

SISTEMA MÓVEL PARA COMUNICADOS GEORREFERENCIADOS

Júnior Bevilaqqua Mendonça¹

Thiago Grespi Goulart²

Ely Fernando do Prado³

Resumo

Diariamente, milhares de pessoas transitam pelas ruas das cidades e encontram diversos tipos de problemas, como buracos nas ruas, lixo em lugar inapropriado, entre outros. Muitos desses problemas não são resolvidos simplesmente pela falta de comunicação com os responsáveis, pois a população, na maioria dos casos, não sabe a quem informar. O sistema móvel multiplataforma Nossa Cidade busca preencher essa lacuna, ao facilitar essa comunicação. O objetivo é o desenvolvimento de um sistema móvel multiplataforma capaz de permitir o envio de comunicados com foto e georreferenciamento, no qual os cidadãos poderão enviar os problemas encontrados diretamente para o órgão responsável pela resolução. O projeto foi desenvolvido por meio de entrevistas junto às partes interessadas (cidadãos e órgãos responsáveis) a fim de validar a viabilidade do projeto e os requisitos do sistema. Realizaram-se, também, estudos sobre ferramentas de desenvolvimento e codificação de *software*. O resultado desse projeto é uma aplicação móvel, disponível para Android, Windows Phone e iOS, capaz de enviar comunicados com mídias (fotos, vídeos e localização geográfica) diretamente aos órgãos responsáveis pelas cidades. Com isso, criou-se uma plataforma tecnológica que facilita a comunicação entre as partes anteriormente citadas, diminuindo o tempo entre a detecção, comunicação e solução de problemas das cidades. Esse projeto vai ao encontro dos processos de adequação dos municípios ao modelo de cidades inteligentes, os quais forneceram diversos benefícios à sociedade.

Palavras-chave: Aplicativo. Cidades Inteligentes. Comunicação. Multiplataforma.

Abstract

Every day, thousands of people walk through city streets and find different types of problems, such as holes in the streets, garbage in an inappropriate place, among others. Many of these problems are not solved simply by the lack of communication with the responsible ones as the population, in most cases, does not know to whom to inform. The multiplatform mobile system Nossa Cidade seeks to fill this gap by facilitating this communication through the permission of submissions of georeferenced reports or denunciations. The aim of this project is the development of a mobile multiplatform system that allows sending messages with photos and

¹ Graduando em Análise e Desenvolvimento de Sistemas pela Fatec Dr Thomaz Novelino – Franca/SP. Endereço eletrônico: [...].

² Graduando em Análise e Desenvolvimento de Sistemas pela Fatec Dr Thomaz Novelino – Franca/SP. Endereço eletrônico: thiago.grespi90@gmail.com.

³ Mestre em Ciência da Computação. Endereço eletrônico: elyprado@gmail.com

geotagged through which citizens can talk about the problems they have faced directly to the responsible bodies. The project was developed through interviews with interested parties (citizens and responsible bodies) in order to validate the project feasibility and system requirements. Studies on software development and coding tools were also carried out. The result of this project is a mobile application, available for Android, Windows Phone and iOS, capable of sending media releases (photos, videos and geographic location) directly to the responsible bodies of the cities. With this, a technological platform was created that facilitates the communication between the parties previously mentioned, reducing the time between the detection, communication and solution of problems in the cities. This project is in line with the processes of adaptation of municipalities to the model of intelligent cities, which have provided various benefits to society.

