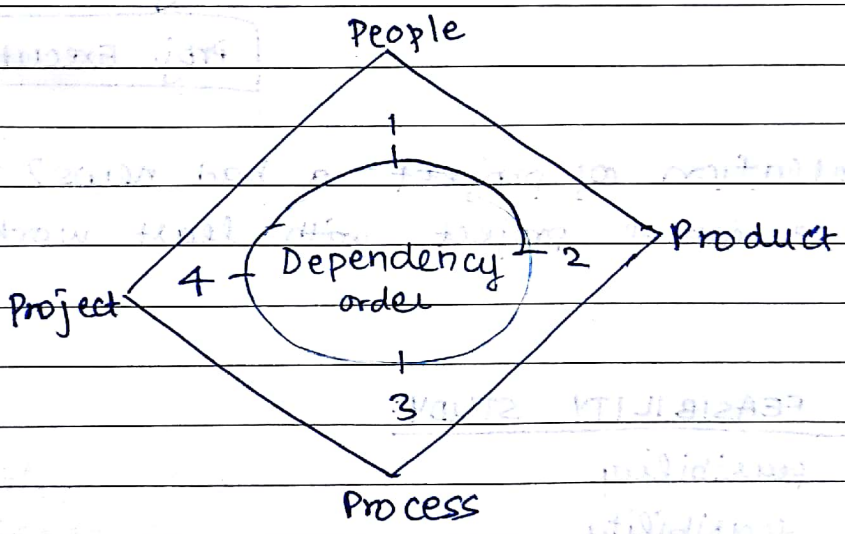
Page No.
Date

SPM

Factors contributing to s/w crisis:

→ Larger problems
→ Lack of adequate training
→ Increased skill shortage
→ Low productivity improvement

FACTORS



What is project?

→ Planned activity
→ Non-routine tasks
→ Planning need
→ Specific objectives
→ Predetermined timespan
→ Someone else does it

Software Projects vs Other proj.s

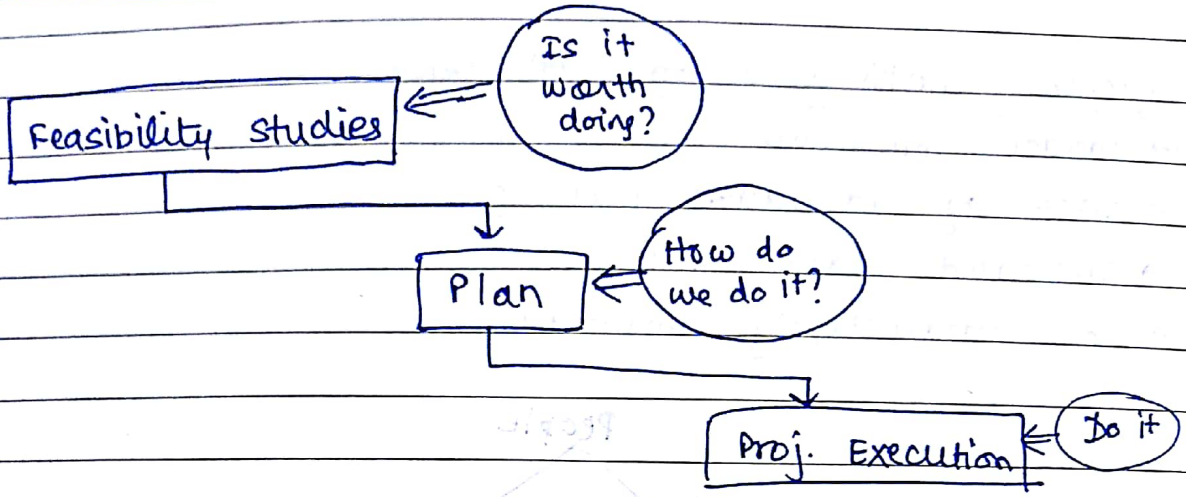
1. Invisibility
2. Complexity
3. Conformity
4. Flexibility

All these short topics → 1m/2m ques.

Ria

Page No.
Date

SPM Activities



→ Is cancellation of project a bad news? → Depends on state. → If proj. not started, then good.

→ How do we cancel project with least work? → By performing feasibility study properly.

TYPES OF FEASIBILITY STUDY:

→ Technical feasibility

→ Economic feasibility

Plan / Method / Methodologies

⇓	⇓	⇓
Used method to carry out an activity	To carry out a plan, we need a way to do it ⇒ method <How plan will be carried out>	Collection of set of methods.

