StoreFront

Design Document

[Overview of the Storefront software 2](#_Toc310879967)

[Customer Management 3](#_Toc310879968)

[Customer Table 3](#_Toc310879969)

[Address Table 3](#_Toc310879970)

[Phone Table 3](#_Toc310879971)

[Email Table 4](#_Toc310879972)

[Credit Card Table 4](#_Toc310879973)

[Customer Functionality 5](#_Toc310879974)

[Add a new customer 5](#_Toc310879975)

[Soft delete a new customer 5](#_Toc310879976)

[Customer Lookup 5](#_Toc310879977)

[Pulls all information regarding the customer 5](#_Toc310879978)

[Verifies Log In Credentials 5](#_Toc310879979)

[Inventory Control 6](#_Toc310879980)

[Product Table 6](#_Toc310879981)

[Order Management 7](#_Toc310879982)

[Order Table 7](#_Toc310879983)

[Order Details Table 7](#_Toc310879984)

[Accounting 8](#_Toc310879985)

[System Requirements/options 8](#_Toc310879986)

# Overview of the Storefront software

StoreFront is an application designed and developed for small companies to use as Point of Sale (POS) clients. The intention is to make this software as OS agnostic as possible. Initially we will target the Linux OS, followed by Windows. The intent is to build for both types of clients and servers.

# Customer Management

Customer data will be maintained in a database, and encrypted. Client server communications will need to be encrypted for any transmission of personally identifying information (pii). The following is a table describing the customer table setup.

## Customer Table

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Name** | **Description** | **Type** | **Size** | **PK** | **FK** | **FK Name** | **Null able** | **Encrypted** |
| **pk\_intCustID** |  | **Int** |  | **Y** | **N** |  | **No** | **No** |
| **strTitle** | **Mr./Mrs./etc** | **Varchar** | **20** | **N** | **N** |  | **Yes** | **No** |
| **strFirstName** |  | **Varchar** | **20** | **N** | **N** |  | **No** | **No** |
| **strMiddleName** |  | **Varchar** | **20** | **N** | **N** |  | **Yes** | **No** |
| **strLastName** |  | **Varchar** | **30** | **N** | **N** |  | **No** | **No** |
| **strPostTitle** | **II/Junior/etc** | **VarChar** | **10** | **N** | **N** |  | **Yes** | **No** |
| **strLoginName** |  | **varChar** | **20** | **N** | **N** |  | **No** | **No** |
| **strPassword** |  | **varChar** | **50** | **N** | **N** |  | **No** | **Yes** |
| **blDeleted** |  | **Boolean** |  | **N** | **N** |  | **No** | **No** |

Each customer may have more than one Address. The addresses should be soft deleted in the event a client makes a typo entry, or Law Enforcement needs to see former shipping information.

## Address Table

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Name** | **Description** | **Type** | **Size** | **PK** | **FK** | **FK Name** | **Null able** | **Encrypted** |
| pk\_intAddressID |  | Int |  | Y | N |  | No | No |
| Fk\_intCustID |  | Int |  | N | Y | Customer.pk\_intCustID | No | No |
| strAddress1 |  | Varchar | 100 | N | N |  | No | No |
| strAddress2 |  | Varchar | 100 | N | N |  | Yes | No |
| strCity |  | Varchar | 50 | N | N |  | No | No |
| strState |  | VarChar | 2 | N | N |  | No | No |
| strZipCode |  | varChar | 10 | N | N |  | No | No |
| strType | (s) Shipping  (h) Home  (b) Billing | varChar | 1 | N | N |  | No | No |
| blDeleted |  | Boolean |  | N | N |  | No | No |

Each customer may have more than one Phone Number. The phone numbers should be soft deleted in the event a client makes a typo entry, or Law Enforcement needs to see former information.

## Phone Table

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Name** | **Description** | **Type** | **Size** | **PK** | **FK** | **FK Name** | **Null able** | **Encrypted** |
| pk\_intPhoneID |  | Int |  | Y | N |  | No | No |
| Fk\_intCustID |  | Int |  | N | Y | Customer.pk\_intCustID | No | No |
| intPhoneNumber | Saved w/o Dashed or Periods | Int |  | N | N |  | No | No |
| strType | (c) Cellular  (h) Home  (b) Business | varChar | 1 | N | N |  | No | No |
| blRecvTxt | Opt in for text adverts. Only for Cellular | Boolean |  | N | N |  | Yes | No |
| blDeleted |  | Boolean |  | N | N |  | No | No |

Each customer may have more than one email address. The addresses should be soft deleted in the event a client makes a typo entry, or Law Enforcement needs to see former information. Additionally, we should check the addresses to make sure two users aren’t using the same email address at the same time. (Soft deleted addresses should not count towards this rule)

## Email Table

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Name** | **Description** | **Type** | **Size** | **PK** | **FK** | **FK Name** | **Null able** | **Encrypted** |
| pk\_intEmailID |  | Int |  | Y | N |  | No | No |
| Fk\_intCustID |  | Int |  | N | Y | Customer.pk\_intCustID | No | No |
| strEmailAddress |  | Varchar | 100 | N | N |  | No | No |
| strType | (p) Personal  (b) Business | varChar | 1 | N | N |  | No | No |
| blRecvNewsLetter | Opt in for adverts. | Boolean |  | N | N |  | No | No |
| blDeleted |  | Boolean |  | N | N |  | No | No |

Each customer may have more than one Credit Card. The credit card should be soft deleted in the event a client makes a typo entry, or Law Enforcement needs to see former information.

