Wasserfall-Workflow

A picture containing diagram

Description automatically generated

|  |  |
| --- | --- |
| Projektzeitskala |  |
| Planen |  |
| Erstellen |  |
| Prüfen |  |
| Freigeben |  |

TPM-Workflow

Diagram

Description automatically generated

|  |  |
| --- | --- |
| Prozessgruppen |  |
| Initiierung |  |
| Planung |  |
| Ausführung |  |
| Abschluss |  |
| Überwachung |  |

PRINCE2-Struktur

A picture containing diagram

Description automatically generated

|  |  |
| --- | --- |
| PRINCE2-Prozesse |  |
| Business Case |  |
| Organisation |  |
| Qualität |  |
| Plan |  |
| Risiken |  |
| Änderung |  |
| Status |  |
| PRINCE2-Themen |  |
| PRINCE2-Prinzipien |  |

Workflow im agilen Prozess

Diagram

Description automatically generated

|  |  |
| --- | --- |
| Sprint |  |
| Anforderungsanalyse |  |
| Design |  |
| Entwickelung |  |
| Test |  |
| Prüfung |  |
| Lieferung |  |

Kanban-Struktur

A picture containing shape

Description automatically generated

|  |  |
| --- | --- |
| Fertigung |  |
| Verteilung |  |
| Verbrauch |  |
| Kanban-Karte/-Etikett zurück zur Tafel |  |
| Kanban-Tafel |  |
| Rote Zone |  |
| Rote Zone löst Fertigung aus (z.B. Verbrauch wiederherstellen) |  |

Online Kanban-Board

Graphical user interface, application

Description automatically generated

|  |  |
| --- | --- |
| Online Kanban-Board |  |
| Offen |  |
| In Arbeit |  |
| Erledigt |  |
| Team A |  |
| Design T1 |  |
| Präsentation T2 |  |
| Validierung T3 |  |
| Datensammlung T1 |  |
| Design T3 |  |
| Validierung T2 |  |
| Datensammlung T3 |  |
| Konstruktion T3 |  |
| Team B |  |
| Validierung T4 |  |
| A/B-Test T4 |  |
| Modell 1 Test T4 |  |
| Modell 2 Test T4 |  |
| Modell 3 Test T4 |  |
| Datensammlung T4 |  |
| Analyse T4 |  |

Kanban-Prinzipien

Diagram, timeline

Description automatically generated

|  |  |
| --- | --- |
| Kanban-Prinzipien |  |
| 1. Keine Änderung am Anfang |  |
| 2. Verbesserung durch inkrementell und evolutionäre Änderungen |  |
| 3. Führungsverantwortung auf allen Ebenen |  |
| 4. Einhaltung der bestehenden Regeln und Verantwortlichkeiten |  |

Kanban-Praktiken

Diagram

Description automatically generated

|  |  |
| --- | --- |
| Kanban-Praktiken |  |
| Visualisierung |  |
| Minimierung laufender Arbeiten |  |
| Flussmanagement |  |
| Bewertung und Verbesserung |  |
| Verdeutlichung der Regeln |  |
| Feedbackschleifen |  |

Gedränge (engl. Scrum) beim Rugby



Agile Scrum Framework

Diagram, text

Description automatically generated

|  |  |
| --- | --- |
| Stakeholderkontakt |  |
| Produktinhaber |  |
| Backlogartikel |  |
| Produktbacklog |  |
| Entwicklungsteam |  |
| Arbeitsprognose durch das Team zur Erreichung des Sprintziels |  |
| Sprintplanung |  |
| Thema 1: Backlogartikel prognostizieren |  |
| Thema 2: Arbeit planen (z.B. Aufgaben) |  |
| Sprintbacklog |  |
| Produktbacklog anpassen |  |
| Daily Scrum |  |
| Sprint (max. 1 Monat) |  |
| Scrum Master |  |
| Iterative, inkrementelle Entwicklung und Lieferung |  |
| Potenziell freigebbares Inkrement |  |
| Sprintprüfung |  |
| Sprint-Retrospektive |  |

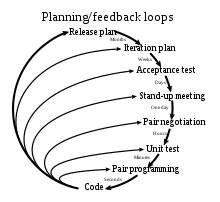
Agile Scrum-Ereignisse

A picture containing utensil, table, indoor, sitting

Description automatically generated

|  |  |
| --- | --- |
| Ereignisse |  |
| Sprintplanung |  |
| Daily Scrum |  |
| Sprintprüfung |  |
| Sprint-Retrospektive |  |
| Produktbacklog |  |
| Sprintbacklog |  |
| Inkrement |  |
| Sprint |  |

