consultar as fiscalizações registradas, por meio da rota /eventosFiscalizacao. Ao fazer a consulta, é retornada uma lista com todas as fiscalizações armazenadas no sistema, contendo ID da fiscalização no sistema, nome do fiscal que realizou a fiscalização, número serial da bomba em que foi feita a fiscalização, data e hora da fiscalização. É possível retornar dados de apenas uma fiscalização, por meio do ID da fiscalização. Essa consulta retorna data e hora da fiscalização, número serial da bomba em que foi feita a fiscalização, dados do fiscal que realizou a fiscalização (CPF, nome e Unidade Federativa), lista de alterações de parâmetros referentes ao evento de fiscalização (ID da alteração de parâmetros no sistema e descrição do resultado da alteração de parâmetros), lista de cargas de software referentes ao evento de fiscalização (ID da carga de software no sistema e descrição do resultado da carga de software), lista de ensaios de bico referentes ao evento de fiscalização (ID do ensaio de bico no sistema, ID do abastecimento em que foi feito o ensaio, número do INMETRO e descrição do resultado do ensaio de bico), lista de eventos de manutenção referentes ao evento de fiscalização (ID do evento de manutenção e descrição do resultado do evento de manutenção), lista de eventos de interrupção referentes ao evento de fiscalização (ID do evento de interrupção e descrição do resultado do evento de interrupção), lista de verificações de dispositivos indicadores referentes ao evento de fiscalização (ID da verificação, ID do dispositivo indicador verificado e descrição do resultado da verificação) e lista de verificações de dispositivos transdutores referentes ao evento de fiscalização (ID da verificação, ID do dispositivo transdutor verificado e descrição do resultado da verificação). É necessário ter um token válido do SGI do INMETRO para realizar a consulta.

Cenário Alternativo 1 - Sem conexão de rede: caso não seja possível estabelecer conexão de rede, o sistema mostra uma mensagem de falha.

Cenário Alternativo 2 - Token inválido: caso não seja possível validar o token de acesso, a API retorna uma mensagem de erro.

2.3.9 Consultar Certificados

Caso de Uso: Consultar Certificados.

Ator: Observador.

Cenário Principal: A API do backend disponibiliza a opção de consultar os 'Certificados OM-BR' através da rota /certificados. A consulta retorna a lista dos certificados OM-BR armazenados no sistema, contendo serial do certificado, estabelecimento vinculado e serial da BMC, sendo possível filtrar por cada um desses campos. Ao informar o serial de um certificado específico, a consulta retorna os campos CNSubject, publicKey, notBefore, notAfter e o estado de revogação para o certificado correspondente. É necessário ter um token válido do SGI do INMETRO para realizar a consulta.

Cenário Alternativo 1 - Sem conexão de rede: caso não seja possível estabelecer conexão de rede, o sistema mostra uma mensagem de falha.

Cenário Alternativo 2 - Token inválido: caso não seja possível validar o token de acesso, a API retorna uma mensagem de erro.

2.3.10 Consultar Estabelecimentos

Caso de Uso: Consultar Estabelecimentos.

Ator: Observador.

Cenário Principal: Há na API do backend a opção de consultar os estabelecimentos, por meio da rota /estabelecimentos. Ao fazer uma consulta, é retornada uma lista com todos os estabelecimentos armazenados no sistema, contendo ID do estabelecimento no sistema, Unidade Federativa e razão social. É possível retornar informações de apenas um estabelecimento, por meio do ID. Essa consulta retorna o ID no sistema SGI, CNPJ, razão social, endereço, município e Unidade Federativa. É necessário ter um token válido do SGI do INMETRO para realizar a consulta.

Cenário Alternativo 1 - Sem conexão de rede: caso não seja possível estabelecer conexão de rede, o sistema mostra uma mensagem de falha.

Cenário Alternativo 2 - Token inválido: caso não seja possível validar o token de acesso, a API retorna uma mensagem de erro.

2.3.11 Consultar Bombas de Combustíveis

Caso de Uso: Consultar Estabelecimentos.

Ator: Observador.

Cenário Principal: A API do backend disponibiliza a opção de consultar as bombas de combustíveis através da rota /bmcs. A consulta retorna uma lista com todas as bombas de combustíveis cadastradas no sistema, contendo informações como: serial, modelo, fabricante e estabelecimento. Também é possível filtrar a pesquisa utilizando o serial da bomba de combustíveis. Essa consulta retorna o modelo, fabricante, dispositivos indicadores (contendo o seu número na própria bomba de combustíveis e sua própria versão de software), transdutores (contendo o seu número na própria bomba de combustíveis e sua própria versão de software), além de informações do estabelecimento onde a bomba está localizada: CNPJ, razão social. É necessário ter um token válido do SGI do INMETRO para realizar a consulta.

Cenário Alternativo 1 - Sem conexão de rede: caso não seja possível estabelecer conexão de rede, o sistema mostra uma mensagem de falha.

Cenário Alternativo 2 - Token inválido: caso não seja possível validar o token de acesso, a API retorna uma mensagem de erro.

2.3.12 Consultar Abastecimentos

Caso de Uso: Consultar Abastecimentos.

Ator: Observador.

Cenário Principal: Existe na API do backend a opção de consultar todos os abastecimentos armazenados através da rota /abastecimentos. Ao fazer uma consulta, é retornada uma lista com todos os abastecimentos, contendo as informações a seguir: ID do abastecimento no sistema, o estabelecimento onde o abastecimento foi feito, número serial da bomba responsável pelo abastecimento, data e hora do abastecimento. É possível filtrar essa consulta pelo ID do abastecimento no sistema. O retorno contém os seguintes dados do abastecimento selecionado: estabelecimento, número serial da bomba de combustível responsável, booleano indicando se o abastecimento é ou não de ensaio, data, hora e lista com os pacotes de dados do abastecimento (o pacote de dados contém: ID do pacote de dados, data e hora do registro do pacote de dados, assinatura digital, identificador único do dispositivo controlador que registrou o pacote, ID do abastecimento vindo da memória do dispositivo controlador, ID do abastecimento vindo da memória do dispositivo transdutor,

constante de calibração da bomba de combustível, volume de combustível do abastecimento medido pelo dispositivo transdutor, volume de combustível do abastecimento medido pelo dispositivo controlador, valor monetário da transação, identificação serial física única do dispositivo transdutor, preço por litro do combustível e ID do transdutor que registrou o pacote). É necessário ter um token válido do SGI do INMETRO para realizar a consulta.

Cenário Alternativo 1 - Sem conexão de rede: caso não seja possível estabelecer conexão de rede, o sistema mostra uma mensagem de falha.

Cenário Alternativo 2 - Token inválido: caso não seja possível validar o token de acesso, a API retorna uma mensagem de erro.

2.3.13 Consultar Manifestação

Caso de Uso: Consultar Manifestação.

Ator: Observador.

Cenário Principal: A API do backend disponibiliza a opção de consultar manifestações através da rota /manifestacoes. A consulta retorna uma lista com todas as manifestações cadastradas no sistema, contendo: ID da manifestação no sistema, ID do estabelecimento no sistema ao qual se refere a manifestação, número serial da bomba de combustível à qual se refere a manifestação, data e hora da manifestação. A consulta pode ser filtrada utilizando o ID da manifestação. O retorno trará as seguintes informações de somente uma manifestação: ID do abastecimento, data e hora, órgão destinatário da manifestação, tipo da manifestação, assunto, descrição e ID da manifestação na API do FalaBR. É necessário ter um token válido do SGI do INMETRO para realizar a consulta.

Cenário Alternativo 1 - Sem conexão de rede: caso não seja possível estabelecer conexão de rede, o sistema mostra uma mensagem de falha.

Cenário Alternativo 2 - Token inválido: caso não seja possível validar o token de acesso, a API retorna uma mensagem de erro.

2.4 Diagramas de Sequência

Os diagramas de sequência permitem ilustrar as comunicações entre os componentes do sistema. Os diagramas contêm os atores e componentes envolvidos em

cada comunicação e o fluxo de dados entre eles. São descritos também os dados enviados e recebidos em cada requisição.

2.4.1 Aplicativo Consumidor

Ao iniciar o aplicativo consumidor, é exibido um tutorial apresentando algumas instruções para uso. Após finalizar o tutorial, é mostrada a tela inicial do aplicativo contendo uma lista com as últimas leituras realizadas e opção para realizar nova leitura. A sequência dos procedimentos de inicialização do aplicativo é ilustrada na Figura 5. Os passos de exibição do tutorial são realizados apenas na primeira execução do aplicativo, nas próximas execuções é exibida diretamente a tela inicial, com a lista de últimas leituras e as opções.

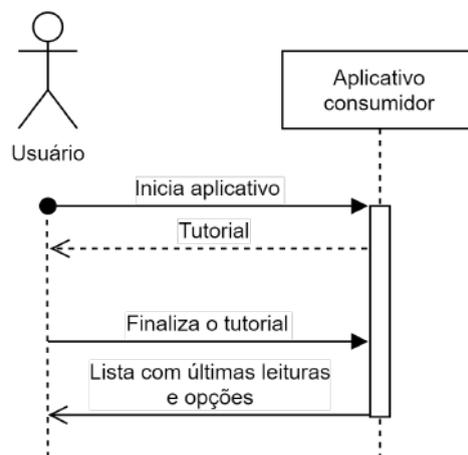


Figura 5: Procedimentos de inicialização do aplicativo.

Ao escolher a opção para realizar nova leitura, o aplicativo automaticamente verifica as permissões do uso do bluetooth e inicia o escaneamento de bombas de combustível próximas. Quando finalizado, o escaneamento retorna para o usuário as bombas de combustível encontradas. A lista de bombas de combustível é exibida na tela do aplicativo e o usuário pode selecionar qual bomba de combustível foi utilizada para realizar o seu abastecimento. Nesse momento, o smartphone do usuário irá conectar-se com a bomba de combustível selecionada através de conexão bluetooth. A sequência dos procedimentos de escaneamento e conexão com a bomba de combustível é ilustrada na Figura 6.

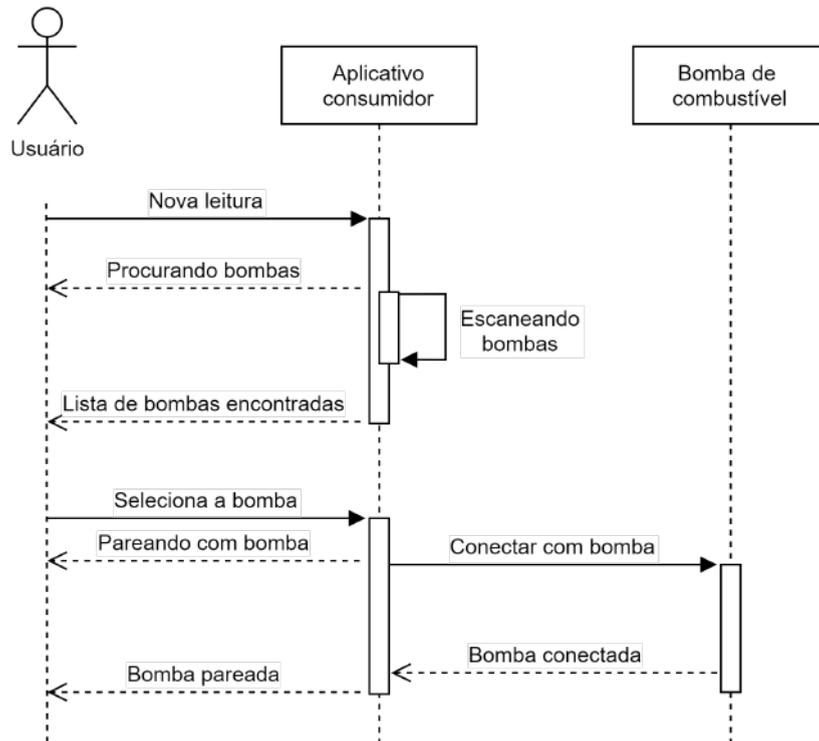


Figura 6: Procedimentos de escaneamento e conexão com a bomba de combustível.

Para cada dispositivo bluetooth encontrado, o aplicativo verifica se este possui as características necessárias para ser uma BMC. Isso é feito por meio de uma comparação entre os serviços e características que o dispositivo bluetooth carrega e uma matriz, armazenada localmente no aplicativo, que contém o serviço e as características de cada fabricante de bomba de combustível. Caso ocorra compatibilidade entre o que está na matriz com o que o dispositivo bluetooth carrega, o aplicativo tem a confirmação que o equipamento é uma bomba de combustível. Desta forma, são listados na tela apenas os dispositivos que correspondem a BMCs.

Após a seleção da bomba de combustível, o aplicativo consulta algumas informações relevantes do equipamento: a identificação do fabricante, o modelo e o número de série. A sequência de procedimentos que ocorrem nesse processo podem ser vistos na Figura 7. As requisições para estas consultas são descritas nas tabelas 1, 2 e 3.

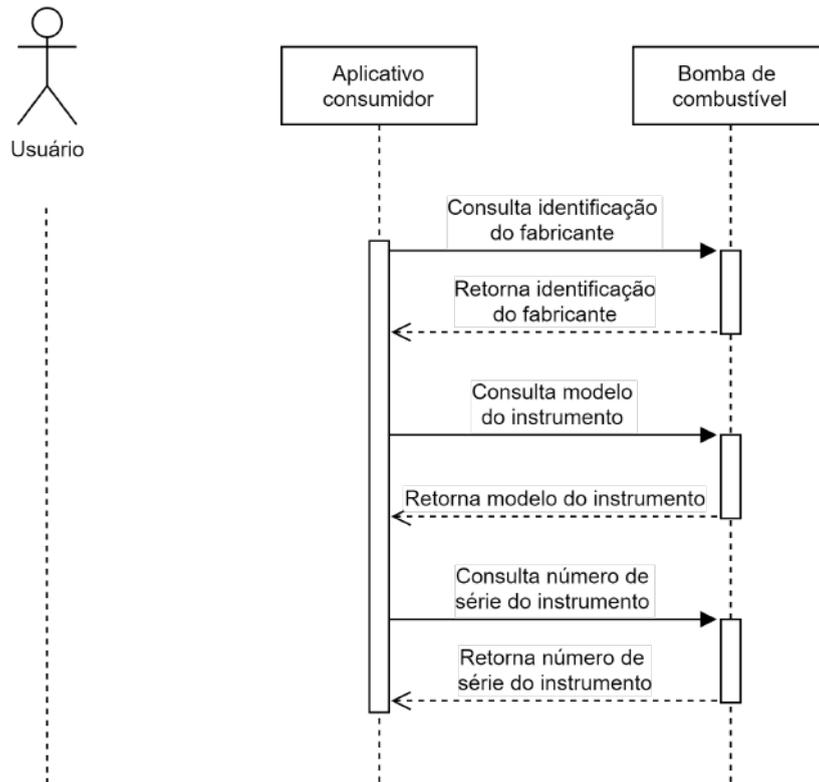


Figura 7: Procedimentos para obtenção dos dados da bomba de combustível.

Tabela 1: Requisição - Identificação do fabricante.

Requisição	Identificação do fabricante.
Para	Bomba de combustível (Comando 03h).
Descrição	Permite consultar a identificação do fabricante do instrumento.
Parâmetros	Sem parâmetros.
Retorno	Identificação do fabricante.

Tabela 2: Requisição - Modelo do instrumento.

Requisição	Modelo do instrumento.
Para	Bomba de combustível (Comando 05h).
Descrição	Permite consultar a identificação do modelo do instrumento.
Parâmetros	Sem parâmetros.
Retorno	Identificação do modelo.

Tabela 3: Requisição - Número de série do instrumento.

Requisição	Número de série do instrumento.
Para	Bomba de combustível (Comando 06h).
Descrição	Permite consultar o número de série do instrumento que identifica unicamente a bomba de combustível.
Parâmetros	Sem parâmetros.
Retorno	Número de série do instrumento.

Logo na sequência, o aplicativo consulta o número de dispositivos (indicadores e transdutores) instalados na bomba de combustível. A sequência dos procedimentos de consulta de dispositivos instalados é ilustrada na Figura 8.

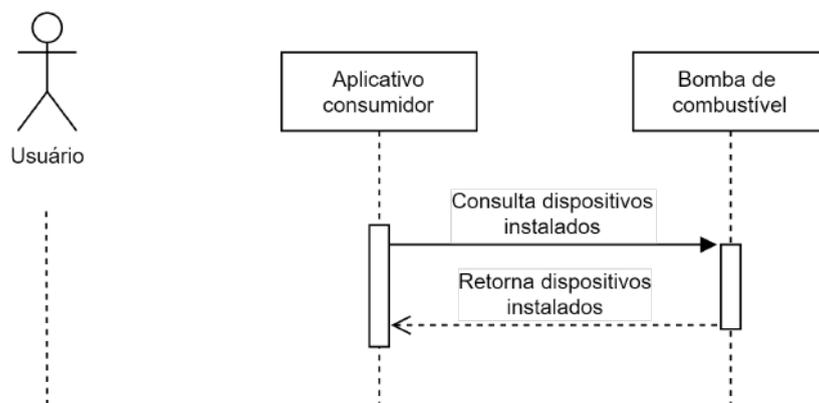


Figura 8: Consulta dos dispositivos instalados

A consulta de dispositivos instalados na bomba de combustível é realizada através do comando 40h da lista de comandos das bombas medidoras de combustíveis. A especificação da requisição é descrita na Tabela 4.

Tabela 4: Requisição - Dispositivos instalados.

Requisição	Dispositivos instalados.
Para	Bomba de combustível (Comando 40h).
Descrição	Permite consultar a quantidade de dispositivos transdutores e indicadores instalados na bomba de combustível.
Parâmetros	Sem parâmetros.
Retorno	Número de dispositivos indicadores e número de dispositivos transdutores.

Após isso, o aplicativo utiliza o número de série requisitado anteriormente para consultar no servidor à qual estabelecimento a bomba de combustível está vinculada. Os dados do estabelecimento são mostrados ao usuário para que este confira a vinculação do instrumento. A sequência dos procedimentos de consulta de número de série e estabelecimento é ilustrada na Figura 9.

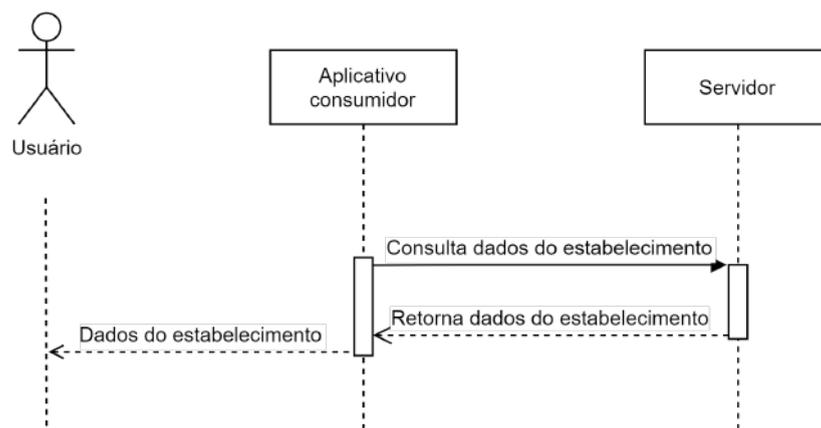


Figura 9: Consulta de dados do estabelecimento.

A consulta dos dados do estabelecimento é realizada através da API do servidor, como descrito na Tabela 5. Caso o smartphone esteja sem conexão com a Internet, a

consulta de estabelecimento não acontece e a seção do aplicativo que exibiria esses dados é ocultada.

Tabela 5: Requisição - Dados do estabelecimento.

Requisição	Dados do estabelecimento.
Para	Servidor.
Descrição	Permite consultar a qual estabelecimento a bomba de combustível está vinculada, caso esteja vinculada.
Parâmetros	Número de série do instrumento.
Retorno	Razão Social, CNPJ e endereço do estabelecimento ou nulo caso não esteja vinculada.

Na sequência, é solicitado ao usuário que informe o dispositivo indicador utilizado no seu abastecimento. A sequência dos procedimentos de solicitação do dispositivo indicador é ilustrada na Figura 10.

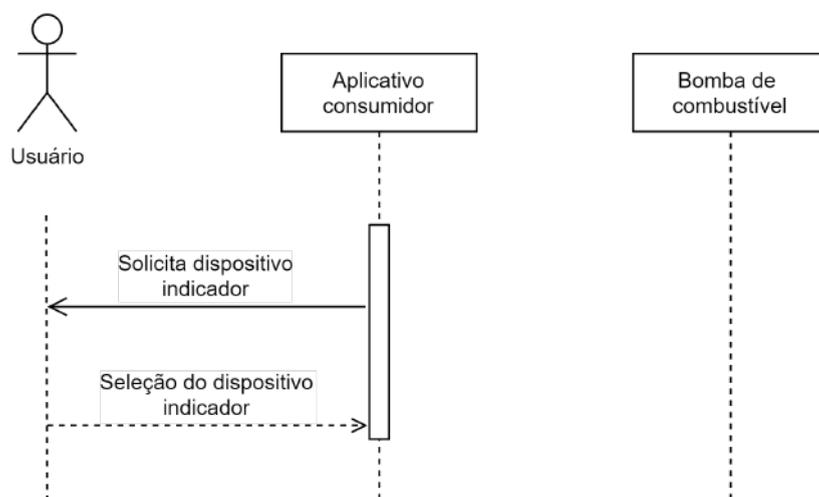


Figura 10: Procedimento para consulta do dispositivo indicador.

Após isso, o aplicativo busca os dados relativos ao último abastecimento realizado por meio do dispositivo indicador selecionado pelo usuário. Esse procedimento é iniciado com a chamada do comando 41h da bomba de combustível, responsável pela consulta dos dispositivos associados ao último abastecimento, como descrito na Tabela 6. Na sequência, para cada dispositivo transdutor associado consulta-se o pacote de dados de medição com

o comando 42h. O quadro envolvendo esse passo na Figura 11 representa o laço de repetição para cada dispositivo transdutor associado. Se o abastecimento tiver apenas um dispositivo transdutor associado, então o laço é iterado apenas uma vez. A requisição de consulta de dados de medição é descrita na Tabela 7.

Tabela 6: Requisição - Dispositivos associados ao abastecimento.

Requisição	Dispositivos associados ao abastecimento.
Para	Bomba de combustível (Comando 41h).
Descrição	Permite consultar os dispositivos transdutores associados ao último abastecimento.
Parâmetros	Dispositivo indicador.
Retorno	Lista de dispositivos transdutores.

Tabela 7: Requisição - Pacote de dados de medição.

Requisição	Pacote de dados de medição.
Para	Bomba de combustível (Comando 42h).
Descrição	Permite consultar as informações referentes ao último abastecimento de um dispositivo transdutor.
Parâmetros	Dispositivo transdutor.
Retorno	Pacote de dados de totalização da medição.

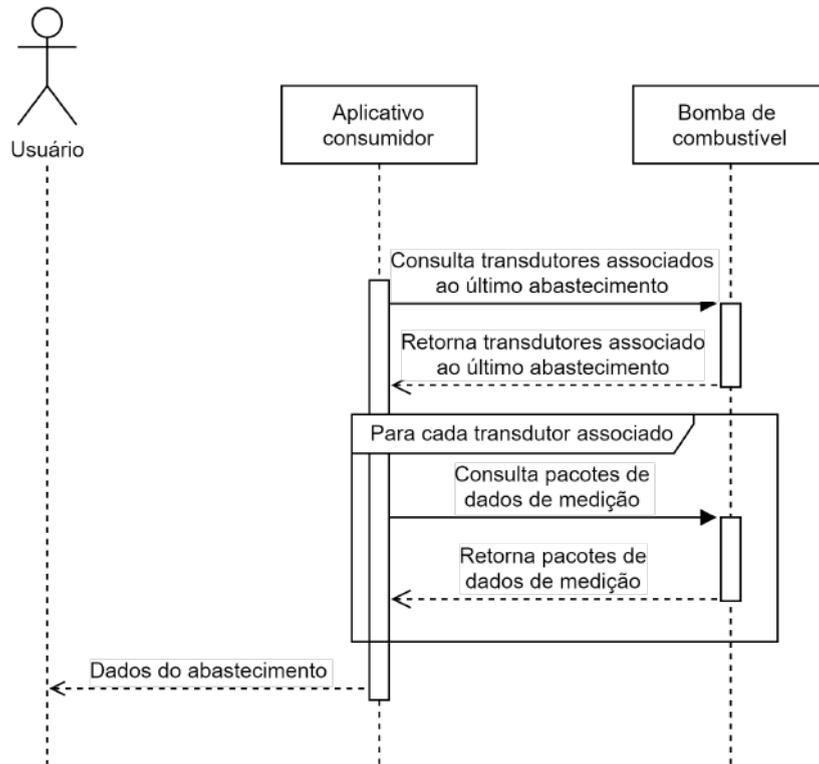


Figura 11: Obtenção dos dados do abastecimento.

Além disso, o aplicativo solicita ao servidor a verificação da assinatura digital e do certificado digital. Para isso, para cada dispositivo transdutor associado consulta-se o certificado digital do transdutor com o comando 43h e assinatura digital do pacote de dados com o comando 44h, como ilustrado na Figura 12. As requisições de consulta de certificado digital e assinatura digital são descritas nas Tabela 8 e 9 respectivamente.

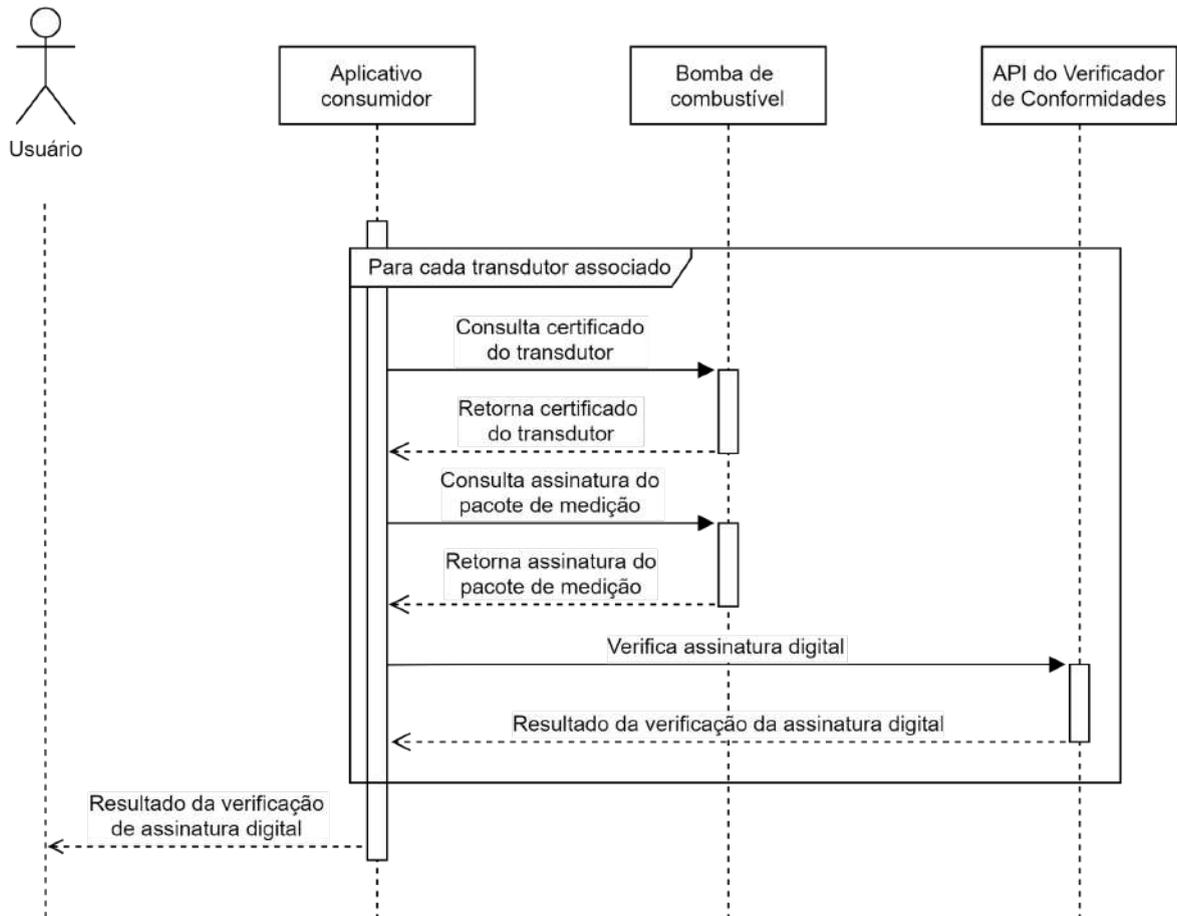


Figura 12: Verificação das assinaturas digitais.

Tabela 8: Requisição - Certificado digital.

Requisição	Certificado digital.
Para	Bomba de combustível (Comando 43h).
Descrição	Permite consultar o certificado digital do dispositivo transdutor.
Parâmetros	Dispositivo transdutor.
Retorno	Certificado digital do dispositivo transdutor.

Tabela 9: Requisição - Assinatura digital.

Requisição	Assinatura digital.
Para	Bomba de combustível (Comando 44h).
Descrição	Permite consultar a assinatura digital do pacote de dados de medição do último abastecimento de um dispositivo transdutor.
Parâmetros	Dispositivo transdutor.
Retorno	Assinatura digital do pacote de dados.

Por último nesta etapa, o aplicativo solicita à API do Verificador de Conformidades a verificação da assinatura digital de cada pacote de dados de medição. A solicitação de verificação de assinatura é descrita na Tabela 10. Ao usuário será informado apenas a totalização dos dados do abastecimento, assim como apenas o resultado final da verificação das assinaturas. Será informado ao usuário que a assinatura digital é válida apenas se a assinatura digital de todos os pacotes de dados forem válidas. Caso o smartphone esteja sem conexão com a Internet, a verificação é feita parcialmente de forma local, sem verificação da validade dos certificados digitais ou lista de certificados revogados.

Tabela 10: Requisição - Verificação de assinatura digital.

Requisição	Verificação de assinatura digital.
Para	API do Verificador de Conformidades.
Descrição	Permite verificar a assinatura digital de pacote de dados de medição.
Parâmetros	Pacote de dados de medição, assinatura digital e certificado do transdutor.
Retorno	Resultado da verificação de assinatura.

Independentemente se o smartphone estiver com conexão à Internet ou não, os dados do abastecimento são armazenados localmente no dispositivo. Os dados ficam com o envio ao servidor pendente, cabendo ao usuário compartilhar ou não os seus abastecimentos.

Após visualizar as informações do abastecimento, o usuário também pode realizar uma manifestação através do aplicativo. Isso é possível por meio do botão “Relatar problema”, na tela principal de um abastecimento. Ao clicar nesse botão, surgirá um pop up, informando o usuário que os dados do abastecimentos também serão enviados para avaliação. Ao clicar em “Confirmar”, o aplicativo é redirecionado para a página de login do Gov.br no navegador de Internet, onde o utilizador deverá inserir suas credenciais. O aplicativo envia as informações de login para o sistema do Gov.br, que realiza a autenticação do usuário. Se as credenciais estiverem corretas, o consumidor estará logado. A sequência de procedimentos executados durante a realização de um login Gov.br e uma manifestação é ilustrada na Figura 13. O login é feito de acordo com a Tabela 11. A manifestação é registrada no servidor através da requisição descrita na Tabela 12.

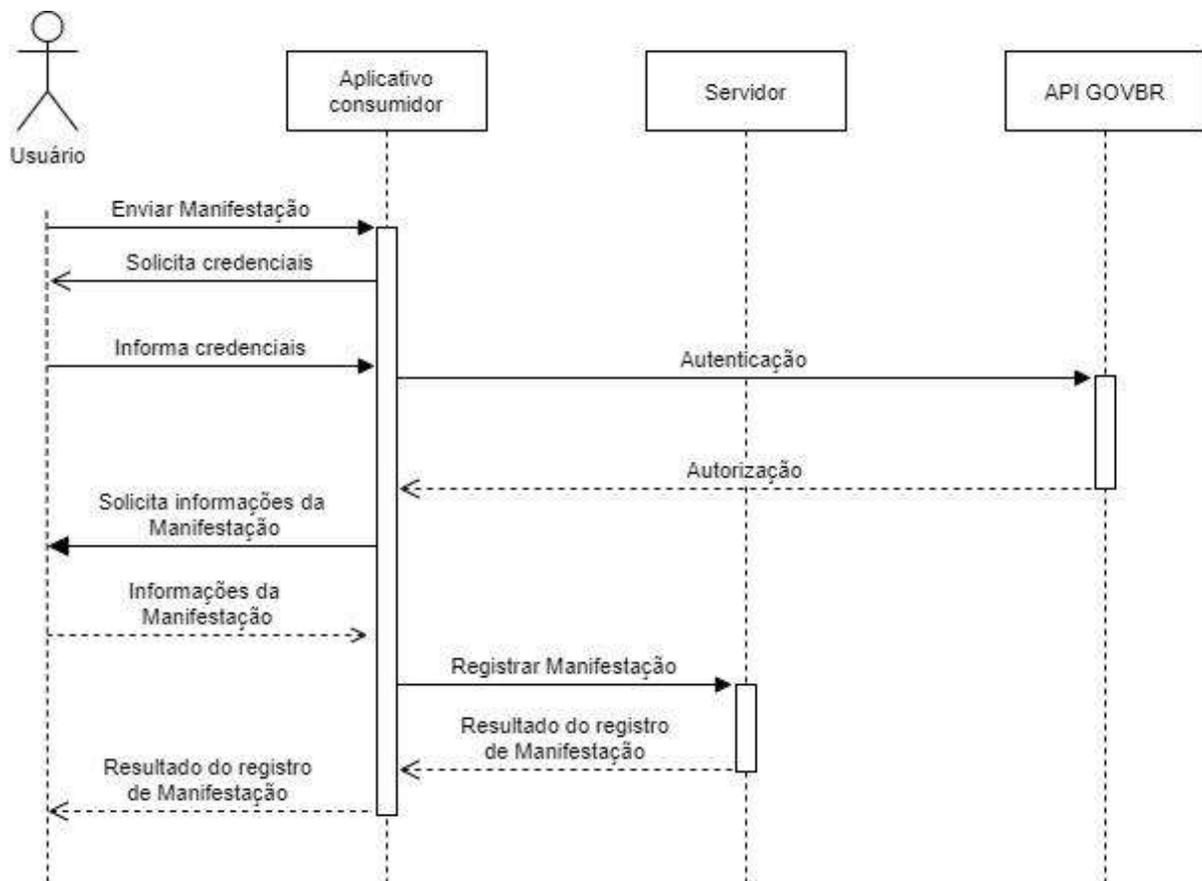


Figura 13: Procedimento para realização de manifestação.

