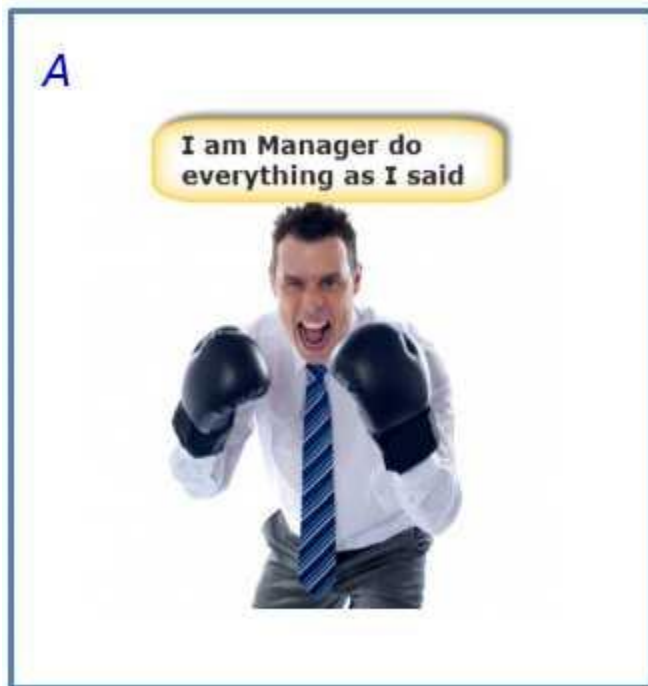# How to Create a Test Plan

Let's start with following scenario

In a meeting, you want to discuss the Test Plan with the team members, but they are not interested - .

In such case, what will you do? Select your answer as following figure

☐ A) I am Manager do everything as i said

☐ B) OK, let's me explain why we need a Test Plan

As a Test Manager, you must explain them the importance of Test Plan rather than force the team to do what you want.

Making Test Plan has multiple benefits

- Test Plan helps us determine the **effort** needed to validate the quality of the application under test
- Help people outside the test team such as developers, business managers, customers**understand** the details of testing.
- Test Plan **guides** our thinking. It is like a rule book, which needs to be followed.
- Important aspects like test estimation, test scope, test strategy are **documented** in Test Plan, so it can be reviewed by Management Team and re-used for other projects.

How to make a Test Plan

You already know that making a **Test Plan** is the most important task of Test Management Process. Follow the seven steps below to create a test plan
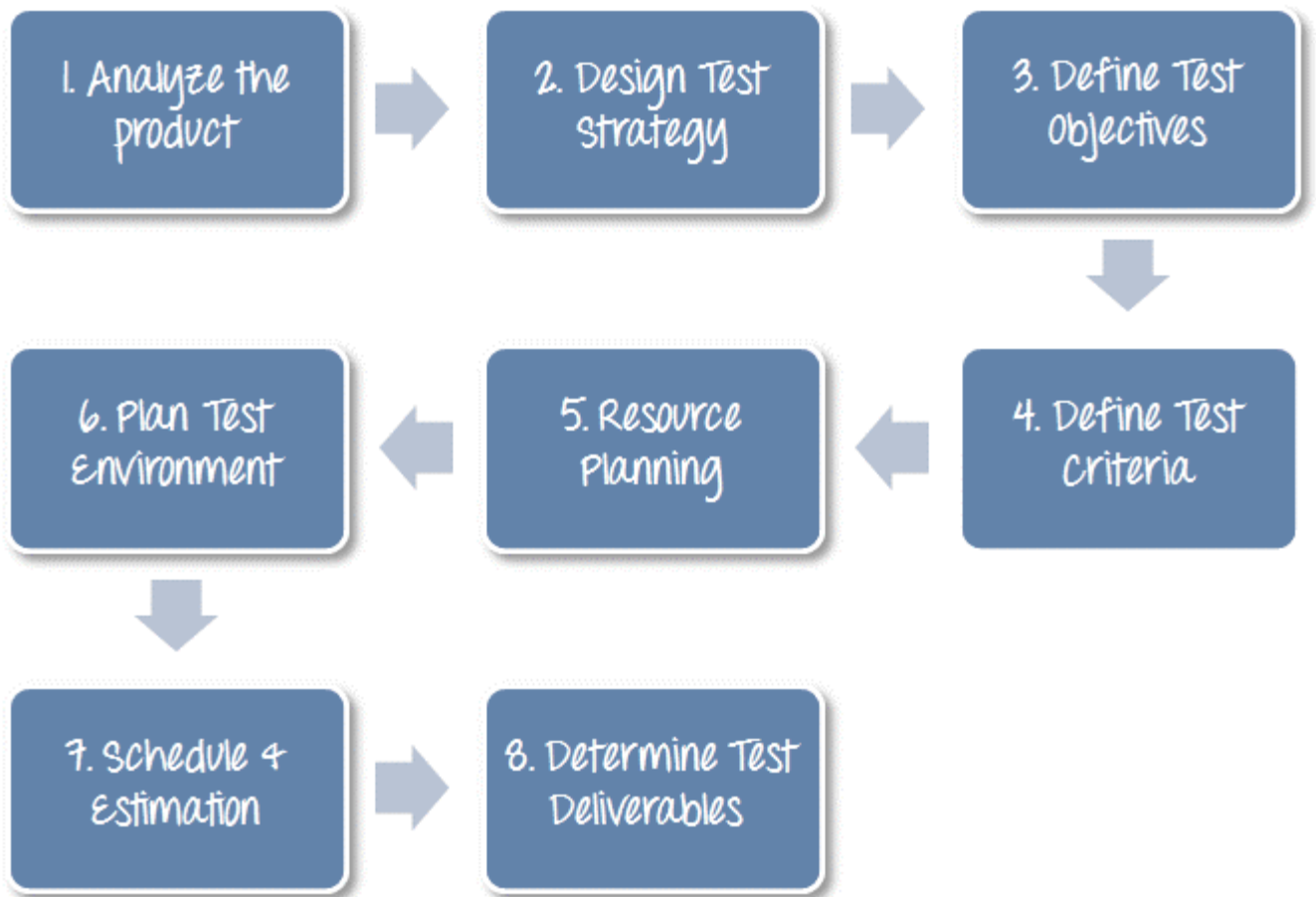
Step 1) Analyze the product

