

intended number include Institute tools methods separate expected also time agile example oriented may analysis black executed failure team system Assurance whether following steps process development design source often regression acceptance requirements defects set defect used tester box one quality product developers value vs integrated failures usually fault offered different conditions well information needed Certification end white Input exit tested case two program testers unit functionality considered Certifications based cases usually reports data faults bugs inputs project common results International SQA latest level coverage performance found



# Software Testing Methods and Techniques

- Testing types
- Reporting tools and tracking user stories / bugs
- White/black box testing

## Waterfall

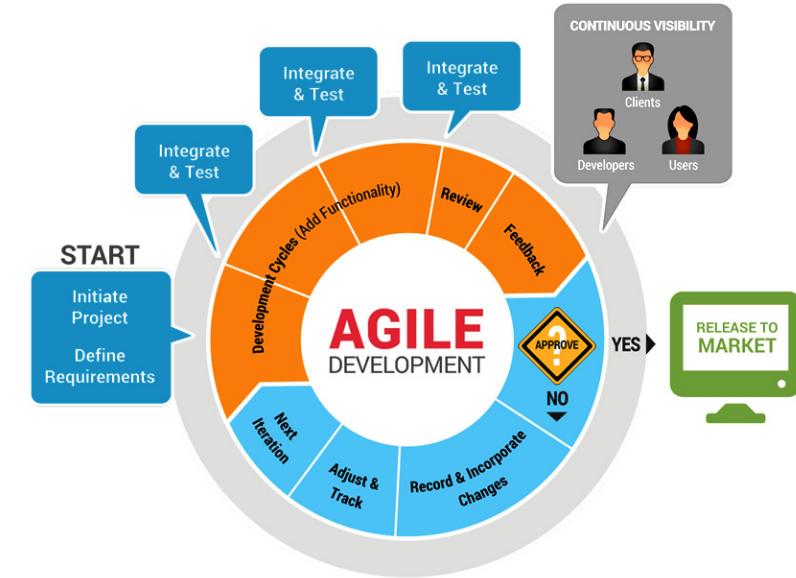
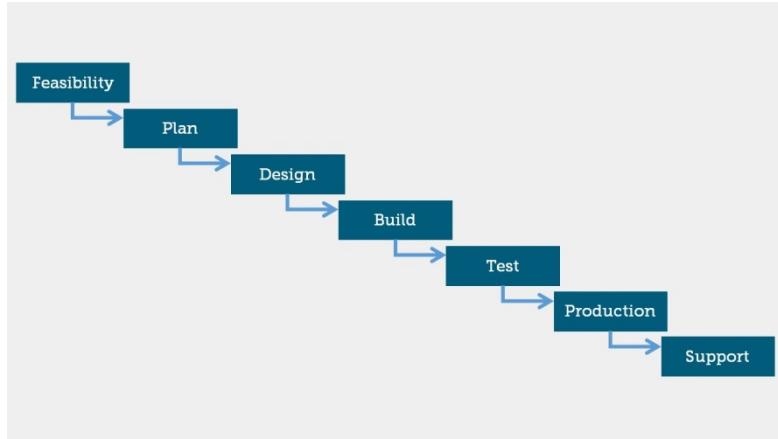
Sequence the tasks that lead to a final deliverable and work on them in order. One task must be completed before the next one begins, in a connected sequence of items that add up to the overall deliverable. It's an ideal method for projects that result in physical objects (buildings, computers), and project plans can be easily replicated for future use.

The power of this methodology is that every step is preplanned and laid out in the proper sequence. While this may be the simplest method to implement initially, any changes in customers' needs or priorities will disrupt the sequence of tasks, making it very difficult to manage.

## Agile

- The core of the Agile methodology was developed by 17 people in 2001 in written form. Their Agile Manifesto of Software Development put forth a groundbreaking mindset on delivering value and collaborating with customers. Agile's four main values are expressed as:
  - Individuals and interactions over processes and tools
  - Working software over comprehensive documentation
  - Customer collaboration over contract negotiation
  - Responding to change over following a plan

