

SOFTWARE REQUIREMENT SYSTEM FOR PERSONAL INVENTORY MANAGEMENT SYSTEM

1. INTRODUCTION

1.1. PURPOSE

Personal Investment Management System (PIMS) is designed to assist users in keeping track of their money invested in institutions such as banks and stock exchanges.

This document is intended to outline the features of PIMS in order to serve as a reference.

On one hand it serves as a guide to the developers, and a website validation document for the prospective client on the other.

1.2. SCOPE

1.2.1. IN SCOPE FEATURES

1.2.1.1. Managing a single user's investment, which includes maintaining bookkeeping information about entities such as Portfolio, Security, and Transaction.

1.2.1.2. Calculation of the investor's net worth and rate of return on investment (ROI).

1.2.1.3. Giving the user alerts on requests

1.2.1.4. Obtaining current share prices from the internet.

1.2.1.5. User authentication.

1.2.2. OUT OF SCOPE FEATURES

1.2.2.1. Features for actual securities purchase and sale. That is, the actual buying and selling of shares/security occurs outside of PIMS.

1.2.2.2. Gains/losses tax computations

1.2.2.3. Any market related prediction.

1.3. DEFINITIONS, ACRONYMS, AND ABBREVIATIONS

1.3.1. ACRONYMS AND ABBREVIATIONS

1.3.1.1. SRS: website Requirement Specification

1.3.1.2. WWW: World Wide Web

1.3.1.3. ROI: Rate of Investment

1.3.1.4. GUI: Graphical User Interface

1.3.2. DEFINITIONS

1.3.2.1. Transaction: A true event involving the movement of personal funds. In the context of shares, it is the purchase or sale of a group of shares in the same company, whereas in the context of a bank, it is the deposit or withdrawal of funds to/from one's account.

1.3.2.2. Security: A collection of all transactions involving a company share or a bank account.

1.3.2.3. Portfolio: A set of securities

1.3.2.4. Net-Worth: The sum total of the investor's money in the form of shares and bank balances.

1.3.2.5. ROI: The user's interest gained for a specific investment. In the case of a bank account, it is the annual interest, and in the case of a company share, it is the dividend.

1.4. REFERENCES

1.4.1. Appendix A: Formula for ROI calculation for shares.

1.4.2. Appendix B: Formula for Net-Worth calculation.

1.4.3. Appendix C: User Screens.

1.5. OVERVIEW

The remainder of this SRS is structured as follows: Section 2 provides an overview of the website. It describes the user's expected level of proficiency, some general constraints encountered while developing the website, and some assumptions and dependencies that are assumed. Section 3 outlines the specific requirements that the website is expected to meet. Various use cases provide functional requirements. There are also some performance requirements and design constraints.

Section 4 discusses some potential future system extensions. Finally, the appendices in Section 5 describe the formula for calculating ROI for shares and user screens, respectively.

2. OVERALL DESCRIPTION

2.1. PRODUCT PERSPECTIVE

2.1.1. PIMS is aimed at a person who has a significant number of investments in the stock market and banks and thus requires website assistance for bookkeeping and investment computations. For the purposes stated above, PIMS should be user-friendly, 'quick to learn,' and dependable website.

2.1.2. PIMS is designed to be a stand-alone product that does not rely on the availability of any other website. It should be able to run on both UNIX and Windows platforms.

2.2. PRODUCT FUNCTIONS

Class of use cases	Use cases	Description of use cases
Use cases related to System authorization	Login	<i>Login into PIMS</i>
	Change Password	<i>Change PIMS password</i>
Use cases related to portfolios	Create portfolio	<i>Creates a new portfolio</i>
	Rename portfolio	<i>Rename an existing portfolio</i>
	Delete portfolio	<i>Delete an existing portfolio</i>
Use cases related to securities	Create security	<i>Creates a new security in a given portfolio</i>
	Rename security	<i>Renames an existing security</i>
	Delete security	<i>Deletes an existing security</i>
Use cases related to transactions	Add transaction	<i>Add a transaction to a security</i>
	Edit transaction	<i>Edit an existing transaction</i>
	Delete transaction	<i>Delete an existing transaction</i>
Use cases related to Information display	Display investment	<i>Display information of the entire investment</i>
	Display portfolio	<i>Display information about a given portfolio</i>
	Display security	<i>Display information about a given security</i>
Use cases related to computations	Compute net-worth	<i>Compute net -worth of investment/portfolio/security</i>
	Compute ROI	<i>Compute ROI of a given security</i>
Use cases related to share prices	Get current share price	<i>Download the current share price from the net</i>
	Edit share price	<i>Edit the price of a share already present in the list</i>
Use cases related to alerts	Set alerts	<i>Set alert giving date and details</i>
	Show alerts	<i>Show all the pending alerts</i>
	Delete alerts	<i>Delete an already set alert</i>

