

Hotel Descanso

Rafael Franco e Vitor Veiga

Objetivo:

Desenvolver um software para o gerenciamento de um hotel.

Requisitos Funcionais:

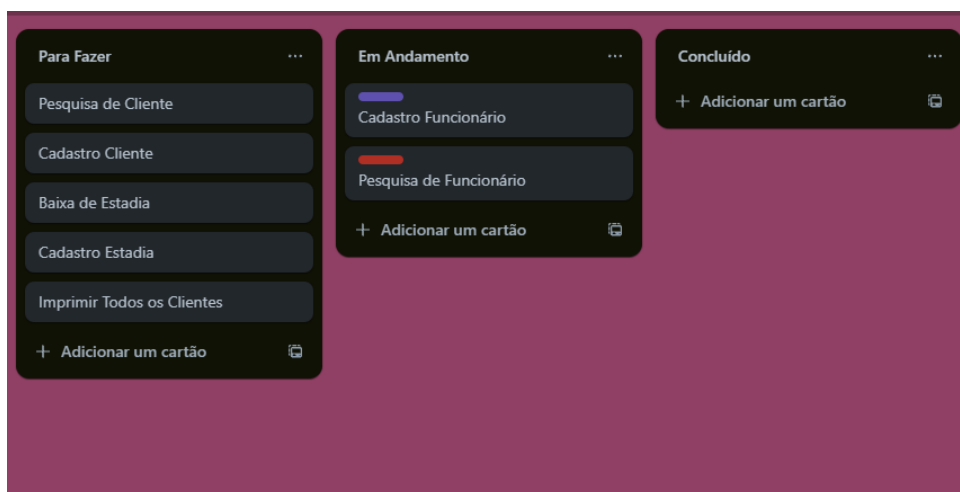
- Cadastro Cliente
- Cadastro Funcionário
- Cadastro Estadia
- Baixa de Estadia
- Pesquisa de Cliente
- Pesquisa de Funcionário
- Imprime Todos Clientes

Backlog do produto ao longo do desenvolvimento:

Sprint 1: A figura 1 apresenta o quadro kanban criado no “Trello”, as tarefas listadas e a divisão delas entre os membros do grupo na primeira sprint.

