



Bundesamt für
Kartographie und Geodäsie

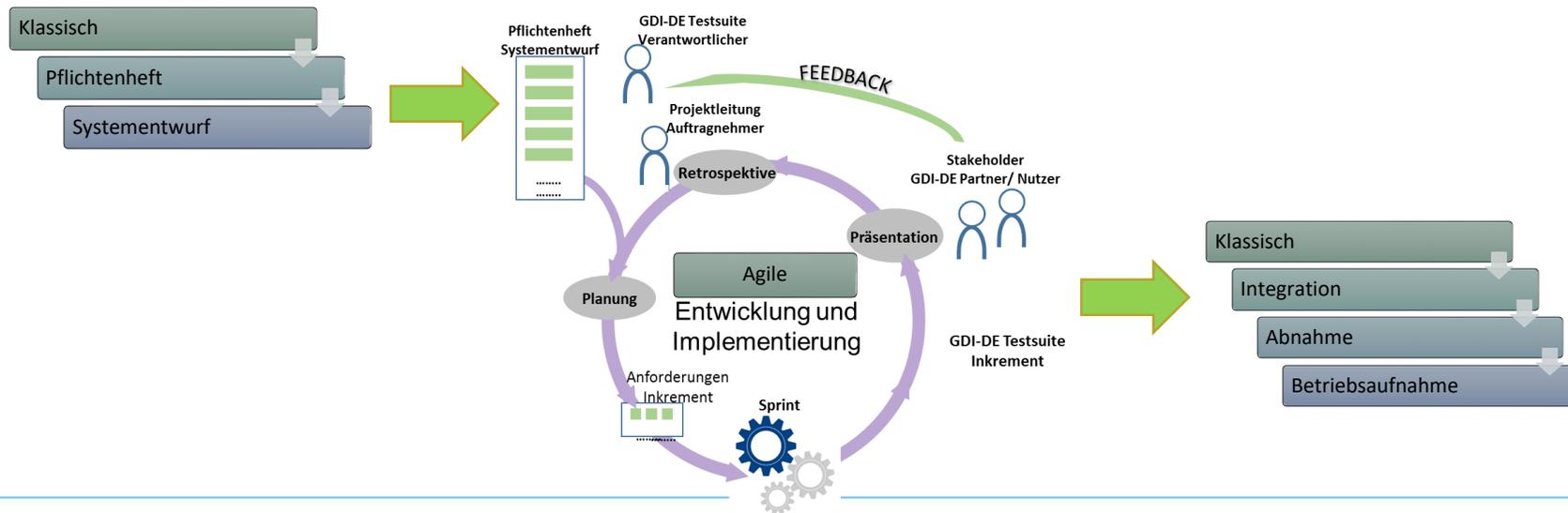


Neuentwicklung der GDI-DE Testsuite

Marc Jansen und Manuel Fischer

Projekteigenschaften / Rahmenbedingungen

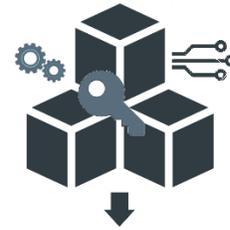
- Die Neuentwicklung wird im Auftrag der GDI-DE durchgeführt
- Relaunch-Projekt zu vorhandener GDI-DE Testsuite Software
- Projektlaufzeit April 2019 bis August 2020
- Projektpartner **terrestris | interactive instruments | sopra steria**
- V-Modell XT als inhaltlicher Rahmen | Hybrides Projektvorgehensmodell



