

Usando o FIWARE para Desenvolvimento de Aplicações de Cidades Inteligentes

Cláudio Trindade, Arthur Souza, Jorge Pereira,
Juliana Oliveira, Nélio Cacho, Thaís Batista,
Everton Cavalcante
Departamento de Informática Aplicada - DIMAP
Universidade Federal do Rio Grande do Norte- UFRN
Natal, Brasil.
{ claudiotrindade.cc, arthurecassio, jorgepereirasb,
judf.oliveira, neliocacho, thaisbatista,
evertonranielly}@gmail.com

Frederico Lopes, Marcos Maia, Stefano Loss
Instituto Metrópole Digital
Universidade Federal do Rio Grande do Norte
Natal, Brasil.
{ fred.lopes, marcosmaia422,
momoloss10}@gmail.com

Resumo— O FIWARE é uma plataforma de middleware desenvolvida por um projeto da Comissão Europeia para promover a construção de um ambiente para Internet do Futuro. Atualmente o FIWARE está sendo usado no projeto SmartMetropolis como a plataforma de middleware para promover o desenvolvimento de aplicações para cidades inteligentes, através de uma parceria com a prefeitura da cidade de Natal. Nesse contexto, esse trabalho tem como objetivo apresentar a implementação de uma aplicação utilizando o FIWARE para monitoramento da velocidade média das vias e também das câmeras da cidade de Natal.

Palavras-chave: *Aplicação; Desenvolvimento; Cidade Inteligente; Fiware.*

I. INTRODUÇÃO

O FIWARE é uma plataforma de middleware para Internet do Futuro[1] desenvolvida através de um parceria público privada (FI-PPP¹) lançada pela Comissão Europeia². A plataforma tem como principal objetivo oferecer suporte para o desenvolvimento de aplicações e serviços inteligentes para a Internet do Futuro em diversos setores, tais como cidades inteligentes, logística, energias renováveis, transportes sustentáveis, etc. O suporte ao desenvolvimento das aplicações e serviços se dão através do conjunto de APIs abertas, genéricas e extensíveis. As APIs são implementadas através de componentes denominados de habilitadores genéricos (*GEs – Generic Enablers*).

Os habilitadores genéricos estão divididos e organizados dentro da arquitetura do FIWARE através de capítulos técnicos. Os capítulos técnicos são conjuntos de funcionalidades responsáveis por prover determinados serviços dentro da plataforma, como segurança, infraestrutura, etc.

Atualmente, a release 4 da especificação do FIWARE possui 7 capítulos técnicos, que são: 1) *Cloud Hosting*; 2) *Data/Context Management*; 3) *Internet of Things (IoT) Services Enablement*; 4) *Applications, Services and Data Delivery*; 5) *Security*; 6) *Interface to Networks and Devices (I2ND) Architecture*; 7) *Advanced Web-based User Interface*;

O FIWARE vem sendo adotada para dar suporte ao desenvolvimento de cidades inteligentes por 31 cidades, entre países como Finlândia, Dinamarca Bélgica, Portugal, Itália, Espanha e Brasil [2].

O FIWARE também foi a plataforma de middleware escolhida para dar suporte ao desenvolvimento do ambiente de cidade inteligente no projeto SmartMetropolis³ da Universidade Federal do Rio Grande do

Norte (UFRN), que tem como principal objetivo promover o desenvolvimento de aplicações e infraestrutura voltadas para criação de cidades inteligentes. A escolha pelo FIWARE se deu através de um estudo que realizou um levantamento dos requisitos das principais plataformas de middleware da literatura e indústria, e apontou o FIWARE como a plataforma que atende ao maior conjunto de requisitos dentre os requisitos levantados.

Atualmente, a cidade de Natal, através de uma parceria firmada entre a prefeitura e a UFRN, está se juntando as demais cidades que utilizam o FIWARE como plataforma de middleware para cidades inteligentes. Essa parceria permitiu inserir a cidade de Natal no grupo de trabalho da iniciativa do IEEE Smart Cities [3].

Dentro desse contexto, esse trabalho tem como principal objetivo apresentar a implementação de uma aplicação utilizando a plataforma FIWARE para monitoramento da velocidade média das principais vias e câmeras da cidade de Natal.

II. HABILITADORES GENÉRICOS

O FIWARE possui diversos habilitadores genéricos que estão agrupados de acordo com suas funcionalidades. Dentre eles, o conjunto mínimo de GEs que possibilitam a construção de aplicações utilizando o FIWARE é formado pelos GEs, *Orion Context Broker* e *Wirecloud*.

O *Orion Context Broker* é o GE que faz parte do capítulo técnico de *Data/Context Management*, e consiste em um *broker* para geração de contexto que utiliza o padrão Publish/Subscriber [4]. Esse broker é responsável por enviar e receber dados de contexto das aplicações que utilizam o FIWARE.

O *Wirecloud* é o GE que faz parte do capítulo técnico de *Applications, Services and Data Delivery*, e é responsável por fornecer um ambiente para desenvolvimento de aplicações Mashup [5], disponibilizando um ambiente para desenvolvimento de aplicações que integram dados heterogêneos, lógica de aplicação e componentes de interface gráfica. Esse GE permite que aplicações sejam construídas através de simples conexões entre componentes, denominados de *widgets* e *operators*. Os componentes são desenvolvidos utilizando Javascript, HTML e CSS.

