Virksomhedsprojekt 2023-2024

Lundin, Phillip
kxg220
Seoud, Mahmood
tbc115

Abstract—In our startup venture, we've crafted two interactive learning tools for the Computer Systems course with 200+ students. We set a hypothesis that these learning tools enhance the understanding of caching concepts and virtual memory address translation better than without them. Implementation involved e-learning tools with interactive exercises and robust facit-checking, evaluated through direct and anonymous student feedback. This paper presents survey outcomes on student satisfaction and preferences, assessing usability for potential integration into other courses. The study offers insights into future learning tool design, prioritizing a student-centric approach to optimize the learning experience. Notably, we've developed these learning tools with reusability in mind, ensuring adaptability for future implementations in diverse contexts, such as other courses or exercises.

I. Introduction

The project in Practice (University, 2023b) during this paper will concern our idea SysMentor which introduces a set of interactive tools and simulations suited for Computer Systemer (University, 2023a) students that make it visually appealing and engaging to learn. The project is facilitated as a course for the Department of Computer Science DIKU - Copenhagen of University with Troels Henriksen as Supervisor of the project and Jakub Rubinowski as course instructor of the Project in Practice.

We will take a look at one of the tools from Oleks Shturmov (Shturmov, 2015) that supports conventional learning and reflect upon the things we found especially helpful.

As an addition to these tools, we have recently introduced the interactive exercise Virtual memory address translation (VMAT) (Lundin & Seoud, 2023c) and visual simulator Cache (Lundin & Seoud, 2023a) to enhance the learning of these particular exercises for the exam.

The first tool, an interactive learning resource, is directly tailored to a specific exam exercise or question. Students actively engage with the tool by inputting their answers and receiving instant feedback to ascertain the correctness of their responses. The second tool falls within the category of visual simulation tools, dedicated to aiding students in visualizing the underlying concepts of the cache. Unlike the direct integration with exam exercises, this tool provides a visual simulation environment where users can input their answers, allowing them to assess and confirm the correctness of their responses. These tools, whether directly linked to exam exercises or serving as visual simulations, share a common feature: the ability for users to input their answers and receive feedback. In instances where users are unsure of the correct answer, they also provide a mechanism to insert a solution. That is why, we hypothesize that students improve in VMAT and Cache exercises when introduced to these tools that interactively assess their knowledge or simulate a concept.

> The vision is to create a platform with exercises and interactive diagrams suited for CompSys students that make it visually appealing and engaging to learn.

Fig. 1: SysMentor's vision

The primary goal of introducing these tools is to establish an interactive learning system, motivating students to actively assess their comprehension of the course material. The second reason is connected to exam preparation. Historically, there has been a recap session at the end of the course where exam exercises are thoroughly reviewed. However, when speaking with students, they indicated this approach to be too late and too stressful for effectively grasping the concepts. (Appendix I) The online environment we have provided allows students to commence their learning earlier, avoiding last-minute cramming in the final week of the course.

We have made use of both interviews and surveys to analyze student satisfaction with introducing VMAT and Cache regarding their respective assignment and exercises for the exam. This is followed up by a discussion and analysis of the feedback from the students concerning the most optimal way of learning for the students regarding Computer systems.

The rest of the paper is organized as follows: Section II explores the reasons behind the relevance of online tools in greater detail. Section III delves into the Computer systems course. In Sections IV, we provide details on the development process of our ideas and the tools, including the testing phase. The following Section V showcases the survey results and students' feedback. Section VI conducts an analysis of the survey results, examining correlations between students' satisfaction and different learning tools. Lastly, Section VII wraps up our work.

II. Background

A. Why online tools?

For almost a decade, the Machine Architecture (ARK) course at DIKU has been providing students a variety of

helpful online interactive tools [1]. Notably, Oleks Shturmov (Shturmov, 2014) played a key role for us in developing some of these tools.

A study focusing on student satisfaction with e-learning tools for the Computer Architecture and Organization course highlights the positive impact of such tools. The research emphasizes how interactive elements enhance the learning process, promoting student engagement and comprehension (**Student-satisfaction**). Moreover, our real-world experiences, (detailed in Appendix E) involved studying how an interactive dice game tool could enhance the understanding of probability. This emphasizes the significance of interactive tools, particularly those accessible online. Such tools not only facilitate usage at the user's convenience, but also eliminate the need for downloads, making them accessible from any location. Shturmov's tools provided two particularly valuable features. The instant feedback proved to be instrumental, allowing for swift identification of patterns and understanding of the underlying concepts after completing some exercises.

Another noteworthy feature was the log of completed exercises. This log not only showcased the student's attempts but also presented the correct answers. This feature grants students the opportunity to reflect on the reasons behind their incorrect answers. It allows them to backtrack through their steps, pinpointing where errors occurred and providing a chance to rectify those mistakes. These features contributed significantly to our tools. A significant motivation stems from the statistic that a substantial portion (25%) ("Karakterstatistik for kurser og projekter", 2022) of students at DIKU experience failure in the Computer Systems (CompSys) exam. we sought to contribute to a more favorable exam outcome for students by implementing tools that could enhance their learning.

Many students encountered difficulty as solutions to exercises were not readily available. By developing tools that provide instant feedback and exercise logs, we aimed to improve students' learning with the resources necessary to understand and rectify their mistakes independently.

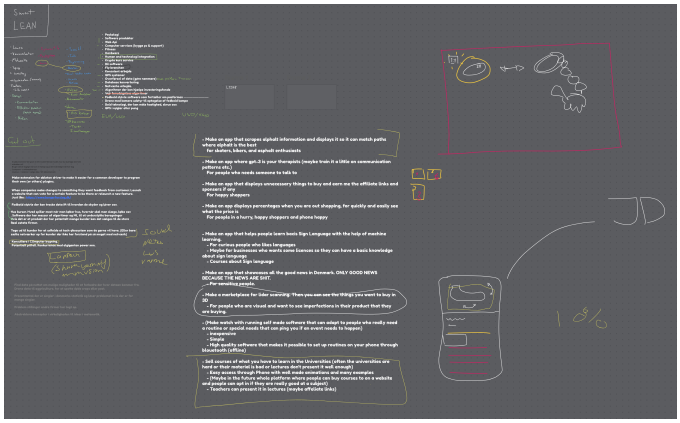
The starting phase of SysMentor has been through the commercialization of the project in practice around campus at DIKU. We have been keen on starting our own idea as being independent is exciting.

The basis of the choice of SysMentor comes from two things.

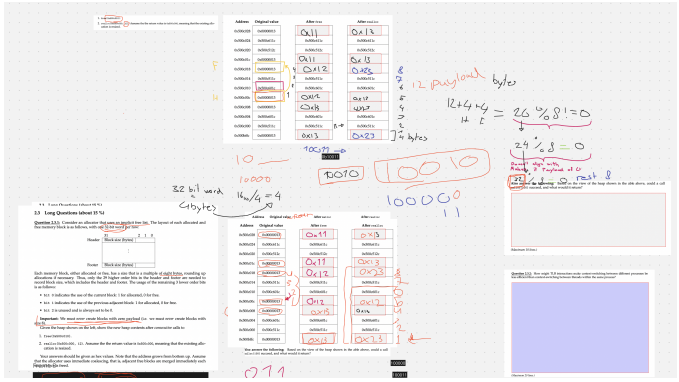
Firstly, a coherent brainstorm of ideas in a shared infinite canvas software program called Miro "Miro", n.d. In this canvas, we have collected our mutual interests, such as psychology, technology, and programming. Hereafter we listed all the ideas for an interesting project to take. We have listed all from computer-building consultancy, and house price prediction models to applications that could teach sign language.

Secondly, as students ourselves, we have participated in the CompSys course and we struggled with the course material. We realized that the exercises presented could be confusing and sometimes there were no answers to the problems.

[1]The course Machine Architecture, along with Operating Systems and Concurrent Programming, and Computer Networks, were combined into a one-semester course called Computer Systems in fall 2016



(a) ideation phase of business ideas



(b) Exercises from Computer Systems

Another key consideration was the inherent difficulty in commencing studies due to overload and ambiguous study materials referenced in chapter V by the interview in Appendix A. To streamline the learning process, we aimed to create tools that offer a cohesive and structured approach, making it easier for students to navigate the material.

The goal of the project is to create a minimum of 2 interactive exercises on a suitable platform, emphasizing virtual memory address translation, cache hit/miss, and heap operations. These exercises should be engaging, informative, and accessible for students at various levels of proficiency.

III. Computer Systems

This section describes the Computer Systems course, by analyzing its topics, learning objectives, and exam.

