Promoting Privacy Considerations in Real-World Projects in Capstone Courses with Ideation Cards

YING TANG, MORGAN L. BROCKMAN, and SAMEER PATIL, Indiana University Bloomington, United States

Nearly all software built today impinges upon end-user privacy and needs to comply with relevant regulations. Therefore, there have been increasing calls for integrating considerations of compliance with privacy regulations throughout the software engineering lifecycle. However, software engineers are typically trained in the technical fields and lack sufficient knowledge and support for sociotechnical considerations of privacy. Privacy ideation cards attempt to address this issue by making privacy compliance understandable and actionable for software developers. However, the application of privacy ideation cards in real-world software projects has not yet been systemically investigated. The effectiveness of ideation cards as a pedagogical tool has not yet been examined either. We address these gaps by studying how teams of undergraduate students applied privacy ideation cards in capstone projects that involved building real-world software for industry sponsors. We found that privacy ideation cards fostered greater consideration and understanding of the extent to which the projects aligned with privacy regulations. We identified three main themes from student discussions of privacy compliance: (i) defining personal data; (ii) assigning responsibility for privacy compliance; and (iii) determining and exercising autonomy. The results suggest that application of the cards for real-world projects requires careful consideration of intersecting factors such as the stage at which the cards are used and the autonomy available to the developers. Pedagogically, ideation cards can facilitate low-level cognitive engagement (especially the cognitive processes of meaning construction and interpretation) for specific components within a project. Higher-level cognitive processes were comparatively rare in ideation sessions. These findings provide important insight to help enhance capstone instruction and to improve privacy ideation cards to increase their impact on the privacy properties of the developed software.

CCS Concepts: • **Social and professional topics → Computing education**; • **Software and its engineering → Designing software**; • **Security and privacy → Software and application security;**

Additional Key Words and Phrases: Privacy, compliance, reflective design, ideation cards, software design, privacy laws, privacy regulation

ACM Reference format:
Ying Tang, Morgan L. Brockman, and Sameer Patil. 2021. Promoting Privacy Considerations in Real-World Projects in Capstone Courses with Ideation Cards. *ACM Trans. Comput. Educ.* 21, 4, Article 34 (October 2021), 28 pages.
https://doi.org/10.1145/3458038

This research is supported by National Science Foundation (NSF) Grants No. CNS-1548779, No. CNS-1727574, No. DGE-1821782, and No. DGE-1821822.

Authors' addresses: Y. Tang, Faculty of Education, Southwest University, 2 Tiansheng Rd, Beibei, Chongqing, China 400715; email: yingtang@swu.edu.cn; M. L. Brockman, Indiana University Bloomington, 901 E. 10th St., Bloomington, Indiana, United States; email: morbrock@indiana.edu; S. Patil, School of Computing, University of Utah, 50 Central Campus Drive, Salt Lake City, Utah 84112; email: sameer.patil@utah.edu.

1946-6226/2021/10-ART34 $15.00
https://doi.org/10.1145/3458038

1 INTRODUCTION

Privacy can be characterized as the right of individuals to have control over data about themselves [4]. With rapid technological progress and the evolution of the Internet from "a web of pages to a web of people" [40], companies and governments are gaining new powers of surveillance and manipulation over citizens by gathering, sharing, and using the vast amount of personal data generated by people's everyday online and offline activities [16, 34, 51]. Users increasingly recognize that receiving personalized services and other benefits often requires disclosure of personal data. At the same time, users report being concerned about how their data is collected, used, and stored. In a 2019 Pew survey of 4,272 American adults, 81% of the respondents felt that they had little control over the data collected about them by companies, 79% said they were not confident that companies would handle their personal data responsibly, and 70% said their data is less secure compared to five years ago [3].

The past couple of decades have seen the issue of personal data protection addressed via a number of important privacy regulations, such as **European Union (EU) General Data Protection Regulation (GDPR)** [19] and various U.S. state-level privacy laws, including the most recent **California Consumer Privacy Act (CCPA)** [7]. Complying with these laws by implementing the regulatory requirements in software systems necessitates translating complex social, legal, and ethical matters into technical system requirements and operation [24], thus raising a number of challenges for software professionals. First, legal regulations are written using jargon not easily accessible to system designers and developers [2]. Second, consideration of legal implications is typically handled by legal compliance teams rather than developers [45]. Third, regulations are usually framed in general terms to ensure broad coverage at high levels, thus making it difficult to apply them to low-level specifics of software implementation [12].

To include privacy considerations *within* the software lifecycle, rather than considering them as an afterthought relegated to those handling quality assurance and legal compliance, researchers and regulators have proposed the ***Privacy by Design* (PbD)** framework for software development [8, 9]. The central philosophy of PbD is to "create a sustainable data protection system through the early use of adapted privacy enhancing technologies in the design of the processing operations and throughout the lifecycle of the data" [10]. However, PbD has been criticized as vague and ineffectual because it does not include specific tools and methods to train software engineers to deploy these principles, models, and mechanisms into real-world systems [24, 49, 53, 54, 58]. To facilitate practical application of PbD principles, Luger et al. [36] proposed a set of ***Privacy Ideation Cards* (PICs)** to make privacy regulations more understandable and accessible to software professionals. However, their empirical investigation of the application of PICs was limited to pre-defined system descriptions created by the researchers themselves. As a result, the impact of PICs has not yet been explored in real-world software development.

Further, it has been widely recognized that adequately trained software professionals are key to facilitating effective translation of privacy principles into system requirements [30, 49]. Accordingly, instilling privacy proficiency in software students before they graduate and join the workforce can promote more effective consideration of privacy compliance in the software industry in general [14, 30, 35]. Although most software engineering curricula require students to work on real-world projects as a capstone experience, these projects typically do not explicitly require students to consider privacy aspects related to the projects. PICs could be used to help students learn about and address privacy aspects of their capstone projects. However, their utility as a pedagogical tool for inexperienced software professionals has not yet been examined.

To fill the two gaps identified above, we formulated the following two research questions:

— ***RQ1:*** What are the main considerations related to privacy compliance in real-world software projects?
— ***RQ2:*** To what extent do privacy ideation cards help students consider the privacy aspects pertaining to real-world software?

We addressed the above two research questions by conducting three iterations of ideation sessions using PICs in undergraduate capstone courses involving real-world projects. Each iteration involved one cohort of software students who engaged in team sessions in which the teams applied PICs to reflect on the privacy aspects related to their capstone projects. We analyzed how student teams applied PICs to their capstone projects using the reflective practitioner's perspective [26, 48].

Based on our analysis, we make the following contributions:

— We empirically demonstrate the extent to which PICs are applicable to real-world software projects, highlighting their important strengths and shortcomings as a practical tool for software professionals.
— We apply the reflective practitioner's perspective to software engineering education and show that PICs can be a useful pedagogical tool to help software students learn about privacy.
— We surface several themes that can serve as the foundation for privacy education in the computing disciplines.
— We apply our insight to provide a number of suggestions for improving PICs, pedagogical strategies, and software development practices to support privacy considerations.

In the following section, we review current approaches to privacy compliance in software engineering and summarize Lugar et al.'s [36] study on utilizing PICs that inspired our work. We additionally discuss the reflective practitioner perspective in the context of software development that guided our analysis of student learning in ideation sessions using PICs. Next, we present the research design, including the context and participants, the ideation activity procedures, and details of the data collection and analysis. Based on the analysis, we proceed to answer the two research questions mentioned above. We then discuss the insight gained from the findings to present a range of privacy considerations that are not yet fully supported in the training of software engineers and apply the insight to provide suggestions for improving PICs, software engineering education, and software development practices. Finally, we acknowledge a few limitations before concluding with proposing promising directions for future research.

2 RELATED WORK

In this section, we review three areas of scholarship that inform our research: (1) the PbD framework and previous attempts to apply this framework to real-world software development, (2) the use of ideation cards to support design, including software design and development, and (3) the reflective practitioner perspective that guided our analysis of student reflection and learning using PICs along with its connection to software development.

2.1 Privacy by Design

PbD emphasizes the importance of considering privacy as an integral part of the products or services from the outset, rather than tacking it later on as a response to problems [9, 14]. PbD outlines seven guiding principles [8]:

(1) Proactive not Reactive; Preventative not Remedial;
(2) Privacy as the Default Setting;

(3) Privacy Embedded into Design;
(4) Full Functionality—Positive-Sum, not Zero-Sum;
(5) End-to-End Security—Full Lifecycle Protection;
(6) Visibility and Transparency—Keep it Open; and
(7) Respect for User Privacy—Keep it User-Centric.

Despite the comprehensiveness of the PbD approach, researchers have pointed to a disconnect between the tenets of PbD and its application in practice [24, 49, 53, 54, 58]. Simply put, it is hard to translate the ideas of PbD into a set of practices that are useful and meaningful to software designers [54]. Spalding and Tsai [52] outlined several phases of the software development lifecycle and proposed a set of strategies to help prioritize PbD in each of these phases. These strategies include interviewing users in the early stages to understand diverse user needs, working with cross-disciplinary teams during the middle stages to ensure effective communication, and conducting surveys to understand reasons for user abandonment [52].

Some studies have attempted to capture privacy and security requirements at the early stages of software development. A branch within such studies has aimed to extricate design principles from convoluted legal texts. For instance, Antón and Earp [2] analyzed a set of privacy policies and presented a taxonomy of privacy requirements so that designers could comprehend the policies and apply them to reduce privacy vulnerabilities of websites. Similarly, Maxwell et al. [39] proposed a legal cross-reference taxonomy to guide engineers in deriving privacy compliance requirements from laws and regulations that contain internal or external cross-references within the legal text. Another branch of research has focused on privacy risks associated with specific technologies. For example, Alqassem [1] proposed a framework to analyze the **Internet of Things** (**IoT**) user interaction in real time and take into account the complexity and unpredictable changes in the interaction that affect privacy and security. Contextually appropriate privacy requirements for mobile devices is another area that has received substantial research attention. For example, Thomas et al. [56] developed a problem analysis framework that helps extract and refine privacy requirements associated with mobile applications.

During the middle stages of software development, researchers have investigated the implementation of PbD to help developers actualize design ideas in code. For example, Rubinstein and Good [47] argue that privacy considerations should be included in the definition of software requirements and proposed the application of several engineering and usability principles in privacy design practices. Similarly, van Rest et al. [58] suggest that domain-specific applications would benefit from the use of design patterns grounded in PbD, such as privacy needs identification, vulnerability assessment, revocable privacy, and privacy statements. Likewise, Hoepman [27] derived eight privacy design strategies based on current data protection legislation to help with the design and evaluation of privacy-friendly software systems.

