Faculty of Engineering & Technology
Subject Name: Fundamental of software engineering
Subject Code: 203105255
B.Tech.: IT Year: 2021-22 Semester: 4(ITA1)

CERTIFICATE

This is to certify that Mr./Ms. ***……. Jaynesh..Modi……*** *with enrolment no.* ***……200303108024……*** *has successfully completed* ***his****/her laboratory experiments in the* ***Fundamental Of Software Engineering laboratory (203105255)*** *from the department of* ***……..Information Technology(4ITA1)……*** *during the academic year* ***……..2021-2022……..***

Date of Submission: ……. Staff In charge: …….

Head of Department: ……

INDEX

SR. NO.	TITLE	Number of hours	Page No.	Performance date	Assessment date	Marks out 0f 10	Sign
1	Project Definition and objective of the specified module and Perform Requirement Engineering Process.	2					
2	Identify Suitable Design and Implementation model from the different software engineering models.	2					
3	Prepare Software Requirement Specification (SRS) for the selected module.	2					
4	Develop Software project management planning (SPMP) for the specified module.	2					
5	Do Cost and Effort Estimation using different Software Cost Estimation models.	2					
6	Prepare System Analysis and System Design of identified Requirement specification using structure design as DFD with data dictionary and Structure chart for the specific module.	2					
7	Designing the module using Object Oriented approach including Use case Diagram with scenarios, Class Diagram and State Diagram, Collaboration Diagram, Sequence Diagram and Activity Diagram.	2					
8	Defining Coding Standards and walk through.	2					
9	Write the testcases for the identified module.	2					
10	Demonstrate the use of different Testing Tools with comparison.	2					
11	Define security and quality aspects of the identified module.	2					

PRACTICAL 1

AIM: - Project Definition and objective of the specified module and Perform Requirement Engineering Process.

HOME SERVICE PROVIDER: -

We understand your needs for small but difficult to solve issues like changing the broken tap of the kitchen or lamp in the study, making a fancy table from the waste wood, repairing the geyser, pest control, and lots more home services.

We know there are many 'How-to' guides on Google & YouTube, but professional assistance is needed to complete such tasks efficiently and correctly.

We are the best online service provider in India for any such issues at home, office, or any other place. You can book the services online, and our professional will be at your doorstep with the solution.

Take our online home services and spend the time with your beloved family as well as following your career/passion being stress-free.

INNOVATIVE FEATURES OF THE SYSTEM: -

1. Mobile response web design which allows the user to access the same site from any type of device like smartphone, tablets, smart tv, pc, laptop.
2. Provides the Services at Minimum charges & following the COVID-19 protocols.
3. It is the one stop solution for your everyday house chores whether it is related to electricity, earthing, plumbing, cleaning, repairing, or managing the stuff.
4. We want to become your all-time favorite and reliable directory with all important details and fair reviews when you are in a hurry or not in the mood to take any crap on your loving holiday.
5. We will help you to get services at your doorsteps, just like you get home deliveries of your favorite products.

SCOPE OF THE SYSTEM: -

1) We will only work in those areas where we are able to reach for examples cities and towns. We won't be able to come at Villages.
2) Would likely to provide all the services regarding…

- CLEANING
- PLUMBING
- ELECTRICIAN
- PAINTER
- CARPENTER
- GARDENER
- TAILOR
- DRIVER
- COOK

3) It will be developed in CSS, HTML, JAVASCRIPT & BOOTSTRAP.

MAIN MODULE OF THE SYSTEM: -

a) Admin Module.
b) User Module.
c) Search and view operation by site visitor (guest) for the available services.
d) Advertisement about the various types of services.
e) Online Payment, Cash Payment etc.
f) Facility of House requirement services.

REQUIREMENT OF SYSTEM: -

DEFINITION OF REQUIREMENT ANALYSIS:

Analysis of system requirements involves a clear understanding of the application to be
developed with the view of removing all ambiguities from user perception.

REQUIREMENT SPECIFICATION:

- **Non-functional requirement:**

The non-functional requirements include those that are implicit and improve the quality of the software.

- **Security:**

There is a facility for security of data. Authenticated users can access the application.

- **Easy to use:**

The system is easy to use and a good GUI.

- **Reliability:**

Reliability is assured by carrying out several tests for various test data as well as real live data and the output result matches the actual result. The system has been tested thoroughly which is described in the testing part. Hence, the system is reliable. The system supports generation of the printed reports.

