Practical-1

Aim: Phases in Software Development Project, Overview, Need, Coverage of Topic

Select process model Introduction:

In Iterative model, iterative process starts with a simple implementation of a small set of the software requirements and iteratively enhances the evolving versions until the complete system is implemented and ready to be deployed.

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 is then reviewed in order to identify further requirements. This process is then repeated, producing a new version of the software at the end of each iteration of the model.

Iterative Model design:

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).

Following is the pictorial representation of Iterative and Incremental model:

Iterative and Incremental development is a combination of both iterative design or iterative method and incremental build model for development. "During software development, more than one iteration of the software development cycle may be in progress at the same time." and "This process may be described as an "evolutionary acquisition" or "incremental build" approach."

In incremental model the whole requirement is divided into various builds. During each iteration, the development module goes through the requirements, design, implementation and testing phases. Each subsequent release of the module adds function to the previous release.
The process continues till the complete system is ready as per the requirement.
The key to successful use of an iterative software development lifecycle is rigorous validation of requirements, and verification & testing of each version of the software against those requirements within each cycle of the model. As the software evolves through successive cycles, tests have to be repeated and extended to verify each version of the software.
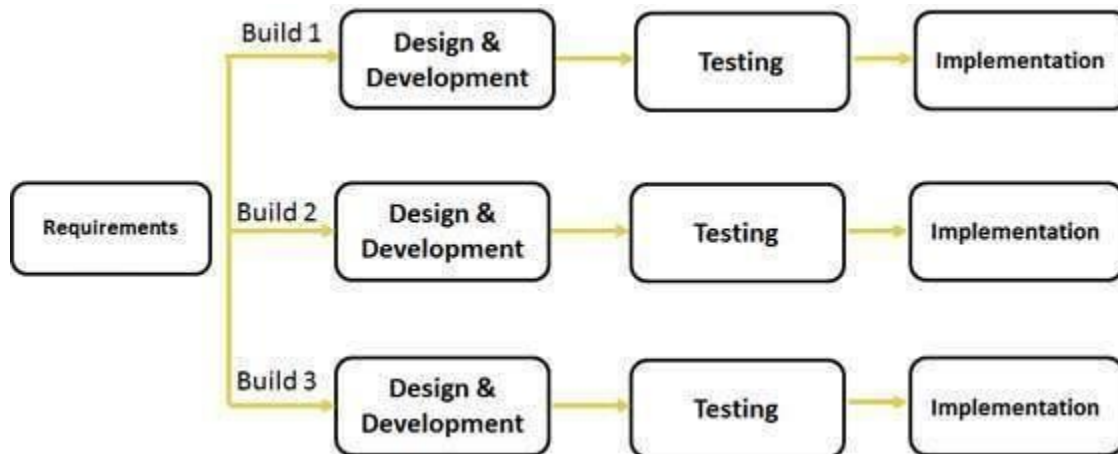
Iterative Model Application:

Like other SDLC models, Iterative and incremental development has some specific applications in the software industry. This model is most often used in the following scenarios:

- Requirements of the complete system are clearly defined and understood.

- Major requirements must be 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.

- Resources with needed skill set are not available and are planned to be used on contract basis for specific iterations.

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

Iterative Model Used in Our System:

• First we gathered information from the user.

• After requirement gathering, we started designing of our project, and created modules.

• Then we moved to create data flow of our project and various other things.

• Now we are creating our first module and from that module we are going to enhance our project and repeat the process.

Phases of SDLC:
• **Communication:**

    Mostly related to talk of customer and system analyst about project requirements, time-limit (period) and bargaining about cost etc.

• **Planning:**

    After getting requirements from customer, one make plan about project like how many days it would take, he/she analyze existing system find problem in it, and in doing so one should focus on customer requirements.

• **Modelling**

    modelling includes various types of diagrams like UML diagrams like Class, activity, sequence, state and E-R diagram etc. Designer decides which software development model will be good for project and choose it. This is a very important phase of SDLC. A prototype of a system designed by designer.

• **Construction:**

    Construction part includes coding of system based on model which prepared in modeling phase Programmer do coding any platform on our project would be easily constructed and tested. After coding Q.A. test system and suggest changes to programmer if needed.

- **Deployment:**

   After completing coding and testing one deliver system to the customer and take feedback From customer and if customer suggest any changes then one should concern that and add that feature. After few months if any problem detected then service will also be provided to customer. Based on feedback upgraded system will be provided.

Overview of the system:

   This system provides an effective way to learn web designing and developing languages with certification.

Need of the system:

 Pre Develop system has no Provide Security so Revenue was not generate. So Anyone download and misuse the website.

Coverage of topic:

   The modules can be implemented by the software product will be-
   1. Registration
      - ✓ Login
      - ✓ New user? Register
      - ✓ Forgot Password
      - ✓ Visitors home
   2. Upload Material
   3. Download Material
   4. Search Material
   5. Ratings And feedback

Practical-2

Aim: Preparing Software Requirements Specifications: Purpose, Scope, Feasibility Study, Functional and Non-Functional Requirements

Purpose of system:

This document serves for engineering students who want to achieve some good marks in examination. They refer to this website and download the material of particular subject. Here in this website, students & professors are connected and can easily upload & download the documents. This website is used for Diploma and Degree students.

Scope of the System

Various publishing books have different contains and it is difficult to refer the book according to syllabus.so generally students prepare notes from one or more reference books and they can upload it.

