PROJECT QUALITY ASSURANCE

Module 10

Project quality assurance, standards, ISO 9000, CMMI model, ISO15504, Software project cases : version control, refactoring, unit testing methodology, testing plan, test cases, test automation, postmortem report

Lecturer. Tuyatsetseg Badarch, PhD. Associate professor

Mongolian National University

QUALITY OF THE CASE OF IT PROJECT

Quality means “conformance to requirements”
The best testers can only catch defects that are contrary to specification.
Testing does not make the software perfect.

If an organization does not have good requirements engineering practices then it will be very hard to deliver software that fills the users’ needs, because the product team does not really know what those needs are.

Lecturer. Tuyatsetseg Badarch, PhD. Associate professor

https://www.zarantech.com/blog/quality-assurance-project-management/

SOFTWARE QUALITY ASSURANCE (SQA) STANDARDS

Software quality assurance (SQA) is a process which assures that all software engineering processes, methods, activities and items are monitored and comply against the defined standards. These defined standards could be one or a combination of any like **ISO 9000, CMMI model, ISO15504,** etc.

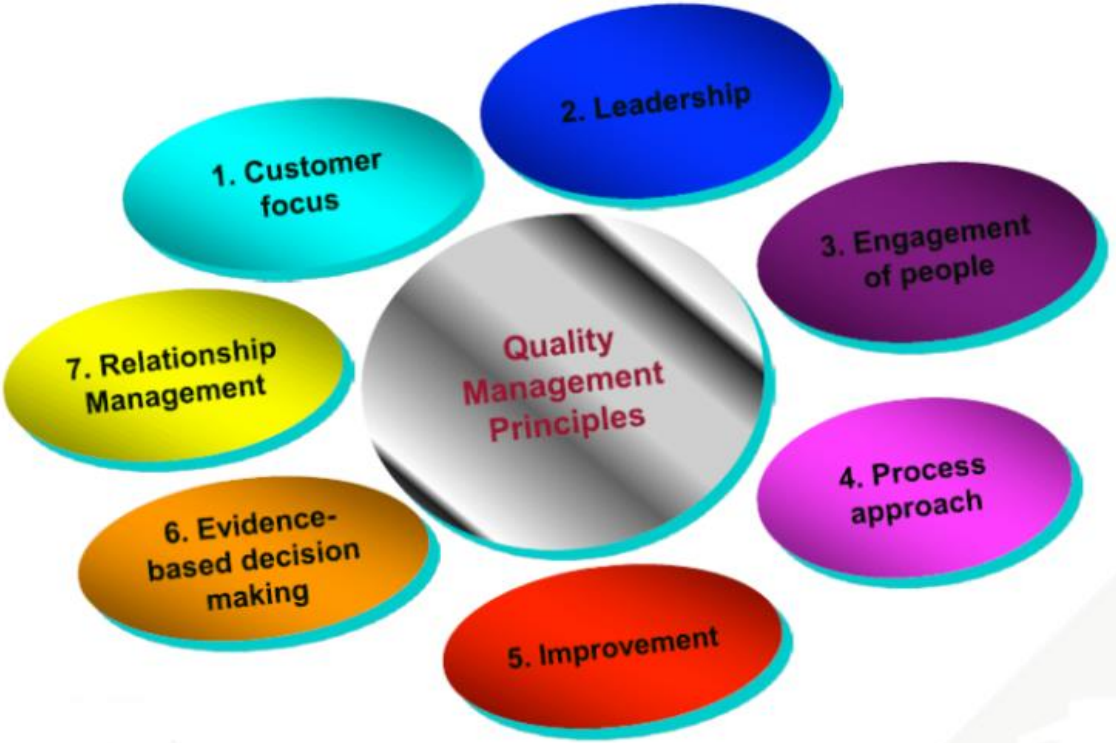
Lecturer. Tuyatsetseg Badarch, PhD. Associate professor

https://sixsigmastudyguide.com/cause-and-effect-matrix/

7 PRINCIPLES OF ISO 9000

ISO 9000: based on seven quality management principles.

These all help the project/organizations to ensure that their products or services are aligned with the customer needs.



Lecturer. Tuyatsetseg Badarch, PhD. Associate professor

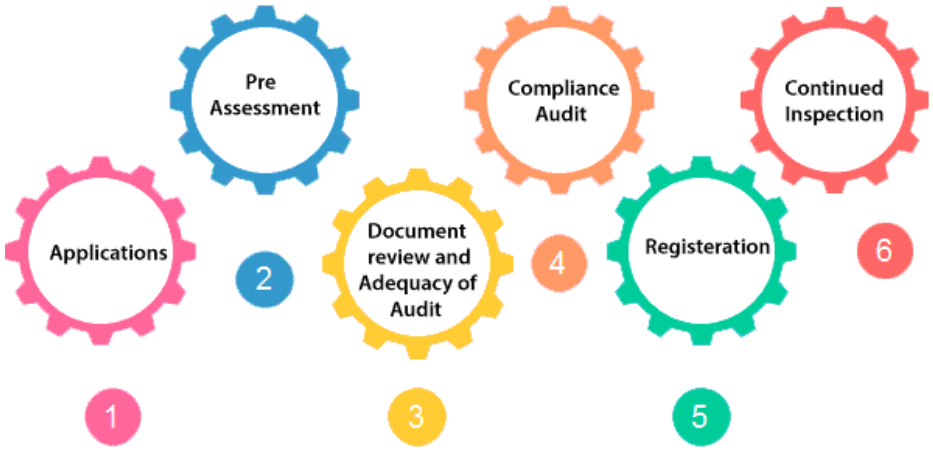
https://www.softwaretestinghelp.com/software-quality-assurance/

ON QUALITY STANDARDS

Quality assurance (QA) is a way of preventing mistakes and defects in project.

QA Models and standards:

ISO 9000 defines as "part of quality management focused on providing confidence that quality requirements will be fulfilled"



