

Preventing Technical Debt by Technical Debt Aware Project Management

Marion Wiese
Universität Hamburg
Hamburg, Germany
wiese@informatik.uni-hamburg.de

Matthias Riebisch
Universität Hamburg
Hamburg, Germany
riebisch@informatik.uni-hamburg.de

Julian Schwarze
Gruner + Jahr GmbH
Hamburg, Germany
schwarze.julian@guj.de

Abstract—Technical Debts (TD) are problems of the internal software quality. They are often contracted due to tight project deadlines, for example quick fixes and workarounds, and can make future changes more costly or impossible. TD prevention should be more important than TD repayment, because subsequent refactoring and reengineering is usually more expensive than building the right solution from the beginning. While there are numerous works on TD repayment, solutions for TD prevention are understudied. This paper evaluates a framework that focuses on both TD prevention and TD repayment. It was developed by and applied in an IT unit of a publishing house. The unique contribution of this framework is the integration of TD management into project management. The evaluation was carried out by a study based on ticket statistics and a structured survey with participants from the observed IT unit and a comparison unit. The evaluation shows that the adoption of this framework leads to a raised awareness for the contraction of TD. This results in benefits like more rational discussions and decisions, TD prevention and timelier repayment of TD tickets.

Index Terms—Technical Debt, Technical Debt Awareness, Technical Debt Aware Project Management, Technical Debt Repayment, Technical Debt Prevention

I. INTRODUCTION

In their definition of Technical Debts (TD), Avgeriou et al. [1] describe TD as “a collection of design or implementation constructs that are expedient in the short term, but set up a technical context that can make future changes more costly or impossible.” In a technical metaphor to financial debt a sub-optimal implementation or design is interpreted as debt. The resulting problems are interpreted as interest rates, the refactoring as repayment of the debt and the refactoring cost as principal.

Many papers focus on TD repayment while the topic of TD prevention is understudied ([2]). However, in a long-term view, it can be expected that it is cheaper to implement the optimal solution right from the beginning especially when considering possible interest payments ([3], [4]). Therefore, TD prevention is worth more consideration ([4]).

The originally named cause for TD contraction is the problem of tight project deadlines as described in the paper of W. Cunningham ([5]). Other causes have been identified by different research papers ([6], [2], [1], [7]), e.g. bad design decisions, unavailability of a key person, neglected technical improvements, lack of education or parallel development.

Nevertheless, tight timelines are the TD cause that was identified by many researches as the most pressing cause ([8], [4], [7], [6], [1]). However, to the best of the authors notice no studies provide a solution for the problem of tight timelines that is feasible in an industry environment.

A good example of the timeline problems can be taken by the German government’s reduction of value-added taxes from 07/01/2020 to 12/31/2020 as part of the management of the corona crisis. The corresponding law was only passed two days before it came into force. To reflect this change in the software systems, adjustments to these systems were necessary in many German IT units, which were therefore subject to a very tight schedule. Well-structured project planning or adoption of good development practices were mostly not possible.

In many cases, the constant time pressure leads to a phenomenon in developer behavior. In preemptive obedience the developers search for the fastest and easiest solution assuming a tight schedule, even when there is no time pressure. In these cases, it is not common practice to step back and think about or discuss different solution options. There may be frequent situations where the sub-optimal but faster to implement solution must be chosen due to project timelines. However, it should be a goal to make it a conscious decision of the whole team to choose the optimal or sub-optimal solution.

At this point, TD prevention has a good chance for success by introducing a special form of tickets. These tickets that we call TD tickets make the contraction of TD explicit. The awareness for the contraction of TD is raised among all team members including business analysts and managers. By this, a basis for TD prevention is created.

Furthermore, TD tickets integrate TD management into project management. Project managers become responsible for the repayment of the TD they contracted during their project and therefore get an extrinsic motivation to reduce the contraction of TD. By this, this paper presents a solution to the problem that is presented by Martini et al. ([7]) as “*Split of budget in Project budget and Maintenance budget boosts the accumulation of debt*”.

The evaluation in this paper shows that TD tickets prevent TD and achieve a timely repayment of TD items.

These TD tickets are part of a framework developed in industry and a new contribution to the State of the Art. Further parts of the framework are continuous TD repayment and

maintenance projects which are already described in some research papers, e.g. by Steve McConnell ([9]).

The contribution of this paper is therefore the presentation and evaluation of the TD tickets as a means of TD prevention. The evaluation consists of a survey and ticket statistics. It will provide evidence that TD prevention and timely TD repayment can be reached by adopting this framework. Other benefits like more rational discussions and decision-making are highly appreciated side-effects.

In the following section we will introduce the framework. Section III will present the research questions (RQs) regarding the effectiveness of the framework that will be answered by this paper. The methodology of the evaluation will be explained in Section IV. The results of the evaluation will be presented in Section V and comprise ticket statistics and the results of a survey carried out in the observed IT unit and a comparison unit. The results will be discussed in Section VI. In this section the RQs will be answered, threats to validity will be discussed and the paper will be embedded in related work. The paper ends with the conclusion and future work in Section VII.

II. FRAMEWORK PRESENTATION

A framework for managing TD has been developed and established in an IT unit in the beginning of 2018. It has been utilized ever since. The IT unit resides in a German publishing house with a size of more than 9000 employees worldwide.

In this IT unit, TD accumulation leads to increasing cycle times, unnecessary errors in the systems, and discontent of the developers. The biggest problems of this IT unit are tight timelines. There is a high market pressure in this industry sector and contracts or laws need to be fulfilled in time. The tight timelines often mean that “*well-structured project planning*” or “*adopting good development practices*” as proposed by [8] are not sufficient.

These problems are recognized by the management and sufficient priority to handle TD is given. The IT unit including the management is willing to change its processes. The main goal for this change is predefined by the manager of the IT unit: TD have to be prevented in any case where this is possible without impairing the deadlines. Whenever a set deadline leads to TD, this TD shall be repaid timely.

On this basis, a framework to help manage TD was created. Especially, two factors of this framework shall support TD prevention: the overall raised awareness for TD contraction and the integration of TD management in the project management process. The resulting framework consists of four management categories for TD management which are presented below.

A. Management Categories

The framework comprises four management categories for handling different kinds of TD related tasks. There is a fifth management category for handling all functional requirements which will not be presented in detail as it does not deal with TD.

