

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

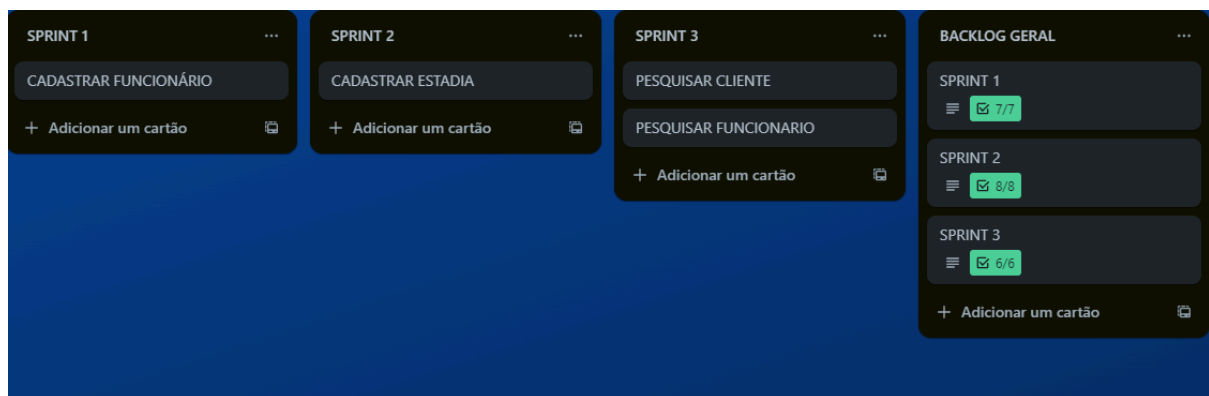
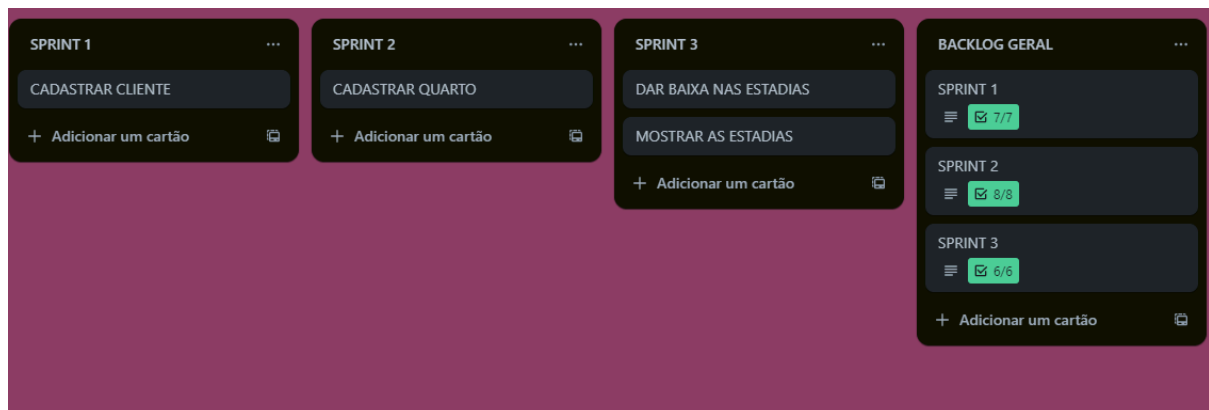
Matheus Santos Dias Gomes e Victor Sverberi Maia
Sistema de Gestão de Hotel - Hotel Descanso Garantido

Apresentação:

Trabalho realizado pelas matérias de Algoritmos e Estruturas de Dados I e Fundamentos de Engenharia de Software, lecionado pelos professores Laerte e João Paulo Aramuni, no qual foi proposto a criação de um sistema na linguagem C com o intuito de melhorar a gestão de um hotel, chamado “Hotel Descanso Garantido”, para isso foi disponibilizado um prazo 15 dias para realizar todo o código, nesse sistema criado será possível realizar o cadastro de clientes, funcionários, quartos e estadias, além de realizar consultas e baixar estadias.

