

Causes	Definition	Examples of citations
Deadline	A certain period of time defined by team, project manager and/or customer to deliver a determined activity, feature or product	- "The rush of managers (customers) that want to receive something working as soon as possible"; - "Tight deadline".
Adoption of contour solutions as definitive	Refers to a shortcut or temporary solution that the team takes to solve a problem and/or finish some activity	- "Adoption of contour solutions as definitive".
Bad code	Refers to code that is written anyway, without considering its quality	- "Poorly developed code" - "Low quality code"
Change in project	Refers to changes in the project during the development process	- "The next sprints increased, and it was necessary to include more people and increase the project schedule"; - "Constant request for changes in the project".
Change of requirement	Refers to changes in the project requirements during the development process	- "Change of requirement in a short time"; - "Constant change of requirement".
Change of scope	Occurs when there is a change in the customer's expectations during the development process	- "Extrapolation of the initial context of the project"; - "Client changing deadlines (always anticipating) or project scope".
Choice of inadequate technology / tool / platform	Refers to the adoption of tool, technology or platform not/less suitable for the project needs	- "Use of tools that complicate testing..."; - "Lack of tool that collaborated versioned code".
Complexity of the project	Occurs when the software has high complexity, making its development and maintenance difficult	- "Underestimating the complexity of what has been developed"; - "The application was quite complex, so I had to devote more time to the implementation of specific functions".
Concern just with back-end development		- "The concern only with the back-end development".
Cost	Refers to the cost of performing an activity	- "...changing this stage ends up being costly"; - "High cost in the short term"; - "Cost of modularizing x delivery".
Customer does not know his own needs	Occurs when customers do not know their own needs, making it difficult to collect requirements and the project definition	- "Customer ignorance of their own needs"; - "Non-definition of the project by the customer".
Customer does not listen the project team	Occurs when the customer is stubborn and insists on something that the project team does not recommend	- "Customer imperative".
Developers do not like to do some activities	Refers to activities in the development process that some people do not like to perform	- "The fact of being an activity that developers do not like to do"; - "Resistance on the part of the developer in documenting".
Discontinued component	Occurs when the component is no longer updated by your supplier	- "Discontinued component, or that did not receive update for more than three years".
Documentation that does not exist	Occurs when there is no documentation for the project	- "Lack of documentation"; - "Nonexistent documentation".
External component dependency	Occurs when the project is dependent of an external component	- "The fact that Angular 2 functionality is not yet stable, even with a deadline to fix the bugs".
Focus on producing more at the expense of quality	Occurs when there is a greater concern in delivering products faster, regardless of	- "Project manager committed to the delivery of requirements and not to

	their quality	quality"; - "Focus on the amount of functionality delivered and not on the quality of what was being delivered".
High turnover of the team	Refers to modifications in software teams	- "Turnover of teams"; - "Constant reallocation of analysts by the IT manager".
Inaccurate or complex requirement	Refers to failure, lack of clarification, complexity or poor definition in the requirements collected	- "Lack of detail in requirements"; - "Lack of clarification of requirements"; - "Poorly defined requirements".
Inaccurate time estimate	Occurs when the time to perform an activity is improperly estimated	- "Badly stipulated deadlines"; - "Lack of adequate estimation".
Inadequate data model	Data model that is not the most suitable for the project	- "Non-concise data modeling".
Inadequate impact and risk analysis	Occurs when the project team does not have a future view of the risks and their impacts on the project	- "Lack of correct identification of risks"; - "Lack of knowledge on the risks and their impacts on the project".
Inadequate management decision	Refers to decisions taken by project managers that are not very appropriate for a particular situation	- "The decision of the top management of the company".
Inappropriate / poorly planned / poorly executed test	Refers to project that is poorly tested, or even when the tests were poorly planned or do not have good coverage	- "Lack of testing"; - "Failed tests".
Inappropriate planning	Refers to problems in project planning	- "Lack of prioritization of activities"; - "Deficiency in project planning (disorganization)".
Inappropriate use of tools	Occur when a tool is used improperly for a particular purpose	- "Inappropriate use of tools".
Issues in requirements elicitation	Means problems regarding the requirements elicitation and their validation	- "Poor-elicited requirements".
Lack of a well-defined process	Refers to the lack of a sustainable methodology aimed at creating and maintenance of guides that would increase the productivity and development of the software team	- "Lack of processes"; - "Lack of methodology, simply the bosses come with the project and we start to do it without even understanding what to do".
Lack of automated testing	Project that has not been tested in an automated way	- "Lack of automated testing".
Lack of awareness of the importance of testing and refactoring	Occurs when the team do not recognize the importance of documenting and testing the software	- "To think that certain tasks (tests, refactoring) are not important"; - "Little importance given to white box and black box tests".
Lack of change control	Lack of control of changes made during the development process	- "Uncontrolled Changes"
Lack of code review	Refers to the source code of the project that does not go through the review stage to ensure quality	- "Lack of peer review"; - "Lack of code review...".
Lack of commitment	Nonprofessional commitment of stakeholders to fulfill the tasks assigned to them	- "Stakeholders not engaged"; - "Little commitment of the development team".
Lack of domain knowledge	Refers to the lack of knowledge about functional, technical and operational specifications of the project	- "Improve the understanding of business rules".
Lack of experience	Refers to the lack of experience, obtained through the practice in certain software development activities	- "Lack of experience of programmers"; - "Little experience of the people involved in the team".
Lack of information	Refers to the lack of information about something related to the project	- "Misinformation".
Lack of interest in acquiring knowledge	Refers to the lack of interest of project team members in seeking knowledge to develop	- "Lack of interest and willingness of the team to acquire knowledge".