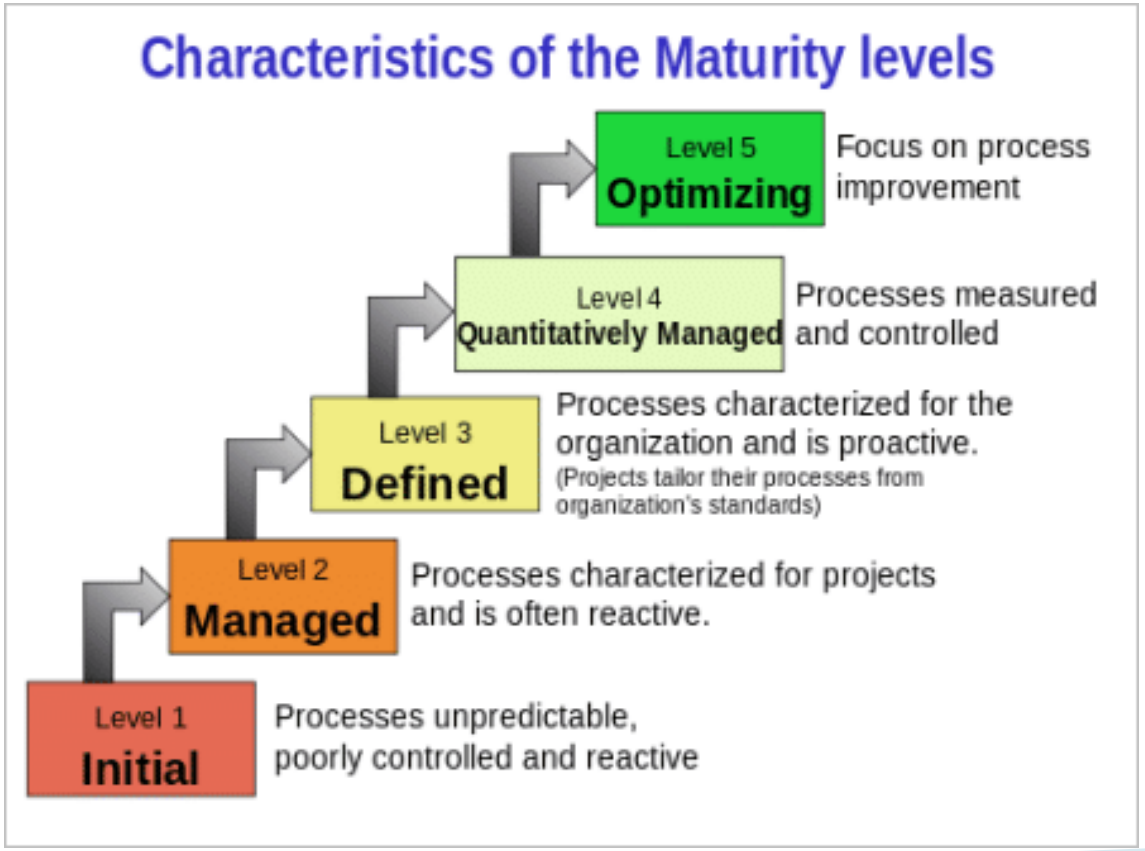
Lecturer. Tuyatsetseg Badarch, PhD. Associate professor

https://www.javatpoint.com/software-engineering-iso-9000-certification

PRINCIPLES OF CMMI

CMMI level:

Capability Maturity Model Integration was originated in software engineering.
It can be employed to direct process improvement throughout a project, department, or an entire organization.



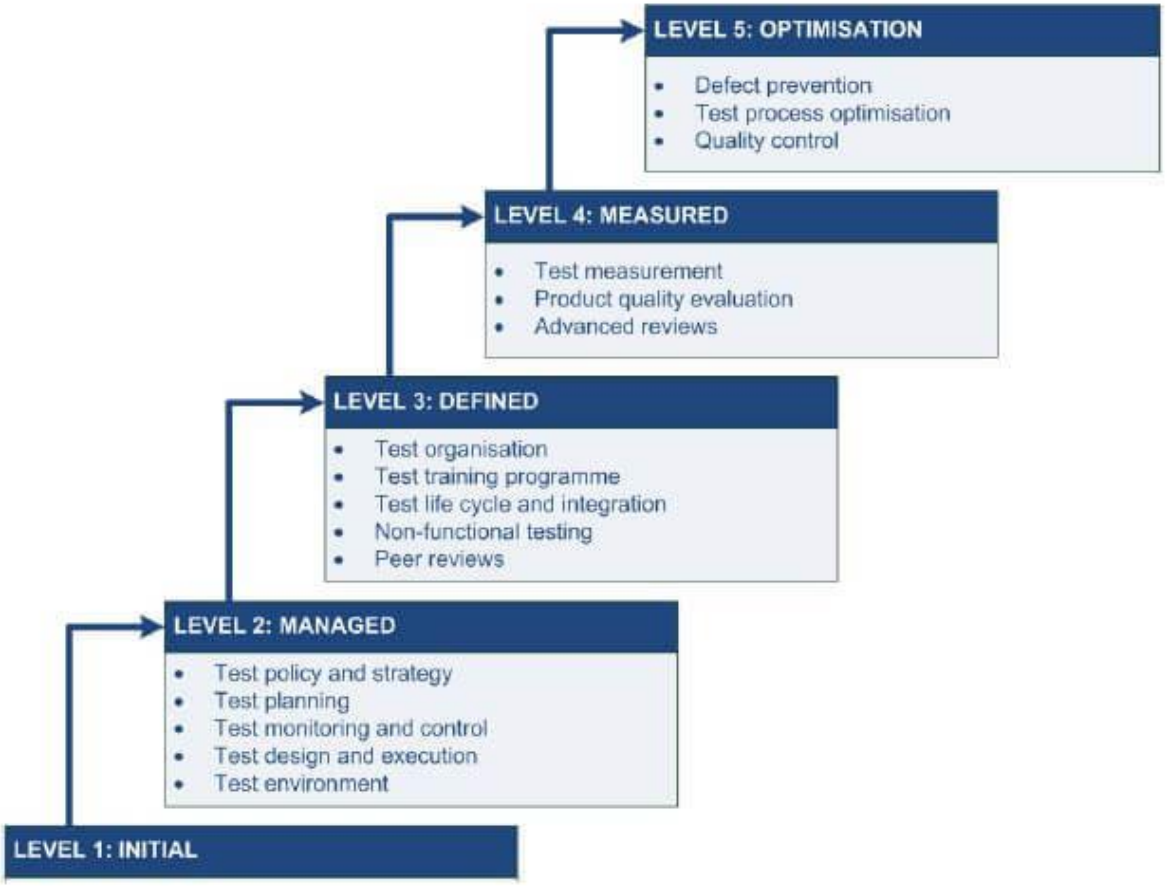
Lecturer. Tuyatsetseg Badarch, PhD. Associate professor

https://www.softwaretestinghelp.com/software-quality-assurance/

PRINCIPLES OF TMMI

TMMi :

achieves a higher capability for producing high-quality products with fewer defects and closely meets the business requirements.