A. The Course Structure

The course provides an overview of machine architectures, operating systems, and data networks. Targeted at computer science students, it aims to develop a fundamental understanding of concepts within operating systems, networks, data representation and cache and machine architecture. These topics include logical components for building computational units, memory components, and Boolean algebra.

- Theoretical lectures with 2 classes per week;
- Teaching assistant sessions with 2 classes per week;

Category	Hours
Lectures	62
Preparation (estimated)	109
Theoretical exercises	62
Project work	175
Exams	4
Total	412

Fig. 3: Work distribution for Computer Systemer

- One weekly assignment

One lecture accounts for 2 hours and is followed by one exercise class that accounts for 2 hours. This was facilitated every Monday and Wednesday. Every Friday, a small get-together is held where students can come and get help with both assignments and exercises from Teacher assistants. At the end of the course, a recap session is held for students to go through the most prominent topics of the course as a means to practice for the exam.

This course weighs 15 ECTS, which corresponds to 412 hours of total learning time. This time is distributed across one 16-week semester. According to the University of Copenhagen, the lectures consist of approximately 100 students while the exercise classes are carried out in smaller groups of about 25 students.

The Syllabus covers relevant topics of the following according to KU (University, 2023a)

- Number representations, arithmetic, and boolean algebra
- Instruction sets, (symbolic) machine language, processor architecture, and memory hierarchies
- Threads, scheduling, and synchronization
- Processes and virtual memory
- File systems and I/O devices
- Data networks
- Data security

B. Exam

The exam is conducted as a 4-hour written exam. The exam tests the student's competence in processor architecture, storage hierarchies, operating systems, and computer networks. As well as their ability to analyze program performance based on knowledge of processor architecture, storage hierarchies, and operating systems.

It also seeks to determine if they can design and implement the functionality for various operating systems and network components. Students should also be able to reason about the correctness of multithreaded programs, including strategies to avoid race conditions and deadlocks.

Lastly, students are evaluated on their ability to discuss simple security properties for a given system. (University, 2023a).

IV. Methods

A. Mom tests and interview

As we developed our idea, the primary emphasis during the startup's initial phase was to gather information. In the lead-in period, we were presented with tools that aided our understanding of the underlying learning challenges associated with the course. One such tool is the mom-test, originally crafted by Rob Fitzpatrick to foster in-depth conversations (Duffy, 2018). The mom-test involves conducting interviews where participants are unaware that they are being interviewed, providing genuine responses with minimal bias or preconceptions. We applied these tests with participants from Copenhagen University, Copenhagen Business School, and the IT-University, extending the reach of the mom tests across diverse disciplinary domains.

Another tool from the lead-in period was the utilization of interviews. Before developing these tools, we conducted a targeted interview, choosing a teaching assistant (TA) at Computer Systems as the interviewee, believing it would provide deeper insights compared to a regular student. This TA signed a declaration of consent and agreed to its recording beforehand. This declaration of consent can be found in Appendix D (fig. 14). Down below we have listed some of the focus areas we talked about in the interview. The full

- Oplevelser med kursusundervisningen (Experiences with courselectures)
- Kontekst (Context)
- IT-System
- Interaktion mellem underviserne og Elever (Interaction between the teacher and pupils)

Fig. 4: Selection of focus areas

interviewing focus areas from this interview can be found in Appendix F.

The insights gained from both the mom-tests and interviews enabled us to create two frameworks: a Value Proposition Canvas (VPC) (fig. 19) and a Sustainable Business Model Canvas (SBMC) (fig. 20) (Appendix H). The VPC framework aids in understanding and visualizing the value of a product or service, while the SBMC helps articulate the overall impact of the product or service (Duffy, 2018).

On top of the interview, we did surveys for both VMAT and Cache individually after having a handful of users try them out (fig. 5)

1) "How many points do you think you can achieve in the exams VMAT/Cache?"
2) "Was there anything you especially liked or disliked about the tool?"
3) "Did you find the facit button useful"

Fig. 5: Survey questions post using the VMAT/Cache tool

The interviews and the surveys conducted provided insights into the experiences of students. Some of the feedback received has been implemented into the tools, while certain aspects are earmarked for future work. The specifics of these implementations will be covered in the discussion VI. It's important to emphasize that student feedback played a significant role throughout the development and iteration phases.

B. Exercise Development

Our software development strategy focus was on how we can reuse components. This means we have to figure out which parts can be used by more tools (generic) and which parts need to be specially made for a particular task (specific).

Let's take the example of facit checking in our software applications. This is a generic feature because, no matter which exercise we are working on, checking user input against the correct solution is always important. This feature makes sure we follow the same process for evaluating user interactions in different software tools. By having a identical way of doing facit checking, we make the validation process more efficient and save time on coding. this can be seen in fig. 6

```
function deepEqual(object1: any, object2: any): boolean {
    const keys1 = Object.keys(object1);
    const keys2 = Object.keys(object2);

    if (keys1.length !== keys2.length) {
        return false;
    }

    for (const key of keys1) {
        const val1: any = object1[key];
        const val2: any = object2[key];
        const areObjects = isObject(val1) && isObject(val2);
        if (
            areObjects && !deepEqual(val1, val2) ||
            !areObjects && val1 !== val2
        ) {
            return false;
        }
    }

    return true;
}

function isObject(object: InputFields): boolean {
    return object != null && typeof object === 'object';
}
```

Fig. 6: Code For deepEqual

In Javascript, non-primitive values are compared to their references, thus there was a need for the function `deepEqual` that takes in two arguments the facit and the users input, returning if they are equal.

```
    function DiscreteSliderValues(props: DiscreteSliderValuesProps) {
    const defaultValue = props
        .marks
        .find((mark) => mark.label === props.value.toString())?.value
    return (
        <Box sx={{ width: 1 / 2 }}>

            <Slider
                defaultValue={defaultValue}
                getAriaValueText={valuetext}
                step={null}
                style={{ width: '203px' }}
                onChange={(_e, value): any => props.handleSliderChange(value)}
                marks={props.marks}
            />
        </Box>
    );
    }
```

Fig. 7: Code For DiscreteSliderValues

Another generic feature is seen in fig. 7. This modification was implemented by leveraging the state objects within the React framework. This approach streamlined the process of adjusting settings as it involved updating a single variable. To facilitate this, we designed a component called `DiscreteSliderValues`, which requires `marks` (a list of marks) and `handleSliderChange` (a function to set the state) as properties.

With this component in place, whenever the slider is adjusted, we can easily set it to the specific mark chosen at that moment. This versatile component is employed for adjusting settings related to sets, ways, and the size of the random address.

In contrast to generic components, specific feature that can significantly differ from tool to tool is the implementation of tables visible to the user. For instance, the TLB tables are vastly different from the Cache tables regarding their shapes. In TLB tables, the lines grow horizontally whereas in Cache they grow vertically. This can be seen in fig. 8

Set	Tag	PPN	Valid	Tag	PPN	Valid
0	0×2B2	0xE3	1	0×14D	0xE8	1
1	0×61	0xC4	0	0×2D	0xC4	0
2	0×443	0xF8	1	0×512	0xC0	0
3	0×172	0xF3	0	0×171	0xB6	0

(a) TLB table.

	Valid	Tag	Block
Set 0	1	0000001101	Mem[208-215]
	1	0000000100	Mem[64-71]

(b) Cache table.

Fig. 8: TLB tables and Cache tables

In our software development strategy, we aim for a balance between generic and specific parts.

1) Virtual Memory Address Translation (VMAT): The main features and user interface of the VMAT tool are presented here https://abdsecondhand.site/VMAT/dist/index.html.
This web-based interactive exercise tool draws heavy inspiration from previous years' Computer systems and is developed using React, a JavaScript framework with Typescript.

The primary purpose of the VMAT tool is to facilitate interactive exercises for practicing Virtual Memory Address translations. It aims to imitate technology methods, including TLB hit, Page hit, and Page fault.

The tool offers features such as the ability to highlight bits for a better overview, conduct hypothesis testing, and more.

To highlight bits with a specific color, simply click and drag them. Insert the determined VPN, TLB index, TLB tag, etc. in the form. If inserted correctly they will notify you and toast you! "Insert facit" does what it says, inserts the facit at the given input field.

Virtual memory, as a concept, serves as an essential abstraction in modern computer systems. It's not a tangible, physical entity but rather a strategic approach to overcoming the limitations of finite physical memory. When a process attempts to access a virtual address, whether for loading or storing data, the virtual address is transformed through Virtual Memory Address Translation (VMAT).

In this process, the virtual address is mapped to its corresponding physical address, making it real and tangible for the computer's hardware. It allows processes to operate under the assumption of having more memory than physically exists.

Fig. 9: Snapshot of VMAT tool

When developing VMAT, we drew significant inspiration from the exam structure, which typically includes three VMAT-like exercises. You can find an example of such an exam in Appendix J.