	new skills	
Lack of IT Governance	Refers to the lack of development of a structured set of skills, responsible for planning, implementing, controlling and monitoring projects	- "Lack of governance, of understanding that IT is part of the business as a whole".
Lack of technical knowledge	Refers to the unfamiliarity with any activity or artifact of the project	- "Lack of knowledge of the team in tests" - "The person who manages the development has no technical knowledge about software development"
Lack of knowledge of technology	Refers to the lack of knowledge about the usage, the operation or the purpose of a particular technology	- "Lack of proper training in object orientation" - "Lack of knowledge of ReactJS"
Lack of motivation	Refers to the demotivation of team members in continuing to work on a particular project or company	- "Lack of motivation of human resources"
Lack of pair programming	Refers to lack of the adoption of the technique whereas a couple of programmers work together in one workstation, so that one revises the work of the other in order to guarantee quality in the codification	- "Lack of pair programming"
Lack of qualified professional	Occurs when unprepared professionals performing a certain activity or lack of professionals prepared to carry it out	- "Absence of specialist to carry out specific activities" - "Professionals unable to work"
Lack of external quality	Refers to when the project does not meet quality standards	- "Lack of quality in deliveries" - "Lack of balance between the result and the total quality"
Lack of refactoring	Refers to the process does not consider the improvement of the internal structure of the code without changing its external behavior	- "Lack of code refactoring" - "There was no code refactoring at the beginning of the problem"
Lack of requirements analysis	Refers when the requirements of a customer are not detailed completely	- "Lack of requirements analysis"
Lack of reuse practices	Occurs when existing software component or software component knowledge is not used for the construction of a new software.	- "Need to create the culture of reusability"
Lack of specific team	Occurs when there is no specific team to perform any software process activity, such as test team, development team, maintenance team, documentation team, etc.	- "Lack of testing team and documentation systems"
Lack of team communication	Occurs when there is communication problem between team members	- "Lack of integration of business areas, analysts of systems, developers and the test area" - "Lack of integration of the areas involved in the project"
Lack of traceability of bugs	Occurs when bugs are not traceable and therefore, there is no control over them	- "The lack of control resolution of bugs. Because there was a lot of recurrence"
Lack of training	Refers to the lack of training in the organization, either to get better the level of knowledge in the development team, or even knowledge to adopt a particular technology	- "Lack of internal training for capacitation" - "There was no training for the technology that was new to the company"
Lack of transparency between customer and	Refers to lack of knowledge and confidence of customer about activities developed by	- "There was extreme mistrust between the customer and the development

