

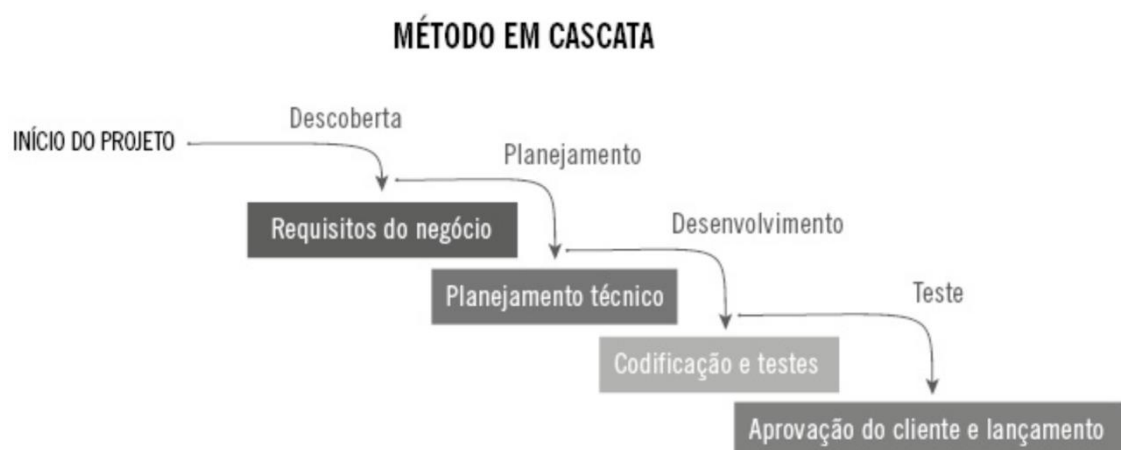
## Prefácio

- Desenvolvimento de software
  - Metodo cascata
  - Diagrama de Gantt
  - Citando a primeira vez o SCRUM: Surgiu em contraste ao método de desenvolvimento de softwares hardware cascata.
    - Pequenas equipes
    - Priorização de tarefas
    - Sprints
    - Aprimoramento contínuo
    - Feedback dos consumidores (professores)
    - Reuniões breves diárias
    - Como adaptar isso ao contexto acadêmico?

## Capitulo 1

- Caso da ineficiência do FBI( pag 10)

O que tinha dado errado e como a situação foi resolvida são o motivo por que estou escrevendo este livro. Não era uma questão de inteligência. Não era que a agência não tivesse as pessoas certas nos lugares certos e também não era uma questão de tecnologia errada. Também não tinha nada a ver com ética no trabalho ou com o estímulo adequado de competitividade. Era por causa da *maneira* como as pessoas estavam trabalhando. A *maneira* como a maioria das pessoas trabalha. A maneira como nós achamos que o trabalho *precisa* ser feito, porque foi assim que aprendemos a fazê-lo.



- 
- “Esses diagramas são realmente algo impressionante de se ver. O único problema é que eles estão sempre errados. Sempre”
- SCRUM: Metafora do rugby
  - Alinhamento cuidadoso
  - Unidade de propósito

- Clareza de objetivo
- “Todo projeto envolve descoberta de problemas e surtos de inspiração” P 15

O Scrum acolhe a incerteza e a criatividade. Coloca uma estrutura em volta do processo de aprendizagem, permitindo que as equipes avaliem o que já criaram e, o mais importante, *de que forma* o criaram. A estrutura do Scrum busca aproveitar a maneira como as equipes realmente trabalham, dando a elas as ferramentas para se auto-organizar e, o mais importante, aprimorar rapidamente a velocidade e a qualidade de seu trabalho.

P15

- **Ciclo de Inspeção e Adaptação:** De tempos em tempos, pare de fazer aquilo que está fazendo, revise o que já fez e verifique se ainda deveria estar fazendo aquilo e como poderia estar fazendo melhor
- “Os resultados são inquestionáveis. A empresa tem duas opções: mudra ou morrer”
- **Priorização daquilo que agrega mais valor.**
- **Manifesto Ágil:**
  - Pessoas em vez de processos
  - Produtos que realmente funcionam em vez de documentação de como ele deveria funcionar
  - Trabalhar com o cliente em vez de negociar com eles
  - Responder às mudanças em vez de seguir um plano
- **Remoção do obstáculo**
  - Tarefa da gerência-> identificar e remover um obstáculo para **fluxo calmo e rápido**

- “Scrum funciona com a definição de objetivos sequenciais que devemos concluídos num período definido”

No Scrum chamamos esses ciclos de Sprint [corrida de velocidade de curta distância]. No início de cada ciclo, acontece uma reunião para planejar o Sprint. A equipe decide a quantidade de trabalho que acredita ser capaz de realizar nas duas semanas seguintes. Eles escolhem as tarefas na lista de prioridades, as anotam em *post-its* e os colam na parede. A equipe decide quantas tarefas será capaz de executar em duas semanas.

- Depois de mostrarem o que conseguiram fazer — e é aqui que as ideias de Ohno entram —, a equipe discute não o que fizeram, mas *como* fizeram. Eles perguntam: “Como podemos trabalhar melhor no próximo Sprint? Quais foram os obstáculos que tivemos de remover durante esse período? Quais são os obstáculos que estão diminuindo o nosso ritmo?”.

- “Não trabalhando por mais tempo mais trabalhando melhor.” -> ficaram mais rápidos -> descobriram o que as atrasavam -> eliminaram os obstáculos

- E, na opinião dele, qual era a parte mais eficaz do Scrum? “*As demonstrações*. O trabalho voltado para um produto demonstrável com frequência”. A cada duas semanas a equipe do Sentinel se reunia e demonstrava tudo o

- E foi isso que fez as coisas funcionarem, afirma Johnson. “Scrum não é sobre os desenvolvedores. Mas sim sobre os clientes e os *stakeholders*. Sério, era uma mudança organizacional. Mostrar o *produto real* era a parte mais eficaz.”

