



UNIVERSIDADE TECNOLÓGICA FEDERAL DO PARANÁ
PROGRAMA DE PÓS-GRADUAÇÃO LATO SENSU
ESPECIALIZAÇÃO EM DESENVOLVIMENTO WEB

MARCOS APARECIDO HUSS

**AVALIAÇÃO DE UM SISTEMA ONLINE DE CONTROLE DE TAREFAS
SOB A ÓTICA DE MÉTODOS ÁGEIS E GERENCIAMENTO DE PROJETOS**

MONOGRAFIA DE ESPECIALIZAÇÃO

LONDRINA – PR
2014

MARCOS APARECIDO HUSS

**AVALIAÇÃO DE UM SISTEMA ONLINE DE CONTROLE DE TAREFAS
SOB A ÓTICA DE MÉTODOS ÁGEIS E GERENCIAMENTO DE PROJETOS**

Monografia de especialização apresentada no
Câmpus Londrina da Universidade Tecnológica
Federal do Paraná como requisito parcial para
obtenção do título de “Especialista em
Desenvolvimento Web”. Orientador: Prof. Dr.
André Domingues.

LONDRINA – PR
2014



TERMO DE APROVAÇÃO

Título da Monografia

AVALIAÇÃO DE UM SISTEMA ONLINE DE CONTROLE DE TAREFAS SOB A ÓTICA DE MÉTODOS ÁGEIS E GERENCIAMENTO DE PROJETOS

por

MARCOS APARECIDO HUSS

Esta monografia foi apresentada às 15h00 do dia **06** de **DEZEMBRO** de **2013** como requisito parcial para a obtenção do título de ESPECIALISTA EM DESENVOLVIMENTO WEB. O candidato foi arguido pela Banca Examinadora composta pelos professores abaixo assinados. Após deliberação, a Banca Examinadora considerou o trabalho

_____.

(aprovado, aprovado com restrições ou reprovado)

Prof. Dr. André dos Santos Domingues
(UTFPR)

Prof. Dr. Elias Canhadas Gengivir
(UTFPR)

Prof. Me. Thiago Prado de Campos
(UTFPR)

Visto da coordenação:

Prof. Me. Thiago Prado de Campos
Coordenador da esp. em Desenvolvimento Web

Prof. Dr. Walmir Eno Pottker
Coordenador de Pós-Graduação Lato Senso

Dedico este trabalho a todos que ajudaram, não somente neste período, mas sim em todo meu caminho na vida profissional, sendo esta mais uma conquista a ser comemorada por todos.

AGRADECIMENTOS

Agradeço ao orientador Prof. Dr. André Domingues, a todos os colaboradores da ID Agência Digital, que se envolveram de qualquer forma ou mesmo com qualquer atitude que me inspirou a realizar este trabalho. Não posso deixar de agradecer também aqueles que me acompanharam durante todo meu crescimento profissional e pessoal até aqui, com palavras de incentivo e apoio. Meu muito obrigado a todos.

“Reunir-se é um começo, permanecer juntos é um progresso, e trabalhar juntos é um sucesso.”

Henry Ford

RESUMO

HUSS, Marcos Aparecido. Avaliação de um sistema online de controle de tarefas sob a ótica de Métodos Ágeis e Gerenciamento de Projetos. 2013. 52 f. Monografia (Especialização em Desenvolvimento WEB) - Programa de Pós-Graduação em Tecnologia, Universidade Tecnológica Federal do Paraná. Londrina, 2013.

Esta monografia contém uma avaliação do sistema de gerenciamento de tarefas ID Controle, desenvolvido no ano de 2010 pela empresa ID Agência Digital após uma consultoria de planejamento estratégico. O sistema, quando concebido, não obteve suporte de nenhuma metodologia de desenvolvimento ou técnica de gerenciamento, sendo então desenvolvido a partir do conhecimento tácito dos colaboradores presentes na empresa. Este estudo objetivou a análise do sistema ID Controle sob o aporte teórico das Metodologias Ágeis e do Gerenciamento de Projetos e diagnosticou a possibilidade de várias iniciativas coerentes para uma melhoria significativa de suas funcionalidades, proporcionando assim, um panorama do alcance qualitativo da ferramenta.

Palavras-chave: Metodologia, ágil, gerenciamento, projeto e ferramenta.

ABSTRACT

HUSS, Marcos Aparecido. Evaluation of an Online System Control Tasks in the Perspective of Agile Methods and Project Management. 2013. 52 f. Monograph (Web Development Specialization) - Graduate Program in Technology, Universidade Tecnológica Federal do Paraná. Londrina, 2013.

This monograph contains a evaluation of an Online System control Tasks in the Perspective of agile methods and Project Management, ID Control, developed in 2010 by the company Digital ID Agency after consulting strategic planning. The system, as designed, did not find support for any development methodology or technical management, and then developed from the tacit knowledge of employees in the firm. This study aimed to analyze the system ID in the theoretical control of Agile Methodologies and Project Management and diagnosed the possibility of several initiatives consistent for a significant improvement of their functionality, thus providing an overview of the scope of qualitative tool.

Keywords: Methodology, agile, management, and design tool.

LISTA DE FIGURAS

Figura 1 - Nível de custo e pessoal durante o projeto.	18
Figura 2 - Seqüência típica de fases no ciclo de vida de um projeto.	19
Figura 3 - A relação entre as partes interessadas e o projeto.	20
Figura 5 - Exemplo de painel de acompanhamento de fluxo com Kanban.	36
Figura 6 – Interface Principal do OpenProj – Itens da tarefa e gráfico de Grantt.....	38
Figura 7 – Tela de exibição de Diagrama de Rede.....	39
Figura 8 - Exemplo da interface do Trello.....	40
Figura 9 - Exemplo da interface do Marqueeed.	41
Figura 10 - Exemplo da interface do Asana.....	42
Figura 11 - Exemplo da interface do Basecamp.....	43
Figura 12 - Tela de visualização de tarefas pendentes: ID Controle.	45
Figura 13 - Tela de formulário de dúvidas do projeto: ID Controle.	46
Figura 14 - Tela de cadastro de projeto: ID Controle.....	48
Figura 15 - Tela de detalhamento da tarefa: ID Controle.....	48
Figura 16 - “Tráfego” das tarefa e colaboradores do projeto: ID Controle.....	50

SUMÁRIO

1. Introdução	12
1.1 Problema	12
1.2 Objetivo	13
1.3 Organização do Trabalho	13
2. Referencial Teórico	15
2.1 Gerenciamento de projetos	15
2.1.1 Conceitos do Gerenciamento de Projeto	15
2.1.2 Características de um projeto	15
2.1.3 Ciclo de vida do projeto	17
2.1.4 Partes interessadas no projeto	19
2.1.5 Causas de fracassos em projetos	22
2.1.6 Benefícios do Gerenciamento de Projetos	24
2.2 Metodologias tradicionais	25
2.3 Metodologias ágeis	25
2.4 Principais metodologias ágeis utilizados	31
2.4.1 Scrum	32
2.4.2 Dynamic System Development Method (DSDM)	32
2.4.3 Extreme Programming (XP)	32
2.4.4 Feature Drive Development (FDD)	33
2.4.5 Cristal Clear	33
2.4.6 TDD	34
2.4.7 Importância de uma ferramenta de apoio	34
3. ferramentas de apoio	36
3.1 Kanban	36
3.2 OpenProj	37
3.3 Trello	39
3.4 Marqeed	40
3.5 Asana	41
3.6 Basecamp	42
4. Avaliação do Sistema online ID Controle	44
4.1 Breve histórico do desenvolvimento do ID Controle	44
4.2 Avaliação e métodos de apuração	45
4.3 Melhorias	50
5. Conclusão	52
6. Bibliografia	53

1. INTRODUÇÃO

Em novembro de 2009 a empresa de desenvolvimento de web sites ID Agência Digital passou por uma técnica de análise de cenário SWOT, ou PFOA (Potencialidades, Fraquezas, Oportunidades e Ameaças), com intuito de executar um planejamento estratégico de negócios.

O resultado da análise SWOT mostrou que a empresa apresentava grande dificuldade e deficiência em coordenar projetos comercializados, culminando em atrasos de cronogramas frequentes, falta de comunicação entre os membros das equipes de desenvolvimento e informações desencontradas, incompletas e muitas vezes desnecessárias.

Nesse cenário, os colaboradores presentes na empresa nesse período desenvolveram um sistema online denominado ID Controle, capaz de registrar e organizar as tarefas individualmente de cada membro da equipe, porém, sem nenhum conhecimento ou aporte teórico sobre metodologias ou gerenciamento de projeto, e seu desenvolvimento baseou-se apenas nas experiências e sugestões dos colaboradores da empresa.

O projeto propõe então uma avaliação proveniente da análise da ferramenta ID Controle com base na literatura pertinente ao tema, levantando conceitos, métodos e técnicas que possam ser aplicadas a fim de melhorar o desempenho da mesma, visando melhor interação interna da empresa e também de seus clientes.

1.1 Problema

A qualidade em gerenciar projetos com qualidade é de suma importância no ambiente empresarial de modo geral, um projeto com problemas de gerenciamento significa prejuízos e descontentamento para todas as partes envolvidas, ou seja, empresa e clientes.

Diante do cenário da empresa de já utilizar uma ferramenta de apoio no gerenciamento de projetos, o ID Controle, é preciso analisar se ela pode ser inserida ou adaptada em metodologias mais eficazes de gerenciamento de projetos e se suas funcionalidades atuais condizem com as expectativas de melhora em seu desempenho.

1.2 Objetivo

Realizar uma avaliação da Ferramenta ID Controle com base nas literaturas disponíveis sobre os temas Metodologias Ágeis e Gerenciamento de Projetos a fim de apontar quais funcionalidades e características disponíveis na ferramenta se enquadram nas fundamentações teóricas levantadas.

Assim, diante do embasamento teórico, diagnosticar a possibilidade de adaptações, incrementos e ações coerentes para uma melhoria significativa de suas funcionalidades, e acima de tudo, proporcionar um amplo panorama do alcance qualitativo da ferramenta.

1.3 Organização do Trabalho

Neste capítulo foi apresentado uma introdução descritiva com intuito de criar visão geral do trabalho, relatando suas principais informações para a compreensão da monografia, seguido da problemática previamente identificada, o objetivo e as contribuições esperadas na conclusão do trabalho.

No Capítulo 2 é levantado todo aporte teórico para sustentar o desenvolvimento do trabalho, baseado nas referências bibliográficas pertinentes ao tema proposto. Na sequência, no Capítulo 3, Levantamento de Ferramentas de Apoio, foram selecionadas algumas ferramentas similares à analisada, levando em consideração suas funcionalidades e características.

Seguindo a seqüência no capítulo 4 Avaliação do Sistema online ID Controle, analisamos a ferramenta sob a ótica das Metodologias Ágeis e Gerenciamento de Projetos citando algumas considerações pertinentes ao sistema.

Finalmente são apresentadas as considerações finais desse trabalho no capítulo 5 Conclusão, onde são explanados os resultados da análise, assim como propostas de melhorias da ferramenta que foi objeto do estudo. Seguido das Referências Bibliográficas no Capítulo 6 encerrando esta monografia.