The different tasks are recorded as tickets in a comprehensive project backlog and tagged and handled depending on their category. The framework includes guidelines for the recording and processing of these tickets depending on their management category. It was decided that the person that has the most interest in the completion of the tasks of one category should be the one that is responsible for this category. These may be architects or business analysts. Table I shows an overview of the management categories, their responsible persons and repayment terms.

1) *Maintenance*: In this framework, tickets of the maintenance category are technically driven tickets that do not have a direct impact on the user's perception of the system.¹

Recording: Every maintenance ticket must contain a one-sentence information that describes the impact of the ticket. The description must be written in a way that is understandable for business analysts and managers. *Processing*: Ten percent of the planned capacity of every sprint can be invested for maintenance tickets according to the architect's decision. *Goal*: The goal of the maintenance tickets is to repay the unintentionally contracted TD continually and to allocate time for other maintenance tasks, e.g. technical adaption.

Examples for this category are the upgrade of a third-party-library or the refactoring of bad structured code.

2) *Maintenance Project*: All tasks that belong to the maintenance category as stated above but that require more than five days development time are handled as a maintenance project. This is based on the practice that is also applied for functional requirements and business projects.

Recording: A maintenance project roadmap is produced and prioritized by the architects. *Processing*: The roadmap is part of the overall project roadmap as defined and prioritized by the unit managers. *Goal*: The goal of maintenance projects is to allocate enough time to conduct maintenance tasks with a long duration or maintenance tasks requiring more organization, e.g. when more than one team is involved.

An example for this category is the version upgrade of a central database that is used by more than one application. Another example would be the change of the underlying architecture, e.g. from a SOA to a microservice architecture.

3) *Technical Debt*: Tickets of this category, so-called TD tickets, describe tasks that are necessary for internal software quality during the implementation of a functional requirement, e.g. time for clean coding, good design or to comply with a specified architecture. These tasks are not mandatory to implement the required functionality and can therefore be implemented after a reached deadline if this is necessary for the project.

Often two or more solutions are discussed, where one follows standards and the specified architecture and the others more or less deviate from the optimal solution but may

¹We use the term “maintenance” as it is often used in agile organizations. However, implementation of new or changing functional requirements is called adaptive maintenance in research, standardization, and sometimes in classical organizations.

TABLE I
TICKET CATEGORIES, RESPONSIBLE PERSONS AND TIME

Management Category	Responsible	Recorded By	Payback Time	Contingent
Maintenance	architect	developer	continually / architect decision	maintenance (10%)
Maintenance Project	architect	architect	management decision	project
Technical Debt	business analyst	all	after deadline, part of project	project
Deconstruction	architect	all	as soon as possible	project / functional req.
Functional Requirement	business analyst	business analyst	business analyst decision	project / functional req.

therefore be faster to implement. This idea of TD follows its original introduction by Cunningham ([5]).

Recording: In such cases, it is consciously decided to record two tickets. One ticket, a functional requirement ticket, describes the sub-optimal solution to be implemented before the deadline. By this ticket, TD is contracted. The other ticket, the TD ticket, describes the optimal solution and the tasks to repay the contracted TD. These TD tickets are usually identified during estimation meetings when the details of a functional requirement are discussed in the team and the effort is estimated. **Processing:** The uniqueness of this framework is that the repayment of intentionally contracted TD is part of the project plan. Thus, no project can be finished before the TD tickets are processed. After reaching the planned milestone for the main deployment and keeping to the tight schedule, the project plan does not end. Instead, TD tickets are prioritized and repaid in a subsequent project phase. This can be seen in Figure 1. This transfers the responsibility for the TD repayment to project management and thereby integrates TD management into project management. **Goal:** Evaluating different solution options in advance raises the overall awareness to only take on TD intentionally and prevents unnecessary TD. Furthermore, by making the project manager responsible for the TD accumulated during the project, his willingness to contract TD decreases. In other words, a feedback loop is generated and the behavior of the team and the project manager related to TD contraction changes. Additionally, the project and unit managers get an overview of the accumulating technical debt of a given project while it is still running. Thereby, the managers are given an opportunity to intervene or adjust the project plan and the following projects early on. Finally, by this approach TD are repaid timely.

For the example from Section I this may mean that, to adhere to the schedule, hard coded value-added taxes are adjusted in the code and are not refactored to a flexible solution. The corresponding TD ticket to introduce a centralized tax variable, is recorded during the estimation meeting. This TD will then be repaid in an orderly manner after the law came into force and in the “TD repayment phase” of the “tax adjustment” project. On the other hand, as different options for these changes are discussed during the estimation meeting, this can also mean that due to the low effort the centralization may be the better option and TD are prevented.

4) **Deconstruction:** A ticket of the deconstruction category is a special kind of TD ticket that cannot be processed during the project. After the implementation of a new solution there

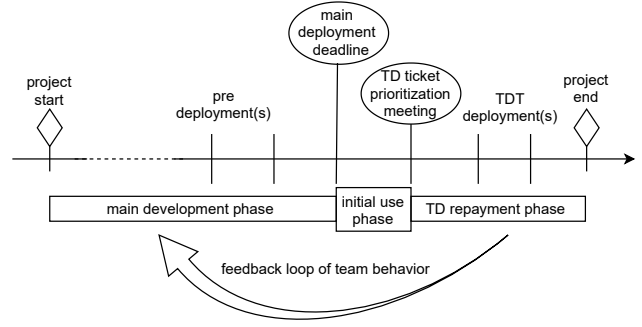


Fig. 1. Project plan with included TD repayment phase

is sometimes the need to keep legacy code in parallel for the time being. These tickets then comprise the deconstruction of this legacy code when it is no longer needed. They can also comprise the deletion of deprecated code parts after all consumers of this parts are detached.

Recording: These tickets have to contain an additional information describing the time when the deconstruction can take place. **Processing:** The deconstruction ticket has to be repaid as soon as possible as part of the contingent for functional requirements. **Goal:** The main goal of this tickets is to avoid cluttering of code and by this support code comprehension.

An example of this kind of tickets is the need for a new business rule for the next business year. The application should not be deployed on New Year's Eve to minimize deployment risks. Therefore, the application will be deployed in advance and must contain a date-driven switch to choose between the old and new business rule. When the old business rule is no longer needed it can be deconstructed.

