**UNIVERSIDADE TECNOLÓGICA FEDERAL DO PARANÁ**

**CAMPUS PONTA GROSSA**

**DAINF – DEPARTAMENTO ACADÊMICO DE INFORMÁTICA**

**DION MAICON DUARTE**

**LUCAS EMANUEL RAMOS FERNANDES**

**PROJETO DE ARQUITETURA DE SOFTWARE**

**PONTA GROSSA**

**2015**

**SUMÁRIO**

[1. Referências 3](#_Toc432701509)

[2. Metas e Restrições da Arquitetura 4](#_Toc432701510)

[3. Visão de Casos de Uso 6](#_Toc432701511)

[4. Visão Lógica 7](#_Toc432701512)

[a. Descrição dos casos de uso na visão lógica 8](#_Toc432701513)

[5. Visão de Dados 9](#_Toc432701514)

[a. Sobre Confidencialidade 9](#_Toc432701515)

[b. Sobre Integridade 10](#_Toc432701516)

[c. Sobre Disponibilidade 10](#_Toc432701517)

[d. Mapeamento dos dados 10](#_Toc432701518)

[6. Tamanho e Desempenho 11](#_Toc432701519)

[7. Qualidade 12](#_Toc432701520)

# Referências

**Documentos internos:**

Os documentos citados abaixo se encontram inseridos neste documento e sua autoria é própria

dos autores deste documento.

Visão de Caso de Uso

Visão Lógica

Visão de dados

**Documentos Externos:**

Especificação dos requisitos de software (APS1 Eng.Soft. 02/2015)

Estudo de caso da SATI (Semana de Atualização em Tecnologia da

Informação), material da disciplina de engenharia de software, elaborado pelo prof. Mathias Talevi Betim. Disponível em: <<http://moodle.pg.utfpr.edu.br/mod/resource/view.php?id=28856>>. Acesso mediante usuário e senha da instituição.

MySQL reference manuals. Disponível em : <<http://dev.mysql.com/doc/>> .

Acesso em 10 out. 2015. disponível em versão eletrônica.

# Metas e Restrições da Arquitetura

**Requisitos funcionais**

**Cadastro**

**[RF001] Cadastrar Edição da SATI**

**Descrição do caso de uso:** Este caso de uso permite que o administrador cadastre uma nova edição da SATI, para que o ano da edição possa ser diferenciado dos demais.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Prioridade**: |  | Essencial |  | Importante |  | Desejável |

**Entradas e pré-condições**: A edição precisa de dados bancários.

**Saídas e pós-condição**: Um evento será cadastrado com datas de inicio e fim definidas, assim como datas limites para pagamento e inscrição nas programações do evento.

**[RF002] Cadastrar Programações**

**Descrição do caso de uso:** Este caso de uso permite que o administrador cadastre programações , para que a agenda do evento seja criada.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Prioridade**: |  | Essencial |  | Importante |  | Desejável |

**Entradas e pré-condições**: A programação deve estar entre as datas de inicio e fim do evento, deve ter um custo , local, uma carga horária e pelo menos um responsável.

**Saídas e pós-condição**: Uma agenda para o evento é definida e nenhuma programação pode ser adicionada ao evento.

**[RF003] Cadastrar Usuário**

**Descrição do caso de uso:** Este caso de uso permite que o participante possa se cadastrar, para que possa interagir com o sistema.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Prioridade**: |  | Essencial |  | Importante |  | Desejável |

**Entradas e pré-condições**: Dados pessoais do participante.

**Saídas e pós-condição**: Um cadastro para o participante.

**[RF004] Inscrever um Participante**

**Descrição do caso de uso:** Este caso de uso permite que o participante possa se inscrever em programações cadastradas no evento.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Prioridade**: |  | Essencial |  | Importante |  | Desejável |

**Entradas e pré-condições**: Cadastro do participante no sistema. Participante não pode ser o responsável pela programação que deseja se inscrever.

