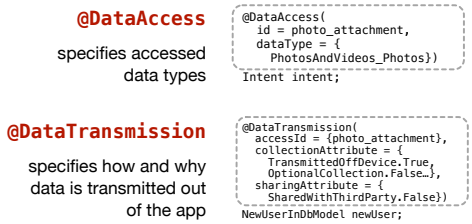
# RESEARCH STATEMENT

Tianshi Li

My research in **human-computer interaction (HCI), privacy, and software engineering** is on designing systematic, effective, and usable developer support for privacy. With the proliferation of computing systems that rely on personal data, developers are facing increasing responsibilities to protect user privacy throughout the software development life cycle. While we often take it for granted that developers should handle all the tasks well, most developers are not privacy experts and are already overloaded with other more salient requirements such as functionality and performance. Unfortunately, existing developer support for privacy is limited, fragmented and ad-hoc, which increases the implementation and maintenance cost for privacy and deepens the awareness and knowledge barriers.

My research aims to offer **one coherent solution for multiple privacy requirements** by augmenting the **integrated development environment (IDE)** with **plugins** to offer easy-to-access privacy support, and requiring developers to add **privacy annotations** in their code. With one set of annotations, my tools offer privacy support in multiple aspects, including 1) detection of sensitive API calls and third-party SDKs to support accurate understanding, documentation, and disclosure of data practices, 2) just-in-time reminders and lightweight code repair features (quick-fixes) to help developers conform to best practices, and 3) annotation-based declarative programming to generate in-app privacy notices and privacy nutrition labels required by app stores.



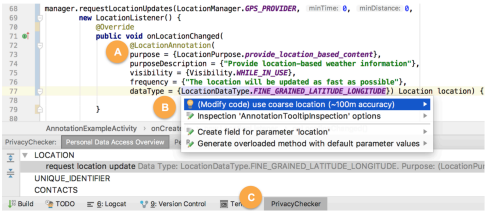

**Privacy annotations** are key elements that support **my IDE plugins for privacy**. By asking developers to provide one set of privacy annotations, my plugins help them increase knowledge about privacy and fulfill privacy requirements with less work.

As an interdisciplinary scholar, I do research to tackle complex social problems (e.g., privacy) from *problem understanding* using a human-centered approach, to *problem solving* using a technical approach. I conduct empirical studies using methods including interviews, surveys, content analysis, and large-scale data analysis to study the challenges developers face with regard to privacy. I then design, build, evaluate, and deploy a series of IDE plugins to tackle these problems. My research has yielded publications at top-tier HCI and privacy venues, such as CHI, UbiComp, CSCW, TOCHI, and PETS. My work also makes a broader impact through tool release and outreach to developer communities, industry, and policymakers. I have successfully collaborated with researchers in both academia (CMU, Stanford, George Washington University, University of Edinburgh, University of Bristol) and industry (Google). My work started out as part of the DARPA Brandeis project which aimed to dramatically improve smartphone privacy and was later funded by NSF as well. It was also supported in part by a CMU CyLab Presidential Fellowship. My work has been recognized with an ACM CHI best paper honorable mention award and I was named an EECS Rising Star in 2022.

## INCREASING AWARENESS AND KNOWLEDGE OF PRIVACY

As I set out on my research about helping developers with privacy in 2017, most work on privacy focused on end-users and surprisingly little research looked into how to leverage developers' capacity to better improve privacy for users. Some prior literature examined developers' security attitudes and built developer tooling for security, while paying little attention to practical tooling for addressing privacy issues. In my UbiComp 2018 paper [1], I identified **three crucial challenges** via semi-structured interviews with Android developers: lack of ability to consider all privacy principles in the system development process, unawareness of existing resources that provide best privacy practices, and misunderstandings about their apps' data practices due to unexpected data collection of third-party libraries and turnover among development teams.

These issues suggest the need for **tooling to keep developers better engaged in protecting user privacy**, which inspired my first work, Coconut, an IDE plugin to help developers build privacy-friendly Android apps [1]. Coconut detects sensitive API calls and helps developers add privacy annotations. It offers real-time privacy suggestions based on the annotations. The privacy overview panel aggregates information from annotations to help developers manage all data practices in one place. My developer study showed that adding privacy annotations were perceived as easy and helpful. The privacy suggestions effectively reduced excessive data use based on Android best practices. The privacy overview panel helped developers write more accurate privacy policies. We have open-sourced Coconut[1].




