Bridging the Gap in Stakeholder Value Delivery in your ML deployment

Joy Joseph

Founder @ Syneractiv

At the risk of being Captain Obvious, let me start with saying that if you depend on data, Machine Learning (ML) is an important component of your production engine. It sits at the heart of AI-driven decision-making, enabling it to learn, improve and adapt from data, vs. just explicitly following programmed rules. These projects often fail to deliver beyond basic use cases, struggle to adapt to changing business needs, and incur frequent rework. One obvious yet often understated reason is that stakeholder needs, which are dynamic and vary across different groups and time periods, are not clearly captured and aligned with the ML model's development.

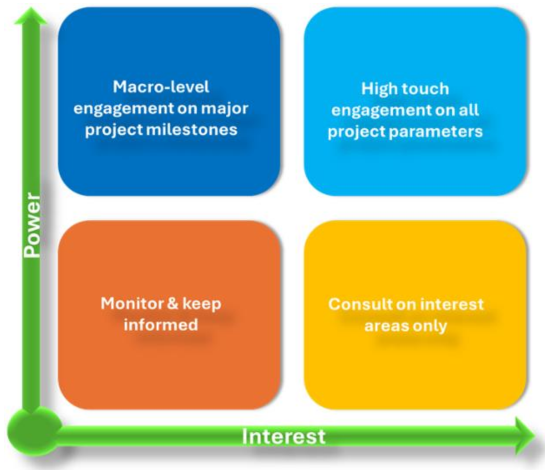
A simple yet effective solution to this issue is the creation of a Stakeholder Requirements Document (SRD), a well-established approach, traditionally used in Tech/DevOps but perfectly suitable for MLOps. The SRD ensures that stakeholder needs are well understood and integrated into the ML project from the outset. Without it, running an ML project is like embarking on a journey without a map or GPS—you may eventually reach your destination, but it will likely involve numerous detours and dead-ends.

Steps to Create a Stakeholder Requirements Document (SRD)

A strong SRD provides a clear blueprint for aligning ML models with stakeholder requirements and helps avoid scope creep, miscommunication, and misalignment. There are six key steps to creating an SRD:

1. **Stakeholder Analysis**

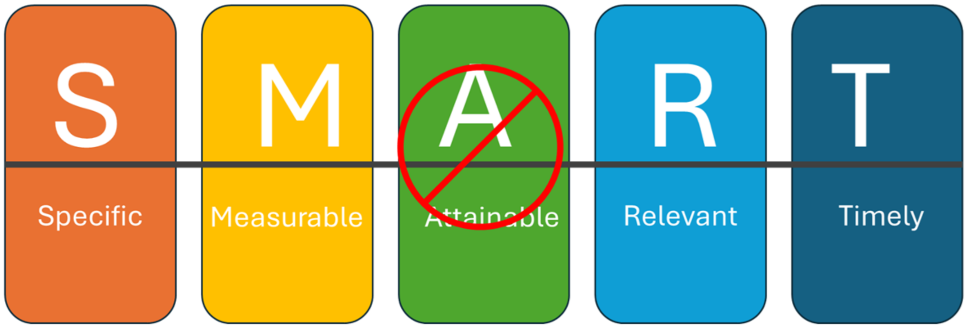
The first step is to identify all stakeholders involved in the project and their roles. A stakeholder matrix or McKinsey's DARE framework can help categorize stakeholders into roles like Deciders, Advisors, Recommenders, and Executors. A Power-Interest Grid can then help prioritize stakeholders based on their influence and interest in the project.



For example, project output users or privacy/compliance leads would likely have high power and interest, while the finance team budgeting the project would have power but less interest in the details of the ML model. Third parties responsible for executing model recommendations would likely have high interest but low power to directly influence the ML deliverable (think tech vendors, consultancies, agency partners and the like).

2. **Define Project Goals**

The project goals should be specific, measurable, relevant, and timely, with a direct business impact. Achieving this requires interviewing stakeholders to understand their expectations, KPIs, and how they intend to use the model's outputs. While traditional goal-setting frameworks like SMART emphasize attainability, it's important in ML to also remain open to innovation and evolution.



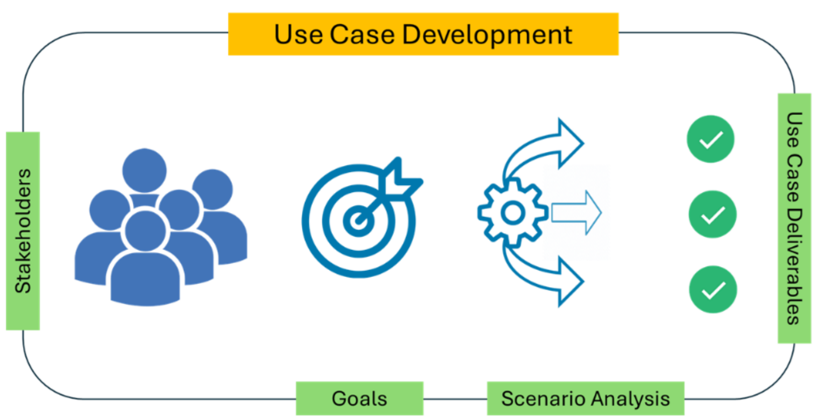
Basically…stop sandbagging your ML models by only focusing on low hanging fruit!