3. SPECIFIC REQUIREMENTS

3.1. FUNCTIONAL REQUIREMENTS

We describe the functional requirements by giving various use cases:

3.1.1. USE CASE 1: LOG IN

PRIMARY ACTOR: User

PRE-CONDITION: Nil

MAIN SCENARIO:

1. Start the application. User prompted for login and password.
2. User gives the login and password.
3. System does authentication.
4. Main screen is displayed.

ALTERNATE SCENARIO:

1. Authorization fails
 - (a) Prompt the user that he typed the wrong password
 - (b) Allow him to re-enter the password. Give him chances.

3.1.2. USE CASE 2: CREATE PORTFOLIO

PRIMARY ACTOR: User

PRE-CONDITION: User logged in

MAIN SCENARIO:

1. User initiates the “create portfolio” functionality.
2. System asks the user for the portfolio name.
3. User enters the portfolio name.
4. An empty portfolio is created.

ALTERNATE SCENARIO:

1. Portfolio with same name exists

(a) System asks the user for a different name.

(b) The user enters a different name.

(c) An empty portfolio gets created.

3.1.3. USE CASE 3: RENAME PORTFOLIO

PRIMARY ACTOR: User

PRE-CONDITION: User logged in

MAIN SCENARIO:

1. User initiates the “rename portfolio” functionality.
2. System asks for the portfolio to be renamed and the new name.
3. User enters the new name.
4. Portfolio is renamed.

ALTERNATE SCENARIO:

(a). The portfolio whose name is supposed to change does not exist.

1. Renaming fails, the error message is displayed.

(b). Portfolio with the same new name exists.

1. Renaming fails, the error message is displayed.

3.1.4. USE CASE 4: DELETE PORTFOLIO

PRIMARY ACTOR: User

PRE-CONDITION: User logged in

MAIN SCENARIO:

1. User initiates the “delete portfolio” functionality.
2. System asks for the name of the portfolio.
3. The portfolio is deleted.

ALTERNATE SCENARIO:

- (a). Portfolio does not exist.
1. Deletion fails, error message is displayed.

3.1.5. USE CASE 5: ADDING A TRANSACTION

PRIMARY ACTOR: User

PRE-CONDITION: User logged in

MAIN SCENARIO:

1. User selects the portfolio and the security.
2. User initiates the “add transaction” functionality.
3. System prompts the user for following details, depending on the security type
 - (i) If the security is of ‘share’ type:
 - a. Number of shares transacted
 - b. Type of transaction: buy or sell
 - c. Price of one share
 - d. Date of transaction
 - e. Any detail user might want to enter.
 - (ii) If the security is of bank type:
 - a. Amount of money transacted

- b. Type of transaction: credit or debit.
- c. Date of transaction
- d. Any detail user might want to enter.

New transaction is created.

ALTERNATE SCENARIO:

(a). Invalid attribute value is entered. For example, field (i)(a) is not an integer or (i)(c) is not a float.

- 1. User is shown the error displayed and prompted to enter the incorrect fields again.

3.1.6. USE CASE 6: EDITING A TRANSACTION

PRIMARY ACTOR: User

PRE-CONDITION: User logged in

MAIN SCENARIO:

- 1. User selects the portfolio, security and transaction.
- 2. User initiates the “edit transaction” functionality.
- 3. System asks the user to enter new fields for the transaction.
The fields are the same as given in Use case 1.
- 4. The modified transaction is saved.

ALTERNATE SCENARIO:

Invalid attribute value entered. For example field (i)(a) is not an integer or (i)(c) is not a float.

- 1. User is shown the error displayed and prompted to enter the incorrect fields again.

3.1.7. USE CASE 7: DELETING A TRANSACTION

PRIMARY ACTOR: User

PRE-CONDITION: User logged in

MAIN SCENARIO:

1. User specifies the portfolio, security and transaction.
2. User initiates the “delete transaction” functionality.
3. The transaction is deleted.

