From design to deployment

# Managing Machine Learning Projects

Simon Thompson

MEAP

MANNING

**MEAP Edition**
**Manning Early Access Program**
# Managing Machine Learning Projects
**From design to deployment**
**Version 8**



For more information on this and other Manning titles go to
manning.com

# brief contents

# 2

# Pre-project: from opportunity to requirements

**This chapter covers**

- Building an understanding of the type of project that's needed and the expectations of scale and structure that the stakeholders have.
- Setting up a presales/pre-project process.
- Understanding the requirements for the ML element?
- Understanding the data asset that you will work with.
- Understanding the general requirements for the project.
- Getting to grips with the tools and infrastructure that will be needed to deliver successfully.

Project success and failure is significantly defined by the pre-project / presales activity that surrounds it so – you need to get this right! The job is to move from the knowledge that there's an opportunity to get paid for an ML project to an authorized and budgeted job that you can use to pay your mortgage.

The purpose of this chapter is to lay out the activities and actions that need to happen to understand if an ML project is possible, if it's useful to do it and to figure out what kind of effort is going to be required to get it done (and who by). It's tempting to gold plate these activities because all of them could be done in deep, deep detail and would be all the better for that. Unfortunately, we live in a competitive world and sometimes there is little time or money to be spent on securing projects. Realistically, it's got to be the case that the organizational commitment that's needed to support deep dives into customer data or access to high performance servers just won't exist until the ink dries on the contracts. At that point and it becomes everyone's job to make it happen, but before then it's all just theory. Necessarily then the work that you do before funding is secured and time can be blocked out will just be a shadow of what will happen later.

Nonetheless, the less focus and effort that goes into this process the larger the risk that you, your team and your organization are taking on. Failure to understand the business requirements for the project risks your team misdirecting their efforts, and it risks you bidding too low to provide the resourcing that is required to deliver. Failure to understand the data resources that are available means that it will be impossible to make any kind of judgement about how to approach the project with ML, or the prospects for success. Failure to understand the security, privacy or ethical considerations means that you could be exposing you, your team and your organization to embarrassment and liabilities. Looking at all of these facets of the project now allows you to make some timely and effective decisions that could make life a lot better later.

In some ways these issues arise for any project, but there are some specific risks for ML projects that must be addressed.

- It's often easy to develop ML models, but developing models that have the right properties to be used to solve a particular business problem is much harder.
- Poor quality or inaccessible data is a major risk for ML projects, until data is obtained project progress will typically be stalled. Poor quality data introduces considerable friction and slows down project progress.
- Data sourcing and useage constraints can mean that it's unethical or illegal to use the results of an ML project. For example, if the origin of personal data is unknown, using it may be a violation of privacy and the effective owners of the data may not consent to its use.
- It's hard to predict the performance of ML algorithms in learning models a-priori. Despite the team's best efforts, results may be disappointing.
- Misunderstanding or not anticipating the IT architecture that the ML system will be deployed into in production can mean that the results of the project will be unusable.

Work that can be done to mitigate these issues is described later in this chapter and in chapter 3. As promised in section 1.5 of Chapter 1, a list of tasks that are required to deliver the pre-project activity are given in the pre-project backlog below. After that the work required to set up the pre-project activity is described before the work required to understand the requirements that the client has is outlined. Subsequent sections tackle understanding the Data resources, Security & Privacy & Ethics and IT architecture.

## 2.1 Pre-Project Backlog

Below is a list of tasks that have been developed to summarize the activities required to create the outcomes that are needed for the pre-project to be successful. This can be used as a "Presale Backlog." Each of these can be a ticket in a system like Jira or Gitlab which will then allow you to track progress and prevent tasks from being forgotten. Having used a ticketing system to track progress on this will come in handy then as it will be easy to see that a meeting should be run and easy to see who was responsible for each task and what they did.

| # | Item |
|---|---|
| PS1 | Set up a project backlog / task board and start using it. |
| PS2 | Create a project document repository and make available to project team. |
| PS3 | Establish a risk register, determine what's unknown in the diligence and the estimates and what would be required to deal with/mitigate. |
| PS4 | Create an Organizational Model to support your knowledge of the customer and the customers challenges<br>Undertake organizational analysis:<br>Map stakeholders for the project and the mapping of these people to the organization chart.<br>Map of impacts to organization chart: which business units will be affected?<br>Map of impacts to business priorities (ie. Increased revenue, decreased costs, growth of market) |
| PS5 | Understand the system architecture and non-functional constraints |
| PS6 | Get a Data Sample and document what is known about the data resources. Statistical properties of the data , Non-functional properties (scale, speed, history). System properties (where it is, what it lives on) |
| PS7 | Check & Document Security & Privacy Requirements – include as project assumptions. |
| PS8 | Check & Document CSR and Ethical Requirement.<br>Challenge and feedback and include as project assumptions.<br>Create PDIA & AIA document |
| PS9 | Develop a high-level delivery architecture.<br>The architecture should cover dev, test and production components (sometimes also pre-production/staging) and be able to support the customers non-functional requirements such as availability, resilience, security and throughput.<br>Try to qualify this architecture with the appropriate stakeholders for feedback.<br>Document key aspects of the architecture as assumptions for the project |
| PS10 | Understand the business problem: Use the consensus to build a project hypothesis, validated with customer and validated with delivery team. Ensure that this clearly communicated and documented in any contractual agreement. |
| PS11 | Undertake project diligence – will the stakeholders be available? Is the data available and manageable? What team members will be available and what skills do they have? |
| PS12 | Create an estimate of the work for a model project delivering on the project hypothesis that is required taking account of the team that is likely to be available and the scale of the work that is needed.<br>Ensure that all project risks are accounted for in your estimate |
| PS13 | Create and socialise team design & resourcing. |
| PS14 | Run a review meeting and go through a checklist to ensure that the presales process has been properly completed. |

Tickets PS1 ar PS9 is covered in this chapter, which deals with identifying and documenting the requirements for the project. PS10 to PS14 are covered in Chapter 3 which uses those requirements to create estimates and proposals to secure the funding and get the project ready to go.

So, the first thing to do is to hit PS1 by finding the right ticketing system to use and setting up this backlog. This could be Jira, Gitlab, Github or Microsoft ADO or many other options. As soon as you have done that you can sign off PS1! Congratulations the pre-project work is started, and you have made progress. Well done you.

PS2 and PS3 are next up as by setting up a project management infrastructure (building on the ticketing system) you're making it far easier for you to make progress with everything else.

## 2.2 Project Management Infrastructure

PS2 and PS3 are tickets that call for project management infrastructure to be set up and brought into use. As such they're a good place to start. They're listed below:

**Project Management Infrastructure Tickets**

**PS2**

- Create a project document repository and make available to project team.

**Project Management Infrastructure Tickets**

**PS3**

- Establish a risk register, determine what's unknown in the diligence and the estimates and what would be required to deal with/mitigate

The first step, as per PS2 is completing the pre-sales process is to create a shared project document repository where the documentation covering the presales activity is to be kept. The repository might be used for the whole project, although customer data retention and management requirements may mean that a migration to another, customer owned, and standardized repository might be needed. However, the information gathered at this step of the project will be extremely useful right through to the end of the delivery, and probably beyond so being organised about documentation from now on is crucial.