Keywords: *Application. Communication. Multiplatform. Smart Cities.*

1 Introdução

Buracos no asfalto, terrenos baldios com mato alto e lixo, desperdício de água, focos do mosquito transmissor da dengue e incontáveis outros problemas são vistos todos os dias pela população durante seu trajeto pela cidade, indo ao trabalho, voltando da faculdade ou simplesmente passeando. Muitos desses problemas não são solucionados por falta de comunicação eficaz junto aos responsáveis pela administração da cidade, pois muitas vezes a população não sabe a quem reclamar e nem mesmo como informar os órgãos competentes. Os municípios, que por diversas vezes chegam a enfrentar dificuldades de locomoção, de sanidade, etc., pelos problemas encontrados nas ruas da cidade ficam por vezes de mãos amarradas ao deparar com um desses problemas. As informações de contato para que possam informar os problemas encontrados nem sempre estão disponíveis, não aparecem com frequência na mídia tradicional, o que dificulta que os próprios responsáveis por manter a limpeza e funcionalidade da cidade tomem conhecimento dos eventos e ocorridos que prejudicam o bem-estar dos que ali vivem e transitam.

Esta situação gera problemas de locomoção para o transporte público, que necessita locomover-se pela cidade, possibilitam a criação e proliferação de animais peçonhentos e que espalham doenças na população, diminuem a efetividade de programas de conscientização, entre outros. Todas estas consequências geram e

umentam os gastos e despesas de uma cidade, tornando a operação cada vez mais cara e ineficiente.

Portanto justifica-se a criação de um sistema que permita que munícipes façam comunicados sobre problemas encontrados na cidade. Os comunicados são enviados diretamente para a ouvidoria da prefeitura, que é responsável por atender as solicitações da população e encaminhá-los ao setor responsável, para que haja uma avaliação de como e em quanto tempo o problema poderá resolvido.

O projeto tem como objetivo desenvolver um sistema móvel multiplataforma capaz de permitir o envio de comunicados com foto e georreferenciamento. Esse aplicativo tem como requisito a funcionalidade de consultar e acompanhar os comentários sobre o comunicado realizado. No escopo do projeto também está incluso o desenvolvimento de um sistema web onde os usuários poderão consultar, acompanhar e comentar sobre os comunicados abertos, sendo esses usuários pessoas pertencentes à comunidade, prefeituras, organizações, entre outras.

Foram aplicados questionários com o público alvo a fim de levantar e validar os requisitos do sistema, além de estudos sobre as ferramentas de desenvolvimento e codificação do *software*. Os conceitos teóricos que envolvem a pesquisa, as ferramentas utilizadas e a metodologia são descritos nas próximas seções.

No capítulo 2 é apresentado o referencial teórico, descrevendo os temas: PHP, Cordova e Ionic

2 Referencial teórico

Segundo ONU (2013) é estimado que cerca de 50% da população mundial vive em áreas urbanas, e até 2050 esse número vai crescer para 70%. O resultado desse crescente número de pessoas vivendo nos centros urbanos são cidades cada vez mais densas, trazendo grande desafio para os administradores, em questões relacionadas à tráfego, acesso a recursos básicos, aumento dos custos e inúmeros outros aspectos. Isso nos levou a implementar uma abordagem que reduza esses custos, facilitando assim a disponibilidade de recursos básicos à população (água, esgoto, saúde, educação, segurança e serviços de suporte). Essa abordagem usa como base os sistemas de informação, usados para otimizar a tomada de decisão e implementação de novas técnicas de *design*, financiamento, governança e operação

dos serviços e infraestrutura urbana. Denominado Cidades Inteligentes, esse termo começou a ser usado no final da década de 1990, porém somente a partir de 2004 vem sendo difundido, por conta do grande avanço dos meios de comunicação e a facilidade da população em acessar novas tecnologias.

Os exemplos de tecnologias que podem ser utilizados no conceito de cidades inteligentes são numerosos, tais como: rede de sensores sem fio, computação em nuvem, *internet* das coisas, sistemas de informação geográficos, que podem ser otimizados quando bem sincronizados com recursos humanos (população) e fornecem dados que facilitam a administração das cidades. Diversas tecnologias já estão em uso dentro do conceito ligado a cidades inteligentes, na área de transportes, prédios e redes inteligentes e até em monitoramento ambiental. Implementar conceitos de cidades inteligentes requer o uso de Tecnologia como ferramenta auxiliar, que providenciam benefícios aos municípios.

