QA&QC DURING THE SOFTWARE DEVELOPMENT LIFE CYCLE

Roxana Soporan
Andra Banto

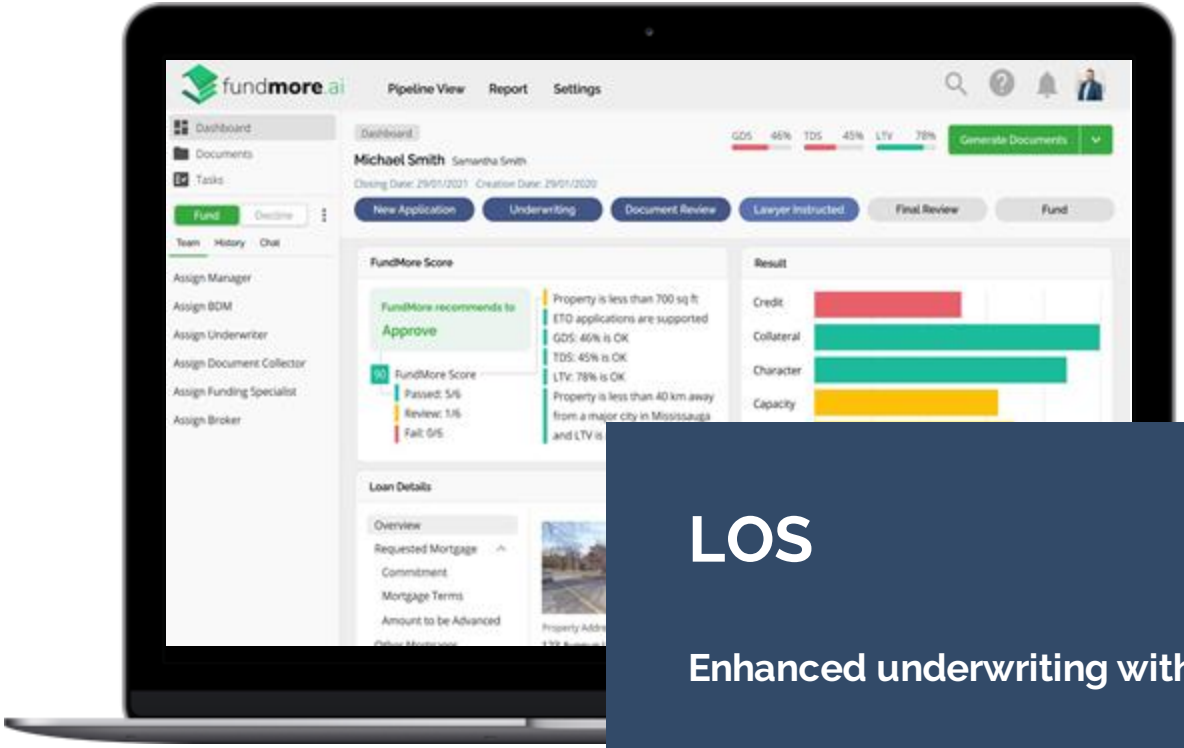
April 22, 2024

Agenda

01	Introduction
02	QA & QC and Testing
03	Continuous Improvement (CI)
04	SDLC, STLC
05	Test Planning, Design, Execution & Reporting
06	Test Levels and Test Types
07	Agile using Scrum
08	Stories
09	Tasks
10	Test Suites and Test Cases
11	Bugs and everything in between
12	Life in a company, post pandemic and current situation

Become acquainted with us. Who we are,

our purpose



FundMore AI

LOS

Enhanced underwriting with instant decisioning

Better Borrower Experience

Expedite and deliver a fully-transparent and automated underwriting process that's ideal for both lenders and borrowers.

Reduced Risk

Improve your company's risk management using predictive modelling and pattern recognition to assess risk and provide real-time reasoning.

Faster

Reduce funding times and application evaluation by more than 90% without increasing your risk.

Higher ROI

An efficient loan underwriting process lets your business assess and fund more files, boosting your profits, increasing operational efficiency, and reducing your expenses.

Book Your Demo

FULLY CUSTOMIZE YOUR WORKSPACE

Your company has its own approach to the mortgage underwriting process. FundMore.ai offers a modular experience that you can customize based on your needs.

Every component has flexible settings that you can manipulate, panels are removable, and the entire platform is built with personalization in mind.

FUNDMORE IQ

Never miss an important document again. FundMore.ai simplifies document collection using automated tools to help you capture and process documents into a secure platform.

Documents are easily accessible by underwriters, and clients can quickly submit documents. Our platform can reduce the cost of collection, processing, and verification by up to 90%.

$5.3T

Total Available Market

Funded mortgage volume in **North America**

$113B

Service Obtainable Market

25% of mortgage volume funded in **Canada**

$392B

Service Accessible Market

Funded mortgage volume by **Canadian** Lenders

* Residential mortgage volume in $CDN

Quality assurance and Quality control:

what is the difference?

QA & QC


QUALITY ASSURANCE
Focus on the prevention of defects
Proactive process
Process-based approach
Manages Quality
QUALITY CONTROL
Focus on the identification of defects
Reactive process
Product-based approach
Verify the Quality
Jelvix
Source: Scrum Time
jelvix.com

