

Instituto Federal de Educação, Ciência e Tecnologia de Goiás
Câmpus Formosa
Tecnologia em Análise e Desenvolvimento de Sistemas

Sistema Inteligente de Divulgação de Informações do IFG-Formosa

Autor: Matheus de Carvalho Sobrinho
Orientador: Prof. Me. Daniel Saad Nogueira Nunes

Formosa, GO
2017

Matheus de Carvalho Sobrinho

Sistema Inteligente de Divulgação de Informações do IFG-Formosa

Monografia submetida ao curso de graduação em Tecnologia em Análise e Desenvolvimento de Sistemas do Instituto Federal de Educação, Ciência e Tecnologia de Goiás, câmpus Formosa, como requisito parcial para obtenção do título de Tecnólogo em Análise e Desenvolvimento de Sistemas.

Instituto Federal de Educação, Ciência e Tecnologia de Goiás

Câmpus Formosa

Orientador: Prof. Me. Daniel Saad Nogueira Nunes

Formosa, GO

2017

Matheus de Carvalho Sobrinho

Sistema Inteligente de Divulgação de Informações do IFG-Formosa/ Matheus
de Carvalho Sobrinho. – Formosa, GO, 2017-
62 p. : il. (algumas color.) ; 30 cm.

Orientador: Prof. Me. Daniel Saad Nogueira Nunes

Trabalho de Conclusão de Curso – Instituto Federal de Educação, Ciência e Tec-
nologia de Goiás
Câmpus Formosa , 2017.

1. Veiculação. 2. Divulgação.

CDU 004.42

Matheus de Carvalho Sobrinho

Sistema Inteligente de Divulgação de Informações do IFG-Formosa

Monografia submetida ao curso de graduação em Tecnologia em Análise e Desenvolvimento de Sistemas do Instituto Federal de Educação, Ciência e Tecnologia de Goiás, câmpus Formosa, como requisito parcial para obtenção do título de Tecnólogo em Análise e Desenvolvimento de Sistemas.

Trabalho aprovado. Formosa, GO, 13 de Março de 2017:

Prof. Me. Daniel Saad Nogueira
Nunes
Orientador

Prof. Me. Sirlon Thiago Diniz Lacerda
Convidado 1

Prof. Esp. Murilo de Assis Silva
Convidado 2

Formosa, GO
2017

Resumo

Este trabalho apresenta o Sistema Inteligente de Divulgação de Informações do IFG - Formosa (SID) que, utilizando-se de conceitos de sinalização digital, veicula informações a serem repassadas no ambiente do Câmpus de forma simples, efetiva, interativa e dinâmica. De modo a alcançar estas características, uma série de decisões foram realizadas, como projetar a sua arquitetura de modo a flexibilizar os dispositivos compatíveis com o formato das divulgações e a inserção de códigos QR para que o usuário do sistema consiga interagir com o SID e obter detalhes sobre a informação repassada. O SID também possui a integração com a rede social Facebook ao realizar publicações em um perfil específico sempre que uma nova divulgação é inserida, isto possibilita a unificação de veículos distintos de disseminação de informações, um baseado em sinalização digital e outro em redes sociais.

Palavras-chaves: Veiculação. Sinalização digital. Facebook.

Abstract

This work presents a system entitled as “Sistema Inteligente de Divulgação de Informações do IFG - Formosa” (SID), which utilizes concepts of digital signage to propagate informations on IFG - Formosa *campus*’ environment in a simple, effective, interative and dynamic fashion. In order to achieve those features, a series of decisions were made such as: project the system architecture in such a way to allow several kinds of devices to use SID and to insert QR codes into the propagations in order to enable the users obtaining the details of such propagations being served. The system also has an integration with the social network Facebook, as it publishes the information in a specific profile when they are inserted into the system. This represents an unification of two different mechanisms of information dissemination, one based on digital signage and other on social media.

Key-words: Propagation. Digital signage. Facebook.

Sumário

1	INTRODUÇÃO	11
1.1	Motivação	13
1.2	Objetivos	14
1.2.1	Objetivos Gerais	14
1.2.2	Objetivos Específicos	15
1.3	Metodologia	15
1.4	Organização do Documento	16
2	REFERENCIAL TEÓRICO	19
2.1	Sinalização digital	19
2.2	Raspberry Pi	20
2.2.1	Especificações da plataforma	20
2.3	Arquitetura Cliente-Servidor	21
2.4	Linguagens de programação	23
2.5	Banco de dados	24
2.6	Interface de programação de aplicativos	25
2.7	Linguagem de modelagem unificada	26
2.8	Metodologia de desenvolvimento SCRUM	27
2.9	Zend Framework e Doctrine	28
2.10	Código QR	29
2.11	Trabalhos relacionados	30
3	SID	33
3.1	Visão Geral	33
3.1.1	Módulo administrador	34
3.1.2	Módulo cliente	37
3.2	Arquitetura do SID	38
3.2.1	Solução para implantação do SID	43
4	INTEGRAÇÃO	45
4.1	Visão Geral	45
4.1.1	Elementos da API utilizados no SID	47
4.1.2	Integração	48
5	RESULTADOS	51
6	CONSIDERAÇÕES FINAIS	55

6.1 **Trabalhos futuros 56**

Referências 59

1 Introdução

De acordo com (MESSIAS, 2005), o campo de conhecimento da Ciência da Informação padece de fragilidade à medida que se tem problemas ou dificuldades no que tange a definição da diversidade, objetivos e campos de atuação da informação e seu contexto. Neste sentido, entende-se que a informação deve ser de fato definida e bem estruturada, com seus objetivos claros e sua área atuação relacionada ao contexto em que será aplicada, atenuando o descaso ou a dificuldade de assimilação por parte do interessado.

A linguagem oral, a panfletagem, o preenchimento de murais e *outdoors*, *sites*, e até mesmo o uso de *banners*, são instrumentos simples e comumente utilizados para divulgação de informação. A falta de agilidade, efetividade e interação desses meios de divulgação de informação, assim como seus custos elevados, impossibilitam a absorção de informação de maneira adequada e efetiva (BARRETO, 2002).

Os meios de comunicação rígidos, obsoletos e não interativos deixam o canal de comunicação deficiente, o que pode acarretar em uma não efetividade ou dificuldade de repasse da informação, e assim dificulta a retroalimentação e compromete a assimilação do interessado nesta informação. Neste contexto, o Instituto Federal de Educação, Ciência e Tecnologia de Goiás – IFG Câmpus Formosa, compartilha destes mesmos problemas referentes aos meios de comunicação rígidos e não interativos (BISPO; NUNES, 2015).

A necessidade de elementos mais atrativos e menos rígidos no Câmpus podem, a princípio, ser sanados com a utilização da sinalização digital. Em resumo, (MACHADO, 2010) conceitua a sinalização em sua forma digital como uma tecnologia que tem o objetivo de exibir informações por meio de painéis digitais, reforçando a ideia de veiculação da informação de forma ágil, efetiva ao interessado e preponderantemente utilizada como aparato para os problemas relacionados aos meios rígidos de veiculação de informação presentes no Câmpus.

A informação e, a partir desta, o conhecimento estão em todas as esferas e áreas de estudo ou contextos, são considerados essenciais tanto do ponto de vista cotidiano e acadêmico quanto profissional e, quando transformados pelas ações dos indivíduos, tornam-se competências valorizadas, gerando benefícios sociais e econômicos que estimulam o desenvolvimento e são, ainda, recursos fundamentais para formação e manutenção das redes sociais (TOMÁEL; ALCARÁ; CHIARA, 2005). Desta forma, a utilização da sinalização digital em conjunto às redes sociais torna-se uma forma de aproveitar esta estrutura fornecida por estas redes para que a informação a ser repassada transcenda o ambiente do Câmpus e seja repassada de forma efetiva ao interessado.

O Sistema Inteligente de Divulgação de Informações do Instituto Federal de Goiás,

Câmpus Formosa, ou simplesmente **SID**, trata-se de uma solução que almeja complementar de forma eficiente aos meios de comunicação do Câmpus. Propõe uma implementação em ambiente WEB com divisão de seus módulos de serviço para implantação em diversos ambientes ou dispositivos. O SID, tem como principal finalidade divulgar informações através da inserção em um de seus módulos de forma fácil e dinâmica, na qual o operador não precisará possuir conhecimentos que não sejam necessários para o processo de divulgação de informações no contexto em que será implantado, ou seja, suas rotinas tendem a serem fáceis e objetivas e qualquer pessoa poderá operá-lo sem obstáculos de usabilidade. Além disso, possui integração com a rede social Facebook, o que permite a unificação entre dois veículos de comunicação.

O SID consiste, em sua essência, de um sistema em plataforma WEB baseado em sinalização digital. É dividido em dois módulos de utilização, sendo um deles o módulo administrador e outro o cliente que, em conjunto, executam processos relativos respectivamente à disponibilização de informações para o interessado e apresentação ao mesmo.

O sistema tem grande potencial para suprir as necessidades de informações do Câmpus, com a disponibilização de notícias essenciais, assim como lembretes de prazos para inscrições em eventos ou cursos, prazos para elementos que subsidiam a vida de quem da informação dependa, processo executado a partir de telas digitais, como TVs de LCD ou LED espalhadas pelo Câmpus. O SID utiliza-se de código QR na interface de apresentação ao usuário final, dispondo uma forma de adentrar de forma mais efetiva na informação que está sendo repassada. Sua interface é estruturada automaticamente, fazendo com que o operador disponha de pouco trabalho para criação de uma nova divulgação, diferente do que ocorre em outras soluções.

Uma alternativa economicamente viável e utilizada neste trabalho poderá ser efetivada através de um computador *Raspberry Pi*, que tem como grandes vantagens o tamanho e consumo energético reduzidos, servindo como elemento propagador das informações exercendo função redistributiva para um monitor a ele acoplado. Outras informações sobre esta solução adotada serão descritas mais adiante no decorrer deste trabalho.

O SID será de grande importância uma vez que os instrumentos de repasse de informações no Câmpus Formosa restringem-se aos meios não interativos ou obsoletos, e isto dificulta o processo de assimilação da informação de forma ágil e eficiente.

A integração do sistema SID com a rede social Facebook tem como ponto forte a unificação entre estes sistemas que veiculam informação de forma simples e efetiva. Para realizar essa tarefa, o sistema necessita utilizar uma interface de programação de aplicativos do Facebook, denominada de Graph API ([FACEBOOK, 2017](#)) e tendo como forma de sanar um dos principais problemas encontrados nos sistemas baseados em sinalização digital utilizados atualmente, ou seja, aumentar o processo de veiculação de informação

a partir da sua integração com a rede social Facebook.

Com a significância que as redes sociais trazem para o ambiente produtivo e mercadológico, portas de disseminação de informação são abertas com elementos de integração como a Graph API e portanto, o SID deverá/poderá fazer uso dessa ferramenta para influenciar na permanência do usuário com a informação de forma efetiva, possibilitando que o usuário possa adentrar no contexto da informação também fora do Câmpus, fortalecendo o contexto a ser repassado e garantindo resultados relevantes sobre a assimilação dessa mesma informação unificada em sistemas distintos.

1.1 Motivação

As mudanças na tecnologia da informação ocorridas nos últimos 50 anos organizam e ordenam todas as atividades associadas com a ciência da informação (BARRETO, 1997). Antes destas mudanças, o homem dispunha de poucos dispositivos tecnológicos que o ajudavam a interagir com o mundo e com as informações de forma natural, usando seus sentidos em ações elementares como acionar uma alavanca ou apertar um botão (RIBEIRO; ZORZAL, 2011). Na atualidade, os elementos tecnológicos são usados como elementos de pesquisa, produção, entretenimento, *marketing* ou sinalização em sua forma digital, entre outros, onde a informação é tratada como o principal objeto a ser manipulado por esses elementos.

A sinalização, em sua forma digital, é um dos pontos que encontra-se em defasagem no IFG - Câmpus Formosa. Precisamente, a não interatividade nos meios de comunicação utilizados no Câmpus, uma vez que os possíveis meios estão voltados a estruturas de comunicação rígidas, resulta em uma assimilação da informação ineficiente e acarreta na não efetivação do objeto a ser transmitido.

Os veículos de divulgação de informações do Câmpus concentram-se na colagem de papéis em murais, ou até mesmo em locais inapropriados, nas paredes, por exemplo. Ao assumir que as únicas formas de transitar informações em uma ambiente relativamente grande como o Câmpus Formosa, aceitam-se os encargos de todos os problemas nesse processo, desde o trabalho de retirada dos papéis de informações dos murais, quando os mesmo encontrarem obsoletos, até a ineficiente assimilação do objeto a ser repassado.

Outro problema que pode ser associado a este modo de disseminação de informação como o mural é a falta de conhecimento sobre o objeto veiculado, ocasionando, muitas vezes, na não assimilação dessa informação pelo real interessado ou suposto interessado.

O SID pretende sanar tais problemas utilizando-se da sinalização digital para transmitir informações por meio de telas espalhadas pelo Câmpus e com a sua integração com a rede social Facebook, que servirá de ferramenta adicional para este trâmite de informa-

ções no Câmpus de forma efetiva e dinâmica, possuindo transparência ao operador. Com isso, o SID dispõe de ferramentas amplamente utilizadas no contexto WEB, da utilização de códigos QR em cada divulgação como modo de aproximar o interessado aos detalhes relativos à divulgação. Seu banco de dados é centralizado em um servidor, possibilitando que diversos tipos de dispositivos possam ser usados para apresentação das informações por ele repassadas. Desta forma, objetivando a manutenibilidade e portabilidade de suas funções, e podendo ser utilizado por pessoas sem nenhum conhecimento de sistemas de informação acarretando em facilidade de inserção e de usabilidade de maneira geral.

Assim, a implantação do SID para a assimilação de informação ocorre por meio de mecanismos que induzem o conteúdo ao interessado, eliminando a ideia de procura, a facilitar e diminuir o caminho de entrega e na efetivação da informação.

A sinalização digital proporciona ao meio uma forma atrativa, ágil e moderna no que tange ao trâmite de informações. Essa forma de divulgação de informações traz consigo mecanismos de publicidade com incontáveis possibilidades de interação com o interessado na informação. O SID torna possível a sinalização digital na realidade do Câmpus.

A estrutura do ambiente do IFG - Câmpus Formosa proporciona a sinalização em sua forma digital vantagens significativas, pois possui locais apropriados para veiculação de informação no Câmpus de modo a maximizar a assimilação da mesma, como pontos de transição e de lazer como a lanchonete.

A integração de um sistema como o SID com o Facebook atrai usuários, além de automatizar a dinâmica de atividades do sistema e, como elemento mais factível, propicia a união dos dois sistemas com propósito de divulgação de uma mesma informação através de diferentes meios.

1.2 Objetivos

1.2.1 Objetivos Gerais

Como principal objetivo deste trabalho, está o desenvolvimento do Sistema Inteligente de Divulgação do IFG - Formosa para que supra as necessidades, e contribua para divulgação de informações no contexto do Câmpus de forma atrativa e dinâmica.

O sistema deverá ser capaz de proporcionar objetividade e simplicidade das informações a serem repassadas e ser integrado à rede social Facebook para que as informações disponibilizadas no SID sejam também inseridas fora do ambiente do Câmpus e dentro dessa rede social a partir do perfil do SID no Facebook, com a finalidade de unificar diferentes veículos de comunicação.

