Unit - V

Project Management:

@ Building S/w is complex, many people involved and long period so project should be Managed.

@ It starts before technical work starts.

Definition:

@ Planning, Monitoring and control of the people process and events.

@ encompases of measurement metrics, estimation, risk analysis, schedules tracking and control.

The Management Spectrum:

SPM focuses on four P.

@ people
@ product
@ process
@ project

People: (people management capability matu Model). [PM-CMM] → guide organization in creation of a Mature Software process.

Recruiting, selecting, performance Man Training, career development, Team culture development.

The Product:

@ Product which must be met by S/w
@ Alternative
@ Technical
@ Objectives

Objective & Scope should be developer & customer. Solution considered.

Management consh identify the overall

@ task set → enable the framework adapted to the characteristics

@ task set → task, milestone, Quality assurance point.

Project:

@ Overall development cycle.

People:

@ Most important ingredient in the project.

Stake holders:

@ Senior Manager → define the business and control the practitioners.

@ Project Manager → plan, motivate, organize

@ Practitioners → deliver technical

@ customers → Specify requirements

@ End users → interact with the so

Team leaders:

@ Motivation - encourage technical people produce their best. (Push or pull)

@ organization - Mold existing process

@ Ideas or innovation - encourage people Create & feel creative

The Software team:

Best team depends on manager Style, no. of people in the team, their sk problem difficulty.

Problem factors: @ difficulty of the problem

of program created in high level languag

Estimation of project cost and effort:

Factors affecting project estimation:

The accuracy of a software project

- degree to which planner has properly estimate the size.
- ability to translate the size estimate i human effort, calender time & money.
- Degree to which the project plan reflect the ability of the s/w team.

Project estimate option:

Achieving reliable cost and effort.

- Delay estimation until project is compl
- Base estimate on similar projects. t have been already completed.
- Use decomposition technique to c cost and effort estimate.
- Use one or more empirical estim model. for cost and effort estimate.

1 is not practical 2 work reason 3 b4 done in landen.

Project estimation approaches:

- Decomposition technique (divide &
- Empirical estimation models (histo

Decomposition technique (problem-solving)

- Decompose the problem

is Quite similar to past effort
- less likely to be innovative

Random paradigm
- loosely and depend on individual initiative
- Team struggle when orderly performance is required.

Open paradigm:
- much of the innovation
- work is performed collaboratively with heavy communication and consensus based decision making.

Synchronous paradigm:
- natural compartmentalization of a problem
- Organize team members to work on piece of the problem
- little active communication

Agile team:
adopts many of the characteristic of Successfull Software project teams.

Coordination & communication:
- The scale of many development e is large, leads to complexity, confusion and difficulties in coordinating team.

The product:
- Quantitative estimates and organ plan are required.

Software Scope:
Context: How does the Sfw built to be system product or business

programmer

(a) difference would not affect → is long

(b) affect → on project

(c) Govern programma the cost of the project

Product complexity:

a) Application programs — 1

b) Utility programs — 3

c) System programs — 9

Programmer month (PM)

$PM_{ap} = 2.4 * (KDSI)^{1.05}$

$PM_{up} = 3.0 * (KDSI)^{1.12}$

$PM_{sp} = 3.6 * (KDSI)^{1.2}$

Calculating Development time for a pro[ject]

$TDEV_{ap} = 2.4 * (PM)^{1.05}$

$TDEV_{up} = 3.0 * (PM)^{1.12}$

$TDEV_{sp} = 3.6 * (PM)^{1.2}$

Calculating Average Staffing Level (SL):

$SL_{ap} = PM_{ap} / TDEV_{ap}$

$SL_{up} = PM_{up} / TDEV_{up}$

$SL_{sp} = PM_{sp} / TDEV_{sp}$.

Estimation of cost on line of code should include house keeping code.

Product size

(a) large size product

@ Partitioning or problem elabora.
@ The functionality and content that must be delivered
@ The process that will be used deliver it.

Melding the product and the process.

@ function to be orgineered must Pass through the set of framework activities.