**Saídas e pós-condição**: Uma inscrição na programação desejada será realizada e confirmada mediante o pagamento do boleto a ser solicitado pelo participante.

**[RF005] Inscrever um Participante**

**Descrição do caso de uso:** Este caso de uso permite que o participante possa se inscrever em programações cadastradas no evento.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Prioridade**: |  | Essencial |  | Importante |  | Desejável |

**Entradas e pré-condições**: Cadastro do participante no sistema. Participante não pode ser o responsável pela programação que deseja se inscrever.

**Saídas e pós-condição**: Uma inscrição na programação desejada será realizada e confirmada mediante o pagamento do boleto a ser solicitado pelo participante.

**Gerenciamento**

**[RF001] Gerenciar Pessoas**

**Descrição do caso de uso:** Este caso de uso permite que o administrador gerencie os usuários do sistema para que possa controlar as informações do sistema, pesquisar e alterar cadastros de usuários.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Prioridade**: |  | Essencial |  | Importante |  | Desejável |

**Entradas e pré-condições**: dados de usúario cadastrado no sistema.

**Saídas e pós-condição**: o usúario tem dados alterados ou excluidos conforme a necessidade do administrador para facilitar o gerenciamento e a correção de eventuais erros.

**[RF002] Gerenciar Programações**

**Descrição do caso de uso:** Este caso de uso permite que o administrador gerencie as programações, para que possa alterar, pesquisar, apagar ou acrescentar dados.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Prioridade**: |  | Essencial |  | Importante |  | Desejável |

**Entradas e pré-condições**: a programação deve estar cadastrada.

**Saídas e pós-condição**: a mesma programação com dados alterados que não confitem com as restrições na agenda do evento.

**[RF003] Gerenciar Edições**

**Descrição do caso de uso:** Este caso de uso permite que o administrador gerencie as edição do evento, para que possa alterar, pesquisar, apagar ou acrescentar dados.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Prioridade**: |  | Essencial |  | Importante |  | Desejável |

**Entradas e pré-condições**: o evento deve estar cadastrado.

**Saídas e pós-condição**: o mesmo evento alterado conforme a necessidade do administrador.

**[RF004] Gerenciar Participação**

**Descrição do caso de uso:** Este caso de uso permite que o administrador controle a presença dos participantes nas programações, para que possa confirmar o pagamento.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Prioridade**: |  | Essencial |  | Importante |  | Desejável |

**Entradas e pré-condições**: .

**Saídas e pós-condição**: o mesmo evento alterado conforme a necessidade do administrador.

**Relatórios**

**[RF001] Gerar Certificado**

**Descrição do caso de uso:** Este caso de uso permite que o participante solicite ao sistema o certificado de participação na programação, para que comprove sua participação no evento.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Prioridade**: |  | Essencial |  | Importante |  | Desejável |

**Entradas e pré-condições**: o participante deve ter pago o boleto e também frequentado a programação.

**Saídas e pós-condição**: o participante tem a opção de imprimir o certificado ou que o mesmo seja enviado para o e-mail cadastrado.

**[RF002] Gerar Relatório Financeiro**

**Descrição do caso de uso:** Este caso de uso permite ao administrador gerar um relatório em formato PDF, que contém toda a movimentação financeira do evento.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Prioridade**: |  | Essencial |  | Importante |  | Desejável |

**Entradas e pré-condições**: Programações com participantes inscritos e custos de realização de cada programação.

**Saídas e pós-condição**: O relatório deve conter os custos por programação, incrições confirmadas.

**[RF003] Gerar Boleto**

**Descrição do caso de uso:** Este caso de uso permite ao participante gerar um boleto que contém os custos de suas inscrições, para que efetuado o pagamento, possa confirmar sua presença nas programações inscritas.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Prioridade**: |  | Essencial |  | Importante |  | Desejável |

**Entradas e pré-condições**: o participante deve ter inscrições válidas no evento.

**Saídas e pós-condição**: O boleto será emitido de acordo com as opções do participante,opções essas que permitem descontos no pagamento.

