Metrics For Requirements

Team listItGrocery

08.05.2023

What is Function Point metric

- developed by Allan J. Albercht in 1979 at IBM, further modified by the International Function Point Users Group (IFPUG).
- express the amount of business functionality an information system (as a product) provides to a user
- measures software size based on inputs, outputs, inquiries, files and interfaces
- helps to estimate the development effort, cost, and schedule of a software project
- can be used for benchmarking, productivity improvement, and quality assurance

How to measure requirement metrics

- measuring quality, completeness and consistency
- quality metrics:
    - Completeness
    - Correctness
    - Consistency
    - Understandability
    - Testability
    - Stability
    - Volatility
    - Traceability

How to calculate Function Point metric

- Identify the Functional User Requirements
- Categorize the Requirements:
    - External Inputs (EI)
    - External Outputs (EO)
    - External Inquiries (EQ)
    - Internal Logical Files (ILF)
    - External Interface Files (EIF)

- Assign Complexity Weights
- Count the Instances

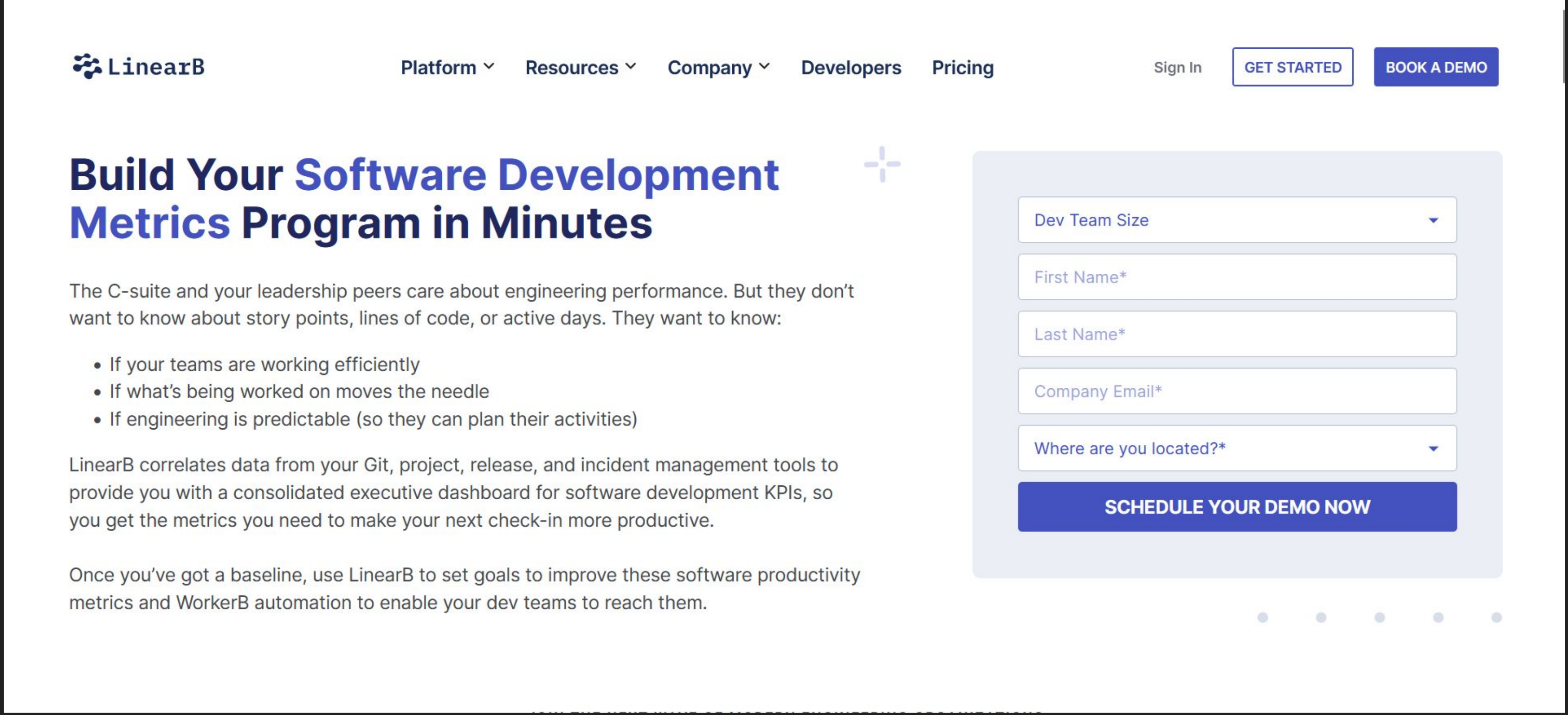
- Count the Instances:
    - Calculate Unadjusted Function Points (UFP)
        - UFP = (Low complexity weight x Number of Low instances) + (Average complexity weight x Number of Average instances) + (High complexity weight x Number of High instances)
    - Calculate Technical Complexity Factor (TCF)
    - Calculate TCF Adjusted Function Points (AFP)
        - AFP = UFP x TCF

- Apply Environmental Complexity Factor (ECF)
- Calculate Adjusted Function Points (AFP):
    - Adjusted Function Points = AFP x ECF

Calculation example: e-commerce website

- Define the metric: Completeness of requirements (%)
- For example 50 requirements
- requirements that are covered by test cases -> 40/50
- Calculate the completeness metric:
  - Divide the number of requirements covered by test cases by the total number of requirements and multiply by 100.
  - Completeness = (40/50) x 100 = 80%

Automatic tools



Build Your Software Development Metrics Program in Minutes

The C-suite and your leadership peers care about engineering performance. But they don't want to know about story points, lines of code, or active days. They want to know:

- If your teams are working efficiently
- If what's being worked on moves the needle
- If engineering is predictable (so they can plan their activities)

LinearB correlates data from your Git, project, release, and incident management tools to provide you with a consolidated executive dashboard for software development KPIs, so you get the metrics you need to make your next check-in more productive.

Once you've got a baseline, use LinearB to set goals to improve these software productivity metrics and WorkerB automation to enable your dev teams to reach them.