development team	team.	team, so that the lack of transparency eventually masked the true faces of the problems that should be solved through software. After the break of this barrier, the domains appeared with extreme ease and the solutions began to be delivered to the customer”
Lack of understanding	Occurs when the practitioners and the customer do not comprehend the impact on the delivery of immature artifacts.	- “Lack of understanding of the customer that this problem should be better worked to be better resolved” - “Lack of understanding of the management in knowing that TD is future burden”
Lack of validation	Refers to the lack of acceptance of the project with the defined requirements	- “Lack of validation” of what was projected” with the stakeholder”
Low productivity	Refers to a decrease in productivity in performing activities throughout the development process	- “Lack of team productivity” - “Productivity issues for code delivery”
Manager's lack of awareness of customer needs	Occurs when managers cannot understand the true needs of the customer	- “Project managers unprepared for the actual needs of the customer (lack of empathy)”
No bug fix	Occurs when the team is aware of the presence of bugs but does not correct them	- “No bug fix”
Non-adoption of good practices	Refers to the non-use of good practices that would facilitate the accomplishment and maintenance of activities in the project	- “Employment of bad design practices” - “Lack of use of good software development practices”.
Non-compliance with non-functional requirements	Occurs when the project is not aligned with its pre-set of non-functional requirements	- “The UI model presented by the designer was: the whole form on a single screen. The performance question was not considered”
Non-compliance with policies established by management	Occurs when team members do not follow the policies set by project managers	- “Noncompliance with the policies established by the company's governance to meet the immediacy”
Non-sharing of knowledge	Occurs when any member of the team does not share knowledge, such as explaining how something works or how to do some activity	- “The lack of a team to exchange ideas”
Outdated / incomplete documentation	Occurs when software documentation is outdated, unfinished, or simply missing in the software project	- “Incomplete documentation” - “Few professionals who eventually leave the company, taking with them the existing documentation, it being up to the substitute professional to understand the software through the code”
Poor allocation of resources	Occurs when there is a lack of financial or human resources in the company	- “Resource and performance limitations not analyzed and not taken into account” - “Lack of financial conditions”
Poor choice of framework	Occurs when the used framework is not the most appropriate for the team activities	- “Choose the bad framework to use”
Poor design	Refers to poorly designed project, such as high coupling, high cohesion, and poorly made data structure	- “Poorly designed database structure” - “In addition to failing domain definitions, the architectural design of the components provoked a very high level of coupling, which contributed to confusion with the customer, since when a part of the code was corrected, everything else was broken and

		consequently not could deliver value to the customer"
Poor scope definition	Occurs when there are problems in defining the scope of the project	- "Scope definition issues" - "Lack of more detailed definition of project scope"
Poorly elaborated SLAs	Occurs when the SLA of the project is poorly elaborated	- "Poorly crafted SLAs (without knowledge about cause)"
Pressure	Occurs when there is high pressure on team members to meet deadlines and speed deliveries	- "Pressure for delivery time" - "Aggressive delivery pressure" - "Customer pressure to accelerate the project" - "Pressure to meet short deadlines"
Problems in architecture	Refers to problems in the definition of software components, their external properties, and their relationships with other software	- "Lack of well-defined architecture life cycle" - "Lack of definition of a software architecture"
Problems with test environment	Refers to problems in the infrastructure where the test will run, including hardware configurations, software, automation tools, staff involved, organizational aspects, supplies, network and documentation.	- "Problems with test environment"
Project management / monitoring not effective	Refers to inadequate management during project development	- "Not following planning" - "Lack of understanding of managers"
Required infrastructure unavailable	Means that the company does not have the necessary infrastructure to perform some activity	- "Lack of adequate infrastructure to implement good solutions" - "The company's infrastructure sector still does not work with cloud computing"
Structural change in the involved organizations	Means that the company has undergone structural changes, and this has somehow altered standards in the development process.	- "Macroeconomic environment, changes in the structure of the companies involved"
Team overload	Means that the development team has accumulated activities, either because of a lack of adequate management or because members have left the team	- "Reduced Project Team" - "The accumulation of activities did not allow a loophole for extensive refactoring, so this was being pushed to the end of the project" - "Accumulation of activities / team overload"
Test not performed	Means that the project has not been tested	- "Lack of testing" - "Non-testing" - "Lack of functional testing by the supplier"
The company does not give importance to documentation	Means that the company has no culture on documentation and thus does not recognize its importance	- "The manager's vision that documentation is non-productive time" - "The company does not give much importance to the documentation of the projects"
Third party team involved in the project	Means a dependency of the development team on people from other companies	- "Dependence on inadequate solutions provided by third parties" - "Third party team involved in the project"
Version Incompatibility	Means that the version used is not compatible with the version required to perform an activity	- "The system being deployed with a new feature uses an older version of .Net that is incompatible with a third-party library, so proper system operation is compromised because of this incompatibility"

