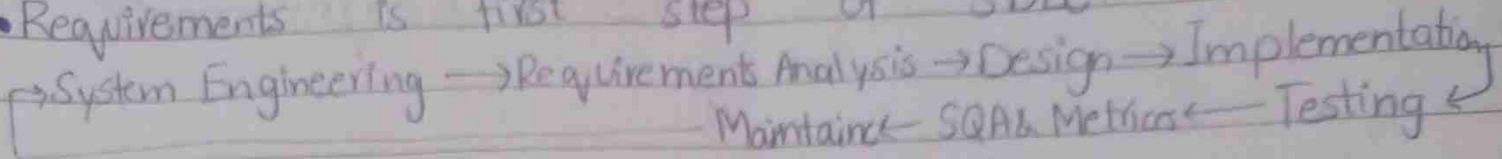


SRE

Date: _____

Requirements is first step of SDLC



Hope ^{L.P} فارہیں سے ملے لیں **Wikipedia**

Wish ^{unaffordable} فارحہ سے ملے لیں A single documented need of what a particular product or service should be

Desire ^{can't be avoided/foremost} فارحہ سے ملے لیں prioritize, afford or perform

Need ^{can't be avoided/foremost} prioritize, afford or perform

Require ^{can't be avoided/foremost} prioritize, afford or perform

⇒ Require is more strong than need

• While designing in condition or capability needed by a user to solve problem or achieve objective

• Functionality of Software

FS document (function specific document)

It is Output of R.E process

Foremost req. must be released first

Features are not technically/financially feasible can be ignored or released later

FS is used in testing

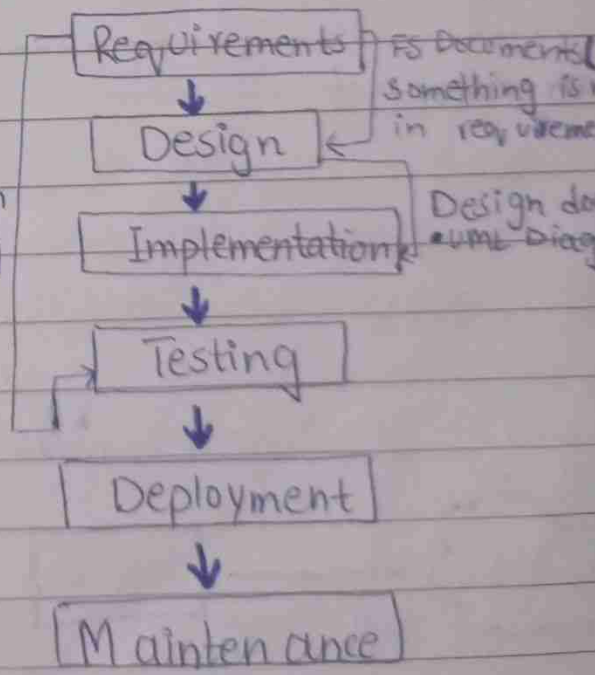
Requirements thing that is needed or wanted (optional)

Need Want

Compulsory Optional

can't be ignored can survive without want

Teacher need I need marker coffee



FS documents will also be missed in next steps of development. It creates huge loss in terms of cost, in terms of times.

① Requirement must be documented.

② Something in mind is not requirement

SRS

FS

more detail than FS

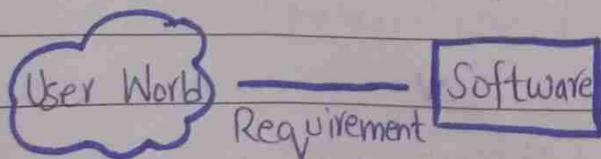
Functional specification

CMMI (capability maturity model integration)

• used for certification of software houses.

Requirement

Need, constraints, interface of any stakeholders that must be fulfilled by proposed software product during its development.



Role of Requirements

① Agreement (between system developers, customers, end-user)
• legal contract

② For software design
• Detect free • Technically feasible

③ For verification & validation

④ For system evolution
→ System evolution = change(old system, new system)
change(old req., new req.)

Source of Requirements

① Stakeholders (Client)

② Documents (process that organization use)

③ Existing System (flaws of old is req. of new system)

④ Application Domain (U study the domain of that application which U form).

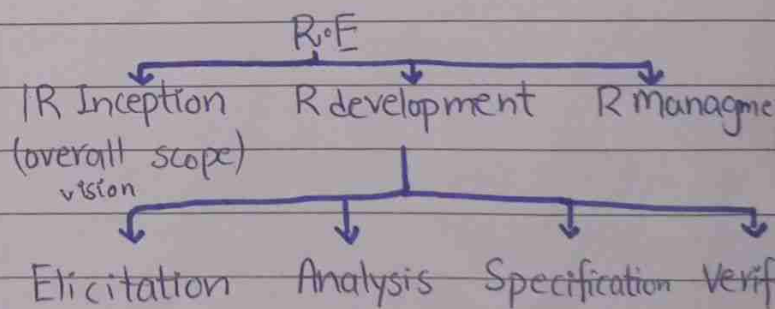
Criteria for good Require

① Necessary (prioritize need but if system works without it, so it's not necessary)