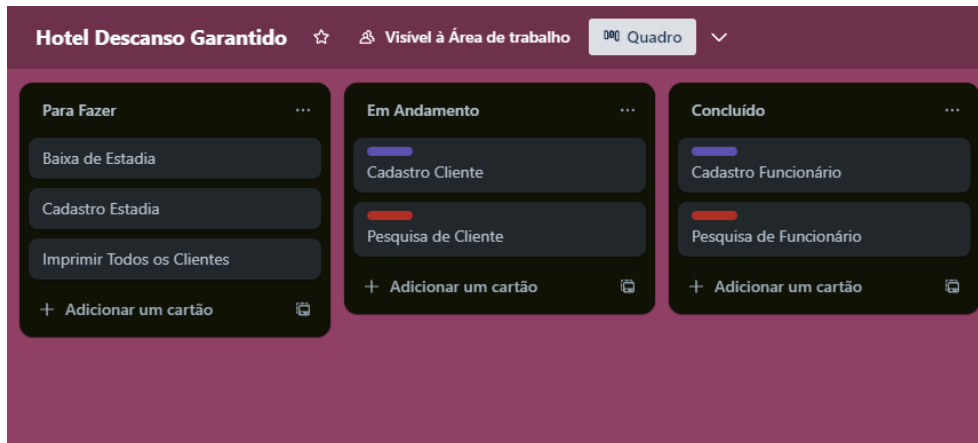
Roxo – Rafael Franco

Vermelho – Vitor Veiga



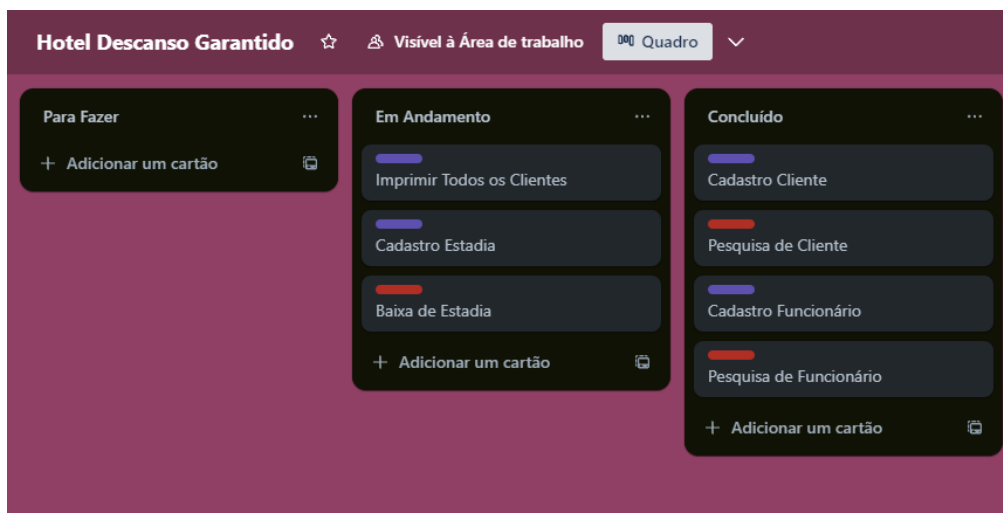
Fonte: Elaborado pelo autor

Sprint 2: A figura 2 apresenta a evolução do desenvolvimento do software, com as funções da sprint 1 concluídas e as novas tarefas designadas ao time na sprint 2.



Fonte: Elaborado pelo autor

Sprint 3: A figura 3 apresenta o último estágio de desenvolvimento do projeto, em que as últimas funções são implementadas.



Fonte: Elaborado pelo autor

Entrega: A figura 4 apresenta a entrega final do software com as principais funções implementadas.



Fonte: Elaborado pelo autor

Lista de Assinatura das funções e parâmetros:

- **Cadastro Clientes:** “void cadastroCliente(struct Cliente clientes[], int *contadorClientes, int numMaxClientes)”

A função cadastra as informações dos clientes e armazena em um documento externo, não possui retorno.

- **Cadastro de Funcionário:** “void cadastroFuncionario(struct Funcionario funcionarios[], int *contadorFuncionarios, int numMaxFuncionarios)”

A função cadastra as informações dos funcionários e armazena em um documento externo, não possui retorno.

- **Cadastro de Estadia:** “void cadastroEstadia(struct Estadia estadias[], int *contadorEstadias, int numMaxEstadias)”

A função cadastra a estadia de um cliente no hotel e salva em um arquivo externo, não possui retorno.

- **Baixa de Estadia:** “void darBaixaEstadia(struct Estadia estadias[], int *contadorEstadias)”

A função calcula o período de estadia do cliente e retorna o valor a ser cobrado.

- **Pesquisa de Cliente:** “void pesquisarClientes(struct Cliente clientes[], int contadorClientes)”

A função recebe o nome do cliente, pesquisa ele no sistema e fornece todas as informações cadastradas sobre ele.

- **Pesquisa de Estadia:** “void pesquisarEstadia(struct Estadia estadias[], int contadorEstadias)”

A função recebe o número de um quarto e retorna ID do cliente, data de entrada, data de saída, número de diárias e o número de hóspedes.

- **Imprimir Todos os Clientes:** “void listarClientes(struct Cliente clientes[], int contadorClientes)”

A função imprime todos os clientes cadastrados no sistema.

Casos de Teste do Software:

Teste 1 – Menu

Entradas Válidas: Números inteiros	Classes Válidas: 0 a 7	Resultado Esperado: Entra na função correspondente	Classes Inválidas: Superior a 7 Inferior a 0	Resultado Esperado: Reiniciar menu de opções
---	------------------------------	---	---	--

Entradas	Resultado	Aprovado?
Valor – 1	Dispara função de cadastro de cliente	Sim
Valor – 2	Dispara função de cadastro de funcionário	Sim
Valor – 3	Dispara função de cadastro de estadia	Sim
Valor – 4	Dispara função baixa de estadia	Sim
Valor – 5	Dispara função para pesquisar cliente	Sim
Valor – 6	Dispara função para pesquisar estadia	Sim
Valor – 7	Dispara função de imprimir todos os clientes	Sim
Valor – 0	Encerra o programa	Sim

