

Menu
-gerenciador Gerenciador; -void limparTela() const; -int lerOpcao() const;
Menu(); +void exibirMenuPrincipal(); +void cadastrarPaciente(); +void cadastrarFisioterapeuta(); +void recuperarPaciente();


```
-drf: string
-vector<string> pacientes
#vector<pair<int, int>> horariosOcupados
#vector<pair<int, int>> horarios_livres

Fisioterapeuta(const string& nome,int hora, int
const string& telefone, const string& drf)

+string getDRF() const;
+void atualizarFisioterapeuta(vector <Fisiotera
+void adicionarpaciente(const string& nome_p:
+void adicionarHorarioocupado(int hora,int min
+void mostrarpacientes() const;
+vector <pair<int, int>> getHorariosLivres () coi
+const vector <pair<int, int>> gethorariosOcup:
+void atualizarDados(const string& nome, int h
+void atualizarDRF(const string& novoDRF);
```

```
+void pesquisarPaciente();
+void pesquisarFisioterapeuta();
+void listarPacientesPorHorario();
+void mostrarFisioPorHorario();
+void atualizarDadosPessoa();
+void atualizarDadosPaciente();
+void atualizarDadosFisioterapeuta();
+void mudarHorarioPaciente();
+void removerPaciente();
+void removerFisioterapeuta();
```

```
gerenciador();
~gerenciador();
+void cadastrarpaciente(Paciente* paciente);
+void cadastrarfisioterapeuta(Fisioterapeuta* fisioterapeuta);
+void pesquisarPacientePorNome(const string& nome) const;
+void pesquisarFisioPorNome(const string& nome) const;
+void listarPacientesPorHorario() const;
+void atualizarPessoa(const string& cpf, const string& novoNome, int novaHora, int novoMinuto, int novaldade, const string& novoTelefone);
+void atualizarPaciente(const string& cpf, const string& novoResponsavel, const string& novoTipoPacote, bool novoPagamento);
+void atualizarFisioterapeuta(const string& cpf, const string& novoDRF);
+void mudarHorarioPaciente(const string& cpf, int novaHora, int novoMinuto);
+void removerPaciente(const string& cpf);
+void removerFisioterapeuta(const string& cpf);
+ vector<Pessoa*> pessoaList;
+vector <Paciente*> pacienteList;
+vector <Fisioterapeuta*> fisioList;
+void salvarDados();
+void carregarDados();
```

```
Pessoa

#string nome;
#int hora;
#int minuto;
#string cpf;
#int idade;
#string telefone;
#vector<Pessoa*> pessoas;
#vector<Pessoa*> pessoaList;

+ Pessoa(const string& nome, int hora, int minuto,const string& cpf, int idade, const string& telefone);
virtual ~Pessoa() {}
+string getNome() const;
+int getHora() const;
+int getMinuto() const;
+string getCPF() const;
+int getIdade() const;
+string getTelefone() const;
+virtual void atualizarDados (const string& nome, int hora, int minuto, int idade, const string& telefone)= 0;
```

```
Fisioterapeuta

minuto, const string& cpf,int idade,

peuta> & fisioList, const string &CPF, int idade, const string& DRF );
aciente);
uto);

nst;
ados ();
ora, int minuto, int idade, const string& telefone) override;
```

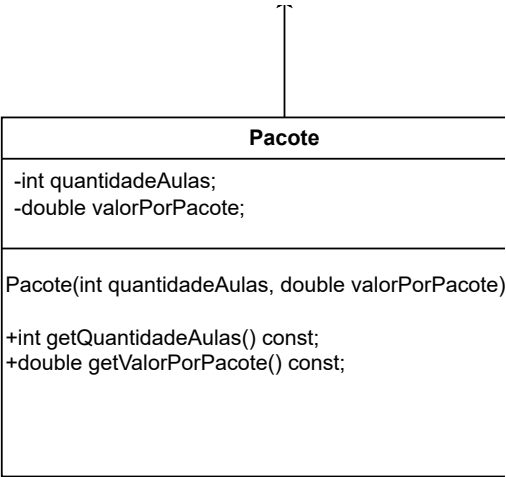
```
Paciente

-bool pagamento;
#string responsavel;
#string tpacote;

Paciente(const string& nome, int hora, int minuto,const string& cpf, int idade, c
bool pagamento, const string& responsavel, const string& tpacote);
+string getResponsavel() const;
+string getTipoPacote () const;
+bool verificarpagamento() const;
+void setPagamento(bool pagamento);
+void atualizarDados(const string& nome, int hora, int minuto, int idade, const :
+void atualizarResponsavel(const string& novoResponsavel);
+void atualizarTipoPacote(const string& novoTipoPacote);
+void atualizarPagamento(bool novoPagamento);
```

const string& telefone,
string& telefone) override;

nt horaInicio, int minutoInicio, int horaFim, int minutoFim) const;



);