



# PUC

ISSN 0103-9741

Monografias em Ciência da Computação  
nº 13/12

## **Adaptação na Prática de um Setor Público às Metodologias Ágeis**

**Bruno Siqueira Silva**  
**Gustavo Robichez de Carvalho**  
**Otávio Albuquerque Ritter dos Santos**

Departamento de Informática

**PONTIFÍCIA UNIVERSIDADE CATÓLICA DO RIO DE JANEIRO**  
**RUA MARQUÊS DE SÃO VICENTE, 225 - CEP 22451-900**  
**RIO DE JANEIRO - BRASIL**

## Adaptação na Prática de um Setor Público às Metodologias Ágeis

Bruno Siqueira Silva<sup>1</sup> Gustavo Robichez de Carvalho<sup>1</sup>  
Otávio Albuquerque Ritter dos Santos<sup>2</sup>

<sup>1</sup>Laboratório de Engenharia de Software – Pontifícia Universidade Católica do Rio de Janeiro (PUC-RJ)

<sup>2</sup>Ancine – Agência Nacional do Cinema

bsiqueira@gmail.com, robichez@gmail.com, otavio.ritter@gmail.com

**Abstract.** This work approaches the use of agile methodologies in software development in the context of governmental companies based on traditional contracts. Software development in such companies is mostly outsourced, which makes it difficult to hire highly qualified professionals and to efficiently use software development processes. In this paper, we explore this scenario with a case study in the context of Ancine - the national film cinema from Brazil. This study investigates the applicability of agile methods, specifically the Scrum framework, in outsourcing based software development. We present the results of our case study and discuss the current hiring process in governmental companies.

**Keywords:** scrum; agile methods; software development; governmental companies; outsourcing

**Resumo.** Este trabalho aborda as metodologias ágeis no desenvolvimento de sistemas para órgãos públicos em um modelo de contratação tradicional. O desenvolvimento de software em tais instituições é, em sua grande parte, terceirizado, o que dificulta a contratação de profissionais altamente qualificados e implantação de processos de desenvolvimento eficientes. Não distante deste cenário, está a Ancine, onde o desenvolvimento é terceirizado e apoiado por um processo em cascata. Um estudo de caso foi desenvolvido a fim de verificar e provar a aplicabilidade dos métodos ágeis, mais especificamente o Scrum, no contexto desta organização. Este trabalho apresenta os resultados deste estudo de caso e reflete sobre a forma de contratação atual dos órgãos públicos.

**Palavras-chave:** scrum; métodos ágeis; desenvolvimento de software; órgão públicos; terceirização

---

**Responsável por publicações**

Rosane Teles Lins Castilho  
Assessoria de Biblioteca, Documentação e Informação  
PUC-Rio Departamento de Informática  
Rua Marquês de São Vicente, 225 - Gávea  
22451-900 Rio de Janeiro RJ Brasil  
Tel. +55 21 3527-1516 Fax: +55 21 3527-1530  
E-mail: [bib-di@inf.puc-rio.br](mailto:bib-di@inf.puc-rio.br)  
Web site: <http://bib-di.inf.puc-rio.br/techreports/>

# Sumário

1	Introdução	1
1.1	Histórico	1
1.2	Cenário Atual	1
1.3	A abordagem ágil – fazer mais com menos	2
1.4	Proposta	3
2	Metodologias ágeis	3
2.1	Scrum	4
2.1.1	Processo Scrum Detalhado	5
2.1.2	Os Artefatos do Scrum	6
2.1.2.1	Backlog do Produto	6
2.1.2.2	Backlog do Sprint	7
2.1.2.3	Incremento do Produto	7
3	Áreas Envolvidas	7
3.1	Laboratório de Engenharia de Software da PUC-Rio	7
3.2	Área de TI - Ancine	7
3.3	SFO	8
4	Estudo de Caso	9
4.1	Planejamento	9
4.2	Organização do Projeto	10
4.3	O desenvolvimento	11
4.3.1	Visão geral	11
4.3.2	Tecnologia	12
4.4	Fatores Sociais	13
4.5	Formulário	13
4.5.1	Área TI	13
4.5.2	Área Cliente	16
4.6	Discussão	17
5	Conclusão	20
	Referências	21

# 1 Introdução

## 1.1 Histórico

Para que seja melhor compreendido o cenário atual de TI no setor público e mais especificamente na Administração Pública Federal, é necessário entender as reformas efetuadas no período compreendido entre 1979 e 1998 que gradualmente redefiniram o papel do Estado na produção de bens e na prestação de serviços públicos, bem como reorganizaram e fortaleceram as atividades governamentais. Neste contexto surge a preocupação com a eficiência e a efetividade da máquina administrativa, com a descentralização e com a implantação de controles gerenciais, a melhoria da capacidade da burocracia e a democratização da gestão orientada ao cidadão, com mecanismos de transparência e controle social.

Com a reforma administrativa de 1995 liderada por FHC e Bresser-Pereira, é feita uma clara distinção entre “atividades principais” a serem desempenhadas pelo Estado e “atividades auxiliares” que poderiam ser terceirizadas. Segundo Bresser-Pereira, “as atividades principais são as atividades propriamente de governo. São ações de legislar, regular, julgar, fiscalizar, policiar, definir políticas, fomentar. Mas para que essas funções possam ser realizadas, é necessário contar com o apoio de uma série de atividades ou serviços auxiliares: limpeza, vigilância, transporte, coperagem, serviços técnicos de informática e processamento de dados, etc. Estes serviços devem, em princípio, ser terceirizados já que são serviços de mercado, com substancial economia para o Tesouro.”

Como resultado direto desta estratégia, os servidores públicos responsáveis pelas atividades de TI nas organizações públicas foram colocados em um nível hierárquico distante do pensamento estratégico. Outra consequência foi o aumento da escassez de recursos humanos qualificados no segmento de TI em função de sua baixa remuneração ou mesmo pela carreira de menor prestígio na administração pública. O nível de terceirização aumentou chegando em alguns órgãos a assumir a totalidade das atividades de TI, inclusive na sua gestão, comprometendo muitas vezes a eficiência dos serviços públicos, motivo principal da reforma do Estado.

No que se refere a área de desenvolvimento de software, as consequências mais graves foram: falta de uma metodologia de desenvolvimento (processos ad-hoc) que garantisse a qualidade do produto final, pouca cultura de gestão de projetos e planejamento alinhado com as necessidades estratégicas da organização, desenvolvimento sob demanda de soluções pontuais criando ilhas de informação e com pouco foco no valor entregue, maior distanciamento entre a “área meio” que era a TI e “área fim” aumentando as barreiras de comunicação e entendimento dos requisitos e processos de negócio que deveriam ser automatizados.

## 1.2 Cenário Atual

É notório que a área de TI passa a ter cada vez mais um papel chave nas organizações públicas em função de demandas de governo eletrônico (serviços virtuais ao cidadão e interoperabilidade entre diferentes organizações), automatização de processos operacionais, transparência e integração de dados, análise e monitoramento de informações, segurança de informação, coleta de indicadores, etc. Tais projetos devem possuir alta qualidade na sua execução e geração de seus artefatos, visto que, na maioria, possuem grande impacto social. Tal qualidade torna-se virtualmente inatingível, já que, muitas

vezes, são realizadas dentro de prazos agressivos, orçamentos contingenciados, com requisitos que evoluem a todo momento, e apoiadas em processos complexos de contratação, que, raramente, são concluídos em menos de 6 (seis) meses.

Neste cenário, tem-se observado o crescente investimento do governo para profissionalização da gestão e de carreiras específicas para TI, bem como a readequação de estruturas organizacionais e implantação de modelos de Governança com o objetivo de promover uma maior integração entre as áreas de negócio e tecnologia de informação. Atualmente, considera-se imprescindível que os gestores e trabalhadores do campo da TI participem de forma ativa do processo de elaboração das políticas públicas, contribuindo com soluções tecnológicas e de processo para uma execução mais eficaz e eficiente.

Na área de desenvolvimento de software, em um levantamento feito pelo Tribunal de Contas da União (TCU) em 2010 junto a 315 instituições da Administração Pública Federal, mais da metade dos respondentes declararam, em relação à disciplina de processo de software, que além de não possuírem um processo definido, não chegam a alcançar nem “um processo informal repetido várias vezes e que implementa conceitos de qualidade de processo”. Os resultados sugerem que a aferição da qualidade do desenvolvimento está seriamente comprometida gerando produtos que não satisfazem as necessidades seja pelo aspecto funcional, de tempo ou custo.

