

RUI ROSSI DOS SANTOS

E-COMMERCE PARA EDITORAS

TOLEDO – PR

2006

RUI ROSSI DOS SANTOS

E-COMMERCE PARA EDITORAS

Projeto de software baseado em caso fictício desenvolvido para fins didáticos. Ilustra os passos necessários para a análise, a modelagem e a implementação de uma aplicação para a web utilizando a Orientação a Objetos como metodologia norteadora de todo o processo e diversas tecnologias.

TOLEDO – PR

2006

SUMÁRIO

1. ANÁLISE	1
1.1 – SISTEMA ATUAL.....	1
1.2 – SISTEMA PROPOSTO.....	1
1.2.1 – Atores	1
1.2.2 – Requisitos funcionais	1
2. MODELAGEM.....	3
2.1 – MODELAGEM ESTRUTURAL	3
2.2 – MODELAGEM COMPORTAMENTAL	5
2.3 – MODELAGEM DO BANCO DE DADOS	6
REFERÊNCIAS BIBLIOGRÁFICAS	8
ANEXO 1 – SCRIPT SQL PARA O MYSQL	9

LISTA DE TABELAS

Tabela 1 – Lista de atores	1
Tabela 2 – Relação entre casos de uso e atores	2
Tabela 3 – Dicionário de dados	4

LISTA DE FIGURAS

Figura 1 – Associações entre classes de negócio.....	3
Figura 2 – Dependências entre casos de uso.....	5
Figura 3 – Modelo conceitual de dados	6
Figura 4 – Modelo físico de dados	7

1. ANÁLISE

Neste capítulo será realizada a apresentação da empresa onde foram identificadas as necessidades a serem atendidas através da modelagem e implementação de uma nova aplicação e os requisitos funcionais a serem atendidos.

1.1 – SISTEMA ATUAL

A organização que fora tomada como base para a especificação dos requisitos do software a ser desenvolvido consiste em uma editora de livros que atualmente recebe pedidos por telefone e por e-mail e registra todas as operações relacionadas em um aplicativo desktop.

1.2 – SISTEMA PROPOSTO

Para atender às necessidades identificadas junto à editora, a aplicação a ser desenvolvida precisará atender a um conjunto de requisitos levantados. A idéia central é o desenvolvimento de uma aplicação para a web através da qual a editora possa automatizar a comercialização de seus livros. Com ela, o cliente poderá gerenciar seu cadastro, realizar buscar por livros, registrar pedidos, confirmar pagamentos e consultar seus pedidos.

1.2.1 – Atores

A Tabela 1 contém a lista de atores identificados no sistema atual e que terão papel relevante em relação ao software a ser desenvolvido, tendo em vista que deverão ter acesso a uma ou mais funções do mesmo.

Tabela 1 – Lista de atores

Ator	Atribuições
Cliente	Cadastra-se, consulta os livros, faz pedido, efetua e confirma pagamentos.
Usuário	Representa os funcionários da editora que manterão atualizados os cadastros de autores, assuntos e livros.

1.2.2 – Requisitos funcionais

Os requisitos funcionais da aplicação a ser desenvolvida constituem uma listagem completa das operações que ela precisará realizar, ou seja, das funções que nela deverão estar

disponíveis para serem utilizadas pelos seus usuários. A Tabela 2 contém a relação de todas as funções que o sistema a ser construído deverá conter.

Tabela 2 – Relação entre casos de uso e atores

Número	Descrição	Ator
1	Cadastro de usuário	Usuário
2	Cadastro de autor	Usuário
3	Cadastro de assunto	Usuário
4	Cadastro de livro	Usuário
5	Cadastro de cliente	Cliente
6	Busca de livros	Cliente
6	Registro de pedido	Cliente
7	Consulta de pedidos	Cliente
11	Relatório de usuários	Usuário
12	Relatório de autores	Usuário
13	Relatório de assuntos	Usuário
14	Relatório de livros	Usuário
15	Relatório de clientes	Usuário
16	Relatório de pedidos	Usuário
17	Gráfico de vendas por cliente	Usuário
18	Gráfico de vendas por produto	Usuário
19	Gráfico de faturamento mensal (Barras)	Usuário

Diferentes usuários do sistema terão interesse a diferentes operações do mesmo. Tendo isso em vista, fez-se a especificação do usuário que irá utilizar cada uma das suas diferentes operações.

