Group-1

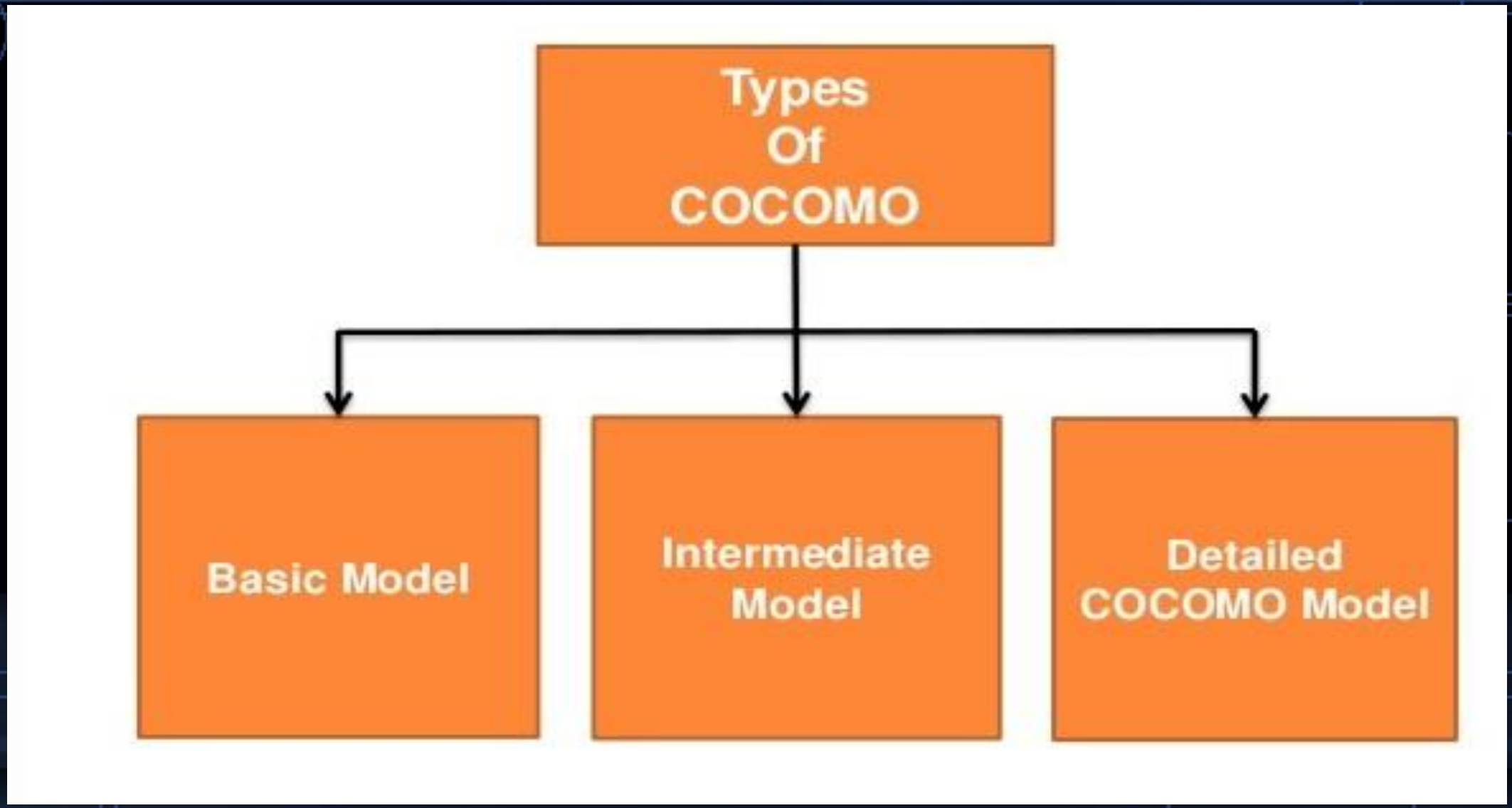
PROJECT NAME : ONLINE NEWS PORTAL

Presentation – 3 : COCOMO Method

Overview of COCOMO Method

- COCOMO stands for Constructive Cost Model
- Developed by Barry Boehm in 1981
- One of the most widely used S/W estimation model
- It is a open model as all of the details are published
- It is used to model :-
  - ✓ Effort required to develop S/W
  - ✓ Project Duration
  - ✓ Cost