Um ponto determinante a se considerar é que, no ambiente público, existe a proliferação de metodologias criadas a partir de experiências, não necessariamente de sucesso, dentro das organizações. O que acaba acarretando em um efeito em cascata em termos de aplicação de práticas, modelos, editais, e fóruns. Criam-se mitos e a avaliação cuidadosa de resultados fica comprometida de certa forma pois, em geral, erros são punidos e o mal funcionamento de determinada metodologia nem sempre é discutido abertamente.

A limitação de recursos humanos somados às complexas estruturas políticas de cadeia de comando criam um ambiente desafiador para o desenvolvimento de soluções de software no setor público. No caso específico de TI, este desafio é potencializado pela própria dinâmica da área de tecnologia que precisa acompanhar o ritmo de evolução tecnológica e as demandas da sociedade por mais serviços e transparência das informações. As equipes pequenas com pouco suporte e maturidade em processos são as que mais sofrem para quebrar o ciclo vicioso a que estão submetidas, já que estão trabalhando constantemente de forma reativa, gerando um grande tempo em retrabalhos.

### **1.3 A abordagem ágil – fazer mais com menos**

Para os órgãos públicos que não possuem uma metodologia de desenvolvimento ou se a possuem, não a usam por ser rígida ou complexa, cheia de artefatos e atores, que possuem recursos humanos escassos e limitada competência na disciplina de gestão de projetos, onde produtos de software são desenvolvidos sem ao menos haver priorização com um planejamento prévio e onde a forte hierarquia e a pressão constante por resultados minam esforços de estruturação de processos da área de TI, é necessário encontrar uma saída que, ao mesmo tempo, agregue valor para a organização sem que o processo de trabalho e o gerenciamento fiquem comprometidos.

Para este contexto, muito comum na Administração Pública Federal, a saída pode ser a aplicação de métodos ágeis de desenvolvimento de software e gestão não somente na área de TI, mas também nas áreas funcionais quando possível. Métodos ágeis não significam necessariamente executar algo mais rápido, mas criar valor em contextos onde é necessária a adaptabilidade e flexibilidade frente aos requisitos e prioridades.

A premissa da abordagem ágil é assumir que mudanças são inevitáveis e que não sabemos todos os aspectos do projeto no seu início. Seus processos são bem simples e acolhem as mudanças nos requisitos de negócio e no design, mesmo em fases mais avançadas do desenvolvimento. Baseiam-se num ciclo de vida iterativa e incremental. O uso de iterações permite que conjuntos de funcionalidades sejam produzidos e entregues para o cliente constantemente e com tempo reduzido entre cada versão intermediária. Isso torna o feedback para o produto e para o próprio processo mais rápido e eficiente, facilitando o gerenciamento de mudanças e a medição dos resultados.

Em função do foco no valor a ser entregue, é possível afirmar que este modelo tende a entregar mais funcionalidades “úteis” com a mesma quantidade de recursos, justamente pela constante priorização e foco no negócio somada a participação ativa do usuário no processo. No que se refere a terceirização, os métodos ágeis já estão relativamente assimilados na maior parte das empresas que prestam serviços de desenvolvimento de software e soluções para Internet, em especial as micro-empresas e startups, que abraçaram esse modelo inclusive na sua gestão. Estes métodos fornecem uma maior transparência dos serviços prestados justamente pelo constante ciclo de entregas e mensuração de resultados facilitando a gestão e fiscalização dos contratos.

## **1.4 Proposta**

A proposta deste trabalho é experimentar e avaliar a utilização prática de metodologias ágeis em uma instituição pública, coletando informações sobre o impacto no aprendizado, qualidade do código-fonte, produtividade e satisfação do cliente.

Para isso, um estudo foi executado dentro de uma superintendência da Ancine (Agência Nacional do Cinema) e realizado em parceria com a equipe do LES (Laboratório de Engenharia de Software da PUC-Rio). Ao final, foi distribuída uma pesquisa para os participantes com questões quantitativas e qualitativas. Ao longo do trabalho, são apresentados os dados positivos e negativos do piloto para que esta experiência possa auxiliar a avaliação desta prática em outros cenários.

Este trabalho está organizado da seguinte forma: a seção 2 apresenta os conceitos das metodologias ágeis. A seção 3 apresenta o perfil das diferentes áreas organizacionais envolvidas no projeto. A seção 4 apresenta a execução e avaliação do estudo de caso e a seção 5 a conclusão.

## **2 Metodologias ágeis**

Métodos ágeis referem-se a um grupo de processos de desenvolvimento de software baseados em desenvolvimento iterativo, onde os requisitos e as soluções para os problemas evoluem através da colaboração das equipes.

Métodos ágeis, geralmente, promovem um processo gerencial que encoraja:

- Freqüente inspeção e adaptação do que foi implementado às necessidades observadas no momento;
- Trabalho colaborativo em equipes auto-gerenciáveis;
- Um conjunto de boas práticas de engenharia de software, permitindo entregas freqüentes com alto grau de qualidade;

- Um conjunto de ferramentas de apoio ao processo e ao desenvolvimento, que provêm uma produtividade mais elevada;
- Uma abordagem de negócio que alinha o desenvolvimento com as necessidades do cliente e com os interesses da organização;

Existem várias abordagens para processos ágeis. Esta seção apresenta o framework utilizado no estudo de caso: Scrum.

## 2.1 Scrum

Scrum é um framework para o gerenciamento de projetos e, seu foco principal, é ser utilizado em projetos de desenvolvimento de software. Ele tem sido usado com sucesso para isso, porém, teoricamente, pode ser aplicado em qualquer contexto no qual um grupo de pessoas necessite trabalhar junto para atingir um objetivo comum.

As práticas do Scrum baseiam-se em um esqueleto incremental e iterativo (Ilustração 1). O esqueleto funciona da seguinte maneira: no início da iteração, a equipe revisa o que deve ser feito. Depois, seleciona o que acredita que pode compor um incremento no final da iteração. No fim da iteração o incremento é apresentado, permitindo sua validação junto aos *stakeholders*<sup>1</sup> do produto.

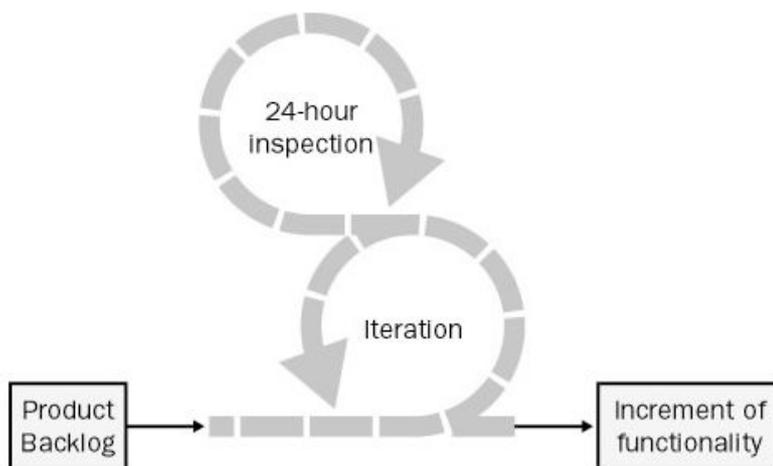


Ilustração 1 - Esqueleto do Scrum (Schwaber, 2004)

Na ilustração acima, o círculo inferior representa a iteração das atividades de desenvolvimento. A saída de cada operação é um incremento do produto. O círculo superior representa o trabalho realizado durante um dia e avaliado na reunião no início do dia seguinte, onde os membros da equipe se encontram para disseminar entre eles o que foi realizado no dia anterior e definir o que será feito no dia corrente.

---

<sup>1</sup> Qualquer envolvido diretamente no uso ou na confecção do produto.