# Project management methods



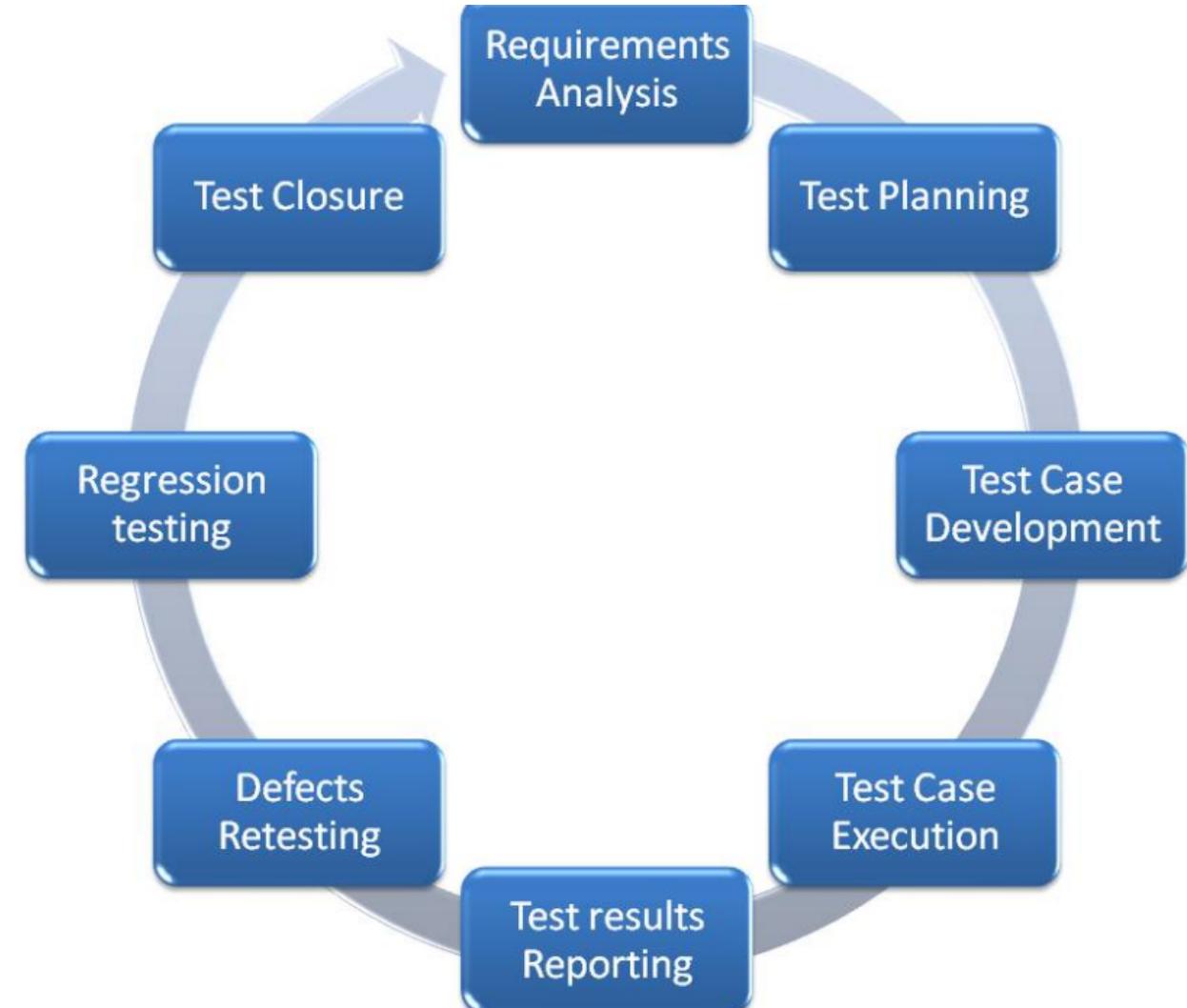
# Definition and the GOAL of Testing

Process of creating a program - phases:

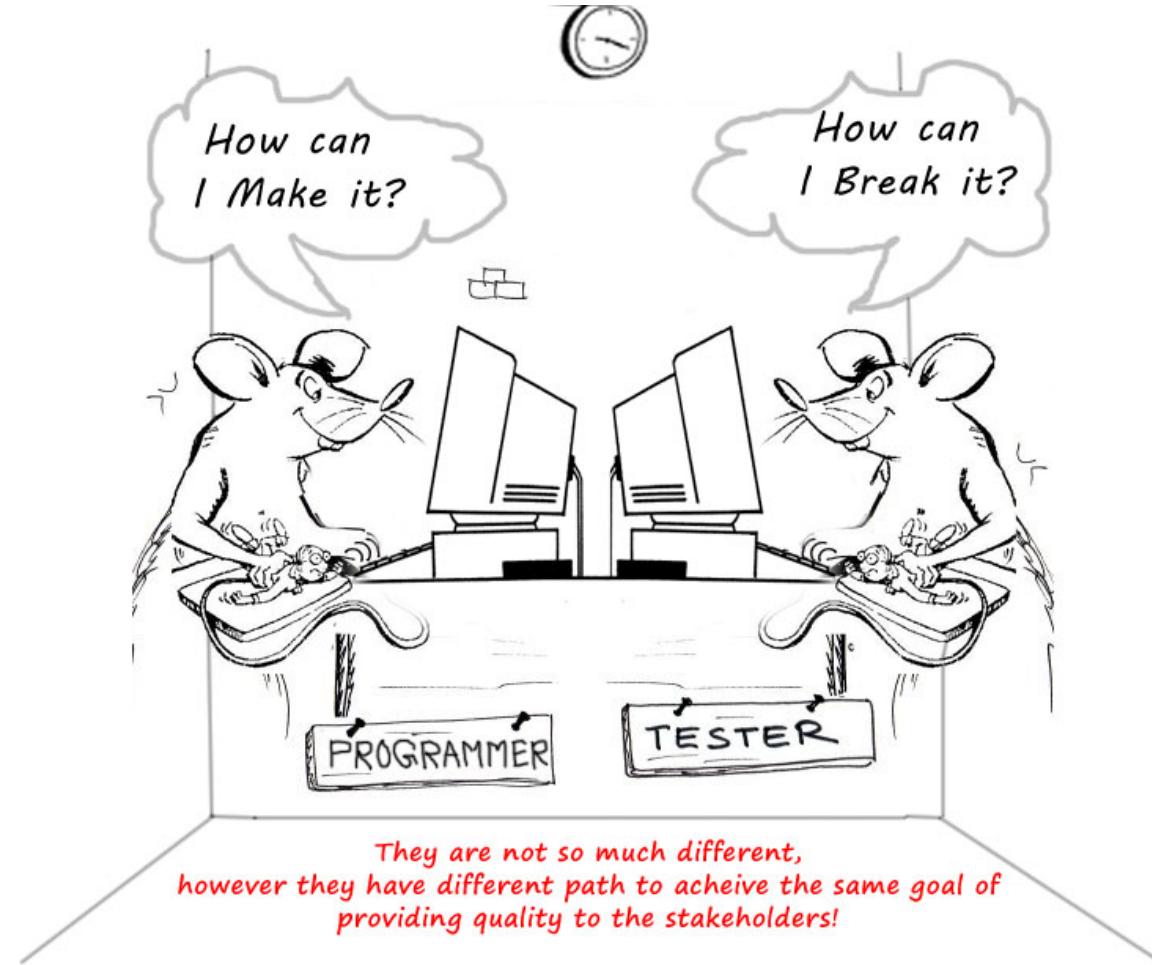
- defining a problem
- designing a program
- building a program
- analyzing performances of a program
- final arranging of a product