2.1 PHP

A linguagem de programação PHP em seu início tinha um acrônimo recursivo para Personal Home Page Tools. Esta foi criada no outono de 1994 por Rasmus Lerdorf. Essa linguagem era formada por um conjunto de *scripts* escritos em linguagem C, voltado para a criação de páginas dinâmicas. Em 1995 o código fonte do PHP foi liberado para que mais desenvolvedores pudessem ajudar no melhoramento da linguagem. Duas pessoas marcaram muito a história do PHP: Andi Gutmans e Zeev Suraski, que reescreveram todo o código-fonte dando início ao PHP 3 disponibilizado oficialmente em 1998. No final de 1998, o PHP já estava presente em cerca de 10% dos domínios da *internet*, neste mesmo período o significado da sigla PHP mudou para: Hypertext Preprocessor. Zeev e Andi começaram a trabalhar em uma reescrita do núcleo do PHP, para melhorar o seu desempenho em aplicações mais complexas. O nome foi rebatizado para Zend Engine (Zeev + Andi), a partir disso a linguagem só vem sendo aprimorada, na época da escrita desse artigo o PHP está em sua versão 7.

Atualmente estima-se que o PHP está sendo utilizado em mais de 80% dos servidores existentes, tornando-a disparadamente a linguagem mais utilizada para

desenvolvimento *web*. Entre as maiores plataformas estão: YouTube, Wikipédia, Facebook.

A grande adoção da linguagem PHP acontece pois esta é uma linguagem *server side*, ou seja, uma linguagem que é executada apenas no servidor, diminuindo os gastos de recursos do cliente. Sua licença é gratuita e o código-fonte aberto sendo disponível para o público para aprimorá-lo. É também compatível com vários bancos de dados, entre eles estão MySQL, PostgreSQL, Sybase, Oracle, SQL Server e muitos outros.

O PHP pode ser executado em vários sistemas operacionais entre eles estão o Linux, Unix e Windows, havendo poucas diferenças nos diferentes sistemas. O PHP organiza o mundo real em conjuntos de objetos, incorporando e manipulando dados assim sendo classificada uma linguagem orientada a objetos. (NIEDERAUER, 2004), (DALL'OGGIO, 2015).

2.2 Cordova

Apache Cordova é um *framework* para desenvolvimento de aplicações híbridas multiplataforma para dispositivos móveis que, através de suas APIs, permite o acesso a funções nativas do aparelho, como sensores, câmera, GPS e acelerômetro.

Começou a ser desenvolvido no ano de 2008 pela empresa Nitobi em um evento chamado iPhoneDevCamp, realizado em São Francisco e ganhou destaque entre os desenvolvedores alguns anos depois. Em 2011, a Adobe Systems comprou a Nitobi, renomeou o *framework* para PhoneGap e doou seu código para Apache Software Foundation, iniciando assim o projeto chamado de Apache Cordova.

Atualmente é utilizado por diversos desenvolvedores e grandes empresas, alguns exemplos são: o aplicativo oficial do *wikipédia*, o *facebook* utiliza uma bifurcação da versão do *framework* em seu SDK Mobile, a plataforma de desenvolvimento Worklight da IBM e o Halo Waypoint, um jogo virtual integrado com o XBOX-Live para iOS e Android.

A ferramenta tem como vantagens a facilidade de manutenção, o baixo custo de produção, o desenvolvimento para diversas plataformas, como Android, iOS e Windows Phone, a partir de um único código fonte e o código aberto. Porém, em

alguns sistemas operacionais não existe suporte para algumas APIs e existe a necessidade de ter um navegador instalado no dispositivo. Tais características fazem com que o desenvolvimento seja simples e com uma curva de aprendizado pequena e rápida. (DUMONT, 2018), (PALUDO, 2018)

2.3 Ionic

Lançado em 2013, o Ionic⁴ é um *framework* de desenvolvimento, também conhecido como *SDK (Software Development Kit)* focado em componentes visuais para aplicativos para dispositivos móveis. O objetivo do Ionic é facilitar o desenvolvimento dessas aplicações, através da disponibilização de ferramentas e componentes HTML, CSS e funcionalidades JavaScript pré-estabelecidas, que evitam que o programador tenha que implementar todos os elementos do zero.