Tabela 11: Requisição - Autenticação do consumidor.

Requisição	Autenticação do consumidor.
Para	API do GOVBR.
Descrição	Permite realizar a autenticação do usuário consumidor no aplicativo.
Parâmetros	CPF e senha.
Retorno	Token de autorização e permissões.

Tabela 12: Requisição - Registro de manifestação.

Requisição	Registro de manifestação.
Para	Servidor.
Descrição	Permite enviar uma manifestação a ser registrada na base de dados.
Parâmetros	Tipo (reclamação ou denúncia), destinatário, assunto, descrição, local, nome do envolvido, função do envolvido, órgão/empresa, nome do usuário, CPF do usuário e identificador do abastecimento.
Retorno	Resultado do registro da manifestação.

Após todo o procedimento de realizar nova leitura ter sido finalizado, esta irá aparecer na lista de últimas leituras. Ao selecionar uma leitura é possível rever os dados do abastecimento. Caso o abastecimento ainda não tenha uma manifestação associada, será possível realizar uma. A sequência de procedimentos realizados ao selecionar uma leitura e realizar uma manifestação é ilustrada na Figura 14.

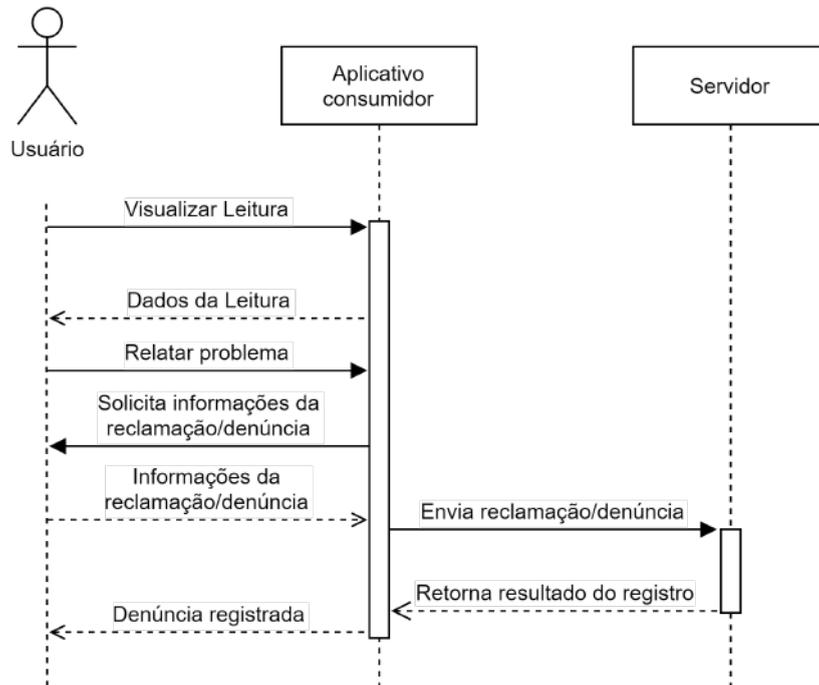


Figura 14: Procedimentos para rever leitura e realizar manifestação.

Com uma leitura já concretizada, o usuário pode fazer o envio dos dados para o servidor, a partir da tela inicial do aplicativo. Esse procedimento ocorre por meio do botão “Deseja enviar seus dados para o INMETRO?”, que estará disponível sempre que houver pelo menos um abastecimento no histórico que ainda não foi enviado ao servidor. A sequência desse procedimento é ilustrada na Figura 15 e detalhada na Tabela 13. Vale lembrar que o aplicativo necessita de uma conexão com a internet para realizar esse processo.

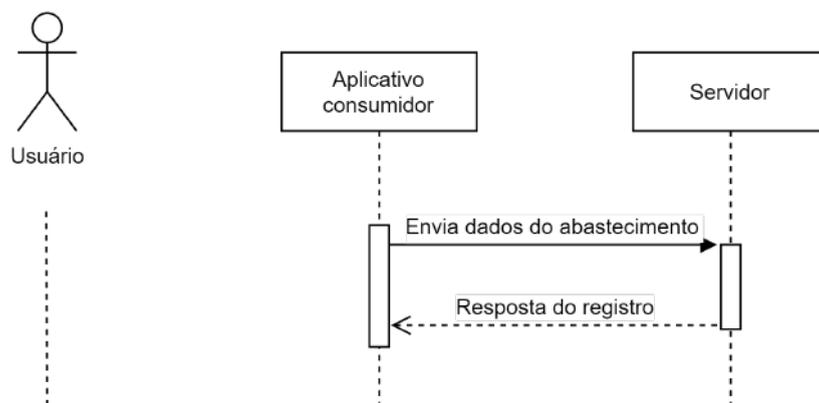


Figura 15: Envio de dados do abastecimento para o servidor.

Tabela 13: Requisição - Envio de dados do abastecimento.

Requisição	Envio de dados do abastecimento.
Para	Servidor.
Descrição	Permite registrar na base de dados os dados coletados pelo aplicativo referentes a um abastecimento.
Parâmetros	Pacote de dados de medição, assinaturas digitais dos pacotes de dados de medição e número serial da bomba de combustível.
Retorno	Resultado do registro dos dados do abastecimento.

2.4.2 Aplicativo Fiscal

Ao iniciar o aplicativo fiscal, são exibidas as leituras de dados de bombas de combustíveis realizadas que ainda não foram enviadas para o servidor e a opção de realizar nova leitura. Ao escolher a opção de realizar nova leitura, o aplicativo realiza o procedimento de escaneamento e mostra ao fiscal a lista de bombas de combustíveis encontradas. Nesse momento, é solicitado ao fiscal que selecione a bomba de combustível que ele deseja obter informações e o aplicativo estabelece a conexão bluetooth com a bomba selecionada. A sequência dos procedimentos de inicialização, escaneamento e conexão com a bomba de combustível é ilustrada na Figura 16.

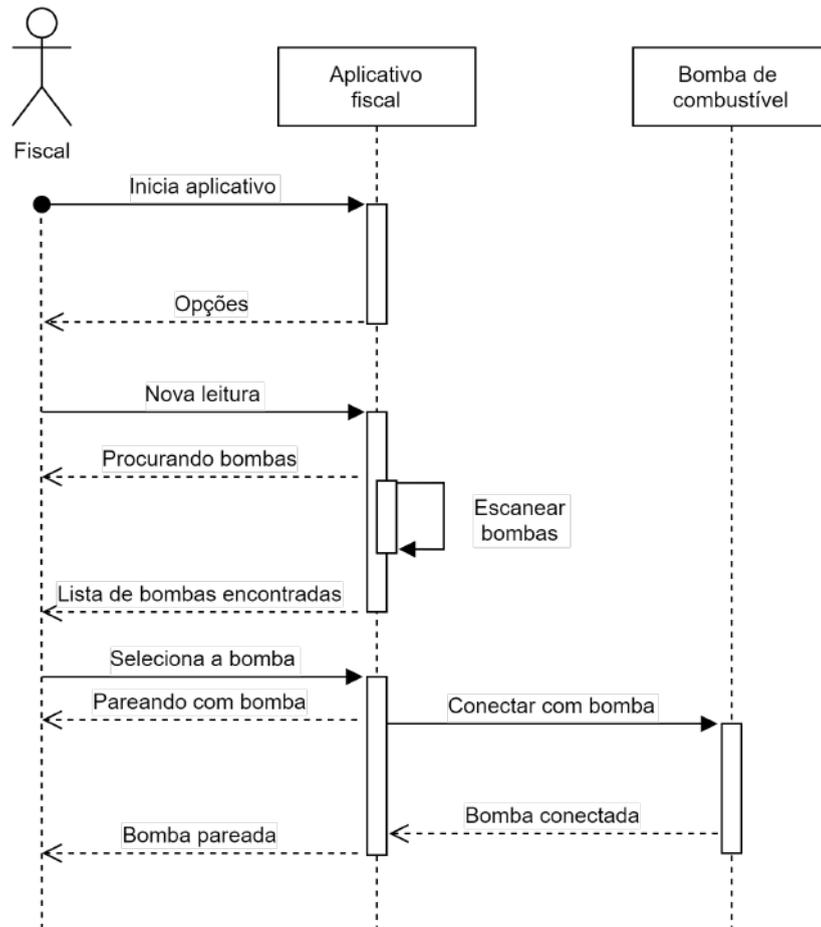


Figura 16: Procedimento de inicialização, escaneamento e conexão com a bomba de combustível.

Após realizar a conexão com a bomba de combustível, o aplicativo consulta algumas informações relevantes do equipamento: a identificação do fabricante, o modelo e o número de série. A sequência de procedimentos que ocorrem nesse processo podem ser vistos na Figura 7, pois são os mesmos do aplicativo do consumidor. Da mesma forma, as requisições para essas consultas estão descritas nas Tabelas 1, 2 e 3.

Logo após, o aplicativo consulta o número de dispositivos (indicadores e transdutores) instalados na bomba de combustível. Como a sequência dos procedimentos de consulta de dispositivos instalados é a mesma para o aplicativo do consumidor, ela pode ser encontrada na Figura 8. A requisição também é a mesma, podendo ser encontrada na Tabela 4.

Na sequência, o aplicativo utiliza o número de série requisitado anteriormente para consultar no servidor à qual estabelecimento a bomba de combustível está vinculada. A sequência dos procedimentos de consulta do estabelecimento é a mesma do aplicativo do consumidor, ilustrada na Figura 19. A consulta dos dados do estabelecimento é realizada através da API do servidor, como descrito na Tabela 5. Caso o smartphone esteja sem

conexão com a Internet, os dados relacionados ao estabelecimento aparecem como indisponíveis.

Após isso, o aplicativo realiza o procedimento de verificação de integridade de software para cada dispositivo. Para isso, o aplicativo utiliza o número de dispositivos instalados, que foi buscado anteriormente. Para cada dispositivo instalado, é consultada a versão de software com o comando 01h e é necessário solicitar a geração de hashes de intervalos da memória de programa do dispositivo. A solicitação de geração de hashes de intervalos da memória é realizada através do comando 02h da bomba de combustível. A sequência do procedimento de verificação de integridade de software é ilustrada na Figura 17. As requisições para consultar a versão de software e leitura de hash da memória são descritas na Tabela 14 e 15, respectivamente.

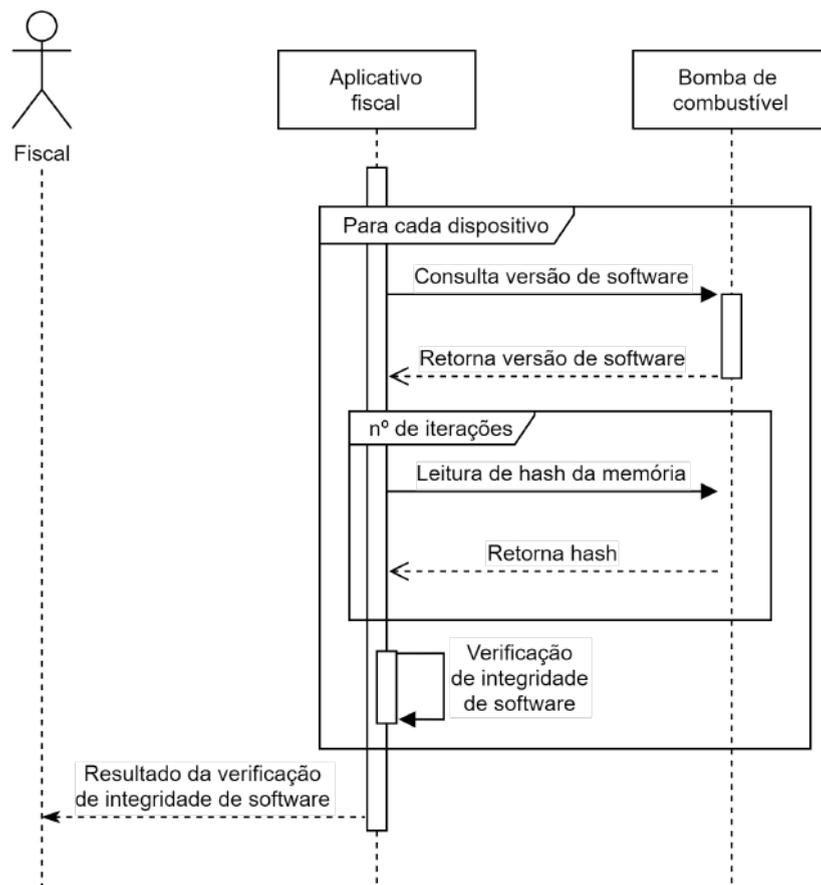


Figura 17: Procedimento de verificação de integridade de software.

Tabela 14: Requisição - Versão de software.

Requisição	Versão de software.
Para	Bomba de combustível (Comando 01h).
Descrição	Permite consultar a versão do software instalado em um componente do instrumento.
Parâmetros	Identificação do componente.
Retorno	Versão do software.

Tabela 15: Requisição - Leitura de hash.

Requisição	Leitura de hash.
Para	Bomba de combustível (Comando 02h).
Descrição	Permite consultar o hash de um intervalo da memória de programa de um dispositivo.
Parâmetros	Identificação do dispositivo, início do intervalo, fim do intervalo e semente.
Retorno	Hash do intervalo de memória.

O aplicativo também realiza a verificação dos certificados digitais para cada dispositivo transdutor encontrado anteriormente. Esse processo inicia com a chamada do comando 43h, que retorna o certificado digital do transdutor. Com a posse do certificado, o aplicativo utiliza uma função verificadora, que retorna o resultado da verificação do certificado digital. Essa operação ocorre para todos os transdutores encontrados na bomba de combustíveis, como é possível verificar na Figura 18. A requisição para consulta do certificado digital do dispositivo transdutor está contida na tabela 8 deste relatório.

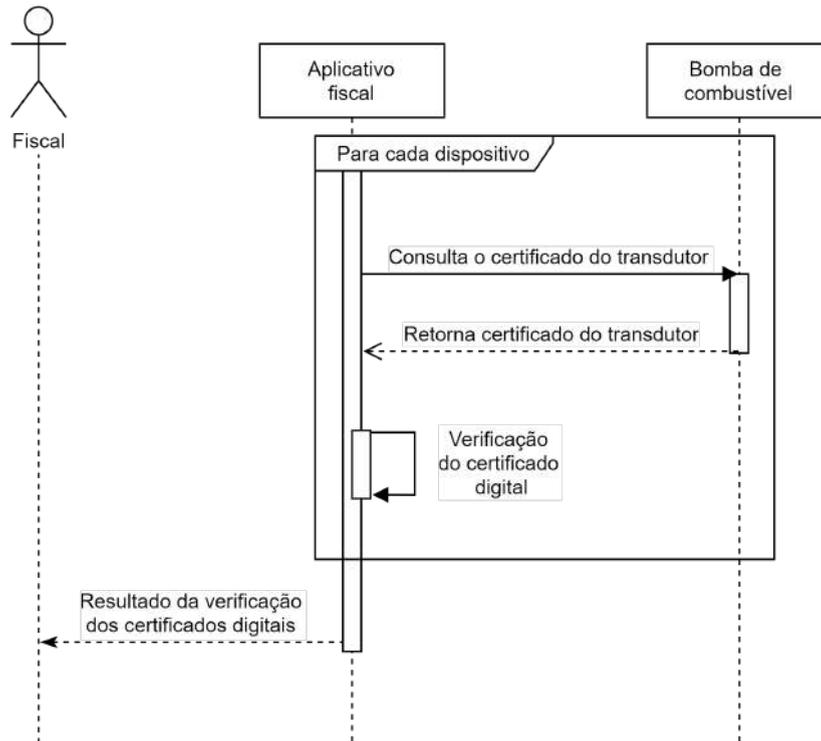


Figura 18: Procedimento de verificação dos certificados digitais.

Outras informações da bomba de combustível também podem ser visualizadas pelo fiscal. Para isso, o aplicativo realiza as consultas de alterações de parâmetros, operações de carga de software, registros de interrupções e registros de manutenções. A sequência de procedimentos para obter os dados de alterações de parâmetros e operações de carga de software é ilustrada na Figura 19

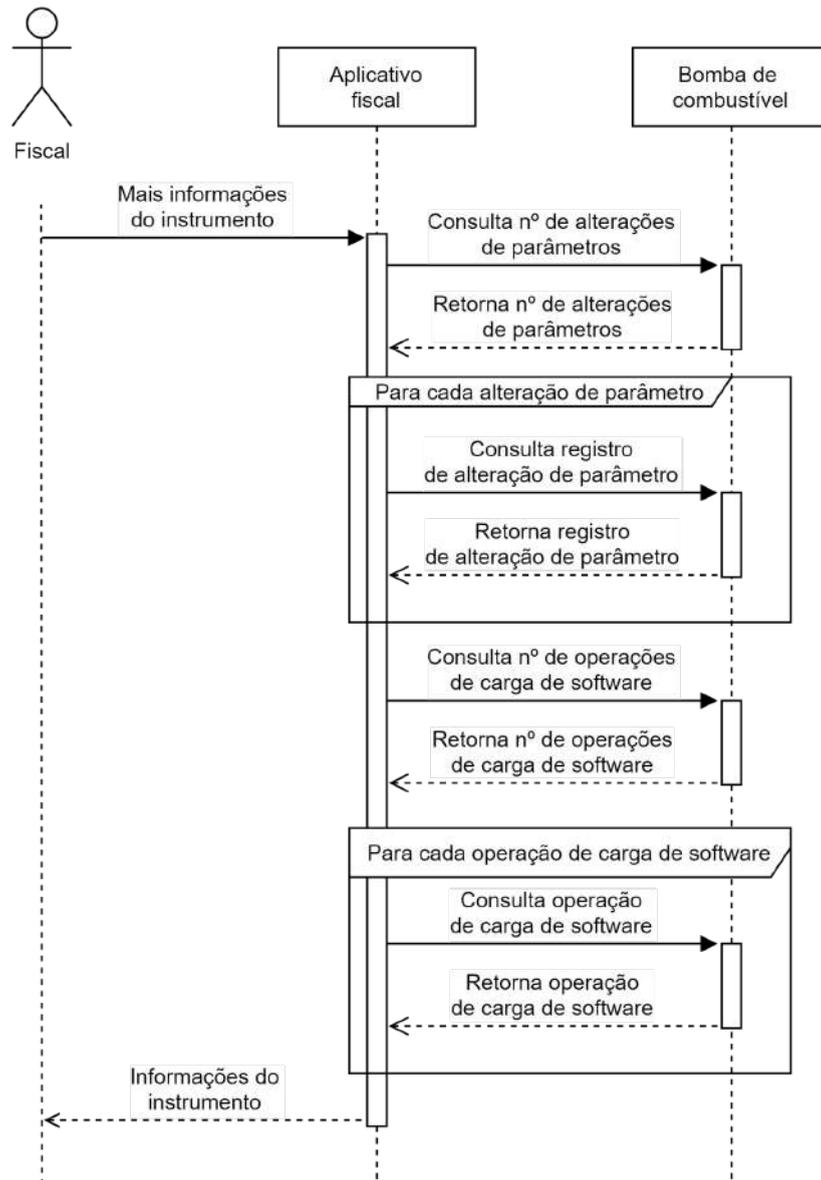


Figura 19: Consulta de alterações de parâmetros e operações de cargas de software..

A requisição para consulta do número de alterações de parâmetros relevantes é realizada através do comando 07h da bomba de combustível, como descrito na Tabela 16. Para cada alteração de parâmetro é realizada a consulta do registro da alteração através do comando 08h, como descrito na Tabela 17.

Tabela 16: Requisição - Número de alterações de parâmetros.

Requisição	Número de alterações de parâmetros.
Para	Bomba de combustível (Comando 07h).
Descrição	Permite consultar o número de alterações de parâmetros relevantes do instrumento.
Parâmetros	Sem parâmetros.
Retorno	Número de alterações de parâmetros.

Tabela 17: Requisição - Registro de alteração de parâmetro.

Requisição	Registro de alteração de parâmetro.
Para	Bomba de combustível (Comando 08h).
Descrição	Permite consultar um registro de alteração de parâmetro relevante.
Parâmetros	Índice do registro.
Retorno	Registro da alteração de parâmetro.

A requisição para consulta do número de cargas de software é realizada através do comando 09h da bomba de combustível, como descrito na Tabela 18. Para cada carga de software é realizada a consulta do registro da carga de software através do comando 0Ah, como descrito na Tabela 19.

Tabela 18: Requisição - Número de cargas de software.

Requisição	Número de cargas de software.
Para	Bomba de combustível (Comando 09h).
Descrição	Permite consultar o número de operações de cargas de software do instrumento.
Parâmetros	Sem parâmetros.
Retorno	Número de operações de cargas de software.

Tabela 19: Requisição - Registro de carga de software.

Requisição	Registro de carga de software.
Para	Bomba de combustível (Comando 0Ah).
Descrição	Permite consultar o registro de uma carga de software.
Parâmetros	Índice do registro.
Retorno	Registro da carga de software.

Para completar os dados obtidos do instrumento, são consultados os registros de interrupções e manutenções. A sequência de procedimentos para essas consultas são ilustradas na Figura 20.

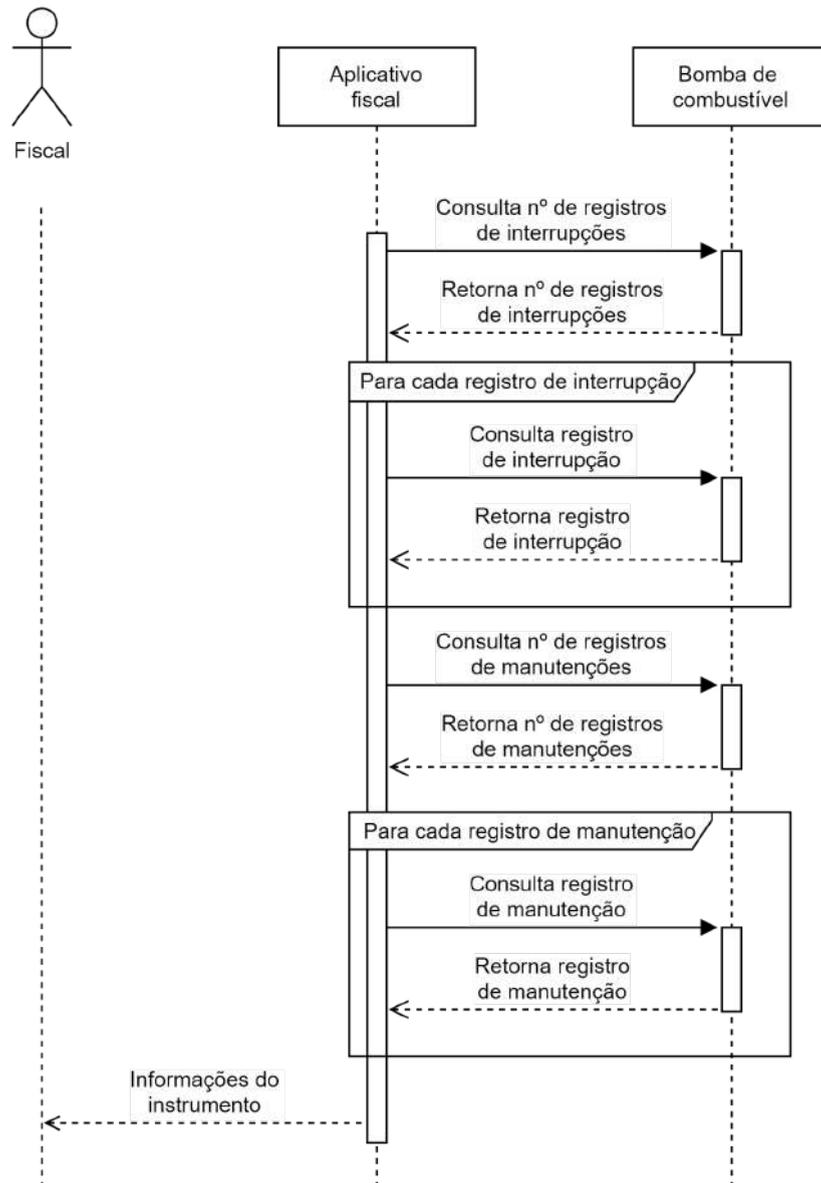


Figura 20: Consulta de registros de interrupções e manutenções.

A requisição para consulta do número de interrupções é realizada através do comando 45h da bomba de combustível, como descrito na Tabela 20. Para cada interrupção é realizada a consulta do registro da interrupção através do comando 46h, como descrito na Tabela 21.

Tabela 20: Requisição - Número de interrupções.

Requisição	Número de interrupções.
Para	Bomba de combustível (Comando 45h).
Descrição	Permite consultar o número de interrupções de funcionamento do instrumento.
Parâmetros	Sem parâmetros.
Retorno	Número de interrupções.

Tabela 21: Requisição - Registro de interrupção.

Requisição	Registro de interrupções.
Para	Bomba de combustível (Comando 46h).
Descrição	Permite consultar um registro de interrupção do instrumento.
Parâmetros	Índice do registro.
Retorno	Registro da alteração de parâmetro.

A requisição para consulta do número de eventos de manutenção é realizada através do comando 47h da bomba de combustível, como descrito na Tabela 22. Para cada evento de manutenção, é realizada a consulta do registro do evento de manutenção através do comando 48h, como descrito na Tabela 23.

Tabela 22: Requisição - Número de eventos de manutenções.

Requisição	Número de eventos de manutenções.
Para	Bomba de combustível (Comando 45h).
Descrição	Permite consultar o número de eventos de manutenções.
Parâmetros	Sem parâmetros.
Retorno	Número de interrupções.

Tabela 23: Requisição - Registro de interrupção.

Requisição	Registro de interrupções.
Para	Bomba de combustível (Comando 46h).
Descrição	Permite consultar um registro de manutenção.
Parâmetros	Índice do registro.
Retorno	Registro da manutenção.

Após isso, o procedimento de fiscalização básico estará finalizado e a fiscalização já aparecerá na tela inicial do aplicativo, na seção que contém o histórico de fiscalizações. Nesse momento, é necessário que o fiscal insira o número do INMETRO referente ao equipamento fiscalizado. Na sequência, o fiscal será redirecionado para a tela principal da fiscalização, que reunirá todas as informações previamente coletadas, organizadas em seções.

A primeira e a segunda seção contém todas as informações referentes à bomba de combustível fiscalizada e ao estabelecimento a que ela pertence. A terceira seção desta tela expõe os resultados relacionados à verificação de integridade de software da bomba de combustível. É possível repetir a verificação de integridade por meio do botão “Repetir verificação”. Na quarta seção, estão os resultados da verificação dos certificados digitais. Também é possível repetir a verificação utilizando o botão “Repetir verificação”. A quinta seção é responsável pelos abastecimentos de ensaio. Selecionando-a, o fiscal será redirecionado para uma tela muito parecida com a tela inicial do aplicativo do consumidor. A função aqui também é muito similar ao aplicativo do consumidor: possibilitar ao fiscal a função de verificação de abastecimentos, seguindo os passos elucidados na seção 2.4.1 deste relatório. A sexta seção exibe os registros buscados durante o procedimento de fiscalização básica. É possível verificar os registros separados em quatro categorias: alteração de parâmetros relevantes, cargas de software, interrupções no funcionamento do instrumento e eventos de manutenção.

Os dados de toda fiscalização realizada são armazenados de forma local. Porém, o fiscal pode fazer o envio de suas leituras para o servidor do Sistema Medida Inteligente. Isso pode ser feito por meio do botão “Reportar leitura”, na parte inferior da tela principal da fiscalização, ou pelo botão “Enviar fiscalização”, contido na tela inicial do aplicativo.

No momento em que o fiscal solicitar o envio dos dados, será requisitado que ele realize uma autenticação. A autenticação do usuário fiscal é realizada através da API do SGI. O fiscal deve inserir seu CPF, senha e UF. Após inseridas as informações, o aplicativo

envia-as para o Servidor e este encaminha para a API do SGI uma solicitação de autenticação. Se as credenciais estiverem cadastradas, o servidor retorna a autorização de acesso.

Após a autenticação, o aplicativo realiza o procedimento de envio dos dados. O registro completo de uma fiscalização inclui os dados do instrumento, o resultado da verificação de integridade de software, os certificados digitais, uma lista de abastecimentos de testes (incluindo os pacotes de dados de medição e assinaturas digitais dos pacotes de medição) e os registros de logs do instrumento. O procedimento para autenticação do fiscal e para enviar o registro completo dos dados para o servidor é ilustrado na Figura 21.

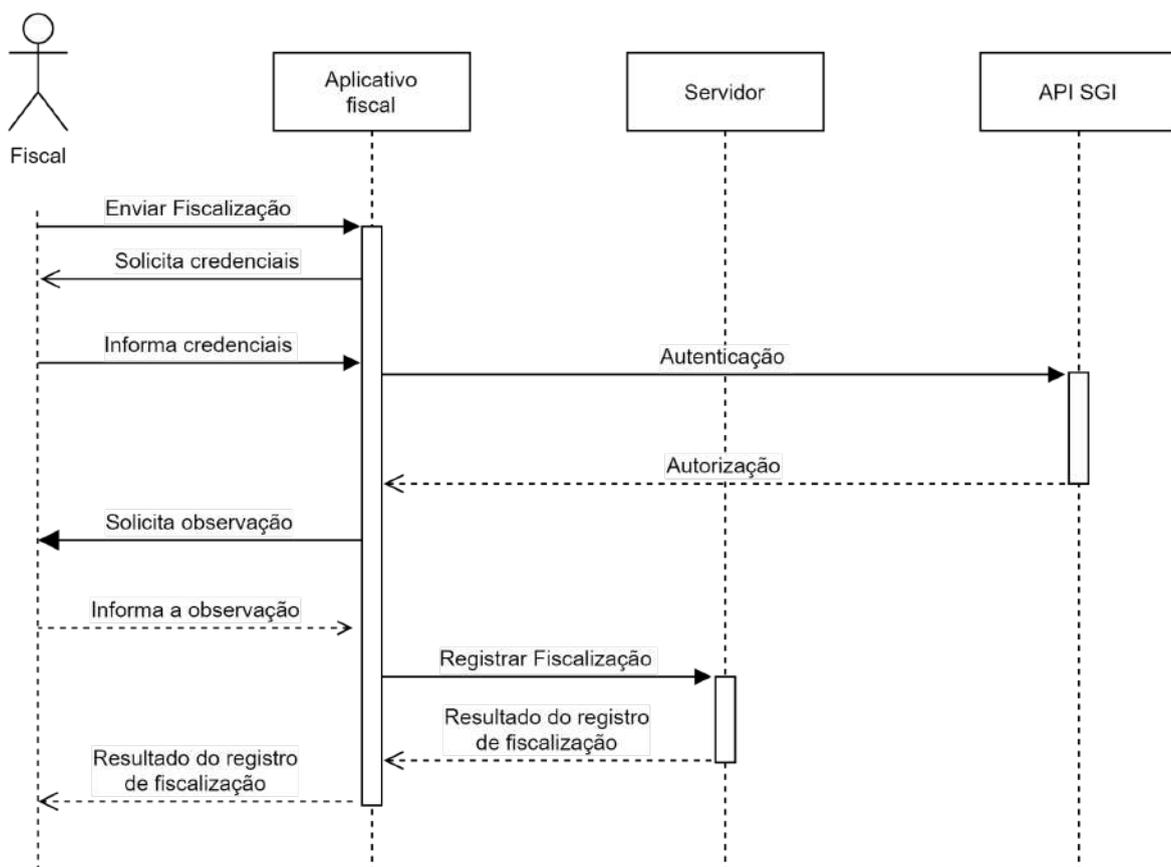


Figura 21: Procedimento para enviar os dados da leitura para o servidor.

As requisições para autenticação do usuário fiscal e para registrar leitura no servidor são descritas nas Tabelas 24 e 25.

Tabela 24: Requisição - Autenticação do fiscal.

Requisição	Autenticação do fiscal.
Para	API do SGI (encaminhada pelo Servidor).
Descrição	Permite realizar a autenticação do usuário fiscal no aplicativo.
Parâmetros	CPF, senha e UF.
Retorno	Token de autorização, nome do fiscal e permissões.

Tabela 25: Requisição - Registrar leitura.

Requisição	Registrar leitura.
Para	Servidor.
Descrição	Permite registrar na base de dados os dados obtidos da bomba de combustível.
Parâmetros	Token de autorização, número de série do instrumento, versão de software dos dispositivos associados, identificação do fabricante, modelo do instrumento, lista de alterações de parâmetros relevantes, lista de cargas de software, listas de dispositivos indicadores e transdutores instalados na bomba, lista de abastecimentos de teste (dispositivos associados, pacotes de dados de mediação, assinatura digital dos pacotes de dados), lista de certificados digitais, resultados das verificações dos certificados, resultado da verificação de integridade de software, lista de registros de interrupções, lista de registros de manutenções e local point.
Retorno	Resultado do registro da leitura.

No momento em que os dados de uma fiscalização são recebidos pelo servidor, são realizadas algumas verificações. Inicialmente é verificado se o fiscal associado já está registrado na base de dados, caso não esteja, são inseridos o CPF e nome do fiscal na base de dados. Além disso, consulta-se se a bomba de combustível já está registrada na base de dados, caso não esteja é então criado o registro de uma nova bomba de

combustível na base de dados. Quando um novo registro de bomba de combustível é criado, é necessário consultar a qual estabelecimento esta bomba está vinculada. Para isso, consulta-se a API do SGI. Ao informar o número serial da bomba de combustível a API retorna a identificação do estabelecimento vinculado. Se este estabelecimento não estiver registrado na base de dados, então solicita-se à API do SGI os dados do estabelecimento e cria-se um novo registro na base de dados. A Figura 22 ilustra o procedimento de vinculação da bomba de combustível e obtenção dos dados do estabelecimento.