*A process of executing a program with the goal of finding errors.*

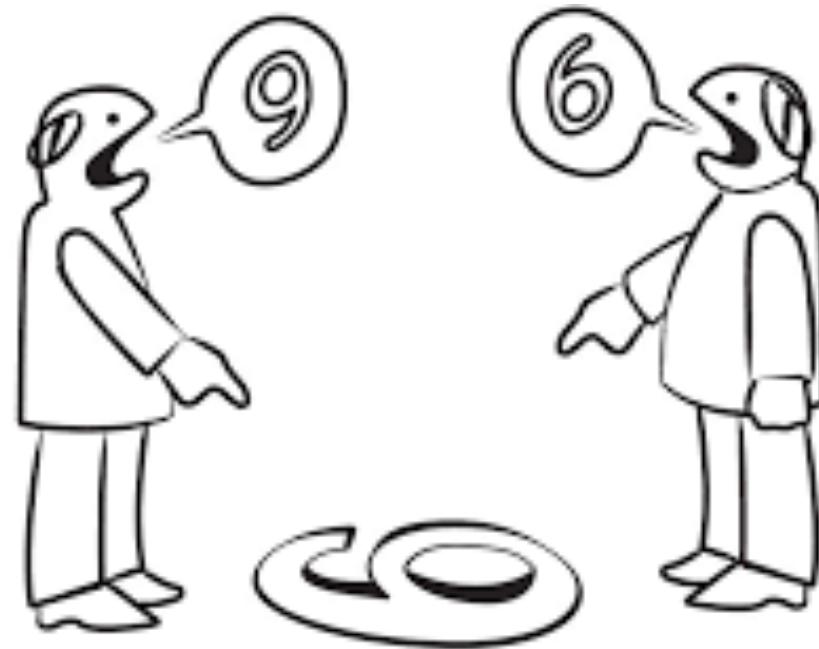
Good test cases = good testing



# Who should test ?



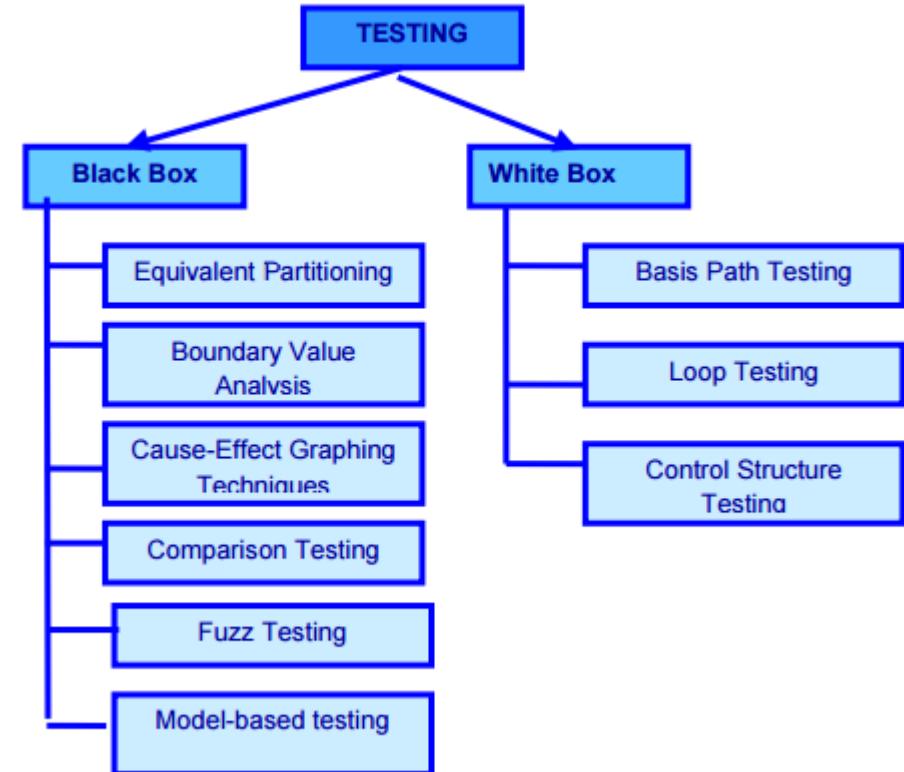
# Perspective ...



- Static vs. dynamic testing
  - Static – code review (verification)
  - Dynamic – test cases (validation)
- The box approach
  - White box testing
    - Code Complexity Tools (IDEs, Plugins, JAVA - CYVIS)
    - Code Coverage Tools (JAVA - Emma)
    - Memory Testing and Profiling Tools (HeapAnalyzer, JProfiler)
    - Unit Testing Tools (JUnit)
  - Black box testing

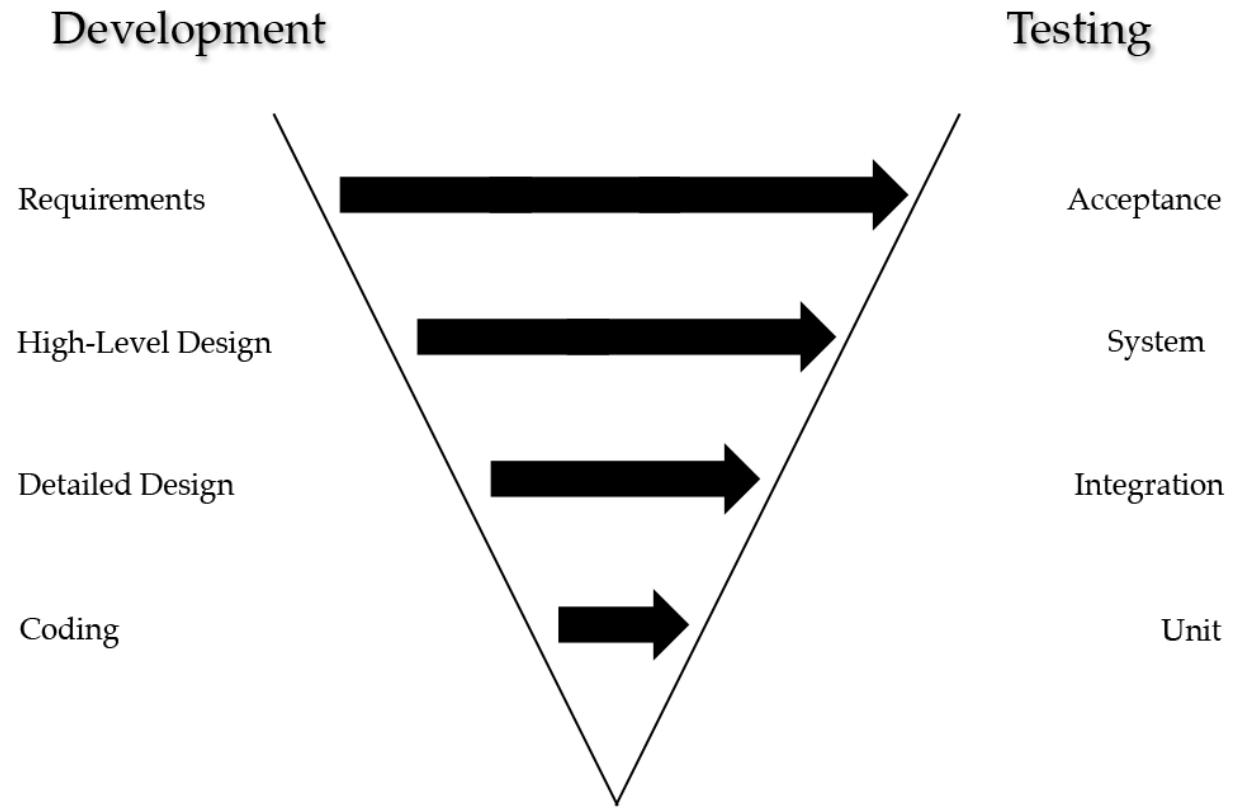
# General Classification of Test Techniques

