
 <p>INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA SÃO PAULO</p>	<p>INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA DE SÃO PAULO CAMPUS – São Paulo</p>	
SPOWEB1		
Linguagem de Programação	Projeto Hotelaria	3º e 4º. Bimestre
Profª. Claudete de Oliveira Alves		Observação: Equipe de 4 alunos



Observação: a imagem acima é apenas uma ilustração

Sistema de Controle de Hotelaria

Introdução ao Projeto de Controle de Hotelaria

Imagine que você é um desenvolvedor de software contratado por um grande hotel para criar um sistema que facilite a gestão das operações diárias. O hotel quer garantir que seus hóspedes tenham uma estadia confortável e que todos os serviços sejam geridos de forma eficiente. Seu desafio é criar um sistema que permita ao hotel controlar reservas, aluguéis de quartos, serviços adicionais e pagamentos de forma organizada.

História

O Hotel Encantado é um lugar maravilhoso que recebe visitantes de todo o mundo. Para oferecer a melhor experiência possível, o hotel quer implementar um sistema digital que ajude a organizar tudo, desde as reservas de quartos até o pagamento das contas. O sistema deve ser fácil de usar tanto para os hóspedes quanto para os funcionários do hotel.

Requisitos do Sistema:

1. Cadastro de Hóspedes:

- Antes de alugar um quarto, o hotel precisa ter as informações básicas do hóspede, como nome, endereço e contato. Se o hóspede já estiver registrado, suas informações devem ser atualizadas, se necessário.

2. Reserva e Aluguel de Quartos:

- ✚ Os hóspedes devem poder reservar quartos online antes de chegarem ao hotel. Além disso, ao chegarem, eles podem alugar um quarto disponível na recepção.

3. Serviços Adicionais:

- ✚ O hotel oferece serviços como restaurante, lavanderia e outros que os hóspedes podem solicitar durante sua estadia. Esses serviços devem ser registrados no sistema.

4. Consumo do Frigobar:

- ✚ Cada quarto tem um frigobar com bebidas e lanches. O sistema deve registrar o que o hóspede consome para que seja cobrado na conta final.

5. Gerenciamento de Diárias:

- ✚ As diárias dos quartos vencem ao meio-dia. O sistema deve calcular as diárias e adicionar o custo à conta do hóspede semanalmente.

6. Pagamento de Contas:

- ✚ Os hóspedes devem pagar suas contas, que incluem diárias, serviços e consumo do frigobar, antes de encerrar a estadia ou semanalmente.

7. Encerramento de Estadia:

- ✚ Quando um hóspede decide ir embora, ele deve quitar qualquer valor pendente. O sistema deve facilitar esse processo para que tudo esteja em ordem antes do check-out.

8. Registro de Funcionários:

- ✚ É importante saber qual funcionário fez o check-in ou check-out de um hóspede para manter um registro preciso das operações.

Desafio:

Desenvolva a interface do sistema utilizando HTML e CSS para estruturar e estilizar as páginas. Utilize JavaScript para implementar a lógica do sistema, incluindo o gerenciamento de dados e interações com o usuário. Considere a criação de um banco de dados simulado usando arrays ou objetos JavaScript para armazenar informações sobre hóspedes, reversas e aluguéis de quarto, serviços adicionais, consumo do frigobar, gerenciamento de diárias, pagamentos de contas, encerramento de estadia, registro de funcionários

Perguntas para gerar reflexões para o desenvolvimento do Projeto de Controle de Hotelaria.

1. Quais informações são essenciais para registrar um hóspede no sistema?
2. Como o sistema deve lidar com hóspedes que já se registraram anteriormente?
3. Como os hóspedes poderão fazer reservas de quartos? O sistema deve permitir reservas online?
4. O que acontece se um hóspede quiser alugar um quarto diretamente na recepção?
5. Que tipos de serviços adicionais o hotel oferece e como eles devem ser registrados no sistema?
6. Como o sistema deve atualizar a conta do hóspede quando um serviço adicional é utilizado?
7. Como o sistema deve registrar o consumo de itens do frigobar?
8. De que maneira o consumo deve ser adicionado à conta final do hóspede?
9. Como o sistema deve calcular as diárias dos quartos?