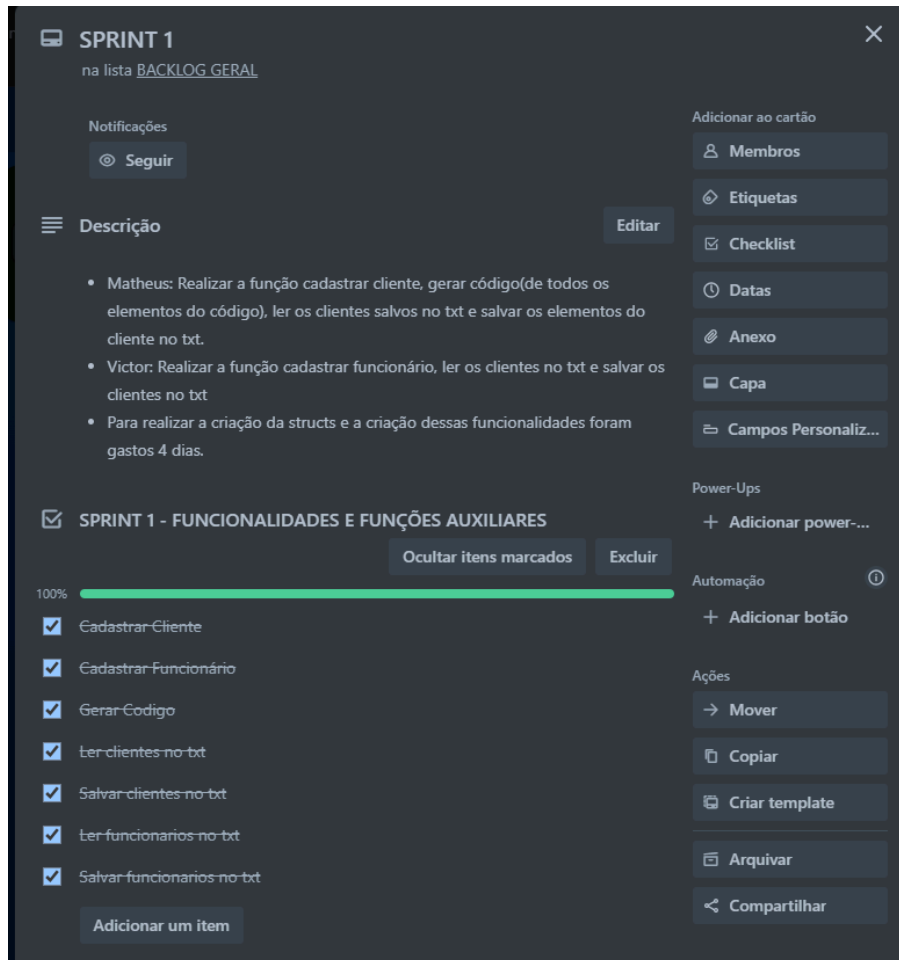
Backlog do Produto:

Tela do Matheus e Victor respectivamente:



Nelas demonstram o que foi feito em cada sprint e ao abrir o backlog geral de cada sprint mostrará todas as funcionalidades e funções utilizadas no projeto. Além disso, retrata o tempo gasto em cada sprint.

SPRINT 1- BACKLOG GERAL



Com base nessa imagem é possível notar que todas as tarefas já foram realizadas em seu devido prazo, pois o código já está pronto, nessa sprint, foram feitas as funcionalidades de realizar cadastro de funcionário e clientes e além disso criado as funções auxiliares para a realização do projeto, portanto, foram gastos 4 dias.