1.2.2 Objetivos Específicos

- Colocar o trabalho no limiar dos mundos acadêmico e mercadológico abrangendo de forma consistente o que é aplicado atualmente na área da sinalização digital e o que pode ser efetivado no Câmpus.
- Desenvolver o sistema SID a partir da metodologia de desenvolvimento ágil SCRUM.
- Proporcionar autonomia ao sistema SID no que tange sua integração e os dados a serem por ele, fornecendo junto ao Facebook mecanismos de controle fácil e intuitivo para o administrador do SID.
- Abordar da forma minuciosa a visão do sistema a ser repassada ao usuário convencional, e exemplificar a sua arquitetura e condições para codificação da função a ser executada por esse sistema.
- Inserir códigos QR nas divulgações apresentadas pelo módulo cliente de modo a viabilizar mais detalhes intrínsecos às divulgações.
- Utilizar o método de autenticação fornecida pela Graph API como método padrão de acesso ao módulo administrador do SID.
- Integrar o módulo administrador com a Graph API para que todas as divulgações inseridas no SID sejam, de forma transparente, postadas no perfil do SID no Facebook.
- Propor uma metodologia de implantação do sistema SID em conjunto com a plataforma *Raspberry Pi*, objetivando economia energética, financeira e de espaço físico.

1.3 Metodologia

A inserção de um sistema de sinalização digital para resolução de problemas relativos aos meios rígidos de comunicação é a principal diretriz para efetivação deste trabalho.

Um levantamento do estado da arte deve ser realizado para que seja possível identificar as possíveis soluções e suas vantagens e deficiências. A partir deste levantamento, será possível obter uma base de comparação com o SID e a perspectiva do que é desejável em um sistema de sinalização digital que melhor adequa-se à realidade do IFG, câmpus Formosa.

Ao utilizar-se da revisão de bibliografia como meio de direcionamento do trabalho e a partir do uso de comparações entre ferramentas desenvolvidas com o propósito de sinalização digital, tenta-se analisar tais soluções com o intuito de suprimir os pontos negativos e concatenar ao processo de desenvolvimento os elementos principais que são

responsáveis por efetivar a veiculação de informação ao sistema de forma descentralizada com o auxílio de ferramentas utilizadas no contexto WEB.

A metodologia presente neste trabalho está direcionada aos aspectos específicos do desenvolvimento de ferramentas computacionais para melhoria nos processo de comunicação e veiculação de informações. Este desenvolvimento é inserido no contexto de desenvolvimento ágil, que possui a finalidade de desenvolver sistemas em pequeno período de tempo através da metodologia SCRUM (COHN, 2000) que, resumidamente, consiste de com encontros semanais (*sprints*) para definição da funcionalidade a ser desenvolvida no sistema. Seguindo esta diretriz, a estruturação de sistemas que utilizam esta metodologia de desenvolvimento tendem a possuir resultados parciais no decorrer de cada *Sprint*. Esta metodologia encaixa-se bem no desenvolvimento do SID por tratar-se de um sistema que possui escopo e cronograma reduzido.

A partir da utilização da metodologia SCRUM no desenvolvimento do SID são incorporados nos diversos *sprints* semanais os objetivos de desenvolvimento das funcionalidades do sistema, como a implementação dos dados gerenciados a nível de banco de dados, modelagem e manutenção do sistema e suas respectivas paginas de gerenciamento, inserção códigos QR na estrutura do sistema e integração do SID ao Facebook.

1.4 Organização do Documento

O Capítulo 2, de Referencial Teórico, tem como objetivo colocar o leitor a par dos elementos que serão utilizados tanto para o desenvolvimento do sistema SID quanto para conceituar as ferramentas utilizadas neste trabalho. Desta forma, este Capítulo repassa brevemente os conceitos relacionados à sinalização digital, aspectos relacionados aos computadores da plataforma *Raspberry Pi*, conceitos da arquitetura Cliente-Servidor, descreve também linguagens de programação e seus conceitos relacionados, banco de dados com o modelo relacional de forma preponderante e por fim é mostrado os atributos que permeiam as Interfaces de Programação de Aplicativos - API's, além de abordar o estado da arte do trabalho, o que possibilita ter a visão das soluções encontradas em implantação no mercado atual.

A diante, no Capítulo 3, são descritos os aspectos intrínsecos do sistema SID, como a visão geral destes, a divisão em módulos de gerenciamento em administrador e cliente e a descrição de sua arquitetura através da Linguagem de Modelagem Unificada - UML. Também neste capítulo contextualiza-se o seu banco de dados e por fim é descrita a solução de implantação adequada para o contexto do Câmpus.

O Capítulo 4 aborda o processo de integração do SID com a rede social Facebook. Inicialmente descreve-se a estruturação da Graph API . Posteriormente é descrito como foi desenvolvida esta integração e quais foram os resultados iniciais dela.

Os resultados deste trabalho são expostos no Capítulo 5, e tem como principal objetivo demonstrar os aspectos relativos ao à construção e integração do sistema SID bem como comparar o sistema com outras soluções presentes na literatura.

Por fim, o Capítulo 6, organiza as considerações finais do trabalho e repassa ao leitor o horizonte de possibilidades que o SID viabilizou, ao descrever os trabalhos futuros tanto relativos às funcionalidade que podem ser inseridas na estrutura do sistema quanto às diversas possibilidades de integração com a própria rede social Facebook.

2 Referencial Teórico

Este capítulo tem a intenção de inserir, de forma superficial, o leitor no contexto dos conceitos e ferramentas utilizadas neste trabalho, aproximando-o desde aqueles relacionados à sinalização digital até ferramentas amplamente utilizadas para comunicação entre Interfaces de Programação de Aplicativos ou API's como o JSON.

2.1 Sinalização digital

Muitas especializações no campo do *design* gráfico surgiram no decorrer do ultimo século, com mais foco na informação sendo repassada, sendo por meio da linguagem, da imagem ou dos símbolos e desdobrou-se em vários caminhos, entre esses a sinalização digital ([CARDOSO et al., 2011](#)).

Ainda de acordo com ([CARDOSO et al., 2011](#)), a sinalização é o processo de veiculação de informações, no qual tem-se o princípio de marcar ou sinalizar algo, transmitir informação mediante um disposição adequada de sinais, regulamentando o fluxo de pessoas e veículos, preferencialmente antecipando a demanda. De forma direta, é possível definir a sinalização digital como um produto de *design* utilizado para orientar, informar e guiar os usuários com informações repassadas por elementos digitais, como *banners* em sites *webs* ou TVs distribuídas em locais difusos.

Embora o termo sinalização digital possa ser genérico e bastante intuitivo, à medida que foi sofrendo modificações ao longo dos anos, atrelou novos atributos, como por exemplo: o relacionamento com o ambiente, introdução de cores realistas e a completa concatenação com a Informática. De acordo com ([CARDOSO et al., 2011](#)), um projeto de sinalização digital pode ser caracterizado pelo resultado da combinação de seus vários subsistemas, podendo ser, informativos, gráficos, físico/formal, construtivo, ambiental, de acessibilidade e segurança e normativo.

Um projeto de sinalização digital pode ser capaz de alterar o ambiente onde o mesmo é enquadrado. Segundo ([UEBELE, 2007](#)), os projetos de sinalização que entram para fornecer um sistema gráfico completo e de forma eficiente são capazes de dar a vez a uma identidade ou aparência e transformar a natureza dos ambientes em que elas ocupam, a fim de fornecer orientação para seus usuários.

Os benefícios ligados a um projeto de sinalização devidamente estruturado de forma gradativa de acordo com o *feedback* dos usuários do mesmo podem ser associados de forma análoga ao conceito de assimilação da ciência proposta por ([ALBAGLI, 1996](#)), onde o mesmo dispõe que há rápida assimilação na vida cotidiana dos indivíduos, dos

artefatos técnico-científicos transformados em objetos de consumo, dada a velocidade com que vêm ocorrendo as inovações nesse campo de estudo.

Estes conceitos serão aplicados ao SID, que pretende fornecer uma rápida assimilação do conteúdo a ser repassado e se adequar ao ambiente do interessado, com velocidade na logística na assimilação da informação.

2.2 Raspberry Pi

A tecnologia teve sua disseminação em escala global com a necessidade de automização dos processos produtivos a partir da revolução industrial no século XVIII, saindo da forma de produção artesanal para se chegar aos processos mecanizados e automatização de tarefas. Esse interesse em sua utilização se deu pelo fato de que existe um enorme potencial de aplicação de tecnologia nos mais variados setores industriais e ao impacto que seus resultados podem dar ao desenvolvimento tecnológico e econômico (DURAN; MORAIS; MATTOSO, 2006).

Os interesses no uso da tecnologia, podem exigir diversas restrições, tais como, economia de espaço ou até mesmo de energia elétrica, porém, atualmente quando adquire-se o pequeno espaço físico e a capacidade de operar com pequenos gastos de energia, pode ocasionar em métricas de processamento limitado, nas quais esses aparelhos nanotecnológicos na maioria das vezes são incapazes de executar funções que para um computador de mesa consideravelmente barato processa sem mais delongas.

Nesse sentido, a plataforma *Raspberry Pi* é descrita como sendo um dispositivo de pequeno porte que exige pequeno gasto de energia e é configurável e adaptável ao contexto em que se deseja aplicá-lo. De acordo com (PI, 2012), trata-se de um computador do tamanho de um cartão de crédito e de custo acessível desenvolvido pela *Raspberry Pi Foundation*, com a intenção de estimular o ensino de ciência da computação em escolas de ensino básico.

2.2.1 Especificações da plataforma

Computadores da plataforma *Raspberry Pi*, tem em sua composição, um processador que pode ser comparado ao de um celular com baixo preço na atualidade. Esta plataforma possui diversas variações no que tange frequência de processamento, conectividade com a rede e preço final ao consumidor. (RICHARDSON; WALLACE, 2013).

O meio de armazenamento persistente da plataforma *Raspberry Pi* é feita através de um cartão de memória SD. Portas USB 2.0 estão presentes em seu *hardware*, que normalmente são usadas para permitir a interação ao *software* instalado.

O *Raspberry Pi*, utiliza o sistema operacional GNU/Linux para efetivar sua fina-

lidade. O Linux é de forma abrangente, apenas o *kernel*, e um sistema operacional vem a ser algo a mais que isso, onde para melhor gerenciamento é necessário uma coleção de *drivers*, serviços e aplicações para compor de forma efetiva o sistema operacional (RICHARDSON; WALLACE, 2013). Contudo, a distribuição recomendada pela detentora dos direitos do *Raspberry Pi*, pelo baixo teor de processamento de seus dispositivos, é a *Raspbian*, baseada na distribuição *Debian*, desenvolvida de forma a suprir as necessidades na plataforma.

O conector HDMI dá suporte ao áudio e vídeo, eliminando a necessidade de outro periférico para divisão dos dois suportes, e, de acordo com (RICHARDSON; WALLACE, 2013), suporta até 14 resoluções de vídeo diferentes.

Um conector micro USB é usado para fornecer energia, e não como uma porta USB de propósito geral.

O dissipador de calor trata-se de um pequeno objeto de metal acoplado aos *chips* de modo a bastante área de superfície para proporcionar um melhor resfriamento e melhor capacidade de processamento destes *chips*. O *chipset* do *Raspberry Pi* foi projetado para aplicações móveis, de modo que um dissipador de calor não é necessário na maioria das vezes. No entanto, existem casos em que pode-se querer executar o *Raspberry Pi* em altas frequências ou processar números por um longo período de tempo, e o chip poderá então aquecer (RICHARDSON; WALLACE, 2013), tornando a utilização de um dissipador de calor desejável.

Um computador da plataforma *Raspberry Pi* pode ser visualizado na Figura 1.

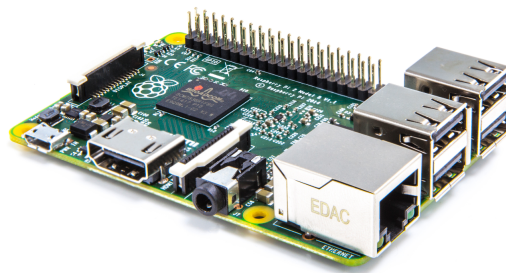


Figura 1 – Raspberry Pi 3.

2.3 Arquitetura Cliente-Servidor

De acordo com (KUROSE et al., 2010), o núcleo essencial para o desenvolvimento de aplicações de rede é a escrita de sistemas que são executados em dispositivos finais diferentes e que se comunicam entre si.

Boa parte destes sistemas adotam modelos de gerenciamento centralizado, tais como aqueles baseados no protocolo *Domain Name System - DNS*, ou aqueles baseados

no *File Transfer Protocol - FTP*, como mostrado por (KUROSE et al., 2010). Esse tipo de arquitetura tem como principal característica a gerência centralizada das informações por meio de requisição ou resposta, onde a requisição é feita por um cliente e a resposta é dada a esse cliente pelo servidor central, de acordo com os parâmetros passados na requisição.

A WEB também não é organizada de modo diferente. Tem como principais elementos:

- Protocolo HTTP: protocolo da camada de aplicação da WEB, define o formato ou até mesmo a sequência das mensagens que poderão ser passadas entre o navegador e o Servidor WEB (KUROSE et al., 2010).
- Navegador (*WEB Browser*): Lado cliente da aplicação, funciona como uma interface hipermídia que se comunica com o servidor através do protocolo HTTP, para acessar recursos ou serviços disponibilizados pelo servidor (SILVA; FERNANDES et al., 2002).
- Servidor WEB: correspondem ao lado servidor da aplicação, responde as requisições dos clientes com as respostas HTTP.
- Sistemas *Helpers*: são os subsistemas que permanecem na máquina servidora, ou seja, recebem requisições, fazem o tratamento necessário e devolvem a resposta do processo ao cliente. Podem ser escritos em linguagens WEB como Java ou PHP.

Os navegadores são o lado cliente de uma aplicação HTTP. São tachados como aplicações que disponibilizam o serviço de forma dinâmica e simples aos usuários de fato. Exemplos de navegadores são: Mozilla Firefox, Google Chrome, Microsoft Explorer e o Netscape Navigator (CONFORTO; SANTAROSA, 2002).

Os servidores possuem estrutura mais complexa no que tange o processo e tratamento da requisição. O serviço oferecido por esses servidores é baseado em um modelo de melhor esforço e uma das consequências de sua aplicação é a forma de como as requisições são tratadas, ou seja, de maneira uniforme, sem nenhum tipo de diferenciação ou priorização. Assim, a forma de tratamento de requisições é feita de forma que a primeira requisição a chegar será a primeira a ser atendida pelo servidor WEB (TEIXEIRA; SANTANA; SANTANA, 2005). De acordo com (SOUZA et al., 2006), os clientes enviam requisições que solicitam ou remetem informações ao servidor, que deve atendê-las em tempo útil.

Visando o uso de recursos computacionais de baixo custo, e implementação de recursos de forma compartilhada, essa arquitetura traz benefícios por ser barata em termos monetários e por existir diversas soluções de código aberto disponibilizadas à comunidade,

sendo uma dessas ferramentas o *Apache Tomcat*, amplamente utilizado para desenvolvimento de aplicações na linguagem *Java*, de forma mais específica, um *container* de *servlets*, resumidos como pequenos processos que são executados no lado do servidor a partir de uma requisição de um usuário. (KOZOVITS; FEIJÓ, 2003).

