

GPS - Gestão de projeto de software

**Professor
Emiliano S.
Monteiro**

1. Levantamento de requisitos

1.1. Requisitos do usuário e casos de uso

1.2. Requisitos de sistema

2. Análise, avaliação e aprovação

2.1. Estudo de viabilidade/análise de risco

2.2. Escolha de metodologia de desenvolvimento

2.3. Escolha da equipe de projeto

2.4. Escrita do TAP - Termo de abertura de projeto

3. Projeto

3.1. Projeto da arquitetura do sistema

3.2. Projeto do banco de dados

4. Instalação de Configuração de ambientes

4.1. Ambiente de desenvolvimento

4.2. Ambiente de testes

4.3. Ambiente de produção

5. Desenvolvimento

5.1. Desenvolvimento de telas e protótipo

5.2. Codificação e versionamento

6. Testes

6.1. Teste unitários

6.2. Teste de integração

6.3. Teste de usabilidade

7. Documentação

7.1. Documentação para o usuário final

7.2. Documentação para o administrador do sistema

7.3. Desenvolvimento de material de treinamento

8. Treinamento

8.1. Treinamento para administradores do sistema

8.2. Treinamento para usuários finais

8.3. Disponibilização de treinamento on-line

9. Operações

9.1. Instalação e configuração

10. Escrita do TEP - Termo de encerramento de projeto

11. Manutenção

12. Avaliação

1. ALM Octane da **Microfocus**

2. Ca Agile Central

3. Caliber **Microfocus**

4. IBM Rational DOORS Next Generation

5. **Visual Trace** Spec

6. Casecomplete

7. codeBeamer ALM Requirements Management

8. Kovair ALM Studio

9. Modern Requirements4TFS

10. objectiF RM

11. OneDesk Product Management

12. **MODELIO** BA

13. Polarion REQUIREMENTS da Siemens

14. Requirement One

15. Yonix

Bizagi é para modelagem BPMN

Brainstorming → caso de uso → StarUML

Mapa mental

Problema

Palavras chaves

1. Levantamento de requisitos
 - 1.1. Requisitos do usuário e casos de uso
 - 1.2. Requisitos de sistema
- 2. Análise, avaliação e aprovação**
 - 2.1. Estudo de viabilidade/análise de risco**
 - 2.2. Escolha de metodologia de desenvolvimento**
 - 2.3. Escolha da equipe de projeto**
 - 2.4. Escrita do TAP - Termo de abertura de projeto**
3. Projeto
 - 3.1. Projeto da arquitetura do sistema
 - 3.2. Projeto do banco de dados
4. Instalação de Configuração de ambientes
 - 4.1. Ambiente de desenvolvimento
 - 4.2. Ambiente de testes
 - 4.3. Ambiente de produção
5. Desenvolvimento
 - 5.1. Desenvolvimento de telas e protótipo
 - 5.2. Codificação e versionamento
6. Testes
 - 6.1. Teste unitários
 - 6.2. Teste de integração
 - 6.3. Teste de usabilidade
7. Documentação
 - 7.1. Documentação para o usuário final
 - 7.2. Documentação para o administrador do sistema
 - 7.3. Desenvolvimento de material de treinamento
8. Treinamento
 - 8.1. Treinamento para administradores do sistema
 - 8.2. Treinamento para usuários finais
 - 8.3. Disponibilização de treinamento on-line
9. Operações
- 9.1. Instalação e configuração
10. Escrita do TEP - Termo de encerramento de projeto
11. Manutenção
12. Avaliação



1. Criação de cenários, Ensaios, estudos em meios de comunicação, análises de mercado, análise da concorrência, contratação de consultores, opiniões de assessores, estudo de produtos similares, observação da concorrência, mapas mentais, relatórios operacionais e gerenciais.
2. Escolha de metodologia: RUP, Agile, RAD, etc.
3. Ferramenta para avaliação de recursos humanos: Skill Base, TalentSoft, Rexx Systems, Coppelis SkillsMinds, SparkHire, Codility, OrangeHRM e Sentrifugo.
4. A escrita do TAP ser em qualquer editor de texto ou no próprio sistema de projetos, inclusive existem templates prontos disponíveis gratuitamente na web.

