



UNIVERSIDADE DO ESTADO DO RIO DE JANEIRO

Junho 2012
ISSN 1413-9014
E-ISSN 2317-2193

CADERNOS DO IME

Série Informática

Volume 33

Um Sistema Colaborativo de Apoio à Construção de Roteiros de Viagem 7

Daniel Ramos, Fellipe Ferrão, Rosa Maria Costa, Maria Clicia S. de Castro

Aplicando a Experimentação na Engenharia de Requisitos: O Caso do Método ERi*c 19

Antonio de Padua A. Oliveira, Julio Cesar Leite, Luiz Marcio Cysneiros

CADERNOS DO IME

Série Informática

Volume 33

Publicação do Instituto de Matemática e Estatística da Universidade do Estado do Rio de Janeiro. Periodicidade semestral, com circulação em junho e dezembro.

Ricardo Vieirals de Castro
Reitor

Paulo Roberto Volpato Dias
Vice-Reitor

Maria Georgina Muniz Washington
Diretora do Centro de Tecnologia e Ciência

Geraldo Magela da Silva
Diretor do Instituto de Matemática e Estatística

Pedro Antonio Sarubbi
Vice-Diretor do Instituto de Matemática e Estatística

Normalização, divulgação e distribuição:
Biblioteca do Centro de Ciências de Tecnologia A (CTC/A) da rede Sirius
de Biblioteca da UERJ – ctca@uerj.br

Organização do volume 33:
Alexandre Sztajnberg
Fabiano de Souza Oliveira

Correspondência:
Universidade do Estado do Rio de Janeiro
Instituto de Matemática e Estatística
Editores do Cadernos do IME - Série Informática
Rua São Francisco Xavier, 524 - Pavilhão Reitor João Lyra Filho,
6º andar, sala 6019 B
Maracanã - 20550-900 – Rio de Janeiro, RJ

Telefax: +55 21 2334-0144
e-mail: cadernos_inf@ime.uerj.br
site: <http://www.ime.uerj.br/cadernos/cadinf/>

Os artigos enviados para publicação deverão ser inéditos, com exceção de resumos ou teses, são de responsabilidade de seus autores, e não refletem, necessariamente, a opinião do IME. Sua reprodução é livre, em qualquer outro veículo de comunicação, desde que citada a fonte.

Produção e editoração gráfica:
Alexandre Sztajnberg
Fabiano de Souza Oliveira

Organizadores:

Alexandre Sztajnberg

Fabiano de Souza Oliveira

Contato: cadernos_inf@ime.uej.br

Junho 2012
ISSN 1413-9014
E-ISSN 2317-2193

CADERNOS DO IME

Série Informática

Volume 33

Um Sistema Colaborativo de Apoio à Construção de Roteiros de Viagem 7

Daniel Ramos, Fellipe Ferrão, Rosa Maria Costa, Maria Clícia S. de Castro

Aplicando a Experimentação na Engenharia de Requisitos: O Caso do Método ERi*c 19

Antonio de Padua A. Oliveira, Julio Cesar Leite, Luiz Marcio Cysneiros

Nota dos Editores

No início do ano de 2012 o Conselho Editorial decidiu apoiar o projeto Portal de Publicações Eletrônicas da UERJ. Com isso, houve a necessidade de se planejar e executar a cópia e a reestruturação da base de volumes disponíveis na página original, para o novo sistema, uma instalação do SEER/OJS.

Para esta atividade contamos com o apoio de da equipe do DEPEXT, que administra o portal, e com o trabalho de duas alunas da graduação, bolsistas PIBIC, Anna Carolina Cunha e Ana Carolina Santos, a quem agradecemos. Além de trilharem o caminho para o uso da versão disponível do sistema, elas corrigiram várias pendências nas versões finais de Volumes anteriores, geraram os arquivos em PDF e inseiram os mesmos no portal. Com isso, todos os volumes eletrônicos disponíveis na página original do Cadernos do IME - Série Informática, <http://magnum.ime.uerj.br/cadernos/cadinf/>, estão disponíveis no portal, <http://www.e-publicacoes.uerj.br/index.php/cadinf>.

Por esta razão a atividade editorial teve sua prioridade um pouco diminuída. Desta forma, o Volume 33, que deveria ter sido publicado em Junho de 2012 está sendo publicado em Setembro de 2013. Esperamos que na sequência a atividade editorial volte ao regime e que os volumes 34 e 35 sejam publicados em breve.

Reiteramos os agradecimentos aos revisores internos e externos, que fizeram suas avaliações em tempo. Agradecemos também a paciência dos autores pela demora no processo.

Setembro de 2013

Conselho Editorial

Alexandre Sztajnberg
Ana Leticia Luboc
Antônio de Pádua A. Oliveira
Fabiano de Souza Oliveira
Guilherme Lucio Abelha Mota

Maria Clícia Stelling
Marinilza Bruno de Carvalho
Neide Santos
Rosa Maria Moreira da Costa
Vera Maria Benjamim Werneck

Contato: cadernos_inf@ime.uerj.br

Revisores *ad hoc* internos

Ana Leticia Luboc
Fabiano de Souza Oliveira
Leandro Marzulo

Maria Clícia Stelling de Castro
Vera Maria Benjamim Werneck

Revisores *ad hoc* externos

Claudia Cappelli (Dr., UniRio)
Sean Wolfgang Matsui Siqueira (Dr., UniRio)
Kate Cerqueira Revoredo (Dr., UniRio)
Kleber Vieira Cardoso (Dr., UFG)
Denis Silva da Silveira (Dr., UFPE)
Ricardo Engelbert (MSc., Engelbert e Cia, Ltda.)
Marcelo Lobosco (Dr., UFJF)

Cadernos do IME – Série Informática

ISSN 1413-9014

E-ISSN 2317-2193

Apresentação

Para o Volume 33 dos Cadernos do IME – Série Informática dois artigos foram selecionados.

O primeiro, Um Sistema Colaborativo de Apoio à Construção de Roteiros de Viagem, dos autores Daniel Ramos, Fellipe Ferrão, Rosa Maria Costa, Maria Clícia S. de Castro, apresenta uma aplicação Android para apoiar o usuário na confecção de roteiros de viagem e posterior acompanhamento em tempo real. A aplicação integra diferentes tecnologias de localização e mapas. Um protótipo foi desenvolvido.

O segundo artigo, Aplicando a Experimentação na Engenharia de Requisitos: O Caso do Método ERi*c, dos autores Antonio de Padua A. Oliveira, Julio Cesar Leite, Luiz Marcio Cysneiros, apresenta uma avaliação de como o método Eri*c, proposto na tese de Doutorado do 1º autor, favorece as competências do Engenheiro de Software comparado com seu método base, o Framework i*. A avaliação segue o processo recomendado por Travassos et al. [Travassos, 2002] em “Introdução à Engenharia Experimental”, refinado por Mafra (2006) - vide artigo. A avaliação foi dividida em 3 experimentos, onde os procedimentos do processo foram aplicados. Como resultado, conclui-se que o método Eri*c contribui com 3 competências úteis e agrega detalhamento adequado em 5 competências em relação ao Framework i*.

Renovamos o convite aos pesquisadores para enviarem suas contribuições para os próximos números dos Cadernos do IME – Série Informática, para mantermos o mesmo padrão de qualidade obtido na presente edição.

Setembro de 2013

Alexandre Sztajnberg e Fabiano de Souza Oliveira
Organizadores do Volume 33

cadernos_inf@ime.uerj.br

Um Sistema Colaborativo de Apoio à Construção de Roteiros de Viagem

Daniel Ramos, Fellipe Ferrão, Rosa Maria Costa, Maria Clícia S. de Castro

Departamento de Informática e Ciência da Computação
Universidade do Estado do Rio de Janeiro (URJ)
Rua São Francisco Xavier, 524 – 20.559-900 – Rio de Janeiro – RJ – Brazil
rosamcosta@yahoo.com, clícia@ime.uerj.br

Abstract. *This paper presents a system that integrates GPS technology with a mashup that combines Google Maps with an application developed for mobile devices. This application enables the collaborative creation of travel itineraries, providing information related to places chosen by users. The Mashup informs the user's location in real time and displays information about their surroundings.*

Resumo. *Este artigo apresenta um sistema que integra a tecnologia GPS com um mashup que combina o Google Maps com uma aplicação desenvolvida para equipamentos móveis. Este aplicativo possibilita a criação colaborativa de roteiros de viagem, disponibilizando informações relacionadas aos lugares escolhidos pelos usuários. O Mashup informa a localização do usuário em tempo real e mostra informações referentes ao seu entorno.*

1. Introdução

A tecnologia móvel tem recebido uma atenção especial de pesquisadores e desenvolvedores. Em particular, a crescente difusão e popularidade de telefones celulares, amplia as possibilidades de comunicação, de entretenimento e de acesso a informações. A criação de aplicativos para celulares é uma área bastante promissora, pois contempla um grande número de usuários e cria mercados com perspectivas de altos lucros comerciais. Neste caso, os aplicativos se tornam mais elaborados, com tendências de compartilhamento e de fácil acesso às informações, caracterizados pelos novos direcionamentos estratégicos da Internet.

Neste contexto, a Web 2.0 se destaca por explorar um modelo interativo, que valoriza a inteligência coletiva, a participação colaborativa e a propagação rápida de conhecimento entre os usuários que interagem através da *web*. Ela tem sido considerada como a segunda geração de comunidades e serviços, que visa incentivar a criatividade e o compartilhamento [O'Reilly, 2005].

Visando integrar as possibilidades da Web 2.0 aos serviços móveis, este artigo apresenta um sistema que integra a tecnologia GPS (*Global Position System*) ao serviço de utilização de mapas do Google, para gerar um *mashup* que apóia usuários na definição e acompanhamento de roteiros de viagem. Esse sistema é denominado *Próxima Parada*. Ele informa a localização do usuário em tempo real e disponibiliza informações relacionadas ao entorno do local onde o usuário se encontra. Dessa forma, o usuário pode determinar pontos de parada, programando o roteiro de sua viagem.

Este artigo está organizado da seguinte forma: na Seção 2 descrevemos brevemente os conceitos e as tecnologias utilizados no sistema *Próxima Parada*. Na Seção 3 apresentamos uma visão geral do sistema. Na Seção 4 mostramos a especificação e o sistema desenvolvido. Finalizamos, apresentando alguns comentários.

2. Conceitos e Tecnologias

A tecnologia móvel tem exigido cada vez mais a utilização de diferentes ferramentas e tecnologias desenvolvidas especialmente para ela, devido as suas particularidades. Os celulares, na última década, se tornaram o meio de comunicação mais bem sucedido do mundo. Só no Brasil, são cerca de 175 milhões de acessos móveis (Anatel, 2010), e esse crescimento não tem sido muito diferente em outros países. Em nível mundial o celular tem aproximadamente 3 bilhões de usuários. As empresas têm percebido a necessidade de investir na padronização de aplicações e componentes, de modo a melhor aproveitar o potencial desses dispositivos.

A Web 2.0 [O'Reilly, 2005] pode ser entendida como um conjunto de princípios e práticas voltados para a Internet, tendo como características: ser uma plataforma de serviços, explorar uma arquitetura que enfatize a participação, ter potencial de escalabilidade, mesclar fontes de dados e de transformação de dados, disponibilizar software utilizável em vários tipos de dispositivos e aplicações que atuam como potencializadores da inteligência coletiva. A Web 2.0 baseia-se em um modelo interativo que valoriza a participação colaborativa e a propagação de conhecimento rápida entre os usuários que interagem através de *web sites*.

