

# Personal Shopper Device

## Architecture Requirement Specification

Version 2.1

November 22, 2015

### **Authors:**

Jordan Callero,  
Sukhman Ghumman,  
Hunter Hammond,  
Conor Leeds,  
Nanya Ugwuh.

### **Mentor:**

Christopher Wildt

**Seattle Boffins**

**IASA**

## Table of Contents

1	Overview.....	3
2	Conceptual Architecture .....	3
2.1	Assumptions .....	4
2.2	Context Diagram & Its Components .....	4
2.3	Components of PSS.....	5
3	Information Architecture .....	7
3.1	Conceptual Data Model.....	7
3.2	Logical Data Model.....	8

Version Number	Changes
2.1	Complete Reformating Overhaul to match up with the Business Document
2.0	Updated the Conceptual Data Model and Reformated
1.0	

## 1. Overview

PERSONAL SHOPPER SYSTEM (PSS) is a system that will allow shoppers to purchase items using a mobile device. This will allow valid members to scan items as they shop, store and update their shopping cart, process their payments, and trigger the printing of receipts at designated kiosks. Such a system would provide shoppers with a quick and easy way to purchase items, saving them time and effort. This document has main two sections, conceptual architecture and information architecture. Conceptual architecture is a high level description of the system, including interactions and associations amongst the components it comprises of. The PSS' main interactions include the membership validation system to validate the customer's membership, the point of sales system which processes the various payments methods, the item database which provides details on items the shopper is buying, and the transaction history system which stores all finished shopper transactions. Information architecture describes the high level data model and its structure, showing the associations of various things the components consist of.