### 2.1.1 Processo Scrum Detalhado

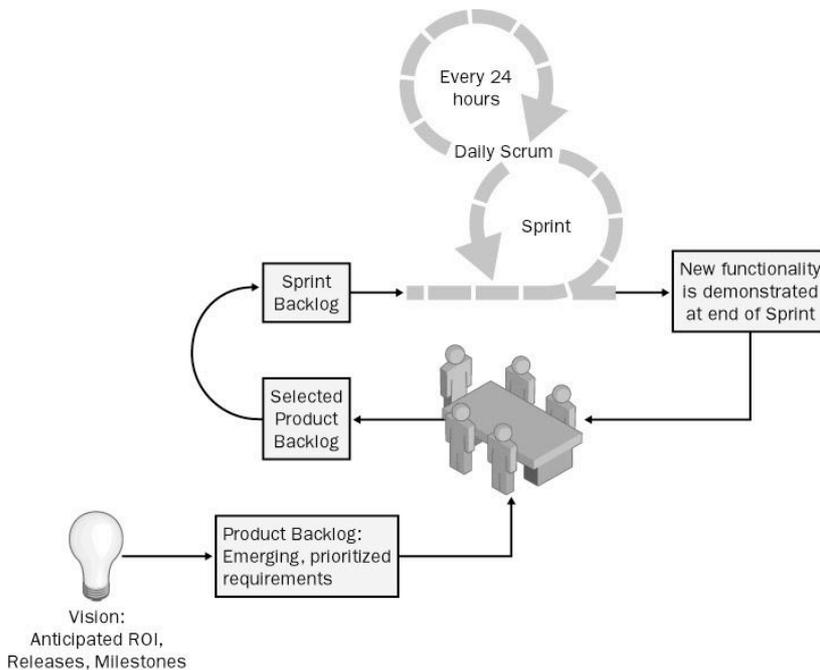


Ilustração 2 - Processo Scrum Detalhado (Schwaber, 2004)

Como apresentado na Ilustração 2, um projeto Scrum começa com a visão do sistema a desenvolver. O *Product Owner* (a partir deste ponto, nos referiremos a este papel como PO) é responsável por especificar os objetivos e requisitos do projeto e por prover a visão em uma maneira que maximize seu retorno de investimento (ROI - *Return of Investments*). O PO formula um plano do projeto que inclua o *backlog* do produto. O *backlog* do produto é uma lista de requisitos funcionais e não funcionais (incluindo solicitações de alterações, correções, funcionalidades novas etc.) que, quando transformados em funcionalidades, irão prover a implementação inicial do sistema. O *backlog* do produto representa os requisitos do sistema ainda a serem desenvolvidos ou aprimorados e é priorizado de maneira que, os itens que irão gerar mais valor possuam maior prioridade e sejam divididos em entregas diferentes. A priorização dos itens é um passo inicial, mas sua atualização é realizada ao longo da execução do projeto.

Todo o trabalho é feito em *sprints*. Cada *sprint* é uma iteração de 2-4 semanas, iniciado com uma reunião de planejamento, onde o PO e a equipe decidem juntos o que será realizado na próxima iteração. Selecionando os itens mais prioritários do Backlog, o PO passa para a equipe o que deseja que seja desenvolvido na iteração e a equipe por sua vez diz para o PO o quanto eles acreditam que poderá ser realizado.

A reunião de planejamento possui duas partes. Na primeira, o PO apresenta os itens de maior prioridade que deseja que sejam realizados pela equipe, enquanto esta tira o máximo de dúvidas sobre os itens apresentados. Quando a equipe já sabe o suficiente sobre os itens (ainda na primeira parte da reunião), ela seleciona os itens que acredita que podem fazer parte do *Sprint* dentro do limite de tempo estabelecido, e que compõem um incremento funcional do sistema. Durante a segunda parte da reunião, a equipe planeja o *Sprint*. Como a equipe é responsável por gerenciar seu próprio trabalho, ela precisa de um plano para começar o *Sprint*. As tarefas que irão compor este plano são organizadas em um *Backlog* do *Sprint* (o *Backlog* do *Sprint* não é o mesmo que o *Backlog* do projeto. Ambos são mais detalhados na seção 2.1.2). As tarefas evoluem conforme o *Sprint* evolui.

Todo início de dia a equipe se reúne para uma reunião de 15 minutos. Nesta reunião, cada membro da equipe responde a três perguntas: O que você fez ontem? O que fará hoje? Existe algum impedimento para realizar seu trabalho hoje? O objetivo desta reunião é sincronizar o trabalho de todos os membros da equipe, marcar alguma reunião necessária para a evolução da iteração e identificar possíveis fontes de problemas que afetem a continuidade do trabalho.

Ao término de cada *Sprint* são realizadas a revisão do *sprint* e uma reunião de retrospectiva.

O objetivo da revisão do *sprint* é a apresentação das funcionalidades terminadas ao *PO* e aos *stakeholders*. A revisão do *sprint* começa com a equipe apresentando o objetivo do *sprint*, Os itens do *backlog* planejados para o *sprint* e quais foram terminados. Diferentes membros da equipe podem debater sobre o que ocorreu bem ou mal no *sprint*.

A maior parte da revisão do *sprint* é gasta com os membros da equipe apresentando funcionalidades, respondendo perguntas dos *stakeholders* e verificando mudanças desejadas. O *PO* define com os *stakeholders* e a equipe mudanças potenciais no *Backlog* do produto baseadas no *feedback* da apresentação. No final da revisão do *sprint*, o *Scrum Master* apresenta a data e local da revisão do próximo *sprint* para o *PO* e os *stakeholders*.

Na reunião de retrospectiva, os únicos participantes são a equipe técnica, o *Scrum Master* e o *PO* (este é opcional).

A reunião começa com os membros da equipe respondendo duas perguntas:

- O que foi bem no ultimo *sprint*?
- O que pode melhorar no próximo *sprint*?

O *Scrum Master* anota as respostas em um formulário e a equipe debate sobre as melhorias potenciais para o próximo *sprint*. O *Scrum Master* não está na reunião para prover respostas, mas para auxiliar a equipe a encontrar meios para melhorar o processo *Scrum*. O objetivo final da reunião é revisar o processo *Scrum* para torná-lo mais eficiente e agradável na próxima iteração.

Após a reunião de retrospectiva, uma nova reunião de planejamento é marcada, iniciando um novo *sprint*.

Juntas, a reunião de planejamento, a reunião diária, a revisão do *Sprint* e a retrospectiva do *Sprint* constituem as práticas empíricas de inspeção e adaptação do *Scrum*.

## 2.1.2 Os Artefatos do Scrum

Um artefato é o resultado de uma ou mais atividades dentro do contexto do desenvolvimento de um software ou sistema e possuem algum critério de aceitação definido no processo (Schwaber, 2004). Artefatos que são entregue ao cliente (ou usuário) são conhecidos como produtos. Abaixo são apresentados os artefatos gerados ao longo da execução de um projeto utilizando a metodologia *Scrum*.

### 2.1.2.1 Backlog do Produto

O *backlog* do produto é a lista formada por requisitos a serem desenvolvidos, solicitações de alterações, correções, novas funcionalidades, etc. O *PO* é responsável por prover seu conteúdo, e a priorização dos itens registrados. O *backlog* do produto nunca está completo, e é simplesmente uma estimativa inicial (e contínua) dos requisitos. Conforme o produto e seu ambiente de desenvolvimento evoluem, o *backlog* do produto também evolui a partir de atualizações realizadas pelo *PO*.

### 2.1.2.2 Backlog do Sprint

O Backlog do Sprint define o trabalho (as atividades) que a equipe seleciona do *backlog* do produto e que corresponde a um incremento do sistema sendo desenvolvido ou mantido.

A equipe técnica e o *Scrum Master* definem uma versão inicial da lista dessas atividades na primeira parte da reunião de planejamento. É nela que é definido o que será desenvolvido e isso em concordância com as necessidades do *PO*. Ou seja, a versão inicial do backlog do sprint é criada a partir do backlog do produto. No final da segunda parte da reunião, com o planejamento refinado, eles possuem uma compilação final desta lista. As atividades devem ser divididas de maneira que levem entre 4 a 16 horas para serem finalizadas. Atividades mais longas que 16 horas, devem ser revistas e subdivididas em atividades menores. Somente a equipe pode alterar o *backlog* do *sprint*.

### 2.1.2.3 Incremento do Produto

O Scrum requer uma equipe para construir funcionalidades novas do produto a cada iteração. Este incremento deve ser potencialmente funcional, uma vez que o *PO* pode optar por colocá-lo em produção imediatamente. O incremento precisa estar largamente testado, bem estruturado, e com o código bem escrito para que seja gerado um sistema executável, pronto para o uso e com a documentação adequada. Esta é a definição de "pronto" do incremento.