The future is unpredictable, especially in a rapidly evolving space like ML. However, goals can be future-proofed to some degree by accounting for potential shifts in market dynamics that may impact model behavior like data drift and feature evolution, the project can adapt to unforeseen challenges and keep up with fast-changing environments.

This doesn't mean solving problems that don't exist today but rather anticipating some obvious ones and leaving enough flexibility to allow the solution to be adjusted.

3. Requirements and Use Cases

The next step is to document the business requirements and use cases that the ML model needs to support. This can also be done through the same stakeholder interviews by adding questions on use cases.

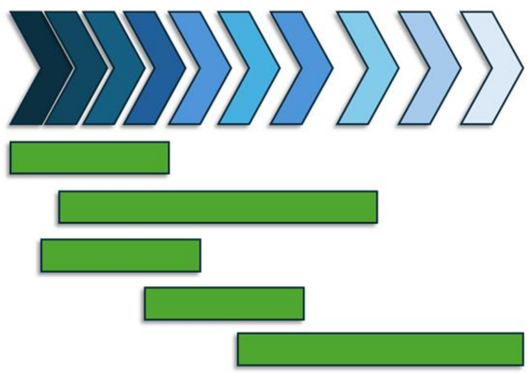


Use case analysis involves fleshing stakeholder goals with deliverables and turning them into concrete scenarios. This could include mockups or prototypes, which help stakeholders visualize the potential impact. Specific deliverables of each use case become business requirements.

4. Final Scope and Timeline

Once the requirements are prioritized, define the project's scope and timeline. Not all requirements can be addressed in a single project, so it's essential to identify and focus on the most critical ones. This prioritization process should consider both stakeholder influence and the business impact of each requirement. A collaborative meeting or voting process can ensure alignment on what features will be included in the final scope. The final scope document should be reviewed and ratified by all stakeholders to guarantee agreement and commitment.

Consider using a project management tool to manage milestones– I often see ML teams using Jira (Atlassian) to track project milestones but it may be worthwhile to integrate Jira with a tool like Asana to improve real-time accessibility of project progress to stakeholders.



One way or the other, GANTT it up!

5. Define Model Evaluation Criteria & Output

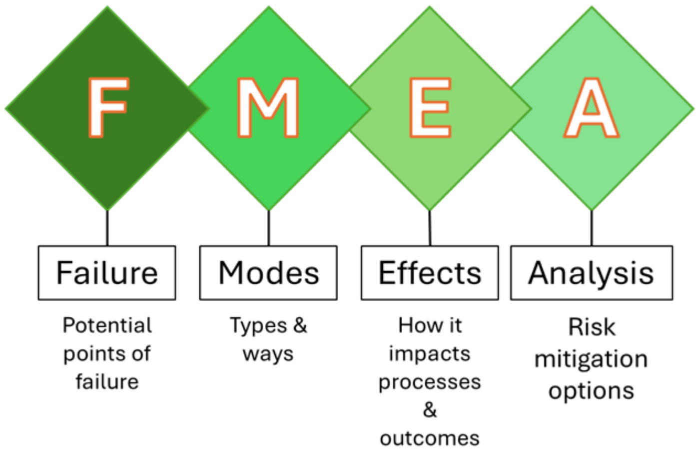
With ML, the evaluation criteria and output can be vast, but only the most relevant ones should be chosen. These depend on stakeholder priorities, whether they are focused on understanding the model's outputs, its predictive accuracy, or feature importance. For instance for explantory ML, sharing exploratory data analysis with stakeholders to elicit insights can aid more intuitive feature engineering.

If forecasting accuracy rather than explantory power is important, out-of-sample validation metrics become central. Similarly, regular model updates or adjustments based on stakeholder expectations should be accounted for, such as daily updates for fast-changing data or features.

It is also good to validate model metrics and output by evaluating them within each use case in step 3. A customer targeting use case might reveal the need for a boundary analysis to account for the effect of repeated exposure to the same audience that is skewing conversion predictions by shifting the decision boundary threshold.

6. Fail-safe and Trouble-shooting Protocols

Despite careful planning, unforeseen challenges may arise, such as shifts in data characteristics or feature assumptions. Reviewing possible post-deployment outages with stakeholders serves the dual purpose of managing expectations and receiving inputs on model failure scenarios. A Failure Modes and Effects Analysis (FMEA) is a great way to identify potential failure points and define contingency plans.



For example, if an ML model predicts patient behavior based on coupons that are tied to drug trials, a feature might inadvertently cause target leakage. A proper FMEA review with the right stakeholder can identify and preempt this failure mode.

Parting Thoughts

Creating a robust SRD is a non-trivial but essential step in any ML project. By ensuring that all stakeholder needs are identified, understood, and documented at the outset, the ML team can ensure that the final model delivers value and meets business objectives. The SRD process should not be rushed, and stakeholders should be educated early on about its importance. Taking the time to draft a comprehensive SRD will ultimately lead to more successful ML deployments that drive stakeholder value, ensuring that ML projects are not only technically sound but also strategically impactful.

Joy.Joseph@Syneractiv.com