Politecnico di Torino

Politecnico di Torino

Ingegneria Informatica

A.a. 2023/2024

Sessione di laurea Aprile 2024

PrivacyManager: gestione personalizzata della privacy su smartphone

Relatori:

Luigi De Russis

Juan Pablo Saenz Moreno

Candidato:

Alessio De Gregorio

Ringraziamenti

Ringrazio i miei genitori, che hanno sempre creduto in me e mi hanno sempre supportato, aiutandomi a superare ogni difficoltà.

Grazie a mio fratello e mia sorella, William e Rebecca, per essermi stati sempre accanto nonostante la distanza, alleggerendo il peso del percorso.

Grazie poi a tutta la mia famiglia, che ha sempre creduto in me e mi ha sempre dimostrato affetto e sostegno.

Infine, ringrazio Angela, per la pazienza, l'amore e l'incoraggiamento che mi ha dimostrato in ogni momento. La sua vicinanza e il suo supporto sono stati fondamentali. Grazie di cuore.

Indice

Elenco delle tabelle

Elenco delle figure

Capitolo 1

Introduzione

1.1 Contesto

Gli smartphone sono diventati parte integrante della nostra vita di tutti i giorni, offrendo comodità, connettività e accesso a vaste informazioni. Tuttavia, possono presentare potenziali problemi di privacy e sicurezza nel loro utilizzo quotidiano. Una delle preoccupazioni più frequenti, ad esempio, riguarda la raccolta dei dati che può essere messa in atto da parte delle aziende fornitrici delle applicazioni con lo scopo di rivenderle o di mostrare pubblicità mirata, oppure l'accesso alla localizzazione del dispositivo in maniera poco trasparente per l'utente [1].

Diversi studi sono stati condotti per analizzare la percezione e la fiducia che gli utenti finali *(end-user)* hanno riguardo la privacy e la sicurezza dei dati personali durante l'utilizzo del proprio smartphone. I risultati ottenuti dimostrano come la maggior parte degli utenti abbia una mancanza di fiducia nei confronti delle applicazioni che installa sul proprio dispositivo e sui mezzi messi a disposizione dal sistema operativo dello smartphone per gestire le autorizzazioni delle stesse e la protezione dei dati personali [1, 2, 3, 4]. Ad esempio, è stato mostrato come la maggior parte degli utenti intervistati creda che le impostazioni di sicurezza del proprio smartphone non siano sufficienti per potere gestire la privacy dei propri dati, a causa dello scarso controllo che percepiscono di avere. Infatti, una delle funzionalità più richieste riguarda l'ottenere un maggiore livello di controllo sul tipo di dati che vengono condivisi e la possibilità di scegliere quali condividere e quali no [4].

Il sistema Android, in particolare, utilizza il meccanismo delle autorizzazioni per fornire dei controlli che hanno l'obiettivo di aumentare la consapevolezza dell'utente e limitare l'accesso di un'app ai dati sensibili [5]. In particolare, dalla versione 6.0, del 2015, il modello è stato migliorato per poterne aumentare la comprensibilità per gli utenti. Infatti, a partire da questa versione vengono introdotte le autorizzazioni

richieste a runtime, in cui le app sono obbligate a richiedere all'utente in fase di esecuzione l'autorizzazione a determinate funzionalità che vengono considerate particolarmente sensibili (come l'autorizzazione *READ_CALENDAR*) [6].

L'introduzione di questo meccanismo si è rilevato efficace, in quanto è stato dimostrato come gli utenti tendano a prestare maggiore attenzione quando un'applicazione richiede di accedere a specifiche funzionalità, mostrando in determinati casi recalcitranza nel proseguire con l'utilizzo della stessa [3]. Si può dunque affermare che dei passi in avanti sono stati fatti con le versioni più moderne di Android, ma ancora non è sufficiente, vista l'assenza di un maggiore livello di granularità nella concessione (e revoca) delle autorizzazioni e la mancanza di trasparenza da parte delle applicazioni che le richiedono [3]. Infatti, pur avendo migliorato il modello delle autorizzazioni, la percezione degli utenti è ancora quella di possedere un controllo troppo ristretto riguardo i propri i dati e il modo in cui autorizzare o bloccare l'accesso ad essi [4].

Ciò che emerge maggiormente in questo quadro generale è la consapevolezza da parte degli utenti per quanto riguarda i rischi che possono essere correlati alla mancanza di sicurezza per i propri dati personali [2]. Tuttavia, ciò che manca principalmente è la capacità difendersi in maniera adeguata, oltre alla presenza di un sentimento di scarsa fiducia nei confronti dei mezzi che vengono messi a disposizione dagli smartphone [4]. Infatti, ciò che è stato spesso osservato è che, anche nel caso in cui gli utenti dichiarino di essere preoccupati per la loro privacy, poi nel concreto non agiscono per cercare di proteggerla, in alcuni casi per mancanza di conoscenza dei possibili mezzi di difesa e in altri per la preoccupazione di incontrare difficoltà nella ricerca degli stessi mezzi. Considerando anche i casi di maggiore consapevolezza e preoccupazione riguardo la protezione della propria privacy e la volontà di agire in futuro, infatti, l'aspettativa comune è quella di incontrare delle difficoltà nella gestione delle impostazioni del proprio smartphone. In particolare, si è mostrato come molti degli utenti intervistati non saprebbero nemmeno dove iniziare a guardare per trovare l'impostazione di loro interesse [1]. Collegato a questo punto, c'è anche una diffusa mancanza di fiducia nel fatto di potere gestire in maniera efficace i propri dati, rimanendo di fatto impotenti nella risoluzione del problema [1].