Lecturer. Tuyatsetseg Badarch, PhD. Associate professor

https://www.softwaretestinghelp.com/software-quality-assurance/

QUALITY ASSURANCE (QA) PROCESS

A process-based approach

Lecturer. Tuyatsetseg Badarch, PhD. Associate professor

https://www.zarantech.com/blog/quality-assurance-project-management/

TOOLS FOR QUALITY ASSURANCE PROCESS

Four main tools

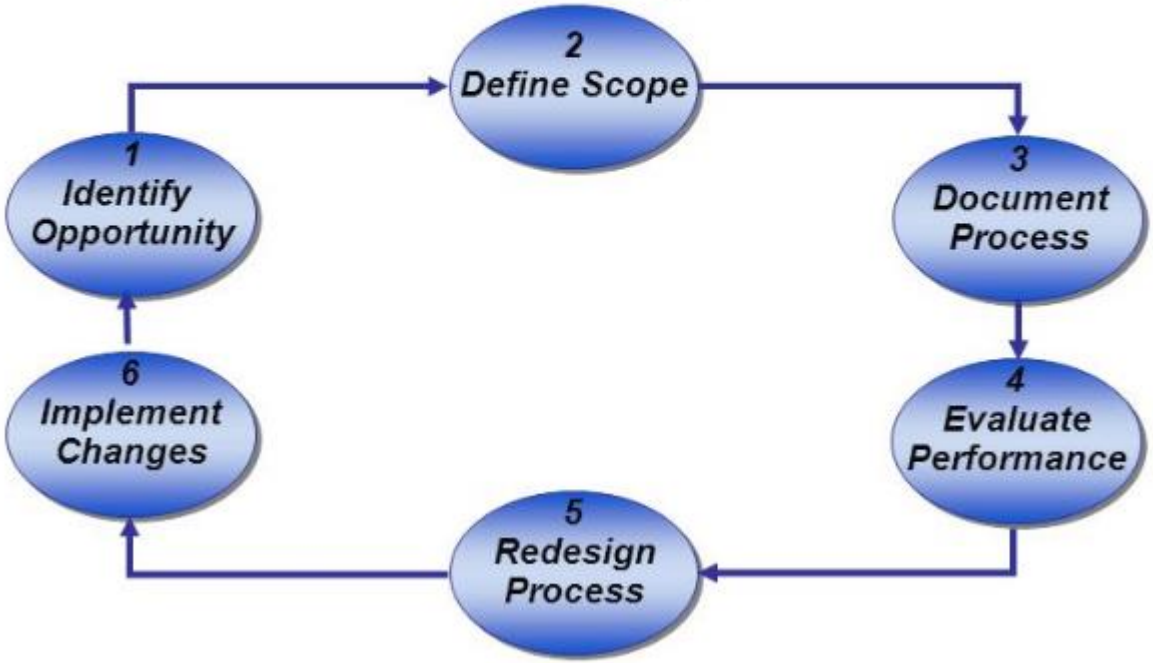
- 1. Process analysis
- 2. Quality audit
- 3. Quality management
- 4. Quality control

Lecturer. Tuyatsetseg Badarch, PhD. Associate professor

https://www.zarantech.com/blog/quality-assurance-project-management/

PROJECT QUALITY ANALYSIS

- **Process analysis** is the documentation and detailed understanding of how work is performed and how it can be redesigned.

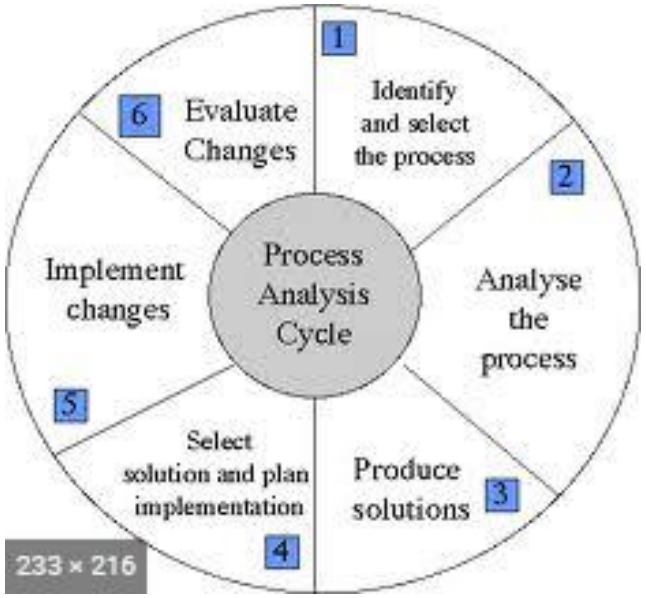


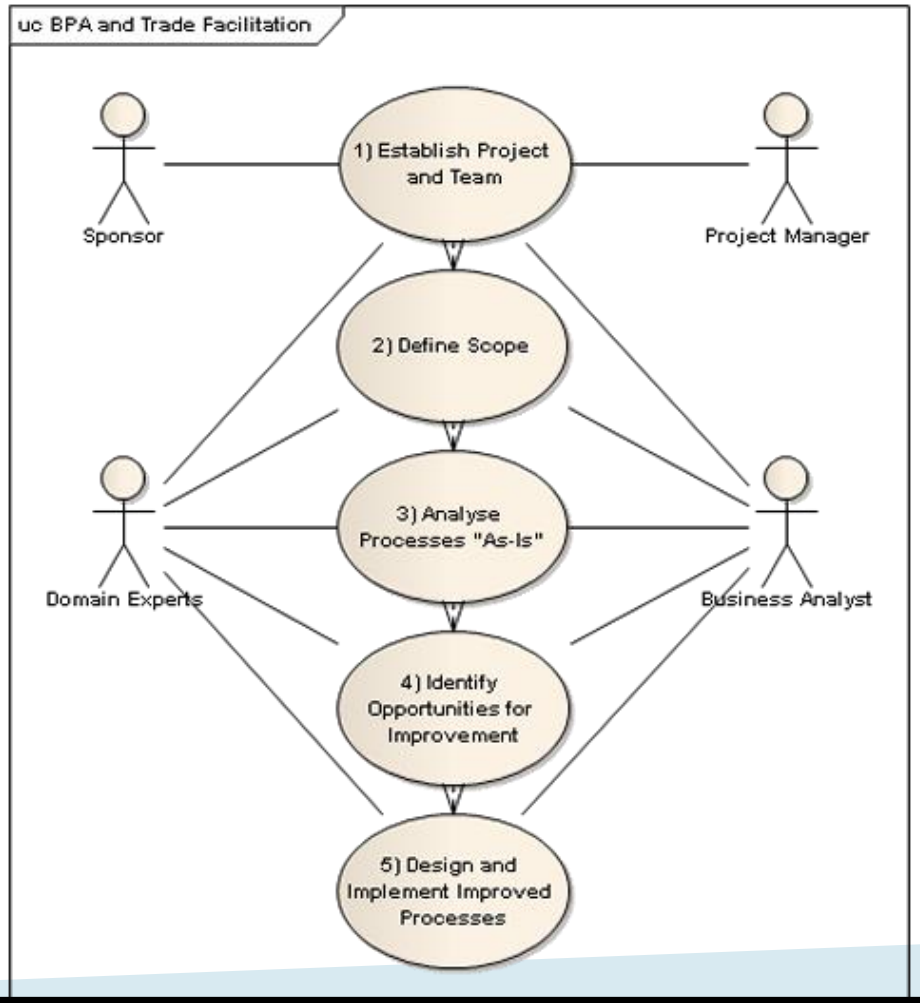
Lecturer. Tuyatsetseg Badarch, PhD. Associate professor