3.1.8. USE CASE 8: DISPLAY INVESTMENT

PRIMARY ACTOR: User

PRE-CONDITION: User logged in

MAIN SCENARIO:

1. User selects the option of viewing the Investment (from the left pane, refer user screens in Appendix C).
2. System computes the net worth for each portfolio.
Refer to use case 16.
3. System displays the name of all portfolios with their net-worth. It also shows the net-worth of the entire investment by adding up the net-worth of each portfolio.

3.1.9. USE CASE 9: DISPLAY PORTFOLIO

PRIMARY ACTOR: User

PRE-CONDITION: User logged in

MAIN SCENARIO:

1. User selects the option of viewing a particular portfolio (selects a portfolio from the left pane, refer user screens in Appendix A).
2. System computes the net-worth and ROI of each security. Refer use cases 16 and 17.
3. System displays the securities in the portfolio with the net-worth and ROI computed. It also displays the net-worth of the portfolio by summing up the net-worth of each security.

3.1.10. USE CASE 10: COMPUTE NET WORTH

PRIMARY ACTOR: System

PRE-CONDITION: User logged in and investment/portfolio/security
Specified

MAIN SCENARIO:

1. System computes net-worth for investment/portfolio/security specified (Appendix B shows the detailed method for computing the net-worth).

3.1.11. USE CASE 11: COMPUTE RATE OF INVESTMENT

PRIMARY ACTOR: System

PRE-CONDITION: User logged in and security specified

MAIN SCENARIO:

1. System computes ROI for the security specified (Appendix A shows the detailed method for computing the ROI).

3.1.12. USE CASE 12: GETTING THE CURRENT PRICES OF SHARES

PRIMARY ACTOR: User

PRE-CONDITION: User logged in and internet connection exists

MAIN SCENARIO:

1. User initiates the “download current prices” functionality.
2. The system downloads the current share prices from a particular Website.

ALTERNATE SCENARIO:

1. Not able to download (due to network failure, site down, ...)
(a) Display current value saved from before; ask the user to change it

3.1.13. USE CASE 13: EDITING THE SHARE PRICES

PRIMARY ACTOR: User

PRE-CONDITION: User logged in

MAIN SCENARIO:

1. User initiates the “Edit share prices” functionality.
2. The system shows the list of all the companies and their current share prices (as known to the system).
3. User edits the price of the company share he wants to edit.

ALTERNATE SCENARIO:

- 1.The entered price is invalid
 - (a) User is shown the error displayed
- 2.The company name does not exist in the list of companies
 - a)This is not handled by PIMS. PIMS is customized for the 760-800 odd companies registered in the NSE.

3.1.14. USE CASE 14: SETTING ALERTS

PRIMARY ACTOR: User

PRE-CONDITION: User logged in

MAIN SCENARIO:

1. User initiates the “set alert” functionality.
2. The system asks the user for the date and details of the alert.
3. The alert is set.

3.1.15. USE CASE 15: SHOW THE PENDING ALERTS

PRIMARY ACTOR: User

PRE-CONDITION: User logged in

MAIN SCENARIO:

1. User initiates the “show pending alerts” functionality.
2. System shows the pending alerts.

3.1.16. USE CASE 16: DELETE ALERTS

PRIMARY ACTOR: User

PRE-CONDITION: User logged in

MAIN SCENARIO:

1. User initiates the “delete alerts” functionality.
2. User is asked for the alert to be deleted.
3. The alert is deleted.

3.2. PERFORMANCE REQUIREMENTS

3.2.1. 90% of the responses should be within 2 sec, except for downloading current prices for which more time is acceptable.

3.3. DESIGN CONSTRAINTS

3.3.1. SECURITY: The files containing information about securities and portfolios should be protected from malicious deformations.

3.3.2. FAULT TOLERANCE: Data should not be corrupted in the event of a system crash or power outage.

3.4. EXTERNAL INTERFACE REQUIREMENTS

The user interface is divided vertically into two panes. The Investment tree is displayed in the left pane and expands and contracts in response to user input. The information related to the investment/portfolio/security specified on the left pane is displayed in the right pane. Appendix B depicts the intended user interface.

