

**INSTITUTO FEDERAL DE MINAS GERAIS
CAMPUS SÃO JOÃO EVANGELISTA**

JOEL KENNEDY DA SILVA; JOSÉ VICENTE JÚNIOR DOS SANTOS SANTIAGO

**DESENVOLVIMENTO DE UM SISTEMA DE INFORMAÇÃO COLABORATIVO
PARA PROBLEMAS INFRAESTRUTURAIS DE UMA CIDADE**

SÃO JOÃO EVANGELISTA

2018

JOEL KENNEDY DA SILVA; JOSÉ VICENTE JÚNIOR DOS SANTOS SANTIAGO

**DESENVOLVIMENTO DE UM SISTEMA DE INFORMAÇÃO COLABORATIVO
PARA PROBLEMAS INFRAESTRUTURAIS DE UMA CIDADE**

Trabalho de conclusão de curso apresentado ao Instituto Federal de Minas Gerais - *Campus* São João Evangelista como exigência parcial para obtenção do título de Bacharel em Sistemas de Informação.

Orientador: Prof. Me. Rosinei Soares de Figueiredo

Coorientador: Prof. Me. Bruno de Souza Toledo

Coorientador: Prof. Me. Edson Batista de Sena

SÃO JOÃO EVANGELISTA

2018

FICHA CATALOGRÁFICA

S581d Silva, Joel Kennedy da; Santiago, José Vicente Júnior dos Santos.
2018

Desenvolvimento de um sistema de informação colaborativo para problemas infraestruturais de uma cidade. / Joel Kennedy da Silva; José Vicente Júnior dos Santos Santiago. – 2018.
49f; il.

Trabalho de Conclusão de Curso (Bacharelado em Sistemas de Informação) – Instituto Federal de Educação, Ciência e Tecnologia de Minas Gerais – Campus São João Evangelista, 2018.

Orientador: Me. Rosinei Soares de Figueiredo.
Coorientador: Me. Bruno de Souza Toledo.
Coorientador: Me. Edson Batista de Sena.

1. Aplicação Web Progressiva. 2. Problemas Urbanos. 3. Infraestrutura.
I. Silva, Joel Kennedy da. II. Santiago, José Vicente Júnior dos Santos.
III. Instituto Federal de Educação, Ciência e Tecnologia de Minas Gerais – Campus São João Evangelista. IV. Título.

CDD 658.4038011

Elaborada pela Biblioteca Professor Pedro Valério

Instituto Federal de Educação, Ciência e Tecnologia de Minas Gerais
Campus São João Evangelista

Bibliotecária Responsável: Rejane Valéria Santos – CRB-6/2907

**JOEL KENNEDY DA SILVA; JOSÉ VICENTE JÚNIOR DOS SANTOS
SANTIAGO**

**DESENVOLVIMENTO DE UM SISTEMA DE INFORMAÇÃO COLABORATIVO
PARA PROBLEMAS INFRAESTRUTURAIS DE UMA CIDADE**

Trabalho de conclusão de curso apresentado ao Instituto Federal de Minas Gerais - *Campus* São João Evangelista como exigência parcial para obtenção do título de Bacharel em Sistemas de Informação.

Aprovado em: 28/11/2018

BANCA EXAMINADORA



Orientador: Prof. Me. Rosinei Soares de Figueiredo
Instituto Federal de Minas Gerais - *Campus* São João Evangelista



Coorientador: Prof. Me. Bruno de Souza Toledo
Instituto Federal de Minas Gerais - *Campus* São João Evangelista



Convidado: Prof. Dr. Bruno Pellizzaro Dias Afonso
Instituto Federal de Minas Gerais - *Campus* São João Evangelista



Convidado: Prof. Esp. Fernando Henriques Mafra
Instituto Federal de Minas Gerais - *Campus* São João Evangelista

SÃO JOÃO EVANGELISTA

2018

AGRADECIMENTOS

Agradecemos primeiramente à Deus, que nos deu o dom da vida e nos abençoa todos os dias com o seu amor infinito. À nossa família, que nos apoiaram muito com palavras de incentivo. Obrigado IFMG-SJE pela oportunidade de fazermos o curso Bacharelado em Sistemas de Informação. Agradeço por nos oferecer professores incríveis, um ambiente de estudo agradável e muitos estímulos para participar de atividades acadêmicas. Somos gratos não só aos professores, mas também à direção, ao pessoal do administrativo, da limpeza e demais colaboradores da instituição. Aos amigos, meu muito obrigado, por torcerem e vibrarem com a nossa conquista.

RESUMO

O presente trabalho trata-se de um Sistema de Informação Colaborativo de problemas infraestruturais de uma cidade. O uso de *softwares* em rede para compartilhamento de opiniões, notícias, informações, entretenimento e demais itens, tem sido algo cada vez mais presente na vida do ser humano. Partindo dessa afirmativa, o referido trabalho associou o desenvolvimento de um aplicativo à busca de soluções para os problemas de infraestrutura encontrados no dia-a-dia dos cidadãos que residem nos meios urbanos, pois, até então não havia nenhum programa com tal finalidade específica. A metodologia aplicada para seu desenvolvimento foi de caráter descritivo, que objetiva descrever e fazer uma análise do tema em estudo. A pesquisa aplicada teve caráter qualitativo, considerando que toda metodologia descritiva é qualitativa e que a mesma visou fazer um levantamento de dados baseada na compreensão de comportamentos e opiniões, buscando garantir a viabilidade do *software*. Para o Ambiente de Desenvolvimento Integrado (IDE) utilizou-se uma versão de teste do *Sublime Text*, a Linguagem de Programação Hypertext Processor (PHP) para codificação, a Linguagem de Marcação de Hipertexto (HTML) e Folhas de Estilo em Cascata (CSS) para criação da interface gráfica. Para que o sistema pudesse ser versátil e acessado independente do dispositivo e lugar, o tipo de *software* adotado foi uma Aplicação Web Progressiva (PWA). Após o desenvolvimento do sistema, foram realizados testes no mesmo, objetivando encontrar possíveis erros. O aplicativo foi testado por cidadãos de São João Evangelista e região, tendo obtido resultados satisfatórios quanto à sua viabilidade.

Palavras-chave: Aplicação Web Progressiva. Problemas Urbanos. Infraestrutura.

ABSTRACT

The presente work is about a Collaborative Information System of infrastructure troubles of a city. The use of softwares in a network for sharing of opinions reviews, news and information, entertainment and other itens, has been every time more present in human life. Based on this statement, the referred work associated the development of a app the search for solutions to the problems of infraestructure found in the day-to-day of the citizens that reside in the urban means because, until then there was no program with such a specific finality. The methodology applied for their development was from descriptive character, that aims describe and do a analyze from the theme of study. The research applied has qualitative character, considering that all descriptive methodology is qualitative and that the same aimed to do a data collection based in the understanding of behaviors and opinions, searching ensure the viability of software. Was used the Integrated Development Environment (IDE), a test version of the Sublime text, the programming language Hypertext Processor (PHP) for codification, the Hypertext Markup Language (HTML) and Cascading Style Sheets (CSS) for creation of the graphic interface. For that the system could be versatile and accessed regardless of device and place, the tipe of adopted software was a Progressive Web Application (PWA). After the system development, was realized tests in the same, objectifying find possible mistakes. The app was tested by citizens of São João Evangelista and region, having obtained satisfying results about your viability.

Key-words: Progressive Web Application. Urban troubles. Infraestructure.

LISTA DE FIGURAS

Figura 1 - Tela de login	32
Figura 2 - Cadastrar	32
Figura 3 - Todas as notícias	33
Figura 4 - Criar notícia	34
Figura 5 - Minhas notícias	35
Figura 6 - Editar.....	35
Figura 7 - Seta	36
Figura 8 - Botão usuário	36
Figura 9 - Sobre	37
Figura 10 - Telas cobile conjunto 1	37
Figura 11 - Telas mobile conjunto 2.....	38

LISTA DE SIGLAS

APIs - Interface de Programação de Aplicativos

CSS - *Cascading Style Sheets*

DI - *Design de Interação*

GPS - Sistema de Posicionamento Global

HTML - *Hypertext Markup Language*

IFMG-SJE - Instituto Federal de Minas Gerais - *campus* São João Evangelista

MVC - *Model-View-Controller*

PHP - *Hypertext Preprocessor*

PWA - *Progressive Web Application*

SIG - Sistemas de Informação Geográfica

TI - Tecnologia da Informação

UML - *Unified Modeling Language*

URL - *Uniform Resource Locator*

W3C - *World Wide Web Consortium*

SUMÁRIO

1. INTRODUÇÃO.....	10
2. REFERENCIAL TEÓRICO	13
2.1. INFRAESTRUTURA URBANA	13
2.1.1. Infraestrutura Urbana no Brasil	14
2.2. ENGENHARIA DE SOFTWARE	15
2.2.1. Processo de Software	15
2.2.1.1. Análise e Elicitação de Requisitos	16
2.2.1.2. Projeto e Implementação.....	16
2.2.1.3. Validação de Software	17
2.2.2. Diagramas UML.....	18
2.3. APLICAÇÕES TRADICIONAIS	18
2.3.1. Aplicação Web.....	19
2.3.2. Aplicação Nativa	20
2.3.3. Aplicação Híbrida	20
2.4. APLICAÇÃO WEB PROGRESSIVA	21
2.4.1. Web App Manifest	22
2.4.2. Service Workers	22
2.5. SISTEMA DE INFORMAÇÃO GEOGRÁFICO	22
2.6. DESIGN DE INTERAÇÃO	23
2.7. FERRAMENTAS	24
2.7.1 Astah.....	24
2.7.2 CodeIgniter	25
2.7.3 Bootstrap.....	25
2.7.4 API do Google Maps	25
2.7.5 Servidor Web.....	26

2.7.6. MYSQL	26
2.8. TRABALHOS CORRELATOS	27
3. METODOLOGIA	28
3.1. NATUREZA DA PESQUISA	28
3.2. POPULAÇÃO E AMOSTRA	28
3.3. MÉTODOS E PROCEDIMENTOS	29
3.4. TRATAMENTO DE DADOS.....	30
3.5. PROTOTIPAÇÃO	31
4. RESULTADOS E DISCUSSÃO	39
4.1. ANÁLISE QUALITATIVA DO SOFTWARE.....	39
5. CONSIDERAÇÕES FINAIS	41
REFERÊNCIAS	42
APÊNDICES	48

1. INTRODUÇÃO

Em muitas cidades, existe uma grande quantidade de problemas infraestruturais. Tal problema advém de diversos fatores, em que destaca-se o grande aumento populacional a partir do século XX. Segundo Zmitrowicz e Angelis Neto (1997), infraestrutura urbana pode ser conceituada como um sistema técnico de equipamentos e serviços necessários ao desenvolvimento das funções urbanas, podendo estas funções serem vistas sob os aspectos sociais, econômicos e institucionais.

Pela visão de Hudson *et al.* (1997), o termo infraestrutura refere-se a todas as facilidades que fornecem serviços públicos essenciais de transportes, utilidades, energia, telecomunicações, áreas de lazer e de práticas de esportes e moradia. Seguindo a mesma definição, pode-se entender utilidade como os serviços de instalação de água e esgoto, instalações elétricas, coleta de lixo, entre outros serviços que utilizem áreas públicas para movimentar seus produtos e que, para tanto, necessitam de autorização do órgão competente.

Com o avanço tecnológico, equipamentos e *softwares* são utilizados para gerar facilidade no comércio, logística, gestão de pessoas e em muitas outras áreas. Uma das principais finalidades da Tecnologia da Informação (TI) é gerar subsídios para a comunicação, permitindo que usuários compartilhem e tratem informações numa velocidade muito maior que nos métodos manuais e tradicionais. Segundo Maia (2012), a internet revolucionou positivamente a sociedade, tornando-se um espaço de comunicação, política, economia, democracia, diversão, lazer e do exercício de liberdade de expressão.

