

PROJETO IF ODONTO - NORMALIZAÇÃO (ETAPA 3)

Nesta etapa será realizada a normalização do banco de dados, método empregado para aumentar a qualidade do projeto de BD. O processo de normalização consiste em um conjunto de regras, denominadas formas normais, que devem ser aplicadas na tabela não normalizada. Ele permite, de forma gradual, substituir um conjunto de entidades e relacionamentos por um outro, mais eficiente em relação às anomalias de atualização (inclusão, alteração e exclusão), sendo necessária para garantir a integridade do modelo de dados.

Para realizá-lo, eliminaremos as redundâncias a partir do modelo lógico de dados, de acordo com as chamadas formas normais (1FN, 2FN, 3FN). O processo de normalização para o banco de dados IF Odonto será feito nos tópicos que se seguem.

- **Tabelas do BD**

Na etapa anterior, foi feita a transformação entre modelos. Essa transformação gerou um modelo lógico composto por cinco tabelas devidamente relacionadas entre si. Para fins de organização, essas tabelas serão aqui lembradas:

Consulta (idConsulta, idFuncionario, idPaciente, dataConsulta, valorConsulta, procedimento, formaPagamento)
idFuncionario referencia Funcionario
idPaciente referencia Paciente

Clinica (cnpjClinica, nomeClinica, telClinica, endClinica, idFuncionario, idGerente, chaveAcesso)
idFuncionario referencia Funcionario

Funcionario (idFuncionario, cpfFunc, nomeFunc, endFunc, salário, dataNascFunc, dataAdmissao, cnpjClinica)
cnpjClinica referencia Clinica

Dentista (idFuncionario, idDentista, CRO, especialidade)

idFuncionario referencia Funcionario

Paciente (idPaciente, cpfPaciente, nomePaciente, telefonePaciente, endPaciente, dataCadastro, indicacao, dataNascPaciente)

- **Primeira Forma Normal (1FN)**

A Primeira Forma Normal diz respeito à primeira etapa da normalização. Pode-se dizer que uma tabela está na 1FN se todos os seus atributos forem atômicos, ou seja, não são permitidos atributos multivalorados ou compostos.

Para normalizar uma tabela e deixá-la na 1FN é necessário **decompor os atributos compostos e transformá-los em atributos simples, e criar uma tabela para cada atributo multivalorado**. No segundo cenário, a nova tabela irá conter o atributo multivalorado mais a chave primária da tabela inicial, que passa como chave estrangeira. Dessa forma, a chave primária da nova tabela será composta.

Aplicando essa regra para cada uma das tabelas do banco de dados, teremos o seguinte:

Para a tabela Consulta:

Nesta tabela, temos o atributo multivalorado procedimento, visto que em uma consulta um paciente poderá fazer mais de um procedimento. Dessa forma, será criada uma nova tabela chamada Procedimento, que terá o atributo multivalorado nomeProcedimento e a chave primária da tabela Consulta(idConsulta), que passará como chave estrangeira. Logo, a chave primária da nova tabela será composta. Além disso, o atributo procedimento sairá da tabela Consulta.

Assim, tem-se a seguinte configuração:

Consulta (idConsulta, idFuncionario, idPaciente, dataConsulta, valorConsulta, formaPagamento)

idFuncionario referencia Funcionario

idPaciente referencia Paciente

Procedimento (nomeProcedimento, idConsulta)

idConsulta referencia Consulta

Para as tabelas Clínica e Funcionário:

Nessas tabelas, não encontramos nenhum atributo multivalorado, ou seja, não será necessário criar uma nova tabela. Temos o atributo endClinica na tabela Clínica e endFunc na tabela Funcionario, ambos compostos, visto que endereço contém estado, cidade, bairro, rua e número. Dessa forma, é necessário realizar o desmembramento e transformá-los em atributos simples.

Assim, as tabelas ficarão da seguinte maneira:

Clinica (cnpjClinica, nomeClinica, telClinica, rua, bairro, numero, cidade estado, idFunGerente, chaveAcesso)

idFuncionario referencia Funcionario

Funcionario (idFuncionario, cpfFunc, nomeFunc, rua, bairro, numero, cidade, estado, salário, dataNascFunc, dataAdmissao, cnpjClinica)

cnpjClinica referencia Clinica

Para a tabela Dentista:

Nessa tabela, temos o atributo multivalorado especialidade, já que no contexto desse sistema um dentista pode ter mais de uma especialidade. Dessa forma, é necessário criar uma nova tabela, chamada Especialidade, que terá como atributos idFuncionario, que é chave primária de Dentista, e nomeEspecialidade, o atributo multivalorado. Os dois atributos formarão a chave primária composta da nova tabela, com idFuncionario referenciando a

tabela Funcionario e, conseqüentemente, sendo também chave estrangeira. Além disso, o atributo sairá da tabela Dentista.

Assim, teremos:

Dentista (idFuncionario, idDentista, CRO)

idFuncionario referencia Funcionario

Especialidade (idFuncionario, nomeEspecialidade)

idFuncionario referencia Funcionario

Para a tabela Paciente:

Nessa tabela, temos o atributo endPaciente, que é composto, já que endereço contém estado, cidade, bairro, rua e número. Logo, é necessário realizar o desmembramento desse atributo e transformá-lo em atributos simples.

Assim, teremos a seguinte tabela:

Paciente (idPaciente, cpfPaciente, nomePaciente, telefonePaciente, rua, bairro, numero, cidade, estado, dataCadastro, indicacao, dataNascPaciente)

- **Resultado**

Realizado o processo de normalização através da 1FN, tem-se que a 2FN é aplicada somente em tabelas que possuem chave primária composta e que, além disso, tenham outros atributos que não façam parte da chave primária. Desse modo, nenhuma tabela do banco de dados IFODonto se encaixa nesse requisito, tendo em vista que as tabelas Funcionario, Clinica, Dentista, Paciente e Consulta possuem chaves primárias simples e que as tabelas Procedimento e Especialidade, mesmo com chaves primárias

compostas, não possuem outros atributos não participantes da chave. Além disso, não seja possível aplicar a 3FN, já que um dos requisitos para isso é que as tabelas estejam na 2FN.

Dessa forma, conclui-se o processo de normalização do banco de dados com as seguintes tabelas:

Consulta (idConsulta, idFuncionario, idPaciente, dataConsulta, valorConsulta, formaPagamento)

idFuncionario referencia Funcionario

idPaciente referencia Paciente

Procedimento (nomeProcedimento, idConsulta)

idConsulta referencia Consulta

Clinica (cnpjClinica, nomeClinica, telClinica, rua, bairro, numero, cidade estado, idFunGerente, chaveAcesso)

idFunGerente referencia Funcionario

Funcionario (idFuncionario, cpfFunc, nomeFunc, rua, bairro, numero, cidade, estado, salário, dataNascFunc, dataAdmissao, cnpjClinica)

cnpjClinica referencia Clinica

Dentista (idFuncionario, idDentista, CRO)

idFuncionario referencia Funcionario

Especialidade (idFuncionario, nomeEspecialidade)

idFuncionario referencia Funcionario

Paciente (idPaciente, cpfPaciente, nomePaciente, telefonePaciente, rua, bairro, numero, cidade, estado, dataCadastro, indicacao, dataNascPaciente)