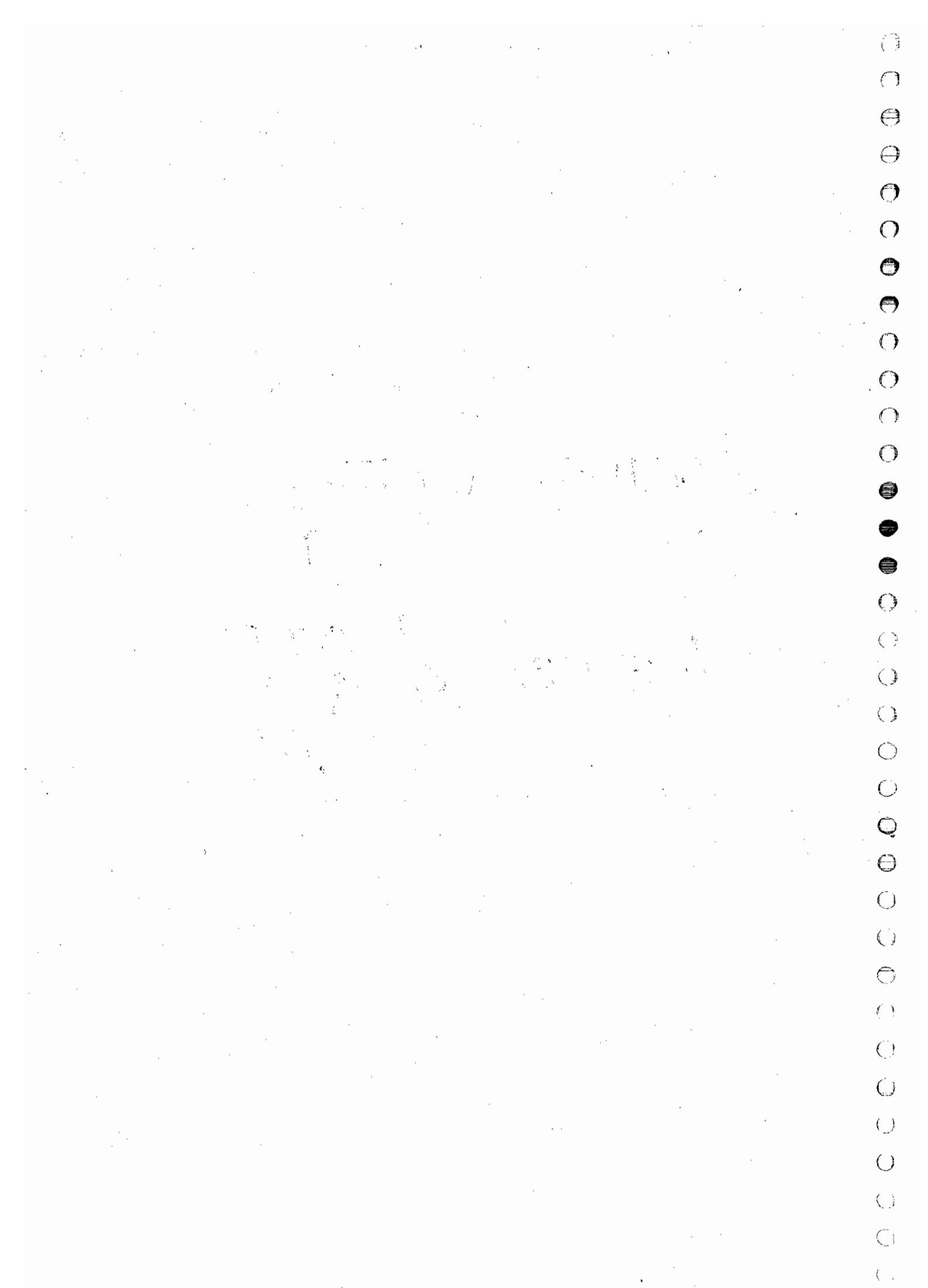


Software Testing?

Manual of QTP

Testings.



We come here for Sustainability of Job.

Q1. Why I choose Software testing ?

1. Testing is directly linked with Quality,
2. My Profile is suitable for testing.
3. Testing is not associated with BOOM & HYPE. It's a natural growth.

(Where the tolerance of defect will reduce whereas Quality of product will increase)

4. Testing is forever.
5. Testing is Mandatory for industry.
6. Testing is Independent of technology.
7. In testing there will be very good scope.

Scope means

- 1. Geological scope (Global scope)
- 2. Chronological scope (forever / All time)
- 3. Job scope - (100%) -

of tester must be :-

1. Quality Mind Setup.
2. Test to break attitude.
3. That person should be 3rd Person.
4. Internal Customer. (Act like customer)

Means:-

→ Does not know anything about the developer's Development -

10.02.2015

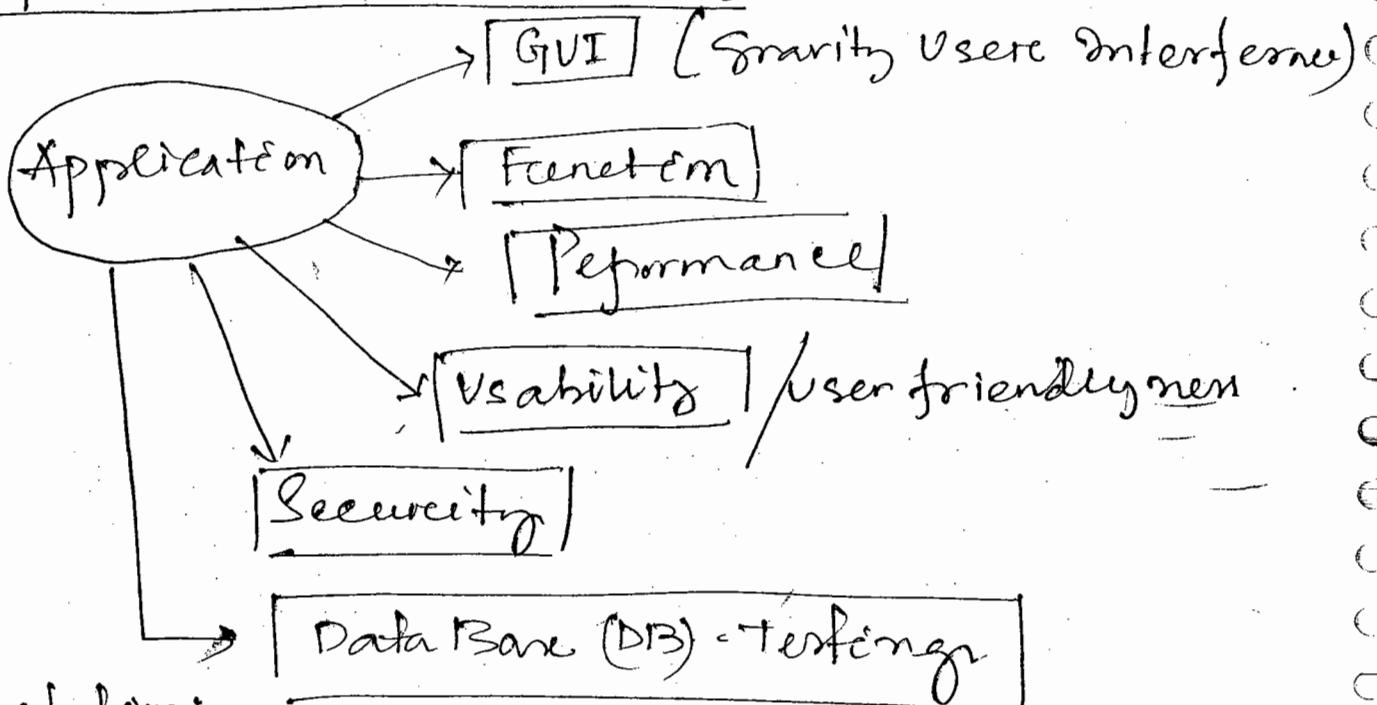
Testing Industry:- / Expectation of Industry



/ challenge

- ↑ ① Customer is more aware Quality con. in the Indus..
- ↓ ② Tolerance towards Defect decrease.
- ③ Evolution of new tech.
- ④ Complexity of application.
- ⑤ Integration of different tech.

Expectation of Industry:-



Expectation:-

* Domain Knowledge.

Health care / Medical / Banking / Finance

C. Commerce / Investors

* Automation Skill.

* ~~External~~ Internal Structural Knowledge (ISK)

i.e. C. Pm. 1

* Communication Skill.

* Analytical Skill.

→ Challenges that foreman in Industry:-

challenger face by the tester in Industry.

1. frequent change of technology. there are associated with various domain.
2. Comprehension ability on Customer Requirements.
3. Bug hunting skill i.e. Lacking.
4. Inability to deal with the counter Part.
5. Working under unpractical tight-
~~deadlines~~. Schedules.

Strategy:-

→ class Room training (converse) Right-Converse
(Must be learn proper sequence) Converse Right

→ Case study
→ Exercise.

→ Project Training.

Key factor - Job Sustainbility

→ Placement Training Program (PTP)

① How to answer Tech. Question.

② Explain the proj. ③ How to prepare for Interview

④ HR Question / How to answer HR.

⑤ Personality Development

Manual testing	QC Manage	QTP Test automation
----------------	--------------	------------------------

Dt. 11.02.2015

Quality:— It is defined as

Conformance to the specifications of the customer. In other words quality is defined as the justification of the requirement— of the customer so as to produce customer satisfaction.

NOTE: Quality is not a physical entity. It is purely psychological. That is the reason why a product that is liked by some customers or not liked by some other customers.

Derivation of quality:—

1. Defect:— It is defined as the deviation from requirements. It must be visible.
2. Error:— It is defined as the mistake that— associated with program.

Defect

1. It is visible to the user.
2. It is associated with functionality.
3. It is encountered by the customer/test engineer.

Error

1. It is invisible.
2. Error is always associated with the program.
3. It is encountered by the developer.

Defect

Error

4. Defect is an effect

4. Error is the cause.

3. BUG:- It is defined as the deviation from the requirement. Hence it is conceptually same as defect. ~~because~~ usage wise it differs from the defect. Defect is used by the customer as well as professionals, whereas as the term BUG is used by mostly the professionals.

MISTAKE/FAULT:-

It is defined as the deviation from the right way of doing the task.

5. ISSUE:- It is defined as the obstruction to the task/work flow.

6. FAILURE:- It is defined as its a temporary state of inability to reach the destiny of the task.

Work Around:-

As per as issue/failure come to role/we can always use alternatives to overcome them that known as "work around".

7. RISK:- It is a possible, Potential, adverse(cave) effect of the task.

Type of Risks:-

There are two types of Risk.

1. Known Risk

2. UnKnown Risk.

For the Known Risk there will be always other lot's of available. (in terms of contingencies) It the lot's available to nullify the risk.

Type of Mistakes:-

There are 2 types of mistake

1. Repairable mistake

2. Irreparable mistake.

Quality Gap:-

It's a missing quantity of Quality.

Quality gap can be understood properly from the point of vendors and customers.

NOTE:-

1. Vendors point of view - quality gap is the gap between the dream quality and delivered quality.

2. Customer's Point of view - Quality gap is the gap between the desired quality and delivered quality.

Note:-

From the quality gap every test engineer must understand the fact that he must delivered the product with better quality every time and next time while compared to the previous quality.

In other word there is no falling of quality gap.

Redefining the Quality:-

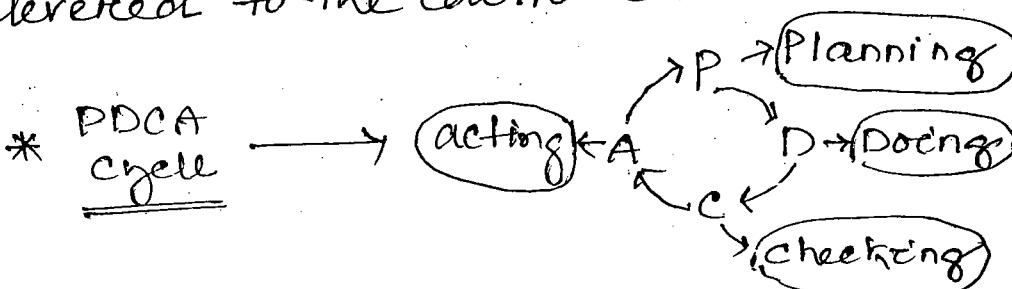
Quality is defined as not only presents of requirement or absents of defect but also presents of value. The term value is also known as usability which means user friendliness.

Note:- Every test engineer must understand that he must ensure usability to the product apart from best functionality.

→ Quality Policy:-

It is nothing but a document in which the benchmark level of quality is described so that every product has to maintain the same whenever it is delivered to the customer.

→ Diagram:-



→ Quality Plan:- It's basically a document in which the activities like planning the task, Doing the task according to plan, Checking the results and actioned upon the task with refinements or basically described by the management to ultimately achieve quality.

→ Cost of Quality:-

It can be understood in terms of the following various types.

1. Defect Removal cost (DRC)
2. Defect Appraisal cost (DAC)
DPC
3. Defect Preventive cost (DPC)

1. Defect Removal cost (DRC): It is the cost that is spent for removing the defect from the product where it is being used by the customer.

2. DAC :- It is the cost that ~~incurred~~ ^{uncovered} for removal of the defect from the product just before it is delivered to the customer.

3. DPC:- It is the cost that is spent by the management right from the beginning of the development to prevent the defect ^{screen} future.

Quality factors:

If this is the list of factors that can qualify the product as quality product. Hence there quality factors. Quality factor has described as follows.

1) correctness:-

In case all the requirements are justified the state of the product is known as correctness, that provides quality to the product.

2) usability:-

Apart from function perfection, if the product has user friendliness it is said to have usability which makes the product qualitative.

3) Testability:- If the product is examinable to encounter the defect and to make it defect free, the product is said to be testable that

C - correctness.

U - Usability

T - Testability.

R - Reliability

S - Scalability.

M - Maintainability

P - Portability.

makes the product Qualitative.

1) Reliability:-

It is the ability of a product to exhibit function perfection for longer durations.

Key :- [Perfection for longer duration]

2) Scalability:-

It is an ability of a product to be adaptable for future requirement so that the same product can be used in future with minimum customization.

3) Maintainability:-

It is the state of a product that allows rectification as well as enhancement depends on the need; If the product is maintainable it is said to be quality product.

4) Potability:-

Potability is the ability of a product to be compatible with multiple environment, the potability products are qualitative.

* How to achieve Quality? [DDD - Defect forofile Document]

Some processes will be done

1. Identification of defect.
2. Document the defect.
3. ~~Review~~ Subjecting defect for Rectification.
4. Ensure that the product is defect free.

Testing :-

It is basically a process in which defects are identified, documented, subjected for rectification and ensured that the product is defect free in order to provide quality to the product and hence customer satisfaction.

Test Policy:-

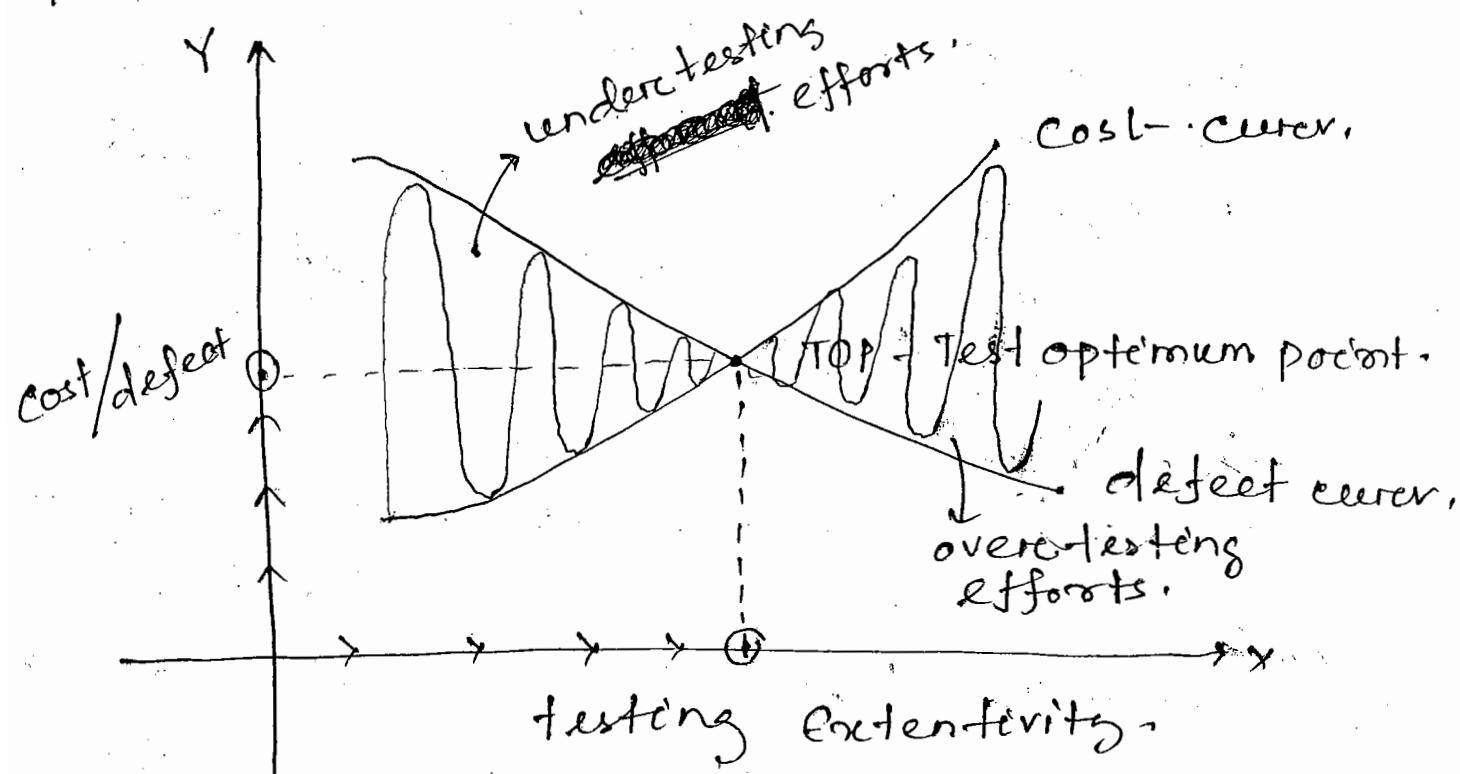
It basically a document that describes testing activities like test planning, test design, test execution, Result analysis, Test reporting etc to provide quality to the product.

Cost of testing:-

It is the cost that spent for the entire testing task. In fact testing is never ending process. Practically it has to be stopped at some critical point which is known as Test termination point "TOP" to determine

[TOP - Test optimality point]
where test stop.

definite budget for testing. Test optimum point can be determine by following graph.



from the above graph the following factors are retured.

1. Test optimum Point (TOP) be determine.
2. To determine where to stop testing
3. To determine the required margin budget for testing.
4. Under testing effort as well as over testing can be determine from the above graph.

Q/ hence exactly the testing comes in software project development life cycle?

SDLC - Software / System Development Life Cycle.

Project Origination / Project Acquisition / Pre SDLC

Process: —

Setting

the project is basically originated from the need of a customer. vendor must provide this solution to address the need of the customer in terms of the software application

VENDOR :- vendor is solution provider.

Software :- It is the set of program that can be executed to run the application.

Program :- It is the set of sequential statements that can be executed to get the desired output.

Application :- It is the set of program / program that can be executed to address the need of customer and to provide the required solution.

Project:

- It is an outcome prepared with the help of requirements that are coming from the customer outside.

Product:

1. It is an outcome prepared to the help of requirements that are originated from within the company.

NOTE:- Projects are delivered to specific customers whereas products are delivered to any/many customers, depends on the need.

RFP - (Request for Proposal):- It is the process in which customer request the vendor to send a project proposal for the development of the entire solution.

RFO - (Request for Quotation):- It is the process in which customer request the vendor to send ~~for~~ a quotation (~~&~~ tentative budget) for the development of the entire solⁿ.

Project signing:-

It is the process in which an official agreement is made betⁿ the customer & vendor in such a way specific project is

developed by specific vendor; with no specific margin; delivered by specific deadline etc are documented.

Apart from project signing maintenance cost (free of cost / Nominal cost / Market cost) must be mentioned in the document. So that the vendor has to offer maintenance service accordingly.

Statement of Work (SOW) :-

It is the process by which the scope of work is documented so that the vendor has to deliver exactly the same (Not less/ Not more than that). This document is known as Statement of work.

Kick off meeting:- It is the 1st meeting conducted with in the development company in order to do the following.

1. To discuss the overview of the project.
2. To review the profile of customer.
3. To Select project and development teams tentatively.
4. The leads- Project manager (PM) and Quality League (QL) are select for the development & testing team respectively.
5. The participants are steering committee members like MD/CEO/BD/HOO along with key managers.

like technical manager / software quality manager / HR manager.

PIN Release:- Once the PM is selected he sends an email to the CEO ask him for formal permission to start the project development activities.

Once He gets the approval he will start the project development activities with a Symmetric Scientific process "Software Development Life cycle".

PIN - Project Initiating Note

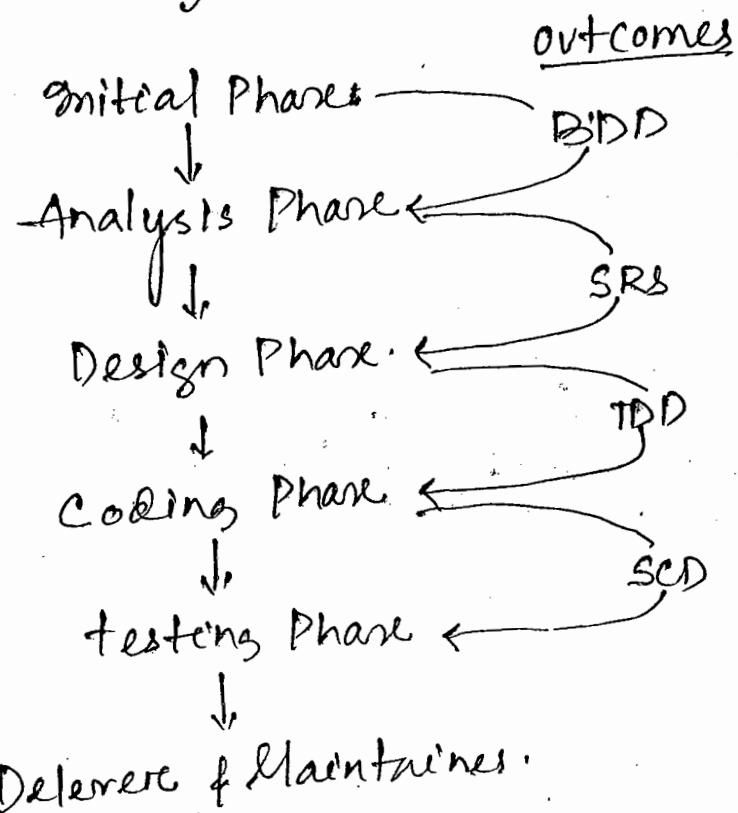
complete

Chapter - 2

SDLC:- (Software Development Life Cycle) :-

SDLC Phase:-

SDLC have the following Phases.



Each phase contains Responsible / Role/ process / outcome.

1) Initital Phase:-

1. Responsibility:- To gather the requirement of the customers.

2. Role:- Business Analyst (BA) & engagement manager (EM).

3. processes:-

- (i) To interact with customers & end users.
- (ii) To gather complete requirements.
- (iii) BA has to study the existing system.

(iv) To do the gap analysis.

(v) To highlight these gaps in terms of new requirements.

(vi) To document all the requirement.

4. Outcome:-

Having done the above responsibilities to ^{BA} BA prepareds the document known as "Business Design Documents (BDD)".

Template:- It's a proforma/format in which the required field are present for which the corresponding information can be given to prepare specific document.

NOTE:- Every role is provided with templates for the respective documents prepared by them within the organisation.

FRS - Function Requirement Specification.

BRS - Business Requirement Specification.

BDS - Business Design Specification.

Strategies:-

Engagement Manager:- (EM)

When ever the new requirement need highlighted by the BA it will be evaluated and a percentage can be given by EM hence he will be involved in monetary talks.

B] Analysis:- (High level Requirement - HLR)
System ~~Specifcation~~ Requirement - SR = SW + HW

1. Responsibility :- To convert High level Requirements in to System Requirements.

2. Role :- System Analyst.

3. Process:-

(i) To review the BDD in order to understand the requirements.

(ii) To do the feasibility study - what requirements can be implemented & what can't be.

(iii) To create functional and nonfunctional requirements under software requirements;
Configuration detail under hardware requirements.

(iv) Technology selection will be done under this phase.

(v) Project plan will be ~~earlier~~ laterly started during this analysis phase.

(vi) To collect the entire system requirements and to document that-

4. Outcomes:- SRS - System Requirement Specification

The outcome of analysis phase

is System Requirement Specification SRS

Contents of SRS:-

SRS contains.

S/W Requirements.	
Functional :	<ul style="list-style-type: none"> (1) Login screen (2) Entry of username (3) Entry of password. (4) Log in (5) Home page & P
Non-functional.	<ul style="list-style-type: none"> (1) Login screen must be app. quickly once app. is launched (2) Log in activity \leq 10 sec.
H/W Req: (Hardware Requirement)	Windows OS - Win7/Win8/Vista Server - Window 2003/2008 Configuration - I5 / Ram 4GB / HD 500GB.

c] Design:-

1) Responsibility:

To create a blueprint for the project development.

2) Role:- Chief Architect. (CA) and Technical lead (TL)

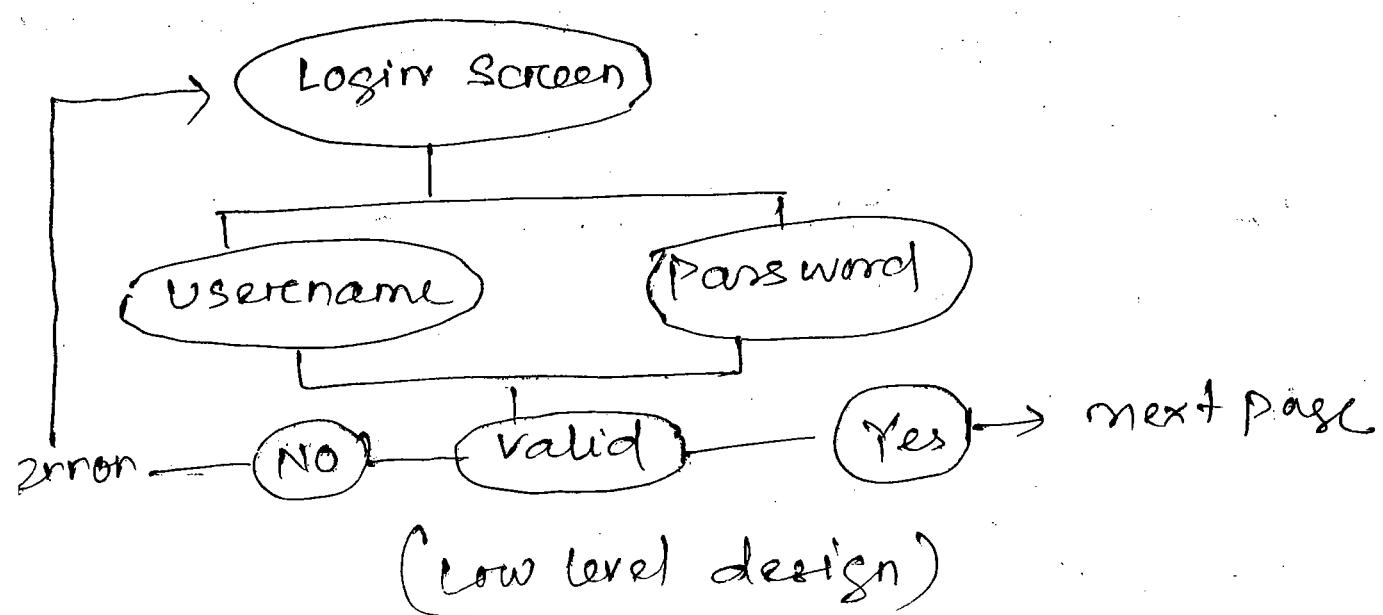
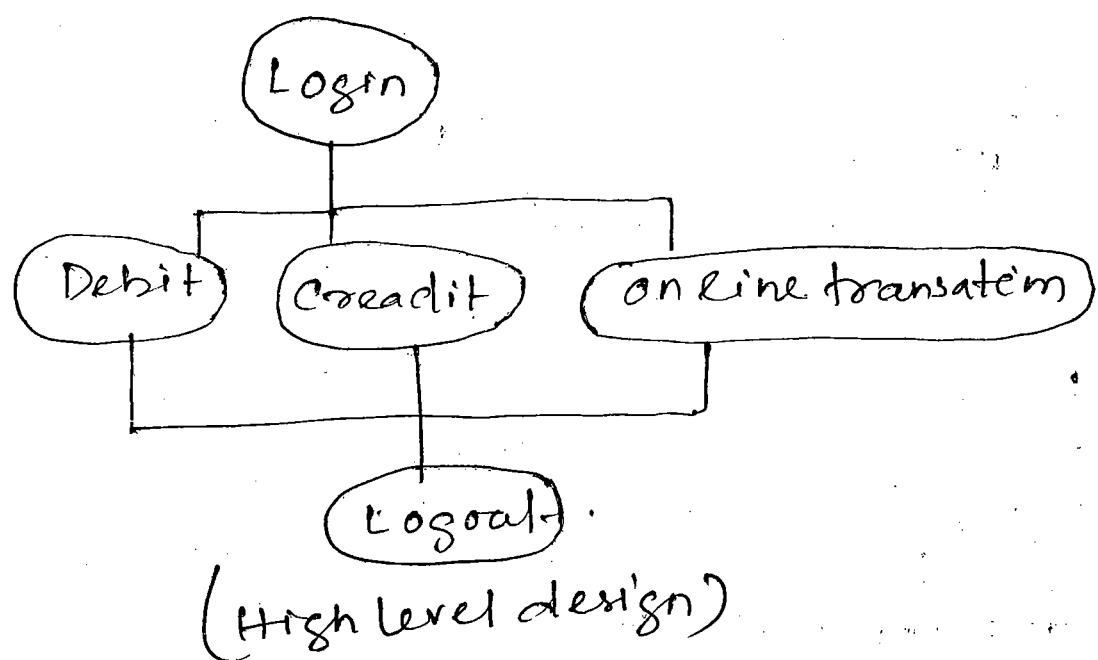
3) Process:

Design Process is always done in the following levels.

(*)

(i) High Level Design:- It is the process in which how many modules a single project can be divided into is decided. High level design is always done by chief architect.

(ii) Low Level Design:- It is the process in which how many sub modules / units a single module can be divided into is decided. Low level design is done by the technical lead.



(4) Output:-

Having done the design designer prepare the document known as "Technical design document (TDD)".

Content of Technical Design Documents:-

a. High level design information:- this field describes various modules from the point of high level purpose. In other word over view of the modules is describe.

b. Low level design information:- this field describes various technical aspects in terms of the following diagrams.

1. Class diagram:-

This section describes various classes that are used for developing the application.

2. Object diagram:-

This section describes various object that are created based on various classes in creating the application.

3. Activity diagram:-

These diagram describe the activities of various objects that are participated in the implementation.

4. Sequence diagram:- These diagram describe the sequence of activities.

5. Deployment Diagram:- These diagram explain how the installation of the application can be done once the total application is developed.

6. Dataflow diagram:- These diagram describe the flow of data ~~among~~ the modules

c. Frontend design Specification:-

This section describes the GUI development required for the corresponding developments.

d. Backend design Specification:-

This section describes the database design requirements so that the database developer can make user friendly for creating the required database table.

E. Functional Specification:-

Pseudo ~~code~~ code (Algorithm) for the developer to implement/develop actual coding in any ~~language~~ specific programming language.

D] Coding:-

1. Responsibility :- To implement the design for development of code.

2. Role:- Developers / programmers.

3. Process:-

The development team is provided set of guidelines to ensure consistency, understandability and maintainability of the programs. These guideline are known as coding standards.

Ex:-

1. Leave 4 char. left margin.

2. Inserting the comment for each specific block of code.

3. ~~following~~ ^{forming} Coloure coding for different types of variable.

4. Declaration Convintion for classes, function, Sub-procedures etc.

4. Out come:-

The outcome of codings is known as "Source code documents(CSD)".

~~SCD~~

NOTE:- The ~~SCD~~ prepared in the coding Phase will not be send to the testing Phase as such but it will be converted into different formats as given below.

1. In case window base application:- the source code is converted into executable format (.exe file)
2. In case web application:- the SCD is converted into deployable formats and it sent to the testing phase along with URL, (address of the page)

Testing :-

1. Responsibility:- To validate the functionality of an application.

2. Roles:- Test engineer / (TE)

3. Process:-

1. BDD / SRS Review:- Test engineer study these document to understand the functionality of an application.

2. Preparing Reviewing Report (R.R):-

Test engineer while reviewing can list out all the query for which the clarifications

are required, a specific document as known as Review report.

(3). Sending Review Report to BA:-

Review is send to BA for the sake of clarification.

(4). TCD Preparation :- [Test case document(TCD)]

Test engineer prepare TCD after having functional knowledge of an application.

Test case - is an angle/case/perception.

Test case:- It is defined as an angle/case/perception with which a specific functionality can be tested to ensure its functional perfection by making it defect free.

(5). TCD- Execution:-

Once the application is developed and released the test engineer can execute the TCD to carry out testing on it.

(6) Identification & Documentation of Defects:-

While testing tester will identify and list them out in a separate document known as "Defect profile Document (DPD)".

(7) Bug Reporting & Test reporting :-

Bug Reporting:- It is the process in which the DPD is sent to development team for the sake of Rectification of defect.

Test reporting:- It is the process in which TRD (Test Report Document) is send to the high level management to let them know the status of testing, stability of functionality & the productivity of testers.

Outcome:- tested ok element

Interview Question:-

N1/ RR/TCD are prepared before testing
∴ DPD is prepared while testing.
∴ TRD is prepared after testing.

N2/ outcome of testing is defect free refine product.

N3/ The following activities are parallel task of the tester while development goes on,

1) To Review the BDD

2) To prepare the RR. (Review Report)

3) To communicate with BA for the

4) To prepare TCD.

Saturday 14th February and Sunday 15th February
are holidays.

16th February 2015

F/Delivery & Maintains:-

Delivery

(1) Responsibility:-

Once the product is developed,
thoroughly tested it will be sent to the customer.
- mere environment for business usage.

(2) Role:- Project Manager (PM) / SQA - Software
Quality Manager and Deployment / Delivery
Manager (DM).

(3) Process:-

(i) In case Local customer the DM along
with the team can visit the clients ^{place} along
with the team set up the environment,
deploy the application and make it
available to the client/ customer so that he
can start using it.

(ii) In case Remote customer the DM will
have an access to the customer business server
set up the environment, deploys the applica
- tion and make it available to the client

- (iii) Some time client may be pre occupied and once the deployment to be done later on. In such cases the required software dump can be kept in the business server along with the guidelines for the deployment known as "deployment document".
- (iv) Once the total application/sol^{on site} is developed an ~~outside~~ team is selected with some of developer and tester that can be sent to client place for the required services at the time of deployment.

Note:- Once the application is deployed the system is tested against the following check list. Lendre the process known as "Green box testing".

- (a) To check if the installation is successful.
- (b) To check if the total application is executable.
- (c) To check if there is a compatibility with respect to input device (Keyboard / mouse / joystick etc)
- (d) To check if there is a compatibility w.r.t output devices (Monitor / scanner /

- (e) To check for the compatibility with respect to secondary device (USB cable, External hard disk, etc)
- (f) To check for the compatibility with various platform / operating system
- (g) To check if the system is compatible with the required software used along with the application.

Once the ~~above~~ green box testing is done the onsite team will conduct training for the end-usages, to make them friendly to the application.

Outcome:-

Documents that are associated with delivery phase:-

1. Certification Document (CD) :- Indicates that the functionality is perfect.
2. User Manual :- (UM) the guide line for using the product.
3. Deployment document :- (DD) Guide line for installation of product.
4. Software Delivery Note (SDN) :- To specify minor issue and to give some useful information.

Maintenance:-

Responsibilities: — To enhance the product or to rectifying the defects in product.

Role: — Change control Board (CCB), (PM+SQM+BA)
Developers and Testers

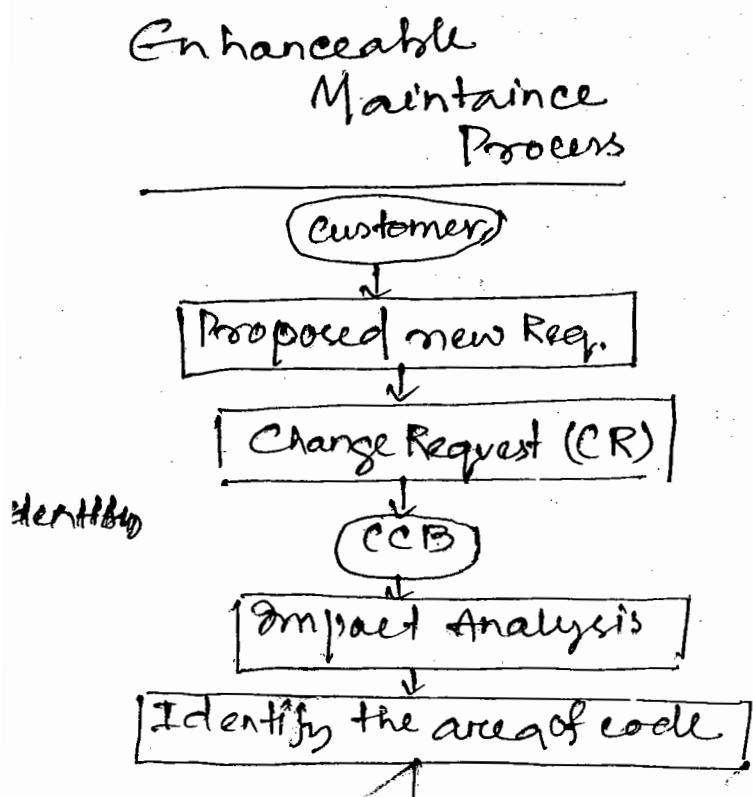
Process :-

Maintenance process is associate with the following two types of implementation.

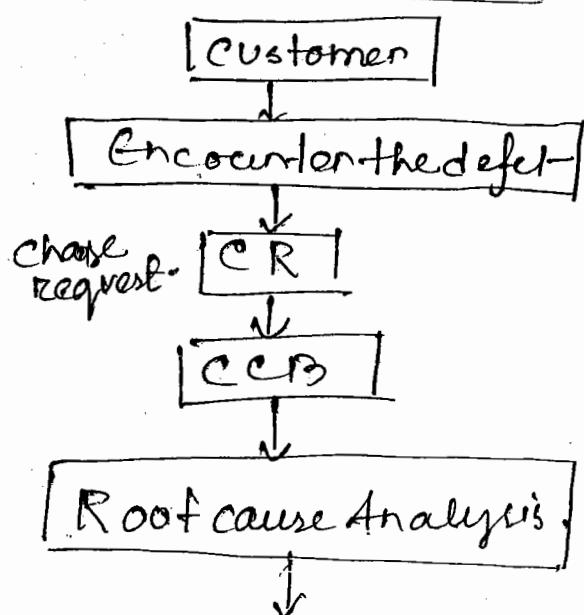
(i) Enhanceable Maintenance:- (To implement new requirement / features)

(ii) Corrective Maintenance:- (To rectifying the defects found in the product used by customers)

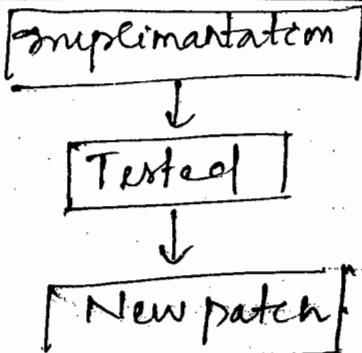
Enhanceable Maintenance Process



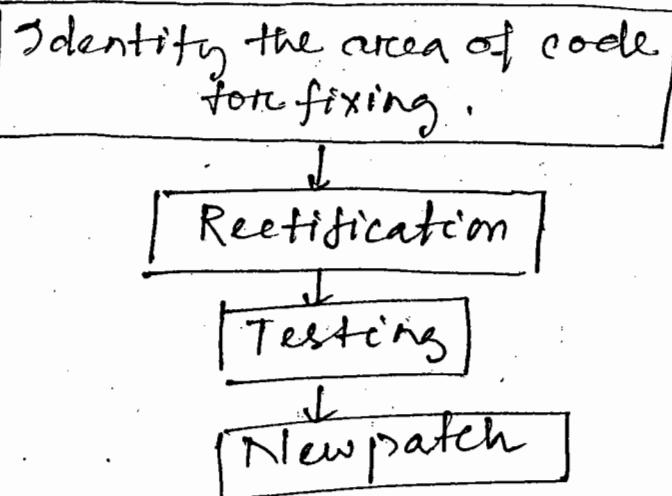
Corrective Maintenance Process



Enhanceable-Maintain Process



Corrective Maintenance



Out come:-

Depends on the enhanceable or corrective maintenance the outcome can be enhanced product or Refined product.

Kinds of Testing:-

Depends on what to be tested and where exactly testing is conducted the following kind of testing having classified

1. conventional testing. (C.T)

2. Unconventional testing (U.C.T)

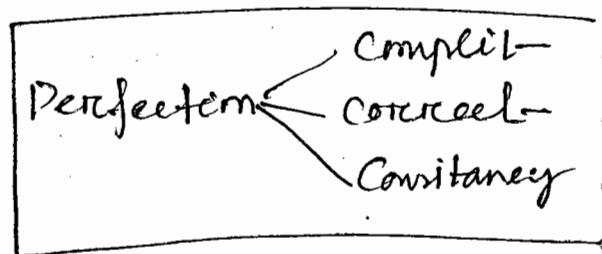
3. Process testing (P.T)

1. Conventional testing:-

Its a kind of testing in which the product is tested after coding to check its functional perfection.

(a) Conventional testing always depends after coding.

- (b) conventional testing focusing on functionality of an application.
- (c) conventional testing is done by test engineers.
- (d) conventional testing is always ^{also} referred as "validation testing".



Unconventional Testing (UVT)

It is another kind of testing in which, conformance present in the document is tested to ensure information perfection (completeness / correctness / consistency of information).

- (a) UVT happens from the beginning of the SDLC till the end and in all the phases.
- (b) UVT focuses on information perfection.
- (c) UVT is always done by Quality Control engineer (QC engineer).
- (d) UVT testing is also referred as "Verification testing".

3. Process Testing (PT) :-

It's another kind of testing in which the roles and responsibility are tested to ensure that the task are done as per the guideline/ process.

a. Process testing happens from the begining of HLDL till the end and in all the phases.

b. Process testing focuses on the roles and responsibility to ensure the completeness of task as per guid line.

c. It is done by Quality Assurance Engineer (QA Engineer).

d. Process testing is also referred as "Verification testing".

(Q.A/Q.C/Test Engineer)

Q.A Activities:-

1. QA Belongs to process oriented activity rather than product oriented activity,

2. QA Belongs to Preventive action rather than the corrective action.

3. QA always focus on Roles and Responsibilities and to ensure perfect accomplishment of the task as per the guidelines.

4. QA Activity is always associated with audit rather than inspection.

QC Activity:-

1. QC belongs to product oriented activity.
2. QC belongs to corrective activity.
3. QC always focus on identifying defects on the documents not on products. Segregate/ Separate them and correct the information to ensure information perfection.
4. QC activity includes both inspection & audits

Tester Activity:-

1. Tester Activity belongs to product oriented activity.
2. Tester belongs to corrective activity.
3. Tester focus on functional perfection by making the product defect free.
4. Testers are nothing to do with audits or inspection.

Interview Question:-

Q1. Differentiate among QA & QC & Testing?

Q2. Does testing belong to QA or QC.

Ans:- Science the testing as well as QC are product oriented activities testing belongs to

QC:: Science the testing ~~is~~ is nothing to do with process it does not belong to QA.

Method of Testing:-

Application:- Application is a combination of function factor & structural (program) factor depends on what factor of an application he tested and by whom it is tested there are basically two methods of testing.

- (fundamental)
(testing) 1. Black Box Testing (BBT)
2. White Box Testing (WBT)

1. Black Box Testing:- (BBT)

It is a method of testing in which application is tested without having internal structural knowledge.

(Design
(or)
Programming
Environment (or)
Knowledge.)

Black Box Testing focuses on functional factors of an application and it always done by test engineers.

2. White Box Testing (WBT):—

It is another method of testing in which one can perform testing on application (program of an application). Having internal Structural knowledge (Specially programming knowledge).

White Box Testing focusing on structural part of application and it always done by developer.

Type	Focus	Role	ISK - statin.
1. BBT	function	TE	X
2. WBT	Structural	Dev.	✓
3. GBT	function	TE	✓

3. GRAY BOX TESTING (GBT) :-

It is another derived method of testing in which functionality of an appl' is tested ^{thoroughly} having internal structural knowledge.

4. STATIC Testing : —

It is another method of testing in which an application is tested without being executed. Ex:- GUI testing (look & feel)
Document testing.

Testers as well as QC will execute this testing.

17.02.2015

Technique of Methods:-

1. Black Box Testing Technique:-

Test engineers while doing BBT they will always follow the following Technique.

A. ECP - (Equivalents Class Partitioning)

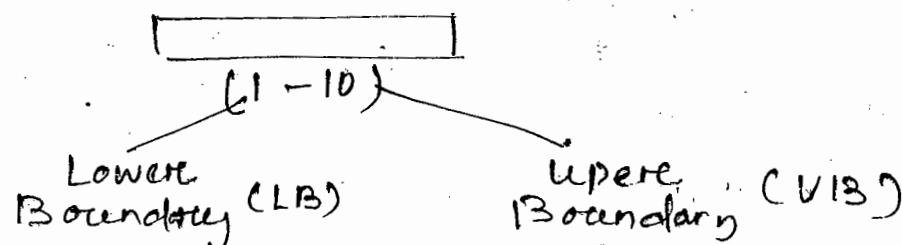
It is the technique used for dividing the test data into two equal parts - valid & invalid based on the business rule.

Exclusivity Emphasized Condition :-

For example the text box must accept alfa numeric / lower case / and 3 to 5 char. long String.

B. Boundary value analysis :- (BVA)

This technique is used for optimizing the number of test data element whenever there is a range kind of output.



$(LB-1, LB, LB+1)$ Min. $(UB-1, UB, UB+1)$

Ans - 0, 1, 2 5 9, 10 (11
invalid valid invalid inputs

C. Encore Gaining (EG) :-

This technique is used for testing an application not with the functional knowledge But with the past experience and domain expertise to find out the blind spot.

D. Decision Table (DT) :-

This technique is used for tabulating various condition for the sake of testing as specific functionality. The Decision Table is as given below.

	C ₁	C ₂	C ₃	C ₄	C ₅	C ₆
User Name	✓	✓	✗	✗		✓
Password	✓	✗	✓	✗	✓	
Result	H.P	E	E	E	E	E

↗ Home page

E. State Transition (ST) :-

This ^{technique} provided the systematic sequential test flow & so that testing can be covered properly.

F.) Use Case Testing Technique:- (UC)

This technique is used for creating the test cases based on use case.

Use case:- Use case is the description of the functionality in terms of diagrams that describes actors, actions and responses.

2. White Box Testing Technique :- (WBT)

Developers while doing white Box testing can complement the following technique to cover various factors of the program.

- A. Statement Coverage:- To check if each statement is executed at least once.
- B. Condition Coverage:- To check the implemented condition, if it is working fine.
- C. Path coverage:- To check the execution flow of the program.
- D. Branch coverage:- To check if the logic is branching out as per the design specification.
- E. Function coverage:- To check if the function are properly called from within the calling area.
- F. Dataflow coverage:- To check the data is flowing within the program as per Data flow diagram (DFD).

g. Loop Coverage :- To check the number of iterations, Repetition, are happening as per the design.

H. Baosts Path Coverage :-

To check for the cyclomatic complexity (logical complexity) in order to element the complexity of the code.

3/GRAY BOX TESTING TECHNIQUE :- (GBT) :-

(A) The tester engineer while testing the functionality must have internal structural knowledge as describe below.

(a) Testing can be done with design knowledge.

(b) testing can be done by environmental knowledge.

(c) testing can be done with programming knowledge.

(B) When ever the tester identifies the defect when it is ^{being} sent to developer it must be associate with the hint - (the root cause of defect)

4. Static Testing Techniques:-

static testing is associated with following technique:

(A) Review:- It is the process in which either checking or study can be usually conducted on the documents for the sake of refinement or understanding that contain of the document respectively.

(B) Walk Through:-

It is the process in which a more detailed study / checking can be conducted to have overall understanding / ^{Shallow} code testing with respect to document / work done.

(C) Inspections:- It is a process of checking conducted without any prior or estimate.

(D) Audits:- It is a process of checking conducted with the prior or estimation.

There are two type of audits.

a. Internal audits:- conducted within the company by QA/QC. The outcome will be internal audit report that talk about the productivity of employees.

6. External Audit:- These are conducted by the external auditors, the outcome is external audit report that can talk about the maturity of company.

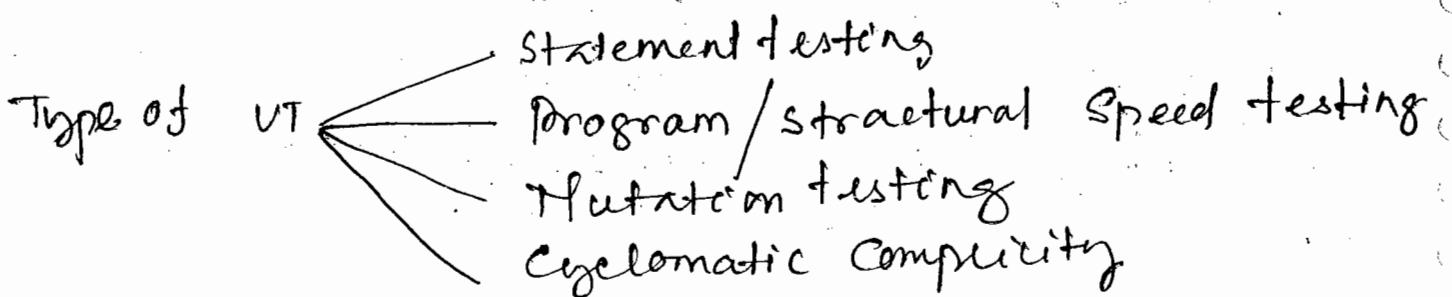
* Levels of testing:-

Testing happens in term of validation (QA/QC) in the pre development phase. Similarly testing happens in the post development phase in term of validation (conventional testing). In the same way there is a need for testing while development phase.

As the development happens in various level testing must also happen in various levels, in terms of level of testing.

(1) Unit testing:- It is the first level of testing in which soon after unit is prepared it will be tested for its desired O/P for a given input.

unit testing belongs to white box testing and it is always done by developers.



2. Module Testing :-

Once the units are developed to form a module, it will be tested for its functional perfection, known as Module testing.

Module testing belongs to Black Box testing and it always done by test engineers.

Type of Unit Testing:-

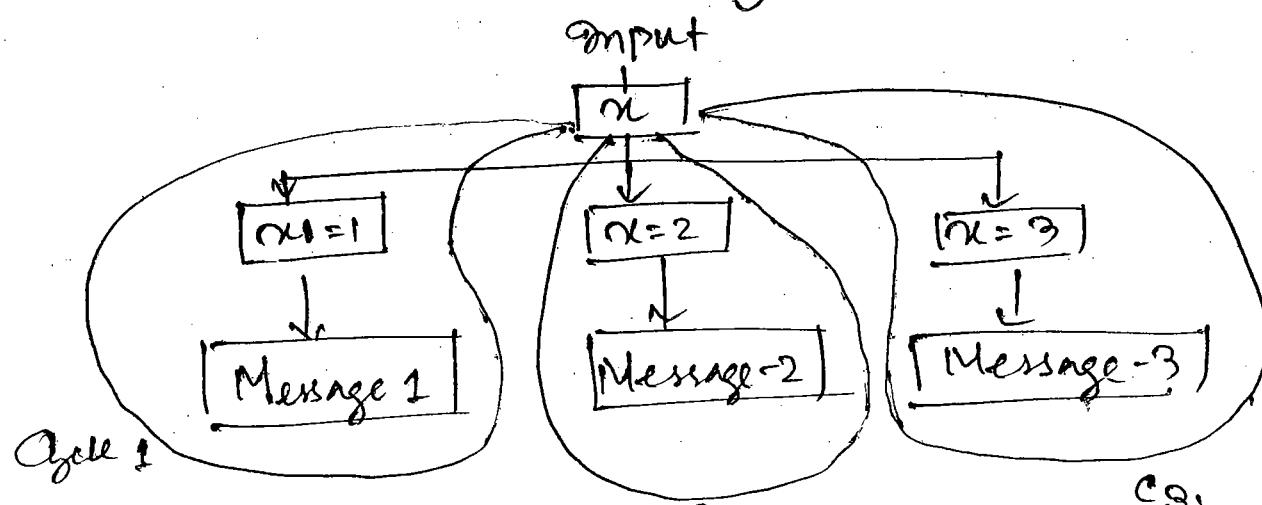
Unit testing can be done by the developers with the following types:

3). Statement Testing / Statement Coverage Testing:-

Developer check each and every statement present in the program if it executed at least one.

4) Cyclomatic Complexity Check:-

Under Cyclomatic complexity checking, developers identify the complexity of logic in term of multiple cycles to eventually reduce the complexity.



(c) Structural / program speed technique:-

Under this testing developers try to analyse the code to decrease the execution time and to increase the performance by doing code optimization.

S1:

a = Human

b = Scientist

c = Artist

a = b

b = c

Point(a) = artist

S2:

a = Human

b = Scientist

c = Artist

a = a+b

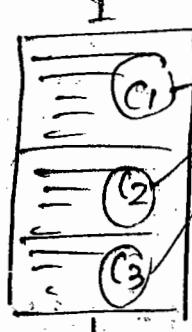
b = c+a

Point(a) → Human
+ artist.

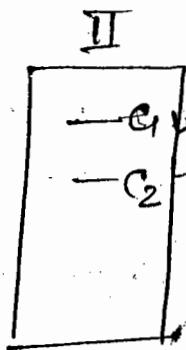
(d) Mutation testing:-

Under this testing the original program is associated with some changes of the logic intentionally to check the program's dynamic enough to identify the changes. The changed version of program is known as Mutants. Some mutants are tested that is called mutation testing.

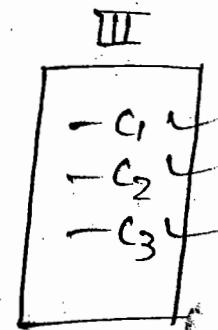
If all the changes are identified then mutation testing is passed. Otherwise it is failed.



Case-I: No changes are identified then mutation testing is failed



2 changes - 2 identified mutation testing is failed.



all changes are identified then mutation testing is passed

Module Testing :-

Test engineer can perform module testing. In terms of following types

- a) GUI Testing:- To check the look and feel of the application (Background colour/checking alignment)
- b) Functional Testing:- To check the functionality of an application. Function testing can be done in terms of tree testing, ~~and~~ to check the tree flow of an application!! also to conduct retesting to check the tree flow of an application.
- c) Input domain testing:- It is the kind of testing in which the test engineer checks each and every field at the time of entering the inputs to see if the business rule is properly implemented or not. Ex:- To check the user name text box ^{takes} states, only alphanumeric /lower case / 3 to 8 characters (24)

1) Database testing:-

Test engineer under database testing can check the following aspects of database.

1. Database (DB) integrity testing:-

In this type of testing the capacity of database to retain the information as it is, can be tested.

2. Database (DB) o/p operation testing:-

In this testing the capacity of the database in association with the application can display the database contained as it is on the ~~and~~ GUI, can be tested.

3) Integration Testing:-

When ever the modules are integrated soon after they prepared the testing is performed on integrated module known as integration testing.

Scope of integration testing:-

1. Check the individual behaviour of module.
2. Check the integrated behaviour of Module.
3. To check the data flow amount the module as per DFD (Data flow diagram).
4. To check if the user is able to navigate among all the screen that off various modules.

Types of Integration testing:-

Depends on at what level the integration testing depends there are 2 types of integration.

① Low level integration testing (LLIT):-

It is a type of integration testing in which when two programs are integrated it is tested for desired op. Hence it belongs to white box testing (WBT). If it always done by developer.

② High level Integration testing (HLIT):-

It is another type of integration testing for which the navigation among various screens that belongs to various module at high level is tested. This testing is belongs to Black Box Testing (BBT). And it always done by test engineer.

Note:- By default integration testing is low level integration testing (white Box Testing WBT).

Approaches of Integration testing:-

Dt. 18.02.2015

Depends on the nature of customers and how the modules are integrated there are basically 4 approaches of integration.

1. TOP TO BOTTOM APPROACHES:-

a. When the customer does not interfere either in the development process or in the integration process. then this approach is proposed.

b. As in when the child module are created they are integrated with the parent. Since the direction of integration is from top to bottom. It is known as top to bottom approaches.

c. The integration between the two modules may not happened due to the missing of the other module.

To resolved this a temporary dummy program is used in the place of missing child module known as "STUB".

2. Bottom Up Approaches:-

a. when ever the customer interferes the development process as well as the integration process from the begin till the end. this approach is proposed.

(b) As and when the parent modules are developed they are integrated with the child modules. Hence the direction of integration is from bottom to Up/Top. This is known as bottom up approach.

(c) When core integration does not happen between the two modules due to the missing of the other parent module to resolve this a temporary dummy program is used in the place of missing parent module "DRIVER".

3. Sandwich Approach:— (Hybrid Approach).

a) This approach is followed when core the customer interferes the development process as well as integration occurs in the middle.

b) Hence it is a combination of top to Bottom and Bottom up approach. It is known as Sandwich/Hybrid Approach.

c) In this approach both stubs as well as drivers are used.

4. Big Bang Approach:

a) Customer maturity does not have any effect on this approach.

b) Once the entire module are developed then only integration process will start.

(C) No stubs or drivers are required in this approach.

System Testing:

Once the application software is developed completely, it will be deployed onto customers specified simulated environment to create the complete system.

If the testing is conducted on the entire system it is known as System testing.

System testing include the following type of testing.

(1) GUI Testing.

(2) Function Testing

(3) Database testing.

(4) Non functional testing as given below.

(i) Compatibility testing

(to check the compatibility with various operating system / browsers)

(ii) Security testing

(to rule out unauthenticity, validity and to protect from destructive elements)

(iii) Configuration testing:- (to check the adaptability of the system with various hardware devices.)

(iv) Inter operability testing:- (to check with the association capacity of the system with other application).

(v) Performance testing:- (to check various aspect of application as describe below)

(a) Load testing:- (To determine the load bearing capacity of an application)

(b) Performance testing:- (to check the quickness of the application in response)

(c) Stress testing:- (to check the stability of an application)

(d) Volume testing:- (to check the data handling capacity of an application etc.)

Science various type of testing are covered under system testing. It is consider to be a full fledged testing.

System testing is black box testing and it always done by test engineers.

5) User Acceptance Testing:-

It is a final level of testing for which the application is tested for the presence of customer to ensure if the user acceptance criteria (Derived from implicit requirements) is justified in the application that can produce customer delight.

User Acceptance testing belongs to black box testing & is always done by basically customer or the test engineer if not the customer.

User Acceptance testing .

Types of testing:-

User Acceptance testing can basically done in terms of following types.

(1) α testing (Alpha testing):-

It is a type of user acceptance testing conducted on an application is final testing just before it is delivered to the customer.

α testing is always done by the customer / test engineer .

(ii) Beta testing:-

Once the product is delivered to the customer.

It is tested for the first time in customer environment.
Known as Beta (B) testing.

B testing is done by customers/end-users

and B tester. (Third party testers)

It is critical testing as compared to α testing.

(iii) Port testing:-

It's a type of testing in which the application is tested by the actual team but in the customer place.

Difference betⁿ α testing & B testing.

α testing.	B testing.
(i) α testing is done as a final testing.	(i) B testing is done as critical testing.
(ii) α testing is done in the development company.	(ii) B testing is done in the customer environment.
(iii) α testing is done by customer/tester	(iii) B testing is done by customer/enduser/B tester.
(iv) α testing can be done only on projects.	(iv) B testing can be done both on projects & products.

Type of Testing:-

1. Smoke Testing.
2. Sanity Testing.
3. Regression testing.
4. Re-Testing.
5. Static testing.
6. Dynamic testing.
7. Alpha (α) Testing.
8. Beta (β) Testing.
9. Pilot testing.
10. Installation testing.
11. Compatibility testing.
12. Configuration testing.
13. Usability testing.
14. Accessibility testing.
15. Security testing.
16. Scalability testing.
17. Monkey testing.
18. Exploratory testing.
19. Adhoc testing.
20. Forced error testing.
21. End to end testing.
22. Heuristic testing.
23. Parallel testing.
24. Buddy testing.
25. Pair testing.
26. Load testing.
27. Performance testing.
28. Stress testing.
29. Reliability testing.
30. Volume testing.
31. Localization testing.
32. Internationalization testing.
33. Multi-language testing.
34. Concurrency testing.
35. Endurance testing.

1. Smoke Testing :-

It's a type of critical / short span / & Non-detailed testing conducted on an application to check if all the windows / objects / feature / Basically available to conduct detail testing later on.

2. Sanity Testing : -

2nd phase - Smoke testing is called Sanity testing.
It's also critical / short span / Non-detailed testing conducted on an application to check if all the windows / objects / feature / for properly available to carry out detail testing later on. Hence Smoke testing is conceptually same as Sanity testing but differ from each other perceptually. Smoke testing is associated ~~with~~ with -ve perception whereas Sanity testing is with +ve perception.

3. Regression testing :-

~~Regression~~ Regression testing can be done in two ways depends on the objective.

(1) Bug Regression testing :- It is a type of regression testing in which an already tested functional is once again tested in order to check if the old bugs are really rectified and also to check if there are any new bugs due to the rectification of old bugs.

(ii) Functional Regression testing:-

It is another type of regression testing in which an already tested functionality is once again tested to check if the existing right functionality got affected due to the addition of new functionality.

$$\boxed{\text{complit testing } CT(A) = \text{Regression testing } RT(\text{old}) + \text{First time FT(NEW) testing}}$$
$$\boxed{CT(A) = RT(\text{old}) + FT(\text{new})}$$

4. Re-Testing:-

It is a type of testing in which an already tested functionality is once again tested to ensure if the defect is reproducible if at all any to rule out the environmental issue and to ensure functional robustness ^{strong}.

Notes:-

Retesting can be done in the first ~~first~~ and ^{build} built whereas as regression testing can be done from the second build only. Retesting can also be done under regression testing.

5. Static testing:- (Already covered under Method of testing)

6. Dynamic testing:-

If an application is tested when it's executed

It is known as dynamic testing.

Example - functional testing.

Q) function testing belongs to

- (a) BBT (Black Box testing)
- (b) WBT (White Box testing)
- (c) GBT (Grey Box testing)
- (d) ST (Static testing)
- (e) None.

Date: 19.02.2015

10) Installation testing:-

Its a type of testing in which installation of the application is tested if its successfully done as per the guid line (GL) provided in deployment document.

11) Compatibility testing:-

Its a type of testing in which mostly the products are tested on various environments that are simulated in order to check if the product is compatible with all these environments.

Compatibility can be tested with respect to operating system, Processor, Application Server, Database Server, etc.

3) Configuration Testing:-

Its a type of testing in which the application is tested with various configuration (capacities) of the hardware components like Client-machine, Server machine, other external devices like Printer, Scanner etc. to ensure the adaptability of application.

3) Usability Testing:-

Its a type of testing in which the application is tested for user friendlyness. apart from functional perfection.

1) Accessibility Testing:-

Its a type of testing in which application is tested against accessibility criteria if it's implemented in the application to make it user friendly to the special user (abnormal / Handicapped / mentally retarded user).

Note:- accessibility criteria is the check list define by "us federal regulation act" at 508 in association with w3c (world wide web consortium)

15) Security Testing:-

It's a type of testing in which the application is tested if it is secured from the point of authentication, access control, encryption/decryption and protection.

1. Authenticity check:-

The security system is tested in such a way that the valid users must be allowed and the invalid users should not be allowed. In other words one has to check if the application is password protected or not.

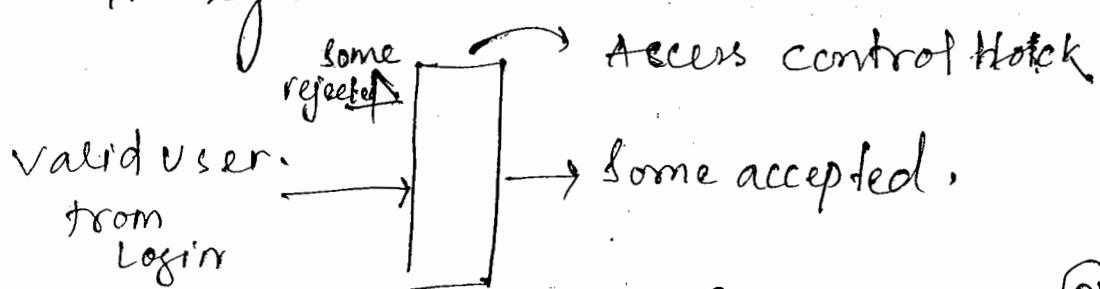
Example- Login Screen



2. Access Control check:-

The security of the system can be tested in such a way that the desirable request must be allowed and the undesirable request must not be allowed.

