

UNIVERSIDADE TECNOLÓGICA FEDERAL DO PARANÁ
CAMPUS PONTA GROSSA
DAINF – DEPARTAMENTO ACADÊMICO DE INFORMÁTICA

THIAGO TRACZYKOWSKI
GIOVANE AIRES

PROJETO DE ARQUITETURA DE SOFTWARE
SATI

PONTA GROSSA
2015

SUMÁRIO

<u>REFERÊNCIAS</u>	3
<u>Documentos Externos</u>	3
<u>Documentos Internos</u>	3
<u>METAS E RESTRIÇÕES DA ARQUITETURA</u>	4
<u>VISÃO DE CASOS DE USO</u>	5
<u>VISÃO LÓGICA</u>	12
<u>Visão Geral</u>	12
<u>Realizações de Casos de Uso</u>	13
<u>VISÃO DE DADOS</u>	18
<u>TAMANHO E DESEMPENHO</u>	20
<u>QUALIDADE</u>	21

1. REFERÊNCIAS

São referência nesse projeto os seguintes documentos:

1. Documentos Externos

Modelo Projeto de Arquitetura de Software RUP cedido pelo Professor Mathias Talevi Betim na disciplina de Engenharia de Software da Universidade Tecnológica Federal do Paraná.

2. Documentos Internos

Utilizado como referência o backlog por meio do sítio easyBacklog, o qual era a intenção construir uma backlog nas especificações Scrum de um Sistema de Gerenciamento para a SATI.

2. METAS E RESTRIÇÕES DA ARQUITETURA

De acordo com a arquitetura escolhida, o sistema deverá ter como características:

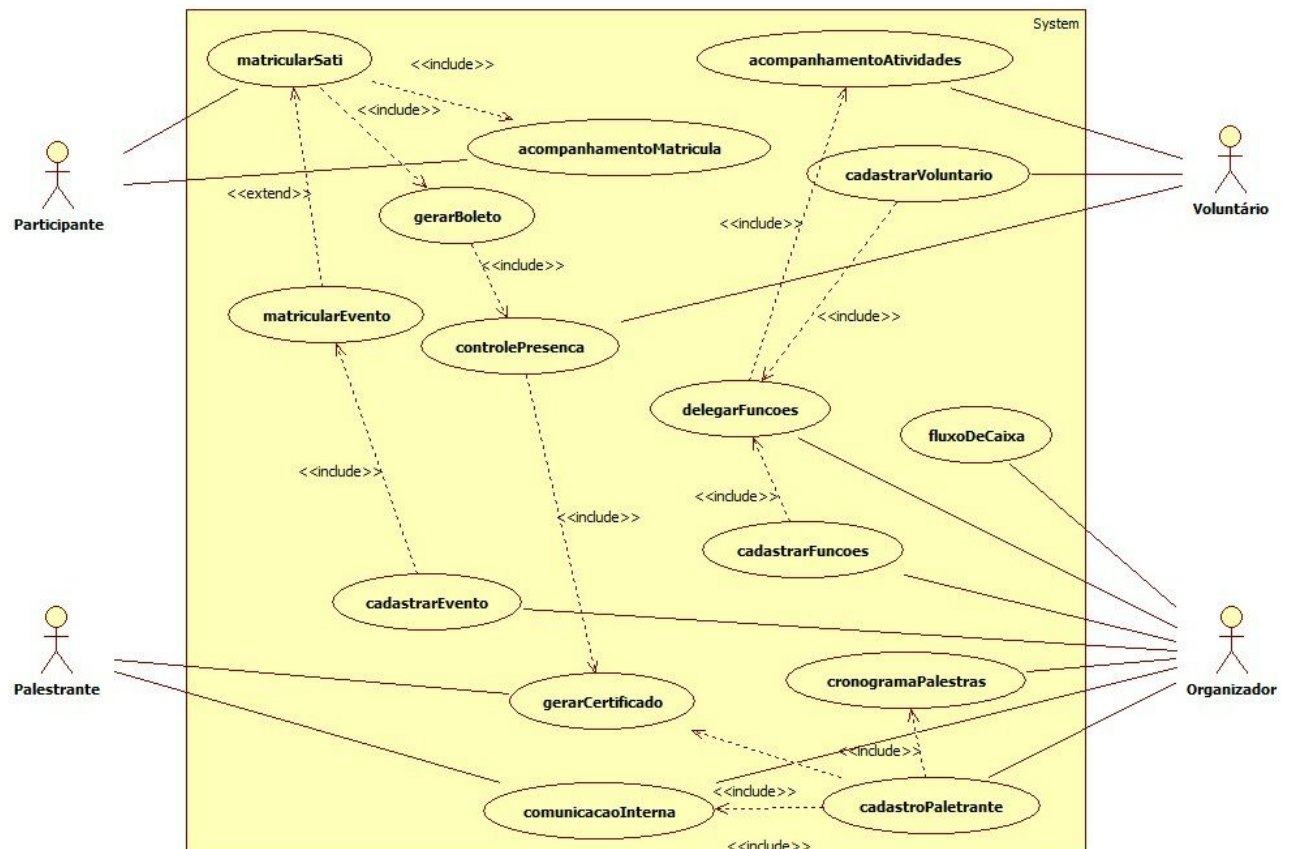
- Modularidade: o sistema deve ser desenvolvido em camadas, sendo uma delas a interface do usuário.
- Extensibilidade: Serão utilizadas ferramentas open-source e de livre acesso.
- Reusabilidade: o sistema deverá ser desenvolvido em módulos, tornando-o reutilizável em outros projetos.
- Portabilidade: O sistema deverá ser compatível como os mais utilizados navegadores comerciais sem utilizar recursos específicos de algum dos navegadores.