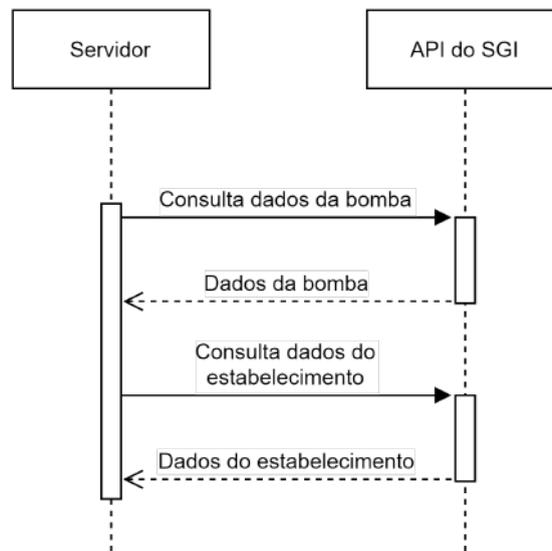


Figura 22: Procedimento para obtenção vinculação da bomba de combustível e dados do estabelecimento.

As requisições para consultar a vinculação de uma bomba de combustível e obtenção dos dados de um estabelecimento são descritas nas Tabelas 26 e 27, respectivamente.

Tabela 26: Requisição - Dados da bomba de combustível.

Requisição	Dados da bomba de combustível.
Para	API do SGI.
Descrição	Permite descobrir a vinculação de uma bomba de combustível ao estabelecimento.
Parâmetros	Número serial e número do INMETRO da bomba de combustível.
Retorno	Identificação do estabelecimento, número do Inmetro, marca e modelo.

Tabela 27: Requisição - Dados do estabelecimento.

Requisição	Dados do estabelecimento.
Para	API do SGI.
Descrição	Permite consultar os dados de um estabelecimento.
Parâmetros	Identificação do estabelecimento.
Retorno	CNPJ, razão social e endereço.

3 Modelagem do Banco de Dados

As bases de dados de certificados digitais, objetos metrológicos e registros metrológicos foram modeladas como um único banco de dados relacional. Inicialmente, apresenta-se na seção 3.1 as tabelas dos bancos de dados com os atributos que a compõem. Em seguida, apresenta-se o diagrama de entidade-relacionamento com o relacionamento entre as tabelas na seção 3.2

3.1 Tabelas

Nesta seção são apresentadas as tabelas do banco de dados.

3.1.1 Tabela de Estabelecimentos

Na tabela de estabelecimentos são armazenados alguns dados dos postos de combustíveis como razão social, CNPJ e endereço. O registro de estabelecimentos tem como objetivo permitir a vinculação das bombas de combustível.

Na base de dados, todas as bombas de combustível estarão vinculadas a um estabelecimento. Isso irá permitir listar as bombas de combustível de um estabelecimento ou consultar a qual estabelecimento a bomba de combustível pertence.

Os detalhes da tabela de estabelecimentos são definidos na Tabela 28.

Tabela 28: Tabela - estabelecimentos

Atributo	Tipo	Descrição
id (PK)	INT	ID interno autogerado do estabelecimento.
id_SGI	VARCHAR(255)	ID do estabelecimento no SGI.
CNPJ	CHAR(14)	CNPJ do estabelecimento.
razao_social	VARCHAR(255)	Razão social do estabelecimento.
UF	ENUM	Unidade Federativa onde o estabelecimento está localizado.
municipio	VARCHAR(255)	Município onde o estabelecimento está localizado, dentro da Unidade Federativa definida anteriormente.

endereco	VARCHAR(255)	Endereço do estabelecimento dentro do município e UF definidos anteriormente.
local	POINT	Coordenadas (latitude e longitude) da localização geográfica do estabelecimento.

3.1.2 Tabela de Bombas Medidoras de Combustível (BMCs)

Na tabela de bombas medidoras de combustível são armazenadas as informações sobre as bombas de combustível como tipo do instrumento, modelo e fabricante. Essa tabela está no centro da maior parte das operações, já que ela se relaciona com quase toda a base de dados direta ou indiretamente. Cada BMC está associada a um estabelecimento e a um conjunto de dispositivos indicadores e transdutores. A vinculação do estabelecimento é realizada na própria tabela de BMCs, enquanto os dispositivos são associados através do identificador único da BMC nas suas respectivas tabelas. Além disso, eventos de fiscalização e manifestações de denúncias também são vinculados à BMC que originou os dados pela identificação única da BMC.

Os detalhes da tabela de bombas medidoras de combustível são mostrados a seguir na Tabela 29.

Tabela 29: Tabela - bombas_medidoras_combustivel

Atributo	Tipo	Descrição
id (PK)	INT	ID interno autogerado da BMC.
serial (PK)	VARCHAR(255)	Número serial da BMC.
numero_inmetro	INT	ID da BMC no SGI.
modelo	VARCHAR(255)	Identificação do modelo da BMC.
fabricante	VARCHAR(255)	Identificação do fabricante da BMC.
local	POINT	Local de instalação do instrumento.
fk_estabelecimen tos_id (FK)	INT	ID interno do estabelecimento ao qual a BMC pertence.

3.1.3 Tabela de Dispositivos Indicadores

Na tabela de dispositivos indicadores é armazenada a relação entre os dispositivos indicadores e as bombas de combustível. A relação é informada por um comando da BMC que retorna os dispositivos instalados no instrumento de medição. Cada dispositivo indicador recebe um identificador serial dentro da BMC (ex. Indicador 1 da bomba de nº serial X). Essa informação chega até a base de dados através de registros de eventos de fiscalização, que informam os dispositivos associados ao instrumento de medição aferido. A relação entre dispositivo indicador e BMC é armazenada como uma chave primária dupla, para garantir que não haja mais de um dispositivo indicador de mesmo número dentro de uma mesma bomba medidora de combustível. Para facilitar a referência aos indicadores, eles também recebem uma identificação unívoca interna do banco de dados nessa tabela.

Os detalhes da tabela de dispositivos indicadores são definidos na Tabela 30 a seguir.

Tabela 30: Tabela - dispositivos_indicadores

Atributo	Tipo	Descrição
ID(PK)	INT	Identificação unívoca autogerada do dispositivo indicador dentro do banco de dados.
fk_bmc_serial (FK)	VARCHAR(30)	Número serial da BMC a qual o dispositivo indicador pertence.
numero_na_bmc	INT	Número do dispositivo indicador em relação à BMC que ele pertence.
versao_software	DECIMAL	Número da versão do software instalado no dispositivo indicador.

3.1.4 Tabela de Dispositivos Transdutores

Na tabela de dispositivos transdutores é armazenada a relação entre os dispositivos transdutores e as bombas medidoras de combustível. Cada dispositivo transdutor é associado a uma bomba de combustível. Portanto, para cada dispositivo transdutor é armazenado o número serial associado à BMC e uma identificação unívoca dentro da base de dados.

A tabela de transdutores se relaciona com a tabela de pacotes de dados dos abastecimentos, pois os abastecimentos são formados por um conjunto de pacotes de dados e estes por sua vez estão associados aos transdutores. Além disso, a tabela de transdutores se relaciona com a tabela de certificados digitais, pois cada transdutor possui um certificado digital associado a ele.

Os detalhes da tabela de dispositivos transdutores são definidos na Tabela 31.

Tabela 31: Tabela - dispositivos_transdutores

Atributo	Tipo	Descrição
ID (PK)	INT	ID único autogerado do dispositivo transdutor.
numero_na_bmc	INT	Número do dispositivo transdutor em relação à BMC que ele pertence.
fk_bmc_serial(FK)	VARCHAR(255)	Número serial da BMC a qual o dispositivo transdutor pertence.
versao_software	DECIMAL	Número da versão do software instalado no dispositivo transdutor.
serial_certificado	VARCHAR(255)	Número serial do certificado associado a esse transdutor.

3.1.5 Tabela de Registros Metrológicos

Na tabela de registros metrológicos são armazenados os dados sobre os abastecimentos realizados pelas bombas de combustível. Cada registro metrológico é vinculado a um dispositivo indicador dentro da BMC vinculado ao abastecimento. Além disso, o registro metrológico é composto por um conjunto de pacotes de dados emitidos por dispositivos transdutores associados ao abastecimento. Dessa forma, esta tabela armazena um identificador único e a vinculação do dispositivo indicador da BMC. A identificação única do abastecimento é utilizado para indexação dos pacotes de dados.

Os detalhes da tabela de registros metrológicos são definidos na Tabela 32 a seguir.

Tabela 32: Tabela - registros_metrologicos

Atributo	Tipo	Descrição
id (PK)	INT	Identificador interno autogerado do registro metrológico.
fk_indicadores_id (FK)	INT	Identificador interno do dispositivo indicador que contém os dados sobre este registro metrológico.
registro	VARCHAR(255)	Detalhamento do registro metrológico

3.1.6 Tabela de Pacotes de Dados dos Abastecimentos

Na tabela de pacotes de dados dos abastecimentos são armazenados os dados emitidos por um dispositivo transdutor referente a um abastecimento, como constante de calibração, volume, valor por litro, valor monetário, etc. Também são armazenados a identificação do dispositivo transdutor que emitiu o dado e o identificador do registro metrológico que totaliza os dados do abastecimento.

Os detalhes da tabela de pacote de dados dos abastecimentos são definidos na Tabela 33 a seguir.

Tabela 33: Tabela - pacotes_dados_abastecimentos

Atributo	Tipo	Descrição
id (PK)	INT	Identificador interno autogerado do pacote de dados.
fk_registros_id (FK)	INT	Identificador interno do registro metrológico ao qual o pacote de dados pertence.
timestamp	TIMESTAMP	Data e hora da criação do pacote de dados.
controlador	VARCHAR(255)	Código da identificação unívoca do dispositivo controlador que fez a medição do abastecimento.
id_abastecimento_controlador	VARCHAR(255)	Identificação do abastecimento vinda dos dados registrados pelo dispositivo controlador.

id_abastecimento_transdutor	VARCHAR(255)	Identificação do abastecimento vinda dos dados registrados pelo dispositivo transdutor.
constante_calibracao	DECIMAL	Constante de calibração do dispositivo transdutor.
volume_transdutor	DECIMAL	Volume do combustível usado no abastecimento medido pelo dispositivo transdutor.
volume_controlador	DECIMAL	Volume total da transação de acordo com a medição do dispositivo controlador.
valor_monetario	DECIMAL	Valor monetário total da transação.
preco_litro	DECIMAL	Preço por litro do combustível no momento da transação.

3.1.7 Tabela de Certificados Digitais

Na tabela de certificados são armazenados os certificados digitais emitidos pelas Autoridades Certificadoras para os dispositivos transdutores. Nessa tabela são armazenadas colunas como chave-pública, número serial, *common name*, validade e o próprio certificado. Além disso, é armazenado um campo para indicar se o certificado foi revogado. Esse campo é atualizado conforme o recebimento de novas listas de certificados revogados.

Os detalhes da tabela de certificados são definidos na Tabela 34 a seguir.

Tabela 34: Tabela - certificados

Atributo	Tipo	Descrição
serial (PK)	VARCHAR(255)	Número serial do certificado digital.
certificado	BLOB	Certificado digital inteiro codificado em formato DER.
chave_publica	VARCHAR(255)	Chave pública do certificado codificada em formato 'base64'.

cn_subject	VARCHAR(255)	“ <i>Common name</i> ” (nome comum) do “ <i>subject</i> ” (requerente/titular(?)) do certificado, de acordo com a nomenclatura padrão dos campos do certificado.
not_before	INT	Data de criação do certificado, de acordo com a nomenclatura padrão dos campos do certificado. Armazenada como horário Unix, definido como o número de segundos passados desde 1 de Janeiro de 1970 (<i>epoch</i>).
not_after	INT	Prazo de validade do certificado, de acordo com a nomenclatura padrão dos campos do certificado. Armazenado da mesma forma que o atributo anterior.
revogado	TINYINT(1)	Valor numérico representando verdadeiro (1) ou falso (0) para dizer se o certificado já foi revogado ou não.

3.1.8 Tabela de Fiscais

Na tabela de fiscais são armazenados alguns dados dos fiscais registrados no sistema. Os dados dos fiscais são obtidos automaticamente através da API do SGI, sem a necessidade de um novo cadastro. O registro de fiscais é utilizado apenas para associar o agente fiscal que emitiu um evento de fiscalização. Portanto, a tabela de fiscais se relaciona com a tabela de eventos de fiscalização através do CPF do fiscal.

Os detalhes da tabela de fiscais são definidos na Tabela 35 a seguir.

Tabela 35: Tabela - fiscais

Atributo	Tipo	Descrição
id (PK)	INT	ID interno autoogerado do fiscal.
CPF	CHAR(11)	CPF do fiscal.
unidade_federativa	INT	ID do estado do fiscal segundo padrão INMETRO.

nome VARCHAR(255) Nome completo do fiscal.

3.1.9 Tabela de Eventos de Fiscalização de BMCs

Na tabela de eventos de fiscalização de BMCs são armazenadas as informações obtidas pelos fiscais em eventos de fiscalização das bombas medidoras de combustível. Os eventos de fiscalização de tipos específicos (por exemplo, cargas de software e alteração de parâmetros) são associados a um evento de fiscalização desta tabela para centralizar a data e fiscal que realizou a fiscalização de certa bomba de combustível associada ao evento. São armazenados um identificador único do evento de fiscalização, o número serial da bomba de combustível correspondente, a data e hora do registro e o identificador do fiscal responsável pela obtenção dos dados.

Os detalhes da tabela de eventos de fiscalização de BMCs são definidos na Tabela 36 a seguir.

Tabela 36: Tabela - eventos_fiscalizacao

Atributo	Tipo	Descrição
id (PK)	INT	Identificador interno do evento de fiscalização.
fk_bmc_serial (FK)	VARCHAR(30)	Número serial da BMC da qual foi feita a fiscalização.
fk_fiscal_CPF (FK)	CHAR(11)	CPF do fiscal que realizou a fiscalização.
timestamp	TIMESTAMP	Data e hora do registro do evento de fiscalização.

3.1.10 Tabela de Manifestações

Na tabela de manifestações são armazenadas as reclamações/denúncias feitas pelos usuários através do aplicativo consumidor. A tabela armazena dados como destinatário, assunto, descrição e tipo de denúncia. Além disso, são armazenados um identificador único da denúncia, a data e hora do registro e o identificador do abastecimento relacionado à denúncia. Para fins de segurança, não são armazenadas informações referentes ao usuário que realizou a manifestação; tais informações como nome, CPF ou e-mail do usuário estarão apenas na plataforma do Fala.BR.

Os detalhes da tabela de manifestações são definidos na Tabela 37 a seguir.

Tabela 37: Tabela - manifestacoes

Atributo	Tipo	Descrição
id (PK)	INT	Identificador interno da manifestação.
fk_abastecimentos_id	INT	ID do abastecimento que causou a manifestação.
timestamp	TIMESTAMP	Data e hora que o usuário relatou a manifestação.
destinatario	INT	ID do órgão público ao qual o usuário deseja destinar a manifestação.
tipo	INT	ID do tipo de denúncia (reclamação ou denúncia).
assunto	INT	ID do assunto/tópico da denúncia.
descricao	VARCHAR(255)	Descrição textual da denúncia.
id_manifestacao	INT	ID da manifestação na base de dados do Fala.BR.

3.1.11 Tabela de Registros de Interrupção

Na tabela de registros de interrupção são armazenados os dados sobre os registros de interrupção enviados nos eventos de fiscalização. Cada registro de interrupção é vinculado a um evento de fiscalização. Dessa forma, esta tabela armazena um identificador único, a vinculação do evento de fiscalização e um detalhamento do registro. A identificação única do registro de interrupção é utilizada para indexação e manipulação no banco.

Os detalhes da tabela de registros de interrupção são definidos na Tabela 38 a seguir.

Tabela 38: Tabela - registros_interrupcao

Atributo	Tipo	Descrição
id (PK)	INT	Identificador interno autogerado do registro metrológico.

fk_eventos_fiscalizacao_id (FK)	INT	Identificador interno do evento de fiscalização.
registro	VARCHAR(255)	Descrição do registro metrológico.

3.1.12 Tabela de Carga de Software

Na tabela de carga de software são armazenados os dados sobre as cargas de software enviadas nos eventos de fiscalização. Cada carga de software é vinculada a um evento de fiscalização. Dessa forma, esta tabela armazena um identificador único, a vinculação do evento de fiscalização e um detalhamento da carga de software. A identificação única do registro de interrupção é utilizada para indexação e manipulação no banco.

Os detalhes da tabela de registros de interrupção são definidos na Tabela 39 a seguir.

Tabela 39: Tabela - carga_software

Atributo	Tipo	Descrição
id (PK)	INT	Identificador interno autogerado da carga de software.
fk_eventos_fiscalizacao_id (FK)	INT	Identificador interno do evento de fiscalização.
registro	VARCHAR(255)	Descrição da carga de software.

3.1.13 Tabela de Registro de Evento de Manutenção

Na tabela de registro de evento de manutenção são armazenados os dados sobre os registros de evento de manutenção enviados nos eventos de fiscalização. Cada registro de evento de manutenção é vinculado a um evento de fiscalização. Dessa forma, esta tabela armazena um identificador único, a vinculação do evento de fiscalização e um detalhamento do registro de evento de manutenção. A identificação única do registro de evento de manutenção é utilizada para indexação e manipulação no banco.

Os detalhes da tabela de registros de interrupção são definidos na Tabela 40 a seguir.

Tabela 40: Tabela - registro_evento_manutencao

Atributo	Tipo	Descrição
id (PK)	INT	Identificador interno autogerado do registro de evento de manutenção.
fk_eventos_fiscali zacao_id (FK)	INT	Identificador interno do evento de fiscalização.
registro	VARCHAR(255)	Descrição do registro evento de manutenção.

3.1.14 Tabela de Alteração de Parâmetros

Na tabela de alteração de parâmetros são armazenados os dados sobre alterações de parâmetros de bombas de combustível enviadas nos eventos de fiscalização. Cada alteração de parâmetro é vinculada a um evento de fiscalização. Esta tabela armazena um identificador único, a vinculação do evento de fiscalização e um detalhamento da alteração de parâmetro. A identificação única da alteração de parâmetros é utilizada para indexação e manipulação no banco.

Os detalhes da tabela de registros de interrupção são definidos na Tabela 41 a seguir.

Tabela 41: Tabela - alteracao_parametros

Atributo	Tipo	Descrição
id (PK)	INT	Identificador interno autogerado da alteração de parâmetros
fk_eventos_fiscali zacao_id (FK)	INT	Identificador interno do evento de fiscalização.
registro	VARCHAR(255)	Descrição da alteração de parâmetros registrada.

3.1.15 Tabela de Ensaio de Bicos

Na tabela de ensaio de bicos são armazenados os dados sobre os ensaios de bicos enviados nos eventos de fiscalização. Cada ensaio de bico é vinculado a um evento de fiscalização e a um abastecimento. Dessa forma, esta tabela armazena um identificador único, a vinculação do evento de fiscalização, a vinculação com o abastecimento, um número do inmetro e um detalhamento do resultado do ensaio. A identificação única é utilizada para indexação e manipulação no banco.

Os detalhes da tabela de ensaio de bicos são definidos na Tabela 42 a seguir.

Tabela 42: Tabela - ensaio_bico

Atributo	Tipo	Descrição
id (PK)	INT	Identificador interno autogerado do ensaio de bico.
fk_eventos_id (FK)	INT	Identificador interno do evento de fiscalização.
fk_abastecimento_s_id (FK)	INT	Identificador interno do abastecimento.
numero_inmetro	VARCHAR(255)	Número do inmetro usado no ensaio de bico.
resultado	VARCHAR(255)	Descrição do resultado do ensaio de bico.

3.1.16 Tabela de Verificação de Dispositivos Indicadores

Na tabela de verificação de dispositivos indicadores são armazenados os dados sobre as verificações de software de dispositivo indicador enviados nos eventos de fiscalização. Cada verificação de dispositivo indicador é vinculada a um evento de fiscalização e a um dispositivo indicador. Dessa forma, esta tabela armazena um identificador único, a vinculação do evento de fiscalização, a vinculação com o dispositivo indicador e um detalhamento do resultado da verificação. A identificação única é utilizada para indexação e manipulação no banco.

Os detalhes da tabela de verificação de dispositivo indicador são definidos na Tabela 43 a seguir.

Tabela 43: Tabela - verificacao_dispositivo_indicador

Atributo	Tipo	Descrição
id (PK)	INT	Identificador interno autogerado da verificação de dispositivo indicador.
fk_eventos_fiscali zacao_id (FK)	INT	Identificador interno do evento de fiscalização.
fk_indicador_id (FK)	INT	Identificador interno do dispositivo indicador.
resultado	VARCHAR(255)	Descrição do resultado da verificação de dispositivo indicador.

3.1.17 Tabela de Verificação de Dispositivos Transdutores

Na tabela de verificação de dispositivos transdutores são armazenados os dados sobre as verificações de software de dispositivos transdutores enviados nos eventos de fiscalização. Cada verificação de dispositivo transdutor é vinculada a um evento de fiscalização e a um dispositivo transdutor. Dessa forma, esta tabela armazena um identificador único, a vinculação do evento de fiscalização, a vinculação com o dispositivo transdutor e um detalhamento do resultado da verificação. A identificação única é utilizada para indexação e manipulação no banco.

Os detalhes da tabela de verificação de dispositivo transdutor são definidos na Tabela 44 a seguir.

Tabela 44: Tabela - verificacao_dispositivo_transdutor

Atributo	Tipo	Descrição
id (PK)	INT	Identificador interno autogerado da verificação de dispositivo transdutor.
fk_eventos_fiscali zacao_id (FK)	INT	Identificador interno do evento de fiscalização.
fk_indicador_id	INT	Identificador interno do dispositivo

(FK)		transdutor.
resultado	VARCHAR(255)	Descrição do resultado da verificação de dispositivo transdutor.

3.1.18 Tabela de Modelos de Bombas Medidoras de Combustíveis

Na tabela de modelos de bombas medidoras de combustível são armazenados os dados sobre os modelos de bombas de combustíveis dos fabricantes. Esta tabela armazena um identificador único, um fabricante, um modelo e dois nomes de dois *comma separated values files (CSVs)*, um para os transdutores e outro para os indicadores, que contém as informações de hashes de intervalo de memória para o software de dispositivos desse modelo.

Os detalhes da tabela de modelo de BMC são definidos na Tabela 45 a seguir.

Tabela 45: Tabela - modelo_bmc

Atributo	Tipo	Descrição
id (PK)	INT	Identificador interno autogerado do modelo.
fabricante	VARCHAR(255)	Descrição do fabricante da BMC.
modelo	VARCHAR(255)	Descrição do modelo da BMC
csv_indicadores	VARCHAR(255)	Nome do arquivo CSV com as informações sobre os hashes de intervalo de memória de software dos dispositivos indicadores deste modelo em específico.
csv_transdutores	VARCHAR(255)	Nome do arquivo CSV com as informações sobre os hashes de intervalo de memória de software dos dispositivos transdutores deste modelo em específico.

3.1.19 Tabela de Tabelas de Referência de Software

Na tabela de tabelas de referência de software são armazenados os dados sobre as tabelas de referência de software cadastradas. Esta tabela armazena um identificador único, um fabricante, um tipo de dispositivo e a versão de software. A identificação única é utilizada para indexação e manipulação no banco.

Os detalhes da tabela de tabelas de referência de software são definidos na Tabela 46 a seguir.

Tabela 46: Tabela - tabelas_referencia_software

Atributo	Tipo	Descrição
id (PK)	INT	Identificador interno autogerado das tabelas de referência de software.
fabricante	VARCHAR(255)	Descrição do fabricante da BMC.
tipo_dispositivo	ENUM(...)	Descrição do tipo de dispositivo a que a tabela se refere.
versao_software	DECIMAL	Versão de software do dispositivo.

3.1.20 Tabela de Hash de Intervalo de Memória

Na tabela de hash de intervalo de memória são armazenados os hashes dos intervalos de memória do software dos dispositivos transdutores e indicadores, de acordo com as especificações de cada fabricante. Esta tabela armazena um identificador único, a vinculação com a uma tabela de referência de software, um endereço inicial, um endereço final e o hash para esse intervalo de memória. A identificação única é utilizada para indexação e manipulação no banco.

Os detalhes da tabela de tabelas de referência de software são definidos na Tabela 47 a seguir.

Tabela 47: Tabela - tabelas_referencia_software

Atributo	Tipo	Descrição
id (PK)	INT	Identificador interno autogerado das tabelas de referência de software.
fabricante	VARCHAR(255)	Descrição do fabricante da BMC.
tipo_dispositivo	ENUM(...)	Descrição do tipo de dispositivo a que a tabela se refere.
versao_software	DECIMAL	Versão de software do dispositivo.

3.2 Diagrama do Banco de Dados

O diagrama do banco de dados está definido na Figura 23 a seguir. Ele expressa a relação entre as tabelas apresentadas na seção 3.1 usando a notação “*Crow’s Foot*” (pé de corvo) para indicar a cardinalidade.

A tabela de certificados representa a Base de Dados de Certificados OM-BR. As tabelas de bombas medidoras de combustíveis, estabelecimentos, dispositivos indicadores e dispositivos transdutores representam a Base de Dados de Objetos Metrológicos. Já as tabelas de eventos de fiscalização, alteração de parâmetros, registros de interrupção, cargas de software, registros de eventos de manutenção, ensaio de bico e verificação de dispositivos transdutor e indicador, denúncias, registros metrológicos e pacotes de dados de abastecimentos representam a Base de Dados de Registros Metrológicos. Por fim, as tabelas de fiscais, modelos de bombas de combustíveis, hashes de intervalo de memória e tabelas de referência de software são tabelas auxiliares.

Pode-se entender que a tabela de bombas medidoras de combustíveis está no centro do modelo, relacionando-se direta ou indiretamente com a maioria das outras tabelas. À tabela de bombas medidoras de combustíveis são associados os estabelecimentos, os eventos de fiscalização e denúncias. Também são vinculados os dispositivos indicadores e transdutores. Indiretamente, são vinculados os certificados digitais, através dos dispositivos transdutores, e dados dos abastecimentos através dos dispositivos indicadores.

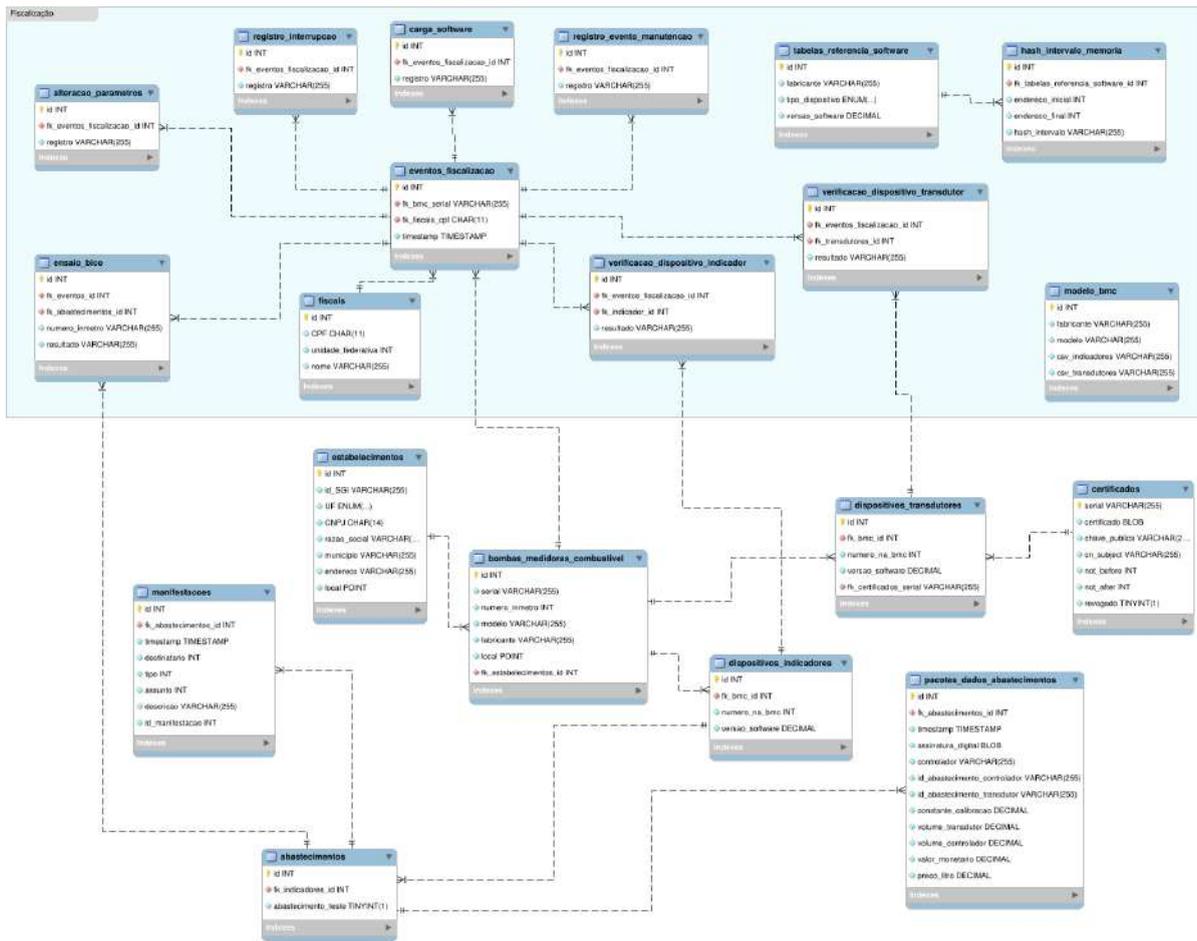


Figura 23: Diagrama do banco de dados.

4 Tecnologias Utilizadas

As tecnologias bases utilizadas são listadas a seguir para cada macro-componente do sistema.

- Servidor:
 - Plataforma: Java 18.
 - Framework: Spring Boot Versão 2.54, Hibernate Versão 5.6.5.
 - Gerenciador de dependências: Maven Versão 3.8.2.
 - Banco de dados: MySQL Versão 8.0.28+.
 - Containerização: Docker Versão 20.10.17, Docker Compose Versão 2.9.0.

- Aplicativo Móvel:
 - Plataforma: Flutter versão 2.10.5.
 - Bibliotecas Principais:
 - Flutter Blue versão 0.8.0
 - SQLite versão 2.0.2

5 Aspectos de Segurança

Nesta seção são detalhados alguns aspectos de segurança do projeto, tanto na parte dos aplicativos móveis (front-end) quanto no servidor (back-end). Para cada setor, serão descritos em tópicos medidas de segurança já adotadas para prevenir ataques e também aspectos que podem causar vulnerabilidades.

5.1 Aplicativos móveis

Os aspectos de segurança desta seção se referem tanto ao aplicativo do consumidor quanto ao aplicativo do fiscal.

5.1.1 Medidas adotadas

- Tokens de acesso já expirados são apagados da base de dados dos aplicativos móveis para que seja armazenado apenas o token necessário ao acesso.
- O usuário pode deletar informações do banco de dados do celular, como fiscalizações e abastecimentos, por meio dos aplicativos.

5.1.2 Possíveis vulnerabilidades

- Não existe a opção de proteger os aplicativos com senhas, então é possível que um invasor do celular modifique informações já armazenadas no banco de dados do celular tanto no aplicativo do consumidor quanto no aplicativo do fiscal.
- É possível que dados de login do usuário no GovBR ou SGI do INMETRO fiquem armazenados na memória temporária do celular após a obtenção do token de acesso, o que permitiria que um invasor do celular acessasse a conta do usuário posteriormente.

5.2 Servidor

Os aspectos de segurança a seguir se referem à aplicação back-end em execução no servidor, incluindo o banco de dados.

5.2.1 Medidas adotadas

- Tokens de acesso não são armazenados no servidor de forma íntegra para realizar o controle de acesso, mas sim o seu *hash*, o que impede de extrair informações pessoais dos tokens dentro do servidor.
- Dados pessoais do usuário do aplicativo consumidor, como CPF, nome, e-mail e localização não são armazenados em qualquer tabela do banco de dados ou relacionadas a abastecimentos ou manifestações.
- As requisições ao servidor possuem são feitas pelo protocolo HTTPS, logo são criptografadas enquanto o certificado TLS do servidor ainda estiver válido.
- A biblioteca Hibernate checa textos inseridos para possíveis ataques de injeção SQL de forma automática.

5.2.2 Possíveis vulnerabilidades

- O banco de dados não possui comunicação criptografada com TLS por padrão. Caso o banco de dados seja executado em uma máquina diferente da aplicação, a transmissão de dados não seria criptografada. Para mitigar isso, é necessário configurar um certificado TLS para o banco de dados e executar o MySQL especificando o caminho desse certificado, o caminho para o certificado de sua autoridade certificadora, versão do TLS desejada, dentre outras variáveis de ambiente.
- O registro de *logs* do banco de dados não está ativado por padrão, então modificações feitas na base de dados não são registradas diretamente pelo banco de dados. Contudo, devido à execução do banco de dados via Docker, é possível visualizar os *logs* de requisições à base de dados pelo *container* Docker que executa a aplicação.
- As assinaturas de pacotes de dados de abastecimentos não são verificadas pelo servidor, apenas pelo aplicativo do consumidor, o que torna possível enviar assinaturas inválidas para o servidor caso a API seja acessada por outro canal de comunicação que não seja o aplicativo.

- As verificações de software dos dispositivos das bombas de combustível não são realizadas pelo servidor, apenas pelo aplicativo do fiscal, o que torna possível enviar verificações inválidas para o servidor caso a API seja acessada por outro canal de comunicação que não seja o aplicativo.
- Os resultados de eventos de fiscalização (como registros de eventos de manutenção e de alteração de parâmetros) não possuem formato especificado e portanto podem conter qualquer informação possível de ser enviada por texto, incluindo dados sensíveis ou maliciosos, caso a API seja acessada por outro canal de comunicação que não seja o aplicativo.

6 Considerações Finais

Neste relatório foi apresentada a modelagem do Sistema Medida Inteligente, incluindo a definição da arquitetura e requisitos, descrição de casos de uso e diagramas de sequência, tecnologias utilizadas e aspectos relevantes de segurança. Os documentos complementares “Descrição da API”, “Manual do Aplicativo do Consumidor” e “Manual do Aplicativo do Fiscal” trazem instruções para instalação e utilização da API e dos aplicativos.

No fim desta fase do projeto, o backend do sistema estava sendo migrado para a infraestrutura do INMETRO e se encontrava em fase de testes funcionais (apenas para garantir que estava rodando corretamente). Assim como, a instância do Verificador de Conformidades também estava sendo migrada. Os aplicativos do consumidor e do fiscal estavam sendo testados com as BMCs dos fabricantes e eram atualizados conforme necessidade, para garantir a interoperabilidade entre os fabricantes, o correto funcionamento e a integração dos outros serviços.