**Financeiro**

**[RF001] Confirmar Pagamento**

**Descrição do caso de uso:** Este caso de uso permite ao administrador confirmar o pagamento de incrições de um participante, para que o mesmo possa frequentar a(s) programação(ões) que se inscreveu.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Prioridade**: |  | Essencial |  | Importante |  | Desejável |

**Entradas e pré-condições**: Uma tabela(lista) com inscrições pedentes a serem confirmadas pelo administrador.

**Saídas e pós-condição**: a alteração do status de confirmação de pagamento do particiante.

**Autentificação**

**[RF001] Logar Sistema**

**Descrição do caso de uso:** Este caso de uso permite que ao administrador logar no sistema, para que possa gerenciar os eventos e programações que é responsável.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Prioridade**: |  | Essencial |  | Importante |  | Desejável |

**Entradas e pré-condições**: Login e senha cadastrados.

**Saídas e pós-condição**: Uma tela que permite a interação com o sistema a nível de administrador.

**[RF002] Cadastrar Administrador**

**Descrição do caso de uso:** Este caso de uso permite que ao administrador cadastrar outros administradores, para que a tarefa de manter o evento e as programações possam ser divididas e passadas a seus responsáveis.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Prioridade**: |  | Essencial |  | Importante |  | Desejável |

**Entradas e pré-condições**: Login e senha cadastrados. Dados pessoais do novo administrador do sistema.

**Saídas e pós-condição**: Uma nova conta com privilégios administrativos.

**Requisitos não-funcionais**

**[NF001]****Usabilidade**

A interface com o usuário é de vital importância para o sucesso do sistema. Principalmente por ser um sistema que não será utilizado diariamente, o usuário não possui tempo disponível para aprender como utilizar o sistema.

O sistema deve apresentar interface amigável e intuitiva prevalecendo o bom uso das ferramentas de programação para dar o usuário o mínimo de interação necessárias para realizar seu objetivo.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Prioridade**: |  | Essencial |  | Importante |  | Desejável |

**[NF002] Desempenho**

Embora não seja um requisito essencial ao sistema, deve ser considerada por corresponder a um fator de qualidade de software.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Prioridade**: |  | Essencial |  | Importante |  | Desejável |

**[NF003] Hardware e Software**

Visando criar um produto com maior extensibilidade, reusabilidade e flexibilidade, deve ser adotar como linguagem principal de desenvolvimento Java seguindo cuidadosamente as técnicas de orientação a objetos.

O uso da linguagem Java permite não especificar qual será o sistema operacional e a máquina em que o programa irá executar. No entanto, essa máquina deverá se comunicar com um sistema de banco de dados.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Prioridade**: |  | Essencial |  | Importante |  | Desejável |

