

|  |  |  |  |
| --- | --- | --- | --- |
| **Academic Year** | **Module** | **Assessment Number** | **Assessment Type** |
| A19 | Software Engineering (ADipIT01) | A1 | Group Report |

|  |  |
| --- | --- |
| Group Name:  Student Name: | Paranoid  Aashish Khanal(NP03A180074)  Arun Shrestha(NP03A180081)  Barsha Bhandari(NP03S190035)  Bishesh Adhikari(NP03A180063) |
| Section: | ADC7 |
| Lecturer: | Er. Lokesh Gupta |
| Submitted on: |  |

REVISION

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| S.N. | Date | Work | Feedback | Remarks |
| 1. | 2019-12-29 | -Introduction  -Fact Finding Technique  -SRS  -Diagrams | -Change format of written name  -Add feed back  -Use prefix before name  -add fact finding technique  -add functional decomposition diagram |  |
| 2. | 2020-01-06 | -Data Dictionary | -finish upto UI |  |
| 3. | 2020-01-13 | -Class Diagram  -State Transition Diagram  -Sequence Diagram  -Wireframe | -redesign wireframe |  |
| 4. | 2020-01-20 | -Wireframe |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

# Introduction

Valley Mart Departmental Store is a one stop destination for shoppers throughout Kathmandu valley looking for everyday commodity products. Valley Mart is quite popular in the valley as it is favorably located, large, and has dedicated customer support. The store has wide range of items on sale, and has recently set up a clothing store as well, a recent addition to its quality service. The store recently decided to computerize its stock management and billing process, and upgrade to an obsolete desktop-based system specification are to be considered and digitized.

# Fact Finding Technique

Fact finding is procedure of assortment of information and data dependent on strategies which contain inspecting of existing records, look into, perception, polls, interviews, prototyping and joint prerequisites arranging. Framework investigator utilizes reasonable certainty discovering strategies to create and execute the current existing framework. Gathering required certainties are critical to apply apparatuses in System Development Life Cycle since instruments can't be utilized productively and successfully without appropriate extricating from realities. Actuality discovering strategies are utilized in the beginning period of System Development Life Cycle including framework investigation stage, structure and post usage audit. Realities remembered for any data framework can be tried dependent on three stages: information certainties used to make helpful data, process-capacities to play out the targets and interface-plans to connect with clients (UK Essay, 2016).

The following are some popular fact-finding techniques:

## Examining Documentation

• It can be helpful to increase some knowledge concerning how the need for a database emerged.

• It can helps to identify the part of the organization associated with the problem.

• It can be useful to comprehend the present framework.

## Interviewing

• It empowers assortment of data from people up close and personal.

• It includes finding out facts, verifying facts, clarifying facts, generating enthusiasm, getting the end-user involved, identifying requirements, and gathering ideas and opinions.

## Observing the Organization in Operation

• Conceivable to either participate, or watch, an individual perform exercises to find out about the framework.

• Useful when legitimacy of information gathered is in question or when the complexity of certain parts of the framework prevents a reasonable clarification by the end-clients.

## Research

• Use computer trade journals, reference books, and the internet (including client groups and bulletin boards).

• Give data on how other have tackled same issues, in addition to whether programming package exist to sole or even mostly solve the problem.

## Questionnaires

• Conduct surveys through questionnaires, which are special-purpose documents that allow facts to be gathered from a large number of people while maintaining some control over their responses.

Above are some technique of fact-finding. But we used Research and Interview method to do this project.

* Research

No one is talented enough in every field and same goes for the programmers and developers. Since this case is related to a department store, it mainly involves the accounting portion. So, we should have knowledge about accounting terms and also implementing accounting principles in a program. Internet, text books related on fundamental of accounting helps us a lot to complete this project.

* Interview

Every other requirement elicitation in SDLC works better on its own way however we chose sample questionnaire because it serves as pre implementation checklist and covers key SDLC areas and points that should be considered by project team before software development. These questions should be completed before system go-live.

Some of the sample questions that match every organization are: Was a project plan developed and approved prior to the project? Were periodic meetings held to discuss the progress of the project? And many more covering the organizational topics.

Learning to construct quality questions is a skill that takes time and practice. Our time and the time of our customer is on the line, so it makes sense to do all you can to ask question that provide accurate, insightful responses. So through these surveys the analysis phase can be maximized and save time in design and implementation phase as well.

