Quality Rating
About me

Timofey Surguchenko

2003 - 2005 – QA Engineer, SWsoft
2005-2010 – QA Manager, Parallels
Desktop for Mac, Parallels
2010 - now – Head of QA, Parallels

80 people in my department
Built QA team and process from the scratch
Parallels Desktop for Mac – main product in portfolio with 3M+ users worldwide
Agenda

• Quality Rating
• What is this
• How we use it
• Benefits
• Possible issues to avoid
The problem

• 5 years ago...
• Completely rewritten product
  • 100s of features
  • 1000s of bugs
  • 90+ team
  • 8 months
• Tight competition with VMware

• Where are we now!?
What is Quality Rating?

Simple numeric metric
Set by testers
Shows some aspects of product or its components quality state

Req.  →  Dev.  →  Test  →  Release

QR is not enough
QR is enough

Simple numeric metric to assess product quality state
Quality Rating as defined by Parallels

5: Clearly ready to ship
No problems at all.

4: Can be shipped into production, but there are several non-critical issues

3: Can NOT be shipped into production, but is OK for evaluation

2: Not OK even for evaluation, blo/cri issues on main paths

1: Totally unusable/untestable

QR – very simple assessment on component readiness to ship
Main benefit: focus on end-user experience

No:
- bugs
- functionality
- requirements

Yes:
- end-user comfort
- simplicity, integrity, usability
- readiness to ship

QR helps team better focus on “end-user”
Simple birds-eye overview on product components

With QR, general state becomes obvious
## Closer look

<table>
<thead>
<tr>
<th>Feature</th>
<th>Priority</th>
<th>Status</th>
<th>1st QR</th>
<th>QR</th>
</tr>
</thead>
<tbody>
<tr>
<td>Critical</td>
<td>in QA</td>
<td>3</td>
<td>5</td>
<td></td>
</tr>
<tr>
<td>Critical</td>
<td>in QA</td>
<td>3</td>
<td><strong>2 (was 3)</strong></td>
<td></td>
</tr>
<tr>
<td>Major</td>
<td>in QA</td>
<td>3</td>
<td><strong>3 (was 5)</strong></td>
<td></td>
</tr>
<tr>
<td>Major</td>
<td>in dev</td>
<td>3</td>
<td>NA</td>
<td></td>
</tr>
<tr>
<td>Major</td>
<td>in dev</td>
<td>3</td>
<td>NA</td>
<td></td>
</tr>
<tr>
<td>Major</td>
<td>in QA</td>
<td>4</td>
<td>5</td>
<td></td>
</tr>
</tbody>
</table>

Closer look and dynamics available
Simplified communication on all levels

How is this feature quality?

Feature has 1 critical and 15 major issues. There are visual performance issues on certain screens. Clicking Next at “Checkout” screen sometimes produces fatal error, reason is yet unknown. Functional testing must be redone after recent core component change. Integration testing is not yet completed. Also...

QR = 2

Speaking QRs is much shorter and simpler
Sometimes devs refuse to pass features to QA

I want to run more tests!
I want to check more configurations!
I want my feature work great!

Developers are not that effective testers 😊

Don’t hesitate to force passing feature to QA
Quality Rating as a part of team KPI

QR is:
subjective
oriented towards feature, not person
“live”

So, does not work as KPI

Don’t do this if you want piece and collaboration in your team!
Quality Rating is subjective metric

Varies from person to person

Requires
  - good management and proper tester selection
  - trust between team members

QR must be properly reasoned with:
  - bugs
  - blocked tests

Quality rating must be reasoned with properly registered bugs
Summary

1. QR – shipment readiness assessment

2. Benefits:
   a. (!) focus devs and testers on end-user experience
   b. quickly get overview of project quality state
   c. simplify communication

3. Possible issues:
   a. developers may start overdoing features
   b. can not be part of team KPIs

4. Remember:
   a. QR is subjective metric
   b. should be well-reasoned
Thank you

Спасибо

mailto: timsus@gmail.com

twitter: @sstims