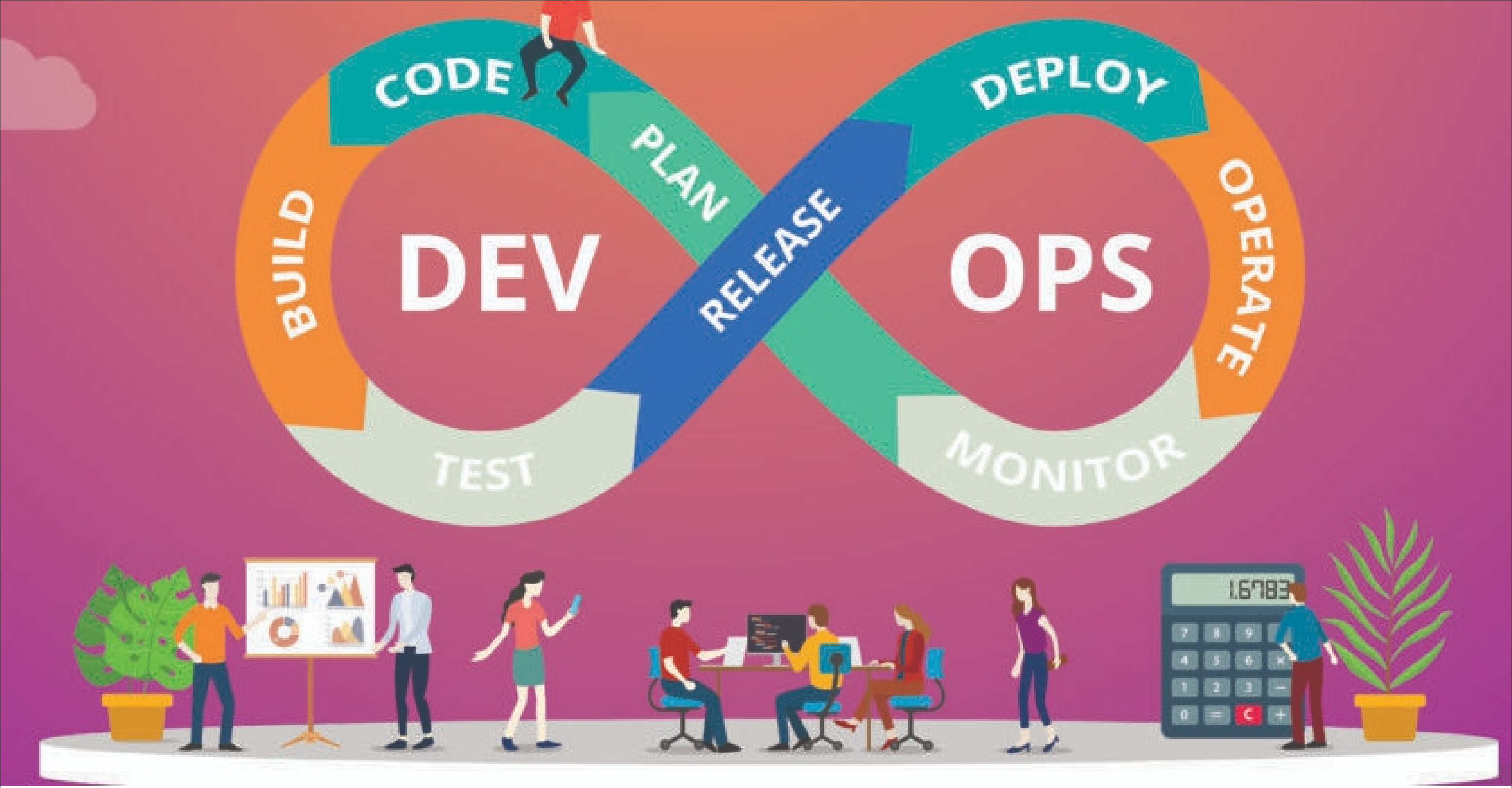
Udapeople

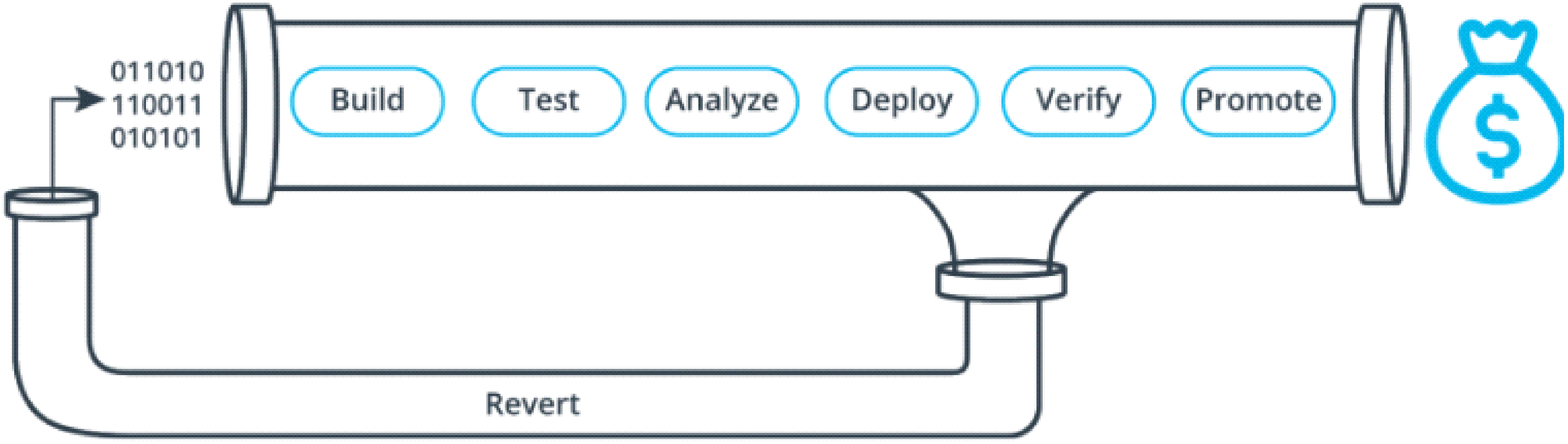
CONTINUOUS DELIVERY PROPOSAL



Why Continuous Delivery?

Continuous Delivery

- Cont. Delivery = Cont. Integration + Cont. Deployment
- CI/CD Stages & Pipeline



WHAT IS CI

Continuous integration is a coding philosophy and set of practices that drive development teams to implement small changes and check in code to version control repositories frequently. Because most modern applications require developing code in different platforms and tools, the team needs a mechanism to integrate and validate its changes.

WHAT IS CD

Continuous delivery picks up where continuous integration ends. CD automates the delivery of applications to selected infrastructure environments.

WHY DO YOU NEED CI/CD ?

- Investing more time in a release cycle than delivering value
- Going through integration hell every time you finish a feature
- Code gets lost because of botched merges
- Unit test suite hasn't been green in ages
- Deployments contribute to schedule slip
- Friction between ops and development departments
- Only one engineer can deploy a system
- Deployments are not cause for celebration

CD FACTS

LESS

- Bugs in production
- Time in testing
- Human error
- Time to market
- Downtime from deploy-related crash/bug
- Costs from unused resources

CI/CD BENEFITS

- Less developer time on issues from new developer code
- Less bugs in production and less time in testing
- Prevent embarrassing or costly security holes
- Less human error, Faster deployments
- Less infrastructure costs from unused resources which means extra cost avoidance
- New value-generating features released more quickly which means an increase in revenue
- Less time to market which means increased customer satisfaction
- Reduced downtime from a deploy-related crash or major bug
- Quick undo to return production to working state

Thank You