

Desenvolvimento de Aplicação IoT utilizando a Plataforma Integrada Softway4IoT e Fiware

Yan Trindade, Luiz Calixto, Lucas C. Jorge, Andre G. Damaceno, Jacir L. Bordim

¹Departamento de Ciência da Computação – Universidade Federal de Brasília (UnB)
Caixa Postal 70910-900 – Brasília – DF – Brazil

{trindade.yan, calixtoluiz0, lucascj29, andre.garrido}@gmail.com, bordim@unb.br}

Abstract. *Internet of Things (IoT) covers a variety of devices, including sensors and actuators with different features and capabilities. The heterogeneity of IoT devices requires appropriate mechanisms for managing, collecting and making data available appropriately. In this context, the present work aims to evaluate two platforms that complement each other in the management of IoT devices. To this end, this paper presents an application that uses generic and reusable API's of the Fiware platform with the multi-sensor management capabilities of the Softway4IoT platform to abstract the different communication protocols in the development of a test application.*

Resumo. *A Internet das Coisas (IoT) abrange uma diversidade de dispositivos, incluindo sensores e atuadores com funcionalidades, capacidades e características bastante distintas. A heterogeneidade dos dispositivos IoT requer mecanismos adequados de gestão, coleta e disponibilização de dados. Neste contexto, o presente trabalho tem como objetivo avaliar duas plataformas que se complementam no gerenciamento de dispositivos IoT. Para tal, este trabalho apresenta uma aplicação que utiliza API's genéricas e reusáveis da plataforma Fiware com o poder de gerenciamento de múltiplos sensores da plataforma Softway4IoT para abstrair os diferentes protocolos de comunicação no desenvolvimento de uma aplicação de teste.*

1. Introdução

A Internet das Coisas (IoT) pode ser definida com uma rede de objetos com capacidade de comunicação, monitoramento, sensoriamento e compartilhamento de informações [Ashton 2009]. Em 2023, é esperado que existam mais de 14.7 bilhões de dispositivos IoT e, com eles, surge a necessidade de plataforma que permitam gerenciar tais dispositivos, armazenar e processar dados gigantescos para a construção de aplicativos inteligentes como *Smart Cities*, *Smart Healthcare*, dentre outros.

Tendo em vista a heterogeneidade dos dispositivos IoT, em particular no que concerne tecnologias de comunicação distintas, surge a necessidade de desenvolver mecanismos que possibilitem orquestrar estes dispositivos [Augustin et al. 2016, SEMTECH 2015]. Neste contexto, a Universidade Federal de Goiás - UFG desenvolveu o Softway4IoT [UFG 2020b], uma plataforma que permite o gerenciamento *web* de diversos *gateways*, dispositivos de borda interligados aos sensores e atuadores. A Universidade de Brasília, integrou este grupo em momento posterior, visando o desenvolvimento de uma aplicação que possibilitasse a integração destes dois ambientes: Fiware [Fiware Foundation 2021] e Softway4IoT [UFG 2020b].

Este documento apresenta uma demonstração de integração do ambiente Fiware com a tecnologia Softway4Iot. Foi utilizado como aplicação de teste o monitoramento veicular GPS em um ônibus escolar com notificações de trajeto e proximidade. O objetivo é unir as API's genéricas e reusáveis da plataforma Fiware com o poder de gerenciamento de múltiplos sensores e abstrair os diferentes protocolos de comunicação com o *gateway* Softway4IoT. A aplicação teste escolhida visa explorar as características, benefícios e limitações destas plataformas em um ambiente que requer a conexão entre os dispositivos IoT e o *gateway* Softway4IoT bem como a publicação e disponibilização de dados de localização e demais informações do veículo. Desta forma, a aplicação fornecerá dados que possibilitem a identificação de cada veículo, sua localização, bem como dados específicos para gerenciamento da frota, como velocidade média e percurso, além de permitir o envio de mensagens de alerta quando o veículo se aproximar do ponto de interesse (parada de ônibus) do usuário.

