1. Os princípios de projeto estabelecem uma filosofia que prevalece sobre as atitudes e ações do desenvolvimento, orientando as atividades para realizar o projeto. Os conceitos de projeto devem ser estabelecidos e entendidos antes de aplicar a prática de projeto que deve levar a criação de varias representações do software que servem como uma guia para a atividade de construção que se segue.
2. Inicia-se após a análise e modelagem dos requisitos e tem como objetivo gerar um modelo ou representação que servirá como base para construção do produto.
3. Cada um dos elementos do modelo de requisitos fornece informações necessárias para criar os quatro modelos de projeto necessários para uma especificação completa. O modelo de requisitos, manifestado por elementos baseados em cenários, baseados em classes, orientado a fluxos e comportamentais, alimenta tarefa de projeto. O projeto gera um projeto de dados;/classes, um projeto de arquitetura, um projeto de interfaces e um projeto de componentes.
4. Projeto de dados/classes: transforma os modelos de classes em realizações de classes de projeto e nas estruturas de dados necessárias para implementação do projeto.

Projeto de arquitetura: Define as relações entre os principais elementos estruturais do software, estilos de arquitetura e padrões de projeto, considerando os requisitos e restrições.

Projeto de interfaces: descreve como o software se comunica com sistemas que operam em conjunto e com as pessoas que o utilizam.

Projeto de componentes: transforma elementos estruturais da arquitetura em uma descrição procedural dos componentes.

1. O projeto é a etapa na qual incorporamos qualidade ao software, fornece representações que nos permite transformar precisamente os requisitos em produto e serve como base para definir atividades de apoio da Engenharia de Software.
2. Deve implementar todos os requisitos explícitos contidos no modelo de requisitos e deve acomodar todos os requisitos implícitos.

Deve ser um guia legível e compreensível para aqueles que realizam testes e dão suporte ao software.

Deve fornecer uma visão completa do software, tratando dos domínios de dados, funcional e comportamental, do ponto de vista da implementação.

1. Modelagem é o processo de desenvolvimento de modelos abstratos de um sistema, representa o sistema com alguma notação gráfica (geralmente a Unified Modeling Language – UML), Os modelos são utilizados para ajudar a extrair os requisitos durante o processo de projeto, descrevem o sistema para os profissionais que o implementam, auxiliam nas atividades de documentação da estrutura e da operação do sistema.
2. UML se tornou uma linguagem de modelagem padrão para modelagem orientada a objetos. A UML tem muitos tipos de diagrama e, dessa forma, apoia a criação de muitos tipos de diferentes modelos de sistema. Os diagramas de atividades da UML podem ser utilizados para definir modelos de processo do negócio.
3. Cinco tipos de diagrama podem representar a essência de um sistema:

Diagramas de atividades: mostram as atividades envolvidas em um processo ou processamento de dados;

Diagramas de casos de uso: mostram as interações entre um sistema e seu ambiente;

Diagramas de sequência: mostram as interações entre os atores e o sistema e entre os componentes do sistema;

Diagramas de classe: mostram as classes de objeto no sistema e as associações entre elas;

Diagramas de estado: mostram como o sistema reage aos eventos internos e externos;

1. Mostram apenas os outros sistemas no ambiente, e não como o sistema que está sendo desenvolvido é usado, revela como o sistema é utilizado em processos de negócio mais amplos , os diagramas de atividades da UML podem ser utilizados para definir modelos de processo do negócio.

11. AAAFE

12. A modelagem de interação do usuário é importante para auxiliar na identificação dos requisitos, destaca problemas de comunicação que possam vir a surgir, Ajuda a compreender se uma estrutura proposta de sistema pode fornecer desempenho e confiança requeridas pelo sistema, digramas de caso de uso e diagramas de sequência podem ser utilizados para modelagem de interação.

13. Desenvolvidos para auxiliar na elicitação de requisitos, cada caso de uso representa uma tarefa distinta, que envolve interação externa com um sistema, os atores de um caso de uso podem ser pessoas ou outros sistemas, são representados esquematicamente para fornecer a visão geral de um caso de uso e em forma textual mais detalhada.

16. Usados para modelar as interações entre os atores e os objetos no sistema, Mostram a sequência de interações que ocorrem durante um caso de uso em particular ou uma instância de um caso de uso, Os objetos e atores envolvidos são listados na parte superior do diagrama, com divisões verticais pontilhadas a partir deles, As interações são indicadas por setas anotadas.

18. Mostram a organização de um sistema em termo de seus componentes, que compõe este sistema e seus relacionamentos, Podem ser modelos estáticos, que mostram a estrutura do projeto do sistema, ou modelos dinâmicos, que mostram a organização do sistema em execução, Modelos estruturais são criados durante o projeto de arquitetura.

19. São usados quando se está desenvolvendo um modelo de sistema orientado a objetos, para demonstrar suas classes e as associações entre elas, uma classe de objeto pode ser considerada como uma definição geral de um tipo de objeto do sistema, uma associação é uma ligação entre as classes que indica que há algum relacionamento entre essas classes, objetos representam algo do mundo real: um paciente, uma receita médica, um médico, etc.

22. Técnica para gerenciar complexidade, ao invés de aprender as características detalhadas em todas as entidades, classificamos as entidades em classes mais gerais e aprendemos com as características de cada classe, a generalização nos permite supor que membros diferentes destas classes possuem características comuns.

23. São modelos do comportamento dinâmico do sistema enquanto este está em execução, Mostram o que acontece ou o que deveria acontecer quando um sistema responde a um estímulo do seu ambiente.

Podem-se considerar estímulos como sendo de dois tipos:

Dados: alguns dados que chegam e precisam ser processados pelo sistema.

Eventos: alguns eventos que acontecem disparam o processamento do sistema. Os eventos podem ter dados associados, embora não ocorra sempre.

24. Vários sistemas de negócios são sistemas de processamento de dados, dirigidos principalmente a dados, São controlados pela entrada de dados no sistema com quase nenhum processamento de eventos externos, os modelos dirigidos a dados mostram a sequência de ações envolvidas no processamento dos dados que entram, e na geração de uma saída associada, são particularmente úteis para a análise de requisitos por poderem ser usados para mostrar o processamento fim-a-fim de um sistema.

25. Sistemas em tempo real geralmente são dirigidos a eventos, com um mínimo de processamento de dados, a modelagem dirigida a eventos mostra como um sistema responde a eventos internos e externos, baseia-se na suposição de que um sistema tem um número limitado de estados e que os eventos (estímulos) podem causar transição de um estado para outro.