Existe uma piada antiga na comunidade do Scrum: uma galinha e um porco estão caminhando pela estrada, e a galinha diz:

“Ei, porco, eu estava pensando que a gente devia abrir um restaurante.”

“E qual vai ser o nome do restaurante?”, pergunta o porco.

“Que tal ‘Presunto e Ovos’?”

- “Não, obrigado”, responde o porco. “Eu teria que me comprometer, mas você só teria de se envolver.”
- O NOVO FORD TEM MAIS LINHAS DE CÓDIGO QUE O FACEBOOK E TWITTER JUNTOS!!!!!!

- “O que o SCRUM faz é promover a união das equipes para criar grandes projetos e isso exige que todos não apenas vejam o objetivo final, mas que também façam entregas incrementais para atingí-los. P24

### *PONTOS PRINCIPAIS*

**Planejar é útil. Seguir cegamente os planos é burrice.** É simplesmente *tentador demais ficar desenhando diagramas sem fim*. Todo o trabalho que precisa ser feito em um projeto de grande porte deve ser definido para todos verem — mas, quando os planos detalhados se deparam com a realidade, eles viram ruínas. Inclua no seu método de trabalho a possibilidade de mudança, descoberta e inovação.

**Inspecção e Adaptação.** De tempos em tempos, pare de fazer o que está fazendo, revise o que já fez e verifique se isso ainda é o que você deveria estar fazendo e se existe uma maneira de fazer melhor.

**Mudar ou morrer.** Ficar preso ao modo antigo de fazer as coisas, de mandar ou controlar e manter uma previsibilidade rígida resultará apenas no fracasso. Nesse meio tempo, a concorrência que estiver disposta a mudar deixará você comendo poeira.

**Fracasse rápido para que possa corrigir o problema o quanto antes.** A cultura corporativa costuma dar mais valor a formulários, procedimentos e reuniões do que à criação de valor palpável que pode ser verificado a curtos prazos pelos usuários. O trabalho que não resulta em valor real é loucura. Trabalhar em um produto em ciclos curtos possibilita um feedback inicial do usuário, permitindo que você possa eliminar imediatamente tudo aquilo que obviamente constitui um desperdício de esforço.

- 

## Capítulo 2 As origens do SCRUM

- Observar, avaliar, decidir e agir
- Não saia em busca de cascatas
- “A direção não impunha ordens, ao contrário, os executivos eram líderes facilitadores focados em retirar obstáculos do caminho”
- Ciclo PDCA

Quando treino as pessoas para usar o Scrum, é este o exemplo que uso: aviões de papel. Divido-as em equipes e digo que o objetivo é construir o maior número de aviões de papel que conseguirem e que sejam capazes de voar até o outro lado da sala. Haverá três papéis a serem desempenhados em cada equipe: uma pessoa verificará quantos dos aviões feitos conseguem voar de verdade; outra trabalhará na produção, mas também estará atenta ao processo em si e buscará maneiras de fazer com que a equipe construa aviões melhores e acelere o trabalho. Os demais participantes deverão se concentrar em fazer o maior número de aviões capazes de atravessar a sala voando no tempo definido.

Então, eu conto que faremos três ciclos de seis minutos para o processo de construção. As equipes têm um minuto de cada ciclo para planejar (P) como vão construir o avião; três minutos para fazer (D) — construir e testar quantos aviões realmente voam; e, por fim, terão dois minutos para verificar (C). Nessa fase a equipe busca meios para melhorar o processo de construção. O que deu certo? O que deu errado? O design deve ser alterado? Como o processo de construção pode ser melhorado? Então, eles vão agir (A). No universo de Deming, “agir” significa mudar o modo de trabalhar, com base em resultados e informações ambientais reais. É a mesma estratégia usada pelo robô de Brooks.

Realize esse ciclo três vezes, esteja você construindo aviões de papel ou espaçonaves de verdade, você vai melhorar de forma significativa (em uma razão de duas a três vezes mais rápido e com pelo menos o dobro da qualidade). Esse ciclo PDCA, uma ideia radical quando Deming a apresentou para os japoneses, foi o que fez a Toyota se tornar a empresa automobilística número 1 do mundo. E é dessa forma que qualquer tipo de produção “enxuta” (o termo americano para o uso dos conceitos do Sistema Toyota de Produção), ou desenvolvimento Scrum de produção, é feito.

## PONTOS PRINCIPAIS

**Hesitação é a morte.** Observe, avalie, decida, aja. Saiba onde está, avalie suas opções, tome uma decisão e aja!

**Procure respostas.** Sistemas adaptativos complexos seguem algumas regras simples, que aprendem a partir do ambiente em que se encontram.

**Grandes equipes.** São multifuncionais, autônomas, capazes de tomar decisões e motivadas por um objetivo transcendente.

**Não adivinhe.** Planeje (*Plan — P*), Faça (*Do — D*), Verifique (*Check — C*), Aja (*Act — A*): planeje o que vai fazer. Faça. Verifique se o resultado é o que desejava. Aja de acordo com as informações e faça as coisas de outra maneira. Repita em ciclos regulares e, ao fazer isso, obtenha um aprimoramento contínuo.

**Shu Ha Ri.** Primeiro, aprenda as regras e as formas e, uma vez que as dominar, faça inovações. Por fim, em um estado elevado de domínio, descarte as formas e apenas seja — com todo o aprendizado internalizado as decisões são tomadas de forma quase inconsciente.