The goal was to ensure that the exercise we created would be familiar to students when they encountered it during the exam. In the standard exam format, three cases are consistently presented: TLB hit, Page hit, and Page fault. However, the exam may vary from year to year by altering the size of TLB sets or other parameters.

To address these variations, our tool needed to be adaptable. This brings us to the settings component.

Within the settings, users can adjust factors such as the number of sets and ways. Additionally, users can practice specific scenarios, such as TLB Hit, Page Hit, or Page Fault, if they feel the need. This allows users to customize the VMAT exercise according to their preferences.

Another customizable aspect is the difficulty of the exercise, represented by the size of the address. This setting, too, can be adjusted within the settings. We achieved this by modifying the range of the random number generated for the address. For specific details on these settings, refer to Fig. 10.

2) Cache hit/miss: This section is about cache management in computer systems, specifically focusing on cache hits and misses. The goal here is to bridge the gap between theory and hands-on experience, giving students a solid grasp. This section seeks to shed light on the process of creating such interactive exercises that effectively demonstrate and reinforce the principles of a Cache. We will also reflect on how we can create random assignments, without them being too random or difficult.

The user interface is presented here https://abdsecondhand.site/CACHE/dist/index.html This interface has two main features: Cache hit/miss and cache simulation. You can switch between these features

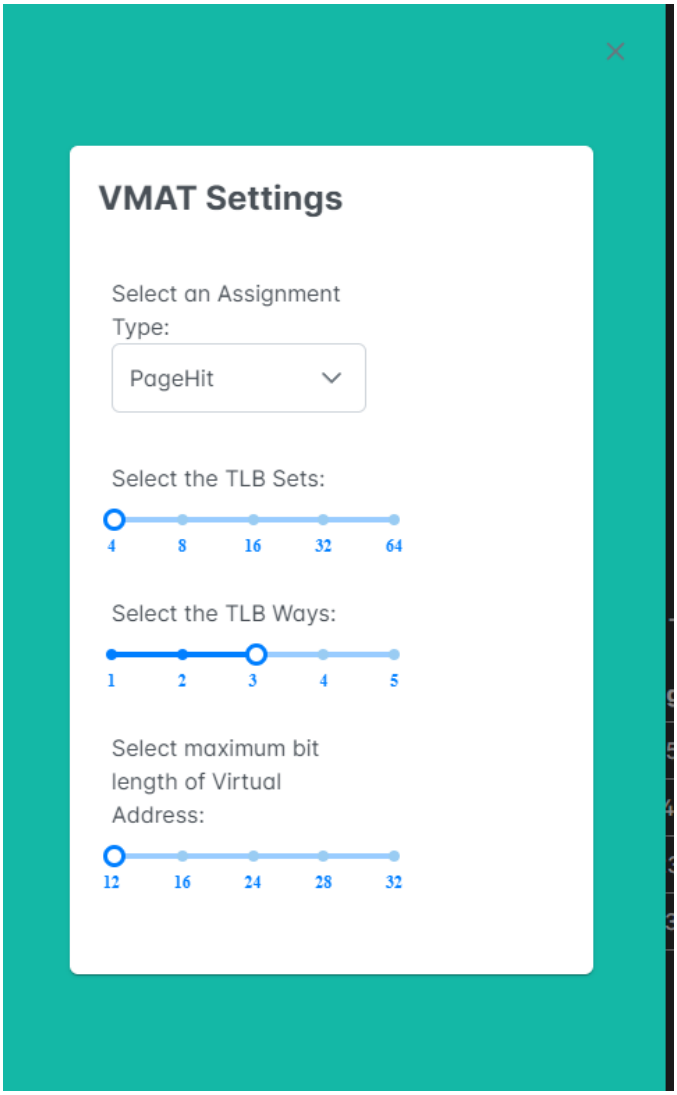


Fig. 10: Virtual Memory Address Translation settings

using the two buttons labeled ”guess” and ”input” located below the bit address. The first state represents the Cache Hit/Miss State - Here, your task is to determine whether the current address is a cache hit or a cache miss based on the provided cache information below. If a cache miss occurs, the system will automatically update the cache line. You can hover your mouse over the cache to see which line was updated after the last miss. The second state represents the Cache Imitation State. In this state, your goal is to replicate the cache based on a given address like in the first state. If it is a miss, you need to input the tag and block allocation manually.

```
export interface Cache {
  numSets: number;    // The number of sets in the cache
  blockSize: number;
  linesPerSet: number;
  sets: CacheSet[];   // The sets in the cache
}
  function generateRandomAssignment(cache: Cache, probability: number): string {
    if (!isCacheEmpty() && Math.random() <= probability / 100) {
      return createCacheHitAssignment(cache);
    } else {
      return createCacheMissAssigment(cache);
    }
  }
```

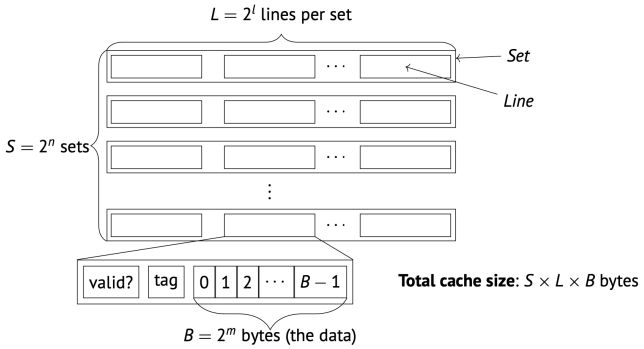


Fig. 11: Representation of the cache, the code mimics

The provided TypeScript code defines an interface called `Cache`, which represents the structure of a cache. It includes properties such as `numSets` (indicating the number of sets in the cache), `blockSize`, `linesPerSet`, and an array of `sets`. The interface is designed to closely resemble the cache representation presented in the slides of the Cache organization and operation chapter ((Henriksen", 2022)).

The code also features a function named `generateRandomAssignment`. This function takes a `Cache` object and a probability parameter. It allows for the creation of random cache assignments, providing control over generating both cache hit and cache miss assignments. The probability parameter determines the likelihood of either assignment occurring. If the cache is empty, the function returns a cache miss assignment.

Cache blocks are organized in a specific pattern. The decision of which address should be stored in a cache block is influenced by the set index and the tag. When a cache block is looked up at a particular set index, and the tag matches while the validity bit is also valid, it results in a cache hit. This means that the required data for the given address can be swiftly retrieved from the cache, bypassing the slower process of accessing the main memory.

On the other hand, if the lookup does not result in a cache hit, it leads to a cache miss. In such cases, the system writes the data to the cache, ensuring that the next time data needs to be retrieved from that address, it is already present in the cache. This proactive caching strategy helps minimize latency by preloading frequently accessed data into the cache, improving overall system performance compared to repeated retrieval from slower memory.

We have decided that concepts like write-through, write-back, write allocate, write no allocate or other caching policies are out of the scope of what we are trying to encompass. We also have decided that Least Recently Used (LRU) storing mechanism does not fit in our scope either.

This cache serves as a simulation tool specifically designed to complement the VMAT (Virtual Memory Address Translation) context, as discussed in IV-B1. Similar to the VMAT exercise, we have implemented configurable settings to fine-tune parameters such as sets, ways, and block size within the cache. This configurability is facilitated through the use of the `DiscreteSliderValues` and `deepEqual` components, elaborated upon in fig. 7 and 6.

In the context of this cache simulation exercise, adjusting the number of sets and lines per set serves as a means to control the difficulty level. By doing so, the total cache size is influenced, and addresses are constrained to be less than or equal to the total cache size. This intentional flexibility allows users to tailor the simulation to their desired difficulty level.

As mentioned, we have incorporated some valuable features from Shturmov's tool (Shturmov, 2015), notably the log. While our implementation may not replicate his tool entirely, it adheres to similar principles. Specifically, when a student completes a task, we log the state of the cache. This log feature allows users to review the evolution of the cache state over time, providing a valuable resource for tracking and understanding the progression of cache interactions.

Additionally, whenever a student answers a question correctly, the tool highlights the specific cache block that is either being read or written to. This visual feedback not only acknowledges the correctness of the response but also provides a real-time illustration of the cache interactions.

Lastly, we recorded the amount of traffic to these tools with the help of Hostinger analytics (Sentika, 2004) and also made a landing page where we retrieved emails (Lundin & Seoud, 2023b) of people who wanted to sign up for more exercises.

V. Results

The development of the tools is showcased on a landing page which can be accessed on https://abdsecondhand.site The code can also be found on github on these repositories:
https://github.com/MahmoodSeoud/VMAT and
https://github.com/MahmoodSeoud/cache-hit-miss

From the mom-tests and interview we were able to conclude the following reoccurring problems of the course.