Extreme Programming



|  |  |
| --- | --- |
| Planung/Feedbackschleife |  |
| Releaseplan |  |
| Monate |  |
| Iterationsplan |  |
| Wochen |  |
| Akzeptanztest |  |
| Tage |  |
| Stand-Up-Meeting |  |
| Ein Tag |  |
| Verhandlung im Programmiererpaar |  |
| Stunden |  |
| Modultest |  |
| Minuten |  |
| Paarprogrammierung |  |
| Sekunden |  |
| Code |  |

Prinzipien des Lean Management

Diagram

Description automatically generated

|  |  |
| --- | --- |
| Kundenbedürfnisse |  |
| Mehrwert ermitteln |  |
| Wertstrom abbilden |  |
| Workflow entwerfen |  |
| Pull-System einführen |  |
| Prozess verbessern |  |

Vergleich der Methoden des agilen und des klassischen Projektmanagements

Table

Description automatically generated

|  |  |
| --- | --- |
| Anforderungen |  |
| Kunde |  |
| Dokumentation |  |
| Größenordnung des Projekts |  |
| Organisationsstruktur |  |
| Modellpräferenzen |  |
| Klassisches Projektmanagement |  |
| Klare Anforderungen und wenig Änderungen |  |
| Nehmen nicht am Prozess teil |  |
| Formale Dokumentation |  |
| Groß |  |
| Linear |  |
| Anpassung an Änderungen |  |
| Agiles Projektmanagement |  |
| Klare Anforderungen und viele Änderungen |  |
| Enge Zusammenarbeit |  |
| Implizites Wissen |  |
| Klein bzw. mittelgroß |  |
| Iterativ |  |
| Änderungen im Voraus bedenken |  |

DevOps-Pipeline

Diagram, timeline

Description automatically generated

|  |  |
| --- | --- |
| Planen |  |
| Entwickeln |  |
| Build erstellen |  |
| Testen |  |
| Freigeben |  |
| Bereitstellen |  |
| Ausführen |  |
| Überwachen |  |

DevOps mit kontinuierlicher Integration und Bereitstellung

Diagram, timeline

Description automatically generated

|  |  |
| --- | --- |
| Planen |  |
| Entwickeln |  |
| Build erstellen |  |
| Testen |  |
| Freigeben |  |
| Bereitstellen |  |
| Ausführen |  |
| Überwachen |  |
| Kontinuierliche Integration |  |
| Kontinuierliche Lieferung |  |
| Kontinuierliche Bereitstellung |  |

Monolithische Architektur im Vergleich zur Microservice-Architektur

Diagram

Description automatically generated

|  |  |
| --- | --- |
| Monolithische Architektur |  |
| Benutzeroberfläche |  |
| Geschäftsebene |  |
| Datenschnittstelle |  |
| DB |  |
| Microservice-Architektur |  |
| Microservice-Benutzeroberfläche |  |
| Microservice |  |

Kontinuierliche Integration und Bereitstellung anhand von Git, Jenkins, Docker und Kubernetes

Diagram

Description automatically generated

|  |  |
| --- | --- |
| Quelle committen |  |
| Git |  |
| Erstellen und testen |  |
| Jenkins |  |
| Bereitstellen |  |
| Image pushen |  |
| Docker |  |
| Image pullen |  |
| Kubernetes |  |

Docker-Komponenten

Diagram

Description automatically generated

|  |  |
| --- | --- |
| Client-Docker-CLI |  |
| REST-API |  |
| Server-Docker-Daemon |  |
| Docker |  |
| Managt |  |
| Netz |  |
| Container |  |
| Image |  |
| Datenvolumen |  |