## Capítulo 3 – Equipes

- Foco no desempenho da equipe, invés do desempenho individual (masterização de BLOOM)

**1. Transcendência:** elas têm um senso de propósito além do comum. O objetivo percebido por todos permite que eles transformem o ordinário em extraordinário. De uma forma bastante verdadeira, a decisão de não estar na média, mas ser grandioso, muda o modo como eles se enxergam e o que são capazes de fazer.

**2. Autonomia:** as equipes se auto-organizam e se autogerenciam; têm o poder de tomar as próprias decisões sobre como fazer o próprio trabalho e têm o poder de fazer com que tais decisões sejam acatadas.

**3. Interfuncionalidade:** as equipes possuem todas as habilidades necessárias para concluir o projeto: planejamento, projeção, produção, vendas e distribuição. E tais habilidades alimentam e reforçam umas as outras. Nas palavras de um membro da equipe que projetou uma nova câmera revolucionária da Canon: “Quando todos os membros da equipe se encontram em uma sala grande, a informação de alguém se torna sua, mesmo sem você perceber. Então, você começa a pensar no que é melhor fazer primeiro e o que é melhor fazer depois pelo grupo como um todo, e não apenas de acordo com o que é melhor para você”.<sup>12</sup>

- Um dos conceitos chaves do SCRUM é que as equipes decidem por si mesmo como irão realizar o trabalho
- A responsabilidade da gestão da empresa é estabelecer os objetivos estratégicos
- 3 perguntas tradicionais do SCRUM
  1. O que você fez desde a última vez que nos encontramos
  2. O que você vai fazer até nosso próximo encontro
  3. O que está atrapalhando o seu trabalho?
  4. TRABALHO DO LIDER -> certificar-se que tudo que estava atrapalhando a equipe esteja solucionado até o próximo encontro



## SCRUM UDEMY

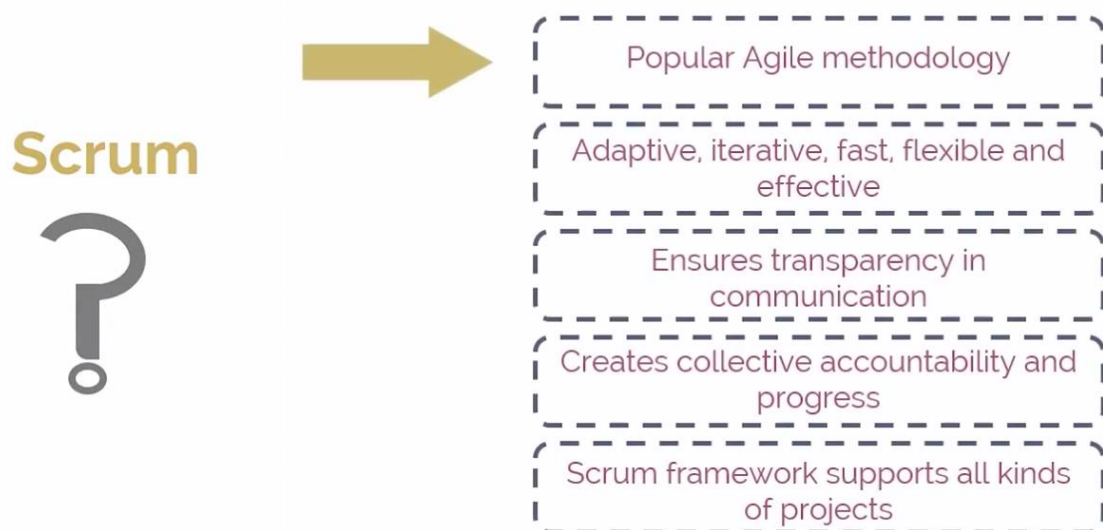
### CAP 1 SCRUM OVERVIEW

1) Scrum Project involves a **collaborative** effort to create a new product, service, or other results as defined in the **Project Vision Statement**

2) Projects impacted by

- Time
- Cost
- Scope
- Quality
- Resources
- Organization Capabilities
- Other Limitations
  - That make it difficult to plan, execute manage and succeed

## SCRUM



Supports product and service development in all types of industries and in any type of project, regardless of its complexity