In the post-development stages, the core privacy-related tasks are gathering and analyzing user feedback on privacy-related matters [52]. Although it is valuable to understand user evaluation of the product for future improvement, the lack of effectiveness of such late consideration of privacy compliance is what prompted calls for PbD in the first place [9]. A core tenet of the PbD philosophy is that privacy compliance should be "Preventative, not Remedial" [8].

The work discussed above demonstrates a growing interest in employing PbD throughout software design, development, and testing. However, the strategies, tactics, and patterns provided in the literature are largely sets of higher-level guidelines. Although these guidelines are useful as a direction to pursue, they do not provide the vehicle to reach the destination. Moreover, there is a lack of empirical evidence examining how software developers consider and implement

privacy-related matters within various phases of the software lifecycle. Our study attempts to fill these gaps.

2.2 Ideation Cards

Ideation cards are practical design tools intended to stimulate reflection and creative thinking with the aim of helping to "define constrained design problems within a broader overall problem space" [23]. Ideation card activities have been used in multiple disciplines for various purposes, such as approaching problems from an indirect angle in creative dilemmas with cards for *Oblique Strategies* [18], attending to human values during the design process by using the *Envisioning Cards* [21], brainstorming security threats via the *Security Cards* [15], and raising awareness and altering perceptions of computer security by playing with cards in the *Control-Alt-Hack* game.

Luger et al. [36] explored the use of ideation cards for discussions related to data protection and privacy aspects of software. They developed a deck of ideation cards with one suit pertaining to privacy regulations. This "Regulation" suit contains cards that cover four major concepts in EU GDPR: (1) data breach notification, (2) explicit and informed consent, (3) the right to be forgotten, and (4) privacy by design. To contextualize privacy compliance and address other key factors in developing usable systems, the complete card deck includes three other suits: System, User group, and Constraint. Luger et al. [36] examined the use of these cards in a study that included 21 software professionals with varied backgrounds and specialties (e.g., user experience, software architecture, programming, and engineering) and between 1 and 16 years of industry experience. These participants were grouped into four teams of five or six members each. The teams used the ideation cards in a pre-defined system design scenario in which cards from each suit were drawn progressively within the session, with the Regulation cards drawn last [36]. As each card was drawn, the team discussed modifications to the system based on the content of the card. Based on these sessions, Luger et al. [36] found that ideation cards can serve as a useful prompt in the software design and development process to help improve regulatory compliance regarding privacy aspects. Discussions pertaining to privacy during these sessions revealed several themes, including different interpretations of the designers' roles in privacy compliance and the location of data storage [36].

However, Luger et al. [36] used carefully crafted scenarios specifically designed to impinge upon privacy. Therefore, there is a need to examine the extent to which their observations hold true in real-world software projects that are more diverse and complex in nature. Moreover, the participants in Luger et al. [36] were industry professionals. The extent to which novice developers, such as software students who are about to join the workforce, can use PICs remains to be studied.

2.3 Reflective Learning and Practice

The principal function of PICs is to "prompt and encourage reflection on aspects of data protection law" [36]. Reflection is a conscious learning activity that involves examining one's experiences, actions, and emotions and interpreting them in order to learn from them [5]. By critically reflecting on their assumptions, habits, and beliefs, professionals can transform their "taken-for-granted" frame of reference [5, 41] and draw out new knowledge and higher-order understanding. Therefore, reflection has been viewed as a vital component of professional practice and an integral part of higher-level thinking and learning [37].

Reflective learning has been a popular approach in project-based disciplines that involve artifact design and development, such as architecture and music composition. Reflection can facilitate a learning dialogue between the implicit and subconscious experience of an experiential learning activity and the explicit consideration of the concept in a conscious manner [31]. Hazzan [26] examined software engineering education practices and proposed reflective learning as a habit-of-mind

for software development. The justification for adopting a reflective practitioner's perspective in software engineering education stems from a dual perspective:

> "…from an architectural-design perspective and from an engineering perspective. The combination of these two points of view captures the development of software systems as a process which, on the one hand, is guided by creative thinking, and on the other hand, receives feedback from the engineering-scientific mode of thinking by pointing to reliability, complexity, and other engineering considerations" [26].

By adopting a reflective thinking mode, software designers and developers are better able to examine an issue of concern and change their conceptual perspectives and courses of actions [6].

Reflection is a cognitive process that can range from a lower or shallow level to a deeper or transformational level [25, 28, 33, 50]. The aim of developing and enhancing higher-order thinking in learning activities has been a major educational goal across disciplines, including computing education [38, 57]. Problematically, however, reflection is an internal cognitive process that cannot be directly observed. This raises the question of how to evaluate reflection and related practices. To address this problem, Leung et al. [33] proposed an analytical framework for reflective learning by drawing upon Schön's reflective learning theory [48], Resnick's higher-order thinking occurrences [46], and Donald's working model of higher-order learning [17]. The framework describes five levels of cognitive processes in reflective learning, with each process containing different learning tasks that signify cognitive involvement. From low to high cognitive levels, these five processes are as follows:

(1) **Meaning construction:** to frame the characteristics of an idea.
(2) **Interpretation:** to understand the meaning of an idea.
(3) **Change:** to change the current practices or perception.
(4) **Validation:** to verify the validity of ideas.
(5) **Generalization:** to develop a general inference for practices.

Privacy compliance is typically not a topic familiar to software students [2, 45]. As a stimulant for reflective learning, PICs are a promising thinking tool to expose students to a set of privacy-related concepts and considerations. When software students engage physically with PICs, they interact with the underlying concepts associated with the cards and ponder the relevance and implication of the cards through discussion [23]. We examined the reflective learning processes of meaning construction and mental reinvestment induced by the use of PICs. To that end, we adopted a modified version of Leung et al.'s [33] analytical framework to examine the aspects of learning that emerged in the ideation activity.

3 METHOD

Capstone projects allow students to put their skills to the test to solve real-world problems [59]. Therefore, capstone projects are instrumental in helping students develop the hard and soft skills required for their future careers. Those who sponsor these projects can make progress on lower-priority activities and gain access to a pool of graduating software developers for recruitment [22].

We addressed our research questions with an empirical investigation of student teams in a two-term capstone course in Informatics at the University of California, Irvine, a large public university in the United States. As a requirement for this course, students work in teams of four to six on projects provided by external sponsors, such as corporations, government organizations, educational institutions, and non-profits. For instance, the student teams we studied were creating an email scheduler, improving the **User Interface** (**UI**) design of a video game, developing a conversational application, and so on. These projects revolve around developing a software system, at

Table 1. Summary of the Student Teams who Participated in the Study Across the Three Iterations of the Study, Indicating Whether the Ideation Sessions were Conducted in the First or the Second Term of the Capstone Course

Academic year	Term of course	Number of teams	Number of students
2015–2016	Second (Winter 2016)	8	35
2016–2017	First (Fall 2016)	7	31
2017–2018	First (Fall 2017)	9	40
TOTAL		24	106

least at the level of a functional prototype or proof-of-concept, along with associated documentation such as requirements (in the form of use cases or user stories), designs (in the form of **Unified Modeling Language** [**UML**] diagrams or **UI/User eXperience** [**UX**] mockups), test cases, and test results. Most students who take the course have no prior real-world industry experience. However, approximately 15–20% of them have limited exposure to industry via internships. In essence, the capstone projects in the course provide teams with a supervised and controlled real-world experience analogous to that of an individual industry internship.

For the study, we developed a set of PICs for the U.S. regulatory context and used these cards to engage each team in an ideation session. During these sessions, the teams discussed the application of the drawn cards to their respective capstone projects. We conducted the ideation sessions in three different offerings of the course: (1) Fall 2015–Winter 2016; (2) Fall 2016–Winter 2017; and (3) Fall 2017–Winter 2018. The ideation sessions were conducted as a mandatory part of the course. However, it was optional to permit the use of the data for research. To avoid coercion, the course instructor was not involved in the administration of any study procedures and had no knowledge of whether a student consented to permit the session data to be used for research purposes. Similarly, the researchers were not involved with the course in any other way and had no knowledge of, or influence over, grades connected to the ideation session component of the course. All study procedures were reviewed and approved by the **Institutional Review Boards** (**IRBs**) of Indiana University and the University of California, Irvine.

We conducted the ideation sessions with the first student cohort in the second term of the course (i.e., Winter 2016). Upon initial data analysis, we discovered that the ideation sessions, although beneficial, could not induce a meaningful impact on the projects because they occurred too late within the project schedule. Therefore, we moved the subsequent ideation sessions to the first term of the course offerings (i.e., Fall 2016 and Fall 2017, respectively). Table 1 summarizes the information of student teams who participated in the three iterations of the ideation sessions, and Table 2 provides a brief description of each capstone project.

In the following subsections, we explain the development of the PICs used in the study, the protocol followed during the ideation sessions, and the data collection and analysis approach.

3.1 Development of Privacy Ideation Cards

Inspired by the work of Luger et al. [36], we used an equivalent deck of ideation cards (see Appendix), containing 29 cards in total. These cards are grouped under three themed suits: *User* (9), *Constraint* (11), and *Regulation* (9). The *User* and *Constraint* suits cover two regularly considered factors in the design process. They were included in the card deck to contextualize the discussion. Our PICs cover the U.S. regulatory context by selecting privacy guidelines from the **Office of Economic Co-operation and Development** (**OECD**) [44] and the Fair Information Practice Principles from the **Federal Trade Commission** (**FTC**) [20], which provide a basis for the U.S. regulatory framework. We consulted with privacy lawyers in the United States to use these

Table 2. Summary of the Student Team Projects Including the Academic Year in which the Project was Developed, Team Name, Project Nature (Frontend or Backend), and Description

