Unit Test 1

Software testing and Quality Analysis

Shiva Saran
BE-B 20

Q2. A) Identify Problematic Area of SDLC

Problematic areas of SDLC are as follows:

- No time allocated for good design and architecture
- Code becomes unreadable and unmaintainable
- Code is disorganized limiting amount of software engineers and/or changes to be active at any time.
- Makes full and patch releases difficult or impossible without severe downtime or refactoring of code.
- Simple configuration or system administration requires a code change not variable/parameter change.
- Makes change extremely difficult – not scalable or object orientated.
- Duplication of solutions i.e. not sharing such as application and hardware servers
- Makes releases extremely difficult i.e. No thought how to release the project – to many manual processes in a release. Releases take several hours and even days. Build an object to large to get through the door.
- Lack of necessary documentation or to much unnecessary documentation
- No or very little enterprise development, testing, or security standards
- Non-technical staff getting involved in making technical decisions causes the two most common features in struggling and/or failing technology projects:
- Pressure from business to implement change fast. Management scheduling without understanding impact and risk
- Inability of IT to explain and the business to understand the intricate nature of technology projects.

Q2. B) Differentiate between Software tools and Techniques

Software Tools	Solving Techniques
Tools are an organizations analytical asset that assist in understanding a problem through data and try to indicate possible solutions.	Techniques indicate more about a process used in measurement , analysis and decision making during problem solving.
Quality tools are applied for solving problems faced by projects and functional terms while improving quality in organization.	Improving quality of products and services offered to customers requires methods and techniques of solving problems associated with development and processes used during their lifecycle.
Tools may be hardware/software and physical/logical tools.	It can be accomplished by both qualitative and quantitative methods but problem definition becomes easier when we put some measures.
Accuracy and speed of the tools is much higher compared to performing all transactions and calculations manually.	Qualitative problem solving refers to understanding a problem solution using only qualitative indexes such as high , medium ,low etc. depending on the something is improving from present status and so forth.
Decision support offered of the tool is independent of personal skills and there is least variation from instance to instance.	Quantitative problem solving requires specification of exact measures of exact measures in numerical terms such as the cost of the 32.5% during last quarter or time required to one product is reduced by 32 minutes.
Tools may mean more cost and time to learn and implement.	It may follow define, measure,monitor,control and improve cycle.

Q3. A) Define the following

1. *Error*

   When the system produces an outcome, which is not the expected one or a consequence of a particular action, operation, or course, is known as error.
   Error or mistake leads to a defect and usually raises due to various reasons. It may be system specification issue or design issue or coding issue, which leads to a defect. Error leads to defects and if the defect uncovered by QA leads to Failure.

2. *Defects*

   A software bug arises when the expected result don't match with the actual results. It can also be error, flaw, failure, or fault in a computer program. Most bugs arise from mistakes and errors made by developers, architects. Some common types of defects are Arithmetic Defects Logical Defects, Syntax Defects, Multithreading Defects, Interface Defects and Performance Defects.

3. *Failure*

   Under certain circumstances, the product may produce wrong results. It is defined as the deviation of the delivered service from compliance with the specification.
   Not all the defects result in failure as defects in dead code do not cause failure. Reason for failure could be due to Environmental conditions, which might cause hardware failures or change in any of the environmental variables; Human Error while interacting with the software by keying in wrong inputs or Failures may occur if the user tries to perform some operation with intention of breaking the system.

4. *Mistakes in Software*

   Mistake in Software is basically an human error made unintentionally during software development cycle. It leads to a defect and usually raises due to various reasons.

Q3. B) Explain term Customer is King

- External Customer - outside the organization, who pay for the project
  - End-user customers
  - Manufacturer (OEM) for suppliers.
- Internal Customer - people within your organization who receive your work
- In many situations, service providers have multiple clients/ customers and therefore find it useful to identify “core customers”
- A customer’s perception is their reality.
- It is easier to keep your customers happy than attract new ones.
- Complaints spread like wildfire on the internet.
- Without customers we do not have a business.
- Brands win or lose by how well they wow customer.
- TQM’s Customer Approach
  “the customer defines quality.”
  “the customer is always right.”
  “the customer always comes first.”
  “quality begins & ends with the customer.”