In the Engineering field some students make some Notes , that are ready for a good preparation. Those students who help to poor students for passing their exams for that they can upload their notes. Another thing that some good professors can upload materials for the students.
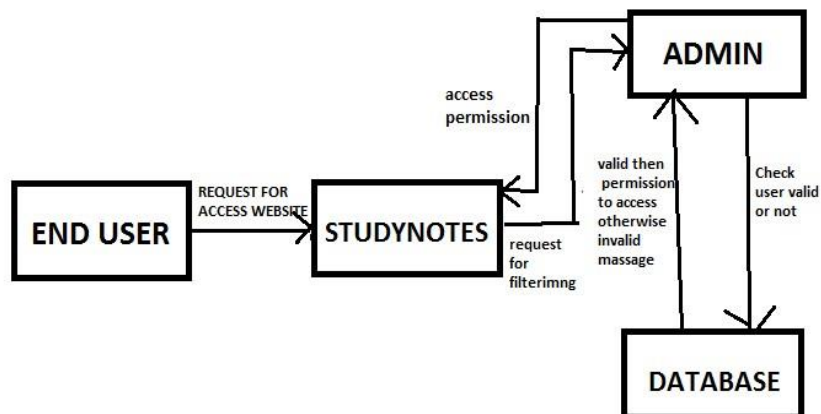
Feasibility study:

Preliminary investigation examines project feasibility, the likelihood the system will be useful to the organization. The main objective of the feasibility study is to test the Technical, Operational and Economical feasibility for adding new modules and debugging old running system. All systems are feasible if they are given unlimited resources and infinite time. There are aspects in the feasibility study portion of the preliminary investigation:

1. **Technical Feasibility :**

   In this project technically it is possible for administrator to contact the wide range of professors & students virtually (without being in physical contact with them). They also download, upload various Material. So it's Technically Feasible.

2. **Operational Feasibility:**

   It is feasible for our project to make use of functions like upload & download which will easily download with by Subject Name and by branch various functions are also possible e.g. user can give ratings from that downloaded it.

3. **Economic Feasibility:**

   A system can be developed technically and that will be used if installed must still be a good investment for the organization. In the economic feasibility, the development cost in creating the system is evaluated against the ultimate benefit derived from the new systems. Financial benefits must equal or exceed the costs. The system is economically feasible. It does not require any additional hardware or software.

Functional requirements:

 Description of Modules:

1. Secure registration and profile management facilities for end users
2. Easily Upload Material.

3. Adequate searching mechanisms for easy and quick access to particular topic in particular subjects.
4. Easily Download Material.

5. Give Ratings for Material and give feedback.

6. Regular updates to registered customers of the website about new arrivals and facilities.
7. Saving checkpoint that you have visited last.
8. Maintaining access database of regular customers of different needs.

- **Technical issues:**

The project will require <u>Java script</u> (client side scripting language), <u>PHP</u> (server side scripting language) as a front end & at the back end the database <u>SQL</u> (database storage) will be running for various function modules.

• Functions of various user of the system:

There are mainly 3 user of the product:

- **Students**
  - ✓ Login
  - ✓ Download Material
  - ✓ Upload Material
  - ✓ Give Ratings
  - ✓ Give feedback

- **Professors**
  - ✓ Login
  - ✓ Upload Material
  - ✓ View Material
  - ✓ Give Feedback

- **Admin**
  - ✓ Database Management
  - ✓ Solve the bugs of Product
  - ✓ Update The website
  - ✓ Remove bad Ratings Material
  - ✓ Notify The Users to New Material Arrivals

Non- Functional Requirements:

1. **Security:**
   - o It should be providing password to log on to the system.
   - o He/she should be able to see the record of Available Subject Notes.

2. **Reliability:**

   - o Application should be highly reliable and it should generate all the updated information in correct order.

3. **Availability:**

   - o Any information about customer should be quickly available from any computer to the authorized user. The previously visited customer's data must also be maintained and should be made available to the system authorized by simply entering his registration-Id.

4. **Maintainability:**

   - o The application should be maintainable in such a manner that if any new requirement occurs then it should be easily incorporated in an individual module.

5. **Portability:**

   - o The application should be portable on any system incorporating any hardware interface.

6. **Reusability:**

   - o The same system will be used for each new guide or end user.

Practical-3

Aim: Preparing Software Requirements Specifications: Interface Requirements, Performance Requirement, Design Constraint, Preliminary Schedule

Interface Requirements:

1. **GUI**

   GUI 1: The home page should shows the subjects to learn and details of admin and guide. GUI 2: The form to registers for advanced learning materials and asking queries GUI 3: This interface will get the information about the web guide and validates it. GUI 4: This interface does the validation of end users and provides a common messaging interface between end user and guide.

2. **Hardware Interface**

   Hardware Interface 1: The system should be run over the internet
   Hardware Interface 2: The customer must have broadband or Dial up or wireless internet data connection to access the system

3. **Software Interface**

   Software Interface 1: Students & Professors should be easily Registration.
   Software Interface 2: The details of Student & Professors should be transmitted to the Database server.
   Software Interface 3: The feedback system is connected to the database system.

Performance Requirements:

1. **Response Time**
2. **Administrator response**
3. **Throughput**
4. **Capacity**
5. **Resource utilization**

Design Constraints:

1. **System language used : PHP, oracle database**
2. **Development tools: Edit++**

Preliminary Schedule:

Time (In months or days)	Date (From-to)	Requirements gathering	Analysis	Design	Coding	Testing
0.5month	19/12/2016 To 02/01/2017	Yes	----------------	----------------	----------------	----------------
1month	03/01/2017 To 03/02/2017	-----------------	Yes	----------------	----------------	----------------
1month	04/02/2017 To 04/03/2017	-----------------	----------------	Yes	----------------	----------------
1month	05/03/2017 To 05/04/2017	-----------------	----------------	----------------	Yes	----------------
0.5month	06/04/2017 To 20/04/2017	-----------------	--------------	--------	-------------	Yes

schedule chart

—— schedule chart