

Ideal Day e Priorização

Métodos Ágeis no Planejamento



Isabella Fonseca

isabella@powerlogic.com.br

Atua desde 1999 como consultora em e-Business para grandes empresas utilizando Java EE. É Certified ScrumMaster e gerente da equipe de desenvolvimento e de projetos do eCompany Portal - primeira solução de EIP (Enterprise Information Portal) do país, e do eCompany Process - solução de definição e controle de Processos Corporativos integrada ao Gerenciamento de Projetos (EPM e APM), de Requisitos e de Produtos (Application Lifecycle Management). Ambos são gerenciados com SCRUM e certificados MPS.Br nível F. Integrante da equipe de SEPG (Software Engineering Process Group) da Powerlogic Consultoria e Sistemas.



Fernanda Alves

fernanda.alves @powerlogic.com.br

Consultora em gerenciamento de projetos (APM e EPM) e processos. Atua em levantamento, análise, desenho e descrição de processos corporativos para modelos de melhoria de processos utilizando metodologias diversas. Integrante da equipe de SEPG (Software Engineering Process Group) da Powerlogic Consultoria e Sistemas.



Márcia Alves

marcia.alves@powerlogic.com.br

Graduanda em Administração de empresas com habilitação em Marketing, pelo Unicentro Newton Paiva, é consultora em Gerenciamento de Processos. Atualmente é Gerente de Projetos para implantação de melhorias - para alcance do nível de maturidade C pelo modelo MPS.BR (Melhoria de Processos do Software Brasileiro)

utilizando metodologia SCRUM. Integrante da equipe de SEPG (Software Engineering Process Group) da Powerlogic Consultoria e Sistemas.

Neste artigo, trataremos sobre o método de estimativa Ideal Day (ID) e uma forma de priorização de trabalho dos requisitos relativos a um projeto (Release), baseando-nos em estudos e aplicação na área de Pesquisa e Desenvolvimento da Powerlogic Consultoria e Sistemas. As estimativas de tamanho de cada funcionalidade/requisito serão apresentadas utilizando o método acima, criando indicadores para seu gerenciamento e acompanhamento e ainda servindo para realimentar ciclos (Sprints) futuros com dados estatísticos.

O que é Ideal Day?

Imagine uma parede 1mx1m. Independente de quem a concebeu, ela continuará tendo o mesmo tamanho. Por outro lado, seu tempo de entrega, variará de acordo com o executor designado para tal. Ideal Day funciona da mesma maneira.

Um Ideal Day corresponde à quantidade de trabalho que um profissional de nível sênior, com fluência nas tecnologias e ferramentas envolvidas (Ideal Developer) consegue realizar, em 08 (oito) horas de trabalho dedicadas (sem interrupções).

É importante que se compreenda que o "Dia Ideal", com 08 (oito) horas de trabalho sem interrupções, de um "desenvolvedor ideal", raramente irá ocorrer na prática, e portanto deve ser utilizado unicamente como "moeda" estável para quantificação de tamanho de referência e balizador ideal de produtividade.

É uma estimativa empírica, executada por especialistas ("Expert Judgment") para desenvolvimento com base em "exploração adaptativa". Segundo estudos mais recentes da escola ágil, a estimativa empírica é uma maneira sensata de se

prever o tamanho de requisitos em uma dinâmica de "requisitos evolucionários", com práticas de "exploração e adaptação", especialmente se acompanhada por:

- Realimentação iterativa da "velocidade", a partir de dados históricos, preferencialmente, coletados durante o mesmo projeto, para a mesma equipe.
- Previsão sobre uma mesma "ordem de grandeza", neste caso que não ultrapasse o espaço de algumas horas para alguns poucos dias.
- Realização de consenso entre especialistas, com técnicas de comunicação e convergência como a do Pocker Planning
- Utilização da técnica de PERT (Program Evaluation and Review Technique)
- Utilização de balanceamento como a técnica Cocomo (COnstructive COst Model)

Irão contribuir para que um "Ideal Day" não aconteça, na prática, em um dia típico:

- Natureza humana do desenvolvedor (comer, beber, alongar, socializar, sono, mal-estar eventual, etc.)
- Deficiências técnicas do desenvolvedor (desconhecimentos do assunto ou tecnologia específicos)
- Interrupções da empresa (reuniões administrativas, conversa com o 'chefe', ligações de clientes)
- Interrupções pessoais
- etc..