## 3 Áreas Envolvidas

### 3.1 Laboratório de Engenharia de Software da PUC-Rio

O Laboratório de Engenharia de Software (LES) faz parte do Instituto de Tecnologia de Software do Departamento de Informática da PUC-Rio e desenvolve projetos inovadores de pesquisa avançada em áreas de engenharia que venham a requerer a aplicação e análise de técnicas e soluções altamente complexas. O LES liderará a execução do projeto desejado, sendo formada para tanto, uma equipe multidisciplinar com pesquisadores de alto nível, que permita alcançar os objetivos do estudo e pesquisa pretendidos e aproveitar a experiência da PUC-Rio nas áreas de arquitetura e metodologias de desenvolvimento de software.

Para a execução do projeto, o LES contou com 4 profissionais compondo a equipe Scrum: um Scrum Master e três desenvolvedores. A equipe foi formada por um mestre, um estudante do mestrado e dois da graduação.

### 3.2 Área de TI - Ancine

Os projetos da Ancine são normalmente desenvolvidos por empresas terceirizadas, incluindo sua manutenção corretiva e evolutiva. Do ponto de vista de desenvolvimento de soluções, os analistas de TI se concentram em análise de negócio e gestão de projetos. O Processo de Desenvolvimento utilizado é fortemente baseado no UP (Unified Process) e possui execução faseada, simulando cascata com artefatos a serem entregues para o término e aceite de cada fase. O uso dessa estratégia é reforçado pelo fato de que cada fase corresponde a um percentual de pagamento para a contratada caso os seus entregáveis sejam aprovados.

As fases estão definidas na seguinte seqüência: levantamento de requisitos, projeto/design/modelos, construção, testes, homologação e implantação. Ao final de cada fase, há uma etapa de validação, normalmente realizada pela área de qualidade.

Dois problemas apontados pela Ancine no uso deste processo são o excesso de documentação ao longo de cada fase e a rigidez do processo diante da necessidade de flexibilização em função do contexto dos projetos.

Outro problema decorre da falta de uma cultura de testes que não sofra impacto direto da urgência dos prazos. As evidências de teste validadas pela Ancine são em sua maioria fruto de testes manuais orientadas por planos de teste simplificados ou mesmo de caráter exploratório mas que acabam por ser muito limitados em função de prazos fixos de implantação. Apesar do contrato atual de desenvolvimento exigir testes automatizados para novos sistemas, estes tem pequeno percentual de cobertura e não são atualizados para suportar a regressão das manutenções subseqüentes, o que reduz sua própria eficácia e encarece todo o processo. As evidências de testes unitários, normalmente, não são apresentadas pela empresa terceirizada pela dificuldade da Ancine em processar e validar essa informação – o cenário de métricas de teste geradas automaticamente por meio de um ambiente de integração contínua é o desejado a médio prazo.

Um resultado desta configuração é a alta quantidade de não-conformidades na primeira rodada de validação de qualquer artefato, ocasionando muitas vezes a sua rejeição, e aumentando o tempo do projeto, visto que outras etapas formais de validação se tornam necessárias. Segundo a Ancine, em seu histórico de projetos, são necessárias, no mínimo, 3 rodadas de validação para obter a aprovação de um artefato.

Em uma pesquisa realizada no início do estudo de caso, a Ancine identificou que as três principais razões para a grande quantidade de falhas nos projetos são:

- Falta de testes durante o desenvolvimento dos sistemas;
- Falta de técnica apropriada para testar os softwares;
- Falta de comprometimento das empresas terceirizadas em desenvolver software com qualidade;

### **3.3 SFO**

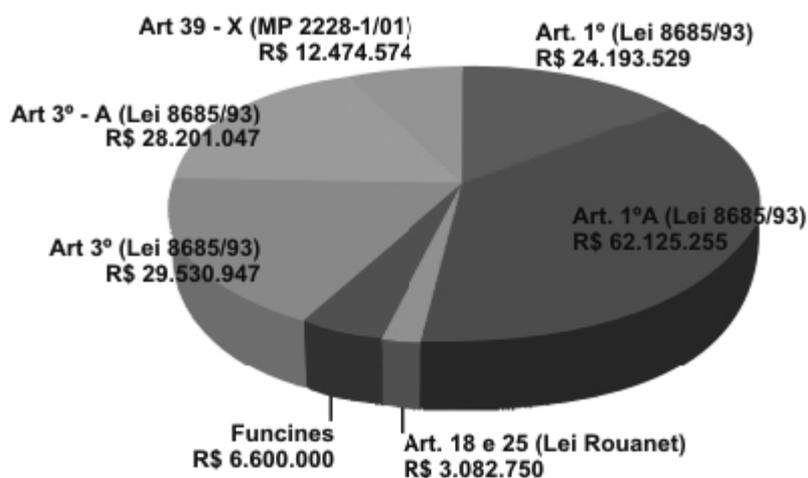
A SFO (Superintendência de Fomento da Ancine) é o setor da Ancine responsável pela análise, acompanhamento e prestação de contas dos projetos submetidos à Ancine.

O Fomento Indireto compreende o financiamento a projetos por meio de patrocínio, investimento ou co-produção, com utilização dos mecanismos de incentivo fiscal criados pela legislação audiovisual.

Por meio de mecanismos de renúncia fiscal, em 2010 (ano em que o filme brasileiro atingiu a participação de 18,78% do market share) foram mobilizados mais de R\$ 180 milhões para a viabilização de projetos audiovisuais para os segmentos de mercado de TV Aberta, TV Paga, Salas de Exibição e Vídeo Doméstico além de projetos de infraestrutura, aquisições de ações e festivais internacionais <sup>2</sup>.

---

<sup>2</sup> <http://boletim.ancine.gov.br/especial001/>



**Ilustração 3 - aplicação do incentivos fiscais em 2010**

Ao longo dos últimos seis anos, houve um aumento significativo não só do rol de instrumentos de financiamento público do audiovisual, mas também do volume de recursos movimentados, que praticamente dobrou nesse período.

Entre 2005 e 2010, houve um aumento de R\$ 34 milhões de investimentos de recursos públicos federais administrados pela Ancine. Neste período houve também aumento de 62% do número de novos projetos apresentados à Ancine, bem como da complexidade desses projetos. O crescimento do mercado se reflete também no número de agentes econômicos regulados, que passou de 629 em 2005 para 1338 em 2010.

No cenário atual, o fluxo de entrada de pedidos de avaliação de projetos é intenso: a cada semana, 15 novas demandas de análise entram, fora os processos já em andamento e os em análise de diligência.

Todo o processo é manual. Os proponentes enviam a documentação necessária pelo Correio ou protocolam nas cidades onde a ANCINE possui escritórios. Esta documentação enviada é então revisada pelos analistas da Agência por meio de um processo de triagem documental com o objetivo de verificar se cada proposta reúne condições de análise, seguindo depois por outras etapas de análise técnica (disciplina de direitos) e orçamentária (cotejamento do roteiro/orçamento).

## 4 Estudo de Caso

### 4.1 Planejamento

Foi definido que a solução a ser implementada permitiria a automatização do processo de aprovação/avaliação de projetos dentro da SFO. Com o sistema informatizado, os proponentes poderiam entregar toda a documentação de maneira digital e as informações prestadas pelos agentes viriam com uma melhor qualidade e consistência, posto que, determinadas críticas de negócio poderiam ser realizadas automaticamente. O valor de negócio desta solução estaria então na melhoria dos serviços prestados pela Ancine à sociedade e na otimização operacional da própria SFO na redução das diligências para retificação das informações prestadas em tempo de submissão dos projetos e na possível paralelização de atividades, uma vez que deixa de se tornar necessário o uso de papel impresso como insumo de uma tarefa.

A SFO foi escolhida como área cliente para o piloto pelas seguintes razões:

- A carência de um sistema automatizado para apoiar seu processo de trabalho: até o momento, toda análise realizada pela SFO era praticamente manual, onde o fluxo do processo de análise e aprovação se refletia diretamente na tramitação do processo administrativo físico (papel).
- O momento de definição de um novo processo de trabalho, abrindo a oportunidade para discussão e interesse da área cliente em apresentar as funcionalidades.
- Participação ativa e comprometimento do líder executivo da SFO durante as sessões de *review* e planejamento. O superintendente participou desde o início das decisões sobre o desenvolvimento do produto.

