Investigating Cognitive Workload during Comprehension and Application Tasks in Software Testing

Daryl Camilleri
Department of Computer Science
University of Malta
daryl.camilleri.11@um.edu.mt

Chris Porter
Department of Computer Information Systems
University of Malta
chris.porter@um.edu.mt

Mark Micallef
Department of Computer Science
University of Malta
mark.micallef@um.edu.mt

***Abstract* — Software testers are an integral part of software development teams, and consequently need to understand from different perspectives the project entrusted to them. While developers might be required to understand a particular module or area of specialisation within a project, testers' comprehension requirements are more far-reaching [1]. Gaining insights into how testers fare in different comprehension tasks is useful because it sheds light on how we could potentially support the efforts of the testing community. This paper reports the results of a laboratory experiment involving 15 professional software testers. Using NASA Task Load Index as our instrument of choice, we asked participants to carry out eight comprehension and application tasks across four categories (test case design, test automation, bug finding and adequacy analysis). We then analysed the data collected to seek to understand the effect of different task types, education level and participant experience on effectiveness and cognitive workload.**

The results suggest that, while experience is a key element in successful task completion, this is also influenced by task type. In fact, the more experienced persons actually tended to fare worse than their less experienced counterparts in certain tasks (namely, test case design and adequacy analysis). Level of education had no significant bearing on successful task completion but differences in cognitive workload could be observed for both experience and education-level variables.

I. Introduction

Program comprehension is a prerequisite for the effective completion of most tasks in a software engineering context. Sneed [1] argues that the program code is not the only artefact that should be of concern. More specifically, effective members of a development team would need to understand the code, the environment in which it is deployed, the domain which it serves, the stakeholders involved and so on. This is especially the case for software testers, whose comprehension requirements tend to be more demanding than those of a developer. Vanitha and Alagarsamy [2] define software testing as "one of the five main technical activity areas of the software engineering life-cycle that still poses substantial challenges". Other than what seems to be a simple process of checking a sample of runs, software testing encompasses various intricate challenges and enfolds a mixture of activities and techniques.

Indeed, with the constantly growing demand for software and its complexity, ensuring that the software performs as per the required level of quality is becoming highly critical and expensive [3]. From a comprehension perspective, while developers might be required to understand a particular module or area of specialisation within a project, the testers' comprehension requirements usually span a significantly wider area of a project. They also typically deal with a broader range of stakeholders and are expected to carry out a variety of tasks having significant comprehension prerequisites.

In this context, it would be desirable to gain insights into the cognitive workload experienced by testers as they comprehend a task, understand what is required and apply their understanding in completing the task. Such an insight would help guide recruitment, training, work allocation and mentoring efforts within organisations. To this end, we approached this work by posing the following research questions:

- RQ1: How are cognitive load and effectiveness in software testers affected when carrying out different types of testing tasks?
- RQ2: How are cognitive load and effectiveness influenced by an individual's experience and education?

The rest of this paper is organised into five further sections, with Section II providing the necessary background on cognitive workload measurement. Section III outlines the methodology we have adopted in this study. Section IV explores the results and provides the bases for Section V, where the results are discussed in the context of the research questions posed above. Finally, through Section VI, we submit proposals regarding future work, based on the observations made.

II. Cognitive Workload

The term *cognitive workload* (or *mental workload*) is widely used and, while having many definitions, most of them converge on two main aspects: stress and strain [4]. The first refers to the demands of the task, whereas the second refers to its impact on the person carrying it out.

When adapting a definition of mental workload, Galy et al. [5] make reference to Young and Stanton's [6] claim that one should also consider "the amount of attentional resources necessary to perform task as a function of task demand,

DOI reference number: 10.18293/SEKE2022-155

environmental context in which the task is performed, and past experience of individual with task" [5].