1.2 Obiettivi

Nello scenario descritto, il ruolo del proprietario dello smartphone, l'utente finale, che in genere non ha esperienza nello sviluppo o nella sicurezza informatica, è fondamentale per mantenere il dispositivo (e le sue informazioni) sicuro sotto molti punti di vista. Infatti, pur essendo stato introdotto il meccanismo delle autorizzazioni, e in generale gli aggiornamenti più recenti portino sempre maggiormente in

primo piano la sicurezza dei dati degli utenti, è necessario che siano questi ultimi ad avere conoscenza di cosa è possibile concedere e negare durante l'utilizzo delle applicazioni e imparare i corretti meccanismi che vengono messi a disposizione dai sistemi. La difficoltà di navigazione all'interno delle impostazioni del dispositivo è un problema portato alla discussione da parte di molti utenti, come accennato al paragrafo precedente. Diventa quindi importante fornire a questi degli strumenti che possano semplificarne l'utilizzo. Al contempo, è importante andare incontro alle richieste degli utenti che richiedono un maggiore controllo e una maggiore trasparenza riguardo i propri dati, avvertendoli in maniera adeguata quando questi vengono raccolti e in che modo.

Diversi lavori in letteratura dimostrano l'effettiva applicabilità delle tecniche di *End-User Development (EUD)* in diversi contesti, ad esempio ambienti mobile [7], smart home [8] e IoT [9]. Tuttavia, è stato fatto poco per quanto riguarda gli aspetti legati alla sicurezza e alla privacy nell'utilizzo degli smartphone.

L'obiettivo di questa tesi è la progettazione ed implementazione di un'applicazione mobile per dispositivi Android che possa aiutare gli utenti a proteggere la propria privacy e acquisire una maggiore consapevolezza riguardo le autorizzazioni che vengono concesse alle applicazioni installate sul proprio smartphone.

L'applicazione adotta l'approccio *End-User Development*. Il vantaggio principale di questo tipo di approccio è quello di permettere anche agli utenti meno esperti la definizione di misure di sicurezza per la protezione dei dati e della privacy sulla base di determinate condizioni selezionate dall'utente stesso. In questo modo, si lascia quindi a quest'ultimo un elevato grado di personalizzazione unito al tempo stesso ad una semplicità di utilizzo, evitando la richiesta di particolari competenze tecniche. Così facendo, ci si pone l'obiettivo di avvisare gli utenti in maniera consona del comportamento adottato dalle applicazioni installate sullo smartphone riguardo i propri dati personali e fornire loro uno strumento che permetta di intervenire in maniera meno articolata rispetto agli strumenti nativi forniti dal sistema operativo.

Lo scopo principale dell'applicazione è quindi duplice, ovvero, fornire uno strumento user-friendly per la gestione della privacy e della sicurezza dei propri dati, e permettere di personalizzare la gestione degli stessi a seconda dei parametri di maggiore interesse per l'utente finale.

1.3 Struttura della tesi

La tesi inizia con un'introduzione sulla gestione della sicurezza e della privacy degli utenti e sulle impressioni percepite al riguardo dagli stessi. Con un breve richiamo sulle tecniche di End-User Development si motiva così l'interesse allo sviluppo di questo progetto.

Nel capitolo 2 “Stato dell’arte” verrà trattata in maggiore dettaglio la tematiche dell’End-User Development, presentando alcune delle soluzioni esistenti. Al contempo, verranno analizzati i temi di sicurezza e privacy in ambiente mobile e il modo in cui gli utenti percepiscono questo aspetto.

Nel capitolo 3 “Analisi e progettazione” si passerà quindi ad una analisi dei maggiori fattori di rischio per gli utenti, focalizzando l’attenzione sul ruolo delle autorizzazioni nei sistemi Android. In questo modo, si potrà passare ad una definizione preliminare del sistema che verrà realizzato.

Nel capitolo 4 “Implementazione” verranno quindi illustrate nel dettaglio le scelte architetturali compiute e la realizzazione completa del sistema.

Il capitolo 5 “Valutazione” spiegherà poi il processo utilizzato per valutare l’usabilità del sistema, analizzando inoltre i dati raccolti e presentando i risultati.

Il capitolo 6, infine, concluderà il documento presentando alcune riflessioni finali, spiegando anche i limiti di questa ricerca e i possibili sviluppi futuri.

Capitolo 2

Stato dell'arte

2.1 End-User Development

Secondo la definizione proposta da Lieberman et al., l'End-User Development può essere descritto come un insieme di metodi, tecniche e strumenti che permettono agli utenti di sistemi informatici, che agiscono come sviluppatori software non professionisti, di creare, modificare o estendere un artefatto software [10]. Uno dei motivi principali nella diffusione di questo paradigma è dunque la possibilità lasciata ad utenti senza background tecnico specifico, di potere realizzare e personalizzare artefatti a seconda delle proprie esigenze. Una differenza principale con lo sviluppo software tradizionale, dunque, è che il risultato che si vuole raggiungere ha principalmente una utilità personale, piuttosto che un utilizzo pubblico [11].

L'utilizzo dell'End-User Development porta sicuramente dei vantaggi. Infatti, visto come nel mondo di oggi i sistemi informatici sono largamente diffusi in ogni aspetto della nostra quotidianità, è diventata una sfida complessa per gli sviluppatori soddisfare i bisogni di una vasta gamma di utenti in campi così diversificati, considerando la limitata conoscenza di certi domini e la lentezza nei processi di sviluppo. L'End-User Development può quindi essere un aiuto in questo ambito, visto che sono proprio gli utenti finali a conoscere il proprio contesto e i propri bisogni meglio di chiunque altro.

