**Roll # 49 , 35 , 27 , 28**

software assigment

Contents

SOFTWARE PROJECT MANAGEMENT

SOFTWARE QUALITY ASSURANCE

SOFTWARE TESTING

# SOFTWARE PROJECT MANAGEMENT

Software project management refers to the branch of project management dedicated to the planning, scheduling, resource allocation, execution, tracking and delivery of software and web projects.

Project management in software engineering is distinct from traditional project management in that software projects have a unique lifecycle process that requires multiple rounds of testing, updating, and customer feedback. Most IT-related projects are managed in the [agile style](https://www.wrike.com/project-management-guide/faq/what-is-agile-methodology-in-project-management/), in order to keep up with the increasing pace of business, and iterate based on customer and stakeholder feedback.

## RULES & RESPONSIBILTY

software project managers may have to do any of the following tasks:

Planning: This means putting together the blueprint for the entire project from ideation to fruition. It will define the scope, allocate necessary resources, propose the timeline, delineate the plan for execution, lay out a communication strategy, and indicate the steps necessary for testing and maintenance.

Leading: A software project manager will need to assemble and lead the project team, which likely will consist of developers, analysts, testers, graphic designers, and technical writers. This requires excellent communication, people and leadership skills.

Execution: The project manager will participate in and supervise the successful execution of each stage of the project. This includes monitoring progress, frequent team check-ins and creating status reports.

Time management: Staying on schedule is crucial to the successful completion of any project, but it’s particularly challenging when it comes to managing software projects because changes to the original plan are almost certain to occur as the project evolves. Software project managers must be experts in risk management and contingency planning to ensure forward progress when roadblocks or changes occur.

Budget: Like traditional project managers, software project managers are tasked with creating a budget for a project, and then sticking to it as closely as possible, moderating spend and re-allocating funds when necessary.

Maintenance: Software project management typically encourages constant product testing in order to discover and fix bugs early, adjust the end product to the customer’s needs, and keep the project on target. The software project manager is responsible for ensuring proper and consistent testing, evaluation and fixes are being made.

## Manage software project management

A recent article in Forbes [suggests](https://www.forbes.com/sites/forbestechcouncil/2017/05/15/eight-essential-software-developer-management-tips/) that there are eight ways to improve and streamline the software project management process; these eight suggestions include:

Take non-development work off your team’s plate to let them focus on developing

Motivating your team by sharing others’ success stories—like those of tech giants, which will inspire and excite your team