A. Measuring Cognitive Workload

The two most widely used instruments for measuring cognitive workload are the subjective workload assessment technique (SWAT) [7] and the NASA-Task Load Index (NASA-TLX) [8]. SWAT measures three dimensions of cognitive workload (time load, mental effort and psychological stress) whereas NASA-TLX has six subscales (mental demand, physical demand, temporal demand, performance, effort and frustration). We have chosen to focus on NASA-TLX because of its wide application across different domains, and its multi-dimensional assessment of workload, which provides a richer insight into the sources of workload over SWAT.

1) NASA-Task Load Index: NASA-TLX [8] is a multi-dimensional scale designed to obtain workload estimates from one or more operators as they are performing a task or immediately afterwards. Since its publication in the 1980s, NASA-TLX has been cited extensively and used in several fields, ranging across nuclear power plant control rooms, certification of aeroplanes, operating rooms, computer-generated fighting, and designing of websites [9]. It consists of a multi-item questionnaire which, when processed, provides an overall task load index with a range between 0 and 100. The higher the rating, the more demanding the task would be. The instrument also provides measurements of six subscales, as indicated above (i.e., mental demand, physical demand, temporal demand, performance, effort and frustration). A weighted load index could also be obtained following a pairwise comparison of these subscales. Like other interface metrics and questionnaires, the TLX application cannot tell what to repair, but it assists research in understanding if variations made to an interface generated a better or deteriorated workload. Although the instrument has most commonly been used in studies that contain physical components [10], existing literature on the topic also includes a substantial amount of work where the study was used to analyse ergonomics in software systems in which the physical component was not necessarily of concern. On the basis of a survey of 500 studies, undertaken 20 years after first being developed [9], its creator noted that while the instrument was originally developed for use in the aviation sector, it had grown to be used in a wide variety of sectors, not least in software engineering.

III. Methodology

In this section, we present the methodology and discuss key decisions taken during its design. All material related to the methodology and results is available on our OSF repository[1].

A. Task Design

Task selection and design was of critical importance in this study. Given the rich spectrum of activities in which software testers are involved, it was important to choose a reasonable subset of tasks that could be carried out in the limited context and time-frame of a lab-based experiment.

1) Tasks Taxonomy: Hrabovská et al. [11] carried out a wide ranging review of software-testing process models. As part of this review, they identified five groups of practices, as follows: planning (21 practices); design (9 practices); set-up (12 practices); execution (13 practices); and monitoring (17 practices). These groups collectively characterise the spectrum of tasks that testers carry out, depending on which process model they follow. We employed this knowledge to guide us in selecting a pragmatic subset of tasks that could be carried out in a lab setting, in a restricted amount of time (approximately one hour). This led us to focus on these four practices: (1) test case design; (2) test automation; (3) exploratory test execution or bug finding; and (4) test adequacy analysis.

In order to minimise participant fatigue, we set out to ensure that the experiment would take approximately one hour. After factoring in an estimated 10 minutes for participant on-boarding and exit interviews, we calculated 40 minutes for data collection. Hence, we deemed it best to design a series of eight tasks, each of which we estimated would take 3-5 minutes to complete. Each task was to be preceded by a 2-minute calming fish-tank video, which enabled participants to reset their mental state in preparation for the task. We also opted to include two practice tasks to be carried out at on-boarding stage, and thus helping to reduce possible participant anxiety due to unfamiliarity during data collection.

We decided to distribute the eight tasks as follows: three *test case design* tasks of increasing difficulty, three *test automation* tasks of increasing difficulty, one *bug-finding* task and one *adequacy analysis* task. During each task, participants were required to read a concise specification or a short snippet of C# code presented on screen. They were then asked to verbally explain how they would complete a specific task related to what they were observing on the screen. For example, after being presented with the specification for a feature, participants were asked to outline how many tests would be required for testing the implementation of that specific feature. The full set of tasks can be found in our replication pack.

B. Data Collection

Data was collected in two ways. Firstly, participants consented to the recording of their onscreen activity and their voice. This enabled us to evaluate, at a later stage, the success rate in the completion of each task. Secondly, participants filled in a NASA-TLX evaluation on paper for each task.