Universidade Federal de Santa Catarina
Laboratório de Segurança em Computação

Estudo de Usabilidade de Assinaturas Digitais em Objetos Metrológicos

Descrição da API

Versão 1.1

Florianópolis
2022

Sumário

1 Introdução	4
1.1 Objetivos	4
2 Servidor	5
2.1 Componentes e ferramentas	5
2.2 Configuração	5
2.3 Execução	8
3 Definições	9
3.1 Endereço base	9
3.2 Passagem de informações	9
3.3 Uso de parâmetros de entrada e saída	9
3.4 Códigos de erro	10
4 Descrição das rotas da API	11
4.1 Aplicativo Consumidor	11
4.1.1 GET /bmcs/localizacao	11
4.1.2 GET /bmcs/{serial}/estabelecimento	12
4.1.3 POST /abastecimentos	13
4.1.4 POST /manifestacao	16
4.2 Aplicativo Fiscal	19
4.2.1. POST /tokenSGI	19
84.2.2 POST /fiscalizacao	22
4.3 Rotas de consulta	29
4.3.1 GET /bmcs	30
4.3.2 GET /bmcs/{serial}	31
4.3.3 GET /estabelecimentos	34
4.3.4 GET /estabelecimentos/{id}	36
4.3.5 GET /certificados	38
4.3.6 GET /certificados/{serial}	39
4.3.7 GET /abastecimentos	41
4.3.8 GET /abastecimentos/{id}	43
4.3.9 GET /manifestacoes	46
4.3.10 GET /manifestacoes/{id}	48
4.3.11 GET /eventosFiscalizacao	49
4.3.12 GET /eventosFiscalizacao/{id}	51
5 Arquivos	58
5.1 Tabelas de referência	58
5.1 Endereços da cadeia certificadora	60
6 Considerações Finais	62

1 Introdução

Este relatório técnico apresenta a API (Application Programming Interface) criada para uso pelo sistema Medida Inteligente desenvolvido pelo Laboratório de Segurança em Computação (LabSEC) da Universidade Federal de Santa Catarina. A API do sistema é a forma de acessar os dados da aplicação contidos no servidor e se comunicar com ele via solicitações de acesso aos seus recursos.

1.1 Objetivos

Este documento tem como objetivo:

- a) Fornecer instruções para execução do servidor.
- b) Definir os formatos usados pela API e informações úteis sobre ela.
- c) Descrever as rotas disponíveis da API do sistema.

2 Servidor

Nesta seção são descritos os componentes e ferramentas utilizadas para a criação do servidor e instruções para sua execução.

2.1 Componentes e ferramentas

O servidor é composto pelo conjunto Aplicação Java e Banco de dados MySQL. A aplicação é escrita na versão 17 do Java com base nos *frameworks* Spring para injeção de dependências e Hibernate para o gerenciamento do banco de dados e usando as bibliotecas AuthO (para geração, validação e decodificação de tokens), BouncyCastle (para fins criptográficos) com Maven encarregado do gerenciamento das dependências e *Bucket4j* para controle e limitação de requisições.

2.2 Configuração

É possível executar a aplicação e o banco de dados na mesma máquina ou em máquinas separadas.

A máquina responsável pela execução da aplicação deve ter a plataforma de virtualização *docker* instalada.

No arquivo *my.cnf* residem pré-configuradas as opções para execução do banco de dados. As seguintes opções devem ser alteradas de acordo com as necessidades de conexão para o banco:

- `max_connections`: Número máximo de conexões simultâneas ao servidor
- `thread_cache_size`: Tamanho da cache para conexão ao banco de dados

No arquivo *.env* residem pré-configuradas várias opções para conexão com o banco de dados, localização de certificados e conexão com as APIs. As seguintes opções devem ser alteradas de acordo com o ambiente que o servidor se encontra:

- Banco de dados:

- `MYSQL_DB_HOST`: Nome ou endereço IP host do banco de dados
- `MYSQL_DB_PORT`: Porta do banco de dados
- `MYSQL_DB_NAME`: Nome da base de dados
- `MYSQL_DB_TEST_NAME`: Nome da base de dados para testes
- `MYSQL_DB_USER`: Nome do usuário do banco de dados
- `MYSQL_DB_PASSWORD`: Senha para o usuário do banco de dados

- Configurações gerais:

- TZ: *Time zone* para o servidor (Valor default: America/Sao_Paulo).
- SERVER_PORT: Porta no qual o servidor receberá requisições.

- **Chaves para SSL:**

- SSL_KEY_STORE_FILE: Caminho completo para o arquivo de certificado para autenticação do servidor no protocolo HTTPS. Este deve estar em formato PKCS12, extensão .p12.
- SSL_KEY_STORE_PASSWORD: Senha para o arquivo de certificado.

- **Fala.Br:**

- CONFIG_FALABR_URL: URL de envio de manifestações para o Fala.br
- CONFIG_FALABR_TOKEN: Token para envio de manifestações para o Fala.br

- **Gov.BR:**

- CONFIG_GOVBR_URL: URL base para login no Gov.BR.
- CONFIG_GOVBR_JWK_PROVIDER_URL: URL que provê o token de validação para o Gov.BR.
- CONFIG_GOVBR_TOKEN_ISS: Emissor para validação do token do Gov.BR
- CONFIG_GOVBR_TOKEN_AUD: Audiência para validação do token do Gov.BR
- CONFIG_GOVBR_REDIRECT_URL: URL para redirecionamento após login no Gov.BR
- CONFIG_GOVBR_CLIENT_ID: Client ID para login no Gov.BR
- CONFIG_GOVBR_SECRET: Senha para login no Gov.BR

- **SIG:**

- CONFIG_SIG_ACCESS_TOKEN_KEY: Chave simétrica para os access tokens do SIG.
- CONFIG_SIG_ACCESS_TOKEN_ISS: Emissor para os tokens de autenticação do SIG.
- CONFIG_SIG_RESPONSE_PUBLIC_KEY_FILE: Caminho para o arquivo com a chave pública para validação dos tokens de resposta do SIG.
- CONFIG_SIG_RESPONSE_TOKEN_ISS: Emissor para os tokens de resposta do SIG.
- CONFIG_SIG_REQUEST_PRIVATE_KEY_FILE: Chave pública para assinatura dos tokens enviados para o SIG.
- CONFIG_SIG_REQUEST_PUBLIC_KEY_FILE: Chave privada para assinatura dos tokens enviados para o SIG.
- CONFIG_SIG_URL_LOGIN: URL que efetua o login no SIG.
- CONFIG_SIG_URL_CONSULTA_ESTABELECIMENTO0: URL que efetua a consulta de estabelecimentos no SIG.
- CONFIG_SIG_URL_CONSULTA_BOMBA: URL que efetua a consulta de bombas no SIG.

- **Limitação de requisições:**

- **CONFIG_RATE_LIMIT_EXCEEDED_MESSAGE:** Mensagem exibida em caso de excesso de requisições para a API.
- **CONFIG_RATE_LIMIT_TOKENS_PER_REQUEST:** Quantos tokens de limite uma requisição consome do total disponível para o solicitante.
- **CONFIG_RATE_LIMIT_TOKENS_TIME_TO_REFILL_MINUTES:** Tempo para que o servidor renove totalmente os tokens de limite do solicitante.
- **CONFIG_RATE_LIMIT_TOKENS_LIMIT_IP:** Quantidade de requisições que um usuário não autenticado pode realizar sem que os tokens de limite sejam renovados.
- **CONFIG_RATE_LIMIT_TOKENS_LIMIT_GOVBR:** Quantidade de requisições que um usuário autenticado via GovBR pode realizar sem que os tokens de limite sejam renovados.
- **CONFIG_RATE_LIMIT_TOKENS_LIMIT_SGI:** Quantidade de requisições que um usuário autenticado via SGI pode realizar sem que os tokens de limite sejam renovados.
- **CONFIG_RATE_LIMIT_TOKENS_TIME_TO_REFILL_MINUTES:** Tempo em minutos até a recarga para a capacidade máxima do total disponível.

2.3 Execução

Após todos os arquivos e configurações devidamente apontados e definidos pode-se inicializar o contêiner da API.

A máquina hospedeira deverá possuir os pacotes docker e docker-compose devidamente instalados e configurados.

Para executar a aplicação a máquina usando o banco de dados containerizado:

```
docker-compose up db -d  
docker-compose up medida-inteligente-app
```

Para a execução usando um banco de dados externo:

```
docker-compose up medida-inteligente-app
```

3 Definições

Nesta seção são apresentadas as informações necessárias para utilização da API.

3.1 Endereço base

O endereço base da aplicação é <https://inmetro.labsec.ufsc.br/>. As requisições são feitas via mensagens HTTPS.

3.2 Passagem de informações

As respostas das requisições são passadas pelo corpo (body) da mensagem HTTPS no formato JSON (definido como `application/json` no cabeçalho da mensagem). Caso haja parâmetros de entrada passando pelo corpo da requisição, eles também devem ser informados usando o formato JSON.

Os parâmetros de entrada e campos de saída de cada rota serão definidos por meio de tabelas na seção 4, em que a cada linha será descrito o nome de um campo, o tipo do campo, e uma descrição breve do significado do valor do campo.

Rotas GET retornando corretamente irão retornar com status HTTP 200 (OK), enquanto as rotas POST retornam como 201 (CREATED), logo as tabelas de campos de saída se referem a esses status em específico.

3.3 Uso de parâmetros de entrada e saída

Os parâmetros de entrada serão divididos em parâmetros de URL, parâmetros de cabeçalho e parâmetros de corpo da requisição.

Nos parâmetros de URL, filtros de consulta são passados como *query parameters*, identificados por “(query)” nas tabelas de parâmetros de entrada. Caso um recurso específico seja identificado unicamente pela URL são usados *path parameters*, identificados por “(path)” nas tabelas de parâmetros de entrada.

Com exceção dos filtros de consulta, todos os parâmetros de entrada descritos são obrigatórios.

Os dados necessários para solicitações POST são passados como parâmetros de corpo enquanto tokens de acesso (*access token*) são passados pelo cabeçalho (*header*) da mensagem.

3.4 Códigos de erro

Os códigos de erro previstos pela aplicação que poderão ser retornados como código HTTP são: 400 (Bad Request), no caso de envio de dados incorretos ou duplicados; 401 (Unauthorized), no caso de tokens de acesso inválidos, ou 404 (Not Found), no caso de recursos não encontrados pela URL.

4 Descrição das rotas da API

Nesta seção são descritas as rotas da API, divididas entre rotas para o aplicativo consumidor, rotas para o aplicativo fiscal e rotas de consulta geral.

4.1 Aplicativo Consumidor

As rotas a seguir são destinadas para uso do aplicativo consumidor, que tem como objetivo receber informações dos abastecimentos e manifestações dos usuários, e fornecer dados dos estabelecimentos e localizações das bombas de combustíveis.

4.1.1 GET /bmcs/localizacao

Essa rota tem como função a consulta da localização das bombas de combustível cadastradas no sistema. Não há parâmetros de entrada pois a rota não requer autenticação e não possui filtros de consulta.

- **Campos de saída:**

Campo	Tipo	Descrição
localizacoes:	<i>array</i>	Lista de coordenadas geográficas.
latitude	<i>decimal</i>	Latitude (eixo Y) da coordenada.
longitude	<i>decimal</i>	Longitude (eixo X) da coordenada.

- **Exemplo de JSON de saída:**

```
{
  "localizacoes":
  [
    {
      "latitude": -34.57428,
      "longitude": -34.57428
    },
    {
      "latitude": -34.57428,
      "longitude": -34.57428
    }
  ]
}
```

```
}  
  ]  
}
```

4.1.2 GET /bmcs/{serial}/estabelecimento

Essa rota tem como função a consulta do estabelecimento ao qual pertence uma determinada bomba de combustível, identificada pelo seu número serial. A rota não requer autenticação.

- **Parâmetros da URL:**

Parâmetro	Tipo	Descrição
serial	<i>string (path)</i>	Número serial da bomba de combustível.

- **Exemplo de URL:**

/bmcs/serial1/estabelecimento

- **Campos de saída:**

Campo	Tipo	Descrição
UF	<i>string</i>	Sigla da Unidade Federativa do estabelecimento.
endereco	<i>string</i>	Endereço completo do estabelecimento.
municipio	<i>string</i>	Município do estabelecimento.
CNPJ	<i>string</i>	CNPJ do estabelecimento.
razaoSocial	<i>string</i>	Razão Social do estabelecimento.

- **Exemplo de JSON de saída:**

```
{
```

```
"UF": "SC",
"endereco": "R. Eng. Agrônomo Andrei Cristian Ferreira,
88040-900",
"municipio": "Florianópolis",
"CNPJ": "83899526000180",
"razaoSocial": "Campus Universitário Reitor João David Ferreira
Lima"
}
```

4.1.3 POST /abastecimentos

Essa rota tem como função o envio das informações de um abastecimento para armazenamento no servidor.

- **Parâmetros do cabeçalho:**

Parâmetro	Tipo	Descrição
access_token	<i>string</i>	Access token JWT provido pelo GovBR, codificado.

- **Parâmetros do corpo:**

Parâmetro	Tipo	Descrição
serialBMC	<i>string</i>	Número serial da bomba de combustível que realizou o abastecimento.
numeroNaBMCIndicador	<i>int</i>	Número do dispositivo indicador relativo à bomba de combustível.
pacotesDadosAbastecimento:	<i>object</i>	Todos os pacotes de dados, identificados pelo número do dispositivo transdutor relativo à bomba de combustível como sendo a chave.

{númeroNaBMC}:	<i>object</i>	Número relativo do dispositivo transdutor que gravou o pacote de dados em relação à sua bomba de combustível.
timestamp	<i>string</i>	Data e hora do registro do pacote de dados, no formato "YYYY-MM-DD hh:mm:ss".
assinaturaDigital	<i>string</i>	Assinatura digital do pacote de dados codificada em base64.
controlador	<i>string</i>	Identificador único do dispositivo controlador que registrou o pacote.
idAbastecimentoControlador	<i>string</i>	ID do abastecimento, vindo da memória do dispositivo controlador.
idAbastecimentoTransdutor	<i>string</i>	ID do abastecimento, vindo da memória do dispositivo transdutor.
constanteCalibracao	<i>decimal</i>	Constante de calibração da bomba de combustível no abastecimento.
volumeTransdutor	<i>decimal</i>	Volume de combustível do abastecimento, medido pelo dispositivo transdutor.
volumeControlador	<i>decimal</i>	Volume de combustível do abastecimento, medido pelo dispositivo controlador.
valorMonetario	<i>decimal</i>	Valor monetário da transação do abastecimento.
precoLitro	<i>decimal</i>	Preço por litro do combustível.
idFisicoTransdutor	<i>string</i>	Identificação serial física única do dispositivo transdutor.

- **Exemplo de JSON de entrada:**

```
{
  "serialBMC": "1",
  "numeroNaBMCIndicador": 1,
  "pacotesDadosAbastecimento": {
    "1": {
      "timestamp": "2021-21-23 18:02:32",
      "assinaturaDigital": "uuLZEekrm64nbJMZR5A0Hqa6pCuJsF...",
      "controlador": "3",
      "idAbastecimentoControlador": "30",
      "idAbastecimentoTransdutor": "1",
      "constanteCalibracao": 7.4,
      "volumeTransdutor": 125.33,
      "volumeControlador": 125.33,
      "valorMonetario": 78.45,
      "precoLitro": 4.49,
      "idFisicoTransdutor": "id1"
    }
  }
}
```

- **Campos de saída:**

Campo	Tipo	Descrição
idAbastecimento	<i>int</i>	ID do abastecimento inserido.

- **Exemplo de JSON de saída:**

```
{
  "idAbastecimento": 1
}
```

4.1.4 POST /manifestacao

Essa rota tem como função o envio de uma manifestação referente a um abastecimento para o servidor e encaminhamento dela para a plataforma do Fala.BR. Ela retorna não só a identificação da manifestação feita dentro desta API, mas também dados vindos diretamente do encaminhamento da manifestação ao Fala.BR, como número de protocolo e código de acesso.

- **Parâmetros do cabeçalho:**

Parâmetro	Tipo	Descrição
access_token	<i>string</i>	Access token JWT provido pelo GovBR, codificado.

- **Parâmetros do corpo:**

Parâmetro	Tipo	Descrição
idToken	<i>string</i>	ID token JWT vindo do login no GovBR, codificado em base64.
manifestacao:	<i>object</i>	Dados da manifestação.
idAssunto	<i>int</i>	Identificação do assunto, cujos valores estão descritos abaixo.
tipoManifestacao	<i>int</i>	Identificador do tipo de manifestação, cujos valores estão descritos abaixo.
textoManifestacao	<i>string</i>	Descrição textual da manifestação.
emailManifestante	<i>string</i>	E-mail do manifestante.
idAbastecimento	<i>int</i>	Identificador interno do abastecimento.

- **Parâmetro idAssunto:**

Valor	Descrição
-------	-----------

1	Preço por litro anunciado (poster) difere do preço no dispositivo indicador.
2	Etanol fora de especificação (cor ou densidade).
3	Veículo apresenta falhas após abastecimento.
4	Valor da nota fiscal difere da bomba.
5	Recusa da emissão da Nota Fiscal.
6	Mau estado de conservação da bomba.
7	Selagem (lacre) rompida.
8	Marca de verificação (selo do Inmetro) vencida ou ausente.
9	Mau funcionamento do dispositivo indicador (exemplo: apagou).
10	Assinatura digital inválida.
11	Certificado digital inválido.
12	Data e/ou hora de medição incorreta(s).
13	Resultado da medição no app difere da bomba.

- **Parâmetro tipoManifestacao:**

Valor	Descrição
1	Reclamação.
2	Denúncia.

● **Exemplo de JSON de entrada:**

```
{
  "idToken": "eyJraWQiOiJyc2ExIiwiaWF0IjoiU1MyNTYifQ.ewogIC...",
  "manifestacao": {
    "idAssunto": 1,
```

```
"tipoManifestacao": 1,
"textoManifestacao": "Texto exemplo",
"emailManifestante": "exemplo@exemplo.com",
"idAbastecimento": 1
}
}
```

● **Campos de saída:**

Campo	Tipo	Descrição
IdManifestacaoFalaBR	<i>int</i>	ID da manifestação na API do Fala.BR.
CodigoAcesso	<i>string</i>	Código de acesso da manifestação na API do Fala.BR.
Links:	<i>array</i>	Lista de links relacionados à manifestação.
rel	<i>string</i>	Tipo de relacionamento do recurso com a entidade alvo do link.
href	<i>string</i>	URI do link.
NumeroProtocolo	<i>string</i>	Número de protocolo da manifestação na API do FalaBR.
IdManifestacaoMedidaInteligente	<i>int</i>	ID interno da manifestação nesta API.

● **Exemplo de JSON de saída:**

```
{
  "IdManifestacaoFalaBR": 49113,
  "CodigoAcesso": "dqfn2089",
  "Links": [
    {
      "rel": "self",
```

```
"href":  
"https://treinafalabr.cgu.gov.br/api/manifestacoes/49113"  
}  
],  
"NumeroProtocolo": "00106000060202268",  
"IdManifestacaoMedidaInteligente": 287  
}
```

4.2 Aplicativo Fiscal

As rotas a seguir são destinadas para uso do aplicativo fiscal, que tem como função receber dados de fiscalização das bombas de combustível que serão armazenados no servidor.

4.2.1. POST /tokenSGI

Essa é a rota de login para o usuário fiscal, que retorna um token de acesso JWT vindo da API do SGI do Inmetro.

- **Parâmetros do corpo:**

Parâmetro	Tipo	Descrição
cpf	<i>string</i>	CPF do usuário.
senha	<i>string</i>	Senha do usuário.
idUF	<i>int</i>	ID numérico da unidade federativa do usuário, conforme valores descritos abaixo.

- **Parâmetro idUF:**

Valor	Descrição
2	Rio Grande do Sul (RS).

3	Santa Catarina (SC).
13	São Paulo (SP).
14	Rio Grande do Norte (RN).
15	Paraná (PR).
16	Rio de Janeiro (RJ).
17	Tocantins (TO).
18	Acre (AC).
19	Pará (PA).
20	Amazonas (AM).
21	Roraima (RR).
22	Rondônia (RO).
23	Amapá (AP)
24	Goiás (GO) ou Distrito Federal (DF).
25	Mato Grosso (MT).
26	Sergipe (SE).
27	Pernambuco (PE).
28	Alagoas (AL).
29	Piauí (PI).
30	Bahia (BA).
31	Ceará (CE).
33	Espírito Santo (ES).
34	Maranhão (MA).
35	Minas Gerais (MG).


```
tWWlPakUyTmpNd01USTFNamtzSW1wMGFTSTZJbG95ZWxOWmJubzFSRmhNVlRSV1QzQWlMQ0p6ZFdJaU9q
SXpPU3dpY0hKmk1qb2lZemN4TmpJNVlUazRZVGcxTlRsaVpEVmxZekUzWm1Ga1lUUXpOMlk1TURZNFpqB
G1NakpqWlNjC0ltWjFibDlwWkNjNk1qTTVMQ0p6Y25aZmRXNW1YMmxrSWpWdWRXehNMQ0oxYzJWeVgzVn
VabDlwWkNjNk1qSWlMQ0prYVh0ZmFXUWlPakV5TlN3aWRlQmZiRzluYVc0aU9qRXNjblpsY250cGIyNWZ
ZMjlrWlNjNk9UazVmUS5JTtk1VF9xUWI0Y3BLV0h6dW1faWpEdS14WFFkSGpqcGQwUTZuWVdiTk9BIiwi
dG9rZW5fdHlwZSI6ImJlYXJlciIsImV4cGlyZXNfaW4i0jMwMDAwLCJuc19jcGYiOiIwMDAuMDAwLjAwM
S05MSIsIm5vX2Z1bmNpb25hcm1vIjoivXN1XHUwMGmxcmlvIERtbHJzIiwidXNlc191bmZfaWQiOiIyIi
widHBfdHJvY2Ffc2VuaGEi0jAsImFvX2NoYW5jZWxhIjoxLCJkc191cmwiOiJodHRweyUzQSUyRiUyRnB
sYXkuZ29vZ2x1LmNvbSUyRnN0b3JlJTJGYXBwcyUyRmRldGFpbHMlM0ZpZCUzRGNvbS5pbm1ldHJvLmdv
di5ici5hcHBpbm1ldHJvIiwidHBfYXR1YWxpemFjYW8i0jEsImRzX3ZlcnNpb25fbmFtZSI6IjAuMi45I
iwic2Vydm1jbyI6W3siY2Rfc2Vydm1jbyI6IjEiLCJkc19zZXJ2aWNvIjoiu2Vydm1cdTAwZTdvcyBkZS
BCQS9CTy9ESSJ9LHsiY2Rfc2Vydm1jbyI6IjEiLCJkc19zZXJ2aWNvIjoiu2Vydm1cdTAwZTdvcyBkZSB
GaXNjYXpFcdTAwZTdcdTAwZTNvIn0seyJjZf9zZXJ2aWNvIjoimTYiLCJkc19zZXJ2aWNvIjoiu2Vy
dm1cdTAwZTdvcyBkZSBGaXNjYXpFcdTAwZTdcdTAwZTNvIENyb25vIn0seyJjZf9zZXJ2aWNvIjoiu0
SIsImRzX3NlcnZpY28i0iJTZXJ2aVx1MDBlN29zIGRlIE1ldHJvbG9naWEifSx7ImNkX3NlcnZpY28i0i
IyMiIsImRzX3NlcnZpY28i0iJMZXZhbhRhbWVudG8gUGF0cm1tb25pYWwifV19.97LUiaxZEFy3T7c2tM
IxqMt_I48VTA2wiIBfAHAl2QmSlirqjXuELb7LrjnFouyH-HoPIZsn-aV0111kJjRwA"
}
```

84.2.2 POST /fiscalizacao

Essa rota permite ao fiscal enviar dados da fiscalização de uma bomba de combustível ao servidor, incluindo dados como registros de eventos de fiscalização, manutenção e interrupção do funcionamento da bomba, além de verificação de dispositivos, ensaios de abastecimentos, dentre outros.

- **Parâmetros do cabeçalho:**

Parâmetro	Tipo	Descrição
token_sgi	<i>string</i>	Token de acesso contido dentro do Token completo retornado pelo SGI.

- **Parâmetros do corpo:**

Parâmetro	Tipo	Descrição
-----------	------	-----------

access_token	<i>string</i>	Token JWT completo do fiscal, como recebido do SGI.
bmc:	<i>object</i>	Informações sobre a bomba medidora de combustível na qual foi feita a fiscalização.
serial	<i>string</i>	Número serial da bomba de combustível.
numeroInmetro	<i>string</i>	Número do Inmetro da bomba de combustível na API do SGI do Inmetro.
modelo	<i>string</i>	Modelo da bomba de combustível.
fabricante	<i>string</i>	Fabricante da bomba de combustível.
local:	<i>object</i>	Coordenadas da localização geográfica da bomba de combustível.
latitude	<i>double</i>	Latitude (eixo Y) da coordenada.
longitude	<i>double</i>	Longitude (eixo X) da coordenada.
indicadores:	<i>array</i>	Lista de dispositivos indicadores da bomba de combustível.
idRelativo	<i>int</i>	ID numérico do dispositivo indicador em relação à bomba de combustível.
versaoSoftware	<i>decimal</i>	Versão de software do dispositivo.
resultadoVerificacaoSoftware	<i>string</i>	Descrição do resultado da verificação de software neste dispositivo indicador.

transdutores:	<i>array</i>	Lista de dispositivos transdutores da bomba de combustível.
idRelativo	<i>int</i>	ID numérico do dispositivo transdutor em relação à bomba de combustível.
versaoSoftware	<i>decimal</i>	Versão de software do dispositivo.
resultadoVerificacaoSoftware	<i>string</i>	Descrição do resultado da verificação de software neste dispositivo transdutor.
certificado	<i>string</i>	Certificado do dispositivo transdutor, codificado em base64.
dadosFiscalizacao:	<i>object</i>	Dados da fiscalização.
alteracaoParametros	<i>string array</i>	Lista de descrições de resultados das alterações de parâmetros feitas na bomba de combustível.
cargasSoftware	<i>string array</i>	Lista de descrições de resultados das cargas de software feitas na bomba de combustível.
eventosInterrupcao	<i>string array</i>	Lista de descrições de resultados dos eventos de interrupção de funcionamento feitos na bomba de combustível.
eventosManutencao	<i>string array</i>	Lista de descrições de resultados dos eventos de manutenção feitos na bomba de combustível.
abastecimentos:	<i>array</i>	Lista de abastecimentos feitos na bomba de combustível pelo fiscal.
numeroNaBMCIndicador	<i>int</i>	Número do dispositivo indicador

		relativo à bomba de combustível.
pacotesDadosAbastecimento:	<i>object</i>	Todos os pacotes de dados, identificados pelo número do dispositivo transdutor relativo à bomba de combustível como sendo a chave.
{númeroNaBMC}:	<i>object</i>	Número relativo do dispositivo transdutor que gravou o pacote de dados em relação à sua bomba de combustível.
timestamp	<i>string</i>	Data e hora do registro do pacote de dados, no formato YYYY-MM-DD hh:mm:ss.
assinaturaDigital	<i>string</i>	Assinatura digital do pacote de dados codificada em base64.
controlador	<i>string</i>	Identificador único do dispositivo controlador que registrou o pacote.
idAbastecimentoControlador	<i>string</i>	ID do abastecimento, vindo da memória do dispositivo controlador.
idAbastecimentoTransdutor	<i>string</i>	ID do abastecimento, vindo da memória do dispositivo transdutor.
constanteCalibracao	<i>decimal</i>	Constante de calibração da bomba de combustível no abastecimento.
volumeTransdutor	<i>decimal</i>	Volume de combustível do abastecimento, medido pelo dispositivo transdutor.
volumeControlador	<i>decimal</i>	Volume de combustível do

	<i>al</i>	abastecimento, medido pelo dispositivo controlador.
valorMonetario	<i>decim al</i>	Valor monetário da transação do abastecimento.
precoLitro	<i>decim al</i>	Preço por litro do combustível.
idFisicoTransdutor	<i>string</i>	Identificação serial física única do dispositivo transdutor.

- **Exemplo de JSON de entrada:**

```
{
  "access_token": "eyJhbGciOiJIJFZlI1NiIsInR..",
  "bmc": {
    "serial": "12345678",
    "nr_inmetro": "6727393",
    "modelo": "modelo",
    "fabricante": "fabricante",
    "local": {
      "latitude": "34.57428",
      "longitude": "34.57428"
    },
    "indicadores": [
      {
        "idRelativo": 1,
        "versaoSoftware": 3.0,
        "resultadoVerificacaoSoftware": "descrição resultado"
      },
      {
        "idRelativo": 2,
        "versaoSoftware": 3.1,

```

```
        "resultadoVerificacaoSoftware": "descrição resultado 2"
    }
],
"transdutores": [
    {
        "idRelativo": 1,
        "versaoSoftware": 3.1,
        "resultadoVerificacaoSoftware": "descrição resultado 1",
        "certificado": "MIIDYjCCAuBAgIBATAFBgMrZXEwgYExC..."
    },
    {
        "idRelativo": 2,
        "versaoSoftware": 3.2,
        "resultadoVerificacaoSoftware": "descrição resultado 2",
        "certificado": "EIIDJkgkjTUDMKPLnYRDPkNF0KnbyuDv..."
    }
],
},
"dadosFiscalizacao": {
    "alteracaoParametros": [
        "array",
        "de",
        "strings"
    ],
    "cargasSoftware": [
        "array",
        "de",
        "strings"
    ],
    "eventosInterrupcao": [
        "array",
```

```
"de",
"strings"
],
"eventosManutencao": [
  "array",
  "de",
  "strings"
],
"abastecimentos": [
  {
    "serialBMC": "12345678",
    "numeroNaBMC": 1,
    "abastecimentoTeste": true,
    "pacotesDadosAbastecimento": {
      "250": {
        "timestamp": "2021-12-23 18:02:23",
        "assinaturaDigital": "uuLZEekrm64nbJMZR...",
        "controlador": "3",
        "idAbastecimentoControlador": "30",
        "idAbastecimentoTransdutor": "1",
        "constanteCalibracao": 3.4,
        "volumeTransdutor": 52.50,
        "volumeControlador": 52.50,
        "valorMonetario": 364.988,
        "precoLitro": 7.019,
        "idFisicoTransdutor": "id1"
      },
      "251": {
        "timestamp": "2021-12-23 18:02:23",
        "assinaturaDigital": "uuLZEekrm64nbJMZR...",
        "controlador": "3",
```

```
        "idAbastecimentoControlador": "30",
        "idAbastecimentoTransdutor": "1",
        "constanteCalibracao": 3.4,
        "volumeTransdutor": 52.50,
        "volumeControlador": 52.50,
        "valorMonetario": 364.988,
        "precoLitro": 7.019,
        "idFisicoTransdutor": "id2"
    }
}
]
```

- **Campos de saída:**

Campo	Tipo	Descrição
message	<i>string</i>	Descrição do resultado.

- **Exemplo de JSON de saída:**

```
{
  "message": "Fiscalização registrada"
}
```

4.3 Rotas de consulta

As rotas a seguir são destinadas à consulta dos dados armazenados no sistema, referentes aos envio realizados pelos aplicativos. A priori as rotas foram disponibilizadas apenas para visualização dos dados por agentes do sistema, portanto é necessária a autenticação pelo sistema do SGI do Inmetro.

4.3.1 GET /bmcs

Essa rota permite listar todas as bombas de combustível cadastradas, mas também é possível filtrá-las para visualizar todas as bombas de determinado modelo, fabricante e/ou estabelecimento.

- **Parâmetros do cabeçalho:**

Parâmetro	Tipo	Descrição
token_sgi	<i>string</i>	Token JWT vindo do SGI do Inmetro.

- **Parâmetros da URL:**

Parâmetro	Tipo	Descrição
modelo	<i>string (query)</i>	Modelo das bombas de combustível.
fabricante	<i>string (query)</i>	Fabricante das bombas de combustível.
estabelecimento	<i>int (query)</i>	ID do estabelecimento das bombas de combustível.

- **Exemplo de URL:**

/bmcs?fabricante=Fabricante2&estabelecimento=1

- **Campos de saída:**

Campo	Tipo	Descrição
serial	<i>string</i>	Número serial da bomba de combustível.
modelo	<i>string</i>	Modelo da bomba de combustível.
fabricante	<i>string</i>	Fabricante da bomba de combustível.
estabelecimento	<i>int</i>	ID do estabelecimento da bomba de combustível.

- **Exemplo de JSON de saída:**

```
[
  {
    "serial": "108991346",
    "estabelecimento": 1,
    "fabricante": "Fabricante2",
    "modelo": "Modelo1"
  },
  {
    "serial": "20749076",
    "estabelecimento": 1,
    "fabricante": "Fabricante2",
    "modelo": "Modelo2"
  }
]
```

4.3.2 GET /bmcs/{serial}

Essa rota permite a consulta dos dados de uma única bomba de combustível, identificada na URL pelo seu número serial.

- **Parâmetros do cabeçalho:**

Parâmetro	Tipo	Descrição
token_sgi	<i>string</i>	Token JWT vindo do SGI do Inmetro.

- **Parâmetros da URL:**

Parâmetro	Tipo	Descrição
serial	<i>string (path)</i>	Número serial das bombas de combustível.

- **Exemplo de URL:**

/bmcs/serial1

- **Campos de saída:**

Campo	Tipo	Descrição
modelo	<i>string</i>	Modelo da bomba de combustível.
fabricante	<i>string</i>	Fabricante da bomba de combustível
CNPJestabelecimento	<i>string</i>	CNPJ do estabelecimento da bomba de combustível.
razaoSocialEstabelecimento	<i>string</i>	Razão social do estabelecimento da bomba de combustível.
indicadores:	<i>array</i>	Lista de dispositivos indicadores da bomba de combustível.
numeroNaBmc	<i>int</i>	Número do dispositivo em relação à bomba de combustível.
versaoSoftware	<i>decimal</i>	Versão de software do dispositivo.
transdutores:	<i>array</i>	Lista de dispositivos transdutores da bomba de combustível.
numeroNaBmc	<i>int</i>	Número do dispositivo em relação à bomba de combustível.
versaoSoftware	<i>decimal</i>	Versão de software do dispositivo.