Types of COCOMO Method



Steps of COCOMO Method

Step-1 : **Calculate the Count Total (CT) by 5 information domains**

- ***Number of user inputs***
- ***Number of user outputs***
- ***Number of user inquiries***
- ***Number of files***
- ***Number of external interfaces***

Step-2 : **Assign complexity level out of 5 & calculate the Function Points (FP) by the given formula**

$$FP = CT * [0.65 + 0.01 * \sum Fi]$$

Step-3 : After selecting Programming Language calculate the **Line Of Code (LOC)**

Thus the required **KLOC** is determined for the software project

Step-4 : This is the *final step* of the basic COCOMO model. Here we will find **Efforts** and **Duration** of the project.

Types of step-4 selection Method

Organic Model: Relatively small, simple software projects in which a small teams with good application experience work to a set of less than rigid requirement. The equation for the Effort (E) and Development time (D) for this model are:

E = 2.4 * (KLOC)^1.05 D = 2.5 * (E)^0.38

Semi-Detached Model: An intermediate (in size and complexity) software project in which teams with mixed experience levels must meet a mix of rigid and less than rigid requirements. The equation for the Effort (E) and Development time (D) for this model are:

E = 3.0 * (KLOC)^1.12 D = 2.5 * (E)^0.35

Embedded Model: A software project that must be developed within a set of tight hardware, software and operational constraints. The equation for the Effort (E) and Development time (D) for this model are:

E = 3.6 * (KLOC)^1.20 D = 2.5 * (E)^0.32

Construct the COCOMO model from our Project perspective

Use case – 1 News Category Module

Step-1 : Calculate Count Total (CT)

Information Domain Values

Measurement Parameter	Count		Simple ○	Average ◉	Complex ○		Total
Number of user inputs	2	X	3	4	6	=	8.00
Number of user outputs	2	X	4	5	7	=	10.00
Number of user inquiries	3	X	3	4	6	=	12.00
Number of files	2	X	7	10	15	=	20.00
Number of external interfaces	2	X	5	7	10	=	14.00
Count=Total							64.00

Count Total

Step-2 : Assign complexity & Calculate Function Points (FP)

Complexity Weighting Factors

// heading of the second table Rate each factor on a scale of 0 to 5:

(0 = No influence, 1 = Incidental, 2 = Moderate, 3 = Average, 4 = Significant, 5 = Essential):

Question	0	1	2	3	4	5
1. Does the system require reliable backup and recovery?	○	○	○	○	●	○
2. Are data communications required?	○	○	●	○	○	○
3. Are there distributed processing functions?	●	○	○	○	○	○
4. Is performance critical?	○	○	○	●	○	○
5. Will the system run in an existing, heavily utilized operational environment?	●	○	○	○	○	○
6. Does the system require on-line data entry?	○	○	○	○	●	○
7. Does the on-line data entry require the input transaction to be built over multiple screens or operations?	○	○	○	●	○	○
8. Are the master file updated on-line?	○	○	○	●	○	○
9. Are the inputs, outputs, files, or inquiries complex?	○	○	○	●	○	○
10. Is the internal processing complex?	○	●	○	○	○	○
11. In the code designed to be reusable?	○	○	●	○	○	○
12. Are conversion and installation included in the design?	○	○	○	●	○	○
13. Is the system designed for multiple installations in different organizations?	○	○	●	○	○	○
14. Is the application designed to facilitate change and ease of use by the user?	●	○	○	○	○	○
Total 30.00						

Show Total of weighting Factor

The Function Points is: Show Function Points 60.80

Step-3 : Calculate the Line Of Code (LOC)

