



Universidade do Estado de Santa

Catarina Centro de Ciência Tecnológicas - CCT

EDIANA DA SILVA DE SOUZA

EDUARDO DE PAULA PAZINI

ÊRICA PETERS DO CARMO

DOCUMENTAÇÃO DO PROJETO:

SISTEMA DE RESERVA DE SALAS PARA UDESC – CAMPUS CCT

Joinville (SC)

2018

1. INTRODUÇÃO

O presente documento tem como finalidade apresentar as funcionalidades, requisitos e diagramas do sistema de reservas de salas para a universidade do estado de Santa Catarina – campus CCT, cujo gerencia a locação das salas do campus. A partir da explanação acerca do levantamento e análise dos requisitos, casos de uso e dos diagramas de classe e de sequência, espera-se que o funcionamento do sistema torne-se compreensível ao usuário.

1.1 VISÃO GERAL DO DOCUMENTO

- Seção 2: apresenta-se o estudo acerca dos levantamentos de requisitos do sistema;
- Seção 3: realiza-se a análise dos requisitos levantados anteriormente;
- Seção 4: apresentação dos casos de uso;
- Seção 5: apresentação do diagrama de classes do sistema;
- Seção 6: protótipos do sistema;
- Seção 7: cronograma de atividades
- Seção 8: modelo de processo de desenvolvimento;
- Seção 9: estimativas de esforço e custo;
- Seção 10: orçamento;
- Seção 11: gráfico de gantt
- Seção 12: testes de unidade
- Seção 13: participação de cada membro
- Seção 14: dificuldades encontradas e tomada de decisões
- Seção 15: repositório GitHub

1.2 DESCRIÇÃO DAS SIGLAS

Na seção 3 utilizamos a sigla F e a sigla NF nos códigos dos requisitos que referem-se à Requisito Funcional e Requisito Não-Funcional, respectivamente.

2. LEVANTAMENTO DOS REQUISITOS E VISÃO GERAL DO SISTEMA

2.1 ESCOPO

O sistema de reserva de salas realiza o gerenciamento interno das dependências do campus CCT, tendo como funcionalidades principais o controle das salas, das pessoas que realizam a reserva e diminui o risco de choque de

horário em reservas.

2.2 BENEFÍCIOS DO SOFTWARE

O sistema trará para o usuário que tem a função de reservar a sala, maior controle referente aos horários e salas. Com as informações centralizados no sistema, faz-se possível a otimização de processos, redução de erros e organização das informações - visto que atualmente é feito manualmente, ou seja, em papéis.

2.3 USUÁRIO

O sistema será utilizado somente pelos servidores do campus responsáveis por alugar as salas.

2.4 SAÍDAS DO SOFTWARE

O sistema apresentará como saída os seguintes relatórios:

- Lista dos alunos da graduação, alunos do mestrado e servidores cadastrados;
- Lista das salas que uma pessoa reservou;
- Lista das reservas que uma sala possui;
- Lista de todas as reservas de todas as salas;
- Lista de todas as salas do campus, bem como seu bloco;

2.5 PROBLEMAS QUE SERÃO RESOLVIDOS COM O USO DO SOFTWARE

A utilização do sistema irá resolver os problemas relacionados à descentralização das informações relacionadas ao gerenciamento das reservas, trazendo melhorias no controle dos dados.

2.6 ASPECTOS DO AMBIENTE EM QUE O SOFTWARE SERÁ UTILIZADO

- Computador com acesso ao sistema desenvolvido;
- Servidor responsável pela função de reservas salas;

3. ANÁLISE DOS REQUISITOS

Os requisitos foram divididos nas seguintes categorias:

- Requisitos funcionais: Descrevem elementos que fazem parte das funcionalidades oferecidas pelo sistema.
- Requisitos não funcionais: Descrevem elementos que não fazem parte das funcionalidades oferecidas pelo sistema mas fazem parte do suporte dessas funcionalidades.

3.1 REQUISITOS FUNCIONAIS

<F01> Cadastro de alunos da graduação e do mestrado

Descrição: *tela:* o sistema contém uma tela para o cadastro de alunos da graduação e do mestrado. Para tornar essa funcionalidade possível, a tela conta com campos que devem ser preenchidos pelo usuário. Esses campos são: nome (campo de texto que aceita qualquer tipo de caractere), sobrenome (campo de texto que aceita qualquer tipo de caractere), nascimento (campo de texto formatado para data, ou seja, DD/MM/AAAA), curso (caixa de combinação populada pelos cursos que o campus CCT oferece), cpf (campo de texto formatado para cpf, ou seja, CCC.CCC.CCC-CC), matricula (campo de texto que aceita somente dígitos), semestre de início (campo de texto formatado para AAAA/M) e formação (caixa de combinação populada com os tipos de formação - graduando ou mestrando). O usuário deve clicar em um botão de confirmação - cadastrar - para realizar o cadastro ou em um botão de cancelamento se acaso deseja cancelar a operação e voltar para o menu principal. *funcionalidade:* ao realizar o cadastro, o aluno é adicionado ao banco de dados.

<F02> Busca de alunos da graduação e do mestrado

Descrição: *tela:* o sistema contém a funcionalidade de buscar um determinado aluno através de uma tela onde o usuário deve escolher - pela caixa de combinação - qual o cpf e o primeiro nome do aluno escolhido. As informações deste irão aparecer em campos semelhantes ao da tela de cadastro. Se acaso deseja voltar para o menu principal, deve clicar no botão para retornar. *funcionalidade:* essa função procura e retorna o aluno - relacionado com o cpf fornecida pelo usuário - dentro do banco de dados.

<F03> Alteração de alunos da graduação e do mestrado

Descrição: *tela:* o sistema permite que o usuário altere as informações anteriormente cadastradas de um determinado aluno. Para isso, conta com uma tela semelhante à de cadastro (podendo ser alterado todos os outros campos, exceto cpf e matrícula). O usuário deve clicar em um botão de confirmação para realizar a alteração, ou em um botão de cancelamento se acaso deseja cancelar a operação e voltar para o menu principal. *funcionalidade:* essa função busca o aluno com o

mesmo cpf fornecido pelo usuário e reatribui os novos dados inseridos nele.

<F04> Exclusão de alunos da graduação e do mestrado

Descrição: *tela:* o sistema permite a exclusão de um aluno através de uma tela semelhante a tela de buscar aluno, onde deve ser escolhido - pela caixa de combinação - qual o cpf e o primeiro nome do aluno. Após consulta das informações no banco de dados, as informações do aluno aparecem na tela para verificação. O usuário deve clicar em um botão de confirmação para realizar a exclusão, ou em um botão de cancelamento se acaso deseja cancelar a operação e voltar para o menu principal. *funcionalidade:* essa função busca o aluno no banco de dados - para confirmação dos dados de exclusão - e retira ele do banco.

