

PIMPRI CHINCHWAD EDUCATION TRUST'S

PIMPRI CHINCHWAD COLLEGE OF ENGINEERING
DEPARTMENT OF COMPUTER ENGINEERING

A Software Engineering Case Study on

One Stop Shop

Presented by:

ASHWIN MOHAN – TECOA101
SALONI AVHAD – TECOA105
VIVEK BULANI – TECOA117
MOHIT BURKULE – TECOA118

INDEX

- ❖ Introduction
- ❖ Stakeholders
- ❖ SRS
- ❖ Requirements
- ❖ Process Model
- ❖ USE CASES
- ❖ Feasibility

Introduction

- The Traditional way of shopping utilizes more time. Every time the customer need to do shopping he has to go to the shops or market and perform the necessary actions, which may not be so feasible all the time. It may be a hard-hitting task for the customers and the shopkeepers too.
- Internet shopping system is specifically developed for the customers who can directly buy goods or materials from home through internet connection on a mobile or a system.
- Users will have the provision of giving ratings and reviews.
- The system removes the role of intermediate dealers and managers and connect customers directly to the manufacturers.
- The online shopping system presents an online display of an order cut off time and an associated delivery window for items selected by the customer. The system accepts the customer's submission of a purchase order for the item in response to a time of submission being before the order cut off time.
- Recommendations will be provided to customers on the basis of previous purchases, user reviews and ratings.
- Merchants can upload their products' images and give appropriate prices. The prices and validity will be checked at the administrator level.

Stakeholders

1. The customers.
2. The merchants.
3. Developers.
4. Security Consultants.
5. Customer care.
6. Transport suppliers.
7. Social media team.
8. Administrator.

Software Requirement Specifications

Purpose The purpose of this SRS is to specify the requirements of the web based software application, which is an online shopping system. This Software Requirements Specification provides a complete description of all the functions and specifications of modules. This document contains the software requirements of online shopping.

Project Scope The document is the one that describes the requirements along with interfaces for the system. This system allows the customer's to maintain their cart for add or remove the product over the internet.

System Description The Online Shopping system (OSS) application enables vendors to set up online shops, customers to browse through the shops, and a system administrator to approve and reject requests for new shops and maintain lists of shop categories. Also the developer is designing an online shopping site to manage the items in the shop and also help customers purchase them online without having to visit the shop physically. The online shopping system will use the internet as the sole method for selling goods to its consumers.

Functional requirements

Functional Requirements This section provides requirement overview of the system. Various functional modules that can be implemented by the system will be -

1. **Hardware Requirements** This system will work on client-server architecture. It will require an internet server and which will be able to run the application. The system should support some commonly used browser such as IE, Chrome etc.

2. Software Requirements

Registration If customer wants to buy the product then he/she must be registered, unregistered user can't go to the shopping cart.

1. **Login** Customer logs in to the system by entering valid user id and password for the shopping.
2. **Changes to Cart** Changes to cart means the customer after login or registration can make order or cancel order of the product from the shopping cart.
3. **Payment** For customer there are many type of secure billing will be prepaid as debit or credit card, post paid as after shipping, check or bank draft. The security will provide by the third party like Pay-Pal etc.
4. **Logout** After the payment or surf the product the customer will logged out.
5. **Report Generation** After all transaction the system can generate the portable document file (.pdf) and then sent one copy to the customer's Email-address and another one for the system data base to calculate the monthly transaction .

Non Functional Requirements

Availability The system should be available at all times, meaning the user can access it using a web browser, only restricted by the down time of the server on which the system runs. In case of a hardware failure or database corruption, a replacement page will be shown. Also in case of a hardware failure or database corruption, backups of the database should be retrieved from the server and saved by the administrator. Then the service will be restarted. It means 24 X 7 availability.

Reliability The system provides storage of all databases on redundant computers with automatic switch over. The reliability of the overall program depends on the reliability of the separate components. The main pillar of reliability of the system is the backup of the database which is continuously maintained and updated to reflect the most recent changes. Thus the overall stability of the system depends on the stability of container and its underlying operating system.