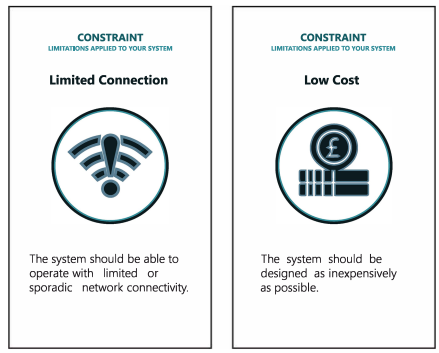
Academic Year	Team #	Project Nature	Project Description
2015–2016	1	Frontend	A mobile application that helps students browse available tasks and get recruited by companies.
	2	Frontend	A dashboard for electronics that collects data and displays it such that users can read it better and arrange it the way they want.
	3	Frontend & Backend	A cloud document management system to be integrated with other products.
	4	Frontend & Backend	A Chrome extension to clear all saved form data, such as browsing information and cookies, with a single button.
	5	Backend	A sales trend system that pulls data from various sources to check effectiveness of the sales strategies.
	6	Backend	A data exploration tool to gather user information, such as name, age, address, consumer information, and IP address, for a market aggregator.
	7	Frontend	A book recommendation page to be integrated in the university system where professors post recommended readings.
	8	Frontend	A form-building tool to be integrated in other products.
2016–2017	9	Frontend	A dashboard for a database provided by the company.
	10	Frontend	A video game UI, such as the player perspective, audio, and replay.
	11	Frontend	A dashboard for a marketing company to aggregate and track their analytics and marketing tools, including e-commerce and social media (e.g., Facebook and Instagram) data.
	12	Backend	A scheduling management system for healthcare providers.
	13	Frontend & Backend	A customized email scheduler for end users to receive a visualization of their dashboards.
	14	Frontend & Backend	A smart office assistant application to increase the productivity and efficiency of meetings (e.g., by tracking topics and progress).
	15	Backend	A web application that uses community user input of temperature data to improve the accuracy of weather prediction.
2017–2018	16	Backend	A system to help an instructor recruit project sponsors for courses.
	17	Frontend & Backend	A suite of mini games to help parents, educators, and medical professionals detect signs of early speech development delay based on children's interactions with the games.
	18	Backend	A real-time currency conversion tool.
	19	Frontend	A system that helps a bio-med company automate scheduling across its different campuses.
	20	Backend	A parking monitor system that uses a camera to detect the movement of cars in parking spots.
	21	Frontend	The sign-up function on the corporate website of a data provider.
	22	Frontend	The patient login function of a healthcare application.
	23	Backend	Improvements to the system for unsubscribing from email newsletters.
	24	Frontend	An application to train people regarding communication skills for job interviews by analyzing responses to various questions and scenarios.

guidelines for creating a set of *Regulation* cards comparable to those derived from the EU GDPR. Figure 1 shows two example cards from each of the three suits.

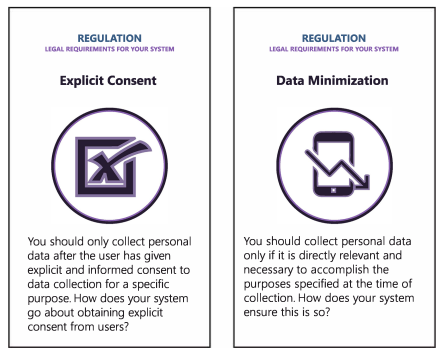
3.2 Ideation Session Protocol

In the ideation sessions, the teams were asked to apply the cards to their software projects. They drew cards from the deck and discussed the relevance of the cards to their projects. Figure 2 shows the general flow of activities during each ideation session. As instructed, each team drew one *User* card, one *Constraint* card, and two *Regulation* cards, in that order. However, we let the teams skip the drawn card if the team members found that it was not applicable or relevant to their

(a) *User* cards



(b) *Constraint* cards



(c) *Regulation* cards

Fig. 1. Examples of ideation cards: two *User* cards (orange), two *Constraint* cards (blue), and two *Regulation* cards (purple).

project. Time permitting, we allowed the teams to draw one or two more cards if the team members expressed interest in doing so. Based on the experience with the first two cohorts, we decided to have the teams in the third cohort draw two *Constraint* cards to produce more detailed and richer discussion. In the ideation session, the team members discussed each drawn card separately for about 5 minutes per card, ending with an approximately 10-minute discussion of all drawn cards considered together as a set. We provided the suggested duration for the discussion as such limits "can symbolize and facilitate the possibility for meaningful use in a brief amount of time" [21]. The facilitator's main responsibilities were to introduce the cards and the activity at the start of

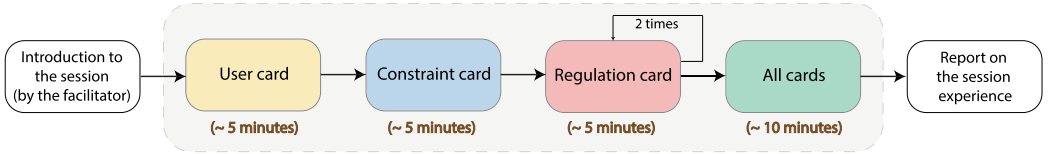


Fig. 2. Protocol of each ideation session, wherein the student team drew one *User* card, one *Constraint* card, and two *Regulation* cards as instructed and then discussed the application of the cards to the project.

Table 3. Summary of Cards That Each Team Drew in the Ideation Sessions

Team #	*User* Cards	*Constraint* Cards	*Regulation* Cards
1	Second Language	User Control	Breach Notification; Explicit Consent
2	Mental Health	Social Sharing; User Control*; Limited Connection	Subject Access; Data Minimization
3	Gender Spectrum	Minimal Distraction	Security; Notice
4	Mental Health	Limited Connection*; Social Sharing; User Control	Accountability*; Data Minimization*; Notice*
5	Older People	Social Sharing	Explicit Consent; Subject Access*
6	Second Language*; Ex-Offenders*; Visual Impairment	Low Cost; Low Energy	Security; Data Minimization*; Notice*; Purpose Specification
7	Mental Health	Social Sharing	Explicit Consent; Notice
8	Visual Impairment	Limited Connection	Notice; Purpose Specification
9	Mental Health	Careful Collection	Subject Access; Purpose Specification
10	Older People; Ex-Offenders	Data Minimization	Explicit Consent; Breach Notification
11	Gender Spectrum*; Second Language	Low Energy*; Social Sharing	Security; Purpose Specification
12	Second Language	Minimal Distraction	Notice; Data Minimization
13	Visual Impairment	User Control; Breach Notification	Breach Notification; Subject Access
14	Mental Health*; Visual Impairment	Minimal Distraction	Data Quality; Purpose Specification
15	Mental Health	Social Sharing	Explicit Consent; Security
16	Country of Residence	Low Energy*; Data Monetization*	Data Minimization*; Subject Access*
17	Mental Health; Second Language	Privacy; Minimal Distraction	Subject Access; Data Quality
18	Country of Residence*; Visual Impairment*; Children	Social Sharing*; Default Sharing*; Minimal Distraction; Careful Collection*	Data Quality; Breach Notification*; Data Minimization
19	Older People; Second Language	Minimal Distraction	Subject Access; Data Quality
20	Gender Spectrum*; Visual Impairment; Older People*; Second Language	Privacy; Low Cost	Data Minimization*; Purpose Specification; Data Quality; Explicit Consent
21	Second Language; Low Literacy	Data Minimization; Social Sharing*; Privacy	Subject Access; Accountability*; Notice
22	Gender Spectrum	Low Cost; Minimal Distraction	Data Minimization; Explicit Consent
23	Older People*; Mental Health	Default Sharing*; Low Cost	Purpose Specification*; Data Quality; Data Minimization*; Security
24	Visual Impairment	Data Minimization*; Low Cost	Subject Access; Security; Purpose Specification

* = Cards that were found inapplicable to the team projects.

the session and answer any procedural questions and monitor discussion time during the session. Table 3 lists the sets of cards that the teams drew in their respective ideation sessions. A researcher assisted by the last author acted as the facilitator for the first cohort while the last author was the facilitator for the latter two cohorts.

After the facilitator introduced the cards and the activity at the beginning of the ideation session, the team members engaged in a general discussion of their project. There were no designated topics for this discussion, and team members were free to talk about anything of their choice. For

example, some teams gave a brief overview of their project for the facilitator's benefit, and some others chose to use this time to organize their project-related tasks.

Following the general discussion, the facilitator instructed the teams to draw a *User* card from the shuffled, face-down card deck and discuss how their system might support the needs of the user group presented on the card. For example, a team who drew the `Older People` card proposed that this user base might be "not used to using technology all the time" so their system would have to display information in simpler terms.

After 5 minutes of discussing the *User* card, the facilitator instructed the teams to draw a *Constraint* card to discuss how their system might deal with the presented constraint. For example, a team who drew the `Low Energy` card considered making the system available offline so that the system could operate with low levels of power consumption when necessary.

Next, the facilitator instructed the teams to choose cards from the *Regulation* suit. Students engaged in a 5-minute discussion on how the regulatory concept of the first drawn card was related to their project and repeated the process with a second card. For example, a team who drew the `Explicit Consent` card decided to provide a checkbox for their users to confirm that they explicitly consented to data collection. In the rest of the article, we focus on the discussion stimulated by the *Regulation* cards. While *User* and *Constraint* cards were useful for contextualizing the ideation session, we exclude the data and results related to these cards as these are orthogonal to the scope of the research questions addressed in this article.

When the teams finished discussing the drawn cards individually, the facilitator asked the teams to have a 10-minute overall discussion considering *all* cards drawn during the session. Students took a broad look at the scenario as a whole and talked about what they needed to do to satisfy the various requirements stated in the drawn cards taken together. We video recorded all sessions and additionally took photographs of any artifacts, such as drawings, white board notes, and so on, produced by the student teams during the sessions.

One week after the activity, we asked the teams to submit a report on their ideation session experience, including a summary of any changes they planned to make to their projects based on the discussions that occurred during the ideation sessions. The changes could pertain to any aspects of the project, such as changing the user interface, adding components connected to privacy compliance (e.g., a notice to inform users of the purpose of data collection), raising privacy issues with their sponsors, and so on.

3.3 Data Collection and Analysis

We collected and analyzed data only for the sessions for which *all* team members consented to the research use of the data. Data was collected from two main sources: video recordings of the ideation sessions and post-session reports from the teams. Audio from the recordings was transcribed verbatim, and the video was used to add relevant contextual information, such as the specific card that was being discussed. For 24 student teams, all members (106 participants in total) consented to having their ideation sessions be part of the research, and 21 out of these 24 teams submitted post-session written reports.

We analyzed the transcripts and post-session reports using qualitative content analysis [29]. To answer RQ1 (i.e., *"What are the main considerations related to privacy compliance for real-world projects?"*), we adopted a bottom-up approach using techniques from grounded theory [13], allowing common themes to emerge from the data without *a priori* assumptions. Specifically, we adopted Straussian grounded theory [13, 55] and coded for themes as a team using an iterative process, starting with open coding, followed by axial and selective coding.