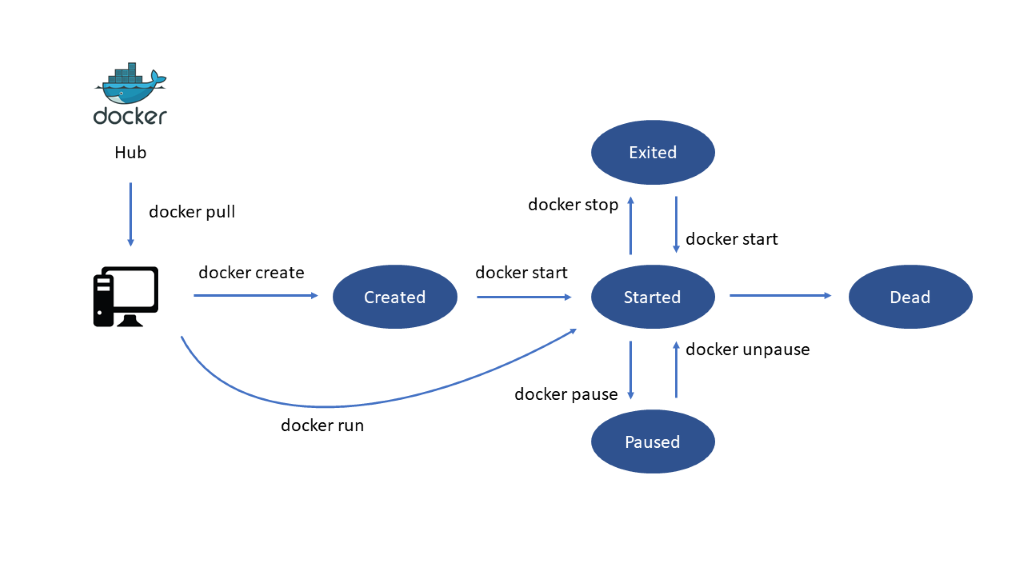
Docker-Image im Vergleich zu Docker-Container

Graphical user interface, diagram, application

Description automatically generated

|  |  |
| --- | --- |
| Docker-Image |  |
| Docker-Images |  |
| Image-Registrierung |  |
| Docker ausführen |  |
| Ist eine Instanz von |  |
| Docker-Container |  |
| Docker ps -a |  |
| Host-Betriebssystem |  |

Übersicht des Docker-Lebenszyklus



|  |  |
| --- | --- |
| Docker |  |
| Hub |  |
| Docker-Pull |  |
| Docker erstellen |  |
| Docker ausführen |  |
| Erstellt |  |
| Docker starten |  |
| Beendet |  |
| Gestartet |  |
| Angehalten |  |
| Docker stoppen |  |
| Docker erneut starten |  |
| Docker anhalten |  |
| Ende |  |

Kontinuierliche Integration und Bereitstellung anhand von Git, Jenkins, Ansible, Docker und Kubernetes

Diagram

Description automatically generated

|  |  |
| --- | --- |
| Quelle committen |  |
| Git |  |
| Erstellen und testen |  |
| Jenkins |  |
| Liefern |  |
| Ansible |  |
| Image pushen |  |
| Bereitstellen |  |
| Docker |  |
| Image pullen |  |
| Kubernetes |  |

Kubernetes-Cluster

Graphical user interface, application, website, Teams

Description automatically generated

|  |  |
| --- | --- |
| Kubelet |  |
| Docker |  |
| Knoten |  |
| Master |  |
| Kubernetes-Cluster |  |

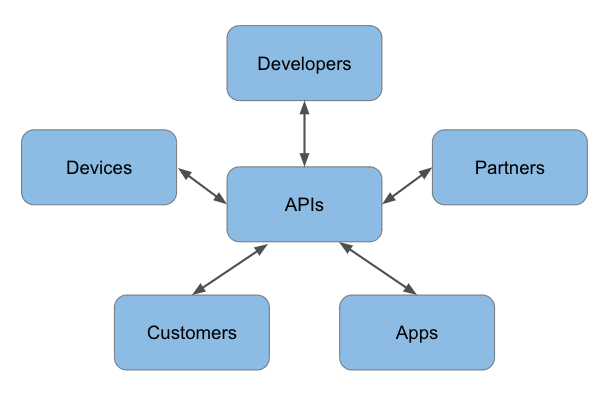
Kubernetes-Pods

Chart, bubble chart

Description automatically generated

|  |  |
| --- | --- |
| Kubelet |  |
| Docker |  |
| Pod |  |
| Knoten |  |

API-zentrische Ansicht



|  |  |
| --- | --- |
| APIs |  |
| Entwickler |  |
| Partner |  |
| Apps |  |
| Kunden |  |
| Geräte |  |

