

Evolução do Portal do Software Público Brasileiro:

**pesquisa e desenvolvimento para uma nova geração
integrada de plataformas abertas e colaborativas**

Universidade de Brasília

CDT

Centro de Apoio ao Desenvolvimento Tecnológico

LAPPIS

Laboratório Avançado de Produção, Pesquisa e Inovação em Software

Faculdade UnB Gama

MídiaLab

Laboratório de Pesquisa em Arte Computacional

Instituto de Artes

Plano de Trabalho

Outubro de 2015

Resumo

Este documento apresenta o plano de trabalho para o projeto de pesquisa “Evolução do Portal do Software Público Brasileiro: pesquisa e desenvolvimento para uma nova geração integrada de plataformas abertas e colaborativas”, elaborado pela Universidade de Brasília (UnB), através dos professores do Laboratório Avançado de Produção, Pesquisa e Inovação em Software (LAPPIS), da Faculdade UnB Gama (FGA), e pelos professores do Laboratório de Pesquisa em Arte Computacional (MídiaLab), do Instituto de Artes (IdA) da UnB.

Dados do Convenente

Entidade Proponente				CNPJ	
Fundação Universidade de Brasília – FUB				00.038.174/001-43	
Endereço					
Campus Universitário Darcy Ribeiro Gleba A – Faculdade de Tecnologia					
Cidade	UF	CEP	DDD/Telefone	E-mail	
Brasília	DF	70070-946	(61) 3107-5596	psuarez@cdt.unb.br	
*Conta Corrente	Banco		Agência	Praça de Pagamento	
Nome do Responsável				CPF	
Paulo Anselmo Ziani Suarez				604.476.990-91	
CI/Órgão Exp.	Cargo	Função		Matrícula	
05301817 CRQ-RS	Professor	Diretor do CDT		991384	
Endereço				CEP	
Campus Universitário Darcy Ribeiro – Gleba A – Ed.CDT – Asa Norte				70.910-900	
Esfera Administrativa					
Federal					

Dados Cadastrais do Ministério do Planejamento

Órgão/Entidade Concedente Ministério do Planejamento, Orçamento e Gestão – MP				CNPJ 00.489.828-0003-17	
Endereço Esplanada dos Ministérios - Bloco C - Térreo, sala T-08					
Cidade Brasília	UF DF	CEP 70070-900	DDD/Telefone (61) 2020-1390	E-mail @planejamento.gov.br	
Conta Corrente	Banco	Agência		Praça de Pagamento	
Nome do Responsável Loreni Fracasso Foresti				CPF 264.939.500-15	
Endereço SQS 103 – Bloco J – Apto 63 – Asa Sul					
Cidade Brasília	UF DF	CEP 70243-100	DDD/Telefone (61) 2020-1400/1434	E-mail loreni.foresti@planejamento.gov.br	
CI/Órgão Exp. 6.002.998.026/SSP-RS	Cargo Secretária	Função DAS 101.6	Matrícula 138.028-6		

1. Introdução

Software expressa uma solução abstrata dos problemas computacionais. O software, em um sistema computacional, é o componente que contém o conhecimento relacionado aos problemas a que a computação se aplica. Por isso, o software é algo de interesse geral, uma vez que vários aspectos relacionados a ele ultrapassam as questões técnicas, como por exemplo: o processo de desenvolvimento do software; os mecanismos econômicos que regem esse desenvolvimento e seu uso; o relacionamento entre desenvolvedores, fornecedores e usuários do software; os aspectos éticos e legais relacionados ao software [Kon et al., 2011]. O que define e diferencia o software livre do que podemos denominar de software restrito passa pelo entendimento desses quatro pontos dentro do que é conhecido como o ecossistema do software livre [Meirelles, 2013].

O princípio básico desse ecossistema é promover a liberdade do usuário, sem discriminar quem tem permissão para usar um software e seus limites de uso, baseado na colaboração e num processo de desenvolvimento aberto [Kon et al., 2011]. Software livre é aquele que permite aos usuários usá-lo, estudá-lo, modificá-lo e redistribuí-lo, em geral, sem restrições para tal e prevenindo que não sejam impostas restrições aos futuros usuários. Normalmente, esse software existe por meio de projetos de desenvolvimento que estão centradas em torno de algum código-fonte acessível ao público, geralmente em um repositório na Internet, onde desenvolvedores e usuários podem interagir [Meirelles, 2013]. O código é necessariamente licenciado sob termos legais formais que estão de acordo com as definições da Free Software Foundation ¹ ou da Open Source Initiative ².

No contexto do Governo Federal Brasileiro para se entender o software livre e como usá-lo de forma mais apropriada dentro do órgão públicos, em pesquisa realizada pelo Ministério de Ciência, Tecnologia e Inovação, em 2005, entre usuários, desenvolvedores e empresas, figuravam entre os fatores que motivam usuários e desenvolvedores a adotar o Software Livre: redução de custos, maior flexibilidade para adaptação, maior qualidade, maior autonomia do fornecedor e maior segurança. Entendendo essa lógica e com a finalidade de adaptar, bem como fornecer uma segurança jurídica mais robusta a legislação de nosso País, o Governo Federal criou em 2005 o modelo do Software Público Brasileiro (SPB), que entre os usuários estão ofertantes e demandantes de soluções, organi-

¹gnu.org/philosophy/free-sw.html

²opensource.org/docs/definition.html

zados em comunidades, criadas em torno de cada solução de software. A intensidade de participação varia desde um observador interessado no software até líderes de comunidades. Essa diversidade é derivada do modelo de produção do software livre, no qual baseou-se o SPB para sua formação. A percepção do potencial que representava a participação da sociedade no desenvolvimento do software e o conceito de bem público foram adaptadas do ponto de vista jurídico, assim levando o Ministério do Planejamento, Orçamento e Gestão (MP) a formular o conceito de software público. Essa base jurídico-institucional permitiu a criação de um ambiente virtual (um portal) para a disponibilização de software como software público. Esse modelo é definido por uma rede que se auto-organiza e cuja produção se caracteriza pela intensa participação colaborativa entre indivíduos, empresa ou prestadores de serviço, universidades e instituições interessadas na evolução de um determinado projeto de software.

O conceito de software público diferencia-se do de software livre em alguns aspectos, destacando-se a atribuição de bem público ao software. Embora haja algumas diferenças entre o que é um software livre e um software público brasileiro, há princípios comuns, como a tendência da descentralização na tomada de decisões, o intenso compartilhamento de informações e os processos de retroalimentação decorrentes do uso dos artefatos produzidos. Em outras palavras, todo software público é um software livre também.

Nesse cenário, para potencializar com que mais projetos de software livre, em particular aqueles desenvolvidos no Brasil, sejam também um software público, bem como as instituições e empresas melhor trabalhem com esse modelo de desenvolvimento e negócio ao não tratar o software em si como um produto, mas um meio para geração de serviços, também no ecossistema do Software Público Brasileiro, é importante se trabalhar em rede e prover um ambiente para tal. Atualmente, quando falamos em Portais e Redes Sociais é comum nos reportarmos àquelas chamadas Redes Sociais Digitais em que a comunicação entre seus membros se dá por meio de computadores ligados a Internet ou por dispositivos móveis. Mas, o conceito de Redes ou Comunidades Sociais é bem anterior ao advento da Internet. De forma genérica, o termo rede pode ser entendido como canais de comunicação entrelaçados e ligados por pontos de conexão em comum pelos quais transitam elementos que irão se diferenciar em função do tipo de rede (de amigos, de trabalho, acadêmico, religioso, sexual etc) a que se referirem. Por analogia, é possível interpretarmos o conceito de redes sociais como sendo grupos de pessoas que se conectam utilizando um mesmo canal no qual compartilham interesses

em comum. Por se fundamentarem em relações e interações sociais, tais redes apresentam-se como fenômenos dinâmicos que representam a estrutura e a organização de um certo grupo social em um dado momento sócio-histórico cultural.

Desta forma, tradicionalmente, a possibilidade de formação de redes ou comunidades sociais relaciona-se com a construção de laços baseados em relações tais como a de parentesco, vizinhança, afiliações, proximidade física ou mesmo espiritual. Quanto mais fortes se constituem esses laços, mais estáveis e duradouras se tornam as interações entre os atores que compõem determinada rede social. Vale salientar que para um grupo se constituir como comunidade não basta viver em um mesmo território ou espaço geográfico, pois “o que une uma comunidade não é sua estrutura, mas um estado de espírito, um sentimento de comunidade” [Venturelli, 2004]. Pensar em redes sociais virtuais é também pensar em questões complexas como globalização e construção de identidades individuais e coletivas. Assim, na sociedade digital, se a informática associada às telecomunicações globalizou questões políticas, econômicas e culturais, ela também vem provocando mudanças profundas nas relações sociais construídas com base nessa mediação realizada pelo computador. Com o advento da Internet multiplicaram-se, de forma incessante, as possibilidades de criação, concepção, estruturação, organização e desenvolvimento das redes sociais na web. Por meio de ferramentas de Comunicação Mediada pelo Computador (CMC), tornaram-se possíveis as interações em escala mundial, nas quais as barreiras de tempo e espaço são rompidas. Para a autora as redes sociais funcionam na internet com softwares sociais.

O conceito de software social como uma ferramenta de análise dos novos processos de mobilização e organização social. Ao longo da história da comunicação nunca se conseguiu criar ferramentas que suportassem a discussão simultânea e instantânea entre vários atores. Para ele as ferramentas anteriores, tanto as difusoras, ou seja, um ponto irradiando para muitos outros, como a TV, quanto às pessoais que estabelece diálogo entre dois pontos tinham limitações e eram pouco complexas quanto às intervenções de seus membros. Nesse sentido, quanto maior o grupo, maior a dificuldade de sustentar uma relação dialógica dentro dele. Hoje, as ferramentas sociais da internet suportam, de forma nativa, um diálogo complexo. Constata-se hoje que as estruturas colaborativas de rede comportam, desde a sua gênese, exigências de participação socialmente complexa.

A comunicação em rede transcende fronteiras, a sociedade em rede é global, é baseada em redes globais. Então, a sua lógica chega a países de todo o planeta e difunde-se através do poder integrado

nas redes globais de capital, bens, serviços, comunicação, informação, ciência e tecnologia. Aquilo a que chamamos globalização é outra maneira de nos referirmos à sociedade em rede. Um importante efeito dessa nova forma de comunicação foi a facilidade trazida para o encontro de pessoas dentro desse ambiente no ciberespaço e o conseqüente desenvolvimento de um fenômeno contemporâneo conhecido como Comunidades Virtuais. Apesar de já existir, mesmo que de forma precária e menos abrangente, nos anos 70 e 80 do século passado, o termo Comunidades Virtuais foi difundido a partir da década de 90 quando Rheingold (1996) o definiu como um tipo de ambiente comunicacional onde se formam agregações humanas globalmente conectadas online (Internet) nas quais se estabelecem redes de relações sociais em que são compartilhados valores, interesses, objetivos e sentimentos em comum.

O mesmo termo é assim definido por [Venturelli, 2004], as comunidades virtuais são agrupamentos humanos baseados não mais na vizinhança real, mas nas vizinhanças virtuais, metafóricas e simulatórias, sobre afinidades de interesses. De um modo geral, as comunidades virtuais caracterizam-se como espaços a serem habitados e construídos coletivamente, onde indivíduos de diferentes lugares podem se tornar cidadãos e interagir com o tempo, o espaço e os mais variados desejos. Adotaremos a definição de Comunidades Virtuais como aquela em que estas se apresentam como um espaço construído coletivamente onde são formados agrupamentos humanos em um ambiente online no qual, por meio de interações mediadas pelo computador e/ou por dispositivos moveis conectadas a Internet, são compartilhados sentimentos, valores, crenças, interesses e objetivos comuns.

Acreditamos ser importante esclarecer que os termos Rede Social na Internet ou Comunidades Virtuais terão a mesma definição apresentada acima por se tratarem, em sua essência, da mesma coisa. Entretanto, vale ressaltar que não devemos confundi-los com o termo site de redes sociais. Sites de redes sociais são as plataformas, sistemas ou suportes informáticos que permitem que as pessoas, ao utilizá-los, criem as suas próprias Redes Sociais ou Comunidades Virtuais. Deste modo, o que vai tornar uma rede ser um Portal e rede social da forma que a definimos serão as interações e as intervenções realizadas neste projeto de pesquisa de inovação tecnológica que visa contribuir com a Evolução do Portal do Software Público Brasileiro: pesquisa e desenvolvimento para uma nova geração integrada de plataformas abertas e colaborativas.

Este projeto visa desenvolver pesquisas para a Nova Geração do Portal do Software Público Brasileiro (Portal SPB) que tenha como principal característica a cooperação e o compartilhamento

de conhecimento em comunidades na ampliação de parcerias e reforço da política de software livre no setor público.

O Portal SPB objetiva estabelecer as condições de cooperação que regularão os compromissos entre os partícipes, visando o uso e desenvolvimento de soluções públicas de tecnologia da informação (TI) e do desenvolvimento de práticas de governança e gestão de tecnologia da informação, mediante a implementação conjunta de ações, programas, projetos e atividades de interesses comuns entre os partícipes.