How can you test a product **without** any information about it? The answer is **Impossible.** You must learn a product **thoroughly** before testing it.

The product under test is Guru99 banking website. You should research clients and the end users to know their needs and expectations from the application

- Who will use the website?
- What is it used for?
- How will it work?
- What are software/ hardware the product uses?

You can use the following approach to analyze the site

Interview client, designer and developer

Review product and project documentation

Perform product walkthrough

Now let's apply above knowledge to a real product: **Analyze** the banking website http://demo.guru99.com/V4.

You should take a **look around** this website and also **review** product documentation. Review of product documentation helps you to understand all the features of the website as well as how to use it. If you are unclear on any items, you might **interview** customer, developer, designer to get more information.

Step 2) Develop Test Strategy

Test Strategy is a **critical step** in making a Test Plan. A Test Strategy document, is a high level document, which is usually developed by Test Manager. This document defines:

* The project's **testing objectives** and the means to achieve them
* Determines testing **effort** and **costs**

Back to your project, you need to develop Test Strategy for testing that banking website. You should follow steps below

Step 2.1) Define Scope of Testing

Before the start of any test activity, scope of the testing should be known. You must think hard about it.

- The components of the system to be tested (hardware, software, middleware, etc.) are defined as "**in scope**"
- The components of the system that will not be tested also need to be clearly defined as being "**out of scope**."

Defining the scope of your testing project is very important for all stakeholders. A precise scope helps you

- Give everyone a **confidence & accurate information** of the testing you are doing
- All project members will have a **clear** understanding about what is tested and what is not

How do you determine scope your project?

To determine scope, you must –

- Precise customer requirement
- Project Budget
- Product Specification
- Skills & talent of your test team

Now should clearly define the "in scope" and "out of scope" of the testing.

- As the software requirement specs, the project Guru99 Bank only focus on testing all the **functions** and external interface of website **Guru99** Bank (**in scope** testing)
- Non functional testing such as **stress**, **performance** or **logical database** currently will not be tested. (**out of**scope)

Problem Scenario

The customer wants you to test his API. But the project budget does not permit to do so. In such a case what will you do?

Well, in such case you need to convince the customer that API testing is extra work and will consume significant resources. Give him data supporting your facts. Tell him if API testing is included in-scope the budget will increase by XYZ amount.

