Domain Driven Design

FIAP - CAMPUS PAULISTA - TURMA 1TDSPT - PERÍODO NOTURNO



Participantes

Igor Gabriel Pereira Marcondes - RM553544

Luccas dos Anjos Correia da Silva - RM552890

Daniel Fernando Alves de Magalhães - RM553785

Sumário

Descrição do problema	. 3
•	
Descrição da solução idealizada	. :
Diagrama de Classes	. 1

Dossiê do Paciente

Descrição do problema

Acompanhamento por meio de um portfólio que engloba toda a trajetória do paciente naquela especialidade, compensando a ausência de acompanhamento médico direto. Isso permite que o médico análise todos os tratamentos previamente realizados na especialidade específica e tome decisões sobre exames e prescrições com base em informações já disponíveis, caso existam.

O sistema tecnológico apoia na tomada de decisões, distinguindo entre consultas necessárias para solucionar diferentes tipos de doenças, independentemente de sua frequência. Isso assegura a eficácia do tratamento, evitando procedimentos redundantes.

Descrição da solução idealizada

Desenvolver um software associado ao CPF que centraliza todas as consultas, documentos em formato PDF e análises médicas e apontamentos do paciente ao longo da vida de uma pessoa naquela especialidade, independentemente de convênios, particulares ou do Sistema Único de Saúde (SUS). Esse sistema garantiria acesso abrangente a todos os profissionais de saúde, permitindo uma visão completa dos tratamentos e histórico de saúde do paciente. O foco principal seria facilitar o acompanhamento e planejamento de tratamentos eletivos para o paciente.

Diagrama de Classes

User - name: String - email: String - nascimento; String - cpf: String - password: String + User () + User (parameter) + getters (): void + setters (): void + validarEmail (): + validarData (): + validarCPF (): + Registro () Paciente Funcionario - phone: String - cargo: String - cep: String - idFuncionario: String - sexo: String - tipoSangue; String + Funcionario () + Funcionario (parameter) - Declaracao: String + Paciente () + getters (): void + Paciente (parameter) + setters (): void + getters (): void + validarld (): void + setters (): void + MenuFuncionario (): void + validarPhone (): void + Diagnostico (): void + validarCEP (); void + Registro (): void + validarSexo (); void + validarTipoSangue (): void + MenuPaciente (): void + Declaracao (): void + Registro (); void