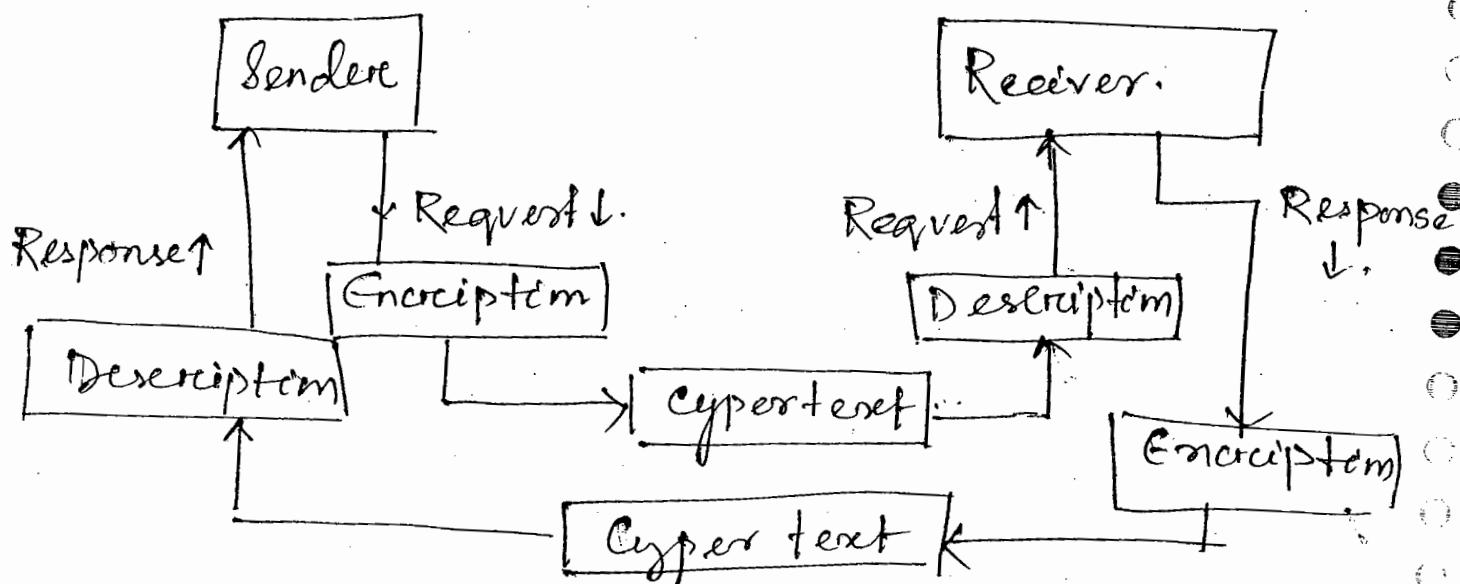
Example- Proxy Server.



3. Encryption & Decryption check:-

The security testing can be done in such a way that the information is travelling between the two nodes in terms of hypertexts, with the help of encryption / decryption, at the required levels.

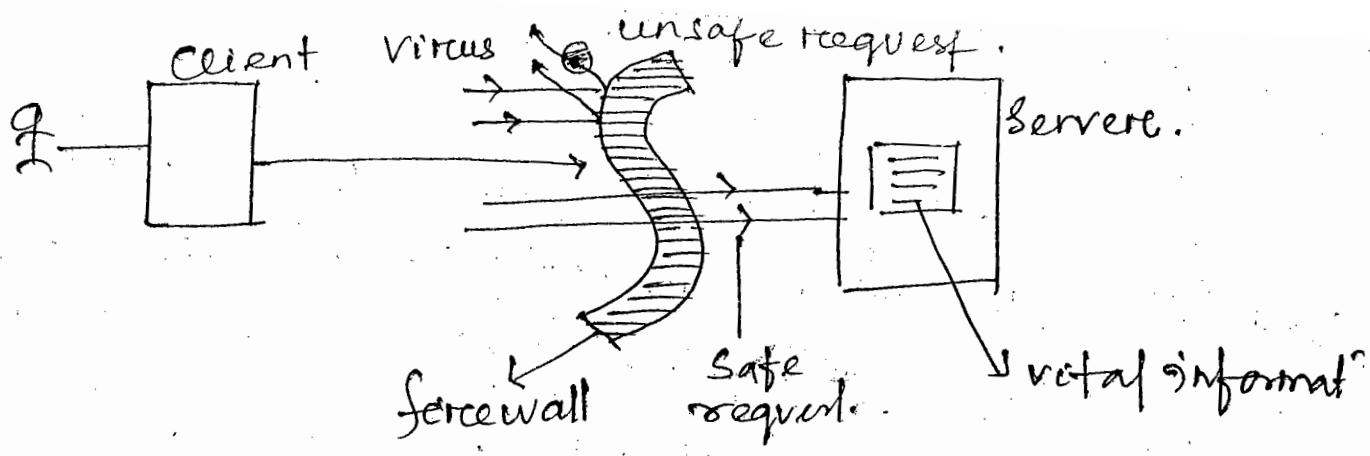
Ex:- Digital signature / Fingerprint



4. Protection check:-

The security testing can be conducted on backend of server in such a way that safe requests are allowed. If unsafe requests that are associated with destructive agencies like virus etc. are not allowed to protect vital information present in Server.

Ex:- fire wall.



16) Scalability testing:-

It's a type of testing in which the application is tested for the adaptability of future requirement with out made changes and with minimum customization.

17) Monkey testing :- / Gorilla testing / Simianji testing

It's a type of testing in which abnormal beyond capacity and more volume of data related operation are performed intentionally on the application to check if it's instable.

18) Exploratory Testing :-

It's a type of testing in which the functional knowledge will not be there for the test engineer initially & he will come to know it while exploring the application & perform testing simultaneously.

→ (freestyle) no rule/conformal.

19. Adhoc Testing:-

It's a type of conformal / Random / freestyle testing conducted on an application without using Test case Document (TCD). to cover the uncovered functional area in the TCD.

In other words Adhoc testing ensure complete test coverage if it is done along with the formal testing.

	BDD	TCD	Type of testing
Process oriented Company	✓	✓	Formal testing + Adhoc testing (FT + AT) 90% + 10%
Semi Process company	✓	✗	Adhoc testing (100%)
Process less.	✗	✗	Exploratory (60%)

20. Forced Error testing:- (-ve testing)

Its a type of -ve testing in which the displayed error message is tested for its correctness (it should be apt / precise / clear / understandable)

21) End to end Testing:-

It's a type of testing in which interoperability, (internal environmental components Operational capacity) of the system is tested.

22) Heuristic Testing :- It is a type of testing in which application is tested with past experience and the domain expertise, to find out bug and errors.

23) Parallel Testing:-

It's a type of testing in which usually the products are tested again against the benchmark (well known products) to determine the quality of the product.

24) Buddy testing:-

It is a type of testing in which the product is tested incrementally when the development goes on.

25) Pair testing:-

It is a type of testing in which testers ~~share~~ share the knowledge and check the application.

Performance aspect ^{of an} application:-

Apart from look and feel and functionality performance factor of an application can be tested in terms of following types of testing.

26. (a) Load testing:-

It's a type of testing in which load bearing capacity of an application can be determined.

Load testing can be done in the following 3 ways,

(i) to apply the load with a constant increment.

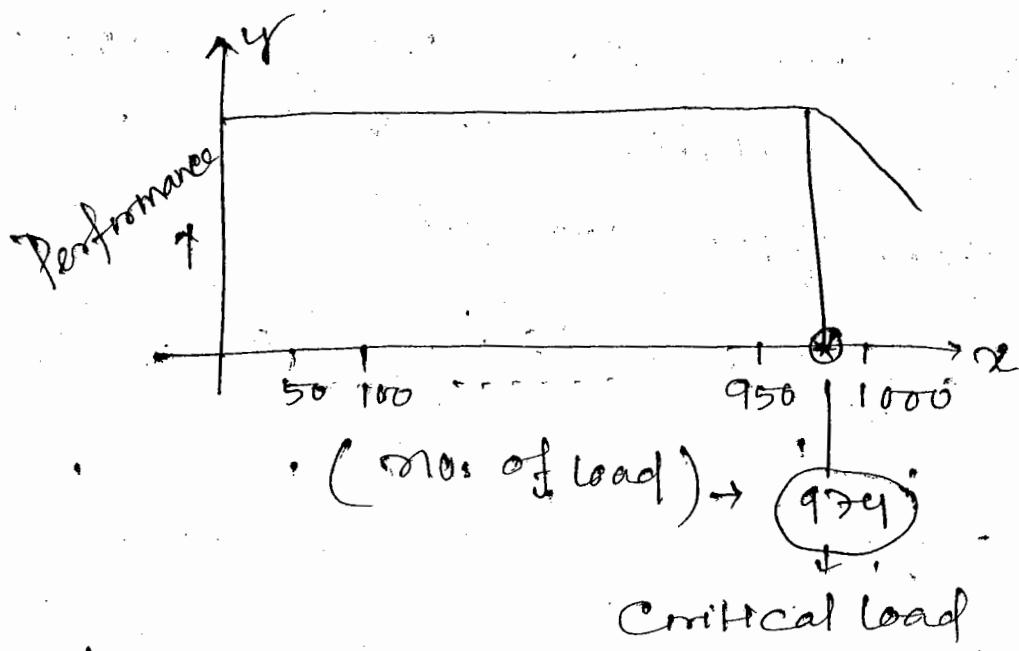
(ii) to apply the required target load on the application.

(iii) to apply huge load of all of sudden at once to check the load bearing capacity.

Sl. No	Load	Result	Comments
1	50	✓	✓
2	100	✓	
3	150	✓	
4	200	✓	
	250	✓	
	500	✓	

Target load:-

550	✓
600	✓
650	✓
700	✓
750	✓
800	✓
850	✓
900	✓
950 →	✓
1000 →	✓



(a) Initial Load :- Load with which the load testing starts.

(b) Incremental Load :- Load that added to previous load constantly to carry out next load testing.

(c) Target Load :- Load that is suppose to be supported by the application.

(d) Critical Load :- The maximum load beyond which the ~~poor~~ performance start ~~degrade~~ing.

27) Performance Testing:-

Its the type of testing in which list of standard task are perform on the application to determine the actual response time which can be compare with expected response time to make conclusion on the performance as per the aspect of the application.

S.No	Tasks	Expected RT	Actual RT	Remarks	Comments
1	T ₁	10 sec.	12 sec.	X	↑ 2 sec
2	T ₂	12 sec.	10 sec.	✓	—
3	T ₃	8 sec.	8 sec.	✓	—
4	T ₄	10 sec.	15 sec.	X	↑ 5 sec.

28) Stress Testing:-

Its a type of testing in which both normal as well as abnormal action are performed on an application to check if it is stable.

29) Volume Testing:-

It is a type of testing in which huge volumes of data related operations are performed on an application to check the handling capacity of an application.

30/ Reliability Testing:- (Soak testing)

It is a type of testing for which both normal as well as abnormal activities are performed by the multiple user on the application for a longer duration (according to industry standards 48hr. to 72 hr.) to check if the application is stable.

20/02/2015

31/ Multilanguage Testing:-

It is a type of testing for which application developed base on unicode (universal code) can be tested with multiple languages input to check if the application is compatible for all the multiple language.

The above testing done ~~can be~~ in two way,

(1) Localization testing.

(2) Globalization / Internationalization testing.

32/ (1) Localization testing:-

It is a type of testing for which test engineer gives the input to the application for multiple language like Spanish / French / English / Sanskrit etc. to check if the application provided proper output.

3) Globalization/Internationalization testing :-

It's a type of testing in which the test engineer can give input to multiple languages which intern can be given to language converter tools to provided the output an english which is global.

4) Concurrency testing :-

It's a type of testing in which the race condition is simulated in order to encountered the deadlock situation.

Race condition :- It is the condition in which multiple task are performed on the application simultaneously to access the common resource.

Deadlock :- It's the state in which the requested resource is freezed (Blocked from usage).

35) Endurance Testing:-

It's a type of testing in which test engineer can perform the specific task repeatedly/ Multiple task sequentially to check if the memory leak is associated with the application.

Memory leak :- It is a state of memory in which there is no space available (due to the previous continuous task) to perform further task.

SDLC MODELS:-

Depends on the nature requirement, nature of project and the style of development there are various models that are followed by the development team as describe below.

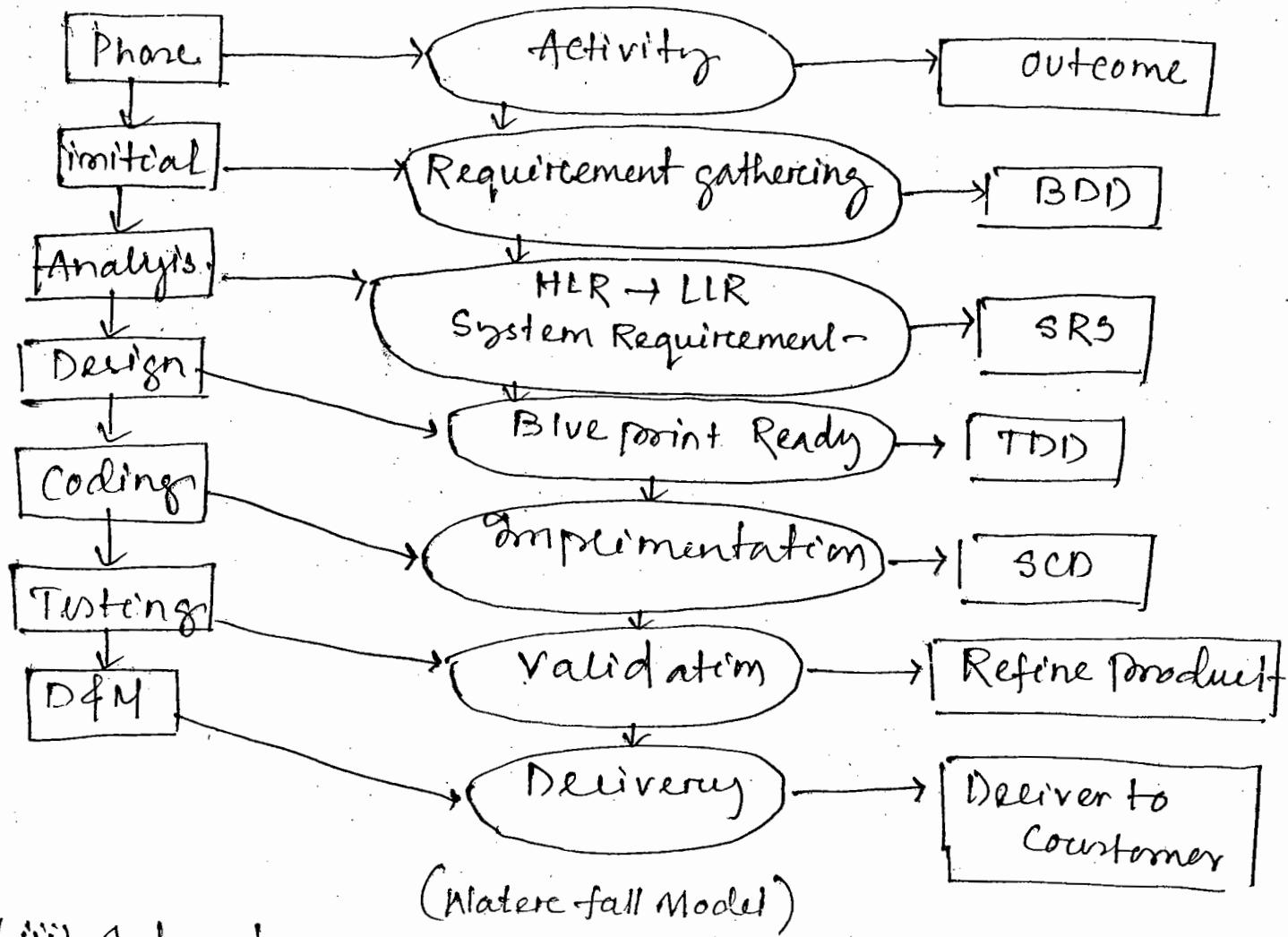
1. Waterfall Model
2. Prototype Model.
3. Evolutionary Model.
4. Spiral Model
5. Fish Model.
6. V model
7. Rad Model / RAD Model.
8. RUP Model
9. Incremental Model.
10. Agile Model.

Waterfall Model :-

(i) This model is proposed whenever the requirements are pretty clear.

(ii) Process :-

From the next diagram SDLC Phases are predefined and linear in which the respective pre-defined activities are done in a sequential manner to produce respective documents as outcomes.



(iii) Advantages:-

- (a) Since it is a simple model all types of company can follow this.
- (b) Monitoring & Management becomes ~~easy~~^{easy} because it is transparent.

(ii) Disadvantages:-

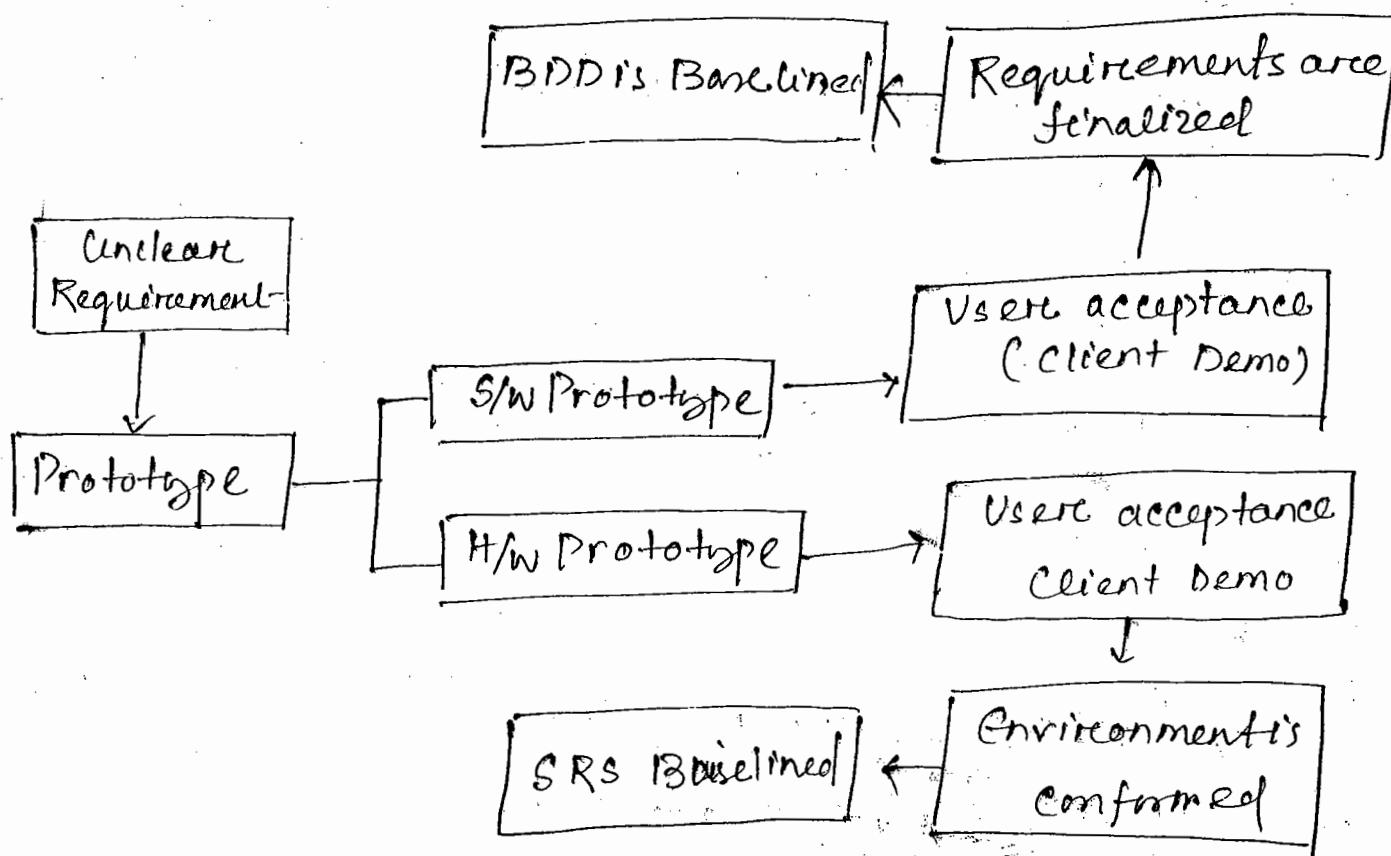
- (a) This mode does not support for the project with frequently change requirement—,
- (b) It does not support long term projects.
- (c) Lot of planning & efforts are required.

PROTOTYPE MODEL:-

It is initially, rapidly created story model for the project.

(i) This model is prioritized when the requirements are not clear.

(ii) Process:- (Prototype model)



From the above diagram a prototype is created in terms of S/W (Software) as well as H/W (hardware) prototype.

Software prototype is subjected to client demo to finalized the requirement and to freez the BDD. Similarly Hardware prototype is subjected to client demo to finalized the environment and to Conformed / freezed the SRS.

(iii) Advantages:-

- (a) Requirement can be finalized.
- (b) Environment can also be forced.
- (c) Customer credibility and satisfaction increased.
- (d) Customer can have future look of the product.

(iv) Disadvantages:-

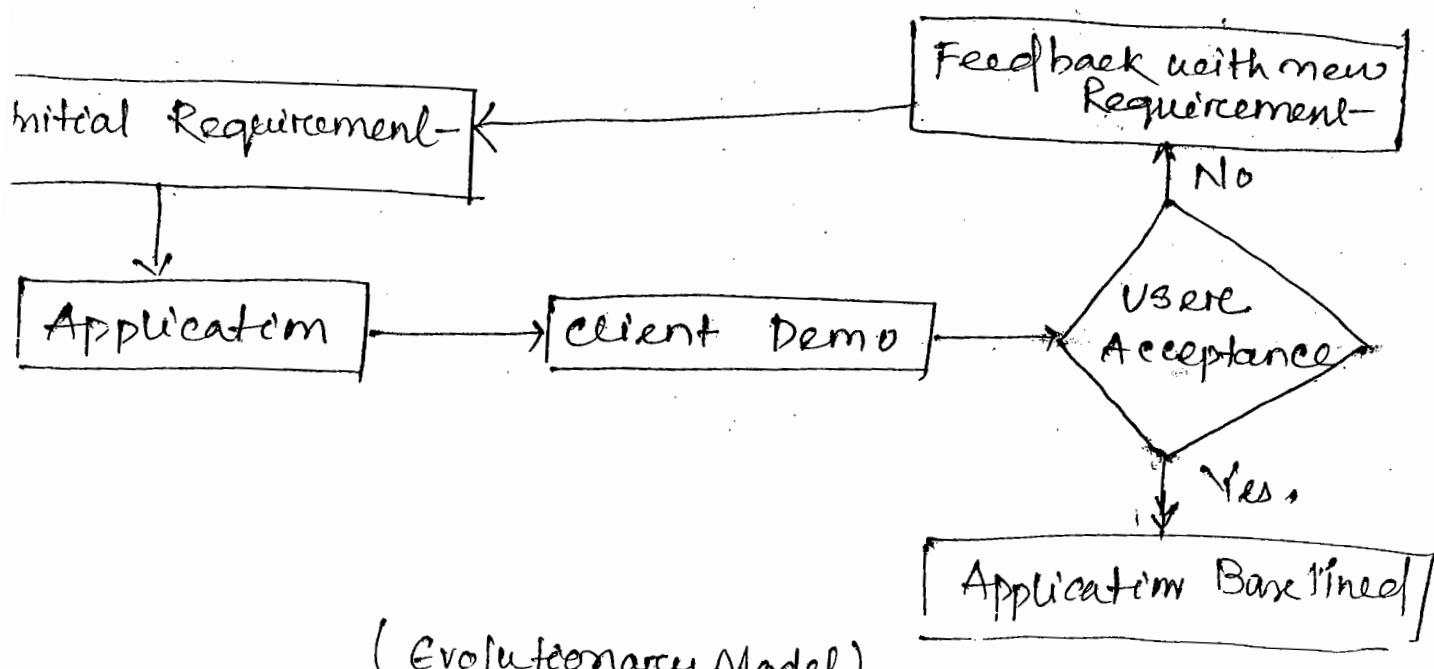
investment —

- (a) The prototype cost has to be incurred by the development company
- (b) In case the prototype is associated with functionality there is every danger that the customer make go for it claiming the actual product.

EVALUATIONARY MODEL:-

(i) Whenever the requirements are frequently changed this model is proposed.

PROCESS:-



→ from the above diagram the solution is prepared base on the initial requirement. That can be sent to the client for the acceptance. In case the customer proposes change in the requirement / new requirement it will be treated as feedback in which lotⁿ is once again prepared. This loop is continued until the final lotⁿ is accepted. Once it is accepted the application is handed.

(iii) Advantage :-

(a) This model is appropriate for the projects that are associated with frequently changed requirements.

(iv) Disadvantage :-

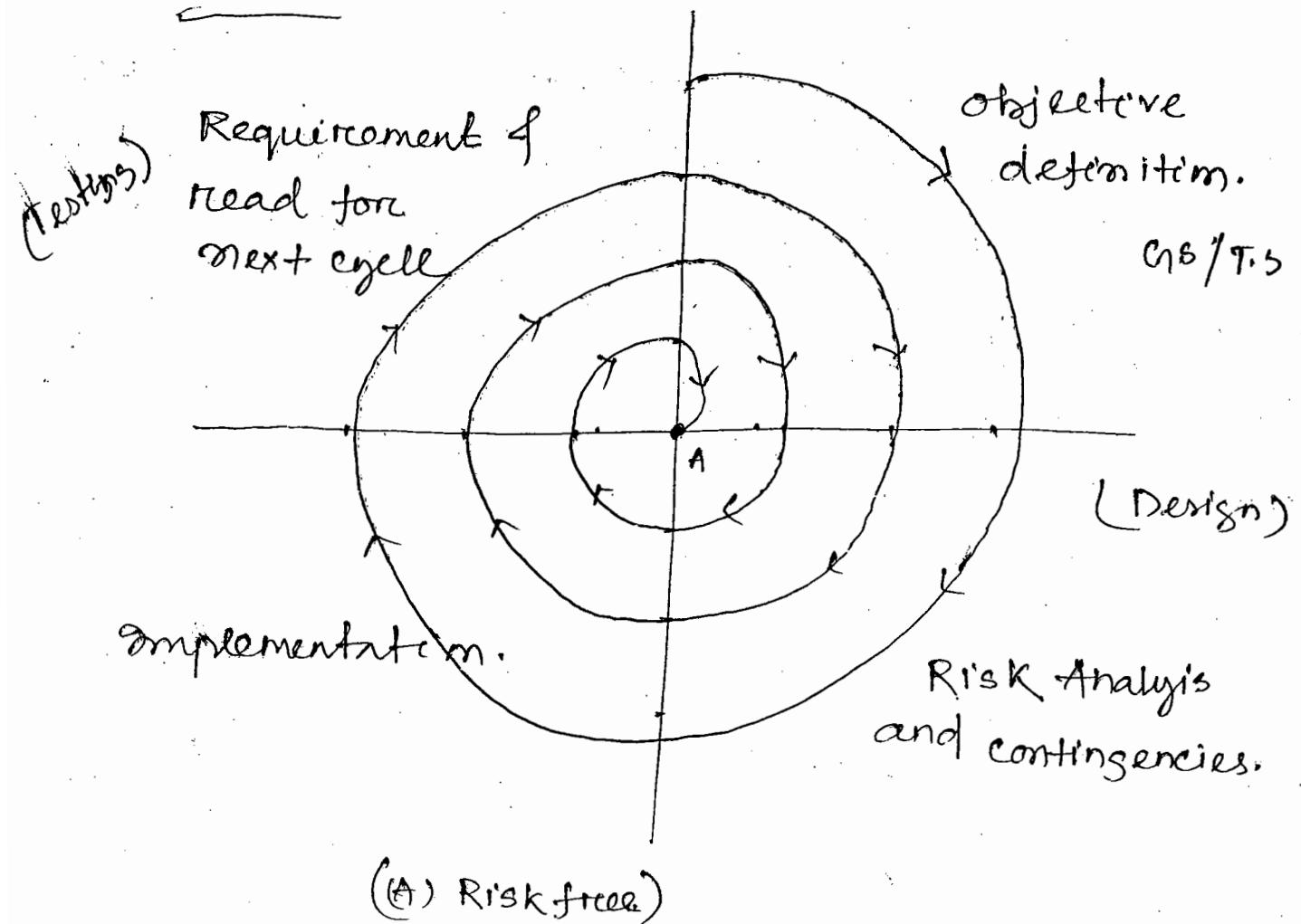
(a) Since it is not transparent, project management / monitoring become stop.

(b) Project deadline can not be determined

SPIRAL MODEL :-

1. Whenever the projects are risk based this module are proposed.

2. Process:-



from the above diagram this model can be implemented in terms of 4 phases.

(i) objective definition:- Goals/Target are defined over in this phase for every cycle.

(ii) Risk Analysis & contingencies:- To identify various risk along with the contingencies.
~~Set that can be modified~~ . In other word appropriate design is created.

~~Set that can be modified~~

(iii) Implementation:- To develop a SOP based on designed.

(iv) Refinement of read for next cycle :-

The SOP developed is tested; refinement is done for the process and ready to go for the next cycle of production.

3. Advantages:-

- (a) It is well suited for Risk based project.
- (b) each phase in this model is flexible & the customizable.

4. Disadvantages:-

- (a) This model is not advisable for using normal and non-risk base project.

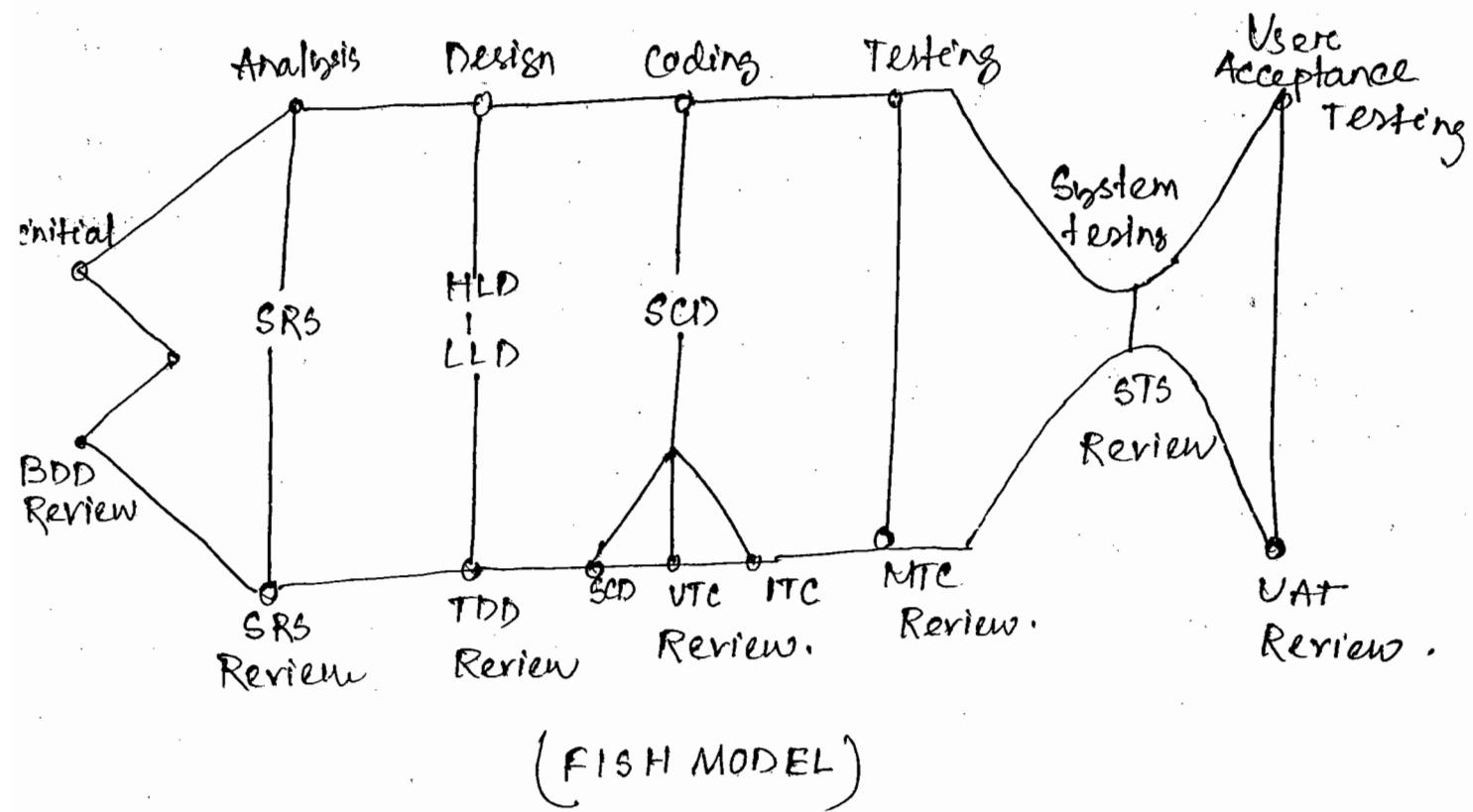


Fish Model:-

1. When ever the review need to be done all the document fish model is proposed.

2. Process:-

Advantage & disadvantages are same as of the model.



From the above diagram each and every Phase can produce the respective documents which will undergo review process to ensure that the information present in the document is complete, correct and consistent.

3. Advantage:-

- (a) To have the better quality of the product.

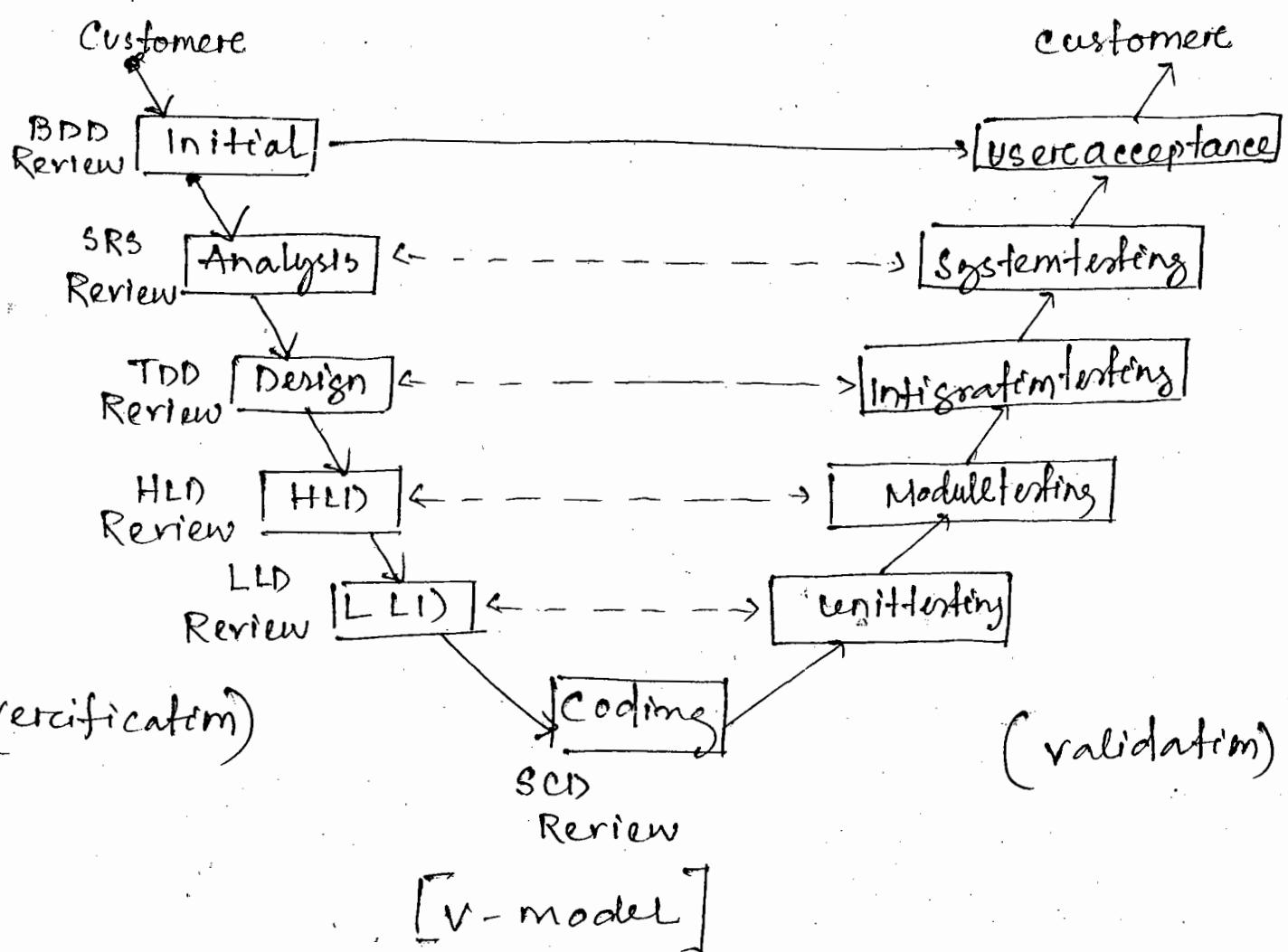
4. Disadvantage:-

- (a) ^{more} Time and cost consumed.

- (b) more planning and more effort is required.

V- Model :-

1. When the project development is to be associated with verification as well as validation this model is proposed.
2. Process :- Advantages & disadvantages are same as fish model.



From the above diagram verification of the document is pick depicted on the left leg of 'V'. whereas as the validation of the application of coding is depicted right leg of 'V'.

on the left side detail of each phase is described whereas as the reality is described on the right side.

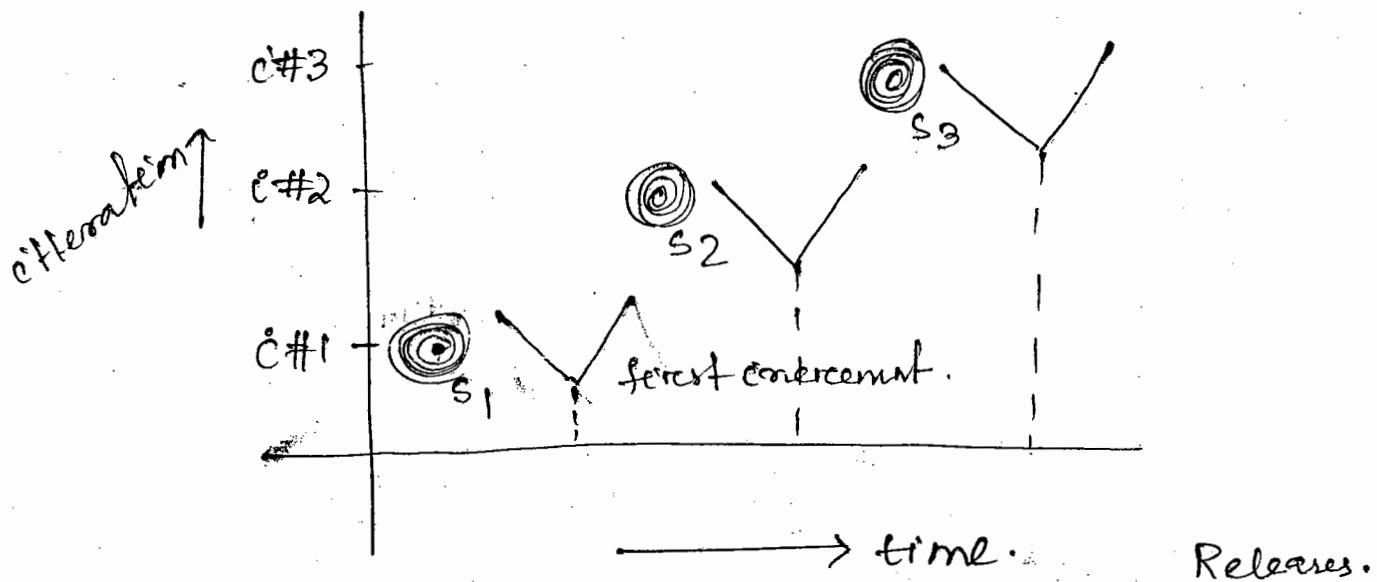
Type of V-model:-

(i) Prototype V model :-

In case the requirements are not cleared and wants to implement verification as well as validation then this model is proposed.

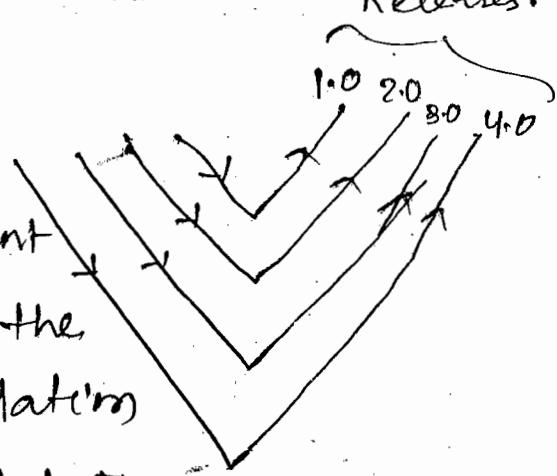
(ii) Incremental V model :-

Incremental V model are used whenever the project are huge and the solⁿ is to be delivered in the form of milestones basis.



(iii) Spiral V model :-

When ever the requirement are frequently change and at the same time verification validation has to be followed this model is proposed.



RAD Model :- (Rapid Application Development Model)

1. When ever the requirements are clear and at the same time solution have to be delivered quickly.
2. To deliver the soln quickly, the development team must be strong.
3. Formal process can be compromised.
4. Direct communication is established between the customer and development team.
5. Co-operation and coordination must be there on either side - customer as well as development team.
6. In case any of the above are not followed the product can't be delivered quickly.

Note:- In RAD Model the development team must be allowed to copy the code if not developed from the already existing resources in order to deliver the product quickly.

want prove otherwise can't believe.

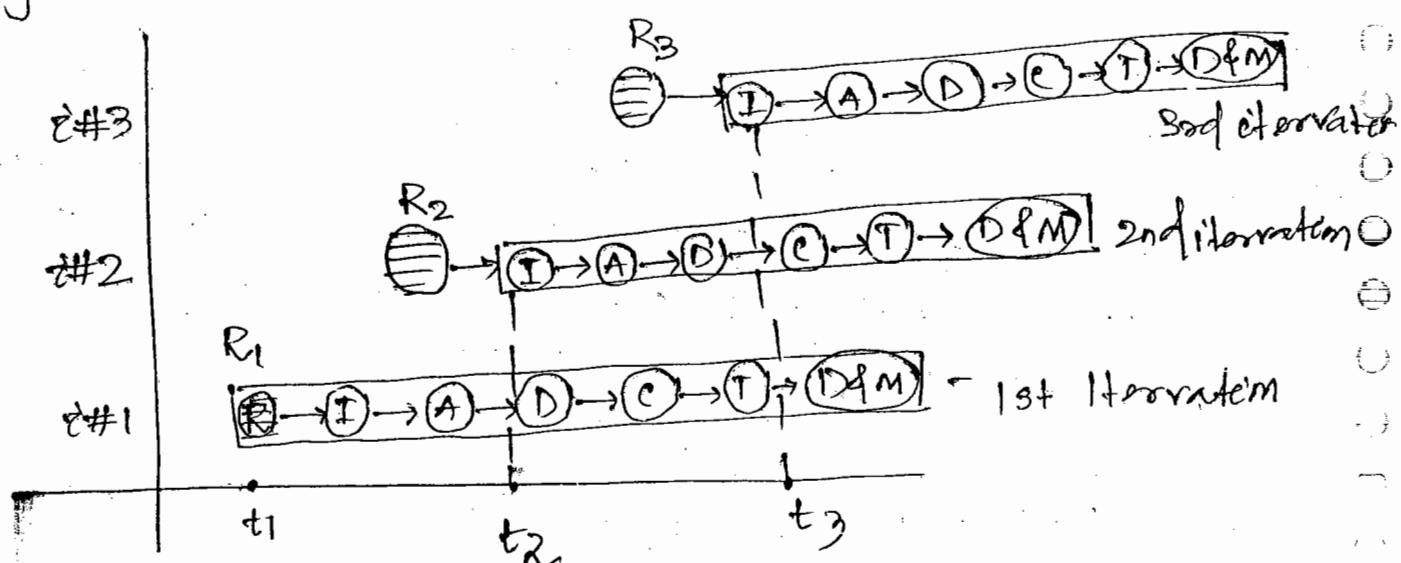
RUP Model :- (Rational unified process Model)

1. In case the development process is to be associated with documentation this module is proposed.
2. This model is complimented in term of following Phases.
(i) Inception (ii) Implementation
(iii) Conceptual (iv) Validation.

- (A) Inception:- To gather the requirements.
- (B) Conception:- To design / create Design.
- (C) Documentation:- To document entire information developed regarding the project.
- (d) Implementation:- To developed the code base on design.
- (e) Validation:- To test the final Solution, to make it defect free & qualitative.

3) Incremental Model:-

This model is proposed whenever the Project is huge in volume.



from the diagram we can understand that the product is delivered to the client in terms of multiple instalments. Hence it is known as incremental model.

Advantage:-

- (a) Though there are intermediate individual software, each software is ensured with utmost quality.
- (b) Project monitoring & management becomes easy.

Disadvantage:-

- (a) The entire software may not be available at once.

21.02.2015 (Saturday) & 22.02.2015 (Sunday) are holidays.

continuing
↑

Date - 23/02/2015



Agile Process:-

1. In case the requirements are changed from time to time, the software is to be delivered quickly and solution is to be delivered in the instalment basis.

Agile Process Model is used.

2. CR proposed by the customers can be accepted by the development team at any stage of the development.

3. Formal process can be compromised and customized.

4. The requirements are basically segregated into various sets of based on the priority known as user stories.

5. For a given set of requirements (User stories) an intermediate output is developed known as Sprint.
6. Sprint is developed with the development-life cycle known as Sprint cycle.
7. Usually the Sprint duration will be 2 to 3 weeks; sometimes 1 month
8. Once the Sprint is developed it can be delivered to the customer, once it is accepted under Sprint delivery.
9. In order to monitor the productivity of the team, quality of the Sprint from time to time meetings are conducted are known as "Sprint meetings"
10. The total project team members must exhibit total commitment towards the entire product development.
11. In Agile process has specific roles such as
 - a) Product owner (to give/gather the req.)
 - b) Scrum master (to see that Sprint development goes smoothly)
 - c) Project Team (Developers/ Testers)
 - d) Business Executives (BEx)

Interview Question:-

Q1> Which model is followed by your company?

or (1) hybrid Model

(2) Agile Model.

Complete.

Chapter - 3

Environment:-

Application is the combination of 3 factors.

Presentation Logic (PL) + Business Logic (BL) + Data Base (DB)

$$A = PL + BL + DB.$$

Type of Environment:-

Depends on how the three components of the apps are deployed into the environment the following type of environment can be/have been classified.

(i) Standalone environment:- (standalone env.)

This environment belongs to 1 tier architecture. In this environment - PL, BL, DB are deployed in single computer. hence it is known as stand alone environment.

Ex:- PC with Oracle & VB application.

Drawbacks:-

- (a) It is a single user system.
- (b) User are isolated from each other.
- (c) Since no transparency, there is a chance for duplication of data.

(ii) Client Server Environment:-

This environment belongs to two tier architecture, for this environment there the two components are low volume and high volume communication.

PL, BL & DB can be deployed onto this environment—
in the following ways:

1. PL + BL in the client machine and DB is in server machine, this type of client is known as thick client.
2. PL is present in the client machine. BL + DL is present in the server machine. This type of client is known as thin client.

Drawbacks:-

(a) Since BL is present in each and every client machine, maintenance of the BL (Business Logic) or modification of Business Logic (BL) from time to time is rather difficult. Hence the following solution is suggested.

(iii) Web/Non web (or) Internet/Intranet Environment:-

This environment belongs to 3 tier architecture. In this environment there are 3 computers one is client Server, another is application server and other is Data Base Server. In which Presentation Logic (PL), Business Logic (BL) and DataBase (DB) are kept and maintained respectively.

Three tier architecture can be divided into two types as describe below.

1. Web/Internet Environment :- If one web server is present in the App. Server it is known as 3-tier Architecture Web environment / Internet environment

2. Nonweb/Intranet Environment :- If one web server is not present in the App. Server. It is known as 3-tier Architecture non web environment / Intranet environment.

Web server :- It's the server which is present in app. servers and it is capable of processing the web request made by the web users, that are associated with just web content. It is not responsible for processing complex process oriented request.

Difference b/w Web server & App server.

Web Server

1. Web server is a subset of App server.
2. Web server can process web content based request.

App Server

1. App server is super set of web server.
2. App server can process complex process oriented request apart from the content based.

Drawback:-

(a) When 3-tier architecture being used there is every possibility that business logic may be increased from top time to time which may ~~interfere~~ affect the performance of App. server. Address this issue the following environment has been introduced.

④ Distributed Environment:-

This environment belongs to n-tier architecture. Since the business logic is distributed among 'n' number of app. servers this environment is known as distributed environment that of the n-tier architecture.

Types of Architecture:-

Depends on the project and the technology with which the project will developed, the following type of architecture are defined by TM and CA → (Chief Architecture)
(Tech. Manager)

(I) 1 tier Architecture:-

Only one container is design to keep and maintain PL, BL & DB

(II) 2-tier Architecture:-

Two components are defined to hold and maintain PL, BL & DB in the manner described in client server environment.

(III) 3-tier Architecture:-

Three components are defined to store and maintain PL, BL & DB in the following 2 types:

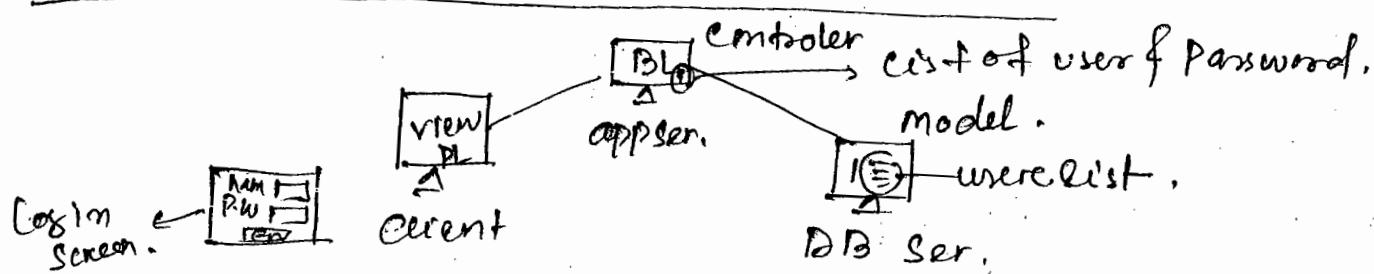
- ① 3-tier Architecture internent environment (web environment)
- ② 3-tier Architecture intranet environment (non-web. environment)

(IV) N-tier Architecture:-

Multiple app server components are defined to keep and maintain BL distributed among them.

MVC Architecture:-

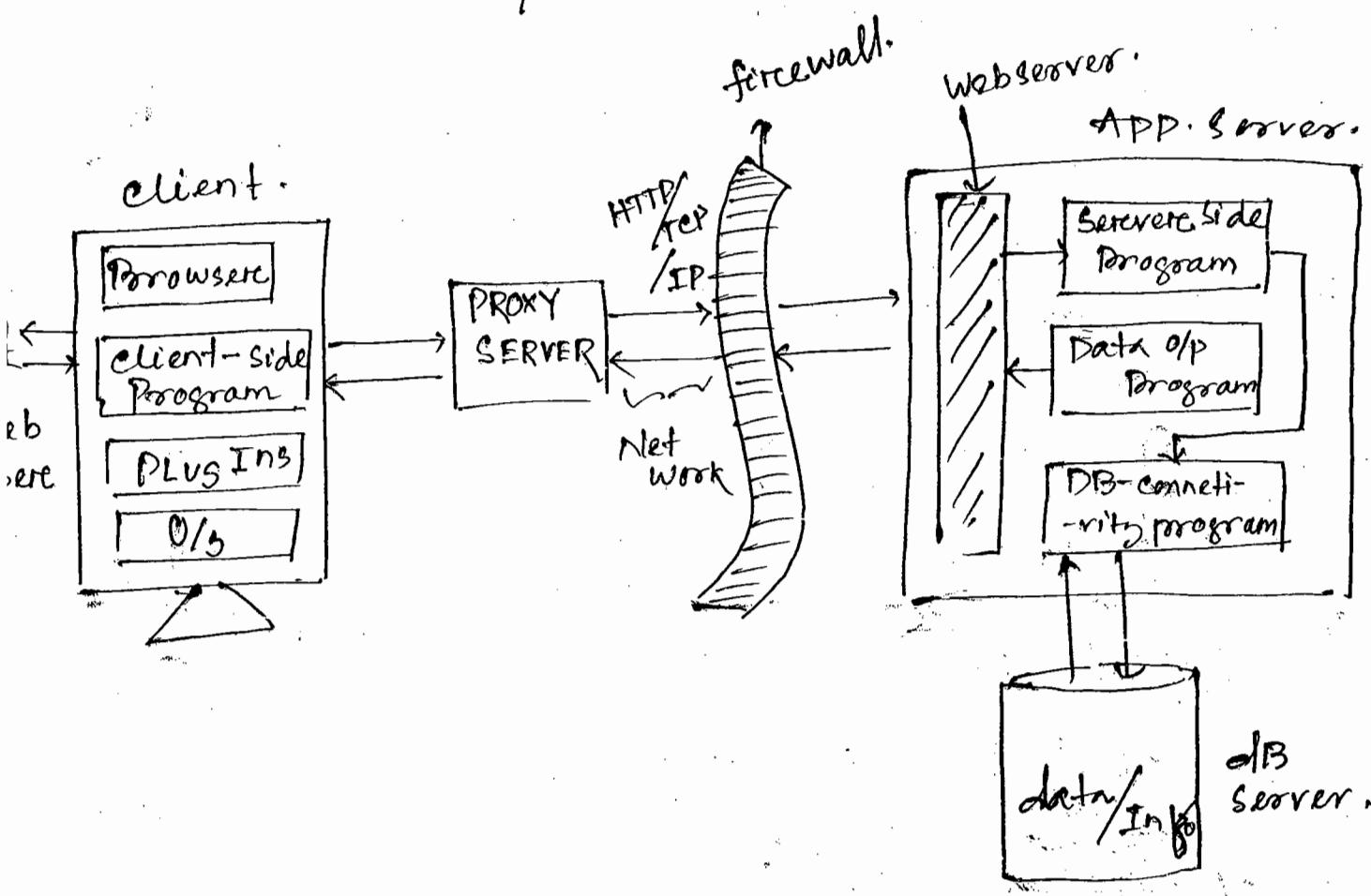
Module view controller Architecture:-



This is the fundamental architecture that belongs to 3 tier Architecture. Model component represents the database that has the database in it.

view component represent the client which can PL unit. Similarly, controller component represent apps server that can contain the business logic from processing the request to create corresponding responses.

Web Architecture / Web Environment :-



Web environment can be basically understood in terms of major components like client, App server and DB server.

- ① Client:- It is a key component which acts like an interface between the web user and the app server.
- It can take the request from the user pass it on to app server and finally the response

of the app server and send it back to the user.
Client has the following sub content.

① Browser:- It acts like an interface between the ~~web~~ web user and client. It is a software that is capable of executing html code and to display the web content to the web user in terms of ~~web~~ ^{web} page.

Ex:- IE, Mozilla Firefox, NetScape Navigator, Google Chrome, Opera.

② client side program:-

This program in terms of PL beside in client to carry out the following responsibilities.

(a) To validate the user entries.

(b) Implement special effect to the web page.

Ex:- VB script / Java Script etc.

③ Plug Ins:- It's a small piece of code or patch that can be made available in the client machine to provide compatibility between the technologies of the browser.

Ex:- Flash Plugin.

④ Operating System:- (OS)

It is the interface between the user and the machine and converts the high level language into low level language / machine language / binary code (0,1).

Ex:- Linux / windows / MAC / Ubuntu.

(B) Proxy Server :-

Date - 24.02.2015

It is a secondary component present in the web environment to monitor the user's activities.

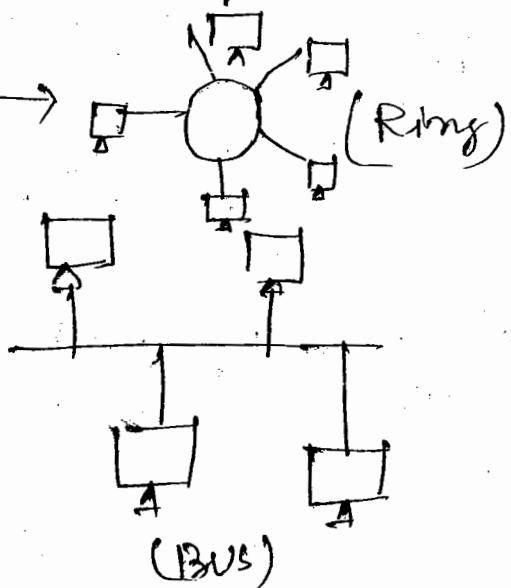
As per the configuration of settings of system admin, the undesirable requests are blocked. Also proxy server creates event log that contains the information like list of sites visited, content downloaded, list of links that are clicked, time spent for each site etc. Hence proxy server can be used in the web environment as a means of security.

Ex:- CCProxy, squid Proxy.

① Network :- Network is the means of communication between two computers so that they can talk to each other. There are various network topology that are followed depends on requirement.

Ex:- Ring topology.

Bus topology



② Protocols:- In the web environment basic protocols that are used are HTTP/TCP/IP.

HTTP is used for proper communication. Whereas TCP/IP will ensure safe reaching of data package to the destiny with acknowledgement.

(C) Firewall:- It is also a means of security that can be installed as a layer before the app-server. So that the request ~~are~~ evaluated to protect Business logic and vital information present in the Application Server.

Firewalls are two types - (i) Software firewall
(ii) Hardware firewall

Ex:- Net gear fire wall.

PF Sences.

Note:- while security testing the testing team must identify fire wall holes (the area that are supposed to be restricted or not restricted) so that the perfect security can be provided to the app server.

(D) APP. Server:-

It is the major component that received the request from the client, process it if required contact to database and sends the response back to the client.

Ex:- Tomcat Apache Server. (Java app.)

BEA Web logic Server.

IBM websphere Server.

The app server has following ~~sub~~^{Contain} components

1. Web Server:-

Web server is the main component present in app server that is responsible for providing the web content to web user.

Web server can serve web page to the user with the following process.

- a. It receives the request from client-
- b. It will evaluate the request- if it contains basic or complex base request.
- c. If it contains base it takes the responsibility to process it.
- d. In this process it contacts DNS - (Domain naming Service server) that is capable of- Converting high level address into low level address. That includes type of protocols, servername / IP address, Port number, Path of the required page.
- e. With the low level address that the required page is picked up and the same is sent back to the client.

(f) Client has the browser to execute the HTML code and display the web content in term of the required web page to the Web user.

Ex:- Internal Information Server (IIS) from Microsoft.

2o Server Side Program:-

This program will decide in the app server in term of BL to process the request and to prepared the corresponding response. In this connection it may contact the database for the required data while processing.

Ex:- (i) JSP - Java Server
(ii) ASP - Active Server
(iii) Servlets.

3o Data output program:-

The main purpose of this program is to keep the response in the specific format like normal text, tables, bar charts, Pie chart and wide varieties of graphical format. The final format given to the web server which in turn can be sent to client.

4o DB connectivity program:-

This program are mainly used for getting connection to the database while processing the request.

(E) Data Base:-

It is a major component in the web enterprise application that can store and maintain data as well as information.

Ex. oracle / SQL / MySQL / UDB DB2 / Sybase.

Q) Explain the Project environment for various technology like java/.Net:-

(A) JAVA App. Testing Environment

①	Technology	Jdk, Jdbc, EJB, STRUTS, Hypermedia
②	Browsers	Mozilla Firefox.
③	O/S	Window xp.
④	App Server	Tomcat apache / BEA web logic
⑤	Web Server	
⑥	DB Server	Oracle 11g.
⑦	PM - tool / software Conf' management (SCM) tools / common Repository (CR) TOOL	visual source site (vss) 6.0 / control / concurrent version system (cvs) / share point.
⑧	Testing Tools	QTP 11.0
⑨	Bug tracking Tools	Bugzilla.
⑩	TM Tools	QC (Quality center)
⑪	Script Language	VB Script.

(B) .Net APP. testing Environment

①	Technology	MS. Net frame work 4.0.
②	Browser	Internet Explorer.
③	OS	Window. XP.
④	App Server	Tomcat Apache / BEA weblogic / IBM web sphere.
⑤	Web Server	IIS (Internet info server)
⑥	DB Server	SQL Server
⑦	PM TOOLS / SCM / CRTOLS	VSS 6.0 / CVS / share point ✓
⑧	Testing tool	QTP. 11.0
⑨	Bug tracking Tool	Bugzilla / track plus / DR trackers PVC trackers.
⑩	TM Tools	QC (Quality center.)
⑪	Script Language	VB Scripting.

5) Type of Application:-

There are various types of application that are classified based on following criteria.

(a) Environmental Based :- Standalone / client server / internet, intranet / distributed application / desktop app.

(b) Nature of business - Based :-

B-2-C - Business to Customer.

B-2-B - Business to Business.

(c) Hardware capacity Based :-

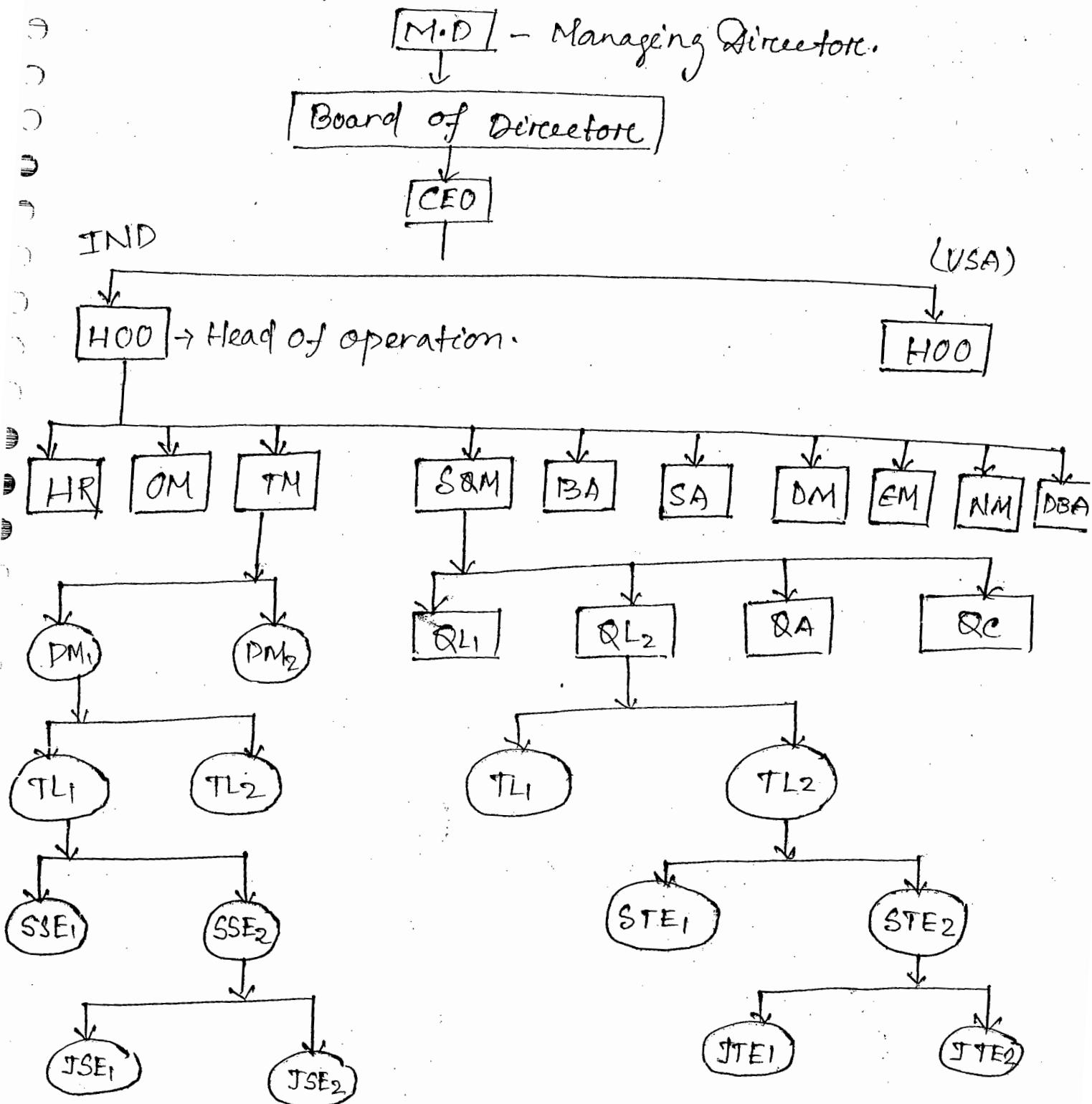
PC based, AS400, Mainframe (MF), Supercomputer based application.