@ communication, planning, modeling, Construction and deployment

Process decomposition:

@ Review customer Request
@ plan & schedule a formal, facilated meeting with all Stakeholders.
@ conduct research to Specify the proposed Solution & existing approaches.
@ Prepare working document & agenda
@ conduct the meeting
@ jointly develop mini-Specs, develop case → describe Sw from user point of view.
@ Review each mini Spec or use-case correctness, consistency & lack of ambiguity
@ assemble mini Spec is to Scoping ument.
@ Review Scoping document or Usecase
@ Modify Scoping document required.

logic Sizing:
- identify the type of application
- establish its magnitude on a qualitative scale
- refine the magnitude within original range

Function point sizing:
- develop estimates of information domain

Standard component sizing:
- estimate the no. of occurrence of each standard component.
- Use historical project data to
- estimate the delivered size per standard component.

Change Sizing: (existing S/w that is modified)
- estimate number and type of modification.

problem-based estimation:
LOC and FP data are used.
- as estimation variables to 'size' each elem
- as baseline metrics collected from past Projects and used in conjuction with Estimation variables to develop cost and effort Projections.
- Begin with bounded statement of
- Decompose the statement of scope problem
- estimate individually
- LOC & FP is then estimated for
- Alternatively choose another

... an overall ... ... avg ...

The expected size value S

$S = (S_{opt} + 4S_m + S_{pess})/6$

... historical LOC & FP data is then compared to S in order to cross ... process based estimation:

- Identify the set of functions the S/W need to perform
- Identify series of framework need to perform each function.
- Estimate the effort require to accomplish each function.
- Apply average labour rate to the effort.
- Compute total cost & effort for each function & each framework activity
- Compare the resulting values to those obtained by way of the LOC and FP estimate.

⇒ both set agree → highly reliable
⇒ otherwise conduct further investigation

Estimate with use case

- represent an external view of the ...
- do not address complexity of fn & features
- can describe ...
- ... effort.

n ... represents an adjustment b
Ly percent of Loc avg
defined locally → diff b/w this proj & avg
Sa - Actual Scenario per use case
Sh - Average Scenario per Use case
Pa - actual page per use case
Ph - avg page per Use case.

Reconciling estimates:

- Result gathered from various estim
technique must be reconciled to produce a
single estimate of effort, projects duration and
cost.
if widely divergent estimate occur then
investigate the following causes.
- The Scope of the project is not
adequately understood (or) misinterpreted.
- Productivity data used for problem
based estimation technique are inappropriate for
the application, obsolete, misapplied

Emperical Estimation Model:

- effort as a function of Loc &
- Loc & FP computed and entered
an estimation model.
- data are derived from limite

Sample.

COCOMO Model:

- Constructive cost Model
- Cost estimation Model
thousand of PM bas

Basic Model, Intermediate Model,

Organic mod:
- Relatively small
- Simple project with
- good experience

Semi-detached project:
- Intermediate projects
- Mixed experience.

Embedded projects:
- projects with tight hardware
- Software and operational constrain are handled.

COCOMO II Model:
- extension of cocomo Model.
- hierarchy of estimation Model

Application composition Model: Used in early st
Early design stage Model: Used once requir have been stabl
Post architecture stage Model: Used during co of the Software

Object point - count of no. of screen, reports, com
Complexity - no. of source & data tables
Once complexity determined those are weigh according to the table.
Object point is then counted Multiplying original no. of object by the weighting factor. and summing.
when increase is applied.
% of increase is estimated & object-

$E = 180\ Bt^3$ is person-months for $E > 2$ person months

t → years ; $P = 12{,}000$.

$t_{min} = 8.14 \times \frac{33{,}200}{12{,}000^{0.43}} = 12.6$ calendar months.

$E = 180 \times 0.28 \times (1.05)^3 = 58$ person-months.

Source Line of code:

- Only source line that are delivered are includ
- SLOC created by project staff.
- SLOC logical line of code
- Declaration are also counted
- comments are not counted.

Single source line may be several physical line.

- if-then-else → counted as one SLOC
  counted as several DSI.
- DSI → delivered source instructions.