A Developers add **privacy annotations** by filling out a skeleton annotation added with the help of Coconut.

B Developers use the **Coconut quick-fix** to modify their code to conform to the best practices (using coarse-grained location for getting location-based weather information)

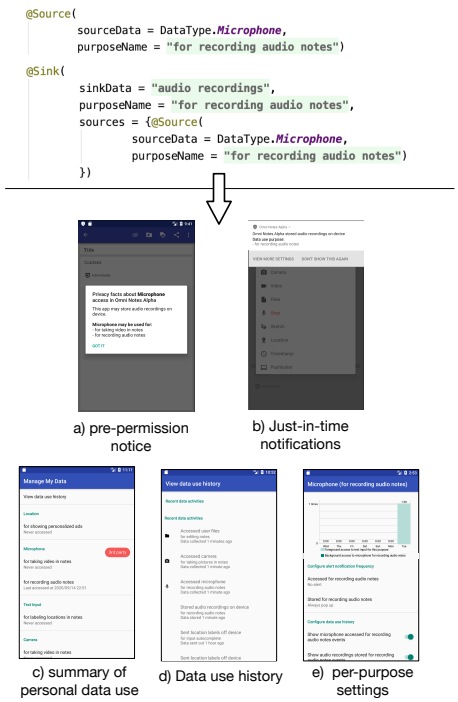
C Developers use the **privacy overview panel** to audit their data use.

## REDUCING BURDEN FOR HANDLING PRIVACY REQUIREMENTS

To further inspect developers' attitudes and practices regarding privacy in a natural condition when not explicitly prompted about the concept, I conducted qualitative content analysis studies of privacy-related discussions on online developer communities [2, 3]. My research showed that developers' online discussions about privacy are largely driven by the need to comply with legal and platform requirements for privacy. However, **developers felt these requirements were more burdensome than beneficial**, and had trouble understanding why they were helpful to enhance privacy.

Informed by these findings, I looked at a specific category of privacy requirements – implementing privacy notices. Recent years have seen an emerging body of legislature (e.g., GDPR, CCPA) and app store platforms (e.g., Apple, Google) that require developers to create various types of privacy notices to improve data transparency. However, it takes a lot of effort for developers to keep pace with the ever-changing legal and policy requirements, because it would be distributed all over the code and developers do not know best design practices for the privacy notice UIs.

Towards this end, I designed and built Honeysuckle [4], a developer tool that draws on the classic idea of declarative programming, **allowing developers to create in-app privacy notices by annotating details about data usage** at the data access and transmission instances. Honeysuckle consists of three key components: an IDE plugin subsystem developed atop Coconut that directly interacts with developers, a build system (Gradle) plugin that inserts the generated code at compile time, and a privacy notice library whose functions are called by the generated code. In my lab studies, developers created in-app privacy notices much faster with a significantly lower cognitive load using annotation-based code generation than calling the library functions. Honeysuckle allows developers without expert knowledge to create




Developers add **privacy annotations**, and Honeysuckle **converts** them **to a variety of of in-app privacy notices**

[1] https://coconut-ide.github.io

privacy notices following the best practices with little extra work. Having developers focus on what rather than how, Honeysuckle demonstrates **a viable path to promote standard privacy notice designs**, making it easier for users to compare privacy practices across apps.

In addition to building tooling for privacy requirements, I also seek to better support developers by **making an impact on policy making**. In 2020, My colleague at Stanford and I submitted comments to the California Attorney General's Office for the February round of the California Consumer Privacy Act (CCPA) rulemaking process[2]. Our suggestions emphasize the important role that platforms play in educating, supporting, and auditing developers to create proper privacy notices on mobile devices.