Tuttavia, nonostante i vantaggi descritti, è necessario discutere anche delle criticità di questo tipo di approccio. Infatti, se da un lato gli utenti finali sono coloro che conoscono meglio di chiunque altro i propri bisogni, dall'altro la mancanza di competenze tecniche specifiche porta a delle difficoltà nell'assicurare la qualità dell'artefatto prodotto, dando così origine a possibili problematiche. Il tema di maggiore gravità che può emergere in questo contesto è legato alla sicurezza del software prodotto. Infatti, se da un lato le debolezze in materia di sicurezza di un software sono da imputare alle cattive pratiche di programmazione degli

sviluppatori, non bisogna dimenticare il ruolo degli utenti dei sistemi e il modo in cui utilizzano gli stessi [12]. In questo senso, dare in mano ad utenti non esperti degli strumenti che non sono in grado di comprendere a pieno può risultare deleterio se non accompagnati in maniera adeguata.

È stato investigato, ad esempio, l'effetto dei task automation systems (*TAS*) nel campo dell'Internet of Things (IoT), utilizzati per la definizione di regole *trigger-action*. Infatti, se da un lato i TAS permettono la definizione di diversi comportamenti, dall'altro non si dedicano abbastanza agli aspetti che possono rendere i dispositivi smart vulnerabili a possibili minacce per la privacy e la sicurezza [13]. Inoltre, le regole create utilizzando piattaforme di questo tipo potrebbero nascondere comportamenti inaspettati di cui gli utenti finali potrebbero non accorgersi a causa della loro limitata conoscenza tecnica [14, 15].

Tenendo in considerazione i motivi sopra citati, dunque, diventa importante per gli sviluppatori rilasciare strumenti per l'End-User Development che possano essere in grado di garantire la realizzazione di artefatti software efficaci dal punto di vista della privacy e della sicurezza, senza però per questo inficiare la semplicità di utilizzo che caratterizza questo tipo di soluzioni.

2.1.1 Soluzioni esistenti

Le applicazioni dell'approccio End-User Development sono varie ed utilizzate in ambiti differenti. Tra i campi principali che è possibile citare, considerando quelli che hanno visto un largo impiego dell'EUD, ci sono sicuramente l'Internet of Things (IoT) e le smart home. Con la diffusione di questi due campi, infatti, la tendenza ricercata è stata quella di trasformare gli utenti finali da semplici consumatori a produttori attivi dei comportamenti dei propri ambienti smart.

Una smart home è definibile come un ambiente domestico che fa utilizzo di processori e strumenti (come i sensori) e attuatori (ad esempio, le luci domestiche) connessi tra di loro che automatizzano determinati comportamenti. Con IoT, invece, ci si riferisce ai dispositivi stessi che sono connessi attraverso internet. Negli anni il mercato di dispositivi IoT è cresciuto moltissimo, e ci si aspetta che per il 2030 il numero di dispositivi IoT connessi possa raggiunger i 125 miliardi [9].

Uno dei punti salienti dell'home automation è il fatto di come sia guidata dalle tecniche di EUD [8]. La popolarità dell'End-User Development è dovuta soprattutto alla diffusione di piattaforme intuitive orientate agli utenti. Tra le piattaforme più utilizzate ci sono i cosiddetti Task Automation Systems (*TAS*), i quali permettono di definire meccanismi di comportamento per dispositivi interconnessi tramite l'utilizzo del paradigma Trigger-Action Programming (*TAP*). In questo modo, viene data all'utente la possibilità di creare regole personalizzate del tipo Event-Condition-Action (*ECA*).

Tra le piattaforme di questo tipo, una delle più diffuse è IFTTT (*if-this-then-that)* [16]. Tale servizio permette di creare delle funzionalità connettendo tra di loro diversi dispositivi IoT in maniera personalizzata, e codificando il comportamento specifico tramite la definizione di semplici regole *if-then*. In questo modo, l'utente che definisce una nuova regola seleziona il componente di trigger, che descrive l'evento che attiva la regola, e l'azione, che definisce il comportamento da effettuare una volta scatenato il trigger.

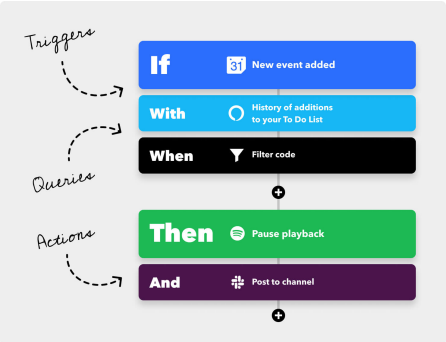


Figura 2.1: Definizione di un applet con IFTTT. Immagine tratta da IFTTT [16]

Tuttavia, la semplicità di utilizzo di tale piattaforma non è accompagnata da un altrettanto adeguato livello di protezione per gli utenti. Infatti, non è previsto un meccanismo di supervisione nella definizione della regola, e ciò potrebbe portare all'esposizione involontaria di informazioni e dati privati riguardo l'ambiente domestico dell'utente [17]. A tale proposito, si è analizzato infatti come circa il 50% delle regole definite dagli utenti (*recipes*) sono potenzialmente pericolose, esponendo possibili rischi di secrecy violation e/o integrity violation [18].

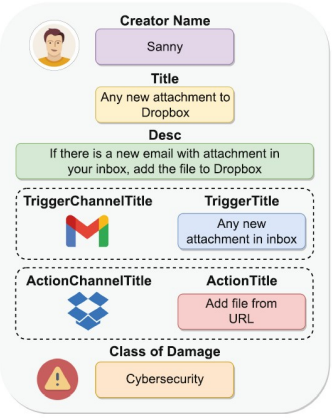


Figura 2.2: Esempio di applet possibilmente rischioso. Immagine tratta da Breve et al. [17]

Per potere limitare i rischi derivanti dall'utilizzo di piattaforme di questo tipo da parte di utenti non esperti, sono state proposte alcune soluzioni che hanno l'obiettivo di migliorare la protezione dei dati degli utenti. Tra queste, è possibile citare iRULER, che permette di identificare le vulnerabilità presenti nelle regole definite [19], A3ID, che è in grado di rilevare conflitti presenti tra diverse regole che interferiscono fra loro [20], e ProvThings, che permette di analizzare la provenienza dei dati all'interno delle attività del sistema e segnalare possibili comportamenti malevoli [21].