O restante deste documento está organizado da seguinte forma: A Seção 2 apresenta uma visão geral das plataformas IoT e em maiores detalhes Fiware e Softway4IoT. Após, a Seção 3 apresenta a integração entre as plataformas. A Seção 4 possui os resultados do trabalho, apresentando a aplicação IoT desenvolvida pela equipe da UnB, com vistas a utilizar de forma conjunta as principais funcionalidades e características dos projetos, além de citar os trabalhos futuros a serem feitos. Por fim, A Seção 5 apresenta uma discussão sobre vantagens e desvantagens da escolha de utilizar a integração entre as plataformas Fiware para o desenvolvimento de aplicações IoT e discorre sobre trabalhos futuros.

2. Plataformas IoT

Com a popularização de dispositivos IoT, surgiram plataformas de apoio ao desenvolvimento de aplicações, como o AWS IoT Plataforma [AWS 2021], Microsoft Azure IoT Hub [Azure 2021], Google IoT Platform [Google 2021]. Alternativas de código aberto que visam fornecer serviços de gerenciamento de dados incluem o Kaa [Kaa 2021] ThingSpeak [ThingSpeak 2021], e Fiware [Fiware Foundation 2021]. No que segue, centraremos nossa discussão na plataforma Fiware.

O Fiware é um projeto patrocinado pela Comissão Europeia para o desenvolvimento de tecnologias para a Internet das coisas. A iniciativa Fiware fornece um conjunto de interfaces de programação de aplicativos (APIs) para desenvolver aplicativos inteligentes, com base em especificações abertas e livres de royalties [S. Krco 2014]. O Fiware especifica componentes chamados *habitadores* genéricos que constituem os blocos de construção para aplicativos inteligentes em diferentes áreas. Esses *ativadores* genéricos abrangem aspectos que incluem, mas não estão limitados a, segurança, gerenciamento de dados, hospedagem em nuvem e gerenciamento de dispositivos. Para maiores detalhes sobre outras plataformas direcionamos o leitor para a referência [Agarwal and Alam 2020]. O Fiware define um conjunto universal de padrões para o gerenciamento de dados de contexto, o que facilita o desenvolvimento de soluções inteligentes para diferentes domínios, como cidades inteligentes, indústria inteligente, alimentos agroalimentares e energia inteligente. O Fiware permite gerenciar informações de contexto, processando essas informações e informando os atores externos, permitindo que eles atuem e, portanto, alterem ou aprimorem o contexto atual. O Context Broker é o componente principal da plataforma. Ele permite que o sistema execute atualizações e acesse o estado atual do

contexto. A Figura 1 apresenta um diagrama dos principais componentes do Fiware. A arquitetura do Fiware foi concebida com o objetivo de reaproveitar um conjunto de componentes conhecidos como Generic Enablers (GE), esses GEs estão descritos a seguir:

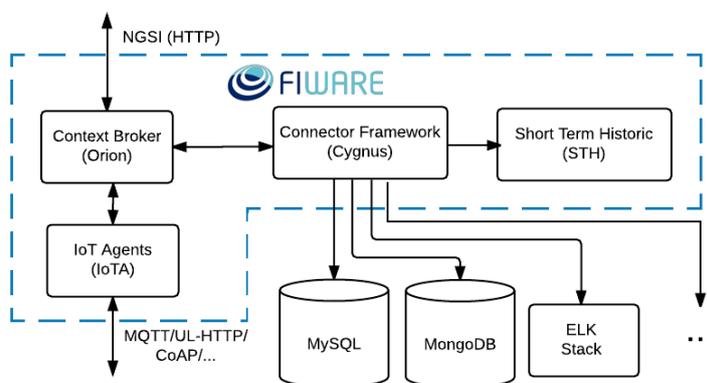


Figura 1. Esquemático da Arquitetura Fiware

ORION - Context Broker: O uso do Orion Context Broker (com dados de contexto apropriados fluindo através dele) é suficiente para que um aplicativo se qualifique como "Powered by Fiware". Portanto, a arquitetura consistirá em dois elementos: O Orion Context Broker que receberá solicitações usando a API NGSI (*Next Generation Service Interface*) do Fiware. A NGSI é uma API RESTful via HTTP que possui como principais operações o registro e a descoberta de entidades de contexto.