É neste contexto em que os mashups [Merril, 2006] se inserem. Eles podem ser considerados um tipo de aplicação da Web 2.0, uma vez que seguem os seus princípios. *Mashups* são serviços criados pela combinação de diferentes aplicativos para a Internet.

As aplicações de software que rodam em dispositivos móveis só se expandiram a partir da criação de sistemas operacionais específicos. O Android desenvolvido inicialmente pela Google e posteriormente, pelo grupo *The Open Handset Alliance* [Open Handset Alliance, 2007] é um sistema operacional móvel, sendo considerado como uma plataforma multitarefa, sendo adotado por várias empresas de telefonia celular. O Android é um ambiente que segue um padrão no qual desenvolvedores e criadores de hardware podem se adequar para gerar produtos que sejam compatíveis entre si. O desenvolvimento do Android foi baseado no Kernel 2.6 Linux. Ele é responsável pelo gerenciamento de memória, *threads*, redes e *drivers*, além de segurança. Esse gerenciamento permite o paralelismo das aplicações. Ele executa tanto os serviços em segundo plano, não visíveis pelos usuários, quanto os aplicativos acessados e utilizados pelos usuários. No desenvolvimento de aplicações, é utilizada a linguagem Java [Oracle, 2007], permitindo usufruir praticamente de todos os recursos que essa plataforma oferece.

Por outro lado, o sistema de posicionamento GPS (*Global Position System*) [Dana, 1997] utiliza satélites artificiais para definir uma localização geográfica baseado em dados de altitude, latitude e longitude. A partir das coordenadas de pelo menos 4 satélites, utilizando o mesmo sistema de referência entre eles e a antena do receptor, é possível calcular, a partir de suas distâncias, em que posição (latitude, longitude e

altitude) o receptor se encontra. O princípio de funcionamento do GPS é a recepção de sinais por um receptor, que pode estar presente em um aparelho celular, por exemplo.

Vários serviços baseados em localização (*Location Based Service* - LBS) são disponibilizados para telefonia móvel. O Yelp [Yelp, 2011] para celulares com sistema operacional Android oferece um guia de cidades (*online*) que auxilia pessoas a encontrar lugares para comer, comprar, beber e lazer. As informações são baseadas em opiniões informadas por uma comunidade. Sua funcionalidade é similar a proposta deste trabalho, pois é um software livre, mas difere porque não utiliza a tecnologia GPS.

Em [Kuhnen, 2003] é descrito um protótipo de uma aplicação LBS que identifica um estabelecimento a partir de uma localização informada, também usa a tecnologia GPS, mas não disponibiliza informações sobre os estabelecimentos que indica.

Estes conceitos são a base para o desenvolvimento de um aplicativo para celulares, que é descrito a seguir.

3. O Sistema Próxima Parada

Atualmente, uma parcela cada vez maior da sociedade recorre a Internet para obter mais conhecimento e informação, aumentando a demanda de sistemas eficazes e inteligentes, capazes de reunir em uma navegação rápida todo o conteúdo que necessitam. É com esse objetivo que o sistema proposto está estruturado. Ele possibilita que o usuário do sistema possa compartilhar informações com outros usuários. A idéia é compartilhar experiências de viagens, implementando o conceito de computação colaborativa. Todo o conteúdo do sistema é construído através da interação entre seus usuários, evoluindo e expandindo o sistema a cada vez que for difundido e utilizado.

A Figura 1 apresenta uma visão geral do sistema *Próxima Parada*.

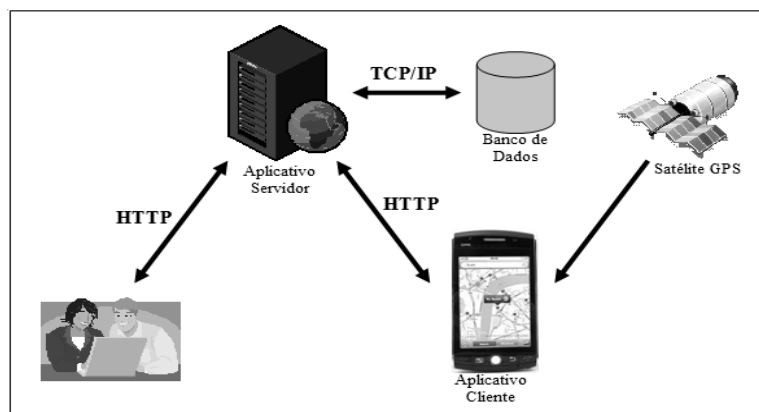


Figura 1. Visão geral do sistema *Próxima Parada*

Como pode ser observado na Figura 1, o sistema *Próxima Parada* possui basicamente dois módulos: Servidor e Cliente, que estão descritos a seguir.

3.1. Módulo Servidor (*mashup site*)

O módulo Servidor possibilita que os usuários cadastrem informações sobre suas viagens, destacando seus destinos e pontos de interesse. Para isso, o Servidor utiliza os mapas disponibilizados pelo *mashup site* do Google [Silva, 2009]. Após selecionar um destino, o usuário pode optar por cadastrar no mapa locais que julgou interessantes, ou selecionar pontos já cadastrados por outros usuários e inserir informações complementares. Tais pontos de interesse são cadastrados de acordo com o seu tipo, sendo identificados por uma imagem específica. Por exemplo, restaurantes, postos de gasolina, pontos turísticos, hotéis, entre outros.

Para cada ponto selecionado o usuário pode adicionar comentários, fornecendo críticas ou comentários, para que outros usuários possam obter essas informações a respeito desses locais.

3.2. Módulo Cliente

O módulo Cliente disponibiliza uma consulta rápida a pontos selecionados através do celular. Atualmente os celulares integram diversas funcionalidades, acesso a Internet e navegação através de GPS. Aproveitando essas facilidades, o módulo Cliente as combina para disponibilizar aos seus usuários consultas para definir roteiros de viagem e acesso às informações sobre os pontos escolhidos.

Através do GPS, o sistema identifica a posição do usuário, levanta todos os pontos adjacentes aos locais onde o usuário se encontra, ou planeja ir, os desenha na tela do celular, possibilitando a consulta de seu conteúdo.

O Cliente consome as informações através de um *Webservice* que faz parte do sistema Servidor (*mashup site*), permitindo que as informações estejam sempre sincronizadas com o que está ocorrendo na Web.

4. Especificação e Implementação do Sistema Próxima Parada

Esta seção descreve a documentação base para o desenvolvimento do sistema, demonstrando o seu fluxo de funcionamento e o mapeamento das suas funcionalidades. A seguir, são descritos os requisitos do sistema *Próxima Parada*, as especificações dos casos de uso e a sua implementação.

4.1. Requisitos Funcionais

O sistema atende aos seguintes requisitos funcionais: i) permitir que um usuário da Internet se cadastre no sistema e utilize seus serviços; ii) possibilitar ao usuário a manutenção de dados pessoais, bem como os dados de seus mapas, no sistema servidor; iii) permitir ao usuário tornar seus pontos públicos ou mantê-los privados; iv) localizar os mapas relativos às viagens ou aos pontos de interesse e exibir ou não, os pontos definidos como públicos; v) possibilitar ao usuário o acesso aos pontos cadastrados, através do sistema cliente do celular; e vi) permitir que determinados tipos de usuários atuem como administradores, removendo usuários e excluindo comentários indesejáveis.

4.2. Requisitos Não Funcionais

Os seguintes requisitos não funcionais estão presentes no sistema: segurança, desempenho e usabilidade, que estão descritos a seguir.

- **Segurança.** Os módulos devem garantir a confiabilidade e segurança das informações cadastradas pelos usuários, caso estes não desejem tornar seus dados públicos (dados pessoais e rotas de viagem).
- **Desempenho.** Considera o tempo de resposta. Neste caso, o módulo Cliente deve ter uma rápida atualização da localização através do GPS, para aumentar a confiabilidade do usuário.
- **Usabilidade.** A usabilidade está relacionada à interface gráfica, ela deve ser amigável, simples e acessível. O módulo Cliente, por estar disponível em aparelhos portáteis, deve possuir uma interface especialmente preparada para utilização em pequenas telas. É importante a utilização de fontes não muito pequenas e cores que sejam facilmente diferenciáveis uma das outras. Além de estar preparada para utilização em ambientes externos, de modo que diferentes luminosidades ou ruídos externos não influenciem a visualização das informações. O módulo Servidor deve possuir uma interface clara e intuitiva, de modo a permitir ao usuário utilizar o sistema sem necessitar de nenhum tipo de treinamento.

4.3. Diagramas de Caso de Uso

4.3.1 Atores

Para este modelo de casos de uso foram identificados três tipos de atores em nosso sistema: o Usuário, o Administrador e o Robô, que estão descritos a seguir.

Usuário. Localizado em ambos os sistemas Cliente e Servidor, o usuário do sistema é responsável pelo cadastro e consulta das informações relativas aos percursos das viagens. Ele pode estar em algum terminal conectado à Internet realizando tarefas como cadastro, consulta e manutenção das informações inseridas, ou pode estar em qualquer outro local, em que seja possível utilizar de seu celular com GPS e acesso à Internet, para acesso às informações do sistema. O sistema deve estar preparado para um grande número de usuários, da ordem de milhões, acessando o sistema simultaneamente, já que sua utilização por este ator se pode se tornar freqüente.

Administrador. Localizado no sistema Servidor, o administrador do sistema é responsável pela manutenção e moderação do sistema. Ele pode ser visto como um tipo avançado de usuário, pois tem o poder de excluir comentários e marcações de usuários e até mesmo para bloqueá-los. O número de administradores é pequeno, podendo ser menor do que 0,5% do total de usuários, além de sua utilização ser ocasional.

Robô. Localizado no sistema Cliente, o robô é o ator responsável pela atualização da localização dos atores nos mapas do sistema. Através da conexão com o GPS, ele informa ao sistema a coordenada física sobre o mapa onde se encontra o usuário no momento em que este estiver acessando o sistema através do celular. Sua interação com o sistema é constante. O número de robôs ativos é proporcional à quantidade de usuários conectados ao sistema através do celular.

4.3.2 Caso de Uso

A Figura 2 mostra o diagrama de caso de uso dos módulos Cliente e Servidor, com o intuito de descrever todas as ações do sistema atribuídas aos seus respectivos atores.

