

---

**Suppera Solutions**

---

## **Modelo de Classe e Dados**

**Versão <1.0>**

<i>mPrime</i>	Versão: 1.0
Modelo de Classe e de Dados	Data da Versão: 26/06/2005
Suppera_ModeloDeClassesEDados.doc	

## Histórico de Revisões

<b>Data</b>	<b>Versão</b>	<b>Descrição</b>	<b>Autor</b>
26/06/2005	1.0	Elaboração do documento de Modelo de Classe e Dados	Thiago Fernandes

<i>mPrime</i>	Versão: 1.0
Modelo de Classe e de Dados	Data da Versão: 26/06/2005
Suppera_ModeloDeClassesEDados.doc	

## Conteúdo

1.	Introdução	4
2.	Escopo	4
3.	Estórias do Usuário contempladas	4
3.1	Primeira Iteração	4
3.2	Segunda Iteração	4
4.	Testes de Aceitação	4
5.	Referências	8

mPrime	Versão: 1.0
Modelo de Classe e de Dados	Data da Versão: 26/06/2005
Suppera_ModeloDeClassesEDados.doc	

## Plano de Testes de Aceitação

### 1. Introdução

Este documento tem como objetivo apresentar o Modelo de Classe e de Dados do primeiro release do projeto mPrime, um Framework que dará suporte ao processo de Gestão de Risco para Ambientes de Múltiplos Projetos de Desenvolvimento de Software, e será implementado como um add-in para o Microsoft Project.

A arquitetura, aqui apresentada, contém a forma final da modelagem do sistema para este release, ela foi desenvolvida incrementalmente, de acordo com a implementação das histórias dos usuários em cada iteração.

### 2. Escopo

Os testes deste documento são referentes ao primeiro release, que será dividido em cinco iterações. Cada uma delas consiste na implementação das histórias descritas pelo usuário/cliente pela ordem de prioridade. Ou seja, as primeiras tarefas a serem implementadas são as de maior prioridade. Estas histórias são todas apresentadas numa reunião inicial, mas seus detalhamentos, definição de requisitos e de testes só serão realizados antes do início da implementação de cada uma delas.

Os testes de aceitação são produzidos antes mesmo da própria implementação da história, induzindo um desenvolvimento direcionado a cobrir o maior número de casos possíveis. Isto porque, antes de codificar, o programador deverá pensar em todas as possibilidades de fluxo. Além disso, após os testes, as correções de código necessárias são feitas seguidas de uma nova sessão de testes, formando um ciclo entre alterações e os testes das mesmas. Isto gera um código mais confiável e otimizado, conforme a metodologia de desenvolvimento adotada (eXtreme Programming).

De acordo com o que foi exposto acima, este documento deverá ser incrementado a cada nova iteração do sistema. Logo, ele só será concluído ao final do release.

### 3. Estórias do Usuário contempladas

Como processo de desenvolvimento está dividido em cinco iterações, a descrição das histórias será feito de acordo com esta divisão. Esta é uma técnica de eXtreme Programming que tenta minimizar ao máximo a consequência do risco de mudança de requisitos. Desta forma, este documento será feito de forma incremental e iterativa, de forma que a cada iteração as histórias a serem implementadas sejam inseridas neste documento.

#### 3.1 Primeira Iteração

Número	Nome	Descrição
02	Identificar Riscos	Determinar quais riscos são prováveis de afetar o projeto e procede a documentação das características de cada um deles.
03	Quantificar Riscos	Analisa numericamente a probabilidade de ocorrência do risco e suas consequências nos objetivos do projeto, bem como a extensão do risco global do projeto.
04	Qualificar Riscos	Avalia qual a probabilidade de ocorrência e o grau de impacto dos riscos identificados no Plano de Gestão de Risco. Prioriza o risco de acordo com o seu efeito potencial sobre os objetivos

mPrime	Versão: 1.0
Modelo de Classe e de Dados	Data da Versão: 26/06/2005
Suppera_ModeloDeClassesEDados.doc	
	do projeto .

### 3.2 Segunda Iteração

Número	Nome	Descrição
05	Responder Riscos	Desenvolver planos que possam sempre que possível eliminar o risco.
07	Monitorar Riscos	Desenvolver rotinas de acompanhamento da execução do projeto, de forma a espelhar a realidade e as tendências de risco ao longo do ciclo de vida do projeto.

## 4. Testes de Aceitação

Para cada estória de usuário contemplada neste documento, um conjunto de testes de aceitação é identificado pelo cliente e pelo Testador. No documento esses testes devem ser catalogados seguindo o template abaixo

Estória	Identificar Riscos	
Número	Ação	Resultado Esperado
01	Ao selecionar no menu mPrime, inserir riscos.	Mostrar um pequeno Wizard com os alguns passos para a inserção de um risco.
02	Ao clicar no botão inserir riscos da mPrime Toolbar.	Mostrar um pequeno Wizard com os alguns passos para a inserção de um risco.
03	Ao clicar na opção inserir riscos dentro de da Tab “Tasks” (Left Pane).	Mostrar um pequeno Wizard com os alguns passos para a inserção de um risco.
04	Ao selecionar no menu mPrime, inserir riscos a partir da WBS.	Inserir riscos no projeto baseado nas tarefas definidas pelo usuário (Tipo do Risco, Tarefa Associada, Classe, Atributo, Elemento).
05	Ao clicar no botão “inserir riscos a partir da WBS” da mPrime Toolbar.	Inserir riscos no projeto baseado nas tarefas definidas pelo usuário (Tipo do Risco, Tarefa Associada, Classe, Atributo, Elemento).
06	Ao preencher o wizard para a inserção de um risco.	O usuário primeiro escolherá a classe do risco, em seguida o elemento e por último o atributo (Sendo possível ao usuário definir novas classes, elementos e atributos). Depois o usuário escolherá qual o tipo de risco (também pode ser definido um novo). E por último ele define a que tarefa(s) o risco no projeto

mPrime	Versão: 1.0
Modelo de Classe e de Dados	Data da Versão: 26/06/2005
Suppera_ModeloDeClassesEDados.doc	

		está associado
Estória	Quantificar Riscos	
Número	Ação	Resultado Esperado
01	Ao preencher o wizard para a inserção de um risco.	O usuário também deverá definir a probabilidade do risco acontecer.