A arquitetura do software será baseada nos requisitos funcionais do sistema, crescendo e iterando sobre essas funcionalidades até se tornar um sistema robusto.

Cada módulo do sistema deverá ser independente do outro, podendo ser atualizado, modificado e ou substituído quando necessário

As interfaces do sistema devem ter como foco a fácil utilização pelo usuário, tornando a experiência de usar o sistema agradável..

3. VISÃO DE CASOS DE USO



Nome:	matricularSati
Descrição:	Ao clicar no botão de matricula na Sati, o usuário é levado à tela de cadastro, onde ele insere suas informações e confirma, sendo redirecionado para a impressão do boleto e posteriormente à seu perfil no sistema.
Fluxo de Eventos:	<p>Usuário clica no botão de cadastro de matrícula.</p> <p>Sistema redireciona para tela de cadastro.</p> <p>Usuário insere e confirma suas informações.</p> <p>Sistema redireciona para tela de impressão de boleto.</p> <p>Usuário opta por imprimir ou salvar o boleto.</p> <p>Sistema redireciona ao perfil do usuário recém-criado.</p>
Relacionamento:	matricularSati includes acompanhamentoMatricula, gerarBoleto.

Nome:	matricularEvento
Descrição:	Ao clicar no botão de matricula de eventos, o usuário é direcionado a uma tela onde estão listados os eventos em que ele ainda não foi matriculado, ele deve selecionar por meio de uma checkbox o evento desejado e clicar em confirmar onde será redirecionado à tela do boleto.
Fluxo de Eventos:	<p>Usuário clica no botão de matricula de eventos.</p> <p>Sistema redireciona para tela de matricula de eventos.</p> <p>Usuário escolhe qual evento se matricular por meio de uma checkbox e confirma suas informações.</p> <p>Sistema redireciona para tela de impressão de boleto.</p> <p>Usuário opta por imprimir ou salvar o boleto.</p>
Relacionamento:	matricularEvento extend matricularSati.