# Visão de Casos de Uso

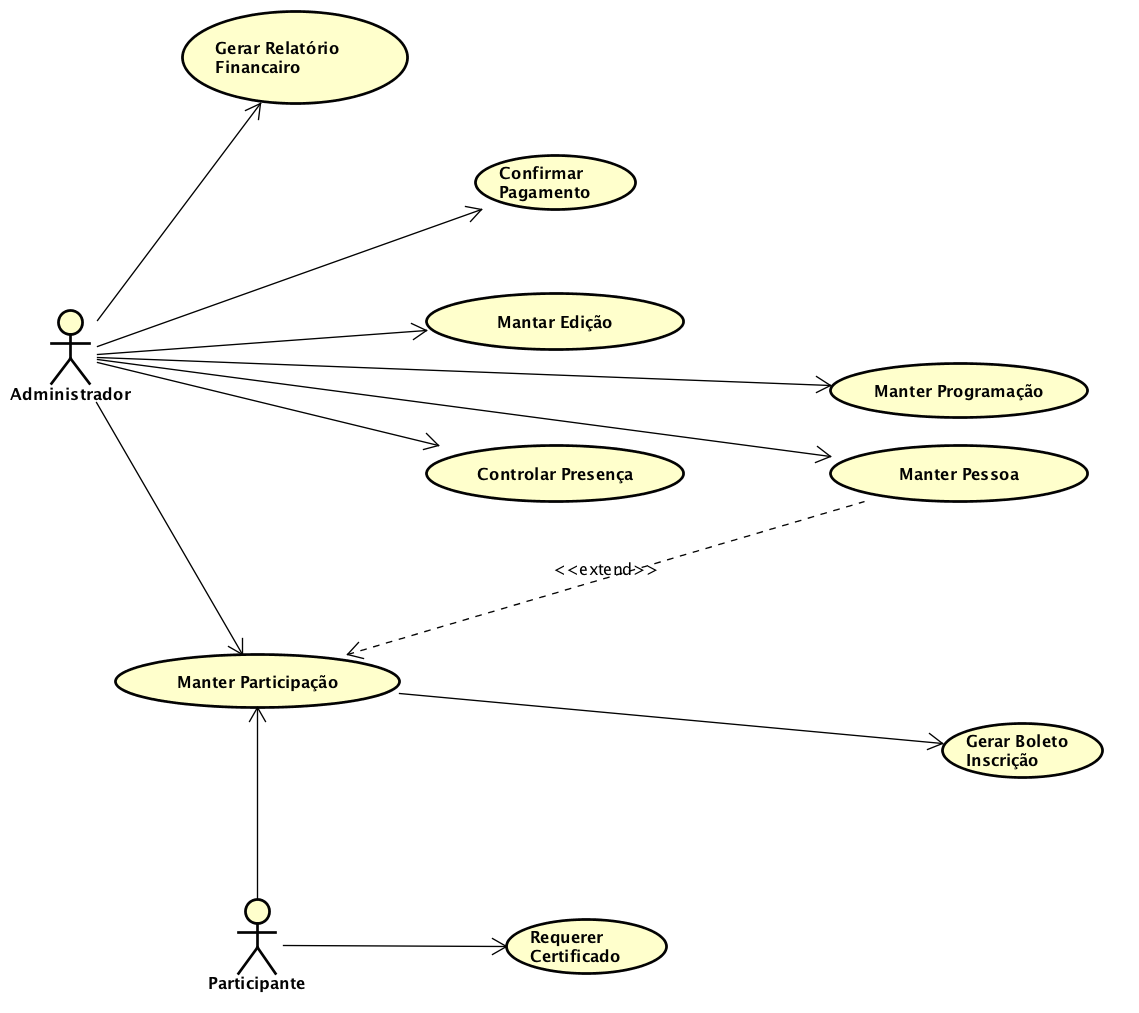
Para o sistema de administração da Semana de Atualização em Tecnologia da Informação foi necessário somente um diagrama de caso de uso em sua representação. Isso se deve ao fato de que o sistema não é complexo e não possui uma grande quantidade de módulos a serem desenvolvidos.

Figura 1. Diagrama de Caso de Uso

# Descrição dos casos de uso na visão de casos de uso

Dentro do Diagrama de Caso de usos acima, três casos de uso se destacam por serem de fundamental importância no sistema, são estes:

1. Requerer Certificado

O participante da edição da SATI pode requisitar um certificado com a carga horária da programação que participou.

1. Fluxo de Eventos

* Participante entra no sistema com suas informações.
* Participante procurar a edição do evento e as programações que estava inscrito.
* Se o participante presenciou a programação, poderá requisitar o certificado.
* Após todos os dados serem informados, o participante pode gerar o certificado.

1. Classes envolvidas no processo: Pessoa, Programação, Edição e Participação.
2. Requisitos Derivados

* Para participar de uma programação, é necessário o pagamento da mesma.
* É possível a geração de um PDF como certificado.
* O certificado pode ser enviado por e-mail.
* Certificado deve ter as informações da programação e do participante. Carga horária é essencial.
* Para gerar o certificado, é necessário conferir se o participante estava presente na programação.