- **Exemplo de JSON de saída:**

```
{
  "razaoSocialEstabelecimento": "UFSC",
  "CNPJestabelecimento": "83899526000182",
  "fabricante": "Fabricante2",
  "modelo": "Modelo1"
```

```
"indicadores": [  
  {  
    "numeroNaBMC": 1,  
    "versaoSoftware": 0.1  
  },  
  {  
    "numeroNaBMC": 2,  
    "versaoSoftware": 0.1  
  },  
  {  
    "numeroNaBMC": 3,  
    "versaoSoftware": 0.1  
  }  
],  
"transdutores": [  
  {  
    "serialCertificado": "1",  
    "numeroBMC": 1,  
    "versaoSoftware": 0.1  
  },  
  {  
    "serialCertificado": "1",  
    "numeroBMC": 2,  
    "versaoSoftware": 0.1  
  }  
],  
}
```

4.3.3 GET /estabelecimentos

Essa rota permite listar todos os estabelecimentos cadastrados, mas também é possível filtrá-los para visualizar todos os estabelecimentos de determinada unidade federativa ou município, além de também ser possível procurar por CNPJ ou razão social.

- **Parâmetros do cabeçalho:**

Parâmetro	Tipo	Descrição
token_sgi	<i>string</i>	Token JWT vindo do SGI do Inmetro.

- **Parâmetros da URL:**

Parâmetro	Tipo	Descrição
razaoSocial	<i>string (query)</i>	Razão social do estabelecimento.
cnpj	<i>string (query)</i>	CNPJ do estabelecimento.
uf	<i>int (query)</i>	ID da unidade federativa dos estabelecimentos, conforme descrito abaixo.
municipio	<i>string(query)</i>	Município dos estabelecimentos.

- **Parâmetro uf:**

Valor	Descrição
2	Rio Grande do Sul (RS).
3	Santa Catarina (SC).
13	São Paulo (SP).
14	Rio Grande do Norte (RN).
15	Paraná (PR).
16	Rio de Janeiro (RJ).

17	Tocantins (TO).
18	Acre (AC).
19	Pará (PA).
20	Amazonas (AM).
21	Roraima (RR).
22	Rondônia (RO).
23	Amapá (AP)
24	Goiás (GO) ou Distrito Federal (DF).
25	Mato Grosso (MT).
26	Sergipe (SE).
27	Pernambuco (PE).
28	Alagoas (AL).
29	Piauí (PI).
30	Bahia (BA).
31	Ceará (CE).
33	Espírito Santo (ES).
34	Maranhão (MA).
35	Minas Gerais (MG).
36	Mato Grosso do Sul (MS).
37	Paraíba (PB).

- **Exemplo de URL:**

/estabelecimentos?uf=3

- **Campos de saída:**

Campo	Tipo	Descrição
id	<i>int</i>	ID do estabelecimento.
uf	<i>string</i>	Sigla da Unidade Federativa do estabelecimento.
razaoSocial	<i>string</i>	Razão social do estabelecimento.

- **Exemplo de JSON de saída:**

```
[
  {
    "uf": "SC",
    "id": 1,
    "razaoSocial": "UFSC"
  },
  {
    "uf": "SC",
    "id": 2,
    "razaoSocial": "UFSC2"
  }
]
```

4.3.4 GET /estabelecimentos/{id}

Essa rota permite a consulta dos dados de um único estabelecimento, identificado na URL pelo seu ID no banco de dados.

- **Parâmetros do cabeçalho:**

Parâmetro	Tipo	Descrição
token_sgi	<i>string</i>	Token JWT vindo do SGI do Inmetro.

- **Parâmetros da URL:**

Parâmetro	Tipo	Descrição
id	<i>int (path)</i>	ID do estabelecimento.

- **Exemplo de URL:**

/estabelecimentos/1

- **Campos de saída:**

Campo	Tipo	Descrição
id	<i>int</i>	ID do estabelecimento.
idSGI	<i>string</i>	ID do estabelecimento vindo da API do SGI do Inmetro.
cnpj	<i>string</i>	CNPJ do estabelecimento.
razaoSocial	<i>string</i>	Razão social do estabelecimento.
endereco	<i>string</i>	Endereço do estabelecimento.
municipio	<i>string</i>	Município do estabelecimento.
UF	<i>string</i>	Sigla da Unidade Federativa do estabelecimento.

- **Exemplo de JSON de saída:**

```
{
  "UF": "SC",
  "endereco": "CAMPUS UNIVERSITÁRIO REITOR JOÃO DAVID FERREIRA
LIMA 88.040-970",
  "municipio": "Florianópolis",
  "id": 1,
  "cnpj": "83899526000182",
  "idSGI": "popularBDParaInserirEstabelecimento0.7160281",
```

```
"razaoSocial": "UFSC"
}
```

4.3.5 GET /certificados

Essa rota permite listar todos os certificados, mas também é possível filtrá-los para visualizar todos os certificados de uma determinada bomba de combustível.

- **Parâmetros do cabeçalho:**

Parâmetro	Tipo	Descrição
token_sgi	<i>string</i>	Token JWT vindo do SGI do Inmetro.

- **Parâmetros da URL:**

Parâmetro	Tipo	Descrição
bmc	<i>string (query)</i>	Número serial da bomba de combustível à qual os certificados pertencem.

- **Exemplo de URL:**

```
/certificados?bmc=serial1
```

- **Campos de saída:**

Campo	Tipo	Descrição
serialCertificado	<i>string</i>	Número serial do certificado.
estabelecimento	<i>string</i>	ID do estabelecimento do certificado.
serialBMC	<i>string</i>	Número serial da bomba de combustível do certificado.

- **Exemplo de JSON de saída:**

```
[
  {
    "serialBMC": "10.8991346",
    "serialCertificado": "1",
    "estabelecimento": 1
  },
  {
    "serialBMC": "10.8991346",
    "serialCertificado": "1",
    "estabelecimento": 1
  },
  {
    "serialBMC": "20.749076",
    "serialCertificado": "2",
    "estabelecimento": 1
  }
]
```

4.3.6 GET /certificados/{serial}

Essa rota permite a consulta dos dados de um único certificado, identificado na URL pelo seu número serial.

- **Parâmetros do cabeçalho:**

Parâmetro	Tipo	Descrição
token_sgi	<i>string</i>	Token JWT vindo do SGI do Inmetro.

- **Parâmetros da URL:**

Parâmetro	Tipo	Descrição
serial	<i>string (path)</i>	Número serial do certificado.

- **Exemplo de URL:**

/certificados/1

- **Campos de saída:**

Campo	Tipo	Descrição
CNSubject	<i>string</i>	“Common name” (nome comum) da pessoa para a qual o certificado é destinado.
publicKey	<i>string</i>	Chave pública do certificado.
notBefore	<i>string</i>	Data de criação do certificado.
notAfter	<i>string</i>	Data de validade do certificado.
revogado	<i>boolean</i>	Indica se o certificado já foi revogado ou não.

- **Exemplo de JSON de saída:**

```
{
  "notAfter": "2025-08-26 20:19:07.0",
  "revogado": false,
  "CNSubject": "RTM 556/2016 Bomba Medidora de combustivel 1",
  "publicKey":
"#033A00ABFD AFBABE0BC1068ADEF75FD59D397A33D9C3133CCF68B7B33A36929B
67478E2B9F766F2A14DE578C953C7C98E7A1AFB34BC1E610C2F6F900",
  "transdutor": {
    "numeroBMC": 2,
    "BMC": "10.95565647",
    "id": 7
  },
  "notBefore": "2021-08-27 20:19:07.0"
}
```

4.3.7 GET /abastecimentos

Essa rota permite listar todos os abastecimentos, mas também é possível filtrá-los para visualizar os abastecimentos de certo estabelecimento ou bomba de combustível, além de considerar se são abastecimentos de teste ou não e filtrar por período de tempo.

- **Parâmetros do cabeçalho:**

Parâmetro	Tipo	Descrição
token_sgi	<i>string</i>	Token JWT vindo do SGI do Inmetro.

- **Parâmetros da URL:**

Parâmetro	Tipo	Descrição
bmc	<i>string (query)</i>	Número serial da bomba de combustível que realizou os abastecimentos.
estabelecimento	<i>string (query)</i>	ID do estabelecimento em que foram realizados os abastecimentos.
ensaio	<i>boolean (query)</i>	Indica se os abastecimentos são de teste/ensaio ou não.
dataInicio	<i>string (query)</i>	Início do período de tempo em que se deseja filtrar os abastecimentos, no formato “YYYY-MM-DD hh:mm:ss”.
dataFim	<i>string (query)</i>	Final do período de tempo em que se deseja filtrar os abastecimentos, no formato “YYYY-MM-DD hh:mm:ss”.

- **Exemplo de URL:**

`/abastecimentos?ensaio=true&dataInicio=2019-02-02 22:00:00`

- **Campos de saída:**

Campo	Tipo	Descrição
id	<i>int</i>	ID do abastecimento.
estabelecimento	<i>int</i>	ID do estabelecimento em que foi realizado o abastecimento.
bmc	<i>string</i>	Número serial da bomba de combustível que realizou o abastecimento.
data	<i>string</i>	Data e hora do abastecimento, no formato "YYYY-MM-DD hh:mm:ss".

- **Exemplo de JSON de saída:**

```
[
  {
    "estabelecimento": 9,
    "bmc": "SERIAL_TESTE_224037008",
    "id": 11,
    "data": "2022-04-11 22:39:49.0"
  },
  {
    "estabelecimento": 19,
    "bmc": "SERIAL_TESTE_-2078497209",
    "id": 21,
    "data": "2022-04-11 22:43:56.0"
  },
  {
    "estabelecimento": 24,
    "bmc": "SERIAL_TESTE_-1772608230",
    "id": 26,
    "data": "2022-04-11 22:44:51.0"
  }
]
```

]

4.3.8 GET /abastecimentos/{id}

Essa rota permite a consulta dos dados de um único abastecimento, identificado na URL pelo seu ID.

- **Parâmetros do cabeçalho:**

Parâmetro	Tipo	Descrição
token_sgi	<i>string</i>	Token JWT vindo do SGI do Inmetro.

- **Parâmetros da URL:**

Parâmetro	Tipo	Descrição
id	<i>int (path)</i>	ID do abastecimento.

- **Exemplo de URL:**

/abastecimentos/26

- **Campos de saída:**

Campo	Tipo	Descrição
estabelecimento	<i>int</i>	ID do estabelecimento em que foi realizado o abastecimento.
bmc	<i>string</i>	Número serial da bomba de combustível que realizou o abastecimento.
abastecimentoEnsaio	<i>boolean</i>	Indica se o abastecimento é de ensaio ou não.
data	<i>string</i>	Data e hora do abastecimento.

pacotesDadosAbastecimento:	<i>array</i>	Lista de pacotes de dados do abastecimento.
id	<i>int</i>	ID do pacote de dados.
data	<i>string</i>	Data e hora do registro do pacote de dados, no formato “YYYY-MM-DD hh:mm:ss”.
assinaturaDigital	<i>string</i>	Assinatura digital do pacote de dados codificada em base64.
controlador	<i>string</i>	Identificador único do dispositivo controlador que registrou o pacote.
idAbastecimentoControlador	<i>string</i>	ID do abastecimento, vindo da memória do dispositivo controlador.
idAbastecimentoTransdutor	<i>string</i>	ID do abastecimento, vindo da memória do dispositivo transdutor.
constanteCalibracao	<i>decimal</i>	Constante de calibração da bomba de combustível no abastecimento.
volumeTransdutor	<i>decimal</i>	Volume de combustível do abastecimento, medido pelo dispositivo transdutor.
volumeControlador	<i>decimal</i>	Volume de combustível do abastecimento, medido pelo dispositivo controlador.
valorMonetario	<i>decimal</i>	Valor monetário da transação do abastecimento.
idFisicoTransdutor	<i>string</i>	Identificação serial física única do dispositivo transdutor.
precoLitro	<i>decimal</i>	Preço por litro do combustível.

transdutor	<i>int</i>	ID do transdutor que registrou o pacote.
------------	------------	--

- **Exemplo de JSON de saída:**

```
{
  "abastecimentoEnsaio": true,
  "estabelecimento": 24,
  "bmc": "SERIAL_TESTE_-1772608230",
  "pacotesDadosAbastecimento": [
    {
      "precoLitro": 0.6366706432703896,
      "volumeControlador": 0.7856791613576795,
      "idAbastecimentoTransdutor":
" idAbastecimentoTransdutor_TESTE_1067417743",
      "controlador": "controlador_TESTE_-144964928",
      "idFisicoTransdutor": "1514746090",
      "transdutor": 2,
      "idAbastecimentoControlador":
" idAbastecimentoControlador_TESTE_-1931790266",
      "volumeTransdutor": 0.8246296729130912,
      "valorMonetario": 0.4797075455248516,
      "assinaturaDigital":
"uuLZEekrm64nbJMZR5A0Hqa6pCuJsFgTaiRj1oLZIN\\ /RHRDSP6hFq0GjqomDcqz
5p6DYJIGDISa9YdAcF5B0df0fvmi5DdZKgy05Gs5QovlY9i\\ /ryrWkbDVL1pvexIe
wII54f0DgckzqTf0rwC\\ /gk1DxSszoGB5edhYtCXIgojZARoGCLZZ8kDl70bL06kL
iF\\ /CDsA8EqugdHpf1WmkvtWlWfNopvqk1902Aygp80w9JVQ022FM6CAk4kQaK7pZ
262UzEttAtk\\ /+iZIPmjzaSdQAAqRpQQY4W\\ /a00M9On4ZFDiCn11T9V6MK\\ /+d
RBX9kCaTqVYvThoINpXvu9MKH",
      "id": 28,
      "constanteCalibracao": 0.7295677417522305,
      "data": "2022-04-11 22:44:51.0"
    }
  ]
}
```

```

    }
  ],
  "data": "2022-04-11 22:44:51.0"
}

```

4.3.9 GET /manifestacoes

Essa rota permite listar todas as manifestações cadastradas, mas também é possível filtrá-las para visualizar as manifestações referentes a um certo estabelecimento ou bomba de combustível, além de filtrar por período de tempo.

- **Parâmetros do cabeçalho:**

Parâmetro	Tipo	Descrição
token_sgi	<i>string</i>	Token JWT vindo do SGI do Inmetro.

- **Parâmetros da URL:**

Parâmetro	Tipo	Descrição
bmc	<i>string (query)</i>	Número serial da bomba de combustível à qual se referem as manifestações.
estabelecimento	<i>string (query)</i>	ID do estabelecimento ao qual se referem as manifestações.
dataInicio	<i>string (query)</i>	Início do período de tempo em que se deseja filtrar as manifestações, no formato "YYYY-MM-DD hh:mm:ss".
dataFim	<i>string (query)</i>	Final do período de tempo em que se deseja filtrar as manifestações, no formato "YYYY-MM-DD hh:mm:ss".

- **Exemplo de URL:**

/manifestacoes?estabelecimento=2&dataInicio=2019-02-02

22:00:00

- **Campos de saída:**

Campo	Tipo	Descrição
id	<i>int</i>	ID da manifestação.
estabelecimento	<i>int</i>	ID do estabelecimento ao qual se refere a manifestação.
bmc	<i>string</i>	Número serial da bomba de combustível à qual se refere a manifestação.
data	<i>string</i>	Data e hora da manifestação, no formato "YYYY-MM-DD hh:mm:ss".

- **Exemplo de JSON de saída:**

```
[
  {
    "estabelecimento": 9,
    "bmc": "SERIAL_TESTE_224037008",
    "id": 11,
    "data": "2022-04-11 22:39:49.0"
  },
  {
    "estabelecimento": 19,
    "bmc": "SERIAL_TESTE_-2078497209",
    "id": 21,
    "data": "2022-04-11 22:43:56.0"
  },
  {
    "estabelecimento": 24,
    "bmc": "SERIAL_TESTE_-1772608230",
    "id": 26,
    "data": "2022-04-11 22:44:51.0"
  }
]
```

```
}  
]
```

4.3.10 GET /manifestacoes/{id}

Essa rota permite a consulta dos dados de uma única manifestação, identificada na URL pelo seu ID.

- **Parâmetros do cabeçalho:**

Parâmetro	Tipo	Descrição
token_sgi	<i>string</i>	Token JWT vindo do SGI do Inmetro.

- **Parâmetros da URL:**

Parâmetro	Tipo	Descrição
id	<i>int (path)</i>	ID da manifestação.

- **Exemplo de URL:**

/manifestacoes/393

- **Campos de saída:**

Campo	Tipo	Descrição
abastecimento	<i>int</i>	ID do abastecimento ao qual se refere a manifestação.
data	<i>string</i>	Data e hora da realização da manifestação, no formato “YYYY-MM-DD hh:mm:ss”.
destinatario	<i>string</i>	Órgão de destino da manifestação.
tipo	<i>string</i>	Tipo de manifestação.

assunto	<i>string</i>	Assunto da manifestação.
descricao	<i>string</i>	Descrição textual da manifestação.
idFalaBR	<i>int</i>	ID da manifestação na API do Fala.BR.

- **Exemplo de JSON de saída:**

```
{
  "assunto": "Etanol fora de especificação (cor ou densidade).",
  "tipo": "Denúncia",
  "data": "2022-03-02 17:40:35.0",
  "idFalaBR": 49292,
  "abastecimento": 390,
  "destinatario": "Ouvidoria Inmetro",
  "descricao": "teste | Mensagem do sistema: Etanol fora de especificação (cor ou densidade)."}
}
```

4.3.11 GET /eventosFiscalizacao

Essa rota permite listar todos os eventos de fiscalização cadastrados pelos fiscais, mas também é possível filtrá-los para visualizar as fiscalizações referentes a um certo estabelecimento ou bomba de combustível, feitas por um determinado fiscal ou realizadas em certo período de tempo.

- **Parâmetros do cabeçalho:**

Parâmetro	Tipo	Descrição
token_sgi	<i>string</i>	Token JWT vindo do SGI do Inmetro.

- **Parâmetros da URL:**

Parâmetro	Tipo	Descrição
bmc	<i>string (query)</i>	Número serial da bomba de combustível em que foram feitas as fiscalizações.
estabelecimento	<i>string (query)</i>	ID do estabelecimento em que foram feitos os eventos de fiscalização.
fiscal	<i>string (query)</i>	CPF do fiscal que realizou as fiscalizações.
dataInicio	<i>string (query)</i>	Início do período de tempo em que se deseja filtrar as fiscalizações, no formato "YYYY-MM-DD hh:mm:ss".
dataFim	<i>string (query)</i>	Final do período de tempo em que se deseja filtrar as fiscalizações, no formato "YYYY-MM-DD hh:mm:ss".

- **Exemplo de URL:**

/eventosFiscalizacao?estabelecimento=2&dataInicio=2019-02-02
22:00:00

- **Campos de saída:**

Campo	Tipo	Descrição
id	<i>int</i>	ID do evento de fiscalização.
fiscalNome	<i>int</i>	Nome do fiscal que realizou e registrou o evento de fiscalização.
bmc	<i>string</i>	Número serial da bomba de combustível em que foi feita a fiscalização.
data	<i>string</i>	Data e hora do registro do evento de fiscalização, no formato "YYYY-MM-DD hh:mm:ss".

- **Exemplo de JSON de saída:**

[

```

{
  "estabelecimento": 9,
  "bmc": "SERIAL_TESTE_224037008",
  "id": 11,
  "data": "2022-04-11 22:39:49.0"
},
{
  "estabelecimento": 19,
  "bmc": "SERIAL_TESTE_-2078497209",
  "id": 21,
  "data": "2022-04-11 22:43:56.0"
},
{
  "estabelecimento": 24,
  "bmc": "SERIAL_TESTE_-1772608230",
  "id": 26,
  "data": "2022-04-11 22:44:51.0"
}
]

```

4.3.12 GET /eventosFiscalizacao/{id}

Essa rota permite a consulta dos dados de um único evento de fiscalização, identificado na URL pelo seu ID.

- **Parâmetros do cabeçalho:**

Parâmetro	Tipo	Descrição
token_sgi	<i>string</i>	Token JWT vindo do SGI do Inmetro.

- **Parâmetros da URL:**

Parâmetro	Tipo	Descrição
id	<i>int (path)</i>	ID do evento de fiscalização.

- **Exemplo de URL:**

/eventosFiscalizacao/1

- **Campos de saída:**

Campo	Tipo	Descrição
data	<i>string</i>	Data e hora do registro do evento de fiscalização, no formato “YYYY-MM-DD hh:mm:ss”.
bmc	<i>int</i>	Número serial da bomba de combustível em que foi realizada a fiscalização.
fiscal:	<i>object</i>	Dados do fiscal que realizou a fiscalização.
CPF	<i>string</i>	CPF do fiscal.
nome	<i>string</i>	Nome completo do fiscal.
unidadeFederativa	<i>int</i>	Unidade Federativa do fiscal.
alteracoesParametros:	<i>array</i>	Lista de alterações de parâmetros referentes ao evento de fiscalização.
id	<i>int</i>	ID da alteração de parâmetros.
registro	<i>string</i>	Descrição do resultado da alteração de parâmetros.
cargasSoftware:	<i>array</i>	Lista de cargas de software referentes ao evento de fiscalização.
id	<i>int</i>	ID da carga de software.
registro	<i>string</i>	Descrição do resultado da carga de

		software.
ensaiosBico:	<i>array</i>	Lista de ensaios de bico referentes ao evento de fiscalização.
id	<i>int</i>	ID do ensaio de bico.
abastecimento	<i>int</i>	ID do abastecimento em que foi feito o ensaio.
numeroInmetro	<i>string</i>	Número do Inmetro.
resultado	<i>string</i>	Descrição do resultado do ensaio.
registrosManutencao:	<i>array</i>	Lista de eventos de manutenção referentes ao evento de fiscalização.
id	<i>int</i>	ID do evento de manutenção.
registro	<i>string</i>	Descrição do resultado do evento de manutenção.
registrosInterrupcao:	<i>array</i>	Lista de eventos de interrupção referentes ao evento de fiscalização.
id	<i>int</i>	ID do evento de interrupção.
registro	<i>string</i>	Descrição do resultado do evento de interrupção.
verificacoesDispositivosIndicadores:	<i>array</i>	Lista de verificações de dispositivos indicadores referentes ao evento de fiscalização.
id	<i>int</i>	ID da verificação.
dispositivoIndicador	<i>int</i>	ID do dispositivo indicador verificado.
resultado	<i>string</i>	Descrição do resultado da verificação.
verificacoesDispositivosTransdutores:	<i>array</i>	Lista de verificações de dispositivos transdutores referentes ao evento de

		fiscalização.
id	<i>int</i>	ID da verificação.
dispositivoTransdutor	<i>int</i>	ID do dispositivo transdutor verificado.
resultado	<i>string</i>	Descrição do resultado da verificação.

- **Exemplo de JSON de saída:**

```
{
  "data": "2022-04-14 15:50:45.0",
  "alteracoesParametros": [
    {
      "id": 32,
      "registro": "array"
    },
    {
      "id": 33,
      "registro": "de"
    },
    {
      "id": 34,
      "registro": "strings"
    }
  ],
  "fiscal": {
    "unidadeFederativa": "RS",
    "CPF": "00000000181",
    "nome": "marco"
  },
  "cargasSoftware": [
    {
```

```
    "id": 35,
    "registro": "array"
  },
  {
    "id": 36,
    "registro": "de"
  },
  {
    "id": 37,
    "registro": "strings"
  }
],
"registrosManutencao": [
  {
    "id": 41,
    "registro": "array"
  },
  {
    "id": 42,
    "registro": "de"
  },
  {
    "id": 43,
    "registro": "strings"
  }
],
"verificacoesDispositivosIndicadores": [
  {
    "resultado": "resultadoVerificacaoSoftware",
    "dispositivoIndicador": 18,
    "id": 26
  }
]
```

```
    },
    {
      "resultado": "resultadoVerificacaoSoftware2",
      "dispositivoIndicador": 27,
      "id": 28
    },
    {
      "resultado": "resultadoVerificacaoSoftware",
      "dispositivoTransdutor": 20,
      "id": 29
    },
    {
      "resultado": "resultadoVerificacaoSoftware",
      "dispositivoTransdutor": 30,
      "id": 31
    }
  ],
  "verificacoesDispositivosTransdutores": [
    {
      "resultado": "resultadoVerificacaoSoftware",
      "dispositivoTransdutor": 18,
      "id": 26
    },
    {
      "resultado": "resultadoVerificacaoSoftware2",
      "dispositivoTransdutor": 27,
      "id": 28
    }
  ],
  "bmc": "SERIAL_TESTE_-2115716575",
  "ensaiosBico": [],
```

```
"registrosInterrupcao": [  
  {  
    "id": 38,  
    "registro": "array"  
  },  
  {  
    "id": 39,  
    "registro": "de"  
  },  
  {  
    "id": 40,  
    "registro": "strings"  
  }  
]  
}
```

5 Arquivos

Além da API, o servidor fornece alguns arquivos que são consultados pelo aplicativo do fiscal. Esses arquivos estão localizados em *src/main/resources/static*.

5.1 Tabelas de referência

O arquivo */referencia-fabricantes/tabelas-referencia.json* tem como função descrever as versões de software dos dispositivos indicadores e transdutores das bombas de combustível, detalhando a data de atualização de cada versão e qual é o valor correto do software instalado em cada intervalo de memória com o auxílio de arquivos .csv adicionais na pasta */referencia-fabricantes*.

Estruturalmente, ele é um arquivo JSON que tem como chaves a data da última atualização da referência como um todo e uma lista de indicadores e outra de transdutores, que descreve em detalhes qual a versão de software de cada dispositivo. Para cada dispositivo é descrito o seu fabricante, sua versão de software, a data da última atualização desse software e o nome do arquivo CSV em que está a referência dos valores corretos da memória de cada intervalo de memória do dispositivo, de acordo com o seu fabricante e versão atual de software.

No arquivo de *tabelas-referencia.json* a formatação deve manter o seguinte padrão:

```
{
  "ultima_atualizacao": data da última atualização do arquivo em Unix Timestamp,
  "indicadores": [
    {
      "fabricante": nome do fabricante como retornado pelo comando 03h,
      "versao_software": versão de software do dispositivo como retornado pelo comando 01h,
      "ultima_atualizacao": data da última atualização dos dados do dispositivo ou da tabela de referência em Unix Timestamp,
      "arquivo": nome do arquivo, incluindo a extensão .csv, com a tabela de referência do dispositivo
    }
  ],
  "transdutores": [
    {
      "fabricante": nome do fabricante como retornado pelo comando 03h,
      "versao_software": versão de software do dispositivo como retornado pelo comando 01h,
      "ultima_atualizacao": data da última atualização dos dados do dispositivo ou da tabela de referência em Unix Timestamp,
```

```
        "arquivo": nome do arquivo, incluindo a extensão .csv, com a tabela de
referência do dispositivo
    }
]
}
```

- **Exemplo de arquivo de tabela de referência de software:**

```
{
  "ultima_atualizacao": 1663761792,
  "indicadores": [
    {
      "fabricante": "wertco",
      "versao_software": "0B14",
      "ultima_atualizacao": 1663761792,
      "arquivo": "indicador_wertco_0B14.csv"
    }
  ],
  "transdutores": [
    {
      "fabricante": "gilbarco",
      "versao_software": "AC1232",
      "ultima_atualizacao": 1663761792,
      "arquivo": "transdutor_gilbarco_AC1232.csv"
    }
  ]
}
```

Nos arquivos com as tabelas de referência (extensão .csv) a formatação deve manter o seguinte padrão:

```
endereco_inicial,endereco_final,hash_memoria
8 bytes em hexadecimal,8 bytes em hexadecimal,n bytes em hexadecimal
```

- **Exemplo de arquivo de tabela de referência:**

```
endereco_inicial,endereco_final,hash_memoria
0000,0002,ca978112ca1bbdcafac231b39a23dc4da786eff8147c4e72b9807785afee48bb
```

5.1 Endereços da cadeia certificadora

No arquivo `/verificador/enderecos-icp.json`, são apontados os endereços dos certificados e LCRs da cadeia certificadora. Devem ser incluídos os arquivos para a AC-Raiz, AC-INMETRO e uma lista de ACs Finais. Esses endereços são utilizados para fazer os download dos arquivos no aplicativo do fiscal, para poder realizar a verificação dos certificados localmente.

No arquivo a formatação deve manter o seguinte padrão:

```
{
  "ac_raiz": {
    "cert": endereço do certificado da AC-Raiz,
    "lcr": endereço da LCR da AC-Raiz
  },
  "ac_inmetro": {
    "cert": endereço do certificado da AC-INMETRO,
    "lcr": endereço da LCR da AC-INMETRO
  },
  "acs_finais": [
    {
      "cert": endereço do certificado da AC-Final 1,
      "lcr": endereço da LCR da AC-Final 1
    },
    {
      "cert": endereço do certificado da AC-Final 2,
      "lcr": endereço da LCR da AC-Final 2
    }
  ]
}
```

- **Exemplo de arquivo de endereços da cadeia certificadora:**

```
{
  "ac_raiz": {
    "cert": "https://inmetro.labsec.ufsc.br/verificador/ombr/AC_RAIZ.pem",
    "lcr": "http://repositorio.labsec.ufsc.br/homologacao/raiz/raiz-v6/raiz-v6.crl"
  },
  "ac_inmetro": {
    "cert": "https://inmetro.labsec.ufsc.br/verificador/ombr/AC_INTERMEDIARIA.pem",
    "lcr":
"http://repositorio.labsec.ufsc.br/homologacao/intermediaria/inmetro/ac-inmetro.crl"
  },
  "acs_finais": [
    {
      "cert": "https://inmetro.labsec.ufsc.br/verificador/ombr/AC_FINAL.pem",
      "lcr":
"http://repositorio.labsec.ufsc.br/homologacao/final/labsec-inmetro/ac-labsec-inmetro.crl"
    }
  ]
}
```

```
]
}
```

6 Considerações Finais

O sistema apresentado foi construído como prova de conceito para o estudo de caso de uso das comunicações entre os objetos metrológicos e as demais entidades envolvidas contemplado no plano de trabalho. A prova de conceito inclui a implementação da base de dados de objetos metrológicos e registros metrológicos, assim como a API que permite popular e consultar o banco de dados.

Sendo assim, este relatório apresentou os componentes e ferramentas utilizadas para a criação do servidor e instruções para executá-lo. Também descreveu-se como utilizar a API e detalhou-se as rotas implementadas para o aplicativo do consumidor, para o aplicativo do fiscal e as rotas de consultas gerais.



Universidade Federal de Santa Catarina
Laboratório de Segurança em Computação

Estudo de Usabilidade de Assinaturas Digitais em Objetos Metrológicos

Manual do Aplicativo do Consumidor

Versão 1.1

Florianópolis
2022

1 Introdução	3
1.1 Objetivos	3
2 Dependências de software utilizadas	4
3 Execução do aplicativo	5
4 Utilização do aplicativo	6
4.1 Tutorial	6
4.2 Tela inicial	9
4.3 Seleção da bomba de combustíveis	10
4.4 Informações do abastecimento	12
4.5 Mais informações	14
4.6 Enviando dados para o Inmetro	17
4.7 Manifestações	18
5 Considerações Finais	21
Anexo A - Configuração de endpoints	22
Anexo B - Obtenção de logs	23

1 Introdução

Este manual apresenta as instruções necessárias para a instalação e execução do aplicativo do consumidor do sistema Medida Inteligente desenvolvido pelo Laboratório de Segurança em Computação (LabSEC) da Universidade Federal de Santa Catarina.

1.1 Objetivos

Este documento tem como objetivo apresentar instruções para:

- a) Instalação das dependências de software necessárias.
- b) Execução do aplicativo.
- c) Utilização do aplicativo.

2 Dependências de software utilizadas

Será utilizado o software de desenvolvimento Flutter para realizar a instalação do aplicativo.

Para executar o aplicativo é necessário:

- SDK do Flutter, versão 2.10.5 (siga as instruções para baixá-la e instalá-la [nesse site](#))
 - A versão utilizada do Android Studio é a “Arctic Fox | 2020.3.1 Patch 4”
 - A versão utilizada do NDK é a 23.1.7779620
 - (android studio > SDK manager > SDK tools)
 - A versão utilizada da API do Android é 30 (Android 11.0 (R))
 - (android studio > SDK manager > SDK platforms)
- Um celular Android com versão Android igual à 5.0 (Lollipop) ou superior.

Além disso, é importante garantir que o celular permita o modo de depuração por USB ao ser conectado ao computador. Instruções para habilitar esse modo podem ser encontradas [aqui](#).

Caso o Android Studio já esteja instalado no computador e ocorra algum problema relacionado a sua versão, recomenda-se reinstalar o Android Studio com a versão mais recente.

3 Execução do aplicativo

Após possuir todas as dependências necessárias no computador pode-se então executar o aplicativo. Para isso, recomenda-se utilizar o terminal do sistema operacional. Os passos a seguir mostram como executar o aplicativo:

1. Extraia o zip do aplicativo para algum diretório desejado.
2. Em seguida, abra um terminal e navegue até o diretório da pasta do aplicativo.
3. Antes de iniciar o aplicativo é necessário rodar alguns comandos para garantir que as dependências estejam instaladas. Para isso, rode o comando *flutter pub get*.
4. Após seu término, rode *flutter pub upgrade*.
5. Conecte o celular no computador por meio de um cabo USB.
6. Para iniciar o aplicativo, rode o seguinte comando:

```
flutter run > app_log-DDMMAAAA_HHMM.txt
```

Substitua o DDMMAAAA_HHMM pela data e hora do momento da execução. Isso irá permitir diferenciar os logs caso mais de uma execução seja realizada. E espere até que o aplicativo seja iniciado no celular.

7. Deve aparecer um aviso no celular para permitir a depuração. Caso você perca o aviso e não consiga aceitá-lo, repita o passo anterior.
8. Após o término da execução do aplicativo, os logs do terminal estarão salvos no arquivo *app_log-DDMMAAAA_HHMM.txt* dentro do diretório atual.

4 Utilização do aplicativo

Aqui são apresentadas instruções para a utilização do aplicativo.

4.1 Tutorial

Ao abrir o aplicativo pela primeira vez, telas de tutorial são mostradas (Figuras 1, 2 e 3). Basta seguir os passos e, após seu término, o aplicativo lhe levará para a tela inicial. Caso deseje pular o tutorial, basta clicar no texto *Pular*, localizado na parte superior direita da tela. Após o término do tutorial, clique em *Começar* para começar a utilizar o aplicativo.