Non Functional Requirements

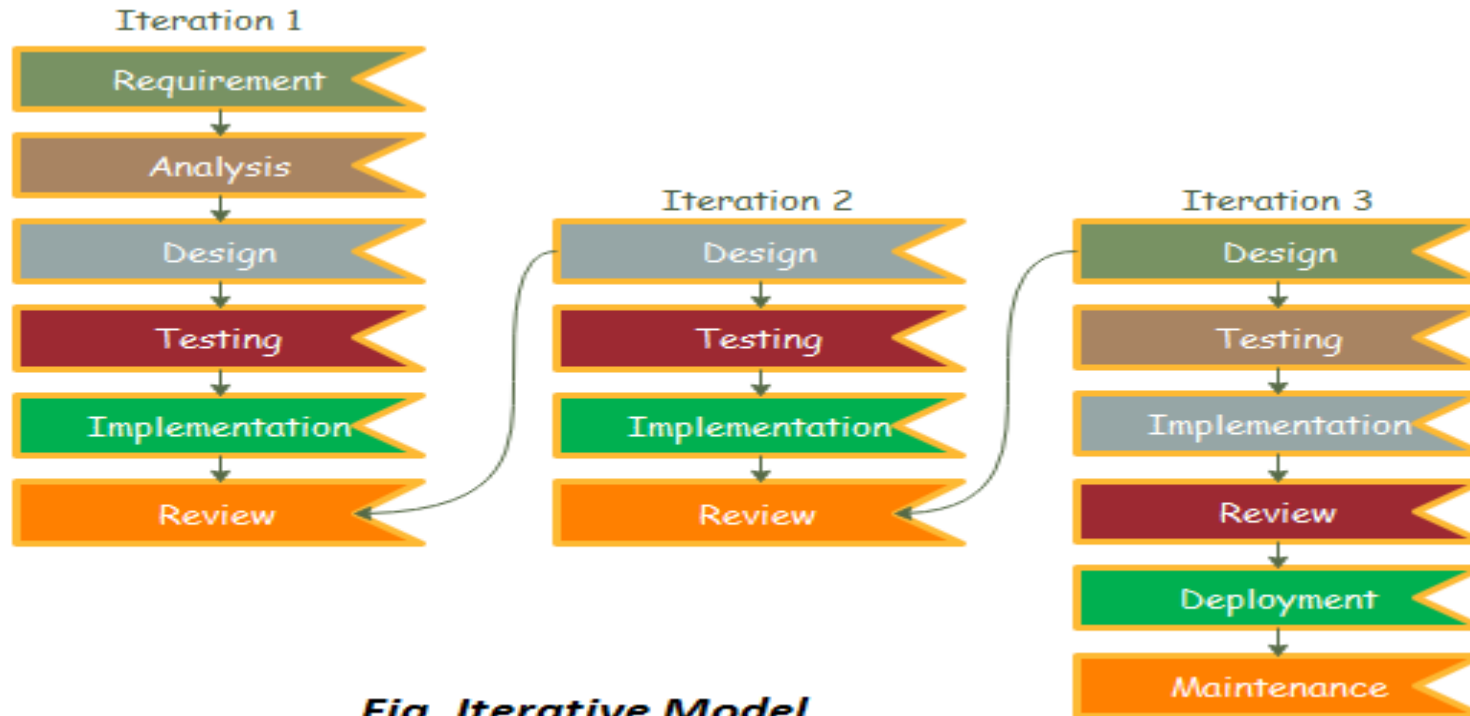
Performance There is no performance requirement in this system because the server request and response is depended on the end user internet connection.

Scalability The system is scalable. New features (modules) can be added whenever required.

Security The system will use SSL (secured socket layer) in all transactions that include any confidential customer information. The system must automatically log out all customers after a period of inactivity. The system should not leave any cookies on the customer's computer containing the user's password. The system's back-end servers shall only be accessible to authenticated administrators. Sensitive data will be encrypted before being sent over insecure connections like the internet.

Maintainability A commercial database is used for maintaining the database and the application server take care of the site. In case of a failure, a re-initialization of the program will be done. Also the software design is being done with modularity in mind so that maintainability can be done efficiently

Process Model : Iterative



Iterative process model

Iterative process starts with a simple implementation of a subset of the software requirements and iteratively enhances the evolving versions until the full system is implemented. At each iteration, design modifications are made and new functional capabilities are added. The basic idea behind this method is to develop a system through repeated cycles (iterative) and in smaller portions at a time (incremental).

An iterative life cycle model does not attempt to start with a full specification of requirements. Instead, development begins by specifying and implementing just part of the software, which can then be reviewed in order to identify further requirements. This process is then repeated, producing a new version of the software for each cycle of the model.

Consider an iterative life cycle model which consists of repeating the following four phases in sequence:

A Requirements phase, in which the requirements for the software are gathered and analysed. Iteration should eventually result in a requirements phase that produces a complete and final specification of requirements.

