



---

UT Offering  
well architected lakehouse  
& MLOps at scale

---



# Overview

## 1. Agile Requirements Engineering

- UML / BPMN to model and engineer requirements

## 2. Best-in-class Software Architecture

- Derivate of TOGAF (Business, Technology, Deployment, Security)

## 3. Agile planning, development, and testing

- Scrum, SAFe, LeSS, Kanban

## 4. Quantitative, metrics-based QA system (ISO / IEC 25010:2019)

- Product quality constantly monitored; findings incorporated into sprints

## 5. Secure and professional deployment and operations

- Automated, repeatable and auditable



## Standards & Practices

Ultra Tendency complies with the following standards and practices:

- **ISO 27001** Information Security (all offices are certified)
- **TISAX** Information Security (all offices are certified)
- **ISO 22301** Business continuity (Organization is certified)
- **ISO 9001** Quality Management (H1 2023 certified)
- **ISO 31000** Risk Management (is applied)
- **TOGAF 9.2** For all architectural modeling
- **BPMN / UML** Requirements engineering
- **DMBoK 2 / ISACA DMM** Data maturity assessments
- **Project management** PMI, HERMES, PM2, CRO, CCRO



# Agile Requirements Engineering

- Interviews with key stakeholders to define their requirements without room for interpretation
- Requirement visualization via UML/BPMN leads to Business Architecture
- Experienced POs define user stories
- User Stories managed in backlog
- “Definition of Ready” for user stories
- Scrum planning to obtain “buy-in” of developers on user stories
- “Definition of Done” for delivered user stories



# The UT Approach to Platform Architecture

## **Business Architecture**

Targets business stakeholders and communicates use cases via UML use case diagrams or BPMN process diagrams

## **Technology Architecture**

Targets system architects and software developers by providing detailed component diagrams that explain how a system is composed out of software components

## **Deployment Architecture**

Targets developers and system administrators, and describes how logical software components are deployed to physical or virtual infrastructure nodes

## **Security Architecture**

Targets CISOs and Security specialists and describes in detail how the system complies with general and corporation-specific security requirements



# UT approach to MLOps at Scale

How to build Data Products that last



## Benefits of MLOps

Introduced characteristics	Expectation	Benefit
Automatization	Automating the workflow of the ML process saves time and avoids human-induced errors this increases the <b>maintainability</b> and <b>reliability</b> of roll-outs.	Increased efficiency, agility, time savings
Standardization	Better <b>collaboration</b> between teams, <b>reduces compatibility issues</b> . Model, data, code versioning increases <b>reproducibility</b> . As such <b>scaling</b> of ML workflows becomes easier and <b>increases the time to market</b> for new developments.	Increased scalability and security, lower complexity, better collaboration, increased security
Observability	Insights on model metrics as well as data allows measuring important performance and quality KPI's (e.g. model or data drift). This enables experts to act upon events where certain thresholds are breached and limit the impact by steering against it. This directly results in <b>fewer product issues</b> and <b>faster mean time to recover</b> due to increased <b>observability</b> .	Increased monitorability, service performance, reliability



## MLOps Offering Brief

UT is able to offer a complete MLOps-tailored project lifecycle, from initial vision workshop to the final project handover. UT maintains 3 project phases, while also offering complementary trainings by UT Academy (UTA):

- During an initial **Vision Workshop**, we discover the project goals, needs and evaluate its requirements.
  - Afterwards, **Phase 0** translates the needs and scope of the MLOps project into a technical and deployment architecture according TOGAF methodology, for which a viable initial MVP is to be created.
  - In the actual **Implementation Phase**, through Agile Development, we incrementally expand and build upon the bare-bones MVP, using Agile development during the implementation phase.
- > Through UT Academy (UTA), training can be offered where to generate necessary in-house skills for independent operations by the customer.