A Nova Geração do Portal do SPB será um ecossistema de colaboração e compartilhamento para o desenvolvimento de software baseado em plataforma web. Esta plataforma integrará ferramentas de software de suporte ao desenvolvimento aberto, colaborativo, compartilhado e seguro de software. Os participantes do ecossistema poderão ser desenvolvedores, usuários ou prestadores de serviços, permitindo estreito relacionamento entre ofertantes e demandantes, de maneira própria aos ambientes abertos de inovação.

O ecossistema do Software Público Brasileira, com a evolução prevista neste plano de trabalho, contemplará as seguintes macro funcionalidades:

Visão do portfólio de software, para permitir que um membro navegue para encontrar projetos de seu interesse; Visão do software, para fornecer as ferramentas de colaboração que permitam a um membro participar em um projeto específico; Visão da gestão social dos projetos (ou das soluções), para proporcionar a gestão de projetos de colaboração organizados na forma de redes sociais, facilitando a ampla e contínua comunicação aberta entre os membros do ecossistema; Visão da inteligência coletiva baseada na geração, reunião e compartilhamento de conhecimento resultantes dos processos de identificação e resolução de problemas realizados pelas comunidades suportadas pelo ecossistema; Visão dos mecanismos de avaliação das vulnerabilidades e da qualidade das soluções disponibilizadas no ecossistema, para permitir que os membros do ecossistema tenham conhecimento segurança de sistemas de informação e promovam soluções mais seguras e confiáveis.

Nesse ecossistema proposto, as comunidades e núcleos de desenvolvimento trabalham em colaboração aberta, o que facilita identificar um problema ou encontrar um projeto, ler sobre ele, entendê-lo, contribuir como voluntário, apropriar do conhecimento e participar da solução.

2. Apresentação do Convenente

Para a realização do projeto, a Secretaria de Logística e Tecnologia da Informação (SLTI) optou pela cooperação com duas unidades acadêmicas da Fundação Universidade de Brasília (FUB), com o apoio do Centro de Desenvolvimento Tecnológico-CDT, de modo a utilizar pesquisas aplicadas e transferência de tecnologias e de conhecimento em temas avançados relativos ao projeto: MidiaLab (do Instituto de Artes) e o LAPPIS (da Faculdade do Gama).

O MídiaLab Laboratório de Pesquisa em Arte Computacional da Universidade de Brasília (UnB) foi criado em 1986 e possui linhas de pesquisas realizadas em diversas áreas como sistemas de informação, computação móvel, tecnologias da Internet e outras. Os projetos do Laboratório já receberam prêmios nacionais e internacionais e contam com a participação de bolsistas de Iniciação Científica, estagiários e estudantes da graduação e do programa de pós-graduação em Arte e computação, linha de pesquisa em arte e tecnologia, que trabalham em diferentes propostas, envolvendo a criação de animação, vídeos, arte computacional, dispositivos não convencionais de interação, ciberintervenções urbanas, realidade aumentada urbana (RUA), entre outros. Os projetos envolvem questões socio-artísticas e políticas no contexto da arte, ciência e tecnologia realizados em estreita colaboração com outras áreas de pesquisa como a ciência da computação, mecatrônica e comunicação, para propor projetos inovadores, artísticos e tecnologicamente interessantes para se pensar a sociedade hoje, buscando deste modo concretizar as sinergias existentes entre o tecido criativo, o ensino superior e a investigação realizada na Universidade de Brasília, proporcionando um contexto de formação avançada e promovendo a criação de projetos economicamente sustentáveis.

O Laboratório Avançado de Produção, Pesquisa & Inovação em Software (LAPPIS), localizado na Faculdade UnB Gama (FGA) foi criado em 2012 e foi concebido para atuar em áreas tecnológicas desde sistemas de informação até os sistemas embarcados, em especial, objetivando as oportunidades de pesquisas teóricas e aplicadas. Em que pese sua recente criação, o laboratório, possui cooperações em andamento e concluídas com os segmentos da Administração Pública e Iniciativa Privada. Cabe ressaltar projetos já realizados com a Presidência República em parceria com o Serpro, Positivo Informática S/A, além da colaboração na evolução de diversas ferramentas de software livre. O LAPPIS tem como objetivo contribuir com o desenvolvimento de projetos de software ao passo que complementamos a formação de Engenheiros de Software (capazes de lidar com problemas ao pensar

em soluções computacionais e implantá-las efetivamente), por meio de Métodos Ágeis, Software Livre, Segurança e Trabalho colaborativo centrado nas pessoas. Com isso, oferece aos alunos de graduação em Engenharia de Software a oportunidade de aplicar os conceitos e as tecnologias em um ambiente produção de software real, sob orientação de professores especialistas nas áreas dos projetos do laboratório. Além da sua atuação na Faculdade UnB Gama, o LAPPIS tem uma parceria de colaboração com o Centro de Difusão de Tecnologia e Conhecimento, que é um projeto iniciado em Agosto de 2004, com a proposta de união de esforços entre o setor público e as universidades que fazem parte do estado com objetivo de ampliar o conhecimento da sociedade no uso do software livre. O CDTC é um projeto de intenso impacto social, ampliando as liberdades individuais com o acesso da tecnologia pela sociedade, tendo ainda o estado uma grande economia de recursos a partir do usos de licenciamentos livres, permitindo assim que a economia com os recursos despendidos anualmente em licenças proprietárias de softwares, garanta o aquecimento de um mercado emergente, permitindo o acesso e apropriação tecnológica pelo próprio mercado nacional. Com o CDTC como parceiro, além do laboratório fisicamente localizado na UnB-Gama, o LAPPIS conta com um espaço físico, com excelente estrutura para as reuniões e desenvolvimento de projetos, no Campus da UnB Darcy Ribeiro.

O Centro de Apoio ao Desenvolvimento Tecnológico da Universidade de Brasília – CDT/UnB – criado em 1986, é uma unidade gestora responsável pela transferência de tecnologia, prestação de serviços especializados e interação da Universidade com empresários, empreendedores, governo e sociedade em geral. Os programas, produtos e serviços do Centro apoiam a criação de novos negócios ou desenvolvimento de projetos de pesquisa, estimulando o potencial empreendedor e desenvolvendo parcerias estratégicas. O CDT/UnB possui autonomia para negociar e gerir contratos, acordos e convênios dentro de sua área de atuação. Portanto tem agilidade e flexibilidade para executar projetos com eficiência e menor custo, sem necessidade de utilizar formas alternativas, como fundações de apoio. O CDT/UnB atua a partir dos seguintes princípios:

- Negócio: Educação e pesquisa para o empreendedorismo, inovação e transferência de tecnologia.
- Visão: Ser o centro de excelência no apoio a gestão da inovação tecnológica, transferência de tecnologia e estímulo ao empreendedorismo.
- Missão: Apoiar e promover o desenvolvimento tecnológico, a inovação e o empreendedorismo, em âmbito nacional, por meio da integração entre a universidade, as empresas e a sociedade

em geral, contribuindo para o crescimento econômico e social.

A equipe do Centro é multidisciplinar, composta por mais de 70 profissionais envolvendo doutores, mestres, especialistas, graduados e de nível médio, com formação em diversas áreas do conhecimento. O CDT/UnB é também responsável pela proteção do conhecimento gerado na Universidade e sua transferência para o mercado, seja na forma de licenciamentos de ativos protegidos, seja na geração de projetos cooperativos de P&D ou ainda por meio de consultorias e serviços tecnológicos. O CDT já foi contemplado com os seguintes prêmios:

- 2000, 2006, 2007, 2009, 2010 e 2011- Prêmio Finep de Inovação Tecnológica
- 2010 - Prêmio Sinfor de Tecnologia da Informação
- 2003 - 4º Prêmio Excelência em Tecnologia Sebrae
- 2000 - Prêmio IEL de Interação Universidade-Indústria-Empresas Juniores
- 1999 - Prêmio Incubadora do Ano 99

3. Justificativa

O Portal do Software Público Brasileiro - SPB, inaugurado em 2007, na prática, é um sistema web que se consolidou como um ambiente de compartilhamento de projetos de software. Oferece um espaço (comunidade) para cada software. A comunidade é composta por fórum, notícias, chat, armazenamento de arquivos e downloads, wiki, lista de prestadores de serviços, usuários, coordenadores, entre outros recursos. Teve um crescimento expressivo contando, hoje, com mais de 60 comunidades de desenvolvimento e mais de 200.000 usuários cadastrados. O SPB abrange também, o 4CMBR que é o grupo de interesse voltado para soluções de tecnologia para municípios, o 5CQualiBr que é um grupo que trabalha para evoluir a qualidade do Software Público Brasileiro, o 4CTecBr, um portal destinado a colaboração no desenvolvimento de Tecnologias Livres, o Mercado Público Virtual que é um grupo de empresas e pessoas que prestam serviço nos softwares ofertados no Portal e o AvaliaSPB que avalia a entrada dos softwares candidatos a software público. O ambiente do SPB não proporciona a integração com ambientes colaborativos externos, especialmente com redes sociais. A plataforma escolhida na ocasião da criação foi o framework OpenACS, que continua sendo utilizada na atual versão.

Entretanto, a evolução do SPB foi comprometida desde 2009, quando framework OpenACS foi descontinuado. Com isso, não tendo versões lançadas a partir daquele ano. Por isso, hoje, é necessária a evolução para novas tecnologias, que tenham maior suporte das comunidades de desenvolvimento, utilize linguagens de programação com maior velocidade de desenvolvimento e permita a integração com ambientes colaborativos externos, em especial, redes sociais. Além disso, é preciso realizar a manutenção evolutiva das funcionalidades existentes e também o desenvolvimento de novas funcionalidades para o Portal do SPB .

Entre as consequências desse cenário, a equipe de analistas do SPB, alocada na Secretaria de Logística e Tecnologia da Informação (SLTI) do Ministério do Planejamento, Orçamento e Gestão, ocupa boa parte dos seus esforços com a manutenção corretiva para prover o mínimo de estabilidade do ambiente atual, que se encontra bastante vulnerável. Dessa forma, uma equipe de analistas que deveria estar pensando nas estratégias de uso e como melhor aproveitar tal ecossistema tecnológico, não consegue também colaborar com a evolução do portal e do modelo do SPB em si.

Além da falta de evolução do framework OpenACS e do Portal como um todo, as demais partes,

que compõem o SPB, como por exemplo o “Avalia SPB” e o “Mercado Público” são separadas, sem autenticação integrada, sem uma identidade visual e interface com o usuários uniforme, bem como não sendo adequada para os dispositivos móveis. Também, o sistema de controle de versão de código-fonte do software disponível para as comunidades dos projetos hospedados no SPB é de uma geração anterior às usadas pelos principais projetos de código aberto no momento. Da mesma forma, o sistema de acompanhamento de tarefas, funcionalidades e defeitos não contempla todos os recursos disponíveis nas plataformas mais usadas pelos desenvolvedores e líderes de projetos de software livre, atualmente.

Entre outros pontos, a falta de uma categorização do compartilhamento de código e artefatos dos projetos faz com que as informações não sejam facilmente encontradas. Sem um sistema de indexação de conteúdos para uma busca mais inteligente, os gestores públicos têm dificuldades para encontrar um software público que atenda sua necessidade, conforme é normatizado pela IN04/2010, para se justificar a compra de um software de licença restritiva (software proprietário).

No contexto do Governo Federal, a Secretaria de Logística e Tecnologia da Informação (SLTI) normatiza, desenvolve e fomenta políticas públicas na área de tecnologia da informação, através do Decreto nº 7579, que trata da regulamentação do Sistema de Administração dos Recursos de Tecnologia da Informação (SISP), constitui-se uma importante atribuição da SLTI manter em operação o Portal do Software Público Brasileiro (PSPB). Neste momento, o Portal SPB precisa ter sua dinâmica ampliada, bem como reformulada, para se alinhar com as demandas de um novo modelo de funcionamento que contemple mais ainda a participação social e colaborativa (governos, empresas e indivíduos). Dessa forma, este Plano de Trabalho foi concebido como premissa para atender a iniciativa número 6 do Planejamento estratégico da SLTI de 2012 - Ampliar a oferta e utilização de Software Público e no indicador SLTI 08 – Softwares Públicos Evoluídos do PDTI 2012-2013 do Ministério do Planejamento, Orçamento e Gestão e tem alinhamento com as demandas de reformulação do modelo de funcionamento para a Nova Geração do Portal mais voltado para o uso dentro da própria Administração Pública Federal (APF).

Um dos passos para a concretização de uma nova geração do Portal SPB é a integração de novas tecnologias, desde uma plataforma colaborativa até sistemas de controle de versão e monitoramento da qualidade do código-fonte, gerenciadas e apresentadas em uma plataforma integrada no back-end e, em especial, no front-end para que os usuários e as comunidades dos projetos tenham um conjunto

de recursos para encontrarem os projetos, bem colaborarem em torno de um software público. Nesse contexto, foi proposto o presente Plano de Trabalho que embasa o correspondente termo de cooperação com o Ministério do Planejamento, Orçamento e Gestão, tendo como ponto de partida que o desenvolvimento de um software livre, que norteia o desenvolvimento de um software público, possui características distintas do modelo de desenvolvimento de software restrito (privativo ou proprietário). Portanto, a relação com o mercado, o processo de desenvolvimento e o produto a ser oferecido devem ser abordados de maneiras distintas [Kon et al., 2011]. Devido às atividades de produção de código, documentação, relatos de erros entre outras, as comunidades de software livre e público vêm construindo coletivamente sistemas de software reconhecidamente de qualidade, em um ambiente de colaboração constante para atualização e evolução desses sistemas.

