ESTIMATION OF SOFTWARE

SOFTWARE PROJECT MANAGEMENT

Project planning involves:

- Project scope definition (WBS)
- Software estimation (cost req.)
- Software scheduling (time req.)
  - ✓ Network diagram & Gantt charts

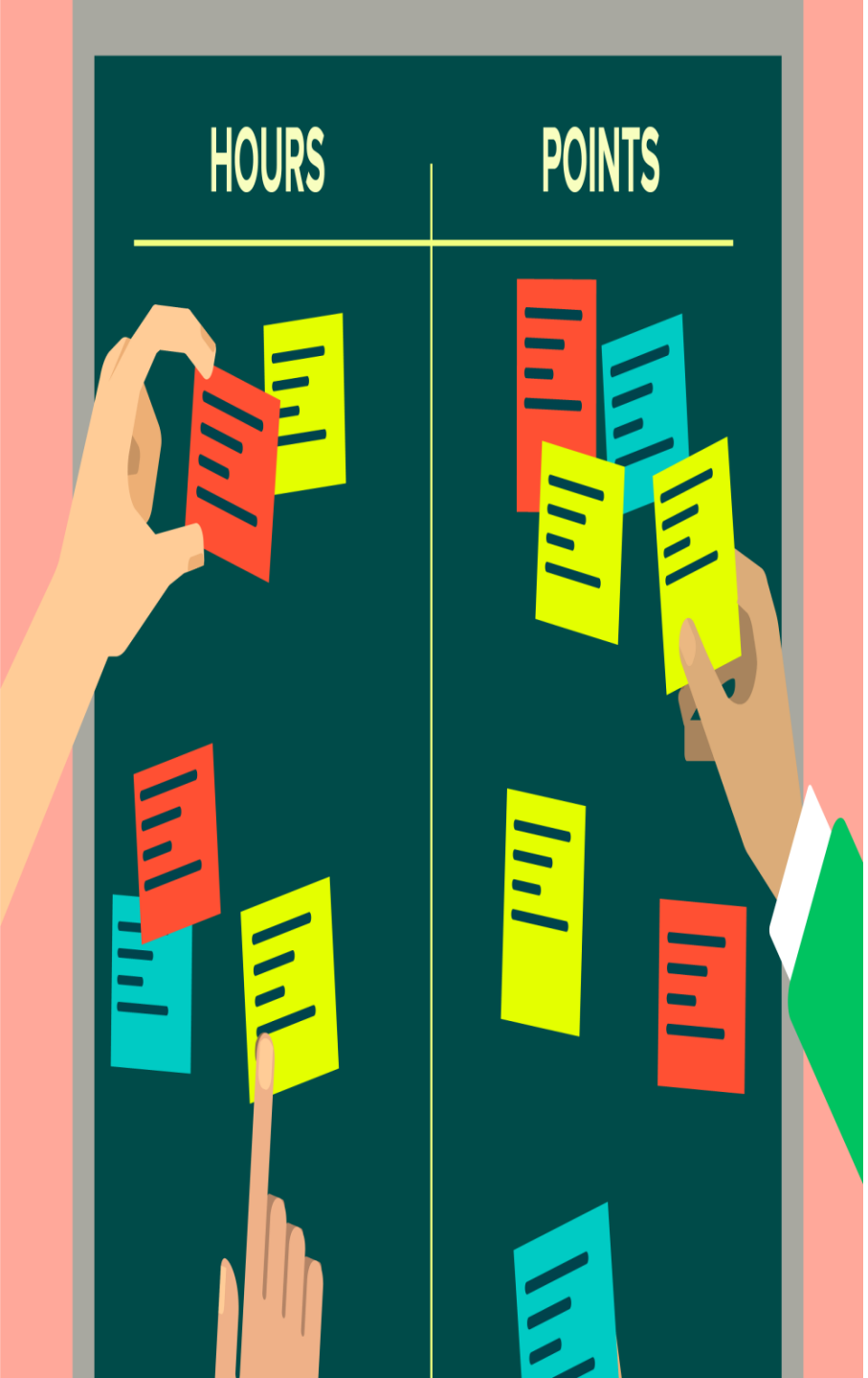
SOFTWARE PROJECT ESTIMATION

Project Estimation is done on the basis of:

- Project size
- Project Cost
- Effort required
- Project duration

METRICS USED FOR PROJECT SIZE ESTIMATION

- Lines of codes (LOC)
- Functional points (FPs)

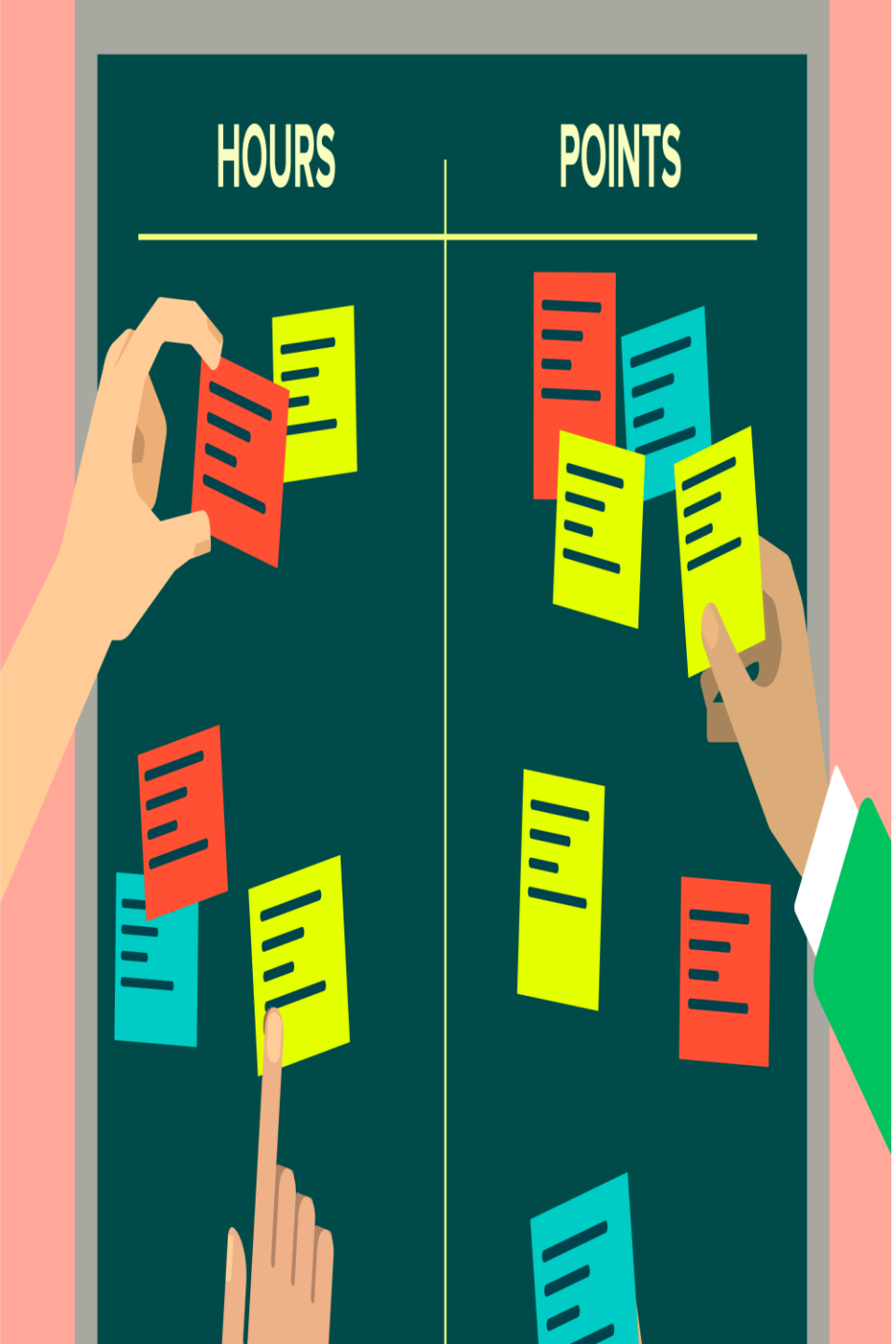


SOFTWARE PROJECT ESTIMATION

LOC and FP data are used in two ways during software project estimation:

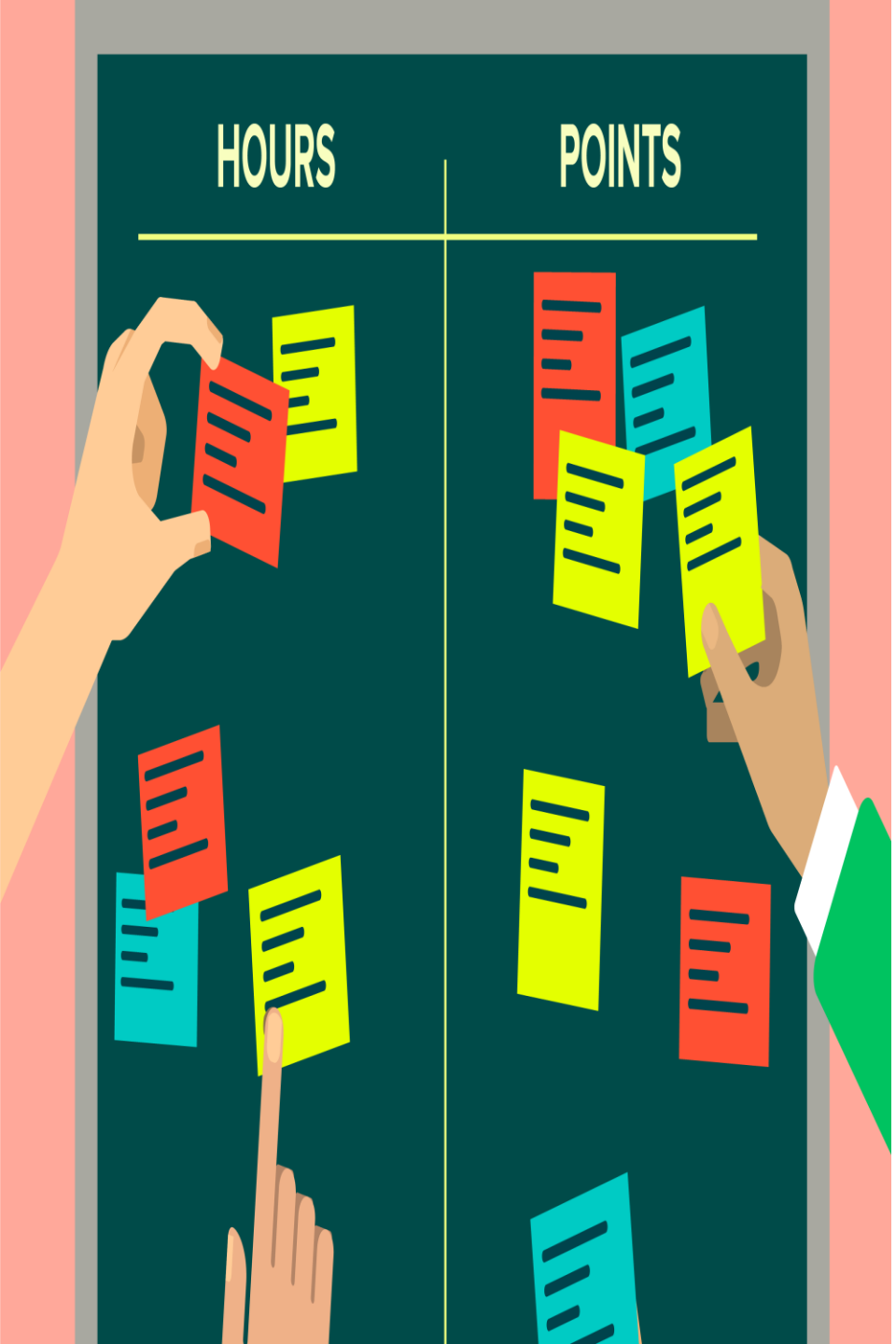
- as estimation variables to "size" each element of the software
- as baseline metrics collected from past projects and used in conjunction with estimation variables to develop cost and effort projections.
- Measured in LOC/pm or LOC/pm[6]

LINES OF CODE ESTIMATION



LOC

- Count every line of source code except blanks (improve code readability) or comments (improves code understandability).
- Count each declaration, actual code containing logic and computations
- Advantages:
  - ✓ Easy to count and calculate
- Disadvantages:
  - ✓ Not a good metric for estimation as its highly dependent on the programming language used.



LOC

- Break projects into features
- Estimate lines of code by analyzing past data. For example, a similar component has been created before and consider its optimistic, most likely and pessimistic values.

$$\text{Expected size} = (S_{\text{optimistic}} + 4 * S_{\text{most likely}} + S_{\text{pessimistic}}) / 6$$

- In upcoming example, the values acquired for the task 3D geometric analysis for past data are :

Optimistic =4600 LOC, most likely = 6900 LOC, pessimistic = 8600 LOC.

So the Estimated LOC will be calculated using above formula :

$$\text{Expected LOC} = \frac{4600+(4*6900)+8600}{6} = 6800 \text{ LOC}$$

LOC BASED ESTIMATION EXAMPLE

- The mechanical CAD software will accept two- and three-dimensional geometric data from an engineer.
- The engineer will interact and control the CAD system through a user interface that will exhibit characteristics of good human/machine interface design.
- All geometric data and other supporting information will be maintained in a CAD database.
- Design analysis modules will be developed to produce the required output, which will be displayed on a variety of graphics devices.
- The software will be designed to control and interact with peripheral devices that include a mouse, digitizer, laser printer, and plotter.