2. REFERENCIAL TEÓRICO

2.1 Gerenciamento de projetos

Nesta seção foi executado o aporte teórico sobre gerenciamento de projetos afim de embasar com a literatura pertinente ao tema a avaliação da ferramenta que é o objeto de estudo deste trabalho.

2.1.1 Conceitos do Gerenciamento de Projeto

O PMBOK 4ª Edição (2008), define projeto como “um empreendimento temporário cujo objetivo é criar um produto, serviço ou resultado distinto e único”.

Quando intitulamos um projeto como temporário, isso quer dizer que ele deve ter início, meio fim bem definidos, já o fato de ser único, determina que este mesmo projeto tenha suas próprias características, diferente de outros projetos que já tenham sido concebidos.

Projeto é um empreendimento não repetitivo, caracterizado por uma seqüência clara e lógica de eventos, com início, meio e fim, que se destina a atingir um objetivo claro e definido, sendo conduzido por pessoas dentro de parâmetros predefinidos de tempo, custo, recursos envolvidos e qualidade. (Vargas, 2005, p.7).

Assim, sabemos que para existir um projeto, é necessário um saber quais os desejos do cliente, suas metas e o resultado esperado com o trabalho. Definindo então quais os objetivos deve-se passar ao planejamento, a fim de estimar o tempo de cada atividade envolvida. Além do tempo, é necessário também determinar as ferramentas, orçamento disponível, profissionais envolvidos, equipes de trabalho, sendo todos esses intitulados como recursos necessários para a conclusão do projeto.

2.1.2 Características de um projeto

O Guia de Conhecimento em Gerenciamento de Projeto, o Pmbok 4ª edição (2008), traz como principais características de um projeto:

- **Temporário:** Aqui como foi abordado anteriormente um projeto deve início e fim bem definidos, estimando de acordo com cada projeto o tempo de execução, isto que em sua maioria, os objetivos são para que resulte em produtos ou serviços que serão utilizados durante muito tempo, um software, por exemplo, pode ser utilizado durante anos, se foi bem projetado e atende as necessidades do cliente de maneira satisfatória;
- **Empreendimento não repetitivo:** Essa característica pressupõe que o projeto deve ser algo novo, exclusivo, único, pois cada novo projeto apresenta necessidades inerentes a um determinado cliente e suas necessidades, e que os objetivos dele resultam em algo novo;
- **Sequência clara e lógica de eventos:** Ordenar todas as atividades em uma ordem lógica, que torne o projeto de fácil acompanhamento é imprescindível para que os objetivos do mesmo sejam alcançados;
- **Início, meio e fim:** Um projeto deve respeitar um ciclo de vida bem definido com uma ordem bem clara das etapas, onde o planejamento é o início, a execução como meio do processo e a encerramento do cronograma, ou entrega como etapa final do projeto;
- **Objetivo claro e definido:** Definir objetivos, metas a serem alcançadas é essencial para se executar um projeto. Saber claramente quais os objetivos a serem alcançados determina aonde se quer chegar e o porquê das atividades a serem executadas;
- **Conduzido por pessoas:** Ferramentas de gestão são necessárias para a execução de um projeto, porém pessoas são extremamente

necessárias para a execução, alimentação da ferramenta, com as informações necessárias para sua execução;

- **Projetos utilizam recursos:** Recursos pessoais, financeiros, prazos são necessários para determinar como cada atividade será desenvolvida;
- **Parâmetros predefinidos:** Estabelecer prazos, valores, custos, materiais, pessoas, e equipamentos envolvidos e a qualidade exigida em um planejamento com total precisão é praticamente impossível. No decorrer do projeto é que vamos identificar e quantificar esses como será usado esses recursos. Porém, determinar estimativas no início do projeto como referências do projeto, servirá posteriormente para sua avaliação;
- **Individualidade:** Este conceito diz que um projeto deve atender a necessidades únicas, e que cada novo projeto apresentará algo novo, atendendo a necessidades inerentes àquele caso em questão, com soluções inovadoras e exclusivas;
- **Elaboração progressiva:** Aqui determinamos que o desenvolvimento do projeto deverá apresentar várias etapas, quanto mais etapas, mais específico e detalhado o desenvolvimento.

2.1.3 Ciclo de vida do projeto.

Um projeto deve ter fases de desenvolvimentos para serem gerenciáveis, essas devem ter ligação entre si e uma sequência lógica. Cada fase é interligada a outra pela entrega ou finalização, ou seja, por finalizações de cada fase, terminada a primeira, iniciamos a segunda.

Esse procedimento torna-se o Ciclo de Vida do projeto, o conjunto dessas fases, que podem se desmembrar em subfases de acordo com a complexidade do projeto.

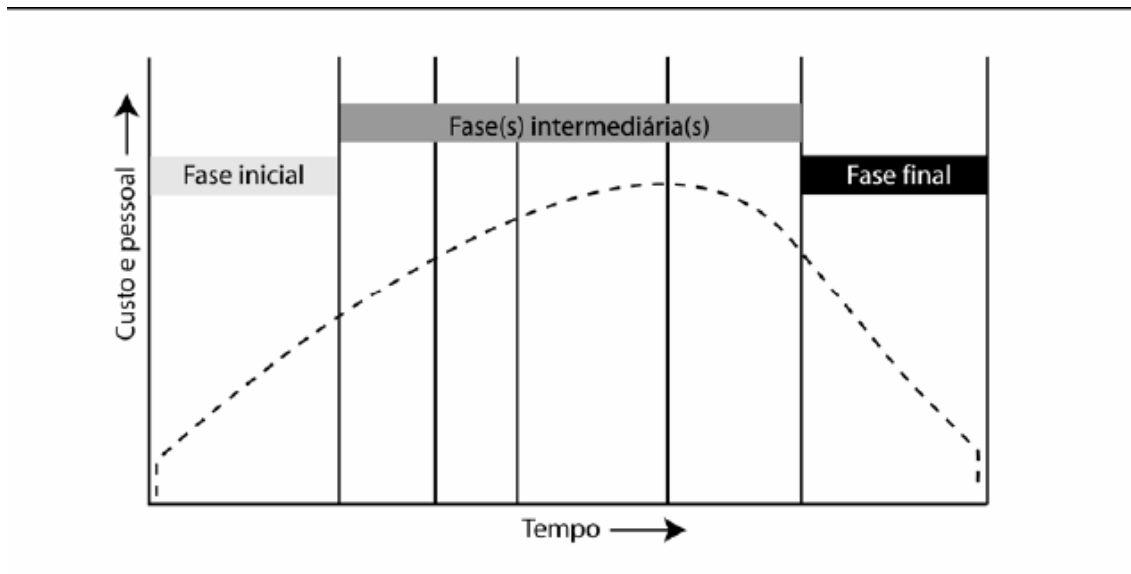


Figura 1- Nível de custo e pessoal durante o projeto.

Fonte: Pmbok 3ª edição (2004).

É possível observar na Figura 1 o custo de execução do projeto durante seu ciclo de vida. Fica claro que na fase inicial os gastos são menores, é onde colhemos informações e as necessidades do cliente, quando migramos para a fase intermediária, a de execução do projeto, onde percebemos o aumento dos custos e a necessidade de aumentar o pessoal envolvido, seguindo desta maneira até a fase final, onde começam a diminuir a quantidade de pessoas trabalhando, tendo em vista que entramos na fase final do projeto, seu término.

De acordo com o Pmbok 3ª edição (2004), os projetos variam de tamanho e complexidade. Não importa se são grandes ou pequenos, simples ou complexos, todos devem seguir o ciclo de vida a seguir:

- Início do projeto;
- Organização e preparação;
- Execução do trabalho do projeto e
- Encerramento do projeto.

Porém, segundo o Pmbok 3ª edição (2004), mesmo depois de uma fase terminada isso não conclui que o início da próxima está autorizado. Para que se obtenha um controle mais eficaz, cada fase é formalmente iniciada

para produzir uma saída dependente da fase do Grupo de Processos de iniciação, especificando o que é esperado e permitido para a mesma. Conforme a Figura 2 mostra a seguir, podemos ter metas inseridas em subfases, subfases podem obter duas autorizações para diferentes tarefas em fases subsequentes, todas culminando no que são chamadas de saídas de fase, passagens de fase, ou pontos de término.

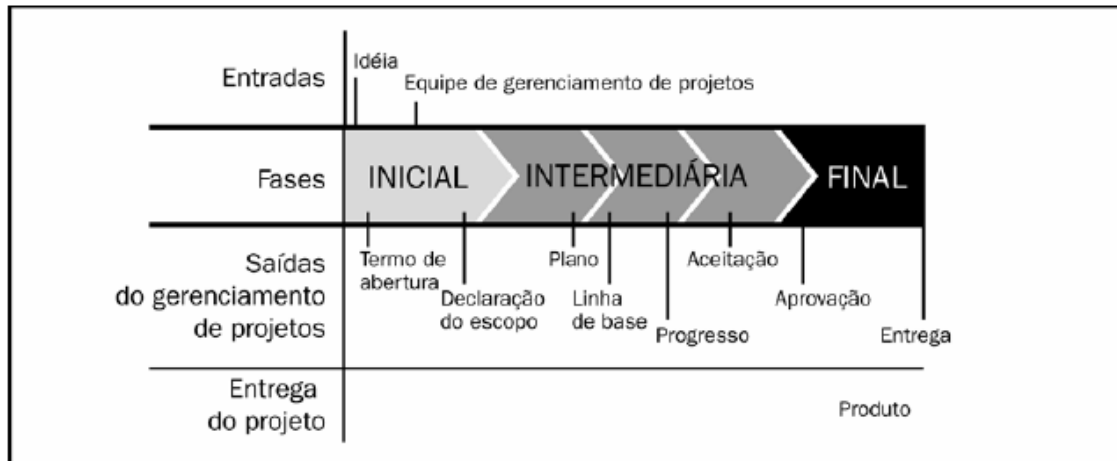


Figura 2 - Seqüência típica de fases no ciclo de vida de um projeto.

Fonte: Pmbok 3ª edição (2004).

2.1.4 Partes interessadas no projeto

Partes interessadas no projeto são pessoas e organizações ativamente envolvidas no projeto ou cujos interesses podem ser afetados como resultado da execução ou do término do projeto. Eles podem também exercer influência sobre os objetivos e resultados do projeto. A equipe de gerenciamento de projetos precisa identificar as partes interessadas, determinar suas necessidades e expectativas e, na medida do possível, gerenciar sua influência em relação aos requisitos para garantir um projeto bem-sucedido. (Pmbok 3ª edição, 2004, p.24). Na Figura 3 podemos identificar a relação entre os envolvidos no projeto, veja abaixo:

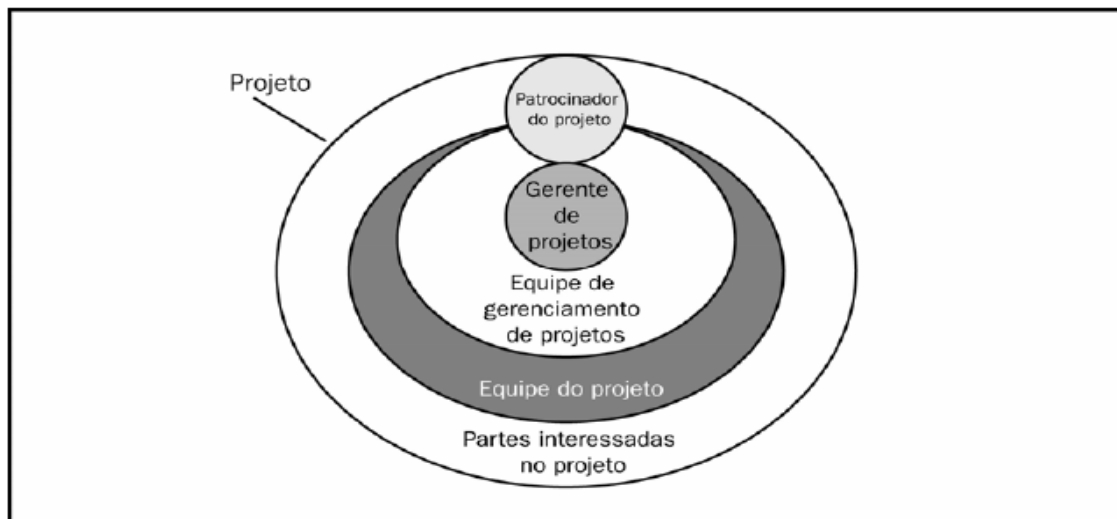


Figura 3 - A relação entre as partes interessadas e o projeto.

