

Prospectiva (Frutal).

Aplicativo móvel institucional para a UEMG-Frutal.

Pedro Henrique Martins dos Santos.

Cita:

Pedro Henrique Martins dos Santos (2016). *Aplicativo móvel institucional para a UEMG-Frutal*. Frutal: Prospectiva.

Dirección estable: <https://www.aacademica.org/repositorio.digital.uemg.frutal/41>

ARK: <https://n2t.net/ark:/13683/pZsz/XpS>



Esta obra está bajo una licencia de Creative Commons.
Para ver una copia de esta licencia, visite
<https://creativecommons.org/licenses/by-nc-nd/4.0/deed.es>.

Acta Académica es un proyecto académico sin fines de lucro enmarcado en la iniciativa de acceso abierto. Acta Académica fue creado para facilitar a investigadores de todo el mundo el compartir su producción académica. Para crear un perfil gratuitamente o acceder a otros trabajos visite: <https://www.aacademica.org>.



**Aplicativo móvel
institucional para
a UEMG-Frutal**

**Pedro Henrique
Martins dos Santos**

COLEÇÃO
Produzir Cidadania

EDITORA
PROSPECTIVA

Pedro Henrique Martins dos Santos

Aplicativo móvel institucional para a UEMG-Frutal

Frutal-MG
Editora Prospectiva
2016

Copyright 2016 by Pedro Henrique Martins dos Santos

Capa: Jéssica Caetano

Foto de capa:

<http://www.uemgfrutal.org.br/fotos/NotI20151029093828.jpg>

Revisão: o Autor

Edição: Editora Prospectiva

Editor: Otávio Luiz Machado

Assistente de edição: Jéssica Caetano

Conselho Editorial: Antenor Rodrigues Barbosa Jr, Flávio Ribeiro da Costa, Leandro de Souza Pinheiro, Otávio Luiz Machado e Rodrigo Portari.

Contato da editora: editorapropectiva@gmail.com

Página: <https://www.facebook.com/editorapropectiva/>

Telefone: (34) 99777-3102

Correspondência: Caixa Postal 25 – 38200-000 Frutal-MG

Santos, Pedro Henrique Martins dos.

Aplicativo móvel institucional para a UEMG-Frutal. Frutal: Prospectiva, 2016.

ISBN: 978-85-5864-008-4

1. Android 2. Aplicativo Móvel. 3. Programação Extrema. I. Santos, Pedro Henrique Martins dos Santos. II. Universidade do Estado de Minas Gerais. III. Título.

Dedico esse trabalho aos meus pais e a minha irmã, pois sem o apoio deles seria impossível passar por esta etapa.

AGRADECIMENTOS

Agradeço aos meus pais, Euler e Débia por me mostrarem que o conhecimento é o melhor caminho para a resolução de qualquer problema e por se importarem com meu futuro investindo em minha educação, serei eternamente grato.

Agradeço minha irmã, Gabriela Zaki por ser sempre doce e compreensível, apesar da distância, sempre estive ao meu lado me alegrando com seus risos.

Agradeço a Sara, minha amiga, namorada e companheira, por me mostrar que quando se está ao lado de quem se ama o impossível se torna algo ilusório. Você é minha dádiva.

Aos meus grandes amigos Vitor e Lucas, por terem compartilhado todos esses anos, com muitas risadas, boas histórias e grandes ideias que ainda iremos colocar em prática. Vocês são para vida.

Agradeço a Oitava turma de Sistemas de Informação, todos vocês que passaram por ela foram sensacionais. Em especial: Carolina Mendonça, Adriano Neves e Welder Martins.

Agradeço meus amigos de Belo Horizonte, que apesar da distância sempre estiveram me apoiando e sendo verdadeiros irmãos. Em especial: Agatha Pedersen, Gustavo Figueiredo e Eduardo Siriani.

Agradeço à Ordem Demolay, por me torna um ser humano cada dia melhor, em especial meus estimados Irmãos Moisés Augusto, Gustavo Chaves e Leonardo Borges pelas viagens, companheirismo e autêntico espírito fraternidade, que a luz seja sempre nosso caminho.

Agradeço aos meus companheiros da República EmBHdos, vocês me mostram que apesar de todas as diferenças a amizade é algo maior.

Agradeço a todos os professores do curso de Sistemas de Informação, vocês foram essenciais para minha formação.

Agradeço a Kamila Rios, pela paciência nessa etapa final, e por mostrar que perfeição deve ser sempre almejada.

Agradeço ao meu orientador, Daniel Bruno, mais conhecido como DB, que me ensinou que sou capaz de andar com meus próprios pés e me mostrou o melhor do meu potencial. Também por aceitar orientar este trabalho e ser paciente nos momentos que eu lhe enviava mensagens em todas as redes sociais possíveis.

Agradeço então ao Pai Celestial, por ter me dado força e pessoas para me acompanhar e agradecer nesta etapa.

A todos, meu sincero muito obrigado.

“Isso também vai passar.”
Chico Xavier.

LISTA DE FIGURAS

- Figura 1 - Desenvolvimento de Software Adaptável.
- Figura 2 - Fluxo do Processo *Scrum*.
- Figura 3 – Fluxo da XP.
- Figura 4 – Modelo de Comunicação – *Publish-Subscriber*.
- Figura 5 – *Push* de Mensagem usando GCM.
- Figura 6 - Aplicativo VIVER UFMG.
- Figura 7 – Aplicativo USP.
- Figura 8 – Diagrama de Caso de Uso do Usuário
- Figura 9 - Representação do Menu.
- Figura 10 -Tela Inicial do Aplicativo.
- Figura 11 - Menu de Navegação.
- Figura 12 - Lista dos Eventos Institucionais.
- Figura 13 - Detalhes dos Eventos Institucionais.

Figura 14 - Menu de Navegação após o Teste de Aceitação.

Figura 15 - Detalhes dos Eventos Institucionais após o Teste de Aceitação.

Figura 16 - Página de Cadastro de Eventos Institucionais.

Figura 17 - Página de Cadastro de Notícias.

Figura 18 - Notificação *push* referente ao Cadastro de um Novo Evento.

Figura 19 - Tabelas do Banco de Dados.

Figura 20 - Avaliação da Utilização do Aplicativo.

Figura 21- Avaliação do Profissionalismo do Aplicativo.

Figura 22 - Utilidade do Aplicativo.

Figura 23 - Facilidade do Uso do Aplicativo.

Figura 24- Indicação do Aplicativo.

Figura 25 - Satisfação com Resultado Final.

LISTA DE QUADROS

Quadro 1 – Comparativo entre os Métodos de Desenvolvimento Ágil.

Quadro 2 – Exemplo de vetor do tipo misto utilizando o JSON.

Quadro 3 – Exemplo de vetor do tipo misto utilizando o XML.

Quadro 4 – Comparativo entre Aplicativos Institucionais.

Quadro 5 – CRC referente aos Murais de Notícias.

Quadro 6 - CRC referente aos Eventos Institucionais.

LISTA DE ABREVIATURAS E SIGLAS

ASD – *Adaptive Software Development* (Desenvolvimento de software adaptativo)

CRC – *Class Responsibility Card* (Classe – Responsabilidade - Cartão)

CEDECOM – Centro de Comunicação

DHMCM – Dispositivo Híbrido Móvel de Conexão Multirredes

EAI – *Enterprise Application Integration* (Integração de aplicações empresariais)

GPS – *Global Positioning System* (Sistema de Posicionamento Global)

HTTP – *Hypertext Transfer Protocol* (Protocolo de Transferência de Hipertexto)

JSP – *JavaServer Pages*

KIS – *Keep it simple* (Mantenha a simplicidade)

LAN – *Local-area Networks*

MOM – *Message Oriented Middleware*

MVP – *Minimum Viable Product* (Produto Viável Mínimo)

NUPSI – Núcleo de Prática em Sistemas de Informação

OSI – *Open Systems Interconnection* (Interconexão de Sistemas Abertos)

PROGRAD – Pró-Reitoria de Graduação

PO – *Product Owner* (Dono do Produto)

SD – Sistemas Distribuídos

SGBD – Sistema de Gerenciamento de Banco de Dados

TDD – *Test Driven Development* (Desenvolvimento Orientado por Testes)

UML – *Unified Modeling Language* (Linguagem Unificada de Modelagem)

XML – *eXtensible Markup Language*

XP – *Extreme Programming* (Programação Extrema)

SUMÁRIO

AGRADECIMENTOS	6
1. INTRODUÇÃO	18
1.1. Objetivos.....	20
1.1.1. Objetivo Geral	20
1.1.2. Objetivos Específicos.....	21
1.2. Justificativa.....	22
1.3. Metodologia.....	23
1.4. Organização da Monografia.....	24
2. FUNDAMENTAÇÃO TEÓRICA	25
2.1. Desenvolvimento Ágil.....	25
2.1.1. Programação Extrema.....	31
2.2. Aplicativos Móveis.....	39
2.2.1. Sistemas Distribuídos.....	41
2.2.2. <i>Message Oriented Middleware</i> (MOM).....	46
2.2.3. <i>Web Services</i>	51
2.2.4. Servidor de Aplicação.....	58
2.2.5. Banco de Dados.....	59
2.2.6. Linguagem de Programação.....	61
2.2.7. Produto Viável Mínimo.....	62

3. TRABALHOS RELACIONADOS.....	64
3.1. Aplicativo da Universidade Federal de Minas Gerais....	64
3.2. Aplicativo da Universidade de São Paulo.....	67
3.3. Comparativo entre os aplicativos.....	69
4. CONCEPÇÃO E DESENVOLVIMENTO DO APLICATIVO INSTITUCIONAL DA UEMG FRUTAL	71
4.1. Funcionalidades do Aplicativo.....	71
4.2. Utilização da XP do desenvolvimento do aplicativo.....	77
4.2.1. Planejamento.....	77
4.2.2. Projeto.....	79
4.2.3. Codificação.....	81
4.2.4. Teste.....	82
4.2.4.1. Entrega do primeiro protótipo do aplicativo.....	83
4.2.4.2. Entrega da versão do aplicativo institucional.....	88
4.3. Manipulação dos Dados do Aplicativo.....	91
4.4. Persistência dos dados.....	96
5. AVALIAÇÃO DO APLICATIVO.....	98
6. CONCLUSÃO.....	107

REFERÊNCIAS BIBLIOGRÁFICAS.....	56
APÊNDICE A – Documentação do aplicativo.....	114
APÊNDICE B – Versão inicial do Aplicativo Institucional	131
APÊNDICE C – Estórias de Usuários	137
APÊNDICE D - Aplicativo móvel em sua última versão	140
APÊNDICE E – Telas da Aplicação Web.....	152
APÊNDICE F – Questionário de Avaliação	156

1. INTRODUÇÃO

O *smartphone* pode ser considerado um importante instrumento midiático na atualidade. Este possui diversas funcionalidades como: tocar música, executar jogos, enviar mensagens rápidas (SMS), tirar fotos, fazer vídeos, acessar a internet e também pode ser usado como dispositivo de localização. Os *smartphones* transcenderam o seu objetivo básico inicial, o de realizar apenas a comunicação por voz.

Os *smartphones* estão cada vez mais acessíveis aos usuários e, muitos desses usuários apresentam uma necessidade constante de se manterem informados e conectados. O acesso à informação é facilitado devido à inerente mobilidade dos *smartphones*.

Segundo Lemos (2007), o *smartphone* é um “Dispositivo Híbrido Móvel de Conexão Multirredes” (DHMCM), definido como híbrido porque agrega diversas funções (e.g. telefone,

máquina fotográfica e processador de texto), e como multirredes porque pode empregar diversas redes, dentre elas a rede celular, redes portáteis (e.g. *Bluetooth* e Infravermelho), redes Wi-Fi e GPS (*Global Positioning System* – Sistema de Posicionamento Global).

Os principais sistemas operacionais para *smartphones* são o Android, da Google, o iOS da Apple e o Windows Phone, da Microsoft. O iOS é um sistema considerado “fechado” pois é configurado para apenas hardwares da Apple. Já o Android e o Windows Phone podem ser executados em hardwares de diferentes fabricantes (LECHETA, 2009).

Segundo a INFO (2011), a popularidade do sistema Android está ligada ao grande número de produtos que tem compatibilidade com o mesmo, o que possibilita que os usuários tenham mais opções de escolha na aquisição dos *smartphones*.

As informações na UEMG, unidade de Frutal, não alcançavam seu público alvo de forma eficiente.

Visando a melhoria da comunicação, os profissionais da área de Tecnologia da Informação estão investindo cada vez mais nas plataformas móveis. Assim, no presente trabalho, é apresentado o desenvolvimento de um Aplicativo Institucional, desenvolvido em Android, para a Universidade do Estado de Minas Gerais, unidade de Frutal, com a finalidade de estreitar a comunicação dentro da comunidade acadêmica.

1.1. Objetivos

1.1.1. Objetivo Geral

Este trabalho tem por objetivo propor um canal de comunicação para a Universidade de Estado de Minas Gerais (UEMG), unidade de Frutal, afim de ser usado por representantes da instituição para auxiliar na difusão das informações institucionais relevantes para a comunidade acadêmica.

1.1.2. **Objetivos Específicos**

Este trabalho propôs desenvolver um aplicativo móvel institucional que tivesse a capacidade ser utilizado como:

1. Canal das informações administrativas, como contatos dos principais setores administrativos da unidade;
2. Canal das informações dos coordenadores dos cursos de graduação;
3. Difusor do guia acadêmico da unidade;
4. Canal de divulgação de notícias relacionadas a instituição;
5. Canal de divulgação de eventos universitários;
6. Canal de divulgação do calendário acadêmico;
7. Canal de divulgação de telefones úteis para o meio acadêmico em geral.

1.2. Justificativa

Os *smartphones* têm se tornado comum na vida atual das pessoas e seu potencial tem sido utilizado por diversas empresas privadas e públicas com o objetivo de aumentar sua proximidade com seu público.

Este trabalho se justifica considerando o potencial dos *smartphones* para ampliar o alcance da informação de forma rápida e prática. Além disso, a UEMG, unidade de Frutal, ainda não possui um aplicativo para *smartphones*. O desenvolvimento e implantação de um aplicativo institucional para unidade de Frutal pode auxiliar tanto os setores administrativos quanto a comunidade acadêmica (professores e estudantes), pois o acesso à informação fica facilitado para ambas as partes.

Dados como e-mails institucionais, nomes dos responsáveis pelos departamentos, coordenadores dos cursos, eventos universitários e notícias

acadêmicas estariam disponíveis a todos de forma rápida e eficiente.

1.3. Metodologia

Foi realizado um levantamento bibliográfico sobre o desenvolvimento de aplicativos móveis, sistemas distribuídos e aplicativos institucionais relacionados existentes. Na concepção inicial do aplicativo foi utilizado o conceito de MVP (*Minimum Viable Product* -Produto Viável Mínimo) para gerar uma versão mais simples do produto. O aplicativo foi desenvolvido para a plataforma Android utilizando a metodologia de Programação Extrema (XP) no processo de desenvolvimento.

Como parte do processo de desenvolvimento, foram elaborados histórias de usuários, diagramas de casos de uso e um modelo do banco de dados. Esses artefatos serão detalhados no Capítulo 4.

Ainda durante o processo de desenvolvimento, foram realizadas entrevistas abertas com diversos

representantes da comunidade acadêmica, entre eles: a assessoria de comunicação da unidade, representantes do Diretório Acadêmico, funcionários ligados à Diretoria, alunos e professores. As entrevistas tiveram como objetivo auxiliar no levantamento adequado de requisitos e validar o aplicativo em desenvolvimento.