Com esta revolução da internet um novo tipo de aplicação dominou o mercado de *softwares*, sendo estes conhecidos como *Web Applications*. Segundo Pressman (2002) pode ser considerada aplicação Web desde um site mais simples até um mais completo. Ferreira (2015) define aplicação Web como um *software* executado em uma *Web browser* e que são mantidas e atualizadas sem a necessidade de distribuí-lo e instalá-lo em computadores cliente.

Murarolli e Girotti (2015) asseguram que, em relação a recursos do dispositivo, as aplicações Web tem dificuldade de acessar funcionalidades como câmera e Sistema de Posicionamento Global (GPS) do smartphone, podendo fazer o acesso apenas por meio de Interface de Programação de Aplicativos (APIs) complexos, às vezes tão onerosa quanto o desenvolvimento nativo, com o intuito de sanar esta deficiência surgiu um novo modelo de aplicação Web conhecido como *Progressive Web Application* (PWA). Ater (2017) define Aplicação Web Progressiva como um novo ramo de aplicativos que combinam as

funcionalidades da aplicação nativa com a fluidez da Web, eles começam como simples sites e à medida que o usuário o utiliza e progressivamente coleta informações e se transforma para melhor atender o dispositivo utilizado para acessá-lo.

Segundo Murarolli e Girotti (2015), a principal vantagem de se optar pelo desenvolvimento de aplicações Web é a possibilidade de estas serem executadas em qualquer plataforma, simplesmente considerando a compatibilidade do navegador, de forma oposta ao desenvolvimento de aplicativos nativos, os quais exigem um projeto específico para cada sistema operacional.

Mediante a essas afirmativas, surge à necessidade da realização de um estudo, tendo como questão norteadora: qual a contribuição de uma PWA para a população que enfrenta os problemas infraestruturais urbanos?

Tendo em vista os pontos anteriores, este trabalho teve por objetivo geral desenvolver uma *Progressive Web Application* (PWA) para o compartilhamento de problemas infraestruturais de um determinado local de forma *crowdsourcing*.

Segundo Howe (2009), o *crowdsourcing* é um novo mecanismo na Web que modifica a forma como as informações são coletadas, ou seja, informações que antes proviam de um grupo seletivo de indivíduos passam a vir de diversos grupos de indivíduos, em suma da população.

Um Sistema de Informação *crowdsourcing* para problemas infraestruturais urbanos pode facilitar a comunicação entre os cidadãos de uma cidade que estiverem conectados a ele e ainda permitir que os órgãos competentes possam ter conhecimento do problema que precisa ser sanado, sem toda a complicação e ineficiência dos métodos arcaicos como telefonemas e gabinetes, que muitas vezes geram custos maiores e congestionamento de linhas.

A PWA desenvolvida funcionará da seguinte forma: os usuários têm as opções de compartilhar e visualizar um problema e também fazer alterações no status do mesmo. No processo de compartilhamento, o usuário deverá informar a localização e a categoria do problema e poderá adicionar fotos do mesmo. Seguindo aos parâmetros de uma PWA, esta aplicação poderá ser utilizada através de computadores e dispositivos móveis, com a opção complementar ao ser acessada de um dispositivo móvel a sincronização com o GPS e a câmera do aparelho, além de permitir que notificações sejam emitidas. Para que o projeto tenha êxito destacam-se os seguintes objetivos específicos: a) levantar os requisitos para o desenvolvimento da PWA; b) desenvolver uma PWA íntegra; c) desenvolver a PWA com uma interface intuitiva; e d) levar a população uma forma de compartilhar os problemas infraestruturais de sua cidade.

A PWA não foi a primeira opção, pois pensou-se em fazer uma Aplicação Web, porém, isto não possibilitaria aos usuários utilizar dos recursos do dispositivo de acesso, sendo que a aplicação nativa nunca foi uma opção pelo fato de se escolher apenas uma plataforma para atender. Então optou-se pela PWA visto que esta pode ser utilizada em qualquer dispositivo que possua um navegador. Este fato possibilita que esta possa ser utilizada por qualquer usuário, seja pelo *desktop*, smartphone *Android*, *iPhone*, *iPod*, dentre outros.

2. REFERENCIAL TEÓRICO

Esta seção tem por finalidade apresentar o posicionamento de outros autores sobre assuntos relevantes para o projeto em questão e está dividida em seis tópicos: Infraestrutura Urbana, Engenharia de Software, Aplicações Tradicionais, Aplicação Web Progressiva, Sistemas de Informação Geográfico e Trabalhos Correlatos.

2.1. INFRAESTRUTURA URBANA

Entende-se por infraestrutura urbana como o conjunto de sistemas técnicos de equipamentos e serviços necessários ao desenvolvimento das funções urbanas atendendo aos aspectos social, econômico e institucional (ZMITROWICZ; NETO, 1997).

Sob o aspecto social infraestrutura urbana visa promover adequadas condições de moradia, trabalho, saúde, educação, lazer e segurança. No que se refere ao aspecto econômico, infraestrutura urbana deve propiciar o desenvolvimento das atividades produtivas, isto é, a produção e comercialização de bens e serviços. E sob o aspecto institucional entende-se que a infraestrutura urbana deva propiciar os meios necessários ao desenvolvimento das atividades político-administrativas, entre os quais se inclui a gestão da própria cidade (ZMITROWICZ, 1997).

O sistema de infraestrutura urbana tendo como base os subsistemas técnicos setorial, a fim de facilitar sua compreensão, são classificados em: subsistema de Infraestrutura de Transporte, subsistema de Infraestrutura de Energia e Telecomunicação e subsistema de Saneamento. Subsistemas esses essenciais para que uma cidade possa fluir evitando o surgimento de alagamento de ruas, congestionamento de trânsito e falhas nos serviços de atendimento à população por parte do poder público, dentre outros (PUPPI, 1981).

O Subsistema de Infraestrutura de Transporte consiste nas vias responsáveis pelo fluxo de veículos. Para Puppi (1981), o subsistema de Infraestrutura de Transporte molda a configuração topográfica ao ponto de permitir o deslocamento fácil rápido e seguro entre pontos. Além disso, os espaços destinados ao subsistema de Infraestrutura de Transporte deverão conter redes e equipamentos de infraestrutura que compõem os demais subsistemas.

O Subsistema de Infraestrutura de Energia e Telecomunicação engloba sistema de transmissão de energia elétrica, postos de abastecimento de combustível, gasodutos e oleodutos, sistema de comunicação telefônica e redes de fibra ótica. De acordo com Zmitrowicz (1997), o subsistema de Infraestrutura de Energia e Telecomunicação é responsável por gerar e distribuir energia para a população, ou seja, produzir energia, realizar

a posteação, realizar as ligações prediais (conjunto de dispositivos que ligam a rede de distribuição ao medidor de consumo dos imóveis).

Já o Subsistema de Saneamento tem como objetivo assegurar o adequado escoamento das águas provenientes das chuvas que caem nas áreas urbanas. Também faz parte desse sistema o abastecimento de água, de esgoto, coleta e descarte do lixo. Segundo Mascaró (1994) e Puppi (1981), o subsistema de Saneamento garante o abastecimento de água potável suficiente para todos os usos, bem como a coleta desta quando não mais atender mais aos requisitos de usabilidade, mesmo para tarefas secundárias. Mascaró (1994) e Puppi (1981) também afirmam que este é responsável por drenar a água proveniente das chuvas e realizar a coleta de entulhos a fim de permitir uma melhor circulação de pessoas, proteção das edificações bem como prevenir inundações.

2.1.1. Infraestrutura Urbana no Brasil

Segundo Oliveira *et al.* (2011), em meados do século XX houve um grande êxodo rural para as cidades, causando um crescimento descontrolado das cidades tanto geograficamente como demograficamente, este crescimento não foi acompanhado pela infraestrutura urbana, devido a este fator foi improvisada uma infraestrutura mínima para parte da população.

Apesar de a situação mudar com o passar do tempo, ainda existem pessoas que vivem como uma infraestrutura mínima. Tendo estes pontos em vista, em 10 de julho de 2001 foram estabelecidas as diretrizes gerais da política urbana com a Lei nº 10.257/2001, sendo que no Artigo 2º, inciso I “[...] compreende o direito à terra urbana, à moradia, ao Saneamento ambiental, à infraestrutura urbana, ao transporte e aos serviços públicos, ao trabalho e ao lazer, para as presentes e futuras gerações” (BRASIL, 2001).

Porém, estas tentativas de melhorar a infraestrutura urbana não estão tendo êxito, e isto vem acompanhada da falta de manutenção da infraestrutura existente, ocasionando problemas como buracos em rodovias, descarte incorreto de lixo, esgotos a céu aberto entre outros. Telles (2013) em sua publicação revela que ainda no começo do Século XXI, levantamentos identificaram uma grande quantidade de pessoas que morrem por causas atribuídas à precariedade do atendimento das necessidades mínimas de água, esgoto e higiene.

2.2. ENGENHARIA DE SOFTWARE

Inúmeros programas são escritos por diversas pessoas, pessoas envolvidas com a ciência e a engenharia, desenvolvem programas para tratar seus dados, pessoas envolvidas com logística, muitas vezes escrevem códigos para suprir suas necessidades, alguns desenvolvem até mesmo por *hobby*. Embora programas possam ser escritos para diversas finalidades, a maior parte desses é feito como atividade profissional.

Softwares construídos de forma profissional geralmente são utilizados por outras pessoas além de seus desenvolvedores e, na maioria das vezes, são desenvolvidos por equipes e não apenas um indivíduo, nesse contexto surge a Engenharia de Software.

Pressman (2011, p. 39) fez uso da definição do Instituto de Eletricistas, Engenheiros e Eletrônicos (IEEE) para Engenharia de Software: “A aplicação de uma abordagem sistemática, disciplinada e quantificável no desenvolvimento, na operação e na manutenção de *software*, isto é a aplicação da Engenharia de Software”.

Já Ferreira (2013), descreve a Engenharia de Software como “uma disciplina de engenharia relacionada com todos os aspectos da produção de *software*, desde a especificação até a manutenção”.

A Engenharia de Software torna-se necessária no desenvolvimento profissional para subsidiar os cuidados para que a aplicação siga as especificações corretas aumentando as chances de sucesso do projeto. Ela tem por objetivo apoiar o desenvolvimento profissional de *software*, mais do que a programação individual. Ela inclui técnicas que apoiam especificação, projeto e evolução de programas, que normalmente não são relevantes para o desenvolvimento de *software* pessoal (SOMERVILLE, 2011).

2.2.1. Processo de Software

No desenvolvimento profissional, em que utiliza-se a Engenharia de Software é preciso seguir princípios e heurísticas de um processo de *software*.

Processo de *software* pode ser visto como o conjunto de atividades, métodos, práticas e transformações que guiam pessoas na produção de *software* (FALBO, 2005). O processo de *software* se divide em várias etapas, sendo estas análises, projeto, implementação e testes, apresentadas a seguir.

2.2.1.1. Análise e Elicitação de Requisitos

A primeira etapa do processo de *software* é a análise e elicitação de requisitos, em que consiste no entendimento de “o que” será desenvolvido.

Hull *et al.* (2005) definiu requisitos como uma declaração que identifica característica, capacidade ou fator de qualidade que ressalta a necessidade de um produto ou processo para o qual será adotada uma solução.

Outra definição dada para requisitos é: os requisitos são as características que definem os critérios de aceitação de um produto. A engenharia tem por objetivo colocar nos produtos as características que são requisitos (FILHO, 2000).

Nessa etapa de levantamento, surge a Engenharia de Requisitos (ER) para auxiliar com técnicas a definição correta dos requisitos de um *software*.

