**UNIVERSIDADE PAULISTA**

**Gabriel Alcantara Gragnani**

**RA: N58245-6**

**MEDICINA INTELIGENTE**

Sistema hospitalar integrado

**CAMPINAS – SP**

**2022**

**UNIVERSIDADE PAULISTA**

**Eliezer Valencio**

**Gabriel Alcantara Gragnani**

**Geovanna Santos Ribeiro**

**Henrique Xavier de Sousa**

**MEDICINA INTELIGENTE**

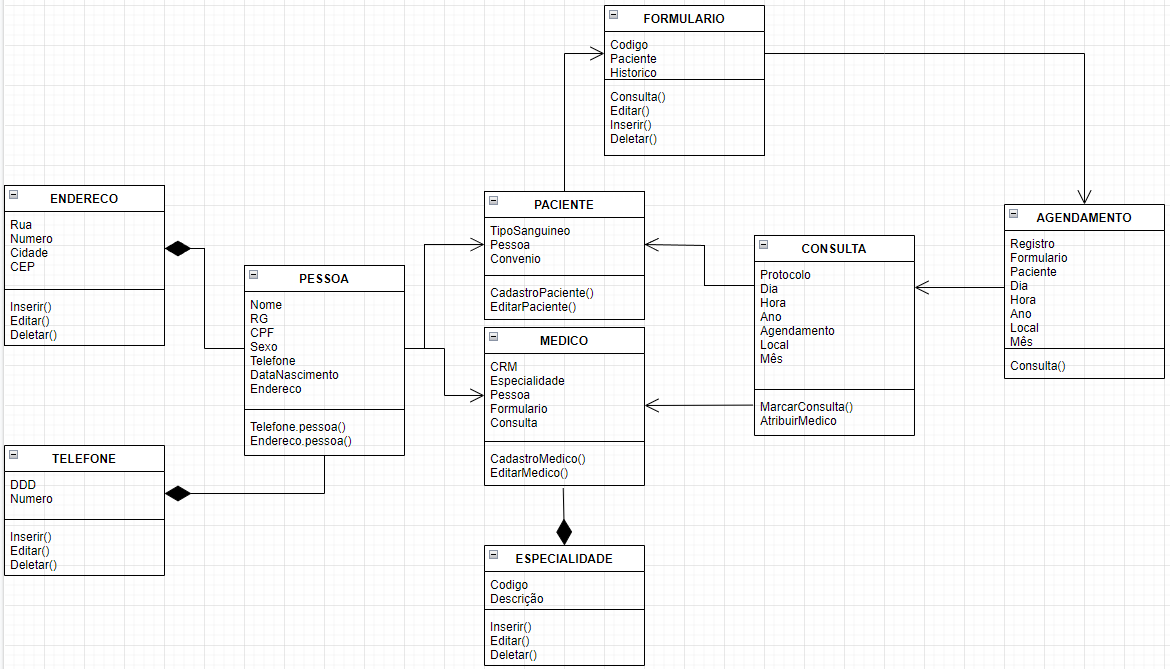
Sistema hospitalar integrado

**CAMPINAS – SP**

**2022**

**Diagrama de classe**

Foram criadas as classes endereço, telefone, pessoa, paciente, medico, especialidade, formulário, agendamento e consulta, onde cada classe possui informação da outra e irá conter em melhorias futuras o método de herança para pegar os dados de outra classe (FIGURA 1).

FIGURA 1 – DIAGRAMA DE CLASSE

No diagrama (FIGURA 1) é possível observar que a pessoa herda as informações da classe endereço e telefone, com isso é gerado o cadastro de uma pessoa. Após o cadastro de uma pessoa ela se identifica como um paciente precisando de um agendamento ou como um médico e sua especialidade na área, se a pessoa for identificada como um paciente seguirá um formulário para preencher algumas informações e realizar um agendamento para uma consulta com um médico.

**Programação**

Foram criadas as classes endereço, telefone, pessoa, paciente, formulário, agendamento, consulta, especialidade e medico; onde tem objetivo de cadastrarem e coletarem informações uma das outras.

using System;

public class Program

{

public class endereco

{

public string rua {get;set;}

public double numero {get;set;}

public string cidade {get;set;}

public double cep {get;set;}

public endereco()

{

rua = "rua 1";

numero = 0;

cidade = "";

cep = 0;

}

}

public class telefone

{

public double ddd {get;set;}

public double numero {get;set;}

public telefone()

{

ddd = 0;

numero = 0;

}

}

public class pessoa

{

public string nome {get;set;}

public endereco end {get;set;}

public telefone tel {get;set;}

public double rg {get;set;}

public int cpf {get;}

public string sexo {get;set;}

public pessoa()

{

tel = new telefone();

end = new endereco();

nome = "";

rg = 0;

cpf = 0;

sexo = "";

}

}

public class paciente

{

public pessoa pes1 {get;set;}

public string TipoSanguineo {get;}

public paciente()

{

pes1 = new pessoa();

TipoSanguineo = "";

}

}

public class formulario

{

public string codigo {get;}

public paciente pac1 {get;set;}

public string historico {get;set;}

public formulario()

{

pac1 = new paciente();

historico = "";

codigo = “”;

}

}

public class agendamento

{

public formulario form1 {get;set;}

public paciente pac2 {get;set;}

public int mes {get;set;}

public int dia {get;set;}

public int ano {get;set;}

public int hora {get;set;}

public string local {get;set;}

public string registro {get;}

public agendamento()

{

form1 = new formulario();

pac2 = new paciente();

mes = 0;

dia = 0;

ano = 0;

hora = 0;

local = "";

registro = “”;

}

}

public class consulta

{

public string protocolo {get;}

public int dia {get;set;}

public int mes {get;set;}

public int hora {get;set;}

public int ano {get;set;}

public agendamento agend1 {get;set;}

public string local {get;set;}

public consulta()

{

agend1 = new agendamento();

protocolo = "";

dia = 0;

mes = 0;

hora = 0;

ano = 0;

local = "";

}

}

public class especialidade

{

public string codigo {get;}

public string descricao {get;set;}

public especialidade()

{

codigo = "";

descricao = "";

}

}

public class medico

{

public especialidade espec1 {get;set;}

public pessoa pes2 {get;set;}

public formulario form2 {get;set;}

public consulta consult1 {get;set;}

public string crm {get;}

public medico()

{

espec1 = new especialidade();

pes2 = new pessoa();

form2 = new formulario();

consult1 = new consulta();

crm = "";

}

}

public void cadastrar\_endereco(double endereco)

{

//método para realizar o cadastro do endereço de um usuário

}

public void cadastro\_telefone(double telefone)

{

//método para realizar o cadastro do telefone de um usuário

}

public void cadastro\_pessoa(double pessoa)

{

//método para realizar o cadastro de uma pessoa

}

public void cadastro\_paciente(double paciente)

{

//método para realizar o cadastro de uma pessoa registrada como paciente ao dar entrada no hospital, obtendo suas informações

}

public void c\_form(double formulario)

{

//método para preenchimento de informações sobre acontecimentos e sintomas do paciente

}

public void c\_agendamento(double agendamento)

{

//método para agendar uma consulta no sistema

}

public void c\_consulta(double consulta)

{

//método que contém as informações do agendamento e do local da consulta

}

public void c\_especialidade(double especialidade)

{

//método que contém informações sobre qual área o médico é formada ou tem maior especialidade

}

public void cadastro\_medico(double medico)

{

//método que contém informações sobre o médico, como por exemplo sua especialidade e seu CRM para poder exercer a função

//também contém as informações sobre os formulários dos pacientes

}

public static void Main()

{

}

}