

Projeto e Desenvolvimento de Sistemas I

Guilherme Silva Daires
Lucas Rosa Alves
Marcus Vinicius Thomé

Nome do sistema

ConsultEasy - Clínica Médica

Versão do Documento

1.0

Descrição

O sistema é responsável pela administração das consultas da clínica e o controle de dados dos pacientes e médicos, permitindo o cadastro dos mesmos e realizando o agendamento de consultas, além de possibilitar o controle delas.

Motivação

Para uma clínica médica é necessário que haja uma gestão assertiva no agendamento das consultas, já que são utilizados dados pessoais de médicos e pacientes. Possuir um sistema que automatize, proteja e facilite a manipulação destes dados, não só evita possíveis erros ou colisões nos agendamentos das consultas, mas também proporciona um melhor atendimento aos pacientes e melhores condições para os funcionários.

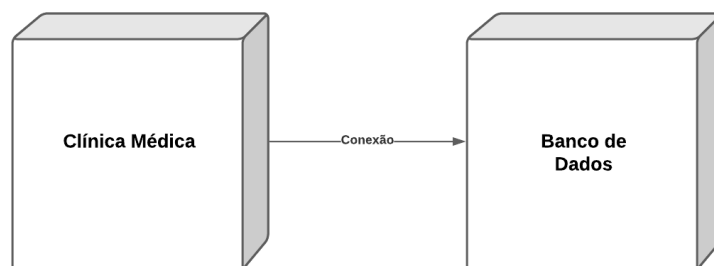
Plataforma-alvo

Microsoft Windows 10

Principais tecnologias utilizadas

.NET Framework 5.0, Docker, SQL Server e Linguagem de programação C#

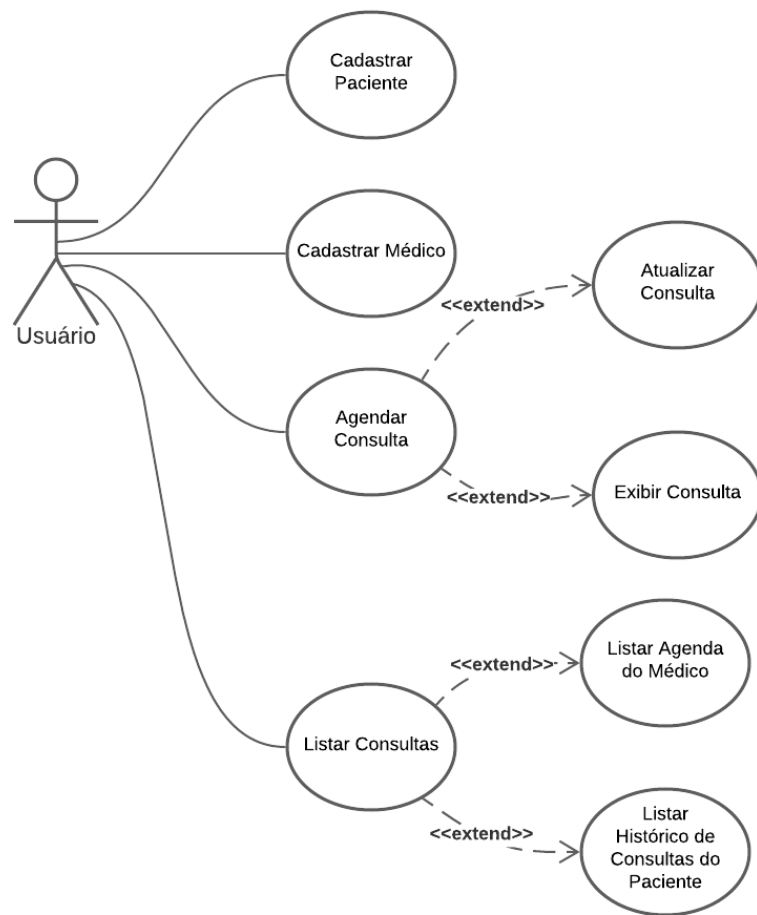
Diagrama de Implantação



O *software* ConsultEasy opera em um único computador, sendo composto apenas por suas interfaces visuais, suas classes e pela integração com o banco de dados. Não é necessária conexão com *internet* nem outros computadores para a sua execução.

