

Aplicabilidade e influência da metodologia Scrum em desenvolvimento de sistemas

Alex Ribeiro, Douglas Ruani, Marcelo Minato, Rafael Vidoto, Larissa Pavarini da Luz

Curso de Tecnologia em Análise e Desenvolvimento de Sistemas – FATEC – Campus de Garça

014 3471-4723 – Garça – SP – Brasil

alex_tiago_15@hotmail.com, douglasruani@hotmail.com, marcelo.minato@gmail.com,
rafaelvidotodeazevedo@gmail.com, larissa.luz01@fatec.sp.gov.br

Abstract. *This article aims to explain in a practical way the use of the Scrum methodology in systems development. To understand the applicability of the methodology will be considered the architecture of a real system developed by a team composed of four members. With the present scenario defined the applicability of the Scrum methodology is adequate because it does not prescribe how the product should be developed, but aims to promote discovery, inspection and adaptation essential elements considering a newly formed team starting from the idea of software responsible for the management of hospital services orders. In the course of the present article will be presented the maturation of the initial idea, the development, the integration of the team, and the possible changes in the scope of the project in order to diagnose in a simple and objective way the influence of the Scrum methodology in each process of the project until its effective delivery.*

Resumo. *O presente artigo visa explicar de forma prática o uso da metodologia Scrum em desenvolvimento de sistemas e a sua influência no cenário a qual aplica-se. Para compreensão da aplicabilidade da metodologia será considerado a arquitetura de um sistema real desenvolvido por uma equipe composta por quatro integrantes. Com o presente cenário definido a aplicabilidade da metodologia Scrum se mostra adequada pois a mesma não prescreve como o produto deve ser desenvolvido, mas visa promover descoberta, inspeção e adaptação elementos essenciais considerando uma equipe recém-formada que tem como ponto de partida apenas a ideia de um software responsável pelo gerenciamento de ordens de serviços hospitalares. Ao decorrer do presente artigo será apresentado o amadurecimento da ideia inicial, o desenvolvimento, a integração do time, e as possíveis mudanças no escopo do projeto afim de diagnosticar de forma simples e objetiva a influência da metodologia Scrum em todas as vertentes do projeto até sua efetiva entrega.*

1. Introdução

Scrum, nome derivado de uma atividade que ocorre durante um jogo de *rugby*, é um modelo ágil de processo desenvolvido por Jeff Sutherland e sua equipe no início da década de 90. Podemos definir *Scrum* como um *framework* estrutural utilizado para o gerenciamento do desenvolvimento de produtos, evidenciando que a metodologia não é utilizada como processo ou técnica para construção de produtos, mas abre a possibilidade de combinação ou complemento por diversos processos, métodos ou técnicas considerando o contexto específico para sua aplicabilidade.

Em ambientes relacionados a Tecnologia da Informação é comum e bastante disseminado a cultura dos métodos ágeis, porém pouco é discutido sobre sua real influência, considerando a migração de uma sociedade industrial pautada pelos métodos propostos por Taylor e Fayol para um novo cenário de produção que exige agilidade, multidisciplinaridade, foco em pessoas onde todos se tornaram clientes, sejam internos ou externos, e a busca de ambos se resume a uma palavra: excelência.

Segundo Chiavenato (2014) este processo transitório se resume na evolução de uma sociedade industrial para uma sociedade da informação¹ onde automaticamente é provocado um novo modelo mental gerando questionamentos quanto a maneira de se enxergar o individualismo e o coletivo, conforto e a mudança, multidisciplinaridade e especialidades, felicidade e resultado, qualidade e entrega e as metodologias ágeis vem de maneira sutil intervir em tais questões disseminando uma cultura participativa levando o indivíduo a dedicar-se ao desenvolvimento de algo que traga orgulho para si e para todos os envolvidos.

Considerando a abordagem proposta pelas metodologias ágeis o objetivo do presente artigo é em um cenário real verificar as influências de sua aplicabilidade, para isso foi definido a tarefa do desenvolvimento de um software responsável pelo gerenciamento de ordens de serviços em um ambiente hospitalar por uma equipe composta por quatro pessoas, e para o gerenciamento do projeto foi adotado especificamente a metodologia *Scrum*, levando em conta sua fácil aplicabilidade e grande difusão no setor de tecnologia da informação.