Ria

Page No.
Date 8 8 18

SDLC Models

Softwares remain useful b/w reqt analysis phase & maintenance phase, (before retirement phase)

Using throwaway prototype model decreases software dev.^{t} cost.

- Waterfall
- Iterative enhancement
- Incremental
- Prototype
- Spiral

→ Build & fix
→ Waterfall
→ Incremental
→ Evolutionary process models < Rapid prototyping / Spiral
→ Agile Process models → XP
→ Adaptive s/w dev.
→ Dynamic sys. dev.
→ Scrum
→ crystal
→ Feature-driven dev.
→ Agile modelling
→ Object-oriented LC models → Fountain
→ Rational Unified Process.

Page No.
Date

Criteria for deciding model :

(1) organisation
(2) management style
(3) skills of employees
(4) Product nature

OBJECT - ORIENTED LIFECYCLE MODELS

Incorporates:

- Iteration
- Parallelism
- Incremental development

→ Diff. b/w traditional & OO approach
→ Fountain model
→ RUP

Assignment 1 on RUP (20-08-18)

RUP
- Static
- Dynamic

Static → 5 elements

- Role
- Activities
- Artifacts
- Discipline
- Workflows

Dynamic

- Inception
- Elaboration
- construction
- Transition

} Essential activities & artifacts

+ 3-4 diagrams

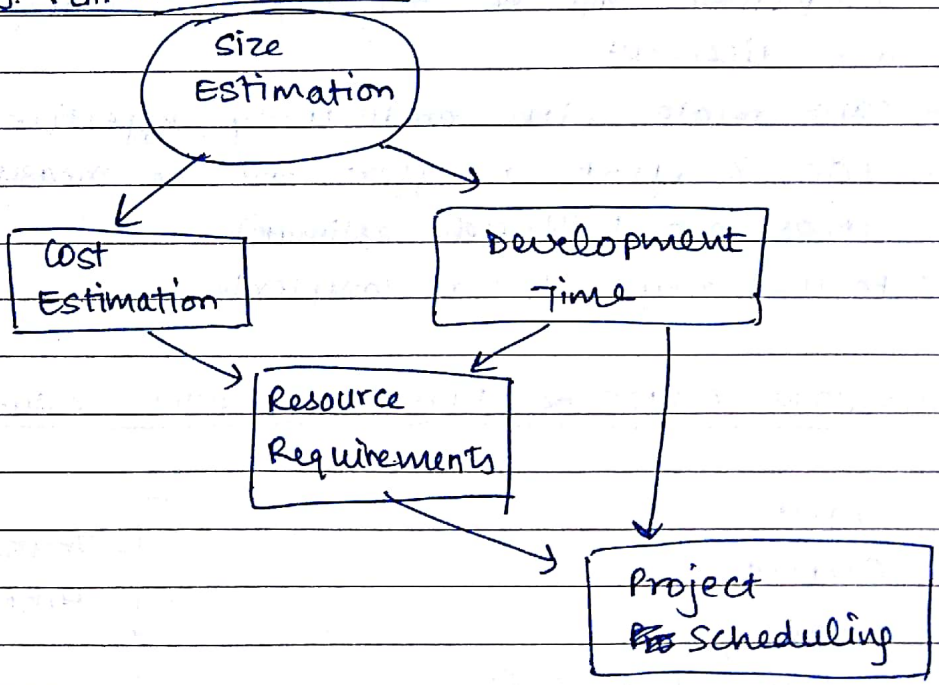
Page No.
Date

SOFTWARE PROJECT PLANNING

To conduct successful software project, we must have following info:

- Scope of work to be done
- Risk to be incurred
- Resources read
- Task to be accomplished
- cost to be expended
- Schedule to be followed

SOFTWARE PROJ. PLANNING ACTIVITIES :



(1) Size Estimation
- → LOC
- → Function count
- → Feature count

} in syll.

Ria

Page No.
Date

LOC → Defn → no comments, blank lines. & # preproc. are counted.

Advantages of LOC

- widely used & universally accepted
- Directly relates to end product
- Easily measured upon project completed
- Measures software from developer's point of view.