ConsultEasy: O primeiro módulo do sistema, é responsável pela interface visual do sistema e pela manipulação dos dados. Já o segundo módulo, bem como responsável pelo armazenamento das informações dos pacientes e das consultas no banco de dados. Escrita utilizando a linguagem C# e Windows.Forms, utilizando o framework Docker para realizar a conexão com o Banco.

Diagrama de casos de uso



Usuário: usuário do sistema. Pode cadastrar um paciente, médico ou uma nova consulta, no qual é possível exibir e atualizar as informações da consulta e listar as consultas do médico ou mesmo listar o histórico de consultas de um paciente.

Cadastrar paciente: Realiza o cadastro do paciente informando o nome, data de nascimento, telefone e profissão.

Cadastrar médico: Realiza o cadastro do médico informando o nome, CRM, especialidade e tempo médio da consulta.

Agendar consulta: Realiza o cadastro da consulta, passando as informações do médico, do paciente, informando também a data da consulta e o horário no qual ela será realizada.

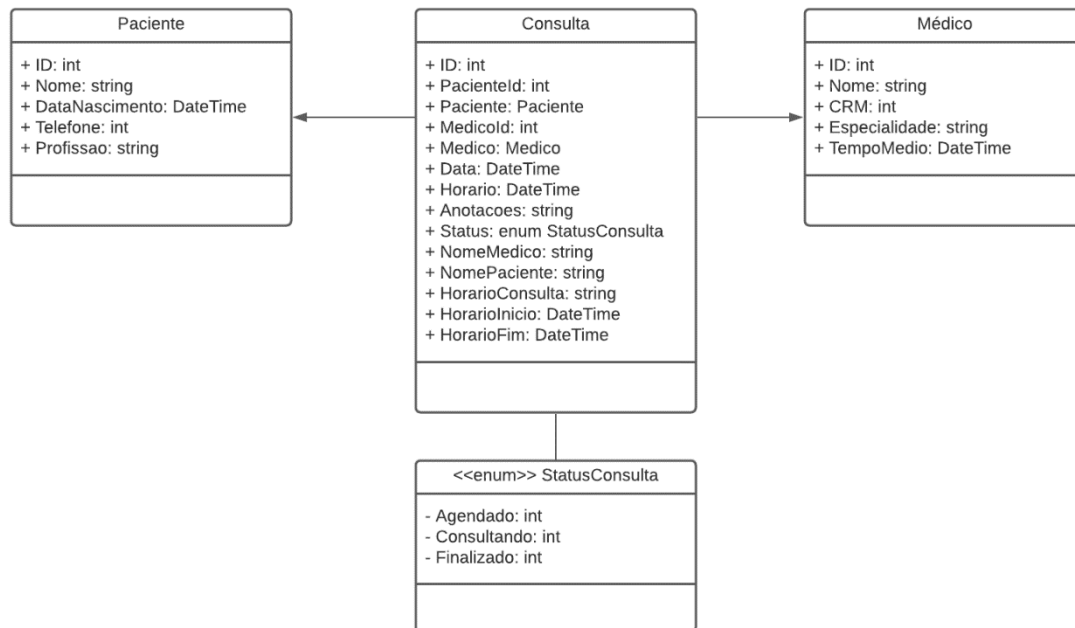
Atualizar consulta: Realiza a alteração das consultas existentes no sistema.

Exibir consulta: Realiza a exibição da consulta contendo as informações passadas pelo usuário e pelo sistema durante o cadastro.

Listar agenda do médico: Realiza a listagem de consultas de um determinado médico naquele dia.