# Software Requirement Specification (SRS)

The document which describes the nature of a project, software or application is called software requirement specification. It can also be defined as a manual of a project provided it is prepared before you kick-start a project/application. A software document is primarily prepared for a project, software or any kind of application.

A Software requirement specification document describes the intended purpose, requirements and nature of a software to be developed. It also includes the yield and cost of the software (Bandakkanavar, 2018).

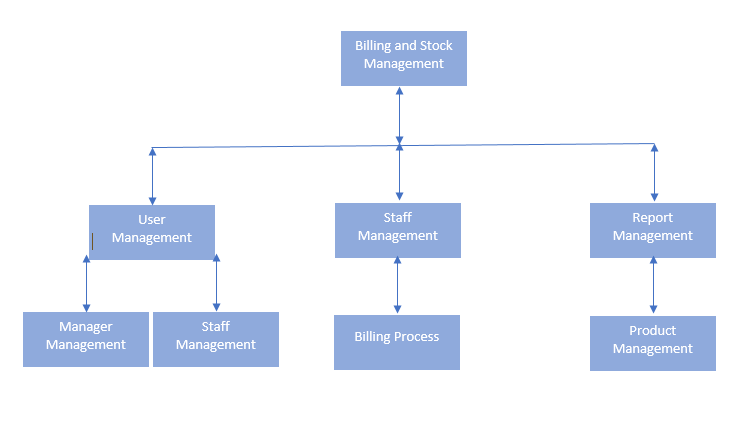
## System Requirement Specification (SRS) Legend

