



*Técnico em Desenvolvimento de Sistemas Online*

# ANÁLISE DE SISTEMAS E PROJETOS

# GEEaD - Grupo de Estudo de Educação a Distância

## Centro de Educação Tecnológica Paula Souza

### Expediente

GEEaD – CETEC  
GOVERNO DO ESTADO DE SÃO PAULO  
EIXO TECNOLÓGICO DE INFORMAÇÃO E COMUNICAÇÃO  
CURSO TÉCNICO EM DESENVOLVIMENTO DE SISTEMAS  
FUNDAMENTOS DE INFORMÁTICA

*Autores:*  
*Eliana Cristina Nogueira Barion*  
*Marcelo Fernando Iguchi*

*Revisão Técnica:*  
*Lilian Aparecida Bertini*

*Revisão Gramatical:*  
*Juçara Maria Montenegro Simonsen Santos*

*Editoração e Diagramação:*  
*Flávio Biazim*

**São Paulo – SP, 2019**

# APRESENTAÇÃO

Este material didático do Curso Técnico em Desenvolvimento de Sistemas modalidade EaD foi elaborado especialmente por professores do Centro Paula Souza para as Escolas Técnicas Estaduais – ETECs.

O material foi elaborado para servir de apoio aos estudos dos discentes para que estes atinjam as competências e as habilidades profissionais necessárias para a sua plena formação como Técnicos em Desenvolvimento de Sistemas.

Esperamos que este livro possa contribuir para uma melhor formação e aperfeiçoamento dos futuros Técnicos.

---

# AGENDA 13

---

## METODOLOGIA ÁGIL





## MERGULHANDO NO TEMA...

Na agenda anterior, observamos que as metodologias tradicionais funcionavam de maneira muito limitada. As tarefas seguiam uma ordem linear de desenvolvimento, fazendo com que se fizesse necessário atender a todos os objetivos antes de iniciar o desenvolvimento do projeto. Cada uma das fases somente poderia ser iniciada depois que a tarefa anterior fosse finalizada, dessa forma, o que já estava pronto era difícil de ser verificado enquanto as outras etapas estivessem em andamento. Essas metodologias traziam alguns problemas, como:

- Perda de tempo no processo de planejamento e atraso no desenvolvimento;
- Dificuldade em antecipar as necessidades futuras;
- Atraso na entrega do produto;
- Erros apenas percebidos quando o produto final fosse entregue ao cliente.

Esses e outros problemas fizeram com que essas metodologias tradicionais fossem repensadas na década de 1980, com o manifesto ágil, propondo uma nova metodologia que prevê:

**Indivíduo e interações:** mais que processos e ferramentas, indicando que devemos utilizar mais interação entre os indivíduos do que processos engessados como vimos no modelo cascata.

**Software em funcionamento:** mais que documentação abrangente: clientes querem resultados vendo o software em funcionamento.

**Colaboração com o cliente:** responder a mudanças mais que seguir um plano.

Deve haver um grande entrosamento entre o analista e o usuário, viabilizando, assim, o desenvolvimento do sistema. A partir do momento em que o analista conhece o problema do usuário, basta transformar as suas necessidades em especificações técnicas para continuar o seu trabalho.

A colaboração com o cliente consiste em fazer um levantamento de dados e fatos para descobrir o que realmente precisa ser feito. É o conhecimento do problema pelo analista que consiste em fazer um levantamento de dados e fatos para descobrir o que realmente precisa ser feito.

A metodologia Scrum tem como objetivo estruturar o trabalho em equipe no planejamento e desenvolvimento de softwares. Os projetos são divididos em ciclos (tipicamente mensais) chamados de **Sprints**, com times multifuncionais e auto gerenciáveis. A estrutura do Scrum procura aproveitar o que a equipe tem de melhor em relação à otimização de tempo, rapidez e qualidade do trabalho.

A essência dessa metodologia baseia-se na ideia de verificar de tempos em tempos se o projeto está no caminho correto e se o que está sendo desenvolvido está de acordo com a expectativa do cliente. Essa ação é chamada de “inspeção e adaptação”.

Todos os dias são executados um conjunto de atividades do projeto, por exemplo:

Em sua casa você tem uma lista de afazeres, como: ir ao supermercado, levar as crianças à escola, organizar suas mochilas para certificar-se de que o material da aula está em ordem etc. Todas essas atividades devem ser priorizadas e cumpridas diariamente, assim são os projetos com suas atividades pontuais.

Vamos conhecer a composição da equipe:



O **Product Owner** é o representante dos stakeholders no projeto, ou seja, a parte interessada, este é responsável pelo retorno financeiro e define com os desenvolvedores qual ação deve ser executada primeiro e a cada sprint (conjunto de atividades) é feita uma nova redefinição.



**Team** é a equipe que irá desenvolver o projeto, com auxílio do Product Owner. A equipe deve conter no mínimo 6 pessoas e no máximo 10, organizadas, auto gerenciáveis e multifuncionais. É a equipe que irá definir o que irão realizar em cada sprint, definindo a melhor estratégia para atingir o sucesso do projeto.