2. MODELAGEM

Tendo sido feito o levantamento das necessidades da empresa e a especificação dos requisitos da aplicação a ser construída para atendê-las, resta ainda realizar a modelagem da mesma. É o que será feito neste capítulo através da reprodução de diagramas da UML que representam diferentes visões do software a ser desenvolvido.

2.1 – MODELAGEM ESTRUTURAL

Neste primeiro tópico, procurar-se-á realizar a exposição uma visão estrutural da aplicação a ser construída.

Figura 1 – Associações entre classes de negócio

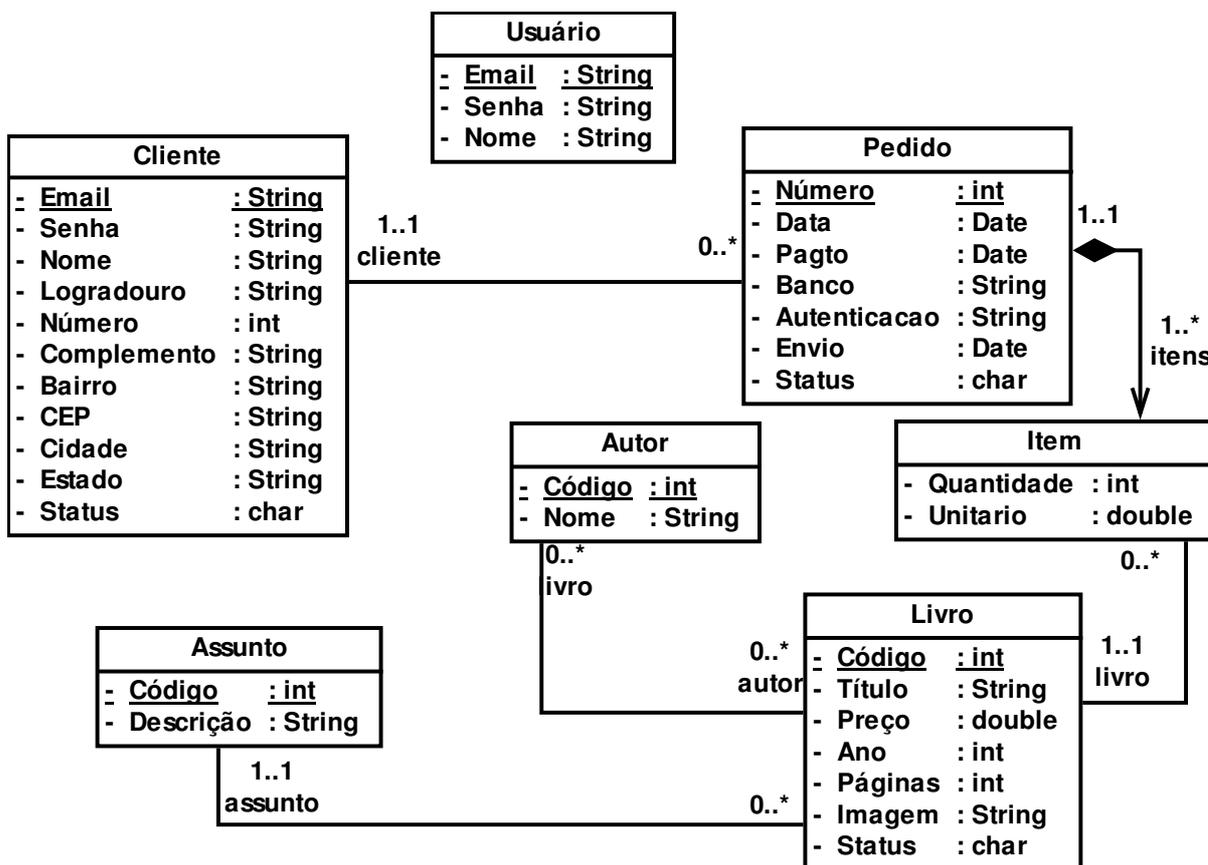


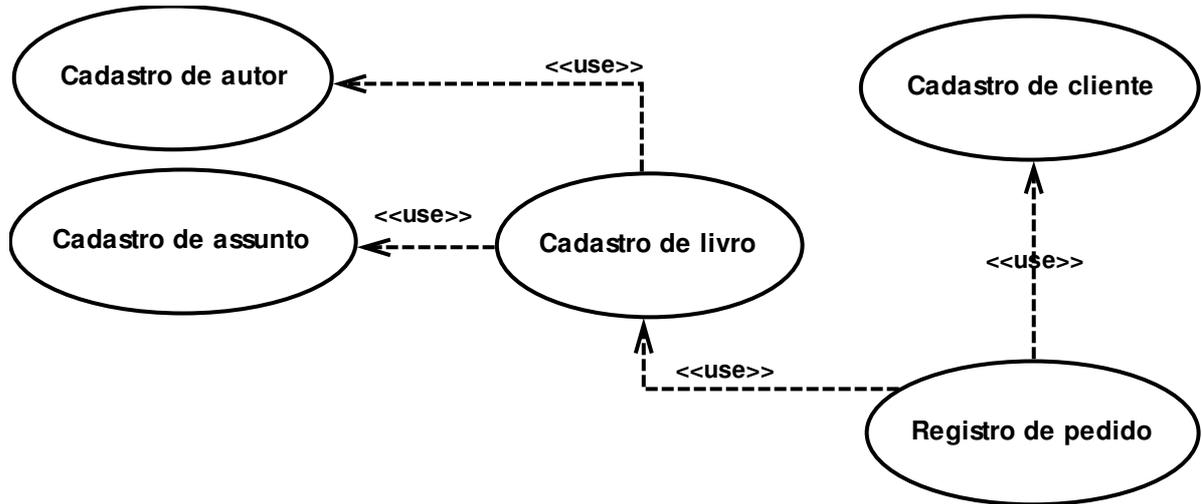
Tabela 3 – Dicionário de dados

Classe	Atributo	Descrição
Usuário	Email	O endereço eletrônico a ser utilizado como login
	Senha	A senha a ser utilizada para acessar o sistema
	Nome	O nome completo do operador
Assunto	Código	Um número inteiro positivo para identificar o assunto
	Descrição	A descrição textual do assunto
Autor	Código	Um número inteiro positivo para identificar o autor
	Nome	O nome completo do autor
Livro	Código	Um número inteiro positivo para identificar o livro
	Título	O título completo do livro
	Preço	O preço de venda atual do livro
	Ano	O ano de publicação do livro
	Páginas	O número total de páginas do livro
	Imagem	Caminho e nome do arquivo com uma imagem da capa do livro
