

PONTIFÍCIA UNIVERSIDADE CATÓLICA DE MINAS GERAIS
INSTITUTO DE CIÊNCIAS EXATAS E INFORMÁTICA
UNIDADE EDUCACIONAL PRAÇA DA LIBERDADE
Bacharelado em Engenharia de Software

Trabalho interdisciplinar: Fundamentos de Engenharia de Software e Algoritmos e Estruturas de Dados I

Nome dos integrantes do grupo:

Ana Corina Damas Batista

Davi Gonçalves Brandão

Milena Lara Reis Ferreira

Nome do sistema:

Viver Bem

Apresentação

O sistema desenvolvido tem como objetivo atender à Clínica Viver Bem, que oferece atendimento nas especialidades de cardiologia, dermatologia e clínica médica, melhorando a sua forma de organizar registros de médicos, pacientes e consultas, que até então eram feitos à mão ou em tabelas do excel. Trata-se de uma aplicação de console feita utilizando-se a linguagem C.

Backlog do produto

A figura 1 mostra a organização inicial do projeto. No aplicativo Trello, definimos o backlog do projeto e as tarefas da primeira sprint já foram alocadas na coluna “A fazer”, recebendo assim maiores especificações: checklist de funcionalidades e restrições e comentários dentro do card (figura 2). Utilizamos um mesmo board para toda a equipe, uma vez que o Trello permite atribuir membros a cada card.




Figura 1 - Backlog do projeto: início

criar função CADASTRAR PACIENTE

na lista [A Fazer](#)

Membros



+

Etiquetas

Prioridade alta

Implementação

+

Descrição

Adicione uma descrição mais detalhada...

☒ Propriedades da entidade

Excluir

0%

☐ - código

☐ - nome

☐ - endereço

☐ - telefone

☐ - data de nascimento

Adicionar um item

☒ Restrições

Ocultar itens marcados

Excluir

33%

☐ Garantir que não haverá mais de um paciente com o mesmo código

☐ Se quiser pode gerar o código do paciente automaticamente

Figura 2 - Detalhamento de um card de tarefa

A figura 3 mostra o status do backlog ao final da primeira sprint. Concluídas as funcionalidades, elas são revisadas por colegas e em seguida alocadas na fileira de testes, que optamos por realizar mais ao final.