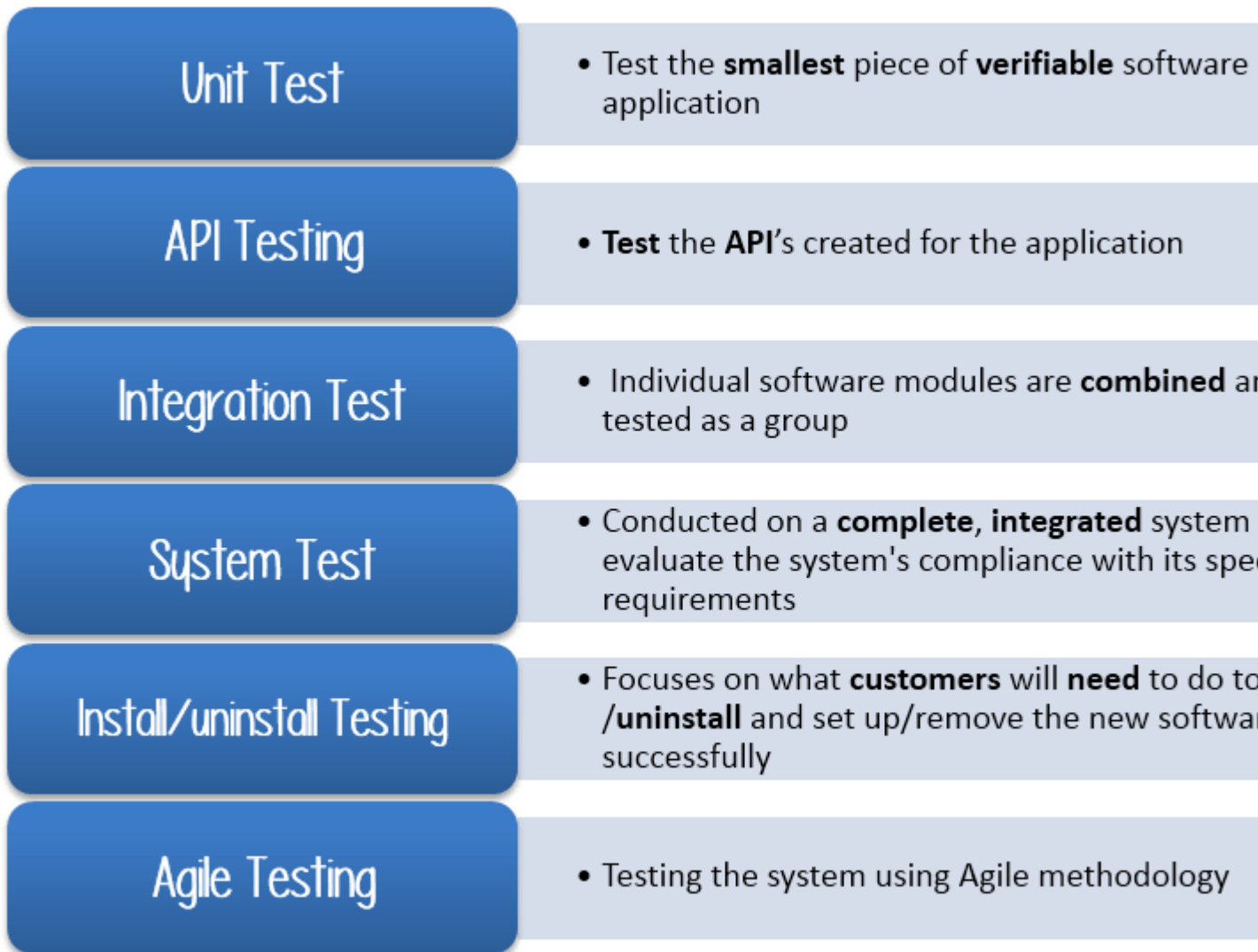The customer agrees and accordingly the new scopes, out of scope items are

- In-scope items: Functional Testing, API Testing
- Out of scope items: <u>Database testing</u>, hardware & any other external interfaces

Step 2.2) Identify Testing Type

A **Testing Type** is a standard test procedure that gives an expected test outcome.

