Department of Computer Engineering

Academic Term: First Term 2023-24

Class: T.E /Computer Sem – V / Software Engineering

Practical No:	**4**
Title:	**Calculating function points of the Project**
Date of Performance:	
Roll No:	**9567, 9552 , 9623**
Team Members:	**Shruti Patil, Mrunal Kotambkar, Dhruv Mayekar**

Rubrics for Evaluation:

Sr. No	Performance Indicator	Excellent	Good	Below Average	Total Score
1	On time Completion & Submission (01)	01 (On Time)	NA	00 (Not on Time)	
2	Theory Understanding(02)	02(Correct)	NA	01 (Tried)	
3	Content Quality (03)	03(All used)	02 (Partial)	01(rarely followed)	
4	Post Lab Questions (04)	04(done well)	3 (Partially Correct)	2(submitted)	

Signature of the Teacher:

Department of Computer Engineering

Academic Term: First Term 2022-23

Class: T.E /Computer Sem – V / Software Engineering Signature of the Teacher:

Aim

To calculate function points for the E-Waste Management System.

a) **Number of external inputs (EIs)**
   1. Tie up with Recycling centers and Tech giants
   2. Selling Details (images, cost, functionality etc)
   3. Payment Details
   4. Open source posts
   5. System workers
   6. User credentials

b) **Number of external outputs (EOs).**
   1. Billing details
   2. Online & Offline meeting schedule
   3. Waste pickup schedule
   4. Errors and acknowledgement messages

c) **Number of external inquiries (EQs).**
   1. Device troubleshooting meet scheduling
   2. DIY projects
   3. Meeting & Waste pickup scheduling

d) **Number of internal logical files (ILFs)**
   1. Home page interface with navigation to all sections
   2. Transaction section
   3. Open source platform
   4. Meets & Pickup scheduling section

e) **Number of external interface files (EIFs)**
   1. Google maps
   2. Payment gateways
   3. Online Meeting gateways

Information Domain Value	Count		Weighting factor Simple	Average	Complex		
External Inputs (EIs)		×	3	4	6	=	
External Outputs (EOs)		×	4	5	7	=	
External Inquiries (EQs)		×	3	4	6	=	
Internal Logical Files (ILFs)		×	7	10	15	=	
External Interface Files (EIFs)		×	5	7	10	=	
Count total	→						

The Fi (i=1 to 14) are value adjustment factors (VAF) based on responses to the following questions:

1. Does the system require reliable backup and recovery?
   5-Essential

2. Are specialized data communications required to transfer information to or from the application?
   5-Essential

3. Are there distributed processing functions?
   0-No influence

4. Is performance critical?
5-Essential

5. Will the system run in an existing, heavily utilized operational environment?
4 - Significant

6. Does the system require online data entry?
5 - Essential

7. Does the online data entry require the input transaction to be built over multiple screens or operations?
5 - Essential

8. Are the ILFs updated online?
3 - Average

9. Are the inputs, outputs, files, or inquiries complex?
5 - Essential

10. Is the internal processing complex?
4 - Significant

11. Is the code designed to be reusable?
4 - Significant

12. Are conversion and installation included in the design?
5 - Essential

13. Is the system designed for multiple installations in different organizations?
4 - Significant

14. Is the application designed to facilitate change and ease of use by the user?
5 - Essential

POSTLAB:

Critically evaluate the Function Point Analysis method as a technique for software sizing and estimation ,discussing its strengths and weaknesses.

Function Point Analysis (FPA) is a widely used method for software sizing and estimation. It quantifies the functionality provided by a software application, which is independent of technology and programming language. While FPA has been valuable in many software development projects, it has its strengths and weaknesses:

Strengths of Function Point Analysis:

1. Technology-agnostic: FPA is technology-independent. It focuses on the functionality provided by the software rather than the underlying technology. This makes it applicable to various types of software projects, regardless of the programming language or platform used.

