E-COMMERCE SITE PROPOSAL

Harvest Natural Foods Ltd.

NOVEMBER 6, 2025

FENG LI
Winnipeg, Manitoba

CONTENTS

Executive Summary

This proposal outlines a comprehensive plan to develop a fully-featured Ruby on Rails e-commerce platform for Harvest Natural Foods Ltd. The proposed solution will provide a robust online storefront with complete product management, secure checkout processing, and comprehensive order tracking capabilities.

Description of the E-commerce Site

Business Overview

Harvest Natural Foods Ltd.

Business Description

Harvest Natural Foods is a well-established Winnipeg-based retailer of organic and natural foods, serving the Manitoba community for 15 years. The company currently operates two brick-and-mortar locations in Winnipeg with a team of 23 employees, including store managers, nutritionists, and customer service representatives.

- **Current Operations:**

Harvest currently sells its products exclusively through its physical retail locations. They offer a diverse range of organic produce, natural supplements, eco-friendly household products, and specialty dietary items. While they have built a loyal customer base through their in-store experience, they recognize the growing need to expand into the online marketplace to remain competitive and serve customers who prefer the convenience of home delivery.

- **Products for Online Sale:**

The online store will feature its complete product catalogue, including:

- Organic fresh produce and frozen foods
- Natural vitamins and supplements
- Gluten-free and allergen-friendly products
- Eco-friendly household and personal care items
- Specialty teas, coffees, and health beverages

- Organic snacks and packaged goods

- **Target Demographic:**

The primary target demographic for this online store includes:

- **Age Range:** 25-55 years old
- **Location:** Primarily Winnipeg and surrounding Manitoba communities, with potential expansion to other Canadian provinces
- **Lifestyle:** Health-conscious individuals and families who prioritize organic, sustainable, and ethically-sourced products
- **Income Level:** Middle to upper-middle class households with disposable income for premium natural products
- **Shopping Behavior:** Busy professionals and parents who value the convenience of online shopping and home delivery
- **Values:** Environmental sustainability, health and wellness, and supporting local businesses

Database Structure Description

The proposed database structure is designed to handle complex e-commerce operations while maintaining data integrity and supporting future scalability. The schema accounts for product management, customer orders with multiple items, quantity tracking, historical price preservation, and tax calculation.

Database Tables

1. users

Store customer account information and authentication credentials.

Column Name	Data Type	Notes
id	integer	Primary key
email	string	Unique, required
encrypted_password	string	Hashed password
username	string	Unique identifier
created_at	datetime	
updated_at	datetime	

2. provinces

Store Canadian provinces and territories with their respective tax rates.

Column Name	Data Type	Notes
id	integer	Primary key
name	string	Province/territory name
code	string	Two-letter code (e.g., MB)
gst_rate	decimal(5,2)	GST percentage
pst_rate	decimal(5,2)	PST percentage
hst_rate	decimal(5,2)	HST percentage
created_at	datetime	
updated_at	datetime	

3. addresses

Store customer shipping and billing addresses.

Column Name	Data Type	Notes
id	integer	Primary key
user_id	integer	Foreign key to users
province_id	integer	Foreign key to provinces
street_address	string	
city	string	
postal_code	string	
address_type	string	'shipping' or 'billing'
created_at	datetime	
updated_at	datetime	

4. categories

Organize products into logical groupings for navigation and filtering.

Column Name	Data Type	Notes
id	integer	Primary key
name	string	Category name
description	text	Category description
created_at	datetime	
updated_at	datetime	

5. products

Store product information and current pricing.

Column Name	Data Type	Notes
id	integer	Primary key
name	string	Product name
description	text	Product details
price	decimal(10,2)	Current price
stock_quantity	integer	Inventory count
on_sale	boolean	Sale status flag
is_new	boolean	New product flag
created_at	datetime	
updated_at	datetime	

6. product_categories

Join table to support many-to-many relationship between products and categories (allows products to belong to multiple categories).

Column Name	Data Type	Notes
id	integer	Primary key
product_id	integer	Foreign key to products
category_id	integer	Foreign key to categories
created_at	datetime	
updated_at	datetime	

7. orders

Store customer order headers with totals and status information.

Column Name	Data Type	Notes
id	integer	Primary key
user_id	integer	Foreign key to users
address_id	integer	Foreign key to addresses
status	string	'pending', 'paid', 'shipped'
subtotal	decimal(10,2)	Pre-tax total
gst_amount	decimal(10,2)	GST charged
pst_amount	decimal(10,2)	PST charged
hst_amount	decimal(10,2)	HST charged
grand_total	decimal(10,2)	Final total with taxes
stripe_payment_id	string	Payment processor reference
created_at	datetime	Order date
updated_at	datetime	

8. order_items

Store individual products within each order with historical pricing.

Column Name	Data Type	Notes
id	integer	Primary key
order_id	integer	Foreign key to orders
product_id	integer	Foreign key to products
quantity	integer	Number of units ordered
price_at_purchase	decimal(10,2)	Product price when ordered
subtotal	decimal(10,2)	quantity × price_at_purchase
created_at	datetime	
updated_at	datetime	

9. payments

Store payment history.