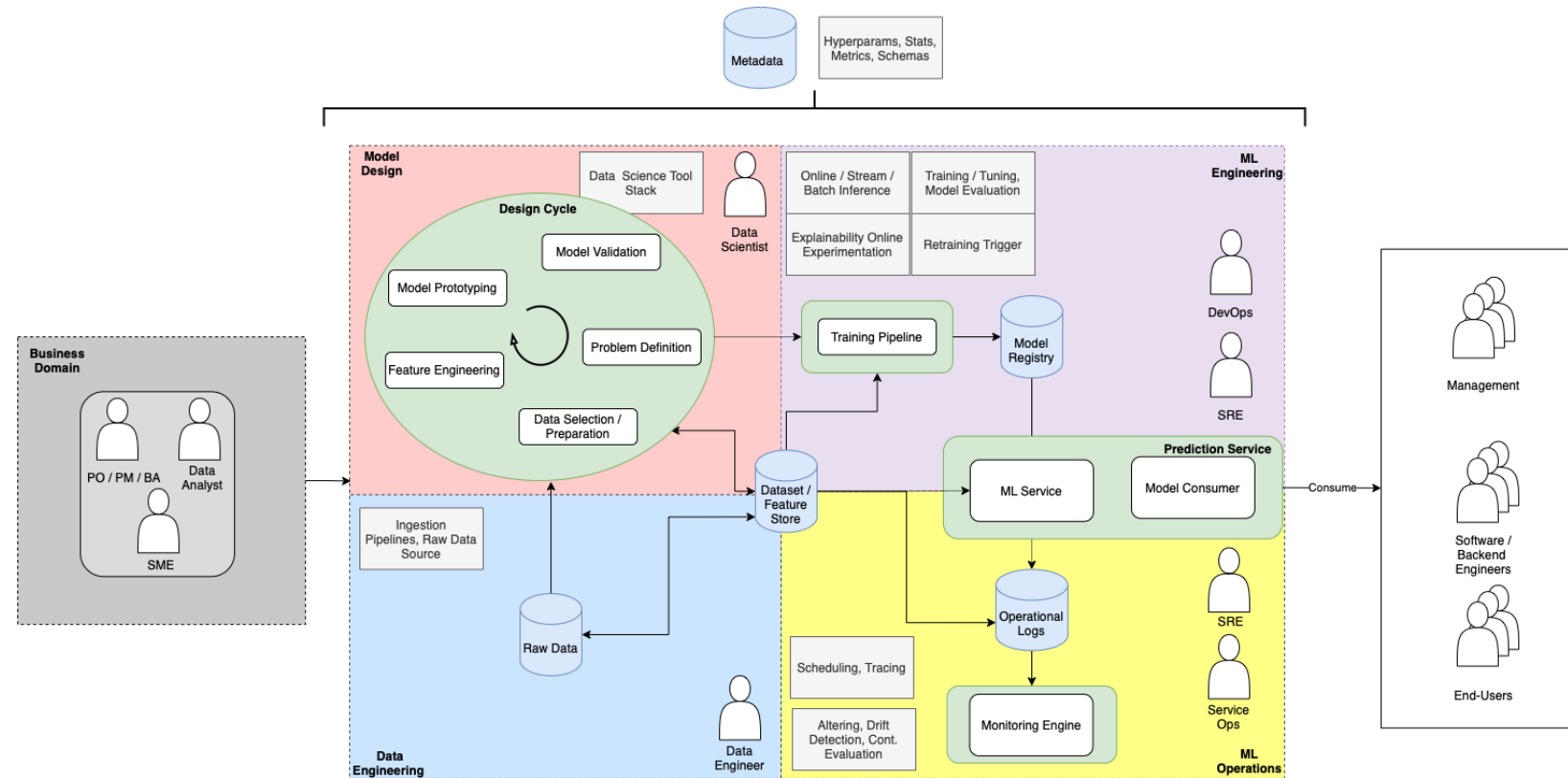




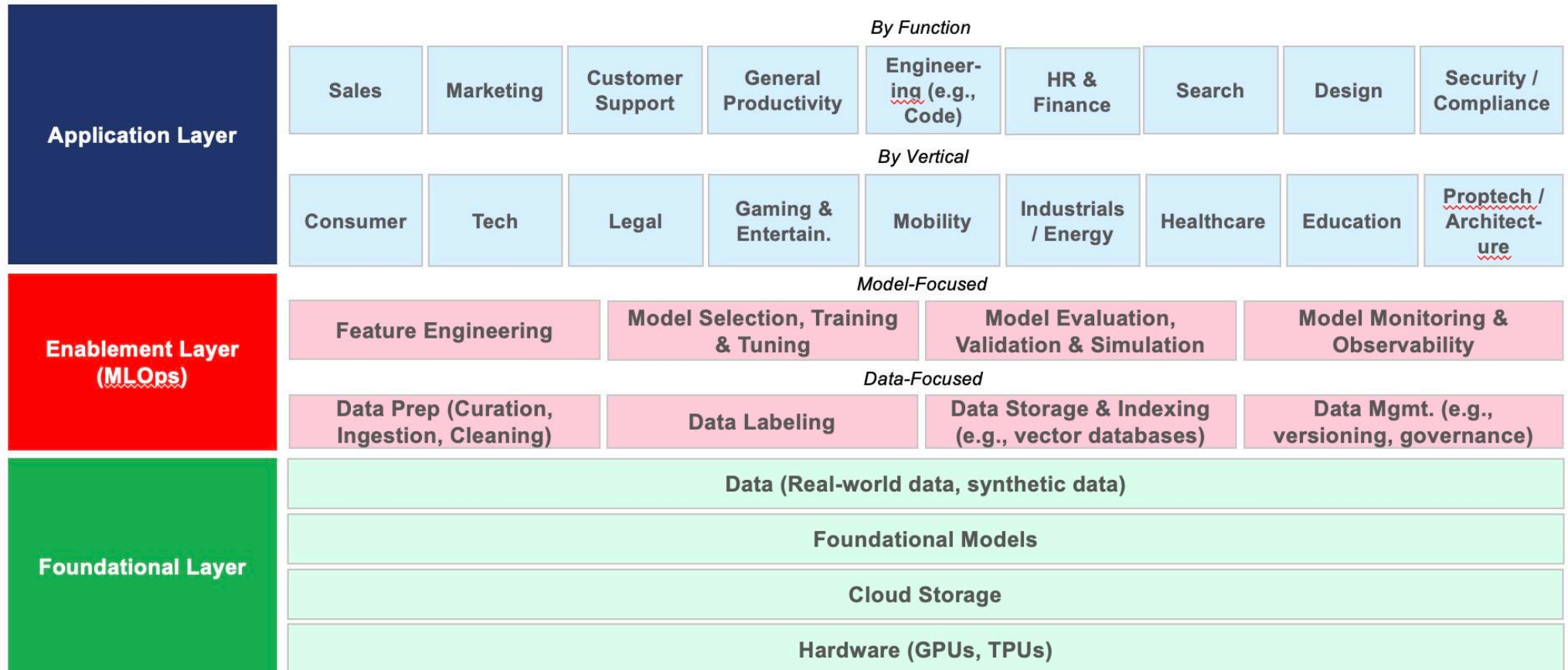
# MLOps Understanding

## UT mentality

- **Development Lifecycle:** Streamlined, ML development process from requirements to release with Development-, Test- and Release management practices.
- **Versioned Artifacts:** Implement Versioning systems to store and annotate several Artifacts of the ML lifecycle: code, data and models.
- **Automation of workflows:** Implement ML specific services to automate processes such as model training, data preparation.
- **Knowledge transfer:** Guide technological and support experts through the overview and details of the platform.