Listar histórico de consultas do paciente: Realiza a exibição do histórico do paciente, no qual são exibidas as consultas efetuadas com um determinado paciente na clínica.

Diagrama de classes



Paciente: classe que possui cinco propriedades, ID, Nome, DataNascimento, Telefone e Profissao, que devem ser utilizadas para armazenar as informações do objeto do tipo Paciente que virá a ser instanciado.

- ID: Uma propriedade do tipo inteiro, que será autoincrementado no banco de dados;
- Nome: Uma propriedade do tipo string, que recebe o nome do paciente;
- DataNascimento: Uma propriedade do tipo DateTime que recebe a data de nascimento do paciente;
- Telefone: Uma propriedade do tipo int, que recebe o telefone do paciente;
- Profissao: Propriedade do tipo string, que recebe a profissão do paciente.

Medico: classe que possui cinco propriedades, ID, Nome, CRM, Especialidade e TempoMedio, que devem ser utilizadas para armazenar as informações do objeto do tipo Médico que virá a ser instanciado.

- ID: Uma propriedade do tipo inteiro, que será autoincrementado no banco de dados;
- Nome: Uma propriedade do tipo string, que recebe o nome do médico;
- CRM: Uma propriedade do tipo int, que recebe o CRM do médico;
- Especialidade: Uma propriedade do tipo string, que recebe a especialidade do médico;
- TempoMedio: Uma propriedade do tipo DateTime, que recebe o tempo médio da consulta de um médico.

Consulta: classe que possui quatorze propriedades, ID, PacienteId, Paciente, MedicoId, Medico, Data, Horario, Anotacoes, Status, NomeMedico, NomePaciente, HorarioConsulta, HorarioInicio e HorarioFim, que devem ser utilizadas para armazenar as informações do objeto do tipo Consulta que virá a ser instanciado.

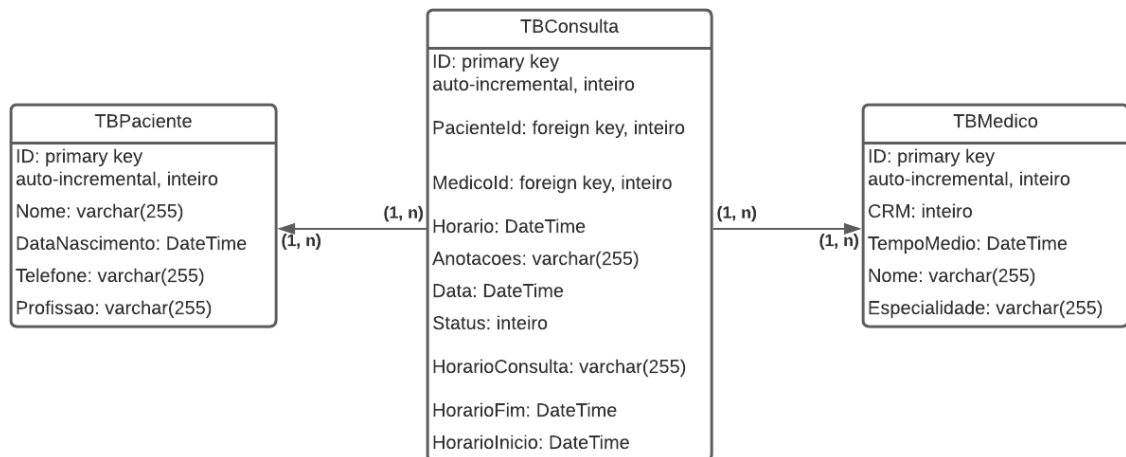
- ID: Uma propriedade do tipo inteiro, que será autoincrementado no banco de dados;
- PacienteId: Uma propriedade do tipo int, que receberá o id do paciente previamente cadastrado;

