Helping Mobile Application Developers Create Accurate Privacy Labels

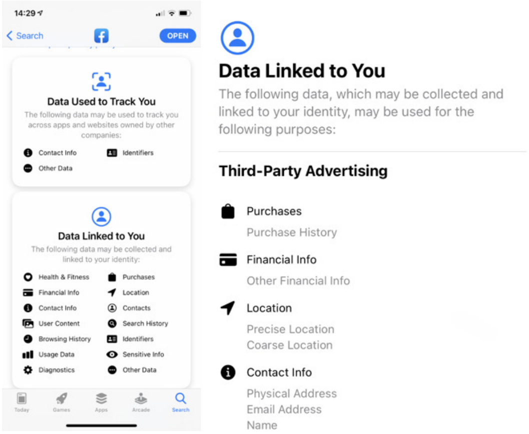
Jack Gardner*, Yuanyuan Feng†, Kayla Reiman*, Zhi Lin*, Akshath Jain* and Norman Sadeh*
*School of Computer Science
Carnegie Mellon University, Pittsburgh, USA, {jhgardne, zlin2, kreiman, arjain}@andrew.cmu.edu, {sadeh}@cs.cmu.edu
†Computer Science Department
University of Vermont, Burlington, USA, yuanyuan.feng@uvm.edu

***Abstract*—In December, 2020, Apple began requiring developers to disclose their data collection and use practices to generate a "privacy label" for their application. The use of mobile application Software Development Kits (SDKs) and third-party libraries, coupled with a typical lack of expertise in privacy, makes it challenging for developers to accurately report their data collection and use practices. In this work we discuss the design and evaluation of a tool to help iOS developers generate privacy labels. The tool combines static code analysis to identify likely data collection and use practices with interactive functionality designed to prompt developers to elucidate analysis results and carefully reflect on their applications' data practices. We conducted semi-structured interviews with iOS developers as they used an initial version of the tool. We discuss how these results motivated us to develop an enhanced software tool, *Privacy Label Wiz*, that more closely resembles interactions developers reported to be most useful in our semi-structured interviews. We present findings from our interviews and the enhanced tool motivated by our study. We also outline future directions for software tools to better assist developers communicating their mobile app's data practices to different audiences.**

***Index Terms*—Privacy labels, mobile applications, compliance, developers, Privacy Engineering**

1. Introduction

For the past decade, researchers have been investigating the potential for privacy labels as a standardized notice to assist consumers in understanding digital privacy [1]–[3], yet without large scale adoption, limited research examined developer perspectives of these labels. The December 2020 introduction of privacy labels on the iOS App Store was the first real-word rollout. Google has also announced plans to release a similar Android privacy label in 2022.[1] Apple now requires iOS developers to provide app privacy information when adding or updating their applications in the App Store.[2] Then, they synthesize the information provided by developers into a standardized label format to help iOS users understand the privacy details for each application (see Fig 1).



Data Used to Track You

The following data may be used to track you across apps and websites owned by other companies:

Contact Info
Identifiers
Other Data

Data Linked to You

The following data may be collected and linked to your identity:

Health & Fitness
Purchases
Financial Info
Location
Contact Info
Contacts
User Content
Search History
Browsing History
Identifiers
Usage Data
Sensitive Info
Diagnostics
Other Data

Data Linked to You

The following data, which may be collected and linked to your identity, may be used for the following purposes:

Third-Party Advertising

Purchases
Purchase History

Financial Info
Other Financial Info

Location
Precise Location
Coarse Location

Contact Info
Physical Address
Email Address
Name

Figure 1. Apple privacy label example

The Apple privacy label contains three sections of information (i.e., data used to track you, data linked to you, and data not linked to you) and provides both a high level summary as well as a detail page of an application's data collection and use practices. While earlier studies show privacy labels can enhance users' understanding of the privacy notice and available choices [1]–[3], how to generate accurate privacy labels in a real-world context remains under explored. The credibility and quality of any privacy labels obviously depend on their accuracy.

Currently, Apple requires developers to fill out a web form containing 32 data types and six purposes to provide details on which data is used by their app and how, but does not systematically verify the accuracy of the information provided by developers - automating this process is not feasible today (e.g., dead code, difficulty of assessing what happens in the cloud or verifying third-party practices), and systematic manual verification would be prohibitive. Therefore, the accuracy of these privacy labels largely depends on developers' expertise and dedication as well as their understanding of the data practices intended to be captured. A report from The Washington Post suggested that a number of iOS apps have inaccurate privacy labels [4]. A recent study by Li et al. revealed developers' misunderstanding in creating labels [5], which resulted in under-reporting or over-reporting some of the data practices of their apps. This is partially due to the extensive adoption of Software Development Kits (SDKs) and third-party libraries in app development, which makes it complicated and time-consuming for developers to fig-

[1] https://android-developers.googleblog.com/2021/07/new-google-play-safety-section.html
[2] https://developer.apple.com/news/?id=vlj9jty9

ure out the data collection and usage of their applications [6].

Failure to provide accurate privacy labels may have dire consequences. In the US, the Federal Trade Commission works to combat unfair and deceptive practices that infringe upon individual privacy rights and ensure that organizations act in-line with the privacy notices they provide to the public.[3] The California Consumer Privacy Act and California Privacy Rights Act also provide new privacy rights to Californians, place data protection obligations on business, and grant enforcement authority to the state's Office of the Attorney General.[4] The EU's General Data Protection Regulation also provides broad authority to European regulators to hold organizations accountable for ensuring that they are accurately disclosing their data practices,[5] and can entail significant financial penalties.

To ease the compliance burden developers may face and to address challenges in accurately reporting data collection and use practices, we developed ***Privacy Label Wiz (PLW)***, a tool to help iOS developers examine their source code and generate accurate privacy labels. This paper details an iterative set of modifications to *Privacy Flash Pro (PFP)* [7], [8], an open source software tool that served as the basis for our development of a new tool, *Privacy Label Wiz (PLW)*. We leverage PFP's static analysis framework that scans iOS applications' Swift source code and third-party libraries in Swift or Objective-C, detects certain data types, and allows developers to better understand the iOS permissions used by their application. *PLW* has the specific goal of helping developers create accurate privacy labels. It leverages results of static code analysis functionality to guide the process and prompt developers to reflect on their apps' data practices. The tool is designed to be highly interactive, recognizing the limitations of static analysis functionality applied to the code of mobile apps and leveraging interactions with developers to make up for these limitations. The highly interactive nature of the tool is further intended to align with typical software development workflows. We report on an initial set of user tests conducted with a first version of our tool and how results of these tests informed our later development process. We also discuss how these results highlight the potential for this type of tool to improve developers' understanding of the data practices of their apps and contribute to the development of more accurate privacy labels. Despite being relatively crude, the initial version of *PLW* proved very useful in helping us further elucidate developers' needs. In particular, by conducting our initial evaluation of the tool in the form of semi-structured interviews in which an interviewer knowledgeable in privacy helped supplement the shortcomings of the tool, we were able to identify interactions with developers that contributed to helping them refine their original privacy labels. Those developers further reported that as a result of these interactions they had a clearer understanding of their application's privacy practices, and had gained a better appreciation of the importance of carefully considering privacy in the development process. We discuss how these findings, and the feedback we received from developers, contributed to improvements to the design of *Privacy Label Wiz* and outline future directions for our tool.

2. Related work

2.1. Alternative ways to improve privacy notices

Prior research has documented a variety of issues with privacy notices including significant mismatch between the meaning of privacy policies and users' understanding as well as a lack of uniformity in policy content and format [9]. One approach to increasing privacy policy comprehension is the development of layered privacy policies [10] that use a standardized top-layer containing concise descriptions and corresponding links to sections of a full policy. Privacy icons offer another method to more quickly provide privacy information to consumers and can be integrated with web interfaces [11] to give users a quick glance at a site's privacy practices. Policy templates,[6] (e.g., those designed to support GDPR compliance) also promote standardized policy section headings and can recommend key topics for organizations to include in their privacy policies.

While these approaches are a step in the right direction towards providing informative standardized notices, policy formats with standardized sections still allow policy owners a high degree of flexibility in the content they choose to provide, which means that consumers are still faced with understanding a wide variety of policy content [12]. Additionally, the use of icons can be problematic since people are often interested in different sections of privacy policies, and it is difficult to provide relevant information in related icon descriptions [12]. Icons may also be misinterpreted if no accompanying text is provided [13], and icons along with other visual privacy notices may present accessibility issues [14]. As privacy labels are one means to address some of these challenges, we focus our work on these labels and review related prior research below.

2.2. "Privacy labels" as effective privacy notices

Providing privacy notices to consumers about applications' data collection practices before obtaining consent remains a current wide-spread approach to disclosing privacy practices, of which these labels are a part. Ideally, consumers read privacy policies and then agree to the policy based on understanding what companies plan to do with their data. However, the cognitive time and cost of reading privacy policies can make notice and choice impractical [15]. Privacy labels serve to increase the usability of notices so that consumers can make informed choices about their privacy and give more meaningful consent. While several studies have found that merely relying on notice and choice is inadequate to protect user privacy, improving notices is still valuable [12], [16]. However, the benefit of improved privacy notices can only be fully realized when these notices are accurate [12].

[3] https://www.ftc.gov/news-events/topics/protecting-consumer-privacy-security/privacy-security-enforcement

[4] https://oag.ca.gov/privacy/ccpa

[5] https://gdpr.eu/what-is-gdpr/

[6] https://gdpr.eu/privacy-notice/

Originally proposed in 2009 and tested in 2010, privacy labels have been discussed in academic circles for over a decade in the context of solving issues with privacy policies [1]. Important for our work, Carnegie Mellon University researchers, including one of the co-authors, first studied the potential for mobile app privacy labels in 2013 and found that by providing individuals with a "privacy facts" page in the app store descriptions of mobile apps, individuals selected applications with fewer permissions [3]. The clear presentation of privacy details benefited users over the preexisting application permissions page that users basically disregard. More recently, researchers have extended the privacy label approach to Internet of Things (IoT) devices [17], [18]. The present work was conducted after Apple's roll-out of privacy labels for iOS applications, aiming to improve the accuracy of these labels through our developer tool.

2.3. Barriers to create accurate privacy labels

Developers may make conscious and unconscious choices that compromise both the privacy and the accuracy of privacy notices. Developers may sacrifice users' privacy to implement features that will generate profits for applications, such as using an advertising network without evaluating the data collection practices of that network [19]. Additionally, as developers range from those who make applications in their spare time to those making applications for large companies, their level of expertise and the help they have access to also vary greatly [20]. Studies have shown that many apps suffer from potential compliance issues with developers struggling in particular with the disclosure of practices associated with the sharing of data with third parties (e.g., third-party libraries) [21], [22]. For instance, in a systematic analysis of over one million Android in the Google Play Store, Zimmeck et al. reported identifying an average of over 3 potential compliance issues per application [23].

A significant barrier for developers to correctly identify and convey their applications' data practices is the widespread use of software development toolkits (SDKs) for both functionality and advertising. An SDK is a set of software tools and programs provided by mobile platforms (e.g., iOS, Android) or third parties (e.g., Facebook, Twitter) that allow developers to build applications using existing services. For example, Facebook's SDK will enable applications to have social login features via Facebook accounts. While SDKs enhance applications with more features and greatly simplify the development process, they often also collect user data, track user behaviors, and send those data back to SDK vendors, which has often been the source of compliance issues. The phenomenon of privacy leakage via SDKs has been well documented for over a decade, especially on the Android platform [22]–[25]. A recent case study showed that the SDKs included in applications can collect users' private data (e.g., geographic locations, device identifiers) and send them back to vendors via User Datagram Protocol (UDP) connections even when the application is not used [26].

Moreover, SDKs may also provide sample code with privacy invasive defaults that are unknown to developers, which impact developers' coding decisions towards potentially privacy-violating options. While application developers may assume that SDKs abide by privacy laws, the SDK platforms place the burden on developers to be responsible for knowing what data is collected [21]. In summary, these barriers in the application development ecosystem prevent developers from creating accurate privacy labels. In practice, it seems unfair to expect all developers to have the necessary privacy expertise to identify and disclose these issues. Instead this is an area that really calls for the development of tools to assist developers.

2.4. Developer tools as a solution

The lack of resources and inability to accurately analyze and report data usage in mobile applications leads to inconsistencies between applications' stated privacy practices and the data they actually use [22]. Prior research shows that the static analysis of applications' code coupled with dynamic analysis of the related privacy policies can help developers and the managers of mobile application stores better understand when applications are under reporting their privacy practices [23]. As these tools become more common, new efforts have been made to increase the usability of static analysis tools and to ensure easy integration into developers' workflows [27].

Static analysis can also help alleviate some of the load that developers face in understanding how their code relates to privacy protection. Coconut, an Android studio plug-in developed in 2018 showed promising results in helping developers improve their privacy knowledge, allowing them to write better privacy policies [28]. Privacy Flash Pro, released in 2021, is a tool designed to help iOS developers by combining static code analysis, a policy template, and a wizard-based questionnaire [7], [8].

The studies above focus on improving accuracy when developers are filling out privacy policies, which are a long standing area where privacy expertise is needed. However, Apple's roll-out of privacy labels was the first time that developers were asked to answer privacy questions about their applications in a specific format with potentially unfamiliar definitions. Challenges to using Apple's web form, including developers' preconceived notions about words (e.g., "tracking") that do not match Apple's definitions, have already been documented [5]. Other challenges, such as struggling with the complexity of memorizing new definitions, or having knowledge blind spots, are also common. This work builds on previous efforts to help developers understand both how the code they write relates to privacy and how the SDKs they are using may need to be reported. We were motivated to build a tool capable of matching an application's code to Apple's privacy label. To our knowledge, our work is the first attempt to help developers fill out Apple's privacy label web form.

3. The development of *Privacy Label Wiz*

Leveraging the static code analysis of Privacy Flash Pro [7], [8] *Privacy Label Wiz (PLW)* detects whether data is being used in an application by analyzing the function calls in iOS applications as well as the use of third-party libraries. By restructuring this analysis framework, *PLW* provides a step-by-step guide to assist developers' privacy

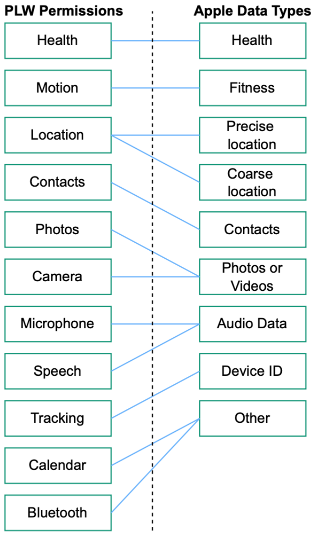


Figure 2. Map of permissions detected by *PLW* to data types defined by Apple.

label creation process by encouraging them to carefully review the data collection practices of their applications.