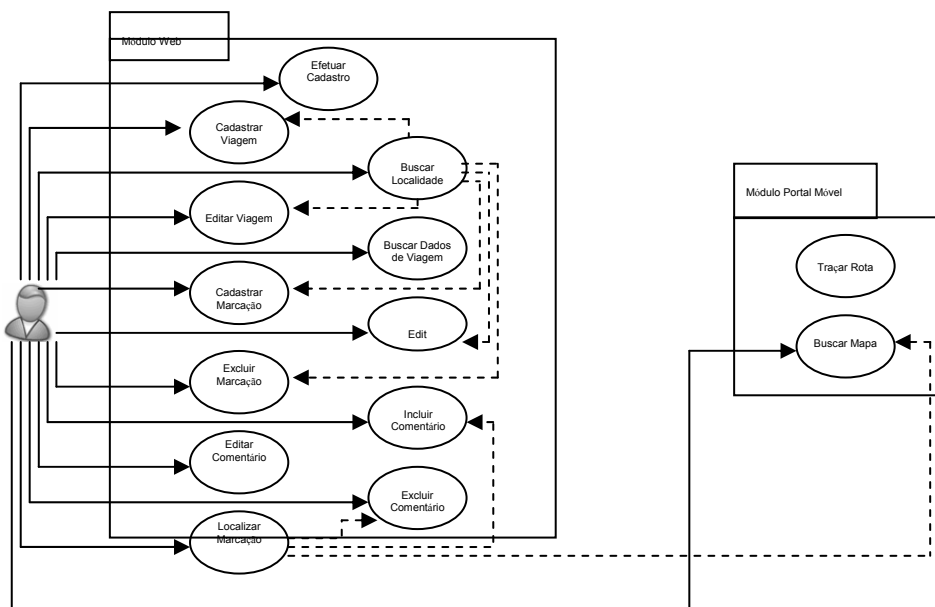


Figura 2. Diagrama de caso de uso do ator usuário

Como podemos observar, o ator usuário possui uma função essencial tanto no módulo Cliente, quanto no módulo Servidor, fornecendo todo o conteúdo necessário para que a integração dos serviços ocorra.

A Figura 3 mostra os diagramas dos atores robô e administrador, que dão suporte a todas as operações realizadas pelo usuário.

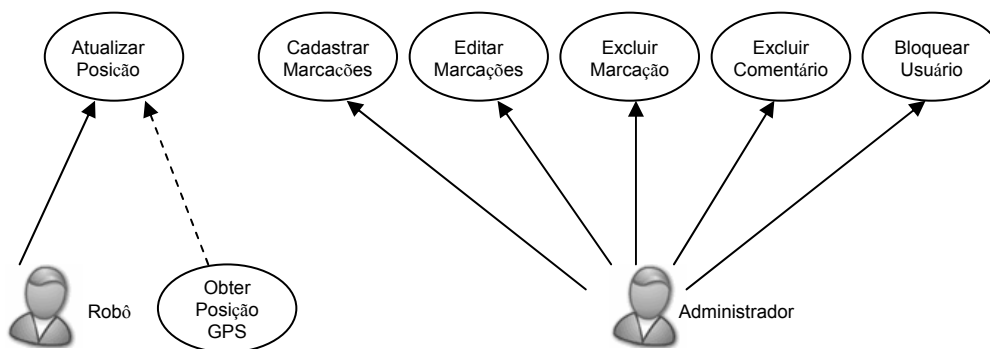


Figura 3. Diagrama de caso de uso dos atores administrador e robô

4.4. Implementação do Sistema

A implementação proposta utiliza dois diferentes aplicativos. O primeiro é um aplicativo Servidor *online*, responsável pelo cadastro e manutenção das marcações e informações referentes às rotas de viagem do usuário. Ele pode ser acessado através de

qualquer computador que possua uma conexão à Internet. O segundo aplicativo é um Cliente do aplicativo Servidor. Ele deve ser acessado pela Internet através do celular com GPS e exibe ao usuário as informações cadastradas no aplicativo Servidor, atualizadas de acordo com sua posição geográfica indicada pelo GPS, ou por sua interação direta.

Para utilizar o sistema *Próxima Parada* é necessário realizar o acesso ao sistema através de um *login* e uma senha. O usuário pode, então, buscar o mapa de uma determinada região, percorrer o mapa e marcar os pontos que acha interessante conhecer ou ter como referência, traçando assim um roteiro de viagem.

Nos pontos cadastrados o usuário pode, ao se encontrar em um local, utilizar seu celular com GPS para sua localização física em relação às marcações e pontos já cadastrados no sistema Servidor. Na medida em que ele se desloca em relação ao mapa, essas informações são atualizadas. É possível também, para uma determinada marcação no mapa, visualizar os comentários de outras pessoas em relação àquele ponto.

Os aplicativos foram desenvolvidos segundo o modelo de arquitetura MVC (*Model View Controller*) [Burbeck, 1992] em Java (Oracle, 2007) (versão jre_1.6.0_13), através da IDE Eclipse [Genuitec, 2003] por se tratar de uma linguagem gratuita bastante conhecida e com número crescente de usuários. Também são utilizados o Tomcat [The Apache Software Foundation, 1999] como Servidor Web (versão 6.0.20) e a *framework* Hibernate para a persistência de dados.

4.5. Classes

A seguir são citadas as principais classes referentes ao projeto do sistema *Próxima Parada*. É importante ressaltar que existem outras classes que também fazem parte do sistema, mas que não são apresentadas e que estão relacionadas ao suporte do ambiente.

FacesServlet. Classe interna ao *framework* JSF, responsável por coordenar todas as requisições feitas no *browser* de Internet, interpretando-as e selecionando a classe interna ao sistema responsável por tratar a requisição.

Usuário. Classe responsável por manipular e persistir no banco de dados, todas as atividades relacionadas ao usuário do sistema, dando suporte ao cadastro, controle de acesso e por manter o histórico de atividades no sistema.

Viagem. Classe responsável por manipular e persistir no banco de dados, todas as atividades relacionadas a viagens. Ela possui uma série de validadores para que o usuário possa obter através de fragmentos de endereço, uma localização geográfica precisa do local que está sendo pesquisado. Para isso, essa classe utiliza métodos de acesso sofisticados ao *mashup site* do Google Maps [Silva, 2009]. Essa classe também provê as posições geográficas necessárias para os posicionamentos dos pontos no mapa.

Ponto. Classe responsável por manipular e persistir no banco de dados, todos os pontos de interesse de uma determinada viagem. Ela possui métodos de integração com todas as outras classes do sistema, compondo uma série de validadores para eles.

Comentario. Classe responsável por manipular e persistir no banco de dados todos os comentários relacionados aos pontos de interesse de uma determinada viagem.

4.6. Descrição de Atividade e Diagramas de Sequência

4.6.1 Aplicativo Servidor

O aplicativo Servidor é disponibilizado como um portal de acesso na Internet e pode ser acessado por qualquer usuário cadastrado. Outra característica desse aplicativo é o suporte a *Webservice*, tornando possível a comunicação entre o Servidor e os aplicativos Clientes de maneira fácil, rápida e segura.

Nesse aplicativo além do cadastro das informações sobre as rotas de viagem, é utilizado o serviço oferecido pela Google Inc. para utilização de mapas (*mashup*), os quais, ao serem visualizados pelos usuários, fornecem a interface gráfica para a inclusão dos pontos do roteiro de viagem.

As marcações (pontos) e comentários inseridos no mapa, assim como as informações do usuário, ficam armazenados em um banco de dados. É importante ressaltar que somente as referências dos mapas utilizados, como posições geográficas, são armazenadas. Esses dados são utilizados para buscar a visualização dos mapas, serviço oferecido pelo Google. Não há armazenamento de imagens ou mapas pelo Servidor.

O fluxo de funcionamento principal do sistema *Próxima Parada* está exemplificado através a um cenário. Nesse cenário, o usuário cadastra determinadas localidades ou pontos (país, estado e cidade) de sua próxima viagem para que, posteriormente estando nessa cidade, ele possa se localizar em relação a elas.

Ao inserir corretamente suas informações para acesso ao sistema, é exibida a tela inicial. Caso o usuário tenha viagens anteriores cadastradas, ele visualiza um resumo de suas últimas viagens.

Uma vez cadastrada a viagem, o sistema utiliza os dados geográficos inseridos para carregar e exibir o mapa do local referenciado, onde o usuário pode cadastrar os pontos do mapa que farão parte do seu roteiro de viagem ou que poderão lhe servir de referência. Selecionando no mapa uma determinada localidade para ser ponto, a tela de Cadastro de Ponto é exibida para inclusão de informações e escolha da imagem relacionada àquele ponto, que é visualizada no mapa.

A Figura 4 ilustra a tela de visualização do mapa de viagem.

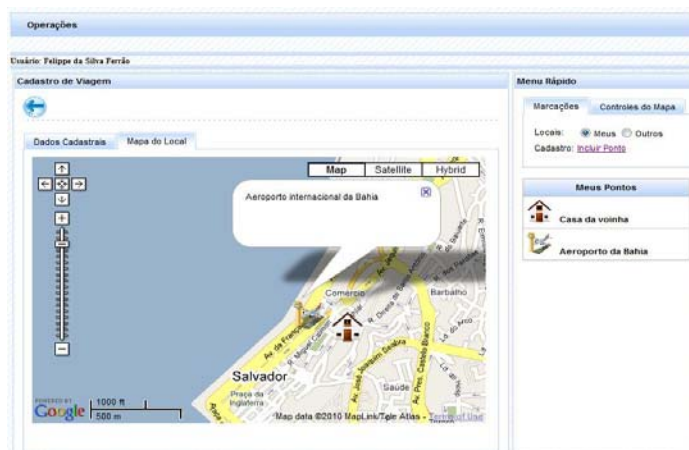


Figura 4. Tela de Visualização do Mapa da Viagem para a cidade de Salvador

Para cada ponto, o usuário tem a opção de inserir um comentário sobre aquele ponto, para auxiliar a sua consulta. O usuário pode repetir o ciclo de cadastro de pontos e colocar todas as referências que formarão o seu roteiro, com os devidos comentários.

Uma vez que todos seus pontos tenham sido cadastrados, o fluxo de cadastro de sua viagem está terminado.

4.6.2 Aplicativo Cliente

O aplicativo Cliente foi desenvolvido utilizando a tecnologia Android, permitindo que ele seja simples e robusto, características ideais para aplicativos que necessitam ser executados em celulares e PDAs.

Através do acesso ao aplicativo Cliente pelo celular, o usuário é capaz de buscar todas as suas viagens e seus respectivos pontos cadastrados no sistema Servidor. O aplicativo Cliente se comunica com o aplicativo Servidor através de *WebService*, disponibilizando todo o conteúdo *online* de forma interativa.

Supondo que o usuário se encontra na cidade cadastrada e através de seu celular com GPS, acesse o sistema para visualizar sua rota de viagem. Ele terá a viagem para Salvador que foi cadastrada anteriormente disponível para visualização, como mostrado na Figura 5. O mapa com seus pontos é exibido e o usuário consegue então se localizar em relação às suas rotas e referências registradas.

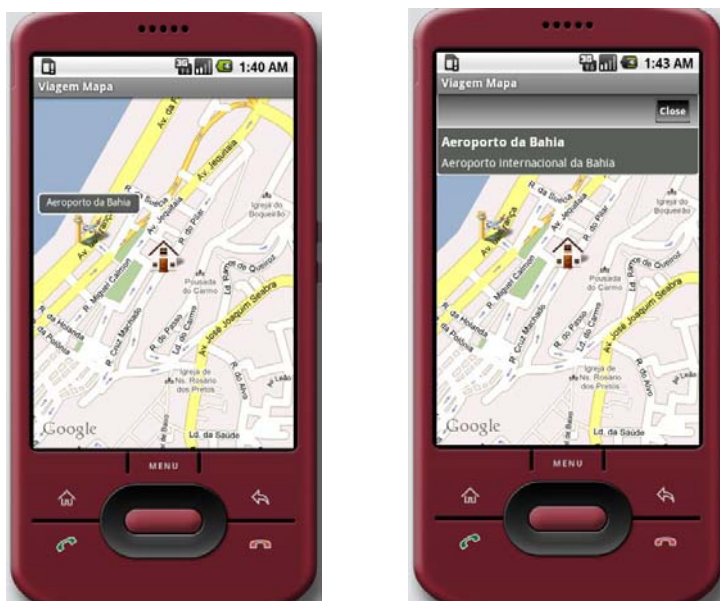


Figura 5. Tela de Seleção de Viagens, com os pontos cadastrados previamente, visualizados no aplicativo Cliente

Os diagramas de Sequência dos aplicativos Servidor e Cliente pode ser visto nas Figuras 6 e 7.

