

1. INTRODUCTION

1.1 Purpose:

- The purpose of this document is to present a detailed description of the PDF Extractor.
- It will explain the purpose and features of the Software, the interfaces of the Software, what the Software will do, the constraints under which it must operate and how the Software will react to external stimuli.
- This document is intended for both the stakeholders and the developers of the.

1.2 Scope of Project:

- This software ensures that we can extract text from even hand written data so searching can be made easy.
- The system has been developed using Python. So that all features are added in this Software. At the time of handling this system, it allows multi-user environment facility.
- The burden of manual work is reduced because of the computerization of the system.
- The system is user friendly and built in Window environment.
- Updating is very much user friendly and easy.
- It reduces data redundancy.

1.3 Major Issues of Existing System:

- The existing system lack proper GUI (Graphical User Interface) .
- Person having basic skills of computer cannot use the existing system.
- The system can only be executed by trained programmer.
- The system lacks the basic functionalities like extracting text from Images , books etc.
- Platform dependency.

1.3. References

- *IEEE Std 830-1998 IEEE Recommended Practice for Software Requirements Specifications. IEEE Computer Society, 1998.*

1.4 Overview of Document:

The next chapter, the Approach section, of this document gives .The third chapter, Requirements Specification section, of this document is written primarily for the developers and describes in technical terms the details of the functionality of the product.

Both sections of the document describe the same software product in its entirety, but are intended for different audiences and thus use different language.

2. APPROACH

Preliminary study is generally taken to collect the information (facts) from the existing software so that automated system can work in favor of user.

Facts are collected by:

2.1 Interviewing

2.2 Questionnaire

2.3 Observation

These techniques are not mutually exclusive indeed in practice more than one technique will be employed to establish the facts. During the course of an interview records may be inspected and a questionnaire may be completed.

2.1 INTERVIEWING:

The interview is a fact-to-fact exchange of information. It is a communication channel between the analyst and the organization. Interviewing is used to gain in support or understanding from the user for a new idea or method. The interview provides an excellent opportunity for the analyst to establish rapport with user personnel.

After the interview:

- Record the data at the earliest opportunity.
- Check the facts as soon as possible.
- Get agreements to the facts.

Advantages:

- Problem can be seen directly.
- Directly meet the organization.

2.2 QUESTIONNAIRE:

The questionnaire can be used best as a fact finding tool when the recipient is physically removed from the analyst and travel is prohibited for person, where there are many potential recipient and when the information is intended to verify similar information gathered from other sources.

Advantages:

Respondent is given time to assemble the required information, this saves his time as well as of the system analyst at the actual meeting. The answers are in some logical sequence hence easy to record the facts.

2.3 OBSERVATION:

- Observation can bring in missed facts, new ways to improve the existing procedures, duplicate work done inadvertently, etc.
- Observation can bring in what other fact finding methods cannot! But this task is delicate because people do not like to be observed when they work.
- It is not the quantity of time observed is important but the unorthodox angles of observation of the work content and methods are going to be rewarding.

Observation can look for:

- Operational inefficiencies.
- Alternate routes and procedures.
- Interruptions in the normal flow of work.
- The usage of files and documents.

Informal communication channels, etc.

- On site observation provides close view of the working of the real system.
- It is therefore easy to observe people, objects, documents and occurrences of events.

For this project, interview technique is used, interview is taken of the administrator (who is going to be only end user of the system).

Three sets of questions have been prepared:

1. For Administrators

- What is being done while extracting text from image?
- How it is being done?
- What are the problems that may arise?
- If a problem arises, how will it be solved?

Conclusion:

- They thoroughly read all the long documents just to search for a single keyword.
- Manually (Needs 2-3 Person)
- Data Redundancy
- It is very difficult process to find the Single Keyword from the register as it takes so much of time to find a single keyword . Register Keeper have to check it thoroughly, line by line.

3. SYSTEM PROFILE

- The proposed system is having many advantages over the existing system.

It requires less overhead and is very efficient.

