FSPM Assignment # 2

HASSAN ALI
K20-1052 BSE-7B

Functional Points

- **External Inputs:** Admin login inputs, admin registration input, customer registration, payment information, employee registration, employee login, product detail, customer history. **(Count: 8)**

- **External Outputs:** Notification messages, status response, reports for sales or customer history. **(Count: 3)**

- **External Inquiries:** Product availability check, customer history, employee records. **(Count: 3)**

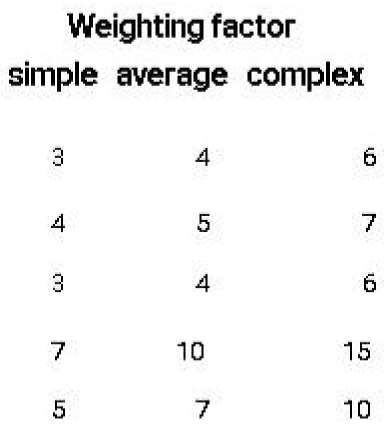
- **Internal Logical Files:** System configuration file, user account data, product data, booking records, payment records, Employee data. **(Count: 6)**

- **External Interface Files:** Database for records, third party payment system(assuming), external storage for backups. **(Count: 3)**

Weighting Factors

Assuming average for External input and external output, simple for External Inquiries and Internal logical files and complex for External Interface files

Total count = (8*4)+ (3*5)+(3*3)+(6*7)+(3*10) = 123



Weighting factor		
simple	average	complex
3	4	6
4	5	7
3	4	6
7	10	15
5	7	10

Complexity Adjustment Values

Calculating value adjustment factor based on 14 below points.

1- Reliable backup and Recovery: 3
2- Communication Required: 3
3- Distributed processing functions: 2
4- Performance Critical? : 4
5- System run in heavily utilized OS environment?: 2
6- On-line data entry: 3
7- Input transactions over multiple operations/screens: 3
8- Master files updated online: 3
9- Complex input, output, files, inquiries: 2
10- Internal processing complex: 2
11- Code reusability: 3
12- Conversion and installation included in design: 3
13- Multiple installations in different organizations: 2
14- Facilitate change and ease of use: 3

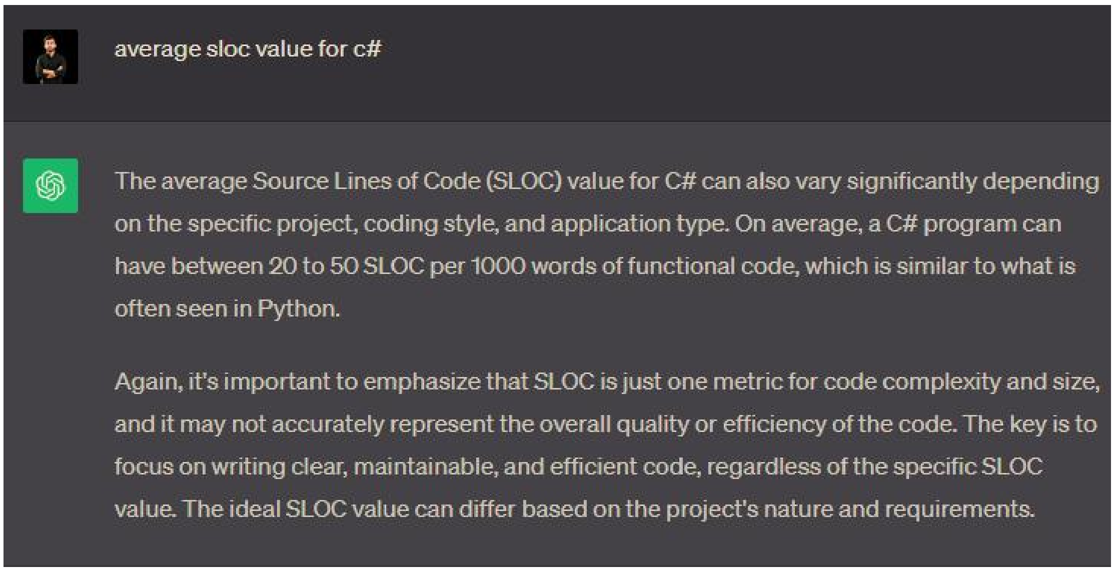
Total sum for VAF = 35

Computation of Functional Points

Weighted Functional Points= Count Total for FP * (0.65 + 0.01 * (Value Adjustment Factor))

Weighted FP= 123 * (0.65 + 0.01 * (35))
Functional Points = 123

Estimation of Lines of Code



average sloc value for c#

The average Source Lines of Code (SLOC) value for C# can also vary significantly depending on the specific project, coding style, and application type. On average, a C# program can have between 20 to 50 SLOC per 1000 words of functional code, which is similar to what is often seen in Python.