(d) Technology Based :-

Java, Dotnet, C, C++, VRP, CRM, Amadeus System, Flash, multimedia Apps. etc.

(e) Domain Based :- Banking, finance, insurance, health care, e-commerce, Education, telecom etc. application.

Hierarchy of an Organization:-



PRO - Public Relation officer.

- (1) HOO/COO - Chief of operation.
- (2) OM - Operation Manager.
- (3) TM - Technical Manager.
- (4) CA - Chief Architect.
- (5) SQM - Software Quality Manager.

- (6) SA - System Analysis.
- (7) DM - Delivery / Deployment Manager.
- (8) NM - Network Manager.
- (9) EM - Engagement Manager.
- (10) DBA - Data base Administration.

Developer

PM - Project Manager.

TL - Technical Lead.

SSE - Senior Software Engineer / Developer.

JSE - Junior Software Engineer / Developer.

Tester:-

QL - Quality Leads.

QA - Quality Assurance.

QC - Quality Control.

TL - Test Lead.

STE - Senior Test Engineer / Senior Tester.

JTE - Junior Test Engineer / Junior Tester.

Communication System :-

Communication System plays vital role in making the team members effective and efficient and coordinate among each other, and produce qualitative soft.

Communication System can be effectively implemented basically with the following process.

① Intimation Process:-

It is the process in which soon after the tasks are complete it will be informed to the point of contact (lead under which we do the project)

② Escalation Process:-

It is the process in which either issues or slippages can be intimated to the high level management so has to make them understand the situation and to provide solution as soon as possible.

Escalation has to happen in terms of multiple levels like. level - 1, level - 2 and so-on.

Responsibilities:-

TE/STE

- 1) To validate the functionality scope of the application.
- 2) To coordinate testing team and test lead.
- 3) To review busni. docs.
- 4) To prepare review reports.
- 5) To communicate with BA for clarification.
- 6) To do peer review and to raise review comments.
- 7) To create test scenarios and test cases.
- 8) To do test execution.
- 9) To prepare DDP's (Defect Profile Documents)
- 10) To do regression testing.
- 11) To do various types of testing.

- 12) To prepare test log (Passed and failed test case)
 - 13) To create defect metrices.
 - 14) To do RTMC Requirement traceability matrix
 15. To attend status Meeting.
 16. To prepare defect status reports.
 17. To prepare daily/weekly status Reports.
 18. To escalate issues / slippages to TL/QL
-

TL

- 1) Responsible for 1 modular project
- 2) Coordinating b/w core test team & and QL.
- 3) To ease testing tasks and to assign them to team.
- 4) To conduct KT sessions.
- 5) To create crucial tests / resources / test scenarios and test cases.
- 6) To review test scenarios and test case doc.
- 7) To monitor test execution.
- 8) To conduct status meetings & internal review meetings.
- 9) To create minutes of meetings
- 10) To create weekly status meeting reports
- 11) To create build test and final build test Summery reports .

QL

- 1) Responsible for entire project quality.
- 2) Coordinate between TL and S&M/high level.
- 3) Management customer.
- 4) To prepare test plan for entire project.
- 5) To do the test estimation with respect to time, cost of effort.
- 6) To motivate the team.
- 7) To organize periodic proj. meetings (PPM) and Periodic Project report (PPR)
- 8) QL ensure that closure activities to the required quality achieved.
- 9) To do the risk analysis and to do root cause analysis.
- 10) To create minutes of meeting.

S&M

- 1) Responsible for all the projects and their quality.
- 2) Co-ordinate between QL - testing team and HLM / customer.
- 3) To define test strategies.
- 4) To address the test kick off meeting (KOM)
- 5) To act like facilitator of all the required test wps to the project.
- 6) To customize the template/doc. in association with SEPG group. (Software engineering process group).

- (7) To motivate team.
- (8) To ensure the final delivery of a product.
- (9) To certify the product that delivered to the customers.

Complete

Chapter - 4

STLC:- (Software testing life cycle.)

It is the process in which testing activities is done effectively, efficiently and with the great optimization. In order to produced quality product for the end.

STLC have the following phase in it:-

- ① Test sign in.
- ② Test Planning.
- ③ Test Analysis & Test design.
- ④ Test Execution.
- ⑤ Result Analysis.
- ⑥ Bug tracking.
- ⑦ Bug reporting & test reporting.
- ⑧ Regression Testing.
- ⑨ UAT (User acceptance criteria)
- ⑩ Test sign off.

① Test Sign In:-

It is the 1st phase of STLC in which the following activities are accomplished before the actual testing activities take off.

- (a) Test kick off meeting (TKOM) conducted.
- (b) The participants in the test kick off meeting are SQM, QL, TEs, STEs, JTEs
- (c) This meeting is addressed by either SQM or QL.
- (d) The main agenda of the meeting is to invoke test administration.
- (e) The main responsibilities of test administration is to make all the required I/p for testing like: I/p documents, templets, software parameters, Hardware parameter, tools required for testing, guidelines required for testing etc.) are made ready and available

TSD - Test Strategy Document-

If Harind discussed the information about the Project the strategy can be created for the testing in terms of "TSD" Test Strategy Document.

The industry can follow the strategy of testing as describe below:

- ① Exhaustive testing:- This strategy used for testing an application thoroughly for along time beyond practical parameters. Since it takes vital resources like time, money

and effort in large proportion, practically it does not advisable.

2. Adhoc testing:- This type of strategy makes the testing team to perform testing on an application randomly without following any test case document. Consequently the required quality can not be achieved. Hence that is not recommended.

3. Optimal testing:- In this strategy the testing can be carried out on the application with practically visible inputs, to achieve the required levels of quality. hence this strategy is highly recommended for the testing team.

Contented of TSD:-

TSD contains the following field of information that can describe the strategy of testing

(i) objective :- This field describes the Purpose / Goal of the TSD

(ii) Budget :- This field describes the tentative budget for testing. Usually the standard ratio of the budget allocated to development team & testing team.

are 65% and 35%, respectively. However the ratio varies from project to project depends on expected level of quality from the customers.

(iii) Kinds of testing:- The different kinds of testing like conventional, unconventional & Process testing are described here depends on the applicability.

(iv) Methods of testing:- Depends on the applicability this section describe various methods like Black Box, White Box, Gray Box and static testing method.

(v) Level of testing:- Various testing level like unit, module, integration, system and user acceptance testing are described here.

(vi) Types of testing:- The list of types of testing that are planned for the project are mention in this section.

(vii) Ways of testing:- This section can described manual testing ways and if required the ways of automation testing.

Test design technique:- This section describe various black box testing technique that are used for the project - for example, ECP, BVA, Decision table etc.

Test Configuration Management:- TCM

It is the process in which whenever the requirement is changed/Modified all the related documents in the testing must be maintained accordingly. The TCM/ SCM can be effectively implemented

Change control:- It is the process in which whenever the requirement is changed the corresponding document can must be updated accordingly to keep them in sync with each other.

Version/Version Control:-

It is the program in which whenever the modification of the requirement happened new version of the document is created with modification while the previous version is retained apart from which it provides naming convention with the document along with the version number.

Ex:-

Pr. name - Mod name - Doc name -  V#

NetBiz - Services - BDD - 1.0

SCM/TCM can be maintained in two ways manual and automation. In automation SCM tools are used like VSS, CVS, Share point etc.

Cost of Metrics:-

It is the process in which the task is measured to have the detail about it in order to ensure the clarity, credibility and profitability of the work.

Test Matrix:-

Matrix is the process in which information consistency and completeness can be ensured with specific processes CRM and RTM respectively.

CRM :- (cost reference Matrix) is the process in which the related document are cross referred with each other so as to ensure the information consistency.

RTM:- (Requirement traceability Matrix) It is the process in which either the accomplished task of the document is tested for all the requirements to ensure the completeness of the task/information. In other word it is the process to check for all the requirements that are covered in the task/document to ensure completeness.

Test terminology:-

This section covered all term that are used in the testing department.

Test automation plan:-

This section describe the following points regarding the automation.

- (1) List of functionalities to be automated.
- (2) List of functionalities that not to be automated.
- (3) Selection criteria for automation tools.
- (4) The list of selected tools for the automation.
- (5) Frame work that to be implemented for automation/automated.

Test communication System:-

The test communication can be implemented in the following ways.

(a) Meetings:- Testing team members can meet among themselves and also with the management face to face in the formal meetings, as described below.

(i) Daily meets:- Testing team members along with QTL meet daily to discuss the status of the task, Productivity and issue.

(ii) By weekly meets:- Testing team members along with TL and QL meet twice in a week.

(iii) Weekly meets:- Testing team members along with TL and QL meet we can have meeting with HOD to update the status, Productivity and to discuss the issues and sleepage.

(b) Mailing system:- The testing team can officially communicate to all the individual associated with the project with the mailing system. Mostly with the help of tool outlook express.

(c) Common Repository:- (CR)

It's a means of communication through which various documents are provided

task. Since it is a common shared repository, server and control system. Documents are preferred to be kept in the common repository.

(d) Bug tracking tool / test management tool :-

On the Bug tracking tool or test management tool acts like means of communication in such a way that the defect information can be communicated to development team, TL, QL for their respective task.

② Test Planning :-

Once the TSD is over, taking that as an input the next stage - test planning is started.

Test planning :- It is the process in which the testing activities are ensured with effectiveness, efficiency and the optimization.

Test Plan :- It is a strategy document that is created as an outcome of test planning. In other words it is the documents that describe what to test, when to test, how to test, who is testing and where testing.

I/p Document required for Test Plan :- A/p doc. or bare document, the following documents are required to prepare test plan

- (i) BDID, (which can give what to test)
- (ii) PP, (which can provide when to test)
ie ending date of development - So

(iii) VSD (That provide strategies for testing)

(iv) SRS (which can provide test environmental details)

Q) Who will prepare Test Plan?

Test plan can be understood in two ways.

① Master test plan (MTP)

② Functional test plan (FTP) which is nothing but TCD.

Hence the answer for this question depends on if the TP is MTP or FTP. In case FTP Science is TCD it is the whole responsibilities of test engineer. In case it is MTP it is the responsibility of S&L or QL to prepare this document.

Level of document :-

Depends on the scope there are 4 levels of documents.

① Organization level document:- The scope of this document is the entire organization.

Ex:- Quality Policy

Security Policy.

② Project level documents:- These documents are confined/limited for project.

Ex:- Project Plan.

Test plan.

(3) Module level documents:- These documents are limited to module. Ex:- BDD, TCD etc.

(4) Individual level documents:- They are confined to the personal level.

Ex:- Review Report
Time sheet.

Index of Test Plan:-

96.02.2015

The content of the test plan is as describe below

1) Introduction.

(1.1) Objective

(1.2) Baseline documents.

2) Test Scope

(2.1) Feature that have to be tested

(2.2) Feature that have not to be tested

3) Resource Planning.

4) ~~Requirements~~ scheduling

5) Test Strategy.

6) Test criteria.

7) Test deliverables.

8) Test environment.

9) Staffing & training.

10) Risks and contingencies.

11) Assumptions.

12) Author, Reviewer and Approve authority
Information.

) Introduction:-

1.1 Objective:-

This section describe objective/goal/purpose of the test plan document.

1.2 Base documents:-

Input documents that are required for test plan is listed out here.

Like - BDD, TSD, Project Plan, SRS.

) Test scope:-

2.1 Features to be tested:- List of all the functionality that are supposed to be tested will be listed out here.

2.2. Features not to be tested:-

List of functionality that are not supposed to be tested based on the following criteria can be mentioned here.

1. Out of scope

2. List priority base functionalities.

3. Less risk functionality.

4. Future feature.

5. Implicit requirement related functionality.

3. Resource Planning :- This section contains roles and responsibilities as described in a table.

Sl. No.	Roles	Responsibility
1	SQM	W.W.W
2	QL	W.W.W
3	TL1	W.W
4	TL2	W.W
5	STE1	W.W.W
6	STE2	W.W.W
7	JTE1	W.W
8	JTE2	W.W

(Table)

Resource Planning i.e. number of resources required for the projects depends on the following criteria.

- (1) Number of functional point / volume of the functionality.
- (2) Complexity of functionality.
- (3) Availability of time.
- (4) Availability of skill professional.
- (5) Resource available in the bench.
- (6) Level of quality accepted by the customer.

The ratio between the developer team & testing team can be maintained based on the type of project as describe ~~bolds~~ below in the table.

Type of Project.	Develop to testing Ratio.	Duration.
1. Desktop/stand alone application. 2. Simple web apps. 3. Web site (normal) 4. less complex enterprise Apps. etc	D:T 3:1	5 months to 6 months.
1. Mobile apps. 2. Embedded System app 3. Mainframe app. 4. Hardware related app. 5. Network related app. etc	1:1	7 months to 10 months.
1. Robotic apps. 2. Satellite control app. 3. Aeronautical system etc.	1:3	12 months to 18 months.

4. Scheduling:-

This section describes roles and responsibilities and also starting date, ending date, and the duration for each responsibilities as given in the table.

Sl no	Roles	Responsibility	Starting date	Ending date	Duration
1	-	-	-	-	-
2	-	-	-	-	-
3	-	-	-	-	-

(5) Test Strategy:-

[Note: Already cover in the PTSO contain.]

(6) Test Criteria:-

6.1) Entry Criteria:- The condition that are required to start the testing can be mentioned here.

6.2) Exit Criteria:- The condⁿ that are required for stopping the testing can be mentioned here.

6.3) Acceptance Criteria:- The condition that are required for stopping testing successfully are mentioned here.

6.4) Suspension Criteria:- The condⁿ that are required for stopping testing unsuccessfully are mentioned here.

7) Test deliverable:- All the listed documents created under testing department are considered to be delivered. Some of the deliverables are given below.

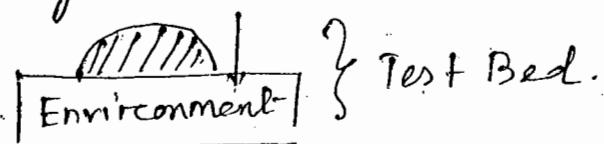
① test scenario Document (TSD)

- (2) Test Plan.
- (3) Test Strategy Document (TSD)
- (4) Test Case Document (TCD)
- (5) DPD.
- (6) Test Log
- (7) TRD - Test report Documents.
- (8) RR - Review Reports. etc.

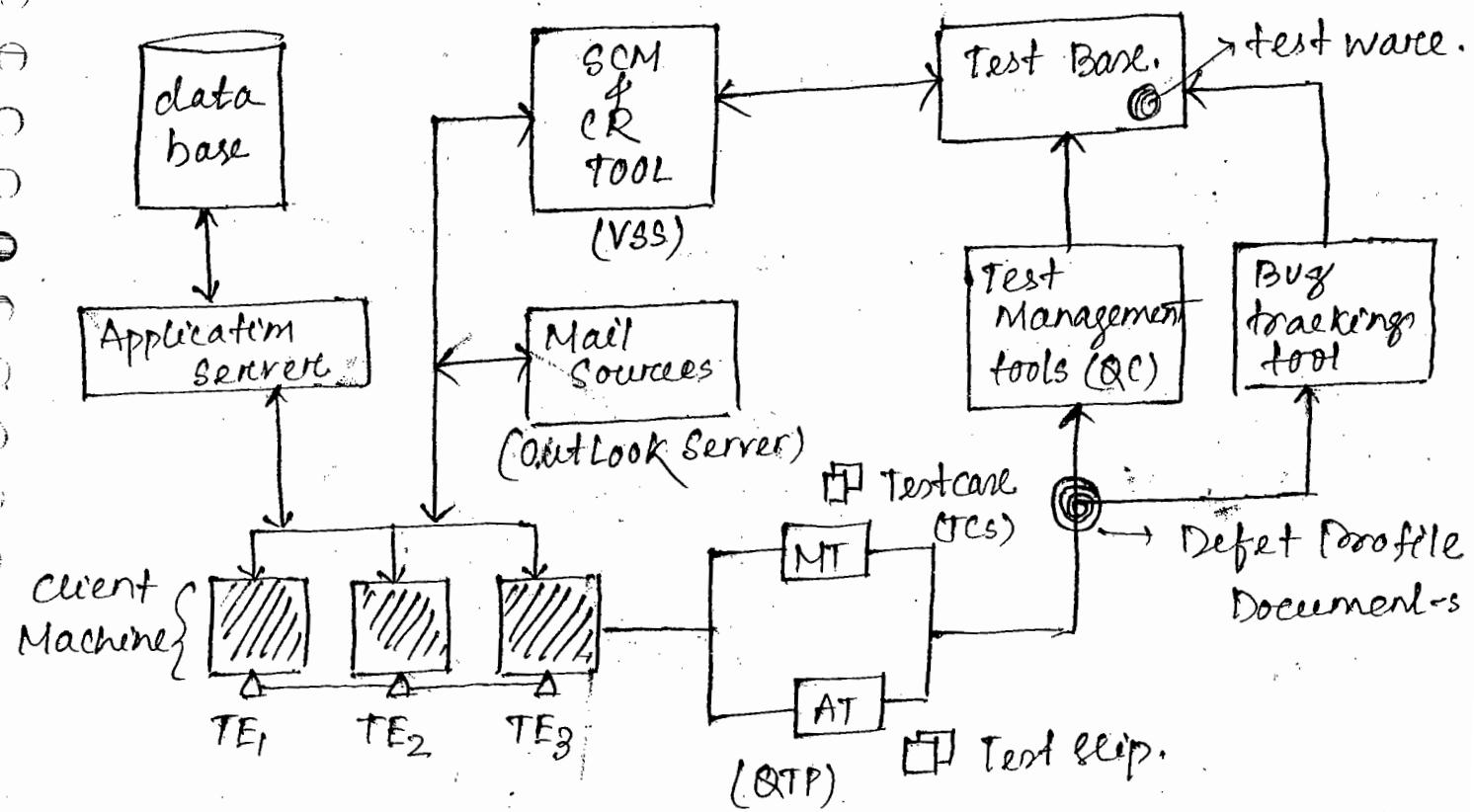
8) Test Environment:-

This section describe test environment which includes Software parameters and Hardware parameters (that are of client component, apps server component & Data base server component). Testing tools, Bugtracking tools, test management tools, Common Repository / SCM tool, Documents that are associated, templete & guideline etc.

The complete test environment along with the product that deployed is known as test bed.



Test environment Diagram:-



(9) Staffing and Testing:-

Backup plans for human resources and training plan for the team members can be describe here.

(10) Risk & Contingencies :-

This section describe the list of possible risk along with the corresponding solⁿ to nullify them in term of contingencies. Some of the risk are given below.

- ① Leak of proper documents
- ② Leak of time for testing.

(3) Lack of Resources.

(4) Lack of proper co-ordination & co-operation.

(5) Lack of proper Guideline / process.

⑪ Assumption:-

All the list of assumption can be listed out in this section.

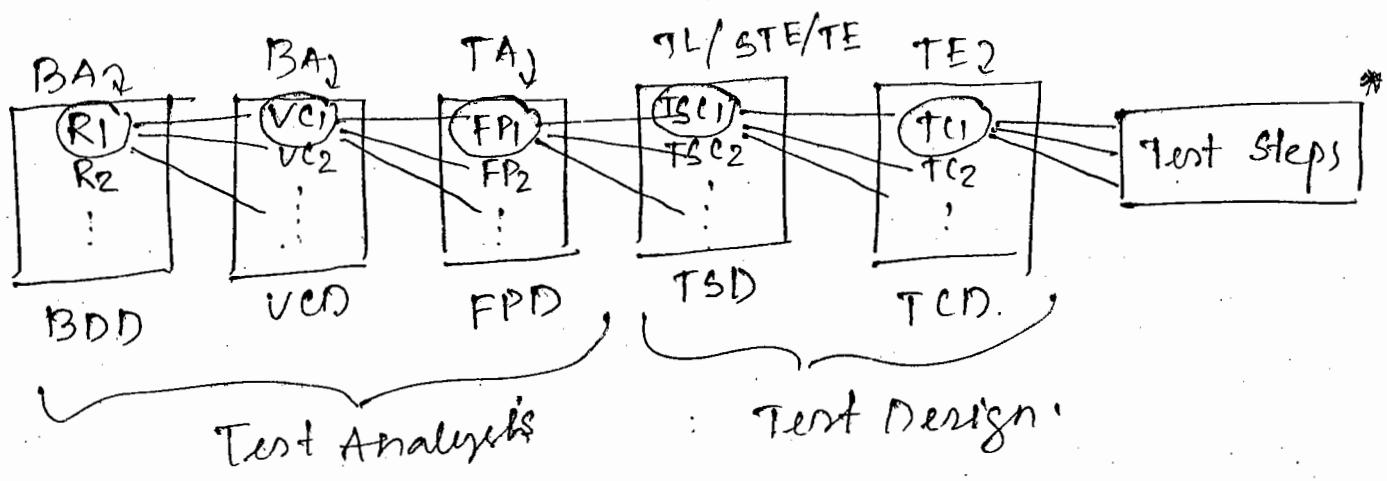
⑫ Author, Reviewer and Approve Authority information

This section deserve Author, Reviewer and Approve Authority information.

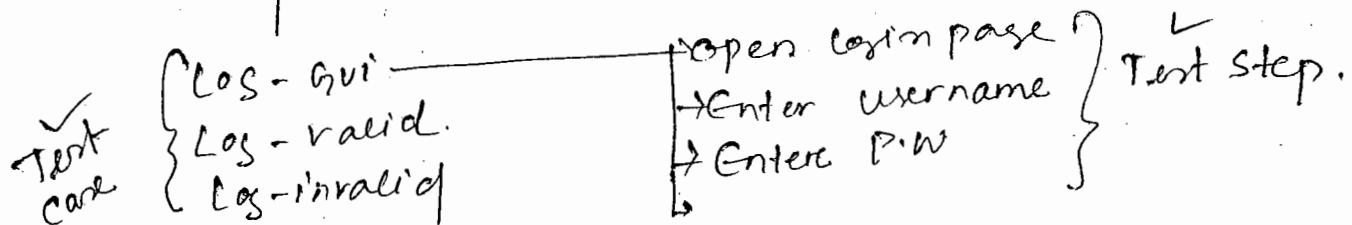
⑬ TEST ANALYSIS of TEST DESIGN:-

⑬.1 Analysis:- It is the process in which the Test Analysts review BDD and UCD and identified the list of functional area to be tested in terms of functional point. These functional points are listed out in separate document are known as function point Document (FPD).

V'case / use case:- It is the combination of actors, actions, and their responses in terms of depiction (Pictorial diagrams). Usually they are UML (Unified modified language) diagram created by BA.



Ex:- Login check Test Scenario.



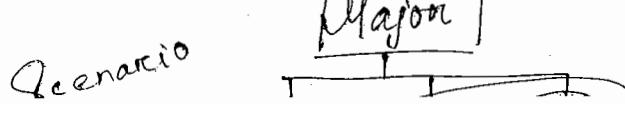
3.2 Test Design :-

It is the process in which the testing team can review BDD, VCD, FPD, TSD and creates/identifies various test scenarios and eventually creates detailed test cases.

Test design can be done in two level as described below.

(3.2.1) High level test design:- It is the process in which the testing team identifies various test ~~Scenarios~~ Scenarios from the functionality of an application.

Scenarios:- Scenario is defined as a task or a combination of task / functionality or a combination of functionalities which is a part of major functionality.

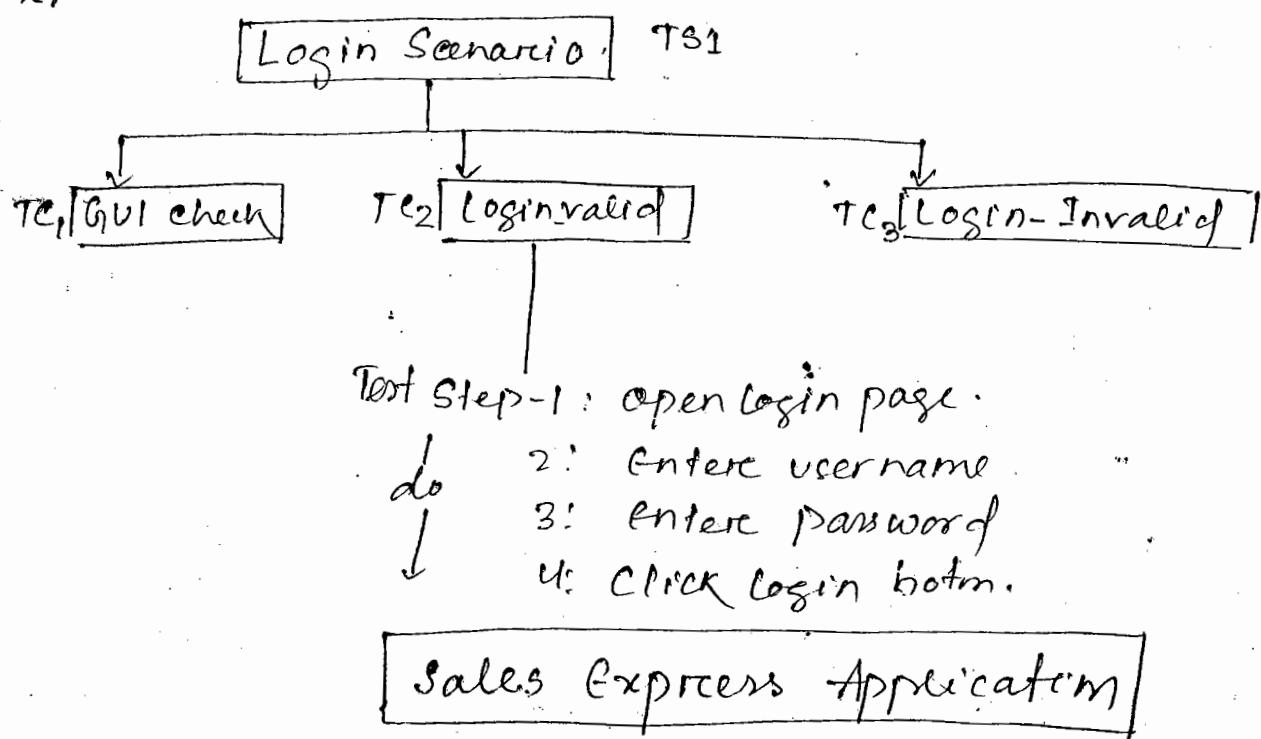


Note:- Scenario can be further divided into multiple test cases.

(3.2.2) Low level design:- It is the process in which detailed test cases are created based on the respect to test scenario.

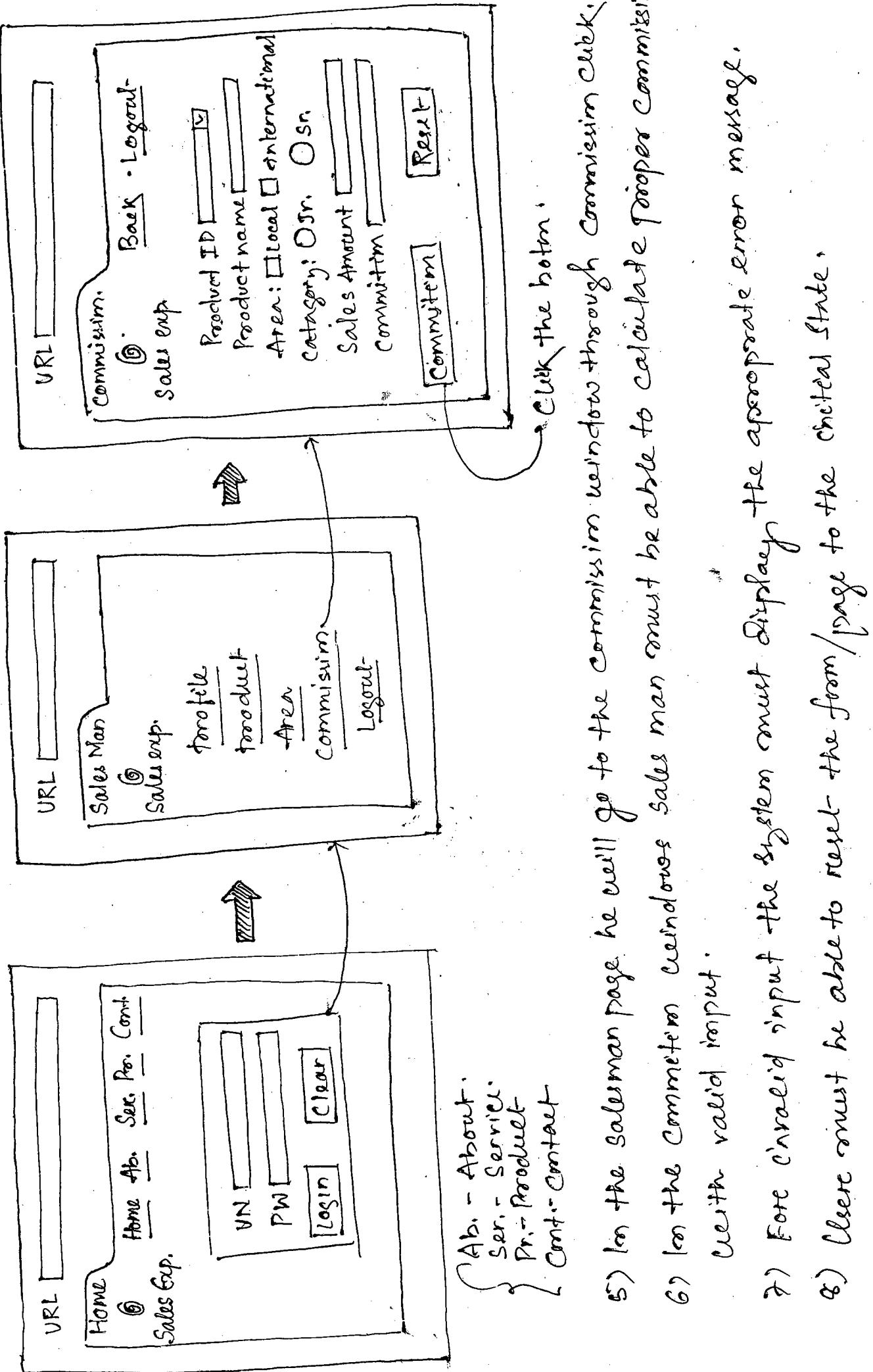
Note:- Test case can further divided into test step.

Ex:-



Functional Point of the next application:-

- ① Sales Express app can be launch with the URL - www.SalesExpress.com.
- ② With valid input- salesman can enter into the corresponding personal page.
- ③ With invalid input the system must display the appropriate error message.
- ④ The user is able to clear the unwanted entries in



- 5) In the Salesman page he will go to the commission window through Commission click.
- 6) In the Commission window sales man must be able to calculate proper commission with valid input.
- 7) For invalid input the system must display the appropriate error message.
- 8) User must be able to reset the form/page to the initial state.

27/02/2015

Template for test Scenario Document:-

Test Scenario ID	Main functionality	Sub functionality	Test scenario name	Reference	Testcase name	Testcase Description
TS-001	SE	HP	TS_SE_HP_UI	VCD/RSS	TC_SE_HP - UI-LF TC_SE_HP - UI-AUL TC_SE_HP - UI-Serl.	To verify UI of Home Page.
TS-002	SE	HP	TS_SE_HRMP			

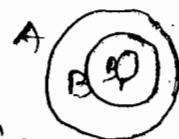
Test Scenario ID :- Unique number given to the Scenario that is identified.

Main functionality :- It is the parent functionality from which Scenario is identified.

Sub functionality :- It is the child functionality from which Scenario is identified.

Test Scenario Name :- It is the name of the Scenario that usually starts with the prefix TS and reflex Particular Scenario along with the Parent and Child Scenario functionality.

Ex:- TS-A-B-B₁,



whereas TS indicate Scenario,
A is main functionality,
B is sub functionality,
B₁ is specific Scenario.

Referance:- this field has the basic information available from BDD/VCD/FPD depends on which the scenario is created.

Ex:- BDD/page 35 / see 1.2.1

Test case Name:- this field contains the name of the cases that are identified under each scenario.

Ex:- TC-A-B-t1

or TC-A-B1-t1

→ where TC indicates test case.

→ A and B are Parent and child functionality respectively.

→ B1 is the scenario.

→ t1- specific test case.

Test case Description:- this field describes one line description of each test case.

Sale Express

Homepage

1) UI

2) ML

3) Logout

4) Clear

Homepage

Salesman

1) UI

2) ML

Salesman

Commission

1) UI

2) ML

3) Comm.

4) Reset

Test RC.ID	Main func	Sub func	Test Scenario Name.	Ref.	Test case Name	Test Description
S-001	SE	HP	TS_SE_HP_UI	VCD/PSS	TC_SE_HP_UI_LF	To verify look and feel of HP.
S-002	SE	HP	TS_SE_HP_ML	"	TC_SE_HP_ABUL TC_SE_HP_HBL TC_SE_HP_SerL TC_SE_HP_Proj.L. TC_SE_HP_Con.L	verified About link ver. Home Link Ver. Service Link Ver. Proj. Link Ver. Contact Link
S-003	SE	HP	TS_SE_HP_Login	"	TC_SE_HP_Login valid TC_SE_HP_Login invalid	veri. Login Link.
S-004	SE	HP	TS_SE_HP_Clear	"	TC_SE_HP_Clear valid TC_SE_HP_Clear invalid	veri Clear Link.
S-005	HP	SM	TS_HP_SM_UI	"	TC_HP_SM_UI_LF	verified look & feel of SM Page.
S-006	HP	SM	TS_HP_SM_ML	"	TC_HP_SM_ML_Profi TC_HP_SM_ML_Prod. TC_HP_SM_ML_Area TC_HP_SM_ML_Conn. TC_HP_SM_ML_Logout	verified Profi Link " Prod. Link " Area " " Conn. " " Logout link.
S-007	SM	COM	TS_SM_COM_UI	"	TC_SM_COM_UI_LF	Ver. look & feel of com page.
S-008	SM	COM	TS_SM_COM_ML	"	TC_SM_COM_ML_BK TC_SM_COM_ML_Logout	verified back link verified Logout link
S-009	SM	COM	TS_SM_COM_COM	"	TC_SM_COM_COM Valid TC_SM_COM_COM_invalid	To verify comm. Link.
S-010	SM	COM	TS_SM_COM_Reset	"	TC_SM_COM_Reset	To verify Reset link.

Low Level Design :-

It is a process in which detailed test cases are identified and documented and finally produce TCs. for the given scenario.

Case Study No-1 :-

Prepare test scenarios for the above 3 screens that are Sales Express application.

A. Purpose of test case :-

- (1). To carry out testing in a Planned manner.
- (2). To organised testing in a Proper manner.
- (3). To estimate testing effort.
- (4). To ensure test coverage.

B. Type of test cases :-

Different types of test cases are classified based on the following criteria.

(i) What to test

1.1) GUI Test cases:- For testing the look & feel of application.

1.2) Functional test cases:- for testing the functionality of an application. There are 3 types of functional test case.

1.2.1) +ve test case:- used for testing +ve flow with the help of valid / p. data.

1.2.2) -ve test case:- used for testing -ve flow with the help of invalid / p. data.

1.2.3) Performance test case:- Use for testing performance aspects of

- (a) Load test cases: to test the load bearing capacity of app.
- (b) Performance test cases:- use for testing the responds time for app.
- (c) Stress test cases:- use for checking the stability of application.
- (d) Volume test cases:- use for checking the data handling capacity of app.

(2) How to test :-

(2.1) Detail testing related test cases:-

Ex:- function test cases .

(2.2) Non Detail testing related test cases:-

Ex:- Smoke / sanity test cases .

(3) Based on functional factor :-

(3.1) Functional test case:-

(3.2) Non Functional test case:- Used for testing non functional aspects of an application .

Ex:- GUI, Usability, compatibility, Configuration, installation, Inter operability, performance, security test case

Co TCD - Template Study:-

TCD- templates have the following fields.

Header info	TCD-ID :- _____	Reviewed by :- Panini
	TCD Prepared by :- Abhilash Dey	Reviewed on :- _____
	TCD Prepared on :- _____	Approved by :- Jagdish
		Approved on :- _____
<ol style="list-style-type: none">1. <u>Objective</u> :-2. <u>Test Scenario</u> :-3. <u>Test Procedure</u> :-4. <u>Detail TCs</u> :-		
Test case table.		
	Table 1	Table 2
	

Header information:- This section contain the information about the author, Reviewer and approval authorities along with the respective dates.

Test objective:- this section describe the purpose of the TCD prepared.

Test scenarios:- This section describes the list of scenarios for which the detailed test case are created.

Test Procedure:- This section describes testing procedure in terms of GUI testing, +ve testing & -ve testing to cover the visual look and feel, +ve flow and -ve flow with the help of GUI test cases, +ve test cases and negative testcases respectively.

* Detail test cases:- This section describes list of detailed testcases. When the test cases are created the following points are to be followed.

- (i) Create the test case in tabular form-
- (ii) Test cases can be written under the classification of GUI, +ve and negative test case.
- (iii) Use hyper linkage ^{technique} when test cases are created. to ensure that the test case is isolated from the test data.
- (iv) write the test case in term of instruction rather than test statement.
- (v) when testcase are written testcase guideline can be followed
- (vi) when writing test cases use of test design technique (Black Box Design Technique)

* Test case table:-

Test case table have the following field:

- (a) test case ID
- (b) test case Description.

- (C) Expected value.
- (D) Actual value.
- (E) Result
- (F) Seriarity.
- (G) Prioricity.
- (H) Reference.

1. Test case ID:-

Every test case must have unique ID in order to uniquely identify, easily and quickly excess the test case.

2. Test case Description:- This field describe what to be done while testing in terms of instruction to the tester.

3. Expected Value:- It's nothing but expected behavior of an application.

- 3.1) Expected value is always derived from requirements.
- 3.2) Expected value is always determined before testing.
- 3.3) Expected value is derived by the/determined by the test engineer.

4. Actual value:- It's the actual behavior of an application.

- 4.1) Actual value is always derived from the application.
- 4.2) Actual value is always determined while testing.
- 4.3) Actual value is derived by the developer.

5. Result:- This field describe the test results in terms of Pass or fail depends on matching or mismatching when ~~are~~ expected value are compared with actual value.
6. Serriarity:- This is an expression that describe the importants of test cases.
7. Priority:- It is an expression that describe the sequence of the test cases to be executed base on the Serriarity.
8. Refarance:- This field describe the path of information based on which the test case are created.

Test data:- It is basically a data used for testing specific functionality to ensure functional robustness. Usually this data is kept in terms of data table with specific names that are made available tho to the tester while testing through hyperlink.

NOTE:-
Test data is basically prepared by the test engineers; sometimes provided by Project Manager (PM); Rearly it comes from the customers in term of live data.

Do Criteria for good test case:-

1. A test case with high probability of catching the defect.
2. A test case which is written in term of instruction but not in statement.
3. A test case must be reusable.
4. A test case must be reproducible (must reflect the meaning consistently).
5. A test case must be / should be neither too simple nor too complex.
6. A test case must cover the entire functionality.

E. Guidelines for test cases:-

There are 3 types of guidelines provided for writing 3 types of testcases as described below.

1. GUI Test case Guidelines:-

As per as the GUI test case are concerned the following guidelines can be followed.

- (1) To check the Availability for all the feature/object
- (2) To check the alignments

Note:- If requirement are associated with alignments, the tester must go by the alignments; In case no requirement the tester must go by universal standard.

- (3) To check for the consistency. (from the point Colours scheme / size of objects / font type)
- (4) To check the Spelling / grammar.
- (5) To check the important feature in the screen.

2) Positive test case guide lines:-

1. The tester must have the mind setup.
2. The test engineer must focus on the flow of an application.
3. The test engineer must use of the valid data to perform the testing.

3) Negative test case guide lines:-

1. The tester must have the mind setup.
2. The test engineer must focus on the flow of an application.
3. The test engineer must use of the invalid data to perform -ve testing.

Review Process of TCD:-

Once the TCD is prepared it will undergo the review process as describe below

1. Self Review:- Review done on TCD by the author i.e. test engineer.
2. Peer Review:- ~~Author~~ Authors will exchange the documents and perform review on each other's TCDs.
3. Final Review:- TCD is finally reviewed by BA ~~by~~ for approving it finally.

(Note:- Review process mostly ensured correctness and consistency)

No RTM updation:- (Requirement traceability Matrix)

It is the process in which the test cases are verified in such a way that they have covered all the requirements or not to ensure the completeness of test design.

As a part of this activities the tester will enter test case ID in the RTM document against the corresponding requirements, to check if all the requirements are covered. As given in the table below

Req. ID	Req.	Test Scenario ID	TG IDs	Comments
R-001	=	1, 2, 3	2, 3, 4, 5, 6, 7, 8, 9, 10	
R-002	==	4, 5, 6	11, 12, 13, 14 15, 16, 17, 18, 19, 20, 21	
R-003	==	7, 8	22, 23, 24 25, 26, 27	
R-004	=	9		R-004 not covered
R-005	=	10, 11, 12	28, 29, 30 31, 32, 33, 34 35, 36	

→ No testcase mean Req. is not covered.

Note: From above blank in the table it is understood that requirement 4 is not covered in the TCD. This makes the tester to refine the TCD to the Completeness.

To Baseline TCD :-

Once the TCD is reviewed and RTM practice is done on it it will be made perfect and it finalized by BA under the process TCD Baseline.

To Publish TCD :-

It is the process in which the base line TCD is officially made available to rest of the role for the sake of official usage.

Date: 02.03.2015

Inputs Required for prepared/ preparing TCD:-

A) Prototype Screen:-

In the company where is the prototype model is used the GUI Screens are developed as Prototype screen in such company tester may use them as inputs for test design.

[Note:- It is not mandatory all the time.]

B) Functionality:- While writing testcases functionality of an application is key input

C) Business Rule:- It is explicitly emphasize condition based on which valid and invalid data are prepared.

Ex:- Business Rules are two types of.

(i) GUI Business Rule.

(ii) Functional Business Rule.

(i) GUI Business Rule :-

Condition associated with GUI feature

Ex:- Logo must be top left corner of window.

(ii) Functional Business Rule:-

Condition associated with functionality

Ex:- User name text box allows 4 to 7 char. Long alpha numeric, lowercase string.

(D) Test Case Template:-

TCO template must be provided by company

for writing test case

(E) Test case Guidelines:-

Test case guideline must be followed while writing better test cases.

(F) Validation types:-

Validation means checking the input given by the users to the application program. The developer can implement 2 types of validation.

(FLV) (i) Field Level Validation:- Checking happens at the field level itself unless the right input is given to the correct field, the user will not be allowed to enter the input for the next field.

(RLM) (ii) Record Level Validation:- Checking happens at the time of the entire record is submitted.

Note:- Test engineer can prepare test data tables depends on the type of validation.

G) Test Data:-

Before writing test cases test datatables can be prepared so that they can be hyperlinked to the test cases.

H) Test Design technique:-

The following test design ~~technique~~ technique are used in the respective situation.

(1) ECP - This technique is used when ever there is a pinpoint/straightforward o/p. And to derived valid and invalid data.

(2) BRA:- This technique is used when ever there is a range kind of i/p and to derived valid/invalid input.

[Used for test Box] Both the above

(3) Decision table technique is used for non input-operation like clicking/Selecting/Checking the check box etc.

(4) Straight transition technique:- It is used to cover the test design in systematic/sequential manner.

(5) Use case Design:- By using usecase technique one can handle user and environment in

Types of test Design:-

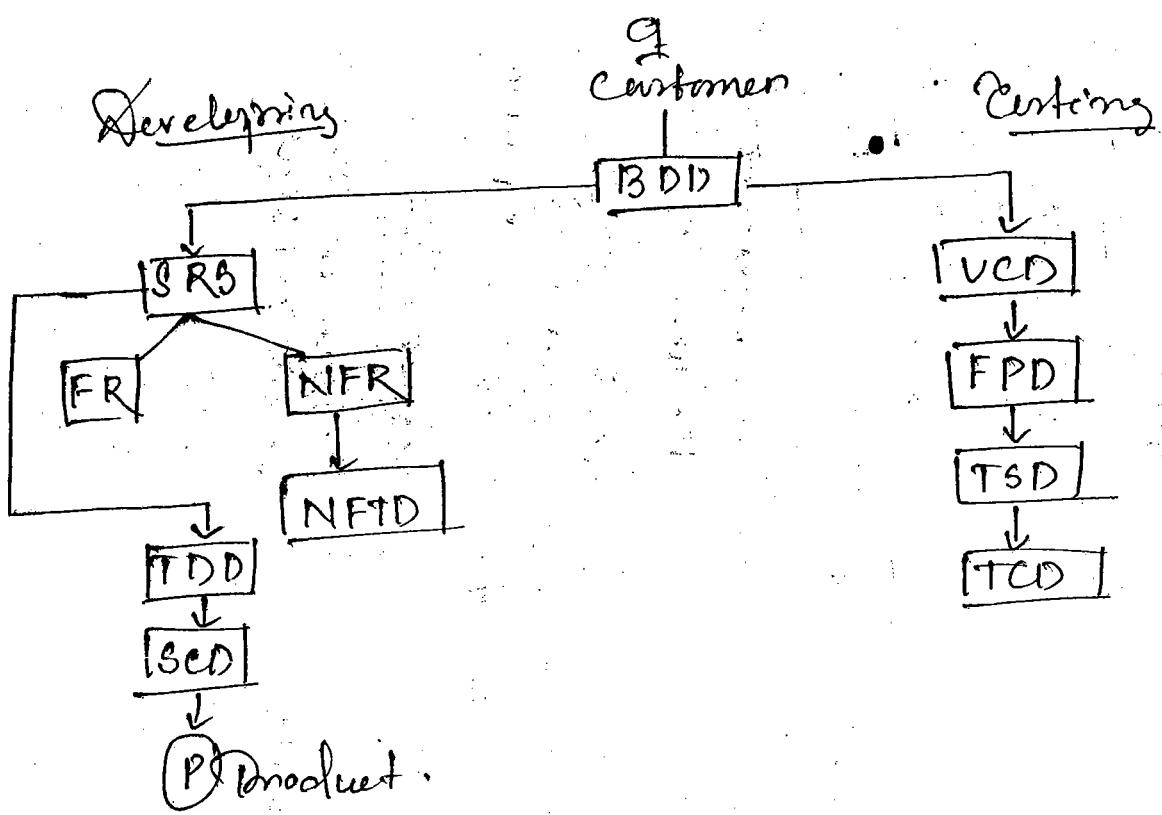
Depends on the input given to the tester there are basically 4 types of test design that he can prepared

(i) Function spec. base:- It is a type of test design in which the tester reviews business documents like BDD understand the functionality and prepare the test cases.

(ii) Use case base:- It is another type of test design for which the test engineer reviews UCD, understand the functionalities and prepared the test cases.

(iii) UI base:- (User interface). It is another type of test design in which testing team along with BA interacts with the customer gets the requirements, understand the functionalities and prepared test cases.

(iv) Non function test design:- Apart from the above 3 types of functional test design there is a non-functional test design in which the test engineer reviews non-functional requirement present in SRS. And finally prepared non-functional test cases.



Case Study - 2 :-

Preparation of test data & testcase under low level test design for the login functionality of sales case application.

Functional Spec.

Use Case Documents

Purpose: The purpose of login functionality is to validate the authenticity of the user.

Business Rule:
(i) For valid user is taken to the corresponding home page.
(ii) For invalid user the appropriate error message must be displayed by the system.
(iii) User must be able to clear unwanted entries.

Sines Rule:
1) Login box is placed in such a way that the companies logo is on the top left near user in the login box.

2) All the object in the login box must be under one line.

3) Functional Business Rule:

Sno	Obj. Name	Obj. Type	Business Rule
1	username	Text Box	U-3 ch. AN + \$ + LC
2	Password	T.B.	3-5 ch. AN + LC
3	Login	Push Button	—
4	Clear	Push Button	—

Use Case Id: U.C_SE_HP_Login.

Use Case Description:-

This use case describes login functionality of the sales express application.

Actor:- Admin user, normal user (sales man), Customer.

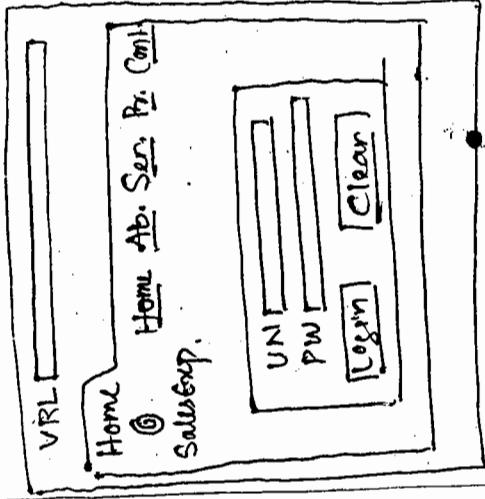
Pre Condition:-

- All the type of user must be created before login.
- User credential must be their in the database.

Post Condition:-

- Admin user must go to Admin HomePage.
- Normal user must go to its respective HomePage (Sales man HomePage / Customer HomePage)

UML diagram / Activity UML Diagram:-



- Showing the UI to the customer obtained the requirements.
- Document all these requirements in the Business Requirements Document.

Business Rules:-
Same as mention on the left side.

Developer has implemented Record level validation for Login Screen.

Process:-

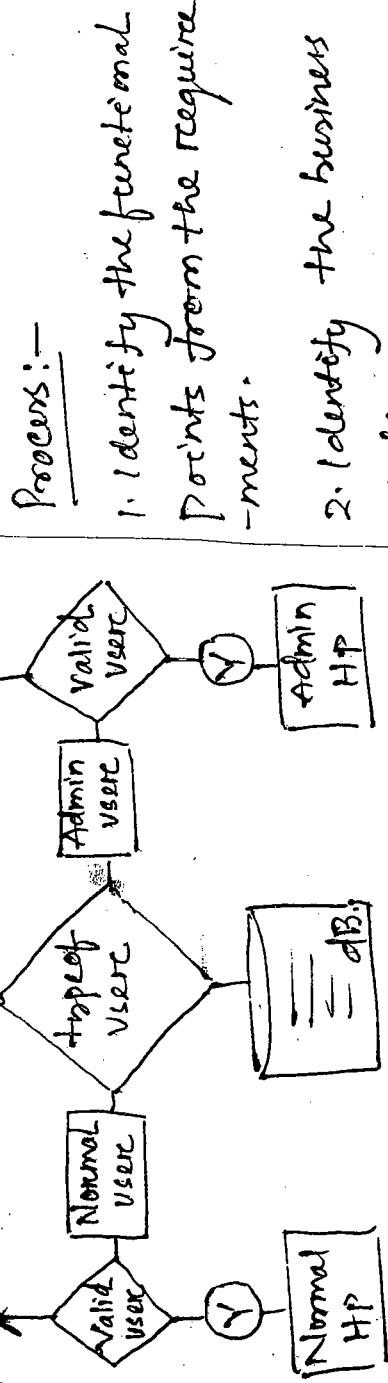
- Identified the functional point.
- Identify the business rule.
- Understand the functional point.
- Understand the Business rule.

Prepair test data.

Prepair test cases.

Test Business Rule:-

Record level validation.



Event tables:-

Maintable

Sr. No	Action	Response
1.	Enter www.abc.com in URL Box	HP of SE with login screen appears.
2.	Enter V-UN & V-PI & click login.	Respective HP must be displayed
3.	Enter Invalid UN.	Go to A7-1
4.	Enter Invalid PI	Go to A7-2
5.	Enter value of Click on Clear	Go to A7-3

- Identify the functional points from the requirements.
- Identify the business rule.
- Understand the functional points.
- Understand the Business Rule.
- Prepare test data.
- Prepare test cases.

Alternate table -1

SL No	Action	Response
1	Enter IV- UN if valid-PW & click on log in Path.	System must display appropriate error message.

Alternate table -2

SL No	Action	Response
1	Enter V- User Name and Invalid - PW & Click on login Path.	System must display appro. error message

Alternate table -3

SL No	Action	Response
1	Enter Valid UN & PW and click on clear	System must clear all the data.

Business Rule:-

- Identify the function point.
- Identify the Business Rule.
- Understand the function point
- Understand the Business rule.
- Prepare test data.
- Finally prepare test cases.

Same as mention at the left side.

Validation type:- Record level validation.

Process:- Side.

Preparation of Test Data: —

- (1) observe the business rule and understand them.
- (2) understand and identified that the business rule associated with input (pinpoint / range) and non-input operators based on which various test design technique are planned
- (3) Using this technique test engineer must derived valid and invalid data conditions.
- (4) Create and identify and create valid condition and thereby create valid test data tables, base on the following criteria.

(a) Min - Table

(b) (Min + 1) - Table

(c) (Middle) Table

(d) (Max - 1) Table

(e) Max Table.

Username	<input type="text"/>	Mandatory field
Password	<input type="text"/>	
E-ID	<input type="text"/>	
Category	<input type="text"/>	
<input type="button" value="Login"/>	<input type="button" value="Clear"/>	optional field

From the above diagram another criteria can be consider as follows.

(f) (All M-valid + All opt. valid)

(g) Similarly create invalid test data table based on following criteria.

(a) (Min - 1) Table

(b) (Max + 1) Table

(c) 1 mandatory - Invalid if Rest of valid (1M-IV & R-V)

(d) 1 mandatory - Blank if Rest of valid (1M-B & R-V)

(e) 1 mandatory - Invalid (All-M-IV & All-Opt-V)

- (f) All-mandatory - valid & opt. invalid (All-M-r & All-opt.-r)
- (g) All-mandatory-blank & opt. valid (All-M-B & opt.-r)
- (h) All-mandatory-blank & All opt. blank (All-B & A-O-B)

Valid data table:-

T.D - ECP (S.T) straight Target
BVA - R.

Table-1

minitable

SLNo	Name of obj.	Type of I/p	Actual i/p
1	UN	4ch.	abcd, abc1 1234
2	PW	3ch.	abcd, ab1 123

Table-2

(min+1)

Table.

SLNo.	Name of obj.	Type of I/p.	Actual o/p.
1	UN	5ch.	abcde, ab123 12345
2	PW	4ch.	abcd, ab12 1234

Table-3

Middle table.

Consider(s)

SLNo.	Name of obj.	Type of I/p	Actual o/p.
1	UN	5ch.	abcde, abc12 12345
2	PW	4ch.	abcd, ab12 1234

Table-4

middle value.

Consider(s)

SLNo	Name of obj.	Type of I/p	Actual o/p.
1	UN	6ch.	abcdef, abc123 123456
2	PW	4ch.	abcd, ab12 1234

Table-5

(max-1) Table

SLNo	Name of obj.	Type of I/p	Actual i/p
1	UN	6ch.	abcdef, abc123 123456
2	PW	4ch.	abcd, ab12 1234

Table - 6
(max) table:-

SLNo	Type of Obj.	Type of IP	Actual IP.
1	VN	7ch.	1234567, abcdefg abcd123
2	PS	5ch.	12345, abcde abc12

All mandatory and optional value:-
 when creating this table assume that-
 employ ID must be the type numbers and valid Category
 are employee / customer.

Table - 7

All mandatory valid - All optional valid.

SLNO	object name.	type of field	Input
1	VN	4ch.	abcd.
2	PW	3ch.	abc
3	E. ID.	Number.	001
4	Category	E/c	Employee

Invalid test data table :-

Table - 8

(min-1) table.

SLNO	Object name.	Type of field	Input
1	VN	3ch.	abc 123, abi
2	PSW	2ch.	ab 12 a1

Table - 9

(max+1)

Table ..

SLNO	Object name	Type of field	Inputs
1	UN	8 ch.	abcdefg 12345678
2	PW	6 ch.	abcdef 123456

Table - 10

All mandatory - Invalid and Rest of valid.

Table - 10.a

SLNO	Object name	Type of field	Inputs
1	UN	3 ch.	abc, 123
2	PW	3 ch.	abc, 123

Table - 10.b

SLNO	Object name	Type of field	Inputs
1	UN	4 ch.	abcd, 1234,
2	PW	2 ch.	ab, 12

Table - 11

All mandatory - Blank - Rest of valid.

Table - 11.a

SLNO	Object name	Type of field	Inputs
1	UN	-	-
2	PW	3 ch.	abc, 12, 3

Table - 11.b

SLNO.	Object name	Type of field	Inputs
1	UN	4 ch.	abcd, 1234
2	PW	-	-

Table-12

All mandatory valid - All optional invalid

SLNo	Object name	Type of field	Input
1	Username	3ch.	123, abc
2	Password	3ch.	12 , ab
3	EID	Number	001
4	Category	E/C	Employee

Table-13

All- Mandatory valid - All optional invalid

SLNo	Object name	Type of field	inp.
1	Username	4ch.	abcd
2	Password	3ch.	abc
3	EID	Character	abcd
4	Category	testengineer	tester

Table-14

All - mandatory Blank - All optional valid

SLNo	Object name	Type of field	inp.
1	Username	-	-
2	PW	-	-
3	EID	Number	001
4	Category	E/C	Employee

Table-15

All-mandatory Blank - All opp - Blank

SLNO	Object name	Type of field	zip
1	Username	—	—
2	Password.	—	—
3	EID	—	—
4	Category	—	—

Date:- 03.03.2015

Login Object Table

SL.No.	Object/feature.
①	username text box.
②	username label
③	Password text box.
④	Password label
⑤	Login button.
⑥	Clear button

Test Document for login functionality.

1. Header Information :-

TC_SE_HP_Login.

Object::- TCD ID:-

TCD Prepared by:- Abhilash Dey

TCD Prepared on:- 2 days.

Reviewed by:- Panini

Reviewed on:- 2 days.

Approved by:- BA

Approved on:- 4 days.

2. Objective:-

The purpose of this TCD is to validate the authenticity of user, in other words to test the login functionality.

3. Test Scenario:- TS_SE_HP_Login

(or)

This TCD is executed whenever the user tries to access the application.

4. Test procedure:-

To prepare GUI testcases to test/cover the Look & feel of an application under GUI testing;

To prepare +ve testcase to cover the +ve flow of an application under +ve testing;

To prepare -ve testcase to cover the -ve flow of an application under -ve testing;

Note:-

Test case number, test case description, expected value, severity, priority and reference column are filled before testing.

Actual value column is filled while testing.

Result column is filled after testing.

Test Case Step.	Test Case Description.	Expected value.	Actual value	Result	Severity	Priority	Reference.
	GUIT TESTING CASE	CASE (TC_SE_HP_Login_UT)					
-001	check the availability of all the objects and features .	All the objects/features must be available as per login_obj_table.					
-002	check for the alignment of object/features	All the object must be certain line.					
-003	check for the consistency of object/features .	All the objects of feature must be consistent.					
-004	check for the spelling/grammer of the font.	The font shouldn't be associated with spelling/gram. mistake.					
-005	check for the position of logo with respect to login screen.	Logo must be the top left corner of screen.					
S-N06	Enter Username & Password as per MinTable-1 click Login.	Positive test case TC_SE_HP-Login-Valid.	Enter Username & Password as per MinTable-1 click Login.	System must display Corresponding home page.			do.
S-007	Logout						

Test Case ID	Description	Test Data	Test Result	Notes / Reference
-009	Enter Username & Password as per mid-table. Click on Login.	do —		
-009	Enter Username & Password as per (max-1) - table. Click on Login	do —		
-010	Enter Username & Password as per max_table. Click on Login.	do —		
-011	Enter Username, Password, E.Id category as per all M_valid & all opt. valid click login.	do —		
-012	Enter Username & Password & click clear tc_SE_HP_Clear_valid (complit)	System must clear the entries.	+ve Test Case (TC_SE_HP_Login_invalid)	
-013	Enter Username and Password as per (min-1) table & click login.	The system must be display appropriate error message		
-014	Enter Username and Password as per (max+1) table & click Login.	do —		

Test Step	Test Case Description.	Expected value	Actual value	Result	Priority	Reference
3-015	Enter UN & PS as per All man - IV - Rest valid table 10 click login.	— do —	— do —	—	—	—
3-016	enter UN & PS as per All man - Blank - Rest of valid table - 11 click login .	— do —	— do —	—	—	—
3-017	enter UN, PW, EID of category as per All man - 11 - f - All opt. valid. click login . table-12	— do —	— do —	—	—	—
3-018	Enter UN, PW, EID of category as per all man - v - f - All opt. invalid. click login . table-13	— do —	— do —	—	—	—
3-019	All - M - B f All op. valid table - 14 click login .	— do —	— do —	—	—	—
3-020	All - M - B f All - opt . Rest blank table -15 click login .	— do —	— do —	—	—	—
3-021	Enter UN, PW, EID of category and click clear. it clear all the entries .	—	—	—	—	—
3-022	Put UN , PW, EID of category blank and then click clear.	System must be retain with login screen as it is preferable with user friendly message. nothing is enter to clear	—	—	—	—

Case Study - II :-

Prepare the test data as well test cases in order to validate Commission windows functionality.

Functionality:-

- (a) The purpose of the commission windows is to calculate the appropriate commission based on the sales amounts.
- (b) For valid input commission has to be calculated properly.
- (c) For invalid input appropriate error message has to be displayed.
- (d) The user must be able to reset the windows to the initial state.
- (e) 10% commission for Juniors and 15% commission for Seniors.

Business Rule:-

- (f) User must be able to go back to the previous screen.
- (g) User must be able to log out.

Business Rule:-

(a) GUI Business Rule :-

- 1. Logo must be on the top left corner of window.
- 2. All the objects must be center aligned.
- 3. ~~Logo~~

(b) Functional Requirements:-

SLNo.	Object name.	Object type.	Business Rule.
1.	Product ID	Combo Box	"Select" is the text by default.
2.	Product name	Text Box	"Auto field" non editable.
3.	Local	Check Box	Default Selection.
4.	International	Check Box	No Default Selection
5.	Female	Radio button	— do —
6.	Female	— do —	default Selection
7.	Sales amount	Text box	Editable. Numbers only
8.	Commission — TB	Text Box	"Auto field" non editable.
9	Commission — PB.	Push Button	—
10	Reset	Push Button	—
11	Back	Link	—
12	Logout	Link	—

Validation type:

Developer has implemented field level validation for the commission window.

Data table Preparation:- (Datatable + Field level)

Since it is a field level validation and there is a lot of scope for non-input operation, decision table technique may be used for creating several decision table as given below.

DT - Decision table

Table-1 (Product ID - DT)

SINo	Operation	Response
1	Observe Product ID Combo box	By default 'Select' is displayed
2	Click the button of Combo box	drops down all the options
3	Select an option	Selected option must be displayed

Table-2 (Product name - DT)

SINo	Operation	Response
1	Keep the cursor in the text box	The text box allows the cursor to be placed
2	Enter text in the text box	Entry is not allowed
3	Try to edit the text	Editing is not allowed

Table-3 (Local_DT)

SLNO	operation	Response
1.	observed the object	By default, the check box is Selected.
2	DeSelect the check box	Check box must be successfully De Selected
3	ReSelect the check box	The check box must be once again Selected successfully

Table-4 (International_DT)

SLNO	operation	Response
1.	Observed the object	No default Selection.
2.	Select the check box	must be Selected successfully.
3.	Uncheck the check box	Check box must be de-selected.

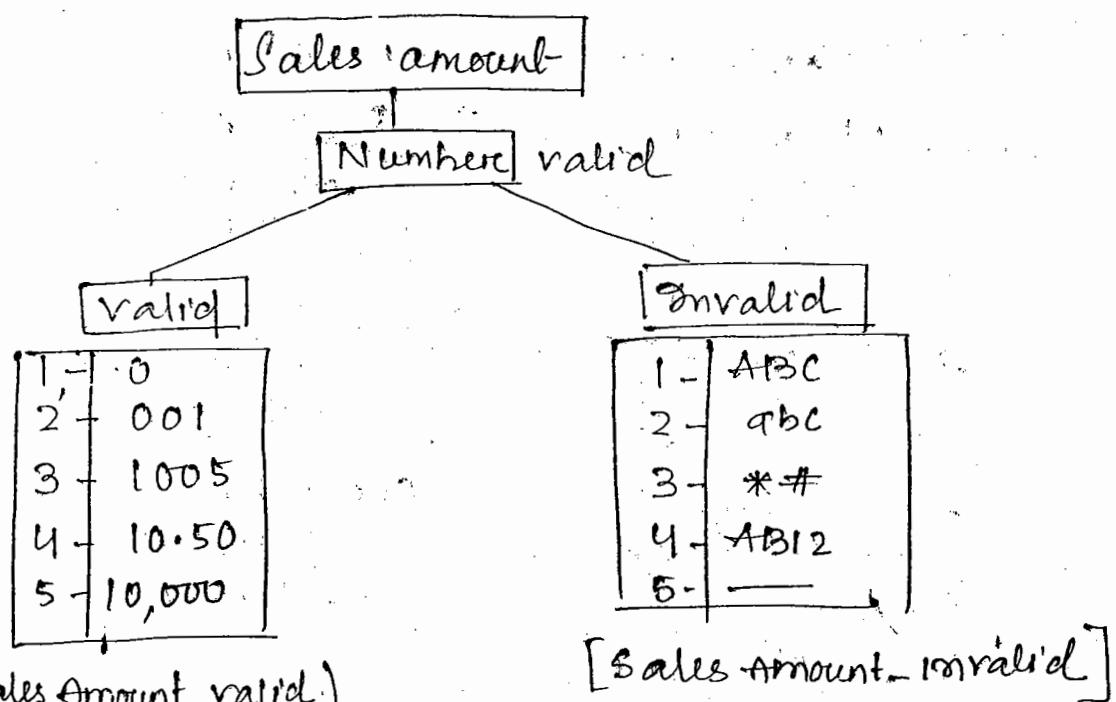
Table-5 (Tunior_DT)

SLNO	operation	Response
1	observed the Tunior radio button	No default Selection.
2	Select the radio button	radio button must be selected successfully
3	Select other radio button	This radio button is de-selected successfully

Table-6 (Seniorz-Dr.)