## 2. Conceptual Architecture

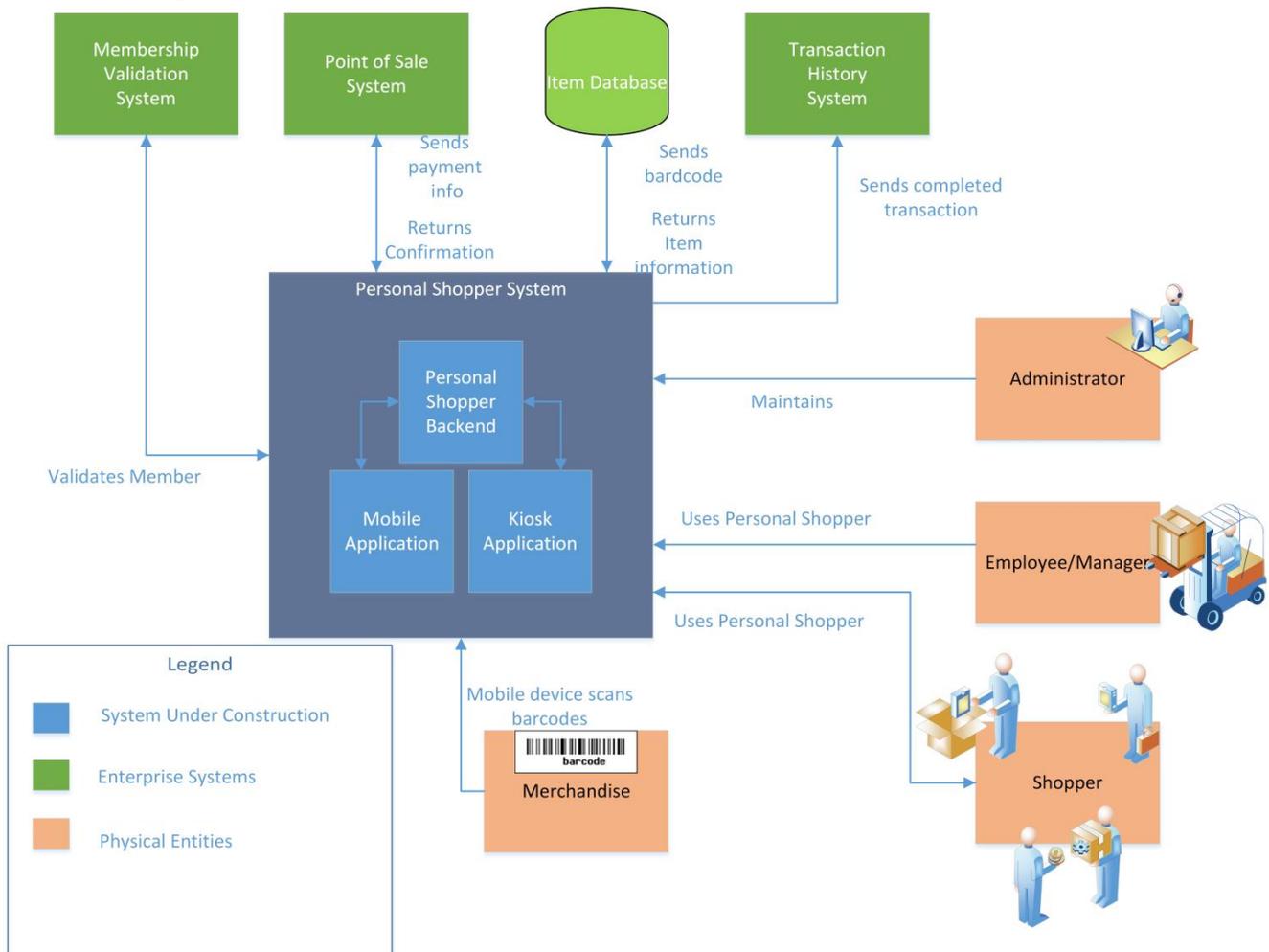


Figure 1: Context Diagram

## **2.1 Assumptions**

The Assumptions in this document are:

- Enterprise systems are already in place.
- Membership validation system, point of sale system, item database, and the transaction history system are managed and updated by enterprise systems.

## **2.2 Context Diagram & Its Components**

### **2.2.1 Personal Shopper System:**

This is the system we are constructing and it encapsulates the Mobile Application, Kiosk Application, and the Personal Shopper Backend. It is the central communication hub for the rest of the diagram components, receiving input from physical entities and interacting with enterprise systems accordingly.

### **2.2.2 Membership Validation System:**

The mobile application for the PSS requires member authentication for access to the application's full functionality, such as scanning and purchasing items. This system checks for a valid membership and lets the PSS know whether authentication was successful.

### **2.2.3 Point of Sale System:**

Responsible for processing payment information for all store transactions. This system will return confirmation of a successful or unsuccessful payment to the PSS.

### **2.2.4 Item Database:**

This database stores all item information necessary for identifying items scanned in the store. A physical scanning device produces a unique identifier used to query this database and return the item information to the PSS Mobile Application.

### **2.2.5 Transaction History System:**

Once a transaction is completed in the PSS, it is pushed to this system.

### **2.2.6 Administrator:**

These are persons who are responsible for maintaining the PSS and addressing issues, either software or hardware related.

### **2.2.7 Employee/Manager:**

These are persons employed by the store. Their responsibilities will include aiding shoppers with the mobile or kiosk applications.