1. Levantamento de requisitos
 - 1.1. Requisitos do usuário e casos de uso
 - 1.2. Requisitos de sistema
2. Análise, avaliação e aprovação
 - 2.1. Estudo de viabilidade/análise de risco
 - 2.2. Escolha de metodologia de desenvolvimento
 - 2.3. Escolha da equipe de projeto
 - 2.4. Escrita do TAP - Termo de abertura de projeto
- 3. Projeto**
 - 3.1. Projeto da arquitetura do sistema**
 - 3.2. Projeto do banco de dados**
4. Instalação de Configuração de ambientes
 - 4.1. Ambiente de desenvolvimento
 - 4.2. Ambiente de testes
 - 4.3. Ambiente de produção
5. Desenvolvimento
 - 5.1. Desenvolvimento de telas e protótipo
 - 5.2. Codificação e versionamento
6. Testes
 - 6.1. Teste unitários
 - 6.2. Teste de integração
 - 6.3. Teste de usabilidade
7. Documentação
 - 7.1. Documentação para o usuário final
 - 7.2. Documentação para o administrador do sistema
 - 7.3. Desenvolvimento de material de treinamento
8. Treinamento
 - 8.1. Treinamento para administradores do sistema
 - 8.2. Treinamento para usuários finais
 - 8.3. Disponibilização de treinamento on-line
9. Operações
 - 9.1. Instalação e configuração
10. Escrita do TEP - Termo de encerramento de projeto
11. Manutenção
12. Avaliação



1. Desktop open source: OpenProj, dotProject, Open Workbench, GanttProject, jsProject, ProjectLibre
2. Desktop/server: Microsoft Project, **CA PPM**.
3. Open source web: **qdPM**, Feng Office, eyeOS, Collabtive, dotProject, ProjectPier, **Mantis Bug Tracker**, Rukovoditel, The Bug Genie, PHProjekt, TaskFreak, todayu, Flyspray, **Kanboard**, phpCollab, Snipe-IT, Admidio, SOPlanning, Traq, **ZenTao**, Eventum, Bugs, ProjeQtOr, WebCollab, TestLink, Manage Your Team, Burden
4. Ferramentas para desenho: **Visio**, Project Pencil, Dia, Umbrello, StarUML.
5. Ferramentas para modelagem de dados: **MySQLWorkbench**, ERWin, Embarcadero ERStudio, Fabforce designer
6. Nclass eng. Reversa
7. Scriptcase → gera diagramas do que já feito!
8. Adianti Builder & Workana → DER
9. Enterprise Manager SQL Server → DER

1. Levantamento de requisitos
 - 1.1. Requisitos do usuário e casos de uso
 - 1.2. Requisitos de sistema
2. Análise, avaliação e aprovação
 - 2.1. Estudo de viabilidade/análise de risco
 - 2.2. Escolha de metodologia de desenvolvimento
 - 2.3. Escolha da equipe de projeto
 - 2.4. Escrita do TAP - Termo de abertura de projeto
3. Projeto
 - 3.1. Projeto da arquitetura do sistema
 - 3.2. Projeto do banco de dados
- 4. Instalação de Configuração de ambientes**
 - 4.1. Ambiente de desenvolvimento**
 - 4.2. Ambiente de testes**
 - 4.3. Ambiente de produção**
5. Desenvolvimento
 - 5.1. Desenvolvimento de telas e protótipo
 - 5.2. Codificação e versionamento
6. Testes
 - 6.1. Teste unitários
 - 6.2. Teste de integração
 - 6.3. Teste de usabilidade
7. Documentação
 - 7.1. Documentação para o usuário final
 - 7.2. Documentação para o administrador do sistema
 - 7.3. Desenvolvimento de material de treinamento
8. Treinamento
 - 8.1. Treinamento para administradores do sistema
 - 8.2. Treinamento para usuários finais
 - 8.3. Disponibilização de treinamento on-line
9. Operações
- 9.1. Instalação e configuração
10. Escrita do TEP - Termo de encerramento de projeto
11. Manutenção
12. Avaliação



1. Microsoft System Center (**Configuration Manager**, **Operation Manager**, Hiper-v Server, Service Manager e Orchestrator).
2. AWS CodeDeploy, Bamboo, Octopus **Deploy**.
3. Microsoft Azure.