Sl.no.	Operation	Response
1	Observe radio button	No default selection.
2	Select the other radio button	The radio button must be de-selected.
3	Reselect the radio button	It must be selected successfully.

Table-7



Sales Amount:- Since Sales amount field is ^{scope} text box and editable there is a ~~rule~~ for input operation. Hence ECP technique can be used to determine the following valid and invalid tables. for the business rule - "Numbers only".

table-8 (Commission_DT) (text box) (Commission_TB-DT)

SNo	operation	Response
1.	Keep the cursor in the text box	the text box allows the cursor.
2.	Enter text in the text box	Entry will not allow
3.	Modified the text	Modification is not allowed

table-9 (Commission_PB-DT)

SNo	operation	Response
1.	Fill all the field and click on Commission button	The appropriate commission must be calculate
2.	Without entering /Selecting anything click on commission .	Error Message

table -10 (Reset-DT)

SNo	operation	Response
1.	make all the entries and click on Reset	Clear all the entries and window is kept initial state
2.	Don't enter>Select anything and click Reset	Window has to be retain as it's

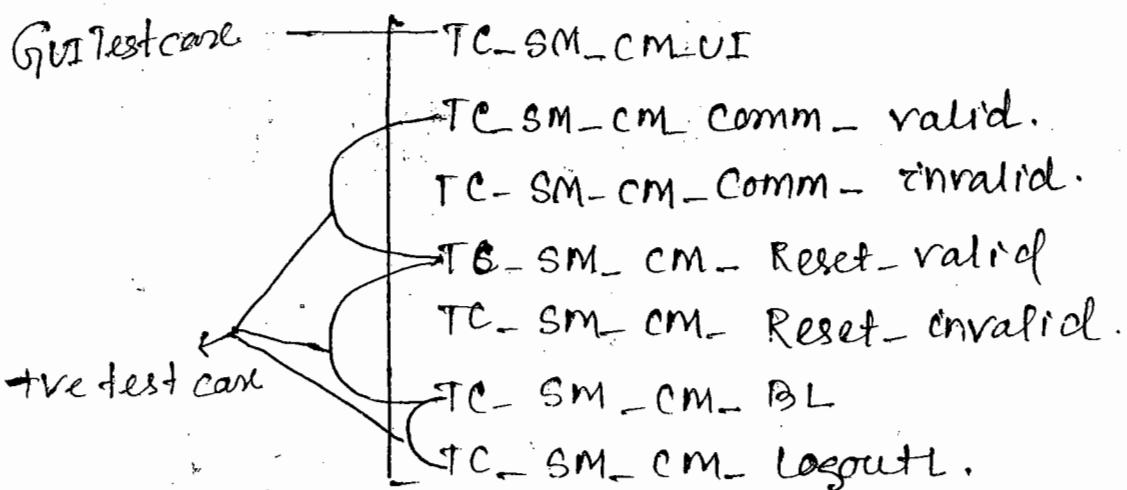
table-10 (Back_DT)

Sl no	operation	Response
1.	Click on Back Link	must take the user to previous window

table-11 (Logout_DT)

Sl no	operation	Response
1.	Click the Logout link	System must display the homepage with the login window.

Test cases for Commission Window Condition apply:-



Test case step	Test case Description	Expected value.	Actual value	Result	Severity	Priority	Reopen co.
					Severity	Priority	Reopen co.
1	Check for the availability of object / feature.	All the objects / features must be available as per Commis. Obj - table)					
2	Check for alignment of the object	All the objects must be centerline.					
3	Check for Consistency of object / feature.	All object feature must be Consistency.					
4	Check for the spelling / grammar of the text.	this font should not be associated with Spelling & Grammar mistake.					
5	Check the position of logo with respect to login screen.	Logo must be on the top left corner of the login screen.					

TC-SM-CM-BL / TC-SM-CM-L4.

Response as per the
table.

Do the operation as
per product ID-DT
and check for the
responses.

Do the operation as
per product ID-DT
and check for the
responses.

Do the op^rs as per
local DT and check for
the response.

Do operation as per
international - DT and
check for the response

Do operation as per
Junior - DT and check
for the response.

Step	Description	Expected value	Actual value	Result	Severity	Priority	Reference
11	Do operation as per Senior-DT and check for the response Enter the value as per Do operation as per Sale amount valid and successfully) Check for the response	— do —					
12	Do operation as per Comm-FB-DT) & check for response Sale amount valid and successfully) Check for the response	Response as per table.	— do —				
13	Do operation as per Comm-FB-DT) & check for response	— do —					
14	Do operation as per (Comm-FB-DT) & check for response	— do —					
15.	Do operation as per (Rest-DT) & check for response	— do —					
16	Do operation as per (Back-1ST) & check for response	— do —					
17	Do operation as per (Logout DT) & check for response	— do —					

Step.	Description	Expected value	Actual value	Result	Inventory Priority Referrals
018	Try to enter text in the product name text box.	System should not allow the entry.			
019	Try to Select two radio button option.	System must not allow both option to be selected.			
020	Enter the values in the sales amount text box as per Sales Amount _ invalid table.	Inappropriate error message must be displayed.			
021	Try to edit / enter in the sales amount field.	System must not allow editing.			
022	Don't enter / click anything on commission button.	System must display error message.			
023	as 'Don't enter / select anything and click next button.'	The system must retain the previous input.			

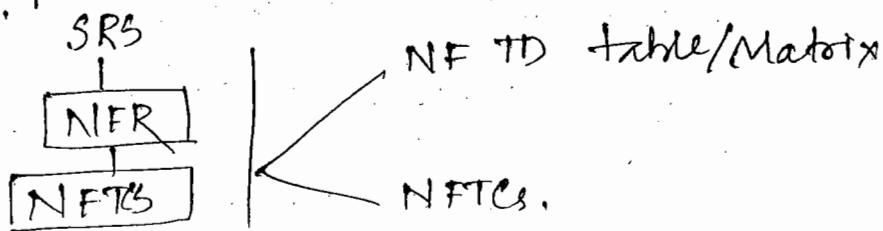
Commission Object table:-

SL NO.	Object name/ feature.
1.	Product ID (control Box)
2.	Product Name Text Box
3.	Local check Box
4.	International CheckBox
5.	Junior Radio button
6.	Senior Radio button
7.	Sales amount
8.	Commission Text box
9.	Commission Pass Button
10.	Reject Pass Button
11.	Back link
12.	Logout link

Case Study - 4:-

Date :- 04/03/2015

To prepare non-functional test data/Matrix tables and to prepare non-functional test cases, force sales express application.



In order to prepare non-functional test cases the test engineer must review non-functional requirements before performing testing on the non-functional feature like compatibility, usability, performance, configuration installation, interoperability, security etc.

Performance (load, performance, stress, volume, spike, Reliability)

Note - The scope of non function testing is the entire application.

Usability_table :-

Slno.	Item name	Response/Expected.
	Check the following items;	
(1).	Labels	must be available.
(2).	meaning of labels.	meaning of label must be understandable.
(3).	Spacing bet' the Label & object.	One line space.
(4).	Space bet' two labels.	One line Space
(5).	Color & contrast (combination)	must be with in the optimum label.

Sl.no	Item name.	Response/Expected
(6)	Icons.	must be present for frequently used command commands.
(7)	Check for Tool tip	must be displayed for the icons.
(8)	Visibility of important feature	must be kept in the highly visible area.
(9)	Date format	must be mention in specific format.
(10)	Error message	must be appropriate clear and precise.
(11)	Progress bar task bar/ Status bar Scrole bar	must be made available when required.
(12)	Help document	must be available.

(2) Performance:-

Performance can be tested for various factors as describle in the following table.

(a) Load_table:-

Sl.no.	Load/amount of load.	Response.
1.	50	✓
2.	100	✓
3.	150	✓
4.	200	✓
5.	250	✓

b) Performance_table:-

Sl.No	Task	Expected Resp. time.	Actual Resp. time.	Result-
1.	Task 1 (Login)	≤ 1.0 sec		
2.	Online Payment	≤ 5.0 sec		
3.	Commission Calculation.	≤ 3.5 sec.		

c) Stress_table:-

Sl.No.	Tasks along with users	Expected Response
①.	Login task with 10 users.	stable.
②	Login task with Online payment with 25 users.	stable.
③	Commission task with 40 users	stable.

d)

Volume_table:-

Sl.No.	Tasks along with volume to be performed for the app.	Expected Response.
1.	Check with 50,000 to 1,00,000 user	Sys must be able to handle.
2.	Check with 10,000 to 20,000 product	— do —
3.	Check with 1 to 2 lakhs trans.	— do —

(e) Spike-table:-

Scno.	Huge Load	Expected Response
1.	1000 user at a time.	Preferable Stable

(f) Reliability-table:-

Scno.	task with duration	Expected Response
1.	<u>Login task for 20 user</u> for <u>12 hr.</u> continuously	stable.
2.	<u>online payment task with 25 users</u> for <u>24 hr.</u>	stable.
3.	<u>Commission task with 50 users</u> for <u>48 hr.</u>	stable.
4.	<u>50 user Login + 25 user online payment + 10 user Commission enter combination for 72 hr.</u>	stable.

(3) Compatibility %:-

Compatibility table:- / Compatibility matrix table:- (CMT)

ScNo.	Compatibility Item.	Expected Response
1.	Browser.	
	1. IE.	Yes
	2. NN.	Yes.
	3. Google Chrome	Yes.
	4. Mozilla Firefox	Yes.
	5. Opera mini	Yes.
	6. Others.	Yes/No.

SNo.	Compatibility Item.	Expected Response.
2.	<u>Client OS:-</u>	
1.	win 7.	yes.
2.	win 8.	yes.
3.	win xp.	yes.
4.	win vista.	yes.
5.	Redhat Linux.	yes.
6.	Mac.	yes/no.
7.	other.	
3.	<u>Server OS.</u>	
1.	win 2003	yes.
2.	win 2008.	yes.
3.	Redhat Linux.	yes.
4.	others.	yes/no.
4.	<u>Application Server.</u>	
1.	TOM cat Apache.	yes.
2.	IBH webSphere.	yes.
3.	BEA Web Logic.	yes.
4.	IIS webserver.	yes no .
5.	others.	yes/no.
5.	<u>Database Server.</u>	
1.	Oracle Server (Oracle)	yes.
2.	SQL Server.	yes.
3.	DB2 (Microsoft) comes from IBM)	yes.
4.	Sybase (-)	yes.
5.	My SQL	yes.
6.	others.	yes/no.

(4) Configuration_table:-

Sl.No.	Item to be consider	Expected Response
1.	<u>Network topologies.</u>	
1.	Star.	yes.
2.	Ring.	yes
3.	Hub.	yes
4.	Bus.	yes
5.	Mesh.	yes
6.	other.	yes/No.
2.	<u>Printers</u>	
1.	Dot matrix	yes
2.	Inkjet	yes
3.	Laser	yes.
4.	Others	yes/No.

(5) Installation_table:-

Sl.no.	Item to be check	Expected Response
1.	Check for <u>Setup program</u> must be available.	
2.	Check for the easiness of installation. must be easy to instal	
3.	Check for the windows while instalat'n. must be with less complexity and easy to understand.	
4.	Check for the instalation flow must be in line/according to the req deployment document.	

SNo	Item to be check	Expected Response.
5.	Check for the disk space after installation.	Must be as per the Specification.
6.	Check for the configuration installation.	Must be configuration successfully.

6) Enter operability - table:-

SNo.	Item to be consider.	Expected Response.
1.	For online payment - check with master card.	Payment must be successful.
2.	Check with VISA Card.	— do —
3.	Check with American Express Card.	Payment must be successful.

7) Enter Security - table:-

SNo.	Security Item/feature to be tested.	Expected Response.
1.	Check for the authenticity of user.	Valid user must be allowed & invalid user must be block.
2.	Check for the access control.	Proxy Server must block the undesirable req & must allow desirable request.
3.	Check for the Encryption / Decryption.	Coding and decoding of the information must be successful and it can be broadcase end.

4. check for the protection of vital information in servers.

usually the fire wall must protect the information by blocking the request associated with viruses.

Test Case Table :-

Testcase no.	Testcase Description.	Expected value.	Ar. Res. S. P. SRS Ref.
TC-001	check for the usabili city of an application as per useability table.	Response as per table.	
TC-002	check for the load table of an app^n as per us load table.	— do —	
TC-003	check for performance of an app^n as per performance table.	— do —	
TC-004	check for stress of an app^n as per stress table.	— do —	
TC-005	check for volume of an app as per volume table.	— do —	
TC-006	check spike of the app as per spike table.	— do —	
TC-007	check Reliability of apps. as per Reliability table.	— do —	
TC-008	check Compatibility of apps as per Compatibility table.	— do —	

Test Case No	TC Description	Expected Value
C-009	Check configuration of the apps as per Configuration table.	— do —
C-010	Check installation of the apps as per Installation table.	— do —
C-011	Check interoperability of apps as per Interoperability table.	— do —
C-012	Check security of apps as per Security table.	— do —

Test Generation:-

Once the TCD is prepared before the test execution is started the following procedure must be accomplished.

1. Build Release process:-

(a) Once the developers create programs, the PM converted them into single copy of original source code and keep the same in the common repository, known as Code Base/TRUNK.

(b) PM prepares several builds and level them based on the purpose of to whom they are sent to, under the process known as Tagging. Several build are described below.

(i) QC-Build-1.0:-

- This build is send to QC
- This build contain source code.
- This build is meant for code walkthrough (It is a process in which program is tested against the coding standard whether they are implemented by the development team or not)

(ii) Dev-Build-1.0:-

- This build is send to another developer.
- It contain source code.
- This build is meant for code review (It is a process in which syntax and the logic is tested)

(iii) TE-Build-1.0:-

- This build is send to the test engineer.
- It contains functionality (exe format)
- This build is meant for validating the functionality

(iv) Prod-Build-1.0:- (Production Build)

- This build is send to the customer directly via production team.
- It contain functionality.
- It is meant for the business usage for the customer usages.

(C) Once the build are created the same is implemented to various roles.

(D) In response the QL will download the testing build & make it available to all the test engineers.

2. Build Deployment Process:-

→ Before even the build is released the test engineer must ensure the basic environment set up.

→ When the build is released the test engineer will review SRN - "Software release Note" (It is a document sent to the testing team along with build that contains known issues, some useful information and some time test data).

→ Test engineer will also review deployment document to understand the deployment process.

→ Test engineer will deploy the first build as per deployment document (in other words installation testing is automatically accomplished) (BVT).

→ Test engineer will perform build verification testing in order to check if the build is deployed successfully or not.

Test Execution:-

It is the process in which TCD is executed to carry out testing on application.

Process:-

1. Test engineer will consider each and every test case and respective test step to implement them on application.

2. Absorbed each and every response of the application while testing.

3. Each and every response must be documented in form of actual value under actual value column of TCD.

Levels of test execution:-

Test execution will be carried out in various levels as described below,

SLNo.	levels.	Type of testing	Purpose.
1.	0 level.	smoke testing	Availability check
2.	1 level	Detail functional testing	functionality check
3.	2nd level	Regression testing	to check for the old bugs as well as new bugs
4.	3rd level.	Final regression testing	to conform if the build is defect free.

Result Analysis:-

Date - 05.03.2015

It is the process in which test results are concluded in terms of pass or fail.

Process:- Consider each and every executed test case. focus on expected value and actual value and compare them.

3. Depends on matching or non-matching Conclude the result as pass or fail respectively.

Note:- Apart from pass or fail the following statement can be used as test results.

DTBD - (To be decided/ Done)

1. < TBD > (To be decided/To be done) :-

This notation is used as a result for the test cases that are associated with incomplete requirement/requirement yet to be decided.

2. < N/A > (Not Applicable) :-

This notation is used as a result for the test cases whose functionality are out of scope.

3. < Blocked > :-

This notation is used for the test cases associated with the functionality which are block/not available.

BUG TRACKING:-

Defn:- It is the process which information related to Bug is track from its origination till it gets closed.

(b) Type of Defects:-

Depends on whence exactly the defects are identified from, the following types of defects are identified.

1. Product Bug: (Present in the functionality of product)
2. Environmental Bug: (If the environment setup is not correct)
3. TCD Bug: (Due to wrong test case)
4. Documental Bug: (Due to incorrect info. present in document)
5. Test data Bug: (Due to incorrect test data)
6. SCD/Program Bug: (Mistake associated with program)

(c) Process:-

- (i) Identify the Bugs.
- (ii) Classifying the Bugs.
- (iii) Conduct retesting in order to reproduce BUG.
- (iv) Document the BUG with complete information.
- (v) Assign Severity and Priority for the BUG.
- (vi) Status - "New" or "Open" must be assigned to the newly identified ~~new~~ BUG.

(d) Types of Bug Tracking:-

Depends on how the bug tracking is done there are

two type:

(i) Manual Bug tracking:- Once the defects are identified by the tester they are documented in a template provided by the company, Known as DPD

Drawback:-

1. No transparency of defect information.
2. Duplication of defect information is possible.

(ii) Automation Bug tracking:-

Bugtracking process can be automated with the help of Bugtracking tools like Bugzilla, trackless, DR track etc, Pre-trackers, QC etc.

Advantage:-

1. Transparency of defect info. is provided.
2. Duplication of defect info. is reduced.

DPD -Template Study:-

Once the defects are raise they are documented in the DPD template that has the following fields in it.

1. Defect number.
2. Defect Description.
3. Step to reproduce.
4. Submitter.
5. Data Submission.
6. Module name.
7. Version number.
8. Build number.
9. Severity.
10. Priority.
11. Assigned to.
12. Status.

1. Defect number:- Every defect must be given unique number in order to locate uniquely, access easily and quickly. Ex: D-001, D-002.

2. Defect Description:- This field contains the description of defect in precise and perfect, clear manner. It should be concise.

3. Step to reproduce:-

This section contains the list of sequential guidelines to reach the defect.

Ex:- Valid Username →
Valid Password →

4. Submitter: -

This field contain the name of the test engineer.

5. Data Subitem: -

This field contain the when the defect is occurred.

6. Module Name: -

Name (and module) of the module must be mention here.

7. Version Number: -

A specific number of a version from which the defect is raised must be mention here.

Note: Version numbers indicate how many time the functionality is modify.

8. Build number: - Specific number of build in which the defect will found must be mention here.

Note:

Build number indicates number of time that the functionality released for testing.

9. Seriousity: -

Seriousity is an expression that describe the seriousness of defect.

Types of Seriousity: - In order to express various degree of Seriousness of defect the following type of Seriousity have been classified.

1. Fatal 3. Mission-critical.

2. Major 4. Minor.

(Broken link / fatal link)

1. Fatal :-

If the defect is associated with marginal block consisting of important feature and hard coding issue. It is considered to be fatal defect.

Hard coding:- It is a state of program in which undesirable constant value are present, that takes away the dynamic nature and always obey exhibits the static nature (constant op.)

Ex:- 404 Page not Found error.
500 Internal Server error.

2. Major :-

If the defects are associated with mal-functionality / wrong functionality such defects are known as Major defect.

Ex:- wrong op., incorrect display, wrong link
wrong calculation.

3. Minor :-

If the defects are associated with look & feel (GUI) issue it is considered to be minor defect.

Ex:- (i) Miss alignment-

(ii) inconsistency.

(iii) Wrong spelling

(iv) Background colors. (if the requirement is available)

4. Suggestion :-

In fact suggestion is not a defect but it is a note to a developer. in order to add value to the application.

- Ex:-
- (i) Correcting of error message.
 - (ii) Proposing user friendly message.
 - (iii) Correcting the background colour. (If the requirement is not there.)

10. Priority:-

* Triage Meeting:-

It is a special meeting conducted in order to resolved the issue between the developer and the test engineers. The participants of this meeting are.

Testing team, developing team along with PM, SQM and BA.

Priority:-

It is an expression that describe the sequence of the defects for the sake of rectification.

types of priority:-

There are various priority defined based on severity as describe below.

Sl.no	Seriousity	Priority.
1.	Fatal	Critical/ showstopper. (P ₁)
2.	Major	High. (P ₂)
3.	Minor	Medium. (P ₃)
4.	Suggestion	Low. (P ₄)

Q)

[S↑ + P↓]

Functionality is not available & that out of scope.

[S↓ + P↑]

If customer contention is more about

Q1. Describe the situation for the defect with High Severity & Low Priority?

Ans:- If the defect is associated with out of scope area functionality then it is consider to be low priority defect though its high severity.

Q2. Describe the defect with low Severity with high Priority?

Ans:- If the defect/suggestion is associated with user acceptance criteria and science it create a lot of impact on the customer though it is less severe it must be consider high priority defect/suggestion.

Q3. Who sets the Priority of a defect?

Ans:- Priority of defect is basically set by test engineer and if required it customize by the PM depends on situation.

11. Assign To :-

This column contain the list of developer entered against the corresponding defects under the process known as "defect assignment" usually done by PM.

12. Type of defect:-

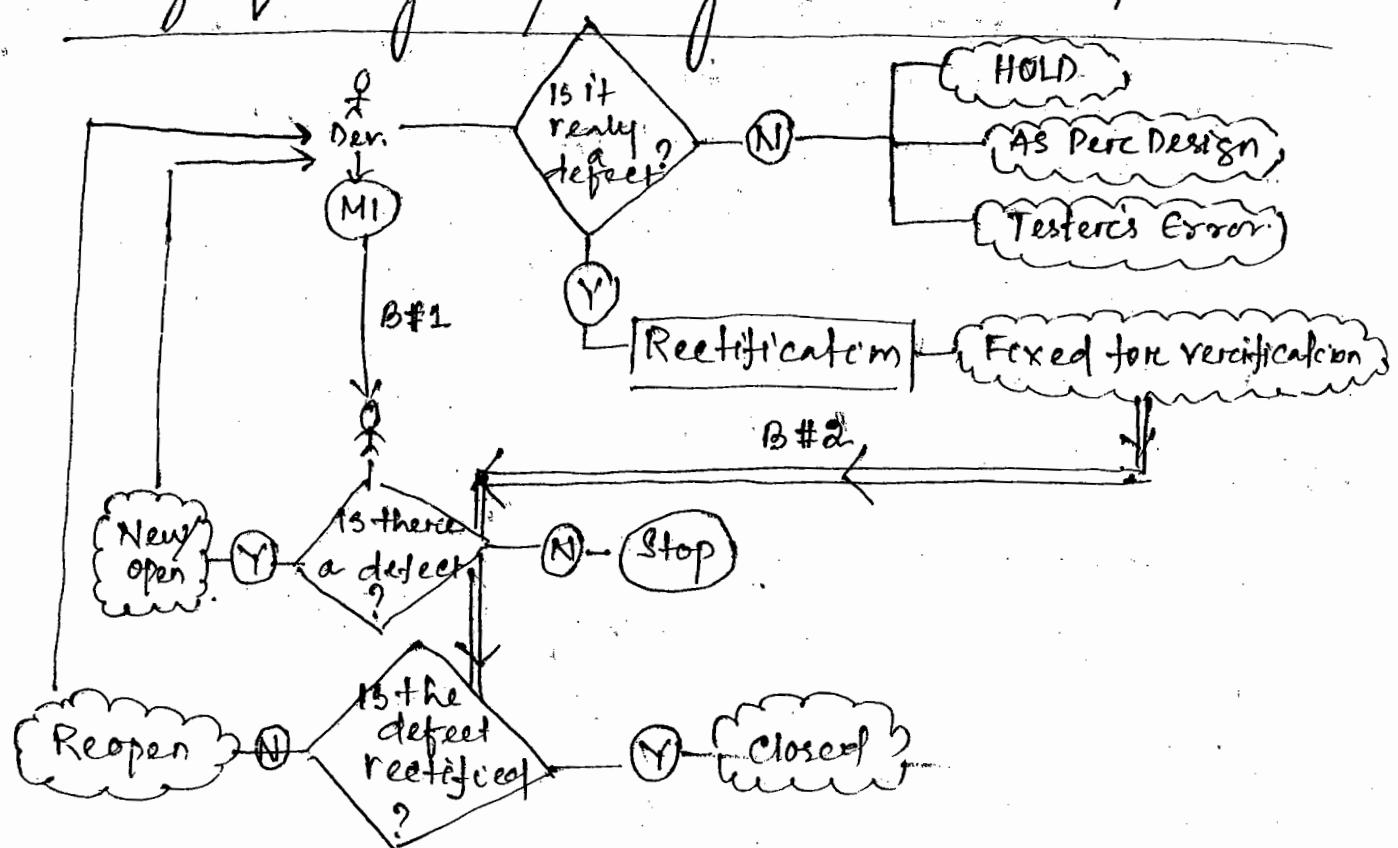
This field must contain the type of defects like GUI, functional, environmental, documental TCD, Test data etc. to indicate to what area the defect belongs to.

13. Status:- It is an expression that describes a state of a defect when it undergoes an event and crossing a specific milestone while travelling from tester to developer & vice-versa.

Note:-

Test engineer is officially responsible for raising the defect as well as closing the defect. One must remember that the developer cannot close the defect.

Buglife Cycle / Bug Resolution Process:-



From the above diagram the following status can be assign to the defects.

- (1) New/Open:- Soon after the defect is identified for the first time the test engineer assigns this status to the defect, indicating that it's a new born, fresh defect.
- (2) Fix for verification:- Once the defect is accepted and rectified by the developer he will assign this status to the defect, indicating that it is rectified and ready for regration testing.
- (3) Closed:- Once the test engineer checks the rectified defect and if it is found really rectified he will assign this status to defect, indicating that this defect is no more present in product.
- (4) Reopen:- On regration testing if the tester comes to know that the defect is not really rectified he will assign this status to the defect, indicating that this defect is still present in the product.

Some time the defect sent by the tester may not be accepted by the developer due to some reason. In this case the following status are assigned.

- (5) HOLD :- (Pending / deft / Postponed)

Whenever the defect is associated with incomplete requirement - it will not be accepted by developer and the same is assigned with a status.

"HOLD" - indicating that the defect is rectified only.

after sufficient information comes.

(6) AS PER DESIGN:-

When ever the modification happens with certain functionality on a emergency base, defects associated with these areas may not be accepted by the developer. The defects are assigned with the status indicating the implementation / modification is done based on the requirements of the customer.

(7) Tester Error:-

In case tester raises wrong defect due to wrong understanding and wrong design which is not accepted by developer and he will assign this status, indicating the tester has not tested properly.

Note:-

Reopen is not good for Developer;

Tester error is not at all good for test engineer.

Note:-

There may be extra internal status followed by company; some status are renamed with some other name.

Date: 06.03.2015

Bug Reporting:-

It is the process in which DPD is sent to the development team for the sake of defect rectification.

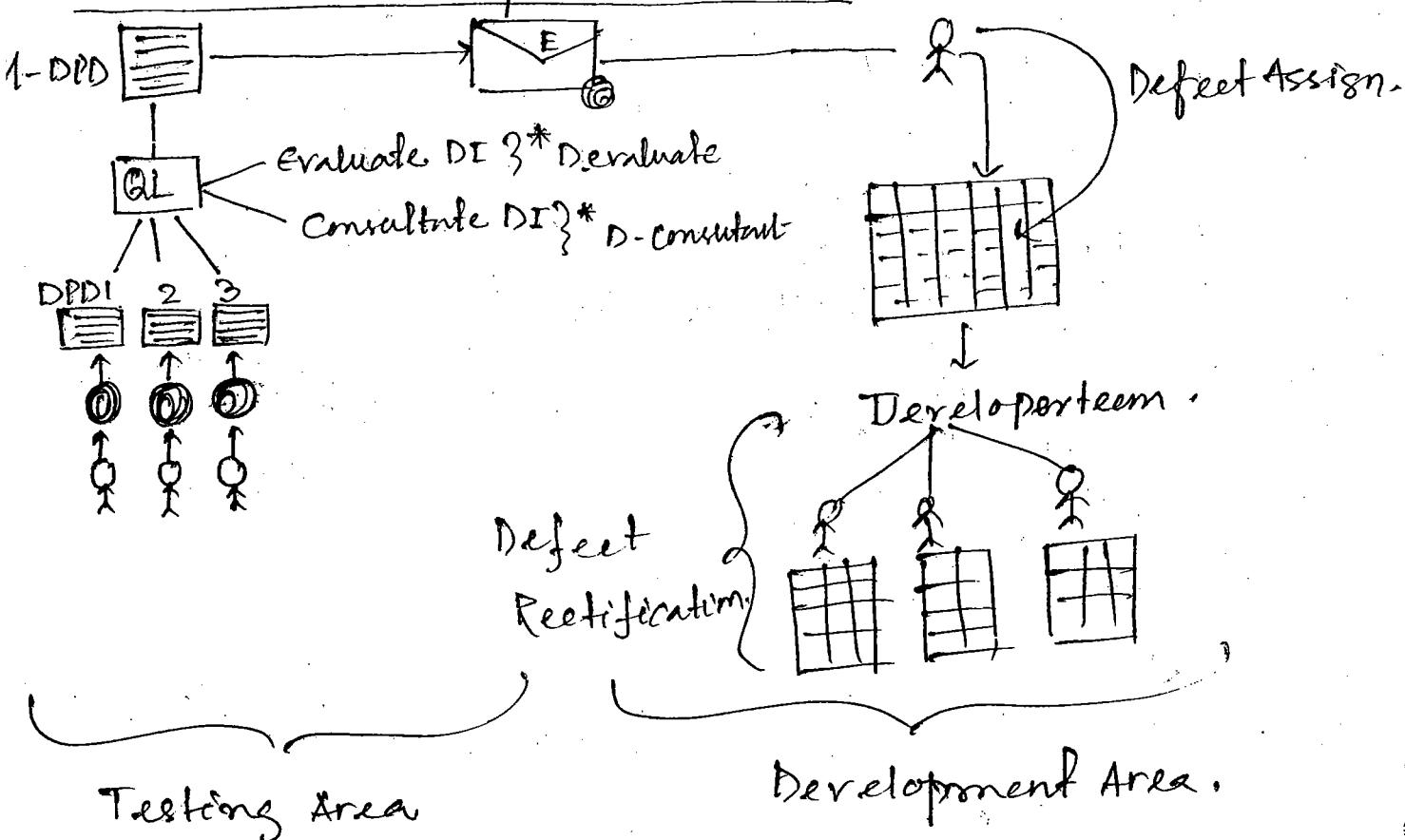
Types of Bug reporting process:- Depends on how the DPD are sent there are basically 3 processes that are followed by the industry.

1. Classical Bug Reporting Process.

2. Repository Base Bug Reporting Process.

3. Toll base Bug Reporting process.

1. Classical Bug Reporting Process:-



From the above diagram the following activity are done in the's process.

1. Tester perform testing on there individual functionality and raise/Prepare individual DPD that can be send to QL.

2. QL will evaluate the defect information and consolidate into a single document known as master DPD.

3. The Master DPD is sent to DM through an email as an attachment.

4. PM receives the DPD and performs "Defect Assignment" to the respective developers and share the same with development team.

5. The development team will start defect rectification making the functionality refined.

Drawbacks:-

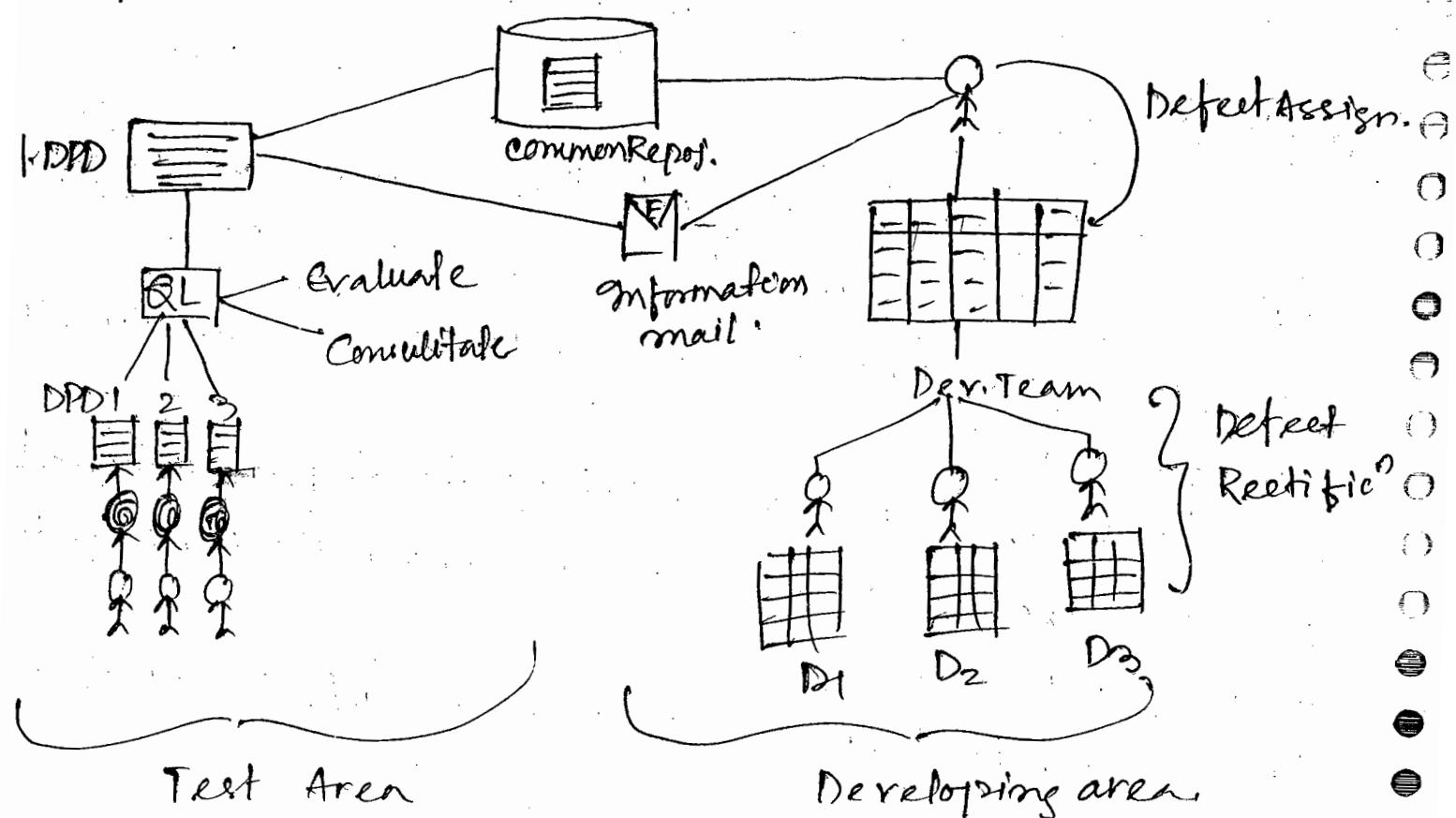
a. No security for defect information.

b. QL job is Tedium/difficult.

c. No provision for the development team to know the defect information while testing.

To address/mulifying the above drawback the new got is introduced known as Repository Based reporting Process.

2. Repository Base Bug Reporting process:-



Drawbacks:-

1. QL's job is still tedious.
2. No provision for the development team to know the defect information while testing.

Advantage:-

1. Defect information is made secure.

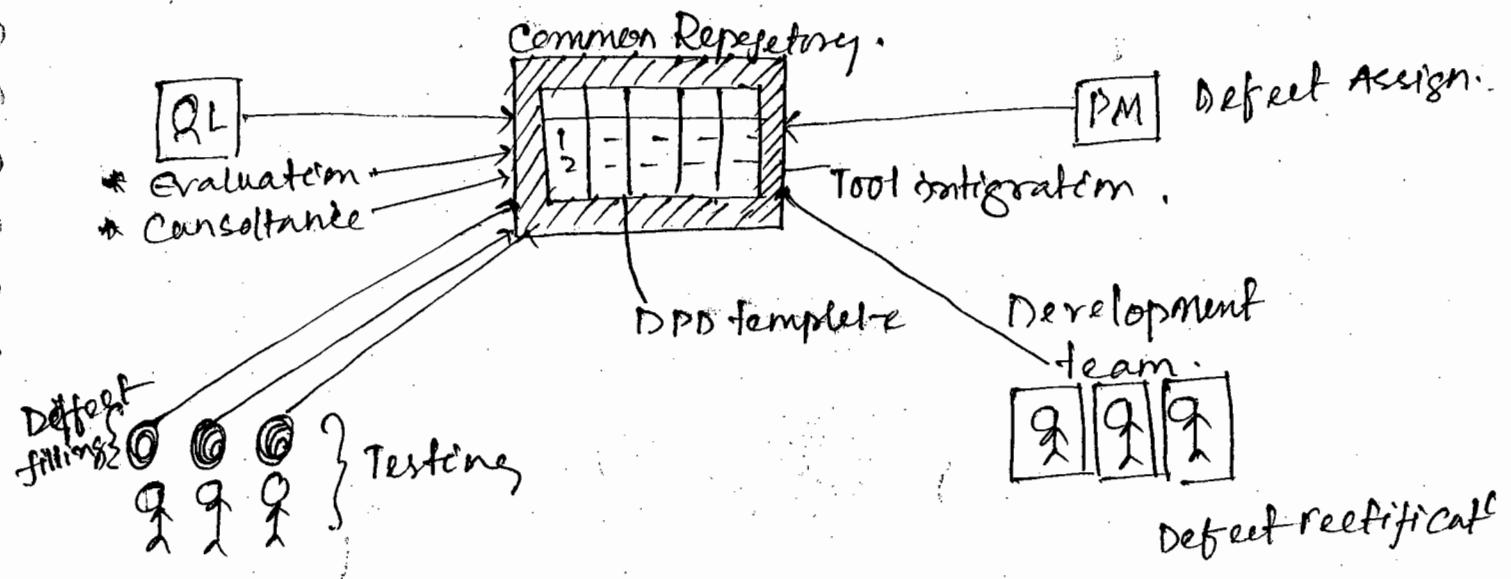
Process:-

1. Common repository is introduced in this process.
2. Individual DPD prepared & send to QL.
3. QL Evaluates & consults the defect info. to prepared Master DPD.
4. Master DPD is upload to common repository by QL.
5. The same is intimated to the PM through email.

POOL Phase

Development team start working towards defect rectification.

3. TOOL Base Bug Reporting Process:-



Process:-

1. In this process bug tracking tool is introduced that can be integrated with common repository.
2. Tools is capable of providing DPD template.
3. Defect information can be seen by all the roles at any point of time.
4. While testing happens defect filling (CTE), defect evaluation & Consultation (QL), Defect communication, defect assignment (PM) and defect rectification (Developer) are simultaneously done. Hence lot of time is saved.

[Note: Since the bug reporting process is done quickly it is considered to be the most efficient process.]

Advantages :-

1. Defect information is more secured.
2. QL Job is rather made easy.
3. Provision is there for the developer to see the defect information while testing goes on.

Test Reporting :-

It is the process in which test log and TRD (Test reporting Doc.) are sent to the high level management to let them know the status of testing, stability of functionality and the productivity of testing team.

Test Log:- It is the document in which information of passed test cases and failed test cases is given as described below.

Test case number.	Test case name.	Execution Status.	Comments.
TC_001	TC_SE_HP_Login_valid	Passed	✓
TC_002	TC_SE_HP_Login_invalid	Failed	To be reviewed To be execute again
TC_003	TC_SM_CM_comm_valid	On valid.	To be reviewed To be execute again
TC_004	TC_SM_CM_Reqt_invalid	Passed	✓
TC_005	TC_SM_CM_BL	<TBD>	Not execute
TC_006	TC_HP_SM_PL	<Blocked>	- do -

TRD :- (Test Report Document)

Depends on at ~~what~~ what level the TRD are prepared

there are two types of TRD as described below.

BTRD/NTRD:- (Build Test Report Doc./Normal Test Rep. Doc.)

It is the TRD which prepared at the end of each build testing.

Template of BTRD/NTRD:- (Content of BTRD/NTRD)

Every BTRD template has the following field in it.

Module Name:- _____
Version name:- _____
Build #:- _____

Tester's name:- _____
Tested on:- _____

1. Objective:-

2. Activities:-

3. Roles & Responsible:-

4. Slippage:-

5. Total defect & Defect Matrix:-

6. Test case Passed/Failed:-

7. Defect Detection Rate:-

8. Build Test Summary:-

FTRD (Final Test Report Doc.) :-

This is a type of TRD that can be prepared after the final build testing is over.

FTRD contain the following.

1. All individual BTRD summary.
2. Final Build Summary report.
3. All the fields that are BTRD are present.
4. Apart from the above fields it has the recommendation note for release.

Note:- Depends on the defects present in the final build the release code can be +ve or -ve.

→ * Defect retriification process.

Regression testing:-

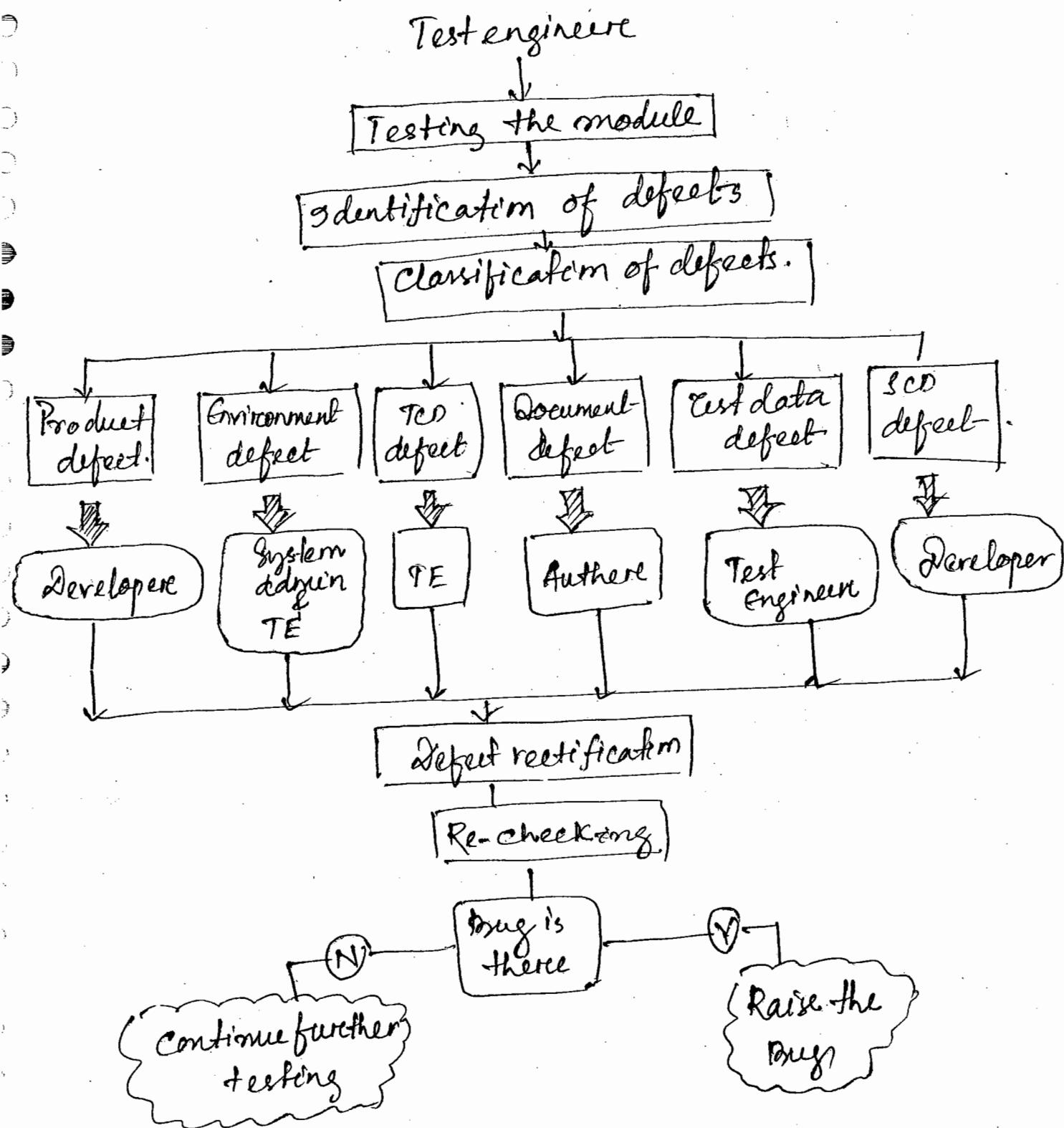
Process of regression testing:-

* ~~Defect retriification process.~~

1. To identify smoke/ sanity test cases.
2. To identify Failed test cases.
3. To identify Dependent test cases.
4. To identify user Acceptance criteria related test case.
5. To identify Business components related test case.
6. To execute smoke/ sanity test cases, (To check the availability of object/feature).
7. To execute failed test cases (To check old defects are really rectified)
8. To execute dependent test case (To check if there are any new defects due to the rectification old defects)
9. To execute user acceptance criteria related test case (To check if the user acceptance criteria is not effected).

10. To execute business component related test cases
(to check if the key functional area from the front of business are okay)

Defect Rectification Process:-



7. User Acceptance testing:-

Before user acceptance testing goes on test closure activity will take place as described below.

Test closure:-

Test closure is the state/process where in/in which all the testing related activities are officially put to an end based on exit criteria, test coverage and defect density analysis.

Test closure usually done base on the following criteria:-

(1) Exit criteria:- Testing can be stopped based on the following exit criteria.

(a) When all the functionality are cover under testing.

(b) All the fatal defects/bugs are rectified.

(c) When the number of defects must be within the threshold/critical value.

(d) When the risk of the product is within the tolerable limits.

(e) When the total budget of testing is ~~exceeded~~ exhausted.

(f) When the dead line is approach.

(g) Test coverage Analysis:-

In this process testing team and the test management will ensure test design RTM & test execution RTM are done. Based on that they can come to a conclusion that the test coverage is complete.

3. Defect density analysis:- This is a process in which defect density (number of defects per unit/module/submodule functionality) is determined in terms of percentage module wise to decide whether the module can be further tested or not.

Sl.no.	Module name	Defect density.	Comments.
1	Login.	84.	OK
2	Commission.	124.	OK
3	online payment	104.	OK
4	Services.	347.	No OK to be regression testing

from the above table Service module have more defect density and so it has to undergo further regration testing.

4. Deferred Bug Analysis:- It's the process in which the entered project team focuses on deferred bugs to check if they can really Postpond. In case any defect can be rectified (if it is possible), they can be rectified even at this stage.

4. Test review meeting:-

It is a final meeting conducted regarding testing on which testing activities are terminated based on exit criteria reports, test coverage, Bug density, deferred bug analysis report.

User Acceptance Testing:-

Process of VAT:-

- a) VAT Plan must be prepared. (noting but VAT ~~is~~ test case document)
- b) Based on the implicit requirement user acceptance criteria is created based on which the VAT test cases are prepared.
- c) Most of the time VAT is carried out on the application from the point of implicit requirements to ensure customer satisfaction.
- d) Basically the customer must be encouraged to carry out VAT so all the favorable conditions along with the required input must be provided to the customer.
- e. Testing team can do this whenever based on the request of the customer.
- f. It can be conducted in the development company itself before delivery in terms of alpha testing.
- g. It can ^{also} ~~not~~ be conducted in the customer place in term of beta testing.
- h. Once the VAT is done, VAT Report can be prepared
- i. Ensure the entered user acceptance criteria is signed off (It means all the test cases must be pass)

(10) Test sign off:-

It is the last phase of STLC which with which the total lifecycle is comes to end. This space is associated with following activities.

1. Ensure all the delivery doc. are ready and available.
2. Archiving test ware, test environment and test infrastructure (preferable build wise)
3. Move the above information to the control system.
4. Lesson learn and best practices must be documented.
5. Focus on failed test cases and the corresponding defects along with the severity.
6. Focus on open/reopen defects and closed some of the defects if they can be really closed.
7. Ensure user acceptance criteria is sign off.
8. Testing team is Role off by the QL.
9. Some tester are selected to be a part of onsite team. Some are assigned to another on going project; Some are kept at bench.
10. Finally QL is Role off by the S&M.

Dt. 09.03.2015

Characteristic Features of Test Engineers:-

- ① Test engineer must have quality oriented mind setup.
- ② Test engineer must have Test to Break attitude
- ③ Every test engineer must be tactful & diplomatic in nature. (How u convince developer for defect?)
- ④ Every test engineer must have good communication skill and good drafting skill.
- ⑤ Every test engineer must have internal structural knowledge.
- ⑥ Test engineer must have / be intuitive and creative.
- ⑦ Every test engineer must have judgement skill.
- ⑧ ~~Concepts~~ like territority/Priority Judgement as well as judgement call.

* Judgement call:- It's a attitude of being initiator to continue the task despite use for the benfit of the organization.

Taxonomy / Terminology

- 1. Defect product.
- 2. Defective product.
- 3. Quality Assurance.
- 4. Quality control.
- 5. Inspection.
- 6. Audit.
- 7. Process violation.
- 8. NCR.
- 9. SCR.
- 10. CAPA.
- 11. MRM.
- 12. PPM.
- 13. PPR.
- 14. Metrics.
- 15. Matrix.
- 16. Escalation. (Issue/slippage sent ^{to higher level})
- 17. SRN (Dev → ^{PM} Test)
- 18. SDN.
- 19. Prototype.
- 20. Benchmark.
- 21. Slippage.
- 22. Indemnity call.
- 23. Review. (Check/study) detail process.
- 24. Review Report.
- 25. Peer Review. (contain content)
- 26. Walk-through.
- 27. Code-walkthrough.
- 28. Code-Review.
- 29. Version.
- 30. Impact Analysis. Build.
- 31. Change Request.
- 32. Impact Analysis.
- 33. Workaround (Alternet solution).
- 34. Hard coding.
- 35. Baseline (freezing process).
- 36. Publish.
- 37. Share point / RSS / cus.
- 38. Templates.
- 39. Deployment Documents.
- 40. Kick off meeting.
- 41. Demo.
- 42. Check-in.
- 43. Check-out.
- 44. Patch.
- 45. Race condition. (concurrency test).
- 46. Memory leak. (endurance test).
- 47. Test bed.
- 48. Test harness.
- 49. API.
- 50. ABI (App. Binary Interface).
- 51. BRT. (Build verification Test) (Litmus testing).

• Defect products:- If the product is associated with non-conformance and if it is functionally okay such products are known as defect products.

* Note:- customers may used this products as the severity of defect is less if the products are functionally okay.

2. Defective products:- If the products are associated with non-conformance and if they are functionally not okay such products are known as defective products.

Note:- Defective products are not used by the customers as their functionality is effected due to the high severity of defect.

• Process Violation:- It is an act of deviating from the guidelines.

• NCR:- (Non conformance Raised)

If any role is found violate in the process a defect is raised on him known as NCR.

• SCM:- Software Configuration Management
(or) Requirement change Management.

• CAPA:- (Corrective action & Preventive action)

Corrective action:- It is the process in which repairable mistakes are corrective.

Preventive action:- It is the process through which whenever the irreparable mistakes are encountered, the note

repeat mistake in future. In other words in this process preventive methods are implemented.

11. MRM (Management Representative Meeting)

It is basically conducted to discuss the status of the company.

1. Usually CEO will address the meeting.
2. Overview status of the company is discussed.
3. Growth rate in terms of finance, technology, human resource is discussed.
4. Projects that are in pipeline are discussed.
5. Internal audit reports are discussed to know the productivity.
6. Customers appreciation and comments are discussed.
7. HR and technical issues are discussed.
8. Individual achievement or acknowledged.

12. PPH (Periodic Project Meeting)

It is the meeting conducted periodically (ideally once in a week) to discuss the status of the project which include the followings.

1. Percentage of completion / in completion.
2. Roles & Responsibilities involved.
3. Slippages - task slippage / ~~schedule~~ ^{schedule} slippage / effort slippage.
4. Total number of defect and defect matrix.
5. Test log information.
6. HR & Technical issues.

2. individual achievements and technologies.

Note:- Hoo will be addressing the meeting while the entire testing team is present.

13. PPR :- (Periodic Project Report) :-

It is the document prepared by 'QL' and it send to "HOO" for the review before the "PPM" is conducted.

21. Slippage:- Deviation from planning

- - Schedule slippage } Apart from these.
 - Task slippage }
 - Effort slippage.
 - Defect slippage (Hidden / Retained defect)
 - Test case slippage.

(Defect suppose to be identified or not identified)

(Test cases that are suppose to be written or not written).

4. Review Report:- It is the outcome of review.

a. Code Optimization:-

It is the process in which number of line and complexity of code is reduced in order to increasing the performance of code. Hence it is also known as fine tuning.

b. Change Request:- It is the requirement proposed by the customer when the project is on going.

33. Impact Analysis:-

It is the process in which impact of the newly proposed requirement (if it is implemented right away) is determined. If the impact is more the implementation is deferred. If the impact is less the CR can be implemented.

36. Baseline:- It is a process in which the information in the documentation is finalised / frozen.

(cont)

(Anticipation of conformation of information)

37. Publish:- It is the process in which the baseline documents are officially used with official certification.

43. CheckIn:- It is the process in which authors will upload the document to the Common Repository.

44. Checkout:- It is the process in which the authors will download the original document for the sake of modification.

[Note:- usually when the document is checked out it will not be used by others until it is checked in.]

49. Test Harness:- It is the setup test cases are setup test script are any two that can be derive testing on to the application under test (AUT).

50. API:- (Application Programming Interface)

It is an interface between an application and the 3rd party service, acts like a mediator so that the application can successfully use of this service.

51. ABI:- (Application Binary Interface):-

It is an interface acts like mediator between the two environments when ever the application is transported in terms of binary format.

45. Patch:-

It is the process in which unavailable, unusable and不稳定 build is rectified into available, useable and testable build in short span of time with minimum efforts.

Note:- When the build is released again with Patch it is considered to be patch build but not build number 2. (as build no 1 is not counted due to the fact that it is a fatal build).

Date - 10.03.2015

Quality Standards :-

Quality standards are basically guideline that are defined for the organization to make it productive so as to deliver quality products.

Type of Quality Standards:-

There are various type of quality standards that are available in the market out of which most preferred standards are:

ISO, CMM and 6-sigma.

ISO:- International Organization of Standardization.

Type of ISO:- 9001, 9002, 9003,

Ex:- ISO: 9001:2008

Type of ISO

year of implementation.

Types of ISO:-

1) ISO: 9000 → This type defines guidelines to guide newly startup company or unmonitored company to help them to carry out the work in a proper manner.

2) ISO: 9001 → This type of ISO defines guidelines for the companies where in all the activities are done properly as per guidelines.

3) ISO: 9002 → This type of ISO defines the guidelines for the organization with all the activities that ~~are~~ ISO 9001 accept plan.

4) ISO: 9003 → This type of ISO defines the guidelines for the organization exclusively for testing activities.

5) ISO: 9004 → This type of ISO defines the guidelines for the organization specially for continual improvement and ^{research} resources and development activities.

CMM :- (Capability Maturity Model)

It is another type of Quality Standard which help the organization in focusing on microthings to macrothings so that ultimately Quality is produced.

Level of CMM:-

→ Capital level) :-

1) CMM:1 → This level defines guidelines in such a way that they can follow their own guidelines without with the strong team. Such organization can be assert as CMM1 company.

2) CMM:2 → (Repeatable level) :-

This level defines the guidelines - which includes
1. every task is well defined 2. key process must be repeatable
In other words it tells us to do smart work rather than hard work.

3) CMM:3 → (Defined level)

This level defines guidelines -

- 1. objective avoidance. (Every task must be defined with an objective/goal)
- 2. All must follow guideline

(discipline is implemented)

- 3. All thing must be documented (documentation is implemented)

4) CMM:4 → (Managed level)

This level defines guidelines and apart from them it instructs the organisation to follow matrices such Company will fall into CMM:4

5) CMM:5 → (Optimized level)

This level defines the guideline to go guide R&D and continual improvement apart from the rest of the activities.

CMM_i → Hence i stands for integration; integration of software & hardware, in other words hardware factor is considered to be the key factor when ever quality software is to be produced.

PCMM (or) CMM_P :- This level will primarily focus on the factor people. It believes that if the employees are happy the productivity are more and the final product will be qualitative. So the employees of these company are provided with more benefits to keep them ~~long~~ happy.

Six-Sigma:-

It is another quality standard that can be followed by the industry to get the best quality.

Nature of Six-Sigma:-

1. Six Sigma is implemented with multiple cycle of production.
2. Six Sigma implementation is associated with creation of multiple graphs that are basically produced with two key factors - average and standard deviation calculated from each production cycle.

Implementation of Six-Sigma:- DMAIV - 6-Sigma Cycle.

6-Sigma cycle :- It can be implemented with the following milestones

1. Define:- In this space targets / goals / expected results are define based on requirement. Once the target are set one production cycle is open to get the product.

2. Measure:- Once the production is done it is tested for the requirements. In this process there is every possibility that the deviations are identified.

3. Analyse:- This page is meant for analysing the root causes of deviation

4. Improve:- Based on the analysis the existing process can be modified/refined. Once the process is improved another production cycle is run.

5. Verify:- Once the production is done with the improved process the product is tested for the following things.

- a) to check for the requirement in the product as usual
- b) to check if there is any effect of improvement of process.

Note:- If it is effective then continue the cycle for the rest of the production cycle. If not DMAIC process is implemented where C stands for change (off process)

• Why 6-sigma?

When multiple cycle production are happens, at time average and standard deviation are calculated and a graph is initially developed with these values. As the system approaches requirement there is a transformation in the graph in such a way it adds numerical values on the graph starting from

each other. moving toward) the production as well as testing can be stopped whenever the graph is absorbed that it is ~~Space or Correlation~~. Spanned over the x axis by 6-unit, at this stage if the quality of product measure it will be 99.73% which is equivalent to 100%. Hence 6 sigma graph is taken as a benchmark and is standardised for the required quality.

3.4 DPMO

$\frac{\text{defect per million opportunity}}{\text{Test cases Tc}}$

Software development equivalent expression for quality is 3.4 defect per million test cases. In other word 1 million test cases are written then one can able to find 3.4 defects. Practically it is zero defects.

3.4 DPMO ≈ 0

Agile Process:-

Apart from the formal SDLC process models there is another process model that is used by many companies today is agile process.

In other words it is the process in which solution can be delivered to the customer quickly in terms of multiple instalments giving a room for customers to send the CRs. at any stage of development.

Need of Agile process:-

To understand the need of "agile process" first of all we need to look for the challenges of the

Industries as described below.

1. Unclear requirements:-

Customers may not be able to express the requirement properly.

2. Frequent change of requirement:-

From time to time depends on the condⁿ of the business the requirements are tend to change.

3. Environmental change:-

Since multiple organization are collaborating with each other there is every possibility that the softⁿ may be developed / upgraded which are adaptable for frequently change environment.

4. Quick solution:-

Since the competition is prevailed (goingon) in the market the customer may be need of the solution quickly.

5. Technology/Domain change:-

There is every possibility that the technology / domain may be changed from time to time hence there is a need for us to developed compatible softⁿ to sustain in the market.

Note: In view of above challenges in the market there is a need for us to customized the formal process and to defined easy going procedure, known as "Agile process".

Agile Process Manifesto/Principle :-

1. People and interaction are more important than the process and the tools/technology.
2. Working system (solution) is more important than documentation.
3. Customer collaboration is ~~important~~ rather than contract negotiation.

Agile Methodologies :-

Agile process can be implemented with various methodologies as given below.

1. Scrum methodology.
2. Extreme program methodology (XP)
3. Dynamic system development methodology
4. Adaptive development methodology.

But all the above scrum gain popularity due to practical advantage and so it is integrated with agile process.

Characteristic feature of Agile Scrum Methodology:-

1. Business Advantages, we have in short span of time.
2. Rapid and repeated development & testing of the working system.
3. This methodology is incremental as well as interactive.
4. The scrum team is self managed and self organised.
5. Working system is important rather than have documentation.
6. Resources availability and their capacity are considered before they get into the actual work world.

7. The entire solution is divided into intermediate solution and are delivered in instalment basis.

8. Close monitoring of the productivity as well as quality is done, almost in daily basis.

9. Agile scrum methodology focuses on 3 key area-

a. People/ Roles.

b. Documents/ Artifacts.

c. Interaction/ Meetings/ Ceremonies.

Roles :- The following roles are available when agile scrum process is followed

1. Product Owner:-

a. Product owner can be basically a customer, Key stake holder (Partner) or It can be business analysis.

b. Product owner is the actual user of the product.

c. Product owner is responsible for defining the features for the product.

d. Product owner is responsible for prioritizing the features of the product.

e. Product owner must be always available for the scrum team and clarifies all the queries of Scrum team.

f. Product owner is responsible for defining dead lines for the products/ Sprint.

g. Product owner is responsible for gathering requirement from customers.

h. Product owner defines acceptance criteria for user stories.

2. Scrum Masters:-

a. The scrum master is none other than the project manager who acts as a facilitator.

b. Scrum master maintains and manages the entire scrum team.

c. Scrum master has the responsibilities to monitor the productivity and status of the sprint development on the daily basis.

d. Scrum master will follow up the impediments (issues) and try to provide solution.

e. Scrum master shield/safeguard the scrum team from external interferences.

f. Scrum master will ensure that scrum value and practices are maintained effectively.

3. Scrum Team:-

a. Scrum team includes developers, test engineers, DBA, Business executives apart from PO & SM.

b. Scrum team may contain 6 to 8 members.

c. Scrum team is responsible for design, development and testing of each and every sprint for the product.

d. They maintain proper co-operation & co-ordination to finish the task in a specified intervals of time.



Agile Process :- Agile process is implemented with the following mile stones in terms of conducting several meetings and creating various documents as described below

(A) Preproject/Kickoff meeting :-

1. This meeting is conducted just before sprint planning meetings.

2. Participants of the meeting are basically key stakeholders, management of development company CHRO, and tentively assign project members.

3. This meeting is mainly conducted to select Product owner and to conform Scrum master and Scrum team.

4. Once Product owner (PO) is selected he starts interacting with the customer and the end user of the product to gather the requirements.

5. Requirements are gathered in terms of epics and user stories.

6. Once the requirements are gathered they can be maintained in the document known as Product backlog.

7. The Scrum team members will do the home work on the information related to the user stories that is to be discussed under the sprint plan meeting.

Epic :- Epic is nothing but long user story or undefined user story. The Scrum team can start with epic and eventually break it down into small user stories.

User Story :- User story is nothing but meaning full requirement.

1. User stories are kept and maintain not in terms of huge documentation but in terms of simple paragraph.

2. Every user story must be simple, realistic (feasible), estimatable, negotiable, complete and testable.

3. Example:- As an admin I want ~~the~~ login activity to be locked by the system whenever the user ~~try~~ ~~went~~ to attempted login action 3 time with incorrect ~~or~~ credential.

4. Every user story must be associated with the corresponding user acceptance criteria.

5. PO is basically responsible for acceptance criteria however any Scrum team members is allowed to define user acceptance criteria.

6. Testing team can be create test scenarios & test cases based on user stories and the corresponding acceptance criteria.

Product Backlog :-

1. It is basically a document that is created as an outcome of pre-project kick off meeting.

2. This document contain all the list of user stories.

3. Every user story will be assign in a appropriate priority, depends on its importance. Some of the priority are small, medium, large, extra large.