Ao final do desenvolvimento, o aplicativo foi avaliado, utilizando um questionário, em relação ao uso e aceitação por uma amostra da comunidade acadêmica.

1.4. Organização da Monografia

O trabalho está dividido em 6 capítulos começando com uma introdução sobre o tema apresentado. O Capítulo 1 traz os motivos que justificam a escolha do tema e os objetivos do trabalho. O Capítulo 2 descreve um conjunto de conceitos e tecnologias para a solução do problema em questão. O Capítulo 3 contempla uma comparação entre outros trabalhos

relacionados ao tema. O Capítulo 4 apresenta as etapas e técnicas de desenvolvimento do aplicativo. O Capítulo 5 disponibiliza o questionário que foi utilizado para aferir o uso do aplicativo e, por fim, o Capítulo 6 traz as conclusões finais do trabalho.

2. FUNDAMENTAÇÃO TEÓRICA

Para o desenvolvimento deste trabalho um conjunto de conceitos e tecnologias foram pesquisados e analisados com o objetivo de compor uma solução relacionada à disponibilidade da informação na UEMG Frutal. Esses conceitos são descritos a seguir.

2.1. Desenvolvimento Ágil

Os métodos tradicionais de desenvolvimento de software normalmente não possuem a capacidade de prever como o sistema irá evoluir com o tempo e negligencia as condições do mercado e as necessidades dos clientes, que mudam e se

transformam. Em alguns casos, não são definidos os requisitos antes do início do projeto. Assim, frequentemente os sistemas desenvolvidos extrapolam cronogramas e orçamentos e não atendem satisfatoriamente os requisitos do cliente (PRESSMAN, 2011).

Os métodos de desenvolvimento ágeis surgiram como uma alternativa às formas tradicionais de desenvolvimento de *software*, visando reduzir os custos das mudanças ao longo do processo de formação do *software*, eliminando os gastos com a documentação e com a burocracia (PRESSMAN, 2011).

As metodologias ágeis se destacam por serem adaptativas ao invés de preditivas, são orientadas a pessoas e não a processos e existe uma clara preocupação com os desenvolvedores (FOWLER, 2005).

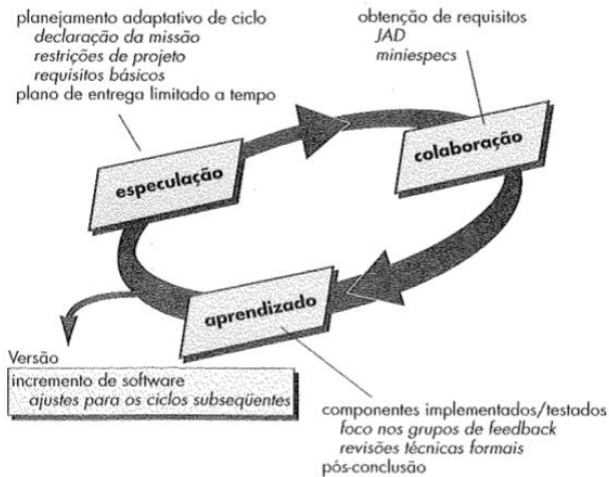
Existem diversos modelos de desenvolvimento ágil, como *Adaptive Software Development* (ASD)

(PRESSMAN, 2011), *Scrum* (PRESSMAN, 2011) e Programação Extrema (XP) (PRESSMAN, 2011).

O ASD é uma metodologia de desenvolvimento de *software* e sistemas complexos proposta por Jim Highmuth. O ASD baseia-se na colaboração humana e na auto-organização das equipes. Essa metodologia possui três fases: especulação, colaboração e aprendizagem, que são definidas como o “ciclo de vida” da ASD, conforme é ilustrado na Figura 1.

A dinâmica das equipes auto organizadas, colaboração interpessoal e a aprendizagem individual são os destaques da ASD, que permitem gerar uma possibilidade de sucesso referente a entrega do produto final (PRESSMAN, 2011).

Figura 1 - Desenvolvimento de Software Adaptável.

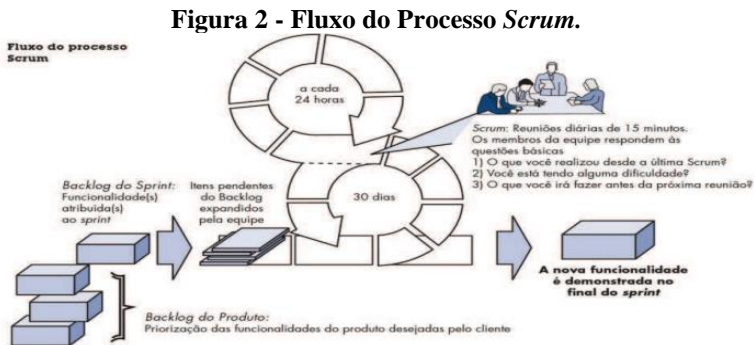


Fonte: Pressman (2011).

O *Scrum* é outro método de desenvolvimento ágil de software usado para conduzir as atividades estruturais como requisitos, análise, projeto, evolução e entrega. Em cada uma das atividades ocorrem tarefas que devem ser realizadas em um padrão chamado *sprint* (Ciclo de trabalho no *Scrum*). A equipe *Scrum* define o número de *sprints*

necessários para realizar o trabalho. Essa quantidade varia de acordo com cada atividade, do tamanho e da complexidade do produto (PRESSMAN, 2011).

Um conjunto de ações de desenvolvimento é incorporado ao *Scrum*, como listagem das prioridades dos requisitos ou funcionalidades do projeto (*Backlog*), urgência de *sprint*, reuniões diárias curtas com perguntas chave respondidas por todos os membros da equipe e entrega de *demos* para avaliação da funcionalidade e apresentação para o cliente (PRESSMAN, 2011). Na Figura 2 é mostrado o fluxo do processo do *Scrum*.



Fonte: Pressman (2011).

No Quadro 1 é apresentada uma comparação entre os métodos de desenvolvimento ágil, visando mostrar suas principais características.

Quadro 1 - Comparativo entre os Métodos de Desenvolvimento Ágil.

Características	ASD	Scrum	XP
Abordagem	Interativo	Incrementos interativos	Incrementos interativos
Tempo de interação	4-8 semanas	2-4 semanas	1-6 semanas
Comunicação	Informal	Informal	Informal
Envolvimento do cliente	Envolvido por meio das <i>releases</i>	Envolvido com o PO (<i>Product Owner</i>)	Envolvido
Documentação	Básica	Básica	Básica
Especialidades	TDD ¹ (<i>Test Driven Development</i>), estórias de usuários, refabricação	<i>Sprint, Backlog, Planning poker, Scrum master</i>	Ciclos de aprendizado

Fonte: o autor (2015).

¹ Técnica para o desenvolvimento de software com pequenos ciclos de repetições. Instiga o código simples e inspira confiança (BECK, 2008).

No presente trabalho foi utilizado o método de desenvolvimento ágil denominado Programação Extrema (XP). A XP possui baixo tempo de interação para desenvolvimento de *software* que sofrem mudanças no processo de desenvolvimento, visa projetar algo simples e é o processo menos burocrático em relação aos demais mencionados acima.

Os outros métodos de desenvolvimento ágil são mais bem aplicados quando se tem uma equipe formada. Sendo este, um trabalho individual, o método que se tornou mais viável para o desenvolvimento do aplicativo foi a XP, mesmo não gerando uma documentação rica. Este método é melhor discutido no item a seguir.

2.1.1. Programação Extrema

Programação Extrema (XP) pode ser definida como uma metodologia leve para equipes de desenvolvimento de pequeno a médio porte que

desenvolvem *softwares* com grandes e constantes modificações (BECK, 2008).

Dentre as práticas que são adotadas na XP, são abortadas as constantes revisões do código, além do aprimoramento dos testes, da busca pela simplicidade do *software* visando a definição e redefinição da arquitetura, bem como testes constantes de integração e, por fim, das interações curtas (BECK, 2008).

A XP é uma forma de desenvolvimento que permite ao cliente receber uma quantidade significativa de resultados da equipe de desenvolvimento. A sua organização gira em torno de um conjunto de valores e práticas que asseguram um retorno do investimento realizado pelo cliente em um *software* (TELES, 2004).

A comunicação entre os desenvolvedores e os clientes é enfatizada, incentivando a colaboração entre ambas as partes por meio de metáforas (estórias dos usuários) para os principais temas,

feedback contínuo e a redução da documentação (PRESSMAN, 2011).

Por intermédio da atividade de ouvir o cliente é criado um conjunto de estórias conhecidas como “estórias dos usuários”, descrevendo os resultados, as características e as funções do software. As estórias são escritas pelo cliente e colocadas em cartões ou fichas (PRESSMAN, 2011).

A XP limita os desenvolvedores a produzir apenas necessidades imediatas, visando obter simplicidade. O objetivo é criar um projeto simples que pode ser implantado em um código (PRESSMAN, 2011).

O *feedback* é uma parte importante na XP, que é dividido em três partes: o próprio software, a parte do cliente e os membros da equipe de desenvolvimento. Assim que as estórias de usuários ou casos de uso são entregues pelos clientes, as mesmas são utilizadas como base para a realização de testes de aceitação. A interação da equipe com o cliente faz com que se tenha um *feedback* rápido, diminuindo o

impacto no custo e no cronograma (PRESSMAN, 2011).

A programação orientada a objetos é padrão de desenvolvimento na XP, que envolve ainda um conjunto de regras e práticas de quatro atividades metodológicas: planejamento, projeto, codificação e teste (PRESSMAN, 2011).

O levantamento de requisitos começa com o planejamento. Nesse momento é realizada a atividade de “ouvir”, ou seja, entender o ambiente comercial do *software* para poder compreender os principais fatores e funcionalidades (PRESSMAN, 2011).

A criação das estórias dos usuários tem início com a descrição das funcionalidades, das características e dos resultados esperados do *software* em desenvolvimento. As estórias são separadas e colocadas em fichas, em que o cliente atribui um respectivo valor (prioridade). Assim, uma avaliação é feita pela equipe, relacionando os custos em semanas de desenvolvimento. Quando a primeira versão do *software* é desenvolvida, mede-se a velocidade do

projeto por meio da quantidade de estórias de usuários que foram implementadas. No decorrer do projeto, as estórias podem ser acrescentadas, modificadas e eliminadas.

A XP tem como premissa manter sempre a simplicidade do projeto, tendo como base o princípio KIS (*Keep It Simple* - Mantenha a simplicidade) (PRESSMAN, 2011).

Uma forma de romper o modo processual do pensamento é a utilização de CRCs (*Class, Responsibilities and Collaboration Cards* - Cartões de Classe, Responsabilidade e Colaboração), pois permite às equipes pensarem em cenários orientados a objetos. Os cartões CRC organizam as classes, descrevem suas responsabilidades e também as demais classes às quais essas podem interagir. Os cartões são feitos de forma colaborativa pela equipe e são utilizados para realizar incrementos do *software* (PRESSMAN, 2011).

Depois de construídas as estórias dos usuários para a elaboração do projeto, inicia-se a etapa de

codificação, cuja palavra-chave é a programação em pares. Isso indica que duas pessoas devem trabalhar juntas em uma mesma estação de trabalho, fazendo com que os possíveis problemas sejam rapidamente solucionados (PRESSMAN, 2011).

Na programação em pares ocorre uma divisão de responsabilidade entre a dupla: um foca nos detalhes de codificação de determinada parte do projeto e outro verifica se todos os padrões estão sendo seguidos. A vantagem é que os desenvolvedores estão sempre focados no problema em questão. No presente trabalho não foi utilizada a técnica de programação em pares, pois a codificação foi feita por apenas um desenvolvedor.

Existe uma técnica na XP que ocorre quando um problema difícil surge no projeto, é a chamada refabricação, em que o *software* passa por um processo de alteração visando a melhoria do código, mas sem interferir na estrutura interna e modificando apenas o código interno para o seu aperfeiçoamento (PRESSMAN, 2011).

Os testes de aceitação fecham um ciclo da XP. São também chamados de testes de clientes. Eles são responsáveis por aprovar as estórias de usuários implantadas como parte da próxima versão do *software* e verificam se as características e as funcionalidades são viáveis, e como devem ser revisadas (PRESSMAN, 2011).

A Figura 3 exhibe o fluxo da XP explicado anteriormente.

Figura 3 – Fluxo da XP.



Fonte: Pressman (2011).

A XP aponta duas questões importantes: os programadores poderão trabalhar naquilo que realmente interessa e não enfrentarão as dificuldades sozinhos, podendo assim, tomar as decisões para as quais estiverem preparados e não se manifestarão em decisões que não são qualificados (BECK, 2008).

Em um *software*, normalmente apenas 20% das funcionalidades totais são frequentemente utilizadas, e isso equivale a cerca de 80% do valor geral do *software* para o usuário. Cerca de 64% das funcionalidades raramente são utilizadas pelo usuário, gerando um desperdício no tempo de desenvolvimento. A XP atenta para desenvolver algo simples e que atenda às reais necessidades do cliente (TELLES, 2004).

Toda a equipe, gerencial e operacional deve se comprometer em obter o maior valor possível de cada semana de programação. Assim, a XP diminui os riscos dos projetos, otimiza as respostas às mudanças no negócio e aumenta a produtividade (BECK, 2008).

A implantação da XP neste trabalho é detalhada no Capítulo 4, em que são mostradas as estórias dos usuários e as técnicas apresentadas nesta metodologia de desenvolvimento.

2.2. Aplicativos Móveis

O desenvolvimento dos *smartphones* ocorreu de forma paralela ao crescimento dos aplicativos móveis. Diversos tipos de aplicativos surgem todos os dias visando melhorar o cotidiano do usuário. A possibilidade de acesso à informação a qualquer momento e em qualquer lugar aumentou a quantidade de novos aplicativos e serviços (NAKAMURA, 2015).

Uma vantagem dos aplicativos móveis em relação aos *softwares* para *desktop* é a proximidade do usuário. A consulta é realizada de forma fácil e rápida. Outra vantagem é que as aplicativos móveis levam as informações ao usuário de forma proativa,

ou seja, ocorre sem que o mesmo faça algum tipo de solicitação direta (NAKAMURA, 2015).

Abaixo são exemplificadas algumas áreas que já foram exploradas em relação aos aplicativos:

- **Aplicativos Pessoais:**

Acesso à informação, organização de contatos e transferência de arquivos.

Exemplo: *WhatsApp Messenger*² – Aplicativo de mensagens instantâneas.

- **Aplicativos Corporativos:**

Acesso remoto às informações corporativas e apoio à tomada de decisão.

Exemplo: *Fluig*³, da TOTV SA – plataforma que une todos os instrumentos de trabalho da empresa.

² Disponível em:

<<https://play.google.com/store/apps/details?id=com.whatsapp&hl=pt-BR>>. Acesso em 29 de nov. 2015.

³ Disponível em:

<<https://play.google.com/store/apps/details?id=br.com.fluig&hl=pt-BR>>. Acesso em 29 de nov. 2015

- **Aplicativos Financeiras:**

Transações eletrônicas e operações bancárias.

Exemplo: Banco do Brasil⁴ – acesso às diversas funcionalidades oferecidas pelo Banco do Brasil.

O desenvolvimento de aplicativos móveis envolve o entendimento de diversos processos e tecnologias como sistemas distribuídos, MOM⁵ (*Message Oriented Middleware*), *web services* (Serviços Web), servidor de aplicação e linguagem de programação. Esses itens são detalhados a seguir.