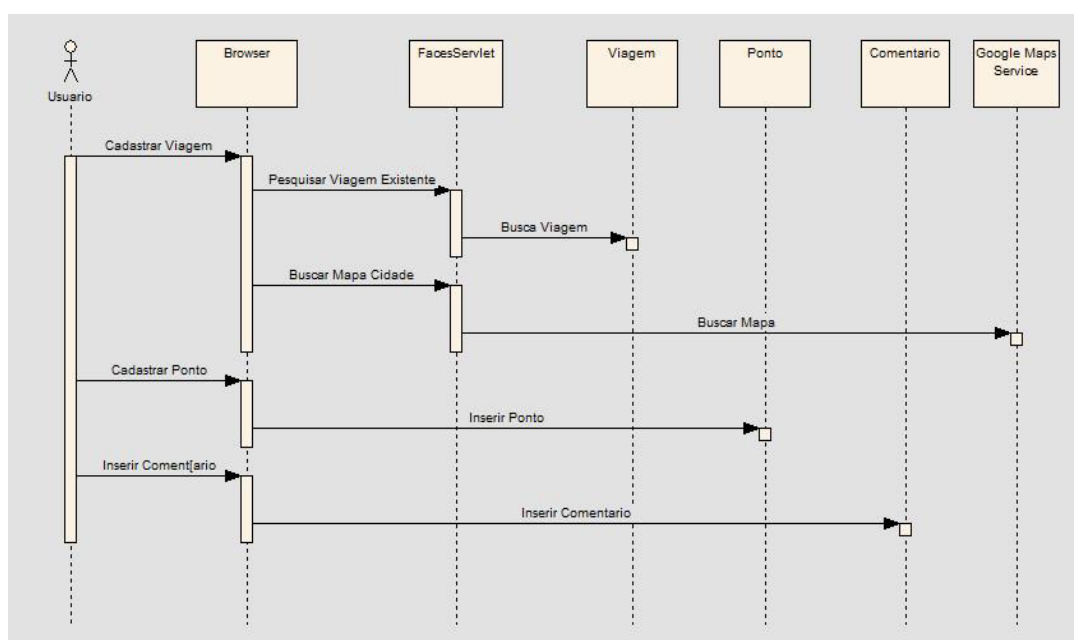


Figura 6. Diagrama de Seqüência do aplicativo Servidor

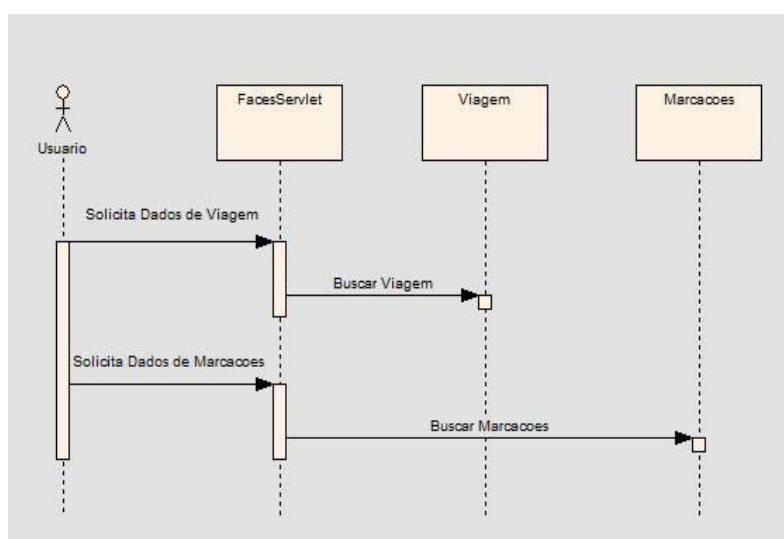


Figura 7. Diagrama de Seqüência do aplicativo Cliente

5. Conclusões

Este trabalho descreveu uma aplicação voltada para apoiar um usuário na construção e armazenamento de rotas de viagem, para posteriores consultas via celular. Por ser de fácil entendimento e utilização, não exigindo nenhum tipo de treinamento ou manual, esta aplicação poderia se difundir em diferentes contextos relacionados a viagens.

As maiores dificuldades encontradas no desenvolvimento do sistema relacionaram-se à integração das tecnologias utilizadas. No caso do Android, por ser uma tecnologia relativamente nova, ainda que haja documentação no *site* oficial do projeto, enfrentamos dificuldades para encontrar respostas para dúvidas sobre as limitações da plataforma. Alguns fóruns, *wikis* e grupos de discussão foram

fundamentais para a superação dos problemas, reforçando a idéia de colaboração, que caracteriza a fase atual da Internet.

A idéia central do sistema é oferecer meios para que o usuário acesse informações sobre lugares que vai visitar. Porém, esse sistema pode ser expandido para outros propósitos, visto que é possível visualizar pontos e comentários de uma região, cadastrados por outros usuários, viajantes, habitantes, comerciantes e empresários daquela localidade. O Sistema poderia até mesmo ser disponibilizado como um serviço para outro Sistema que necessite de dados de referência em mapas.

7. Referências

- Anatel. (2010) Brasil tem mais de 175 milhões de acessos móveis. Disponível em <http://www.anatel.gov.br/Portal/exibirPortalNoticias.do?acao=carregaNoticia&codigo=19695>, Acesso em agosto de 2010.
- Cornell, G. and Horstmann, C., (2000) Core Java 2: Fundamentos. Volume 1. Editora: Makron Books.
- Dana, P., (1997) Global Positioning System Overview. Disponível em http://www.colorado.edu/geography/gcraft/notes/gps/gps_f.html. Acesso em julho de 2009.
- Kuhnen, Alex, (2003) Protótipo de uma aplicação LBS utilizando GPS conectado em celular para consultas de dados
- Georeferenciados, (2003) Universidade Regional de Blumenau, Santa Catarina. Disponível [http://campeche.inf.furb.br/tccs/2003 II/2003-2alexkuhnenap.pdf](http://campeche.inf.furb.br/tccs/2003%20II/2003-2alexkuhnenap.pdf). Acesso em março de 2012.
- Merril, D., (2006) Mashups: The New Breed of Web Applications. Disponível <http://www.ibm.com/developerworks/xml/library/x-mashups/index.html>. Acesso em agosto de 2010.
- Open Handset Alliance, (2007) Disponível em <http://www.openhandsetalliance.com/index.html>. Acesso em agosto de 2009.
- Silva, L., (2009) Mashups com Google Maps API. Revista Java Magazine. Editora DevMedia. Edição 64.
- O'Reilly, T., 2005. What Is Web 2.0. Disponível em <http://oreilly.com/web2/archive/what-is-web-20.html>. Acesso em janeiro de 2011.
- The Apache Software Foundation, (1999) Disponível <http://tomcat.apache.org/>. Acesso em agosto de 2010.
- Genuitec, (2003) My Eclipse Infinite Possibilities. Disponível em <http://www.myeclipseide.com/htmlpages-func-display-pid-18.html?gclid=CPqM5ZD5xKkCFYfs7QodbVilWw>. Acesso em agosto de 2010.
- Oracle, (2007) The Java EE 5 Tutorial. Disponível em <http://download.oracle.com/javaee/5/tutorial/doc/bnadr.html>. Acesso em agosto de 2010.

Burbeck, S., (1992) Applications Programming in Smalltalk-80(TM): How to use Model-View-Controller (MVC). Disponível <http://st-www.cs.illinois.edu/users/smarch/st-docs/mvc.html>

Yelp, (2011) Disponível em [http:// www.yelp.com](http://www.yelp.com). Acesso em março de 2012.

Aplicando a Experimentação na Engenharia de Requisitos: O Caso do Método ERi*c

Antonio de Padua A. Oliveira¹, Julio Cesar Leite², Luiz Marcio Cysneiros³

¹ Departamento de Informática — Universidade do Estado do Rio de Janeiro (UERJ)
Rua São Francisco Xavier, 524 - Sala 6019 – CEP 20559-900 – RJ – Brazil

² Departamento de Informática – Pontifícia Universidade Católica do Rio de Janeiro (PUC-Rio) – Rua Marques de São Vicente, 225 – CEP 22453-900 – RJ – Brazil

³ School of Information Technology – York University, (YorkU)
Toronto – Ontario – Canada

padua@ime.uerj.br, julio@inf.puc-rio.br, cysneiro@yorku.ca

Abstract. *Some researches in Software Engineering said that only experiments in the centre of scientific process can verify new theories and point to needed corrections and to explore critical factor of these new theories. They also said that new methods, techniques and tools in Software Engineering should not be published without before be validated and compared with others that are working. This work shows three experiments application, which had the aim of evaluating ERi*c Method competencies. The Method ERi*c was developed in the doctoral thesis named Intentional Requirements Engineering: A Method for Requirements Elicitation, Modeling, and Analysis.*

Resumo. *Alguns pesquisadores em Engenharia de Software afirmam que somente os experimentos, no centro do processo científico, podem verificar novas teorias e, indicar as correções cabíveis e explorar os fatores críticos dessa nova teoria. Afirmam também que novos métodos, técnicas e ferramentas em Engenharia de Software não deveriam ser publicados sem antes serem experimentados, validados e comparados com os existentes. Este trabalho apresenta a aplicação de três experimentos, os quais tiveram com o objetivo avaliar as competências fornecidas pelo Método ERi*c da tese de doutorado intitulada: Engenharia de Requisitos Intencional: Um Método de Elicitação, Modelagem e Análise de Requisitos.*

1. Introdução

Segundo o trabalho “Introdução a Engenharia Experimental” [Travassos, 2002], para a Engenharia de Software existem quatro métodos relevantes com a finalidade de condução de experimentos. São eles: (1) o método científico que retira do ambiente observado um modelo para definir o conjunto de propriedades que devem ser analisadas; (2) o método da engenharia que estuda as soluções já utilizadas e aplica alguma suposta evolução na tentativa de achar uma vantagem adicional em relação às soluções iniciais; (3) o método analítico, ou matemático, que propõe uma teoria bem formalizada, obtém os resultados da aplicação da mesma e compara esses resultados com os resultados empíricos e (4) o método experimental que propõe e submete

repetidamente o novo método a situações para observação do comportamento com o objetivo de comprovação e aprimoramento.

Travassos [Travassos, 2002] afirma que somente experimentos, como o centro do processo científico, podem verificar novas teorias e, dessa maneira, indicar as correções cabíveis e explorar os fatores críticos dessa nova teoria. Travassos vai mais além quando afirma que novos métodos, técnicas e ferramentas em Engenharia de Software não deveriam ser publicados sem antes serem experimentados, validados e comparados com os existentes. Mas, complementando, Travassos ressalva que nenhum experimento pode levar a uma evidência absoluta e cita Basili [Basili, 1996], que apresenta com detalhes quais são os propósitos e as vantagens da experimentação na Engenharia de Software.