	Status	D = Disponível em estoque E = Esgotado
Cliente	Email	O endereço eletrônico a ser utilizado como login
	Senha	A senha a ser utilizada para acessar o sistema
	Nome	O nome completo do cliente
	Logradouro	O tipo e o nome do logradouro em que reside
	Número	O número da residência do cliente
	Complemento	Um complemento ao endereço
	Bairro	O nome do bairro
	CEP	O código de endereço postal
	Cidade	O nome da cidade
	Estado	A sigla do estado
	Status	A = Ativo B = Bloqueado
Pedido	Número	Um número inteiro positivo para identificar o pedido
	Data	A data de registro do pedido
	Pagto	A data em que o cliente efetuou o pagamento do pedido através de depósito ou transferência bancária
	Banco	O banco para o qual o cliente transferiu o valor do pagamento
	Autenticação	Os dados relativos à autenticação do depósito ou da transferência bancária realizada pelo cliente para o pagamento do pedido
	Envio	A data em que o pedido foi despachado para o cliente
	Status	A = Aguardando pagamento P = Pedido pago pelo cliente D = Pedido despachado para o cliente
Item	Quantidade	A quantidade de cada livro incluso no pedido
	Unitario	O preço cobrado por cada unidade dos livros

2.2 – MODELAGEM COMPORTAMENTAL

A Figura 2 contém um diagrama de caso de uso que ilustra as dependências existentes entre as diversas operações da aplicação a ser desenvolvida.