A arquitetura Cliente-Servidor é usada em sistemas industriais, onde existe uma necessidade de comunicação de vários clientes de maneira concomitante, com acesso a banco de dados ou serviços disponibilizados pelo servidor com o intuito de atender determinado objetivo organizacional. A comunicação Cliente-Servidor constitui uma maneira versátil de distribuir atualizações para um grande número de usuários. Basicamente, cada cliente envia sua mensagem de atualização apenas para o servidor, que terá então a responsabilidade de encaminhar a informação para os demais clientes, sendo dessa forma, utilizada também como arquitetura para jogos eletrônicos *on-line* (KOZOVITS; FEIJÓ, 2003).

2.4 Linguagens de programação

A programação de computadores é uma disciplina que se mostra relativamente importante e as metodologias de projeto, ferramentas de desenvolvimento de software e linguagens de programação ainda estão em seu processo de evolução, o que torna a área de desenvolvimento de sistemas um ambiente totalmente mutável e que exige o aprendizado contínuo (SEBESTA, 2009). O processo de aprendizado de uma nova linguagem de programação pode ser algo simples ou em pior hipótese, algo extremamente doloroso quando a estrutura da nova linguagem é influenciada por estruturas divergentes em outras linguagens de programação na qual o indivíduo detém.

A linguagem de programação é vista como uma ferramenta capaz de auxiliar no desenvolvimento de soluções computacionais (ALMEIDA et al., 2002). Desta forma, para se implementar um algoritmo em um computador, é preciso descrevê-lo de uma forma que o computador em questão entenda, ou seja, esteja apto a executá-lo. O instrumento de conversão de linguagem de máquina, a linguagem que os computadores entendem, para a linguagem conhecida pelos seres humanos é preciso que se tenha habilidades de alguma linguagem de programação, responsável por fazer essa conversão.

As linguagens de programação podem ter várias taxonomias, desde linguagens de total baixo nível como a linguagem de montagem *Assembly*, até linguagens de alto nível como PHP - *Hypertext Preprocessor*. Sendo a linguagem PHP utilizada para estruturação no lado do servidor neste trabalho.

As linguagens conhecidas como “não-estruturadas” são mais sofisticadas que as linguagens estruturadas, por serem mais flexíveis, por terem seus comandos sem vínculo direto com o processador, tornando seu uso possível em diferentes plataformas (GUDWIN,

1997).

Nesse sentido, as linguagens “estruturadas” ou procedurais, de acordo com (GUDWIN, 1997), se diferenciam das linguagens não estruturadas pela existência de algoritmos que determinam uma sequência de chamadas de procedimentos, que constituem o programa, ou seja, existe a necessidade de escrita de forma linear de todo o programa. Linguagens procedurais mais comuns são o C, o Pascal e o Fortran, conhecidas e aplicadas com peso na área de sistemas operacionais e sistemas bancários.

De acordo com (SOUZA, 2009) é mais comum iniciar-se na programação de computadores com linguagens estruturadas, pois exigem conhecimento básico dos elementos que são necessários para processar um programa simples de soma de dois números, e podem ser colocados em um contexto de relacionamentos lógicos, operadores de atribuição, entrada e saída de dados, sequência, decisão e iteração, com uma abordagem tradicional levada para o ensino inicial de programação de sistemas computacionais.

A Orientação a Objetos é vantajosa pelo fato de existir maior abstração de situações do dia a dia de maneira mais fidedigna. Essa abstração é feita por representações do mundo real, pensada a partir de objetos (MELO, 2004).

Nas linguagens orientadas a objetos, o foco principal é na lógica de relacionamento entre suas diversas classes de objetos, onde, objetos são instâncias dessas classes, que determinam qual informação um objeto contém ou possa vir a conter e como deve ser a forma de manipulação do mesmo. Nesse tipo de linguagem de programação é importante ressaltar que existem elementos que ajudam no desenvolvimentos de sistemas orientados a objetos, como o RUP (*Rational Unified Process*), que auxilia no gerenciamento de escopo do sistema assim como as suas diversas fases de desenvolvimento (MARTINS, 2010).

2.5 Banco de dados

A descrição simplista de banco dados torna-se necessária pelo fato do termo ser usado constantemente no decorrer do trabalho. Nesse ponto, o conceito é auto elucidativo e segundo (ELMASRI; NAVATHE, 2011), a principal ideia que se deve retirar quando se fala em banco de dados é que esse se trata de uma coleção lógica e coerente de dados com um significado inerente.

Os bancos de dados e os sistemas de bancos de dados relacionais ou não, se tornaram componentes essenciais no cotidiano da sociedade moderna (ELMASRI; NAVATHE, 2011). Nesse sentido, é impossível retirar esse relacionamento entre sistemas de banco de dados e sistemas desenvolvidos para gerenciamento de banco de dados, tachados também como aplicações tradicionais de banco de dados. Os dados, elemento indissociável para esses sistemas, são fatos que podem ser gravados e que possuem um significado implícito,

ou seja, os elementos são indissociáveis por terem relações de forma implícita e valor tangível se agregado ao mundo real, onde a relação desses dados resulta na informação, um dos maiores responsáveis pelas rotinas diárias de todos atualmente.

O conceito de banco de dados citado pode ser considerado genérico, uma vez que o mesmo pode ser usado para diferentes finalidades, mas sempre levando em consideração o relacionamento entre seus elementos básicos, ou seja, relacionamento entre os dados, com o intuito de gerar valor agregado à informação (ELMASRI; NAVATHE, 2011). Com efeito, uma agenda telefônica pode ser considerada como um banco de dados uma vez que possui valor agregado às informações que estão contidas na mesma.

Quando o conceito de banco de dados é colocado no limiar da computação, adentra sobre o conceito de Sistemas Gerenciadores de Banco de Dados - SGBD, que de acordo com (ELMASRI; NAVATHE, 2011) é uma espécie de coleção de programas que permite aos usuários criar e gerenciar um banco de dados, onde a construção desse se dá pelo armazenamento desses dados em mídias apropriadas controladas pelo SGBD.

Um banco de dados relacional, é construído baseando-se em elementos indissociáveis, como tabelas, que por meio dessas também chamadas de entidades, é possível a manipulação dos campos presentes nessa tabela, através do conceito de entidade-relacionamento, que se trata do relacionamento entre as entidades presentes em um banco de dados. Tais relacionamentos são feitos a partir de chaves primárias ligadas a chaves estrangeiras, podendo acontecer em diversas outras formas, tais como relacionamento de “um para um”, “muitos para muitos” ou “um para muitos”, não restringindo o relacionamento a esses exemplos.

Dessa forma, a presença de banco de dados no cotidiano é simples onde uma lista telefônica pode adentrar nesse conceito, que de forma geral pode ser dito como o armazenamento de dados relacionados, ou a representação de alguns aspectos do mundo real, sendo chamado também, de minimundo ou universo de discurso - UoD, sendo essas mudanças nesse minimundo refletidas em um banco de dados (ELMASRI; NAVATHE, 2011).

2.6 Interface de programação de aplicativos

O termo Interface de Programação de Aplicativos - API, consiste em um conjunto de padrões e rotinas elaboradas com base em uma estrutura de aplicação já definida, servindo para a utilização das funcionalidades desse software, encapsulando sua implementação, e provendo seus serviços internos de maneira simples e transparente para outros sistemas externos que requisitarem, necessitando muitas das vezes serem credenciados pela API para a utilização de seus recursos, através de linguagens de programação conhecidas por ambas as partes. Nesse sentido, vale destacar que essa forma de integração

depende de linguagem de programação e, muitas vezes fazem referência somente aos dados que são alocados no banco de dados desse software da API em questão. Por exemplo, um sistema WEB escrito na linguagem de programação *Java* não poderá prover recursos com premissas destinadas a linguagem de programação PHP, ou seja, o recurso deve ser disponibilizado e assim deve existir congruência na comunicação pois linguagens diferentes são incapazes de se comunicar de forma efetiva.

Quando se tem como objetivo a padronização das informações a serem repassadas por uma Interface de Programação de Aplicativos, pode-se fazer uso da forma textual JSON, que tem como conceito a representação de dados estruturados em uma coleção de pares no formato de chave/valor (CROCKFORD, 2006). Sua utilização deverá sempre ser descrita dessa forma, ou seja, a utilização de uma chave para a identificação do valor atribuído a essa chave. A chave tem sua representação no modo texto e os valores podem ser definido nos mais variados formatos, como booleano, nulo, objeto, ou até mesmo uma sequência ordenada de valores. O JSON é amplamente utilizado no trâmite de informações, pois tem sua estrutura independente de linguagem de programação, também apresenta fácil criação, manipulação e análise (SPORNY et al., 2014). Entretanto, o JSON também apresenta limitações, uma vez que a sua estrutura está restrita ao uso de chave e de seu respectivo valor, e torna-se inviável ao ser aplicado para o contexto hipermídia, e também apresenta falhas ao tentar-se integrar dados de diferentes fontes, pois pode existir conflitos e ambiguidades nos pares de chave/valor (SALVADORI et al., 2015).

2.7 Linguagem de modelagem unificada

A UML trata-se de uma linguagem gráfica para visualização, especificação e documentação de artefatos de sistemas. Serve como forma padronizada e planejamento da arquitetura de projetos de sistemas, incluindo aspectos conceituais como processos de negócios e funções do sistema, e de itens concretos como as classes escritas em determinada linguagem de programação, mas não estando restrita a ela, ou seja, serve de base para desenvolvimento da arquitetura do sistema independentemente da linguagem de programação a ser utilizada no desenvolvimento, assim como esquematiza o banco de dados e componentes a serem reutilizados no *software* (BOOCH; RUMBAUGH; JACOBSON, 2006).

Neste trabalho, em questão à UML, possui foco com o que diz respeito aos diagramas de casos de uso e de atividades. As atividades são as especificações dadas a um comportamento parametrizado que é expresso como um fluxo de execução a partir de elementos sequenciados, de forma que o menor elemento possível corresponde a uma ação individual. Define-se caso de uso como sendo a representação das funcionalidades providas por um sistema (MELO, 2004).

A formação dos diagramas de atividades representam a execução de ações e os fluxos que são disparados pela conclusão de outras ações ou atividades. A formação dos diagramas de casos de uso apresenta os atores e seus relacionamentos que expressam a funcionalidade de um sistema. Diagramas de classes representam a essencial das classes objetos utilizadas em sistemas. E diagramas de sequencia indicam o fluxo de atividades executadas a nível de sistema (MELO, 2004). Diagramas de classes 5 e de sequencia 10 poderão ser encontrados nos apêndices deste trabalho e estão descritos de forma simples e de fácil leitura. O motivo destes dois últimos diagramas não estarem presentes no texto é que tais diagramas são abstrações simples de outros citados no texto, como por exemplo o diagrama de atividades de inserção e o diagrama de sequência para este mesmo processo. Estes processos podem ser visualizados nas Figuras 9 e 10, respectivamente.

Os diagramas de casos de uso são mais simples que os diagramas de atividades e estão interessados em exemplificar o uso do sistema pelos diversos usuários que irão operar o mesmo desde a sua implantação até a sua finalização. A partir disso, os diagramas de atividades serão exemplos das ocorrências a partir de estímulos do operador do sistema para realização de determinada tarefa, objetivando um resultado previsto pelo sistema a partir dos modelos relacionados aos casos de uso.

A perspectiva mais utilizada para a construção de *softwares* na atualidade é a orientada a objetos, no qual de acordo com (BOOCH; RUMBAUGH; JACOBSON, 2006), um objeto é alguma coisa geralmente estruturada a partir do vocabulário do espaço do problema ou do espaço da solução, e que uma classe é essa descrição de conjuntos de objetos comuns. A utilização dessa perspectiva é atualmente vista como a principal solução para problemas ou projetos de soluções que são, de forma estrita, presentes no mundo real, sendo assim convertida para a visão de sistemas orientados a objetos.

2.8 Metodologia de desenvolvimento SCRUM

O SCRUM trata-se de uma metodologia ágil para gestão e planejamento de projetos de sistemas computacionais. Nesta metodologia, os projetos são divididos em ciclos semanais, quinzenais ou mensais que são chamados de *Sprints*.

Esforçando-se para liberar o processo de quaisquer barreiras, o seu principal objetivo é conseguir uma avaliação correta do ambiente em evolução, adaptando-se constantemente ao “caos” de interesses e necessidades, indicado e utilizado para o desenvolvimento de softwares em ambientes complexos, onde os requisitos mudam com certa frequência e de forma excessiva, sendo o caminho utilizado para aumentar produtividade nesses tipos de sistemas (BISSI, 2007).

Possui sua estruturação a partir de vocabulários como *Backlog*, *Sprint*, *Sprint Backlog*, *Dayling* SCRUM. Resumidamente, o *Backlog* é a lista de todas as funcionalidades

a serem desenvolvidas durante o projeto completo, sendo bem definido e detalhado no início do trabalho, deve ser listado e ordenado por prioridade de execução. O *Sprint* é o período não superior a 30 dias, onde o projeto e suas funcionalidades são desenvolvidas.

O *Sprint Backlog* é descrito como o trabalho a ser desenvolvido num *Sprint* de modo a criar um produto a apresentar ao cliente. Outro termo bastante utilizado nesta metodologia é a de *Scrum Master* e *Scrum Team*. O *Scrum Master* é tratado como o indivíduo responsável pela gestão do projeto e também de liderar os *Scrum Meetings*, que são normalmente os papéis dos engenheiros de *software* responsáveis pela modelagem conceitual do sistema a ser desenvolvido. O *Scrum Team* é traduzido como a equipe de desenvolvimento responsável por acatar os requisitos em cada *Sprint* e construir a funcionalidade no sistema.

O *Dayling SCRUM* é descrito como uma reunião diária para que possa ser retiradas as dúvidas presentes no desenvolvimento, o que geralmente é chamado de reunião em pé, não excedendo 15 minutos de conversa.

Desta forma, as fases do SCRUM podem ser divididas em fase de planejamento, desenvolvimento e encerramento. A fase de planejamento está restrita aos processos de iniciação e descrição das funcionalidades requeridas no sistema, baseando-se no conhecimento do sistema como um todo. O desenvolvimento é a forma de implementação da funcionalidade ou parte dela no sistema, respeitando os acordos e tempo exigidos. Por fim, a fase de encerramento está ligada à entrega do produto e possui elementos de trabalho como os testes de sistema, avaliação de documentação feita e treinamento de usuário para manuseio do *software* (BISSI, 2007).

2.9 Zend Framework e Doctrine

Um *framework* pode ser entendido com o conjunto de componentes interligados que gerenciam a aplicação, o que possibilita que o desenvolvedor se envolva somente aos aspectos relativos às regras de negócio do sistema. O padrão de projeto Model, View e Controller - MVC, é um dos mais utilizados atualmente para desenvolvimento de sistemas orientados à objetos (BONOTO; JUNIOR,). A forma de visão deste padrão é feito da divisão entre processamento da lógica de negócio do sistema e do que é visto e estimulado pelo usuário final deste mesmo sistema.

A utilização de *frameworks* facilita a organização de uma aplicação, realiza injeção de dependência se necessário, valida campos com lógica de negócio e ajuda a tornar a estrutura do projeto organizada na maioria das vezes dentro de uma modelo MVC (BONOTO; JUNIOR,). MVC faz parte de uma boa prática de projetos de aplicações Web modernas.