SM – F - 1





# Functional Decomposition Diagram

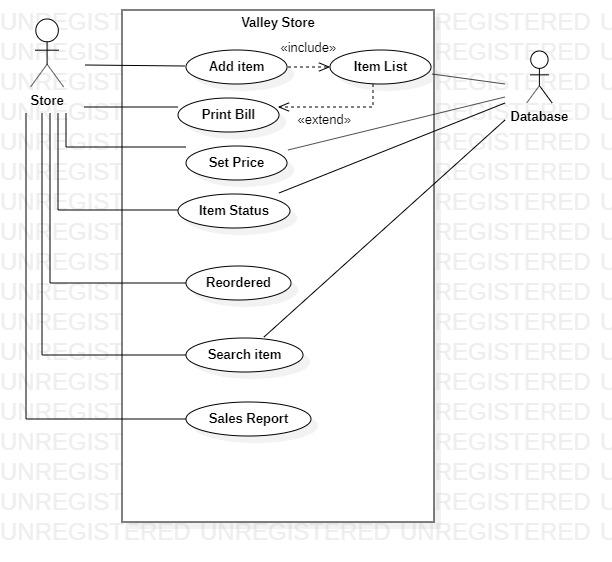


# Requirement Specification System (SRS) Table

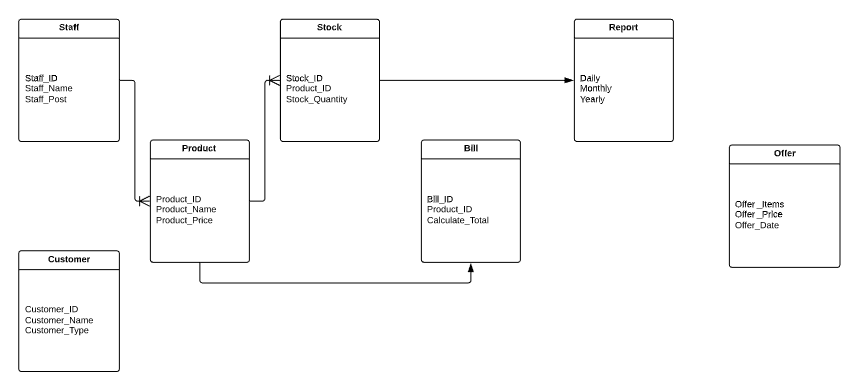
|  |  |  |
| --- | --- | --- |
| Requirement ID | Requirement Specification | Moscow |
| SM – F -1 | System Should allow staff to record items for stock management | Must have |
| SM – F -1.1 | System Should allow staff to manage occasional packages | Could have |
| SM – F -2 | System automatically generate total amount, discount, VAT, and deducts quantity in the Item’s stock | Must have |
| RM – F -1 | System Should list Items with status expired, items to be reordered, items with maximum sales, items with minimum sales | Should have |
| RM – F -2 | System should allow users to generate a report of all the sales as daily, monthly and sales of particular item. | Must have |
| RM – F -2.1 | System should allow users to record items for stock  management. | Could have |
| RM – F -3 | System should allow store, manage occasional packages such as add items to  the packages, set Price for the package, set expiry Date | Must have |
| UM – F -1 | System Should allow staff to login into dash board through login portal. | Must have |
| UM – F -2 | System should notify store to restock the products | Could have |
| RM – F -4 | System allows users to generate a report of all the sales | Must have |
| SM – F -3 | System should allow staff to check product details. | Must have |
| UM-NF-1 | System must be interactive and delay must be less. | Must have |
| RM-NF-1 | System must receive the information without any changes in information | Must have |
| SM-NF-1 | System must be reliable and must have strategy for error detection and a strategy for correction during securing the sensitives details | Must have |
| SM-NF-2 | System must have the ability to send information again if the internet service gets disrupted while sending information | Could have |
| SM-NF-3 | System program must react accordingly and transverse quickly between its states | Must have |
| SM-UF-1 | System should display all the feature and product | Must have |
| UM-UF-2 | System must allow the user to undo the changes that they have done earlier. | Could have |
| RM-UF-1 | System should not be interrupted while the multiple user is using the same page. | Must have |
| UM-UF-3 | System should allow the user to edit their personal infqormation. | Must have |
| UM-UF-3 | System should inform the user to renew or update. | Must have |

# System Modeling

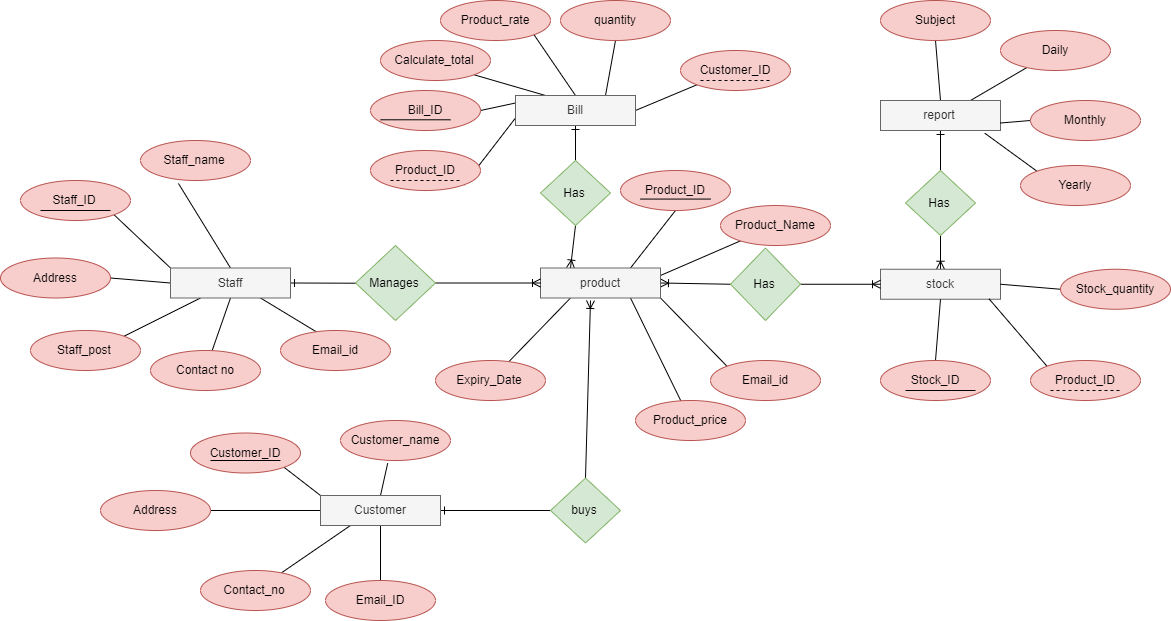
## Context Model (Use Case Diagram)