III. RESEARCH QUESTIONS

As described in Section I, two main problems concerning TD are the questions on how to prevent TD and how to deal with TD despite having a tight schedule and fixed deadlines. For this question, the described framework provides a possible solution found and developed in practice. With subsequent research questions (RQ) this paper provides an evaluation of this framework. Especially its feasibility and its effectiveness regarding the aforementioned problems is evaluated.

(RQ1) Framework application

- (RQ1.1) Do practitioners find the framework reasonable?
- (RQ1.2) Are the processes of the framework feasible in practice?

This will be the basis for all other evaluations as it does not make sense to further analyze the impact of the framework should it not be feasible in practice. These questions will be evaluated by a survey of the members of the observed IT unit.

(RQ2) Framework effectiveness

- **(RQ2.1)** Does the framework lead to raised awareness for the contraction of TD?
- **(RQ2.2)** Are the TD items taken on more consciously when using the framework?
- **(RQ2.3)** Are the TD items paid back timely?

These aspects are the goals of the framework as presented in Section II. The evaluation of this question shall show whether these goals were reached. The evaluation is conducted by a survey and the tickets statistics (for the last part).

(RQ3) Framework benefits

- **(RQ3.1)** Can TD be prevented by the adoption of the framework?
- **(RQ3.2)** Are there other benefits arising from the adoption of the framework?
- **(RQ3.3)** Do these benefits justify the additional effort?

The purpose of these questions is to gather the impact of the framework. The main goal of TD prevention is evaluated, as well as secondary benefits concerning team discussion and decision-making process. The correlation between the survey variables will be used to answer this RQ. Finally, we ask for the justification of the additional effort in relation to the benefits to substantiate the feasibility of the framework.

IV. RESEARCH METHOD

The presentation and evaluation of the aforementioned framework is the contribution of this paper. The presentation took place in Section II. To answer RQ 1.2 and RQ 2.2, we evaluated the recording and processing statistics for TD tickets and maintenance tickets. All other RQs are answered by the results of a survey. The survey targets participants from the IT unit that uses the framework (observed IT unit) and a comparison unit that does not use the framework.

A. Ticket Statistics

To evaluate RQ 1.2 and RQ 2.2 the number of tickets are presented in terms of categories, priorities, and time as descriptive statistics. As the backlog includes some old and invalid tickets the extracted data had to be cleaned up manually first. This was carried out in consultation with the architects of the unit. The statistics for maintenance projects and deconstruction tickets are not evaluated because there is still insufficient data for this.

B. Survey

The method of a survey was chosen to benefit from the input of all participants that are willing to share their experience. For all RQ's corresponding questions were asked in this questionnaire.

TABLE II
COMPOSITION OF UNITS AND SURVEY PARTICIPANTS

	observed unit		comparison unit	
	members	particip.	members	particip.
manager	2	n/a	2	n/a
architect	2	n/a	1 ^a	n/a
business analyst	8	6	1 ^a	1
developer	15	9	5	4
operations	5	2	0	0
sum	32	17	8	5

^aarchitect and business analyst are the same person

1) *Survey Participants:* All members of the observed IT unit are asked to fill out the questionnaire. Additionally, a comparison unit is asked to complete the questionnaire. The comparison unit is led by the same unit manager which increases the comparability.

The results of the comparison unit gave us the opportunity to validate the descriptive statistics. The participants were asked to which team they belong, but due to works council regulations not which role (e.g. architect, manager) they inhabited. The composition of the observed unit and the comparison unit are presented in Table II. The response to the survey is also shown in this table. All questions and evaluations of the survey can be accessed in the additional material².

2) *Questionnaire Construction:* To avoid misunderstandings the questionnaire starts with a short information about the term TD. For statistical reasons, background information, e.g. the team membership of the participant, are queried. The questionnaire is further divided into two main parts: (I) assessment of the framework and (II) effects and benefits of the framework.

Part (I) shows the practical feasibility of the framework and provides the answers to RQ1.1 and RQ1.2. For every management category of the framework the same set questions regarding the reasonableness and feasibility are asked. Each part starts with a short information about the management category as it was also documented in the IT unit's wiki. Part (I) was not filled out by the comparison unit as these participants did not use the framework.

The second part comprises four subsections: TD Awareness (RQ2.1), Comparison of optimal and sub-optimal solutions (RQ 2.2), Observed benefits of comparison (RQ3.1 and RQ3.2), and Justification of the additional effort (RQ 3.3).

All parts comprised a set of assertions that the participants were asked to validate using the following Likert scale ([10]): applies - rather applies - rather does not apply - does not apply - cannot answer. The *cannot answer* option was given as some questions could not be answered by all members of the unit. The participants were explicitly asked to only use this option in these cases.

Finally, two more open questions in the end gave the participants an opportunity to point out especially good parts of the framework and parts that need improvement.

²<https://doi.org/10.5281/zenodo.4616485>

A first questionnaire was filled by two developers that knew about the framework as a pilot test. Based on this, the questions were reduced and focused. Furthermore, one manager gave feedback on the construction of the optimized questionnaire which was also incorporated.

3) *Survey Evaluation*: The questionnaire is evaluated using the statistics software SPSS³ for closed questions and the qualitative research software MaxQDA⁴ for open questions.

First, descriptive statistics for all closed questions are created for an exploratory data analysis. For simplification and analysis purposes the answers for the questions of part (II) are dichotomized. This means the answers *applies* and *rather applies* as well as *rather does not apply* and *does not apply* are summarized. All values are presented as percentage to align the output of the observed unit and the smaller comparison unit. For the sake of clarity, only the *applies* answers are shown in the following figures.

As a second step, hypothesis for RQ2 are developed on the basis of this exploration and the significances of the hypotheses are evaluated. The Mann-Whitney U-Test ([11]) is used because it can be applied for not normally distributed data on an ordinal scale. Only significant findings are presented.

Regarding the comparison unit, the benefits (RQ 3.2 and RQ3.3.) cannot be assessed but must be interpreted as expected benefits. As a result, no differences between the units can be evaluated regarding the benefits. Hence, it is interesting to analyze correlations between all survey participants who take on TD consciously and the benefits they assessed. These correlations are evaluated using the ϕ -coefficient of Pearson's χ^2 -correlation ([12], [13]) for dichotomous variables. Significant and strong correlations are deduced and presented.

Lastly, the open questions were evaluated using open coding.

