Test-Focused Knowledge Sharing using A3-assisted Communication

- Lars Meskestad
Testers’ Requirements

- Overall goals
- System overview
- Functions in context
- Spend less time searching for design specific information

<table>
<thead>
<tr>
<th>Category</th>
<th>Statements - Specific</th>
</tr>
</thead>
<tbody>
<tr>
<td>Test Goals</td>
<td>I know the specific goals for testing in the project</td>
</tr>
<tr>
<td>Test Level</td>
<td>I know exactly what level of testing to perform on each part of the system that I am responsible for</td>
</tr>
<tr>
<td>System Overview</td>
<td>I have sufficient information about the functional structure of the system</td>
</tr>
<tr>
<td>Requirements</td>
<td>Quality of the current requirements provide me with sufficient value for the purpose of testing</td>
</tr>
<tr>
<td>Design</td>
<td>I have sufficient information about how features/functions in scope are designed to work</td>
</tr>
<tr>
<td>Interfaces</td>
<td>I know how features/functions are related to each other so that I can test interfaces and dependencies</td>
</tr>
<tr>
<td>Category</td>
<td>Statements - General</td>
</tr>
<tr>
<td>Availability</td>
<td>I spend more time writing test cases or executing them than on gathering information</td>
</tr>
<tr>
<td>Detail</td>
<td>The detail of information about the system is sufficient for me to adequately test it</td>
</tr>
<tr>
<td>Collaboration</td>
<td>The current situation allows me to work closely with developers, stakeholders and architects</td>
</tr>
<tr>
<td>Test Efficiency</td>
<td>The current situation allows me to be an effective tester</td>
</tr>
<tr>
<td>Test Coverage</td>
<td>The current situation allows me to create a sufficient amount of tests for my area of responsibility</td>
</tr>
<tr>
<td>Quality Confidence</td>
<td>I believe that the system will be properly tested based on the current state and validity of information</td>
</tr>
</tbody>
</table>
Causes

- Complex information database
- Documents with large and complex texts
- Models are often object-oriented
- Knowledge in the heads of experts
- Various mental models
- Knowledge evolves and changes
- Temporary knowledge sharing
A3 - Feedback

- General and concise
- Quick overview
- Little noise
- Multiple necessary aspects
A3 – Event Performance

- Functions involved
- Components under test
- Relationship between software and hardware
- Measurables
A3 – Activities/Functions

- Main activities
- Functional flow
- Major decisions
A3 – Feedback

- Insight in process
- Overview of measurables
- Easier to learn from
- Conversation platform
- May decrease code reading
## Findings – Benefits

### Evaluation – Benefits of A3

<table>
<thead>
<tr>
<th>Benefit</th>
<th>Strongly Agree</th>
<th>Agree</th>
<th>Slightly Agree</th>
<th>Neutral</th>
<th>Slightly Disagree</th>
<th>Disagree</th>
<th>Strongly Disagree</th>
<th>NPS</th>
</tr>
</thead>
<tbody>
<tr>
<td>Can make it <strong>easier to communicate</strong> with colleagues</td>
<td>3</td>
<td></td>
<td>4</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>+7</td>
</tr>
<tr>
<td>Can <strong>prevent misinterpretation</strong> between designers, architects and testers</td>
<td>3</td>
<td>2</td>
<td>2</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>+5</td>
</tr>
<tr>
<td>Can <strong>increase awareness</strong> about system aspects that are poorly understood</td>
<td>5</td>
<td></td>
<td>2</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>+7</td>
</tr>
<tr>
<td>Can make it easier to <strong>maintain the validity</strong> of information over time</td>
<td>4</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td></td>
<td>1</td>
<td></td>
<td>+3</td>
</tr>
<tr>
<td>Could <strong>prevent several severe issues</strong></td>
<td>1</td>
<td>4</td>
<td>2</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>+5</td>
</tr>
<tr>
<td>Can help me in my <strong>daily work</strong></td>
<td>3</td>
<td></td>
<td>2</td>
<td>2</td>
<td></td>
<td></td>
<td></td>
<td>+5</td>
</tr>
</tbody>
</table>
Findings - experience

- No special tool needed
- Good conversation platform
- Allows quick feedback
- Available and transparent knowledge
- Contextual understanding

- Need modelling skills
- Capable of abstraction
- Understand various views

- Works when informal
Conclusion

- Increases knowledge sharing between testers and rest of project members
- Creates awareness about little known areas
- Emphasizes essential knowledge
- More available and transparent knowledge
- Can prevent serious problems

- Skilled modeller(s) should be involved (abstraction)
- Should have a support system
- May benefit from specific roles
Insight into eachother’s knowledge