2. User-Centric: FPA takes a user-centric perspective by measuring the software's functionality and how it serves the needs of end-users. This focus on user requirements helps in aligning software development with user expectations.
3. Standardized Measurement: FPA is based on well-defined rules and guidelines, as provided by the International Function Point Users Group (IFPUG). This standardization makes it easier for organizations to use FPA consistently across different projects and teams.
4. Estimation Accuracy: When applied correctly, FPA can provide accurate estimates for project effort, cost, and duration. By quantifying the functionality, it enables more informed decision-making in project planning and management.
5. Basis for Benchmarking: FPA allows organizations to benchmark their projects against industry standards and best practices. This helps in identifying areas for improvement and achieving better project outcomes. Weaknesses of Function Point Analysis:

1. Complexity: FPA can be complex and time-consuming, especially for large and complex software projects. It requires detailed analysis of requirements, which can be challenging to perform accurately.
2. Subjectivity: FPA requires judgment and subjectivity in determining factors like complexity and weighting. Different analysts may interpret requirements differently, leading to potential variations in estimation.
3. Learning Curve: Training and experience are necessary to become proficient in FPA. This can be a barrier for small organizations or teams that lack the resources to invest in training.
4. Limited Coverage: FPA primarily measures the functional aspects of software, which are user-facing. It may not fully capture non-functional requirements, architectural complexities, or technical constraints, which are also critical in software development.
5. Maintenance Overhead: As the project evolves, maintaining the FPA counts may require constant updates and adjustments, which can be resource intensive.
6. Not Suitable for All Projects: FPA may not be appropriate for certain types of projects, such as research and development efforts, highly innovative projects, or projects with vague or rapidly changing requirements.
7. Overhead for Small Projects: For very small projects, the overhead associated with FPA may not be justified. The effort required for FPA analysis might outweigh the benefits of estimation accuracy.

Propose strategies to manage and mitigate uncertainties in function point estimation and how they can impact project planning and resource allocation.

Managing and mitigating uncertainties in function point estimation is crucial for accurate project planning and resource allocation. Function point estimation is subject to various uncertainties due to factors like changing requirements, inexperienced estimators, and the inherent complexity of software projects. Here are strategies to help manage and mitigate these uncertainties and their impact on project planning and resource allocation:

1. Use Historical Data:
   - Strategy: Rely on historical data from previous projects to identify trends and patterns in function point counts and project outcomes.
   - Impact on Project Planning: Historical data provides a baseline for estimation, allowing for more accurate resource allocation and schedule planning.
2. Expert Involvement:
   - Strategy: Involve experienced estimators who are well-versed in the specific FPA method and the domain of the project.
   - Impact on Project Planning: Expertise reduces estimation uncertainty and results in more accurate function point counts, leading to more realistic project plans.
3. Continuous Requirements Refinement:
   - Strategy: Continuously refine and clarify requirements as the project progresses. This helps in reducing ambiguities and uncertainties related to scope.
   - Impact on Project Planning: Frequent requirement updates can lead to more accurate estimations and better resource allocation.
4. Sensitivity Analysis:
   - Strategy: Conduct sensitivity analysis to identify the impact of potential changes in function point counts on the project schedule and resource needs.
   - Impact on Project Planning: Sensitivity analysis helps in quantifying the potential impact of estimation uncertainties, allowing for better risk management and resource contingency planning.
5. Buffer Allocation:
   - Strategy: Allocate buffers or reserves in the project schedule and resource allocation to account for estimation uncertainties.
   - Impact on Project Planning: Buffers provide flexibility and help in

accommodating potential delays or resource overages due to uncertainties. 6. Range Estimation:

- Strategy: Instead of providing a single-point estimate, estimate a range of function points (e.g., best-case, worst-case, most likely).
- Impact on Project Planning: Range estimation helps in acknowledging the uncertainty and allows for more conservative resource allocation. 7. Collaborative Estimation:
- Strategy: Involve cross-functional teams, including developers, business analysts, and users, in the estimation process to reduce biases and errors. • Impact on Project Planning: Collaborative estimation helps in capturing a broader perspective and more realistic estimations.

8. Regular Estimation Reviews:

- Strategy: Periodically review and update function point estimates as the project progresses, considering new information and changes. • Impact on Project Planning: Regular reviews help in keeping estimations aligned with project reality, leading to more accurate resource allocation. 9. Documentation and Traceability:
- Strategy: Maintain thorough documentation of the estimation process, including assumptions and constraints. Ensure traceability of requirements to function points.
- Impact on Project Planning: Proper documentation and traceability provide transparency and help in revisiting and refining estimations when necessary.

10. Risk Management:

- Strategy: Develop a risk management plan that addresses estimation uncertainties and outlines mitigation strategies.
- Impact on Project Planning: A well-defined risk management plan prepares the project team to deal with potential issues arising from estimation uncertainties.