### **2.2.8 Shopper:**

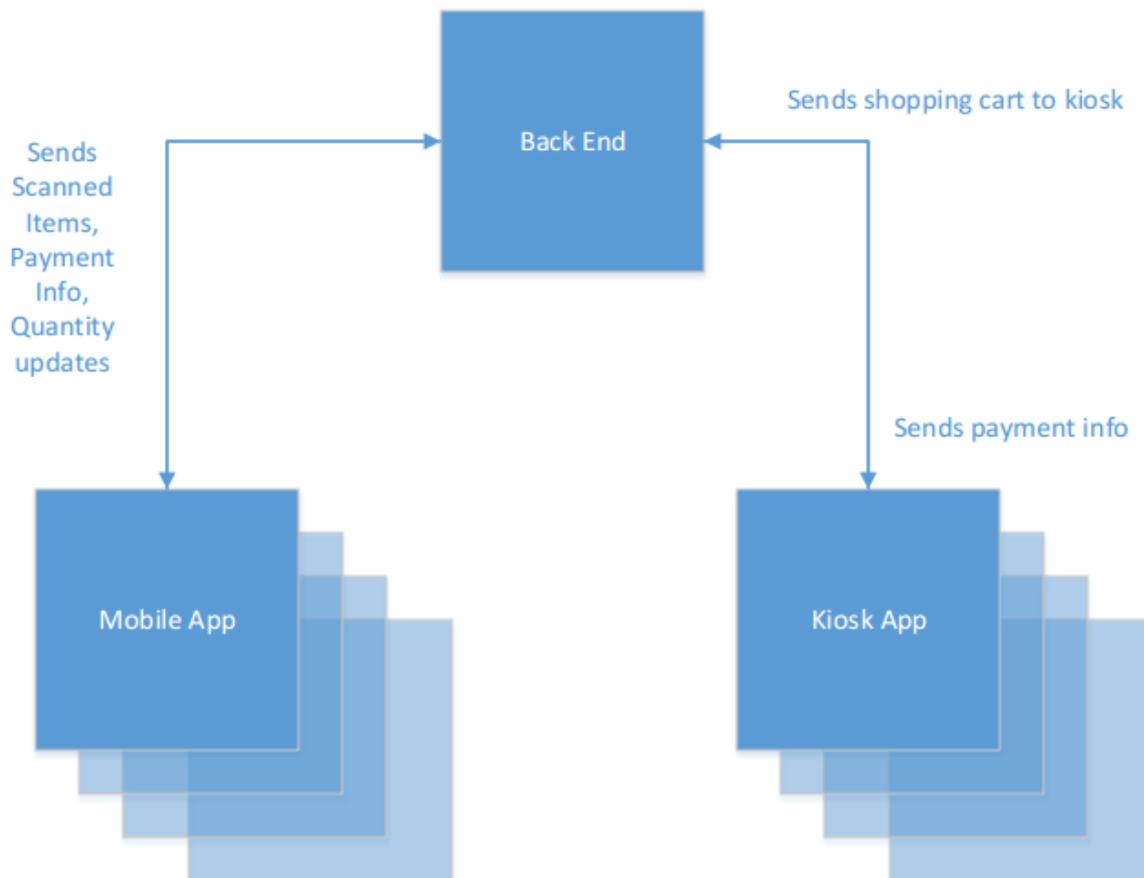
The primary user of the mobile and kiosk applications. These persons must have valid store memberships in order to purchase merchandise.