- **Functional Requirement:**

1. USER LOGIN:

This feature used by the user to login into system. They are required to enter user id and password before they are allowed to enter the system. The user id and password will be verified and if invalid id is their user is allowed to not enter the system.

Requirements:

- User id is provided when they register
- The system must only allow user with valid id and password to enter the system
- The system performs authorization process which decides what user level can access to.
- The user must be able to logout after they finished using system.

2. ADMINISTRATOR:

Administrator of the system has all the rights and authorities to view as well as to modify and update the system whenever required for example he can add, remove, edit, area, valid zip code list, categories.

3. REGISTER A NEW USER:

This feature can be performed by all users to register new user to create account

Requirement:

- System must be able to verify information
- System must be able to delete information if information is wrong
- System must be able to search the database based on select search type
- System must be able to filter book based on keyword entered
- System must be able to show the filtered book in table view

4. SEARCH SERVICES:

This feature is found in home services. We can search services based on user choices by agent name.

Requirement: -

- System must be able to search the database based on select search type
- System must be able to filter book based on keyword entered
- System must be able to show the filtered book in table view

	VIEW	SEARCHING	MODIFY	ADD	WRITE
ADMINISTRATOR	Y	Y	Y	Y	N
GUSET	Y	Y	N	Y	Y

5. REGISTER AS EMPLOYEE:

This feature is used to register as new employee on the basis of their services skills.

Requirement: -

1. System must be able to verify information/document.
2. System must be able to not allow to same employee Id.
3. System must be able allow in registering at different services at one time.

6. SEARCH SERVICE SPECIFICATION: -

- In this feature user can be able to get services as per his requirement.
- **FEASIBILITY STUDY:**

Feasibility study is the study of the application to check whether the application made is feasible or not. It is very useful to check whether the application works as per requirement or not. It is undertaken to determine the possibility of developing completely new applications.

There are four feasibility studies that are considered.

- Technical feasibility
- Operational feasibility
- Implementation feasibility

7. SEARCH EMPLOYEE DETAILS: -

This feature is used to get details of employee to the customer only in an isolation for their security reason.

8. MAKE PAYMENT: -

This feature is used to get details of payment to the customer.

❖ **TECHNICAL FEASIBILITY:**

a) It determines that work for the project is done with the present equipment and existing software technology.
b) As suggested for technical feasibility services will be user friendly and has a better GUI.
c) Does the proposed equipment have the technical capacity to hold the data required to use the new system?
d) Are there technical guarantees of accuracy, reliability, ease of access and data security.

❖ OPERATIONAL FEASIBILITY:

It covers mainly two aspects. It determines how the proposed system will fill in the current operation and what will happen if the job retraining and reconstructing may be needed at the end of the implementation system. The operational feasibility checks whether the user who is going to be using the system is able to work with the software in which the system is coded! System is very user friendly. Level of security and any other access control constraints are high.

➢ IMPLEMENTATION FEASIBILITY:

- As we have mentioned that we are going to use PHP, HTML, CSS, MYSQL to develop this project, we found that these technologies are easy to learn and then use.
- There is no copyright issue we face in development.
- We can use any hardware configuration of pc/laptop to complete development of this project.
- Once project is developed, it can be hosted on any web server with any operating system.

➢ FEATURES OF THE SYSTEM:

- Also provides and interacting chat box where they are also given suggestion on the base of user's interest.
- Users can also read and write the reviews about the book.
- There is also a feature in which fine will be calculate automatically.
- Also tracks the user's record.

➢ SYSTEM REQUIREMENT STUDY AND ASSUMPTION:

- Hardware requirements

The online system that we have built requires some specific hardware configuration; I have mentioned the basic minimum hardware recommendations to run the system adequately. Any higher configuration hardware would only add to the performance of the system. the minimum hardware would only add to the performance of the system. The minimum hardware requirements to run the system properly are as follows.

- 20 GB hard disk
- 512 DDR RAM
- Network card/network connection
- 1.4 GHZ processor
- Software requirements
- Operating System: any modern operating system like window, Linux, MACOS
- Web Server: Internet Information Services (IIS) Server or Apache.
- Database: MySQL
- Server-side scripting language: PHP
- Operating System: any modern operating system like window, Linux, MACOS
- Web Browser: any modern browser like Firefox Mozilla / Internet Explorer.
- Technology used for development
- Front – End: HTML, CSS, JS, AJAX, BOOTSTRAP
- Back – End: PHP, MYSQL
- Tools: Notepad++/Sublime, Amp
- Operating System:
- Operating System: any modern operating system like window, Linux, MACOS.