Drawbacks

- No universal definition
- ✓ Varies with programmer's ability
- ✓ Language dependent
- Difficult to measure before in new projects
- ✓ Comparison cant be performed if platform / language are different.
- ✓ cant relate with productivity / effectiveness of product.
- LOC & effort → effort cant be measured using LOC. (Does not accurately estimate)
- Focuses only on lexical complexity, not structural & algorithmic complexity.

FUNCTION COUNT or FPA (Func. Point Analysis) :

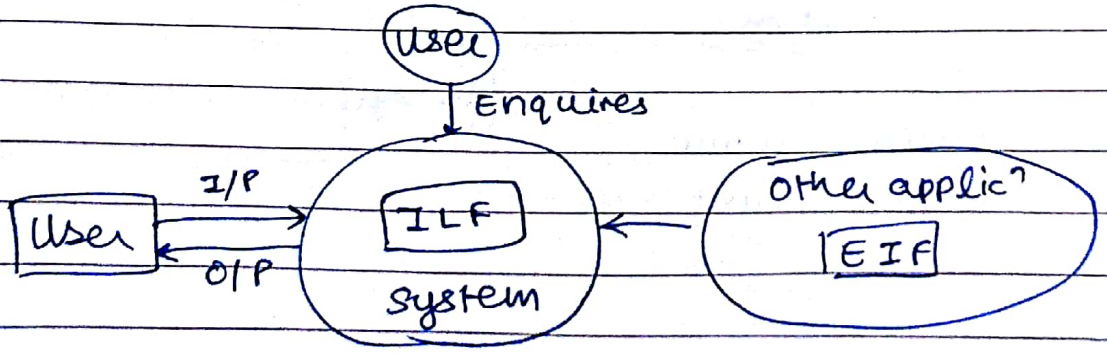
→ Based on user's POV. (perspective)

- Input
- Output
- Enquiry

} Transactional func" types

- Internal Logical Files
- External Interface Files

} Data functional types



Page No.
Date

$$UFP = \sum_{i=1}^{5} \sum_{j=1}^{3} z_{ij} w_{ij}$$ → unadjusted function point

$$FP = UFP * CAF$$

$$CAF = 0.65 + 0.01 * \sum_{i=1}^{14} F_i$$ → complexity Adjustment Factor

↓ Factors of influence.

Productivity = FP / PM person month
Quality = Defect / FP
Cost = Rs / FP
Documentation = Pages of documentation / FP

+1 class

Page No.
Date

COCOMO Model : Cost Constructive Model

COCOMO

Basic Intermediate Detailed

3 modes
- Organic → 2-50 KLOC → familiar, expert
- Semi-detached → 50-300 KLOC → Avg. experience
- Embedded → >300 KLOC → Realtime proj w/ complex interfaces v. little prev. experience

	Size (KLOC)	Experience	Innovation	Deadline
Organic	2-50	Familiar	Little	Not tight
Semi-detached	50-300	Avg. exp.	Medium	Medium
Embedded	300+	v. little exp	Significant	Tight

Effort, $E = a_b (KLOC)^{b_b}$ person months

Duration, $D = c_b (E)^{d_b}$ months

	a_b	b_b	c_b	d_b
Organic	2.4	1.05	2.5	0.38
Semi-detached	3.0	1.12	2.5	0.35
Embedded	3.6	1.20	2.5	0.32

Avg. staffing = $\frac{E}{D}$ ~~months~~ persons

Productivity = $\frac{KLOC}{E}$ KLOC / person months

Page No.
Date

Ques Size = 400 KLOC ⇒ Embedded

$E = 3.6 (400)^{1.2} = 4772.8$ person months $= 4773$

$D = 37.59$ months

Avg staffing $= \frac{E}{D} = 126.96$ persons $= 127$ ppl

Productivity $= \frac{400}{4772.8} = 0.0838$ KLOC/pm

Ques $E = 1133.12$. Avg exp & schedule not v. tight
Find KLOC

∴ semi-detached

$1133.22 = 3 (KLOC)^{1.12}$

$377.73 = x^{1.12}$

$x = 200$ KLOC

Intermediate → cost drivers

- Product Attr. — req. s/w reliability; size of application DB; complexity of project
- Hardware Attr. — RT performance constraint; Memory constraint; VM volatility; TA time
- Personal Attr. — Analyst cap.; programmer cap.; Application experience; VM experience; PL experience
- Proj. Attr — Modern prog. tactics; Use of SW tools; Reqd dev. schedule

