***Manual Testing – The Kiran Academy***

Syllabus:

* What is s/w testing and why it is needed?
* SDLC
* Black-Box testing
* White-Box testing
* Grey-Box testing
* Test scenarios
* Test case
* Test data
* Bug/defect reporting
* Bug reporting through- Mantis hub tool
* STLC
* Tracebility matrix
* s/w verification & validation
* QA/QC
* Static & dynamic testing
* Test data creation techniques:
* BVA
* EG
* EP

**Lecture Notes**

**GUI(Graphical User Interface):** Front end of application.

**Database**: You can say backend of application.

**Middleware:** Making interaction between GUI & Database possible.

Middleware

Database

GUI

* **Software project**: specially designed for particular/specific customer or organization considering there requirements.
  + **Eg-** Banking software for internal employees.
* **Software product:** s/w is designed considering market/global requirements for all people.
  + **Eg-** WhatsApp.
* **Web-based s/w:** Actually running on the server & which requires connectivity to run.
* **Desktop-based s/w:**
  + s/w which contains physical(.exe) file on your computer which you need to install to run the software on your localhost(computer itself) with or without connectivity.
* **Software Development Life Cycle(SDLC):**
  + Is a process followed for a s/w project, with in s/w organization. It consists of a detailed plan describing how to develop, maintain, replace & alter or enhance specific s/w.
  + **Phases:**
    1. Requirement gathering
    2. Analysis
    3. Design
    4. Coding
    5. Testing
    6. Delivery & maintenance

1. **Requirement gathering:**
   * Business requirements are gathered here.
   * This phase is focus of the product manager & stake holders. Meetings with manager & stake holders and users are held to determine the requirements.
   * Who is going to use the system?
   * How will they use the system?
   * What data should be the input into the system?
   * What data should be the output by the system?
   * Roles involved: Business Analyst(BA), System architects
   * Outcome: **System Requirement Specification(SRS)** document.
2. **Analysis:**
   * After requirement gathering these requirements are analyzed for their validity & possibility of developing the requirements in the system.
   * Requirement analysis is the most important & fundamental stage in SDLC. It is performed by both development & testing team.
   * Roles involved: Dev. & QA team, Architects & Project managers.
   * Outcome: Final **SRS** document approved by customer, technology selection for both dev. & QA.
3. **Design:**
   * During this part of design phase, the consultants/architects break down the system into pieces that can be programmed.
   * System design helps in specifying hardware & system requirements and helps in defining overall system architecture. System design specifications serve as input for the next phase of the model.
   * Roles involved: Architects & team.
   * Outcome: **Technical Design Document(TDD)**
4. **Coding:**
   * Actual coding/development starts, & the product is built in coding phase.
   * The work is divided in modules/units & actual coding is started in this phase & it focuses on developers.
   * Coding is one of the LONGEST phase of SDLC
   * Roles involved: Developers & architects
   * Outcome: Programs or Modules or Application.
5. **Testing:**
   * In this phase tester executes the test cases against the application, reports defect & retest the fixed defects.
   * During this phase unit, integration, system, user acceptance testing is done.
   * Roles involved: Testers & developers.
   * Outcome: Defects, test summary report, test plan, test case document.
6. **Delivery & Maintenance:**
   * Delivery: After successful testing the product is delivered/deployed to the server to make it live.
   * During the delivery phase, customer will perform User Acceptance Testing(UAT) in real time environment with tester.
   * Once when the customer start using developed system then the actual problems come up & need to be solved from time to time. This process where care is taken for developed product is known as Maintenance.
   * Roles involved: Testers, developers, customer, business team, architects, PM and delivery manager.

* **Black-Box testing:**
  + The technique of testing without having technical knowledge of an application is called black box testing.
  + Eg- Requirement in **Business Requirement Document(BRD):** After entering valid username and password welcome profile should appear.
  + It is also known as **Specification-Based Testing** technique.

Software/application

output

input

* **Advantages of black-box testing:**
  + Well suited & efficient for large code segments.
  + Code access is not required.
  + Clearly separates user’s perspective from developer’s perspective through visibly defined roles.
  + Large number of moderately skilled testers can test the application with no knowledge of implementation, coding & OS etc.
* **Disadvantages of black-box testing:**
  + Limited coverage since only a selected number of test scenarios are performed.
  + Inefficient testing because tester only has limited knowledge of an application.
  + Blind coverage, since the tester cannot target specific code segments or error prone areas.
  + The test cases are difficult to design.
* **White-Box Testing:**
  + The technique of testing with having technical knowledge of an application is white box testing.
  + It is also known as:
    - Clear box testing
    - Open box testing
    - Transparent box testing
    - Code based testing
    - Glass box testing
  + It is software engineering based on inner working(logic) of an application & revolves around internal testing.
* **Advantages of white-box testing:**
  + As the tester has knowledge of the source code, it becomes very easy to find out which type of data can help in testing the application effectively.
  + It helps in optimizing the code.
  + Extra LOC can be removed which can arise some hidden defects/bugs.
  + Due to tester’s knowledge about the code maximum coverage is attended during test.
* **Disadvantages of white-box testing:**
  + Since a skilled tester is needed to perform white box testing, cost is increased.
  + Sometimes it is impossible to investigate every nook & corner to find out hidden errors that may create problem after delivery.
  + It is difficult to maintain white box testing as the use of specialized tools like code analyzers & debugging tools are required.
* **Grey-Box testing:**
  + It is a technique of testing with limited test knowledge of internal working of an application.
  + Basically, it is combination of White-Box + Black-Box Testing.
  + Eg- Server errors, application logs, communication protocol etc.

Server error 500