<F05> Cadastro de servidores

Descrição: *tela:* o sistema contém uma tela para o cadastro de servidores, visto que esses também podem reservar salas. Para tornar essa funcionalidade possível, a tela conta com campos que devem ser preenchidos pelo usuário. Esses campos são: nome (campo de texto que aceita qualquer tipo de caractere), sobrenome (campo de texto que aceita qualquer tipo de caractere), nascimento (campo de texto formatado para data, ou seja, DD/MM/AAAA), cpf (campo de texto formatado para cpf, ou seja, CCC.CCC.CCC-CC), id (campo de texto que aceita somente dígitos), e cargo (caixa de combinação populada com alguns cargos presentes no campus CCT). O usuário deve clicar em um botão de confirmação - cadastrar - para realizar o cadastro ou em um botão de cancelamento se acaso deseja cancelar a operação e voltar para o menu principal *funcionalidade:* ao realizar o cadastro, o servidor é adicionado ao banco de dados.

<F06> Busca de servidores

Descrição: *tela:* o sistema contém a funcionalidade de buscar um determinado servidor através de uma tela onde o usuário deve escolher - pela caixa de combinação - qual o cpf e o primeiro nome do servidor. As informações deste irão aparecer em campos semelhantes ao da tela de cadastro. Se acaso deseja voltar para o menu principal, deve clicar no botão para retornar. *funcionalidade:* essa função procura e retorna o servidor - relacionado com o cpf fornecida pelo usuário - dentro do banco de dados.

<F07> Alteração de servidores

Descrição: *tela:* o sistema permite que o usuário altere as informações anteriormente cadastradas de um determinado servidor. Para isso, conta com uma tela semelhante à de cadastro (podendo ser alterado todos os outros, exceto id e cpf). O usuário deve clicar em um botão de confirmação para realizar a alteração, ou em um botão de cancelamento se acaso deseja cancelar a operação e voltar para o menu principal. *funcionalidade:* essa função busca o servidor com o mesmo

cpf fornecido pelo usuário e reatribui os novos dados inseridos nele.

<F08> Exclusão de servidores

Descrição: *tela:* o sistema permite a exclusão de um servidor através de uma tela semelhante a tela de buscar aluno, onde deve ser escolhido - pela caixa de combinação - qual o cpf e o primeiro nome do servidor. Após consulta das informações no banco de dados, as informações do servidor aparecem na tela para verificação. O usuário deve clicar em um botão de confirmação para realizar a exclusão, ou em um botão de cancelamento se acaso deseja cancelar a operação e voltar para o menu principal. *funcionalidade:* essa função busca o servidor no banco de dados - para confirmação dos dados de exclusão - e retira ele do banco.

<F09> Visualização dos alunos e servidores

Descrição: *tela:* o sistema contém a funcionalidade de fazer a listagem de todos os alunos e servidores presentes no sistema. Através de uma tela, as informações de cada indivíduo aparecem em uma tabela. Não é preciso que o usuário preencha nenhum campo na tela. *funcionalidade:* essa função busca todos os indivíduos presentes na tabela de alunos e servidores, armazenada no banco de dados.

<F10> Cadastro de salas e blocos

Descrição: para essa funcionalidade, decidiu-se que somente os desenvolvedores podem ter acesso à esse cadastro, visto que não é comum a criação de novas salas e blocos. Portanto, como uma forma de diminuir a sobrecarga de informação para o usuário, não há uma tela específica para isso em sua visualização, pois isso é feito pelos criadores manipulando diretamente o banco de dados. São necessárias as seguintes afirmações: bloco, quantidade de salas naquela bloco, número da sala e o tipo da sala.

<F11> Visualização das salas e blocos

Descrição: *tela:* o sistema contém a funcionalidade de fazer a listagem de todas as dependências presentes no sistema, isso é, as salas e os blocos cadastrados. Através de uma tela, as informações aparecem em uma tabela. Não é preciso que o usuário preencha nenhum campo na tela. *funcionalidade:* essa função busca todas as salas e blocos armazenados no banco de dados.

<F12> Cadastro da reserva

Descrição: *tela:* o sistema contém uma tela, separada em duas guias, para o cadastro de reservas. Para tornar essa funcionalidade possível, a tela conta com campos que devem ser preenchidos pelo usuário. Esses campos, na primeira guia, são: usuário (caixa de combinação populada com alunos e servidores), período (caixa de combinação populadas com: manhã, tarde ou noite), data (campo de texto formatado para data, ou seja, DD/MM/AAAA), e bloco (caixa de combinação

populada com todos os blocos). A partir dessas informações é calculado os campos da segunda guia, que são os campos salas disponíveis - visto que quando o aluno é graduando, somente pode reservar salas do tipo 'normal'; já se for aluno mestrado ou servidor, pode reservar salas do tipo 'laboratório' também - e o campo código reserva. Portanto, o campo salas disponíveis (caixa de combinação populada com as salas) e o código de reserva (valor inteiro gerado automaticamente pelo sistema) formam a segunda guia da tela. No fim, o usuário deve clicar em um botão de confirmação - reservar - para realizar o cadastro ou em um botão de cancelamento se acaso deseja cancelar a operação e voltar para o menu principal. *funcionalidade:* ao realizar a reserva, uma nova reserva é adicionada ao banco de dados.

<F13> Busca da reserva

Descrição: *tela:* o sistema contém a funcionalidade de buscar as reservas separadamente, através de uma tela onde o usuário deve escolher - pela caixa de combinação - qual o código da reserva e o primeiro nome do usuário. As informações referentes a reserva deste irão aparecer em campos semelhantes ao da tela de cadastro de reserva. *funcionalidade:* essa função procura e retorna a reserva do usuário, relacionando com o código de reserva, que está armazenada no banco de dados.

<F14> Exclusão da reserva

Descrição: *tela:* o sistema permite a exclusão de uma reserva através de uma tela semelhante a tela de buscar reserva, onde deve ser escolhido - pela caixa de combinação - qual o código da reserva e o primeiro nome do usuário. Após consulta das informações no banco de dados, as informações da reserva aparecem na tela para verificação. O usuário deve clicar em um botão de confirmação para realizar a exclusão, ou em um botão de cancelamento se acaso deseja cancelar a operação e voltar para o menu principal. *funcionalidade:* essa função busca a reserva no banco de dados - para confirmação dos dados de exclusão - e retira ele do banco.

<F15> Visualização das reservas

Descrição: *tela:* o sistema contém a funcionalidade de fazer a listagem de todos as reservas presentes no sistema. Através de uma tela, as informações de cada reserva aparecem em uma tabela. Não é preciso que o usuário preencha nenhum campo na tela. *funcionalidade:* essa função busca todos as reservas armazenada na tabela do banco de dados.

<F16> Visualização das reservas de uma sala

Descrição: *tela:* o sistema contém a funcionalidade de fazer a listagem de todos as reservas de uma sala, que estão presentes no sistema. Através de uma tela, as informações de cada reserva aparecem em uma tabela. O usuário deve escolher - pela caixa de combinação - qual sala e bloco deseja. *funcionalidade:* essa função

busca todos as reservas de uma sala, que estão armazenadas no banco de dados.

<F17> Visualização das reservas de um aluno ou servidor

Descrição: *tela:* o sistema contém a funcionalidade de fazer a listagem de todos as reservas de um usuário, seja ele aluno graduando, aluno mestrando ou servidor. Através de uma tela, as informações de cada reserva aparecem em uma tabela. O usuário deve escolher - pela caixa de combinação - qual usuário deseja.
funcionalidade: essa função busca todos as reservas de um usuário, que estão armazenadas no banco de dados.