To answer RQ2 (i.e., *"To what extent do ideation cards help students reflect and learn the privacy aspects pertaining to real-world software?"*), we proceeded top-down and used the validated coding

Table 4. The Operational Coding Framework Used to Analyze Student Reflection in the Ideation Sessions, Including the 14 Observable Tasks That Correspond to One of Five Cognitive Processes, Along with Examples of Each Task from the Student Discussions of Regulation Cards

Cognitive Process	Code	Observable Task	Example
1. Meaning construction	MA	Ask a question.	"Is it just like securing the data?"
	MI	Identify relevant information.	"In terms of the scope of our project, the data is straight up from that database."
	MC	Connect information to personal experience and practice.	"Yeah; it's like when you have a Fitbit."
2. Interpretation	IS	Specify an important or controversial issue.	"Their location too would be important because we don't want people finding other people's addresses."
	IC	Compare with familiar norms, ideas, or practices.	"Google just tells you the highlights, like in three bullet points."
	IP	Propose a hypothesis or assumption.	"They probably already have used it before."
3. Change	CR	Revise or replace a previous idea or practice.	"So basically we have to lock every option until we see a signature."
	CS	Suggest a new idea or practice without making concrete plans.	"We keep the database simple."
	CA	Adopt a new idea or practice with plans to implement.	"We could do some sort of referral system where the physician gives you a referral code to enter when you first login."
4. Validation	VG	Agree with the information provided.	"Yeah, something like that sounds like a good idea."
	VI	Identify flaws in, or express doubt over, information provided.	"I don't think our filter handles any of those."
	VS	Assess knowledge and practice.	"I think the idea of security was there because we'd been working with IP addresses, but we couldn't figure out how."
5. Generalization	GD	Draw a conclusion based on research or experience.	"So basically, step 1, they create the account. …Step 2, they are in …and then they can upload photos."
	GP	Plan to apply or look into the information provided.	"I think we should talk to [sponsor] about it as well as consider the terms and agreements."

framework provided by Leung et al. [33], with the following few modifications to tailor it to our study:

(1) We operationalized the code "Connect information to experience and practice" to mean "Connection information to personal experience and practice" because students described their personal experience anecdotally in relation to the card drawn.
(2) We combined the original codes "Revise an idea or practice" and "Replace an idea or practice" as they were not distinctively separable in our data.
(3) We extended the definition of code "Identify flaws in the information provided" by adding "express doubt over" to capture situations in which students were unsure about the information provided.
(4) We removed the code "Apply information provided in other contexts" because we did not observe any such task in our data.

Table 4 presents our adjusted coding framework and provides examples from the data to illustrate the codes.

The first two authors independently applied the above modified framework to code a random sample of 10% of the data and compared the results of the individual coding. The two coders discussed and resolved all discrepancies until they reached consensus. The second author then coded the rest of the data, supervised by the first author. We identified 14 tasks, each connected to one of five cognitive processes. The second author then applied the 14 codes corresponding to each of these tasks to all ideation session transcripts, yielding a total of 2,642 coded segments within the transcripts. We engaged in a second coding pass to check the results for accuracy and consistency. For this article, we used the 636 coded segments pertaining to the *Regulation* cards. We do not report the remaining coded segments as they related to the *User* and *Constraint* cards.

For a quantitative comparison of the extent to which the teams responded to different *Regulation* cards, we used the number of lines in the respective transcripts to measure how much the teams discussed each card. We operationalized a line as one interactive round, where a team member continued the discussion by building on the ideas of others or proposing a new idea. The number of lines is a reasonable metric because it shows the level of engagement independent of the influence of other factors, such as differences in talking speed and pauses in interaction. Specifically, for every card, we counted the number of lines of discussion that were about that card in each transcript and added them across all transcripts. We then divided the sum by the number of times the card was drawn across all sessions to yield the average number of lines prompted by the card.

4 FINDINGS

The analyses mentioned in the previous section enabled us to evaluate the application of PICs as a tool for the design and development of real-world software and a pedagogical instrument for teaching privacy-related matters to software students. We discuss each aspect in turn.

4.1 RQ1: Privacy Compliance in Real-World Projects

Application of PICs to real-world software projects surfaced three main considerations about privacy compliance. Specifically, developers need to determine (1) the connection of their system with various kinds of privacy-affecting data, (2) their responsibilities regarding the collected data, especially if they are a third party receiving data collected by another party, and (3) their autonomy in proposing and implementing privacy-related functionalities. The following subsections cover each aspect in detail.

4.1.1 Defining Personal Data. In line with the principles from which they were derived, the *Regulation* cards use the term "personal data" to refer to the data that is subject to privacy regulations and compliance. The OECD defines personal data as "any information relating to an identified or identifiable individual (data subject)" [44]. However, what data was considered personal, and thus subject to the regulations presented on the cards, was often unclear to the students. For example, in *Team 8*'s discussion, one member argued that "name, or even gender associated with name" could be sensitive and should be treated as potential personal data, while another member felt personal data is only those pieces of information that are really sensitive, such as "bank account information." In another case, members of *Team 20*, which was building a parking monitor with a camera hanging at the top to see a parking spot, claimed that their system did not proactively collect any personal data:

> "We did not intentionally collect who you are, what your name is, how old you are …just care about the car." – (*Team 20*)

However, as the discussion went on, one team member changed his opinion:

> "There is a personal perspective to it because it has the license plate and people's faces on it." – (*Team 20: Student 5*)

These examples show that the teams often struggled to reach a consensus on what should be treated as personal data. In many cases when they were indecisive, their definitions focused more on the repercussions of potential data breaches rather than the nature of the information that makes data subjects identifiable.

4.1.2 Assigning Responsibility for Privacy Compliance. The projects often used data from multiple sources, including that collected independently by external services. In such cases, team members were often confused about whether they should be held responsible for data protection and whether the regulations were applicable to the data made accessible to them as a third party. For example, *Team 4* built their browser extension using a paid service by Google and decided that the `Data Minimization` and `Notice` cards could not be applied to the project. Instead, the team members felt that Google assumed the responsibility to adhere to these regulations.

In another example, *Team 5* was building a sales trend system that did not collect data directly from people but acquired its data from various third-party sources, such as social media services. Although such data included personally identifiable information, the team members quickly asserted that the `Subject Access` card did not apply to their project because they were not the active data collection agent. *Team 5* responded similarly to the other *Regulation* card, `Explicit Consent`:

> "Well, the data that we have has been obtained from other people so they've already have gotten their permission. You would think that if it is collecting data about them, they've already given explicit consent. It's like Ticketmaster. You give your consent as you're using Instagram, as you're using Facebook, they're collecting that data from Facebook and Instagram." – (*Team 5*)

In general, the teams that were not the ones directly collecting the information rarely decided on proactive measures to comply with the privacy regulations and seldom chose to have further discussions over related issues, even though they were using the data actively in their projects.

4.1.3 Determining and Exercising Autonomy. The teams developed the projects based on the requirements provided by the sponsors. We noticed that the teams working in a large organizational context were more limited in their decisions, while those working on more independent projects tended to generate more concrete plans and make relevant changes to respond to the privacy regulations.

Some teams (e.g., *Teams 3, 7*, and *21*) were developing test-version projects to be incorporated within the respective company's existing infrastructure and systems and felt that they lacked sufficient autonomy, control, or knowledge to suggest and influence changes to the provided requirements. For example, *Team 3*, which was developing a cloud document management system to be integrated with the other products of the company, commented on the `Security` card:

> "We don't know their [host site] architecture so we don't know what protections they have against like data loss or whatever." – (*Team 3*)

As a result, the team members decided that they need not worry about the matter because security was a matter to be addressed by others in the sponsor's company.

On the contrary, the teams dealing with standalone systems (e.g., *Teams 1, 9*, and *22*) seemed to have much more autonomy over privacy compliance measures. For example, members of *Team 22*

Table 5. Summary of Interaction Related to Each *Regulation* Card Across All Teams in the Three Sessions, Including the Sum of the Number of Lines in the Transcripts Where the Students Discussed a Specific *Regulation* Card, the Number of Times the Card was Drawn, and the Mean Number of Lines in the Transcripts Covering Discussions of the Card (Sorted from Highest to Lowest)

***Regulation* Card**	**Total Number of Lines**	**Number of Ideation Sessions**	**Mean Lines per Ideation Session**
Explicit Consent	288	7	41.1
Subject Access	303	9	33.7
Breach Notification	134	4	33.5
Security	150	6	25.0
Purpose Specification	197	8	24.6
Notice	167	7	23.9
Data Quality	124	6	20.7
Data Minimization	175	9	19.4
Accountability	37	2	18.5
TOTAL	1,575	58	27.2

discussed the `Explicit Consent` card by pondering over their roles as developers and actively performing a critical examination of their plans.

> "Essentially it is already being covered in the current design. The waiver that they sign is the one that [the sponsor] is actually getting updated by lawyers right now. So in a weird way, we are covered in that a lawyer has written that part for us. [The sponsor] is the one pushing for it. We just have to ensure that, especially since it is on our end and not the other parts of the app that the user is signing, no data can be acquired until that waiver is signed. So we have to basically lock every option until we see a signature." – (*Team 22*)

These examples illustrate that levels of perceived and actual autonomy in the development process can influence the extent to which developers will engage with privacy aspects, including compliance.

4.2 RQ2: Reflection and Learning Supported by Privacy Ideation Cards

To understand the impact of PICs on learning about privacy compliance in software development, we compared how students responded to the different PICs and examined the various cognitive processes stimulated by the PICs during the ideation sessions.

4.2.1 Responses to Different Regulation Cards. The mean number of lines of team discussion for each *Regulation* card varied from 18.5 to 41.1 (see Table 5), with a mean of 27.2 across all cards. The `Explicit Consent` card induced the most extensive discussion (41.1 lines on average). The teams discussed whether they had included a user consent element in the design, whether they should seek explicit consent before or after user login, and whether consent was applicable if they received data originally collected by another service. On the other hand, the `Accountability` card generated the least amount of discussion (18.5 lines on average). In the two cases when this card was drawn, one team decided this card was not applicable because the project did not collect any personal data (*Team 4*), while the other team was unable to understand the meaning of the card (*Team 21*). The discussions regarding the other cards fell somewhere in between these two ends.

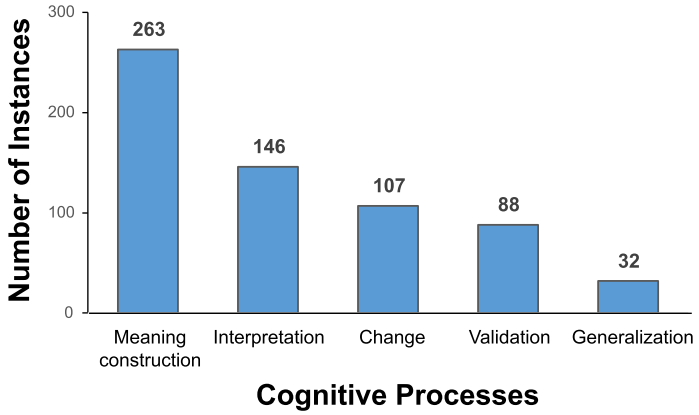


Fig. 3. Distribution of cognitive processes across all teams that participated in the ideation sessions. The processes of "Meaning construction" and "Interpretation" were dominant.