C. Experimental Procedure

Participants were welcomed to the lab, introduced to the experiment and given time to review and sign consent forms. Once the formalities were completed, the participants were introduced to the two practice tasks in order to familiarize themselves with the experiment. At this point, participants iteratively watched a fish-tank video to reset their mental state, carried out a task and completed a NASA-TLX assessment for the task. When all tasks were completed, an exit interview was carried out and the experiment was concluded.

[1]https://osf.io/gnyv7/?view_only=963454403e5c42acbd344d8d8e2c80cd

IV. Results

This section explores the data collected in the experiment guided by the research questions posed in Section I.

A. Participant Demographics

Following initial screening of participants, we selected 20 individuals, of whom 15 made it to the lab and successfully completed the assigned tasks. In terms of experience, 3 participants (20%) had up to 2 years' experience, 8 participants (53%) had between 3 and 5 years of experience, and 4 participants (27%) had 6 years' experience or more. Education levels consisted of 2 participants (13%) having a diploma level of education, 9 participants (60%) holding a first degree, and 4 participants (27%) having postgraduate qualifications. Unfortunately, the gender balance of our cohort was heavily skewed towards male participants, who constituted 14 participants (93%).

B. Task Performance

We post-processed the collected data towards establishing the extent to which participants were successful in their allocated tasks. For each task, we classified the participants' individual performance as *not successful*, *mostly successful* or *successful*. Although determining task success was not a primary goal of this experiment, it provided another dimension from which to evaluate the research questions.

Out of 120 attempts, 24 (20%) were unsuccessful, 44 (37%) were mostly successful and 52 (43%) were successful. Success decreased as tasks became more difficult within each category. Whilst 60% of participants completed the first test design task successfully, this was only the case with 33% in the third task. Similarly, 80% of participants completed the first test automation task successfully, whereas none were successful with the third one. However, is worth noting that while the number of completely unsuccessful candidates increased with each level of difficulty in test case design, the number of unsuccessful attempts at test automation tasks remained constant at 7% (one participant). The bug-finding task had a reasonable level of success, when taking into account that the participants did not know that they were expected to find 10 bugs. Finally, the participants seemed to find test-adequacy analysis the most challenging, with only 27% getting it right and 47% getting it wrong.

1) Effect of Experience on Task Success: We analysed the success by task type and participant experience. The data indicates that experience is a determining factor in task success with 0-2, 3-5 and 6+ year cohorts being successful or mostly successful 92%, 81% and 69% of the time respectively. It is interesting to note that, overall, relatively inexperienced testers had a higher success rate, outperforming individuals more experienced in test case design and test adequacy analysis. We believe that this is due to the recent nature of their formal training. However, experience seems to play a key role in determining success in test automation and bug-finding tasks.

2) Effect of Education on Task Success: When analysing task success by education, the data at hand suggested that the level of education did not have a significant impact on performance. When considering all tasks collectively, the participants having a diploma were successful or mostly successful 81% of the time, participants having a first degree were successful 79% of the time, and those with postgraduate degrees were successful 81% of the time. This contrasts with the more varied success rates when grouping participants by experience.

C. Overall Cognitive Workload

We began our analysis by taking a high-level view of the cognitive workload generated by tasks among our participants. This was done by analysing the distribution of NASA-TLX scores across task types and participants. We did this from the point of view of participant experience and the participants' level of education.

1) Analysing Workload by Experience: When grouping the NASA-TLX scores by task type and participant experience, one notices that the general trend was for cognitive workload to decrease with experience. This was particularly evident in test case design and test adequacy tasks. Both are activities which are taught in all testing curricula, but require repeated practice in order to be applied confidently.

Interestingly, the less proficient testers experienced a lower cognitive workload when carrying out implementation tasks and bug-finding tasks. We believe that this is due to a number of reasons. Firstly, less experienced testers are likely to be fresh graduates, having completed a degree programme focusing on programming skills. Hence, their comfort zone at this point would consist mostly of coding. Secondly, it is probable that experienced testers would specialise in certain subfields of software testing. Therefore, a test engineer specialising primarily in building regression test automation frameworks would be out of touch with bug-hunting skills (and vice-versa). This argument is further strengthened when one notes that the more experienced testers were subject to extreme upper and lower whisker values, which indicate individuals who have specialised in or away from that particular skill.