Nas próximas seções apresentamos a aplicação dos procedimentos da experimentação, segundo o processo recomendado por Travassos et al. [Travassos, 2002] em “Introdução à Engenharia Experimental”. Detalhamos os objetivos a serem atingidos, as questões a serem respondidas e as métricas utilizadas, de acordo com o método GQM – “Goal, Question, Metric” [Basili, 1994]. Também detalhamos o planejamento do processo e ao final apresentamos a operacionalização do processo através dos questionários que o apoiaram. A última seção apresenta um resumo dos resultados obtidos.

2. Definição Sucinta do Processo de Experimentação

Para avaliar e validar o método Engenharia de Requisitos Intencional - ERi*c [Oliveira 08a] foi aplicado o “Processo de Experimentação em Engenharia de Software” obtido em [Mafrá, 2006], que foi definido por Wohlin et al. (2000) e estendido por Amaral (2003). O Processo de Experimentação possui dois subprocessos, o Processo de Execução e o Processo de Empacotamento.

As finalidades das etapas do *Processo de Experimentação* são apresentadas a seguir. A *etapa de definição* vai expressar o experimento em termos de objetivos e problemas; a *etapa de planejamento* vai: determinar o projeto do experimento, definir a instrumentação a ser utilizada e analisar os aspectos de validação dos resultados. A *etapa de execução* vai cuidar da coleta dos dados do experimento. E a etapa de *análise e interpretação* vai estudar detalhadamente os dados coletados. O processo possui dois pontos de controle, o primeiro existe para avaliar a necessidade de replanejar e o segundo existe para avaliar a efetividade da etapa de análise dos resultados. O *Processo de Empacotamento* faz a atividade de organizar e guardar as informações a respeito do *Processo de Execução*.

3. Os Estudos da Experimentação

A motivação envolvida pelo processo de experimentação que executamos foi guiada pela idéia de avaliar se o Método ERi*c ajuda as competências do Engenheiro de Requisitos (ER).

Para a avaliação das competências fornecidas precisaríamos comparar o Método ERi*c com o **Framework i*** [Yu, 1995] já que o Método ERi*c se propõe a suprir a lacuna da elicitação de metas que ainda não está equacionado pelos métodos GORE, inclusive pelo Framework i*, como também contribuir para a redução dos problemas de

escalabilidade do **Framework i*** [Yu, 1995] relatados por Pastor et al. [Pastor, 2006] e por Estrada et al. [Estrada, 2006]. Conseqüentemente, seguindo o processo de experimentação de Mafra e Travassos [Mafra, 2006] nossa etapa de planejamento previa a realização de duas experimentações. A primeira devia avaliar se as competências atribuídas ao **Framework i*** eram úteis e se estavam adequadamente definidas segundo o ponto de vista de engenheiros de requisitos em relação às competências de métodos já dominados, ou já ensinados como Análise Essencial e Orientação a Objetos. A segunda devia comparar as competências fornecidas pelo Método ERi*c com as competências atribuídas pelo **Framework i*** [Yu, 1995]. Porém, nosso planejamento inicial foi alterado, incluímos mais uma experimentação. Influenciados pela necessidade do “pôster” [Oliveira, 2008b] aceito no CAISE 08, avaliamos separadamente se a estratégia de diagnósticos “i* Diagnoses” trazia competências úteis, e com a definição adequada, em relação às etapas anteriores do próprio Método ERi*c.

O objetivo da experimentação foi único. Mas, a experimentação foi desmembrada em três para que a avaliação dividida pudesse trazer, apesar do trabalho triplo, mais retorno de informação para a tese de [Oliveira, 2008a]. A próxima seção apresenta as etapas do estudo das competências e sua aplicação nos três experimentos realizados.

4. O Estudo das Competências do Método ERi*c e do Framework i*

I ETAPA DE DEFINIÇÃO

I.1 Objetivo global

O objetivo global da experimentação do método Engenharia de Requisitos Intencional - ERi*c é: *“Estabelecer quais são as vantagens fornecidas pelo Método ERi*c em relação ao Framework i*, em termos de competências para a Engenharia de Requisitos de Sistemas Multi-Agentes, na elicitação, na modelagem e na análise dos requisitos”*.

I.2 Objetivos da medição

Os objetivos da medição, guiados pelo processo, devem identificar:

- (i) Quais são as competências adquiridas?
- (ii) Quais são as competências consideradas úteis?
- (iii) Quais são as competências consideradas inúteis?
- (iv) Quais são as competências em potencial?

I.3 Objetivos do estudo

Identificar se o método proposto Engenharia de Requisitos Intencional - ERi*c, contribui para melhorar as “competências” atribuídas à Engenharia de Requisitos de Sistemas Multi-Agentes do ponto de vista dos engenheiros de requisitos.

I.4 Questões e métricas

Com a idéia de aplicar a abordagem GQM [Basili, 1994] para a avaliação do objetivo global definido no parágrafo 1.1 anterior definimos nos parágrafos a seguir os objetivos (Goals), as perguntas investigadas (Questions) e as métricas (Metrics) da abordagem.

GOAL: “Estabelecer quais são as vantagens fornecidas pelo Método ERi*c em relação ao Framework i*, em termos de competências para a Engenharia de Requisitos de Sistemas Multi-Agentes”.

Q1: Quais competências são fornecidas pelo Método ERi*c e não fazem parte do Framework i*?

⇒ **Métrica:** Lista das competências fornecidas pelo Método ERi*c que não fazem parte do Framework i*.

Q2: Quais competências são fornecidas pelo Método ERi*c e são consideradas inúteis pelos engenheiros de requisitos?

⇒ **Métrica:** Lista das competências fornecidas pelo Método ERi*c e que são consideradas inúteis pelos engenheiros de requisitos.

Q3: Quais competências são fornecidas pelo Método ERi*c e fazem parte do Framework i* e são consideradas úteis pelos engenheiros de requisitos, cujo detalhamento deve ser modificado?

⇒ **Métrica:** Lista das competências fornecidas pelo Método ERi*c que fazem parte do Framework i* e são consideradas úteis pelos engenheiros de requisitos, cujo detalhamento deve ser modificado.

Q4: Quais competências são fornecidas pelo Método ERi*c e não fazem parte do Framework i*, mas que são consideradas úteis pelos engenheiros de requisitos?

⇒ **Métrica:** Lista das competências fornecidas pelo Método ERi*c que não fazem parte do Framework i*, mas que são consideradas úteis pelos engenheiros de requisitos.

II ETAPA DE PLANEJAMENTO

II.1 Definição das hipóteses

A HIPÓTESE, para acompanhar o desmembramento da experimentação, teve que ser também desmembrada e, assim, enunciada em separado para cada estudo executado. A INSTRUMENTAÇÃO e a SELEÇÃO DO CONTEXTO dos experimentos foram definidas apenas uma vez porque elas não sofreram mudança. Por outro lado, a DEFINIÇÃO DAS VARIÁVEIS dos experimentos precisou ser redefinida para cada um dos experimentos.

II.1.1 Definição das hipóteses para a experimentação 1

Hipótese nula (H_0): As competências fornecidas pelo método Framework i^* são equivalentes às competências anteriormente assimiladas.

C1 - conjunto de competências fornecidas pelo Framework i^*

C0 - conjunto de competências anteriormente assimiladas

$$H_0: C0 - (C1 - C0) = \emptyset$$

Significa que não existem competências fornecidas pelo Framework i^* que não fazem parte das competências anteriormente assimiladas.

Hipótese alternativa (H_1): As competências fornecidas pelo Framework i^* não são equivalentes às competências anteriormente assimiladas.

C1 - conjunto de competências fornecidas pelo Framework i^*

C0 - conjunto de competências anteriormente assimiladas

$$H1: C0 - (C1 - C0) \neq \emptyset$$

Significa que existem competências fornecidas pelo Framework i^* que não fazem parte das competências anteriormente assimiladas.

Hipótese alternativa (H_2): Na lista das competências fornecidas pelo Framework i^* e que não fazem parte das competências anteriormente assimiladas existem competências que são consideradas úteis pelos engenheiros de requisitos, cujo detalhamento deve ser modificado.

Cx - conjunto de competências fornecidas pelo Framework i^* e que não fazem parte das competências anteriormente assimiladas.

Cy - conjunto de competências que são consideradas úteis pelos engenheiros de requisitos, cujo detalhamento deve ser modificado.

$$H2: Cy - (Cx - Cy) \neq \emptyset$$

Significa que existem competências fornecidas pelo Framework i^* que não fazem parte das anteriormente assimiladas, mas são consideradas úteis pelos engenheiros de requisitos e cujo detalhamento deve ser modificado.

II.1.2 Definição das hipóteses para a experimentação 2

Hipótese nula (H_0): As competências fornecidas pelo Método ERi*c são equivalentes às competências fornecidas pelo Framework i^* .

C1 - conjunto de competências fornecidas pelo Método ERi*c

C0 - conjunto de competências fornecidas pelo Framework i^*

$$H0: C0 - (C1 - C0) = \emptyset$$

Significa que não existem competências fornecidas pelo Método ERi*c que não fazem parte do Framework i*.

Hipótese alternativa (H₁): As competências fornecidas pelo Método ERi*c não são equivalentes as competências fornecidas pelo Framework i*.

C1 - conjunto de competências fornecidas pelo Método ERi*c

C0 - conjunto de competências fornecidas pelo Framework i*

H₁: C0 – (C1 – C0) ≠ ∅

Significa que existem competências fornecidas pelo Método ERi*c que não fazem parte das competências fornecidas pelo Framework i*.

Hipótese alternativa (H₂): Na lista das competências fornecidas pelo Método ERi*c e que não fazem parte das competências fornecidas pelo Framework i* existem competências que são consideradas úteis pelos engenheiros de requisitos, cujo detalhamento deve ser modificado.

Cx - conjunto de competências fornecidas pelo Método ERi*c e que não fazem parte das competências fornecidas pelo Framework i*.

Cy - conjunto de competências que são consideradas úteis pelos engenheiros de requisitos, cujo detalhamento deve ser modificado.

H₂: Cy – (Cx – Cy) ≠ ∅

Significa que existem competências fornecidas pelo Método ERi*c que não fazem parte do Framework i*, mas são consideradas úteis pelos engenheiros de requisitos, cujo detalhamento deve ser modificado.

Cabe ressaltar que essa avaliação está considerando o Método ERi*c sem a etapa de diagnósticos porque a mesma não tinha sido apresentada aos participantes no treinamento do Método ERi*c.

II.2 Descrição da instrumentação

Em cada experimento estudado e para cada competência fornecida, foram oferecidas as escolhas definidas a seguir.

Resultado: N competências com valores (P; U; A) onde:

P – PRESENÇA {0 → não fornecida; 1 → fornecida}

U – UTILIDADE {0 → não útil; 1 → útil}

A – ADEQUAÇÃO {0 → sem modificação; 1 → com modificação}

II.3 Seleção do contexto

O contexto é caracterizado por quatro dimensões. Nosso estudo teve o processo on-line porque os participantes responderam ao questionário da entrevista ainda durante o período do curso na universidade, no transcurso da experimentação. Os participantes

foram todos **alunos**. O estudo é considerado **modelado** porque os alunos aplicaram os métodos em um problema de uma empresa de seguros fictícia. A dimensão de generalidade é considerada específica, pois a experimentação foi sobre o contexto **específico** de uma modelagem de sistema orientado a metas.

II.4 Seleção dos participantes

Como participantes da experimentação tivemos alunos da graduação da UERJ e alunos da graduação da PUC-Rio da área de Engenharia de Software. Os alunos foram consultados e avisados que se tratava de uma experimentação científica. Somente aqueles que concordaram em participar do experimento foram incluídos na pesquisa.