## Credit Card Table

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Name** | **Description** | **Type** | **Size** | **PK** | **FK** | **FK Name** | **Null able** | **Encrypted** |
| pk\_intCCID |  | Int |  | Y | N |  | No | No |
| Fk\_intCustID |  | Int |  | N | Y | Customer.pk\_intCustID | No | No |
| intCCNum |  | Int |  | N | N |  | No | Yes |
| intExpMonth |  | Int |  | N | N |  | No | No |
| intExpYear |  | Int |  | N | N |  | No | No |
| intSecNum |  | Int |  | N | N |  | No | No |
| strType | (v) Visa  (m) Master Card  (d) Discover | varChar | 1 | N | N |  | No | No |
| blDeleted |  | Boolean |  | N | N |  | No | No |

## Customer Functionality

The Customer Class should be able to:

### Add a new customer

Check to see if the client existed in the database before. If so, their old information should be used as a starting point. No original information should be presented to the customer until they have verified the data themselves. Verify the required information is available and sensible. Where possible, verify information accuracy. The information should be formatted and displayed on a web page.

### Soft delete a new customer

When a customer is soft deleted, all information should be soft deleted. If a client comes back, they should have to re-input their ancillary information to verify it is the client, and no data should be presented to them from the old record.

### Customer Lookup

A user can look up the customer using their Name, or their phone number.

### Pulls all information regarding the customer

Pulls all ancillary information, including:

* + Addresses
  + Credit Card info
  + Phone numbers
  + Email Addresses
  + Orders

### Verifies Log In Credentials

Checks login and password against those stored in the database.

# Inventory Control

The inventory control system will need to keep track of what is currently in stock, what is on order, and what are the projected sales for the next period.

## Product Table

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Name** | **Description** | **Type** | **Size** | **PK** | **FK** | **FK Name** | **Null able** | **Encrypted** |
| pk\_intProductID |  | Int |  | Y | N |  | No | No |
| strProdDesc | Description of the product | Varchar | 256 | N | N |  | No | No |
| strSKU | SKU number (Unique) | Varchar | 50 | N | N |  | No | No |
| intQuantity |  | Int |  | N | N |  | No | No |
| intOnOrder |  | Int |  | N | N |  | No | No |
| dtNextRcvDate |  | Datetime |  | N | N |  | No | No |
| intInvLevel |  | Int |  | N | N |  | No | No |
| intCost |  | Int |  | N | N |  | No | No |
| intPrice |  | Int |  | N | N |  | No | No |

# Order Management

The product should be able to interface with the preceding sections to accept orders, and track them through delivery. This should include both online and register sales, as well as the shipping information and contents.

## Order Table

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Name** | **Description** | **Type** | **Size** | **PK** | **FK** | **FK Name** | **Null able** | **Encrypted** |
| pk\_intOrderID |  | Int |  | Y | N |  | No | No |
| Fk\_intCustomerID |  | Int |  | N | Y | Pk\_intCustomerID | No | No |
| fk\_intAddressID |  | Int |  | Y | T | pk\_intAddressID | No | No |
| dtOrderPlacedDate |  | Datetime |  | N | N |  | No | No |
| dtOrderShippedDate |  | Datetime |  | N | N |  | No | No |
| dtOrderDeliverdDate |  | datetime |  | N | N |  | No | No |
| str\_PymtType | (c) Credit Card  (d) Cash/Money Order  (k) Check | varchar | 2 | N | N |  | No | No |
| Fk\_intCCID |  | Int |  | N | Y | Pk\_intCCID | Yes | No |
| fk\_trackingID |  | Varchar | 100 | N | Y | Shipping company | Yes | No |

## Order Details Table

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Name** | **Description** | **Type** | **Size** | **PK** | **FK** | **FK Name** | **Null able** | **Encrypted** |
| pk\_intOrderDetailID |  | Int |  | Y | N |  | No | No |
| fk\_intOrderID |  | Int |  | N | Y | pk\_intOrderID | No | No |
| fk\_intProductID |  | Int |  | N | N | pk\_intProductID | No | No |
| intQuantity |  | Int |  | N | N |  | No | No |
| intLineTotal |  | Int |  | N | N |  | No | No |

# Accounting

We will need to track all incoming and outgoing monies, as well as accounts owed.

## Account Table

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Name** | **Description** | **Type** | **Size** | **PK** | **FK** | **FK Name** | **Null able** | **Encrypted** |
| pk\_IntAcctID |  | Int |  | Y | N |  | No | No |
| strAcctNum |  | Varchar | 20 | N | N |  | No | No |
| strAcctDesc |  | varchar | 255 | N | N |  | No | No |
| blIsCredit |  | Boolean |  | N | N |  | No | No |
| intBalance |  | int |  | N | N |  | No | No |

## Ledger Table

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Name** | **Description** | **Type** | **Size** | **PK** | **FK** | **FK Name** | **Null able** | **Encrypted** |
| pk\_intLedgerID |  | Int |  | Y | N |  | No | No |
| fk\_intOrderID |  | Int |  | N | Y | pk\_intOrderID | Yes | No |
| Fk\_IntAcctId |  | Int |  | N | Y | Pk\_IntAcctID | No | No |
| intCredit |  | Int |  | N | N |  | Yes | No |
| intDebit |  | Int |  | N | N |  | Yes | No |
| Fk\_InvoiceID |  | Int |  | N | Y | Pk\_intInvoiceID | Yes | No |

## Invoice Table

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Name** | **Description** | **Type** | **Size** | **PK** | **FK** | **FK Name** | **Null able** | **Encrypted** |
| pk\_intInvoiceID |  | Int |  | Y | N |  | No | No |
|  |  |  |  |  |  |  |  |  |

1. Accounts Receivable
2. Accounts Payable

# System Requirements/options

1. Large multi-location clients
   1. Large companies may need a central server for multiple locations
   2. Sub-servers to act as go-betweens to limit down time.
2. Speed
3. Database servers supported.