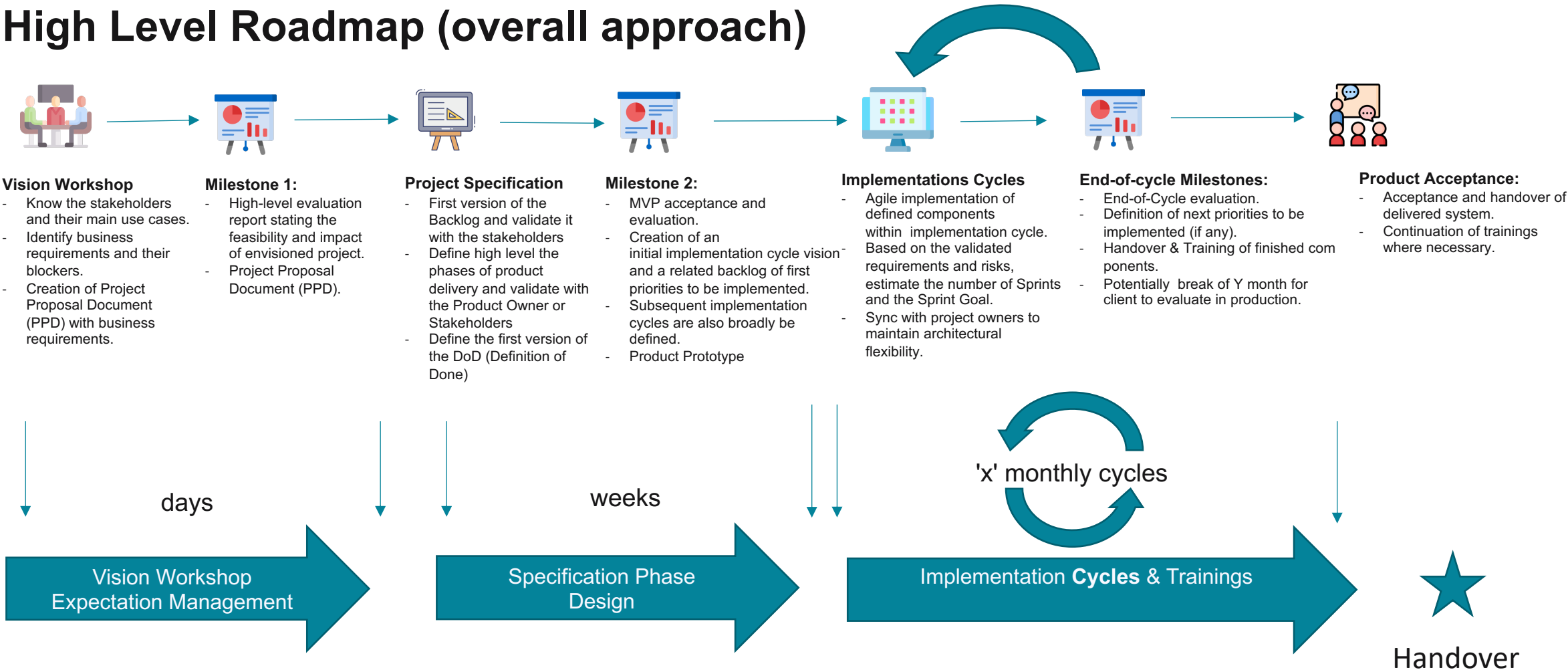


**Note w.r.t. ML Design cycle:** UT MLOps offering does not focus on / include the actual ML design, though could assist in creating a proper environment or platform on which the ML design cycle could be performed.