- Equivalence Partitioning
- Boundary Value Analysis
- Cause-Effect Graphing Techniques
- Comparison Testing
- Fuzz Testing
- Model-based testing



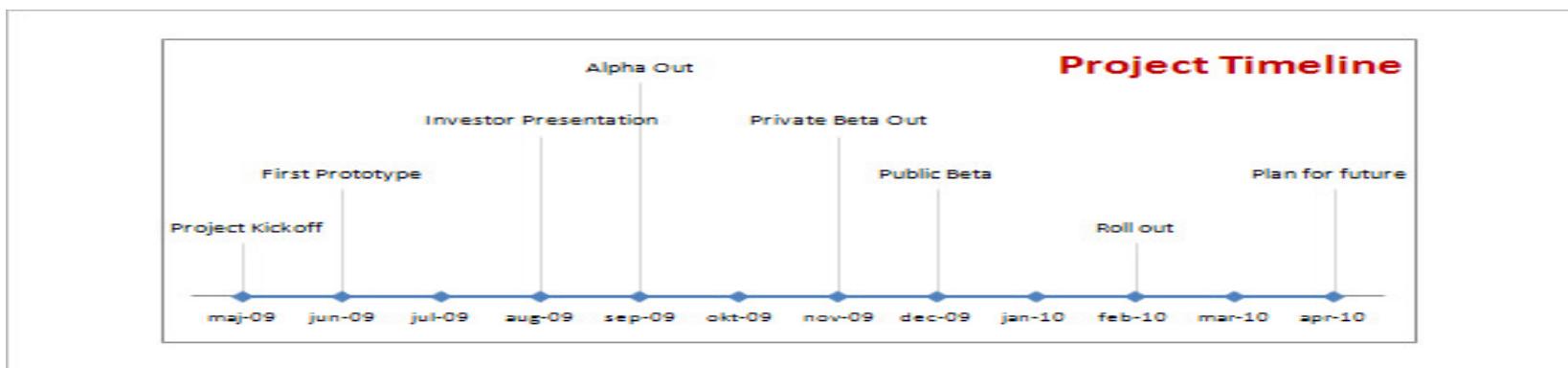
## V-Model:

- Unit testing
- Integration testing
- Component interface testing
- System testing
- Operational acceptance testing



# Testing types

- Installation testing
- Compatibility testing
- Smoke and Sanity tests
- Regression testing
- Acceptance testing
- Alpha testing
- Beta testing
- Functional vs Non-functional testing
- Continuous testing
- Destructive testing
- Performance testing
- Usability testing
- Accessibility testing
- Security testing
- Internationalization and localization
- Development testing
- A/B testing
- Concurrent testing
- Conformance or type testing
- And probably many others ...



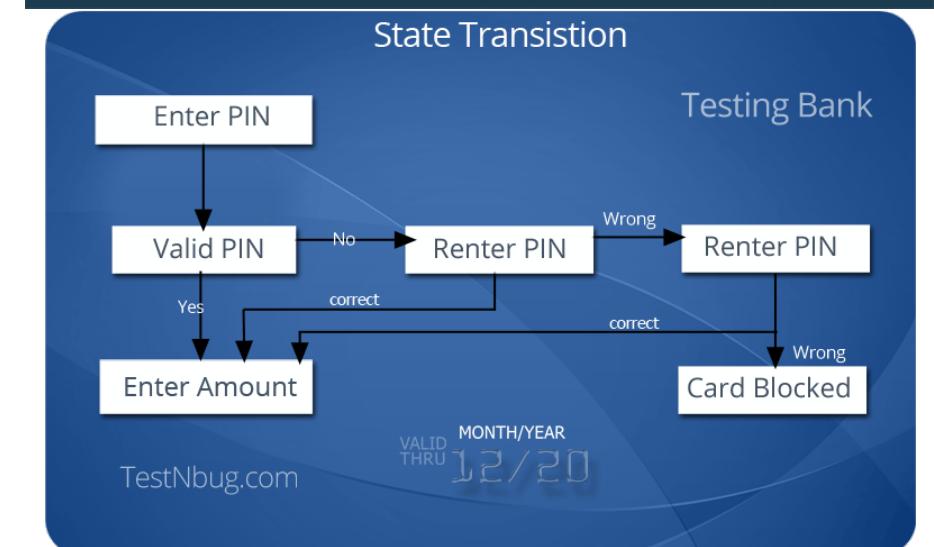
# Test Techniques According to the Project of the IEEE Computer Society, 2004

