

DESAFIOS GERENCIAIS E TÉCNICOS NA MUDANÇA DE TECNOLOGIA TECNOLOGIA E A REENGENHARIA NO PROCESSO DE DESENVOLVIMENTO DE SOFTWARE¹

Cleber Nardelli², Fábio Alexandrini³, José Ernesto de Fáveri⁴

RESUMO

O artigo aborda o desafio que as empresas produtoras de software enfrentam a partir deste novo paradigma, que é a migração de suas aplicações legadas da plataforma desktop para web, utilizando-se para isto de boas práticas e técnicas já conhecidas e consagradas de engenharia de software. Para abordar o problema estudado que envolve as principais dificuldades e as necessárias mudanças para alteração da tecnologia de desenvolvimento de software, este artigo fora baseado no estudo de caso em uma empresa de desenvolvimento de software da região do Alto Vale do Itajaí, seus produtos desenvolvidos são voltados para a área pública municipal. O principal objetivo do estudo deste caso foi analisar a situação dos softwares já desenvolvidos, como artefatos de documentação produzidos durante a sua concepção, verificando como o conteúdo existente poderá nortear a migração desses softwares legados para a tecnologia de desenvolvimento web, contribuindo desta forma para que novos projetos de migração deste porte façam uso de algumas das técnicas abordadas neste artigo e em consequência no melhor aproveitamento do tempo para o desenvolvimento desses projetos bem como a garantia da obtenção de um grau de qualidade maior.

Palavras-chave: mudança de tecnologia no desenvolvimento de software, engenharia de software, software legado, software web.

¹ Artigo Científico apresentado ao curso de Pós-Graduação em Desenvolvimento Web.

² Bacharel em Sistemas de Informação e Pós-Graduado em Gestão da Tecnologia da Informação pela UNIDAVI, Professor do Curso de Sistemas de Informação da UNIDAVI e Professor do Curso de Sistemas de Informação da FAMESUL, e-mail: clebernardelli@gmail.com.

³ Doutor e Mestre em Engenharia de Produção e Sistemas pela UFSC, Professor dos Cursos de Administração, Automação Industrial, Engenharia de Produção e Sistemas de Informação da UNIDAVI, e-mail: fabalex@unidavi.com.br.

⁴ Doutor em Educação, Professor titular do curso de Administração da UNIDAVI.

INTRODUÇÃO

O desenvolvimento de aplicações para a plataforma web⁵ é uma realidade que chama a atenção, não só pelas características de ambiente distribuído, multiplataforma, etc., mas também pela recente onda que toma conta da grande maioria das empresas produtoras de software “a migração de seus produtos para web”.

Contudo, temos hoje um cenário onde as aplicações existentes estão mal documentadas, ou ainda não estão documentadas. Quando a maioria dessas aplicações foi desenvolvida, não havia grandes preocupações com métodos eficientes de desenvolvimento, cada empresa desenvolvia software da maneira como lhe era possível ou cômodo, os poucos padrões existentes para documentação, especificação e construção de software não eram seguidos ou pouco seguidos.

Segundo Pressman (2006), na década de 1990 o “boom” da web veio com o surgimento de diversas empresas “pontocom”. Neste mesmo período surgiu um novo paradigma no desenvolvimento de aplicações, chamado de – desenvolvimento web (ou *webapp*) – que se preocupava mais com alcançar prazos estipulados do que com técnicas de desenvolvimento de software já existentes e de uso comprovado pela engenharia de software, ele também comenta:

“Nos primeiros dias da World Wide Web (entre 1990 e 1995), os “sites web” eram formados de pouco mais do que um conjunto de arquivos de hipertexto ligados que apresentavam informação usando texto e um pouco de gráficos. Com o passar do tempo, a HTML foi crescendo com ferramentas de desenvolvimento (por exemplo XML, Java) que habilitaram os engenheiros web a fornecer capacidade computacional junto com a informação.” (PRESSMANN, 2006, p. 379).

As tecnologias de desenvolvimento web evoluem constantemente. Por volta do ano de 2003, surgiu um novo conceito batizado de *web 2.0*, havia uma mudança visível no modo como as pessoas e os negócios estavam usando a web, nesta mesma onda o desenvolvimento das aplicações para esta nova web também teria que ser revisto.