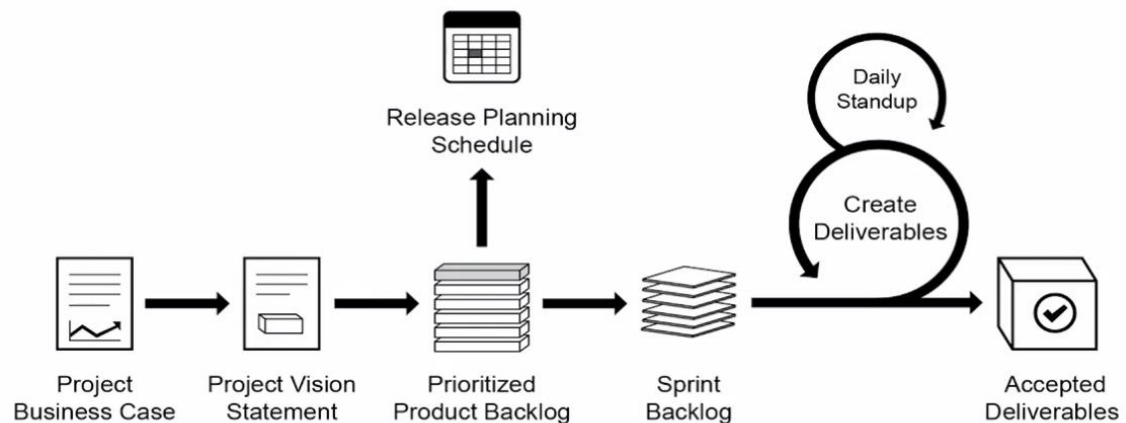
## Scrum Teams



Cross-Functional	Self-Organized
Empowered	Use Sprints

Sprints are short time spans in which a certain amount of work must be done

## Scrum Flow



### PRIORITIZED PRODUCT BACKLOG

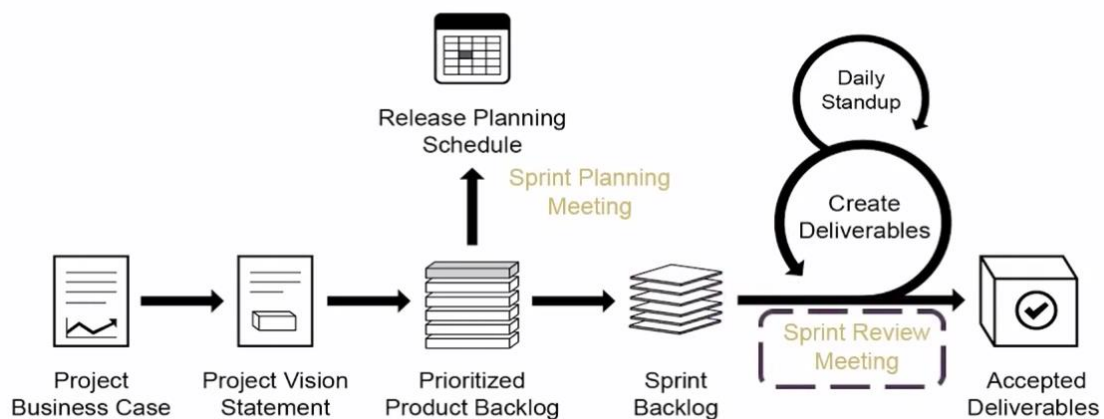
Contains a prioritized list of business and project requirements written in the form of User Stories



User Stories are considered for inclusion in the Sprint.

A Sprint generally lasts between one and six weeks.

Short, highly focused Daily Standup Meetings are conducted.

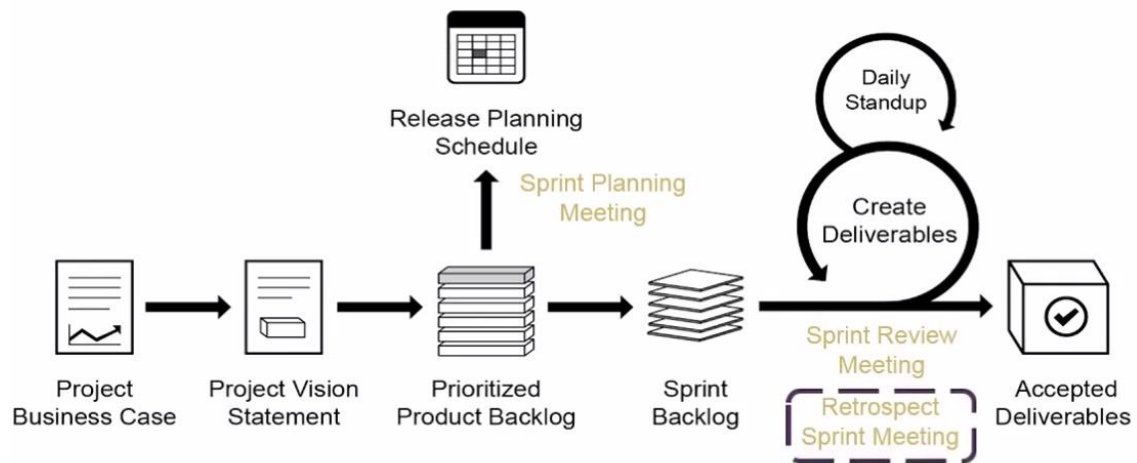


## DEPOIS DO SPRINT

Product Owner and relevant stakeholders are provided a demonstration of the Deliverables.

The Product Owner accepts the Deliverables only if they meet the Predefined Acceptance Criteria.





## Improve processes and performance

### CAP 2 -> history of SCRUM

In the mid 80's, Hirotaka Takeuchi and Ikujiro Nonaka defined a flexible and all-inclusive product development strategy where the development team works as a unit to reach a common goal. Similar to rugby approach.

Ken Schwaber and Jeff Sutherland elaborated on the Scrum concept and its applicability to Software Development in 1995 in Austin, Texas.

Object-Oriented Programming, Systems, Languages & Applications (OOPSLA) conference held in 1995 in Austin, Texas

### CAP 3 -> Why use SCRUM

#### Benefits of using SCRUM

##### 1. Adaptability

Empirical process control and iterative delivery make projects

- a. adaptable and open to incorporating change when it occurs

2. Customer Centric

Ensures a customer-oriented framework in a Scrum-controlled project.

a.

3. Continuous Delivery of Value

Delivery of value through the Ship Deliverables process as frequently as the customer requires.

a.

4. Early Delivery of High Value

The Create Prioritized Product Backlog process ensures that the highest value requirements of the customer are satisfied first.

a.

5. Continuous Feedback

Deliverables are improved progressively Sprint by Sprint, through the Groom Prioritized Product Backlog process.

a.

Continuous Feedback is provided through the Conduct daily standup and demonstrate and Validate Sprint processes.

b.

6. Transparency

Information emitters like a Scrumboard and Sprint Burndown

a. Chart are shared, which leads to an open working environment.

Sprint Review Meetings demonstrate potentially shippable products to stakeholders, keeping them full informed about the current status and progress of the project.

7.

Sprint Review Meetings demonstrate potentially shippable products to stakeholders, keeping them full informed about the current status and progress of the project.

8.

9. Continuous Improvement

Deliverables are improved progressively Sprint by Sprint, through the Groom Prioritized Product Backlog process.

a.

10. Sustainable Pace

People involved can work at a sustainable pace that they can, in theory, continue indefinitely.

a.