Os requisitos apresentados acima foram divididos em categorias de acordo com o objeto sobre os quais agem. Quase todos os requisitos funcionais aqui listados são evidentes, ou seja, são funcionalidades visíveis ao cliente, exceto pelo cadastro de salas e blocos.

Código	Nome	Categoria	Tipo de requisito
F01	Cadastro de alunos da graduação e do mestrado	Controle de usuários	Normal
F02	Busca de alunos da graduação e do mestrado	Controle de usuários	Esperado
F03	Alteração de alunos da graduação e do mestrado	Controle de usuários	Normal
F04	Exclusão de alunos da graduação e do mestrado	Controle de usuários	Normal
F05	Cadastro de servidores	Controle de usuários	Normal
F06	Busca de servidores	Controle de usuários	Esperado
F07	Alteração de servidores	Controle de usuários	Normal
F08	Exclusão de servidores	Controle de usuários	Normal
F09	Visualização dos alunos e servidores	Controle de usuários	Diferencial
F10	Cadastro de salas e blocos	Gerenciamento de salas e blocos	Esperado
F11	Visualização das salas e blocos	Gerenciamento de salas e blocos	Diferencial
F12	Cadastro de reserva	Gerenciamento de reservas	Normal

F13	Busca de reserva	Gerenciamento de reservas	Esperado
F14	Exclusão de reserva	Gerenciamento de reservas	Normal
F15	Visualização das reservas	Gerenciamento de reservas	Diferencial
F16	Visualização das reservas de uma sala	Gerenciamento de reservas	Diferencial
F17	Visualização das reservas de um aluno ou servidor	Gerenciamento de reservas	Diferencial

3.2 REQUISITOS NÃO-FUNCIONAIS

<NF01> Usabilidade: interface gráfica

Descrição: Para facilitar a utilização do sistema, o software foi desenvolvido com a utilização de interfaces gráficas

<NF02> Usabilidade: padrões de projetos

Descrição: Para auxiliar a padronização do sistema e de suas funções, o sistema foi desenvolvido com o padrão de projeto MVC.

<NF03> Segurança: banco de dados

Descrição: Para auxiliar no armazenamento e segurança de dados, o sistema desenvolvido foi integrado ao banco de dados postgresQL.

<NF04> Confiabilidade: testes

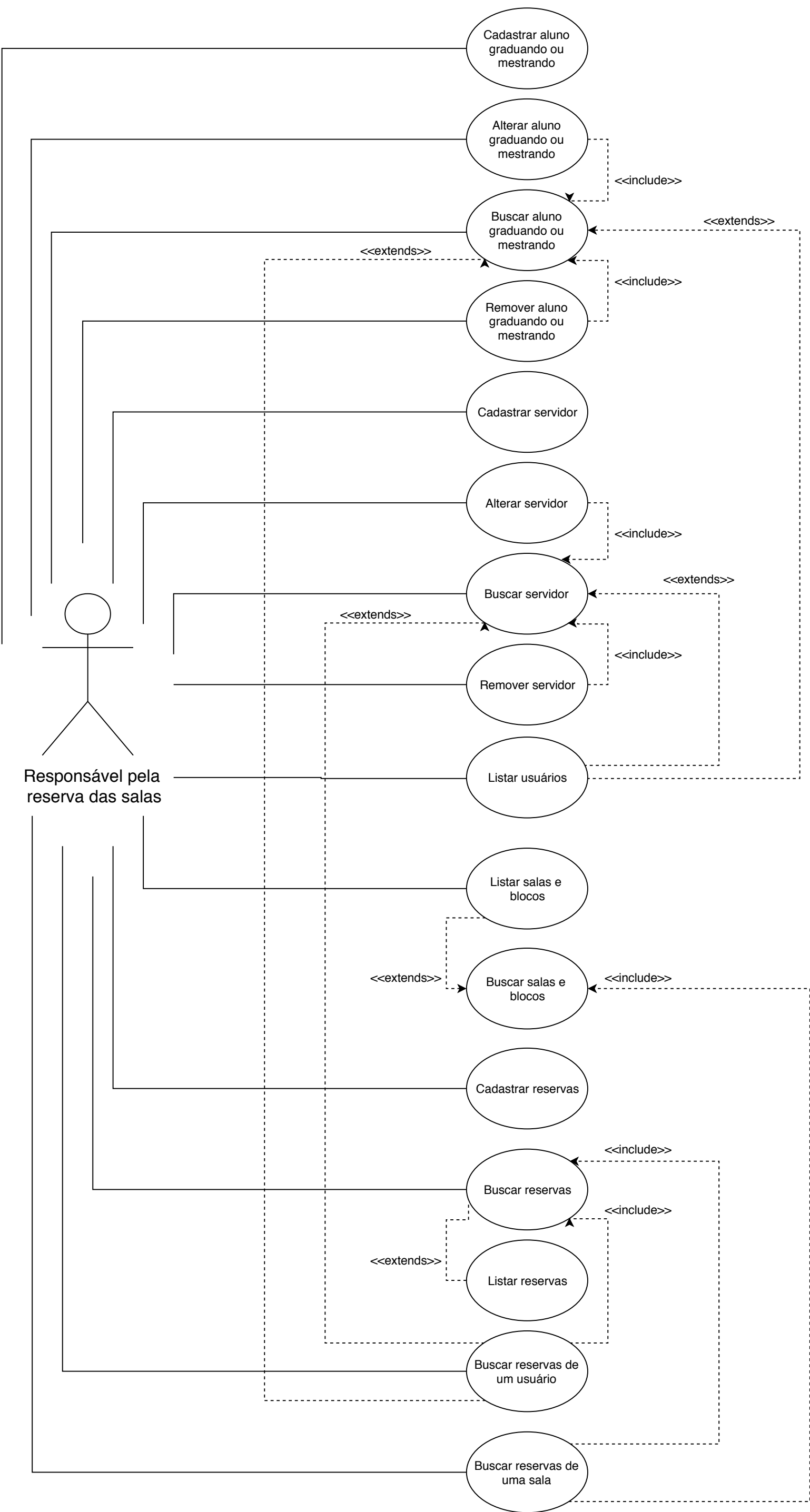
Descrição: Para auxiliar na confiabilidade do sistema, foi desenvolvido testes unitários nos pontos considerados críticos do sistema.

Os requisitos apresentados acima foram divididos em desejáveis e obrigatórios.

Código	Nome	Categoria	Tipo de requisito
NF01	Usabilidade: interface gráfica	Obrigatório	Normal
NF02	Usabilidade: padrões de projetos	Desejável	Esperado
NF03	Segurança: banco de dados	Desejável	Diferencial
NF04	Confiabilidade: testes	Obrigatório	Esperado

4 . DIAGRAMA DE CASOS DE USO

4.1 DESCRIÇÃO DOS CASOS DE USO



<UC01> Cadastrar aluno graduando ou mestrando

Descrição: este caso de uso permite o cadastro de alunos graduando ou mestrando. O ator responsável pela reserva das salas pode realizar esse cadastro no sistema.

Pré-condições: possuir informações do aluno a ser cadastrado.

Pós-condições: o novo aluno é armazenado na base de dados do sistema.