Product Backlog

US ₁	Barn S
US ₂	Commission M
US ₃	Donofe making L
US ₄	Online Payment XL

(B) Sprint Planning Meeting:-

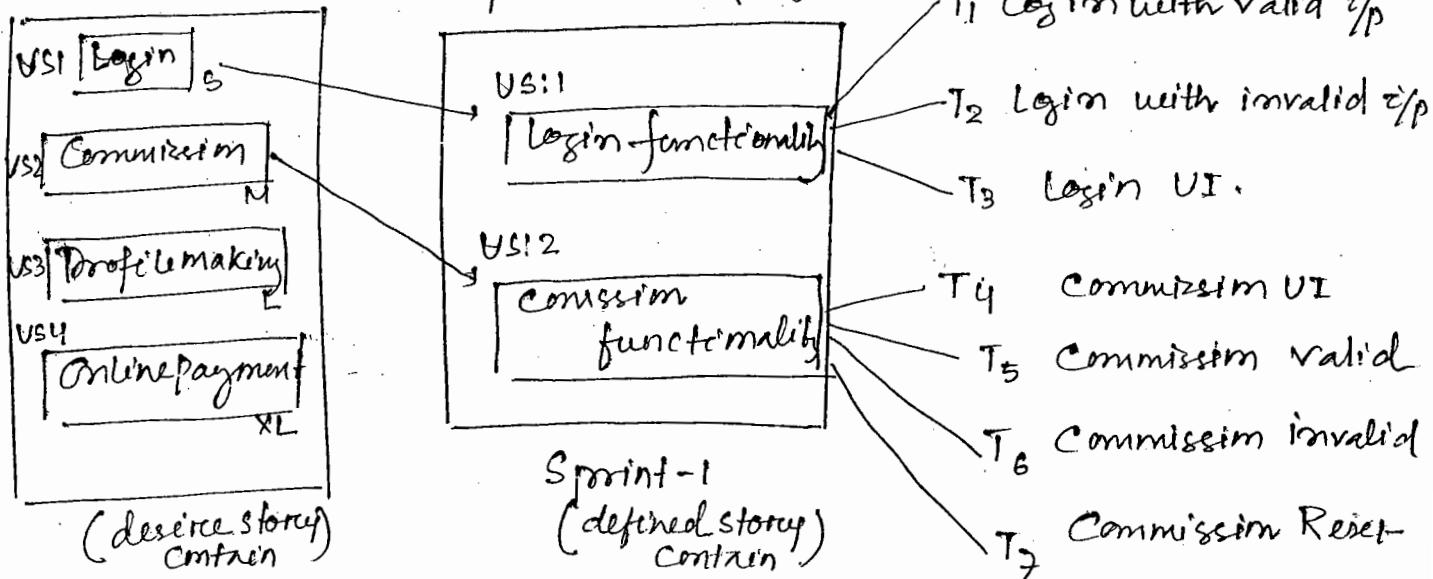
1. This meeting is conducted after pre-project kick off meeting as a starting mile stone for the sprint.
2. The participants of the meeting are - Product owner (PO), Scrum team (ST), Scrum Master (SM)
3. Product owner takes one user story out of product back log and put before the scrum team for the sake of brain storming (discussions).
4. The entire scrum team will be involved in the brain storming process to discuss about all aspects of user stories.
5. Scrum team will be involved in sizing the user story based on the complexity. In other words they will define story points. For the user stories
6. Scrum team can identify the number of tasks associated with each user stories.
7. Scrum team is also responsible for estimating the tasks in days or hours.

8. Scrum master is responsible for conducting this meeting.

9. Once the desired user stories are selected for the sake of implementation on the scrum sprint, all these are kept and maintained in a document known as sprint backlog.

10. Sprint planning meeting will last for 8 hours.

Product backlog Sprint Backlog



Story Points:-

1. Story point is a digital expression of a complexity of user stories.

2. The effort and estimation of the task are basically designed based on the story points.

3. Ensure the user stories with small story points. To ensure perfection in estimation.

4. Usually the story points are associated with Fibonacci Series like (1, 2, 3, 5, 8, 13, 21,).

1,2,3 → Small Complexity.

5,8 → Medium. "

13,21 → Large. "

5. To size these user stories lot of brain storming / discussion is required.

6. When ever the sizing is to be done for user stories the following criteria must be consider

1. Complexity of user story.
2. Skill to implement that user story.
3. Dependency of the user story with other user stories.
4. Acceptance criteria.

Estimation:-

It is a process in which number of hours required for each and every task of the entire sprint is defined.

Sprint Backlog:-

1. This is a document created as an outcome of Sprint Planning meeting.

2. This document contain all the selected defined user stories for the implementation of sprint.

3. Every sprint backlog can contain the total number of task not more than 250 to 300.

4. Estimation for each task should not go beyond 12 to 15 hours. If any task is found more then this it must be split in to two task.

5. The information present in the sprint backlog can be modify by the scrum team when the sprint is being implemented.

~~Note:~~ product owner does not have any right to modifies this information when the sprint is being implemented. He can modify the information either before the sprint is started or after the sprint is done.

(c) - Sprint Implementation :-

1. Once the Sprint backlog is freezed the sprint implementation will starts.
2. Development team is involved in preparing the design and the corresponding source code.
3. Test engineers can perform validation on the implemented user stories.
4. The productivity of the team, status of the Sprint, Quality of the Sprint is monitored on daily basis by conducting daily Scrum meetings.

So the ~~schedule~~ status of the Sprint is very well monitored with the help of document known as "Burn down chart".

5. Once the sprint is completed it will attained the status "done".

Daily Scrum Meeting:-

1. Main objective of this meeting is to know the status of Sprint on daily basis.
2. Usually the meeting should not last for more than 15 min.
3. Usually all the member will standing as a group in circle to discuss the status and so it also known as stand up meeting.
4. The main agenda of this meeting is to address the following question by every Scrum team member?
 - Q1) What did u do yesterday?
 - Q2) What are you planning to do today?
 - Q3) Are there any impediments?

5. Scrum master takes the responsibilities to follow up these issues and provides the solⁿ to the team.

Burn Down Chart:-

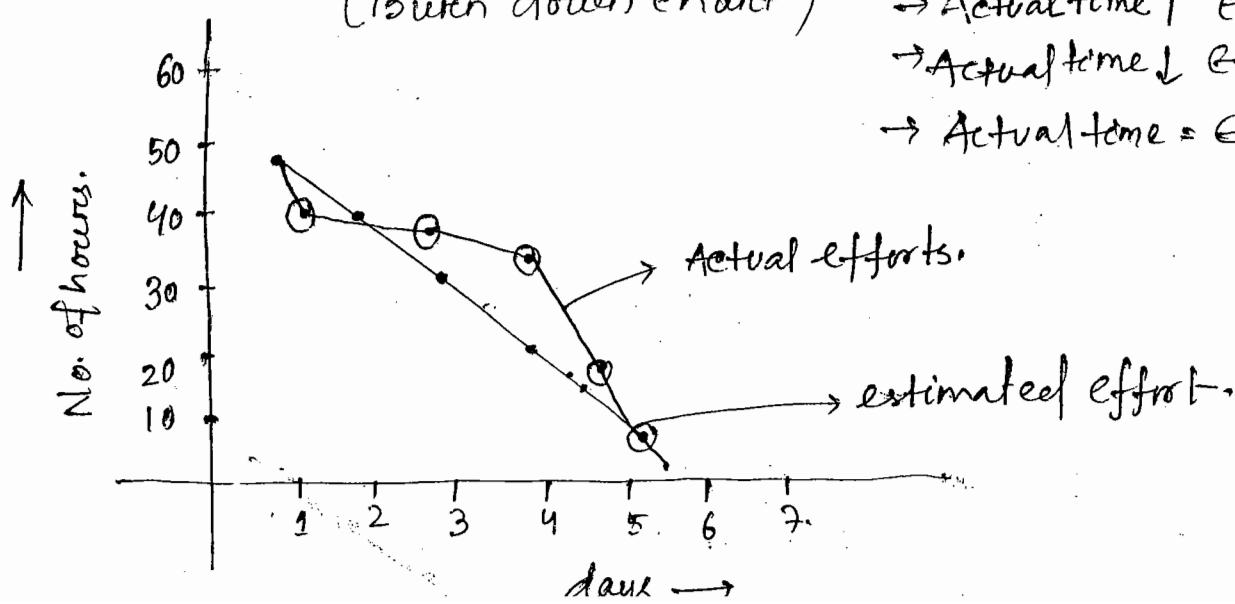
1. The main purpose of burn down chart is to know the status of the efforts on daily basis.

2. This document describes estimated efforts and actual efforts. 3. The following is an example of burn down chart to prepare for 7 days sprint for 3 user stories and 6 tasks and also with 4 resources.

User Stories	Task	Effort	D ₁	D ₂	D ₃	D ₄	D ₅	D ₆	D ₇
US1	T ₁	3	2						
US1	T ₂	5	3						
US1	T ₃	8	2						
US2	T ₄	10	4						
US2	T ₅	8	3						
US3	T ₆	8	2						
Estimated →		42	36	30	24	18	12	6	0
Actual →		40	16						

(Burn down chart)

- Actual time ↑ Estimate
- Actual time ↓ Estimate
- Actual time = Estimate



4. Burn down chart is consider to be a deliverable during daily meets.

5. Burn down chart will enable to make a graph that describes the progress of the estimated and actual efforts.

6. The actual effort graph will not be a straight line and it have ups and down in between & eventually reaching zero as shown in figure.

Done:-

Velocity :- It is a total number of stories point achieved by the team.

Note:- velocity is so important that based on which the productivity of team can be estimated.

→ Done:- If the Sprint is completely developed, thoroughly teste and ready for shipment, the state of Sprint is known as done.

D. Sprint Review Meeting:-

1. Once the Sprint is done the Sprint review meeting is conducted to display the sprint before the customers.

2. Participants of this meeting are PO, Stakeholders, Customer, management, Scrum master and Scrum team.

3. "PO" will check the Sprint for all the user stories whether they are properly implemented as per the acceptance criteria.

4. During this meeting finished user stories and unfinished user stories are identified.

5. Sprint review meeting will last for 2 to 3 hr.

6. During this meeting customers feed back, feelings

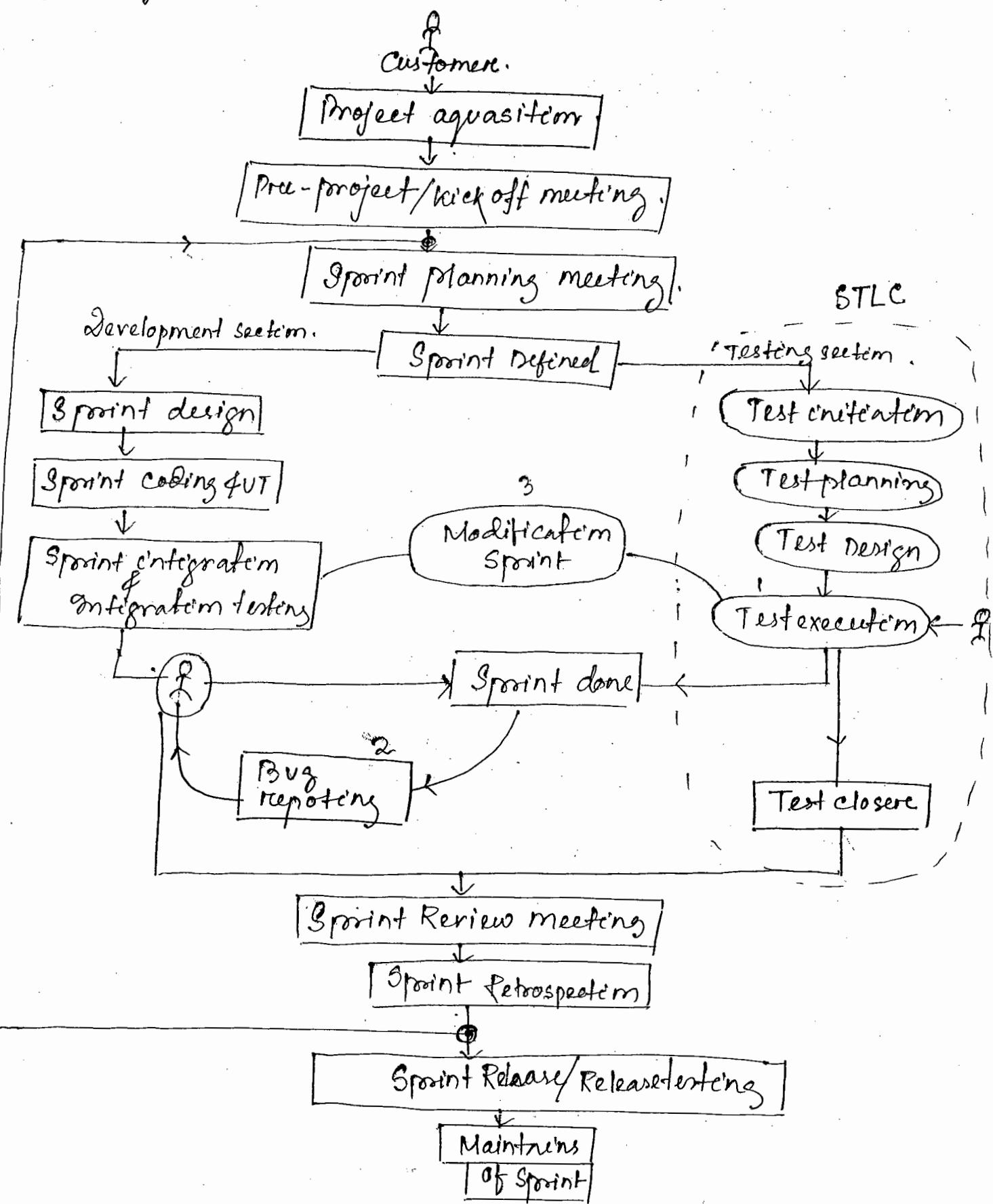
E. Sprint Retrospective Meeting:-

1. Once the Sprint review meeting is done, this meeting will take place.
2. Participants are Scrum master and Scrum team.
3. Main agenda of this meeting is to do the following task:-
 - (a) Documentation of customer feed back.
 - (b) Best practices followed.
 - (c) Lessons learnt.
4. Team must compliment the best practices learnt from the past and must avoid bad practices in the upcoming sprint.
5. This meeting help to maintain continual improvement of the Scrum practices.

F. Product Backlog Refinement Meeting:-

1. Once the retrospective meeting is done, the Scrum team will focus upon the product backlog and refine it in terms of defining the undefined user stories and also by adding new user stories given by the customer.
2. Product owner (PO) will arrange and conduct this meeting.
3. Participants are PO, Scrum master and Scrum team.

Correlative understanding between Sprint development life cycle and Sprint testing life cycle:-



STLC:- (Sprint test life cycle)

From the above diagram Sprint test life cycle have the following phases in it.

1. Test initiation:- When the development of sprint goes on testing activities are initiated by testing team and eventually developed test strategy.
2. Test Planning:- Master test plan is prepared during this phase with the help of test strategy doc.
3. Test execution:- Testing team reviews userstories and finally developed test scenario, testcases and test data to validate test sprint.

~~Sample user story :-~~

→ # User Story ID: US-001
→ # User Story Name: US-001 SP1_SE_HP
→ # User Story Description: to implement entire functionality of homepage of SE (Sales express)

→ # Task:-

- ① Development of UI of Homepage.
- ② Implementing ML in Homepage.
- ③ Implementing login functionalities.

→ # Acceptance Criteria:-

1. The Home page should successfully run with ~~browsers~~ browsers like IE, Mozilla, safari, Google chrome, Opera,
2. Home page should run successfully on

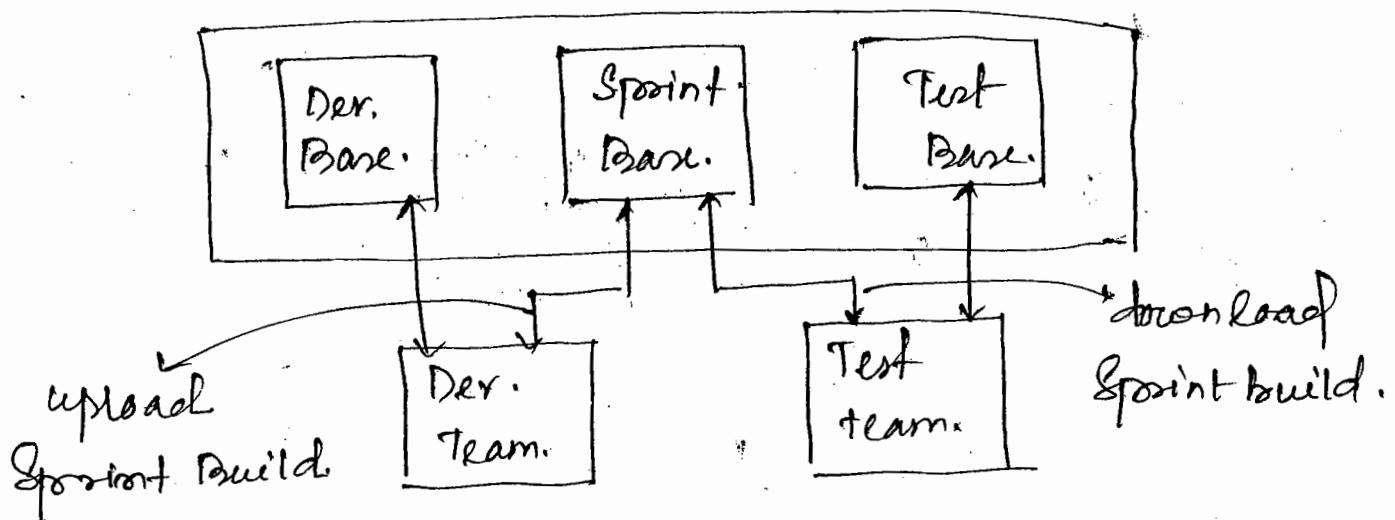
3. On Server operating system like windows 2003/2008, Redhat linux

4. The login activity must be done in less than or equal to 1sec.

Note:- From the above user stories we can understand that there are 3 task which can also be consider as events. while test design test scenario is created for testing each task. We know that for each Scenario we can create multiple test case and for each test cases multiple test steps.

4. Test Execution:-

Before the test execution the sprint build will be released and deployed in the test environment. as described in the diagram below.



5. Test closure:-

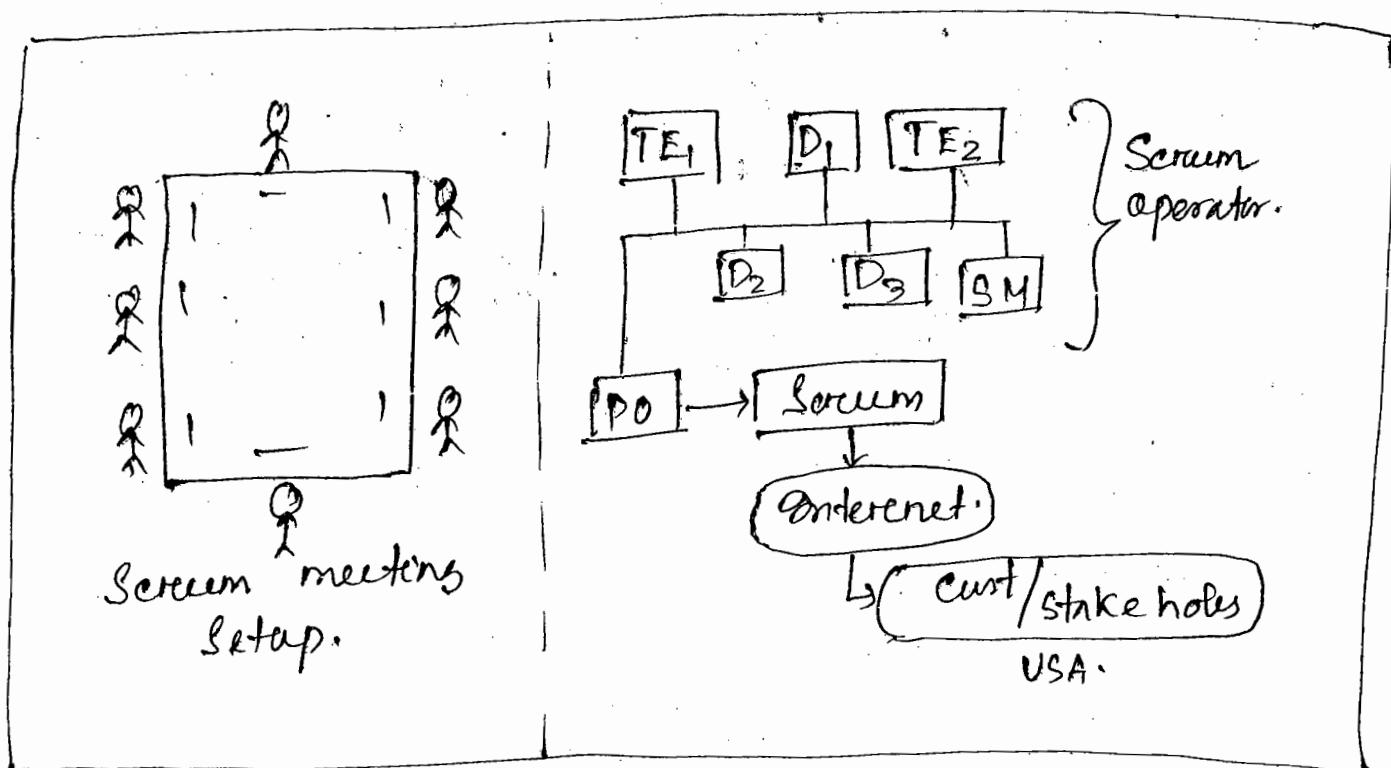
This process in which once the sprint is done with the complete testing, testing activities are put to an end based on certain criteria.

Date - 12.03.2014

Understanding the Burnout down chart:-

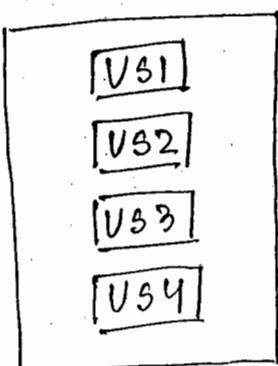
1. The figure reflected in the chart do not represent work done but work left in hours.
2. If the actual efforts is same as estimated efforts that means task is progress on time.
3. If the actual efforts left is more then the estimated efforts left. It mean the task is lagging behind and the Pace of the task has to be increase.
4. If the actual efforts left is less then the estimated efforts left it means that the work is progressive in advanced.

Scrum Environment Diagram:-

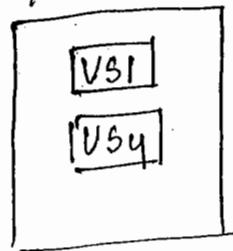


Brief Agile - Scrum Model:-

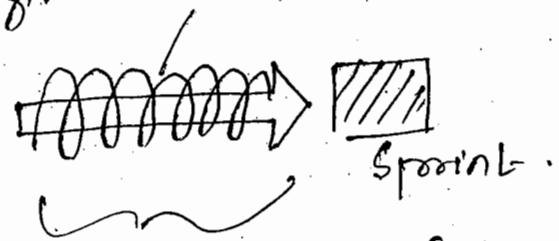
Product Backlog.



Sprint Backlog.



Daily scrummeetings.

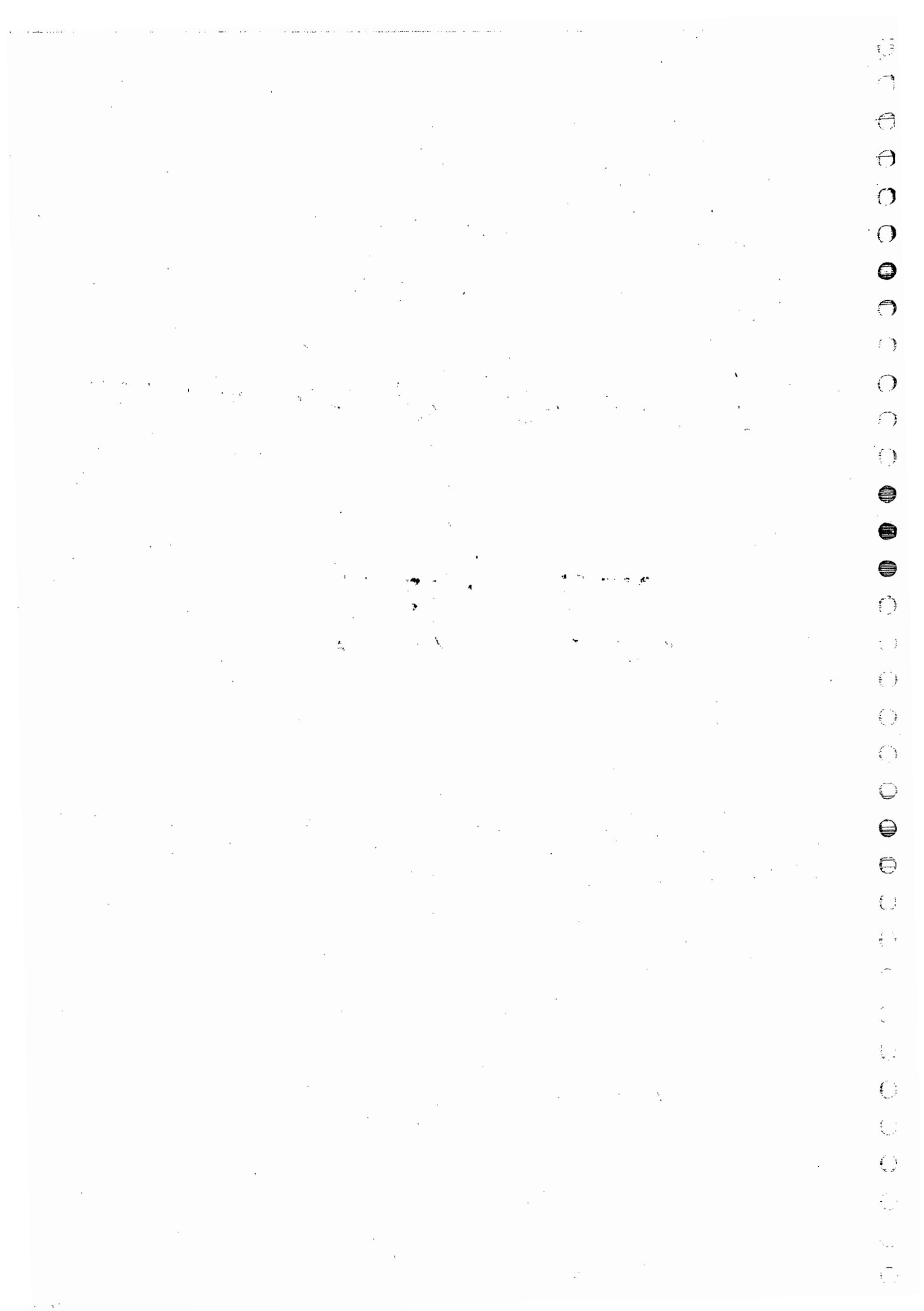


Sprint Implement-
(2-3) w.

Complete.

卷之三

Functional Test Automation.
using
QTP/UFT



Automation Testing:-

Dt. 12.03.2014

Ways of testing :-

Depends on the ways that the testing is carried out

there are two ways.

1. Manual testing:
2. Automation testing.

1. Manual testing:-

It is the way of testing in which all the testing activities like planning, designing, execution etc. are accomplished with absolute manual efforts.

Drawback:-

1. More men are / manpower is required.
2. More time is consumed.
3. It is associated with ~~the~~ tedious process.
4. Repeated testing may not be so easy.
5. Simultaneous activities can not be performed while testing.
6. Human errors.

Solutions to address the above problem :-

1. V-Users :- (V-Virtual)

V-users are virtual users i.e. not real users but they can perform real action on the application hence more manpower issue is resolved.

2. Software component:-

Since the solution is software component it gets executed faster without any tedious / fatigue.

3. Programmatic approach:-

This feature allows the tester to record the activities in order to develop the programmatic statement. Which is capable of producing the action back onto the application when ever it is executed. Hence this solution eliminate the issue of test repeatable.

4. RENDEZVOUS Point:-

This feature is used for collecting all the requirement from the users (May be in different timing) but released them simultaneously on to the application. Hence simultaneous issue is resolved.

5. TOOL BASED:-

The solution proposes a specific tool to carry out testing and so errors can be drastically reduced.
(Human errors)

The above solution can be collectively called as Automation testing.

Automation testing:-

It is another way of testing in which all the drawback of manual testing are effectively address & nullified. Apart from this it provide speed and accuracy for the testing flow.

Advantages of Automation:-

The Cest has given above under solution.

- 1) V-users.
- 2) Software component.
- 3) Programmatic approach.
- 4) RENDEZVOUS Pocnt.
- 5) TOOL Based.

Disadvantages of Automation:-

1. Automation tools are usually costly and so all the company can not effort go for it.
2. If the automation is not planned properly there is every danger that it may consume multiples of manual testing time.
3. Automation required lot of skill professionals.
4. A single testing tool cannot provided complete automation solution.

Scope of Automation:-

1. Since all the area of testing can be covered with manual testing, manual testing is mandatory; Since automation testing cannot be performed in all the area of testing, and since it is performed only the certain area automation testing is optimal.
2. The automation testing is planned usually on the stable product which is basically stabilized by the manual testing. Hence Manual testing always precedes automation testing.

What is to be Automated?—

1. Wherever there is a repeated effort in testing, it can be automated (Regression testing)
2. Wherever there is a complexity in testing (Re-testing) and also big end-to-end process.
3. Wherever simultaneous activities have to be done in testing (Concurrency / Load testing)
4. In the case of testing the information of standard object in the application windows automation tools can be used.
5. Wherever the complete documentation is to be created automation tool can be used as it provides templates.
6. Wherever various reports up to be generated for the corresponding testing activities, automation tools are used.

What an Automated tool can not do? (Limitations of automation)

1. Look and feel of the application window can not be tested.
2. Tool can not suggest and so we can not add a value to the application.
3. Usability testing can't be done by the automation tool.
4. Judgement skill like severity/priority judgement can not be done by the automation.

Type of Automated Tools:-

Depends on the objective of testing, there are various types of automation tools as described below

1. Functional test automation tools:-

These tools are used to carry out functional test automation. In other words functional testing can be automated with this tools.

Ex:- QTP/UFT, Selenium, winrunner, Silktest, RFT (IBN tool), SQA ROBO (Software Quality Assurance ROBO)

UFT- Unified functional testing.

2. Performance testing tools:-

These tools are used for automating the performance testing on the application. (G. meter)

Ex:- Load runner, Web load etc., silk test performer.

3. Test management tools:-

These tools are used for automatic test management activities.

Ex:- QC, Test director, ALM (Application lifecycle management)

4. SCM Tools:-

These tools are used for automatic configuration management.

Ex:- VSS, CVS, share Point etc.

5. Bug tracking tools:-

These tools are used for automating bug tracking process.

Ex:- Bugzilla, QC, track Plus, PVC tracker, PR tracker etc

Tool Table:-

S.I.NO.	Tool-name	Company	Purpose	Type.	Script language.
1.	Win Runner	Mercury interactive Inc. USA	1. Window based, Web based apps. 2. GUI, Functional, DB testing.	Functional	TSL (Test Script Language)
2.	QTP/UFT	HP	1. Window based, web based apps: 2. GUI, functional, DB testing 3. Java/MS.net/ Mainframes /ERP/CRM etc.	- do -	VB Script
3.	Skill test	Segue Inc, USA.	1. Java APPS, Other apps also 2. GUI, Functional, DB . 3. Window /web based app.	- do -	4-test-language
4.	SQA + Rational Rose. Suite	SQA + Rational Rose.	1. GUI testing 2. Functional testing	- do -	SQA Basic/ VB
5.	Selenium.	thought work.	1. Web base apps only . 2. GUI, func ⁿ , DB testing. 3. Java/MS.net/mainframes ERP/SAP/CRM	- do -	Core Java
6.	Test director	Mercury interactive inc.USA.	1. Test Management- 2. Reporting.	Test- management	No script.
7.	Quality Center	HP	1. Test Management- 2. Reporting.	- do -	No script.
8.	Load runner	HP.	1. Load testing 2. Performance testing 3. Stress testing	Performance tools	1. GUI V-UserScript (Frontend opn) 2. DB V-UserScript (Backend opn) 3. RTE V-UserScript 4. Record Script

Test Management Automation :- (QC)

(a) Test management:-

Test management can be understood of the following responsibilities.

1. Test management is responsible for test conducting test kick-off meeting.

2. Test management is responsible for estimation and scheduling.

3. Test management is responsible for planning test activities.

4. Test management is responsible for organising the testing activities as per the plan.

5. Test management is responsible for monitoring test activities.

6. Test management is responsible for preparation of report and taking the same to high level management.

7. Testing ^{management} team acts like a mediator between core testing team and the high level management.

8. Test management is responsible for signing off of testing activities.

(b) Why test management:-

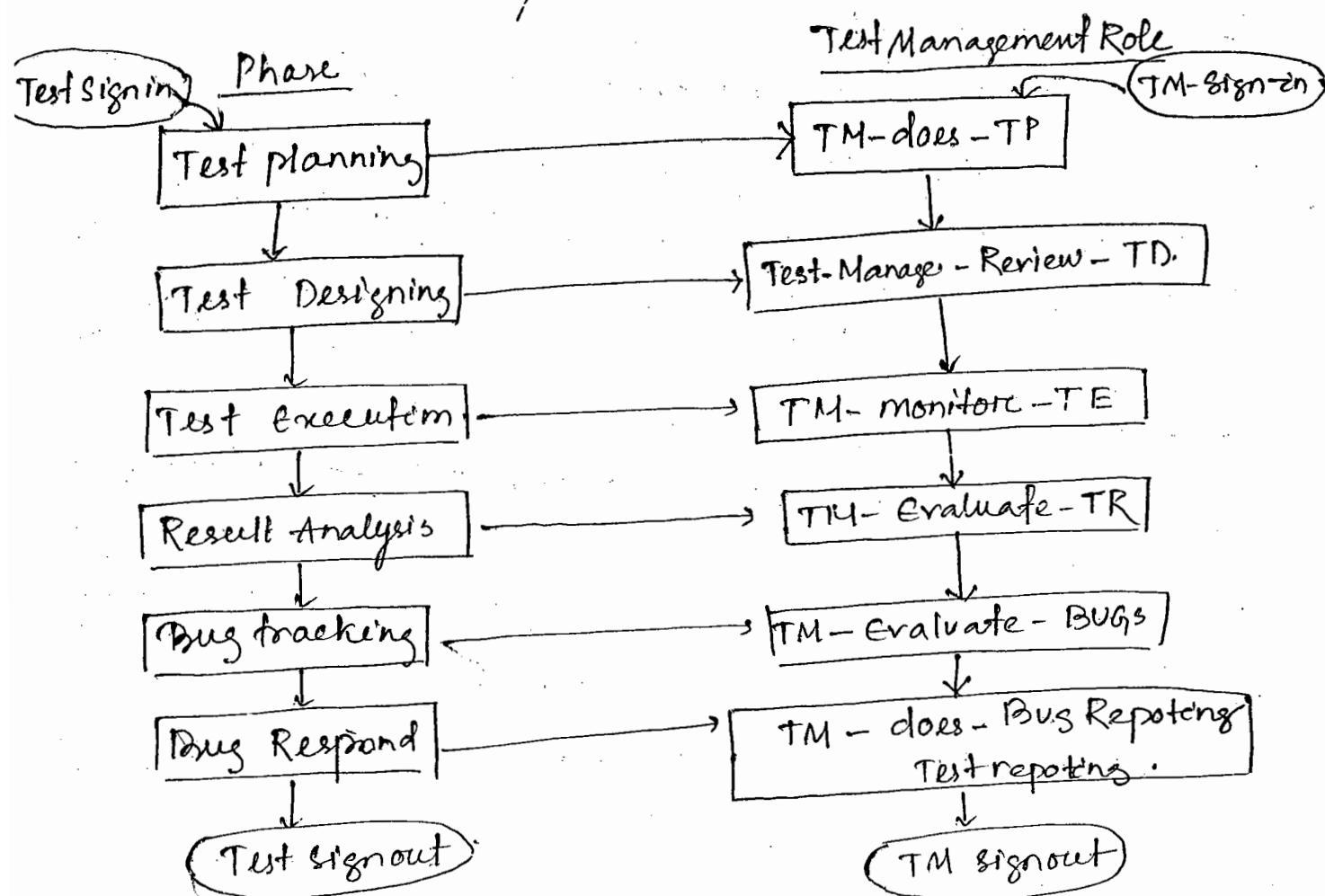
1. To ensure that the testing activities are effectively done.

2. To ensure that the testing activities are efficiently done.

3. To ensure that the testing activities are done with almost optimization.

4. To Motivate and Inspire the teams for more productivity.

(c) How Test management / process of test management :-



(d) Ways of test Management:-

There are two ways of test management and they are

- ① Manual test management
- ② Automation test Management

Why to Automate test Management:-

Since the QTP provides 50% automation for the SDLC and to make it 100% another solution is to be integrated ~~in~~ QTP with QTP known as Test management tool, which is capable of providing automation for test planning, Bug tracking and Bug reporting. Also to make the test management more effective & efficient it

needs to be automated with the help of tools like QC

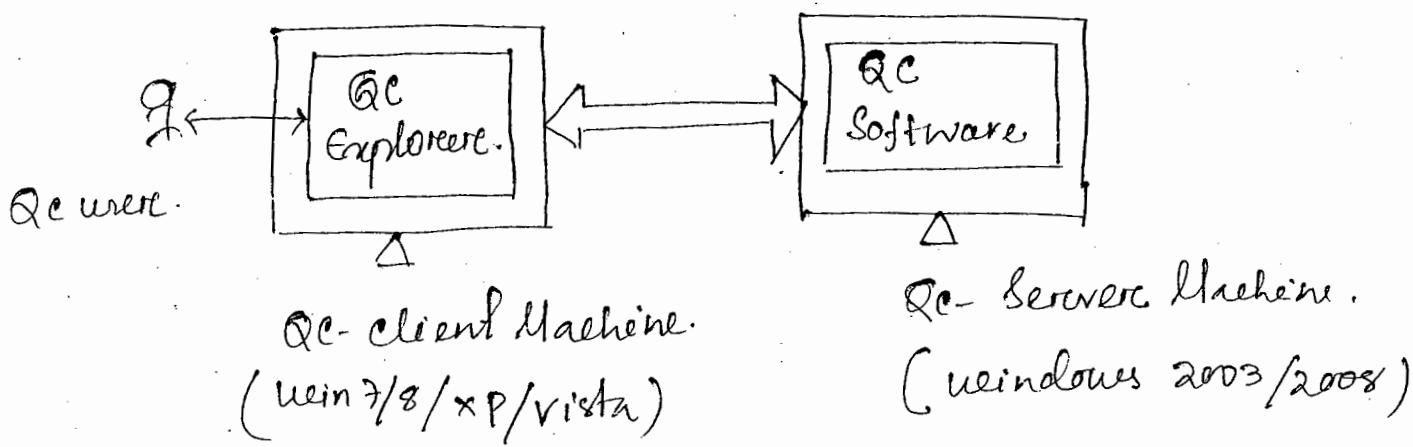
(f) Types of test Management tools:-

QC/ALM :- This is a test management tools comes from HP.

Test director :- It is the management tools which was there before QC and was basically from mercury interactive inc. USA.

(QC) Quality Center :-

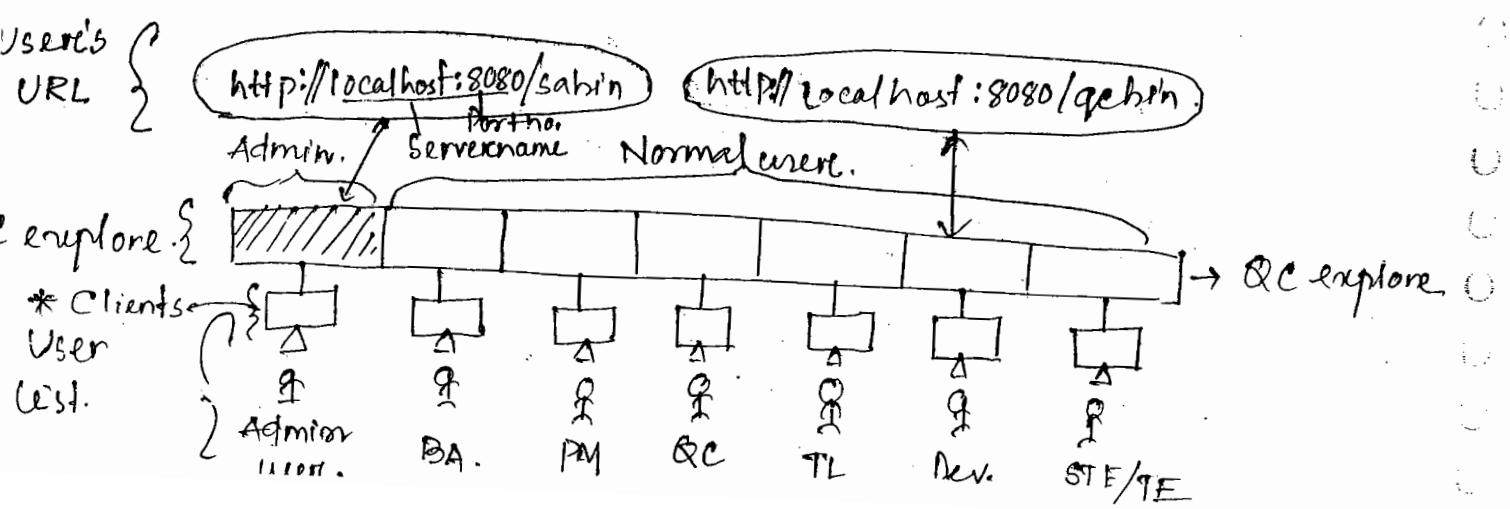
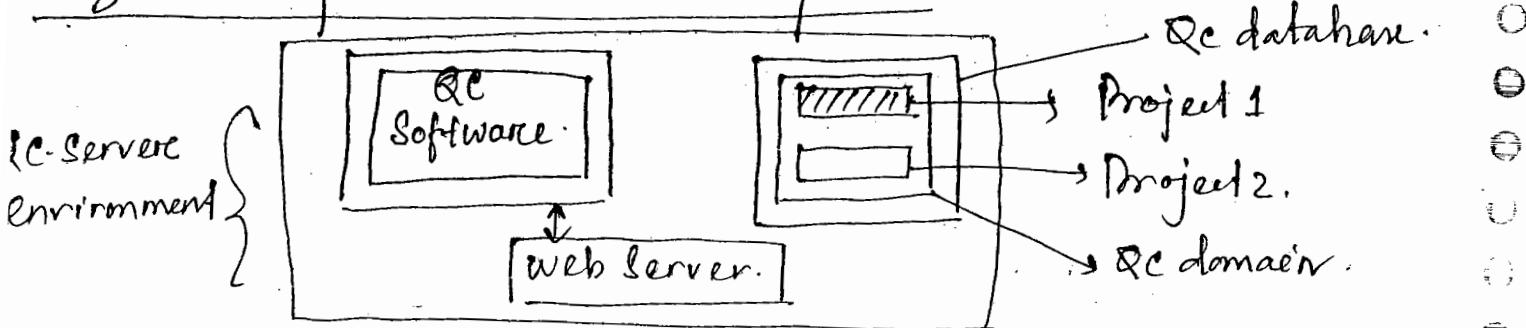
Date - 16.03.2015



1. QC is basically web based test management tools that comes from "HP".
2. QC can be run on windows based as well as web based environments.
3. QC can be run 2 tier architecture as well as 3 tier architecture.
4. QC can be understood in terms of client component and server component.
5. QC Client component has QC explorer installed in it.
6. QC Server component has QC Software installed in it.

- 6. QC explorer acts like an interface/Browser between the QC user and QC Software/products.
- 7. QC client can run on the operating system like windows 7/8/xp/vista.
- 8. QC Server can run on windows 2003/2008.
- 9. QC provides/has a powerful systematic framework which is used to create, maintain, manage different types of information by different type of roles.
- 10. QC has the facility to create business component.
- 11. QC has powerful database system in order to stored Project related information. It has the capacity to integrate with other databases like Oracle/SQL server etc.
- 12. QC has the feature known as dashboard which is responsible of creating wide varieties of reports

High level Operational Overview of QC:-



From the above diagram the following points can be understood:

1. Multiple clients can access the QC server in terms of multiple user roles.
2. Each user can access QC server through QC Explorer which acts like QC browser.
3. The request through web server can reach QC which in turn can contact the QC database to store and maintain information related to project testing.
4. There are basically two types of users that can access QC:- admin user (site admin), normal user
5. One/some of the normal users can be converted into Project admin to carry out some extra responsibility.
6. Admin is responsible for creating domains, corresponding Project repositories and the corresponding users.
7. Normal users will create corresponding information based on specific responsibilities.
8. Admin will access the QC with the following URL

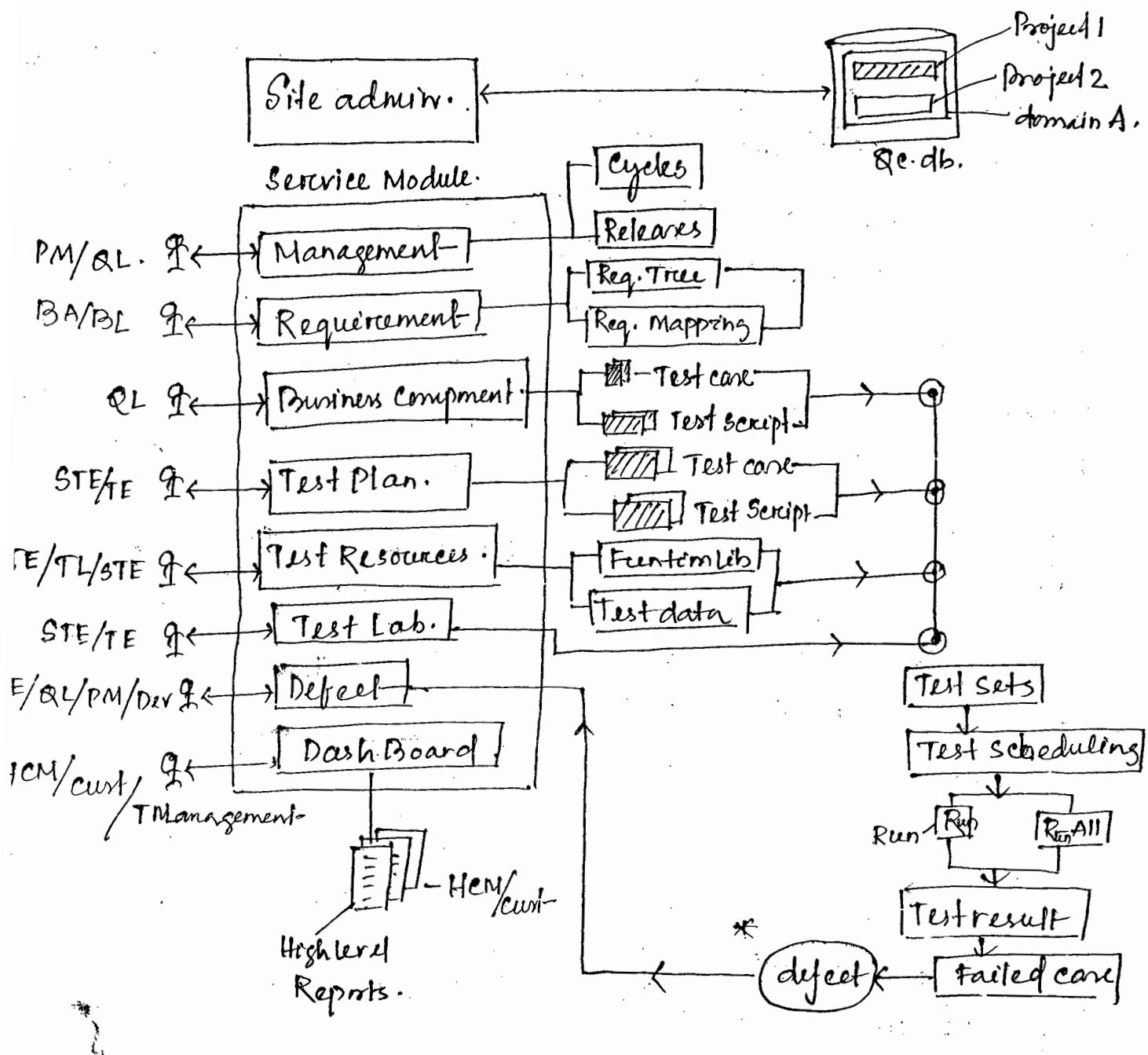
http://localhost:8080/Sabin
↓ ↓ ↗
Servername Portno. site admin

9. Normal user will access the QC server with the following URL:

URL: http://localhost:8080/QCbin

10. QC server can create the database by making use of standard database like SQL Server or Oracle.
11. The information will be stored in a specific project repository that created under specific domain which is created in a specific database.

Low level Architecture Overview of QC:-



QC architecture is can be combination of admin module and service module.

Admin Module:-

1. Admin module is used for creating the specific domain for the project.
 2. It is used for creating project repositories for the domain.
 3. It is used for creating various users.
 4. It is used for creating permissions of the users.
 5. It is used for creating QC database to store the case

6. It's used for storing and maintaining QC license information. (How many license are basically available, how many are active and how many are inactive etc.)

Service Module :-

Service Module have various submodules with specific responsibility as describe below :

1. Management :-

a. This feature is used for creating maintaining and managing the project information required for the test manager. In other words It's mainly used for creating the number of cycle and nos of ~~releases~~ releases.

b. PM/QL will be using this features.

2. Requirement :-

a. Used for creating , maintaining and managing the requirement information.

b. This feature is used by BA/QL.

c. It's used for defining the scope of testing.

d. It provides a requirement tree in which requirements are created and maintained in a structure / classified manner.

e. It provides Requirement mapping facility (RFTM).

f. It provides status graph that describe the status of task.

3. Business Component :-

a. Business component exist in terms of test case / test scripts.

b. This feature is used for creating the business components.

- c. Snapshot can be while creating the business component one must give the inputs like.
- detail (purpose)
 - Snapshot (screen shot)
 - Parameters (input for execution)
 - Design steps (test cases)
 - and Automation (test script)

- d. It provides requirement mapping & status mapping graph.

4. Test Plan:-

- a. It is used for creating the test design (test cases/test script).
- b. It is used by test engineer.
- c. It is used for defining the test strategy.
- d. It is used for creating test plan tree (structure, classified, sequential test cases)
- e. It is used for requirement mapping and status graph.

5. Test Resources:-

- a. test resources like test data, recovery scenarios, function library etc can be created maintain and managed.

- b. It is used by TL/STE/TE.

Q.

6. Test Lab:-

- a. It is basically used for test execution (Manual/automation)
- b. It is used by STE/TE to conduct test execution.
- c. It is used for creating various test sets.
- d. It is used for scheduling the execution of test sets.

- e. It is used for executing one test at a time ~~etc~~ or all the test at a time.
- f. It is used for requirement mapping and looking for status graph.

7. Defects:-

- a. Used for creating maintaining and managing the defect information.
- b. This feature provide DPD template.
- c. TE will access this feature for defect filing / documentation. QL will access for defect ~~for~~ evaluation. PM will access for defect assignment and developer will access for the defect rectification.
- d. It's used for demonstrating Bug resolution Process (Bug life cycle).

8. Dash Board:-

- a. Main purpose of the feature is to generate wide varieties of report.
- b. This feature can be ~~access~~ by test management/ High level management/ customer.
- c. Different types of graphs/ reports like:
 - normal test report
 - Table format report
 - Piechart
 - Bar graphs etc
- d. These graphs are used for taking immediate and appropriate decision as they are detailed Perfect reports.

Date - 17.03.2015

Demonstration of QC :-

Roles and Responsibility associated with QC :-

1. Site Administrator :-

- To create project domain and project repository for the sake of testware.
- To create all the users associated with QC
- To convert some of the normal users (usually QL & PM) into Project admin users.

2. Project Administrator (Usually QL) :-

- To assign all the users to the project.
- To assign permissions to the users depends upon their responsibility.
- To assign the module access to the user based on upon the scope of responsibilities.
- To create releases and the respective cycles for the project to be tested.
- To create the tasks for the testing team and to assign them to the respective testers.
- To create the subtasks/ sub topics to the main tasks/ main topics.

3. Normal User :- (Testers responsibilities)

- To view the requirements in terms of the task assigned to him.
- To create test Scenario's
- To create corresponding test cases to the respective scenario along with the test step.

- c. To do the requirement mapping for the test design.
- d. To conduct test execution.
- e. To file/document the defect to/in QC.

How to launching QC:-

(a) Navigation:-

- click start button.
 - Select all program
 - Quality center select
- ↓

QC Explorer

click.

QC Explorer/ Browser is opened up.

Start

Task of site admin :-

Enter / Select : <http://serverqc:8080/satin>

Click Go button.

Login

Login :- [Enter username & Password]

Click on Login to

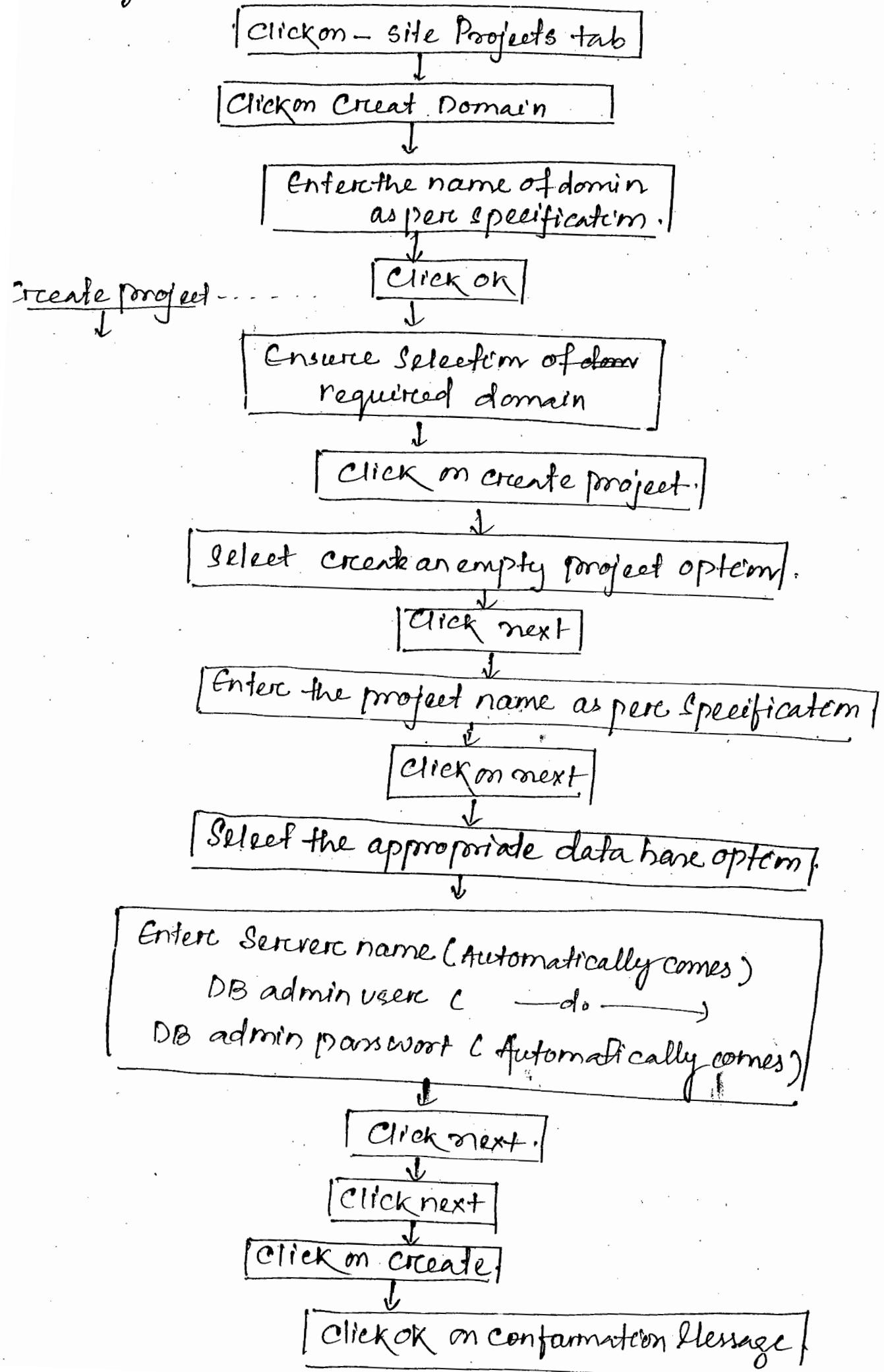
See the Site admin home page

Tasks:- 1

↳ To create project domain & Project repository as per the following specification.

- a. domain name (SE_Domain_17)
- b. Project name (SE_Project_17)

Narigation:-



Note:- in case project repository is used to be control system select enable versioning option check box option.

To complete task 2:- to create the required users for QC as per the following specification.

Sl. No.	Role.	Username	Password.
1.	Project Manager	PM-17	PM PW
2.	Quality Lead.	QL-17	QL PW
3.	Tester.	TE1-17	TE1 PW
4.	Tester.	TE2-17	TE2 PW
5.	Developer.	dev-17	DEV PW
6.	Business Analyst.	ba-17	BAPW

Navigation:-

Click on site users after selecting the required project.

↓
Click on add user icon/New user icon.

Enter Username as per specification (PM-17)

↓
Click on OK

↓
Select the user, click on Password icon.

↓
Password window come / Enter PW as per specification (PM PW)

↓
Click on OK

Click on OK on confirmation window.

↓
Repeat the above step for the rest of users.

Such as: QL

Tester1
Tester2
Dev
BA.

Task-3:-

To convert the following users as project administrator:-

Project Administrator

→ Quality Lead

→ Project Manager.

Inigation:-

Go to site project and click on the project user on the right side

Click on add user icon

Select Add from user list option.

Select the required user on the right side.

Click on left position arrow mark

Repeat the above process for required users

Check/Select the check box option under project administrator field, for the respective users to

Click on Logout having done all the responsibilities

Project Administrator task:-

Enter the "URL": http://serverqc:8080/qcbin

on qc explorer window to get the login page

Click on Go

Enter / type Login ID & Password ~~that we kept of QL~~
as Project admin.

Click on authenticate button.

Ensure appropriate domain and Project are displayed in given box.

Click on login to go to
QL Home page.

Click on close button on QC initial page.

Task 1:-

To assign the users to the projects.

Navigation:-

Go to tools click on
Customize

Click on project user
link.

Click on add user button
at the bottom

Select "Add user from the list of Site users"

↓
click on OK

Repeat the same process for the rest of users.

Task-2:- To assign task privileges to the selected users as per the following specification.

Sl.No.	Role	Permi
1.	BA	Viewer.
2.	PM.	Project Management TD Admin.
3.	QL	TD Admin.
4.	TE1.	QA Tester.
5.	TE2	QA Tester.
6.	DEV	Developer.

Navigation:-

Select specific user under
Project user list

↓
Select the option from the list of
"not members of" and move the same
to the members of list.

Repeat the above procedure for all the users
and click on save button.

↓
Click on confirmation window

Task-3:

To assign module access privilege to the user
as per the following specification.

Group	Defect-testplan	test Lab.	Requirement	Business Component	Management
Developers	✓	✗	✗	✓	✓
Project manager	✓	✓	✓	✓	✓
QA Tester	✓	✓	✓	✓	✗
TD Admin/QL	✓	✓	✓	✓	✓
Viewers	✓	✓	✓	✓	✓

Navigation:-

↓
Click on module access link.

As per the specification, customize the module
access by writing '✓' or '✗' under each and
every module against the specified roles.

↓
Click on save

↓
Click on Return to come to Homepage.

Task-4 To create various releases and cycles for the corresponding projects as per the following specification.

Project: SE_PROJECT

Releases: SE_R-1.0

SE_R-2.0

SE_R-3.0

Cycle_SE_HP_Build 1.0
Cycle_SE_HP_Build 2.0
Cycle_SE_HP_Build 3.0

Navigation:-

Click on Management module icon on left side.

Click on new folder icon

Enter SE_Project and click OK.
Proj.name

Select the project folder
Release
Click on the new folder icon.

Enter Releases folder name
as per specification.

Repeat the above process to
create rest of releases

Select the 1st releases folder and
click on new cycle icon.

Enter new cycle name as per
specification.

Repeat the above process for multiple
cycles.

Task 5:- To create the tasks for the tester and assign them as per the following specification.

Project: SE_PROJECT

Tasks:

- | | | |
|----------|-------|-------|
| 1. SE_HP | _____ | Tes1. |
| 2. SE_SM | _____ | Tes2 |
| 3. SE_CM | _____ | Tes1. |

Navigation:-

click on Requirement module icon.

Select the requirement folder to create requirement tree.

Click on new folder icon

Enter the name of folder as per specification

click on ok.

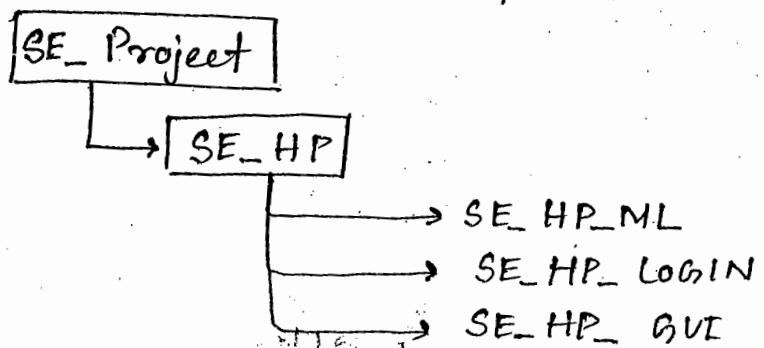
Select the project folder and create sub folders as per specification.

Select the subfolders and click on the information right below the author column.

Select an appropriate role to assign selected module to him. Repeat the same process for all modules.

Task 6:-

To create sub-tasks/sub-topics for the main tasks/topics. as per the following specification.



Navigation:-

Select the main folder.

Select one of the subfolders under project folder.

Click on new requirement icon.

Select the type of requirement (functional etc) and enter requirement name as per specification. click ok.

New window will come where we again enter the req. name and req. type then click on close after complete all the req.

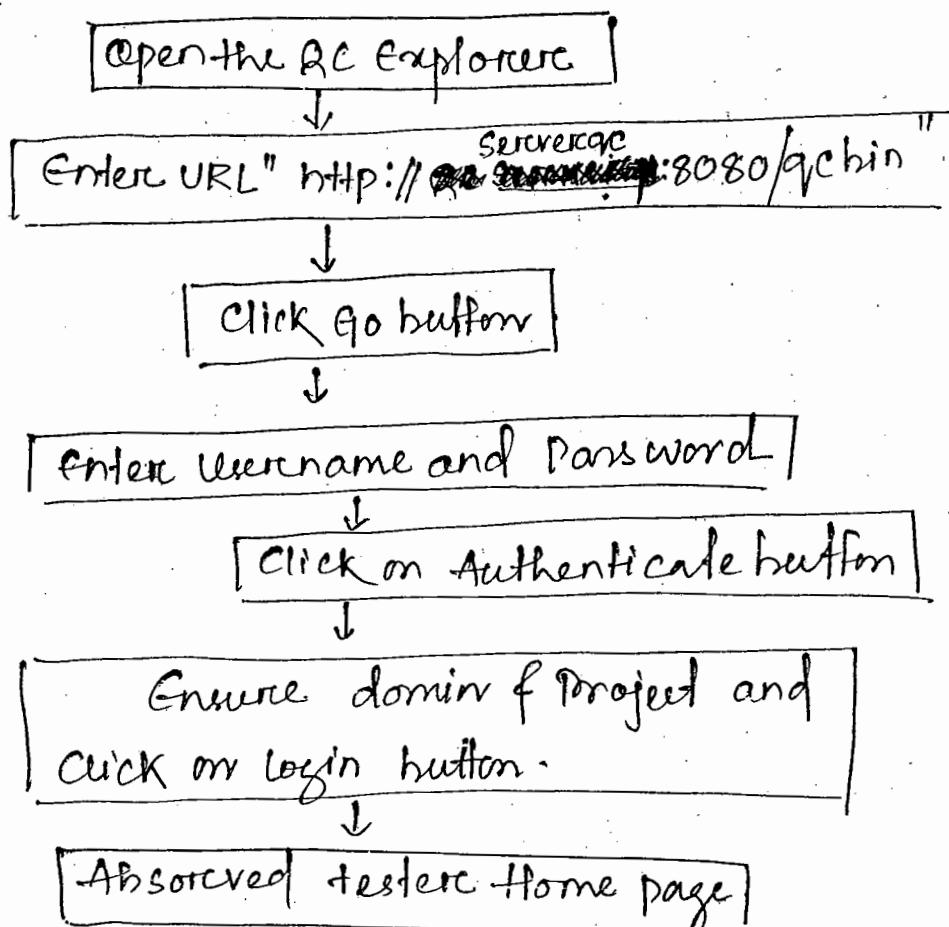
Normal Users (Test Engineers) :-

Dt. 18.03.2015

Task-1:-

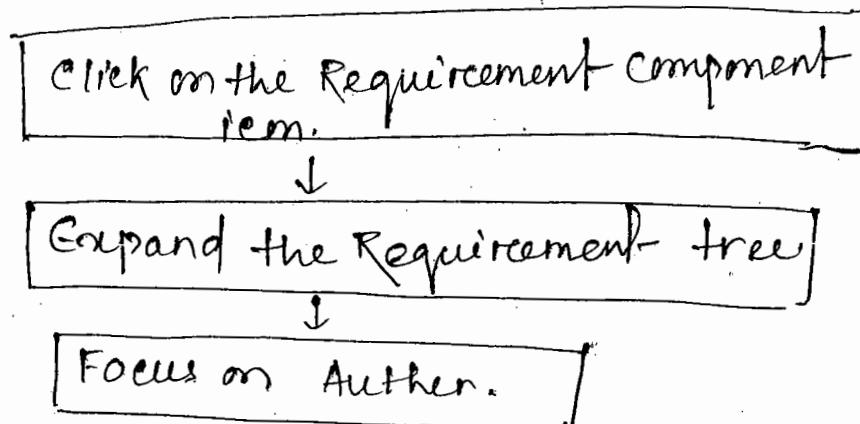
To enter into tester's homepage of QC

Navigation:-



Task-2:- Review the task/requirement that are assign to the tester.