- Paciente: Uma propriedade do tipo Paciente, que receberá um objeto com as informações de um Paciente cadastrado em um banco de dados;
- MedicId: Uma propriedade do tipo int, que receberá o id do médico previamente cadastrado;
- Medico: Uma propriedade do tipo Medico, que receberá um objeto com as informações de um Médico cadastrado em um banco de dados;
- Data: Uma propriedade do tipo DateTime, que recebe a data da consulta;
- Horario: Uma propriedade do tipo DateTime, que recebe a hora da consulta;
- Anotacoes: Uma propriedade inicialmente nula do tipo string, que pode ser alterada após o início de uma consulta;
- Status: Uma propriedade do tipo enum StatusConsulta, que receberá o status de uma consulta;
- NomeMedico: Uma propriedade do tipo string, que recebe o nome do médico para uma posterior exibição da consulta na data grid. Esta propriedade não será armazenada no banco de dados;
- NomePaciente: Uma propriedade do tipo string, que recebe o nome do paciente para uma posterior exibição da consulta na data grid. Esta propriedade não será armazenada no banco de dados;
- HorarioConsulta: Uma propriedade do tipo string, que recebe a hora da consulta para uma posterior exibição da consulta na data grid;
- HorarioInicio: Uma propriedade do tipo DateTime, que recebe a hora do início da consulta. Salvo apenas para registro da consulta;
- HorarioFim: Uma propriedade do tipo DateTime, que recebe a hora do fim da consulta. Salvo apenas para registro da consulta;

StatusConsulta: enumeração que possui três itens, Agendado, Consultando e Finalizado.

- Agendado: atribui o status “Agendado” para a consulta que for agendada;
- Consultando: atribui o status “Consultando” para a consulta que foi iniciada, porém não finalizada;
- Finalizado: atribui o status “Finalizado” para a consulta que foi encerrada.

Diagrama de banco de dados



TBPaciente: tabela que possui cinco atributos, ID, Nome, DataNascimento, Telefone e Profissao, que devem ser utilizadas para armazenar as informações do objeto do tipo Paciente do sistema no banco de dados.

- ID: Um atributo do tipo inteiro, de chave primária, não nulo que será autoincrementado;
- Nome: Um atributo do tipo varchar, não nulo, que armazenará o nome do paciente;
- DataNascimento: Um atributo do tipo DateTime, não nulo, que armazenará a data de nascimento do paciente;
- Telefone: Um atributo do tipo inteiro, não nulo, que armazenará o telefone do paciente;
- Profissao: Um atributo do tipo varchar, não nulo, que recebe a profissão do paciente.

TBMedico: tabela que possui cinco atributos, ID, Nome, CRM, Especialidade e TempoMedio, que devem ser utilizadas para armazenar as informações do objeto do tipo Médico do sistema no banco de dados.

- ID: Um atributo do tipo inteiro, de chave primária, não nulo que será autoincrementado;
- Nome: Um atributo do tipo varchar, não nulo, que armazenará o nome do médico;
- CRM: Um atributo do tipo inteiro, unique, não nulo, que armazenará o CRM do médico;
- Especialidade: Um atributo do tipo varchar, não nulo, que armazenará a especialidade do médico;
- TempoMedio: Um atributo do tipo DateTime, não nulo, que armazenará o tempo médio da consulta de um médico.

TBConsulta: tabela que possui dez atributos, ID, Pacienteld, Medicold, Horario, Anotacoes, Data, Status, HorarioConsulta, HorarioFim e HorarioInicio, que devem ser utilizadas para armazenar as informações do objeto do tipo Consulta do sistema no banco de dados.

