Leonardo Di Francescantonio¹
Fernando Luiz de Almeida²
Mauro Maia Laruccia³

O desenvolvimento de projetos, com o auxílio da engenharia de software e metodologia ágil SCRUM

Recebido em: 26/10/2012 - Aprovado em: 12/11/2012 - Disponibilizado em: 26/12/2012

Resumo

Engenharia de Software tem se mostrado uma grande ferramenta de trabalho para o auxílio no desenvolvimento de software. Dentre esses conceitos e metodologias encontra-se o SCRUM, metodologia ágil que facilita o gerenciamento do desenvolvimento, facilitando a comunicação da equipe, controle do tempo, distribuição da atividade, trazendo maior qualidade ao produto final.

Palavras-chave: Engenharia de Software, SCRUM, Desenvolvimento de Software, Metodologias Ágeis, Java.

Abstract

Software engineering has proved a great tool to aid work in software development. Among these concepts and methodologies is SCRUM, Agile methodology that facilitates the management of the development, facilitating team communication, time management, distribution of activity, bringing higher quality at the end product.

Keywords: Software Engineering; SCRUM; Software Development; Agile Methodologies; Java

1. Introdução

Há algum tempo, os softwares vem se tornando uma ferramenta imprescindível nas empresas. Esse avanço tem impactado,

não somente as grandes corporações, mas também as pequenas e médias empresas. O que acarretou um grande aumento da demanda de software e com isso, o conceito de Engenharia de Software

¹Leonardo Di Francescantonio - Graduando em Faculdade de Computação e Informática da Fundação Armando Álvares Penteado – FAAP. Rua Alagoas, 903, São Paulo, Brasil. E-mail: leonardo.francescantonio@gmail.com

² Fernando Luiz de Almeida - Professor Doutor da Faculdade de Computação e Informática da Fundação Armando Álvares Penteado – FAAP. Rua Alagoas, 903, São Paulo, Brasil. E-mail: flalmeida@faap.br

³ Mauro Maia Laruccia - Professor Doutor da Faculdade de Economia e Administração PUC/SP e da Fundação Armando Álvares Penteado - FAAP e Professor Pesquisador do Grupo de Pesquisa e Iniciação Científica, orientador da Oficina de Artigos das Faculdades Integradas Campos Salles. Rua Nossa Senhora da Lapa, 284, São Paulo 05072-000, Brasil. E-mail: mauro.laruccia@gmail.com

ganhou força, passando, inclusive, a ser visto como uma importante disciplina dos cursos relacionados a desenvolvimento de aplicações para computadores.

Conceitualmente, a Engenharia de Software compreende um processo, um conjunto de métodos (práticas) e ferramentas que possibilitam aos profissionais desenvolverem softwares de altíssima qualidade (PRESSMAN, 2011; SOMMERVILLE, 2011).

É comum existirem problemas entre desenvolvedores e clientes, ao decorrer do processo de criação e amadurecimento de projetos, devido à dificuldade na comunicação, prejudicando o desenvolvimento de softwares de qualidade dentro de prazos cada vez mais apertados. Com a Engenharia de Software, o desenvolvimento torna-se mais organizado e produtivo, pois, a partir da combinação de inúmeras técnicas, o processo de comunicação entre clientes e desenvolvedores é otimizado, resultando em softwares complexos e de qualidade, que respeitam os prazos de desenvolvimento.

Com base nos conceitos e métodos abordados em engenharia de software, foi proposto o desenvolvimento de um sistema para uma corretora de seguros, com o objetivo de gerenciar as informações referentes a planos e seguros saúde, abrangendo a adesão de pessoas físicas e pessoas jurídicas, Omicron Zeta GIPSS - Sistema de Gerenciamento de Informações de Plano e Seguro Saúde – é um sistema dividido em cinco seguimentos de informação: Empresa, Funcionário, Cliente Individual, Prestadores e Relatórios. Cada uma dessas frentes de atuação, com exceção dos Relatórios, conta com

funcionalidades de Cadastro, Edição e Consulta, ações essas que são executadas respeitando os perfis de acesso definidos pelo administrador do sistema, que além das funcionalidades citadas, possui o acesso e controle dos usuários.