One thing to remember is that your organisation likely has a document retention policy; this may require the destruction of documentation after an particular period or at the end of the project. Alternatively, it may require that the documentation is archived so that it can be found later. Retention policies should be checked, but the information gathered at this point of the project is likely to be the property of your organisation. If pre-sales fails and there is no

project proper, then these documents will be useful if the customer returns with another project in the future.

But, importantly in all cases the documentation developed and captured now will support the development of your team and your working practice. By getting this done you are capturing value from day one, but you are also helping yourself in the future. It's really common to think "oh, I came across a problem like that before and then we decided…" If you can remember that and pull out the documents, you'll find that you've got a real leg up for whatever you are doing at that moment in the future.

The other thing to do on day one is to set up a risk register. Determining & surfacing what might go wrong and what's unknown is one key step in creating a project that is manageable. This is a way to prevent important issues from being forgotten and it's a way to establish the difference that the work on the project is making. As you move an identified risk from live to retired problems are solved and they've been solved by you and the team. Whoot! You did a good thing.

Risk items can be handled by turning them into questions to be explored; if the objectives of the project are substantially defined in terms of questions that need to be answered it's substantially less risky. This approach exposes uncertainty that will have to be dealt with before business value can be created. Exposing questions in this way informs customers of the value of the exploration that is going to have to be done.

So, setup a project risk register. This sounds like a complex and fancy thing, but actually it's really simple. A risk register is a document identified & versioned in the document repository (of course!) which records all the risks identified and the actions identified and discharged which are aimed at mitigating them. If the actions are successful, the risk register will also record that the risks are mitigated and discharged from the register.

In the project proper the identification and management of risks is part of the project heartbeat (about which more soon) and managed in a weekly meeting with the key project stakeholders. All parties accept the entry of new risks onto the register and agree that they are dealt with or not. In the presales process risks are managed closely by the presales team, they are the concern of the project team as assessing and controlling the project risks at this stage defines the estimates that the team can provide and will underpin the decision by the client to adopt the team's proposal or not.

## 2.3 Understanding Requirements

Having set up a working project infrastructure with the ticketing system, document repository and the risk register the real work starts. PS4 calls for the development of the requirements of the project, it's shown below to make it easier to see what's got to be done.

**Requirements Tickets**

PS4

- Create an Organisational Model to support your knowledge of the customer and the customers challenges
- Undertake organisational analysis:
- Map stakeholders for the project and the mapping of these people to the organisational chart.
- Map of impacts to organisational chart: which business units will be effected?
- Map of impacts to business priorities (ie. Increased revenue, decreased costs, growth of market)

PS5

- Understand the system architecture and non-functional constraints

You need to get to know your customer and figure out what they are going to need to happen by the time their budget is spent to call this project a success. This knowledge will enable you to sign up to the spirit of the project as well as the letter of the contract, and it will make negotiating and managing change in the project much easier and less fraught.

### 2.3.1 Funding Model

The first challenge is to understand the funding model for the project. There are three types of projects: fixed price, time and materials, and mission driven. Fixed price and time and materials projects are aimed at delivering a specific outcome which will often be defined up front. Mission driven projects are more exploratory and are aimed at improving the performance of an area of a business – or for a smaller business (or bigger project) transforming it overall. The type of project being delivered has a big impact on the way that it should be managed and the approach that should be used.

Because a fixed price/fixed time project should deliver the defined result in a specific time, the risk of the project is borne by the organization delivering it. It's important to note that there are two ways that the risk can materialize when the project goes wrong. On the one hand, the team will experience "crunch" and overwork to deliver the goods, on the other hand, the costs of the project will escalate and damage the business that is delivering it. Usually both these things happen – the team goes into crunch the cost is borne by that team, management think it's got away with it and then someone leaves, a new person has to be brought in, and because it's no longer "on them," the costs materialize for the organization. Verheyen discusses the challenge of dealing with fixed price contracts using agile approaches and concludes that fixed price contracts are so problematic as to be simply immoral! (Verheyen 2012)

Despite their pathologies, fixed price/results-oriented contracts are a business reality that many delivery teams deal with every day. This is because they provide a well understood mechanism for creating a contract that customers will sign up to and can authorize payment for. In fact, this type of arrangement is so simple that even where formal contracts don't exist, working to a fixed resource/fixed time structure can provide clarity and produce buy in from business stakeholders. But the biggest virtue of working to fixed price/fixed time (and approximately fixed outcome) structures is that it's transparent. The team knows what they are signing up to, the customer knows (as much as a customer can know) what they are going

to get. The trade-off is that the risk of fixed price projects is largely shifted onto the delivery team. The team can end up being pushed to make up for misestimated or mispriced projects by working overtime; as a team leader, you need to guard against this with the investigation and preparation you do before the project starts.

Projects on a time and material basis work based on the customer paying the team until the project is finished – whenever it is finished, or the customer runs out of money. Time and materials projects have their own pathologies. For example, it's easy for unrealistic expectations to be set and for the project team and other technical stakeholders to be unaware of the real expectations and goals of the project until a late state. This in turn ends up precipitating a situation where the pressures on the budget holders spill over to the team and they find themselves with impossible targets and demands – or just a project that is deemed to be a failure. However, it's generally agreed that time and materials project risk is much more shifted towards the client stakeholders.

Contrasted to fixed price or time and materials, a mission driven project can sometimes seem like a dream. The team has a high-level mission and an agenda of ideas and hunches that are supported by their stakeholders. These are the things that the team goes after on their odyssey to get a result. In the best case, as the team gets deeper and deeper into the project they see more and more opportunities, but in the worst case they see more and more problems. Folks in the team can become energized and engaged because of the importance and value of the work - they can come to see themselves as having agency and importance through saving the business etc. On the other hand, folks can get switched off by the never-ending story of false starts and disappointing outcomes. Sometimes the achievements of the team are understood and acknowledged, but sometimes the achievements are subsumed into other business initiatives.

If the project you are investigating looks likely to be funded on a time and materials basis, or if it's a more open and mission driven project, then this chapter and the next may be less relevant to you and your team. On the other hand, using a structured pre-project process helps for all three types of projects: for fixed price it's giving your organization evidence for making a bid (or no-bid) at a particular level. For time and materials projects, it's a way to limit the dangers of crunch and stakeholder frustration. For mission driven projects, it's a way of focusing the team and helping the team and the organization to manage and structure how they are going after the strategic opportunities that they want to grasp.

### 2.3.2 Business Requirements

Having established the funding model for the project and made the decision to do a structured pre-project investigation, the next step is to look at the business objectives that the customer has. Requirement's analysis is often seen as part of "big design upfront," but for an ML project there are some issues that must be understood to determine if the project is practical at all. No matter how agile the team is they will not be able to make a very large general model execute very fast for a lot of users on an old slow processor cheaply.