1. Gerar Boleto

O participante da edição da SATI gerar um boleto com todas as inscrições de programações que não foram pagas.

1. Fluxo de Eventos

* Participante entra no sistema com suas informações.
* O sistema procura pelas informações da edição atual.
* Participante seleciona a opção de pagamento.
* O sistema retorna todas as programações inscritas e não pagas.
* Sistema gera um boleto com os valores necessários para o pagamento.

1. Classes envolvidas no processo: Pessoa, Programação, Edição e Participação e Boleto.
2. Requisitos Derivados

* Boleto contém as informações para o pagamento, bem como os valores de desconto.
* É possível a geração de um PDF como boleto.

1. Gerar Relatório Financeiro

A administração da edição da SATI pode gerar um relatório contendo todas as informações de caixa durante aquele evento.

1. Fluxo de Eventos

* O administrador entra no sistema.
* O administrador seleciona qual edição deseja gerar as informações.
* O sistema calculo todas as entradas e saídas daquela edição.

1. Classes envolvidas no processo: Programação, Edição e Participação e Boleto.
2. Requisitos Derivados

* Não se aplica.

# Visão Lógica

Para estruturar o sistema, serão necessárias as seguintes classes: Edição, Dados Bancários, Pessoa, Programação, Participação e Boleto. Por ser um sistema pequeno, todas as classes são de vital importante e desempenham uma função importante. Dentro do sistema, os casos de uso mais relevantes são: Requerer Certificado, Gerar Boleto e Gerar Relatório Financeiro.



Figura 2. Diagrama de Classes

Dentro do sistema, há no total 6 classes que possuem atividades especificas e auxiliam no funcionamento correto do software. Estas são:

* **Dados Bancários**: Esta classe é responsável por controlar os dados bancários de uma edição. Auxilia na verificação de dados e correção de valores dos atributos.
* **Edição**: Controla as instâncias de eventos da SATI. Possui um método para verificação de saldo, que é responsável por gerar um relatório financeiro completo ao final da uma edição. Este relatório contém informações sobre as entradas e saídas e o saldo daquela edição. A classe também conta com um método para gerar agenda, que fornece um calendário com as informações dos eventos pertencentes a uma edição. Cada edição pode ter uma ou mais programações e a informação de dados bancários.
* **Programação**: A classe programação possui a função de controlar os eventos dentro de uma edição da SATI. Tais eventos podem ser palestras, minicursos, workshops, maratonas de programações, entre outros. Cada programação requer uma pessoa responsável por ministra-lo, e pode ter nenhum ou muitos participantes. Pode possuir custo para equipamentos, softwares, entre outros recursos. A classe programação, permite a criação dos objetos programação, esses objetos identificam as programações da edição da SATI através de seus atributos, uma única função está definida e utiliza os atributos do objeto para efetuar calculo de custo da programação.
* **Boleto**: A classe boleto é responsável por controlar um grupo de participações em diferentes programações por uma mesma pessoa. É possível gerar um arquivo PDF para poder realizar o pagamento do boleto, através do método gerar boleto1. Quando o boleto é pago, ou seja, método confirmar pagamento é executado, as participações associadas a ele serão atualizadas como válidas, e a pessoa é autorizada a participar da programação.
* **Pessoa**: A classe pessoa, permite a criação de objetos do tipo pessoa, esses objetos guardam dados pessoais e uma referência ao tipo de pessoa (o tipo especificado no modelo, refere-se a especialização de pessoa, que pode ser administrador ou participante) sera usado pelo sistema como referência em praticamente todas as operações, pois para ter uma conta no sistema, é necessário ter um objeto do tipo pessoa que será usado para validar campos e está inserido na regra de negocio da aplicação.
* **Participação**: A classe participação, permite a criação de objetos do tipo participação, seus atributos são necessários para controle de presença e validação da participação. Os objetos da classe podem executar duas funções, os objeto que têm seus atributos validados, estão habilitados a solicitarem a função de gerar certificado. Todo objeto do tipo participação está estritamente relacionado a pelo menos uma programação e consequentemente a edição da SATI. A participação possui um boleto que pode ser usado para validar sua participação. O atributo presença é necessário para gerar o certificado de participação.