Segundo DEITEL (2008), o crescimento da web 2.0 está vinculado principalmente à:

“[...] Primeiro, o hardware estava se tornando cada vez mais barato e rápido, com capacidade de memória e velocidade aumentando a altas taxas. A lei de Moore afirma que o poder do hardware dobra a cada dois anos, enquanto o preço permanece essencialmente o mesmo, o que permite o desenvolvimento de aplicações com alta demanda que antes teria sido impensável. Segundo, o uso da internet de banda larga explodiu – um estudo da Pew Internet em março de 2006, revelou que 42 por cento dos adultos norte-americanos tinham internet de alta velocidade. Terceiro, a farta disponibilidade de software de código aberto resultou em opções de software personalizáveis de custo mais baixo e até mesmo gratuitas, o que facilitou a criação de novas empresas de Web 2.0 e diminuiu grandemente o custo do fracasso. Quarto, diferentemente da Web 1.0 (o estado da web durante a década de 1990 e início da década de 2000), há vários modelos tipo “faça-fácil” disponíveis para monetizar os códigos na web 2.0 – a geração imediata de lucro, ainda que modesto, possibilita um crescimento mais estável das novas empresas.” (DEITEL, 2008, p. 3).

Como principal ponto da “explosão” da web 2.0, dos citados por DEITEL, pode ser enfatizado o uso de softwares de código aberto, disponível para que qualquer um

⁵ WWW ou World Wide WEB, em português “Rede de Alcance Mundial”.

possa usar e (quando possível) modificar com pouca ou nenhuma restrição. Criou-se um novo universo onde as comunidades virtuais colaboram para desenvolver softwares por vezes até melhores do que softwares proprietários, como é o caso da linguagem de desenvolvimento PHP⁶.

A web 2.0 trouxe consigo tecnologias inovadoras que hoje estão em nosso cotidiano, como as redes sociais. Essas redes mudaram consideravelmente a maneira como interagimos e nos comunicamos.

Como inovação recente podemos destacar o surgimento da versão 5 do HTML⁷. Trata-se da nova versão do padrão mundialmente conhecido como HTML. Esta versão vem de encontro as necessidades atuais da web, dispensando em muitos casos o uso de bibliotecas e pacotes auxiliares para tarefas como animação.

REENGENHARIA DE SOFTWARE E O PROCESSO

Como cenário principal podemos considerar um software que por muito tempo (de 10 a 15 anos) tem servido aos princípios norteadores da organização. Ele é utilizado regularmente como forma de operacionalização dos negócios, além de gerar algumas informações gerenciais como forma de apoio a tomada de decisão, mas ele está ficando velho, quebra com frequência, as reparações estão levando mais tempo do que se prevê e por fim deixou de representar a tecnologia mais evoluída em termos de desenvolvimento de sistemas.

Segundo Furlan (1994, p. 40) a reengenharia é um “Conjunto de técnicas e ferramentas orientadas à avaliação, reposicionamento e transformação de sistemas de informação existentes, com o objetivo de estender-lhes a vida útil e ao mesmo tempo, proporcionar-lhes uma maior qualidade técnica e funcional”.

Este é o ponto onde a maioria das empresas produtoras de software se encontra. Este é o momento de mudança. Esta mudança não significa apenas selecionar a tecnologia mais recente em termos de linguagem de desenvolvimento, e reescrever todas as telas utilizando-a. Neste contexto outras atividades são necessárias, como avaliar o que deverá ser migrado, como isto será feito, quais as regras de negócio que definirão as funcionalidades que o software deverá prover aos usuários.

Sobre este assunto, Pressmann (2006) comenta que:

“O software é frequentemente a realização de diversas regras de negócio. À medida que essas regras se modificam, o software também deve ser modificado. Hoje em dia, importantes empresas tem dezenas de milhares de programas de computador que apoiam as regras de negócio antigas. À medida que os gerentes trabalham para modificar as regras, a fim de conseguir maior eficiência e competitividade, o software deve acompanhar o ritmo. Em alguns casos isso significa a criação de novos sistemas[...]” (PRESSMANN, 2006, p. 682).

O processo de migração de todo este artefato produzido para uma nova plataforma de desenvolvimento, dependendo do tamanho do software é bastante

⁶ Acrônimo de *PHP: Hypertext Preprocessor*, linguagem de programação de computadores interpretada, livre e utilizada para web. É executada no servidor de aplicações.

⁷ Acrônimo em inglês de *HyperText Markup Language*, que significa *Linguagem de Marcação de Hipertexto*. Linguagem simplificada de marcação utilizada para produção de páginas web. É executada na estação de trabalho (lado cliente).

dispendioso e pode selar o sucesso ou o fracasso de um projeto. Como será feito para que nada seja esquecido durante este processo, causando grandes prejuízos na entrega do produto final?

Invariavelmente a resposta para esta pergunta passa não só pela elaboração de um processo definido, mas também pela mudança de alguns papéis dentro da organização. Para que um bom processo possa ser executado são necessárias outras variáveis como Pessoas e Tecnologia.