No primeiro experimento participaram alunos de duas turmas da UERJ e de uma turma da PUC-Rio. Não houve distinção de qual turma o aluno pertencia, eles foram considerados como um único grupo de participantes. No segundo e no terceiro experimentos participaram exclusivamente alunos de uma turma da UERJ, somente para esta turma foi ensinado o Método ERi*c.

II.5 Definição das variáveis

II.5.1 Definição das variáveis para a experimentação 1

As **variáveis independentes** se referem à entrada da experimentação, são elas:

- a lista das competências fornecidas pelo Framework i*.

As **variáveis dependentes** se referem à saída da experimentação, são elas:

- o grau da similaridade entre as competências fornecidas pelo Framework i* e as competências anteriormente assimiladas;
- o grau da utilidade das competências fornecidas pelo Framework i* e as competências anteriormente assimiladas;
- o grau de adequação das competências fornecidas pelo Framework i*.

II.5.2 Definição das variáveis para a experimentação 2

As **variáveis independentes** se referem à entrada da experimentação, são elas:

- a lista das competências fornecidas pelo Método ERi*c.

As **variáveis dependentes** se referem à saída da experimentação, são elas:

- o grau da similaridade entre as competências fornecidas pelo Método ERi*c e as atribuídas ao Framework i*;
- o grau da utilidade das competências fornecidas pelo Método ERi*c e as competências atribuídas ao Framework i*;
- o grau de adequação das competências fornecidas pelo Método ERi*c.

III ETAPAS DE EXECUÇÃO E DE ANÁLISE

Os experimentos foram executados em seqüência, eles aconteceram em períodos distintos. Todos os treinamentos foram exatamente iguais, foram ministrados pelo mesmo professor, com o mesmo material didático e com a mesma carga horária. Cada experimentação teve um questionário específico conforme está apresentado nas seções seguintes.

III.1 Operacionalização e análise da experimentação 1

III.1.1 - Questionários das Competências e do Perfil do Participante

O questionário completo da experimentação do Framework i* encontra-se em [Oliveira, 2008a] para a “Avaliação da Aplicação do Framework i* - Caso da Seguradora Imperial”. Uma descrição resumida do conteúdo das competências é apresentada no Quadro 1.

Quadro 1. Quadro resumo das competencias avaliadas na Experimentação 1

	COMPETÊNCIA	DESCRIÇÃO
A	Identificação de atores	Com a utilização do i* você é capaz de identificar os atores de uma organização.
B	Identificação de dependências	Com a utilização do i* você é capaz de identificar as dependências dos atores de uma organização.
C	Identificação de oportunidades	Com a utilização do i* você é capaz de identificar as oportunidades dos atores de uma organização.
D	Elicitação de metas concretas	O i* é capaz de descobrir as metas concretas dos atores de uma organização.
E	Elicitação de metas flexíveis	O i* é capaz de descobrir as metas flexíveis dos atores de uma organização.
F	Representação de metas	O i* é capaz de representar as metas concretas e as metas flexíveis dos atores.
G	Identificação de agentes, posições e papéis	O i* é capaz de identificar os agentes, as posições e os papéis dos atores.
H	Representação de agentes, posições e papéis	O i* é capaz de representar nos modelos os agentes, as posições e os papéis dos atores.
I	Análise de variabilidades	O i* é capaz de fazer análise de alternativas para a fase de desenho (design) do software.
J	Níveis de detalhamento	O i* permite o engenheiro de requisitos fazer modelos com níveis de detalhamento .
K	Simplificação da complexidade	O i* permite o engenheiro de requisitos fazer modelos de complexidade simples .
L	Utilização de blocos construtores	O i* permite nos modelos SD e SR construir blocos construtores .
M	Guia para elaboração dos modelos	O i* possui um guia para orientar o engenheiro de requisitos na elaboração dos modelos.

III.1.2 Resultados do Estudo

No Quadro 2 são listados os resultados dos questionários sem tratamento estatístico. Foram recebidas 17 respostas de participantes para a experimentação 1.

Quadro 2. Resumo dos Resultados do Questionário da Experimentação 1

Competência		P	U	A
A	Identificação de atores	3:14	0:17	4:13
B	Identificação de dependências	9:8	0:17	6:11
C	Identificação de oportunidades	17:0	0:17	13:4
D	Elicitação de metas concretas	9:8	0:17	9:8
E	Elicitação de metas flexíveis	5:12	0:17	10:7
F	Representação das metas	5:12	0:17	5:12
G	Identificação de agentes, papéis e posições	6:11	0:17	9:8
H	Representação de agentes, papéis e posições.	7:10	0:17	11:6
I	Análise de variabilidades	15:2	0:17	12:5
J	Níveis de detalhamento	11:6	0:17	10:7
K	Simplificação da complexidade	10:7	0:17	7:10
L	Utilização de blocos construtores	13:4	0:17	11:6
M	Guia para elaboração dos modelos	13:4	0:17	11:6

Uma análise superficial das respostas do Quadro 2, que agrupou as respostas como **n=negativa : p=positiva**, nos permite verificar os problemas:

- Seis competências [C], [I], [J], [K], [L] e [M] tiveram claramente o reconhecimento da presença “P” contestado pelos participantes. Uma sinalização importante para essas competências foi o reconhecimento unânime da utilidade delas pelos entrevistados, apesar das mesmas não terem a presença reconhecida pela maioria.
- O atributo “A” adequação que aborda o detalhamento de como as competências são tratadas pelo Framework i* mostra que as respostas não possuem predominância positiva ou negativa. Duas possibilidades: ou o treinamento foi insuficiente ou o Framework i* não fornece detalhes para as competências de interesse.
- Ainda para as competências [I] e [L], o atributo “A” mostra que o número de participantes que reclamaram do detalhamento é maior que o número de participantes que reconheceram a presença “P” da competência. A análise das respostas também pode indicar que as perguntas foram mal formuladas ou mal explicadas, pois as respostas ficaram sem sentido. São os casos das competências: [I] *Análise de variabilidades* e [L] *Utilização de blocos construtores*.

O tempo de aprendizado do treinamento foi de 4 horas, três aulas de aproximadamente 80 minutos. Ele foi considerado insuficiente pelos participantes. A maioria dos participantes precisou de pelo menos mais 3 horas estudando o material fornecido.

III.1.3 Análise e Interpretação dos Resultados da Experimentação 1

III.1.3.1 Validação dos Dados da Experimentação 1

Não foram observadas respostas não válidas, as quais estariam erradas do ponto de vista dos valores lógicos válidos. Por exemplo, a resposta concomitante de negação da

presença de alguma competência com a negação da utilidade da competência em conjunto com a indicação da necessidade de modificação da competência não ocorreu.

III.1.3.2 Aplicação do teste estatístico

Para cada competência foi aplicado o teste qui-quadrado para definir:

- se pode ser considerado que a competência é fornecida;
- se pode ser considerado que a competência é útil;
- se pode ser considerado que a competência não necessita detalhamento.

A matriz do Quadro 3 apresenta o comportamento fictício para **p** respostas positivas e **n** respostas negativas enquanto a distribuição ideal aconteceria sem a ocorrência de nenhuma resposta negativa.

Quadro 3. Comportamento fictício

DISTRIBUIÇÃO	RESPOSTA		TOTAL
	positiva	negativa	
IDEAL	17	0	17
REAL	p	n	17
TOTAL	17+p	n	34

Determinação do grau de liberdade (d. f.): Grau de liberdade representa o número de variáveis independentes que fazem parte do estudo. Como os dois grupos, ou as duas variáveis, em estudo (o ideal e o real) são independentes, o grau de liberdade (d. f.) é 2 (dois). Para o nível de significância de 0,05, que representa a probabilidade de aceitar a hipótese nula (H_0) quando ela é falsa, obtemos da tabela de valores críticos de qui-quadrado [Box 78]. Valor crítico = (d.f. = 2, α = 0,05) = **5,991**.

III.1.3.3 Memória de Cálculo dos Valores do qui-quadrado

Na matriz do Quadro 4, o_{ij} são as frequências obtidas nas respostas do questionário, e_{ij} são as frequências esperadas e o_{it} é o total na linha i .

Quadro 4. Frequências obtidas nas respostas

DISTRIBUIÇÃO	RESPOSTA				TOTAL
	positiva		negativa		
IDEAL	o11	e11	o12	e12	o1t
REAL	o21	e21	o22	e22	o2t
TOTAL	ot1		ot2		T

Frequências esperadas: $e_{ij} = (\text{total da linha } i \times \text{total da coluna } j) / T$.

Cálculo aplicando o qui-quadrado para a observação (**5:12**)

$$e_{11} = (29 \times 17) / 34 = 14,5 \quad e_{12} = (5 \times 17) / 34 = 2,5$$

$$e_{21} = (29 \times 12) / 34 = 10,29 \quad e_{22} = (5 \times 17) / 34 = 2,5$$

$$\begin{aligned}\text{qui-quadrado} &= \sum (o_{ij} - e_{ij})^2 / e_{ij} = \\ &= (17 - 14,5)^2 / 14,5 + (12 - 14,5)^2 / 14,5 + (0 - 2,5)^2 / 2,5 + (5 - 2,5)^2 / 2,5 = \\ &= 0,43 + 0,43 + 2,5 + 2,5 = \underline{\underline{5,86}}\end{aligned}$$

Como o valor calculado (**5:12**) **5,86** é menor que o valor crítico **5,99** → **H₀** não pode ser rejeitada. O que significa que a competência fornecida pelo Framework **i*** não é equivalente à competência anteriormente assimilada.

Para qualquer observação (**n:p**) com vantagem superior a calculada anteriormente (**5:12**), ou seja maior número de respostas positivas **p** e conseqüentemente menor número de respostas negativas **n**, podemos, sem calcular, concluir que aquelas observações também não podem rejeitar a hipótese nula, porque todas certamente têm o valor do qui-quadrado calculado menores ainda que o valor crítico. São elas: (**4:13**), (**3:14**), (**2:15**), (**1:16**) e (**0:17**).

Para comprovar que o valor receberia um acréscimo e para aplicar o qui-quadrado com uma fórmula simplificada, fizemos o cálculo para a observação (**6:11**), a qual é menos favorável que a observação (**5:12**): Usando a formula simplificada abaixo:

$$\text{Qui-quadrado} = [(o_{11} \times o_{22}) - (o_{12} \times o_{21})]^2 \times T / (o_{.1} \times o_{.2} \times o_{1.} \times o_{2.})$$

$$\text{Qui-quadrado} = ((17 \times 6) - (11 \times 0))^2 \times 34 / (28 \times 6 \times 17 \times 17) = \underline{\underline{7,28}}$$

Distribuição	(- 7 : + 10)	(- 6 : + 11)	(- 5 : + 12)	(- 4 : + 13)	(- 3 : + 14)
Valor qui-2		7,28	5,86		

Valor crítico = **5,99**

III.1.3.4 Análise dos Resultados da Experimentação

A idéia da aplicação do teste qui-quadrado consiste na comparação da distribuição das respostas dos participantes com uma distribuição ideal. A distribuição ideal seria aquela que não recebe respostas negativas. Ou seja, a competência é recebida, é útil e não precisa de alteração no detalhamento. Essa distribuição ideal teria o comportamento representado pela expressão: {1, 1, 0}

Para nosso estudo estamos considerando o Framework **i*** como ideal e a comparação se faz com o real que no caso são as competências consideradas antes do treinamento do **i***.

