Students' Perceptions on Engaging Database Domains and Structures

Miedema, Daphne; Taipalus, Toni; Aivaloglou, E.A.

DOI
10.1145/3545945.3569727

Publication date
2023

Document Version
Final published version

Published in
SIGCSE 2023 - Proceedings of the 54th ACM Technical Symposium on Computer Science Education

Students' Perceptions on Engaging Database Domains and Structures

Daphne Miedema
d.e.miedema@tue.nl
Eindhoven University of Technology
Eindhoven, the Netherlands

Toni Taipalus
toni.taipalus@jyu.fi
University of Jyväskylä
Jyväskylä, Finland

Efthimia Aivaloglou
e.aivaloglou@tudelft.nl
Open University
Heerlen, the Netherlands
Delft University of Technology
Delft, the Netherlands

ABSTRACT

Several educational studies have argued for the contextualization of assignments, i.e., for providing a context or a story instead of an abstract or symbolic problem statement. Such contextualization may have beneficial effects such as higher student engagement and lower dropout rates. In the domain of database education, textbooks and educators typically provide an example database for context. These are then used to introduce key concepts related to database design, and to illustrate querying. However, it remains unstudied what kinds of database contexts are engaging for novices. In this paper, we study which aspects of database domain and complexity students find engaging through student reflections on a database creation assignment. We identify six factors regarding engaging domains, and five factors for engaging complexity. The main factor for domain-related engagement was *Personal interest*, the main factor for complexity engagement was *Matching information requirements*. Our findings can help database educators and book authors to design engaging exercise databases targeted for novices.

CCS CONCEPTS

• **Applied computing** → **Education**; • **Information systems** → **Relational database model**.

KEYWORDS

database, education, context, engagement, design

ACM Reference Format:
Daphne Miedema, Toni Taipalus, and Efthimia Aivaloglou. 2023. Students' Perceptions on Engaging Database Domains and Structures. In *Proceedings of the 54th ACM Technical Symposium on Computer Science Education V. 1 (SIGCSE 2023), March 15–18, 2023, Toronto, ON, Canada.* ACM, New York, NY, USA, 7 pages. https://doi.org/10.1145/3545945.3569727

1 INTRODUCTION

Data management skills are becoming more and more central to the tasks of a computer scientist. Data management is of high importance in institutions using large amounts of data, for organizing data distribution and decentralized query execution. It is also core to the area of artificial intelligence, as no model can be trained without data. As such, data management techniques are also an essential part of computer science education.

Data management is typically taught by means of toy examples [23], with domains such as store ordering systems, movie rentals and company employees [21]. These examples are accessible: most students have some idea of what data could be involved in these domains, which they can use as a scaffold to remember the database schema. As such, the domain of the data is the inherent context of the database, but determining what makes a good database context not been studied in-depth before.

We can deepen students interest in course material by ensuring that the projects they work on are authentic, and by giving them a choice in the what and how of the project [2]. For example, teaching CS1 in the students major's context has led to increased student success rates [22]. Increased interest in a project may also lead the students to spend more time on an assignment [6]. Furthermore, increased interest makes students become more active learners, which in turn increases motivation and learning [11]. However, it has been shown that more complex databases have the possible downside of making learning SQL more difficult [21].

In this paper, we examine factors that students think make databases engaging, to answer the question: *What kind of databases do novices deem engaging to study database design, implementation and querying, and why?* We split this into two research questions:

RQ1 Which factors describe an engaging database application domain for students?
RQ2 Which factors are key to an engaging database in terms of structural complexity?

To answer these questions, we ran a study in which students designed and implemented their own relational database and queries. We asked them to elaborate on what they thought was engaging about the domain of their database, and the complexity of it. We identified six domain related factors, and five complexity related factors. Our findings can help teachers and textbook writers use and create more engaging examples, such that students can learn more effectively.

2 RELATED WORK

Whether or not students will benefit from teaching and assignments that are contextualized has been a point of discussion in the computer science education research community for some time now. On one hand, researchers argue that context can increase motivation and reduce student dropout by engaging the student in a concrete way by addressing real-world questions [5, 7, 13, 19]. On

SIGCSE '23, March 15–18, 2023, Toronto, ON, Canada.

