Scrum Masters, Inc.

Project Checkpoint 2 Report

By Ian Weiss, Ryan Fino, Steve Cuzzi, Joe Golubieski, Shane Mulcahy

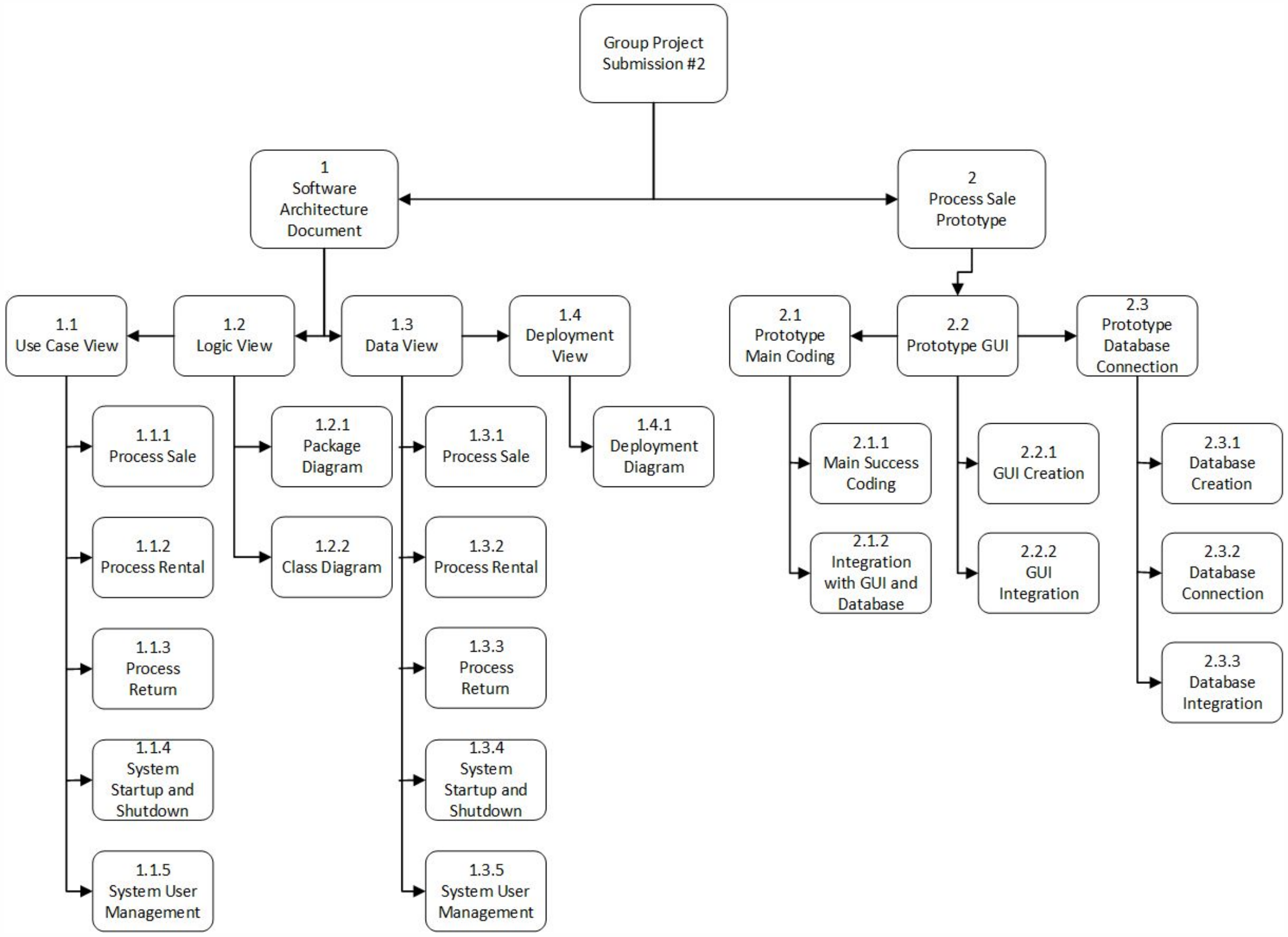
WBS and RM

Responsibility Matrix

Task	**Ian W.**	**Joe G.**	**Ryan F.**	**Shane M.**	**Steve sucks C.**
Process Sale Data View			X		
Process Sale Use Case View			X		
Process Rental Data View		X			
Process Rental Use Case View		X			
Process Return Data View					X

Process Return Use Case View					X
System User Management Data View				X	
System User Management Use Case View				X	
System Startup and Shutdown Data View	X				
System Startup and Shutdown Use Case View	X				
Package Diagram			X		
Class Diagram			X		
Deployment Diagram		X			
Prototype Database					X
Prototype GUI				X	
Prototype Main Coding	X				

Work Breakdown Structure


Group Project Submission #2
1 Software Architecture Document
2 Process Sale Prototype
1.1 Use Case View
1.2 Logic View
1.3 Data View
1.4 Deployment View
2.1 Prototype Main Coding
2.2 Prototype GUI
2.3 Prototype Database Connection
1.1.1 Process Sale
1.1.2 Process Rental
1.1.3 Process Return
1.1.4 System Startup and Shutdown
1.1.5 System User Management
1.2.1 Package Diagram
1.2.2 Class Diagram
1.3.1 Process Sale
1.3.2 Process Rental
1.3.3 Process Return
1.3.4 System Startup and Shutdown
1.3.5 System User Management
1.4.1 Deployment Diagram
2.1.1 Main Success Coding
2.1.2 Integration with GUI and Database
2.2.1 GUI Creation
2.2.2 GUI Integration
2.3.1 Database Creation
2.3.2 Database Connection
2.3.3 Database Integration

Software Architecture Document

Scrum Masters Inc. POS System

Architectural Representation

This SAD summarizes the architecture from the following views:

Logic View: a view showing how the system logic operates and the relationships between the classes in the code

Data View: a view of how the data moves through each node and process

Use Case View: a view of how the code directly connects to properly manage the flow of information when solving a use case

Deployment View: a view showing how the setup of the physical systems relate and communicate to each other

Architectural Factors

Please refer to the Supplementary Specification Table for details on architectural factors.