Navigation:-

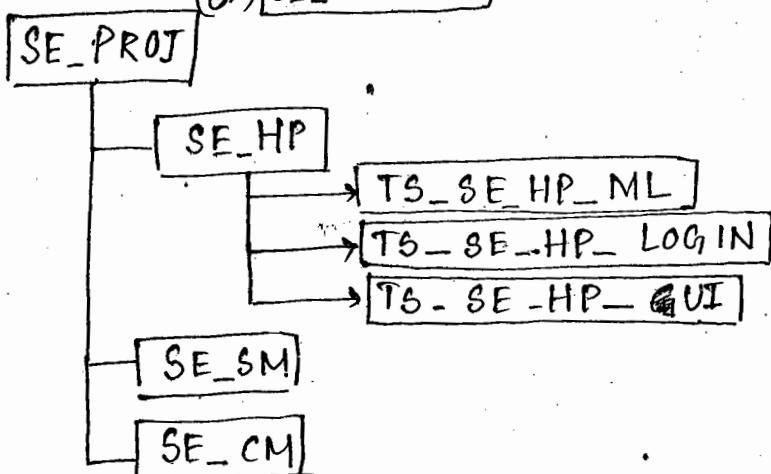


Task-3

To create scenarios for a specific module as per the following specification.

Specification:

(or) SE_PROJECT



Navigation:-

click on Test Plan module icon



Select the Subject folder and click on new folder icon.



Enter project folder name



Select the project foldername and create Sub folders as per specification.

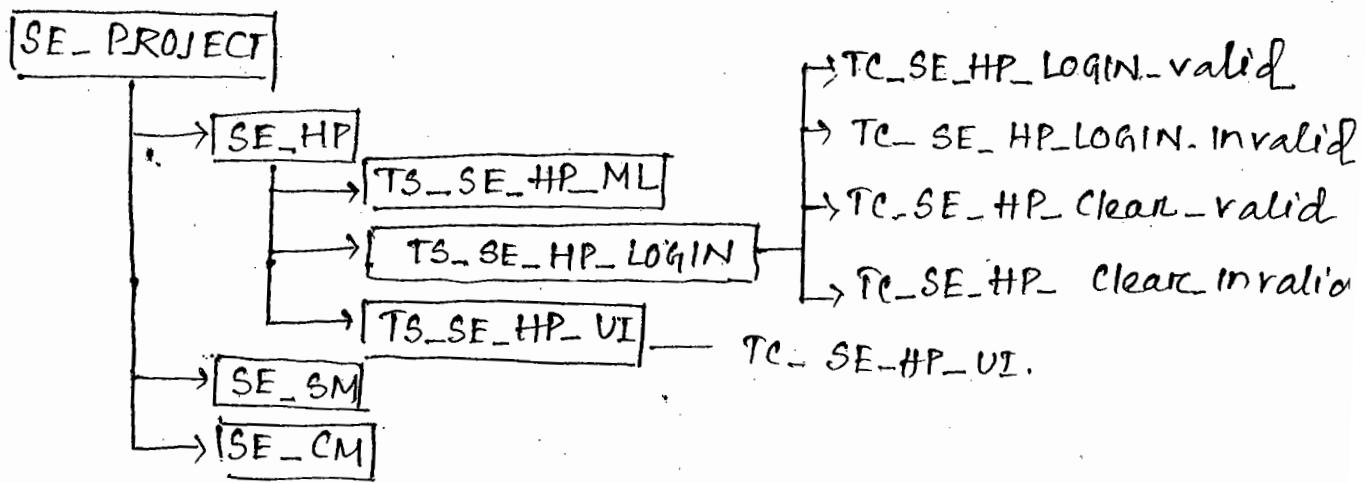


Select the Sub folder and create ~~the~~ set of subfolders in terms of scenario as per specification.

Task 4:-

To create test cases for the respective test scenarios as per the following specification.

Specification:-



Navigation:-

Select the specific test Scenario's folder and click on "new test" icon.

↓
Enter type of test (like Manual)

↓
Enter test case name as per specification.

↓
Repeat the same procedure for all the test cases.

↓
Select a specific test case.

↓
Click on design step on the right side).

↓
Click on new step icon.

Enter the test case description and expected values/
Expected behaviour and click on 'ok'
for ex:- Enter UN & PW as per table one
and click ok.

To attached click on attachment icon
again click on attachment icon.

Select the file in the open dialog
box click open.

Continue this process till all the
test steps are complete

Task-5 :- To do the requirement mapping to the test
cases.

Navigation :-

Click on Requirement
module icon.



Click on requirement menu
that present in main menu.



Click on requirement details
we get requirement detail window.



Click on test coverage
icon 

Select on the right side expand test plan tree and select the corresponding testcase.

↓
Click on left point execute or arrow mark to map it to requirement.

Absorbed that the selected testcase are moved to the left side.

↓
Click on the upward arrow mark to see the coverage chart

↓
Click OK

Task-6:-

To execute test cases in QC on the application window

Navigation:-

↓
Click on test Lab module icon.

↓
Create a project folder with the specific name like SE_Project.

↓
Select the project folder and click on new test set icon to create the test set with the specific name

↓
Testsetname - Login - Testset.

↓
Select the test set click on execution grid on right side.

Click on Select test button/icon.

while click, test plan tree
is automatically come on
right side.

Select a specific test case from test
Plan tree on right side.

Click on Left point execute

Click on Run button.

click on "begin run" button to
Perform test execution on the
application window which already
open.

while executing each and every step
actual value and its status must be updated.

Once the testing is done click
on "End Run button" / Red button.

23K7:-

To file the defect in a defect profile document
of QC..

Navigatm :-

click on defects module icon.

DDP template provided.

click on new defect icon.

On the form the following mandatory field must be enforced.

- *1. Summary *4. Severity
- *2. Detected by 5. Description (optional)
- *3. Detected on

and rest of the optional fields.

error message window for valid input that may be written in Summary.

Sample.

valid user name and valid Password as per table 2 entered and consequently the system displayed error message instead of home page. ^{written} in description.

Sample.

Click on submit button.

If another defect then continue this step.

(or)

Repeat the same procedure for various defect.

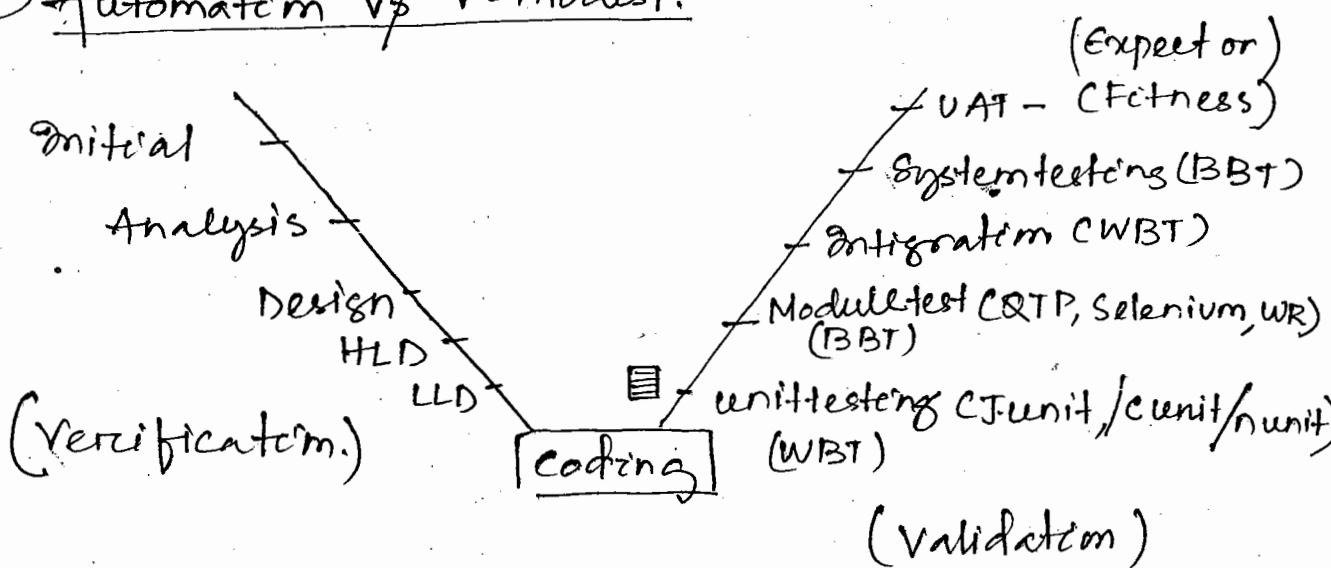
+

click on Logout come out from Qe

Function test Automation Tool (QTP)

Function test automation tool is basically used for automating functional test activity of an application.

① Automation Vs V-model:-



From the above diagram the following points can be understood.

1. Hardly any scope of automation for verification.
2. Unit testing can be automated with white box testing tools like J-unit (Java program), C-unit (for 'c' program), N-unit (for '.net' program).
3. Module testing can be automated with black box testing tools like QTP, Selenium, WR etc.
4. Integration testing can be automated with WBT tools.
5. System testing can be automated with black box testing tools.
6. User Acceptance testing tools can be automated by the tools like fitness, expect.

2. Automation v/s level & type of testing:-

Levels.	Functionality	Performance	non-functional
0 - levels. (Smoke/Sanity)	Manual/ Automat	Automation	Manual.
I - levels. (Detail function testing)	Manual	Automation	Manual
II - levels (Regression/ Retesting)	Automation	Automation	Manual
III - levels. (Final regression testing.)	Automation	Automation	Manual.

From the above table:-

1. Smoke testing can be done manually whereas as Sanity testing can be done with automation.
2. For the first time detail testing can be done with manual testing only.
3. Regression testing and retesting can be automated.
4. Performance testing can always be done with automation.
5. Other nonfunctional testing except performance testing can be done with manual testing always.

Evolution of Automation tools:-

1. Evolution 1 / 1st Generation:-

Market are using commercial tools in combination of private language.

Ex:- WinRunner (WR) → TSL

SQA Robo → SQA Basic

SILK test → Utest language.

2. Evolution 2 / 2nd Generation:-

Market are using the commercial tools along with public language.

Ex:- QTP → VB Script.

Rational Robo → VB.net

Test complete → C#

3. Evolution 3 / 3rd Generation:-

Market are using the open source tools along with public language.

Ex:- Selenium → Java

SAHI → VB Script.

4. Evolution 4 / 4th Generation:-

Market started depend on script language rather than tools. Test engineer is able to create test script independently with the help of several script language to carry out testing on the AUT. Hence the tester try to becomes automation expert rather than tool expert so. script language like:-

1. Perl

2. Ruby

3. VB script

4. Java

5. Python.

Generic Behaviour of Automation Tools:-

① Tool can focus on the AUT:—

(a) Manual Invoking:- Application window can be invoke manually to be tested by the tool.

(b) Tool Invoking:- Tool has the capacity to invoke the application window and then focus on it to carry out testing.

② Tool is capable of capturing the information of the objects present in the application window.

(a) Tool has feature known as "Learning" which is basically a process in which the object information is captured.

(b) The captured object information is stored in a location known as "Object Repository" (OR)

(c) "Object information" present in 'OR' is used for identifying the objects during run time while testing

③ Teach the tool how to carry out testing

or test engineer teaches the tool how to carry out testing:—

(a) Tool has the "Recording feature", which is the process in which the tool is kept under observation mode.

(b) During recording tool creates programming statements for the corresponding action while recordings.

[Note: for reaction/response no statements are generated]

(c) The outcome of recording is basically known as "operational script".

(d) Since operational statement are not enough to perform testing it need to be enhanced with the verification statement which can be created with the feature known as "check point". Hence

Testing = Operation + Verification.

④ Tool execute the test scripts to carry out testing on the AUT :-

② Before executing the test script, the AUT must be kept under "base state" (initial state in which all the fields are kept empty).

(b) While execution operational script will emulate operation and verification statements will perform testing. Result are concluded and display in the test window once the execution is over.

(c) While execution captured information (CI) is compared with Run Time Information (RTI) to ensure that the object is identified successfully so that the operation verification and hence testing is successful.

$\begin{pmatrix} CI \\ RTI \end{pmatrix} \rightarrow ID \rightarrow \text{operation} \rightarrow \text{Verified} \xrightarrow{\text{(Successful)}} \text{Testing}$

(d) sometime the test script needs to be refined to justify the total test requirement. This refinement can be done with various feature like Parameterization, Synchronization, error handling, recovery scenario etc.

for example:- Parameterization is the process in which the constant value present in the script are replaced with parameters to make the script generic and dynamic to be used by / to be used for multiple test data elements/test data i/p..

1. Mechanism of Automation:

1. Learning.
2. Identification.

2. Activities of Automation:

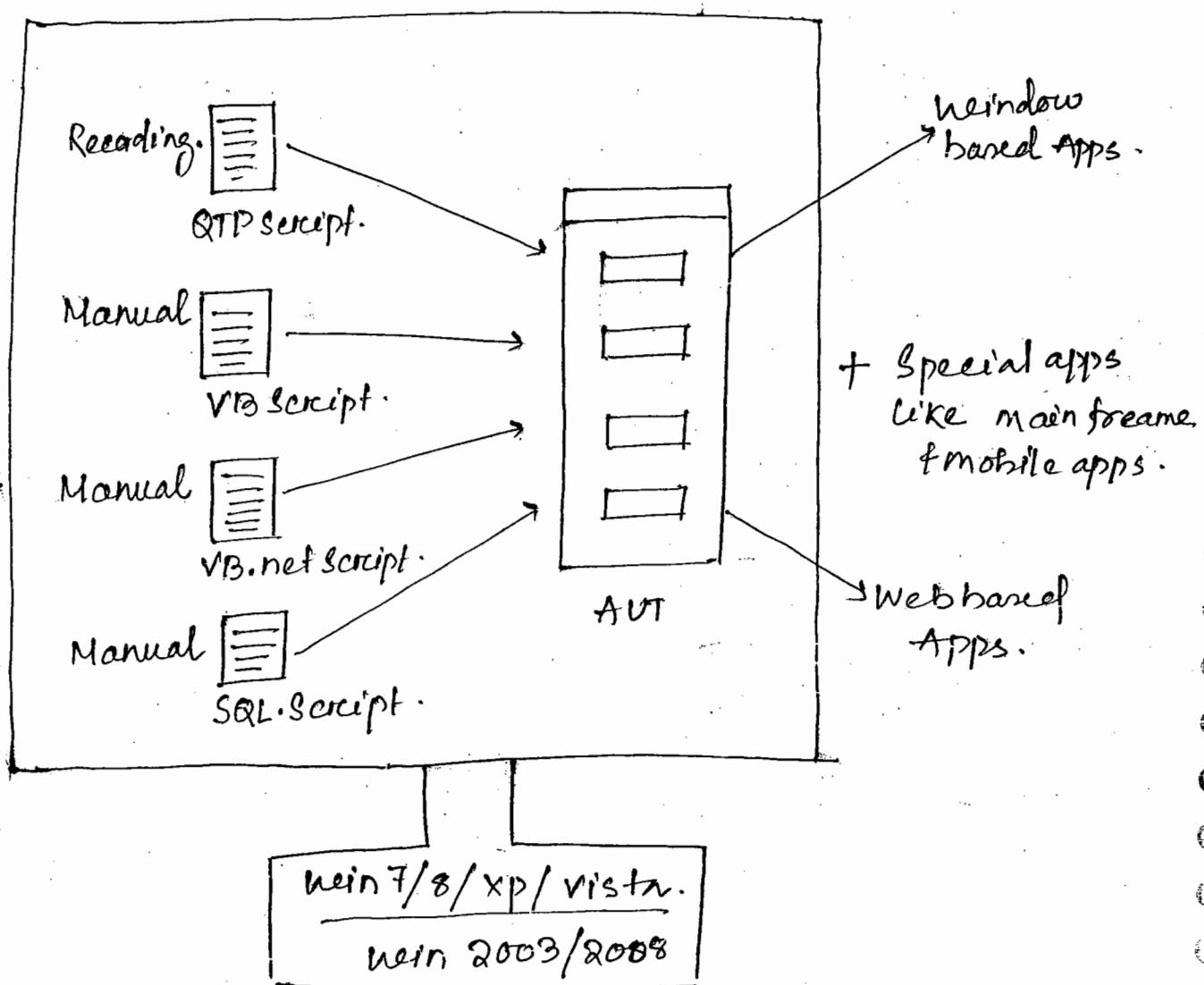
1. Recording.
2. Execution.

3. Level of Automation:

1. Generation of basic test (Operational Script.)
2. Enhancement of the test adding verification statement to operational statement
3. Refinement of the test. (Refining the script with various feature to justified test requirement)

QTP 11.0

Quick Test Professional (QTP) is functional test automation tool that comes as a licensed product from "Hp". It cannot offer test management automation and performance test automation solutions.



Vere.No.	Year.
5.5	→ 2002
6.5	→ 2003
7.6	→ 2004
8.0/8.1	→ 2005 → 2006
9.0	→ 2006
9.1/9.2	→ 2007
* 9.5	→ 2008

Vers. No.	Year
10.0	2009 - 2010
11.0	2010 - 2011
11.5	2011 - 2012
12.0	2013 - 2014

About QTP :-

1. QTP is functional test automation tools capable of testing functionality, GUI and database of an application.
2. QTP is capable of testing wide range application like Java, .Net, C/C++, Embedded System, (SAP) ERP, CRM (Siebel), and Special application like mainframe, mobile apps etc.
3. QTP is capable of testing window based as well as web base applications.
4. QTP can be run on the machine of the operating system
win 7/8/xp/vista/win 2003/win 2008
5. On recording, QTP generates its own script known as QTP Scripts which is obviously supported by the tool.
6. Manually written VB Script are supported by the QTP.
7. Manually developed .net Script are supported by the QTP.
8. SQL Script (SQL statements) are supported by the QTP. while testing.

Q. Why QTP is required for the testing?

Ans:-

(1) QTP has more compatibility with various technologies
(2) QTP is adaptable for various environment like, Standard alone, client server, intranet/internet(web) and distributed environments.

(3) QTP has Key Word Driven (KW) view feature:-

It's capable of generating non programmatic statement apart from programmatic statement

(4) QTP has Active screen feature:-

It's a series of snapshot that created for the corresponding action while recordings.

(5) QTP has Low level Recording (LLR) feature:-

It is a recording mode to recode or to create meaningful and appropriate statement for the operation that are perform on the non-standard object (Objects that are created with the technologies not compatible with QTP)

(6) Smart Identification:-

In case normal identification fail to identify the dynamically changed object QTP has an alternative solution known as smart identification with which it try tries to identify the object.

⑦ QTP has descriptive programming facility:-

Descriptive program is a way of programming

in which object information is directly kept in it avoiding object repository. So that identification of the object can be done faster. Hence test performance is increased.

⑧ QTP has Function library facility:-

It is the storage place where the test engineer can create, maintain and manage several user defined function.

⑨ QTP has AOM feature (Automation object model) :-

QTP has the framework / facility to use / create various objects for the sake of implementation of the test automation.

⑩ QTP supports Modular Structure :-

The test engineer can write multiple script file which can be integrated. One script file can be called from within the other script file.

⑪ QTP is very much user friendly tool so that the test engineer can configure, create, enhanced, refine, debug, execute the test scripts and do the result analysis quickly and easily.

⑫ QTP uses VB script as a scripting language so that the tester can create and maintain the test script easily and quickly.

Main Component of QTP:-

QTP can be basically divided into the following components

① Add-in Manager:-

It is the first window that is popped up/come up when ever the QTP started that contains the list of add-in, so that the tester can select required add-in, based on the application to be tested.

Add-in :- It is a plugin program which can be executed on selection to provide the compatibility between the technology and QTP.

② Menu/Icon Bar:-

* MenuBar contains list of menus and submenus items in terms of classified activities of QTP.

There are two types of menu that we can deal with.

1. Standard Menu:- which always appear and available to the tester.

2. Contextual Menu:- This menu comes in to visibility only when ever it's required at the time of writing the scripts manually by the tester.

* Icon Bar contains various icons for frequently used instruction.

3. Test Pane:-

It is a kind of integrated development environment-(IDE) for QTP where test can be developed, modified, executed, refined etc. Test pane can be divided into two windows.

1. Expert view:- It's a program base editor in which tester can record or manually generate the test script, the test script can be enhanced, refined, debug, executed in this editor.

2. Keyword view:- It's an editor in which non-programmatic statements can be generated simultaneously and the information can be displayed in terms of the following fields.

a) Item:- It contains all the list of objects that are participated in the recording.

b) Operation:- It contains list of operations performed on the objects.

c) Values:- It contains list of value ~~as~~ is associated with operation.

d) Documentation:- It contains the description of activities in single lines.

4. Action Screen:-

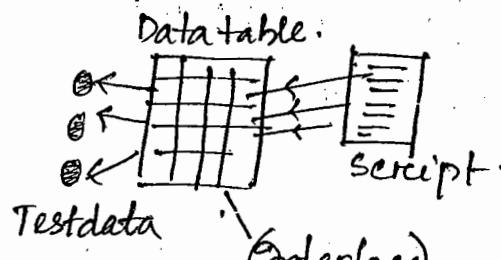
a. It is the feature that can display series of snap-shots for the corresponding action while recording.

b. It is also known as visual test recorder.

c. The information present in the script can be modified with the help of active screen if not the actual screen.

(5) Data Table:-

- a. This feature is used for holding the test data.
- b. The data can be created manually and also data can be imported from external resources.
- c. Data table acts like an interface between the test data & test script.



- d. Data table has two sheets:-

1. Global sheet:- That contains the data which can be shared by all the script file.

2. Local Sheet:- This contains the data which will be used by the current script file only. Hence it is local.

- e. Data table can be of two type.

1. Design time Datatable:- Data table that is refers to outside execution.

2. Runtime Datatable:- It is the data table that can be referred to while execution.

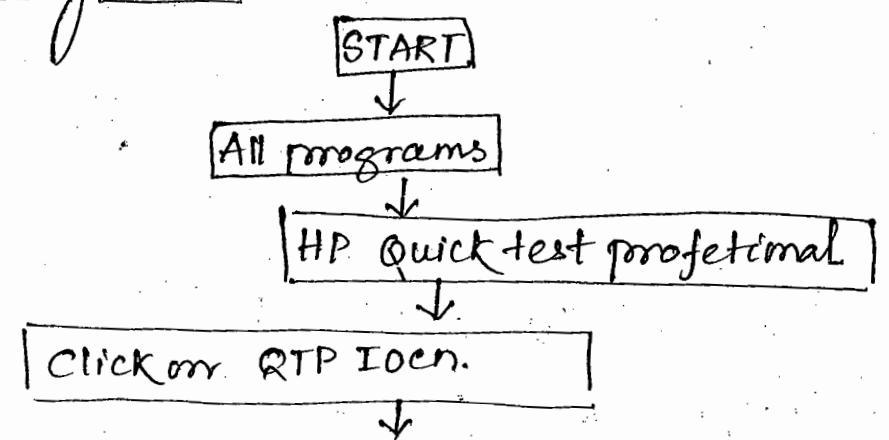
Note: Runtime datatable details can be viewed in the result window once the test execution is over.

⑥) Debug Viewers :-

It is a tool provided by QTP to view the information related to the script while de-bugging.

de-bugging :- It is the process in which errors that are associated with script are encuntered so as to make the script error free.

Lunching QTP:-



Select the required AddIn and click on OK in the AddIn manager window.

Note:- The required addin can be purchased so that they can be displayed along with the default addin.
(Active X, Visual basic, Web)

To invoke login dialog box navigation

Start all programs

Select HP Quick Test professional

Sample application

Click on flight icon

To send the documentation to the high of keyword view to high level management (HLM) :-

Clipboard! - temporary memory where info. will store for temporary

[Go to edit]



[Click on copy documentation to clipboard]



[Open another notepad right click on it and past it.]



[Save it and send it as an attachment through an email.]

To view active screen:- (It will be save internally)

Navigation:-

[Go to view]



[click on active screen]

To view data table:-

[Go to view]



[click on datatable]



[Double click on column header enter the column name in dialog box and click on OK.]



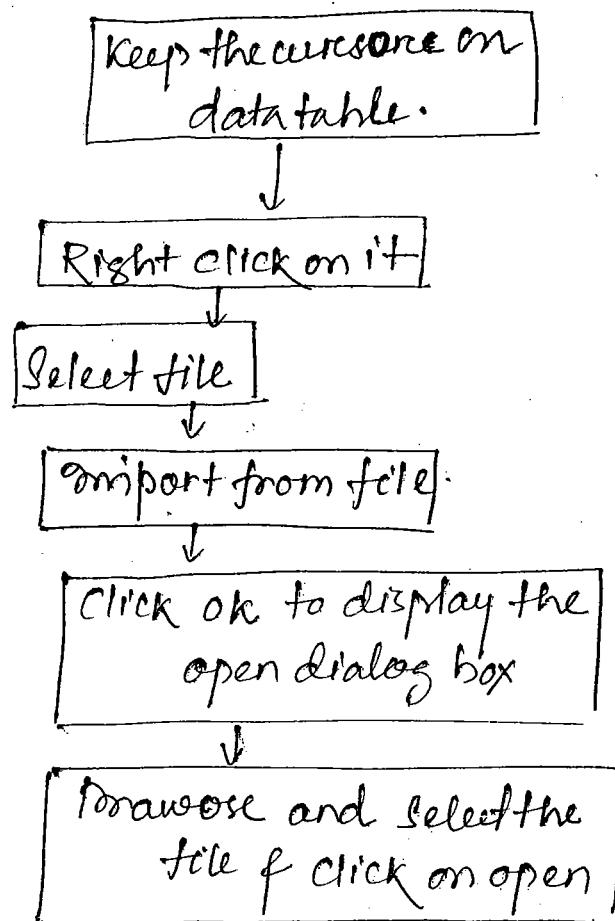
[Repeat this step for creating several columns.]



↓
Enter the data manually under each column.

- To import the data from other resources :-

Navigation:-



- To view Debug viewer:-

Goto view
click on Debug viewer

Chapter-2

Q How to make the QTP to repeat the operation and to carry out testing on an application:-

Q Recording:-

Q 1. What is recording?

It is the process in which tool is brought into an observation mode so that the tool can generate the script for the activities.

Recording session is the time span between the start recording and the stop recording.

Q 2. Operational overview of recording?

a. A tool is brought to observation mode.

b. The outcome of recordings are programmatic, non-programmatic, statements and active screen.

c. Since learning happens while recording object repository file can be created simultaneously apart from programmatic, non programmatic statements and active screen.

D Navigation :- (To record login activity)

Open the Login Screen
Keep it aside.

Click on record icon.

Click on OK on Record and Run setting window.

Perform operation one by one under Login dialog box.

Observed the QTP script generated



Click on stop button.

Example:-

Dialog("Login").Activate.



<Type of object>(<L-Name of the object>).
Object
child("Child Name").

(or) Dialog("Login").winEdit("Agent Name")

Formula:-

<Type of object>("<L-Name of the object>").
Object
child("Child Name").

<Method>"value"

Note:-

1. In window base application there are two fundamental parents
 - (a) window & dialog box.
2. Usually objects are presiding with technologies

Ex:- WinEdit—

Java Edit.

Web button.

Ques-1.

Write a QTP script to automate registration activity in the registration window by entering name, age and qualification as given in the following window that developed in visual basic/viaccess.

Registration

Name:	<input type="text"/>
Age:	<input type="text"/>
Qualification:	<input type="text"/>
<input type="button" value="Submit"/>	<input type="button" value="Cancel"/>

Ans:-

Window ("Registration"). Active.

Window ("Registration"). VB Edit ("Name"). Set "Abhilash".

Window ("Registration"). VB Edit ("Age"). Set "24".

Window ("Registration"). VB Edit ("Qualification"). Set "B.Tech".

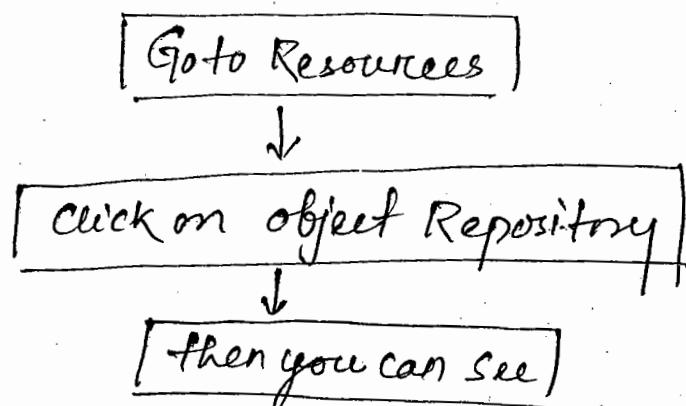
Window ("Registration"). VB Button ("submit"). Click.

Note:-

Logical names are case sensitive.

Value must be kept on double code unless it is variable

Navigatiom:- To view the object repository.



Note: Object repository contains only the information of the objects that are participated in the recording.

Recording Modes:-

Depends on types of objects and type of operation
there are basically 3 recording modes.

1. Normal recording Modes:-

In case operation that are performed
on the standard objects that to be recorded this
recording mode is used.

Ex:- clicking the button, entering value in text box
etc.

2. Analog recording Modes:-

In case continuous ~~rec~~ operations are
to be record this recording mode is used.

Ex:- Drawing, signature.

3. Low level recording mode:-

In case operation performed on non-
standard objects are to be recorded this recorded
mode is used.

Ex:- clicking the button whose technology is
not compatible with QTP.

[Note: Normal recording mode is a default -]
recording mode

Navigat'm:-

1. Open the paint window and minimize it.
2. Click on Record button.
3. Click on ok.
4. Click on automation.
5. Click on analog recording option.
6. Select the radio button option based on the desktop windows.
7. Click on Start Analog (Record button).
8. Activate the paint window.

Navigat'm :- Low level recording :- Dt. 23/03/2015

Navigat'm:-

open the window where in non-standard objects are presents.

Click on record button.

Click ok in record recent me setting window

Click on automation.

click on low level recording

Record the operations

click on stop recording

absorbed the low level recorded script.

Note:- Position of the operations that are recorded will be reflected in the script in terms of 'x' co-ordinate & 'y' co-ordinate.

Note:- A single script file can have several types of scripts recorded with normal, Analog and Low level recording modes.

It is always ensured that most of the recorded script is done with normal recordings.

Invoking of the application window:-

1. Application can be invoked manually.

Record and reentime settings window:-

The purpose of the window is to make the settings to indicate that the application has to be invoked manually or with the tool.

There is a provision on this window to make the settings for web application as well as window based application while recording as well as running of the tool.

By the tool tree ~~tool~~ settings.

Navigat'm:-

Go to automation
click on it

Select record for run
Setting option & click on
it.

Click on windows application's tab
In case window based application

Select the radio button option
depends on how the application
is to be invoked.

In case 2nd option provide path of the
ANT with help of (+) icon.

Click OK

Note:-

In case web application and radio button is to be
selected in the web tab and enter the URL and
name of the browser as inputs.

Invoking application window programmatically:-

Without depending on tool settings or invoking manually one can invoke the application window with the help of following features programmatically.

- ① Using the function :- "InvokeApplication" ^{→ No space} as given below.

Syntax:-

[Φ - space]

Invoke application Φ "Path of Application"

- ② Using "SystemUtil" object along with "Run" methods:-

Syntax:-

SystemUtil.Run Φ "Path of application"

Note:-

Mostly Invoke application is used for window based application. whence as ~~SystemUtil~~

SystemUtil.Run can be used for both ~~as~~ window based as well as web based application.

Note:-

When ever the QTP Script is written manually it is always the best practice that the corresponding object information is loaded on object repository so that the object information & can be supply automatically without mentioning the parent.

Q How to encrypt the "Password" while executing the script
manually:-

Q Navigation:-

Write ~~the~~ set secure method for
password in `edit`.

↓
Go to start → all programs

↓
HP QTP

↓
Select Tool

↓
Select Password encoder and
click on it.

↓
Enter the password and
click on generate button in the
Password encoder dialog box

↓
Click on copy button

↓
Click on close

↓
Keep the cursor after set secure
method with a single space and
paste it there.

Note:-

When the script file is executed it will be successful
provided the corresponding object information is
present in the object repository.

Chapter-3

Object Repository :-

Overview:-

- It is a container in which capture object information is stored and maintained.
- It acts like an interface between the script and the captured information.
- Captured information can be classified into two types
 - 1) Logical names
 - 2) Physical Properties
- Logical name acts like an interface between the Script and physical properties.

Creation of Object repository:-

- Object repository can be created in the following ways.

1) Automatic Creation:-

Since learning happens while recording object repository file can be created automatically as and when the recording happens.

2) Explicit creation:-

Object repository file can also be created separately / explicitly through manual method when ever it is required.

Navigaton:-

open the required window



Goto resources and click on object repository.



Goto object and click on

"add objects to local..."



keep the created hand icon on the required object and click on it.



click ok on confirmation window.

Note: If the object is added for the 1st time, it will be always added along with the parents

Adding all the objects at a time: —

go to Object & Select

"add object to local..."



keep the hand icon on windows/dialogbox and click on it



click ok on the confirmation window.

then filter window come.



Select all object type ~~object~~ and click on ok.

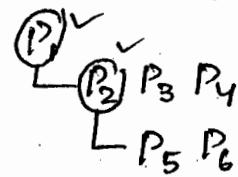
Gray colour : ref

Green colour : text box

Filter window option:-

① Selected object only:-

Used for selecting the parent objects but not with the child objects.



② Default object type:-

Objects that are associated with the operational scope are added or capture.

③ All object types:-

All the object which include Parents, Children, static and default objects.

④ Selected object type:-

Using this feature set of specific family objects like only buttons or only text boxes etc are captured.

Types of Repository files:-

Depend on how the object information is stored and used there are 2 types of repository as described below.

Per action Repository file:-

When "m" number of script files are created, "m" number of object repository files are created. These repositories are confined to the respective script file. Hence they are local repositories. Such kind of repositories are known as "per action Repository". It is a local repository.

Disadvantages :-

Whenever the common object information is modified, maintenance / updating the corresponding information of the object present in multiple repositories is rather difficult.

2. Shared Repository files:-

When in number of script file are created there will be only one object repository file that is created to store the entire object information which can be shared by the multiple script file.

It is Global repository .

Advantages:-

Maintain / updating the common object information is rather easy.

Creation and usage of shared repository file:-

Navigaton:-

Go to resources



Select Repository Manager



Go to object & click on add objects



Click on the window & click ok on the confirmation window .

All object select on the filter & click ok

Save the file in the specific location .

While saving the name must not proceed with

Open the script / create the script
to which shared repository will be
attached.

↓
Go to resources

↓
click on associated Repository files.

↓
Click (+) to select the path of
Shared repository file.

↓
Move the script file name from the
available block to the associated block.

then click on OK

↓
Execute the file and check it
and to observe the association.

Note: When a shared repository file is associated,
a read only copy of object information is
reflected at the local repository and it will
be erased once the shared repository file is
desassociated.

Programmatic Association:-

Share repository file can be dynamically
associated with the following programmatic statement -

Repositories collection.add @"Path of SRF"

SRF - Shared Repository file.

Operations performed on object Repository:-

1. Adding a new object:-

- New object info can be added whenever a new object is actually added to the application window.

2. Deleting an object:-

One can delete the unwanted object (Select the object → right click on it → click on delete)

3. Renaming an object:-

Select go to the object → right click on it
→ click on Rename → enter new name

4. Highlighting the object in the AUT:-

Select the logical name in the object repository

↓
go to view.

↓
click on highlight in application.

↓
observed the corresponding object
is highlighted.

5. Locating the logical name in the OR for a specific object:-

Consider an object in AUT

↓

Go to view on object Repository window.

↓

Click on Locate in repository.

Keep the hand icon on the specific object
and click on.

Say OK on confirmation window.

Above the corresponding logical name is
highlighted.

Note:- Logical name changed in the OR can be reflected
in the script also but not vice versa.

Types of Objects:-

Depends on various factors the following types
of objects are classified.

Based on Compatiblity:-

1. Standard object:- compatible to RTP.
2. Non-standard obj:- not compatible to RTP.

Based on scope:-

1. Local objects :- Present in Peraction repository
& is available to current script file.
2. Global objects:- Present in Shared repository file.
Available to multiple script file.

Based on change:-

1. Static object:- non operational/non changing
objects during run time.
2. Dynamic objects:- Dynamically changed object
during run time.

Based on Location:-

1. Test objects:- object present in object repository.
2. Run time objects:- objects present in AUT during runtime.

Utility objects:-

1. Object with special purpose that can be used to implement automation for standard operation.
Ex:- system util object for invoking the application.

Date - 24/03/2015

Objects Spy:- (No capture/only show)

It is basically a tool provided by the QTP in order to display object information right there at the application.

When to use Object Spy:-

1. Whenever the test script is stuck due to identification issue, to sacrifice the information of the objects.
2. Whenever we developed the descriptive program to know the object information object spy is used.
3. Whenever the tester write the script manually to help in providing the syntax in term of properties and methods, object spy is used.

Navigation:-

open the login window
for example.

↓
Go to tools & click on "object spy"

keep it side.

↓
click on the hand icon.

Keep the hand icon on any specific object to see the corresponding information in the object spy window.

↓
click on the specific object in order to give properties and operation in the respective tabs of objects point.

When we click
it will go
frozen
See all
properties

Behaviour of QTP towards Object Information:-

The behaviour of QTP can be classified into two kinds of behaviour

- (i) Normal Behaviour
- (ii) Smart Behaviour.

Normal Behaviour can be understood in terms of normal learning and normal identification ; Smart behaviour can be understood in terms of smart learning and smart identification .

Normal Behaviour:

Normal learning

Normal Identification.

- ① Ensure that the "enable Smart Identification" check box option is not selected.
- ② QTP capture unique properties but not all the properties, to identify the object ~~uniquely~~ uniquely.

Note - ~~unique~~ properties are basically set by the test engineer and QTP capture the same as per settings; these properties can be customized.

- ① When the script is executed captured information present in "OR" is used for identifying the object ~~uniquely~~ uniquely.
- ② Scene identification is successful operation are emulated on the objects successfully.

- ③ QTP captures mandatory properties initially.
- ④ In case mandatory properties are not enough QTP captures assistive properties.
- ⑤ In case mandatory & assistive properties are not enough QTP capture ordinal identifier information to identify the object with out fail.

Note:-

① Odinal Identifier information can be maintain in the following type.

1. Location ID:- For each and every object there is a unique Location ID ranging from (0 to n).

2. Index:- Every object has unique index number ranging from (0 to n).

3. Loading time:- In the web application every object has unique loading time IDs ranging from (0 to n).

Mandatory properties:-

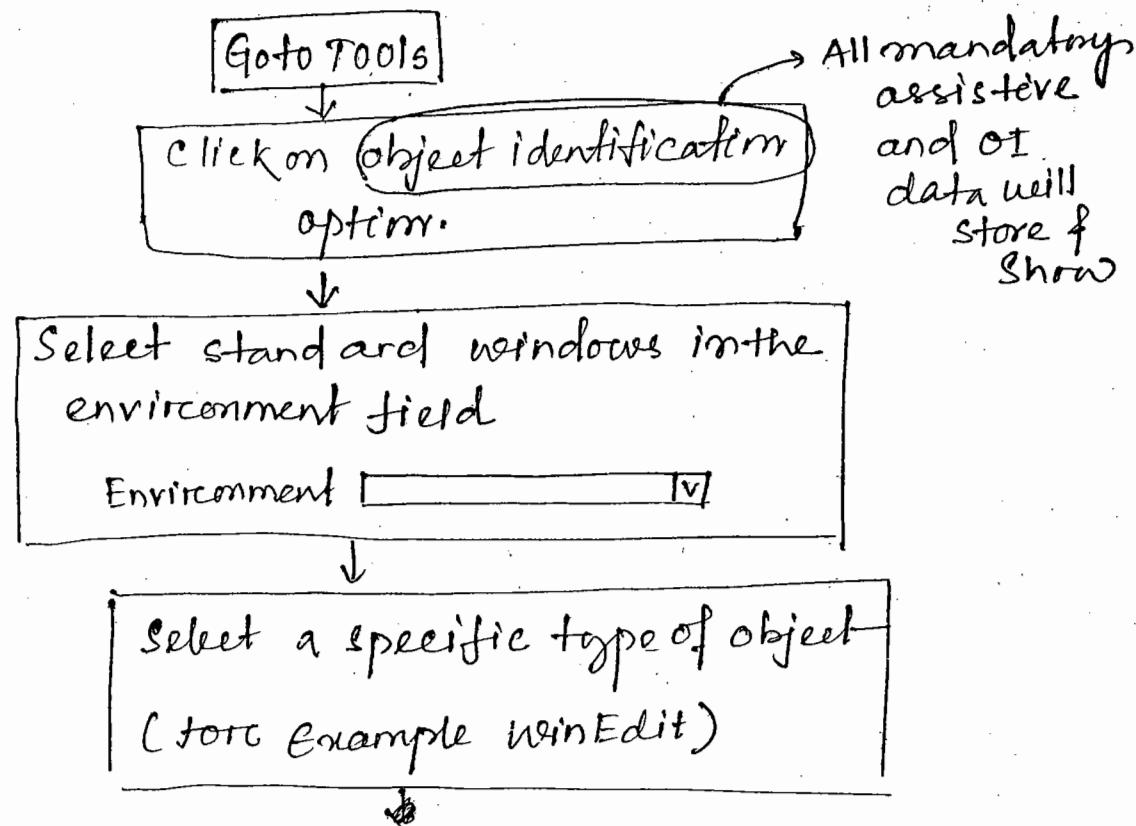
These are the properties that are capture compulsorily by the QTP.

Assitive properties:-

When mandatory properties are not sufficient these are the extra properties that can be added to mandatory properties to identifying the object uniquely.

⑥ All the captured information
is stored in object repository
(OR).

Normal Behaviour Navigation :-



① QTP depending on Mandatory :-

Goto object identification
window

↓
Set the following properties.

Mandatory Properties
attached text
native class

Assitive Properties
windows id

2) QTP depending on Mandatory + Assistive:-

Navigatiom:- (Box)

Mandatory Properties.
Native class.

Assistive Properties.
attached text

3) QTP depending on Mandatory + Assistive + Odinal Identifier:-

Mandatory Properties
Native class.

Assistive Properties
height

Odinal Location

Blocking the Odinal Identifier from usage:-

Go to object repository



Select the object (like Ajenname)



Select the odinal identifier information value.



Click on the button on right side.



Odinal Identifier window come

Select none against Identifier type.



Click on OK

Smart Behaviour of QTP :—

It can be understood in terms of Smart Learning and Smart Identification.

Smart Learning

1. "Enable Smart Identification" option must be selected / Selected.
2. Now allow to QTP, capture initially mandatory properties, assistive properties and ordinal identifier information depends on the requirement.
3. This capture information is stored in object repository.
4. QTP Captures set of base filter properties as well as set of optional ~~filter~~ properties as per the settings.

Base filter properties:-

These are the basic properties that are captured and used when OR information is not supported.

Smart Identification.

1. When the test script is executed QTP enters into object repository location and consider mandatory & assistive properties but not ordinal identifier and try to identify the object uniquely.
2. QTP leaves the OR and enter into smart location. Considering base filter properties. It try ~~to~~ to identify the objects uniquely.
3. In case base filter properties are not enough 1st optional filter property is added to base filter properties to identify the objects uniquely; same process is continue until the last properties.

Optional filters for properties:-

These are the set of properties captured and used when the bare filter properties are not supported.

5. The above captured information will not be stored in 'OR'. But it is kept under a smart location.

4. In case bare filter as well as option filter properties are not sufficient, QTP leaves the smart location and re-enters into OR to consider ordinal identifier information to uniquely identify the object.
5. In case ordinal identifier information is blocked QTP cannot identify the object and it will through the error message.

Smart Behaviour Navigation:-

Making the QTP depending on smart location that is base filter proposes.

Specification:-

Mandatory Prop. Native class.	Assisted Prop. height	Ordinal Iden. location	BTP Native class. height
		Optional filter properties attached text.	Smart identification.

Select enable smart identification



click on configure button



Enter Base filter properties and optional filter properties as per specification.

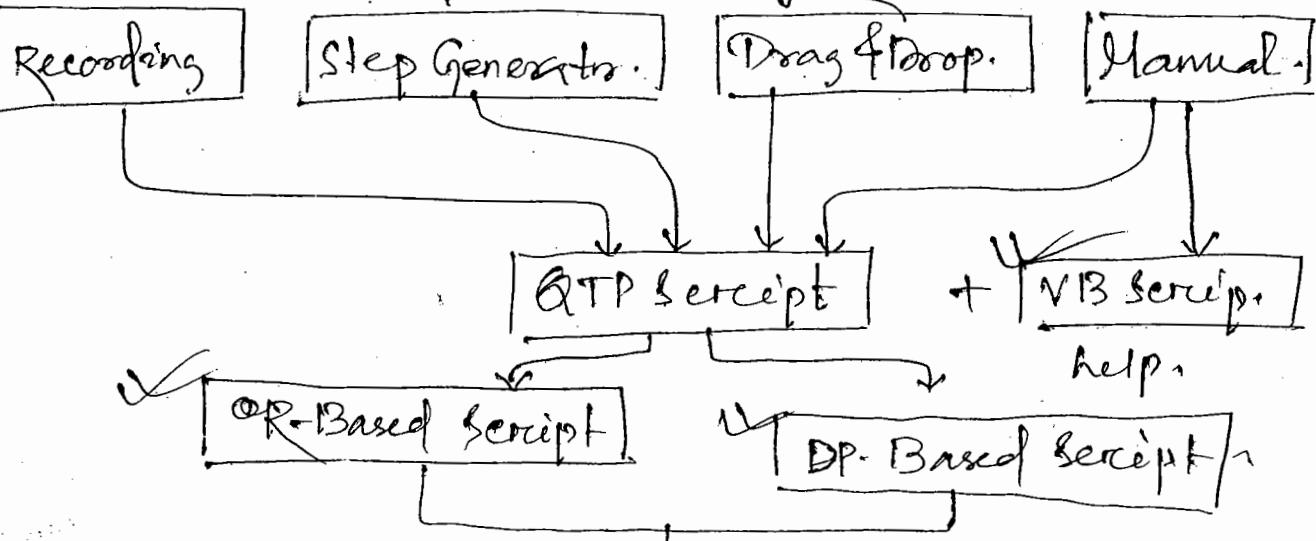


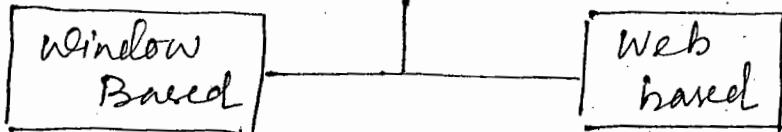
Note:- after executing the test script result windows can be open / test result window can be open. Expand the tree at left side observe the black hat for the object which indicates that they have been identify using smart identification; On selection of these objects smart identification process can be seen on the right side.

Date - 25/03/2014

Automation Test Methodology :-

Keyword driven.





- | | |
|--------------------|---------------------------------|
| 1) VB Script | 1) HTML |
| 2) .Net | 2) HTML + VB Script/Javascript. |
| 3) Java | 3) HTML + .Net. |
| 4) Siebal | 4) HTML + Java |
| 5) Standard window | 5) HTML + SAP. |
| 6) SAP | 6) HTML + Siebal |

Mainframe
apps + Web services + mobile application.
How to write

From the above diagram the industries follows the following automation test methodologies

1. Recording:- Using the recording feature Operational script is generated which can be further enhanced with variational statements with the help of check points.

2. Keyword driven methodology:-

This methodology includes the following features.

(i) Step Generator:- It is a tool used for generating the script statements with the help of object information loaded into object repository.

(ii) Drag and Drop Method:-

Just drag the object into the test pane script statement can be generated.

③ Manual Method:-

Manually test engineer can create the following types of script.

(i) VB Script:- Test engineer can make use of VB Script to generate the test script either by integrating this to neither the QTP Script or by providing Pre defined objects that can be used while testing.

(ii) QTP Script:- QTP scripts are manually generated in terms of two types of scripts.

(a) OR Based Script:- Test engineers can create test scripts manually that depend on "OR" for identification.

(b) DP Based Script:- Test engineers manually can create by keeping the object info in the script itself. Hence these scripts do not depend on "OR". This is the most ~~expensive~~ oftenly used method in the industry.

using the above methodology testing can be done on basically window based application as well as web based applications.

(a) Window Based Apps:- There are several window based application that are developed with the technology like standard windows, Visual Basic, .Net Java, SAP and Siebel etc. These can be tested with mostly DP Based QTP Scripts that are generated manually.

(b) Web based Apps:- There are several web based application that are developed with the technologies like HTML, HTML+VBscript/Javascript, HTML+.Net, HTML+Java, HTML+SAP, and HTML+Siebel etc. These application can be tested mostly with DP Based QTP script that are developed manually by the tester.

Exercise for Manual QTP Script:-

Q. Write QTP script for voting activity by entering the voter name in the voter text box, age of the voter and locality by clicking vote button in the Polling window.

Ans:-

Invoke app /c:/Demo/APP/poll.exe.
Window ("Polling").Activate.

Poll window

Votername	<input type="text"/>
age	<input type="text"/>
Locality	<input type="text"/>
<input type="button" value="vote"/>	

Window ("Polling") .win Edit ("Votername:") .Set "Abhilash".
Window ("Polling") .win edit ("age:") .Set "24".
Window ("Polling") .win edit ("Locality:") .Set " ".
Window ("Polling") .win button ("vote") .click.

Step Generator:-

It's a tool provided by QTP that can generate test scripts statements with the help of the object object information from the object repository.

Navigation:-

Load the object information of an application window into OR.

↓
Goto Insert & click on "step generator"

↓
Select utility object/functions to invoke the application in category field.

↓
Select the appropriate object / function. into object field.

↓
Select appropriate operation / method into operation field.

↓
Enter the specific values for the arguments (or give the path).

↓
In Generated step the path / Specific value is automatically generated

↓
Select the insert another step check box in case further steps are to be generated.

Drag & drop Method:-

Navigation:-

Open the object repository
in which the object information
is already stored.

Depends on the statements that needs
to be created the corresponding objects
are drag and drop from the "OR" to
the Test Pain.

Fill the values and enhance
with the required statements.

VB Script

Chapter - 4

→ about VB-Script.

1 → Variable?

2 → Declaring the variable.

3 → Assigning the value to variable.

4 → Data type.

5 → Option Explicit

6 → Message Box.

7 → Input Box.

8 → Operator.

9 → Expression.

10 → Condition.

11 → Arrays.

12 → Redim

13 → Pos

14 → Dim vs Redim.

15 → Control flow statements.

16 → Functions:-

(a) What is func'?

(b) Type of function.

(c) Study on SDF

(d) Type of UDF

(e) Creating UDF & Usage.

(f) Associating function to the script., ,

17) Sub procedure.

18) Difference b/w

Func' & Subprocedure.

About VB Script :-

- It is basically a scripting language that comes from Microsoft without any license.
- VB script can be executed successfully on windows platform due to "windows scripting host" that comes with windows operating system.
- It is basically used for validating the user inputs on the client side and also to make the window more effective.
- Unlike other conventional language it does not require specific editors and the program can be generated in any normal editors like notepad, edit + etc.
- VB script is based on interpretation rather than compilation.
- VB script is userfriendly scripting language so that the scripts can be developed quickly and easily.

Creation and Usage of VB Script:-

- Step-1 Create a new note pad.
- Step-2 Write VB script on it.
- Step-3 Save the file with ".vbs" extension.
- Step-4 Double click on the VB script file icon to execute the program.
- Step-5 Right click on the icon and click on the edit to modify the script.

Variable:-

1. It is basically a container used for storing the data.
2. When a variable is declared the space is allocated in the memory.
3. In VB scripting declaration of variable is not mandatory.
4. Value of the variable can be changed dynamically during run time.

Declaration Statement:-

"dim a"

Variable 'a' can be declared with the following statement.

Ex:- "dim a"

Multiple variable can be declared in the single declaration statement.

Ex:- "dim a,b,c"

~~nature to the
declaration of variable:-~~

Assignment of value to the variable:-

Values can be assign to any variable with the help of "equal to" operator as given below.

Ex:- $a=10$.

Datatypes:-

In VB scripting any variable that can take any data of any type at any time hence conceptually the datatype is "VARIANT".

5. Option Explicit :-

It is the key word usually mentioned as a 1st statement of the program that indicates all the variables that are used in the program must be declared compulsorily.

With Option Explicit

dim a,b

a=10

b=20

c=(a+b) —*

without option explicit

dim a,b

a=10

b=20

c = (a+b)

Note:- without declaring a variable program work fine but it is always a good practice to declare variables.

The main purpose of option explicit is to ensure the declaration of all variables and also to protect the values of the variable.

6. Message Box:-

It's a pop up window that can be displayed during execution in order to display the output of the program. Also it can display any string value given to it and contains of the variable given to it.

Ex:-

Msg Box ("Hello")



(String Value)
Given

dim x

x= ("Hello")

Msg Box (x)



(variable given)

Task:-

- Write a VB script to get the 1st name and last name in two variables and to stored the full name in a 3rd variable that can be displayed as output.
- Put of a program . Please ensure that all the variables are declared .

Sols:-

✓ Option Explicit

Dim dim a,b,c

a = "Abhilash"

b = "Dey"

c = a+b

* MsgBox (c)

✓ MsgBox O/p = Abhilash Dey O/p = Sridhar.

Option Explicit

dim a,b,c

a = "Sridhar"

b = "mindy"

c = a+b

Comment statement

MsgBox ("Sridhar")

Input Box:-

that can be display

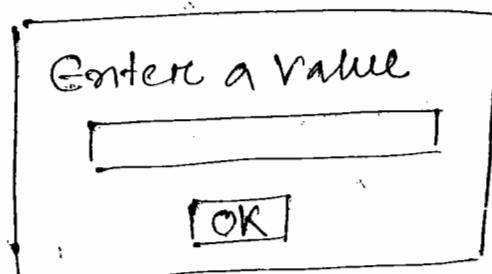
It is a dialog box, during reentime to take the input

which can be used in the program further.

Syntax

x = InputBox ("Enter a value")

Execution:-



Q) Write a program to get 1st name and last name dynamically and to display the same as an outcome of program

option Explicit.

Dim a, b, c

a = "Aman" inputbox ("Enter the 1st name")

b = "Kumar" inputbox ("Enter the last name")

c = a + b

msgbox(c)

Dynamically

Date - 26/03/2015

Task:-

Write a program to get the marks of ~~each~~ two subjects dynamically and to display the average of them.

option Explicit.

Dim a, b, c.

a = inputbox ("Enter the mark of 1st subject")

a = cint(a)

b = inputbox ("Enter the mark of 2nd subject")

b = cint(b)

c = $\frac{a+b}{2}$

msgbox(c)

When ever mathematical operation come in to pic. It good practice to convert name into integers and then make it a arithmetic operation. } with the help of function "cint()"

8) Operators:-

Operator is something that doing to operands together to perform a specific operation or to express specific relation.

Operators are of 4 types.

(a) Arithmetic operators:- This operator used for performing arithmetic operation like $+$, $-$, $*$, $/$, \mod .

Ex:- 1) $26/5 = 5.2$

2) $26/5 = 5$

3) $26 \mod 5 = 1$ (remainder)

(b) Relational operators:- used for expressing the relation between the two operators.

Ex:- $>$, $<$, \geq , \leq , $=!$.

(c) Logical operators:- These operators are used for commitment, omission and option.

Ex:- AND, NOT, OR

(d) Concatenation operators:- Used for joining two strings together.

Ex:- "&" " + "

(&)

a = Hello
b = Sridhar.
c = (a & b)

(+)

a = Hello
b = Sridhar
c = (a+b)

$$"10" + 20 = 30$$

$$10 + "20" = 30$$

$$"10" + "20" = 1020$$

$$10 + 20 = 30$$

Behaviour of '+' Operator:-

- a. When 2 operators are strings, "+" acts like concatenation operator.
- b. If any one of the operators has integer value it acts like arithmetic operation.

For Example :- "10" + 20 = 30

$$10 + "20" = 30$$

$$"10" + "20" = 1020$$

$$10 + 20 = 30$$

Note: When ever the inputs are given through input box it will be always consider as strings.

(9) Expression:-

It is the combination of an operator and the operands that can expressed either a specific operator or specific relation.

Ex:- $(x > 10)$, $(a+b+c)$
(relation) (operation)

(10) Condition:-

It is a criteria based on which a specific block of code can be executed or exempted from execution. In other words it is an expression that is preceded with "if" and followed by "then".

Ex:- if ($x >= 25$) then

11) Arrays:-

- a. It is the variable in which multiple values can be stored.
- b. Arrays can be created in two ways -
 - i) capacitive based declaration.
 - ii) value based declaration.

(i) Capacitive based declaration:-

while declaring the array variable capacitive is reflected in the declaration.

Ex:- `dim a(2), sum`

$$a(0) = 10$$

$$a(1) = 20$$

$$a(2) = 30$$

$$\text{sum} = a(0) + a(1) + a(2)$$

`Msgbox (sum)`

10	20	30
0	1	2

a(2)

Task!- WAP to get the name, qualification and percentage from each student dynamically to display the entire detail in a single record. as an outcome of program

Note - use capacitive based arrays for the above program.

~~Options Explicit~~

`dim a(2), c`

`a(0) = InputBox ("Enter the name")`

`a(1) = InputBox ("Enter the qualification")`

`a(2) = InputBox ("Enter the Percentage")`

`c = a(0) & a(1) & a(2)`

`MsgBox (c)`

(i) for kept space in the record box:-

```
c = a(0)&"fac1)&"fac2)  
Msgbox(c)
```

(ii) Value based declaration:-

In this type of declaration values are assigned while declaration itself. These value can be refer to with the help of corresponding array index value.

```
dim a
```

```
a = Array(10, 20, 30)
```

```
s = a(0) + a(1) + a(2)
```

```
Msgbox(s)
```

Note: Values need not be explicitly assigned as they are already assigned while declaration.

Task:- WAP for the previous task but by making use of value based declaration.

Ans:-

```
dim a, c
```

Ans:- ~~a = Array("name", "qualification", "Percentage")~~

```
c = a(0)&a(1)&a(2)
```

```
Msgbox(c)
```

~~a = Array InputBox("Enter name", "Enter qualification", "Enter Percentage")~~

Ans:-

Dim a, s, x, y, z

x = inputbox ("enter name")

y = inputbox ("enter Specification")

z = inputbox ("enter Percentage")

a = array (x, y, z)

s = a(0) & " " & a(1) & " " & a(2)

msgbox (s)

~~QUESTION~~

Types of arrays:- Depends on how the values are stored
In the array variables there are two types

1 - one dimensional (single dimensional) array

2 - Multi dimensional arrays

Ex:- a(2)

→ Ex:- a(2,2) this array can store values as
given below

a(0,0)

a(0,1)

a(0,2)

a(1,0)

a(1,1)

a(1,2)

a(2,0)

a(2,1)

a(2,2)

(12) Re-dim:-

It is a key word used for changing the capacity of array or variable dynamically during run time.

Ex:- `Re-dim a(2)`

$$a(0) = 10$$

$$a(1) = 20$$

$$a(2) = 30$$

^{Preserve}
`Re-dim, a(4)` —— (Resize dynamically)

$$a(3) = 40$$

$$a(4) = 50$$

$$S = a(0) + a(1) + a(2) + a(3) + a(4)$$

`MsgBox(S)`.

Note:- In case any array variable is to be enhanced dynamically, it must be declared as scalable/enhancable array variable using `redim`. If `dim` is used in the beginning it cannot be enhanced during run time as `dim` represent fixed/locked array variable.

Note:- When ever `Redim` is used the previously stored value get deleted.

Preserved Key Word:-

It is the keyword which is usually associated with `Re-dim` to preserve previously stored value whenever the capacity of a array is changed dynamically.

Correct program

ReDim a(2)

a(0)=10

a(1)=20

a(2)=30

ReDim Preserve a(4)

a(3)=40

a(4)=50

$$S = a(0) + a(1) + a(2) + a(3) + a(4)$$

MsgBox(S)

DIM vs REDIM:-

DIM

1) Dim is used for declaring fixed array variable.

2) Dim. is used for normal variable ~~variable~~ as well as array variable.

3) Using dim one can't resize of array variable dynamically.

RE DIM

1) It is used for declaring Scalable array variable.

2) Redim is used for declaring only array variable.

3) Re-dim is used for resize of capacity of array variable dynamically.

5) Control Flow Statements:-

CFS

Conditional Statement

- If ~~area~~(cond)
- If (Cond) then else
- If (Cond) then, elseif (Cond") then
else
- Select statement

Iterative

These statements are used for implementing logical flow of program. There are two types of control flow statement. (i) conditional statements. (ii) Iterative statements

(i) Conditional Statement:-

(a) If condition:

```
if <cond> then  
  {≡? B1}  
end if
```

(b) if <cond> then

```
if <cond> then  
  {≡? B1}  
else {≡? B2}  
endif
```

(c) if <cond> then

{≡? B₁}

else if <cond2> then

{≡? B₂}

else

{≡? B₃}

endif

(d) Selective Statement

Select Case <variable>

case "val1"

{≡? B₁}

case "val2"

{≡? B₂}

⋮

End Select

Exercise:-

Write the program to display status of Rasan Card that will be issued to individual. Depends on annual income $\leq 1L$ white rasan card

~~dimension~~

~~if i <= 100000 then~~

Ans:-

dim i

i = inputbox ("enter income")

if (i ≤ 100000) then

msgbox ("Eligible for white Rasan card")

Exercise:-

In continuation of the previous task write to display the Pink card eligibility for the individual income more than 1LK. per annum.

Ans:-

dim i

i = inputbox ("enter income")

if (i > 100000) then

msgbox ("Eligible for white Rasan card")

else

msgbox ("Eligible for Pink Rasan card")

Exercise:-

Write a program to display the result status of the student depends on the mark he obtained as per the following specification.

(1) $\geq 35 \rightarrow$ Pass

(2) $\geq 60 \rightarrow$ 1st class

(3) $\geq 50 \rightarrow$ S. class

(1) $\geq 70 \rightarrow$ Distinction

Ans:-

X Dim M
m = inputbox ("enter mark")
if cm >= 35 then
msgbox ("Pass")
else if cm >= 60 then
msgbox ("1st class")
else if cm >= 50 then
msgbox ("2nd class")
else if cm >= 70 then
msgbox ("distinction")
else
msgbox ("fail")

O/P

always same

Ans:-

✓ Dim M
m = inputbox ("enter mark")
if cm >= 70 then
msgbox ("distinction")
else if cm >= 60 then
msgbox ("first class")
else if cm >= 50 then
msgbox ("second class")
else if cm >= 35 then
msgbox ("Pass")
else
msgbox ("fail")

Always maintain
the order.

4. Exercise:-

Write a program to display the corresponding days for the corresponding numbers given to the program from 1 to 7 where 1 for Sunday
7 for Saturday.

Ans:-

Dim n

n = input box ("enter numbers from 1 to 7")

Select case n

case "1"

msg box ("display Sunday")

case "2"

msg box ("Monday")

case "3"

msg box ("Tuesday")

case "4"

msg box ("Wednesday")

case "5"

msg box ("Thursday")

case "6"

msg box ("Friday")

case "7"

msg box ("Saturday")

end case

(II) Iterative Statements

(a) do - while loop

do
{ }
while

Loop while (<condition>)

(II) While Loop:-

```
while <condition>
  {
    ==
  }
while end.
```

(III) For Loop:-

```
for i=1 to 3
  {
    ==
  }
Next
```

Date - 27.03.2015

* Write a program to display 1, 2, 3, 4 one after another.

(i) Using While Loop.

```
Dim i
i = 1
while (i ≤ 4)
  msg box (i)
  i = i + 1
while end.
```

Op = 1, 2, 3, 4

```
(i) Dim i
i = 1
while (i ≤ 4)
  i = i + 1
  msg box (i)
while end.
Op = 1, 2, 3, 4.
```

(ii) Using Do-While Loop.

```
Dim i
i = 1
do
  msg box (i)
loop while (i ≤ 4)
```

Op = 1, 2, 3, 4.

(iii) WAP to display 1, 2, 3, 4 one after another by using while loop.

```
Dim i
i = 1
while (i ≤ 4)
  i = i + 1
  msg box (i)
while end.
```

Op = 1, 2, 3, 4.

By using for loop:-

Dim i

For i=1 to 4

MsgBox ci)

Next.

Note:- In the case of for loop assignment of the variable (starting and ending counters) are done automatically, which include in the syntax; Also incrementing the counters (variable values) will be automatically done. Hence no need of writing incrementing logic.

Exercise :-

Write a program to store the name, age and qualification in an array variable dynamically and display it in terms of single record.

Note: Use for loop for storing the values.