Each testing type is formulated to identify a specific type of product bugs. But, all Testing Types are aimed at achieving one common goal "**Early detection of** all the defects before releasing the product to the customer"

The **commonly used** testing types are described as following figure

Unit Test	• Test the **smallest** piece of **verifiable** software application
API Testing	• **Test** the **API**'s created for the application
Integration Test	• Individual software modules are **combined** a tested as a group
System Test	• Conducted on a **complete, integrated** system evaluate the system's compliance with its spe requirements
Install/uninstall Testing	• Focuses on what **customers** will **need** to do to /**uninstall** and set up/remove the new softwa successfully
Agile Testing	• Testing the system using Agile methodology

There are **tons of Testing Types** for testing software product. Your team **cannot have** enough efforts to handle all kind of testing. As Test Manager, you must set **priority** of the Testing Types

- Which Testing Types should be **focused** for web application testing?
- Which Testing Types should be **ignored** for saving cost?

Now let's practice with your project. The product you want to test is a banking website.

Which Testing Types should you focus in this case?

Select All that Apply

☐ A) Unit testing

☐ B) API Testing

☐ C) Integration Testing

☐ D) System Testing

☐ E) Install/Uninstall Testing

☐ F) Agile testing

Step 2.3) Document Risk & Issues

Risk is future's **uncertain event** with a probability of **occurrence** and a **potential** for loss. When the risk actually happens, it becomes the '**issue'.**

In the article Risk Analysis and Solution, you have already learned about the 'Risk' analysis in detail and identified potential risks in the project.

In the Test Plan, you will document those risks

Risk	Mitigation
Team member lack the required skills for website testing.	Plan **training course** to skill up your members
The project schedule is too tight; it's hard to complete this project on time	Set**Test Priority** for each of the test activity.
Test Manager has poor management skill	Plan leadership training for manager
A lack of cooperation negatively affects your employees' productivity	**Encourage** each team member in his task, **and inspire** them
Wrong budget estimate and cost overruns	Establish the scope before beginning work, pay a lot of atte... planning and constantly track and measure the progress

Step 2.4) Create Test Logistics

In Test Logistics, the Test Manager should answer the following questions:

- **Who** will test?
- **When** will the test occur?

Who will test?

You may not know exact names of the tester who will test, but the **type of tester** can be defined.

To select the right member for specified task, you have to consider if his skill is qualified for the task or not, also estimate the project budget. Selecting wrong member for the task may cause the project to**fail** or **delay**.

Person having the following skills is most ideal for performing software testing:

- Ability to **understand** customers point of view
- Strong **desire** for quality
- **Attention**to detail

- Good **cooperation**

In your project, the member who will take in charge for the test execution is the **tester.** Base on the project budget, you can choose in-source or out-source member as the tester.

When will the test occur?

Test activities must be matched with associated development activities.

You will start to test when you have **all required items** shown in following figure

Step 3) Define Test Objective

Test Objective is the overall goal and achievement of the test execution. The objective of the testing is finding as many software defects as possible; ensure that the software under test is **bug free** before release.

To define the test objectives, you should do 2 following steps

1. List all the software features (functionality, performance, GUI…) which may need to test.
2. Define the **target** or the **goal** of the test based on above features

Let's apply these steps to find the test objective of your Guru99 Bank testing project

You can choose the 'TOP-DOWN' method to find the website's features which may need to test. In this method, you break down the application under test to **component** and **sub-component**.

