Análise e Projeto de Sistemas

Lucas Nogueira Padrão

Análise e Projeto de Sistemas

Como analisar, planejar, desenvolver e implementar sistemas de informação

viena

1ª Edição
Santa Cruz do Rio Pardo/SP
Editora Viena
2014

Dados Internacionais de Catalogação na Publicação (CIP)
(Câmara Brasileira do Livro, SP, Brasil)

Padrão, Lucas Nogueira
Análise e projeto de sistemas : como analisar, planejar, desenvolver e implementar sistemas de informação / Lucas Nogueira Padrão.
-- Santa Cruz do Rio Pardo, SP : Editora Viena, 2014. --
(Coleção premium)

ISBN 978-85-371-0390-6

1. Banco de dados - Projeto 2. Banco de dados - Criação 3. Banco de dados - Desenvolvimento 4. Banco de dados - Estudo e ensino 5. Informática I. Título. II. Série.

14-09701 CDD-005.74

Índices para catálogo sistemático:

1. Banco de dados : Projeto : Computadores : Processamento de dados 005.74

Autor: Lucas Nogueira Padrão
Revisão Ortográfica: Tássia F. A. Carvalho
Capa: Luciane Mendonça
Diagramadora: Adriana Araújo
Revisão de Diagramação: Camila Ceccatto da Silva Perez e Graciele Alves de Mira
Supervisão Editorial: Karina de Oliveira

ISBN: 978-85-371-0390-6

1ª Edição - 12/2014 - SCRPardo / SP

Impresso no Brasil

Dedico este trabalho ao meu amigo, companheiro, afilhado e irmão mais novo, Arthur.

L.N.P.

"Todas as grandes realizações caracterizam-se pelo extremo cuidado e o infinito esmero, nos mínimos detalhes."

Elbert Hubbard

Prefácio

Para todas as coisas que queremos fazer na vida são necessários diversos passos para alcançá-las. Em geral, na nossa vida pessoal damos passos por meio da intuição e do bom senso. Contudo, quando estamos falando de um produto comercial altamente complexo, com o envolvimento de diversas pessoas e etapas, devemos desenvolver um método adequado para atingir o objetivo final.

Após a Revolução Industrial, as fábricas desenvolveram inúmeros meios para conceber, produzir e vender seus produtos com o menor gasto e a maior qualidade possíveis. Com o software não poderia ser diferente.

Para que um software atinja seu objetivo final, são necessárias diversas etapas que serão analisadas e discutidas neste trabalho. Por isso que a frase de Hubbard se encontra na epígrafe deste livro. A análise e projeto de sistemas consiste em um extremo cuidado e infinito esmero para chegar a um sistema de qualidade.

Todas as sugestões serão bem-vindas e poderão ser remetidas a llnp_6@hotmail.com.

Autor

Sumário

Lista de Siglas e Abreviaturas

XP _____ *eXtreme Programming.*
DSDM _____ *Dynamic Systems Development Method.*
UML _____ *Unified Modeling Language.*
PMBOK _____ *Project Management Body of Knowledge.*
JAD _____ *Joint Application Development.*
CEP _____ *Código de Endereçamento Postal.*
MVC _____ *Model-View-Controller.*
SGBD _____ *Sistemas Gerenciadores de Banco de Dados.*
DML _____ *Data Manipulation Language.*
DDL _____ *Data Definition Language.*
CNPJ _____ *Cadastro Nacional de Pessoa Jurídica.*
CPF _____ *Cadastro Pessoal Física.*
CRM _____ *Customer Relationship Management.*
ANSI _____ *American National Standards Institute.*
ISO _____ *International Organization for Standardization.*
SQL _____ *Structured Query Language.*
PL _____ *Procedual Language.*
QBE _____ *Query By Example.*
ISBN _____ *International Standard Book Number.*
OLAP _____ *On-Line Analytical Processing.*
OLTP _____ *Online Transaction Processing.*
KB _____ *Kilobyte.*
DARPA _____ *Defense Advanced Research Projects Agency.*
ARPANET _____ *Advanced Research Projects Agency Network.*
IMP _____ *Interface Message Processor.*
NCP _____ *Network Control Protocol.*
DNS _____ *Domain Name System.*
LAN _____ *Local Area Network.*
MAN _____ *Metropolian Area Network.*
WAN _____ *Wide Area Network.*
TCP/IP _____ *Transmission Control Protocol/Internet Protocol.*
IHL _____ *Internet Header Length.*
UDP _____ *User Datagram Protocol.*
URL _____ *Uniform Resource Locator.*
HTML _____ *HyperText Markup Language.*
RMI _____ *Remote Method Invocation.*
TI _____ *Tecnologia da Informação.*
CRC _____ *Class-Responsability-Collaborator.*
EAP _____ *Estrutura Analítica do Projeto.*
ITIL _____ *Information Tecnology Infrastructure Library.*
ITSMF _____ *Information Technology Service Management Forum*
CCTA _____ *Central Computer and Telecommunications Agency.*