- The new proposed PDF EXTRACTOR system will eradicate the issues that they are facing now and this software is created in such a manner that it can be used by anyone , even a person who have just a little knowledge of working of computer.
- It also gives us a path to wherever we want to save to file which lacks in the previous system as well as the option to select the desired file.

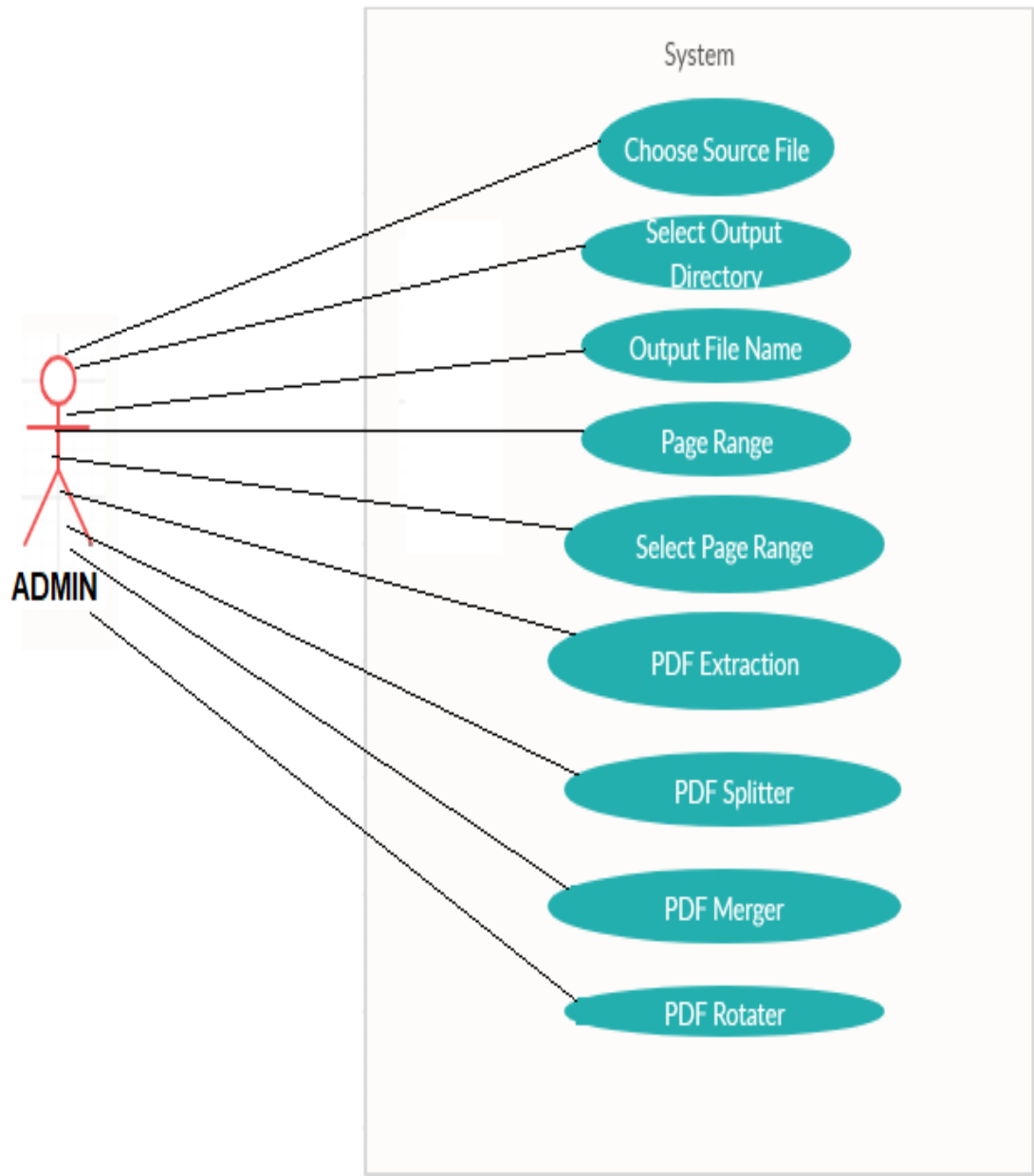
3.1 Functional Requirements should include:

- To be able to extract text from PDF files.
- To be able to extract text from image as well.
- To make searching a keyword easier .
- The system is user friendly and built in Window environment.