PRACTICAL 2

AIM: Identify Suitable Design and Implementation model from the Different Software engineering models.

❖ LIST OF MODELS:

- Linear Sequential Waterfall model
- V-Model
- Prototype Model
- RAD Model.
- Evolutionary Software Process Model
  - Incremental Model
  - Spiral Model
- Iterative Model

1. Linear Sequential Waterfall Model:

The Waterfall Model was the 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 before the next phase can begin and there is no overlapping in the phases. The Waterfall model is the earliest SDLC approach that was used for software development. The waterfall Model illustrates the software development process in a linear sequential flow. In this waterfall model, the phases do not overlap.

2.V-Model:

The V-model is an SDLC model where execution of processes happens in a sequential manner in a V-shape. It is also known as **Verification and Validation model**. The V-Model is an extension of the waterfall model and is based on the association of a testing phase for each corresponding development stage. This means that for every single phase in the development cycle, there is a directly associated testing phase. This is a highly-disciplined model and the next phase starts only after completion of the previous phase.

3. Prototype Model:

The prototype model requires that before carrying out the development of actual software, a working prototype of the system should be built. A prototype is a toy implementation of the system. A prototype usually turns out to be a very crude

version of the actual system, possible exhibiting limited functional capabilities, low reliability, and inefficient performance as compared to actual software. In many instances, the client only has a general view of what is expected from the software product.

4.RAD Model:

The **RAD (Rapid Application Development)** model is based on prototyping and iterative development with no specific planning involved. The process of writing the software itself involves the planning required for developing the product. Rapid Application Development focuses on gathering customer requirements through workshops or focus groups, early testing of the prototypes by the customer using iterative concept, reuse of the existing prototypes (components), continuous integration and rapid delivery.

5. Incremental Model :

Incremental Model is a process of software development where requirements divided into multiple standalone modules of the software development cycle. In this model, each module goes through the requirements, design, implementation and testing phases. Every subsequent release of the module adds function to the previous release. The process continues until the complete system achieved.

6.Spiral Model :

The spiral model combines the idea of iterative development with the systematic, controlled aspects of the waterfall model. This Spiral model is a combination of iterative development process model and sequential linear development model i.e. the waterfall model with a very high emphasis on risk analysis. It allows incremental releases of the product or incremental refinement through each iteration around the spiral.

7. Iterative Model:

In the 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 to identify further requirements.

DIFFERENCE BETWEEN MODELS:

Properties of Model	Water-Fall Model	Incremental Model	Spiral Model	Rad Model
Planning in early stage	Yes	Yes	Yes	No
Returning to an earlier phase	No	Yes	Yes	Yes
Handle Large-Project	Not Appropriate	Not Appropriate	Appropriate	Not Appropriate
Detailed Documentation	Necessary	Yes but not much	Yes	Limited
Cost	Low	Low	Expensive	Low
Requirement Specifications	Beginning	Beginning	Beginning	Time boxed release
Flexibility to change	Difficult	Easy	Easy	Easy
User Involvement	Only at beginning	Intermediate	High	Only at the beginning
Maintenance	Least	Promotes Maintainability	Typical	Easily Maintained
Duration	Long	Very long	Long	Short
Risk Involvement	High	Low	Medium to high risk	Low
Framework Type	Linear	Linear + Iterative	Linear + Iterative	Linear
Testing	After completion of coding phase	After every iteration	At the end of the engineering phase	After completion of coding
Overlapping Phases	No	Yes (As parallel development is there)	No	Yes
Maintenance	Least Maintainable	Maintainable	Yes	Easily Maintainable
Re-usability	Least possible	To some extent	To some extent	Yes
Time-Frame	Very Long	Long	Long	Short
Working software availability	At the end of the life-cycle	At the end of every iteration	At the end of every iteration	At the end of the life cycle
Objective	High Assurance	Rapid Development	High Assurance	Rapid development
Team size	Large Team	Not Large Team	Large Team	Small Team
Customer control over administrator	Very Low	Yes	Yes	Yes

DIAGRAM FOR SELECTED MODEL AND ITDESCRIPTION:

❖ Iterative 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." This process may be described as an "evolutionary acquisition" or "incremental build" approach."
- In this 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 a 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.