## LEVERAGING DOMAIN-SPECIFIC KNOWLEDGE TO IMPROVE PRIVACY

While developers are oftentimes not privacy experts, they are the experts of their apps. Their domain-specific expertise plays an important role in handling many privacy requirements. For example, Apple and Google recently introduced their own designs of privacy nutrition labels to their mobile app stores to provide a clear, uniform, and brief summary of data practices. Both platforms consider developers alone are responsible for reporting accurate data practices to create those labels. However, my CHI 2022 papers uncovered prevalent inaccuracies of the self-reported privacy labels [5, 6]. Among eight apps that already had a privacy label, six were re-created inconsistently during our study. I found that the developers' misunderstandings of the label terminology and the third-party SDKs' data practices **impede them from leveraging their app-specific knowledge to create accurate privacy labels**.

Hence, I designed, built, and evaluated Matcha [7], an IDE plugin that **leverages the synergies between developers' knowledge and source code analysis** to create accurate privacy nutrition labels for Android apps. Matcha was developed atop Coconut and Honeysuckle. By detecting sensitive API calls that may access and transmit personal data, Matcha guides developers to add privacy annotations to complement important details such as purposes and backend storage practices that cannot be inferred automatically. Matcha then translates the annotations to the label. My lab studies with Android developers working on a real-world app that they developed showed that Matcha improved the accuracy of data safety labels than using the Google Play console, and all developers preferred Matcha to the developer console. We have open-sourced Matcha[3] and released it on JetBrains official plugin marketplace and have seen the plugin gaining adoption by real-world developers.


My research on helping developers create accurate privacy nutrition labels have **attracted attention from many circles**. I presented my work to the ACT association, which is a global trade association for small and medium-sized technology companies. As a result of my presentation, the ACT association has released an open letter[4] to the platforms to call for attention to the issues identified by my research. I presented my work to researchers and practitioners at Apple and Google that work on privacy labels and have started to cause improvements in their designs. I was also invited to present my work on privacy nutrition labels to the staff at FTC, with the hope of generating more policy impact.

[2] https://cyberlaw.stanford.edu/blog/2020/02/ccpa-comments-round-two-battle-do-not-sell-button
[3] https://matcha-ide.github.io
[4] https://actonline.org/2022/10/14/an-open-letter-to-platforms-regarding-privacy-nutrition-labels/

## FUTURE RESEARCH AGENDA

I plan to apply my core strengths in HCI and building systems to support the development of safe, trustworthy and respectful software systems. I hope to collaborate with people in security and privacy, software engineering, PL, and AI/ML to tackle the problem by drawing on ideas from different fields.

**Towards holistic privacy support for mobile app development.** Privacy in mobile app development still faces many challenges, including the needs for better privacy controls and support for auditing. *How can we engage and support developers to tackle these challenges?* For my short-term goal, I plan to use privacy annotations and design better tooling to help developers build mobile apps that natively support privacy notices and controls, and to make it easier to audit the apps' data use. I seek to expand annotation-based privacy notice generation to privacy control generation; build systems to support end-to-end auditing with both privacy annotations for frontend data collection and backend data storage; and use static and dynamic program analysis to help different entities audit annotated apps.

**Privacy developer support for emerging technologies.** Emerging technologies, such as AR/VR, IoT, and blockchain, present new challenges for developing privacy-preserving apps. *How can we help developers easily build compelling and privacy-preserving apps with these technologies?* To this end, I plan to categorize threats into general threats that are consistent with mobile apps and unique threats about specific technologies. Then, I aim to design developer support to tackle the corresponding issues, drawing on my prior research on mobile app developer support. For example, I plan to transfer annotation-based privacy notice generation from mobile apps to VR apps. In addition, I intend to study the re-identification risks of collecting body motion data that do not apply to mobile apps, and designing tooling that supports privacy-preserving body motion data collection for VR app developers.

**Democratizing PETs.** Many privacy-enhancing technologies (PETs) such as differential privacy hold great promise for protecting user privacy, yet they remain technically challenging for developers to integrate and conceptually unintuitive for both developers and users to make sense of. *How can we empower average developers to take advantage of these technologies and build usable privacy support for end-users?* From a developer's perspective, this requires more abstraction and tooling support to lower the technical barriers. From a user's perspective, this requires in-depth understanding of how those PETs align with their privacy needs. In my future research, I plan to promote the adoption of PETs by addressing needs from both the developers and users, and my rich experience in both developer-centered [1, 2, 3, 4, 5, 6, 7, 8] and user-centered research [9, 10, 11, 12] can help with this goal.