LOC BASED ESTIMATION EXAMPLE

Functions	Estimated LOC
2D geometric analysis	5300
3D geometric analysis	6800
UI and control facilities	2300
Database management	3350
Design analysis modules	8400
Computer graphics display facilities	4950
Peripheral control	2100

LOC BASED ESTIMATION EXAMPLE

- As per historic data, it is deduced that:
  - Organization can write lines of code = LOC = 620 LOC/person
  - Payment to a personnel = Cost= $ 8000/month
  - Cost/LOC = $\frac{8000}{620} = 12.9 \approx \$ 13$
  - Effort required for the project= $\frac{\sum \text{Estimated LOC for each function}}{LOC}$

$$= \frac{5300+6800+2300+3350+8400+4950+2100}{620}$$

$$= \frac{33200}{620} \approx 53 \text{ persons}$$

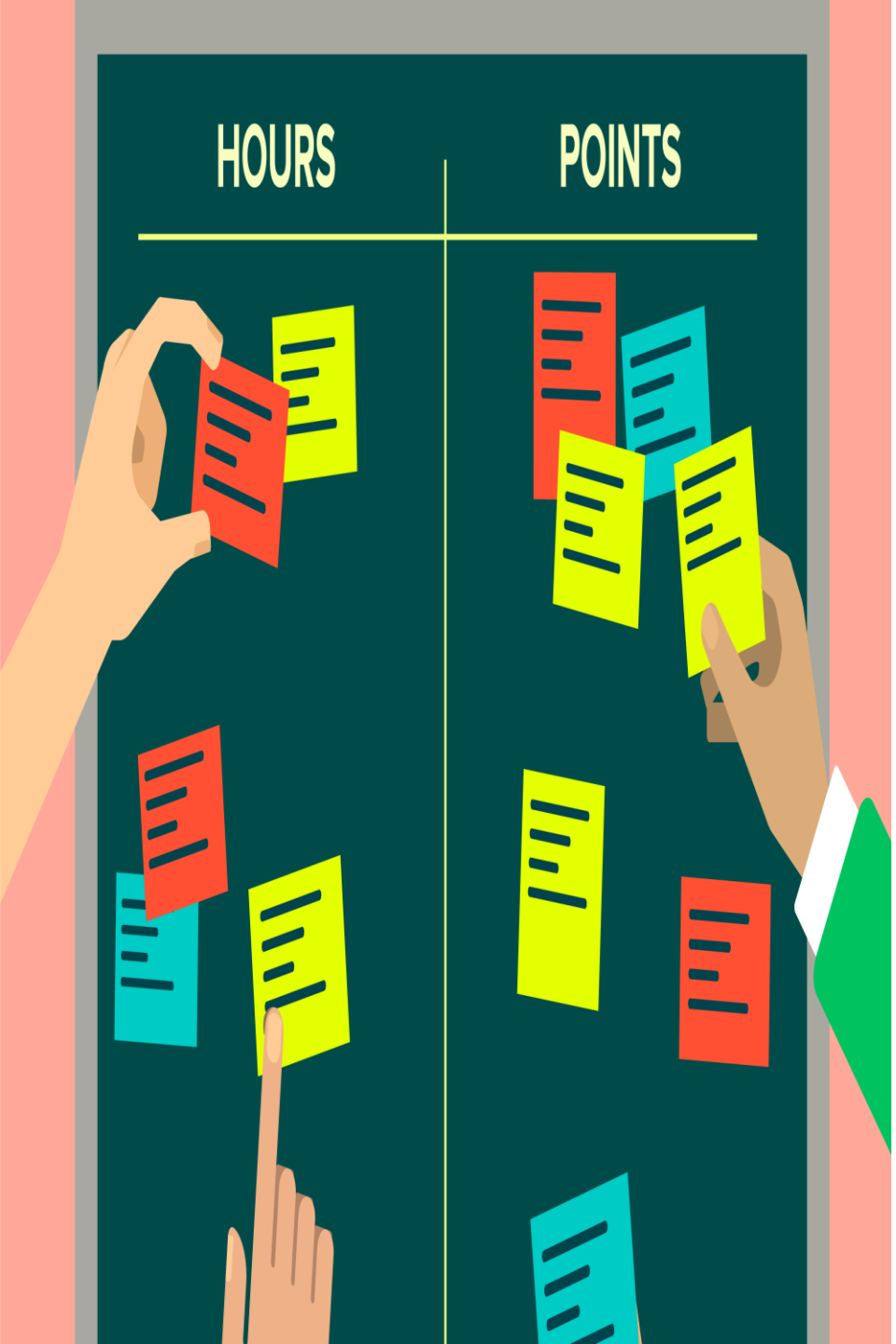
  - Project cost to be paid = $\sum_{\text{Estimated LOC for each function}}$ * Cost/LOC

$$= (5300+6800+2300+3350+8400+4950+2100) * 13$$

$$= \$ 4{,}31{,}600$$

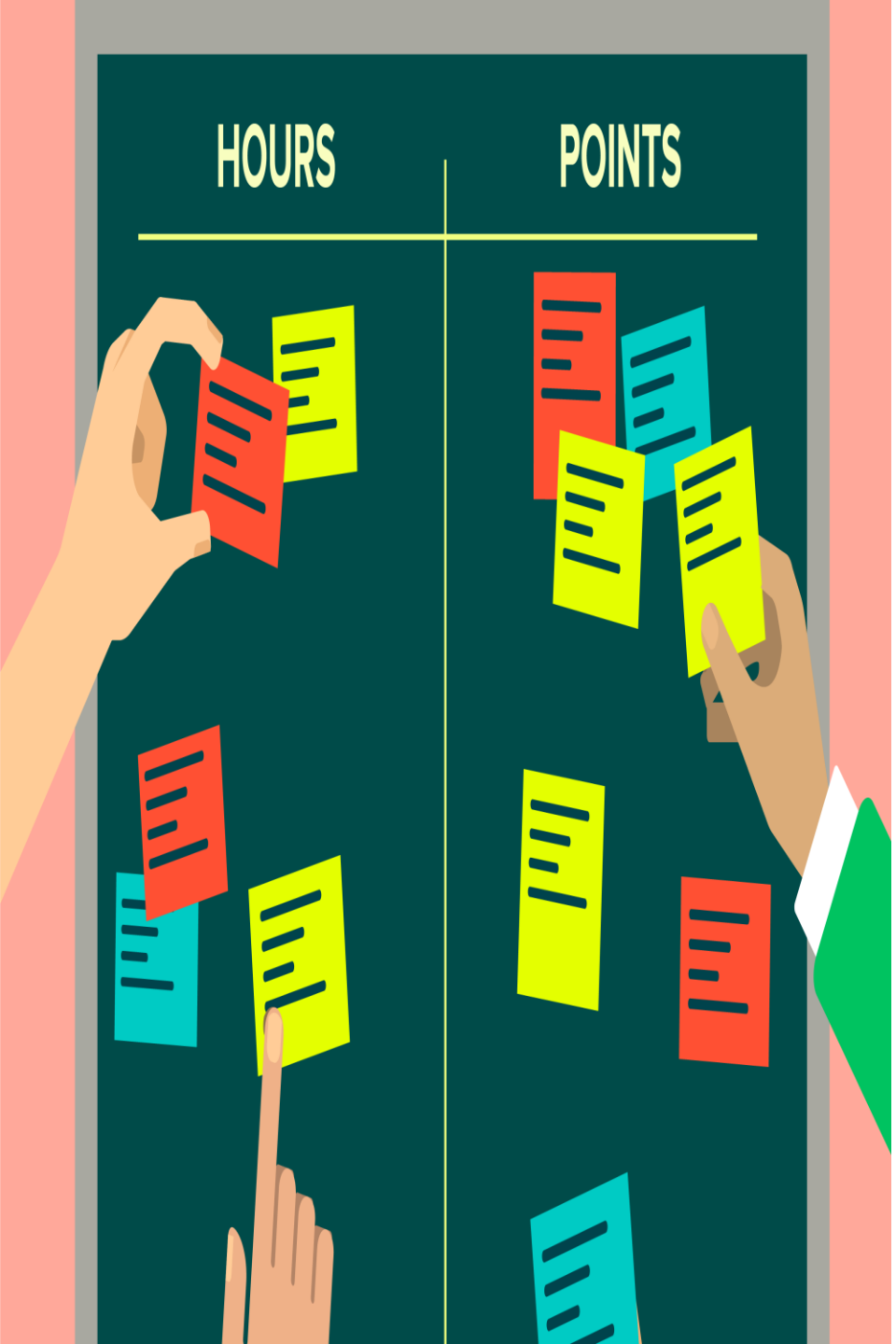
A "person month" is the metric for expressing the effort (amount of time) personnel devote to a specific project. Approximately equals to 140 hours