As lacunas do Quadro 5 foram preenchidas com o valor “1” quando o resultado da aplicação do teste qui-quadrado não pode rejeitar a hipótese nula e com “0” no caso contrário.

Dentre as 13 competências investigadas observamos que 3 estão reconhecidamente presente no Framework **i***, são elas A, E e F. Todas as 3 competências foram consideradas úteis e 2 dessas competências tiveram o detalhamento da competência considerado adequado pelo experimento, são elas A e F.

Quadro 5. Resultado das Estatísticas da Experimentação 1

	Competências												
	A	B	C	D	E	F	G	H	I	J	K	L	M
Presença	1	0	0	0	1	1	0	0	0	0	0	0	0
Utilidade	1	1	1	1	1	1	1	1	1	1	1	1	1
Adequação	1	0	0	0	0	1	0	0	0	0	0	0	0

Como nosso objetivo do estudo da experimentação é avaliar se o Método ERi*c promove vantagens de competências em relação ao Framework i* deixamos para fazer as análises quantitativas e qualitativas após a aplicação do experimento 2.

III.2 Operacionalização e análise da experimentação 2

O Estudo das Competências do Método ERi*c

III.2.1 Questionários das Competências e do Perfil do Participante

O questionário da experimentação [Oliveira 08a] foi essencialmente o mesmo que foi aplicado na experimentação 1.

III.2.2 Resultados do Estudo

A seguir (Quadro 6) são apresentados os resultados do questionário. Foram recebidas 9 respostas dos participantes da experimentação 2.

Quadro 6. Resumo dos Resultados do Questionário da Experimentação 2

Competência		P	U	A
A	Identificação de atores	1:8	0:9	1:8
B	Identificação de dependências	4:5	0:9	2:7
C	Identificação de oportunidades	9:0	2:7	7:2
D	Elicitação de metas concretas	0:9	0:9	2:7
E	Elicitação de metas flexíveis	3:6	0:9	3:6
F	Representação das metas	1:8	0:9	3:6
G	Identificação de agentes, papéis e posições	7:2	1:8	8:1
H	Representação de agentes, papéis e posições	3:6	0:9	4:5
I	Análise de variabilidades	6:3	5:4	9:0
J	Níveis de detalhamento	6:3	0:9	3:6
K	Simplificação da complexidade	3:6	3:6	3:6
L	Utilização de blocos construtores	7:2	0:9	4:5
M	Guia para elaboração dos modelos	6:3	0:9	7:2

Uma análise superficial das respostas, antes da análise estatística, nos permite verificar pontos que deveriam ser melhorados no Método ERi*c e por isso demandaram alterações no detalhamento das heurísticas no método.

Outras respostas também chamaram a nossa atenção:

- A maioria dos participantes não reconheceu que o método favorece a [L] Utilização de blocos construtores apesar da unanimidade da resposta pela utilidade “U” dessa competência.
- A análise das respostas também permite verificar que algumas perguntas foram mal formuladas ou o treinamento não foi satisfatório, pois as respostas ficaram sem sentido para os atributos “P” e “U”. São os casos das competências: [I] Análise de variabilidades teve a utilidade “U” rejeitada e [L] Utilização de blocos construtores não teve a presença “P” reconhecida.

A análise do perfil dos participantes ressaltou que o treinamento de 10 horas e 40 minutos, 8 aulas de aproximadamente 80 minutos, foi considerado insuficiente pelos participantes.

III.2.3 Análise e Interpretação dos Resultados da Experimentação 2

III.2.3.1 Validação dos Dados da Experimentação 2

Não foram observadas respostas não válidas as quais estariam erradas do ponto de vista dos valores lógicos válidos. Por exemplo, a resposta concomitante de negação da presença de alguma competência com a negação da utilidade da competência em conjunto com a indicação da necessidade de modificação da competência não ocorreu.

III.2.3.2 Aplicação do teste estatístico

Para cada competência foi aplicado o teste qui-quadrado para definir:

- se pode ser considerado que a competência é fornecida;
- se pode ser considerado que a competência é útil;
- se pode ser considerado que a competência não necessita detalhamento.

A pergunta a ser respondida pelo teste qui-quadrado é a seguinte: A partir de que valores (**n= negativo : p= positivo**) os resultados podem ser considerados como ideais para a distribuição com probabilidade α (nível de significância) de erro?

Para o nível de significância de 0.05, que representa a probabilidade de aceitar a hipótese nula (H_0) quando ela é falsa, obtemos da tabela de valores críticos de qui-quadrado [Box 78]. Valor crítico = (d.f. = 1, α = 0.05) = **3.84**.

H_0 : Existe diferença em relação a competência fornecida pelo Método ERi*c e a competência anteriormente assimilada no Framework i*. O grau de liberdade foi considerado 1 porque o Método ERi*c é dependente do Framework i*.

Aplicando o qui-quadrado para a observação **4:5** e usando a fórmula simplificada:

9	0	9
5	4	9
14	4	18

$$\text{qui-quadrado} = [(o_{11} \times o_{22}) - (o_{12} \times o_{21})]^2 \times T / (o_{.1} \times o_{.2} \times o_{1.} \times o_{2.})$$

$$\text{qui-quadrado} = ((9 \times 4) - (5 \times 0))^2 \times 18 / (14 \times 4 \times 9 \times 9) = \underline{\underline{5,14}}$$

Aplicando o qui-quadrado para a observação **3:6**

$$\text{Qui-quadrado} = ((9 \times 3) - (6 \times 0))^2 \times 18 / (15 \times 3 \times 9 \times 9) = \underline{\underline{3,60}}$$

Aplicando o qui-quadrado para a observação **2:7**

$$\text{Qui-quadrado} = ((9 \times 2) - (7 \times 0))^2 \times 18 / (16 \times 2 \times 9 \times 9) = \underline{\underline{2,25}}$$

Distribuição	(- 5 : + 4)	(- 4 : + 5)	(- 3 : + 6)	(- 2 : + 7)	(- 1 : + 8)
Valor qui-2		5,14	3,60	2,25	

$$\text{Valor crítico} = \underline{\underline{3,84}}$$

Como o valor calculado (**3:6**) **3,60** é menor que o valor crítico **3,84** → **H₀** não pode ser rejeitada. O que significa que o Framework i* não é equivalente ao anteriormente assimilado para essa competência. Podemos, sem calcular concluir que as competências com vantagem superior também não podem rejeitar a hipótese nula, porque todas certamente têm o valor do qui-quadrado calculados menores ainda que o valor crítico. São elas: **(1:8)** e **(0:9)**.

Para este segundo estudo estamos considerando o Método ERi*c como ideal e a comparação se faz com o real que no caso são as competências antes do treinamento do Método ERi*c.

Quadro 7. Resultado das Estatísticas da Experimentação 2

	Competências												
	A	B	C	D	E	F	G	H	I	J	K	L	M
Presença	1	0	0	1	1	1	0	1	0	0	1	0	0
Utilidade	1	1	1	1	1	1	1	1	0	1	1	1	1
Adequação	1	1	0	1	1	1	0	0	0	1	1	0	0

As competências “C”, “G”, “H”, “I”, “L” e “M” tiveram, na época do experimento, retorno que elas precisavam um refinamento maior no detalhamento. A competência “I” teve a utilidade não reconhecida e as competências “C”, “G”, “L” e “M” receberam retorno que são úteis e com detalhamento inadequado apesar de não estarem completamente presentes no Método ERi*c.

III.2.3.3 Análise quantitativa

Numa comparação superficial entre o FRAMEWORK i* e O MÉTODO ERi*c são analisadas as frequências das competências quanto a presença, utilidade e adequação do detalhamento.

⇒ **Métrica:** Lista das competências fornecidas pelo Método ERi*c que não fazem parte do Framework i*.

$$\text{PUA}_{1p} \{1, X, X\} = \{A, E, F\} \text{ e } \text{PUA}_{2p} \{1, X, X\} = \{A, D, E, F, H, K\}$$

⇒ **Métrica:** Lista das competências fornecidas pelo Método ERI*c e que são consideradas inúteis pelos engenheiros de requisitos.

$$PUA_{1p} \{1, 0, X\} = \emptyset \text{ e } PUA_{2p} \{1, 0, X\} = \{I\}$$

⇒ **Métrica:** Lista das competências fornecidas pelo Método ERI*c que fazem parte do Framework i* e são consideradas úteis pelos engenheiros de requisitos, cujo detalhamento deve ser modificado.

$$PUA_{1p} \{1, 1, X\} \cap PUA_{2p} \{1, 1, X\} = \{A, E, F\}$$

⇒ **Métrica:** Lista das competências fornecidas pelo Método ERI*c que não fazem parte do Framework i*, mas que são consideradas úteis pelos engenheiros de requisitos.

$$UA_{2p} \{1, X, X\} - PUA_{1p} \{1, X, X\} = \{D, H, K\}$$

III.4 Análise das Hipóteses

III.4.1 Análise das Hipóteses - Experimentação 1

$H_0 \rightarrow$ O resultado do questionário mostrou que 3 das 13 competências do Framework i* foram oferecidas e também foram reconhecidas como úteis. Podemos concluir que existe diferença entre a lista de competências oferecidas pelo Framework i* e as competências anteriormente assimiladas. (Hipótese alternativa H_1)

Dentre as competências oferecidas pelo Framework i* e consideradas úteis duas delas precisa de maior detalhamento. (Hipótese alternativa H_2)

III.4.2 Análise das Hipóteses - Experimentação 2

$H_0 \rightarrow$ O resultado do questionário mostrou que 6 das 13 competências do Método ERI*c foram oferecidas e também foram reconhecidas como úteis. Podemos concluir que existe diferença entre a lista de competências oferecidas pelo Método ERI*c e as competências do Framework i* anteriormente assimiladas. (Hipótese alternativa H_1)

Dentre as competências oferecidas pelo Método ERI*c e consideradas úteis 2 delas precisavam de maior detalhamento. (Hipótese alternativa H_2)

III.5 Operacionalização e análise da experimentação 3

Faz o Estudo das Competências do “i* Diagnoses”. De modo a avaliar o desempenho dos alunos na aplicação da estratégia de Diagnósticos i*, executamos um experimento com 14 participantes e fizemos a medição da eficácia do resultado.

III.5.1 Questionários das Competências e do Perfil do Participante

O questionário completo da experimentação do i* Diagnoses encontra-se em [Oliveira 08a] para a “Avaliação da Aplicação do i* Diagnoses - Caso Serviço de Recados na Secretaria Eletrônica”.

Descrição do experimento:

O experimento foi desenvolvido em duas seções. Na primeira seção foi pedido que os alunos fizessem em uma aula um Modelo SD e na aula seguinte um Modelo SR, para o caso SERVIÇO DE RECADOS NA SECRETARIA ELETRÔNICA.

Quadro 8– Quadro resumo das competencias avaliadas na Experimentação 3

	COMPETÊNCIA	DESCRIÇÃO
A	Identificação de problemas com atores	A utilização da técnica de diagnósticos é capaz de identificar problemas com os atores .
B	Identificação de problemas com dependências estratégicas	A utilização da técnica de diagnósticos é capaz de identificar os problemas com as dependências estratégicas entre atores de uma organização.
C	Identificação de oportunidades	A utilização da técnica de diagnósticos é capaz de identificar as oportunidades dos atores.
D	Verificação de metas concretas	A utilização da técnica de diagnósticos é capaz de verificar as metas concretas dos atores.
E	Verificação de metas flexíveis	A utilização da técnica de diagnósticos é capaz de verificar as metas flexíveis dos atores.
F	Verificação de funcionalidades	A utilização da técnica de diagnósticos é capaz de verificar as funcionalidades representadas.
G	Verificação de entidades tipo recurso	A utilização da técnica de diagnósticos é capaz de verificar os recursos representados.