https://decode.agency/article/quality-assurance-vs-quality-control/

https://digitaltransformationpro.com/data-quality-simple-6-step-process/

QUALITY ANALYSIS





Lecturer. Tuyatsetseg Badarch, PhD. Associate professor

https://www.papermasters.com/process-analysis.html

https://digitaltransformationpro.com/data-quality-simple-6-step-process/

QUALITY AUDIT PROCESS

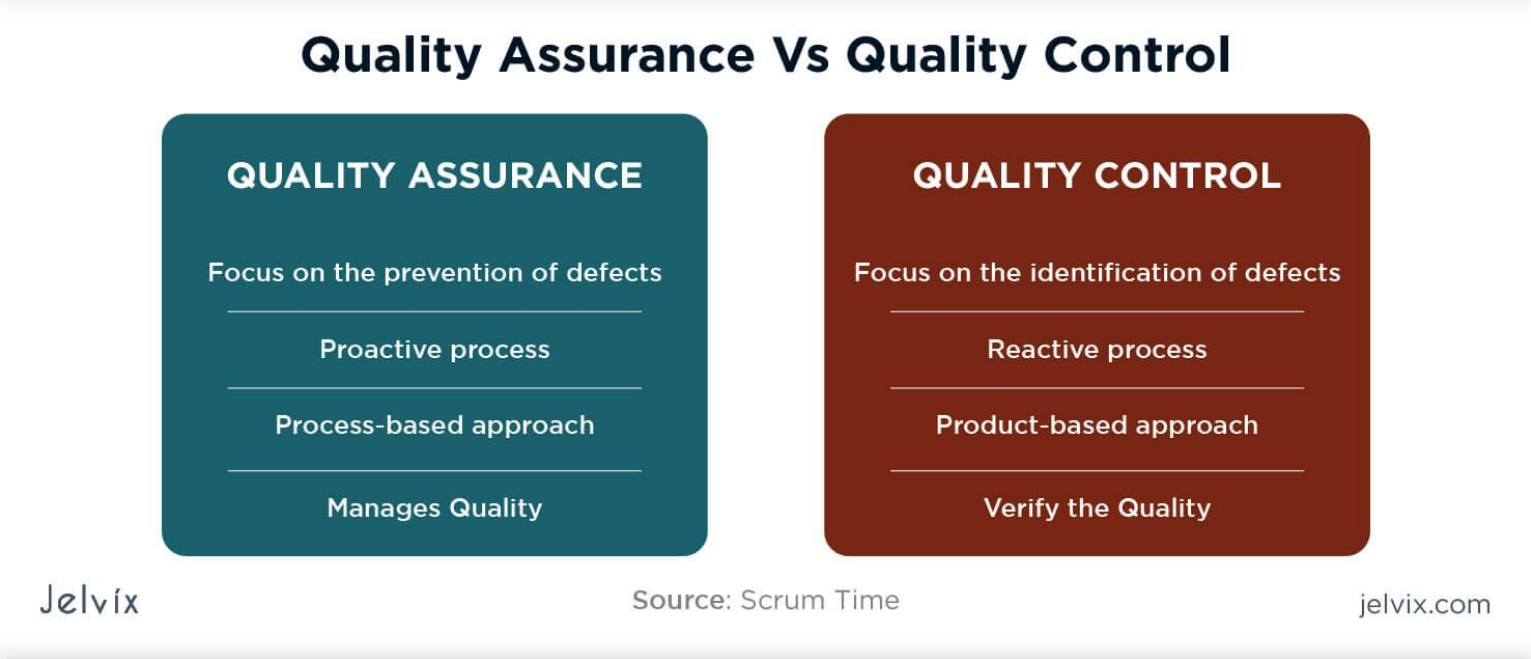
- To show the missing or inefficient policies, procedures and/or processes
- To review quality metrics such as defect frequency, budget deviation, failure rate, on-time performance, and gauge project activities against the quality baseline.

Lecturer. Tuyatsetseg Badarch, PhD. Associate professor

https://www.zarantech.com/blog/quality-assurance-project-management/

https://www.bdo.com/insights/assurance/corporate-governance/2018-delivering-on-our-audit-quality-intent/delivering-on-our-audit-quality-intent

DIFFERENCE BETWEEN QUALITY ASSURANCE AND QUALITY CONTROL



Tools: cost-benefit analysis, control charts, cause and effect diagrams and design of experiments.

Tools: management and technical reviews, inspection (code reviews) and testing.

Lecturer. Tuyatsetseg Badarch, PhD. Associate professor

https://decode.agency/article/quality-assurance-vs-quality-control/

https://decode.agency/article/quality-assurance-vs-quality-control/

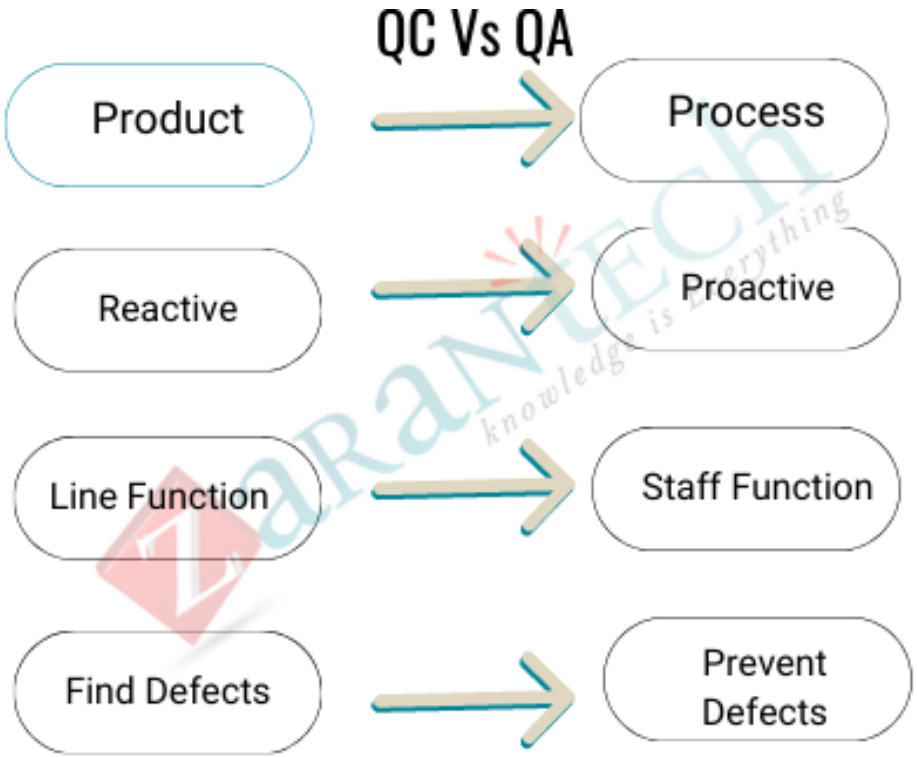
BENEFITS OF QUALITY ASSURANCE AND QUALITY CONTROL

The Benefits of QA + QC

- High quality output
- Eliminate waste
- Increase the operation efficiency
  Improve customer satisfaction
- Reduce rework

then

- Save a lot of money
- Promote confidence
- Motivate team



Lecturer. Tuyatsetseg Badarch, PhD. Associate professor

https://decode.agency/article/quality-assurance-vs-quality-control/

https://decode.agency/article/quality-assurance-vs-quality-control/

PROJECT QUALITY ASSURANCE TECHNIQUES: CAUSE/EFFECT MATRIX

Step 1: First identify the customer requirements (voice of customer). Based on conducting surveys, focus groups, ways to collect their priorities. *(in next slide; at the top of the X-Y diagram)*

