**Testing techniques and types of testing**

**Testing Techniques:**

1. White box testing: internal structure of system is known. we run codes and can see them.
2. Black box testing : Internal structure of the system is not known. Just checking if its working.(a key opens a lock)
3. Grey box testing: partial knowledge of system is known .

May have some access to dB to update etc.

Internal structure, actors,complexisty

* **BVA(BOUNDARY VALUE ANALYSIS):**

Testing the peek conditions-> max and min conditions , inner and outer values, typical valus and error values.

* Done when there is time constraint.
* Eg: [15,26]
* 14, 15,16,25,26, 27
* **Equivalence class partitioning:**

If sample data exceeds the boundary values, then it is eq. class.

Age: -ve or above 200 etc : are random number- common knowledge to give i/p

6-digit passcode:

-2 different cases

-more than 6: error (>6)

-Exactly 6 digits: success

-less than 6 (<6)

* **Error guessing:**

Eg: age -ve

Space in mandatory field

Boundary value, equivalence partitioning

* **Decision table based testing:**

Uses combination of testing fields.

To submit what fields are t be filled -> name, email etc

* **State Transition**

E.g.: state transition of atm machine.

From start state to end state covering success and failure conditions and all the paths leading to success and failure.

* **Functional and nonfunctional testing:**
  + Features are tested by feeding the inputs.
  + Non-functional aspects of the system are checked.(eg: font sizes, color, reliability, security.
* **Smoke testing Sanity testing**
  + Smoke tests the build for a platform.
  + Sanity testing covers the functional requirements(eg: sending mail etc.)
* **Regression and Retesting**
  + Regression testing tests the fixed bugs along with the previous features implemented.
  + Retesting is testing only the fixed issues.
* **Alpha testing and Beta testing:**
  + Testing before the release is alpha.
  + After the release is beta.
* **Ad hoc testing and Exploratory testing**
  + Random testing- no reference to the test cases , without plan and documentation.
  + Explore the projects and identify the defects on the fly.

3 types: free styles, scenario based, strategy based.

* **Usability testing, Happy path testing, risk-based testing**
  + How usable- look, seed, performance etc.
  + Test on a positive flow-test only the positive scenarios.
  + Based on priority testing the functionalities-which has the highest impact on the business.(done when there is time constraint)-u still have risk of not testing remaining functionalities.
* **Context driven testing:**
  + Based on the context of the project.
  + Test cases must be written and reviewed. And new testers test only these after the review.
* **Performance testing**
  + Speed , responsiveness etc.
* **Recovery testing**
  + Recovery on failure
* **Security testing**
  + Checking for penetration by hacking techniques
* **Volume testing**
  + A huge volume of data is fed to check its peek condition.
* **Scalability testing**
  + Efficiency in scaling up/down etc.
  + Load vs stress: l load is gradual increase in load(checks performance), sudden increase in load(checks stability)
* **Quality assurance**
  + Defect prevention oriented- writing extensive test cases.
* Static testing hence a type of verification
* Reviews and audits – whatever u have written gives the proper coverage.
* **Quality control**
  + Defect detection
  + Dynamic testing
  + S/w testing
  + Part of STLC