National Technische Komponenten der GDI-DE



Geodatenkatalog.de



GDI-DE Registry



Geoportal.de



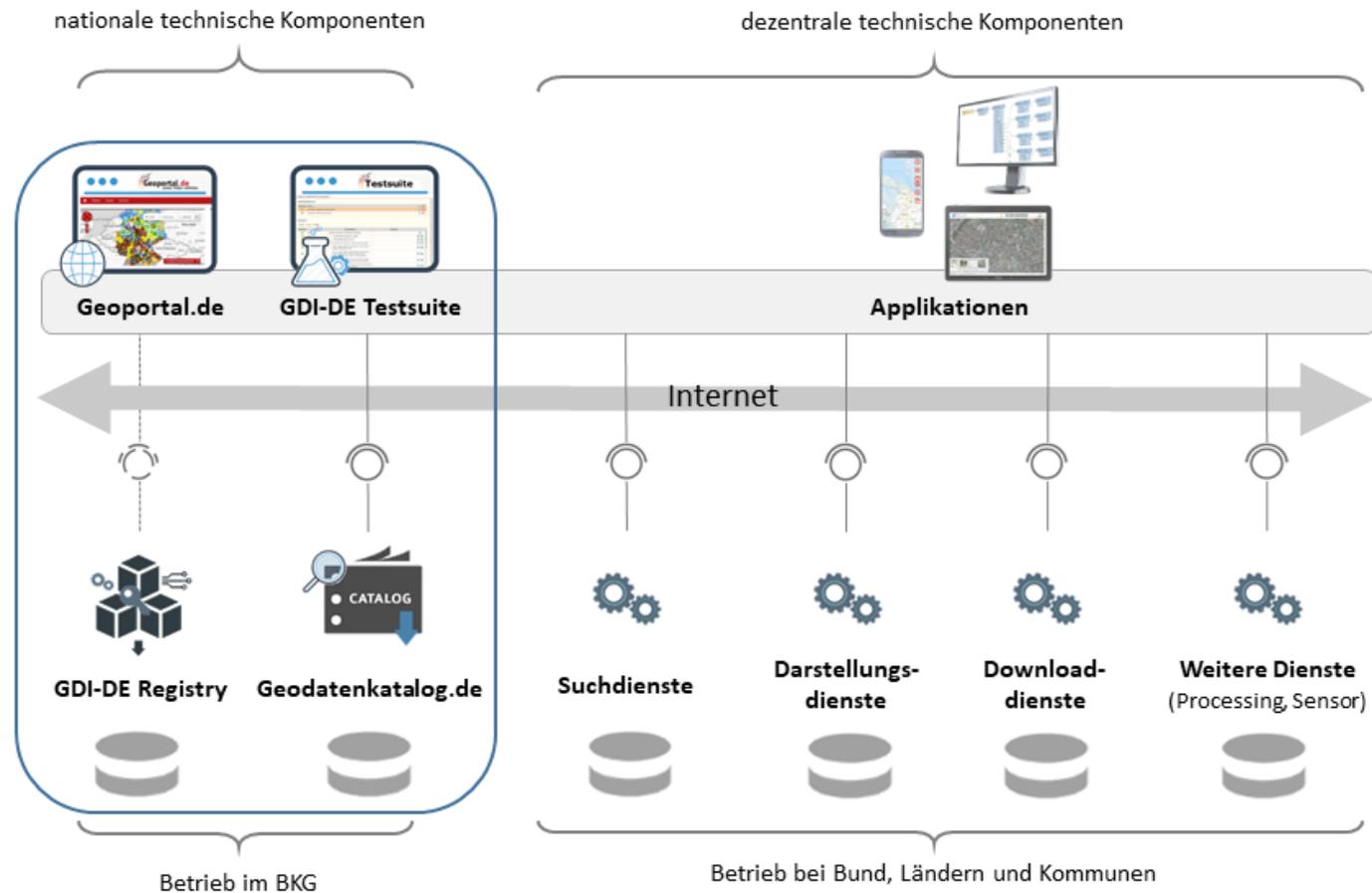
GDI-DE Testsuite

- Zentrale Testplattform der Qualitätssicherung innerhalb der GDI-DE
- Konformität Metadaten und Geodatendienste zu nationalen und internationalen Standards (z.B. INSPIRE)
- **Open Source Software**
- Seit 2011 unter <https://testsuite.gdi-de.org/gdi/> frei verfügbar
- Webanwendung | Download-Möglichkeit für lokale Nutzung | SOAP-Schnittstelle
- Tests für die Konformitätsprüfung von
 - Metadaten
 - Katalog-/Suchdiensten (CSW)
 - Karten-/Darstellungsdiensten (WMS)
 - Downloaddiensten (WFS, Atom)
- Qualitätstest für Geodatendienste (Leistungsfähigkeit und Verfügbarkeit)