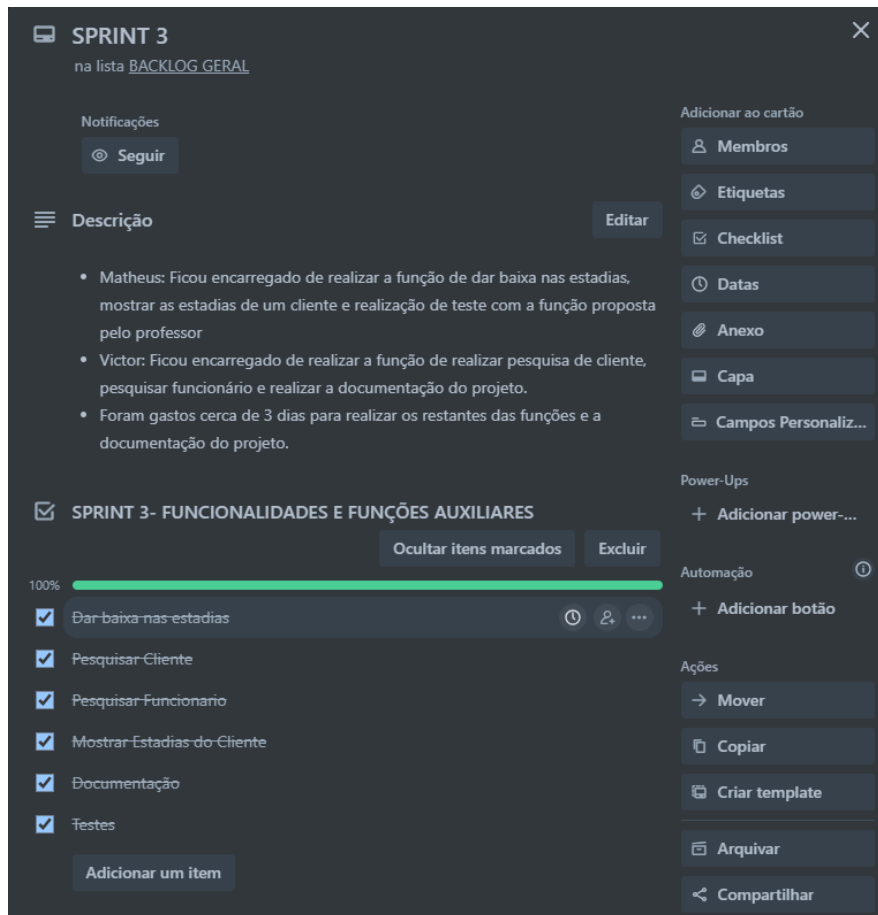
SPRINT 2- BACKLOG GERAL

The screenshot shows a 'SPRINT 2' backlog card within a 'BACKLOG GERAL' list. The card has a dark theme and includes several sections:

- Notificações:** A 'Seguir' button with a bell icon.
- Descrição:** A section with an 'Editar' button and a list of bullet points:
 - Matheus: Ficou encarregado com a criação da função para realizar cadastro de quarto, ler quartos no txt, salvar quartos no txt e resetar status dos quartos
 - Victor: Ficou encarregado com a criação da função para realizar cadastro de estadias, calcular diárias, ler estadias no txt e salvar estadias no txt.
 - Foram gasto cerca de 8 dias nessa sprint.
- SPRINT 2- FUNCIONALIDADES E FUNÇÕES AUXILIARES:** A section with a progress bar at 100% and a list of tasks, each with a checkbox:
 - ☒ Cadastrar-Quarto
 - ☒ Cadastrar-Estadia
 - ☒ Calcular-Diarias
 - ☒ ler-quartos-no-txt (highlighted with a tooltip showing a clock, user, and menu icon)
 - ☒ Salvar-quartos-no-txt
 - ☒ ler-estadias-no-txt
 - ☒ Salvar-estadias-no-txt
 - ☒ Resetar-status-dos-quartosBelow the list is an 'Adicionar um item' button.
- Adicionar ao cartão:** A sidebar with buttons for 'Membros', 'Etiquetas', 'Checklist', 'Datas', 'Anexo', 'Capa', and 'Campos Personaliz...'.
- Power-Ups:** A section with a '+ Adicionar power-...' button.
- Automação:** A section with a '+ Adicionar botão' button and an information icon.
- Ações:** A sidebar with buttons for 'Mover', 'Copiar', 'Criar template', 'Arquivar', and 'Compartilhar'.

A sprint 2 foram realizadas outras duas funcionalidades, que são a de cadastrar quartos e estadias, ademais, foram implementadas as diversas funções necessárias para manter o projeto seguindo as devidas restrições propostas. Desse modo, foram gastos cerca de 8 dias, além da escrita do código teve a lógica para realizar o cadastro das estadias.