V. RESULTS

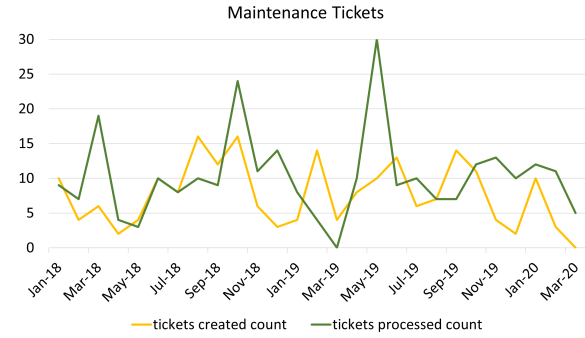
A. Ticket Statistics

The timelines in Figure 2(a) and Figure 2(b) show the development of ticket counts over time to answer RQ2.3. The maintenance tickets are created and processed more or less continually.

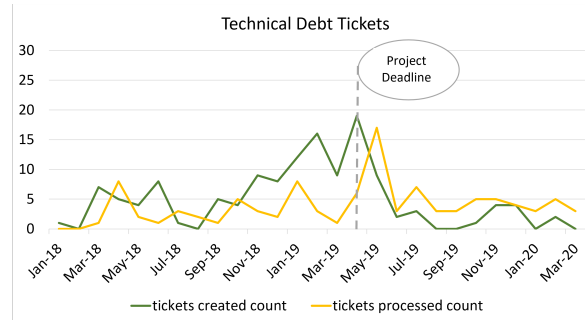
The timeline for TD tickets show a peak of contracted TD in spring 2019 due to a project with a tight timeline. No maintenance or TD tickets were processed in March 2019. Accordingly, the TD tickets as well as the maintenance tickets show a processing peak after the reached deadline at the end of April 2019. In summer and autumn 2019 the TD tickets with a lower priority are processed which means that this project is still not finished (see Figure 1). Other projects start in parallel to this project and development capacities must be divided between the projects.

B. Survey

1) *Assessment of the Framework*: Most of the survey participants agree that it is generally reasonable to record



(a) Maintenance tickets by time



(b) TD tickets by time

Fig. 2. Ticket statistics with project deadline at the end of April 2019

and process all four types of tickets (Fig. 3(a)/ 3(d)). Most of them also agree that the procedures of the framework for recording and processing are reasonable (Fig. 3(b)/ 3(e)). When asked if the framework is working as intended, the agreement decreases, but still, most of the survey participants state that the procedures work well and are therefore helpful (Fig. 3(c)/ 3(f)). For RQ1.1 and RQ1.2 this means that the framework is reasonable and feasible in practice.

2) *Effects and benefits of the Framework*: First, the survey participants are asked if they and their team usually recognize that TD items are or were contracted before or after the point in time when they were contracted. The goal of this question is to identify whether the participants are aware of the contraction of TD items to answer RQ2.1. As can be seen in Figure 4(a) there is no significant difference between the units. Both units state with more than 75.0% agreement that they usually recognize it when TD items are contracted on.

An obvious difference can be seen in Figure 4(b). The units are asked whether they compare different solution alternatives and whether they estimate respective implementation costs, principal, and interest rates for taking on TD. To answer RQ2.2, we assume conscious decisions to take on TD, if these comparisons are made. 75.0% of the participants of the observed unit agree to compare quick and optimum (or more) solutions and even calculate implementation cost for this comparison. For the comparison unit 80.0% survey participants do not agree to this which means they usually do not compare different solutions and estimate implementation costs.

³<https://www.ibm.com/de/de/analytics/spss-statistics-software>

⁴<https://www.maxqda.de/>

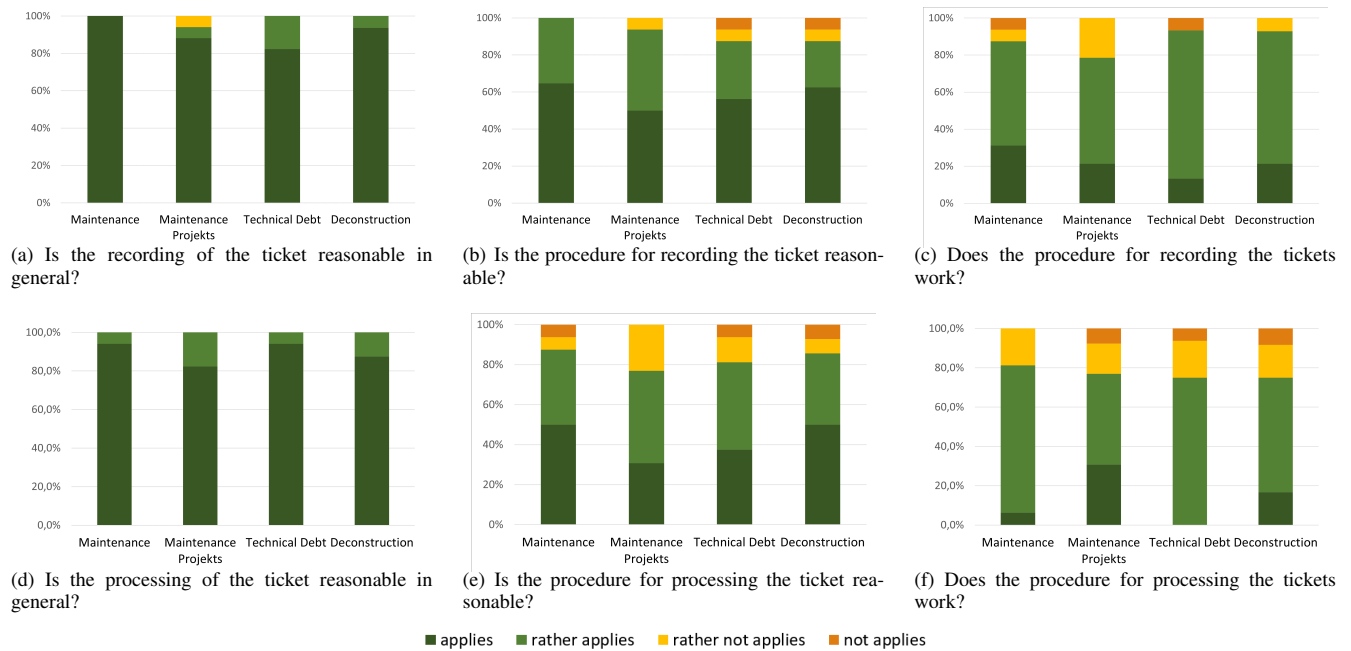


Fig. 3. Appropriateness and usability of the framework as assessed by the observed IT unit