A partir de tal cenário será explanado a aplicabilidade da metodologia *Scrum* e sua influência em todos os agentes e aspectos inerentes ao cumprimento da tarefa proposta.

2. Uma metodologia ágil - *Scrum*

Em 2001 um grupo formado por Kent Beck e outros renomados desenvolvedores assinaram o Manifesto para o desenvolvimento ágil de software², que possui as seguintes diretrizes como base: indivíduos e interações devem ser mais valorizados do que processos e ferramentas, software em funcionamento deve ser mais valorizado do

¹ CHIAVENATO, Idalberto. **Introdução à Teoria Geral da Administração**. Barueri: Manole, 2014

² Manifesto para o desenvolvimento ágil de software: Disponível em: <<https://www.manifestoagil.com.br/>>

que documentação abrangente, colaboração com cliente deve possuir mais valor que negociação de contratos e resposta a mudanças devem estar acima de seguir um plano.

Segundo Pressman (2011, p.81)

A engenharia de software ágil combina filosofia com um conjunto de princípios de desenvolvimento. A filosofia defende a satisfação do cliente e a entrega de incremental prévio; equipes de projetos pequenas e altamente motivadas; métodos informais; artefatos de engenharia de software mínimos e, acima de tudo, simplicidade no desenvolvimento geral. Os princípios de desenvolvimento priorizam a entrega mais que a análise e projeto (embora essas atividades não sejam desencorajadas); também priorizam a comunicação ativa e contínua entre desenvolvedores e clientes.

Considerando tais preceitos uma das metodologias que incorpora a ideia tratada no manifesto ágil é o *Scrum* que tem como base as atividades propostas no desenvolvimento estrutural considerando requisitos, análise, projeto, evolução e entrega.

Para cada atividade a ser realizada durante o ciclo de desenvolvimento a metodologia *Scrum* propõe uma serie de “rituais”, sendo eles definição do *product backlog* onde é definido as funcionalidades desejadas e elencadas por importância, normalmente definidos em uma reunião de quatro horas, uma vez definido o *product backlog* é realizado o *sprint planing* onde é estruturado o planejamento para cumprimento das tarefas propostas, no planejamento a tarefa macro é fragmentada em pequenas tarefas chamadas *history* onde é definido o esforço necessário para a execução da mesma, para que a equipe tenha ciência de todo processo diariamente deve ser realizada uma reunião de quinze minutos onde as seguintes questões devem ser respondidas:

- O que você fez desde a última reunião?
- Você tem algum obstáculo?
- O que você vai fazer antes da próxima reunião?

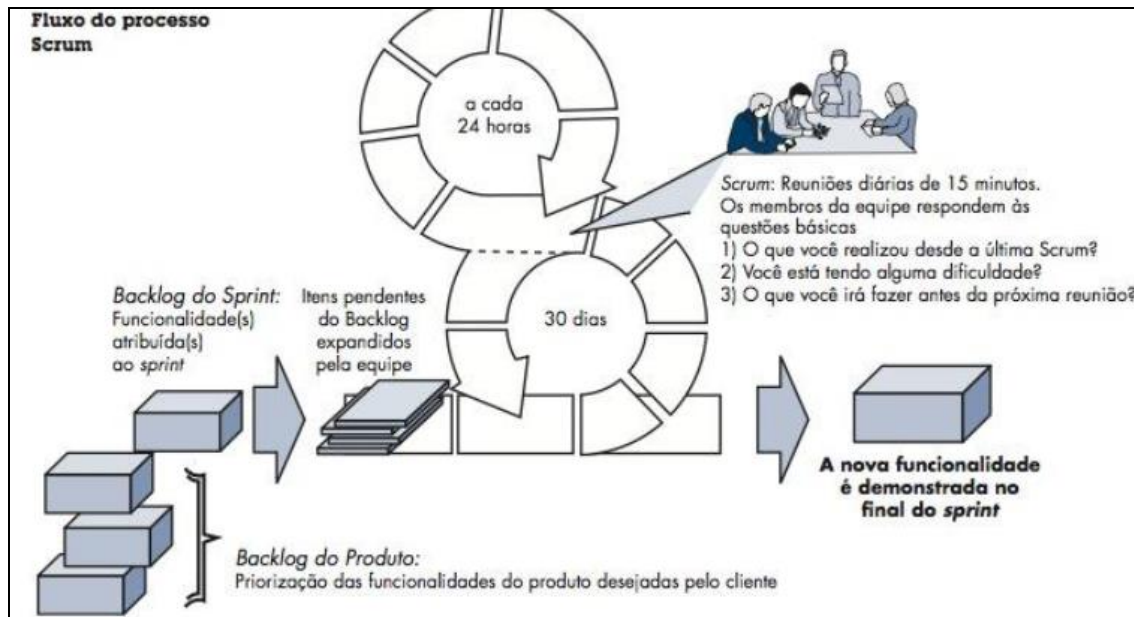
Todo processo de trabalho para cumprimento das *historys* deve ser executado dentro de um período de tempo normalmente de duas semanas a trinta dias, denominado *sprint*.