2) Analysing Workload by Education: When considering all NASA-TLX scores regardless of tasks, one notes that participants with the lowest level of education tended to experience the lightest cognitive workload. The highest score for this cohort was 71, compared to 97 and 73 for undergraduates and postgraduates respectively. This pattern was driven by scores related to *test design*, *test automation* and *bug-finding* task categories, but not *test adequacy analysis*. The measurements for the latter category suggest that higher levels of education result in a lighter cognitive workload when carrying out adequacy analysis. However, it is to be noted that, since there was only one adequacy analysis task and one bug-finding task, the sample plots for these tasks were equivalent to the number of participants in each education group. For instance, the sample of diploma graduates was only 2.

The cohort of participants exposed to the largest cognitive load tended to be first-time graduates, with the top of their interquartile range clearly exceeding 70 for test automation tasks and bug-finding tasks. The resulting mean overall tasks for undergraduates was 52, with 44 for diploma holders and 49 for postgraduates.

D. Individual NASA-TLX Scales

We also examined how participants fared in individual scales of NASA-TLX, the main categories being: (1) mental demand; (2) physical demand; (3) temporal demand; (4) effort; and (5) frustration. It is to be noted that, although we have charted the values for physical demand, we have opted not to analyse this aspect for these tasks. The main reason for this being that it did not offer a scale of interest for our tasks.

1) NASA-TLX Scales by Experience: Beginning with the scales related for test case design tasks (Figure 1(a)), we have noted that the less proficient participants experienced the heaviest cognitive load in all the scales, with notable peaks for mental demand, frustration and effort. This is interesting when considering that this group accomplished these tasks at optimal levels, outperforming the other two cohorts. It is also interesting to note that their performance score was higher than that of the other cohorts, indicating that they did not feel they were successful with the task.

In the test automation tasks (Figure 1(b)), mental demand, frustration, effort and performance were quite similar for all three groups, with the least experienced group recording the lowest values. As regards the temporal demand value, there was a marked spike among the participants with 3-5 years' experience.

The radar chart for bug finding (Figure 1(c)) indicates elevated levels of effort for both the group with 0-2 years' experience and the group having 3-5 years' experience. The more experienced testers in the group seem to have been minimally affected in all scales except mental demand. Moreover, on the basis of the compiled data, we established that the most experienced cohort performed exceptionally well in this task.

Finally, following an analysis of test adequacy analysis (Figure 1(d)), we observed a somewhat similar picture to test case design. More precisely, the least experienced cohort registered the highest levels of scales, compared to other cohorts, even obtaining an average score of 75 for effort. This effort paid off, with the group significantly outperforming the others in test adequacy analysis. It is worth nothing that all groups scored a very low score on the performance scale, which would suggest that participants felt they were successful in this particular task.

2) NASA-TLX Scales by Education: When analysing individual NASA-TLX scales from the perspective of the participants' education, one of the most notable values is the level of frustration experienced by diploma-level holders when carrying out the test adequacy analysis task. Being the most pronounced component in this task it called for particular attention, when one considers that all other scales for the same task scored similar values to other cohorts. This outcome may be due to a lack of training in this particular technique at diploma level.

One also notes that first-degree graduates tended to be exposed more than other cohorts, scoring an average of 60 in every task category except test case design. Postgraduates had similar temporal demand readings in test automation and bug-finding tasks. Unlike the other two groups, diploma holders tended rarely to experience any significant temporal demand.

V. Discussion

This section seeks to address each of the two research questions defined in Section I, discussing the respective outcomes on the basis of the results presented in Section IV.

A. Effect of Types of Testing Task (RQ1)

The discussion of RQ1 revolves around effectiveness and cognitive load.