4. USE CASES:

1. Administrator Use cases:

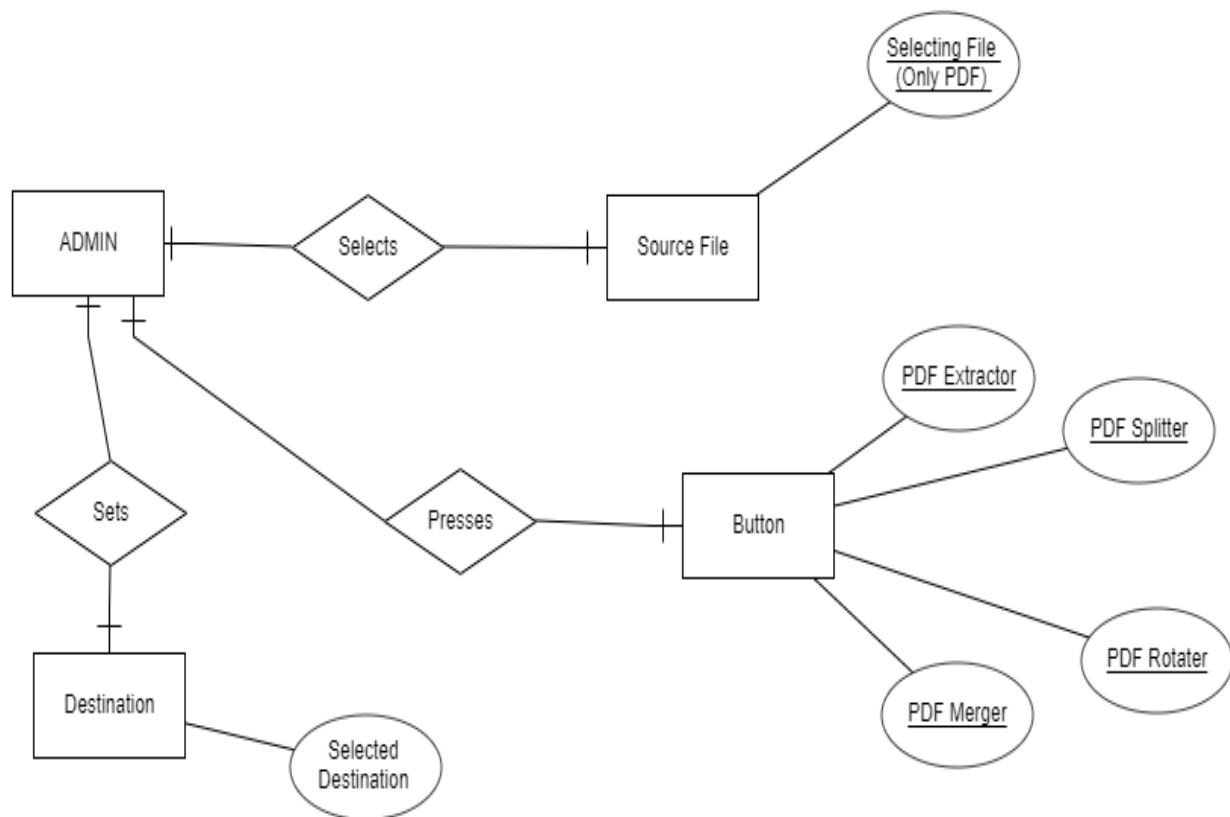
The Administrator has the following sets of use cases:



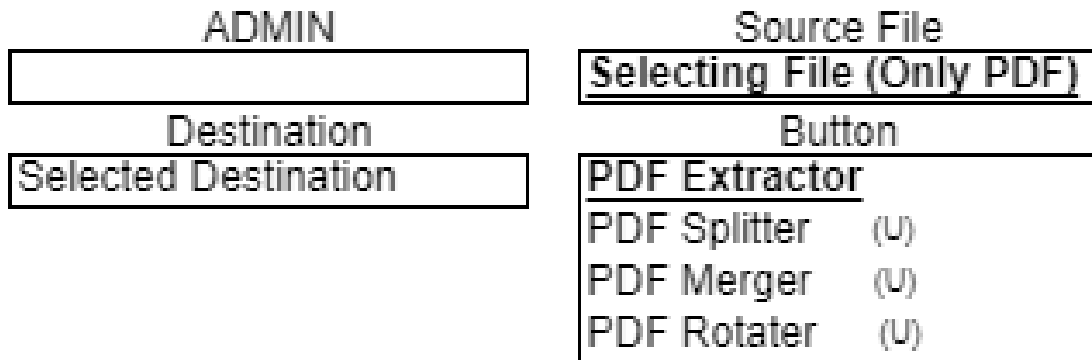
5. Module(s):

1. Target PDF File Explorer
2. Converted PDF File Explorer
3. Selecting the number of pages
4. Selecting between a Text PDF or Snapshot PDF
5. Drop box for more selections
6. Opening the path of converted file