In the previous topic, you have already analyzed the requirement specs and walk through the website, so you can create a **Mind-Map** to find the website features as following
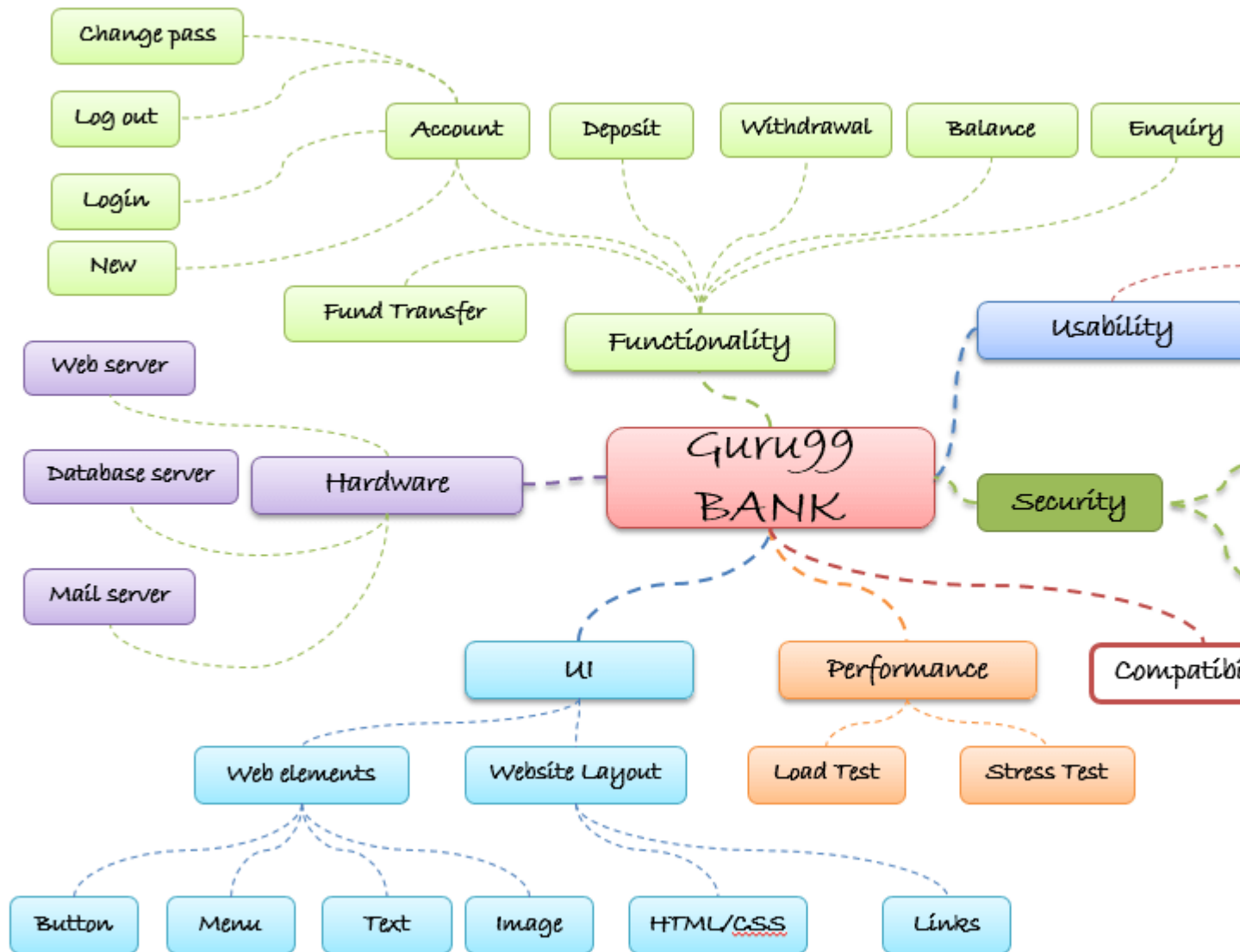
This figure shows all the features which the Guru99 website may have.

Based on above features, you can define the Test Objective of the project Guru99 as following

- Check that whether website Guru99 **functionality**(Account, Deposit…) is working as expected without any error or bugs in real business environment
- Check that the external interface of the website such as **UI** is working as expected and & meet the customer need
- Verify the **usability** of the website. Are those functionalities convenient for user or not?

Step 4) Define Test Criteria

Test Criteria is a standard or rule on which a test procedure or test judgment can be based. There're 2 types of test criteria as following

Suspension Criteria

Specify the critical suspension criteria for a test. If the suspension criteria are met during testing, the active test cycle will be **suspended** until the criteria are **resolved**.