- Ad hoc testing (skill, intuition and experience)
- Exploratory testing
- Equivalence partitioning
- Boundary-value analysis
- Decision table - logical relationships between inputs and outputs
- State based testing (states and transitions)
- Testing from formal specifications
- Random testing

Decision Table

	Col 1	Col 2	Col 3	Col 4
Input 1	INV	VAL	INV	VAL
Input 2	INV	INV	VAL	VAL
Output	ERR	ERR	ERR	Login Success

INV - Invalid ; VAL - Valid ; ERR - Error





## **Instrumente de raportare si management al defectelor. Instrumente de management al testelor**

# **Test management tools. Tests reports and bugs management tools.**

Curs

- **Test session**

- Interval de timp in care se executa o suita de teste pe o versiune a produsului software; pe parcursul unei sesiuni de testare se pot crea test cases noi, verifica defecte, raporta noi probleme/imbunatatiri.

- **Test session reports**

- Zona testata.
- Note detaliate cu privire la tipul de testare efectuat.
- O lista cu toate defectele gasite.
- O lista de intrebari deschise.
- Toate fișierele utilizate sau create pentru a sprijini testarea.
- Mentiuni in ceea ce priveste timpul petrecut pentru executarea testelor vs. timpul petrecut pentru a investiga imbunatatiri ale produsului.
- Grafic cu timpul petrecut pentru:
  - Testarea - crearea si executarea testelor.
  - Investigare / raportare defecte.
  - Configurare sesiune de test sau alte activități non-testare.
- Momentul începerii și durata sesiunii de test.

# Metrici in testarea software

S.No.	Testing Metric	Data retrieved during test case development & execution
1	No. of Requirements	5
2	Avg. No. of Test cases written per Requirement	20
3	Total no. of Test cases written for all requirements	100
4	Total no. of Test cases Executed	65
5	No. of Test cases Passed	30
6	No. of Test cases Failed	26
7	No. of Test cases Blocked	9
8	No. of Test cases un executed	35
9	Total No. of Defects identified	30
10	Critical Defects count	6
11	High Defects Count	10
12	Medium Defects Count	6
13	Low Defects Count	8

## De ce avem nevoie de Test Case management tool?

- ne ajuta sa avem date despre calitatea produsului in timp real pe parcursul proiectului;
- sa fim organizati;
- sa putem refolosi teste;
- sa stim in ce directie mergem cu calitatea produsului din timp astfel incat eventuale decizii de corectare sa fie luate la momentul corect.

# Test case management tool - Caracteristici

- Gestionarea printr-o interfata web a test case-urilor si a sesiunilor de testare
- Versionarea si prioritizarea testelor
- Importarea/exportarea testelor
- Asocierea testelor cu defecte, cerinte si specificatii, si generarea automata a matricii de trasabilitate a cerintelor
- Planificarea de campanii de testare si definirea atributelor importante in aceste campanii: responsabil, selectie teste care urmeaza sa fie executate din baza de date, pe ce platforme se testeaza etc
- Baza de date pentru rezultatele campaniilor de testare
- Generare automata de rapoarte de testare in timp real
- Suport si integrare cu instrumente de testare automata. De exemplu pentru testarea de performanta se poate programa o suita de teste automate care sa inceapa la o anumita ora si sa intoarcă rezultatele direct in baza de date pentru rezultate.
- Definire de metrii de testare si includerea lor in rapoartele de testare
- Integrarea bidirectionala cu un sistem de bug tracking astfel incat atunci cand un test case este failed sa se poate deschide un bug direct din tool-ul de test management si actualiza in sistemul de bug tracking
- Integrarea cu serverul de email pentru trimiterea automata a raportelor si a notificarilor
- Integrarea cu alte servicii: continuous integration, release management

# Test case management tool

- Testlink
- Zephyr by Atlassian (add-on to JIRA)
- QA Complete
- Qtest by QASymphony
- Pivotal Tracker