Architectural Decisions (Technical Memos)

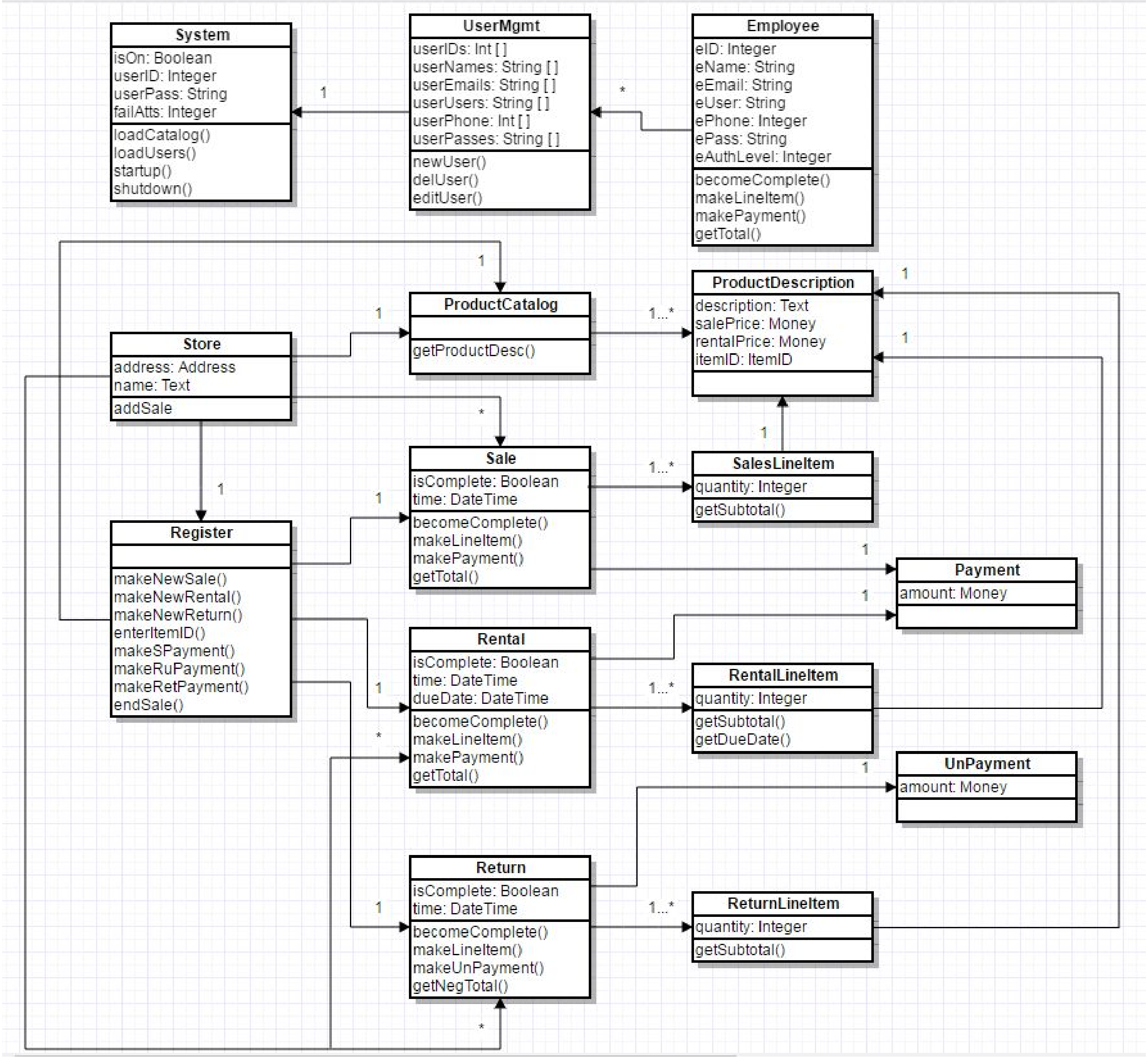
Issue: Legal- Tax Rule Compliance

Solution: Create/Purchase a tax calculator component. Calculate tax on items using this new component.

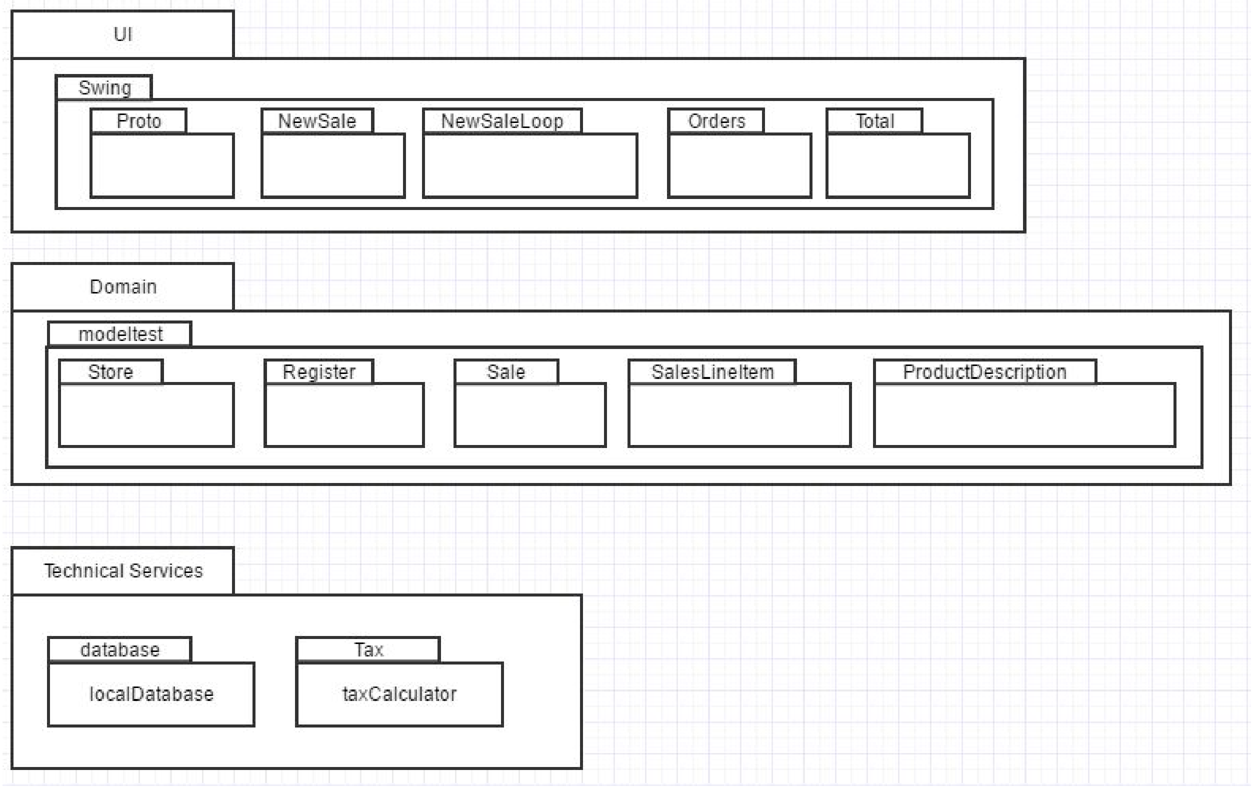
Issue: Reliability- Recovery from Remote Service Failure

Solution: Switch to local database view. Store any changes to the database/transactions for upload into remote database at a later time.