2.2.1. Sistemas Distribuídos

Os Sistemas Distribuídos (SD) tiveram seu início pela formação de Redes Locais de Computadores (LAN - *LocalArea Networks*) de alta velocidade, em que as informações e os recursos podem ser

⁴ Disponível em: <

<https://play.google.com/store/apps/details?id=br.com.bb.android&hl=pt-BR> >. Acesso em 29 de nov. 2015.

⁵ Termo explicado na seção 2.2.2 na página 27.

compartilhados pelas máquinas. Define-se um sistema distribuído como um conjunto de máquinas independentes que se apresentam para os usuários finais como um sistema único e coerente (TANENBAUM, 2008).

O acesso aos recursos remotos e aos compartilhamentos de forma controlada, consistente e uniforme são algumas das premissas do SD, pois visa facilitar a vida do usuário e da aplicação, independentemente de onde a interação esteja ocorrendo. Exemplos de recursos compartilhados são impressoras, computadores, dados e páginas Web (TANENBAUM, 2008). Neste contexto, este trabalho realiza o compartilhamento de dados e de páginas Web.

Uma vantagem do SD é a capacidade de ocultar que os processos e recursos se encontram fisicamente separados em outros computadores. Transparência é como se conceitua essa característica (TANENBAUM, 2008).

Alguns tipos de transparência podem ser aplicados no SD, tais como:

- Ocultar a diferenciação de dados e no modo de acesso;
- Ocultar o local onde o recurso está localizado;
- Ocultar mudança de localização;
- Ocultar que o recurso é replicado;
- Ocultar que o recurso pode ser compartilhado por vários usuários;
- Ocultar falha e a recuperação de um recurso.

Os SD podem ser classificados em três tipos: sistemas de computação distribuídos, sistemas embutidos distribuídos e sistemas de informação distribuídos.

Os sistemas de computação distribuídos são utilizados em tarefas que exigem alto desempenho, como a computação em *cluster* (conjunto de computadores com rede local de alta velocidade em que cada nó possui o mesmo sistema operacional) e computação em grade (conjunto de computadores

montados, em que determinados conjuntos de nós representam domínios administrativos diferentes) (TANENBAUM, 2008).

Os sistemas embutidos distribuídos são considerados estáveis por terem nós fixos, ausência de controle administrativo humano e por serem pré configurados. Se encaixam da forma mais eficiente possível no ambiente que residem (TANENBAUM, 2008).

Neste trabalho, o tipo de SD utilizado é o sistema de informação distribuído, mais bem explicado abaixo.

A separação das aplicações em componentes distribuídos criou a necessidade de realizar a integração para que existisse uma comunicação direta entre todas as aplicações de um dado contexto. Esse procedimento tornou-se comum. Surgiu então um segmento da indústria destinada apenas à integração de aplicações empresariais chamadas EAI (*Enterprise Application Integration*) (TANENBAUM, 2008).

Para a realização de transações em SDs, são necessárias conceitos especiais. A determinação do escopo de uma transação pode ser feita utilizando, por exemplo, os termos-chave `BEGIN_TRANSACTION` e `END_TRANSACTION`. Entre esses dois é formado o conteúdo da transação. As transações, por sua vez, são divididas em quatro propriedades fundamentais: atomicidade, consistência, isolamento e durabilidade, habitualmente chamado de ACID (TANENBAUM, 2008).

A atomicidade é definida como uma propriedade que assegura às transações completarem todas as suas operações; caso alguma dessas não ocorra, então nenhuma operação é realizada. A consistência garante que a transação respeite todas as normas de integridade configuradas no sistema (e.g. relacionamento entre chaves de tabelas de um banco de dados), para que não exista nenhuma divergência no instante da transação. O isolamento significa que, se duas ou mais transações forem executadas em um

mesmo momento, uma não deverá interferir na outra. Por fim, a durabilidade remete ao fato de que não importa o que aconteça no sistema, quando se finaliza uma transação, seus resultados serão permanentes. Nenhuma falha após a finalização poderá resultar na perda da transação (TANENBAUM, 2008).

2.2.2. *Message Oriented Middleware* (MOM)

Middleware é um mediador entre dois programas existentes e independentes. O seu objetivo é realizar a independência das aplicações com o sistema de transmissão (BERNSTEIN, 1996).

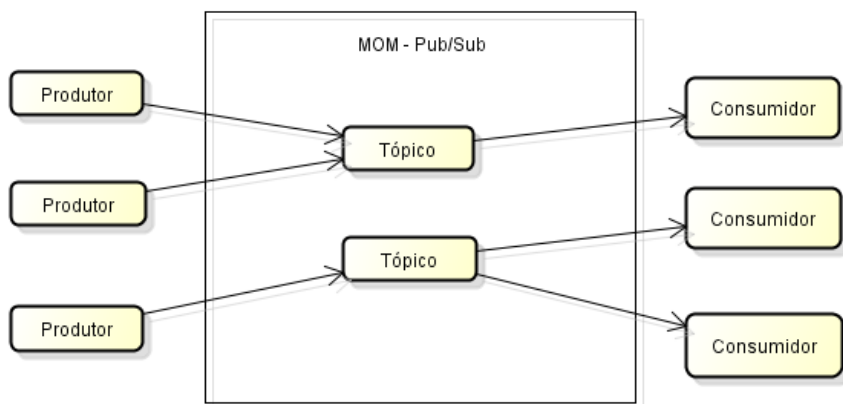
Uma dificuldade das aplicações é a interação entre si, pois normalmente se encontram em domínios administrativos diferentes. Pela falta de comunicação e de um domínio não ter controle sobre o outro, surgiu na necessidade da EAI desenvolver os MOMs (*Message Oriented Middleware*).

Existem diversas tecnologias que tentam solucionar a questão da integração dos aplicativos, como *web services*, *Enterprise Service Bus* (ESB), *Service Oriented Architecture* (SOA), entre outros.

A vantagem do MOM é a troca de mensagens desacopladas, não sendo necessário, por exemplo, que o consumidor e produtor das mensagens tenham consciência um do outro. Também não exige que ambos estejam disponíveis isocronicamente.

A Figura 4 é um exemplo da comunicação do tipo *Publish-Subscriber*, comumente utilizado em MOMs, em que as mensagens enviadas podem ser recebidas por uma quantidade ilimitada de consumidores (*Subscriber*). Esse modelo se baseia em canais de comunicação chamado de tópicos (*Topic*). As aplicações produtoras se registram nos tópicos a fim de enviar mensagens. Já as aplicações consumidoras se registram para recebê-las.

Figura 4 – Modelo de Comunicação – *Publish-Subscriber*.



Fonte: o autor (2015).

Um serviço atual que utiliza conceitos de MOM é o *Google Cloud Messaging* (GCM)⁶, que é um serviço fornecido pela Google para o servidor de mensagens *downstream* e *upstream*. Esses são utilizados para processamentos em segundo plano e estabelecem uma conexão segura entre o servidor e

⁶ Disponível em: <<https://developers.google.com/cloud-messaging/>>. Acesso em: 02 de dez. de 2015.

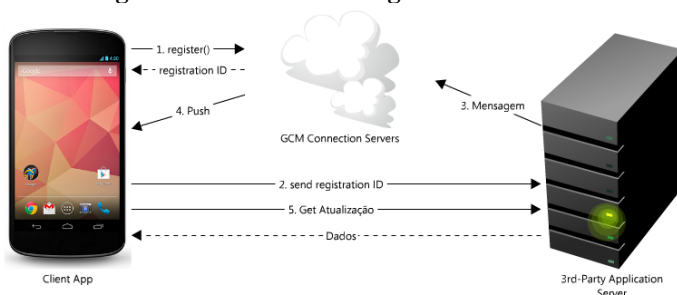
os *smartphones*. O serviço tem como finalidade alertar os usuários com mensagens de chat ou notificações.

O usuário não necessita estar ativo para receber a informação.

Na Figura 5 é exibido em detalhes o funcionamento do GCM para realização de uma mensagem.

Na etapa 1, *register()*, ocorre o registro do *smartphone*. Esse método retorna o *registration ID*, que é passado para o servidor da aplicação para que posteriormente possam ser enviadas mensagens aos destinatários cadastrados.

Figura 5 – Push de Mensagem usando GCM.



Fonte: Site DevMedia⁷.

O fluxo de envio de mensagem é considerado “servidor > aplicação”. O servidor de aplicação envia uma mensagem para o servidor do GCM. As mensagens são colocadas em uma fila de envio, sendo essa ação responsabilidade do GCM.

O aplicativo não precisa estar *online*. Caso esteja *off-line* é feita uma réplica local dos dados das mensagens e o GCM envia a mensagem assim que o *smartphone* estiver *online*. A forma de recebimento

⁷ Disponível em: <<http://www.devmedia.com.br/google-cloud-messaging-introducao/29776>>. Acesso em: 12 ago. 2015.

da mensagem também considera o “servidor > aplicação”.

A mensagem enviada para um determinado aplicativo é recebida por meio de um serviço que fica constantemente verificando se existe alguma modificação, chamado de *BroadcastReceiver*. O *framework* do GCM é responsável por tratar as mensagens que são envidas para o *smartphone*.

Essa tecnologia veio para substituir a técnica de *Polling*, na qual o *smartphone* fica constantemente verificando se existe alguma atualização no servidor, aumentando significativamente o consumo de recursos. O uso do GCM permite a redução do processamento e do consumo de memória, rede de dados e de bateria.

2.2.3. *Web Services*

A Arquitetura de *Web Services* é utilizada para criar a possibilidade de realizar interoperabilidade

entre aplicações que rodam em diferentes plataformas. (COULOURIS, 2008).

A aplicação tem sua linguagem original traduzida para uma linguagem universal, um formato intermediário como XML (*eXtensible Markup Language*), JSON (*JavaScript Object Notation*), CSV (*Comma-separated values*), entre outros (COULOURIS, 2008).

Atualmente as empresas visam otimizar seus processos e melhorar a troca de informações dentro e fora da organização. A comunicação entre sistemas é dinâmica, segura e não necessita de supervisão humana (COULOURIS, 2008).

Um conjunto de operações que pode ser usada por um cliente na internet pode ser entendido como uma interface do serviço web, podendo essa ser fornecida por diversos recursos (COULOURIS, 2008).

O acesso às operações é realizado utilizando requisições e respostas formatadas em XML, JSON ou outro formato, podendo ser transmitidas pelo

Protocolo de Transferência de Hipertexto (HTTP - *Hypertext Transfer Protocol*).

Uma das principais características dos serviços web é o processamento de mensagens SOAP (*Simple Object Access Protocol*) formatadas em XML. O protocolo SOAP foi criado inicialmente para possibilitar a chamada remota de métodos em uma interação cliente-servidor pela Internet. Esse protocolo escolheu o uso do XML por sua segurança, pois os documentos, ou partes deles, podem ser assinados, gerenciados por chaves de segurança e criptografia (COULOURIS, 2008).

O XML é recomendado pela W3C⁸ (*World Wide Web Consortium*), responsável pela padronização da *World Wide Web* para anotação de documentos. Algumas das vantagens, como já dito, são a flexibilidade, robustez e relativa simplicidade, porém, para algumas situações mais simples, o XML pode se tornar inviável, abrindo espaço para outras tecnologias (RICKARDSON e RUBY, 2007).

⁸ Disponível em: < <http://www.w3.org/> >. Acesso em: 07 jul. 2015.

Uma estratégia alternativa ao SOAP é a tecnologia REST (*Representational State Transfer* – Transferência Representacional de Estado) que se refere à junção de várias arquiteturas, como XML, HTTP e JSON, visando descrever qualquer interface web. O REST destaca-se na manipulação dos dados ao invés das interfaces (RICKARDSON e RUBY, 2007).

As premissas dos REST estão detalhadas abaixo:

- O uso do HTTP, pois define um conjunto de operadores POST, GET, PUT e DELETE que são relacionadas ao CRUD (*Create, Read, Update e Delete*) para a consistência dos dados;
- Uma mensagem HTTP possui toda informação necessária para a envio do pedido;
- Os recursos são identificados por meio URI (*Uniform Resource Identifier* – Identificador Uniforme de Recursos), normalmente

utilizando URL (*Uniform Resource Locator* – Localizador Uniforme de Recursos);

Conforme citado acima, uma das linguagens utilizadas pelo REST é JSON. Essa foi a linguagem utilizada no desenvolvimento do aplicativo aqui proposto no que diz respeito à estruturação dos dados entre servidor e aplicativo móvel (RICKARDSON e RUBY, 2007).

O JSON é um formato de texto para realizar a serialização de dados estruturados. Pode representar quatro tipos primários (e.g. *string*, números, booleanos e nulos) e dois tipos estruturados (objetos e vetores). Essa tecnologia foi criada com o objetivo de ser simples, portátil, textual e direcionada a *JavaScript* (linguagem de programação *client-side* para navegadores web) (RICKARDSON e RUBY, 2007).

O JSON é geralmente produzido por serviços que esperam ser consumidos pela parte cliente de aplicação. Para o navegador é mais simples obter uma estrutura de dados *JavaScript* com uma estrutura

de dados JSON, do que em um documento XML (RICKARDSON e RUBY, 2007).

Na utilização de requisições na Web, o JSON é um recurso eficiente, pois pode transportar um objeto serializado e transformá-lo em um objeto *JavaScript*. O JSON também está sendo cada vez mais utilizado para transporte de informações e está incluído nos principais navegadores da internet existentes, permitindo construção de aplicações Web interativas (RICKARDSON e RUBY, 2007).

Os Quadros 2 e 3 exemplificam como um vetor de tipo misto pode ser representado em JSON (Quadro 2) e em XML (Quadro 3). Pode-se notar que a notação da JSON é um pouco mais enxuta que a da XML.

Quadro 2 – Exemplo de vetor do tipo misto utilizando o JSON.

```
[3, "Exemplo JSON"]
```

Fonte: o autor (2015).

Quadro 3 – Exemplo de vetor do tipo misto utilizando o XML.

```
<value>
  <array>
    <data>
      <value><i4>3</i4></value>
      <value><string>Exemplo XML</stri
    </data>
  </array>
</value>
```

Fonte: o autor (2015).

A notação de objetos JSON possui uma formatação leve para troca de dados, fácil de ler e escrever. Por ser leve é muito utilizado no desenvolvimento de aplicativos móveis para realizar a comunicação entre os dados (FONSECA e SIMOES, 2007).

No desenvolvimento de aplicativos web a utilização de *servlet* e JSP (*JavaServer Pages*) é uma combinação poderosa.

As *Servlets* são classes Java com estruturas bem definidas instaladas junto a um servidor que implementa um *Servlet Container* (Servidor que permite a execução de *Servlets* normalmente chamado de Servidor de Aplicações Java) e possibilita adicionar conteúdo dinâmico no servidor web (DEITEL, 2005).

O JSP é uma tecnologia utilizada para tornar dinâmico o conteúdo das páginas web e é como uma extensão do *servlet*. O JSP vem para complementar a parte visual e o *servlet* funciona como um controlador interno (DEITEL, 2005).

2.2.4. Servidor de Aplicação

Um servidor de aplicação é responsável por disponibilizar um local para instalação e execução de

aplicações, também conhecido como *software de middleware* (DEITEL, 2005).

Neste trabalho foi desenvolvido uma aplicação servidora utilizando *Eclipse Java EE IDE for Web Developers*, que facilitou a criação de toda a infraestrutura necessária para a mesma. Como este projeto tem um objetivo funcional e visa integrar sistemas diferentes que rodam em arquiteturas distintas (a aplicação web e o aplicativo móvel), foi utilizado *web services* sobre um *container GlassFish* versão 4.0 *open-source*, que suporta todas as especificações da API Java EE (SAMPAIO, 2007).