Testquadrant nach Marick

Technologieorientiert

Geschäftsorientiert

Zur Entwicklungsunterstützung

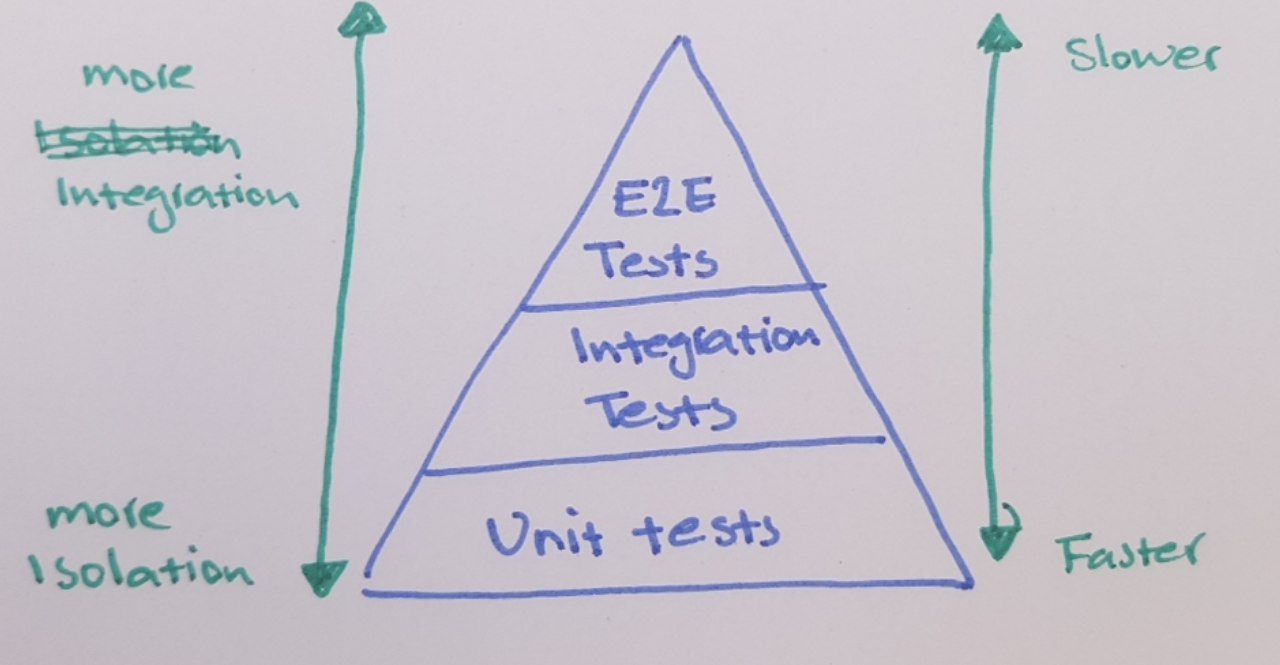
Produktprüfung

Diagram

Description automatically generated

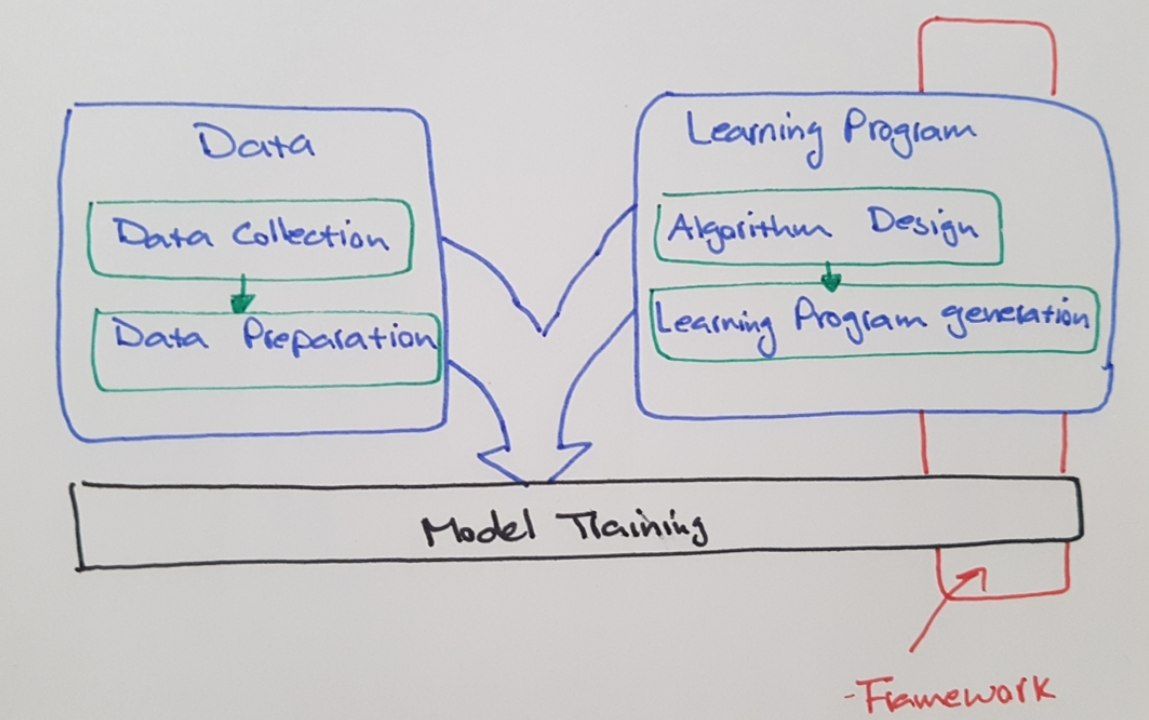
|  |  |
| --- | --- |
| Geschäftsorientiert |  |
| Technologieorientiert |  |
| Entwicklungsunterstützung |  |
| Produktprüfung |  |
| Q2 |  |
| Funktionale Akzeptanztests |  |
| (Automatisiert) |  |
| Q3 |  |
| Usability-Tests |  |
| Explorative Tests |  |
| Vorzeigeprojekt |  |
| (Manuell) |  |
| Q1 |  |
| Modultests |  |
| Integrationstests |  |
| Systemtests |  |
| Q4 |  |
| Nicht-funktionale Akzeptanztests |  |
| (Manuell/Automatisiert) |  |