Nome:	gerarBoleto
Descrição:	Ao clicar no botão de Gerar Boleto, o usuário é direcionado à uma tela onde são listados os boletos já gerados, dando-lhe opção para tirar segunda via.
Fluxo de Eventos:	<p>Usuário clica no botão Gerar Boleto.</p> <p>Sistema redireciona à tela onde é listado os boletos já gerados.</p> <p>Usuário seleciona um boleto por meio de uma checkbox e clica no botão de confirmar.</p> <p>Sistema gera segunda-via do boleto selecionado.</p> <p>Usuário opta por imprimir ou salvar o boleto.</p>
Relacionamento:	gerarBoleto include controlePresenca

Nome:	acompanhamentoMatricula
Descrição:	Ao clicar no botão de Acompanhamento de matricula o usuário é levado a uma tela onde são listados todos os eventos em qual está matriculado, incluindo a própria Sati e é apresentado o status da matricula, se está aprovada, pendente, paga, ou aguardando pagamento.
Fluxo de Eventos:	<p>Usuário clica no botão Acompanhamento de Matricula.</p> <p>Sistema redireciona usuário para tela de Acompanhamento de Matricula.</p>
Relacionamento:	

Nome:	controlePresenca
Descrição:	Voluntário clica no botão de Chamada, é redirecionado à uma tela listando todos os eventos do qual ele está responsável por realizar a chamada, ao selecionar por meio de uma, ao clicar no botão chamada do evento, o voluntário é redirecionado para uma listagem de todos os alunos matriculados naquele evento, podendo alterar o status de presente e ausente dos mesmos.
Fluxo de Eventos:	Voluntário clica no botão de Chamada. Sistema redireciona usuário para uma tela com a listagem de eventos do qual ele é responsável. Voluntário clica no botão de chamada do evento escolhido. Sistema redireciona para uma tela com a listagem dos alunos matriculados no evento. Voluntário altera status de presença dos alunos e clica em confirmar. Sistema apresenta uma mensagem de sucesso ou erro.
Relacionamento:	controlePresenta include gerarCertificado.

Nome:	cadastrarVoluntario
Descrição:	Usuário clica no botão “Voluntarize-se” e confirma o cadastro.
Fluxo de Eventos:	Usuário clica no botão “Voluntarize-se”. Sistema abre um popup pedindo confirmação. Usuário confirma. Sistema apresenta mensagem de sucesso.
Relacionamento:	cadastrarVoluntario includes delegarFuncoes

Nome:	acompanhamentoAtividades
Descrição:	Voluntário clica no botão de acompanhamento de atividades para visualizar uma listagem de funções delegadas a ele e seu status, se já foi finalizada ou ainda está para acontecer.
Fluxo de Eventos:	Voluntário clica no botão de Acompanhamento de Atividades Sistema redireciona para tela de Acompanhamento de Atividades
Relacionamento:	

Nome:	delegarFuncoes
Descrição:	Organizador clica no botão Delegar Funções e é direcionado para uma tela onde há dois combo boxes onde o organizador seleciona o voluntário e a função a ser delegada e clica em confirmar.
Fluxo de Eventos:	Organizador clica no botão Delegar Funções Sistema direciona para a tela de delegação de funções Organizador seleciona por meio das combo boxes o voluntário e a função desejada e clica em confirmar Sistema apresenta uma mensagem de sucesso.
Relacionamento:	delegarFuncoes include acompanhamentoAtividades

Nome:	cadastrarFuncoes
Descrição:	Organizador clica no botão Cadastrar Funções e é direcionado para uma tela onde deve inserir os dados da função.
Fluxo de Eventos:	Organizador clica no botão Cadastrar Funções Sistema redireciona para uma tela de cadastramento de funções. Organizador insere os dados da função e clica em confirmar. Sistema apresenta mensagem de sucesso.
Relacionamento:	