Além disso, uma vez que o SISP é a estrutura usada pelo Governo Federal para a organização do trabalho, controle, supervisão e coordenação da área de TI na administração direta, autárquica fundacional, o projeto apresentado neste plano de trabalho promoverá com que pesquisas baseadas no estado-da-arte da engenharia de software e das tecnologias colaborativas e abertas, possam ser aplicadas, bem como impactar diretamente uma tecnologia e serviço estratégico do Governo Federal, com potencial desdobramentos econômicos e sociais.

Por fim, mesmo com as limitações citadas, o Portal do Software Público Brasileiro teve em 2013 mais de 600 mil visitantes únicos, com mais de 1 milhão de visitas/acessos, gerando mais de 16 milhões de visitadas nas páginas, com um total de mais de 49 milhões de hits no Portal SPB. Avaliando apenas as comunidades dos projetos I3Geo, CAU, CACIC e Geplanes, houve mais de 15 mil downloads e 4 mil mensagens nos fóruns. Essa amostra estatística ilustra bem o potencial do Software Público Brasileiro, bem como as expectativas de seus usuários e colaboradores para a evolução do Portal e do modelo em si.

4. Objeto

4.1. Identificação

Evolução do Portal do Software Público Brasileiro: pesquisa e desenvolvimento para uma nova geração integrada de plataformas abertas e colaborativas.

4.2. Descrição

Desenvolver e operacionalizar, na Secretaria de Logística e Tecnologia da Informação – SLTI, a Nova Geração do Portal do Software Público Brasileiro (Portal SPB) que tenha como principal característica a cooperação e o compartilhamento de conhecimento em comunidades na ampliação de parcerias e reforço da política de software livre no setor público.

5. Metas

As metas são definidas a partir de uma visão modularizada dos objetivos gerais do projeto. Nesse contexto, ao afirmar-se que algo terá um estudo gerado, implica no fluxo completo de pesquisas do mesmo, que inclui também as pesquisas de técnicas de Interação Humano-Computador para projetar interfaces visando atingir metas de usabilidade do sistema e de experiência do usuário.

1. Gerar estudos sobre rede sociais e de desenvolvimento colaborativo para o Portal Software Público Brasileiro: Ambiente integrado com ferramentas de comunicação e desenvolvimento colaborativo, que irão compor a nova plataforma, de modo a evoluir o atual Portal do Software Público Brasileiro.
2. Gerar estudos para um ambiente de Ativos de Software e interação de instituições governamentais no Portal: Evolução da plataforma integrada de colaboração, contendo o cadastro de instituições governamentais que criam e/ou utilizam projetos de software, para serem relacionados com os Ativos de Software catalogados no ambiente.
3. Gerar estudos sobre ambientes para prover a colaboração das instituições de ensino em Projetos do Software Público Brasileiro: No ambiente da plataforma integrada de colaboração será criado um perfil de comunidades para servir de modelo para uso por parte das instituições de ensino.
4. Gerar estudos sobre repositórios de código-fonte integrado a um sistema de gerenciamento de projetos para os Projetos de Software Público Brasileiro e Ativos de Software: Ambiente denominado de "Versionamento" que evoluirá para ser integrado ao ambiente da plataforma integrada de colaboração.
5. Gerar estudos sobre ambientes de monitoramento de software e repositório do projeto: Integrar à plataforma uma ferramenta de análise de métricas de código-fonte dos repositórios dos projetos, para prover o acompanhamento da qualidade interna do produto de software como por exemplo coesão, acoplamento, complexidade ciclomáticas, complexidade estrutural, entre outras, colaborando no acompanhamento da qualidade e as decisões de projeto do ponto de vista do software.
6. Gerar estudos sobre mecanismos para o incentivo para uso do Portal: Evolução na plataforma de integração de um painel que contenha informações relacionadas à participação dos membros nos projetos (canais de comunicação e nos ambientes de desenvolvimento) e mecanismo de

- estímulo aos usuários e instituições para relatarem suas experiências de uso dos projetos.
7. Gerar estudos sobre interação do Portal com outros dispositivos: Evolução do núcleo da plataforma para a disponibilização de um serviço computacional que ofereça para outras aplicações e/ou aplicativos móveis um mecanismo para consumir dados disponibilizados pelo Portal.
 8. Gerar estudos para migração dos serviços e dos dados atuais do Portal do Software Público Brasileiro para a nova plataforma integrada de colaboração: Planejar a migração dos dados dos projetos, comunidades e usuários do Portal do Software Público Brasileiro, de acordo com os ambientes e serviços que serão disponibilizados, seguindo as prioridades que serão estabelecidas para manter a consistência dos dados, conforme as funcionalidades e recursos disponíveis durante a evolução para a nova plataforma. Em outras palavras, realizar estudos técnicos com vistas a criação de scripts automatizados para migração dos dados do atual portal. Ademais, quando não se aplicar uma automação, apontar aqueles dados que necessitarão de migração manual.
 9. Gerar estudos sobre a estrutura física e lógica para manter o Portal em funcionamento: Definição e implantação da infraestrutura necessária para disponibilização adequada dos serviços do Portal do Software Público Brasileiro, gerando as documentações necessárias para a manutenção da plataforma, do ponto de vista de administração de sistemas.
 10. Gerar estudos sobre sustentação da plataforma: Padronizar os ambientes de desenvolvimento, para o gerenciamento de lançamento de novas versões, via automação (da maior quantidade possível de processos operacionais) dos procedimentos de atualização e evolução de toda plataforma.

6. Metodologia de Desenvolvimento

A Engenharia de Software tem evoluído suas práticas e metodologias em busca de padrões que regem o desenvolvimento de software de qualidade dentro dos escopos, custos e prazos desejados. O modelo dito tradicional tem como característica um conjunto grande e detalhado de documentação que deve, supostamente, ser utilizada ao longo de todo o ciclo de desenvolvimento. Entretanto, percebendo que os objetivos principais do desenvolvimento de software não estavam sendo alcançados, alguns líderes da indústria e academia começaram a adotar métodos mais simples de trabalho que apresentaram melhores resultados em projetos de software. Em 2001, líderes que estavam desenvolvendo projetos fora dos padrões industriais se reuniram para trocar experiências e trabalhos. Este grupo se tornou a Aliança de Desenvolvimento Ágil e escreveram o Manifesto Ágil que apresenta os princípios e valores que o grupo considera ser determinante para o desenvolvimento de software. Neste sentido, os métodos de desenvolvimento ágeis de software são métodos que implementam os seguintes valores:

- Indivíduos e interações acima de processos e ferramentas;
- Software operante acima de documentações grandes e completas;
- Colaboração do cliente acima de negociações contratuais;
- Responder à mudanças acima de seguir a um planejamento.

Esse conjunto de valores não descartam a importância dos elementos citados à direita das sentenças, mas evidenciam que estes são menos importantes diante dos primeiros elementos citados. Em outras palavras, apesar da documentação ser importante, o foco principal deve estar na entrega de software de valor para o cliente e na interação e a consequente comunicação entre as pessoas. Além disso, os métodos ágeis exaltam a simplicidade, feedback contínuo e adaptação à mudanças que podem ser obtidos a partir de comunicação face à face, qualidade de código e entrega contínua de software.

Dada a oportunidade de adoção de métodos ágeis no desenvolvimento do presente trabalho, a metodologia será baseada em uma combinação das metodologias Scrum e Extreme Programming. Destacam que XP e Scrum complementam um ao outro bem, com o XP provendo suporte para aspectos mais técnicos enquanto o Scrum provê práticas e técnicas para gerenciamento, planejamento e acompanhamento. Assim, com base nas experiências destacadas em [Schwaber and Beedle, 2001] e

[Fitzgerald et al., 2006], na motivação de adoção de métodos ágeis no desenvolvimento de software moderno, serão apresentadas os métodos XP e Scrum, suas principais características e práticas que serão utilizadas no desenvolvimento do presente projeto.

6.1. eXtreme Programming - XP

O Extreme Programming - XP é uma metodologia de desenvolvimento de software com foco em agilidade de equipes e qualidade de projetos, apoiado em cinco valores: comunicação; feedback; simplicidade; coragem e respeito. O XP propõe, portanto, a utilização de um conjunto de práticas conhecidas para serem aplicadas disciplinadamente em busca do desenvolvimento de software de qualidade dentro dos prazos e custos determinados. A seguir será apresentado como o XP é composto e como as práticas auxiliam alcançar os objetivos do desenvolvimento ágil, assim como elas se complementam.

6.1.1. Ciclo de Desenvolvimento

Assim como no Scrum, o XP define ciclos de desenvolvimento para que o produto de software seja incrementado a partir das prioridades do cliente. Neste ponto, a metodologia pressupõe fases pequenas de desenvolvimento e comunicação com cliente para que as entregas de software sejam constantes e em menor espaço de tempo. Vale ressaltar que a técnica de histórias de usuário é também apoiada pelo XP para definição de requisitos.

Durante os ciclos de desenvolvimento o foco principal está na equipe e na qualidade do produto em desenvolvimento. Tal qualidade se traduz principalmente em simplicidade de projeto e features cobertas por testes automatizados que deve ser manutenível e extensível pela equipe técnica. Para tanto, faz-se necessário a constante troca de conhecimento técnico a respeito do produto e revisões de código, além de se manter padrões de codificação e zelo pela propriedade coletiva do código.

6.1.2. Planning Poker - Jogo do Planejamento

Dado que a metodologia é baseada em iterações, o planejamento de cada ciclo é dado por um conjunto selecionado de User Stories que serão implementados e incrementados no produto ao fim

da iteração. Esta seleção é realizada a partir de um “jogo de planejamento” que envolve tanto a equipe de desenvolvimento quanto o cliente. Assim, a escolha é acordada priorizando as User Stories que agregam maior valor para o cliente e considerando-se a capacidade produtiva e opinião técnica da equipe sobre as mesmas.

A equipe de desenvolvimento consegue mensurar o esforço necessário para desenvolvimento de uma História de Usuário a partir de uma medida relativa chamada Story Point. Esta medida é baseada na complexidade técnica, possíveis dificuldades e quantidade de trabalho necessário para implementação da funcionalidade. Essa medida é utilizada na estimativa de produtividade da equipe de desenvolvimento e deve ser levada em consideração no planejamento de novas iterações. Evidentemente, à medida que o software evolui e todos compreendem melhor os requisitos e tecnologias envolvidas, a capacidade produtiva da média de Story Points por iteração pode aumentar, assim como determinados fatores podem influenciar negativamente nesta métrica.

Para realizar a estimativa de esforço de cada História de Usuário, a equipe realiza um “jogo” em que todos os membros técnicos se reúnem e recebem um conjunto de cartas com números que serão utilizados para as estimativas, geralmente, possuindo os valores da sequência de Fibonacci como sugerido por [Cohn, 2005]. A equipe então define uma História de Usuário como referência, atribuindo-a a menor unidade representativa de pontos, sendo referência para estimativas de outras funcionalidades. O jogo segue com a escolha e leitura de uma História de Usuário, abrindo a oportunidade para que todos os membros escolham uma carta com o valor estimado para a mesma sem mostrar para os outros colegas. Após todos terem realizado a suas escolhas, todos apresentam as cartas e então são discutidos os porquês de possíveis pontuações extremas, tanto super quanto subestimadas. Em seguida, são realizados novos ciclos do jogo, repetindo-se os passos descritos, até que haja uma convergência para definir a pontuação da História de Usuário.

Dada a estimativa de cada requisito e a priorização dos mesmos por parte do cliente, é acordado quais são as próximas funcionalidades que serão desenvolvidas na próxima iteração, respeitando-se evidentemente as considerações e capacidade da equipe técnica.

6.1.3. Desenvolvimento

Tendo em visto as necessidades citadas, o XP enfatiza a prática de Programação em Pares que consiste em duas pessoas trabalhando juntas em um mesmo computador destinando ao mesmo tempo esforços para o desenvolvimento de determinada funcionalidade. A prática é realizada com um programador pilotando, ou seja, escrevendo o código, enquanto o outro o acompanha com ideias e revisões, onde após algum tempo os dois trocam de posição. Apesar do dobro de recursos está sendo utilizado nesta prática, à todo tempo o código desenvolvido está sendo revisado por duas pessoas diminuindo inconsistência com padrões e taxa de defeitos, além de evitar a personalização do código cujas consequências se traduzem em produtividade e qualidade ao longo do projeto. Por fim, vale ressaltar a troca de conhecimento técnico e de negócio neste processo que é reforçada com o rodízio de pares, ou seja, os pares são sempre modificados ao longo de todo o ciclo de desenvolvimento.

6.1.4. Testes

Dentre todas as práticas enfatizadas pelo XP, testes se destacam pois cumprem papel fundamental em vários níveis para que os princípios ágeis sejam praticados. Primeiramente, destacam-se os testes de aceitação que são construídos no verso da Histórias de Usuário, especificando condições de entregas de novas funcionalidades que são parâmetro para verificação do cliente.