GDI-DE Testsuite - Funktionsweise

- GDI-DE basiert auf dienstorientierter Architektur -> standardisierte Schnittstellen
- Konformität der Dienste/Metadaten ist wichtig!



GDI-DE Testsuite - Funktionsweise

Metadatenatz



GDI-DE Testsuite



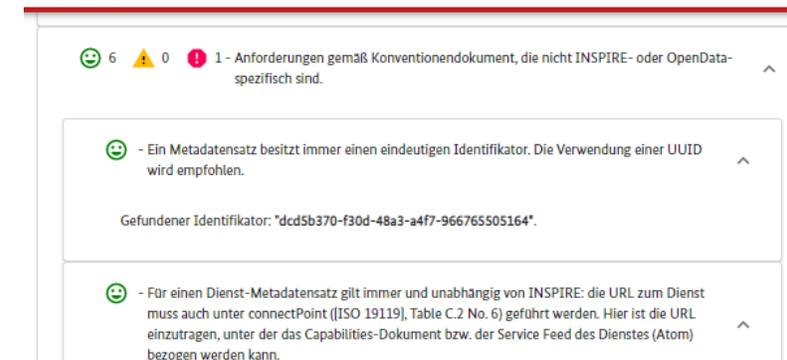
Testklasse



Report



```
<?xml version="1.0" encoding="UTF-8"?>
<!-- Dieses Metadatendokument ist ein fiktiver Beispiel Datensatz zur Veranschaulichung -->
<gmd:MD_Metadata xmlns:gmd="http://www.isotc211.org/2005/gmd"
  xmlns:gco="http://www.isotc211.org/2005/gco"
  xmlns:xlink="http://www.w3.org/1999/xlink"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://www.isotc211.org/2005/gmd http://schemas.opengis.net/csw,
  <gmd:fileIdentifier>
    <gco:CharacterString>e44365f4-b0a3-4d13-a90d-a0ed9aaa8012</gco:CharacterString>
  </gmd:fileIdentifier>
  <gmd:language>
```