4.2.2 Distribution of Tasks Across the Cognitive Processes. We identified a total of 636 observable instances of 14 cognitive tasks covering five types of cognitive processes that occurred during the ideation session discussions related to the *Regulation* cards. Among the five cognitive processes connected to these tasks, students engaged primarily in the surface-level cognitive process of "Meaning Construction" ($n = 263, 41.4\%$). The highest-order cognitive process, "Generalization," was observed the least ($n = 32, 5\%$). The disproportionate focus on the lower-level tasks indicates that the ideation sessions led the teams to consider framing the basic concepts and characteristics of privacy compliance for the software they were developing, but the students found it difficult to generalize the concepts further to proceed to higher-order tasks, such as proposing concrete plans to change current practices, verifying the validity of their perceptions, and extrapolating to other scenarios. Figure 3 shows the distribution of the five cognitive processes across all 24 teams that participated in the ideation sessions.

In terms of specific tasks, the top three represented in the data were "MI: Identify relevant information" ($n = 121, 19\%$), "MA: Ask a question" ($n = 118, 18.6\%$), and "CS: Suggest a new idea or practice without making concrete plans" ($n = 91, 14.3\%$). The least prevalent cognitive tasks were "CR: Revise or replace a previous idea or practice" ($n = 5, 0\%$), "GD: Draw a conclusion based on research or experience" ($n = 10, 1.6\%$), and "CA: Adopt a new idea or practice with plans to implement" ($n = 11, 1.7\%$). Figure 4 presents the distribution of the 14 tasks reflective of the different cognitive processes.

The majority of surface-level cognitive tasks were performed when students were trying to make sense of the concepts related to the *Regulation* cards and determine the characteristics of the underlying ideas. When the teams drew a *Regulation* card, the team members asked each other about its meaning and confirmed whether the card was applicable to their projects. For example, *Team 19* discussed the `Subject Access` card, trying to figure out the exact requirements associated with the card.

— *Team 19: Student 1*: "Okay, so Subject Access. System should provide means for establishing the existence and nature of any personal data held about a data subject, the purpose of its use, and the identity of the data controller. It should let the users have access to their data."
— *Team 19: Student 2*: "So, is it just like securing the data?"

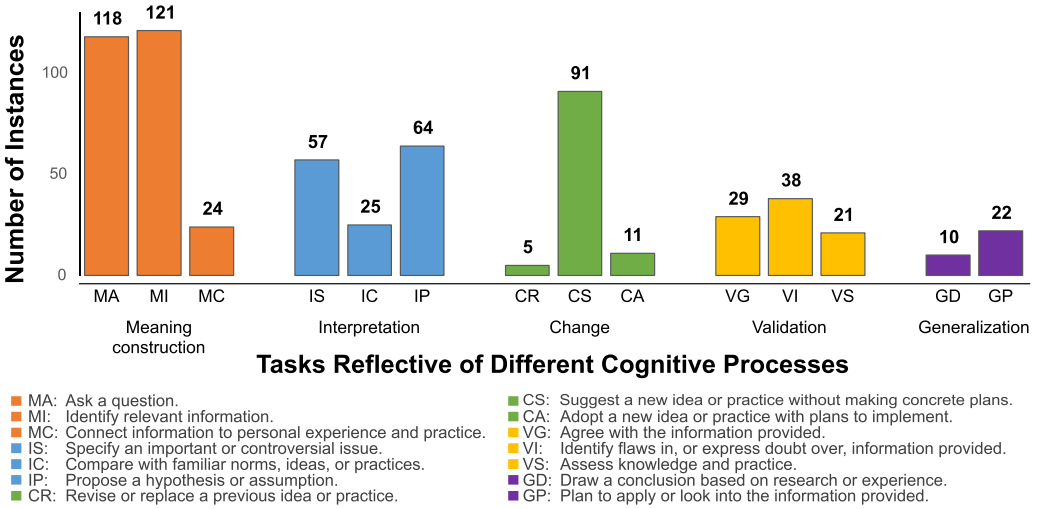


Fig. 4. Distribution of the tasks reflective of cognitive processes across all teams that participated in the ideation sessions. The most frequently occurring tasks were "MI: Identify relevant information," "MA: Ask a question," and "CS: Suggest a new idea or practice without making concrete plans."

— *Team 19: Student 1*: "Oh, so just like you should know why you're storing the data that you're storing, pretty much? Like you shouldn't be storing random information about the users…"

According to GDPR, subject access allows individuals to request and receive a copy of all personal data that a company or organization has collected about them. Yet, this team seemed to confuse it with the principles of data security and purpose specification.

In another example, *Team 13* expressed confusion over the `Breach Notification` card, questioning why users would need to be notified. The team members seemed to believe that users might not care about their data anymore once it was handed over to the organization. They felt that notifying the users about a data breach was not as important as notifying the organization's administration. As a result, the students were not concerned about complying with the principle, as they did not grasp the rationale and utility of doing so. This indicates a general lack of understanding and appreciation for the basic values and concepts of privacy compliance.

— *Team 13: Student 1*: "Breach notification. You're required to inform users of data breaches, loss damage, or illicit access without undue delay."
— *Team 13: Student 2*: "Letting people know or figuring out if there was a breach?"
— *Team 13: Student 1*: "Yeah, there was a breach…"
— *Team 13: Student 2*: "Why does it matter? It's not like anyone signs up for it (laughs). Seriously, though, like the only person it would have to notify is the admin."

Despite the above challenges, the PICs did inspire the student teams to make adjustments and actively comply with several regulations. However, the majority of the changes were proposed without concrete implementation plans, as reflected in the code "CS: Suggest a new idea or practice without making concrete plans." For example, immediately after *Team 23* drew the `Data Quality` card, one student asserted that the solution would be to "keep the data simple," but the team did not continue on to suggest any actionable changes to their system to keep the data simple. Another example is the response of one member from *Team 1* to the `Breach Notification` card: "If there was a breach then the only possible way would be emailing. We have an email service." Yet, there

was no further conversation about how to operationalize this idea and actively use the emailing system to generate and deliver breach notifications.

On rare occasions, teams were able to propose changes with tangible plans. For example, *Team 12*, which was developing a scheduling management system for healthcare providers, engaged in a long discussion about the `Notice` card. The team members decided to follow the privacy standards of the **Health Insurance Portability and Accountability Act** (**HIPAA**) and implement the change with a concrete plan of "encrypting the data."

> "Yeah, we'll need to obfuscate all data held in the database so that if anybody gets the database dump, all our information is obfuscated, and we'll probably need to look at encrypting the traffic to and from the cloud so that people aren't intercepting packets." – (*Team 12*)

Overall, most discussions of the *Regulation* cards remained at the surface level of cognitive engagement. Higher-level cognitive processes were comparatively rare in the ideation sessions. Since the discussions were limited by the scope of the projects, it is perhaps to be expected that generalization tasks were underrepresented, as these typically require experience and reflection that occur over a longer term. During the ideation sessions, the majority of reflection focused on meaning construction and interpretation, with limited occurrences of changing or consolidating existing ideas (see Figure 3). The prevalence of lower-level cognitive tasks indicates a general lack of basic understanding and consideration of privacy compliance in the initial project design to begin with. In other words, students were unfamiliar with privacy regulations and principles pertaining to the systems they were building and/or did not take privacy compliance into active consideration during requirements gathering and subsequent design generation. As a result, they typically lacked any *existing* experiences or practices that they could criticize or consolidate. On the other hand, the cards inspired the teams to rethink their systems and prompted changes in their original approaches, as evidenced by the practices of "suggesting a new idea or practice without making concrete plans" or "adopting a new idea or practice with plans to implement."

5 DISCUSSION

The motivation behind our research was to understand how PICs could be applied as a privacy compliance tool for real-world software projects and as a pedagogical instrument to improve privacy proficiency of software students as novice developers who are about to join the workforce. Our findings show that PICs are a promising tool for engaging software professionals with issues of privacy compliance in their projects. At the same time, the findings point to a range of considerations for successful application of PICs within software development processes. Pedagogically, PICs induce surface-level cognitive processes, at least at the first attempt.

5.1 Privacy Compliance in Real-World Projects

Unlike the simulated design scenarios used by Luger et al. [36], we examined the application of PICs in **real-world** software projects. Luger et al. [36] intentionally constructed their scenarios to impinge upon legal and ethical issues regarding data use. In our study, the software projects were not fictitious, but assigned by external sponsors to address real-world problems independent of our research. Our exploration revealed that the application of PICs in-the-wild differs from that in pre-crafted scenarios in several notable ways.

The features and requirements of real-world software projects are framed by the in situ properties of the system and shaped by software industry practices. As a result, in contrast to simulated design scenarios created specifically with the PICs in mind, not all *Regulation* cards might be applicable to a given project. For example, *Team 9*, which developed a dashboard for an

existing database, did not need to collect any data directly. Therefore, the *Regulation* cards drawn by the team, `Subject Access` and `Purpose Specification`, were not directly applicable to the development activities for the project. In addition, in a typical real-world industry project, developers are not the only, or even the main, decision makers for privacy compliance matters. As our study shows, developer autonomy is greatly influenced by the characteristics of the project and the nature of the organization. The teams working on a more independent system had more freedom to respond to the regulations, while the teams developing a system with multiple dependencies with other systems in the organizations tended to consider themselves as mere bystanders in matters of privacy compliance. A lack of sufficient autonomy can prevent developers from being motivated to consider and engineer privacy compliance in the systems they develop, thus undermining the core vision of PbD. PICs can help address this issue by serving as tool that can increase the engagement of software engineers in discussions of privacy implications of the software they build.

Ambiguity regarding data classification poses a challenge when applying PICs in real-world projects. The PICs we used did not include a specific definition for the term "personal data," relying instead on a shared understanding of the term as understood in common practice. While a definition is essential from a regulatory point of view, we found that it was difficult to disentangle *personal* information from *sensitive* information from a software developer's point of view. Some considered the term personal data as being applicable only to highly sensitive pieces of data, such as social security numbers, medical records, or banking information. Many were unsure whether personal data covers seemingly benign pieces of information, such as name, gender, age, and so on. While it is reasonable that sensitive personal data is afforded greater protection, many privacy-related regulations require special treatment for any information that can reveal someone's identify. As such, personal data can cover a wide variety of information, such as name, location, online identifiers, license plates, biometrics, and so on. Our study shows that a lack of knowledge and agreement over personal data can get in the way of more nuanced discussions regarding privacy implications, thus affecting privacy compliance.

It was unclear who was responsible for privacy compliance when data was accessed as a third party. For instance, many teams seemed to be relieved that they collected no data themselves, instead using the data provided by other services and platforms, such as Google and Twitter. Since they did not proactively collect or store any user data themselves, they considered themselves not responsible for privacy compliance aspects. However, as those handling the data, third-party software developers are not free from the liability. Using third-party data may even complicate the situation as system developers need to know more details about collection and compliance processes used by the first party that obtained the data.