2.2.5. Banco de Dados

O banco de dados tem como definição um conjunto dados ou registros relacionados e um conjunto de sistemas para fornecer acesso a esses dados (GAMA e SCHEER, 2004).

Como objetivo principal, o banco de dados permite a um determinado sistema o armazenamento e recuperação das informações, facilitando o acesso

para os interessados à informação (GAMA e SCHEER, 2004).

Os bancos de dados são criados e mantidos por um conjunto de *softwares* chamados Sistemas de Gerenciamento de Banco de Dados (SGBD). Conforme explícito no nome, tem como o objetivo gerenciar o banco de dados, auxiliando nas especificações do tipo de dados, nas restrições, construção, armazenamento de mídias, entre outros (TESSARO, 2013).

O modelo mais aplicado para elaboração de banco de dados é o modelo relacional, sendo esse a relação de uma tabela com outra (TESSARO, 2013).

O SGBD tem algumas funções básicas para a gestão do banco de dados e essas incluem manter a integridade, restrições, segurança, restauração e eficiência do mesmo (GEREMIA, 2010).

Existem diversos tipos de SGBD, por exemplo: *PostgreSQL*, *FireBird*, *Oracle*, *SQL Server*, *MySQL*. Neste trabalho foi escolhido *MySQL* para gestão do banco de dados. O *MySQL*,

por ter licença gratuita, código aberto, capacidade de processar e gerenciar uma grande quantidade de informações, se tornou um dos SGBD mais populares atualmente. Tem disponibilidade de bibliotecas compatíveis com diversas linguagens, tais como: C, *Python*, *Microsoft .NET*, JAVA; e também possui compatibilidade com os sistemas operacionais *Windows*, *Linux* e *Mac OS*.

2.2.6. Linguagem de Programação

Java foi a linguagem de programação escolhida para o desenvolvimento do aplicativo proposto neste projeto pelo fato do grande número dos *smartphones* oferecerem suporte à sua plataforma. Além disso, Java é uma linguagem robusta e escalável (SAMPAIO, 2007).

O desenvolvimento do aplicativo móvel foi realizado por meio da IDE (*Integrated Development Environment*) do Android Studio 1.1.0, projetado

especificamente para o desenvolvimento de aplicações Android.

2.2.7. Produto Viável Mínimo

O desenvolvimento do Produto Viável Mínimo (MVP - *Minimum Viable Product*) vem para auxiliar os empreendedores na elaboração de um novo produto. É definido como uma versão de um produto para teste. É um marco crucial no ciclo de vida de um empreendimento, pois auxilia o começo do processo de aprendizagem de forma mais rápida (RIES, 2011).

O MVP é considerado um testador de hipóteses do negócio, visa iniciar o processo de aprendizagem do produto, mas sem finalizá-lo. Dessa forma, se contrapõe ao desenvolvimento tradicional que envolve longo período de aperfeiçoamento do produto (RIES, 2011).

O MVP visa o menor tamanho possível do produto para que se possa finalizá-lo em período

curto, com um valor de produto proporcional para que o cliente faça a utilização desse produto. O MVP deve ter funcionalidades que tornem o produto coeso e útil (RIES, 2011).

O conceito MVP é muito utilizado nas empresas de *startup*. Essas empresas são de pequeno porte, recém-criadas, em fase de desenvolvimento e de pesquisa de mercado (RIES, 2011).

3. TRABALHOS RELACIONADOS

Devido à popularidade dos aplicativos móveis, diversas instituições de ensino utilizam essa tecnologia para a melhoria na divulgação da informação interna.

Neste capítulo são analisados dois aplicativos institucionais visando suas principais características. Ambos foram utilizados como base para o desenvolvimento do aplicativo institucional para Universidade do Estado de Minas Gerais, unidade de Frutal.

3.1. Aplicativo da Universidade Federal de Minas Gerais

O aplicativo Viver UFMG foi desenvolvido por estudantes dos cursos de Engenharia de Controle e Automação (Lucas Pires) e de Engenharia de Sistema (Alexandre Ribeiro e Daniel Durães) sobre a coordenação da Prograd (Pró-Reitoria de Graduação)

e com o apoio do CEDECOM (Centro de Comunicação)⁹.

A ferramenta facilita o acesso à informação e aos serviços da UFMG, disponibiliza e-mails de contatos dos principais órgãos administrativos da universidade, calendário acadêmico, mapas do campus, horários de ônibus internos e informações relacionado à vida acadêmica, como suporte nas matrículas, planos de estudos, assistência estudantil e normas acadêmicas.

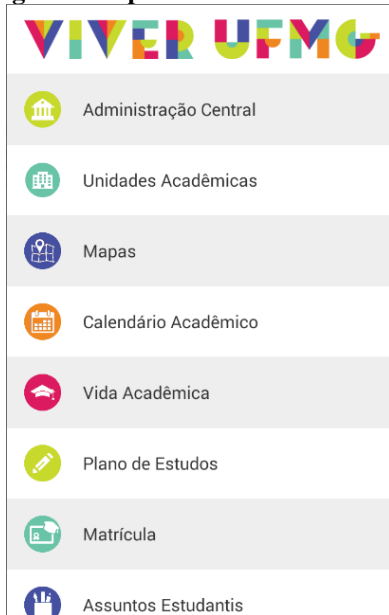
O aplicativo Viver UFMG tem como principal característica a divulgação dos dados institucionais da universidade, visando facilitar o dia-a-dia das pessoas que fazem parte do campus da UFMG.

Na Figura 6 é exibido o *layout* principal do aplicativo Viver UFMG.

⁹ Disponível em:

<https://play.google.com/store/apps/details?id=com.alelu.ufmguia&hl=pt_BR>. Acesso em: 29 de nov. 2015.

Figura 6 - Aplicativo VIVER UFMG.



Fonte: Google Play Store¹⁰.

¹⁰ Disponível em

<<https://play.google.com/store/apps/details?id=com.alelu.ufmguia&hl=pt-PT>>. Acesso em: 14 fev. 2015.

3.2. Aplicativo da Universidade de São Paulo

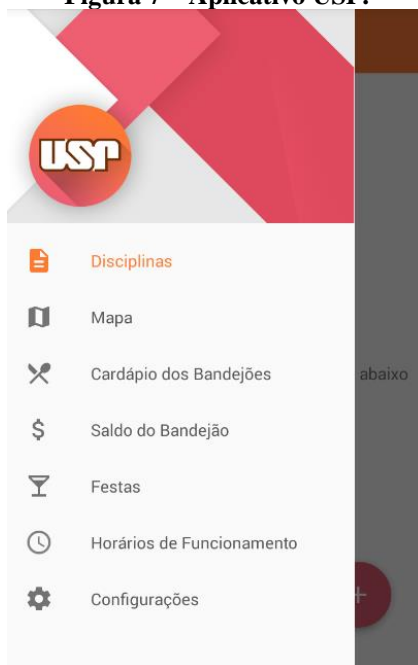
O aplicativo da USP (Universidade de São Paulo) foi desenvolvido pelo estudante Douglas Iacovelli.

A ferramenta fornece informações aos estudantes, disponibilizando o cardápio e o saldo do crédito dos restaurantes universitários, o mapa dos ônibus, a listagem das festas e os horários de funcionamentos dos portões. Fornece ainda, função de controle de faltas e detalhes sobre as aulas de cada aluno.

O aplicativo da USP tem como principal característica fornecer um meio que facilite a vida acadêmica do discente, por intermédio de informações que possam diminuir o estresse do dia-a-dia.

A Figura 7 exhibe o *layout* principal do aplicativo da USP.

Figura 7 – Aplicativo USP.



Fonte: Google Play Store¹¹.

¹¹ Disponível em:

<<https://play.google.com/store/apps/details?id=com.iacovelli.usp&hl=pt-PT>>. Acesso em: 14 de fev. 2015.

3.3. Comparativo entre os aplicativos

Abaixo é feito um breve comparativo entre os dois aplicativos citados:

- **Viver UFMG:**
Visa a divulgação das informações administrativas e amparo na vida acadêmica do aluno dentro da instituição;
- **USP:**
Visa divulgar informações não instrucionais como festas, cardápio, pontos de ônibus e não traz questões administrativas da instituição.

O aplicativo da Universidade do Estado de Minas Gerais teve como proposta unir essas duas vertentes, ou seja, disponibilizar tanto informações administrativas quanto informações para o cotidiano daqueles que utilizam o espaço da unidade de Frutal.

No Quadro 4 é exibido um comparativo entre os três aplicativos institucionais, mostrando suas funcionalidades de modo geral.

Quadro 4- Comparativo entre Aplicativos Institucionais.

Funcionalidades	UFMG	USP	UEMG
Calendário Acadêmico	X		X
Restaurante Universitário		X	
Divulgação Eventos		X	X
Divulgação Notícias			X
Mapa da Instituição	X	X	
Normas da Instituição	X		X
Vida Acadêmica	X		X
Telefones Úteis	X		X

Fonte: o autor (2015).

4. CONCEPÇÃO E DESENVOLVIMENTO DO APLICATIVO INSTITUCIONAL DA UEMG FRUTAL

Este capítulo descreve em detalhes o desenvolvimento do aplicativo institucional para Universidade do Estado de Minas Gerais, unidade de Frutal.

4.1. Funcionalidades do Aplicativo

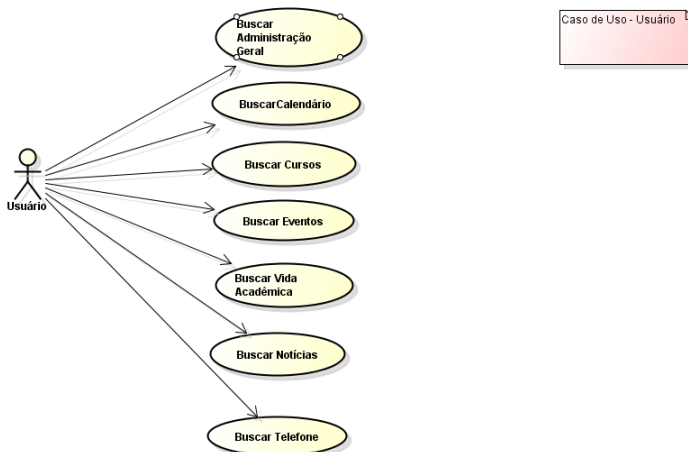
Nesta seção são apresentadas as funcionalidades disponíveis no aplicativo Institucional da Universidade do Estado de Minas Gerais, unidade de Frutal.

Visando uma melhoria na documentação do aplicativo foi elaborado o diagrama de caso de uso da

UML (*Unified Modeling Language* - Linguagem Unificada de Modelagem)¹².

A Figura 8 exibe a interação do usuário com as funcionalidades do aplicativo. A documentação completa pode ser encontrada no Apêndice A.

Figura 8 – Diagrama de Caso de Uso do Usuário

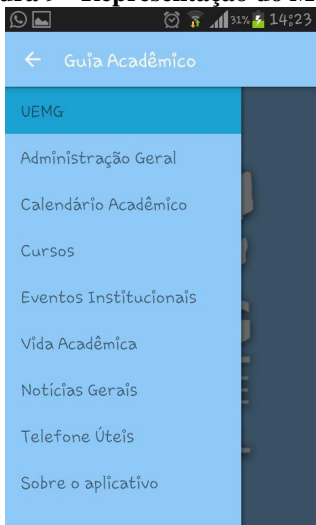


Fonte: o autor (2015).

¹² Linguagem Unificada de Modelagem é uma linguagem padrão para modelagem orientada a objetos. Fonte: GUEDES G. T. A. UML 2: Uma Abordagem Prática. [s.l. :s.n.], 2009. p. 488.

O aplicativo possui um menu de opções em que o usuário poderá selecionar o item que for do seu interesse. O menu foi separado em Administração Geral, Calendário Acadêmico, Cursos, Eventos Institucionais, Vida Acadêmica, Notícias Gerais, Telefones Úteis e Sobre. A Figura 9 exibe o menu citado acima.

Figura 9 - Representação do Menu.



Fonte: o autor (2015).

A primeira opção, Administração Geral, tem como objetivo trazer para os usuários informações referentes aos setores administrativos da Unidade. Contém informações como nome do local, responsável e e-mail de contato. As informações estão dadas em:

- Diretoria;
- Departamentos;
- Setores;
- Representação estudantil;

A segunda opção, Calendário Acadêmico, traz o calendário oficial da Universidade, com datas dos feriados, dias letivos e reposições aos sábados.

A terceira opção, Cursos, tem o objetivo de divulgar os nomes e e-mail dos sete coordenadores dos cursos de graduação da unidade. Essa informação visa facilitar o contato com os representantes de cada curso.

A quarta opção, Eventos Institucionais, traz para os usuários uma lista de todos os eventos

relacionados à Universidade, detalhando o local do evento, data e hora de início e descrição de informações importantes. Essa opção facilita a integração dos eventos da Unidade para toda a comunidade acadêmica.

A quinta opção, Vida Acadêmica, tem como objetivo trazer para o usuário diversas informações sobre as normas administrativas da Universidade. Está dividida nos seguintes itens:

- **Biblioteca:**
Contempla horário de funcionamento, dados necessários para solicitar o cartão da biblioteca, prazos de entrega, multas, renovação e reservas de livros;
- **Matrícula:**
Contempla dados necessários para realizar, renovar e trancar a matrícula do discente;
- **Cartazes:**
Contempla regras para fixação de cartazes;
- **Colação de Grau:**

Contempla o funcionamento e a emissão de diplomas referente à colação de grau;

- **Avaliações:**

Contempla informações sobre provas de segunda chamada e reprovação;

- **Regime Especial:**

Contempla explicações sobre o funcionamento do regime especial;

- **Taxas de Serviços:**

Contempla os valores referentes às taxas administrativas;

- **Transferência:**

Contempla informações sobre mudança de turno e turma, e aproveitamento de estudos.

A sexta opção, Notícias Gerais, traz todas as notícias que possam interessar à comunidade acadêmica tais como: editais de convocação, chamadas públicas, entre outras notícias que a assessoria de comunicação julgar importante.

A oitava opção, Telefone Úteis, tem como objetivo listar diversos telefones da cidade de Frutal

para que os usuários tenham essa informação de forma rápida.

A nona opção, Sobre, traz os créditos ao desenvolvedor e orientador do projeto.

4.2. Utilização da XP do desenvolvimento do aplicativo

Conforme explanado no Capítulo 2, o processo de desenvolvimento do aplicativo foi realizado utilizando metodologia XP. As principais etapas da XP são: o planejamento, o projeto, a codificação e os testes. Essas etapas são mais bem detalhadas abaixo.

4.2.1. Planejamento

O desenvolvimento em XP utiliza o diálogo presencial com o cliente para obter a maior quantidade de detalhes possíveis para cada estória, tendo como objetivo garantir que os envolvidos no

projeto tenham a chance de poder expressar suas ideias de forma clara.

Uma pessoa ou grupo com participação, ações ou investimentos em um determinado processo são considerados as partes interessadas nos resultados do projeto e são chamados de *stakeholders*.

No presente trabalho foram determinados alguns *stakeholders* para a elaboração das estórias dos usuários. Os selecionados foram a assessora de comunicação da Unidade, um docente do curso de Sistemas de Informação e um discente de cada curso de graduação para que fossem elaborados os cenários.

O diferencial da XP é a constante integração com o cliente no andamento do projeto.