Para cada encerramento de *sprint* deve ser realizada uma reunião de fechamento onde é revisado o trabalho levantando pontos positivos e negativos, assim como o desenvolvimento efetuado até aquele momento, que deve ser apresentado partindo da premissa que sugestões de melhorias são bem-vindas.

Todos os *sprints* realizados devem culminar na entrega do *backlog* proposto, ou seja, da tarefa macro, onde enfim é realizado o encerramento discutindo testes de software, documentação, treinamentos, pontos positivos e negativos.

Abaixo é possível visualizar na Figura 1 um modelo prático da aplicabilidade da metodologia Scrum e os rituais propostos.

Figura 1: Fluxo do processo Scrum.



Fonte: PRESSMAN, 2011 (p.96)

Dentro do processo descrito acima a metodologia *Scrum* propõe a definição de três papéis, sendo eles:

Product Owner: que deve ser o responsável por apresentar os interesses dos *stakeholders*, este é responsável pelo *product backlog*, e deve definir os fundamentos iniciais do projeto.

Scrum Master: responsável em garantir a aplicabilidade do *Scrum*, conduzir reuniões e apresentações, deve garantir que o time envolvido saiba trabalhar com *Scrum* e esteja executando o seu papel dentro do time.

Equipe ou Time: responsável por desenvolver as tarefas propostas em cada *sprint*, esta deve ser auto gerenciável e organizável para garantir o cumprimento das mesmas.

3. Caracterização do Ambiente

O ambiente para aplicação da metodologia *Scrum* no presente artigo foi acadêmico, inicialmente foi definido um grupo de trabalho composto por quatro alunos onde os mesmos deveriam utilizar quatro aulas semanas para o desenvolvimento de um projeto de *software*, o processo de desenvolvimento seria acompanhado pelo professor que basicamente assumiria o papel de *stakholder*.

A definição do *software* a ser desenvolvido ficou a critério da equipe de trabalho, onde definiu se o desenvolvimento de um sistema para gestão de ordens de serviço em um ambiente hospitalar.

O prazo para entrega do *software* em funcionamento foi previamente estipulado considerando quatro meses de trabalho iniciando em março e encerrado em junho.

A partir de tal cenário deu-se início a aplicabilidade da metodologia *Scrum* adaptando a mesma a realidade a qual estava inserida a equipe de trabalho.

4. Correlação da metodologia *Scrum* com o ambiente

O relacionamento das práticas realizadas no *Scrum* fora inserida no ambiente em questão a partir da definição do *software* a ser desenvolvido, onde posteriormente como previsto nas práticas da metodologia foi realizada uma reunião para definição do *product backlog*, na reunião foi realizado o levantamento de todos os requisitos do sistema e o ciclo de desenvolvimento.

Em um primeiro momento é perceptível o objetivo de integração e pertença considerando a situação proposta pelo *Scrum*, onde toda equipe recebe o projeto, abre-se espaço para discussão onde todos os envolvidos são encorajados a falar, retirando a mentalidade do mandar e executar gerando por sua vez uma mentalidade participativa.