Fluxo normal:

1. Escolhe-se se deseja cadastrar aluno graduando ou mestrando
2. Ator (responsável pela reserva das salas) preenchem as informações do aluno a ser cadastrado.
3. O sistema verifica se os campos foram preenchidos corretamente.
4. A qualquer momento antes de salvar o novo aluno na base de dados, o ator pode cancelar o cadastro.
5. Ator salva o aluno na base de dados.

Fluxo alternativo:

1. Ao tentar cadastrar o aluno na base de dados do sistema, o ator recebe um alerta do sistema indicando que o aluno a ser cadastrado já existe na base de dados. Assim, se continuar com as mesmas informações, o aluno não poderá ser salvo na base de dados.
2. Ao tentar cadastrar o aluno na base de dados do sistema, o ator recebe um alerta do sistema indicando que não preencheu todos os campos da maneira correta. Caso seja atendido o que se pede no alerta, o aluno é salvo na base de dados.

<UC02> Alterar aluno graduando ou mestrando

Descrição: este caso de uso permite a modificação das informações referentes aos alunos. O ator responsável pela reserva das salas pode alterar os alunos no sistema.

Pré-condições: aluno estar cadastrado e possuir informações do aluno a ser alterado.

Pós-condições: o aluno é rearmazenado no sistema.

Fluxo normal:

1. Ator busca o aluno na base de dados do sistema.
2. Altera-se as informações na qual se deseja.
3. O sistema verifica se os campos foram preenchidos corretamente.
4. A qualquer momento antes de alterar o novo aluno na base de dados, o ator pode cancelar a ação.
5. Se caso quiser proceder com a ação, o ator altera o aluno na base de dados.

Fluxo alternativo:

1. Ator busca o aluno na base de dados do sistema, porém, o aluno não está cadastrado. O sistema mandará um alerta indicando que não é possível alterar o aluno se ele não existir na base de dados.
2. Ao tentar alterar o aluno na base de dados do sistema, o ator recebe um alerta do sistema indicando que não preencheram todos os campos

corretamente. Caso seja atendido o que se pede no alerta, o aluno é alterado na base de dados.

<UC03> Buscar aluno graduando ou mestrando

Descrição: este caso de uso permite realizar a consulta de um aluno na base de dados. O ator responsável pela reserva das salas pode consultar os alunos no sistema.

Pré-condições: aluno estar cadastrado.

Pós-condições: será exposto as informações do aluno.

Fluxo normal:

1. Ator busca o aluno pelo seu cpf na base de dados do sistema.
2. O sistema apresentará as informações do aluno pesquisado.

Fluxo alternativo:

1. Ator busca o aluno na base de dados do sistema, porém, o aluno não está cadastrado. O sistema mandará um alerta indicando que não é possível visualizar as informações do aluno se ele não existir na base de dados.

<UC04> Remover aluno graduando ou mestrando

Descrição: este caso de uso permite excluir um aluno na base de dados.

O ator responsável pela reserva das salas pode excluir os alunos no sistema.

Pré-condições: aluno estar cadastrado.

Pós-condições: será exposto as informações do aluno para proceder com a exclusão.

Fluxo normal:

1. Ator busca o aluno pelo seu cpf na base de dados do sistema.
2. O sistema apresentará as informações do aluno pesquisado.
3. A qualquer momento antes de excluir o aluno na base de dados, o ator pode cancelar a ação.
4. Se caso quiser proceder com a ação, o aluno é excluído na base de dados.

Fluxo alternativo

1. Ator busca o aluno na base de dados do sistema, porém, o aluno não está cadastrado. O sistema mandará um alerta indicando que não é possível excluir um aluno se ele não existir na base de dados.

<UC05> Cadastrar servidor

Descrição: este caso de uso permite o cadastro de servidores. O ator responsável pela reserva das salas pode realizar esse cadastro no sistema.

Pré-condições: possuir informações do servidor a ser cadastrado.

Pós-condições: o novo servidor é armazenado na base de dados do sistema.

Fluxo normal:

1. Ator (responsável pela reserva das salas) preenchem as informações do servidor a ser cadastrado.
2. O sistema verifica se os campos foram preenchidos corretamente.

3. A qualquer momento antes de salvar o novo servidor na base de dados, o ator pode cancelar o cadastro.
4. Ator salva o servidor na base de dados.

Fluxo alternativo:

1. Ao tentar cadastrar o servidor na base de dados do sistema, o ator recebe um alerta do sistema indicando que o servidor a ser cadastrado já existe na base de dados. Assim, se continuar com as mesmas informações, o servidor não poderá ser salvo na base de dados.
2. Ao tentar cadastrar o servidor na base de dados do sistema, o ator recebe um alerta do sistema indicando que não preencheu todos os campos da maneira correta. Caso seja atendido o que se pede no alerta, o servidor é salvo na base de dados.

<UC06> Alterar servidor

Descrição: este caso de uso permite a modificação das informações referentes aos servidores. O ator responsável pela reserva das salas pode alterar os servidores no sistema.

Pré-condições: servidor estar cadastrado e possuir informações do servidor a ser alterado.

Pós-condições: o servidor é rearmazenado no sistema.

Fluxo normal:

1. Ator busca o servidor na base de dados do sistema.
2. Altera-se as informações na qual se deseja.
3. O sistema verifica se os campos foram preenchidos corretamente.
4. A qualquer momento antes de alterar o novo servidor na base de dados, o ator pode cancelar a ação.
5. Se caso quiser proceder com a ação, o ator altera o servidor na base de dados.

Fluxo alternativo:

1. Ator busca o servidor na base de dados do sistema, porém, o servidor não está cadastrado. O sistema mandará um alerta indicando que não é possível alterar o servidor se ele não existir na base de dados.
2. Ao tentar alterar o servidor na base de dados do sistema, o ator recebe um alerta do sistema indicando que não preencheram todos os campos corretamente. Caso seja atendido o que se pede no alerta, o servidor é alterado na base de dados.

<UC07> Buscar servidor

Descrição: este caso de uso permite realizar a consulta de um servidor na base de dados. O ator responsável pela reserva das salas pode consultar os servidores no sistema.

Pré-condições: servidor estar cadastrado.

Pós-condições: será exposto as informações do servidor.

Fluxo normal:

1. Ator busca o servidor pelo seu cpf na base de dados do sistema.
2. O sistema apresentará as informações do servidor pesquisado.

Fluxo alternativo:

1. Ator busca o servidor na base de dados do sistema, porém, o servidor não está cadastrado. O sistema mandará um alerta indicando que não é possível visualizar as informações do servidor se ele não existir na base de dados.

<UC08> Remover servidor

Descrição: este caso de uso permite excluir um servidor na base de dados.

O ator responsável pela reserva das salas pode excluir os servidores no sistema.

Pré-condições: servidor estar cadastrado.

Pós-condições: será exposto as informações do servidor para proceder com a exclusão.

Fluxo normal:

1. Ator busca o servidor pelo seu cpf na base de dados do sistema.
2. O sistema apresentará as informações do servidor pesquisado.
3. A qualquer momento antes de excluir o servidor na base de dados, o ator pode cancelar a ação.
4. Se caso quiser proceder com a ação, o servidor é excluído na base de dados.