ACM ISBN 978-1-4503-9431-4/23/03.
https://doi.org/10.1145/3545945.3569727

the other hand, contextualization can lead to negative side-effects such as high cognitive load [4] and thinking of newly learnt things as only applying in the current situation [12].

In CS education research, contextualization has received attention mainly outside the realm of database education. In perhaps the study most related to this one, Yue used a semi-realistic database (Sakila by MySQL) to increase engagement in their database course by moving away from the typical "toy" example databases [23]. The database was fully integrated in the course and used across multiple assignments. Students found these assignments both interesting and useful, although they also found it difficult because it is more complex than the examples used in textbooks. This increase in database complexity has also been shown to negatively affect successful query formulation among novices [21].

Bouvier et al. ran a study to see whether students perform better on contextualized programming exercises, versus plain, unembellished ones [4]. For this purpose, they made two versions of the same exercise, with and without context. The contextualized one used astronomy as its domain, and the plain one just mentioned numbers. They found that there was no significant difference in performance between the two versions of the exercise [4]. Similar results were also obtained by Leinonen, Denny and Whalley, who saw that, e.g., students found it easier to implement a linear equation solver when they had context for the problem, than without. Furthermore, context led to a longer time on task in seconds, but the contextualized problems also required fewer attempts [15]. Craig, Smith and Peterson ran a study in which they compared performance on contextualized programming assignments with performance on non-contextualized assignments [8]. In contrast to Leinonen et al., they did not find a measurable effect, leading them to the conclusion that other factors, besides context, impact difficulty and performance more.

Beyond these discussions on performance, it is also interesting to see what students themselves think of contextualization. In 2004, Rich, Perry and Guzdial ran a study in which they compared a contextualized CS1 course to a standard one [19]. Their aim was to attract and retain female students in their CS1 course, and their results suggest that they were successful: female students expressed greater enjoyment and interest in the course than those in the standard course [19]. In similar vein, Cliburn and Miller found that students liked to have a choice as to how to demonstrate their skill levels, rather than all following the same assignment [6].

It is also worth noting that while contextualization is not a simple concept, measuring its benefits can also be difficult. Rader et al. examined whether students would be more interested in socially relevant contextualized programming tasks than in other types of tasks [18]. In a survey, students responded that they like assignments that show how computer science can be used to benefit society. However, when ranking projects in order of most interesting to least interesting, students showed a higher preference to simpler game-based projects. This preference may be partially explained by the complexity and open-endedness of the humanitarian problem descriptions [18]. Additionally, in the context of computer security education, DeWitt and Cicalese found that students were more engaged with real-world stories over fictitious scenarios. However, context and content need to be in balance to accommodate both traditional computer science students, and those that want to learn to apply techniques in real-world context [9].

Finally, choosing the right context may be difficult. Yarosh and Guzdial applied a narrative contextualization in their media computation CS2 course, in which they used one running example throughout the course [22]. They found that while most of the students agreed that context made the class more interesting, some students found the context to rather taking away from learning the course contents in greater depth. Therefore, engagement is not only about the presence or absence of context. Some other dimensions of engagement include connectedness [10], positive emotions [1, 17, 20], passion [1, 17] and motivation [1, 20].

Engaging assignments can make active students out of passive recipients of knowledge, with active students reaching higher levels of motivation and learning [11]. This leads us to the question of which of these aspects of engagement we can utilize with the context of our database assignments. In this paper, we focus on student perceptions on engaging database education. With our research questions, we aim to identify factors that educators can use to guide exercise database design, as it remains unclear which factors contribute to engagement in database education.

3 RESEARCH SETTING

3.1 Data Collection

The data were collected from three cohorts (springs of 2019-2021) of mainly third year software engineering and information systems science students at a Finnish university. The students had to have taken a basic course on databases and data management before taking this course. The students were shown a data privacy statement prior to participation, and could then opt-in for the study, where participation carried no incentives or penalties. Out of the 68 students, 56 (82%) chose to participate (15/17 from the first cohort, 8/10 from the second, and 33/41 from the third).