O Zend Framework é um *framework* para a linguagem PHP desenvolvido pela Zend ([FRAMEWORK, 2017](#)) e pelos responsáveis por esta linguagem. Possui sua estruturação baseada no MVC, é modular, escalável, integrado com diversos *WEB Services* e com suporte a internacionalização a partir de tradutores automáticos. É estruturado em blocos de construção que podem ser usados peça por peça com outras aplicações e *frameworks*, ele também é extensível e fácil para adaptar a estrutura as necessidades, possui várias ferramentas de codificação de criptografia e segurança que você precisa, suporta múltiplos idiomas, e possui uma base de usuários muito ativa para obtenção de ajuda ([BONOTO; JUNIOR,](#)).

Outro *framework* utilizado em conjunto com o Zend Framework é o Doctrine ([DOCTRINE, 2017](#)). Doctrine é um Mapeador Objeto-Relacional - ORM para PHP que fornece persistência transparente para objetos do PHP. Ela fica em cima de uma poderosa camada de abstração de banco de dados. Object-Relational Mappers tem a principal tarefa de fazer a conversão entre objetos (PHP) em linhas de dados relacional ([ANDRADE, 2011](#)). Desta forma, a utilização em conjunto destes dois *frameworks* propicia à construção de sistemas orientados a objetos em um tempo curto utilizando-se de metodologias de desenvolvimento ágil.

2.10 Código QR

O código QR trata-se de um símbolo bidimensional cuja implementação se deu inicialmente em 1994 pela Denso, uma das principais empresas do grupo Toyota, e pela ISO/IEC 18004 em junho de 2000. Com isso, foi visto como a solução para o controle de peças automotivas na produção de grandes montadoras ([SOON, 2008](#)). Estes códigos possuem capacidade de armazenar objetos de diferentes naturezas, como URLs, textos, posições geográficas, *etc.*

Dentre as suas características podem ser citadas as seguintes:

- Possui vantagem superior aos códigos de barras lineares, pois a sua capacidade de armazenamento textual é maior.
- Por possuir patente de domínio público, pode ser utilizado por qualquer pessoa.
- Padrão dos dados inseridos não é pré-requisito para utilização, podem ser definidos de maneira genérica no padrão textual.
- A maioria dos sistemas operacionais dos celulares atuais possuem aplicativos disponíveis e gratuitos que executam a leitura rápida e simples de códigos QR.

Um código QR que armazena uma URL pode ser visualizado na Figura 2.



Figura 2 – Exemplo de código QR que armazena a URL: [<https://dsnnunes.com.br/SID/>](https://dsnnunes.com.br/SID/)

2.11 Trabalhos relacionados

As soluções que utilizam a sinalização digital encontradas na literatura atuante estão restritas a processos compostos de elementos que atrelam etapas numerosas, diversificadas e difíceis de gerenciar de forma simples. Dentre estas soluções estão a *Raspberry Slideshow* e *Raspberry Digital Signage* (EMOTIONS, 2017). De maneira resumida, a primeira trata-se de uma distribuição GNU/Linux própria para dispositivos *Raspberry Pi* em que o usuário insere um *pendrive* com conteúdo (principalmente imagens e vídeos), e o sistema trata de executar um *slideshow* apresentando o conteúdo sequencialmente e sem distrações. Alternativamente, o sistema pode ler um arquivo de texto, denominado `remote-urls.txt`, indicando os endereços WEB do conteúdo a ser adquirido e apresentado. Já a solução nomeada como *Raspberry Digital Signage* baseia-se em fornecer um navegador WEB livre de distrações que exibe recursos de um determinado endereço eletrônico de publicidade, necessitando assim de implementação a nível de serviço que alimente o conteúdo a ser publicado no endereço eletrônico.

As soluções supracitadas apresentam problemas no que tange o processo de desenvolvimento arquitetural das soluções. Na primeira solução a interação entre as funcionalidades do sistema são definidas a partir de um único módulo, desempenhando as funções que seriam restritivas de um cliente e também de administrador, impossibilitando, por exemplo, que o objeto de divulgação possa ser visualizado por outras plataformas como *smartphones*. Outro problema é que as divulgações devem ser previamente formatadas pelo seu operador para então serem apresentadas ao interessado, dificultando a usabilidade e o processo de manutenibilidade. A segunda solução trata-se de um apresentador de página WEB em tela cheia que, desta forma, necessita de implementação de outra aplicação que faça toda a gerência e publicação das divulgações o que novamente acarreta em falta de usabilidade e autonomia da solução. Além disso, nenhuma das soluções pos-

suem forma de integração com redes sociais, dificultando assim, o processo de interação em tempo real com a informação.

Outra solução encontrada na literatura corresponde ao *software* Xibo, que trata-se de um sistema baseado em arquitetura cliente-servidor completo e flexível de sinalização digital que permite diversas customizações, na qual cada divulgação tem opção de estruturação das suas informações. Também suporta diferentes tipos de mídias como vídeos, imagens, texto, relógios, dados tabulares, *etc.* Possui gerenciador de conteúdo incluso (CMS) e possibilita que o servidor CMS e o módulo cliente estejam em dispositivos separados (XIBO, 2017). No entanto, não possui integração com as redes sociais além de ser necessário que a cada divulgação a ser inserida tenha de ser estruturada como será a sua forma de apresentação.

A solução Info-beamer oferece aos seus utilizadores vídeos totalmente animáveis, possibilitando customização enquanto estão em execução. Trata-se de uma ferramenta robusta e poderosa, porém necessita de operadores capacitados e com conhecimento de programação, pois toda a sua manipulação é feita a nível de código em linguagem própria, necessitando assim, que o operador possua conhecimentos sobre linguagens de *script* para que seus efeitos sejam criados de forma atrativa. Possui distribuição própria para dispositivos Raspberry Pi. Consegue interoperar com outras funcionalidades WEB, desde que auxiliada através de outras aplicações que geram consultas a partir de arquivos JSON (INFO-BEAMER, 2017), incluindo a sua integração com as redes sociais, porém não presente nativamente no sistema.

A Tabela 1 introduz um comparativo com algumas funcionalidades consideradas importantes para sistemas que trabalham com a implantação de sinalização digital, na qual seus elementos comparativos são descritos a seguir:

- I) Comprometimento com o propósito: o sistema em questão possibilita a veiculação de informações através de mecanismos de sinalização digital?
- II) Criação simples de divulgações: o operador possui facilidade de incluir novas divulgações com aspecto atrativo?
- III) Portabilidade: é possível visualizar a divulgação em diferentes dispositivos?
- IV) Integração com redes sociais: o sistema integra-se nativamente de alguma forma com redes sociais?
- V) O conteúdo é gerido em um dispositivo diferente em que é apresentado, fortalecendo a descentralização e manutenção?

¹ Esta solução não permite a inclusão de divulgações, ela simplesmente apresenta as divulgações presentes em uma URL.

² Esta solução não apresenta integração nativa com redes sociais. No entanto, aplicações externas podem

Tabela 1 – Tabela comparativa de elementos das soluções.

Quesito/Solução	Raspberry Slideshow	Raspberry Digital Signage	Xibo	Info-beamer
I	Sim	Sim	Sim	Sim
II	Sim	NSA ¹	Sim	Não
III	Não	Sim	Sim	Não
IV	Não	Não	Não	Não ²
V	Não	Sim	Sim	Não

criar arquivos JSON que possibilitem essa integração.

3 SID - Sistema inteligente de divulgação

Conforme o conceito amplo de sinalização observado por (PIRES, 1999), o processo de sinalização pode ser aplicado como o sistema alarmante de uma rede de telecomunicações, transformando redes inertes, sem vida, em máquinas dinâmicas, poderosas capazes de proporcionarem diferentes serviços aos utilizadores.

O Sistema Inteligente de Divulgação de Informações do IFG Formosa – SID, tem como principal objetivo fornecer informações através de sinalização em sua forma digital do IFG - Câmpus Formosa, servindo como facilitador e gerenciador dessas informações (BISPO; NUNES, 2015).

Nesta seção, o sistema SID será detalhado, inicialmente partindo de maneira macroscópica, adentrando nos conceitos que levaram a sua construção, repassando uma visão geral, e posteriormente de maneira microscópica, objetivando a explicação de conceitos intrínsecos à sua construção, como diagramação de casos de uso do sistema, especificação de alguns atributos do sistema e observação das atividades elencadas em seus processos.

3.1 Visão Geral

O SID encontra-se estruturado para ser utilizado no ambiente WEB e tem como objetivo agilizar e efetivar o processo de assimilação da informação no Câmpus de maneira atrativa e dinâmica. O sistema está atendendo ao objetivo principal, sendo esse a divulgação de informações através de uma plataforma WEB dividida em dois módulos de sistema, sendo um desses módulos o administrador, que tem como função o gerenciamento de divulgações no SID e o módulo cliente, que tem como objetivo a propagação das divulgações adicionadas pelo módulo administrador através de monitores espalhados pelo ambiente do Câmpus.

Essa divisão de módulos baseada na arquitetura cliente-servidor, mostrou-se necessária por haver diferenças das funções atribuídas a cada um com relação às divulgações.

O administrador é responsável por gerir as informações do sistema, inserindo, alterando, e retirando as divulgações, ou seja, faz o gerenciamento das informações a serem repassadas ao módulo cliente que, por sua vez, de forma simples, tem o trabalho de apresentar as divulgações de maneira atrativa e dinâmica.

O Sistema utiliza o método de autenticação padrão, fornecida pela Graph API, que baseia-se na verificação das credenciais do perfil do SID no Facebook. Este método apresenta-se seguro pois está em conformidade com sistemas criptográficos seguros.

Através da Graph API também é possível realizar publicações em um determinado perfil. Dessa forma, qualquer divulgação que for inserida no sistema, é repassada como forma de publicação no perfil do SID no Facebook, automatizando o processo e não necessitando que haja a inserção da mesma informação em veículos de comunicação distintos. Em suma, todos os dados inseridos no SID são estruturados para serem publicados em seu perfil no Facebook de forma transparente.

O SID foi desenvolvido em linguagens utilizadas no ambiente WEB, escrito na linguagem PHP, utiliza-se da orientação a objetos em sua arquitetura no que tange o lado referente ao servidor, e linguagem de *script* como JavaScript (FLANAGAN, 2006) com o auxílio da biblioteca JQuery (SILVA, 2010) para estruturar as informações referentes ao lado do cliente. Utiliza-se o Sistema Gerenciador de Banco de Dados PostgreSQL (POSTGRESQL, 2017) para que as informações utilizadas em seu contexto seja de caráter persistente. Baseado em desenvolvimento ágil, utilizou-se da metodologia SCRUM com *sprints* semanais para definição das funcionalidades a serem construídas (COHN, 2000). Fez uso dos *frameworks* de desenvolvimento Zend Framework (FRAMEWORK, 2017) em sua segunda versão, que tem a finalidade de introduzir a estruturação de código em modelo, visão e controle e implementação da metodologia SCRUM para agilizar o processo de desenvolvimento. e o Doctrine (DOCTRINE, 2017) em sua terceira versão, utilizado para efetivar a comunicação a partir do banco de dados relacional e o modelo orientado a objetos utilizado no SID, proporcionando facilidade nas transações de banco de dados.

3.1.1 Módulo administrador

O SID foi desenvolvido baseado na arquitetura Cliente-Servidor, e, de forma sucinta, viabiliza a adição de divulgações em seu módulo de administrador para que este consiga acessá-las e repassá-las ao módulo cliente.

Todos os processos de gerência de requisitos funcionais, não funcionais e das regras de negócio do sistema estão acopladas a este módulo, que tem por base a gerencia de um sistema de inserção, alteração, exclusão e listagem de divulgações.

A divisão em dois módulos delimita os papéis dos usuários e mantenedores do sistema. Esta decisão motivou-se pelo fato de que o gerenciamento possa ser exclusivo ao módulo administrador, flexibilizando a escolha do módulo cliente e deixando a maior parte do processamento destinada ao servidor. Assim, o projeto do módulo cliente é simplificado, visto que ele só precisa lidar com processos relativos à estruturação da apresentação e divulgação desta para o usuário final. Com esta decisão de projeto, o módulo cliente pode ser implantado em máquinas cujo poder computacional é limitado.

Quanto a autenticação do administrador do SID, optou-se por escolher o método de autenticação prestado pelo Facebook. Este método é baseado em sistemas criptográficos

seguros, viabilizando uma série de outras características como reaproveitamento de código. O Capítulo 4 tratará de explorar a questão da integração com o Facebook por meio da Graph API.

Por divulgação, o sistema SID, entende como um conjunto de elementos que tem por finalidade a abstração de determinada informação, sendo esses elementos:

1. **Legenda:** É um campo de texto destinado a informação a ser repassada de forma sucinta ao usuário final.
2. **Link:** Este será apresentado no módulo cliente em forma de código QR, facilitando o acesso aos detalhes da informação que estarão compostos no *link* definido.
3. **Prioridade:** Define o tempo em que a divulgação aparecerá no módulo cliente, sendo definida como “Alta”, “Mediana” e “Baixa”.
4. **Data de término:** Esse elemento tem como função a organização de divulgações que poderão permanecer obsoletas no sistema, com esse elemento pode ser definido quais serão as divulgações que em determinado período de tempo irão aparecer para o usuário final.
5. **Imagem:** Resume-se em uma representação visual que abrangerá a divulgação a ser mostrada no módulo cliente, ou seja, tem como intuito, chamar a atenção do usuário final e colocar o contexto da divulgação de forma apresentável ao mesmo.

A Figura 3, consiste em representar como os elementos supracitados são inseridos a partir da visão do operador do sistema no módulo administrador. O diagrama de sequência para esta funcionalidade encontra-se na Figura 10.

Figura 3 – Página de inserção no módulo administrador.

Ao ser efetivada uma inserção de uma nova divulgação, o administrador deverá se atentar para os elementos a serem repassados para tratamento no sistema, sendo que

elementos vazios não serão permitidos, retornando, quando emitidos, uma mensagem informando o ocorrido ao operador do sistema. Essa ocorrência motiva-se pelo fato de que em um sistema que tem como objeto a assimilação de informações repassadas aos seus usuários, campos vazios não serão de valor para os mesmos, então por esse motivo, informações sucintas e simples devem ser repassadas para que o objeto seja entendível para os seus interessados.

A introdução dos elementos é efetuada através de preenchimento de campos de texto.

O elemento legenda é restrito a 80 caracteres com o intuito de possibilitar que o módulo cliente consiga apresentá-la em tempo hábil através de um movimento linear suave com sentido da esquerda para direita. Este movimento objetiva a criação de um aspecto de dinamicidade enquanto o interessado na informação se atenta à outros detalhes desta.

O campo *link* deverá ser preenchido de maneira análoga pelo administrador, o qual deve inserir uma URL. O módulo administrador do SID então pode converter a URL para uma imagem representando um código QR a ser apresentada no módulo cliente. A utilização de códigos QR, os quais podem ser interpretados por aplicativos de *smartphones*, permite ao sistema adicionar informações relevantes de maneira sucinta. Desta forma, um transeunte que se interessar por uma divulgação, pode obter mais detalhes ao consultar o código QR associado a mesma. A utilização destes códigos possibilita a interação do usuário com o SID.