1) Task Effectiveness: The results indicate that participant effectiveness decreased when we increased the difficulty level of tasks in both the test design category and the test automation category. However, the pattern of diminished effectiveness differed substantially between the two categories. In test design tasks, the number of participants who were completely unsuccessful in their attempts, increased from 7% to 27% to 40% for each successive task and difficulty increment. More encouragingly, the percentage for the test automation category remained constant at 7%. A closer look at our raw data revealed that the failing participant was a different person in each test automation task, leading us to conclude that the failure may be due to lack of familiarity with the specific test automation technique being used in that task. In contrast, the participants who failed the second test design task, also failed the third one, and were joined by three new participants who were similarly unsuccessful in the task. This suggests that test design tasks tend to be more cohesive in nature than their automation counterparts.

The number of partially successful candidates in the test design tasks remained relatively constant from one task to the other (33%, 27%, 27%). We have interpreted this to suggest that, with test case design, participants either know a technique or they do not, with little room for a middle ground. On the other hand, test automation tasks saw the partially successful range going from 13% to 47% to 93%, but with no candidates completing the task successfully. This would suggest that there are multiple ways in which to carry out the same test automation task and that the nature of test automation would allow a wider margin of error without resulting in complete failure.

The bug-finding task was not based on any specific technique but relied on the participants' level of observation and ability to detect anomalies. More than half the participants (53%) managed to find all 10 bugs on the screenshot, while 27% found at least 8 (which was our boundary for a partially successful rating).

Finally, participants found test adequacy analysis, the most challenging task of all with 47% failing the task completely and only 27% completing it successfully.

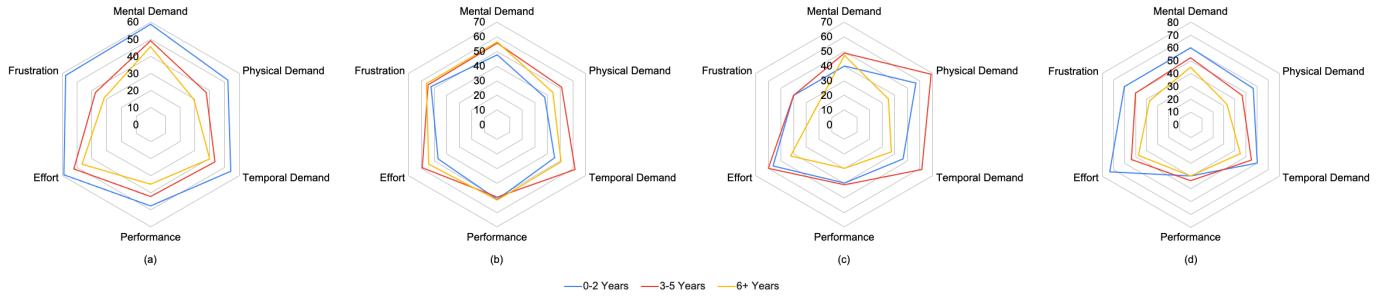

Fig. 1. NASA-TLX scales by experience for (a) test design, (b) test automation, (c) bug finding, and (d) adequacy analysis

2) Cognitive Workload: When analysing the NASA-TLX scores across all participants, we observed that both the test design tasks and the bug-finding task had a mean score of 47. However, there were differences in the distribution of the scores. Whereas interquartile scores (middle 50% of participants) for test design tasks were compacted between 34 and 60, the interquartile range for bug finding ranged from 26 to 74. This suggests that test design tasks generate a more consistent cognitive load than bug finding, which tends to be more varied. Test automation tasks generated a mean cognitive load of 55, with an interquartile range of 39 to 70. Finally, test adequacy analysis generated a mean load of 52 with an interquartile range of 40 to 63. This makes test adequacy the most compactly distributed task category.