In terms of scheduling the ideation session, students found the PICs activity particularly useful at the beginning of the project. Due to logistical constraints, the first iteration of the ideation sessions was carried out in the second term of the course, 2 weeks before code freeze. Most students reported that it was simply too late to introduce any of the privacy-related ideas spurred by the sessions into design and development. Based on the experience of the first cohort, we moved the ideation sessions in the other two iterations to the middle of the first term, just after the teams received their project details and started early work on their plans. Students in the latter two iterations considered the timing beneficial for them to take privacy into consideration early on.

> "It's good since we have not actually started coding, it's good for us to start thinking about all the things." – (*Team 21*)

Aligned with the philosophy of PbD, scheduling the ideation sessions at the early stages of software development prods software engineers to adopt proactive measures to address privacy considerations in the development process. For example, when discussing the `Privacy` card, *Team 17* proposed the addition of a consent form:

> "If you're asking for information when they sign up for it, that's collecting stuff too. Because right now we have their names, their birthdays, their contact information . . . that's big stuff. And the fact that they are participating in this . . . if that stuff is tied together. So to go back to this all you really need is a consent form, preferably an IRB [Institutional Review Board] form that promises that it's going to be private." – (*Team 17*)

5.2 Learning about Privacy via the Reflective Ideation Activity

We systematically examined student engagement in the ideation session as a *reflective learning* opportunity. We found that ideation cards were successful in triggering lower-level cognitive processes for specific components within a project. As Figure 3 shows, 81.1% of the 636 cognitive tasks we observed fell into the lower-level categories (i.e., Meaning construction, Interpretation, and Change). A closer look at the the distribution of specific codes and the content of the interaction suggests that students had not previously considered privacy compliance issues at all. As a result, the PICs prompted them to spend a considerable amount of time discussing and negotiating the meaning of each card and confirming whether the card was relevant to their projects in the first place.

These findings suggest that PICs are a promising pedagogical tool to expose student software developers to knowledge about privacy regulations as an initial step for improving the privacy properties of their software. This observation further confirms the gap between advocating the importance of privacy requirements and the practical application of these requirements in real-world software development. To many developers, privacy consideration come second to ensuring that software is functional. In addition, project sponsors and managers might view privacy considerations as the purview of legal compliance teams and/or designated privacy managers, so they may not task developers with privacy requirements [45]. As a result, software engineers are neither trained to keep privacy in mind when developing software nor equipped with sufficient knowledge to understand and comply with up-to-date regulations. PICs can help address this issue by providing a learning opportunity for students and sponsors, thus advancing the PbD agenda.

When comparing the amount of interaction generated by each *Regulation* card, we noted that the `Explicit Consent`, `Subject Access`, and `Breach Notification` cards ranked at the top, while `Accountability`, `Data Minimization`, and `Data Quality` spurred the least discussion. In discussions spurred by the PICs, students often drew upon their previous experiences as *users* of particular technologies (e.g., Google apps). For instance, students from *Team 1* realized that they could apply the `Explicit Consent` card when users sign up, "like how people have the terms and regulations." They continued that explicit consent "is for protecting the company more. . . We need the legal protection. We have to have the explicit consent of 'I have signed this waiver. By signing here, I say I have read this and I understand this.'" The discussions during the ideation sessions indicated that students leverage implicit knowledge of privacy compliance measures that are observable in the systems and services they use. Practical examples based on the UX of widely used systems can serve as a lead-in topic that instructors can use to make privacy compliance more relatable by building upon the prior knowledge of the students as *users*. On the other hand, cards like `Accountability` and `Data Minimization` are associated with background processes that are not as directly accessible to students in their everyday use of technology. Consequently, students may find these cards vague and hard to decode.

To comply with privacy regulations, a common strategy adopted by the students was emulating the operation of mainstream services. It is undeniable that studying and following established practices and standards provides easily accessible solutions to tackle issues. Inexperienced developers, in particular, are more likely to be influenced by the background set by mainstream services and consider it as the correct way to comply with privacy regulations. However, this

practice can be problematic if students follow these practices unconditionally without reflection, even when they notice problems with typical privacy compliance measures. For example, one member of *Team 1* commented: "It's funny that people don't read it [the privacy consent] but they just check it anyway. Even for Facebook and Snapchat…no one reads it. They just press okay, you can take all my photos." The reliance on using familiar practices as standard approaches points to the need for privacy-related critique of common design patterns prior to incorporating them within other projects.

6 IMPLICATIONS

We discuss the implications of our findings from two perspectives: software engineering education and PIC design.

6.1 Implications for Software Engineering Education

Based on the analysis of the ideation sessions, we propose the following measures to facilitate the use of PICs as a pedagogical instrument to help software students learn about privacy-related practices.

Hold an information session at the beginning. For promoting more higher-order discussion over PICs in ideation sessions, we propose holding a brief information session or workshop to go over key privacy principles prior to the ideation sessions. Such an introduction could include a set of definitions that clarify the key terms related to the PICs as well as examples of typical privacy compliance practices in industry.

Provide PICs as a reference tool. Because of the time constraints in our ideation sessions, each team could draw only four to six cards in total. The discussion therefore focused mainly on limited topics directly related to the drawn cards. Moreover, due to the specifics of each project, some *Regulation* cards were not applicable. Students wished to see the other cards to gain a more comprehensive consideration of the compliance requirements. Accordingly, we suggest providing the entire set of ideation cards as a reference tool for capstone courses so that students can utilize them throughout the project as needed.

Repeat ideation sessions. When students received the initial instructions for the projects, they began by clarifying requirements and planning the development steps. Applying PICs at early stages of the projects may have contributed to the lower-level discussion of the meaning and applicability of the PICs in the projects. It could be beneficial to conduct the ideation sessions multiple times during the capstone course to capture different stages of the projects. Along with the ideation sessions, instructors could assign a privacy-focused deliverable. For instance, such a deliverable could mimic privacy impact assessments that are being used increasingly in relation to technology [43, 62]. Further, the task related to such a deliverable can serve as a self-directed learning opportunity in which students externalize their thoughts and document the improvement in their understanding and practices for developing privacy-compliant software.

Adopt active strategies to facilitate discussion. In order to prompt more higher-order reflection, we suggest that the facilitator provide guidance and clarification during the ideation sessions. This could help students better differentiate seemingly similar cards. In addition, some facilitation strategies, such as introducing real-world cases, inviting questions, asking for clarification, suggesting comparisons, prompting summary, and so on, can promote more higher-order thinking in reflective discussion [42, 60]. In this regard, the facilitator could play a role akin to that of a product manager or a privacy manager.

Incorporate privacy in software engineering education. Privacy regulations and principles are becoming increasingly vital for personal and corporate use of technology. Yet, current software engineering curricula lack adequate training for these purposes. Therefore, in the long run, there is a pressing need for privacy-related curricular content in software engineering education. Such curricula can ensure that students will be equipped with basic privacy proficiency prior to entering the workforce. Research on privacy-related education has thus far focused mostly on *user* education by challenging and correcting misconceptions that guide their online behavior [11, 32, 61]. However, educating software professionals on privacy-related matters is equally important because they are the ones who design and develop the systems that must be privacy compliant and serve the privacy needs and expectations of the users.

Communicate proactively with sponsors about privacy compliance. To improve privacy protection within their projects, some teams in our study planned to bring up privacy-related matters with their sponsors.

> "There is a lot we don't know like a lot of stuff that he [the sponsor] takes care of in the backend that we don't really touch, so it could be good just to have that conversation with him to start." – (*Team 21*)

Working within an organization necessarily restricts the freedom of developers to work as they please. However, the organizational context should not be a barrier that prevents software engineers from designing proactively for privacy compliance. Without empowering developers with the knowledge and autonomy they need to make privacy-related design decisions, PbD will remain disconnected from the development stage of the software lifecycle in practice. To address this issue within capstone courses, we advocate greater communication and transparency between student teams and their industry sponsors regarding the handling of privacy-related matters. In particular, the relationship between the teams and their sponsors should emphasize shared interest and responsibility regarding privacy compliance.

6.2 Implications for PIC Design

In addition to perceptions of the ideation activities, the reports submitted by the teams after the ideation sessions pointed to several suggestions for PIC design.

Make the cards aesthetically pleasing and fun. The aesthetics of the cards and the game element involved in their use contributed to the students enjoying use of the cards as stimulative props for reflection. For instance, the students found the color schemes and icons attractive ("The cards are really cute." "The icon is very good."). The inclusion of the game element (i.e., card playing) made the learning activity enjoyable and relatable as the students had fun drawing and examining the cards ("I almost feel like it's a game." "It's like we are playing a game."). As one student commented, "It feels natural to start talking about these cards."

Avoid lengthy descriptions and jargon. Most prominently, the students suggested rewording the text on some *Regulation* cards for greater clarity:

> "They [the cards] were kind of wordy, some of them. The ones with multi-line sentences were hard to understand . . . like the `Subject Access` one. I don't know what this is saying. You read it like three times." – (*Team 17*)

There is a tension between sufficiently capturing complex regulatory nuance while keeping the text short and easily understandable by a lay person. Clearer and shorter descriptions could reduce the cognitive burden of understanding the cards. It would also be worthwhile to refine

the descriptions through an iterative process that incorporates student feedback. In fact, the card design process could itself be educational by serving to engage software students in learning about privacy aspects of technology and helping develop an appreciation for the PbD approach.

Include illustrative examples. Supporting examples to illustrate application of the concept connected to a card could help enhance comprehension. The students sometimes failed to differentiate the nuance between different cards because they lacked prior exposure to the basic concepts presented in the ideation cards. For instance, the students complained that some cards presented overlapping concepts, thus coming across as repetitive. For example, a student argued that the `Notice` and `Purpose Specification` cards cover a similar concept.

> "I noticed that the two cards we happened to get are fairly similar: `Notice` and `Purpose Specification`. Both go with the idea that you have to be explicit and make sure the user knows what the information is and what it's being used for. So those kind of went hand in hand." – (*Team 8*)

In cases such as the one mentioned above, illustrative examples could help students grasp nuanced differences and become familiar with basic privacy-related concepts. However, providing examples could have a priming effect that prevents a full examination of the matter. Therefore, if examples are included, they should be chosen carefully to avoid constraining people's thought processes.

7 LIMITATIONS AND FUTURE WORK

Our study has the following main limitations. First, we were able to conduct the ideation session only once during each offering of the two-term course because of logistical constraints. Therefore, our results might not present a comprehensive view of privacy compliance as an *ongoing* practice throughout the software development lifecycle. Future studies could repeat the ideation sessions multiple times during a project and examine the changes across the stages in privacy compliance proficiency and processes.

Second, participants in the study were university students. While the capstone course facilitated an investigation of PICs as a pedagogical tool, the students were not yet industry professionals. Future studies with industry professionals could help understand the generalizability of PICs as a training tool for experienced developers.