11. Efficient Development Process

It is observed that time-boxing and minimizing non-essential work leads to higher efficiency levels.

a.

12. Motivation

- Conducting Daily Standup and Retrospect Sprint processes lead to greater levels of motivation among employees.
- a.
13. Faster Problem Resolution
- Collaboration and colocation of cross-functional teams leads to faster problem solving
- a.
14. Effective Deliverables
- Prioritized Product Backlog process and regular reviews after creating deliverables ensures effective deliverables to the customer.
- a.
15. Collective Ownership
- The Approve, Estimate, and Commit User Stories process allows team members to take ownership of the project.
- a.
16. High Velocity
- Enables highly skilled cross-functional teams to achieve their full potential and high velocity.
- a.
17. Innovative Environment
- Learning, and adaptability lead to an innovative and creative work environment.
- a.
- 18.

#### **CAP 4 – Scalability of Scrum**

How fast is scalable and fast is SCRUM

ERRO -> SCRUM serve apenas para pequenos projetos

SCRUM

- Equipes no SCRUM devem ter de 6 a 10 membros
- Quando num membros > 10 -> formar múltiplos times SCRUM

Quem irá coordenar o trabalho entre os diversos times SCRUM?

Convenio de SCRUM de SCRUMs



SCRUM of SCRUM é o processo que assegura a sincronização das múltiplas equipes SCRUM

Scrum Teams are represented



Provide updates about progress

Discuss challenges faced during the project

Coordinate activities

No set rules regarding the Frequency of Meetings

Factors defining the frequency:

Inter-team dependency

Size of the project

Scrum Guidance Body recommendations

Level of complexity



## Cap 5 Purpose of SBOK

Por que SBOK guide ? ->

Por se precisa do SBOK Guide

Em que ele se baseia

Muitas organizações tem escolhido SCRUM invés de modelos de gerenciamento tradicional

Por que princípios e práticas SCRUM tem tido maior impacto

SBOK -> guia necessário para organizações para implementar SCRUM

- ➔ Baseado na experiência de centenas de organizações e especialistas com o modelo SCRUM

### Who uses SBOK and how it is valued?

SBOK is especially valuable to the Scrum Core Team members

#### SCRUM CORE TEAM MEMBERS

##### Product Owner



Use the SBOK Guide to fully understand the Scrum framework

Use the SBOK Guide particularly to understand the customer/stakeholder-related concerns

##### Scrum Master



Use SBOK Guide to learn their specific role in overseeing the application of Scrum Framework to Scrum projects

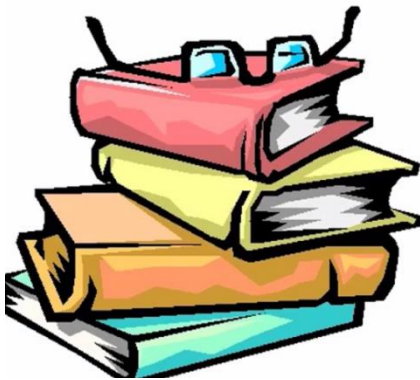
## Scrum Team Members



Use the SBOK Guide to better understand Scrum processes

Use the SBOK Guide to understand the associated tools that may be used to create the project's product or service

## Reference source



- Portfolio Product Owner
- Portfolio Scrum Master
- Program Product Owner
- Program Scrum Master
- Scrum Guidance Body
- Stakeholders: sponsor, customer, and users