Segundo Pressmann(2006), “[...] um processo de software define a abordagem que é adotada quando o software é elaborado. Mas a engenharia de software também inclui tecnologias que constituem um processo[...]”.

ADMINISTRAÇÃO PÚBLICA E E-GOV

A ideia de colaboração existente por trás da web 2.0, abriu novos modelos de negócio. Alguns desses negócios seriam praticamente inviáveis dez anos atrás. O serviço de pregão eletrônico on-line, onde uma empresa pode ofertar seus produtos para diversas outras instituições é um exemplo.

Observando este cenário a administração pública juntamente com iniciativa privada, formaram o conceito de e-gov.

No artigo “Prefeitura Virtual: a Internet a serviço da comunidade”, os autores se referem a este quadro da seguinte forma:

“As novas tecnologias de informação e comunicação têm o potencial para melhorar a eficiência na prestação de serviços públicos e privados, melhorar a inserção das empresas locais numa economia cada vez mais globalizada e competitiva através de um relacionamento mais ágil com o governo [...]”. (ALEXANDRINI, Fábio; PISKE, Ingobert; PISKE, Ricardo, 2005, p 2).

Neste aspecto a administração pública municipal, como um todo tem procurado diversificar a oferta de serviços disponíveis não só para empresas, mas também para o cidadão.

Além disso, atos legais também incentivam a prestação de serviços à comunidade. Em maio deste ano encerrou o prazo para que prefeituras com mais de 100.000 habitantes se adequassem à lei complementar n. 131, de 2009, também conhecida como Lei da Transparência. Através deste serviço prestado em tempo real o cidadão poderá acompanhar os gastos realizados pela administração pública. É dever das prefeituras divulgar o lançamento e o recebimento de toda a receita das unidades gestoras.

ABORDAGEM DO CASO DE ESTUDO

No caso em estudo existe um cenário semelhante ao mencionado anteriormente. O software que deverá passar por um processo completo de reengenharia é um ERP para órgãos públicos municipais, que conta com 27 módulos, conforme apresentado no quadro 1.

MÓDULOS DO SISTEMA ERP DE GESTÃO PÚBLICA		
Gestão Orçamentária/Financeira e Contábil	Escrita Fiscal	Fiscalização Fazendária
Gestão de Arrecadação	Gestão de Dívida Ativa	IPTU/Taxas

ISSQN/Taxas	Nota Fiscal Eletrônica	Tarifa de Água
Receitas Diversas	Contribuição de Melhoria	Ouvidoria
Protocolo	Processo Digital	Legislação
Praça de Atendimento	Portal do Cidadão	Procuradoria
Patrimônio	Compras e Licitações	Almoxarifado
Frotas	Planejamento	Business Intelligence
Gestão de Pessoal (RH)	Controle Interno	Fiscalização de Obras e Posturas

Quadro 1 - Módulos do Sistema ERP da Gestão Pública Municipal

Fonte: Acervo do Autor

Na tabela 1, é exibida também uma coluna identificando a quantidade de linhas de código por módulo, embora não seja determinante, isto ajuda a identificar fatores como tamanho e complexidade, necessários para que seja estabelecido um planejamento para o desenvolvimento.

Vários módulos estão desenvolvidos sob a plataforma desktop tradicional, outra parte dos módulos já se encontra desenvolvido para plataforma web, porém nesses casos ou tratam-se de módulos novos ou módulos pequenos que foram refeitos quase que em sua totalidade.