Measuring Quality

If you cannot define it

you cannot achieve it

If you cannot measure it

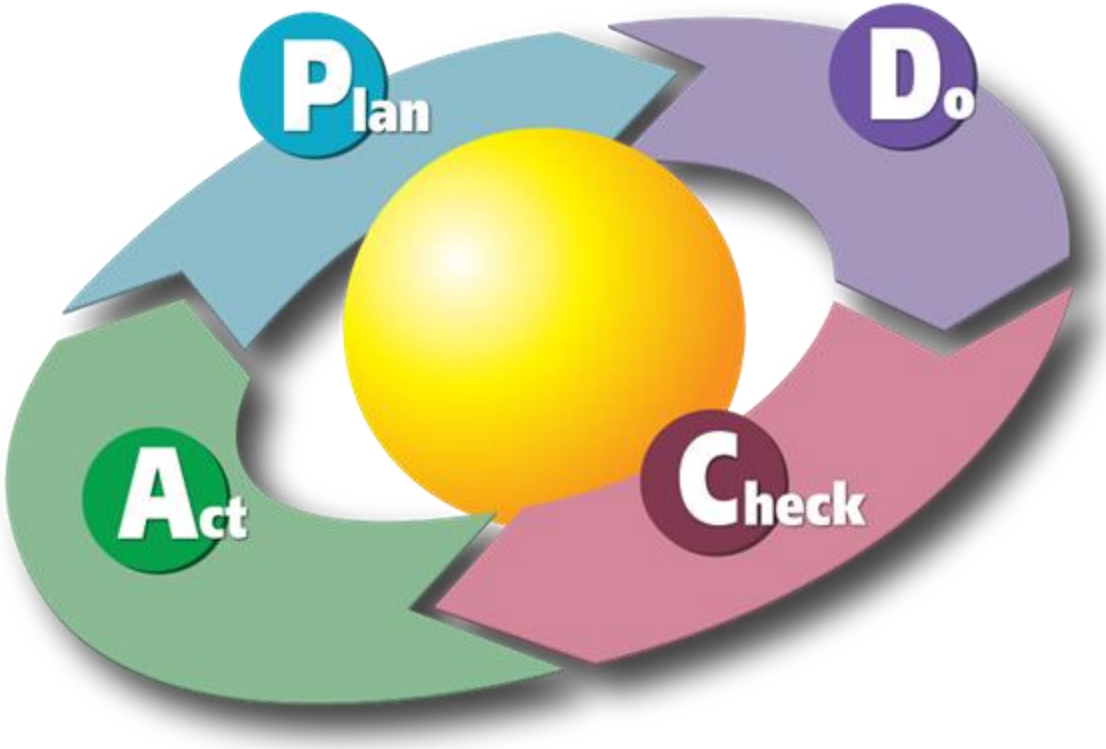
You do not know how you are progressing

You do not know when you have arrived

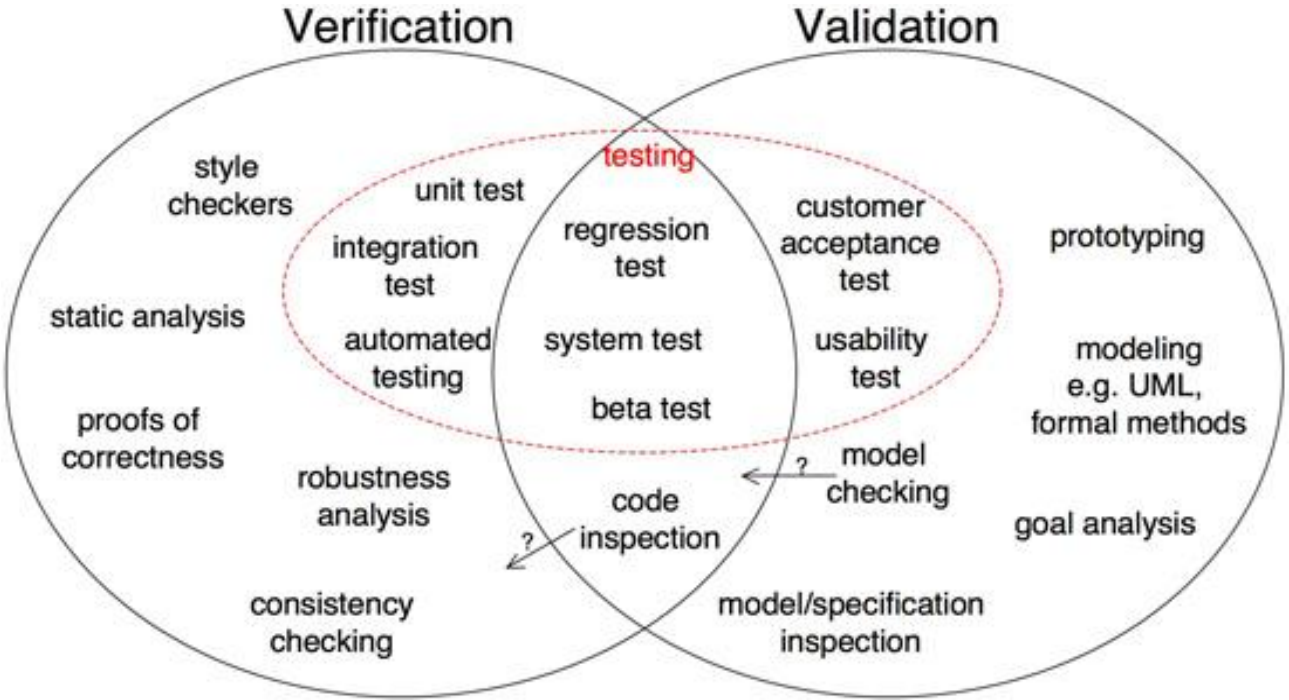
You cannot demonstrate it

Quality Management System (QMS)

