CSC 316

INTRODUCTION TO SYSTEM ANALYSIS AND DESIGN

TITLE: *Analysis and Design of an E-commerce (ordering, billing, inventory and accounting) System for Yem Yem Supermarket*

	NAME	MATRIC NO.	ROLE
1.	OLORUNFEMI-OJO, Daniel Tomiwa	190805503	Senior Systems Analyst/ Designer
2.	ANIMASHAUN, Sofiat Arike	190805520	Secretary/ Logistics

2021/2022 SESSION

TABLE OF CONTENTS

4.0 Conclusion

APPENDICES: Sample codes if any, reports, results, inputs and outputs, etc.

1.0 INTRODUCTION

1.1. Background of the Study

In the modern day shopping business paradigm, it is extremely important to have an online presence on the internet where customers can browse the stores catalog and purchase products of interest. The existing commerce system at Yem Yem is limited to the physical store. It is good but we live in a digital world with digital marketplaces. This document discusses the underlying technologies used to design an E-commerce system (ordering products via the internet, billing customers, assessment of inventory and valuation of sales via the system) for Yem Yem supermarket.

1.2. Statement of Problem

The existing system at Yem Yem

- For ordering: physical purchase
- For payment: cash, bank transfer, POS
- For inventory control: physical count & QuickBooks Desktop Accounting software
- For accounting: QuickBooks Desktop Accounting software

Most of Yem Yem's repeat customers are people who live close by. In this age of technology, purchase of goods should not be limited by proximity. Having only a physical purchase system limits the customer base.

Inventory management is performed via physical count and the use of Quickbooks Desktop Accounting software. The Quickbooks program is also used for accounting purposes. Physically counting inventory is prone to human errors. With the desktop

version, real time access to inventory and accounting records cannot be accessed anywhere.

1.3. Objectives

- Introduce a business model of buying goods over the internet
- Offer a variety of payment solutions
- Balance the rate of demand for goods with inventory stocking
- Accumulate financial information about the performance of the system

1.4. Project's Scope and Constraints

Scope

The proposed system's scope includes

- A Business to Client (B2C) ordering and billing system. The system will not compute VAT on products i.e. Yem Yem are left with the responsibility of adding VAT to selling price of items.
- An administration panel for inventory management and accounting.

Constraints

Cost of implementation of the Information System is high.

1.5. Alternative Solutions

Pre-packaged e-commerce solutions like Shopify and WooCommerce that have the capability of synchronizing with the QuickBooks API/SDK to fetch and update inventory.

1.6. System Description

The E-commerce System is a Web Application (accessible on both Desktop and Mobile) that synchronizes and interacts with products via Quickbooks API/SDK. The system

provides its users with the ability to order groceries from the comfort of their homes and select a preferred payment option (strictly B2C). The system also has an administration panel, which offers creation and management of products and product categories, inventory management and provides statement of cash flow all of which are gotten via the QuickBooks API/SDK. Only Yem Yem management will have access to the administration panel.

1.7. Feasibility Assessment

1.7.1. Economic Analysis

Using Break-Even analysis, the economic feasibility study shows that embarking on this project will be worth it because in less than a year, the benefits will outweigh the cost of embarking on the project.

$$Break\ Even\ Ratio = \frac{Yearly\ NPV\ Cash\ Flow - Overall\ NPV\ Cash\ Flow}{Yearly\ NPV\ Cash\ Flow}$$

	A	B	C	D	E	F	H
1	COMPANY NAME	YEM YEM SUPERMARKET					
2	PROPOSED PROJECT	E-COMMERCE SYSTEM					
3							
4		**YEAR 0**	**YEAR 1**	**YEAR 2**	**YEAR 3**	**YEAR 4**	**TOTAL**
5	NET ECONOMIC BENEFITS (₦)	0.00	28,800,000.00	45,600,000.00	62,400,000.00	72,000,000.00	
6	DISCOUNT RATE(10%)	1	0.909	0.826	0.7513	0.683	
7	PV OF BENEFITS (₦)	0.00	26,179,200.00	37,665,600.00	46,881,120.00	49,176,000.00	
8							
9	NPV OF ALL BENEFITS (₦)	0.00	26,179,200.00	63,844,800.00	110,725,920.00	159,901,920.00	**159,901,920.0**
10							
11	ONE TIME COSTS (₦)	1,544,200.00					
12	RECURRING COSTS (₦)	0.00	1,204,200.00	1,204,200.00	1,204,200.00	1,204,200.00	
13	DISCOUNT RATE(10%)	1	0.909	0.826	0.7513	0.683	
14	PV OF RECURRING COSTS (₦)	0.00	1,094,617.80	994,669.20	904,715.46	822,468.60	
15	NPV OF ALL COSTS (₦)	1,544,200.00	2,638,817.80	3,633,487.00	4,538,202.46	5,360,671.06	**5,360,671.06**
16							
17	OVERALL NPV (₦)						154,541,248.9
18							
19	OVERALL ROI (OVERALL NPV/NPV OF ALL COSTS)						28.8
20							
21	BREAK-EVEN ANALYSIS						
22	YEARLY NPV CASHFLOW	1,544,200.00	25,084,582.20	36,670,930.80	45,976,404.54	48,353,531.40	
23	OVERALL NPV CASHFLOW	1,544,200.00	23,540,382.20	60,211,313.00	106,187,717.54	154,541,248.94	
24	Project break even occurs between Year 0 and 1 = (25,084,582.20 - 23,540,382.20) / 25,084,582.20 = 0.061						
25	**Break-Even occurred at 0.061 years**						

1.7.2. Technical Analysis

The operational requirements for the E-commerce system are:

- **Hardware (minimum):**
    - Memory: 2GB
    - Processors: Intel Dual Xenon
    - Storage: 20GB SSD
    - Bandwidth: 1TB
- **Software:**
    - Relational Database Management System: MySQL
    - Backend (application logic and REST service): PHP
    - Frontend (visual elements): HTML, CSS & Javascript
    - Web Server: Apache/Nginx
    - Server OS: Linux (Ubuntu 18.04+/Debain 8+)
    - Payment Gateway: Paystack, Flutterwave
    - QuickBooks Online & API/SDK

Upon conducting excessive research on the operational requirements, the project is technically feasible reasons being

- LAMP (Linux, Apache, MySQL & PHP) stack is an ideal prototype for creating an E-commerce system. This stack is widely used, properly documented and has massive support.
- Paystack and Flutterwave do not require service payment but rather charge a rate of 1.4% - 3.8% on transactions, which is capped at NGN 2000.
- Yem Yem already use the desktop version of Quickbooks. The process of migrating from the desktop version to the online version is a matter of a few button clicks and takes only a few minutes.

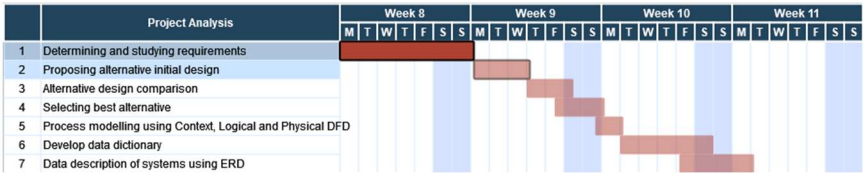
- The hardware stack is popular and readily available.