Em outro nível do projeto, testes são realizados à todo momento com objetivo de exercitar as menores unidades do código desenvolvido. Os testes unitários são testes escritos pelos próprios programadores para cada unidade de software que o mesmo desenvolveu. Esses testes devem ser automatizados para que o programador possa ter rápido feedback a respeito das funcionalidades e mudanças implementadas e para que os mesmo venham a ser executados regularmente em todo o sistema quando, por exemplo, forem implementadas estratégias de integração contínua. Nesta estratégia, os módulos construídos do software são integrados freqüentemente e os respectivos testes são executados para verificar as unidades e se a integração foi realizada corretamente, facilitando a entrega constante de software.

6.1.5. Design

Como já mencionado, a qualidade do produto é fundamental segundo os princípios ágeis. A metodologia XP visa alcançar a qualidade interna do produto com o design simples. Isso é possível a partir da perspectiva de que a arquitetura do software surge a partir da construção de cada funcionalidade e não a partir de atividades específicas de design que visam a implementação de uma arquitetura baseada em extensões futuras e adaptável.

Entretanto, o XP encoraja a prática de refatoração que consiste na mudança da estrutura interna do código sem alterar seu comportamento externo. Esta prática é realizada sistematicamente com uma série de passos e técnicas que buscam construir, evolutivamente, a extensibilidade e manutibilidade da arquitetura. A refatoração se torna possível à medida que existem testes unitários exercitando o comportamento externo do software, nos quais o programador pode realizar as modificações necessárias e verificar se as modificações mantiveram o comportamento esperado.

Visando ainda apoiar a simplicidade do código, qualidade interna e testes, o XP recomenda o desenvolvimento orientado à testes - TDD onde alguns testes unitários são escritos primeiramente, seguido da implementação do código necessário para passar nestes testes e refatoração do código produzido. Como resultado, o código escrito é mais simples, pois somente o necessário foi implementado, e coberto por testes já escritos anteriormente.

7. Gestão do Projeto

Gerenciar um projeto é aplicar conhecimentos, habilidades, ferramentas e técnicas às atividades do projeto a fim de atender aos requisitos de negócio. O gerenciamento inclui: (i) identificar requisitos; (ii) adaptação às diferentes necessidades à medida que o projeto é planejado e realizado; (iii) balancear as restrições conflitantes do projeto que incluem, mas não se limitam ao escopo, qualidade, tempo e custo [IEEE, 1998].

Para se aplicar os conceitos de controle de processo faz-se necessário diferenciar duas modalidades de metodologia, não concomitantes, aplicáveis ao gerenciamento de projetos: metodologia com processo definidos (ou prescritivos) e a metodologia empírica. A primeira modalidade define o contexto do projeto e determina o escopo, custo, tempo e qualidade que se consolidam em um detalhado plano, onde as entradas são estabelecidas logo no início do projeto [Martins, 2007]. Já a segunda, define os resultados de forma abrangente e superficial, começando por um contexto inicial, que evolui e se adapta ao longo da execução [Schwaber, 2010].

7.1. Gerenciamento de Processos

A metodologia com processos definidos e prescritivos, também chamada de abordagem tradicional, é a mais adequada em situações onde os passos a serem executados, em geral, são conhecidos, como por exemplo, na implantação de uma infraestrutura de TI [Martins, 2007]. Frederick W. Taylor, considerado o pai da Teoria Clássica da Administração (Teoria de Taylor), desenvolveu em seus estudos teóricos, quatro princípios básicos [Silva, 2001]

1. Desenvolvimento de um método científico para o trabalho do operário. Esse princípio reforça o planejamento inicial necessário em um processo prescritivo. Esse planejamento envolve a busca do melhor processo de executar uma atividade no menor tempo possível.
2. Estabelecimento de processo científico de seleção e treinamento do operário. O princípio insere a especialização e treinamento de recursos humanos.
3. Cooperação entre as gerências e os operários. Com o intuito de garantir que o trabalho fosse realizado de acordo com o que foi planejado e da maneira certa, esse princípio implanta o conceito de controle de atividades.

4. Divisão do trabalho dos operários em função da sua especialização. Esse princípio refere-se à padronização de atividades e tarefas que um processo prescritivo possui. Cada atividade passa a ter um responsável especialista por desenvolvê-la.

Esses princípios transferem a responsabilidade de decisão sobre as tarefas, que antes eram exercidas pelos operários, para as mãos de um gerente. Pressupondo que existe uma melhor e padronizada forma de trabalho, cabe então ao gerente encontrá-lo. Essa forma de organizar o trabalho estimula a produção em série, de massa, a baixos custos, e a extrema especialização (automatização do ser humano) [Silva, 2001]. Esse arcabouço teórico é base de sustentação de diferentes métodos de desenvolvimento e gestão da produção, como por exemplo o guia PMBoK, no contexto do gerenciamento de projetos de software.

As primeiras abordagens formais sobre o gerenciamento de projetos de software tiveram a intenção de criar um processo eficiente e previsível, fortemente baseado em um planejamento minucioso das atividades a serem conduzidas. Essa metodologia procurava modelar o desenvolvimento de software como uma linha de produção, um processo ou sequência de processos, que pudesse ser seguido repetitiva e mecanicamente. Essas abordagens esbarraram em dois obstáculos notáveis e inerentes a própria natureza do desenvolvimento de software: a instabilidade dos requisitos ao longo de um projeto e a dependência do processo como um todo de pessoas e da comunicação entre elas.

No desenvolvimento de software é o mesmo que dizer que há uma processo padrão de desenvolvimento, onde há separação de funções e papéis, e que as atividades são determinísticas, previamente definidas, cabendo ao gerente identificar e sequenciar logicamente essas atividades, de forma a seguir um plano preditivo, definido nos meses iniciais de um projeto. Um software possui diversas características que o transformam em algo com propriedades intrínsecas, com vários estados possíveis, fazendo do software algo altamente complexo [5cqualibr, 2010].

O desenvolvimento de software é semelhante a criação de uma receita, um processo de aprendizado envolvendo tentativas e erros, o que difere acentuadamente do passo-à-passo característico de se seguir uma receita [Poppendieck and Poppendieck, 2011]. Nesse contexto, surgiram os métodos com os ágeis, introduzindo no gerenciamento do desenvolvimento de software uma mentalidade de receptividade a mudanças e resultados rápidos, priorizando a interação entre pessoas e as atividades que adicionam valor do ponto de vista do usuário. Esses princípios foram agregados aos métodos

ágeis através da assimilação e adaptação de padrões de melhoria global (da eficácia de uma organização) oriundos também da indústria de manufatura, como por exemplo: a teoria da Visão Sistêmica e Teoria das Restrições.

7.2. Gerenciamento segundo a teoria crítica

Muitas organizações de desenvolvimento de software estão se esforçando para se tornar mais ágeis. Equipes ágeis bem sucedidas estão produzindo software de alta qualidade para melhor atender, mais rapidamente, as necessidades dos usuários, com um custo menor e maior qualidade do que as equipes tradicionais. Elas são capazes de levar resultados aos usuários finais mais rapidamente e com um maior grau de satisfação dos mesmos, provendo maior visibilidade para o processo de desenvolvimento, bem como com uma maior previsibilidade, como consequência, e não como causa [Cohn, 2005].

Primeiro, um método empírico requer frequentemente transparência, inspeções e adaptações durante o projeto, que é definido de forma inexata e pode gerar resultados imprevisíveis. É mais adequado e indicado para projetos de inovação e criação de novos resultados, como o desenvolvimento de software, por exemplo. A metodologia empírica, um dos pilares dos métodos ágeis, é indicada nas situações onde as entradas do processo variam e o processo é muito complexo para produzir resultados semelhantes [Martins, 2007]. O método empírico foi introduzido pela escola das relações humanas e visava contrapor-se à escola clássica, desde a década de 20 [Silva, 2001].

Segundo, a concepção de sistema aberto (das áreas da Física e Biologia, ou seja, aqueles que tem interação com o meio) reconhece que o sistema mantém um relacionamento dinâmico com seu ambiente, ou seja, o sistema recebe várias entradas, atua sobre elas, e exporta determinadas saídas. O modelo dinâmico dos sistemas abertos considera ainda que um sistema é composto de subsistemas inter-relacionados, afirmando que o todo não é simplesmente como a soma das partes individuais. Recentemente essa teoria também tem sido utilizada para explicar o comportamento da produção de um software [Silva, 2001].

Terceiro, ao encontro da teoria de sistemas abertos, aproxima-se a teoria das restrições, a qual busca aumentar a capacidade produtiva da organização através de melhorias no fluxo do processo, ao enxergar a organização como um sistema em que todas as partes são inter-relacionadas [Katayama, 2011]. A aplicação desta visão no desenvolvimento ágil é contemplada buscando identificar os pontos críticos

no processo ou pontos de alavancagem (aqueles que representam os gargalos) e melhorá-los.

Quarto, os sistemas complexos adaptativos são redes com grande número de agentes que interagem entre si e com muitas características em comum. São complexos, pois, estão além da capacidade descritiva e são adaptativos, pois, são capazes de se adaptar a novas condições impostas a eles. Organizações são sistemas adaptativos complexos, e criativos e inovadores quando dão lugar para a inovação na proximidade do caos [Lorenz, 1995]. Trata-se de um estado paradoxal, pois é estável e instável ao mesmo tempo, orientado pela contraditória dinâmica da competição e cooperação, da amplificação e restrição, da exposição a tensão criativa e proteção dela [Iamamoto and Gaillard, 1999] Por fim, mudanças em sistemas acontecem de forma mais freqüente e profunda, e que é cada vez mais difícil prever e controlar tais mudanças [Sato et al., 2005]. Isso exige uma nova forma de aceitá-las e não controlá-las. Baseado nisso, a teoria da complexidade e os sistemas adaptativos complexos oferecem uma maneira para as organizações conviverem com elas. Nesse contexto de evolução das empresas em direção a sistemas evolutivos complexos, o papel do gerente de projetos como uma autoridade central controladora desaparece, dando lugar aos agentes de mudança ou aos agentes reguladores, trabalhando no sentido de influenciar as pessoas e viabilizar a auto-organização dos projetos nas organizações.

Em suma, todas essas teorias juntas, ou seja (i) Escola das Relações Humanas (foco no indivíduo); (ii) Sistemas Abertos (relacionamento dinâmico entre as partes); (iii) Restrições (foco em mudanças e progresso contínuo); (iv) Visão Sistêmica (o todo não é igual à soma das partes, não é igual), e (v) Sistemas Complexos e Adaptativos (estão além da capacidade descritiva e se adaptam à dinâmica do sistema), reforçam, justificam e validam, o processo empírico e as metodologias ágeis para gerenciamento do desenvolvimento.

7.3. Iniciativas de Adoção de Métodos Ágeis na Administração Pública Federal

As metodologias ágeis começaram a ganhar espaço no início da década de 2000, regidas pelo Manifesto Ágil, e desde então vêm ganhando crescente popularidade. No cenário mundial, elas já são metodologias bastante difundidas entre diversos setores. Atualmente, diversas organizações públicas brasileiras estão iniciando investimentos em adoção de contratações de fornecedores de software utilizando métodos ágeis e, portanto, estão começando a difundir tais métodos também no setor

público [Melo et al., 2012].

Recentemente, o Tribunal de Contas da União publicou o Acórdão no 2314/2013, o qual contém um relatório de levantamento elaborado pela Secretaria de Fiscalização de Tecnologia da Informação (SEFTI), cujo objetivo foi conhecer as bases teóricas do processo de desenvolvimento de software com metodologia ágil, além de conhecer experiências práticas de contratação realizadas por instituições públicas federais. Nesse acórdão são relatadas as experiências de adoção e contratação de serviços utilizando métodos ágeis no contexto do (i) Tribunal Superior do Trabalho (TST); (ii) Banco Central do Brasil (Bacen); (iii) Instituto do Patrimônio Histórico e Artístico Nacional (IPHAN); (iv) Instituto Nacional de Estudos e Pesquisas Educacionais Anísio Teixeira (INEP), e Supremo Tribunal Federal (STF) (TCU, 2013).

No referido acórdão, baseado nas experiências citadas, é concluído que, apesar dos riscos inerentes em qualquer processo de adoção a adaptação de novas metodologias na Administração Pública Federal (APF), a adoção de métodos ágeis no âmbito da APF em contratos para desenvolvimento de software é demonstradamente viável.

7.4. MGP-SISP e Métodos de Gestão deste Projeto

Outra tendência que tem sido notada recentemente em projetos ágeis, à medida em que esses vão sendo adotados em maior escala, é a combinação das técnicas ágeis com algumas técnicas consideradas tradicionais, produzindo uma metodologia híbrida mais apropriada para organizações maiores [West and Grant, 2010]. Nesse contexto, a utilização de métricas de software, projeto e processo vêm auxiliar o gerenciamento de equipes ágeis adicionando uma formalização necessária para maior previsibilidade, como consequência e não como fim, especialmente em grandes projetos, como o objeto deste plano de trabalho.