Step 2: Assign priority factor (1-10) for each of *the customer outputs.*

Step 3: List *all possible key input variables* or the improvement factors of the process in each row, the Xs in the X-Y diagram

Step 4: Assess the relationship between key input variables (step 3) to the customer outputs (step 2) and rank each *input variables (Xs)* accordingly. Relationship is based on geometric progression scale (0,1,3 and 9), 0 being no impact, 1-low impact,3-medium impact and 9- Input has strong impact or correlation on output.

Step 5: Cross *multiply the customer output priority numbers with correlation rankings parameters and sum the each row at extreme right column*

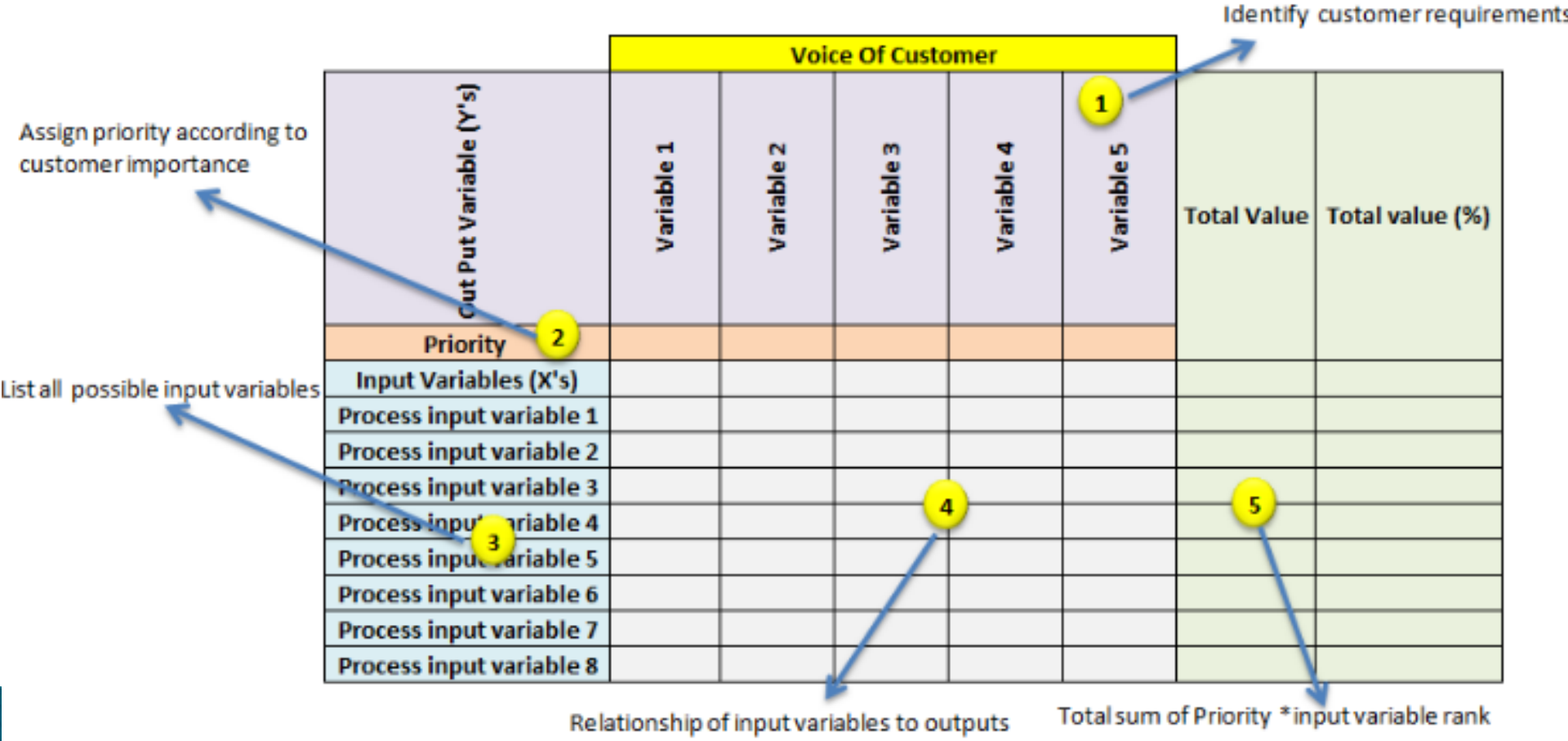
Step 6: Finally, determine *the rank*.

Lecturer. Tuyatsetseg Badarch, PhD. Associate professor

https://sixsigmastudyguide.com/cause-and-effect-matrix/

PROJECT QUALITY ASSURANCE TECHNIQUES: CAUSE/EFFECT MATRIX

Place those priorities in at the top of the X-Y diagram



Lecturer. Tuyatsetseg Badarch, PhD. Associate professor

https://sixsigmastudyguide.com/cause-and-effect-matrix/

AN EXAMPLE OF AN X-Y DIAGRAM OF CAUSE/EFFECT MATRIX

XYZ coffee shop franchise at San Diego downtown area reporting sales de-growth for the past 6 months. Six sigma team conducted root cause analysis and also they want to see the key process inputs that are impacting the process.

1. key customer requirements

	Voice Of Customer				
Out Put Variable (Y's)	Taste	Price	Delivery time	Aroma	Ambience
Priority (Output Rating)	9	6	3	3	4
Input Variables (X's)					
Coffee Flavor	9	3	3	9	1
Coffee Quantity	1	3	3	3	1
Brew time	3	0	9	1	1
Coffee Creamer	9	3	0	9	0
Machine Temp.	1	1	9	1	0
Filter	1	0	0	1	0
Beans freshness	9	1	1	9	1

2. Assign priority factor for each of the customer outputs.

3. List all possible key input variables

4. Assess the relationship between key input variables to the customer outputs and rank variables

5. Cross multiply the customer output priority numbers with correlation rankings and sum the each row

Lecturer. Tuyatsetseg Badarch, PhD. Associate professor

https://sixsigmastudyguide.com/cause-and-effect-matrix/

AN EXAMPLE OF AN X-Y DIAGRAM OF CAUSE/EFFECT MATRIX

XYZ coffee shop franchise at San Diego downtown area reporting sales de-growth for the past 6 months. Six sigma team conducted root cause analysis and also they want to see the key process inputs that are impacting the process.