Scale drivers:

- important factors contributing to a project duration & cost are scale drivers.
- Set each scale driver describe project.

5 Scale drivers are

Precedent, Development flexibility, Architecture Risk Reduction, team cohesion, process maturity

cost drivers:

personal factors	product factors	platform factors	proje
Analyst capability	Required S/w Reliability	Execution time constraints	use
Application experience	Database size	Main Storage constraint	Req.
programmer capability	S/w product complexity	Platform volatility	multi
personal continuity	Required Reusability		
platform experience	Documentation match		

② Project

Estimated effort = $\frac{NOP}{PROD}$

Developer's experience/ capability	very low	low	Nominal	high	
Environment maturity/capability	very low	low	Nominal	high	
PROD	4	7	13	25	

The Software equation:

dynamic multivariable model

Derived from productivity of

collected over 4000 contemporary.

$$E = \frac{LOC \times B^{0.333}}{P^3} \times \frac{1}{t^4}$$

E – effort in person-month

t – project duration

B – "Special skill factor" 13

P – "productivity parameter". → collected from past data

P = 2000 → Real time embeded S/w

P = 10,000 → telecommunication & system S/w

P = 28,000 → business S/w application.

2 independent

Effort = 2.94 * EAF * (KSLOC)^E

EAF — Effort adjustment factor derived from cost drivers.

E — Exponent derived from 5 scale drivers.

Effort adjustment factor

EAF = 1.34 * 1.09
= 1.46

COCOMO II Schedule equation:

No. of yrs to complete your S/w project.

Duration = 3.67 * (Effort)^{SE}.

avg. staffing = 42.3 person-months / (12.1 months)

= 3.5 people

Make-Buy decision:

→ Based on cost

① Purchase or buy the Software

② Reuse Existing partially built compon

③ Build the system from Scratch

④ Contract the Software develop me to an outside vendor.

Expected cost = $\sum$ path probability of decision tree Pa * Estimated path cost.

System 8 → Reuse

Expected cost$_{Reuse}$ = 0.40($2.75K) + [0.60 (
$310K) + 0.80 ($490K)

= $110K + [0.60 ($62K +

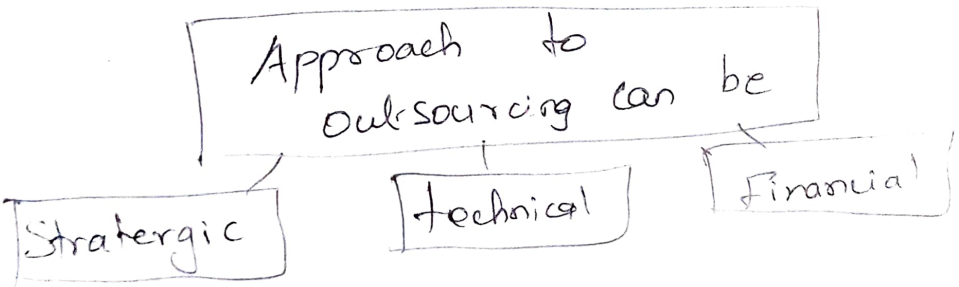
Contract = 0.60 ($ 350K) + 0.40 (

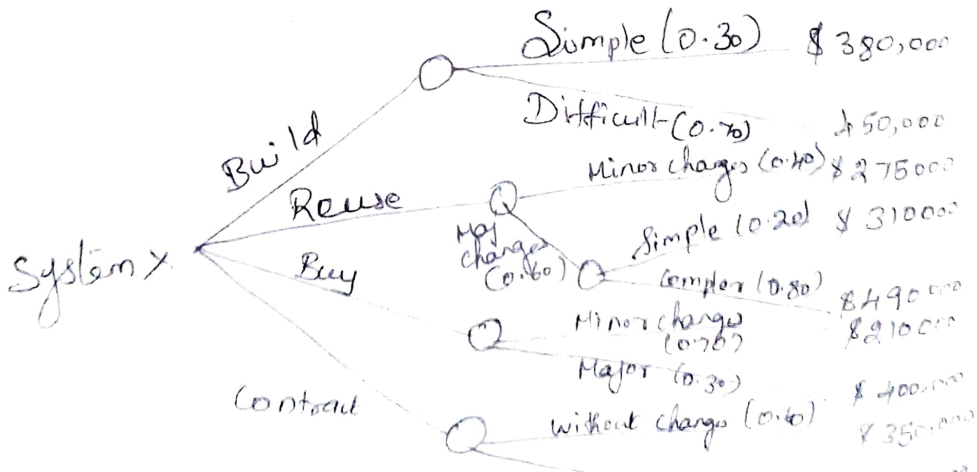
= $ 410 K.