Um esclarecimento nesse cenário é que o Núcleo de Contratações de Tecnologia da Informação (NCTI), ligado à Comissão de Coordenação do Sistema de Administração dos Recursos de Tecnologia da Informação do Governo Federal (SISP), normatizada a Metodologia de Gerenciamento de Projetos do SISP (MGP-SISP), que, em suma, é uma adaptação do PMBoK às necessidades da Administração Pública Federal, e por conseguinte, trás consigo todos os pressupostos teóricos e limitações supracitados da Teoria de Taylor.

Neste projeto, no que diz respeito exclusivamente à gestão tática, será utilizado o método baseado no conjunto de boas práticas de gerenciamento de projetos do PMBoK. Para apoio à essa atividade, e em atendimento ao normativo interno do CDT/FUB, será utilizada a ferramenta DotProject. Trata-se de ferramenta de software livre para automação e suporte aos processos de gerenciamento, prescritos no PMBoK, que foi evoluída para atender as necessidades do CDT/UnB. Por meio dessa ferramenta será possível que os gestores da SLTI/MP acompanhem, do ponto de vista tático, a execução do escopo geral, tempo e custo do projeto.

De outro lado, para a gestão operacional, serão utilizados métodos ágeis de engenharia de software, como o Scrum e eXtream Programming (XP), além da filosofia de gestão de desenvolvimento de software LEAN [Poppendieck and Poppendieck, 2011], que favorecem o desenvolvimento de escopo variável, de maneira que este seja progressivamente refinado, de forma iterativa e incremental, até que ele seja encerrado. Em outras palavras, levam-se em conta o esperado aprendizado das equipes da FUB e SLTI/MP ao longo da execução do projeto, as equipes acomodarão as mudanças. Nesse contexto, destaca-se o uso de ferramentas de gestão de projetos como o Redmine, além do uso do Kanban físico.

Dessa forma, a equipe da FUB dará suporte a elaboração e validação de um modelo aplicável de gestão de forma a garantir à SLTI/MP o registro de padrões e indicadores de atendimento (e de conformidade ao atendimento) que permitam o acompanhamento da execução das atividades previstas no projeto, bem como o atendimento aos resultados(seus marcos e prazos) e na qualidade exigida do desenvolvimento (evolução) das soluções sistêmicas a serem produzidas, sendo inclusive, objeto de pesquisa-ação.

7.5. Métodos e Práticas de Gerenciamento do Projeto

A seguir são apresentados brevemente abordagens, métodos e práticas de processo que serão utilizadas pelas equipes da FUB.

7.5.1. Scrum

O Scrum é um framework que começou a ser desenvolvido na década de 90 (SCHWABER, 1996) e que se concentra nos aspectos gerenciais. Assim, propõe um conjunto de papéis e cerimônias que visam a rápida interação entre as pessoas, acompanhamento diário do desenvolvimento e adaptação do processo.

Papéis

Neste processo são definidos três papéis os quais devem interagir de diversas maneiras para o desenvolvimento do software com aplicação dos princípios ágeis: Scrum Master, responsável por ensinar e acompanhar as práticas do Scrum e resolver impedimentos que possam interferir no processo; Product Owner, responsável por representar as necessidades do cliente, interagindo diretamente com a equipe de desenvolvimento, definindo prioridades e critérios de aceitação e qualidade baseados nos valores do cliente; Equipe de Desenvolvimento, conjunto de pessoas responsáveis por iterar atividades de desenvolvimento das funcionalidades do software.

Ciclo de Desenvolvimento

O desenvolvimento com Scrum é baseado em sprints (iterações) de curto prazo, uma à três semanas, onde um conjunto de atividades são executadas em cada uma delas. A partir de um conjunto de requisitos do cliente as sprints são planejadas em uma reunião intitulada Planejamento de Sprint onde todos os papéis devem estar presentes. Nesta cerimônia os requisitos são explicados e priorizados pelo Product Owner enquanto são também discutidos pela Equipe de Desenvolvimento. A partir de um acordo, considerando as expectativas e prioridades do cliente, assim como a capacidade produtiva estimada pela própria Equipe de Desenvolvimento, são selecionados um conjunto de requisitos que serão desenvolvidos ao longo da Sprint, compondo o Sprint Backlog.

Com o Sprint Backlog definido, a Equipe de Desenvolvimento diariamente utilizam de diversas técnicas ágeis de desenvolvimento, que serão posteriormente apresentadas na sub-seção que introduz o XP, para desenvolver os requisitos de acordo com os critérios definidos para os mesmos. No início de cada dia é realizado uma cerimônia chamada Daily Meeting cujo o tempo de realização não deve ultrapassar 15 minutos. Nesta reunião os membros da Equipe de Desenvolvimento ficam em pé e relatam o que está sendo feito, além de ressaltar eventuais problemas existentes para execução de

suas tarefas, permitindo um acompanhamento diário e rápido.

Ao fim da Sprint ainda existem duas reuniões. A primeira é a Revisão da Sprint onde a Equipe de Desenvolvimento apresenta o trabalho feito. Nesse mesmo encontro, o Product Owner faz testes e verifica se a meta foi atingida. A segunda reunião está relacionada à melhoria contínua do processo e é chamada Retrospectiva da Sprint. Nesta ocasião, a equipe analisa o próprio trabalho e avalia oportunidades de melhorias para os próximos ciclos de produção. Desta forma, várias iterações seguem e o software é incrementado a partir da aplicação do Scrum e da colaboração dos papéis entre si. Requisitos Os requisitos do software são especificados por meio de User Stories ou Histórias de Usuário que são escritos na linguagem natural do usuário final ou cliente, onde são capturadas as necessidades deste usuário como parte da execução do seu trabalho. Uma Histórias de Usuário captura três informações essenciais para compreensão do requisito: Quem a realiza, O Que ele faz e o Porquê de sua necessidade. Por serem muito simples, as User Stories podem ser complementadas com a descrição de testes de aceitação que definem condições para desenvolvimento da história, documentam regras e removem possíveis ambiguidades.

O conjunto de história de usuário já identificados para um software compõe o Backlog do software. Este artefato é uma lista organizada e priorizada de tudo que deve ser necessário para um software. O Backlog do software evolui assim como o software e seu ambiente. Como resultado dos planejamentos de sprint, tem-se um conjunto de itens selecionados do Backlog do software que farão parte do próximo incremento do software.

7.6. Do ateste das etapas e organização do fluxo de trabalho

Operação (produção)

De acordo com o ciclo de produção representado na figura acima, a cada 4 meses a equipe CDT/FUB liberará um resultado $R[1, \dots, 7]$ para uso externo, no ambiente de operação disponibilizado pela DTI/MP. Serão, ao todo, 7 etapas de resultados a serem disponibilizadas ao longo da execução do projeto. O credenciamento para liberação de um resultado (protótipo) em ambiente de produção se dará: i) após o fim do período de homologação e ii) ter recebido o ateste definitivo por parte do da equipe da SLTI/MP.

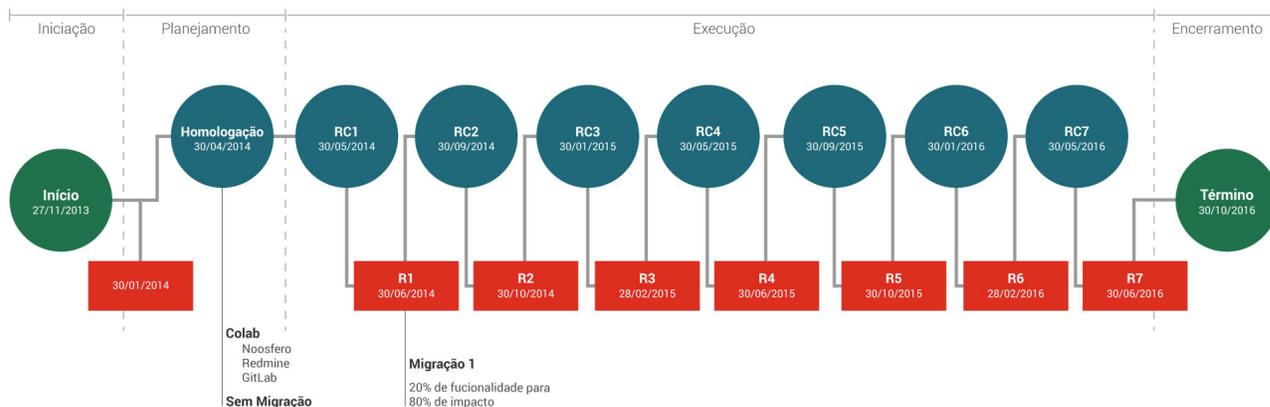


FIGURA 1. *Etapas de entregas de resultados*

Homologação

O período de ateste e homologação, por parte do MP, se dará imediatamente após a disponibilização de um resultado candidato RC-[1,...,7] no ambiente disponibilizado pela DTI/MP. Decorrerá um período de 30 dias, a partir da disponibilização do resultado candidata, para que os últimos ajustes, testes finais de migração dos dados, e correções de pequenos defeitos que por ventura, venham se materializar.

A linha de desenvolvimento do resultado candidato em homologação não terá atividades de desenvolvimento de novos requisitos. O objetivo é estabilizar a versão do resultado, com vistas à liberação para o ambiente de produção.

O credenciamento para liberação de um resultado candidato em ambiente de homologação se dará: i) após o fim do período de desenvolvimento de um ciclo (3 meses) e a consequente liberação dos resultados intermediários-RI; e ii) ter recebido o ateste provisório por parte do da equipe da SLTI/MP.

Resultados Intermediárias

Mensalmente a equipe CDT/FUB disponibilizará uma realease intermediária no ambiente de desenvolvimento e testes disponibilizado pela DTI/MP. Trata-se de oportunidade da equipe da SLTI/MP poder realizar de forma não controlada diferentes testes, retroalimentando o desenvolvi-

mento da equipe CDT/FUB com feedbacks de possíveis alterações. Todas as alterações solicitadas por parte da equipe SLTI/MP serão tratadas como itens de backlog, e portanto, serão priorizados juntamente com as demais histórias de usuários.

O credenciamento para liberação de um resultado intermediário em ambiente de desenvolvimento e testes se dará: i) após o fim de dois ciclos de desenvolvimento (sprints); ii) da integração do incremento de software; iii) da execução do suítes de testes integrado; iv) da formalização da entrega, por meio da execução da suíte do testes automatizados.

Ciclos de desenvolvimento

A execução das etapas acontecerão preferencialmente nas dependências da UnB, não se limitando a este espaço, caso seja necessário. As sprints terão duração de 15 dias, representando a cadência do tempo em cada ciclo de desenvolvimento. Será medida a capacidade de produção da equipe CDT/FUB, de forma a se identificar o limitador do trabalho. A partir de então, a quantidade de trabalho possível de ser realizada dentro de cada ciclo da etapa será utilizada como parâmetro para que as equipes CDT/FUB e SLTI/MP a utilizar nas reuniões de planejamento, que ocorrerão no primeiro dia de cada ciclo de desenvolvimento.

Ao final de cada ciclo, será realizada a reunião de revisão da sprint, onde a equipe CDT/FUB apresenta para a equipe SLTI/MP o incremento do protótipo construído no ciclo, caracterizando assim, o desenvolvimento guiado pelas prioridades elencadas pelo ministério. Ainda, ao final de duas sprints será gerado um protótipo intermediário e disponibilizada no ambiente de desenvolvimento e testes provido pela DTI/MP.

7.7. As equipes

Por meio de um trabalho coordenado e interdependente entre as equipes da Secretaria de Logística e Tecnologia da Informação SLTI/MP, representando o Departamento de Governança e Sistema de Informação – DGSI, e da Universidade de Brasília, as etapas de cada fase serão sucessivamente e progressivamente refinadas, discutidas, executadas e documentadas. As atividades do projeto que estejam a cargo dos profissionais da SLTI/MP serão orientadas pelos pesquisadores. Para organização e trabalho entre as equipes, será exigida uma estrutura mínima da SLTI/MP. Assim sendo, a

organização das equipes se dará da forma com ilustrada na Figura 2.

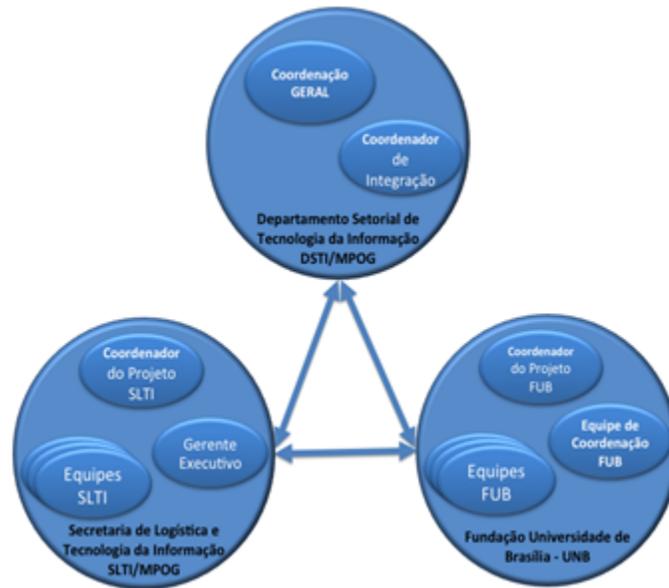


FIGURA 2. Distribuição de equipes.

As equipes responsáveis pelos esforços técnicos serão lideradas por um gerente executivo, no caso da SLTI/MP, e por uma equipe de coordenação, no caso do CDT/FUB.