1. key customer requirements

	Voice Of Customer					
Out Put Variable (Y's)	Taste	Price	Delivery time	Aroma	Ambience	Total Value
Priority (Output Rating)	9	6	3	3	4	
Input Variables (X's)						
Coffee Flavor	9	3	3	9	1	139
Coffee Quantity	1	3	3	3	1	49
Brew time	3	0	9	1	1	61
Coffee Creamer	9	3	0	9	0	126
Machine Temp.	1	1	9	1	0	45
Filter	1	0	0	1	0	12
Beans freshness	9	1	1	9	1	121
						553

2. Assign priority factor for each of the customer outputs.

3. List all possible key input variables

4. Assess the relationship between key input variables to the customer outputs and rank variables

5. Cross multiply the customer output priority numbers with correlation rankings and sum the each row

for coffee flavor = 9*9+3*6+3*3+9*3+1*4 = 139

Lecturer. Tuyatsetseg Badarch, PhD. Associate professor

https://sixsigmastudyguide.com/cause-and-effect-matrix/

AN EXAMPLE OF AN X-Y DIAGRAM OF CAUSE/EFFECT MATRIX

XYZ coffee shop franchise at San Diego downtown area reporting sales de-growth for the past 6 months. Six sigma team conducted root cause analysis and also they want to see the key process inputs that are impacting the process.

1. key customer requirements

3. List all possible key input variables

2. Assign priority factor for each of the customer outputs.

4. Assess the relationship between key input variables to the customer outputs and rank variables

	Voice Of Customer						
Out Put Variable (Y's)	Taste	Price	Delivery time	Aroma	Ambience	Total Value	Total value (%)
Priority (Output Rating)	9	6	3	3	4		
Input Variables (X's)							
Coffee Flavor	9	3	3	9	1	139	25.1%
Coffee Quantity	1	3	3	3	1	49	8.9%
Brew time	3	0	9	1	1	61	11.0%
Coffee Creamer	9	3	0	9	0	126	22.8%
Machine Temp.	1	1	9	1	0	45	8.1%
Filter	1	0	0	1	0	12	2.2%
Beans freshness	9	1	1	9	1	121	21.9%
						553	

5. Cross multiply the customer output priority numbers with correlation rankings and sum the each row

for coffee flavor = 9*9+3*6+3*3+9*3+1*4 = 139

The ranks based on the highest sum total and highlight the critical few variables.

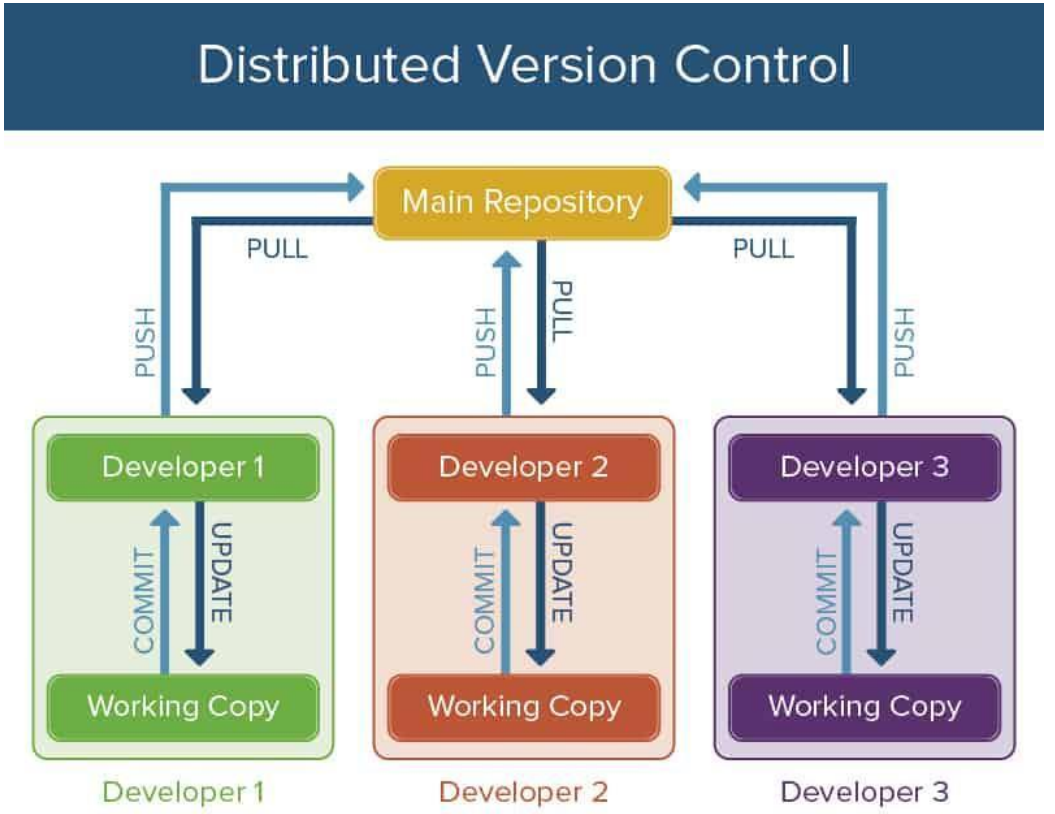
Lecturer. Tuyatsetseg Badarch, PhD. Associate professor

https://sixsigmastudyguide.com/cause-and-effect-matrix/

THE ROLE OF QA-VERSION CONTROL IN THE SOFTWARE DEVELOPMENT LIFE CYCLE

In a copy-modify-merge system, multiple people can work on a single file at a time.
When a programmer wants to update the repository with his changes, he retrieves all change.

In a lock-modify-unlock system, only one person can work on any file at a time.
A programmer must check a file out of the repository before it can be modified.



Lecturer. Tuyatsetseg Badarch, PhD. Associate professor

https://www.smartsheet.com/software-version-control

SOFTWARE QUALITY ASSURANCE (SQA) PLAN

SQA plan is for review and audit:

- Purpose section
- Reference section
- Software configuration management section
- Problem reporting and corrective action section
- Tools, technologies and methodologies section
- Code control section
- Records: Collection, maintenance and retention section
- Testing methodology

Lecturer. Tuyatsetseg Badarch, PhD. Associate professor

https://www.softwaretestinghelp.com/software-quality-assurance/

THE ROLE OF QA IN THE SOFTWARE DEVELOPMENT LIFE CYCLE

Lecturer. Tuyatsetseg Badarch, PhD. Associate professor

https://www.slideshare.net/adil.mughal/quality-assurance-in-sdlc

TEST PLAN FOR QUALITY ASSURANCE

The main work product is the test plan.
The test plan documents the overall approach to the test.
The goal of test planning is to pass all of the required requirements in the software.
It shows how the tests will be organized, and outlines all of the testers' needs which must be met in order to properly carry out the test. In many ways, the test plan serves as a summary of the test activities that will be performed.
The test plan should be inspected by members of the engineering team and senior managers.

