**Function Point method**

**Description:-**

* A Function Point (FP) is a unit of measurement to express the amount of business functionality, an information system (as a product) provides to a user. FPs measure software size. They are widely accepted as an industry standard for functional sizing.

**There are two types of functions** −

* **Data Functions**
* **Transaction Functions**

**Data Functions**

There are two types of data functions −

* Master files
* Interface

Data Functions are made up of internal and external resources that affect the system.

**Master files**

* Master files are a user identifiable group of logically related data or control information that resides entirely within the application boundary. The primary intent of an ILF is to hold data maintained through one or more elementary processes of the application being counted.
* A Master files has some logical structure and it is stored in a file.

**Interface**

* Interface is a user identifiable group of logically related data or control information that is used by the application for reference purposes only. The data resides entirely outside the application boundary and is maintained in an interface by another application.

**Transaction Functions**

There are three types of transaction functions.

* External Inputs
* External Outputs
* External Inquiries

**External Inputs**

* External Input (EI) is a transaction function in which Data goes “into” the application from outside the boundary to inside. This data is coming external to the application.

1. Data may come from a data input screen or another application.
2. An EI is how an application gets information.
3. Data can be either control information or business information.
4. Data may be used to maintain one or more Internal Logical Files.

**External Outputs**

* External Output (EO) is a transaction function in which data comes “out” of the system. Additionally, an EO may update a Master files. The data creates reports or output files sent to other applications.

**External Inquiries**

* External Inquiry (EQ) is a transaction function with both input and output components that result in data retrieval.

|  |  |  |  |
| --- | --- | --- | --- |
| **Factors** | **Simple** | **Average** | **Complex** |
| Input | 3 | 5 | 10 |
| Output | 3 | 5 | 10 |
| Master files | 3 | 5 | 10 |
| Inquiries | 3 | 5 | 10 |
| Interface | 3 | 5 | 10 |

|  |  |  |  |
| --- | --- | --- | --- |
| **Factors** | **Values** | **Complexity Factor** | **Product** |
| Input | 11 | 5 | 55 |
| Output | 11 | 5 | 55 |
| Master files | 13 | 5 | 65 |
| Inquiries | 12 | 5 | 60 |
| Interface | 3 | 5 | 15 |

**Unadjusted Function** = Sum of product

= 55+55+65+60+15

=**250**

**Technical Influence Factors**:-

|  |  |
| --- | --- |
| Technical Influence Factors | Value [from 0 to 5] |
| Data communications | 0 |
| Performance | 3 |
| Heavily used configuration | 2 |
| Transaction rate | 2 |
| Online data entry | 3 |
| End user efficiency | 5 |
| Online update | 3 |
| Complex processing | 2 |
| Reusability | 3 |
| Installation ease | 0 |
| Operations ease | 5 |
| Portability | 4 |
| Maintainability | 3 |
| Distributed data processing | 0 |

**DI=41**

**TCF=0.65+ (0.01\*36) =1.01**

**FP=UFP\*TCF**

**=250\*1.01**

**=252 person month**

**Development Time:-**

**COCOMO Model**:-

COCOMO is one of the most widely used software estimation models in the world.

* This model is developed in 1981 by Barry Boehm to give estimation of number of man-months it will take to develop a software product.
* COCOMO predicts the efforts and schedule of software product based on size of software.
* COCOMO has three different models that reflect complexity
* Basic Model
* Intermediate Model
* Detailed Model

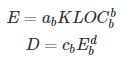
Similarly, there are three classes of software projects.

1. Organic mode:-In this mode, relatively simple, small software projects with a small team are handled. Such team should have good application experience to less rigid requirements.
2. Semi-detached projects: - In this class intermediate project in which team with mixed experience level are handled. Such project may have mix of rigid and less than rigid requirements.
3. Embedded projects: - In this class, project with tight hardware, software and operational constraints are handled.

**Basic Model**

The basic COCOMO model estimate the software development effort using only Lines of code

Various equations in this model are



Where, E is the effort applied in person-months,

D is the development time in chronological months and

KLOC is the estimated number of delivered lines of code for the project

**KLOC**=252\*53

=**13,382=13K**

**a=2.4**

**b=1.05**

**E=2.4(13) ^1.05**

**=35 person month.**

**Development Time=2.5(35) ^0.38**

**=9.65 Months**