Continuous Improvement

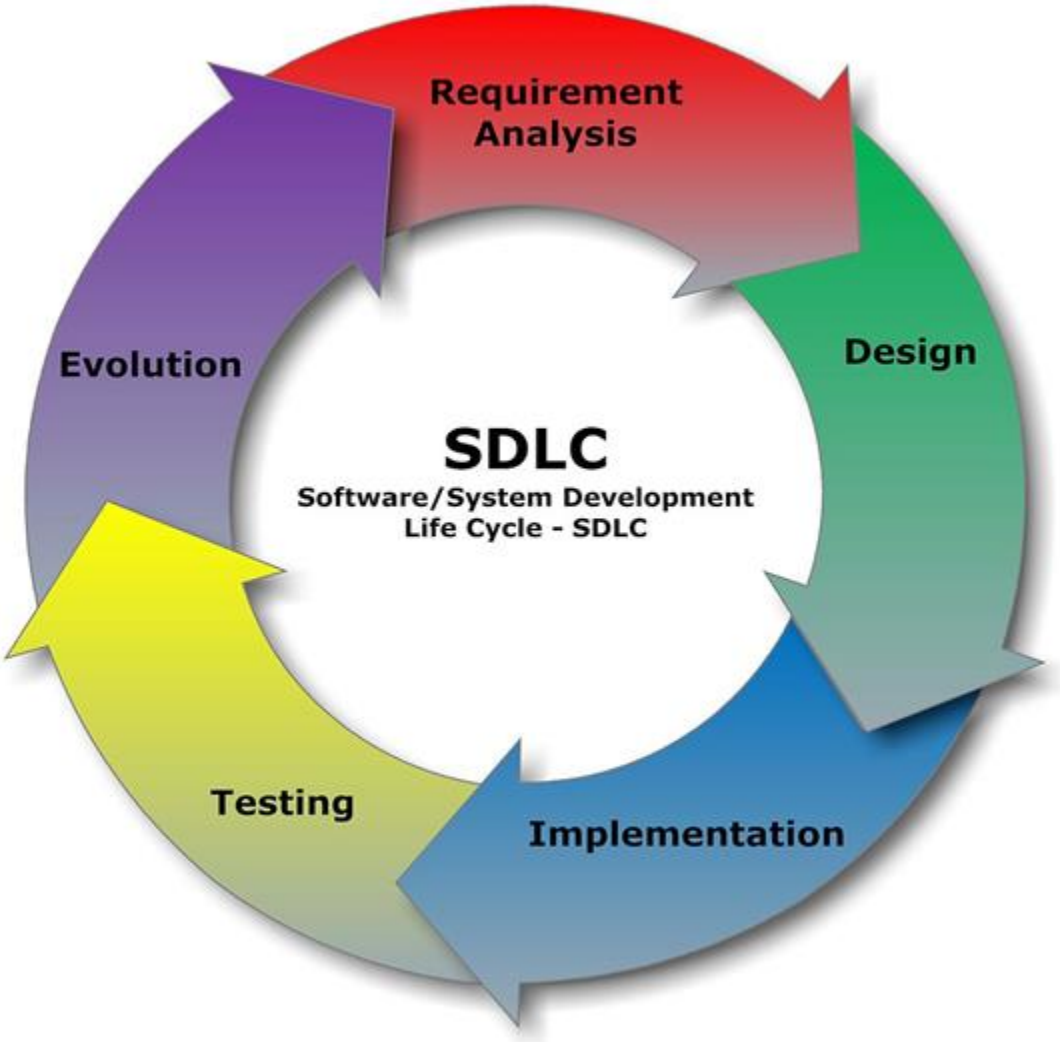


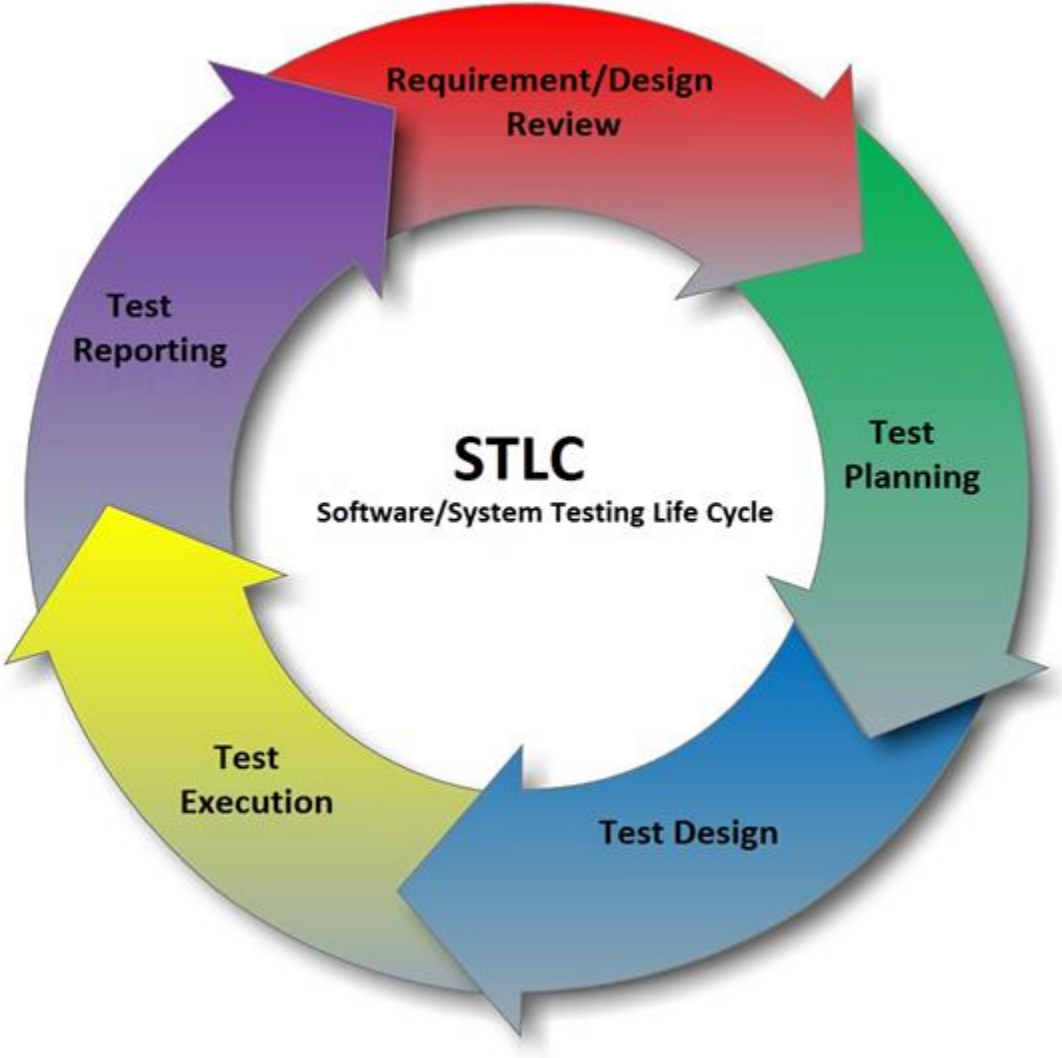
Verification & Validation

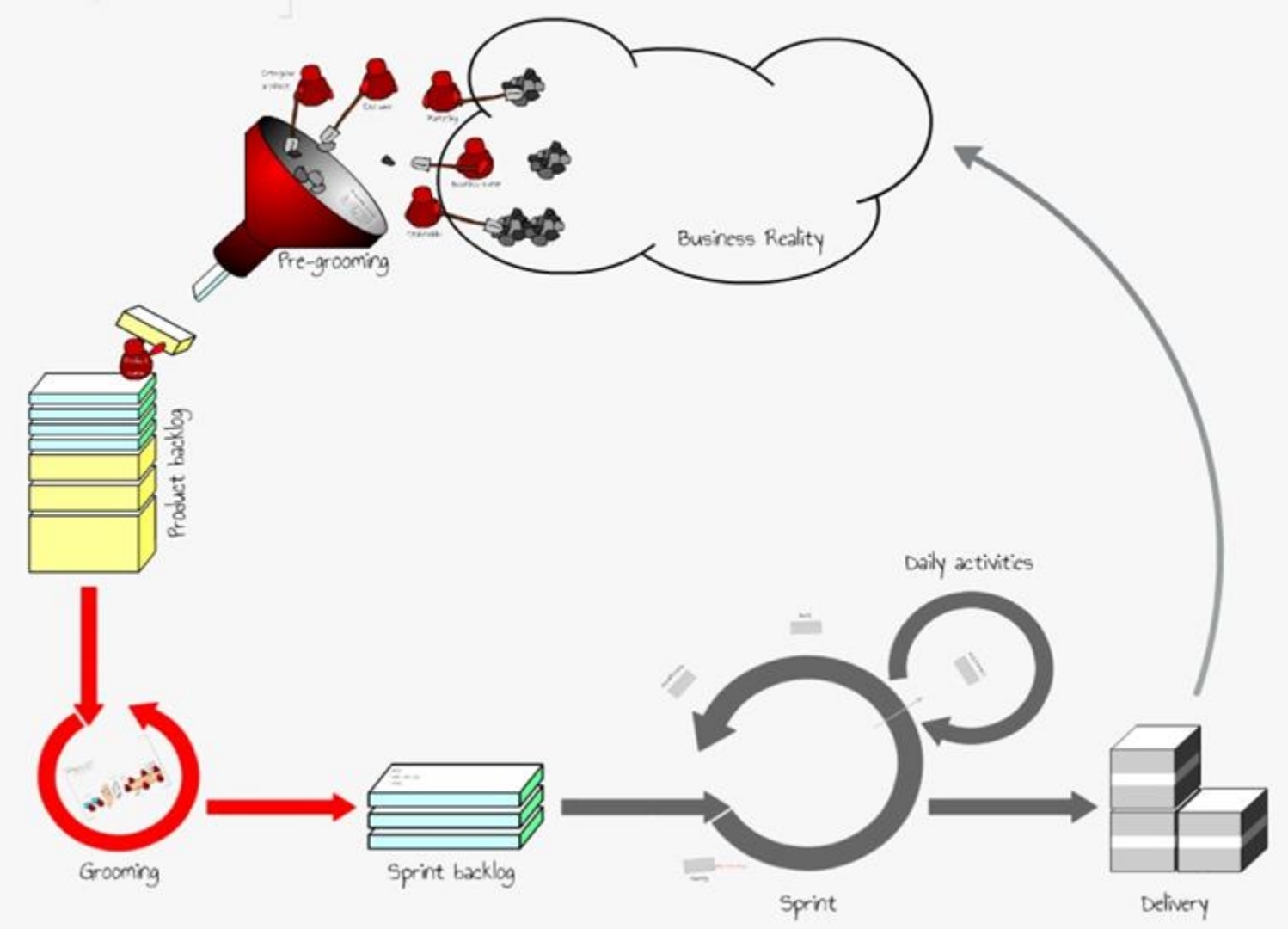


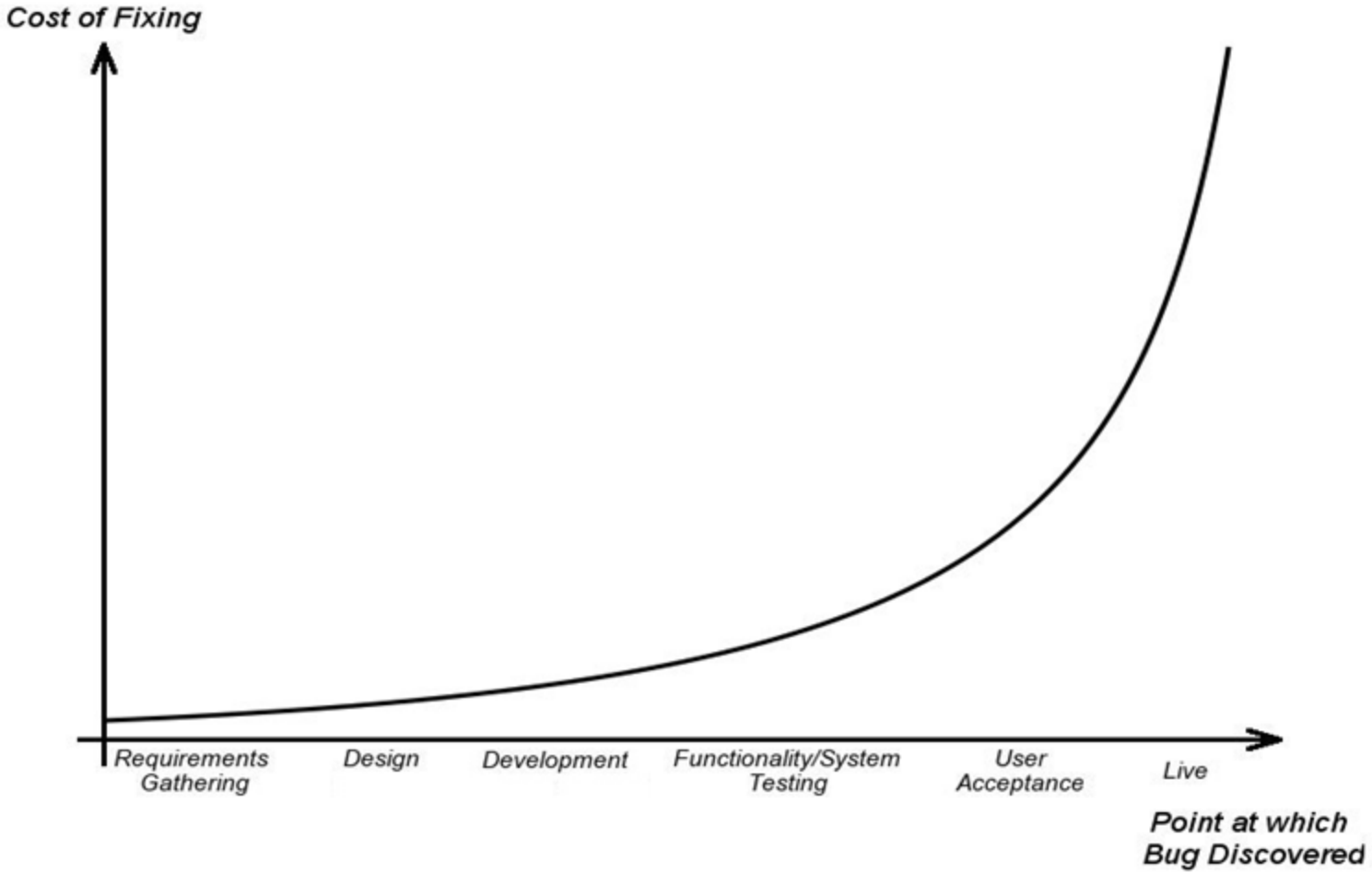
Why is software testing necessary?

"On 4 June 1996 the maiden flight of the Ariane 5 launcher ended in a failure. **Only about 40 seconds after initiation of the flight sequence, at an altitude of about 3700 m, the launcher veered off its flight path, broke up and exploded."**