The participants were asked to design and implement a relational database that they deemed engaging for novices who are learning database design and querying. The assignment consisted of multiple tasks, among which the following two are relevant for this study:

- Choose an application domain for their database that they deemed engaging for a novice to practice querying, and reflect on why that particular domain is engaging.
- Design the database structure to be engaging for a novice to practice querying, and reflect on why that particular structure is engaging. The minimum requirement for the number of tables is five.

In this study we analyze the students' reflection reports from the perspective of our research questions. As the language of instruction was Finnish, all questions (including the ones above), databases and reflections were written in Finnish. All answers were translated through Google Translate and checked by a native speaker.

3.2 Data analysis

From the 56 participants mentioned above, 2 provided the database but did not reflect on their reasoning. These students are not included in the data, so we were left with 54 participants in total. These participants are indicated by P01 through P54.

Our analysis was done in four steps: mining factors, calibration of coding, coding remaining data, and summarizing. We applied directed content analysis, which is recognized as a viable option when previous literature exists but is deemed incomplete [14].

Mining factors Before we started our analysis, we used existing literature to create a list of factors to search for in the reflections. We started this approach from the ITiCSE working group report on contextualized programming education [4]. From this report we retrieved five initial factors: Perceived relevance, Retention, Connection to practice, Impact, and Exploring social and ethical issues. All five of these were used for coding both domain and complexity factors.

Calibration of coding After setting up this baseline of factors, all authors individually coded the same subset of the data (20%). For this we used both the existing factors and introduced new factors based on the text in the reflections. Afterwards, we aggregated to see where (dis)agreements arose in the coding. To calibrate our coding, we discussed all cases together, as well as any new factors that may have been introduced by any of the authors. New factors were either introduced separately, merged with related factors, or discarded after discussion.

Coding remaining data Then, two authors split the remaining data in half and coded individually. For all cases that were not immediately clear, meetings were held to discuss best fits. No new factors were introduced during this time.

Summarizing After the coding was done, we performed analysis by examining and summarizing all codes per factor. This work was calibrated by working according to an example provided by one of the authors. Within this process of summarizing, we also decided to merge or drop some factors, as they were not contributing to our research question. First, we merged the two types of versatility from both domain and complexity questions, the gist of both was *versatility in difficulty and versatility in extensibility*. We also dropped a complexity factor around the concept of *enough data to practice, no empty result tables*, and a complexity factor where students decided how many columns to create by *intuition*. Neither of the latter two can be used as instructional guidelines, which is why we decided to leave them out.

4 RESULTS

4.1 Engaging domains

D1. Familiarity. Familiarity, including ease and understandability, was mentioned as a factor by 25 students. They speculated that a database domain that is familiar through, e.g., personal life, previous studies, or work would be engaging. For example, P07 wrote:

> "The database related to listening to music is interesting for the student, because almost everyone uses some kind of music streaming service every day."

Several students pointed out that a familiar domain decreases the initial threshold for starting the learning process towards database theory and query writing. P03 wrote:

> "[...] simpler target areas make it easier to internalize new things and maintain interest in the subject more effectively."

On the other hand, a domain not necessarily familiar to many students, but in turn easy to understand, was considered another option for an engaging database domain. In contrast, students highlighted that a domain that would require extensive familiarization would take away from the intended primary learning outcomes of a database course:

> "[...] financial data repository with special terms and complex modeling can cause gray hair for beginners [...] and therefore I don't think is as well suited for training." - P20

D2. Connection to practice. Related to Familiarity was the factor Connection to practice, mentioned by 30 students. They speculated that understanding the database domain and, e.g., learning to query in this particular domain would also help them in their future work. P07 wrote that:

> "I am interested in the music and entertainment industry in general, so combining the two tells me about my own interests and desire to get a job in the industry."

In contrast, a domain that could be considered niche or rare would be relatively useless outside the scope of a database course. Another consideration included in this factor was the view of increasingly common application domains such as electronic sports and electronic commerce. For example, P01 wrote:

> "[...] useful for working life as it could be real and buying from an online store is growing all the time. This will require more database workers who can handle relational databases in that topic."

D3. Learning opportunity or challenge. 27 students indicated that a domain that provides a learning opportunity or challenge was engaging.

