IMPROVING QUALITY THROUGH PEER REVIEW

Krasimir Baylov,
Senior Software Engineer,
Musala Soft
AGENDA

• Quality
• Verification
• Peer Reviews
• Managers’ Role
• Questions
QUALITY VIEWS

Manager

Software Developer

QUALITY
QUALITY

Cost of Conformance

+ 

Cost of Non-conformance

= 

Cost of Quality
VERIFICATION

Why is it so often ignored?

Cost of a single defect

- Requirements
- Design
- Implementation
- Dev Testing
- Acceptance Testing
- Maintenance
Peer Reviews are industry best practice

“The review of work products performed by peers during development of the work products to identify defects for removal.”
PURPOSE AND BENEFITS

- Verify the work product meets requirements
- Identify defects for removal
- Reveal assumptions
- Reduce risk
PEER REVIEW FORMS

- Inspection
- Team Review
- Walkthrough
- Pair Programming
- Peer Deskcheck
- Passaround
- ...

Most scientists regarded the new streamlined peer-review process as ‘quite an improvement.’
A real-world case study

Before Code Review

After Development
After QA test ($200/fix)
After Customer ($1000/fix)

A real-world case study

After Code Review

- After Development
- After Code Review ($25/fix)
- After QA test ($200/fix)
- After Customer ($1000/fix)

Before Code Review

Cost of fixing bugs: $174k
+ Cost of 194 latent bugs: $194k
--------------------------------------------------
Total Cost: $368k

After Code Review

Cost of fixing bugs: $120k
+ Cost of 32 latent bugs: $32k
--------------------------------------------------
Total Cost: $152k

Before Code Review

Cost of fixing bugs: $174k
+ Cost of 194 latent bugs: $194k
--------------------------------------------------
Total Cost: $368k

After Code Review

Cost of fixing bugs: $120k
+ Cost of 32 latent bugs: $32k
--------------------------------------------------
Total Cost: $152k
• Requirements
• Design
• Implementation
  – Critical components
  – Complex components
  – New employee’s work
  – New technology or platform
• Test Plans
• Test Cases
**GUIDING PRINCIPLES**

<table>
<thead>
<tr>
<th>Guiding Principle</th>
</tr>
</thead>
<tbody>
<tr>
<td>Don’t wait until the product is finished! Inspect early and often!</td>
</tr>
<tr>
<td>Make first review early (when 10-15% of the work is finished)</td>
</tr>
<tr>
<td>Combine formal and informal reviews</td>
</tr>
<tr>
<td>Limit inspection meeting within 2 hours</td>
</tr>
<tr>
<td>Limit the amount of reviewed code up to 300 lines</td>
</tr>
<tr>
<td>Aim for an inspection rate of 200 lines of code per hour</td>
</tr>
</tbody>
</table>
• Provide resources and time for peer review
• Include peer review in the project schedule
• Set policies, expectations and goals
• Ask for status report
MANAGERS ROLE

COMMON MISTAKE – NEVER ASSESS INDIVIDUALS BASED ON PEER REVIEW DATA!
• Provide a nonthreatening environment so that open discussions can occur

• Make peer review part of your software engineering culture
• Software architecture evaluation and implementation conformance evaluation
• Path coverage testing
• Load, stress, and performance testing
• Continuous integration
• ...
• “All Models are wrong but some are useful.” George Box

• Tailor peer review so that it best fits your needs!
QUESTIONS

kbailov@gmail.com
krasimir.baylov@musala.com