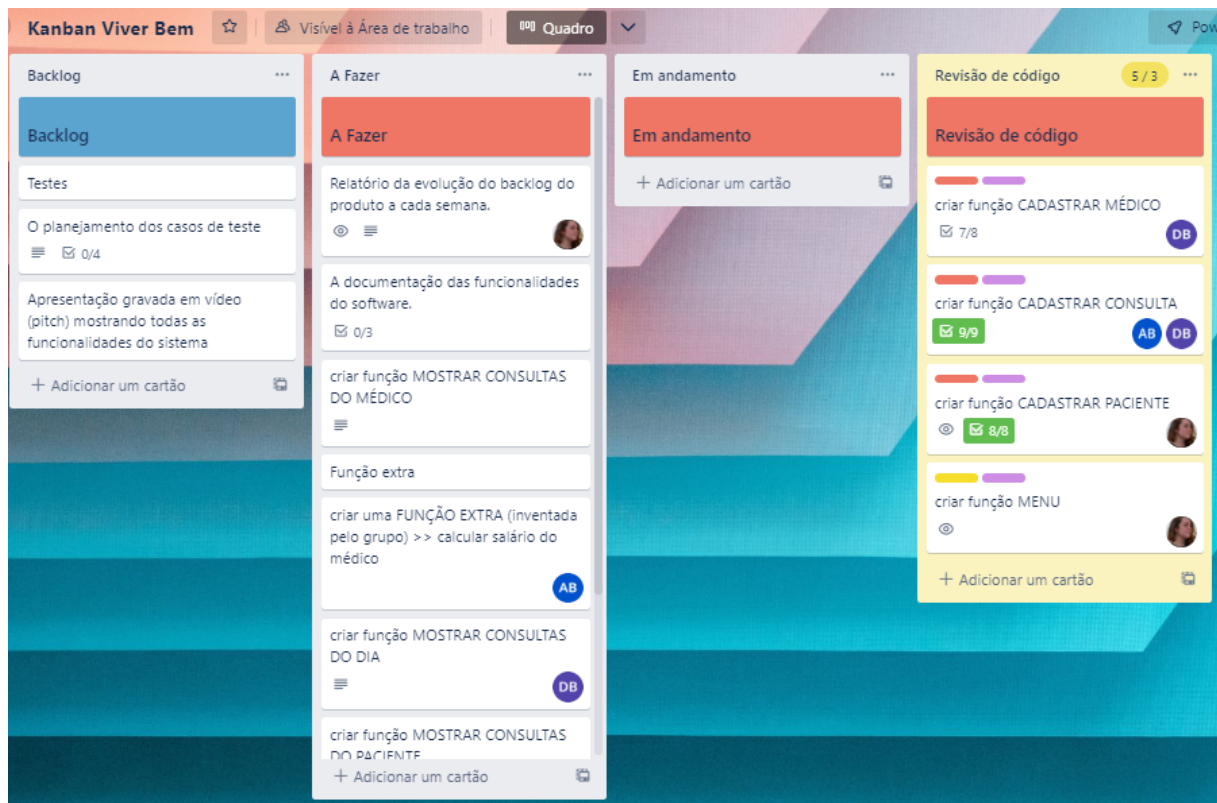


Figura 3 - Final da Sprint 1

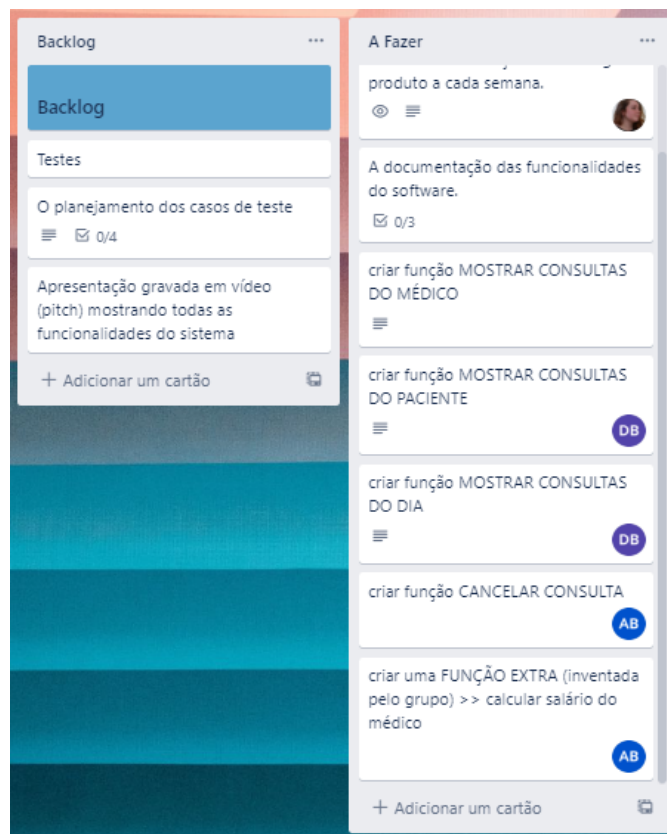


Figura 4 - Início da Sprint 2

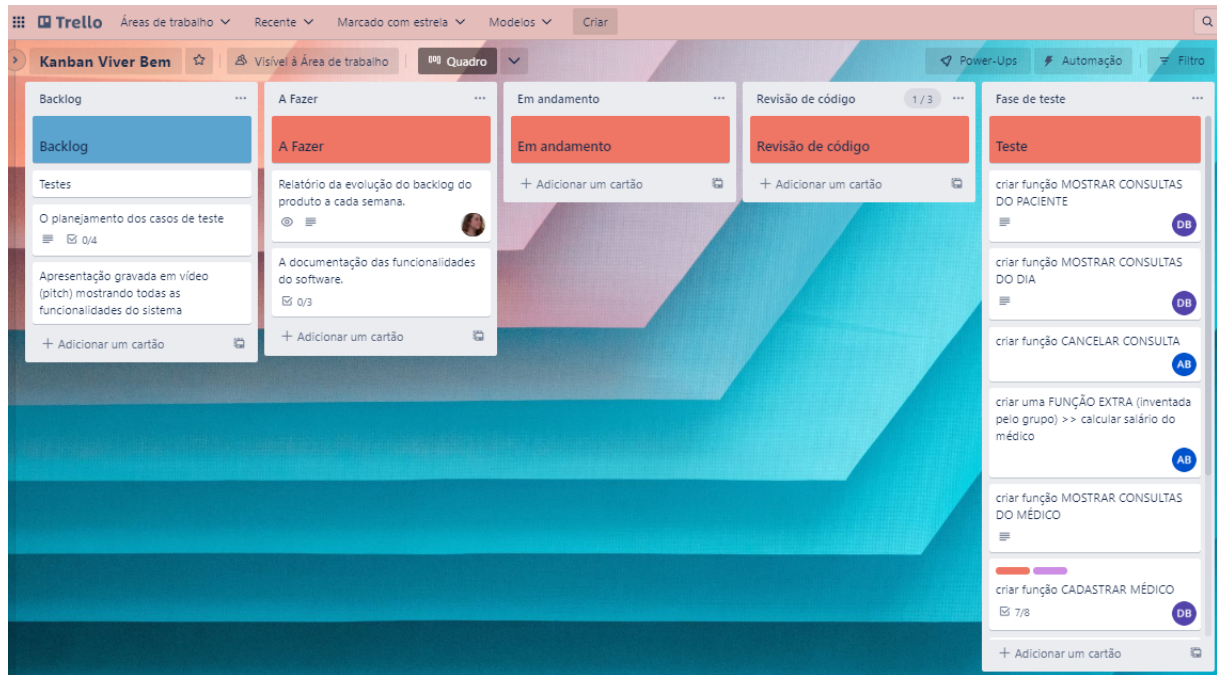


Figura 5 - Final da Sprint 2

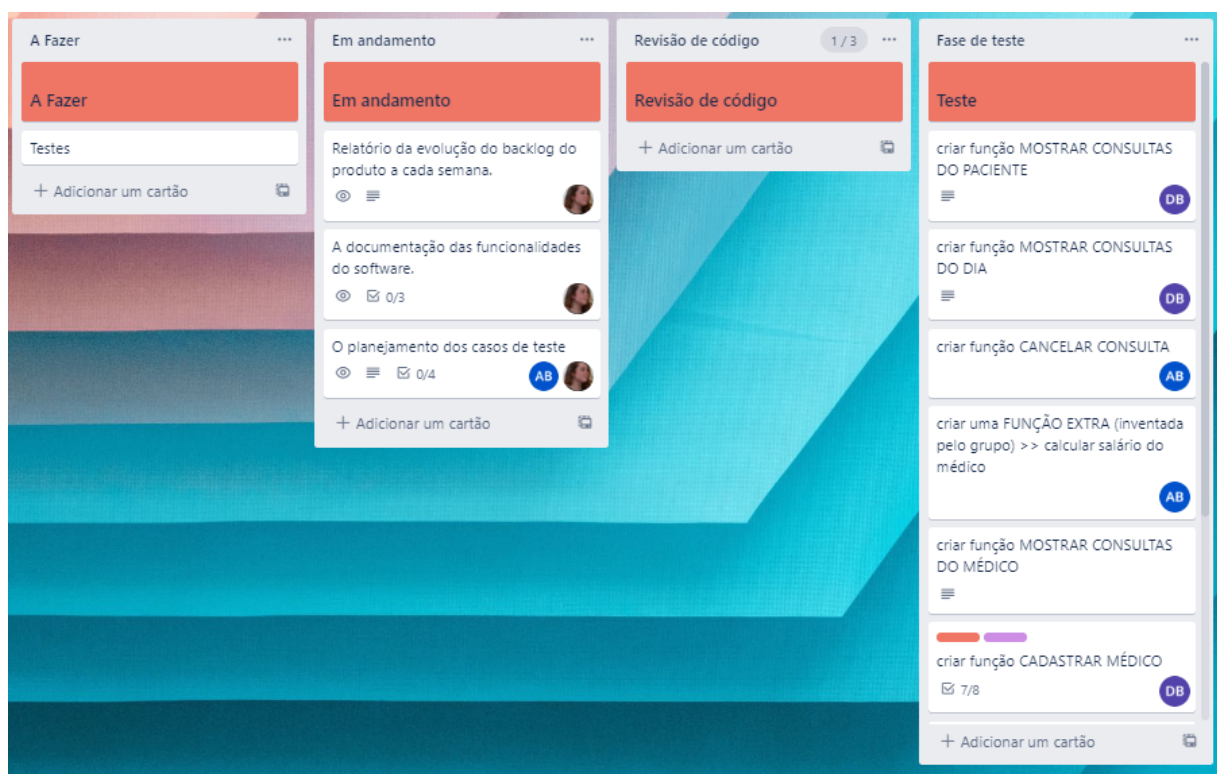


Figura 6 - Início da Sprint 3

Lista de assinaturas das funções e parâmetros

1. `void menuPrincipal();`

Procedimento que exibe no console o menu principal, que oferece opções de executar subprogramas, exibir menu de subprogramas ou encerrar o programa.

2. **void menuProgramas();**

Procedimento que exibe no console o menu de subprogramas para cadastrar, exibir, alterar ou excluir registros de pacientes, médicos e consultas, entre outros.

3. **int localizaPaciente(FILE *fPacientes, int codigoPaciente);**

Função que verifica se existe paciente com o código informado nos registros, recebendo como parâmetros um ponteiro para endereço do arquivo onde os pacientes estão registrados e o código do paciente.

4. **void cadastraPaciente(FILE *fPacientes);**

Procedimento que adiciona um paciente aos registros. Recebe como parâmetro um ponteiro para o endereço do arquivo onde os pacientes serão registrados.