4. FUTURE EXTENSIONS

4.1. PIMS is designed to be a single-user application. A potential future enhancement would be to make it multi-user.

5. APPENDIX

5.1. APPENDIX A

5.1.1. Assume that the following were the characteristics of the I th transaction for a specific company's share.

5.1.2. The amount of money exchanged = m_i

5.1.3. Time elapsed between the date of the transaction and the day on which the ROI is calculated = n_i

5.1.4. Transaction type = buy or sell

5.1.5. If the transaction was a buy, set $\text{sgn}(i)$ to +1, otherwise to -1.

5.1.6. Then, the rate of investment, r , is got by solving the equation

5.1.7. where M is the total amount in that share and summation is performed across all transactions.

5.1.8. The ROI formula for banks is as follows:

The ROI for a bank is constant, namely the bank's interest rate.

5.2. APPENDIX B

5.2.1. The formula for calculating net-worth of a security:

5.2.1.1. If the security is of the type "share," net-worth = $C \cdot N$, where C is the current value of the security's share and N is the number of shares that the user owns.

5.2.1.2. If the security is of the "bank" variety, the net-worth is simply the total amount of money in the bank.

5.2.2. The formula for calculating net-worth of a portfolio:

5.2.2.1.Net worth of a security = \sum Net-worth of i th security. (where the summation ranges over all the securities.)

5.2.3. The formula for calculating net-worth of entire investment:

Net worth of entire investment = \sum Net-worth of i th portfolio. (where the summation ranges over all the portfolios.)

5.3. APPENDIX C

5.3.1. The diagram below shows the outline of the main user screens.



Investment Screen (displaying the portfolios and net-worth)

Show Alerts	Set Alerts	Delete Alerts	Download Prices	Edit Prices
Change Password	Help	About	Exit	
Investment <ul style="list-style-type: none"> Port1 Port2 Infosys SBI 	Securities UBI	Worth 1629.74	Monthly ROI 0.19%	
	Net Worth 1629.74			
<div> <div>Add Security</div> <div>Delete Security</div> <div>Rename Security</div> </div>				

Portfolio Screen (displaying the securities and)

Show Alerts	Set Alerts	Delete Alerts	Download Prices	Edit Prices												
Change Password	Help	About	Exit													
Investment <ul style="list-style-type: none">Port1<ul style="list-style-type: none">UBIPort2<ul style="list-style-type: none">InfosysSBI	<table><thead><tr><th>Notes</th><th>Date</th><th>Amount Transacted</th><th>TransactionType</th></tr></thead><tbody><tr><td>"deposit"</td><td>25-1-2002</td><td>1000.0</td><td>credit</td></tr><tr><td>"withdrawal"</td><td>25-6-2002</td><td>500.0</td><td>credit</td></tr></tbody></table>	Notes	Date	Amount Transacted	TransactionType	"deposit"	25-1-2002	1000.0	credit	"withdrawal"	25-6-2002	500.0	credit			
Notes	Date	Amount Transacted	TransactionType													
"deposit"	25-1-2002	1000.0	credit													
"withdrawal"	25-6-2002	500.0	credit													