So, there are three general types of requirements that need to be understood.

- Functional requirements: what will the system do, and for who? What is the function of the models that are going to drive the delivered system? What classification/recommendation/labelling task are the models built from the customer's data expected to do? How well must the models perform in terms of accuracy, robustness to strange events and data and reliability in the face of change?
- Non-Functional requirements: how fast must the models go? What throughput is required? What latency must the model react within? How much must executing the models cost in terms of money and carbon?
- System requirements: where is the model to live? How will it be maintained? What systems must it integrate with? How will the results of the model be consumed and what work is required to make it consumable? What resilience/business continuity measures are required?

It isn't realistic to tackle the list above (functional, non-functional and system) in order, instead a process of clarification, reflection and deepening of understanding is needed. Then with this picture in hand you will be able to pin down the specifics in terms of the list above, and what the implications of providing it in a particular way will be. So, how to start?

The obvious first step is to listen to what the client or sponsor says that they want. This may be articulated at a high level, and it may be that the client lacks the technical background to describe their requirement in a way that's technically achievable. On the other hand, you might get a detailed and coherent specification straightaway. Having listened carefully to the clients understanding you need to go deeper.

#### BUSINESS REQUIREMENTS: WHY?

It's most important that you ask why the customer has the needs and objectives that they have articulated. If you can understand this then it becomes possible for you to do several things:

- Fit the customers' requirements to technically realisable solutions.
- Refine the customers' requirements to provide more value.
- Develop several alternative routes to value to be explored during the project.

Imagine a customer who wants to create a smart building: the articulated objective is to develop models that use the sensor data gathered throughout the building to control the heating and air-conditioning more efficiently. Why?

The answers could be:

- To reduce costs.
- To improve the environment within the building for its users.
- To reduce carbon consumption.
- To reduce the use of particular chemicals in the air conditioning.
- To improve the image of the building company.
- Because we've been told to do it.

All of these are valid reasons for the objective; all imply potential alternative solutions.

The next question to ask is who?

### BUSINESS REQUIREMENTS: WHO?

Something simple to do is to obtain an organisational chart for the customer: what is the organisation that you are delivering to and where does the customer sit in it? An example chart is shown in Figure 2.1 below that identifies the departments and responsibilities that are important to The Bike Shop customer. The people who are trying to get the project done are in IT, but the end users are in the manufacturing organisation and in the marketing and ops sections of the retail department. The project touch points are indicated with dots in Figure 2.1, and it's interesting to both see where the project is going to rub up against the customer's organisation and where it's not.

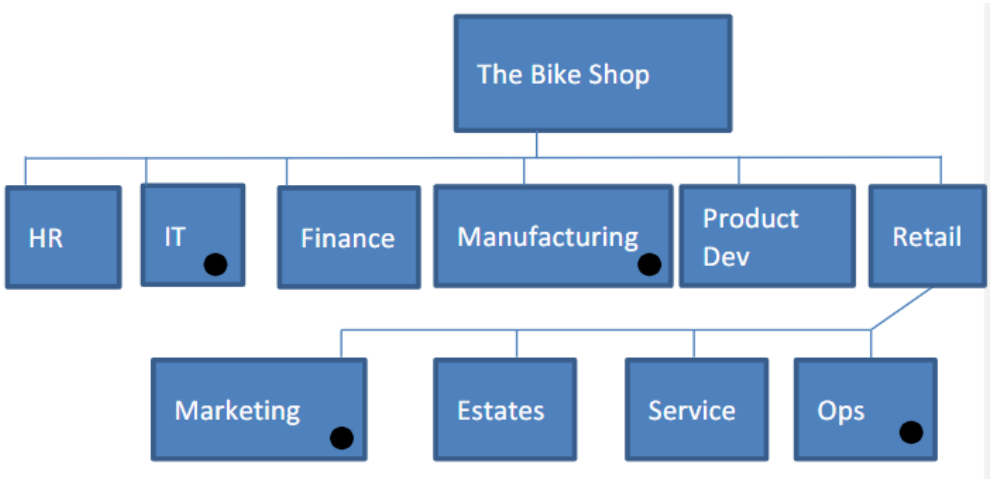


Figure 2.1 The Bike Shop Organisation Chart. Black dots represent stakeholders and users.

Locating and decorating an org chart with the names and roles of the contacts and users is a good start, but a deeper understanding can be built using more formal tactics.

The concept of building an Organisational Model comes from the CommonKADS knowledge engineering methodology (Guus Schreiber 2000). They propose developing a model of your customer iteratively focusing on the issues that have brought you into the engagement:

- Problems and opportunities: a shortlist of the perceived problems and opportunities that the customer has used to justify the engagement with you.
- Organisational context: the attributes of the organisation that put the problems into perspective; mission/vision of the organisation; external factors effecting the organisation (competition, regulation, economy); strategy of the organisation; the value chain it sits in (who does it buy from, who does it sell to, what are the ultimate customers and producers of the goods and services that they organisation deals in.
- Solutions: ideas about the solutions that you might offer.

In the KADS world it was suggested that this knowledge can be obtained from the stakeholders in the organisation. KADS uses a quite outmoded hierarchy in their stakeholder map. Things have moved on, so in a modern organisation the stakeholders might be:

- Budget holders – these are the shot callers that you bring value to, they might be your customer but also the folks from the finance organisation or procurement who have to agree that the money allocated to the customer has been spent properly or that the organisations procurement policy & standard has been complied with.
- Business experts – folks who understand the domain and how your system will connect to their business.
- End users – the people who are going to be using your system and will be impacted by it
- Security signoff – the person who will agree that your system is compliant with the organisational security standards you must hit
- System signoff – the person who will agree that you've designed and implemented the system compliantly.
- Data admin – the people who can give you access to the data resources you will need
- Data protection signoff – the person who will agree that you've handled the data compliantly
- QA signoff – the person who will agree that you've achieved a working / quality system

Many of these people have a veto on whether this project is going to be a success or failure. The challenge is to identify them all and to then figure out what it is they are going to demand from you and the team. Determine who they are, what they want and how to talk to them.

This is an intimidating list. Realistically you will have to prioritize who it is that you are going to approach. Working with the stakeholders in your organisation (not the customer), make sure that you are licensed to engage with the people who you have identified. The engagement managers and account development executives that are running the opportunity in a consultancy will not thank you for derailing a bid by approaching someone you are not allowed to talk with for commercial reasons. If this is an internal project there will be politics to be considered, funding is competed for, and approaching the wrong stakeholder at this point could lead to the project being vetoed before it has a chance to form.

Once you do have a qualified list of contacts there are basically two questions that you need to get answers on:

- Organisational context: what is the mission that their unit has? What are the sources of business pressure (regulation, competition, supply, disruption)? How do the customers make their money and justify their place in their organisation?
- Problems and opportunities: why does the person that you are engaging with need a solution? Could they be more productive? Are they wasting time on repetitive tasks, are they unable to take good decisions because they lack information, are they overwhelmed with options, do things move to fast?