Durante o desenvolvimento, foram colocadas em prática técnicas abordadas pela Engenharia de Software, tendo em mente um trabalho em conjunto com o cliente, que participou ativamente de todo o processo.

2. Metodologia

A construção do projeto foi orientada por alguns modelos de engenharia de software, que auxiliam na criação de um projeto de qualidade, são eles:

- Etapas do arcabouço, que orientam no desenvolvimento de software dividindo o mesmo em cinco etapas: Comunicação, Planejamento, Modelagem, Construção e Implantação;

- Filosofia do modelo espiral, que serviu de base para o desenvolvimento em conjunto com o cliente, sugerindo que o cliente faça parte da criação, motivando-o a se envolver com o projeto;

- Processo Unificado, RUP (Rational Unified Process), que orienta no desenvolvimento, apoiando-se nos melhores recursos e características dos modelos convencionais de processos de software;

- Documentação UML (Linguagem Unificada de Modelagem), que auxiliou na

criação de diagramas ilustrativos dos processos e da comunicação do software;

- Método Ágil de Gestão - Scrum, que auxiliou no controle e gestão do tempo e das atividades do projeto;

- Desenvolvimento em camadas, dividindo o desenvolvimento em três frentes: Camada de Apresentação, Camada Lógica e Camada de Acesso a Dados.

O projeto foi iniciado com conversas entre a corretora Jr&Marto Consultoria e Seguros (cliente) e a equipe de desenvolvimento, onde foram identificadas algumas necessidades e possibilidades de melhorias. Dentre elas, a dificuldade no gerenciamento das informações de clientes e prestadores de serviço.

Depois de mapear os pontos de melhoria, foi criado junto ao cliente um escopo de trabalho, que tem como principal ponto a informatização dos processos, criando uma ferramenta que facilite o gerenciamento e controle das informações.

Juntamente com o cliente foi desenvolvido o cronograma de trabalho, que possibilita estimar o prazo de entrega do sistema, prevendo reuniões periódicas para que sejam alinhadas ideias e sugestões relevantes ao desenvolvimento do projeto.

A documentação referente ao projeto foi desenvolvida baseada nos diagramas UML, ilustrando o projeto de acordo com os parâmetros definidos em conjunto com o cliente, elaborando Diagramas de Caso de Uso, Diagramas de Classe e Diagramas de Sequência, a fim de facilitar e agilizar o desenvolvimento do sistema.

Para demonstrar ao cliente o *layout* das telas do sistema foram desenvolvidos protótipos de tela (PRESSMAN, 2011), que foram submetidas à aprovação e teste do cliente, utilizando filmagens para melhor acompanhar as reações durante os testes, com o intuito de desenhar interfaces gráficas adequadas para facilitar o uso.

Como ferramenta de desenvolvimento foi utilizado o NetBeans (NETBEANS, 2012), por meio de linguagem Java, utilizada como ambiente de desenvolvimento das principais funcionalidade e telas do sistema, visando solucionar da melhor forma as necessidade do cliente.

Após a conclusão do desenvolvimento do projeto, o cliente passou por um treinamento, para possibilitar a melhor utilização do sistema. Por fim, o sistema foi implantado, e utilizado em produção.

3. Desenvolvimento

Ao iniciar a comunicação com o cliente foi possível identificar suas necessidades, que possuía as informações de clientes e prestadores de serviços, armazenadas fisicamente. Havia grande interesse por parte do cliente em buscar uma solução informatizada que se adequasse aos seus custos, uma vez que muitas das soluções existentes no mercado requerem grande investimento.

Observando essa realidade, foi proposta a criação de um software que ajudasse a gerenciar as informações utilizadas na corretora, para agilizar os processos e facilitar o acesso, trabalhando a informação de forma segura e integrada.

Na definição do escopo do projeto, ficou definida a criação de um sistema de cadastro e consulta de informações referentes a convênios e seguros saúde, com uma hierarquia de acesso que garantisse a segurança da informação, além de relatórios que facilitasse a visualização de resultados conforme a necessidade dos gestores.

Junto ao cliente foi elaborado um cronograma, no qual se contemplou o tempo estimado para o planejamento, construção, implantação e treinamento da equipe.

O trabalho foi dividido em etapas como é sugerido pelo arcabouço (PRESSMAN,

2011), utilizando a metodologia SCRUM (SOMMERVILLE, 2011).