1. Levantamento de requisitos
 - 1.1. Requisitos do usuário e casos de uso
 - 1.2. Requisitos de sistema
2. Análise, avaliação e aprovação
 - 2.1. Estudo de viabilidade/análise de risco
 - 2.2. Escolha de metodologia de desenvolvimento
 - 2.3. Escolha da equipe de projeto
 - 2.4. Escrita do TAP - Termo de abertura de projeto
3. Projeto
 - 3.1. Projeto da arquitetura do sistema
 - 3.2. Projeto do banco de dados
4. Instalação de Configuração de ambientes
 - 4.1. Ambiente de desenvolvimento
 - 4.2. Ambiente de testes
 - 4.3. Ambiente de produção
- 5. Desenvolvimento**
 - 5.1. Desenvolvimento de telas e protótipo**
 - 5.2. Codificação e versionamento**
6. Testes
 - 6.1. Teste unitários
 - 6.2. Teste de integração
 - 6.3. Teste de usabilidade
7. Documentação
 - 7.1. Documentação para o usuário final
 - 7.2. Documentação para o administrador do sistema
 - 7.3. Desenvolvimento de material de treinamento
8. Treinamento
 - 8.1. Treinamento para administradores do sistema
 - 8.2. Treinamento para usuários finais
 - 8.3. Disponibilização de treinamento on-line
9. Operações
- 9.1. Instalação e configuração
10. Escrita do TEP - Termo de encerramento de projeto
11. Manutenção
12. Avaliação



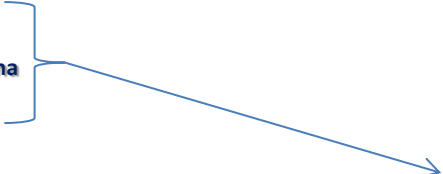
1. Ferramentas de desenvolvimento RAD que possam criar GUIs
2. Axure RP 8
3. Project Pencil, Visio, etc...
4. Ferramentas CASE

1. Levantamento de requisitos
 - 1.1. Requisitos do usuário e casos de uso
 - 1.2. Requisitos de sistema
2. Análise, avaliação e aprovação
 - 2.1. Estudo de viabilidade/análise de risco
 - 2.2. Escolha de metodologia de desenvolvimento
 - 2.3. Escolha da equipe de projeto
 - 2.4. Escrita do TAP - Termo de abertura de projeto
3. Projeto
 - 3.1. Projeto da arquitetura do sistema
 - 3.2. Projeto do banco de dados
4. Instalação de Configuração de ambientes
 - 4.1. Ambiente de desenvolvimento
 - 4.2. Ambiente de testes
 - 4.3. Ambiente de produção
5. Desenvolvimento
 - 5.1. Desenvolvimento de telas e protótipo
 - 5.2. Codificação e versionamento
- 6. Testes**
 - 6.1. Teste unitários**
 - 6.2. Teste de integração**
 - 6.3. Teste de usabilidade**
7. Documentação
 - 7.1. Documentação para o usuário final
 - 7.2. Documentação para o administrador do sistema
 - 7.3. Desenvolvimento de material de treinamento
8. Treinamento
 - 8.1. Treinamento para administradores do sistema
 - 8.2. Treinamento para usuários finais
 - 8.3. Disponibilização de treinamento on-line
9. Operações
 - 9.1. Instalação e configuração
10. Escrita do TEP - Termo de encerramento de projeto
11. Manutenção
12. Avaliação

