Metodologias:

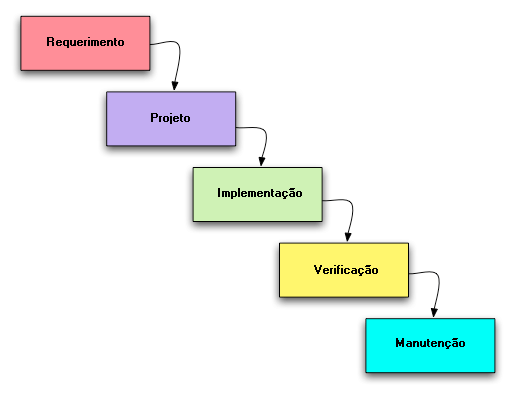
* Pesquisa para usar na prova:
* Modelos clássicos
* Metodologias ágeis
* Outros modelos (Alternativos ou não recomendados)

Precisa ter na pesquisa:

* Explicar como funciona e seus principais pontos
* Citar vantagens e desvantagens

Cascata:

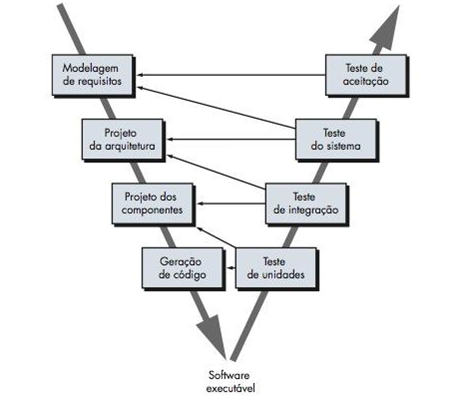
* Possui intuito de otimizar a gestão de projetos;
* Baseado em uma sequência hierárquica de etapas sequenciais;
* Possui controle fixo das funções;
* Possui algumas etapas como: levantamento de requisitos, planejamento, modelagem, desenvolvimento, testes e a implantação;
* Levantamento de requisitos: É a parte que fazemos um levantamento com perguntas de coisas que vão estar no projeto, quais as ferramentas devem ser usadas ou se fica a critério da equipe de desenvolvimento, coisas que ficaram como melhoria futura. Isso tudo é levantado de acordo com os critérios passados pelo cliente;
* Planejamento: O planejamento é uma das partes mais importantes do projeto, onde se começa conter as escolhas principais como por exemplo a divisão de papéis dentro do projeto, prazos de entrega de cada parte, orçamento do projeto;
* Modelagem: Dentro dessa etapa é estipulado os aspectos gerais do projeto como a modelagem dos dados no sistema, a metodologia, ferramentas/plataformas usadas e como seria trazer isso da vida real para um código de forma adepta e correta;
* Desenvolvimento: Se chegamos nessa etapa significa que na etapa anterior conseguimos chegar a um mapa de processo de desenvolvimento contendo os aspectos gerais, e como o nome diz essa hora começamos a dar vida ao nosso projeto trazendo ele um pouco mais perto para data de entrega com foco em entregar o solicitado acima;
* Testes: A parte de testes visa garantir qualidade e consistência do projeto, mais também encontrando erros e falhas que podem e devem ser melhorados antes de entregar ao cliente, a metodologia de teste depende da sua equipe e o que eles querem. Mas normalmente segue o padrão de olhar os requisitos e os documentos referentes ao escopo do seu projeto, com base nisso é feito os testes e avaliado;
* Implementação: Podemos chamar essa fase de fim, já que aqui o seu projeto ficou pronto, está funcionando e cumprindo com o que foi solicitado e agora irá ser implementado no seu real ambiente de uso, como a distribuição dele ao cliente e as futuras manutenções, atualizações e reparações. Essa fase de adaptação do seu cliente é algo bem importante por isso é recomendado o reforço do suporte ao cliente;
* Vantagens: Visibilidade sobre o projeto, clareza de cada função, documentação e registro, planejamento;
* Desvantagem: Modelo muito burocrático, rigidez das etapas, participação curta dos clientes devido que eles não podem interferir ou alterar nada no meio do projeto, fases seguintes só podem iniciar após as iniciais terminarem suas respectivas funções trazendo um longo período de tempo para a conclusão do projeto, inflexível;
* Imagem como exemplo:



* Referencia: https://casadaconsultoria.com.br/wpcontent/uploads/2016/09/etapas-modelo-cascata.png

V-Model:

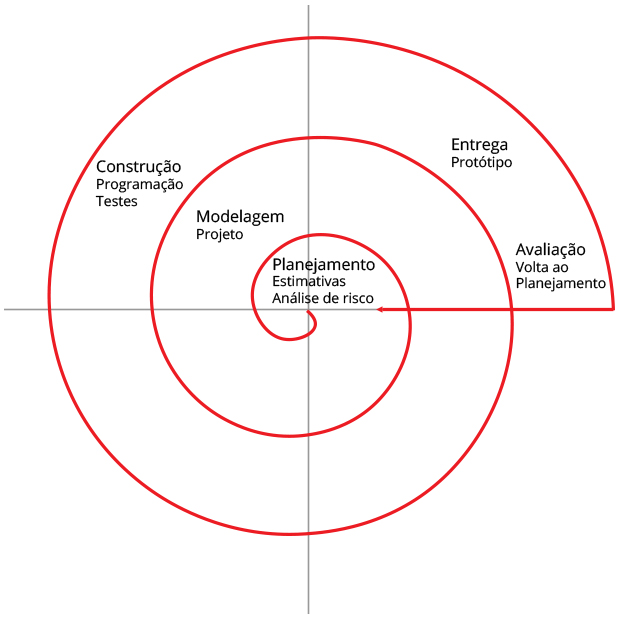
* O modelo em V foi uma melhoria da cascata, visando interligar o planejamento e construção com os testes organizados assim garantindo a consistência do produto;
* Ele possui fase ascendentes de baixo para cima no V e fase descendente associada de cima para baixo V como na imagem abaixo:



* Referência: https://arquivo.devmedia.com.br/artigos/Higor\_Medeiros/modelo-cascata/figura2.png
* Esse modelo tem como principal foco a qualidade que possui um ciclo, sendo ele a fase do design, implementação e validação;
* O modelo em v funciona da forma que o lado esquerdo seja a parte de definição e planejamento, o canto inferior sendo a implementação e o lado direito a verificação;
* Ele possui suas como:
* definição e planejamento: seria a parte dos requisitos de sistema, software, e o design;
* implementação: início de começar com base nos requisitos e os demais;
* verificação/validação: Onde realiza os testes tanto do quesito código, como design, verifica se atende os desejos dos clientes e definidos pela equipe;
* Vantagens: segurança, confiabilidade, início e fim bem definido em cada parte, alta qualidade, facilidade de gerenciar as partes, verificação e validação constantes podendo nos entregar um projeto cada vez mais bem trabalhado e funcionando.
* Desvantagens: Possui pouca flexibilidade, prazo de entrega extenso, custo bem alto inicialmente, pouca adaptabilidade
* É indicado usar esse modelo em projetos críticos com requisitos bem definidos, que precisam de uma segurança e confiabilidade boa, documentação bem detalhada e com normas rígidas;
* Deve evitar usá-lo em projetos que há mudanças ao longo do projeto, ou de feedback contínuo;

Modelo Spiral:

* É uma metodologia de desenvolvimento de software, ideal para projetos complexos e grandes;
* O modelo spiral tem como intuito ser utilizado em projetos que buscam sempre uma atualização, já que ele meio que “não possui um fim”;
* Conforme o projeto se avança a spiral se expande, nos mostrando a evolução até ter um resultado final



* Referencia: <https://lms.ev.org.br/mpls/Custom/Cds/COURSES/2510-PROJ_SIST_TI/pag/img/1_2_10a.jpg>
* No modelo spiral suas principais partes são divididas em 4:
* Definição de objetivos: Elas são crucias para definir as restrições, e preparar um plano todo do gerenciamento detalhando os possíveis riscos;
* Avaliação e redução de riscos: Para cada risco levantado na etapa anterior é levantado uma análise dele para descobrir uma forma de reduzir esse risco ou evitar o mesmo. Um exemplo seria a dificuldade em esclarecer um requisito, significando que há um risco de requisito inadequado, onde a forma de reagir nessa situação seria realizar um protótipo e apresentá-lo ao cliente coletando seu feedback e esclarecendo o que ele realmente desejae manter uma revisão continua;
* Avaliação tem como objetivo revisar o progresso do software, verificar se atende aos requisitos do cliente e avaliar a qualidade e os riscos.
* Além disso, serve para identificar ajustes necessários e reavaliar riscos, já que novos problemas podem surgir.
* O feedback obtido nessa fase é fundamental para orientar o planejamento da próxima iteração, garantindo que o projeto continue alinhado com os objetivos e as necessidades do cliente.
* Implementação e validação: Com os outros tópicos acima definidos agora é escolhido um modelo para desenvolvimento do projeto, pode-se usar modelos diferentes em cada volta onde esse tópico é refeito ou atualizado;
* Planejamento e especificação: Nessa parte e verificado tudo que já foi feito e planejar como serão as novas votas do spiral para o projeto;
* Vantagens: redução de riscos, tempo de implementação reduzido, versatilidade, não faz distinção entre o desenvolvimento e a manutenção, melhor uso em projetos grandes;
* Desvantagens: alta dificuldade de clientes que queiram usar esse modelo devido as mudanças, custo alto, prazo de entrega, e mão de obra;

