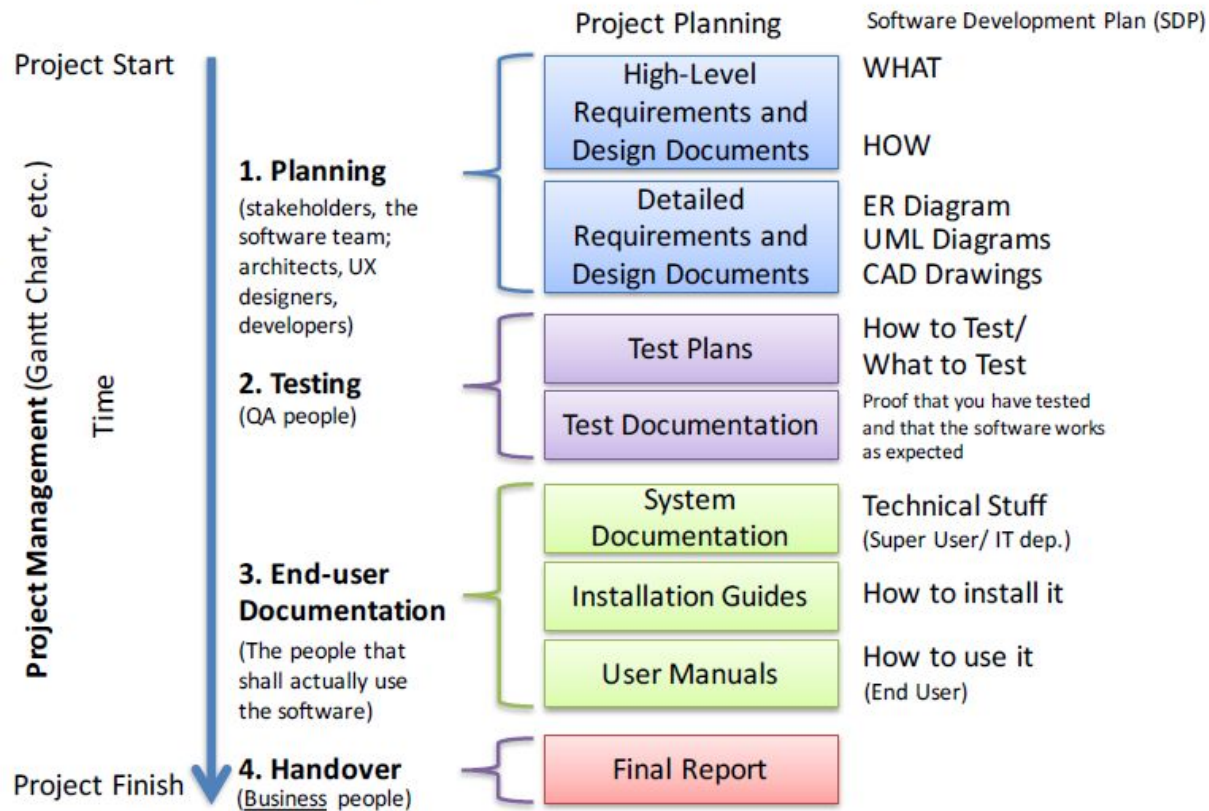


# **System Administration & Training**

---

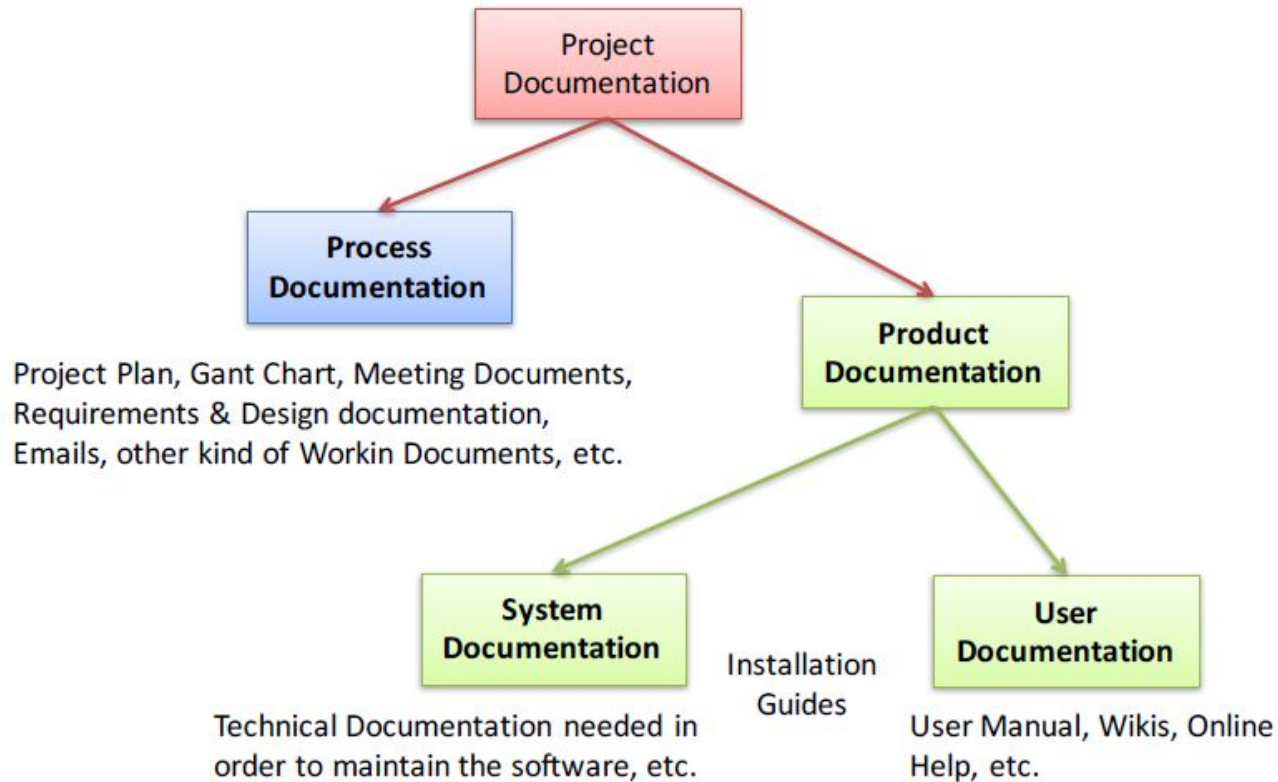
VIBHUTI PATEL

# Project Documentation



# S/w Project Documentation Categories

---



# Software Process Documentation

---

1. Software Development Plan (SDP)
2. Software Requirements Specifications (SRS)
3. Software Design Documents (SDD)
4. Software Test Documents (STD)

# Software Requirements & Design

---

## Requirements (WHAT):

- **WHAT** the system should do
- Describe what the system should do with Words and Figures,etc.

## SRS – Software Requirements Specification

---

## Software Design (HOW):

- **HOW** it should do it
- Examples: GUI Design, UML, ER diagram, CAD, etc.

## SDD – Software Design Document.

Many don't separate SRS and SDD documents, but include everything in a Requirements document.

In practice, requirements and design are inseparable.

# Software Requirements Document

---

The software requirements document is the official statement of what is required of the system developers.

Should include both a definition of user requirements and a specification of the system requirements.

It is **NOT** a design document. As far as possible, it should set of WHAT the system should do rather than HOW it should do it.

# SRS Document Structure

## Introduction

- Purpose, Definitions, System overview
- Scope of Work, References

## Overall description

- Product perspective: System Interfaces, User Interfaces, Hardware interfaces, Software interfaces, Communication Interfaces, Memory Constraints, Operations, Site Adaptation Requirements