IoT Agent: Um Agente IoT é um componente que permite que um grupo de dispositivos envie seus dados e seja gerenciado a partir de um Context Broker usando seus próprios protocolos nativos. Os agentes de IoT também devem ser capazes de lidar com aspectos de segurança da plataforma Fiware (autenticação e autorização do canal) e fornecer outros serviços comuns ao programador de dispositivos. Os agentes de IoT já existem ou estão em desenvolvimento para muitos protocolos de comunicação e modelos de dados da IoT tais como HTTP e LoRaWAN.

Cygnus: Cygnus é um componente de persistência dos dados existentes na plataforma Fiware. Este componente é usado para persistir dados de contexto em bancos de dados de terceiros usando o Apache Flume, criando uma visão histórica do contexto.

Short Term History: Sua principal responsabilidade é a geração de informações agregadas de contexto de séries temporais sobre a evolução no tempo desses valores de atributo da entidade. Em relação à geração de informações agregadas de contexto de séries temporais, o STH gerencia 4 conceitos principais: período de agregação (entre datas), origem (desde), deslocamento (offset), amostras (número de resultados).

Na IoT, é comum que os dispositivos apresentem tecnologias de comunicação distintas, tais como LoRa / LoRaWAN, 4G / 5G e Wi-Fi [UFG 2020b]. A proposta de integração com o Softway4IoT une as API's genéricas e reusáveis do Fiware com a funcionalidade de gerenciar diversos sensores sem considerar as particularidades das tecnologias de comunicação.

Para conectar esses dispositivos à Internet, é preciso que um elemento de rede realize a tradução entre as diversas tecnologias utilizadas. Neste contexto, o projeto SOFTWAY4IoT [UFG 2020b], desenvolvido pela UFG é um *gateway* responsável por prover uma infraestrutura para o desenvolvimento de aplicações e gerenciar sensores e atuadores de forma escalável fornecendo a infraestrutura necessária à aplicação. O SOFTWAY4IoT tem como foco, além de prover a interoperabilidade entre tecnologias distintas, a de:

- Permitir a redução do impacto da escolha de uma tecnologia sem fio para IoT ainda não consolidada e não que não sejam interoperáveis com a arquitetura TCP/IP da Internet;
- Evitar a exposição pública de dispositivos IoT na Internet, ou seja, minimizar riscos potenciais relacionados à segurança da rede dos dispositivos;
- Permitir a conectividade do dispositivo à infraestrutura para coleta e processamento de dados, normalmente uma infraestrutura de computação em nuvem.

Para prover tais funcionalidades, o SOFTWAY4IoT incorpora tecnologias de Rede Definida por Software (SDN) [Arbiza 2016] e Rádio Definido por Software (SDR) [Sun et al. 2015]. Com o uso das funcionalidades providas por uma SDN, o SOFTWAY4IoT pode criar redes virtuais isoladas e implementar políticas de segurança de rede de forma rápida e flexível. De forma semelhante, o SOFTWAY4IoT utiliza tecnologias de SDR para viabilizar a comunicação sem fio com várias tecnologias usando apenas uma interface física. Ainda, o uso de SDN permite a utilização ou mesmo substituição das tecnologias de comunicação por meio de soluções implementadas em software. Por fim, o SOFTWAY4IoT incluirá a integração do *gateway* de comunicação com uma infraestrutura de computação de névoa para fornecer maior garantia de conectividade entre os sensores e/ou atuadores e a aplicação de “inteligência”, bem como aplicações de suporte que requerem baixa latência e facilitam a virtualização de os componentes da solução [UFG 2020b].

3. Integração Fiware e Softway4IoT

A integração das plataformas Fiware e Softway4IoT estão descritos no relatório de integração elaborado pela UFG [UFG 2020a]. O objetivo desta integração é de unir as APIs genéricas e reusáveis da plataforma Fiware com o poder de gerenciamento de múltiplos sensores e abstrair os diferentes protocolos de comunicação com o *gateway* Softway4IoT.

A Figura 2 demonstra a integração da plataforma Softway4IoT e Fiware. Há duas formas de integração entre as plataformas, uma via requisição HTTP para os sensores que estão ligados a dispositivos que possuem um endereço IP e a outra utilizando o protocolo LoRa para sensores ligados a dispositivos que não estão conectados à rede e não possuem um endereço IP. Na aplicação desenvolvida fizemos a aplicação se comunicar por meio da utilização de requisições HTTP com o IoT Agent do Fiware.