A Design phase, in which a software solution to meet the requirements is designed. This may be a new design, or an extension of an earlier design.

An Implementation and Test phase, when the software is coded, integrated and tested.

A Review phase, in which the software is evaluated, the current requirements are reviewed, and changes and additions to requirements proposed.

Why we have chosen iterative model?

Requirements of the complete system are clearly defined and understood.

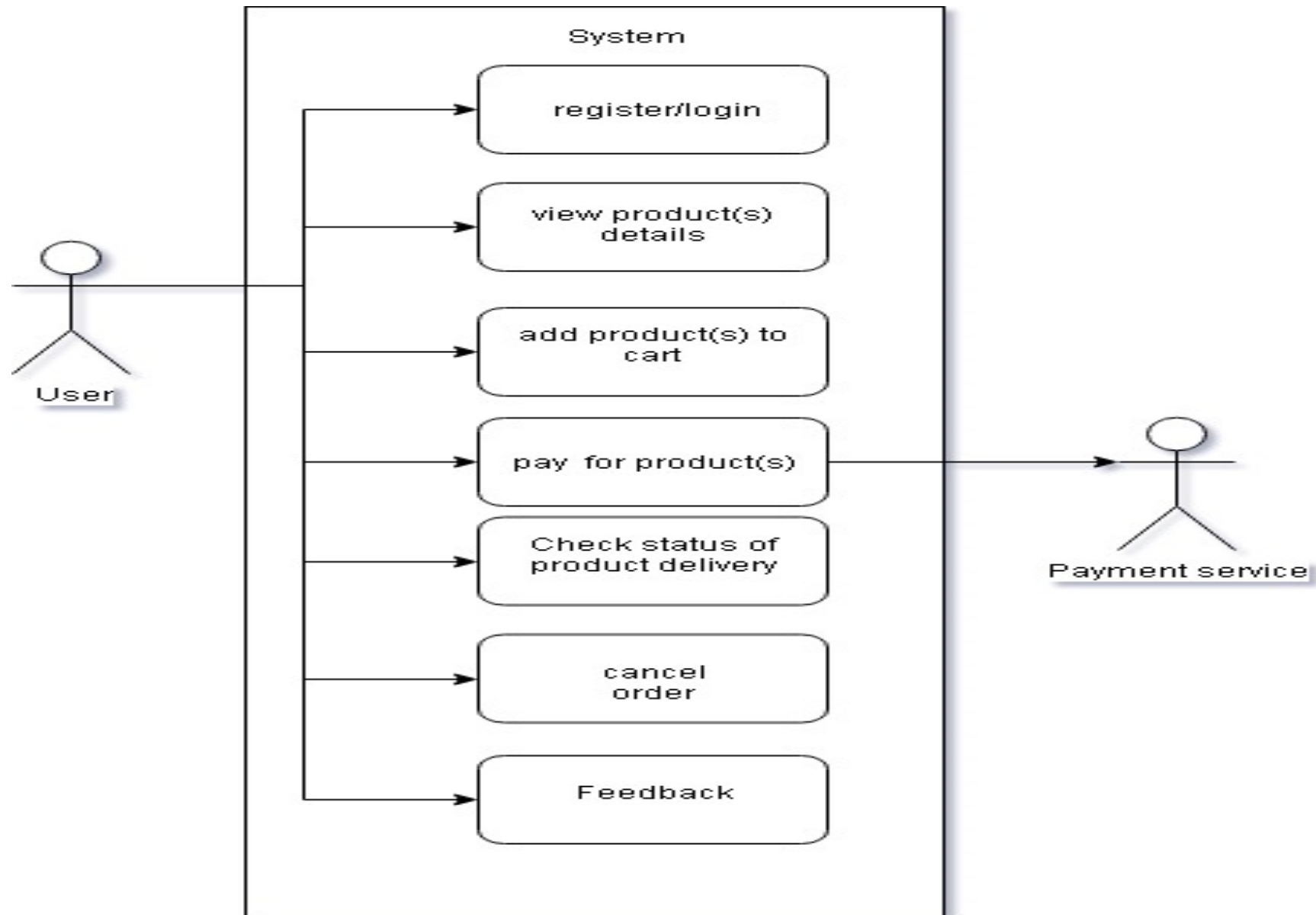
Major requirements are defined; however, some functionalities or requested enhancements may evolve with time.

There is a time to the market constraint.