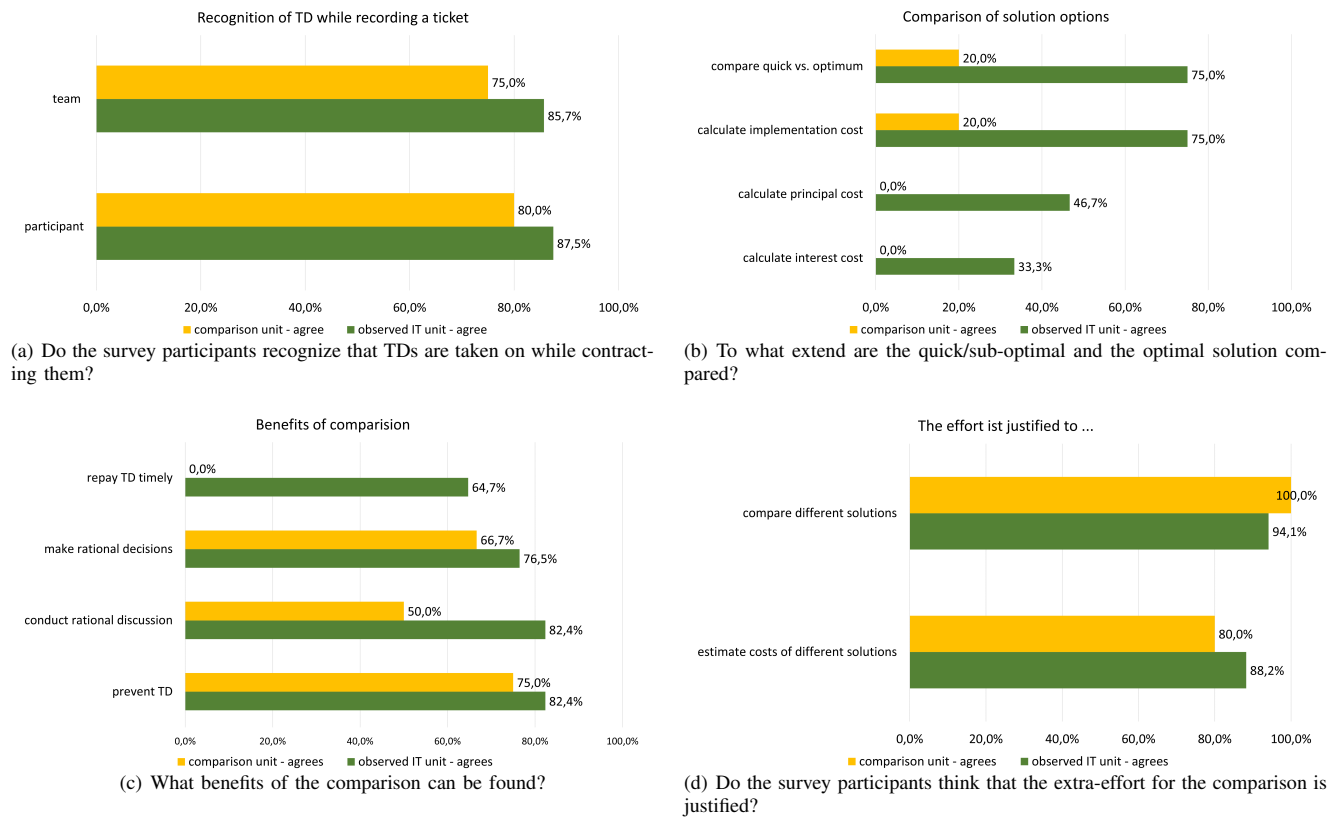


Fig. 4. Effects and benefits of the Framework - Descriptive Statistics

TABLE III
SIGNIFICANCE OF H_R -HYPOTHESIS

H_R	variable	exact significance
$H_{R,1}$	comparison	0.058
$H_{R,2}$	implementation cost	0.048
$H_{R,3}$	principal cost	0.002
$H_{R,4}$	interest cost	0.015

Many participants of the observed unit agree that they also estimate principal (46.7%) and interest costs (33.3%), while no participants of the comparison unit do this.

These observations lead to the following research hypothesis (H_R) in comparison to respective null hypothesis (H_0):

$H_{R,1}$: The observed unit compares quick and optimum solutions more often than the comparison unit. ($H_{0,1}$: The observed unit does not show a different behavior than the comparison unit in terms of comparing quick and optimum solutions.)

$H_{R,2-4}$: The observed unit estimates the implementation (and principal and interest) costs for different solutions more often than the comparison unit. ($H_{0,2-4}$: The observed unit does not show a different behavior than the comparison unit in terms of estimating the implementation (and principal and interest) costs for different solutions.)

Table III shows the significances of these hypotheses that were measured with the Mann-Whitney U-Test. The exact significance for $H_{R,2-4}$ hypotheses is below 0.05 therefore these hypothesis can be accepted as they are significant to the 5% significance level. This means the possibility that these hypotheses are wrong is lower than 5%. The exact significance for $H_{R,1}$ hypothesis is 0.058 which means the possibility that this hypothesis is wrong is only a little higher than 5%. It can be accepted to the 10% significance level. In summary, this means that the adoption of the framework leads to the named effects of a more conscious comparison of different solutions options and their respective costs.

To answer RQ3.1 and 3.2, the participants are asked what benefits they expect (comparison unit) or assessed (observed unit) from the comparison of different solution option and therefore of the conscious contraction of TD. Figure 4(c) shows that both units mostly agreed that TD could be prevented by this comparison. Especially the observed unit assessed that discussions were led more rational (82.4% agreement) and more rational decisions were made (76.5% agreement). Timely repayment (RQ2.3) of TD was not expected by the comparison unit but assessed by many participants of the observed unit (64.7% agreement). This may partly be due to a misunderstanding that the observed unit was unconsciously assessing the benefits of the entire framework and not just the benefits of the comparison of solution options. There is no obvious difference between the units regarding these questions which means no hypotheses can be derived.