3.1. Analysis framework

PLW analyzes the code of iOS applications based on the use of iOS permissions in applications' Swift code. As some iOS permissions do not align well with the Apple data types that developers are asked to report in the App Store, *PLW* does not solely rely on this analysis. Instead it shows developers its findings while also informing them about this misalignment and prompts them to answer whether they may be collecting additional data types. Similarly, as Apple defines data collection as holding onto data "... for a period longer than necessary to service the transmitted request in real time", *PLW* cannot simply rely on static code analysis. Understanding how collected data may be linked to an individual or used to track an individual is another area where interacting with developers is crucial to supplement static analysis results. For instance, if a developer stores a data field like last login date on the same database line as the user ID, it could be considered linked. Accordingly, *PLW* uses static analysis results to trigger questions designed to prompt developers to think about scenarios just like this.

3.2. Data mapping and interface design

We map the 11 permissions that *PLW* can detect via the analysis of Swift code to the related data types defined by Apple. We alert users to this mapping as they work through the *PLW* user interface and conceptually show this relationship in Fig. 2. *PLW* also makes clear to users that there may not be a direct mapping from detected permissions to Apple's data types and allows developers to say they are "not sure" whether they collect and use a given data type as shown in Fig. 7.

In the refined version of *PLW*, we design a user interface that resembles Apple's web form for submitting privacy labels but simplify the interface structure so that developers interact with one page (see Fig. 3) for each data type their application may collect and also allow developers a degree of flexibility as they use the tool so they can review their entries after taking an initial pass through the tool. In particular, when developers say they are "not sure" whether they collect a given data type, we provide a summary page in the UI that lets developers see their entries thus far and allows them to revisit their answers for specific data types as needed to eventually report an answer when they complete Apple's web form. In a set of reflection questions at the bottom of the summary page, *PLW* also suggests that developers err on the side of caution to report a given data practice and informs them they can update their answer upon further consideration. Fig. 12 presents this section of the interface.

In the summary page, we also provide guiding questions for developers to review to help them think of additional resources they could consult to better understand the data types they collect and how they may be using those data. This set of questions is based on aspects of the data collection process that developers had trouble understanding [5].

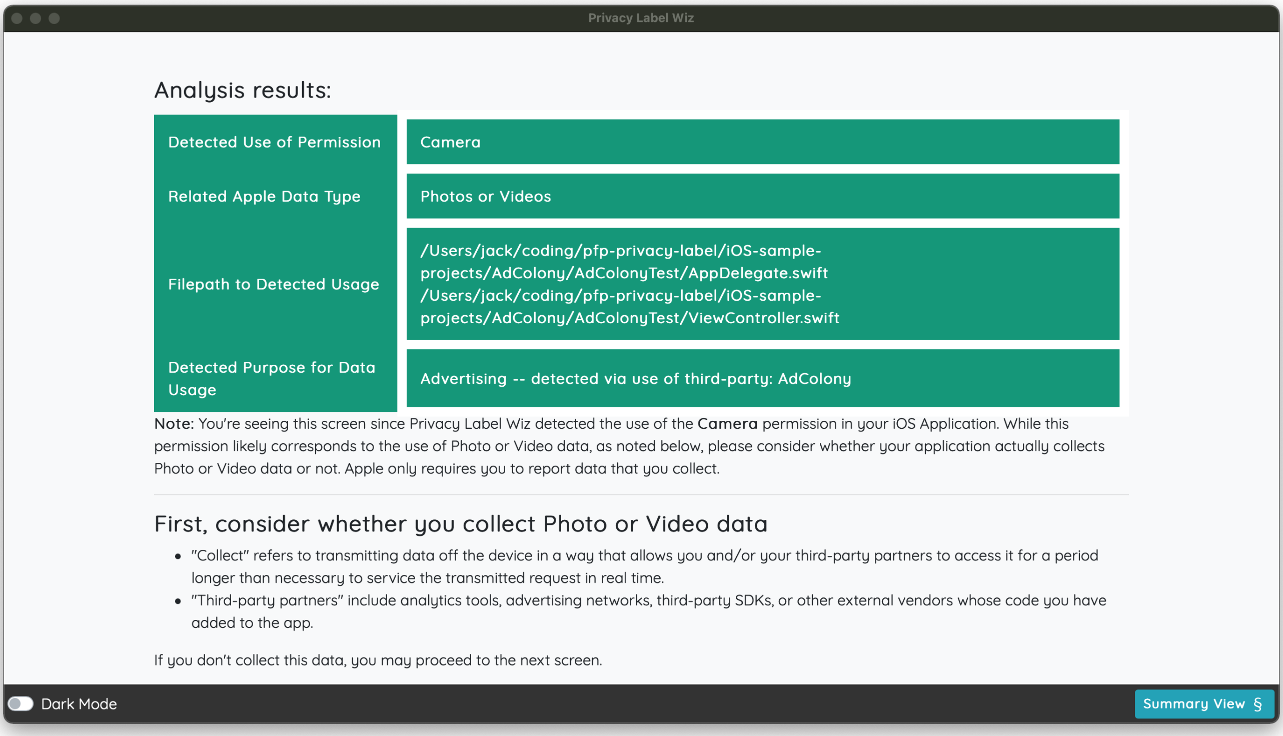
4. Initial usability study with developers

4.1. Study design and recruitment

We conducted an initial usability study to examine the potential value of *PLW* and evaluate how this tool could be improved to best help developers in the future. We aimed to recruit developers who had submitted an application to the Apple App Store so we could obtain feedback from developers that had completed Apple's privacy label process. The usability study required developers to run an initial version of *PLW* locally on their iOS application code and review a series of guiding questions while they worked through the label creation process. Contextual interviews [29] were used to learn about their experiences with *PLW*. We recruited four iOS developers, three of which had completed privacy labels. We posted recruitment messages to a variety of platforms that are shown starting at Fig. 19. We also clearly stated that *PLW* would only locally scan an application's source code and no data of source code would be collected to address developers' potential concerns. Our study protocol was approved by Carnegie Mellon University's Institutional Review Board.

4.2. Study results

We find that calling developers' attention to each step of the process, the data types involved, and how their data types are used prompted them to consider more deeply their application's data collection practices. However, the initial version of *PLW* did not stand alone. Developers expressed that *PLW* lacked user friendliness, and in the interviews, it was necessary to explain the purpose of each section of our UI, confirming that our tool required additional improvements before it could be valuable outside of the structured interview environment. Further, developers would have liked if we provided more up-front clarity on the limitations of the software tool to detect the data types they used. As one developer expressed disappointment that



Analysis results:

Detected Use of Permission	Camera
Related Apple Data Type	Photos or Videos
Filepath to Detected Usage	/Users/jack/coding/pfp-privacy-label/iOS-sample-projects/AdColony/AdColonyTest/AppDelegate.swift /Users/jack/coding/pfp-privacy-label/iOS-sample-projects/AdColony/AdColonyTest/ViewController.swift
Detected Purpose for Data Usage	Advertising -- detected via use of third-party: AdColony

Note: You're seeing this screen since Privacy Label Wiz detected the use of the **Camera** permission in your iOS Application. While this permission likely corresponds to the use of Photo or Video data, as noted below, please consider whether your application actually collects Photo or Video data or not. Apple only requires you to report data that you collect.

First, consider whether you collect Photo or Video data

- "Collect" refers to transmitting data off the device in a way that allows you and/or your third-party partners to access it for a period longer than necessary to service the transmitted request in real time.
- "Third-party partners" include analytics tools, advertising networks, third-party SDKs, or other external vendors whose code you have added to the app.

If you don't collect this data, you may proceed to the next screen.

Figure 3. The refined UI design displays example results for an open source project and explains the analysis results to the developer.

no data types were detected in their application's code, such an acknowledgement could have made it clear that this might happen from the start. An improved introduction to the guiding questions we present to developers to help them think more deeply about their data collection practices as well as an emphasis on the interactive nature of *PLW* could have also clarified that certain data types may not be detected. As one developer described Apple's definitions as "dense and hard to read", developers also requested additional information around how Apple's data types are defined (e.g., a short description or easily understood example relating to each data type).

In spite of the shortcomings in the initial version of *PLW*, using the software tool during the interview process led one developer to update their label after realizing their data should not be considered linked to users when they were prompted to consult documentation on the third-party service they used. By reading this documentation, this developer saw that the third-party service specifically provided information on how to complete Apple's privacy labels and understood that the functionality of the particular version of the third-party service they used did not constitute linking. The process of considering each of Apple's data types while using the initial version of *PLW* also led another developer to consider in more detail the difference between collection of product interaction data and user generated content. By understanding that product interaction involved clicking on certain features in the application while user generated content involved responses to specific questions that the user was asked to fill out, this developer updated their label to report that they had collected a data type for analysis purposes.

5. Refining *Privacy Label Wiz*

The findings reported above informed the redesign of our tool. By refining the UI, we provide additional guidance to developers, give them the ability to re-trace their steps during the privacy label creation process, and clarify the interactive nature of *PLW* by ensuring developers have a clear understanding of its underlying analysis framework. The remainder of this section discusses major refinements to our design.

Clarify the role of *PLW*: To better explain to developers how *PLW* works and to inform them that *PLW*'s goal is to use static analysis to guide them through an interactive series of screens that helps them systematically consider their mobile application's data practices, after developers load their iOS application into *PLW*, we add a new overview page (see Fig. 4) that at a high-level describes how *PLW* locally conducts static-analysis on the iOS application's source code and identifies iOS permissions that are used by the application. This page also previews the series of screens to follow and describes to developers that when they continue through the *PLW* interface, each following screen will focus on a single data type where developers will be asked to answer a series of questions that correspond to the information they would have to provide in Apple's web form: the purpose for collecting the data type, whether the data is linked to a user, and whether it is used to track the user. This description also includes that the iOS permissions detected by *PLW* may not map directly to the data types defined by Apple.

Pagination of *PLW* UI: *PLW*'s old design only had a results page where developers were presented with all results of *PLW*'s analysis in addition to a list of Apple's data types that do not map directly to an iOS permission that *PLW* can detect. In the current version, our goal is to avoid overwhelming developers with information and to allow them to think about how their application relates to each of Apple's data types one-by-one. Additionally, this new UI structure more closely resembles that of the Apple web form in which developers go through a series of pages where they first select the set of data types they collect and for each data type proceed to move through three screens where they enter the purposes for collecting that data type as well as linking and tracking information (see Figs. 5-8). However, instead of having three separate screens for each data type, we combine these screens into a single page. *PLW's* process also differs from Apple's in that it has developers start by providing information for the data types *PLW* detected and then allows developers to review a list of other data types they may be collecting to then provide information for those data types as necessary.

Room for Uncertainty: As many developers are not privacy experts and could be working through the privacy label completion process for the first time, it is unlikely that developers will be sure about the choices they make. To allow for this, we add the option for a developer to mark "not sure" alongside the standard yes-no options that are included in pages where developers are asked to provide information about the data types they collect. In a summary page that developers can access during or after they are done entering the relevant information for each data type, indicators appear (see Fig. 10) next to the data types for which a developer provided an uncertain answer. This additional option may also prove useful as there may not always be a direct mapping between the permissions detected by *PLW* and the data types defined by Apple. We address this by including additional text in the initial set of screens that developers may see regarding the data types detected by *PLW*. As mentioned earlier, the summary page also includes text that encourages developers to report data practices if they are not sure it is performed.

Tracking progress: As *PLW* is after all a wizard, we now provide progress indicators. At the start of the series of pages where developers provide information about their data types, we include a pagination to allow developers to easily navigate through the interactive portion of the tool.

Summary page: After developers move through a series of pages for each data type they collect, they arrive at a summary page (see Fig. 9) where they have the chance to review all of the answers they've provided and to select additional data types as needed. Developers can also access this page at any point during the interactive portion of the *PLW* UI via a summary button in the footer of each screen.

6. Discussion

Based on our process of redesigning *PLW* and what we have learned from developers, the remainder of this section outlines further improvements of our tool and future directions for software tools that support developers in privacy compliance.

6.1. Static analysis

By scanning developers' code, *PLW* is able to see which third-party libraries are called and which permissions are used. However, this does not actually show which data are being collected, how long the data are being stored, or the purpose of the data collection. Furthermore, even when correctly identifying third-party libraries, the tool does not have the capacity to keep track of those libraries' privacy policies and codebase changes over time. Therefore, it is not possible for this tool to answer all questions that are needed for the privacy label in a way that is guaranteed to be correct long-term, and it is also not possible for this tool to answer questions about how data is used. Given that Apple's definition of data collection relies on the data being stored for "longer than is necessary" for the function of an application, *PLW* does not know which data types are collected. We worked to mitigate this limitation by providing the developers with guiding questions and examples in the *PLW* UI, although this guidance can likely be refined with additional testing.

6.2. Errors in library detection

Since *PLW* is an adaptation of the published open source tool Privacy Flash Pro, we relied on the existing analysis framework. In two of the initial interviews with developers, *PLW* was unable to identify Firebase, a common third-party library. This is likely because the developers' iOS applications were structured differently than the open source applications we used to test the initial version of *PLW*. The current iteration of the tool could be improved by better detecting libraries outside of the Pods (third-party library) section of Apple's integrated development environment (IDE), Xcode, and it could also be improved by querying Cocoa Pods' (a common place to get Apple SDKs) website in order to identify the practices and existence of more than just the 300 applications that the Privacy Flash Pro team originally added.

6.3. Providing developers with the resources they need

Given the array of third-party services that developers use, additional resources that allow developers to more easily understand the data collection and use practices of these services would help developers create more accurate data collection disclosures. For example, a centralized repository that provides privacy-related resources for frequently used third-party services would help developers more quickly locate information on data collection practices that are specific to a service. Some vendors such as OneTrust have started to build these repositories, though they are not integrated with tools like *PLW*. Ideally, a tool like *PLW* would directly connect to relevant privacy summaries for third-party libraries, enabling developers to readily assess the privacy implications of using these libraries or services.

Google's Checks [7] is a recently introduced platform, announced in February 2022, that aims to analyze mobile applications and their data sharing practices by looking at

[7] https://checks.area120.google.com

network flows, SDKs used, and an application's privacy policy to help developers achieve privacy compliance. Future tools would ideally help developers identify privacy compliance issues during their development process, help them better understand how their application's data collection and use practices specifically relate to privacy compliance requirements, and if necessary, help them identify alternative options.