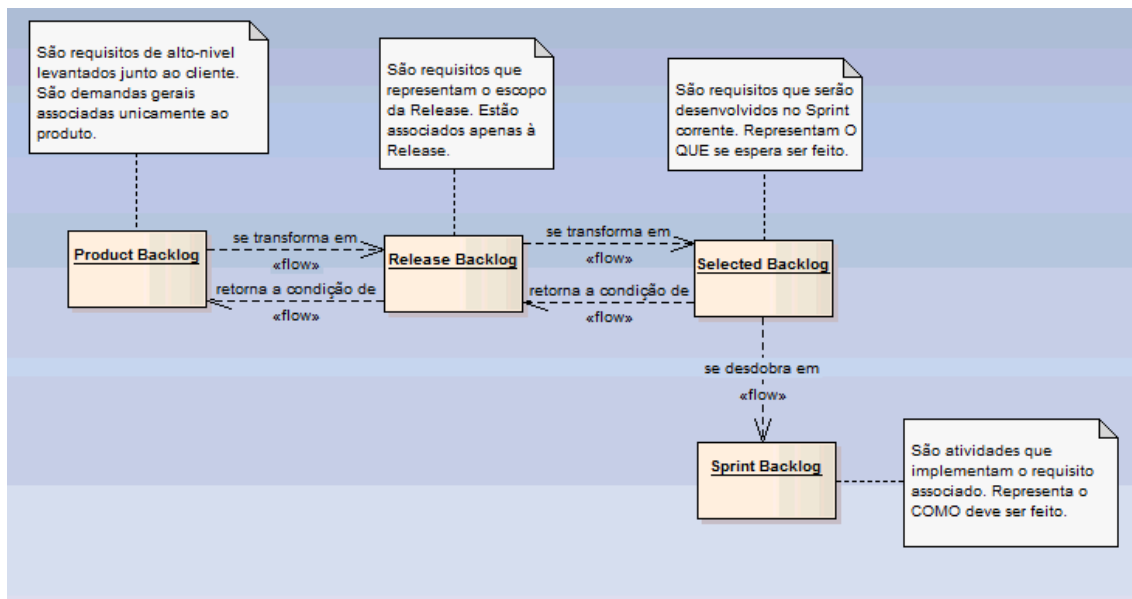
Dessa forma, a equipe deve ter sua velocidade medida pelo tempo gasto para se implementar um Ideal Day. Quanto menos tempo, maior a velocidade, e maior a produtividade da mesma. Na realidade, não é importante conhecer a

velocidade individual e sim a média da equipe. Para se manter uma unidade, não é interessante expor se um integrante executa suas atividades em mais ou menos tempo. É uma dinâmica do grupo! Ele deve aprender como interagir melhor para a busca de entrega de maior retorno de valor para o cliente. Se for necessário utilizar de técnicas como pair-programming para agilizar o desenvolvimento e validação de um requisito, o time deve escolher este caminho. Também pode-se utilizar peer-review para verificações e validações, assumir outro papel (trocar de “chapéu” dentro da equipe), etc para convergir para o objetivo definido.

Ideal Day e SCRUM

O SCRUM é um framework de processo ágil utilizado para gerenciar e controlar o desenvolvimento de um produto de software através de práticas iterativas e incrementais. É composto por um conjunto de boas práticas de gestão que admite ajustes rápidos, acompanhamento e visibilidade constantes e planos realísticos. Por ter ciclos (Sprints) curtos, sua natureza leva a utilização de requisitos de granularidade pequena, aplicando o conceito de pilha – não há alocação prévia de recurso - desconsiderando a velocidade individual e, portanto, favorecendo o uso de métodos de estimativa, como Ideal Day.

Seus artefatos principais são o Product Backlog e Sprint Backlog – artefatos que representam seus requisitos/atividades além de Burndown charts e impediment backlogs. Para representar o ciclo de vida de um requisito, usa-se derivações do Product Backlog, como Release Backlog e Selected Backlog. A figura 1 exemplifica o ciclo mencionado.



- Figura 1 -

O método de estimativa utilizado para o Selected Backlog reúne técnicas modernas voltadas à estimativa de requisitos em projetos cujo índice de imprevisibilidade é notadamente alto, seja por envolverem tecnologias de ponta, inovações que requerem desenvolvimento "iterativo", baseado em "exploração e adaptação", ou mesmo ineditismo do que se está por produzir - ou seja, previsão para 'criação de novo produto', e não para 'manufatura de mais do mesmo produto'.