Nome:	fluxoDeCaixa
Descrição:	Organizador clica no botão de Fluxo de caixa e é direcionado para uma tela onde são listadas as entradas e saídas de dinheiro e onde pode cadastrar nova entrada ou saída de dinheiro.
Fluxo de Eventos:	Organizador clica no botão de Fluxo de Caixa Sistema redireciona para uma tela que apresenta as entradas e saídas de dinheiro em forma de tabela. Organizador insere os dados de nova entrada ou saída e clica em confirmar. Sistema atualiza tabela com novo registro.
Relacionamento:	

Nome:	cadastroPalestrante
Descrição:	Organizador clica no botão Cadastrar Palestrante e é direcionado para uma tela onde deve inserir os dados do palestrante.
Fluxo de Eventos:	Organizador clica no botão Cadastrar Palestrante Sistema redireciona para uma tela de cadastramento de palestrante. Organizador insere os dados do palestrante e clica em confirmar. Sistema apresenta mensagem de exito.
Relacionamento:	cadastroPalestrante includes cronogramaPalestras, gerar certificado, comunicacaoInterna

Nome:	cadastrarEvento
Descrição:	Organizador clica no botão de Cadastro de Eventos e é direcionado à uma página onde deve inserir os dados do evento e confirmar.
Fluxo de Eventos:	Organizador clica no botão Cadastrar Evento Sistema redireciona para uma tela de cadastramento de eventos. Organizador insere os dados do eventoe clica em confirmar. Sistema apresenta mensagem de exito.
Relacionamento:	cadastrarEvento includes matricularEvento

Nome:	comunicaolInterna
Descrição:	Palestrante/Organizador clicam no botão de Comunicação Interna e são direcionados à uma tela nos moldes de e-mail, selecionado o usuário destinatário e escreve a mensagem clicando enviar.
Fluxo de Eventos:	Palestrante/Organizador clicam no botão de Comunicação Interna. Sistema direciona para tela de comunicação interna. Palestrante/Organizador escolhem destinatário por meio de combo box e escrevem sua mensagem, posteriormente clicando em enviar. Sistema envia uma mensagem para o destinatário.
Relacionamento:	

Nome:	cronogramaPalestras
Descrição:	Organizador clica no botão cronograma de palestras, onde é direcionado para uma tela no estilo calendário contendo todas as palestras cadastradas, ao clicar num dia, aparece um popup solicitando a palestra a ser definida para aquele dia.
Fluxo de Eventos:	Organizador clica no botão Cronograma de Palestras. Sistema redireciona para tela de cronograma de palestras. Organizador clica em um dia. Sistema abre um popup solicitando qual palestra será cadastrada para aquele dia e seu horário. Organizador seleciona palestra e hora por meio de combo boxes e clica em Ok. Sistema atualiza calendário.
Relacionamento:	

Nome:	gerarCertificado
Descrição:	Palestrante/Participante/Voluntário clicam no botão Gerar Certificados, são direcionados a uma tela listando todas as atividades quais foram participadas onde podem clicar no botão de Gerar Certificado onde é direcionado para o Certificado ser impresso.
Fluxo de Eventos:	Palestrante/Participante/Voluntário clicam no botão Gerar Certificados. Sistema redireciona para uma tela listando os eventos em que foram participados. Palestrante/Participante/Voluntário clicam no botão Gerar Certificado associado ao evento desejado. Sistema direciona para o Certificado ser impresso.
Relacionamento:	