Example: If your team members report that there are **40%** of test cases failed, you should **suspend** testing until the development team fixes all the failed cases.
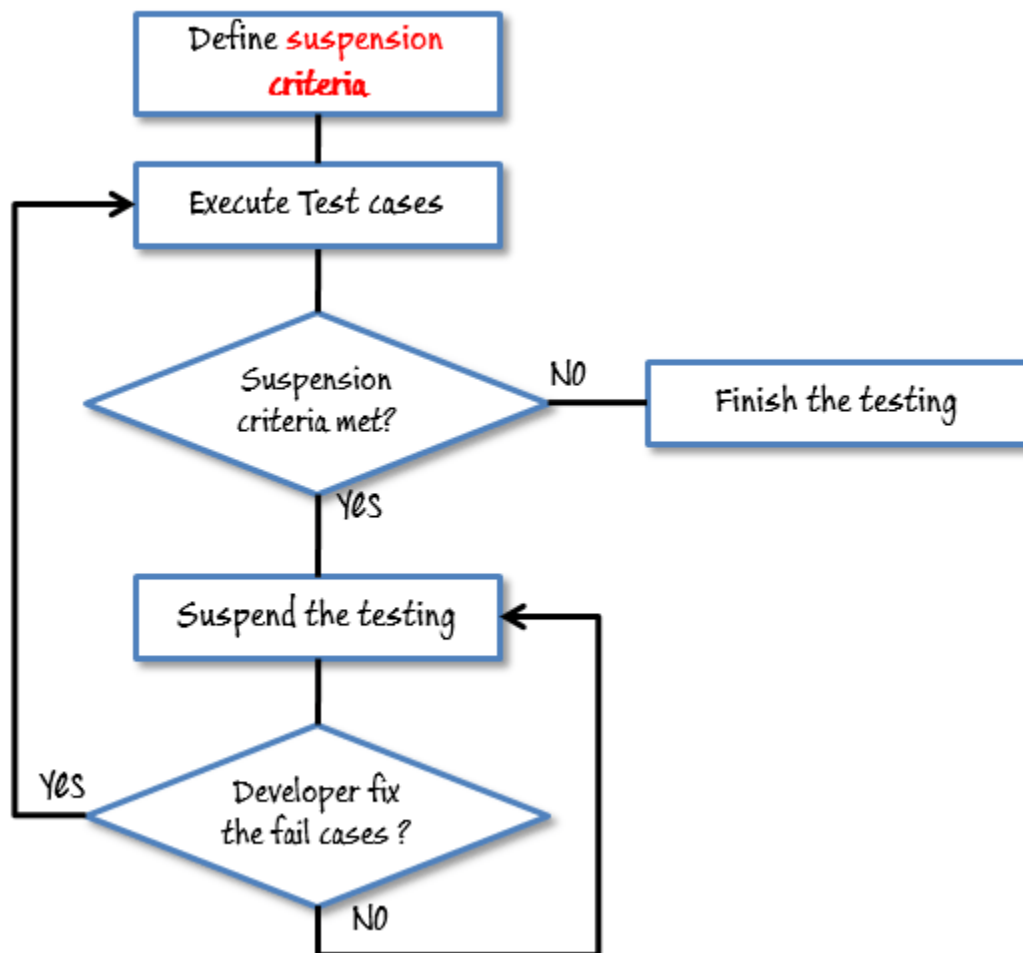
Exit Criteria

It specifies the criteria that denote a **successful** completion of a test phase. The exit criteria are the targeted results of the test and are necessary before proceeding to the next phase of development. Example: **95%** of all critical test cases must pass.

Some methods of defining exit criteria are by specifying a targeted **run rate** and **pass rate**.

- Run rate is ratio between **number test cases executed/total test cases** of test specification. For example, the test specification has total 120TCs, but the tester only executed 100 TCsàthe run rate is $100/120 = 0.83$ (83%)
- Pass rate is ratio between **numbers test cases passed / test cases executed**. For example, in above 100 TCs executed, there're 80TCs passedàthe pass rate is $80/100 = 0.8$ (80%)

This data can be retrieved in Test Metric documents.

- **Run** rate is mandatory to be **100%** unless a clear reason is given.
- **Pass** rate is dependent on project scope, but **achieving high pass rate** is a goal.

Example: Your Team has already done the test executions. They report the test result to you, and they want you to confirm the **Exit Criteria.**

In above case, the Run rate is mandatory is **100%,** but the test team only completed 90% of test cases. It means the Run rate is not satisfied, so do NOT confirm the Exit Criteria

Step 5) Resource Planning

Resource plan is a **detailed summary** of all types of resources required to complete project task. Resource could be human, equipment and materials needed to complete a project

The resource planning is important factor of the test planning because helps in **determining** the **number** of resources (employee, equipment…) to be used for the project. Therefore, the Test Manager can make the correct schedule & estimation for the project.

This section represents the recommended resources for your project.