Realimentação iterativa da "velocidade"

A velocidade deve ser reajustada, a cada final de Sprint, para um ajuste apropriado que maximize a previsibilidade. Quanto maior o tamanho de um requisito a ser estimado, mais aumentam as possibilidades de desvios, tanto relativas (maior percentual de erro), quanto absolutas (maior tamanho do erro, para um mesmo percentual). Por isso, em um processo empírico gerenciado, deve-se "decompor" itens do Selected Backlog que ultrapassem 01 Ideal Day em itens do Sprint Backlog, rastreáveis entre si, mas de implementação

independente. Estes itens podem ser chamados no XP de "estórias do usuário" ou similares a "quebrar cenários de Caso de Uso" no Processo Unificado.

Portanto, a velocidade é calculada através do número de horas que a equipe como um todo gasta para implementar um trabalho equivalente a 01 (um) Ideal Day.

A reunião de planejamento de uma Release ou Sprint, deve contemplar a estimativa dos requisitos por meio da técnica de Poker Planning.

A prática do Poker Planning é a seguinte:

1. Todos devem possuir "cartas" contendo os intervalos discretos de previsão, e mais uma de estouro, contendo "?", para indicar o julgamento do item do "Selected Backlog acima da Ordem de Grandeza".

2. O Selected Backlog deve ser lido e discutido por todos, e em seguida, sem comentarem seus votos, todos devem apresentar as cartas com a previsão que julgam de maior aproximação

3. Caso não haja uma convergência óbvia (média aproximada entre todos), ou caso alguém apresente o "?", deve-se rediscutir o requisito, principalmente ouvindo-se os argumentos daqueles que votaram com maior desvio.

Em função da discussão, pode-se:

- 3.1 Melhorar a especificação do item do Selected Backlog.

- 3.2. Decompor o item do Selected Backlog, mantendo a rastreabilidade.

- 3.3. Simplesmente prestar-se mais esclarecimentos aos votantes

4. Por fim, deve-se proceder com uma nova votação, e retornar ao passo 3, até que todos entrem em um consenso sobre o valor.

A previsão deve ser realizada em escalas discretas, entre 1/8, 1/4, 1/2 e 1 Ideal Day. Não há necessidade de se expressar refinamentos nesta escala, pois a margem de erro é considerada maior que um possível refinamento da aproximação.

Outro ajuste importante para aprimoramento de previsão empírica é o uso por todos os envolvidos de previsões dos limites, e não da ideal. Ou seja, a estimativa é realizada a partir das seguintes questões:

1. Qual o tamanho mínimo em Ideal Days do item do Selected Backlog considerando-se, por exemplo, um baixo impacto ou instabilidade provocados pela sua implementação? – Melhor Caso.
2. Qual o tamanho máximo em Ideal Days do Selected Backlog considerando-se um maior impacto e instabilidade? – Pior Caso.

Este tipo de análise contribui para a reflexão sobre situações limítrofes para pior ou para melhor, deste modo aprimorando o resultado ideal.

Aliado a isso, técnicas como PERT, abordam uma terceira medida, mais provável, que garantem, estatisticamente, maior assertividade nas estimativas executadas. Portanto, consideramos, como melhor prática, o alinhamento entre Ideal Day e PERT. Ainda, é aconselhável utilizar matrizes de rastreabilidade de requisitos para que se tenha uma dimensão do impacto (número e artefatos afetados) do Selected Backlog a cada estimativa ou alteração no requisito.