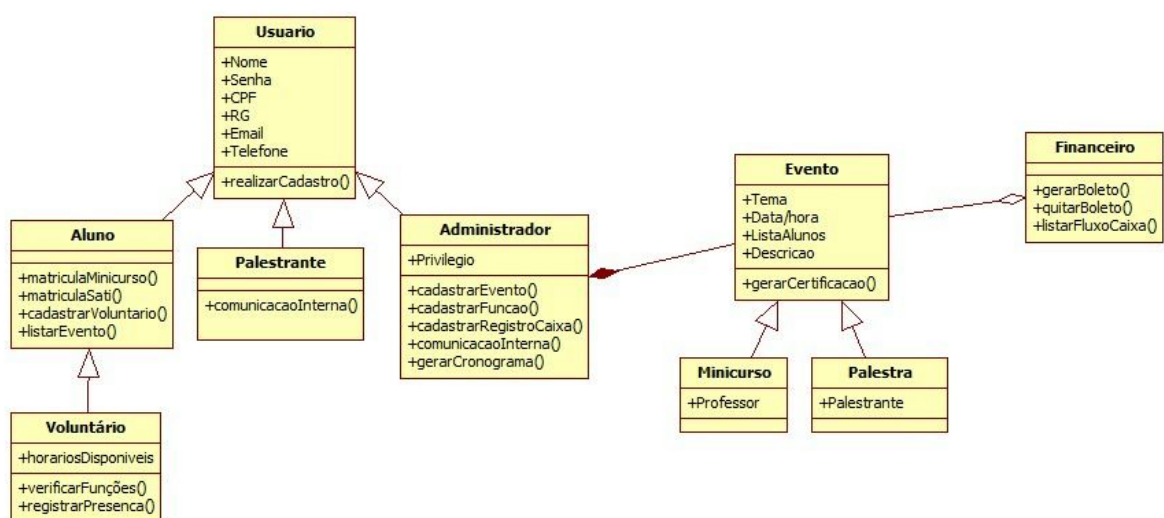
4. VISÃO LÓGICA

1. Visão Geral

Utilizando a arquitetura de design baseada em MVC, iremos dividir todas as classes do Diagrama Geral de Classe em pacotes de controle, modelo e visão. Onde

- Um **controlador (controller)** envia comandos para o modelo para atualizar o seu estado. O controlador também pode enviar comandos para a visão associada para alterar a apresentação da visão do modelo
- Um **modelo (model)** armazena dados e notifica suas visões e controladores associados quando há uma mudança em seu estado. Estas notificações permitem que as visões produzam saídas atualizadas e que os controladores alterem o conjunto de comandos disponíveis. Uma implementação *passiva* do MVC monta estas notificações, devido a aplicação não necessitar delas ou a plataforma de software não suportá-las.
- A **visão (view)** Gera uma representação (Visão) dos dados presentes no modelo solicita do modelo.

Apresenta-se de modo geral as classes de acordo com o diagrama abaixo:



2. Realizações de Casos de Uso

Nome:	matricularSati
Descrição:	Ao clicar no botão de matricula na Sati, o usuário é levado à tela de cadastro, onde ele insere suas informações e confirma, sendo redirecionado para a impressão do boleto e posteriormente à seu perfil no sistema.
Fluxo de Eventos:	Usuário clica no botão de cadastro de matrícula. Sistema redireciona para tela de cadastro. Usuário insere e confirma suas informações. Sistema redireciona para tela de impressão de boleto. Usuário opta por imprimir ou salvar o boleto. Sistema redireciona ao perfil do usuário recém-criado.
Relacionamento:	matricularSati includes acompanhamentoMatricula, gerarBoleto.

Nome:	matricularEvento
Descrição:	Ao clicar no botão de matricula de eventos, o usuário é direcionado a uma tela onde estão listados os eventos em que ele ainda não foi matriculado, ele deve selecionar por meio de uma checkbox o evento desejado e clicar em confirmar onde será redirecionado à tela do boleto.
Fluxo de Eventos:	Usuário clica no botão de matricula de eventos. Sistema redireciona para tela de matricula de eventos. Usuário escolhe qual evento se matricular por meio de uma checkbox e confirma suas informações. Sistema redireciona para tela de impressão de boleto. Usuário opta por imprimir ou salvar o boleto.
Relacionamento:	matricularEvento extend matricularSati.