Lecturer. Tuyatsetseg Badarch, PhD. Associate professor

https://www.testmonitor.com/qa-testing-best-practices-a-guide-for-software-qa-professionals

TEST PLAN FOR QUALITY ASSURANCE

The main work product is the test plan.
The test plan documents the overall approach to the test.
The goal of test planning is to pass all of the required requirements in the software.
It shows how the tests will be organized, and outlines all of the testers' needs which must be met in order to properly carry out the test. In many ways, the test plan serves as a summary of the test activities that will be performed.
The test plan should be inspected by members of the engineering team and senior managers.

Lecturer. Tuyatsetseg Badarch, PhD. Associate professor

https://www.slideshare.net/adil.mughal/quality-assurance-in-sdlc

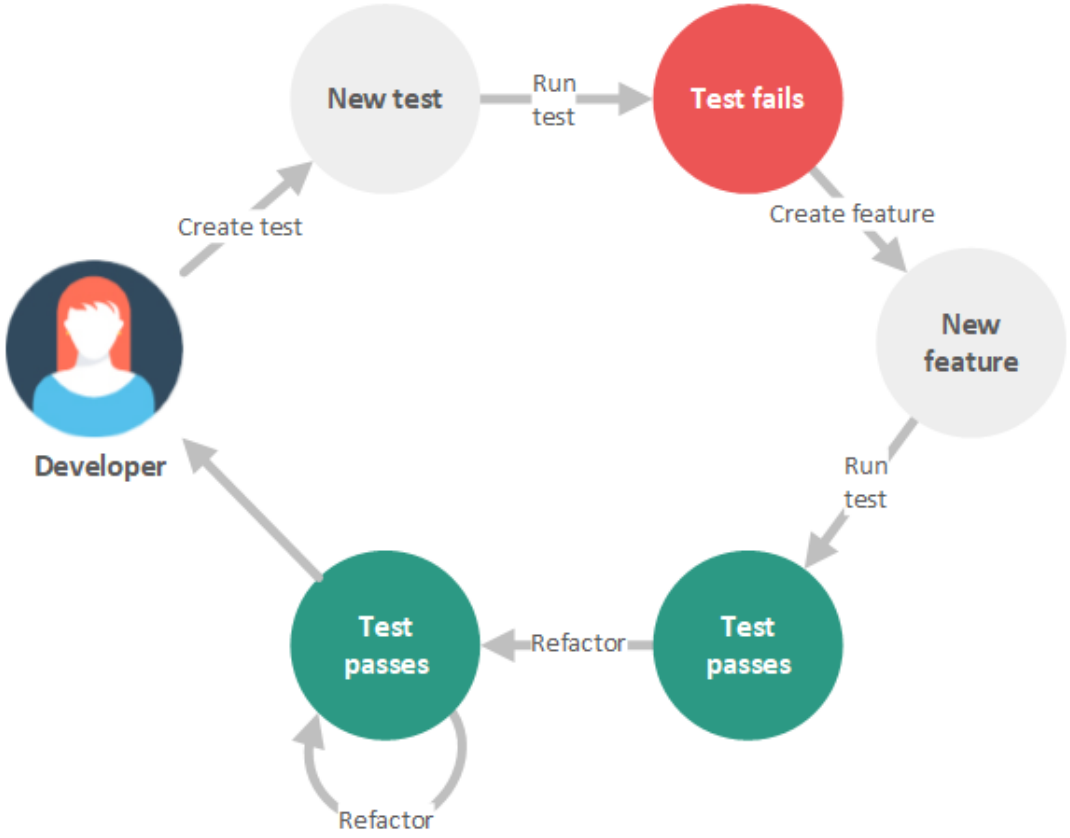
REFACTORING FOR QUALITY ASSURANCE

Refactoring does not affect the behavior of the software.

Refactoring works especially well during *code reviews.*

Because refactoring is a change to the design, it impacts the design review.

If previously reviewed code is refactored, changes to that should be distributed to the review team.



Lecturer. Tuyatsetseg Badarch, PhD. Associate professor

https://www.smartsheet.com/software-version-control

DEFECT TRACKING FOR QUALITY ASSURANCE

The defect tracking system is a special program that testers/programmers use to record and track defects.
The report based on a software usually contains

Defect_ID – Unique identification number for the defect.
Defect Description –information about the module in which Defect was found.
Version – Version of the application
Steps – Detailed steps along with screenshots with which the person can reproduce the defects.
Date Raised – Date when the defect is raised
Detected By – Name/ID of the tester who raised the defect
Status – Status of the defect , more on this later
Fixed by – Name/ID of the developer who fixed it
Date Closed – Date when the defect is closed
Severity level – the impact of the defect on
Priority – fixing urgency

Lecturer. Tuyatsetseg Badarch, PhD. Associate professor

https://freshworks.io/implementing-quality-assurance-in-software-development-lifecycle/

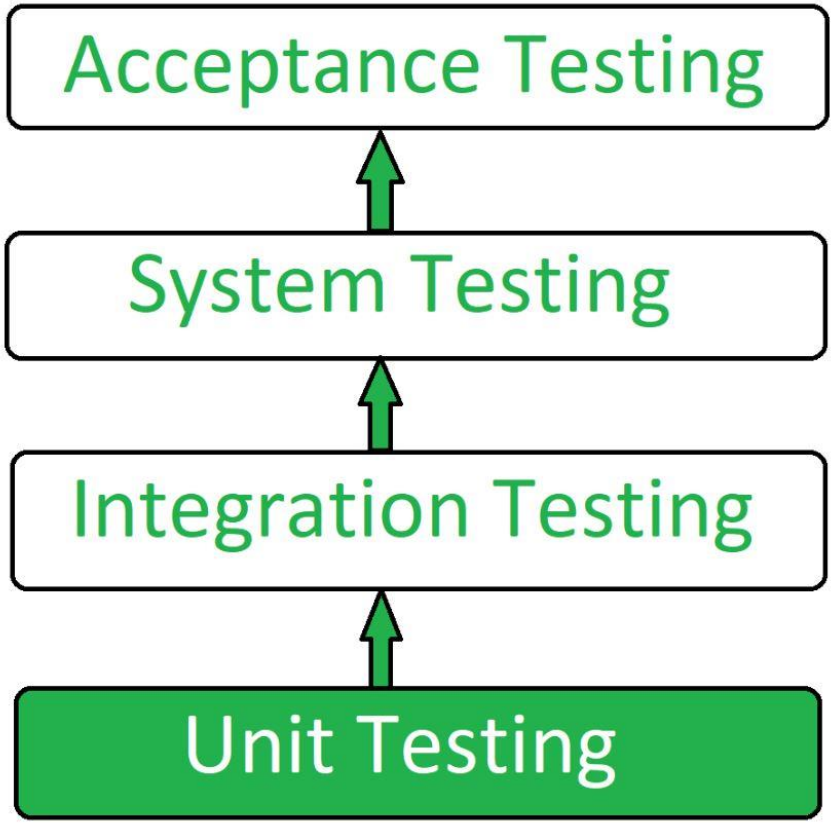
UNIT TESTING FOR QUALITY ASSURANCE

Before a build is delivered, the programmers/testers should execute unit tests to verify that each unit functions properly.