# Test case management tool - Zephyr

Select Versions **0.4.6** + Create New Cycle

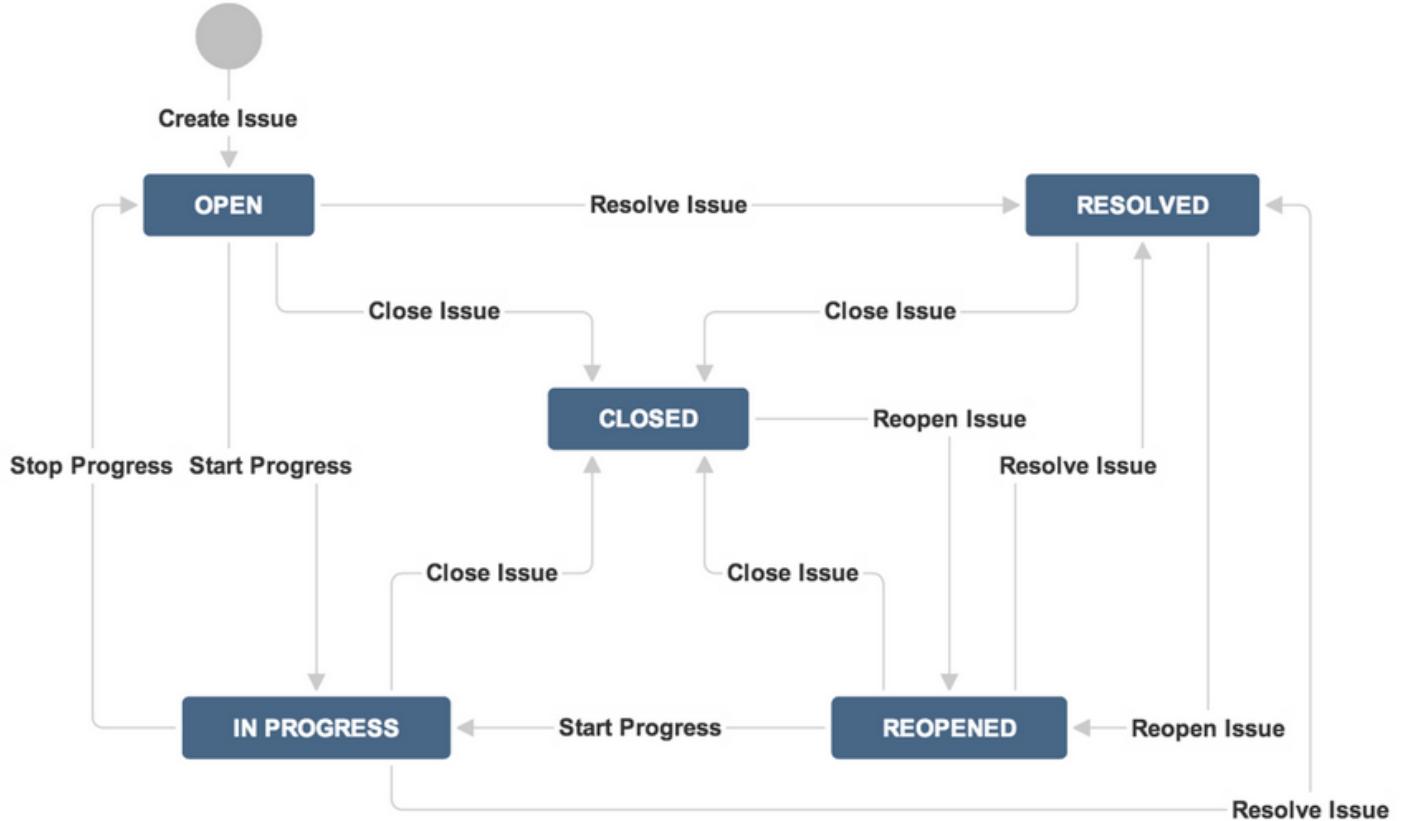
## My Account options

Starting On: 2015-08-16 Build: **0.4.6** Verify all functionalities implemented under My Account. 4 100% ⚙️

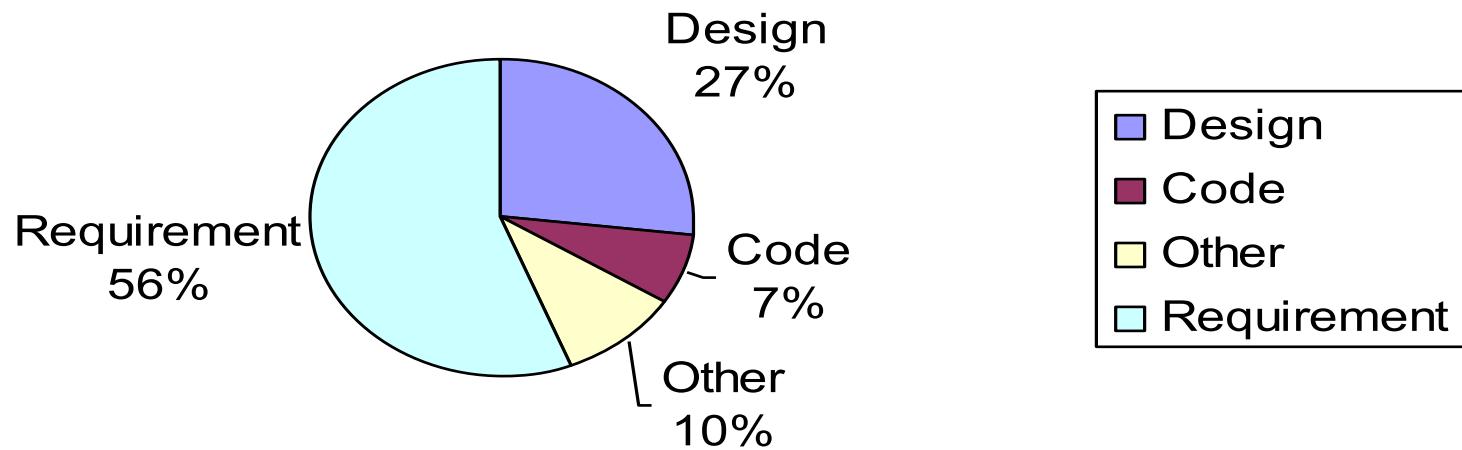
ID	Status	Summary	Defect	Component	Label	Executed By	Executed On	Action
KU-652	FAIL	User > Account Security page (change password)	[2] KU-916, KU-917	USER ACCOUNT	-	Alina Pacurar	08-19-2015 14:39:28	E
KU-653	FAIL	User > Privacy page	[2] KU-919, KU-920	USER ACCOUNT	-	Alina Pacurar	08-19-2015 14:56:18	E
KU-654	FAIL	User > Notifications	[2] KU-862, KU-863	USER ACCOUNT	-	Alina Pacurar	08-19-2015 15:20:11	E
KU-655	PASS	My account left side bar	-	USER ACCOUNT	-	Alina Pacurar	08-19-2015 15:28:19	E