Para controle efetivo do ciclo de desenvolvimento a equipe de trabalho optou na utilização da plataforma Trello, uma plataforma web gratuita que possui a finalidade de gerenciar graficamente os processos inerentes a um projeto, a ferramenta atendeu todas as necessidades da equipe e foi de extrema importância para a aplicabilidade da metodologia *Scrum* pois todo processo de trabalho foi consolidado na mesma e toda equipe conseguia obter uma visão macro do que estava ocorrendo durante determinados ciclos de trabalho.

O modelo de gerenciamento das tarefas a serem cumpridas vem de encontro com o princípio de equipes auto gerenciáveis e auto organizáveis, sendo visível que não existe a figura de um gerente pois todos os integrantes têm acesso a “fiscalizar” o andamento da tarefa de qualquer integrante, considerando o sentimento de pertença do time é perceptível na equipe que o processo de trabalho de um integrante afeta o desempenho de todo time, onde na verdade todos acabam assumindo o papel de gerente.

Definido o *backlog* foi realizada a reunião de abertura do *sprint 1* que contemplava a entrega de toda diagramação referente as funcionalidades do software, o ciclo de trabalho do primeiro *sprint* foi estipulado dentro do período de um mês, onde completo o período foi realizada a reunião de fechamento do *sprint 1* demonstrando as tarefas cumpridas dentro do período e relacionando os pontos positivos e negativos dentro do ciclo de trabalho. Durante as aulas foram realizadas as reuniões diárias previstas no *Scrum*.

Em tal momento todo time entrega o trabalho executado dentro do ciclo previsto, retirando a responsabilidade de apenas uma pessoa pelo sucesso ou fracasso do projeto. Outro ponto relevante é o levantamento dos pontos positivos e negativos onde é possível gerar um histórico para que a equipe aprenda com os erros e continue com os acertos princípio este previsto pela metodologia.

Também é importante pensarmos nos fatores humanos gerados por este ritual de fechamento, onde todos de forma pública são encorajados a levantar pontos positivos e negativos, gerando por sua vez uma situação de exposição coletiva levando o indivíduo

a pensar sobre a qualidade do seu trabalho e desempenho no projeto, considerando que tal fator pode ser exposto como ponto negativo ou positivo.

Dando sequência ao desenvolvimento foi realizado a abertura do *sprint 2* onde definiu se como tarefa a entrega do layout de todas as telas que iriam compor o sistema, o prazo para entrega das tarefas propostas foi de um mês. Encerrado o ciclo de trabalho foi realizado a reunião de fechamento de *sprint* com o levantamento dos pontos positivos e negativos referentes ao cumprimento da tarefa.

A partir de tal momento percebe se a integração do time considerando que todos sabem exatamente o que está ocorrendo no projeto, e conhecem a maneira de trabalho de cada indivíduo.

A partir do *sprint 3* iniciou se o trabalho técnico do projeto onde como tarefa foi definido a entrega de toda estruturação do banco de dados utilizado pelo *software*, os processos previstos no *Scrum* foram seguidos durante o ciclo de trabalho. Durante a reunião de fechamento do *sprint 3* o *stakholder* sugeriu uma alteração no escopo do projeto, toda estrutura estava considerando o ambiente *desktop*, porém foi sugerido a aplicação do *software* para a plataforma web, considerando que toda mudança é bem-vinda na metodologia *Scrum* a sugestão entrou para o backlog a ser discutido no *sprint* seguinte.

A mudança por sua vez normalmente causa um certo nível de desconforto, porém é importante observar que a mesma é discutida e não imposta, cabendo a equipe analisar o seu ritmo de trabalho e nível de conhecimento técnico para no momento oportuno aplicar a mudança.

O *sprint 4*, foi definido como ultimo *sprint* onde a tarefa a ser cumprida foi a entrega efetiva do software com todas as suas funcionalidades em operação. O *sprint* foi conduzido conforme as normas da metodologia *Scrum* e devidamente encerrado.

Ao realizar a reunião de fechamento foi identificado que uma tarefa proposta não foi cumprida, a análise de viabilidade para migrar o software para plataforma web, sendo assim como previsto na metodologia a tarefa foi inclusa no *backlog* para análise posterior considerando que o objetivo central do projeto foi atingido dentro do ciclo de trabalho de quatro *sprints*.