O Softway4IoT possui na sua arquitetura um componente *gateway manager* responsável por gerenciar um ou mais *gateways*. O *gateway*, por sua vez, é gerenciado ou orquestrado pelo *gateway manager*. No *gateway*, a aplicação *web* desenvolvida foi containerizada utilizando *Docker* [Docker 2021]. Ou seja, o *gateway* tem a responsabilidade de prover a infraestrutura necessária para execução da aplicação encapsulando as

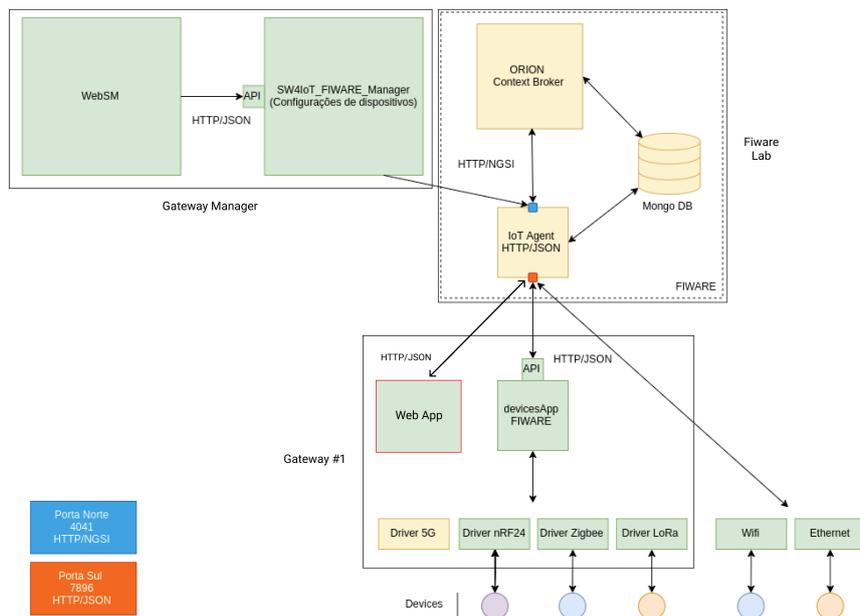


Figura 2. Esquemático da arquitetura de integração do Softway4IoT e Fiware (adaptado de [UFG 2020a])

dificuldades de infraestrutura e *deploy*. Já o Fiware LAB está anexo às instâncias do *gateway manager* e do *gateway*. As APIs REST e componentes reusáveis do Fiware estão acessíveis e disponíveis a todos os *gateways* e aplicações gerenciados pelo *gateway manager*. A comunicação entre os componentes ocorre por meio de APIs REST. Já entre os sensores podem ocorrer diversos protocolos diferentes e são todos gerenciados pela interface WEBSM do Softway4IoT.

3.1. Fiware + Softway4IoT: Caso de Uso

Os benefícios da integração entre as plataformas inclui a escalabilidade no gerenciamento de sensores; abstração das tecnologias de comunicação; robustez durante o processo de *deploy*; e, redução no uso de recursos de desenvolvimento de software. Na aplicação de teste usando as duas plataformas, as tecnologias acima descritas foram empregadas em um contexto específico de gerenciamento de frotas em áreas de campus universitários pois na UnB há uma grande dificuldade em controlar a própria frota de veículos. A aplicação de teste utiliza um *gateway manager*, o Fiware e um *gateway*.

A aplicação de teste tem como contexto um sistema de gerenciamento de frotas de veículos para campi universitários. Esse contexto também pode ser aplicado em outras organizações com necessidades semelhantes. O objetivo desta aplicação é demonstrar a integração entre as plataformas Fiware e Softway4IoT. A aplicação de teste terá três principais componentes: o *gateway manager*, O Fiware e um *gateway*. Esses componentes foram implantados em uma infraestrutura em nuvem pública provida pela empresa Digital Ocean [Sun et al. 2019].