1
Análise de Sistemas

1. Análise de Sistemas

Em seu livro sobre Metas, o autor Brian Tracy afirma que é a clareza de objetivo que leva à realização da meta. E de fato, sem sabermos o que queremos, jamais vamos atingir algum objetivo. Se quero comprar uma bola, devo ter isso em mente e realizar todos os processos necessários para adquiri-la. Se quero produzir um software, devo saber que tipo de software quero produzir para iniciar seu processo de desenvolvimento.

Analisar consiste em enumerar ou descrever as principais características de um objeto. Para analisar a bola que queremos comprar, deveríamos dizer quais cores estão presentes na bola, o tamanho, o formato, a rigidez. Tudo isso tendo em vista o esporte que praticaremos com a bola: futebol, vôlei, tênis, rugby etc.

A Análise de Sistemas consiste em esclarecer as características do software que queremos. Assim poderemos ter a clareza para desenvolver um software que atenda exatamente às necessidades do cliente.

Vamos imaginar um software que realiza cálculos básicos. Devemos ter em conta que o software deva calcular as operações básicas (adição, subtração, multiplicação e divisão) retornando o resultado correto. Então, a análise de software deve responder à pergunta "Quais tarefas o software precisa realizar?". Perceba que uma resposta parcial ou incorreta levará a um software que não atenda à necessidade do cliente. Se um cliente quer somar números, não faz sentido oferecer uma calculadora que calcule a raiz cúbica dos números informados (uma operação mais complexa) ainda que funcione corretamente.

Em geral, o Analista de Sistemas é uma peça fundamental, que conhece um pouco de todos os elementos envolvidos no sistema, desde pessoas até tecnologias e práticas comuns. Sua função básica consiste em colher os desejos do cliente e transformá-los em algo realizável tecnologicamente.

Os sistemas de informação têm como consequência a mudança da organização. Por isso, o analista de sistemas é um personagem envolvido em um processo delicado, o processo de transformação. Assim, eles devem compreender o ambiente técnico da organização para que o sistema projetado seja condizente com as atividades envolvidas. Desse modo, o estudo de administração de empresas e produção se torna um quesito necessário ao analista de sistemas, afinal, este é visto como um funcionário para resolver problemas empresariais.

Não raramente, os analistas precisam se comunicar diretamente com os usuários, gerentes e programadores, para que possam ter a visão integrada do projeto a ser desenvolvido. Desse modo, habilidade de comunição é imprescindível ao analista de sistemas. Porém, há diversas especializações na área de análise de sistemas.

- **Analista de Negócios**: Analisa os aspectos empresariais do sistema.
- **Analista de Sistemas**: Descobre como resolver os problemas empresariais com o sistema.
- **Analista de Infraestrutura**: Assegura o suporte do novo sistema na empresa.
- **Gerente de Mudanças**: Planeja e executa as mudanças necessárias para evitar danos e perdas empresariais.
- **Gerente de Projetos**: Gerencia a equipe com o objetivo de atingir um objetivo concreto.

1.1. Ciclo de Vida

Da mesma forma que na engenharia civil, o processo de construção de um sistema é composto por estágios. Em geral, a literatura técnica comporta quatro tipos de estágios ou fases, que podem ser imaginadas como passos para atingir o estágio final, que consiste no sistema pronto para o uso.

As quatro fases são as seguintes:

- **Planejamento**: Estágio em que as oportunidades são identificadas, o projeto é definido e o plano do projeto é designado.
- **Análise**: Fase para determinar a estratégia de desenvolvimento, recolher os requisitos e modelar os processos que serão automatizados pelo sistema.
- **Projeto**: Etapa em que todos os componentes do sistema são planejados, projetados.
- **Implementação**: Fase em que o sistema é construído, instalado e manutenido.