Due to the simplicity of operation and its low risk, the project is technically worth undertaking.

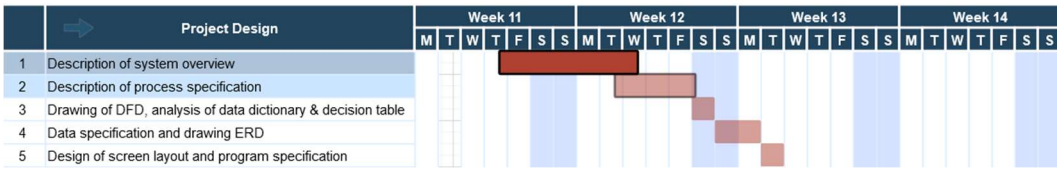
1.7.3. Schedules, Timeline and Resource Analysis

Schedules

	Project Planning	Duration	Start	ETA
1	Creation of project team	1 day	07.12.21	07.12.21
2	Establishing relationship with the case study	35 days	08.12.22	25.01.22
3	Approval of project topic	1 day	25.01.22	25.01.22
4	Developing project initiation plan	4 days	26.01.22	29.01.22
5	Describing project scope, alternatives and feasibility	4 days	30.01.22	02.02.22



	Project Analysis
1	Determining and studying requirements
2	Proposing alternative initial design
3	Alternative design comparison
4	Selecting best alternative
5	Process modelling using Context, Logical and Physical DFD
6	Develop data dictionary
7	Data description of systems using ERD



	Project Design
1	Description of system overview
2	Description of process specification
3	Drawing of DFD, analysis of data dictionary & decision table
4	Data specification and drawing ERD
5	Design of screen layout and program specification

Timeline Analysis

Using the Program Evaluation and Review Technique **PERT**, (a three point activity estimating technique that considers estimation uncertainty and risk by using three estimates to define an approximate probability for an activity's duration), we have deduced the following about the project's timeline:

S/N	Activity	O (wks)	R (wks)	P (wks)	ET (wks)
1.	Project Planning	1	3	8	7

2.	Analysis of Existing System	1	2	3	2
3.	Design of the Proposed System	4	5	6	5
	TOTAL	6	10	14	14

$$\boldsymbol{ET} = \frac{(O + (4R) + P)}{6}$$

$$where\ \boldsymbol{O} = optimistic\ time, \boldsymbol{R} = realistic\ time, \boldsymbol{P} = pessimistic\ time,$$

$$\boldsymbol{ET} = estimated\ time$$

Resource Analysis

In order to accomplish this project, humans as a resource and certain material resources are needed. Human resources, inferring the man power, individuals or the project team involved in building the proposed system. The material resources required are the equipment such as computers which each team member is expected to have one project or no project, internet connectivity of which an arrangement can be made amongst the team.

1.8. Recommendation

From experience, Zube recommends the E-commerce system should be deployed using foreign hosting companies instead of Nigeria-based hosting companies. Most of the local hosting companies cannot guarantee 99.9% uptime and those who can are very expensive. The foreign companies offer cheap and scalable plans with high bandwidth. In addition, the system and its user's security should be of major concern account while developing the system. The customers should not be exposed to a system that is

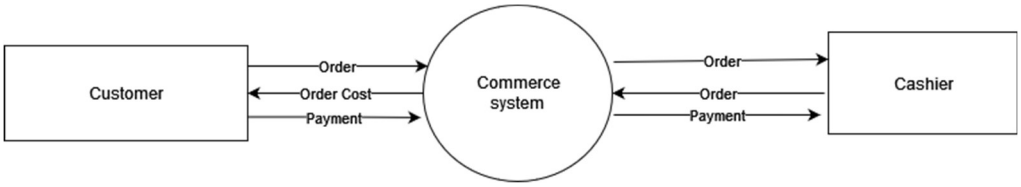
vulnerable to cyber attacks. Input data should be filtered input to prevent Cross Site Scripting and SQL injection attacks. All forms posting data to the registered domain should contain an authentication token attached to the user's session to prevent Cross Site Request Forgery.

2.0 ANALYSIS OF THE EXISTING SYSTEM

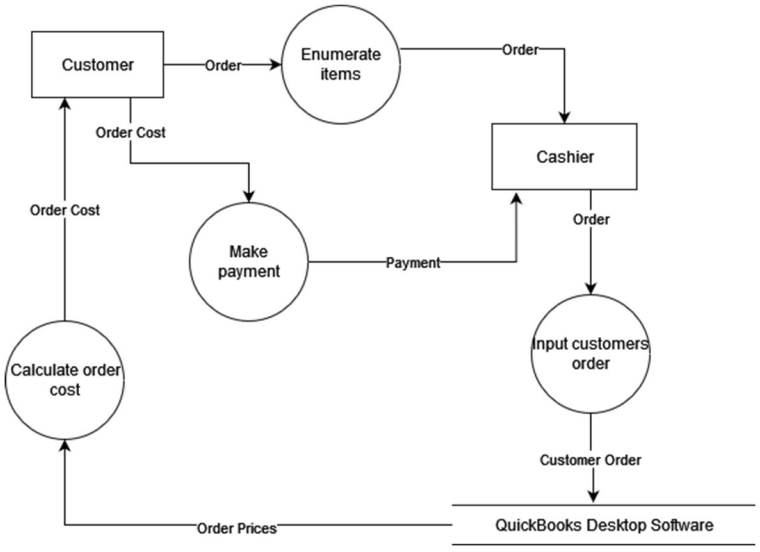
2.1. Process Description of the System(using context and DFDs)

The existing commerce system at Yem Yem entails the customer physically going to the store and selecting items to be purchased. The order is documented using QuickBooks desktop software. Upon submission, the inventory size updates on the QuickBooks desktop software by subtracting the quantities in the order from the inventory size. The customer proceeds to pay via bank transfer, POS terminal or cash. A receipt is printed for the customer. Besides tracking inventory with the Quickbooks software, inventory size of some items are counted manually on a daily or weekly basis.