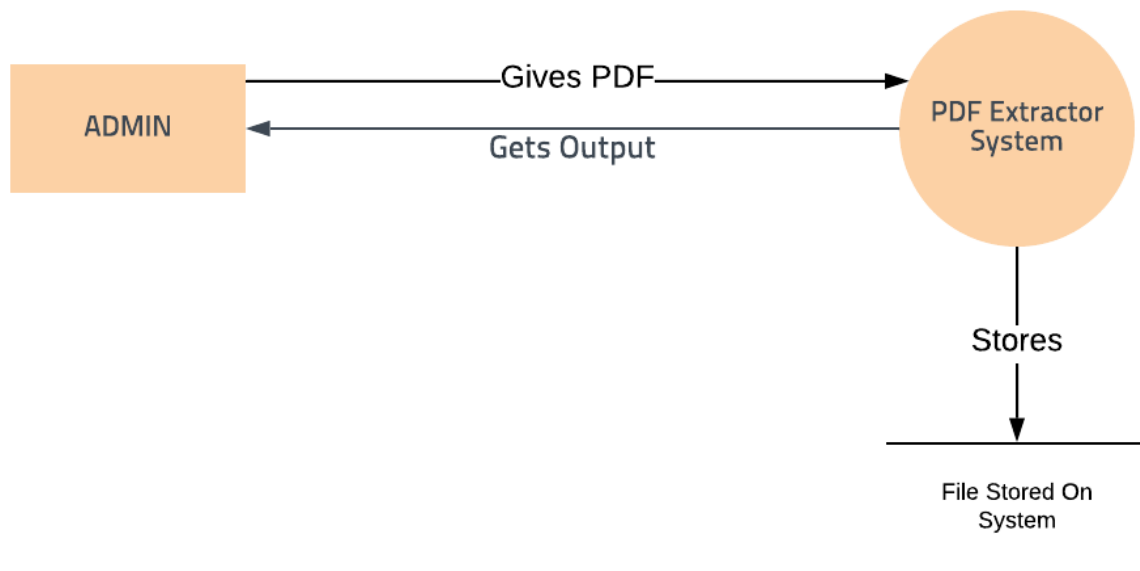
6. ER Diagram



6.1 Relational Schema

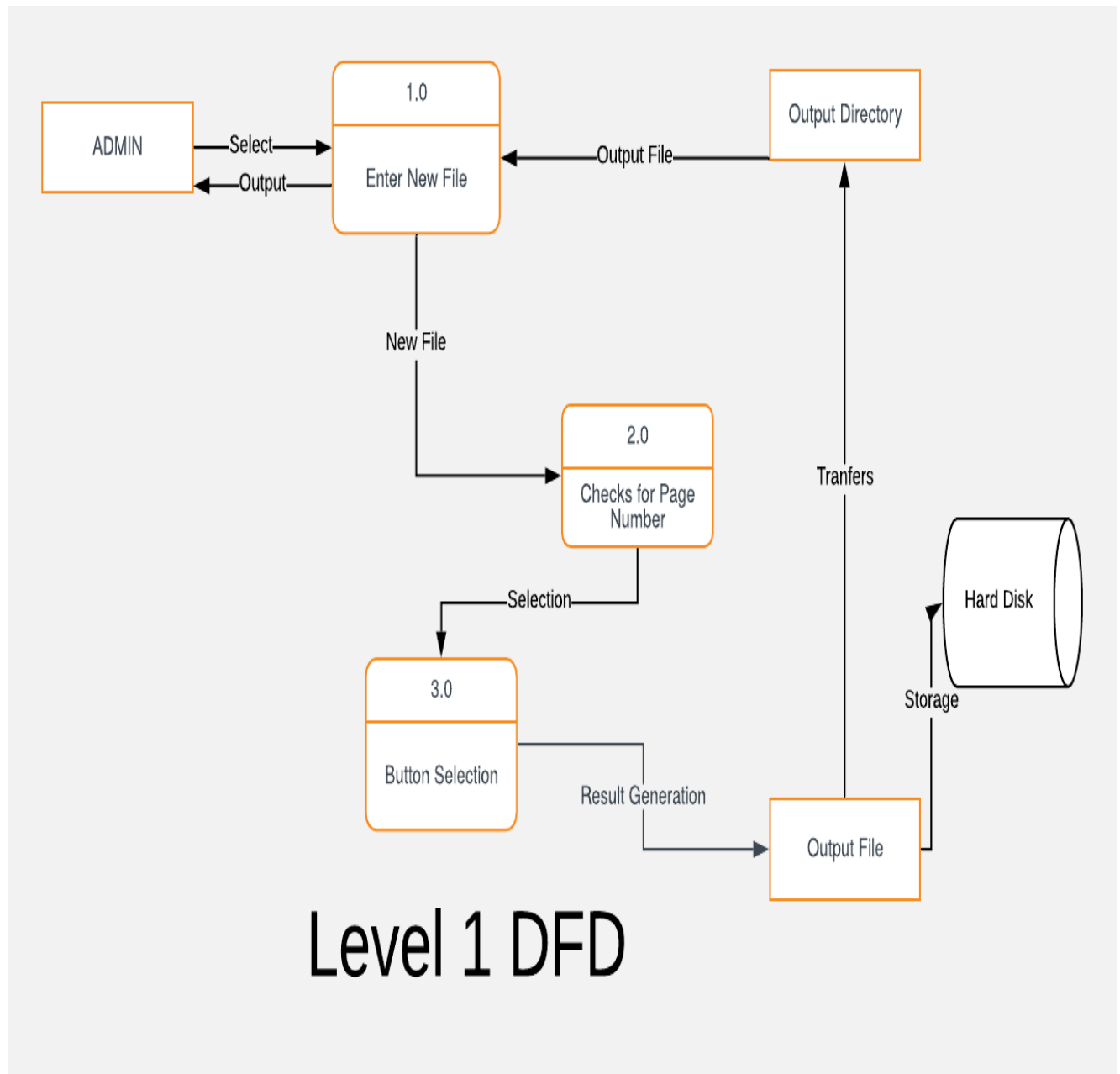


7. Data Flow Diagram:



Zero Level DFD

Level 1-DFD :



9. FEASIBILITY:

The document provides the feasibility of the project that is being designed and lists various areas that were considered very carefully during the feasibility study of this project such as Technical, Economic, Operational and Schedule feasibilities.

8.1. Technical Feasibility:

- The system must be evaluated from the technical point of view first. The assessment of this feasibility must be based on an outline design of the system requirement in the terms of input, output, programs and procedures. Having identified an outline system, the investigation must go on to suggest the type of equipment, required method developing the system, of running the system once it has been designed.
- Technical issues raised during the investigation are:
- Does the existing technology sufficient for the suggested one?
- Can the system expand if developed?