A ER foi definida por Lamsweerde (2000, p. 5) como:

[...] a identificação dos objetivos a serem atingidos pelo futuro sistema, a operacionalização de tais objetivos em serviços e restrições, e a atribuição de responsabilidades pelos requisitos resultantes a agentes humanos, dispositivos e *software* (LAMSWEERDE, 2000, p. 5).

Já Zave (1997, p. 315) definiu ER como:

[...] é o ramo da Engenharia de Software que está preocupado com os objetivos do mundo real para as funções e restrições aplicáveis a sistemas de *software*. Está também preocupado com o relacionamento destes fatores para especificações precisas do comportamento do *software* e com sua evolução no tempo e através de famílias de produtos (ZAVE, 1997, p. 315).

A ER tem como objetivo criar e manter um documento de requisitos de sistema. Esses documentos estão relacionados à avaliação de se o sistema é útil para a empresa, obtenção de requisitos, conversão dos requisitos em alguma forma padrão e a verificação de se os requisitos realmente definem o sistema que o cliente deseja (SOMMERVILLE, 2011).

2.2.1.2. Projeto e Implementação

Após a análise e elicitação de requisitos a etapa seguinte é projeto, esta fase consiste em definir “como” o *software* será desenvolvido. Um projeto de *software* é uma descrição da estrutura do *software* a ser implementado, dos modelos e estruturas de dados usados pelo sistema, das interfaces entre os componentes do sistema e, às vezes, dos algoritmos usados (SOMMERVILLE, 2011, p. 25).

Existem vários métodos para se criar um projeto de *software*, cada qual com um conjunto de princípios e notações. Dentre os diversos podemos citar: Object-Oriented Design (OOD), que significa design orientado a objetos e projeto estruturado e desenvolvimento estruturado de Jackson (PRESSMAN, 1995). Existem diversas metodologias para modelagem e OOD criadas ao longo dos últimos anos. Embora existam interpretações particulares em cada metodologia, todas elas apresentam a programação orientada a objetos como um novo paradigma, ou seja, uma nova maneira de pensar, e não simplesmente uma técnica de programação (FURLLAN, 1998).

Após o encerramento do projeto de *software*, este deve seguir para a próxima etapa, a implementação. O estágio de implementação do desenvolvimento de *software* é o processo de conversão de uma especificação do sistema em um sistema executável (SOMMERVILLE, 2011).

2.2.1.3. Validação de Software

Sommerville (2011) define validação e verificação como uma etapa que tem a intenção de mostrar que um *software* se adequa às suas especificações para o qual foi construído ao mesmo tempo em que satisfaz as especificações do cliente. Teste de programa em que o que o sistema é executado com dados de testes simulados é a principal técnica de validação.

Segundo Delamaro *et al.* (2007) e Sommerville (2011), no teste de *software* há três grandes fases:

- a) **Teste de desenvolvimento:** Consiste em testar de forma independente cada componente, por quem o desenvolveu. Os componentes podem ser entidades simples, como funções ou classes de objetos, ou podem ser agrupamentos coerentes dessas entidades (MASSOL; HUSTED, 2003).
- b) **Teste de sistema:** Consiste em integrar os diversos componentes que compõe o *software* a fim de encontrar problemas gerados por esta integração. Também visa mostrar que o sistema satisfaz seus requisitos funcionais e não funcionais, bem como testar as propriedades emergentes do sistema.
- c) **Teste de aceitação:** É o último estágio do processo de teste, o *software* é testado com dados fornecidos pelo sistema e não por dados de um simulador. Os testes de aceitação

também podem revelar problemas de requisitos em que os recursos do sistema não atendam às necessidades do usuário ou o desempenho do sistema seja inaceitável.

2.2.2. Diagramas UML

Um tópico importante no desenvolvimento de *softwares* profissionais é a modelagem de dados. Segundo Araújo (2008) “modelagem de sistemas, tanto a nível funcional quanto de dados, é um requisito fundamental para a obtenção de produtos de *software* de maior qualidade e confiabilidade”.

A modelagem é feita através de diagramas e documentações, e uma das formas de realizar esse tipo de modelagem é através da utilização da Linguagem de Modelagem Unificada (UML).

Guedes (2014) afirma que “a UML é uma linguagem visual utilizada para modelar sistemas computacionais por meio do paradigma de Orientação a Objetos”. Por meio dos diagramas UML é possível representar *softwares* sob diversas perspectivas, além de que facilita a comunicação entre todas as pessoas envolvidas no projeto do desenvolvimento do sistema, portanto, é considerada uma das linguagens mais expressivas para modelagem sistemas orientados a objetos, pois resume os principais métodos disponíveis (OMG, 2005a) (OMG, 2005b) (OMG, 2005c) (OMG, 2006).

A utilização dessa linguagem pode servir para modelar todas as etapas do processo do desenvolvimento de *software*, produzindo todas as documentações e artefatos necessários em cada etapa (GUDWIN, 2010).

2.3. APLICAÇÕES TRADICIONAIS

Até o final de 2014, quando se pensava em criar uma aplicação tinha-se como alternativas as aplicações tradicionais, são estas aplicações *Web*, *Nativa*, *Híbrida* e *Desktop*.

Aplicação *desktop* é um *software* desenvolvido para ser executado somente em computadores padrão, *desktop*, *notebook* ou computador de uso geral (MICROSOFT, 2010). Devido a fator da execução desta estar condicionada a um *personal computer* (computador

peçoal), em que esta não atende aos critérios de praticidade impostos a este trabalho, e por isso, este modelo de aplicação não será abordado no presente trabalho.

Os tópicos Aplicação Web, Aplicação Nativa e Aplicação Híbrida apresentam uma breve descrição dos tipos de aplicações abordadas nesse estudo, seguidos de suas vantagens e desvantagens.

2.3.1. Aplicação Web

O termo Aplicação Web é definido por Kappel *et al.* (2006) como um sistema de *software* baseado em tecnologias e padrões do *World Wide Web Consortium* (W3C) que oferece recursos e/ou serviços através de um *Web Browser*.

Segundo Kappel *et al.* (2006), a facilidade de acesso a informações e serviços proporcionada pela web tem possibilitado o uso em larga escala dessas aplicações na sociedade, como suporte à realização de atividades cotidianas. Mais do que apoiar atividades, as aplicações web estão contribuindo para influenciar a forma como as pessoas efetuam compras, procuram informações, comunicam-se, estudam (educação à distância) e realizam atividades de entretenimento (PRESSMAN; LOWE, 2009).

Segundo Raj e Tolety (2012), as Aplicações Web apresentam as seguintes vantagens e desvantagens:

- a) **Vantagens:** Uma Aplicação Web não requer nenhum tipo de instalação, podendo ser acessada pelo *Web Browser* do dispositivo, portanto, os dados da aplicação são armazenados em um servidor externo, permitindo que atualizações sejam feitas sem que haja manutenção no dispositivo onde é feita a utilização da mesma.
- b) **Desvantagens:** Visto que as Aplicações Web utilizam a internet para trocar informações com o servidor, estas podem perder desempenho devido à instabilidade de conexão e rede. Além disso, este tipo de aplicação não pode acessar *hardware* e *software* do dispositivo tais como: câmera, GPS, entre outros. Por ser executada em um *Web Browser* o desenvolvedor tem menos controle quanto à forma como o conteúdo desta será processado pelo diferentes *Web Browser*.

2.3.2. Aplicação Nativa

Uma aplicação nativa/embarcada é um *software* desenvolvido para executar em uma plataforma específica. Segundo Charmmas *et al.* (2013), as Aplicações Nativas precisam seguir um pacote de desenvolvimento de *software* específico para o qual estiver sendo desenvolvido, chamado Kit de desenvolvimento de *software* (SDK).

De acordo com Redda (2012), Toledo e Deus (2012), as Aplicações Nativas apresentam as seguintes vantagens e desvantagens:

- a) **Vantagens:** Devido ao fato de estar instalada no próprio dispositivo e possível à manipulação de dados *offline*, permitindo também o uso sem perda desempenho. Este também pode utilizar dos recursos de *hardware* e *software* do dispositivo.
- b) **Desvantagens:** Requer uma quantidade de armazenamento do dispositivo destinada a ela. Além disso, as plataformas são uma grande barreira, pois diferente da Aplicação Web que são executadas pelos *Web Browser*, neste tipo de aplicação necessita ser desenvolvida uma aplicação para cada plataforma.

2.3.3. Aplicação Híbrida

Uma Aplicação Híbrida é modelo de aplicação que é parcialmente web e parcialmente nativa. Segundo Charkaoui *et al.* (2014), Aplicação Híbrida é desenvolvida utilizando tecnologias web e é executada dentro do container nativo do dispositivo móvel.

Os aplicativos híbridos são populares porque permitem desenvolvimento multiplataforma, utilizando o mesmo HTML para diferentes sistemas operacionais – como através de ferramentas como o Cordova, PhoneGap e Sencha Touch permitem, inclusive compilando para o formato nativo, reduzindo custos de produção. Para Redda (2012), Toledo e Deus (2012), as Aplicações Híbridas apresentam as seguintes vantagens e desvantagens:

- a) **Vantagens:** A interface de usuário pode ser reutilizada em diferentes plataformas utilizando recursos da plataforma nativa. Na Aplicação Híbrida, diferente da web a possível acessar o *hardware* e o *software* do dispositivo.
- b) **Desvantagens:** A Aplicação Híbrida tem menor desempenho em comparação a Nativa devido ao fato de esta executar na *Web Browser*. Mesmo com a reutilização da interface em diferentes plataformas, a aplicação não terá a aparência de uma Aplicação Nativa.

2.4. APLICAÇÃO WEB PROGRESSIVA

Um *Progressive Web Application* (PWA) é uma aplicação web que faz uso do melhor das tecnologias que a web pode oferecer para obter comportamentos de aplicações nativas. A ideia de uma PWA é ir adicionando progressivamente, funcionalidades que antes só eram possíveis em aplicativos nativos (FRANSSON; DRIAGUINE, 2017).

Para Malavolta (2016), as PWAs podem ser descritas como aplicativos especiais da Web que utilizam progressivamente a tecnologia da Web moderna para, por exemplo, ativar: uso off-line, processamento em segundo plano e enviar notificações.

Segundo o *Google Developers*, os PWAs possuem as seguintes características:

- a) **Responsividade:** Adapta-se a qualquer fator de forma: *desktop*, celular, *tablet* ou similares.
- b) **Trabalhar offline:** Aprimorado com *Service Workers* para trabalhar *offline* ou com baixa qualidade redes.
- c) **Aparência Nativa:** Parece um aplicativo, porque o modelo de *shell* da aplicação separa as funcionalidades para um dispositivo móvel das funcionalidades para web.
- d) **Atualizado:** Sempre atualizado graças ao processo de atualização do prestador do serviço, ou seja, quando uma atualização é feita no servidor, automaticamente ela será enviada a todos os dispositivos que acessarem a aplicação através daquele servidor.
- e) **Segura:** Com auxílio do processo de atualização de cache do *service worker* e da própria web, a PWA deverá estar sempre na versão mais atualizada, visto que a mesma não é armazenada ou instalada no dispositivo do qual é utilizada e sim acessada através de um servidor, onde estará toda a informação e dados da mesma.
- f) **Detectável:** É identificável como um “aplicativo” graças ao manifesto da web e ao serviço do Consórcio World Wide Web (W3C), permitindo que os mecanismos de pesquisa o encontrem, logo, o ícone da aplicação poderá ser encontrado em meio aos aplicativos do dispositivo que a utiliza ou mesmo a aplicação poderá ser encontrada através da busca de aplicativos em um aparelho móvel ou através de um domínio na internet.
- g) **Instalável:** Permite que os usuários adicionem aplicativos que considerem mais úteis em sua tela inicial sem o incômodo de uma loja de aplicativos.
- h) **Linkável:** Compartilhe facilmente o aplicativo via Localizador Uniforme de Recurso (URL), não requer instalação complexa.