A Figura 3(b) ilustra a integração com sensores que se comunicam por meio do protocolo LoRa e precisam enviar as informações para o *gateway* pois a instância

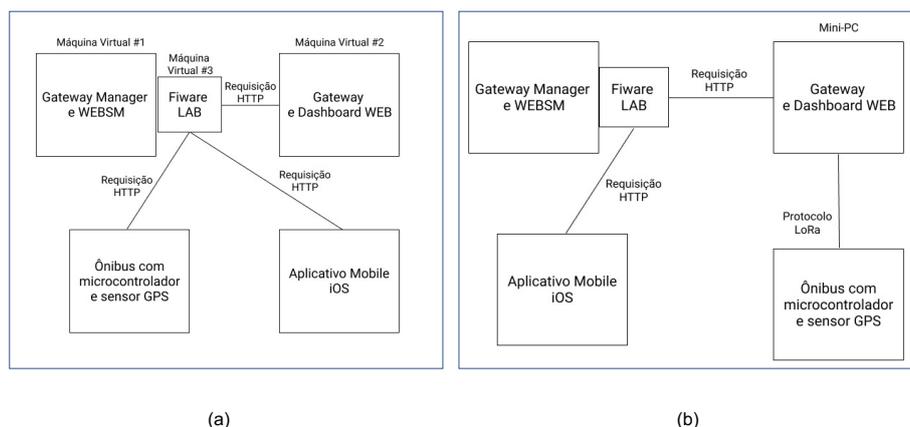


Figura 3. Esquemático geral da solução de rastreamento de frota utilizando a integração por requisição (a) HTTP e (b) LoRa

do IoT Agent nesta integração permite somente requisições HTTP e em formato JSON [Friesen 2016] logo o sensor necessita enviar este dado para o Softway4IoT e este se encarrega da entrega no formato aceito pelo Fiware. Todo o gerenciamento dos dispositivos é feito pela interface *web* do Softway4IoT WEBSM. A segunda forma de integração é via requisição HTTP feita diretamente para o Fiware.

A Figura 3(a) representa a comunicação via requisições HTTP desde o envio da posição (x, y, z) do sensor GPS atrelado ao ônibus para o Fiware que é o *backend* da aplicação até a leitura e gerenciamento dos trajetos do ônibus feita pela aplicação *web* e leitura da última posição do ônibus e trajeto pelo aplicativo para celular. No Fiware o ônibus itinerante foi abstraído na forma de uma entidade utilizando o componente Orion. A entidade pode ser entendida como uma classe que possui atributos e os atributos são os campos a serem alterados por um ou mais sensores que fazem parte do contexto desse dispositivo. A entidade é identificada unicamente por um id alfanumérico. Temos algumas constantes na entidade veículo como placa, cor, modelo, ano e atributos que serão alterados a cada leitura do sensor como a posição (x, y, z) e a velocidade.

O sensor faz a leitura de velocidade, latitude, longitude, altitude, data e horário da leitura. Os dados são encapsulados em um JSON [Friesen 2016] e enviados via requisição HTTP para a API do Fiware. O Fiware armazena esse valor no seu banco de dados MongoDB [Chodorow 2013].

Os aplicativos *web* desenvolvido em javascript utilizando o framework React [Aggarwal 2018] e o aplicativo Mobile desenvolvido em Swift [Neuburg 2016] fazem a leitura da última posição (dados de contexto) armazenados no Fiware. A Figura 4 é a entidade criada no Fiware que representa o ônibus e armazena seus respectivos dados de contexto e atributos utilizados na aplicação.

4. Resultados

Para fins de teste de integração mencionado na seção anterior, foi desenvolvido um sistema de gerenciamento de frotas de veículos escolares que é composto por duas aplicações, uma *web* e outra *mobile* que possuem funcionalidades obtidas por meio do Fiware e sensores que são gerenciados pelo Softway4IoT de forma escalável e robusta.

```

1  {
2  {
3    "id": "001",
4    "type": "Car",
5    "carBrand": "Honda",
6    "carColor": "Preto",
7    "carModel": "Civic",
8    "carYearModel": "2016/2017",
9    "dateModified": "2016-12-07T11:10:00.00Z",
10   "department": "CIC",
11   "isPublic": true,
12   "lastMaintenance": "2016-12-07T11:10:00.00Z",
13   "location": {
14     "type": "Point",
15     "coordinates": [
16       -47.871676,
17       -15.760356
18     ]
19   },
20   "nextMaintenance": "2016-12-07T11:10:00.00Z",
21   "plate": "JHS 1589",
22   "route": "[[-47.872337,-15.762786], [-47.872896, -15.760700], [-47.870329, -15.759990], [-47.869318,-15.758646]]",
23   "speed": "0.0",
24   "tags": [
25     "Operando"
26   ]
27 }
28 }