Conforme ilustrado na Figura 1, foi definido que cada funcionalidade do sistema seria equivalente a um *Product Backlog*, sendo cada parte dessa funcionalidade, tais como Interface, Regra de Negócio, Acesso aos Dados, tidos como um *Sprint Backlog*, adotando o tempo médio de uma semana para cada Sprint, resultando em entregáveis (*Deliverable*) semanais. Entretanto, devido à composição da equipe, não houve necessidade de reuniões diárias (*Daily Scrum*).

Figura 1: Ciclo SCRUM

Foi criada uma *Task Board*, para auxiliar no controle das atividades, conforme ilustrado na Figura 2. A mesma ficou com a seguinte estrutura:

•**TO DO:** *Sprints Backlogs* não iniciados, atividades que estão aguardando para estar na etapa de desenvolvimento;

•**DOING:** Fase de desenvolvimento, nesse momento os *Sprints Backlogs* estão em construção;

•**TO VERIFY:** Momento em que as funcionalidades são submetidas à validação do cliente, nessa etapa, o cliente entrava em contato com a

interface, e vídeos eram gravados, durante a execução, como é sugerido no processo de prototipação;

•**DONE!**: Funcionalidade concluída, última fase do desenvolvimento da funcionalidade, *Deliverable*, após a validação a funcionalidade estava finalizada.

•**IMPEDIMENTS**: Nesse espaço do quadro, eram identificados os impedimentos, e dificuldades que impactavam a sequência do desenvolvimento.

•**TO DISCUSS**: Itens a serem discutidos, ideias que agregariam funcionalidade, porém caso não houvesse não impactaria no resultado final, seriam tratados como segundo plano, dentro das possibilidades do cronograma.

•**UNPLANNED ITEMS**: Espaço para sinalizar funcionalidades novas que não haviam sido identificadas no escopo inicial, funcionalidades de menor prioridade.

TO DO	DOING	TO VERIFY	DONE !	IMPEDIMENTS
LOGIN ESQUECI MINHA SENHA			VERIFICAÇÃO USUÁRIO INTERFACE GRÁFICA	PROBLEMAS DE USUÁRIO
MENU PROGRESS BAR			INFO USUÁRIO INTERFACE USUÁRIO SELOUS	
BUSCAR USUÁRIO		INTERFACE	PESQUISA USUÁRIO	
EDITAR USUÁRIO		INTERFACE	EDIÇÃO	
CONSULTAR USUÁRIO		INTERFACE	CONSULTA USUÁRIO	
CADASTRAR FUNCIONÁRIO INTERFACE GRÁFICA USUÁRIO				
BUSCAR FUNCIONÁRIO INTERFACE PESQUISA				
EDITAR FUNCIONÁRIO INTERFACE EDIÇÃO				
CONSULTAR FUNCIONÁRIO INTERFACE CONSULTA				
				TO DISCUSS EU QUERO CRIAR NA TABELA DE BUSCA
				UNPLANNED ITEMS SELECIONAR MODO DE UMA PRESTADORA POR EMPRESA

Figura 2: Task Board

Buscando uma melhor forma de observar a evolução no desenvolvimento, foi adotado o *Burndown Chart*, o qual teve fundamental importância na identificação de problemas durante desenvolvimento, alertando a equipe do projeto sobre em que momento era necessária maior atenção para o cumprimento do prazo.

Como é possível observar na Figura 3, que apresenta um gráfico onde o tempo é

representado no eixo X, e as funcionalidades pendentes no eixo Y, as linhas representadas no gráfico referem-se ao desenvolvimento planejado (linha A), e ao desenvolvimento real (linha B), ou seja, quando a linha B está abaixo da linha A, entende-se que o desenvolvimento está dentro do período planejado, no caso inverso o desenvolvimento está atrasado.

Figura 3: *Burndown Chart*

Em uma segunda fase do projeto, foram elaborados diagramas para a criação da documentação UML do projeto, utilizando a ferramenta de modelagem StarUML (STARUML, 2012).

O Diagrama de Caso de Uso foi utilizado para definir quais os níveis de

acesso seriam criados para atender as necessidades do cliente e definir as funcionalidades do sistema. A Figura 4 ilustra o os níveis de acesso para as funcionalidades referentes ao módulo de Pessoa Física.

