

PONTIFÍCIA UNIVERSIDADE CATÓLICA DE MINAS GERAIS
INSTITUTO DE CIÊNCIAS EXATAS E INFORMÁTICA
UNIDADE EDUCACIONAL CORAÇÃO EUCARÍSTICO
Bacharelado em Engenharia de Software

Integrantes: André Nestor Celino e Felipe Araújo Monteiro
Hotel Descanso Garantido

Apresentação:

Descanso Garantido é um hotel que tem como objetivo atender bem seus clientes, além disso, procura fidelizá-los. Está localizado no centro de Itacaré – BA e possui alguns funcionários com os seguintes cargos: (recepcionista, auxiliar de limpeza, garçom, gerente). Acontece que até hoje o Hotel Descanso Garantido fazia seus controles de estadias, clientes e funcionários em planilhas do excel e cadernos, o que tem gerado diversos problemas para a organização. Sem falar que muitas vezes um mesmo quarto é reservado para mais de um cliente. Diante dos problemas vividos pela Descanso Garantido, o hotel resolveu contratar uma empresa desenvolvedora de sistemas. Sendo assim, é necessário compreender a real necessidade do hotel e desenvolver um software específico. A seguir foi descrito como deverá ser o sistema, bem como suas restrições.

Estrutura do Projeto:

O projeto está dividido em várias classes, cada uma responsável por uma parte específica do sistema. As principais classes são:

- Cliente
- Estadia
- Funcionário
- Quarto

Planejamento dos Casos de Teste:

Classe Cliente

Função	Entrada	Procedimento de Teste	Saída Esperada
Cliente::setNome	“João”	Chamar setNome(“João”) e verificar com getNome()	“João”
Cliente::setCpf	“12345678900”	Chamar setCpf(“12345678900”) e verificar com getCpf()	“12345678900”
Cliente::setTelefone	“31999999999”	Chamar setTelefone(“31999999999”) e verificar com getTelefone()	“31999999999”

The screenshot shows a C++ IDE with a project named 'CPUNHAN LIMITED'. The code is organized into several files: main.cpp, cliente.cpp, funcionario.cpp, estadia.cpp, quarto.cpp, cliente.h, estadia.h, and quarto.h. The main.cpp file contains the following code:

```
11
12 cout << "Digite:" << endl;
13 cout << "1 - Cadastrar um cliente" << endl;
14 cout << "2 - Procurar um cliente" << endl;
15 cout << "3 - Calcular pontos de fidelidade" << endl;
16 cin >> menuCliente;
17
18 switch (menuCliente) {
19     case 1:
20         cadastrarCliente();
21         break;
22     case 2:
23         buscarCliente();
24         break;
25     case 3:
26         calcularPontosFidelidade();
27         break;
28     default:
29         cout << "Valor inválido" << endl;
30         break;
31 }
32
33
34 bool idExistenteCliente(int id) {
35     // Verifica se o ID já existe no arquivo
36     ifstream arquivo("cliente.txt");
37     if (!arquivo.is_open()) {
38         cout << "Erro ao abrir arquivo para leitura" << endl;
39         return false;
40     }
41
42     // Para cada linha do arquivo, lê o ID e verifica se é igual ao ID
43     string line;
44     while (getline(arquivo, line)) {
45         if (line.find("ID: " + to_string(id)) != string::npos) {
46             arquivo.close();
47             return true;
48         }
49     }
50 }
```

The console output shows the following sequence of events:

```

Digite:
1 - Sessão do cliente
2 - Sessão do funcionario
3 - Cadastrar quarto
4 - Cadastrar uma estadia
5 - Finalizar
1
Digite:
1 - Cadastrar um cliente
2 - Procurar um cliente
3 - Calcular pontos de fidelidade
1
Digite os 3 dígitos referentes do CPF: 123
Digite o nome do cliente: João
Digite o endereço: Rua Ichurusbago
Digite o telefone: 3199999999
Cliente cadastrado com sucesso

```

Função Principal

A função 'main' vai ser o responsável por gerenciar o menu principal do programa e também por chamar as funções apropriadas com base na escolha do usuário.

```
1  #include <iostream>
2  #include <locale.h>
3  #include "cliente.hpp"
4
5  using namespace std;
6
7  int main()
8  {
9      setlocale(LC_ALL, "Portuguese");
10
11      int menu;
12
13      cout << "Digite:" << endl;
14      cout << "1 - Sessão do cliente" << endl;
15      cout << "2 - Sessão do funcionário" << endl;
16      cout << "3 - Cadastrar quarto" << endl;
17      cout << "4 - Cadastrar uma estadia" << endl;
18      cout << "5 - Finalizar" << endl;
19      cin >> menu;
20
21      switch (menu) {
22      case 1:
23          sessaoCliente();
24          break;
25      case 2:
26          sessaoFuncionario();
27          break;
28      case 3:
29          cadastroQuarto();
30          break;
31      case 4:
32          sessaoEstadia();
33          break;
34      case 5:
35          exit(1);
36      default:
37          cout << "Valor inválido";
38          break;
39      }
40
41      return 0;
42  }
```

Relatório de Execução dos Testes:

Função	Entrada	Saída Esperada	Saída Real	Passou no teste?
Cliente::setNome	"João"	"João"	"João"	Sim
Cliente::setCpf	"12345678900"	"12345678900"	"12345678900"	Sim
Cliente::setTelefone	"31999999999"	"31999999999"	"31999999999"	Sim

Casos de testes do Software:

Entrada	Saída Esperada
Nome: "João", CPF: "12345678900", Telefone: "31999999999"	Objeto Cliente com dados corretos

Relatório de Execução de Testes

Entrada	Saída Esperada	Saída Real	Passou no Teste?
Nome: "João", CPF: "12345678900", Telefone: "31999999999"	Objeto Cliente com dados corretos	Objeto Cliente com dados corretos	Sim