Unit testing is to create a set of tests for each unit to verify that it performs its function correctly.

Each unit test is a small block of code which exercises a specific behavior of one unit.

The effective way for programmers to do unit testing is to use a framework or a piece of software that automatically runs the tests and reports the results.



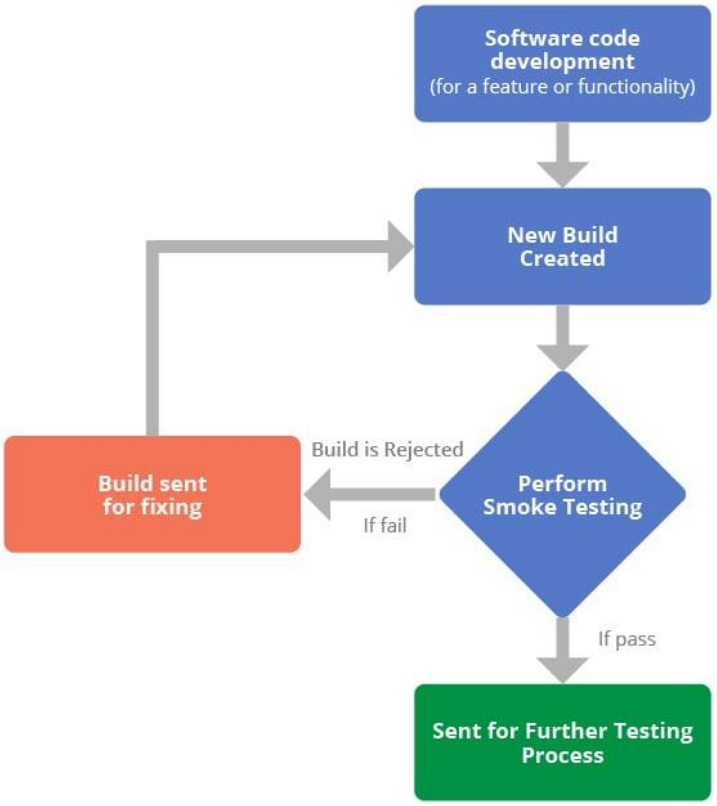
Lecturer. Tuyatsetseg Badarch, PhD. Associate professor

https://freshworks.io/implementing-quality-assurance-in-software-development-lifecycle/

SMOKE TEST FOR QUALITY ASSURANCE

A smoke test is a subset of the test cases that is typically representative of the overall test plan.

Smoke tests are good for verifying proper deployment or other non invasive changes.



Lecturer. Tuyatsetseg Badarch, PhD. Associate professor

https://freshworks.io/implementing-quality-assurance-in-software-development-lifecycle/

TEST AUTOMATION FOR QUALITY ASSURANCE

Test automation is a test case in which testers/programmers employ a special software to eliminate repetitive tests.

This can save the testers a time if many iterations of testing will be required.

Quality Assurance does not only relate solely to the “testing” of software, but also includes tests:

- Writing test cases
- Testing Bugs
- Regression Testing
- Preparing bug reports
- Unit tests
- Refactoring and more tests, mainly

Lecturer. Tuyatsetseg Badarch, PhD. Associate professor

https://freshworks.io/implementing-quality-assurance-in-software-development-lifecycle/

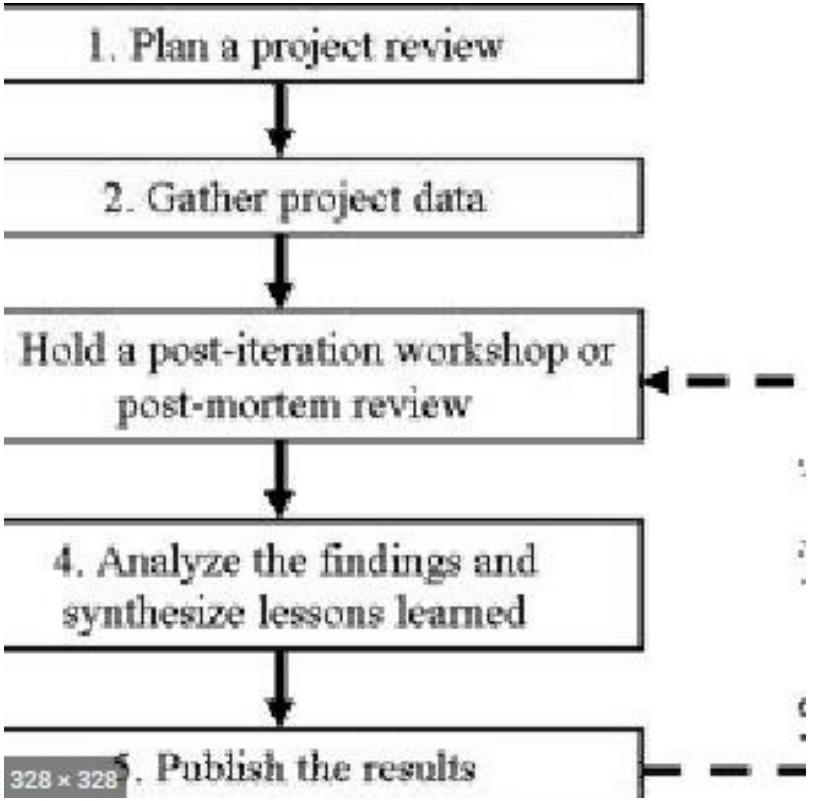
POSTMORTEM REPORTS FOR QUALITY ASSURANCE

The postmortem report is an overall account of the team's experience in building the IT project or software project, and of the experience of the users and stakeholders in working with the team.

The report should contain

- an assessment of how the team members, users, and stakeholders perceived the end product
- assessed the decisions made throughout the project.

The purpose of the post-mortem report is to highlight the team's successes and identify any problems which should be fixed in future releases.



Lecturer. Tuyatsetseg Badarch, PhD. Associate professor

https://freshworks.io/implementing-quality-assurance-in-software-development-lifecycle/

Thank you

Lecturer. Tuyatsetseg Badarch, PhD. Associate professor