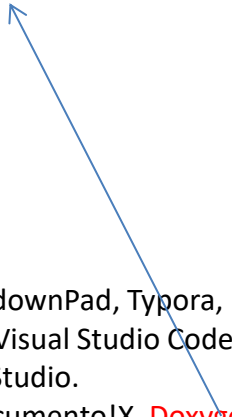


xUnit, DUnit, JUnit, unittest, mstest, nunit

- 1. Levantamento de requisitos
 - 1.1. Requisitos do usuário e casos de uso
 - 1.2. Requisitos de sistema
- 2. Análise, avaliação e aprovação
 - 2.1. Estudo de viabilidade/análise de risco
 - 2.2. Escolha de metodologia de desenvolvimento
 - 2.3. Escolha da equipe de projeto
 - 2.4. Escrita do TAP - Termo de abertura de projeto
- 3. Projeto
 - 3.1. Projeto da arquitetura do sistema
 - 3.2. Projeto do banco de dados
- 4. Instalação de Configuração de ambientes
 - 4.1. Ambiente de desenvolvimento
 - 4.2. Ambiente de testes
 - 4.3. Ambiente de produção
- 5. Desenvolvimento
 - 5.1. Desenvolvimento de telas e protótipo
 - 5.2. Codificação e versionamento
- 6. Testes
 - 6.1. Teste unitários
 - 6.2. Teste de integração
 - 6.3. Teste de usabilidade
- 7. Documentação**
 - 7.1. Documentação para o usuário final**
 - 7.2. Documentação para o administrador do sistema**
 - 7.3. Desenvolvimento de material de treinamento**
- 8. Treinamento
 - 8.1. Treinamento para administradores do sistema
 - 8.2. Treinamento para usuários finais
 - 8.3. Disponibilização de treinamento on-line
- 9. Operações
- 9.1. Instalação e configuração
- 10. Escrita do TEP - Termo de encerramento de projeto
- 11. Manutenção
- 12. Avaliação



```
/*  
Asdflajflkdsfj  
*/
```




Editores: Microsoft Office, MarkdownPad, Typora, LibreOffice, Haroopad, TextPad, Visual Studio Code, Plugins para RadStudio e Visual Studio.
Documentadores: Helpndoc, Documento!X, **Doxygen**, NDoc, JavaDoc, Doc-O-Matic, SharpDox, HelpManual, Epsilon Content.

1. Levantamento de requisitos
 - 1.1. Requisitos do usuário e casos de uso
 - 1.2. Requisitos de sistema
2. Análise, avaliação e aprovação
 - 2.1. Estudo de viabilidade/análise de risco
 - 2.2. Escolha de metodologia de desenvolvimento
 - 2.3. Escolha da equipe de projeto
 - 2.4. Escrita do TAP - Termo de abertura de projeto
3. Projeto
 - 3.1. Projeto da arquitetura do sistema
 - 3.2. Projeto do banco de dados
4. Instalação de Configuração de ambientes
 - 4.1. Ambiente de desenvolvimento
 - 4.2. Ambiente de testes
 - 4.3. Ambiente de produção
5. Desenvolvimento
 - 5.1. Desenvolvimento de telas e protótipo
 - 5.2. Codificação e versionamento
6. Testes
 - 6.1. Teste unitários
 - 6.2. Teste de integração
 - 6.3. Teste de usabilidade
7. Documentação
 - 7.1. Documentação para o usuário final
 - 7.2. Documentação para o administrador do sistema
 - 7.3. Desenvolvimento de material de treinamento
8. Treinamento
 - 8.1. Treinamento para administradores do sistema
 - 8.2. Treinamento para usuários finais
 - 8.3. Disponibilização de treinamento on-line
- 9. Operações**
- 9.1. Instalação e configuração
10. Escrita do TEP - Termo de encerramento de projeto
- 11. Manutenção**
12. Avaliação

Gerir O.S.