The project should be developed such that the necessary functions and performance are achieved within the constraints. The project is developed within latest technology. Through the technology may become obsolete after some period of time, due to the fact that never version of same software supports older versions, the system may still be used. So there are minimal constraints involved with this project.

8.2. Economic Feasibility:

- The developing system must be justified by cost and benefit. Criteria to ensure that effort is concentrated on project, which will give best, return at the earliest. One of the factors, which affect the development of a new system, is the cost it would require.
- The following are some of the important financial questions asked during preliminary investigation:
- The costs conduct a full system investigation.

- The cost of hardware and software.
- The benefits in the form of reduced costs or fewer costly errors.
- Since the system is developed as part of project work, there is no manual cost to spend for the proposed system. Also all the resources are already available, it gives an indication of the system is economically possible for development.

8.3. Behavioral Feasibility:

This includes the following questions:

- Is there sufficient support for the users?
- Will the proposed system cause harm?
- The project would be beneficial because it satisfies the objectives when developed and installed. All behavioral aspects are considered carefully and conclude that the project is behaviorally feasible.

Requirement analysis is the most important part to be considered before developing the projects as we must be aware of the objective that wish to achieve by making this project. This analysis finds out the problem faced by the user and clients and the possible solutions of those problems. We can use some questionnaires to finds out the problems faced by the clients. This System will be helping administrator for doing all the related jobs so the analysis part include problems faced by the Patient Monitoring administration and other facilities excepted by him from the new system. The manual handling of all Patient Monitoring functions may produce numerous types of problems.

8.4. Schedule Feasibility

Time evaluation is the most important consideration in the development of project. The time schedule required for the developed of this project is very important since more development time effect machine time, cost and cause delay in the development of other systems. A reliable Patient Monitoring application can be developed in the considerable amount of time.

9. DRAFT MODEL:

As this is a Small Project and Requirements are well understood. Waterfall model is simple and easy to understand and use.

Reasons for using waterfall model:

The Waterfall Model was first Process Model to be introduced. It is also referred to as a linear-sequential life cycle model. It is very simple to understand and use.

In a waterfall model, each phase must be completed fully before the next phase can begin. This type of Software development model is basically used for the project which is small and there are no uncertain requirements. At the end of each phase, a review takes place to determine if the project is on the right path and whether or not to continue or discard the project. In this model testing starts only after the development is complete. In waterfall model phases do not overlap.

When to use the waterfall model:

1. This model is used only when the requirements are very well known, clear and fixed.
2. Technology is understood.
3. There are no ambiguous requirements
4. Product definition is stable.
5. Ample resources with required expertise are available freely
6. The project is short.

Waterfall model is simple to implement and also the amount of resources required for it are minimal. It is easy to manage as each phase has specific deliverables and a review process.

In this model, output is generated after each stage, therefore it has high visibility. In this model phases are processed and completed one at a time. Phases do not overlap.

Different Phases of Waterfall Model are:

- **Requirement Gathering and analysis** – All possible requirements of the system to be developed are captured in this phase and documented in a requirement specification document.
- **System Design** – The requirement specifications from first phase are studied in this phase and the system design is prepared. This system design helps in specifying hardware and system requirements and helps in defining the overall system architecture.

- **Implementation** – With inputs from the system design, the system is first developed in small programs called units, which are integrated in the next phase. Each unit is developed and tested for its functionality, which is referred to as Unit Testing.
- **Integration and Testing** – All the units developed in the implementation phase are integrated into a system after testing of each unit. Post integration the entire system is tested for any faults and failures.
- **Deployment of system** – Once the functional and non-functional testing is done; the product is deployed in the customer environment or released into the market.
- **Maintenance** – There are some issues which come up in the client environment. To fix those issues, patches are released. Also to enhance the product some better versions are released. Maintenance is done to deliver these changes in the customer environment.

