Assignment. No.03

Semester	T.E. Semester V – Computer Engineering
Subject	Software Engineering
Subject Professor In-charge	Dr. Sachin Bojewar
Assisting Teachers	Dr. Sachin Bojewar

Student Name	Deep Salunkhe	
Roll Number	21102A0014	
Grade and Subject Teacher's Signature		

Assignment Number	03
Assignment Title	LC/NC(low code or no code)

Do research to understand concept of LC/NC (low code or no code) and prepare a short report max 300 words to cover following:
- **It's impact on LOC estimation.**
- **Suggest parameters to be considered for size estimation in LC/NC.**

In recent years, Low Code (LC) and No Code (NC) development platforms have gained substantial momentum in the software development landscape. These platforms allow developers to create applications with minimal hand-coding, significantly reducing the lines of code (LOC) traditionally associated with software projects. This paradigm shift has profound implications for LOC estimation and necessitates a revaluation of size estimation parameters.

Impact on LOC Estimation:
1. **Reduced LOC:** LC/NC platforms empower developers to achieve a wide range of functionalities with visual interfaces and pre-built components. As a result, traditional LOC estimates, which rely on manual coding efforts, may no longer accurately reflect the complexity and effort required for LC/NC projects.
2. **Ambiguity in Measurement:** Estimating LOC in LC/NC projects becomes challenging due to the absence of conventional code files. While some platforms generate code in the background, it is often hidden from the developer. This opacity makes it difficult to ascertain the exact number of lines of code produced.
3. **Focus on Functionality:** Rather than counting lines of code, LC/NC emphasizes delivering desired functionalities. Therefore, traditional LOC metrics may become less relevant as success metrics shift towards meeting user requirements efficiently.

Suggest Parameters for Size Estimation in LC/NC:
1. **Function Points (FP):** Transitioning from LOC to Function Points is an effective size estimation approach for LC/NC projects. FP measures the application's functionality, considering inputs, outputs, and user interactions. It provides a more meaningful metric for project size and complexity.
2. **User Stories:** LC/NC projects often employ Agile methodologies and User Stories to define requirements. Estimating size based on the number and complexity of User Stories provides a practical alternative to LOC.
3. **Component Count:** Assessing the number of components (widgets, screens, modules) created within the LC/NC platform can serve as a proxy for project size. This can be further refined by categorizing components by complexity.

Assignment. No.03

4. **Integration Points:** Consider the number of integrations required with external systems or databases. Complex integrations may indicate a larger project size.
5. **Data Volume:** For data-intensive applications, size estimation can incorporate the volume of data to be processed, stored, or migrated.
6. **User Base:** The number of users or concurrent users can influence project size. Large user bases may require more extensive testing and scalability measures.
7. **Customization Level:** Assess the degree of customization needed within the LC/NC platform. Highly customized solutions may involve more effort and, consequently, a larger project size.

In conclusion, LC/NC development is reshaping the software development landscape by reducing the emphasis on lines of code and prioritizing functionality. To adapt to this paradigm shift, software professionals should consider alternative size estimation parameters such as Function Points, User Stories, component count, integration points, data volume, user base, and customization level. These parameters provide a more accurate reflection of project size and complexity, aligning estimation practices with the evolving nature of software development.