Developers would also benefit from compliance tools that let them record how they reached a certain conclusion. As we further develop *PLW*, we look to add the ability for users to add comments or notes that would let them document certain resources they may have consulted (e.g., the privacy policy of a third-party service) to better support compliance and to help users more easily update their labels as part of new releases.

6.4. Importance of interviewing developers

Software developers come from many different backgrounds, and may be familiar with different vocabulary and have different assumptions about what terms mean. For example, Li et al. found that some developers expressed confusion about topics that they felt they should know, and tracking was one of the definitions that developers often had preconceived notions about that actually prevented them from being able to understand Apple's definition [5]. A tool that scans a developer's code and provides instructions for how to think about questions that cannot be answered by static analysis alone requires instructions that are meaningful to the developer. While we sought to apply strategies such as having the simplest instructions possible and thus minimizing cognitive overload, there will never be a single set of written instructions that can work for everyone. This is a challenge with both our interface and Apple's. While we design *PLW* to resemble interaction with a privacy expert, it is hard for the software tool to measure up to receiving one-on-one guidance on the completion of privacy labels.

Given that *PLW*'s initial instructions were not not fully able to answer every developer question, our interviews erred on the side of giving developers help when they requested it. Some developers felt comfortable reading over the definitions that Apple provides along with the guiding questions provided in the initial *PLW* UI. However, others requested clarification and were interested in discussing whether their use of a data type met the criteria specified in Apple's web form. Thus, these conversations provided additional guidance to developers beyond the scope of the software tool, potentially leading them to make changes to their privacy labels they would not have made with the use of the software tool alone. Interviewing developers with the updated version of *PLW* will be a crucial step in ensuring that its improved instructions provide sufficient guidance to developers and limit the need for individual guidance.

6.5. Future evaluation of *PLW's* effectiveness

While our initial set of interviews was formative and guided us to refine *PLW*, a future within-subject study examining how developers fill out labels both with and without the improved *PLW* will better evaluate its effectiveness. This future study will ideally include a larger sample of developers with published applications and will give developers access to *PLW* after they had filled out Apple's web form on their own to see if they update their label. We plan to use the think aloud technique to identify where challenges are present. We also plan to develop a validated post-study survey to ask developers about their experiences using *PLW* to rigorously assess its utility and to examine any known inaccuracies. One challenge is that we cannot know the ground truth regarding what data applications collect and use for different purposes, as per Apple's definition. However, a privacy-expert's evaluation of the completed labels could still identify some inaccuracies and check for trends on whether developers were more likely to accurately complete certain parts of the privacy labels after using *PLW*.

6.6. Simplifying privacy labels across platforms

While Apple is the only platform that currently requires developers to fill out these labels, similar requirements are being rolled out in the Google Play Store, with slightly different definitions [30]. In addition to studying whether static analysis tools like *PLW* can be helpful, future research could explore methods to help developers navigate the differences in process and definitions across platforms. We already see in our work that Apple's niche definitions can cause problems for developers, and given that many developers create apps that are deployed on both Apple and Android platforms, they will likely have to address each platform's requirements separately.

7. Conclusion

The interviews we have conducted and the improvements we have made to *PLW* illustrate the potential of software tools that statically analyze code for iOS apps to help developers fill out labels more accurately and to learn more about privacy. This analysis could further be supplemented with dynamic code analysis [26] to better understand runtime behavior and inform developers how this behavior could relate to their privacy labels. Although more rigorous testing is still needed, our work shows the potential for a tool that mimics a conversation with a privacy engineer to help developers improve their labels and increase their privacy compliance knowledge. In particular, resembling the steps in Apple's privacy label interface, addressing the limitations of the static analysis process underlying *PLW*, and providing a chance for developers to revisit their choices while reviewing examples of data collection practices and comments to simplify the data type and collection definitions issued by Apple has the potential to substantially benefit developers and streamline the privacy label creation process.

Large operators in mobile application ecosystems could also simplify the privacy label creation process by providing descriptions of how the permissions used in mobile applications may relate to the data types they define. Further, describing common actions that mobile applications perform and how they meet the definitions provided in the privacy label process may help developers better tie the function of their application to the practices they should report.

Acknowledgments

We are grateful to Aerin Zhang and our reviewers for their feedback and help in refining this work. This research was supported in part by grants from the National Science Foundation Secure and Trustworthy Computing program (CNS-1801316, CNS-1914486). Part of the work on mobile app privacy compliance and static code analysis was conducted under the Usable Privacy Policy Project(https://usableprivacy.org - NSF Grant CNS-1330596). Additional support for this work was provided in the form of an unrestricted gift from Google's privacy-related faculty award.

References

[1] P. G. Kelley, J. Bresee, L. F. Cranor, and R. W. Reeder, "A "nutrition label" for privacy," in *Proceedings of the 5th Symposium on Usable Privacy and Security*, SOUPS '09, (New York, NY, USA), Association for Computing Machinery, 2009.

[2] P. G. Kelley, L. Cesca, J. Bresee, and L. F. Cranor, "Standardizing privacy notices: An online study of the nutrition label approach," in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '10, (New York, NY, USA), p. 1573–1582, Association for Computing Machinery, 2010.

[3] P. G. Kelley, L. F. Cranor, and N. Sadeh, "Privacy as part of the app decision-making process," in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '13, (New York, NY, USA), p. 3393–3402, Association for Computing Machinery, 2013.

[4] G. Fowler, "I checked apple's new privacy 'nutrition labels'. Many were false," *The Washington Post*, 2021.

[5] T. Li, K. Reiman, Y. Agarwal, L. F. Cranor, and J. I. Hong, "Understanding challenges for developers to create accurate privacy nutrition labels," in *CHI Conference on Human Factors in Computing Systems (CHI '22)*, (New York, NY, USA), Association for Computing Machinery, 2022.

[6] S. Morrison, "The hidden trackers in your phone, explained," *Vox*, 2020.

[7] S. Zimmeck, R. Goldstein, and D. Baraka, "PrivacyFlash Pro: Automating privacy policy generation for mobile apps," in *28th Network and Distributed System Security Symposium (NDSS 2021). NDSS*, 2021.

[8] S. Zimmeck, P. Story, R. Goldstein, D. Baraka, S. Li, Y. Feng, and N. Sadeh, "Compliance traceability: Privacy policies as software development artifacts," in *Open Day for Privacy, Usability, and Transparency (PUT), Stockholm, Sweden*, 2019.

[9] J. R. Reidenberg, T. Breaux, L. F. Carnor, B. French, A. Grannis, J. T. Graves, F. Liu, A. McDonald, T. B. Norton, R. Ramanath, N. C. Russell, N. Sadeh, and F. Schaub, "Disagreeable privacy policies: Mismatches between meaning and users' understanding," *Berkeley Tech. L.J.. Berkeley Technology Law Journal*, vol. 30, no. IR, p. 39.

[10] "Multi-layered notices explained," *The Center for Information Policy Leadership*, 2004.

[11] S. Egelman, J. Tsai, L. F. Cranor, and A. Acquisti, "Timing is everything? The effects of timing and placement of online privacy indicators," in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '09, (New York, NY, USA), p. 319–328, Association for Computing Machinery, 2009.

[12] L. F. Cranor, "Necessary but not sufficient: Standardized mechanisms for privacy notice and choice," *J. on Telecomm. & High Tech. L.*, vol. 10, p. 273, 2012.

[13] H. Habib, Y. Zou, Y. Yao, A. Acquisti, L. Cranor, J. Reidenberg, N. Sadeh, and F. Schaub, "Toggles, dollar signs, and triangles: How to (in)effectively convey privacy choices with icons and link texts," in *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems*, CHI '21, (New York, NY, USA), Association for Computing Machinery, 2021.

[14] F. Schaub, R. Balebako, A. L. Durity, and L. F. Cranor, "A design space for effective privacy notices," in *Eleventh Symposium On Usable Privacy and Security (SOUPS 2015)*, (Ottawa), pp. 1–17, USENIX Association, July 2015.

[15] A. M. McDonald and L. F. Cranor, "The cost of reading privacy policies," *ISJLP*, vol. 4, p. 543, 2008.

[16] F. H. Cate, "The limits of notice and choice," *IEEE Security & Privacy*, vol. 8, no. 2, pp. 59–62, 2010.

[17] Y. Shen and P.-A. Vervier, "IoT security and privacy labels," in *Annual Privacy Forum*, pp. 136–147, Springer, 2019.

[18] P. Emami-Naeini, Y. Agarwal, L. F. Cranor, and H. Hibshi, "Ask the experts: What should be on an IoT privacy and security label?," in *2020 IEEE Symposium on Security and Privacy (SP)*, pp. 447–464, IEEE, 2020.

[19] A. H. Mhaidli, Y. Zou, and F. Schaub, "We can't live without them! App developers' adoption of ad networks and their considerations of consumer risks," in *Fifteenth Symposium on Usable Privacy and Security (SOUPS 2019)*, (Santa Clara, CA, USA), pp. 225–244, USENIX Association, Aug. 2019.

[20] R. Balebako, A. Marsh, J. Lin, J. I. Hong, and L. F. Cranor, "The privacy and security behaviors of smartphone app developers," in *Workshop on Usable Security*, 2014.

[21] M. Tahaei and K. Vaniea, "Developers are responsible: What ad networks tell developers about privacy," in *Extended Abstracts of the 2021 CHI Conference on Human Factors in Computing Systems*, pp. 1–11, 2021.

[22] S. Zimmeck, Z. Wang, L. Zou, R. Iyengar, B. Liu, F. Schaub, S. Wilson, N. Sadeh, S. Bellovin, and J. Reidenberg, "Automated analysis of privacy requirements for mobile apps," in *2016 AAAI Fall Symposium Series*, 2016.

[23] S. Zimmeck, P. Story, D. Smullen, A. Ravichander, Z. Wang, J. R. Reidenberg, N. C. Russell, and N. Sadeh, "Maps: Scaling privacy compliance analysis to a million apps," *Proc. Priv. Enhancing Tech.*, vol. 2019, pp. 66–86, 2019.

[24] J. Lin, S. Amini, J. I. Hong, N. Sadeh, J. Lindqvist, and J. Zhang, "Expectation and purpose: Understanding users' mental models of mobile app privacy through crowdsourcing," in *Proceedings of the 2012 ACM Conference on Ubiquitous Computing*, UbiComp '12, (New York, NY, USA), p. 501–510, Association for Computing Machinery, 2012.

[25] J. Kim, Y. Yoon, K. Yi, J. Shin, and S. Center, "ScanDal: Static analyzer for detecting privacy leaks in android applications," in *MoST 2012: Mobile Security Technologies*, vol. 12, (Los Alamitos, CA, USA), IEEE, 2012.

[26] J. Reardon, N. Good, R. Richter, N. Vallina-Rodriguez, S. Egelman, and Q. Palfrey, "Jpush away your privacy: A case study of Jiguang's Android SDK," *International Computer Science Institute*, 2020.

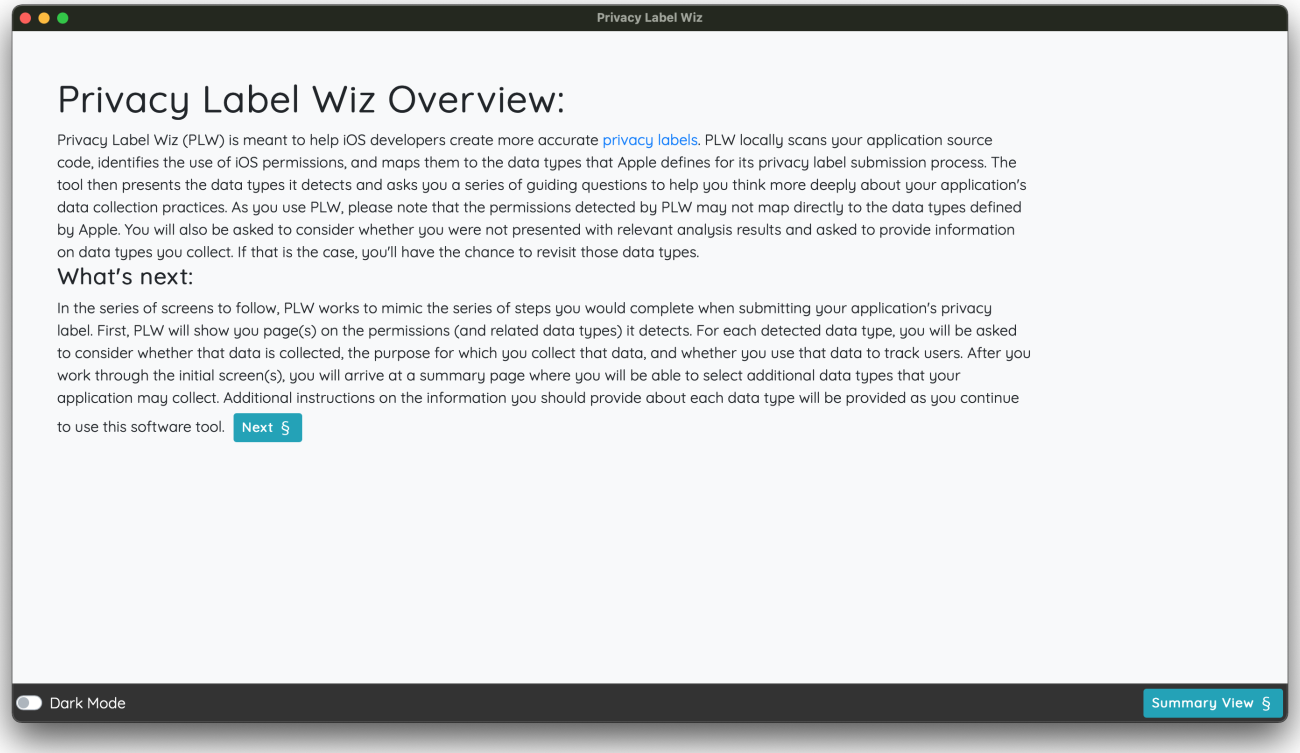
[27] D. Tiganov, L. Nguyen Quang Do, and K. Ali, "Designing UIs for static analysis tools," *ACM Queue*, 2021.

[28] T. Li, Y. Agarwal, and J. I. Hong, "Coconut: An IDE plugin for developing privacy-friendly apps," *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, vol. 2, no. 4, pp. 1–35, 2018.

[29] K. Holtzblatt and S. Jones, "Conducting and analyzing a contextual interview (excerpt)," in *Readings in Human–Computer Interaction*, pp. 241–253, Elsevier, 1995.

