Trustworthiness in testing and CI/CD
Software Center day Denmark

Kristian Sandahl, Eduard Enoiu, Jean Malm
Projects at Theme 1 Continuous Delivery

- #6 Enterprise Scale Continuous Integration and Delivery
- #18 Data Visualization for Continuous Integration
- #29 Modeling and Analyzing Collaborating Machines
- #30 Aspects of Automated Testing
Transparency of the CI/CD flow
## TABLE V

**IMPORTANCE, FREQUENCY, EFFORT AND TIME WITH RESPECT TO INFORMATION NEEDS**

<table>
<thead>
<tr>
<th>ID</th>
<th>Information Need</th>
<th>Importance</th>
<th>Frequency</th>
<th>Effort</th>
<th>Time</th>
</tr>
</thead>
<tbody>
<tr>
<td>C1</td>
<td>How much confidence do we have in the release to deploy to the customers?</td>
<td>4.8</td>
<td>4.9</td>
<td>5.0</td>
<td>15-20</td>
</tr>
<tr>
<td>CC6</td>
<td>Is the given feature ready to release to customers?</td>
<td>4.5</td>
<td>4.6</td>
<td>5.0</td>
<td>15-20</td>
</tr>
<tr>
<td>B3</td>
<td>Is the bug fix ready to release to customers?</td>
<td>4.5</td>
<td>4.6</td>
<td>5.0</td>
<td>15-20</td>
</tr>
<tr>
<td>C2</td>
<td>How much confidence do we have in the test suite?</td>
<td>4.1</td>
<td>4.9</td>
<td>5.0</td>
<td>&gt;20</td>
</tr>
<tr>
<td>C3</td>
<td>How much confidence do we have in stand-alone projects to be merged into the master branch/baseline?</td>
<td>4.1</td>
<td>4.7</td>
<td>4.8</td>
<td>&gt;20</td>
</tr>
<tr>
<td>CC2</td>
<td>What is the status/health of new code changes?</td>
<td>4.1</td>
<td>4.6</td>
<td>4.8</td>
<td>15-20</td>
</tr>
<tr>
<td>CC4</td>
<td>Which change request does the specific commit implements?</td>
<td>4.0</td>
<td>4.7</td>
<td>3.5</td>
<td>10-15</td>
</tr>
<tr>
<td>CC1</td>
<td>Does the final release to customers include my code?</td>
<td>4.0</td>
<td>3.7</td>
<td>2.5</td>
<td>5-10</td>
</tr>
<tr>
<td>T3</td>
<td>In which environment/machine do specific test cases fail?</td>
<td>3.8</td>
<td>4.7</td>
<td>4.5</td>
<td>&gt;20</td>
</tr>
<tr>
<td>T7</td>
<td>Which test cases are flaky?</td>
<td>3.7</td>
<td>4.7</td>
<td>5.0</td>
<td>&gt;20</td>
</tr>
<tr>
<td>CC5</td>
<td>Is the given feature implemented?</td>
<td>3.6</td>
<td>4.6</td>
<td>4.5</td>
<td>10-15</td>
</tr>
<tr>
<td>B1</td>
<td>Which bugs have been fixed in the specific release?</td>
<td>3.2</td>
<td>4.3</td>
<td>3.3</td>
<td>10-15</td>
</tr>
</tbody>
</table>
Visualization Eiffel Store

https://github.com/eiffel-community/eiffel-store
Solution by Axis and students from Göteborg
Ongoing and future activities for project #18

- Using fast network visualization methods
- Prediction of Eiffel events
Bottlenecks and trustworthiness in automated testing
Diversity-based testing

Coverage

https://gitlab.liu.se/azeah70/diversitybasedtesting

Removed, too similar test cases
Flaky tests detection

Factors:
- Traceback coverage
- Flip frequency
- TC size
- Test smells

https://gitlab.liu.se/azeah70/multifactorftdetector
Data visualisation in continuous integration and delivery: Information needs, challenges, and recommendations

Improving Continuous Integration with Similarity-based Test Case Selection

Empirical Analysis of Practitioners’ Perceptions of Test Flakiness Factors

A Multi-factor Approach for Flaky Test Detection and Automated Root Cause Analysis

An Evaluation of Machine Learning Methods for Predicting Flaky Tests

First LiU SWC PhD, 2022-10-06

Dr. Azeem Ahmad
Future directions

- Testing of nn modules in a CI context
- How and by whom is confidence in the product built?
- The role of (automated) testing in sustainable SE
Trustworthy and Human-Centered AI-infused Testing

• Assure that AI-infused testing is sufficiently trustworthy to be deployed at industrial scale
• We identified
  • Challenges: assigning responsibility, bias in decision-making and lack of participation
  • Approaches: explicability, supervision and diversity.
• A checklist for ethical AI in testing and human-centered test design.
Static Analysis of Test Code

- **Aim:** Leverage Existing Tool Development Effort

- **We explored existing state-of-the-practice static analysis tools for C/C++**
  - Chosen based on usage for production

- **We identified:**
  - Existing tools’ challenges to understand test behaviour (1)
  - General lack of test-specific analysis functionality (2)

- **We proposed:**
  - Rule-based refutation of noisy reports (3)
  - Enhancements of existing checks (1)
Contact: kristian.sandahl@liu.se
jean.malm@mdu.se
eduard.paul.enoiu@mdu.se