Considerando o exposto até o momento podemos afirmar que a aplicabilidade da metodologia *Scrum* gerou um ambiente humanizado para o desenvolvimento do produto através dos rituais propostos, que embora sejam de simples aplicação possuem grande influência nas pessoas que são os agentes executores, tais influências são geradas devido a situações de metas compartilhadas, alto nível de comunicação entre os indivíduos, situações de exposição, e a aplicação de tais situações de forma repetitiva acaba se tornando hábito para os indivíduos envolvidos gerando por sua vez um novo modelo mental quanto a forma de se trabalhar.

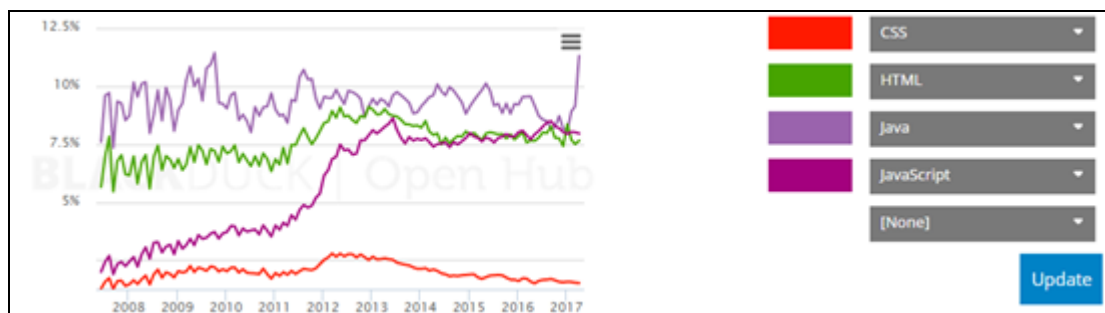
5. Desenvolvimento

Além da aplicabilidade da metodologia *Scrum* que basicamente gerencia e define os ciclos de trabalho, no contexto geral do projeto para cumprimento das tarefas foram

realizadas pesquisas e utilizado *frameworks* específicos para o desenvolvimento propriamente dito.

Inicialmente foi definido a linguagem de programação a ser utilizada, para tal definição foi considerado a relevância das linguagens em uso no ano de 2017 conforme mostra a Figura 2.

Figura 2: Relevância linguagem de programação



Fonte: BLACK DUCK, 2017

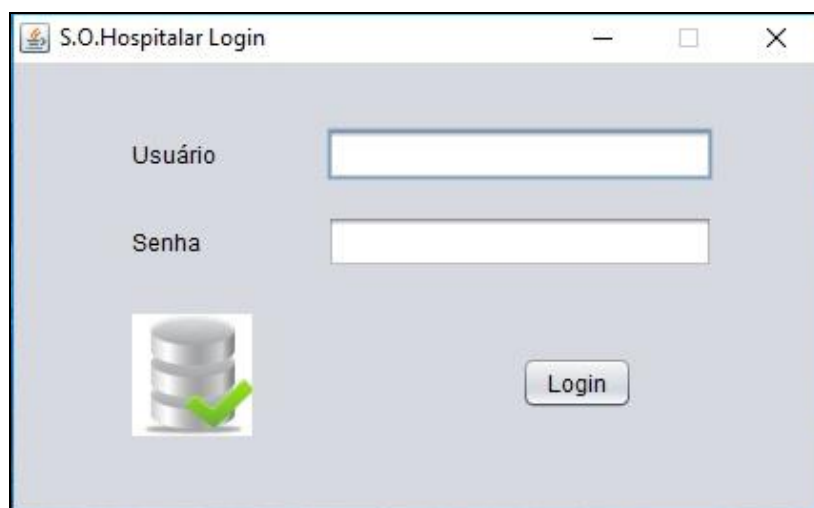
O gráfico demonstra que no período específico de junho de 2017 a linguagem de programação em maior evidência é o Java, como métrica foi realizado um comparativo entre as linguagens de programação que os membros da equipe possuem domínio. Considerando a relevância da linguagem e a IDE para desenvolvimento já configurada no ambiente acadêmico optou-se pela utilização do Java.