Figura 4: Caso de Uso – Módulo Empresa

Os níveis de acesso foram definidos em uma hierarquia de 1 a 5, sendo cinco, o nível mais alto da cadeia, equivalente ao nível de administrador.

O Diagrama de Classes (Figura 5), que apresenta de forma simplificada como estão distribuídas as classes do sistema, e como ocorre o relacionamento entre elas.

Figura 5: Caso de Classe Simplificado

Todo o sistema foi desenvolvido a partir do conceito de Orientação a Objeto (DEITEL e DEITEL, 2005), respeitando uma arquitetura de camadas, para um

melhor aproveitamento das classes do sistema, evitando redundâncias e acesso descontrolado ao banco de dados, conforme ilustra a Figura 6.

Figura 6: Arquitetura de Camadas

O Diagrama de Sequência retrata de forma genérica como está o fluxo da informação dentro do sistema. A Figura 7 ilustra o processo de cadastro, desde o

cadastramento da informação realizada pelo usuário, até o momento em que o sistema retorna uma mensagem com a confirmação do cadastro.

Figura 7: Caso de Classe – Cadastro (Genérico)

Em paralelo a elaboração dos diagramas UML, foi criado o Modelo Entidade Relacionamento, que auxiliou na criação do banco de dados que foi estruturado

utilizando a ferramenta MySQL Workbench (ORACLE CORPORATION, 2012) conforme ilustra a Figura 8.

Figura 8: MER - Banco de Dados

Como já mencionado anteriormente, o desenvolvimento do sistema, foi realizado com a ferramenta NetBeans (NETBEANS, 2012), utilizando a linguagem de programação Java, e em conjunto com o cliente, foram criadas interfaces gráficas e funcionalidades para melhor solucionar as necessidades de gerenciamento das informações da corretora.

A Figura 9 é uma ilustração da tela de *Login* do sistema, que além da funcionalidade de validação de usuários, conta com um recurso, para configuração de IP e PORTA do servidor para acesso ao banco de dados.

Figura 9: Tela Login

Após a validação do usuário, a tela Menu, (Figura 10) é construída conforme o nível de acesso do usuário. A árvore das funcionalidades (canto superior esquerdo, Figura 10-A) é estruturada apresentando as funcionalidades de acordo com a hierarquia de acessos, além da barra de menus, que segue a mesma lógica, além de possuir alguns itens apenas para o administrador do sistema. No canto inferior esquerdo (Figura 10-B), estão apresentadas as informações do usuário que está logado no sistema.

Figura 10: Tela Menu

Na Figura 11, está ilustrado a tela Buscar Empresa, que segue o padrão dessa funcionalidade independente da informação em questão.

Nessa tela, a busca pode ser realizada de duas formas, tanto através de CNPJ,

quanto pela Razão Social. Ao executar a busca, a funcionalidade retorna em lista os itens encontrados. Na parte inferior da interface, estão as ações que o usuário pode realizar (consultar, editar, selecionar prestadora ou cancelar).

Figura 11: Tela Buscar Empresa

A tela apresentada a seguir, Figura 12, ilustra a interface para o cadastramento de novos funcionários, essa tela conta, com uma lógica para a validação do CPF, além de identificar os campos tidos como de

preenchimento obrigatório, foi criada uma inteligência para a montagem do *ComboBox* CIDADE, onde a partir do estado selecionado, é realizada uma busca no banco de dados para o preenchimento.

Funcionários.Cadastrar

NOME* <input type="text"/>	ENDEREÇO <input type="text"/>
CPF* <input type="text"/> - <input type="text"/> - <input type="text"/> RG <input type="text"/> - <input type="text"/> - <input type="text"/>	NÚMERO <input type="text"/> COMPLEMENTO <input type="text"/>
EXPEDIDOR <input type="text"/> DATA EXPEDIÇÃO <input type="text"/>	CEP <input type="text"/> BAIRRO <input type="text"/>
NACIONALIDADE <input type="text" value="BRASILEIRO(A)"/>	ESTADO <input type="text"/> CIDADE <input type="text"/>
DATA DE NASCIMENTO* <input type="text"/>	PRESTADORA* <input type="text"/>
ESTADO CIVIL <input type="text" value="SOLTEIRO(A)"/>	PLANO* <input type="text"/>
NOME DA MÃE* <input type="text"/>	CARTEIRA CONVÊNIO* <input type="text"/>
EMPRESA* <input type="text"/>	DATA EMISSÃO CARTEIRA <input type="text"/>
CARGO <input type="text"/>	INÍCIO VIGÊNCIA* <input type="text"/>
DATA DE ADMISSÃO* <input type="text"/> STATUS <input type="text" value="ATIVO"/>	BANCO <input type="text"/>
	AGÊNCIA <input type="text"/> CONTA <input type="text"/>