## 4.2 Organização do Projeto

A organização do projeto foi pensada para refletir um cenário de terceirização típica de desenvolvimento de software no âmbito governamental, onde a área de TI da Ancine contrata uma empresa especializada para atender às demandas de negócio dos seus usuários.

Assim, o time da PUC/LES passou a simular uma empresa terceirizada e a TI fazer o papel de intermediadora e facilitadora do processo junto ao usuário.

Dentro dos papéis do Scrum tivemos a seguinte organização:

- Scrum Master e time: representado pelo time da PUC/LES que se auto-organizou na distribuição das tarefas e planejamento das iterações. Durante os reviews com o usuário, o time era representado pelo Scrum Master que apresentava o produto e capturava o feedback dos POs. O fato de ter apenas o Scrum Master e não o time todo presente no review era apenas pela dificuldade de logística posto que as apresentações eram feitas no centro do Rio de Janeiro pela manhã e na maior parte das vezes incompatível com as tarefas acadêmicas do restante do time que concentrava suas atividades no período da tarde.
- PO (PO): representado pela Superintendência de Fomento e mais especificamente por dois de seus coordenadores. Ressalte-se a ampla participação do próprio Superintendente no processo o que muitas das vezes agilizou a tomada de decisões durante as reuniões de revisão da sprint e de planejamento.
- TI da Ancine: A área de TI da Ancine exerceu dois tipos de responsabilidade dentro da organização de projeto. A primeira foi ser também um PO principalmente para aspectos não-funcionais relacionados à solução como usabilidade e integração com sistemas legados. A segunda foi ser uma extensão do próprio time fazendo uma espécie de tradução das necessidades de negócio de regulação do audiovisual, auxiliando a priorização e planejamento bem como na definição das regras de negócio das funcionalidades.

Dado o número reduzido de colaboradores e a dependência de fornecedores, é importante que as organizações públicas definam uma estrutura de projeto adequada para potencializar o negócio e entregar soluções de alto valor agregado.

## 4.3 O desenvolvimento

### 4.3.1 Visão geral

Na primeira reunião do projeto, foram definidos dois módulos que seriam implementados no sistema: módulo de habilitação e de aprovação. Para cada um deles, foi definida uma data de entrega, com aproximadamente um mês e meio dedicado para cada um. A partir da primeira reunião, também foi definido um backlog inicial de alto nível.

Foi feita para a área cliente e TI uma apresentação sobre Scrum, com o objetivo de disseminar as metodologias ágeis entre as diferentes partes do projeto, conscientizando cada um sobre o formato do processo de trabalho dos meses seguintes, a importância da participação ativa e da divisão de responsabilidades de todos. Em geral, os conceitos das metodologias ágeis foram bem compreendidos por todos (Como pode ser visto na seção 4.5). Imediatamente após a apresentação foi feita a primeira reunião de planejamento. Nesta reunião, foi apresentado o backlog inicial. O PO imediatamente se sentiu confortável para redefinir, detalhar com mais precisão e repriorizar o backlog. A partir do resultado deste trabalho, as primeiras histórias foram selecionadas como candidatas para a primeira sprint.

No planejamento da primeira sprint, como ainda não havia histórico de estimativas, as primeiras histórias foram pontuadas pela equipe e incluídas automaticamente na sprint.

O tempo definido para as sprints foi de duas semanas. Ao final de cada sprint foi realizada a reunião de revisão da sprint na Ancine, com o PO e os stakeholders. Os feedbacks recebidos em cada reunião foram inseridos no backlog e repriorizados juntamente com as outras histórias.

Junto à equipe de desenvolvimento, foram realizadas as retrospectivas da sprint, onde os membros puderam descrever as oportunidades de melhoria no processo de trabalho.

Fora as conversas nas reuniões de revisão, toda a conversa com o PO se deu por emails. A equipe de desenvolvimento tinha liberdade para contatar o PO diretamente sem necessidade de passar por alguém de hierarquia mais alta. Neste ponto, a área de TI agiu como intermediador, cobrando do PO a necessidade de resposta rápida para o andamento do desenvolvimento.

Das cinco sprints, em duas delas foi necessário re-agendar (normalmente para três ou quatro dias depois do agendado) as reuniões de revisão da sprint, devido a imprevistos na agenda da área cliente. Nestes cenários, ao invés de iniciar novas histórias, foi dada prioridade para melhorar testes automatizados e resolver bugs, embora não houvesse uma listagem prévia de defeitos levantada.

Após dois meses do início do projeto, pelo andamento e pela complexidade do primeiro módulo, foi constatado que não seria possível finalizar os dois módulos no prazo definido para o piloto. A partir deste cenário, uma reunião foi realizada na própria PUC-Rio com dois objetivos:

- Realizar uma apresentação sobre a tecnologia na própria PUC, por parte da equipe para a área de TI e a área cliente.
- Com um prazo limite definido para o projeto, rever o Backlog junto ao PO e chegar num acordo de sua redução para se adequar ao tempo restante. Atividade que foi realizada com sucesso.

### 4.3.2 Tecnologia

A Ancine ressaltou no início do projeto estar aberta às possibilidades tecnológicas sugeridas pela PUC, mas que daria preferência para tecnologias em Java, visto que todos os seus sistemas já estavam configurados utilizando o JBoss e que facilitaria a integração com os outros sistemas.

A PUC optou por desenvolver o sistema utilizando o Play Framework, que é um framework desenvolvido em Java voltado para aplicações web com arquitetura MVC e *stateless*<sup>3</sup>. O Play framework foi arquitetado para se adaptar ao desenvolvimento ágil, permitindo um trabalho incremental, sem necessidade de grandes definições arquiteturais antecedendo ao projeto, visto que sua arquitetura já é bastante padronizada.

A curva de aprendizado do framework foi extremamente rápida. Isso pode ser observado na evolução das sprints:

- Na primeira nenhuma história planejada foi totalmente fechada, na segunda sprint foram fechadas 4 histórias. As duas primeiras sprints totalizaram 19 pontos.
- A terceira sprint sozinha englobou 4 histórias que totalizaram 18 pontos.
- A quarta e a quinta sprints (a quinta sprint possuiu um formato mais próximo de burndown list) totalizaram 4 histórias somaram 32 pontos. A maior parte deles finalizados na sprint 4

O uso do Play Framework se mostrou extremamente produtivo, com curva de aprendizado rápida e retorno eficiente para o cliente.

Para auxiliar na compreensão do sistema e na evolução dos requisitos durante as revisões da sprint, foram utilizados diagramas UML para apoiar as apresentações trazendo maior base para discussão dos requisitos por parte da área cliente.

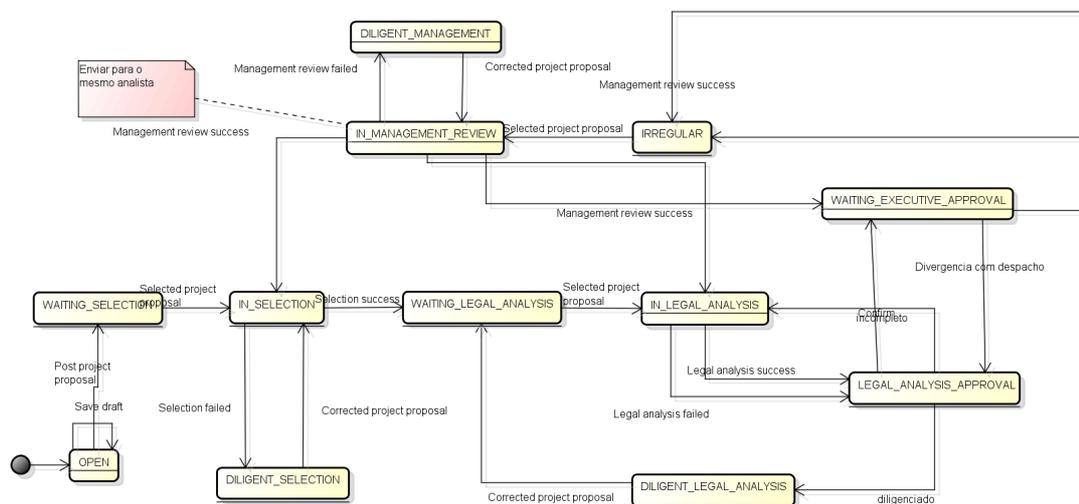


Ilustração 4 - Parte do Diagrama de Estado Utilizado como documentação de apoio

<sup>3</sup> Protocolo de comunicação que trata cada *request* como uma transação independente não relacionada a nenhum *request* anterior, fazendo a comunicação consistir entre *request* e *response*. Um protocolo *stateless* não requer que o servidor retenha na sessão informações ou estados de cada par de comunicação pela duração de vários *requests*.