15 cost drivers

Intermediate : $E = a_i (KLOC)^{b_i} \times EAF$

$D = c_i (E)^{d_i}$

	a_i
	3.2
	3.0
	2.8

Page No.
Date

Ques 400 KLOC. Embedded

Pool 1	Pool 2
- Very high capable - Little exp. in prog. lang. (PL) used.	- Dev. of low quality - Lot of exp. w/ PL used
↓	↓
0.82, 1.14	1.29, 0.95
EAF = 0.82 × 1.14	EAF = 1.29 × 0.95

Manager has to choose one pool of candidates. Which one should he hire? (Calc. effort & duration to compare)

$E_1 = 2.8 (400)^{1.2} = 3712.2 * EAF = 3470$

$E_2 = 4549$ E should be less ⇒ Pool 1

Ans: Pool 1

COCOMO: Detailed
- Phase sensitive effort multipliers
- Three level product hierarchy (module, subsystem, system level)

(Remember)

(1) Plan / Requirement Phase : Effort = 6 to 8%
Dev. time = 10 to 40%.

(2) Design Phase : Effort = 16 to 18%
Dev. time = 19% to 38%

(3) Programming : Effort = 48% to 68%
Dev. time = 18% to 64%

(4) Integration & Test : Effort: 16% to 34%
Time: 18% to 34%.

Page No.
Date

Adjustment Factor = 0.4 DD + 0.3 C + 0.3 I

DD → estimated design docs that need to be modified

S, Size (Equivalent) = $\frac{S \times A}{100}$

$E_p = M_p E$

$D_p = \tau_p D$

M_p & τ_p (bottom)

	Plan/Req	System Design	Detailed Design	Prog. (coding)	Integ & Test
Organic small (S≈2)	0.06	0.16	0.26	0.42	0.16
	0.10	0.19	0.24	0.39	0.18
Organic Medium (S≈32)	0.06	0.16	0.24	0.38	0.22
	0.12	0.19	0.21	0.34	0.26
Semidetached Medium (S≈32)	0.07	0.17	0.25	0.33	0.25
	0.20	0.26	0.21	0.27	0.26
semidetached large (S≈128)	0.07	0.17	0.24	0.31	0.28
	0.22	0.27	0.19	0.25	0.29
Embedded large (S≈128)	0.08	0.18	0.25	0.26	0.31
	0.36	0.36	0.18	0.18	0.28
Embedded Extra-large (S≈320)	0.08	0.18	0.24	0.24	0.34
	0.40	0.38	0.18	0.16	0.30

Page No.
Date

Project → To develop full screen editor:
(1) Screen edit = 4K
(2) Command line interpreter = 2K
(3) File I/O = 1K
(4) Cursor Movt = 2K
(5) Screen Movt = 3K

COCOMO: (1) Overall cost & schedule estimates
(2) cost & schedule for diff. phases.

Total: 12 KLOC ⇒ Organic → Intermediate

EAF = 1 (if not given)