# High Level Roadmap (overall approach)





# 1. Vision Workshop (4 hours – 2 days)

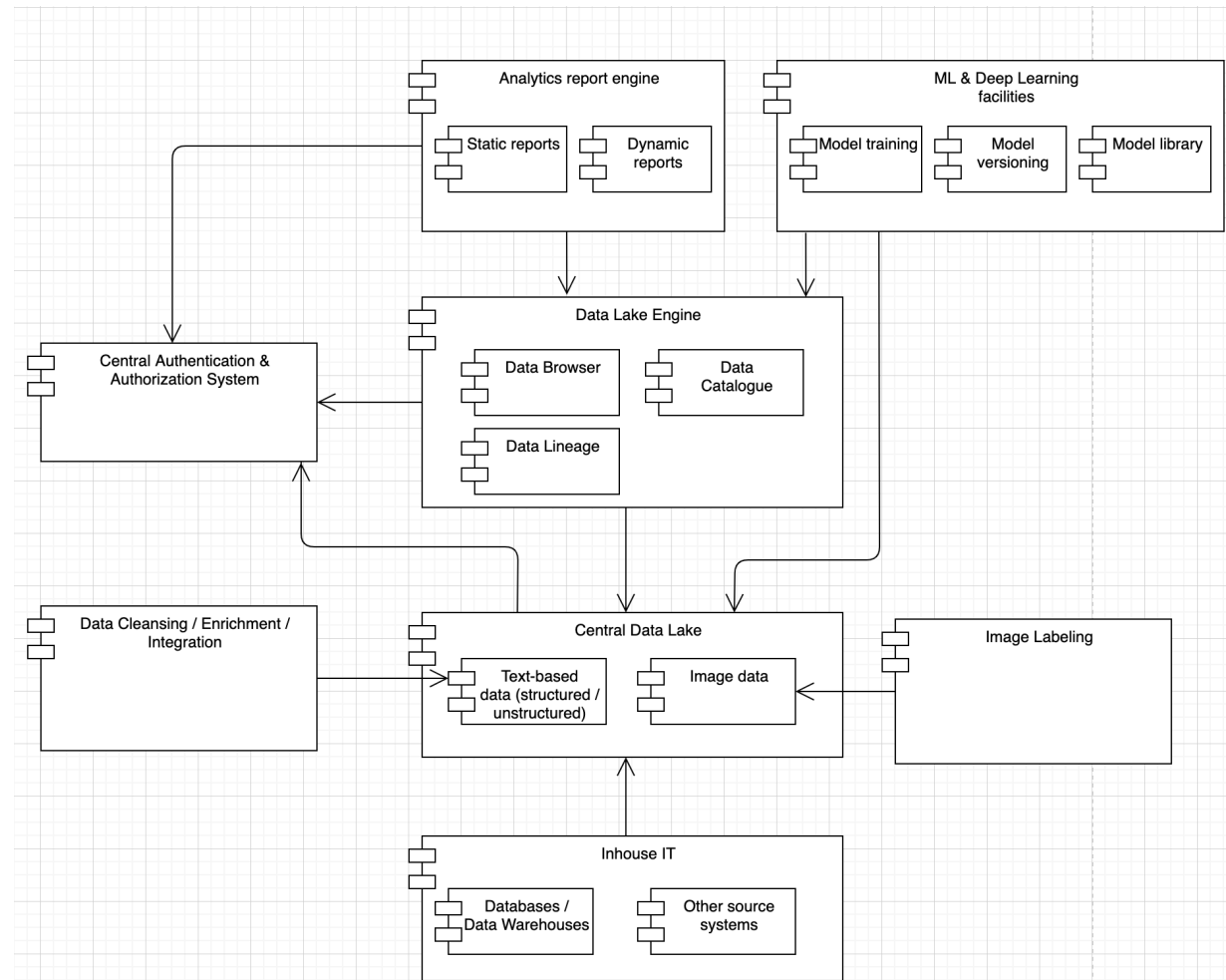
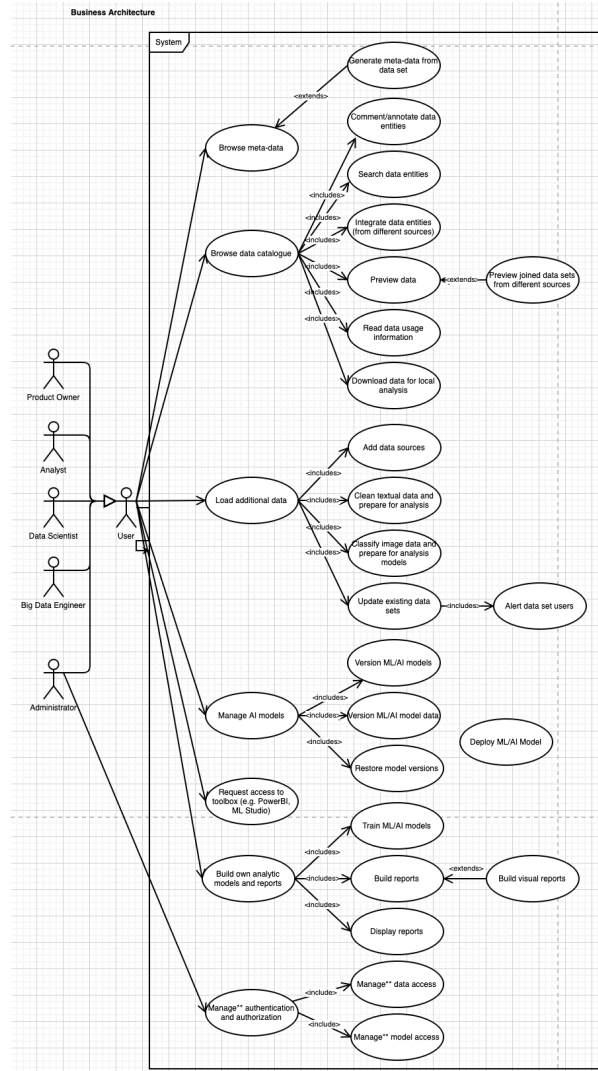
Step	Concept	Deliverables
1 - Strategy	<ul style="list-style-type: none"> <li>- Invite strategical stakeholders and domains representatives to a workshop (e.g. Organization Management, Domain representatives, Process Manager, Program Manager)</li> <li>- Exercise: Risk/Opportunity-Matrix</li> </ul>	Understanding and documentation of strategical risks, opportunities and expectations as well as business prioritization.
2 - Technical	<ul style="list-style-type: none"> <li>- Invite technical stakeholder to a workshop (e.g., Technical Management, Data Scientists, ML Engineers &amp; Architect, DevOps Engineers, Service Operations and SREs)</li> <li>- Exercise: Visualize high-level architecture and locate pain-points</li> </ul>	Understanding and documentations of technical risks, opportunities and expectations including pain-points and priorities.
3 - Showcase	Show principles and real examples of MLOps implementations. Focus on the different components and their functions.	Establish a common understanding of MLOps and different maturities as well as the cost, risk and benefit associated with it. Moreover, the Client need to be able to understand the functionality and how they potentially map to their usecases.
4 - Unification	- Combine all use cases created and confirm, reevaluate or extend it in accordance with the new input.	Prioritized Use Case Map, including high level <ul style="list-style-type: none"> <li>➤ Business / Functional Requirements</li> <li>➤ Architectural Requirements &amp; Priority of System Quality Attributes</li> <li>➤ Security Requirements</li> <li>➤ Infrastructure &amp; DevOps Requirements</li> </ul> Rate MLOps components according to MoSCoW
5 - Evaluation	Evaluate if a transformation towards MLOps is feasible and desirable and to what extent / to what kind of implementation level?	High Level applicability evaluation report for the identified use cases. Define several architectures according to use case, risk and maturity