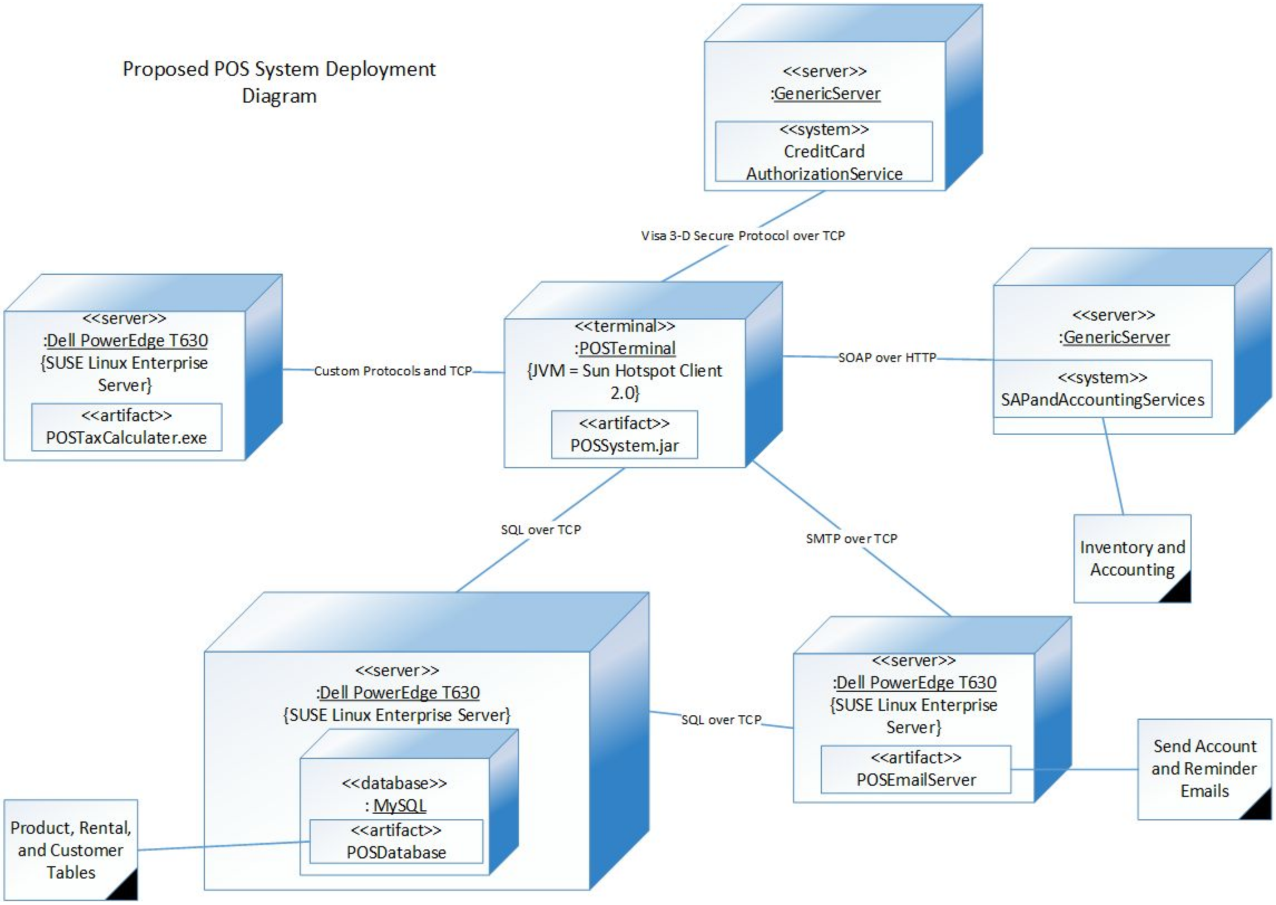
Class Diagram



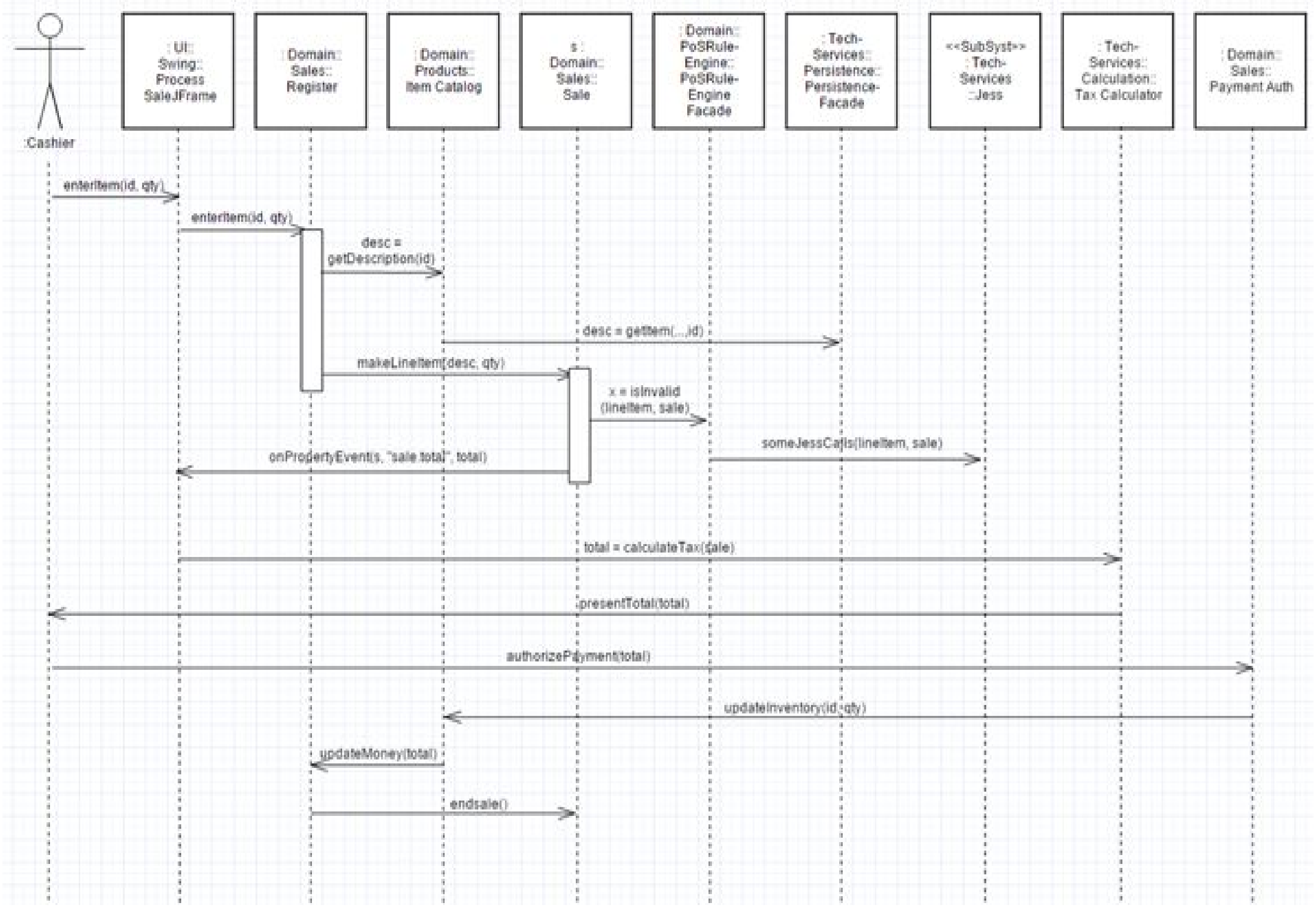
Package Diagram



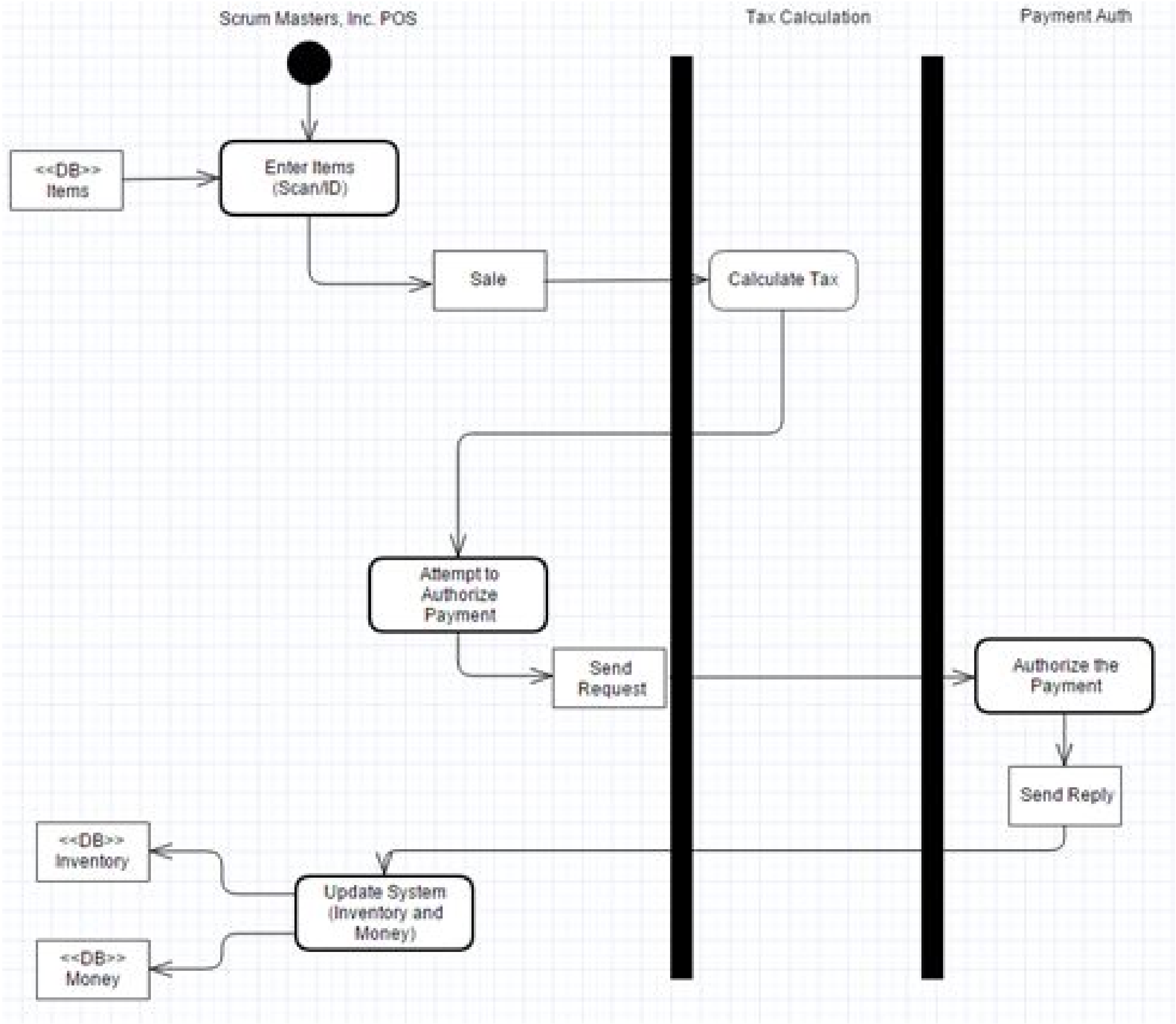
Deployment Diagram



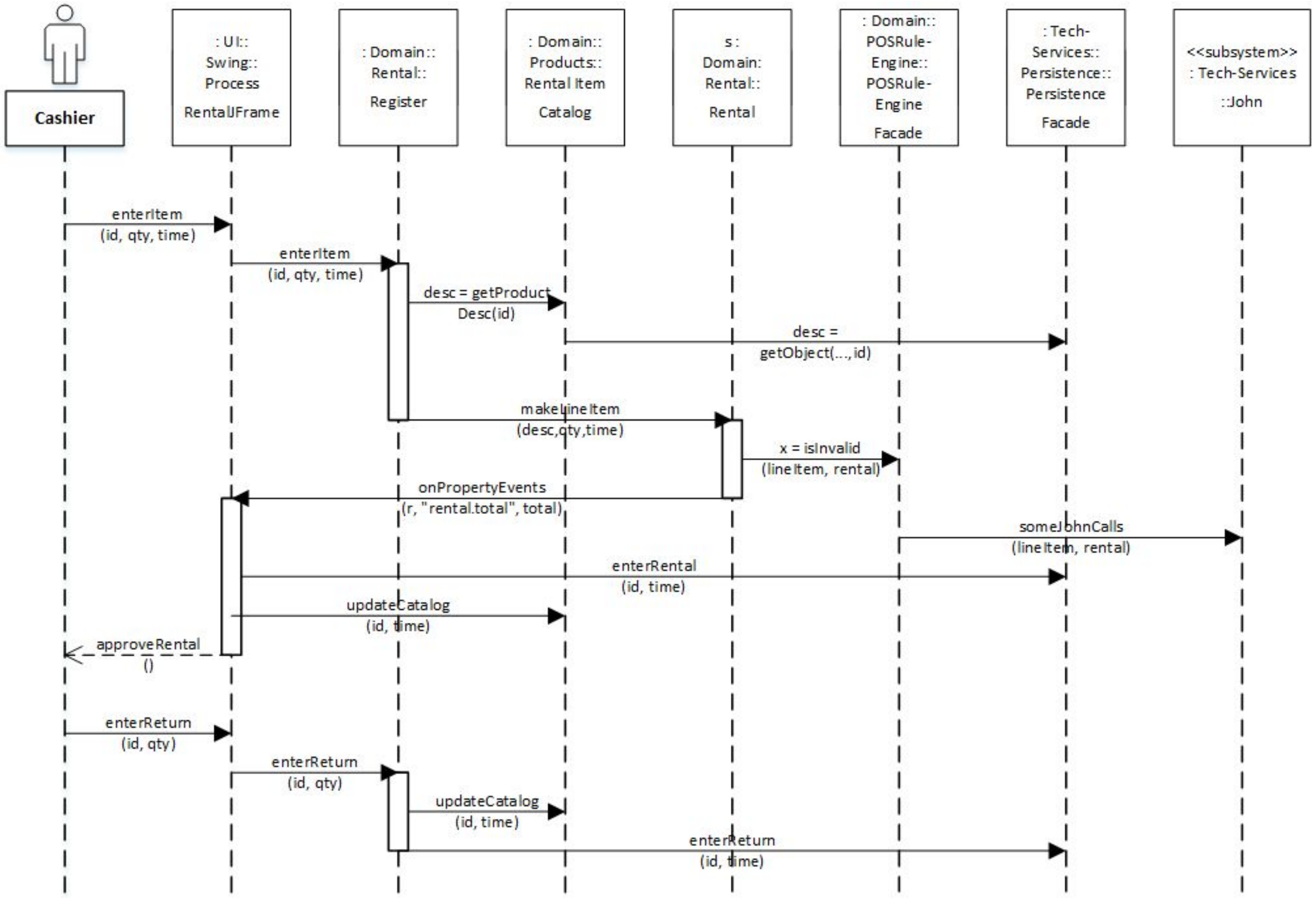
Process Sale Use Case View