O Ionic tem como dependência o *framework* Cordova, portanto para rodar nos dispositivos, este último precisa estar previamente instalado. O Ionic não deixaria de funcionar totalmente em um ambiente sem o Cordova, porém todo o resultado do desenvolvimento poderia ser visto apenas através de um *browser* de um computador ou de um dispositivo móvel, ou seja, o Cordova é um requisito caso o desenvolvedor queira instalar e executar a aplicação direto no dispositivo móvel, sem o uso direto de um navegador.

Outra característica do Ionic é o fato de ser baseado no Angular, que é um *framework javascript* focado em produtividade. Por ter adoção mundial, o Angular traz uma grande vantagem ao Ionic: programadores que já conhecem e utilizam o Angular podem utilizar a base de conhecimento que já possuem. (GOIS, 2017)

3 Desenvolvimento

Este capítulo apresenta os métodos utilizados para desenvolver este trabalho, incluindo desde a análise de requisitos até a codificação.

⁴ Site oficial do Ionic: <<https://ionicframework.com/>>

3.1 Entrevistas

Todo produto, para ser desenvolvido, passa por inúmeras fases antes de entrar de fato na produção. Uma dessas fases é a identificação das partes interessadas, onde todos que têm interesse no projeto, seja financeiro, estratégico ou qualquer outro, são ouvidos pelos desenvolvedores do produto. Nesta fase, as partes interessadas (empresas, pessoas, organizações não governamentais, representantes de cidades, estados e países, etc.) são submetidos à uma série de perguntas, entrevistas, reuniões e conversas. Essas técnicas de levantamento de informação são implementadas para que o produto a ser desenvolvido atenda às necessidades de seus clientes, que são as partes interessadas. Sem esse tipo de ação, as chances de um produto não atender às expectativas dos desenvolvedores são grandes. Para tanto, todas as partes interessadas no projeto foram entrevistadas, a fim de levantar os requisitos do produto e atender às necessidades dos usu[ários]. Para isto, uma pesquisa com uma pequena parte do nosso público alvo através de uma ferramenta de perguntas e respostas *online*, na qual levantamos 176 respostas. Logo após, várias reuniões, entrevistas e workshop foram realizados com os funcionários da ouvidoria e de secretarias de Franca-SP.

Nestas ações, coletamos respostas chave de como a plataforma deve se comportar, quais recursos devem apresentar e quais funcionalidades deve possuir.

3.2 Codificação

Para o desenvolvimento do aplicativo foi utilizado o *framework* Ionic, no qual o principal objetivo é tornar o HTML 5 multiplataforma e responsivo, podendo ser executado em qualquer dispositivo que possua um navegador da *web* instalado. Essa ferramenta faz com que o tempo de desenvolvimento seja reduzido.

No uso do Ionic para a construção do aplicativo Nossa Cidade, foram utilizados diversos componentes pré-estabelecidos no *frameork*. Esses componentes são pré-definidos e estilizados pelo Ionic, portanto apenas adequamos alguns deles ao padrão de cores e elementos utilizados no projeto, como campos de texto, botões, janelas do aplicativo, etc.

Alguns desses componentes do Ionic precisaram receber um tratamento de estilos que não é contemplado pelo *framework*, e para essa estilização usamos uma linguagem própria para isso, o CSS (*Cascade Style Sheet*).

A Folha de Estilos em Cascata (*Cascade Stylish Sheet - CSS*) é um arquivo no qual os programadores trabalham no *design* e no visual da aplicação, adequando e melhorando a aplicação para que, ao interagir com o aplicativo, o usuário consiga navegar de forma simples e fluída.