- Availability of reliable S/w
- Experience of developer or vendor
- Conformance to Requirement
- Local politics
- Likelihood of changes in the Soft

Outsourcing

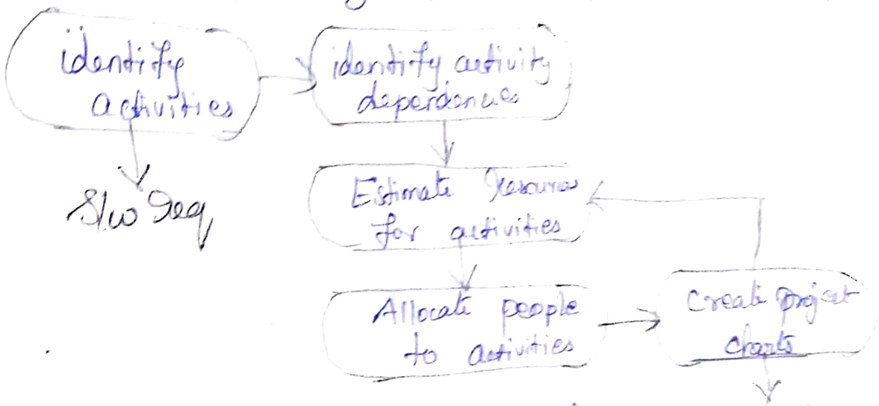
SE activities are contracted to a third Party who does the work at lower





Project. Estimate the time & resource of the activities.

- activities arranged in coherent sequence
- continually updated
- initial estimate can be made optimistically
- Total work is seperated in to small activities.
- time required for each activity must be determined by project manger.
- parallely conduct the task for perform



Problems may acquire → leave
→ H/w fail
→ resource not available

activity ... & bar charts

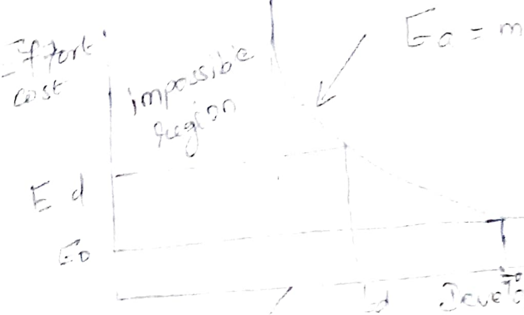
Relationship between people & Effort:

- Myth – More people then deadline can achived
- if More people all should be trained to tech time is simply wasted.

One effort determined – no. of people determined

$E_a = m(t_d{}^4 / t_a{}^4)$

E_a = effort in person-months
t_d = nominal delivery time schedule
t_o = optimal development time
t_a = actual delivery time



Product that must be accomplished ... particular project.

- Collection of taskset → S/w Project
- High Quality S/w developed Unnecessary work avoided
- task set vary upon type of the project

Concept development Project → new business idea or new application development Project → Satisfy specific ... Application upgradation Project → existing S/w application ... major ... Application maintenance Project → Correct, adapt or ... Reengineering Projects → legacy S/m are rebuilt ... or completely.