1) It can be unpleasant to answer questions at the lectures because you can not be 100% sure if it is correct. Often the lectures are structured around asking questions to students.
2) There are not always answers to exercises, resulting in the results you have come up with can not be verified
3) Students are fans of concrete material with examples or live coding or other forms of hands-on means of teaching
4) More slides are often not better
5) The practical exercises are better than the theoretical ones
6) It is first when you work with the material that you gain the best understanding of the study material

We did not put much focus into the analytics part. Trivial data is available on the hosting platform the website is hosted on. As of the time of writing this, it shows around 10 unique IP addresses with a total of 656 requests to the domain. We can see that maybe 5% of the total requests comes from crawlers [2] or bots.

A. Student Assessment

An interview was conducted with 3 students whom were enrolled in the Computer systems course. They were in the

[2]Web crawler bots index web content for search results

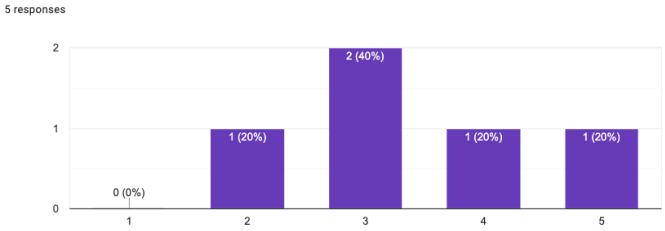


Fig. 12: Confidence in solving the VMAT exercise for the exam

process of going through the recap sessions of the course. We first presented them with an exam exercise of the cache. After some frustration of being unable to solve the exercise, we presented the Cache exercise tool for them and let them play with it. Here's a condensed version of the feedback from the cache interview:

- Users found the tool initially challenging to understand.
- Difficulty using colors for bit allocation and avoiding repetition of colors.
- Lack of clarity on the significance of a green outline after a cache miss.
- Limited utilization of the log feature.
- Users noted excessive scrolling.
- Users eventually completed exercises, gaining an understanding of cache hit/miss and line insertion.
- Building intuition remained challenging.
- Users explored settings to understand the impact on set association.
- One user suggested that the cache tool would be more beneficial if it resembled the exam format.

1) Survey feedback: Based on the survey feedback, participants demonstrated a generally high level of confidence in solving the VMAT exercises. However, there remains room for improvement, as the majority of respondents fell within the 3-4 point range (as seen in fig 12). Interestingly, users appreciated the randomization aspect of the exercises but expressed confusion, emphasizing the need for a proper introduction. Specifically, 5 participants found the facit tool helpful, while one participant refrained from providing feedback on this aspect.

We have identified areas that possess a general structure that every type of exercise would benefit from. Other courses would benefit from these points as well if their exercises and material follow a set of exercises that may have some variables but all have a definite answer like in most exercises in Computer systems.

Settings & Configurability: Having settings that conform to the user's behavior and preferences is ideal, as it will maximize their interest and motivation and give the user a broader understanding of the subject. As for VMAT altering the TLB sets- and ways will give an understanding of how it affects the table and the user can quickly make and test hypotheses. Additionally, if a user, finds specific tasks, like a *Page Fault* exercise hard, they can specifically train that type of exercise

Logging Reflection and improvement are crucial aspects of the learning process, and the cache assignment highlighted a notable challenge. The issue revolves around the ability of users to trace back and learn from their actions, especially when the cache overwrites the information once an answer is submitted. This problem emphasizes the importance of a feature found in the IEEE-754 tool ((Shturmov, 2015)), which allows users to review and learn from their previous interactions.

Input to facit check Interactivity is great for fostering engagement, but it is also essential for users to test their hypotheses. In both the VMAT and Cache we have systems that check for the input from users and reference it to a facit. This provides instant feedback to users.

Randomness in terms of seeding Exercises that are fit for automation, will benefit from randomness to maintain the generation of interesting exercises. Another aspect of randomness is controlling the randomness with seeding as users will be able to share or retrain specific exercises.

VI. Discussion

A. Reflection on Tool Development and Usability Challenges

The library React (Walke, May 29-2013) was a good fit for creating these exercises since it accomplished developing tools on a website and was quickly able to launch it on the web for users to try. Additionally, we were adamant about having the flexibility that React offers with its components, which are rendered individually. This is especially important because its dynamic will make it easy to have components that can be used in any course or exercise that follows a general systematic methodology. While the general methodologies (V-A1) provide valuable insights into what should be incorporated in an exercise to enhance learning, we can not deny the fact that users faced confusion regarding the usability of the product. This confusion stemmed from a lack of introductory guidance, hidden features, and a substantial amount of scrolling required to navigate through exercises. An improvement to this list could include the addition of an introduction/tutorial in the form of a short video and, ideally, enhancements to the user experience (UX).

B. Customer segments

At some point, we were perplexed about which customer segment to focus on. Our initial discussions revolved around the problem highlighted in section V: "More slides are often not better." We observed through the interviews, that teachers often used a plethora of slides in their best effort to convey information using familiar presentation software. Which made us consider creating a tool to assist instructors and professors in formulating their material. The idea was to develop an easy-to-use software program capable of quickly creating diagrams, inspired by a concept found on the website (n.d.). We quickly abandoned this approach, again considering that professors might not have the time to learn a new tool, as their primary focus is facilitating lectures. This prompted us to have a more student-focused approach.

In conducting the mom-tests it posed a significant challenge as it proved difficult to pinpoint exact issues due to the diverse learning styles among individuals. An illustrative example emerged during interviews and several mom-tests: some participants felt overwhelmed by the volume of slides and lectures (see Appendix B3 and B4), while others demonstrated ease in dissecting the material and slides to identify key concepts (see Appendix B1 and B2).

Upon focusing our interviews and mom-tests on key aspects, a prevailing observation was that the course lectures often revolved around numerous questions. This approach generated apprehension among students, who were concerned about the potential stigma of answering questions incorrectly in class. Additionally, a substantial portion of the lectures (depending on the course) centered around posing questions. On the contrary, participants expressed a strong preference for lectures that incorporated concrete examples to underscore essential points in the material. Striking the right balance poses a challenge for educators, as too many slides may lead to information overload, given the sheer volume of content presented at once.

Notably, exercise classes were highlighted as effective, as individuals found that they gained a deeper understanding of the material when allowed to engage with it on a personal level.

C. Evolution of Exams and Balancing Technological Stability

The longevity of the tools, both Cache and VMAT, compiled from Typescript and React into HTML, JavaScript, and CSS, is indeed contingent on the stability and compatibility of the underlying technologies. Given the continuous evolution of these technologies and the potential for updates in dependencies or React itself, there is a risk that the compilation may face challenges in the future.

One preventive measure is to minimize dependencies or consider developing the tools using pure HTML, CSS, and JavaScript, similar to the IEEE tool (Shturmov, 2015), which has demonstrated lasting utility over almost a decade.

The question of whether to transition to a more interactive, digitalized exam format is a valid one. While such a shift could offer numerous benefits, including enhanced assessment of students' understanding and a more streamlined, digital process, it also requires careful consideration of available resources. The feasibility of implementing a comprehensive, interactive exam format depends on factors such as the budget, time, and expertise available for the course.

D. Exam Relevancy in Contrast to Learning

The interviews, particularly those focused on the cache tool in Appendix I, provided a significant revelation for our team. The discrepancy between the developed cache simulation and the cache in a previous exam highlighted a crucial question: to what extent should we align our tools with the exam format while still ensuring a solid understanding of the core concepts?

The challenge lies in finding a balance between creating a tool that mirrors exam conditions and prioritizing the fundamental understanding of cache principles. While the cache simulation aimed to provide a learning experience rather than replicate exam scenarios, student feedback indicated a preference for tools that closely resemble exam conditions. This raises the broader question of how exams themselves are facilitated VI-C, especially considering the inherent difficulty of simulating the dynamic states of a cache within a static PDF document.

It's acknowledged that students might not always discern the most effective learning approach, especially when influenced by proximity to exams. Striking the right balance is crucial — creating exercises that students find relevant while ensuring they gain valuable insights beyond mere exam preparation.

E. Seeds in exercises

Implementation of controlled randomness is particularly beneficial for instructors. This feature is designed for instructors by allowing them to curate assignments that align specifically with their educational objectives. Instructors can select exercises that are especially relevant to the training needs of their students. For instance, An exercise where one has to watch out for the valid bit in the cache exercise. Additionally, seeding the randomness could serve as analytics, enabling the tracking of student performance on specific exercises. By logging the exercises posing challenges and tracking popular exercises, both instructors and the SysMentor team gain insights into user engagement.