### **2.2.9 Merchandise:**

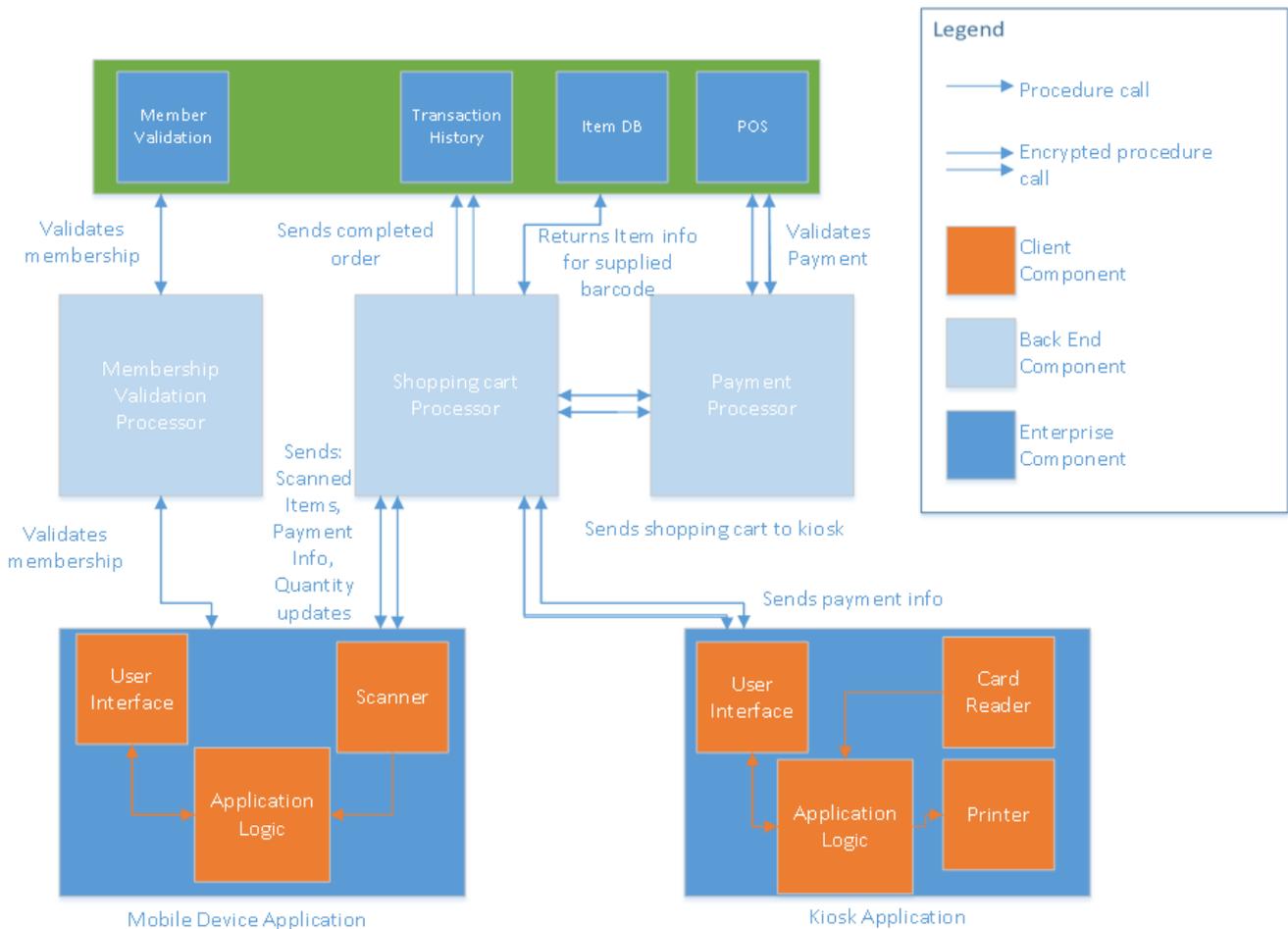
In-store items which are physically labeled with barcodes, which must be scanned using a mobile device, which uses the PSS backend to perform an Item Database lookup. If a scan is successful the item is added to the personal shopping cart viewable on the mobile device.

## 2.3 Components of PSS

PSS adheres to client-server architecture. The servers are dedicated to managing network traffic and procedure calls made by the clients in order to interact with the existing enterprise systems. Clients here are the workstations (mobiles or kiosks) on which users run the application. Clients provide an interface to allow users to request services of the server and to display the results returned by the server. In this situation, the server waits for client requests to arrive and then responds back to them accordingly. In the following diagram we mainly show the multiple mobile applications and kiosk applications running in parallel whose requests are being handled by the server's back end. Client-server architecture provides a more secure, reliable, scalable system and has easy data flow which improves performance as well.



**Figure 2: Sub components of the PSS**



**Figure 3: Components of PSS**

### 2.3.1 Mobile Device Application

Shoppers will be using this application to scan store merchandise and interact with their virtual shopping cart, including viewing, deleting, and modifying quantity of scanned merchandise. Once store membership is validated, the application can scan merchandise and transmit this information to the shopping cart processor for payment and completion.