- ID: Um atributo do tipo inteiro, de chave primária, não nulo que será autoincrementado;
- Pacienteld: Um atributo do tipo inteiro, chave estrangeira (Fará o relacionamento com a tabela TBPaciente para a obtenção dos dados), não nulo, que receberá o id do paciente previamente cadastrado;

- **Medicoid:** Um atributo do tipo inteiro, chave estrangeira (Fará o relacionamento com a tabela TBMedico para a obtenção dos dados), não nulo, que receberá o id do médico previamente cadastrado;
- **Horario:** Um atributo do tipo DateTime, não nulo, que armazenará a hora da consulta;
- **Anotacoes:** Um atributo inicialmente nulo do tipo varchar;
- **Data:** Um atributo do tipo DateTime, não nulo, que armazenará a data da consulta;
- **Status:** Um atributo do tipo inteiro, que armazenará o índice do enum StatusConsulta;
- **HorarioConsulta:** Um atributo do tipo string, inicialmente nulo, que armazenará a hora da consulta para uma posterior exibição da consulta na data grid;
- **HorarioFim:** Um atributo do tipo DateTime, inicialmente nulo, que armazenará a hora do fim da consulta;
- **HorarioInicio:** Um atributo do tipo DateTime, inicialmente nulo, que armazenará a hora do início da consulta;

Relacionamentos:

- **TBConsulta – TBMedico:** (1, n) Um médico pode estar relacionado a várias consultas, mas uma consulta, poderá possuir apenas um único médico;
- **TBConsulta – TBPaciente:** (1, n) Um paciente pode estar relacionado a várias consultas, mas uma consulta, poderá possuir apenas um único paciente.

Requisitos Funcionais

RF01: O programa deve possuir um formulário principal no qual exibe um menu (menuStrip) com as funcionalidades do sistema. O primeiro item do menu será o “Cadastro” no qual ao clicar exibe submenu, com as opções “Cadastrar Paciente” e “Cadastrar Médico”. O segundo item do menu será o “Consulta” no qual ao clicar, exibe um submenu com a opção de “Agendar Consulta”. O terceiro item é o de “Agenda” no qual ao clicar no submenu, apresenta três opções sendo “Agenda Médico”, “Historico do Paciente” e “Consultas do Dia”. O último item do menu é o de “Sair” no qual ao ser clicado, apresenta uma mensagem para o usuário, perguntando se de fato ele gostaria de fechar o sistema ou não, caso ele clique em não, a mensagem é fechada e se sim o sistema será encerrado.

RF02: Ao clicar no item “Cadastrar Paciente”, deverá abrir um formulário no qual o usuário fará o cadastro do paciente. Ele possuirá os campos de texto Nome, Telefone e Profissão, também contará com um datePicker para selecionar a data de nascimento. Todos os campos são obrigatórios, portanto, se houver ao menos um campo sem informação nenhuma, ele deverá alertar o usuário de qual campo deve ser preenchido. Utilizar um controle errorHandler para realizar o alerta. Caso não haja problemas com a entrada, o sistema deverá salvar as informações do paciente e armazenar no banco de dados.

RF03: Ao clicar no item “Cadastrar Médico”, deve abrir um formulário no qual o usuário fará o cadastro do médico. Ele possuirá os campos de texto Nome, Especialidade e CRM, também contará com um datePicker para definir em minutos o tempo médio que o médico leva para consultar. Todos os campos são obrigatórios, portanto, se houver ao menos um campo sem informação nenhuma, ele deverá alertar o usuário de qual campo deve ser preenchido. Utilizar um controle errorHandler para realizar o alerta. Caso não haja problemas com a entrada, o sistema deverá salvar as informações do paciente e armazenar no banco de dados.

RF04: Ao clicar no item de “Agendar Consulta”, deve abrir um formulário no qual o usuário fará o agendamento da consulta. Ele possuirá duas caixas de texto com uma lista de informações(comboBox), o primeiro para selecionar o paciente (este comboBox terá como fonte de dados uma lista de nomes dos pacientes cadastrados anteriormente no banco de dados) e o segundo para selecionar o médico (este comboBox terá como fonte de dados uma lista de nomes dos médicos cadastrados anteriormente no banco de dados). Além disso, o formulário deve conter dois elementos para a obtenção de data e hora(datePicker), o primeiro para selecionar a data da consulta a ser agendada e o segundo para selecionar o horário da mesma. Deverá ser feita a verificação de disponibilidade de data e horário das consultas de acordo com o médico, uma consulta não pode ser agendada no mesmo horário de outra, ou no intervalo de atendimento de uma consulta, esta verificação terá como base as consultas previamente agendadas e cadastradas no banco de dados e o tempo médio (informado no cadastro do médico) do médico selecionado para a consulta. O usuário não poderá agendar consultas para um datas já passadas. Caso não haja problema com as entradas e verificações realizadas, o sistema deve salvar as informações e armazena-las no banco de dados.