Fonte: Pmbok 3ª edição (2004).

Muitas vezes as partes interessadas no projeto podem ser difíceis de identificar, isso pode acarretar muitos problemas posteriormente, um exemplo são projetos onde é preciso a participação de um departamento jurídico, que pode acarretar em novas regras e documentações não previstas a fase de planejamento do projeto.

Outro ponto importante é que o envolvimento de todos os interessados em um determinado projeto pode significar influências positivas e negativas. As partes positivas desfrutam dos resultados bem-sucedidos, enquanto as partes interessadas negativas enxergam somente resultados negativos a partir do sucesso do projeto. Geralmente essas partes negativas são frequentemente negligenciadas pela equipe do projeto, que corre o risco de não conseguir avançar seus projetos a finais bem sucedidos. Já no caso das partes interessadas positivas, acabam sendo atendidas da melhor forma possível quando ajudam o projeto, obtendo, por exemplo, as permissões necessárias para o andamento do projeto.

De acordo com Pmbok 3ª edição (2004), as principais partes interessadas em projetos incluem:

- **Gerente de projetos:** A pessoa responsável pelo gerenciamento do projeto.
- **Cliente/usuário:** A pessoa ou organização que utilizará o produto do projeto. Podem existir várias camadas de clientes. Por exemplo, os clientes de um novo produto farmacêutico podem incluir os médicos que o receitam, os pacientes que o utilizam e as empresas de saúde que pagam por ele. Em algumas áreas de aplicação, os termos cliente e usuário são sinônimos, enquanto em outras, cliente se refere à entidade que adquire o produto do projeto e usuários são os que utilizarão diretamente o produto do projeto.
- **Organização executora:** A empresa cujos funcionários estão mais diretamente envolvidos na execução do trabalho do projeto.
- **Membros da equipe do projeto:** O grupo que está executando o trabalho do projeto.
- **Equipe de gerenciamento de projetos:** Os membros da equipe do projeto que estão diretamente envolvidos nas atividades de gerenciamento de projetos.
- **Patrocinador:** A pessoa ou o grupo que fornece os recursos financeiros, em dinheiro ou em espécie, para o projeto.
- **Influenciadores:** Pessoas ou grupos que não estão diretamente relacionados à aquisição ou ao uso do produto do projeto mas que, devido à posição de uma pessoa na organização do cliente ou na organização executora, podem influenciar, positiva ou negativamente, no andamento do projeto.

- **PMO:** Se existir na organização executora, o PMO poderá ser uma parte interessada se tiver responsabilidade direta ou indireta pelo resultado do projeto.

2.1.5 Causas de fracassos em projetos

Segundo Vargas (2005), mesmo a quantidade de benefícios conquistados pelos projetos, grande parte deles não é concluído ou não apresenta o resultado esperado. Essas falhas são obstáculos naturais ou externos que estão fora do controle da organização, e que muitas vezes podem apenas ser minimizados ou evitados através de um gerenciamento de riscos adequado. As falas mais comuns são:

- Mudança na estrutura organizacional da empresa;
- Riscos elevados no meio ambiente;
- Mudanças na tecnologia disponível;
- Evolução nos preços e prazos;
- Cenário político-econômico desfavorável.

Porém as denominadas falhas gerenciais são consideradas causas de grande parte dos insucessos, e podem ser em sua maioria evitadas, são elas:

- As metas e os objetivos são mal-estabelecidos, ou não são compreendidos pelos escalões inferiores;
- Há pouca compreensão da complexidade do projeto;
- O projeto inclui muitas atividades e muito pouco tempo para realizá-las;
- As estimativas financeiras são pobres e incompletas;
- O projeto é baseado em dados insuficientes, ou inadequados;
- O sistema de controle é inadequado;

- O projeto não teve um gerente de projeto, ou teve vários, criando círculos de poder paralelos aos previamente estabelecidos;
- Criou-se muita dependência no uso de softwares de gestão de projetos;
- O projeto foi estimado com base na experiência empírica, ou *feeling* dos envolvidos, deixando em segundo plano os dados históricos de projetos similares, ou até mesmo análises estatísticas efetuadas;
- O treinamento e a capacitação foram inadequados;
- Faltou liderança do gerente de projeto;
- Não foi destinado tempo para as estimativas e o planejamento;
- Não se conheciam as necessidades de pessoal, equipamentos e materiais;
- Fracassou a integração dos elementos-chave do escopo do projeto;
- Cliente/projeto tinham expectativas distintas e, muitas vezes, opostas;
- Não se conheciam os pontos-chave do projeto;
- Ninguém verificou se as pessoas envolvidas nas atividades tinham conhecimento necessário para executá-las;
- As pessoas não estavam trabalhando nos mesmos padrões, ou os padrões de trabalho não foram estabelecidos.

Muitas das falhas citadas acima são rotineiras e passam despercebidas na maioria dos projetos que acabam em fracasso, isso por que muitas vezes é difícil distinguir o fracasso, que pode ser parcial, pode ser reconhecido como sucesso em outro ponto de vista, tornando a avaliação de seus resultados ainda mais difícil. O gerente de projeto tem então a tarefa de controlar junto a sua equipe as possibilidades de insucessos citados acima.

2.1.6 Benefícios do Gerenciamento de Projetos

Gerenciar projetos se mostra muito eficaz em diversos aspectos, como, concluir trabalhos dentro de prazos estimados e trabalhar dentro de orçamentos definidos pela organização. Outra vantagem é que não é preciso restringir o gerenciamento de projetos somente aos grandes, mais complexos e de alto custo, ele pode ser aplicado em qualquer projeto, de qualquer tamanho.

Segundo Vargas (2005), os benefícios em destaque são:

- Evita surpresas durante a execução dos trabalhos;
- Permite desenvolver diferenciais competitivos e novas técnicas, uma vez que toda a metodologia está sendo estruturada;
- Antecipa as situações desfavoráveis que poderão ser encontradas, para que ações preventivas e corretivas possam ser tomadas antes que essas situações se consolidem como problemas;
- Adapta os trabalhos ao mercado consumidor e ao cliente;
- Disponibiliza os orçamentos antes do início dos gastos;
- Agiliza as decisões, já que as informações estão estruturadas e disponibilizadas;
- Aumenta o controle gerencial de todas as fases a serem implementadas devido ao detalhamento ter sido realizado;
- Facilita e orienta as revisões da estrutura do projeto que forem decorrentes de modificações no mercado ou no ambiente competitivo, melhorando a capacidade de adaptação do projeto;
- Otimiza a alocação de pessoas, equipamentos e materiais necessários;
- Documenta e facilita as estimativas para futuros projetos.

2.2 Metodologias tradicionais

Segundo Martins (2007) as metodologias de desenvolvimento de softwares tradicionais, ou também chamadas de abordagens clássicas, tem como característica a definição de seus processos, onde todo seu conceito, assim como, o escopo de entregas é definido no início do projeto. Existe então a necessidade de uma documentação muito bem detalhada e volumosa.

Tais métodos tradicionais são mais indicados em situações onde os requisitos são bem conhecidos, com raras alterações e adaptações. Devido também ao alto conhecimento dos processos a serem executados, estes dificilmente variam, baseados nos requisitos e no que se espera do resultado final.

De acordo com Caetano (2010), dependendo do projeto as metodologias tradicionais podem deixar os desenvolvedores amarrados a requisitos desatualizados, que muitas vezes não correspondem às reais necessidades dos clientes. Diante disso, empresas de desenvolvimento de software tendem a buscar outros métodos que estão mais aptos a suprir mudanças e necessidades do cliente. A união dos fatores agilidade, adaptabilidade e qualidade do produto final resultam na satisfação do cliente.

2.3 Metodologias ágeis

Atualmente a competitividade dos mercados gera nas empresas necessidade de se destacar perante seus concorrentes. Métodos mais ágeis e abertos a novas interações e/ou mudanças são de interesse não só da área de Desenvolvimento de Softwares, mas de qualquer outro segmento (MARTINS, 2007). Diante desse contexto, a qualidade no gerenciamento de projetos de desenvolvimento de softwares torna-se um diferencial de peso dentro das empresas frente à concorrência.

Há algumas décadas já existiam métodos e processos tradicionais de desenvolvimento que hoje passaram a ser considerados pesados por conta de sua quantidade excessiva de documentação. Documentação muito extensa implica em gasto de tempo para executá-la e, isso se torna inviável, principalmente quando tratamos de projetos que envolvem equipes reduzidas.

Segundo Koscianski e Soares (2007) pode-se traçar uma comparação entre as metodologias clássicas e ágeis por meio da Ilustração 1, onde se analisa a relação entre o custo de mudança no software e a etapa em que se encontra. A linha pontilhada representa uma estimativa de custo relativo à metodologia clássica, já a linha contínua exibe o que se espera de melhorias na aplicação de metodologias ágeis.

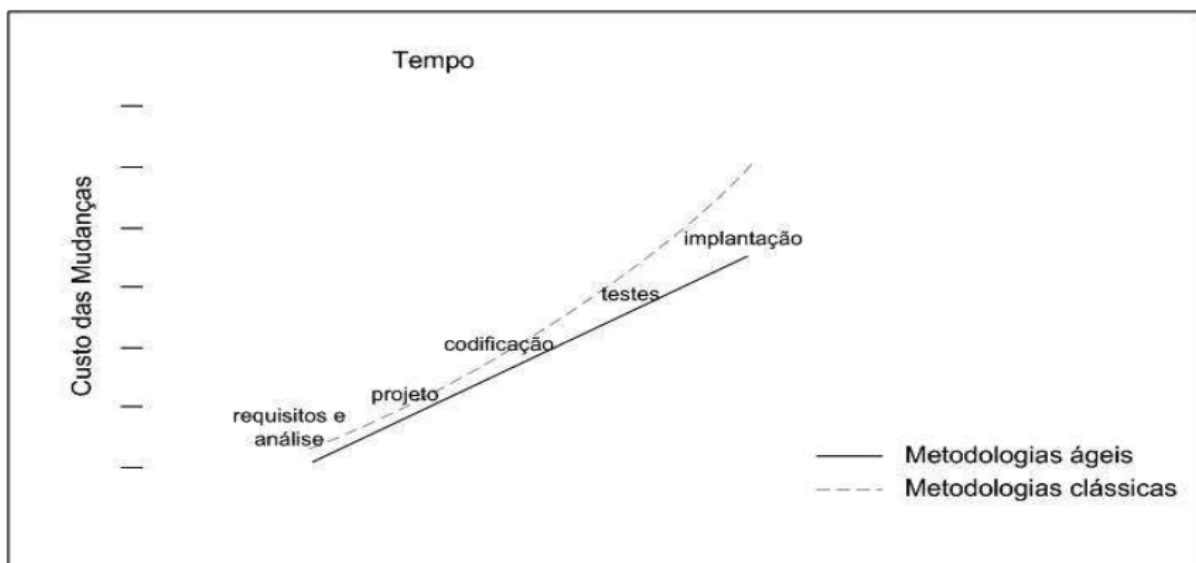


Figura 4- Comparação do custo de mudanças entre metodologias clássicas e ágeis.