### 2.3.2 Kiosk Application

Shoppers must swipe their membership card at an in-store kiosk in order to print their receipt and pay any outstanding balance. The kiosk retrieves cart and payment information from the shopping cart processor.

### 2.3.3 Membership Validator Processor

Shoppers using the Mobile Device Application are required to authenticate themselves to gain access to scanning and purchasing merchandise through the application. This component communicates with the existing Enterprise infrastructure to validate memberships via a membership ID number located on a store-given card.

### 2.3.4 Shopping Cart Processor

Performs encrypted procedure calls to transmit and receive Mobile Device and Kiosk Applications data including merchandise, payment, and transaction history details. Processing involves performing the lookup of scanned merchandise in the Item Database, pushing completed transactions to the

Transaction History System, and sending and receiving payment information between applications and the Payment Processor.

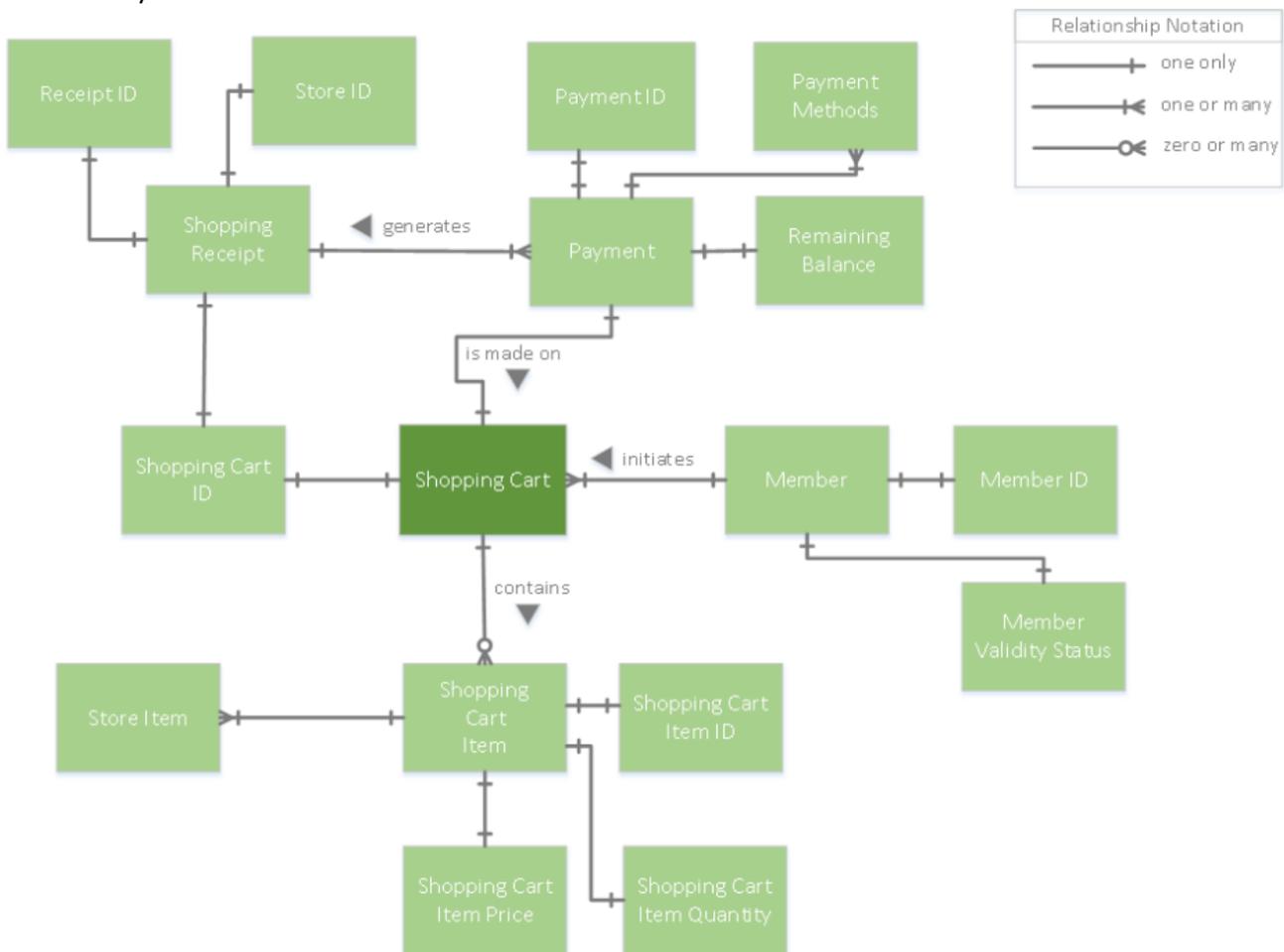
### 2.3.5 Payment Processor

Receives and transmits encrypted payment information between the Shopping cart Processor and the Enterprise Point of Sale System. Responsible for updating the outstanding balance during a transaction.

## 3. Information Architecture

### 3.1 Conceptual Data Model

The conceptual data model, otherwise known as the semantic data model, provides a high level view of the structure of the data model. In the PSS, such a model identifies the 'things' which are significant to the system together with the associations between the 'things'. These associations are represented using cardinality notations such as one to one association, one to many association, and many to many association.



**Figure 4: Conceptual Data Model**

## 3.1.1 Shopping Cart:

It holds all items which are planned to be purchased by a member. It is initiated by a member and can contain zero, one, or many shopping cart items.

## 3.1.2 Shopping Cart Item:

This is an item contained in a shopping cart. Generally, one store item is usually associated with one or more shopping cart items. A shopping cart item is composed of a shopping cart item identification (ID), price, and quantity.

## 3.1.3 Member:

This is an individual who initiates a shopping cart while being a valid store member. A member is composed of a member identification (ID) and a validity status.

## 3.1.4 Payment:

This is the amount payable for item(s) within a shopping cart. It is composed of a Payment ID and a Remaining Balance (Amount to be paid). A payment can be made using several payment methods.

## 3.1.5 Payment Method:

This is the means through which a payment is made for items within a shopping cart. A payment can be made with one or many payment methods..

## 3.1.6 Shopping Receipt:

This is a member's written proof that shows payment for items within a shopping cart has been received by the store. A payment for shopping cart item(s) generates one receipt only. A shopping receipt is composed of one receipt ID, one shopping cart ID, and one store ID where the shopping took place.

