

Universidade Federal Rural do Semi-Árido
Centro Multidisciplinar Pau dos Ferros
Curso: Gerência De Configuração E Mudanças
Disciplina: Engenharia de Usabilidade
Prof.: Alysson Filgueira Milanez

Atividade

Emanuel Bruno Duarte de Moraes
Lucas Matheus Alexandre Da Silva
Taísso Reni De Souza Melo

Pau dos Ferros RN
2024

DEFINIÇÃO

O SICC é uma solução inovadora de software para a reserva de cabines na biblioteca da Ufersa-PDF, que permite automatizar processos que eram feitos com uso de papel, na qual, consome muito tempo. O software vai receber dados dos alunos na hora da reserva das cabines, armazenando-os em um banco de dados, onde o usuário poderá verificar as informações de quem reservou as cabines, e também se existem cabines livres para a reserva.

O software irá contar com algumas funcionalidades onde poderão ser realizados o login dos funcionários, importar/exportar alunos, reservas das cabines e/ou seu histórico. A função que irá importar/exportar um aluno é muito importante, visto que, isso influenciará diretamente na redução da recorrência do procedimento. Ou seja, na próxima vez que o estudante for utilizar as cabines da biblioteca, este simplesmente irá fazer seu login.

As reservas das cabines é fundamental no seguimento da automatização, pois com ela iremos diminuir a utilização do papel e o tempo gasto para a realização da tarefa. A utilização do login para cada funcionário influencia diretamente nas possíveis notificações que os usuários das cabines podem receber quanto para a administração dos cadastros.

INFORMAÇÕES EXTRAS

O projeto será hospedado no [Github](#), utilizando a linguagem de programação [Python](#).
Repositório do projeto: [Link](#).

REQUISITOS DO PROJETO

Requisito	Descrição
RF001 - Cadastrar Administrador	O sistema deve permitir cadastrar um novo administrador com login e senha.
RF002 - Editar Administrador	O sistema deve permitir a edição dos dados do administrador.
RF003 - Remover/desativar conta do Administrador	O sistema deve permitir a exclusão da conta do administrador por outro administrador e a permissão do administrador de realizar login (conta ativa/desativa).
RF004 - Cadastrar Servidor	O administrador pode cadastrar servidores, com base nas informações de login e senha.

RF005 - Editar Servidor	O administrador pode editar servidores.
RF006 - Remover/desativar conta do Servidor	O administrador pode excluir a conta ou retirar a permissão do servidor de realizar login.
RF007 - Cadastrar Cabine	O sistema deve permitir que o administrador cadastre uma nova cabine com base no número, funcionalidade da cabine e capacidade.
RF008 - Editar Cabine	O sistema deve permitir aos administradores editar as informações das cabines.
RF009 - Remover Cabine	O sistema deve permitir aos administradores a remoção das cabines cadastradas no sistema.
RF010 - Importar/cadastrar Dados dos Alunos	O sistema deve permitir ao administrador realizar o cadastro dos alunos por meio de formulário.
RF011 - Reservar Cabine	O servidor deve realizar a reserva de discentes associados a uma cabine, incluindo os materiais que dispõem para uso.
RF012 - Autenticar Aluno	O sistema deve permitir que o discente realize a autenticação quando necessário para visualizar uma cabine.
RF013 - Realizar Login	O sistema deve permitir o login do administrador, com base nas informações do requisito RF001 para o administrador e RF004 para o Servidor.
RF014 - Bloquear Cabine	O sistema deve permitir o bloqueio do acesso à cabine, em casos como não estar no horário de atendimento ou por algum defeito na cabine.
RF015 - Liberar Cabine	O sistema deve liberar a cabine para o estado livre, quando for necessário desbloqueá-la ou se o horário alocado para uso finalizar ou caso o usuário deixe a cabine.