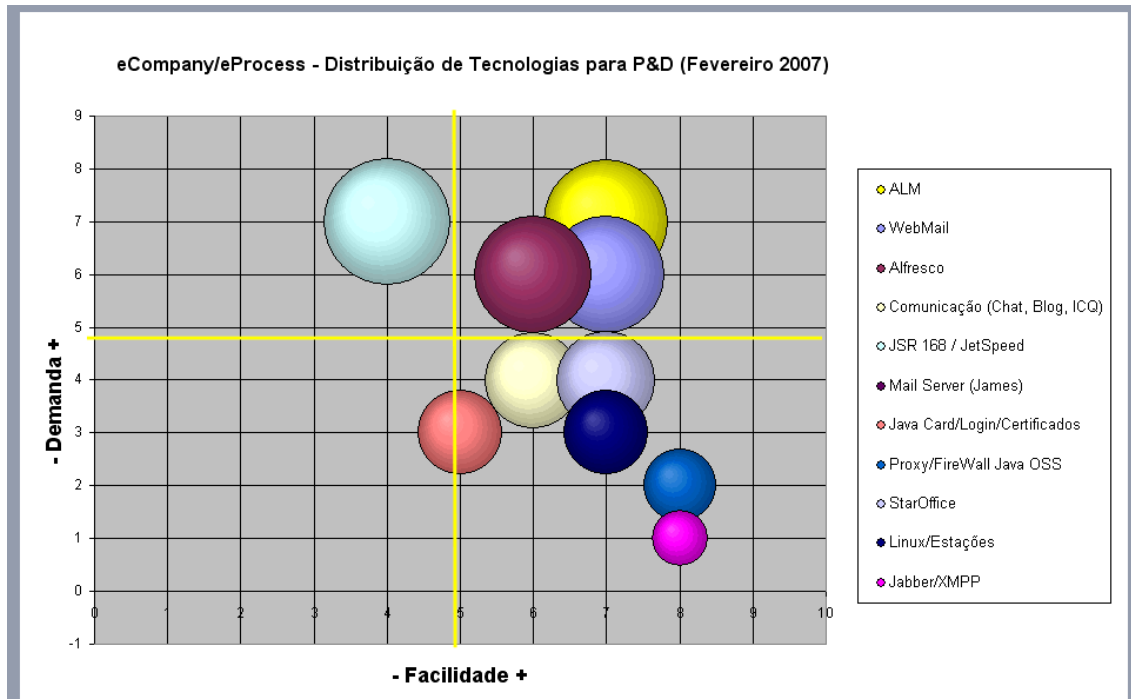
Etapas de priorização da pilha de Product Backlog utilizando Business Value

Business Value (BV) ou Valor de Negócio reflete a importância estratégica de uma funcionalidade do produto para o mercado. O responsável pelo produto deve manter a lista de Product Backlog constantemente atualizada e avaliada com seu business value atribuído, podendo ser revisado a qualquer momento. Esta prática faz parte da primeira etapa de priorização, que permite ao responsável classificar os itens de maior valor para o mercado e obtenha maior retorno para o negócio sobre o investimento. Ao iniciar uma Release, será possível identificar e selecionar os possíveis itens do Product Backlog que farão parte do escopo a ser desenvolvido. É interessante estipular um valor limite para a distribuição entre os itens da pilha. Cada funcionalidade deve ter seu valor compreendido na escala estabelecida pela organização, por exemplo, de 0 a 100.

Outra variável que deve ser inserida na fórmula de priorização dos itens que irão compor o escopo da Release (Release/Selected Backlog) é a facilidade de implementação do requisito, executando assim a segunda etapa de priorização. Quanto maior a facilidade, menor deve ser seu esforço de implementação. A figura 2 exibe um gráfico de quatro quadrantes de itens do escopo que foram classificados em função seu BV e sua facilidade de implementação. Itens que se concentram no quadrante superior direito são os que devem ser priorizados e, seguramente, são os que mais representam a maximização do resultado.

As coordenadas do gráfico seguem os valores de BV determinados pelo responsável do produto e a facilidade de implementação estimada em consenso pela equipe de desenvolvimento. O tamanho de cada bolha representa

proporção de valor de negócio de cada item considerando a lista em que ele está inserido. Caso seja necessário, cabe a cada organização determinar um diferente “peso” para esta representação.



- Figura 2 -

A última etapa de refinamento da priorização de requisitos leva em consideração o tamanho do requisito estimado pela equipe de desenvolvimento. Ela organiza a ordem de execução dos requisitos da pilha selecionados na segunda etapa e norteia a equipe de desenvolvimento. Segundo o SCRUM, a própria equipe tem autonomia para tal atividade e esta prática garante o comprometimento dos envolvidos para o alcance dos objetivos.

Dado isso, a fórmula final deve ser a seguinte:

Priorização final da pilha = $BV / \text{Tamanho do requisito}$

Em caso de empate – dois requisitos com mesmo valor final de priorização – deve-se considerar primeiramente seu BV como critério de desempate e, como segundo critério, a criticidade sendo:

1 – Baixa

3 – Normal

5 – Alta

7 – Urgência

9 – Emergência

A Fórmula de priorização para desempate deve ser representada como $(BV/Tamanho\ do\ requisito) * criticidade$.

