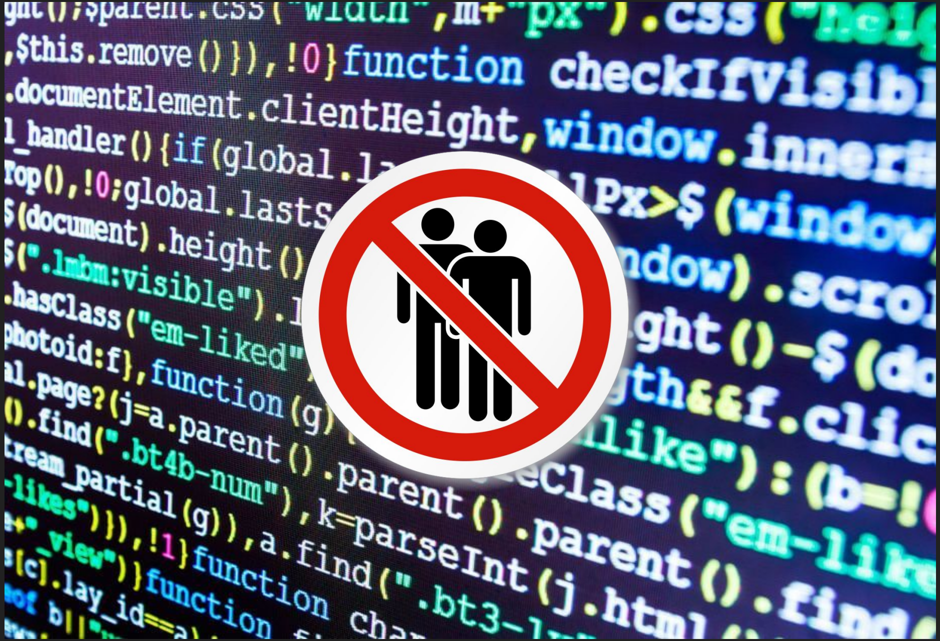
Business benefits of CI/CD

Current difficulties

- New version of software deploying too rarely
- Release preparation takes too long
- Code is almost unmaintainable
- More bugs bugs on every deployment
- Developers are exhausted and scared of not breaking build when adding new features
- Small amount of people actually know how to deploy a new version of a product

How can we improve it?

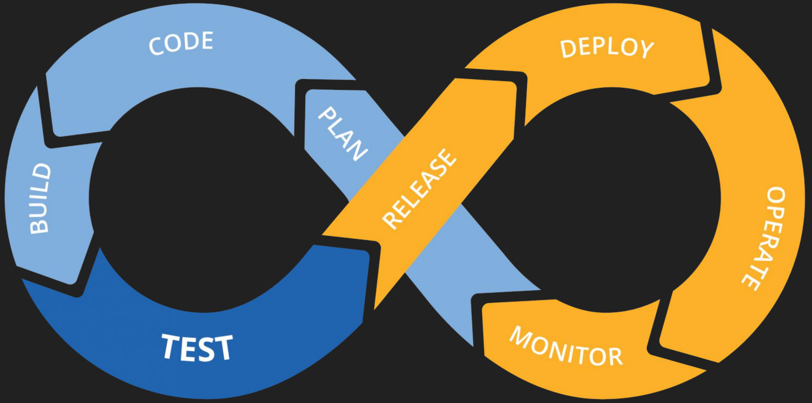
- CI/CD
- What is it?
- Possible difficulties
- Involve human as little as possible
- **AUTOMATE EVERYTHING**
- Biggest benefit → less money spent on avoidable problems
- Possible difficulty → CI/CD is not easy to build from a scratch



- Small incremental changes with short time between > Big feature deliveries in big time span
- Problems are easier to acknowledge
- Product quality is better
- Developers are more confident about introducing new features
- Unit tests → retaining code quality and lesser problems in production
- Automated releases → not in need of specific people constantly

CI/CD concept

- Continuous integration
    - practice of merging all developers' working copies to a shared mainline several times a day
    - special importance on test automation which results in generating great quality deployable artifacts
- Continuous delivery
    - automated way of releasing new features introduced in software product
- Continuous deployment
    - allows frequent deployments of new version of product - extends Continuous delivery
    - consists of instantiating the new version of a microservice and retiring the old version as it has drained all the requests in flight



In conclusion

- Broken builds → Rollbacks (users won’t even notice a problem)
- Less problems on deployment = Developers more confident
- Early bug detection
- More frequent releases = Happier customer
- Greatly reduced costs in long run
- Deployment time which took days are not even worth talking about
- Lesser downtimes = More money