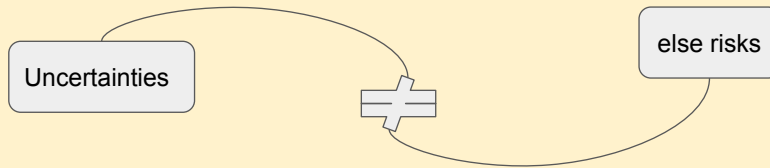


Risk Management, task 1

Keyword: **Uncertainties**

Risk management has to do with having a plan also for those uncertainties that potentially could affect the project, forecasting them at every work item and having tools for dealing with them.

Ruling out definitions

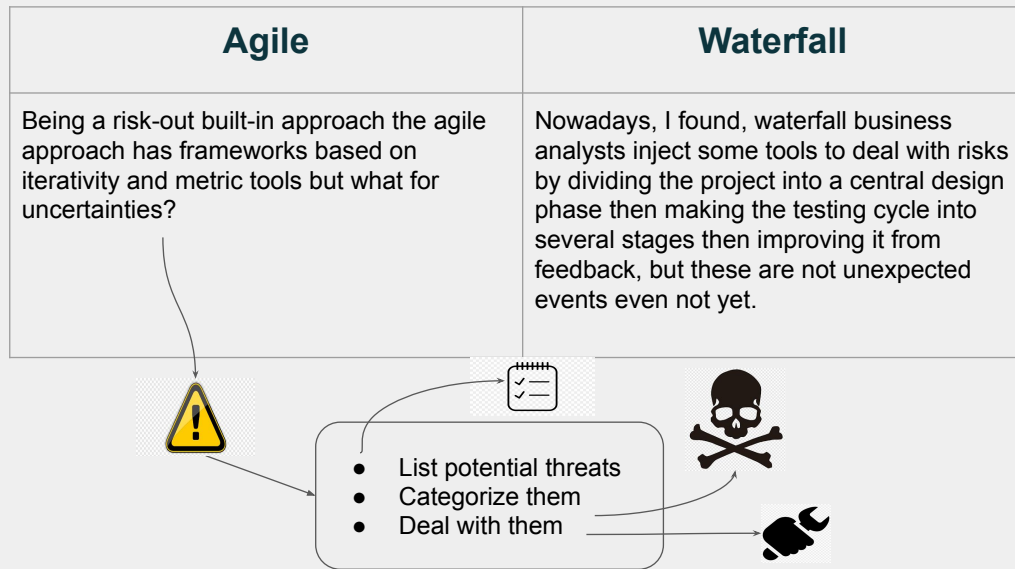


Every project has potential difficulties, or its accomplishment involves puzzling collaterals problems, the difference is that all that stuff is processed and already integrated into the timeline by the designing stage of the project.

Or as in the agile approach, by dividing the project by deliveries, and making deliveries on time sets as in the Scrum application (sprint) or using metric tools (also Scrum) so prolifically as in the Kanban methodology, or making incrementals as continuously (Kanban) as required.

Risk expected are of course the subject of the method but uncertainties, unexpected issues are a different thing.

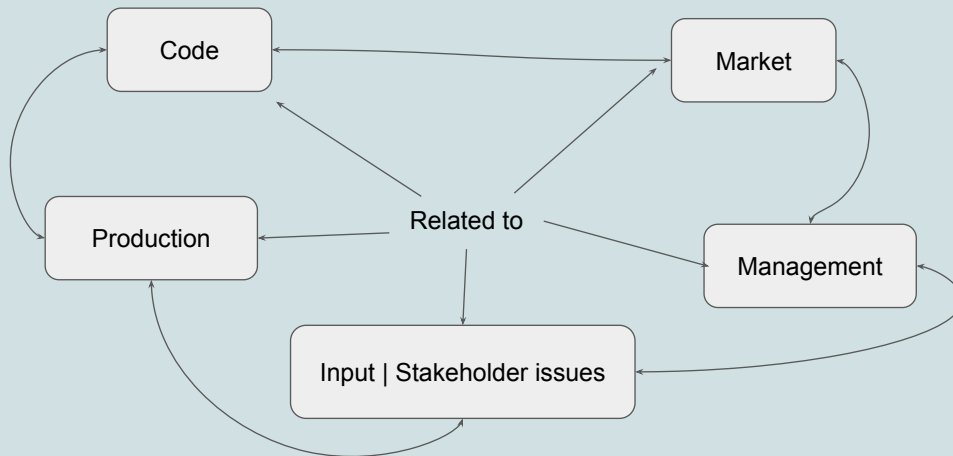
Dealing with uncertainties, have a plan b in case of plan b fails