Nome:	gerarBoleto
Descrição:	Ao clicar no botão de Gerar Boleto, o usuário é direcionado à uma tela onde são listados os boletos já gerados, dando-lhe opção para tirar segunda via.
Fluxo de Eventos:	<p>Usuário clica no botão Gerar Boleto.</p> <p>Sistema redireciona à tela onde é listado os boletos já gerados.</p> <p>Usuário seleciona um boleto por meio de uma checkbox e clica no botão de confirmar.</p> <p>Sistema gera segunda-via do boleto selecionado.</p> <p>Usuário opta por imprimir ou salvar o boleto.</p>
Relacionamento:	gerarBoleto include controlePresenca

Nome:	controlePresenca
Descrição:	Voluntário clica no botão de Chamada, é redirecionado à uma tela listando todos os eventos do qual ele está responsável por realizar a chamada, ao selecionar por meio de uma, ao clicar no botão chamada do evento, o voluntário é redirecionado para uma listagem de todos os alunos matriculados naquele evento, podendo alterar o status de presente e ausente dos mesmos.
Fluxo de Eventos:	<p>Voluntário clica no botão de Chamada.</p> <p>Sistema redireciona usuário para uma tela com a listagem de eventos do qual ele é responsável.</p> <p>Voluntário clica no botão de chamada do evento escolhido.</p> <p>Sistema redireciona para uma tela com a listagem dos alunos matriculados no evento.</p> <p>Voluntário altera status de presença dos alunos e clica em confirmar.</p> <p>Sistema apresenta uma mensagem de sucesso ou erro.</p>
Relacionamento:	controlePresenta include gerarCertificado.

Nome:	fluxoDeCaixa
Descrição:	Organizador clica no botão de Fluxo de caixa e é direcionado para uma tela onde são listadas as entradas e saídas de dinheiro e onde pode cadastrar nova entrada ou saída de dinheiro.
Fluxo de Eventos:	<p>Organizado clica no botão de Fluxo de Caixa</p> <p>Sistema redireciona para uma tela que apresenta as entradas e saídas de dinheiro em forma de tabela.</p> <p>Organizador insere os dados de nova entrada ou saída e clica em confirmar.</p> <p>Sistema atualiza tabela com novo registro.</p>
Relacionamento:	

Nome:	cadastroPalestrante
Descrição:	Organizador clica no botão Cadastrar Palestrante e é direcionado para uma tela onde deve inserir os dados do palestrante.
Fluxo de Eventos:	<p>Organizador clica no botão Cadastrar Palestrante</p> <p>Sistema redireciona para uma tela de cadastramento de palestrante.</p> <p>Organizador insere os dados do palestrante e clica em confirmar.</p> <p>Sistema apresenta mensagem de sucesso.</p>
Relacionamento:	cadastroPalestrante includes cronogramaPalestras, gerar certificado, comunicacaoInterna

Nome:	cadastrarEvento
Descrição:	Organizador clica no botão de Cadastro de Eventos e é direcionado à uma página onde deve inserir os dados do evento e confirmar.
Fluxo de Eventos:	Organizador clica no botão Cadastrar Evento Sistema redireciona para uma tela de cadastramento de eventos. Organizador insere os dados do evento e clica em confirmar. Sistema apresenta mensagem de sucesso.
Relacionamento:	cadastrarEvento includes matricularEvento

Nome:	fluxoDeCaixa
Descrição:	Organizador clica no botão de Fluxo de caixa e é direcionado para uma tela onde são listadas as entradas e saídas de dinheiro e onde pode cadastrar nova entrada ou saída de dinheiro.
Fluxo de Eventos:	Organizador clica no botão de Fluxo de Caixa Sistema redireciona para uma tela que apresenta as entradas e saídas de dinheiro em forma de tabela. Organizador insere os dados de nova entrada ou saída e clica em confirmar. Sistema atualiza tabela com novo registro.
Relacionamento:	

Nome:	cadastroPalestrante
Descrição:	Organizador clica no botão Cadastrar Palestrante e é direcionado para uma tela onde deve inserir os dados do palestrante.
Fluxo de Eventos:	Organizador clica no botão Cadastrar Palestrante Sistema redireciona para uma tela de cadastramento de palestrante. Organizador insere os dados do palestrante e clica em confirmar.