Programming Language	LOC/FP (average)	Select
Assembly Language	320	○
C	128	○
COBOL	105	○
Fortran	105	○
Pascal	90	○
Ada	70	○
Object-Oriented Languages	30	◉
Fourth Generation Languages (4GLs)	20	○
Code Generators	15	○
Spreadsheets	6	○
Graphical Languages (icons)	4	○

LOC/F P: Show LOC/FP 1824.00

Step-4 : Calculate Efforts and Duration

Software Project	a_b	b_b	c_b	d_b	Select
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	○

Calculate Effort and Duration

$Effort\ (E) = a_b(KLOC)^{b}_{b} =$ 4.51 $\quad Duration\ (D) = c_b(E)^{d}_{b} =$ 4.43

Reset Data

Average Staff Size = (E/D) = 4.51/4.43 = 1 Person

Let salary of each developer = 35,000 BDT

Total Cost with Overhead = (35,000*1) * (2 * 4.43)

= 310100 BDT

Use case – 2 Add News Module

Step-1 : Calculate Count Total (CT)

Information Domain Values

Measurement Parameter	Count		Simple ○	Average ◉	Complex ○		Total
Number of user inputs	3	X	3	4	6	=	12.00
Number of user outputs	3	X	4	5	7	=	15.00
Number of user inquiries	3	X	3	4	6	=	12.00
Number of files	4	X	7	10	15	=	40.00
Number of external interfaces	1	X	5	7	10	=	7.00
Count=Total							86.00

Count Total

Step-2 : Assign complexity & Calculate Function Points (FP)

Complexity Weighting Factors

// heading of the second table Rate each factor on a scale of 0 to 5:

(0 = No influence, 1 = Incidental, 2 = Moderate, 3 = Average, 4 = Significant, 5 = Essential):

Question	0	1	2	3	4	5
1. Does the system require reliable backup and recovery?	○	○	○	○	○	●
2. Are data communications required?	○	○	●	○	○	○
3. Are there distributed processing functions?	●	○	○	○	○	○
4. Is performance critical?	○	○	○	●	○	○
5. Will the system run in an existing, heavily utilized operational environment?	○	○	○	○	○	●
6. Does the system require on-line data entry?	○	○	○	○	●	○
7. Does the on-line data entry require the input transaction to be built over multiple screens or operations?	○	○	●	○	○	○
8. Are the master file updated on-line?	○	●	○	○	○	○
9. Are the inputs, outputs, files, or inquiries complex?	○	○	○	●	○	○
10. Is the internal processing complex?	○	○	○	○	●	○
11. In the code designed to be reusable?	○	○	●	○	○	○
12. Are conversion and installation included in the design?	○	○	○	○	●	○
13. Is the system designed for multiple installations in different organizations?	○	○	○	●	○	○
14. Is the application designed to facilitate change and ease of use by the user?	●	○	○	○	○	○
Total 38.00						

Show Total of weighting Factor

The Function Points is: Show Function Points 88.58

Step-3 : Calculate the Line Of Code (LOC)

Programming Language	LOC/FP (average)	Select
Assembly Language	320	○
C	128	○
COBOL	105	○
Fortran	105	○
Pascal	90	○
Ada	70	○
Object-Oriented Languages	30	◉
Fourth Generation Languages (4GLs)	20	○
Code Generators	15	○
Spreadsheets	6	○
Graphical Languages (icons)	4	○

LOC/F P: [Show LOC/FP] 2657.40

Step-4 : Calculate Efforts and Duration

Software Project	a_b	b_b	c_b	d_b	Select
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	○

Calculate Effort and Duration

$Effort\ (E) = a_b(KLOC)^{b}_{b} =$ 6.70 $\quad Duration\ (D) = c_b(E)^{d}_{b} =$ 5.15

Reset Data

Average Staff Size = (E/D) = 6.70/5.15 = 1 Person
Let salary of each developer = 35,000 BDT
Total Cost with Overhead = (35,000*1) * (2 * 5.15)
= 360500 BDT

Use case – 3 Add to Archive Module

Step-1 : Calculate Count Total (CT)