Ao finalizar a criação das estórias iniciais foi gerando um protótipo de média fidelidade desenvolvido no Android Studio 1.1.0 que pode ser visto no Apêndice B.

Esse protótipo foi apresentado aos *stakeholders* afim de obter um *feedback* rápido e preciso sobre as estórias inseridas nesse protótipo.

4.2.2. Projeto

As estórias dos usuários foram criadas por meio de conversas informais e foram registradas em CRC (Classe – Responsabilidade - Cartão) com atribuição de prioridades para implementação e mantendo sempre a simplicidade, princípio da XP.

Nos Quadros 5 e 6 são exibidos alguns dos CRCs do projeto. As estórias iniciais dos usuários podem ser encontradas no Apêndice C.

Foi feito o agrupamento das estórias para nortear o desenvolvimento da versão inicial do aplicativo. Após o agrupamento, foram atribuídos valores às estórias para ordenação de prioridades, variando de um (baixa prioridade) à quatro (alta prioridade).

Quadro 5 – CRC referente aos Murais de Notícias.

Mural de notícias – Prioridade: 3	
Responsabilidade	Colaboração
Informações relevantes para os discentes, docentes e funcionários.	Docente
Editais para intercâmbio, projetos de extensão.	Assessoria

Fonte: o autor (2015).

Quadro 6 - CRC referente aos Eventos Institucionais.

Eventos Institucionais– Prioridade: 4	
Responsabilidade	Colaboração
Informações dos eventos que envolvem os docentes, como eventos universitários.	Docente

Fonte: o autor (2015).

Uma técnica utilizada nesta etapa da XP e empregada neste trabalho foi a refabricação. No

protótipo do aplicativo o registro do ID do *smartphone* do usuário era feito apenas ao clicar na opção dos Eventos Institucionais. Caso essa opção não fosse aberta, o ID não era cadastrado no GCM e no servidor de dados. Visando aprimorar a estrutura interna do código para minimizar qualquer tipo de falha.

Foi realizado uma mudança do registro do ID do *smartphone* do usuário no GCM para o momento da instalação do aplicativo. Essa mudança minimizou a possibilidade de falha do cadastro de ID.

4.2.3. Codificação

Após finalizada a implementação das estórias dos usuários, inicia-se os testes de unidades para cada uma dessas estórias.

Existem diversas estratégias para testar um *software*. O teste de unidade foca na verificação na menor unidade do projeto, tendo como finalidade descobrir erros dentro de cada unidade. Esse teste, no

contexto deste trabalho, focou a estrutura dos dados e a lógica interna de processamento.

4.2.4. Teste

Nesta etapa de testes é feito o teste de aceitação, também chamado de teste de cliente. O foco está nas características e na funcionalidade do sistema como um todo.

O parâmetro do teste de aceitação é obtido por meio das estórias dos usuários implementadas na versão corrente do aplicativo. Os *stakeholders* realizam o teste de aceitação para gerar a versão final.

4.2.4.1. Entrega do primeiro protótipo do aplicativo (versão beta)

A XP tem como premissa a aproximação do desenvolvedor e do cliente, fazendo com que o *feedback* ocorra de forma mais rápida.

No término da implantação das histórias foi gerado um versão *beta* para que os *stakeholders* realizassem o teste de aceitação.

A seguir são exibidas algumas das telas principais da versão *beta* do aplicativo institucional.

A Figura 10 ilustra a tela inicial, a Figura 11 exibe o *menu* de navegação, a Figura 12 é mostrado a listagem dos eventos institucionais e a Figura 13 exibe as descrições dos eventos institucionais.

Figura 10 -Tela Inicial do Aplicativo.



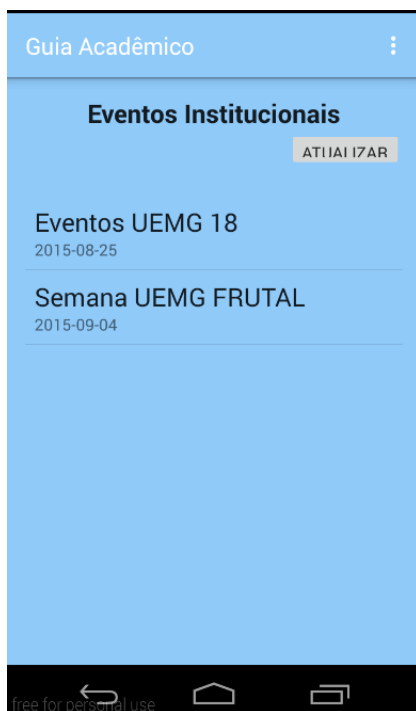
Fonte: o autor (2015).

Figura 11 - Menu de Navegação.



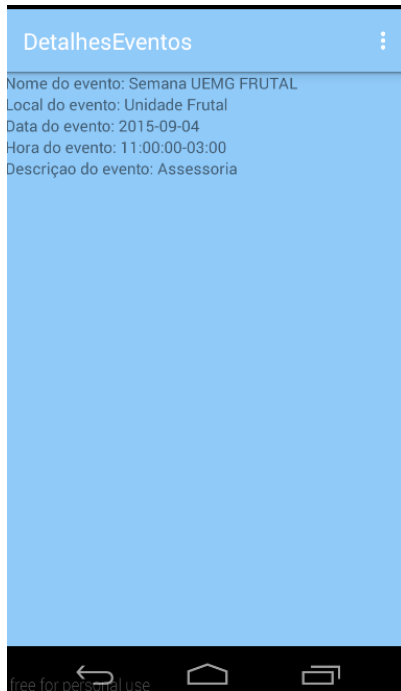
Fonte: o autor (2015).

Figura 12 - Lista dos Eventos Institucionais.



Fonte: o autor (2015).

Figura 13 - Detalhes dos Eventos Institucionais.



Fonte: o autor (2015).

Após realizar o teste de aceitação com os *stakeholders*, observou-se a aprovação de algumas estórias e a criação de outras para serem implementadas em futuras versões do aplicativo.

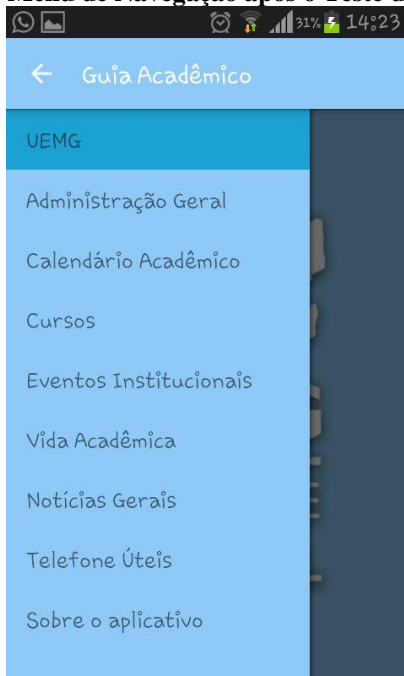
As novas estórias de usuários apontavam melhorias nos *layouts* referentes ao *menu* de navegação e detalhes das atividades institucionais. Os usuários alegaram que a cor do *menu* de navegação era muito forte e a letra dos detalhes do eventos era muito pequena, dificultando a leitura. Para implementar essas novas estórias, o ciclo da XP foi iniciado novamente.

A seguir é discutido a implementação dessas estórias que foram incorporadas à versão *beta*, dando origem a versão atual do aplicativo institucional.

4.2.4.2. Entrega da versão atual do aplicativo institucional

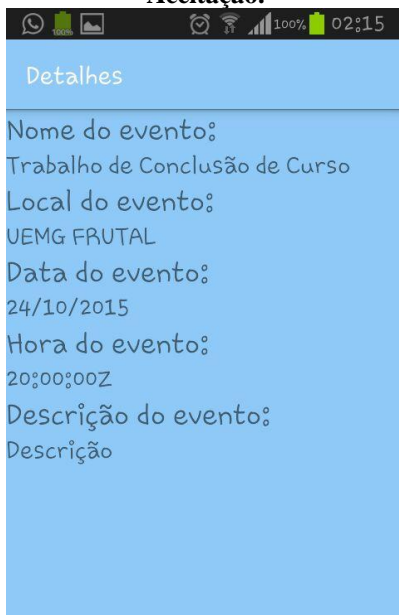
A Figura 14 exibe o *menu* de navegação do aplicativo e a Figura 15 exibe os detalhes dos Eventos Institucionais após as mudanças apontadas pelos *stakeholders*.

Figura 14 - Menu de Navegação após o Teste de Aceitação.



Fonte: o autor (2015).

Figura 15 - Detalhes dos Eventos Institucionais após o Teste de Aceitação.



Fonte: o autor (2015).

As demais telas do aplicativo institucional em sua última versão podem ser vistas no Apêndice D.

4.3. Manipulação dos Dados do Aplicativo

Conforme exposto no Capítulo 2, os *web services* são utilizados para realizar a integração entre duas aplicações diferentes sendo executadas em plataformas distintas.

Neste trabalho foi desenvolvido um *web services* para tornar as informações do aplicativo Android dinâmicas. A integração realizada foi entre a tecnologia JSP e o aplicativo Android.

Existem dois pontos do aplicativo Android que possui informações dinâmicas: os Eventos Institucionais e as Notícias Gerais.

Uma página JSP foi criada para cadastrar os dados dos eventos e das notícias. A assessoria de comunicação ficou responsável por essa função.

A seguir são exibidas algumas telas das páginas web criadas.

A Figura 16 exibe o cadastro dos Eventos Institucionais e a Figura 17 traz o cadastro das Notícias Gerais.

As demais páginas web podem ser vistas no Apêndice E.

Figura 16 - Página de Cadastro de Eventos Institucionais.

Cadastro de eventos

201.16.206.202:8000/AppInstitucional/Eventos.jsp

Cadastro de Eventos Institucionais

Origem:

Assessoria de empresa
 Diretório Acadêmico

Destino:

Alunos
 Professor
 Administrativo
 Todos

Área destinada para cadastro de eventos institucionais e universitários.

Nosso e-mail appoenag@gmail.com

Nome do evento:

Local:

Data do evento: Horário do evento:

Descrição:

Fonte: o autor (2015).

Figura 17 - Página de Cadastro de Notícias.

Cadastro de eventos

201.16.206.202:8000/AppInstitucional/Eventos.jsp

Cadastro de Eventos Institucionais

Origem:

- Assessoria de empresa
- Diretório Acadêmico

Destino:

- Alunos
- Professor
- Administrativo
- Todos

[Linkar item do destino](#)

Área destinada para cadastro de eventos institucionais e universitários.

Nosso e-mail: appoeng@maui.com

Nome do evento:

Local:

Data do evento: / / Horário do evento: : :

Descrição:

Fonte: o autor (2015).

Ao finalizar o cadastro do evento, as informações são gravadas no servidor de dados MySQL e o GCM compreende que houve uma mudança.

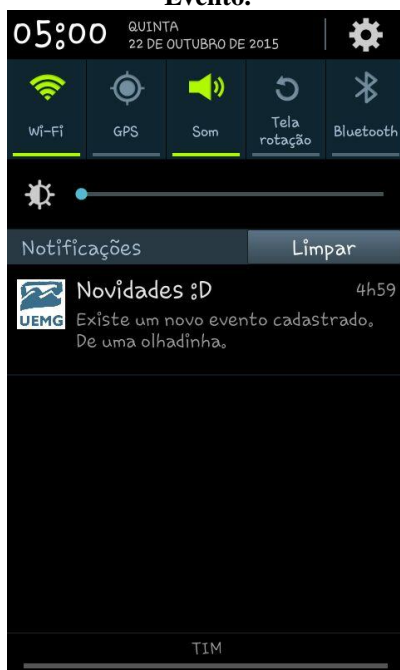
O diferencial em utilizar o GCM é tornar o servidor parte ativa das aplicações, pois ele deixa de ser apenas um local de consulta de informações e passa a fornecê-las sem a solicitação do cliente. É o

momento em que ocorre a utilização dos *web services*, pois é necessário realizar a interação entre o servidor e a aplicativo móvel.

As notificações *push* são utilizadas para que os *smartphones* recebam novas informações. No *push* o servidor avisa ao *smartphone* que existe um conteúdo para ser consumido. O GCM fica responsável pela entrega das notificações para os *smartphones*.

A Figura 18 exhibe uma notificação *push*.

Figura 18 - Notificação *push* referente ao Cadastro de um Novo Evento.



Fonte: o autor (2015).

4.4. Persistência dos dados

Para o armazenamento das informações foi feito uma modelagem do banco de dados. Foram criadas três tabelas referentes aos eventos, às notícias e aos IDs dos *smartphones*. A Figura 19 exibe as tabelas do banco de dados elaboradas juntamente com seus atributos.

Figura 19 - Tabelas do Banco de Dados.

evento	noticia	dispositivo
idevento INT	idnoticia INT	iddispositivo VARCHAR(255)
nome VARCHAR(45)	titulo VARCHAR(45)	
local VARCHAR(45)	descricao VARCHAR(45)	
data_evento DATE	data_criacao DATE	
hora_evento TIME	data_duracao DATE	
descricao VARCHAR(250)		
data_criacao DATE		
origem VARCHAR(10)		
aluno VARCHAR(4)		
professor VARCHAR(4)		
administrativo VARCHAR(4)		
todos VARCHAR(4)		
Indexes	Indexes	Indexes
PRIMARY	PRIMARY	PRIMARY

Fonte: o autor (2015).

O aplicativo móvel não possui um banco local, pois busca as informações dos eventos, notícias e ID dos *smartphones* utilizando o JSON. Isso torna a aplicativo mais leve, pois apenas consome essas informações.

5. AVALIAÇÃO DO APLICATIVO

Neste capítulo é apresentado o resultado obtido por meio de uma avaliação com os *stakeholders* sobre o uso do aplicativo institucional da UEMG. A avaliação foi realizada com discentes, docentes e a assessora de comunicação da unidade e analisou a aprovação dos usuários quanto à utilização do aplicativo como melhoria do canal de comunicação entre o administrativo e o meio acadêmico.

A avaliação foi feita por meio de um questionário disponível na íntegra no Apêndice F.

O questionário foi dividido em nove perguntas fechadas e uma aberta. Os *stakeholders* foram reunidos na sala do Núcleo de Práticas em Sistemas de Informação e o aplicativo foi instalado em seus *smartphones*. Após uma explicação das funcionalidades do aplicativo, os participantes da avaliação tiveram um período aproximado de dez minutos para utilização dessas funcionalidades. Foi feita uma breve explicação sobre o objetivo geral do

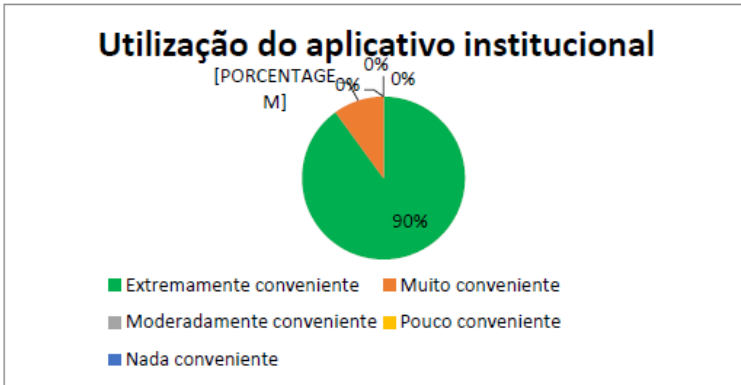
aplicativo e, em seguida, os *stakeholders* responderam ao questionário.

Abaixo, são exibidos os gráficos com os resultados da avaliação.