	Sistema apresenta mensagem de exito.
Relacionamento:	cadastroPalestrante includes cronogramaPalestras, gerar certificado, comunicacaoInterna

Nome:	cadastrarEvento
Descrição:	Organizador clica no botão de Cadastro de Eventos e é direcionado à uma página onde deve inserir os dados do evento e confirmar.
Fluxo de Eventos:	Organizador clica no botão Cadastrar Evento Sistema redireciona para uma tela de cadastramento de eventos. Organizador insere os dados do eventoe clica em confirmar. Sistema apresenta mensagem de exito.
Relacionamento:	cadastrarEvento includes matricularEvento

Elementos de Design Significativos do Ponto de Vista da Arquitetura

São elementos significativos do sistemas os dispositivos de conexão com o banco de dados e sua classe e a classe intitulada de “Pessoa” a qual é um modelo abstrato utilizado por diversas outras classes.

5. VISÃO DE DADOS

O sistema gerenciador de banco de dados utilizado será o PostgreSQL, pois se destaca por ser mais robusto e possuir mais recursos que outros SGBDs, recursos como:

- Transações: confirmação e cancelamento de operações realizadas COMMIT e ROLLBACK;
- Triggers: gatilhos;
- Stored Procedures: Procedimentos Armazenados;
- Views: visões;
- Lock Line: bloqueio em nível de linha;
- Constraints: cláusulas de integridade.

O PostgreSQL possui um sofisticado mecanismo de bloqueio MVCC, suporta tamanhos ilimitados de linhas, bancos de dados e tabelas de até 16TB, aceita vários tipos de sub-consultas, possui mais tipos de dados e conta com um bom mecanismo de FAILSAVE, que é a segurança contra falhas, por exemplo no desligamento repentino do sistema.

Técnica MVCC do PostgreSQL:

Multi-Version Concurrency Control (MVCC) é uma técnica avançada do PostgreSQL que resolve o problema de ambientes multiusuários de banco de dados, os bloqueios que fazem o usuário ficar esperando. Ao contrário de outros SGBDs que utilizam LOCKs para controle de concorrência, o PostgreSQL mantém a consistência dos dados usando um modelo multiversão. Neste modelo, cada transação terá sua versão do banco de dados, estando protegidas de acessar dados inconsistentes que poderiam ser gerados por outras transações. Portanto, o MVCC oferece o isolamento de transações, além de garantir que leituras nunca aguardarão escritas e vice-versa.

6. TAMANHO E DESEMPENHO

- O sistema deve ser capaz de processar as informações de todos os alunos matriculados na Sati simultaneamente.
- Serão métricas para o desempenho do sistema a quantidade de tempo para exibição das informações solicitadas e a quantidade de tempo necessária para realizar uma operação de inserção e atualização.
- O sistema como um todo será utilizado por meio da web, logo seu tamanho não deve passar de 15 megabytes para o cliente. Não tendo limites para o tamanho do banco de dados ao lado do servidor.

7. QUALIDADE

- Falhas devem ocorrer somente ao lado do cliente, sendo tolerada 2 falhas por semana.
- O foco deverá ser a experiência do usuário, sem tempo ocioso ou passos em excesso para acessar as funcionalidades do sistema.
- O sistema terá que ser atualizado por módulos, não impedindo a utilização de parte do sistema caso haja alguma atualização.
- O sistema deverá ser compatível com os mais utilizados navegadores comerciais sem utilizar recursos específicos de algum dos navegadores.