Para complementar a boa interação do usuário com o aplicativo móvel, fez-se uso da linguagem de programação JavaScript, que dispõe de recursos focados a suportar essa interatividade.

Como ferramenta para codificação foi utilizado um *software* chamado Visual Studio Code. Disponibilizado pela Microsoft, o comumente chamado VS Code possibilita uma gama de ferramentas para o desenvolvimento de *software*. Essas ferramentas vão desde *plugins* para auxiliar produtividade, verificação de padrão de código, até um terminal para que comandos do sistema operacional, de outros *softwares* e até de linguagens de programação possam ser executados.

4 Resultados

O projeto teve como resultado o desenvolvimento de um aplicativo para dispositivos móveis capaz de enviar informações diretamente para a prefeitura e os órgãos responsáveis pela resolução do problema relatado.

Na Figura 1 é apresentada a prototipação das telas utilizadas para o desenvolvimento da aplicação:

Figura 1 – Home - iniciar comunicado



4.1 Aplicativo para dispositivos móveis

Nesta seção são apresentadas as interfaces do aplicativo para dispositivos móveis.

Primeiramente tem-se a tela que está apresentada na Figura 1. Esta é a tela inicial do aplicativo, na qual existe um cabeçalho com um botão que aciona o menu lateral. Logo abaixo do cabeçalho encontra-se uma seção explicativa, para que o usuário que ainda não realizou nenhum comunicado saiba o que pode ser feito e por onde começar. Quando este usuário já tiver criado um ou mais chamados, uma lista com esses elementos e um resumo de cada um deles será exibida neste local. No canto inferior direito localiza-se um botão que inicia o fluxo de criação de um comunicado.

Figura 2 – Capturar imagem do problema



Logo após acionar o botão que inicia o processo de criação do comunicado exibido na Figura 1, a aplicação captura a localização atual do dispositivo via GPS e abre a câmera para que o usuário possa capturar uma imagem do problema a ser relatado. Esse processo é exibido na Figura 2.

Figura 3 – Selecionar categoria do problema



Após a captura da imagem exibida na Figura 2, o aplicativo solicita que o usuário selecione uma categoria para o problema evidenciado. Esta seleção da categoria é uma ação importantíssima, visto que, ao final do processo de criação do chamado, o problema relatado é enviado diretamente para a secretaria responsável por aquele tipo de problema. Aqui o usuário seleciona a categoria do problema e confirma, clicando no botão Próximo.

Na Figura 4 encontra-se o resumo do que já foi feito pelo usuário até aqui, dentro do processo de criação de um comunicado. O aplicativo exibe a categoria do problema e a imagem capturada, além de disponibilizar um campo para que o munícipe possa digitar uma mensagem personalizada.

Assim que esses três passos forem concluídos, o botão Finalizar fica liberado para ser clicado. Quando isso acontece, a aplicação salva todas as informações, e envia as evidências do problema relatado para a secretaria que está relacionada à categoria previamente selecionada.

Figura 4 – Escrever mensagem e finalizar comunicado



Figura 5 – Comunicado finalizado



Após salvar todas as informações importantes no banco de dados e enviar o comunicado para o órgão responsável pela resolução, uma tela de agradecimento e confirmação da criação do comunicado é exibida. No rodapé desta página fica o botão que leva o usuário a uma página com a listagem de todos os comunicados já criados por ele.

Figura 6 – Meus comunicados



A Figura 6 exibe a listagem de todos os comunicados já feitos por este usuário. Dentro do *card* que exibe cada comunicado, algumas informações como a data de realização do comunicado e a quantidade de mensagens trocadas com o órgão responsável.

Figura 7 – Detalhes e Chat



Ao clicar em alguns dos *cards* exibidos, é mostrado ao usuário todos os detalhes do comunicado selecionado. A data de criação, uma previsão para a resolução do problema, a quantidade de mensagens trocadas e um indicativo de resolução do problema são exibidos logo abaixo da imagem que fora capturada no momento de criação do comunicado.