Effects	Definition	Examples of Citations
Activities that no one wants to take responsibility for	Refers to activities in the development process that some people do not like to be responsible for.	- "Activities that no one wants to take responsibility for".
Adoption of contour solutions	Refers to a shortcut or temporary solution that the team takes to solve a problem and/or finish some activity	- "Leader instructs developer to give a 'knack'".
Bad code	Use of bad practices in coding activities (e.g., bad variables/methods names, over complex code)	- "Bad workarounds in coding".
Complex system	Occurs when the software has high complexity, making its development and maintenance difficult	- "High complexity"; - "High complexity in code".
Constant need for retest	Constant need for retest	- "With each increment of the software, the main functionalities were tested manually by the system user, which required a lot of time during the homologation stage".
Delivery delay	Non-fulfillment of the deadlines agreed with the customer	- "Six months delay in project delivery". - "Deadlines not met".
Design changes	Need for design changes in the course of the software development process	- "Subsequently there was a need to change the interfaces/rules several times in order to contemplate the reality of the customer...".
Design problems	Issues in design during the software development process	- "Poorly designed methods / classes".
Developer dependency	Dependency of the development team on a developer	- "Dependency of a specific developer for project progress".
Difficulty conducting tests	Difficulty in testing the software	- "Difficult to test the software".
Difficulty in implementing the system	Difficulty in implementing the software	- "Difficulty to deploy new versions of the system (which means longer time and greater error propensity)".
Difficulty in project development	Difficulty in project development	- "Greater difficulty for project development".
Doubt about team capability	Occurs when managers/customers doubt about the ability of team members	- "Ability of the team questioned".
Fall in productivity	Production capacity of the team begins to fall after the occurrence of TD	- "Productivity decrease"; - "Low productivity".
Financial loss	Occurs when a company has financial losses due to issues in the software development	- "Reduction of profits"; - "Decrease of profitability because of the extremely high costs to keep the product in the market".
High turnover of the team	High turnover of the team	- "Turnover of the team".
Impact on production environment	Occurs when the presence of TD affects the production environment	- "Slowness of the production environment after the deploy".
Impaired company image	Impaired company image	- "Image of the company in risk".
Inaccurate time estimate	Time to perform an activity is estimated improperly	- "Badly planned schedule"; - "Frustrated estimates".
Inadequate / non-existent / out-of-date documentation	Encompasses issues such as, incomplete, outdated, or nonexistent documentation	- "missing or inconsistent documentation that hinders the understanding of the system".
Inadequate requirements	Occurs when there are issues in the requirements collected	- "worst requirements".
Inappropriate planning	Refers to problems in project planning	- "Frequent replanning of the project".
Inappropriate tests	Project that was poorly tested, or even when the tests were poorly planned or did not have good coverage	- "Disabling some or all the system in a badly tested update"; - "Fail of load and stress tests".
Inappropriate usability	Inappropriate usability	- "Bad interface usability";

		- "Deficiency in the usability of the system".
Increase in the amount of maintenance activities	Appearance of several problems after the delivery of the product and thus, need to go through various maintenance activities	- "Constant maintenance" - "A lot of maintenance after delivery"
Increased effort	Refers to the increase of effort to perform activities due to the TD presence in the project	- "Considerable increase in effort" - "Greater effort and time for understanding, maintenance and evolution of the software"
Increased risks	Refers to the increase in internal or external risks due to the TD presence in the project	- "Increase of risks"
Infrastructure overload		- "Computing infrastructure overhead to meet software needs"
Lack of commitment of users	Occurs when users do not commit or collaborate with the team	- "Lack of commitment of users"
Lack of credibility of the product		- "Lack of product credibility"
Lack of domain knowledge	Refers to the lack of knowledge about functional, technical and operational specifications of the project	- "Lack of business knowledge" - "Lack of knowledge about the project as a whole"
Lack of technical knowledge	Means unfamiliarity related to any activity or artifact of the project	- "Delay in the implementation of an adjustment or improvements due to the difficulty of a new employee to understand the logic of the system"
Lack of planning	Means that the project has not been planned	- "Lack of planning"
Lack of security	Project has failed to protect information and data, so that unauthorized persons or systems cannot read or modify it and are not denied access to authorized persons or systems.	- "Lack of security"
Lack of standard	Means not adopting a standardization in artifacts such as documentation, coding, etc.	- "Lack of architectural standard" - "Lack of standard"
Lack of training	Means that there is a lack of training in the organization, either to level the level of knowledge in the development team, or even knowledge to adopt a particular technology	- "Lack of training" - "Need of training"
Lack of understanding		"Lack of understanding"
Legal issues due to non-compliance with contracts		- "...and other even more serious problems leading to contract cancellation (caused by excessive court disputes over noncompliance with quality)."
Loss of traceability		- "Loss of traceability"
Low code reuse	Occurs when existing software or software knowledge is not/low used for the development of new software.	- "Difficulty in reusing code" - "Low code reuse"
Low maintainability	Encompasses problems that occur during software maintenance activities, such as increased effort to fix bugs as well as limitation in system evolution	- "Extremely difficult maintenance and evolution"
Low performance	Refers to issues in reaching performance requirements of the software (due to the degraded internal quality of the software)	- "After application growth, performance has become unsustainable and refactoring was inevitable" - "Prejudiced performance"
Low external quality	Refers to any aspect that reduces the quality of an artifact (including errors and	- "Low quality of what was offered known and uncorrected defect".