A Figura 20 exhibe os dados coletados que analisam a utilização do aplicativo de modo geral. O resultado obtido foi que 90% dos participantes que viabilizaram a utilização do aplicativo.

A pergunta feita foi: “Você acredita ser conveniente a utilização do aplicativo institucional da UEMG – Frutal?”

Figura 20 - Avaliação da Utilização do Aplicativo.



Fonte: o autor (2015).

A Figura 21 exibe os dados coletados no que diz respeito ao grau de profissionalismo da aparência do aplicativo, ou seja, a primeira impressão do usuário em relação ao *layout*. 60% dos participantes caracterizaram o aplicativo com a aparência muito profissional. Tal aparência pode ser melhorada para que esses participantes promovam a referida aparência para extremamente profissional.

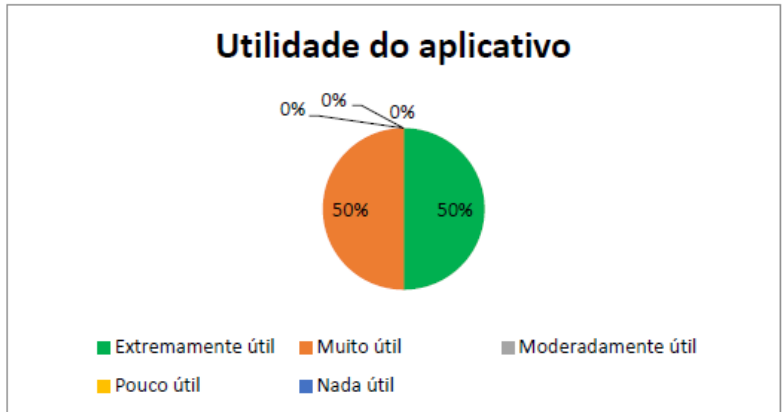
Figura 21- Avaliação do Profissionalismo do Aplicativo.



Fonte: o autor (2015).

A Figura 22 exibe os dados coletados referente à utilidade do aplicativo. Foi possível alcançar um resultado satisfatório, em que os participantes apontaram o aplicativo como útil.

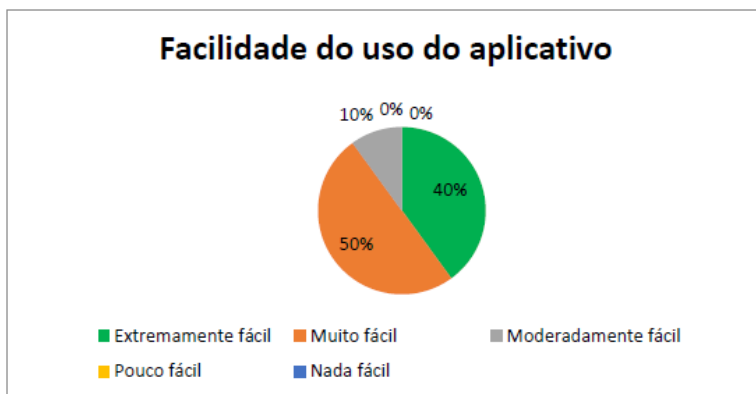
Figura 22 - Utilidade do Aplicativo.



Fonte: o autor (2015).

A Figura 23 exibe os dados coletados no que diz respeito à facilidade do uso das funcionalidades do aplicativo. Os participantes apontaram que as funcionalidades do aplicativo podem ser mais intuitivo, para que sejam mais bem exploradas pelos usuários.

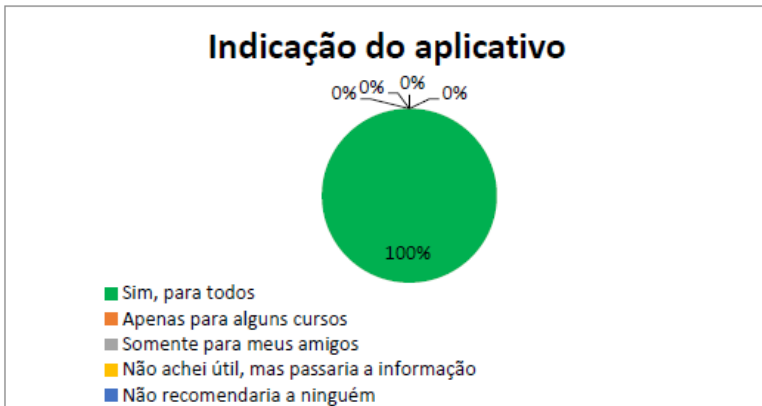
Figura 23 - Facilidade do Uso do Aplicativo.



Fonte: o autor (2015).

A Figura 24 exhibe os dados coletados referente à indicação do aplicativo para outros usuários. Foi possível obter 100% de indicação dos participantes para outros usuários fazendo com que seja uma forma de propaganda do aplicativo para comunidade acadêmica.

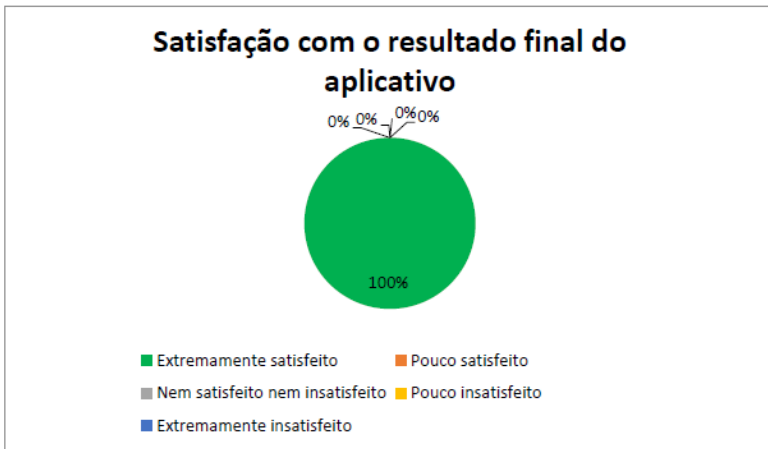
Figura 24- Indicação do Aplicativo.



Fonte: o autor (2015).

A Figura 25 exibe os dados coletados no que diz respeito ao resultado final do aplicativo como um todo, obtendo um resultado satisfatório por parte dos participantes que se mostraram extremamente satisfeitos.

Figura 25 - Satisfação com Resultado Final.



Fonte: o autor (2015).

Após a avaliação da questão aberta foi possível obter um *feedback* dos participantes (*stakeholders*) em relação às funcionalidades que eles achavam necessárias e que ainda não havia sido elaboradas. Por meio das respostas à essa questão novas estórias de usuários foram elaboradas, conforme lista a seguir:

- Inclusão de mais informações sobre os cursos de graduação;

- Dados sobre o estágio obrigatório;
- Tabela com a pontuação das atividades complementares;
- Lembrete para devolução de livros para a biblioteca;
- Valores dos itens da cantina da unidade;
- Inclusão de novos telefones.

A utilização do Aplicativo Institucional foi validada, após a apuração do resultado do questionário, como um canal de comunicação na Universidade do Estado de Minas Gerais.

Ressalta-se que o aplicativo foi aprovado de maneira empírica por usuários alvo e que novas avaliações podem ser conduzidas para validar as novas funcionalidades apontadas pelos mesmos.

6. CONCLUSÃO

O presente trabalho teve como objetivo criar um canal de comunicação entre o setor administrativo e toda a comunidade acadêmica que facilitasse a difusão da informação.

Utilizando o método de desenvolvimento ágil, XP, foi possível realizar o desenvolvimento de um aplicativo móvel para Universidade do Estado de Minas Gerais, unidade de Frutal.

O aplicativo se limita em realizar a divulgação dos dados institucionais e eventos ligados à referida instituição. Esses dados são mostrados apenas se o *smartphone* estiver conectado à rede (Dados móveis ou Wi-Fi).

Foi realizado um questionário para avaliar com mais precisão se o aplicativo atendeu à proposta inicial. O resultado foi de grande valia. Por meio do questionário foi possível verificar que o aplicativo teve alto índice de aprovação. Assim, foi possível verificar que é válida a existência de um aplicativo

institucional para UEMG, unidade de Frutal, pois facilita o acesso à informação de maneira simples e objetiva.

O aplicativo será disponibilizado para o Núcleo de Prática em Sistemas de Informação (NUPSI) da unidade. O NUPSI é um projeto pedagógico de complementação da formação acadêmica, com atividades práticas voltadas para o desenvolvimento das competências e habilidades dos estudantes do curso de Sistemas de Informação. O NUPSI ficará incumbido de realizar as devidas manutenções do aplicativo institucional e desenvolver as futuras versões do software.

Para projetos futuros o aplicativo poderá ser implantado em outras unidades da Universidade do Estado de Minas Gerais. O aplicativo também poderá ser evoluído para realizar uma integração com o sistema WebGiz (Sistema interno de gestão das atividades acadêmicas), permitindo acesso à mais informações, tais como: notas e faltas. Bem como integração com o sistema Moodle, para coletar e

enviar notificações sobre as atividades e outras ocorrências. É possível ainda realizar uma integração com o sistema da biblioteca para permitir a consulta e controle de devolução dos livros da biblioteca.

REFERÊNCIAS BIBLIOGRÁFICAS

- BECK, K. **Programação Extrema (XP) explicada**. São Paulo: Bookman, 2008. p. 182.
- BERNSTEIN, P. A. **Middleware: a model for distributed system services**. [s.l]:New York, 1996. p. 98.
- COULOURIS, G. F. **Sistemas Distribuídos Conceitos e Projeto**. 5. ed. Porto Alegre: Artmed, 2007. p. 780.
- DEITEL, H. M. **Java Como Programar**. 6. ed. São Paulo: Pearson Education, 2005. p. 1110.
- FONSECA, R.; SIMOES, A. **Alternativas ao XML: YAML e JSON**. In: XATA 2007 - XML: Aplicações e Tecnologias Associadas, 5., 2007, Departamento de Informática da Faculdade de Ciências da Universidade de Lisboa. [s.l. :s.n.], 2007 p. 14.
- FOWLER, M. **The New Methodology**. 2005. Disponível em

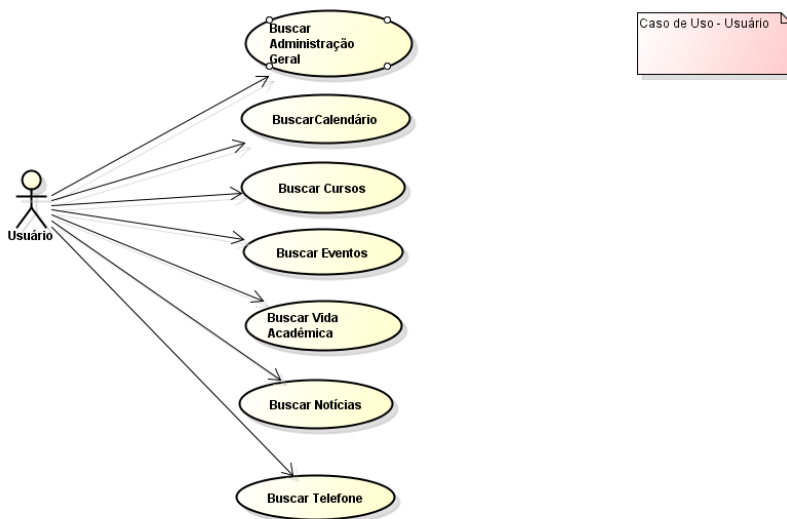
- <<http://www.martinfowler.com/articles/newMethodology.html>>. Acesso em: 13 jul. 2015.
- GAMA, L.G.C; SCHEER, S. **Construção de um repositório para objetos educacionais hipermídia** In: Congresso Nacional de Ambientes Hipermídia para Aprendizagem. 1., Florianópolis: [s.n.], 2004. p. 8.
- GEREMIA, J. **Tutorial de Introdução a Banco de Dados**. 2010. Disponível em:<http://www.telecom.uff.br/pet/petws/downloads/tutoriais/db/Tut_DB.pdf>. Acesso em: 20 de ago. 2015.
- LECHETA, R. R. **Google Android: Aprenda a criar aplicações para dispositivos móveis com o Android SDK**. São Paulo: Novatec, 2007. p. 792.
- LEMOS, A. **Comunicação e práticas sociais no espaço urbano: as características dos Dispositivos Híbridos Móveis de Conexão Multirredes (DHMCM)**. Comunicação, mídia e consumo. v. 4, n.10, 2007. p. 40.

- NAKAMURA, E. F. **Computação Móvel: Novas Oportunidades e Novos Desafios COMPUTAÇÃO MÓVEL: NOVAS OPORTUNIDADES E NOVOS DESAFIOS.** 2015. Disponível em: <<http://www.deinf.ufma.br/~fssilva/palestras/2006/jim/cm/aulas/aula02.pdf> t>. Acesso em 20 de ago. 2015.
- PRESSMAN, Roger S. **Engenharia de Software: uma abordagem profissional.** 7.ed. – Porto Alegre: AMGH, 2011. 780p.
- Revista Info Exame. **POR QUE O ANDROID ENGOLIU A APPLE.** São Paulo: Editora Abril, 2011. p. 58.
- RICKARDSON, L. e RUBY, S. **RESTful Serviços Web,** Rio de Janeiro: Alta Books Ltda, 2007. p. 336.
- RIES, E. **The Lean Startup: How Today's Entrepreneurs Use Continuous Innovation to Create Radically Successful Businesses.** [s.l.]: Crown Business, 2011. p. 320.

- SAMPAIO, C. Guia do Java Enterprise Edition 5:**
Desenvolvido aplicações corporativas. Rio de Janeiro: Brasport, 2007.
- TANENBAUM, A. S. Sistemas Distribuídos princípios e paradigmas.** 2ed. [s.l.]: Prentice Hall, 2008. p.416.
- TELES, V. M. Extreme Programming:** Aprenda como encantar seus usuários desenvolvendo software com agilidade e alta qualidade. São Paulo: Novatec. 2004. p. 320.
- TESSARO, L.B.H. Ferramenta para a utilização de esquema em bancos de dados implementados em MySQL.** Santa Catarina: UNIVALI, 2013. p. 89.

APÊNDICE A - Documentação do Aplicativo

Figura A1 – Diagrama de Caso de Uso do Usuário



Fonte: o autor (2015).

Quadro A1 – Descrição de Caso de Uso do Usuário.

ID	A 1 – 1
Nome do Caso de Uso	Buscar Administração Geral
Atores Primários	Usuário
Propósito	O sistema deve permitir o processamento das informações referente a administração geral da Universidade, com os seguintes atributos: responsável pelo departamento, nome e email.
Visão Geral	O usuário acessa a função de busca Administração geral.
Pré-condições	<ul style="list-style-type: none">▪ Não existe pré-condições para acessar essa informação, basta instalar o aplicativo.
Pós-condições	<ul style="list-style-type: none">▪ Busca as informações referente a administração geral;
Referências cruzadas	
Ação do Ator	Resposta do Sistema
1) Buscar informações da administração geral.	2) O sistema exibe uma lista com todos os itens da referente a administração geral da unidade de Frutal.
3) O usuário escolhe a informação desejada	4) O sistema mostra detalhes da informação.