ATS -Eindeutiger Metadatenatzidentifikator

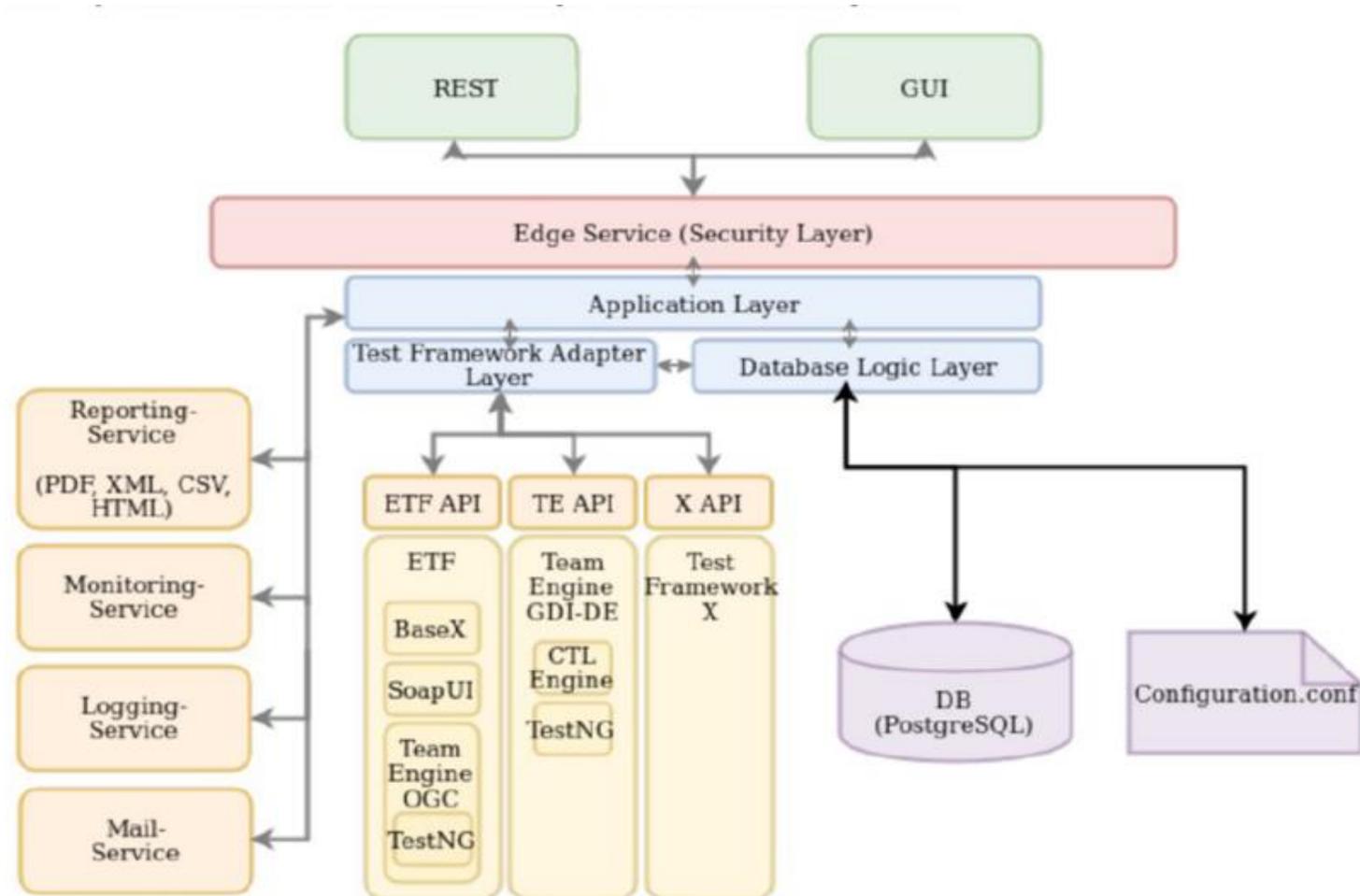
Beschreibung	Ein Metadatenatz besitzt immer einen eindeutigen Identifikator. Die Verwendung einer UUID wird empfohlen.
Testmethode	<ol style="list-style-type: none"> 1. Prüfe, ob das Element <code>fileIdentifier</code> als Unterelement von <code>MD_Metadata</code> vorhanden ist. 2. Prüfe, ob das Element <code>fileIdentifier</code> nicht leer oder NULL ist. 3. Wenn [2] zutrifft: Prüfe, ob der Eintrag eine UUID gemäß RFC4122 ist: <ul style="list-style-type: none"> • 16-Byte-Zahl • hexadezimal notiert • in fünf Gruppen unterteilt
Fehlermeldungen/Warnungen	<ul style="list-style-type: none"> • [F] Wenn Element <code>fileIdentifier</code> nicht vorhanden ist [1]: "Fehler: Es muss ein <code>fileIdentifier</code>-Element vorhanden sein." • [F] Wenn kein Eintrag vorhanden ist [2]: "Fehler: Ein Metadatenatz muss einen eindeutigen Identifikator besitzen. Das Feld <code>fileIdentifier</code> darf daher nicht leer sein. Die Verwendung einer UUID gemäß RFC4122 wird empfohlen (z.B. <code>550e8400-e29b-11d4-a716-446655440000</code>)." • [W] Wenn Eintrag keine UUID ist [3]: "Warnung: Die Verwendung einer UUID gemäß RFC4122 wird empfohlen (z.B. <code>550e8400-e29b-11d4-a716-446655440000</code>). Sollte der bisher verwendete Metadatenatzidentifikator historisch bedingt nicht den Regeln einer UUID entsprechen, so ist dieser im Sinne des Bestandsschutzes weiterhin zulässig."
XPath	<code>MD_Metadata/fileIdentifier</code>