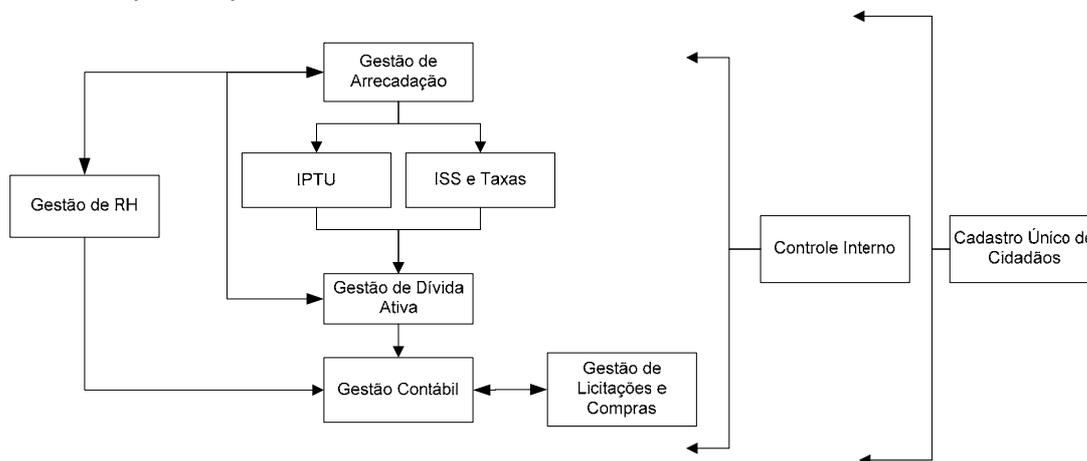


Figura 1 - Integração entre os Principais Módulos

Fonte: Acervo do Autor

Trataremos aqui especificamente da migração dos módulos desktops para a plataforma web. A empresa tem o perfil onde a mesma desenvolve a maior parte das tecnologias utilizadas para o planejamento, criação e implantação dos produtos desenvolvidos. A empresa conta hoje com clientes distribuídos nos estados do Rio Grande do Sul, Santa Catarina, Paraná, São Paulo, Minas Gerais, Espírito Santo e Goiás, porém, como mostra o gráfico 1, na região sul é onde se concentra o maior número de clientes.



Gráfico 1 - Clientes por Região

Fonte: Acervo do Autor

Os clientes são basicamente órgãos Públicos municipais, Prefeituras, Câmaras de Vereadores, Samaes, Fundações, Autarquias, Fundos e também postos de saúde, através do uso do software de saúde.

A empresa conta hoje com clientes nos estados de Santa Catarina, Rio Grande do Sul, Paraná, São Paulo, Minas Gerais, Espírito Santo e Tocantins, porém a maior força de clientes, cerca de 50%, está distribuída na região sul do país. É importante salientar que a entrada de software ERP público em especial para órgãos municipais, em municípios de qualquer estado novo poderá significar alterações em vários requisitos, culminando muitas vezes em necessária reestruturação de estruturas de dados bem como de programas, haja visto que as leis regulamentadoras para prestação de contas em nível estadual são específicas para cada estado.

A empresa também conta com um contingente de pessoal especificamente na área de desenvolvimento suficiente para a demanda atual, porém existe uma pretensão de aumentar este contingente de 10% a 15% ainda no ano de 2010. Esta equipe está bem distribuída se levarmos em conta a proporcionalidade por função, conforme apresentado no gráfico 2.

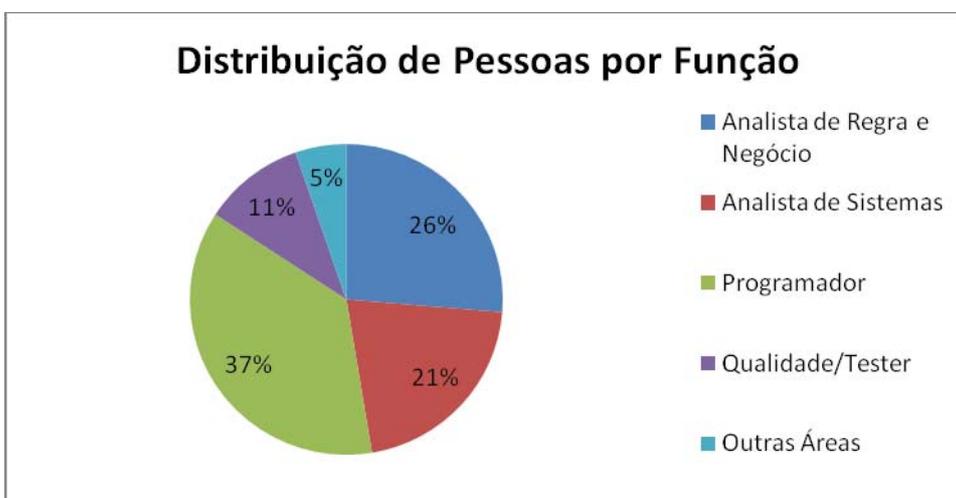


Gráfico 2 - Distribuição de Pessoas por Função

Fonte: Acervo do Autor

De acordo com o gráfico 3, apenas 39% do contingente de pessoal possui mais de 3 anos de experiência com desenvolvimento na plataforma web. Através de uma mescla de profissionais pôde-se organizar a equipe de forma que a entrada de novos colaboradores seja possível, isto fica evidente quando se verifica que os outros 61% possuem menos de 2 anos de experiência.

Porém faz-se necessário que os papéis sejam distribuídos entre as pessoas levando-se em conta não somente o tempo de experiência em desenvolvimento web, mas também experiências passadas em desenvolvimento de software para outras plataformas, principalmente para papéis onde a tecnologia não é fator determinante, como a especificação de requisitos funcionais.