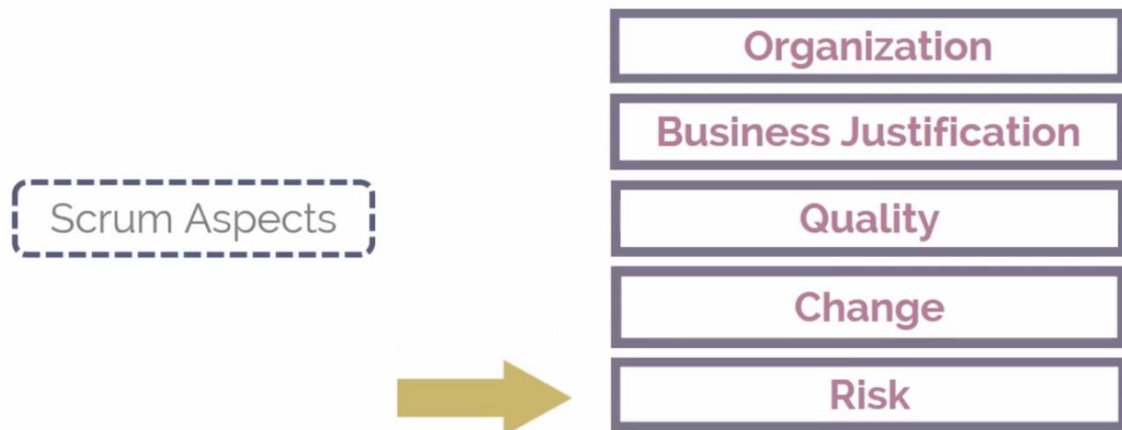
## CERTIFICAÇÃO





## Cap 6 -> Framework of SBOK

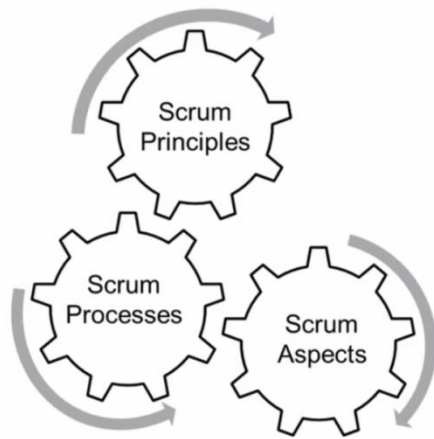
Broadly divided into three key areas



Five phases SBOK



19 processos nessas fases

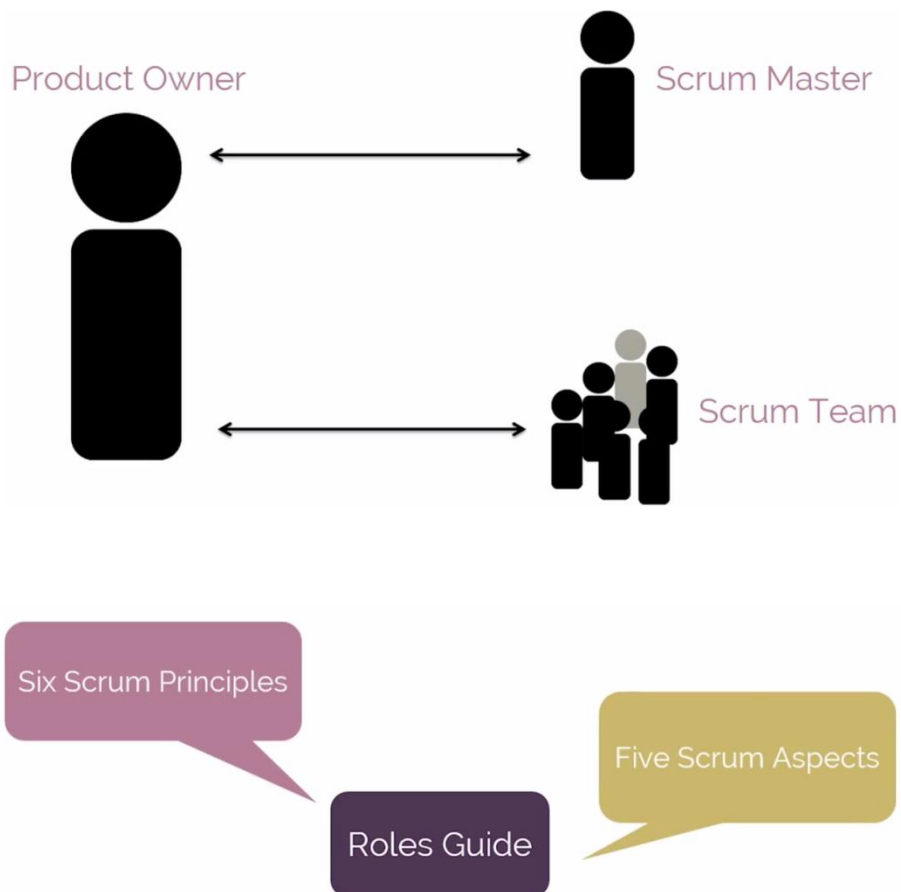


SBOK™ Framework

All areas need to work together for the effective application of Scrum



### Cap 7 – How to Use SBOK Guide



Persons being introduced to Scrum

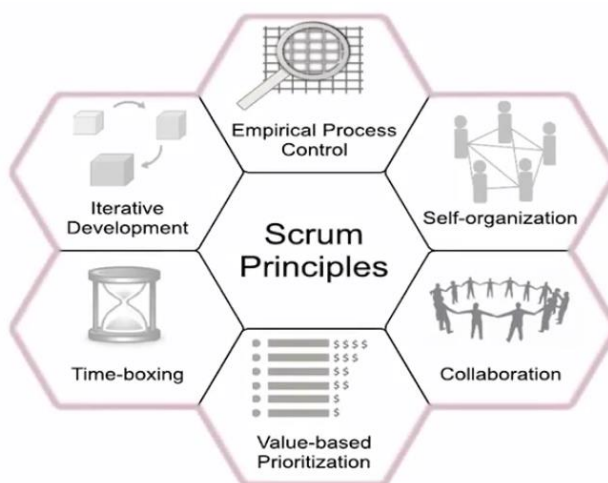
Focus Primarily on mandatory Inputs, Tools and Outputs

Experienced Practitioners

Should read the entire process phases

## Cap 8 – Scrum Principles

Scrum principles are the core guidelines for applying the Scrum framework and should mandatorily be used in all Scrum Projects



## Scrum Principles

Can be applied to any type of project in any organization

To ensure effective implementation of the Scrum framework

Non-negotiable

Must be applied as specified in the *SBOK Guide*

### → Empirical Process Control

- Transparency
- Inspection
- Adaptation

### → Self Organization

- Today Worker
- Self Organize rather than command and control

### → Collaboration

- Awareness
- Articulation
- Appropriation
- Project Management as a shared value-creation process

### → Value Based Priorization

- Deliver maximum business value

### → Time boxing

- Descreve como tempo é uma limitação
- Ajudar a efetivamente gerenciar e executar um projeto
- **SPRINTS**
- **Daily Standup Meetings**
- **Sprints Planning Meeting**
- **Sprint Review Meeting**

### → Iterative Development

- Como gerenciar mudanças
- Construir projetos para satisfazer as necessidades do cliente

## Cap 9 – Scrum Aspects

## Scrum Aspects



Organization
Business justification
Quality
Change
Risk

## 1. Organization ->

### a. Roles and responsibilities

#### b. Roles

##### i. Core Roles

1. Envolvidos na produção do serviço ou produto
2. Indivíduos com Core Roles são **Completamente** envolvidos com o projeto
3. Responsáveis últimos do sucesso de cada iteração e do projeto como um todo
4. Papel designado para:

##### a. Product Owner

- i. Pessoa responsável para adquirir o **valor de negócio** máximo do projeto
- ii. Articular os **requisitos do cliente**
- iii. Manter a **justificação de negócio** para o projeto
- iv. Representa a “voz do cliente”