To answer RQ3.3, the participants are asked whether the effort for comparing and estimating the different solutions is justified by the gained or expected benefits. Both units mostly

TABLE IV
CORRELATIONS OF EFFECTS VS. BENEFITS

(a) Comparing solutions vs. assessing TD prevention and rational decisions as benefit

		comparing solutions		ϕ -coeff
		not applies	applies	signif.
TD prevention	not applies	42.9%	7.7%	0.419
	applies	57.1%	92.3%	0.061
rat. decisions	not applies	66.7%	7.7%	0.623
	applies	33.3%	92.3%	0.007

(b) Estimating the implementation costs of different solutions vs. assessing rational decisions and rational discussions as benefit

		estim. impl. costs		ϕ -coeff
		not applies	applies	signif.
rat. discussions	not applies	50.0%	8.3%	0.471
	applies	50.0%	91.7%	0.035
rat. decisions	not applies	57.1%	8.3%	0.535
	applies	42.9%	91.7%	0.020

(c) Estimating the principal costs vs. assessing rational decisions and rational discussions as benefit

		estim. princ. costs		ϕ -coeff
		not applies	applies	signif.
rat. discussions	not applies	41.7%	0.0%	0.456
	applies	58.3%	100.0%	0.049
rat. decisions	not applies	45.5%	0.9%	0.495
	applies	54.5%	100.0%	0.037

think that the effort is justified (>80.0% agreement). There is also no obvious difference between the units as can be seen in Figure 4(d). No hypotheses are derived.

3) *Correlations regarding the benefits of the framework:*

As shown in the previous section, the most significant difference that arise by adopting the framework is the ongoing comparison between different solutions for every new ticket. As pointed out in Section IV-B3, the benefits of the framework cannot be assessed by the comparison unit. To identify the benefits of the comparison, we therefore analyze correlations between all survey participants who state that they compare different solutions and the benefits they therefore assessed.

Table IV shows all relevant and significant correlations that can be found.

It is important to point out that a correlation between two variables usually lead to two possible interpretations. One interpretation could be that variable A is the cause and B is the consequence and vice versa. In the following part only the most likely and reasonable interpretations are presented.

Table IV(a) shows the correlation between participants that compare sub-optimal and optimal (or more) solutions and participants that have assessed the benefits of TD prevention and rational decision making. The correlation for TD prevention is medium strong (0.419) but with an asymptotic significance of 0.061 only significant to the 10% significance level. The correlation for rational decision making is strong (0.623) and with an asymptotic significance of 0.007 significant even to

the 1% significance level. This can be interpreted to mean that comparing different solution options leads to the named benefits. The possibility that this correlation does not exist, and our assumptions are wrong is higher for TD prevention ($<10\%$) than for rational decision making ($<1\%$).

Table IV(b) shows the correlation between participants that are estimating the implementation costs of different solutions and participants that have assessed the benefits of rational discussions and rational decision making. These correlations are both medium strong (0.471 and 0.535) and significant to the 5% significance level (0.035 and 0.02). This can be interpreted to mean that estimating the implementation costs of different solutions leads to the named benefits.

Table IV(c) shows the correlation between participants that are estimating the principal costs and participants that have assessed the benefits of rational discussions and rational decision making. These correlations are both medium strong (0.456 and 0.495) and significant to the 5% significance level (0.049 and 0.037). This can be interpreted to mean that estimating the implementation costs of different solutions leads to the named benefits.

4) *Open Questions*: Most comments were just made one time and are therefore just single opinions. There were two points that were mentioned three times each.

First, “TD is a topic” expresses that it is already helpful to talk about the topic of TD and not to ignore it which is related to the goal to raise the of the overall TD awareness. This was pointed out by one participant: *“I actually see the main advantage of a conscious handling of the topic in the matter as such, because otherwise it is a topic that is almost ignored in everyday development”*.

The second code is “Sometimes TD is OK” where the participants were considering situations in which cases it is reasonable to keep TD, e.g. in legacy systems. The following was stated by another participant: *“If a system is to be replaced, then I consider technical debts to be justifiable if they disappear after 1-2 years anyway”*.

One other point was mentioned two times which is “Evaluation is difficult”. In research this topic is known as TD prioritization problem. It was phrased by one participant as follows: *“Even if the ‘textbook’ says that the current implementation does not correspond to the architectural specifications, it still seems to be a personal opinion that decides on the fundamental correctness of the specification”*.

Two relevant drawbacks were mentioned by one participant, respectively. One participant stated that the distinction of the categories and the separation of their responsibilities may lead to a separation of the team. *“Although we keep saying that we are a team, we often stay separate and the different tickets in particular help make the differences between the team parts visible”*. Another participant thought that the framework *“(slightly) contradicts the agile principle of providing a functioning solution with company value as quickly as possible. At the same time, however, it levels the stable ground so that we can continue to work on new requirements quickly and without production errors. The advantage outweighs”*.

VI. DISCUSSION

A. Framework application

RQ1.1/RQ1.2: *Do practitioners find the framework reasonable? Are the processes of the framework feasible in practice?*

Figures 3(a) to 3(f) show that there is very little disagreement. Most survey participants find the framework reasonable and think that it does work as intended. Only few participants think that it must at least be optimized. The ticket statistics (see Section IV-A) also indicate that the tickets are recorded and processed as intended. Furthermore, the framework is already used in the IT unit for more than two years. It can therefore be concluded that in an overall view the framework is feasible in practice and that the following findings based on this framework are valid.

B. Framework effectiveness

RQ2.1: *Does the framework lead to raised awareness for the contraction of TD?*

Figure 4(a) shows that most participant are aware of it when they contract TD. Yet, there seems to be no difference between the observed and the comparison unit. The answers to the open questions support the finding that the awareness was raised considerably (“TD is a topic”) and that the participants started to consider when to take on or keep TD and when not to (“Sometimes TD are OK”).

RQ2.2: *Are the TD items taken on more consciously when using the framework?*

Figure 4(b) reveals that in comparison the observed unit shows a significantly different behavior than the comparison unit. The members of the unit using the framework are comparing different solutions, i.e. optimal and sub-optimal, and their costs before deciding for one of them. They may still take on TD by choosing the sub-optimal solution, but this is a conscious decision.

RQ2.3: *Are the TD items paid back timely?*

If TD has to be contracted it is mostly paid back timely which can be seen in Figure 2(b). Even though not all TD tickets may be paid back during their respective projects as intended by the framework. Figure 4(c) shows that two thirds of the participants of the observed IT unit perceive that TD items are paid back timelier. Nevertheless, not all TD tickets are paid back and a third of the participants are still dissatisfied with the timely repayment.