De acordo com o site *Progressive Web Apps*, qualquer tipo de aplicação deve ser confiável, rápida e atraente. Na PWA isto é possível graças a novas APIs (*Application Programming Interface*) e padrões, principalmente *Web App Manifest* e *Service Workers* em conjunto com HTML5, *Javascript* e CSS3.

2.4.1. Web App Manifest

O *Web App Manifest* é um arquivo que obrigatoriamente deve estar presente em todas as *Progressive Web Application*. Segundo *Mozilla Developer Web App Manifest* é um arquivo JASON (*Javascript Object Notation*) que fornece informações importantes sobre a aplicação: versão, nome, descrição e domínios dos quais o aplicativo pode ser instalado.

2.4.2. Service Workers

Segundo Bondança *et al.* (2017), *Service Worker* é uma tecnologia que permite que aplicações web usufruam do processamento em segundo plano do *web browser*. De acordo com a *Google Service Worker* é um *script* que seu navegador executa em segundo plano, separado da página da *Web*, possibilitando recursos que não precisam de uma página da web ou de interação do usuário.

Para Gaunt (2017), funcionalidades que normalmente são associadas aos aplicativos móveis nativos como: sincronizações periódicas em segundo plano, notificações *push* e capacidade de utilização do sistema quando *offline*. Atualmente são disponibilizadas na web por meio de *Service Workers*.

2.5. SISTEMA DE INFORMAÇÃO GEOGRÁFICO

Vive-se em uma era em que a Informática está presente em tudo, isto possibilitou melhoria nos meios de análise de dados, para Silva (1992) há alguns anos atrás quando se falava em geoprocessamento estava referindo-se a cálculos matemáticos para tratamento de informações geográficas, porém hoje em dia este termo também é utilizado para se referir aos Sistemas de Informação Geográfica (SIG). Os Sistemas de Informação Geográfica podem ser

definidos como sistemas computacionais que permitem a captura, manipulação, armazenamento, modelagem, análise e apresentação de dados referenciados geograficamente, ou seja, ele é capaz de facilitar o armazenamento de informações e a captura de dados e de representá-los geograficamente (SILVA, 1992).

Os SIG podem ser utilizados em uma infinidade de áreas, Oliveira (1997) e Silva (1992) apresentam uma relação das diversas áreas de aplicação deste, dividindo-as em cinco grupos:

- a) **Ocupação Humana:** Neste grupo encontram-se áreas como planejamento e supervisão de limpeza, controle epidemiológico, planejamento urbano e redes de infraestrutura.
- b) **Uso da Terra:** Neste grupo encontram-se áreas como classificação de solos, levantamento topográfico, cadastro de propriedades rurais e planejamento de barragens.
- c) **Uso de Recursos Naturais:** Neste grupo encontram-se áreas como classificação de poços petrolíferos, distribuição de energia elétrica e planejamento de gasodutos e oleodutos.
- d) **Meio Ambiente:** Neste grupo encontram-se áreas como controle de queimadas, estudo de modificações climáticas e gerenciamento florestal de desmatamento e reflorestamento.
- e) **Atividades Econômicas:** Neste grupo encontram-se áreas como planejamento de marketing, pesquisas socioeconômicas e distribuição de produtos e serviços.

Apesar de existirem diversos Sistemas de Informação Gerenciais, sendo que o Sistema de Informação Geográfica é a melhor opção quando se fala em analisar áreas. Os SIG diferenciam-se dos demais sistemas de informação pelas suas capacidades de análise espacial, que permite acrescentar valor à informação geográfica, identificar padrões e suportar decisões, transformando dados brutos em conhecimento útil, que pode auxiliar na tomada de resoluções (GOODCHILD *et al.*, 2005).

2.6. DESIGN DE INTERAÇÃO

Para Preece, Rogers e Sharp (2013), muitos *softwares* que requerem a interação dos usuários ao realizar suas tarefas não foram inicialmente projetados visando os usuários e sim como sistemas para executar determinadas funções. Esses sistemas, muitas vezes possuem um funcionamento eficaz, se olhados na perspectiva da engenharia, porém, o usuário do mundo real encontra dificuldades ao utilizá-lo. Para solucionar esse problema, surge o *Design* de

Interação (DI) com objetivo de redirecionar essa preocupação, trazendo usabilidade para o usuário final.

O DI trata-se de um plano ou esquema preconcebido, com o intuito de ser executado, isto é, significa projetar *designs* que se voltem para o desenvolvimento desse esquema mental (ROGERS; SHARP; PREECE, 2013).

2.7. FERRAMENTAS

Para desenvolvimento da aplicação foram utilizadas ferramentas livres (*open-source*), ou seja, ferramentas sem nenhum custo ou versões gratuitas de alguns *softwares*. A escolha das ferramentas foi feita através de estudos, buscando as ferramentas que melhor atendiam as necessidades do projeto. A aplicação foi desenvolvida através de uma combinação de linguagens para criação de interface e programação de suas funcionalidades. Para o desenvolvimento da interface gráfica, foram utilizadas a Linguagem de Marcação de Hipertexto (HTML) e de Folha de Estilo em Cascata (CSS) para criação da interface gráfica, JavaScript e *Hypertext Processor* (PHP) para programação das funcionalidades do sistema e Linguagem Unificada de Modelagem (UML) para modelagem do sistema.

2.7.1 Astah

Para modelagem do sistema, foi utilizada a UML, a linguagem mais comum para a modelagem de um sistema. Nesse processo, foi utilizado o *software Astah Professional 7.2*, que permite criar diagramas em UML. Embora este seja um *software* pago, a empresa libera licenças gratuitas para estudantes, que foi utilizada nesse trabalho. O Astah foi aplicado para o desenvolvimento do Diagrama de caso de uso (APÊNDICE A) e Diagrama de Classe (APÊNDICE B).

2.7.2 CodeIgniter

A programação do sistema, em sua maior parte, foi feita através da linguagem PHP, por se tratar de uma aplicação *web*, para tal, foi utilizado o *framework* CodeIgniter.

O CodeIgniter é um *framework* de código aberto, ou seja, pode ser utilizado por qualquer pessoa, que tem como finalidade acelerar e facilitar o desenvolvimento em PHP. Uma importante característica dessa ferramenta, é que a mesma utiliza o padrão *Model-View-Controller* (MVC). Um dos principais objetivos do padrão MVC é a organização do código em camadas, assim, realizando a separação lógica dos componentes do sistema. Nesse modelo, a organização em camadas é a chave para a independência dos componentes, agrupando-os por responsabilidades em comum (DEVEMEDIA, 2018).

2.7.3 Bootstrap

Dado o objetivo da aplicação, torna-se necessário o uso de um *design* responsivo, visto que a mesma usa-se em diversos dispositivos diferentes. *Web design* responsivo envolve um conjunto de tecnologias e técnicas para que uma aplicação possa funcionar em variados dispositivos do modo mais prático possível. Essa forma de aplicação não é uma preocupação somente dos profissionais da *web*, mas também empresas procuram formas de que seus sites possam ser utilizados independentemente do dispositivo (FISHER; SHARKIE, 2013).

Para acelerar a criação do *design* responsivo na interface da aplicação, foi utilizado o *framework* Bootstrap 4.1.3, que se trata de uma biblioteca com um conjunto de códigos abertos, ou seja, qualquer pessoa pode baixar e utilizá-lo.

2.7.4 API do Google Maps

Para seu bom funcionamento da aplicação desenvolvida, são exigidos serviços de localização. Para adicionar essa funcionalidade, foi utilizado o API do Google Maps. Essa aplicação é um serviço público e gratuito para qualquer desenvolvedor, desde que o usuário final não seja cobrado para utilizar a aplicação desenvolvida. Essa API possibilita diversos

serviços, como marcar um ponto no mapa, identificar a atual localização do dispositivo, ativar serviço de GPS, dentre outros.

2.7.5 Servidor Web

A aplicação foi desenvolvida para uso através da *internet*, e, portanto, torna-se necessário o uso de um servidor *web*.

Atualmente, além de documentos textuais, circulam na *internet* documentos de multimídia (áudio e vídeo). Como consequência, aumenta o tráfego de dados na rede mundial de computadores, tornando-se necessário uma combinação de *hardware* e *software* para armazenar esses dados e distribuí-los na *internet*, dando origem assim, aos servidores *web*. Estes servidores *webs* são responsáveis por armazenar e trocar informações com outras máquinas e também executar scripts e programas (ESCOLA LINUX, 2016).

2.7.6. MYSQL

A aplicação foi desenvolvida para os usuários compartilharem os problemas de infraestrutura das cidades, em que estes problemas devem ser armazenados para posteriormente serem acessados por outros usuários, e para isso utilizou-se o sistema de dados.

Um sistema de banco de dados tem como vantagens minimizar as incoerências, garantir a integridade dos dados, facilitar a realização de qualquer modificação necessária e possibilitar a criação de restrições de acesso, de maneira organizada (THOMSON; WELLING, 2005).

Para o gerenciamento do sistema de banco dados utilizou-se o MYSQL, que é um sistema gerenciador de banco de dados relacional que permite a criação e manipulação de tabelas através de comandos SQL, sendo de código aberto, o que o faz ser usado em grande parte de aplicações gratuitas que gerem suas bases de dados, estando presente na maioria dos servidores *web* (MILANI, 2007).

2.8. TRABALHOS CORRELATOS

Os avanços tecnológicos, em especial na área da TI, vieram com um propósito, facilitar a vida do homem com a automatização de atividades, sejam elas as mais complexas ou simples atividades do dia a dia. Atualmente, existem diversos *softwares* circulando no mercado. A TI é utilizada em muitos setores como, por exemplo: saúde, educação, transporte, financeiro, dentre outros, apresentando um quadro bastante otimista e propiciando avanço científico em várias áreas do conhecimento (BACCILI; DELFINO; VIEIRA, 2011).

Para comprovar a viabilidade do uso de sistemas computacionais na resolução de problemas do dia a dia, foram feitas investigações objetivando analisar trabalhos correlatos ao tema em questão, é valido salientar que os autores não seguiram a mesma linha de pesquisa.

Sun e Mobasheri (2017), realizaram um estudo utilizando dados *crowdsourced* do Strava, um aplicativo web progressivo para corredores e ciclistas, para realizar um estudo de caso sobre a poluição do ar e o ciclismo. Nessa aplicação obtiveram dados de GPS e monitoramento da saúde dos ciclistas.

Santos, *et al.* (2004), utilizou um Sistema de Informação Geográfica para descrição dos casos de tuberculose no município de Ribeirão Preto. O sistema utilizado para a coleta dos dados foi o Epi-Tb e após, para compilação e geocodificação das informações, utilizou-se, como auxílio, o *software* MapInfo Professional.

Nero, *et al.* (2010), desenvolveram um SIG 3D para a área de distribuição de energia elétrica. Esse sistema, permite através de seu Algoritmo Evolutivo (AE) a otimização do processo de reconfiguração e restauração das redes de energia e também um ambiente tridimensional que permite a visualização de elementos da cartografia básica dos principais edifícios da cidade de São Carlos e os ativos da rede de distribuição configurados com o algoritmo.

A pesquisa dos autores supracitados visou o desenvolvimento de uma aplicação para auxílio na resolução de problemas infraestruturais, com funcionalidades diversas, distinguido se assim de todos os trabalhos encontrados. Embora os trabalhos listados possuam linhas de pesquisas diferentes, todos utilizaram recursos e tecnologias semelhantes ao do presente trabalho.

3. METODOLOGIA

Nesta seção são apresentados os meios que levaram aos resultados esperados para este projeto, tais como a natureza da pesquisa, materiais, instrumentos e procedimentos utilizados, população amostra, identificação do caráter da pesquisa e tratamento de dados. Segundo Lakatos e Marconi (2010), a metodologia científica abrange o maior número de itens de um trabalho, pois responde, a um só tempo, as questões como?., com que?, onde? e quando?