Date: _____

- ② Feasible :- (Budget, Time, Technical/algorithm fulfill 99%, not possible) fulfill customer specification
- ③ Correct :- (Besides technical, legally possible) R.E subset of system engineering with developing, discovering, tracing, analyzing, qualifying, communicating and managing requirements
- ④ Concise :- (Simple, unambiguous)
- ⑤ Unambiguous :- (Interpreted in 1-way)
- ⑥ Complete :- (Whole idea, statement) that define system at successive levels of abstraction.
- ⑦ Consistent :- (unflict not with other)
- ⑧ Verifiable :- (proved) ⇒ It is about determining
- ⑨ Design independent :- (not pose specific implementation) • problems with status (as, is)
- ⑩ Non-redundant :- (not duplicate requirement) • objectives to achieve
- ⑪ Traceable :- (tracked, traced throughout system) • changes to bring about for better future.
- ⑫ Allocated :- (requirement is assigned to component of design system)

Activities



Why?

Eng (process of converting specifications of customer into such artifacts that r used by artisans to produce products p that

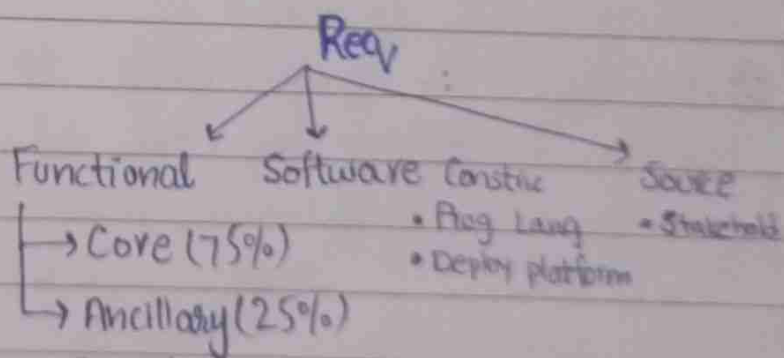
- 70% in specification phase
- 30% in technical solution

Date: _____

	Defects	Efforts	Classification
Requirement	56%	82%	Functionality consideration (that fulfill set of selected business processes and deliver results to end-users.
Code	7%	1%	
Design	27%	13%	
Other	10%	4%	

Stakeholder individual, group of people, organization or other entity has direct or indirect interest in system.

Errors Propagation in Lifecycle from different sources. It based on agencies that provided requirement.



Satisfactory (u make standards, i.e. u make software for deaf patient)

Safety: physical harmless for medical softwares

Security: prevent from external attacks

It is better to detect errors at requirement level bcz it takes less time

Date: _____

Software Footprint (software which is gone for deployment, having size on Harddisk)

Memory Constraint: At time of software execution, uska size kya he on RAM. e.g: minimum itni memory ho gy software execute kre ga.

Data Integrity Protection: It checks up data which user entered in system must be logically correct. ie for room observation.

Response-time functionality: The activity which is appear after a time. This time is called response-time. ie For convert image into text which take 2 min this 2 min is called response-time.

Fault-Tolerance: Software must not be crash but give user an alert msg. It provides an alternative path.

Reliability: Able to save data ie after PC power off, data must be safe.

Inverse requirement: When customer says things must not be part of system (excluded req.)

ie not use red color.

Design & Implementation

constraints: These are specified by technically customer in software construction.

Feel Good functionality: interface should be easy.

Esteem functionality: For user, it should be previlaged.

On-upmanship functionality: Your software have competitive + points againts ur competitive edge.

Usability: Its not vinterface when u reached one point for user reached easily on that point.

niceday

Date: _____

Project is limited time

Based on CMM

Productivity Overall scope, time

Level 1:

limit as such not present
and then maintain.

⇒ Software houses work

one of them (prod, proj)

Level 2:

Requirement Process

Process (series of action to achieve goal)

• In software, requirements r input project is output.

Why Process?

• Quality of Process → Quality of prod

• Garbage in Garbage out

• So get right requirements.

Level 3: standard boiler plate and defined process (set

of activities are documented, validate and testing on every step).

Requirements Life Cycle

Stakeholder req

Testing is with respect to req

Acceptance Testing

① Requirement Elicitation

② " Negotiation

③ Specification

④ Validation

System

System

Subsystem

Integration

① Req discovered through consult with stakeholders

Component req

Component testing

② Req are analysed and conflicts resolved through negoti

③ A req document is requir

④ Document is checked for consistency and completeness.

niceday

Business Requirement

Define project's business need, as well as criteria of its success.

If customer satisfaction ↑, business automatically ↑.

Most of business req is financial.

It must be resolved before functional and non-funct require.

It comes from funding sponsors, corporate executives, marketing managers

Success criteria (If financial should be comp on time)

1.1 Background

Describe history or situation provide information to previous proj

1.2 Business Opportunity

comparative evaluation of exist proj

1.3 Business Objectives

It is quantitative

should be measurable way

1.4 Vision Statement

concise vision statement.