## 4.4 Fatores Sociais

Um aspecto fundamental na experiência relatada neste documento foi a relação de confiança entre os diversos atores durante o desenvolvimento do produto.

Do ponto de vista do Cliente, através das revisões periódicas e constantes, houve uma significativa melhoria na percepção de estimativa e custo da funcionalidade (ou mudança de requisito) e o que é viável de ser realizado dentro do prazo demandado (em outras palavras a produtividade do time é medida pelo software entregue e apresentado nas reuniões). Assim, em incontáveis situações, o PO avaliou se era mesmo necessário investir em uma mudança ou funcionalidade contra uma outra que estava com menor prioridade mas deveria subir na urgência em função do amadurecimento das regras de negócio. Não é exagero afirmar que permanentemente o PO estava analisando os prós e contras dos cenários de priorização auxiliado pelo Scrum Master e pela própria área de TI, aumentando a confiança em seu produto. Esse exercício ajudou a colocar o PO como parte ativa da equipe, e não apenas como um participante passivo e distante da criação do produto.

Do ponto de vista da TI da Ancine, como o PO participava ativamente do processo decisório, as responsabilidades nas tomadas de decisão não ficavam apenas com a equipe técnica, mas eram divididas também com a área cliente. A avaliação dos formulários respondidos (seção 4.5.2) mostrou que a área cliente conseguiu melhorar o relacionamento com a TI da empresa, eliminando vícios organizacionais, como a interminável busca de culpados em casos de atraso ou inadequação na entrega de projetos. Como todos participaram do processo, os erros que encontrados foram assimilados como experiência para as próximas iterações em uma atitude pró-ativa e colaborativa que deveria ser, em tese, o comportamento normal nas organizações.

## 4.5 Formulário

Ao final do estudo de caso, foram enviados dois formulários de avaliação do projeto para preenchimento da Ancine. O primeiro para a área cliente e o segundo para a área de TI. Ambos foram fortemente baseados no formulário de avaliação de [MELO, 2010].

### 4.5.1 Área TI

No total, seis pessoas responderam o questionário. Nas quais, três desenvolvedores, dois analistas de requisitos e um gerente. A maioria já trabalhava na instituição há mais de seis anos, já possui mais de cinco anos de experiência com projetos de desenvolvimento. Já haviam participado de outros projetos de desenvolvimento, implementados por terceirizadas e afirmam que o período de apresentação do sistema era acima de 6 meses.

Durante o projeto, 1 profissional considerou seu papel próximo ao do PO, 2 se consideraram stakeholders e 3 desenvolvedores. Os papéis identificados como desenvolvedores podem se justificar pelo apoio dado ao desenvolvimento, provendo scripts SQL, ajustes de infra e a manutenção do sistema posterior ao estudo de caso.

Abaixo, é apresentada uma tabela com os principais conceitos apresentados pelo projeto e quantos profissionais se identificaram com qual nível de aprendizado.

	Imediatamente compreensível	Tive certa facilidade	Demorei, mas consegui	Não entendi até hoje
Scrum	3	2		1
Sprint (Iterações)	5			1
Revisão da Sprint	4	1		1
Backlog do Produto	4	1		1
Backlog da Sprint	4	1		1
Priorização do Backlog	5			1
Histórias	4	1	1	
Tarefas	4	1	1	

**Tabela 1 - Entendimento dos Conceitos do Scrum pela Área TI**

Abaixo, são apresentados dois depoimentos das principais dificuldades identificadas no aprendizado de metodologias ágeis:

*"Acredito que o mais difícil é a mudança da cultura, principalmente por parte dos stakeholders que tem que participar mais ativamente do projeto. No caso do projeto piloto os stakeholders perceberam um ganho de produtividade, principalmente por que cada interação era apresentado software executável, ficando mais palpável sua percepção e pedidos de alterações"*

*"Em primeiro lugar foi entender todo o funcionamento desta metodologia, em especial do SCRUM. Ou seja, como era planejado o dia-a-dia, as tarefas, as histórias, etc. Em segundo lugar, outra dificuldade foi a de que apesar de comprovarmos a eficiência da metodologia, talvez tenha faltado uma aproximação maior com a PUC, de forma que o aprendizado fosse mais disseminado."*

Abaixo, são apresentadas as características identificadas que mais favoreceram o aprendizado de metodologias ágeis:

- Reuniões periódicas com os clientes;
- Usos de tecnologia apropriada e de fácil entendimento (play framework);
- A rápida entrega de um produto desenvolvido, funcional e de acordo com as expectativas do cliente;
- Reuniões rápidas, sem produção de documentação em excesso;
- Práticas que exigiam maior interação entre a área TI e o cliente;
- Simplicidade;
- Execução sem perder o mínimo de planejamento;
- A possibilidade do cliente acertar os pontos não satisfatórios no sistema em cada iteração;
- Regras de negócio e fluxo do sistema documentados por meio de testes automatizados;

- Mudanças e evolução dos requisitos como parte do processo de construção e não como fatores que causam impacto no andamento do projeto;

Com relação à execução do projeto, a área TI chegou às seguintes conclusões: metodologias ágeis ajudam a aumentar a produtividade do time e a qualidade do código bem como auxiliam no aumento de confiança do usuário em relação ao produto. A maior parte da equipe achou que a satisfação com o sistema entregue foi maior e que o aprendizado de novas tecnologias foi mais fácil. A equipe achou que metodologias ágeis podem ajudar no relacionamento com a terceirizada já que possibilita um fluxo mais constante de entrega e avaliação dos serviços prestados, com maior foco em software funcionando (incluindo testes automatizados) e menor foco em documentação e geração de papel

A área TI identificou como positivo, em geral, os seguintes pontos:

- Interações com periodicidade curta com o cliente;
- Resolução de backlog mais rápida;
- Desenvolvimento apoiado por testes automatizados;
- Equipe de desenvolvimento qualificada;
- Rápida entrega de produtos nas iterações definidas;
- Pouco tempo gasto em documentação, procedimento que consome muito tempo de execução;
- Foco e interesse dos clientes no projeto;
- Reuniões rápidas e definições sem discussões demasiadas;
- Otimização e dinamismo na especificação das funcionalidades pelo cliente;
- Confiança e visibilidade do andamento do projeto;
- Oportunidade de apresentação de novas tecnologias;
- Priorização do usuário em definir o que é realmente importante para o suporte ao seu negócio;

Como maiores dificuldades foram identificadas:

- Necessidade de maior interação entre a executora do projeto piloto e a área de TI da empresa modelo;
- Melhoria na comunicação entre os participantes do projeto. O que melhoraria e equalizaria o conhecimento do escopo do projeto e da tecnologia;
- Ausência de uma apresentação para a área de TI com o objetivo de disseminar e equalizar os conceitos de metodologia ágil;
- Ausência de uma apresentação das tecnologias utilizadas, frameworks e linguagens de programação no projeto;
- Monitoramento do andamento dos sprints pela dificuldade da utilização da ferramenta Rational Team Concert (RTC);
- Baixa priorização dos requisitos não funcionais, como a integração com sistemas legados da empresa;
- Tempo curto para o piloto, gerando redução de escopo em relação às expectativas iniciais do usuário;

Foram destacados dois trechos do campo de observações gerais:

*“O projeto foi muito proveitoso e demonstrou para os participantes que a metodologia ágil é uma ferramenta eficaz para o desenvolvimento de sistemas, aliando a rapidez das entregas com as necessidades dos usuários. De certo modo, provou-se que a entrega acelerada de um sistema(ou parte dele) pode ser realizada sem que haja detrimento da perda ou omissão de requisitos, pois os mesmos são evoluídos a cada iteração, permitindo que os usuários estejam mais centrados no desenvolvimento e auxiliando-os na elucidação e detalhamento destes requisitos. Logicamente, que a metodologia ágil, para atingir a eficiência proposta, necessita de uma equipe experiente e capaz de funcionar de forma coesa e com alto grau de comprometimento com o trabalho e que cultive um bom relacionamento com os envolvidos no projeto.”*

*“Fundamental a participação ativa do Cliente no processo, em especial, do próprio Superintendente que ajudou a construir e evoluir o produto junto com o PO formal”*

#### 4.5.2 Área Cliente

No total, quatro pessoas responderam a avaliação. Duas trabalham na empresa entre dois e cinco anos, e as outras duas entre seis e nove anos. Dois já haviam participado de outros projetos de desenvolvimento e responderam que o tempo para a apresentação do sistema era acima de 6 meses.