O campo prioridade fornece ao sistema um parâmetro para o módulo administrador da proporção de vezes que uma determinada divulgação deve aparecer em relação às demais. Permite que certas divulgações obtenham destaque por tratarem-se de divulgações mais importantes. Contudo, ainda não está implementada no módulo administrador.

A data de término tem como objetivo restringir divulgações que já tiveram seu papel de informar sobre determinado assunto. Desta forma, divulgações obsoletas são excluídas das informações a serem passadas para o módulo cliente. É importante ressaltar que as divulgações expiradas não são automaticamente deletadas do banco de dados do sistema, necessitando que haja interação explícita do administrador para permanente exclusão da divulgação em questão.

Por fim, no campo imagem poderá ser informado o caminho de uma imagem presente na máquina onde encontra-se na máquina do módulo administrador.

A usabilidade do módulo administrador é factível por se tratar de um sistema com arquitetura simples, entretanto detalhes neste processo devem ser observados, como por exemplo o formato da imagem que deve ser do tipo “.gif”, “.jpg”, “.png” ou “.jpeg”.

3.1.2 Módulo cliente

A estruturação visual das divulgações no módulo cliente pode ser visualizada na Figura 4. A imagem serve como principal elemento para a assimilação da informação e por isso mantém-se em toda a proporção da tela utilizada, vídeos poderão ser acoplados em versões futuras, no entanto não sendo possível nesta versão. A legenda é disposta em movimento linear da direita para a esquerda, possibilitando sua leitura de forma dinâmica. Ao lado esquerdo da legenda encontra-se o campo *link*, no qual está convertido para apresentação a partir do código QR, facilitando o acesso à mais detalhes sobre a informação a partir da leitura por aplicativos em *smartphones*. No canto superior direito encontra-se informações relativas à hora e data atual.

A forma de interpretar tais elementos é simples, a legenda é inserida a partir do campo legenda no módulo administrador e assim repassada de forma linear com transição da direita para a esquerda. Tem como principal objetivo colocar o interessado final próximo ao contexto da informação, sendo para isso, que seja necessária coesão entre este elemento e os demais.

A imagem é preenchida pelo elemento repassado no parâmetro imagem no módulo administrador visto anteriormente e está relacionada à uma requisição HTTP no servidor feita de forma transparente pelo próprio navegador WEB utilizado.

O *link* é repassado ao módulo cliente em forma de código QR, que tem como função a transposição do texto destinado ao redirecionamento para detalhes adicionais a partir de uma imagem bidimensional, sendo assim fácil de ser novamente convertida em texto a partir de um leitor de código QR em *smartphones* dos interessados. Esse processo de conversão é executado pelo servidor no módulo administrador, que recebe como parâmetro o texto a ser convertido e repassa ao cliente uma imagem de código QR.

O campo data de termino que é inserido na divulgação no módulo administrador tem como objetivo selecionar apenas as divulgações que não possuem suas datas expiradas, ou seja, uma divulgação que já teve sua função exaurida não será repassada ao módulo cliente, porém não sendo eliminada do banco de dados, necessitando que haja uma iteração do operador do sistema com o módulo administrador para que tal divulgação seja deletada.

O módulo cliente efetua requisições ao servidor e através de *scripts* em Javascript com suporte da biblioteca JQuery formata de maneira atrativa, dinâmica e interativa de acordo com os elementos constituintes da divulgação inseridos pelo operador no módulo administrador.

Todas as imagens são apresentadas a partir de requisições HTTP posteriores ao servidor, e por esse fato, funções públicas, ou seja, desvinculadas dos módulos do sistema, foram definidas, pois sem estas funções existiria a necessidade de autenticação para se obter tais imagens, dificultando as chamadas pelo módulo cliente.

Observou-se como necessário a utilização de *arrays* em JavaScript para inserção dos dados requisitados do servidor. O cliente ao receber esses dados em formato JSON e a partir dos valores atribui um numero inteiro sequencial como identificador da divulgação para cada valor da chave repassada no JSON respectivamente. Assim, não necessitando de que chamadas ao servidor sejam feitas a todo momento, e desta forma diminuindo a frequência de processamento do módulo cliente.

O tempo que cada divulgação recebe é fixo, ou seja, o tempo é ajustado de modo que uma divulgação possa ser vista confortavelmente por um usuário. A princípio, foi definido um tempo padronizado de 60 segundos entre as divulgações. Este tempo foi escolhido empiricamente de modo que uma divulgação conseguisse ser visualizada confortavelmente pelos usuários do sistema e de modo que a legenda conseguisse ser transmitida mais de uma vez.

Na Figura 4 está o modelo atual da interface que é apresentado ao interessado final pelo módulo cliente. O diagrama de sequência para esta funcionalidade pode ser visto na Figura 8.



Figura 4 – Página carregada pelo módulo cliente.

3.2 Arquitetura do SID

A Linguagem de Modelagem Unificada - UML, se trata de uma linguagem gráfica para representação, visualização, especificação e documentação de artefatos de sistemas computacionais, porém não fica restrita a esse, uma vez que pode ser usada para qualquer tipo de representação, não se restringindo somente à *softwares*, e portanto irá servir de subsídio para representação de elementos presentes no sistema, aprofundando em sua arquitetura de maneira simplista (BOOCH; RUMBAUGH; JACOBSON, 2006).

Dessa forma, a UML será adotada para representação dos artefatos do SID. Se trata de uma linguagem de representação de alto nível, usada para apresentar elementos de funções de uso do sistema e sua modelagem baseada em diagramas de suas classes.

O sistema SID utiliza poucas entidades/classes em sua arquitetura. Por tratar-se de uma solução que se restringe a divulgação de informações para o interessado, seu escopo se mostrou com pequena extensão. Entretanto, o fato de haver poucas entidades não induz a um sistema comum, pois sua estrutura deve se atentar a eventos satélites, ou seja, que atuam para que essa divulgação chegue até o interessado final de forma rápida, simples e efetiva.

A separação em dois módulos facilitou a visão e destinação das atividades destinadas a cada entidade, como por exemplo a entidade “Divulgação”, está presente somente para o módulo administrador, restringindo o acesso para o cliente, no qual se alimenta dessas divulgações a partir de chamadas com resultados em formato JSON. A estruturação das classes dos sistema pode ser visto na Figura 5, e possibilita a visualização dos elementos que são manipulados no SID.

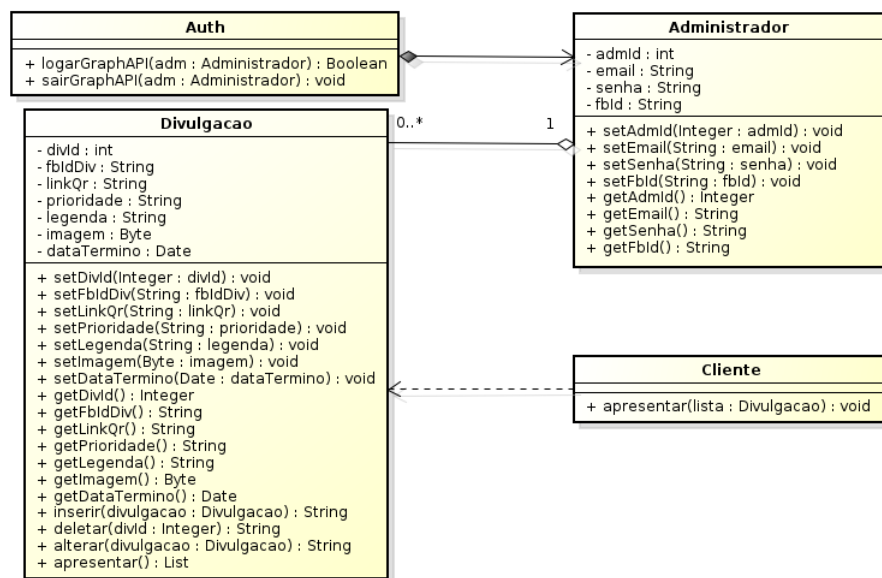


Figura 5 – Representação de classes do SID.

Na Figura 6, cuja função é representar a forma que se obteve acesso ao módulo administrador a partir do perfil do SID na rede social Facebook. O operador do sistema deverá ter propriedade desse perfil, pois somente desta forma que o operador poderá acessá-lo, e para isso será necessário estar logado no perfil do SID no momento em que desejar autenticar-se para operacionalizar o módulo administrador. Caso não esteja logado, será redirecionado para uma pagina *helper*, ou página ajudante responsável por levar o usuário à pagina de *login* desta rede social, disponibilizada pela Graph API possibilitando assim a autenticação para este módulo através do perfil do SID no Facebook.

O operador deverá logar no sistema e a sessão será criada, com isso, este mesmo operador deverá repassar os dados a serem inseridos a partir da página de inserção no módulo administrador mostrada na Figura 3 e, a partir disto, deverá ser checado a consistência dos dados repassados, devendo ser informado ao usuário qualquer fator não

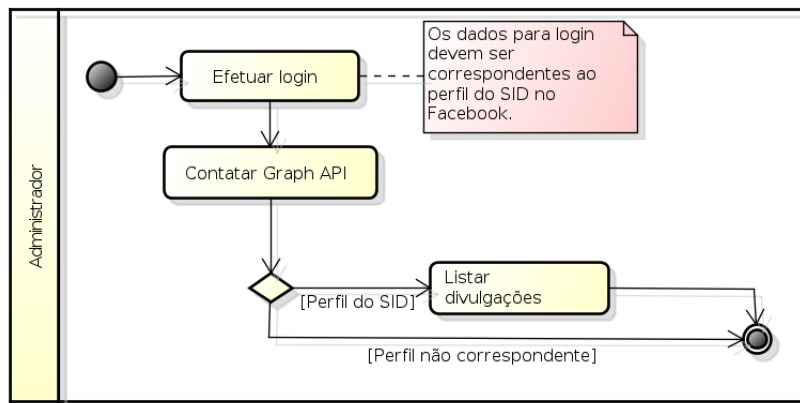


Figura 6 – Atividade de *login* no módulo administrador a partir do perfil do SID no Facebook.

permitido. Quando a divulgação for inserida com sucesso no banco de dados do SID deverão então, os mesmos dados, serem repassados à Graph API para inserção da divulgação também no perfil do sistema no Facebook, finalizando assim o processo de inserção.

O diagrama de caso do uso da Figura 7, tem como finalidade demonstrar os principais processos que estão presentes para o módulo administrador e o único processo destinado ao módulo cliente, para que a divisão dos módulos seja vista de forma simples.

Esta divisão das atividades destinadas ao cliente se manteve pelo fato de que tal módulo possa ser executado em computadores com processamento limitado, deixando assim o módulo administrador com a estruturação e o repasse dessas informações ao cliente de forma simples e rápida como uma lista de objetos em JSON.

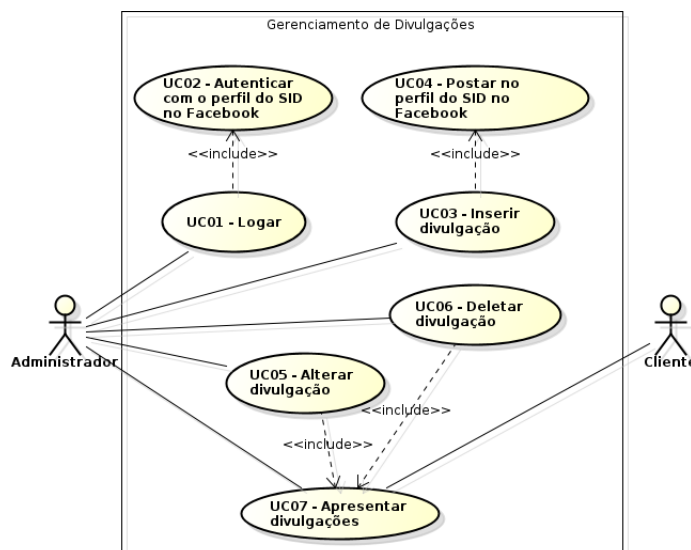


Figura 7 – Caso de uso das principais ações dos módulos administrador e cliente.

O processo de inserção de uma nova divulgação no SID poderá ser feita como mostrado na Figura 9 e a representação do diagrama de sequência para o processo de

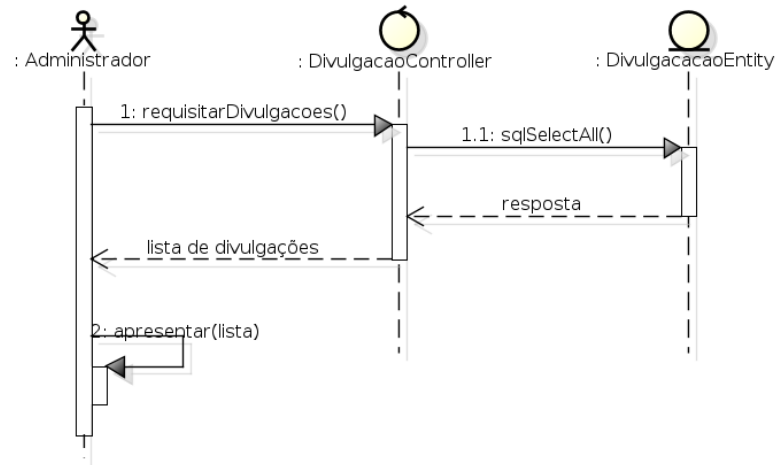


Figura 8 – Diagrama de sequência de requisição de divulgações.

inserção pode ser visualizado na Figura 10.

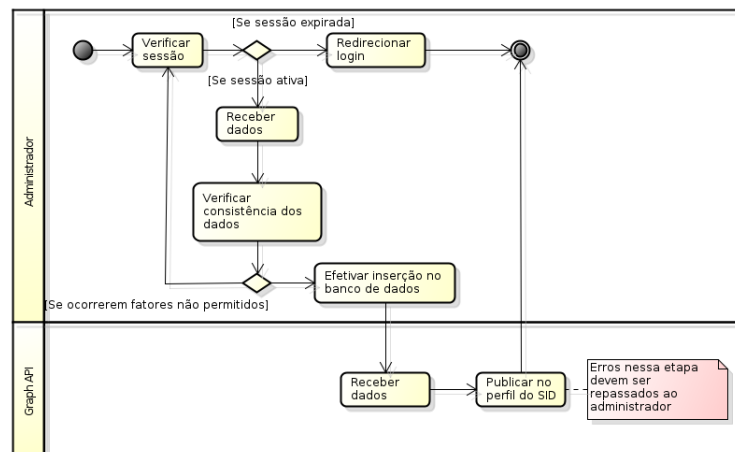


Figura 9 – Atividade de inserção de nova divulgação.

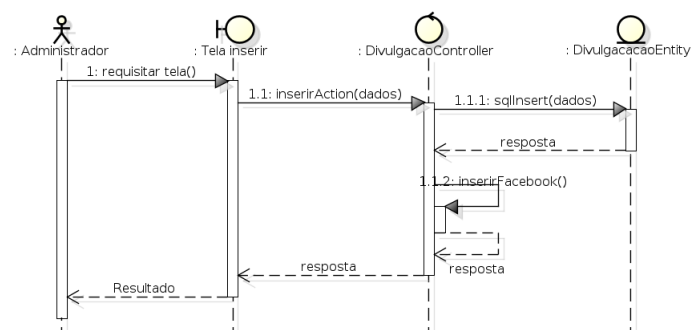


Figura 10 – Diagrama de sequência de inserção de divulgação.