$$E = a_i (KLOC)^{b_i} * EAF$$
$$= 3.2 (12)^{1.05} * 1$$
$$E = 43.48$$

$$D = c_i (E)^{d_i} = 2.5 (43.48)^{0.38}$$
$$= 10.48$$

COCOMO: Calculated LOC first
COCOMO 2: LOC independent

COCOMO II

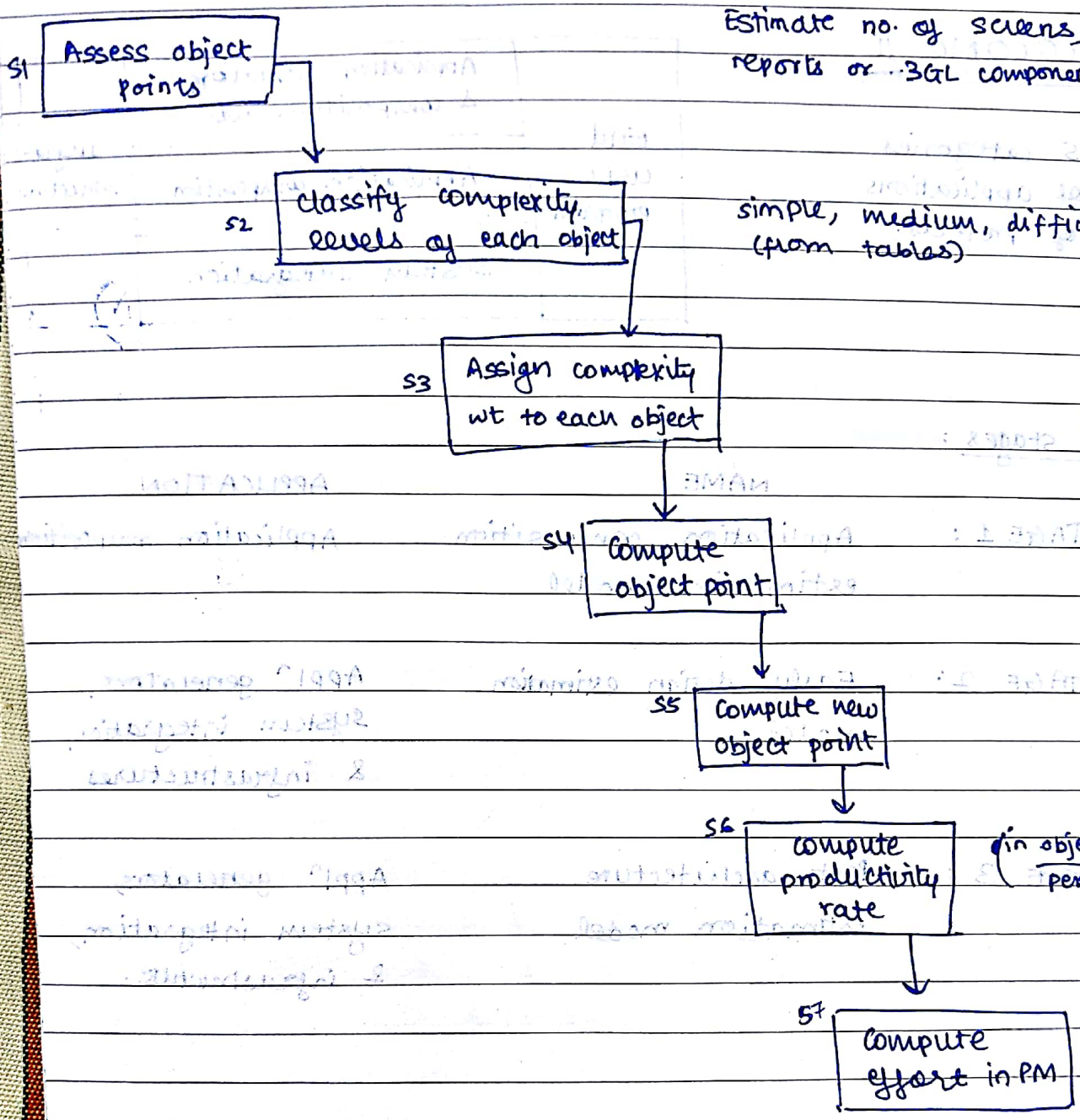
5 categories of applications of projects:

End user program	Application generations & composition aids	Infra-structure
	Application composition	
	System Integration	

5 stages:

	NAME	APPLICATION
STAGE 1:	Application composition estimation model	Application composition
STAGE 2:	Early design estimation model	Appl^n generators, system integration, & infrastructures
STAGE 3:	Post architecture estimation model	Appl^n generators, system integration, & infrastructure.

7 steps



(S1) Screens = 3 ⟨ 2 med. ⇒ 2×2+1×1 ⟩ → Add to get object point
1 simp.

2 reports = 2 ⟨ 1 → diff ⇒ 1×8+1×5
1 → med

(S2)

For screens :

No. of views contained	Total < 4 (<2 server, <2 client)	Total < 8 (2-3 server, 3-5 client)	Total 8+ (>3 server, >8 client)
<3	Simple	simple	Medium
3-7	Simple	Medium	Difficult
≥8	Medium	Difficult	Difficult

SOURCES OF DATA table

(S2)

For reports :

No. of sections contained	Total < 4 (<2 server, <3 client)	Total < 8 (2-3 server, 3-5 client)	Total 8+ (>3 server, >5 client)
0-1	Simple	simple	Medium
2-3	Simple	Medium	Difficult
4+	Medium	Difficult	Difficult

(S3)

Object Type	Complexity Weight		
	Simple	Medium	Difficult
Screen	1	2	3
Report	2	5	8
3GL Component	-	-	10

(S5)

$$NOP = \frac{\text{object point} * (100 - \%\ \text{reuse})}{100}$$

New obj. pt depends on s/w reuse.
NOP = OP if no reus

(56)