Dentro do Scrum, dois usuários identificaram seus papéis como POs, um como stakeholder e um como testador.

Abaixo, é apresentada uma tabela com os principais conceitos apresentados pelo projeto e quantos profissionais se identificaram com qual nível de aprendizado.

**Tabela 2 - Entendimento dos Conceitos do Scrum pela Área Cliente**

	Imediatamente compreensível	Tive certa facilidade	Demorei, mas consegui	Não entendi até hoje
Scrum	2	2		
Sprint (Iterações)	2	2		
Revisão da Sprint	1	3		
Backlog do Produto	1	3		
Backlog da Sprint	1	2		1
Priorização do Backlog	2	2		
Histórias	1	2		1
Tarefas	1	2		1

Duas das maiores dificuldades identificadas pela área cliente no aprendizado de metodologias ágeis foram a interferência do operacional diário, que prejudicou o acompanhamento do andamento do projeto com sua devida constância, e a falta de

preparo do setor para uma metodologia tão diferente da usual, devido ao seu alto nível de exigência na participação no projeto por parte do cliente.

Na contrapartida, os seguintes itens foram identificados como principais facilitadores no aprendizado das metodologias ágeis:

- Fácil interlocução com a equipe de desenvolvimento da PUC;
- A agilidade e o processo contínuo com a apresentação do resultado a cada encontro;
- Apresentação clara da metodologia no início do projeto, determinando o formato o processo de trabalho de antemão;
- A disponibilidade de um projeto que a área cliente possuía domínio para realizar debate qualificado;

Com relação a execução do projeto, foi unânime entre os avaliadores que o uso da metodologia aumentou a produtividade e melhorou o relacionamento com a área de desenvolvimento (TI e PUC), o cliente ficou mais satisfeito com o sistema entregue e a metodologia auxiliou no aprendizado das funcionalidades do sistema.

Abaixo, os pontos positivos do piloto:

- O curto espaço de tempo a transformação de uma idéia de um sistema para um software operacional;
- Redução do tempo de levantamento/atualização de requisitos devido ao contato constante com o software em desenvolvimento;
- Interlocução de qualidade;
- Mudança de perspectiva sobre performance no desenvolvimento de sistemas;
- Conhecimento da técnica de métodos ágeis para a elaboração de sistemas;

Os seguintes pontos foram identificados como necessidades de melhoria para os próximos projetos:

A área cliente identificou a necessidade de aumentar o tempo de desenvolvimento no próximo piloto, pois tempo reduzido não permitiu que o escopo implementado alcançasse a proposta inicial. Também identificou a necessidade da escolha de outra ferramenta de acompanhamento, já que a ferramenta Rational Team Concert (RTC) se mostrou bastante complexa para o usuário.

## **4.6 Discussão**

O desenvolvimento ficou todo concentrado nas instalações da PUC. A área de TI da Ancine agiu como stackholder e facilitador na comunicação com a área cliente. Esta configuração de trabalho foi um dificultador no momento da passagem de conhecimento. Na fase de manutenção do projeto, houve uma certa dificuldade dos profissionais da TI em compreender o Play Framework e as regras de negócio do sistema também.

Uma boa forma de melhorar a passagem de conhecimento teria sido alguns profissionais da área de TI participarem do desenvolvimento, mesmo que com responsabilidade menor, como relata [GABARDO, 2009].

A implantação do sistema também foi problemática, visto que ocorreram conflitos de bibliotecas no servidor da Ancine entre o sistema desenvolvido e os outros já em produção. Este problema não foi previsto com antecedência, pois o foco do piloto foi na construção das funcionalidades e na aplicação do processo, o que gerou atraso na disponibilização do sistema para testes e homologação do usuário no ambiente tecnológico da Ancine. A integração contínua poderia ter adiantado este problema, com um ambiente de teste e *deploy* automático.

A aplicação final contabilizou 120 pontos por função (contagem realizada após o fim do projeto). Pela tabela de prazos do contrato atual de desenvolvimento de software da Ancine (Tabela 3), o projeto tomaria, no máximo, quatro meses e meio, e levou três meses com uma equipe bastante reduzida. O que demonstra os ganhos da utilização de um processo incremental e adaptável.

Tamanho do Sistema, Sítio ou Portal por Ponto de Função	Prazo máximo em dias corridos para conclusão do serviço	Início do Serviço
Menor ou igual a 50	87	Primeiro dia útil contado da aprovação do Anteprojeto pela ANCINE
Maior que 50 até 100	108	
Maior que 100 até 200	135	
Maior que 200 até 300	153	
Maior que 300 até 400	168	
Maior que 400 até 500	180	
Maior que 500 até 600	192	
Maior que 600 até 700	201	
Maior que 700 até 800	210	
Maior que 800 até 900	219	
Maior que 900 até 1000	225	
Acima de 1000	Negociável entre as partes	

**Tabela 3 - Tabela de estimativas de tempo por pontos por função da Ancine**

Como pode ser observado nos resultados dos formulários (seção 4.5 ), houve certa discrepância entre o sistema entregue e o escopo definido no início do projeto. Esta discrepância gerou uma expectativa que não foi alcançada pelo cliente com relação à completude do sistema.

Outro problema que ocorreu com certa frequência foi o constante reagendamento das reuniões de revisão da sprint. Os compromissos da área cliente, algumas vezes, postergaram as reuniões, afetando o planejamento da sprint. Este caso poderia ter sido menos prejudicial caso já existisse um backlog de defeitos identificados, preenchendo o tempo adicional em cada sprint.

O uso da ferramenta Rational Team Concert (RTC) se mostrou difícil para a área cliente, devido à sua interface de média complexidade. No início do projeto, tentou-se realizar a comunicação com o PO pelo RTC, mas a área cliente apresentou uma grande dificuldade na adaptação de seu uso. Após essa tentativa, mantivemos a visão do Backlog do produto em uma planilha online, o que tirou em certo ponto a visibilidade do artefato, por não estar em um sistema de acompanhamento no contexto do projeto. O RTC continuou a ser usado pela equipe para cadastramento das sprints, das histórias em andamento e versionamento do código. É importante ressaltar que não houve um treinamento formal para uso da ferramenta, o que entendemos ser fundamental para uma boa implantação e funcionamento.

Em uma visão organizacional, os maiores desafios para a aplicação deste modelo no contexto público tendem a ser:

- **Envolvimento do Cliente:** o gerente funcional deve alocar seu recurso de negócio ao time de projeto desde o seu início e dar alçada necessária para a tomada rápida de decisão para assuntos como priorização e validação de requisitos por exemplo.
- **Processo da Contratada de Desenvolvimento de Software:** é necessário que não só a contratada esteja preparada para essa nova forma de gestão como também é importante mudar a cultura vigente de focar a remuneração dos serviços pela produção de documentos e artefatos intermediários para software entregue e funcionando, algo palpável e concreto que realmente agregue valor ao Cliente.
- **Escopo Flexível:** contratos que permitem uma flexibilidade no escopo a ser produzido são mais alinhados aos processos ágeis por permitirem que o risco da mudança seja mitigado para a contratada e contratante. Este tipo de contrato ainda não é usual e de difícil assimilação para os setores de compra e orçamento.
- **Quantificação do Valor de Negócio:** a definição de um critério claro de priorização de projetos e requisitos é fundamental para o sucesso desta metodologia. Ao contrário da iniciativa privada onde esta mensuração é facilmente derivada de variáveis de ordem financeira, na área pública o valor de negócio está normalmente associado à percepção de valor para o cidadão como aumento de eficiência e desburocratização. O desafio está em definir critérios claros de priorização envolvendo tanto aspectos técnicos como políticos.