Per quanto riguarda gli ambienti mobile, invece, la diffusione dell'End-User Development è stata più limitata. Da uno studio condotto da Barricelli et al. [22] per l'analisi delle applicazione dell'EUD nei diversi ambiti, si può infatti osservare come l'impiego in ambiente mobile rappresenti una minoranza rispetto alle altre soluzioni esistenti (solo il 6% del totale).

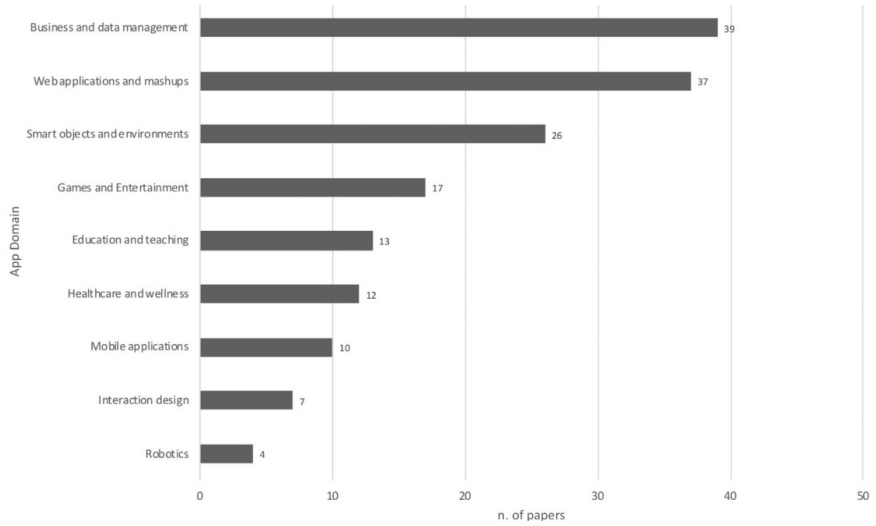
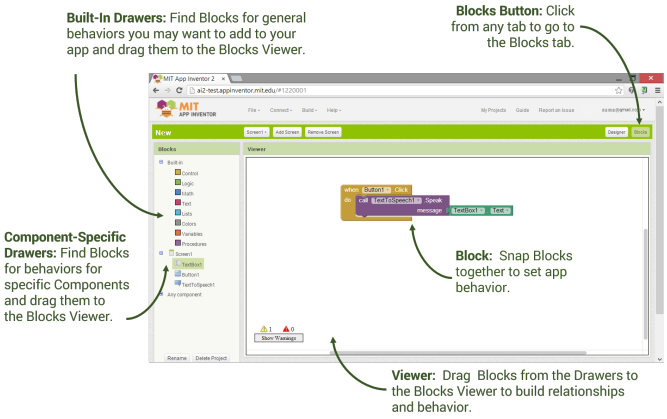
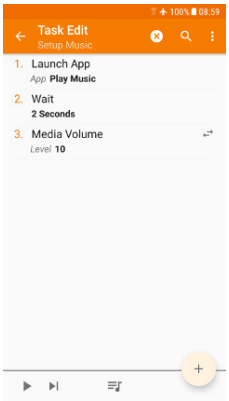


Figura 2.3: Diffusione dell'EUD in diversi ambiti. Immagine tratta da Barricelli et al. [22]

Tra le soluzioni esistenti che vale la pena citare, è presente MIT App Inventor [23], che permette agli utenti lo sviluppo di semplici applicazioni mobile per ambienti Android e iOS tramite l'utilizzo di un approccio block-based, che facilita l'utilizzo per tutti i tipi di utenti, compresi i bambini. Un'altra soluzione diffusa è Tasker [24], che permette la definizione di un insieme di azioni basate su determinati contesti definiti dall'utente. Tuttavia, visto il poco impiego dell'approccio EUD in ambienti mobile, è stato mostrato come soluzioni di questo tipo non abbiano riscosso grande diffusione tra gli utenti a causa della mancanza di consistenza nella terminologia adottata e nelle difficoltà di utilizzo degli strumenti proposti [25].



(a) Immagine tratta da MIT App Inventor [23]



(b) Immagine tratta da Tasker [24]

Figura 2.4: End-User Development in ambienti mobile

2.2 Sicurezza e privacy in ambito mobile

La privacy delle informazioni si può definire come il diritto di individui, gruppi o istituzioni di determinare autonomamente quando, come e in quale misura le informazioni che li riguardano siano comunicate ad altri [26]. In ambito di dispositivi mobile connessi in rete, tale definizione può estendersi riferendosi alla possibilità che hanno gli utenti di controllare l'accesso a parte dei propri dati personali [2]. Con l'estesa diffusione degli smartphone e, contemporaneamente, la crescita costante del mercato delle applicazioni mobili, la collezione di dati e informazioni degli utenti da parte delle aziende fornitrici di servizi è aumentata di pari passo. Diventa dunque importante, in un contesto del genere, attuare delle misure che permettano agli utenti di proteggere in maniera adeguata i propri dati. Esiste infatti, da un lato, il rischio concreto che gli utenti non dispongano dei mezzi necessari per capire la natura dei dati raccolti [27], e dall'altro il comportamento delle aziende che tendono ad ostracizzare la diffusione di meccanismi di controllo più user-friendly e intuitivi con l'obiettivo di raccogliere più informazioni possibili sui dati degli utenti [28].

