logo

**RELATÓRIO**

**BASE DE DADOS II**

Projeto de Banco de Dados – SIGEPOLI

|  |  |
| --- | --- |
| **INTEGRANTES DO GRUPO** | |
| **PRESELEY RODRIGUES – 20181423** | |
| **CURSO:** Engenharia Informática  **TURMA:** EINF\_M2 | **DOCENTE:** JUDSON PAIVA |
| **DATA:** 16/06/2025 | **ANO LECTIVO:** 2024/25 |

## **Implementação de Regras de Negocio**

### **RN01: Professor não pode ter aulas com horários sobrepostos**

### **Implementação:**

### **Mecanismo:**

### **Trigger que dispara antes de inserir uma nova alocação de professor (Alocacao).**

### **Verifica se existe sobreposição de horários usando a condição: (horário\_existente.início < novo\_horário.fim) AND (horário\_existente.fim > novo\_horário.início).**

### **Bloqueia a inserção se houver conflito.**

### **RN02: Matrícula só com vaga disponível e propina paga**

### **Implementação:**

### **Mecanismo:**

### **1. Trigger: Valida se há vagas na turma (capacidade - matrículas ativas > 0).**

### **2. Procedimento MatricularAluno:**

### **Verifica vagas disponíveis.**

### **Checa propina\_paga na tabela Aluno.**

### **Bloqueia matrícula se alguma condição falhar.**

### **RN03: Notas entre 0–20**

### **Implementação:**

### **Mecanismo:**

### **CHECK constraint diretamente na coluna nota.**

### **Rejeita automaticamente valores fora do intervalo [0, 20].**

### **RN04: Garantia válida antes do pagamento**

### **Implementação:**

### **Mecanismo:**

### **Trigger que verifica se a validade da garantia (Garantia.validade) é >= data atual.**

### **Bloqueia pagamento se a garantia estiver vencida.**

### **RN05: SLA < 90% gera multa automática**

### **Implementação:**

### **Mecanismo:**

### **Colunas geradas (generated columns):**

### **multa: Calcula 10% do valor se sla\_efetivo < 90.**

### **valor\_pago: Subtrai a multa do valor original.**

### **Atualização automática ao inserir/atualizar.**

### **RN06: Coordenador deve aprovar carga horária**

### **Implementação:**

### **Mecanismo:**

### **Flag aprovado na tabela Alocacao com valor padrão FALSE.**

### **Aprovação manual via UPDATE (ex: UPDATE Alocacao SET aprovado = TRUE WHERE id = X).**

### **Relacionamento entre Coordenador e Curso via tabela Coordenador:**

### **Mecanismos Adicionais de Controle:**

### **1. Consistência Temporal:**

### **2. Auditoria:**

### **Triggers registram operações críticas.**

### **3. Procedimentos para Integridade:**

### **AlocarProfessor: Valida RN01 antes de inserir.**

### **ProcessarPagamento: Valida RN04 antes de pagar.**

### **4. Modelagem de Hierarquias:**

### **Funcionários como PROFESSOR/ADMINISTRATIVO via ENUM.**

### **Especialização com tabelas Professor e Coordenador.**

### **Diagrama de Implementação das RNs:**

### **Regra Mecanismo Principal Componentes Adicionais**

### **RN01 Trigger BEFORE INSERT Comparação de intervalos**

### **RN02 Trigger + Procedimento Verificação propina\_paga**

### **RN03 CHECK na coluna ---**

### **RN04 Trigger BEFORE INSERT Validação de data**

### **RN05 Colunas Geradas Cálculo condicional**

### **RN06 Flag aprovado Tabela Coordenador**

### **Todas as regras foram implementadas diretamente no esquema do banco de dados, garantindo integridade mesmo se a aplicação falhar.**