DIAGRAMA DE CLASSES

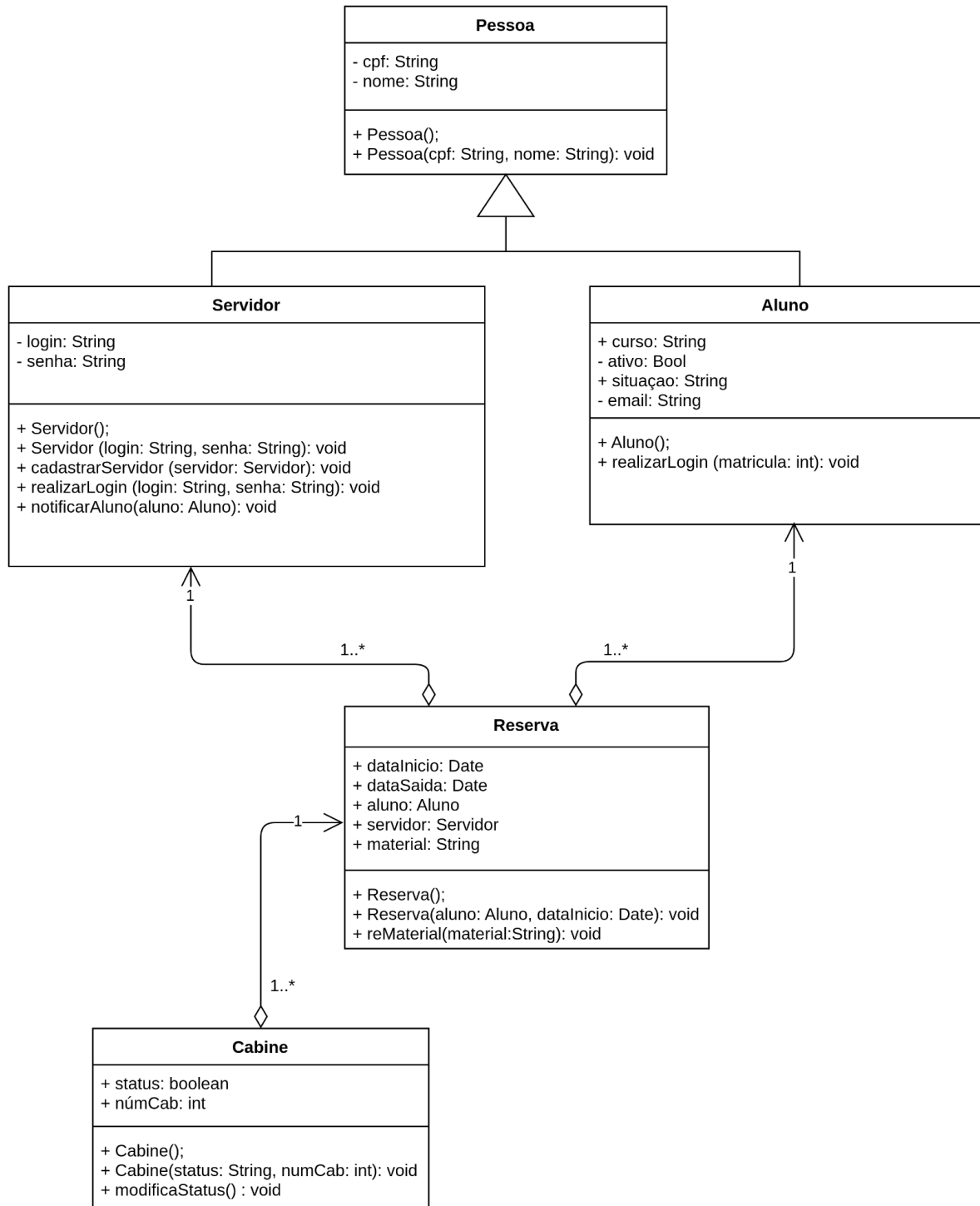
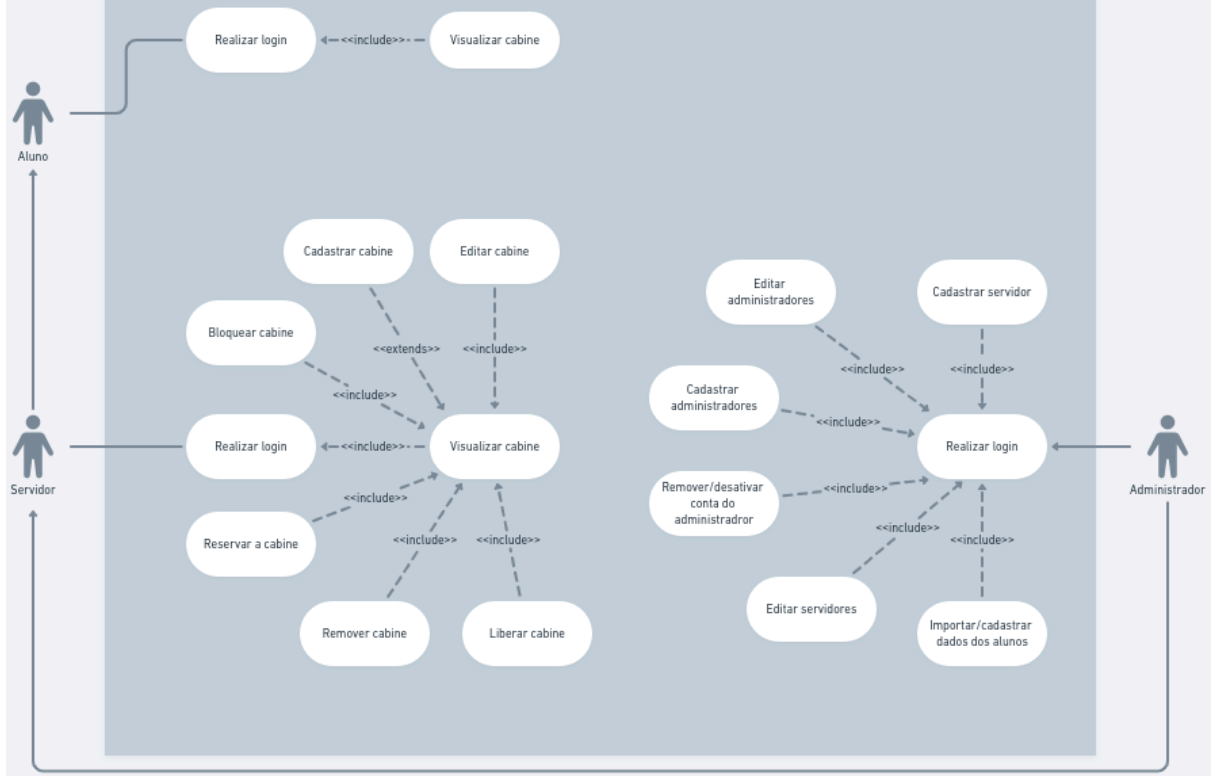


DIAGRAMA DE CASOS DE USO

SISTEMA INTEGRADO DE CONTROLE DE CABINE



TESTES

Os testes serão aplicados no sistema com base em um projeto construído pela framework django:

— Testes Unitários

- Testes para verificar se métodos e views (controlador das informações das páginas em projetos com python/django) estão funcionando corretamente.
- Testes para verificar se partes específicas do código estão funcionando corretamente.
- Será realizado testes utilizando Unittest

— Testes Funcionais

- Simulação de usuários com o uso do sistema.
- Estes testes verificam se as funcionalidades do sistema estão de acordo com o esperado.
- Testando fluxos como cadastros de formulários, funcionalidade dos botões e navegação entre páginas.