



Sistema CRUEL - Cardários do RU para Estudantes Legais

ENUNCIADO

Um Restaurante Universitário (RU) de uma grande Universidade Federal precisa de um sistema para controlar seus cardápios e divulgá-los na internet. Este RU também precisa controlar quantas pessoas se alimentaram no estabelecimento, para gerenciar seus custos e receitas.

REQUISITOS FUNCIONAIS

Visão do Atendente de Caixa

O atendente é quem recebe os clientes e recebe o pagamento. Os clientes podem ser de quatro categorias:

- Professor;
- Servidor;
- Aluno;
- Externo.

O atendente deve poder registrar a entrada de um cliente. O atendente pode:

- **Efetuar Login/Logout**
- **Registrar Cliente:** inserir a entrada do cliente, registrando sua categoria, data/hora e valor cobrado. Não se deve cadastrar qualquer informação pessoal do cliente;
- **Consultar os Registros:** deve-se permitir consultar (em tela) os registros feitos em um determinado dia. Como default (data previamente preenchida), deve mostrar os registros do dia, ordenados de forma decrescente;
- **Remover um Registro:** em caso de lançamento errado, na tela de consulta de registros a atendente pode removê-lo;
- **Alterar um Registro:** em caso de lançamento errado, na tela de consulta de registros a atendente pode alterar a categoria do registro, e todas as informações deverão ser atualizadas;

Visão do Gerente

O gerente é quem efetua os cadastros básicos do sistema e visualiza os relatórios efetuados. O gerente pode:

- **Efetuar Login/Logout**
- **Manter Atendentes:** inserir, atualizar, listar e remover o cadastro de um atendente. Os dados básicos são: Nome, CPF, Endereço completo, telefone, e-mail (que será o login no sistema), senha.
 - **Consultar Atendentes:** deve-se permitir a consulta em tela de atendentes através de um campo de texto. Esse texto deve ser pesquisado como parte do nome, ou parte do e-mail ou CPF.
 - **Remover Atendentes:** o atendente não pode mais se logar no sistema, mas não é removido, seu registro é inativado.

- **Manter Nutricionista:** inserir, atualizar, listar e remover o cadastro de um nutricionista. Os dados básicos são: Nome, CPF, CRN, Endereço completo, telefone, e-mail (que será o login no sistema), senha.
 - **Consultar Nutricionista:** deve-se permitir a consulta em tela de nutricionistas através de um campo de texto. Esse texto deve ser pesquisado como parte do nome, ou parte do e-mail, ou CPF, ou CRN.
 - **Remover Nutricionista:** o nutricionista não pode mais se logar no sistema, mas não é removido, seu registro é inativado.
- **Manter Gerente:** inserir, atualizar, listar e remover o cadastro de um gerente. Os dados básicos são: Nome, CPF, Endereço completo, telefone, e-mail (que será o login no sistema), senha.
 - **Consultar Gerente:** deve-se permitir a consulta em tela de gerentes através de um campo de texto. Esse texto deve ser pesquisado como parte do nome, ou parte do e-mail, ou CPF.
 - **Remover Gerente:** o gerente não pode mais se logar no sistema, mas não é removido, seu registro é inativado. O gerente não pode desativar a si mesmo.
- **Cadastro de Tipo de Cliente:** o gerente deve poder cadastrar os tipos de clientes e os valores de suas refeições;
- **Relatório de Receita Mensal (PDF):** é um relatório contendo tudo o que foi arrecadado com os almoços e jantares durante um mês selecionado (atual por default). O relatório deve trazer os dados agrupados por dia do mês, indicando qual dia da semana é iniciando no dia 01 e terminando no dia correto. Para cada dia do mês deve mostrar o total arrecadado no almoço e no jantar. Deve-se apresentar o valor total arrecadado naquele mês, no final do relatório;
- **Relatório de Receita Anual (PDF):** é um relatório contendo tudo o que foi arrecadado com os almoços e jantares durante um ano selecionado (atual por default). O relatório deve trazer os dados agrupados mês. Para cada mês deve mostrar o total arrecadado no almoço e no jantar. Deve-se apresentar o valor total arrecadado naquele ano, no final do relatório;