The answers to these questions will define the space of needs and opportunities for the functional part of the project. Of course, if there is one obvious and clear functional need ("a system to do X") then all is well; that's the functional requirement. It's likely though that things will be a bit more obscure and you will get a range of needs and ideas. That's ok, by

understanding the constraints on what can be done you will be able to synthesis, qualify and select from the laundry list that you've created with this task.

This leads to the next clarifying question: what?

#### BUSINESS REQUIREMENTS: WHAT?

The first place to start the process of figuring out the "whats" of the system is to try and get a handle on the system/IT architecture in the client organisation and to begin to get a handle on the non-functional requirements in terms of scale and speed.

The first step here is to get an understanding of the IT architecture. Unfortunately, for the last 30 years or so it's not been possible to get a nice chart to represent large organisation's IT setups. It's common for companies to have hundreds or thousands of applications (or even tens of thousands in some cases). So, what's essential is to understand the general policy and facilities that are currently in force in the organisation and the legacy facilities that might impact the project.

The key questions are:

- What kind of data systems are in use: Hadoop? Presto? Oracle? SAP? Is there a single vendor policy ("we are a Microsoft shop") or is there a user / application first policy ("any database so long as it works")?
- What processing systems are available: SPARK? Kubernetes? OpenShift?
- Is there something missing? Is there any vital infrastructure that you think should be available for use, but isn't there? Will that be an issue and what impact will it have on the project and the possibilities for a viable solution?
- Is the organisation on the cloud? Which cloud? What are the policies about using relevant components in that cloud? Often organisations choose not to use some components for cost and security reasons.
- Are there legacy components that will have to be interacted with? For example is some part of the relevant architecture on premiss, while the cool new stuff is in the cloud?

Then you need to understand the scale of the business challenge so that you can determine will the infrastructure that's available be up to the task at hand:

- How many customers are there?
- How much do they spend?
- How many transactions a day does the organisation run?
- How many parties are involved in a typical transaction?
- What are the key trading hours for the organisation?

It's a running certainty that many more questions will come into focus as you investigate more. By getting the answers you will create a picture of the operating environment and the landscape that the system to be created will have to function in. This picture, and the understanding of the functional needs of the system will underpin the creation of the project hypothesis. What is the solution that is going to be the goal of the project by the customer and the team. It'll also be enormously helpful in framing user stories and pinning down more specific requirements as the project evolves. But for now, the model you are working up will tell you and your organisation if there is something real to do with ML, and if that something is feasible in the context that the ML system will be operating in.

Because you are working on an ML project rather than just app development there are a bunch more specific questions that you need to get into. These are investigated in the next few sections staring with the big one in section 2.4. What kind of data are you working with?

## 2.4 Data

Data, Data, Data. People doing ML projects have got to understand the data!

By getting information on the data now it's possible to get insight into the scale and depth of the challenges that the team will face, and what they can really do. There's the core business of understanding the characteristics of the data in statistical terms, but also the data engineering that will be required to set the implementation up, and what the limitations or potentials of that are. PS6 requires that you get an insight into the data that you are going to have to use in this project.

**Data Discovery Tickets**

**PS6**

**Get a Data Sample and document what is known about the data resources.**

- **Statistical properties of the data**
- **Non-functional properties (scale, speed, history)**

**System properties (where it is, what infrastructure it lives on)**

Your client may have a clear idea of the data that you are able to use to train ML with, but it's valuable to delve further into their knowledge of what is available so as to validate and develop ideas about what kind of ML solution might be possible. There are four benefits of doing this:

- The first is that by asking open ended questions about the data available in the client's systems data sources that might not have seemed relevant to the client can be discovered and put to good use.
- The second is that the data sets that are known to the client and are recommended to you can be explored and validated, even if only in a narrow and simple way at this stage.
- The third benefit is that getting some idea of the deficiencies of the data that the customer has to hand can inform you of the need to find data from open source or commercial sources to supplement it.
- Finally, you can get some information about the work that will be required to use the data, both in terms of improving the quality and cleaning it, and by employing methods that squeeze more out of limited data sets.

The first thing on the list is to get a sample of what you are going to be working with. Getting the full dataset would be ideal, but at this stage this may be unrealistic for several reasons, the technical difficulty of extracting a large dataset may be very great and require

funding which may be unavailable at this stage. Additionally, the full data set may contain trade secrets and other intellectual property that cannot be released until a strong contractual relationship is established, and typically the security requirements for access to corporate data stores can't be negotiated in a putative project. Finally, the work required to handle and manage the data may be very substantial, and unaffordable at the moment. However, getting a representative sample should be feasible and extremely important, even the process of obtaining the sample itself may reveal important issues with the customers understanding of the data and data infrastructure.

If the full data is available and the project scale and risk provide commercial justification to pay for the activity (after-all we are still in pre-project, so this is on your coin at this point), then you may want to drag the whole, or parts of the Data Investigation and EDA (Exploratory Data Analysis) exercises from later in this book forward to pre-project. After all, the more depth that you can afford to get now, the fewer risks you face later.

Realistically though, it's much more likely that at this point all that will be available will be a sample.

Questions to ask about statistical properties and things to look for in the data sample include:

- Is it really representative? Are there some data points from across the time period that the data was accumulated? Are there data points from all the source systems? Are there some data points from the extremes of the data ranges?
- What is the range of values in the entities in the sample? Is it sparse with very few or very dissimilar items? Or is it dense with a lot of repeating values?
- How was the data collected? Was it part of a survey? Was it from an experiment? Was it exhaust from a business process? Is it picked up at regular intervals or because of an event?
- Does the data remind you of other data that you have previously used? Is this data that is suitable for processing by a well know ML algorithm without extensive transformation? For example, if it's image data then is it 256*256 pixels with eight colours, or is it Gigapixel images with 2.4M colours? What well known data set is it "like"?

Non-functional questions to ask:

- What is the scale of the data available? If the source data is very large, then the sample provided may be unrepresentative of most of the data even if it is sampled in a sensible way from the source – and often sample data is not systematically sampled.
- How many different underlying data assets or tables were brought together to create the sample? How long did that take and what was the cost / time spent on the queries? Was this data that could be picked up easily from the corporate information architecture or was it prized out by heroic, patient, and ingenious means? Did any of it come from third parties or exotic sources?
- How fast does the data change, how often is it updated and how much arrives how quickly?
- What is the schema of the data assets used to create the sample set? Sometimes samples are provided as large flat tables that are joins from underlying databases – understanding what the source schema is can show where there are problems with this process and show where effort is going to have to be invested in the ETL process to leverage this asset in the project.

Systems questions:

- What platforms host the data; in particular, do your team have the skills required to access and manipulate the data on these platforms? If not how is it expected that the data will be made available to the team?
- Has anything significant happened to the data sets in their life cycles? For example, has there been a data migration or a data quality improvement campaign?
- Which business unit/stakeholder owns the source tables and the derived data, and what organisation owns the system that implements and manages the data tables used? This knowledge will define the investigation into the IT systems context that the team will operate in, the security & privacy regime required and will be significant in the development of the ethical approach to the project required.
- What was the process used to prepare the sample? Although the process used can be guessed at from the answers to the above questions it's always worth trying to get some documentation about this, especially to understand if there are any manual steps like "then we picked through the items to discard the ones that don't fit". In the first phases of the project proper the team may want to reproduce the sample provided to test their understanding of the resources – is there sufficient information to do this?