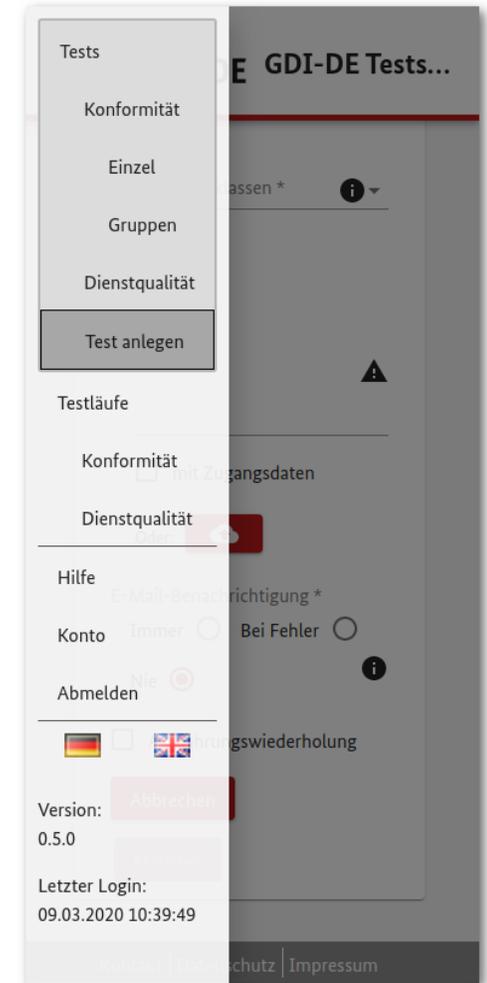
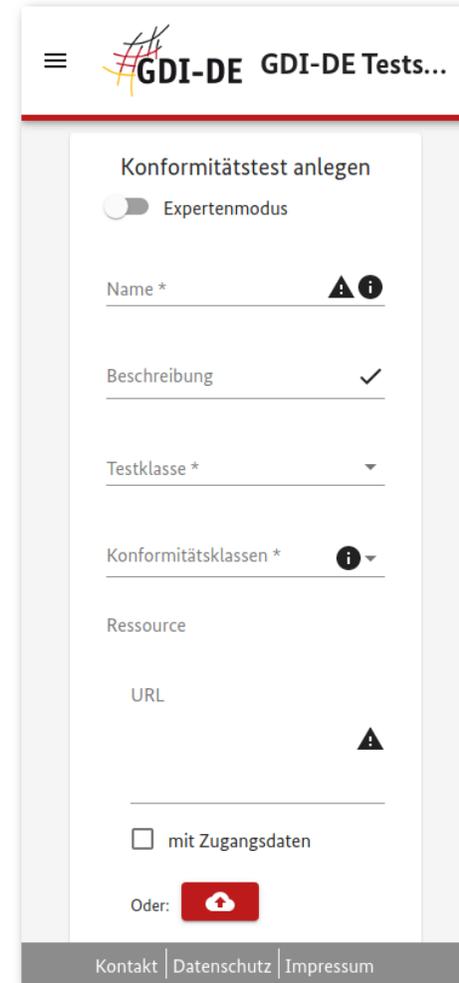
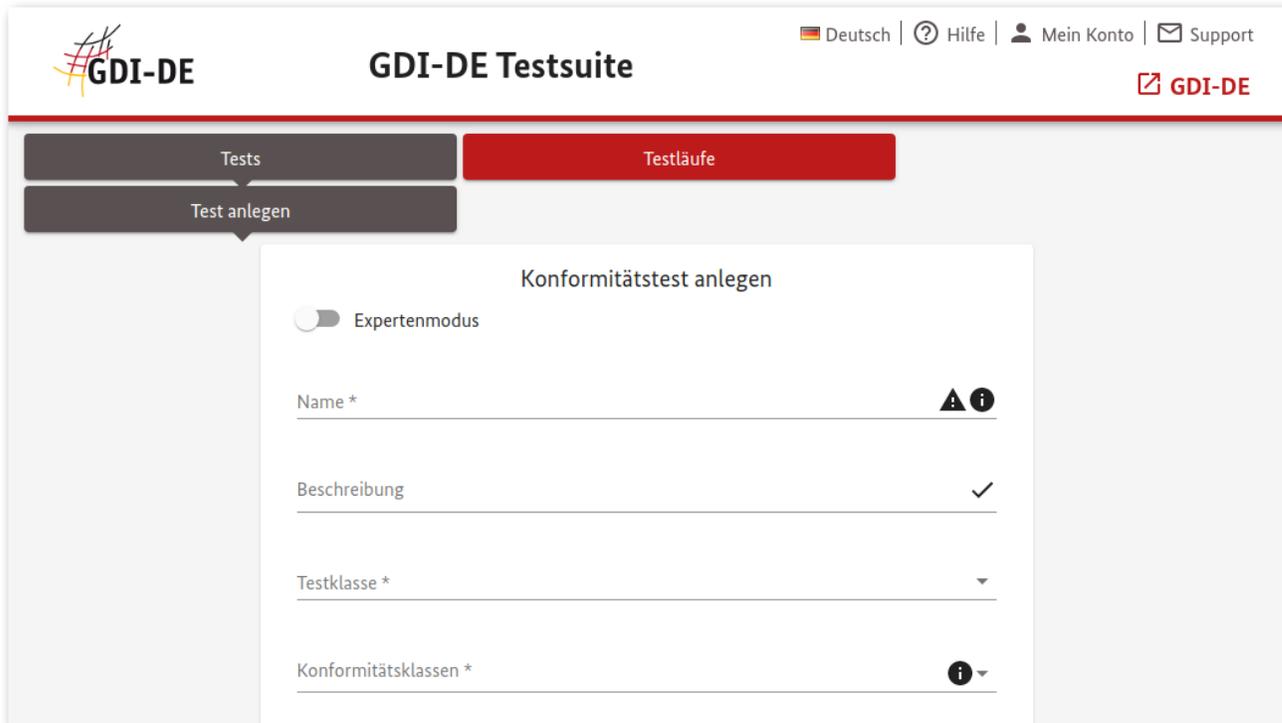
GDI-DE Testsuite -Projektziele