Como IDE para desenvolvimento foi definido o *Netbeans* considerando a familiaridade de todos da equipe com a IDE.

Para arquitetura do banco de dados foi utilizado o *MySQL Workbench* também devido a familiaridade da equipe.

Aliando o desenvolvimento à metodologia *Scrum* foi possível a criação do software apresentado nas Figuras (3 a 7).

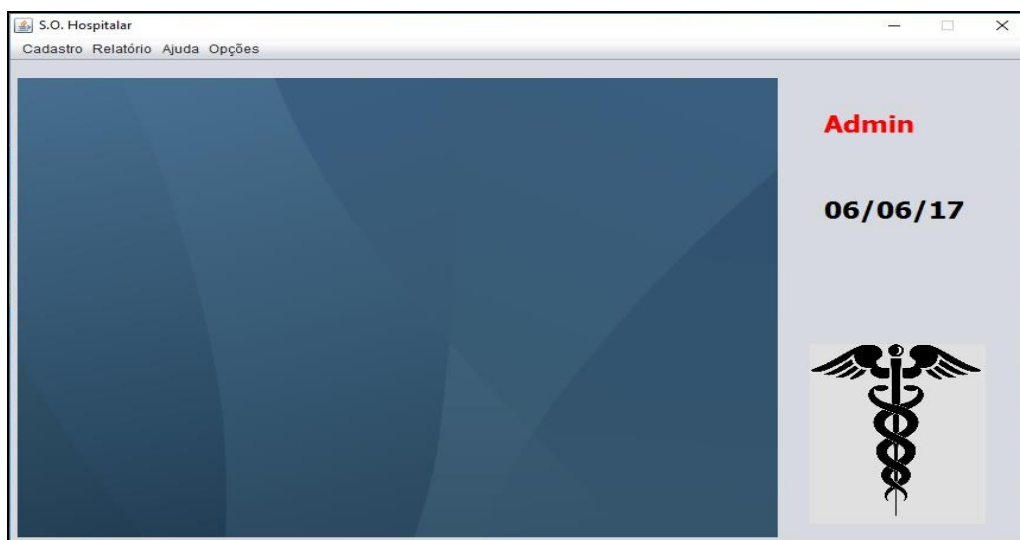
Figura 3: Tela de Login



Fonte: do próprio autor

A tela inicial do software basicamente libera o acesso ao banco de dados para usuários já cadastrados, a mesma também indica a conexão com o banco de dados, caso a mesma esteja inoperante é demonstrado uma figura vermelha, em tal condição o acesso não é liberado.

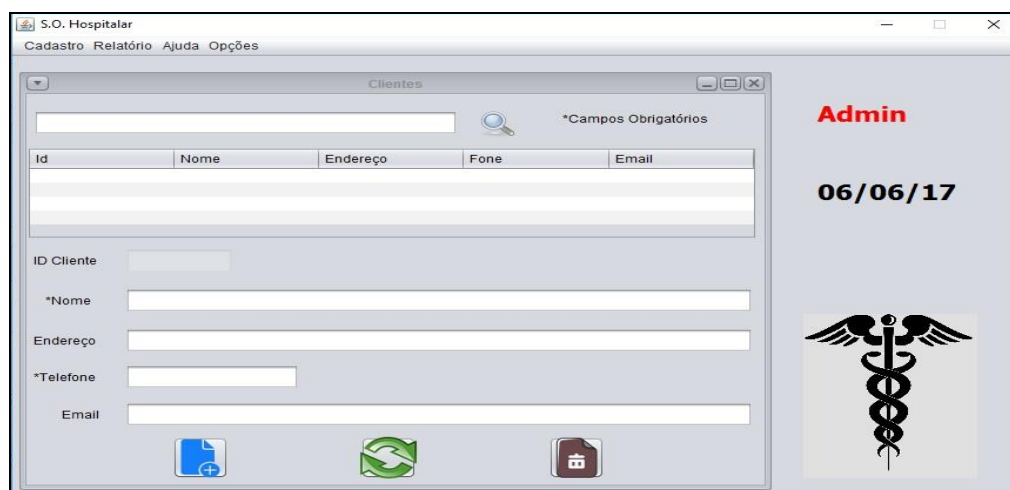
Figura 4: Tela Inicial



Fonte: do próprio autor

A segunda tela do sistema refere-se ao ambiente de trabalho propriamente dito, fornecendo as possibilidades de cadastros, relatórios, ajuda para manuseio do sistema e opções de customização.