5. **void alteraPaciente(FILE *fPacientes);**

Procedimento que modifica os dados de um paciente que está nos registros. Recebe como parâmetro um ponteiro para o endereço do arquivo onde os pacientes estão registrados. O código do paciente é passado via input do teclado.

6. **void imprimePacientes(FILE *fPacientes);**

Procedimento que imprime no console todos os registros de pacientes. Recebe como parâmetro um ponteiro para o endereço do arquivo onde os pacientes estão registrados.

7. **int localizaMedico(FILE *fMedicos, int codigoMedico);**

Função que verifica se existe médico com o código informado nos registros, recebendo como parâmetros um ponteiro para endereço do arquivo onde os médicos estão registrados e o código do médico.

8. **void cadastraMedico(FILE *fMedicos);**

Procedimento que acrescenta um novo registro de médico ao arquivo, que é passado como parâmetro por referência (ponteiro para o endereço do arquivo onde os registros estão salvos).

9. **void alteraMedico(FILE *fMedicos);**

Procedimento que modifica os dados de um médico que está nos registros. Recebe como parâmetro um ponteiro para o endereço do arquivo onde os médicos estão registrados. O código do médico é passado via input do teclado.

10. **void imprimeMedicos(FILE *fMedicos);**

Procedimento que imprime no console os registros de médicos, recebendo como parâmetro um ponteiro para o endereço do arquivo onde os registros estão salvos

11. **void calculaPagamentoMedico(FILE *fConsultas);**

Função que calcula quanto um médico deve receber a partir da quantidade de consultas realizadas. Recebe como parâmetros os ponteiros para os endereços na memória dos que contém os registros de consultas, médicos e de pagamentos.

12. `int localizaConsulta(FILE *fConsultas, int codigoConsulta);`

Função que verifica se existe consulta com o código informado nos registros, recebendo como parâmetros um ponteiro para endereço do arquivo onde as consultas estão registradas e o código da consulta.

13. `int localizaConsultasPorMedico(FILE *fConsultas, int codigoMedico, int dia, int mes, int ano);`

Função para buscar o registro das consultas de um médico por data, recebendo como parâmetros um ponteiro para endereço do arquivo onde as consultas estão registradas, o código do médico, o dia, o mês e o ano da data a ser exibida.