FUNCTIONAL POINTS ESTIMATION



FP ESTIMATION

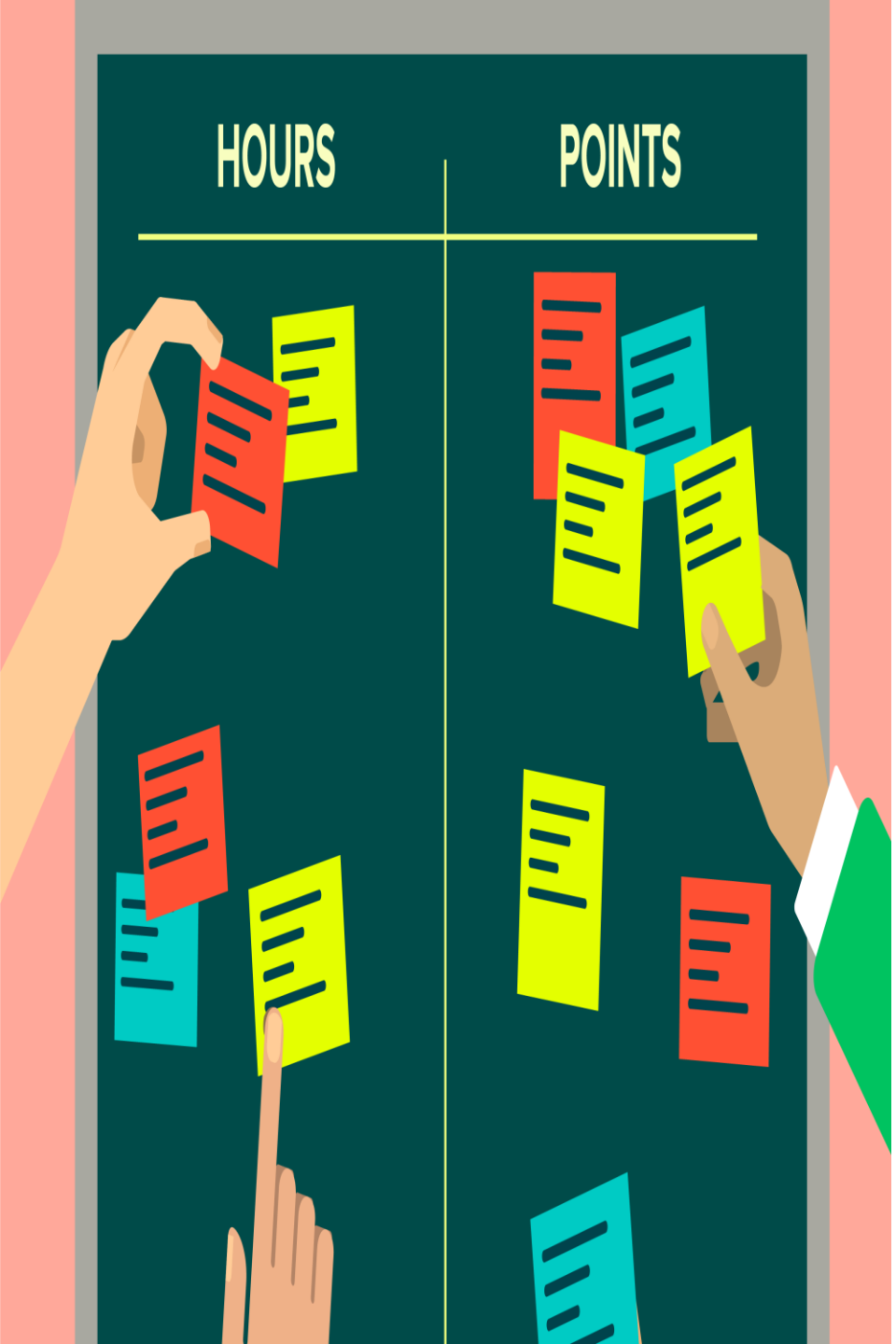
- Not a language dependent estimator
- It measures functionality from a user's point of view. Like what he/she received or entered to the system.
- estimate inputs, outputs, inquiries, files and external
- Here,
  - weight defines the importance and complexity of an operation (LMS would have more I/O operations, phone directory will have more files and searching operations)



IDENTIFY FUNCTIONAL POINTS

Find the number of functions belonging to the following types:

1. External Inputs: Functions related to data entering the system.
2. External outputs: Functions related to data exiting the system.
3. External Inquiries: They leads to data retrieval from system but don't change the system.
4. Internal Files: a user identifiable group of logically related data that resides entirely within the application boundary and is maintained through External Inputs. Log files are not included here.
5. External interface Files: These are logical files for other applications which are used by our system.



FP ESTIMATION

To compute functional points;