In un contesto del genere, inoltre, bisogna anche tenere conto dei problemi di sicurezza associati alle applicazioni mobile. Con il vasto mercato delle app, infatti, criminali informatici potrebbero indurre gli utenti all'installazione di applicazioni malevole. In questo modo, gli utenti meno esperti potrebbero rimanere vittime di malware, ransomware, furto di identità e furto di dati privati. Per questo motivo diventa necessario fornire dei meccanismi di sicurezza per gli utenti con l'obiettivo di proteggere le informazioni salvate sui loro dispositivi. Tenendo in considerazione che i criminali informatici spesso sfruttano falle di sicurezza presenti

in software sviluppato da terzi, molti problemi possono comunque sorgere anche tenendo in considerazioni software sicuri dal punto di vista del design. Infatti, è sempre il fattore umano a rimanere l'anello più debole nell'ambito della sicurezza informatica [29].

Per mitigare i rischi dei problemi esposti e informare gli utenti riguardo ciò che possono fare per proteggere la propria sicurezza e la propria privacy, il sistema Android ha introdotto il meccanismo delle autorizzazioni [5]. Le autorizzazioni vengono utilizzate dalle applicazioni per potere richiedere l'accesso a determinate funzionalità del dispositivo. Fino ad Android 5.1 (*Android Lollipop*, 2014), l'utente, prima di installare l'app sul dispositivo, veniva informato riguardo tutte le funzionalità richieste per permettere all'app di funzionare correttamente. L'unica possibilità data all'utente era quella di accettare tutte le condizioni proposte, oppure rifiutarsi e in questo modo non potere installare l'applicazione. A partire dalla versione 6.0 (*Android Marshmallow*, 2015), invece, si ha l'introduzione delle autorizzazioni richieste a runtime, in cui l'applicazione chiede all'utente l'accesso alla funzionalità specifica solamente durante l'esecuzione dell'applicazione, evitando il modello "tutto o niente" della versione precedente. Dalla versione Android 10 (2019), inoltre, il modello è stato ulteriormente espanso, fornendo agli utenti una maggiore trasparenza su quali applicazioni dispongono di quali autorizzazioni ed essendo introdotta la possibilità di concedere l'autorizzazione sempre, solo durante l'utilizzo, oppure negarla [6].

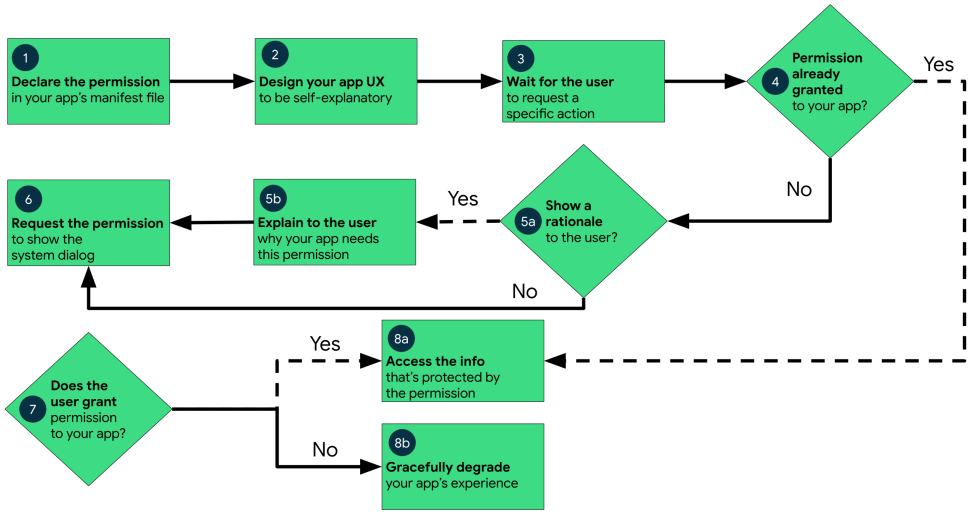


Figura 2.5: Processo di richiesta autorizzazioni a runtime. Immagine tratta dalla documentazione Android [30]

È stato dimostrato come gli aggiornamenti proposti da Android siano stati efficaci, in quanto gli utenti hanno comunicato come il modello delle autorizzazioni

a runtime fosse più intuitivo ed efficace per capire meglio il ruolo di ciascuna funzionalità richiesta [31]. Tuttavia, pur avendo evidenziato dei passi in avanti in questo senso grazie all'evoluzione del modello, la criticità principale rimane quanto effettivamente gli utenti siano in grado di comprendere gli avvisi relativi a ciascuna autorizzazione. Infatti, secondo uno studio in cui sono state sottoposte 906 domande relative al ruolo di ciascuna autorizzazione a 302 partecipanti totali, solamente il 21% delle risposte sono risultate essere pienamente corrette [3]. Un altro fattore emerso, inoltre, riguarda anche il comportamento di una parte degli utenti che non presta la dovuta attenzione alle funzionalità richieste, procedendo ad accettare tutte le autorizzazioni che vengono domandate. Un atteggiamento di questo tipo è stato definito come *warning fatigue* (stanchezza da avviso) e deriva dal fatto di mostrare degli avvisi che non comunicano dei rischi reali per gli utenti, portando questi ultimi a credere che allora nessun avviso è importante [3].

2.2.1 Percezioni degli utenti

Per progettare degli strumenti che hanno lo scopo di migliorare la sicurezza e la privacy degli utenti è necessario capire le loro preoccupazioni e le aspettative di utilizzo di soluzioni simili. Solo analizzando questi aspetti, infatti, è possibile fornire loro degli strumenti efficaci che possano sopperire alle loro mancanze dal punto di vista tecnico.