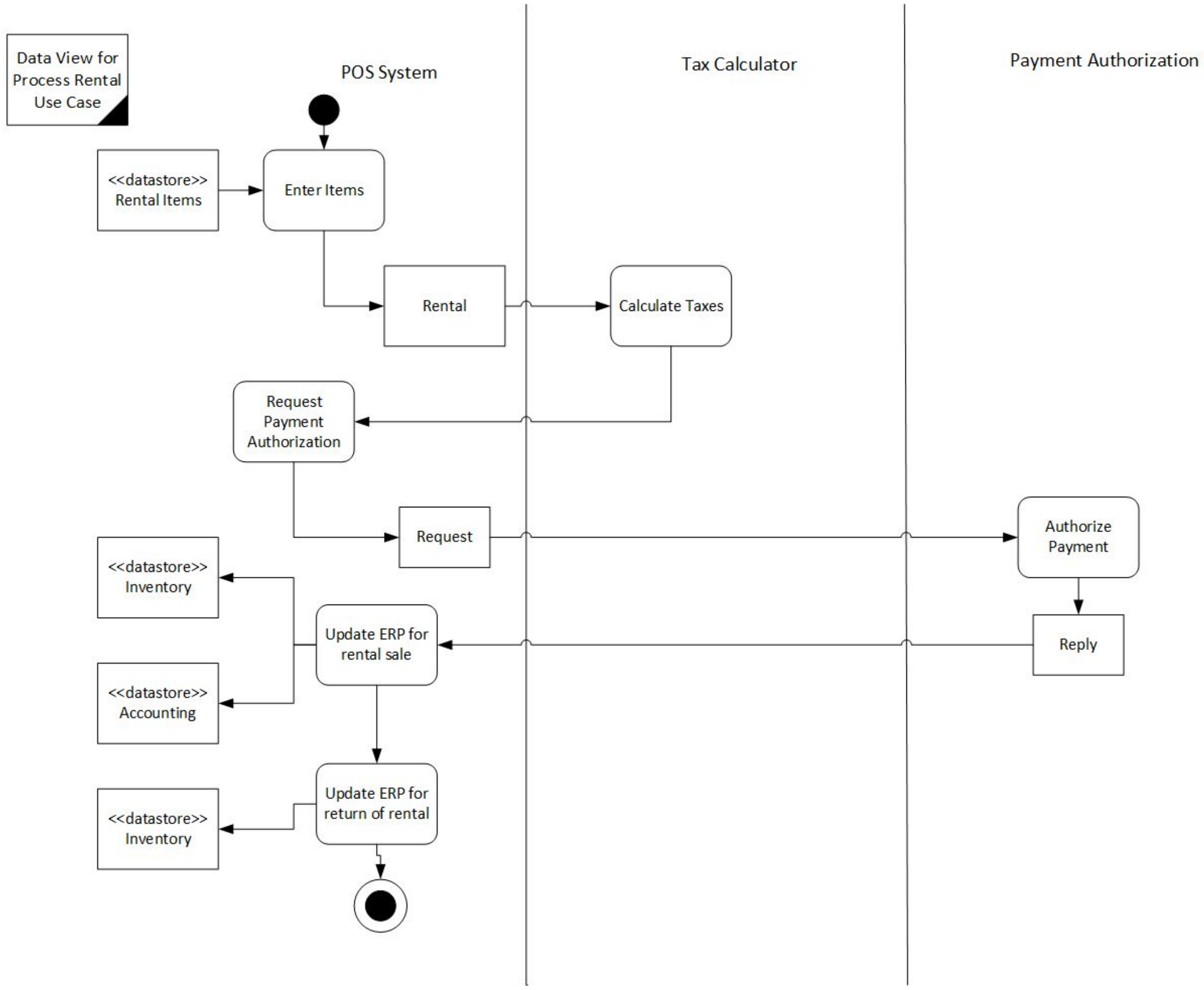
Process Sale Data View



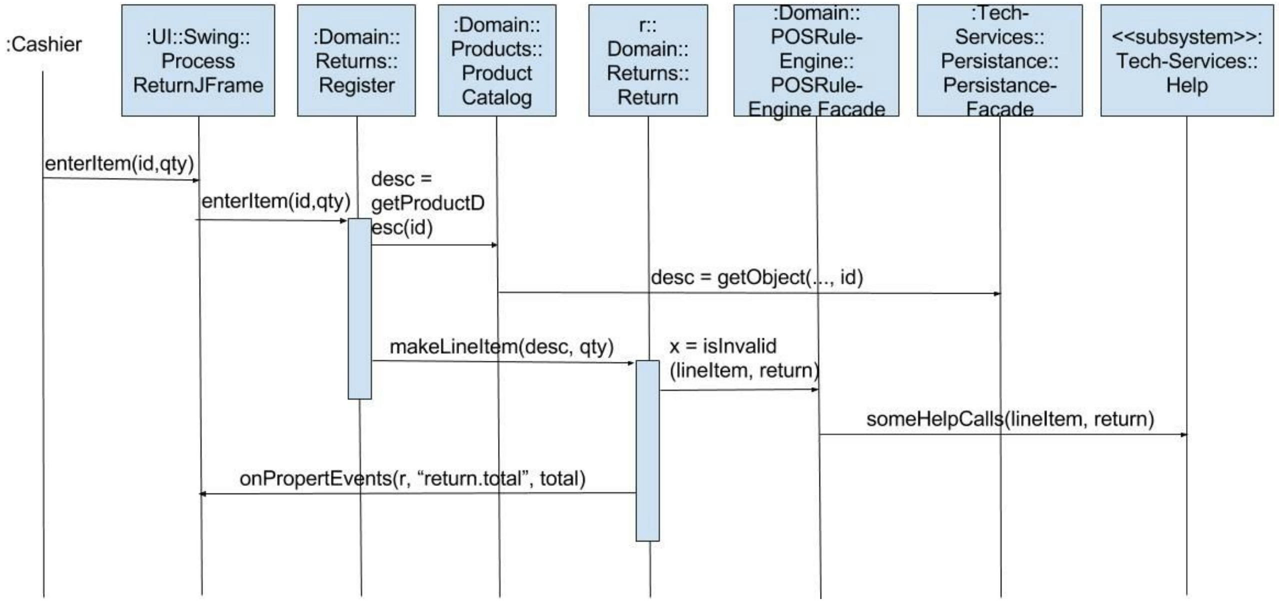
Process Rental Use Case View


Cashier
: UI:: Swing:: Process RentalJFrame
: Domain:: Rental:: Register
: Domain:: Products:: Rental Item Catalog
s : Domain: Rental:: Rental
: Domain:: POSRule-Engine:: POSRule-Engine Facade
: Tech-Services:: Persistence:: Persistence Facade
<<subsystem>> : Tech-Services ::John
enterItem (id, qty, time)
enterItem (id, qty, time)
desc = getProduct Desc(id)
desc = getObject(..., id)
makeLineItem (desc,qty,time)
x = isInvalid (lineItem, rental)
onPropertyEvents (r, "rental.total", total)
someJohnCalls (lineItem, rental)
enterRental (id, time)
updateCatalog (id, time)
approveRental ()
enterReturn (id, qty)
enterReturn (id, qty)
updateCatalog (id, time)
enterReturn (id, time)