- **Verbesserte Benutzerfreundlichkeit**
 - Prototyping mit Einbindung der Nutzer in den Designprozess
 - Responsive Design
 - Intelligente Assistenten, kontextsensitive Tooltips
 - Aussagekräftige Testreports
- **Funktionale Erweiterbarkeit und Horizontale Skalierbarkeit**
 - Modulare Microservice basierte Architektur
- **Wartbarkeit**
 - Durchgängige Testautomatisierung, Code Reviews
 - Continuous Integration, Delivery und Deployment
- **Hochverfügbarkeit**
 - HA-Cluster Installation



Systemarchitektur





- **Gemeinsam getroffene Entscheidung pro REST und contra SOAP**

- **z.b. <https://example.com/bkg-runner/testconfigurations>**

```
▼ {content: [,...],...}
  ▼ content: [,...]
    ▼ 0: {name: "MJ Test 1", description: "", testClassId: "INSPIRE View Services - WMS 1.3.0", file: null,...}
      name: "MJ Test 1"
      description: ""
      testClassId: "INSPIRE View Services - WMS 1.3.0"
      file: null
      testClass: "INSPIRE View Services - WMS 1.3.0"
      notification: "NEVER"
      repetitionInterval: 0
      startDate: "2020-03-09T09:39:56.958Z"
      endDate: null
      username: "marcjansen"
      id: 60
```