Gráfico 3 - Tempo de Experiência Profissional com Desenvolvimento Web

Fonte: Acervo do Autor

Contudo, a empresa conta com um plano de capacitação contínuo, onde são realizados treinamentos internos em diversas áreas e ferramentas de desenvolvimento. Além disso, a empresa realiza constantemente um projeto destinado a captação de pessoal em parceria com universidades da região, onde são disponibilizados treinamentos gratuitos para os alunos dos cursos relacionados a área de informática dessas universidades.

Como característica marcante do segmento de desenvolvimento de software, no qual a empresa está inserida, temos as mudanças significativas em requisitos.

Um requisito de software é basicamente uma expectativa do cliente em relação ao software. Ele pode ser explícito, sendo coletado durante a fase de análise na atividade de levantamento de requisitos ou implícito, neste caso o cliente entende que o analista irá considerar determinada função ou regra de função que considera notória na sua concepção.

“Compreender a natureza dos problemas pode ser muito difícil, especialmente se o sistema for novo. Consequentemente, é difícil estabelecer com exatidão o que o sistema deve fazer. As descrições das funções e das restrições são os requisitos para o sistema, e o processo de descobrir, analisar, documentar e verificar essas funções e restrições é chamado de engenharia de requisitos.” (SOMMERVILLE, 2004, p. 82).

Sommerville classifica ainda os requisitos em funcionais (que são as declarações de funções que o sistema deve fornecer), não funcionais (são restrições sobre serviços ou as funções oferecidas pelo sistema) e de domínio (se originam no domínio de aplicação do sistema, como as leis regulamentadoras).

Este último principalmente por que a administração pública está em constante evolução e as regras de prestação de contas dos governos são alteradas para que adequações sejam feitas para a garantia da transparência nas administrações. Essas regras refletem, por vezes, em mudanças substanciais nos softwares desenvolvidos, sendo necessárias várias horas de análise junto a uma base sólida de conhecimentos, que em sua maioria são pessoas que trabalham exaustivamente na tentativa de acompanhar estas mudanças.

As prestações de contas são periódicas e muitas vezes o intervalo entre uma prestação e outra é definido pelo número total de habitantes do município. O gráfico a seguir demonstra a distribuição proporcional de clientes pelo número de habitantes.

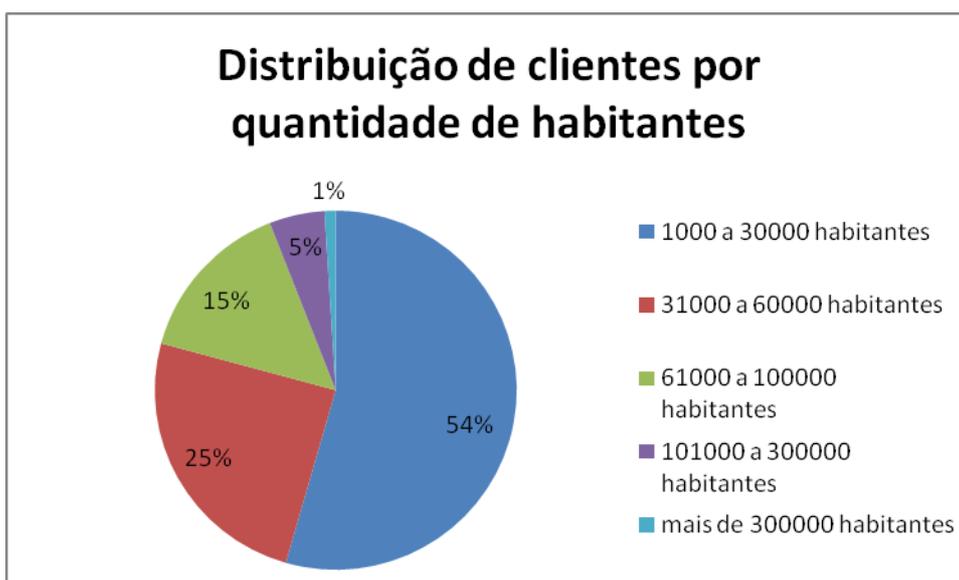


Gráfico 4 - Distribuição de Clientes por faixa de habitantes

Fonte: Acervo do Autor

Atualmente a empresa desenvolve a maior parte de suas aplicações para a plataforma desktop, utilizando-se de diversas linguagens de desenvolvimento dependentes da plataforma de sistema operacional. Isto tem como principal impacto negativo a característica de o software não ser portátil para outras plataformas de sistema operacional, o que muitas vezes encarece o preço da infraestrutura necessária, incluindo as licenças de concessão de uso de software básico.