* Campo obrigatório

CADASTRAR CANCELAR

Figura 12: Tela Cadastrar Funcionário

As telas de consulta e edição foram padronizadas conforme a tela de cadastramento de cada módulo, visando facilitar a identificação dos campos.

Pensando na parte gerencial do sistema, foram desenvolvidos relatórios que possibilitam a impressão das informações, utilizando da ferramenta JasperReport

(JASPERFORGE, 2012) e iReport (JASPERFORGE, 2012) para o NetBeans.

Cada módulo contém uma tela de consulta, a qual possui um botão para a geração de um relatório com as informações que estão visíveis na tela, como ilustra a Figura 13.

DADOS DETALHADOS DO FUNCIONÁRIO				OoZz	JR & MARTO
LEONARDO DI FRANCISCANTONIO OLIVEIRA					
CPF: 356.852.178-10	RG: 47.883.815-3	ENDEREÇO: RUA ALFREDO MARGARA			
EXPEDIDOR: SSP/SP	DATA EXPEDIÇÃO: 04/10/2010	NÚMERO: 8	COMPLEMENTO: APTO 22		
NACIONALIDADE: ITALIANA		CEP: 02064-120	BARRIO: FREGUESIA DO Ó		
DATA DE NASCIMENTO: 11/06/1981		ESTADO: SP	CIDADE: SÃO PAULO		
ESTADO CIVIL: SOLTEIRO(A)		PRESTADORA: AVIMED SAÚDE			
NOME DA MÃE: ANA MARIA DI FRANCISCANTONIO		PLANO: ANPSAF			
EMPRESA: PETROBRAS		CARTESIA CONVÊNIO: 34104			
CARGO: GERENTE		INÍCIO VIGÊNCIA: 04/03/2011	DEPENDENTES: 4		
DATA DE ADMISSÃO: 04/10/2010	STATUS: ATIVO	BANCO: 000 - BANCO SANTANDER S.A.			
		AGÊNCIA: 1234	CONTA: 1234124		

Terça-feira 08 Novembro 2011

Page 1 of 1

Figura 13: Relatório de Dados Detalhados do Funcionário

Na Figura 14 é apresentado um relatório gerencial, que possibilita uma análise das movimentações da carteira de clientes,

utilizando de diferentes filtros para obtenção do relatório como, por exemplo, data de adesão.

EMPRESA POR PERÍODO				OoZz	JR & MARTO
PERÍODO: 01/04/2011 a 30/04/2011					
DATA DO REGISTRO	RAZÃO SOCIAL	CNPJ	ID		
07/04/2011	PETROBRAS	33.000.167/0001-01	1		
12/04/2011	FAAP	01.451.431/0001-89	2		
12/04/2011	VALE DO RIO DOCE	04.867.894/0001-42	3		
13/04/2011	LILU DOCES	01.952.221/0001-54	4		
15/04/2011	HELOÍSA MODAS	45.004.421/0001-29	5		
15/04/2011	OMICRÓN ZETA	11.726.168/0001-03	6		
18/04/2011	DOCE SEGREDO	96.834.294/0001-10	7		
18/04/2011	MANO SALGADOS	95.212.617/0001-96	8		
25/04/2011	QUINTAL DO PASTEL	78.450.085/0001-05	9		
27/04/2011	SMART SOFT	82.174.065/0001-18	10		

Total de empresas registradas no período: 10

Sexta-feira 18 Novembro 2011

Page 1 of 1

Figura 14: Relatório de Adesões de Empresas por período

Após o fim do desenvolvimento, o sistema foi implantado, e o cliente foi devidamente treinado, finalizando assim o projeto, conforme definição das etapas do arcabouço (etapa de Implantação) (PRESSMAN, 2011).

