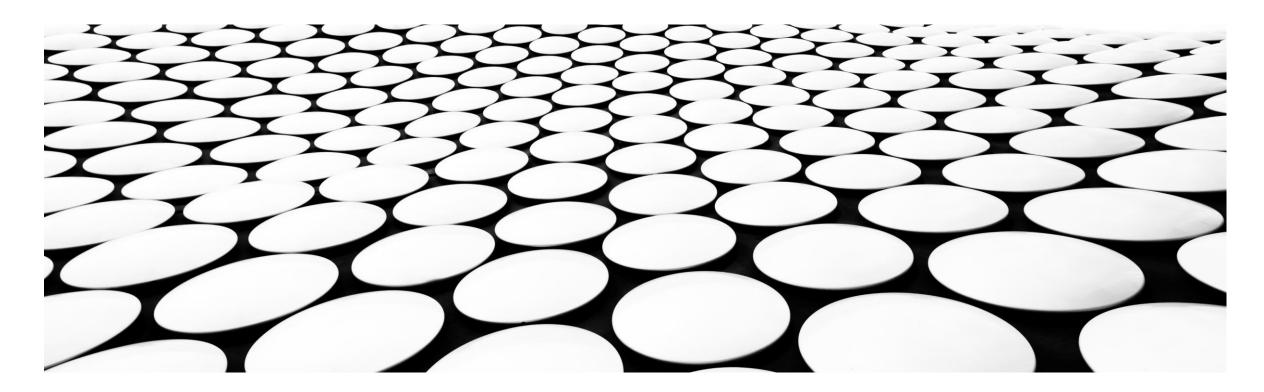
UDAPEOPLE

FUTURE IS HERE



COMPANY CURRENT STATE AND FUTURE ASPIRATION

Revenue Turnover < 2 billion
Unhappy clients
Delay in feature rollout to Market
Competitors way ahead of competition
No continuous delivery at the moment
Unwanted costs too high
Customer base < 10k
Markets - Asia

Revenue Turnover > 3 Billion
Satisfied clients
Features rollouts every week
Be the Market leader
CI/CD enablement
Unwanted costs reduced by 50%
Customer base > 25k
Markets - Asia Europe, America

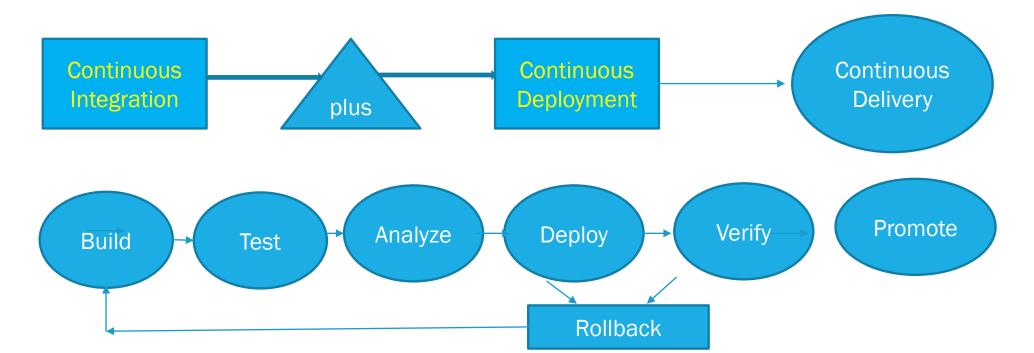
2021 2022 2023

Revenue Turnover > 2 Billion
Happy clients
Quick feature rollout
At par with competition
Strategy for Continuous delivery
Reduced costs
Customer base > 15k

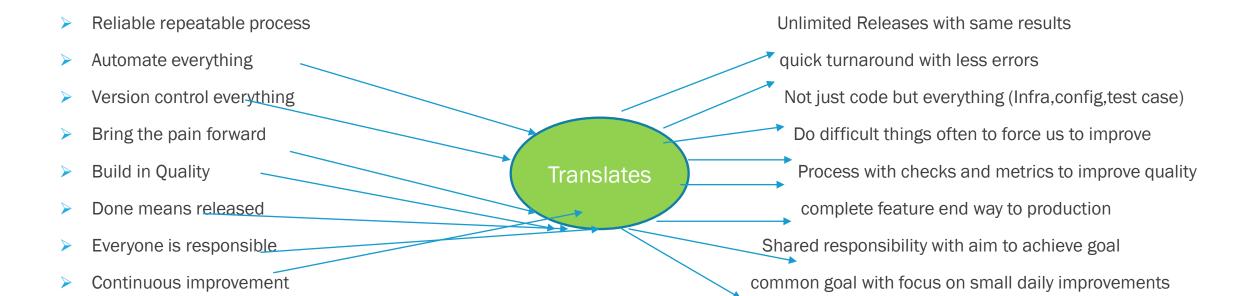
Markets - Asia, Europe

WHAT IS CI/CD

- Continuous Integration ensures code is integrated into a shared repository and building/testing each change automatically, as early as possible - usually several times a day
- Continuous Delivery ensures that all builds have gone through the testing cycles and the final deployable package is available to be deployed into production whenever required
- Continuous Deployment Publishing tested artifacts automatically to production with out any manual intervention in a standardized and consistent manner



CI/CD PRINCIPLES AND BENEFITS



VALUE FRAMEWORK

Increase revenue

Faster and More Frequent Production Deployments leads to New value-generating features released more quickly

Deploy to Production Without Manual Checks lead to Less time to market

Protect Revenue

Automated Smoke Tests leads to reduced downtime from a deploy-related crash or major bug

Automated Rollback Triggered by Job Failure leads to Quick undo to return production to working state

Reduce Cost

Catch errors early lead to less time spent on issues from new development code

Automate Infrastructure Clean up leads to Less infrastructure costs from unused resources

Avoid Cost

Catch unit test failures lead to less bugs in production

Detect Security Vulnerabilities lead to prevent embarrassing or costly security holes