1.2. Planejamento

Talvez a questão mais difícil para um administrador de empresas é saber se há necessidade de construir um novo sistema para a empresa. Certamente a resposta envolve os custos da produção e o lucro ou prejuízo que poderia ser obtido. Também é necessário decidir como a equipe de desenvolvimento irá construir o determinado sistema. Para que isso seja realizado seguramente, seriam necessários dois passos.

O primeiro passo consiste em determinar e avaliar a viabilidade do sistema. Quando um departamento empresarial faz uma requisição de um sistema, em geral, suas necessidades vêm resumidas na forma de um problema a resolver. Contudo a implementação dessa solução pode ser inviável econômica, técnica ou fisicamente. Ainda há a possibilidade de o sistema não ser usado efetivamente e não produzir lucros. Desse modo, a viabilidade deve ser analisada com cuidado.

Já o segundo passo consiste em iniciar a gerência do projeto. Nessa etapa, um gerente de projeto é requisitado para controlar e dirigir o projeto, oferecendo para isso um planejamento do projeto, que descreve como os funcionários irão interagir durante o projeto e como este irá caminhar durante sua construção.

Na etapa de planejamento, o analista de sistemas e os administradores da empresa devem trabalhar juntos para conseguir delinear as necessidades e características do sistema. Embora o objetivo de todo negócio seja o lucro, é necessário investimento para que o negócio se concretize, e a área de tecnologia da informação é uma das áreas que mais carecem de investimento, atualmente. Desse modo, é preciso que o investimento retorne lucro para o proprietário do negócio, de forma que ele precisará acompanhar o sistema durante sua construção a fim de verificar se ele está sendo construído na direção certa, a de gerar lucro.

Certamente o tamanho do projeto irá interferir diretamente no investimento realizado e, de acordo com o tamanho do investimento, mais atenção requererá dos proprietários e investidores.

Como visto anteriormente, os requisitos são as funcionalidades que o sistema deverá prover. Assim, é preciso analisar quais funcionalidades são imprescindíveis para o negócio em questão. Porém, como os proprietários e investidores não detêm

conhecimento técnico, é preciso que as funcionalidades sejam explicadas em linguagem de alto nível, isto é, de forma compreensiva para os interlocutores. Assim, os investidores poderão ter uma noção concreta do valor que o desenvolvimento de um sistema poderá gerar.

Há, pelo menos, duas maneiras de classificar o valor gerado pelos sistemas de informação. Os valores tangíveis são aqueles que podem ser numericamente mensuráveis, como o aumento de 2% dos clientes. Porém, há outros valores que são intangíveis, por serem difíceis de medir numericamente, como a satisfação do cliente. Uma vez identificado o valor gerado pelo sistema e a sua importância para a empresa, o projeto pode iniciar normalmente. Na maioria das grandes empresas, a iniciação do projeto ocorre por meio da requisição de sistema.

A requisição de sistema consiste em um documento que descreve as razões para construção de um sistema específico e a previsão de geração de valor. Por meio de tal documento, o projeto é submetido à aprovação por um comitê, que pode ser formado por executivos da empresa. O comitê investigará as razões da construção do sistema e poderá aprová-la ou rejeitá-la.

1.2.1. Viabilidade

Uma vez aprovado, o projeto deve ser submetido à análise de viabilidade. O comitê requisita uma descrição mais detalhada do projeto e tenta estabelecer os limites e as oportunidades geradas por meio dele. Essa etapa, não menos importante, é chamada de estudo de viabilidade e compõe-se de três partes: análise da viabilidade técnica, análise da viabilidade financeira e análise da viabilidade organizacional.

Resumidamente, a análise de viabilidade técnica consiste em saber se é possível construir o sistema. É bem famoso o caso do diretor de cinema James Cameron, que, para realizar a filmagem da ficção "Avatar", teve de esperar mais de dez anos até que a tecnologia pudesse ser produzida. Assim, há projetos inviáveis tecnicamente, podendo-se afirmar que a análise de viabilidade técnica consiste em uma espécie de análise de risco, já que, caso não haja técnica suficiente para realizar o projeto, ele correrá sério risco de não ser finalizado.

Em geral, três fatores podem ser considerados na análise de viabilidade técnica. O primeiro deles consiste na familiaridade da equipe técnica com a tecnologia desenvolvida. Caso um sistema seja produzido em uma linguagem de programação desconhecida pela maioria dos participantes, é possível que haja desconforto no desenvolvimento do projeto, ocasionando inviabilidade técnica. O segundo consiste no tamanho do projeto. Obviamente, um projeto demasiado grande para uma organização comercial pode ser inviável. Também é preciso considerar, em terceiro lugar, a compatibilidade da tecnologia utilizada no novo projeto com as tecnologias empregadas na empresa. Um sistema de cadastro que exija plataforma Unix em uma empresa onde todos os computadores utilizam plataforma Windows é totalmente inviável, por exemplo.