- Product functions and User characteristics
- Constraints, assumptions and dependencies

## Specific requirements

- External interface requirements
- Functional requirements
- Performance requirements
- Design constraints: Standards Compliance
- Logical database requirement
- Software System attributes: Reliability, Availability, Security, Maintainability, Portability
- Other requirements



# Software Documentation

---

## System/Technical Documentation

- Class Diagrams
- State Diagrams
- Sequence Diagrams
- Code Comments

## User Documentation

- User Manual
- Installation Guide
- Wiki
- Online Documentation and Help

## The set of documents that you have to produce for any system depends

---

- on the contract with the client for the system (the customer)
- the type of system being developed
- its expected lifetime
- the company culture
- the size of the company developing the system
- the development schedule

# Software Project Documentation

---

Documentation produced during a software Project can be divided into 2 Categories:

## **Process Documentation**

These documents record the process of development and maintenance, e.g., Plans, Schedules (e.g., Gantt Charts), etc.

## **Product Documentation**

These documents describe the product that is being developed. Can be divided into 2 sub categories:

---

## 1) **System Documentation**

Used by engineers developing and maintaining the system

## 2) **User Documentation**

Used by the people that is using the system

# Process Documentation

---

## Purpose:

Process Documentation is produced so that the development of the system can be managed.