3.1. NATUREZA DA PESQUISA

A metodologia aplicada no desenvolvimento desse estudo possui caráter descritivo. Segundo Gil (2006), as pesquisas de caráter descritivo têm como principal objetivo a descrição das características de certa população, fenômeno ou então o estabelecimento de relações entre variáveis. Nesse tipo de pesquisa, realiza-se todo o estudo, análise e interpretação dos fatos do meio físico sem que haja interferência do pesquisador. As pesquisas descritivas se assemelham aos estudos de casos, nos quais são coletados os dados e realizadas as análises das relações entre as variáveis que servirão para definir os efeitos resultantes (PEROVANO, 2014).

Quanto ao caráter da pesquisa, nesse estudo foi aplicado o caráter qualitativo, visto que usou-se um nível de realidade que não foi quantificado, mostrando significados, motivos, valores e atitudes, sem fazer uso de meios estatísticos (MINAYO, 2010).

3.2. POPULAÇÃO E AMOSTRA

A população para realização deste trabalho foi composta por cidadãos de São João Evangelista e região, alunos e professores do Instituto Federal de Minas Gerais (IFMG-SJE). Como amostra, foi selecionado um grupo de vinte pessoas dentre a população, a fim de avaliar a viabilidade e o impacto que esse sistema pode gerar de forma positiva na construção de meios e recursos para que sejam solucionados os problemas correspondentes à infraestrutura do município.

3.3. MÉTODOS E PROCEDIMENTOS

A aplicação foi desenvolvida utilizando a linguagem PHP, para agilizar o processo de desenvolvimento utilizou-se o *framework CodeIgniter*, já a interface da aplicação foi construída utilizando HTML, CSS, *JavaScript* e o *framework Bootstrap*.

A primeira etapa deste projeto foi a elicitação dos requisitos do *software*, foi feita uma reunião com nosso orientador, nesta elegemos diversas funcionalidades para o este, ao final escolhemos dentre estas as que tinham mais relevância e assim definimos os requisitos da aplicação (TABELA 1 e TABELA 2).

Tabela 1 - Requisitos Funcionais

CÓDIGO	REQUISITO
RF_01	Cadastro de usuário.
RF_02	Autenticação de usuário.
RF_03	Gerenciar notícia.
RF_03.01	Cadastrar notícia.
RF_03.02	Editar notícia.
RF_03.03	Excluir notícia.
RF_03.04	Alterar status da notícia.
RF_04	Listar notícias
RF_05	Marcar Notícias no mapa
RF-06	Acesso com nível de usuário

Fonte: Elaborada pelos autores.

Tabela 2 - Requisitos não funcionais

CÓDIGO	REQUISITO
RNF_01	A aplicação deve ser Web progressiva.
RNF_02	A aplicação deve ser íntegra
RNF_03	A aplicação deve ser de fácil utilização.

Fonte: Elaborado pelos autores.

Após os requisitos definidos passou-se para a etapa de modelagem da aplicação, nesta foi feito o diagrama de caso de uso (APÊNDICE A) e o diagrama de classes (APÊNDICE B). Nesta também foi feito um esboço da parte visual da aplicação.

A etapa seguinte foi o desenvolvimento, nesta utilizou-se o servidor LocaWeb, pelo fato da PWA só funcionar em servidores HTTPS, nesse etapa fizemos o desenvolvimento de todas as telas da aplicação bem como a configuração dos arquivos *Web APP Manifest* e *Service Worker*.

Na tela principal da aplicação possui diversas notícias de problemas infraestruturais da cidade. Ao abrir uma notícia há uma descrição e um mapa marcando a localização do problema. Para isso, utilizou-se a API do *Google Maps* que permite inserir mapas do *Google* em aplicações de terceiros.

Por ser uma PWA, apresentam-se algumas mudanças quando acessada por um dispositivo móvel, utilizamos o *Web APP Manifest* para definir o ícone da aplicação, como esta será iniciada entre outras mudanças que foram feitas para este ficar semelhante a uma aplicação nativa.

3.4. TRATAMENTO DE DADOS

Após o teste da aplicação, utilizou-se o teste de componentes, em que usuários externos fizeram uso da aplicação e responderam a perguntas de um roteiro de entrevista pré-elaborado, com o objetivo de medir o desempenho e a usabilidade da aplicação.

A Tabela 3 apresenta o roteiro de entrevista aplicado a vinte pessoas selecionadas que participaram do teste do *software*, sendo todos moradores do município de São João Evangelista e região, alunos e professores do Instituto Federal de Minas Gerais (IFMG-SJE). O questionário aplicado é composto por onze questões, sendo as de 1 a 7 visando analisar designer e usabilidade, e as questões de 8 a 10 a viabilidade do *software*.

Tabela 3 - Roteiro de entrevista para Análise do Software

Questão 1	A interface da aplicação é amigável?
Questão 2	A aplicação se adaptou perfeitamente à tela de seu dispositivo?
Questão 3	Os campos da aplicação são intuitivos?
Questão 4	O tempo de resposta da aplicação é satisfatório?
Questão 5	Durante o uso da aplicação surgiu alguma dúvida?
Questão 6	Qual o grau de dificuldade de uso da aplicação?
Questão 7	Como foi sua experiência de uso da aplicação?

Questão 8	A aplicação atende a todas as suas especificações?
Questão 9	A aplicação atendeu a suas expectativas?
Questão 10	A aplicação é adequada ao problema?
Questão 11	Considerações (problemas, sugestões e etc.).

Fonte: Elaborado pelos autores.

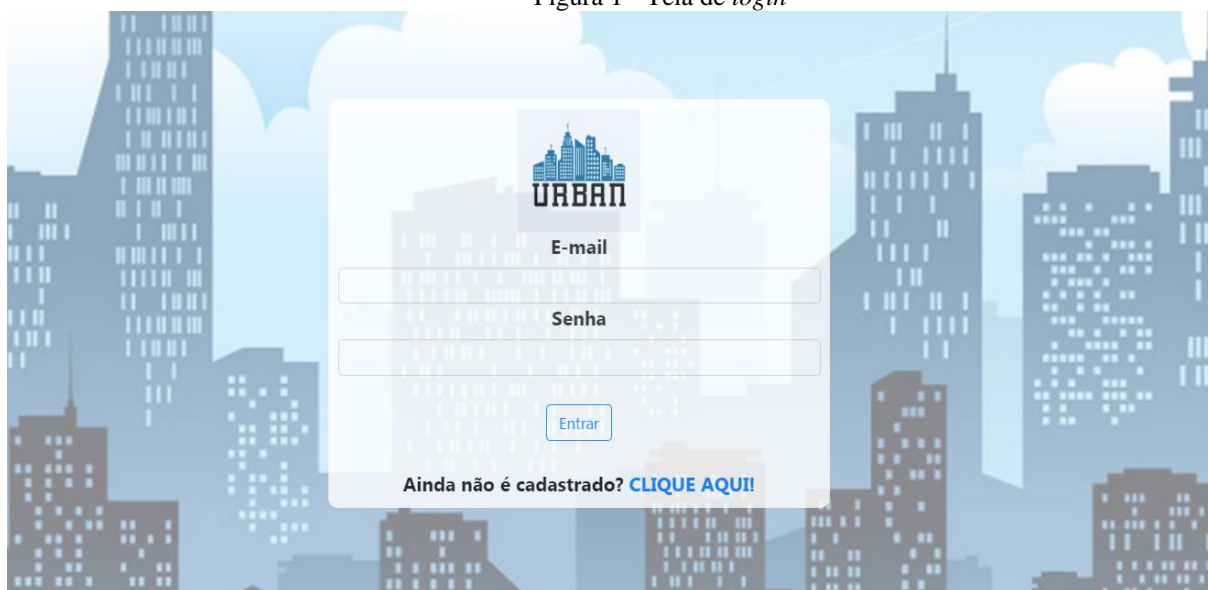
As questões de 1 a 7 foram construídas baseadas parcialmente nas heurísticas de Nielsen (1994), sendo que estas são compostas por dez melhores práticas de desenvolvimento, estas são: a) visibilidade do estado do sistema, correspondência entre o sistema e o mundo real; b) controle e liberdade do usuário; c) consistência e padrões; d) prevenção de erros; e) ajuda aos usuários para reconhecerem, diagnosticarem e se recuperarem de erros; f) reconhecimento em vez de memorização; g) flexibilidade e eficiência de uso; h) *design* estético e minimalista; i) ajuda e documentação. Estas questões buscam entender critérios de usabilidade da aplicação como: *design* amigável, adaptação da aplicação ao dispositivo, tempo de resposta da aplicação e complexidade do uso desta.

A questão 7 foi de cunho discursivo para os respondentes apresentarem sobre a experiência de uso da aplicação levando em consideração o *design* e a usabilidade.

As questões de 8 a 10 tinham por objetivo verificar a viabilidade do *software*, na qual foi verificado se a aplicação atendia a todas as suas especificações e se esta seria adequada para solucionar o problema em questão.

3.5. PROTOTIPAÇÃO

A Figura 1 apresenta a tela de *login* da aplicação, contendo a logo da aplicação com seu nome URBAN e os campos de e-mail e senha para realização do login e os botões “entrar” para usuários cadastrados acessarem sua conta, e “clique aqui” para novos usuários se cadastrarem.

Figura 1 - Tela de *login*

A tela de login apresenta o logotipo 'URBAN' no topo, seguido por campos de entrada para 'E-mail' e 'Senha'. Abaixo dos campos, há um botão 'Entrar' e um link 'Ainda não é cadastrado? CLIQUE AQUI!' em azul. O fundo da tela mostra uma ilustração de um skyline urbano com edifícios e nuvens.

Fonte: Elaborada pelos autores.

Ao ser clicado, o botão “clique aqui” apresenta uma tela solicitando os dados necessários para que seja efetuado o cadastro de usuário (Figura 2).

Figura 2 - Cadastrar



A tela de cadastro, intitulada 'CADASTRAR', contém campos de entrada para 'Nome', 'E-mail', 'Senha' e 'Confirmar Senha'. Abaixo dos campos, há um botão 'Efetuar Cadastro' e um link 'JÁ SOU CADASTRADO!' em azul. O fundo da tela mostra a mesma ilustração de skyline urbano vista na tela de login.

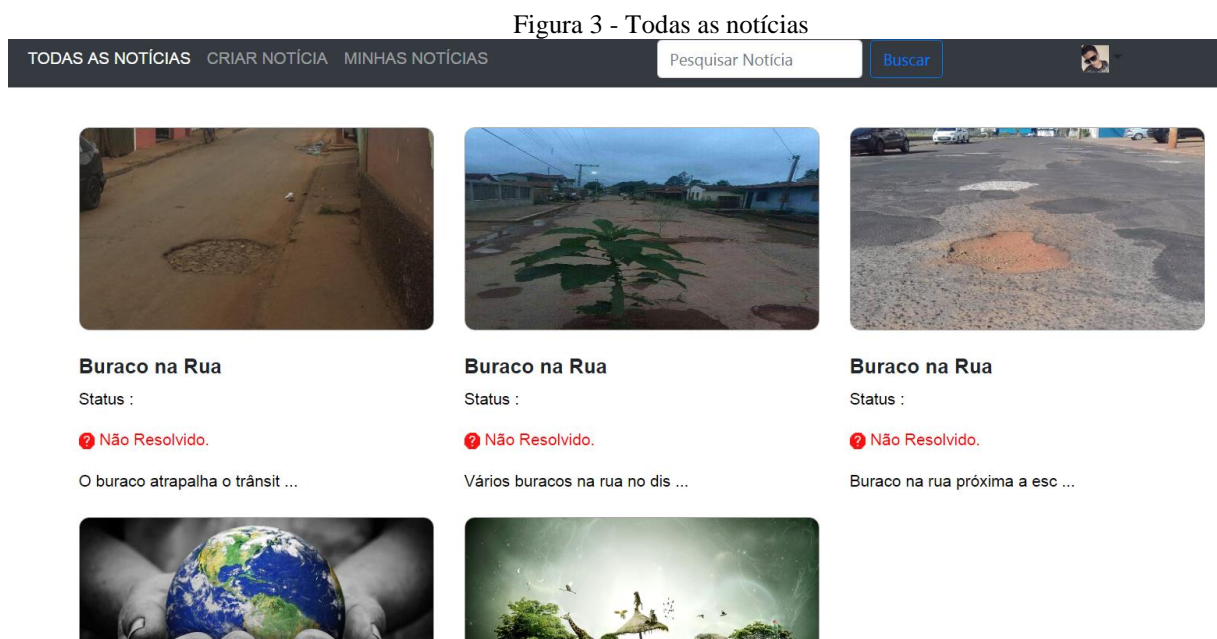
Fonte: Elaborada pelos autores.