Desta forma, todas as atividades atribuídas ao módulo administrador devem ser checadas. Ao logar no SID, uma sessão de usuário no servidor, responsável por gerenciar as atividades desempenhadas através de qualquer requisição, será criada com o intuito de

controlar as ações e evitar que a autenticidade das informações repassadas seja comprometida.

Banco de dados

A estruturação do banco de dados se manteve simples, contendo somente duas tabelas, uma mapeada como entidade “divulgacao” e outra como “adm”. A função da tabela divulgação é de armazenar todos os objetos relacionados às divulgações no SID. A tabela adm se manteve somente para que os dados do operador do sistema sejam mantidos diretamente no banco de dados, eliminando a ideia de autenticação em código ou classe.

O banco de dados do SID foi construído com o Sistema Gerenciador de Banco de Dados - SGBD PostgreSQL, que tem sua estruturação na modelagem de entidade e relacionamento a partir de um modelo relacional (MOMJIAN, 2001). As outras alternativas de bancos de dados a serem comparados com o PostgreSQL são, também, alternativas de *softwares* livres e de acordo com (MILANI, 2008) pode-se comparar esse SGBD com o MySQL ou FirebirdSQL e pode ser comparado com *softwares* gerenciadores de bancos de dados privados como SQL-Server e Oracle.

A Figura 11 demonstra o modelo entidade relacionamento do banco de dados do SID.

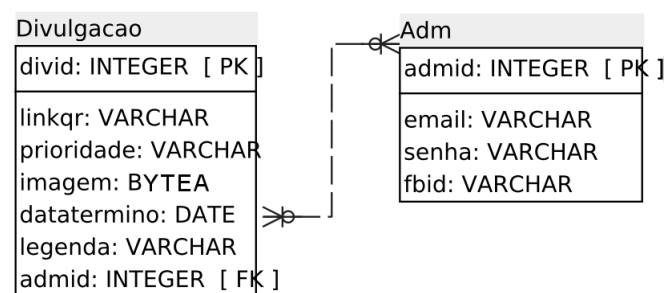


Figura 11 – Modelo de entidade e relacionamento do Banco de dados do SID.

Desta forma ao se escolher entre as diferentes alternativas disponíveis para implantação de um banco de dados é de suma importância saber quais serão os recursos alocados em sua estrutura. O SID, por tratar-se de um sistema com chamadas simples ao banco de dados, tem sua estruturação simples e intuitiva, com poucas tabelas e com relacionamento genérico de muitos para muitos entre suas duas tabelas.

Na tabela “divulgacao” serão guardados os objetos inseridos pelo operador do sistema através do módulo administrador, servindo como ponto principal na regra de negócio do SID. Tais regras de negócio se mostram simples, onde o princípio fundamental é a de execução de operações básicas relacionadas à inserção, alteração, deleção e apresentação.

Os tipos de dados guardados nessa tabela estão restritos aos inteiros, datas e textos. O tipo de dados “BYTEA” permite o armazenamento de cadeias binárias que, no caso do SID, servem para representar a imagem da divulgação através de uma cadeia binária ou sequência de octetos (ou *bytes*) (POSTGRESQL, 2017). Desta forma a imagem é convertida e guardada diretamente no banco de dados uma vez que a estruturação através de diretório de imagem se mostrou ineficiente no contexto do SID, necessitando apenas de funções específicas para conversão de imagem para o tipo “BYTEA” e vice versa (SILVA, 2006).

A tabela “adm” armazena os dados que estão relacionados ao operador do sistema, ou seja, o identificador do perfil do SID no Facebook. Nota-se que os campos “email” e “senha” estão, de certa forma, inativos na versão atual do sistema, uma vez que a autenticação é feita diretamente com o identificador do perfil que resume no campo “fbId”, como mostrado na Figura 11. Pode-se questionar a necessidade de existência destes campos, no entanto, optou-se por deixá-los presentes no banco de dados para que implementações posteriores sejam feitas sem drásticas alterações, e não comprometendo a estrutura do sistema. Com isto, a visão do SID quanto ao seu operador não restringe-se a uma única pessoa, podendo posteriormente, ao término deste trabalho, ser integrado funcionalidade que possam prover a possibilidade de vários outros operadores para o sistema. Os assuntos relacionados aos trabalhos futuros a partir do SID serão elencados no Capítulo 6.

3.2.1 Solução para implantação do SID

Por tratar-se de um sistema que impõe todo o processamento para o servidor, o módulo cliente poderá demandar em pouca capacidade de processamento devido à esta filosofia adotada em todo o sistema conforme disposto na Seção 3.1.1. A partir deste fato, haverá flexibilidade de implantação do módulo cliente em diversos dispositivos com processamento computacional reduzido.

A solução utilizada no contexto do Câmpus possui certa semelhança com a *Raspberry Digital Signage*, pois essa solução permite ser utilizada como módulo cliente do sistema SID, porém a arquitetura de construção destes sistemas diferem-se no que tange a diferença entre os papéis do cliente e do servidor, pois o SID implementa de forma efetiva tanto o papel do cliente e o do servidor sendo que a *Raspberry Digital Signage* possui somente implementação do cliente, impedindo a interação de servidor e cliente em um único sistema. O SID, por sua vez, implementa ambos, ou seja, cliente e servidor em uma arquitetura própria. Com isso, a solução utilizada para apresentar as divulgações para o usuário final neste trabalho se deu com a utilização do módulo cliente, responsável por estruturar os dados repassados pelo módulo administrador, em conjunto ao dispositivo *Raspberry Pi*, que a partir do módulo cliente auxilia no processo de transmissão das divulgações ao interessado final, mas que, por se tratar de um sistema com implementação

WEB o mesmo não está restrito à essa solução, necessitando somente de um navegador WEB para que as funções relativas ao módulo cliente sejam efetivadas. Sua arquitetura depende da utilização de ferramentas amplamente utilizadas no contexto WEB, podendo outras soluções serem pensadas e estruturadas a partir da necessidade de utilização do repasse das informações ao usuário final.

O dispositivo *Raspberry Pi* terá como função disseminar as informações repassadas pelo servidor local que se encontra na rede do Câmpus para a tela que esse dispositivo estiver acoplado. A escolha na implantação desta solução se deu pelo fato de se tratar de uma plataforma mais econômica nos fatores energético, financeiro, físico e de manutenção (LIMA et al., 2016).

Ele serve de instrumento que conterà as informações repassadas ao módulo cliente, na qual a principal funcionalidade é que o mesmo execute processos relativos à chamada dos elementos que estão localizados no servidor local e, de forma simples e dinâmica, mostre as divulgações repassadas por esse servidor ao interessado final.

Desta forma, o computador da plataforma *Raspberry Pi* é configurado para que no momento que seu sistema operacional iniciar, abra o navegador WEB com o endereço para o servidor local do SID. Utilizará a fonte de alimentação da TV acoplada a ele e necessitando apenas que a mesma seja ligada diretamente a uma tomada de energia. Outro requisito básico é que a TV esteja pré-configurada para a entrada de vídeo HDMI para que o redimensionamento da tela seja feita de forma automática ao ser iniciado o nanocomputador.

A forma de conexão com o servidor consiste na utilização do protocolo de comunicação 802.11 que tem como principal vantagem no contexto de aplicação do SID a eliminação de cabos de rede, facilitando o processo e simplificando a forma de uso da plataforma (GAST, 2005).

Portanto, como esta proposta de implantação faz uso da plataforma *Raspberry Pi*, será necessário a criação de um *script* de inicialização do *Raspbian*, seu sistema operacional, que abra o navegador em tela cheia, ou modo quiosque, requisitando o endereço eletrônico de onde estão presentes as divulgações no servidor do SID, e com isto facilitando o processo de implantação.

4 Integração do SID com o Facebook

A integração do SID com o Facebook deve ser primordialmente vista como uma forma de unificação entre dois sistemas com objetivos, de certa maneira, distintos e que servem um ao outro para que o interessado na informação também utilize das divulgações do SID inseridas neste grande meio de comunicação que é o Facebook. Esta unificação entre estes dois veículos de comunicação permite ao operador do SID a simplificação no que tange a divulgação de informações. Com isso, ao inserir qualquer divulgação no SID, será de forma transparente utilizado somente este para veicular informação em dois sistemas, ou seja, ao ser inserida uma divulgação no SID será também inserida de forma automática no perfil do sistema no Facebook.

Neste capítulo será brevemente mostrado como se deu o processo de integração entre esses dois sistemas, e assim descrever os passos necessários para a efetivação do processo de publicação das divulgações inseridas no sistema SID no Facebook através de um perfil criado para este propósito.

4.1 Visão Geral

A Interface de Programação de Aplicativos – API disponibilizada pela rede social Facebook é a Graph, no SID foi utilizada a versão para a linguagem PHP ([FACEBOOK, 2017](#)). Através dela é possível gerenciar dados da rede social com aplicativos externos. É uma API baseada em requisição/resposta HTTP que possibilita a consulta e gerência de dados presentes no Facebook.

Para manipular os dados nesta API é necessário que permissões da mesma sejam repassadas ao aplicativo de gerenciamento que é disponibilizado ao desenvolvedor como forma de mediar os dados de seu sistema com a Graph API. Este sistema de permissões é baseado através de requisições enviadas para análise a partir do que se deseja executar em cooperação com a Graph API. Estas permissões descrevem como devem ser feitas as requisições à Graph API, definindo os tipos de dados que pretende-se gerenciar, o que deve ser feito, como postagem em páginas ou em perfis. Requisições simples como nome, sobrenome de usuário ou e-mail não necessitam de permissões por se tratarem de atributos públicos tratados nesta API.

A estrutura desta interface é modelada através da Teoria dos Grafos, também apresentada em ([CORMEN, 2009](#)), e portanto é constituída de nós, arestas e atributos. Os nós podem representar vários elementos que são geridos na rede social, tais como como fotos, postagens ou links. As arestas são tratadas como as conexões entre os nós. Por fim,

os atributos são informações sobre os nós, tais como idade, nome completo, e-mail, etc.

A autenticação no módulo administrador é feita a partir do perfil do SID presente na rede social Facebook. Este processo já deveria ocorrer mesmo se não fosse reutilizado como forma de autenticação deste módulo, pois existe esta necessidade de que um perfil para comunicação com o aplicativo da Graph API seja selecionado, e que desta forma optou-se por utilizar desta autenticação para que não houvesse diferentes acessos para um mesmo propósito.

O SID ao inserir uma divulgação em perfil próprio no Facebook utiliza-se desses elementos. A imagem selecionada pelo operador torna-se o nó central na aplicação e os outros atributos são definidos como atributos desse nó, ou seja, a legenda e o linkqr, inseridos pelo operador do sistema ao inserir uma nova divulgação, tornam-se atributos relacionados à imagem. As arestas podem ser definidas como os compartilhamentos deste nó representado pela imagem da divulgação.

A conexão com a Graph API deve ser feita a partir de um aplicativo criado no ambiente do desenvolvedor disponibilizado pelo Facebook. Para isso é preciso criar esse aplicativo e repassar à plataforma as informações que se deseja gerenciar a partir das consultas que serão efetuadas nesta interface.

A forma de acesso seguro à Graph API por esse aplicativo é feita a partir de um token de acesso, que é obtido após o *login* do usuário com suas credenciais do Facebook através da aplicação.

Um token de acesso é uma cadeia de caracteres opaca que identifica um usuário, aplicativo ou Página. Ele pode ser usado pelo aplicativo para fazer chamadas da Graph API. Tokens de acesso são obtidos por meio de uma série de métodos, que são implementados e codificados pela Graph API e inclui informações sobre quando o token expirará e qual aplicativo gerou o token. Devido às verificações de privacidade, a maioria das chamadas de API no Facebook precisa incluir um token de acesso ([FACEBOOK, 2017](#)).

Desta forma, o token identifica a sessão do usuário no Facebook. Ao obtê-lo, é disponibilizado controle ao aplicativo sobre as postagens e informações do usuário. De acordo com as especificações na documentação da Graph ([FACEBOOK, 2017](#)), a aplicação então poderá enviar requisições HTTP para os nós ou para as arestas destes nós através de seus identificadores únicos e obter as respostas HTTP do Facebook.

Um sistema de permissões é utilizando na Graph API para controlar o acesso e edição de informações. Assim, para que alguma publicação possa ser feita ou visualizada pelo aplicativo, é necessário possuir as permissões adequadas. As permissões que o aplicativo do SID na plataforma detém são as básicas disponibilizadas a partir da criação inicial de qualquer aplicativo, e desta forma são a “publish_action” e “email”.

A permissão “email” serve para efetuar consultas relacionadas ao email do usuário

que está requisitando à Graph e seu uso torna-se bastante eficiente em sistemas que necessitam de realização de um cadastro específico, podendo suas informações serem trazidas diretamente do Facebook.

A permissão “publish_actions” é a que de fato permite o gerenciamento das informações em perfis de usuários do Facebook. Com essa permissão é possível que a Graph API permita que um usuário de um sistema insira, delete, altere e apresente informações postadas nesse perfil em questão.

4.1.1 Elementos da API utilizados no SID

O SID utiliza-se do nó “photo” como forma de publicação da divulgação no perfil do SID e representa uma foto individual no Facebook. As permissões para gerenciamento deste pode ser repassado a partir da permissão “user_photos” ou “user_posts”, mas que na integração implementada no SID utilizou-se apenas da permissão “publish_actions” que abarca as permissões anteriores de forma conjunta.

Com isso, os atributos do nó “photo” são utilizados para mapear o mesmo na Graph API, por exemplo, o atributo “fbid” tem como valor uma cadeia numérica e seu objetivo é a identificação da foto como um nó específico. Outro atributo importante utilizado neste nó é o atributo “source”, no entanto, cabe ressaltar que esse atributo encontra-se descontinuado em atualizações na Graph API mas que no processo de integração do SID com a mesma ainda está em uso, pois no momento da codificação da integração a Graph API atual era a versão 2.6, porém na versão 2.8 está descontinuada e pode ser substituída com o atributo “images” ([FACEBOOK, 2017](#)).

Neste sentido, o último atributo utilizado neste nó é o “caption”, e trata-se da legenda fornecida pelo operador na inserção da divulgação no SID. Cabe ressaltar também que, pelo mesmo motivo supracitado com respeito à desatualização acelerada da Graph API, encontra-se neste momento substituída pelo atributo “name”, mas que na implementação atual do SID faz uso como atributo “caption”.

As arestas podem ser construídas de diversas formas, como “likes”, “reactions”, “sharedposts” e “comments”. Os “likes” são estruturados na divulgação como forma de pessoas que gostaram/curtiram a publicação. As “reactions” seguem a mesma linha mas que possuem coleções definidas para a ação. Os “sharedposts” representa as postagens em que esta foto é compartilhada. E por fim, os comentários da publicação que são elencados na aresta “comments” ([FACEBOOK, 2017](#)).

As consultas do SID com o Facebook através do aplicativo na plataforma para desenvolvedores disponibilizada por esta mesma rede social, estão restritas à autenticação de usuário, com o perfil do SID no Facebook como requisito para acesso ao sistema em questão, e à postagem de qualquer inserção de divulgação feita no SID ao Facebook,

sendo também necessário a completa estruturação da divulgação para fidedignidade das informações a serem repassadas. Essa estruturação se dá pelo fato de que alguns elementos presentes no objeto divulgação a ser repassado deverão ser omitidos ao Facebook, como por exemplo, a data de termino da divulgação, uma vez que não faz sentido tal informação permanecer na divulgação postada na rede social.