Un primo aspetto da prendere in considerazione riguarda il rapporto che hanno gli utenti con gli strumenti nativi offerti dal sistema operativo del proprio smartphone. È vero infatti che, come analizzato in precedenza, gli aggiornamenti portati avanti da Android per rendere più user-friendly il proprio sistema di gestione della sicurezza e privacy ha portato dei vantaggi, avendo avuto dei riscontri positivi mostrati dagli utenti se paragonati alle versioni precedenti. Tuttavia, a livello più generale, il livello di fiducia da parte degli utenti nei confronti delle misure di privacy a disposizione non ha ancora raggiunto dei livelli sufficienti. Nello specifico, il livello di controllo percepito dagli utenti sui dati memorizzati sul proprio dispositivo in 3 casi su 4 è ancora ritenuto insufficiente [4]. Considerando invece gli strumenti messi a disposizione dal sistema, emerge come molti utenti non siano a conoscenza dell'esistenza di determinate impostazioni oppure creda che determinate impostazioni di privacy siano già attive in maniera predefinita, come ad esempio il blocco della profilazione a scopo pubblicitario [1]. Al contempo, una parte degli utenti comunica di percepire delle difficoltà nella navigazione delle impostazioni del dispositivo, diffondendo per questo motivo la convinzione secondo cui siano le aziende produttrici a rendere consapevolmente difficile trovare e cambiare determinate impostazioni, in maniera tale da massimizzare i profitti derivanti dalla raccolta dei dati [1].

Considerato ciò, è importante interrogarsi anche riguardo quello che gli utenti ritengono maggiormente importante proteggere e in che modo, in maniera tale da andare incontro alle loro richieste negli strumenti che è possibile fornire. Infatti, il livello di rischio percepito dagli utenti non è uniforme per quanto riguarda le autorizzazioni richieste dalle app. Secondo uno studio condotto da Fung et al. [2], tra le autorizzazioni considerate più rischiose dagli utenti rientrano l'accesso alla localizzazione e l'accesso ai file del dispositivi, a discapito di altre funzionalità come l'accesso a foto e video, considerato meno rischioso. Per quanto riguarda i sensori dello smartphone, invece, viene considerato più pericoloso l'accesso al microfono e alla fotocamera rispetto all'accesso alla rete mobile. Le preoccupazioni maggiori derivanti da ciò riguardano maggiormente la raccolta di dati e la profilazione in maniera indesiderata da parte delle aziende, l'accesso alle password salvate sul dispositivo e ai propri dati bancari.

Quello che emerge, dunque, è che gli utenti sono consapevoli di alcuni dei rischi che possono derivare dalla concessione di determinate funzionalità, ma nonostante ciò sono presenti delle difficoltà nell'utilizzo degli strumenti messi a disposizione dal proprio smartphone per difendersi in maniera accurata. Una delle soluzioni adottate da una parte degli utenti, per evitare tale ostacolo, consiste nell'evitare alla radice determinati comportamenti, come ad esempio il salvataggio di dati personali nel dispositivo, la visita di siti web considerati poco sicuri o il download di determinate app [1]. In questo contesto, dunque, diventa importante fornire agli utenti degli strumenti efficaci che possano guidarli in maniera adeguata nella protezione dei propri dati, venendo incontro alle loro difficoltà e aspettative.

Capitolo 3

Analisi e progettazione

3.1 Fattori di rischio per la sicurezza e la privacy

Con la grande diffusione degli smartphone nel mondo e l'espansione sempre maggiore del mercato delle app mobile, è stato necessario per il sistema Android implementare nuove misure di sicurezza e migliorare quelle già esistenti con l'obiettivo di proteggere la sicurezza e la privacy dei propri utenti. Infatti, di seguito alla crescita di questo mercato, di pari passo è aumentato anche il numero di app che presentavano comportamenti malevoli con lo scopo di danneggiare la sicurezza dell'utente o di compromettere la privacy dei suoi dati. Secondo uno studio del 2014 [32], si è stimato che il 92% delle 500 app più scaricate dal Google Play Store portavano con sé potenziali rischi di sicurezza e privacy, e che il 42% di queste presentava componenti malevole che non avrebbero dovuto essere ammesse nei dispositivi degli utenti. Per risolvere questi problemi, durante gli anni Google ha deciso di cambiare le norme di sicurezza sul proprio store, rendendole più stringenti. Infatti, oggi, tutte le app messe in commercio all'interno dello store vengono sottoposte a scansioni automatiche in maniera tale da identificare potenziali comportamenti malevoli e, eventualmente, rimuoverle [33].

Si può quindi affermare che dei passi in avanti sono stati fatti con l'obiettivo di proteggere gli utenti e la sicurezza del loro ambiente mobile. Tuttavia, bisogna anche tenere in considerazione altri fattori. Infatti, nei sistemi Android è possibile l'installazione di applicazioni provenienti da store alternativi, che potrebbero dunque avere misure meno stringenti rispetto al Google Play Store. Inoltre, pur essendo efficace il processo di scansione e rimozione attuato da Google per le app del proprio store, si tratta comunque di un processo automatico basato su tecniche di machine learning, e solamente in un secondo momento viene effettuato un controllo manuale, il che potrebbe risultare in falsi negativi nel caso di malware molto recenti o particolarmente efficaci.

Anche in questo caso, dunque, la responsabilità maggiore riguardo la protezione dei propri dati ricade sulle spalle degli utenti finali, i quali presentano l'onere di scegliere se installare o meno determinate applicazioni sul proprio smartphone. Tuttavia, per alleggerire gli utenti di tale peso, la misura messa in atto dal sistema Android riguarda, come discusso in precedenza, il meccanismo delle autorizzazioni, le quali possono essere utili per bloccare alla base determinati comportamenti potenzialmente dannosi.