(*https://www.esa.int*)


Requirement Analysis
Design
Implementation
Testing
Evolution
SDLC
Software/System Development Life Cycle - SDLC

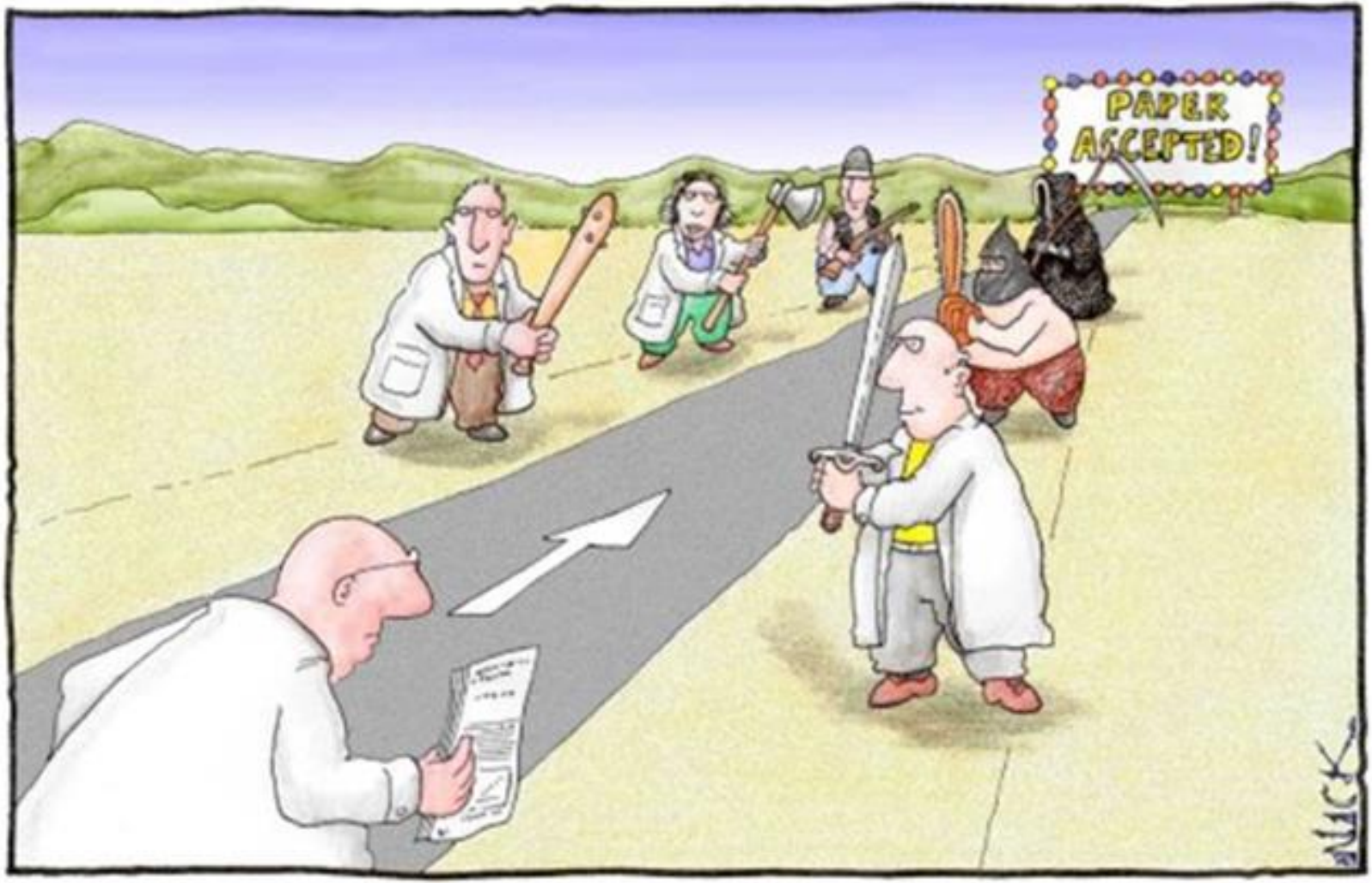

Requirement/Design Review
Test Planning
Test Design
Test Execution
Test Reporting
STLC
Software/System Testing Life Cycle


Business Reality
Pre-grooming
Product backlog
Daily activities
Grooming
Sprint backlog
Sprint
Delivery