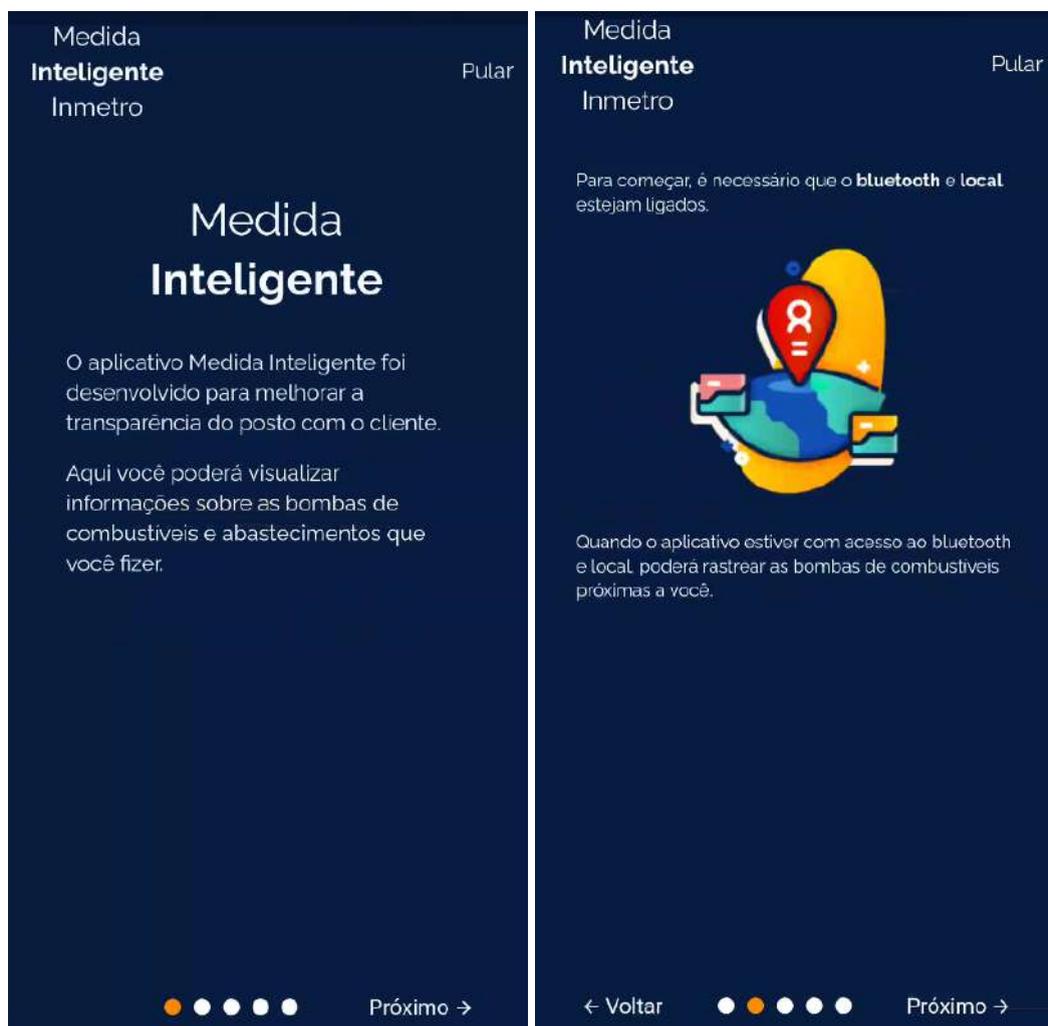


Figura 1: Telas de tutorial.

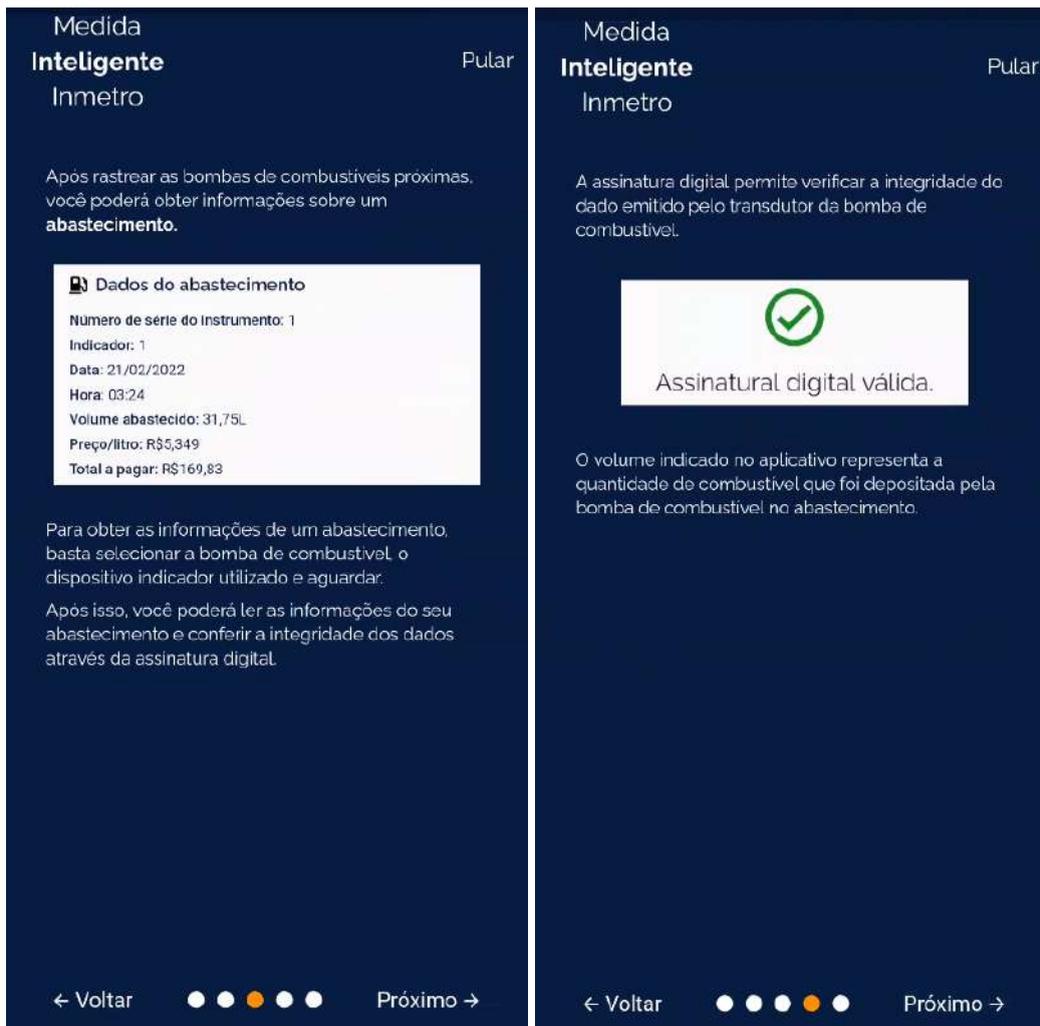


Figura 2: Telas de tutorial.

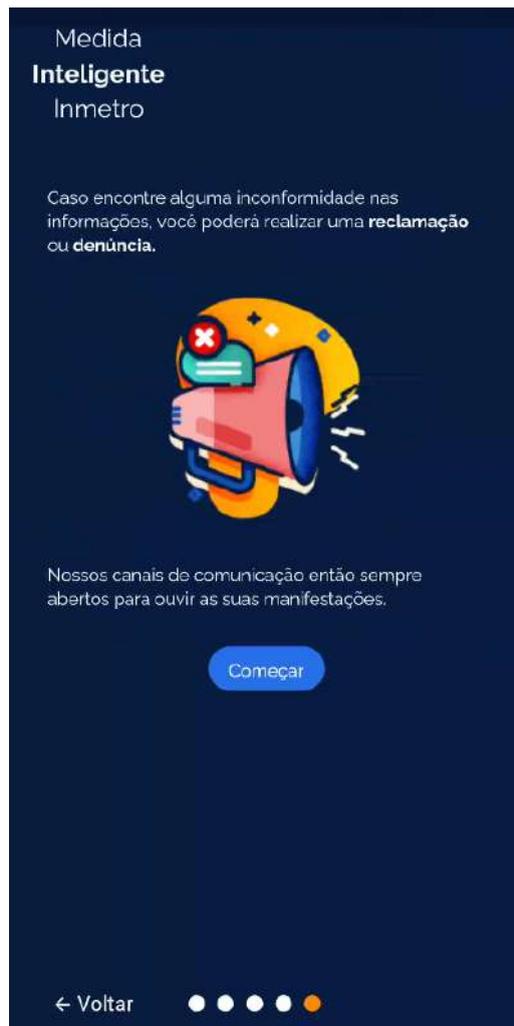


Figura 3: Telas de tutorial.

4.2 Tela inicial

Na tela inicial (Figura 4) encontram-se os botões para iniciar uma nova leitura, ver o mapa de bombas de combustíveis e enviar os dados dos abastecimentos. Também pode ser visto o histórico de leituras na parte inferior da tela, caso haja alguma.

A partir desta tela:

- Para fazer uma nova leitura, clique no botão *Nova Leitura*.
- Para enviar os dados dos abastecimentos clique no hiperlink *Deseja enviar dados para o INMETRO?*, onde será pedido o login Gov.br.
- Caso haja leituras passadas, é possível ver suas informações clicando em alguma delas.

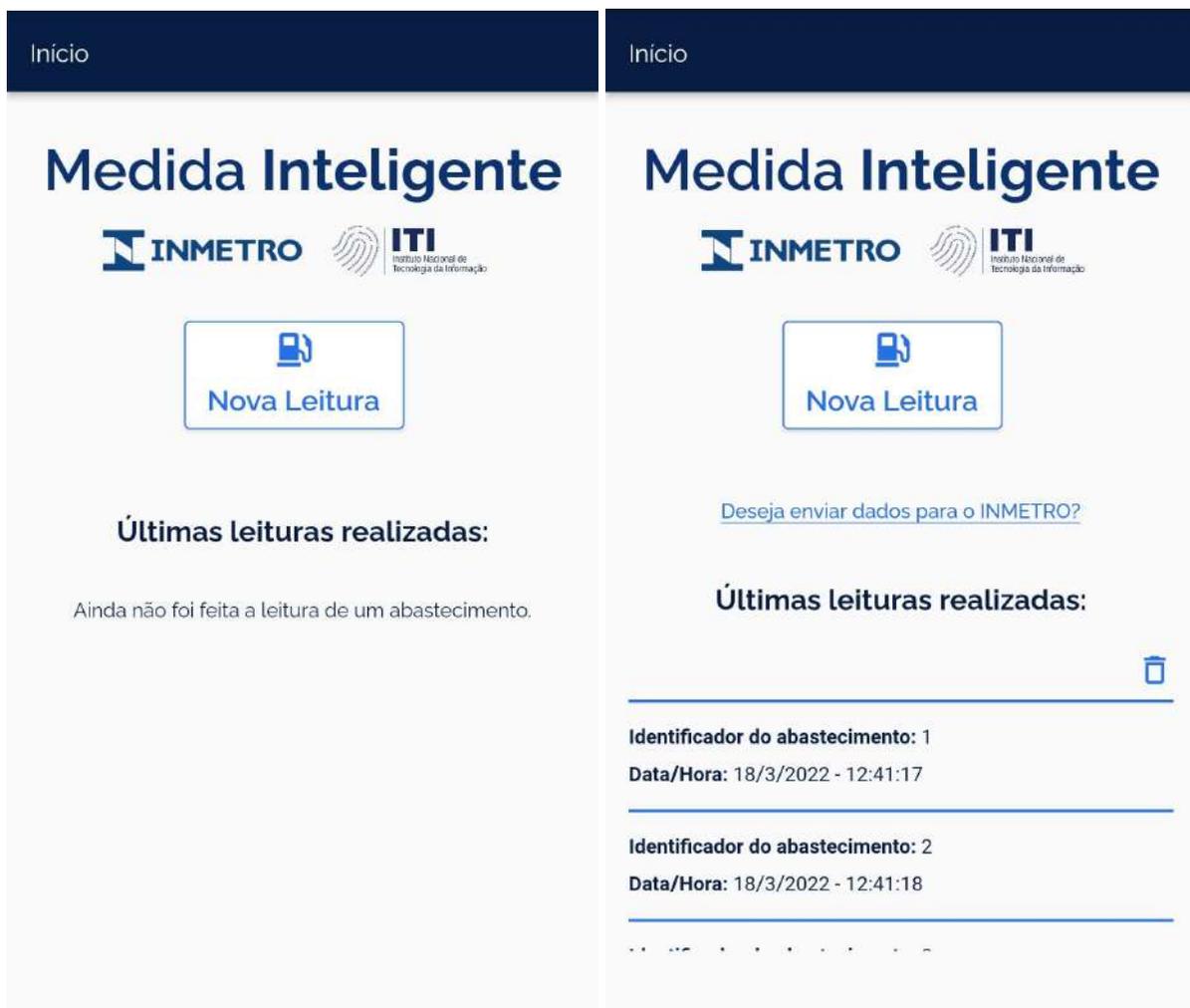


Figura 4: Tela inicial com histórico vazio e tela inicial com abastecimentos no histórico.

4.3 Seleção da bomba de combustíveis

Antes de entrar na tela de seleção, durante o carregamento, o aplicativo irá pedir permissão para acessar a localização e o bluetooth do celular (Figura 5). Escolha a opção de sua preferência. Caso o bluetooth ou localização não estejam ligados, o aplicativo irá mostrar uma tela pedindo para que o ligue.

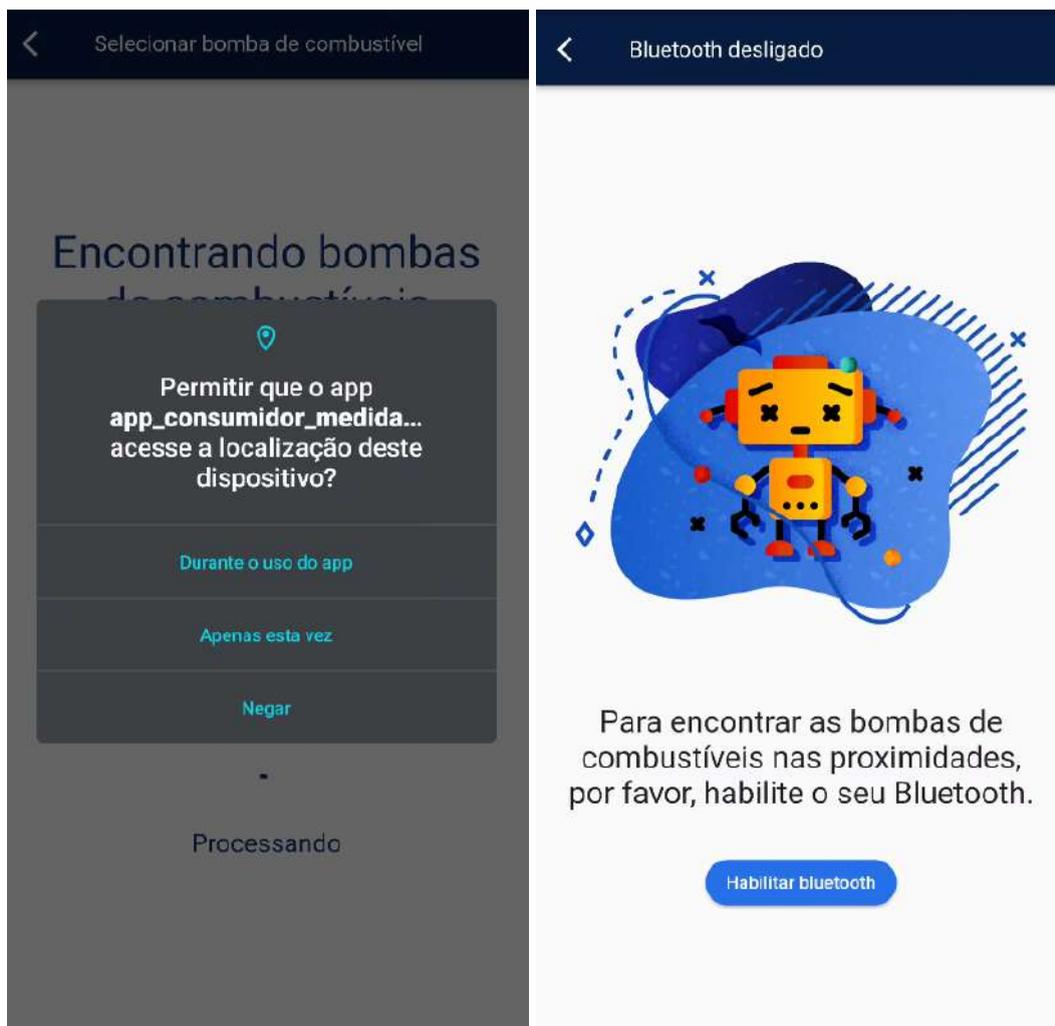


Figura 5: Permissão pedida pelo aplicativo para acessar a localização do celular (à esquerda) e tela indicando que o bluetooth está desligado (à direita)

Na tela de seleção da bomba de combustíveis (Figura 6) serão mostradas todas as bombas de combustíveis disponíveis para conexão, além de alguns botões utilitários:

- O botão 1 da figura realiza um novo escaneamento por dispositivos bluetooth.
- O botão 2 da figura, ao ser clicado para marcar ou desmarcar, também realiza um novo escaneamento por dispositivos bluetooth. Quando marcado, somente as

bombas de combustíveis são mostradas e, quando desmarcado, todos os dispositivos bluetooth detectados são mostrados.

- O campo 3 da figura mostra o local em que as bombas de combustíveis/dispositivos bluetooth são exibidos. Para selecionar algum deles, basta clicar em cima do dispositivo desejado
- Ao clicar no botão 4, será realizada a leitura dos dados do dispositivo selecionado, redirecionando o usuário à próxima tela.



Figura 6: Tela de seleção do dispositivo mostrando somente bombas de combustíveis.

4.4 Informações do abastecimento

Nesta tela (Figura 7) são mostradas as informações sobre o abastecimento, o resultado da verificação do certificado digital e dados relacionados à bomba de combustível, incluindo os dados do estabelecimento. Caso não haja conexão com a Internet ou a bomba ainda não esteja cadastrada no backend do sistema Medida Inteligente, a seção com os dados do estabelecimento será omitida.

A partir desta tela:

- Para repetir a verificação, clique no hiperlink com o texto *Repetir Verificação*. Caso a assinatura esteja válida, não é mostrada a opção de repetir a verificação.
- Para realizar uma manifestação relativa ao abastecimento, ou seja, relatar algum problema, clique no botão *Relatar Problema*.

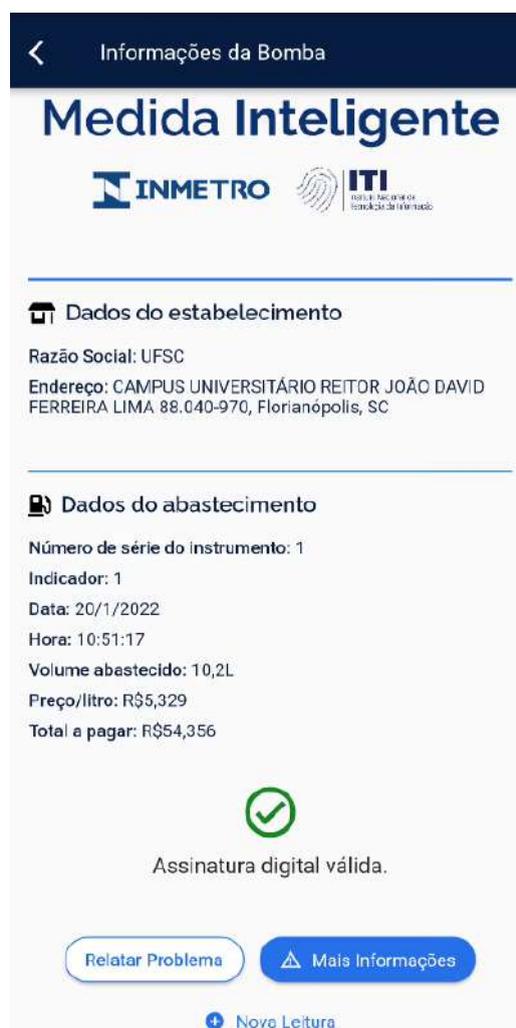


Figura 7: Tela de informações da bomba de combustíveis.

- Para obter mais informações da bomba de combustível e do abastecimento, clique no botão *Mais Informações*.
- Para realizar uma nova leitura, pode-se tanto clicar na seta de voltar no canto superior esquerdo da tela, quanto clicar no botão *Nova Leitura* na extremidade inferior da tela.

4.5 Mais informações

Nesta tela (Figura 8) são mostradas mais informações referentes à operação de abastecimento realizada pelo usuário.

A partir desta tela:

- O botão *Nova Leitura* realiza uma nova leitura.
- O botão *Ver manifestação* mostra o resumo da manifestação realizada pelo usuário.



Figura 8: Tela de informações da operação

No final das informações extras, há painéis expansíveis (Figura 9), que revelam mais informações sobre as assinaturas digitais dos transdutores ao serem clicados. Dependendo do tamanho da tela do celular, é necessário rolar a tela para baixo para que se possa ver todas as informações disponíveis sobre o transdutor selecionado.



Figura 9: Painéis expansíveis que revelam informações adicionais sobre os transdutores utilizados na operação de abastecimento.

Clicando em *Resposta do Verificador de Conformidades* ou em *Certificado*, uma janela modal irá se abrir (Figura 10), mostrando as informações correspondentes ao item selecionado.

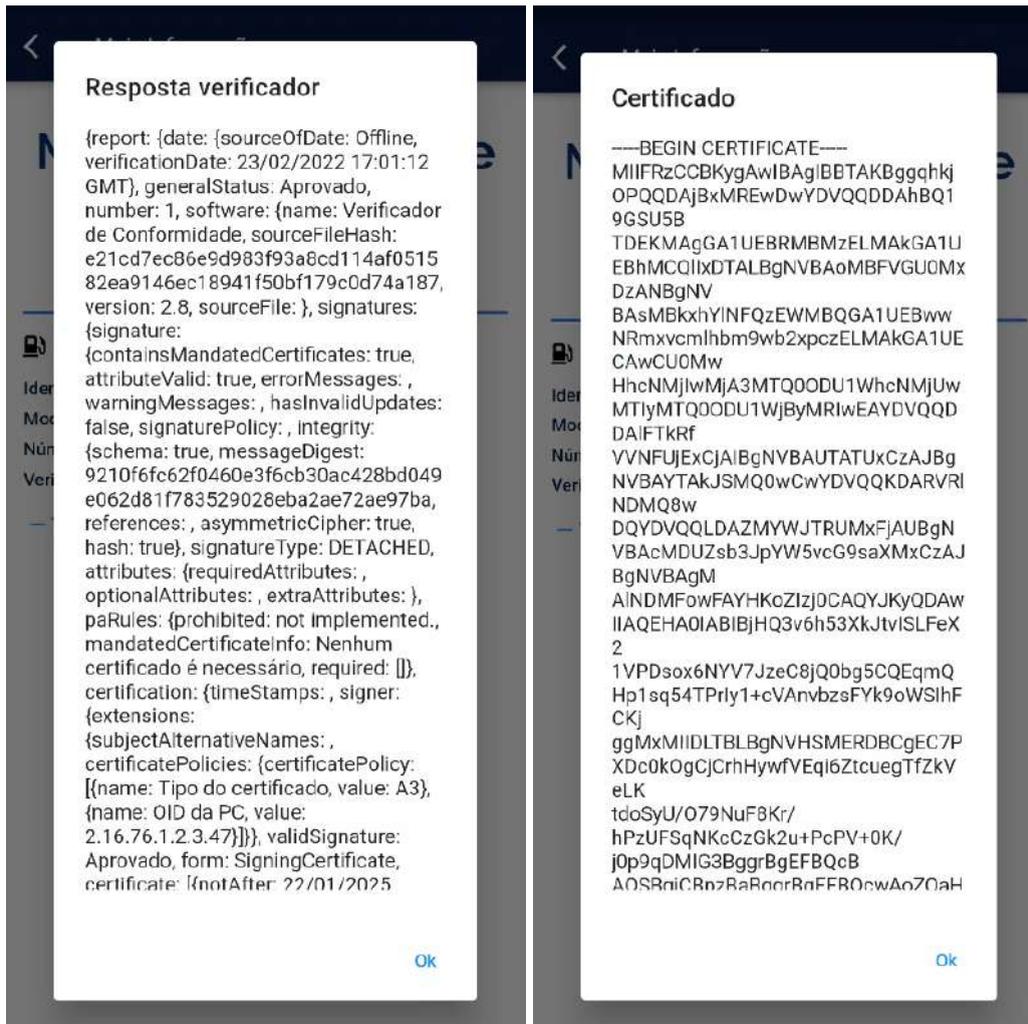


Figura 10: Telas modais abertas ao selecionar os itens *Abrir resposta do verificador* (à esquerda) e *Abrir certificado* (à direita).

4.6 Enviando dados para o Inmetro

Clicando no botão *Deseja enviar dados para o INMETRO* na tela inicial, uma nova janela modal irá se abrir (Figura 11), confirmando com o usuário se ele deseja enviar dados para o INMETRO.

- Clicando no botão *Confirmar*, uma nova página na web irá se abrir para que se possa realizar o login com o *Gov.br*.
- Após preencher os campos e realizar o login, o usuário será redirecionado ao aplicativo já logado em sua conta *Gov.br*.

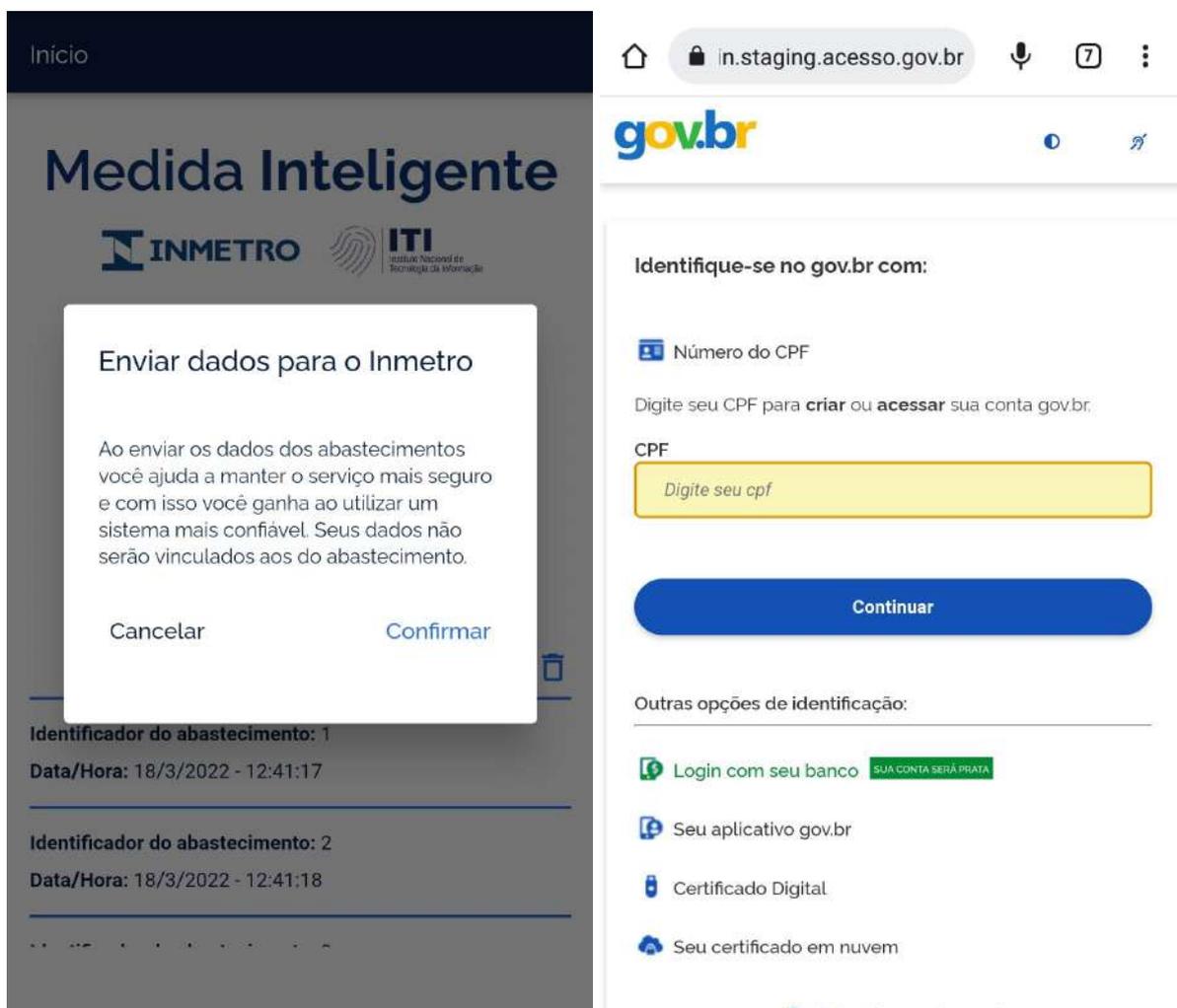


Figura 11: Janela modal aberta confirmando com o usuário se ele deseja enviar dados para o INMETRO (à esquerda) e página na web com o login do *Gov.br* (à direita).

4.7 Manifestações

Antes de ser redirecionado para a tela de manifestação, um modal de confirmação irá aparecer (Figura 12), perguntando se deseja-se relatar um problema, ato que pode ser realizado de maneira anônima ou não.

- Clicando em *Fazer manifestação anônima*, o usuário será redirecionado ao site do Fala.br.
- Clicando em *Confirmar*, caso o login Gov.br ainda não tenha sido realizado, o usuário será redirecionado ao site do gov.br para realizar o login (mesmo procedimento mostrado na Seção 4.6).
- Caso o abastecimento seja feito offline ou a bomba de combustível ainda não esteja cadastrada no backend do Sistema Medida Inteligente, o botão *Confirmar* abrirá diretamente o site do Fala.br.

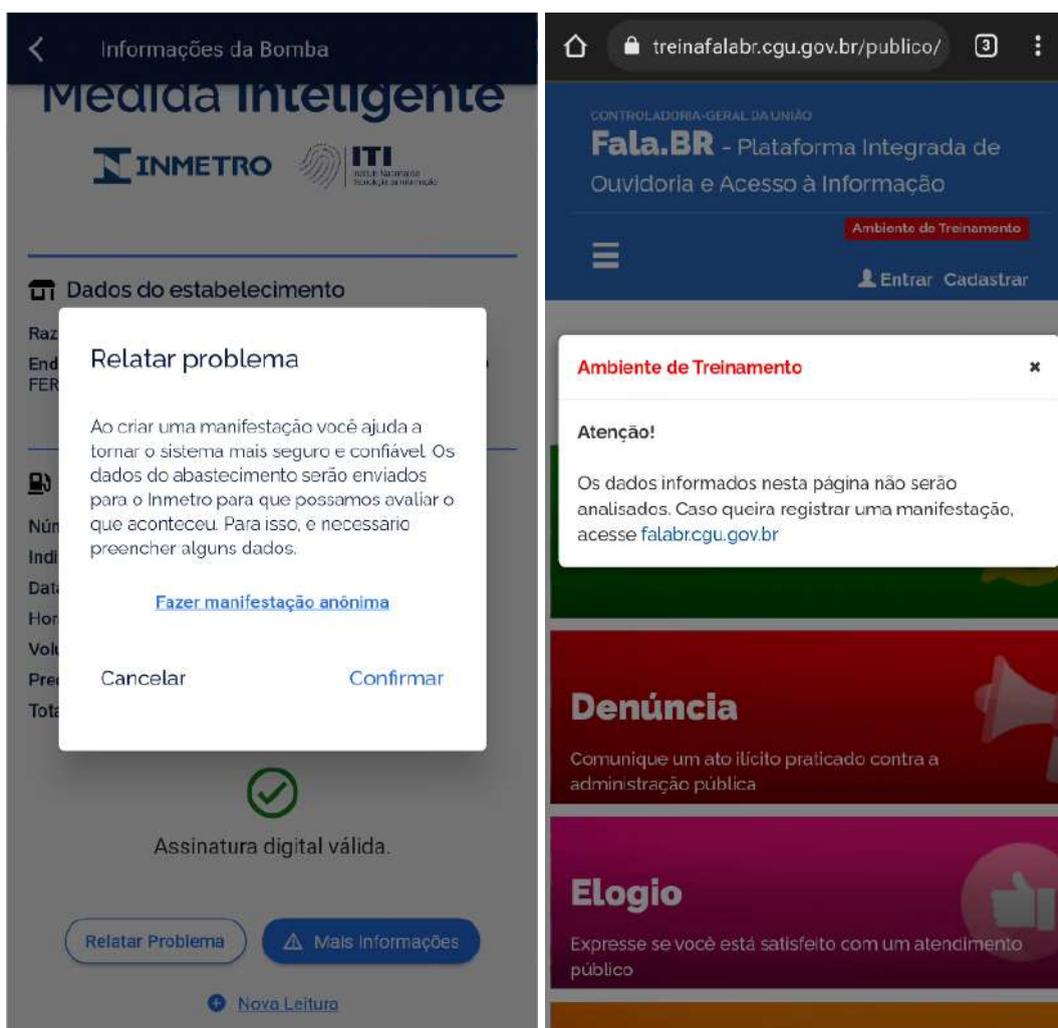


Figura 12: Modal de confirmação ao clicar em relatar problema (à esquerda) e site do Fala.br (à direita) em que o usuário é redirecionado ao clicar em *Fazer manifestação anônima*.

Após ter sido realizado o login do Gov.br, o usuário será redirecionado à tela de manifestação (Figura 13), a qual permitirá que o usuário relate um problema referente ao abastecimento realizado por meio do envio de uma manifestação.

Após ser finalizado o preenchimento da manifestação, o usuário poderá enviá-la clicando no botão *Enviar manifestação*, na parte inferior da tela. Ao clicar nesse botão, a manifestação é enviada para o servidor e encaminhada para o Fala.br. Juntamente com os dados da manifestação são enviados os dados do abastecimento associado, sendo que estes ficam armazenados no servidor. Com isso é possível posteriormente consultar os dados dos abastecimentos que tiveram manifestações realizadas.

