Backlog Prioritization Techniques

Common Agile Approaches to Prioritization of User Stories or Epics

Tom Taylor, Scrum Master & Pega Agilist

Backlog Prioritization

Scrum does not prescribe a specific method for prioritization

- But the PO is responsible for the prioritization of the Product Backlog

- In Scrum, a Sprint Backlog does not need to be prioritized as the Development Team commits to delivering all of the User Stories. However, it is common for teams to maintain prioritization from the Product Backlog

Prioritization techniques covered include:

- MoSCoW

- 20/20

- Risk-Value

- Weighted Shortest Job First (WSJF)

PEGA

MoSCoW Technique

Approach

- Priorities are based on:
  - **M**ust Have, **S**hould Have, **C**ould Have, **W**on't Have
  - Replaces High/Medium/Low (since everything ends up High)
- Must Haves and possibly Should Haves help determine "Minimum Viable Product" (MVP)

Guidelines

- Option 1: Use group consensus to decide M, S, C, W priorities
- Option 2: Assign a point value to M, S, C, W. M should be highest, W should be lowest
  - e.g.: M = 10, S = 7, C = 3, W = 0
- For every feature (User Story or Epic), the stakeholders assign individual points
- Points are totaled for each feature and prioritization is based on the highest scored items

MoSCoW – Point Value Example

	Stakeholder 1	Stakeholder 2	Stakeholder 3	Stakeholder 4	Stakeholder 5	Total
Feature 1	7	10	7	3	7	34
Feature 2	10	10	10	3	10	43
Feature 3	10	10	10	0	7	37
Feature 4	10	10	7	3	3	34
Feature 5	7	10	7	0	7	31
Feature 6	3	10	0	10	0	23
Feature 7	7	10	10	3	7	37

Scale

M	10
S	7
C	3
W	0

While it's easy to see that Feature 2 is the top-ranked item and Feature 6 is the lowest, note that:

- Stakeholder 2 sees all features as equally important

- Stakeholder 4 cares exclusively about the least prioritized feature

- Features 3 & 7 are tied at 37 points – additional discussion needed. Same for 1 and 4

20/20 Technique

Approach

- A side-by-side assessment of 2 features at a time

- Like an eye exam asking "which is clearer - A or B?", determine if A or B is the higher priority

- Also known as Force-Ranking or "Bubble Sorting"

Guidelines

- A quick, effective tool that can be used in teams, with internal stakeholders, or even customers

- Always start at the top of the list, only comparing 2 items at a time

- Forces collaboration – requires good communication and negotiation
  - A facilitator can be useful

20/20 - Example

To be prioritized | **In Comparison** | **Prioritized**

Feature 2

Feature 3 | Feature 1

Feature 4

Feature 5

Feature 6

Feature 7

- Feature 2 is (so far) the top-most item
- Feature 1 is already lower priority than Feature 2
- Feature 3 is also lower than Feature 2, and is being compared to Feature 1
- After Feature 3 is ranked, Feature 4 starts against Feature 2 and continues prioritization continues

Risk / Value Technique

Approach

- Each feature is evaluated on its potential value, along with the perceived risk associated with delivery

- In Scrum, attempt to start with high risk, high value first

- Or, alternatively consider delivering high value, low risk items for "quick wins"

- High risk, but low value items should be tabled or not considered part of MVP

Common Types of Project Risk

- Schedule Risk: "We may not be able to complete this feature in time"

- Cost Risk: "This feature may cost more than we expect"

- Functionality (Business) Risk: "This workflow may be all wrong"

- Technical Risk: "This may not perform at the level we need it to"

Risk/Value - Matrix

Note that MoSCoW Priorization can be applied to the Risk/Value approache as well.

Risk

High

High Risk	High Risk
Low Value	**High Value**
• Won't-Have	• Must-Have

?