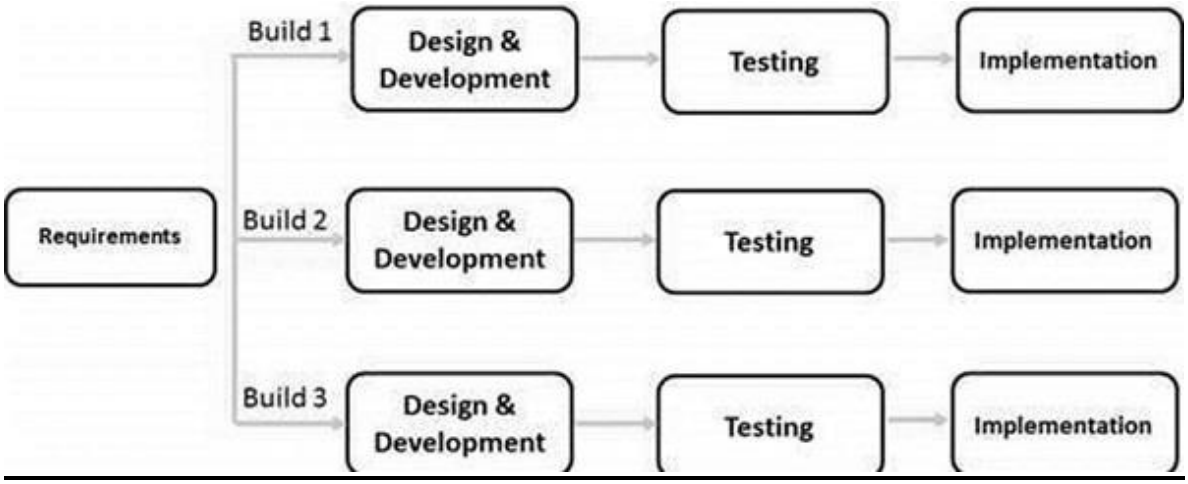
➢ When to use the Iterative Model?

- When requirements are defined clearly and easy to understand.
- When the software application is large.
- When there is a requirement of changes in future
- repeated and extended to verify each version of the software.

➢ Iterative Model - Application

- 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 sets 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.

➢ Diagram Of Iterative Model:



- **The various phases of Iterative model are as follows:**

1. Requirement gathering & analysis: In this phase, requirements are gathered from customers and check by an analyst whether requirements will fulfil or not. Analyst checks that need will achieve within budget or not. After all of this, the software team skips to the next phase.
2. Design: In the design phase, team design the software by the different diagrams like Data Flow diagram, activity diagram, class diagram, state transition diagram, etc.
3. Implementation: In the implementation, requirements are written in the coding language and transformed into computer programmers which are called Software.
4. Testing: After completing the coding phase, software testing starts using different test methods. There are many test methods, but the most common are white box, black box, and grey box test methods.
5. Deployment: After completing all the phases, software is deployed to its work environment.
6. Review: In this phase, after the product deployment, review phase is performed to check the behavior and validity of the developed product. And if there are any error found then the process starts again from the requirement gathering.
7. Maintenance: In the maintenance phase, after deployment of the software in the working environment there may be some bugs, some errors or new updates are required. Maintenance involves debugging and new addition options.

➢ Advantage (Pros) of Iterative Model:

1. Testing and debugging during smaller iteration is easy.
2. A Parallel development can plan.
3. It is easily acceptable to ever-changing needs of the project.
4. Risks are identified and resolved during iteration.
5. Limited time spent on documentation and extra time on designing.