Fonte: Koscianski; Soares (2007).

A documentação é extremamente importante para a qualidade de um projeto, porém, pode ser reduzida, não tão detalhada, economizando tempo, que é um grande fator em termos de diferencial competitivo e dessa maneira os projetos são entregues em menor prazo e com qualidade (KOSCIANSKI; SOARES, 2007).

Segundo Soares (2010), as metodologias ágeis têm se tornada alternativa às abordagens tradicionais para o desenvolvimento de software, entre suas principais características estão: aceitação de mudanças e requisitos durante o processo de desenvolvimento, onde essas situações não apresentem altos custos; as equipes são reduzidas, permitindo uma troca de informações entre os envolvidos de maneira mais eficiente; curtos prazos de entrega das etapas do software, permitindo que o cliente acompanhe essas etapas frequentemente.

As metodologias ágeis são adequadas para situações em que a mudança de requisitos é frequente, de tal maneira que a metodologia deve estar preparada a aceitar as mudanças em vez de tentar prever o futuro (KOSCIANSKI; SOARES, 2007).

A percepção que os usuários têm de suas necessidades também evolui à medida que eles conhecem o sistema. É difícil compreender o valor de uma determinada funcionalidade até que ela seja efetivamente usada, principalmente porque não se pode requerer de um usuário comum a mesma capacidade de abstração que um desenvolvedor possui ao olhar um conjunto de requisitos. (OLIVEIRA, 2003, p. 16).

Levando em consideração que os projetos de softwares e web sites não são atividades profissionais simples, são multidisciplinares, é indispensável que a gerência desses projetos aborde de maneira satisfatória todas essas atividades envolvidas, de maneira a conseguir rendimentos e resultados mais eficientes.

Aceitar mudanças ao invés de evitá-las, confiar na experiência e conhecimento tácito da equipe de desenvolvimento para resolver problemas e eventos imprevisíveis, compõe algumas das características primordiais dos métodos ágeis. Resultante do encontro no ano de 2001 de 17 renomados desenvolvedores em Utah, EUA, a então denominada Aliança do Desenvolvimento Ágil criou o Manifesto Ágil, determinando seus princípios e propósitos, discutidos e desenvolvidos pelo grupo todo.

Seus objetivos são justamente atender as inovações contínuas, adaptabilidade de produtos, entregas com cronograma reduzido, adaptabilidade do processo e das pessoas e por fim resultados confiáveis (MARTINS, 2007).

Conceitos-chave do Manifesto Ágil:

- Indivíduos e interações ao invés de processos e ferramentas: deve-se considerar mais importante as pessoas e a maneira como trabalham e interagem juntas. No processo de desenvolvimento de software há um grupo de pessoas essenciais, como programadores, testadores, gerentes de projeto, designers, modeladores e artistas e caso não haja uma boa convivência e colaboração de entre ambos de nada adianta dispor de boas ferramentas e processos.
- Software operante ao invés de documentações completas: prototipar versões mais simples do projeto onde é possível visualizar o funcionamento de sua arquitetura, conseguir interagir com parte do sistema ao invés de grandes funcionalidades que levariam muito tempo para documentação. É claro que a documentação não deve ser abandonada, mas sim minimizada, deixando-a mais enxuta, usando a ferramenta certa para transmitir a informação respectiva aquele momento.
- Colaboração do cliente ao invés de negociações contratuais: é comum o cliente não ter em mente tudo o que ele espera do resultado final de seu projeto, sua colaboração é essencial. Esse fator tem como consequência novas alterações do projeto, quem geram novos requisitos. O contato direto com o cliente permite entender as responsabilidades da equipe e do mesmo, nesse ponto um contrato pode não ser suficiente, uma comunicação constante com o cliente torna-se essencial para um resultado final satisfatório.

- Responder às mudanças ao invés de seguir um planejamento: à medida que o software vai sendo desenvolvido e ganhando corpo é apresentado ao cliente, assim a percepção do que é realmente necessário pode apresentar mudanças, pois muitas vezes o que é apresentado ao cliente difere de suas expectativas. Responder a essas mudanças, embora existindo um plano de projeto traçado, é essencial que esse plano esteja aberto a essas mudanças que surgirão no decorrer do projeto.

Os quatro valores do Manifesto Ágil deram origem a 12 princípios que sustentam o desenvolvimento ágil (MANIFESTO, 2001):

1. A maior prioridade é satisfazer o cliente através da entrega contínua e rápida de software que possua valor.
2. Mudanças nos requisitos são bem vindas, seja por motivos de alterações de requisitos ou até mesmo por novas necessidades que surgiram no decorrer do desenvolvimento, tais mudanças são aceitas mesmo quando realizadas tardiamente.
3. Frequentemente devem-se entregar versões do software funcionando, de preferência em períodos curtos.
4. Clientes, ou representantes do cliente, devem trabalhar em conjunto com desenvolvedores ao longo de todo o projeto.
5. O projeto deve ser construído com pessoas motivadas. Para tanto, deve ser disponibilizado o ambiente e o suporte necessário, e é imprescindível confiar no potencial da equipe.
6. A conversa “cara-a-cara” é o método mais eficiente para colher informações sobre o projeto com o cliente.

7. Software funcionando é a melhor medida de progresso do projeto e não documentação.
8. Processos ágeis promovem desenvolvimento sustentável. *stakeholders*, clientes e desenvolvedores devem descobrir o ritmo de trabalho e mantê-lo constantemente.
9. Atenção contínua a excelência técnica e a um bom projeto promove a agilidade.
10. Simplicidade é essencial.
11. As melhores arquiteturas, requisitos e projeto provêm de equipes auto organizadas.
12. A equipe deve refletir, em intervalos regulares, como se tornar mais eficiente e, dessa forma, deve ajustar o seu comportamento conforme as necessidades.

Essas mudanças de conceitos e paradigmas sobre processos de desenvolvimento e gerenciamento de softwares são as principais características que diferem as metodologias tradicionais de metodologias ágeis. A abordagem dessas características pelas equipes de desenvolvimento deve permitir a entrega de etapas de maneira contínua, interativa e incremental. Permite também que o cliente solicite e determine o que é prioridade, isso para que o desenvolvedor possa focar-se nessas interações e junto com o cliente realize as validações contínuas do software ou web site. O Manifesto Ágil mostra claramente que processos, ferramentas, documentações extensas, contratos e longos planejamentos têm importância secundária perante os indivíduos, protótipos funcionais etc. Não significa, porém que são dispensáveis, e sim que a colaboração de clientes frente às respostas diante das mudanças é mais precisa. Esses conceitos aproximam-

se melhor da forma como as pequenas e médias empresas trabalham e respondem às mudanças (Koscianski e Soares, 2007).

A Metodologia Ágil pode ser aplicada em qualquer processo de desenvolvimento de software, desde que toda equipe de projeto se adapte as tarefas e processos a fim de aperfeiçoá-las, eliminando barreiras adotando e mantendo os produtos dos trabalhos o mais simples possível.

Assim, é preciso enfatizar a estratégia incremental onde o cliente possa acompanhar o andamento do produto através da visualização, ou do contato com o software funcionando o quanto antes, preferencialmente com datas de entrega reduzidas, isso faz com que o processo torne-se ágil permitindo adaptar modificações rápidas no projeto e nas condições técnicas quando necessário (PRESSMAN, 2006).

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. (Pressman, 2011).

2.4 Principais metodologias ágeis utilizados

Nesta seção serão apresentadas algumas das principais metodologias ágeis existentes, selecionadas por sua presença de mercado, quantidade de profissionais, colaboradores e literatura que reforçam ainda mais os benefícios da aplicação de seus conceitos para o sucesso de seus projetos.

2.4.1 Scrum

Segundo Schwaber (2009), o Scrum vem sendo utilizado para o desenvolvimento de produtos complexos desde o início dos anos 90. O Scrum não é um processo ou uma técnica para o desenvolvimento de produtos, é na verdade um *framework* dentro do qual você pode empregar diversos processos e técnicas. O interessante nessa abordagem é que não se procura um culpado, todos estão em busca da solução pela organização, o analista, o desenvolvedor, o gerente de projetos ou o cliente.

A metodologia Scrum possui uma abordagem flexível em ambientes dinâmicos, onde há sempre adaptações de cronogramas, alterações de escopo, mudança de membros da equipe de desenvolvimento, etc.

Esse método possui alguns princípios semelhantes ao XP, onde as equipes são reduzidas, projetos com requisitos instáveis, muitas vezes desconhecidos, interações curtas, que promove visibilidade para o desenvolvimento (Koscianski e Soares, 2007).

2.4.2 Dynamic System Development Method (DSDM)

Esta metodologia visa desenvolver uma aplicação com a qualidade desejada prevendo um controle de prazos e utilizando protótipos incrementais. Para conseguir, a DSDM foca na interação com o cliente, usuário final e a equipe de desenvolvedores, realizando entregas de protótipos frequentes, com equipe de desenvolvimento autônoma, testes massificados e com definição de prioridades entre a lista de requisitos dada pelo cliente.

Há no DSDM uma recomendação para se utilizar cronogramas a cada intervalo de tempo, e sugere também que em cada incremento de software seja utilizado apenas o necessário para realizar o trabalho, isso facilita o avanço para os demais incrementos do sistema (PRESSMAN, 2006).

2.4.3 Extreme Programming (XP)

De acordo com Teles (2004) o XP é um processo de desenvolvimento de software voltado para:

- Projetos cujos requisitos são vagos e mudam com frequência;
- Desenvolvimento de sistemas orientados a objeto;
- Equipes pequenas, preferencialmente até 12 desenvolvedores;
- Desenvolvimento incremental (ou iterativo), onde o sistema começa a ser implementado logo no início do projeto e vai ganhando novas funcionalidades ao longo do tempo.

Essa metodologia reúne uma série de “Boas Práticas” identificadas pela indústria de desenvolvimento de software, e as aplica constantemente. Revisões de código, testes, integrações rápidas, *feedbacks* com clientes, simplicidade no projeto, dentre outras. A proposta então deste método é utilizar intensamente essas práticas, sendo elas realizadas o tempo todo, o desenvolvimento torna-se mais plástico e adaptável.