qDPM, Mantis, Microsoft Service Manager

1. Levantamento de requisitos
 - 1.1. Requisitos do usuário e casos de uso
 - 1.2. Requisitos de sistema
2. Análise, avaliação e aprovação
 - 2.1. Estudo de viabilidade/análise de risco
 - 2.2. Escolha de metodologia de desenvolvimento
 - 2.3. Escolha da equipe de projeto
 - 2.4. Escrita do TAP - Termo de abertura de projeto
3. Projeto
 - 3.1. Projeto da arquitetura do sistema
 - 3.2. Projeto do banco de dados
4. Instalação de Configuração de ambientes
 - 4.1. Ambiente de desenvolvimento
 - 4.2. Ambiente de testes
 - 4.3. Ambiente de produção
5. Desenvolvimento
 - 5.1. Desenvolvimento de telas e protótipo
 - 5.2. Codificação e versionamento
6. Testes
 - 6.1. Teste unitários
 - 6.2. Teste de integração
 - 6.3. Teste de usabilidade
7. Documentação
 - 7.1. Documentação para o usuário final
 - 7.2. Documentação para o administrador do sistema
 - 7.3. Desenvolvimento de material de treinamento
8. Treinamento
 - 8.1. Treinamento para administradores do sistema
 - 8.2. Treinamento para usuários finais
 - 8.3. Disponibilização de treinamento on-line
9. Operações
- 9.1. Instalação e configuração 
10. Escrita do TEP - Termo de encerramento de projeto
11. Manutenção
12. Avaliação

1. Microsoft system Center (Configuration Manager, Operation Manager, Hiper-v Server, Service Manager e Orchestrator).
2. AWS CodeDeploy, Bamboo, Octopus Deploy.

Introdução

Sobre riscos podemos:

Ou aceitar

A aceitação do risco é justificável quando o custo para a implementar um controle é maior que a ocorrência do evento(desastre)

Ou reduzir

Não significa eliminar

Ou amenizar/compensar (comprar um seguro)

Análise de risco

- É o processo para identificar, analisar, avaliar, suavizar ou transferir o risco é chamado de gerência de risco
- Questões para pensar:
 - O que pode acontecer (evento ou ameaça)?
 - Se for acontecer, quão ruim poderia ser (impacto da ameaça)?
 - Com que frequência poderia acontecer?

Introdução

- ▶ Incerteza: incapacidade de prever resultado corretos
- ▶ É essencial que o processo de análise e avaliação de risco não seja visto como auditoria (auditoria é diferente de análise de risco, o produto final)
- ▶ Análise de risco = identificação das ameaças e vulnerabilidades
- ▶ Avaliação do risco = avaliar o grau de impacto do risco
- ▶ Gerenciamento do risco = é o processo completo de análise, avaliação, priorização e implementação dos controles

Risco

- ▶ Chance de desastre ou perda
- ▶ Possibilidade de ocorrência de perigo
- ▶ Possibilidade da vulnerabilidade ser explorada por uma ameaça
- ▶ Ativo = “coisas” para as quais a empresa determina um valor e demanda proteção
- ▶ Ameaças = eventos ou ocorrências que podem comprometer aspectos básicos de segurança
- ▶ Vulnerabilidades = fragilidades associadas as informações e seus ativos, no qual os mesmos são suscetíveis a ataques.
- ▶ Risco = potencial de uma ameaça explore a vulnerabilidade
- ▶ Severidade = é o grau do dano no ativo

Exemplos

- Processo = procedimentos para cálculo de férias em folha de pagamento ou plantas de engenharia, etc...
- Ativo = notebook
- Vulnerabilidade = estação não gera logs
- Ameaças = roubo de informações em pendrives
- Impactos = perda de segredos para

Cálculo do risco

Risco = probabilidade de ocorrência do evento x custo do evento.

Identifique Ameaças

O primeiro passo de uma análise de risco é a identificação de ameaças, estas podem ser:

Humanos - indivíduos ou organizações, etc.

Operacional – pela interrupção de força, perda de acesso aos bens, falhas em distribuição de serviços diversos, etc.

Reputação – pela perda de um parceiro de negócio ou funcionário de confiança.

Procedimental/funcional – a partir de falhas em contabilidade, sistemas internos de controle, fraudes, etc...

Diversos – insuficiência na qualidade de produtos ou serviços, etc.