Avoid altering the task once its assigned

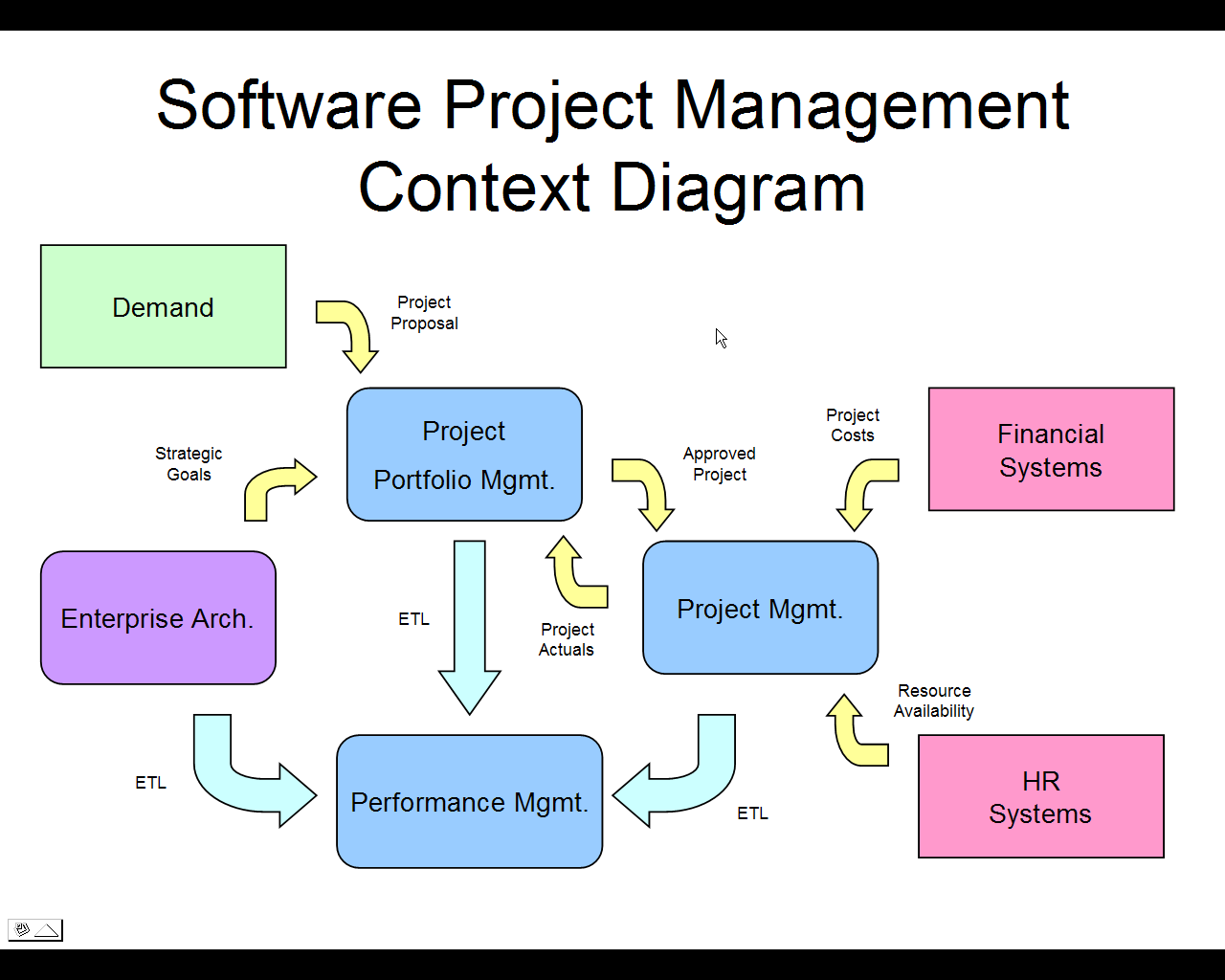
Try to stick to the plan (until it needs to be changed)

Encouraging organization by being organized

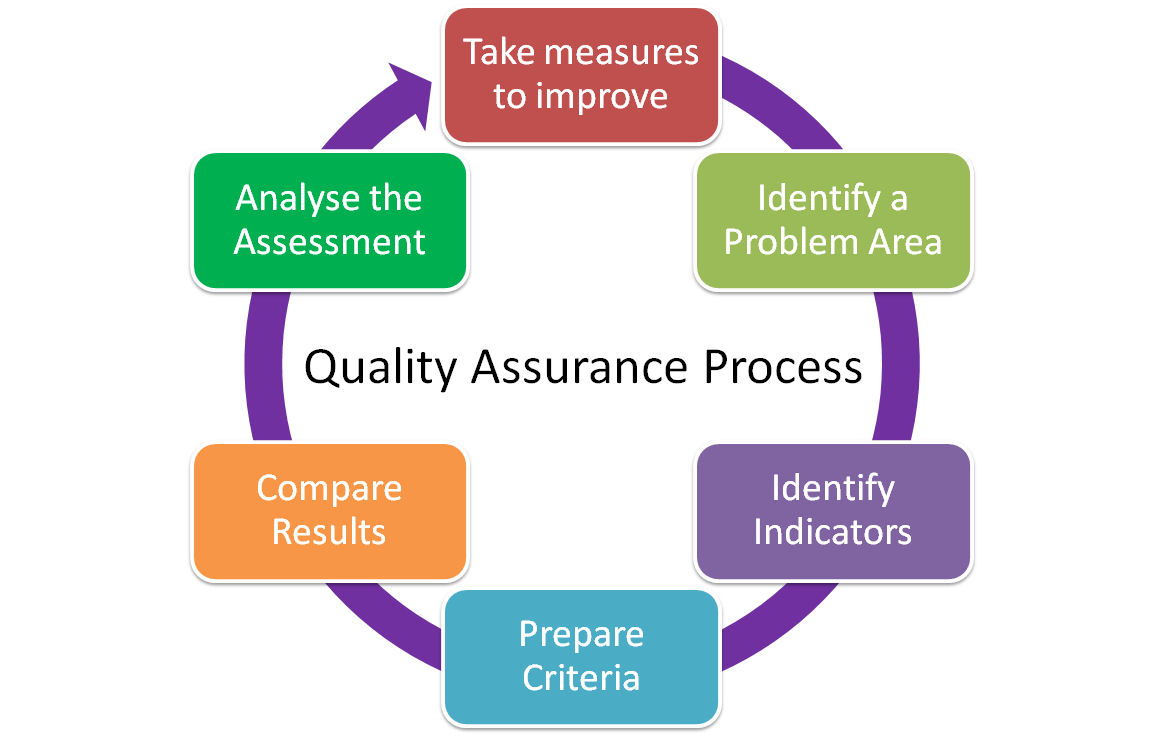
Streamline productivity through effective delegation