Metodologias Ágeis

* Scrum: O scrum trabalha em ciclos, que conhecemos como sprints que possuem um tempo de 1 a 4 semanas, em cada sprint a equipe desenvolve e entrega um certo conjunto de coisas funcionais. Basicamente ele é uma forma de gerenciamento de projetos por meio de um conjunto de valore, práticas e princípios;
* A equipe é composta por um Product Owner (responsável por representar os interesses do cliente e definir o que é ou não prioridade), Scrum Master (responsável por facilitar o processo e resolver os obstáculos) e o time de desenvolvimento que executa/cria o software
* Características: sprints curtas e possui reuniões diárias;
* Seus eventos mais importantes são o Planejamento do Sprint (onde é definido o que será desenvolvido) reunião diária (que seria uma reunião bem curta para alinhar o time),Sprint Retrospective ( seria um tipo de reflexão para melhorar o processo),
* As ferramentas do Scrum são o product Backlog (cita todas as funcionalidades desejadas),Sprint Backlog (que ver o que deve ser feito em uma sprint atual), e o incremento (onde a parte funcional é entregue no fim da sprint);
* Deve ser usado em projetos dinâmicos, pequena equipe e colaborativa, e onde possui requisitos com constantes mudanças;

Kanban

* A metodologia é voltada na visualização do fluxo de trabalho e na eliminação de obstáculos;
* Tem foco na visualização do trabalho com o quadro Kanban, limitar o número de atividades em andamento, e melhorar o fluxo de trabalho;
* Ele funciona em colunas representando o que tem a fazer, em andamento e concluído;
* É usado em projetos que precisam de fluxo contínuo de trabalho e equipes que tem que trabalhar com suporte técnico e desenvolvimento contínuo;

XP (Extreme Programming)

* É uma metodologia mais voltada para o desenvolvimento com foco em qualidade e boas práticas no código;
* XP possui feedback constante com testes e revisões muito frequentes, a programação em pares que como o nome já diz são dois programadores trabalhando juntos em um único código, o desenvolvimento orientado a testes conhecido com TDD com base em escrever primeiro os testes e depois os códigos e as pequenas entregas frequentes com código funcional entregue rápido;
* Mais indicado em projetos que tem alta qualidade no código e muito teste continuo, e mais usados em equipes menores e focado no técnico;

Lean

* O Lean saiu a partir do sistema Toyota de produção que é focado em menos desperdícios e eficiência do projeto,
* Seus objetivos são na eliminação de perdas como dito acima, foca na entrega de produto que agregue valor ao cliente, em aprender algo e por ultimo na melhoria continua;
* É mais usado quando se pretende otimizar processos e redução de custos/perdas , e entregar algo rápido sem desperdícios;

Crystal

* A Crystal é uma metodologia de abrangência muito grande e isso foi projeto pelo criador dela que pensou exatamente que nos projetos a maioria das vezes eram necessários muitos tipos de abordagem e foi por isso que essa metodologia foi criada;
* Suas principais características são o foco na interação humana priorizando a comunicação direta e colaboração entre equipe, ela possui uma adaptação e o tamanho e quão critico é um projeto já que esse processo não é rígido e se adapta de acordo com o projeto, ele tem foco nas entregas frequentes de pequenas partes já prontas do projeto, possui menos burocracia minimizando a documentação extensa e as reuniões excessivas que são consideradas desnecessárias e possui a simplicidade de usar o necessário pra garantir o sucesso do projeto;
* A metodologia tem muitas abordagens de acordo com o tamanho da equipe e o projeto, sendo eles:
* Crystal Clear indicado para equipes de 1 a 6 pessoas, com nível crítico baixo e em pequenos projetos, com alta flexibilidade e praticamente não possui regras;
* Crystal Yellow indicado para equipes de 7 a 20 pessoas, com nível crítico médio e baixa burocracia e possui reuniões diárias;
* Crystal Orange indicado em equipes de 21 a 40 pessoas, com nível crítico alto, nesse modelo temos mais documentação e integração entre as equipes já que possui um time muito grande;
* Crystal Red é usado em equipes com mais de 40 pessoas e em níveis críticos muito altos;
* Tem como princípio Comunicação Pessoal e Direta, Reflexão e Melhoria Contínua, Entregas Frequentes, Segurança e Eficiência;