Na segunda seção, a cada pessoa foi entregue dois diagramas i*, preparados pelo avaliador, que correspondiam ao trabalho da primeira seção, porém com as correções pertinentes, e mais o framework de perguntas para diagnóstico de SDsituations e de SRconstructs. Pedimos que os participantes aplicassem as perguntas formuladas sobre os modelos SD e SR. Eles tiveram 2 horas para fazer o trabalho de diagnóstico. Na segunda seção a avaliação cobriu as 7 competências destacadas no formulário do Quadro 8 e, na avaliação dos trabalhos, 5 tópicos foram considerados para a avaliação da eficácia:

- a) se os construtores i* foram corretamente identificados;
- b) se os participantes entenderam as perguntas;
- c) se os participantes foram capazes de identificar as omissões;
- d) se os participantes identificaram os erros em componentes e
- e) se as omissões de metas flexíveis foram identificadas.

De acordo com esses tópicos o avaliador classificou os resultados como:

A – quase 100% correto
C – quase 50% correto e

B – quase 70% correto
D – menos que 40% correto

O Quadro 9 e a Figura 1 apresentam a tabulação das notas dos trabalhos. Na segunda seção o trabalho de cada aluno foi corrigido por outro aluno que também participou do experimento usando o gabarito distribuído pelo avaliador.

Quadro 9. Mapeamento da correção do exercício SEM diagnóstico

	Participantes													
Competências	1	2	3	4	5	6	7	8	9	10	11	12	13	14
Identificação de alternativas	B	A	D	D	D	B	D	C	D	C	D	D	D	D
Identificação de problemas	B	D	B	C	D	D	B	B	D	C	D	D	C	D
Verificação de metas flexíveis	A	C	D	D	C	C	C	D	D	B	C	C	D	D

A Figura 1 mostra que a estratégia teve resultado positivo em relação à eficácia. O resultado apontou vantagem significativa para as frequências “A – quase 100% correto” e “B – quase 70% correto”.

Os resultados da terceira experimentação, Diagnósticos i*, foram positivos:

- a identificação dos construtores (SDsituations e SRconstructs) foi 100% alcançada por 9 dos 14 participantes;
- o entendimento das questões foi alcançada por 8 dos 14 participantes;
- na identificação das alternativas apenas dois participantes conseguiram 100% e 11 do total de 14 participantes conseguiram resultados que fazem o modelo diagnosticado ser aprimorado. Apenas 3 participantes não chegaram a nenhum resultado proveitoso;
- a identificação de problemas foi o item de pior desempenho: 1 aluno fez 100%, 6 participantes quase alcançaram os 100% e 7 participantes identificaram aproximadamente metade dos problemas;
- na identificação de metas flexíveis 7 de 14 participantes tiveram ótimo resultado e apenas 1 aluno não conseguiu fazer nada da elicitación de metas flexíveis.

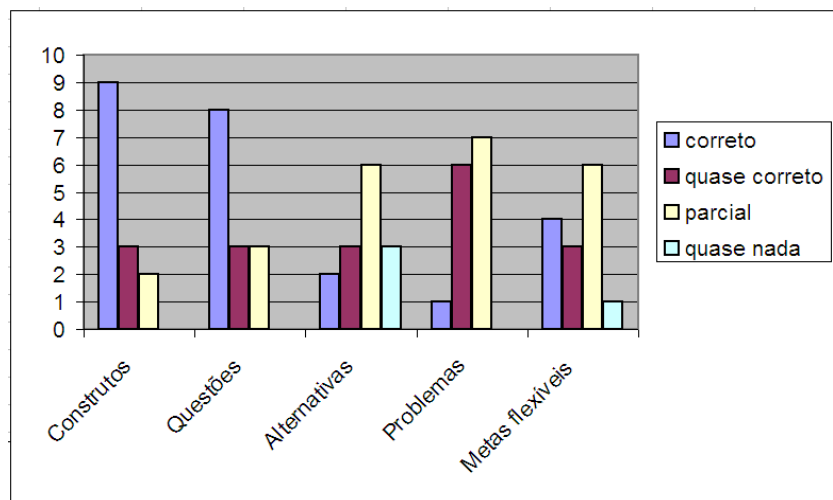


Figura 1. Gráfico de barras da correção do exercício COM diagnóstico

5. Conclusão

Este artigo apresentou os fundamentos da experimentação em engenharia de software e também apresentou com detalhes três experimentos para a avaliação das competências fornecidas pelo Método ERi*c seguindo os passos do trabalho “Introdução a Engenharia Experimental” [Travassos 02].

A experimentação concluiu que o Método ERi*c contribui com a presença de 3 competências úteis {D, H e K}, além das 3 {A, E e F} que o Framework i* já fornece para a engenharia de Requisitos de Sistemas Multi-Agentes. Resultado: contribuição adicional de 23%.

Como resultado as experimentações indicaram também que o Método ERi*c possuía problemas de clareza e de detalhamento nas heurísticas da utilização do método mas, mesmo assim, o Método ERi*c agregou com detalhamento adequado em 5 competências {B, D, E, J e K} além das 2 {A e F} que o Framework i* já fornece. Resultado: contribuição adicional de 38%.

A terceira experimentação avaliou a técnica de Diagnósticos i* e inferiu que com o apoio das perguntas os participantes:

- na identificação das alternativas a melhora foi de 21% (3 OK contra 11) para 79% (11 do total de 14 participantes conseguiram resultados);
- na competência identificação de problemas a melhora dos participantes foi de 8% para 100% (todos os alunos encontraram pelo menos metade dos problemas);
- na identificação de metas flexíveis a melhora no desempenho dos participantes foi de 16% (2 OK contra 12) para 92% (13 em 1).

6. Referências

- Basili, Victor R.; Caldiera, Gianluigi; Rombach, H. Dieter, “The Goal Question Metric Paradigm”, Encyclopedia of Software Engineering (Marciniak, J.J., editor), Volume 1, John Wiley & Sons, 1994, pp. 578-583
- Basili, V. “The Role of Experimentation in Software Engineering: Past, Current, and Future,” Proceedings of the Eighteenth International Conference on Software Engineering (ICSE), Berlin, Germany, March 1996.
- Box, G. E. P.; Hunter, W. G., and Hunter, J. S., Statistics for experimenters. An introduction to design, data analysis and model building. New York, Wiley, 1978.
- Estrada, H; Martínez, A; Pastor, O; Mylopoulos, J.; An Experimental Evaluation of the i* Framework in a Model-based Software Generation Environment; E. Dubois, K. Pohl (Eds.); CAISE 06, LNCS vol. 4001, pp.513-527, Springer-Verlag, Berlin Heidelberg, ISBN: 3-540-34652-X, 978-3-540-34652-4 - 2006.
- Mafra, Sômolo; Travassos, Guilherme; “Estudos Primários e Secundários apoiando a busca por Evidência em Engenharia de Software” - Relatório Técnico: RT-ES-687/06 – Programa de Engenharia de Sistemas e Computação - COPPE/UFRJ – Rio de Janeiro, 2006.

- Oliveira, A. Padua A.; Engenharia de Requisitos Intencional: Um Método de Elicitação, Modelagem e Análise de Requisitos. PhD Tese, PUC-Rio, Março/2008.
- Oliveira, A. Padua A.; Leite, J. C. S. P.; Cysneiros, L. M.; Lucena, C. J.; i* Diagnoses: A Quality Process for Building i* Models - Proceedings of the Forum at the CAiSE'08, Montpellier, France, June 18-20, 2008. CEUR Workshop Proceedings 344 CEUR-WS.org 2008 pp. 9-12.
- Pastor, Oscar; Estrada, Hugo; Martínez, Alicia; The Strengths and Weaknesses of the i* Framework: an experimental evaluation i*, its Applications, Variations and Extensions. Eric Yu et als. (eds.) 2006 - MIT Press.
- Travassos, Guilherme; Gurov, Dmytro; Amaral, Edgard; “Introdução a Engenharia de Software Experimental” - Relatório Técnico: RT-ES-590/02 – Programa de Engenharia de Sistemas e Computação - COPPE/UFRJ – Rio de Janeiro, 2002.
- Yu, E. Modelling Strategic Relationships for Process Reengineering. PhD Thesis, Graduate Department of Computer Science, University of Toronto, Canada, 1995.

CADERNOS DO IME

Série Informática

Normas para publicação de trabalhos

A revista Cadernos do IME é o veículo de divulgação dos trabalhos acadêmicos do corpo docente do Instituto de Matemática e Estatística da Universidade do Estado do Rio de Janeiro.

A revista é semestral, sendo publicada em junho e dezembro. São aceitos trabalhos inéditos e que apresentem elementos de originalidade, em português, inglês ou espanhol.

Os Cadernos do IME aceitam os seguintes tipos de contribuições:

- Artigo técnico, trabalho de pesquisa com elementos de originalidade;
- Tutorial, texto introdutório a áreas específicas da ciência da computação, com revisão de literatura e organização dos conceitos relacionados;
- Resenha / Revisão, revisão de literatura e análise crítica de produtos da área de ciência da computação e informática;
- Resumo, texto destinado à informação sobre trabalhos de graduação e de pós-graduação;
- Comunicação, texto destinado à divulgação de opiniões, pesquisas em andamento, lançamentos e divulgação de eventos.

Os artigos e tutoriais não devem ultrapassar 20 páginas e as demais formas de publicação devem se restringir a no máximo 10 páginas.

Todos os trabalhos enviados para publicação nos Cadernos do IME – Série Informática, independente de tipo de contribuição, devem seguir o modelo para publicação de artigos da SBC – Sociedade Brasileira de Computação.

Os arquivos digitais no formato PDF deverão ser encaminhados aos editores da série, juntamente com uma cópia impressa de boa qualidade.

A submissão é eletrônica e em fluxo contínuo no endereço <http://www.e-publicacoes.uerj.br/index.php/cadinf>. As chamadas de trabalho para volumes especiais estão disponíveis <http://www.ime.uerj.br/cadernos/cadinf/>.

Os conteúdos e pontos de vista expressos nos trabalhos publicados são de inteira responsabilidade dos autores.



Ricardo Vieirals de Castro
Reitor

Paulo Roberto Volpato Dias
Vice-Reitor

Maria Georgina Muniz Washington
Diretora do Centro de Tecnologia e Ciência

Geraldo Magela da Silva
Diretor do Instituto de Matemática e Estatística

Pedro Antonio Sarubbi
Vice-Diretor do Instituto de Matemática e Estatística

Rubens André Sucupira
Departamento de Análise Matemática

Jéssica Ruth Gavia Zurita
Departamento de Estruturas Matemáticas

Mara de Carvalho de Sousa
Departamento de Geometria e Representação Gráfica

Rosa Maria Esteves Moreira da Costa
Departamento de Informática e Ciência da Computação

Guilherme Caldas de Castro
Departamento de Estatística

Augusto Cesar de Castro Barbosa
Departamento de Matemática Aplicada