Figura 5: Tela de cadastro de clientes



Fonte: do próprio autor

Em um ambiente onde é possível a utilização de múltiplas telas simultaneamente como demonstra a figura acima é possível realizar cadastros de usuários solicitantes de serviços e realizar possíveis alterações ou exclusões em tais cadastros.

Figura 6: Tela de cadastro de O.S

The screenshot shows a web application window titled 'S.O. Hospitalar' with a menu bar containing 'Cadastro', 'Relatório', 'Ajuda', and 'Opções'. The main interface is divided into several sections:

- Top Left:** Fields for 'Nº OS' and 'Data'. Below them are radio buttons for 'Orçamento' (selected) and 'OS'. A 'Situação' dropdown menu is set to 'Na bancada'.
- Top Right:** A 'Cliente' section with a search bar labeled '*Id' and a table with columns 'Id', 'Nome', and 'Fone'.
- Bottom Left:** Input fields for '*Equipamento', '*Defeito', 'Serviço', and 'Técnico'. A 'Valor Total' field shows '0'.
- Bottom Center:** A row of five icons: a blue folder with a plus sign, a yellow magnifying glass, a green circular arrow, a red building, and a printer.
- Right Sidebar:** Displays the user role 'Admin' in red, the date '06/06/17', and a medical symbol (Rod of Asclepius).

Fonte: do próprio autor

Dando continuidade as funcionalidades, a tela apresentada acima resume o principal funcionamento do sistema, o cadastro de ordens de serviço, onde é possível cadastrar utilizando clientes já inclusos no banco de dados e fornecer informações importantes para a regra de negócio proposta, estando uma ordem de serviço cadastrada é possível gerencia-la através das situações e relatórios impressos.

5. Resultados Obtidos

Com a aplicação da metodologia *Scrum* além do resultado físico, que é fruto do trabalho coletivo em si gerado mediante as interações entre os integrantes da equipe foi possível identificar a transparência no trabalho onde cada indivíduo manifestava seu posicionamento diário em relação ao cumprimento do que era esperado, tal ritual gerou um sentimento de pertença, o ritual infundiu a ideia de que o projeto é de todos.

Automaticamente o método gerou no grupo uma rotina de inspeção, visando oportunidades e riscos inerentes a atividade de cada integrante para que o objetivo macro fosse alcançado.

Por fim foi perceptível que a aplicabilidade da metodologia exigiu adaptação por parte dos integrantes da equipe, tal percepção se deu mediante ao posicionamento que o *Scrum* define para a sua pratica, onde não existe competição entre os membros da equipe pois o objetivo final depende de todos.

Tais fatores culminaram na agilidade no cumprimento das tarefas pois todo time envolvido sempre possuía total ciência do que estava ocorrendo, quando estava ocorrendo e como estava ocorrendo.

Pode-se concluir que as metodologias ágeis além de fornecer uma nova roupagem ao gerenciamento de projetos disseminam uma nova visão quanto à maneira de se produzir algo tendo como foco o principal agente o executar das tarefas.

Referências

BLACKDUCK. Compare languages. Disponível em:

<<https://www.openhub.net/languages/compare>>/. Acesso em 12 jun 2017.

CHIAVENATO, IDALBERTO. **Introdução à Teoria Geral da Administração**.

Barueri: Manole, 2014

KENT, BECK. **Manifesto para o Desenvolvimento Ágil de Software**. Disponível em:

<<https://www.manifestoagil.com.br/>>. Acesso em 12 jun 2017.

PRESSMAN, ROGER S. Engenharia de Software: Uma abordagem profissional. Porto Alegre: Bookman, 2011.

SABBAGH, RAFAEL. **Scrum Gestão Ágil para Projetos de Sucesso**. São Paulo: Casa do Código, 2015.

SOMMERVILLE, IAN. **Engenharia de Software**. São Paulo: Pearson, 2007.