Let's summarize this concept following the plot sketch of a movie that we all have been watching throughout the history of the software industry. The exhibition or beginning will be that already iconic proposal by Royce in the 70s, outlining a certain sequence of development phases, later called the Waterfall methodology, but also pointing out some inconveniences or risks of failure (to connect with this review on the concept of risk management) and sketching out some corrections. The rising action and the climax in the plot are given by those almost 30 years in which the mentioned method becomes the default method, everyone applies it, and the industry drags its inconsistencies at the same time as the software industry itself evolves and becomes mainstream. The denouement in the plot is given by the Agile movement that arises to face all the supposed risks observed throughout this period, but intentionally or not deals with them by elaborating from the drawbacks already mentioned by Royce himself, so that the Agile approach itself could be a risk management plan respect to the Waterfall method. So, what will be the concept of risk management for a method built-in risk-free? Here is what I have found, being all the so-called

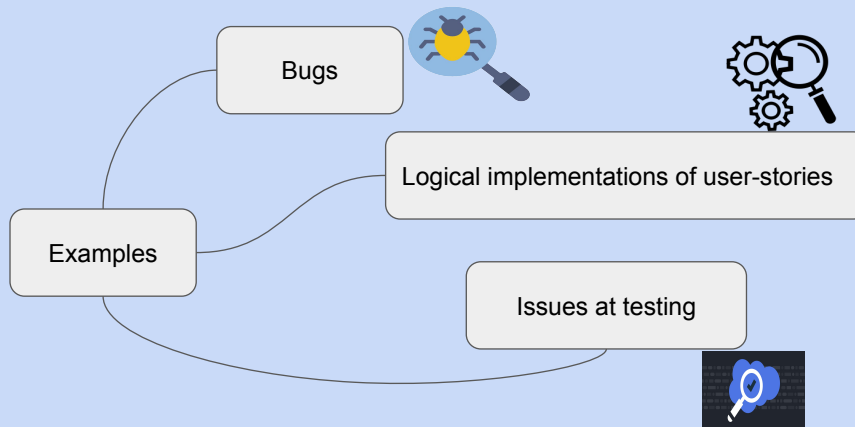
(expected) failure eventualities part of the main plan of Agile, there would be still unpredictable issues or for all those uncertain eventualities, I have found, the Agile system deals with them as another project on its own, and proceeds according to the following plan, namely, listing out the risk events, categorising them out and making a rescue plan one by one mitigating its effect in the target of the sprint in action or the delivery timeline.

Potential Risk that could affect project delivery, task 3.2



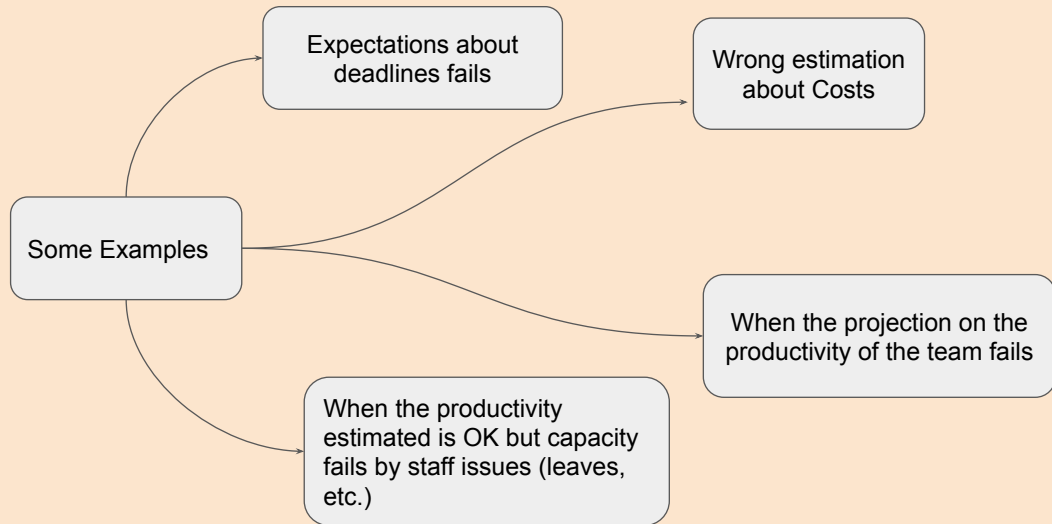
Here shall we scope the idea of potential risk on those eventualities problems when dealing with a project. As we have a plan, the plan is to have a plan for every aspect of the enterprise. I have organized the closest to possible ones into five groups, the first one related to the code construction itself, a second related to the production whose scope shall be as namely, expectations about deadlines, issues at estimating costs, also on the productivity and or when the projection of the capacity of the team fails, a next one related to the input information, such as definition issues or scope, also related to the stakeholder expectance, a four one related to markets such as the user acceptance and a last one which we shall call it recursively management issues and it has to do with a lack of a proper risk plan usually under the myth of being risk-free just because we do work under a risk-proof method, and or when issues has to do with the Management Plan itself.