The image displays two side-by-side screenshots of a mobile application form for creating a complaint. The left screenshot shows the main form with the title 'Medida Inteligente' and logos for INMETRO and ITI. The right screenshot shows the form with the title 'Manifestação' and a 'Resumo' section.

Left Screenshot (Form Creation):

- Header: Manifestação
- Title: Medida Inteligente
- Logos: INMETRO and ITI (Instituto Nacional de Tecnologia da Informação)
- Section: Faça sua manifestação
- Assunto: Seleccione o assunto (dropdown)
- Tipo de Manifestação: Selecionar tipo de manifestação (dropdown)
- Text: Descreva abaixo o conteúdo de sua manifestação. Seja claro e objetivo. Informações pessoais, inclusive identificação, não devem ser inseridas a não ser que sejam essenciais para a caracterização da manifestação.
- Text: Descreva sua manifestação aqui
- Text: Escreva aqui...
- Section: Dados do manifestante
- Nome: Meu Nome
- CPF: [Redacted]

Right Screenshot (Summary):

- Header: Manifestação
- Section: Resumo
- Assunto: Seleccione o assunto (dropdown)
- Tipo de Manifestação: Selecionar tipo de manifestação (dropdown)
- Text: Descreva abaixo o conteúdo de sua manifestação. Seja claro e objetivo. Informações pessoais, inclusive identificação, não devem ser inseridas a não ser que sejam essenciais para a caracterização da manifestação.
- Text: Descreva sua manifestação aqui
- Text: Escreva aqui...
- Section: Dados do manifestante
- Nome: Meu Nome
- CPF: [Redacted]
- Email: [Redacted]
- Button: Enviar Manifestação

Figura 13. Tela de criação de manifestação.

Após a manifestação ser enviada, o usuário é redirecionado à tela de resumo da manifestação (Figura 14). Nela, as informações referentes à manifestação realizada são mostradas. O botão *Voltar ao início* retorna para a tela inicial.



Figura 14. Tela de resumo da manifestação

5 Considerações Finais

Este relatório apresentou instruções para instalação e execução do Aplicativo do Consumidor do sistema Medida Inteligente. Além disso, também contém uma explicação básica das funcionalidades do aplicativo e de como utilizá-lo.

O aplicativo em questão foi construído como prova de conceito para os estudos de viabilidade da comunicação bluetooth com o objeto metrológico e da verificação de assinatura digital em dispositivos móveis. Contudo, o aplicativo também possui outras funcionalidades pensadas para o uso do consumidor, como o histórico de abastecimentos, a criação de manifestações, autenticação com Gov.br e envio de dados para o servidor

Anexo A - Configuração de endpoints

O aplicativo do consumidor realiza comunicação em rede com os seguintes serviços: API Medida Inteligente e Verificador de Conformidades. No caso de migração desses serviços entre ambientes de homologação e produção é necessário alterar os endpoints no código do aplicativo.

Para configurar o endpoint da API Medida Inteligente deve-se informar a URL na variável *baseUrl* do arquivo `lib\src\modules\api\url_config.dart`, como pode ser visto na Figura 15.

```
class UrlConfigAPI {
  static String baseUrl = "inmetro.labsec.ufsc.br";

  static String auth = "https://${baseUrl}/auth";

  static String bmcs = "https://${baseUrl}/bmcs/";
  static String estabelecimento = "/estabelecimento";

  static String manifestacao = "https://${baseUrl}/manifestacoes";
  static String abastecimentos = "https://${baseUrl}/abastecimentos";
}
```

Figura 15. Configuração da URL da API.

E para configurar o endpoint do Verificador de Conformidades deve-se informar a URL na variável *baseUrl* do arquivo `crypto-development\lib\src\url_config.dart`, como pode ser visto na Figura 16.

```
class UrlConfigVerificador {

  static String baseUrl = "inmetro.labsec.ufsc.br";
}
```

Figura 16. Configuração da URL do Gov BR.

Anexo B - Obtenção de logs

Para obtenção dos logs do aplicativo através do .apk, sem ter o código fonte, deve-se seguir os seguintes passos:

1. Instalar o Flutter versão 2.10.5 (siga as instruções para baixá-la e instalá-la [nesse site](#)).
2. Instalar o aplicativo no celular através do .apk.
3. Conecte o celular no computador por meio de um cabo USB.
4. Em seguida, abra um terminal e com o aplicativo aberto no celular rode o seguinte comando:

```
flutter logs > app_log-DDMMAAAA_HHMM.txt
```

Substitua o DDMMAAAA_HHMM pela data e hora do momento da execução. Isso irá permitir diferenciar os logs caso mais de uma execução seja realizada.

5. Execute a leitura de um novo abastecimento no aplicativo clicando em 'Nova Leitura'.
6. Após o término da execução do aplicativo, os logs do terminal estarão salvos no arquivo *app_log-DDMMAAAA_HHMM.txt* dentro do diretório atual.



Universidade Federal de Santa Catarina
Laboratório de Segurança em Computação

Estudo de Usabilidade de Assinaturas Digitais em Objetos Metrológicos

Manual do Aplicativo do Fiscal

Versão 1.1

Florianópolis
2022

1 Introdução	3
1.1 Objetivos	3
2 Dependências de software utilizadas	4
3 Execução do aplicativo	5
4 Utilização do aplicativo	6
4.1 Tela inicial	6
4.2 Seleção da bomba de combustível	10
4.3 Informações da Fiscalização	13
4.4 Informações da Bomba de Combustível	17
4.5 Informações do Estabelecimento	18
4.6 Informações da Operação: Dados da integridade de Software	19
4.5 Certificados Digitais	20
4.6 Abastecimentos de Ensaio	22
4.6.1 Informações do abastecimento	24
4.6.2 Mais informações	24
4.7 Registros	28
5 Considerações Finais	30
Anexo A - Configuração de endpoints	31
Anexo B - Obtenção de logs	32

1 Introdução

Este manual apresenta as instruções necessárias para a instalação e execução do aplicativo do fiscal do sistema Medida Inteligente desenvolvido pelo Laboratório de Segurança em Computação (LabSEC) da Universidade Federal de Santa Catarina.

1.1 Objetivos

Este documento tem como objetivo apresentar instruções para:

- a) Instalação das dependências de software necessárias.
- b) Execução do aplicativo.
- c) Utilização do aplicativo.

2 Dependências de software utilizadas

Será utilizado o software de desenvolvimento Flutter para realizar a instalação do aplicativo.

Para executar o aplicativo é necessário:

- SDK do Flutter, versão 2.10.5 (siga as instruções para baixá-la e instalá-la [nesse site](#))
 - A versão utilizada do Android Studio é a “Arctic Fox | 2020.3.1 Patch 4”
 - A versão utilizada do NDK é a 23.1.7779620
 - (android studio > SDK manager > SDK tools)
 - A versão utilizada da API do Android é 30 (Android 11.0 (R))
 - (android studio > SDK manager > SDK platforms)
- Um celular Android com versão Android igual à 5.0 (Lollipop) ou superior.

Além disso, é importante garantir que o celular permita o modo de depuração por USB ao ser conectado ao computador. Instruções para habilitar esse modo podem ser encontradas [aqui](#).

Caso o Android Studio já esteja instalado no computador e ocorra algum problema relacionado a sua versão, recomenda-se reinstalar o Android Studio com a versão mais recente.

3 Execução do aplicativo

Após possuir todas as dependências necessárias no computador pode-se então executar o aplicativo. Para isso, recomenda-se utilizar o terminal do sistema operacional. Os passos a seguir mostram como executar o aplicativo:

1. Extraia o zip do aplicativo para algum diretório desejado.
2. Em seguida, abra um terminal e navegue até o diretório da pasta do aplicativo.
3. Antes de iniciar o aplicativo é necessário rodar alguns comandos para garantir que as dependências estejam instaladas e atualizadas. Para isso, rode o comando *flutter pub get*.
4. Após seu término, rode *flutter pub upgrade*.
5. Conecte o celular no computador por meio de um cabo USB.
6. Para iniciar o aplicativo, rode o seguinte comando:

```
flutter run > app_log-DDMMAAAA_HHMM.txt
```

Substitua o DDMMAAAA_HHMM pela data e hora do momento da execução. Isso irá permitir diferenciar os logs caso mais de uma execução seja realizada. E espere até que o aplicativo seja iniciado no celular.

7. Deve aparecer um aviso no celular para permitir a depuração. Caso você perca o aviso e não consiga aceitá-lo, repita o passo anterior.
8. Após o término da execução do aplicativo, os logs do terminal estarão salvos no arquivo *app_log-DDMMAAAA_HHMM.txt* dentro do diretório atual.

4 Utilização do aplicativo

Aqui são apresentadas instruções para a utilização do aplicativo.

4.1 Tela inicial

Na tela inicial (Figura 1) encontram-se os botões para iniciar uma nova leitura e fiscalização de bomba de combustível, fazer login no SGI do Inmetro e enviar os dados das fiscalizações da bomba de combustível para o servidor. Também pode ser visto o histórico de leituras na parte inferior da tela, caso haja alguma.

A partir desta tela:

- Para fazer uma nova fiscalização, clique no botão *Nova Fiscalização*.
- Para fazer login no SGI do Inmetro, clique no hiperlink *Fazer login no SGI?*, onde será pedido os dados de login do fiscal. Após o login, o hiperlink dará lugar ao rótulo “Logado como: Nome do usuário”. Caso já existam leituras prévias registradas no celular, também aparecerá na tela o botão *Enviar fiscalização*.
- Caso haja leituras passadas, estas irão aparecer no histórico de leituras realizadas. É possível ver suas informações clicando em alguma delas. Para ver melhor as informações, é possível rolar a tela para baixo a partir da área abaixo do botão *Nova Fiscalização*.
- Para enviar leituras previamente realizadas a partir do histórico, selecione as leituras a serem enviadas, através das caixas de seleção que estão ao lado de cada leitura do histórico, e então clique no botão *Enviar fiscalização*. Caso ainda não tenha sido feito o login, será necessário fazê-lo neste momento.

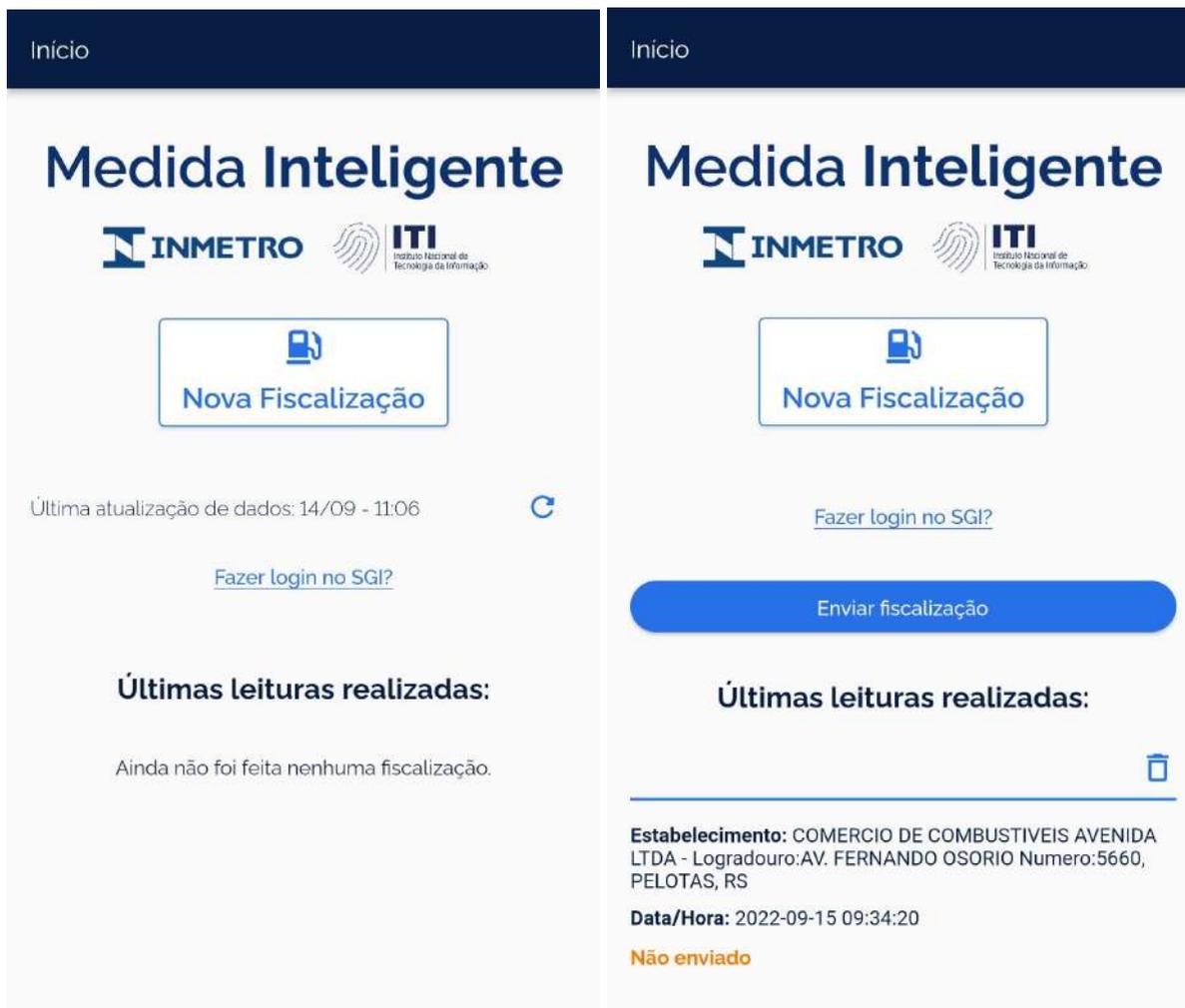


Figura 1: Tela inicial com histórico vazio e tela inicial com fiscalizações no histórico.

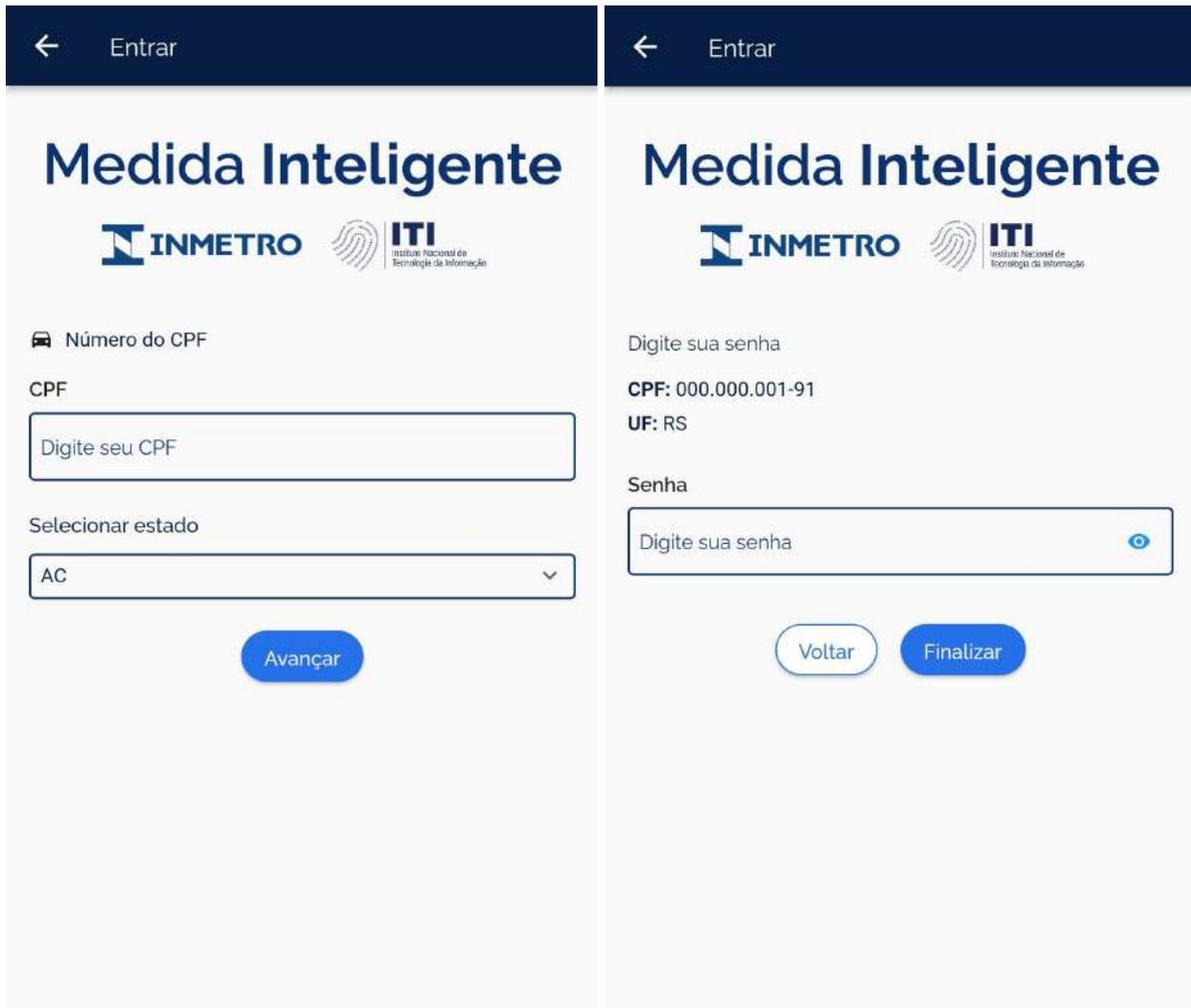


Figura 2: telas de login no SGI

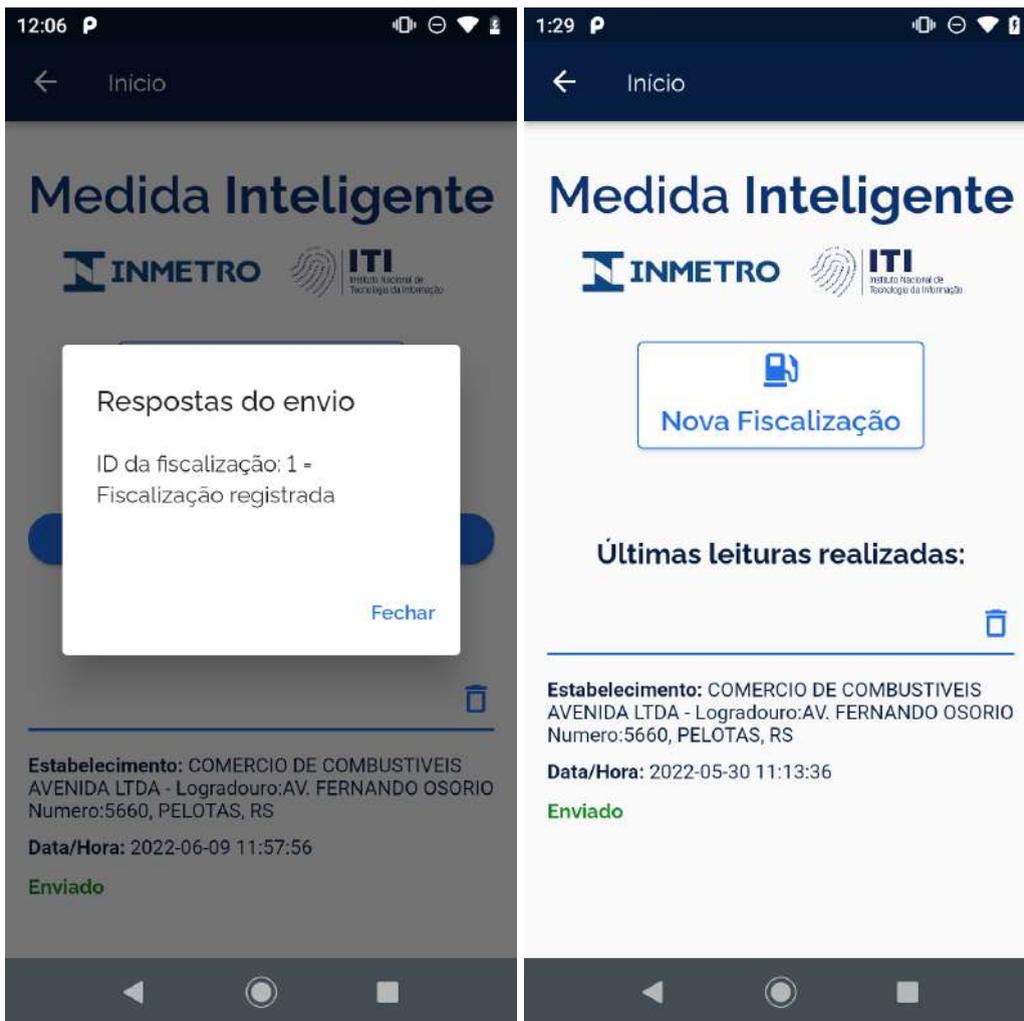


Figura 3: Tela inicial com janela modal confirmando o envio da fiscalização a partir do histórico (à esquerda), tela inicial com login já realizado e todas as leituras no histórico já enviadas (à direita).

4.2 Seleção da bomba de combustível

Ao clicar no botão *Nova Fiscalização*, o aplicativo irá pedir permissão para acessar a localização e o bluetooth do celular (Figura 4). Escolha a opção de sua preferência. Caso o bluetooth ou localização não estejam ativados, o aplicativo irá mostrar uma tela pedindo para que o ative.

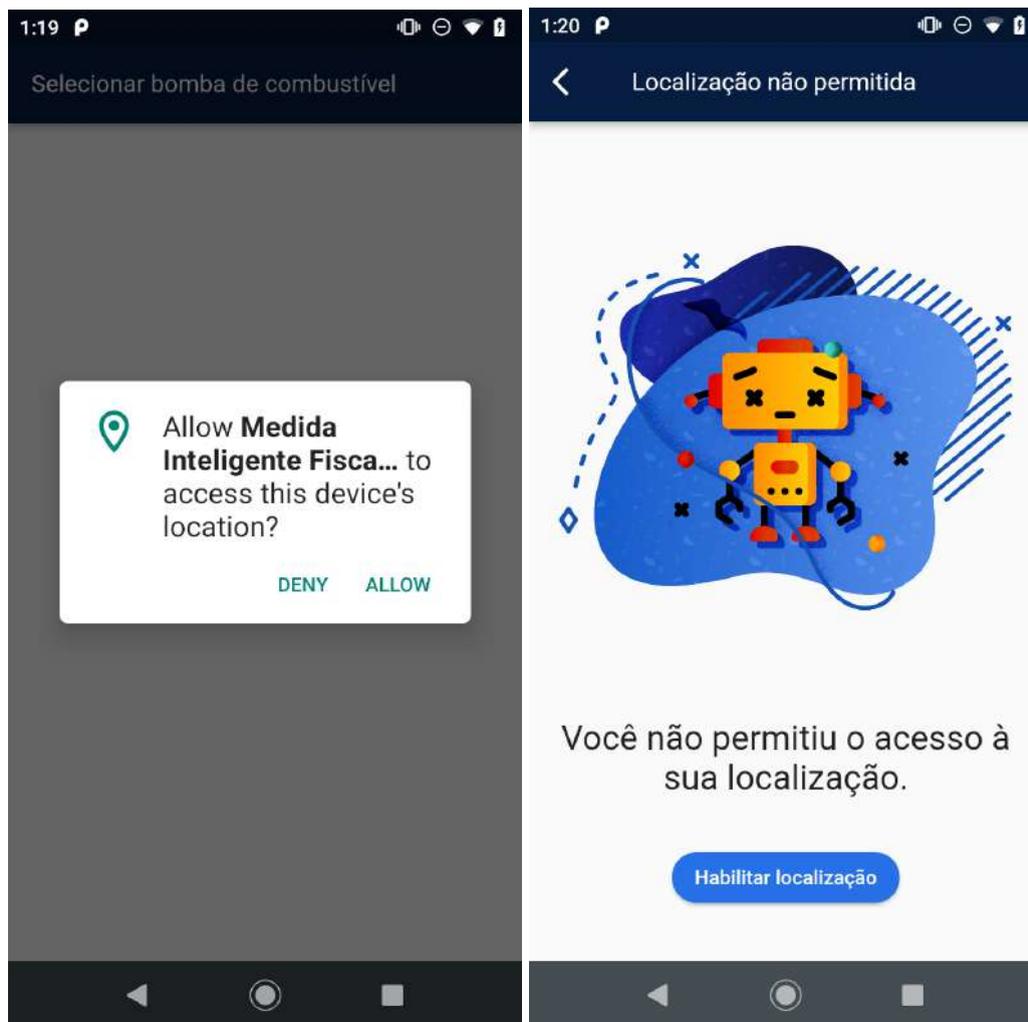


Figura 4: Permissão pedida pelo aplicativo para acessar a localização do celular (à esquerda) e tela indicando que o acesso à localização não foi permitido (à direita)



Figura 5: Tela indicando que o Bluetooth está desligado.

Após a habilitação da localização e bluetooth, será necessário realizar a seleção da bomba de combustível a qual deseja-se fazer a fiscalização. Na tela de seleção da bomba de combustível (Figura 6) serão mostradas todas as bombas de combustível disponíveis para conexão, além de alguns botões utilitários:

- O botão 1 da figura realiza um novo escaneamento por dispositivos bluetooth e atualiza a lista de bombas de combustíveis disponíveis.
- O botão 2 da figura, ao ser clicado para marcar ou desmarcar, também realiza um novo escaneamento por dispositivos bluetooth. Quando marcado, somente as bombas de combustíveis são mostradas e, quando desmarcado, todos os dispositivos bluetooth detectados são mostrados.
- O campo 3 da figura mostra o local em que as bombas de combustíveis/dispositivos bluetooth são exibidos. Para selecionar algum deles, basta clicar em cima do dispositivo desejado

- Ao clicar no botão 4, será realizada a leitura dos dados do dispositivo selecionado, redirecionando o usuário à próxima tela.



Figura 6: Tela de seleção do dispositivo mostrando somente bombas de combustíveis.

4.3 Informações da Fiscalização

Nesta tela (Figuras 7 e 8) são mostradas as informações sobre a bomba de combustível, dados do seu estabelecimento, o resultado da verificação de integridade de software dos dispositivos da bomba, o resultado da verificação dos certificados digitais da bomba, e outros dados relacionados à bomba de combustível, incluindo abastecimentos de ensaio e registros de fiscalização. Caso não haja conexão com a Internet, os dados do estabelecimento aparecerão como indisponíveis.

Logo após a obtenção dos dados, será requisitado ao usuário o número do Inmetro (Figura 7), necessário para buscar algumas informações da bomba de combustível do servidor. Ele pode ser informado digitando o código ou lendo o código de barras. Ao clicar em *Ok* o usuário poderá ver as informações da tela.

A partir desta tela:

- Para ver mais informações sobre a bomba de combustível, incluindo seu número de série, modelo e fabricante, clique no botão *Mais Informações* na seção “Bomba de combustível”.
- Para ver mais informações sobre o estabelecimento da bomba de combustível, clique no botão *Mais Informações* na seção “Estabelecimento”.
- Para ver detalhes da verificação da integridade de software dos dispositivos da bomba, clique no botão *Ver Detalhes* na seção “Integridade de Software”.
- Para ver detalhes dos certificados digitais da bomba de combustível, clique no botão *Ver Detalhes* na seção “Certificados digitais”.
- Para realizar abastecimentos de ensaio, clique no botão *Verificar* na seção “Abastecimentos de Ensaio”.
- Para ver os registros de fiscalização da bomba de combustível, clique no botão *Ver registros* na seção “Registros”.
- Para enviar os dados da leitura da bomba de combustível e sua fiscalização para o servidor, clique no botão *Reportar Leitura*. Caso o usuário ainda não esteja logado, esse botão o redireciona para a tela de login. Se o envio for bem sucedido, será aberta uma janela modal (Figura 8) que confirma o registro da fiscalização no servidor. Após o envio bem sucedido, o usuário é redirecionado para a tela inicial.

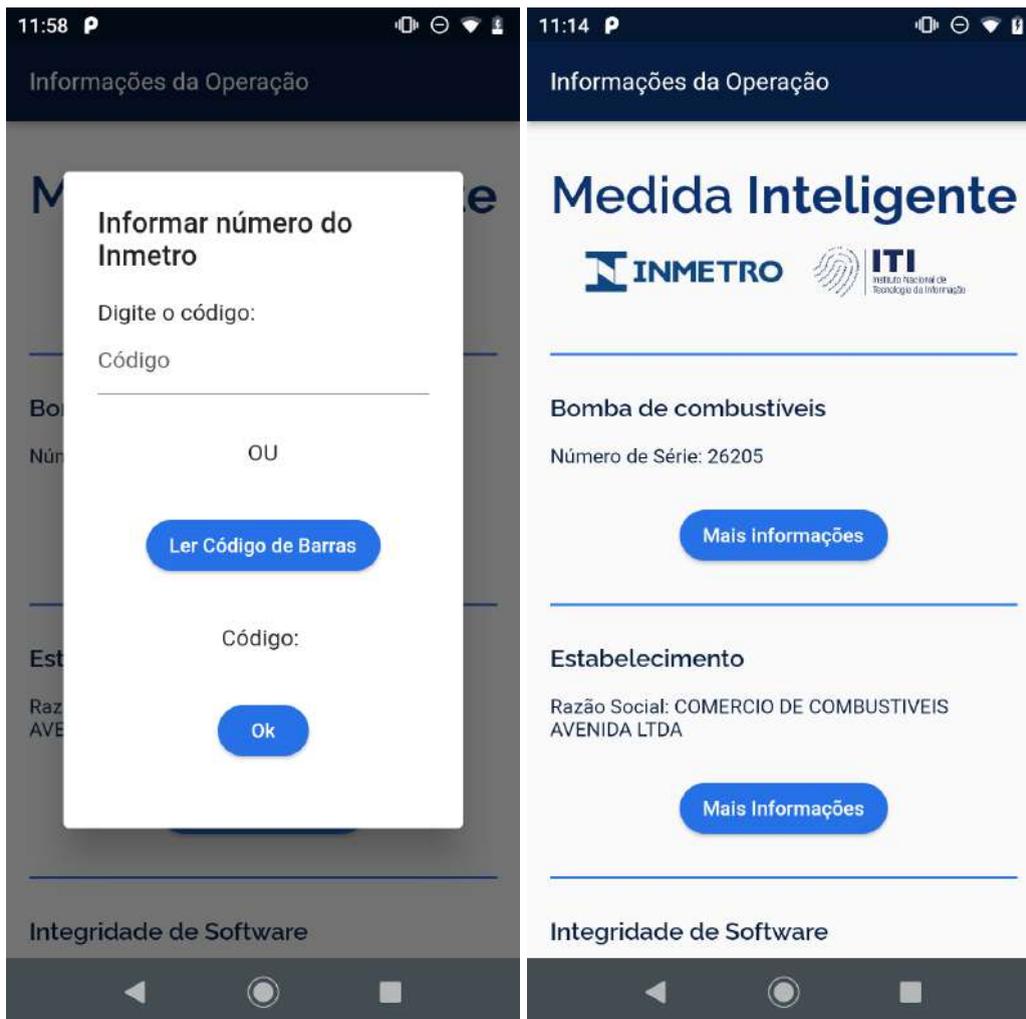


Figura 7: Requisição do número do Inmetro e começo da tela de informações da bomba de combustível.

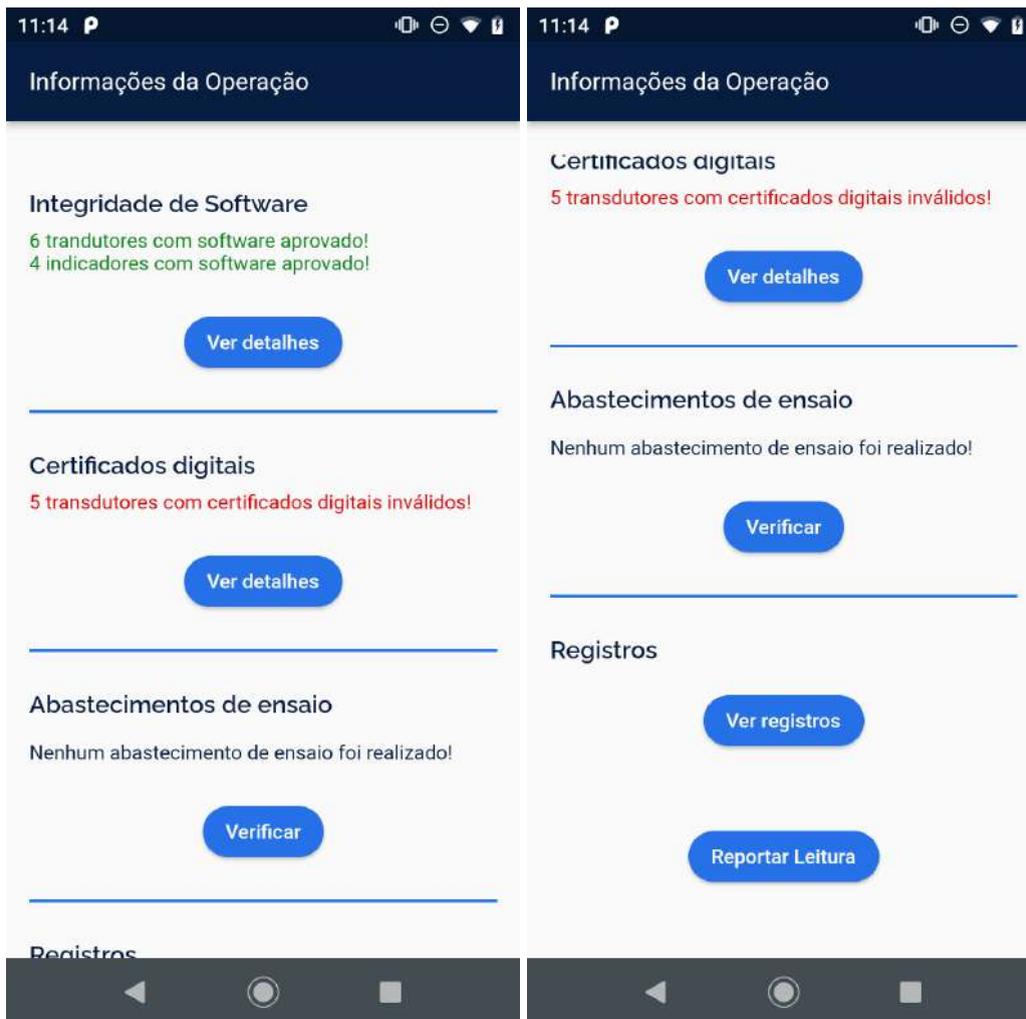


Figura 8: Ao rolar a tela, é possível ver o restante das informações.