Esta tecnologia começou a ser utilizada na empresa há cerca de 15 anos, ainda quando existiam softwares baseados no sistema operacional MS-DOS.

APRESENTAÇÃO E DISCUSSÃO DOS RESULTADOS

Fora realizada uma pesquisa quantitativa no mês de julho de 2010 através da aplicação de um questionário, contendo 14 questões de múltipla escolha. As questões foram elaboradas com o objetivo de demonstrar o quadro atual da empresa através da medição de algumas variáveis. Responderam ao todo 38 pessoas da área de desenvolvimento de software.

Analisando os resultados obtidos na questão que perguntava: “O Seu grau de dificuldade no aprendizado da nova tecnologia (web) é (ou foi)?”, observamos que apenas 7% das respostas foram grande ou muito grande, conforme mostrado no gráfico 5. Isto indica que o aprendizado da tecnologia de desenvolvimento web é simples e sendo assim o tempo de formação de pessoal poderá ser pequeno.

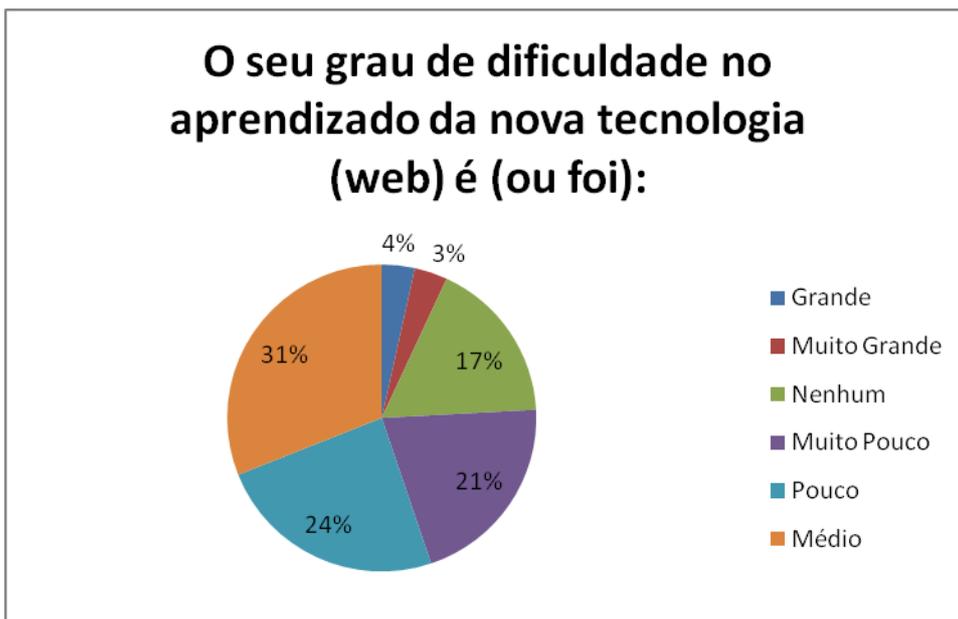


Gráfico 5 - Grau de dificuldade no aprendizado da nova tecnologia web.
Fonte: Acervo do Autor

Ao observarmos ainda alguns resultados inerentes da área de desenvolvimento percebemos, por exemplo, que quando analisamos que aspectos poderiam ser melhorados na atual gestão de requisitos, 70% dos entrevistados responderam que é na descrição da regra de negócio que está o principal ponto de melhora, conforme mostra o gráfico 6.