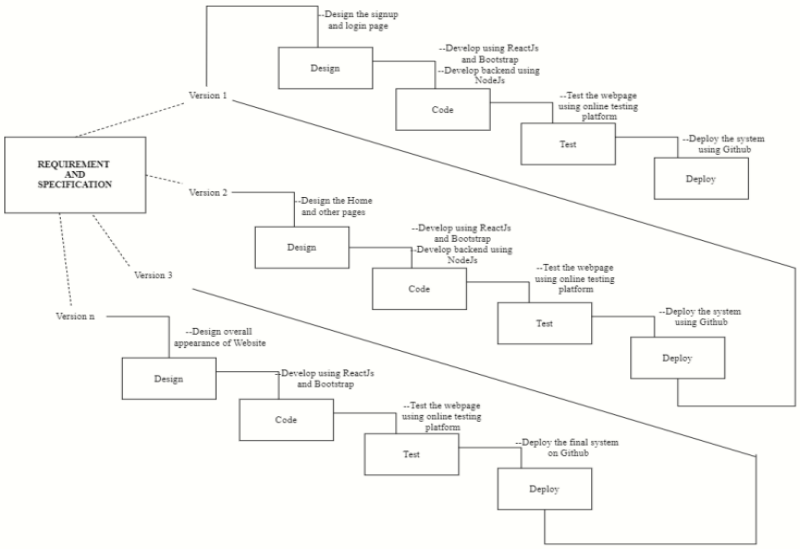
➢ Disadvantage (Cons) of Iterative Model:

1. It is not suitable for smaller projects.
2. More Resources may be required.
3. Design can be changed again and again because of imperfect requirements.
4. Requirement changes can cause over budget.
5. Project completion date not confirmed because of changing requirements.

➢ JUSTIFICATION: THAT WHY YOU CHOOSE THIS MODEL ONLY:

- Some working functionality can be developed and early in the software development life cycle (SDLC).
- It is easily adaptable to the ever-changing needs of the project as well as the client.
- It is best suited for agile organizations.
- It is more cost effective to change the scope or requirements in Iterative model.
- Parallel development can be planned.
- Testing and debugging during smaller iteration is easy.

➢ DIAGRAM OF MODEL ACCORDING TO YOUR DEFINATION:



Practical 3

AIM: - Study Software Requirement Engineering. Student should include SRS document for current semester project.

❖ SRS (Software Requirement Specification):

- The production of the requirements stage of the software development process is Software Requirements Specifications (SRS) (also called a requirements document).
- This report lays a foundation for software engineering activities and is constructing when entire requirements are elicited and analyzed. SRS is a formal report, which acts as a representation of software that enables the customers to review whether it (SRS) is according to their requirements.
- Also, it comprises user requirements for a system as well as detailed specifications of the system requirements.
- The SRS is a specification for a specific software product, program, or set of applications that perform particular functions in a specific environment. It serves several goals depending on who is writing it.
- First, the SRS could be written by the client of a system. Second, the SRS could be written by a developer of the system.
- The two methods create entirely various situations and establish different purposes for the document altogether. The first case, SRS, is used to define the needs and expectation of the users. The second case, SRS, is written for various purposes and serves as a contract document between customer and developer.

❖ SRS should address:

The basic issues that the SRS shall address are the following:

a) Functionality. What is the software supposed to do?

b) External interfaces. How does the software interact with people, the system ‘s hardware, other hardware, and other software?

c) Performance. What is the speed, availability, response time, recovery time of various software functions, etc.?

d) Attributes. What is the portability, correctness, maintainability, security, etc. considerations?

e) Design constraints imposed on an implementation. Are there any required

standards in effect, implementation language, policies for database integrity, resource limits, operating environment(s)

➢ Name of System:

✓ HOME SERVICE PROVIDER.

➢ Assumptions:

✓ Users will directly be able to see our website & if they book any service at that time it will ask for login requirements or sign-up page.

➢ Requirement 1: Register

First the user will have to register/sign up. There are two different types of users.

- The head of company: The manager have to provide details about the name of company, address, phone number, email id, customers review about their work.
- Regular person: The user has to provide details about his/her name of address, phone number, email id.

➢ Req1.1: Sign up

- Input: Detail about the user as mentioned in the description.
- Output: Confirmation of registration status and a membership number and password/OTP will be generated and mailed to the user.
- Processing: All details will be checked and if any error is found then an error message is displayed else a membership number and password will be generated.

➢ Req 1.2: Login

- Input: Enter the membership number and password/OTP which is provided by company.
- Output: User will be able to use the features of software.

➢ Requirement 2: How User will be able to get services.

➢ Req 2.1: Required services details

- Description: List of different types of services available at that time will be generated.

➢ Req 2.2: Search

- Input: Enter the type of service which you want.
- Output: List of all related services with the company name and ratings will be generated.