Human Resource

The following table represents various members in your project team

No.	Member	Tasks
1.	Test Manager	**Manage** the whole project Define project **directions** Acquire appropriate resources
2.	Tester	Identifying and describing appropriate test technique architecture Verify and assess the Test Approach **Execute** the tests, **Log** results, **Report** the defects. Tester could be in-sourced or out-sourced members, budget For the task which required **low** skill, I recommend y choose **outsourced** members to **save** project cost.
3.	Developer in Test	**Implement** the test cases, test program, test suite etc.
4.	Test Administrator	Builds up and ensures test environment and assets are **managed** and **maintained** **Support** Tester to use the test environment for test ex
5.	SQA members	Take in charge of quality assurance Check to confirm whether the testing process is mee requirements

System Resource

For testing, a web application, you should plan the resources as following tables:

No.	Resources	Descriptions
1.	Server	Install the web application under test This includes a separate web server, database server, and application server if applicable
2.	Test tool	The testing tool is to automate the testing, simulate the user operation, generate the test r There are tons of test tools you can use for this project such as Selenium, QTP…etc.
3.	Network	You need a Network include LAN and Internet to simulate the real business and user env
4.	Computer	The PC which users often use to connect the web server

Step 6) Plan Test Environment

What is the Test Environment

A testing environment is a setup of software and hardware on which the testing team is going to execute test cases. The test environment consists of **real business** and **user** environment, as well as physical environments, such as server, front end running environment.

How to setup the Test Environment

Back to your project, how do you set up **test environment** for this banking website?

To finish this task, you need **a strong cooperation** between Test Team and Development Team

You should ask the developer some questions to understand the web application under test**clearly**. Here're some recommended questions. Of course, you can ask the other questions if you need.

- What is the maximum user connection which this website can handle at the same time?
- What are hardware/software requirements to install this website?
- Does the user's computer need any particular setting to browse the website?

Following figure describes the test environment of the banking website www.demo.guru99.com/V4

User

Web Browser

Web Browser

Web Browser

Web Browser

Internet

Web server
Guru99 Bankir
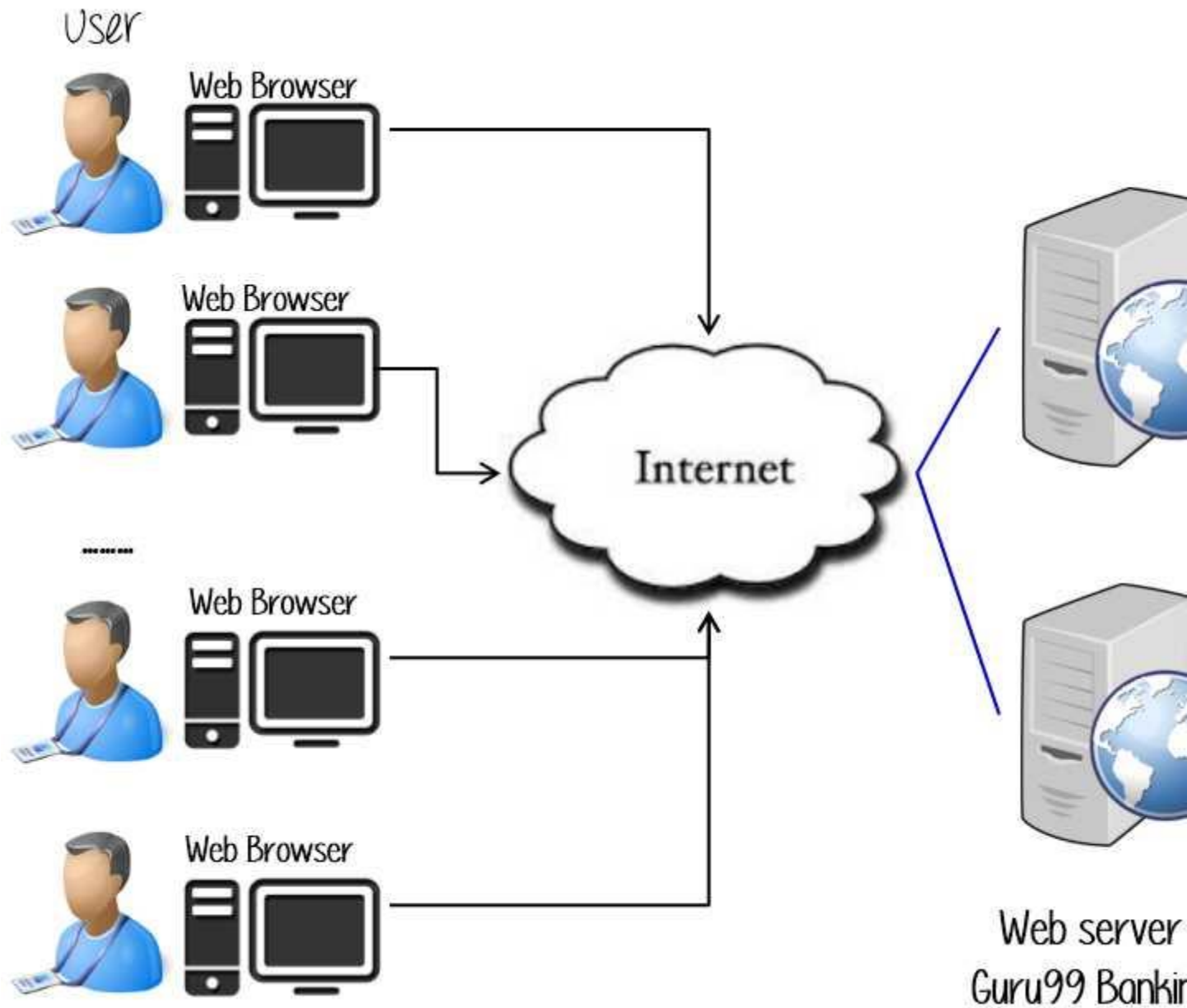
Step 7) Schedule & Estimation