On the one hand, these learning opportunities and challenges could be related to application domains that are complex in terms of the resulting database structure, such that students learn more about database concepts:

> "The student will also learn to design multi-relationship relationships, such as with multiple orders, products, and product categories." - P08

On the other hand, there are learning opportunities to be found in studying the domain itself in order to create correct schemas:

> "Working with a database of publication information can deepen a student's knowledge of scientific publishing, for example, in terms of knowledge of different types of publications." - P06

D4. Versatility There are two sides to versatility as a factor. The first is versatility in the dimension of query difficulty. This is linked to the concept of growth mindset for education, and that of "low floor, high ceiling", meaning that assignments are accessible to all students from novices up, but that it also has room to challenge advanced students [3]. The other side is versatility in a database that is easy to expand in size and complexity.

For versatility in query difficulty, ten students argued that this was a factor that made their domain engaging:

> "I wanted to make a position that would be clear enough for new students, as well as that it would contain complex data that could also be used to challenge older students." - P10

> "[...] there are still opportunities for a bit more complex queries, which makes this particularly interesting." - P15

There were eleven students who deemed versatility in terms of a database structure that is easy to expand a factor for engagement. *"I also felt that the topic I chose was broad enough to implement a wide variety of databases, but still easy to summarize."* (P45) With a flexible structure, a novice could mentally select parts of the database, whereas an expert could use all parts.

D5. Personal interest. The most prominent motivation for rationalizing an engaging domain was personal interest. As many as 33 students argued that the application domain should be interesting for a novice. Students who argued for an interesting domain typically also associated interesting domains with connection to practice through hobbies, such as P16 and P51:

> "The target area feels pleasant to me, because there will be a lot of gigs and listening to music, so I imagine that it will inspire others in the same way." - P16

> "I chose the simplified online gaming store [...] Examples of a real product are the Valve Steam service or the GOG service. I've been playing video games in my spare time since I was young, so the topic seemed natural." - P51

Other students focused on business ideas they or their friends were setting up, by developing a database for their domain:

> "The target area was the business information of a small company called [anonymized]. This company is really in business, but the business is very small. [...] Against this background, I am building a database structure and possibly a web application program. " - P43

Not unexpectedly, domains chosen because of this justification were diverse in nature.

D6. Social and ethical issues. Five students deemed that an engaging application domain should address social or ethical issues. Students argued that the application domain should be one that has a seemingly concrete and big impact on our lives.

> "[...] a more serious approach than handling games, for example. If someone makes a small careless mistake in a game character's hat color, no one may even notice it. On the other hand, if a patient has access to hospital drug stores or the surgeon is prevented from entering the operating room, the consequences can be very serious. [...] The database is not designed for entertainment use, but to ensure and promote the safety of people in the hospital environment." - P12

A domain which, due to its important nature, required careful design, was also considered a source of engagement.

D7. Other. There were five students who argued for something outside of the aforementioned factors. They discussed their chosen domain's engagement with respect to the diversity in data (1), structure of the data (1), and expanding, complementing or iterating a software project initiated during another course (3).

> "I wanted to improve my Programming 2 course assignment, Story Statistics, which currently works with text files. That program is still in use by me today and contains a relatively large amount of data, so some version that uses a decent database could be useful at some point, especially if I wanted to develop the structure of the data somehow." - P32

4.2 Engaging complexity

C1. Simplicity. A commonly argued characteristic of an engaging database schema, mentioned by 24 participants, was that it should be simple and understandable. P17 summarizes this as:

> "The database had to be interesting for the student so a database that looks too challenging will reduce the student's interest, I think."

The main arguments in favor of simplicity were that it makes it easier to practice writing queries (3 students), to populate the database (1), to expand it (2), and to avoid errors (2), which were also linked to increased complexity due to normalization.

Four students explicitly connected schema simplicity to learning, as *"the complexity of the database could become a barrier to learning"* (P06) and *"in terms of learning, a simple data model is easier to adopt"* (P20). Four students designed a simple database to match their beginner database skills, for example:

> "I wanted to keep the model as simple as possible, as this is the first database I have actually started to implement myself. The main focus was on getting everything to work without a big headache." - P42

C2. Matching information requirements. 36 students described designing their database according to the perceived user or information requirements. Those requirements pertain to the database domain or the software application that would be based on the database. In many cases, students provided a justification per table and relationship and they described the process they followed to come up with the final schema, for example P30 describes:

> "I started by thinking about the features I would like to have in an imaginary service. I set out to implement the structure of the database through an ER diagram and wondered what items and what kind of relationships the properties I wanted required. I drew an ER diagram and selected the appropriate attributes for the objects and relationships. I transformed the diagram into a relational model according to the transformation rules."