**Developer support for "X".** Nowadays, developers are tasked with meeting crucial requirements that they may have limited knowledge of. Privacy is one such example. As a long-term goal, I aim to explore this question: *How can we further support developers to deal with other important requirements that are outside of their primary goal and domain expertise?* For instance, accessibility is also a pressing issue in mobile apps that developers face similar challenges for. Developers are not accessibility experts, they have limited knowledge about accessibility best practices, and they have limited time to tackle the various accessibility needs from different users. Hence, I seek to apply similar ideas, such as giving developers real-time accessibility checks on their user interface designs within the IDE and generating interfaces that can adapt to different input/output modality preferences.

**Conclusion.** To make responsible use of technologies a common practice, it is necessary to have effective support for developers. In my research, I design and build systematic, effective, and usable developer support for privacy. In the future, I plan to collaborate with researchers in different areas of computer science to help developers build a safe, trustworthy, and respectful digital world.

## References


[1] **Tianshi Li**, Yuvraj Agarwal, and Jason I. Hong. Coconut: An IDE plugin for developing privacy-friendly apps. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, 2(4):1–35, 2018.

[2] **Tianshi Li**, Elizabeth Louie, Laura Dabbish, and Jason I. Hong. How developers talk about personal data and what it means for user privacy: A case study of a developer forum on reddit. *Proceedings of the ACM on Human-Computer Interaction*, 4(CSCW3):1–28, 2021.

[3] Mohammad Tahaei, **Tianshi Li**, and Kami Vaniea. Understanding privacy-related advice on stack overflow. *Proc. Priv. Enhancing Technol.*, 2022(2):114–131, 2022.

[4] **Tianshi Li**, Elijah B Neundorfer, Yuvraj Agarwal, and Jason I. Hong. Honeysuckle: Annotation-guided code generation of in-app privacy notices. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, 5(3):1–27, 2021.

[5] **Tianshi Li**, Kayla Reiman, Yuvraj Agarwal, Lorrie Faith Cranor, and Jason I. Hong. Understanding challenges for developers to create accurate privacy nutrition labels. In *CHI Conference on Human Factors in Computing Systems*, pages 1–24, 2022. **Best Paper Honorable Mention**.

[6] Yucheng Li, Deyuan Chen, **Tianshi Li**, Yuvraj Agarwal, Lorrie Faith Cranor, and Jason I. Hong. Understanding ios privacy nutrition labels: An exploratory large-scale analysis of app store data. In *CHI Conference on Human Factors in Computing Systems Extended Abstracts*, pages 1–7, 2022.

[7] **Tianshi Li**, Lorrie Faith Cranor, Yuvraj Agarwal, and Jason I. Hong. Matcha: An IDE plugin for creating accurate privacy nutrition labels. *(under review)*.

[8] Mohammad Tahaei, Kopo M Ramokapane, **Tianshi Li**, Jason I. Hong, and Awais Rashid. Charting app developers' journey through privacy regulation features in ad networks. *Proceedings on Privacy Enhancing Technologies*, 1:24, 2022.

[9] **Tianshi Li**, Camille Cobb, Jackie Yang, Sagar Baviskar, Yuvraj Agarwal, Beibei Li, Lujo Bauer, and Jason I. Hong. What makes people install a covid-19 contact-tracing app? understanding the influence of app design and individual difference on contact-tracing app adoption intention. *Pervasive and Mobile Computing*, page 101439, 2021.

[10] **Tianshi Li**, Cori Faklaris, Jennifer King, Yuvraj Agarwal, Laura Dabbish, Jason I. Hong, et al. Decentralized is not risk-free: Understanding public perceptions of privacy-utility trade-offs in covid-19 contact-tracing apps. *arXiv preprint arXiv:2005.11957*, 2020.

[11] **Tianshi Li**, Julia Katherine Haines, Miguel Flores Ruiz de Eguino, Jason I. Hong, and Jeffrey Nichols. Alert now or never: Understanding and predicting notification preferences of smartphone users. *ACM Transactions on Computer-Human Interaction*, 2021.

[12] **Tianshi Li**, Philip Quinn, and Shumin Zhai. C-PAK: Correcting and completing variable-length prefix-based abbreviated keystrokes. *ACM Transactions on Computer-Human Interaction*, 2022.