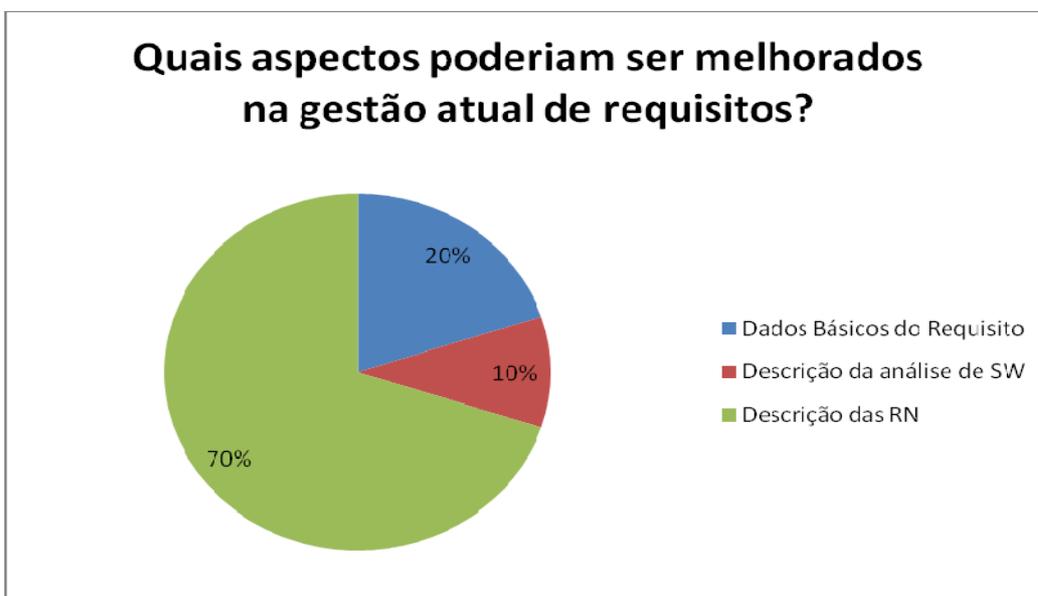


Gráfico 6 - Aspectos que poderiam ser melhorados na gestão de requisitos.
Fonte: Acervo do Autor

Na construção de um novo software, muitos são os fatores a serem considerados, destacando-se aqui uma forte documentação para que a manutenibilidade⁸ do produto final seja suficientemente forte a ponto de suportar por

⁸ Manutenibilidade, na Engenharia de Software é um aspecto de qualidade de software que refere-se à facilidade de manutenção de um software, a fim de corrigir defeitos, adaptar-se a novos requisitos ou de adequar-se à um novo ambiente de trabalho.

algum tempo ajustes que se fizerem necessários, destaca-se aqui a preocupação da maioria dos entrevistados na pesquisa que, conforme o gráfico 6 aponta, a descrição da regra de negócio poderia ser melhorada.

Quando um software deve ser redesenhado ou readequado, diz-se que este deve passar por um processo de reengenharia. Muitas vezes não bastam apenas ajustes ou melhorias no software atual, mas sim seu completo redesenvolvimento. Quando isto ocorrer e, em especial ocasião, houver uma mudança de plataforma tecnológica, devemos ter em mente que o software atual deverá possuir uma documentação eficiente o bastante para que a execução deste tipo de projeto seja possível. Quando esta documentação não existe, devemos criá-la. Este processo chamado de engenharia reversa, concentra-se em buscar das mais variadas fontes o conhecimento necessário sobre o produto atual.

Como resultado deste processo, teremos uma série de requisitos de software, construídos a partir do seu executável (das interfaces disponíveis ao usuário, de consultas, de relatórios, etc), dos *helps* existentes, das regras de negócio legais (leis que devem ser seguidas), do conhecimento técnico de consultores especializados, dentre outros. Após esta etapa concluída, deverá ser realizada uma reunião técnica formal onde os requisitos serão cuidadosamente analisados um a um, à fim de que inconsistências sejam encontradas e solucionadas.

As reuniões técnicas formais, neste ponto, servem também para definir quais os requisitos que serão mantidos inalterados, que serão alterados em sua essência (melhorias incrementais ou legais a fim de corrigir anomalias do software anterior), que serão descontinuados (requisitos que não serão mais necessários ou que a nova versão não irá suportar) e possíveis novos requisitos (necessidades levantadas ao longo do tempo, porém não puderam ser desenvolvidas no software atual).

A reunião técnica formal deve ser conduzida por um especialista na área de negócio cujo software está sendo desenvolvido, ou então por um analista de sistemas também da área afim. Como resultado desta fase o analista de sistemas deverá analisar todos os requisitos aprovados, juntamente com a descrição que norteia sua regra de negócio final, a fim de determinar quais são as estruturas de dados necessárias, as interfaces necessárias, as saídas de dados necessárias, entre outras.

Após algumas análises realizadas junto aos colaboradores da empresa em pesquisa formal, cuja alguns dos resultados estão expressos neste artigo, concluí-se que:

1. Precisamos definir claramente cada requisito que deverá ser desenvolvido. Invariavelmente os softwares que serão migrados não possuem documentação suficientemente forte e eficaz para que um novo projeto possa ser iniciado e concluído com sucesso. Todas as funções e características do software atual devem estar muito bem definidas e documentadas para que questões do tipo “Iremos continuar a fornecer esta função no software?”, possam ser respondidas de forma precisa.

Neste aspecto recomenda-se que uma engenharia reversa seja realizada no software atual e que os requisitos que estão disponíveis no mesmo em forma de funções, regras e/ou características de uso, possam ser integralmente abordados durante a concepção do novo software.

