≪ Share

Last indexed: 11 August 2025 (172577)

Overview

Getting Started

(Proy1)

Development Environment

(Proy3_Pedidos)

Environment & Configuration

VS Code & Debugging

Dependency Management

Development Environment

Sources: Proy3_Pedidos/def proyecto 1-570

processing.

Overview

multi-entity schema supporting clients, restaurants, and platform administrators. The schema implements a comprehensive food delivery workflow from user registration through order fulfillment.

The FoodExpress platform uses a MySQL database with Sequelize ORM to manage a complex

This document covers the database architecture and Sequelize model implementations for the

schema structure that supports multi-role user management, restaurant operations, and order

FoodExpress food delivery platform. It details the entity relationships, model definitions, and

For information about the authentication and middleware components that work with these

models, see <u>4.2</u>. For details about the overall architecture and dependencies, see <u>4.1</u>.

Database Technology Stack

Database Models & Schema

> Relevant source files

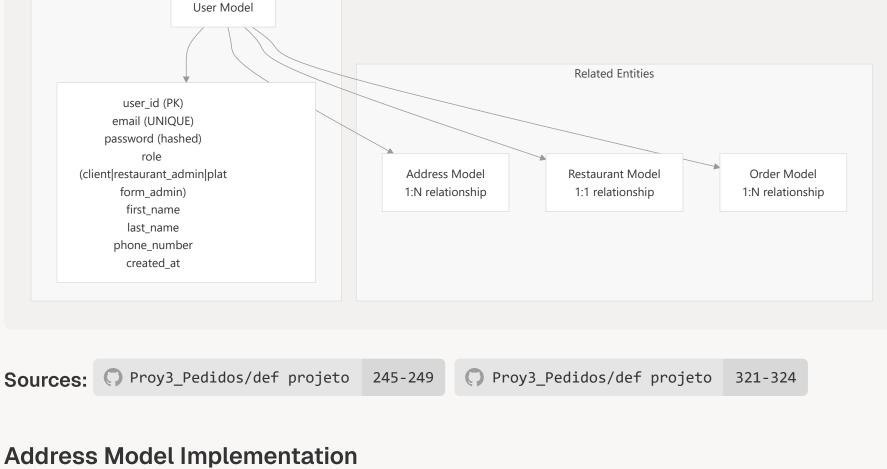
The platform employs the following data persistence technologies:

Component Technology Purpose Primary data storage Database Engine MySQL

ORM	Sequelize		Object-relational map	Object-relational mapping	
Model Configuration	underscored naming		Database field conve	Database field conventions	
Timestamps	Automatic		Created/updated tracking		
Associations	Sequelize relation	ships	Foreign key management		
Sources: Proy3_Pedidos/	models/address.js	1-68	<pre>Proy3_Pedidos/def projeto</pre>	146-147	
Core Entity Models					

