OVERVIEW

- ☐ What does CI/CD stand for? The concepts explained
- ☐ What are our current pain points?
- ☐ CI/CD to the rescue. How we could benefit from DevOps principles
- ☐ What are the challenges we will be confronted with?

WHAT DOES CI/CD STAND FOR? THE CONCEPTS EXPLAINED

- CI/CD consist of three major concepts
  - Continuous Integration
    - Continuous Integration describes the process of merging developer branches to the main branch several times a day. CI puts an emphasis on test automation and finally generates a high quality, deploy artifact.
  - Continuous Delivery
    - In addition to Continuous Integration, Continuous Delivery makes sure that changes of a software product can be released quickly to customers in an automated way and at any point in time.
  - Continuous Deployment
    - Continuous Deployment extends Continuous Delivery in such a way that it allows frequent automated deployments without any human interaction. Typical phases in Continuous Deployment are Infrastructure Provisioning, Smoke Testing, Production Deployments and automated Rollbacks.

WHAT ARE OUR CURRENT PAIN POINTS?

1. Our manual release process is error-prone and always leads to delays of production deployments
2. This in turn often leads to poor software quality since we don’t have time for quality analysis anymore
3. Deployments are pretty complex. Only a chosen few experts are able to understand the whole process and tons of hand crafted helper scripts. No smoke tests and rollback mechanisms.
4. We get late feedback from the business department which prevents us from creating flexible solutions

CI/CD TO THE RESCUE. HOW WE COULD BENEFIT FROM DEVOPS PRINCIPLES (1/3)

- Problem Statement:
  - (1) Manual and error-prone release process and (2) poor software quality
- Solutions:
  - Implement Continuous Integration: automate compiling, testing, code analysis and artifact storage
  - Automate Infrastructure Creation
- Benefits:
  - Cost reduction due to less human errors and faster deployments
  - Reduce complexity and safe manual troubleshooting time

CI/CD TO THE RESCUE. HOW WE COULD BENEFIT FROM DEVOPS PRINCIPLES (2/3)

- Problem Statement:
  - (3) Complex deployments and handcrafted automation which often fail. Missing smoke tests and rollback mechanisms.
- Solutions:
  - Automate today’s manual deployment steps for smoke tests and rollbacks
  - Add automated infrastructure provisioning
- Benefits:
  - The truth lies in the source code and not in the heads of one or two experts. This means that regressions and breaking changes in code as well as in infrastructure deployments can be found much quicker and can be resolved for the whole automation process. And as a plus changes are always documented in source code.
  - Automated Smoke Tests and Rollbacks will protect project revenue due to reduced downtime's from deploy-related crashes and fast and automated rebuilding of production ready state

CI/CD TO THE RESCUE. HOW WE COULD BENEFIT FROM DEVOPS PRINCIPLES (3/3)

- Problem Statement:
  - (4) Late customer feedback
- Solution:
  - Implement Continuous Deployment: automated deployment of changes at any given point in time
  - Involve customers and business stakeholders already in deployment process
- Benefits:
  - Faster feedback cycles of customers lead to higher customer satisfaction rates since they are involved right from the beginning of feature development/deployment and not just at a fixed release date

WHAT ARE THE CHALLENGES WE WILL BE CONFRONTED WITH?

- Establishing CI/CD comes with a high amount of initial cost and learning. At first sight this might seem overwhelming compared to current best practices
- Delivering CI/CD pipelines is not a one time effort, but requires constant support and maintenance as well as continuous development and improvement
- Even though there are some challenges, CI/CD will improve overall business processes and dramatically reduce costs on the long run