Unfortunately, sometimes customers will be unable and unwilling to disclose real data during the pre-project process. This can be because of contractual concerns or simply because their infrastructure requires work before the data can be extracted at all. Many customers will simply have no idea of how to get at the data that's required. The folks that you are talking to probably won't know the answers to all your questions and there won't be time to get them. The solution: put these items on the risk register, and if you are working to a contract make sure that they are written into the statement of work as assumptions. But these are really big risk items, essentially you are flying blind on the ML part of the project unless you have a good idea of what the data that is going to be used is like.

If this is the case, then an alternative option is to push for a short project to understand the ML readiness of the organisation. This should provide the contractual cover to support the extraction and inspection of the data, including provisions on privacy, data retention and use and undertakings on security and data handling. This work will pay for the team to have a strong enough understanding of the data to have some confidence in predicting the kinds of results that they should get in the modelling phase of the project.

Despite the challenge of getting access to data resources at this point in the project, it is essential that every effort is made to both understand and document the data model that the team will be working with and to apprehend the real characteristics of the data. Attempting to dimension and structure an ML project without sufficient knowledge of the data is extremely risky. So, if these tasks are not well bottomed out, it is important to introduce significant contingency items into your project estimate, both in terms of clock-time and funding in order to make sure that your team isn't desperately coding round huge data problems late at night for weeks on end. Bear in mind, if no one knows "what's in there," then that's a strong indicator that there are real problems waiting to be found.

## 2.5 Security & Privacy

**Security and Privacy Tickets**

**PS7**

- **Check & Document Security & Privacy Requirements – include as project assumptions.**

ML projects are tightly coupled to data resources – often sensitive and important data resources that many business processes depend on, or that contain details that are both protected in law and private to individuals.

Any insecure project can create vulnerabilities for an organisation, so it is natural that an ML project will need to meet the security requirements of the organisation(s) that it is working with. To achieve this, it's necessary to engage with the security infrastructure of the target organisations as quickly as possible. In the pre-sales phase of the project the objective of this engagement is to gather the information that will be required to assess and factor in the impact of security constraints and requirements into the project. The sign-off for the security aspects of a system is often handled by a different organisation. Sometimes the security organisation in an organisation is completely decoupled from IT with a CSO reporting directly to the CEO.

In the best case there is a single security stakeholder that can be identified and engaged in the client organisation. More often there will be several security stakeholders that need to be involved in the project.

Figure 2.2 shows an example of the kind of security organisation that may have to be engaged in an ML project. Data sets can be required from several lines of business, including the line of business that has engaged the team for the project. Additional data may be required from group operations – for example pricing and costing information. A cross cutting concern

is IT security, for the development (dev) infrastructure and activity – but also for access to the production platforms (prod) that will be required to deploy the system.

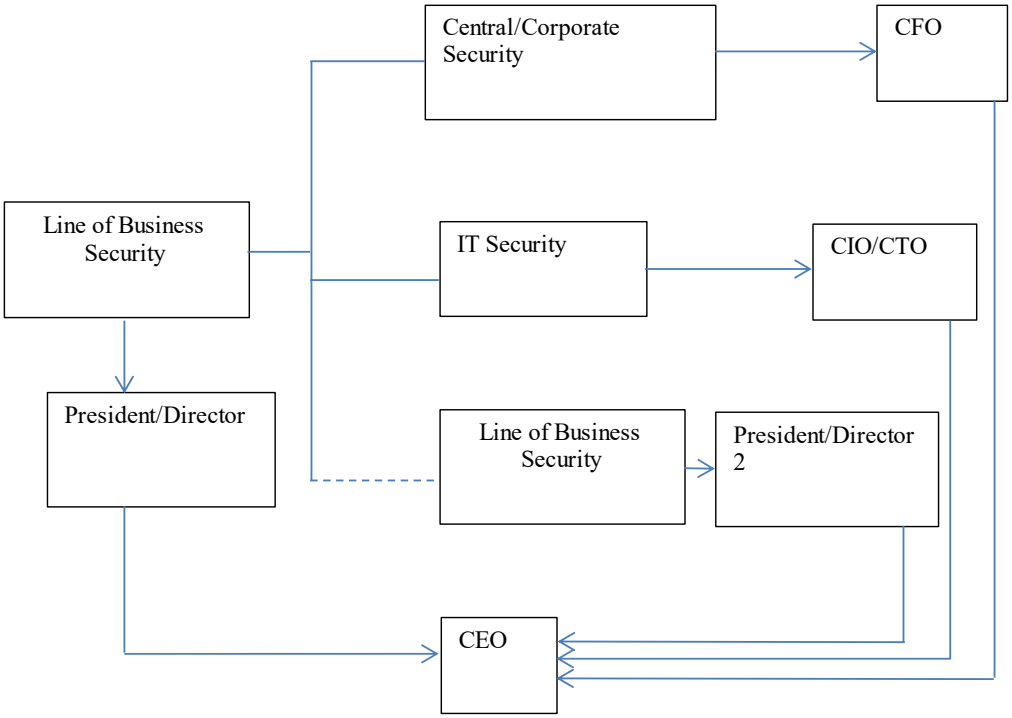


**Figure 2.2 an example of a security organization within a large organization. Each market facing unit owns operational security relevant to that line of business, group functions also have a security function reporting to the CFO & there is an IT security function reporting to the CIO or CTO**

For each of the core data sets, relevant organizations and IT platforms it's necessary to establish who the security stakeholders are. Also, you need to understand what the data privacy issues and requirements are (at top level initially) and what the requirements and processes that will have to be negotiated are.

Equally important is to establish, preferably with the security stakeholders in person, what problems are likely to be exposed during the security processes. Often security folk will be able to say that the requirements are straightforward and unlikely to be a problem. If, on the other hand they are unwilling to give that steer, that it's a sign that a significant hitch could be in prospect. Determining what would need to be done to deal with that problem may be impossible at this time but entering it onto the project risk register is vital. Either the resolution becomes a contractual assumption that provides the team with cover and scope for flexibility, or it becomes a financial problem that needs to be considered carefully when assessing whether this project is viable and how much it may cost.

## 2.6 Corporate Responsibility, Regulation & Ethical considerations

Just as for security, many readers will reach this part of the book and say, "this should be the first thing that's considered", and they are somewhat correct. But it's hard to start thinking about CSR and Ethics before you understand what the project is really trying to do. So, once the project hypothesis is clear in your mind it's time to think critically about whether what you are doing is going to end up collapsing society, killing someone, destroying the rainforest or will simply result in your public humiliation, prosecution and incarceration. This is the task in PS8.

