Stock Market

The retail industry face a challenge in man inventory. Create a Stock market maintenan for the same track levels, handle product

1. Introduction

1.1 Purpose

The document serves as a guide requirement specification for development of Stock maintenan

1.2 Scope

The document defines overall working & functionality performance a non function attribute

1.3 Overview

The Stock maintenance system will efficieny manage inventory & back Stock.

2. General description:

The Stock maintenance system will efficienty manag inventory. Software solution automate inventory manag process for business.

3. Functional Requirements

Stock tracking

Inventory Management

Track Stock level

Reporting & Analysis

4. Interface Requirements
   User interface
   Database interface
   Integration Interface

5. Performance Requirement
   Response time should be minimum.
   Real time updates
   Data entgrebly

6. Design Constraints
   System should be compitable with various
   Software & hardware.
   Robust Security measures

7. Non functional attributes
   Security
   portability
   Scalability
   Reliability.

8. Preliminary Schedule & budget
   eliminate time : 10 months
   estimated budget : 80000

5) Stock Market

Stock
- +id
- +desc
- +item
- +quantity
- +type
- +add()
- +update()
- +delete()

includes

Item
- +item_id
- +item_name
- +item_amt
- +addItem()
- +removeItem()
- +editItem()
- +searchItem()

+boughtFrom

1..*

Seller
- +sellerId
- +sellerName
- +rating
- +address
- +addSeller()
- +deleteSeller()
- +viewSeller()

+boughtBy

Purchase
- +item
- +date
- +qty

Store
- +id
- +name
- +address
- +status
- +removeStore()
- +addStore()
- +updateStore()
- +viewStore()

«enumeration»
Status
Open
Closed
Renovation

SuperMarket
- +description
- +size
- +employees
- +edit()

DepartmentStore
- +owner
- +employees
- +size
- +add()
- +update()
- +delete()

Warehouse
- +size
- +Attribute1
- +view()

Customer
- +cust_id
- +cust_name
- +cust_phone
- +cust_address
- +cust_phoneNo
- +addCustomer()
- +viewCustomer()
- +deleteCustomer()
- +updateCustomer()

+customer_id   +pays

1   0..*

Bill
- +id
- +date
- +totalAmount
- +update()
- +add()
- +delete()