RF05: Ao clicar no item de “Agenda Médico”, deve abrir um formulário no qual fará a exibição das consultas que um determinado médico possui naquele dia. Ele possuirá um comboBox para selecionar o médico para listar as consultas (este comboBox terá como fonte de dados uma lista de nomes dos médicos cadastrados anteriormente no banco de dados). Além disso, também contará com um datePicker que definirá a data em que as consultas deverão ser exibidas (terá como data padrão, a data do dia atual). Contará ainda com uma lista de consultas (dataGridView) que exibirá as informações das consultas (Data, Horário, Médico, Paciente e Status) do médico selecionado na data escolhida. (esta lista(dataGridView) terá como fonte de dados uma lista de objetos que serão obtidos da tabela TBConsulta no banco de dados. Só serão inseridos na lista de objetos, as consultas que tem como valor do campo "Medicoid" o mesmo valor do

campo ID da tabela TBMedico obtido através do nome selecionado no comboBox e como valor do campo "Data" o mesmo valor de data selecionado no dateTimePicker).

Quando o usuário der clique duplo em uma determinada consulta do dataGrid, deverá abrir um formulário que possibilita a visualização dos dados da consulta, possuindo 8 caixas de texto, disponíveis apenas para leitura, são elas: Médico(que exibe o nome do médico), Data(que exibe a data da consulta), Hora(que exibe o horário da consulta), Paciente(que exibe o nome do paciente), Data de Nascimento(que exibe a data de nascimento do paciente), Telefone(que exibe o telefone do paciente), Profissão(que exibe a profissão do paciente) e Anotações(campo inicialmente vazio, que permitirá a adição de valores quando iniciada a consulta). Todos os dados dos campos, deverão ser obtidos no banco de dados. Os dados dos pacientes são obtidos na tabela TBPaciente, através do ID (o ID é obtido através do Nome do Paciente no ComboBox informado no agendamento da consulta). Os dados do médico são obtidos na tabela TBMedico, através do ID (o ID é obtido através do Nome do Médico no ComboBox informado no agendamento da consulta). O formulário contará também com 3 botões, "Iniciar Consulta", "Cancelar" e "Encerrar Consulta".

O Botão "Iniciar Consulta" habilitará a caixa de texto de "Anotações" para edição e possibilitará a adição de valores neste campo que será posteriormente salvo no banco de dados. Além disso, este botão altera o status da consulta de "Agendado" para "Consultando", além de atribuir o horário do início da consulta no campo "HorarioInicio" da tabela "TBConsulta" através do DateTimeNow.

O Botão "Cancelar" fecha o formulário sem salvar as alterações feitas em "Anotações" e fechará o formulário. Ao clicar em "Cancelar" deverá ser exibida uma confirmação de fechamento do formulário através de um MessageBox com as opções "Sim" e "Não", quando clicado em "Sim" deverá ser feito o fechamento do formulário, e para a outra afirmativa, o formulário continuará aberto.

O Botão "Encerrar Consulta" salva os dados inseridos na caixa de texto "Anotações" no banco de dados, além de mudar o Status para "Finalizado" e também atribui o horário do salvamento no campo "HorarioFim" da tabela "TBConsulta" através do DateTimeNow.