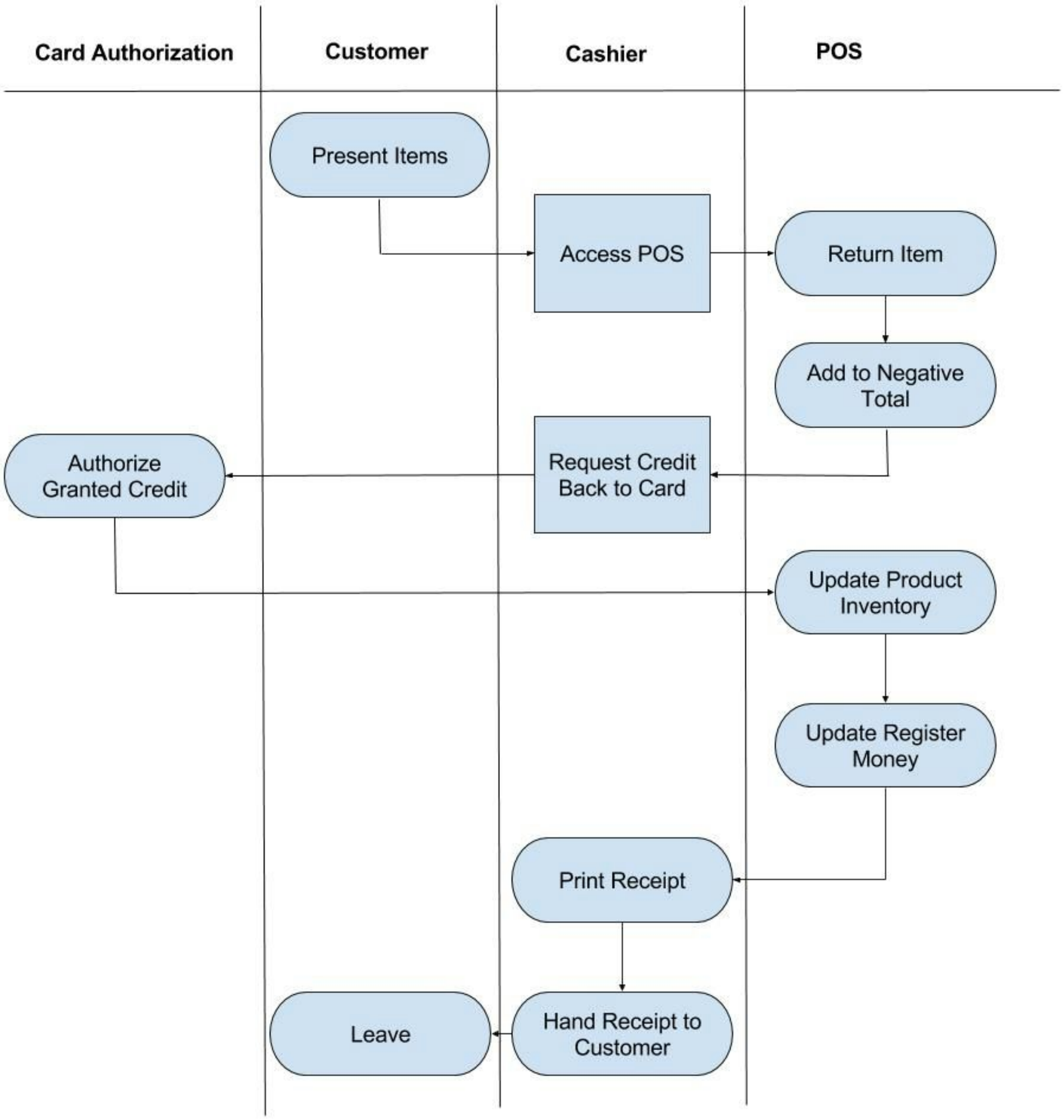
Process Rental Data View



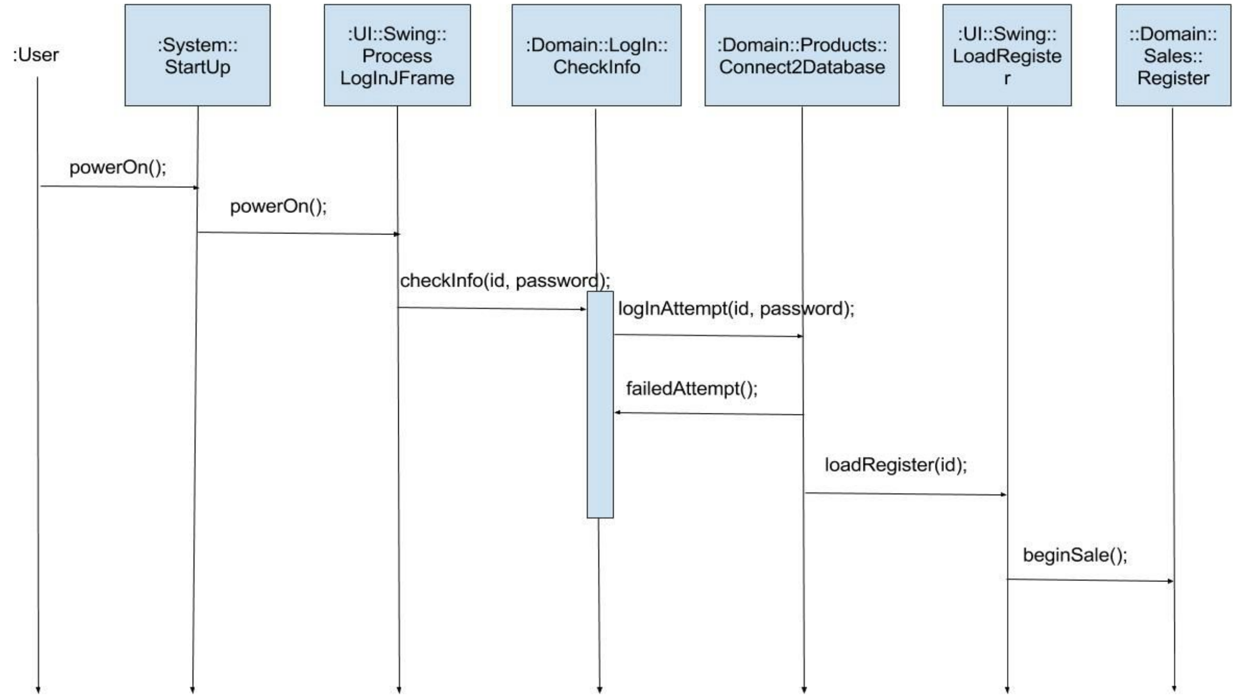
Process Return Use Case View



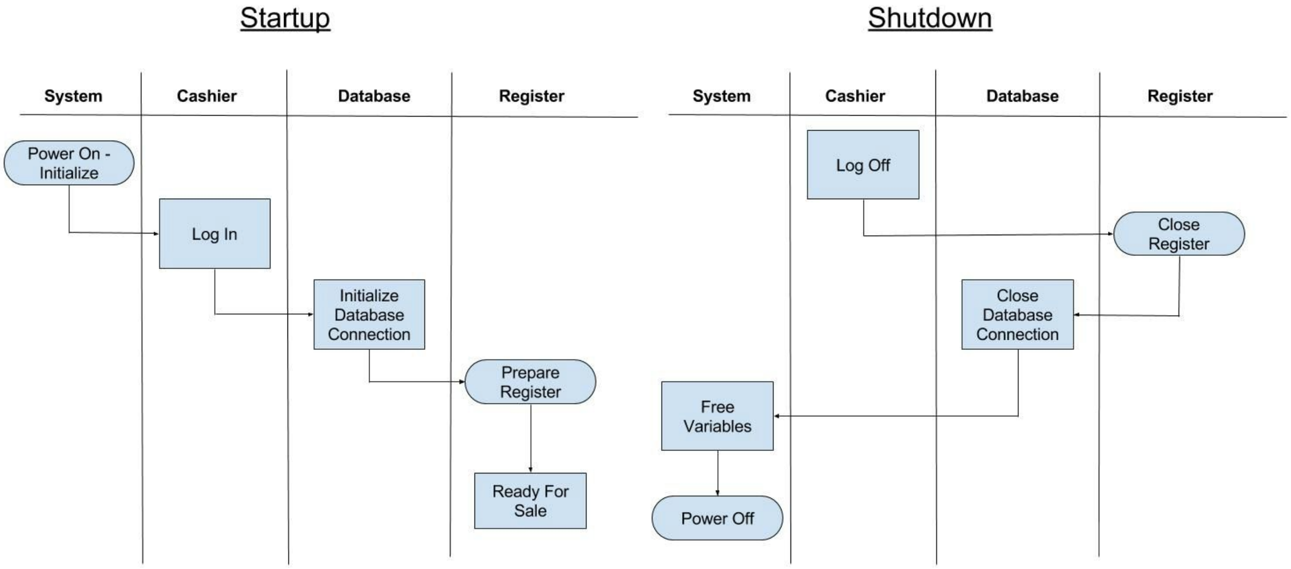
Process Return Data View



System Startup and Shutdown Use Case View

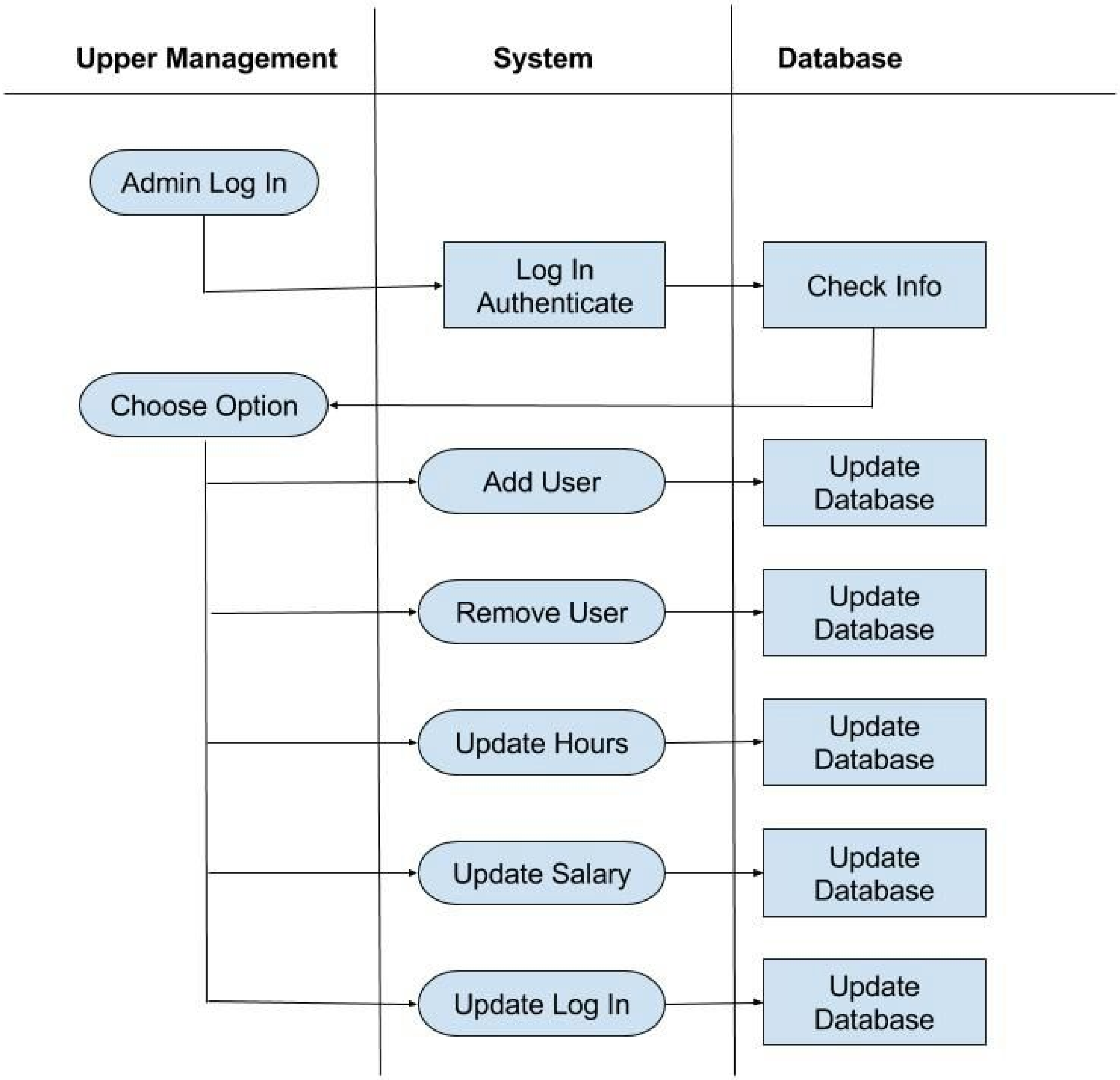


System Startup and Shutdown Data View



System User Management Use Case View

System User Management Data View



<u>Supplementary Specification</u>

This serves as a repository for all requirements of the POS System not outlined already in the use cases.

<u>Functionality</u>

Security: requires users to authenticate themselves
Customizable Rules: the ability to change the rules of the system
Error logging and handling: the ability to track and display any errors the system encounters

Usability

Human Factors: the customer will see a display of the POS System results so the text must be visible from within three feet of the screen and colors that are hard to see or may be related to color blindness should be avoided.

The system must also act quickly and accurately with its processes so that the customer does not have to wait on the system.

Reliability

The system must be able to recover from any crash or failure by recovering its former state with as little data loss as possible and must correctly log the issues that caused the crash so that the problem can be resolved.

Additionally, the system must perform its processes quickly. We hope to have a system that has a transaction time of below 30 seconds 85% of the time.

Supportability

The store must be able to modify the rules and processes of the system to fit their unique needs. The client must also be able to configure the system itself so that it may fit properly in their area of business

Implementation Constraints

This implementation uses Java and will be restricted by any systems that do not support this code base.