[30] "Following Apple's launch of privacy labels, Google to add a 'Safety' section in Google Play," *TechCrunch*. Accessed 16 Nov. 2021.

Appendix A.
The Privacy Label Wiz User Interface



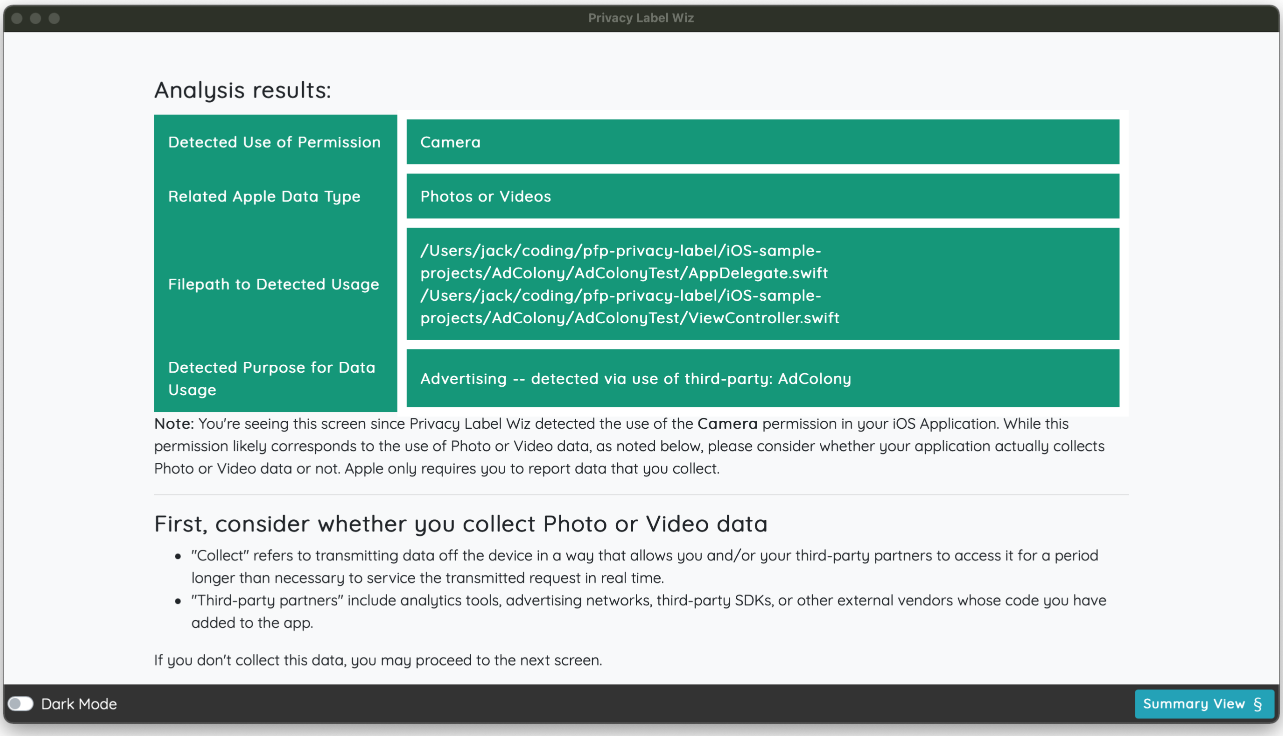
Privacy Label Wiz Overview:

Privacy Label Wiz (PLW) is meant to help iOS developers create more accurate privacy labels. PLW locally scans your application source code, identifies the use of iOS permissions, and maps them to the data types that Apple defines for its privacy label submission process. The tool then presents the data types it detects and asks you a series of guiding questions to help you think more deeply about your application's data collection practices. As you use PLW, please note that the permissions detected by PLW may not map directly to the data types defined by Apple. You will also be asked to consider whether you were not presented with relevant analysis results and asked to provide information on data types you collect. If that is the case, you'll have the chance to revisit those data types.

What's next:

In the series of screens to follow, PLW works to mimic the series of steps you would complete when submitting your application's privacy label. First, PLW will show you page(s) on the permissions (and related data types) it detects. For each detected data type, you will be asked to consider whether that data is collected, the purpose for which you collect that data, and whether you use that data to track users. After you work through the initial screen(s), you will arrive at a summary page where you will be able to select additional data types that your application may collect. Additional instructions on the information you should provide about each data type will be provided as you continue to use this software tool.

Figure 4. *PLW* Overview.



Analysis results:

Detected Use of Permission	Camera
Related Apple Data Type	Photos or Videos
Filepath to Detected Usage	/Users/jack/coding/pfp-privacy-label/iOS-sample-projects/AdColony/AdColonyTest/AppDelegate.swift /Users/jack/coding/pfp-privacy-label/iOS-sample-projects/AdColony/AdColonyTest/ViewController.swift
Detected Purpose for Data Usage	Advertising -- detected via use of third-party: AdColony

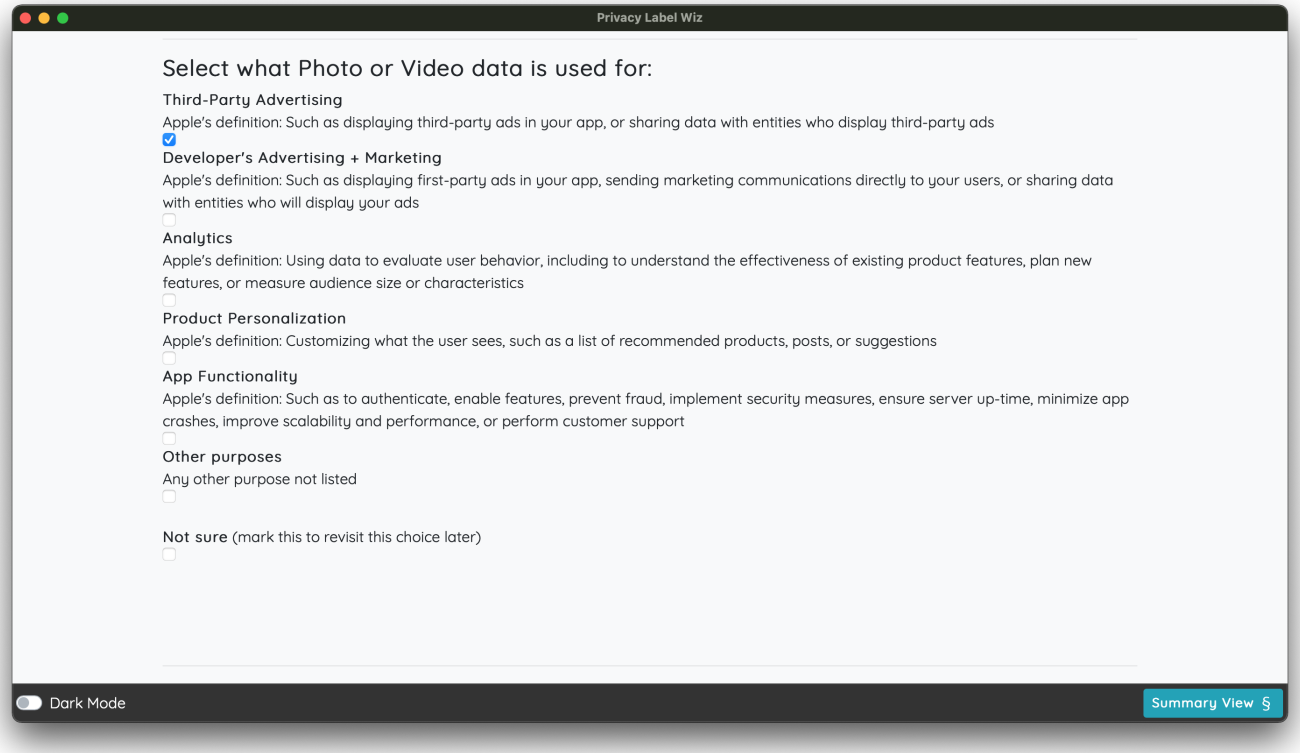
Note: You're seeing this screen since Privacy Label Wiz detected the use of the **Camera** permission in your iOS Application. While this permission likely corresponds to the use of Photo or Video data, as noted below, please consider whether your application actually collects Photo or Video data or not. Apple only requires you to report data that you collect.

First, consider whether you collect Photo or Video data

- "Collect" refers to transmitting data off the device in a way that allows you and/or your third-party partners to access it for a period longer than necessary to service the transmitted request in real time.
- "Third-party partners" include analytics tools, advertising networks, third-party SDKs, or other external vendors whose code you have added to the app.

If you don't collect this data, you may proceed to the next screen.

Figure 5. *PLW* main results.



Select what Photo or Video data is used for:

Third-Party Advertising

Apple's definition: Such as displaying third-party ads in your app, or sharing data with entities who display third-party ads

☑

Developer's Advertising + Marketing

Apple's definition: Such as displaying first-party ads in your app, sending marketing communications directly to your users, or sharing data with entities who will display your ads

□

Analytics

Apple's definition: Using data to evaluate user behavior, including to understand the effectiveness of existing product features, plan new features, or measure audience size or characteristics

□

Product Personalization

Apple's definition: Customizing what the user sees, such as a list of recommended products, posts, or suggestions

□

App Functionality

Apple's definition: Such as to authenticate, enable features, prevent fraud, implement security measures, ensure server up-time, minimize app crashes, improve scalability and performance, or perform customer support

□

Other purposes

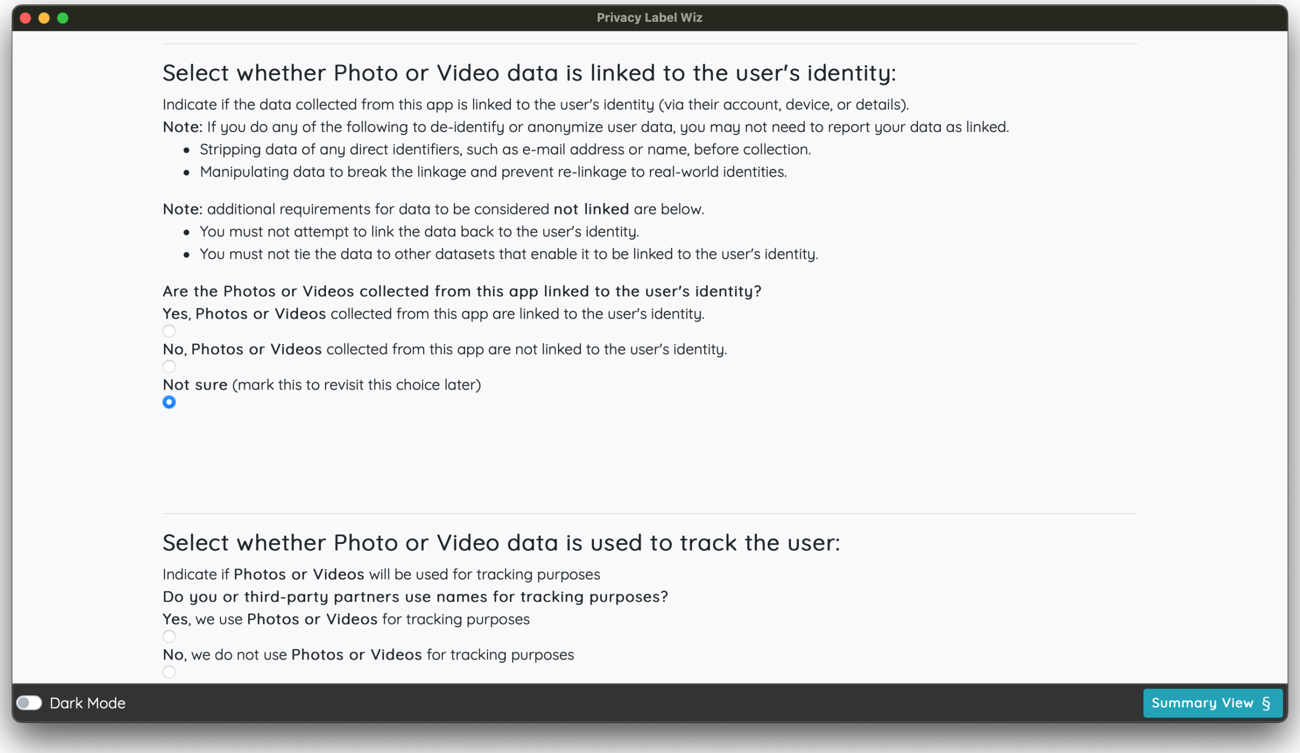
Any other purpose not listed

□

Not sure (mark this to revisit this choice later)

□

Figure 6. Selecting usages on the results screen.



Select whether Photo or Video data is linked to the user's identity:

Indicate if the data collected from this app is linked to the user's identity (via their account, device, or details).

Note: If you do any of the following to de-identify or anonymize user data, you may not need to report your data as linked.

- Stripping data of any direct identifiers, such as e-mail address or name, before collection.
- Manipulating data to break the linkage and prevent re-linkage to real-world identities.

Note: additional requirements for data to be considered **not linked** are below.

- You must not attempt to link the data back to the user's identity.
- You must not tie the data to other datasets that enable it to be linked to the user's identity.

Are the Photos or Videos collected from this app linked to the user's identity?

Yes, Photos or Videos collected from this app are linked to the user's identity.

○

No, Photos or Videos collected from this app are not linked to the user's identity.

○

Not sure (mark this to revisit this choice later)

●

Select whether Photo or Video data is used to track the user:

Indicate if **Photos or Videos** will be used for tracking purposes

Do you or third-party partners use names for tracking purposes?