Propõe-se a constituição de uma equipe de gestão e qualidade, responsável pelos processos de gestão, pelo aceite dos protótipos do projeto, e pelo acompanhamento do projeto; sendo assim constituída:

- 1 Coordenador Geral SE/MP
- 1 Coordenador Inter Projetos DSGI/MP
- 1 Coordenador de Projetos SLTI/MP
- 1 Gerente Executivo SLTI/MP
- 1 Coordenador do Projeto CDT/FUB
- 1 Equipe de Coordenação (professores) CDT/FUB

As equipes técnicas serão formadas por engenheiros de software e artistas-designers, na sua maioria representadas por alunos da UnB, que terão sua formação complementada por meio da oportunidade de participarem como protagonistas em um projeto desta magnitude. As equipes possuem diferentes experiências e qualificações, adequadas a dinâmica e complexidade do projeto.

Cabe salientar que, segundo as orientações da FUB/CDT, a composição da Equipe UnB deve ser formada, preferencialmente por, pelo menos 60% de integrantes do próprio quadro, entre professores pesquisadores e alunos. Dessa forma, isso nos permite que alguns pesquisadores fora do quadro da FUB, notadamente expoentes na comunidade de software livre, também comporão as equipes técnicas de forma a atuarem em execução/coordenação de atividades técnicas que consideramos não triviais.

A transferência de tecnologias e do conhecimento se dará primeiro pela interação direta entre bolsistas e pesquisadores do projeto e profissionais da SLTI/MP, inclusive nas decisões de escolhas técnicas, formas de processo, metodologias. A apresentação de resultados será feita na forma de seminários para apresentação e discussão dos protótipos intermediários e finais, atendendo os normativos vigentes desta cooperação.

7.8. A Estrutura Analítica do Projeto - EAP

As tabelas a seguir representam a visão geral da EAP do projeto. A EAP é uma representação sistêmica do projeto que evidencia seus principais resultados, bem como as fases necessárias a sua conclusão.

Fase de Iniciação/Pacotes de Trabalho	Descrição
1) Assinar TDC	Termo de cooperação com a definição do objeto e proposta inicial das metas, planos e metodologias de trabalho.
2) Planos de Trabalho dos recursos humanos	Criação e formalização dos planos de trabalho individual dos recursos humanos do projeto.
3) Elaborar RT com estudos iniciais sobre Arquitetura da Informação	Estudos para proposição de um modelo de organização do conteúdo e das funcionalidades do portal, bem como os seus trajetos de acesso, atalhos de navegação e rótulos.
5) Elaborar RT com estudos iniciais sobre ambientes de versionamento	Estudos sobre ambientes de versionamento de código-fonte.

5) Elaborar RT com estudo iniciais Arquitetura de Software	Estudos para definição e documentação das principais decisões de projeto no ponto de vista da engenharia de software em termos de componentes, camadas, frameworks, ferramentas, etc.
--	---

Tabela 1: Fase de Iniciação

Fase de Planejamento/Pacotes de Trabalho	Descrição
6) Planos de Trabalho conforme os estudos iniciais	Atualização do plano de trabalho com a definição das metas, cronogramas e metodologia de trabalho.
7) RT de Planejamento dos resultados	Definição e planejamento dos estudos e relatórios técnicos.
8) Pesquisa em sistema de controle de versão	Estudo de ferramentas destinadas a controle de versão de código-fonte.
9) Pesquisa em evolução do Sistema de Gerenciamento	Estudo de ferramentas de gerenciamento de projeto com as ferramentas de forge.
10) Pesquisa em Evolução da plataforma de Integração	Estudo de plataforma de integração de ferramentas/ambientes.
11) Pesquisa em Evolução da plataforma integrada de colaboração	Estudo de plataforma integrada de colaboração para prover interação entre perfis de usuários, comunidades e projetos.
12) Pesquisa em Migração de Software	Estudo sobre o conjunto de esforço destinado à migração de conteúdos e projetos do atual SPB para a nova plataforma a ser construída.

13) Pesquisa em Evolução do Sistema de Identidade Visual	Estudo para Proposição de um sistema de identidade visual para o portal do SPB para integrar de forma coerente todos os seus ambientes.
14) Pesquisa em Evolução da Arquitetura da Informação	Estudo para Proposição de um modelo de organização do conteúdo e das funcionalidades do portal, bem como os seus trajetos de acesso, atalhos de navegação e rótulos.
15) Pesquisa em Evolução da superfície da interface gráfica do portal	Estudo para proposição dos elementos gráficos da interface e aplicação das regras especificadas pelo sistema de identidade visual.
16) Pesquisa em coleta de percepção dos usuários da versão antiga do portal	Estudo de usabilidade com usuários do antigo portal SPB.
17) Estudos de Evolução de plataforma de colaboração do Governo	Estudo para adaptação e melhorias de funcionalidades existentes relacionadas a interação social e colaborativa, principalmente voltada à construção de comunidades em ambiente com foco para organizações da Administração Pública Federal.
18) Estudos de Evolução de plataforma de integração do Governo	Estudo para adaptação do ambiente de integração e colaboração provido pelo Programa Interlegis.

Tabela 2: Fase de Planejamento

Fase de Execução/Pacotes de Trabalho	Descrição
19) Estudos de Evolução de plataforma integrada de colaboração	Estudo de melhorias de funcionalidades existentes relacionadas a interação social e colaborativa, principalmente voltada à construção de comunidades, à adaptação para ativos de software e outros tipos de grupos relacionados ao SPB.
20) Estudo de Evolução da plataforma de Integração	Estudos para adaptação e integração de novos serviços para interoperabilidade e evolução do SPB para utilização dos diversos serviços a partir de interfaces e funcionalidades unificadas.
21) Estudos avançados sobre Migração	Estudos para propor uma conjunto de esforço destinado para a migração de conteúdos, usuários e projetos do atual SPB para a nova plataforma a ser construída.
22) Estudos sobre um mecanismo para disponibilização das informações fornecidas pela plataforma	Estudo com vistas ao desenvolvimento de uma <i>Application Programming Interface-API</i> que funcionará como um serviço que possa ser "consumido" por outras aplicações e/ou aplicativos móveis
23) Estudos sobre a definição de um perfil de comunidade voltado para Instituições de Ensino Superior	Estudo para adaptação na plataforma de tal forma a fornecer um perfil específico para Instituições de Ensino Superior colaborarem com o software público brasileiro.
24) Estudos sobre um mecanismo de busca global na plataforma	Estudo de tecnologia para integrar as buscas nas diferentes plataformas e ambientes do portal
25) Estudos avançados sobre a evolução do sistema de identidade visual	Estudos para proposição de um sistema de identidade visual para o portal do SPB para integrar de forma coerente todos os seus ambientes.

26) Estudos avançados sobre a evolução da estrutura de interação humano-computador do sistema	Estudos para Revisão da narrativa de interação das ferramentas do portal do SPB, revendo a lógica de suas operações, representações e respostas para os seus usuários.
27) Estudos avançados sobre a evolução da arquitetura da Informação	Proposição de um modelo de organização do conteúdo e das funcionalidades do portal, bem como os seus trajetos.
28) Estudos sobre a Evolução da superfície da interface gráfica do portal	Estudos para proposição dos elementos gráficos da interface e aplicação das regras especificadas pelo sistema de identidade visual.
29) Estudos Avançados sobre plataforma de monitoramento de código-fonte	Estudos sobre a qualidade interna e automação do monitoramento de métricas de código-fonte.
30) Estudos Avançados para a definição e automação da infraestrutura	Estudos sobre as técnicas e tecnologias de DevOps (desenvolvimento integrado com operação) para a infraestrutura do novo Portal.
31) Estudos Avançados para a sustentação da plataforma	Estudos sobre os procedimentos e como manter a infraestrutura e a plataforma atualizada.
32) Estudos sobre licenças de software livre	Estudos sobre as licenças de software livre que, do ponto de vista jurídico de legislação brasileira, podem ser utilizadas por projeto do SPB.
33) Relatório técnicos de consolidação dos estudos e resultados	RT com os resultados das fases por etapas do projeto.

Tabela 3: Fase de Execução: macro-atividade e descrição.

Fase de Encerramento/Pacotes de Trabalho	Descrição
34) Elaborar Relatório de encerramento do projeto	RT de Formalização do relatório de encerramento do projeto.
35) Desmobilizar Equipe	Entrega de relatórios dos recursos humanos dos projetos.
36) Elaborar Relatório de Prestação de Contas	RT de Prestação de contas.
37) Compilação de resultados das pesquisas	RT com compilação e divulgação de resultados obtidos nas pesquisas realizadas durante o projeto.

Tabela 4: Fase de Encerramento: macro-atividade e descrição.

8. Cronograma de execução (fases/etapas)

O cronograma de execução, ou seja, as fases e as etapas, está apresentado nas 3 (três) figuras abaixo, seguindo uma sequência de cortes para melhor apresentar tais itens.

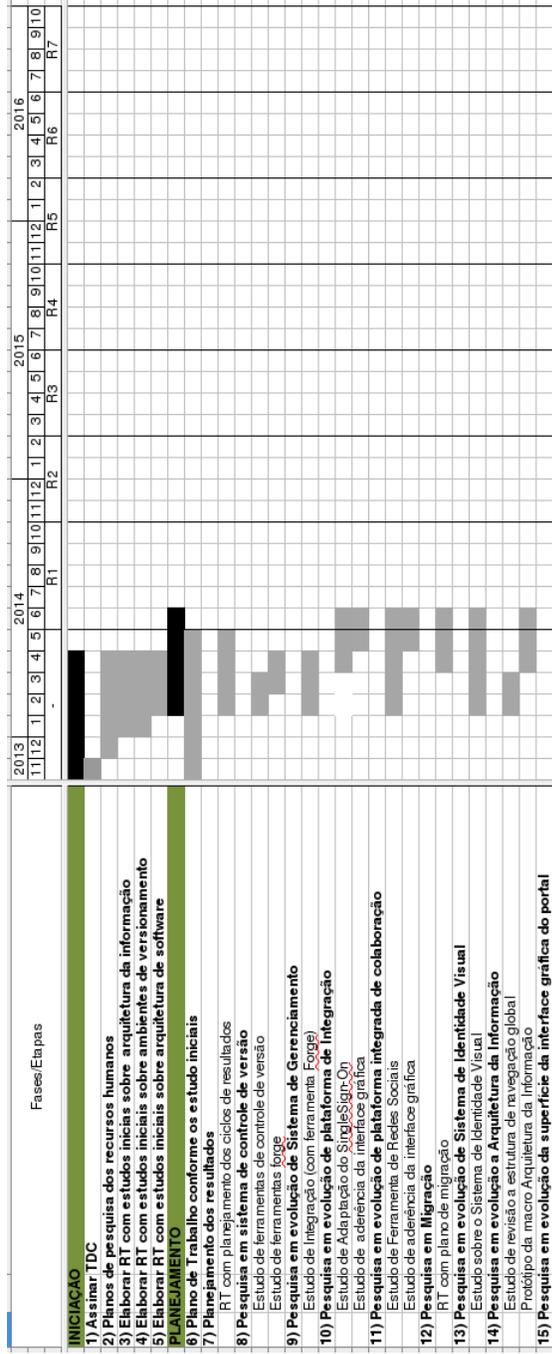


FIGURA 3. Cronograma de execução: corte 1 de 3

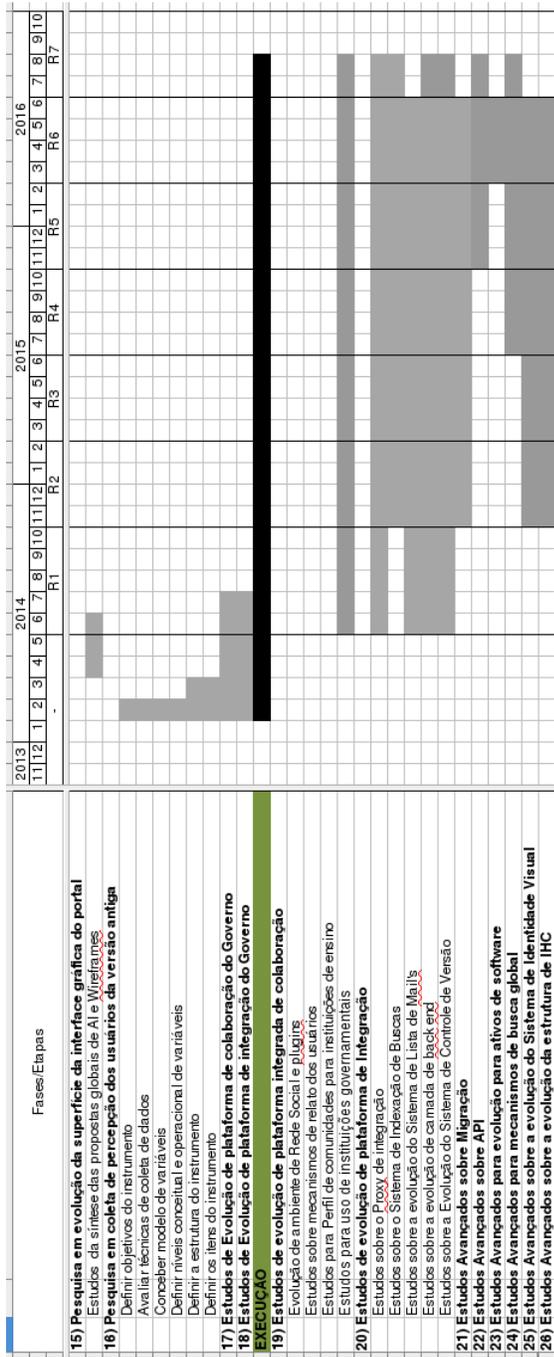


FIGURA 4. Cronograma de execução: corte 2 de 3

9. Orçamento

Este projeto é formalmente concebido como projeto colaborativo no contexto de acordo de cooperação intra Governo Federal. Essa característica faz com que a natureza da execução, além da relação entre as partes se dê em forma diferenciada daquela de contratos comerciais. Nesse contexto, o dimensionamento do projeto possui como base o programa de bolsas de pesquisa do CDT/FUB que é similar aos programas do MCT/FINEP, MCT/CNPq e MEC/CAPES. Dessa forma, o dimensionamento é feito pelo quantitativo de pesquisadores, estudantes, e/ou profissionais que são necessários ao projeto e da especificação do perfil adequado à bolsa. O CDT/FUB, unidade da UnB responsável pela administração do projeto, trabalha com uma grande quantidade de pesquisadores, inclusive de outras áreas de conhecimento, e tem histórico de projetos de Desenvolvimento & Pesquisa com praticamente todas as três Poderes da República.