Productivity = NOP / PM

• Developer's experience & capability • ICASE maturity & capability	Productivity (NOP / PM)
Very low	4
Low	7
Nominal	13
High	25
Very high	50

(57)

$$\text{Effort} = \frac{\text{NOP}}{\text{Productivity}}$$

COCOMO II

Effort
- Application → COCOMO EM → size: estimated by obj. points
- Early design → scaling factor B → 5 factors; cost drivers (7) → mapping
- Post architecture → 17, CD

$$PM_{adj} = PM_{normal} * \left[\prod_{i=1}^{17} EM_i\right]$$

FP converted KLOC using:

	KLOC/FP
C	77
C++	
Ada	
Cobol	
Java	

(1) RELY

	V·low	low	Nom	High	V·high
=					
=					
=					

↑ 17 factors

How to calc. value of these 17 cost drivers?

5 factors: which define Product complexity ⇒ Define 17 EM

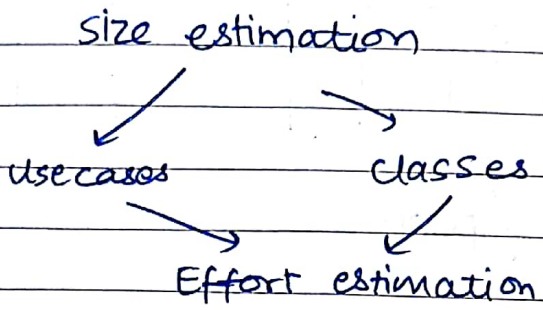
(a) control operations
(b) computational op^n
(c) Device dependent op^n
(d) Data management op^n
(e) User interface management op^n

$$T_{dev,\ nominal} = \left[\phi * (PM_{adj})^{[0.28 + 0.2(B - 0.091)]}\right] * \frac{SCED\ \%}{100}$$

$\phi = 3.67$

SCED → Schedule percentage

Object-Oriented Software Estimation



(I) Lorenz & Kidd Model

→ use of scenario scripts
→ Use of key & support classes

Interface Type	Multiplier
No GUI	2.0
Text-based	2.25
GUI	2.5
Complex GUI	3.0

No. of classes = 17 * No. of scenario scripts

2 types of classes - support
- key

Support Classes : Some common classes used by key classes

SC = No. of key classes * Multiplier

Total no. of classes = SC + KC

Effort = Total no. of classes * (10 to 20 person days)