Visão do Nutricionista

O nutricionista é quem cadastra os ingredientes e cardápios no sistema. Ele pode:

- **Efetuar Login/Logout**
- **Tela Inicial:** deve mostrar uma lista com os dias do mês corrente e uma indicação se já foi feito ou não o cardápio deste dia. A cada registro, deve-se ter um link para que se possa OU cadastrar o cardápio do dia OU consultar o cardápio do dia;
- **Lista de Cardápios do Mês:** a qualquer momento a nutricionista pode consultar os cardápios do mês e ter uma listagem como a que aparece na tela inicial; Também é possível escolher de qual mês se deseja ver o cardápio.
- **Manter Ingrediente:** inserir, atualizar, listar e remover o cadastro de um ingrediente. Os dados básicos são: Nome, Tipo (carne, salada, sobremesa, etc), Descrição.
- **Manter Tipo de Ingrediente:** inserir, atualizar, listar e remover o cadastro de um tipo de ingrediente. O dado básico do tipo é: Nome;
- **Manter Cardápio:** Um cardápio é baseado no dia (almoço e jantar), portanto o cardápio de um dia envolve duas refeições, nas quais deve-se ter: carne, arroz, feijão, acompanhamento, salada e sobremesa. No mesmo dia não se pode repetir nem a carne nem a sobremesa. Entra-se na tela de manutenção de cardápio via Lista de Cardápios do Mês. Se um cardápio já foi cadastrado, ele é visualizado e pode ser alterado. Se não foi cadastrado, então a tela de preenchimento dos ingredientes é aberta.

Visão do Cliente

Para o cliente é disponibilizada uma página do sistema que mostra todos os cardápios da semana atual (domingo a sábado). Não é necessário login.

REQUISITOS NÃO-FUNCIONAIS

Os requisitos não-funcionais mínimos são:

- Senhas devem ser criptografadas;
- O leiaute deve seguir preceitos de usabilidade e ergonomia, usar o ErgoList como direcionador (<http://www.labiutil.inf.ufsc.br/ergolist/>);
- O leiaute deve ser agradável;
- Obrigatório o uso de JSP, Servlets, JSTL/EL, no modelo MVC apresentado (Visão - JSP/JSTL/EL, Controle - Servlets, Modelo - Classes e DAO);
- Todas as datas e valores monetários devem ser entrados e mostrados no formato brasileiro;
- Todos os campos que tiverem formatação devem possuir máscara;
- Todas as datas deverão ser entradas através de calendários;
- Todos os relatórios devem ser feitos em PDF, usando iReport/Jasper;
- As informações devem estar normalizadas no banco de dados.

Para o funcionamento distribuído (usando Webservice) deve-se:

- Implementar duas aplicações diferentes, que rodam em servidores diferentes (que podem estar na mesma máquina, em portas diferentes):
 - **Aplicação do Gerente:** contendo todas as funcionalidades do gerente
 - **Aplicação Operacional:** contendo todas as demais funcionalidades
- Cada aplicação só tem acesso direto ao seu banco de dados específico;
- Usar dois bancos de dados separados:
 - **Banco de Dados do Gerente:** contendo as informações de atendentes, gerentes e nutricionistas. É neste banco de dados que é verificado o login;
 - **Banco de Dados Operacionais:** contendo todas as demais informações, inclusive os cardápios feitos.
- Quando uma aplicação necessita de informações que não estão em seu banco de dados deve:
 - Fazer uma chamada ao Webservice que responde por aquelas informações;
 - Se a aplicação precisa de um conjunto de dados, por exemplo para fazer um relatório, ela pode buscar as informações no momento que necessitar, armazenar em tabelas temporárias, e realizar a tarefa;
- Implementar os Webservices via REST.