A tela “cadastrar” contém os campos “nome”, “e-mail”, “senha” e “confirmar senha”, além dos botões “efetuar cadastro” e “já sou cadastrado”.

Após preencher todos os dados corretamente, ao pressionar o botão “efetuar cadastro” o usuário é redirecionado para a tela inicial, onde poderá fazer a realização do *login*. Ao clicar o botão “já sou cadastrado” o usuário também é redirecionado para a tela de *login*.

Ao entrar no sistema, o usuário é direcionado para a tela contendo todas as notícias cadastradas. O sistema possui um menu fixo com os botões “todas as notícias”, “criar notícia”, “minhas notícias” e buscar. Além disso, esse menu possui um campo para o usuário digitar a busca e uma seta, que dá acesso ao usuário a outros botões. Possui também um campo que apresenta a foto do perfil.

Quando clicado o botão “todas as notícias” o usuário é direcionado para uma página com as notícias cadastradas no sistema, agrupadas em linhas de três a três, contendo o título, *status* e uma breve descrição, conforme a Figura 3.



Fonte: Elaborada pelos autores.

Quando pressionado, o botão “criar notícia” ativa uma página com os campos para que o usuário insira informações sobre a notícia, sendo eles: “tipo de problema”, “descrição” e “localização do problema”. Essa tela possui também os botões “escolher arquivo” e “salvar”.

O campo “tipo de problema” possui algumas predefinições para ajudar o usuário nesse processo de selecionar o tipo. O campo descrição é uma caixa de texto para que o usuário descreva o problema de forma mais detalhada. Já o campo “localização do problema” possui um mapa do Google, para que o usuário escolha o ponto da cidade onde está localizado o problema inserido.

Abaixo dos campos, encontra-se o botão “selecionar arquivo” para que o usuário possa realizar o *upload* de fotos sobre o problema descrito e o botão “salvar” que grava a notícia no banco de dados, de acordo com a Figura 4.

Figura 4 - Criar notícia

TODAS AS NOTÍCIAS CRIAR NOTÍCIA MINHAS NOTÍCIAS Pesquisar Notícia Buscar

CRIAR NOTÍCIA

Tipo de Problema:
Poste Caído

Descrição:

Localização do Problema:

Mapa Satélite

Sorveteria Expressminas Futebol10™

R. Mal. Floriano R. Mal. Floriano

Google Dados cartográficos ©2018 Google Termos de Uso Informar erro no mapa

Adicionar Imagem:
Escolher arquivo Nenhum arquivo selecionado

Salvar

Fonte: Elaborada pelos autores.

Ao clicar no botão “minhas notícias” o usuário é direcionado para uma página contendo as notícias que ele cadastrou, composta por uma imagem, o *status* e uma breve descrição de cada notícia. Abaixo de cada notícia têm três botões: “excluir”, “editar” e “alterar *status*”.

O botão “excluir” permite que o usuário delete a notícia e ao ser clicado, ativa uma caixa de confirmação onde o usuário pode escolher entre os botões “sim” ou “não”.

O botão “editar” permite que o usuário edite as informações da notícia e o botão “alterar *status*”, altera o status da notícia em questão, conforme apresentado na Figura 5.

Figura 5 - Minhas notícias

Fonte: Elaborada pelos autores.

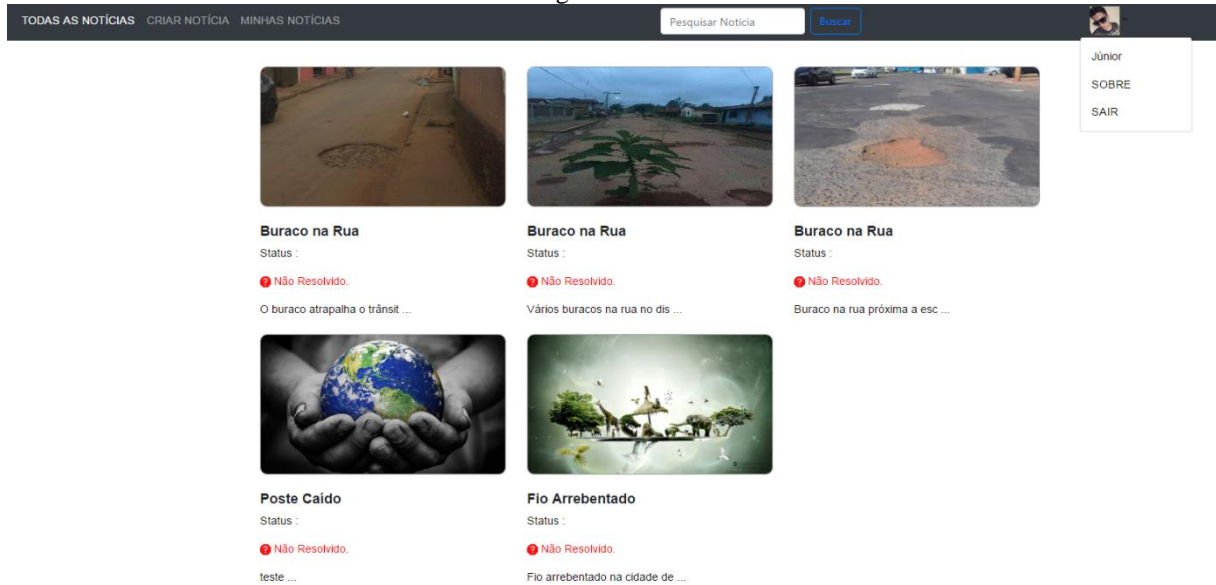
Ao clicar no botão “editar”, uma página semelhante à de criar notícias é aberta para que o usuário possa realizar as edições nos campos necessários, conforme mostra a Figura 6.

Figura 6 - Editar

Fonte: Elaborada pelos autores.

Quando o botão de seta for pressionado, será aberta uma caixa com um botão contendo o nome do usuário, e os botões “sobre” e “sair”, conforme apresenta a Figura 7.

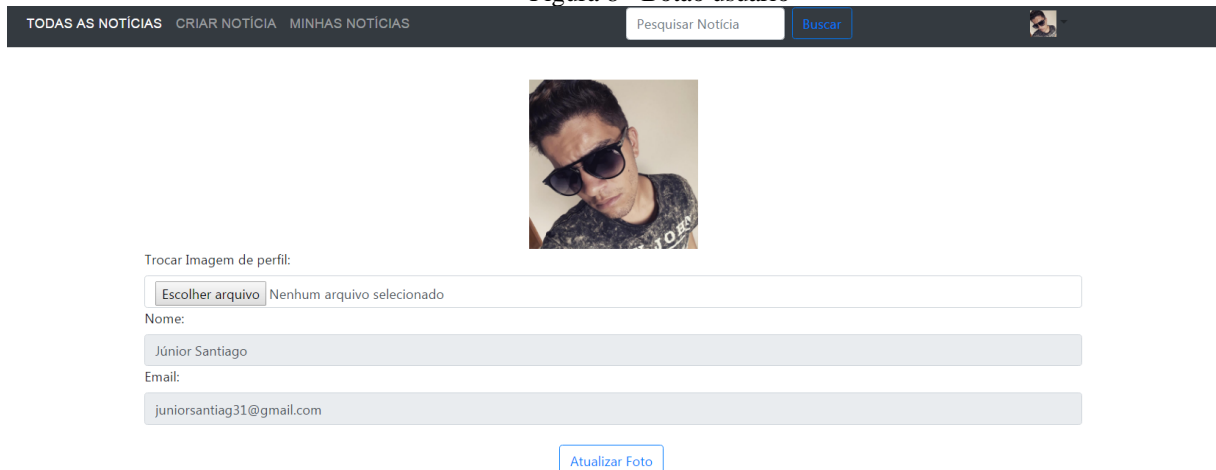
Figura 7 - Seta



Fonte: Elaborada pelos autores.

Ao clicar no botão contendo o nome do usuário, o sistema irá apresentar uma tela contendo a foto de perfil atual, um campo para atualizar a foto e dois campos fixos com nome e e-mail do usuário, de acordo com a Figura 8.

Figura 8 - Botão usuário



Fonte: Elaborada pelos autores.

O botão “sobre” exibe uma página contendo o logotipo do sistema, um campo com uma descrição da aplicação e logo abaixo um campo contendo fotos e informações sobre os desenvolvedores, conforme a Figura 9.

Figura 9 - Sobre

TODAS AS NOTÍCIAS CRIAR NOTÍCIA MINHAS NOTÍCIAS

Pesquisar Notícia

URBAN

Sabe-se que existem inumeros problemas de infraestrutura nas cidades, são estes buracos em ruas, fios de energias arrebitados, esgotos a céu aberto, entre inumeros outros. A não solução deste muitas vezes é devido a esta informação não chegar aos órgãos responsáveis.Tendo em vista este problema foi desenvolvido o URBAN.

O URBAN permite que a população ao encontrar um problema, por meio da aplicação possa fazer uma postagem sobre este, indicando sua localização, anexando foto e o descrevendo. Estes problemas postados pela população estarão disponíveis para consulta na aplicação, levando ao órgãos responsáveis uma nova forma de identificarem os problemas de infraestrutura da cidade.

O URBAN é uma aplicação desenvolvida como Trabalho de Conclusão de Curso(TCC) do curso Bacharelado em Sistemas de Informação do Instituto Federal de Minas Gerais (IFMG) - Campus São João Evangelista.

DESENVOLVEDORES:


José Vicente Júnior dos Santos Santiago
juniorsantiag31@gmail.com


Joel Kennedy da Silva
joelkennedy3@hotmail.com

Fonte: Elaborada pelos autores.

Figura 10 - Telas mobile conjunto 1

77% 15:07

URBAN

E-mail

Senha

Ainda não é cadastrado? CLIQUE AQUI!