Again, it's important to emphasize that SLOC is just one metric for code complexity and size, and it may not accurately represent the overall quality or efficiency of the code. The key is to focus on writing clear, maintainable, and efficient code, regardless of the specific SLOC value. The ideal SLOC value can differ based on the project's nature and requirements.

Therefore taking Avg SLOC as 35 (70/2)
Lines of code= Weighted FP * Avg SLOC
LOC = 123 * 35 = 4305
LOC = 4.305 KLoc

Intermediate COCOMO

Effort (E) = a x (Size)^bx C
Where a=2.4 and b=1.05, C is cost drivers.
Size=kLoc
So E=2.4 x (3.51)^1.05 x C

Cost Drivers.

- RELY (Required Software Reliability): Since the software is intended to automate important operations in an electronic market, it should be fairly reliable. Let's assume it is "Nominal" (e.g., 1.00).
- DATA (Database Size): Given that the software will be handling bookings, product management, customer interactions, and more, we can assume that it will handle a considerable amount of data. Let's assume it is "High" (e.g., 1.14).

- CPLX (Product Complexity): The system includes a variety of features including login, registration, payment processing, booking, and more, which suggests it is complex. Let's assume it is "High" (e.g., 1.17).
- TIME (Execution Time Constraint): There's no specific information in the SRS or Project Charter about execution time constraints. Let's assume it is "Nominal" (e.g., 1.00).
- STOR (Main Storage Constraint): Similar to TIME, there's no specific information about storage constraints. Let's assume it is "Nominal" (e.g., 1.00).
- VIRT (Virtual Machine Volatility): There's no specific information about virtual machine volatility. Let's assume it is "Nominal" (e.g., 1.00).
- TURN (Computer Turnaround Time): No specific information about computer turnaround time is available. Let's assume it is "Nominal" (e.g., 1.00).
- ACAP (Analyst Capability): The SRS is well-documented, which suggests a skilled analyst team. Let's assume it is "High" (e.g., 1.19).
- AEXP (Applications Experience): The team is developing an Electronic Management System, which is somewhat specialized. Let's assume it is "Nominal" (e.g., 1.00).
- PCAP (Programmer Capability): The SRS does not provide enough information to determine this, but we'll assume the team is skilled. Let's assume it is "High" (e.g., 1.17).
- VEXP (Virtual Machine Experience): There's no information about virtual machine experience. Let's assume it is "Nominal" (e.g., 1.00).
- LEXP (Programming Language Experience): The system will be developed using C#, a well-known language. Let's assume it is "High" (e.g., 1.09).
- MODP (Modern Programming Practices): The SRS does not provide enough information to determine this. Let's assume it is "Nominal" (e.g., 1.00).
- TOOL (Use of Software Tools): There's no information about the use of software tools. Let's assume it is "Nominal" (e.g., 1.00).
- SCED (Required Development Schedule): The project has specific milestones with target dates, which indicates some schedule pressure. Let's assume it is "Nominal" (e.g., 1.00).

EAF = RELY * DATA * CPLX * TIME * STOR * VIRT * TURN * ACAP * AEXP * PCAP * VEXP * LEXP * MODP * TOOL * SCED = 1.00 * 1.14 * 1.17 * 1.00 * 1.00 * 1.00 * 1.00 * 1.19 * 1.00 * 1.17 * 1.00 * 1.09 * 1.00 * 1.00 * 1.00 = 1.32

Final Calculations

So E=2.4 x (4.305)^1.05 x 1.32
E= 9.6 staff-months

So **10 staff-months rounded up**

The **development time** expected is:
2.5x(E)^0.38
2.5 x (9.6)^ 0.38 => 5.9 => **6 months.**

Average staff members required:
Effort/Tdev so 9.6/5.9 => 1.62 so 2 staff-member developer required.

Productivity required from the staff person is:
Size/Effort so 4305/9.6 => 448.4 so 450 lines of code per staff-month.

Average Salary for C# developer = 75000

Cost = Effort x Average Salary = 9.6 * 75000 = 720000 PKR

Results

Average lines of code expected	**4.305 KLoc**
Effort in staff-months	**9.6 staff-months**
Time of development	**6 months**
Developers Required	**2 developer required**
Productivity expected	**450 Lines of code per staff-month**
Cost Associated	**720,000 PKR**