In the NASA-TLX scales (see Table I) the type of task had minimal effect on mean mental demand, temporal demand and effort. The exception to this was that the test design had a significantly lower value (45) than the cluster of the other three categories (56, 52, 53). However, performance and frustration were significantly affected by the type of task. The performance scale indicated that participants were most confident with test design and bug finding, but less so with adequacy analysis and test automation. It is worth noting that the maximum value of 50 was nowhere near the higher end of the scale's bounds, thus indicating that the participants were relatively confident in their performance, even if the actual results appeared to point in a different direction.

Frustration also exhibited a certain variability based on the type of task being carried out. Participants found test automation to be the most frustrating category with a score of 54. This was followed by adequacy analysis (49), test design (40) and bug finding (35).

B. Effect of Experience and Education (RQ2)

As discussed in Section IV, the participants' experience in the field had an impact on both their effectiveness and cognitive workload. However, whilst education did demonstrate some variability in cognitive workloads, it had a negligible impact on task effectiveness.

1) Task Effectiveness: The idea that experience would have an effect on task effectiveness was arguably an expected outcome of this work. However, we were surprised to observe that this impact was not always positive with the less experienced participants outperforming more experienced ones in test design and adequacy analysis tasks. This may be due to a combination of two factors. Firstly, less experienced candidates would have just recently been trained in the methodological aspects of testing required by these two types of tasks. Secondly, in our conversations with the participants collectively working across a spectrum of studies over these past years, we have observed that testers tend to eventually settle into a preferred role or specialisation. For example, one might specialise as a test engineer, a test analyst or a test lead. Each of these specialisations would result in certain skills being given less attention in favour of others, over time.

The opposite held true for test automation and bug finding. An interesting point, here, was that the increased effectiveness in test automation is not impressive, in that it ranged from 89% to 92% to 100% as the level of experience increased. One could argue that an 89% success rate is actually to be expected. However, the differences in bug finding were much more pronounced, ranging from 67% to 79% to 100%. We believe that bug finding is one skill that benefits more from a trained eye, developed through experience and practice, as opposed to a technique that could be applied methodologically.

2) Cognitive Workload: The highest mean levels of cognitive workload were exhibited in test automation tasks for candidates with 3-5 years of experience (57) and undergraduates (59). The interquartile range for 3-5 years' experience ranged from 40 to 75, whereas that of the undergraduate cohort ranged from 41 to 78. The means were at 63 and 57 respectively, indicating that experience produced a wider distribution of NASA-TLX scores. In both cases, the scores seemed to be driven by all subscales concurrently, with no specific subscale providing a disproportional influence.

As regards test design tasks, cognitive load decreased as experience increased, going from a mean of 57 (0-2 years) to 47 (3-5 years) to 41 (6+ years). The influence of education in this category was less pronounced and moved in the opposite direction, with means of 43, 46, and 51 for diploma holders, first-degree holders and postgraduates respectively.

Bug finding inflicted the least cognitive demands on both the least educated (34) and the least experienced (42). Among the least-qualified participants, this was driven by low levels of mental demand, frustration, effort and temporal demand, whereas in the least experienced the higher mean was driven

TABLE I
MEAN VALUES FOR NASA-TLX SCALES

	Mental Demand	Physical Demand	Temporal Demand	Performance	Effort	Frustration
Test Design	50	38	45	31	52	40
Test Automation	54	47	56	50	55	54
Bug Finding	47	57	52	38	55	35
Adequacy Analysis	52	45	53	41	56	49

by higher levels of effort (57 vs 35) and temporal demand (47 vs 30). This suggested that a lack of experience generates the need for more effort and concentration than does a lack of education.

C. Threats to Validity

This work is subject to the same threat to external validity as other experiments, in that it is a single experiment. Although we have presented some of the results on the basis of empirical analysis, we cannot claim that these results are representative of the whole testing population. Nevertheless, we are confident that they provide a useful insight and form a foundation for further study. We also mitigated internal validity risks through rigorous experimental procedure and utilising NASA-TLX, which has been used successfully in countless studies.