Figura 2 – Dependências entre casos de uso



2.3 – MODELAGEM DO BANCO DE DADOS

Figura 3 – Modelo conceitual de dados

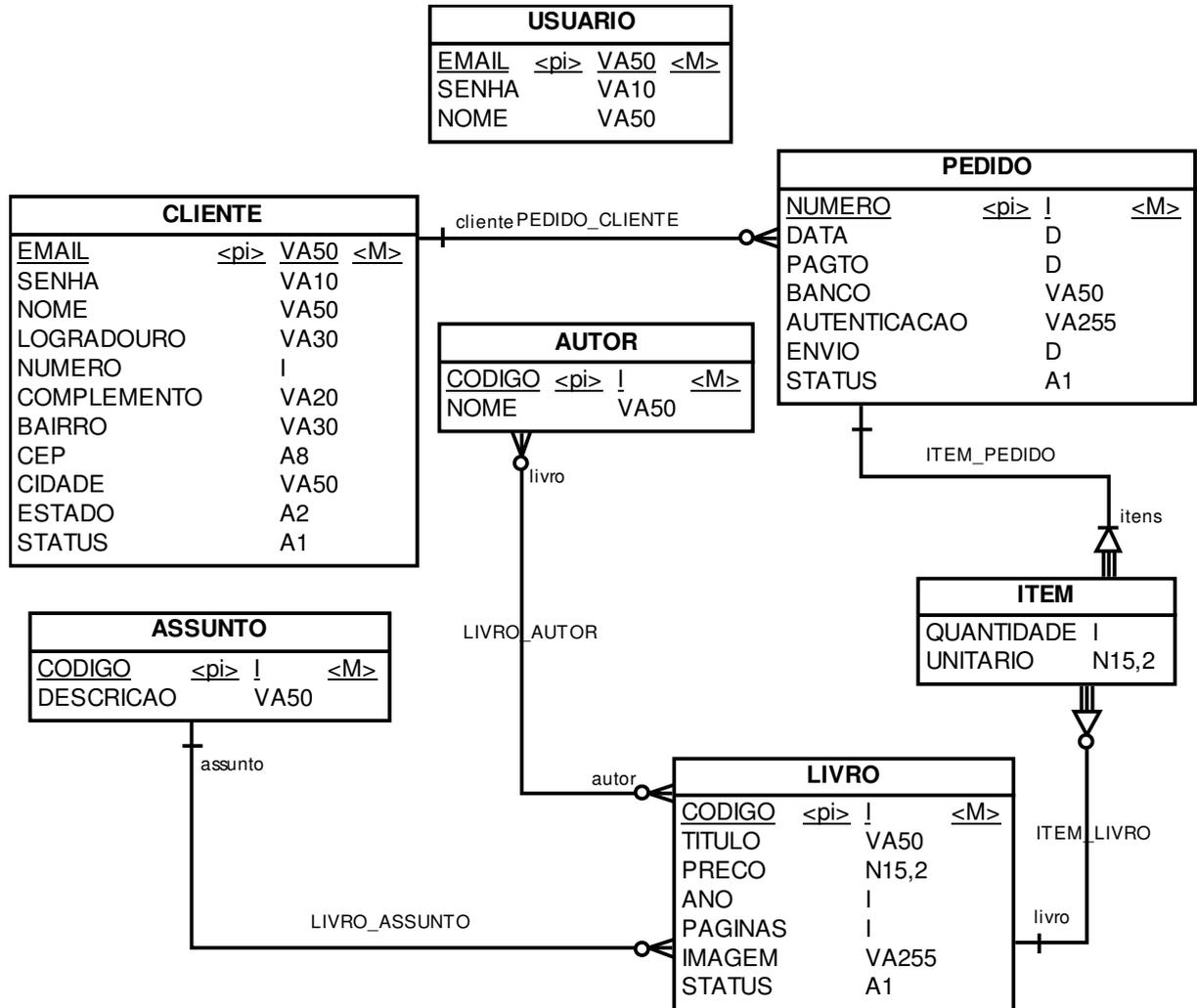
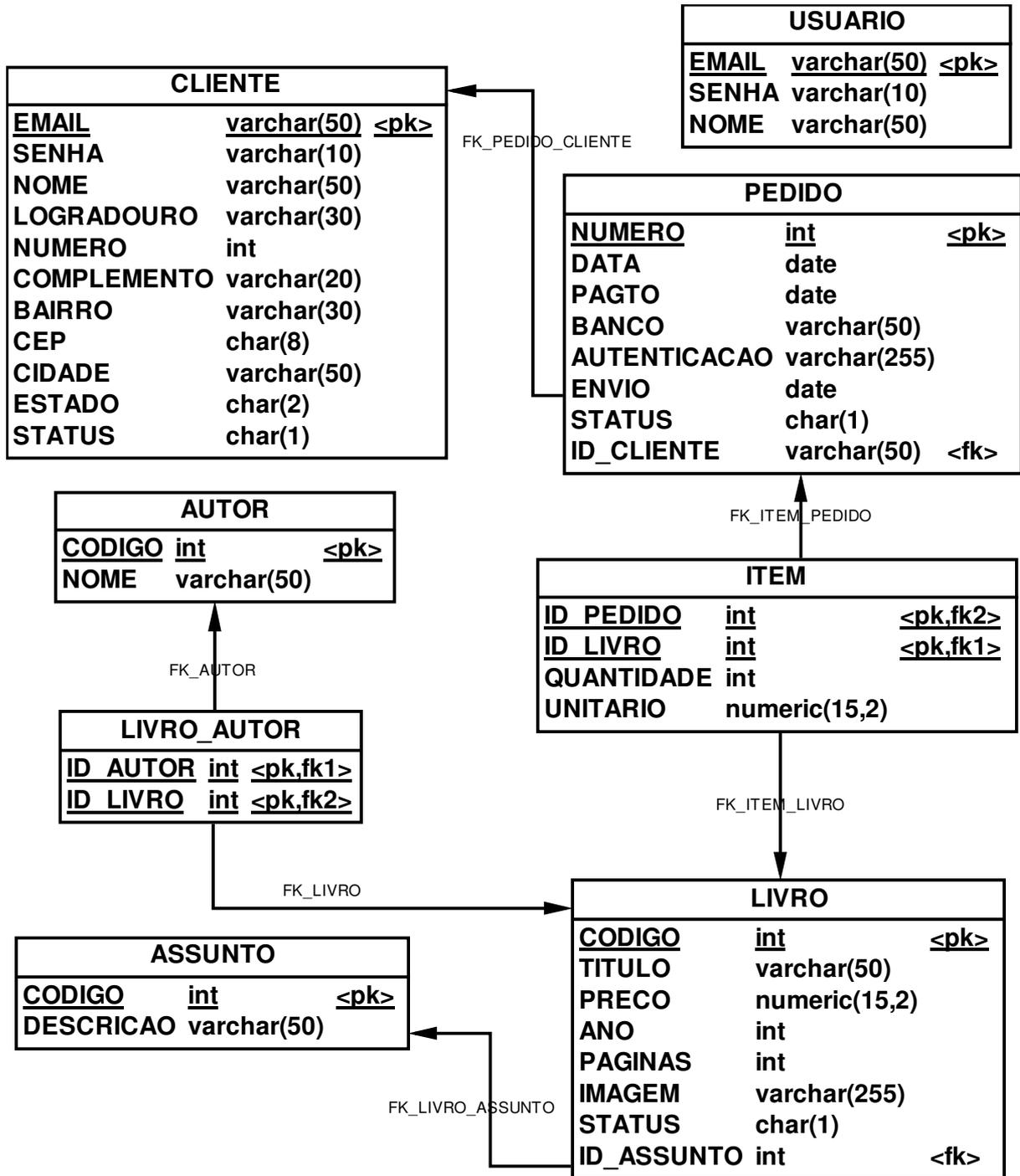