A autenticação do SID com o Facebook se efetivou a partir do conhecimento do parâmetro de controle presente no perfil do SID na própria rede social. O “fbId” presente no banco de dados do SID deve ser correspondente a esse mesmo identificador que é retornado a partir do acesso à Graph API.

Para isso é necessário criar um objeto Facebook, presente nos arquivos de estruturação de comunicação à Graph API. Este objeto deve ter o identificador do aplicativo mapeado como “app_id”, o identificador de segurança mapeado como “app_secret”, a versão da Graph API que está sendo utilizada para comunicação com o Facebook, a versão atual do SID utiliza a v2.6, e por fim deve conter o parâmetro responsável por gerenciar o *upload* dos arquivos a serem repassados ao Facebook, ou seja, a imagem, sendo mapeado como “fileUpload” e recebendo “true” em seu contexto.

O principal parâmetro de retorno é chamado de “access_token” e servirá de chave para o gerenciamento através do aplicativo da Graph API para enviar as divulgações para o perfil do SID. Esse número de acesso tem como principal finalidade autenticar o acesso ao aplicativo como requisito para permissão de sessão criada pelo próprio aplicativo da Graph API.

Portanto, o segredo do aplicativo ou um token de acesso do aplicativo nunca deve ser incluído em qualquer código possível de ser acessado por qualquer pessoa que não seja um desenvolvedor do aplicativo. Isso se aplica a todos os métodos de código que não são protegidos, como o código do lado do cliente (como HTML ou JavaScript) ou aplicativos nativos que poderiam ser descompilados ([FACEBOOK, 2017](#)).

4.1.2 Integração

Qualquer divulgação inserida no perfil do SID no Facebook utiliza-se do nó “photo”, ou seja, a divulgação que o SID repassa à Graph API contém a mesma estrutura de uma imagem postada por um usuário convencional desta rede social.

O processo de integração do SID com o Facebook para o acesso ao módulo administrador é feito a partir da criação de um objeto Facebook estruturado pela Graph API e com isso repassar os parâmetros necessários para tal criação. Assim, o objeto Facebook criado acessa uma de suas funções que é mapeada na API para o PHP como “getRedirectLoginHelper” deve ser repassado para um outro objeto visto como um ajudante ou *helper* que será utilizado para executar de fato a requisição para a Graph API. Prosseguindo o

processo, deverá ser repassado à função do objeto *helper* o endereço para redirecionamento da resposta no SID e as permissões das funcionalidades que se deseja fazer uso.

No processo de *callback* é que será de fato executado a verificação do identificador do perfil do SID, ou seja, se o “fbid” retornado pelo processo de *login* com a Graph API corresponder ao identificador do SID o acesso ao módulo administrador será permitido, caso contrário o acesso não será válido e a página para *login* deverá ser retornada ao suposto operador. Esta verificação é feita a partir do sistema SID, necessitando somente do identificador do perfil que está logado no Facebook e que o retorno seja congruente ao do perfil do SID nessa rede social.

O processo de divulgação das informações a serem inseridas no SID através do módulo administrador no Facebook deve ser transparente ao operador, ou seja, a partir da inserção de uma divulgação no SID deverá ser automaticamente estruturada, esta divulgação, no contexto do Facebook e repassada de forma que o operador não execute operações distintas para tais funcionalidades.

Assim, o envio é feito a partir da criação do objeto Facebook e com a obtenção do “access_token” a partir do *login*, processo anterior a este.

A Figura 12 mostra como é estruturada uma divulgação enviada ao perfil do SID no Facebook através da integração com a Graph API. Neste caso, o campo legenda recebe “Sistema inteligente de Divulgação de Informações do IFG Câmpus Formosa - SID, Conheça mais!”, o campo linkqr “https://dsnnunes.com.br/SID”.

Desta forma, a criação desse objeto Facebook, deverá então, ser carregada a imagem da divulgação como parâmetro para este objeto. Isso é feito a partir do método “fileToUpload” e tendo como parâmetro a ser repassado a imagem da divulgação em questão. Prosseguindo, uma lista de objetos deverá ser criada pois, a mesma deverá ser repassada no momento de envio da divulgação para a Graph API contendo todos os elementos necessários para a correta estruturação da divulgação na rede social. O repasse do “access_token” é imprescindível para a efetivação da publicação no perfil do SID.

A efetivação da publicação é feita a partir da chamada do método “post” do objeto Facebook, restrito somente à imagens por estar-se utilizando do nó “photo” da Graph API, que recebe como parâmetro o local onde deverá ser postada a publicação/imagem no *feed* de notícias do SID, que corresponde, esse local, à “/me/photos”, e o objeto criado anteriormente a este processo que guarda os dados estruturados da publicação, mapeado no SID como “\$data”. Mensagens de erro serão repassadas ao operador sobre a ineficácia no decorrer deste processo.



Figura 12 – Divulgação enviada ao perfil do SID.

5 Resultados

A concepção inicial do SID, passou por diversas reformulações arquiteturais até o estado atual. Estas características iniciais foram expostas por (BISPO; NUNES, 2015). O SID até então não possuía arquitetura WEB e poderia ser visto apenas como um formatador de imagens com inserção do código QR de maneira automática acoplado com um apresentador de *slides*. O presente trabalho adaptou os seus conceitos ao ambiente WEB para viabilizar a sua implantação no Câmpus.

O SID encontra-se em sua segunda versão de acordo com as características expostas no Capítulo 3 e pronto para ser implantado de modo a servir de aparato para suprir as necessidades do Câmpus quanto ao processo de veiculação de informação através da sinalização digital de forma atrativa e dinâmica. Dado que observados os resultados, o SID tem perspectiva de colaborar com o Campus no que tange a criação de uma ambiente adequado para assimilação de informação.

O SID, tem como finalidade agilizar e efetivar o processo de assimilação da informação no Câmpus de maneira atrativa e dinâmica utilizando-se da sinalização digital. Atende ao seu objetivo principal de divulgar informações através de uma plataforma WEB com arquitetura Cliente-Servidor, sendo esta dividida em dois módulos de sistema, administrador e cliente. Esta divisão delimita os papéis dos usuários e mantenedores do SID.

O módulo administrador efetua todas as rotinas responsáveis pelo gerenciamento das informações. A autenticação deste módulo é resultado da integração com a rede social Facebook (SOBRINHO; NUNES, 2016), na qual também possibilita que qualquer informação inserida no SID seja, de forma transparente ao operador, enviada à esta rede social como forma de unificar estes dois veículos de comunicação.

O módulo cliente fica responsável somente por apresentar as informações requisitadas no servidor. Sua estruturação visual pode ser visualizada na Figura 4. Este módulo pode ser visualizado em diferentes tipos de dispositivos, o que contribui para a portabilidade do sistema.

O módulo administrador do SID permite a criação de divulgações interativas através da inserção de códigos QR de maneira automática, desde que o *link* seja inserido durante a inserção da divulgação neste módulo. Desta forma, este *link* inserido no módulo administrador é repassado como forma de código QR para o cliente, servindo como aparato de interação entre o transeunte e a informação.

Em sua interface, o SID procurou facilitar o seu manuseamento pelo operador,

fazendo com que o processo de inserção de informações seja feito de forma simples e rápida. Algumas das soluções encontradas atualmente como a *Raspberry Slide Show* ou *Info-beamer*, não facilitam o processo de inserção divulgações, requerendo que as divulgações sejam previamente formatadas para então serem inseridas.

A proposta de metodologia de implantação do sistema SID em conjunto com a plataforma *Raspberry Pi*, objetivando economia energética, financeira e de espaço físico é compartilhada com algumas das soluções utilizadas atualmente como a *Raspberry Slide Show* e *Raspberry Digital Signage*. A primeira solução utiliza um computador *Raspberry Pi* para gerenciar e apresentar as divulgações, o que acarreta em problemas relacionados à descentralização de suas funções para outros dispositivos, pois cada computador *Raspberry Pi* é visto, de forma análoga ao SID, como servidor e cliente. Desta forma, o SID em relação a esta solução, a supera pois seu módulo cliente pode ser executado em quaisquer dispositivos com um navegador WEB convencional, devido ao desacoplamento dos módulos clientes e servidor e do fato que todo o conteúdo está disponível em um servidor central, bastando os diversos clientes requisitarem os objetos via consultas HTTP.

A segunda solução, *Raspberry Digital Signage*, trata-se simplesmente de uma distribuição que desempenha funções de cliente ao receber as divulgações de um endereço eletrônico de sinalização digital via HTTP. Este processo é semelhante ao que é executado no módulo cliente do SID, fazendo uso de requisições de objetos JSON. No entanto, esta solução nada faz para alimentar o endereço eletrônico com novas divulgações, dependendo de implementação de aplicação externa que possua permissões para gerenciar as divulgações no lado servidor.

Em ambas as soluções, a integração com as redes sociais está restrita a implementações a nível de aplicação presente no servidor não constando suporte nativo. O SID não concentra-se somente em apresentar tais divulgações como ocorre na solução *Raspberry Digital Signage*, como também preocupa-se em implementar em um veículo unificado de divulgação de informação ao conciliar sinalização digital e postagens no perfil do SID no Facebook.

Outra solução encontrada na literatura corresponde ao software Xibo, sistema baseado em arquitetura Cliente-Servidor com sistema gerenciador de conteúdo incluso que permite diversas customizações, necessitando que cada divulgação seja inicialmente estruturada para então ser inserida. Ele possibilita diversidade de formatações visuais repassadas ao interessado. O SID possibilita somente uma única estrutura fixa para todas as divulgações, contudo viabiliza a facilidade, uniformidade e rapidez na inserção de informação na plataforma, o que no caso do Xibo pode ser um pouco mais trabalhoso. O Xibo ainda fica restrito ao conhecimento do operador sobre a disposição dos elementos visuais de forma eficiente possibilitando assim, que haja dificuldade no processo de inserção da informação repassada de modo a garantir que ela seja assimilada pelos usuários,

caso operado por uma pessoa sem os conhecimentos técnicos necessários.

O Info-beamer oferece aos seus utilizadores diversas formas de customizações através de operações executadas diretamente em linguagem de programação própria, o que dificulta que um operador não conhecedor destas linguagens opere o sistema. Suporta interoperabilidade com outras aplicações WEB através de consultas JSON. Cabe ressaltar que esta forma de interoperabilidade não é nativa, pois é necessária a presença de outras aplicações que produzam este tipo de arquivo. Assim, pode-se concluir que esta solução não possui integração nativa com redes sociais, no entanto existe a possibilidade da integração a partir implementações de outras aplicações que façam uso das consultas JSON. O SID, por ser uma aplicação WEB de fácil manuseio, reserva espaço para que seu operador não seja conhecedor linguagens de programação, como os servidores do setor de Comunicação Social. Cabe ressaltar novamente que o SID é integrado à rede social Facebook de forma nativa, efetuando postagens no perfil do facebook atrelado ao Sistema sempre que uma divulgação é inserida.

Outro aspecto tratado pelo SID está em sua forma de gerenciamento e armazenamento das informações. Algumas das soluções comparadas ao SID, como a *Raspberry Slide Show* e Info-beamer, estão interessadas em utilizar processos de inserção complicados e muitas vezes ineficientes, como a leitura de uma lista que contém as imagens a serem repassadas sequencialmente ou inserção a partir de uma linguagem de *script* própria, fazendo com que o operador destes sistemas execute processos vistos como obstáculos quanto à sua utilização. O SID ultrapassa no quesito usabilidade e manutenibilidade tais sistemas, pois sua inserção é feita de forma simples, necessitando somente que o operador contenha os dados a serem inseridos presentes na máquina com acesso ao módulo administrador. Este operador não necessita de ser conhecedor de sistemas de informação ou de linguagens de programação, o que facilita e agiliza o processo de inserção de informações pelo operador do SID.

Assim, pode-se dizer que o SID cumpre com o seu objetivo principal, no qual é de disponibilizar informações através da sinalização digital e atenta-se uma série de requisitos não fornecidos por outras aplicações presentes na literatura.

O SID tem o propósito de sanar os aspectos que ainda não foram ou não são possíveis de serem inseridos nas soluções com ele comparadas acima. A Tabela 2 tem o intuito de adicionar o SID ao quadro comparativo mostrado na Tabela 1 e, de forma sucinta, descrever quais foram os benefícios do sistema com respeito às soluções anteriormente supracitadas. Os elementos comparativos são descritos a seguir:

- I) Comprometimento com o propósito: o sistema em questão possibilita a veiculação de informações através de mecanismos de sinalização digital?
- II) Criação simples de divulgações: o operador possui facilidade de incluir novas divul-

gações com aspecto atrativo?

- III) Portabilidade: é possível visualizar a divulgação em diferentes dispositivos?
- IV) Integração com redes sociais: o sistema integra-se nativamente de alguma forma com redes sociais?
- V) O conteúdo é gerido em um dispositivo diferente em que é apresentado, fortalecendo a descentralização e manutenção?

Tabela 2 – Tabela comparativa de elementos das soluções incluindo o SID.

Quesito/Solução	Raspberry Slideshow	Raspberry Digital Signage	Xibo	Info-beamer	SID
I	Sim	Sim	Sim	Sim	Sim
II	Sim	NSA	Sim	Não	Sim
III	Não	Sim	Sim	Não	Sim
IV	Não	Não	Não	Não	Sim
V	Não	Sim	Sim	Não	Sim

O sistema SID serve de ferramenta para veiculação efetiva de informação no Câmpus. Ao utilizar-se da sinalização digital supera os meios de comunicação rígidos e obsoletos presentes no Câmpus. Não tem como finalidade eliminar tais meios, mas de servir como aparato a mais para que o processo de veiculação de informação ocorra de maneira simples, rápida e dinâmica. Faz uso de ferramentas comumente utilizadas no ambiente WEB, para que seja portátil e descentralizado no que diz respeito à distribuição de informação em diferentes dispositivos através do seu módulo cliente. Possibilita que o sistema seja gerenciado por operadores que não possuem conhecimento em sistemas de informação. Expõe ao transeunte de formas de interação com a informação como sua integração com a rede social Facebook e códigos QR, para que o mesmo permaneça em constante contato com a informação em questão.

6 Considerações Finais

O SID tem por objetivo efetivar e aprimorar os meios de comunicação do IFG - Câmpus Formosa. Utilizando-se da sinalização e de objetos de integração com a rede social Facebook o sistema tem grande potencial para suprir a demanda de informações dentro e fora do Câmpus, sendo dentro a partir de telas espalhadas pelo Câmpus e fora pelas redes sociais a ele integradas. Ressalta-se que atualmente o SID se encontra integrado com o Facebook.

Utilizando-se da arquitetura Cliente-Servidor, o SID é dividido em dois módulos, sendo as funções relativas ao gerenciamento das informações estão restritas somente a um desses. O módulo administrador é capaz de desempenhar funções de entrada e saída de dados assim como o armazenamento dessas informações todas em banco de dados.

Até o momento o SID mantém-se como um sistema simples para divulgação de informações, porém o mesmo tende a crescer o seu escopo e abarcar elementos que sejam dados como objetos de informações e podendo ser a ele incorporados.