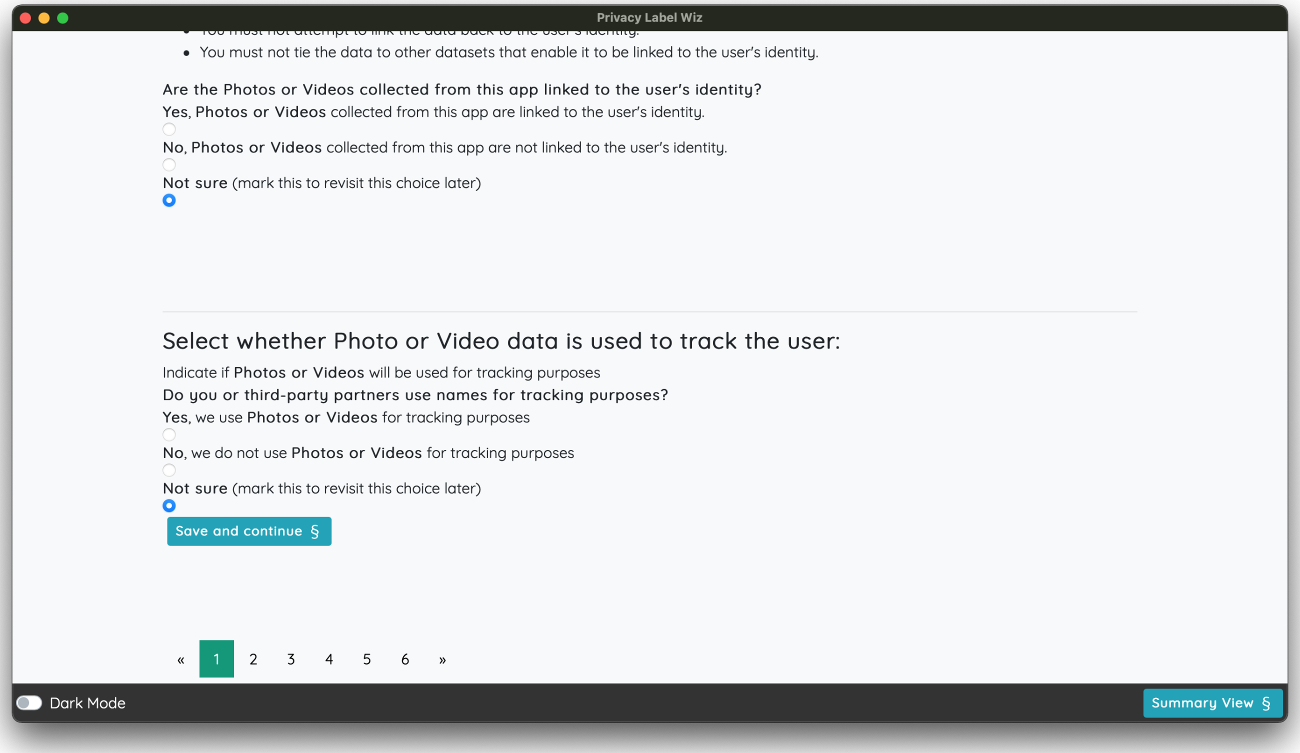
Yes, we use **Photos or Videos** for tracking purposes

○

No, we do not use **Photos or Videos** for tracking purposes

○

Figure 7. Selecting whether data is linked to users on the results screen.



- You must not attempt to link the data back to the user's identity.
- You must not tie the data to other datasets that enable it to be linked to the user's identity.

Are the Photos or Videos collected from this app linked to the user's identity?

Yes, Photos or Videos collected from this app are linked to the user's identity.
○

No, Photos or Videos collected from this app are not linked to the user's identity.
○

Not sure (mark this to revisit this choice later)
●

Select whether Photo or Video data is used to track the user:

Indicate if Photos or Videos will be used for tracking purposes

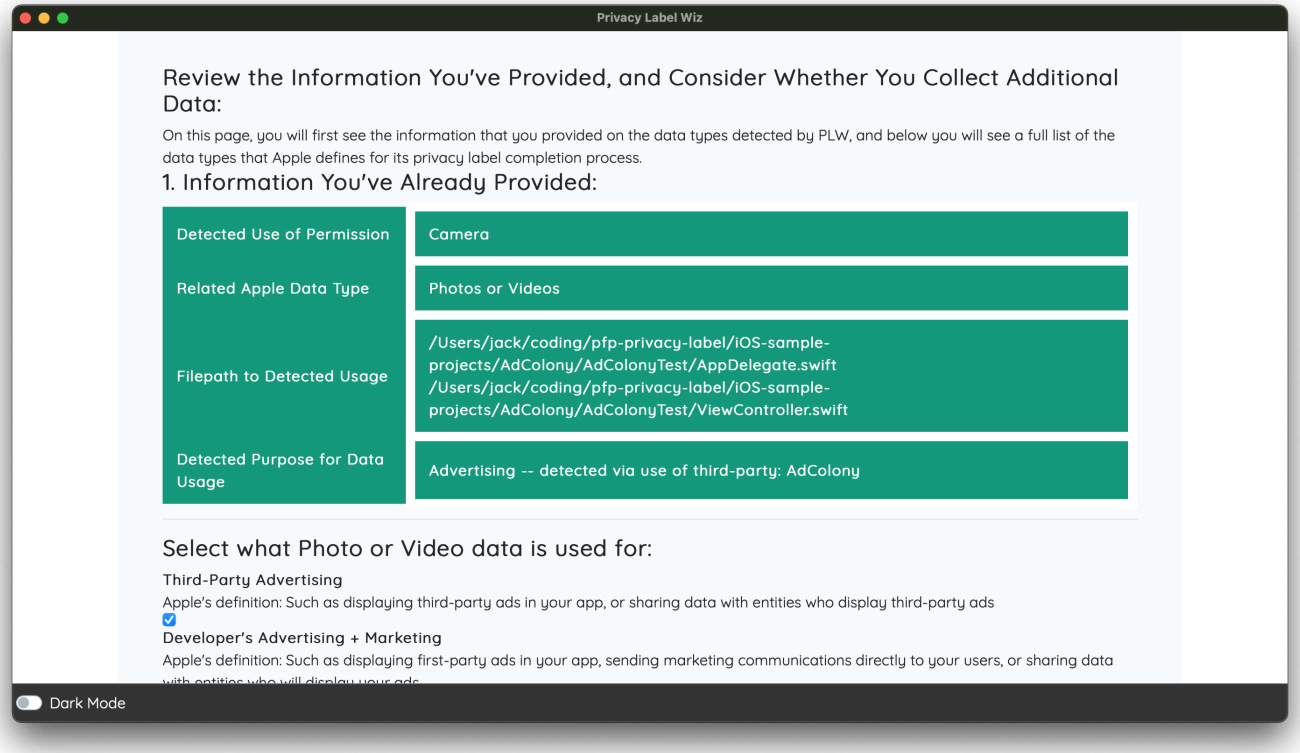
Do you or third-party partners use names for tracking purposes?

Yes, we use Photos or Videos for tracking purposes
○

No, we do not use Photos or Videos for tracking purposes
○

Not sure (mark this to revisit this choice later)
●

Figure 8. Selecting whether data is used to track users on the results screen.



Review the Information You've Provided, and Consider Whether You Collect Additional Data:

On this page, you will first see the information that you provided on the data types detected by PLW, and below you will see a full list of the data types that Apple defines for its privacy label completion process.

1. Information You've Already Provided:

Detected Use of Permission	Camera
Related Apple Data Type	Photos or Videos
Filepath to Detected Usage	/Users/jack/coding/pfp-privacy-label/iOS-sample-projects/AdColony/AdColonyTest/AppDelegate.swift /Users/jack/coding/pfp-privacy-label/iOS-sample-projects/AdColony/AdColonyTest/ViewController.swift
Detected Purpose for Data Usage	Advertising -- detected via use of third-party: AdColony

Select what Photo or Video data is used for:

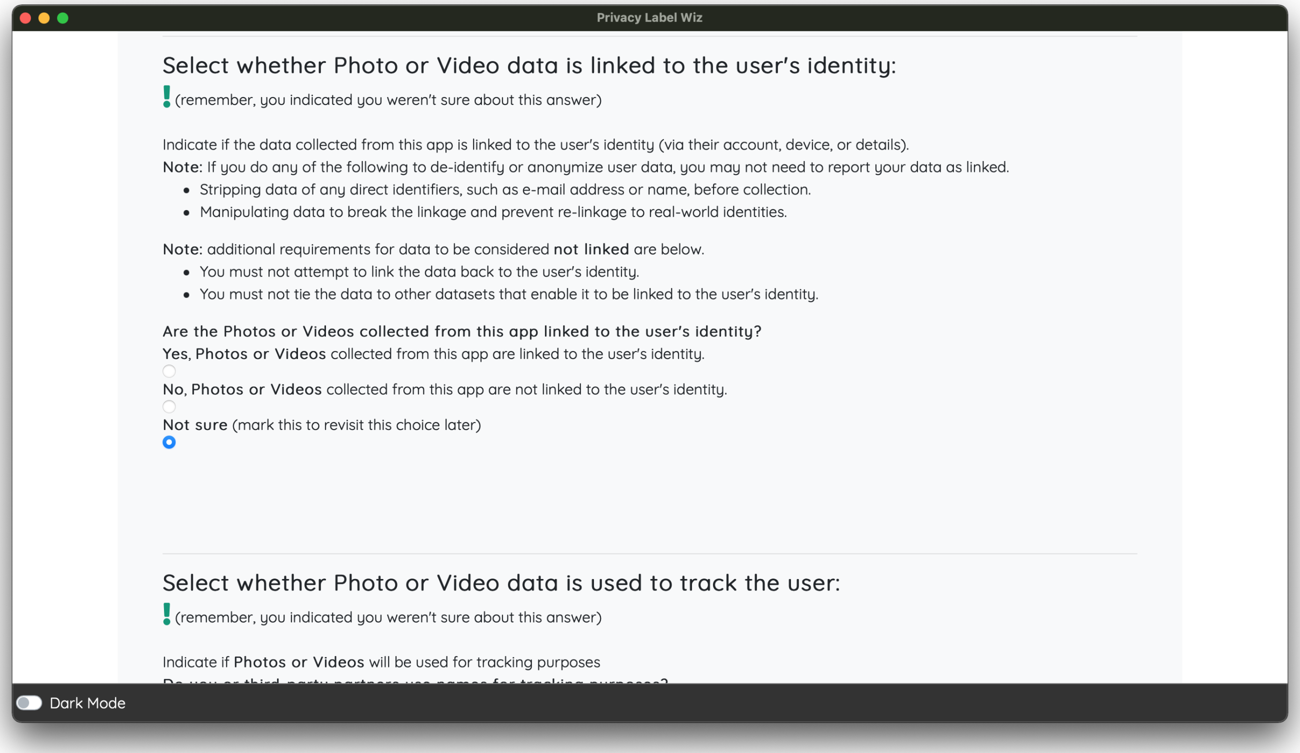
Third-Party Advertising

Apple's definition: Such as displaying third-party ads in your app, or sharing data with entities who display third-party ads
☑

Developer's Advertising + Marketing

Apple's definition: Such as displaying first-party ads in your app, sending marketing communications directly to your users, or sharing data with entities who will display your ads

Figure 9. Top of *PLW* summary page



Select whether Photo or Video data is linked to the user's identity:

! (remember, you indicated you weren't sure about this answer)

Indicate if the data collected from this app is linked to the user's identity (via their account, device, or details).

Note: If you do any of the following to de-identify or anonymize user data, you may not need to report your data as linked.

- Stripping data of any direct identifiers, such as e-mail address or name, before collection.
- Manipulating data to break the linkage and prevent re-linkage to real-world identities.

Note: additional requirements for data to be considered **not linked** are below.

- You must not attempt to link the data back to the user's identity.
- You must not tie the data to other datasets that enable it to be linked to the user's identity.

Are the Photos or Videos collected from this app linked to the user's identity?

Yes, Photos or Videos collected from this app are linked to the user's identity.
○

No, Photos or Videos collected from this app are not linked to the user's identity.
○

Not sure (mark this to revisit this choice later)
●

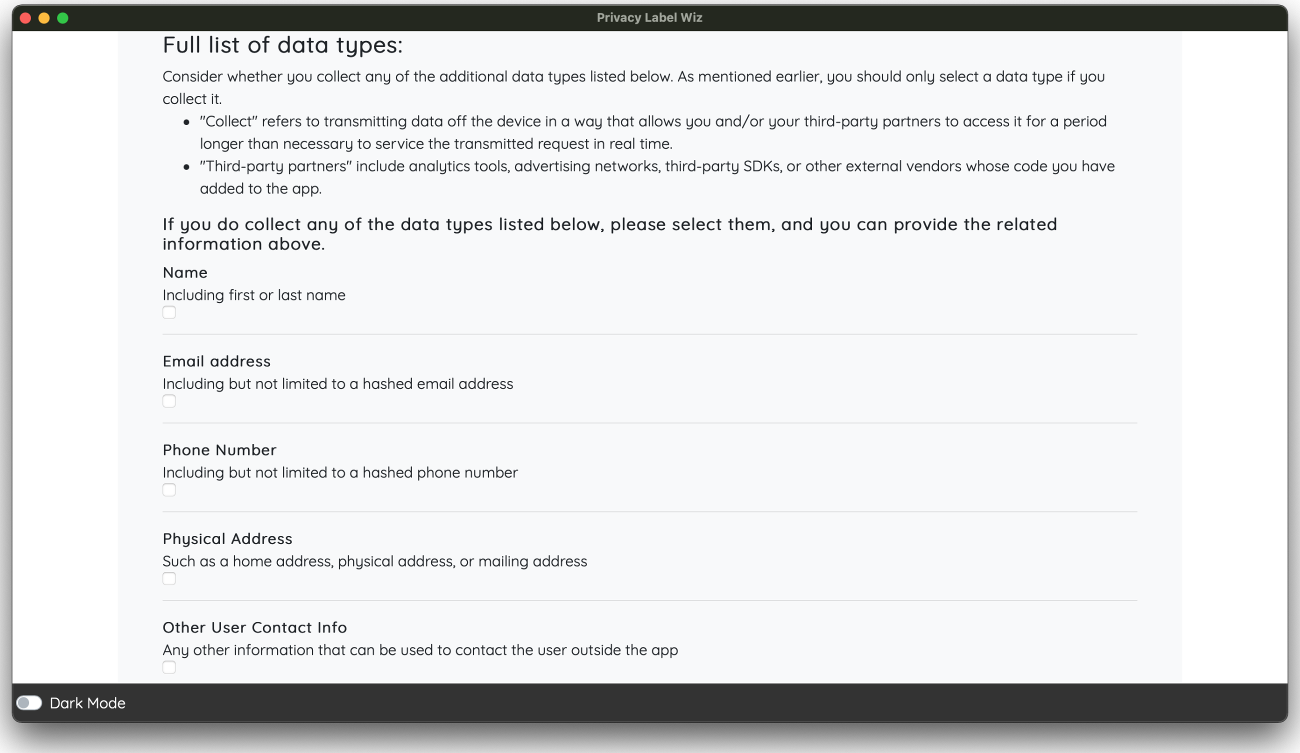
Select whether Photo or Video data is used to track the user:

! (remember, you indicated you weren't sure about this answer)

Indicate if **Photos or Videos** will be used for tracking purposes

Do you or third-party partners use names for tracking purposes?

Figure 10. Warning about "not sure" answer on *PLW* summary page



Full list of data types:

Consider whether you collect any of the additional data types listed below. As mentioned earlier, you should only select a data type if you collect it.