Potential risk related to Code



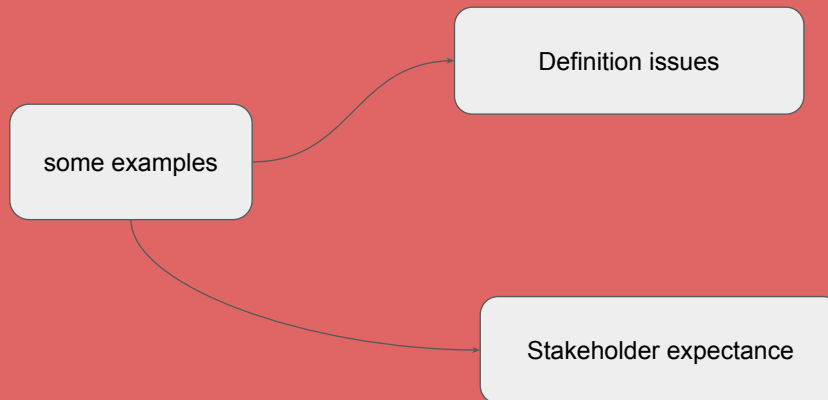
The list of issues related to this chapter could have no end but let's get started with the top three potentially probable, bugs in the code, issues related to the logical implementations of the user-story or when everything seems ready, issues related to deploying, and testing.

Potential risks related to Production



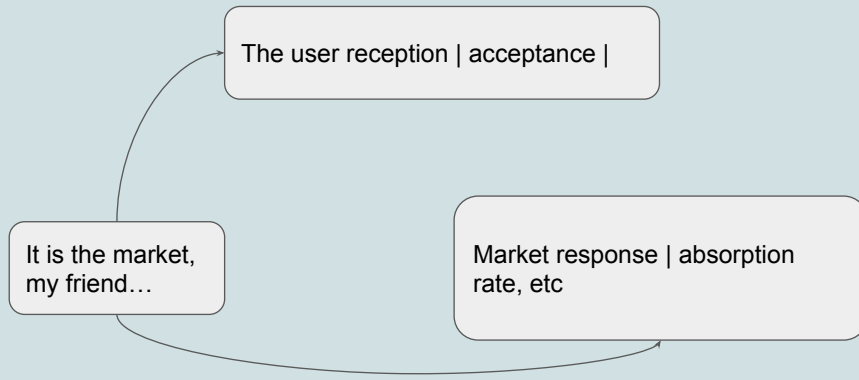
It is not difficult to get into issues at projecting deadlines, specially in order to hit competitors or be aggressive enough, similar at estimating costs. At the internal side this type of risk would be related to the productivity projected, or how much increment the team is ready to make, and or when the projection of the capacity of the team fails, due to unexpected staff issues.

Input info | Stakeholder issues as potential risk



As said, yet the input information is the first commitment in the Agile philosophy there would be still, potential expected issues at defining for instance the scope of increments, when definition is not, of deliveries at any point in the process. Or also related to the stakeholder expectance whose variability is out of the scope of the business.

Output | risks related to Market



However the market response is not easy to decode fully. We could identify at least two type of issues in this ambit, the target user individually, against whom we still could iterar to gain feedback, and the market absorption rate, which being out of scope of the project, strictly, or lesser concerned than what the stakeholders shall be, is a mysterious land.

Potential risks related to Management

The self-evaluation
group of potential
risks

Uncertainties are a real problem |
Underestimating to have a proper
risk management plan

When the application of the
management plan itself fails.
Poor tools, or other
inadequacies

When the inadequacy is given in the implementation of the management plan, usually related with tools usage or lack of tools, implementation, monitoring the planning etc. Also when the unexpected happens, and we found out we do not have a plan b for a plan b failure.