Students commonly described that the database domain led them to the definition of specific attributes for their database, for example:

> "In choosing the number of tables and attributes, I started by thinking about which attributes of the games were essential and important for me to include in the database [..] So I didn't really deliberately choose just that number of tables and columns, but the number of them came as dictated as needed as the plan progressed." - P37

Four students explicitly mentioned matching how a real world database would be created, for example:

> "In my opinion, the database I have created is useful for working life, because the database is built for a specific real purpose, although the format implemented during the course is not exactly the same as it is or would be in real life." - P16

Some aspects related to information requirements were mentioned rarely as driving the database design. These were the customer requirements (mentioned by P05), enabling search conditions (P24), and supporting database expansion (two students), for example:

> "[...] thought that if publishers or developers ever wanted to expand with new columns, the changes would be easy when they already have separate tables for them." - P27

C3. Learning opportunity or challenge. Nine students argued that the database schema should support learning. Most of them referred to learning about databases in general or in the context of the databases course, but three of them explicitly referred to learning as preparation for working life; for example, P06 mentioned:

> "[...] in my experience, a database with five tables that meets the minimum requirements for the assignment would be too small to give the student a correct understanding of the databases that will come up in later working life."

Students also gave us insight into how a database schema and its complexity affect learning. A complex database schema enables practicing with *"memorizing conceptual entities"* (P06), encourages the use of a database schema description (the example of the Entity Relationship diagram was mentioned by a student) when formulating queries, and could

> "[...] demonstrate all the different concepts used in databases (e.g. different types of fact tables in the Kimball model, different types of dimensions in the tables, more metadata columns and tables corresponding to the correct implementation, tables on log data)" - P20

P21 agreed with the learning benefits of a simple database schema:

> "I think learning the basics is supported by moving from simpler schemes to more complex ones. With a simple schema database, it is good to practice making queries and learn to understand logic related to joins, for example."

A complex schema can also make query formulation more challenging, as P55 expresses:

> "Sure, there could have been more tables and columns for the customer's address, for example, but I feel it wouldn't have contributed more to my learning. I think there were enough tables and columns to learn how to use the environment, but not too many to make it too difficult in solving problems."

C4. Versatility. Eight students mentioned that a database schema should be complicated enough to represent various database characteristics (three students) and to enable practicing with query writing (five students). The database characteristics that students referred to were data types, table relationships, and number of columns, for example:

> "[...] there must be enough columns in at least a few tables to ask meaningful questions. In addition, for queries that use aggregate functions, grouping, and sorting at the same time, it is a good idea to have more columns in the table than the number of functions or expressions used, so that the function, grouping, or sorting can be assigned to different columns and more data-rich" - P06

The database schema should also be versatile enough to support extension for more complex query practice, as summarized by P21:

> "In its current form, the structure of the database is, in my opinion, suitably simple, but at the same time it makes it possible to carry out a wide range of database queries."

C5. Practical constraints. 15 students expressed that in their database design they took into account practical constraints, specifically assignment restrictions (9), time limitations (4), or challenges with generating model data (2). Assignment restrictions pertain mainly to the minimum number of five tables that was set. The effect of time limitations is described by P51:

> 'The biggest factor limiting the size of the exercise at the moment is time. The work and other courses take up most of my time and I might have liked to build the database more thoroughly."

Finally, P47 mentioned the effect of limitations of the tool that was used for populating the database with data:

> "Mock data contributed in part to the fact that it largely determined the types of data I used in the tables [...] I also added some of the columns because of the effect of the mock data, because as I looked at it I noticed more things I had previously forgotten about in the design."

C6. Other. Three students mentioned desirable characteristics of the database schema that did not fit into any of the aforementioned ones. Those were to limit the schema to the most significant information and *"focus only on the most important [tables]"* (P08), to match the scope of the course and therefore avoid adding password data *"because with passwords you should be aware of security issues that the course doesn't focus on anyway"* (P27), and to avoid adding calculated columns (P33).