Information Domain Values

Measurement Parameter	Count		Simple ○	Average ◉	Complex ○		Total
Number of user inputs	4	X	3	4	6	=	16.00
Number of user outputs	4	X	4	5	7	=	20.00
Number of user inquiries	2	X	3	4	6	=	8.00
Number of files	4	X	7	10	15	=	40.00
Number of external interfaces	2	X	5	7	10	=	14.00
Count=Total							98.00

Count Total

Step-2 : Assign complexity & Calculate Function Points (FP)

Complexity Weighting Factors

// heading of the second table Rate each factor on a scale of 0 to 5:

(0 = No influence, 1 = Incidental, 2 = Moderate, 3 = Average, 4 = Significant, 5 = Essential):

Question	0	1	2	3	4	5
1. Does the system require reliable backup and recovery?	○	○	○	○	○	●
2. Are data communications required?	○	○	○	●	○	○
3. Are there distributed processing functions?	○	○	●	○	○	○
4. Is performance critical?	○	○	○	●	○	○
5. Will the system run in an existing, heavily utilized operational environment?	○	○	●	○	○	○
6. Does the system require on-line data entry?	●	○	○	○	○	○
7. Does the on-line data entry require the input transaction to be built over multiple screens or operations?	○	○	○	○	○	●
8. Are the master file updated on-line?	○	○	○	○	●	○
9. Are the inputs, outputs, files, or inquiries complex?	○	○	○	●	○	○
10. Is the internal processing complex?	○	○	○	●	○	○
11. In the code designed to be reusable?	○	○	●	○	○	○
12. Are conversion and installation included in the design?	○	●	○	○	○	○
13. Is the system designed for multiple installations in different organizations?	○	○	●	○	○	○
14. Is the application designed to facilitate change and ease of use by the user?	●	○	○	○	○	○
Total 35.00						

Show Total of weighting Factor

The Function Points is: Show Function Points 98.00

Step-3 : Calculate the Line Of Code (LOC)

Programming Language	LOC/FP (average)	Select
Assembly Language	320	○
C	128	○
COBOL	105	○
Fortran	105	○
Pascal	90	○
Ada	70	○
Object-Oriented Languages	30	◉
Fourth Generation Languages (4GLs)	20	○
Code Generators	15	○
Spreadsheets	6	○
Graphical Languages (icons)	4	○

LOC/FP: Show LOC/FP 2940.00

Step-4 : Calculate Efforts and Duration

Software Project	a_b	b_b	c_b	d_b	Select
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	○

Calculate Effort and Duration

$Effort\ (E) = a_b(KLOC)^{b_b} =$ 7.45 $\quad Duration\ (D) = c_b(E)^{d_b} =$ 5.36

Reset Data

Average Staff Size = (E/D) = 7.45/5.36 = 1 Person

Let salary of each developer = 35,000 BDT

Total Cost with Overhead = (35,000*1) * (2 * 5.36)

= 375200 BDT

Total Duration and Cost

Total Duration :

= (4.43 + 5.15 + 5.36) Months

= 15 Months

Total Cost :

= (310100 + 360500 + 375200) BDT

= 1045800 BDT

Comparison

Strength of COCOMO

- More predictable about project
- Clearly understand with highly accuracy
- Account for various factors that affect cost of the project
- COCOMO is factual and easy to interpret
- The drivers are very helpful to understand the impact on the different factors that affect the project costs & duration

Weakness of COCOMO

- Ignore documentation & requirement
- Ignore co- operation, skill, knowledge & parameters
- Hardware requirements are denied
- Dependent on the amount of time spent in each phase
- Personal turnover levels aren’t used


Thank
You
Merci
Asante
Obrigado
Spasibo
Grazie
Kiitos
Dankon
Multumesc
Matondo
Arigato
Dank Je
Kia Ora
Salamat
Jaspaxar
Maake
Cam on ban
Mochchakkeram
Matur Nuwun
Terma Kasih
Nirringrazzjak
Chokrane
Mamana
Vinaka
Welalin
謝謝
感謝
תודות