THE BENEFITS OF SOFTWARE AUTOMATION

The manual software testing by individuals forces organizations to choose between:

- cost,
- user experience, and
- time to market

HOW TO MANAGE SOFTWARE AUTOMATION

There are three strategies:

1. **Continuous integration (CI):**

Focuses on integrating work from individual developers into a main repository multiple times a day to catch integration bugs early and accelerate collaborative development.

2. **Continuous delivery (CD):**

Is concerned with reducing friction in the deployment or release process, automating the steps required to deploy a build so that code can be released safely at any time.

3. **Continuous deployment:**

Takes this one step further by automatically deploying each time a code change is made.

THE BENEFITS OF CI-CD

- Smaller code changes are simpler and have fewer unintended consequences.

- Fault isolation is simpler and quicker.

- Meantime to resolution (MTTR) is shorter because of the smaller code changes and quicker fault isolation.

Testability improves due to smaller, specific changes.

These smaller changes allow more accurate tests.

Elapsed time to detect and correct production escapes is shorter leading to a faster rate of release.

THE COST OF QUALITY

Every developer knows that no bug fix is free.

From the cost of labor to the costs of licensing, lab, energy, and deployment, ensuring the quality of an application can send thousands of dollars down the drain.

And we are not talking of hours of productivity.

Automated testing using CI/CD drives the real cost of quality down.

CI/CD FUTURE

When it comes to software development, several changes have taken place.

Be it reviewing the code, testing it, ensuring confidentiality and security, and deployment, the automation process reduces the effort of development and deployment teams considerably.

The CI/CD and DevOps trends will continue to evolve, leaving space for the market to grow and improve.