- "Collect" refers to transmitting data off the device in a way that allows you and/or your third-party partners to access it for a period longer than necessary to service the transmitted request in real time.
- "Third-party partners" include analytics tools, advertising networks, third-party SDKs, or other external vendors whose code you have added to the app.

If you do collect any of the data types listed below, please select them, and you can provide the related information above.

Name
Including first or last name
□

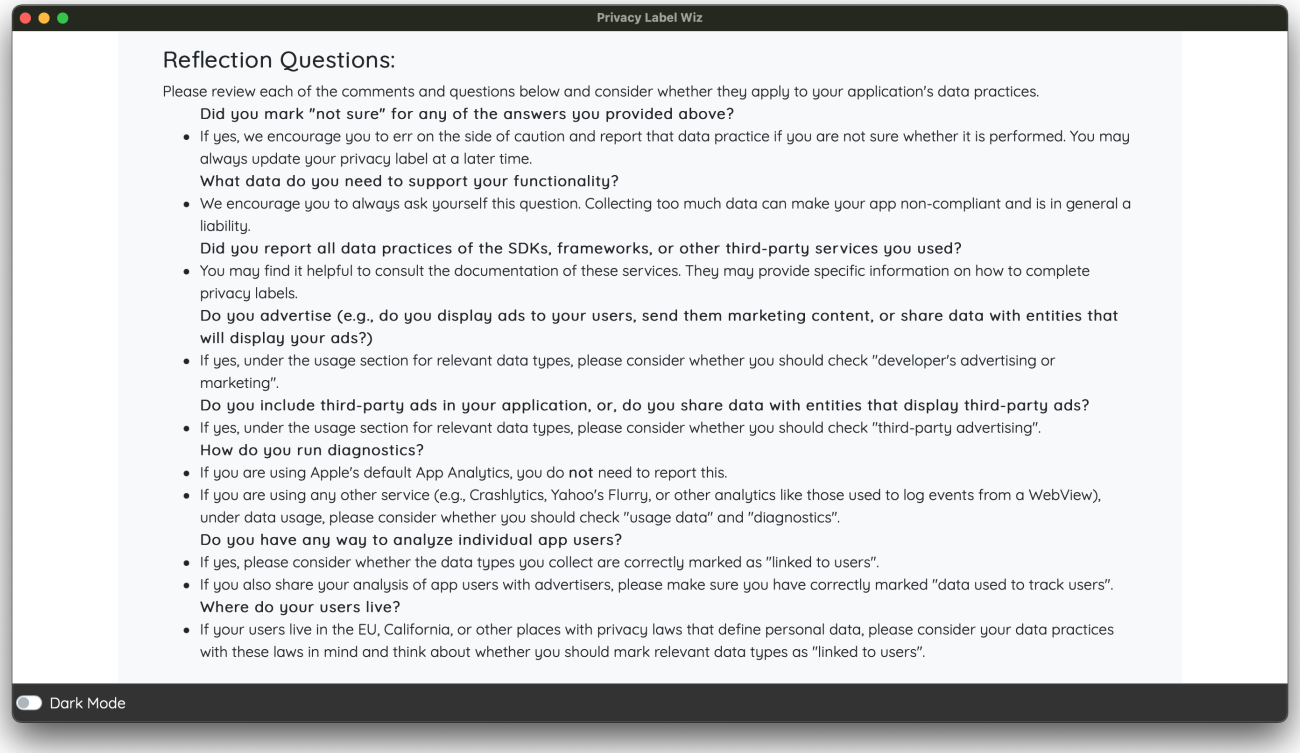
Email address
Including but not limited to a hashed email address
□

Phone Number
Including but not limited to a hashed phone number
□

Physical Address
Such as a home address, physical address, or mailing address
□

Other User Contact Info
Any other information that can be used to contact the user outside the app
□

Figure 11. Selecting additional data types on *PLW* summary page



Reflection Questions:

Please review each of the comments and questions below and consider whether they apply to your application's data practices.

Did you mark "not sure" for any of the answers you provided above?

- If yes, we encourage you to err on the side of caution and report that data practice if you are not sure whether it is performed. You may always update your privacy label at a later time.

What data do you need to support your functionality?

- We encourage you to always ask yourself this question. Collecting too much data can make your app non-compliant and is in general a liability.

Did you report all data practices of the SDKs, frameworks, or other third-party services you used?

- You may find it helpful to consult the documentation of these services. They may provide specific information on how to complete privacy labels.

Do you advertise (e.g., do you display ads to your users, send them marketing content, or share data with entities that will display your ads?)

- If yes, under the usage section for relevant data types, please consider whether you should check "developer's advertising or marketing".

Do you include third-party ads in your application, or, do you share data with entities that display third-party ads?

- If yes, under the usage section for relevant data types, please consider whether you should check "third-party advertising".

How do you run diagnostics?

- If you are using Apple's default App Analytics, you do **not** need to report this.
- If you are using any other service (e.g., Crashlytics, Yahoo's Flurry, or other analytics like those used to log events from a WebView), under data usage, please consider whether you should check "usage data" and "diagnostics".

Do you have any way to analyze individual app users?

- If yes, please consider whether the data types you collect are correctly marked as "linked to users".
- If you also share your analysis of app users with advertisers, please make sure you have correctly marked "data used to track users".

Where do your users live?

- If your users live in the EU, California, or other places with privacy laws that define personal data, please consider your data practices with these laws in mind and think about whether you should mark relevant data types as "linked to users".

Figure 12. Reflection questions on *PLW* summary page

Appendix B. Interview Protocol with the initial *Privacy Label Wiz*

Introduction and setup

Hello, my name is []. Thank you for your interest in our research study. Let's begin with setting up the study environment. In a moment I will provide a link to a GitHub repository where we are hosting the software tool we will ask you to run as a part of this study. When I post the link in the repository, please feel free to take a look at the repository if you'd like. Now, in your terminal, please navigate to a directory where you'd be willing to clone this repository. Once you're there, you can enter this command to clone our branch.

Post in Zoom chat: `git clone -b <branch name>. github url`

Now that the repository has been cloned, we recommend setting up a virtual environment to run our software tool. If you have a preferred method to set up an environment, you may use that. Alternatively, we've provided a short bash script in the cloned repository that will set up a Python virtual environment for you and automatically run the application. The script is called "startpfp.sh", you can run this in your terminal with "bash startplw.sh".

Post in Zoom chat: `bash startplw.sh`

Now the app should be started, developer is viewing the consent form

Now, please take a moment to read over this page and let me know if you have any questions.

Consent form filled out

We will begin recording shortly. At this time, please share your screen and turn off your camera.

If privacy labels were filled out, continue to use PLW

If not filled out

Do you have about an hour of time available? We'd like to ask you to fill out the privacy details for your application before we ask you to use our software tool.

Wait as developer fills out the app privacy details without assistance

Using *PLW*

Now, please take a moment to familiarize yourself with the application. As you use the software tool today, please let us know if you find particular parts of our software tool confusing, and I'll do my best to clarify.

If asked to explain layout:

The right hand side of the application lists the data types that the software tool detected in your application's code. First party types, i.e., data types detected in your application's code are listed first, followed by the data types detected in any third party libraries that the software tool sees you're using.

The left hand side of the application guides you through the process of thinking through your application's data collection

practices and then how that relates to completing Apple's privacy labels.
If asked to explain permissions (PLW Data Types):
The data types under "*PLW* Data Type" are the permissions that our software tool detects in your application's code, and we map those to the data types defined by Apple.
If asked to explain left hand side guidance:
First is an overview of the steps required to complete the app privacy details required by Apple. Then, we offer additional instructions for resources you may want to consult to better think about your data collection practices.
Next, let me ask you to go back to the app privacy details page for your application and make updates as necessary. As you go through the process of updating your details, may I ask that you mention any parts of this process that you find confusing?

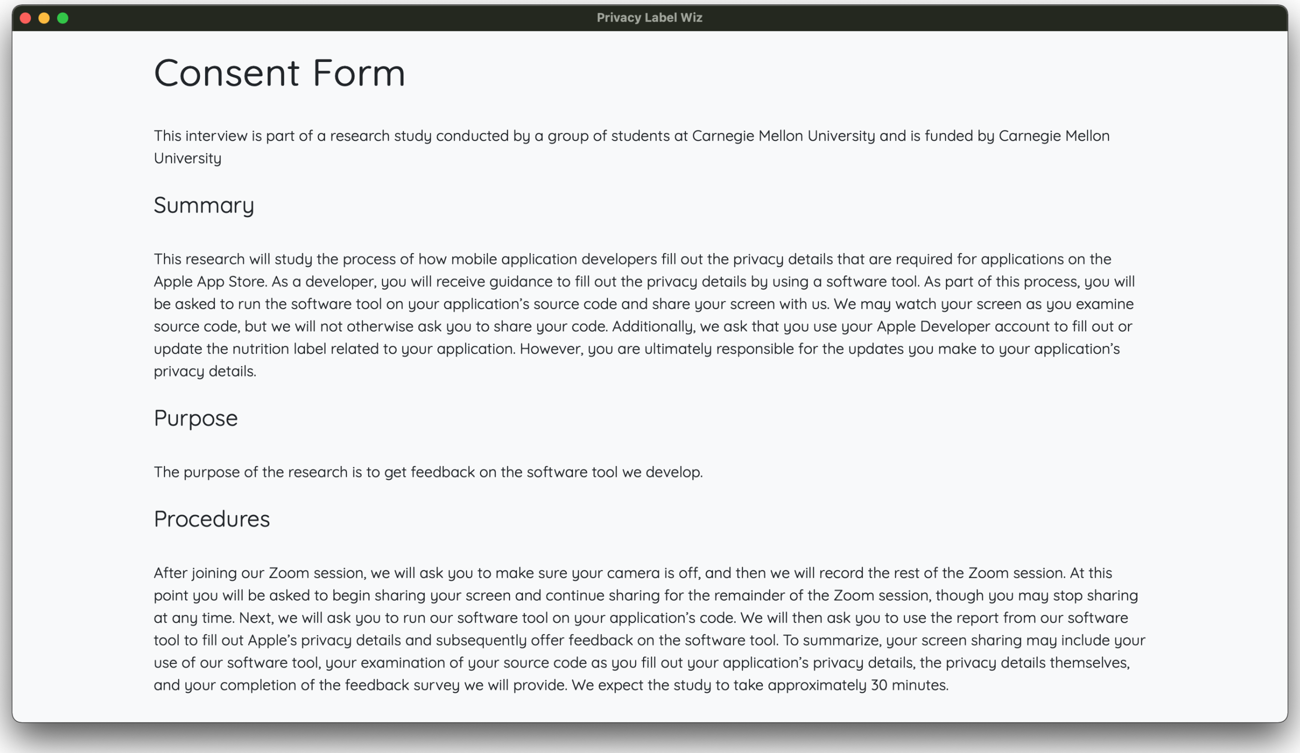
Privacy label process has been reviewed

Now we'll ask you to read through the reflection questions further down on the left hand side of the software tool. As you do this, please keep your privacy label interface open in case you want to make updates.

Closing

Before we conclude, would you like to offer any final thoughts on our software tool or on the process of filling out privacy labels?
Thank you very much for taking part in this interview. Have a great rest of your day.

Appendix C.
Additional screenshots of *PLW*



Consent Form

This interview is part of a research study conducted by a group of students at Carnegie Mellon University and is funded by Carnegie Mellon University

Summary

This research will study the process of how mobile application developers fill out the privacy details that are required for applications on the Apple App Store. As a developer, you will receive guidance to fill out the privacy details by using a software tool. As part of this process, you will be asked to run the software tool on your application's source code and share your screen with us. We may watch your screen as you examine source code, but we will not otherwise ask you to share your code. Additionally, we ask that you use your Apple Developer account to fill out or update the nutrition label related to your application. However, you are ultimately responsible for the updates you make to your application's privacy details.

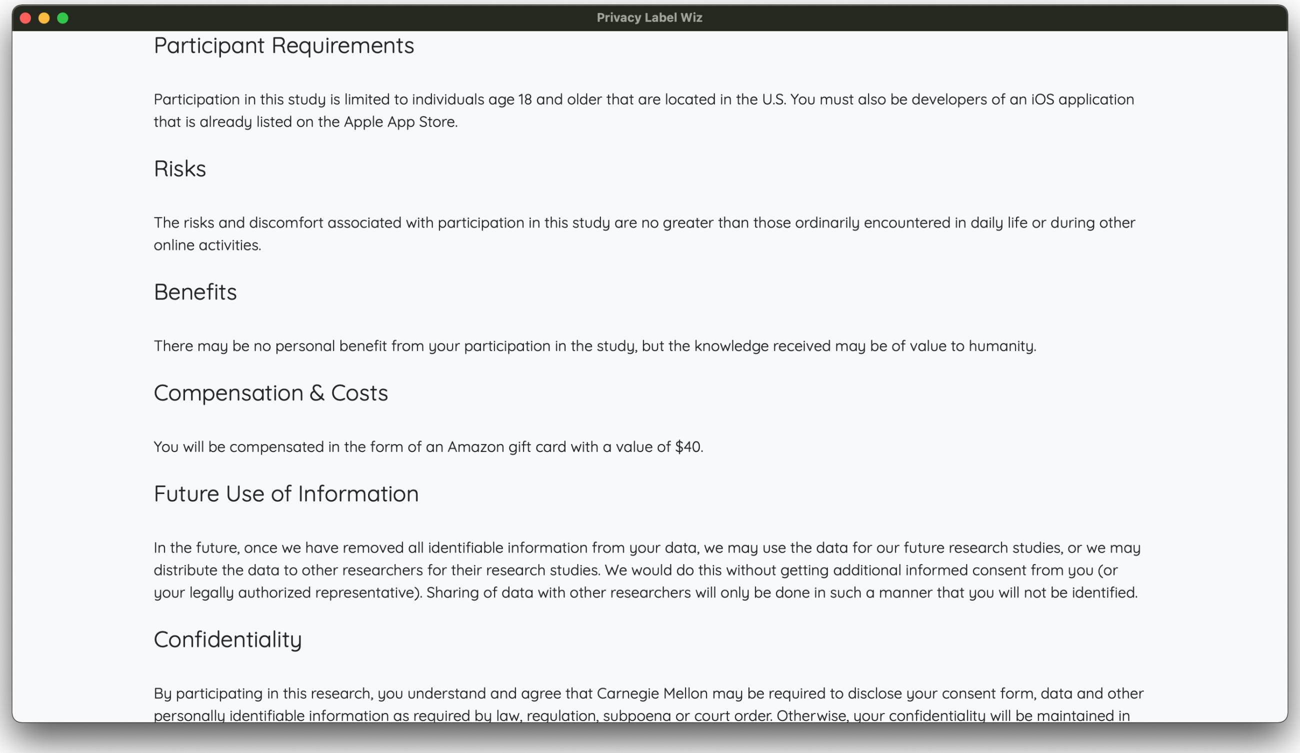
Purpose

The purpose of the research is to get feedback on the software tool we develop.