III. APLICAÇÃO

Atualmente, a secretaria municipal de mobilidade urbana da cidade de Natal (SEMOB) possui um portal^{iv} público para consulta de informações sobre a velocidade média das vias e monitoramento das câmeras espalhadas nos principais cruzamentos das avenidas da cidade. No

entanto, o portal não permite o uso dessas informações por outras aplicações que necessitem desses dados para geração de novas informações, como, por exemplo, o caso de uma aplicação que estima o grau de poluição de uma determinada região. Essa aplicação poderia utilizar da informação da concentração de veículos em determinada região para identificar o grau de poluição que está sendo gerado por aqueles veículos. Para contornar esse problema, nesse trabalho iremos apresentar uma versão desse portal desenvolvido utilizando a infraestrutura e GEs do FIWARE. A figura 1 apresenta a arquitetura da aplicação, onde há um motor para extração de dados desenvolvido utilizando a ferramenta de teste SeleniumHQ^v, para extração dos dados de velocidade média das vias do portal da SEMOB e os enviar para o ambiente do FIWARE através do GE Orion, permitindo, assim, que as aplicações de monitoramento da velocidade média das vias e das câmeras desenvolvidas dentro do Wirecloud possam interagir com os dados providos pelo Orion.



Figura 1 Arquitetura da aplicação

IV. IMPLEMENTAÇÃO

A implementação da aplicação está dividida em duas etapas. A primeira consiste no desenvolvimento do motor de extração de dados do portal da SEMOB para o Orion, o qual foi desenvolvido utilizando a linguagem de programação JAVA. Na parte do extração dos dados foi utilizada a linguagem XPath^{vi} para identificação e seleção dos dados dentro do portal, e a biblioteca NGSi^{vii} para envio das informações para o Orion. O motor de extração fica rodando em um servidor na nuvem e executa o processo de extração dos dados a cada intervalo de tempo configurado. A segunda etapa consiste no desenvolvimento das aplicações dentro do Wirecloud. Nessa etapa, dois componentes foram desenvolvidos: (i) um *widget* (SmartMetropolis_transito_0.0.1.wgt) desenvolvido em HTML, CSS e Javascript, para consultar as informações de velocidade média das vias no Orion, através da biblioteca NGSi do próprio Wirecloud, e as exibirem no formato de tabela no dashboard; (ii) um *operator* (SmartMetropolis_cameras_0.0.1.wgt) desenvolvido em Javascript para consultar e transformar os dados das câmeras no Orion, e as exibirem por meio de um *widget* de mapa (MapView^{viii}). A figura 2 apresenta a conexão entre o *operator* das câmeras e o *widget* do mapa.

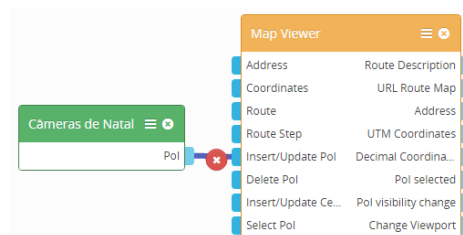


Figura 2 Conexão entre os componentes do Wirecloud

A figura 3 apresenta a aplicação rodando no Wirecloud. De lado temos o *widget* que exibe os dados sobre a velocidade média das vias, e do outro lado o *widget* de mapa que exibe os dados das câmeras coletados pelo *operator* das câmeras.

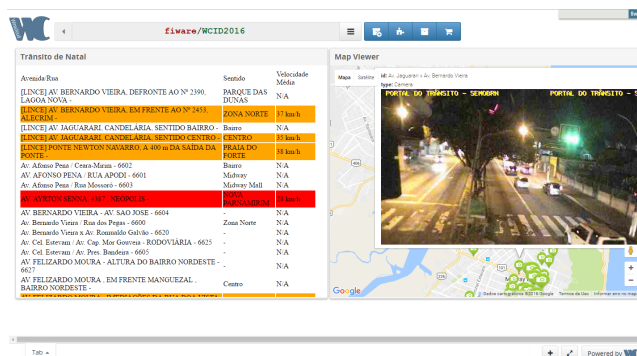


Figura 3 Aplicação no Wirecloud

V. CONCLUSÃO

Esse artigo apresentou a implementação de uma aplicação utilizando a plataforma FIWARE para monitoramento da velocidade média das vias e câmeras da cidade de Natal. Mais detalhes sobre a implementação da aplicação está em disponível na página do projeto^{ix}.

REFERÊNCIAS

- [1] Jianli Pan, Subharthi Paul, Raj Jain (2011) A survey of the research on Future Internet architectures. Communications Magazine, vol. 49, no. 7. pp. 26-36.
- [2] Cities from seven different countries boost FIWARE as standard for Smart Cities. Disponível em <https://www.fiware.org/news/cities-from-seven-different-countries-boost-fiware-as-standard-for-smart-cities/>. Acesso em 30/09/2016.
- [3] IEEE Smart Cities Initiative. Disponível em <http://smartcities.ieee.org/home/ieee-smart-cities-initiative.html>. Acesso em 30/09/2016.
- [4] Patrick Th. Eugester, Pascal A. Felber, Rachid Guerraoui, Anne-Marie Kermarrec (2003). The many faces of publish/subscribe. Journal ACM Computing Surveys (CSUR), vol. 35, no. 2.
- [5] Jin Yu, Boualem Benatallah, Fabio Casati (2008) Understanding Mashup Development. IEEE Internet Computing, vol. 12, no. 5.

ⁱ <https://www.fi-ppp.eu>

ⁱⁱ http://ec.europa.eu/index_pt.html

ⁱⁱⁱ <http://smartmetropolis.imd.ufrn.br/>

^{iv} <http://semobr.gtrans.com.br/portaltransito/index.php>

^v <http://www.seleniumhq.org/>

^{vi} <https://www.w3.org/TR/xpath>

^{vii} <https://www.dropbox.com/s/r9wpsvrsrlcqt/NGSI-10.zip?dl=0>

^{viii} <https://github.com/Wirecloud/map-viewer-widget>

^{ix}

http://consiste.dimap.ufrn.br/smartmetropolis/monitoramento_de_transito