F. Cramming Before the Exam

The cache tool was introduced in a recap session attended by approximately 100 people. However, the observation that the number of unique IP requests is only around 10, including our own, raises some questions. We hypothesize that this discrepancy may be attributed to the cramming nature of studying everything just before the exam, where the cache tool becomes only a small fraction of what students should ideally learn beforehand. Besides, many students already have to take other exams in this time frame.

The explanation of the huge amount of requests compared to the unique IP addresses stems from the VMAT exercise having a refresh functionality implemented, therefore whenever a user creates a new exercise, it will refresh the site, creating a new request each time.

We believe that if we had provided the link to the cache tool during the lectures, especially while students were actively engaged in exercises, we could have fostered a more robust and sustained interest. This approach might have triggered a positive networking effect, where students share the tool with their peers, leading to increased usage. Additionally, distributing the link earlier in the learning process could have allowed individuals more time to explore and understand the benefits of the cache tool.

G. Commercial relevance and realistic outlook

Individuals especially students invest in various courses to improve their understanding of computer science or other courses to trying to excel their career. Users break away from traditional passive learning by being able to test hypotheses.

The commercial relevance lies in the potential demand for our tool, especially if it is tailored to exam preparation. Having the exercises digitalized enables the possibility to always correct or update course material, given that it changes. As of now maximizing the tool's efficiency, it is recommended to use it with the guidance of a mentor. A further improvement would be making a video explaining how this exercise should be solved as an initial step. Students starting anew will not be as confused, as they expressed in the interviews we conducted V-A1.

We estimate approximately half of 29 lectures with exercises as seen in Appendix G can be digitized. This amount of tools presents a significant investment. While the creation of each exercise requires approximately 35 hours, the realistic feasibility depends on factors such as available resources that the university provides. Therefore it is necessary to streamline the content creation process, perhaps through collaborative efforts or utilizing existing such as libraries VI-E with the implementation of the general methodologies that exist in exercise creation in V-A1. Currently, implementing tool-assisted learning for the whole course requires 525 hours of work considering it would take 35 hours per exercise for two developers. Normally 400 hours are allocated for the whole course. If this budget allocated 9% (or 36 hours)[3] for the tools to be developed, it would require 3.15 hours per tool, which is still unrealistic. Alternatively, we could choose to only implement a few of the most important ones in terms of the ones that cause the most problems in learning for students. In this regard, the primary reason for not implementing the heap exercise is that we deemed it necessary to allocate the resources adequately.

VII. Conclusion

In conclusion, our exploration of customer segments led us to prioritize the student learning experience over instructors' adoption challenges. We grappled with the balance between creating tools that mimic exam formats and ensuring students find the exercises meaningful. Future-proofing the tools raises concerns about dependencies, maintenance, and the evolving nature of exams. The incorporation of controlled randomness and analytics holds promise for instructors, enabling tailored assignments and valuable insights into student performance. However, the realistic outlook for creating interactive exercises for CompSys is sparse, as the limitations in time and resources pose a significant challenge.

Participants express a high level of satisfaction with the creation of random tasks and the immediate feedback they receive. This aspect of the learning approach appears to positively impact engagement and learning. It is noteworthy that the ability to generate such tools is not exclusive to Computer Systems but can potentially be implemented in other courses as well.

The scarcity of our data complicates the ability to determine whether participants have truly gained understanding.

[3] A random guestimate to how a resource allocation could take effect.

References

(n.d.).

Duffy, M. S. O. X. (2018). *The iceberg and trhe sustainable entrepreneur*. Fællestrykkeriet Aahus Universitet.

Henriksen", ". (2022). *"memory$_h$ierarchy.pdf"*.

Karakterstatistik for kurser og projekter [https://karakterstatistik.stads.ku.dk/Histogram/NDAB16005E/Wi 2022/B1]. (2022).

Lundin, P., & Seoud, M. (2023a). *Cache* [https://abdsecondhand.site/CACHE/dist/index.html].

Lundin, P., & Seoud, M. (2023b). Sysmentor landing page [https://abdsecondhand.site].

Lundin, P., & Seoud, M. (2023c). *Vmat* [https://abdsecondhand.site/VMAT/dist/index.html].

Miro [https://miro.com/]. (n.d.).

Sentika, A. B. (2004). Hostinger analytics [https://support.hostinger.com/en/articles/5650167-how-to-use-the-analytics-section-on-hpanel].

Shturmov, O. (2014). *Ark15*.

Shturmov, O. (2015). *Ieee-754 game*.

University, C. (2023a). *Computersystemer course material*.

University, C. (2023b). *Project in practice bachelor's degree program* [https://science.ku.dk/studerende/studieordninger/faelles$_s to/Fael del - ENG.pdf/$].

Walke, J. (May 29-2013). *React*.

Appendix

A. Appendix A

Interview with Annonymous student, who completed the computer systems course.

- anon har ikke lyst til at svare på et spørgsmål i plenum ¨fordi jeg ikke er 100 % " og det kommer an på forlæseren fordi der er nogle der er gode, andre der er dårlige
- Anonymt svar ville være godt ifølge anon hvis man kan svare på nogle spørgsmål i forelæsningen.
- anon har det bedst med at prøve selv, hvor han derefter tjekker facit
- Der var ikke noget facit til opgaver i netværk og TA's havde ikke facit.
- Der var virkelig meget stof at læse og mange spørgsmål var meget flyvske under Netværk
- anon mener, at der ikke altid var de nødvendige "tools" til opgaven. Fx så var Valgrind ikke understøttet på M1 Macbooks
- anon synes at det fungere godt når han bruger [Anki](https://apps.ankiweb.net/) til at huske stoffet.
- anon fortæller "Jeg forstod rigtig meget først da jeg læste op til eksamen, men det er først når jeg laver eksamens opgaver, så begyndte jeg at forstå dem"
- Jeg vil slå op i Wikipedia RIGTIG meget i netværksdelen.
- Der var fejl i materialet ret ofte
- Bogen var ikke altid tilstrækkelig
- "Jeg har det som om at forelæsningerne ikke helt styr på hvor meget folk ved om stoffet." siger anon

- Som regel er forelæsningen meget detaljeret, (Så det bliver svært at se hvad der virkelig er vigtigt ift. til eksamenen.)
- "Netværk Compsys og Softwareudvikling der var utrolig mange slides og det var svært at strukturere." fortæller anon

B. Appendix B

Mom tests of different students ranging from Copenhagen business school, Copenhagen University, ITU.

1) Student A: Kollegaen var [22 years old] går på en kandidat uddannelse på ITU

Konklusioner ud fra denne samtale:

- han har brug for at tage sit headset på og fordybe sig i stoffet og med opgaverne. - Han forstår ikke særligt meget til forlæsningerne, men kigger slides igennem efterfølgende alene for at prøve st forstå det - Han læser ikke særligt meget i bøgerne fra studiet - Hvis han ikke kan finde den relevante den relevante info til opgaverne fra slides så spørger han TA's eller chatgpt - Når han spørger chatgpt prøver han tit at få den kun til at give 10

2) Student B: Copenhagen business school, digital management on masters degree. 24 years old, Female.

"Jeg forstår materialet godt, men det er først når jeg sidder med det at jeg forstår det. Når jeg kan bruge min egen viden til at påføre den teori jeg har fået."

"Hvis opgaver var nemme at gå i gang med, vil du så have nemmmere ved at studiere? Ja det ville jeg, det er svært at sætte sig ned og lave noget, og det er også et spørgsmål om jeg gider. Hvis det er nemt, gør det det også nemmere at studere"

"Ambitionsniveau 4 ca. Men kan nemt få 12 i mundtlige fag fordi jeg har nemt ved at forstå det der skal læres."

"Kigger på tidligere noter fra tidligere studerende, kigger på slides og scoper hvad jeg skal lære, derefter går jeg til eksamen og så kan jeg argumenterer mig ud af det" (skriftlige eksamener)

"mikroøkonimi dumpede jeg, men så snart jeg lagde noget arbejde bed i det og påførende formlerne til mit eget viden, så blev det nemmere"

"Mikroøkommi var rigtigt svært, hun gad ikke bruge for meget tid på det. Efter hun dumpede skriftlig og mundtlig eksamen gik det op for hende at hun skulle tage sammen."

3) Student C: anon går på socialrådgiver uddannelsen

Han har fysisk klasse undervisning der. Han synes det er meget nemmere at komme til undervisningen end at skulle læse i en bog hvor han ikke forstår noget.

anon havde sidst problemer med undervisningen da han fik en ny lære i jura, som gennemgik stoffet meget rodet. Hun sprang frem og tilbage i emnerne og pga dette var der kun 1 af hendes elever som bestod alle andre dumpede eksamen.