Abaixo dessas informações, as mensagens trocadas pelo chat são exibidas em ordem cronológica, com um campo de texto para o envio de novas mensagens no rodapé da página.

Durante o desenvolvimento do projeto como um todo, ou seja, este artigo escrito e o desenvolvimento do aplicativo, acumulamos grande aprendizado sobre vários aspectos relacionados ao setor de desenvolvimento de software. Melhoramos competências e entendemos que, ao desenvolver um software, a prioridade é identificar os reais problemas e as necessidades das organizações e seus colaboradores, e propor soluções que se enquadram nos diversos aspectos para que o problema seja resolvido de forma eficaz.

Considerações finais

Conforme os objetivos do projeto de desenvolver um sistema móvel multiplataforma capaz de permitir o envio de comunicados com foto e georreferenciamento, pode-se observar que tais objetivos foram cumpridos. O projeto vem ao encontro das necessidades sociais e econômicas que afetam as cidades dentro do conceito de cidades inteligentes, quando pode utilizar recursos tecnológicos para melhoria da prestação do serviço público e qualidade de vida das pessoas. A utilização da aplicação pode diminuir o tempo gasto entre a observação do problema e sua consequente solução pelos órgãos públicos.

Os próximos passos estão na ampliação de funcionalidades do *software* desenvolvido, visando atender diferentes necessidades para o mesmo público. Todos os processos de análise de requisitos, prototipação de interfaces e testes de usabilidade serão realizados, para que o desenvolvimento e codificação do *software* aconteça da forma planejada.

Agradecimentos

Agradecemos primeiramente a Deus, por nos dar forças nesta jornada e ajudar-nos em cada momento. Agradecemos também aos familiares, que além de todo amor dado, foram forte alicerce nos momentos difíceis.

Referências

Apps Híbridas com Cordova e Ionic, 27/03/2017. Disponível em <<https://medium.com/@lfv89/nessa-s%C3%A9rie-divida-em-3-partes-vou-falar-um-pouco-mais-a-fundo-sobre-desenvolvimento-h%C3%Adbrido-914f22453c83>>. Acesso em 26.mar.2018.

DALL'OGGIO, Pablo. **PHP Programando com orientação a Objetos**. Novatec Editora, 2018.

DUARTE, Fábio. **Cidades inteligentes: inovação tecnológica no meio urbano**. **São Paulo em Perspectiva**, v. 19, n. 1, p. 122-131, 2005.

DUMONT, AARON LESBÃO. **Relatório de estágio supervisionado em desenvolvimento de aplicações híbridas utilizando o Apache Cordova: UM CASE EM GAMEFICAÇÃO**.

FALCÃO, Ana Gabrielle Ramos; BAPTISTA, C. de S.; MENEZES, LC de. **Crowd4City**: utilizando sensores humanos como fonte de dados em cidades inteligentes. In: Proceedings of the 8th Brazilian Symposium on Information Systems. 2012. p. 144-149.

GAMA, Kiev; ALVARO, Alexandre; PEIXOTO, Eduardo. **Em direção a um modelo de maturidade tecnológica para cidades inteligentes**. Simpósio Brasileiro de Sistemas de Informação, VIII, 2012.

GOIS, Adrian. **Ionic Framework: construa aplicativos para todas as plataformas mobile**. São Paulo: Casa do Código, 2017

NIEDERAUER, Juliano. **Desenvolvendo websites com PHP**. São Paulo: Novatec, 2004.

ONU: mais de 70% da população mundial viverá em cidades até 2050, 15/04/2013. Disponível em <<https://nacoesunidas.org/onu-mais-de-70-da-populacao-mundial-vivera-em-cidades-ate-2050/>> Acessado em 30.mar.2018.

PALUDO, Gabriela Cristina. **Easycore**: um protótipo para facilitar a leitura de partituras em dispositivos móveis.

TOLEDO, Jan Miszura; DE DEUS, Gilcimar Divino. **Desenvolvimento em Smartphones-Applicativos Nativos e Web**. 2012.