10. O que deve acontecer se um hóspede ficar além do horário de check-out?
11. Como o sistema deve gerar a fatura para o hóspede?
12. Quais métodos de pagamento o sistema deve suportar?
13. Que informações são necessárias para realizar o check-out de um hóspede?
14. Como o sistema deve garantir que todas as cobranças foram quitadas antes do hóspede sair?
15. Por que é importante registrar qual funcionário fez o check-in ou check-out?
16. Como o sistema deve armazenar essas informações para futuras referências?
17. Como garantir que o sistema seja fácil de usar tanto para os hóspedes quanto para os funcionários?
18. Que tipo de interface o sistema deve ter para facilitar o uso?

Desenvolvimento e Entrega: Sistema de Gerenciamento de Hotelaria

Objetivo: Desenvolver um sistema de gerenciamento para um hotel, abrangendo as funcionalidades de cadastro de hóspedes, reservas e serviços. O projeto será dividido em duas partes: a primeira focada na apresentação, documentação e uma estrutura de código desenvolvido em JavaScript utilizando os recursos aprendidos até o momento presente, e a segunda na implementação de melhorias e conceitos de orientação a objetos. Também será levado em consideração o diferencial inserido no projeto resultado das pesquisas realizadas pelo quarteto.

Parte 1: Apresentação e Documentação

Data de Entrega (documentação, apresentação e código no Moodle): 31/10/2024.

Data da Apresentação: 07/11/2024 (quinta-feira – sorteio no horário de aula)

Requisitos do Sistema

1. Estrutura do Sistema:

- + Desenvolver uma interface de usuário com seções distintas para cada tipo de cadastro: hóspedes, quartos, reservas e serviços.
- + Implementar um menu de navegação para facilitar o acesso às diferentes seções do sistema.

2. Formulários de Cadastro:

- + Hóspedes: Incluir campos para nome, documento de identidade, endereço e contato.
 - + Quartos: Incluir campos para número, tipo e preço por noite.
 - + Reservas: Incluir campos para data de check-in, data de check-out e hóspede.
 - + Serviços: Incluir campos para tipo de serviço e preço.
- Observação:** se a equipe resolver que existe a necessidade de mais campos, podem inserir.

3. Funcionalidades de Reserva:

- + Desenvolver uma interface para realizar reservas, onde o usuário possa selecionar o quarto e as datas de estadia.
- + Implementar a funcionalidade de cálculo do valor total da estadia, considerando o preço do quarto e os dias reservados.

4. Documentação:

- ✚ Elaborar um documento detalhando a estrutura do sistema, funcionalidades implementadas e possíveis melhorias.
- ✚ Incluir diagramas, como diagramas de fluxo ou wireframes, para ilustrar a interface e o funcionamento do sistema.

Parte 2: Código Aprimorado e Orientação a Objetos

Data de Entrega (documentação, apresentação e código no Moodle): 30/11/2024.

Data da Apresentação: 05/12/2024 (quinta-feira – sorteio no horário de aula)

Requisitos do Sistema

1. Melhorias no Código:

- ✚ Refatorar o código para melhorar a organização e a legibilidade.
- ✚ Implementar validações de dados nos formulários.

2. Conceitos de Orientação a Objetos:

- ✚ Utilizar classes e objetos para representar hóspedes, quartos, reservas e serviços.
- ✚ Implementar encapsulamento, herança e polimorfismo onde aplicável.

3. Funcionalidades Adicionais:

- ✚ Permitir edição e exclusão dos cadastros listados.
- ✚ Implementar um histórico de reservas para consulta posterior.

4. Documentação Atualizada:

- ✚ Atualizar a documentação do projeto para refletir as mudanças e melhorias implementadas.
- ✚ Incluir exemplos de código e explicações sobre o uso de orientação a objetos no projeto.

Apresentação Inicial do Sistema de Gerenciamento de Hotelaria em JavaScript

Estrutura da Apresentação

1. Introdução:

- ✚ Breve descrição do projeto e seus objetivos.
- ✚ Explicação sobre a importância de cada funcionalidade (cadastro e gerência de reservas).

2. Demonstração do Código:

Estrutura Básica:

- ✚ Apresentar a organização do código, incluindo a divisão em funções e objetos, se aplicável.
- ✚ Explicar a escolha de bibliotecas ou frameworks, caso sejam utilizados.