77% 15:08

CRIAR NOTÍCIA

Tipo de Problema:
Poste Caído

Descrição:

Localização do Problema:

Mapa Satélite

Adicionar Imagem:
Escolher arquivo Nenhum arquivo selecionado

77% 15:07

Buraco na Rua

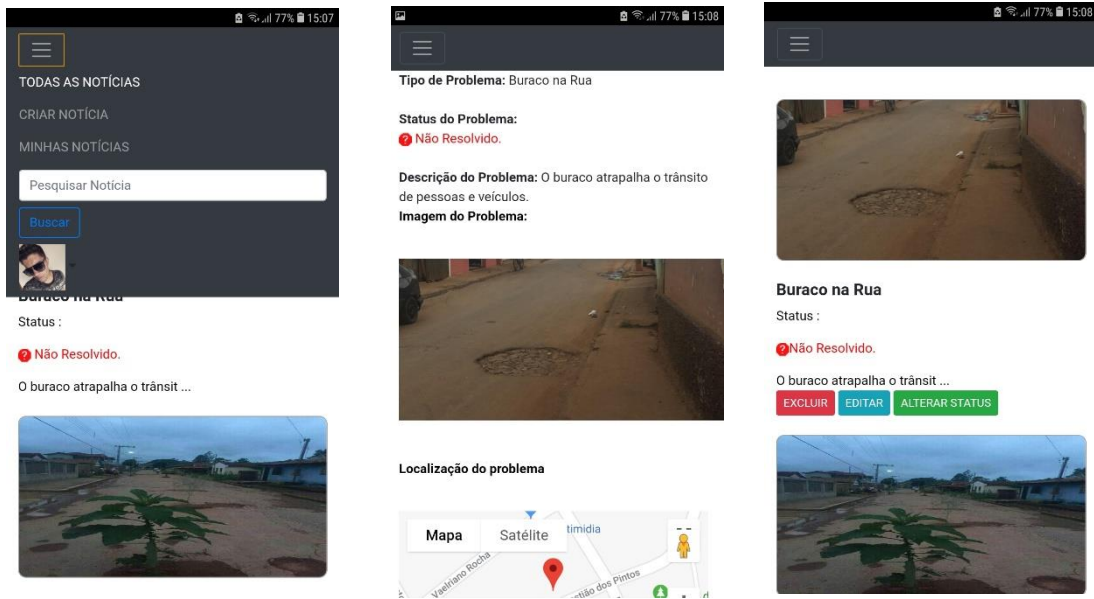
Status :
● Não Resolvido.

O buraco atrapalha o trânsito ...

Buraco na Rua

Fonte: Elaborada pelos autores.

Figura 11 - Telas mobile conjunto 2



Fonte: Elaborada pelos autores.

4. RESULTADOS E DISCUSSÃO

Esta seção apresenta os resultados obtidos através de um questionário aplicado aos envolvidos na fase de testes. A análise utilizada na pesquisa qualitativa, para com os dados do questionário aplicado foi realizada de modo descritivo, considerando que o objetivo é qualificar a experiência do usuário mediante o uso do *software*.

4.1. ANÁLISE QUALITATIVA DO SOFTWARE

Sobre o design da aplicação (questões 1, 2 e 3), segundo os usuários, o *software* possui um *design* agradável com campos de fácil identificação. Além disso, obteve-se nas respostas, que a aplicação se adaptou perfeitamente a tela dos dispositivos.

Na questão 4, que de acordo com usuários o tempo de resposta da aplicação é satisfatório, sendo adequado ao comportamento do processamento das informações.

Com relação às questões 5 e 6 os envolvidos nos testes disseram que a aplicação é fácil de utilizar, e informaram ainda que durante o uso de acesso não surgiu nenhuma dúvida. Segundo Preece *et al* (2002) a usabilidade permite avaliar se um sistema baseado em alguns fatores, entre os fatores mais comum que esta deve atender estão: facilidade de aprendizado, facilidade de uso, flexibilidade, produtividade e satisfação do usuário.

Na questão 7 estes informaram a aplicação possui um design agradável com campos de fácil entendimento. Também disseram que a aplicação é muito bem estruturada.

Com relação às questões de 8 a 10 segundo os respondentes, a aplicação atendeu a todas as especificações que foram descritas no questionário. Informaram, ainda, que esta atendeu as suas expectativas, e acreditam que a aplicação é adequada para o problema. Outros afirmam que muitas pessoas utilizam as redes sociais para notificar problemas e que é útil ter um *software* exclusivamente para isto. Por fim, ainda acrescentaram que a aplicação tem muito agregar e que ainda poderia agregar novas funcionalidades, porém isto não interferiu na qualidade da aplicação atual nem interfere na experiência de uso dos usuários. Para Fischer (1998), a viabilidade também determina a qualidade do sistema, esta mostra o quanto o sistema é útil para o contexto em que foi projetado e em que outro contexto este também pode ser útil.

A questão 11 foi de cunho discursivo para os respondentes apresentarem críticas, sugestões ou dar um *feedback* sobre alguma questão que não tenha sido abordada na pesquisa. Segundo os respondentes o resultado foi positivo, visto que ficou claro que a aplicação é adequada para o problema, além de mostrar que está de acordo com suas especificações e que a aplicação é de fácil utilização o que possibilita que mesmo pessoas com poucos conhecimentos de tecnologia possam utilizá-la, sendo acessível para toda a população.

5. CONSIDERAÇÕES FINAIS

O *software* se encontra finalizado, apresentando as funcionalidades previstas. Todos os objetivos e etapas propostas nesse projeto foram concluídos, tais como o correto funcionamento, o atendimento de todas as funcionalidades propostas e todas as especificações efetuadas conforme planejado. Entretanto por ser uma PWA percebeu-se que outras funcionalidades podem ser acrescentadas.

O presente trabalho contribuiu para que os cidadãos possam ter um ambiente virtual onde compartilhar os problemas infraestruturais de sua cidade e acessá-lo com facilidade onde estiver através de diferentes dispositivos.

Este obteve boa anuência por parte dos envolvidos nos testes, e obteve *feedbacks* valiosos que ajudaram a enriquecer a pesquisa. Percebe-se que o URBAN teve aceitação devido ao fato de sua fácil utilização, além da praticidade de poder acessá-lo de qualquer dispositivo. O *software* de acordo com os testes se adequa ao problema em questão, o que é um ponto positivo, visto que este é seu objetivo.

Como proposta para trabalhos futuros o *software* pode explorar mais os benefícios da PWA, como a utilização de *Services Workers*, mobilidade, responsividade e fácil adaptação para que possa permitir ao usuário postar um problema quando estiver *off-line* ou a aplicação funcionar *off-line*.

Atualmente no *software* desenvolvido existem dois tipos de usuários: Administrador e Usuário Público, na qual seria viável a criação de um terceiro tipo de usuário, dos setores responsáveis pela manutenção da infraestrutura da cidade para serem intermédio entre o Usuário Público e o Administrador.

Outra funcionalidade que pode ser agregada à aplicação é um campo para que os usuários adicionem comentários sobre as notícias, permitindo assim, que os mesmos debatam sobre o tema e troquem informações dentro do sistema.

REFERÊNCIAS

ARAÚJO, M. A. P. Modelagem De Dados: Teoria E Prática. **Saber Digital: Revista Eletrônica do CESVA**, Valença, v. 1, n. 1, p.33-69, ago. 2008.

ATER, Tal. **Building Progressive Web Apps**. Sebastopol: O'reilly Media, 2017.

BACCILI, Valéria Cristina Leite; DELFINO, Sérgio Roberto; VIEIRA, Fernanda Cristina. Aplicabilidade da Tecnologia da Informação no Agronegócio. **RETEC: Revista de Tecnologias**, São Paulo, v. 4, n. 1, 2011.

BONDANC, N.H.N; BRAGA, T.V; BRANCO, R.L.J.Q. **Museus Virtuais e Ecomuseus - Uma experiência fazendo uso de IoT**. In: Congresso Brasileiro de Informática na Educação, 5., 2017, Recife; Anais dos Workshops do VI Congresso Brasileiro de Informática na Educação, 5., 2017, Recife.

BRASIL. **Panorama da biodiversidade nas cidades: ações e políticas**. Ministério do Meio Ambiente. Disponível em: <www.mma.gov.br/publicacoes/cidades-sustentaveis/category/138-gera>. Acesso em: 12 abr. 2018.

CHAMMAS, Adriana *et al.* Um enfoque ergonômico sobre a metodologia de design de interfaces digitais para dispositivos móveis. **Arcos Design**, Rio de Janeiro, v. 2, n. 7, p.145-171, dez. 2013.

CHARKAOUI, S. *et al.* **Cross-platform Mobile Development Approaches**. In: IEEE International Colloquium in Information Science and Technology (CIST), 3, 2014, Casablanca.

DELAMARO, Márcio Eduardo *et al.* **Introdução ao teste de software**. 4. ed. São Paulo: Campus, 2007.

DEVMEDIA. 2018. **Introdução ao framework PHP CodeIgniter**. Disponível em: <<https://www.devmedia.com.br/introducao-ao-framework-php-codeigniter/27346>>. Acesso em: 10 jul. 2018.

FALBO, R. A. **Engenharia de software**. 2005. Disponível em: <https://inf.ufes.br/~falbo/files/ES/Notas_Aula_Engenharia_Software.pdf>. Acesso em: 14 jun. 2018.

FERREIRA, Fernando. **Resumo do Livro Engenharia de Software de Ian Sommerville. Oitava Edição.** Disponível em: <<https://codigocomcafe.wordpress.com/2013/05/17/resumo-do-livro-engenharia-de-software-de-ian-sommerville-8o-ed>>. Acesso em: 14 abr. 2018.

FERREIRA, Guilherme Fischmann. **Arquitetura da informação no desenvolvimento de aplicação web.** 2015. xii, 53 f., il. Monografia (Bacharelado em Ciência da Computação) - Universidade de Brasília, Brasília, 2015.

FISCHER, G. **Beyond ‘Couch Potatoes’: From Consumers to Designers.** In: Proceedings of the 5th Asia Pacific Computer-Human Interaction Conference. IEEE Computer Society, 1998.

FISHER, A.; SHARKIE, C. **Jump Start Responsive Web Design.** Victório - Austrália: SitePoint, 2013.

FRANSSON, Rebeca; DRIAGUINE, Alexander. **Comparing Progressive Web Applications with Native Android Applications:** An evaluation of performance when it comes to response time. 2017. 58 f. Curso de Computer Science, Department Of Computer Science, Linnaeus University, Växjö, 2017.

FURLAN, J.D. **Modelagem de objetos através da UML.** São Paulo: Makron Books, 1998.
GAUNT, Matt. **Service Workers: uma Introdução.** 2017. Disponível em: <<https://developers.google.com/web/fundamentals/primers/service-workers/>>. Acesso em: 20 abr. 2018.

GIL, Antônio Carlos. **Como elaborar projetos de pesquisa.** 4. ed. São Paulo: Atlas, 2016.

GOODCHILD, Mike. *et al.* **Sistemas de Informação Geográfica e Ciências.** 3. ed. Danvers: Third Édition, 2011.

GOOGLE DEVELOPERS. **A new way to deliver amazing user experiences on the Web.** Disponível em: <developers.google.com/Web/progressive-Web-apps/>. Acesso em: 12 abr. 2018.

GOOLGE DEVELOPERS. **Aplicativos da Web progressivos.** Disponível em: <<https://developers.google.com/web/progressive-web-apps/>>. Acesso em: 15 abr. 2018.

GOOLGE DEVELOPERS. **Service Workers: uma Introdução.** Disponível em: <<https://developers.google.com/Web/fundamentals/primers/service-workers/?hl=pt-br>>. Acesso em: 10 abr. 2018.

GUDWIN, Ricardo R. **Introdução à Linguagem UML**. 2010. Disponível em: <<http://www.dca.fee.unicamp.br/~gudwin/ftp/ea976/Estruturais2010.pdf>>. Acesso em: 16 abr. 2018.

GUEDES, Gilleanes T. A. **UML 2: Guia Prático**. 2. ed. São Paulo: Novatec, 2014.

HOWE, Jeff. **O Poder das Multidões: porque a força da coletividade está remodelando o futuro dos negócios**. Rio de Janeiro: Elsevier, 2009.

HUDSON, W. R.; HAAS, R.; UDDIN, W. **Infrastructure management**. New York: Mcgraw-hill, 1997.

HULL, Rodney; KATETE, Rodrick; NTWASA, Monde. **Therapeutic potential of antimicrobial peptides from insects**. 2005. 17 f. Monografia (Especialização) - Curso de Biologia Molecular e Celular, Universidade de Witwatersrand, Joanesburgo, 2005. Disponível em: <[http://www.academicjournals.org/article/article1380122475_Hull et al.pdf](http://www.academicjournals.org/article/article1380122475_Hull_et_al.pdf)>. Acesso em: 17 jun. 2018.