## 4.2. Conceptual Diagram



## 4.3. Detail ERD



# Data Dictionary

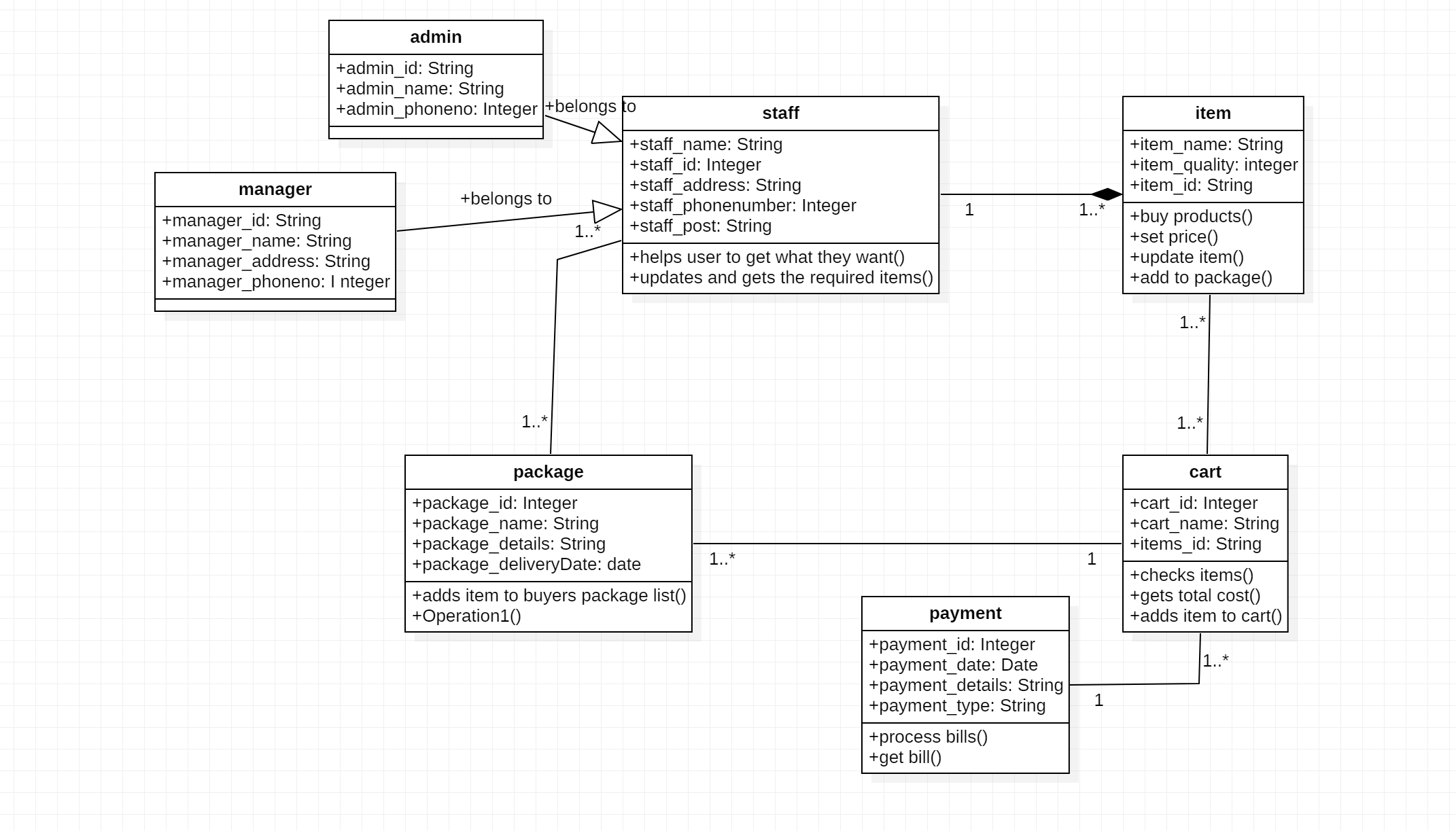
|  |  |  |  |
| --- | --- | --- | --- |
| Entity name | description | aliases | occurrence |
| staff | General term used for the staff of SRS | employee | Each member of the staff will be working for the client’s satisfaction and providing them the information that they need. |
| product | General term for the products for sale in the SRS | product | Each product will be labelled with an id so that they can easily be identified when a user asks for it. |
| customer | General term for the user of the SRS | user | Each member of the customer will be assigned with an id so that the staffs can help them to get what they have ordered. |
| bill | General term for the price of the product purchased by the customer | expenses | Each bill contains its own number and the amount that the user needs to pay after the purchase that he makes. |
| stock | General term used for the remaining products. | stock | Each stock contains of an id which notifies the user and the staff about the products that are available in stock. |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Entity name | multiplicity | relationship | Multiplicity | Entity name |
| staff | 0.1  0.1 | Manages  makes | 0.\*  0.\* | Products  bills |
| products | 0.\*  0.\* | Bought by  stocked | 0.1  1.\* | Customer  stock |
| bill | 0.1 | Made for | 1.\* | products |