For (Target customer)

Who statement of need or opportu

The product name

is product category

that major capabilities, key benefits

Unlike

Our product

Success Metrics

Evaluating the business objec

u measure objective if they are quantitative.

Business risks

Risks are competitors, navigation strategy

niceday

Product Vision

Project Scope

for release 1.0 for release 1.1 2.0 for n

V and S document

owner of document is project sponsor

Template for Document

→ 1. Business requirements

→ 2. Scope and limitation

→ 3. Business context

Date: _____

Assumption and dependencies
assume daily and monthly purchases, product ki dependenci kis cheeze per

Ecosystem Map

Showing the interaction of our system with another system after deploy

Bold Box Entire System

Boxes External system

Arrows

In scope, system in and system out are seen.

Context

Ecosystem

Context Diagram (0 Level Dia.)

define external entities (called terminators)

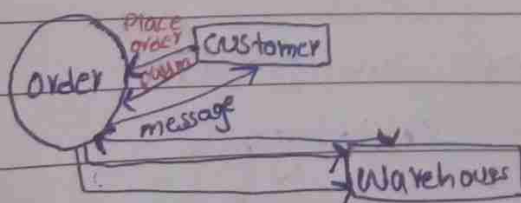
Notation:-

Circle → entire system

Rectangle → external entities (user classes, organizations, other system, hardware devices)

Arrows → flow of data.

Case Study:



Date: _____

ecosystem map event list feature

Deliverables

1.

2.

3.

4.

5.

6.

7.

8.

9.

Deliverables

1.

2.

3.

4.

5.

6.

Date: _____

Prototype:-

Use & discard
backend per
code implement
focus on interface

Use & improve
partial business industry.

logic implement
develop time
MVP gathering for capturing req.

Both are extensively used in

It is situation of elicitation

Basic function

Deliverables

implement

→ Notes during prsnl interview

development time

→ Responses to questionnaires administ

increase → save

→ " " using postal method

when developed

→ Responses to surveys

→ Formats & templates

→ Information from organiza records

→ Organizational process document,
standard and guidelines

→ flowcharts of process steps

→ information/notes on how

inputs are converted into output

Brainstorming:-

Group of people sit together
and discuss their ideas
independently and validate
your ideas

Requirement Gathering:-

Dispersed data to collect
have different resources.

Customer Complaints
suggestion of users

Planning Elicitation

→ Objective

→ Strategy (techniques)

→ Schedule

→ Documents

→ Expected products

→ Elicitation risks

Preparing for Elicitation

follow up :- plan for new
meeting for clarify question of prev meeting

Date: _____

Classify

Every test cases is against 3 possibilities of use case on every requirements.

Human being, software, hardware

1 use case \rightarrow multiple actors

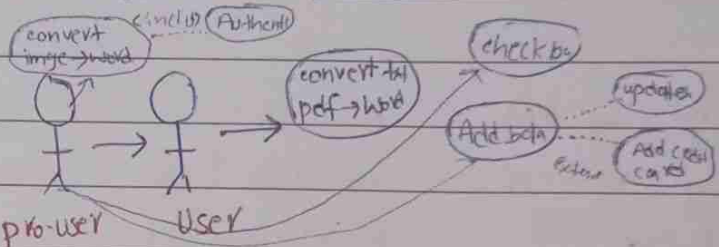
• Active verb present in use case

• Actor \rightarrow Noun, Use case \rightarrow Verb

Pro-user do every work which

base user do and some

additional works.



Extend :- Basic functionality me further add, arrow head towards base use-case.

Include :- Such use case which use, must be, pre-requisite be. Arrow towards adding new use. reusability ke case me. aek hi functionality different jagha me use ho.

• Use Case most in tabular form \rightarrow document
Descriptive form

Pitfalls in Requirements

\rightarrow Don't focus on solution

just say focus on require base user do and some

\rightarrow feasible features compulsory additional works.

\rightarrow Focus on time & budget during the process of requirements

\rightarrow Not preparing well when beginning prsnl interviewers

\rightarrow Prejudices

\rightarrow Untrained person as bus analyst

Use Case

It is used in require elicitation process.

How per focus nahi hota.

System level use case, level 0

we use it when we do testing

Sub-system level use case.

In regression, integration Testing, we do it.

Documenting Use Case::

Analysis of info collect in elri

Date: _____

1 Use Case ID UC-

1 Use Case Name

Actor

Pre-condition

flow of events Sequ & numbered Alternata

Post-condition

Use case ID

UC-01

Use Case Name

Validate PIN

Dependency

None

Actor ATM customer

Pre-condition Working ATM machine

available, card insert and P.M.

Flow of events:-

Customer insert card

System recognize

Alternatives

Post-condition Customer PIN is validate,

Enumerate

Verify

Evaluate: each req for feasibility, time

Bifurcate into \rightarrow PR \rightarrow FRs.

Group FR into logical groups

Group AR into " " " "

Identify requirem are duplicated

Identify contradictory to each other

Identify System interface