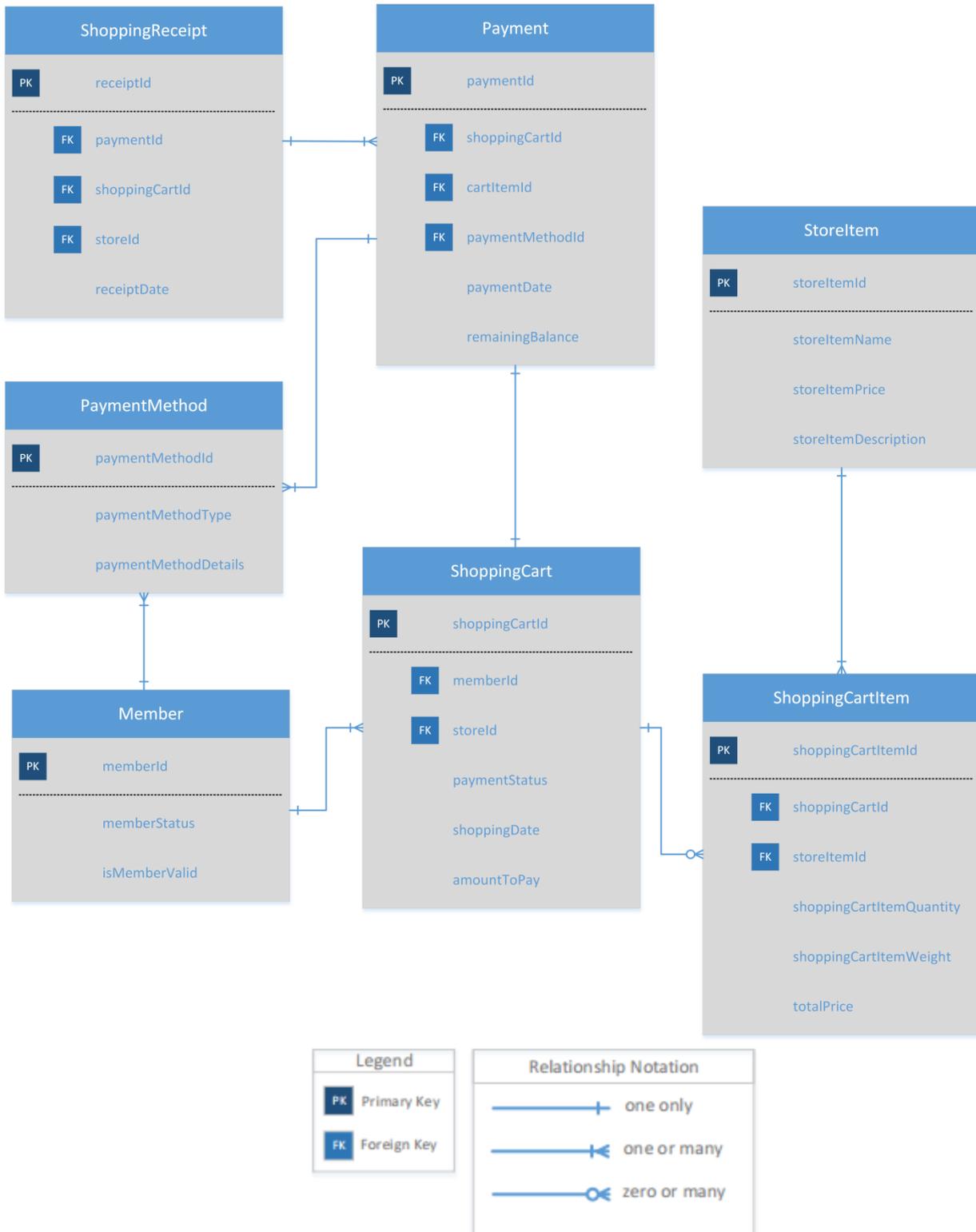
## 3.1.7 Store Item:

This is an item available for sale in a particular store. One or more shopping cart items is associated with one store item.

## 3.2 Logical Data Model

The logical data model extends the conceptual data model in Figure 5 by providing a detailed representation of entities, attributes, and associations that describe the business requirements and define the business rules. The logical data model in Figure 5 of the PSS makes the following assertions:

- A ShoppingCart is created by one Member whose validityStatus must be active.
- A ShoppingCart keeps a record of the memberID of the Member.
- A ShoppingCart will be initiated within a store and will keep a record of the storeID.
- A Member adds one or many ShoppingCartItem to a ShoppingCart.
- At any given time, a ShoppingCart can have zero, one, or many ShoppingCartItem's.
- A store keeps the name, description, and price information about a StoreItem.
- One StoreItem corresponds to one or many ShoppingCartItem's.
- A Payment will be made for one ShoppingCart using one or many PaymentMethod's. Each Payment will keep a record of the paymentMethodId, shoppingCartItemid, and shoppingCartId.
- A ShoppingReceipt will be generated for one complete Payment. Each ShoppingReceipt keeps a record of the shoppingCartId, StoreId, receiptDate, and paymentId.



**Figure 5: Logical Data Model**

### 3.3 Physical Model

Since the physical model is optional, at this stage of the project we have elected not to include it at this time.