Fluxos Alternativos	
ID	A 1 – 2
Nome do Caso de Uso	Buscar Calendário
Atores Primários	Usuário
Propósito	O sistema deve permitir o processamento das informações referente ao calendário acadêmico da Universidade.
Visão Geral	O usuário acessa a função de buscar de Administração geral.
Pré-condições	<ul style="list-style-type: none"> ▪ Não existe pré-condições para acessar essa informação, basta instalar o aplicativo.
Pós-condições	<ul style="list-style-type: none"> ▪ Busca das informações referente ao

	calendário acadêmico;
Referências cruzadas	
Ação do Ator	Resposta do Sistema
1) Buscar informações do calendário acadêmico.	2) O sistema exibe uma lista com calendário acadêmico da unidade de Frutal.
Fluxos Alternativos	
ID	A 1 – 3
Nome do Caso de Uso	Buscar Cursos
Atores Primários	Usuário
Propósito	O sistema deve permitir o processamento das informações referente aos cursos de graduação da Universidade e seus respectivos coordenadores, com os seguintes

	atributos: responsável pelo curso, nome e email..
Visão Geral	O usuário acessa a função de buscar cursos de graduação.
Pré-condições	<ul style="list-style-type: none"> ▪ Não existe pré-condições para acessar essa informação, basta instalar o aplicativo.
Pós-condições	<ul style="list-style-type: none"> ▪ Busca das informações referente aos cursos de graduação;
Ação do Ator	Resposta do Sistema
1) Buscar informações dos cursos de graduação.	2) O sistema exibe uma lista com os cursos de graduação da unidade de Frutal.

Fluxos Alternativos	
ID	A 1 – 4
Nome do Caso de Uso	Buscar Eventos
Atores Primários	Usuário
Propósito	O sistema deve permitir o processamento das informações referente aos eventos institucionais da Universidade, com os seguintes atributos: nome do evento, local, data do evento, hora, descrição, origem.
Visão Geral	O usuário acessa a função de Busca Eventos que retorna todos os eventos cadastrado em forma de lista.
Pré-condições	<ul style="list-style-type: none"> ▪ O usuário deve estar conectado à

	internet para que os dados estejam sincronizados.
Pós-condições	▪ Busca dos eventos cadastrados;
Referências cruzadas	
Ação do Ator	Resposta do Sistema
1) Buscar Eventos cadastrado.	2) O sistema exibe uma lista com todos os eventos institucionais cadastrado.
3) O usuário escolhe o evento desejada	4) O sistema mostra detalhes do evento.
Fluxos Alternativos	
1. Caso não tenha acesso a internet os dados não serão atualizado e aparecerá uma mensagem solicitando acesso a rede.	
ID	A 1 – 5
Nome do Caso de Uso	Buscar Vida Acadêmica
Atores Primários	Usuário

Propósito	O sistema deve permitir o processamento das informações referente aos vida academia da Universidade.
Visão Geral	O usuário acessa a função de Busca Vida Acadêmica que retorna todos os dados necessários para o dia-a-dia do discente.
Pré-condições	<ul style="list-style-type: none"> ▪ Não existe pré-condições para acessar essa informação, basta instalar o aplicativo.
Pós-condições	<ul style="list-style-type: none"> ▪ Informações relevantes para o acadêmico.
Referências cruzadas	
Ação do Ator	Resposta do

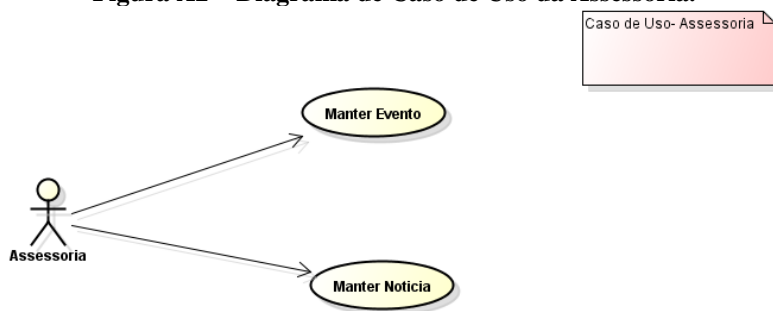
	Sistema
1) Buscar Vida Acadêmica.	2) O sistema exibe vários menus com diferentes categorias.
3) O usuário escolhe categoria desejada	4) O sistema mostra informações referente a categoria selecionada.
Fluxos Alternativos	
ID	A 1 – 6
Nome do Caso de Uso	Buscar Notícias
Atores Primários	Usuário
Propósito	O sistema deve permitir o processamento das informações referente aos notícias institucionais da Universidade, com os seguintes atributos: nome do notícia, link, descrição.
Visão Geral	O usuário acessa a

	função de Buscar Notícias que retorna todas as notícias cadastradas em forma de lista.
Pré-condições	<ul style="list-style-type: none"> ▪ O usuário deve estar conectado à internet para que os dados estejam sincronizados.
Pós-condições	<ul style="list-style-type: none"> ▪ Busca das notícias cadastradas;
Referências cruzadas	
Ação do Ator	Resposta do Sistema
1) Buscar Notícias cadastradas.	2) O sistema exibe uma lista com todas as notícias institucionais cadastradas.
3) O usuário escolhe a notícia desejada	4) O sistema mostra detalhes da notícia.
Fluxos Alternativos	
2. Caso não tenha acesso a internet os dados não serão	

atualizado e aparecerá uma mensagem solicitando acesso à rede.	
ID	A 1 – 7
Nome do Caso de Uso	Buscar Telefone
Atores Primários	Usuário
Propósito	O sistema deve permitir o processamento das informações referente a contatos telefônicos.
Visão Geral	O usuário acessa a função de Busca Telefone que retorna todos os contatos pré-cadastrados.
Pré-condições	<ul style="list-style-type: none"> ▪ Não existe pré-condições para acessar essa informação, basta instalar o aplicativo.
Pós-condições	<ul style="list-style-type: none"> ▪ Lista de números telefones.
Referências cruzadas	

Ação do Ator	Resposta do Sistema
1) Buscar Telefone	2) O sistema exibe vários contatos em uma lista.
3) O usuário escolhe o contato	4) O sistema irá realizar uma ligação telefônica para contato selecionado.
Fluxos Alternativos	

Figura A2 – Diagrama de Caso de Uso da Assessoria.



Fonte: o autor (2015).

Quadro A2 – Descrição de Caso de Uso do Assessoria.

ID	B 1 - 1
Nome do Caso de Uso	Manter Evento
Atores Primários	Assessoria
Propósito	O sistema deve permitir a inclusão, alteração e remoção de eventos institucionais, com os seguintes atributos: nome do evento, local, data do evento, hora, descrição, origem.
Visão Geral	O usuário acessa a função de Cadastro de Evento que possui as opções para inclusão, alteração, remoção e consulta de um evento pelo nome e código de identificação. Após a utilização de qualquer uma das opções o usuário deverá confirmar ou cancelar a operação.
Pré-condições	<ul style="list-style-type: none">▪ Um evento para ser removido não poderá estar presente em nenhum projeto.
Pós-condições	<ul style="list-style-type: none">▪ Um novo cliente é incluído, alterado ou excluído do sistema.

Referências cruzadas	
Ação do Ator	Resposta do Sistema
1) Incluir um novo evento – Evento Institucional	2) O sistema exibe o formulário com todos os campos para o cadastro do evento.
3) O usuário preenche o campos e seleciona a opção de Confirmar.	4) O sistema permite a inclusão dos dados nos campos do formulário.
5) O usuário clica em ok para confirmar a inclusão dos dados.	6) O sistema exibe uma mensagem de confirmação e salva os dados.
Fluxos Alternativos	
4 – Verifica se já existe algum campo obrigatório em branco. Exibe mensagem informando que qual campo está em branco e for o usuário a inserir o dado equivalente.	
5 – Caso o usuário cancele. O sistema não salva nada.	

ID	B 1 - 2
Nome do Caso de Uso	Manter Notícia
Atores Primários	Assessoria
Propósito	O sistema deve permitir a inclusão, alteração e remoção de notícias institucionais, com os seguintes atributos:

	nome da notícia, data de duração, link, descrição.
Visão Geral	O usuário acessa a função de Cadastro de Cliente que possui as opções para inclusão, alteração, remoção e consulta de uma categoria pelo nome. Após a utilização de qualquer uma das opções o usuário deverá confirmar ou cancelar a operação.
Pré-condições	<ul style="list-style-type: none"> ▪ Uma notícia para ser removido não poderá estar presente em nenhum projeto.
Pós-condições	<ul style="list-style-type: none"> ▪ Uma nova notícia é incluída, alterada ou excluída do sistema.
Referências cruzadas	

Ação do Ator	Resposta do Sistema
1) Incluir uma nova notícia.	2) O sistema exibe o formulário com todos os campos para o cadastro da notícia.
3) O usuário preenche o campos e seleciona a opção de Cadastrar.	4) O sistema permite a inclusão dos dados nos campos do formulário.
5) O usuário clica em ok para confirmar a inclusão dos dados.	6) O sistema exibe uma mensagem de confirmação e salva os dados.
Fluxos Alternativos	
4 – Verifica se já existe algum campo obrigatório em branco. Exibe mensagem informando que qual campo está em branco e for o usuário a inserir o dado equivalente.	
5 – Caso o usuário cancele. O sistema não salva nada.	

APÊNDICE B - Versão Inicial do Aplicativo Institucional

Figura B1 - Tela Inicial do Aplicativo.



Fonte: o autor (2015).

Figura B2 - Menu de Navegação.



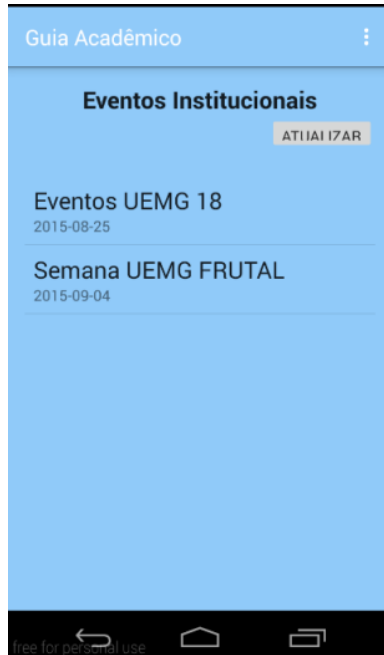
Fonte: o autor (2015).

Figura B3 - Informações dos Cursos de Graduação.

Cursos - Graduação	
^	Administração
Professor	Coordenador do Curso teste@uemgfrutal.com
Vice Cood	Vice CoodCood@uemgfrutal.com
∨	Comunicação Social
^	Direito
Leonardo	Cood do Cursoleo@uemgfrutal.com
∨	Geografia
∨	Sistemas de Informação
∨	Tec. em Alimentos
∨	Tec. em Produção Sucroalcooleira
free for personal use	

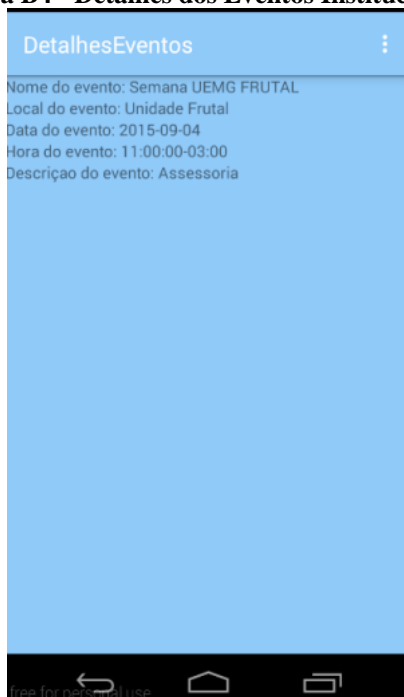
Fonte: o autor (2015).

Figura B3 - Lista dos Eventos Institucionais.



Fonte: o autor (2015).

Figura B4 - Detalhes dos Eventos Institucionais.



Fonte: o autor (2015).

Figura B4 - Notificação de um novo evento.



Fonte: o autor (2015).

APÊNDICE C – Estórias de Usuários

Mural de notícias – Prioridade: 3	
Responsabilidade	Colaboração
Informações relevantes para os discentes, docentes e funcionários.	Docente
Editais para intercâmbio, projetos de extensão.	Assessoria

Eventos Institucionais – Prioridade: 4	
Responsabilidade	Colaboração
Informações dos eventos que envolvem os docentes, como eventos universitários.	Docente
Informações dos eventos que envolvem a faculdade em geral, como aviso de	Assessoria

rematrícula, eventos da Universidade.	
Notificações dos eventos.	Assessoria

Administração geral– Prioridade: 3	
Responsabilidade	Colaboração
Informações referentes à administração geral da Unidade (e-mail de contato).	Assessoria
Informações dos coordenadores dos cursos.	Docente

Calendário Acadêmico – Prioridade: 4	
Responsabilidade	Colaboração
Disponibilidade do calendário acadêmico.	Discente

Telefone Uteis – Prioridade: 1	
Responsabilidade	Colaboração

Informar telefones dos principais pontos da cidade, como alimentação, moradia e lazer.	Discente
--	----------

Vida Acadêmica - Prioridade: 3	
Responsabilidade	Colaboração
Informações sobre as necessidades que podem surgir no decorrer da vida acadêmica	Discente

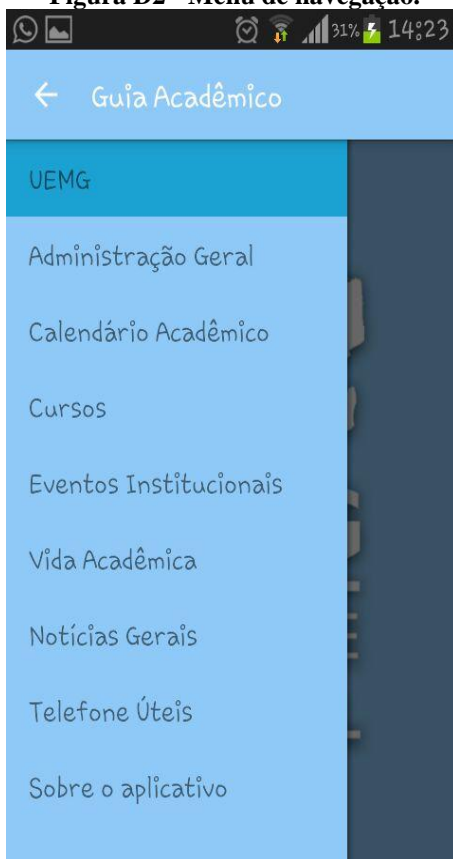
APÊNDICE D - Aplicativo Institucional na Versão Atual

Figura D1 - Tela inicial do aplicativo.



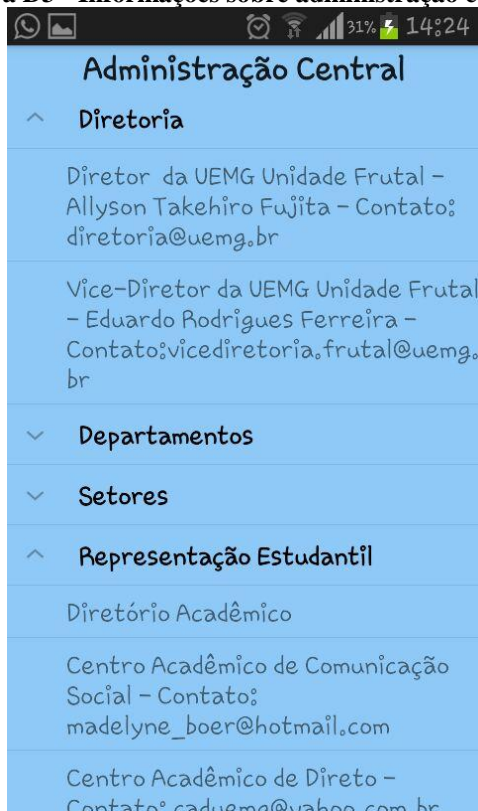
Fonte: o autor (2015).