```

Figura 4. JSON da entidade carro

No que diz respeito ao sistema, não foi necessário desenvolver uma API própria, reusamos apenas as API's e componentes genéricos do Fiware, reduziram-se assim os recursos humanos necessários para o desenvolvimento desta plataforma. A integração utilizada foi a representada na Figura 3(a) por meio de requisição HTTP e conectividade com a internet para o envio de dados do sensor acoplado ao ônibus para a entidade que armazena os dados de contexto no Fiware.

As funcionalidades da plataforma foram obtidas a partir da abstração do ônibus na entidade do Fiware conforme Figura 4. A entidade é responsável pelo armazenamento dos dados de contexto: velocidade, posição, data da última leitura e metadados de ônibus (trajeto, placa, modelo, carro, fabricante). Os dados de contexto da entidade possuem API's para operações CRUD. As aplicações web e mobile consomem dados via API's por meio de reuso do Fiware. Os dados de leitura do sensor são enviados por meio de requisição HTTP para a entidade. Os sensores são gerenciados de forma escalável pelo WEBSM da plataforma Softway4IoT e a complexidade do deploy da aplicação web é reduzida no *Gateway Manager* por meio do uso de Docker.

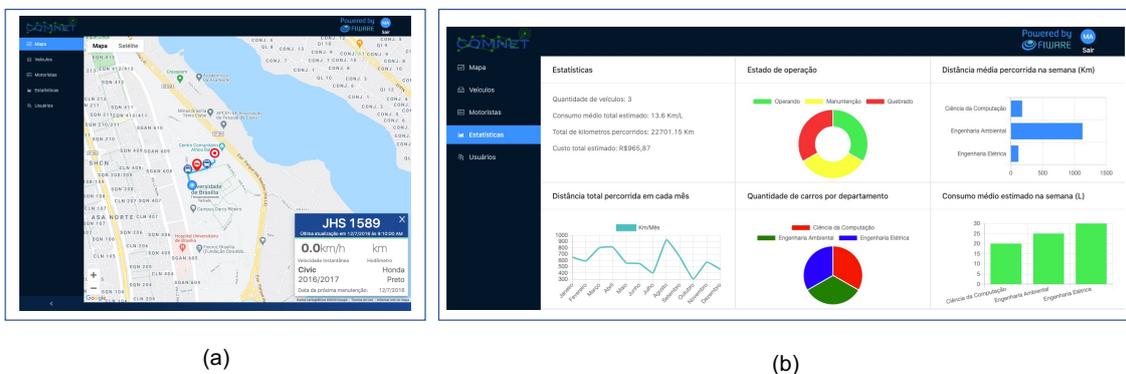


Figura 5. (a) Rota do veículo, denotando o ponto de início, destino e posição atual e (b) Tela de Estatísticas

Nesse contexto, a aplicação envolve o usuário administrador que faz o gerenciamento dos veículos por meio de plataforma *web*. O usuário do veículo de interesse em saber sua posição real e se já passou por determinadas paradas num trajeto fixado para cada veículo conforme ilustrado na Figura 4. O Administrador de frota interage com a aplicação *web* e por meio dela cadastra os veículos, visualiza todos os veículos cadastrados bem como suas rotas e visualiza as estatísticas inferidas por meio dos dados coletados e característicos de cada veículo conforme ilustrado na Figura 5(a) e na Figura 5(b). O usuário visualiza a posição de um veículo de interesse e recebe notificações sobre a posição em relação ao ponto de origem e proximidade de paradas que fazem parte do trajeto deste veículo por meio da aplicação *mobile*. Por fim, temos o dispositivo de hardware que em nosso contexto se trata do sensor acoplado a um microcontrolador localizado no interior do veículo para registrar a posição, velocidade e data da leitura.