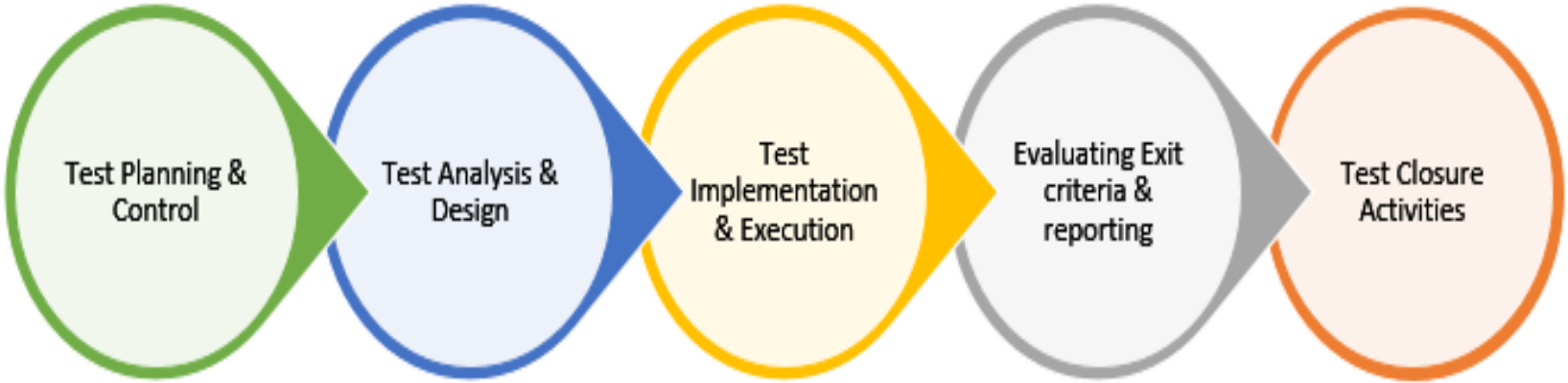

Cost of Fixing
Requirements Gathering
Design
Development
Functionality/System Testing
User Acceptance
Live
Point at which Bug Discovered

Why is software testing important?

"On 4 June 1996 the maiden flight of the Ariane 5 launcher ended in a failure. **Only about 40 seconds after initiation of the flight sequence, at an altitude of about 3700 m, the launcher veered off its flight path, broke up and exploded."**
(*https://www.esa.int*)


PAPER ACCEPTED!

The fundamentals of test process



Test Planning and Control

Test Planning:

Activities:

- Determine the scope and risks and identify the objectives of testing
- Determine the test approach (techniques, test items, coverage, identifying the testing team, resources)
- Schedule test analysis and design tasks, test implementation, execution and evaluation
- Determine exit criteria

Deliverables: Test Policy, Test Strategy, Test Plan

Test Control:

Activities:

- Compare actual progress against the planned progress
- Monitor and document progress, test coverage and exit criteria and provide information on testing through reports
- Initiate corrective actions, if necessary

Deliverables: reports, test data

Test Analysis and Design

The phase where general testing objectives are transformed into tangible test conditions and test designs.

Activities:

- Review the test basis examining the specification for the software that we are testing
- Identify test conditions
- Design the high-level tests
- Evaluate testability of the requirements and the system
- Design the test environment setup and identify any required infrastructure and tools

Deliverables: test conditions, test suites

Test Implementation and Execution

Test Implementation:

Activities:

- Develop and prioritize our test-suites
- Create scenarios and test-cases derived for efficient test execution
- Implement / configure and verify the test environment

Deliverables: test data, test cases, test scripts.

Test Execution:

Activities:

- Execute the test-suites and test-cases
- Compare actual result with expected results
- Log the outcome of the test execution, report discrepancies (bugs, defects, error reports)
- Re-execute the tests that previously failed in order to confirm the fixes

Deliverables: test-execution reports, issues, issue-reports

Evaluating Exit Criteria and Reporting

Evaluating exit criteria is the activity where test execution is assessed against the defined objectives.

Activities:

- Check the test logs against the exit criteria specified in test planning phase
- Assess if more tests are needed or if the exit criteria specified should be changed
- Write a test summary report for stakeholders

Deliverables: Test Summary Report, other reports

Test Closure **activities**

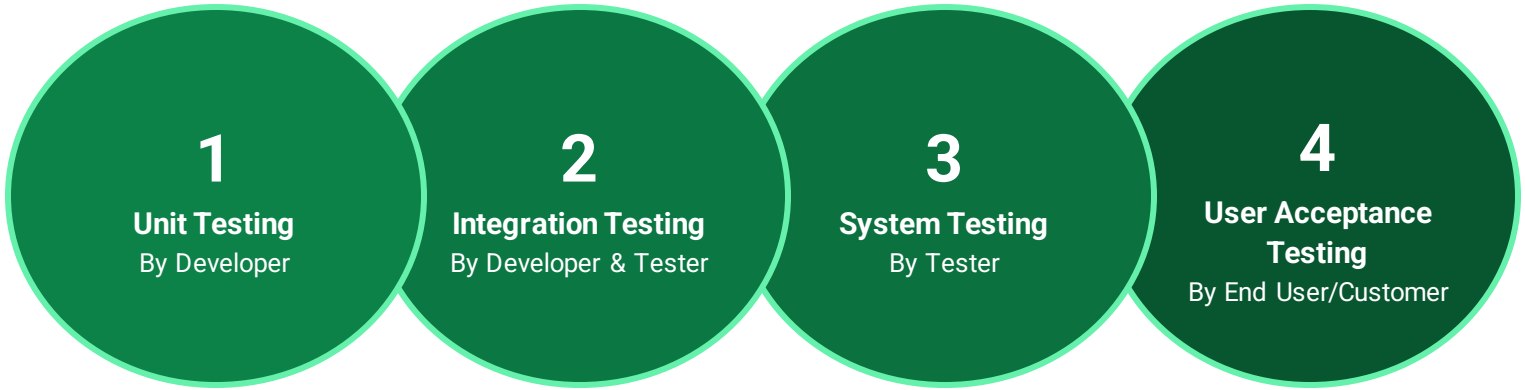
Closing the corresponding testing activities.

Activities:

- Check which planned deliverables are actually delivered and to ensure that all incident reports have been resolved
- Finalize and archive testware such as scripts, test environments, etc. for later reuse
- Handover the testware to the maintenance organization. They will give support to the software and make any bug fixes or maintenance changes.
- Evaluate how the testing went and learn lessons for future releases and projects

Deliverables: testware, archives with testware; process improvement suggestions

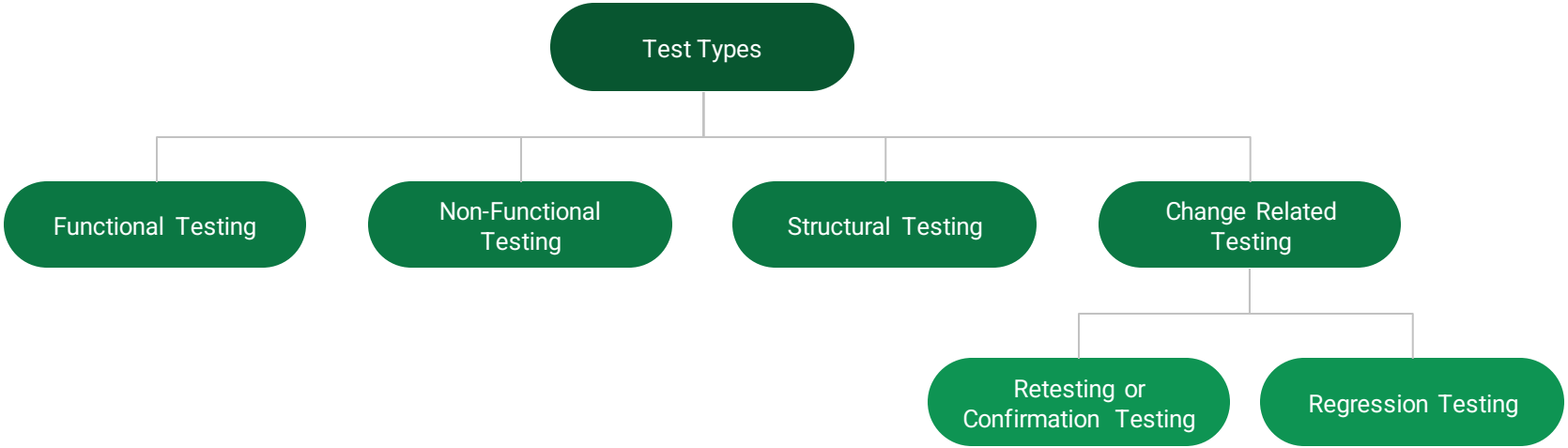
Test Levels


1
Unit Testing
By Developer
2
Integration Testing
By Developer & Tester
3
System Testing
By Tester
4
User Acceptance Testing
By End User/Customer


Unit test vs. Integration test


2 UNIT TESTS, 0 INTEGRATION TESTS
via reddit.com/r/programmerhumor

Test Types


Test Types
Functional Testing
Non-Functional Testing
Structural Testing
Change Related Testing
Retesting or Confirmation Testing
Regression Testing

Questions?

