Gramp's Market

Web Design Proposal

David Sawatzky

Proposal for Gramp's Market

I would like to offer this web design proposal to "Gramp's Market", an independent grocery store that specializes in locally grown fresh fruit and vegetables. The store is family owned and operated with 15 employees. It's been in operation for 22 years and has a stable and loyal customer base. Some customers drive in from as far as 75km away to shop at the store.

The store maintains a unique place in the grocery business. As an independent grocer, it is able to select produce that is more in season, local, and of better quality than other stores where produce is must be shipped in bulk from long distances. As such, while it is not as cheap as other stores, demand for their produce is still high.

The goal for the e-commerce store is to increase the client base and be a more valuable part of people grocery needs by offering to ship produce directly to the customer, as well as increase the Market's visibility by doing their own deliveries. The idea is to deliver to anywhere within a 100km radius of the store within a 4-day delivery window.

With this e-commerce website, I have every confidence that the goals for Gramp's Market will be achieved, and I look forward to developing this with you.

Database Design

Products

A table of all the products sold in the store along with prices, inventory, SKU number, and if applicable, how long it's been on the floor. It also has two boolean attributes for indicating if it's taxable or not.

Orders

A record of any placed orders. Each order can have many products associated with it which will then be shipped as one complete order. The total cost of the order is given here.

Order Products

This is a many-to-many table with products and orders. Product quantities, subtotals, and applicable taxes are calculated and inserted here. This way a change in price in the products table shouldn't affect the price in the order.

Deliveries

The delivery date as well as a record of the driver and truck used to make the delivery.

Customers

The customer submitting the order and receiving the delivery. It also records the distance from the store for delivery planning purposes.

Database Structure

OrderProducts

PK	id
FK	order_id
FK	product_id
	quantity: int
	price_per_item: float
	subtotal: float
	GST: float
	PST: float

Products

PK	id
	name: string
	type: enum
	inventory: int
	floorDate: string
	SKU: string
	price: float
	taxable-GST: boolean
	taxable-PST: boolean
	details: string

Orders

PK	id
FK	date_submitted: string
	date_delivered: string
FK	deliveryId: int
FK	customerId: int
	subtotal: float
	shippingCost: float
	totalGST: float
	totalPST: float
	totalCost: float
	paid: boolean
	transaction_number: string
	delivered: boolean

Deliveries

PK	id
	driver: string
	truck_number: string
	delivery_date: string
	notes: text

Customers

PK	id
	first_name: string
	last_name: string
	apt_number: string
	number: string
	street: string
	city: string
	postal_code: string
	phone: string
	email: string
	distance: float