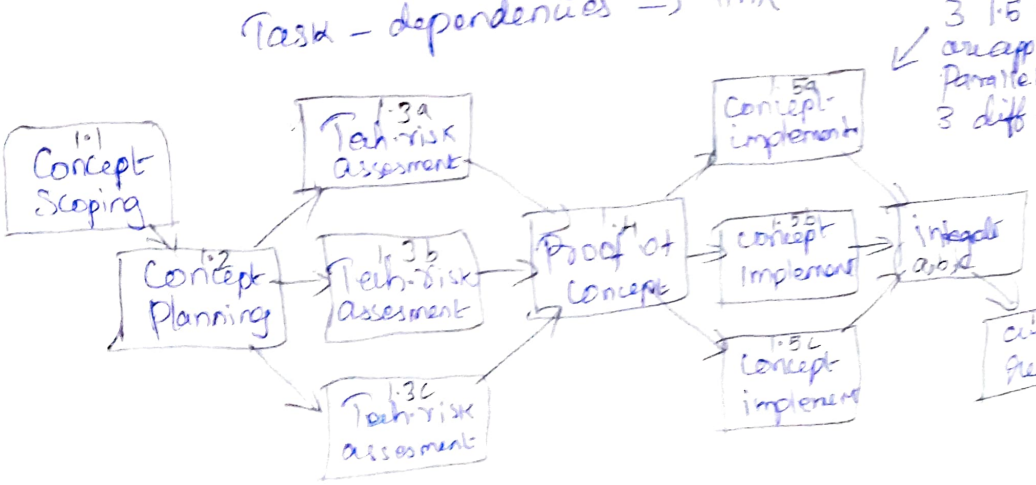
- Defining Scope, Planning, Evaluation of technology ... Concept implementation.

Task network

Task - Small unit of work

Node — activities

Task - dependencies ⇒ link



Up task that
① task are listed
② Horizontal bar indicate time line
③ Multiple bar then concurrency
④ ◆ Shape for milestone

Project table are prepared. all the tasks
listed along with actual start and end

Tracking Schedule
② Determine tasks and milestone
Proceeds
② conduct periodic Meeting:
– Various problem are discuss
② Evaluate the result of all project
② compare actual start date &
Scheduled start date of each of the project
task.
② Determine Milestone of project is
achieved on Schedule date
② Meet informally Software practition
to Solve many problem
help PM
② Assess the Progress of the
Project quantitatively.
② PM is responsible person.
when problem occur the project
Schedule is redefined.
② For severe deadline time box
ed. → series of increment product delivered
to customer.
② within time frame each task
must be implemented & completed
each boundary

for every task of the project.

@ How much % of the project completed

@ BCWS → Budgeted cost of work

@ Estimated cost for work for every individual task. Every point is calculated and total cost is sum

@ All BCWS are summed. Calculation of actual cost $BAC = \sum (BCWS_i)$

@ BCWP → Budgeted cost of work performed is computed. BCWP is sum of all BCWS.

@ BCWS → Values for the Project activities that are planned.

@ BCWP → values of the project activities that are completed.

$$SPI = BCWP / BCWS$$

Software Performance index → if value is 1.0 then project is very efficient.

$$SV = BCWP - BCWS$$

Schedule variance

Project schedule for completion = BCWS/BAC.

→ % of work completed by time t.

$$\text{Percent complete} = BCWP / BAC$$

work task that

@ Cost performance index → whether project is defined within defined budget or not.

@ 1.0 indicate project is within the defined budget.

@ EVA → identify Project performance, cost, Performance & project Scheduling difficulties. This Ultimately help the PM to take appropriate corrective actions.

Error tracking:

@ Developing S/w Project many errors may get generated. PM has to identify all these errors to bring quality of the Software.

@ Process of assessing the S/w in later task are called defect removel efficiency.

@ review → errors identified

@ error remain uncovered called defect

DRE = E / (E+D)

E → error

D → Defect.

E req → Error for requirement Spec

E design → Error per document - design

E code → component - code level

req analysis

Project plan

Prepared in advance in each stage of project completion. info available so it is iterative completed on

Project plan

Introduction
- goals, objectives & constrains

Project Organization
- no. of People involved along with their role must be described in detail.

Risk analysis:
- possibile Risk identified. Risk Reduction strategies must be decided.

Hardware & Software Requirements:
- Required H/w & S/w.