- **& entsprechend z.b. <https://example.com/bkg-runner/testconfiguration/60>**

```
▼ {name: "MJ Test 1", description: "", testClassId: "INSPIRE View Services - WMS
  name: "MJ Test 1"
  description: ""
  testClassId: "INSPIRE View Services - WMS 1.3.0"
  file: null
  testClass: "INSPIRE View Services - WMS 1.3.0"
  notification: "NEVER"
```

- **Unterstützte Testframeworks (ETF, Team-Engine, QoS-Tests)**
- **Besonderheiten des Test Framework Adapter Layers**
 - Kommunikation mit den Testframeworks ETF und der TEAM-Engine (via API)
 - einheitliche API für die Testausführung
 - einfache Erweiterbarkeit zur Unterstützung von weiteren Testframeworks
- **Testframeworks:**
 - TEAM Engine via Java-API, <https://cite.opengeospatial.org/teamengine/>
 - ETF via REST, <http://etf-validator.net/>
 - Eigene? Implementiere Interface, erzeuge korrektes Testpaket, Enjoy 😊



Eingesetzte Tools, Sprachen und Bibliotheken (Auswahl)

- Java 11
- Spring Boot, Hibernate



- TypeScript
- React
- Material UI



- PostgreSQL
- REDIS



- ELK (Elastic Search, logstash, kibana)
- Docker, Helm, Kubernetes k8s

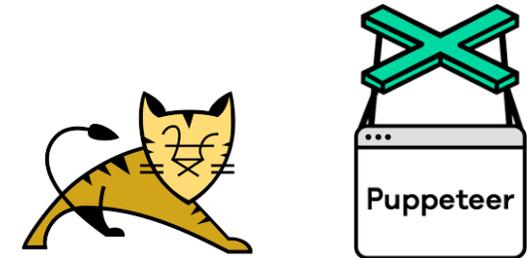
- git / GitLab



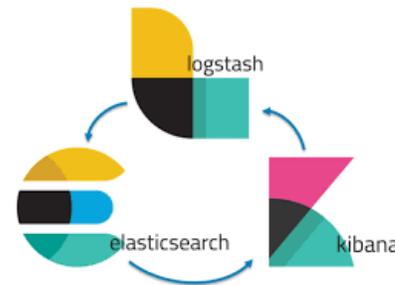