Testautomationspyramide



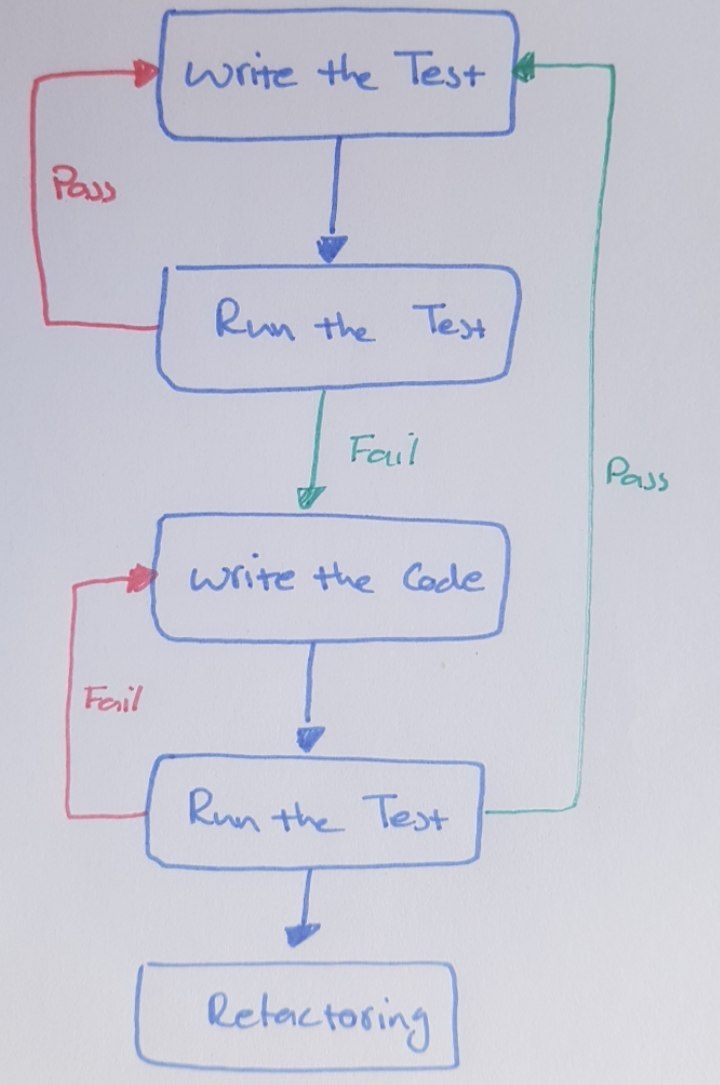
|  |  |
| --- | --- |
| Mehr Integration |  |
| Weniger Integration |  |
| E2E-Tests |  |
| Integrationstests |  |
| Modultests |  |
| Schneller |  |
| Langsamer |  |

Komponenten für ein ML-Modell



|  |  |
| --- | --- |
| Daten |  |
| Datensammlung |  |
| Datenaufbereitung |  |
| Modelltraining |  |
| Learning-Programm |  |
| Algorithmuskonzept |  |
| Learning-Programm-Generierung |  |
| Framework |  |

