REQUIREMENTS PRIORITIZATION

BY: DR. YOUSRA ODEH

AGENDA

- Prioritization
- Why Prioritization?
- Some prioritization techniques
  - **In or out**
  - **Pairwise comparison and rank ordering**
  - **Three-level scale (high- medium- low)**
  - **MoSCoW**
  - **$100**

WHY PRIORITIZATION?

1. When customer expectations are **high** and timelines are **short,** you need to make sure the product delivers the **most critical or valuable functionality as early as possible**
2. Prioritization is a way to deal with **competing demands** for **limited resources**
3. Establishing priority of each product capability lets you **plan construction** to help you provide the maximum business value as quickly as possible at the lowest cost.

- Prioritization is the **customer's responsibility** towards BA and the development team. Remember?
- Some functionality must have high priority because it is required to meet **regulatory demands** for the application

Q1: When can you start prioritising requirements ?

Q2: Why customers don’t like prioritising requirements ?

Q3: If PM didn't’t prioritise requirements, who does then ?

SOME PRIORITIZATION TECHNIQUES

- First, we need to agree that the **highest priority requirements** are those that provide the largest fraction of the total product value at the smallest fraction of the total cost
  - **In or out**
  - **Pairwise comparison and rank ordering**
  - **Three-level scale (high- medium- low)**
  - **MoSCoW**
  - **$100**

© Can Stock Photo

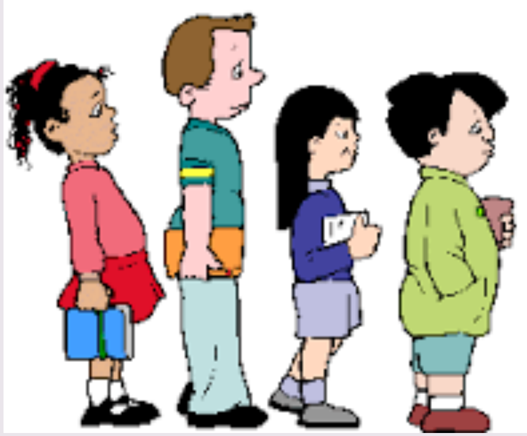
IN .. OUT.. (THE SIMPLEST)

- The simplest of all prioritization methods.
- How does it work ?
  - a group of stakeholders work down a list of requirements and make a binary decision: is it in, or is it out? The binary decision is in relation to business objectives.
  - They keep filtering until having the minimum needed highest priority requirements
  - Once the **in-set** requirements implemented, the stakeholder repeat the process for the remained requirements in the **out-set.**

© 2009 mark du toit.
www.marktoon.co.uk

PAIRWISE COMPARISON AND RANK ORDERING

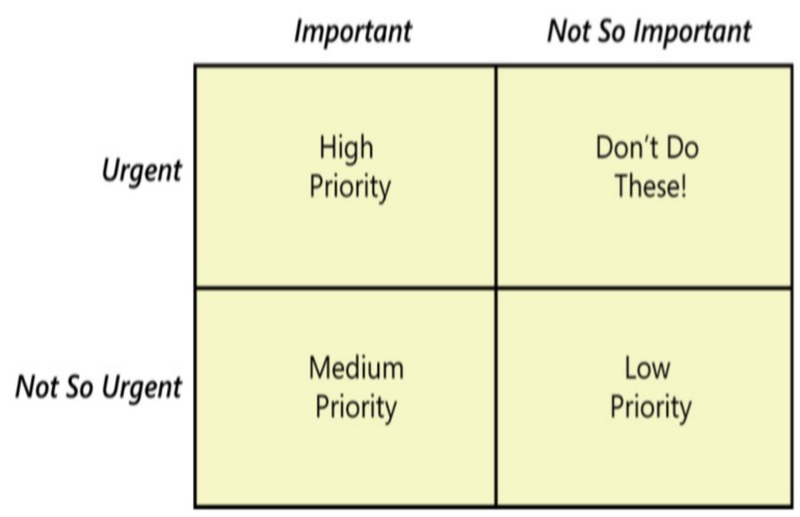
- Assign a unique priority sequence number to each requirement.
- Rank ordering a list of requirements involves making pairwise comparisons between all of them so, you can judge which member of each pair has higher priority
- Comparisons becomes difficult for more than a couple of dozen requirements. (size)
- Grouping requirements into small sets of requirements that have similar priority or that otherwise must be implemented together, is sufficient.



THREE-LEVEL SCALE (HIGH- MEDIUM- LOW)

- Grouping requirements into three categories
- To make the scale useful, the stakeholders must agree on what each level means in the scale they use.
- One way to assess priority is to consider the two dimensions of ***importance*** and ***urgency***

  Every requirement can be considered as being either important to achieving business objectives or not so important, and as being either urgent or not so urgent
- This derives four possible combinations:
- *High-priority* requirements are both important (customers need the capability) and urgent (customers need it in the next release).
- *Medium-priority* requirements are important (customers need the capability) but not urgent (they can wait for a later release).
- *Low-priority* requirements are neither important (customers can live without the capability if necessary) nor urgent (customers can wait, perhaps forever).



	Important	*Not So Important*
Urgent	High Priority	Don't Do These!
Not So Urgent	Medium Priority	Low Priority

FIGURE 16-1 Requirements prioritization based on importance and urgency.

MOSCOW

- The MoSCoW scheme changes the three-level scale of high, medium, and low into a four-level scale.
- The four capitalized letters in the MoSCoW prioritization scheme stand for four possible priority requirement classification:
  - **Must:** The requirement must be satisfied for the solution to be considered a success.
  - **Should:** The requirement is important and should be included in the solution if possible, but it's not mandatory to success.
  - **Could:** It's a desirable capability, but one that could be eliminated. Implement it only if time and resources permit.
  - **Won't:** This indicates a requirement that will not be implemented at this time but could be included in a future release.
- MoSCoW is ambiguous as to timing, particularly when it comes to the "Won't" rating
- We don't recommend MoSCoW.

$100

- Recalling: Prioritization is about thoughtfully allocating limited resources to achieve the maximum benefit from the investment an organization makes in a project.
- Make prioritization more tangible by using money. Just play money.
- The approach works a s follows:
  1. Give the prioritization team 100 imaginary dollars to work with. Team members allocate these dollars to "buy" items that they would like to have implemented from the complete set of requirements.
  2. They weight the higher-priority requirements more heavily by allocating more dollars to them.
  3. Keep weighting requirements by dollars until out of money
  4. Add up the total number of dollars assigned to each requirement to see which ones collectively come out as having the highest priority.
- Another approach is not imaginary money, but real money. As seen in contracts.

QUOTE OF THE WEEK

It is our choices, that show what we truly are, far more than our abilities.
– Joanne Rowling
(Author of Harry Potter novels)

REFERENCES

- This presentation material has been prepared from the references below:
- Karl Wiegers and Joy Betty, Software Requirements, 3rd Edition, Microsoft. **Chapter 16**