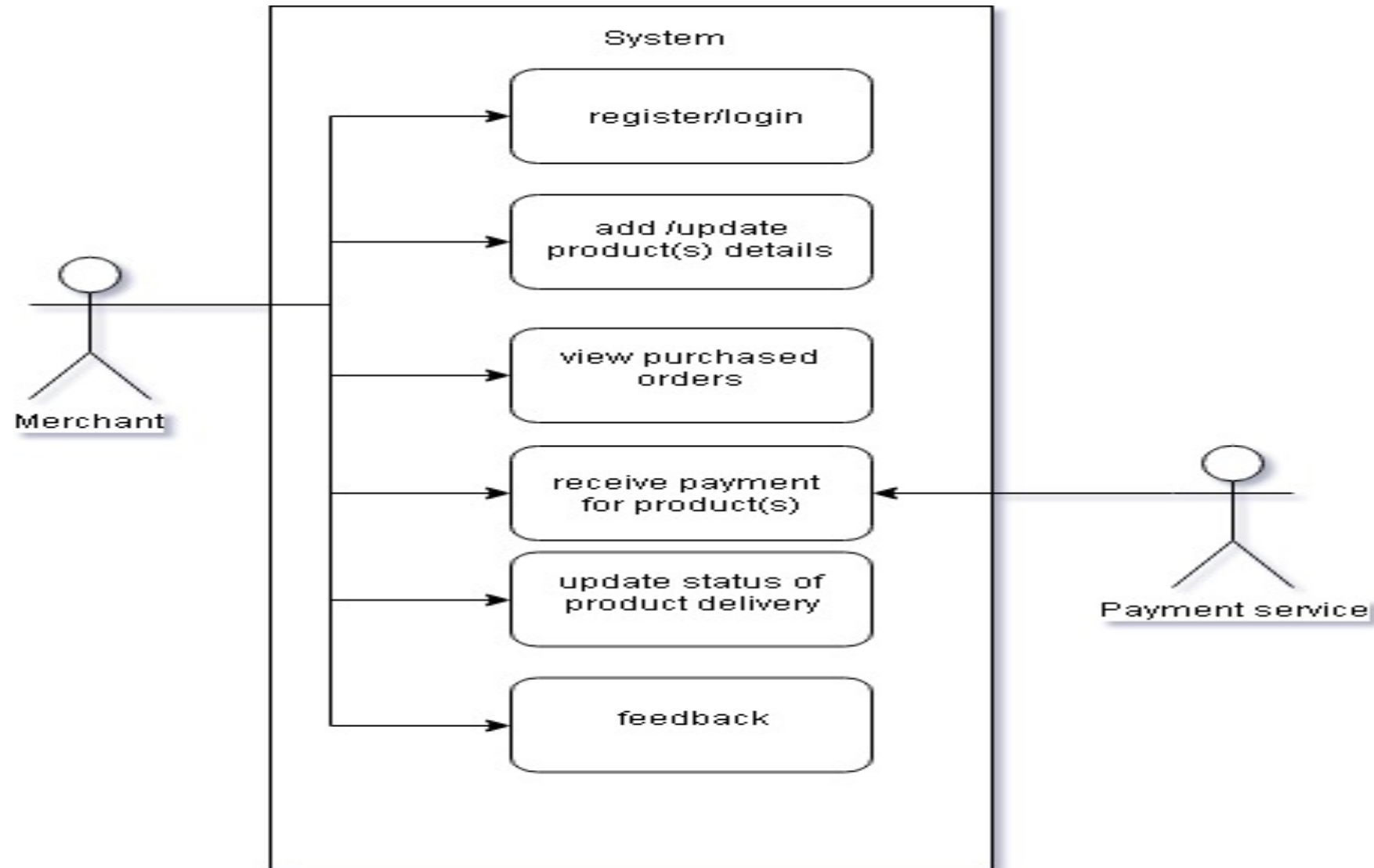
A new technology is being used and is being learnt by the development team while working on the project.

There are some high-risk features and goals which may change in the future.