Lebenszyklus der testgetriebenen Entwicklung



|  |  |
| --- | --- |
| Test schreiben |  |
| Test ausführen |  |
| Code schreiben |  |
| Refactoring |  |
| Pass |  |
| Fail |  |

TDD-Zyklus

Diagram

Description automatically generated

|  |  |
| --- | --- |
| Test mit dem Ziel Fail schreiben |  |
| Test erfolgreich abschließen |  |
| Refactoring |  |

TDD, BDD und ATDD im Vergleich

|  |  |  |  |
| --- | --- | --- | --- |
|  | **TDD** | **BDD** | **ATDD** |
| **Definition** | Entwicklungsansatz zur Implementierung einer Funktion | Entwicklungsansatz auf der Grundlage des Systemverhaltens | Entwicklungsansatz für die Erfassung von Anforderungen (ähnlich wie beim BDD) durch Akzeptanztests, die vor der Implementierung der relevanten Funktionalität geschrieben werden |
| **Hauptfokus** | Modultests | Anforderungen aufgrund des Systemverhaltens verstehen | Schreiben von funktionalen Akzeptanztests |
| **Teilnehmer** | Entwickler | Entwickler, Kunden, Qualitätssicherungsingenieure | Entwickler, Kunden, Qualitätssicherungsingenieure |
| **Sprache** | Ähnlich wie der eigentliche Code | Natürliche Sprache | Natürliche Sprache |

|  |  |  |  |
| --- | --- | --- | --- |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

CD-Pipeline

|  |  |
| --- | --- |
| Code Commit |  |
| Build |  |
| Akzeptanztest |  |
| Performanztest |  |
| Produktion |  |

Orchestrierte Experimente

|  |  |
| --- | --- |
| Datenvalidierung |  |
| Datenaufbereitung |  |
| Modelltraining |  |
| Modellbewertung |  |
| Modellvalidierung |  |