Outro ponto a ser destacado e comum em projetos de manutenção de produtos, é a presença de erros que deverão entrar na pilha do Product Backlog. Pode-se dar um “peso” e modificar a fórmula acima para contemplar este cenário. Uma vez que BV diz respeito a valor de negócio que a inclusão da funcionalidade irá prover para o mercado e erro não acrescenta valor ao produto, sugerimos determinar BV negativo para este caso. Para que este requisito entre corretamente na priorização da pilha, deve-se aplicar seu valor absoluto.

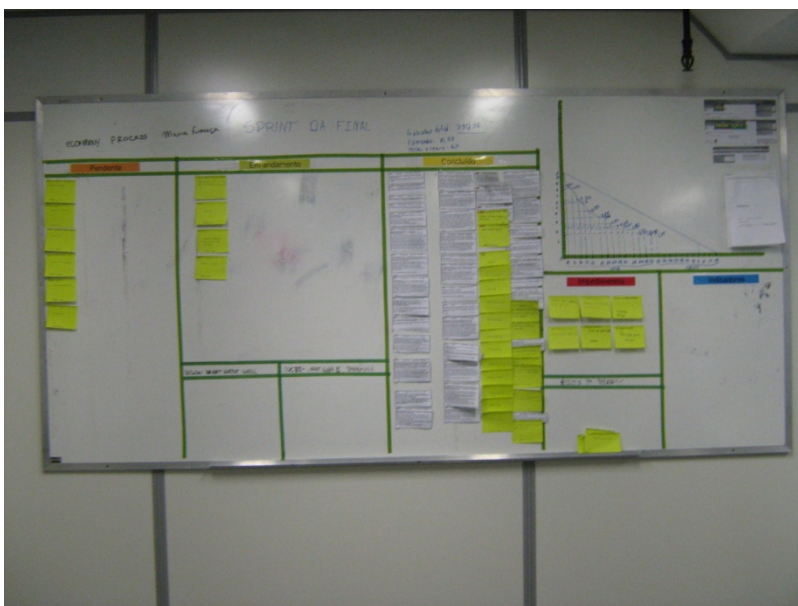
A partir deste resultado, a equipe de desenvolvimento consegue determinar prazos para cada entrega que contempla o escopo acordado. O número de Sprints de implementação necessários é então comunicado ao responsável pelo produto. A partir da definição de sua restrição – escopo, tempo, custo ou qualidade - ele tomará uma decisão e definirá o objetivo maior da Release e do primeiro Sprint. O SCRUM prega o planejamento contínuo, portanto, as demais iterações serão planejadas oportunamente.

Após esta definição, a equipe de desenvolvimento consegue implementar os requisitos seguindo, rigorosamente, do topo para baixo da pilha.

Em sistema enxutos (Lean) de "push", como o Scrum, resolve-se o problema de dependências de "caminho crítico" simplesmente ajustando-se a ordem de execução dos itens de Product Backlog: os que dependem são colocados abaixo de suas dependências.

Agile Radiator

A figura 3 exemplifica um Agile Radiator monitorando um projeto real. Eles garantem visibilidade do projeto a todos os envolvidos. Não há como mascarar o real andamento. O goal fica afixado e os requisitos – através de Post-its - (Selected backlogs) e seus desdobramentos (Sprint backlogs) são posicionados na situação onde se encontram (se ainda não iniciados - planejados, se sendo executados no momento - em andamento e se terminados – 100% concluídos). Eles devem ser posicionados de acordo com a prioridade dos mesmos – Business Value declarado pelo Product Owner. Post-its localizados no topo nos dizem ser de maior prioridade que os posicionados no rodapé do quadro branco.



- Figura 3 –

Caso prático:

Se sua equipe irá trabalhar 4 horas por dia, você devera estipular sua baliza, ou seja, 1 ID = 4 horas e ainda determinar a velocidade média inicial de sua equipe, por exemplo, ID = 10 horas. A primeira estimativa da velocidade da equipe é realmente empírica, as demais serão baseadas em dados históricos.

1) Abaixo, uma lista resumida de itens do Selected Backlog já estimados via Ideal Day durante o Poker Planning.