*

Dim a, x, y, z, s.

x = InputBox ("enter name")

y = InputBox ("enter age")

z = InputBox ("enter qualification")

for

a = array (x, y, z)

{ dim i, a(2), r

For i=0 to 2

a(i) = InputBox ("enter information")

Next

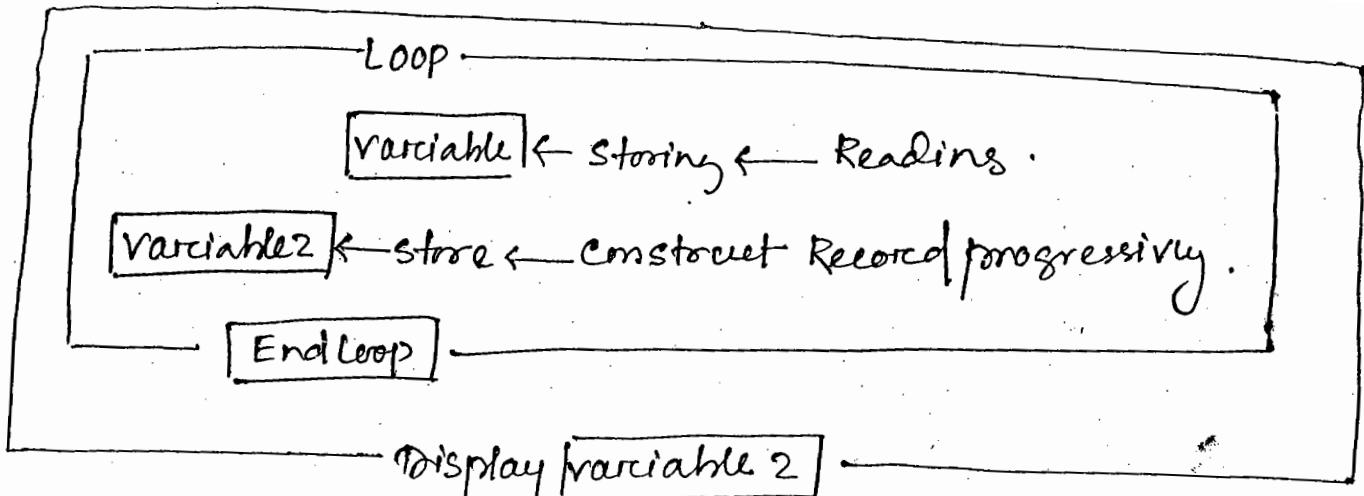
MsgBox xx

r = a(0) + a(1) + a(2) → If you want to display name age & reg.

MsgBox (r)

then:

r = "name" + " " + a(0) + " " + "age" + " " + a(1) +



(ore)

```

dim i, a(2), r
For i = 0 to 2
    a(i) = inputbox ("enter info")
    r = r + a(i)
Next
Msg box(r)

```

Note:- One can join all the elements present in an array variable with a specific delimiter and can display it with the help of join function as describe below.

```

dim i, a(2), r.
For i = 0 to 2
    a(i) = inputbox ("enter info")
Next
r = join(a, " ")
msgbox(r)

```

16) Function:-

1. It is a block of code that can be reused whenever it is required.

Type of Function:-

Depends on how they are created and used there are two type of function

1. System defined function / Predefined function.
2. User defined function.

1. System defined functions:-

These are the pre defined function by the system which can be directly used by the program.

Ex. Message Box.

Input Box.

Join () etc.

2. User defined functions:- These are the function that are created/defined by the user (automated tester) whenever it is required and use them.

Ex:- add(), login() etc.

Type of user defined function:- Depends on the scope there are two types (I) Local function & (II) Global function.

(I) Local function:- These functions are used by the only the current script file.

(II) Global function:- These functions are used by multiple ~~the~~ script file.

2.1 Creation & Usage of user defined function:-

Creation:-

Function <Function name> (Parameters)
 {
 Body of function
 }
Function = O/p (Function)
End function.

Ex:- Add function:-

Function & add (a,b)
 c = a+b
 add = c
End function

Usage of the function:-

Area Calling:-

Called area → {
 dim x, y, z
 x = 10
 y = 20
 z = add (x, y)
 Msgbox (z)
} Function add (a, b)
 c = a + b
 add = c
End function

Show Message box - After Run + F11

Q. Write a program that can display sum of two numbers by calling the function SUM. While SUM is already defined.

Ans:- function sum(a,b)

$$c = a + b$$

$$\text{Sum} = c$$

End function.

Dim x, y, z

$$x = 10$$

$$y = 20$$

$$z = \text{sum}(x, y)$$

msgbox(z)

Calling the function using call statement:-

One can call the function as described below if the return value is not required for further usage.

Syntax: Call Function name (value passed)

Ex:- call add(10,20)

Then above program will be.

me. paci ↑
→ function sum(a,b)
→ $c = a + b$
→ $\text{Sum} = c$
→ msgbox(c)
→ End function.
→ Dim x,y
→ $x = 10$
→ $y = 20$
→ Call sum(x,y)

2.2 Creation of Global functions:-

Global functions can be created in the following areas

1. Function library :- It is the storage place where in functions are created and maintained to be global.
2. External files :- Global functions also can be created in any external file like note pads independently without using QTP resources.

Navigation for function library creation:-

Go to new icon, click on the button.

New

↓
click on the function library option

↓
Create a function

↓
Save the function library
in the desired location. ~~File~~
format / extension.

• qfl - extension of function library file

Note:- function library file has a extension ".qfl"

Usage of function library file:-

Come back to the main
text

Create calling program
where in the function is called

Associate global function to the main text with the following validation/Alarization:-

Step 1 - Go to file & click on setting.

Step 2 - Select the resources.

Step 3 - Click on (+) icon to browse, select the Path of function library file.

Step 4 - Click on Apply and OK

2. External files :-

Step-1 Open any external file like notepad

Step-2 Create the function init

Step-3 Save the file in extantion .VBS

* Associate the function dynamically with the help of following system.

execute file <Path of .vbsfile>

Association statement
Invoking statement
always begin

Step-4. Write the association statement at the beginning.

Example.

Note - execute file "C:\Documents and Settings\user\Desktop\sum.vbs".

```
Dim x,y,z  
x=10  
y=20  
z=Sum(x,y)  
msgbox(z)
```

Types of Association:-

Association of the functions can be done in two ways

(1) Through settings.

(2) Dynamically with the help of "execfile statement"

Passing the values to the parameters of the function:-

When ever the values are passed to the function they can always be passed in two ways.

1. By Ref:- If the argument is defined with "byref" modification of original value is allow.

2. By Val:- If the argument is defined with "byval" the modification of original value would n't be accepted and the original value remains the same.

Note: By default values are passed through by reference i.e. the modified original values are accepted.

Ex:-

Function modify C byref a, byref b)

```
a=a  
b=b+1  
end funcn.
```

dim x,y,z

x=10

y=20

call modify (x,y)

MsgBox y

On -

Function modify C byval a, byval b)

```
a=a  
b=b+1  
end funcn.
```

dim x,y,z

x=10
y=20

call modify (x,y)

MsgBox y

On -

Function modify (a,b)

a=a

b=b+1

end function

dim x,y,z

x=10

y=20

call modify(x,y)

Msgbox (y)

O/p-21.

Subprocedure :-

It is also a piece of code that can be reused by calling subprocedure & from the main text

Subprocedure can be create & used as given

below:-

Syntax

Login Subprocedure:

creation:

```
Sub Login (UN, PW) ——————  
    dialog ("Login"). active.  
    dialog ("Login"). winedit ("Af.Name") .set UN  
    dialog ("Login"). winedit ("Password") .set PW  
    dialog ("Login"). winbutton ("OK") .click  
end sub. ——————
```

Usage:

Login ("Sridhar", "mercury")

Login ("sita", "mercury") .

Difference between function & sub procedure.

functions

Subprocedure

- | functions | Subprocedure |
|--|--|
| (1) functions return a value. | (1) Subprocedure will not return value. |
| (2) Result is assign to the function name. | (2) Result will not be assign in subprocedure name. |
| (3) function can't be execute faster then subprocedure because it have extra value | (3) Subprocedure is executed faster. |
| (4) Function have logical advantages due to return value. | (4) whereas subprocedure does not have logical advantage as it does not have the return value. |

Note:-

While using message box function the following parameters are used.

1. Message to be displayed.
2. Code for buttons.
 - 0 → OK button.
 - 1 → OK & Cancel Buttons.
 - 2 → Yes & No Buttons.
 - 3 → Yes, No & Cancel Buttons.
3. Title of the message box.

Date - 30.03.2015

Recreational Program :-

System Defined Function:-

- 1) len() :- To count the number of characters of strings
Ex:- Write a program to find the length of your name.

Dim k.

$k = \text{len}("Abhilash")$ O/P $\rightarrow 8$

Msg box(k)

- 2) Trim() :-

- To trim off / eliminating the spaces in strings

- 3) LTrim() :- To eliminate left space.

- 4) RTrim() :- To eliminate Right space.

Ex:- Dim k.

$k = \text{LTrim}(" Kohli")$ O/P \rightarrow name: Kohli

Msg box ("name: " & k)

Note:- 'Trim' eliminates lefts as well as Right space.

- 5) Left() :- To get the left part of the strings.

Right() :- To get the right part of the strings.

mid() :- To get the middle part of the strings.

→ WAP to get the 1st 3 char of the string "Abhilash".

Dim k

$k = \text{left}("Abhilash", 3)$ O/P \rightarrow Vir.

Msg box(k)

→ WAP to get the "ll" from "Abhilash".

Dim k

$k = \text{mid}("Abhilash", 4, 2)$ O/P : id.

Msg box(k)

where Abhilash is string

U is string starting position of required
2. no. of char. that we required.

6) LCase() :- To convert into lower case

7) UCase() :- To convert into upper case.

Ex:- Dim K

K = UCase ("abhilash") O/p → ABHILASH.

MsgBox (K)

10) CInt() :- To convert into integer.

CStr() :- To convert into string.

CDbl() :- To convert into Decimal

Round() :- To Round off Nearest Integer

Ex:- Dim X

X = 40.6

O/p → 40.

X = CInt(X)

MsgBox (X)

Dim K, S

K = 40.6

K = Round (K)

S = 39.7

S = Round (S)

MsgBox (S)

O/p = 40

Dim X

K = 40.6

O/p → 40.6 (This string.)

K = CStr (K)

MsgBox (K)

11) Point() :- To display o/p during runtime as well as Post
run time in the QTP external file.

Dim S, K

S = 5.8

K = 40.3

S = Round (S)

K = Round (K)

Pointf ("Value of S is: "%s")

Pointf ("Value of K is: "%d")

O/p = Value of S is : 6
Value of K is : 40.

15) Executefile():-

To associate the external Resource file dynamically.

16) Exit test():-

To come out of the execution of the program.

17) Wait():-

To delay the execution during run time for specified amount of time.

Ex:- $\text{wait}(6)$ → Delay for 6 sec.

18) Instr():-

To get the position of specific string in a given statement.

Q) WAP to know the position of 'You' in the following statement. "I am fine, how are you?"

Dim K

K = Instr("I am fine, How are you?", "you")

Msgbox(K)

O/p = 19

19) Array():- use for declaring array variable. (value based declaration)

20) Isarray():- To check if the variable is array variable and return a value true or false.

Ex:- Dim k

k = Array(40, 6, 406)

S = Isarray(k) O/p - True

Msgbox(S)

21) Isnumeric(): To check if it is the numeric value or return a value True or False.

22) Join() :- To join the strings/values present in a array variable with the specify delimiter.

Ex:- Dim S(1)

S(0) = "Abhilash"

S(1) = "Dey"

Z = Join (S, "*")

MsgBox (Z)

O/p: Abhilash Dey.

23) StrReverse() :- To reverse the string.

Ex: Dim K

K = "Abhilash"

O/p:- healihba

S = StrReverse (K)

MsgBox (K)

24) Eval() :- to calculate long mathematical expression.

Ex: Dim S, K

S = Eval ((20 - 5) / 3 * 2) O/p - 10

MsgBox (S)

25) Invoke application() :- to invoke the application.

Invoke application "C:/sample/Demo/ File : x.exe"

26) Date() :- To display the system date.

27) Time() :- To display the system time.

28) Now() :- To display the time startup.

Dim X, Y, Z

X = Date() O/p : 03/31/15

Y = Time() 12:50:19 am.

Z = Now() 03/31/2015 12:50:20

MsgBox (X)

MsgBox (Y)

MsgBox (Z)

NOTE!- Mostly "Now" function is used while creating event Logs in the projects.

29) Create object() :- To create a specific object.

Ex:- WAP to create xl object.

Dim xl

Set xl = CreateObject ("Excel.Application") .

Xl.

30) Isobject():- To check if it is object or not.

Ex:- WAP to check if it is object and display "yes it is object" in case it is really an object. and display "not an object" if it is not really an object.

Dim x1

Set x1= CreateObject("Excel.Application")

x = IsObject(x1)

If x = "True". then

MsgBox ("It is an object")

else

MsgBox ("It is not an object")

End If.

[Note: True or False are case sensitive.]

31) Strcomp():- To compare two strings.

Return values:-

0 → both are equal.

1 → 1st > 2nd

-1 → 1st < 2nd.

WAP to print the full name in such a way that greater name is proceeded.

Dim S1, S2, K

S1 = "Abhilash"

S2 = "Baburam"

K = StrComp(S1, S2)

If (K=0) then

S = S1 + " " + S2

MsgBox ("Both are equal : " & S)

else If (K=1) then

S = S1 + " " + S2

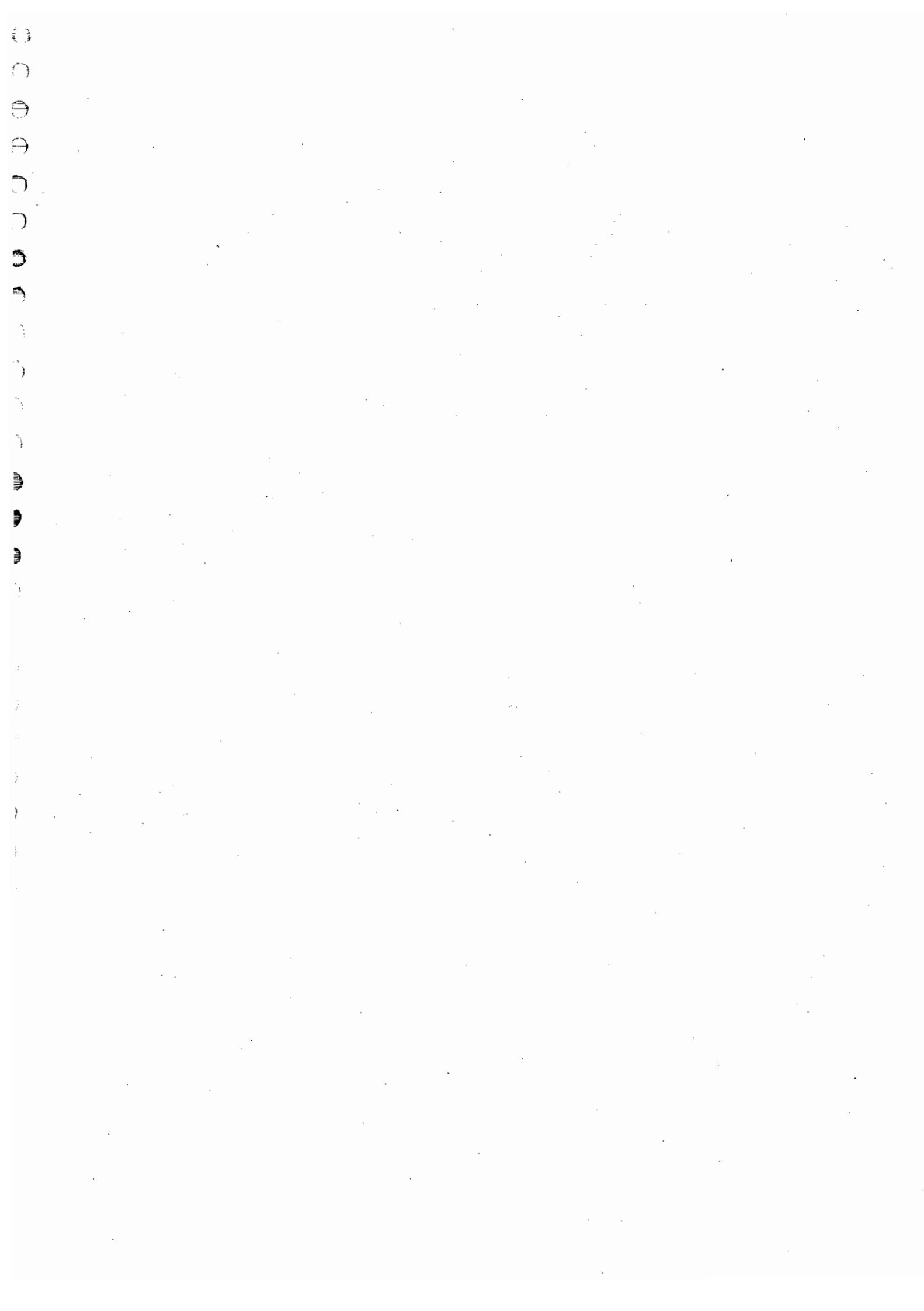
MsgBox (S)

else ~~If~~ S = S2 + " " + S1

MsgBox (S)

O/p: Abhilash Baburam

neither.



Descriptive Programming:-

Date:- 30.03.2015

It is the way of programming in which the object information is directly placed in the script itself rather than object repository, so as to increase the test performance.

Advantage/Purpose/why Descriptive programming:-

1. Using descriptive program one need not depend on object repository(OR).
2. Using descriptive programming test performance is increased.
3. Descriptive program can identify the dynamically changes objects during run time.
4. Once the descriptive program frame work is created same can be used for the future projects with minimum customization.
5. Descriptive program can be executed to carry out testing on the remote AUT.
6. In a single script file both dP based QTP script as well as OR based QTP script can be kept and maintained.

* When the descriptive Program is used ?

- When ever there is enough time for testing. It is always preferred to go for creation of descriptive programming framework.
- When there is no sufficient time it is better to go for "OR" based Scripting

* How to create descriptive program ?

Descriptive program can be created in two ways

- (i) Static Method (Object Description Method)
- (ii) Dynamic Methods (Descriptive object Method)

(i) Static Method:-

This is the method of creating the descriptive program in which list of unique properties are kept in the place of logical names (Usually used in the OR based script)

OR Based Script Statement :-

dialog ("Login").winButton ("ok").click

The above statement is OR based Statement in which Logical names played a key role in identifying the objects.

For the above activity VB based statement can be created as follows .

Syntax:-

Object type ("Property₁ := value₁", "Property₂ := value₂"...n)

Descriptive Program

Properties table:-

Properties table.

Sl.N.	Objects	unique Properties	values
1	Dialog box	text	Login
2	Push button	native class text	Button OK

Then:- DP-Based statement :-

dialog("text:= Login"). winbutton ("native class:= Button",
"text:= OK"). click.

Creation of object table:-

Sl.No	Objects	unique Properties	values
1			
2			

* Task:-

Write a descriptive program to create a DP based QTP
script to automate login activities.

Sl.No	Objects	unique Properties	values
1	Dialog Box	Text	Login
2	Agentname text box	native class attached text	Edit AgentName:
3	Password text box	native class attached text	Edit Password:
4	Push button	native class text	Button OK

10. The Identified objects in the login activities are

1. Dialog Box.
2. Agentname text Box.
3. Password text Box.
4. OK Button.

2. Corresponding unique properties along with the value are kept in the following table.

Sno	Objects	unique Properties	values
1.	Dialog Box	Text	Login.
2.	Agentname text box	Native class attached text	Edit Agent Name:
3.	Password text box	Native class. attached text	Edit Password:
4.	Push button	Native class. text	Button OK

Statement:-

dialog ("Login")

dialog ("text:=Login"). Activate.

dialog ("text:=Login"). win^{edit} ("Native class:=edit", "attached text:=Agent Name:"). Set "Abhilash".

dialog ("text:=Login"). win edit ("Native class:=edit", "attached text:=Password:"). Set "Abhi".

dialog ("text:=Login"). win button ("Native class:=Button", "text:=OK"). click.

(II) Dynamic Method: (Descriptive Object Method)

This is another method of creating descriptive program in which descriptive objects (which are created with the list of corresponding unique properties) are kept in the place of logical names.

There are two type of descriptive objects

(i) Local objects.

(ii) Global objects. that can be created based on the scope.

(i) Creation of Global descriptive objects:-

(a) Go to function library and create descriptive objects based on the following syntax.

Syntax:-

dim db_obj

set db_obj = description.create()

db_obj ("Property 1"). value = "value 1"

db_obj ("Property 2"). value = "value 2"

⋮
⋮

n properties.

Eg:- dialog("Login").winbutton("OK").click

DP Based Statement:-

dim OK_obj, db_obj.

Set OK_obj = description.create

OK_obj("NativeClass").value = "Button"

OK_obj("text").value = "OK"

Set db_obj = description.create

db_obj("text").value = "Login".

(b) Save the function library file in a desired location

(c) Go to the test pain and create the main text in which descriptive objects are used in the place of logical names.

Syntax:-

→ | Object type (descriptive object)

Ex:- dialog(db_obj).winbutton(OK_obj).click

Note:-

Descriptive object in main text should not be kept on " - " double cout. In case double cout it will be treated as logical name.

(d) Associate function library file to the main text.

(e) Execute the main text to carry out testing operation.

Task:-

Write a descriptive program using Global object dynamic method for the login activity.

Ans:- (creation of descriptive object of login.)

dim db_obj, an_obj, PW_obj, OK_obj

Set db_obj = descriptem.create

db_obj("text").value = "Login".

Set an_obj = descriptem.create.

an_obj("Native class").value = "~~Native~~ Edit".

an_obj("attached text").value = "Agentname:".

Set PW_obj = descriptem.create

PW_obj("Native class").value = "Edit".

PW_obj("attached text").value = "Password:".

Set OK_obj = descriptem.create

OK_obj("Native class").value = "Button"

OK_obj("text").value = "OK"

(+) Associate

dialog(db_obj).Activate.

dialog(db_obj).winEdit(an_obj).set "Abhilash"

dialog(db_obj).winEdit(PW_obj).set "Mercury"

dialog(db_obj).winButton(OK_obj).click.

creation of local descriptive Object :-

In case the descriptive objects are created in the same file where there is the main text then these objects become local.

Parameterisation of Descriptive program:-

```
Dim db_txt, tb_nc, an_txt, pw_txt, bt_nc, ok_txt, an_val, pw_val
```

```
db_txt = "Login"
```

```
tb_nc = "edit"
```

```
an_txt = "Agent Name"
```

```
pw_txt = "Password"
```

```
bt_nc = "button"
```

```
ok_txt = "Cancel"/"OK"
```

```
an_val = "Abhilash"
```

```
pw_val = "Mercury"
```

```
dialog ("text:=" & db_txt). activate.
```

```
dialog ("text:=" & db_txt). winedit
```

```
( "native class:=" & tb_nc, attached text:=" & an_txt) set
```

```
an_val.
```

```
dialog ("text:=" & db_txt). win edit ( "native class:=" & tb_nc,
```

```
"attached text:=" & pw_txt) set pw_val.
```

```
dialog ("text:=" & db_txt). winedit ( "native class:=" & bt_nc,
```

```
"text:=" & ok_txt). click.
```

Descriptive program can be parameterized to make it generic to be used by multiple values, hence without modifying the main text the same script can be used with several different values and operations.

parameterised

The above script is generic for login activities.

Usage of Descriptive Objects:- Purpose of DO

1. Descriptive objects can be used to make the descriptive program dynamically.
2. Descriptive objects ~~are used~~ to parameterized descriptive program.
3. Descriptive objects can also be used for the enumeration (counting) the objects and also to list out them.

Task:-

Write a program to enumerate/count all the child objects and display them family (mic class) wise from login dialog box.

Dim db_obj, ObjCollection, C, i, n

Set db_obj = description.create

db_obj("text").Value = "Login"

Set ObjCollection = dialog(db_obj).child objects.

C = ObjCollection.Count

msg box(C)

For i = 0 to C-1

n = ObjCollection.item(i).~~object~~

} for count

msg box(n)

Next

O/P - 8.

O/P - win edit - 2

win edit button - 3

static - 3

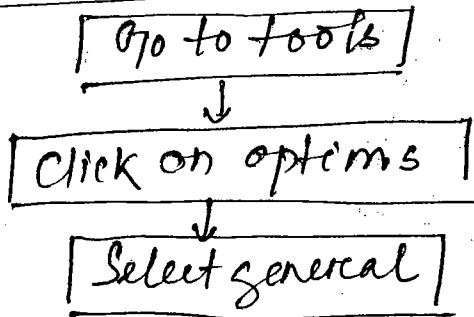
- ① It's always advisable to use "with format" while writing the test script manually.
- ② In order to write the script manually every test engineer must have the knowledge on various methods that are used for various object developed in various technologies executed in various environment.
- ③ Test engineer must be able to develop the script for testing window based applications with various technologies as well as web based application developed in various technology.

① With Format:-

It is a way of scripting in which the following advantages can be obtained.

- a. Eliminating the repetition of parents in the script.
- b. clarity of hierarchy in the script

Navigations of QTP with format setting:-



and select the option "Automatically generate with statement after recording" and also enter minimum number of

Click on Apply and OK

Program for Login Screen in with format Script:-

with Dialog("Login")

• Activate

• win edit ("Agent name:").set "Saidhar"

• win edit ("Agent Password:").set "Mercury"

• win button ("OK") click ..

End with.

Vendoring "With Format" :-

Navigation :- Go to Edit

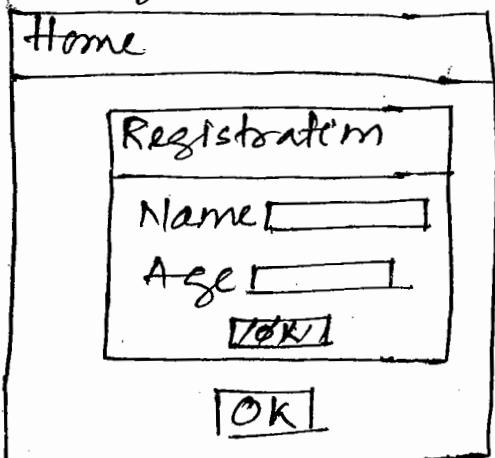
click on advanced



click on remove with Statement
option.

Task!-

Write a QTP based script to automate Registration Activity in a dialog box present in home window as given in the diagram.



P₁: Home

P₂: Reg dialog box.

Name TB NC AT

PW TB NC AT

OK button NC Edit Text

With end

With end .

Ans:-

```
with window ("text:= Home")
    • Activate.

    with . dialog ("text:= Registration")
        • Activate.

        • win edit ("Native class:= edit", "AttachedText:=
                    Name"). set "Abhilash"

        • win edit ("Native class:= edit", "AttachedText:=
                    Age"). set "23"

        • win Button ("Native class:= Button", "text:= OK").
            click

    End with.
```

End with.

From the above Script :-

1. Every parent will have the corresponding with block.
2. When writing the script nested with blocks are possible.
3. Parent with in the other parent must be preceded with dot (.).

Q2 Object - Method table.

No	Objectname	Method/function	Purpose.	Example.
1.	Textbox	Set	to set the value/enter the value into textbox	Dialog("Login").winEdit("Agent name:") <ul style="list-style-type: none"> • Set "Abhilash".
2.	checkbox	Set.	To select the check box Set takes ON/OFF ON → Select 'em. OFF → Deselect 'em.	Dialog("Login").winCkckbx("Normal user").set "ON"
3.	Radio button	Set	To select the radio button (No - ON/OFF for radiobutton)	Dialog("Login").winRadioButton("Admin user").set
4.	any object/ button.	click	To perform click operation.	Dialog("Login").winButton("OK").click.
5.	any object/ image	capture bitmap	To capture the snapshot of any object.	Dialog("Login").winImage("D:\sample\sample.bitmap").capture

6. window/ dialogbox/
any object.

to check if the specified
window/obj / dialog
box is existing.
* exist take time
Parameter that indicate
the program need
specified time for
checking the existence.

wait (t)

To delay time (T)
Seconds

wait (10) - It provided delay for 10 sec.
ex:- dialog ("text": "Success"). exists()
Wait (7)

wait property

delayed based on
specific property
the appears on the
program/steps.

7. window/
any object.

Dialog ("Login"). wait ~~button~~ Button ("OK").
wait property enable, 1, 10,000 msec
Properties value of
Property

8. any object

DBL click.
double click on
object.

Dialog ("Login"). wait button ("OK"). DBL click.

10	Combo box/ listbox	Select.	To select the item from combobox/listbox.	Windows ("form1").win combobox("Country"). •Select("India")
11.	Window/ dialogbox	Activate	To activate window/ dialogbox.	Windows ("form1").close .
12.	Window/ dialogbox	Close	To close window/ dialog box	Windows ("form1").close .
13.	Keyboard input	Type	To perform key board application operation on application under testing (AUT)	Windows ("form1").win edit ("value 1"). Type <u>mic tab</u> ↓ mic-s finds for <u>"mercury index code"</u>

<p>Q. Any run time object</p>	<p>to get the run time objects property value during execution</p>	<p>$x = \text{window}(\text{"Form1"}).writelnEdit(\text{"Result"})$</p> <ul style="list-style-type: none"> • Get-Ro Property C("Text")
<p>5. TestObjects</p>	<p>Get-To Property To set the test object property value during execution.</p>	<p>$x = \text{window}(\text{"Form1"}).writelnEdit(\text{"Result"})$</p> <ul style="list-style-type: none"> • Get-To Property C("Text")
<p>16. TestObject</p>	<p>Set-To Property To set the test object properties value temporarily during run time:</p>	<p>$\text{window}(\text{"Form1"}).writelnEdit(\text{"Result"})$</p> <ul style="list-style-type: none"> • SetToProperty "Text", 50 <p>Or</p> <p>Set the value to test object that present in</p> <p>Or</p>

Exercise:-

Write a program to select Food option and deselect drink option in the flight window.

Ans:- window ("text:=Flight").Activate

win checkbox ("Native class=edit", "text:=Food")
Set "ON".

win checkbox ("Native class=edit", "text:=drink")
Set "OFF".

Exercise:-

WAP to capture user name text box from the welcome window and store it in sample folder of C drive with filename XYZ.

Ans:- window ("text:=Welcome").win edit ("Native class=edit", "Attached text:=username"), capture bitmap
"C:\Sample\XYZ:bitmap".

Exercise:-

WAP to check the success dialog box whether it's existing with in 7 seconds.

Ans:- dialog ("text:=Success").exist (?)

Exercise:-

WAP to provided the delay based on the text property until it becomes stop with respect to the start button of Hello window:

Ans:-

window ("text:=Hello").winButton ("text:=start").Wait property
text, Stop, 10,000

! . Value of button ,

Write a program to select by case option from the Payment mode combobox present in paymentwindow.

Ans:-

```
window("text:=Payment").wincombobox("AttachedText  
:=case") select "100".
```

Exercise:-

WAP to enter the 2nd name skipping of firstname in the Sample window.

Ans:-

```
window("text:=Sample").winedit("AttachedText:=firstname")  
• type mic tab.
```

```
window("text:=Sample").winedit("AttachedText:=2ndname")  
• Set "Dey".
```

Exercise:-

WAP to capture X-co-od & Y-co-od of the Password textbox in the Login window.

dim(x,y)

```
x = window("text:=Login").winedit("AttachedText:=Password")  
• Get Ro Property ("x")
```

```
y = window("text:=Login").winedit("AttachedText:=Password")  
• Get Ro Property ("y")
```

Exercise:-

WAP to set the test property of a button as submit button temporarily in the login dialog box.

Ansi -

dialog ("Login").withButton ("OK").setROProperty ("text") "Submit".

Point to be remembered W.R.T DP Based scripting :-

Principle 1:-

In case the percent depends on OR child can be based
on "OR" as well as "dp" OR

Exi

98

१८

dialog("Login").button("OK").click(). → valid

```
dialog ("Login"), winbutton ("text='OK').click
```

OR

DP.

Postscript 2:-

In case parent depend on DP the child must also be depend on DP, but not OR.

Expt

D17

62

→ parallel:

```
dialog("text:=Login").newButton("OK").click.
```

```
dialog("text=Login").button("text=OK").click();
```

17

PP.

3

DP Based QTP Script for window based technologies:-

Test engineer can prepare test scripts to carry out automation testing on various applications developed in the technology like Standard windows, VB, .net, Java, SAP, Siebel etc.

A. Standard Windows technology:-

1. Introduction:- There are many stand alone desktop applications ~~that~~ that are developed with standard windows technology. These are to be tested with appropriate test scripts.

2. Environment:-

- Install QTP without any addin.
- Install application developed in standard windows.
- Launch QTP without selecting any "addin"
- Launch the AUT.

3. Object property table:-

Technology	Object-Hierarchy Reference - objects	Description Property
Standard windows	Windows ↓ Dialogs ↓ Objects	Window Dialog winbutton. win Radio button. wincheckbox.
		Text
		win edit wincombobox.
		Attached text
	Other objects.	window id (in case problem)

4. Task :-

①

STUDENT STUDENT	
Student ID	<input type="text"/>
Phone No	<input type="text"/>
Qualification	<input type="text"/>
Course Interested	<input type="text"/>
<input type="button" value="Submit"/>	

winwindow (Text)
 winbutton (Text)
 wineedit (AttachedText)

Q: Write a script dp based to automate submission task with the following data.

Student Data:-

1. ID: 123
2. Phone: 0402801243
3. Qualification: M.Tech.
4. Course: Testing tools.

Ans:-

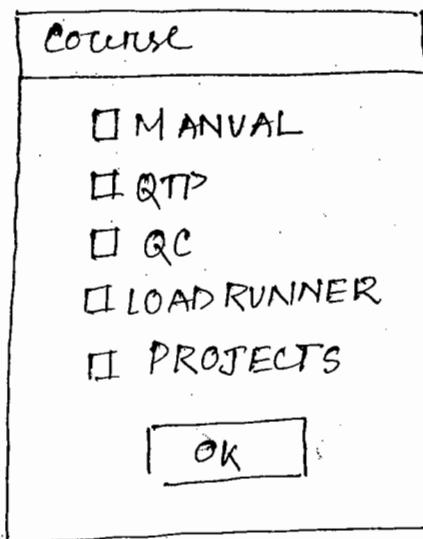
With winwindow ("text:= STUDENT")

- Activate
- wineedit ("AttachedText:= studentID"). set "123"
- wineedit ("AttachedText:= Phone No"). set "0402801243"
- wineedit ("AttachedText:= Qualification"). set "M.Tech"
- wineedit ("AttachedText:= courseInterested"). set "Testing tools"
- winbutton ("text:= Submit"). click
- close

End With.

Example-2

Task:- Develop the script to select the combo package of Course and later on with draw the Load Runner course.



With window ("text:= course")

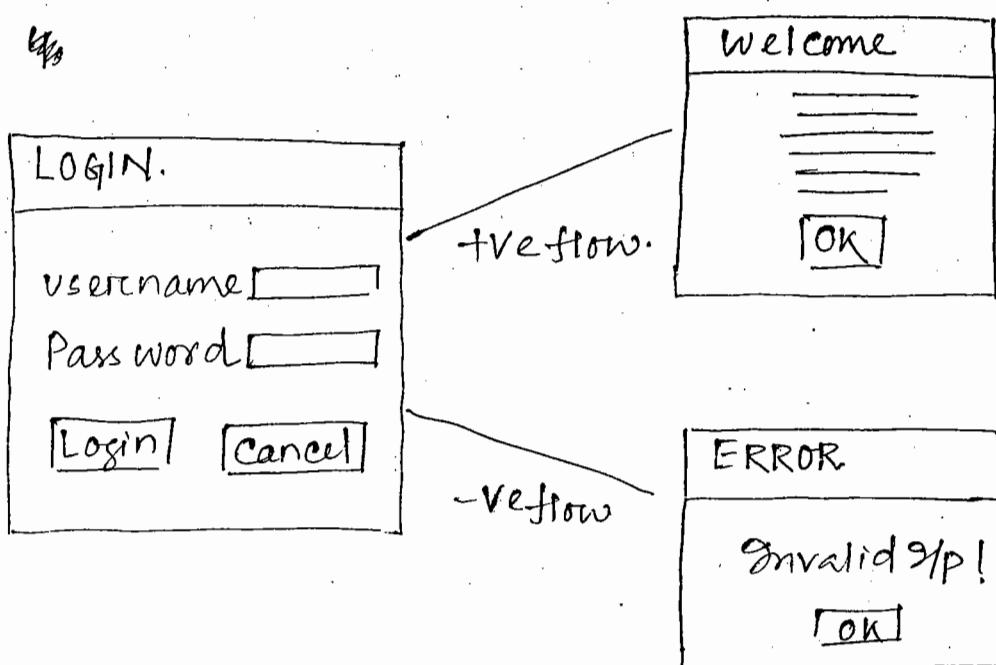
- Activate.
- Wincheckbox ("text:= MANUAL"). Set "ON"
- Wincheckbox ("text:= QTP"). Set "ON"
- Wincheckbox ("text:= QC"). Set "ON"
- Wincheckbox ("text:= LOADRUNNER"). Set "ON"
- Wincheckbox ("text:= PROJECT"). Set "ON"
- Wincheckbox ("text:= LOADRUNNER"). Set "OFF"
- Win ~~button~~ Button ("text:= OK"). click.
- Close

End with.

ask-3:-

1. Automate invoking of sample.exe application that resides in demo folder of D drive.
2. Automate login activity with following information
 - Valid Username Sunny
 - Valid Password SunnyPW

3. Write the script to check if the Login functionality is OK with the above information or with the above credentials.



Ans:-

Systemutil.Run "D:\Demo\Sample.exe"

With window("text:= LOGIN")

- Activate.
- WinEdit ("AttachedText:= Username") . Set "Sunny"
- WinEdit ("AttachedText:= Password") . Set "Sunny PW"
- WinButton ("text:= Login") . click.

End with

If window ("text:= welcome") . Exist (10) Then

MsgBox ("application is OK")

{ Else MsgBox ("application is not OK") }

Elseif window ("text:= ERROR") . Exist (10) Then

B) VB Technology:-

1. Introduction:- Many standalone / desktop window based application can be developed with the help of well known VB technology that comes from Microsoft. Hence the test engineer must aware of the reference object, corresponding methods and the property description to developed the test scripts.

2. Environment:-

- Install QTP with VB addin.
- Install VB Software
- Install the AUT developed in VB.
- Launch QTP with selecting VB addin.
- Launch the AUT for test.

3. Object table:-

No.	Technology	Object	Reference object	Description Properties
1	VB	VB window	VB window	Name
2	VB	VB objects	vb button vb edit vb radio button vb check box vb combobox vb frame	Name
			other objects.	window id (in case no name)

4. TASK:-

Exercise 1:-

1. To developed VB based QTP script for automating flight booking activity.

2. To check if the booking is successful with the following information.

Flight no: IA2880

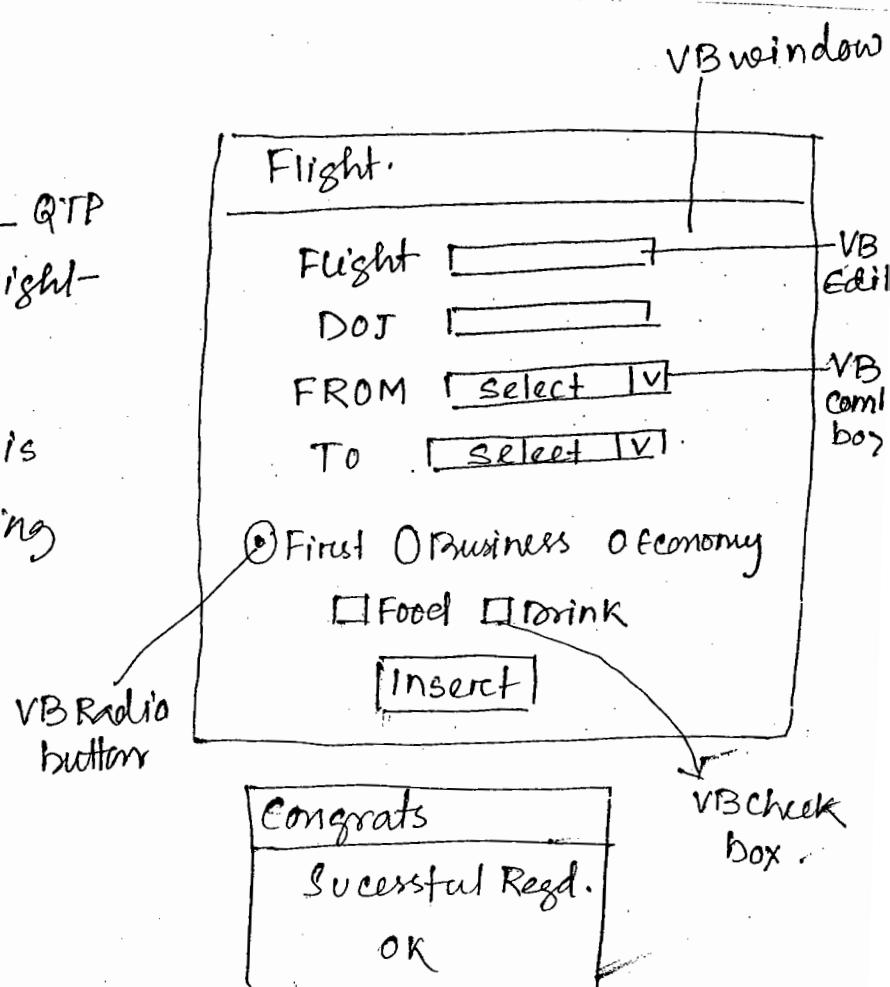
DOJ : 5/5/2011

From: Hyderabad.

To: Delhi

Class: First

Food: Check and Drink.



abc flight appl": [c://Hyderabad\flights\ abc flight.exe]

Ans:- Systemutil.Run "C://Hyderabad\Flight\abcflight.exe".
Name

With VB window ("text:= Flight")

- Activate.

- Vbedit • VB edit ("Name:= Flight"). Set "IA2880".

- Vbedit • VB edit ("Name:= DOJ"). Set "5/5/2015"

- Vbcheckbox • VB checkbox ("Name:= FROM"). Set "Hyderabad".

- Vbcheckbox • VB checkbox ("Name:= TO"). Set "Delhi"

- VbRadioButton • VB RadioButton ("Name:= First"). Set

- Vbcheckbox • VB checkbox ("Name:= FOOD"). Set "ON"

- Vbcheckbox • VB checkbox ("Name:= DRINK"). Set "ON"

- Vb button ("Name:= Insert"). Click.

End with.

If vb window ("Name:= Congrats!") exist(10) Then

```
    MsgBox ("Successful booking")
else
    MsgBox ("Booking not successfully")
endif.
```

Note:- Some time lengthy items are presents in the combo box which are not fully visible. In this case "getitem(i)" function can be used with item index value in it (item value starts with '0') $i = 0, 1, 2, 3, \dots$
Once the item is captured it can be stored in a variable and can be used further

Ex:- On the above/before program change this:-

Line1: Dim X,Y

X = .vbcombobox ("Name := FROM"). getitem(2)
.vbcombobox ("Name := FROM"). select X

Y = .vbcombobox ("Name := TO"). getitem(5)
.vbcombobox ("Name := TO"). select Y

=====

ask 2:-

1. To invoke Wedlock.exe
2. Display the status of Mr. Saidhar during runtime as per the application. for the following info.
 1. Name: Saidhar
 2. Age: 28
 3. Gender - Male

3. Check the application works fine for sridhar
4. Check If the default selection for the gender is ok in the application.

Marriage Eligibility

Name	<input type="text"/>
Age	<input type="text"/>
Gender	<input checked="" type="radio"/> M <input type="radio"/> F
Status	<input type="text"/>
<input type="button" value="OK"/>	

Ans:- Dim X Y
 Systemutil.Run "C:\Hyderabad\mb\wedlock.exe"
 with vbwindow ("Name:= Marriage Eligibility")

Default statement • Activate
 • vbedit ("Name:= Name"). Set "sridhar"
 • vbedit ("~~Name~~:= Age"). Set "28"
 • vbradiobutton ("Name:= Male"). Set
 • vbutton ("Name:= OK"). Click
 X = • vbedit ("Name:= status"). getproperty ("text")

End with

Msgbox (X)

Note:- When ever run time information is to be capture, getproperty method is used. This method takes various parameter for various situations as describe below.

1. "Text" :- This property is used for capturing the display the run time value from textbox, combobox,

② "checked": — This property is used to check the selection status of checkbox, radio button etc; in this case getproperty returns true or false false depends on checked or unchecked

③ "enable": — This parameter is used to check if the objects are enable; In this case getproperty returns true or false depend on enable/disable.

for program continue :-

End with

msgbox(x)

If (x = "Eligible") Then

msgbox ("application is ok for Sandeep")

else

msgbox ("application is not ok for Sandeep")

endif

default
tolerant — y = .vbradiobutton ('Name= Female').getproperty ('checked')

If (y = "True") Then

msgbox ("default selection of gender is ok")

else
msgbox ("default selection of gender is not ok").

end if

```
Dim X Y  
System.util.Run "C:\Hyderabad\MBI\wedlock.exe"  
With VBbutton ("Name:=Marriage Eligibility")  
    • Activate.  
    Y = .Vbradiobutton ("Name:=Female").getproperty ("checked")  
    • Vbedit ("Name:=Name").Set "Saidharc"  
    • Vbedit ("Name:=Age").Set "28"  
    • Vbradiobutton ("Name:=Male").Set  
    • Vbbutton ("Name:=OK").click  
    X = .Vbedit ("Name:=status").getproperty ("text")  
End With.
```

Msgbox(X)

If (X = "Eligible") Then

Msgbox ("application is ok for Saidharc")

Else

Msgbox ("application is not ok for Saidharc")

End If.

If (Y = "True") Then

Msgbox ("default selection of gender is ok")

Else

Msgbox ("default selection of gender is not ok")

End If

C) .Net technology:-

1. Introduction:- This technology comes from microsoft- is being used for the development of complex window based application that are to be tested by the corresponding test script.

2. Environment:-

- a) Crystal QTP with .net addin.
- b) Crystal .net framework.
- c) Crystal application developed in .NET.
- d) Launch QTP with .net addin selected.
- e) Launch the AUT.

3. Object Property table:-

Sl.no.	Technology	Object Hierarchy	Reference Object	Description Property
3.	MS.net.	swfwindow	swfwindow	
			swfbutton swredit swfradioButton swfcheckbox	swfname
		swfobjects.	swfcombobox	windowid classname
			swfframe	

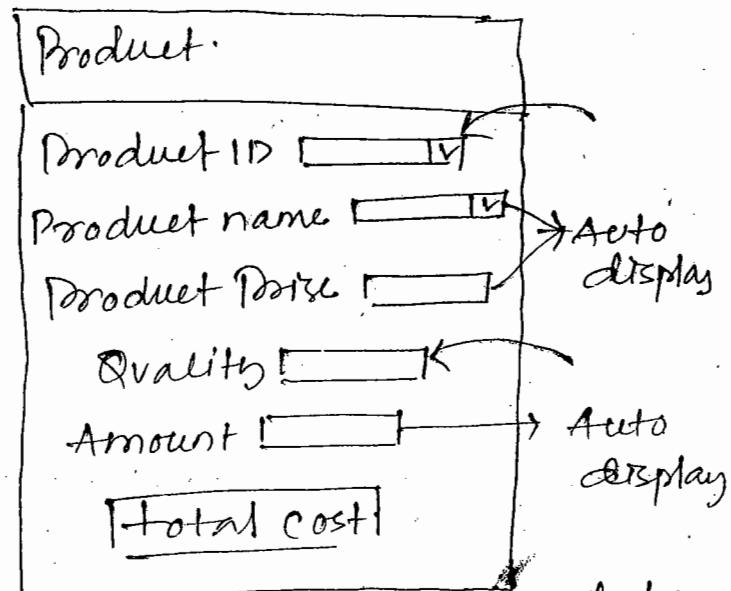
Swf- Short waveflash

4. Task

Exercise - 1 :-

Assumption :- When Product ID is entered product name & Product price fields are auto displayed.

Amount can be displayed
on clicking total cost button.
for the given quantity.



1. Develop the script to automate the total cost calculating activity for the following information.

Product ID: P-001

Name: Electronic watch

Price: \$100

Quantity: 5

- Quantity: 5
2. Check if the application functionality is ok for the above given data.

- given data.

3. Check if the total cost button is disable initially and gets enable only on the entry of quantity.

declaring variable

Ans: $\dim X, \text{tc}, \text{ev}, \text{av}, u, v$

Ans:- Invoking Dim X,fc,env,av,u,s System util. Run "C:\Hyderabad\morse\Product.exe"

with `SwfWindow ("SwfName:= Product")`

Operational Script

• Activar:

(a) ~~x = .swfcombobox C"NameSwfName:=Product1D") .getitem(0.~~

: swtfromComboBox ("swtName:= ProductID") . select x

- SwfcomboBox C"swfName:=ProductID").Select X

• `Swfedit("swfName := Quantity")` • `Set "5"`

•Swf button ("swfName:=TotalCost"), click.

```
function fC = .swfedit("SwfName:= Amocent").getproperty("text")
```

2/ msgbox(tc)

ar=tc

P = 100

q = 5

ev = (P * q)

If (ev = ar) Then

msgbox("application is ok")

else

msgbox("application is not ok")

End if.

(without string)

on string conversion

Verificational script

ar=tc

ar = cstr(ar)

P = 100

q = 5

ev = (P * q)

ev = cstr(ev)

If cstrcmp(ev, ar) = 0 Then

msgbox("app is ok")

else

msgbox("app is not ok")

End if.

a = .swfbutton ("SwfName:= TotalCost").getproperty ("enabled")

b = .swfbutton ("SwfName:= totalCost").getproperty ("enabled")

[Checking default status] AND

If (a = "False") Then AND (b = "True") Then

msgbox("default status of total cost button is ok")

Else msgbox("default status of total cost button is not ok")

Declaring Variable

Dim x, te, ev, av, a, b

Invoking application

System util.Run "C:\Hyderabad\mace\Product.exe
With SWFwindow ("SWFName:= Product")

Operational script

• Activate

a = • SWFbutton ("SWFName:= Total Cost"). getproperty ("Enabled")

x = • SWFcombobox ("SWFName:= ProductID"). getItem (0)

• SWFcombobox ("SWFName:= ProductID"). Set "5" Select x

• SWFedit ("SWFname:= Quantity") . Set "5"

~~Set~~ SWFcomboboxes

b = • SWFbutton ("SWFname := totalcost"). getproperty ("Enable Confirmation")

• SWFbutton ("SWFname := totalcost") . Click

te = • SWF~~button~~ edit ("SWFname:= Amount") . getproperty ("Text")

End With.

Verification Script - String Conversion

av = te

av = CStr(av)

P = 100

q = 5

ev = CP * q

ev = CStr(ev)

If (strcmp(ev, av)) = 0 Then

MsgBox ("app is OK")

else

MsgBox ("app is not OK")

End If.

Checking default status:-

If (a = "False") AND (b = "True") Then

MsgBox ("default status of total cost button is OK")

else MsgBox ("default status of total cost button is not OK")

End If.

d) Java technology :-

1. Introduction:- Industry preferred/opts for java technology as it's open source to develop complex window based application which are to be tested by the respective test script.

2. Environment:-

- a. Install QTP with Java addin.
- b. Install Java software along with JDK.
- c. Install Java application to be tested.
- d. Launch QTP with Java addin selected.
- e. Launch the Java application.

Java Development Kit

3. Invoking/Launch Process of Java application:-

Step 1: copy Java file in the following path.

c:/Programfile/java/Jdk 1.5.22/bin

Step 2: include the path above path by executing 'cd' command on the command prompt.

Step 3: Compile the Java file using java C command
Command as given below

> javac sample.java

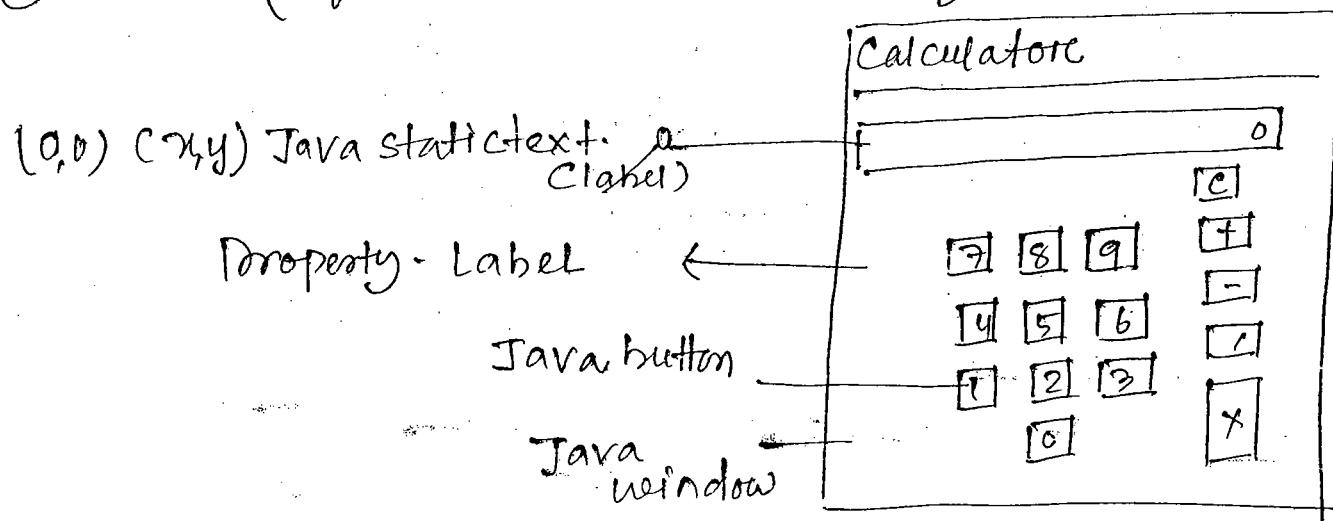
Step 4: Execute Java file to invoke it using
java command as given below.

> java sample.

Sno	Technology	Hierarchy	Reference	Description
			Object	Property
4.	Java	Java Objects	Java window	
			java button	
			java edit	
			java Radio button	
			java statictext	
			java combobox	
			java frame	

5. Task:-

- ① Developed QTP based QTP script to automate addition operation of 10 and 20 and to display the sum during runtime.
- ② To check if the calculator is working fine or not.



Ans!- Dim X, ev, av
With javawindow ("Label:= calculator")

- Activate
- javabutton ("Label:= C").click
- java button ("Label:= !").click
- Java button ("Label:= 0").click
- java button ("Label:= +").click
- Java button ("Label:= 2").click
- Java button ("Label:= 0").click

• javabutton ("Label: ==").click.

X = .javastatictext ("x=0", "y=0").getproperty ("Label")

End with.

msgbox (X)

ev = 30

av = X

If (ev=av) Then

msg box ("calculator is ok")

else

msgbox ("calculator is not ok")

End If.

Date: 03.02.2015

2) SAP Technologies:-

1. Introduction:- Now a days organization are implementing full and complete business solution in terms of ERP. SAP is one among the most important technology. Hence there is need for testing these application.

2. Environment:-

- (1) Instal QTP with SAP addin.
- (2) Instal specific module of SAP.
- (3) Instal the AUT developed SAP.
- (4) Launch QTP with SAP addin.
- (5) Launch the AUT to be tested.

(3) Object Table:-

<u>Sno.</u>	<u>Technology</u>	<u>Object Hierarchy</u>	<u>Object Reference</u>	<u>Description Property</u>
1	SAP	SAPgui session.	Sapgui session.	
2		SAPgui window	Sapgui window	
3		SAPgui button	Sapgui button	Name or Id.
4		SAPgui edit	Sapgui edit	
5		SAPgui table	Sapgui table	
6		SAPgui calendar	Sapgui calendar	

(4) Task:-

Session 1				
HR-Module				
Employees				
S.L	Name	Design	Salary	
1	Tom	QL	75000	
2	Sridhar	TL	50,000	
3	Kalyan	STE	35,000	
4	Sita	TE	25,000	

- (1) Capture salary of Sridhar along with name.
- (2) To capture salary of - Kalyan along with name.
- (3) Display the message of - loan approved for a specific Person whose salary is more than 40,000.

Dim s_1, n_1, s_2, n_2

with sap gui session ("Name:= session1")

• Activate.

with .sapguiwindow ("Name:= HR-module")

• Activate.

$s_1 = \text{Sap guitable} ("Name:= Employees").\text{get cell data}(2, 4)$

$n_1 = \text{Sap guitable} ("Name:= Employees").\text{get cell data}(2, 2)$

$s_2 = \text{Sap guitable} ("Name:= Employees").\text{get cell data}(3, 4)$

$n_2 = \text{Sap guitable} ("Name:= Employees").\text{get cell data}(3, 2)$

End with

If ($s1 > 40000$) then

msgbox ("Loan is approved for " & n1)

else

msgbox ("Loan is not approved for " & n1)

End If.

If ($s2 > 40000$) then

msgbox ("Loan is approved for " & n2)

else

msgbox ("Loan is not approved for " & n2)

Endif.

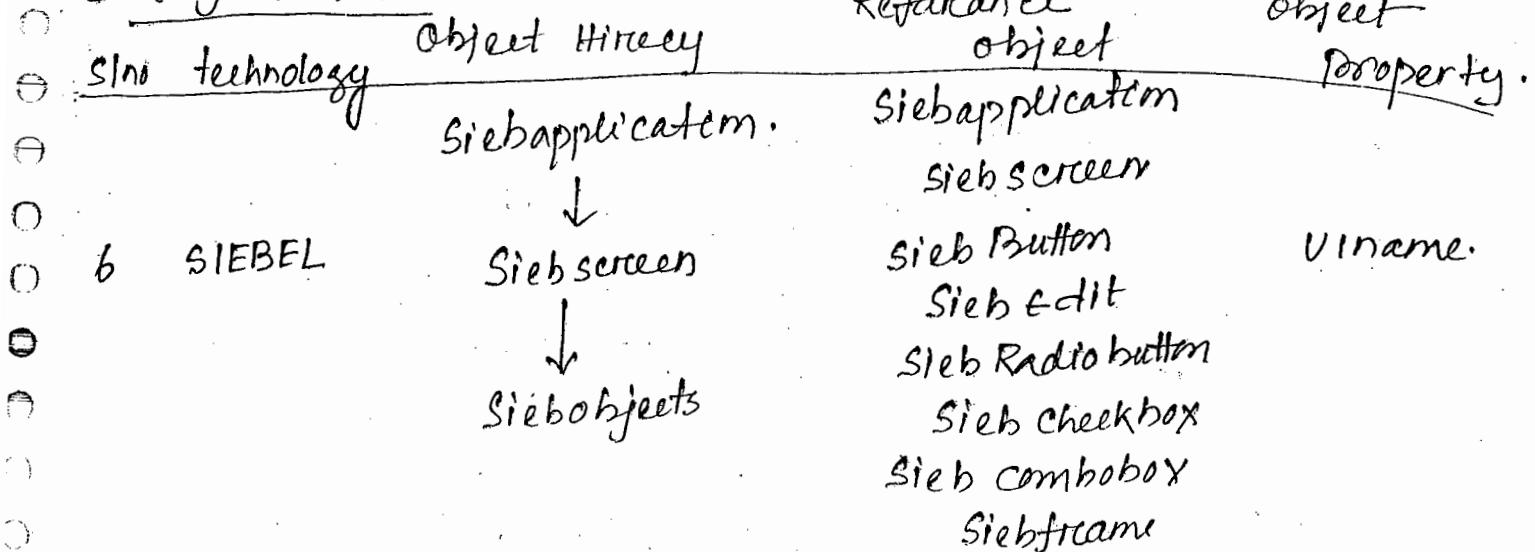
F) SIEBEL Technology:-

1. Introduction:- Industry nowadays developeds long term and healthy relationship to the customers. In this connection there are so many CRM products that are developed with the technology like SIEBEL. Hence the automation tester must be potential to test Siebel window with appropriate reference objects and Method.

2. Environment:-

- a. Install QTP with SIEBEL addin.
- b. Install Siebel CRM framework.
- c. Install Siebel application to be tested.
- d. Launch QTP with Siebel addin select.
- e. Launch the AET.

3. Object-table



6 SIEBEL

Siebscreen



Siebobjects

Sieb Button

Sieb Edit

Sieb Radio button

Sieb Checkbox

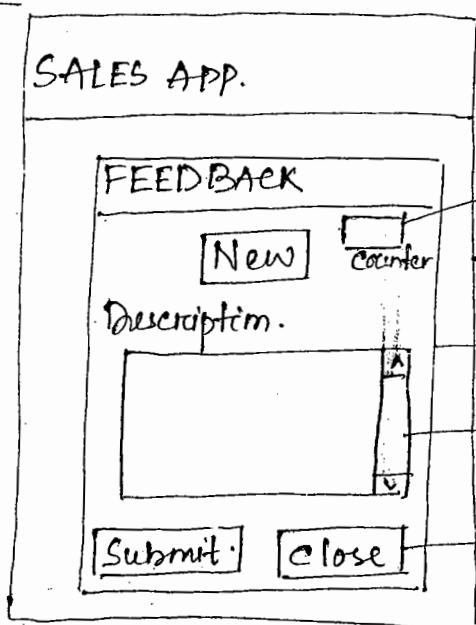
Sieb Combobox

Siebfframe

UIname:

4. Task:-

①



"C:/demo/Apps/sales App.exe"

Feedback counter Path.

Sieb application.

Sieb screen

Sieb edit

Sieb button.

UIName.

1. To Launch the sales app application present in the above Path.
2. Create the a feedback - "Product is ok. But the user guide is not friendly".
3. Submit the feed back and closed the feedback window.

Ans:-

System Util.Run "C:/demo/Apps/Sales Apps.exe".

With Sieb application ("Viname:= SALE APP")

- Activate.

With Sieb screen ("Viname:= FEEDBACK")

- Activate.

1. → • Siebbutton ("Viname:= New"). click.

• Siebbutton ^{Edit} ("Viname:= Description"). Set "Product

is OK. But userguide is not friendly".

• Siebbutton ("Viname:= Submit"). click.

2. → • close

End with.

End with.

ore { • Siebbutton ("Viname:= Close"). click

End with.

End with

1. a = .siebedit ("Viname:= count"). getpropertyc("text")

2. b = .siebedit ("Viname:= count"). getpropertyc("text")

→ If (b=a+1) then

msgbox ("Submission is successfull")

else

msgbox ("Submission is not successfull")

Ans:-

System util. Run "C:/demo/Apps/sales APPS.exe".
with Sieb application ("Uiname := SALE APP")

Activate

With Sieb screen ("Uiname := FEEDBACK")

• Activate

• Siebbutton ("Uiname := New").click.

• SiebEdit ("Uiname :=

a = • Siebedit ("Uiname := count").getproperty("text")

• Siebbutton ("Uiname := New").click

• SiebEdit ("Uiname := Description").set "Product

"is OK. But user guid is not friendly!"

• SiebEdit button ("Uiname := Submit").click

b = • Siebedit ("Uiname := count").getproperty("text")

↓ • Close

EndWith

End With.

If (b = a+1) Then

MsgBox("Submission is successful")

Else

MsgBox("Submission is not successful")

Tools → Objects Spy

Object Spy - Whether the incoming window is dialog box or window or any more. It gives the original logical name

Creating DP Based Script file Insert order operation :-

~~window C" text := Flight Reservation~~

with window C" text := Flight Reservation")

• Activate.

• win Button C" window id := 6"). click.

• Active X C" nativeclass := MSMask Wnd Class", "window id := 0")
• type "040315

• win ComboBox C" attached text := Flyfrom:"). select "London"

• win ComboBox C" attached text := Flyto:"). select "Paris")

• win Button C" text := Flight"). click

• Dialog C" text := Flights table"). win List C" attached text := From", "window id = 2001"). select "Hyderabad".

• Dialog C" text := Flights table"). win Button C" text := OK"). click.

• Activate

• win Edit C" attached text := Name:", "window id := 1014").
set "Sridhar"

• win Radio Button C" text := First"). set

• win Button C" text := f Insert order"). click.

End with.

Assignments:-

1) Write a program to automate update order and delete
Order operation.

- 2) Check if the marriage eligibility application works fine for anybody. ex:-
`z = Inputbox ("Enter your gender M/F")`
`radioButton ("text:=" & z).set .`
- 3. WAP to check the functionality of product window for any input

DP Based QTP Scripts for Testing Web Application

Industry mostly uses complex web based application as their business solution ~~difficult~~ with various technologies which need to be tested with appropriate scripts by the automation tester.

A) HTML Web based application testing :-

1. Introduction :- There are many simple web based application/ website developed in HTML which are to be tested. HTML = Text markup language.

2. Environment:-

- Install QTP with Web addin.
- Install the required Browser
- Install Web application/ website .
- Launch QTP with Web addin selected .
- Launch the Web application/ website .