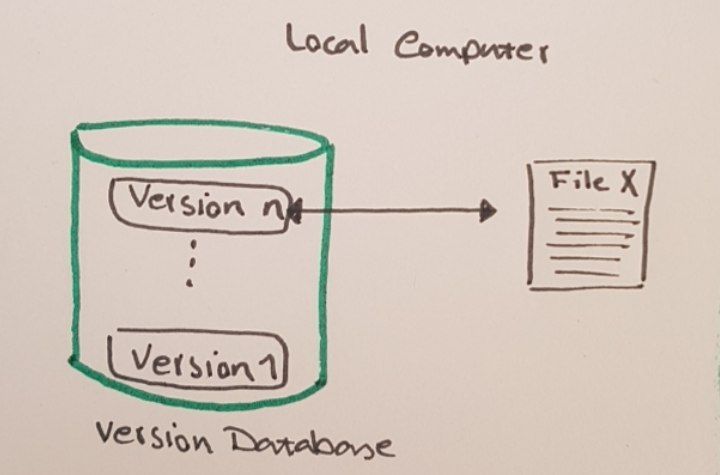
CI/CD für ein ML-Modell

Diagram

Description automatically generated

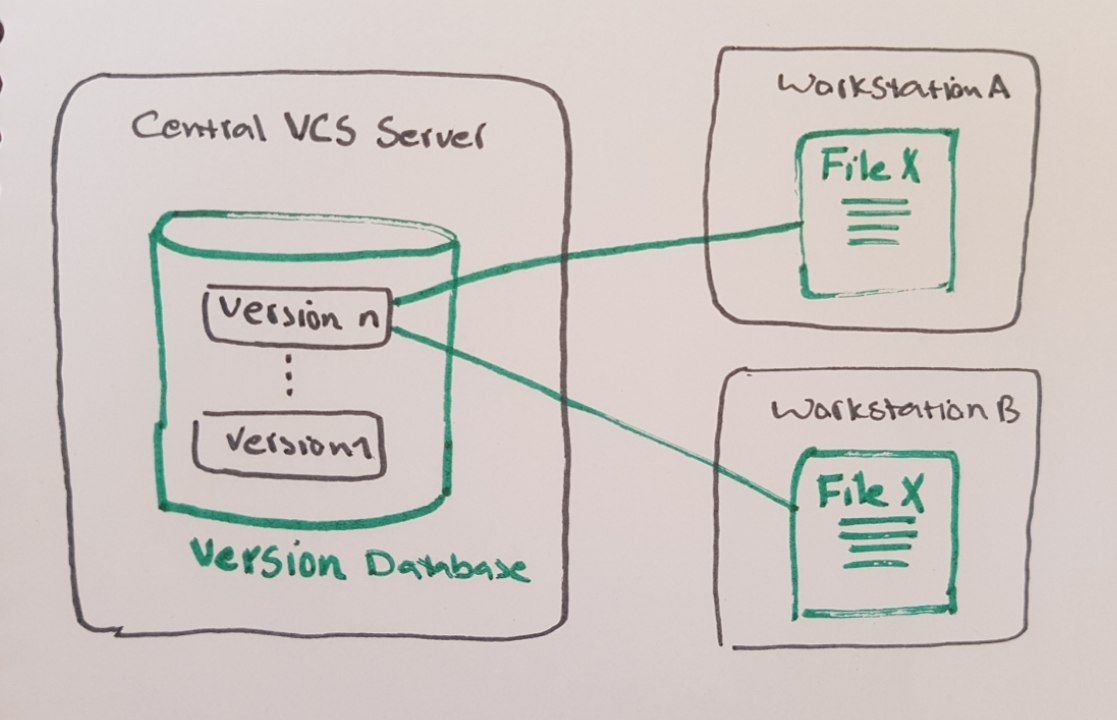
|  |  |
| --- | --- |
| Entwicklung und Experiment |  |
| Pipeline CI |  |
| Pipeline CD |  |
| Kontinuierliches Training |  |
| Kontinuierliche Lieferung |  |
| Überwachung |  |

Lokales Versionsverwaltungssystem



|  |  |
| --- | --- |
| Lokaler Computer |  |
| Version N |  |
| Version 1 |  |
| Versionsdatenbank |  |
| Datei X |  |

Zentrales Versionsverwaltungssystem



|  |  |
| --- | --- |
| Zentraler Server |  |
| Version N |  |
| Version 1 |  |
| Versionsdatenbank |  |
| Workstation A |  |
| Datei X |  |
| Workstation B |  |

Verteiltes Versionsverwaltungssystem

Diagram

Description automatically generated

|  |  |
| --- | --- |
| Server |  |
| Workstation A |  |
| Workstation B |  |
| Datei X |  |
| Version N |  |
| Version 1 |  |
| Versionsdatenbank |  |

Momentaufnahmen in Git als Versionsverwaltung

Diagram

Description automatically generated

|  |  |
| --- | --- |
| Version 1 |  |
| Version 2 |  |
| Version 3 |  |
| Version 4 |  |
| Datei A |  |
| Datei B |  |

Abschnitte in einem Git-Projekt

Timeline

Description automatically generated

|  |  |
| --- | --- |
| Arbeitsverzeichnis |  |
| Staging-Bereich |  |
| .git-Verzeichnis |  |
| (Repository) |  |
| Checkout des Projekts |  |
| Änderungen für den Commit vorbereiten |  |
| Commit |  |

Lokale Repository-Bäume für Git

A close up of text on a whiteboard

Description automatically generated

|  |  |
| --- | --- |
| Hinzufügen |  |
| Commit |  |
| Arbeitsverzeichnis |  |
| Indizieren |  |
| Für Commit vormerken |  |
| Kopfteil |  |

Verzweigung und GitHub

Diagram

Description automatically generated

|  |  |
| --- | --- |
| Masterzweig |  |
| Nebenzweig erstellen |  |
| Commit für Änderungen ausführen |  |
| Pull für Anforderung ausführen |  |
| Änderungen besprechen |  |
| Nebenzweig und Master zusammenführen |  |

Syntaxhervorhebung in integrierter Entwicklungsumgebung

Text

Description automatically generated

|  |  |
| --- | --- |
| import pandas as pd |  |
| import matplotlib.pyploy as plt |  |
| import datetime |  |
| from datetime import datetime |  |
| import plotly as pltly |  |
| import plotly.express as px |  |
| import plotly.graph\_objs as go |  |
| import plotly.io as pio |  |
| from PIL import Image |  |

IDE-Komponenten in Eclipse

Graphical user interface, application, Word

Description automatically generated

Landing Page von Jupyter Notebook

Graphical user interface, application

Description automatically generated

Ein Jupyter Notebook (.jpynb Datei)

Graphical user interface, text, application, email

Description automatically generated

Verwendung von Jupyter Notebook zur Ausführung eines einfachen Python-Codes

Graphical user interface, text, application, email

Description automatically generated

Verwendung von Markdown zur Dokumentation von Klartextcode in einem Jupyter Notebook