2.1.1. Current Physical Data Flow Diagram (CPDFD)



2.1.2. Current Logical Data Flow Diagram



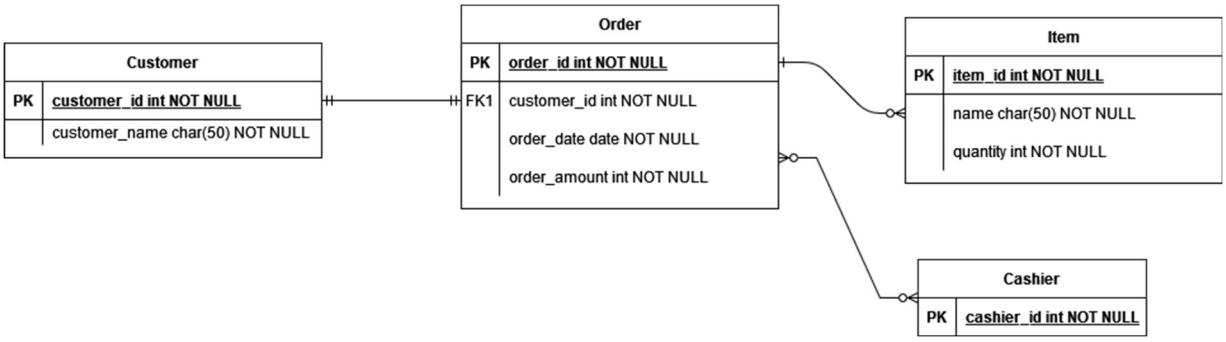
2.2. Data Dictionary

INVENTORY STRUCTURE

Field	Data Type	Description
Name	Text	Name of the product
SKU	Text	Bar code representation of product
Purchase Cost	Numeric	Price of product
Purchase Description	Text	Information about product
Initial Quantity on Hand	Numeric	Amount of item available
As of Date	Date	Date you start tracking the quantity on hand

The important fields are listed above. Images of the inventory structure can be found in the Appendix (ii).

2.3. Data Description of the System (using ERDs)



2.4. Problem Areas

The issues with current system are:

- QuickBooks Desktop Accounting software cannot be accessed anywhere. The admin is required to be physically present in the store to utilise the program.
- Physically counting products is prone to human error.
- The customer base is limited with the physical purchase model.

3.0 DEISGN OF THE PROPOSED SYSTEM

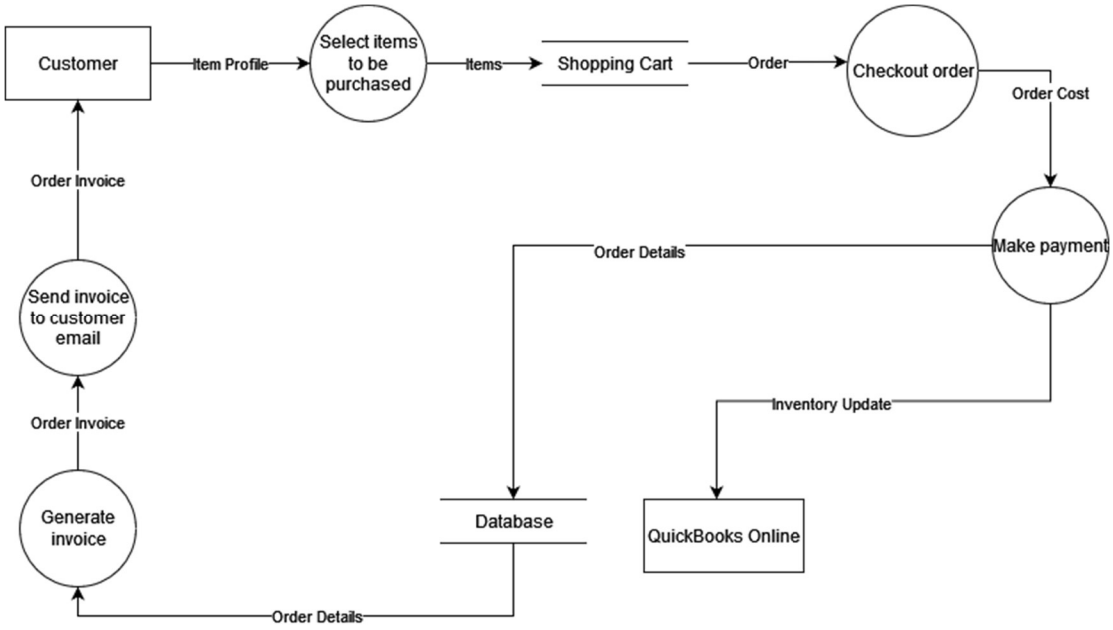
3.1. System Overview

The proposed system not only allows customers to browse catalogs of items and make purchases easily but also automates updating of inventory size when a purchase happens. The shopping cart logic automatically calculates the cost of the items and offers different payment options. Upon successful payment, the order is written to the database and an invoice is generated. In the background, QuickBooks Online handles the necessary inventory size updates.

3.2. Process Specification

The customer logs into the platform and select the items to be purchased. These items are loaded from Quickbooks online through either the V3-PHP-SDK or a raw HTTP request. The items are added to a cart. The cart contains a list of the items to be purchased, the quantity of the items, their cost and the total amount to be paid. The next step is to checkout. There are various payment options available. Cash, POS terminal and bank transfer are the physical payment options available while the online payments are processed by a payment gateway. The payment gateway acknowledges the payment by sending a receipt to the user. A successful online payment triggers a callback. A POST request is sent to Quickbooks to decrease the inventory size of the items bought (raw/SDK). Inventory can be monitored from the QuickBooks online console, products added and removed and inventory listed exported to pdf and Excel formats (see Appendix iii). Whereas if payment is not online, the agent receiving the order notification decreases the inventory size using Quickbooks online upon pickup or delivery. Sales made can be seen in the administration panel. The sales are represented graphically using charts and can also be exported to pdf.

3.2.1. Required Logical Data Flow Diagram



3.2.2. Data Dictionary

Customer Database

Name	Field Size	Datatype	Description
customer_id	5	Numeric	Uniquely identify customer
customer_email	50	Text	Customers email
customer_name	50	Text	Customers fullname

Orders Database

Name	Field Size	Datatype	Description
order_id	5	Numeric	Uniquely identifies an order

customer_id	5	Numeric	References the customer that made the order
order_date		Date	
order_amount	7	Numeric	Total cost of items purchased
order_items		JSON	Contains the items purchased and quantities

QuickBooks Online SDK/API Fields

Name	Datatype	Description
auth_mode	Text	OAuth protocol used
ClientID	Text	Uniquely identifies the client when making a request.
accessTokenKey	Text	Token received from QuickBooks Online by exchanging authorization code.
QBORealmID	Text	The company ID the API request is called against.
baseUrl	Text	Identify the base URL the request is making.

Resource manipulation URI examples are listed under Appendix (iii).