2.4.4 Feature Drive Development (FDD)

Está é uma metodologia de desenvolvimento de software que extrai os benefícios provenientes de processos rigorosos, como modelagem, planejamento prévio e controle de projeto, porém contém também características pertinentes as de processos ágeis, como foco na programação, constante *feedback* com o cliente, e entregas frequentes de versões do produto. Não há preocupação no FDD com tecnologias, ferramentas, procedimento de aquisições, dentre outros fatores, esta prevê práticas apenas para o desenvolvimento de software em si.

2.4.5 Cristal Clear

Desenvolvido por Cockburn entre 1992 e 1997 a *Família Cristal*, como também é conhecida, tem como maior diferencial ser um conjunto de metodologias com foco nos talentos e habilidades das pessoas, permitindo que o processo de desenvolvimento seja moldado conforme as características específicas da equipe. Existe neste método um apoio visual de cores que variam entre amarelo, laranja e vermelho, quanto mais forte a cor, maior a complexidade, tamanho e importância do projeto.

Concluindo, a família *Crystal* possui o desenvolvimento incremental com ciclos de no máximo quatro meses, e enfatiza a comunicação e cooperação das pessoas (ABRAHAMSSON et al, 2002).

2.4.6 TDD

Teste Drive Development, ou TDD, é um conjunto de técnicas, ou abordagens onde a evolução do trabalho propõe ao desenvolvedor escrever testes antes de criar o código, ou seja, se inicia pela implementação de um caso de teste, e depois pelo código necessário para que o se passe pelo teste. É uma técnica associada ao *Extreme Programming*(XP), já visto anteriormente, como única forma de se codificar.

De acordo com Baumeister (2011), um desenvolvimento orientado por testes tem como objetivo principal escrever códigos limpos, tendo como referência algum teste que já tenha falhado, resultando sempre em códigos muito bem testados.

2.4.7 Importância de uma ferramenta de apoio

Como foi descrito acima no referencial teórico, existem vários métodos para o gerenciamento de projetos disponíveis, e a importância de ferramentas adequadas para um projeto que satisfaça as expectativas do cliente são essenciais para seu sucesso. Alguns métodos, como o FDD, não levam em consideração quais ferramentas serão utilizadas, porém não existe nenhuma restrição em utilizá-las.

Isso por que existem hoje diversos softwares com a finalidade de apoiar as metodologias escolhidas pela empresa e sua equipe de trabalho, desde os mais complexos e completos softwares, até mesmo aplicações leves, sem a necessidade de instaladores, com acessos online e disponíveis para a maioria dos dispositivos móveis, como *smartphones*, *tablets*, além também do próprio *desktop*.

Isso resulta em facilidade de comunicação entre os membros de uma equipe, a possibilidade de arquivar as ações efetuadas nas tarefas e analisá-las posteriormente, além de gerenciar com mais facilidade entradas e saídas de projetos novos e finalizados.

3. FERRAMENTAS DE APOIO

Nessa subseção são apresentadas as ferramentas de apoio. Todas as selecionadas para a análise são utilizadas como suporte às metodologias ágeis de desenvolvimento e a gerência de projetos, sendo que, cada ferramenta apresenta funcionalidades peculiares, que podem servir como referência para a avaliação do sistema em questão.

3.1 Kanban

Segundo Moura (1994), Kanban é uma técnica japonesa de gestão de materiais e de produção no momento exato (*Just-in-Time*), que é controlado através do movimento de cartões (Kanban). A inspiração inicial para o desenvolvimento do Kanban, segundo seu criador Taiichi Ohno, foi à análise sobre o sistema de funcionamento dos supermercados americanos.

O Sistema Kanban é usualmente utilizado em quadros e cartões visuais que auxiliam o planejamento da produção e o controle de estoques ou fluxos de produção. De acordo com a quantidade de cartões disponíveis nos quadros, são tomadas as decisões de priorização de produção, configuração de máquinas e até mesmo de paradas de linha para manutenção. Na Figura 5 é apresentado um exemplo:

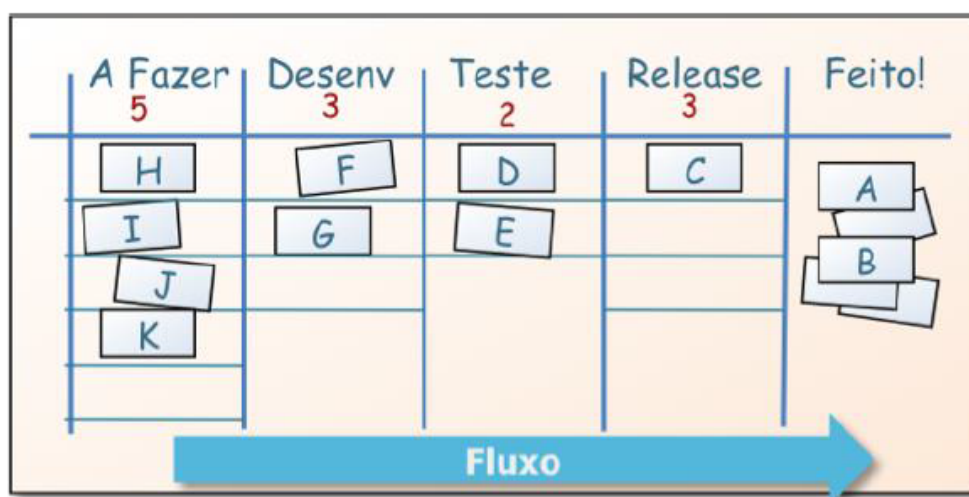


Figura 4 - Exemplo de painel de acompanhamento de fluxo com Kanban.

Fonte: Kanban e Scrum - obtendo o melhor de ambos. 1ª edição (2009).

Segundo Ghisi (2011), devem-se atentar as cinco principais propriedades e características do método Kanban:

- Visualizar o fluxo de trabalho;
- Limitar a quantidade de trabalho em andamento;
- Medir e otimizar o fluxo de trabalho;
- Tornar explícitas as políticas do processo;
- Gerenciar quantitativamente.

O Kanban não é uma metodologia, mas sim um *framework* para implementar mudanças de forma incremental. Esse é um conceito muito importante para que se entenda o Kanban como um todo já que, quando se fala em metodologias, fala-se em conjuntos de práticas e o Kanban não tem nenhuma prática prescrita. Há, nele, somente propriedades que devem guiar a melhoria no processo atual, não importando quais práticas estejam sendo usadas.

Desta maneira o Kanban está presente na área de desenvolvimento de software para auxiliar com seus conceitos e aplicações ainda mais as metodologias ágeis citadas anteriormente. Com ele é possível tornar a equipe mais ágil, dinâmica e flexível, ele deixa bem claro quem são os envolvidos no processo, qual fase do processo e permite a colaboração e supervisão de todos.

3.2 OpenProj

O OpenProj é um software para gerenciamento de projetos de código aberto, ou seja, distribuição gratuita, considerado como uma ótima alternativa a softwares de licença comercial disponíveis no mercado.

Com mais de um milhão de utilizadores é um dos mais conhecidos softwares *open sourced* disponíveis para as plataformas Linux, Mac e Windows e em diversos idiomas, inclusive o português. Apresenta uma interface de fácil utilização que permite ao gerente de projetos elaborar

cronogramas, calcular datas, estabelecer cenários, calcular e visualizar redes PERT, dentre outras funcionalidades.

Segundo Taguchi (2012), certamente ponto positivo deste software é a possibilidade de visualizar uma série de relatórios, tendo em vista que a principal preocupação dos gerentes de projeto é a coordenação das atividades efetuadas durante o projeto, uma maneira extremamente eficaz de acompanhar essas tarefas são as formas de exibição disponíveis, como nos exemplos que podemos visualizar a seguir nas figuras 6 e 7.

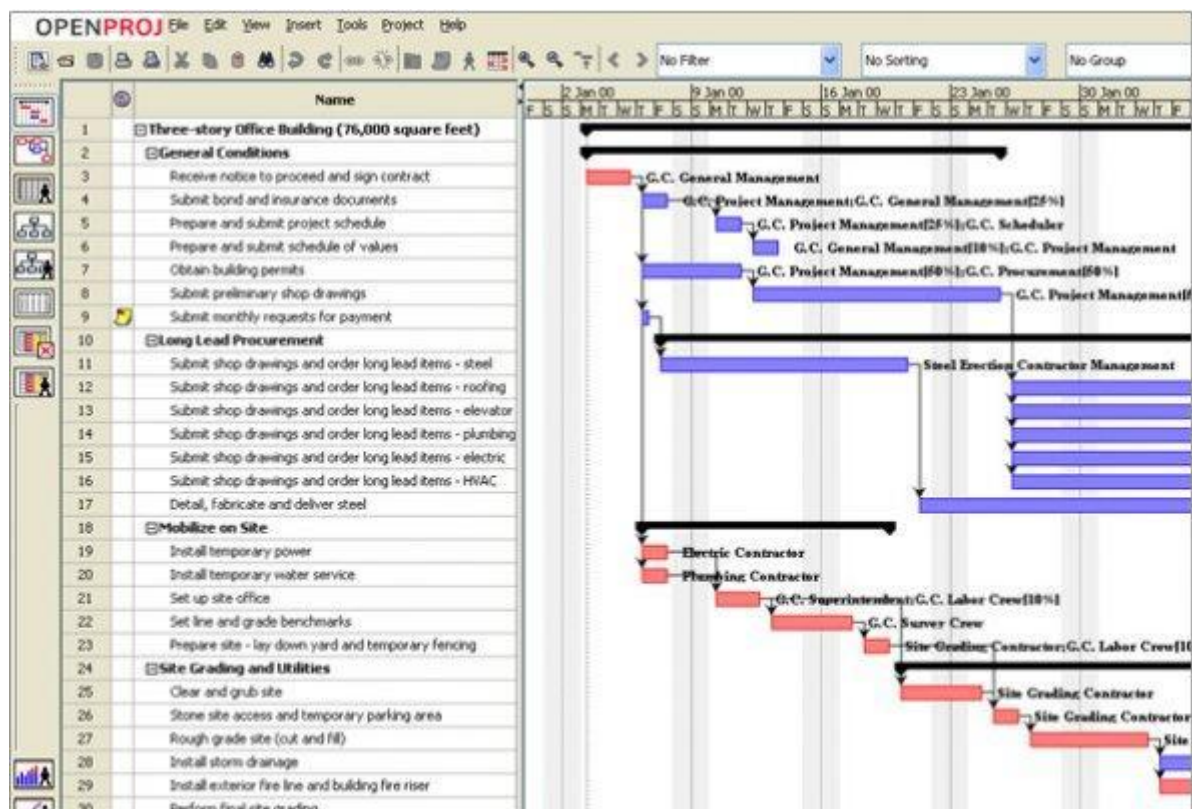


Figura 5 – Interface Principal do OpenProj – Itens da tarefa e gráfico de Grantt.

Fonte: <<http://sourceforge.net/projects/openproj/>> acessado em 12/09/2013.