Fluxo alternativo

1. Ator busca o servidor na base de dados do sistema, porém, o servidor não está cadastrado. O sistema mandará um alerta indicando que não é possível excluir um servidor se ele não existir na base de dados.

<UC09> Listar usuários

Descrição: este caso de uso permite listar todos os usuário - alunos graduandos, mestrandos e usuários - que estão na base de dados. O ator responsável pela reserva das salas pode realizar essa ação.

Pré-condições: usuários cadastrados

Pós-condições: será exposto todos os usuários e suas informações.

Fluxo normal:

1. Ator acessa a área de listagem dos usuários.

Fluxo alternativo:

1. Ator acessa a área de listagem dos usuários, porém, não há usuário cadastrado. O sistema mandará um alerta indicando que não existe nenhum usuário cadastrado na base de dados.

<UC10> Listar salas e blocos

Descrição: este caso de uso permite listar todos as salas e blocos que estão na base de dados (as salas e blocos foram cadastradas pelos desenvolvedores do sistema). O ator responsável pela reserva das salas pode realizar essa ação.

Pré-condições: salas e blocos cadastrados.

Pós-condições: será exposto todos as salas e blocos, bem como o restante das informações pertinentes.

Fluxo normal:

1. Ator acessa a área de listagem das salas e blocos.

<UC11> Buscar salas e blocos

Descrição: este caso de uso permite realizar a consulta de uma sala na base de dados. O ator não tem acesso a este caso de uso diretamente, pois, ele atua indiretamente - <<include>> e <<extends>> - em outros casos de uso.

Pré-condições: sala e blocos cadastrados.

Pós-condições: será exposto as informações da sala.

<UC12> Cadastrar reservas

Descrição: este caso de uso permite o cadastro de reservas. O ator responsável pela reserva das salas pode realizar esse cadastro no sistema.

Pré-condições: possuir as informações da reserva e do usuário que deseja reservar uma sala.

Pós-condições: a nova reserva é armazenada na base de dados do sistema.

Fluxo normal:

1. Ator (responsável pela reserva das salas) preenchem as informações da reserva.
2. O sistema verifica se os campos foram preenchidos corretamente e se já não existe uma reserva para a mesma data e período.
3. A qualquer momento antes de salvar a nova reserva na base de dados, o ator pode cancelar o cadastro.
4. Ator salva a reserva na base de dados.

Fluxo alternativo:

1. Ao tentar cadastrar a reserva na base de dados do sistema, o ator recebe um alerta do sistema indicando que a reserva a ser cadastrada já existe na base de dados. Assim, se continuar com as mesmas informações, a reserva não poderá ser salva na base de dados.
2. Ao tentar cadastrar a reserva na base de dados do sistema, o ator recebe um alerta do sistema indicando que não preencheu todos os campos da maneira correta. Caso seja atendido o que se pede no alerta, a reserva é salva na base de dados.

<UC13> Buscar reservas

Descrição: este caso de uso permite realizar a consulta de uma reserva na base de dados. O ator responsável pela reserva das salas pode consultar as reservas no sistema.

Pré-condições: reserva estar cadastrada.

Pós-condições: será exposto as informações da reserva.

Fluxo normal:

1. Ator busca a reserva pelo seu código e usuário que efetuou a reserva, na base de dados do sistema.
2. O sistema apresentará as informações da reserva pesquisada.

Fluxo alternativo:

1. Ator busca a reserva na base de dados do sistema, porém, a reserva não está cadastrado. O sistema mandará um alerta indicando que não é possível visualizar as informações da reserva se ele não existir na base de dados.

<UC14> Listar reservas

Descrição: este caso de uso permite listar todas as reservas que estão na base de dados. O ator responsável pela reserva das salas pode realizar essa ação.

Pré-condições: reservas cadastrados.

Pós-condições: será exposto todas as reservas e suas informações.

Fluxo normal:

1. Ator acessa a área de listagem das reservas.

Fluxo alternativo:

1. Ator acessa a área de listagem das reservas, porém, não há reserva cadastrada. O sistema mandará um alerta indicando que não existe nenhuma reserva cadastrada na base de dados.

<UC15> Buscar reservas de um usuário

Descrição: este caso de uso permite realizar a consulta de todas as reservas de um usuário que estão na base de dados. O ator responsável pela reserva das salas pode consultar as reservas de um usuário no sistema.

Pré-condições: usuário ter realizado reservas.

Pós-condições: será exposto em uma tabela as informações de todas as reservas.

Fluxo normal:

1. Ator busca o usuário pelo cpf na base de dados do sistema.
2. O sistema apresentará as informações de todas as reservas feitas pelo usuário.

Fluxo alternativo:

1. Ator busca um usuário na base de dados do sistema, porém, o mesmo não realizou nenhuma reserva. O sistema mandará um alerta indicando que não é possível visualizar as informações das reservas se esta não existir na base de dados.

<UC16> Buscar reservas de uma sala

Descrição: este caso de uso permite realizar a consulta de todas as reservas de uma sala estão na base de dados. O ator responsável pela reserva das salas pode consultar as reservas de uma sala no sistema.

Pré-condições: a sala ter reservas.

Pós-condições: será exposto em uma tabela as informações de

todas as reservas.

Fluxo normal:

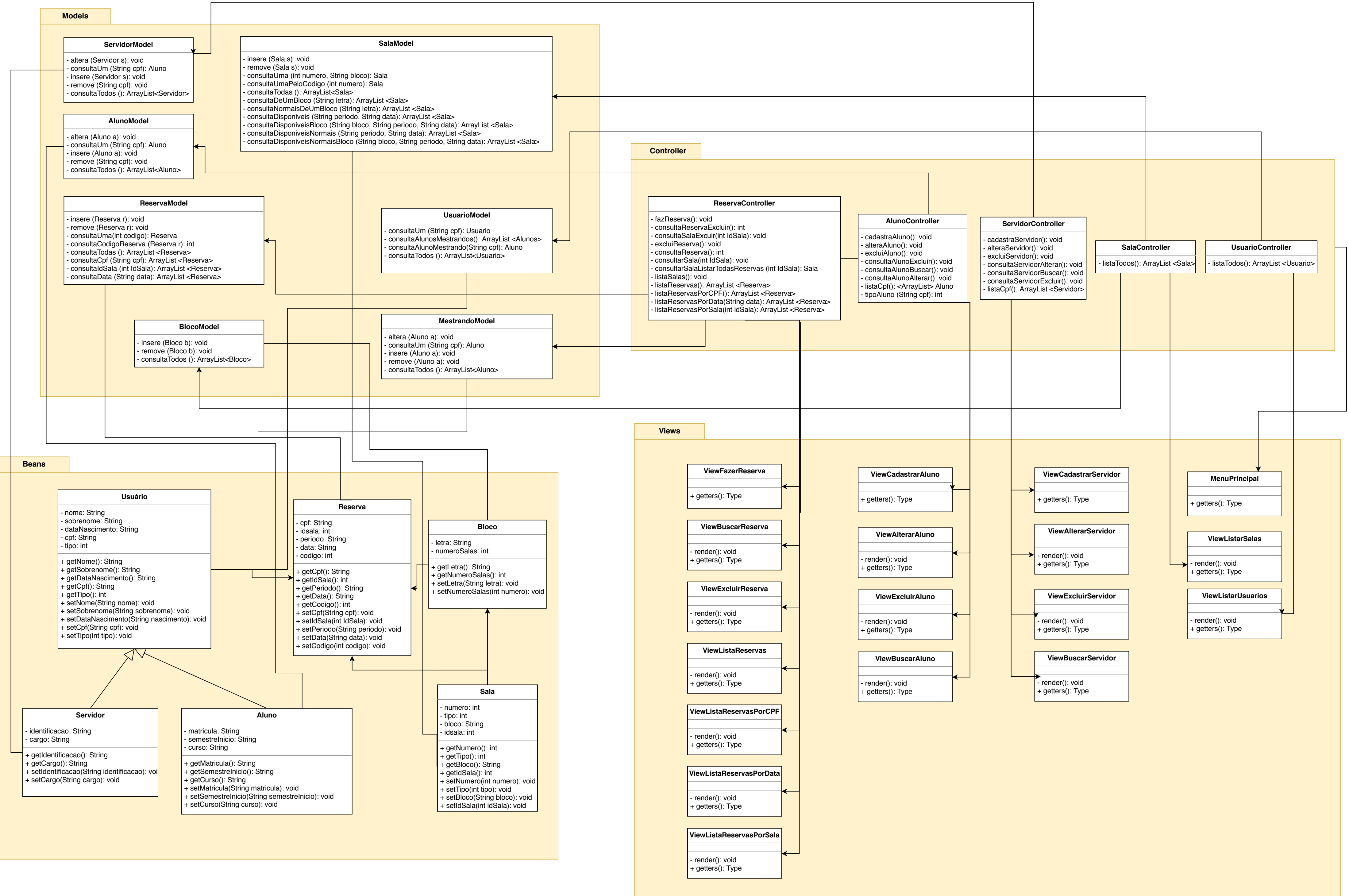
1. Ator busca a sala e bloco na base de dados do sistema.
2. O sistema apresentará as informações de todas as reservas feitas nessa sala.