C. Framework benefits

RQ3.1/RQ3.2: *Can TD be prevented by the adoption of the framework? Are there other benefits arising from the adoption of the framework?*

The most obvious effect of the framework is the conscious contraction of TD by comparing different solution options. In Figure 4(c) the benefits that arise by these are evaluated. Both units assessed or expected benefits of this comparison.

Furthermore, the correlations between the benefits and participants that did this comparison are significant. This paper was able to show that the comparison led to more rational discussions and decisions. The error probability for

this assumption is lower than 5%. The prevention of TD was assessed with 6.1% only to the lower 10% significance level which means that it is only a little less likely that this observation is valid. These benefits can therefore be referred to the use of the framework.

RQ3.3: *Do these benefits justify the additional effort?*

Finally, Figure 4(d) shows that most participants think that the additional effort for comparison and cost estimation is justified and thus backs up the usefulness of this framework.

D. Threats to Validity

Construct Validity: To enhance construct validity, we questioned and compared two units one of which did adopt the framework and one that did not use any method for managing TD. To ensure comprehensibility and to optimize the questionnaire, we did pilot tests. It could be seen as a problem that the survey did not directly but only in open questions ask for drawbacks. The questions whether the effort for the benefits is justified should have remedied parts this problem. Additionally, we not only evaluated the questionnaire but also the ticket statistics which further improves the construct validity, e.g. when evaluating the timely repayment of TD.

Conclusion Validity: The paper was able to show statistically significant differences between the observed and the comparison unit. We used SPSS and standard statistic techniques for the evaluation of the significance of the findings. On the other hand, the sample was small. The comparison unit was considerably smaller than the observed unit and had a slightly different organization. By choosing a unit that was led by the same unit manager this threat was minimized. The survey response was only about 50% of the respective units which can lead to deviations in the evaluation.

Internal Validity: Some valuable effects were observed, but it is possible that the same effects could have been observed in other IT units. This threat was minimized by the comparison with a unit that did not use the framework. Nevertheless, just the management of TD could have led to a better developer morale like presented in [14] and therefore to good ratings. This threat can and should be minimized by replicating the study in other constellations. As correlations do not directly lead to causalities, the mentioned interpretations of the correlations should be validated, e.g. by follow up interviews.

External Validity: It is an important benefit of this paper that the framework was developed and tested in an industry environment and that the framework has been used for a long time. Nevertheless, this was just a case study and can be biased. The framework adoption and the study should be replicated in other units and companies, especially in other economic sectors and countries. By providing the framework information in this paper and the questionnaire and raw data⁵ the study can be replicated to back up the findings.

⁵<https://doi.org/10.5281/zenodo.4616485>

E. Related Work

TD Prevention: In a systematic mapping study in 2015 Li et al. [2] identified eight TD activities one of them being TD prevention which was with seven studies between 1992 and 2013 also one of the obviously understudied activities. In 2016 Yli-Huumo et al. [3] find TD prevention to be “one of the most influential activities of the eight TDM activities that a development team can conduct”. In 2018 Rios et al. [4] started to fill this gap by indicating that TD can be prevented, and that TD prevention is a worthwhile research topic. In their follow-up work [8], they identified actions and impediments to TD prevention by evaluating a questionnaire filled by practitioners. The work did not provide ideas how to implement these actions in practice or how to deal with the impediments.

Timeline Problem: As already mentioned in Section I one of the main root causes for TD to be found in all papers that deal with TD causes is the problem of tight timelines (e.g. [8], [4], [7], [6], [1]). To the best of the authors knowledge no papers contribute a solution to this problem. By using TD tickets, this framework fills this gap and provides a practical method that has impact on two of the actions listed in [8]: (I) “Following a well-defined project planning” by integrating TD management into project management, (II) “Having an effective team” with “good communication on the team” as shown by the impact the framework has on team discussions. Codabux et al. [15] also list different strategies to prevent TD, e.g. “Education and Training”, “Pair Programming”, “Conformance to process and standards”, but they also do not provide a solution on how to stick to the strategies under time pressure. This framework puts the responsibility for TD ticket repayment to the business analysts and therefore project management. Thereby, this paper proposes a solution to the problem of “Split of budget in Project budget and Maintenance budget boosts the accumulation of debt” ([7]).

TD Awareness: One important effect of this framework is the raised awareness for contracting TD. Kruchten [16] already mentioned the importance of awareness: “The first step is awareness: identifying debt and its causes”. In a survey with more than 1,800 participants, Ernst et al. [17] find that that 79% of the participants strongly agree that “Lack of awareness of TD is a problem”. Tonin et al. [18] researched the effects of TD awareness by conducting a classroom study. In contrast to that this paper presents a method to raise the awareness not only for developers but also for business analysts and managers. Finally, this paper presents an evaluation in industry, which provides a higher external validity.

TD Management Case Studies: Other case studies concerning TD management are for example [19] and [20]. They present research findings in the same area as this study as they try to tackle impact of TD in industry. Yet, in [19] Martini et al. follow a more organizational solution to TD management. Guo et al. [20] focus their framework on the TD identification, TD measurement and TD monitoring. Furthermore, Guo et al. provide an idea for TD prioritization. It could be interesting future work to evaluate whether the frameworks can be inte-

grated to a more holistic approach.

VII. CONCLUSION AND FUTURE WORK

With this paper we present and evaluate a framework that includes TD management in project management by so-called TD tickets. TD tickets hand over the responsibility for TD contracted during a project to the project management of the same project. This creates a feedback loop of team and management behavior that leads to TD prevention. As it was developed and extensively used in practice, the framework shows a high external validity and provides the research community with and insight into the practical handling of TD.

The adoption of the framework led to a higher TD awareness and conscious contraction of TD in the case company. This on the other hand leads to the prevention of TD, more rational discussions between the unit members and more rational decisions. Furthermore, the TD items taken on during a project are part of the project and will be paid back timelier. While the idea of maintenance tickets and projects is not new, the idea of TD ticket is a novel approach that could not be found in research literature to the best of the author's knowledge.