Agile Software Development using SCRUM

The SCRUM methodology is a framework within which people can address complex adaptive problems, while productively and creatively delivering products of the highest possible value.

In other words Scrum is a lightweight framework that helps people, teams and organizations generate value through *adaptive solutions* for complex problems.

Agile Software Development using SCRUM

A SCRUM Team is usually formed by 5 to 9 members working together to deliver the required product increments.

In Scrum, there are three roles: **Product Owner, Development Team, and Scrum Master**. Together these are known as the Scrum Team.

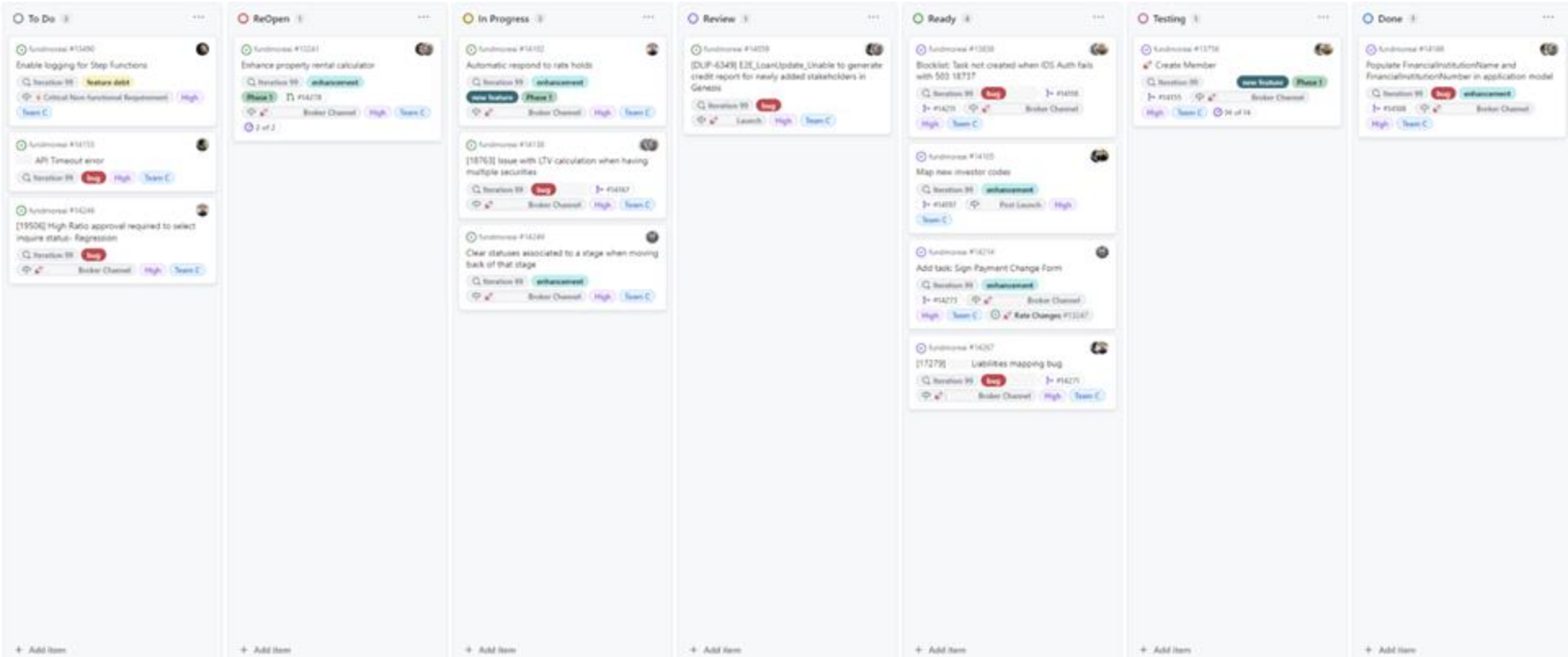
The common roles in our company within a SCRUM Team are:

- Product Owner – "translates" the customer needs to the whole team; refines the Project Backlog (what to do)
- Team Lead – ensures that the process runs smoothly, removes impediments, organizes critical events and meetings (how to do things)
- Developer – writes the code for the project
- Tester – tests the code
- Designer – creates user-centered designs that meet the needs of end-users

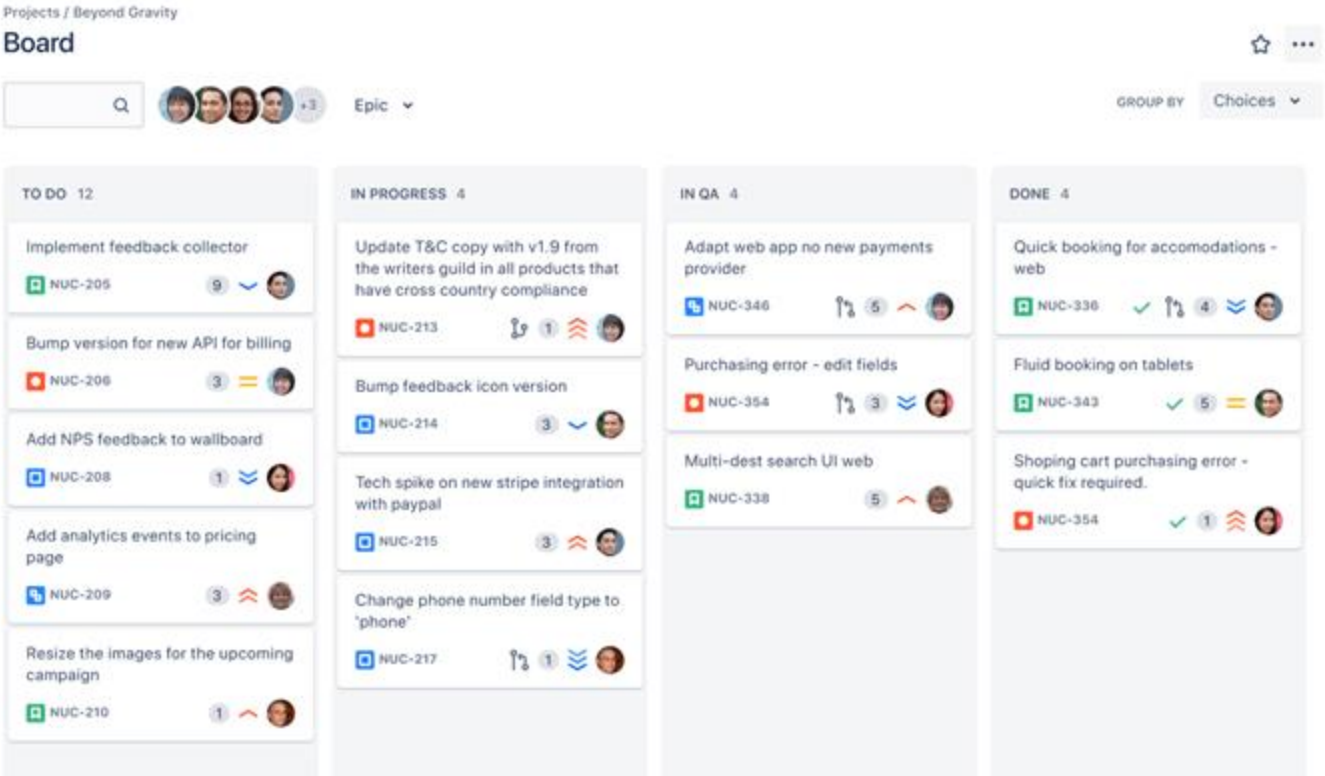
Agile Software Development using SCRUM

- The most common AGILE Project Management tool that is designed to support any AGILE methodology, be it Scrum, Kanban, or something else is Jira Software.
- From sprint boards to reports, the team can plan, track, and manage all their software development projects within a single tool.
- But there are others: Youtrack, Rally, Azure, Helix, Trello, **Github**.