Funcionalidades Implementadas:

Demonstrar o código para o cadastro de uma entidade, como hóspedes ou quartos.

Por exemplo:

```
let quartos = [];  
let quartoid = 1;  
document.getElementById('form-quartos').addEventListener('submit', function(event) { event.preventDefault();  
const numero = document.getElementById('numeroQuarto').value; const tipo =  
document.getElementById('tipoQuarto').value;  
const preco = document.getElementById('precoQuarto').value; });
```

Observação: o código acima é apenas ilustrativo, a equipe deverá inserir o código desenvolvido na sua aplicação

3. Explicação da Lógica:

- ✚ Discutir como as funcionalidades básicas se conectam e como elas serão expandidas na versão final.
- ✚ Descrever a lógica por trás da implementação, como a escolha de atributos e métodos nas classes, por exemplo.

4. Próximos Passos:

- ✚ Explicar as melhorias planejadas para a entrega final, como adição de novas funcionalidades.
- ✚ Discutir como os conceitos de orientação a objetos serão aplicados na refatoração do código.

5. Conclusão:

- ✚ Resumo dos pontos principais apresentados.
- ✚ Abertura para perguntas e feedback.

Artigos e Sites Interessantes

Refatoração

1. REFACTORING GURU. *Refactoring Guru*. Disponível em: <https://refactoring.guru>. Acesso em: 6 out. 2024.
2. FOWLER, Martin. *Martin Fowler's Blog*. Disponível em: <https://martinfowler.com>. Acesso em: 6 out. 2024.

Orientação a Objetos

1. IBM DEVELOPER. *Object-Oriented Design Articles*. Disponível em: <https://developer.ibm.com/technologies/java/articles/j-intro/>. Acesso em: 6 out. 2024.
2. GEEKSFORGEES. *Object-Oriented Programming (OOP) Concept in Java*. Disponível em: <https://www.geeksforgeeks.org/object-oriented-programming-oops-concept-in-java/>. Acesso em: 6 out. 2024.

Modelos de Wireframes e Design de Interfaces

1. SMASHING MAGAZINE. *Smashing Magazine*. Disponível em: <https://www.smashingmagazine.com>. Acesso em: 6 out. 2024.
2. UX DESIGN. *UX Design*. Disponível em: <https://uxdesign.cc>. Acesso em: 6 out. 2024.

3. NIELSEN NORMAN GROUP. *Nielsen Norman Group*. Disponível em: <https://www.nngroup.com>. Acesso em: 6 out. 2024.
4. MIRO. *Wireframes*. Disponível em: <https://miro.com/pt/wireframe/o-que-e-wireframe/> . Acesso em: 5 out. 2024.

Considerações Finais

1. A documentação entregue referente a parte escrita deve seguir as normas ABNT.
2. Construa a apresentação de forma sintética. Evite excesso de conteúdo; inclua apenas o necessário.
3. A entrega no Moodle ocorrerá da seguinte forma: será aberto um questionário onde devem ser entregues na Questão 1 a apresentação do trabalho, na Questão 2 a documentação do projeto seguindo as normas ABNT, e na Questão 3 o projeto zipado.
4. Tanto a apresentação quanto a documentação devem ser entregues no formato PDF.
5. Ao salvar o arquivo em PDF, inclua o nome do "quarteto" seguido da data de entrega.
6. A data de entrega já está agendada, portanto, fiquem atentos. Caso a equipe não consiga entregar o projeto dentro do prazo, haverá uma semana adicional para entrega, com um decréscimo de 20% na nota. *Observação: após as datas de entrega, os projetos não serão aceitos.*
7. A entrega do projeto será feita pelo Moodle e deverá ser realizada por apenas um aluno da equipe. Os nomes dos integrantes da equipe devem estar identificados nos documentos.
8. Conforme avisado em sala de aula, serão verificados na solução: nomes de variáveis e funções dentro do padrão exigido pelo mercado e conforme exemplificado em sala de aula, indentação de código, arquivos separados (HTML, CSS, JavaScript) e outras boas práticas de programação.
9. Não se esqueçam de incluir as bibliografias utilizadas após a conclusão.
10. Lembre-se de demonstrar a lógica utilizada no desenvolvimento do projeto.

Bons Estudos!!!