VI. CONCLUSION AND FUTURE WORK

In this paper, we set out to shed light on the cognitive workload experienced by testers from different cohorts as they attempted to complete a range of tasks typical of the field. Our results uncovered interesting patterns in effectiveness based on the type of task alone. They also indicated that, although experience is a key influence on successful task completion, success is also conditioned by task type. Moreover, more experienced persons tended to fare worse than their less experienced counterparts in certain tasks (test case design and adequacy analysis). Level of education had no significant bearing on successful task completion but differences in cognitive workload could be observed for both experience and education level variables. Here too, it was experience that exerted the strongest influence.

Throughout the course of analysing our data and writing the paper, we have identified a number of shortcomings, which would be addressed as part of our future work.

A. Future Work

This study lays the foundations for a number of opportunities for further exploration. Firstly, it would be useful to observe a better balance of demographic properties, such as a much wider representation of the female population. Moreover, a more balanced sample of education level and experience would be similarly highly desirable. More varied cohorts would shed light on whether the results presented here do indeed hold for a wider population. In addition, it would be beneficial to refine the experimental protocol to balance out the number of tasks within each category and provide the space for more qualitative data through follow-up discussions. This would make it possible to elaborate upon mere numbers, and gain deeper insight into, for example, why the more experienced persons tended to fare worse than their less experienced counterparts in certain tasks.

Once the data would have been sufficiently replicated, we would be in a better position to apply our observations to producing guidelines for companies regarding the management of software testers. At present, the data presented here could be used to inform recruitment decisions, team composition decisions, project management, training paths and promotion ladders in the field of software testing.

REFERENCES

[1] H. Sneed, "Program comprehension for the purpose of testing," in *Proceedings. 12th IEEE International Workshop on Program Comprehension, 2004.*, 2004, pp. 162–171.

[2] A. Vanitha and K. Alagarsamy, "Software testing in cloud platform: A survey," 04 2019.

[3] A. Bertolino, "Software testing research: Achievements, challenges, dreams," 06 2007, pp. 85 – 103.

[4] M. S. Young, K. A. Brookhuis, C. D. Wickens, and P. A. Hancock, "State of science: mental workload in ergonomics," *Ergonomics*, vol. 58, no. 1, pp. 1–17, 2015.

[5] E. Galy, J. Paxion, and C. Berthelon, "Measuring mental workload with the nasa-tlx needs to examine each dimension rather than relying on the global score: an example with driving," *Ergonomics*, vol. 61, no. 4, pp. 517–527, 2018.

[6] J. Q. Young, R. M. Wachter, O. Ten Cate, P. S. O'Sullivan, and D. M. Irby, "Advancing the next generation of handover research and practice with cognitive load theory," *BMJ quality & safety*, vol. 25, no. 2, pp. 66–70, 2016.

[7] G. B. Reid and T. E. Nygren, "The subjective workload assessment technique: A scaling procedure for measuring mental workload," in *Advances in psychology*. Elsevier, 1988, vol. 52, pp. 185–218.

[8] S. G. Hart and L. E. Staveland, "Development of nasa-tlx (task load index): Results of empirical and theoretical research," in *Advances in psychology*. Elsevier, 1988, vol. 52, pp. 139–183.

[9] S. G. Hart, "Nasa-task load index (nasa-tlx); 20 years later," *Proceedings of the Human Factors and Ergonomics Society Annual Meeting*, vol. 50, no. 9, pp. 904–908, 2006. [Online]. Available: https://doi.org/10.1177/154193120605000909

[10] H. Mansikka, K. Virtanen, and D. Harris, "Comparison of nasa-tlx scale, modified cooper–harper scale and mean inter-beat interval as measures of pilot mental workload during simulated flight tasks," *Ergonomics*, vol. 62, no. 2, pp. 246–254, 2019, pMID: 29708054. [Online]. Available: https://doi.org/10.1080/00140139.2018.1471159

[11] K. Hrabovská, B. Rossi, and T. Pitner, "Software testing process models benefits & drawbacks: a systematic literature review," *arXiv preprint arXiv:1901.01450*, 2019.