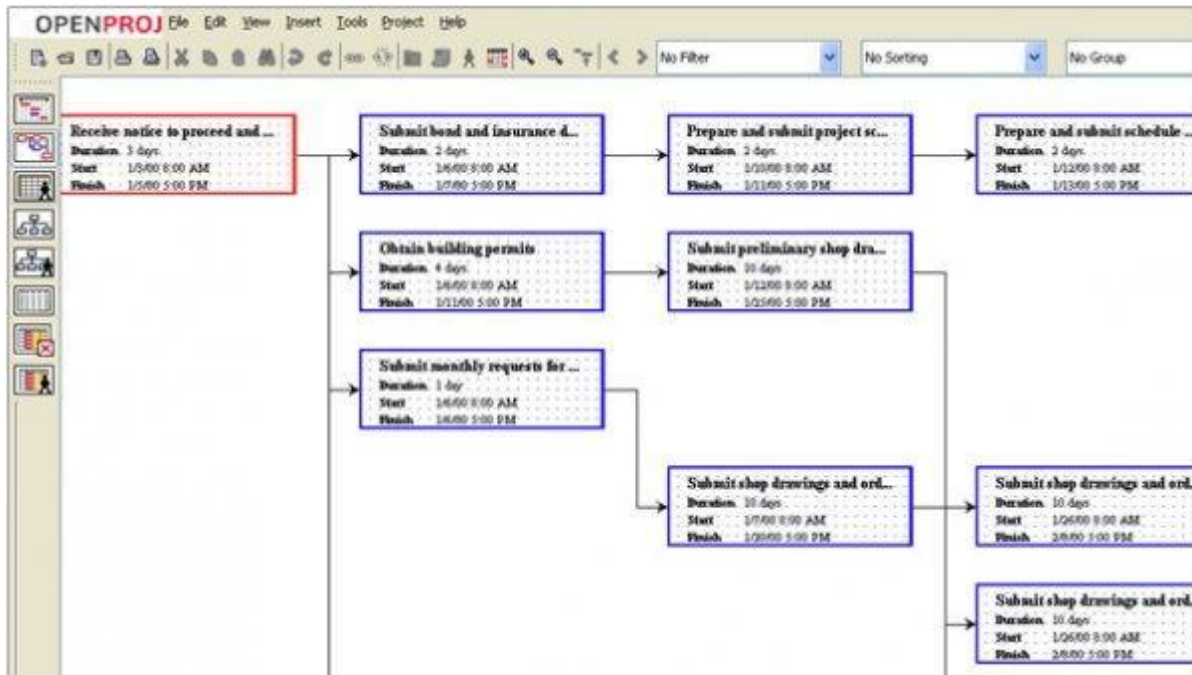


Figura 6 – Tela de exibição de Diagrama de Rede

Fonte: <<http://sourceforge.net/projects/openproj/>> acessado em 12/09/2013.

3.3 Trello.

O Trello é uma aplicação online que permite o gerenciamento de projetos e tarefas. Trabalha com colunas personalizáveis, onde é possível acrescentar novas atividades, projetos, inserir tags para pesquisas posteriores, deletar ou inserir tarefas e pessoas da equipe.

Com uma interface agradável que permite a fácil visualização das tarefas, divide em três colunas baseada no método Kanban, *Fazer*, *Fazendo*, *Feito*, porém é possível criar e editar as colunas da maneira que a equipe desejar. Tem tecnologia de *Drag and Drop*, onde é possível arrastar as janelas com as tarefas entre as colunas, é possível atribuir cores aos integrantes e a suas tarefas. Nele não é possível fragmentar as fases de desenvolvimento em subfases, porém é possível criar mais colunas e inseri-las em seqüência. É muito utilizado atualmente, sendo uma das ferramentas mais conhecidas, disponível em versão gratuita, com aplicativos para sistemas móveis como Android e IOS. A Figura 8 a seguir mostra sua tela principal.

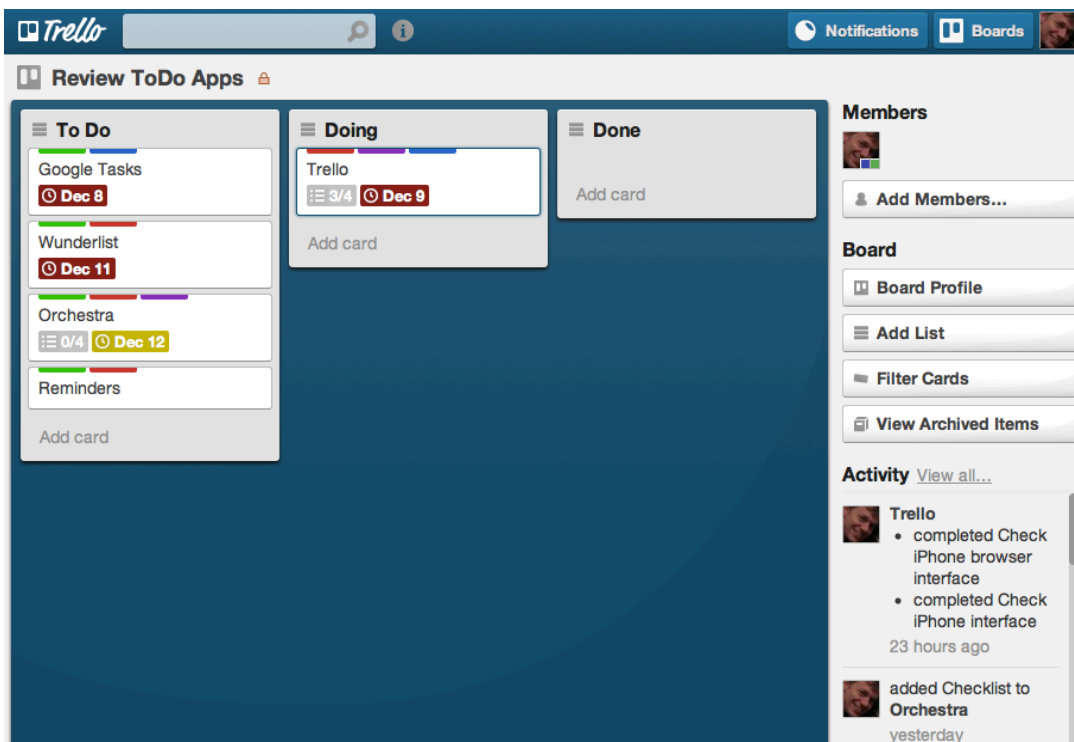


Figura 7 - Exemplo da interface do Trello.

Fonte: <http://www.trello.com> acessado em 05/09/2013.

3.4 Marqued

Essa ferramenta online traz um conceito simples, mas que torna a interação entre a equipe de trabalho de um projeto e o cliente muito ágil e dinâmica. Seu funcionamento consiste em fazer o *upload* das imagens de um possível layout e criar um grupo de trabalho, com colaboradores, clientes, que poderão através da ferramenta poderão desenhar e escrever sobre a imagem registrando assim todas as alterações solicitadas e discutidas. Dentro do contexto dos Métodos Ágeis, o ato de manter o cliente ativo no processo de desenvolvimento do layout é uma de suas essências.

O Marqued mantém informado todos os envolvidos da equipe através de mensagens enviadas sempre que há alguma nova interação com o projeto e também permite níveis de acesso aos participantes, desde apenas permissão de visualização até editar e inserir novas imagens. Abaixo a Figura 9 mostra sua interface.

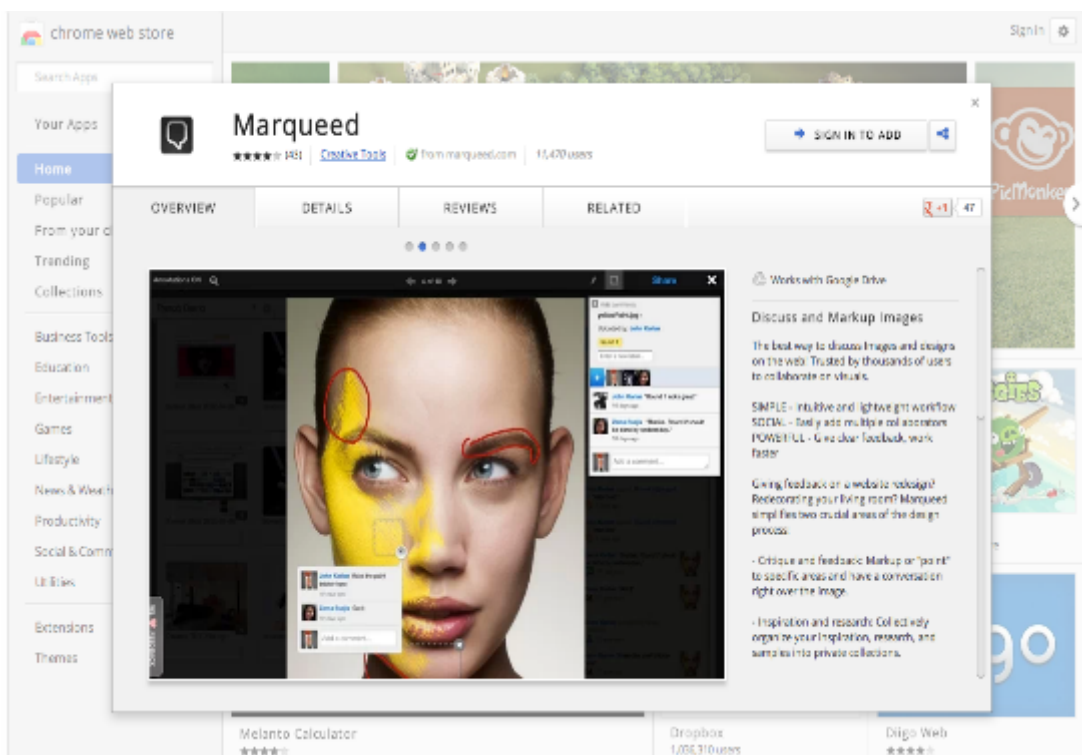


Figura 8 - Exemplo da interface do Marqueed.

Fonte: <http://www.marqueed.com> acessado em 05/09/2013.

3.5 Asana

O Asana pode ser considerado como um gerenciador de tarefas colaborativo, com a possibilidade também de gerenciar projetos. Pode ser utilizado por grupos de trabalho, empresas ou até mesmo para uso individual. Com o intuito de facilitar a colaboração entre os envolvidos no projeto, ele permite com facilidade maior de interação entre os membros da equipe, com alertas de interações via e-mail na maioria das aplicações de gerenciamento de projetos, o Asana tenta unificar todas as informações pertinentes aquele projeto ou tarefa em uma única página.

A aplicação pode ser acessado via navegadores, há aplicativos disponíveis para *Tablets* e *Smartphones* (Android e IOS), que o torna produtivo em situações em que a equipe estiver em atendimentos externos a clientes fora de seu ambiente de trabalho. A seguir, a Figura 10 mostra dois ambientes onde é possível acessar o Asana, *desktop* e *móbile*.