O programa de trabalho, ação, projeto e fonte orçamentária são as seguintes:

- Programa: 04.126.2038 – Democracia e Aperfeiçoamento da Gestão Pública
- Ação: 20U2 – Gestão dos Recursos de Tecnologia da Informação e Logística do Serviço Público Federal
- Projeto: 000A – Gestão do Sistema de Administração de Recursos de Tecnologia da Informação – SISP
- Fonte: 0100 – Recursos ordinários.

9.1. Cronograma de desembolso

Rubrica	Desembolso pelo MP	2013	2014	2015	2016	Total
33.90.14	Diárias - Civil	R\$ 0,00	R\$ 0,00	R\$ 0,00	R\$ 0,00	R\$ 0,00
33.90.14.15	Diárias no exterior - Civil	R\$ 0,00	R\$ 0,00	R\$ 0,00	R\$ 0,00	R\$ 0,00
33.90.20	Auxílio-Financeiro a Pesquisadores	R\$ 154.400,00	R\$ 950.800,00	R\$ 1.260.000,00	R\$ 550.000,00	R\$ 2.915.200,00
33.90.30	Material de Consumo	R\$ 4.125,00	R\$ 12.000,00	R\$ 10.400,00	R\$ 0,00	R\$ 26.525,00
33.90.33	Passagens e Despesas com Locomoção	R\$ 0,00	R\$ 0,00	R\$ 0,00	R\$ 0,00	R\$ 0,00
33.90.36	Outros Serviços de Terceiros - Pessoa Física	R\$ 0,00	R\$ 0,00	R\$ 25.200,00	R\$ 0,00	R\$ 25.200,00
33.90.36.02	Diárias a Colaboradores Eventuais no País	R\$ 0,00	R\$ 0,00	R\$ 0,00	R\$ 0,00	R\$ 0,00
33.90.39	Outros Serviços de Terceiros - Pessoa Jurídica	R\$ 0,00	R\$ 24.000,00	R\$ 30.000,00	R\$ 0,00	R\$ 54.000,00
33.91.47	Obrigações Tributárias e Contributivas	R\$ 0,00	R\$ 0,00	R\$ 5.040,00	R\$ 0,00	R\$ 5.040,00
44.90.52	Equipamentos e Material Permanente	R\$ 0,00	R\$ 60.000,00	R\$ 84.000,00	R\$ 72.000,00	R\$ 216.000,00
	Total	R\$ 158.525,00	R\$ 1.046.800,00	R\$ 1.414.640,00	R\$ 622.000,00	R\$ 3.241.965,00

TABELA 5

Descentralização MP - CDT/FUB

9.2. Detalhamento do orçamento

33.90.20 - Auxílio-Financeiro a Pesquisadores (2013)			
Classificação	Valor	Meses	Valor total
Pesquisador Sênior - Nível A	R\$ 7.000,00	2	R\$ 14.000,00
Pesquisador Sênior - Nível A	R\$ 7.000,00	2	R\$ 14.000,00
Pesquisador Sênior - Nível B	R\$ 5.000,00	2	R\$ 10.000,00
Pesquisador Sênior - Nível C	R\$ 4.000,00	2	R\$ 8.000,00
Pesquisador Sênior - Nível C	R\$ 4.000,00	2	R\$ 8.000,00
Pesquisador Sênior - Nível C	R\$ 4.000,00	2	R\$ 8.000,00
Pesquisador -Pesquisa, Desenvolvimento e Inovação - PDI - Nível C	R\$ 2.500,00	2	R\$ 5.000,00
Pesquisador -Pesquisa, Desenvolvimento e Inovação - PDI - Nível C	R\$ 2.500,00	2	R\$ 5.000,00
Pesquisador -Pesquisa, Desenvolvimento e Inovação - PDI - Nível C	R\$ 2.500,00	2	R\$ 5.000,00
Pesquisador -Pesquisa, Desenvolvimento e Inovação - PDI - Nível C	R\$ 2.500,00	2	R\$ 5.000,00
Pesquisador -Pesquisa, Desenvolvimento e Inovação - PDI - Nível C	R\$ 2.500,00	2	R\$ 5.000,00
Pesquisador -Pesquisa, Desenvolvimento e Inovação - PDI - Nível C	R\$ 2.500,00	2	R\$ 5.000,00
Pesquisador -Pesquisa, Desenvolvimento e Inovação - PDI - Nível C	R\$ 2.500,00	2	R\$ 5.000,00
Pesquisador -Pesquisa, Desenvolvimento e Inovação - PDI - Nível D	R\$ 2.000,00	2	R\$ 4.000,00
Pesquisador -Pesquisa, Desenvolvimento e Inovação - PDI - Nível C	R\$ 2.500,00	2	R\$ 5.000,00
Pesquisador -Pesquisa, Desenvolvimento e Inovação - PDI - Nível D	R\$ 2.000,00	2	R\$ 4.000,00
Pesquisador -Pesquisa, Desenvolvimento e Inovação - PDI - Nível D	R\$ 1.800,00	2	R\$ 3.600,00
Pesquisador -Pesquisa, Desenvolvimento e Inovação - PDI - Nível D	R\$ 1.800,00	2	R\$ 3.600,00
Pesquisador -Pesquisa, Desenvolvimento e Inovação - PDI - Nível D	R\$ 1.800,00	2	R\$ 3.600,00
Pesquisador -Pesquisa, Desenvolvimento e Inovação - PDI - Nível D	R\$ 1.800,00	2	R\$ 3.600,00
Pesquisador -Pesquisa, Desenvolvimento e Inovação - PDI - Nível D	R\$ 1.800,00	2	R\$ 3.600,00
Apoio Técnico à Pesquisa - Nível A	R\$ 1.500,00	2	R\$ 3.000,00
Apoio Técnico à Pesquisa - Nível A	R\$ 900,00	2	R\$ 1.800,00
Apoio Técnico à Pesquisa - Nível A	R\$ 900,00	2	R\$ 1.800,00
Apoio Técnico à Pesquisa - Nível B	R\$ 900,00	2	R\$ 1.800,00
Apoio Técnico à Pesquisa - Nível B	R\$ 900,00	2	R\$ 1.800,00
Apoio Técnico à Pesquisa - Nível B	R\$ 900,00	2	R\$ 1.800,00
Apoio Técnico à Pesquisa - Nível B	R\$ 900,00	2	R\$ 1.800,00
Apoio Técnico à Pesquisa - Nível B	R\$ 900,00	2	R\$ 1.800,00
Apoio Técnico à Pesquisa - Nível B	R\$ 900,00	2	R\$ 1.800,00
Apoio Técnico à Pesquisa - Nível B	R\$ 900,00	2	R\$ 1.800,00
Apoio Técnico à Pesquisa - Nível B	R\$ 900,00	2	R\$ 1.800,00
Apoio Técnico à Pesquisa - Nível B	R\$ 900,00	2	R\$ 1.800,00
Apoio Técnico à Pesquisa - Nível B	R\$ 900,00	2	R\$ 1.800,00
Apoio Técnico à Pesquisa - Nível B	R\$ 900,00	2	R\$ 1.800,00
Apoio Técnico à Pesquisa - Nível B	R\$ 900,00	2	R\$ 1.800,00
Apoio Técnico à Pesquisa - Nível B	R\$ 900,00	2	R\$ 1.800,00
Apoio Técnico à Pesquisa - Nível B	R\$ 900,00	2	R\$ 1.800,00
Total	-	-	R\$ 154.400,00

TABELA 6

Auxílio-Financeiro a Pesquisadores em 2013

33.90.20 - Auxílio-Financeiro a Pesquisadores (2016)			
Classificação	Valor	Meses	Valor total
Pesquisador Sênior - Nível A	R\$ 7.000,00	10	R\$ 70.000,00
Pesquisador Sênior - Nível A	R\$ 7.000,00	10	R\$ 70.000,00
Pesquisador Sênior - Nível B	R\$ 5.000,00	10	R\$ 50.000,00
Pesquisador Sênior - Nível C	R\$ 4.000,00	10	R\$ 40.000,00
Pesquisador Sênior - Nível C	R\$ 4.000,00	10	R\$ 40.000,00
Pesquisador Sênior - Nível C	R\$ 4.000,00	10	R\$ 40.000,00
Pesquisador -Pesquisa, Desenvolvimento e Inovação - PDI - Nível C	R\$ 2.500,00	10	R\$ 25.000,00
Pesquisador -Pesquisa, Desenvolvimento e Inovação - PDI - Nível D	R\$ 2.000,00	10	R\$ 20.000,00
Pesquisador -Pesquisa, Desenvolvimento e Inovação - PDI - Nível C	R\$ 2.500,00	10	R\$ 25.000,00
Pesquisador -Pesquisa, Desenvolvimento e Inovação - PDI - Nível D	R\$ 2.000,00	10	R\$ 20.000,00
Pesquisador -Pesquisa, Desenvolvimento e Inovação - PDI - Nível D	R\$ 1.800,00	10	R\$ 18.000,00
Pesquisador -Pesquisa, Desenvolvimento e Inovação - PDI - Nível D	R\$ 1.800,00	10	R\$ 18.000,00
Pesquisador -Pesquisa, Desenvolvimento e Inovação - PDI - Nível D	R\$ 1.800,00	10	R\$ 18.000,00
Apoio Técnico à Pesquisa - Nível A	R\$ 1.200,00	10	R\$ 12.000,00
Apoio Técnico à Pesquisa - Nível A	R\$ 1.200,00	10	R\$ 12.000,00
Apoio Técnico à Pesquisa - Nível B	R\$ 1.200,00	10	R\$ 12.000,00
Apoio Técnico à Pesquisa - Nível B	R\$ 1.000,00	10	R\$ 10.000,00
Apoio Técnico à Pesquisa - Nível B	R\$ 1.000,00	10	R\$ 10.000,00
Apoio Técnico à Pesquisa - Nível B	R\$ 1.000,00	10	R\$ 10.000,00
Apoio Técnico à Pesquisa - Nível B	R\$ 1.000,00	10	R\$ 10.000,00
Apoio Técnico à Pesquisa - Nível B	R\$ 1.000,00	10	R\$ 10.000,00
Apoio Técnico à Pesquisa - Nível B	R\$ 1.000,00	10	R\$ 10.000,00
Total	-	-	R\$ 550.000,00

TABELA 9

Auxílio-Financeiro a Pesquisadores em 2016

33.90.36 – Outros Serviços de Terceiros – Pessoa Física			
Item	Valor Unitário	Quantidade	Total
Serviços técnicos especializados Nivel A	R\$ 4.200,00	3	R\$ 12.600,00
Serviços técnicos especializados Nivel A	R\$ 4.200,00	3	R\$ 12.600,00
Total	R\$ 8.400,00	-	R\$ 25.200,00
33.91.47 – Obrigações Tributárias e Contributivas			
Item	Valor Unitário	Quantidade	Total
INSS - Patronal A	R\$ 840,00	3	R\$ 2.520,00
INSS - Patronal A	R\$ 840,00	3	R\$ 2.520,00
Total	R\$ 1.680,00	-	R\$ 5.040,00
33.90.39 – Outros Serviços de Terceiros - Pessoa Jurídica			
Item	Valor Unitário	Quantidade	Total
Consultoria PJ – A	R\$ 24.000,00	1	R\$ 24.000,00
Consultoria PJ – A	R\$ 30.000,00	1	R\$ 30.000,00
Total	R\$ 54.000,00	-	R\$ 54.000,00