Work breakdown:
- Various project are grouped together to define the project work break down.
- Deciding project milestone & deliverables is an important task in defining the work breakdown

Project Schedule:
- Tentative Schedule of project activities must be determined.

Report generation:
- Structure of the project report, generated

Risk Management:

Risk → Uncertainity, causes heavy loses.

Risk Management → Making decision based on evaluation factors that threats to the business.

- Risk identification
- Risk Projection

requirement. ... , resource

Technical risk:

- offect Quality and timeline of the pr
- harder to solve

Business risk:

feasibility is suspect

Market risk: no customer
Strategic risk: not following company's business poli
Sales risk: how to sell is not clear
Management risk: responsibile staff leaves
Budget risk: losing overall budget of the prod

Charette proposed - (1) predictable risk (2) unpredictable ri

Known risk → can be identified evaluating project p

Predictable risk - identified in advance based on experi

Unpredictable risk - risk cannot be guessed earlier.

Reactive & proactive risk:

Reactive: corrective action is taken

- new risk comes up one after the other th
action correct problems rapidly.

So team flies in activities.

Fire fighting

- Resource is utilized still not get mana

then project in danger. no preventive action is take

handled on their occurance. older approach.

Proactive: considering probable risk.

- risk identified then probability & input ara
- Objective to avoid risk. (prevention

better than cure. not possible to avoid all risk

risk controlling can done efficiently

- intelligent strategy - used by ma

Step 1 Preparation of risk item check list

Product size: Risk based on overall size of s/w prod

Business impact: Market / Management can be predicted

Customer characteristics: Customer - developer communica

Process definition: exposes important risk items.

Development environment: associated with technology & tool

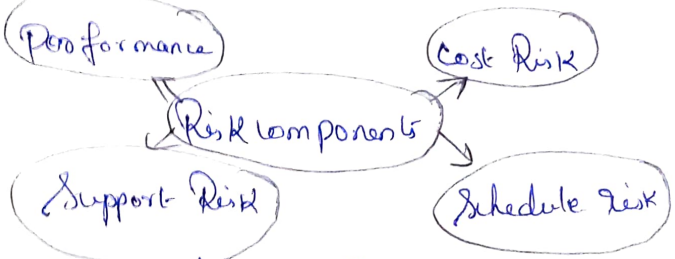
Staff size & Experience: associated with highly experienc Skilled staff.

Technology to be built: Complexity of the s/m should be Understood & related risk item need be iden

After check list Questionarrie is prepared.

Step 2 Creating risk component & drivers list

It is created with probability of occurence.



Performance risk: degree of uncertainity that product will satisfy the requirements.

Cost risk: degree of uncertainity project will maintain budg

Support risk: d.of. Un project developed easy to correct Modify or adapt.

Schedule risk: d.of. Un project will maintain the sched & project delivered in time.

Negigible | Marginal | Critical | catastrophic

Risk projection: (Risk estimation)

There are two way by which risk can be rated

Probability that the risk is real

Consequence of problem associated with the risk.

Risk monitoring
Risk Management.

Risk mitigation:
Preventive the risk occur (Risk av
- communicate with concerned staff to fin
- find out & eliminate all those
- Develop a policy to continue even staff
- everybody acquainted with current deve
- Maintain document in timely manner.
- conduct timely review in order to
- provide additional staff is req.

Risk Monitoring:
- Monitor behaviour of team members as pe
- degree in which team perform with sp
- the type of co-operation among team
- type of problem that are occurring
- availability of job within & outside org.

- check whether predicted risk occurs or not.
- Ensure avoidance step is applied properly or not.
- To gather info which can be useful for analyzing

Risk Management:
if PM successfull in applying the steps
Mitigation effectively then it becomes very much easy
to manage risk.

RMM plan: all risk analysis activities are described
Maintained in database s/m.
after documenting RMMM stopped.

CASE TOOL: [Computer Aided Software Engineering]
- easy to perform
- PM uses CASE
- reduces the amt of effort.
- used conjunction with process Model
- Quality Software

of multiple tool.

- May be single tool or collection
- Communicate with files, db, people, other CASE tools integrated environment.