Figura D2 - Menu de navegação.



Fonte: o autor (2015).

Figura D3 - Informações sobre administração central.



Fonte: o autor (2015).

Figura D4 - Calendário acadêmico.

Calendário

Calendário Acadêmico 2015

SETEMBRO					
Dom	Seg	Ter	Qua	Qui	Sex
		01	02	03	04
06	07	08	09	10	11
13	14	15	16	17	18
20	21	22	23	24	25
27	28	29	30		

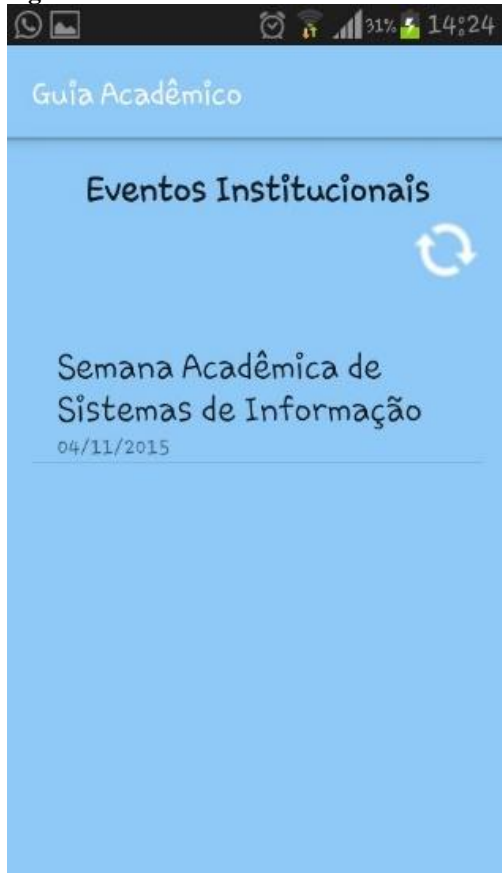
Dias Letivos: 24

Observações:

- 07 – Independência do Brasil
- Dias Acadêmicos do Curso de Administração
- Dias Acadêmicos do Curso de Sistemas de Informação
- 11 a 19- Semana UEMG
- 12 – Dia letivo ref. 2ª feira
- 19 – Dia letivo ref. 3ª feira
- 26 – Dia letivo ref. 4ª feira

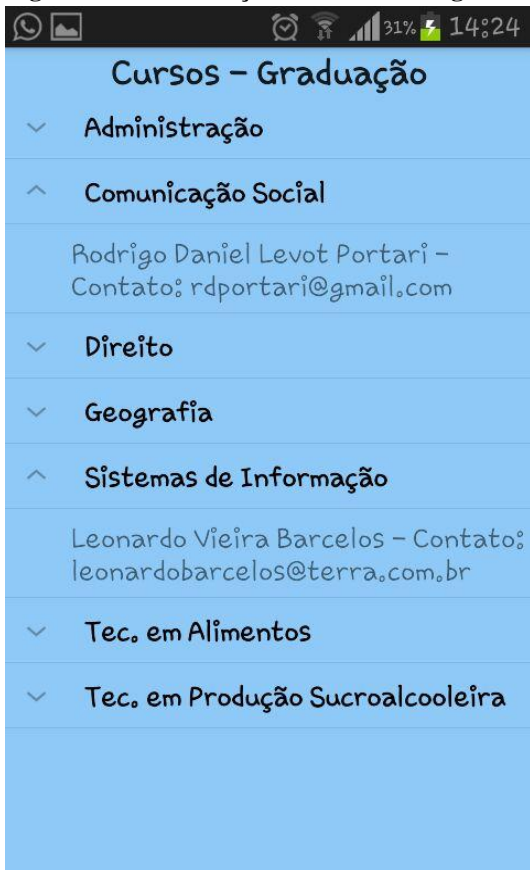
Fonte: o autor (2015).

Figura D5 - Lista dos Eventos Institucionais.



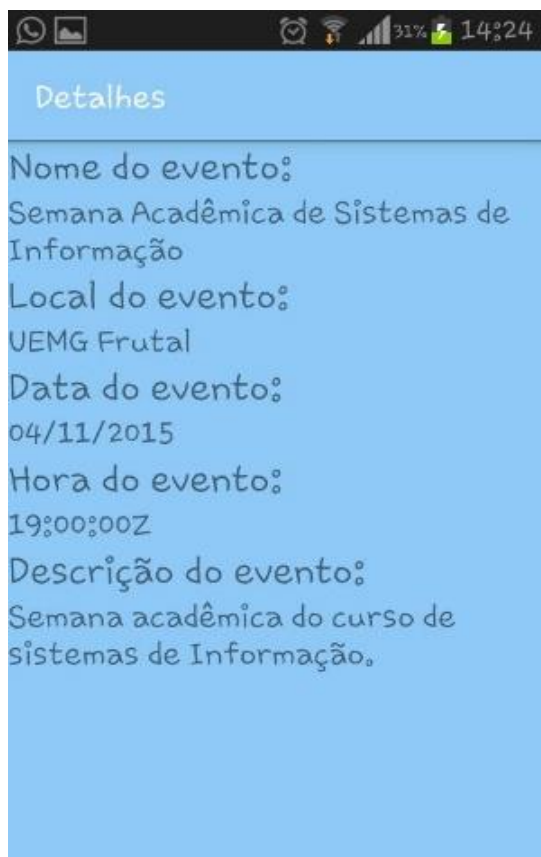
Fonte: o autor (2015).

Figura D6 - Informações dos cursos de graduação.



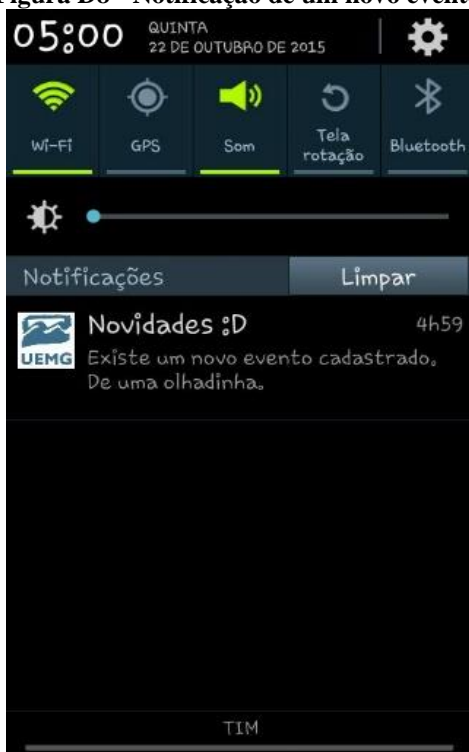
Fonte: o autor (2015).

.Figura D7 - Detalhes dos Eventos Institucionais.



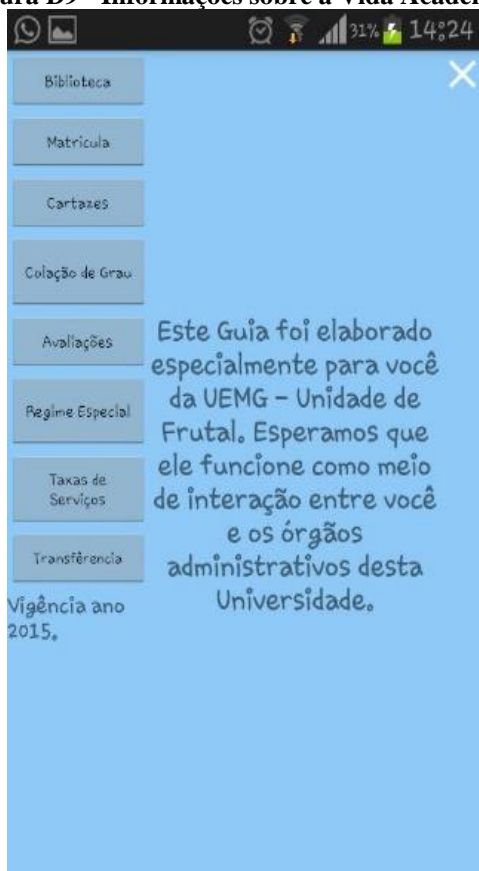
Fonte: o autor (2015).

Figura D8 - Notificação de um novo evento.



Fonte: o autor (2015).

Figura D9 - Informações sobre a Vida Acadêmica.



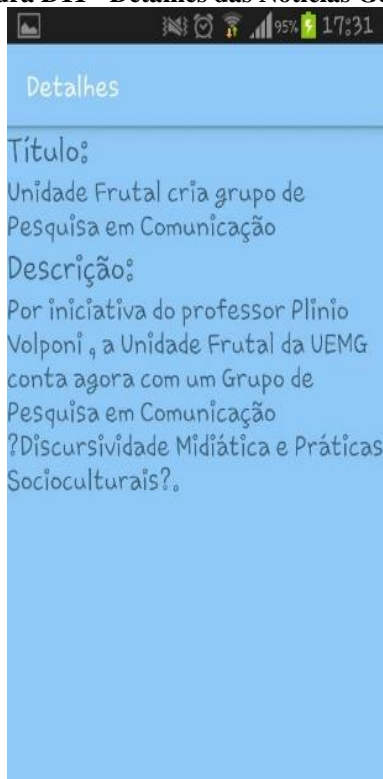
Fonte: o autor (2015).

Figura D10 - Lista das Notícias Gerais.



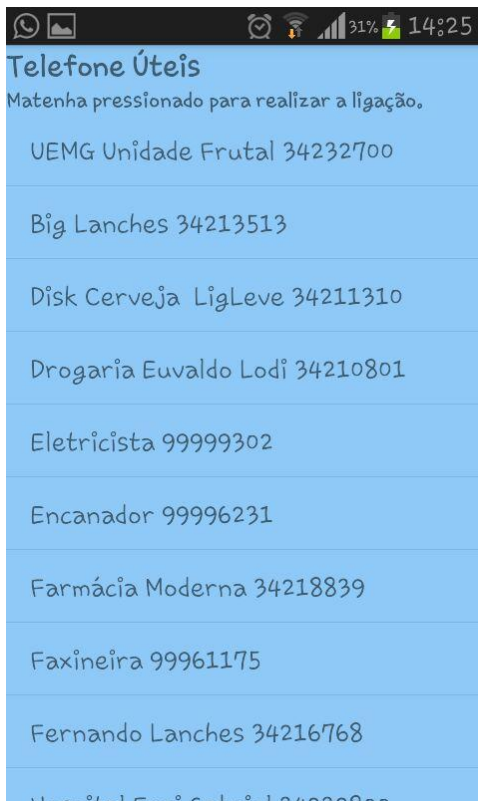
Fonte: o autor (2015).

Figura D11 - Detalhes das Notícias Gerais.



Fonte: o autor (2015).

Figura D12 - Telefones Úteis.



Fonte: o autor (2015)

APÊNDICE E - Telas da Aplicação Web

Figura E1 - Tela do cadastro dos Eventos Institucionais.

Cadastro de Eventos Institucionais

Origem:

- Assessoria de empresa
- Destino Acadêmico
- Destino

Destino:

- Alunos
- Professores
- Administrativo
- Todos

[Limpar lista de destino](#)

Área destinada para cadastro de eventos institucionais e universitários.
Novo e-mail: apacem@pmail.com

Nome do evento:

Local:

Data do evento: Horário do evento:

Descrição:

[Cadastro Evento](#)

Fonte: o autor (2015).

Figura E220 - Consulta dos Eventos Institucionais.

Consulta de eventos

201.16.206.202:8000/AppInstitucional/consultaEventos.jsp

- [Consulta dos noticias](#)

Consulta de Eventos Institucionais

Área destinada para visualizar os eventos cadastrados.

Lista dos eventos cadastrados:


Identificação	Nome do Evento	Data do Evento
1	Semana Acadêmica de Sistemas de Informação	2015-11-04

Caso queira excluir algum eventos basta digitar o número da Identificação do evento.

Número de Identificação:

Fonte: o autor (2015).

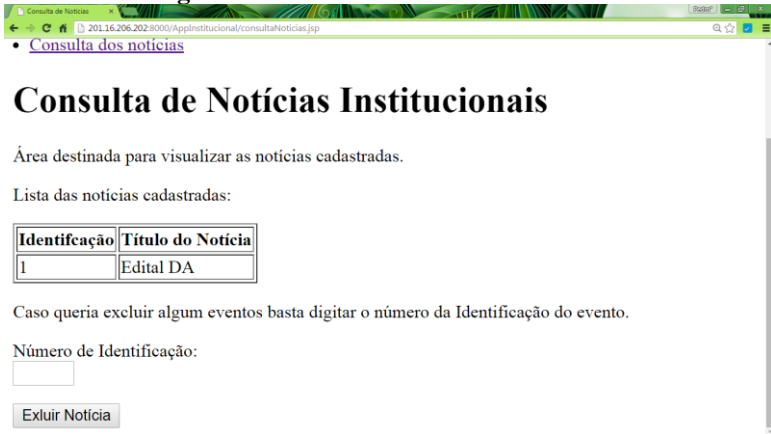
Figura E3 - Cadastro das Notícias Gerais.



The image shows a web browser window with the address bar displaying "201.16.206.202:8000/AppInstitucional/noticias.jsp". The page title is "Cadastro de Notícias". The main heading is "Notícias". Below the heading, there is a paragraph: "Área destinada para cadastro de noticias universitárias em geral." followed by "Nosso e-mail nupsi@gmail.com". The form includes a "Titulo:" label with a text input field, a "Data da duração:" label with a date input field (format: dd/mm/aaaa), and a "Descrição:" label with a large text area. At the bottom of the form is a button labeled "Enviar Notícia".

Fonte: o autor (2015).

Figura E4 - Consulta das Notícias Gerais.



Fonte: o autor (2015).

APÊNDICE F - Questionário de Avaliação

Nome (OPCIONAL):

Curso:

Período:

1. Você acreditar ser conveniente a utilização do aplicativo institucional da UEMG -Frutal?

- Extremamente conveniente
- Muito conveniente
- Moderadamente conveniente
- Pouco conveniente
- Nada conveniente

2. A aparência do aplicativo é profissional?

- Extremamente profissional
- Muito profissional
- Moderadamente profissional
- Pouco profissional
- Nada profissional

3. O aplicativo é útil para você? (Qual a relevância das informações fornecidas)

- Extremamente
- Muito útil
- Moderadamente útil
- Pouco útil
- Nada útil

4. O quão útil é este aplicativo para os calouros?

- Extremamente
- Muito útil
- Moderadamente útil
- Pouco útil
- Nada útil

5. O quão fácil de usar é este aplicativo?

- Extremamente fácil
- Muito fácil
- Moderadamente fácil
- Pouco fácil

Nada fácil

6. De forma geral, você está satisfeito com o resultado?

Extremamente satisfeito

Pouco satisfeito

Nem satisfeito nem insatisfeito

Pouco insatisfeito

Extremamente insatisfeito

7. Como você avalia as informações contidas na opção eventos institucionais?

Excelente

Muito bom

Moderado

Não muito bom

Ruim

8. Como você avalia as informações contidas na opção notícia?

Excelente

Muito bom

- Moderado
- Não muito bom
- Ruim

9. Você recomenda o aplicativo?

- Sim, para todos.
- Apenas para alguns cursos
- Somente para meus amigos
- Não achei útil, mas passaria a informação.
- Não recomendaria a ninguém

10. Que outras funcionalidades você acha que devem ser adicionada ao aplicativo? (Resposta aberta)



Editora Prospectiva