4) Student D: anon synes friheden på uni kan blive for meget. Han arbejder bedst i små perioder og har svært hed at komme I gang.

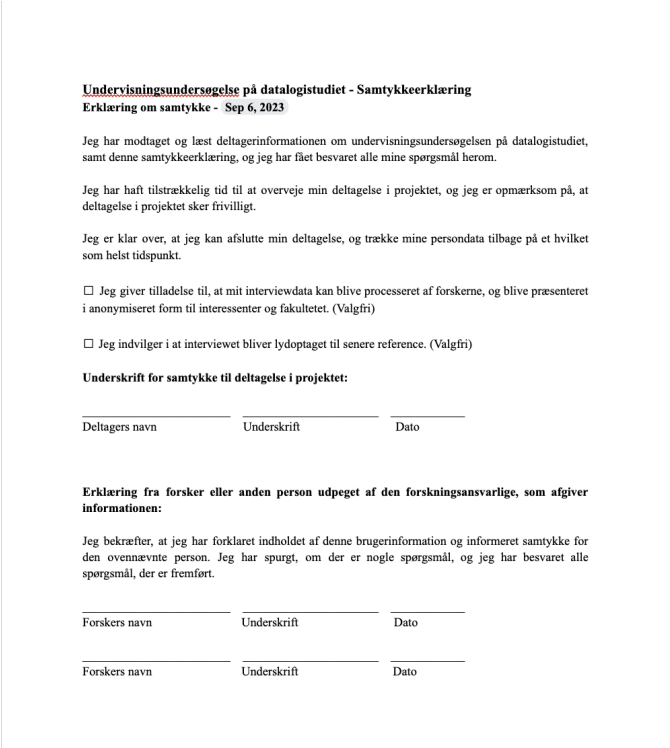
han brugte meget af sin tid i gym på at tænke over hvornår det var færdigt så han kunne få sin hue på og overstået det.

Han var ikke sp glad for at læse og undervisning det var alt for kedeligt.

Fysisk arbejde og rutiner er meget mere ham når det kommer til uddannelse. Han keder sig når han ikke hed hvad han skal lave.

C. Appendix C

Undervisningsundersøgelse



Undervisningsundersøgelse på datalogistudiet - Samtykkeerklæring

Erklæring om samtykke - Sep 6, 2023

Jeg har modtaget og læst deltagerinformationen om undervisningsundersøgelsen på datalogistudiet, samt denne samtykkeerklæring, og jeg har fået besvaret alle mine spørgsmål herom.

Jeg har haft tilstrækkelig tid til at overveje min deltagelse i projektet, og jeg er opmærksom på, at deltagelse i projektet sker frivilligt.

Jeg er klar over, at jeg kan afslutte min deltagelse, og trække mine persondata tilbage på et hvilket som helst tidspunkt.

☐ Jeg giver tilladelse til, at mit interviewdata kan blive processeret af forskerne, og blive præsenteret i anonymiseret form til interessenter og fakultetet. (Valgfri)

☐ Jeg indvilger i at interviewet bliver lydoptaget til senere reference. (Valgfri)

Underskrift for samtykke til deltagelse i projektet:

Deltagers navn | Underskrift | Dato

Erklæring fra forsker eller anden person udpeget af den forskningsansvarlige, som afgiver informationen:

Jeg bekræfter, at jeg har forklaret indholdet af denne brugerinformation og informeret samtykke for den ovennævnte person. Jeg har spurgt, om der er nogle spørgsmål, og jeg har besvaret alle spørgsmål, der er fremført.

Forskers navn | Underskrift | Dato

Forskers navn | Underskrift | Dato

Fig. 13: Undervisningsundersøgelse to the interviewed pupils

D. Appendix D

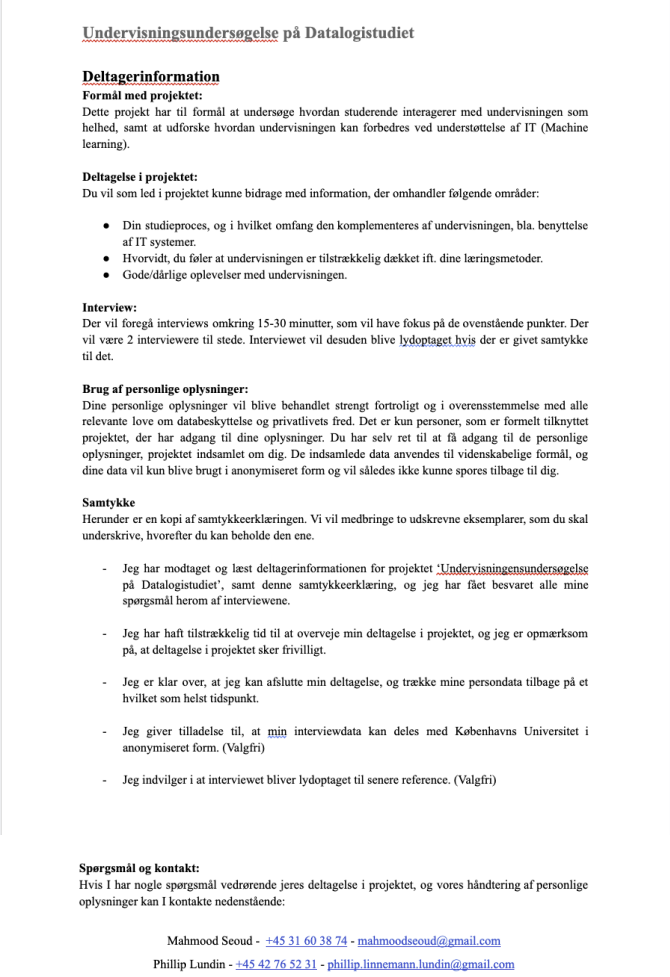
Sammentykkeerklaring

E. Appendix E

The probability iPhone application

	Answer without tool	Answer with tool
Exercise 1	1/36	1/36
Exercise 2	1/6	1/6
Exercise 3	25/36	5/36
Exercise 4	1/6	26/36
Exercise 5	1/4	15/36

TABLE I: Results from iPhone Probability test



Undervisningsundersøgelse på Datalogistudiet

Deltagerinformation

Formål med projektet:
Dette projekt har til formål at undersøge hvordan studerende interagerer med undervisningen som helhed, samt at udforske hvordan undervisningen kan forbedres ved understøttelse af IT (Machine learning).

Deltagelse i projektet:
Du vil som led i projektet kunne bidrage med information, der omhandler følgende områder:

- Din studieproces, og i hvilket omfang den komplementeres af undervisningen, bla. benyttelse af IT systemer.
- Hvorvidt, du føler at undervisningen er tilstrækkelig dækket ift. dine læringsmetoder.
- Gode/dårlige oplevelser med undervisningen.

Interview:
Der vil foregå interviews omkring 15-30 minutter, som vil have fokus på de ovenstående punkter. Der vil være 2 interviewere til stede. Interviewet vil desuden blive lydoptaget hvis der er givet samtykke til det.

Brug af personlige oplysninger:
Dine personlige oplysninger vil blive behandlet strengt fortroligt og i overensstemmelse med alle relevante love om databeskyttelse og privatlivets fred. Det er kun personer, som er formelt tilknyttet projektet, der har adgang til dine oplysninger. Du har selv ret til at få adgang til de personlige oplysninger, projektet indsamlet om dig. De indsamlede data anvendes til videnskabelige formål, og dine data vil kun blive brugt i anonymiseret form og vil således ikke kunne spores tilbage til dig.

Samtykke
Herunder er en kopi af samtykkeerklæringen. Vi vil medbringe to udskrevne eksemplarer, som du skal underskrive, hvorefter du kan beholde den ene.

- Jeg har modtaget og læst deltagerinformationen for projektet 'Undervisningensundersøgelse på Datalogistudiet', samt denne samtykkeerklæring, og jeg har fået besvaret alle mine spørgsmål herom af interviewene.
- Jeg har haft tilstrækkelig tid til at overveje min deltagelse i projektet, og jeg er opmærksom på, at deltagelse i projektet sker frivilligt.
- Jeg er klar over, at jeg kan afslutte min deltagelse, og trække mine persondata tilbage på et hvilket som helst tidspunkt.
- Jeg giver tilladelse til, at min interviewdata kan deles med Københavns Universitet i anonymiseret form. (Valgfri)
- Jeg indvilger i at interviewet bliver lydoptaget til senere reference. (Valgfri)

Spørgsmål og kontakt:
Hvis I har nogle spørgsmål vedrørende jeres deltagelse i projektet, og vores håndtering af personlige oplysninger kan I kontakte nedenstående:

Mahmood Seoud - +45 31 60 38 74 - mahmoodseoud@gmail.com
Phillip Lundin - +45 42 76 52 31 - phillip.linnemann.lundin@gmail.com

Fig. 14: Samtykkeerklæringen for the interviewed pupils

iPhone Probability test:

Takeaways - Thinking more in Numbers and losing count [if we don't have the tool]. Very hard to visualize. - If you have this tool you dont know the behind knowledge. - If questions appear that is not Related to dices it becomes hard to utilize feedback: - Maybe also integrate something so you Can create your own problems. - App needs some explaining on how to use the app.