Financeiros – falha do modelo de negócios, bolsa, queda de taxas de juros, desemprego, etc....

Tecnico – pela perda da vanguarda tecnológica, etc.

Natureza – tempestado, enchentes, furação, etc....

Politico – pelas mudanças em regimes, política governamental, etc...

Outros

Ameaças (threats)

A análise de ameaças é importante porque é fácil olhar as ameaças. Uma forma de agrupá-las é usando diferentes abordagens:

- ▶ Primeiro, primeiro veja em uma lista como a anterior para ver se acha uma
- ▶ Segundo, pense nos sistemas, organização ou estrutura operacional, e nos riscos de cada uma dessas partes
- ▶ Veja se você pode identificar qualquer vulnerabilidade dentro de cada sistema ou estrutura
- ▶ Pergunte (entrevista ou questionários) a outras pessoas, pois elas poderão ter diferentes perspectivas organizacionais sobre as vulnerabilidades.

Estimativa de Risco

- ▶ Uma vez que tenha identificado as ameaças que você enfrenta, o próximo passo é trabalhar as probabilidades de que estes ativos sejam atingidos pela exploração das suas vulnerabilidades.
- ▶ Um abordagem é fazendo a melhor estimativa da probabilidade de ocorrência do evento, e multiplicando essa probabilidade pelo custo do ativo ser reconfigurado se o evento ocorrer, isto lhe dará o valor do risco.

Gerenciando Riscos

- Uma vez detalhado o valor do risco, você poderá procurar as formas de gerenciamento (proteção dos ativos). É importante notar que devem ser adotadas soluções que compense o custo do ativo.
- Não é interessante gastar mais numa solução de proteção do que o próprio valor do bem, em alguns casos é melhor correr o risco.

O risco pode ser gerenciado de várias formas

- ▶ Pelo simples uso dos mesmos ativos:
 - ▶ É a reconfiguração do ativo, mudança de métodos de acesso, mudança na responsabilidade dos usuários, implementando contabilidade, etc....
- ▶ Usando um plano de contingenciamento:
 - ▶ Se você decidiu aceitar o risco, mas decidiu também desenvolver uma plano para minimizar o impacto. Um bom plano de contingenciamento lhe permitirá realizar ações de forma imediata.
- ▶ Investindo em novos recursos (ativos) :
 - ▶ Uma vez identificados os riscos, é hora de decidir se haverá gastos de dinheiro para a aquisição de novos equipamentos de proteção (gastos com seguro ou terceirização).

Halt and catch fire

Revisões

- ▶ Uma vez realizada a análise de risco e exercício de gerenciamento, é válido realizar revisões gerais no trabalho executado, pois as ameaças e vulnerabilidades podem mudar. É interessante que sejam realizadas revisões regulares ou o uso de sistemas automatizados de revisão.
- ▶ Ferramentas para análise de vulnerabilidades: Tenable Nessus, GFI Languard, etc...

Membros da equipe de análise de riscos

- ▶ Security Officer
- ▶ Gerentes de operações de redes (administradores de rede)
- ▶ Analistas de sistemas
- ▶ Gerentes de segurança física
- ▶ Gerente CIPA
- ▶ Principais usuários de sistemas
- ▶ Programadores de sistemas
- ▶ Em alguns casos proprietários da empresa

Atividades centrais da gerência do risco das informações

- Estabelecer uma política de gerência de risco
- Estabelecer uma equipe
- Estabelecer a metodologia e ferramentas
- Mensuração: análise de ameaças, identificação e avaliação dos ativos, análise de vulns, avaliação dos riscos
- Relatórios e recomendações

Qual o valor do ativo?

- ▶ Ativo pode ser tangível ou intangível
- ▶ Depende de....
- ▶ Valor de reposição da informação
- ▶ Valor da informação no mercado
- ▶ Valor de reposição do hardware / software
- ▶ Custo da perda de confidencialidade
- ▶ Custo da perda da integridade
- ▶ Custo da perda da disponibilidade
- ▶ Custo da perda de credibilidade da empresa