Figura 4 – Modelo físico de dados



REFERÊNCIAS BIBLIOGRÁFICAS

- BOOCH, Grady; RUMBAUGH, James; JACOBSON, Ivar. **UML: guia do usuário**. Rio de Janeiro: Campus, 2000.
- FURLAN, José Davi. **Modelagem de objetos através da UML: the unified modeling language**. São Paulo: Makron Books, 1998.

ANEXO 1 – SCRIPT SQL PARA O MYSQL

```
1. CREATE TABLE USUARIO (
2.     EMAIL                VARCHAR(50)                NOT NULL,
3.     SENHA                VARCHAR(10),
4.     NOME                 VARCHAR(50),
5.     PRIMARY KEY (EMAIL)
6. ) TYPE = INNODB;
7.
8. CREATE TABLE CLIENTE (
9.     EMAIL                VARCHAR(50)                NOT NULL,
10.    SENHA                VARCHAR(10),
11.    NOME                 VARCHAR(50),
12.    LOGRADOURO          VARCHAR(30),
13.    NUMERO              INT,
14.    COMPLEMENTO        VARCHAR(20),
15.    BAIRRO              VARCHAR(30),
16.    CEP                 CHAR(8),
17.    CIDADE              VARCHAR(50),
18.    ESTADO              CHAR(2),
19.    STATUS              CHAR(1),
20.    PRIMARY KEY (EMAIL)
21. ) TYPE = INNODB;
22.
23. CREATE TABLE AUTOR (
24.     CODIGO              INT                NOT NULL,
25.     NOME                VARCHAR(50),
26.     PRIMARY KEY (CODIGO)
27. ) TYPE = INNODB;
28.
29. CREATE TABLE ASSUNTO(
30.     CODIGO              INT                NOT NULL,
31.     DESCRICAO          VARCHAR(50),
32.     PRIMARY KEY (CODIGO)
33. ) TYPE = INNODB;
34.
35. CREATE TABLE LIVRO (
36.     CODIGO              INT                NOT NULL,
37.     TITULO             VARCHAR(50),
38.     PRECO              NUMERIC(15,2),
39.     ANO                INT,
40.     PAGINAS           INT,
41.     IMAGEM            VARCHAR(255),
42.     STATUS            CHAR(1),
43.     ID_ASSUNTO        INT                NOT NULL,
44.     PRIMARY KEY (CODIGO)
```

```

45. ) TYPE = INNODB;
46.
47. CREATE TABLE LIVRO_AUTOR (
48.     ID_AUTOR                INT                NOT NULL,
49.     ID_LIVRO                INT                NOT NULL,
50.     PRIMARY KEY (ID_AUTOR, ID_LIVRO)
51. ) TYPE = INNODB;
52.
53. CREATE TABLE PEDIDO (
54.     NUMERO                    INT                NOT NULL,
55.     DATA                    DATE,
56.     PAGTO                     DATE,
57.     BANCO                     VARCHAR(50),
58.     AUTENTICACAO             VARCHAR(255),
59.     ENVIO                     DATE,
60.     STATUS                    CHAR(1),
61.     ID_CLIENTE                VARCHAR(50)       NOT NULL,
62.     PRIMARY KEY (NUMERO)
63. ) TYPE = INNODB;
64.
65. CREATE TABLE ITEM (
66.     ID_PEDIDO                 INT                NOT NULL,
67.     ID_LIVRO                 INT                NOT NULL,
68.     QUANTIDADE                INT,
69.     UNITARIO                  NUMERIC(15,2),
70.     PRIMARY KEY (ID_PEDIDO, ID_LIVRO)
71. ) TYPE = INNODB;
72.
73. CREATE INDEX LIVRO_ASSUNTO_FK ON LIVRO(ID_ASSUNTO);
74. CREATE INDEX LIVRO_FK ON LIVRO_AUTOR(ID_AUTOR);
75. CREATE INDEX AUTOR_FK ON LIVRO_AUTOR(ID_LIVRO);
76. CREATE INDEX PEDIDO_CLIENTE_FK ON PEDIDO(ID_CLIENTE);
77. CREATE INDEX ITEM_PEDIDO_FK (ID_PEDIDO);
78. CREATE INDEX ITEM_LIVRO_FK ON ITEM(ID_LIVRO);
79.
80. ALTER TABLE ITEM ADD CONSTRAINT FK_ITEM_LIVRO FOREIGN KEY (ID_LIVRO)
81.     REFERENCES LIVRO (CODIGO) ON DELETE RESTRICT ON UPDATE RESTRICT;
82.
83. ALTER TABLE ITEM ADD CONSTRAINT FK_ITEM_PEDIDO FOREIGN KEY (ID_PEDIDO)
84.     REFERENCES PEDIDO (NUMERO) ON DELETE RESTRICT ON UPDATE RESTRICT;
85.
86. ALTER TABLE LIVRO ADD CONSTRAINT FK_LIVRO_ASSUNTO FOREIGN KEY (ID_ASSUNTO)
87.     REFERENCES ASSUNTO (CODIGO) ON DELETE RESTRICT ON UPDATE RESTRICT;
88.
89. ALTER TABLE LIVRO_AUTOR ADD CONSTRAINT FK_AUTOR FOREIGN KEY (ID_AUTOR)
90.     REFERENCES AUTOR (CODIGO) ON DELETE RESTRICT ON UPDATE RESTRICT;
91.
92. ALTER TABLE LIVRO_AUTOR ADD CONSTRAINT FK_LIVRO FOREIGN KEY (ID_LIVRO)

```

```
93. REFERENCES LIVRO (CODIGO) ON DELETE RESTRICT ON UPDATE RESTRICT;
94.
95. ALTER TABLE PEDIDO ADD CONSTRAINT FK_PEDIDO_CLIENTE FOREIGN KEY (ID_CLIENTE)
96. REFERENCES CLIENTE (EMAIL) ON DELETE RESTRICT ON UPDATE RESTRICT;
```