What does a SPRINT BOARD look like in Github?



What does a SPRINT BOARD look like in Jira?



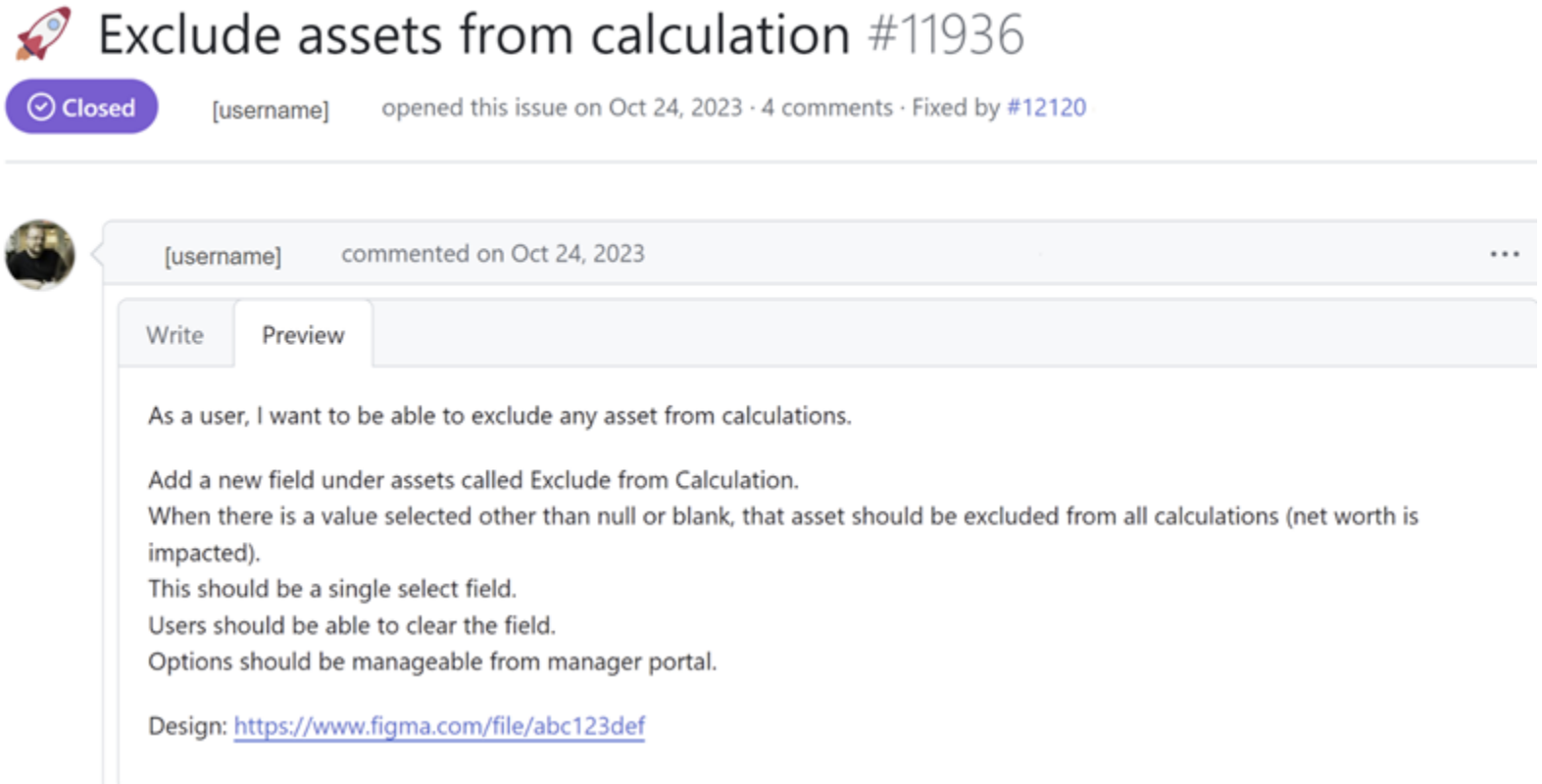
What is a USER STORY?

Is a small, self-contained unit of development work designed to accomplish a specific goal within a product (It can represent a user's need, serve as a planning item in agile software development).

- We can look at it as a (software) requirement formulated in everyday language so it can be easily understood by everyone who reads it; usually follows the simple format:

  E.g: **“As a user I want to perform <this action> so that I can accomplish <this goal>.”**

What does a USER STORY look like?



What is a TEST SCENARIO?
What is a TEST CASE?

Test Scenario:

Is a collective set of test cases which helps the testing team to determine the positive and negative characteristics of the project.

- Gives a high-level idea of what we need to test.
- As a tester, you should put yourself in the end user's shoes and figure out the real-world scenarios and use cases of the applications under test.

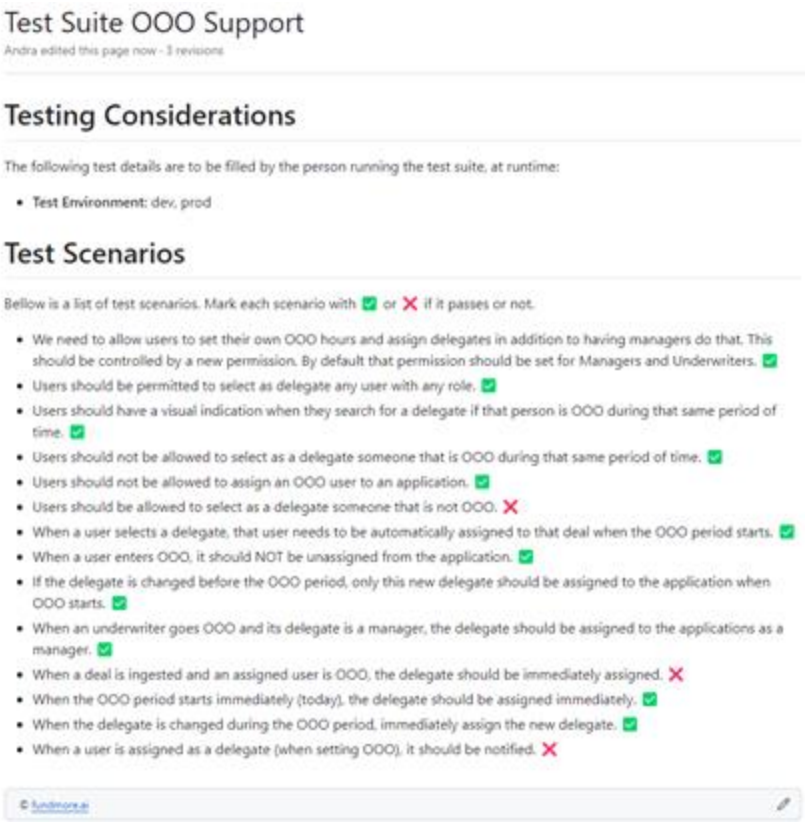
Test Case:

Set of actions executed to verify a particular feature or functionality of the software application.

It contains:

- Input values
- Preconditions
- Steps to reproduce
- Expected results
- Postconditions

What does a TEST CASE look like?



Test Suite OOO Support

Andra edited this page now · 3 revisions

Testing Considerations

The following test details are to be filled by the person running the test suite, at runtime:

- **Test Environment:** dev, prod

Test Scenarios

Bellow is a list of test scenarios. Mark each scenario with ✅ or ❌ if it passes or not.