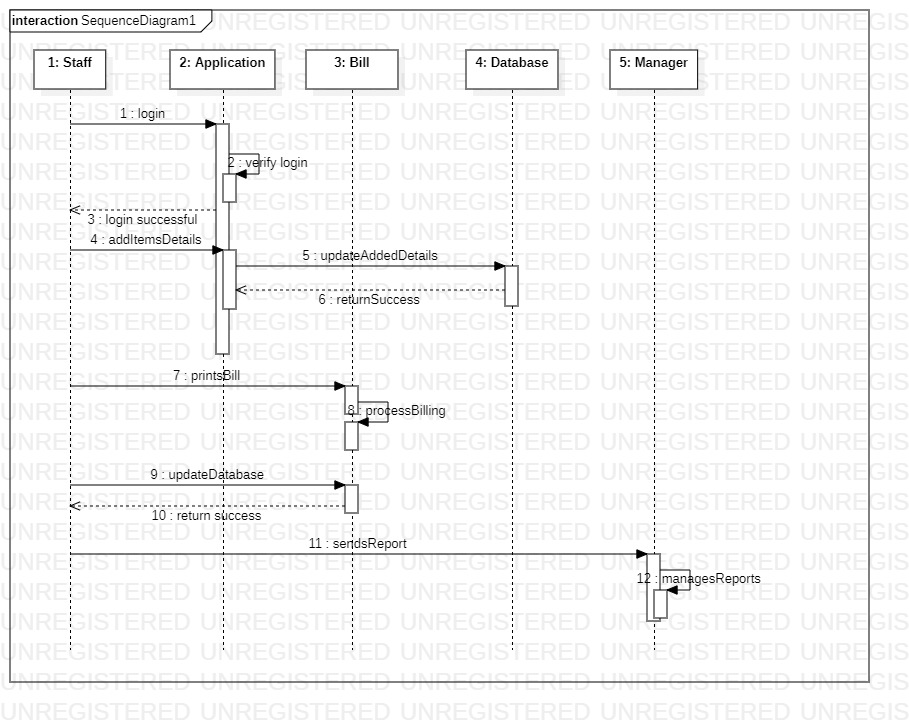
# 

# Process Model

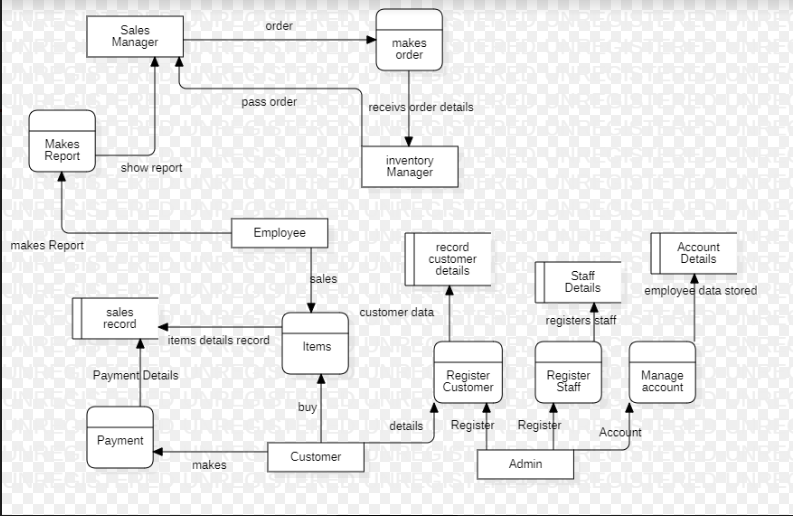
## Class Diagram



## Sequence Diagram



## Upper Level DFD



## 

## 

## 

## State Transition Diagram



# 

# 

# 

# 

# 

# 

# 

## 

# 

# UI Model

## Wireframe Design