For industry, this paper presents a framework that can be adopted, refined, and adapted to their own needs. The idea of including TD management in project management can be an incentive to develop a similar approach. Also, the importance of raising TD awareness and contracting TD intentionally by taking the time to compare different solution options was presented in this paper. The advantages shown by this paper can be helpful when negotiating options to handle TD with the IT management.

For researchers, this paper provides insights to procedures for managing TD developed in industry. The paper indicates the importance to provide a solution for handling and preventing TD even under the pressure of a tight timeline. The paper also confirms the problem that the responsibility for TD needs to be transferred to the whole unit and cannot be solved by developer teams alone. The framework presented in this paper provides a solution for these problems.

For future work, the framework could be complemented by other TD activities, e.g. TD prioritization, TD monitoring and TD identification, and their impact could be evaluated. Especially ideas to TD Prioritization may help to optimize the repayment phase as this was mentioned as a problem. The management of deconstruction tickets in particular is an interesting and promising research topic and should be evaluated in future. Timely repayment of TD is sometimes not performed due to low priority or missing awareness of competing goals. Further research with respect to project management and business value estimation is necessary. Finally, the prioritization of maintenance tasks including TD should be assessed between different application domains and industries.

ACKNOWLEDGMENT

We would like to thank *Gruner+Jahr GmbH - Media Sales Services* and *- Smart Business Solutions* for their contributions to the framework, for completing the questionnaire and for their support of this paper.

REFERENCES

- [1] P. Avgeriou, P. Kruchten, I. Ozkaya, and C. Seaman, "Managing Technical Debt in Software Engineering," *Dagstuhl Reports*, vol. 6, no. 4, pp. 110–138, 2016.
- [2] Z. Li, P. Avgeriou, and P. Liang, "A systematic mapping study on technical debt and its management," *Journal of Systems and Software*, vol. 101, pp. 193–220, mar 2015.
- [3] J. Yli-Huomo, A. Maglyas, and K. Smolander, "How do software development teams manage technical debt? – An empirical study," *Journal of Systems and Software*, vol. 120, pp. 195–218, 2016.
- [4] N. Rios, R. O. Spínola, M. G. De Mendonça Neto, and C. Seaman, "A study of factors that lead development teams to incur technical debt in software projects," in *Proceedings - 44th Euromicro Conference on Software Engineering and Advanced Applications, SEAA 2018*, 2018, pp. 429–436.
- [5] W. Cunningham, "The WyCash portfolio management system," in *Proceedings of the Conference on Object-Oriented Programming Systems, Languages, and Applications, OOPSLA*, vol. Part F1296, no. 2, 1992, pp. 29–30.
- [6] R. Verdecchia, P. Kruchten, P. Lago, R. Verdecchia, P. Kruchten, and P. Lago, "Architectural Technical Debt : A Grounded Theory," in *European Conference on Software Architecture (ECSA 2020)*, A. Jansen, I. Malavolta, H. Muccini, I. Ozkaya, and O. Zimmermann, Eds. Springer, Cham., 2020, pp. 202–219.
- [7] A. Martini, J. Bosch, and M. Chaudron, "Architecture technical debt: Understanding causes and a qualitative model," in *Proceedings - 40th Euromicro Conference Series on Software Engineering and Advanced Applications, SEAA 2014*, 2014, pp. 85–92.
- [8] S. Freire, N. Rios, M. Mendonça, D. Falessi, C. Seaman, C. Izurieta, and R. O. Spínola, "Actions and impediments for technical debt prevention: Results from a global family of industrial surveys," in *Proceedings of the ACM Symposium on Applied Computing*, 2020, pp. 1548–1555.
- [9] S. McConnell, "Managing technical debt," *Construx Inc.*, 2008. [Online]. Available: <http://www.construx.com/uploadedfiles/resources/whitepapers/ManagingTechnicalDebt.pdf>
- [10] A. Joshi, S. Kale, S. Chandel, and D. Pal, "Likert Scale: Explored and Explained," *British Journal of Applied Science & Technology*, vol. 7, no. 4, pp. 396–403, 2015.
- [11] H. B. Mann and D. R. Whitney, "On a Test of Whether one of Two Random Variables is Stochastically Larger than the Other," *The Annals of Mathematical Statistics*, vol. 18, no. 1, pp. 50–60, 1947.
- [12] J. Cohen, *Statistical Power Analysis for the Behavioral Sciences*. New York, USA: Lawrence Erlbaum Associates, 1988.
- [13] K. Pearson, "I. Mathematical contributions to the theory of evolution. —VII. On the correlation of characters not quantitatively measurable," *Philosophical Transactions of the Royal Society of London. Series A, Containing Papers of a Mathematical or Physical Character*, vol. 195, no. 262-273, pp. 1–47, 1901.
- [14] T. Besker, H. Ghanbari, A. Martini, and J. Bosch, "The influence of Technical Debt on software developer morale," *Journal of Systems and Software*, vol. 167, 2020.
- [15] Z. Codabux, B. J. Williams, and N. Niu, "A Quality Assurance Approach to Technical Debt," in *Proceedings of the International Conference on Software Engineering Research and Practice (SERP)*, 2014.
- [16] P. Kruchten, R. L. Nord, and I. Ozkaya, "Technical debt: From metaphor to theory and practice," *IEEE Software*, vol. 29, no. 6, pp. 18–21, nov 2012.
- [17] N. A. Ernst, S. Bellomo, I. Ozkaya, R. L. Nord, and I. Gorton, "Measure it? Manage it? Ignore it? Software practitioners and technical debt," *10th Joint Meeting of the European Software Engineering Conference and the ACM SIGSOFT Symposium on the Foundations of Software Engineering, ESEC/FSE 2015 - Proceedings*, pp. 50–60, 2015.
- [18] G. S. Tonin, A. Goldman, C. Seaman, and D. Pina, "Effects of Technical Debt Awareness: A Classroom Study," in *18th International Conference, XP 2017*, H. Baumeister, H. Lichter, and M. Riebisich, Eds., 2017, pp. 84–100.
- [19] A. Martini, L. Pareto, and J. Bosch, "Towards introducing agile architecting in large companies: The CAFFEA framework," in *Lecture Notes in Business Information Processing*, vol. 212, 2015, pp. 218–223.
- [20] Y. Guo, R. O. Spínola, and C. Seaman, "Exploring the costs of technical debt management – a case study," *Empirical Software Engineering*, vol. 21, no. 1, pp. 159–182, 2016.