**Corporate Responsibility, Regulatory and Ethical Consideration Tickets**

**PS8**

- **Check & Document CSR and Ethical Requirement.**
- **Challenge and feedback and include as project assumptions.**
- **Create PDIA & AIA document**

Ethics are important in an ML project.

Laws and legal duties such as the European GDPR impose a hard boundary on what you should consider doing with ML. At the present time there isn't much specific legislation on ML systems per se. There's a lot of confusion about definitions of "algorithms", and how they should be regulated, but it is very likely that this picture will change in the future. When it does, being cognizant of the relevant laws will be very important and useful. Bear in mind, it's clear that a failure by a team to understand and follow legislation because of ignorance will be seen as just as bad as a deliberate attempt to flout or circumvent the rules.

It's also important to be aware of any laws that apply to domain that you are working in, as well as generic data and ML laws that are in force in the relevant jurisdictions. For example, in medicine, legislation on patient safety and testing may be relevant, in finance legislation on risk and process, and in industry legislation on health and safety. It is necessary to investigate and clarify what if any domain specific legislation will apply to the system under consideration.

But, just sticking to the laws relevant to the project isn't going to be a strong enough approach to create a good outcome for the client, your organisation, and your team. The ICO (Information Commissioners Office) in the UK has developed a framework for the audit of AI ML systems by organisations (ICO UK 2020), and this guidance stresses that AI & ML systems must be accountable for data protection, and this must be demonstrable. The system (in the ICO's view) must:

- Allow the customer to be responsible for compliance
- Allow the risks of the system to be assessed and mitigated
- Allow the documentation and demonstration of how the system is compliant and justify the choices that have been made.

The ICO also notes that "due to the complexity and mutual dependency of the various kinds of processing typically involved in AI supply chains, you need to take care to understand and

identify controller/processor relationships", and that "demonstrating how you have addressed these complexities is an important element of accountability".

In addition to implementation concerns, the ethical implications of the impact of the proposed system must be part of the requirements analysis. These impacts are very real and very far reaching. A number of monographs (Kearns and Roth 2019) and a wider literature of conferences (Association of Computing Machinary (ACM) 2021) and journals (Springer Verlag 2020) address the concerns that have emerged about the impact of AI, ML and algorithmic decision making. These impacts are especially significant to marginalised and disempowered groups within our society.m Additionally, efforts are underway to capture and manage databases of so called "AI Incidents" (AI Incident database 2020). At the time of writing 1225 incidents are recorded in the database. Some examples include the impact of fine-grained work scheduling systems that take no account of common-sense or the needs of employees outside the workplace. Other examples are content moderation and content generation problems on social media, for example by AI bots, and multiple accounts of fatal accidents involving industrial robots.

Although the discussions in the specialist literature are wide ranging and informative, they are necessarily driven from an academic and philosophical standpoint. This means that there is an additional challenge facing teams working in a commercial organisation that are required to both create systems that deliver business value and have ethical integrity.

The business cases that support the development of computer systems have historically often involved ethically questionable trade-offs. In the first wave of office automation many desk-bound jobs were lost because it was possible to implement and cost effectively install computer systems that did the work of hundreds or thousands of claims processors, invoice adjusters or expenses managers. These systems made fewer mistakes and could be reprogrammed to reflect changes in business policy faster than the previous workforce could be retrained (it is sometimes claimed – although the long history of failed system upgrades in this kind of context would make some people a bit doubtful of this). Were these systems unethical? From the perspective of millions of people who lost their jobs they must have appeared so – and indeed there were strikes and protests about these projects (Sale 1997).

However, the consensus was that technological innovation was inevitable, and that organisations or sectors of the economy that failed to innovate and implement these systems would be obsolete and destroyed by competition. Additionally, at the time that these innovations were being developed there were compelling needs to grow the global economy. After all, this was the era of widespread childhood hunger and famine. With hindsight it does seem true that the change in the economy did deliver improvements in the conditions of billions of people, but the growth in inequality in all societies points to the partial application of new technology as a tool that favoured the ruling classes and socially dominant groups in society. The experience of the late 20$^{th}$ century and the third industrial revolution that transformed the economy has changed the perception that innovation is anything but a choice, and sometimes for some people it's a bad one.

These lessons and negative impact of social networking applications on society offer a reminder to ML practioners that business cases that look positive overall must be carefully weighed and balanced with the perspectives of all those impacted. A proposed project, now that the overall hypothesis, user stories and system outline are known, must be reviewed from

the perspective of those impacted, and where possible with their direct input. The personal data impact assessment must be performed, and the system reviewed with respect to the accountability and governance requirements of this application.

The state of the art with respect to the ethical considerations relevant to AI systems is emerging, and at the time of writing subject to considerable debate, for example see (Wired 2021) , (Bender, et al. 2021). Personal and subjective biases are very hard to avoid when evaluating a system's impacts and implications. While every engineer has a duty to attempt to avoid harming others via the solutions that they build, there is a distribution of both experience and capability between engineers. It may be that you and your team are empathic, insightful and have the range of perspectives required to understand the long-term consequences of the solutions that you advance. On the other hand, may also be that you have the same tendency to blind-spots and biases as the average gang of humans and so you don't! Given human fallibility it's sensible to take advantage of structured tools that are intended to introduce process into the evaluation.

One early example is the Algorithmic Impact Assessment tool developed by the government of Canada (Government of Canada 2020). This provides a questionnaire that determines the likely harms and extent of harms for projects that are introducing algorithmic reasoning into business domains. The tool generic is limited in the domain of application, lacking specialist questions with respect to medicine, construction, and manufacturing for example, but provides an indication of the shape and use of tools for this purpose in the future.

Another example is the guidance from The Ada Lovelace Institute on the application of algorithmic impact assessment tools (Ada Lovelace Insitute 2020), which is helpful in suggesting how these tools can be used effectively to support the choices that AI and ML practitioners must make when developing systems. Some models are available that allow a pragmatic approach to delivering a safe & ethical AI system. For example, work on safety in machine learning systems (Hendrycks, et al. 2021) advocates a layered model of safeguards. Figure 2.3 shows the concept of building a set of layers of checks and guards that catch more and more of the mistakes and wrongs that a system might cause. The layers in Figure 2.3 are:

- External safety, or deployment hazards: using a systematic approach to development means that you can pin down causes of failure. Gradually problems can be identified and sorted out by you and your team. The risk register is one mechanism for doing this, running explicit evaluations is another, reviewing with users is one more. Eventually though the system will be released into the wild, so it's important to specify a post development system where it should be safe to run it as well.
- Monitoring: so that they system can be inspected and its behaviour is known to the users and it's owners. The behaviour of the system should be surfaced and recorded and there should be a system of alerting and notification so that issues are brought to the users' and stakeholders' attention.
- Robustness: how it performs should be characterized and tested. The way that they system will act in circumstances that it might be used in should be understood and part of the process of acceptance to service.
- Alignment: can be meaningfully steered and controlled by appropriate humans. Consideration should be given to inclusivity when considering who is in control of the system and the mechanisms for implementing and reporting on that.