In the article Test estimation, you already used some techniques to estimate the effort to complete the project. Now you should include that estimation as well as the schedule to the Test Planning

In the Test Estimation phase, suppose you break out the whole project into small tasks and add the estimation for each task as below

Task	Members	Estimate
Create the test specification	Test Designer	170 man-
Perform Test Execution	Tester, Test Administrator	80 man-h
Test Report	Tester	10 man-h
Test Delivery		20 man-h
Total		**280 man-**

Then you create the **schedule** to complete these tasks.

Making schedule is a common term in project management. By creating a solid schedule in the Test Planning, the Test Manager can use it as tool for monitoring the project progress, control the cost overruns.

To create the project schedule, the Test Manager needs several types of input as below:

- **Employee and project deadline**: The working days, the project deadline, resource availability are the factors which affected to the schedule
- **Project estimation**: Base on the estimation, the Test Manager knows how long it takes to complete the project. So he can make the appropriate project schedule
- **Project Risk** : Understanding the risk helps Test Manager add enough extra time to the project schedule to deal with the risks

Let's practice with an example:

Suppose the boss wants to complete the project Guru99 in **one** month, you already estimated the effort for each tasks in Test Estimation. You can create the schedule as below

Task Name	Start Date	End Date	Feb 2							Feb 9							Feb 16			
			M	T	W	T	F	S	S	M	T	W	T	F	S	S	M	T	W	T
Making Test specification	02/03/14	02/07/14																		
Milestone	02/07/14	02/07/14																		
Perform Test Execution	02/10/14	02/19/14																		
Milestone	02/26/14	02/26/14																		
Test Report	02/20/14	02/25/14																		
Milestone	02/25/14	02/25/14																		
Test Delivery	02/26/14	02/27/14																		
Milestone	02/27/14	02/27/14																		

Step 8) Test Deliverables

Test Deliverables is a list of all the documents, tools and other components that has to be developed and maintained in support of the testing effort.

There are different test deliverables at every phase of the software development lifecycle.
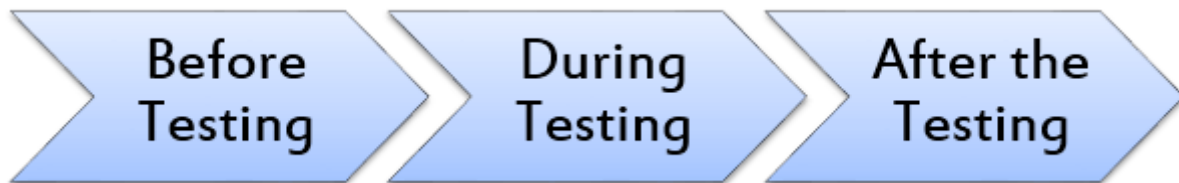
Test deliverables are provided **before** testing phase.

- Test plans document.
- Test cases documents
- Test Design specifications.

Test deliverables are provided **during** the testing

- Test Scripts
- Simulators.
- Test Data
- Test Trace-ability Matrix
- Error logs and execution logs.

Test deliverables are provided **after** the testing cycles is over.

- **Test Results/reports**
- Defect Report
- Installation/ Test procedures guidelines
- **Release notes**