Procedures

After joining our Zoom session, we will ask you to make sure your camera is off, and then we will record the rest of the Zoom session. At this point you will be asked to begin sharing your screen and continue sharing for the remainder of the Zoom session, though you may stop sharing at any time. Next, we will ask you to run our software tool on your application's code. We will then ask you to use the report from our software tool to fill out Apple's privacy details and subsequently offer feedback on the software tool. To summarize, your screen sharing may include your use of our software tool, your examination of your source code as you fill out your application's privacy details, the privacy details themselves, and your completion of the feedback survey we will provide. We expect the study to take approximately 30 minutes.

Figure 13. Consent form 1



Participant Requirements

Participation in this study is limited to individuals age 18 and older that are located in the U.S. You must also be developers of an iOS application that is already listed on the Apple App Store.

Risks

The risks and discomfort associated with participation in this study are no greater than those ordinarily encountered in daily life or during other online activities.

Benefits

There may be no personal benefit from your participation in the study, but the knowledge received may be of value to humanity.

Compensation & Costs

You will be compensated in the form of an Amazon gift card with a value of $40.

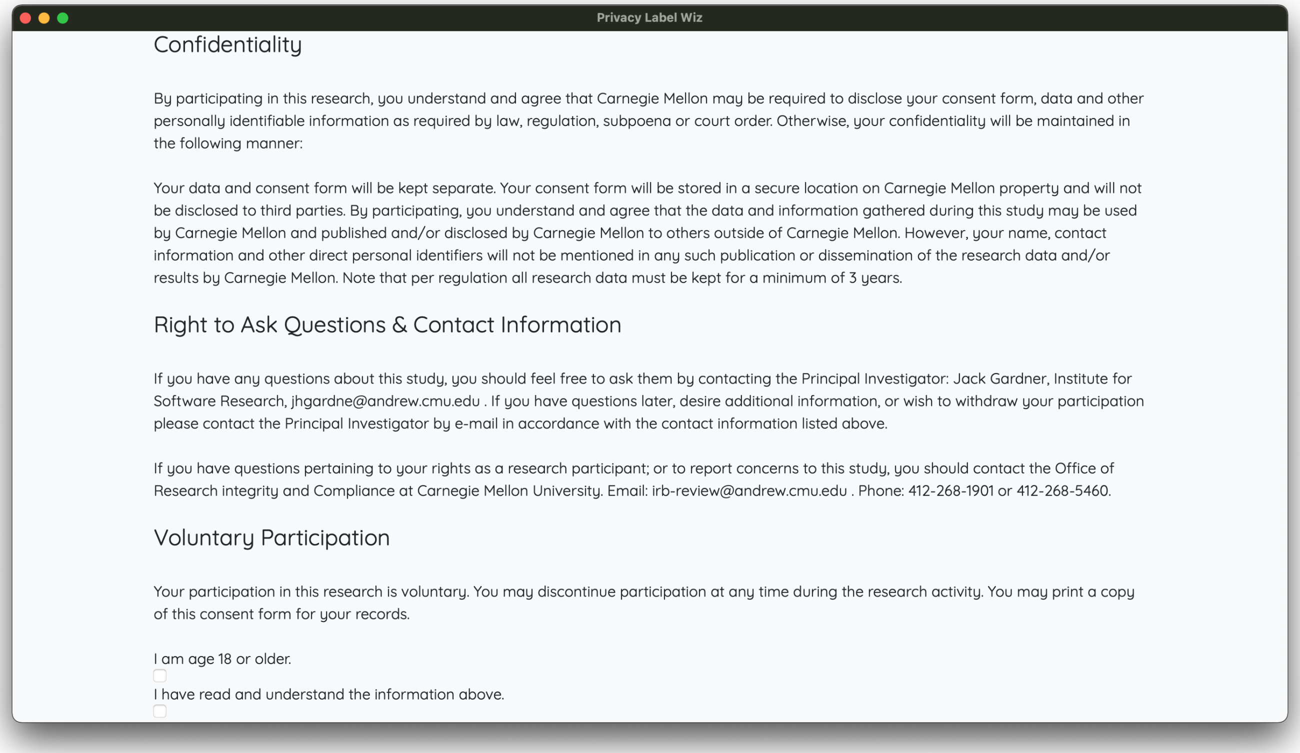
Future Use of Information

In the future, once we have removed all identifiable information from your data, we may use the data for our future research studies, or we may distribute the data to other researchers for their research studies. We would do this without getting additional informed consent from you (or your legally authorized representative). Sharing of data with other researchers will only be done in such a manner that you will not be identified.

Confidentiality

By participating in this research, you understand and agree that Carnegie Mellon may be required to disclose your consent form, data and other personally identifiable information as required by law, regulation, subpoena or court order. Otherwise, your confidentiality will be maintained in

Figure 14. Consent form 2



Confidentiality

By participating in this research, you understand and agree that Carnegie Mellon may be required to disclose your consent form, data and other personally identifiable information as required by law, regulation, subpoena or court order. Otherwise, your confidentiality will be maintained in the following manner:

Your data and consent form will be kept separate. Your consent form will be stored in a secure location on Carnegie Mellon property and will not be disclosed to third parties. By participating, you understand and agree that the data and information gathered during this study may be used by Carnegie Mellon and published and/or disclosed by Carnegie Mellon to others outside of Carnegie Mellon. However, your name, contact information and other direct personal identifiers will not be mentioned in any such publication or dissemination of the research data and/or results by Carnegie Mellon. Note that per regulation all research data must be kept for a minimum of 3 years.

Right to Ask Questions & Contact Information

If you have any questions about this study, you should feel free to ask them by contacting the Principal Investigator: Jack Gardner, Institute for Software Research, jhgardne@andrew.cmu.edu . If you have questions later, desire additional information, or wish to withdraw your participation please contact the Principal Investigator by e-mail in accordance with the contact information listed above.

If you have questions pertaining to your rights as a research participant; or to report concerns to this study, you should contact the Office of Research integrity and Compliance at Carnegie Mellon University. Email: irb-review@andrew.cmu.edu . Phone: 412-268-1901 or 412-268-5460.

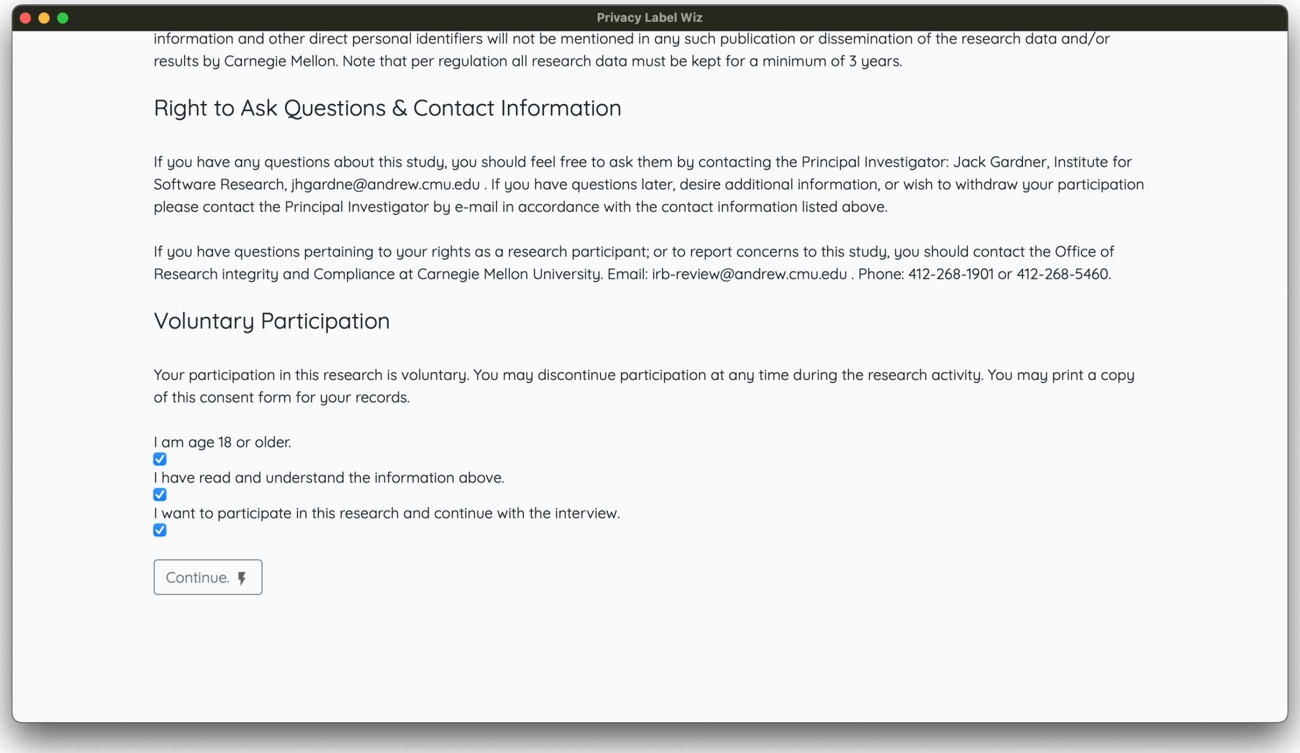
Voluntary Participation

Your participation in this research is voluntary. You may discontinue participation at any time during the research activity. You may print a copy of this consent form for your records.

I am age 18 or older.
□

I have read and understand the information above.
□

Figure 15. Consent form 3



information and other direct personal identifiers will not be mentioned in any such publication or dissemination of the research data and/or results by Carnegie Mellon. Note that per regulation all research data must be kept for a minimum of 3 years.

Right to Ask Questions & Contact Information

If you have any questions about this study, you should feel free to ask them by contacting the Principal Investigator: Jack Gardner, Institute for Software Research, jhgardne@andrew.cmu.edu . If you have questions later, desire additional information, or wish to withdraw your participation please contact the Principal Investigator by e-mail in accordance with the contact information listed above.

If you have questions pertaining to your rights as a research participant; or to report concerns to this study, you should contact the Office of Research integrity and Compliance at Carnegie Mellon University. Email: irb-review@andrew.cmu.edu . Phone: 412-268-1901 or 412-268-5460.

Voluntary Participation

Your participation in this research is voluntary. You may discontinue participation at any time during the research activity. You may print a copy of this consent form for your records.

I am age 18 or older.
☑
I have read and understand the information above.
☑
I want to participate in this research and continue with the interview.
☑

Figure 16. Consent form 4

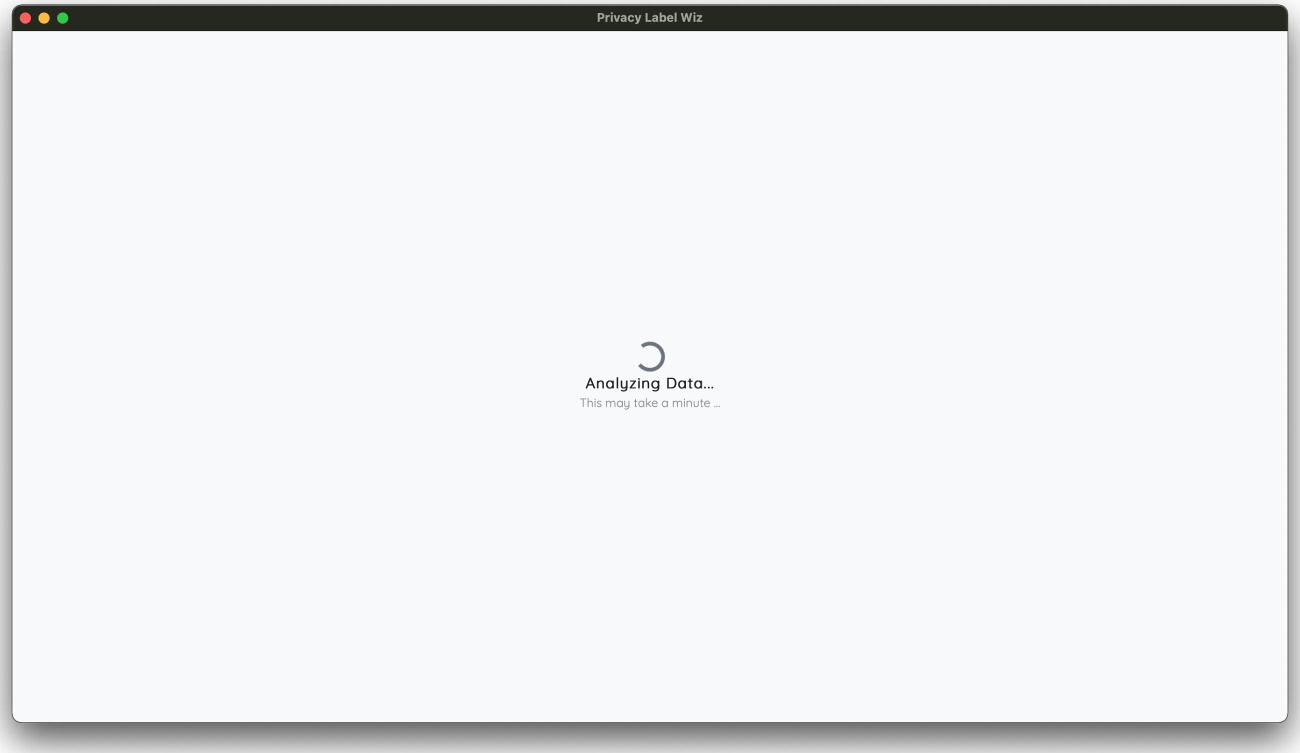
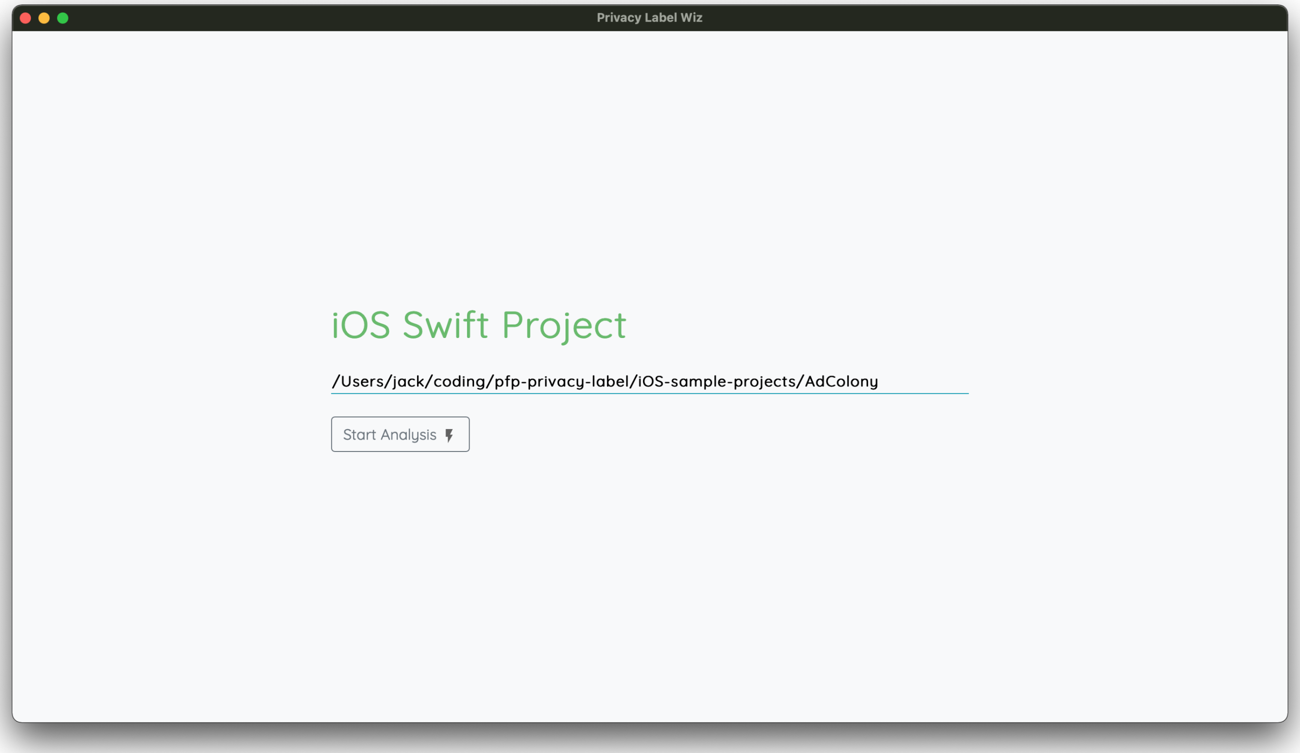