**Scrum Master** é responsável pelo processo Scrum. Auxilia o Team e assegura que a equipe não se comprometa excessivamente com relação àquilo que é capaz de realizar durante um Sprint.

Agora que sabemos como é composta a equipe, vamos entender como funciona as **Sprints**.

Sprints são interações de um ciclo de desenvolvimento em Scrum. Cada uma recebe um incremento do produto, conforme as entregas da equipe. O tempo para cada sprint pode ocorrer em uma semana e terminar no máximo em um mês e nesse período terá que implementar as funções no Sprint Backlog a qual chamamos de histórias. Na figura 12, podemos observar como funciona este processo.

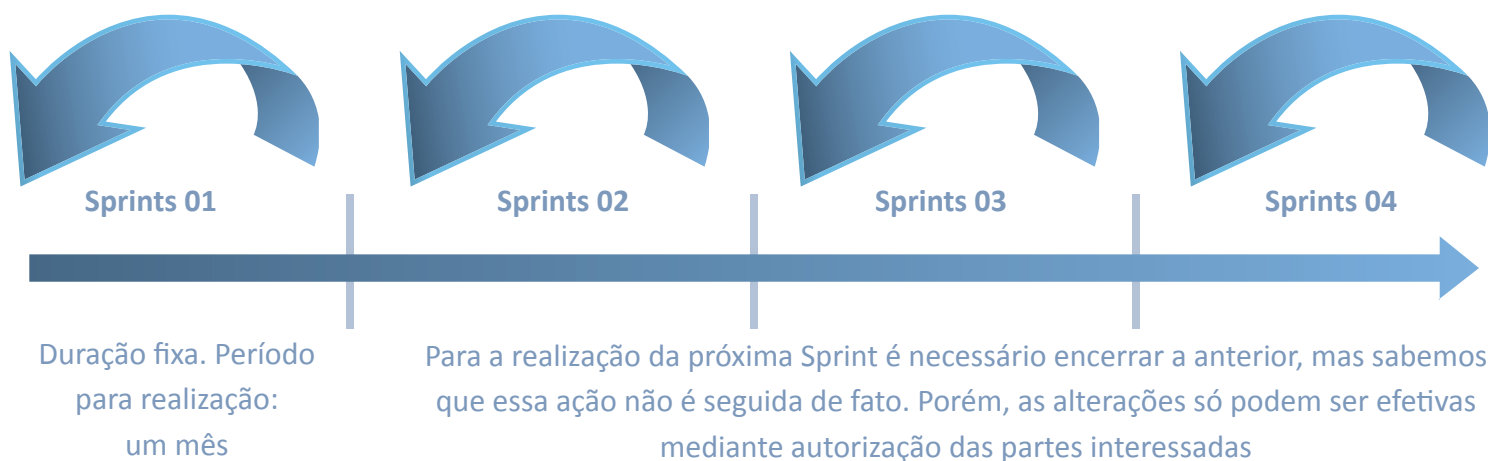


Figura 12 - Arquivo GEEaD

**Sprint Backlog** é a lista de atividades relacionadas pelo time Scrum, e devem ser concluídas no período de cada sprint. Na reunião de planejamento, a equipe relaciona as tarefas e o tempo necessário para execução e o prazo de término das atividades. Para o acompanhamento das tarefas é feita uma tabela com as sprints backlog e o responsável pelas ações. Uma vez por dia, é necessária a atualização da planilha, para que o Scrum Master gere um gráfico com as atividades realizadas e pendentes.

### Sprint Planning - Reunião de Planejamento



*Product Backlog – pode ser discutidas em muitas semanas ou até mesmo em meses de desenvolvimento. Porém, pode ser concluída dentro de uma única sprint. Deve ser analisada pela equipe (time) realizando o sprint planning, ou seja, planejamento de sprint*

Figura - 13

Sabemos que todo desenvolvimento do software é complexo e requer um constante monitoramento e controle com um nível alto de comunicação. Com base nesta complexidade, o Scrum utiliza reuniões diárias de 15 minutos para coletar informações sobre cada sprint e se todos do time compreendem o status de cada etapa.

Toda a reunião Sprint de planejamento é assistida pelo Product Owner, pelo Scrum Master e por toda equipe. No decorrer da reunião, o Product Owner lista as prioridades para cada equipe, que faz questões pertinentes a cada atividade, estimando um prazo coerente com as atividades a serem desenvolvidas.

Durante a reunião cada integrante da equipe deve responder a três perguntas:

*O que você fez ontem?*

*O que você fará hoje?*

*Há algo que possa te impedir de fazer?*



Figura - 14

Com base nesta reunião, o Scrum Master e o time, têm a visão do que está acontecendo de fato com o projeto. Todos os problemas apontados são destacados pelo Scrum Master para que o mesmo consiga minimizar os impactos futuros no projeto.