$$FP = UFP * CAF$$

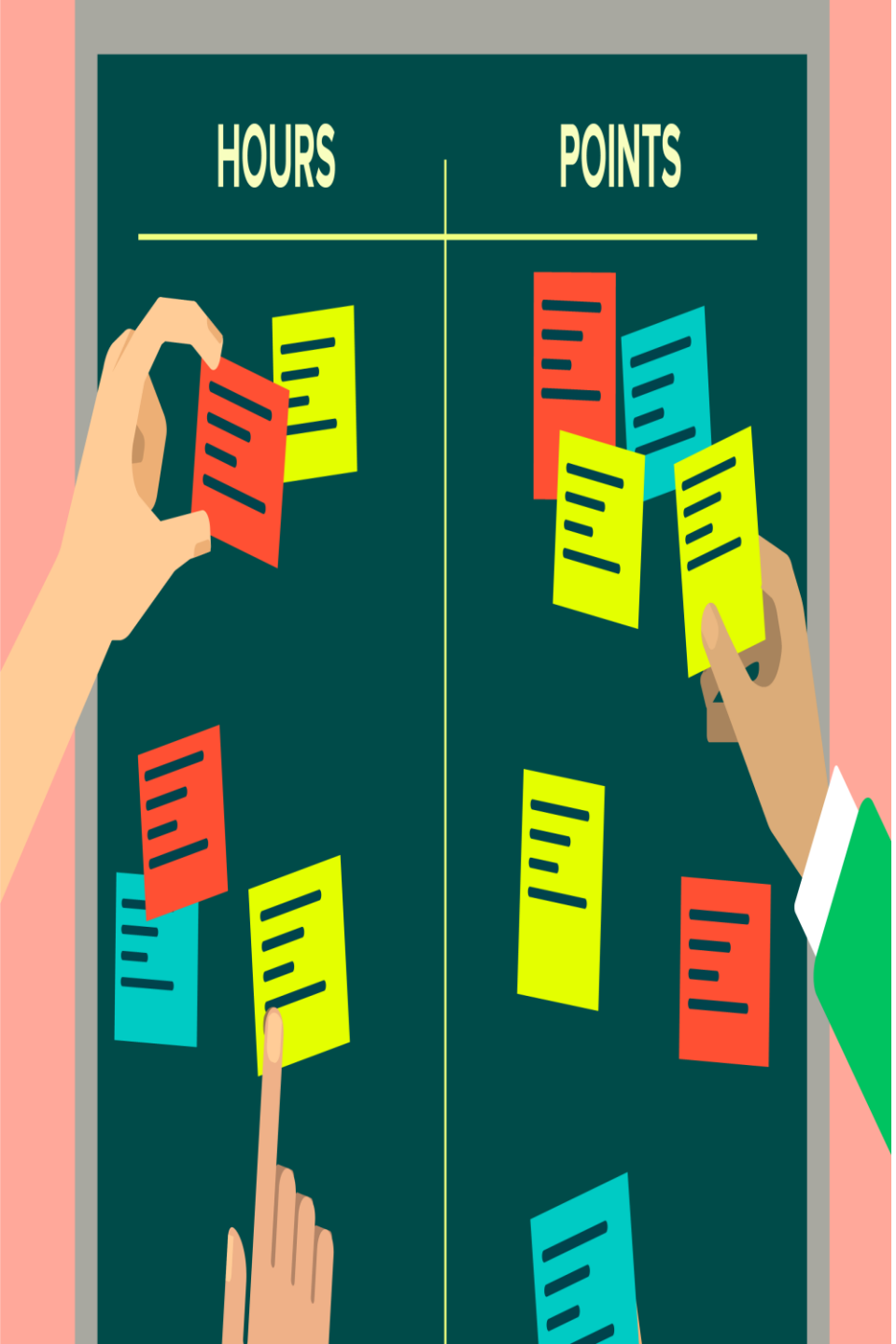
where UFP = unadjusted functional points, CAF = complexity adjustment factor

Now calculate UFP, where each FP is ranked as per complexity. Predefined weights for each category is given in next table.

FP ESTIMATION

Measurement parameter	Counts	Low	Average	High	=count * weighting factor
External inputs		3	4	6	
External outputs		4	5	7	
External enquired		3	4	6	
Internal logic files		7	10	15	
External interface files		5	7	10	
Count total					

Low, average and high are weighting factors. Anyone is selected depending on complexity of the measurement parameter.



FP ESTIMATION

- Calculate CAF

$CAF = 0.65 + (0.01 * \Sigma Fi)$

- Where, Fi is the value adjustment factor depends on the reply to following 14 questions.

FP ESTIMATION

1. Does the system require reliable backup and recovery?
2. Are specialized data communications required to transfer information to or from the application?
3. Are there distributed processing functions?
4. Is performance critical?
5. Will the system run in an existing, heavily utilized operational environment?
6. Does the system require online data entry?
7. Does the online data entry require the input transaction to be built over multiple screens or operations?
8. Are the ILFs updated online?
9. Are the inputs, outputs, files, or inquiries complex?
10. Is the internal processing complex?
11. Is the code designed to be reusable?
12. Are conversion and installation included in the design?
13. Is the system designed for multiple installations in different organizations?
14. Is the application designed to facilitate change and ease of use by the user?

These questions value may vary from 0 to 5, where

0= not important

1= incidental (that values depends on an activity)