Add Transaction

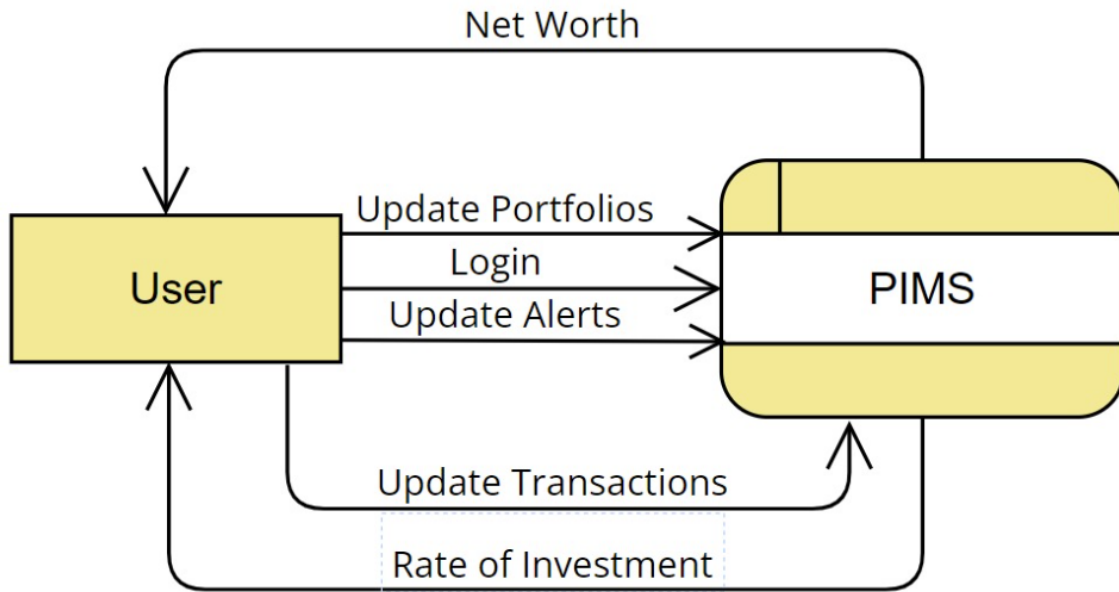
Delete Transaction

Edit Transaction

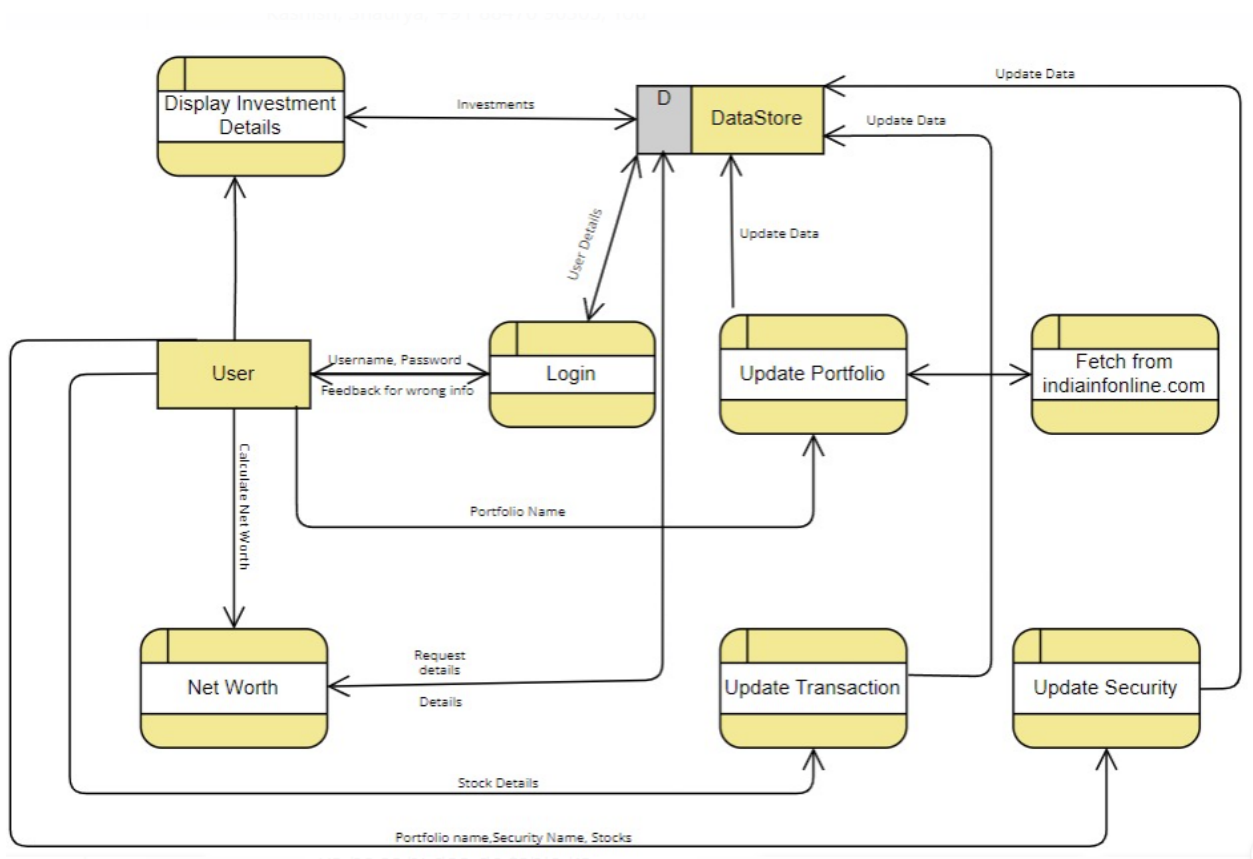
Security screen (displaying all the transactions)

DFDs

LEVEL 0



LEVEL 1



UML