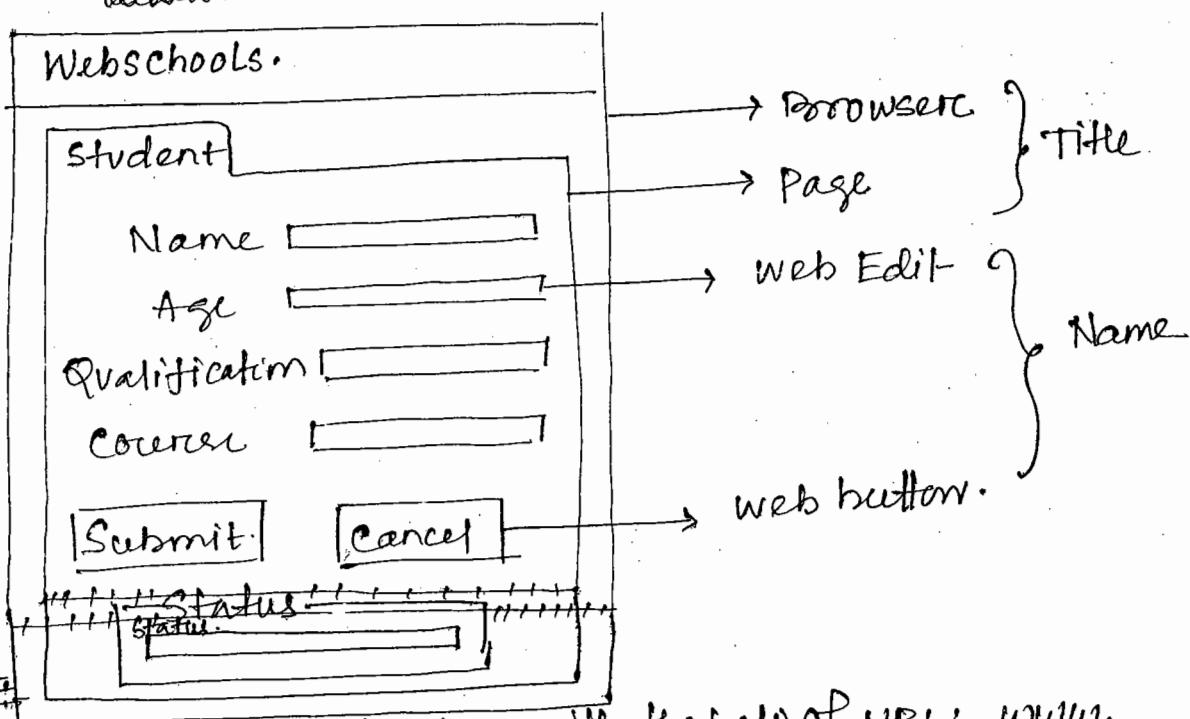
3. Object Property Table:-

Sl.No	Techno.	Object Hierarchy Hierarchy	Reference-Object	Description Properties
1	HTML	<pre> Browser ↓ Page ↓ Frame (opt) ↓ web objects. </pre>	<pre> Browser Page Frame Link Image Web Edit Web button Web radiogroups Web table Web list Web combobox, etc </pre> <p style="text-align: center;">&</p> <p>webelements.</p>	Title or name/ creation time Name/index/ location. HTML tag or index/location

Task :-

Exercise - 1.

Address. www.webschool.com.



Task - 1

1. Launch Web school site with the help of URL:- www.webschool.com.

2. To automate registration activity with the help of following information
 name: gridharc
 age: 25
 Qualification: MCA
 Course : Software testing
3. Close the page and ~~minimize~~ browser.

~~Ans:- QNP~~
 Note:-

~~System util.~~ (A) Web application can be invoked in the following methods :-

- a. System util. run " URL "
- b. System util. run " Path of Browser ", " URL "
- (B) One must consider Browser and first parents that can be identified with title / name property .
- (C) After the Browser one must consider the " Page " as second parents while writing the script .
- (D) Noneed of activation of Web pages (don't write " Activate ")

Ans:-

System util. run " www. ~~george~~ web school . com "

with browser (" title: = Webschools ")

with • Page (" title: = student ")

- Web edit (" Name: = Name "). Set " gridharc ";
- Web edit (" Name: = Age "). Set " 25 "
- Web edit (" Name: = Qualification "). Set " MCA ";
- Web edit (" Name: = Course "). Set " Software testing "
- Web button (" Name: = Submit ") . Click .

with • frame (" Name: = status ")

X = . Web edit (" Name: = status "). Set property (" text ")

End with

msgbox (x)

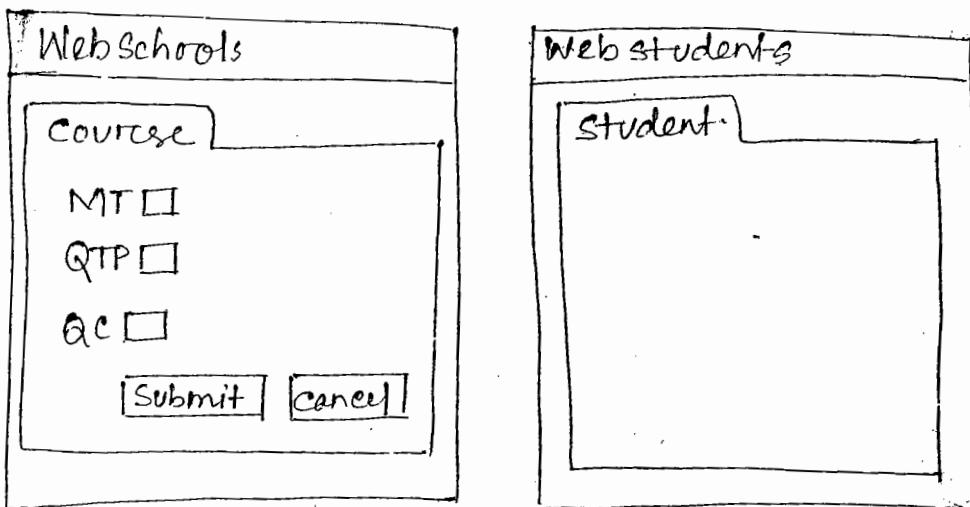
• close

End with.

• close

End with.

Task-2:-



Testcm:-

1. Launch the Web schools application.
2. To automate ^{Course} selection operation in the course page of the Web schools browser while the other browser is open with the same name
3. Check if the cancel button functionality is ok.

To do the operation on a specific browser, from present of many browser we have to chose "Createonetime" that setting from "object spy"

Dim x, y, z, a

Systemutil.Run "www.webschools.com"

With browser ("title:= web schools", "createonetime:=5")

With Page ("title:= course")

- Webcheckbox ("Name:= MT").set "ON"

- Webcheckbox ("Name:= QTP").set "ON"

- Webcheckbox ("Name:= QC").set "ON"

- Web button ("Name:= Submit").click
- Webcheckbox ("Name:= MT").set "ON"
- Webcheckbox ("Name:= QTP").set "ON"
- Webcheckbox ("Name:= QC").set "ON"
- Webbutton ("Name:= cancel").click

$X = \text{Webcheckbox} ("Name:= MT").getproperty ("checked")$

$Y = \text{Webcheckbox} ("Name:= QTP").getproperty ("checked")$

$Z = \text{Webcheckbox} ("Name:= QC").getproperty ("checked")$

End with.

- If ($X = \text{False}$) AND ($Y = \text{False}$) Then
 - $a = 1$
 - else
 - $a = 2$

End if.

If $a = 1$ AND ($Z = \text{False}$) Then.

Msgbox ("cancel is working fine")

End else

Msgbox ("cancel is not working")

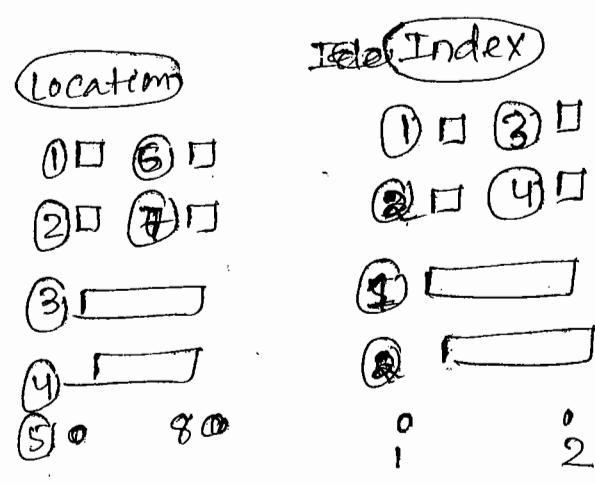
End if

End with.

Task-3 :- www. Sample application

Sample.

1 Registration:	①	②
abc	abc	abc
abc	abc	abc
3 my2	3	
4 my2	1	
Gender		
5	Male	Female
OK		



Radio group.

1. To invoke Sample application.
2. To enter "Sridhar" in "4th textbox"
3. To enter "age 25" in the "2nd xyz textbox"
4. To select male option from gender radio group

Note:-

The title of the browser keeps on changing dynamically.

Points to be noted:-

1. whenever the Browser title is dynamically changed the following syntax has to be used

Browser ("title := •*")
2. In case object have common name/property it can always be identified based on index ID or location etc ID as given in the above window.
3. To select any radio button option from the group the following syntax must be used

Ex:-

~~win Radio~~

Web Radiogroup ("Name:=gender", "index:=0") . Select "#0"

```
System util . Run " www . sample . com "
with browserc (" title : = * ")
with . Page (" title : = * ")
    • web edit (" Name : = abc ", " index : = 3 ") . set " Soidhar "
    • Web edit (" Name : = xyz ", " index : = 1 ") . set " 25 "
    • Web radio group (" Name : = genderc ", " index : = 0 ")
        • select "# 0 "
    • Web button (" Name : = Ok ") . click .
```

End with .

End with .

Task - 4

1. Launch Google site
2. Enter software testing as a search keyword .
3. To click search button in order to display the related site .

Ans:-

```
System util . Run " www . google . com "
with browserc (" title : = Google ")
with . Page (" title : = Google ")
    • Web edit (" Name : = q ") . set " Software testing "
    • Web button (" Name : = Search GoogleSearch ") . click
```

End with .

End with .

B) HTML with Java Script technology:- Dt: 04.04.2015

1. Introduction: — Java script is mainly used as client site program mostly to validate user inputs while displaying error message / confirmation window / Alerts on the web page created with/by HTML. Such operation can be automated with the appropriate scripts.

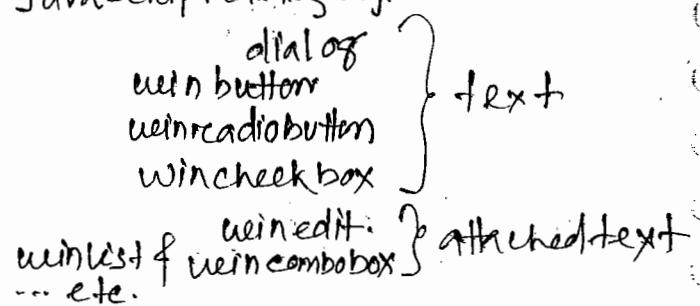
2. Environment:-

- a. Instal QTP with webaddin
- b. Instal browsers that required.
- c. Instal web application. that is ambadeel with java script.
- d. Launch QTP with web addin.
- e. Launch the Web application.

3. Object table:-

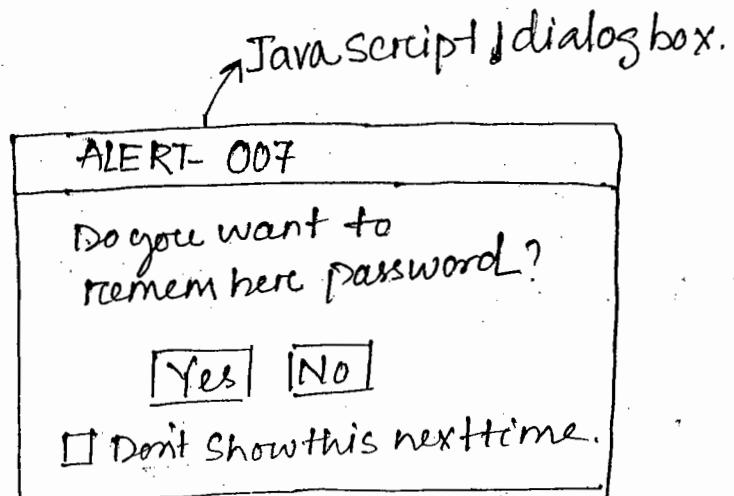
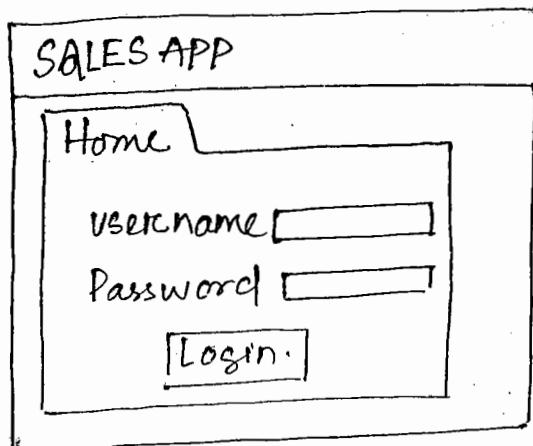
Ino.	Technology	Object Hierarchy	Reference objects.	Description Properties.
			<u>HTML objects:-</u>	
			Browser page	Title/creationtime
			webObjects (Image, frame, webedit btn....)	Name/index/ location
			webElement	HTMLtag/index/ location.
2	HTML & Java Script.	↓ Dialog ↓ win objects.		

JavaScript dialog obj:-

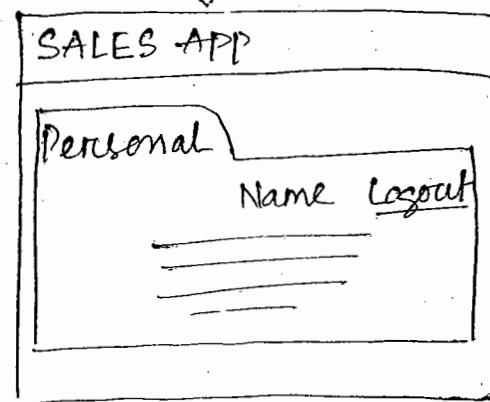


④ Task:-

Exercise 1:-



1. To invoke Sales app web application?
2. To automate login activity?
3. To handle Alert - dialog box by doing the following operation.
 - a. To ask the system not to remember the password?
 - b. To ask the system no to show it again?
4. Logout from the personal page?
5. Check the login is successful:



Ans:- Systemutil.Run "www.SALES APP.com"

with browser("title:= SALES APP")

with • Page ("title:= Home")

- Webedit ("name:= username").Set "Steidhar"
- Webedit ("name:= Password").Set "Mercury"
- Webbutton ("name:= Login").click

EndWith

With • dialog ("text:= ALERT- 007")

- Winbutton ("text:= No").click
- Wincheckbox ("text:= Don't").~~Set~~ Set "ON"
- Winbutton ("text:= No").click

EndWith

EndWith.

{ browser ("title:= SALESAPP").Page ("title:= Personal").link ("name:= Logout").

etc.

EndWith.

* Page ("title:=SALEAPP").Page ("title:= Personal").

Link ("~~link~~ name:= Logout"), click.

EndWith.

For success login activity:-

*) If. Page ("title:= Personal") Then exist (10) Then,
 MsgBox ("Login is successful")
 else
 MsgBox ("Login is not successful")
End If.

1) HTML With Java technology:-

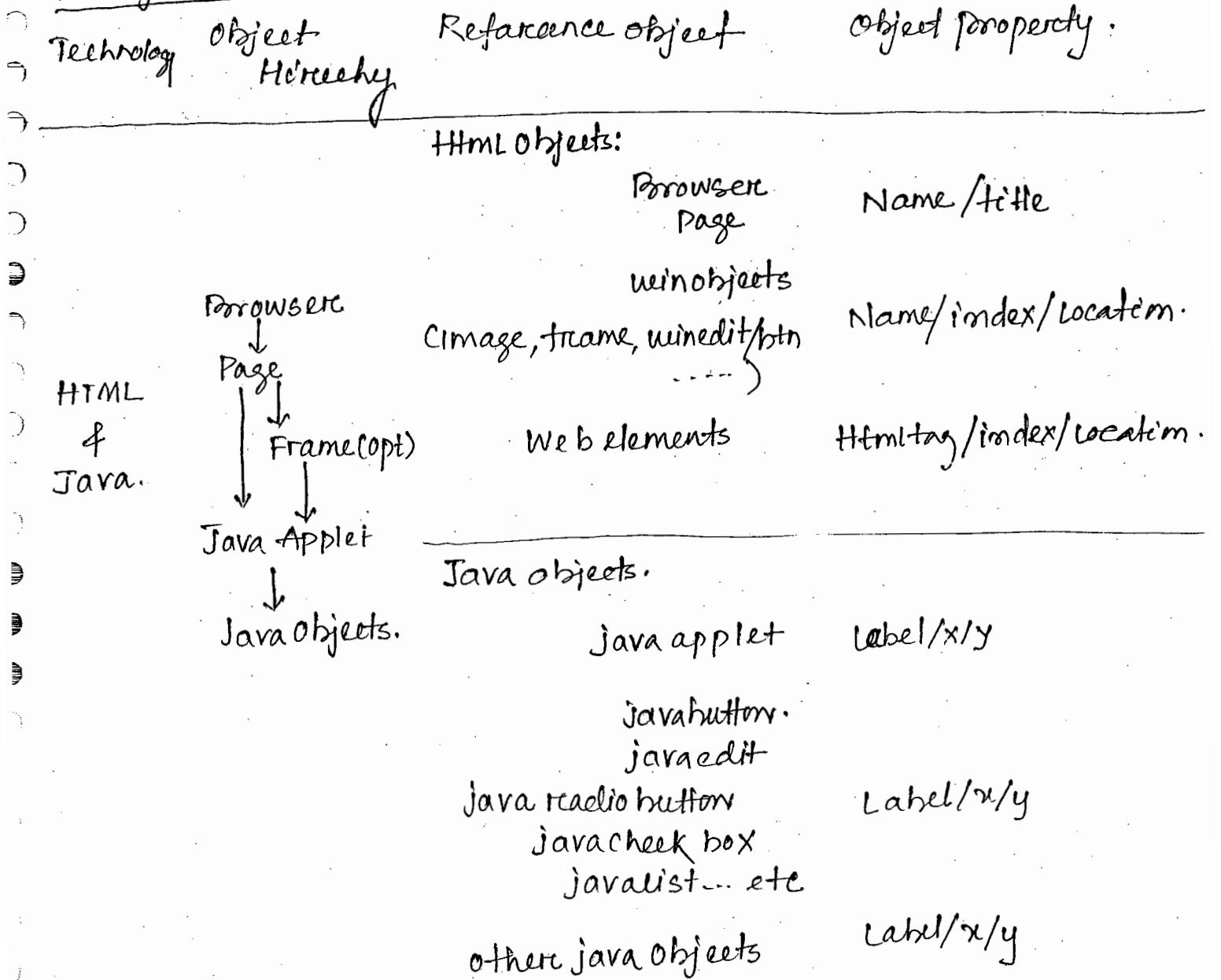
1. Introduction :- Now a days java is used for developing complex web based apps which are to be tested with appropriate automated scripts.

2. Environment:-

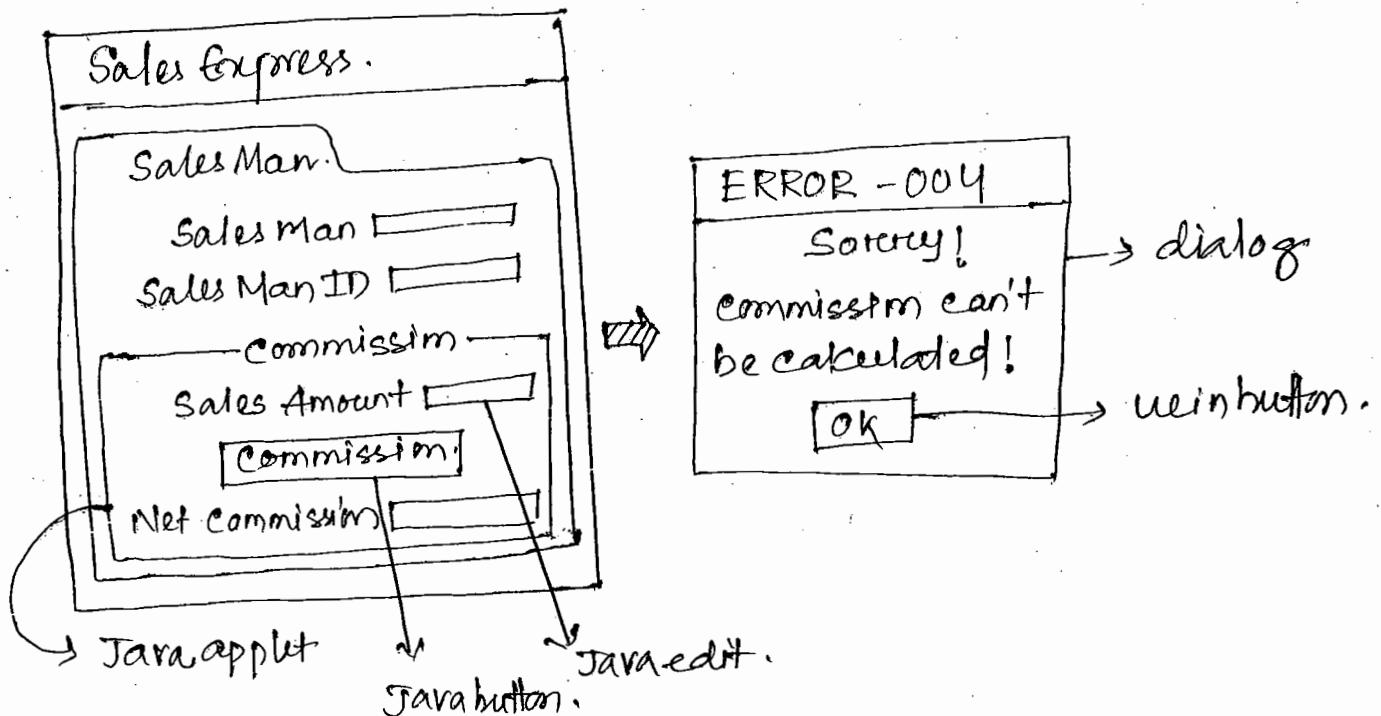
- a. QTP Instal QTP with Java addin.
- b. Instal the required browser along with Java frame work - Jdk.
- c. Instal java web application.
- d. Launch QTP with Java addin select.
- e. Launch the java application.

Note:- For java based web applications the source code is convert into deployable formats - ~~JAVA~~/ Jar/War/Ear which can be deployed into apps server. and the corresponding domain configuration is done to access the web application through ~~client~~ client.

3. Object Table:



4. Task:



Functionality

1. Purpose is to calculate commission.
2. 10% commission on the sales amount calculate.
3. Commission is calculated only for the amount > or = 10,000.
4. Once invalid input the error msg dialog box is appeared as shown in picture.

task:-

1. To invoke sales express web application.
2. To calculate commission with the following specification.
 - a) Sales Man name : Sidhar.
 - b) Sales man ID : SM-007
 - c) Sales amount : 20000
3. To check the application is fine for Sidhar.
4. To check the functionality of the sales express application if it's correct or not.

for java in getproperty:
window base it is ("level")
web base it is ("value")

Ans:-

```
1 Dim c, ev, av
2 System util. Run "www. Sales Express .com"
3 with browser("title:= Sales Express")
4 with .Page("title:= Sales man")
5 • Webedit("name:= Salesman").Set "Sidhar"
6 • Webedit("name:= Salesman ID").Set "SM-007"
7 with .Javaapplet("Label:= commission")
8 • Javaedit("Label:= Saleamount").Set "20,000"
9 • Javabutton("Label:= Comission").click.
10 c = • Javaedit("Label:= Netcommission").getproperty("value")
11 End with
```

av = cstr(c)
ev = (20000 * 10) / 100
rv = cstr(ev)

If strcmp(ev, av) ≠ 0 then

Msgbox ("application is OK for Saidhar")

else

Msgbox ("application is not OK for Saidhar")

Endif

For dynamic Put a line before the given number.

Before(5) :- Sm := Inputbox ("enter name").
Id := Inputbox ("enter id").
sa := Inputbox ("enter amount").

then

Line(5) :- .WebEdit ("name:= Salesman"). Set Sm.

Line(6) :- .WebEdit ("name:= SalesmanID"). Set Id.

Line(8) :- .JavaEdit ("name:= Salesamount"). Set sa.

Comment the line 10 and continue below for
the Question functionality - 3 of 4

If (sa >= 10000) Then

av = cstr(c)

ev = (sa * 10) / 100

rv = cstr(ev)

If strcmp(ev, av) = 0 then

Msgbox ("application is OK for Saidhar")

else

Msgbox ("application is not OK for Saidhar")

Endif

elseif dialog("name:= ERROR-004"). Exist(10)

Msgbox ("application is OK")

else

Msgbox ("application is not OK")

卷之三

① HTML with .Net technology:-

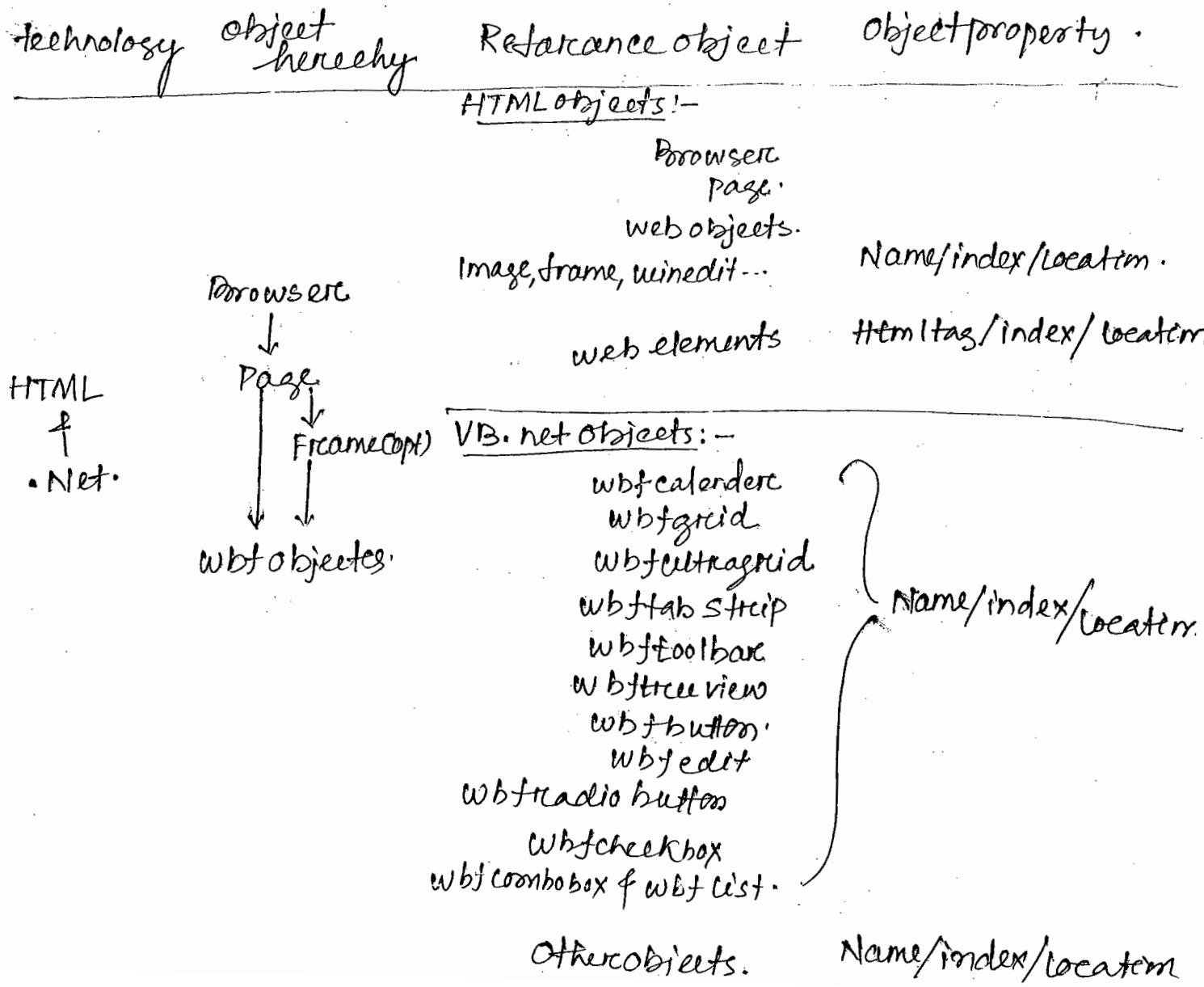
1. Introduction:- Ms.net technology from Microsoft is widely used for developing large scale ~~enterprising~~ application in the web environment which need to be tested with appropriate scripts.

2. Environments:-

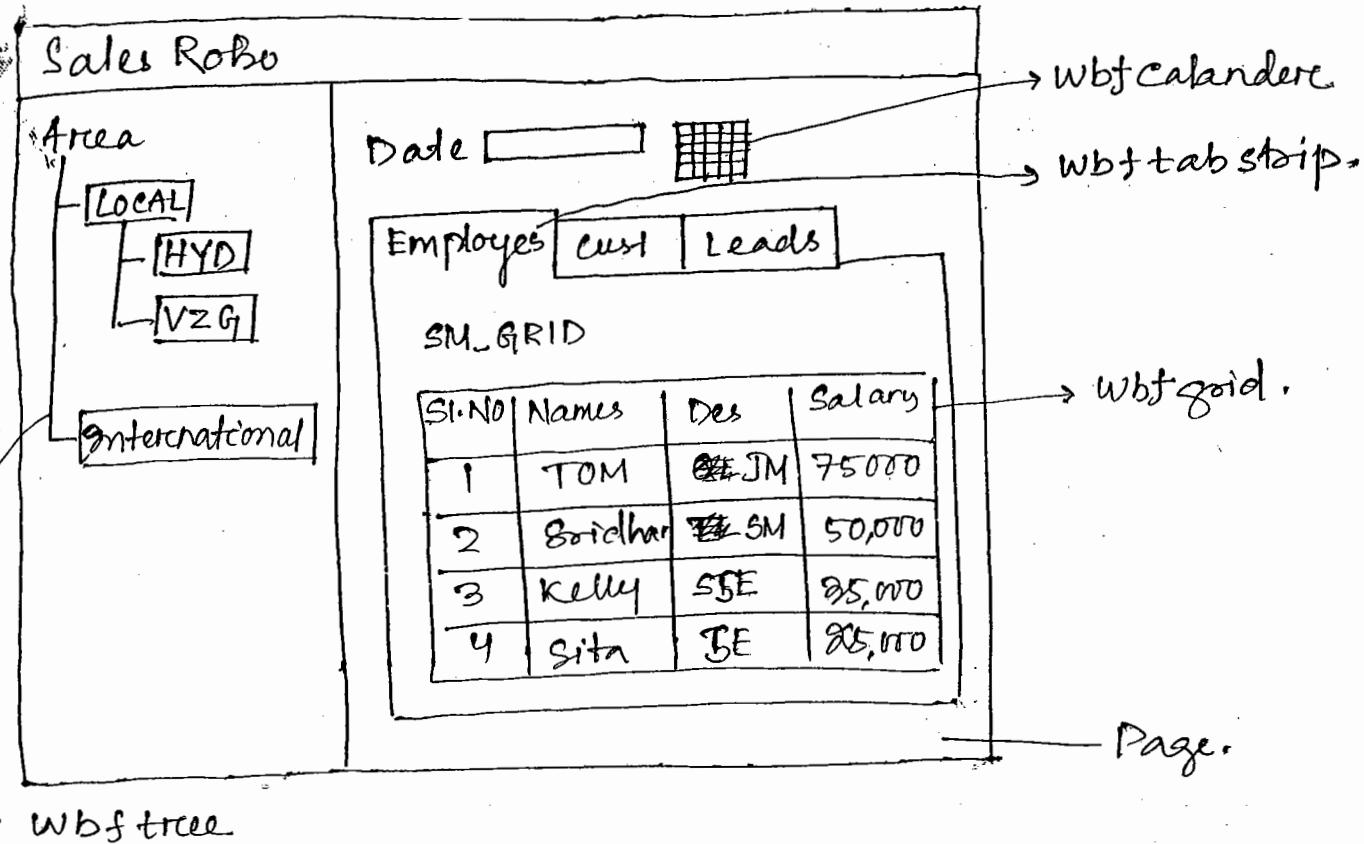
- a. Install QTP with .net addin.
- b. Install browser of the .net framework
- c. Install .net web application.
- d. Launch QTP with .net addin selected.
- e. Launch .net application.

3. Object table:-

WBF - Windows bitmapic framework



4. Task:-



1. To invoke Sales Robo application?
2. To display the required number of Sales App. in grid based on the following activity:
 - (a) To select todays date
 - (b) Select HYD as a Local area.
 - (c) click on employees tab
3. To display all the names from the grid along with the designation during run time in terms of following format. "JM is TOM".
4. To recommend a salary hike for the sales professional whose salary $\leq 30,000$.

Ans:-

```
0 — Dim RC, i, d, n, s
1 — Systemutil.Run "www.SalesRoBo.com"
2 — with browserc ("title:= Sales RoBO")
3 — .with. Page ("title:= Sales ROBO")
4 — • Wbfcalenderc("name:= Date").SetDate "04/04/2015"
5 — • Wbftreeview("name:= Area").Select "Area;Local;HYD"
6 — • Wbftabstrip("name:= Employees").Select "employees"
7 — RC = .Wbftgrid("name:= SM_GRID").RowCount
8 — for i = 1 to RC
9 —   d = .Wbftgrid("name:= SM_GRID").GetCellData(i)
10 —   n = .Wbftgrid("name:= SM_GRID").GetCellData(i, 2)
11 —   msgbox(d & " " & "is" & " " & n)
12 —
13 —   if (s <= 30000) then
14 —     msgbox ("n &" is eligible for hike")
15 —   else
16 —     msgbox ("n &" not eligible")
17 —   Endif
18 —   next
19 — End with.
20 — End with.
```

E) HTML with SAP technology:-

1. Introduction :- ERP in terms of SAP takes vital role in creating full fleged web based application which are supposed to be tested with appropriate test script.

2. Environment :-

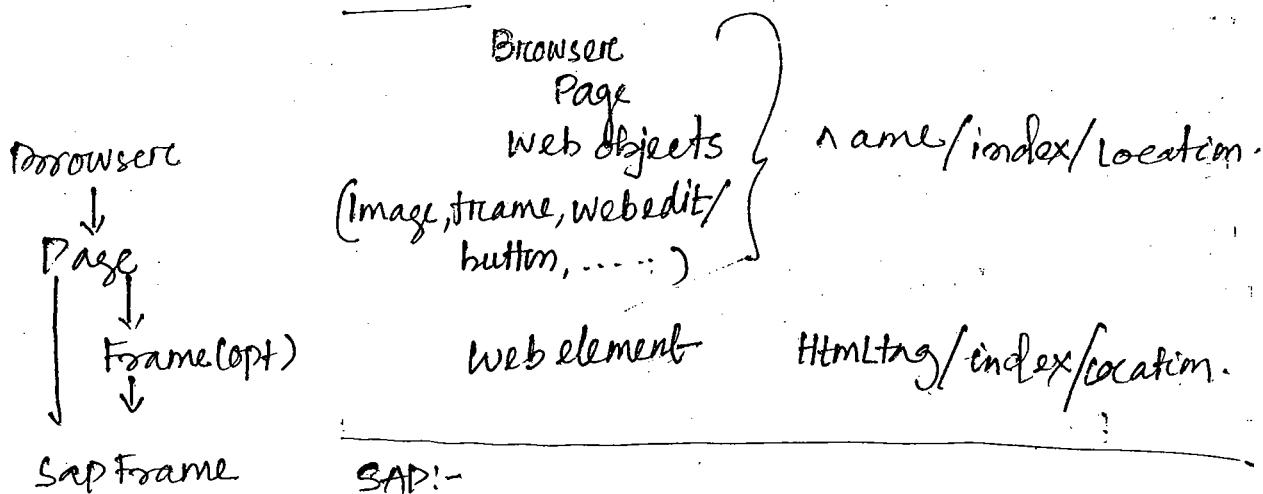
- a. Install QTP with SAP addin.
- b. Install Browser and SAP framework.
- c. Install SAP application.
- d. Launch QTP with SAP addin.
- e. Launch SAP application.

3. Object table:-

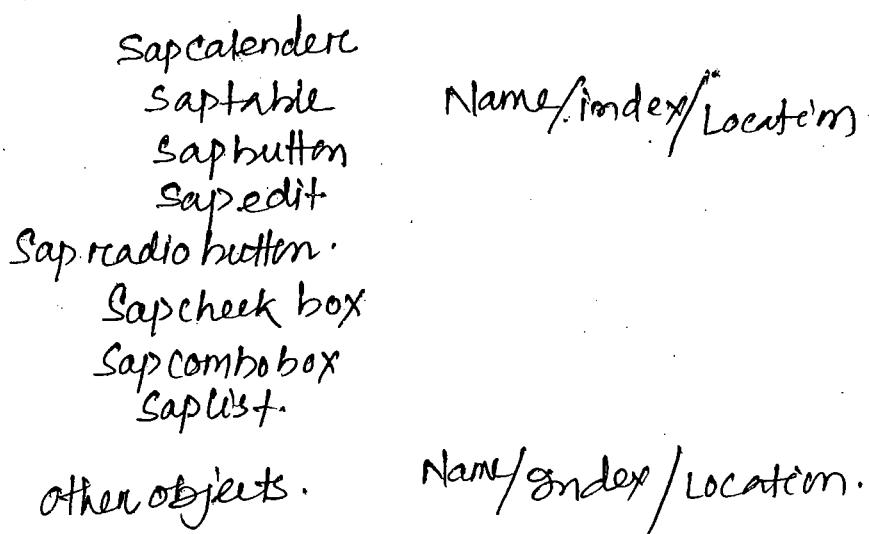
Technology Object hierarchy

Reference object Object property

HTML:-



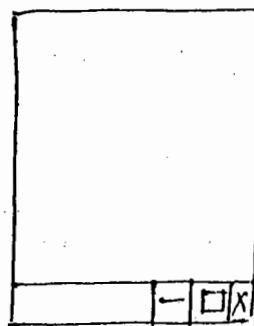
SAP:-



4. Task:-

A to Z Products.

Home
Username
Password
Login



A to Z Products.

Products
Product ID
Product name
Product Price
Quality
Amount
Total cost

1. To automate Login
2. To maximize the product page which is minimize.
3. To check if the product functionality is fine with following information.

Product ID :- 007

Quantity :- 10

Product name :- Parker Pen

Price :- 150

Note:-

Sometime during reutime the test script may encounter minimized window on which the operation are to be performed in this case to maximize the window the following syntax is used.

`x=Browser("title: = *").Object. Hwnd`

HWND - Height width naming directory

`window (" HWND := " & x) . Maximize`

```

1 → Dim ev, av, x
2 → System.util.Run "www: AtoZ Products.com"
3 → with browserc ("title:= Home") "AtoZ products")
4 →     • webradi
5 →         with Page ("title:= Home")
6 →             • Webedit ("name:= Username").set "Soidhara"
7 →             • Webedit ("name:= Password").set "Mercury"
8 →             • Webbutton ("name:= Login") • click
9 → EndWith:
10 → x = browserc ("title:= *").object.HWND
11 → window ("HWND:= " & x).maximize
12 → with Page ("title:= Products")
13 →     • sap combobox
14 →         • Webedit ("name:= ProductID").set "007"
15 →         • sap combobox
16 →             • Webedit ("name:= ProductName").set "ParcheqPen"
17 →             • sap combobox
18 →                 • Webedit ("name:= ProductPrice").set "150"
19 →             • sap combobox
20 →                 • Sapedit ("name:= quantity").set "10"
21 →             • Sapbutton ("name:= Total cost") • click
22 → ev = • Sapedit ("name:= comamount").getproperty("text")
23 → EndWith:
24 → EndWith:
25 → If ev = av Then
26 →     msgbox ("application is fine")
27 → Else
28 →     msgbox ("application is not fine")

```

F) HTML with Siebel technology:-

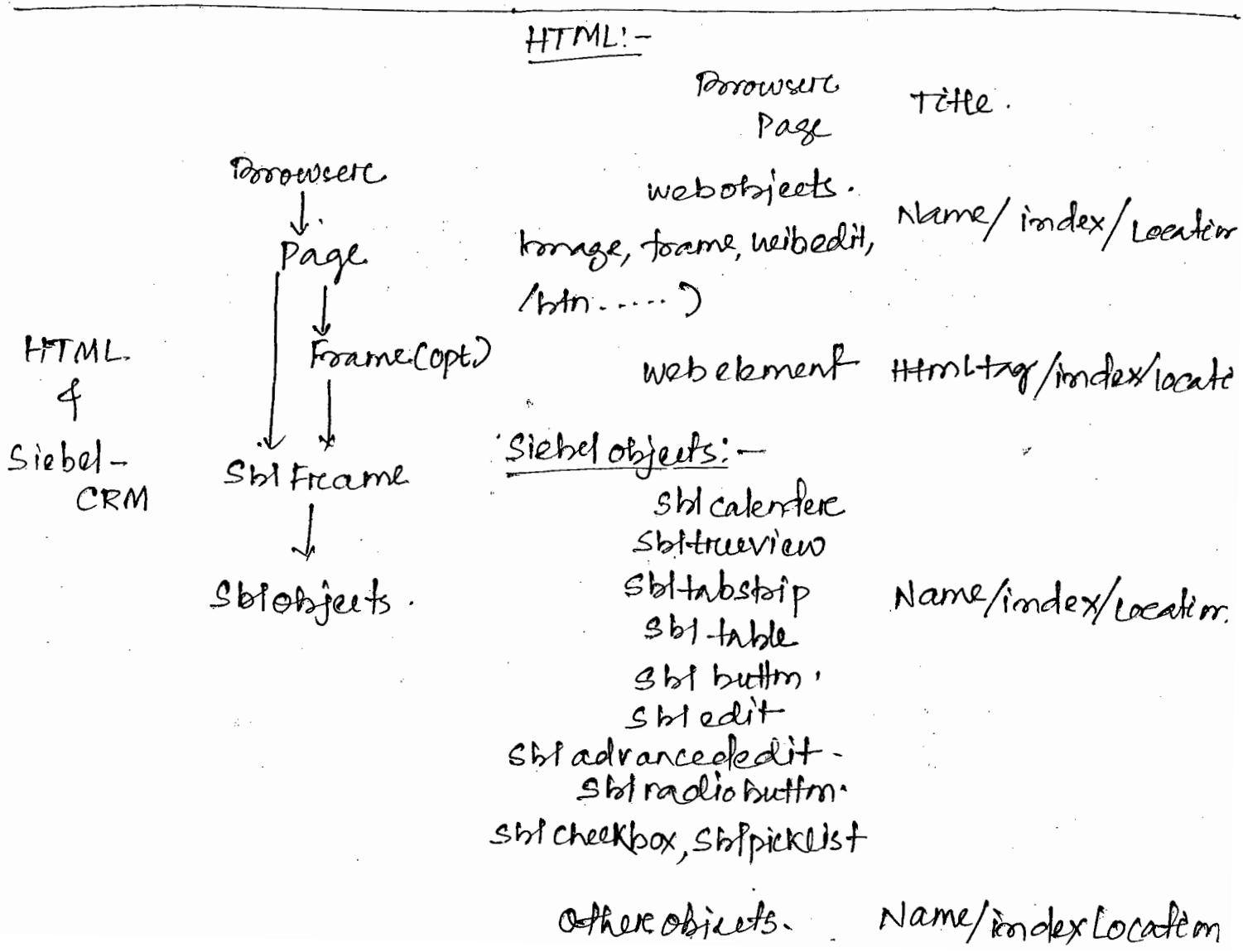
Date - 06.04.2015

① Introduction:- Foremost of the CRM applications Siebel technology is widely used. Such application must be tested with appropriate test scripts.

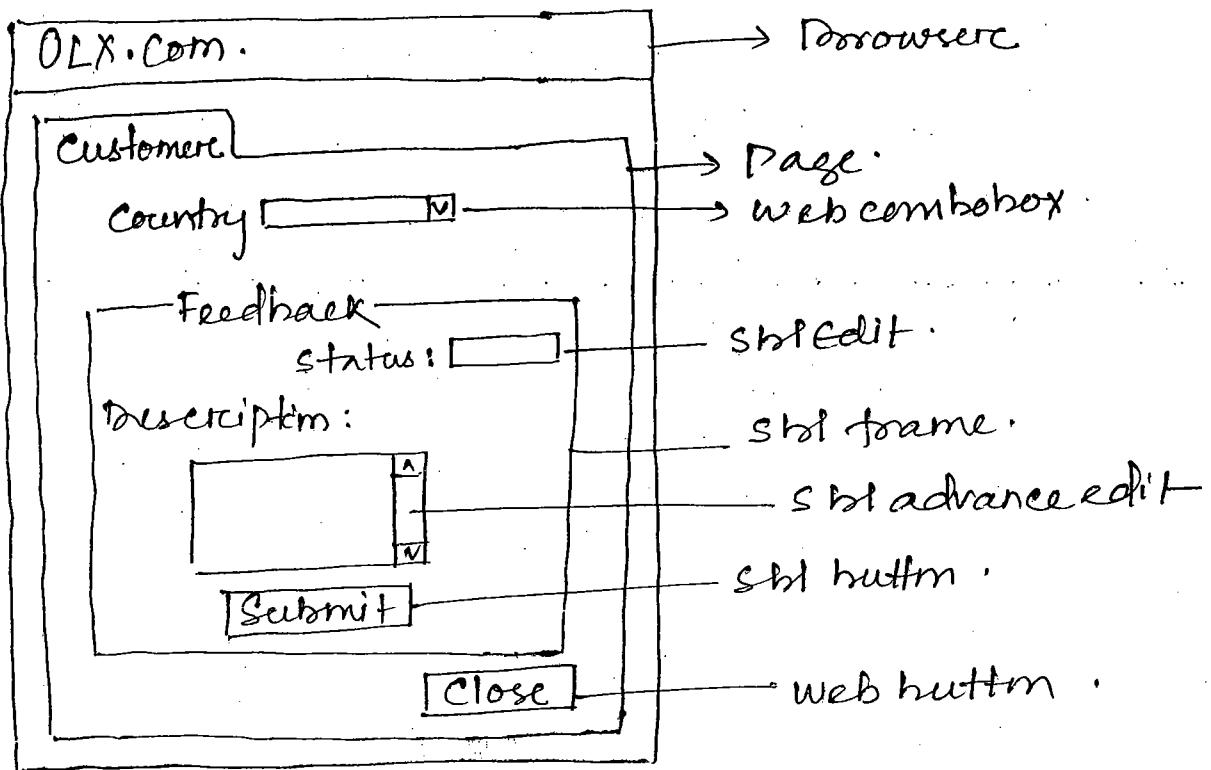
② Environment:-

- Install QTP with Siebel addin.
- Install Browser and Siebel framework.
- Install Siebel application.
- Launch QTP with Siebeladdin.
- Launch Siebel application.

③ Object table:-



1) Task:-



- ① To invoke OLX.com application.
- ② To select the country from which the feedback is send.
- ③ To create and submit feed back as
 "Product is excellent"
- ④ To check the feed back submission is successful
 based on "done" text in status field.

Ans :-

```
Dim x
Systemutil.Run "www.OLX.com"
With browser ("title:=OLX.com")
    With .page ("title:=Customer")
        .webcombobox ("name:=Country").Select ("India")
    End With
    With .SBLframe ("name:=Feedback")
        .SBLadvancededit ("name:=Description").Set
            "Product is excellent"
    End With
End With
```

```

    • Sbbutton ("name:= submit") . click.
    X = • Sbredit ("name:= status") . getproperty ("text")
End withr.

    • Close
End withr.

End withr.

If ('x=done') Then
    msgbox ("feedback Submissim is successfull")
else
    msgbox ("feed back Submissim is not sucessfull")
End else.

```

WEB SERVICES:-

① Web services testing (API testing) : -

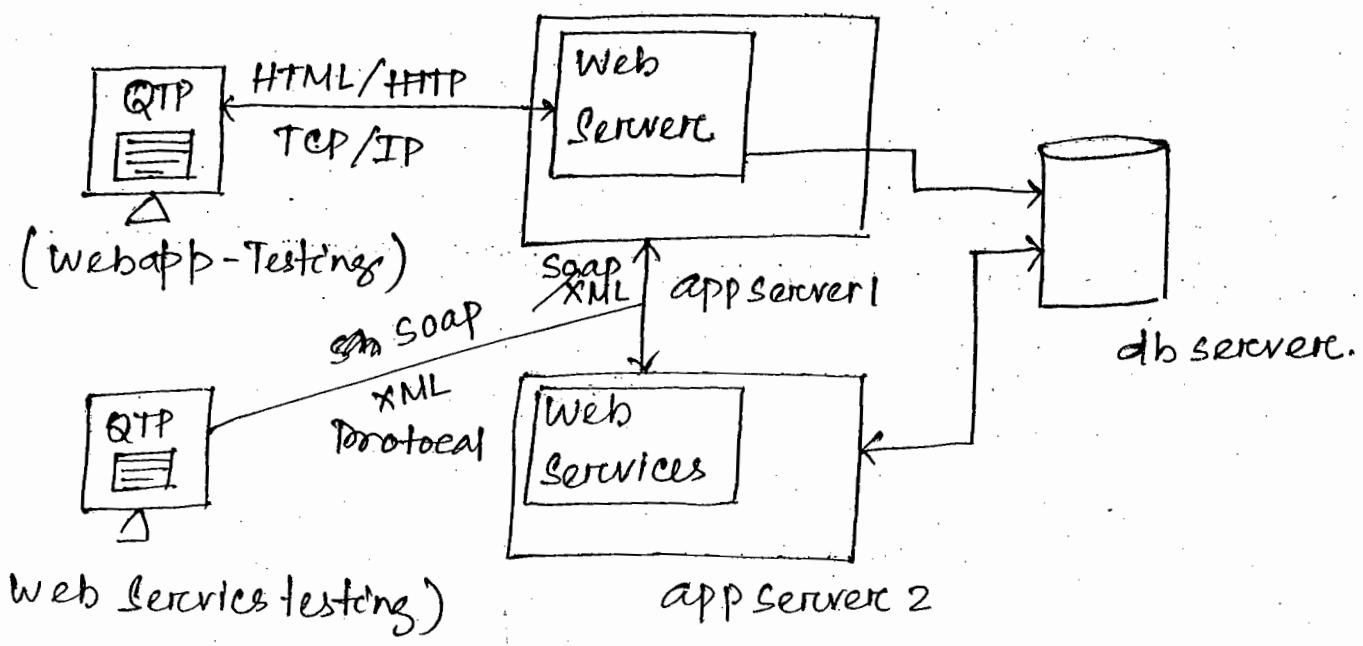
1. Introduction : -

- a. Some special services that are made available in the web server for specific utility are known as web services.
- b. Web site as well as web services all are considered to be web application.
- c. Web application will run on either 3-tier architecture or n-tier architecture.
- d. Web services will run on only on n-tier architecture.
- e. Web applications can have GUI.
- f. Web services have API Based
 - ↳ Application programming interface
- g. It is advisable for the testing team to test the web services offline rather than online.

2. Environment:-

- a. Install QTP with web service add-in.
* http://med.research\Web services\tempconvert.asmx?wsdl
asmx → active server method extention.
WSDL → Web services Description Library.
SOAP → Service oriented object access Protocol.
- b. Testing team must get the following information from the development team.
 1. URL of web service:-
http://med.research\Web services\tempconvert.asmx?wsdl
asmx →
WSDL →
 2. Service Name:- Temp
Ex:- Temp converter (Used for converting temperature)
 3. Port name:-
Temp Convert SOAP (Used for communicating with web services)
- c. Install web services in the web server.
- d. Launch QTP with web services add-in select
- e. Launch the web services to be tested.

Web Service Environment:-



Task! -

Write a program to test temp converter service provided by net research company as per the specifications given below.

- To check if " CelsiusToFahrenheit" method works fine.
- To check if " FahrenheitToCelsius" method works fine.

Ans:-

Dim X,Y,Z,CF,FC

$\begin{cases} X = \text{URL} \\ Y = \text{Temp Converter} \\ Z = \text{Port Name} \end{cases}$

$X = \text{"Http://netresearch/Webservices/tempconverter.asmx?wsdl"}$

$Y = \text{"<?xml version='1.0' encoding='utf-8'?><?tempConverter?>"}$

$Z = \text{"TempConvertSOAP"}$

~~Webservices exp, pr, & d e d e o o q u e e o~~

$CF = \text{Webservices}(X, Y, Z) \cdot \text{CelsiusToFahrenheit}("25")$

$FC = \text{Webservices}(X, Y, Z) \cdot \text{FahrenheitToCelsius}("77")$

If $(CF = 77)$ AND $(FC = 25)$ then

```
msgbox (" web service is ok")
```

Else

```
msgbox (" web service is not ok")
```

Endif.

Implementation:
Note:-

1. Collect URL, Service name and Portname from the development team .
2. Use webservices object by passing the above Parameters to it .
3. Use the appropriate method to be tested for the web services objects with the appropriate parameters .
4. Implement the testing logic and display the result ~~table~~ statement -

)) Mainframe Application Testing:-

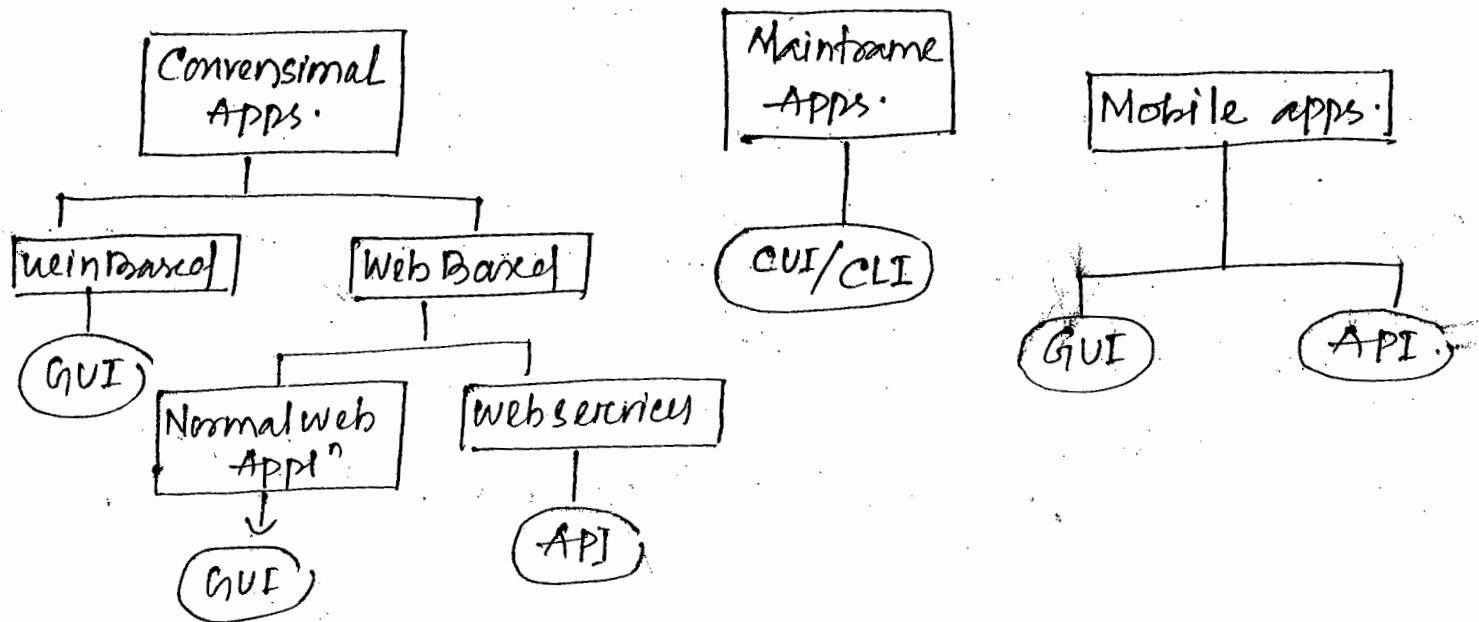
)) Introduction:-

o Mainframe is not a technology but it is a degree of capacity on which the application run they are known as mainframe application.

These application are to be tested by the tester. Usually mainframe environmental capacity is huge

- | |
|------------------------------------|
| → CUI - Character User Interface . |
| → CLI - Command Line Interface . |

b. Generation of application:-



② Environment:-

(Test environment)

a. Install QTP along with TE simulator addin.

b. ~~Install~~ Install TE simulator like

Hercules HS370, Z-Architecture
ESA390

c. Install mainframe application

d. Launch QTP with required addin.

e. Launch Mainframe application through TE simulator

③ Object Property Table

S.I.No	Tech.	Hierarchy.	Ref-objects.	Ref-Properties.
1	Main frame	TE window ↓ TE Screen. ↓ TE field	TE field	field/index/locate

Task
Write a program to automate login activity of
[SE_APP] Screen present in Sales express
application window.

Ans:-

With tewindow ("field id:= sales express").

With .tesseract ("field id:= SE_APP")

- tefield ("field id:= Username").set "Abhi"
- tefield ("field id:= Password").set "Dey3"
- tefield ("field id:= Login").click

End with.

End with.

) Mobile Application Testing:-

1. Introduction:-

- a. Application that can run on mobile devices are known as mobile application.
- b. Development team can develop mobile application with various technology that can run on several platform like Android, IOS, Symbian, WMEC window mobile environment)
- c. installing mobile applications into the actual devices is difficult job for the testing team and so they can go for work around.
- d. industry can seek the help of 3rd party cloud computing servers which can simulate mobile

e. Testing team can upload the application into these servers and try to access them for testing.

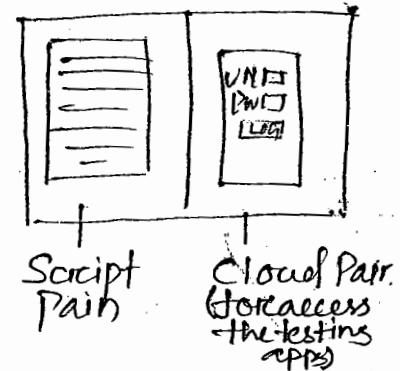
2. Environment:-

- (a) Install QTP with mobile addin (Cloud mobile
installer software)
- (b) Install cloud mobile addin installer software to make the addin visible in the addin manager
- (c) Development team can upload mobile application to the 3rd party servers for the sake of testing

- (d) Launch QTP with the addin selected

→ (3rd party servers Ex:- Perfecto mobile servers)

- (e) Once QTP is launched it will display two windows - script pain and cloud pain through which application window can be accessed

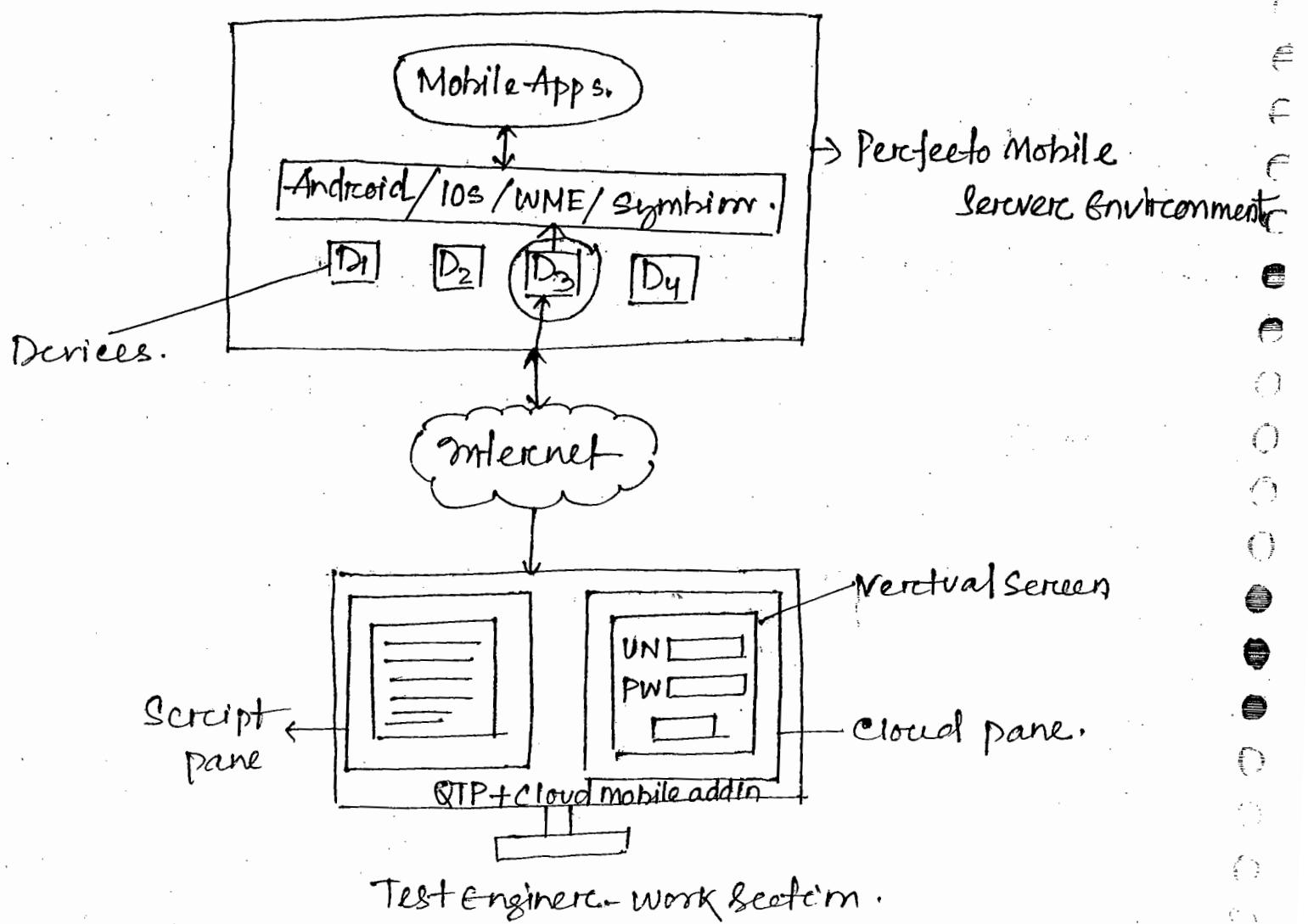


- (f) Initially the cloud pain displays the Login window which can eventually display on login the provision of selecting the devices.

- (g) After selecting a specific device the mobile application can be instal into it and the application window can be seen through the cloud page.

- (h) The tester can developed the script in the script pain and can run it to test the ~~test~~ (AUT).

Mobile environment diagram:-



Types of Mobile Application:-

Depends on the usage and purpose there are 3 types of mobile application.

1. Mobile - Native application:- these applications reside in the device. And once made available to the user to standards icons.

Ex:- calculator, calendar, Angry bird
Candy crush, ~~Facebook~~, converter.

2. Mobile - Hybrid application:- These are the combination of native as well as web application.

Ex:- Google ~~chrome~~ ^{apps}, coming as stock application.

3. Mobile web application:— These are the applications which can be executed under web environment (Ex: internet) Ex:- Facebook, youtube, twiter.

③ Object-table:-

3.1. Object property-table for native Mobile application:

Technology	Hierarchy	Reference object	Description for open file
Native mobile application	Device	device	device id
	M native button M native edit M native combobox	M native button. M native edit. M native combobox	control-label/ control index.
	M native object	M native object.	

Task:- Write a program to select new contacts, enter mobile number, enter the name and save it. where the device id = 777

Ans:-

With device ("device id:= 777")

- M native button ("control-label:= new contact").click
- M native edit ("control-label:= mobilenumber").set "9899999999"
- M native edit ("control-label:= name").set "Abhilash"
- M native button ("control-label:= save").click

End with.

- Resources.goto "Phone"

3.2/ Object table for Hybrid (native+web) application:-

Technology Hierarchy	Reference object	Description Properties
mobile hybrid application	<pre> graph TD Device[Device] --> MObjects[M objects] </pre>	device —— device id Mbutton Mredit Mcombobox } Label/index : etc.

ask!-

Write a program to get the road map or set of instruction in selecting navigate option for the input "charminar at Hyderabad" as destiny

Ans:-

```

with device ("device id:= 777")
  • application.goto "navigate"
  • Mredit ("Label := destiny").set "charminar, HYD"
  • Mbutton ("Label := OK").click
End with.
  
```

3/ Object table for Mobile web application:-

Technology Hierarchy	Reference object	Description properties
mobile web app.	<pre> graph TD Device[Device] --> MWebObjects[M web objects] </pre>	device —— device id Mwebedit Mwebbutton Mwebcombobox } Control-Label } Control-Index : etc.

Task:- write a program to automate google search operator
in the mobile web application. and to display the list
of site for QTP.

Ans:- with device ("device id:= 777")

- webBrowser . goto "www.google.com".
- mWeb edit ("control-label := q ~~QTP~~"). set "QTP"
- mWeb button ("control-label := Google Search"). click.

End with