Figure 17. *PLW* running analysis

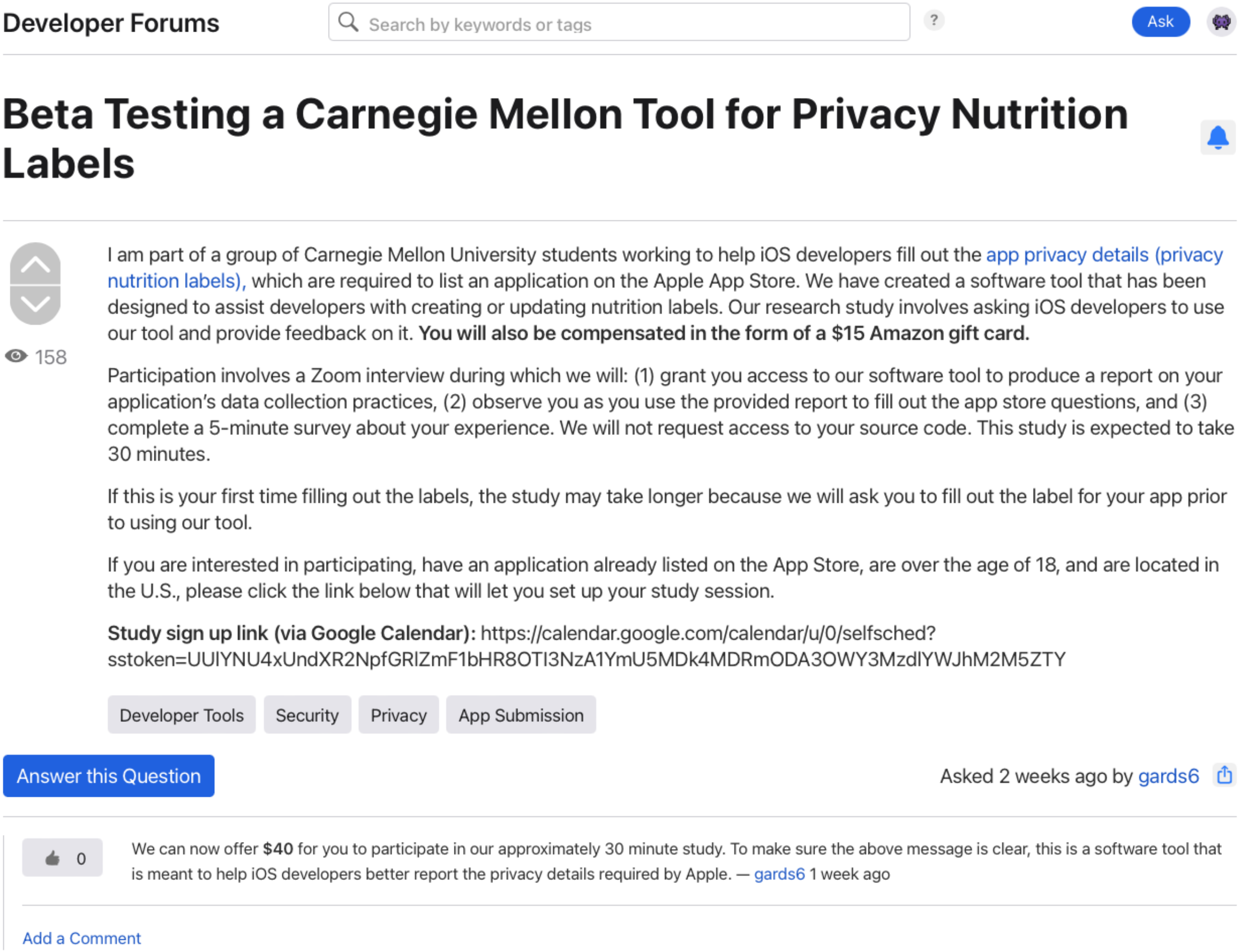


/Users/jack/coding/pfp-privacy-label/iOS-sample-projects/AdColony

Figure 18. Loading iOS project into *PLW*

Appendix D. Recruitment

As our recruitment process was iterative, we provide the messages we used to recruit developers from a variety of sources. These messages start at Fig. 19.



Beta Testing a Carnegie Mellon Tool for Privacy Nutrition Labels

I am part of a group of Carnegie Mellon University students working to help iOS developers fill out the app privacy details (privacy nutrition labels), which are required to list an application on the Apple App Store. We have created a software tool that has been designed to assist developers with creating or updating nutrition labels. Our research study involves asking iOS developers to use our tool and provide feedback on it. **You will also be compensated in the form of a $15 Amazon gift card.**

Participation involves a Zoom interview during which we will: (1) grant you access to our software tool to produce a report on your application's data collection practices, (2) observe you as you use the provided report to fill out the app store questions, and (3) complete a 5-minute survey about your experience. We will not request access to your source code. This study is expected to take 30 minutes.

If this is your first time filling out the labels, the study may take longer because we will ask you to fill out the label for your app prior to using our tool.

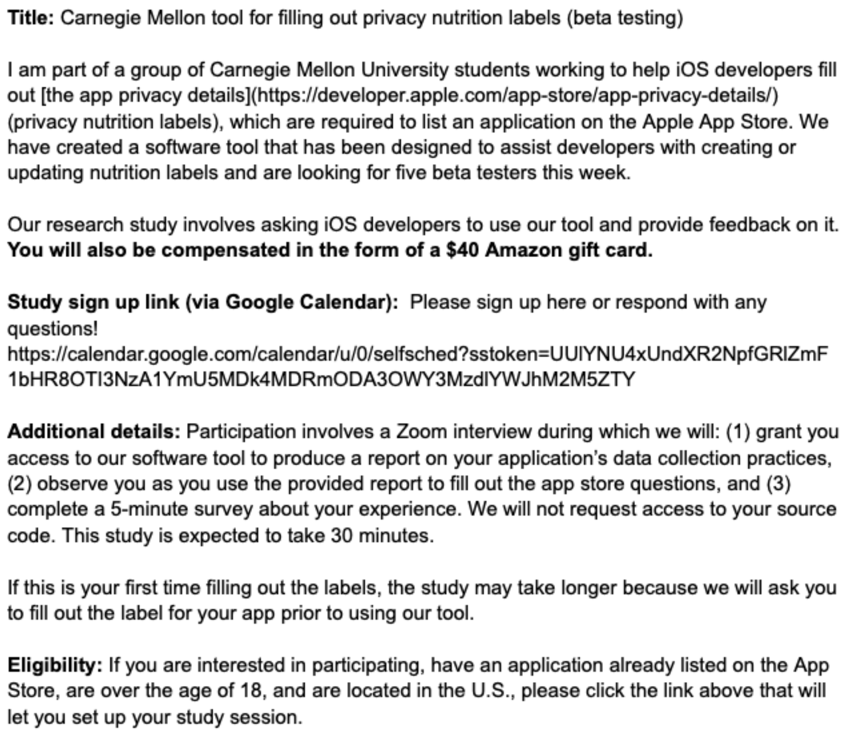
If you are interested in participating, have an application already listed on the App Store, are over the age of 18, and are located in the U.S., please click the link below that will let you set up your study session.

Study sign up link (via Google Calendar): https://calendar.google.com/calendar/u/0/selfsched?sstoken=UUlYNU4xUndXR2NpfGRlZmF1bHR8OTI3NzA1YmU5MDk4MDRmODA3OWY3MzdlYWJhM2M5ZTY

Asked 2 weeks ago by gards6

We can now offer **$40** for you to participate in our approximately 30 minute study. To make sure the above message is clear, this is a software tool that is meant to help iOS developers better report the privacy details required by Apple. — gards6 1 week ago

Figure 19. The first message we posted on Apple Developer Forums



Title: Carnegie Mellon tool for filling out privacy nutrition labels (beta testing)

I am part of a group of Carnegie Mellon University students working to help iOS developers fill out [the app privacy details](https://developer.apple.com/app-store/app-privacy-details/) (privacy nutrition labels), which are required to list an application on the Apple App Store. We have created a software tool that has been designed to assist developers with creating or updating nutrition labels and are looking for five beta testers this week.

Our research study involves asking iOS developers to use our tool and provide feedback on it. **You will also be compensated in the form of a $40 Amazon gift card.**

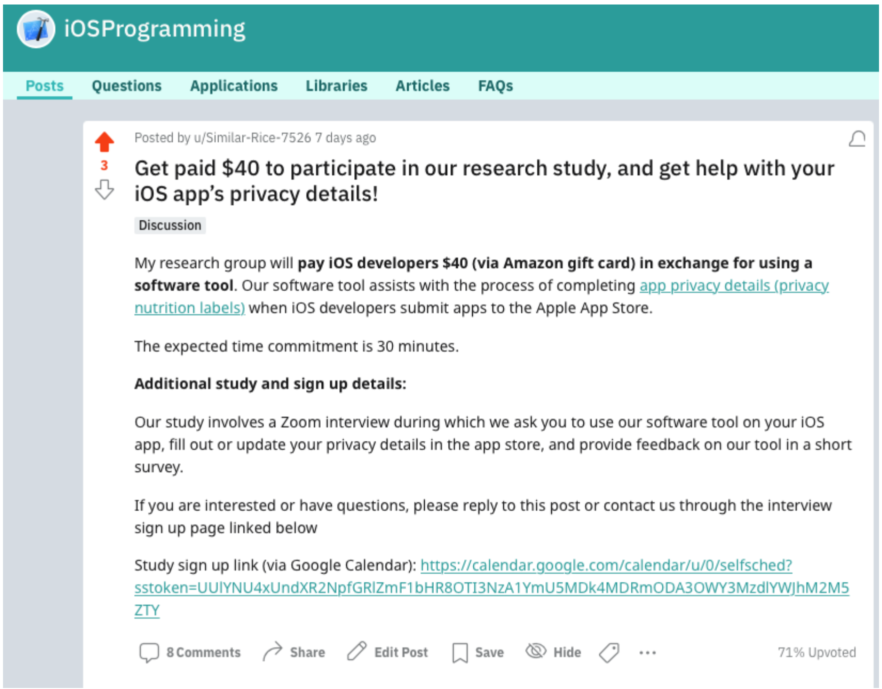
Study sign up link (via Google Calendar): Please sign up here or respond with any questions!
https://calendar.google.com/calendar/u/0/selfsched?sstoken=UUlYNU4xUndXR2NpfGRlZmF1bHR8OTI3NzA1YmU5MDk4MDRmODA3OWY3MzdlYWJhM2M5ZTY

Additional details: Participation involves a Zoom interview during which we will: (1) grant you access to our software tool to produce a report on your application's data collection practices, (2) observe you as you use the provided report to fill out the app store questions, and (3) complete a 5-minute survey about your experience. We will not request access to your source code. This study is expected to take 30 minutes.

If this is your first time filling out the labels, the study may take longer because we will ask you to fill out the label for your app prior to using our tool.

Eligibility: If you are interested in participating, have an application already listed on the App Store, are over the age of 18, and are located in the U.S., please click the link above that will let you set up your study session.

Figure 20. An adapted message we sent to Carnegie Mellon University Computer Science undergraduates



Posted by u/Similar-Rice-7526 7 days ago

Get paid $40 to participate in our research study, and get help with your iOS app's privacy details!

My research group will **pay iOS developers $40 (via Amazon gift card) in exchange for using a software tool**. Our software tool assists with the process of completing app privacy details (privacy nutrition labels) when iOS developers submit apps to the Apple App Store.

The expected time commitment is 30 minutes.

Additional study and sign up details:

Our study involves a Zoom interview during which we ask you to use our software tool on your iOS app, fill out or update your privacy details in the app store, and provide feedback on our tool in a short survey.

If you are interested or have questions, please reply to this post or contact us through the interview sign up page linked below

Study sign up link (via Google Calendar): https://calendar.google.com/calendar/u/0/selfsched?sstoken=UUlYNU4xUndXR2NpfGRlZmF1bHR8OTI3NzA1YmU5MDk4MDRmODA3OWY3MzdlYWJhM2M5ZTY

Figure 21. An abbreviated message we posted on the iOS Programming subreddit