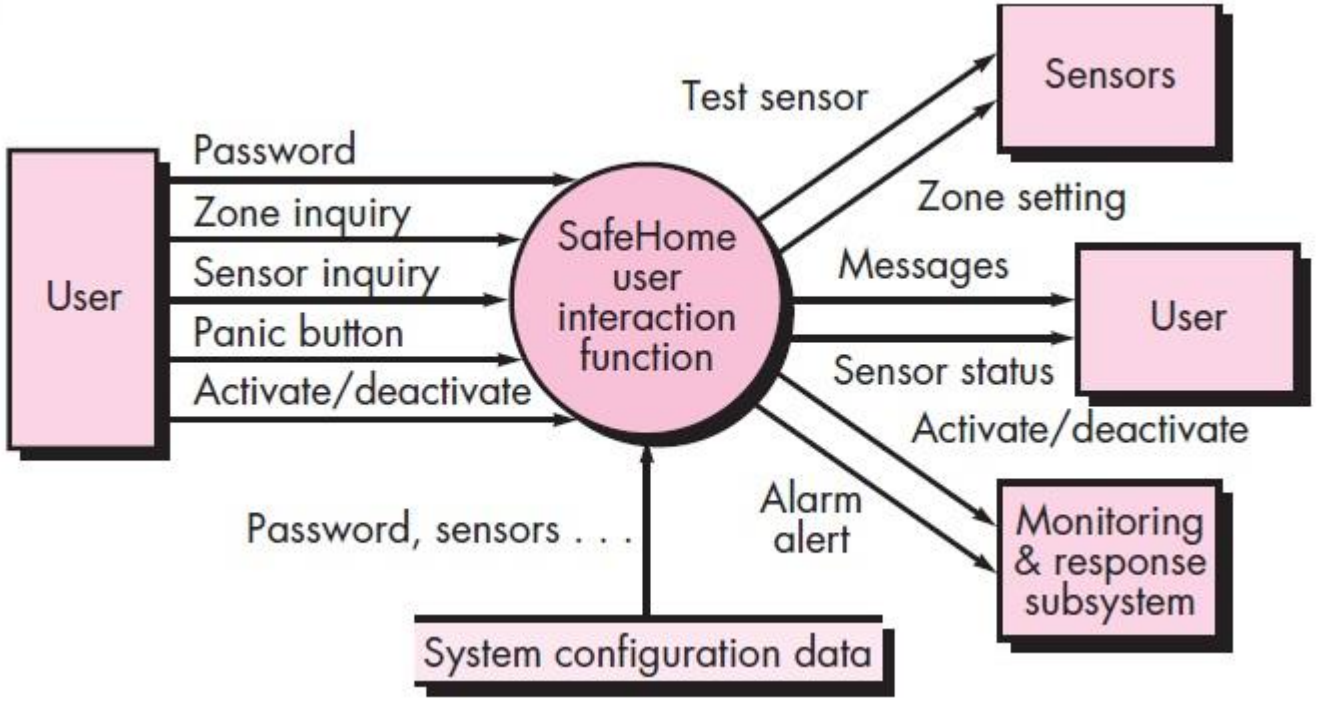
2= moderate

3= average

4= significant

5=essential

FP BASED ESTIMATION EXAMPLE

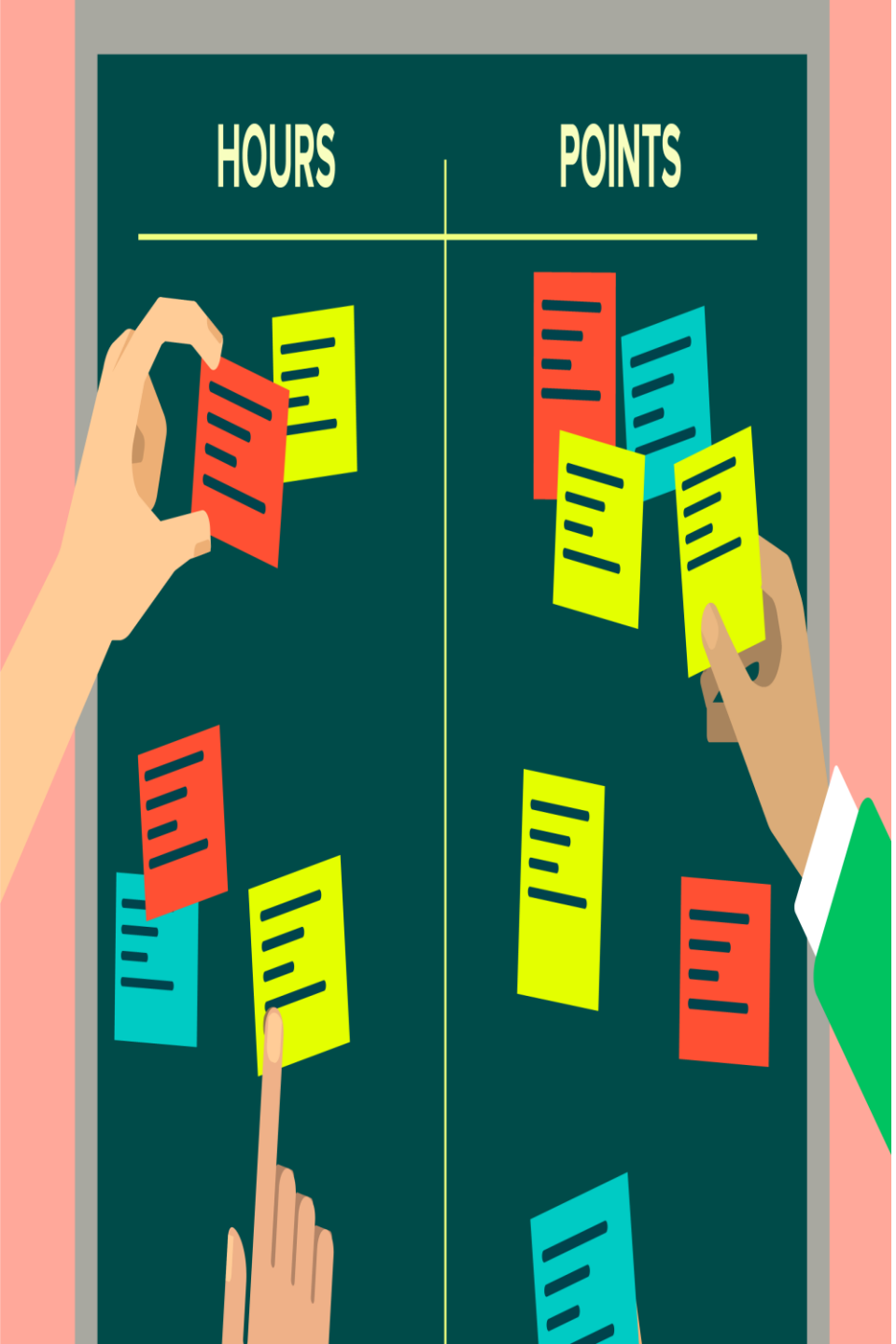


- a set of key information domain measures required for computation of the function point metric.
  - ✓ external inputs: password, panic button, and activate/deactivate.
  - ✓ External outputs: messages and sensor status
  - ✓ external inquiries: zone inquiry and sensor inquiry.
  - ✓ ILF: system configuration file
  - ✓ EIFs: test sensor, zone setting, activate/deactivate, and alarm alert

FP BASED ESTIMATION EXAMPLE

Measurement parameter	Counts	Low	Average	High	=count * weighting factor
External inputs	3	3	4	6	9
External outputs	2	4	5	7	8
External enquired	2	3	4	6	6
Internal logic files	1	7	10	15	7
External interface files	4	5	7	10	20
Count total					50

Calculate for low complexity



FP BASED ESTIMATION EXAMPLE

- Calculate CAF

CAF = 0.65 +(0.01 * **Σ Fi**)

- Lets say, the product is moderately complex product, so all 14 questions answered in moderate form