14. `int localizaQtdConsultasPorMedico(FILE *fConsultas, int codigoMedico, int dia, int mes, int ano);`

Função que retorna a quantidade de consultas de um médico, recebendo como parâmetros um ponteiro para endereço do arquivo onde as consultas estão registradas, o código do médico, o dia, o mês e o ano da data a ser exibida.

15. `void cadastraConsulta(FILE *fPacientes, FILE *fMedicos, FILE *fConsultas);`

Procedimento que adiciona uma consulta aos registros. Recebe como parâmetros os ponteiros para os endereços na memória dos arquivos que contém os registros dos médicos, dos pacientes e das consultas.

16. `void cancelaConsulta(FILE *fConsultas);`

Procedimento que exclui uma consulta dos registros. Recebe como parâmetro um ponteiro para o endereço na memória do arquivo que contém os registros de consultas. Recebe o código da consulta por input no teclado.

17. `void alteraConsulta(FILE *fConsultas);`

Procedimento que modifica os dados de uma consulta que está nos registros. Recebe como parâmetro um ponteiro para o endereço do arquivo onde as consultas estão registradas. O código do paciente é passado via input do teclado.

18. `void imprimeConsultas(FILE *fConsultas);`

Procedimento que imprime no console os registros de consultas, recebendo como parâmetro um ponteiro para o endereço do arquivo onde os registros estão salvos.

19. `void imprimeConsultasPorData(FILE *fPacientes, FILE *fMedicos, FILE *fConsultas);`

Procedimento que imprime no console as consultas de uma determinada data, recebida por input do teclado. Recebe como parâmetros os ponteiros para os endereços na memória dos arquivos que contém os registros dos médicos, dos pacientes e das consultas.

20. void imprimeConsultasPorMedico(FILE *fPacientes, FILE *fMedicos, FILE *fConsultas);

Procedimento que imprime as consultas de um médico, que pode ser localizado através do nome ou do código, recebidos por input do teclado. Recebe como parâmetros os ponteiros para os endereços na memória dos arquivos que contém os registros dos médicos, dos pacientes e das consultas.

21. void imprimeConsultasPorPaciente(FILE *fPacientes, FILE *fMedicos, FILE *fConsultas);

Procedimento que imprime as consultas de um paciente, que pode ser localizado através do nome ou do código, recebidos por input do teclado. Recebe como parâmetros os ponteiros para os endereços na memória dos arquivos que contém os registros dos médicos, dos pacientes e das consultas.

Testes

Casos de teste do software

Teste 1: Cadastrar paciente				
Entrada	Classes válidas	Resultado esperado	Classes inválidas	Resultado esperado
<ul style="list-style-type: none">- Nome: string com até 100 caracteres- Telefone: número de 10 ou 11 dígitos, sendo os dois primeiros dígitos o DDD sem 0- Endereço: string com até 200 caracteres- Data de Nascimento: string com 10 caracteres, incluindo números e caracter especial "/"	<ul style="list-style-type: none">- Nome válido. Ex: Maria dos Santos- Telefone: número de 10 ou 11 dígitos. Ex: 31995951234- Endereço válido com até 200 dígitos. Ex: Rua dos Otoni, 888, Santa Efigênia, Belo Horizonte- Data de nascimento válida, utilizando / para separação de dias, meses e anos. Ex: 11/01/2001	Paciente cadastrado com sucesso.	Nome, telefone, endereço ou data de nascimento escritos incorretamente, excedendo o limite de números/caracteres	Erro ao cadastrar o paciente, necessário informar telefone, número ou data de nascimento novamente.

Teste 2: Cadastrar médico				
Entrada	Classes válidas	Resultado esperado	Classes inválidas	Resultado esperado
<ul style="list-style-type: none">- Nome: string com até 200 caracteres.- Telefone: número de 10 ou 11 dígitos, sendo os dois primeiros dígitos o DDD sem 0- Especialidade: string com até 30 caracteres	<ul style="list-style-type: none">- Nome: string com até 100 caracteres. Ex: Maria dos Santos- Telefone: número de 10 ou 11 dígitos. Ex: 31995951234- Especialidade: string com até 30 caracteres	- Médico cadastrado com sucesso.	- Nome, telefone ou especialidade escritos incorretamente, excedendo o limite de números/caracteres	- Erro ao cadastrar o médico, necessário informar telefone, código, nome ou especialidade.