Teste 2 – Função cadastro de cliente

Entradas Válidas: Nome Endereço Telefone	Classes Válidas: String Int	Resultado Esperado: Cliente cadastrado, reiniciação do menu	Classes Inválidas: Float Double Bool	Resultado Esperado: Valor inválido, reiniciação menu
---	-----------------------------------	--	---	---

Entradas	Resultado	Aprovado?
Nome Cliente	Dado adicionado ao arquivo clientes.txt	Sim
Endereço Cliente	Dado adicionado ao arquivo clientes.txt	Sim
Telefone Cliente	Dado adicionado ao arquivo clientes.txt	Sim
Nome: Augusto Lopes Endereço: Rua mulungu 133 Telefone: 31 912345678	Cliente cadastrado com sucesso.	sim

Teste 3 – Função cadastro de funcionário

Entradas Válidas: Nome Telefone Cargo Salário	Classes Válidas: String Int Float	Resultado Esperado: Funcionário cadastrado, reiniciação do menu	Classes Inválidas: Bool Double	Resultado Esperado: Valor inválido, reiniciação menu
---	--	--	--------------------------------------	---

Entradas	Resultado	Aprovado?
Nome Funcionário	Dado adicionado ao arquivo funcionários.txt	Sim
Telefone Funcionário	Dado adicionado ao arquivo funcionários.txt	sim
Cargo Funcionário	Dado adicionado ao arquivo funcionários.txt	Sim
Salário Funcionário	Dado adicionado ao arquivo funcionários.txt	Sim

Teste 4 – Função cadastro estadia

Entradas Válidas: Data de entrada Data de saída ID do cliente Número do quarto Número de hóspedes no quarto	Classes Válidas: String Int	Resultado Esperado: Estadia cadastrada, reiniciação do menu	Classes Inválidas: Float Double Bool	Resultado Esperado: Valor inválido, reiniciação menu
---	-----------------------------------	--	---	---

Entradas	Resultado	Aprovado?
Data de entrada (yyyy/mm/dd)	Dado adicionado ao arquivo estadias.txt	Sim
Data de saída (yyyy/mm/dd)	Dado adicionado ao arquivo estadias.txt	Sim
ID cliente	Dado adicionado ao arquivo estadias.txt	Sim
Número quarto	Dado adicionado ao arquivo estadias.txt	Sim
Quantidade de hóspedes	Dado adicionado ao arquivo estadias.txt	Sim
Entrada: 20/06/2024 Saída: 01/07/2024 ID cliente: 01 Número Quarto: 02 Quantidade hóspedes: 4	Estadia cadastrada com sucesso. Informações adicionadas em estadias.txt	Sim

Teste 5 – Baixa de estadia

Entradas Válidas: Número do Quarto	Classes Válidas: String Int	Resultado Esperado: Valor a ser cobrado ao hóspede específico	Classes Inválidas: Float Double Bool	Resultado Esperado: Valor inválido, reinicialização menu
--	---------------------------------------	--	---	---

Entradas	Resultado	Aprovado?
ID cliente	Preço da hospedagem	Sim
ID cliente: 01	245.00	Sim

Teste 6 – Pesquisa de Cliente

Entradas Válidas:	Classes Válidas:	Resultado Esperado:	Classes Inválidas:	Resultado Esperado:
Nome cliente	String	Informações do cliente filtrado	Int Float Double Bool	Valor inválido, reinicialização menu

Entradas	Resultado	Aprovado?
Nome Cliente	Informações cadastradas filtradas	Sim
Augusto Lopes	Nome: Augusto Lopes Endereço: Rua mulungu 133 Telefone: 31 912345678	Sim

Teste 7 – Imprimir Todos os Clientes

Entradas Válidas:	Classes Válidas:	Resultado Esperado:	Classes Inválidas:	Resultado Esperado:
Número 7 no menu de opções	Int	Uma listagem de todos os clientes cadastrados e suas informações	Valores diferentes de 7	<p>- Números inferiores a 0 e superiores a 7: Valor inválido, reinicialização do menu</p> <p>- Valores de 0 a 6: outra funcionalidade do sistema</p>

Entradas	Resultado	Aprovado?
Número 7	Listagem de todos os clientes	Sim