Para que você possa entender melhor esses termos mais técnicos, vamos a um exemplo mais real:

Imagine que você tenha uma empresa de desenvolvimento de software – a empresa GestSoftware - e que você, um cliente da empresa X, contrate a sua empresa para fazer o desenvolvimento de um novo site. Um gerente de projetos dessa empresa X, o **Product Owner**, faz o levantamento de todas as necessidades do cliente em relação ao site. Então, esse Product Owner deve se reunir com a **equipe de desenvolvedores** da GestSoftware: os profissionais de design, front-end e back-end para definirem todas as tarefas necessárias para a produção do novo site que está sendo solicitado. Essa reunião se refere à **Reunião de Planejamento**, para que juntos possam formalizar uma lista de tarefas (**Product Backlog**).

Você, o **Scrum Master**, fica encarregado de organizar as reuniões e garantir que todos entendam as tarefas que foram definidas nessa reunião para que a GestSoftware possa atender a todas as necessidades do cliente.

A equipe decide democraticamente que vai entregar a página inicial pronta - Produto até o final da semana (Sprint).



Figura - 15



A equipe então divide o produto em tarefas menores (Sprint Backlog), como por exemplo, criar o layout, desenvolver o HTML e CSS, implantar um sistema de administração de conteúdo e testar tudo em diversos browsers.

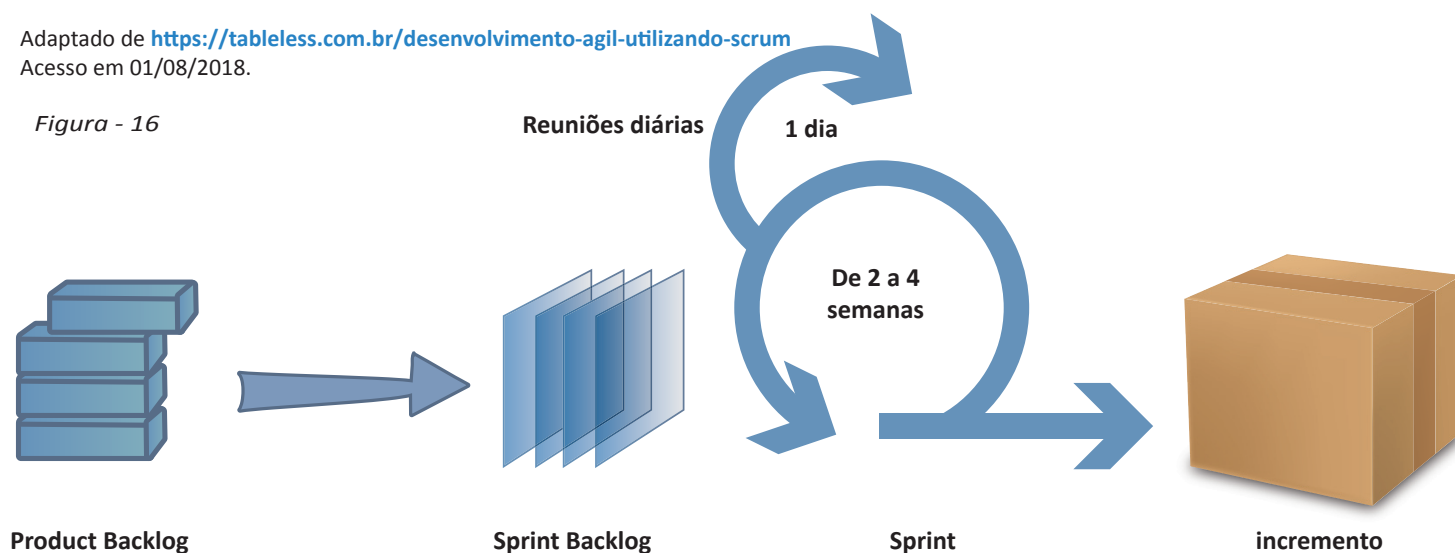
Cada membro da equipe de desenvolvedores fica responsável por determinadas ações e juntos definem um prazo de duração em horas para cada tarefa. Diariamente vocês se reúnem por 15 minutos para acompanhar o andamento do projeto (Sprint Planning).

Ao final da semana, vocês apresentam o resultado (Revisão da Sprint) e a equipe de desenvolvedores se reúne para refletirem sobre as dificuldades e melhorias que podem ser feitas na próxima semana (Retrospectiva da Sprint).

O mais importante é que dessa forma vocês consigam entregar um produto parcial, porém funcional (Incremento) para o cliente e a partir do feedback de todos, vocês decidem qual será a tarefa a ser realizada na próxima semana, por ordem de prioridade. E assim o ciclo recomeça!

Adaptado de <https://tableless.com.br/desenvolvimento-agil-utilizando-scrum>  
Acesso em 01/08/2018.

Figura - 16



*“O planejamento não é uma tentativa de prever o que vai acontecer. O planejamento é um instrumento para raciocinar agora, sobre que trabalhos e ações serão necessários hoje, para merecermos um futuro. O produto final do planejamento não é a informação: é sempre o trabalho.” Peter Drucker.*