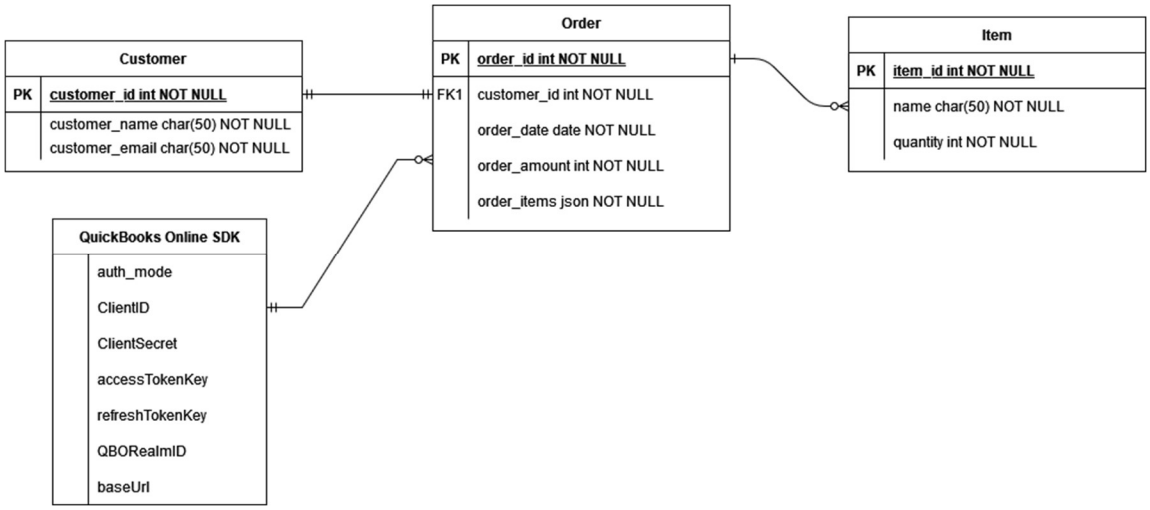
3.2.3. Decision Tables

Condition	Rules

Logged in	T	T	F	F
Payment status	T	F	T	F
Actions				
Redirect to login/register page			**X**	**X**
Redirect to catalogue		**X**		
Write order to DB	**X**			

3.3. Data Specification

3.3.1. Entity Relationship Diagram (ERD)



3.3.2. Database Design (Tables from ERDs)

- Customer

	#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra
☐	1	**customer_id** 🔑	int			No	*None*		AUTO_INCREMENT
☐	2	**customer_name**	char(50)	utf8mb4_0900_ai_ci		No	*None*		
☐	3	**customer_email**	char(50)	utf8mb4_0900_ai_ci		No	*None*		

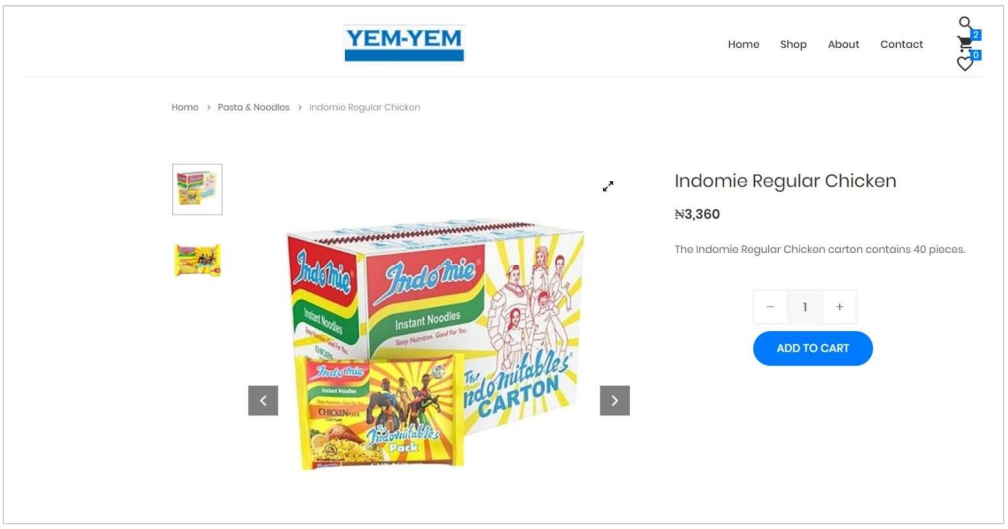
- Order

	#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra
☐	1	order_id 🔑	int			No	*None*		
☐	2	customer_id 🔗	int			No	*None*		
☐	3	order_date	date			No	*None*		
☐	4	order_amount	int			No	*None*		
☐	5	order_items	json			No	*None*		

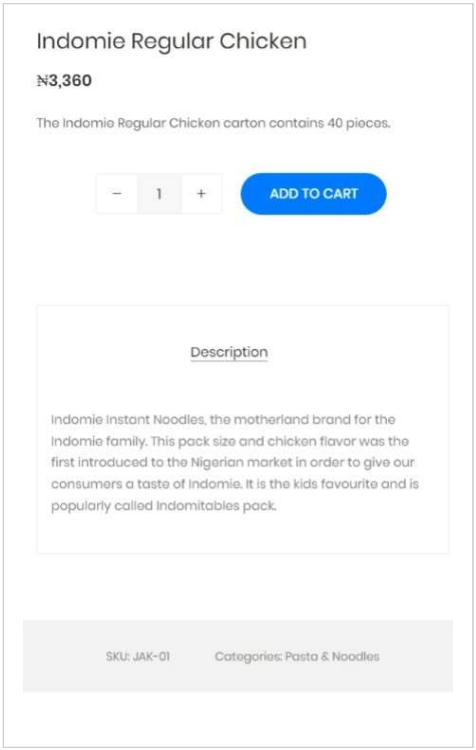
- Appendix (iv) and (v) show Item and QuickBooks SDK structure

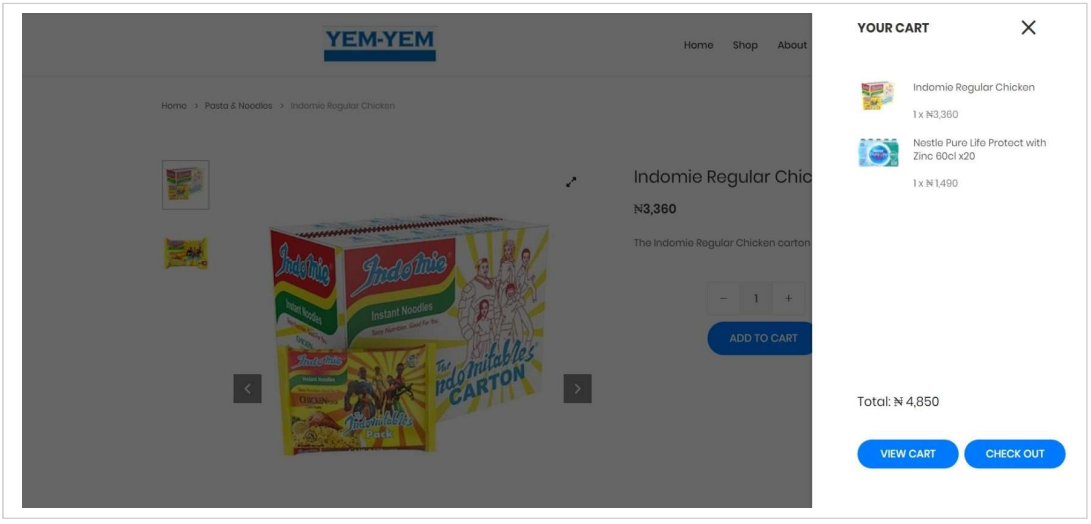
3.4. Screen Layout/ Specification

- Ordering (Desktop & Mobile view)