5 DISCUSSION

In Section 3.2 we introduced that our content analysis was led by both the data and the work by Bouvier et al. on context in CS Education [4]. We'll now reflect on whether the factors we identified are in agreement with theirs.

Perceiving relevance. This theme is reflected in multiple identified factors, but most prominently in D3: Learning opportunity or challenge. The students who mentioned this factor were looking to learn about either their domain or databases in more detail, showing that they saw the relevance of these things. We can also argue that it links to D2: Connection to practice and D6: Social and ethical issues, but as we adapted these directly from alternative factors by Bouvier et al. [4], we leave them out of the discussion here.

Retention. This factor did not come up in any of our students' answers. However, this is likely to be an artifact of the assignment as a short-term reflection. We'd like to argue that this theme is implicitly reflected in D3: Learning opportunity or challenge, as students focused on learning which hopefully leads to retention.

Connection to practice. We adopted this theme in our content analysis and found enough evidence that we kept it in as a standalone factor.

Impact. Bouvier et al. write about this factor as impact on the broader user community. This factor came up regularly and is mostly coded under Personal interest, where students mention using their assignment database for a start-up by themselves or for helping their friends.

Exploring social and ethical issues. This was kept as a separate factor: D6, but it was mentioned only sporadically. In literature, this is described as a factor that may help female students stay engaged in the material. Unfortunately, as we do not have gender identity data of our students, no conclusions can be drawn on this.

A factor that arose from our findings, and is not mentioned by Bouvier et al. is Versatility. It is worth noting that the two notions of versatility (in business domains (D4), and in database complexity (C4)) had subtle but distinct differences. By domain versatility, the students argued for engagement through being able to mentally divide the database into self-contained subsets. This allows students to take the subset that fits their experience level and practice with that. On the other hand, a database with structural versatility was seen to facilitate engagement through being easy to expand.

On the performance side of the contextualization discussion, there are some concerns with the cognitive load associated with contextualization. However, we want to note that data management and databases cannot be taught fully de-contextualized: we need data on some domain to fill our databases and inform our queries. As such, it seems better to adapt (some of) the factors that our students wish for than to select something randomly or keep doing what we have always done.

Furthermore, our findings expand on the notion of "useful and interesting" exercise databases presented by Yue, who found that students prefer more realistic databases in learning database related topics, as these databases are perceived to more closely reflect the databases the students are likely to encounter in their future work [23]. However, Taipalus showed that more complex databases might negatively affect how students learn to query [21]. However, Taipalus also pointed out that more complex databases are not necessarily synonymous to more realistic databases, and student engagement might not depend on realism, but rather on perceived realism. If a teacher can use a relatively simple exercise database that is also argued to be a realistic one, this might be the key to supporting both query formulation learning and engagement.

Our findings support Yue's observations, as several students pointed out that learning opportunities and challenges sustain their engagement. However, we also observed that there are many other, sometimes contradicting factors which all relate to student engagement. Arguably, and although it is possible that a single exercise database may not fulfill all the engagement considerations, teachers should consider the factors presented in Section 4.1 when they design exercise database domains and complexities.

Finally, for our analysis, we chose to work with a database assignment and reflection, instead of the more common questionnaire approach. Rader et al. [18] in their study found that what students say they want to do, differs from what they actually do. The advantage of our approach is that students had to commit to the domain they chose. They could not arbitrarily select something, but had to consider the implications of their choice for the development of the database and queries.

5.1 Limitations and threats to validity

We note several limitations to our study. First of all, the assignment required all databases to contain at least five tables. Although we do not draw any conclusions from calculated complexity metrics, this requirement was often echoed in students argumentation on engagement. Given the findings of Rader et al. [18] that students prefer simpler games, and factor C1: Simplicity, we believe that students' behavior might have been different without this restriction.

Second, there is the issue of students self-selection for participation in this study. Although all students in the course had to complete the exercises we study in this paper, it may be that participating students were more motivated to deliver a good result.