User Entity

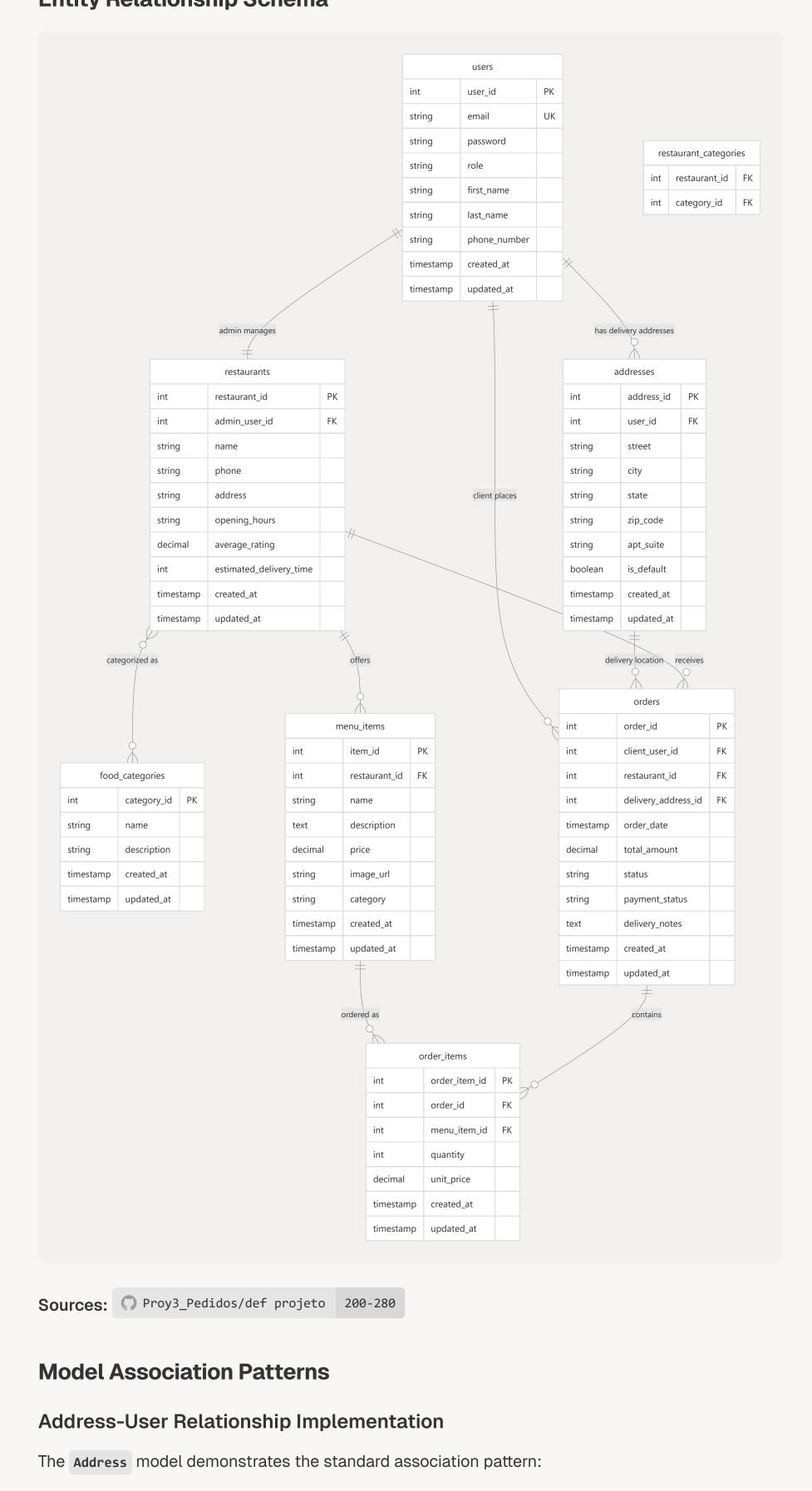
User Model Structure



The Address model demonstrates the Sequelize implementation pattern used throughout the platform:

Address Model Structure

Address.init() addressId: INTEGER PK userld: INTEGER FK street: STRING tableName: 'addresses' belongsTo(User) city: STRING timestamps: true foreignKey: 'user_id' state: STRING underscored: true as: 'user' zipCode: STRING(20) inCascade: true aptSuite: STRING (optional) isDefault: BOOLEAN Sources: Proy3_Pedidos/models/address.js 19-67 **Entity Relationship Schema**



Foreign Key userId: DataTypes.INTEGER References users.user_id **Association Method** belongsTo(models.User) Alias: 'user' Cascade Behavior inCascade: true Automatic cleanup Sources: Proy3_Pedidos/models/address.js 10-16 Proy3_Pedidos/models/address.js 25-32 Many-to-Many Resolution The restaurant-category relationship uses a junction table:

Many-to-Many

Implementation

Implementation

Restaurant Model

belongsToMany

Configuration

FoodCategory Model

belongsToMany

Relationship Type

restaurant_categories (Junction Table) restaurant_id FK category_id FK Sources: Proy3_Pedidos/def projeto 222-234 Proy3_Pedidos/def projeto 259-265 **Order Processing Data Flow** Order Creation Flow express-session cart Order.create() Status Management payment_status **Order Status States** OrderItem.create()

paid | pending | cash

Proy3_Pedidos/def projeto 270-280

Snake_case field names

express-validator

Proy3_Pedidos/def projeto 399-400

Database Constraints

total_amount calculation

pending accepted preparing

ready_for_pickup

 $in_transit$ delivered cancelled

Historical Preservation

unit_price in order_items

Price at order time

underscored

Configuration Value Purpose Automatic created_at / updated_at timestamps true

true

All models follow consistent configuration patterns:

Sources: Proy3_Pedidos/def projeto 40-61

Sequelize Configuration Standards

tableName	Explicit string	Override pluralization		
inCascade	true	Enable cascade operations		
Sources:	<pre>Proy3_Pedidos/models/address.js</pre>	58-65		
Model Validation Patterns				

Address Model Example

Sources: Proy3_Pedidos/models/address.js 25-57

The schema implements validation at multiple levels:

DataTypes.STRING allowNull: false Foreign Key References defaultValue: false DataTypes.INTEGER DataTypes.BOOLEAN

Validation Layers

Sequelize Model Validations

Repository Structure Project 1: Product Management Database & Configuration Product Management System Project 2: Course Platform (Proy2_Cursos) Data Models & Database Authentication & Middleware Development & Testing Project 3: Food Delivery Platform Architecture & Dependencies Database Models & Schema