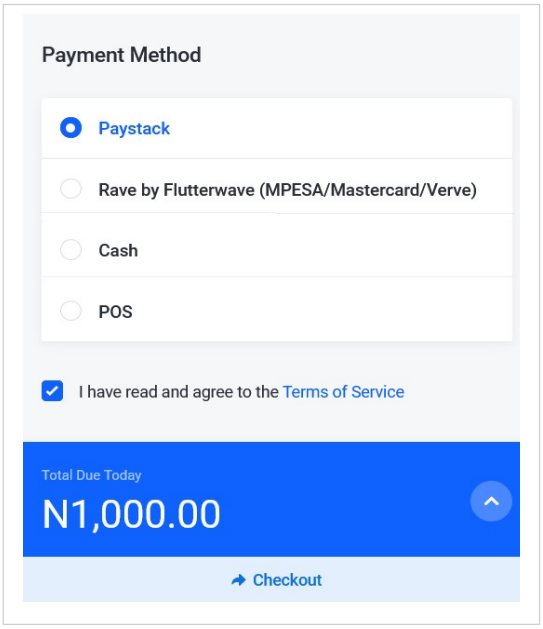


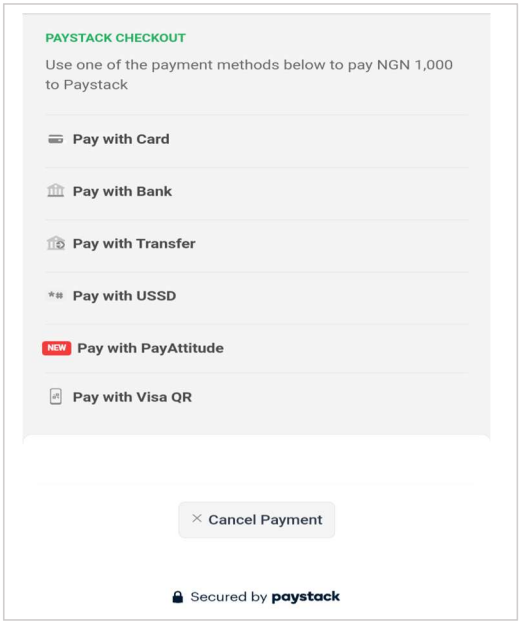

YEM-YEM
Home › Pasta & Noodles › Indomie Regular Chicken
Indomie Regular Chicken
₦3,360

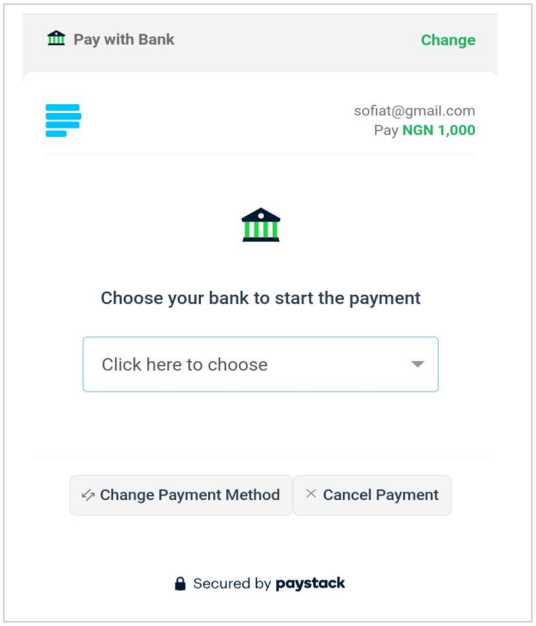

Indomie Regular Chicken
₦3,360
The Indomie Regular Chicken carton contains 40 pieces.
1
ADD TO CART
Description
Indomie Instant Noodles, the motherland brand for the Indomie family. This pack size and chicken flavor was the first introduced to the Nigerian market in order to give our consumers a taste of Indomie. It is the kids favourite and is popularly called Indomitables pack.
SKU: JAK-01
Categories: Pasta & Noodles

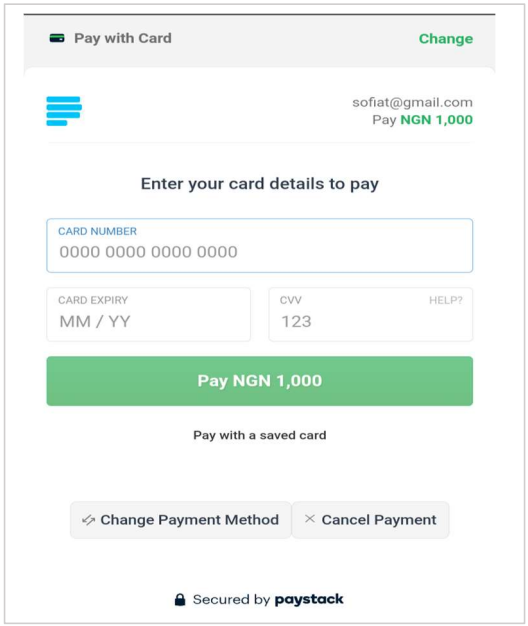

YEM-YEM
Home
Shop
About
Home › Pasta & Noodles › Indomie Regular Chicken
Indomie Regular Chic
₦3,360
The Indomie Regular Chicken carton
1
ADD TO CART
YOUR CART
Indomie Regular Chicken
1 x ₦3,360
Nestle Pure Life Protect with Zinc 60cl x20
1 x ₦1,490
Total: ₦ 4,850
VIEW CART
CHECK OUT