A avaliação da viabilidade técnica, muitas vezes, pode se valer da experiência do analista ou das avaliações de projetos anteriores, que descrevem detalhadamente os problemas encontrados e as soluções propostas para resolvê-los.

Há também de se considerar a questão da viabilidade econômica, que pode ser brevemente descrita em termos de benefícios e custos. Existem custos de desenvolvimento e custos operacionais, além dos benefícios tangíveis e intangíveis. Os custos de desenvolvimento envolvem aqueles custos concernentes à criação do sistema, como salários e despesas de hardware. Já os custos operacionais são aqueles destinados a manter o projeto funcionando, como licensas e gastos com comunicação. Como dito acima, os benefícios tangíveis podem, ao contrário dos intangíveis, ser mensurados.

Sabemos que o homem é um ser cultural e que por causa disso adere fortemente a certos hábitos e comportamentos. Por isso, a mudança da cultura organizacional é uma das mudanças mais difíceis de realizar no mundo corporativo. Assim, é possível que a implantação de um novo sistema de informação possa afetar a cultura organizacional, isto é, os hábitos dos empregados da empresa. Isso quer dizer que muitas vezes o dinheiro gasto no desenvolvimento de um sistema pode ser desperdiçado caso os usuários se recusem a utilizá-lo. Para aferir essa possibilidade, há o que se chama de viabilidade organizacional.

A análise de viabilidade organizacional pretende avaliar se os benefícios virão realmente, caso o sistema seja implantado. É possível encontrar a viabilidade organizacional descobrindo se os benefícios do sistema estão alinhados com os objetivos empresariais naquele momento. Também é possível consultar um grupo de beneficiados ou afetados pelo sistema, bem como de seus possíveis usuários.

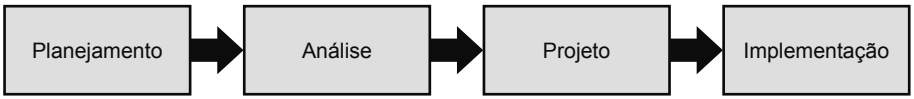
O estudo da viabilidade do projeto é de suma importância para o desenvolvimento dele. Por meio dele, é possível fazer investimentos mais seguros e proteger os profissionais de tecnologia da informação de críticas desnecessárias. É preciso enfatizar a necessidade de revisar diversas vezes o documento da análise de viabilidade.

Modernamente, a gerência de projetos considera a gerência de portfólio uma área especial. Um portfólio consiste na reunião de todos os projetos de uma organização. A gerência de portfólio considera questões como risco, importância, custo e propósito do projeto. Assim, é possível manejar os projetos, de forma que os mais urgentes sejam apressados, e os mais caros, financiados. Obviamente, a menor parte desses projetos consiste em projetos de tecnologia da informação, apesar de essa quantidade vir crescendo significativamente.

1.2.2. Metodologia de Desenvolvimento

Além da prioridade, a fase de planejamento discute qual metodologia utilizar para a construção do sistema.

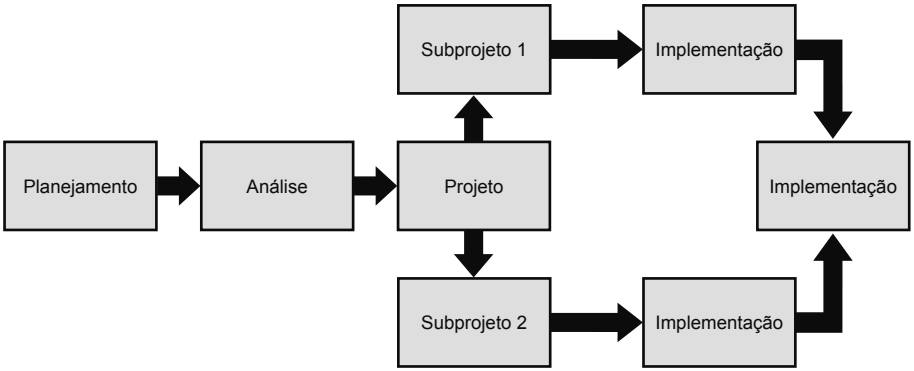
Durante muito tempo, nos primórdios da engenharia de software, era empregada a metodologia em cascata. Isso significa que as partes que constituem o projeto, planejamento, análise, projeto e implementação, são realizadas sucessivamente e uma fase só poderá ser iniciada após o término de outra.