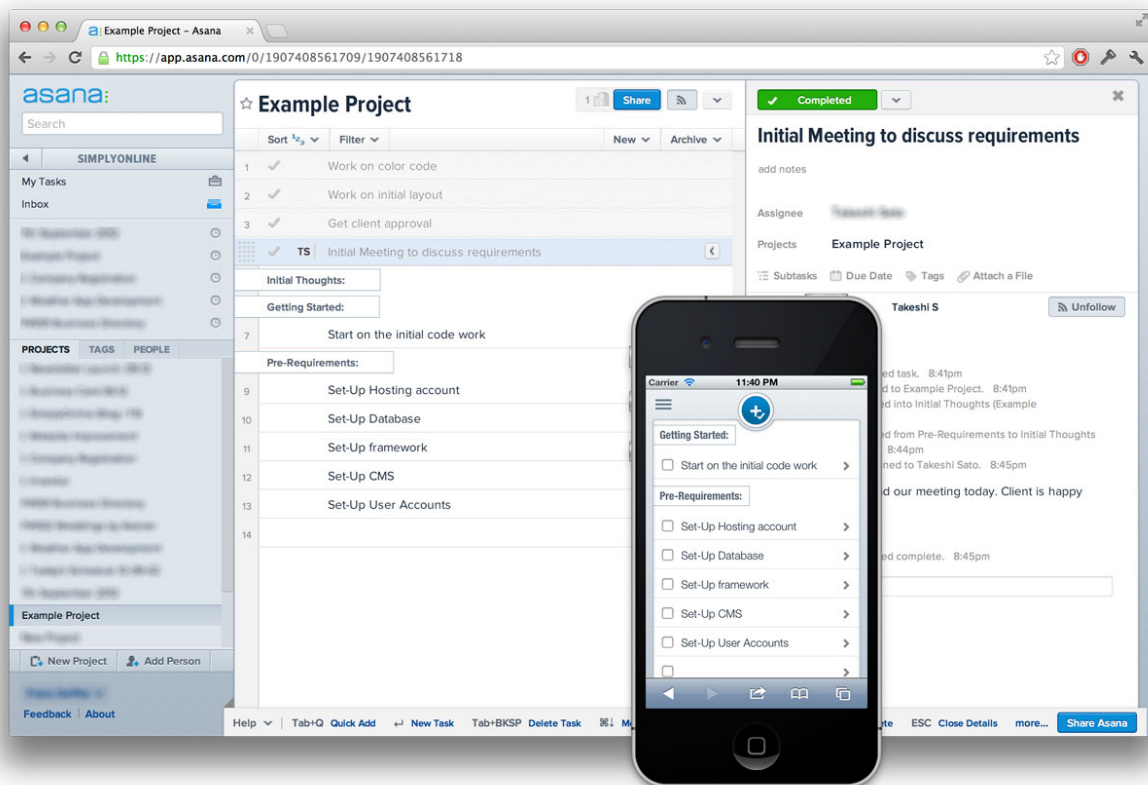


Figura 9 - Exemplo da interface do Asana.

Fonte: <http://www.asana.com> acessado em 05/09/2013.

3.6 Basecamp

O Basecamp foi lançado em 2004 e é uma plataforma web que pode ser utilizado para gerenciamento de projetos em diversas áreas. Com um layout simples e de fácil utilização, esta aplicação possui diversas funcionalidades, dentre elas: Compartilhamento de arquivos, capacidade de armazenar e compilar arquivos e pastas, lista de tarefas em forma de planilha que podem ser visualizadas, indicadas e delegadas aos responsáveis.

Assim como no Asana, metas, tarefas, mensagens, arquivos e dados são integrados com foco na potencialização da comunicação da equipe. Abaixo na Figura 11 podemos visualizar sua interface onde ele utiliza o Gráfico de Gantt.

The screenshot displays the Basecamp web interface. At the top, the browser address bar shows 'http://basecamp.com'. The navigation menu includes 'Basecamp', 'Projects', 'Calendar', 'Daily Progress', and 'Everyone Me'. A search bar is located on the right with the placeholder text 'Jump to a project, person, or search...'. The left sidebar contains two sections: 'Calendars' with checkboxes for 'Vacation Days', 'Birthdays', and 'Company meetings'; and 'Projects' with checkboxes for 'App Store project', 'Hiring project manager', and 'Fresh Brands project'. The main content area is titled 'Next 6 weeks' and shows a calendar grid from Sunday to Saturday. The current date is Thursday, the 9th. Various events are scheduled across the weeks, including 'Fresh Brands: Design phase', 'Vacation: Jennifer P.', 'App Store: Programming phase', 'Camping trip: Jamie', and 'Job Fair'. The interface uses a clean, organized layout with color-coded event bars.

Figura 10 - Exemplo da interface do Basecamp.

Fonte: <http://www.basecamp.com> acessado em 05/09/2013.

4. AVALIAÇÃO DO SISTEMA ONLINE ID CONTROLE

A falta de conhecimento de métodos de desenvolvimento ou gerenciamento de projetos, fez com que a equipe envolvida desenvolvesse uma ferramenta baseada em suas experiências e conhecimentos práticos.

A empresa ID Agência optou pelo desenvolvimento de sua própria ferramenta pois pleiteava um software simples com uma curva de aprendizado curta, onde seus colaboradores pudessem inserir suas experiências profissionais diretamente em seu desenvolvimento.

Então, nesta seção, após o embasamento teórico e levantamento de ferramentas similares ao ID Controle, será feita uma análise do sistema sob a ótica de Metodologias Ágeis e Gerenciamento de Projetos.

4.1 Breve histórico do desenvolvimento do ID Controle

Em novembro de 2009 a empresa ID Agência passou por uma consultoria com o profissional Agnaldo Rodrigues, que aplicou a técnica de análise de cenário SWOT, ou PFOA (Potencialidades, Fraquezas, Oportunidades e Ameaças), com objetivo de criar uma base para a gestão de um planejamento estratégico da empresa em questão.

A grande maioria dos pontos analisados não apresentava problemas ou eram de fácil solução, o que demandava atenção por sua situação mais crítica, era justamente o gerenciamento dos projetos dentro da empresa. A falta de gerenciamento adequado dos projetos e da equipe envolvida culminava em erros clássicos, porém fatais, que comprometiam prazos, qualidade e orçamento. A deficiência de comunicação interna das equipes de desenvolvimento e falta de *feedbacks* constantes com o cliente resultavam em retrabalho constante.

Diante dessa situação e com a análise SWOT concluída, a equipe de desenvolvimento da empresa criou a ferramenta chamada *ID Controle*, tomando-se apenas do conhecimento tácito de seus desenvolvedores e com base nas conclusões da análise para desenvolvê-la.

4.2 Avaliação e métodos de apuração

Com base no que foi levantado com o referencial teórico sobre as metodologias ágeis e o gerenciamento de projetos é possível fazer um diagnóstico sob as funcionalidades disponíveis no sistema atualmente, analisando segundo as seguintes vertentes:

- **É possível enxergar claramente os objetivos e metas do projeto:**

Atualmente o *ID Controle* não exhibe aos membros das equipes conteúdos mais detalhados do projeto em que estão envolvidos, somente os dados das tarefas que devem ser efetuadas naquele momento, seus prazos de entrega, se elas foram iniciadas, se estão pendentes ou paradas temporariamente. O sistema também se limita a exibir somente as tarefas inerentes ao seu colaborador, não é possível que outros envolvidos vejam ou colaborem entre si, Como pode ser observado na Figura 12.

The screenshot shows the user interface for Marcos Huss in the ID Controle system. At the top, there is a header with the user's name and three buttons: TAREFAS, TRÁFEGO, and SAIR. Below the header, there is a navigation menu with buttons for Pendentes, Minhas Tarefas, and Finalizadas. A search bar is also present. The main content area displays a list of tasks:

Data	Tarefa	Responsável	Término	Status
06/09	News Agosto - Terra Roxa Mídias Sociais	(Terra Roxa) (Marcos Huss)	06/09/2013	PENDENTE
ATIVO	Anuncio Revista	(Oleofil Equipamentos) (Marcos Huss)	06/09/2013	TERMINAR
<p>Marcos,</p> <p>tem um arquivo no servidor na pasta: Graficos/OleofilEquipamentos/anuncioRevista , nós vamos fazer uma anúncio de página inteira em uma revista pra eles, por favor veja dessas imagens e logos que temos em nossos BD que não tem qualidade de impressa para solicitar a eles.</p> <p style="text-align: right;">Dúvida</p>				
ATIVO	Calendário 2014	(Terra Roxa) (Marcos Huss)	09/09/2013	INICIADA
14/08	Desenvolvimento revista CD	(Humanish) (Marcos Huss)	10/09/2013	PENDENTE
ATIVO	Frazotto - Front End	(Terra Roxa) (Marcos Huss)	10/09/2013	INICIADA
11/09	Cartão de aniversário	(Terra Roxa) (Marcos Huss)	11/09/2013	CANCELAR

Figura 11 - Tela de visualização de tarefas pendentes: ID Controle.

Fonte: <http://www.idagenciadigital.com.br/idcontrole/> (acessado em 05/09/2013).

- **O mesmo projeto pode ser dividido em várias etapas conforme sua necessidade:**

Um mesmo projeto pode gerar várias etapas, no caso do ID Controle, cada etapa é convertida em uma nova tarefa, essas tarefas não têm relação entre si, não precisam de uma ordem cronológica de entrada e saída dos artefatos gerados.

- **Colaboração do cliente durante o período do desenvolvimento:**

As interações dos colaboradores e clientes são feitas com regularidade, porém são feitas pelos próprios membros da equipe no momento que finalizam uma tarefa e geram o artefato da mesma. O cliente não tem acesso ao ID Controle e não pode interagir com a equipe ou colaborador no momento em que a tarefa está sendo executada.

- **Interação e aceitação de mudanças ao longo do desenvolvimento:**

Quem recebe interações é o gerente do projeto, ele interage com a tarefa e o membro da equipe a quem foi designada, o gerente pode inserir mais detalhes, outras solicitações, correções ou apenas completar alguma instrução que pode estar em falta para a conclusão da tarefa. Caso o colaborador tenha alguma dúvida a respeito da tarefa em questão, pode solicitar através de um campo intitulado “dúvidas”, que informa ambas as partes por meio de e-mails, criando a possibilidade de aceitar interações ao longo do desenvolvimento, como mostrado na Figura 13 a seguir:

ATIVO | Anuncio Revista (Oleofil Equipamentos) (Marcos Huss) Término: 06/09/2013 [TERMINAR](#)

Marcos,

tem um arquivo no servidor na pasta: Graficos/OleofilEquipamentos/anuncioRevista , nós vamos fazer uma anúncio de página inteira em uma revista pra eles, por favor veja dessas imagens e logos que temos em nossos BD que não tem qualidade de impressa para solicitar a eles.

[Dúvida](#)

|

Figura 12 - Tela de formulário de dúvidas do projeto: ID Controle.

Fonte: <http://www.idagenciadigital.com.br/idcontrole/> (acessado em 05/09/2013).

- **Entregas rápidas e frequentes de partes do desenvolvimento:**

Um projeto dentro do *ID Controle* é fracionado em várias tarefas, isso permite que os prazos de execução de cada uma delas tornem-se reduzidos, permitindo ao gerente enviar partes e gerar com frequência partes do projeto, porém, não há como centralizar todas as tarefas que compõem um projeto.

- **Apresentação de versões funcionais:**

Este item acaba sendo definido pela quantidade de tarefas em que um projeto é fracionado, por exemplo, um sistema de uma loja online pode ser dividido em módulos, de produtos, de pedidos, dentre outros. Estes módulos podem ser apresentados regularmente em versões funcionais conforme forem sendo finalizados e liberados, porém, fica claro neste item que o uso constante do sistema se adaptou e acabou por tomar postura equivalente a descrita em metodologias ágeis.