	known defects that are not fixed)	
Low Reliability	Occurs when the project has failed in testing phase, considering the validation of inputs, outputs and operations in relation to application requirements.	- "Loss of confidence of the development team about delivered solutions"
Need for refactoring	Refers to the need of improving the internal structure of the code without changing its external behavior.	- "Need to refactor" - "Pending code refactoring"
Need for skilled professionals to solve problems	Refers to the need of prepared professionals to carry it out project activities.	- "Difficulty in hiring professionals" - "Need for skilled labor to solve problems"
Need of training	Occurs when there is a need of training in the organization, either to get better the level of knowledge in the development team, or even knowledge to adopt a particular technology.	- "Lack of team training"
Non-implementation of functionality	Refers to the waiver of implementing some functionality because it was rejected or would require several complex changes in the application.	- "Rejected features" - "The risk of non-implementation of functionality due to necessary to make the changes"
Non-scalable software	Refers to the lack of ability software increase its demand, because its architecture or code is hard to maintain or add new functionalities and several people work on it.	- "Bad software scalability"
Planning Changes	Occurs when there is change in the project planning due to identification of bugs in the application.	- "Changes in planning due to emergency corrections"
Poor allocation of resources	Refers to the lack of financial or human resources in the company	- "Poor allocation of resources" - "Lack of resources"
Poorly defined scope	Occurs when there were problems in determining the scope of the project	- "No scope definition" - "Poorly defined scope"
Pressure	Refers to large demand of activities to delivery in short deadlines.	- "Delivery pressure, but lack of time to meet with the project team." - "Excessive pressure on the team"
Problem of communication	Occurs when there are communication problems between the stakeholders	- "Failure to communicate with projects without prioritization" - "Improper communication"
Project does not fit customer's expectations	Even following the requirements, the customer does not accept the developed functionalities.	- "Expectation break"
Project not completed	Occurs when the project reaches a point where it is not worth being finalized, and then it is abandoned	- "Did not complete the project" - "Abandoned Project"
Requirements changes	Changes in project requirements during the development process	- "Correction of rules"
Return of previously corrected defects	Occurs when features fixed in previous versions have bugs in other versions	- "Occurrence of critical defects" - "Bugs generated by TDs and unplanned items that usually arise because of them"
Rework	Refers to redoing something that should have been done following quality standard	- "The increasing rework that will be needed when TD is resolved" - "Code that had to be rewritten"
Slow processes	Occurs when the used process has a lot of activities and artifacts that are not necessary	- "Slowness on processes and competition between resources used."
Stakeholder dissatisfaction	Occurs when stakeholders are dissatisfied with the progress of the project	- "Customer dissatisfaction with new functionality deadlines" - "Customer dissatisfaction with product"

		quality"
Stop development activities for debt repayment	Occurs when the project reaches a point where it is necessary to stop the activities so that the TD is paid and does not cause major impacts to the project	- "The main effect was to pause all activities that were being executed to focus on payment"
Stress with stakeholders	Refers to the presence of friction between team members due to various factors such as pressure, deadline, accumulation of activities, etc.	- "Attrition on customer relationship, development team" - "Project team stressed"
Team demotivation	Occurs when the team is discouraged from the daily routine due to various reasons	- "High degree of stress/demotivation of the team involved in the project" - "Disengagement of the technical team"
Team overload	Refers to the development team has accumulated activities, either because of a lack of adequate management or because members have left the team	- "Overload the development team so that it can solve the TDs and at the same time evolve the application"
Wrong decisions in the architecture	Occurs when software architects make a decision in the software design that is not very appropriate for the application	- "Incorrect decisions in the management and / or architecture"