Finally, we reflect on the validity of our study based on the definitions by Maxwell [16]. As our study consists of documents containing student argumentation, there is little risk of misunderstanding or misinterpreting students' accounts on the side of the authors (high descriptive validity). However, there is a threat here, as we cannot guarantee that all students understood the questions correctly. One clear example of this issue is shown in factor D5: Personal interest, where some students outlined engagement in terms of their own personal interest, instead of interest to all novices.

All authors worked together on the content analysis, leading to higher likelihood of reliable data coding. Although some interpretation of student answers is required to extract the factors, we believe that the required level of interpretive validity is met, given that our analysis is qualitative and uses quotes from the student answers. Finally, although our sample size can be considered small for quantitative studies, it is on the larger size for qualitative studies. As such, we believe that the factors are representative (but not exhaustive) for other student populations as well.

We want to conclude this discussion section by acknowledging that, although we can gain information about students' leisure pursuits, it is difficult to create appropriate assignments based on this information. To correctly contextualize according to Cooper and Cunningham, we should make the chosen context a theme throughout the course [7], which is a difficult undertaking. One barrier identified by Blumenfeld et al. is that the assignments need to be difficult enough for students to learn about the subject in the required depth [2]. It may be that not all contextualized assignments are of the same difficulty, as Cliburn and Miller hypothesize about the assignments in their work [6]. Rader et al. also concluded that some of their contextualized assignments were too complex and open-ended [18]. Therefore, there is a trade-off to be navigated between how open we can leave assignments to match student interest, and how consistent and in-depth assignments are.

6 CONCLUSIONS AND FUTURE WORK

In this paper we introduced factors that students perceive as engaging for Database Education. Our findings are based on an assignment in which students designed their own database schema, data and queries. Students were asked to reflect on why their database is engaging. For engaging domains, we found the factors *Familiarity*, *Connection to practice*, *Personal interest* and *Social and Ethical issues*. For engaging complexity, we identified *Simplicity*, *Matching information requirements* and *Practical constraints*. Two shared factors are *Learning opportunity or challenge* and *Versatility*.

This paper represents an exploration of what aspects of context make databases engaging. Although the current findings can already be used to update database assignments, future work should explore in more detail what the effect of adopting our findings is on student engagement, and possibly performance. Furthermore, it would be interesting to compare how such new materials compare to commonly used materials and toy examples from textbooks. Finally, we should explore which domains students seem to find interesting, and whether any trends can be found in this data.

REFERENCES

[1] Elizabeth F Barkley and Claire H Major. 2020. *Student engagement techniques: A handbook for college faculty.* John Wiley & Sons.

[2] Phyllis C. Blumenfeld, Elliot Soloway, Ronald W. Marx, Joseph S. Krajcik, Mark Guzdial, and Annemarie Palincsar. 1991. Motivating Project-Based Learning: Sustaining the Doing, Supporting the Learning. *Educational Psychologist* 26, 3-4 (1991), 369–398. https://doi.org/10.1080/00461520.1991.9653139

[3] Jo Boaler, Jack A Dieckmann, Tanya LaMar, Miriam Leshin, Megan Selbach-Allen, and Graciela Pérez-Núñez. 2021. The Transformative Impact of a Mathematical Mindset Experience Taught at Scale. *Frontiers in Education* 6 (2021), 1–13. https://doi.org/10.3389/feduc.2021.784393

[4] Dennis Bouvier, Ellie Lovellette, John Matta, Bedour Alshaigy, Brett A. Becker, Michelle Craig, Jana Jackova, Robert McCartney, Kate Sanders, and Mark Zarb. 2016. Novice Programmers and the Problem Description Effect. In *Proceedings of the 2016 ITiCSE Working Group Reports* (Arequipa, Peru) *(ITiCSE '16)*. Association for Computing Machinery, New York, NY, USA, 103–118. https://doi.org/10.1145/3024906.3024912

[5] Evelyn Brannock, Robert Lutz, and Nannette Napier. 2013. Integrating authentic learning into a software development course: An experience report. In *Proceedings of the 14th annual ACM SIGITE conference on Information technology education.* 195–200.