Get to know your team and build a rapport

Break down the plan and give them specific daily tasks



# SOFTWARE QUALITY ASSURANCE

* **Software quality assurance (SQA)** is a process which assures that all software engineering processes, methods, activities and work items are monitored and comply against the defined standards. These defined standards could be one or a combination of any like ISO 9000, CMMI model, ISO15504, etc.
* It’s a process to assure the software projects quality
* It’s a process to assure the software projects quality
* SQA incorporates all software development processes starting from defining requirements to coding until release. Its prime goal is to ensure quality.
* Software Quality Assurance Plan
* Abbreviated as SQAP, the software quality assurance plan comprises of the procedures, techniques, and tools that are employed to make sure that a product or service aligns with the requirements defined in the SRS(software requirement specification).
* The plan identifies the SQA responsibilities of a team, lists the areas that need to be reviewed and audited. It also identifies the SQA work products.
* **The SQA plan document consists of the below sections:**
* Purpose section
* Reference section
* Software configuration management section
* Problem reporting and corrective action section
* Tools, technologies and methodologies section
* Code control section
* Records: Collection, maintenance and retention section
* Testing methodology
* SQA Activities
* ***Given below is the list of SQA activities:***
* **#1) Creating an SQA Management Plan:**
* The foremost activity includes laying down a proper plan regarding how the SQA will be carried out in your project.
* Along with what SQA approach you are going to follow, what engineering activities will be carried out, and it also includes ensuring that you have a right talent mix in your team.
* **#2) Setting the Checkpoints:**
* The SQA team sets up different checkpoints according to which it evaluates the quality of the project activities at each checkpoint/project stage. This ensures regular quality inspection and working as per the schedule.
* 

# SOFTWARE TESTING

* Software testing is defined as an activity to check whether the actual results match the expected results and to ensure that the software system is[Defect](https://www.guru99.com/defect-management-process.html)free. It involves execution of a software component or system component to evaluate one or more properties of interest.
* Software testing also helps to identify errors, gaps or missing requirements in contrary to the actual requirements. It can be either done manually or using automated tools. Some prefer saying Software testing as a [White Box](https://www.guru99.com/white-box-testing.html) and [Black Box Testing](https://www.guru99.com/black-box-testing.html).
* In simple terms, Software Testing means Verification of Application Under Test (AUT).
* This tutorial introduces testing software to the audience and justifies it's importance.

IMPORTANCE

* Testing is important because software bugs could be expensive or even dangerous. Software bugs can potentially cause monetary and human loss, and history is full of such examples.
* In April 2015, Bloomberg terminal in London crashed due to software glitch affected more than 300,000 traders on financial markets. It forced the government to postpone a 3bn pound debt sale.
* Nissan cars have to recall over 1 million cars from the market due to software failure in the airbag sensory detectors. There has been reported two accidents due to this software failure.
* Starbucks was forced to close about 60 percent of stores in the U.S and Canada due to software failure in its POS system. At one point store served coffee for free as they unable to process the transaction.
* Some of the Amazon’s third party retailers saw their product price is reduced to 1p due to a software glitch. They were left with heavy losses.

TYPES

* Typically Testing is classified into three categories.
* Functional Testing
* Non-Functional Testing or [Performance Testing](https://www.guru99.com/performance-testing.html)
* Maintenance (Regression and Maintenance)
* Vulnerability in Window 10. This bug enables users to escape from security sandboxes through a flaw in the win32k system.
* In 2015 fighter plane F-35 fell victim to a software bug, making it unable to detect targets correctly.
* 