Third, we could not investigate how the student teams implemented the proposed changes in their projects due to confidentiality concerns of the sponsors. Future studies should study how ideas generated in the ideation sessions influence properties of the code and outcomes of the projects.

Fourth, we used time-limited discussion to focus participant attention and facilitate meaningful discussion in a brief amount of time based on an established protocol of ideation card activities (e.g., [18, 21]). Since complex cognitive tasks typically require extra processing time, the time limit may, however, create constraints that could inhibit students from moving to a higher-order cognitive thinking. Future studies could investigate whether an extended discussion time induces higher-order thinking in ideation activities and examine how the cognitive processes unfold over time.

8 CONCLUSION

Inspired by the philosophy of PbD, we developed and tested a suit of PICs in real-world undergraduate software engineering projects to support student learning of privacy concepts related to technology. By applying the PICs in ideation sessions in three offerings of a capstone course, we found that PICs foster greater student consideration and understanding of the extent to which their software design and implementation align with contemporary privacy regulations. We

demonstrate that PICs are a promising resource that helps fill privacy-related knowledge gaps among undergraduate software students. In addition, PICs can be an effective training tool to promote strategic privacy-related learning among professionals. Application of PICs in real-world software development can enhance privacy compliance and promote greater use of PbD principles in the software industry.

APPENDICES

A IDEATION CARDS: *USER*

Users: The people who use your system.

Number	Card	Description
1	Older People	Your users might be age 65 or older.
2	Mental Health	Your users might suffer from poor mental health.
3	Children	Your users might be children or adolescents.
4	Visual Impairment	Your users might be blind or have other visual impairments that could impact their use of the system.
5	Poor Literacy	Your users might be adults with low levels of literacy.
6	Gender Spectrum	Your users might identify with any of a range of different genders.
7	Ex-Offenders	Your users might be ex-offenders and might include people on probation.
8	Second Language	Your users might understand English as a second or third language at varying levels of fluency.
9	Country of Residence	Your users might reside in a country other than your own.

B IDEATION CARDS: *CONSTRAINT*

Constraints: Limitations applied to your system.

Number	Card	Description
1	Default Sharing	The system should allow for default sharing with third parties.
2	Limited Connection	The system should be able to operate with limited or sporadic network connectivity.
3	Social Sharing	The system should engage with third-party social media services.
4	Low Cost	The system should be designed as inexpensively as possible.
5	Data Maximization	The system should collect as much data as possible about the user.
6	Privacy	The system should enhance a user's perception of privacy.
7	Minimal Distraction	The system should not distract the user from primary goals or day-to-day activities.
8	Data Minimization	The system should collect as little data as possible.
9	Careful Collection	The system should not collect (by accident or intention) any third-party data.
10	Low Energy	The system should rely on low levels of power or energy to operate.
11	User Control	The system should allow the user a high level of control.

C IDEATION CARDS: *REGULATION*

Regulations: Legal requirements for your system.

Number	Card	Description
1	Notice	You should provide notice to users about what data is to be collected, how it will be used and disseminated, and how it will be maintained. How will your system do that?
2	Explicit Consent	You should only collect personal data after the user has given explicit and informed consent to data collection for a specific purpose. How does your system go about obtaining explicit consent from users?
3	Purpose Specification	The purposes for which personal data are collected should be specified at the time of data collection, and you should use the data collected only for the purposes specified. How does this impact your system?
4	Data Minimization	You should collect personal data only if it is directly relevant and necessary to accomplish the purposes specified at the time of collection. How does your system ensure this is so?
5	Data Quality	Personal data should be relevant to the purposes for which it is to be used, accurate, complete, and up to date. How does your system ensure that data quality is maintained?
6	Subject Access	Systems should provide means for establishing the existence and nature of any personal data held about a data subject, the purpose of data use, and the identity of the data controller. How does your system make this possible?
7	Security	Personal data should be protected by reasonable security safeguards against such risks as loss, unauthorized access, destruction, use, modification, disclosure, and so on. What security issues might affect your system, and how does your system address those issues?
8	Breach Notification	You are required to inform users of data breaches (loss, damage, or illicit access) without undue delay. What measures do you have in place for such a scenario?
9	Accountability	A data controller should be accountable for complying with the measures you have in place for protecting personal data of users. How will your system support accountability?

ACKNOWLEDGMENTS

We would like to thank Ewa Luger and Mike Golembewski for insight regarding the use of the privacy ideation cards. We thank Hadar Ziv for providing us access to student teams in the capstone course at the University of California, Irvine. We acknowledge the help of Lesley Fosh for conducting the ideation sessions and Emma Lashley for data processing and organization. We are grateful to the anonymous reviewers who provided constructive feedback that helped improve the manuscript. The content of the article is the work of the authors and does not necessarily reflect the views of the sponsors.

REFERENCES

[1] Israa Alqassem. 2014. Privacy and security requirements framework for the Internet of Things (IoT). In *Companion Proceedings of the 36th International Conference on Software Engineering (ICSE Companion 2014)*. Association for Computing Machinery, New York, NY, 739–741. https://doi.org/10.1145/2591062.2591201

[2] Annie I. Antón and Julia B. Earp. 2004. A requirements taxonomy for reducing web site privacy vulnerabilities. *Requirements Engineering* 9, 3 (Aug. 2004), 169–185. https://doi.org/10.1007/s00766-003-0183-z

[3] Brooke Auxier, Lee Rainie, Monica Anderson, Andrew Perrin, Madhu Kumar, and Erica Turner. 2019. Americans and Privacy: Concerned, Confused and Feeling Lack of Control Over Their Personal Information. Retrieved June 14, 2021 from https://www.pewresearch.org/internet/2019/11/15/americans-and-privacy-concerned-confused-and-feeling-lack-of-control-over-their-personal-information/.

[4] France Bélanger and Robert E. Crossler. 2011. Privacy in the digital age: A review of information privacy research in information systems. *MIS Quarterly* 35, 4 (Dec. 2011), 1017–1042.

[5] David Boud, Rosemary Keogh, and David Walker. 2013. *Reflection: Turning Experience into Learning* (1st ed.). Routledge.

[6] Evelyn M. Boyd and Ann W. Fales. 1983. Reflective learning: Key to learning from experience. *Journal of Humanistic Psychology* 23, 2 (1983), 99–117. https://doi.org/10.1177/0022167883232011 arXiv:https://doi.org/10.1177/0022167883232011

[7] California Civil Code. 2018. TITLE 1.81.5. California Consumer Privacy Act of 2018 [1798.100 - 1798.199.100]. Retrieved June 30, 2021 from https://leginfo.legislature.ca.gov/faces/codes_displayText.xhtml?division=3.&part=4.&lawCode=CIV&title=1.81.5.

[8] Ann Cavoukian. 2011. Privacy by Design: The 7 Foundational Principles. Retrieved June 14, 2021 from https://iapp.org/media/pdf/resource_center/pbd_implement_7found_principles.pdf.

[9] Ann Cavoukian. 2012. Privacy by design. *IEEE Technology and Society Magazine* 31, 4 (2012), 18–19. https://doi.org/10.1109/MTS.2012.2225459

[10] Gauthier Chassang. 2017. The impact of the EU general data protection regulation on scientific research. *ecancermedicalscience* 11, 709 (2017), 12 pages. https://doi.org/10.3332/ecancer.2017.709

[11] Erika Chin, Adrienne Porter Felt, Vyas Sekar, and David Wagner. 2012. Measuring user confidence in smartphone security and privacy. In *Proceedings of the 8th Symposium on Usable Privacy and Security (SOUPS'12)*. Association for Computing Machinery, New York, NY, Article 1, 16 pages. https://doi.org/10.1145/2335356.2335358

[12] Luca Compagna, Paul El Khoury, Alžběta Krausová, Fabio Massacci, and Nicola Zannone. 2009. How to integrate legal requirements into a requirements engineering methodology for the development of security and privacy patterns. *Artificial Intelligence and Law* 17, 1 (2009), 1–30. https://doi.org/10.1007/s10506-008-9067-3

[13] Juliet Corbin and Anselm Strauss. 2014. *Basics of Qualitative Research: Techniques and Procedures for Developing Grounded Theory* (4th ed.). SAGE Publications, Inc.

[14] Lorrie Faith Cranor and Norman Sadeh. 2013. A shortage of privacy engineers. *IEEE Security Privacy* 11, 2 (2013), 77–79. https://doi.org/10.1109/MSP.2013.25

[15] Tamara Denning, Batya Friedman, and Tadayoshi Kohno. 2013. The Security Cards: A Security Threat Brainstorming Toolkit. Retrieved June 30, 2021 from https://securitycards.cs.washington.edu/index.html.

[16] Tamara Dinev, Paul Hart, and Michael R. Mullen. 2008. Internet privacy concerns and beliefs about government surveillance—An empirical investigation. *The Journal of Strategic Information Systems* 17, 3 (2008), 214–233. https://doi.org/10.1016/j.jsis.2007.09.002

[17] Janet Gail Donald. 2002. *Learning To Think: Disciplinary Perspectives. The Jossey-Bass Higher and Adult Education Series* (1st ed.). Jossey-Bass Inc., San Francisco, CA.

[18] Brian Eno and Peter Schmidt. 1975. Oblique Strategies. Retrieved June 30, 2021 from http://www.rtqe.net/ObliqueStrategies/.

[19] European Parliament and Council of the European Union. 2016. Regulation (EU) 2016/679 of the European Parliament and of the Council of 27 April 2016 on the Protection of Natural Persons with regard to the Processing of Personal Data and on the Free Movement of such Data, and Repealing Directive 95/46/EC (General Data Protection Regulation). Retrieved June 30, 2021 from https://eur-lex.europa.eu/eli/reg/2016/679/oj.

[20] Federal Trade Commission (FTC) Division of Financial Practices, Bureau of Consumer Protection. 2000. Privacy Online: Fair Information Practices in the Electronic Marketplace: A Report to Congress. Retrieved June 30, 2021 from https://www.ftc.gov/sites/default/files/documents/reports/privacy-online-fair-information-practices-electronic-marketplace-federal-trade-commission-report/privacy2000.pdf.

[21] Batya Friedman and David Hendry. 2012. The envisioning cards: A toolkit for catalyzing humanistic and technical imaginations. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI'12)*. Association for Computing Machinery, New York, NY, 1145–1148. https://doi.org/10.1145/2207676.2208562

[22] Jay R. Goldberg, Vikram Cariapa, George Corliss, and Kate Kaiser. 2014. Benefits of industry involvement in multidisciplinary capstone design courses. *International Journal of Engineering Education* 30, 1 (2014), 6–13.