[6] Daniel C. Cliburn and Susan Miller. 2008. Games, Stories, or Something More Traditional: The Types of Assignments College Students Prefer. In *Proceedings of the 39th SIGCSE Technical Symposium on Computer Science Education* (Portland, OR, USA) *(SIGCSE '08)*. Association for Computing Machinery, New York, NY, USA, 138–142. https://doi.org/10.1145/1352135.1352184

[7] Steve Cooper and Steve Cunningham. 2010. Teaching computer science in context. *ACM Inroads* 1, 1 (2010), 5–8.

[8] Michelle Craig, Jacqueline Smith, and Andrew Petersen. 2017. Familiar Contexts and the Difficulty of Programming Problems. In *Proceedings of the 17th Koli Calling International Conference on Computing Education Research* (Koli, Finland) *(Koli Calling '17)*. Association for Computing Machinery, New York, NY, USA, 123–127. https://doi.org/10.1145/3141880.3141898

[9] Janine DeWitt and Cynthia Cicalese. 2006. Contextual integration: A framework for presenting social, legal, and ethical content across the computer security and information assurance curriculum. In *Proceedings of the 3rd annual conference on Information security curriculum development.* 30–40.

[10] A Fletcher. 2015. Defining student engagement: A literature review. https://soundout.org/2015/03/29/defining-student-engagement-a-literature-review/

[11] James E Groccia. 2018. What is student engagement? *New directions for teaching and learning* 2018, 154 (2018), 11–20.

[12] Mark Guzdial. 2010. Does Contextualized Computing Education Help? *ACM Inroads* 1, 4 (Dec. 2010), 4–6. https://doi.org/10.1145/1869746.1869747

[13] Nira Herrmann and Jeffrey L Popyack. 1995. Creating an authentic learning experience in introductory programming courses. In *Proceedings of the twenty-sixth SIGCSE technical symposium on Computer Science Education.* 199–203.

[14] Hsiu-Fang Hsieh and Sarah E Shannon. 2005. Three Approaches to Qualitative Content Analysis. *Qual. Health Res.* 15, 9 (2005), 1277–1288. https://doi.org/10.1177/1049732305276687

[15] Juho Leinonen, Paul Denny, and Jacqueline Whalley. 2021. Exploring the Effects of Contextualized Problem Descriptions on Problem Solving. In *Australasian Computing Education Conference* (Virtual, SA, Australia) *(ACE '21)*. Association for Computing Machinery, New York, NY, USA, 30–39. https://doi.org/10.1145/3441636.3442302

[16] Joseph Maxwell. 1992. Understanding and validity in qualitative research. *Harvard educational review* 62, 3 (1992), 279–301.

[17] Great Schools Partnership. 2016. Student Engagement. In *Glossary of Education Reform.* https://www.edglossary.org/student-engagement/

[18] Cyndi Rader, Doug Hakkarinen, Barbara M. Moskal, and Keith Hellman. 2011. Exploring the Appeal of Socially Relevant Computing: Are Students Interested in Socially Relevant Problems?. In *Proceedings of the 42nd ACM Technical Symposium on Computer Science Education* (Dallas, TX, USA) *(SIGCSE '11)*. Association for Computing Machinery, New York, NY, USA, 423–428. https://doi.org/10.1145/1953163.1953288

[19] Lauren Rich, Heather Perry, and Mark Guzdial. 2004. A CS1 course designed to address interests of women. In *Proceedings of the 35th SIGCSE technical symposium on Computer Science Education.* ACM New York, NY, USA, 190–194.

[20] Ellen A Skinner and Michael J Belmont. 1993. Motivation in the classroom: Reciprocal effects of teacher behavior and student engagement across the school year. *Journal of educational psychology* 85, 4 (1993), 571.

[21] Toni Taipalus. 2020. The effects of database complexity on SQL query formulation. *Journal of Systems and Software* 165 (2020), 110576. https://doi.org/10.1016/j.jss.2020.110576

[22] Svetlana Yarosh and Mark Guzdial. 2008. Narrating data structures: The role of context in CS2. *Journal on Educational Resources in Computing (JERIC)* 7, 4 (2008), 1–20.

[23] Kwok-Bun Yue. 2013. Using a Semi-Realistic Database to Support a Database Course. *Journal of Information Systems Education* 24, 4 (2013), 327–336. http://jise.org/Volume24/n4/JISEv24n4p327.pdf