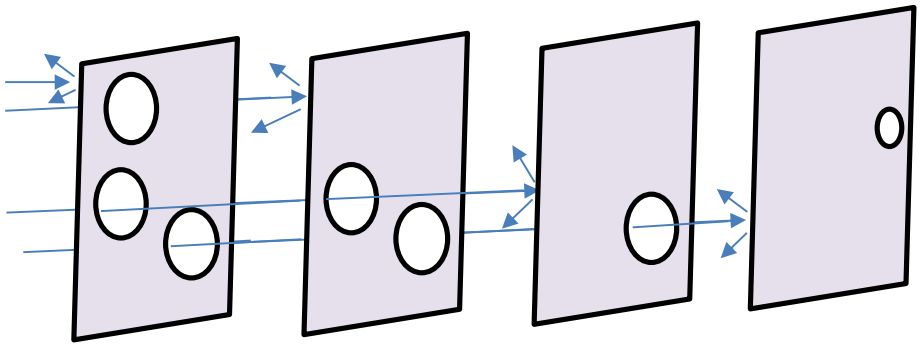

Figure 2.3 Layered model of ML Safety adapted from Hendrycks et-al 2021.

In summary:

- Review the project hypothesis, user stories and system outline to determine the list of stakeholders impacted by the implemented system and by the use of data derived from them or their communities for training the models in the system.
- Review the system from the perspectives of these stakeholders, where possible with their direct input, to determine the impact of the system on them.
- Undertake a systematic assessment of the proposed system using an Algorithmic Impact Assessment tool.
- Communicate the outcomes of the assessment to project stakeholders.

## 2.7 Development Architecture and Process

As well as the system produced for the users, the team needs to produce or onboard onto systems that will allow for the creation and delivery of the models. PS9 captures the requirement to understand what work will be required in getting this done.

**Development Architecture Tickets**

PS9

- Develop a high-level delivery architecture.
- The architecture should cover dev, test and production components (sometimes also pre-production/staging) and be able to support the customers non-functional requirements such as availability, resilience, security and throughput.
- Try to qualify this architecture with the appropriate stakeholders for feedback.
- Document key aspects of the architecture as assumptions for the project

In operational environments there are typically three or four layers of environment that need to be set up and then configured and used to get something in front of a real user. These layers are the development environment where the team will work, the test environment where the system is checked for effectiveness and quality and the production environment where it will actually run. In some cases, there will also be a pre-production environment which is provided due to regulatory or data protection concerns to further screen a tested system for its behaviour in the face of sensitive data. These layers are colloquially called dev, test, prod and pre-prod (or QA) by the folks using them.

Figure 2.4 illustrates how these are arranged and the flows between them and the version control systems that they team use to manage their code and other artefacts.

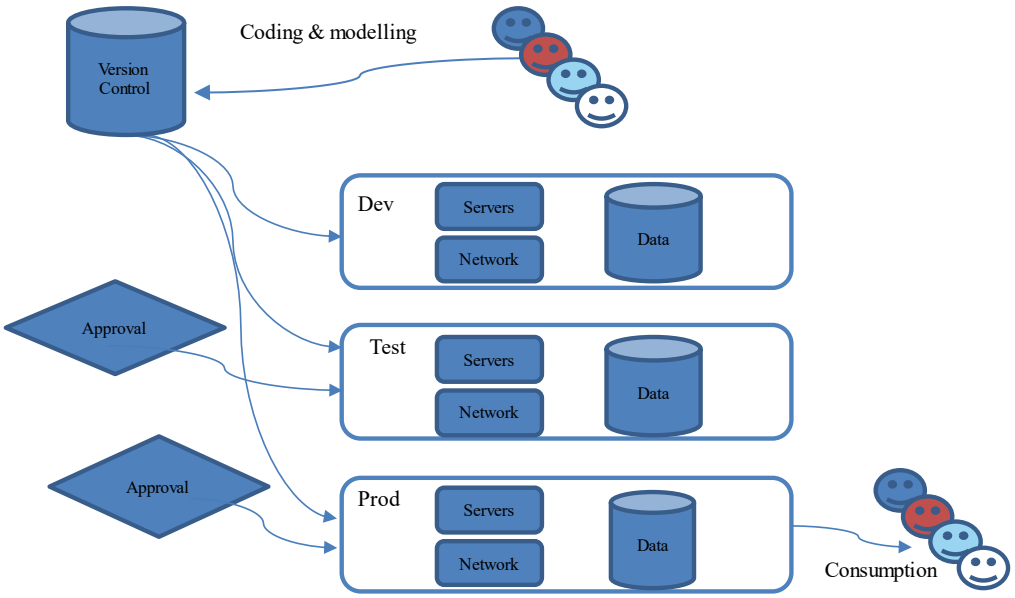


**Figure 2.4 Environments for delivery; sometimes a Pre-prod or staging layer that completely replicates production is also required**

Figure 2.4 shows these three environments.

- Development: the environment where your team will work and create the solution. The "Dev" environment may include specialist tools such as compilers (!) and GPU's/TPU's for model training, or alternatively large parallel compute systems/many core machine for model search and evaluation. Read below for a short discussion about why it might be really important to push for these machines in Dev. Many Dev environments do not contain live/sensitive data; the ML Dev environment will often need to contain live/sensitive data – this needs to be clear and manged effectively.
- Test: where model evaluation will be conducted, and components required for production such as the prediction service will be tested. This environment will typically replicate the production system with high fidelity – apart from the system databases which will usually contain snapshots of data or mocked up data that preserve confidentiality and yet allow for testing.
- Production: the environment that will be used to deliver results to customers. The production environment should be now somewhat understood from the interaction with the organisation during requirements analysis.

To figure out the system aspects that are required for the team to deliver models you need to understand Dev and Test. Also, it's important to understand the flows between these and into the production environment. As well as understanding what the standard processes are in the client organisation, you'll need to drill into the issues that are particular to ML systems.

### 2.7.1 Development Environment

The "Dev" layer of the architecture is where your team will work, and it will provide the support that they will need to deliver quickly and effectively for the client. Because of this you need to establish what's available for them to use.

There is reference material available that outlines some of the mechanisms that an "MLOps" team will use to deliver projects (for example Treviel 2021). The MLOps environment is the set of components that the team will need to run if they are to rapidly iterate and release new models and solutions. These tools will also allow them to control and govern the evolution of the models and work in a systematic way as a team. Either the team will use a customer's MLOps setup or a new one will have to be constructed. If there is an MLOps infrastructure in place, then obviously it's important to verify that it's fit for purpose and to figure out what can be done (if anything) to bring it up to the standard the team need.

Questions to ask and answer when there is no MLOps at the customer:

- Can required source code control systems accommodate the artefacts that will be produced in the project? If not are exceptions possible and agreed?
- Is the data available in Dev?
- Are there suitable servers for modelling work in Dev (i.e., GPU's, multicore, large memory)?
- How does the dev system get into test (with particular attention to the paths from non-standard environments)?
- What testing is required to move from test to Prod?
- What are the timelines for ordering or getting access to the infrastructure for all three

delivery layers?