Um ponto ainda a ser evoluído em futuras experiências utilizando o modelo ágil é a busca de uma maior relação de confiança entre contratado-contratante nas organizações públicas. Temas como escopo flexível, remuneração pela entrega de software, níveis de serviço alinhados com o PO, critérios de colaboradores com maior foco em generalistas em contrapartida dos especialistas do modelo de fábrica de software, critérios de formação de time com ênfase na experiência e atitude comportamental do que propriamente na quantidade de recursos alocados. São todos aspectos que podem e devem ser trabalhados em um Termo de Referência para contratação de desenvolvimento de software a fim de evitar o modelo fabril clássico e ao mesmo tempo incentivar o mercado a adotar soluções que permitam a execução ágil dentro dos limites impostos pela legislação de contratação pública.

Um ponto determinante para o sucesso deste tipo de contratação é que seja projetado um modelo de contratação baseado em evidências de entrega de software e valor ao longo do ciclo de desenvolvimento. A metrificação atual é um elemento que normalmente pauta o relacionamento financeiro entre contratantes e contratadas. O problema disto está no seu mal uso e na complexidade de sua aplicação em cenários não-tradicionais de desenvolvimento. A ausência de aplicação de elementos como produtividade e a aplicação de metodologias baseadas no modelo de ciclo de vida em cascata gera um modelo administrativo de fiscalização do contrato em que se privilegia a entrega de papel e não de software como eventos de pagamento, pois é necessário o controle periódico e parcial do progresso técnico dos atores envolvidos.

Outra discussão importante é que pessoas ainda são elementos cruciais no sucesso de projetos de software. Processos e ferramentas ajudam, mas descaracterizar o fator humano ainda não é possível, tendo em vista a natureza criativa das soluções de tecno-

logia. É importante ressaltar aqui que, o perfil profissional proativo dos desenvolvedores foi um fator de sucesso para o estudo de caso.

A abordagem experimentada neste estudo foi bem sucedida pois contou com a participação da área de TI da Ancine como canal de comunicação com a Área Cliente e também responsável/dona do produto. Evidentemente essa estrutura foi funcional em função do forte conhecimento técnico dos servidores da área de TI. Em outro cenário de organização pública com servidores com menor domínio técnico e maior foco administrativo-burocrático, talvez a experiência não fosse exitosa.

## 5 Conclusão

A reforma administrativa do governo em 1995, incentivou um aumento da terceirização no setor público, causando uma diminuição de servidores especializados na área de TI e uma maior dependência dos órgãos públicos frente a profissionais terceirizados. Este movimento dificultou a assimilação e customização de processos de desenvolvimento de software assim como um maior controle de qualidade dos serviços prestados. Também ocorreu um crescente desalinhamento do planejamento de tecnologia com as necessidades da organização, gerando a impressão de que a área de TI era apenas uma grande “central de chamados” da organização.

Os projetos de desenvolvimento de software hoje na Ancine seguem um modelo de contratação baseado em Ponto de Função (APF) e são desenvolvidos por uma única empresa terceirizada. Seu processo é extremamente faseado com entrega obrigatória de artefatos a cada etapa. A flexibilidade para customização do processo é muito baixa e apresentação de resultados tangíveis para o cliente (software funcionando) demora alguns meses. As solicitações de mudança são recorrentes e em geral geram grande retrabalho nos artefatos já produzidos, aumento do custo do projeto e alongamento dos prazos pactuados.

A fim de entregar valor mais freqüentemente para o cliente em forma de software funcionando e ao mesmo tempo permitir uma maior flexibilidade de requisitos em função do contexto dinâmico de mercado que a Ancine regula e fomenta, uma saída poderia ser a incorporação de métodos ágeis como parte de um portfólio de processos que seria então customizado e instanciado para atender a necessidade de cada projeto.

Um estudo de caso foi desenvolvido a fim de verificar e provar a aplicabilidade dos métodos ágeis no contexto da organização. Como visto neste trabalho, o resultado em geral foi amplamente positivo, não somente nas fileiras técnicas como também junto aos executivos, gerando a expectativa de continuidade e implantação deste novo processo em futuros contratos.

Uma ressalva importante dentro do sucesso deste estudo foi a presença de um risco sempre alto da perda de definição do escopo da primeira versão do produto a ser entregue. Em função da grande flexibilidade para mudanças e o amadurecimento da visão do negócio por justamente ter o software logo em mãos para testar, o sistema nunca se demonstrava “redondo” para ser implantado apesar dos sucessivos refinamentos. A cultura do processo cascata ainda entranhada no cliente de definir de uma vez todos os detalhes e regras de negócio dificultou o entendimento de que a estratégia mais eficiente seria implantar algo mais simples mesmo que incompleto no curto prazo para depois haver as necessárias evoluções de produto conforme o feedback e uso.

Nossas expectativas, a partir desta experiência, é que tanto os processos como os modelos de contratação de desenvolvimento de software, inclusive em órgãos públi-

cos, possam evoluir em sincronia para um modelo de escopo mais flexível e colaborativo entre as partes sem contudo comprometer o controle e a transparência dos serviços prestados, e que o resultado final – software - seja de fato determinante e valorizado.

Um desafio para futuros modelos de contratação é permitir que o trabalho da contratada em função da flexibilidade de escopo e dos refinamentos das funcionalidades possa ser justamente remunerado em função do serviço prestado com a qualidade demandada e não necessariamente pelo tamanho funcional da alteração. A remuneração apenas pelas horas trabalhadas também não é uma opção por ser um convite ao mal uso. Um novo modelo deve incentivar o valor entregue para o cliente e não somente a quantidade de linhas de código ou número de funções programadas.

Do ponto de vista da Governança, os processos ágeis permitem e até incentivam uma maior aproximação entre TI e área de negócio, alimentando uma cultura de entrega de valor que necessariamente tem como referência os objetivos estratégicos da organização. A priorização das funcionalidades e definição das entregas torna-se mais complexa por sua natural flexibilidade exigindo um maior poder de governança interna em relação às decisões e avaliações.

Finalmente, processos ágeis não constituem “bala de prata” nem tem em si a capacidade de resolver todos os problemas e gargalos de tecnologia presentes nos diversos órgãos públicos. Aliando a estratégia de planejamento suficiente com uma execução eficaz permite que as políticas públicas possam ser implementadas mais rapidamente com uma melhor qualidade, viabilizando o software e os serviços por ele oferecidos como ferramentas imprescindíveis de cidadania para os tempos atuais.

## Referências

Estudo de Caso: Implantação de práticas ágeis no contexto de desenvolvimento de sistemas corporativos. CCE PUC/RJ, Otavio Ritter, 2006

GABARDO, Marco A.; GOMES, Alexandre R. **Discussão sobre Motivação de Equipes na Implementação de Métodos Ágeis no Desenvolvimento de Sistemas na Administração Pública Federal**. 2009. Universidade Católica de Brasília, Fundação Universa, Brasília, 2009.

GERENCIANDO PROJETOS DE TI NO CONTEXTO PÚBLICO – Apresentação no Congresso do PMI-RJ/2011, Otavio Ritter e Sergio Jesus, 2011

Governança de TI: transformando a Administração Pública no Brasil / organizadores: Marco Cepik e Diego Rafael Canabarro. – Porto Alegre : WS Editor, 2010

INFORME especial: Superintendência de Fomento. Boletim Ancine. 9, Set. 2011. Disponível em: <http://boletim.ancine.gov.br/especial001/>>. Acesso em: 30 jan. 2012.

MELO, Cláudia de O; Ferreira, Gisele R. M. AgilCoop: Adoção de métodos ágeis em uma Instituição Pública de grande porte - um estudo de caso. Agile Brasil, Jun. 2010. Disponível em: [http://www.agilcoop.org.br/files/WBMA\\_Melo\\_e\\_Ferreira.pdf](http://www.agilcoop.org.br/files/WBMA_Melo_e_Ferreira.pdf)>. Acesso em: 27 jan. 2012.

PLAY Framework. Disponível em: <http://www.playframework.org>>. Acesso em: 30 jan. 2012.

RELATÓRIO DE LEVANTAMENTO Governança de tecnologia da informação na Administração Pública Federal (APF) - Secretaria de Fiscalização de Tecnologia da Informação – Sefti/TCU, 2010

SCHWABER, K. Agile Project Management with Scrum. Washington: Microsoft Press, 2004.