SPRINT 3- BACKLOG GERAL



Na sprint 3 foram realizadas as últimas funcionalidades do sistema que são a de dar baixa nas estadias, pesquisar clientes, pesquisar funcionários e mostrar as estadias de um cliente, além disso foi realizado os testes e toda documentação. Para realizar essa parte foram necessários cerca de 3 dias e sendo possível realizar a entrega do projeto com tudo devidamente elaborado.

OBS: A lista de assinaturas das funções e parâmetros está no outro pdf do próprio github.

CASOS DE TESTE

- Foi realizado na função de calcular diárias, no qual, ela verifica se a data é verdadeira e depois calcula a quantidade de diárias, e no caso de todas os casos verdadeiros ela imprime a mensagem “Todos os testes passaram com sucesso!”. (OBS: O código está postado no Github na pasta testes);
- Outra função que foi testada foi a função de resetar os status dos quartos, no qual, ele verifica se a função está realmente deixando os quartos

desocupados, e caso esteja funcional ela imprime "Todos os testes passaram!" (OBS: O código está postado no Github na pasta testes);

- Outra função que foi testada foi a função de cadastrar clientes, no qual, ele verifica se a função está realmente cadastrando os clientes, e caso esteja funcional ela imprime "Todos os testes passaram!" (OBS: O código está postado no Github na pasta testes);

RELATÓRIO DA EXECUÇÃO DOS TESTES

1. CALCULAR DIÁRIA

- Parâmetros: data de entrada e data de saída
- Não Possui nenhuma entrada de informação
- Verifica se a data digitada é válida, ou seja, verifica se o ano e o mês de saída são maiores do que o de entrada, pois não é possível realizar estadias no passado, portanto, caso isso seja verdadeira ele dá sequência, no caso do ano se o de saída for maior do que o de entrada ele calcula a quantidade de dias em baseado na diferença do ano de saída e ano de entrada, logo após, é somado com a diferença do mês de saída com o mês de entrada e por fim soma com a diferença da data de dia de saída e dia de entrada, sendo retornado o quantidade total de dias.
- No caso de teste para verificar diversos testes como (Mesma data, diferentes dias no mesmo mês, diferentes meses no mesmo ano, diferentes anos, diferentes dias, meses e anos), e no caso de todos os testes sendo verdadeiros é imprimido uma mensagem de que os testes foram um sucesso.

2. RESETAR STATUS DOS QUARTOS

- Parâmetros: Nenhum
- Não Possui nenhuma entrada de informação
- A função abre o txt de quartos e lê todas as informações dos quartos já cadastrados, logo o txt é fechado e aberto sequencialmente para escrita, sendo imprimido para todos os quartos o status como desocupado, e por fim imprime uma mensagem relatando que todos os quartos estão resetados para resetados.
- A função de teste imprime 2 exemplos de quartos no txt e logo após o arquivo é fechado e é chamada a função de resetar status dos quartos, então é reaberto para a leitura, é lido todas as informações do txt e é usada a função strcmp, que compara o status lido no txt com a palavra desocupado e no caso de todos os status estarem de acordo o programa imprime a mensagem de que todos passaram no teste.

3. CADASTRAR CLIENTE

- Parâmetros: Nenhum
- Tem como entrada as informações de nome, endereço e telefone do cliente
- A função serve para cadastrar um novo cliente no sistema, onde é gerado um cliente automático que é inicializado em 1, após é lido o nome, endereço e telefone que serão digitados pelo o usuário, em todas elas é esperado um string, e por fim é chamado uma função de salvar cliente que serve para salvar as informações do cliente no txt.
- A função de teste abre o arquivo clientes.txt e imprime algumas supostas informações para o cliente, logo após o arquivo é fechado e é chamada a função de cadastrar cliente, então o arquivo é reaberto para modo de leitura e com a função assert ele compara todas as informações já gravadas no txt estão de acordo e caso sejam verdadeiras o programa imprime que todos os testes passaram sendo possível verificar se a função está cadastrado corretamente.