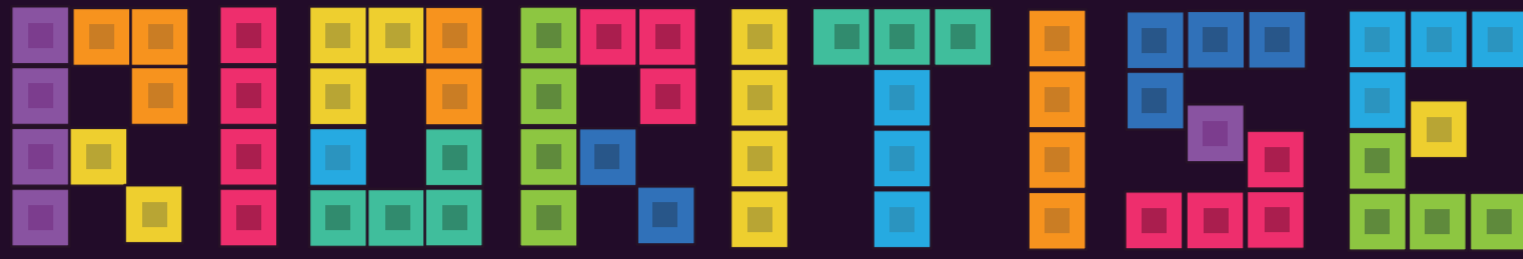


45% OF I.T. PROJECTS SUFFER COST OVERRUNS.

THAT'S WHY IT'S IMPORTANT TO



LIKE YOU MEAN IT!

# CHOOSE YOUR CHALLENGE

ASK YOURSELF, IS IT A

## SMALL / LOW RISK PROJECT

Prioritise by:

### BUSINESS VALUE & TECHNICAL IMPACT

Here you prioritise your task by assigning a qualitative score (high, medium or low) to the business value & technical impact (complexity).

FEATURE #1

ACCEPT CASH PAYMENTS

Business Value  
Medium

Technical Impact  
High

#### PROS

- Simple
- Considers technical & business perspectives

#### CONS

- Not suited for big projects
- Repetitive values
- Hard to coordinate stakeholders
- Ambiguous & limited
- Weight of criteria not clear

OR

### MosCoW

Here you classify the tasks by Must, Should, Could or Won't be performed.

MUST

SHOULD

COULD

WON'T

#### PROS

- Simple
- Forced to "throw the garbage away"

#### CONS

- Division between business value & technical complexity is unclear
- Non-M tasks hardly done
- Weight between "M" tasks unclear

OR

### \$100 METHOD

This is where each team member gets a budget of \$100 & decides which features they want to spend it on.

	EFFORT	COST	INVEST	PRIORITY
FEATURE	5 SP	5\$	25\$	5
FEATURE	8 SP	8\$	48\$	6
FEATURE	3 SP	3\$	12\$	4

#### PROS

- Simple
- Forced to "throw the garbage away"

#### CONS

- Hard to measure monetary value of intangible assets
- People could negotiate their money & create alliances
- Risk of someone allocating all the money into a specific task

## KEEP IN MIND

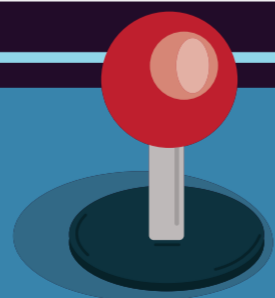
Don't prioritise by:

#### BUSINESS VALUE ONLY

- May block valuable features
- Can create unnecessary technical debt

#### YOUR GUTS

- Removes the objective mathematical layer
- Creates biased results



## BIG / HIGH RISK PROJECT

Prioritise by:

### VALUE, COST & RISK

(Wieggers & Betty Method)

Here you prioritise based on the:

- Relative Business Value** that a feature, user story or functional requirement will provide
- Relative Cost or Effort** of implementing a specific feature
- Risk of Probability** of not getting the feature right the first time

$$\text{Value} = (W_B \times \text{RelativeBenefit}) + (W_P \times \text{RelativePenalty})$$

VALUE FORMULA

$$\text{Value \%} = \frac{\text{value}}{\sum_{i=0}^n \text{value}_i}$$

Where  $W_B$  = Relative benefit weight,  $W_P$  = Relative penalty weight,  $n$  = amount of features

COST FORMULA

$$\text{Cost \%} = \frac{\text{cost}}{\sum_{i=0}^n \text{cost}_i}$$

Where  $n$  = amount of features

RISK FORMULA

$$\text{Risk \%} = \frac{\text{risk}}{\sum_{i=0}^n \text{risk}_i}$$

Where  $n$  = amount of features

Here's how it all comes together:

PRIORITY FORMULA

$$\text{Value \%} = (W_C \times \text{Cost \%}) + (W_R \times \text{Risk \%})$$

Where  $W_C$  = Cost weight,  $W_R$  = Risk weight

#### PROS

- Reduces ambiguity & subjectivity
- Provides an accurate priority
- Extensible & can be adapted to our needs & constraints

#### CONS

- Simple, but could be an overkill in small projects or quick prioritisation
- In agile environments, it's difficult to bring in all perspectives from business stakeholders

## TIPS & TRICKS

Group by:

- Capabilities & dependencies
- Stakeholders
- Define multiple spreadsheets for different granularity & abstraction levels
- Always review the final results