It is an essential component of plan-driven approaches (e.g., Waterfall).

# Categories:

---

## **1. Plans, estimates and schedules**

These are documents produced by managers which are used to predict and to control the software process.

## **2. Reports**

These are documents which report how resources were used during the process of development.

## **3. Standards**

These are documents which set out how the process is to be implemented. These may be developed from organizational, national or international standards.

---

## 4. Working papers

These are often the principal technical communication documents in a project.

They record the ideas and thoughts of the engineers working on the project, describe implementation strategies and set out problems which have been identified.

## 5. E-mail messages, wikis, etc

These record the details of everyday communications between managers and development engineers.

# Software Development Plan (SDP)

---

An SDP normally include the following sections:

**1. Introduction:** This briefly describes the objectives of the project and set out the constraints (e.g., budget, time, etc.) that affects the management of the project

**2. Project Organization (Team Description)** This section describes how the development team is organized, the people involved and their roles in the team. Software Process Model Description (Waterfall, ...), etc.

**3. Risk Analysis**

**4. Hardware and Software Resource Requirements**



---

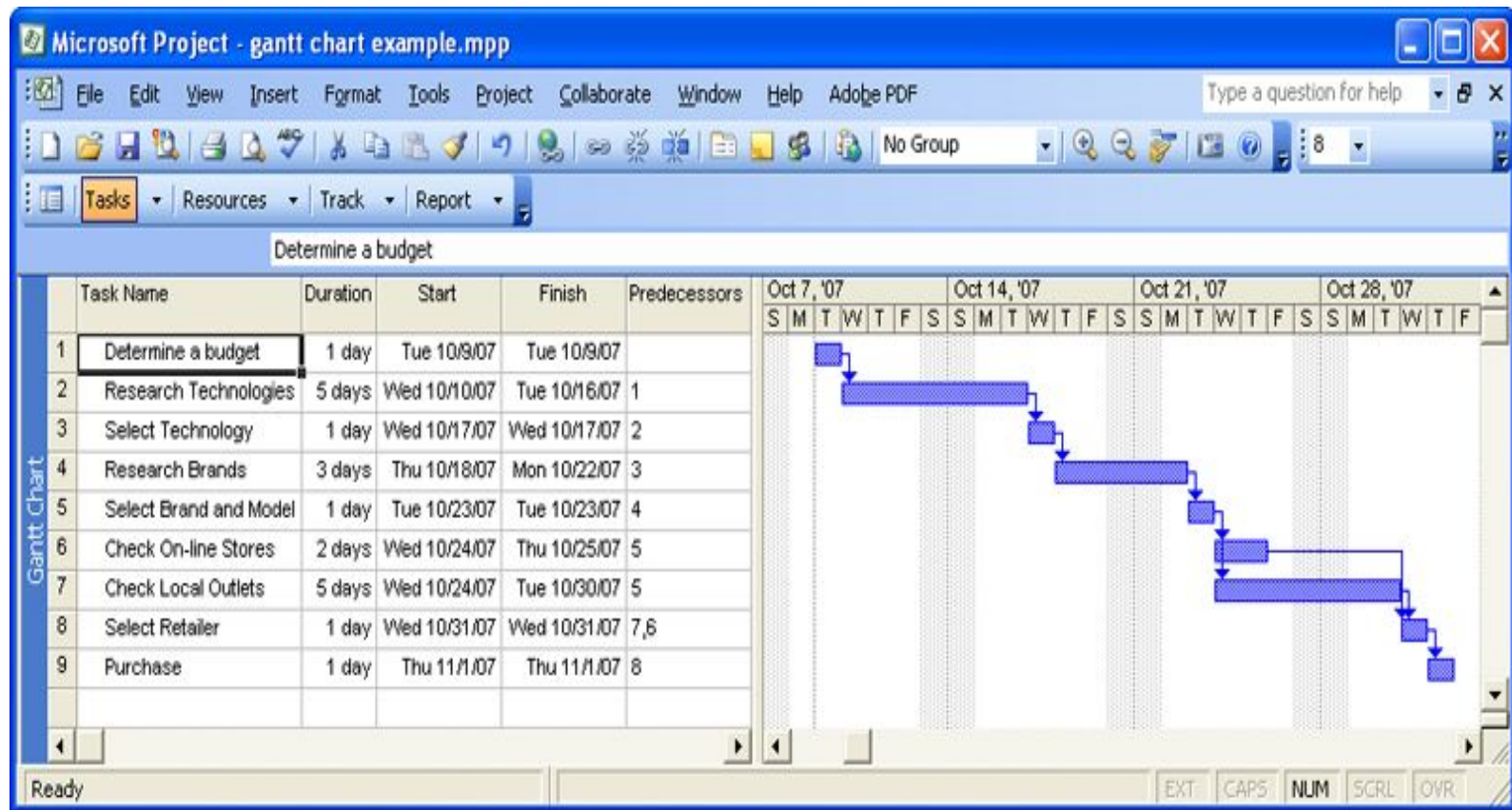
**5. Work Breakdown (WBS: Work Breakdown Structure):** Break down the project in into activities and identifies milestones.

