Introduction

Manual Testing	Automation
1. Takes more time	1. Takes less time
2. Cheaper if executing 1,2, or 3 times	2. Execution is faster
3.Execute Manually First	3. Don't automate anything not linked or completed
4. Non-stable build	4. Executing more than 3 times

SDLC Models

Waterfall (for small project)	-Requirements Gathering -Designing -Implementation -Testing -Deployment -Maintenance	ADV: -simple -easy to follow	DIS: -slow;time consuming -deliver a wrong product -teams don't interact -design/implement to meet requirement -software delivered to next team

SDLC Models (cont)

V-Model	-Test in all phases	ADV: -continuous testing -Usable -Doable	DIS: -if you do not test correctly;wrong product
Iterative (small version of agile)	Repitition		
Spiral	Planning, Risk Analysis, Evaluation, Engineering	-unit testing -create a prototype of each design	-keeps iterating -takes more time -complicated to understand

SDLC Models (cont)

Agile (Iteration Detail Sprints 1-2 weeks)	-Implementation/Developer Testing -QA/Acceptance Testing -(Deployment) -Evaluation/Prioritzation -Detailed Requirements -Design/Analysis	-won't outdat -fast;ad -4 itera

Stages of SDLC (Ask for Access)

1. Requirement Gathering	-what does th client want -analyze,proc - feasibile? h much time?
2. Designing (Architecture; How it will look)	-reliable -estimate time,cost,res analysis w/ te challenges?

Stages of SDLC (Ask for Access) (cont)

6. Maintenance	-enhancements -bug fixes -new feature additions -Acceptance Testing (alpha/beta) -Alpha:done by testers; go to the csutomer's,install ,test it -Beta: Live testing

Software Development Life Cycle

- ⌨ A process to develop quality software
- ⌨ A process we follow to develop and deliver
- ⌨ SDLC is a process used to design, develop, and test high quality software
- ⌨ Continuous Process
- ⌨ Meets or Exceeds customer's expectations
- ⌨ Reaches completion within times and cost estimates
- ⌨ Assures that we are
    - 💻*Meeting the requirements*
    - 💻*Deliver on time*
    - 💻*Should be quality*

Software Testing

- 📱 process of watching with the requirements
- 📱 check for correctness,completeness,and quality
- 📱 if it is missing something it is a bug
- 📱 1)Validation 2)Verification
    - 💻requirement doc
    - 💻business doc
    - 💻software
- 📱 meets requirements
- 📱 works as expected
- 📱 can be implemented with the same characteristics

STLC (Software Testing Life Cycle)

1.Requirement	-properly documented/updated	-verification
2. Design Test Strategy	-how to test -what to test? tools?framework? required resources	-how will you decide what is going to be done
3.Test Planning	-who will test -resources/schedule/TC	-what you are going to test in each phase
4. Test Case Development	-write TC	

STLC (Software Testing Life Cycle) (cont)

5. Enviornment Setup	-software -hardware	-server to deploy testing environment
6. Automation	-automate TC test the software	-automate 1st execute later
7. Test Execution	-execute tests	
8. Report Defects	-report defects -log defect against automation	-tell m-report defects -log defect against automationanager of defects
9. Regression Test	-defects fixed	-what's fixed or not
10. Test Reports	-create and submit	

Principles of Software Testing

Principle 1	-prove there are defects (presence of defects)
Principle 2	-100% testing (exhaustive is impossible)

Principles of Software Testing (cont)

Principle 3	-Early Testing	
Principle 4	-Defect clustering; categorize based on severity, highest priority,needs	
Principle 5	-Pesticide Paradox	-dependencies
Principle 6	-Testing is context dependent	-log defects -what you're testing for
Principle 7	-Absence of errors-fallacy	-chances of missing the errors

Bugs Can Be...

Defect	Incident	Inconsistency
Fault	Anomaly	Product Anomaly
Problem	Variance	Product Incidence
Error	Failure	Feature

Verification & Validation/QA/QC

Verification	-is the requirement proper	-are we on the right track? -static documents	-audits
Validation	-building the right thing	-dynamic	-whether we have built it right -system testing
Quality Assurance	-process oriented -assure quality	-STLC -implement process	-preventive activities
Quality Control	-Product-oriented-	-corrective process -test cases	-one product -corrective process

Levels of Testing

Unit Testing	-check code	-lines of code	-# of TC
Integration Testing	-all the pieces fit when integrated		
System Testing	-Functionality	-end-to-end testing	-performance, accessibility, compatibility

Levels of Testing (cont)

User Acceptance Testing	-Alpha:lab, what's in production	-Beta: Production(end user)

Testing Techniques (Black Box)

- Don't have to understand/touch code
- No access to code
- Check functional and non-functional
- Advantages: efficient, unbiased, easy to execute, non intrusive
- Disadvantages: Localized testing, blind coverage, inefficient, test authority(system testing)

Testing Techniques (White Box)

- understand designs, programming language, unit test cases (execute and develop)
- Advatages: effective, early defect identifying, reveal hidden code flaws, full code pathway capable
- Disadvantages:Difficult to scale, cultural stress, highly intrusive, difficult to maintain

Testing Techniques (Gray Box)

- Mixture of black and white
- Test black box, partial access, can analyze their code, half rights

Testing Techniques (Gray Box) (cont)

- Advantages: Combined benefits, unbiased testing, intelligent test authority
- Disadvantages:Partial code coverage, Defect Identification

Testing Methodologies

Governance Testing	-services conform to standards, policies, and objectives	
Process Testing	-services are operating collectively as specified	-process being followed or not?
Unit Testing/Service-Component Level	-test the code	-functionality
System Testing	-business requirements defined and met acceptance criteria	-entire system works end-to-end

Testing Methodologies (c

Integration Testing	-if interface beha sharing btw the working as spec	
Security Testing	-to ensure prote unauthorized so	
Service level Testing	-Functional testting, Performance Testing, Security Testing	-comp nsive stater of qu

By **mnp003**
cheatography.com/mnp003/

Not published yet.
Last updated 29th August, 2016.