- We need to allow users to set their own OOO hours and assign delegates in addition to having managers do that. This should be controlled by a new permission. By default that permission should be set for Managers and Underwriters. ✅
- Users should be permitted to select as delegate any user with any role. ✅
- Users should have a visual indication when they search for a delegate if that person is OOO during that same period of time. ✅
- Users should not be allowed to select as a delegate someone that is OOO during that same period of time. ✅
- Users should not be allowed to assign an OOO user to an application. ✅
- Users should be allowed to select as a delegate someone that is not OOO. ❌
- When a user selects a delegate, that user needs to be automatically assigned to that deal when the OOO period starts. ✅
- When a user enters OOO, it should NOT be unassigned from the application. ✅
- If the delegate is changed before the OOO period, only this new delegate should be assigned to the application when OOO starts. ✅
- When an underwriter goes OOO and its delegate is a manager, the delegate should be assigned to the applications as a manager. ✅
- When a deal is ingested and an assigned user is OOO, the delegate should be immediately assigned. ❌
- When the OOO period starts immediately (today), the delegate should be assigned immediately. ✅
- When the delegate is changed during the OOO period, immediately assign the new delegate. ✅
- When a user is assigned as a delegate (when setting OOO), it should be notified. ❌

© fundmore.ai

What is a DEFECT/BUG?

- A human being can make an error (mistake) which produces a DEFECT (or a BUG, a flaw) in the code.
- If the defect in the code is executed, it can cause a **failure**, but not all defects result in failures.
- In other words, when **actual result** deviates from the **expected result** while testing, then it results into a defect. Hence, any deviation from the specification mentioned in the product functional specification document is a defect.
- Depending on the organization, the defect is called differently like: **bug**, **issue**, **incident** or **problem**.

How do we report a Bug?

When reporting a defect a tester should:

- Have a clear goal in mind, know exactly what he wants to transmit
- Provide developers, managers and others detailed information about the behavior observed and the defect itself
- Be precise, concise, clear, unambiguous
- Attach screenshots if you can
- Express the observations in a neutral tone, fact-focused and impartial

A defect should mainly contain:

- **Defect ID** - Unique identification number for the defect
- **Title**
- **Severity and priority**
- **Version** - Version of the application in which defect was found (optional)
- **Date Raised** - Date when the defect is raised
- The name of the **person** who found the defect
- **Status** of the defect
- **Description** of the defect with:
  - **Steps to reproduce**
  - **Expected result**
  - **Actual result**
- **Screenshots, logs or videos** which capture and prove the defect found

a short description of the bug

where was the bug reproduced (on what environment)

the steps that need to followed in order to reproduce the issue

what are the expected behavior and the current behavior

link to a video where the bug is reproduced with all necessary steps

Error when setting a Purchase Date #9155

Closed [username] opened this issue on May 3, 2023 · 2 comments

[username] commented on May 3, 2023

Describe the bug

When setting a Purchase Date for some applications, an error pops-up with the message:

```
Something went wrong! ValidationError: "purchaseDate" with value "2023-05-01T21:00:00.000Z" fails to match the required pattern: /^\d{4}-\d{2}-\d{2}$/i
```

Context

Test Environment: dev

Steps To Reproduce

1. Open an application and scroll down to 'Loan Details' -> Requested Mortgage
2. Set a Purchase Date
3. See error

Expected behavior

No error should pop up when introducing a Purchase Date.

Actual behavior

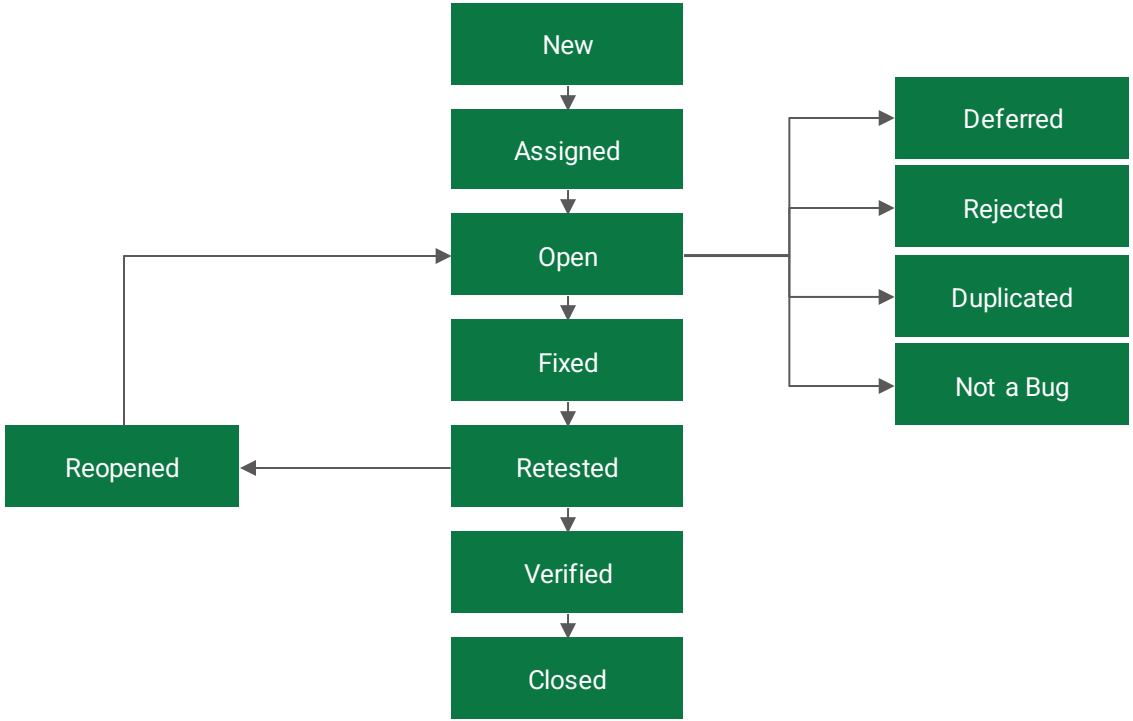
An error pops-up for some applications when introducing a Purchase Date.

Screenshots & Recordings

https://www.loom.com/share/28a3d

Add tasklist

Bug Life Cycle example



Testing & Quality

- Testing is part of quality assurance and it contributes to higher quality.

- Testing helps in **measuring** the quality of software in terms of the number of defects found, tests run, test coverage.

- Testing **identifies defects as early as possible**, before the software is in use, in this way avoiding unwanted cost of the stakeholders.

- Testing activities and QA activities complete each other:
  - the goal of a **Tester** is to find bugs, find them as early as possible and make sure they get fixed.
  - the goal of a **QA person** is to create and enforce standards and methods to improve the development process and to prevent bugs from ever occurring.

- Testing **gives confidence** in the quality of the software, because when testing find defects, the quality of the software increases if those defects are fixed.

Directions in testing area

AUTOMATION TESTING

Current situation

CONFERENCES, WORKSHOPS AND EVENTS

Current situation

WFH/WFO,
TEAMS,

TEAM BUILDINGS,

COMPANY EVENTS

Resources & Bibliography

Useful links:

- https://artoftesting.com/
- https://www.atlassian.com/agile
- https://www.softwaretestingtricks.com/
- https://www.guru99.com/
- http://tryqa.com/

Books:

Foundation of Software Testing – Dorothy Graham

Agile Testing. A Practical Guide For Testers and Agile Teams – Lisa Crispin & Janet Gregory

Thank you!

You can send us questions at:
roxana.soporan@fundmore.ai
andra.banto@fundmore.ai

Quiz time! Let’s play

01 ✓ Access: menti.com

02 ✓ Introduce the code mentioned in the course!

03 ✓ Make sure you fill in your entire name and group number, e.g. RoxanaSoporan_231