**6. Project Schedule:** Shows dependencies between activities, the estimated time required to reach each milestone, allocation of people to activities. (5) and (6) is typically done in a Gantt Chart.

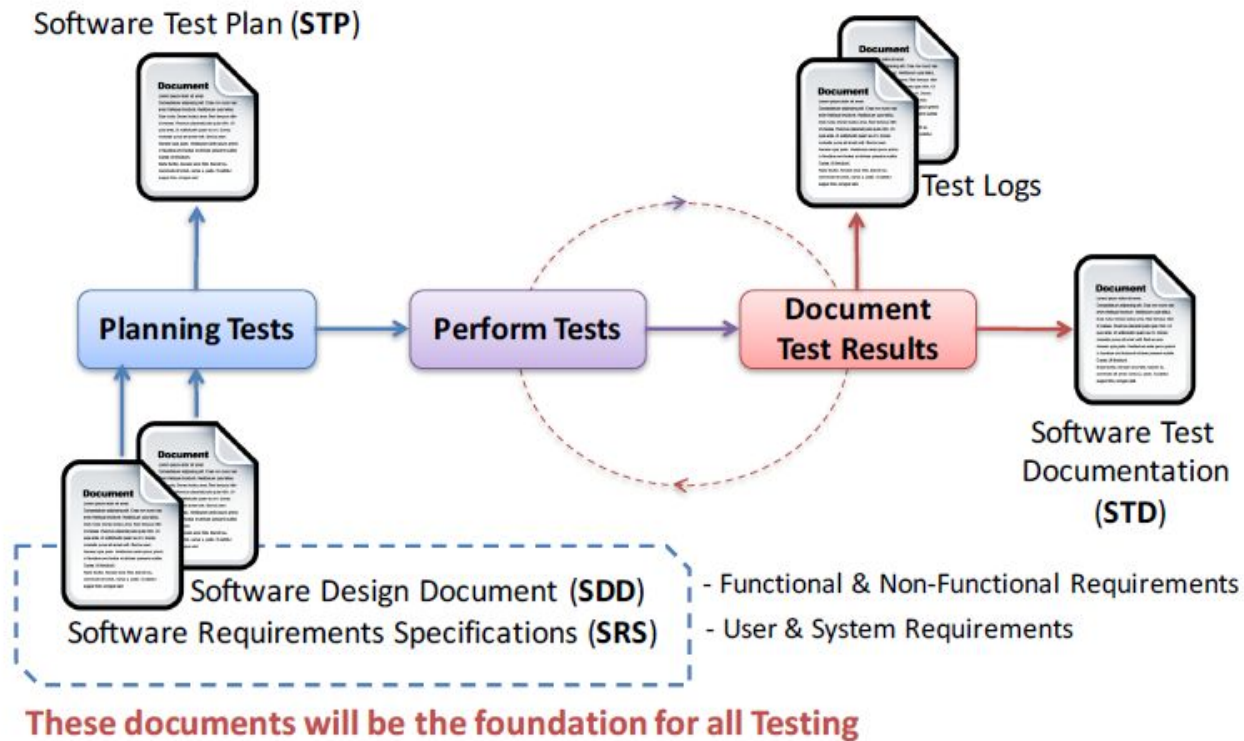
**7. Monitoring and Reporting Mechanisms:** Definition of the Management Report that should be produced, when this should be produced, etc.