➢ **Requirement 3: Manage workload by admin**

➢ **Req 3.1 Update details of the services: -**

➢ **Req 3.1.1 Add Company**

- Input: Enter the details of the services which you want such as name of services, name of company, opening date, quality.
- Output: Confirmation.

➢ **Req 3.1.2 Remove Company**

- Input: Enter the name of the company and quality of company.
- Output: Update the list of the services available by company

➢ **Requirement 4: Payment**

➢ **Req 4.1: Payment Gateway:**

- Input: Card Details, Paytm Wallet.
- Output: User receives the message of debited amount and booking confirmation.

➢ **Requirement 5: Help me...?**

- Input: The customer will ask the admin, or company for available services
- Output: Admin, users will find the available company and provide the link to the customer.

➢ **Requirement 6: Security**

- Passwords and value information is saved in database after hashing and AES encrypted, Admin has a separate login where he can add and modify user data if needed.

➢ **Requirement 7: Easy to use**

- User Friendly.

➢ **Requirement 8: Reliability**

- FAQ's

➢ **Requirement 10: Portability**

- Website can be accessed on any device having any operating system.

Practical 4

AIM:- Develop Software project management planning (SPMP) for the specified module.

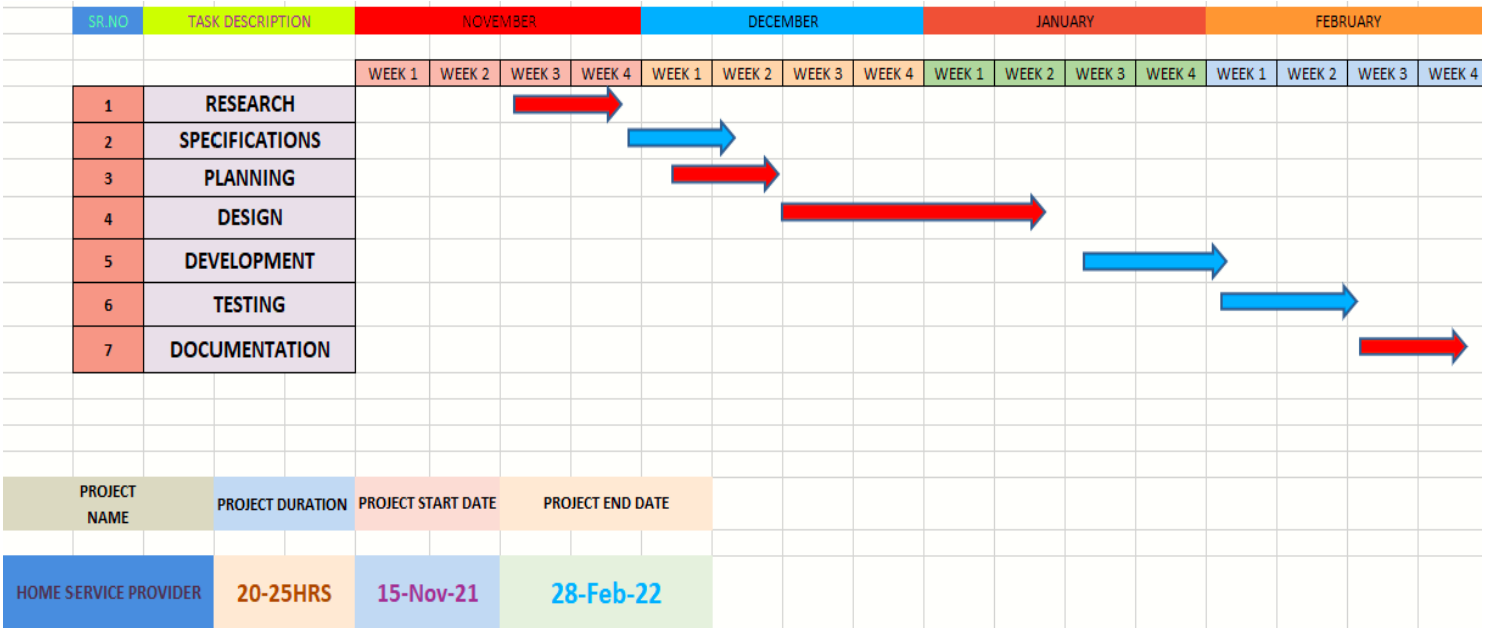
Theory:

- Once project designing is complete, project managers document their plans during a software package Project Management set up (SPMP) document. The SPMP document ought to discuss an inventory of various things that are mentioned below.
- This list will be used as a doable organization of the SPMP document. Organization of the software package Project Management set up (SPMP) document.
- **Introduction:-**
  - Objectives
  - Major Functions
  - Performance Issues
  - Management and Technical Constraints
- **Project Estimates:**
  - Historical Data Used
  - Estimation Techniques Used
  - Effort, Resource, Cost, and Project Duration Estimates
- **Schedule:**
  - Work Breakdown Structure
  - Task Network Representation
  - Gantt Chart Representation
  - PERT Chart Representation
- **Project Resources:**
  - People
  - Hardware and Software
  - Special Resources
- **Staff Organization:**
  - Team Structure
  - Management Reporting

- **Risk Management Plan:**
  - Risk Analysis
  - Risk Identification
  - Risk Estimation
  - Risk Abatement Procedures
- **Project Tracking and Control Plan**
- **Miscellaneous Plans:**
  - Process Tailoring
  - Quality Assurance Plan
  - Configuration Management Plan
  - Validation and Verification
  - System Testing Plan
  - Delivery, Installation, and Maintenance Plan

What Is a Gantt Chart?

A Gantt chart is a bar chart that provides a visual view of project tasks scheduled over time. A Gantt chart is used for project planning: it's a useful way of showing what work is scheduled to be done on specific days. It helps project managers and team members view the start dates, end dates and milestones of a project schedule in one simple stacked bar chart.



Practical 5

AIM:- Do Cost and Effort Estimation using different Software Cost Estimation models.

THEORY:

5.1 Cocomo model:

Cocomo (Constructive Cost Model) is a regression model based on LOC, i.e. number of Lines of Code. It is a procedural cost estimate model for software projects and often used as a process of reliably predicting the various parameters associated with making a project such as size, effort, cost, time and quality.

❖ **The Modes**

- **Organic :**
  2-50 KLOC, small, stable, little innovation
- **Semi-detached :**
  50-300 KLOC, medium-sized, average abilities, medium time-constraints
- **Embedded :**
  > 300 KLOC, large project team, complex, innovative, severe constraints

The constants cocomo model:

Mode	a	b
Organic	2.4	1.05
Semi-detached	3.0	1.12
Embedded	3.6	1.20

5.1.1 These are types of COCOMO model:

1. **Basic COCOMO Model**
2. **Intermediate COCOMO Model**
3. **Detailed COCOMO Model**

1.Basic cocomo model.

Equation of basic cocomo:

$$E = a(KLOC)^b$$

Where,

- E = the Effort in staff months.
- a & b = coefficients to be determined.
- KLOC = thousands of lines of code.

Effort:

KLOC = 3, a = 2.4, b = 1.05

$E=2.4(5)^{1.05}$

E=13.005 staff-month

Development time(Project Duration):

$$\text{TDEV=c}(E)^d$$

Where,

- TDEV is time for development
- c and d are constants to be determined
- E is the effort

Constants for TDEV,

Mode	c	d
Organic	2.5	0.38
Semi-detached	2.5	0.35
Embedded	2.5	0.32

$$\text{TDEV=c}(E)^d$$

$TDEV=2.5(13.005)^{0.38}$

TDEV=6.626 month

Average Staff Size:

$$\text{Staff size=} \frac{E}{TDEV}$$

$$\text{Staff size=} \frac{13.005}{6.626} \frac{staff-\cancel{month}}{\cancel{month}}$$

Staff size=1.962 staff

Productivity:

$$\textbf{Productivity} = \frac{Size}{E}$$

Where,

- LOC = line of code
- E = effort

$$\text{Productivity} = \frac{5000}{13.005} \quad \frac{LOC}{Staff-month}$$

3) Detailed cocomo model

Detailed COCOMO incorporates all characteristics of the intermediate version with an assessment of the cost driver's impact on each step of the software engineering process. The detailed model uses different effort multipliers for each cost driver attribute. In detailed cocomo, the whole software is divided into different modules and then we apply COCOMO in different modules to estimate effort and then sum the effort.

The Six phases of detailed COCOMO are:

• Planning and requirements
• System design
• Detailed design
• Module code and test
• Integration and test
• Cost Constructive model

The effort is calculated as a function of program size and a set of cost drivers are given according to each phase of the software lifecycle.