- Who approves the orders and the spend on the infrastructure?
- Do any of the data systems require special access arrangements? Sometimes databases can only be accessed on the customers premises, or from a safe-listed laptop that's been specially secured to prevent screenshots being shared or other data exporting tricks and workarounds being used.

Issues that need to be addressed as requirements:

- Is there somewhere to host a model repository & feature store?
- Where can a tool to move files and artefacts between environments be hosted? For example, where can a Jenkins server be run?
- Where can data pipeline tools be run – for example where could an Airflow server be hosted to run update and reformatting tasks?
- What effort is required to get these systems in place and who will undertake it?

If there is an MLOps system in place obtain a technical description and get it validated, if possible, by a hands on data scientist, ideally someone who might be a member of the modelling team for the project.

### 2.7.2 Production Architecture

What's available in the development architecture will inform and enable how you go about building models and developing them into a system to solve your clients' problems. The production architecture – the IT kit that's in day to day use in the client – dictates the structure of the systems that you are going to build

Later in the project, when the properties and detailed requirements for the models have been determined, a lot of detailed work will be required to create a detail system architecture that can be implemented. At this stage, what's required is that a high-level solution that could potentially be delivered is developed. To break that requirement up:

- The solution should be defined at a high-level, meaning that the components that will be responsible for the different functions in the system are identified. The way that those components will be used and how they will interact is not defined in detail at this point.
- The solution can be potentially delivered, meaning that the components are available in the client's architecture, and the team and the client know how they can be commissioned and used.

The point of creating this design is that it demonstrates that there is a reasonable way to get the system delivered. If there is a problem in creating this kind of high-level design because some of the components that are required are missing or the team doesn't have experience of using them, then this task has done its job. The fact that there is a gap that must be filled needs to be exposed now, because it will be too late to fix it later in the project.

Returning to the smart building example what are the system components that are required to provide a solution.

The data from the sensors in the buildings needs to flow into a database, an execution environment is required to run a model that's been created to determine the control signals

for the building. The signals need to invoke actions from the building's actuators and information about the events and decisions in the system's life need to be presented to its users and owners so that they understand what's happening. So, the requirements (at a high level) are for:

- A messaging system that can manage the flow of information from the sensors and to the actuators.
- A data base that will hold the history of the sensor information and actuator instructions.
- An execution environment.
- A dashboarding system.
- An authentication system (to manage user accounts).

The system architects for the building owners will know what's currently in use; for example, they may have a pre-existing large database and an authentication system that's used to manage all employees and the access and entry to the buildings. In this example, the resulting architecture would be:

- mySQL for the database
- Tableau for dashboarding
- Active directory for authentication

But the messaging system would be new in the architecture. The question that needs to be asked at this point is: would the introduction of a messaging system like Apache Kafka be acceptable? What will need to be done to get Apache Kafka actually accepted and deployed to production? This work will have to be done by someone, but who is expected to do it, and when will it get done?

## 2.8 Summary & Takeaways

- A structured process to develop a project is necessary if you are going to have any chance of managing the risk of it.
- It's really important to understand how the project will be managed and the project management infrastructure required.
- ML projects have particular features that need to be captured as requirements
- Particular attention needs to be paid to the data assets that are going to underpin the project, as well as getting a picture of what data is notionally available it's important to understand how the data will be accessed and what capability there will be to manipulate and prepare it for use by Machine Learning.
- Specific requirements about the security and privacy of the data asset need to be understood; these can introduce large costs into the project.
- A well understood and fit for purpose development infrastructure will be needed and the IT architecture that the project is going to be delivered into needs to be understood clearly as well.
- Specific consideration to the corporate responsibility and ethical aspects of the project should be built in from the beginning.

## 2.9 References

Ada Lovelace Insitute . "Examining the Black Box." *Ada Lovelace Insitute .* April 2020. (accessed October 2021).

*AI Incident database.* 2020. https://incidentdatabase.ai/discover/index.html?s= (accessed January 27, 2021).

Association of Computing Machinary (ACM). *AAAI/ACM conference on Artificial Intelligence, Ethics and Society.* May 2021. https://www.aies-conference.com/2021/ (accessed January (conference publicity) 28, 2021).

Bender, Emily, Timnit Gebru, Angelina McMillan-Major, and Mitchell Margaret . "On the Dangers of Stochastic Parrots: Can Language Models Be Too Big? ." *FAccT'21, ACM Conference on Fairness, Accountability and Transparency.* Virtual Event, Canada: ACM Conferences , 2021. 610-623. https://dl.acm.org/doi/pdf/10.1145/3442188.3445922.

Government of Canada . *Algorithmic Impact Assessment Tool.* 2020. https://www.canada.ca/en/government/system/digital-government/digital-government-innovations/responsible-use-ai/algorithmic-impact-assessment.html.

Guus Schreiber, Hans Akkermans, Ango Anjewierden, Robert de Hoog, Nigel Shadbolt, Walter Van de Velde and Bob Wielinga. "Knowledge Engineering and Management. The CommonKADS Methodology." Cambridge, Massachusetts: MIT Press (A Bradford Book) , 2000.

Hendrycks, Dan, Nicholas Carlini, John Schulman, and Jacob Steinhardt. *Unsolved Problems in ML Safety.* September 2021. https://arxiv.org/pdf/2109.13916.pdf.

ICO. *ICO Guide to Data Protection.* 21 December 2021. https://ico.org.uk/for-organisations/guide-to-data-protection/guide-to-the-general-data-protection-regulation-gdpr/data-protection-impact-assessments-dpias/when-do-we-need-to-do-a-dpia/#when2 (accessed January 12, 2022).

ICO UK . *Guidance on The AI Auditing Framework. Draft guidance for consultation.* https://ico.org.uk/media/about-the-ico/consultations/2617219/guidance-on-the-ai-auditing-framework-draft-for-consultation.pdf: Information Commissioners Office, UK , 2020.

Kearns, M., and A. Roth. *The ethical algorithm: The science of socially aware algorithm design.* Oxford: Oxford University Press, 2019.

Kreutzer RT, Sirrenberg M. *Understanding Artificial Intelligence .* Cham: Springer, 2020.

Sale, L. "America's New Luddites." *Web Archive .* February 1997. https://web.archive.org/web/20020630215254/http://mondediplo.com/1997/02/20luddites (accessed January 28, 2021).

Springer Verlag. *Journal of AI Ethics .* December 2020. https://www.springer.com/journal/43681 (accessed January 28, 2021).

Taylor and Francis Online. *Journal of Applied Artificial Intelligence .* 2020. https://www.tandfonline.com/toc/uaai20/current (accessed August 26, 2020).

Verheyen, Gunther. "Fixed price bids." 2012. https://guntherverheyen.com/2012/10/07/fixed-price-bids-an-open-invitation-to-bribe-cajole-lie-and-cheat/ (accessed 08 04, 2020).

Wired. "What Really Happened When Google Ousted Timnit Gebru ." *Wired.* June 2021. https://www.wired.com/story/google-timnit-gebru-ai-what-really-happened/ (accessed July 2021)