TABELA 10

Outros Serviços: Pessoa Jurídica, Física e Obrigações tributárias

33.90.30 – Material de Consumo			
Item	Valor Unitário	Quantidade	Total
Material de Consumo (material de escritório, equipamentos não duráveis etc)	R\$ 4.125,00	1	R\$ 4.125,00
	R\$ 12.000,00	1	R\$ 12.000,00
	R\$ 10.400,00	1	R\$ 10.400,00
Total	-	-	R\$ 26.525,00
44.90.52 – Equipamentos e Material Permanente			
Item	Valor Unitário	Quantidade	Total
Equipamentos de laboratório de Informática e redes de comunicação - clientes	R\$ 6.000,00	18	R\$ 108.000,00
Equipamentos de laboratório de Informática e redes de comunicação – smartphones	R\$ 1.200,00	5	R\$ 6.000,00
Equipamentos de laboratório de Informática e redes de comunicação – tablets	R\$ 1.200,00	5	R\$ 6.000,00
Equipamentos de laboratório de informática e redes de comunicação - servidores	R\$ 19.200,00	5	R\$ 96.000,00
Total	-	-	R\$ 216.000,00

TABELA 11

Material de Consumo e Equipamentos e Material Permanente

10. Pesquisa

O presente projeto também tem o objetivo de desenvolver pesquisa aplicada nas áreas de Engenharia de Software, Arte Computacional, Design de Interfaces. Em particular, a experiência no desenvolvimento de software livre, no contexto do software público, com interface centrada no usuário, será utilizada para fomentar a pesquisa e requerer um método específico. Precisamente, os elementos do método que serão investigados neste projeto serão os seguintes: Engenharia de Software Empírica [Wohlin et al., 2000], em Design de Interface em Ambientes Governamentais e em Arte e Tecnologia. A inovação neste projeto poderá ser observada em 3 vetores:

- Arquitetura de uma plataforma livre, voltada ao desenvolvimento de software público e ativos digitais que disporá de tecnologia de integração de serviços provendo suporte à: Sistemas de Versionamento; Sistema de Gerenciamento; Plataforma de Colaboração por meio de Redes Sociais Expressivas; Monitoramento de qualidade de código-fonte e Sistema de Indexação e Busca.
- Arte da interação, que vai além da interação humano-computador, com recursos da computação pervasiva e ubíqua;
- Design de ambientes Governamentais. Interatividade a partir da participação do usuário, considerando aspectos cognitivos, sociais e afetivos;

Processo de desenvolvimento é um elemento-chave na produção de software, podendo ser responsável direto pela qualidade do produto gerado [Münch et al., 2012]. Deseja-se que tal processo leve a maior produtividade e compreensão dos aspectos intrínsecos à produção. Por outro lado, processos de desenvolvimento estão frequentemente sujeitos a restrições de plataforma, como, por exemplo, no desenvolvimento de software embarcado crítico, sistemas de informação, entre outros. No contexto deste projeto, o requisito de se utilizar softwares livres, no contexto do software público, exige que seja feita uma customização de processos de desenvolvimento de software. Em particular, investigaremos como modelos de referência baseados em metodologias ágeis poderão ser customizadas para este projeto. Diferentemente de abordagens tradicionais tais como o Processo Unificado, as metodologias ágeis focam menos em modelos e enfatizam mais código, testes e colaboração entre pessoas, que são guiados por extratos incrementais do sistema englobando um conjunto de requisitos. Neste contexto, um elemento importante da customização é a natureza inovadora do projeto. Esta natureza se apoia

no escopo parcialmente definido e implicará em uma definição de estratégias de interação peculiares entre os desenvolvedores e pesquisadores. Segundo a norma de qualidade [ISO/IEC, 2011], o uso do produto software pode ser classificado e medido por meio de sua qualidade em uso, que influencia e depende de sua qualidade externa, que por sua vez, influencia e depende de sua qualidade interna. Visando a garantia de qualidade dos produtos a serem desenvolvidos, o projeto investigará o uso de técnicas modernas de validação e verificação, como por exemplo, testes automatizados, inspeções guiadas, e análises estáticas de código. Tais técnicas serão adaptadas para o contexto específico do projeto, considerando as diferentes plataformas a serem utilizadas, além do caráter da inovação. Em nível de testes, serão explorados tanto testes unitários como teste de sistema e de usabilidade. Serão investigadas estratégias para alcançar o maior nível possível de automação de tais testes, a fim de diminuir o esforço dispendido na execução dos mesmos e, conseqüentemente, aumentar a produtividade do processo. Serão adicionalmente definidos testes de regressão com a finalidade de garantir a estabilidade das aplicações. Com relação à validação estática, serão feitas inspeções guiadas de código, estudadas e aperfeiçoadas por técnicas de análise estática. Em particular, a abordagem metodológica a ser empregada no projeto é prioritariamente de natureza empírica, qualitativa e de design. A natureza empírica decorre principalmente do foco do projeto na construção de soluções inovadoras aplicadas ao mundo real. Qualitativa porque as pesquisas envolvidas pretendem mergulhar em profundidade na compreensão dos fenômenos envolvidos no desenvolvimento da plataforma, envolvendo não só “o que” ocorre, mas também como os fenômenos ocorrem, em termos da produção de software. Todavia, o projeto envolverá também a realização de estudos quantitativos para descrever e avaliar outras técnicas e ferramentas empregadas, barreiras e facilitadores. Finalmente, a abordagem metodológica é também de design citehevner2004 porque o projeto almeja produzir artefatos sob a forma de um guia metodológico. Abaixo apresentamos uma lista que resume a abordagem metodológica a ser seguida nas atividades de pesquisa do projeto:

- **Concepção Filosófica:** Pragmática [Easterbrook et al., 2007].
- **Método Científico:** Indutivo [Marconi and Lakatos, 2004].
- **Abordagem:** Qualitativa [Monteiro et al., 2011, Wohlin et al., 2000].
- **Métodos de Pesquisa:** Revisão de literatura [Kitchenham et al., 2009]; Pesquisa-Ação [Merriam, 2009]; Estudo de Caso; Design Science [Hevner et al., 2004].
- **Procedimentos de Coleta de Dados:** Observação participante [Merriam, 2009, Wohlin et al., 2000].

- **Procedimentos de Análise:** Classificação temática e análise qualitativa de entrevistas, documentos e registros de observação participante [[Merriam, 2009](#)].

Em termos de fases da pesquisa, inicialmente, serão conduzidas revisões de literatura. Em seguida, serão definidos modelos de customização do processo de software contemplando as particularidades do projeto. Isto será feito usando-se uma abordagem visando a utilização sistemática de métodos ágeis de desenvolvimento de software. Nessa fase, o método de pesquisa usado será Pesquisa-Ação, considerando-se todas as etapas do desenvolvimento. Posteriormente, será conduzido uma análise mais aprofundada dos resultados do estudo de forma a estabelecer conclusões fundadas no método científico.

11. Declaração

Na qualidade de representante legal do Centro de Apoio ao Desenvolvimento Tecnológico - CDT, declaro, para os devidos fins e sob as penas da lei, que inexistem quaisquer débitos em mora ou situação de inadimplência com o Tesouro Nacional, ou qualquer órgão ou entidade da Administração Pública Federal, que impeça a transferência de recursos oriundos de dotações consignadas nos orçamentos da União, na forma deste Plano de Trabalho.

Pede deferimento.

Paulo Anselmo Ziani Suarez

Diretor

Centro de Apoio ao Desenvolvimento Tecnológico

(Local e Data)

12. Aprovação

Aprovo este Plano de Trabalho pela Secretaria de Logística e Tecnologia da Informação do Ministério do Planejamento, Orçamento e Gestão.

Loreni F. Foresti
Secretária de Logística e Tecnologia da Informação
Ministério do Planejamento, Orçamento e Gestão (Local e Data)

Aprovo este Plano de Trabalho pelo Centro de Apoio ao Desenvolvimento Tecnológico.

Paulo Anselmo Ziani Suarez
Diretor
Centro de Apoio ao Desenvolvimento Tecnológico (Local e Data)

Referências

- [5cqualibr, 2010] 5cqualibr (2010). Desenvolvimento Ágil.
- [Cohn, 2005] Cohn, M. (2005). *Agile Estimating and Planning*. Prentice Hall PTR, Upper Saddle River, NJ, USA.
- [Easterbrook et al., 2007] Easterbrook, S., Singer, J., Storey, M., and Damian, D. (2007). *Selecting Empirical Methods for Software Engineering Research*. Springer.
- [Fitzgerald et al., 2006] Fitzgerald, B., Hartnett, G., and Conboy, K. (2006). Customising agile methods to software practices at intel shannon. *Eur. J. Inf. Syst.*, pages 200–213.
- [Hevner et al., 2004] Hevner, A. R., March, S. T., Park, J., and Ram, S. (2004). Design science in information systems research. *MIS Q.*, 28(1):75–105.
- [Iamamoto and Gaillard, 1999] Iamamoto, E. and Gaillard, R. (1999). A revolução digital e a sociedade do conhecimento.
- [IEEE, 1998] IEEE, S. E. S. C. (1998). *IEEE Standard for a Software Quality Metrics Methodology*. IEEE Computer Society, Piscataway, NJ, USA.
- [ISO/IEC, 2011] ISO/IEC (2011). ISO/IEC 25023 - Systems and software engineering - Systems and software Quality Requirements and Evaluation (SQuaRE) - Measurement of system and software product quality. Technical report.
- [Katayama, 2011] Katayama, E. T. (2011). A contribuição da indústria da manufatura no desenvolvimento de software. *Dissertação (Mestrado em Ciência da Computação) - Instituto de Matemática e Estatística, Universidade de São Paulo*.
- [Kitchenham et al., 2009] Kitchenham, B., Pearl Brereton, O., Budgen, D., Turner, M., Bailey, J., and Linkman, S. (2009). Systematic literature reviews in software engineering - a systematic literature review. *Inf. Softw. Technol.*, 51(1):7–15.
- [Kon et al., 2011] Kon, F., Lago, N., Meirelles, P., and Sabino, V. (2011). *Atualizações em Informática (SBC)*, chapter Software Livre e Propriedade Intelectual: Aspectos Jurídicos, Licenças e Modelos de Negócio, pages 59–107. PUC-Rio, Rio de Janeiro.
- [Lorenz, 1995] Lorenz, E. N. (1995). *The Essence Of Chaos*. Jessie and John Danz lectures. Taylor & Francis.
- [Marconi and Lakatos, 2004] Marconi, M. d. A. and Lakatos, E. M. (2004). *Metodologia Científica*. Atlas.

- [Martins, 2007] Martins, J. C. C. (2007). *Técnicas Para Gerenciamento de Projetos de Software*. Brasport, Rio de Janeiro, 1 edition.
- [Meirelles, 2013] Meirelles, P. R. M. (2013). *Monitoramento de métricas de código-fonte em projetos de software livre*. PhD thesis, Instituto de Matemática e Estatística, Universidade de São Paulo.
- [Melo et al., 2012] Melo, C., Santos, V., Corbucci, H., Katayama, E., Goldman, A., and Kon, F. (2012). Métodos ágeis no brasil: estado da prática em times e organizações. *Relatório Técnico RT-MAC-2012-03*. Departamento de Ciência da Computação. IME-USP.
- [Merriam, 2009] Merriam, S. (2009). *Qualitative Research: A Guide to Design and Implementation*. Jossey-Bass higher and adult education serie. John Wiley & Sons.
- [Monteiro et al., 2011] Monteiro, C. V., da Silva, F. Q., dos Santos, I. R., Farias, F., Cardozo, E. S., do A. Leitão, A. R., Neto, D. N., and Pernambuco Filho, M. J. (2011). A qualitative study of the determinants of self-managing team effectiveness in a scrum team. In *Proceedings of the 4th International Workshop on Cooperative and Human Aspects of Software Engineering*, CHASE '11, pages 16–23, New York, NY, USA. ACM.
- [Münch et al., 2012] Münch, J., Armbrust, O., Kowalczyk, M., and Soto, M. (2012). *Software Process Definition and Management*. The Fraunhofer IESE Series on Software and Systems Engineering.
- [Poppendieck and Poppendieck, 2011] Poppendieck, M. and Poppendieck, T. (2011). *Implementando o Desenvolvimento Lean de Software: Do Conceito ao Dinheiro*.
- [Sato et al., 2005] Sato, C. E. Y., Dergint, D. E. A., and Hatakeyama, K. (2005). O papel evolutivo do gerente de projetos. *XI Seminário Latino-Iberoamericano de Gestión Tecnológica*.
- [Schwaber, 2010] Schwaber, K. (2010). What is scrum? *VOLARO With AgileSparks*.
- [Schwaber and Beedle, 2001] Schwaber, K. and Beedle, M. (2001). *Agile Software Development with Scrum*. Prentice Hall PTR, Upper Saddle River, NJ, USA, 1st edition.
- [Silva, 2001] Silva, R. O. (2001). *Teorias da Administração*. Pioneira Thomson Learning, São Paulo.
- [Venturelli, 2004] Venturelli, S. (2004). *Arte: espaço-tempo-imagem*. Editora UnB.
- [West and Grant, 2010] West, D. and Grant, T. (2010). Agile Development: Mainstream Adoption Has Changed Agility – Trends In Real-World Adoption Of Agile Methods. Technical report, Forrester Research.
- [Wohlin et al., 2000] Wohlin, C., Runeson, P., Höst, M., Ohlsson, M. C., Regnell, B., and Wesslén, A. (2000). *Experimentation in Software Engineering: An Introduction*. Kluwer Academic Publishers, Norwell, MA, USA.