KAPPEL, G. *et al.* **An Introduction to Web Engineering: The Discipline of Systematic Development of Web Applications**. Hoboken: John Wiley & Sons, 2006.

LAMSWEERDE, A. **Requirements Engineering in the Year 00: A Research Perspective**. In: International Conference on Software Engineering, 22., 2000, Limerick. Proceedings ACM, 2000, p. 5-19.

LEITE J.C.S.P. **Engenharia de Requisitos**. Rio de Janeiro: PUC-RIO, 1994.

LINUX, Escola. **Servidor Web: O que é e como escolher um para seu site**. 2016. Disponível em: <<https://www.escolalinux.com.br/blog/servidor-web-o-que-e-e-como-escolher-um-para-seu-site>>. Acesso em: 20 out. 2018.

MAIA, R. **Redes cívicas e internet: do ambiente informativo denso as condições da deliberação pública**. Belo Horizonte: UFMG, 2002.

MALAVOLTA, Ivano. **Beyond native apps: Web technologies to the rescue! (Keynote)**. In: International Workshop on Mobile Development, Mobile, 1., 2016, New York.

MASCARO, J.L. **Manual de Loteamento e Urbanização**. Porto Alegre: SAGRA/ D.C. Luzzatto, 1994.

MASSOL, V.; HUSTED, T. **JUnit in Action**. Manning Publications Co. Greenwich, 2003.

MICROSOFT DEVELOPER. **O que é o desenvolvimento para o desktop?** Disponível em: <<https://msdn.microsoft.com/pt-br/ff380143.aspx>>. Acesso em: 13 jun. 2018.

MILANI, A. **MySQL-guia do programador**. Novatec Editora, 2007.

MINAYO, Maria Cecília de Souza. **Pesquisa social: teoria, método e criatividade**. 29. ed. Petrópolis: Vozes, 2010.

MOZILLA DEVELOPER. **Web App Manifest**. Disponível em: <<https://developer.mozilla.org/pt-BR/docs/Web/Manifest>>. Acesso em: 09 abr. 2018.

MURAROLLI, P. L; GIROTTI, M. T. **Inovações Tecnológicas nas Perspectivas Computacionais**. São Paulo: Biblioteca 24 Horas, 2015.

NERO, Marcelo Antônio; *Et al.* **Desenvolvimento De Sig 3d Com Opensource Para A Área De Distribuição De Energia Elétrica**. Revista Brasileira de Cartografia No 62/01. São Paulo. 2010. Disponível em: <<http://www.rbc.lsie.unb.br/index.php/rbc/article/view/235/226>>. Acesso em: 09 dez. 2018.

NIELSEN, J. **Heuristic Evaluation**. New York, NY: John Wiley & Sons, 1994.

OLIVEIRA, E.; FLECK, L.; BECKER, M. **Êxodo rural e sua problemática em relação ao meio ambiente e sociedade civil**. In: Seminário Internacional dos Espaços de Fronteira, 1., 2011, Paraná; Seminário Regional sobre Território, Fronteira e Cultura, 3., 2011, Paraná; Expedição Geográfica da Uniãoeste: espaços de Fronteira - Território e Ambiente, 7., 2011, Paraná. Paraná: Universidade Estadual do Oeste do Paraná, 2011.

OLIVEIRA, Juliane Nascimento. **O Planejamento da Infraestrutura Urbana para o Desenvolvimento do Turismo, Baseado nos Conceitos Descritos no Estatuto da Cidade e na Boa Forma da Cidade**. In: V Fórum Internacional De Turismo Do Iguassu, 5., 2011, Foz do Iguaçu. Artigo. Foz do Iguaçu: Festival de Turismo das Cataratas do Iguaçu, 2011. p. 1 - 16. Disponível em: <<http://festivaldeturismodascataratas.com/wp-content/uploads/2014/01/3.-O-PLANEJAMENTO-DA-INFRA-ESTRUTURA-URBANA-PARA-O-DESENVOLVIMENTO-DO-TURISMO.pdf>>. Acesso em: 17 jun. 2018.

OLIVEIRA, M. P. G. **Sistema Espacial de Apoio à Decisão: Modelos para análise do adensamento de atividades econômicas no espaço urbano**. Dissertação (Mestrado), Escola de Governo de Minas Gerais da Fundação João Pinheiro, Belo Horizonte, 1997.

OMG. **OCL 2.0 Specification**. 2005. Disponível em:
<<https://www.omg.org/spec/OCL/2.0/About-OCL/>>. Acesso em: 14 abr. 2018.

OMG. **Unified Modeling Language: diagram interchange**. 2005. Disponível em:<<https://klevas.mif.vu.lt/~plukas/resources/UML/UML20/05-06-04.pdf>>. Acesso em: 14 abr. 2018.

OMG. **Unified Modeling Language: Infrastructure**. 2006. Disponível em:<<http://sysml.org/docs/specs/OMGSysML-v1.0-07-09-01.pdf>>. Acesso em: 14 abr. 2018.

OMG. **Unified Modeling Language: superstructure**. 2005. Disponível em:<http://www.dca.fee.unicamp.br/~gudwin/ftp/ea977/UML-Superstructure_2.3.pdf>. Acesso em: 14 abr. 2018.

PAULA FILHO, Wilson de Pádua. **Engenharia de Software: fundamentos, métodos e padrões**. S. 1: LTC, 2000. 260 p. Disponível em: <http://aulasprof.6te.net/Arquivos_Aulas/07-Proces_Desen_Soft/Livro_Eng_Soft_Fund_Met_Padrees.pdf>. Acesso em: 18 jun. 2018.

PREECE, J.; ROGERS, Y.; SHARP, E. *Interaction Design: Beyond Human-computer Interaction*. New York, NY: John Wiley & Sons. 2002.

PRESSMAN, R. **Engenharia de Software**. Rio de Janeiro: McGraw Hill, 2002.

PRESSMAN, Roger S; LOWE, David. **Engenharia Web**. Tradução de Daniel Vieira. Rio de Janeiro: LTC, 2009.

PUPPI, I. C. **Estruturação Sanitária das Cidades**. Curitiba, Universidade Federal do Paraná, 1981.

RAJ, C. R.; TOLETY, S. B. **A study on approaches to build cross-platform mobile applications and criteria to select appropriate approach**. In: Annual IEEE India Conference (INDICON), .7, 2012, Cochin.

REDDA, Yonathan Aklilu. **Cross platform Mobile Applications Development: Mobile Apps Mobility**. 2012. 77 f. Dissertação (Mestrado) - Curso de Sistemas de Informação, Norwegian University Of Science And Technology, Trondheim, 2012.

ROGERS, Y.; SHARP, H.; PREECE, J. **Design de Interação: além da interação humano-computador**. 3. ed. Porto Alegre: Bookman, 2013.

SANTOS, Claudia Benedita dos; *Et al.* **Utilização de um Sistema de Informação Geográfica para descrição dos casos de tuberculose.** Bol. Pneumol. Sanit. Rio de Janeiro. 2004. Disponível em: < http://scielo.iec.gov.br/scielo.php?pid=S0103-460X2004000100002&script=sci_arttext&tlng=en>. Acesso em: 09 dez. 2018.

SOMMERVILLE, I. **Engenharia de Software.** 9. ed. São Paulo: Pearson Prentice Hall, 2011.

SOMMERVILLE, I.; SAWYER, P. **Requirements Engineering Good Practice Guide.** Hoboken: John Eiley e Sons Ltd, 1997.

SILVA, J.X. **Geoprocessamento e Análise Ambiental.** Revista Brasileira de Geografia, Rio de Janeiro, v. 54, n. 03 jul/set, 1992, pp. 47-61.

SUN, Yeran; MOBASHERI, Amin. **Utilizing Crowdsourced Data for Studies of Cycling and Air Pollution Exposure: A Case Study Using Strava Data.** Int. J. Environ. Res. Public Health. 2017. Disponível em: < <https://www.mdpi.com/1660-4601/14/3/274/html>>. Acesso em: 09 dez. 2018.

TELLES, D. A. **Ciclo ambiental da água: da chuva à gestão.** São Paulo: Blucher, 2013.

THOMSON, L.; WELLING, L.. **PHP e MySQL Desenvolvimento WEB.** 3. ed. Campus, 2005.

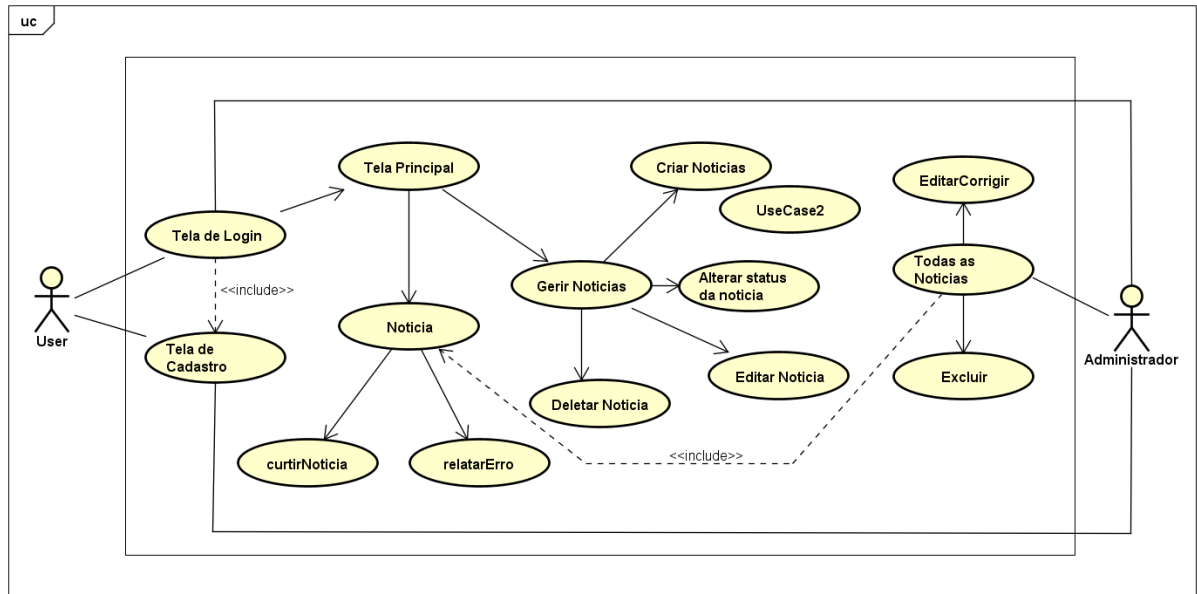
TOLEDO, Jan Miszura; DEUS, Gilcimar Divino de. **Desenvolvimento em Smartphones - Aplicativos Nativos e Web.** Disponível em: <[http://www.cpgls.pucgoias.edu.br/7mostra/Artigos/AGRARIAS EXATAS E DA TERRA/Desenvolvimento em Smartphones - Aplicativos Nativos e Web.pdf](http://www.cpgls.pucgoias.edu.br/7mostra/Artigos/AGRARIAS_EXATAS_E_DA_TERRA/Desenvolvimento_em_Smartphones_-_Aplicativos_Nativos_e_Web.pdf)>. Acesso em: 12 abr. 2018.

ZAVE, Pamela. Classification of research efforts in requirements engineering. **Acm Computing Surveys.** New York, p. 315-321. out. 1997.

ZMITROWICZ, W.; NETO, G. A. **Infraestrutura Urbana.** São Paulo: EPUSP, 1997. (Texto Técnico da Escola Politécnica da USP, Departamento de Engenharia de Construção Civil, TT/PCC/17).

APÊNDICES

APÊNDICE A - Diagrama de caso de uso



APÊNDICE B - Diagrama de classe