Fluxo alternativo:

1. Ator busca uma sala e bloco na base de dados do sistema, porém, não há nenhuma reserva agendada para esta dependência. O sistema mandará um alerta indicando que não é possível visualizar as informações das reservas se esta não existir na base de dados.

5. DIAGRAMA DE CLASSES

As classe utilizadas na implementação foram mapeadas de acordo com o Diagrama de Classes apresentado na próxima página.



6. PROTÓTIPOS DO SISTEMA

Alguns protótipos do sistema estão sendo mostrados abaixo.

Menu principal



Cadastro de alunos

The image shows a design preview window titled "Design Preview [ViewCadastrarAluno]". The window has a light gray background. At the top, the title "CADASTRAR ALUNOS" is centered. Below the title are several form fields for student registration:

- NOME:
- SOBRENOME:
- NASCIMENTO:
- CURSO: (dropdown menu)
- CPF:
- MATRÍCULA:
- SEMESTRE INICIO:
- FORMAÇÃO: (dropdown menu)

At the bottom of the form are two buttons: "CANCELAR" and "CADASTRAR".

Lista de usuários cadastrados

Design Preview [ViewListarUsuarios]

USUÁRIOS CADASTRADOS

CPF	Nome	Usuário

VOLTAR

Cadastrar reserva - guia um

Design Preview [ViewFazerReserva]

Opções Sala

Selecione as opções:

Usuário:

Período:

Data:

Bloco:

CANCELAR AVANÇAR

Cadastrar reserva - guia dois

Design Preview [ViewFazerReserva]

Opções Sala

Escolha a sala:

Salas disponíveis:

Código da reserva:

CANCELAR RESERVAR

Os outros protótipos podem ser vistos acessando diretamente o repositório do GitHub, descrito na seção 15.

7. CRONOGRAMA DE ATIVIDADES

Este cronograma de atividade, usando a metodologia ágil Scrum, formará o product backlog. Para o tempo considerou-se horas pois, acredita-se que seria mais eficiente para medir o esforço - hora trabalhada - de cada membro do projeto.

Abaixo têm-se a tabela que descreve o mesmo.

ATIVIDADE	DESCRIÇÃO	RESPONSÁVEL	TEMPO (horas)
A	Listar atividades	TODOS	0,5
B	Cronograma de atividades	EDIANA	0,5
C	Escopo: levantamento de requisitos e descrições	EDIANA	2,5
D	Escopo: diagrama de caso de uso e descrições	EDIANA	1,5
E	Escopo: diagrama de classe	EDUARDO	2,5
F	Criação do banco de dados e conexão	TODOS	1,0

G	Alterar o projeto para o padrão MVC	ÊRICA	3,0
H	Programação Model e expressões SQL	TODOS	12,0
I	Programação View	EDIANA, ÊRICA	15,0
J	Programação Controller	EDIANA, ÊRICA	15,0
K	Estimativa de esforço e custo	EDUARDO	2,0
L	Orçamento	EDUARDO	1,0
M	Modelo de processo de desenvolvimento	TODOS	CONTÍNUO
N	Gráfico de Gantt	EDUARDO	1,0
O	Listagem do percentual de participação de cada membro	TODOS	1,0
P	Testes de unidade	EDUARDO	10,0
Q	Repositório GitHub	ÊRICA	0,5

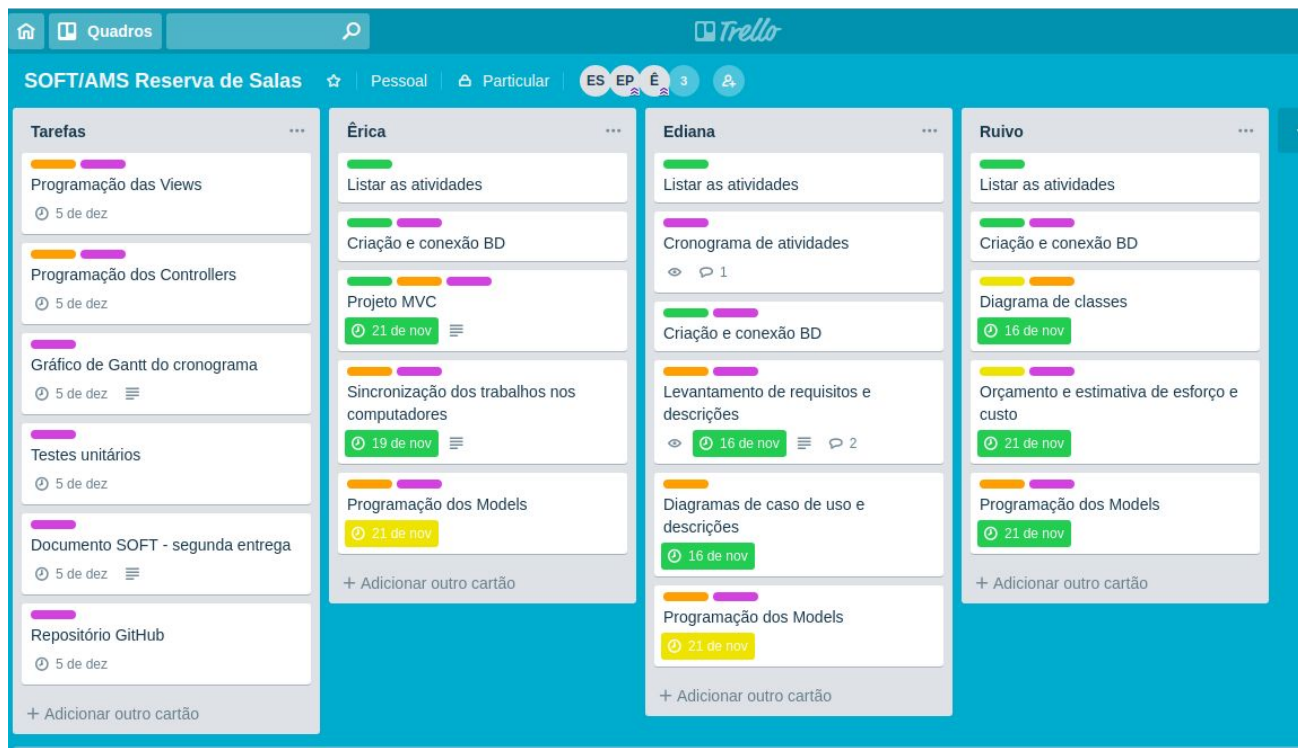
Para ajudar na organização e gerenciamento das atividades, a equipe está utilizando o Trello (uma ferramenta de gerenciamento de projetos).