Metodologia de desenvolvimento em cascata.

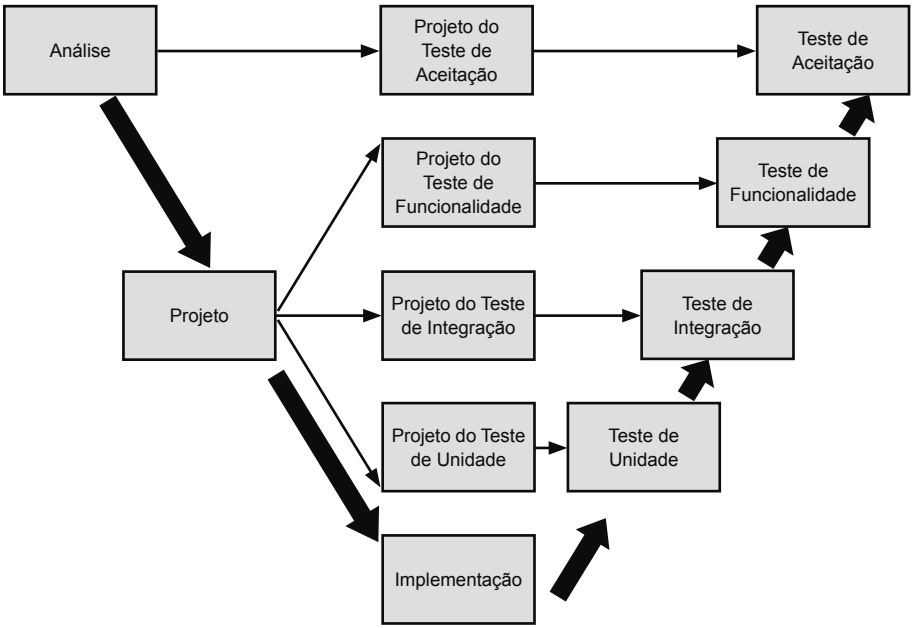
Em geral, cada entrega só era realizada após a finalização da etapa e apresentada ao comitê para aprovação. O grande problema consiste no fato de que a fase de planejamento poderia demorar meses e, caso fosse rejeitada pelo comitê de aprovação, deveria ser reconstruída desde o início. Além disso, como a fase de implementação demorava para ocorrer, muitas vezes os requisitos mudavam ou perdiam o valor que tinham inicialmente. Como solução, algumas variantes do modelo em cascata foram criadas para minimizar tais problemas.

A primeira tentativa foi a criação do modelo de desenvolvimento paralelo em que vários subprojetos eram criados a partir do planejamento e análise iniciais, permitindo terminar o projeto mais rapidamente. Contudo, como os subprojetos não eram independentes por completo, muitas decisões acarretavam consequências em outros subprojetos. Além disso, persistia o problema de que as fases iniciais eram muito longas e entregavam produtos muito volumosos de uma única vez.



Desenvolvimento paralelo.

Também há a metodologia de desenvolvimento em V. O nome advém do fato de o fluxograma aparentar o desenho de um V. O desenvolvimento em V é fundamentado no desenvolvimento em cascata, porém, com modificações. Nesse tipo de metodologia, os testes começam a ser planejados logo após a análise e não há a fase de planejamento. O fato de os testes serem projetados precocemente permite a melhoria significativa da qualidade do produto final, contudo ele ainda padece da rigidez imposta pelo desenvolvimento em cascata, do qual deriva.



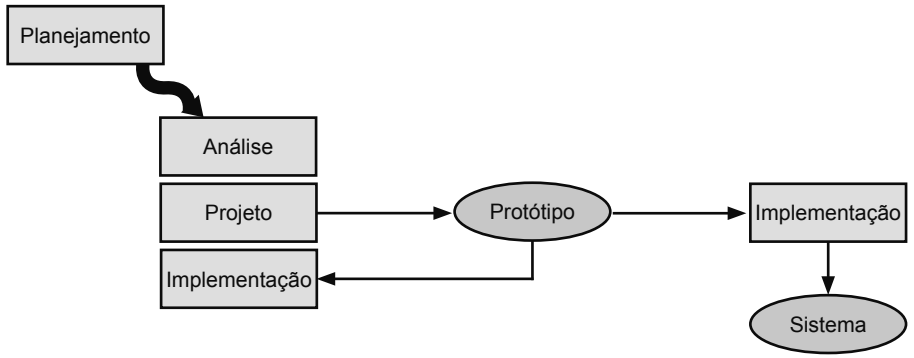
Desenvolvimento em V.