Page No.
Date

Ques GUI & key class = 5 & 15 person days. Find effort

SC = $5 * 2.5 = 12.5$

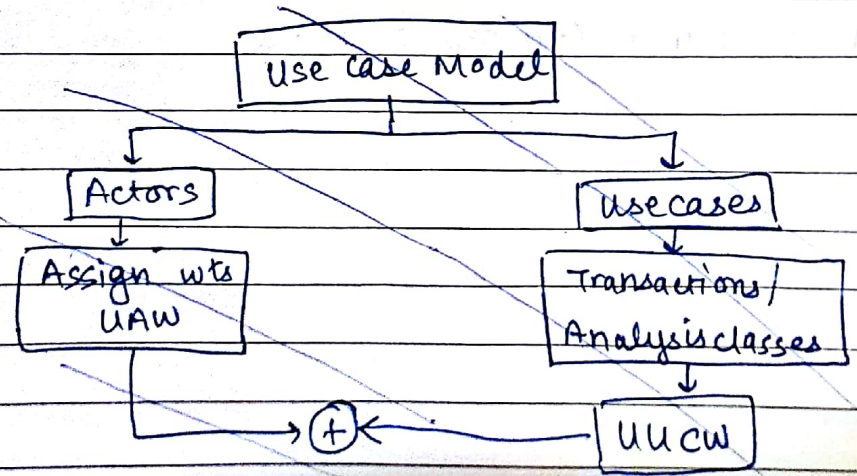
TC = $12.5 + 5 = 17.5$

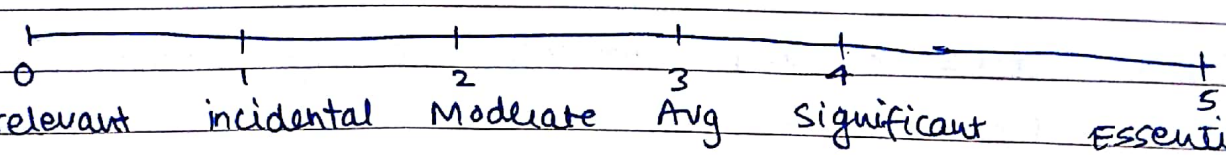
Effort = $17.5 * 15 = 262.5$

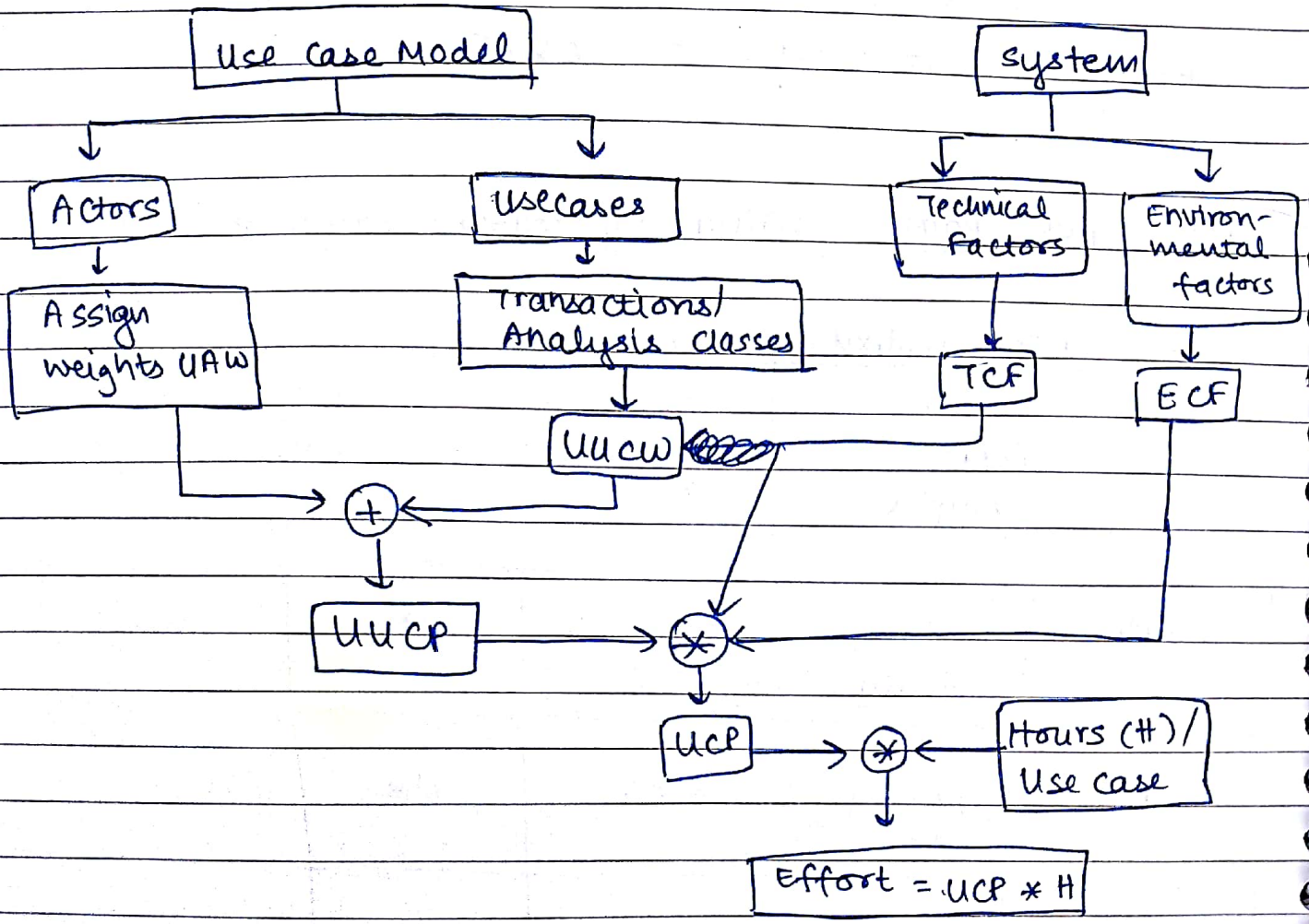
USE CASE POINT : Given by Gustav Karner in 1993