Estória	Qualificar Riscos	
Número	Ação	Resultado Esperado
01	Ao preencher o wizard para a inserção de um risco.	O usuário também deverá definir o impacto do risco ao projeto.

Estória	Responder Riscos	
Número	Ação	Resultado Esperado
01	Ao preencher o wizard para a inserção de um risco.	O usuário também deverá definir recurso responsável pelo risco e também deverá definir o nível de tolerância do Projeto ao Risco (Evitar Risco, Transferir Risco, Mitigar Risco e Aceitar Risco).
02	Caso usuário escolha “Evitar Risco”	O usuário deverá deifinir ações associadas e planos de contingência.
03	Caso usuário escolha “Transferir Risco”	O usuário deve definir quem deve ser o responsável por aquele risco.
04	Caso usuário escolha “Mitigar Risco”	O usuário deverá deifinir ações associadas.
05	Caso usuário escolha “Aceitar Risco” e probabilidade de ocorrência do risco seja maior que 50%	O usuário deve ser alertado sobre esta situação, pois alguma providência deve ser tomada.
06	Ao selecionar no menu mPrime, Matriz de Riscos.	Deverá aparecer no active view do usuário, uma tabela onde será mostrado todos os riscos do projeto (com seus campos), essa tabela não será editável. E mais a direita será montada a árvore de riscos do SEI.
07	Ao clicar no botão “Matriz de Riscos” da mPrime Toolbar.	Deverá aparecer no active view do usuário, uma tabela onde será mostrado todos os riscos do projeto (com seus campos), essa tabela não será editável. E mais a direita será montada a árvore de riscos do SEI.
08	Ao clicar na opção “Matriz de	Deverá aparecer no active view do usuário, uma tabela onde será

mPrime	Versão: 1.0
Modelo de Classe e de Dados	Data da Versão: 26/06/2005
Suppera_ModeloDeClassesEDados.doc	

	Riscos” dentro de da Tab “Reports” (Left Pane).	mostrado todos os riscos do projeto (com seus campos), essa tabela não será editável. E mais a direita será montada a árvore de riscos do SEI.
--	---	--

Estória	Monitorar Riscos	
Número	Ação	Resultado Esperado
01	Quando usuário estiver na matriz de riscos, e clicar no left pane na opção Rankear Riscos.	Os Riscos deverão ser ordenados de acordo com o ranking que inicialmente é definido pelo produto (Probabilidade x Impacto). Entretanto o usuário pode alterar esta ordem através de drag and drop dos riscos.
02	Quando usuário estiver na matriz de riscos, e der um double click em algum risco.	O usuário deve ser levado um dialog de atualização do risco clicado.
03	Quando usuário estiver na matriz de riscos, e clicar no left pane na opção “filtrar por criador dos riscos”.	Deve acontecer uma diferenciação de cor entre os riscos gerados pela a ontologia e os riscos inseridos pelo usuário.
04	Quando usuário estiver na matriz de riscos, e clicar no left pane na opção “filtrar por impacto”.	Deve acontecer uma diferenciação de cor de acordo com nível de impacto dos riscos.
05	Quando usuário estiver na matriz de riscos, e clicar no left pane na opção “inserir risco”.	Mostrar um pequeno Wizard com os alguns passos para a inserção de um risco.
06	Quando usuário estiver na matriz de riscos, e houver um risco selecionado e ele clicar no botão atualizar risco da mPrimeToolBar (Este Botão só deve ficar ativado quando houver um risco selecionado).	O usuário deve ser levado um dialog de atualização do risco selecionado.
07	Quando usuário estiver na matriz de riscos, e houver um risco selecionado e ele clicar no menu atualizar risco do mPrime menu (Este menu só deve ficar ativado quando houver um risco selecionado).	O usuário deve ser levado um dialog de atualização do risco selecionado.
08	Quando usuário estiver na matriz de riscos, e houver um risco	O usuário deve ser levado um dialog de atualização do risco selecionado.

<i>mPrime</i>	Versão: 1.0
Modelo de Classe e de Dados	Data da Versão: 26/06/2005
Suppera_ModeloDeClassesEDados.doc	

	selecionado e ele clicar na opção atualizar risco do left pane	
09	Quando usuario clicar na opcao relatórios do menu mPrime	Deve ser gerado um relatorio com arvore de risco, baseado no modelo de relatorios já gerados pelo project.
10	Quando usuario clicar na opcao relatórios a ToolBar mPrime	Deve ser gerado um relatorio com arvore de risco, baseado no modelo de relatorios já gerados pelo project.

## 5. Referências

<http://www.xispe.com.br/>

- Addison Wesley - Kent Beck, Martin Fowler - Planning Extreme Programming;
- Addison Wesley – Extreme Programming Explained
- Addison-Wesley - Martin Fowler – Refactoring -Improving the Design of Existing Code
- Wesley - Extreme Programming Applied - Playing to Win