4. Resultado

Resultante desta pesquisa científica envolvendo a tecnologia Java, banco de dados MySQL e gerenciamento de projetos com a metodologia ágil Scrum, foi criado o Omicron Zeta GIPSS - Sistema de

Gerenciamento de Informações de Plano e Seguro Saúde, sistema para a automatização do gerenciamento de informações utilizadas na Jr&Marto Consultoria e Seguros, que até então eram realizados de forma manual.

Como já mencionado anteriormente, o processo de desenvolvimento deste projeto contou com o auxílio da Metodologia Scrum, que por sua vez, proporcionou maior velocidade no desenvolvimento e comunicação contínua entre a equipe de desenvolvimento e o cliente, evitando assim, surpresas com o resultado final de cada incremento do software. Além disso, este processo auxiliou na priorização das atividades e na elaboração de planos para mitigar os riscos do projeto (pontos muito exaltados pelo cliente). Porém, ao fazer uso de uma metodologia de gerenciamento há de certa forma, uma burocratização do processo de desenvolvimento, pois acaba provocando um aumento das atividades no processo de desenvolvimento. Mesmo assim, vale ressaltar que a inclusão deste processo, não trouxe prejuízos ao projeto.

Com o sistema em pleno funcionamento, a corretora atingiu seu

objetivo, proporcionando aos seus clientes um atendimento de qualidade, além de um acesso rápido as informações do sistema.

5. Conclusão

Durante todo o processo de desenvolvimento do sistema Omicron Zeta GIPSS - Sistema de Gerenciamento de Informações de Plano e Seguro Saúde, a metodologia SCRUM, junto à Engenharia de Software auxiliaram na modelagem e no processo de comunicação com o cliente.

A Metodologia Ágil Scrum foi de grande relevância no controle e gestão do desenvolvimento, pois através de suas ferramentas, foi possível acompanhar a evolução do projeto, além de auxiliar na tomada de decisão para solucionar os eventuais atrasos no desenvolvimento.

As técnicas do arcabouço utilizadas no desenvolvimento auxiliaram para uma comunicação eficaz com o cliente, facilitando o mapeamento das necessidades e a validação das etapas desenvolvidas, que por sua vez, facilitou o alinhamento com as necessidades identificadas.

Referências Bibliográficas

- COHN, M. **Desenvolvimento de Software com Scrum**: Aplicando métodos ágeis com sucesso. Porto Alegre: Bookman, 2011.
- DEITEL, H. M.; DEITEL, P. J. **Java**: como programar. São Paulo: Pearson Prentice Hall, 2005.
- GONÇALVES, E. **Desenvolvendo Relatórios Profissionais Com iReport™ para Netbeans IDE**. Rio de Janeiro: Ciência Moderna, 2009.

- JASPERFORGE. iReport. **Jasperforge Community**, 2012. Disponível em: <http://jasperforge.org/website/ireportwebsite/IR%20Website/ir_download.html?header=project&target=ireport>. Acesso em: 1 Setembro 2012.
- JASPERFORGE. Plugins. **Jasperforge Community**, 2012. Disponível em: <<http://community.jaspersoft.com/search?query=plugins%20esp%20frs%20optional%20download>>. Acesso em: 3 Setembro 2012.
- NETBEANS. NetBeans IDE 7.2. **NetBeans Community**, 2012. Disponível em: <<http://netbeans.org/downloads/index.html>>. Acesso em: 1 Setembro 2012.
- ORACLE CORPORATION. MySQL Workbench 5.2.42. **MySQL**, 2012. Disponível em: <<http://www.mysql.com/downloads/workbench>>. Acesso em: 2 Setembro 2012.
- PRESSMAN, R. S. **Engenharia de software: uma abordagem profissional**. Porto Alegre: Mc Graw Hill, 2011.
- SIERRA, K.; BATES, B. **Use a Cabeça Java**. Rio de Janeiro: Alta Books, 2010.
- SOMMERVILLE, I. **Engenharia de Software**. São Paulo: Pearson Education - Br, 2011.
- STARUML. StarUML Download. **The Open Source UML/MDA Platform**, 2012. Disponível em: <<http://staruml.sourceforge.net/en/download.php>>. Acesso em: 1 Setembro 2012.