Graphical user interface, text

Description automatically generated

Grafische Darstellung von Daten in Jupyter Notebook

Chart, line chart

Description automatically generated

API-zentrische Ansicht

Diagram

Description automatically generated

|  |  |
| --- | --- |
| Entwickler |  |
| APIs |  |
| Partner |  |
| Apps |  |
| Kunden |  |
| Geräte |  |

Die ersten Zeilen eines Datensatzes mit .head()

Table

Description automatically generated

|  |  |
| --- | --- |
| Out [15]: |  |
| Kelchblattlänge in cm |  |
| Kelchblattbreite in cm |  |
| Blütenblattlänge in cm |  |
| Blütenblattbreite in cm |  |
| Art |  |
| Iris setosa |  |

Übersicht über einen Datensatz mit .describe()

Table

Description automatically generated

|  |  |
| --- | --- |
| In [16]: 1 Iris\_df.describe( ) |  |
| Out [16]: |  |
| Kelchblattlänge in cm |  |
| Kelchblattbreite in cm |  |
| Blütenblattlänge in cm |  |
| Blütenblattbreite in cm |  |
| Anzahl |  |
| Mittel |  |
| Standard |  |
| Minimum |  |
| Maximum |  |

Beispiel eines Streudiagramms zur Visualisierung von Beziehungen zwischen Funktionen

Chart, scatter chart

Description automatically generated

|  |  |
| --- | --- |
| Beziehung zwischen der Länge und Breite des Kelchblatts |  |
| Kelchblattbreite |  |
| Kelchblattlänge |  |
| Iris setosa |  |
| Iris versicolor |  |

Modellentwicklung und Produktionslebenszyklus (I)

Diagram

Description automatically generated

|  |  |
| --- | --- |
| Problemverständnis |  |
| Datensammlung |  |
| Datenbereinigung und -aufbereitung |  |
| Explorative Datenanalyse |  |
| Experimentieren, Funktions- und Modellauswahl |  |
| Hyperparameteroptimierung |  |
| Training, Endauswahl |  |
| Erstbereitstellung und Integration |  |
| Bereitstellung in der Produktion |  |

Modellentwicklung und Produktionslebenszyklus (II)

Diagram

Description automatically generated

|  |  |
| --- | --- |
| Experimentierphase |  |
| Problemermittlung, Datensammlung und -analyse |  |
| Auswahl des ML-Algorithmus und Codierung des Modells |  |
| Experimente mit Daten und Modelltraining |  |
| Einstellen der Modellhyperparameter |  |
| Einstellungs- und Trainingsiterationen |  |
| Produktionsphase |  |
| Datentransformation |  |
| Modelltraining |  |
| Modellbedienung für Vorhersage im Online-/Batchbetrieb |  |
| Monitoring der Modellleistung |  |

Modellentwicklung und Produktion mit Kubeflow

Graphical user interface

Description automatically generated

|  |  |
| --- | --- |
| Experimentierphase mit Kubeflow |  |
| Problemermittlung, Datensammlung und -analyse |  |
| Auswahl des ML-Algorithmus und Codierung des Modells |  |
| Experimente mit Daten und Modelltraining |  |
| Einstellen der Modellhyperparameter |  |
| Einstellungs- und Trainingsiterationen |  |
| PyTorch |  |
| scikit-learn |  |
| TensorFlow |  |
| XGBoost |  |
| Jupyter Notebook |  |
| Fairing |  |
| Pipelines |  |
| Katib |  |
| Produktionsphase mit Kubeflow |  |
| Datentransformation |  |
| Modelltraining |  |
| Modellbedienung für Vorhersage im Online-/Batchbetrieb |  |
| Monitoring der Modellleistung |  |
| Chainer |  |
| MPI MXNet |  |
| TFJob |  |
| KFServing |  |
| NVIDIA TensorRT |  |
| PyTorch |  |
| TFServing |  |
| Seldon |  |
| Metadaten |  |
| TensorBoard |  |
| Pipelines |  |

Umgang mit diesen Werkzeugen

Text

Description automatically generated

|  |  |
| --- | --- |
| Vorsicht! |  |
| Diese Maschine hat kein Gehirn. Bitte das eigene verwenden. |  |

SageMaker Konsole

Graphical user interface, text, application

Description automatically generated

Zugriff auf IAM-Rollen

Graphical user interface, text, application, email

Description automatically generated