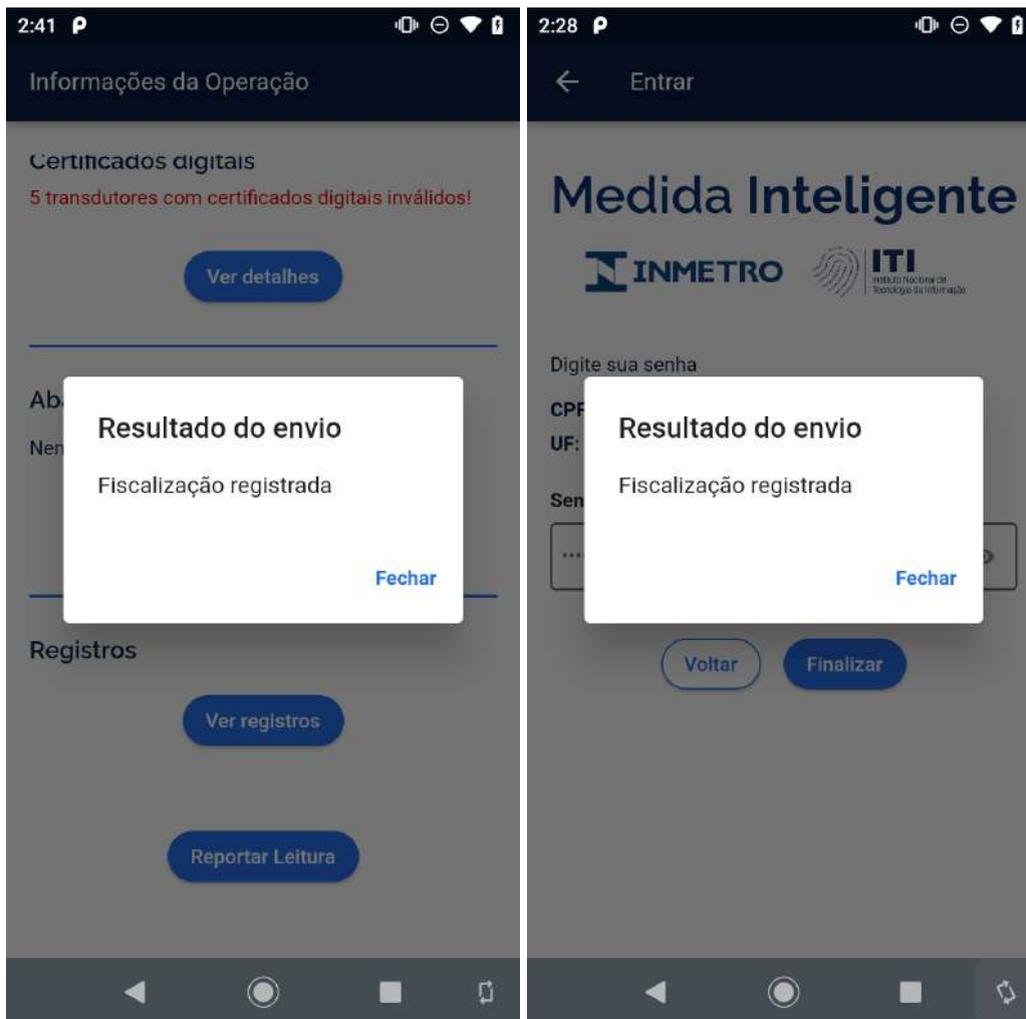


Figura 9: Janela modal de confirmação de envio da fiscalização ao servidor, a partir da tela de informações da fiscalização (à esquerda), e a partir da tela de login do fiscal (à direita) caso o fiscal não tenha feito login ainda no momento do envio da fiscalização.

4.4 Informações da Bomba de Combustível

Nesta tela (Figura 10) são mostradas mais informações referentes à bomba de combustível lida.

A partir desta tela:

- O botão *Voltar* retorna à tela “Informações da Operação” descrita anteriormente.



Figura 10: Tela de informações detalhadas sobre a bomba de combustível.

4.5 Informações do Estabelecimento

Nesta tela (Figura 11) são mostradas mais informações referentes ao estabelecimento da bomba de combustível lida.

A partir desta tela:

- O botão *Voltar* retorna à tela “Informações da Operação” descrita anteriormente.



Figura 11: Tela de informações detalhadas do estabelecimento.

4.6 Informações da Operação: Dados da integridade de Software

Nesta tela (Figura 12) são mostradas mais informações referentes à verificação de software dos dispositivos transdutores e indicadores da bomba de combustível.

A partir desta tela:

- O botão *Voltar* retorna à tela “Informações da Operação” descrita anteriormente.
- O botão *Verificar* repete a verificação de software dos dispositivos, realizada anteriormente quando a bomba de combustível foi selecionada para ser verificada. O botão *Verificar* fica disponível somente enquanto a bomba está pareada com o celular. Ao acessar a fiscalização pelo histórico, não é possível repetir a verificação de software.

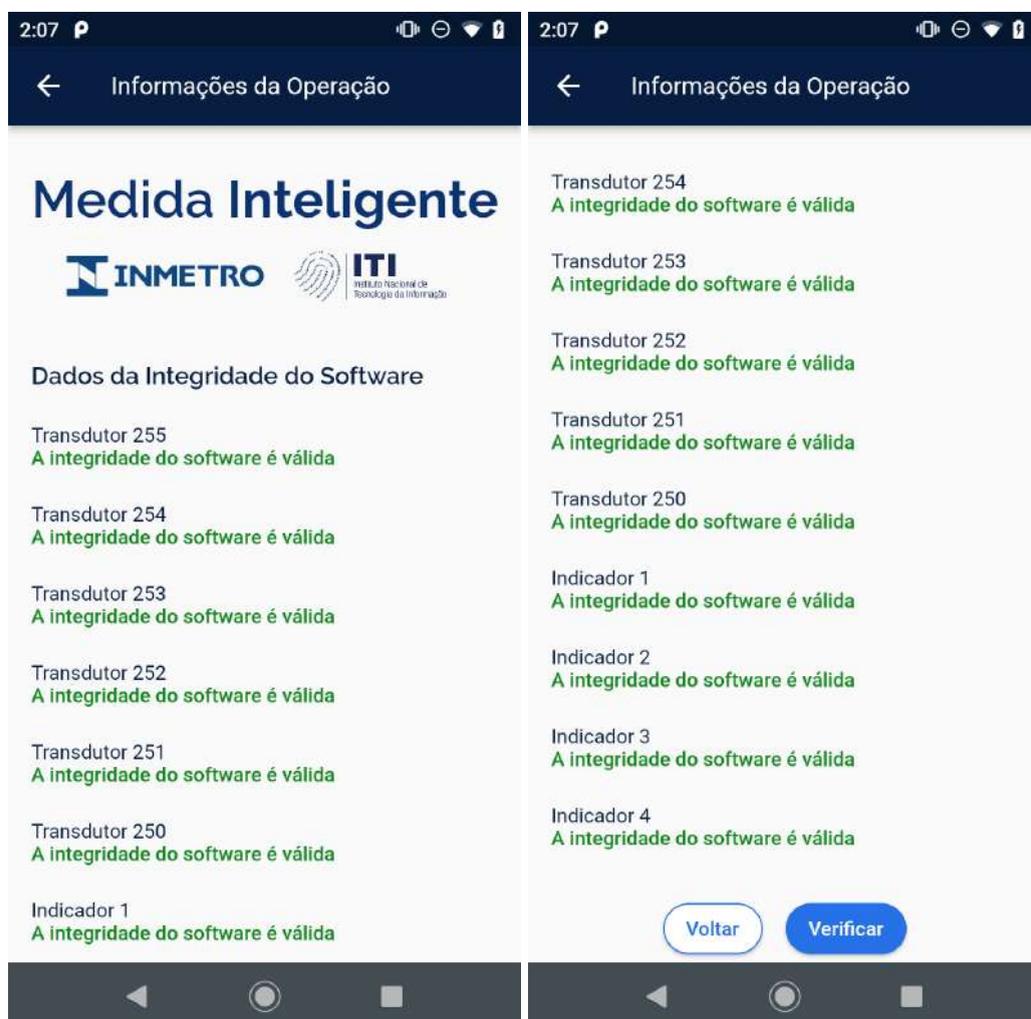


Figura 12: Tela de dados da integridade de software. Ao rolar a tela, é possível acessar os botões *Voltar* e *Verificar*.

4.5 Certificados Digitais

Nesta tela (Figura 13) são mostradas mais informações referentes aos certificados dos dispositivos transdutores da bomba de combustível lida.

A partir desta tela:

- O botão *Voltar* retorna à tela “Informações da Operação” descrita anteriormente.
- O botão *Repetir verificação* repete a verificação da validade dos certificados, realizada anteriormente quando a bomba de combustível foi selecionada para ser verificada. O botão *Repetir verificação* fica disponível somente enquanto a bomba está pareada com o celular. Ao acessar a fiscalização pelo histórico, não é possível repetir a verificação da validade dos certificados.

Para cada dispositivo transdutor, há um painel expansível que revela, quando clicado, mais informações sobre o certificado digital e sua cadeia de certificação. Dependendo do tamanho da tela do celular, é necessário rolar a tela para baixo para que se possa ver todas as informações disponíveis sobre o transdutor selecionado.

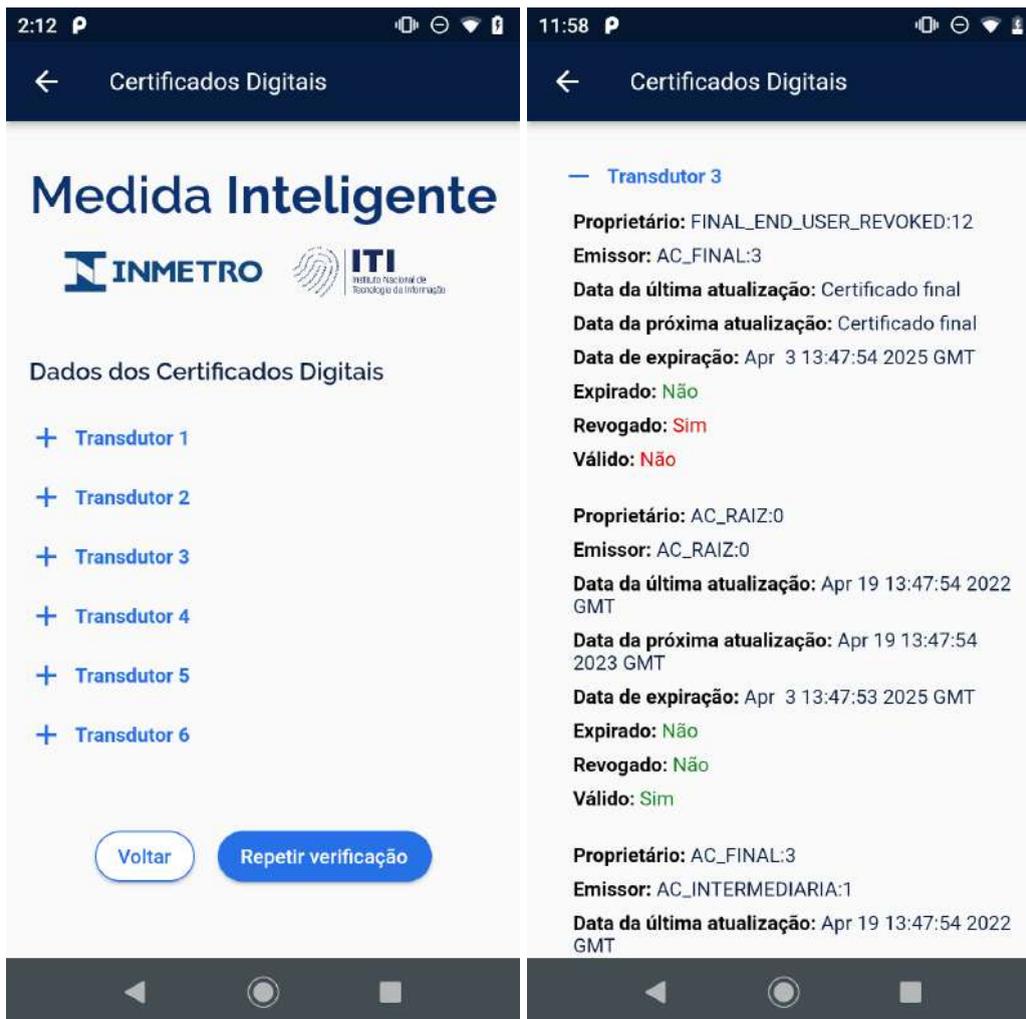


Figura 13: Tela de informações dos certificados digitais, separada por cada dispositivo transdutor da bomba de combustível. Ao abrir o painel expansível, é possível ver os dados do certificado e sua cadeia de certificação.

4.6 Abastecimentos de Ensaio

Nesta tela (Figura 14, esquerda), acessada por meio do botão *Verificar* da figura 8, estão localizadas informações acerca dos abastecimentos de ensaio realizados. O botão *Nova Leitura*, localizado no centro da tela, inicia o processo de um abastecimento de ensaio. Há também um espaço para exibição do histórico de abastecimentos de ensaio relacionados à fiscalização em questão.

A partir desta tela:

- O botão *Nova Leitura* inicia o procedimento para leitura de um abastecimento de ensaio, levando à tela de seleção de lado da bomba de combustível (Figura 14, direita).
- Caso exista um abastecimento de ensaio já realizado, é possível clicar sobre ele, levando à tela de informações do abastecimento.
- O botão *Voltar* retorna à tela “Informações da Operação” descrita anteriormente.

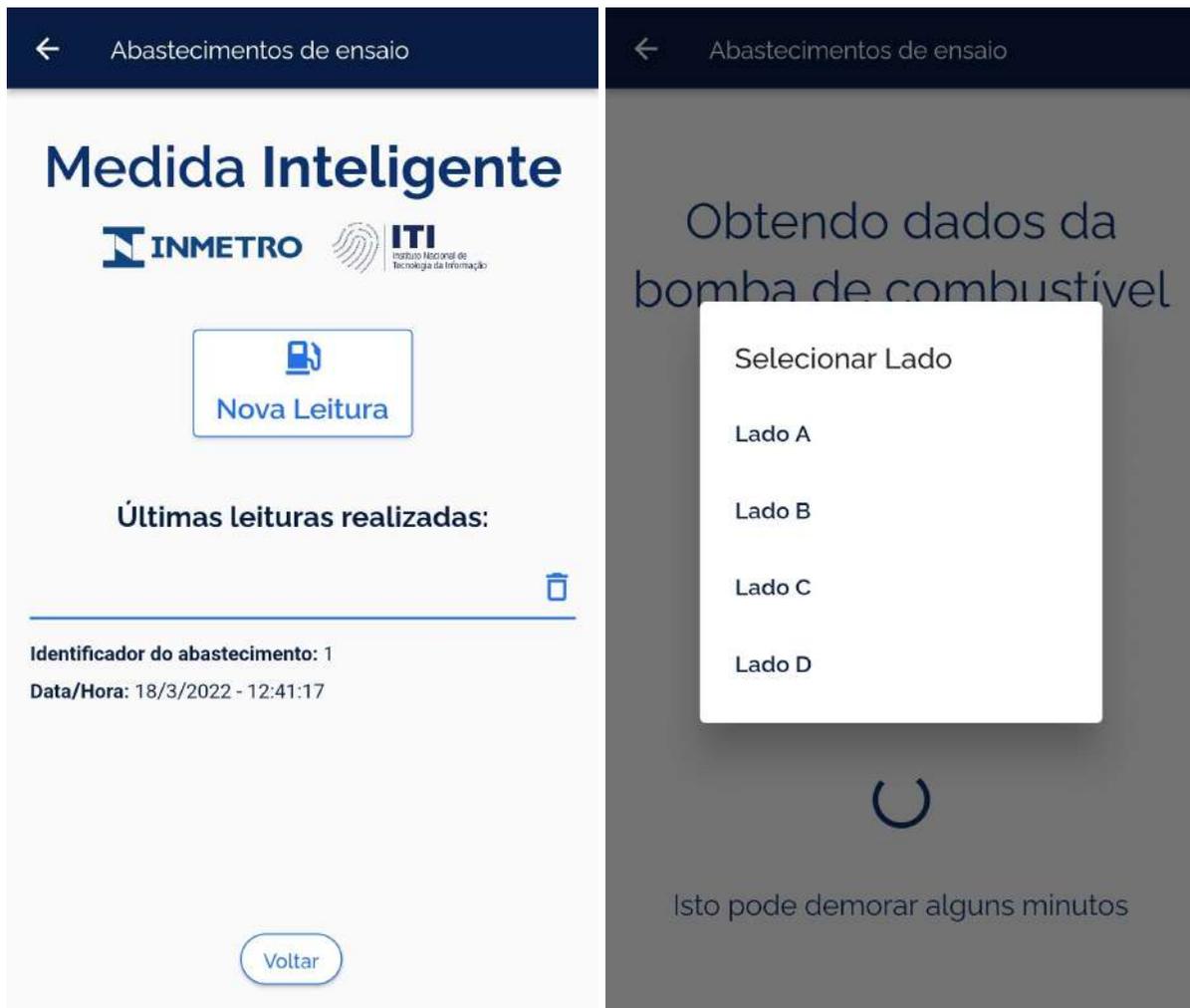


Figura 14: Tela inicial das informações dos abastecimentos de ensaio (à esquerda), janela modal mostrando a seleção de lados da bomba para realizar o abastecimento (à direita)

Ao selecionar o lado da bomba de combustível o aplicativo iniciará a obtenção dos dados da bomba de combustível, mostrando uma tela de carregamento.

4.6.1 Informações do abastecimento

Com a obtenção dos dados finalizada, é apresentada a tela de *Informações do abastecimento* (Figura 15, esquerda). Nesta tela são mostradas as informações sobre o abastecimento, o resultado da verificação do certificado digital e dados relacionados à bomba de combustível, incluindo os dados do estabelecimento. Caso não haja conexão com a Internet, os dados do estabelecimento aparecerão como indisponíveis.

A partir desta tela:

- Para obter mais informações da bomba de combustível e do abastecimento, clique no botão *Mais Informações*.
- O botão *Nova Leitura* retorna à tela *Abastecimentos de ensaio* descrita anteriormente.

4.6.2 Mais informações

A tela *Mais Informações* é exibida ao selecionar o botão de mesmo nome, na tela *Informações do Abastecimento*. Nesta tela (Figura 15, direita) são mostradas mais informações referentes à operação de abastecimento realizada pelo usuário.

A partir desta tela:

- O botão *Nova Leitura* realiza uma nova leitura.

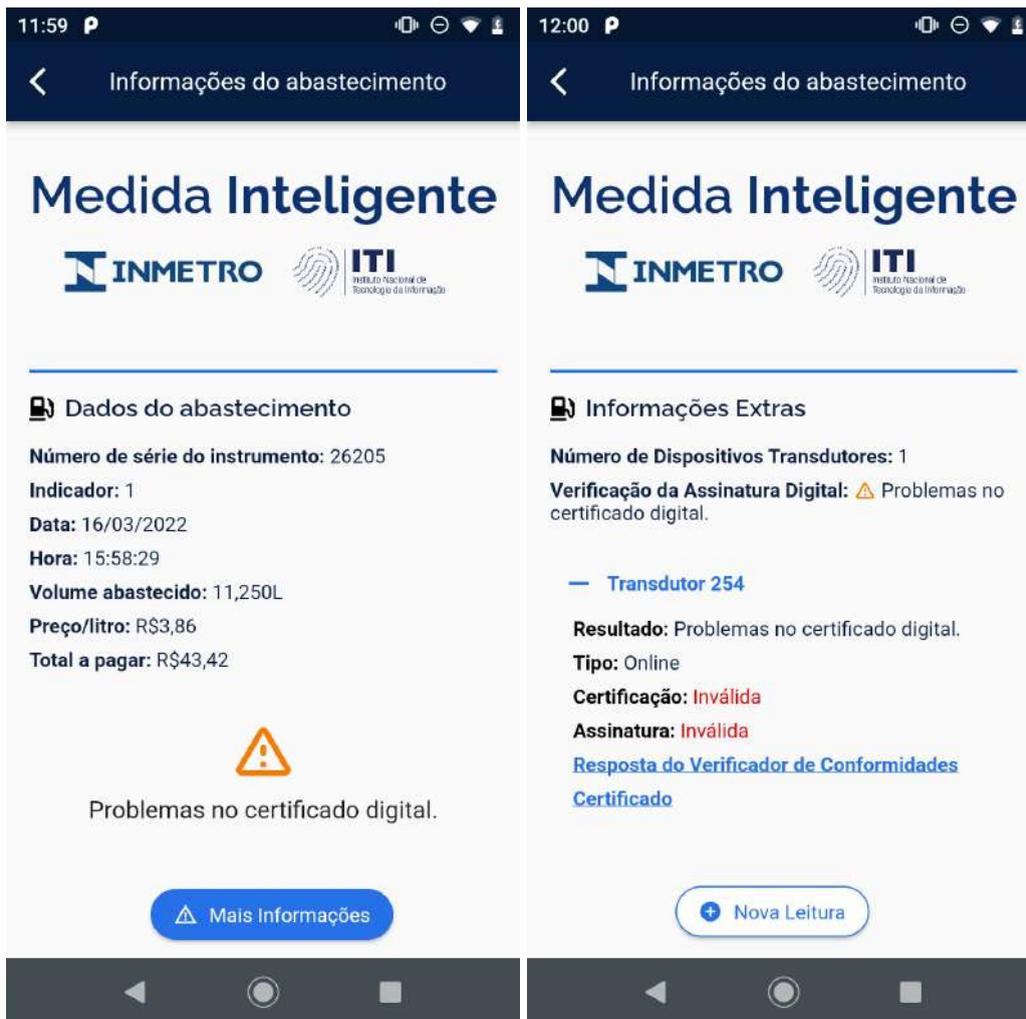


Figura 15: Tela de informações gerais do abastecimento (à esquerda), informações extras do abastecimento com painel expansível aberto (à direita)

No final das informações extras, há painéis expansíveis (Figura 15, direita), que revelam mais informações sobre as assinaturas digitais dos transdutores ao serem clicados. Dependendo do tamanho da tela do celular, é necessário rolar a tela para baixo para que se possa ver todas as informações disponíveis sobre o transdutor selecionado.

Clicando em *Resposta do Verificador de Conformidades* ou em *Certificado*, uma janela modal irá se abrir (Figura 16), mostrando as informações correspondentes ao item selecionado.

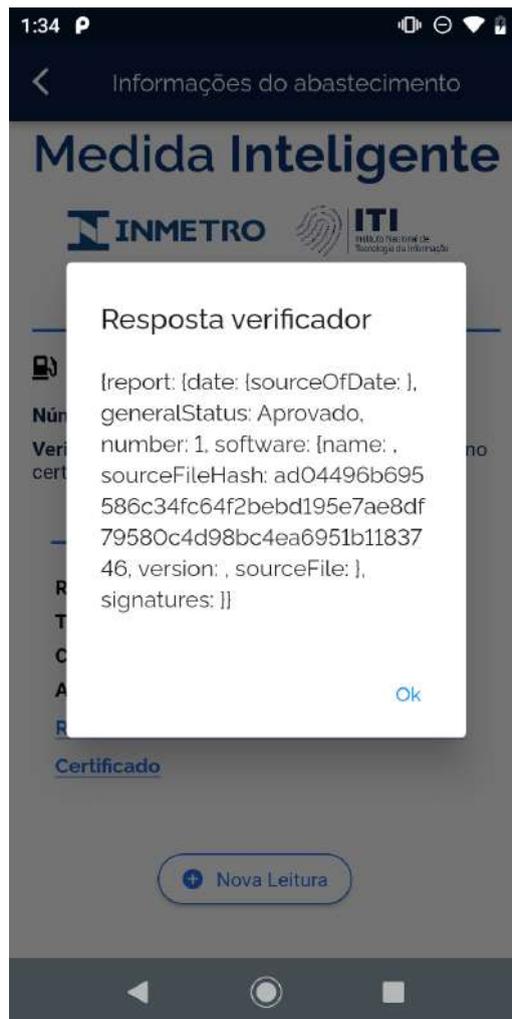


Figura 16: Janela modal que informa a resposta do verificador de assinaturas.

Importante ressaltar que somente é possível realizar abastecimentos de ensaio a partir de uma fiscalização recém feita. Caso uma fiscalização do histórico seja aberta, e o usuário navegue até a seção de abastecimento de ensaio e clique no botão *Nova Leitura*, aparecerá a mensagem demonstrada na Figura 17. Isso porque o aplicativo precisa de uma conexão bluetooth estabelecida com a bomba fiscalizada para requisitar qualquer tipo de informação. Ao acessar uma fiscalização pelo histórico, essa conexão entre aplicativo e bomba já foi desfeita.

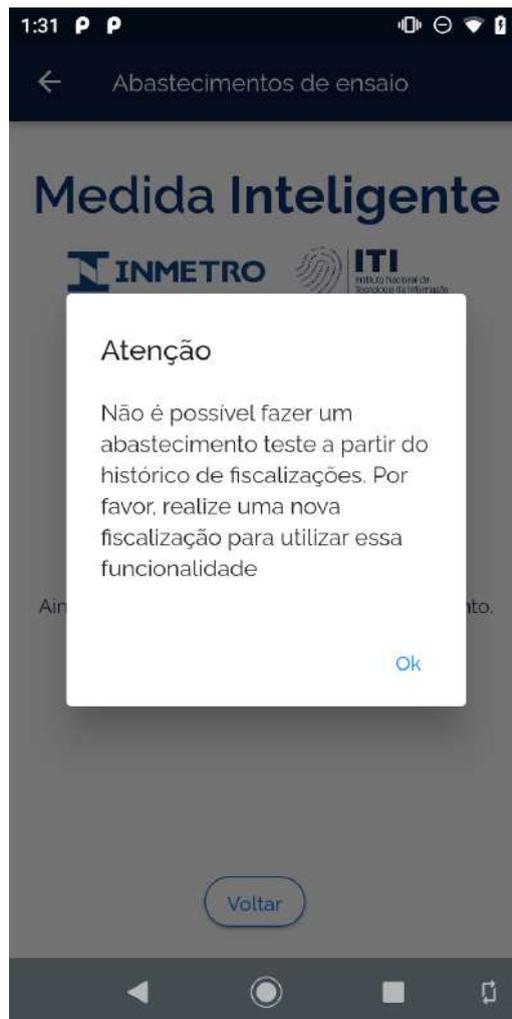


Figura 17: Janela modal que surge quando tenta se realizar abastecimentos de ensaio em uma fiscalização que foi acessada a partir do histórico de leituras.

4.7 Registros

Nesta tela (Figura 18) são mostradas mais informações referentes aos registros da bomba de combustível.

A partir desta tela:

- O botão *Voltar* retorna à tela “Informações da Operação” descrita anteriormente.
- Para ver mais informações sobre os registros de alterações de parâmetros da bomba de combustível, clique no hiperlink *Alteração de parâmetros relevantes*.
- Para ver mais informações sobre os registros de cargas de software da bomba de combustível, clique no hiperlink *Cargas de software*.
- Para ver mais informações sobre os registros de interrupções no funcionamento da bomba de combustível, clique no hiperlink *Interrupções no funcionamento do instrumento*.
- Para ver mais informações sobre os registros de eventos de manutenção da bomba de combustível, clique no hiperlink *Eventos de manutenção*.

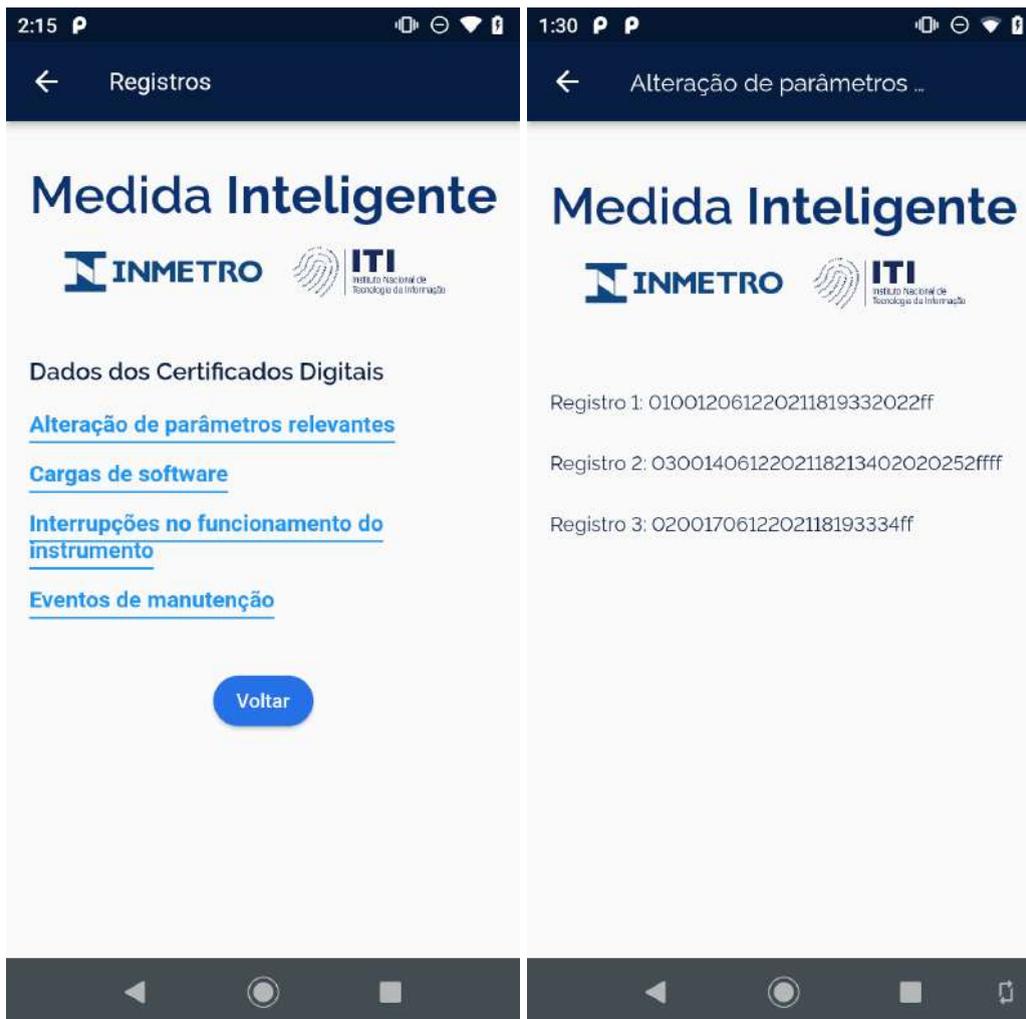


Figura 18: Tela de informações dos registros de fiscalização da bomba de combustível (à esquerda), tela de informações dos registros de alteração de parâmetros da bomba de combustível (à direita). O restante das telas de informações de registros específicos segue o mesmo padrão da tela à direita.

5 Considerações Finais

Este relatório apresentou instruções para instalação e execução do Aplicativo do Fiscal do sistema Medida Inteligente. Além disso, também contém uma explicação básica das funcionalidades do aplicativo e de como utilizá-lo.

O aplicativo em questão foi construído como prova de conceito para os estudos de viabilidade da comunicação *Bluetooth* com o objeto metrológico e da verificação de assinatura digital em dispositivos móveis. Contudo, o aplicativo também possui outras funcionalidades pensadas para o uso do fiscal, como o histórico de leituras, o envio de fiscalizações e a autenticação pelo SGI do Inmetro.

Anexo A - Configuração de endpoints

O aplicativo do fiscal realiza comunicação em rede com os seguintes serviços: API Medida Inteligente e Verificador de Conformidades. No caso de migração desses serviços entre ambientes de homologação e produção é necessário alterar os endpoints no código do aplicativo.

Para configurar o endpoint da API Medida Inteligente deve-se informar a URL na variável *baseUrl* do arquivo `lib\src\modules\api\url_config.dart`, como pode ser visto na Figura 19.

```
class UrlConfig {
  static const String baseUrl = 'inmetro.labsec.ufsc.br';
  static const String getEstabelecimentos = 'estabelecimentos';

  static const String getEstabelecimentoWithBMCSerial =
    'https://${baseUrl}/bmcs/';
  static const String estabelecimentoCompleto = '/estabelecimento';

  static const String sendFiscalizacao = 'https://${baseUrl}/fiscalizacao';
  static const String tabelasReferencia =
    'https://${baseUrl}/referencia-fabricantes/tabelas-referencia.json';

  static const String tabelasReferenciaCSV =
    'https://${baseUrl}/referencia-fabricantes/';

  static const String verificador =
    'https://${baseUrl}/verificador/enderecos-icp.json';

  static const String login = 'https://${baseUrl}/tokenSGI';
}
```

Figura 19. Configuração da URL da API.

E para configurar o endpoint do Verificador de Conformidades deve-se informar a URL na variável *baseUrl* do arquivo `crypto-development\lib\src\url_config.dart`, como pode ser visto na Figura 20.

```
class UrlConfigVerificador {

  static String baseUrl = "inmetro.labsec.ufsc.br";

}
```

Figura 20. Configuração da URL do Gov BR.

Anexo B - Obtenção de logs

Para obtenção dos logs do aplicativo através do .apk, sem ter o código fonte, deve-se seguir os seguintes passos:

1. Instalar o Flutter versão 2.10.5 (siga as instruções para baixá-la e instalá-la [nesse site](#)).
2. Instalar o aplicativo no celular através do .apk.
3. Conecte o celular no computador por meio de um cabo USB.
4. Em seguida, abra um terminal e com o aplicativo aberto no celular rode o seguinte comando:

```
flutter logs > app_log-DDMMAAAA_HHMM.txt
```

Substitua o DDMMAAAA_HHMM pela data e hora do momento da execução. Isso irá permitir diferenciar os logs caso mais de uma execução seja realizada.

5. Execute a leitura de uma nova fiscalização no aplicativo clicando em 'Nova Fiscalização'.
6. Após o término da execução do aplicativo, os logs do terminal estarão salvos no arquivo *app_log-DDMMAAAA_HHMM.txt* dentro do diretório atual.