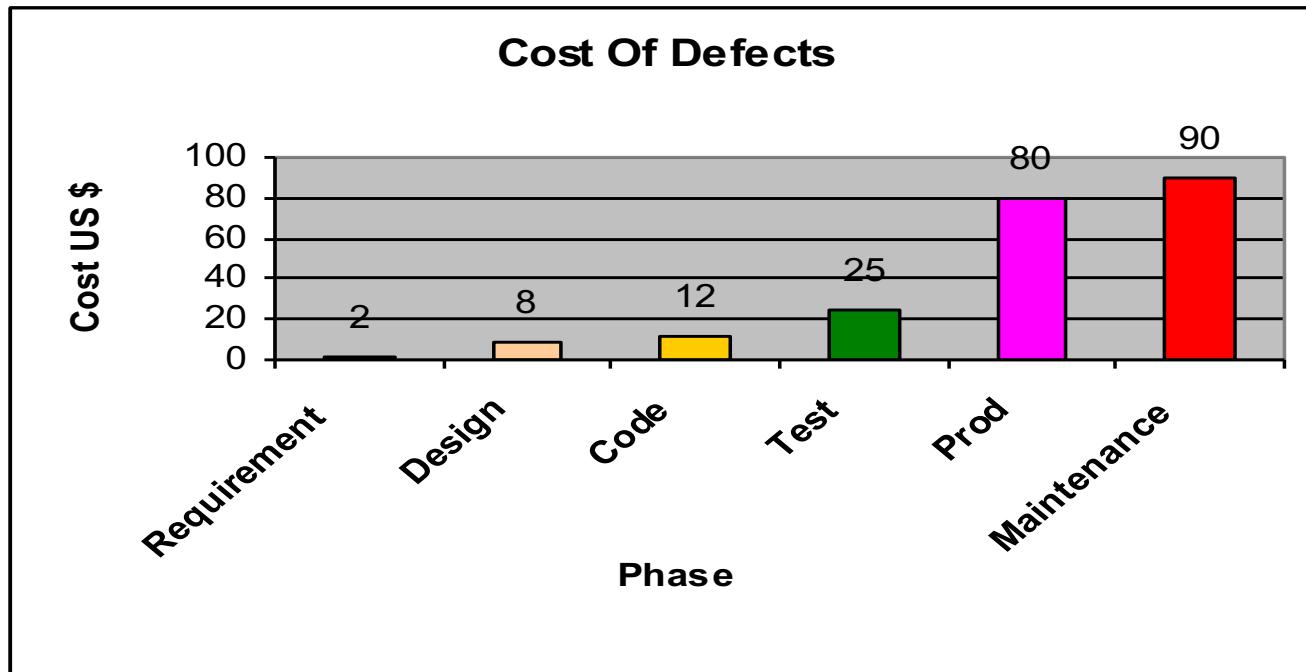
# Defect/Bug Management

- Defect Lifecycle – se defineste prin etapele pe care le parcurge defectul din momentul in care a fost gasit pana cand a fost inchis (rejected/fixed).
- Starile defectelor:

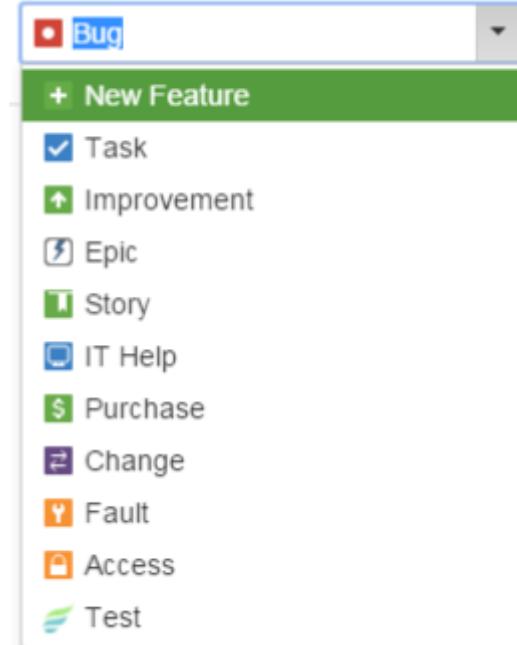


## Defect origination





- Varietatea este definită în funcție de cerințele proiectului
- Cele mai des întâlnite tipuri sunt: bug, improvement, new feature, task, sub-task, epic.



# Defect/issue management tools - Examples

- JIRA;
- Bugzilla;
- Mantis;
- Pivotal tracker;
- Redmine;



Web Properties / WB-4542

## ES: "Total refunded" is not updated in case of a full refund (TD)

[Comment](#) [Schedule Issue](#) [More](#) [Reopen Issue](#) [Admin](#)

## Details

Type: Bug  
Priority: Major  
Affects Version/s: Sprint 2015-12-22  
Component/s: Entitlement Server  
Labels: TD  
Environment: develop  
Story Points: 3  
Sprint: Sprint 2016-01-19  
Business Value: 2

Status: CLOSED (View Workflow)  
Resolution: Fixed  
Fix Version/s: None

## People

Assignee: Eitan Berkovich  
Reporter: Alina Pacurar  
Votes: 0  
Watchers: 2 [Stop watching this issue](#)

## Dates

Created: Dec/11/2015 3:34 PM  
Updated: Yesterday  
Resolved: Yesterday

## Agile

Active Sprint: Sprint 2016-01-19 ends Jan/21/2016  
[View on Board](#)

## HipChat discussions

Dedicated room: [Create a room](#) [Choose a room](#)  
Other rooms: Issue mentioned in 0 rooms

## Git Source Code

0 commits  
[Roll Up](#)

## Description

**Steps:**  
1. Go to [develop.totaldefense.com](http://develop.totaldefense.com)  
2. Add one product to the cart.  
3. Change the quantity to be greater than 1.  
4. Complete the checkout process.  
5. From TD Admin (ES) > CSR Manage Orders - search by email address of your user in order to see the latest subscriptions.  
6. Select one and enter Adjustment Mode.  
7. Full Refund it. See the Total Refunded field.

## Actual results:

Refund completes with success. Refund email is received by the user.  
Total Refunded field is not updated, continues to show "Total Refunded: \$0.00 USD".  
[http://devapi.totaldefense.com/Admin/CustomerService/OrderSummary.aspx?oid=A\\_S00007678\\_1](http://devapi.totaldefense.com/Admin/CustomerService/OrderSummary.aspx?oid=A_S00007678_1)  
user apacurar+visa@untangle.com /parola

## Expected results:

Total Refunded field is updated accordingly with the refunded amount.

## Attachments



55 kB

Dec/11/2015 3:34 PM

## Activity

[All](#) [Comments](#) [Work Log](#) [History](#) [Activity](#) [Emails](#) [Transitions](#) [Git Roll Up](#) [Git Commits](#)

▼ Alina Pacurar added a comment - Yesterday

Issue is not seen anymore. It was fixed during the refunds refactoring.