##### b. Scrum Master

- i. Assegura que o time SCRUM tem um **ambiente apropriado**
- ii. **Guia, ajuda e ensina** os participantes nas praticas SCRUM
- iii. Soluciona impedimentos das equipes
- iv. Se certifica que todos processos SCRUM estão sendo seguidos

##### c. EXEMPLO SCRUM Master



Software Development team  
facing roadblocks

What is the stumbling block here? Lack of expertise required for the project

SCRUM MASTER : Find ways to clear the impediment of lack of expertise

Raise an immediate requirement for hiring software developers

i. Decision based on the cost effectiveness and time consumption

##### d. Scrum Team

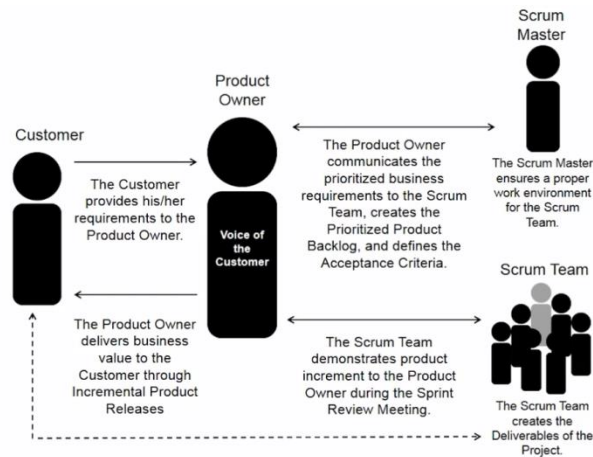
- i. Time responsável por entender os requisitos elencados pelo **PRODUCT OWNER**
- ii. Criar os *entregáveis do projeto*
- iii.

##### ii. Non Core Roles

1. Papéis **opcionais** -> pode incluir todos membros do projeto até mesmo aqueles que não possuem papéis formais dentro da equipe
2. Não são responsáveis pelo sucesso do projeto
  - a. Stake holders

- b. Corpo de guia SCRUM
- c. Vendedores
- d. Chief Product Owner
- e. Chief Scrum Master
  - i. Responsável por coordenar atividades relacionadas a SCRUM em projetos grandes que envolvem múltiplos times SCRUM
  - ii.

3.



4.

## 2. Business Justification

- a. A organização deve avaliar o projeto antes de inicia-lo
  - i. Ajuda os principais tomadores de decisão a entenderem as mudanças necessárias, novos produtos ou serviços
- b. Baseado no conceito de **Value Driven Delivery**
  - i. Qualquer projeto possui
    - 1. Incerteza de resultados positivos
    - 2. Incerteza de completude e sucesso
  - ii. Dadas as incertezas
    - 1. SCRUM busca começar a entregar resultados o mais cedo possível
      - a. Isso traz a possibilidade de **reinvestimento**
    - 2. A adaptabilidade do SCRUM permite ao projeto ser mudado assim como seus processos no caso da justificação de negócio mudar

## 3. Quality

- a. Capacidade de completar e enviar os **entregáveis do projeto** alcançado os **critérios de aceitação** esperados pelo cliente
- b. Para isso, no SCRUM é adotada a **melhora contínua**
  - i. O time SCRUM aprende por experiência e engajamento dos stakeholders para manter o **Prioritized Product Backlog**
  - ii. **Prioritized Product Backlog** não é completo até o fim do projeto
  - iii. Mudanças nos requerimentos de projeto levam a mudanças internas e externas no ambiente de negócio
  - iv. Requerimentos são completados **iterativamente** por meio de **SPRINTS**



- v. **Erros** são consertados imediatamente, não postergados
- vi. **Quality related task** ( documentação, testes desenvolvimento) são completadas num mesmo SPRINT pela mesma equipe
- c. Exemplo:

Example - Internal customers are not happy with a product delivered in terms of 'quality', the issues can be listed in the Prioritized Product Backlog

- d. Melhora contínua com testes repetitivos aumenta a probabilidade de sucesso do projeto
- e. Discussão constante entre Time Scrum e Stakeholders diminui a diferença entre o que os Stakeholders esperam e o que é entregue

#### 4. Change

- a. Todo projeto está exposto a mudanças. É importante que os membros de um time SCRUM entendam que seus processos são planejados para “abraçar” a mudança.
  - i. Deve-se colher ao máximo os possíveis benefícios das mudanças e minimizar os prejuízos.
- b. **SCRUM Acknowledges**
  - i. Stakeholders podem mudar o que querem durante o projeto -> **Requirements Churn( bateria de requerimentos)**
  - ii. Dificilmente, os Stakeholders terão todos os requisitos definidos no começo do projeto
  - iii. SCRUM se adapta à mudança usando pequenos e iterativos SPRINTS
    - 1. Isso dá permissão aos Stakeholders de interagir com o time SCRUM e mudar os requerimentos se necessário

#### 5. Risk

- a. Eventos incertos que podem afetar os objetivos de um projeto contribuindo ou atrapalhando em seu sucesso
- b. **Positive impact**
  - i. Oportunities
- c. **Negative efect**
  - i. Threats
- d. Exemplo

Example

One of the key investors in a project backs off at a prime moment  
This is a risk affecting the project in a negative way

- i. In case, the project finds a better investor willing to invest in a bigger and a better way, it can be considered as an opportunity
- e. Gerenciamento de riscos

- i. Deve ser um processo proativo
- ii. Processo iterativo que começa com o projeto e continua pelo ciclos dele
- iii. Avaliar de acordo com:
  - 1. **Probabilidade de ocorrência**
  - 2. **Impacto da ocorrência**

## Cap 10 SCRUM process

---

**-User stories** are short, simple descriptions of a feature told from the perspective of the person who desires the new capability, usually a user or customer of the system. They typically follow a simple template:

*As a <type of user>, I want <some goal> so that <some reason>.*

User stories are often written on index cards or sticky notes, stored in a shoe box, and arranged on walls or tables to facilitate planning and discussion. As such, they strongly shift the focus from writing about features to discussing them. In fact, these discussions are more important than whatever text is written.