RF06: Ao clicar no item de "Histórico do Paciente" deverá abrir um formulário para a exibição de todas as consultas já feitas com determinado paciente. Este formulário possuirá um comboBox para selecionar o paciente para listar as consultas (este combobox terá como fonte de dados uma lista de nomes dos pacientes cadastrados anteriormente no banco de dados). Conterá ainda com um dataGrid que exibirá as informações das consultas com status "Finalizado" (Data, Horário, Paciente, Médico e Status) do paciente selecionado (este dataGrid terá como fonte de dados uma lista de objetos que serão obtidos da tabela TBConsulta no banco de dados. Só serão inseridos na lista de objetos, as consultas que tem como valor do campo "Pacienteld" o mesmo valor do campo ID da tabela TBPaciente obtido através do nome selecionado no comboBox e como valor do campo "Status", o valor 2, que significa "Finalizado". Quando o usuário der clique duplo em determinado item do dataGrid, deverá abrir um formulário que possibilita a visualização dos dados da consulta, possuindo 9 caixas de texto, disponíveis apenas para leitura, são elas: Médico(que exibe o nome do médico), Data(que exibe a data da consulta), Hora(que exibe o horário agendado para a consulta), Hora Inicio (que exibe o horário em que a consulta começou), Hora Fim(que exibe o horário em que a consulta terminou), Paciente(que exibe o nome do paciente), Data de Nascimento(que exibe a data de nascimento do paciente), Telefone(que exibe o telefone do paciente), Profissão(que exibe a profissão do paciente) e Anotações(que exibe as anotações que o médico fez após o início da consulta). Todos os dados dos campos, deverão ser obtidos no banco de dados. Os dados dos pacientes são obtidos na tabela TBPaciente, através do ID (o ID é obtido através do Nome do Paciente no ComboBox informado no agendamento da consulta). Os dados do médico são

obtidos na tabela TBMedico, através do ID (o ID é obtido através do Nome do Médico no ComboBox informado no agendamento da consulta).

RF07: Ao clicar no item de "Consultas do Dia" deverá abrir um formulário para a exibição de todas as consultas do dia atual. Este formulário contará com uma lista que exibirá as informações de todas as consultas (Data, Horário, Paciente, Médico e Status) do dia atual (esta lista terá como fonte de dados uma lista de objetos que serão obtidos da tabela TBConsulta no banco de dados. Só serão inseridos na lista de objetos, as consultas que tem como valor do campo "Data" o valor do dia atual). Quando o usuário der clique duplo em determinado item da lista, deverá abrir um formulário que possibilita a visualização dos dados da consulta, possuindo 8 caixas de texto, disponíveis apenas para leitura, são elas: Médico(que exibe o nome do médico), Data(que exibe a data da consulta), Hora(que exibe o horário da consulta), Paciente(que exibe o nome do paciente), Data de Nascimento(que exibe a data de nascimento do paciente), Telefone(que exibe o telefone do paciente), Profissão(que exibe a profissão do paciente) e Anotações(campo inicialmente vazio, que permitirá a adição de valores quando iniciada a consulta). Todos os dados dos campos, deverão ser obtidos no banco de dados. Os dados dos pacientes são obtidos na tabela TBPaciente, através do ID (o ID é obtido através do Nome do Paciente no ComboBox informado no agendamento da consulta). Os dados do médico são obtidos na tabela TBMedico, através do ID (o ID é obtido através do Nome do Médico no ComboBox informado no agendamento da consulta).

RF09: Para todos os formulários, todos os controles devem ser redimensionados conforme o redimensionamento do formulário. O formulário deve ser iniciado de forma centralizada na tela.

Requisitos Não-Funcionais

RNF01: O sistema deve ser executado em computadores executando o Microsoft Windows 10 e o .NET Framework 5.0;

RNF02: Para facilitar a utilização, os controles devem ser selecionados na seguinte ordem: de cima para baixo, da esquerda para a direita;

RNF03: O sistema será executado em um computador com processador atual e 4GB de memória RAM, utilizando o sistema operacional Windows.