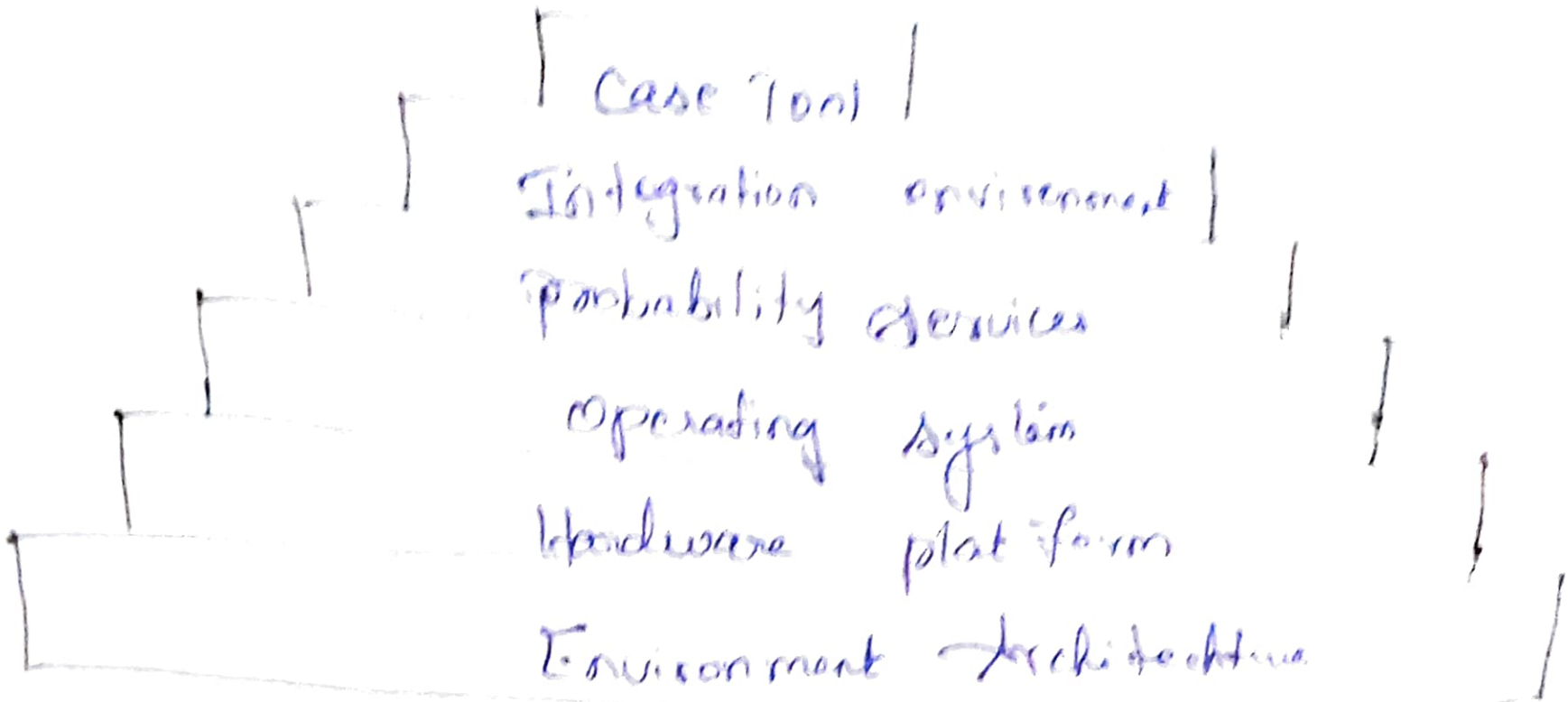


Fig: building block of CASE T

Unit - IV

Testing & Maintenance

Software testing fundamentals - internal & external views of testing, white box testing - basis path testing control structure testing - black box testing Regression testing - Unit testing - Integrating Testing - validating testing - System testing and debugging - Software implementation techniques: coding practices Refactoring - Maintenance & Reengineering BPR Model - Reengineering process Model. Reverse & Forward Engineering.