---

- A Project is a product backlog (lista de funcionalidade, características desejadas) of prioritized requirements
- Product owner is responsible for **prioritizing requirements**

### How do you get that work done?

- SCRUM process: **Definição de atividades e fluxo específico de um proejto Scrum**
  - **5 fases**
    - **Fase 1 : Initiate**
      - **Processo 1 - Create project Vision:** Project Business is reviewed to create Project Vision Statement
      - **Processo 2 – Identify Scrum Master and Stakeholders** : Scrum master identificado
      - **Processo 3 – Form Scrum Team:** Formar time Scrum -> geralmente os membros são selecionados pelo

*Product Owner*, possivelmente com a colaboração do *Scrum Master*

- **Processo 4 – Develop Epic(s):** *Develop Epics -> project vision statements serve as bases to epics(large unrefined user histories) -> User group meeting may be held to develop it*
- **Processo 5 – Create a Prioritized Product Backlog ->** Epic(s) and User's histories são refinados e priorizados. **Done criteria specified**
- **Processo 6 – Conduct Release Planning:** O Core Team revisar o **User Story** no **Prioritized Product Backlog** para desenvolver um **Release Planning Schedule**
- **Fase 2 : Project Plan and Estimate**
  - **Processo 1 – Create User Stories:** História dos usuários e critérios de aceitação são criados -> Usualmente são escritos pelo product owner e pelos Scrum Team Members -> usado para certificar-se que os requerimentos dos clientes são completamente entendidos e descritos. Users histories são incorporados no product backlog
  - **Processo 2 – Approve, Estimate and Commit User Stories:**
    - Product Owen approves User Stories for a Sprint
    - Scrum Master and Scrum Team estimate the efford required
    - Scrum Team commits to develop the customer requeriments in the form of **Approved, Estimated, and Committed User Stories**
  - **Processo 3 – Create Tasks:**
    - As histórias do usuário aprovadas e estimadas são divididas em subtarefas e escritas na **Task List**
    - As vezes, uma **Task Planning Meeting** is held for this purpose
  - **Processo 4 – Estimate Tasks**
    - Cada item na **Task List** é avaliado quanto ao esforço necessário -> Com isso se cria uma **Effort Estimated Task List**
  - **Processo 5 – Create Sprint Backlogs**
    - O time Scrum se reúne para decidir quais tasks devem ser cumpridos na SPRINT
- **Fase 3 – Implement:**
  - **Processo 1 – Create Deliverables**

- Nesse processo, o time Scrum trabalha nas Tasks para criar as Sprint Deliverables
- Um quadro Scrum pode ser usado para acompanhar o trabalho
- Dificuldades e problemas enfrentados pelo time Scrum são escritos no **Impediment Log**
- **Processo 2 – Conduct Daily Standup:**
  - Reunião altamente focada para discutir os problemas e falar sobre os progressos
- **Processo 3 – Groom Prioritized Product Backlog:**
  - **Prioritized Product Backlog** é atualizado constantemente e mantido
  - Reunião de **Prioritized Product Backlog** pode ser marcada
- **Fase 4 – Review and Retrospect**
  - **Processo 1: Convene Scrum of Scrum**

Os representantes dos Scrum se reúnem para manter track and update do progresso, problemas, impedimentos e dependência entre os times Scrum

- **Processo 2: Demonstrate and Validate Sprint**
  - Os entregáveis são mostrados para os Stakeholders
  - Essas reuniões são feitas para garantir a aprovação do produto desenvolvido e fazer os ajustes necessários o mais rápido possível
- **Processo 3: Retrospect Sprint**
  - O Scrum Master e o time Scrum se reúnem para discutir o que foi aprendido no SPRINT
  - Informação é documentada para futura SPRINTS
- **Fase 5 – Release**
  - **Processo 1 – Ship Deliverables**
    - Entregáveis são entregues para os devidos stakeholders
    - Um **Formal Working Working Deliverables Agreement** documenta o sucesso do SPRINT
  - **Processo 2 – Restrospect Project**
    - Stakeholders e Core Team fazem um retrospecto do projeto
    - Identificam, documentam e internalizam as lições aprendidas
    - Essas lições podem ser base de um **Agreed Actionable Improvements document** para futuros projetos

Chapter	Phase	Processes
8	Initiate	1. Create Project vision 2. Identify Scrum Master and Stakeholder(s) 3. Form Scrum Team 4. Develop Epic(s) 5. Create Prioritized Product Backlog 6. Conduct Release Planning
9	Plan and Estimate	7. Create User Stories 8. Approve, Estimate, and Commit User Stories 9. Create Tasks 10. Estimate Tasks 11. Create Sprint Backlog
10	Implement	12. Create Deliverables 13. Conduct Daily Standup 14. Groom Prioritized Product Backlog
11	Review and Retrospect	15. Convent Scrum of Scrums 16. Demonstrate and Validate Sprint 17. Retrospect Sprint
12	Release	18. Ship Deliverables 19. Retrospect Project

## Cap 10 – SCRUM VS Métodos tradicionais

### How the Project is Managed

	Scrum	Traditional
Emphasis	People	Process
Size	Small to large	Large
Domain	Unpredictable/ Exploratory	Predictable
Documentation	Minimal	Comprehensive
Process style	Iterative	Linear
Upfront Planning	Low	High
Perspective to Change	Adaptability	Sustainability
Management Style	Decentralized	Autocratic
Leadership	Collaborative	Command and control
Performance Measurement	Business value	Plan conformity
Returns on Investment	Early/throughout project life	End of project life