[23] Michael Golembewski and Mark Selby. 2010. Ideation decks: A card-based design ideation tool. In *Proceedings of the 8th ACM Conference on Designing Interactive Systems (DIS'10)*. Association for Computing Machinery, New York, NY, 89–92. https://doi.org/10.1145/1858171.1858189

[24] Seda Gürses, Carmela Troncoso, and Claudia Diaz. [n.d.]. Engineering privacy by design. In *Conference on Privacy & Data Protection*, Vol. 14. 25 pages. Issue 3.

[25] Marina Harvey, Debra Coulson, and Anne McMaugh. 2016. Towards a theory of the ecology of reflection: Reflective practice for experiential learning in higher education. *Journal of University Teaching & Learning Practice* 13, 2 (2016), 20 pages. https://ro.uow.edu.au/jutlp/vol13/iss2/2/.

[26] Orit Hazzan. 2002. The reflective practitioner perspective in software engineering education. *Journal of Systems and Software* 63, 3 (2002), 161–171. https://doi.org/10.1016/S0164-1212(02)00012-2

[27] Jaap-Henk Hoepman. 2014. Privacy design strategies. In *ICT Systems Security and Privacy Protection*, Nora Cuppens-Boulahia, Frédéric Cuppens, Sushil Jajodia, Anas Abou El Kalam, and Thierry Sans (Eds.). Springer, Berlin, 446–459.

[28] Tríona Hourigan and Liam Murray. 2010. Using blogs to help language students to develop reflective learning strategies: Towards a pedagogical framework. *Australasian Journal of Educational Technology* 26, 2 (Apr. 2010), 209–225. https://doi.org/10.14742/ajet.1091

[29] Hsiu-Fang Hsieh and Sarah E. Shannon. 2005. Three approaches to qualitative content analysis. *Qualitative Health Research* 15, 9 (2005), 1277–1288. https://doi.org/10.1177/1049732305276687 PMID: 16204405.

[30] Keith S. Jones, Akbar Siami Namin, and Miriam E. Armstrong. 2018. The core cyber-defense knowledge, skills, and abilities that cybersecurity students should learn in school: Results from interviews with cybersecurity professionals. *ACM Transactions on Computing Education* 18, 3 (Aug. 2018), Article 11, 12 pages. https://doi.org/10.1145/3152893

[31] Richard Jordi. 2011. Reframing the concept of reflection: Consciousness, experiential learning, and reflective learning practices. *Adult Education Quarterly* 61, 2 (2011), 181–197. https://doi.org/10.1177/0741713610380439.

[32] Iacovos Kirlappos and M. Angela Sasse. 2012. Security education against phishing: A modest proposal for a major rethink. *IEEE Security & Privacy* 10, 2 (2012), 24–32. https://doi.org/10.1109/MSP.2011.179

[33] Kit H. Leung, Pierre Pluye, Roland Grad, and Cynthia Weston. 2010. A reflective learning framework to evaluate CME effects on practice reflection. *Journal of Continuing Education in the Health Professions* 30, 2 (2010), 78–88. https://doi.org/10.1002/chp.20063

[34] Fan Liang, Vishnupriya Das, Nadiya Kostyuk, and Muzammil M. Hussain. 2018. Constructing a data-driven society: China's social credit system as a state surveillance infrastructure. *Policy & Internet* 10, 4 (2018), 415–453. https://doi.org/10.1002/poi3.183

[35] Nikola Luburić, Goran Sladić, Jelena Slivka, and Branko Milosavljević. 2019. A framework for teaching security design analysis using case studies and the hybrid flipped classroom. *ACM Transactions on Computing Education* 19, 3 (Jan. 2019), Article 21, 19 pages. https://doi.org/10.1145/3289238

[36] Ewa Luger, Lachlan Urquhart, Tom Rodden, and Michael Golembewski. 2015. Playing the legal card: Using ideation cards to raise data protection issues within the design process. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems (CHI'15)*. Association for Computing Machinery, New York, NY, 457–466. https://doi.org/10.1145/2702123.2702142

[37] Karen Mann, Jill Gordon, and Anna MacLeod. 2007. Reflection and reflective practice in health professions education: A systematic review. *Advances in Health Sciences Education* 14, 4 (2007), 595–621. https://doi.org/10.1007/s10459-007-9090-2

[38] Robert J. Marzano. 1993. How classroom teachers approach the teaching of thinking. *Theory Into Practice* 32, 3 (1993), 154–160. https://doi.org/10.1080/00405849309543591

[39] Jeremy C. Maxwell, Annie I. Antón, and Peter Swire. 2011. A legal cross-references taxonomy for identifying conflicting software requirements. In *2011 IEEE 19th International Requirements Engineering Conference*. 197–206. https://doi.org/10.1109/RE.2011.6051647

[40] Roger McNamee. 2019. A Brief History of How Your Privacy Was Stolen. Retrieved June 6, 2021 from https://www.nytimes.com/2019/06/03/opinion/google-facebook-data-privacy.html.

[41] Jack Mezirow. 1997. Transformative learning: Theory to practice. In *New Directions for Adult and Continuing Education*. Number 74. Jossey-Bass Publishers, 5–12.

[42] Barak Miri, Ben-Chaim David, and Zoller Uri. 2007. Purposely teaching for the promotion of higher-order thinking skills: A case of critical thinking. *Research in Science Education* 37, 4 (2007), 353–369. https://doi.org/10.1007/s11165-006-9029-2

[43] Marie Caroline Oetzel and Sarah Spiekermann. 2014. A systematic methodology for privacy impact assessments: A design science approach. *European Journal of Information Systems* 23, 2 (2014), 126–150. https://doi.org/10.1057/ejis.2013.18

[44] Organisation for Economic Co-operation and Development (OECD). 2013. The OECD Privacy Framework. Retrieved June 30, 2021 from https://www.oecd.org/sti/ieconomy/oecd_privacy_framework.pdf.

[45] Paul N. Otto and Annie I. Antón. 2007. Addressing legal requirements in requirements engineering. In *15th IEEE International Requirements Engineering Conference (RE'07)*. 5–14. https://doi.org/10.1109/RE.2007.65

[46] Lauren B. Resnick and Committee on Research in Mathematics, Science, and Education in the Commission on Behavioral and Social Sciences and Education of the National Research Council. 1987. Education and learning to think. (1987), 72 pages.

[47] Ira S. Rubinstein and Nathaniel Good. 2013. Privacy by design: A counterfactual analysis of Google and Facebook privacy incidents. *Berkeley Technology Law Journal* 28 (2013), 1333–1414.

[48] Donald A. Schön. 1990. *Educating the Reflective Practitioner: Toward a New Design for Teaching and Learning in the Professions* (1st ed.). Jossey-Bass Inc., San Francisco, CA.

[49] Stuart S. Shapiro. 2010. Privacy by design: Moving from art to practice. *Communications of the ACM* 53, 6 (June 2010), 27–29. https://doi.org/10.1145/1743546.1743559

[50] Chaklam Silpasuwanchai, Xiaojuan Ma, Hiroaki Shigemasu, and Xiangshi Ren. 2016. Developing a comprehensive engagement framework of gamification for reflective learning. In *Proceedings of the 2016 ACM Conference on Designing Interactive Systems (DIS'16)*. Association for Computing Machinery, New York, NY, 459–472. https://doi.org/10.1145/2901790.2901836

[51] Christopher Slobogin. 2008. *Privacy at Risk: The New Government Surveillance and the Fourth Amendment*. University of Chicago Press.

[52] Kerry Spalding and Tsai Janice Y. 2016. Practical strategies for integrating privacy by design throughout product development process. In *Workshop on Bridging the Gap between Privacy by Design and Privacy in Practice at the 2016 CHI Conference on Human Factors in Computing Systems*. New York, NY, 4 pages. Retrieved June 30, 2021 from https://networkedprivacy2016.files.wordpress.com/2015/11/practicalstrategiespbd_cameraready.pdf.

[53] Sarah Spiekermann. 2012. The challenges of privacy by design. *Communications of the ACM* 55, 7 (July 2012), 38–40. https://doi.org/10.1145/2209249.2209263

[54] Luke Stark, Jen King, Xinru Page, Airi Lampinen, Jessica Vitak, Pamela Wisniewski, Tara Whalen, and Nathaniel Good. 2016. Bridging the Gap between privacy by design and privacy in practice. In *Proceedings of the 2016 CHI Conference Extended Abstracts on Human Factors in Computing Systems (CHI EA'16)*. Association for Computing Machinery, New York, NY, 3415–3422. https://doi.org/10.1145/2851581.2856503

[55] Klaas-Jan Stol, Paul Ralph, and Brian Fitzgerald. 2016. Grounded theory in software engineering research: A critical review and guidelines. In *Proceedings of the 38th International Conference on Software Engineering (ICSE'16)*. Association for Computing Machinery, New York, NY, 120–131. https://doi.org/10.1145/2884781.2884833

[56] Keerthi Thomas, Arosha K. Bandara, Blaine A. Price, and Bashar Nuseibeh. 2014. Distilling privacy requirements for mobile applications. In *Proceedings of the 36th International Conference on Software Engineering (ICSE'14)*. Association for Computing Machinery, New York, NY, 871–882. https://doi.org/10.1145/2568225.2568240

[57] Scott Alexander Turner, Manuel A. Pérez-Quiñones, and Stephen H. Edwards. 2018. Peer review in CS2: Conceptual learning and high-level thinking. *ACM Transactions on Computing Education* 18, 3 (Sept. 2018), Article 13, 37 pages. https://doi.org/10.1145/3152715

[58] Jeroen van Rest, Daniel Boonstra, Maarten Everts, Martin van Rijn, and Ron van Paassen. 2014. Designing privacy-by-design. In *Privacy Technologies and Policy*, Bart Preneel and Demosthenes Ikonomou (Eds.). Springer, Berlin, 55–72.

[59] Jari Vanhanen, Timo O. A. Lehtinen, and Casper Lassenius. 2012. Teaching real-world software engineering through a capstone project course with industrial customers. In *2012 First International Workshop on Software Engineering Education Based on Real-World Experiences (EduRex'12)*. 29–32. https://doi.org/10.1109/EduRex.2012.6225702

[60] Susan Wilks. 1995. *Critical & Creative Thinking: Strategies for Classroom Inquiry*. Heinemann.

[61] Pamela J. Wisniewski, Bart P. Knijnenburg, and Heather Richter Lipford. 2017. Making privacy personal: Profiling social network users to inform privacy education and nudging. *International Journal of Human-Computer Studies* 98 (2017), 95–108. https://doi.org/10.1016/j.ijhcs.2016.09.006

[62] David Wright. 2011. Should privacy impact assessments be mandatory? *Communications of the ACM* 54, 8 (Aug. 2011), 121–131. https://doi.org/10.1145/1978542.1978568

Received September 2020; revised April 2021; accepted May 2021