8. MODELO DE PROCESSO DE DESENVOLVIMENTO

Como modelo de processo de desenvolvimento, optou-se por usar a metodologia ágil Scrum - por familiaridade da equipe com esta - e por acreditar na adaptabilidade, transparência e melhoria contínua que o modelo oferece.

Para isso, decidiu-se que o tempo de cada sprint seria de uma semana. Como visto na tabela anterior, no product backlog já temos o responsável por cada atividade (fizemos isso para estimativa de esforço e custo, que será abordado na próxima seção). Para um melhor acompanhamento do sprint backlog utilizamos o trello.

Durante todo o projeto serão realizados cinco sprints. Atualmente, estamos na terceira sprint. Na página seguinte têm-se uma figura com as atividades realizadas até o momento.



9. ESTIMATIVAS DE ESFORÇO E CUSTO

Durante a elaboração do nosso cronograma, nos utilizamos da técnica de planning poker para a estimativa do esforço e tempo do projeto a ser desenvolvido.

O planning poker foi realizado entre todos os integrantes do grupo e as cartas de cada membro representavam o esforço em equipe-hora. Dezesete atividades foram analisadas e a partir da média das estimativas de cada membro chegamos que o esforço total do projeto é de sessenta e nove horas de trabalho pela equipe. Como o esforço, nesse caso, é proporcional à duração, teremos:

Esforço: 69 equipe-hora;

Duração: 69h.

Na estimativa dos custos, temos que eles estão atrelados unicamente aos recursos de capital e recursos de consumo, uma vez que softwares e frameworks necessários e utilizados para o desenvolvimento de todo o projeto são fornecidos de gratuitamente.

Logo, levando em conta o número de horas que serão trabalhadas, juntamente com o número de reuniões realizadas com o cliente até a entrega do projeto, temos que os custos relativos aos gastos com energia elétrica serão de R\$ 100.00, hardwares será de R\$ 250.00 e locomoção de R\$ 250.00:

Custo total: R\$100.00 + R\$ 250.00 + R\$250.00 = R\$ 600.00.

10. ORÇAMENTO

Para o orçamento do projeto, foram levados em conta todos os custo previamente estimados, o valor de R\$ 25.00/h para os analistas-programadores Java,

considerando estes como desenvolvedores juniores e uma margem de lucro de 15% sobre o subtotal do projeto. Chegando assim ao seguinte orçamento:

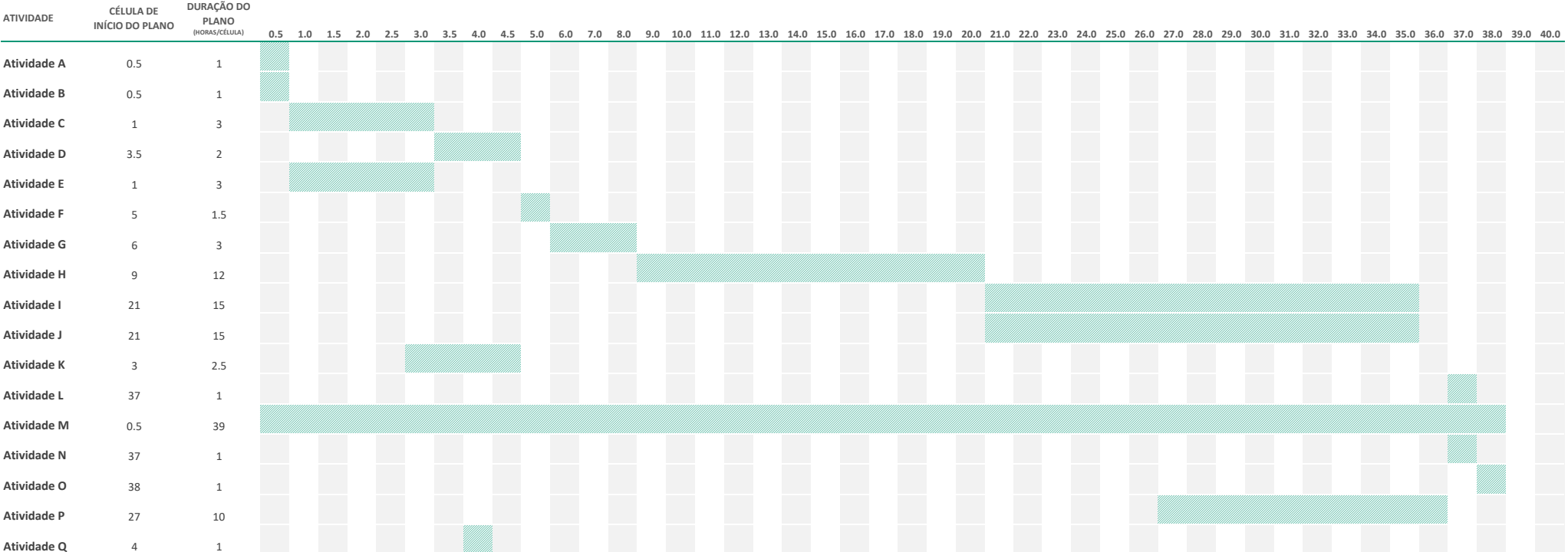
ORÇAMENTO				
ITEM	PRODUTO/SERVIÇO	QUANT.	VALOR	SUBTOTAL: R\$ 2 325.00
1	Desenvolvimento/hora dos programadores	69	R\$ 25.00	ACRÉSCIMO: R\$ 348.75
2	Recursos de capital	-	R\$ 250.00	TOTAL: R\$ 2 673.75
3	Recursos de consumo	-	R\$ 350.00	

11. GRÁFICO DE GANTT

Modelamos o Gráfico de Gantt com base no cronograma de atividades anteriormente definido pela equipe, as atividades recebem as nomeações de A à Q como na tabela do cronograma, no gráfico foi possível ilustrar o avanço das diferentes atividades do projeto. Os intervalos de tempo representando o início e fim de cada fase aparecem como barras coloridas sobre o eixo horizontal do gráfico. Também mapeamos por meio de colunas as células de início de cada atividade e suas respectivas durações.

