Fundamentals & Benefits of CI/CD

Fundamentals of CI/CD

CI/CD consists of 3 major Key Terms

Continuous Delivery

software development mindset where code changes are automatically prepared for a release to production. Continuous delivery expands upon continuous integration by deploying all code changes to a testing environment and a production environment after the build stage. When properly implemented, developers will always have a deployment-ready build artifact that has passed through a standardized test process.

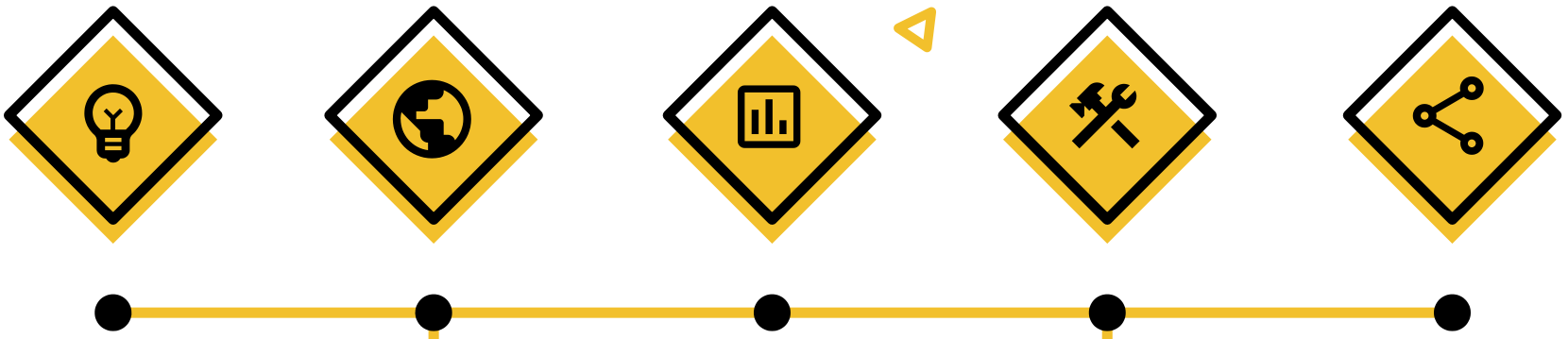
Continuous Integration

The practice of merging all developers' working copies to a shared mainline several times a day.

Continuous Deployment

A software engineering approach in which the value is delivered frequently through automated deployments without human intervention.

Implementing CI/CD Benefits



Cost Reduction as it will result in less developer time on issues from new developer code

Less bugs in production and less time in testing leading to avoiding costs

Avoid costs by preventing embarrassing or costly security holes

Cost Reduction from less Infrastructure costs from unused resources

Less human error, Faster deployments leading to Avoiding Costs

Implementing CI/CD Benefits

Increase Revenue from new value-generating features released more quickly

Increased Revenue due to less time on market

Reduced downtime from a deploy-related crash or major bug leading to Revenue Protection

Protected Revenue as a result of quick undo to return production to working state

“Warning Signs” to switch to CI/CD

Investing more time in a release cycle than delivering value

Code gets lost because of botched merges

Deployments contribute to schedule slip

Only one engineer can deploy a system