## Create Issue

Configure Fields ▾

Project\*  (eg. 3w 4d 12h) [?](#)

Issue Type\*  (eg. 3w 4d 12h) [?](#)

Summary\*

Priority  (eg. 3w 4d 12h) [?](#)

Due Date  (eg. 3w 4d 12h) [?](#)

Component/s  (eg. 3w 4d 12h) [?](#)

Component/s  (eg. 3w 4d 12h) [?](#)

Component/s  (eg. 3w 4d 12h) [?](#)

Affects Version/s  Start typing to get a list of possible matches or press down to select.

Fix Version/s  Start typing to get a list of possible matches or press down to select.

Assignee  (eg. 3w 4d 12h) [?](#)

Assign to me [Assign to me](#)

Reporter\*  (eg. 3w 4d 12h) [?](#)

Environment  Start typing to get a list of possible matches.

Description  (eg. 3w 4d 12h) [?](#)

For example operating system, software platform and/or hardware specifications (include as appropriate for the issue).

Original Estimate  (eg. 3w 4d 12h) [?](#)  
The original estimate of how much work is involved in resolving this issue.

Remaining Estimate  (eg. 3w 4d 12h) [?](#)  
An estimate of how much work remains until this issue will be resolved.

Attachment  [Drop files to attach, or browse.](#)

Labels  Begin typing to find and create labels or press down to select a suggested label.

Epic Link  Choose an epic to assign this issue to.

External issue URL  External issue URL

Priority  Priority

Status Whiteboard  Status Whiteboard

Browser  Browser

Report Type  Report Type

Architecture  Architecture

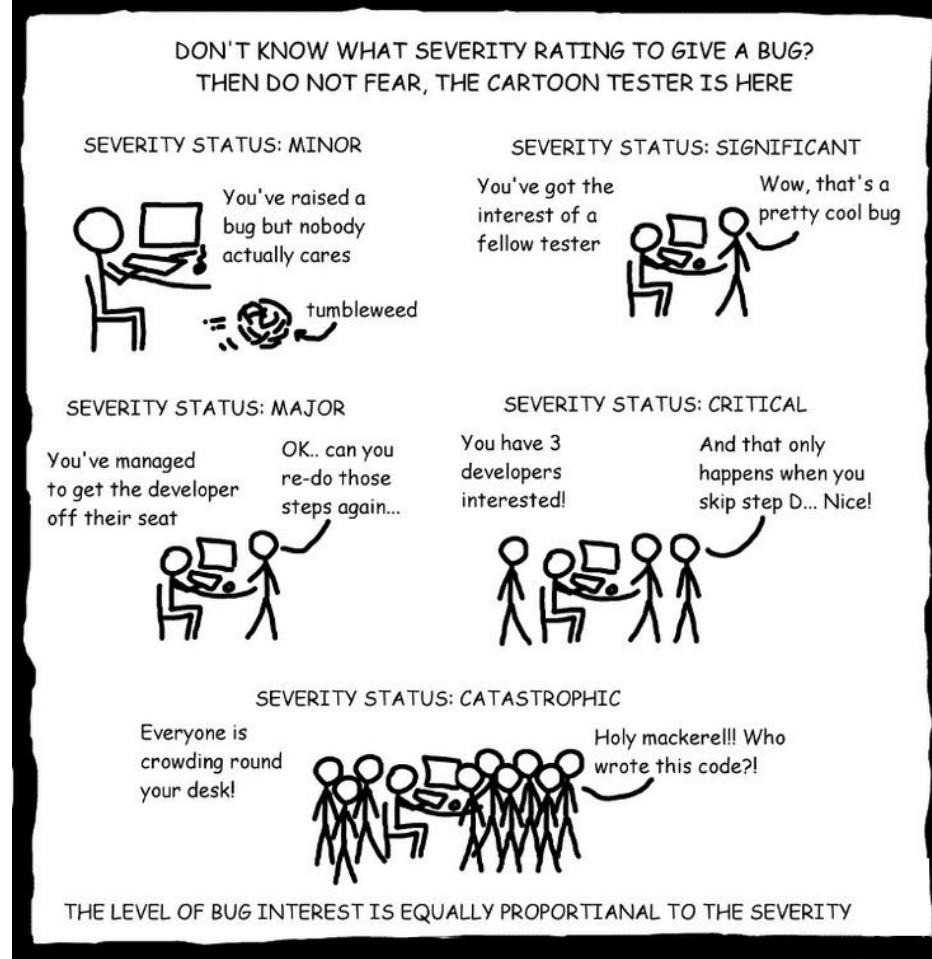
External issue ID  External issue ID

Story Points  Measurement of complexity and/or size of a requirement.

Sprint  JIRA Software sprint field

Affected Host  Enter the host name affected here

# Bug severity



# Defect/issue management - Examples

Create Issue

Project\*

Issue Type\*

Summary\*

Priority

Due Date

Component/s  
Start typing to get a list of possible matches or press down to select.

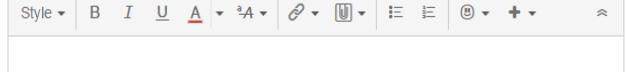
Affects Version/s  
Start typing to get a list of possible matches or press down to select.

Fix Version/s  
Start typing to get a list of possible matches or press down to select.

Assignee  [Assign to me](#)

Reporter\*

Environment  
  
  
For example operating system, software platform and/or hardware specifications (include as appropriate for the issue).

Description  


Create another  Cancel

Configure Fields  Sprint  Web Team Standup 

Show Fields: All | Custom [Where is my field?](#)

Affected Host	External issue ID
Affects Version/s	External issue URL
Architecture	Fix Version/s
Assignee	Labels
Attachment	Priority
Browser	Report Type
Business Value	Reporter
Component/s	Sprint
Description	Status Whiteboard
Due Date	Story Points
Environment	Summary
Epic Link	Time Tracking

# Defect/issue management – Exercitu practic



<https://jira.atlassian.com/projects/DEMO/>

(login cu Google account)

That's all folks!

## Feedback & Learnings & Questions



**Knowledge is the Most Beautiful of Awards**