- Maven



- Puppeteer

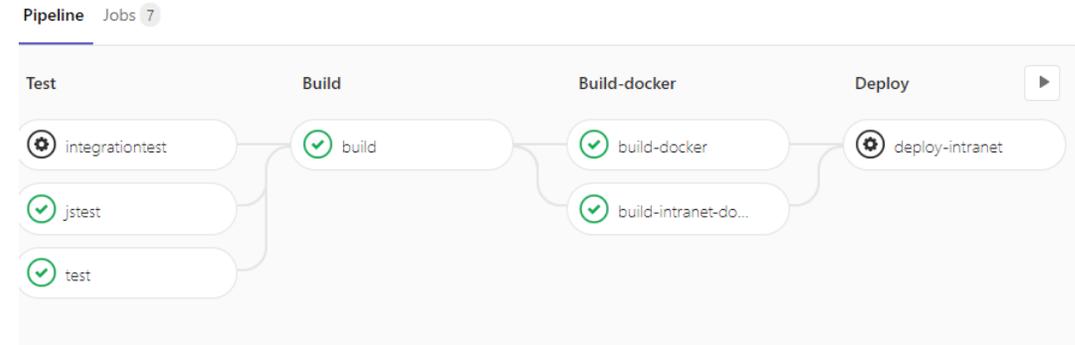


- Tomcat

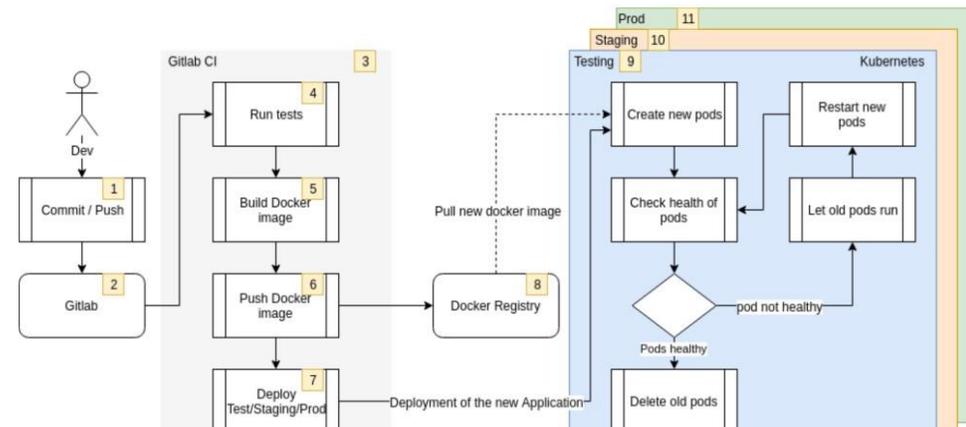


Entwicklungs -und Deploymentprozess

- **Versionsverwaltung mit GitLab**
 - GitLab-CI zur Unterstützung automatisierter Test- und Deploymentsschritte

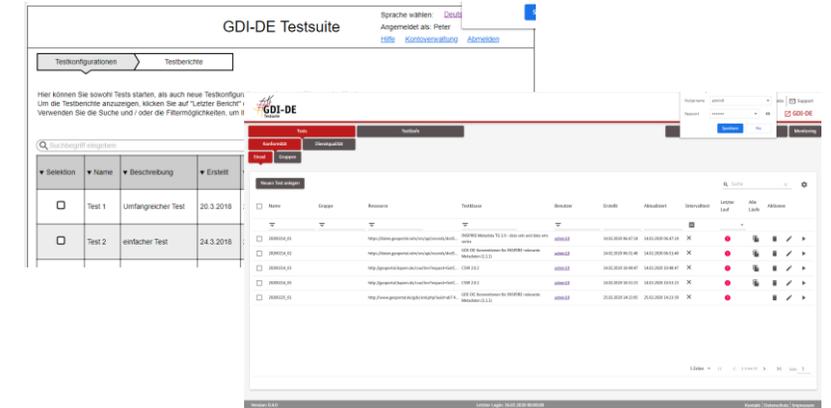


- **Bereitstellungsprozess für Test-, Staging und Produktivumgebung**



Features

- Anwendung ist Responsive und intuitiv bedienbar
- Multifaktor-Authentifizierung für Administratoren
- Lauffähig in einer Cluster-Umgebung (Kubernetes)
- Einfach zu konfigurieren und zu erweitern



Live-Demo

GDI-DE Testsuite

- Gelungenes Projekt auf der Basis von Open Source Entwicklungen
- Sehr gute Zusammenarbeit mit den Projektpartnern
- Erster Anwenderakzeptanztest
- **Geplanter Go-Live ist im Mai 2020 (<https://testsuite.gdi-de.org/gdi/>)**
- **Die Software ist dann als Open Source verfügbar**



Bundesamt für
Kartographie und Geodäsie



Vielen Dank für Ihre Aufmerksamkeit.

Bundesamt für Kartographie und Geodäsie
GDI-Anwendungen
Richard-Strauss-Allee 11
60598 Frankfurt am Main

Manuel Fischer
manuel.fischer@bkg.bund.de
www.bkg.bund.de
Tel. +49 69 6333 – 340