O objetivo principal contido na motivação encontra-se implementado no SID, no qual qualquer informação relativa ao ambiente do Câmpus pode virar objeto a ser repassado aos interessados estruturados em dois módulos que operam em conjunto para que esse processo de divulgação de informações seja efetivado. O SID visa a rapidez, dinamicidade e atratividade das informações nele contidas, com sua interface amigável projetada para ser implantada em qualquer dispositivo com suporte a um navegador WEB o sistema proporciona portabilidade, sendo necessário uma rede interna e conexão à *internet* para o módulo administrador para que as rotinas de inserção na rede social Facebook sejam efetivadas.

O módulo administrador é capaz de efetuar o gerenciamento das informações contidas no banco de dados do sistema. O cliente executa as funções relativas à estruturação das divulgações requisitadas a partir da servidor com o banco de dados do SID. Esse servidor pode ser implementado de forma local uma vez que o sistema SID se resume em uma plataforma WEB e assim necessitando de ferramentas simples para efetuar operações de chamadas a esse servidor na rede local.

A integração do SID com a rede social Facebook efetivou-se a partir da Graph API, construída com o propósito de integração e comunicação dos elementos presentes em seu banco de dados com sistemas diversos construídos por terceiros. Esta integração foi viabilizada com a implementação do sistema de autenticação disponibilizado por esta rede social e para publicação das divulgações inseridas no SID a partir do módulo administrador de forma transparente ao operador.

O SID obtém benefícios consideráveis em relação às outras soluções que são utilizadas atualmente. Por possuir divisão entre gerenciamento e apresentação, possibilita que diferentes dispositivos façam uso de suas divulgações. Desta forma, o principal benefício elencado pelo SID é a forma simples e intuitiva de divulgação, facilitando a portabilidade e manutenção do sistema a nível de código.

Outro aspecto relevante que o SID possui em relação às outras soluções, é a integração com o Facebook de forma nativa, ou seja, não necessita de que aplicações externas sejam desenvolvidas para que suas informações sejam repassadas às redes sociais, como é o caso da solução Info-beamer.

A utilização de código QR como forma de introduzir o interessado em mais detalhes da informação se mostrou bastante vantajosa com relação às soluções atuais, pois em nenhum dos casos tal premissa foi aplicada nesses sistemas.

6.1 Trabalhos futuros

O SID mantém seu escopo arquitetural aberto para inserção de novos módulos ou funcionalidades e, desta forma, poderá expandir-se a partir da necessidade de divulgação de informações do Câmpus.

Poderá receber módulos que possam abarcar outros aspectos do Câmpus tais como informações institucionais destinadas estritamente aos docentes ou técnicos, assim como para a comunidade não acadêmica presente no Câmpus. Como sugestão poderá futuramente acolher a separação entre eventos e divulgações de informações, pois a ideia é que todo evento é uma divulgação mas que nem toda divulgação é um evento e, desta forma, pode existir a separação de tais elementos como requisito para a construção de um novo módulo que efetive de fato essa separação.

Com isso, novos operadores para o módulo administrador poderão ser visados, porém necessitando sempre de que exista uma forma de validação destas divulgações por um administrador mestre, ou seja, que o controle das divulgações sejam feitas por um único operador, definindo o que pode ser repassado ao cliente ou não. Este controle serve para que informações tendenciosas ou inapropriadas sejam repassadas aos interessados mantendo sempre o SID como um sistema que veicula de forma simples, dinâmica e atrativa de informações apropriadas aos interessados.

Haverá também a possibilidade de implementação do módulo cliente com interface compatível para visualização em dispositivos móveis, podendo se feito com a própria estrutura existente possibilitando que um *feed* de divulgações seja repassado para tais dispositivos.

O sistema ainda carece de informações em tempo real, necessitando assim de que

a integração com as redes sociais torne-se requisito importante para o sistema. Como exemplo, a integração com o Facebook poderá ser feita com os comentários, quantidade de *likes* e número de compartilhamentos da divulgação inserida através da Graph API pois até o momento esta integração somente executa o processo de publicação da divulgação inserida no módulo administrador no perfil do SID no Facebook.

Outro fator relacionado à integração está em integrar com outras redes sociais que servem como meio de veiculação de informações na atualidade tais como o Twitter, Youtube ou Instagram.

Portanto, o SID poderá tornar-se em uma ferramenta mais completa para veiculação efetiva de informações tanto no ambiente do Câmpus através do módulo cliente quanto pelas formas de integração com as redes sociais, aumentando seu campo de atuação e beneficiando aqueles que dependem de informação estruturada e precisa, eliminando a ideia de procura desta pelo interessado ao levar ela de maneira atrativa, dinâmica e interativa para os usuários.

Uma avaliação do sistema por transeuntes e uma validação estatística dessa avaliação também poderão ser feitas para aferir a efetividade do sistema SID no Câmpus.

Referências

- ALBAGLI, S. Divulgação científica: informação científica para a cidadania. *Ciência da informação*, v. 25, n. 3, p. 396–404, 1996. Citado na página 19.
- ALMEIDA, E. S. de et al. Ambap: Um ambiente de apoio ao aprendizado de programação. In: *Anais do XXII Congresso da Sociedade Brasileira de Computação*. [S.l.: s.n.], 2002. v. 4, p. 79–88. Citado na página 23.
- ANDRADE, F. F. d. *Desenvolvimento de aplicações web com a utilização dos frameworks CodeIgniter e Doctrine*. Dissertação (B.S. thesis) — Universidade Tecnológica Federal do Paraná, 2011. Citado na página 29.
- BARRETO, A. d. A. A condição da informação. *São Paulo em Perspectiva*, SciELO Brasil, v. 16, n. 3, p. 67–74, 2002. Citado na página 11.
- BARRETO, A. de A. Perspectivas da ciência da informação. *Revista de Biblioteconomia de Brasília*, v. 21, n. 2, p. 156–166, 1997. Citado na página 13.
- BISPO, A. A.; NUNES, D. S. N. Proposta de Sistema Inteligente de Divulgação de Informações do IFG Formosa. In: *Anais de Congresso da VI SECITEC do IFG - Formosa, 16 a 20 de novembro*. [S.l.: s.n.], 2015. Citado 3 vezes nas páginas 11, 33 e 51.
- BISSI, W. Scrum-metodologia de desenvolvimento ágil. *Campo Digital*, v. 2, n. 1, p. 03–06, 2007. Citado 2 vezes nas páginas 27 e 28.
- BONOTO, R. G.; JUNIOR, E. A. O. O padrão model-view-controller apoiado pelo framework zend. Citado 2 vezes nas páginas 28 e 29.
- BOOCH, G.; RUMBAUGH, J.; JACOBSON, I. *UML: guia do usuário*. [S.l.]: Elsevier Brasil, 2006. Citado 3 vezes nas páginas 26, 27 e 38.
- CARDOSO, E. et al. Contribuição metodológica em design de sinalização. *InfoDesign: Revista Brasileira de Design da Informação*. [São Paulo], SBDI. Vol. 8, n. 1 (2011), p. 10-30, 2011. Citado na página 19.
- COHN, M. *Desenvolvimento de software com Scrum: aplicando métodos ágeis com sucesso*. [S.l.]: Grupo A-Bookman, 2000. Citado 2 vezes nas páginas 16 e 34.
- CONFORTO, D.; SANTAROSA, L. M. Acessibilidade à web: Internet para todos. *Revista de Informática na Educação: Teoria, Prática-PGIE/UFRGS*, v. 5, n. 2, p. 87–102, 2002. Citado na página 22.
- CORMEN, T. H. *Introduction to algorithms*. [S.l.]: MIT press, 2009. Citado na página 45.
- CROCKFORD, D. The application/json media type for javascript object notation (json). 2006. Citado na página 26.
- DOCTRINE. *Doctrine*. 2017. Acesso em: 22 fevereiro de 2017. Disponível em: <<http://www.doctrine-project.org/>>. Citado 2 vezes nas páginas 29 e 34.

- DURAN, N.; MORAIS, P. C. d.; MATTOSO, L. H. C. *Nanotecnologia: introdução, preparação e caracterização de nanomateriais e exemplos de aplicação*. [S.l.]: Artliber, 2006. Citado na página 20.
- ELMASRI, R.; NAVATHE, S. B. Sistema de banco de dados. 6ª edição. São Paulo: Person. 837p, 2011. Citado 2 vezes nas páginas 24 e 25.
- EMOTIONS, B. *Raspberry Slideshow*. 2017. Acesso em: 22 fevereiro de 2017. Disponível em: <<http://www.binaryemotions.com/raspberry-slideshow-download/>>. Citado na página 30.
- FACEBOOK. *Graph API Reference*. 2017. Acesso em: 17 fevereiro de 2017. Disponível em: <<https://developers.facebook.com/docs/graph-api/reference>>. Citado 5 vezes nas páginas 12, 45, 46, 47 e 48.
- FALBO, R. d. A.; TRAVASSOS, G. H. A integração de conhecimento em um ambiente de desenvolvimento de software. In: *II Congreso Argentino de Ciencias de la Computación*. [S.l.: s.n.], 1996. Nenhuma citação no texto.
- FLANAGAN, D. *JavaScript: the definitive guide*. [S.l.]: "O'Reilly Media, Inc.", 2006. Citado na página 34.
- FRAMEWORK, Z. *ZendFramework 2*. 2017. Acesso em: 22 fevereiro de 2017. Disponível em: <<https://framework.zend.com/learn>>. Citado 2 vezes nas páginas 29 e 34.
- GAST, M. *802.11 wireless networks: the definitive guide*. [S.l.]: "O'Reilly Media, Inc.", 2005. Citado na página 44.
- GUDWIN, R. R. Linguagens de programação. Campinas: DCA/FEEC/UNICAMP, 1997. Citado na página 24.
- INFO-BEAMER. 2017. Acesso em: 26 fevereiro de 2017. Disponível em: <<https://info-beamer.com/pi>>. Citado na página 31.
- KOZOVITS, L. E.; FEIJÓ, B. *Arquitetura para jogos massive multiplayer*. [S.l.]: PUC, 2003. Citado na página 23.
- KUROSE, J. F. et al. *Redes de Computadores ea Internet: uma abordagem top-down*. [S.l.]: Pearson, 2010. Citado 2 vezes nas páginas 21 e 22.
- LIMA, F. d. A. et al. Implantação e análise de desempenho de um cluster com processadores arm e plataforma raspberry pi. Universidade Federal de Sergipe, 2016. Citado na página 44.
- MACHADO, J. d. A. A sinalização digital como ferramenta de relações públicas: o caso Light SA. *Trabalho de Conclusão do Curso de Comunicação Social*, Porto Alegre: UFRG, 2010. Citado na página 11.
- MARTINS, J. C. C. *Gerenciando Projetos de Desenvolvimento de Software com PMI, RUP e UML*. [S.l.]: Brasport, 2010. Citado na página 24.
- MELO, A. C. *Desenvolvendo Aplicações com UML 2.2*. [S.l.]: Brasport, 2004. Citado 3 vezes nas páginas 24, 26 e 27.

- MESSIAS, L. C. d. S. *Informação: um estudo exploratório do conceito em periódicos científicos brasileiros da área de Ciência da Informação*. Tese (Doutorado) — UNESP, 2005. Citado na página 11.
- MILANI, A. *PostgreSQL-Guia do Programador*. [S.l.]: Novatec Editora, 2008. Citado na página 42.
- MOMJIAN, B. *PostgreSQL: introduction and concepts*. [S.l.]: Addison-Wesley New York, 2001. Citado na página 42.
- PI, R. Raspberry pi. *Raspberry Pi 1 HDMI 13 Secure Digital 34 Universal Serial Bus 56 Python (programming language) 84*, p. 1, 2012. Citado na página 20.
- PIRES, J. Sistemas de telecomunicações i. *Departamento de Engenharia Electrotécnica e Computadores*, 1999. Citado na página 33.
- POSTGRESQL. *Documentação PostgreSQL 8.0.0*. 2017. Acesso em: 17 fevereiro de 2017. Disponível em: <<http://pgdocptbr.sourceforge.net/pg80/datatype-binary.html>>. Citado 2 vezes nas páginas 34 e 43.
- RIBEIRO, M. W. S.; ZORZAL, E. R. Realidade virtual e aumentada: Aplicações e tendências. *XIII Simpósio de Realidade Virtual e Aumentada, Uberlândia-MG-Brasil*, 2011. Citado na página 13.
- RICHARDSON, M.; WALLACE, S. Primeiros passos com o raspberry pi. *Primeira Edição. Novatec Editora Ltda*, 2013. Citado 2 vezes nas páginas 20 e 21.
- SALVADORI, I. L. et al. Desenvolvimento de web apis restful semânticas baseadas em json. 2015. Citado na página 26.
- SEBESTA, R. W. *Conceitos de linguagens de programação*. [S.l.]: Bookman Editora, 2009. Citado na página 23.
- SILVA, M. Jquery-a biblioteca do programador javascript–2ª edição. *Novatec*, 2010. Citado na página 34.
- SILVA, M. J.; FERNANDES, C. T. et al. Ac3as-web: ambiente cooperativo de apoio à avaliação de aprendizagem significativa na web. In: *Anais do Simpósio Brasileiro de Informática na Educação*. [S.l.: s.n.], 2002. v. 1, n. 1, p. 20–30. Citado na página 22.
- SILVA, R. C. d. Benchmark em banco de dados multimídia: análise de desempenho em recuperação de objetos multimídia. 2006. Citado na página 43.
- SOBRINHO, M. de C.; NUNES, D. S. N. Integração do sistema inteligente de divulgação de informações do IFG-Formosa ao Facebook através da Graph-API. In: *Anais de Congresso da VII SECITEC do IFG - Formosa, 17 a 21 de outubro*. [S.l.: s.n.], 2016. Citado na página 51.
- SOON, T. J. Qr code. *Synthesis Journal*, v. 2008, p. 59–78, 2008. Citado na página 29.
- SOUZA, C. M. de. Visualg-ferramenta de apoio ao ensino de programação. *Revista Eletrônica TECCEN*, v. 2, n. 2, p. 01–09, 2009. Citado na página 24.
- SOUZA, R. B. et al. Sisal, um sistema supervisor para elevação artificial de petróleo. In: *Rio Oil & Gas Expo and Conference*. [S.l.: s.n.], 2006. p. 11–14. Citado na página 22.

- SPORNY, M. et al. *Json-ld 1.0-a json-based serialization for linked data. w3c recommendation*. 2014. Citado na página 26.
- TEIXEIRA, M. M.; SANTANA, M. J.; SANTANA, R. H. Servidor web com diferenciação de serviços: Fornecendo qos para os serviços da internet. 2005. Citado na página 22.
- TOMAÉL, M. I.; ALCARÁ, A. R.; CHIARA, I. G. D. Das redes sociais à inovação. *Ciência da informação, Brasília*, SciELO Brasil, v. 34, n. 2, p. 93–104, 2005. Citado na página 11.
- UEBELE, A. *Signage systems & information graphics: a professional sourcebook*. [S.l.]: Thames & Hudson, 2007. Citado na página 19.
- XIBO. *why Xibo?* 2017. Acesso em: 26 fevereiro de 2017. Disponível em: <<http://xibo.org.uk/>>. Citado na página 31.