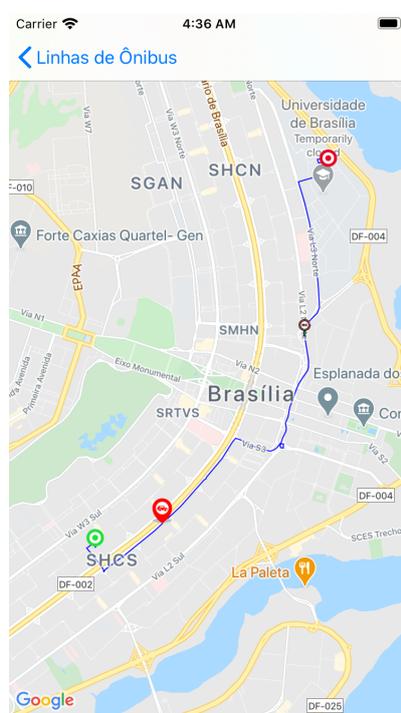


Figura 6. Rota do veículo, denotando o ponto de início, destino e posição atual

As dificuldades encontradas durante o desenvolvimento da aplicação de teste foram a necessidade do estudo e aprendizado da elaboração de *scripts* em Docker para realizar o *deploy*, o que gerou uma complexidade extra aos desenvolvedores. A infraestrutura necessária para a implantação do Softway4IoT e Fiware exige no mínimo três instâncias de máquinas virtuais, conforme manual elaborado pela UFG, o que se torna um fator limite para cenários mais simples. O *deploy* do Softway4IoT faz uso da ferramenta Ansible [Heap 2016] o que exige um conhecimento prévio da ferramenta. Apesar destas dificuldades, o Softway4IoT é um projeto em desenvolvimento pela UFG e eventuais adaptações e melhorias poderá ser contemplada em futuras versões. Destaca-se que embora existam outros *gateways* disponíveis, estes requerem um dispositivo que permita a conexão direta entre os dispositivos IoT e o *gateway*. O Softway4IoT, por outro lado, implementa protocolos nativos de forma a viabilizar a conexão direta entre os dispositivos e

o *gateway*. Portanto, a depender dos requisitos e necessidade da aplicação, o Softway4IoT poderá ser uma alternativa mais apropriada. No caso em particular tratado neste trabalho, várias tecnologias foram consideradas, dentre elas LoRa e Wi-Fi para o desenvolvimento de aplicações.

Como mencionado, a infraestrutura mínima necessária para implantação do Softway4IoT e o conhecimento prévio de outras ferramentas trazem uma complexidade adicional aos desenvolvedores e são um contraponto ao ganho de escalabilidade no gerenciamento de sensores e abstração de protocolos de comunicação. Essa questão sugere um ponto de atenção a aplicações de pequenas escala ou que não utilizam múltiplos protocolos ao optar utilizar esta integração. Por outro lado, o tempo de desenvolvimento de software foi reduzido uma vez que todo o *backend* responsável pelo armazenamento de dados foi feito pelo Fiware e seus GE's.

Uma vez concluída a instalação do Softway4IoT e Fiware, caso a aplicação faça uso somente do Fiware no que tange o armazenamos de dados e reuso de API's para implementar suas funcionalidades, os recursos humanos e tempo relativos ao desenvolvimento de software tendem a reduzir proporcionando facilidades aos desenvolvedores de soluções IoT. Por fim, uma possível evolução desta integração entre as plataformas seria a redução do número mínimo de instâncias necessárias para a instalação em ambiente final e criação de imagens prontas em provedores infraestruturas em nuvem tais como Digital Ocean ou Amazon AWS onde o usuário apenas selecionasse o número de *gateways*, *gateway manager* e instâncias do Fiware, todo o deploy já seria automatizado reduzindo assim o esforço necessário aos desenvolvedores IoT.