Teste 3: Cadastrar consulta				
Entrada	Classes válidas	Resultado esperado	Classes inválidas	Resultado esperado
<ul style="list-style-type: none"> - Código do médico: número de até 5 dígitos, começando com 2, dentro dos códigos de médico cadastrados. - Código do paciente: número de até 5 dígitos, começando com 1, dentro dos códigos de paciente cadastrados - Data: string com 10 caracteres no formato DD/MM/AAAA. O médico informado não deve ter mais de uma outra consulta no mesmo dia. - Horário: string com 5 caracteres no formato HH:MM. Um mesmo médico não pode ter consultas com intervalo de menos de 30 minutos. 	<ul style="list-style-type: none"> - O código do médico já foi cadastrado. - O código do paciente já foi cadastrado. - Data informada não possuía 2 consultas cadastradas para o médico informado. 	<ul style="list-style-type: none"> - Consulta cadastrada com sucesso. 	<ul style="list-style-type: none"> - Código do médico/paciente não cadastrado. - Data excedendo o máximo de 2 consultas por médico. - Intervalo com relação a consultas já cadastradas inferior a 30 minutos 	<ul style="list-style-type: none"> - Data excedendo limite: necessário selecionar outra data - Código do paciente/médico inexistentes, necessário informar novamente e.

Relatório de Execução de Testes

Teste 1: Cadastrar paciente				
Entrada	Classes válidas	Resultado esperado	Classes inválidas	Resultado esperado
<ul style="list-style-type: none"> - Nome - Telefone - Endereço - Data de nascimento 	<ul style="list-style-type: none"> - Nome: string de até 100 caracteres - Telefone: número de 10 ou 11 dígitos - Endereço: string de até 200 caracteres - Data de nascimento: string de 10 caracteres 	Mensagem impressa no console: "Paciente cadastrado com sucesso."	<ul style="list-style-type: none"> - Nome: string com mais de 100 caracteres - Telefone: número com mais de 11 dígitos ou caracteres não numéricos - Endereço: string com mais de 200 caracteres - Data de nascimento: string com mais de 10 caracteres 	Erro; programa não grava dados ou pede novos dados ao usuário.
Relatório de execução de testes				
Entradas		Resultado	Aprovado?	
Aline Muniz 31995950909 Rua dos Otoni, 1001, Santa Efigenia, Belo Horizonte 12/12/2000		Paciente cadastrado com sucesso.	Aprovado	
Eduardo Galeano 31998989898 Avenida Brasil, 777, Funcionarios, Belo Horizonte 27/07/1959		Paciente cadastrado com sucesso.	Aprovado	
Ulisses Machado 31998980000 Rua do Bosque, 123, Floresta, Belo Horizonte 25/11/1960		Paciente cadastrado com sucesso.	Aprovado	
Teresa Cristina Maria Josefa Gaspar Baltasar Melchior Januária Rosália Lúcia Francisca de Assis Isabel Francisca de Pádua Donata Bonosa Andréia de Avelino Rita Liutgarda Gertrude Venância Tadea Spiridione Roca Matilde de Bourbon-Duas Sicílias 31995950909 Rua Bernardo Guimarães, 1001, Santa Efigenia, Belo Horizonte 12/03/1999		Programa finaliza sem realizar registro.	Aprovado	
Miguel Maria do Patrocínio João Carlos Francisco de Assis Xavier de Paula Pedro de Alcântara Antônio Rafael Gabriel Joaquim José Gonzaga Evaristo (31)3388-9945 Avenida Bandeirantes, 444, Mangabeiras, Belo Horizonte		Erro: formato telefone incorreto. Digite novamente.	Aprovado	
Bruno Machado 31998980000 Rua do Bosque de um reino tão-tão-tão distante que ninguém jamais chegou já, onde Judas perdeu as botas, após as montanhas e montes, número 12345678979, Cidade no meio do nada, Reino Tão-tão-tão-distante 25/11/1960		Programa finaliza sem realizar registro.	Aprovado	
Bruno Machado 31998980000 Avenida Bandeirantes, 444, Mangabeiras, Belo Horizonte 25/11/1960		Programa finaliza sem realizar registro.	Aprovado	