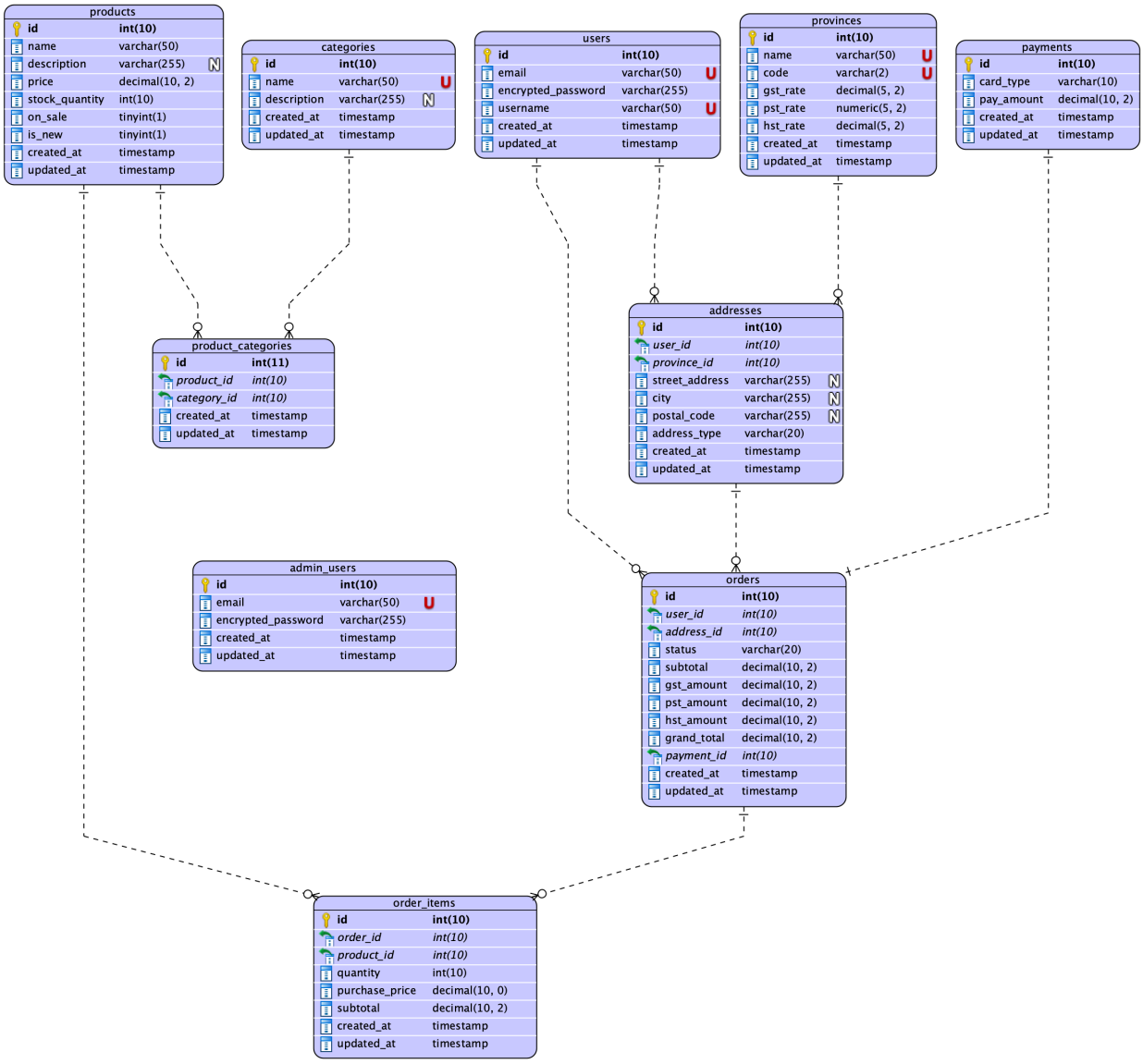
Column Name	Data Type	Notes
id	integer	Primary key
card_type	string	‘credit’ or ‘debit’
pay_amount	decimal(10,2)	
created_at	datetime	
updated_at	datetime	

10. admin_users

Store administrator credentials should be separate from customer accounts.

Column Name	Data Type	Notes
id	integer	Primary key
email	string	Unique, required
encrypted_password	string	Hashed password
created_at	datetime	
updated_at	datetime	

Entity Relationship Diagram



One-to-Many Relationships:

- **users → addresses**: A user can have multiple addresses (one for shipping, one for billing)
- **users → orders**: A user can place multiple orders over time
- **provinces → addresses**: A province is referenced by many addresses
- **orders → order_items**: An order contains multiple line items
- **products → order_items**: A product can appear in many order items

Many-to-Many Relationships:

- **products ↔ categories**: Implemented via join table product_categories (allows products to belong to multiple categories)