5. Conclusão

Este trabalho demonstrou a integração entre as plataformas Fiware e Softway4IoT onde foi desenvolvida uma aplicação de teste de um sistema de gerenciamento de frotas de veículos em que todas as funcionalidades planejadas para a aplicação de teste foram implementadas. A integração das plataformas tem uma dificuldade da necessidade de conhecimento prévio de ferramentas tais como Docker, Ansible e configuração de um provedor de infraestrutura em nuvem ou disponibilidade *hardware* físico conforme requisitos mínimos especificados na documentação de cada plataforma.

No que concerne a aplicação e a integração, como trabalho futuro, há o interesse de avaliar os aspectos de desempenho e escalabilidade da integração e dos seus diversos componentes. Já a respeito da aplicação de monitoramento veicular, outras funcionalidades para métricas e monitoramento de veículos também podem ser desenvolvidas. Por exemplo, a utilização de sensores tais como um leitor de OBDII [Ribeiro 2015], o qual faria a leitura de sensores internos do veículo, poderá aumentar acurácia das informações e prover uma maior riqueza de detalhes que podem ser utilizados para melhor gerenciamento dos veículos.

Referências

Agarwal, P. and Alam, M. (2020). Investigating iot middleware platforms for smart application development. In *Smart Cities—Opportunities and Challenges*, pages 231–244. Springer.

- Aggarwal, S. (2018). Desenvolvimento web moderno usando reactjs. In *Desenvolvimento web moderno usando reactjs*, volume 5, pages 2349–7688.
- Arbiza, L. M. R. (2016). Sdn no contexto de iot: refatoração de middleware para monitoramento de pacientes crônicos baseada em software-defined networking. Master's thesis, Universidade Federal do Rio Grande do Sul.
- Ashton, K. (2009). That internet of things thing. *RFiD J. vol. 22*.
- Augustin, A., Yi, J., Clausen, T., and Townsley, W. (2016). A study of lora: Long range & low power networks for the internet of things. *Sensors*, 16(9):1466.
- AWS (2021). AWS IoT. <https://aws.amazon.com/pt/iot/>.
- Azure (2021). Azure IoT. <https://azure.microsoft.com/pt-br/overview/iot/>.
- Chodorow, K. (2013). *MongoDB: The Definitive Guide*. "O'Reilly Media, Inc."
- Docker (2021). Docker. <https://www.docker.com/>.
- Fiware Foundation (2021). Fiware. <https://www.fiware.org/about-us/>.
- Friesen, J. (2016). *Java XML and JSON*. Springer.
- Google (2021). Google iot. <https://cloud.google.com/solutions/iot?hl=pt-br>.
- Heap, M. (2016). Ansible. *Apress*.
- Kaa (2021). Kaa. <https://www.kaaproject.org/>.
- Neuburg, M. (2016). *IOS 10 Programming Fundamentals with Swift: Swift, Xcode, and Cocoa Basics*. "O'Reilly Media, Inc."
- Ribeiro, H. M. G. (2015). Dados can/obd2 em tempo real de viaturas auto. *Faculdade de Engenharia Universidade do Porto*.
- S. Krco, B. Pokric, F. C. (2014). Designing iot architecture (s): A european perspective, in: Internet of things (wf-iot). *2014 IEEE World Forum on*, 2014.
- SEMTECH (2015). *LoRaTM Modulation Basics*. Semtech Corporation.
- Sun, S., Kadoch, M., Gong, L., and Rong, B. (2015). Integrando virtualização de função de rede com sdr e sdn para redes 4g / 5g. *rede IEEE*, 29(3):54–59.
- Sun, W., Wei, Z., Hong, B., and Yang, Y. (2019). A digital ocean cloud platform architecture baseada em ipv6 smart gateway. In *2019 IEEE 4ª Conferência Internacional sobre Computação em Nuvem e Análise de Big Data (ICCCBDA)*, pages 438–442. IEEE.
- ThingSpeak (2021). Thingspeak. <https://thingspeak.com/>.
- UFG (2020a). Relatório da integração entre FIWARE e SOFTWAY4IoT produzido pela ufg. https://github.com/LABORA-INF-UFG/SOFTWAY4IoT-F3-Docs/tree/master/FIWARE#sw4iot_fiware_manager.
- UFG (2020b). Softway4IoT. <https://softway4iot.labora.inf.ufg.br>.