Teste 2: Cadastrar médico				
Entrada	Classes válidas	Resultado esperado	Classes inválidas	Resultado esperado
- Nome - Telefone: - Especialidade:	- Nome: string com até 100 caracteres. - Telefone: número de 10 ou 11 dígitos, sendo os dois primeiros dígitos o DDD sem 0 - Especialidade: string com até 30 caracteres	Mensagem impressa no console: "Médico cadastrado com sucesso."	- Nome: string com mais de 100 caracteres - Telefone: número com mais de 11 dígitos ou caracteres não numéricos - Endereço: string com mais de 200 caracteres - Data de nascimento: string com mais de 10 caracteres	
Relatório de execução de testes				
Entradas		Resultado	Aprovado?	
Jurema Silva 3133339494 Cardiologista		Médico cadastrado com sucesso	Aprovado	
Drauzio Varela 31995964848 Dermatologista		Médico cadastrado com sucesso	Aprovado	
Bruno Mendes 31999991234 Clínico geral		Médico cadastrado com sucesso	Aprovado	
Teresa Cristina Maria Josefa Gaspar Baltasar Melchior Januária Rosalía Lúcia Francisca de Assis Isabel Francisca de Pádua Donata Bonosa Andréia de Avelino Rita Liutgarda Gertrude Venância Tadea Spiridione Roca Matilde de Bourbon-Duas Sicílias 3133331234 Cardiologista		Programa finaliza sem realizar registro.	Aprovado	
Genaro Mendes 31999991234 Dermatologia, tricologia, saúde capilar		Programa finaliza sem realizar registro.	Aprovado	
Valéria Matos (31)9959-64848 Dermatologista		Erro: telefone deve conter apenas números, solicita novamente numero ao usuário.	Aprovado	

Teste 3: Cadastrar consulta				
Entrada	Classes válidas	Resultado esperado	Classes inválidas	Resultado esperado
<ul style="list-style-type: none"> - Código do médico - Código do paciente - Data - Horário 	<ul style="list-style-type: none"> - Código de médico já cadastrado. - Código de paciente já cadastrado. - Data: string de 8 caracteres no formato "DD/MM/AAAA"; - Horário: string de 5 caracteres no formato HH:MM; 	Consulta cadastrada com sucesso.	<ul style="list-style-type: none"> - Código do médico inexistente. - Médico com 2 consultas marcadas para a data. - Código do paciente inexistente - Data com mais de 8 caracteres -Datas inferiores ao ano 2000. -Datas superiores ao ano 2100; - Hora com mais de 5 caracteres - Horário com menos de 30 minutos de intervalo para consulta já marcada -Horario após as 7:00 e antes das 21:00. 	Erro; solicitar novos dados ao usuário.
Relatório de execução de testes				
Entradas		Resultado	Aprovado?	
2000 1000 12/12/2022 07:00		Consulta cadastrada com sucesso	Aprovado	
2000 1001 12/12/2022 07:30		Consulta cadastrada com sucesso	Aprovado	
2001 1002 15/12/2022 07:30		Consulta cadastrada com sucesso	Aprovado	
2000 1002 12/12/2022 08:00		Erro: Médico já tem 2 consultas cadastradas no dia. Solicita novamente data e hora	Aprovado	
2001 1000 15/12/2022 07:45		Erro: Intervalo entre consultas menor do que 30 minutos. Solicita novamente data e hora.	Aprovado	
2001 1999 15/12/2022 07:45		Erro: Código do paciente inválido. Solicita novamente número ao usuário.	Aprovado	
2999 1000 15/12/2022 07:45		Erro: Código do médico inválido. Solicita novamente um número ao usuário.	Aprovado	