O Gráfico de Gantt será exibido na próxima página.

Sistema de Reserva de Salas para UDESC - Gráfico de Gantt



12. TESTES DE UNIDADE

Levando em conta a modelagem do nosso diagrama de classes, o único pacote em que os testes de unidade deveriam ser implementados era sobre as classes do package controller, nas quais possuímos os métodos de cadastramento, exclusão, listagem, buscas e alteração de alunos, servidores e reservas.

Como nossa cobertura de testes não abrangeria todo o código, optamos por testar os métodos em que a complexidade ciclomática era maior, os métodos são os de cadastramento, alteração e exclusão de alunos e servidores porque nesses métodos precisamos realizar algumas verificações relacionadas ao tipo e por conta dos campos adicionais que possuem em apenas um ou outro objeto. No caso do controller de reserva, testamos apenas os métodos relacionados ao cadastramento e exclusão de uma, seguindo as mesmas justificativas apresentadas anteriormente.

Vale ressaltar que os testes realizados no controller são importantes porque possuem os métodos responsáveis por todos os requisitos levantados e também fazem a conexão direta com o banco de dados. No total, foram realizados 16 testes, com casos de sucesso e falha de cada um dos métodos.

13. PARTICIPAÇÃO DE CADA MEMBRO

Cumprindo com as obrigações vistas na tabela da seção 7, todos os membros realizaram as atividades que haviam escolhido do product backlog. Portanto, o percentual de participação de cada membro, levando em conta as horas totais do projeto (69 horas), é dado pela tabela abaixo.

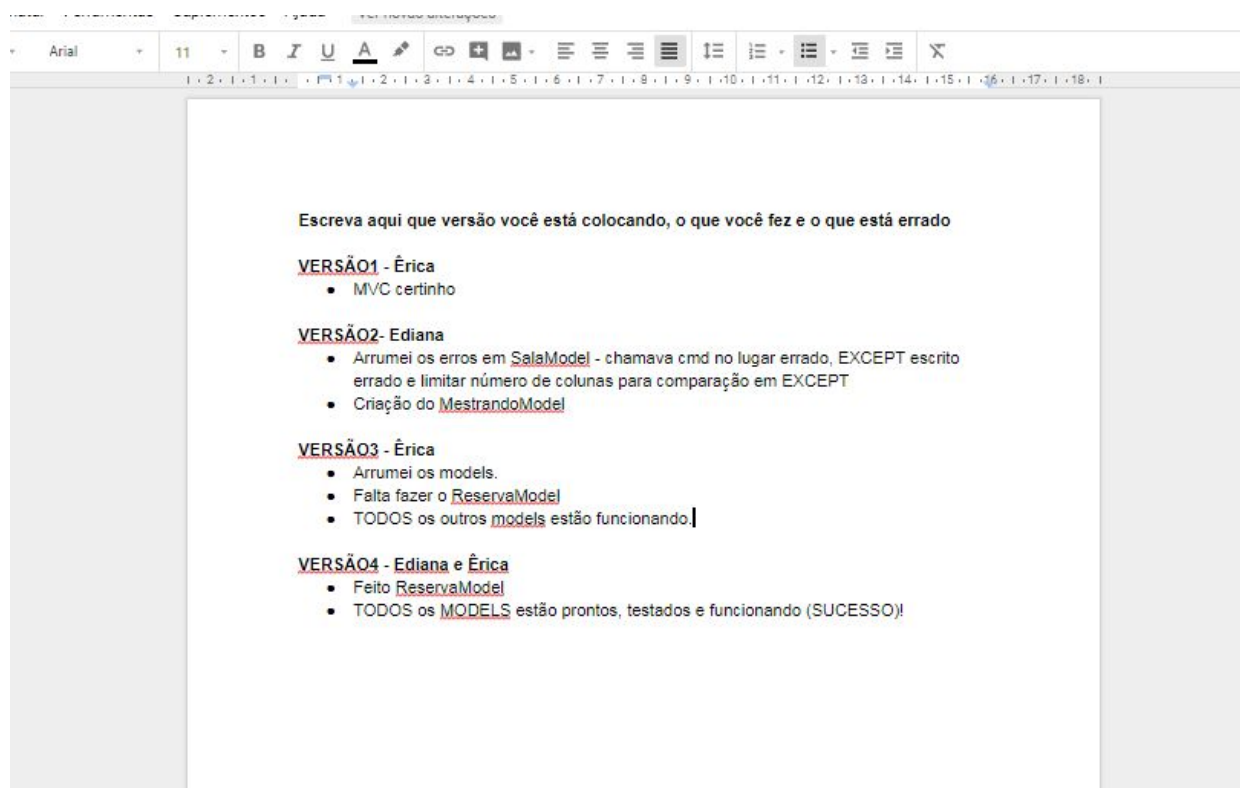
MEMBRO	HORAS EMPENHADAS	PORCENTAGEM
Ediana	24,333	35.26%
Eduardo	21,333	30.92%
Êrica	23,333	33,82%

14. DIFICULDADES ENCONTRADAS E TOMADA DE DECISÕES

A primeira decisão que a equipe tomou, e que definiu o tema do projeto, foi criar e adaptar um banco de dados para o sistema de reserva de salas. Com o banco de dados feito, também realizou-se incrementos de consultas, como:

- Consulta de todas as reservas de uma sala;
- Consulta de todas as reservas de um usuário;
- Consulta de todas as reservas em uma data;
- Consulta de todas as reservas em um período;

Com isso, veio a dificuldade (pois foi a parte do projeto que mais demandou tempo), de alterar todas as programações de view's e controller's para aceitar o banco de dados. Indo de encontro à isso, notou-se a dificuldade da programação paralela - isto é, membros da equipe alterando o código ao mesmo tempo - visto que a equipe não tem familiaridade com os branches do GitHub. Para contornar essa situação, a equipe resolveu que cada membro iria combinar o momento de alterar os códigos fontes, de maneira a criar versões sempre que uma alteração era realizada, de forma a manter uma documentação atualizada do que continha em cada versão, como segue a imagem abaixo¹.



A imagem descreve a documentação das versões pelos membros. O programa seguiu até a versão 10.

Outra decisão importante foi a escolha de cada atividade para cada membro, retirando esta do product backlog. Não houve dificuldade para essa ocasião, mas ela acabou por refletir no trabalho de cada membro da equipe.

Por fim, houve dificuldade também na escolha dos pontos críticos para o teste unitário do sistema. Para resolver essa situação, a equipe realizou uma reunião onde, cada membro da equipe apontou as partes do sistema que deveriam ser testadas e porquê. Os pontos que tiveram intersecção entre os membros da equipe foram testados.

¹ Para ver o documento completo acesse o link:

<<https://docs.google.com/document/d/1ITC2y3o9hc6eTS96gM9yI3GWr-qGedZXnwZHfDEUP8/edit?usp=sharing>>

15. REPOSITÓRIO GITHUB

Endereço do repositório no GitHub:

<https://github.com/ericapetersc/reserva-salas>