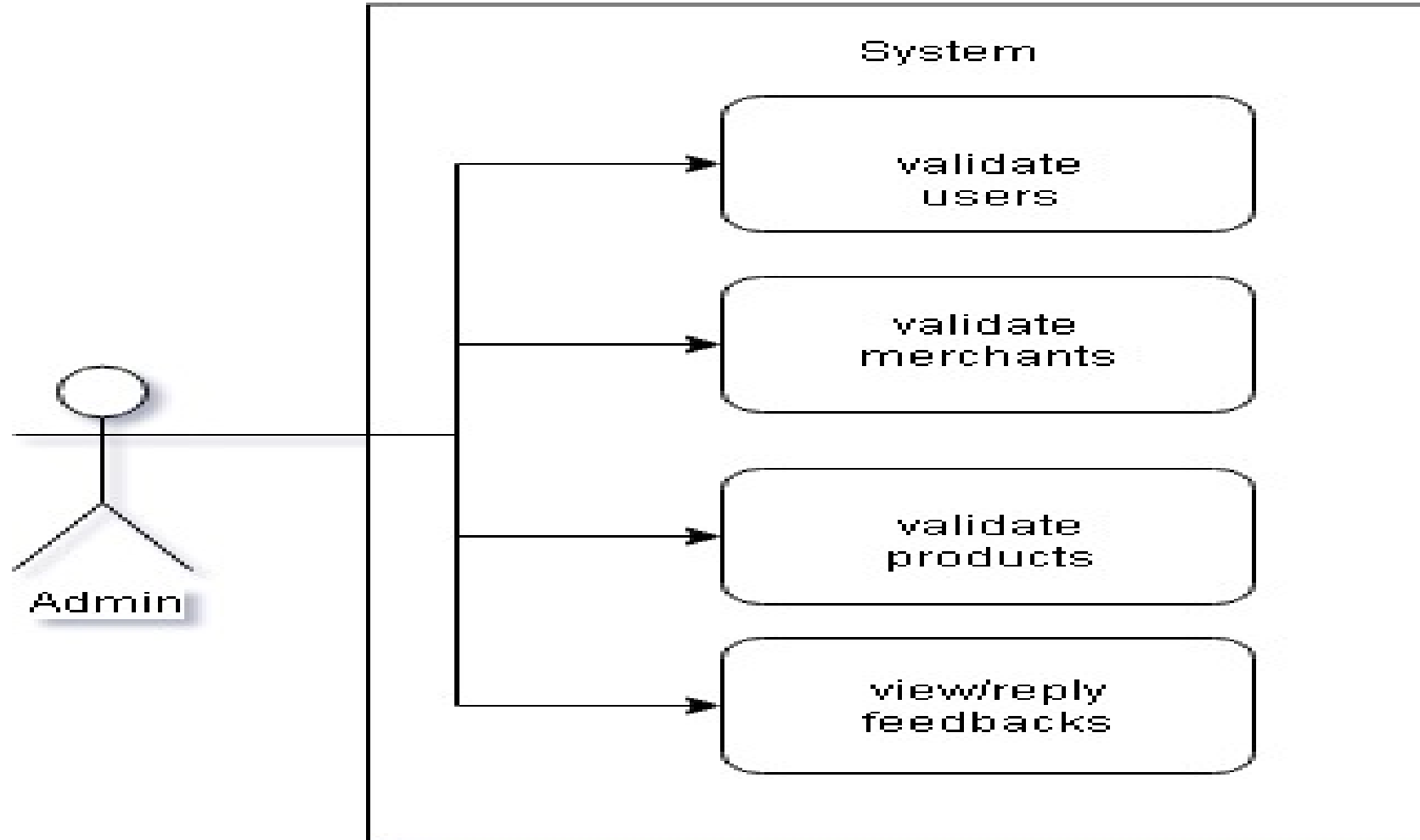
Use case - user



Use case - merchant



Use case - admin



Feasibility Report

Feasibility Report Feasibility is an important phase in the software development process it enables the developers to have an assessment of the product being developed It refers to the feasibility study of the product in terms of outcomes of the product, operational required for implementing it. Feasibility study should be performed on the basis of various criteria and parameters.

1. **Economic Feasibility:** It refers to the benefits or outcomes we are deriving from the product as compared to the total cost we are spending for developing the benefits are less then it is not feasible to develop the product.
2. **Operational Feasibility:** It refers to the feasibility of the product to be operational. Some products may work very well at the design and implementation but many fail in the real time environment. It introduces the study of human resources required and their technical expertise.
3. **Technical Feasibility:** The system is self-explaining and does not need any entire sophisticated training. A system has been built by concentrating on the graphical user interface concepts, the application can also be handled very easily with a novice user. The overall time that a user needs to get trained is less than 15 minutes. The system has been added with features of menu driven and button interaction methods, which makes him the master as he starts working through the environment. As the software that were used as developing this application are very economical and are readily available in the market the only time that is lost by the customer is just installation time