**8. Tools** that you are using

# Gantt Chart



# Test Documentation



# Product Documentation

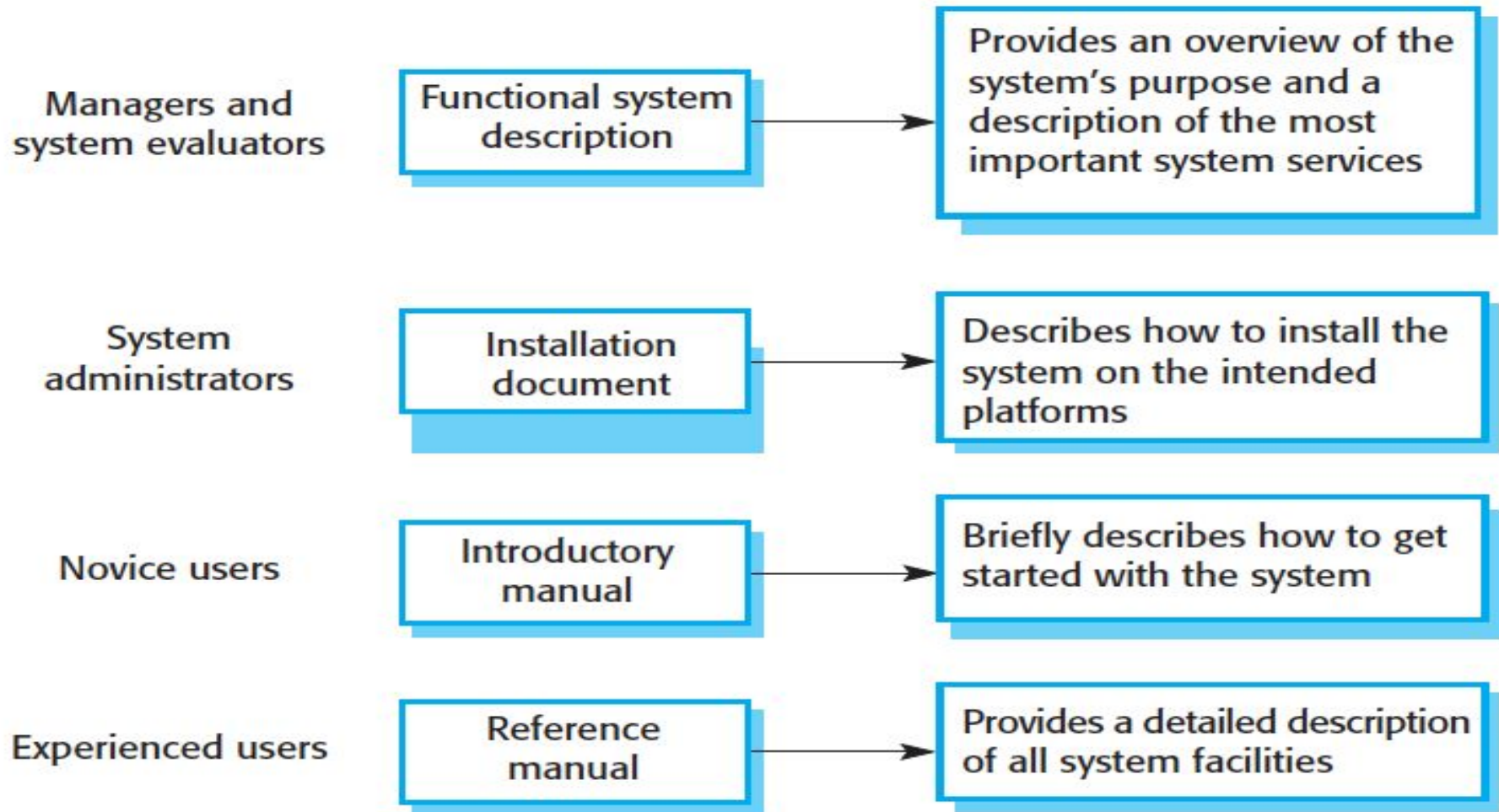
---

## Purpose:

- Describing the delivered software product.
- Unlike most process documentation, it has a relatively long life.
- It must Evolve in step with the product that it describes.
- Product documentation includes
  - **User documentation**, which tells users how to use the software product,
  - **System Documentation**, which is principally intended fo maintenance engineers.

# Product Documentation Types & Readers

---



# User Documentation Readers

---

Users of a system are not all the same.

- The producer of documentation must structure it to cater for different user tasks and different levels of expertise and experience.
- It is particularly important to distinguish between end-users and system administrators:

---

## **1. End-users**

Use the software to assist with some task.

This may be flying an aircraft, managing insurance policies, writing a book, etc. They want to know how the software can help them.

They are not interested in computer or administration details.

## **2. System administrators**

Are responsible for managing the software used by end-users.

This may involve acting as an operator if the system is a large mainframe system, as a network manager if the system involves a network of workstations or as a technical guru who fixes end-users software problems and who liaises between users and the software supplier.

---

**Thank You...**