- a. RF01 – Implementar carrinho de compras – 0,5 ID
- b. RF02 - Cadastrar livros – 0,3 ID
- c. RF03 – Consultar livros por autor– 0,1 ID
- d. RF04 - Implementar recomendação automática de livros – 0,4 ID

2) Pelo conceito de pilha, cada membro deve retirar uma tarefa para executar e apropriar as horas gastas ao final do dia. Este procedimento é necessário para que, ao final do Sprint, seja possível determinar quantos Ideal Days foram concluídos e o seu tempo de execução.

Note que o tamanho de um requisito nunca muda! O que muda é o esforço do mesmo, que depende do recurso alocado - velocidade.

O recurso 1 irá implementar o RF01, que tem 0,5 ID de tamanho. Se é a primeira vez e não tenho dados históricos para determinar o esforço do recurso, vou utilizar a velocidade inicial determinada empiricamente de 10 horas.

Por isso, o requisito terá duração prevista de 5 horas.

Esforço = tamanho x velocidade

Esforço = 0,5 x 10 = 5 horas

De acordo com o número de horas que a pessoa trabalha (se ele trabalha somente pela manhã), será necessário “quebrar” este requisito para que seja

possível concluí-lo em 1 dia de trabalho e mover o post-it de “em andamento” para “finalizado” evidenciando a evolução do trabalho.

Desse modo, atualizando o gráfico Burndown através da subtração do tamanho (0,5 ID) de Ideal Days realizados, será possível acompanhar desvios entre previsto e realizado de maneira efetiva e visual.

Com a apropriação do Sprint, obtém-se dados para calcular a velocidade padrão do grupo. A tabela 1 exemplifica os dados apurados após implementação.

Requisito	Tamanho Previsto	Recurso	Horas real
RF01	0,5 ID	Recurso 1	4,5 horas
RF02	0,3 ID	Recurso 2	2,5 horas
RF03	0,1 ID	Recurso 1	1,5 horas
RF04	0,4 ID	Recurso 2	3 horas
TOTAL:	1,3 ID		11,5 horas

- Tabela 1 -

O recurso 1 implementou os requisitos RF01 e RF03, totalizando entrega de 0,6 ID. Já o recurso 2, os RF02 e RF04, totalizando 0,7 ID.

Para calcular a nova velocidade, após o Sprint:

	Velocidade inicial	ID previsto	ID realizado	Horas real
Recurso 1	10	0,6	0,6	6
Recurso 2	10	0,7	0,7	5,5

As horas de retrabalho serão consideradas, mais tarde, após testes do controle de qualidade, e terão um peso maior no cálculo da velocidade da equipe.

Fórmula para cálculo da velocidade = $\frac{\text{horas_realizadas} + (\text{horas_retrabalho} * 1,3)}{\text{ID realizados}}$

Recurso 1 = $6 + 0 / 0,6 = 10$ horas

Recurso 2 = $5,5 + 0 / 0,7 = 7,8$ horas

Média da equipe = 8,9 horas.

No próximo Sprint, a média a ser considerada será a calculada no Sprint anterior e não mais empiricamente.

A produtividade é extraída da avaliação do número de Ideal Days/Sprint. No primeiro Sprint, a produtividade é igual ao número de Ideal Days entregues.

Produtividade da equipe = 1,3 ID

Considerando a produtividade da equipe, no próximo Sprint será possível alocar itens do Selected Backlog que totalizem o número de Ideal Days entregues no Sprint anterior – no nosso exemplo, 1,3 ID. Ao final do mesmo, deve-se apurar novamente este número e fazer a média entre Sprints, atualizando sempre esta informação.

Você deverá medir novamente e obter mais dados históricos para conhecer a média de produtividade e de velocidade da equipe.

A produtividade da Release, pode ser calculada como :

Produtividade da Release = ID Realizados/ Número de Sprints.

Conclusão

Falar por que usamos a velocidade e quando usamos a produtividade

Bibliografia utilizada

Processo de Desenvolvimento de Software da Powerlogic Consultoria e Sistemas, versão PDS_P&D_v19.

Agile Estimating and Planning, Prentice Hall, 2006, Cohn , M.

Agile Development Blog / SCRUM,

<<http://www.targetprocess.com/blog/2004/12/iteration-velocitys-and-user-story.html>> acessado em 01/09/2008.

Mike Cohn's Blog <<http://blog.mountangoatsoftware.com/?p=15>>, acessado em 12/09/2008.

Artigo: "Por que SCRUM?", ESM, 4ª. ed. (Agosto 2008) escrito por Fonseca, I. e Campos, A.