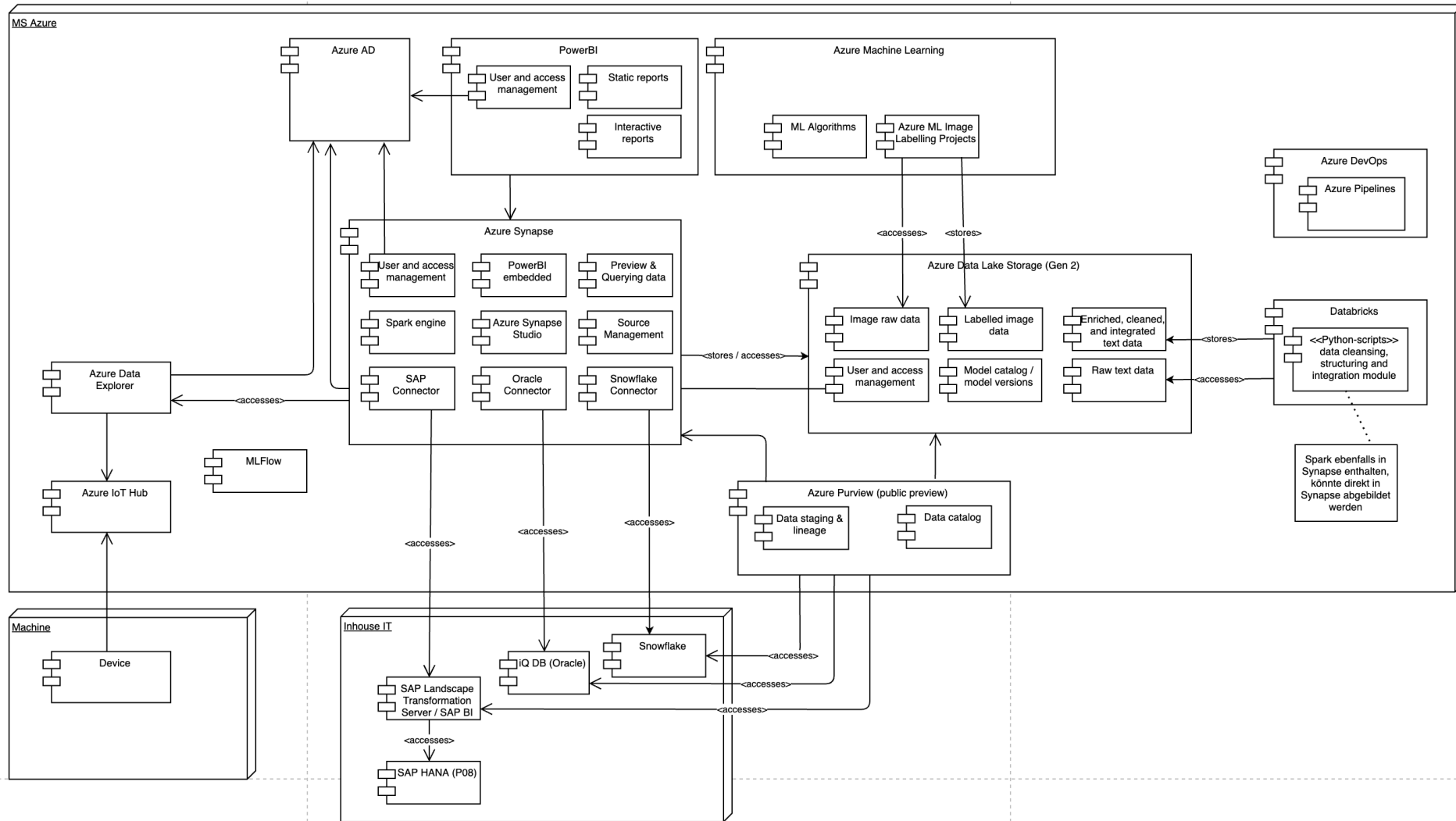


## 2. Specification Phase (4 – 6 weeks)

### Overarching Goals

1. Evaluation of components and the degree of automation
2. Definition of work packages
3. Assessment of risks
4. Estimate the effort and value
5. Implementation of Core-Components as an MVP
6. Evaluation







## 2. Specification Phase(4 – 6 weeks)

Step	Concept	Deliverables
1 – Evaluation	Evaluation of components and the degree of automation	Set of components and efforts that should be included in the target architecture
2 – Work Packages	Definition of work packages	Granular definition of packages containing effort, topology and required skill set.
3 – Risk Assessment	Assessment of risks and their mitigations	<ul style="list-style-type: none"> <li>➤ List of potential risk for the project and their mitigations if available.</li> <li>➤ Determine whether organization is ready for ML Ops projecting.</li> </ul>
4 - Estimation	Estimate the effort and value of all work packages and mitigations	Backlog of all efforts required to achieve the target architecture including their innate value, effort and risk.
5 - MVP	Implementation of core components as a MVP	The goal is to implement the most important and valuable components for the client's architecture and evaluate the impact and potentially re-steer the direction of the implementation phase
6 - Validation	Validation of the packages and architecture regarding the requirements	Offer including a target architecture, work packages and risks <ul style="list-style-type: none"> <li>➤ Business, Tech, Deployment Architecture (Security Architecture).</li> </ul>





## 3. Implementation Cycles

### Overarching Goals

1. Aim for implementation cycle of about 2 months, with a collection of sprints as part of deliverable.
2. Refining MLOps Stack
  1. Agile implementation of cycle vision according to evaluation of previous implementation cycle or deliverables of Phase 0.
  2. Special attention to standardization, automation and observability.
  3. Handover services and systems to the respective experts at end of cycle.
3. Training of Stakeholders according to their involvement level on components finished during relative implementation cycle.
4. Evaluation of result of collection of implementation cycle sprints.
5. Discussion with project owner on acceptance of cycle, update on customer priorities, (sharpening of) definition of subsequent implementation cycle(s).
6. If no more implementation cycles necessary, work towards final handover.



### 3. Implementation (Cyclical)

Step	Concept	Deliverable
1 - Implementation	Agile Implementation of the agreed work packages	<ul style="list-style-type: none"> <li>➤ Product Backlog Management.</li> <li>➤ Execution of the required planning, research, organization, code implementation, creation of documentations, SOP's and SLA's.</li> </ul>
2 - Communication	Communication of status, achieved results and blocking points, Documentation.	<ul style="list-style-type: none"> <li>➤ Update all stakeholder regarding the current state – including the recent progress and blocking point.</li> <li>➤ Document implemented components.</li> </ul>
3 - Review	Review of progress, processes and issues parallel to implementation work.	<ul style="list-style-type: none"> <li>➤ Any changes or additions will be estimated, prioritized and prepared to be included into the Backlog.</li> </ul>
4 - Redefinition	Review status-quo, sync with project owners / client, redefine project needs	<ul style="list-style-type: none"> <li>➤ Redefined work package for new implementation cycle adjusted to developing priorities of the client.</li> </ul>
5 – Restart	Restart in a new implementation cycle using redefined priorities and work packages. Optional break of development.	<ul style="list-style-type: none"> <li>➤ Any changes or additions will be estimated, prioritized and prepared to be included into the Backlog.</li> <li>➤ Handover current implementation and knowledge.</li> <li>➤ Client can test services in production and evaluate whether further implementation and services are required.</li> </ul>
Handovers and Trainings	Ongoing Handover of product during implementation cycle. Training of stakeholders.	<ul style="list-style-type: none"> <li>➤ Validation of product is in accordance with specifications.</li> <li>➤ Training of relevant stakeholders to the degree agreed.</li> </ul>