Purchased Components

Tax Calculator: must support calculators for different countries
Credit Card Authorizer: must support most accepted credit cards

Interfaces

Hardware: Touch screen monitors for cashier use, barcode item scanner, credit card reader, receipt printer, signature reader
Software: the interfaces must support multiple types of systems and interfaces

Application Specific Domain Rules

ID	Rule	Changeability	Source
RULE1	Purchase Discount Ex. Employee - 15% off Senior Citizen - 10% off	High. These rules vary from store to store and may be changed during operation	Store policy
RULE2	Product Discounts Ex. Buy one, get one free	High. These rules vary from store to store and may be changed during operation	Store Policy
RULE3	Business Discounts Ex. Holiday Sales Timed Events	High. These rules vary from store to store	Store Policy
RULE4	Credit payments require customer signature	Low. This rule will not be changed in the near future	Mandatory credit policy
RULE5	Tax rules	Medium. Tax laws can change every year in a state	State Law

Legal Implications

Tax laws for a state must be followed and supported by the system.

All licenses for external software must be purchased or software must be open source

Information in Domains of Interest

Pricing: all products have a price that is tied directly to them and a permanent discount can be applied to them.

Credit and Debit payment: when a credit or debit is approved, the credit company pays the store directly, not the consumer. The consumer is responsible for paying back the credit company, but this is outside the consideration of the store

Sales Tax: these calculations can be complex and outsourcing them to a separately purchased module is the generally accepted practice.

Item ID: The system supports item id’s that are chosen by the store implementing it. They can follow any protocol as long as the id is a sequence of digits.