- Billing

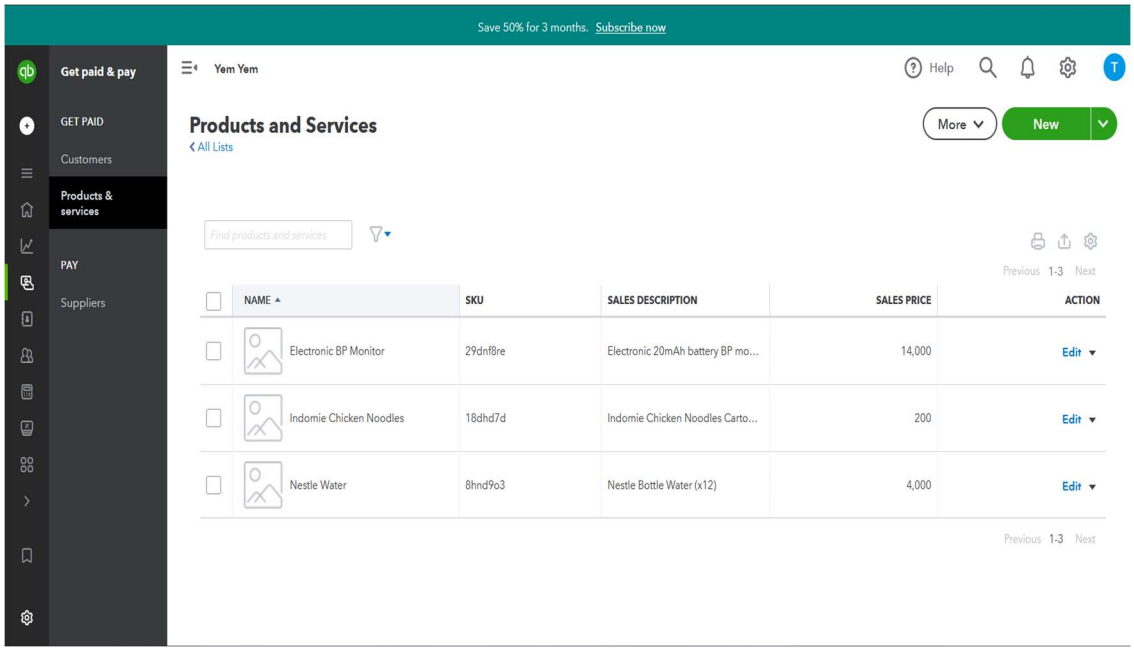




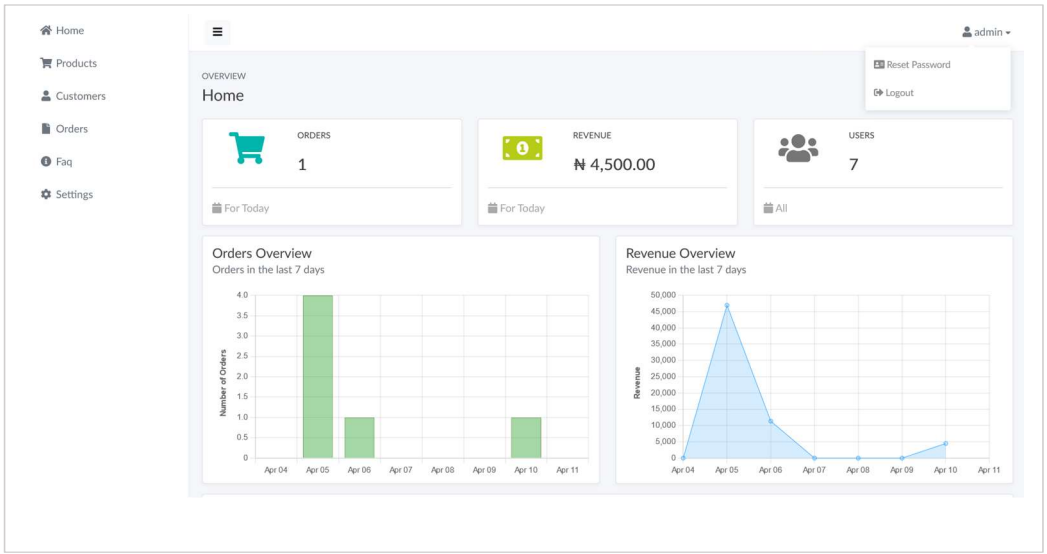


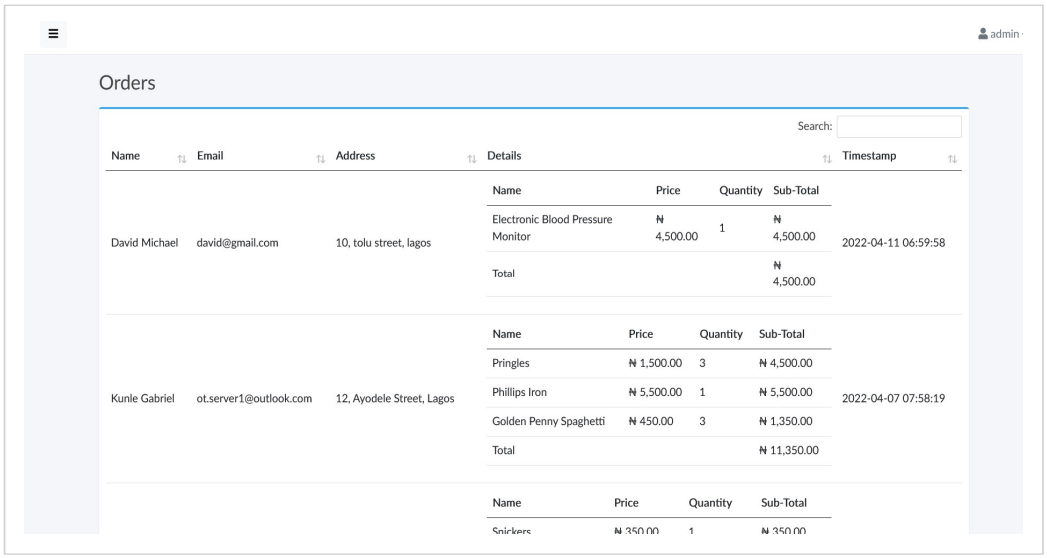


- Inventory

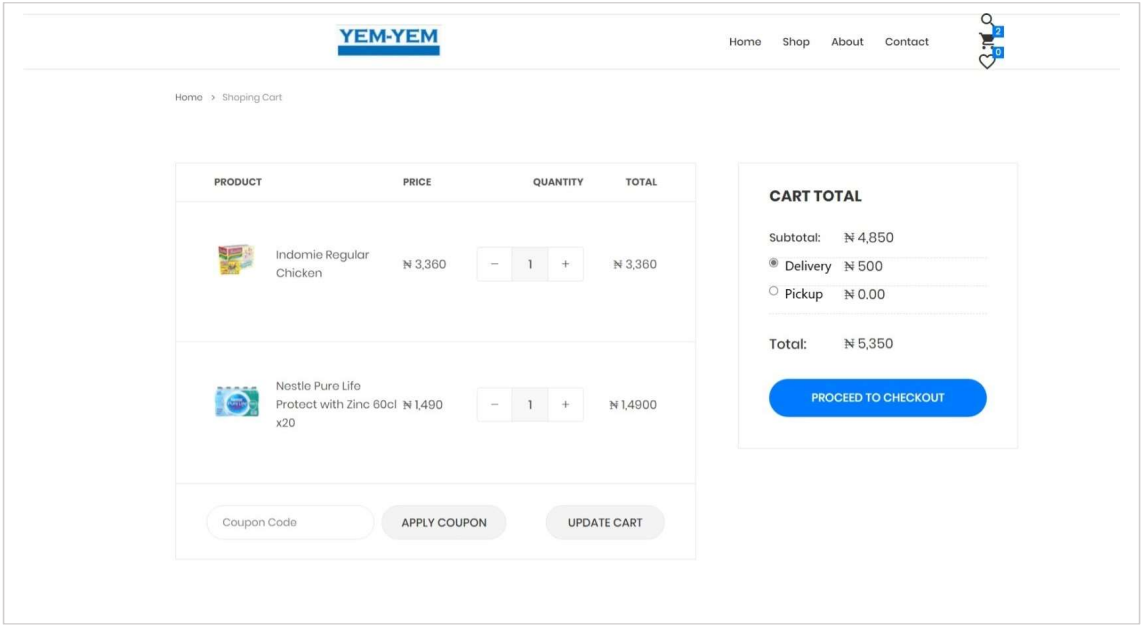

Save 50% for 3 months. Subscribe now
Get paid & pay
GET PAID
Customers
Products & services
PAY
Suppliers
Yem Yem
Help
Products and Services
All Lists
More
New
Find products and services
Previous 1-3 Next
NAME
SKU
SALES DESCRIPTION
SALES PRICE
ACTION
Electronic BP Monitor
29dnf8re
Electronic 20mAh battery BP mo...
14,000
Edit
Indomie Chicken Noodles
18dhd7d
Indomie Chicken Noodles Carto...
200
Edit
Nestle Water
8hnd9o3
Nestle Bottle Water (x12)
4,000
Edit
Previous 1-3 Next