Reuniões técnicas formais que visem a verificação e validação de requisitos são necessárias. Faz-se necessário que neste tipo de reunião sejam elencados os requisitos que serão descontinuados, alterados, inalterados ou a inclusão de novos requisitos;

2. As estruturas de dados que ai estão serão modificadas e sendo assim, existe a necessidade de que seja estabelecida uma forte documentação do tipo “de-para”, onde os analistas possam a cada estrutura alterada ou criada adicionar comentários que serão úteis nas fases seguintes do projeto;
3. É de suma importância que todos os envolvidos saibam exatamente as atividades a serem realizadas, em que fase do processo eles serão necessários e também quais resultados emitidos pelo trabalho deles serão necessários para a realização do trabalho de outros;
4. A realização *workshops* internos visando uma maior integração entre os envolvidos e principalmente a disseminação de conhecimento, bem como a apresentação de ferramentas, técnicas e padrões, é importante e devem ser realizados com frequência.

Recomenda-se como forma de estudo posterior, a realização de uma nova pesquisa interna para reavaliação de resultados, haja vista que em média 30% das pessoas que participaram da pesquisa, deixaram de responder ao menos uma questão, isto pode demonstrar desconhecimento de parte das atividades realizadas ou ainda a sua integração completa com a equipe de trabalho.

CONSIDERAÇÕES FINAIS

Definir uma padronização clara e concisa não basta, precisa-se integrar a equipe para que a troca de informações entre os membros seja realizada juntamente com um alinhamento estratégico dos envolvidos incluindo a direção da empresa, a fim de que os padrões estabelecidos sejam seguidos e que possíveis melhorias sejam implementadas a fim de garantir um amadurecimento contínuo do processo.

Através do estudo de caso realizado em uma empresa de desenvolvimento de software da região do Alto Vale do Itajaí, cuja seu produto principal é um GRP⁹ público municipal, que conta hoje com 27 módulos e clientes nas regiões centro-oeste, sudeste e sul do Brasil. O estudo analisou a situação dos softwares já desenvolvidos, como artefatos de documentação produzidos durante a sua concepção, verificando como o conteúdo existente poderá nortear a migração desses softwares legados para a tecnologia de desenvolvimento web, contribuindo desta forma para que novos projetos de migração deste porte façam uso de algumas das técnicas existentes.

Referente a pesquisa realizada, quando questionado sobre a atual gestão de requisitos, é importante destacar que 70% das respostas aponta para uma necessidade de melhoria na descrição da regra de negócio. O uso de um bom *check-list* para a elaboração desta documentação ajudará em uma melhor definição

⁹ GRP, acrônimo em inglês para *Government Resource Planning*, em português *Software Integrado de Gestão Pública*. É um tipo de software que integra todos os dados e processos de uma organização pública em um único sistema.

dos requisitos. Deve-se observar também o posicionamento físico das pessoas, isto pode influenciar em uma boa ou má descrição. Por exemplo, se um analista de sistemas está muito próximo de um programador, a conversação entre eles facilitará a realização das atividades, porém poderá tornar a documentação fraca e imprecisa, na medida em que não será atualizada com o passar do tempo.

Através de uma gestão eficiente de requisitos é possível determinar com uma confiabilidade maior o que deverá ser feito, além de utilizá-los para outras atividades subsequentes ao desenvolvimento, como a realização de testes e documentação.

Para a gestão de projeto, o controle sobre o desenvolvimento, como os prazos, pode ser realizado buscando-se informações em tempo real, através do status atual de cada requisito.

REFERÊNCIAS

SOMMERVILLE, I. **Engenharia de Software**. 6ª. Edição. São Paulo: Addison Wesley, 2003.

PRESSMAN, R. S. **Engenharia de Software**. 6ª. Edição. São Paulo: McGrawHill, 2006.

DEITEL, Paul J. **Ajax, Rich Internet Applications e Desenvolvimento Web para Programadores**. 1ª. Edição. São Paulo: Pearson Prentice Hall, 2008.

FURLAN, José Davi. **Reengenharia da Informação do mito a realidade**. 1ª. Edição. São Paulo: Makron Books, 1994.

ALEXANDRINI, F. ; PISKE, I. ; PISKE, R. . **Prefeitura Virtual: a Internet a Serviço da Comunidade**. In: III SEGeT Simpósio de Excelência em Gestão e Tecnologia, 2006, Resende RJ. III SEGeT Simpósio de Excelência em Gestão e Tecnologia. Resende RJ : Associação Educacional Dom Bosco (AEDB), 2006.