3.1.1 Android e la gestione delle autorizzazioni

Il meccanismo di gestione delle autorizzazioni è stato introdotto dal sistema Android con l'obiettivo di fornire dei controlli che aumentano la consapevolezza dell'utente e limitano l'accesso di un'app ai dati sensibili [5]. In questo modo, l'intento posto è quello di offrire maggiore protezione contro le applicazioni potenzialmente dannose. Il sistema Android, infatti, fa utilizzo del sandboxing delle applicazioni, in maniera tale da isolare le app le une dalle altre e limitare l'ambiente in cui possono operare, impedendo loro di accedere a determinati dati o funzionalità. L'application sandbox è implementato a livello di kernel, e in questo modo viene esteso a tutto il software posto sopra di esso, incluse librerie di sistema e applicazioni utente [34]. Con questa architettura, dunque, l'unico modo che hanno le applicazioni per potere accedere alle funzionalità del sistema o ad alcuni dati memorizzati è fare una richiesta esplicita per la relativa operazione e ottenere il consenso da parte dell'utente.

Non tutte le autorizzazioni sono equivalenti tra di loro. Ciò che il sistema impone è che le autorizzazioni necessarie per l'app vengano dichiarate esplicitamente nel file sorgente *Manifest* dell'applicazione stessa. La differenza che può essere presente è rappresentata dal modo in cui l'app richieda l'autorizzazione all'utente. Principalmente, è possibile suddividere le varie autorizzazioni in 5 categorie fondamentali, le quali differiscono in base al momento in cui vengono richieste all'utente e da quanto possibilmente elevato può essere il rischio per la sicurezza o la privacy dell'utente:

- *install-time*: vengono concesse automaticamente al momento dell'installazione, senza richiedere il conenso all'utente. Implicano scarsi rischi per la sicurezza e la privacy. Un esempio di questo tipo è l'autorizzazione *android.permission.INTERNET*;
- *normal*: permettono l'accesso a dati e funzionalità che si estendono oltre il sandbox dell'applicazione, ma non sono comunque considerate rischiose per l'utente. Anche in questo caso, vengono concesse automaticamente senza una richiesta esplicita. Un esempio di questo tipo è l'autorizzazione *android.permission.FOREGROUND_SERVICE*;
- *runtime*: sono considerate ad alto rischio per l'utente e richiedono una notifica esplicita per essere concesse. Tali autorizzazioni non vengono concesse al

momento dell'installazione, ma sono richieste solamente quando l'app ne richiede l'utilizzo per la prima volta. Un autorizzazione di questo tipo è *android.permission.CAMERA*;

- *special*: sono considerate molto sensibili e rischiose per l'utente e per essere concesse è necessario che l'utente navighi nella specifica pagina delle impostazioni di sistema per autorizzare l'app in questione. Un esempio di questo tipo è l'autorizzazione *BIND_NOTIFICATION_LISTENER_SERVICE*;

- *signature*: si tratta di autorizzazioni che vengono concesse solamente alle app che sono state firmate con la stessa signature key del creatore dell'autorizzazione. Si tratta di autorizzazioni usate per proteggere le funzionalità più sensibili delle app di sistema. Un esempio di questo tipo è l'autorizzazione *android.permission.MANAGE_EXTERNAL_STORAGE*.

Si può dunque notare come l'architettura delle autorizzazioni sia composta da diversi livelli che vengono incontro alla protezione degli utenti senza però appesantire lo sforzo richiesto a questi per l'utilizzo delle applicazioni. Il principio adottato, infatti, è che se un'autorizzazione non comporta un rischio concreto per l'utente allora non è necessario stressarlo mostrando degli avvisi che non fanno riferimento ad un problema concreto. Invece, man mano che cresce la pericolosità della funzionalità richiesta, allora è chiesto al contempo uno sforzo sempre maggiore all'utente per potere concedere l'autorizzazione in questione.

3.1.2 Autorizzazioni e rischi per la privacy

Nonostante i passi in avanti fatti nelle nuove versioni di Android e il miglioramento strutturale nella gestione delle autorizzazioni, non è ancora possibile ignorare le criticità derivanti dall'attuale sistema. Infatti, è vero che il modello di richiesta delle autorizzazioni a runtime si è rivelato efficace, soprattutto se comparato al modello "tutto o niente" delle prime versioni di Android. Tuttavia, una criticità ancora presente consiste nel livello di granularità intrinseco nelle autorizzazioni stesse.

Per comprendere meglio questo problema, si può ad esempio tenere in considerazione l'autorizzazione relativa alla localizzazione del dispositivo dell'utente, come affrontato anche in diversi lavori in letteratura [35, 36]. Tale funzionalità, infatti potrebbe essere necessaria per fornire il servizio essenziale di una determinata applicazione che ne richiede l'uso, ma potrebbe anche essere utilizzata in maniera poco trasparente per l'utente e per scopi differenti rispetto a quelli che questo effettivamente si aspetta. L'accesso alla localizzazione, infatti, potrebbe essere utilizzato per tracciare gli spostamenti dell'utente senza il suo consenso esplicito, con lo scopo di profilarlo a scopo pubblicitario oppure per conto di aziende terze.

In questo modo, l'applicazione non avrebbe problemi ad accedere a tale funzionalità, essendo stata autorizzata precedentemente dall'utente. In una condizione del genere, dunque, le uniche alternative presenti sono solamente quella di accettare la situazione in maniera passiva, oppure rifiutare il proprio consenso, ma in questo modo non potere più utilizzare i servizi forniti dall'applicazione. Si tratta dunque di un ritorno al problema iniziale dell'approccio "tutto o niente" delle vecchie versioni del sistema Android.

Quello che manca in questo contesto, dunque, è un maggiore controllo per l'utente. Quest'ultimo, infatti, potrebbe anche non essere a conoscenza di questo comportamento adottato dalle applicazioni che utilizza, con un conseguente rischio per la propria privacy. Quello che è evidente in questo contesto, dunque, è l'impossibilità per il sistema di differenziare in maniera efficace le varie casistiche di utilizzo per le autorizzazioni concesse, e la mancanza di consapevolezza per l'utente riguardo l'utilizzo delle app che usa abitualmente.