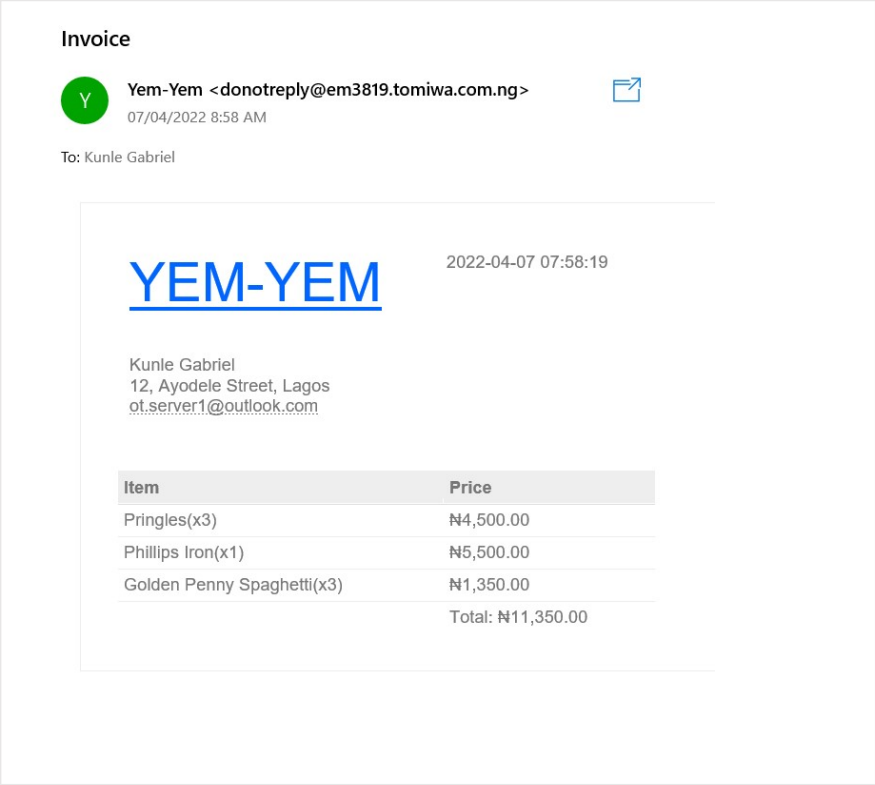
- Accounting


Home
Products
Customers
Orders
Faq
Settings
admin
Reset Password
Logout
OVERVIEW
Home
ORDERS
1
For Today
REVENUE
₦ 4,500.00
For Today
USERS
7
All
Orders Overview
Orders in the last 7 days
Number of Orders
Apr 04
Apr 05
Apr 06
Apr 07
Apr 08
Apr 09
Apr 10
Apr 11
Revenue Overview
Revenue in the last 7 days
Revenue



3.5. Reports/ Form Specifications





3.6. Program/ Module Specifications

3.7. Test Plans

There are no plans to test the system since it will not be implemented.

4.0 CONCLUSION

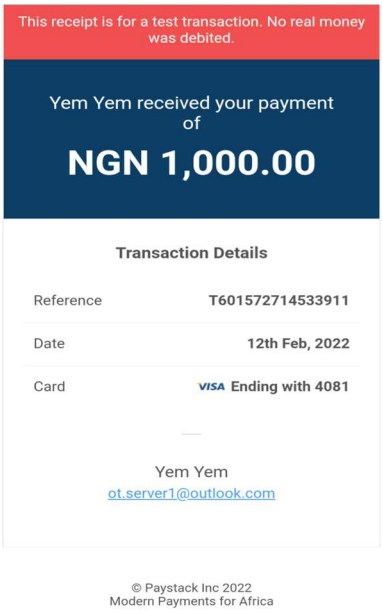
A virtual store that provides the functionality of commercial transactions over the web is a major resource for modern day businesses. The system makes shopping experience more flexible. This project provides easy access for the administrators and managers to view inventory and cash flow via the system. Both Yem Yem and its customers will benefit from the implementation of this system.

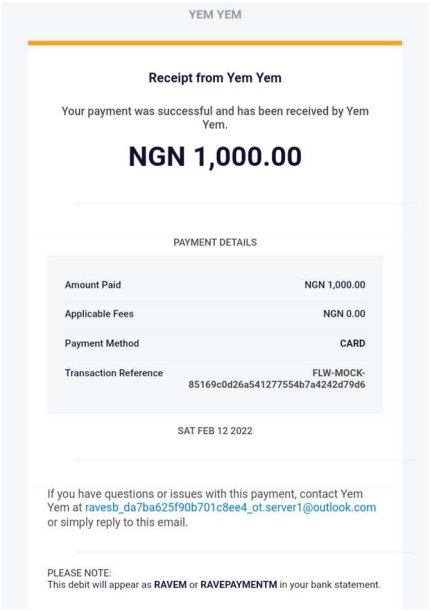
References:

- Modern Systems Analysis and Design Textbook, Ninth Edition
- QuickBooks Online API basic schema and data formats: https://developer.intuit.com/app/developer/qbo/docs/learn/rest-api-features
- What is PERT and how can we use it?: https://www.pmta.co.za/free-resources/what-is-pert-and-how-can-we-use-it#:~:text=PERT%20is%20a%20three%20point,an%20activity's%20cost%20or%20duration
- QuickBooks Online API for inventory management: https://developer.intuit.com/app/developer/qbo/docs/api/accounting/all-entities/item
- QuickBooks V3-PHP-SDK demo: https://github.com/intuit/QuickBooks-V3-PHP-SDK/tree/master/src/_Samples