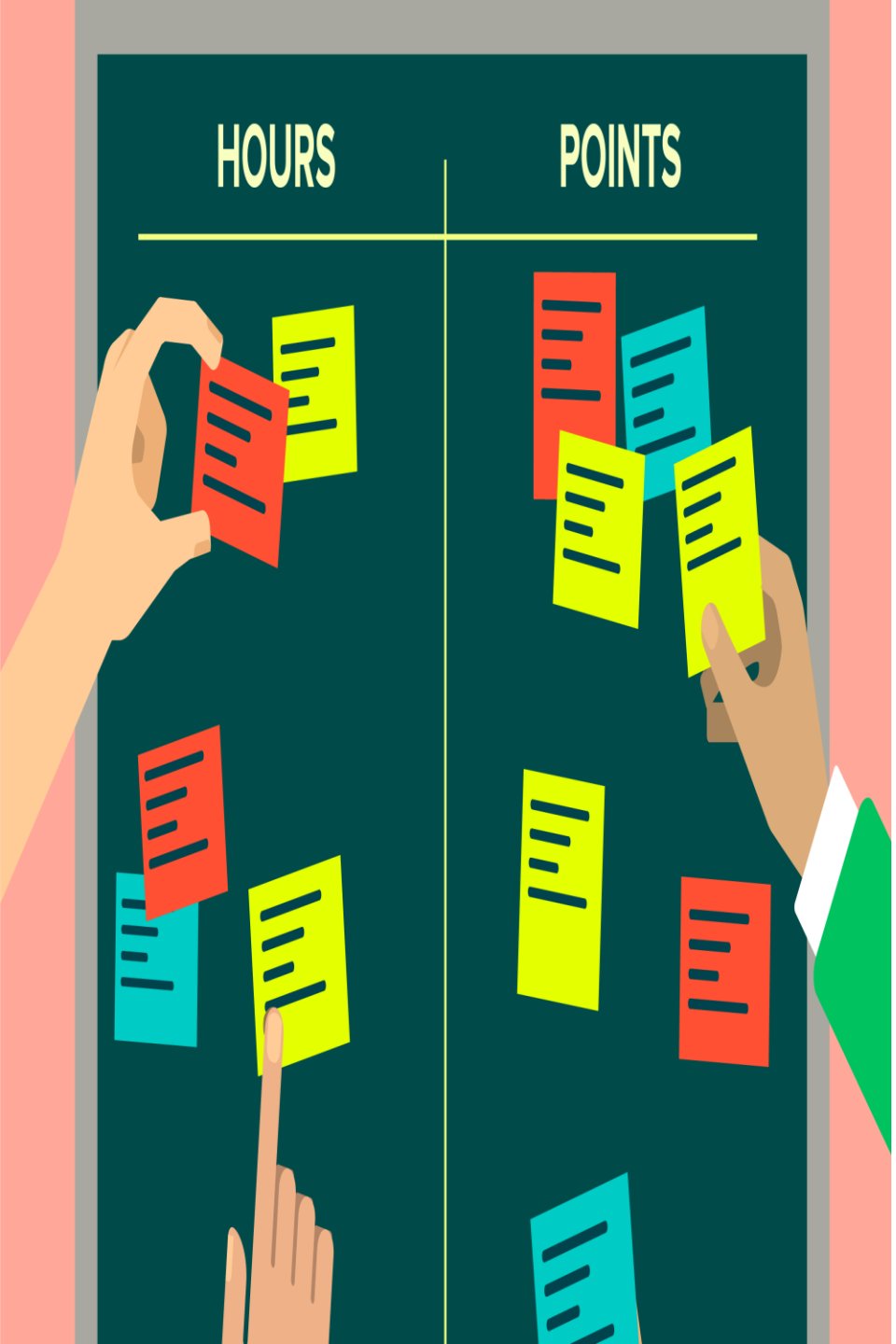
i.e. 2

Σ Fi = 14*2 =28

CAF = 0.65 + (0.01 * 28) =0.93

FP = UFP * CAF

FP = 50 * 0.93 = 46.5



FP BASED ESTIMATION

- If lets say team productivity is 60FP/person-week, then

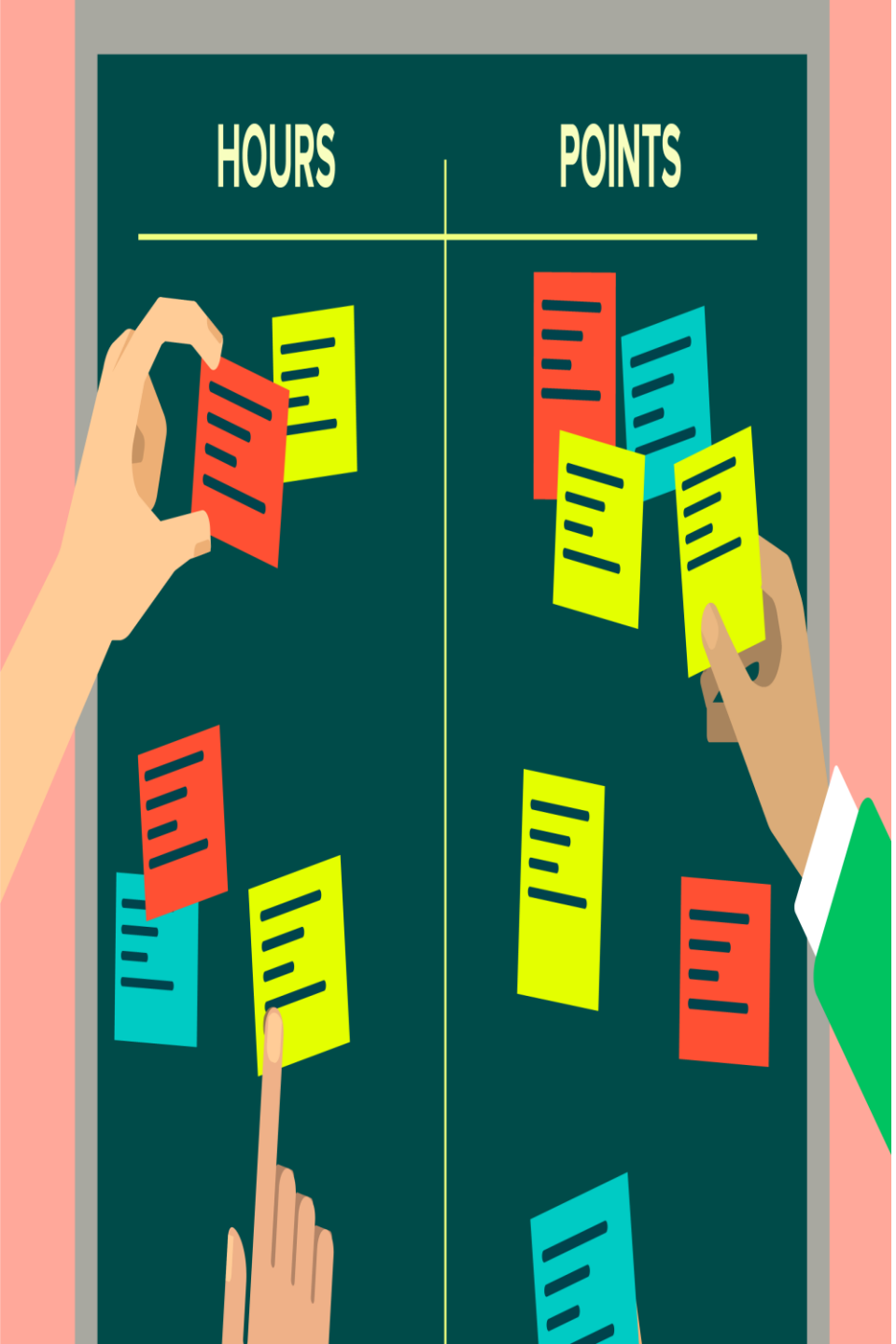
  Effort = FP/productivity

  = FP/60

- If team size =5, then

  Project duration = Effort/ team size

  = Effort/5 weeks



FP BASED ESTIMATION

Assume that past data indicates:

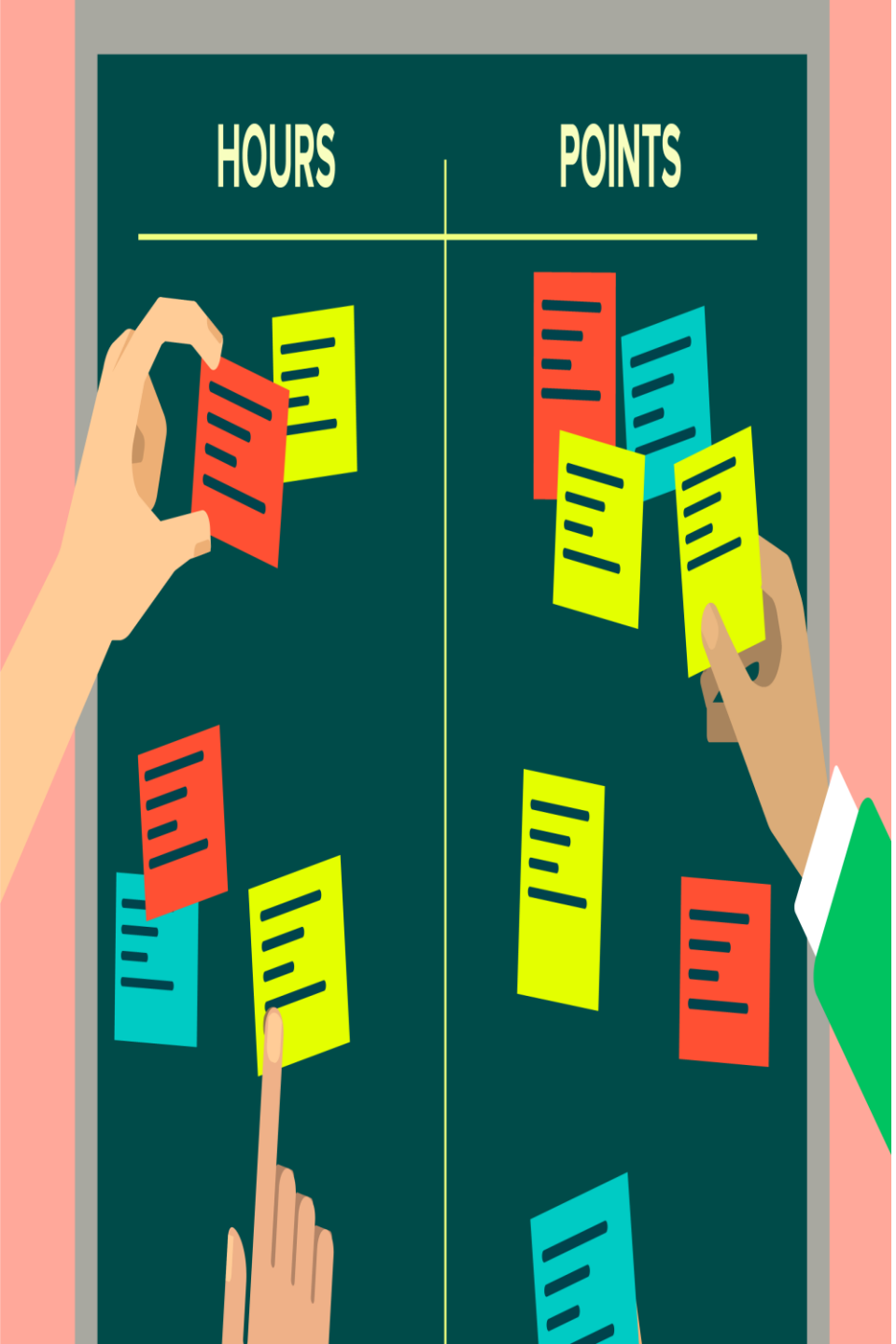
- 1 FP translates into 60 lines of code (an object-oriented language is to be used)
- and that 12 FPs are produced for each person-month of effort.

These historical data provide the project manager with important planning information that is based on the requirements model rather than preliminary estimates.

Assume further that past projects have found:

- an average of three errors / function point during requirements and design reviews
- four errors / function point during unit and integration testing.

These data can ultimately help you assess the completeness of your review and testing activities.



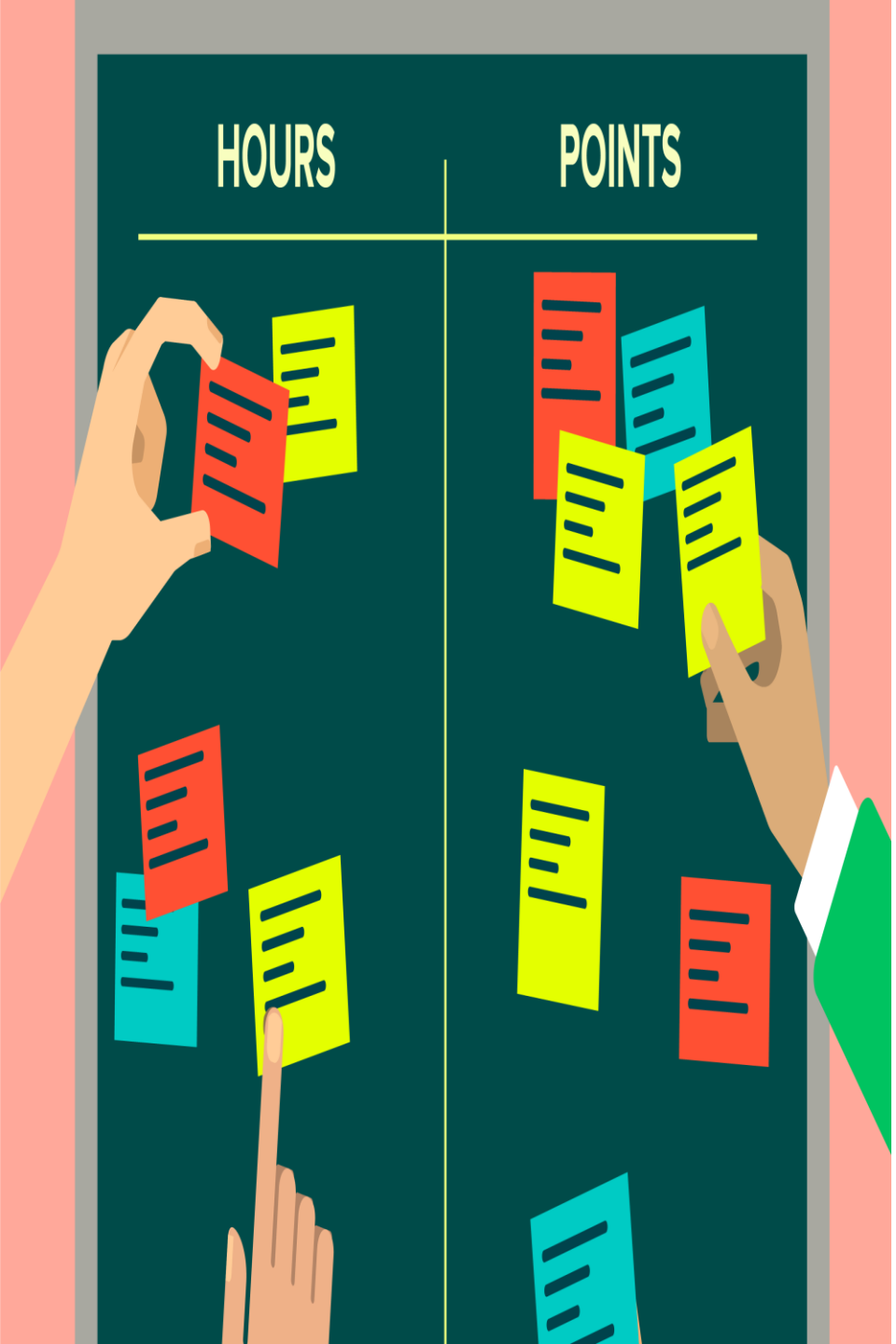
FP BASED ESTIMATION EXAMPLE 2

- User input = 55
- User outputs = 35
- User enquiries = 40
- User files = 8
- External interfaces =5
- Calculate FP = UFP * CF

FP BASED ESTIMATION EXAMPLE 2

Measurement parameter	Counts	Low	Average	High	=count * weighting factor
External inputs	55	3	4	6	?
External outputs	35	4	5	7	?
External enquired	40	3	4	6	?
Internal logic files	8	7	10	15	?
External interface files	5	5	7	10	?
Count total					= ?

Calculate for high complexity



FP BASED ESTIMATION EXAMPLE

- Calculate CAF

CAF = 0.65 +(0.01 * **Σ** Fi)

- Lets say, the product is significantly complex product so all 14 questions answered in moderately form i.e. 4

Σ Fi = ? (if complexity factor is differing for each FP then calculate separately.)

CAF = 0.65 + (0.01 * ?) = ?

FP = UFP * CAF

FP = ? * ? = ?

Complexity factors
5 questions = Average
5 questions = Moderate
4 questions = No influence

$\Sigma Fi = (5 \times 3) + (5 \times 2) + (4 \times 0) = 25$