1

Low

Low Risk	Low Risk
Low Value	**High Value**
• Should-Have	• Must-Have
• Could-Have	• Should-Have

3

2

Low **Value** High

Adopted from: **Agile Estimating & Planning** by Mike Cohn

Weighted Shortest Job First (WSJF)

Approach

- Applies data-driven prioritization based on Cost of Delay and Job Size

Cost of Delay looks at "What's the pain of not having a feature available?" based on 3 considerations of

- Business Value: "What does this feature offer our 'customers' (end-users or internal)?"

- Time Criticality : "How urgently do we need this for use?"

- Risk-Reduction or Opportunity Enablement (RROE): "Is this a factor by which we boost delivery?"
  – Competitive advantage, Speed, Quality, etc.

Job Size, or relative complexity, is the fourth factor

- Job Size is the proxy for how long delivery is expected to take

WSJF – Category Factors

Business Value

- Revenue Growth
- Business Strategy
- Cost-Savings
- Customer Satisfaction

Time Criticality

- Financial Targets
- Conferences
  - PegaWorld, SKO
- Integration
- Acquisitions
- System Retirement
- Analyst Assessments

Risk Reduction/ Opportunity Enabler

- Market Penetration
- Speed, Quality, Delivery
- Architectural/Infrastructural Investment
- Scalability
- Exploration
- Partnerships

WSJF - Formula

WSJF is the Cost of Delay divided by Job Size

$$\frac{\text{Cost of Delay}}{\text{Job Size}}$$
$$\text{(Business Value + Time Criticality + RROE)}$$

- Each feature is assessed on a relative scale by the categories

- Start with the least-ranked feature and give it the lowest number

- Progress upwards through a single category before moving to the next one

- For Job Size, consult with team representatives

WSJF – Ranking Example

Feature	A) Business Value	B) Time Criticality	C) RROE	D) Cost of Delay (A+B+C)	E) Job Size	WSJF Score (D/E)
1 DevOps Investment • New infrastructure • Boosts internal productivity • Extensive training needed	1	2	2	5	2	2.5
2 Mobile Application • New technology – adds risk! • Increase in mobile use • Needed for competitive advantage	2	3	3	9	5 + 2 points for risk	1.6
3 New Quickstart • Refactoring existing functionality • Must-have for conference in 6 months	3	1	1	5	1	5

WSJF – Ranking Example continued

1. Starting with Business Value, the least important item is ranked as a 1. The next important feature is given a 2, and so on

2. Comparative ranking proceeds to Time Criticality, working again from low to high

3. Ranking continues to Risk Reduction/Opportunity Enablement

4. Cost of Delay is the total of columns A, B and C

5. Job Size, like the other factors, is ranked comparatively – with team representation
   – Risk adds +2 points for Mobile Application

6. WSJF Rank is Cost of Delay is divided by the Job Size

Feature	A) Business Value	B) Time Criticality	C) RROE	D) Cost of Delay (A+B+C)	E) Job Size	WSJF Score (D/E)
New Quickstart	3	1	1	5	1	5
DevOps Investment	1	2	2	5	2	2.5
Mobile Application	3	3	3	9	5	1.6

Per the example, the New Quickstart should be the highest prioritized item, followed by DevOps. If Mobile Application is seen as a priority, then explore ways to mitigate the risks, lowering Job Size

WSJF – Do's and Don'ts

Do's

- Increase the weighting for uncertainty
- Keep focus on a single category at a time
- Involve representation from the team for Job Size estimate
- Retrospect on the rankings after delivery

Don'ts

- × Assume actuals (# stories, point totals) for Job Size
- × Think of Job Size in time units, think of complexity instead
- × Be inflexible about the features. Like User Stories, ask if they can be split

Final Prioritization Tips

As a Product Owner, technical User Stories and Spikes will need to be prioritized, along with End-User features.

Be careful to avoid these sometimes used prioritization "techniques":

- Highest Paid Person in the Room

- Squeaky Wheel

- The Hot Potato (the hot topic of the week…or day…or hour…or minute…)