- **O sistema propicia suporte necessário para que os envolvidos tenham confiança em sua utilização:**

O sistema foi desenvolvido em 2010 e desde então sofreu poucas alterações relevantes. Para a equipe de desenvolvedores que o utiliza desde o início, a ferramenta atende as necessidades, pois todos já estão familiarizados e acostumados com a ferramenta. Apesar do sistema se apresentar de maneira simples e com poucas funcionalidades, sempre se mostrou muito estável, passando segurança para seus usuários. Como pode ser observado na Figura 14.

Marcos Huss TAREFAS EQUIPE PROJETOS TRÁFEGO SAIR

Inserir Novo Projeto

Título:
Projeto de teste

Cliente:
Cliente de teste

Gerente de Projeto
Marcos Huss

INSERIR

Figura 13 - Tela de cadastro de projeto: ID Controle.
 Fonte: <http://www.idagenciadigital.com.br/idcontrole/> (acessado em 05/09/2013).

- **O sistema consegue ser ágil a ponto de não diminuir o ritmo de trabalho da equipe envolvida:**

Neste ponto ele atende bem as necessidades da empresa, cadastrar os projetos e tarefas é simples, não exige conhecimento técnico nem treinamentos especiais. Com o preenchimento de um formulário sucinto e com poucos campos as tarefas podem ser inseridas ou salvas como rascunho para poderem ser lançadas posteriormente. É possível visualizar o formulário na Figura 15.

Marcos Huss TAREFAS EQUIPE PROJETOS TRÁFEGO SAIR

Projeto:
139 - Projeto de teste

Início:
18/09/2013

Título da Tarefa:
Mea quas autem possim ut, duo in novum interesset.

Prazo: (dias)
4

Descrição: (+ detalhada possível)

Equipe
Equipe de Design

Membros
Marcos Huss

INSERIR TAREFA

Figura 14 - Tela de detalhamento da tarefa: ID Controle.
 Fonte: <http://www.idagenciadigital.com.br/idcontrole/> (acessado em 05/09/2013).

- **Consegue organizar as informações inseridas, arquivos e requisitos satisfatoriamente:**

São possíveis somente entradas de texto nos formulários de cadastro de projetos e tarefas, os requisitos devem ser preenchidos na descrição do projeto. Existe um campo de busca onde é possível localizar itens cadastrados através dos títulos e descrições dos projetos.

Não é possível alocar arquivos ou pastas no sistema, todos os artefatos gerados a partir das tarefas finalizadas criam notificações ao gerente do projeto que dá o encaminhamento necessário ao projeto.

- **Os membros da equipe conseguem discutir, refletir, com intervalos regulares maneiras mais corretas para ajustar seus comportamentos conforme a necessidade:**

Os membros das equipes de desenvolvimento acabam por discutir e refletir sobre comportamentos e ajustes somente entre a finalização de uma tarefa que gere um artefato a ser usado para dar início a outra tarefa subsequente. Nesse intervalo são discutidas ações que visam a melhorias dos processos envolvidos no projeto. Porém, como não há uma real interação das equipes envolvidas, uma equipe pode simplesmente executar sua tarefa sem levar em consideração dados relevantes para sua conclusão de maneira correta. Há no ID Controle uma tabela demonstrativa denominada “tráfego” onde é possível visualizar quem será o próximo a interagir (separação por cores) com o projeto, porém não é possível identificar qual será a tarefa a ser executada, como pode ser observado na Figura 16.



Figura 15 - “Tráfego” das tarefa e colaboradores do projeto: ID Controle.
Fonte: <http://www.idagenciadigital.com.br/idcontrole/> (acessado em 05/09/2013).

4.3 Melhorias

Alguns incrementos baseados em metodologias ágeis e na gestão de projetos podem melhorar significativamente o desempenho do sistema ID Controle, atendendo melhor toda equipe de desenvolvimento. Os benefícios que a equipe pode usufruir são:

- Reunir e exibir de maneira mais clara o objetivo e metas do projeto, permitir que todos os envolvidos visualizem seu *status*, propiciando a colaboração dos demais membros mesmo que estes ainda não estejam envolvidos com o projeto, um painel de fluxo com Kanban permitiria apoio visual necessário neste caso;
- Alterar o conceito de “tarefas” por “fases” e “subfases” do desenvolvimento, permitindo que o projeto seja único, e não várias tarefas não organizadas e individualmente. Isso permitirá a determinação de quais fases e subfases se enquadrarão nas etapas de planejamento, execução e finalização de um determinado projeto;
- A participação do cliente poderia acontecer com a simples implementação um alerta sempre que uma fase ou subfase iniciasse ou fosse finalizada, tratando o cliente como um

colaborador dentro da equipe, notificando-o sempre que o gerente do projeto julgar necessário;

- Tratar mudanças e alterações como subfases dentro de uma fase pode melhorar a dinâmica e agilidade nesses casos. Sempre que houver uma alteração no escopo da fase, esta já é inserida como subfase no projeto e já é assimilada pela equipe de desenvolvimento;
- Permitir comunicação e interatividade entre os membros das equipes, e também o cliente, deixando claro para todos o envolvidos em que fase o projeto se encontra, quais os colaboradores envolvidos naquele instante, quais serão responsáveis nas fases seguintes, permitindo que todos os envolvidos possam inserir informações, ideias e feedbacks de testes de versões, atualmente o sistema só permite a comunicação do desenvolvedor no momento de sua tarefa com o gerente do projeto;
- Facilitando a comunicação e interação de todos os envolvidos no projeto o sistema pode permitir e sugerir intervalos regulares, pré-determinados ou estabelecidos pelos membros da equipe e gerentes de projeto onde pode-se refletir como ajustar seu comportamento diante das necessidades do projeto.
- Utilizar mais elementos visuais, vistos por exemplo no conceito Kanban, que utiliza cartões distribuídos em colunas que podem ser nomeadas como fases de um projeto, utilizar também cores para determinar urgência, dificuldade e *status* de uma etapa, assim como visto no *framework* Cristal. Essas alterações de interface podem agilizar e tornar mais dinâmico a interação da equipe com o projeto, pois compilam várias informações do projeto com indicações visuais simples.

5. CONCLUSÃO

Como foi mencionado, o ID Controle surgiu de uma necessidade real da empresa ID Agência Digital em gerenciar seus projetos. Porém não se utilizou de nenhuma metodologia ou estudo de gerenciamento de projetos para apoiar sua confecção. Seu desenvolvimento teve como base apenas o conhecimento tácito da equipe e seus colaboradores.

É preciso destacar nesta avaliação que o ID Controle consegue atender a necessidade básica de organização das tarefas individuais dos membros das equipes. Apesar de possuir limitações o sistema é estável e tem bom funcionamento dentro de sua proposta. Em razão de sua simplicidade permite agilidade ao gerente de projetos distribuir as tarefas ao colaborador responsável, o qual permite ao menos controlar entradas e saídas dessas inúmeras tarefas que compõe um projeto.

Podemos concluir com a avaliação, que os conceitos, métodos e técnicas analisadas nas metodologias ágeis e no gerenciamento de projetos podem contribuir para uma evolução do sistema ID Controle. Tendo em vista que sua concepção não teve aporte teórico de nenhuma metodologia, mas assim mesmo apresenta características pertinentes a esses conceitos, é possível incrementar e desenvolver novas funcionalidades que com certeza trarão benefícios reais aos colaboradores e conseqüentemente a empresa e seus clientes.

Fica claro que há inúmeras possibilidades e melhorias a serem feitas, que podem ser colocadas em prática gradualmente, seguindo as necessidades e o aporte teórico levantado é possível tornar a ferramenta mais completa, atendendo ainda mais as necessidades da empresa e clientes, podendo até posteriormente gerar receitas, tornando-se um produto comercializável.

6. BIBLIOGRAFIA

ABRAHAMSSON, Pekka. et al. **Agile software development methods Review and analysis**. Relatório Técnico. Finlândia:VIT Publications. 2002.

Baumeister, H.; Wirsing, M. **Applying Test-First Programming and Iterative Development in Building an E-Business Application**. Disponível em <<http://www.pst.informatik.unimuenchen.de/projekte/caruso/ssgrr2002w.pdf>>. Acesso em: 18 ago. 2013.

CAETANO, Rodrigo. **Metodologias de desenvolvimento: Qual a mais adequada?**. Disponível em:<<http://computerworld.uol.com.br/gestao/2009/08/05/metodologias-de-desenvolvimento-qual-a-mais-adequada/>>. Acesso em: 20 ago. 2013.

GHISI, T. **Kanban no desenvolvimento de Projetos de Software: Entendendo os Desafios e a Receita para o Sucesso**. Disponível em <<http://www.garcia.pro.br/EngenhariadeSW/artigosMA/A6%20-%2045-6-%20Kanbam.pdf>>. Acesso em: 22 ago.2013.

KOSCIANSKI, André; SOARES, Michel dos S. **Qualidade de Software : aprenda as metodologias e técnicas mais modernas para o desenvolvimento de software**. 2.ed. São Paulo: Novatec Editora, 2007. 395p.

MANIFESTO. **Princípios por trás do Manifesto Ágil**. (2001). Disponível em: <<http://www.agilemanifesto.org/iso/ptbr/principles.html> >. Acesso em 31 jul. 2013.

MARTINS, José Carlos C. **Técnicas Para Gerenciamento de Projetos de Software**. Rio de Janeiro: Brasport , 2007. 456p.

MOURA, R. A. **"Kanban: A Simplicidade do Controle da Produção"**. Instituto IMAM. 1994.

OLIVEIRA, Ebenezer Silva de. **Uso de Metodologias Ágeis no Desenvolvimento de Software**. Disponível em <<http://www.cpdee.ufmg.br/~renato/TesesEDissertacoesOrientadas/Monografia-EbenezerSilvaOliveira.pdf>>. Acesso em 18 de ago. 2013.

PMBOK. Um guia do conjunto de conhecimentos em gerenciamento de projetos. 3.ed. Project Management Institute, Inc. 2004. 405p.

PMBOK. Um guia do conjunto de conhecimentos em gerenciamento de projetos. 4.ed. Project Management Institute, Inc. 2008. 337p.

PRESSMAN, R. **Engenharia de Software**. 6ª Edição. São Paulo: McGraw-Hill, 2006. 880p.

SCHAWBER, Ken. **Guia do Scrum**. Scrum Alliance, 2009.

SOARES, Michel dos S. **Metodologias Ágeis Extreme Programming e Scrum para Desenvolvimento de Software**. Disponível em: <<http://revistas.facecla.com.br/index.php/reinfo/article/view/146/38> >. Acesso em: 21 ago. 2013.

TAGUCHI, F. K. **Gerenciamento de Projetos: Serena Open Proj**. Disponível em <http://conhecimentolivres.net/material_publicacao/tecninformacao/projetos/opensproj_apostila.pdf>. Acesso em: 22 ago. 2013.

TELES, V.M. **Extreme Programming**. São Paulo: Novatec Editora, 2004. 320p.
Vargas, R. V. **Gerenciamento de Projetos: estabelecendo diferenciais competitivos**. 6.ed. Rio de Janeiro: Brasport, 2005. 250p.