A rigidez da metodologia em cascata não é compatível com o ambiente organizacional moderno, em que as mudanças ocorrem abruptamente. Por isso, tentou-se desenvolver outras metodologias que pudessem acompanhar a evolução rápida dos negócios. Assim, foi criada a metodologia de desenvolvimento rápido de aplicações, que consiste basicamente em utilizar ferramentas computacionais para auxiliar todas as fases da produção do sistema. Essa abordagem permitiu o aumento da velocidade de produção bem como um melhor entendimento do produto pelo usuário durante a fase de produção. A melhoria obtida pela proposta foi tamanha, que hoje não se é mais possível projetar um sistema sem o auxílio dessas ferramentas.

O desenvolvimento iterativo segue a essa proposta. Ele consiste em dividir o sistema em pequenas versões que são entregues paulatinamente aos usuários. Contudo, cada versão é produzida mediante metodologia em cascata, o que ainda dificulta a flexibilidade. Apesar disso, o desenvolvimento iterativo pode oferecer uma versão do software ao cliente rapidamente, ainda que incompleta. As versões posteriores se encarregarão de concluir o restante do projeto.

Ainda entre as metodologias de desenvolvimento rápido, além do desenvolvimento iterativo, há o que se chama de prototipação, o ato de oferecer ao usuário um protótipo simplificado do sistema, o qual oferece algumas funcionalidades básicas. Por meio da reação do usuário, os desenvolvedores reanalisam, reprojetam e reimplementam o sistema, de forma a solidificar as características aceitáveis e modificar as inaceitáveis.

Essas etapas persistem até a aceitação total do protótipo, que por fim é instalado no ambiente. A grande vantagem da prototipação consiste no fato de facilitar o entendimento dos requisitos pelo analista de sistemas. Muitas vezes, o usuário não consegue esclarecer com precisão suas necessidades, então os desenvolvedores oferecem um sistema provisório para verificar se ele atende às necessidades do usuário.



Prototipação.

Apesar dos esforços e avanços reconhecidos à metodologia de desenvolvimento rápido de aplicações, ainda havia resquícios de rigidez em tais metodologias. Esse fato justificou o aparecimento de novas metodologias de desenvolvimento, como as metodologias ágeis.

Juntamente com Fowler, diversos desenvolvedores e professores de computação escreveram o manifesto ágil, que consiste numa série de prerrogativas concernentes ao desenvolvimento e engenharia de software. Essas prerrogativas se baseavam na observação de que as metodologias de desenvolvimento eram muito burocráticas e fundamentadas excessivamente nos documentos, oferecendo, muitas vezes, um produto insatisfatório e demorado.

Todas as metodologias ágeis se fundamentam essencialmente na programação. Todos os membros da equipe devem ser de fato bons programadores. Na realidade, as metodologias ágeis exigem um conhecimento diversificado na análise, pois supõem que o mesmo programador que irá criar o código deverá compreender bem as necessidades do cliente. Além disso, as metodologias ágeis se concentram em entregar pequenas unidades do sistema, às vezes em tempo hábil inferior a uma semana, de forma que o sistema possa gerar valor para o cliente o mais rapidamente possível.

Por outro lado, essa maneira de agir só se tornou possível com o avanço tecnológico, já que entregar pequenas unidades do sistema sem que isso interferisse nas futuras entregas só foi possível mediante novas linguagens e técnicas de programação. Ademais, por estar fundamentada na produção de código, as metodologias ágeis recomendam fortemente a testagem exaustiva do sistema, que, obviamente, se trata de testes automatizados.

Como cada iteração corresponde a uma entrega, boa parte da documentação é dispensada, deixando a fase de planejamento em nível mínimo. Por isso, toda comunicação deve ocorrer preferencialmente de forma presencial. Isso possibilita o aumento do entendimento do sistema pelas partes envolvidas e agiliza a comunicação.

A comunidade de desenvolvimento suscitou, pelo menos, a criação de três metodologias ágeis distintas, mas que se fundamentam sob os mesmos princípios: **XP** (extreme programming), **DSDM** (dynamic systems development method) e **Scrum**. Aparentemente, o **XP** é o mais utilizado, seguido pelo **Scrum**.