Action complexity	wts
simple	1
Avg	2
Complex	3

simple	no. of T ≤ 3 or Analysis obj ≤ 5	5
Avg	no. of T > 3 & < 7 or AO > 5 & < 10	10
Complex	no. of T ≥ 7 or AO ≥ 10	15







$$TCF = 0.6 + 0.01 \sum_{i=1}^{10} F_i \times W_i$$

$$ECF = 1.4 + (-0.03) \sum_{i=1}^{8} F_i \times W_i$$

Page No.
Date

TCF	W_i
1. Distributing sys	2
2. Application performance	1
3. End user efficiency	1
4. Complex internal processing	1
5. Reusability of source code	1
6. Installation ease	0.5
7. Ease of usability	0.5
8. Portability	2
9. Ease to change	1
10. Concurrency	1
11. Special security issue	1
12. Direct access for 3rd party	1
13 Special customer training provided	1

ECF Factors	W_i
1. Familiarity w/ process	1.5
2. Appl" experience	0.5
3. Analyst capability	0.5
4. Obj. oriented exp	1
5. Motivation	1
6. Part time workers	-1
7. PL difficulty	-1
8. Stable req	2

Not/Slides + Jorgenson { Intro, 4Ps, Mgmt Activities ← Jorgenson → SPM

Page No. _____
Date

Ques Consider airline reservation system.
No. of actors = 5
No. of usecases = 10
Assume all factors to be avg. compute use case point for the project.

~~UAW = 0~~

UUCW = 10 * 10 = 100

UAW = 5 * 2 = 10

UUCP = UUCW + UAW = 110

$TCF = 0.6 + 0.01 * \sum wi * 3 = 0.6 + 0.42 = 1.02$

$ECF = 1.4 + (-0.03) * 3 * 4.5 = 0.995$

$UCP = UUCP * TCF * ECF = 111.639$

Ques Given 20 hrs reqd for each usecase, calculate effort.
Effort = 111.639 * 20 = 2232.78 person hours.

Midsem Syllabus

1. Introduction ⇒ Jorg + slides
2. SDLC Models (Conventional + OO) ⇒ KK aggarwal wali book Lec
3. Software Project Planning
4. O.O software estimation
   - → Lorenz Kid Model
   - → UC Method
   - → class Point Method **
   - → OO Function point **

   (OOSE wali book)
5. Cost benefit analysis → Numericals from slide

Time it takes to start generating surplus of income over outgoing

-ve ⇒ investment
+ve ⇒ return (gains)

Page No.
Date

COST - BENEFIT ANALYSIS

Costs :

- Development cost
- Set up cost
- Operational cost

} identify costs

Benefits :

- Direct benefits
- Assessable indirect benefits
- Intangible benefits

} identify values of benefits

Check whether benefits are greater than costs

(1) NET PROFITS

Year	Cashflow
0	-100,000
1	10,000
2	10,000
3	10,000
4	20,000
5	100,000

⇒ Net profit = 50,000 (Σ cashflow)

(2) Payback period

⇒ kab net profit +ve ho jaaye (= 5th yr here)

(3) RETURN ON INVESTMENT (ROI) :

$$ROI = \frac{\text{Avg. annual profit}}{\text{Total investment}} \times 100$$

$$\text{Avg annual profit} = \frac{\text{Net profit}}{\text{No of yrs}} = \frac{50000}{5} = 10000$$

$$ROI = \frac{10000}{100000} \times 100 = 10\%$$

→ (-ve wala is investment)

Page No.
Date

(4) NET ~~PROFIT~~ PRESENT VALUES (NPV)

If interest rate was 10%, how much do I have to invest now to get Rs 100,000 in 1 year?
Annual rate by which we discount future earnings known as DISCOUNT RATE.

$$\text{PRESENT VALUE} = \text{Value in year } t * \underbrace{\frac{1}{(1+r)^t}}_{\text{discount factor}}$$

r = interest rate
t = no. of years

year	Project A ($r=8\%$)	Project B ($r=12\%$)
0	-8000	-10000
1	4000	2000
2	4000	2000
3	2000	6000
4	1000	2000
5	500	2000
6	500	2000

year	cashflow Proj. A	Discount factor $r=(10\%)$	Discounted cash flow
0	-100,000	$1/(1+0.1)^0 = 1$	-100,000
1	10,000	0.909090	9090.90
2	10,000	0.82644	8264.46
3	10,000	0.751314	7513.14
4	20,000	0.68301	13660.27
5	100,000	0.620921	62092.13

NPV = 628