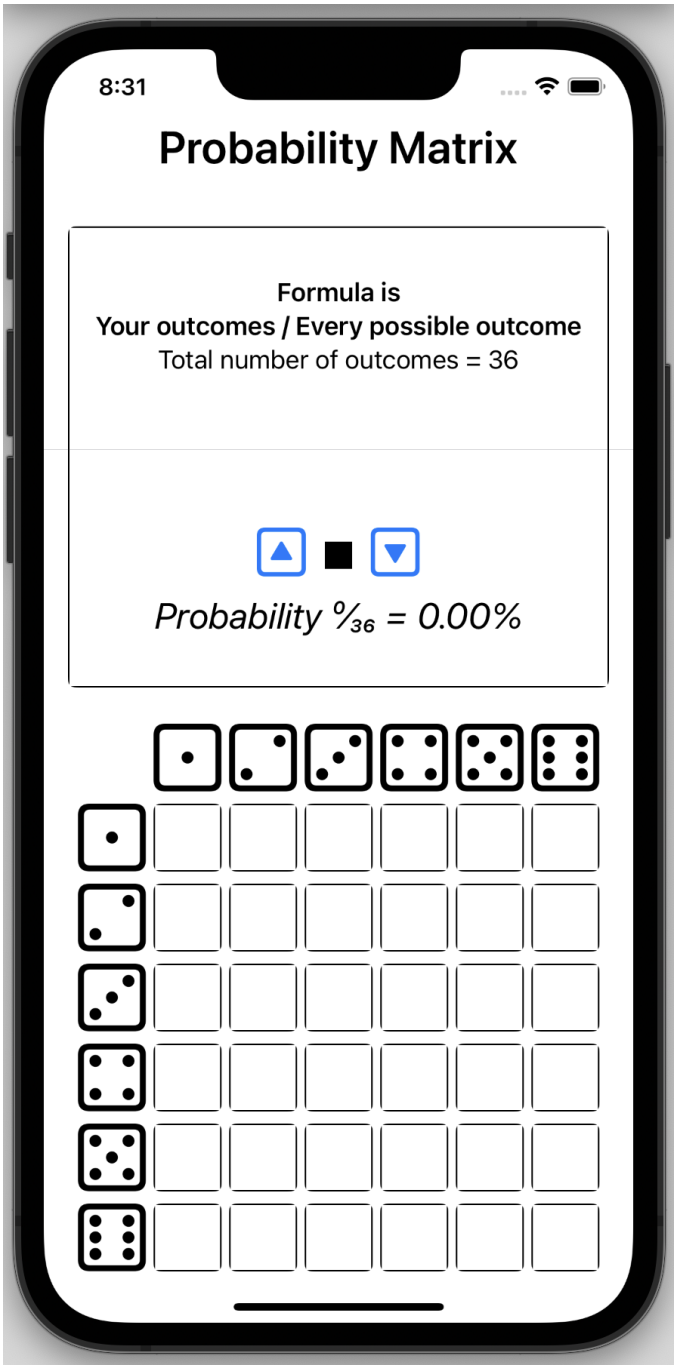


Fig. 15: Application prototype to assess usability methods a

F. Appendix F

Interviewguide fig. 18

G. Appendix G

List of the lectures

- intro_and_c
- assembly_and_machine
- functions_and_text
- computer_arithmetic
- c_pointers
- c_dynamic_memory
- caching
- operating_systems
- virtual_memory_i
- virtual_memory_ii
- concurrency_i
- concurrency_ii
- introduction_to_computer_network
- network_programming
- on-blocking_servers_and_intro_to_security
- http_caching_and_content_distribution
- dns_peer_to_peer_and_udp

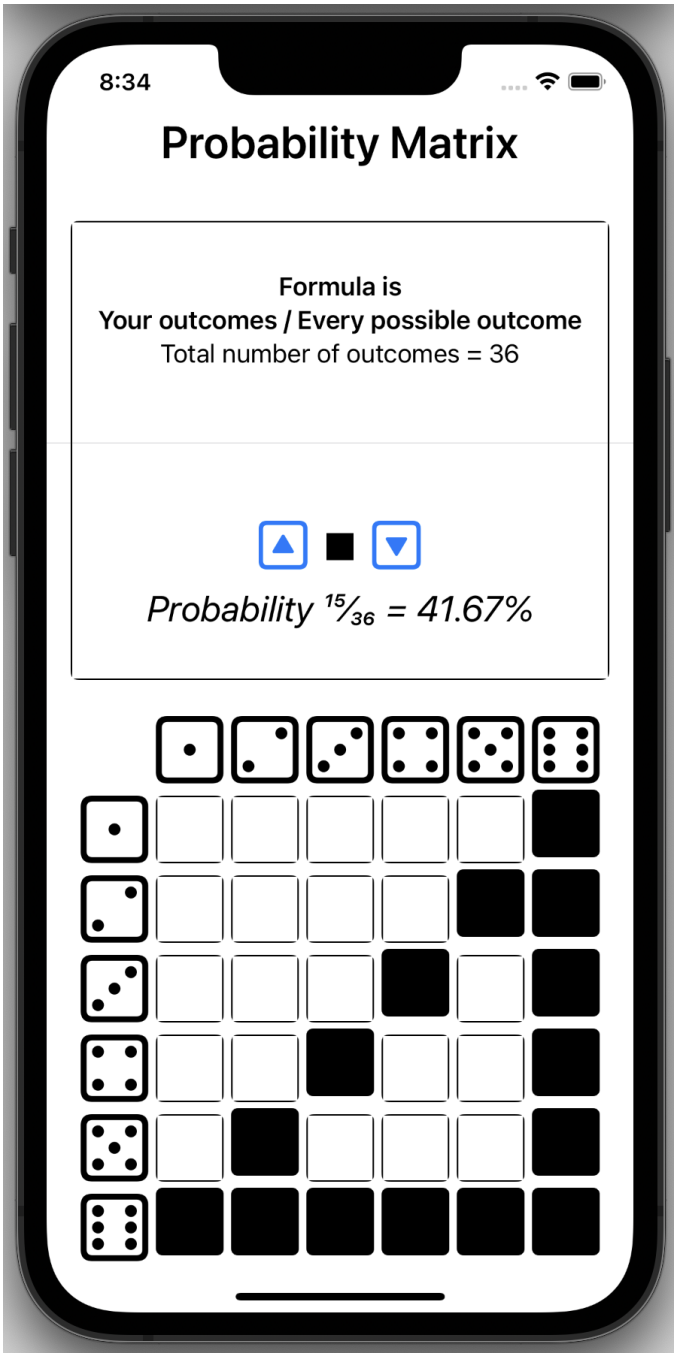


Fig. 16: Application prototype to assess usability methods b

Ex1 What is the probability that you roll 3,6?
Ex2 What is the probability that you roll the same number?
Ex3 What is the probability that you roll a number that is +1 than the previous number
Ex4 What is the probability that you roll more than 5 as the sum ?
Ex5 What the probability that you roll 6 or that the sum of the eyes equal 6?

Fig. 17: Questions for iPhone probability test

Info om brugeren 💡

- Navn
- Definer dit ambitions niveau fra 1-10
- (Hvad er dit gennemsnit?)
- Hvor langt i studiet?

Interview spørgsmål:

Har du nogle spørgsmål i forhold til interviewet?

Oplevelser med kursusundervisningen 📽️

- Hvad er din yndlings-del af kurset? (Compsys)
- Hvad er et et undervisningsbehov, som kursusundervisningen understøtter på en god måde?
  - Definer selv *god*.
- Hvad er et undervisningsbehov, som kursusundervisningen understøtter dårligt?
- (Er der en bestemt del du har haft problemer med?)
- (Hvis der var én ting, du kunne ændre ved kursusundervisningen , hvad ville det så være?)
- (Hvordan føler du, at kursusundervisningen passer ind i dit studieliv?)
  - Er kursusundervisningen ensformigt, eller er der forskelle på, hvordan det fungerer, fra studerende til studerende?
    - Hvilke forskelle?