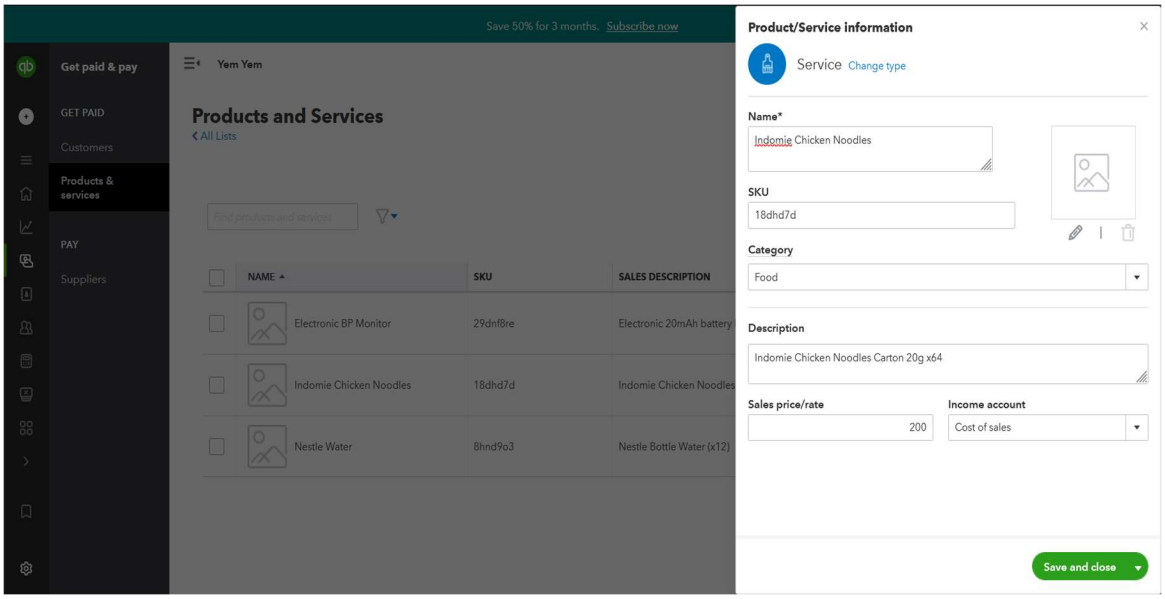
APPENDIX

i. Billing Receipts

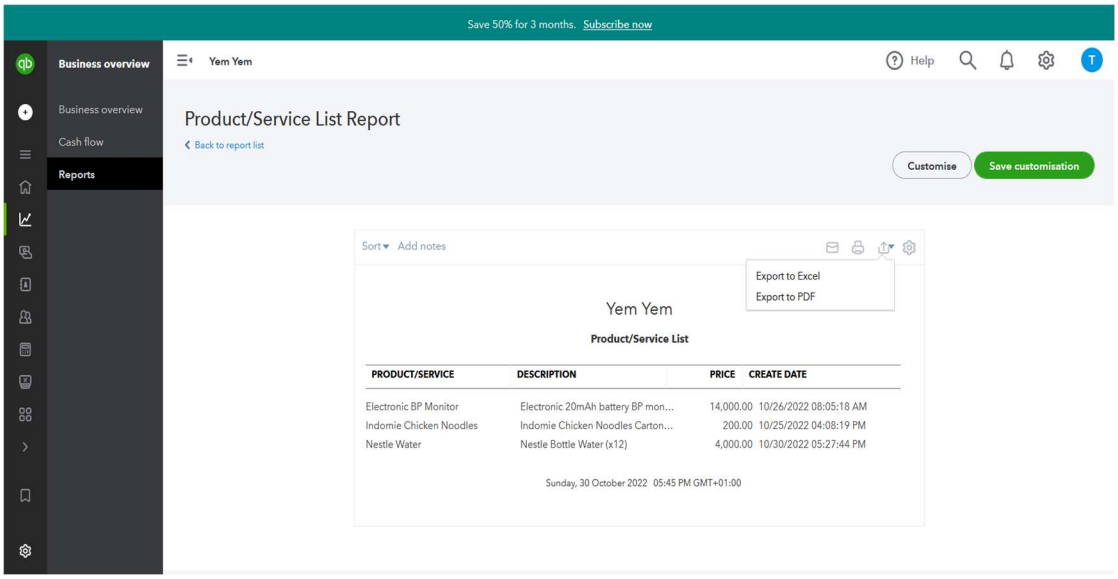




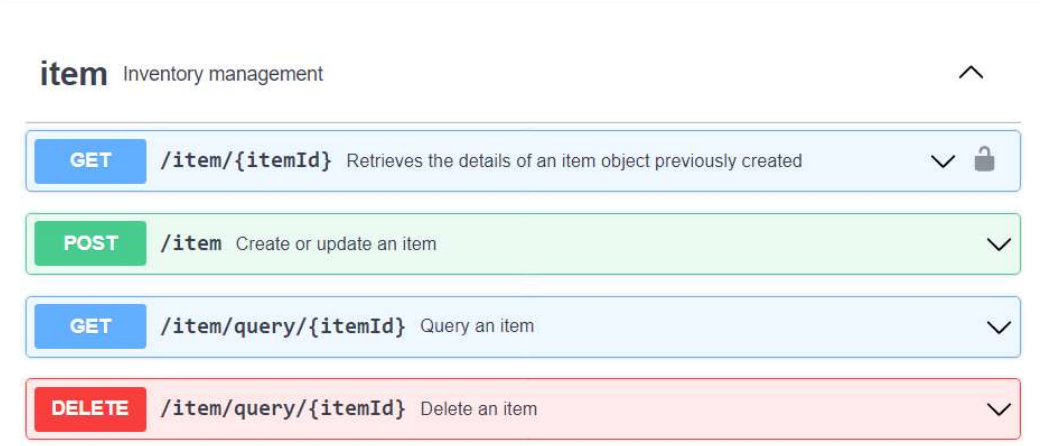
ii. Inventory structure



iii. Exporting inventory to pdf or Excel



iv. QuickBooks Online API/SDK URI examples



v. QuickBooks Online item object model

```
Item ⌄ {
    trackQtyOnHand          boolean
    name                    string
    id                      integer($int64)
    value                   integer
    qtyOnHand               integer($int32)
    active                  boolean
    metaData                ⌄ {
                              createTime            string($date-time)
                              lastUpdatedTime       string($date-time)

                            }
    time                    string($date-time)

}
```

vi. QuickBooks Online SDK instance

```php
<?php
//Replace the line with require "vendor/autoload.php" if you are using the Samples from outside of _Samples folder
include('../config.php');

use QuickBooksOnline\API\Core\ServiceContext;
use QuickBooksOnline\API\DataService\DataService;
use QuickBooksOnline\API\PlatformService\PlatformService;
use QuickBooksOnline\API\Core\Http\Serialization\XmlObjectSerializer;
use QuickBooksOnline\API\Facades\Invoice;

// Prep Data Services
$dataService = DataService::Configure(array(
  'auth_mode'       => 'oauth2',
  'ClientID'        => "XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX",
  'ClientSecret'    => "XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX",
  'accessTokenKey'  => "XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX..XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX",
  'refreshTokenKey' => 'XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX',
  'QBORealmID'      => "xxxxxxxxxxxxxxxxx",
  'baseUrl'         => "development"
));
```

vii. Current and proposed system Activity Entity Tables

Current System

S/N	ENTITY	ACTIVITY
1	Customer	Make an order
2	Cashier	Input order and quantities
3		Cost is calculated
4	Customer	Make payment
5	Cashier	Submit order
6		Inventory is updated
7		Print receipt

Proposed System

S/N	ENTITY	ACTIVITY
1	Customer	Add items to cart
2		Checkout
3		Select payment method
4		Make payment
5		Order is recorded in database
6		Invoice is sent to customers email
7	QuickBooks Online	Update inventory size