# Visão de Dados

O sistema gerenciador de banco de dados (SGBD) escolhido é o MySQL. Para realizar esta decisão foram considerados os seguintes atributos: Confidencialidade, Integridade e Disponibilidade, bem como a integração entra este software e a IDE utilizado no projeto. Todo o controle sobre as informações do sistema, como por exemplo as regras de negócios, será realizado dentro da codificação do projeto.

# Sobre Confidencialidade

O MySQL oferece métodos de segurança de acesso para que somente as pessoas autorizadas consigam acessar. Além de que dentro do projeto em si é possível determinar quem está autorizado a acessar determinadas tabelas e dados armazenados.

# Sobre Integridade

É possível a programação de backups regulares, e controle do log de funções desempenhas no banco de dados. Há também o controle de quando efetuar o commit de manipulação de dados, no caso de dependência de dados entre tabelas.

# Sobre Disponibilidade

O software oferece um grande potencial de desempenho. As falhas do software são poucas e então há um risco mínimo de que não seja possível manipular dados por conta do SGBD.

# Mapeamento dos dados

O banco de dados será utilizado somente para o armazenamento. Nenhuma procedure ou trigger serão criados para a manutenção dos dados.



Figura 3. Modelo Lógico

# Tamanho e Desempenho

As operações realizadas no sistema refletem diretamente no banco de dados do sistema, o banco de dados deve susportar no mínimo 500 conexões simultâneas, dentro dos padrões de SGBD's é um número pequeno e como a quantidade de dados envolvidas nas operações de DML(Data Manipulation Language) e o grau das relações é pequena, não esperasse atraso, cabe apenas cuidados na modelagem que evitem as anomálias de atualização que podem ocorrer.

A portabilidade que a linguagem de programação java proporciona, permite que os usuários executem a aplicação de diversos sistemas operacionais, usuários de smartphones dependem da instalação do java em seus dispositivos. Até a data de elaboração deste documento não há previsão da aplicação receber atualizações. O suporte ao usuário primário deve ser dado primeiramente pelos administradores do sistema que tem as ferramentas necessárias para gerenciar o sistema de forma geral. Na geração de relatórios e boletos esperasse um pequeno atraso, embora pequeno, pode ser ser perceptível ao usuário.

# Qualidade

O sistema a ser desenvolvido na linguagem de programação java, permite uma portabilidade maior que não depende do sistema operacional. Do sistema operacional esperasse que o mesmo tenha uma conexão de rede e recursos mínimos necessários para executar uma java virtual machine(JVM). Outra necessidade essencial para o funcionamento da aplicação é um servidor de banco de dados disponível na rede.

O hardware necessário é o mínimo para executar as tarefas acima. Os executáveis do modo administrador serão distribuidos aos administradores, o administrador poderá se conectar com o banco de dados via usuário e senha, previamente cadastrados. O módulo de admistração tem menos de 20 mb e executa sobre a JVM.

O SGBD pode executar na rede apartir de qualquer máquina com recursos mínimos para ter confiabilidade e segurança em suas operações, em endereço(IP) acessível para os usuários(administradores e participantes).

O módulo participante é similar ao de administração com menos privilégios e executável menor.

Praticamente todas as operações realizadas pelos usuários utilizam o banco de dados da aplicação. Logo esperasse um bom retorno e um controle de segurança apartir dos usuários com privilégios administrativos, para a segurança a aplicação conta com autentificação para uso dos módulos de administrador e participante.

Os relatórios e boletos serão emitidos em formato PDF. O visualizador de PDF deve estar instalado no computador que solicita o relatório/boleto.