Kontekst 🏫

- Kunne du beskrive hvordan du følger kursus materialet i løbet af en studiedag?
  - Hvornår gør du dette?
  - Hvordan passer det ind i din rutine?
- Bruger du andre ressourcer eller værktøjer ved siden af kursusundervisningen ? E.g. papir, andre programmer, apps, devices mfl.? (til dit studie)
  - Hvad bruger du?
  - Hvordan bruger du det?
- Hvilke materiale læser du op på hyppigst?

IT-System 💻

- Føler du, at kursusundervisningen er tilstrækkeligt ift. andre lignende systemer, du har brugt? (fx. youtube)
  - Har der været nogle værktøjer, som du føler har været med til at give dig en bedre forståelse for et specifikt emne (i undervisningen) , hvilke?
  - Andre fra nettet (uden for undervisningen), hvilke?
- Hvilke systemer vil du ønske kursus underviseren havde inkoorporeret?
  - Har du set nogle som, gjorde undervisningen interessant, eller lettere at forstå?

Interaktion mellem underviserne og Elever 🤝

- Føler du, at kursus underviseren har gjort sig overvejelser om hvordan præcis *din* rolle får den bedst mulige oplevelse? Hvorfor / Hvorfor ikke?
- Har du nogensinde haft brug for at indberette et ønske og / eller en fejl til kursus underviseren?
  - Hvordan var din oplevelse med det?
- Ved du hvordan du indberetter ønsker til videreudvikling / forbedringer af kursus undervisningen?
  - (Hvis de aldrig har haft brug for det: Hvis du alligevel ville indberette et ønske, hvordan ville du så gøre det?)
- Føler du, at du har indflydelse over kursets udvikling?
  - Føler du, at din indflydelse er tilstrækkelig?

Fig. 18: Interview Guide to the students in Preliminary Investigation

- tcp_and_reliable_communication
- server_performance
- security_across_the_network
- digital_logic
- single_cycle_datapath
- simulation_and_A5

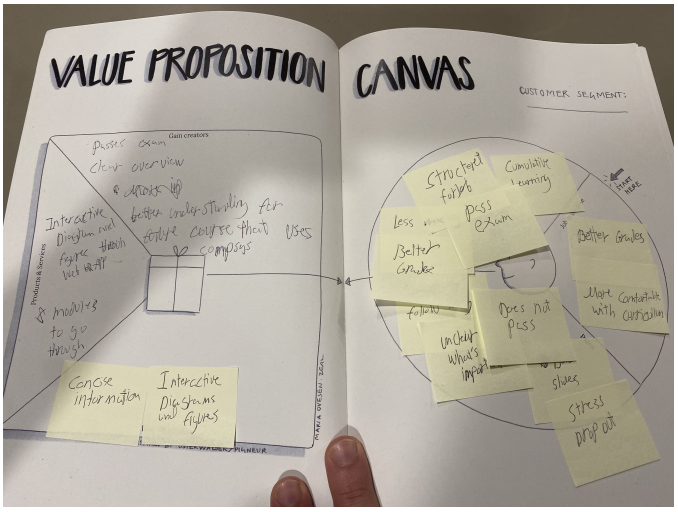


Fig. 19: Value Proposition Canvas of SysMentor

- pipelining
- advanced_microarch
- advanced_uarch_2
- multiprocessors
- network_layer_forwarding_ip
- network_layer_routing

H. Appendix H

Value Proposition Canvas and Sustainable Business Model canvas fig. 19 20

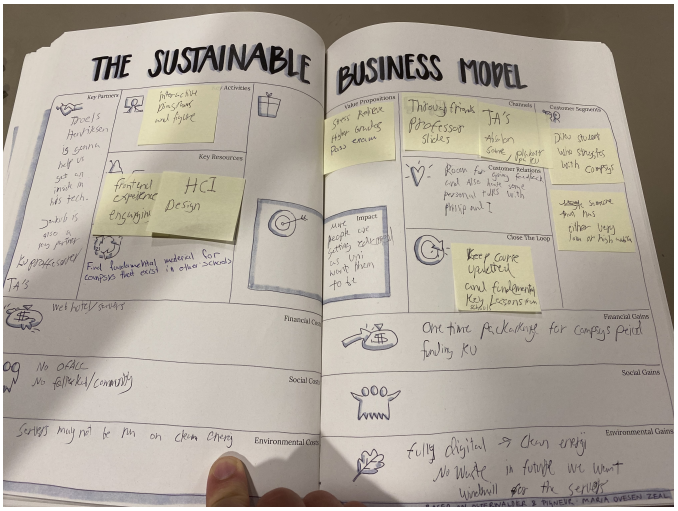


Fig. 20: Sustainable Business Model canvas of SysMentor

I. Appendix I

Interview feedback for the cache Gruppen udviste også at recap var godt, men dumt at det skulle ligge i enden af kurset.

Gruppen havde svært ved at løse eksamen opgaven. Det skal siges at gruppen heller ikke har gennemgået det fra recap.

Brugere tæller med musen uden at bruge farven. Det er først efter de bliver fortalt at de kan bruge farvene at de bruger dem.

Der er lidt misforståelse af hvad den grønne row gør når man holder musen over cachen.

Efter lidt hjælp kunne brugerne godt lave en exercise med input. brugere har specielt svært ved memoryblock address

"hvordan får man den samme farve?"

Brugeren klarede en opgave med 3 bit sets.

"Men opgaven er i hex, kan man få opgavenj i hex"

brugeren undersøger eksamensopgaven. og tænker over hvad er en 4-way set associate?" de sæteerer den til 4-way set associate i toolet for at se hvad der sker.

Det er svært at bygge intuition, da opgaven udleder nogle informationer der gør at man skal vide hvordan cachen fungerer. Altså 16-bytes blocks med 16 bytes hver = 2 sets

Den er lidt svært at forstå til at starte med

En tutorial kunne være godt - en popup

Der er nogle ting der er svære at se. Deter svært at få et godt overblik over det hele på én gang da man skal scrolle rigtig meget

Memory var rigtig svært at forstå.

Cachen kunne være godt at ligne den til eksamen. folk vil ikke bruge tid på den i egen fri vilje.

Offline tool.

J. Appendix J

VMAT exam exercise 21

Question 2.2.2: Consider a system with the following properties:

- Memory is byte-addressed.
- Virtual addresses are 13 bits wide.
- Physical addresses are 14 bits wide.
- The page size is 64 bytes.
- The TLB is 3-way set associative with 4 sets and 12 total entries. Its initial contents are:

Set	Tag	PPN	Valid	Tag	PPN	Valid	Tag	PPN	Valid
0	11	00	0	11	15	1	04	00	1
1	0A	10	0	0F	10	1	1F	2F	1
2	12	34	1	00	00	0	00	00	1
3	13	15	1	00	12	0	10	0A	1

- The page table contains 12 PTEs:

VPN	PPN	Valid	VPN	PPN	Valid	VPN	PPN	Valid	VPN	PPN	Valid
31	33	1	13	11	1	01	02	1	02	01	1
41	01	1	02	33	0	0B	0D	0	09	17	1
00	00	1	10	21	0	13	32	0	03	43	1

Note that all addresses are given in hexadecimal. In the following questions, you are asked, for various virtual addresses, to show the translation from virtual to physical addresses in the memory system just described. *Hint: there is one TLB hit, one page table hit, and one page fault (not necessarily in that order). This should help you double-check your work.*

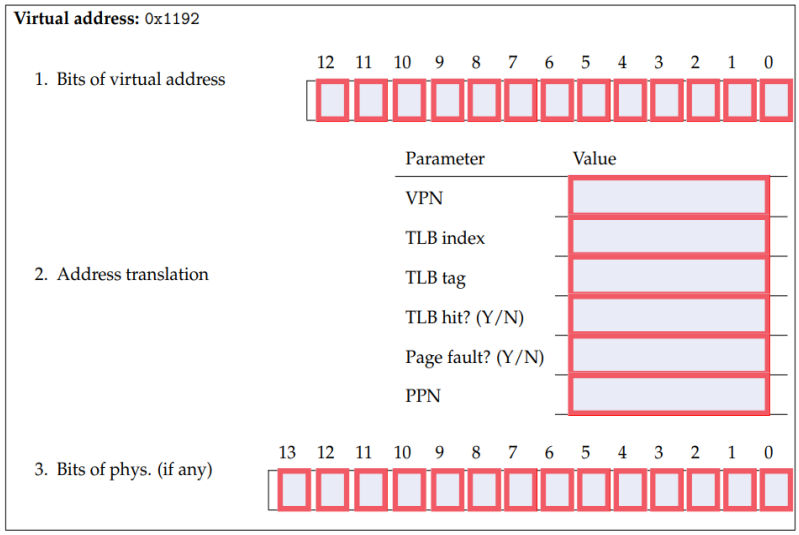


Fig. 21: VMAT exercise from exam 2021/22