3.2 Soluzione proposta

Tenendo in considerazione i problemi presenti negli attuali sistemi mobile e le lacune che presentano le soluzioni esistenti, oltre ai vantaggi che ha offerto l'approccio End-User Development in altri contesti, si è ricavata l'idea dell'applicazione ***PrivacyManager***.

Il principio alla base dell'applicazione è quello di lasciare all'utente la libertà di creazione di *regole di sicurezza* secondo i parametri di suo interesse. Una volta creata e attivata la regola, il sistema agirà in maniera trasparente per l'utente, operando in background un'operazione di monitoraggio per verificare che la regola venga rispettata. Qualora ci sia un'applicazione che violi la regola creata, dunque, il sistema si farà carico di notificare l'utente e permettergli di decidere come intervenire.

Considerati i problemi derivanti dal modello delle autorizzazioni di Android e le lacune presenti in esso, si è deciso di porre come base per la definizione della regola di sicurezza proprio le *autorizzazioni*. Come discusso nelle sezioni precedenti, non tutte le autorizzazioni che Android rende disponibili sono ugualmente critiche o rischiose per gli utenti. Sarebbe quindi inutile per gli utenti definire delle misure di salvaguardia basate su delle funzionalità che non compromettono in alcun modo la propria sicurezza. Per questo motivo, seguendo anche il lavoro proposto da Liu et al. [37], si è deciso di tenere in considerazione solamente quelle autorizzazioni considerate rischiose per la sicurezza e la privacy degli utenti. Di seguito è quindi riportata una sintesi riguardante tali autorizzazioni:

Autorizzazione	**Descrizione**	**Possibile rischio**
Localizzazione	Posizione geografica del dispositivo	Attacchi o malware location-based; pubblicità location-based
Telefono	Informazioni e caratteristiche associate alla telefonia, come informazioni sul registro chiamate o sul numero di telefono	Rischio di privacy
Contatti	Lista di contatti telefonici dell'utente	Divulgazione di informazioni sui contatti
Calendario	Attività registrate sul calendario dell'utente	Divulgazione di informazioni sugli impegni dell'utente
SMS	Lettura e invio di SMS	Invio di messaggi senza consapevolezza dell'utente, possibilmente per l'iscrizione a servizi a pagamento
Fotocamera	Cattura di immagini	Accesso alla funzionalità senza consapevolezza dell'utente
Microfono	Registrazione di conversazioni	Rischio di privacy per utilizzo del sensore senza consapevolezza dell'utente
Memoria	Modifica all'unità di archiviazione interna o esterna	Furto di informazioni o salvataggio indesiderato di dati

Tabella 3.1: Autorizzazioni rischiose per la sicurezza e la privacy dell'utente

Una volta posta questa base, si può quindi passare alla definizione di una seconda componente essenziale legata alla prima, ovvero le *applicazioni*. L'idea alla base dell'applicazione, infatti, è quella di avvertire l'utente qualora un'app stia usando un'autorizzazione critica in determinate condizioni. Una volta avvertito di ciò, l'utente può quindi decidere di intervenire come meglio crede. Si è deciso, per questo punto, di lasciare all'utente la possibilità di scegliere il piano di *azione* che preferisce secondo due alternative:

- *segnalazione*: il sistema notifica l'utente che un'applicazione sta utilizzando un'autorizzazione sensibile. Una volta avvertito di ciò, viene anche data all'utente la possibilità di revocare facilmente l'autorizzazione in questione senza la

necessità di navigare fra i menu delle impostazioni di sistema, semplificandone dunque il processo;

- *arresto*: il sistema rileva che un'applicazione sta utilizzando un'autorizzazione sensibile, la arresta e notifica l'utente dell'avvenimento.

È possibile quindi schematizzare il funzionamento dell'intero sistema per come segue:

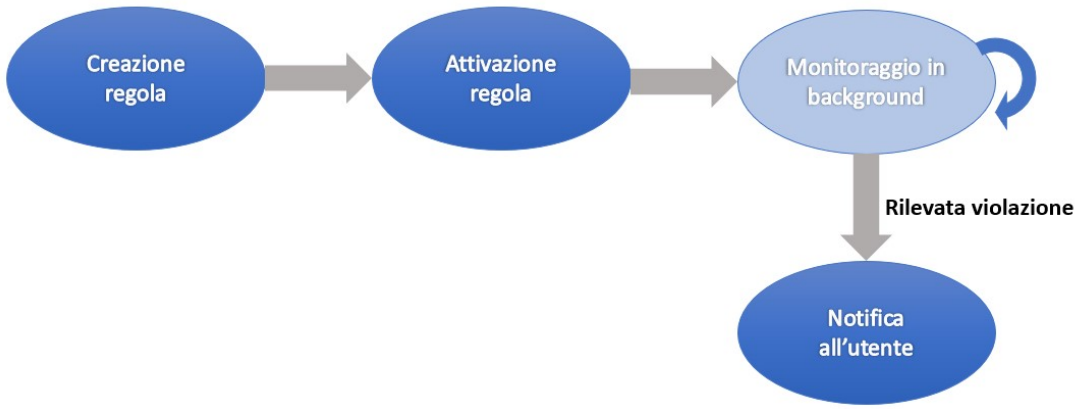


Figura 3.1: Funzionamento del sistema

Definito dunque questo flusso di base, all'utente viene inoltre data la possibilità, in fase di creazione della regola, di definire un insieme di condizioni a cui deve sottostare il sistema per potere attivare il monitoraggio. Vengono quindi applicati in questo contesto i principi fondamentali dell'End-User Development, secondo cui all'utente finale viene data la possibilità di estendere e personalizzare l'artefatto software secondo le proprie esigenze, realizzando dunque un prodotto con un'utilità strettamente personale. Di seguito vengono quindi riportate le *condizioni* che è possibile personalizzare e viene spiegato in che modo è influenzato di conseguenza il processo di monitoraggio:

Condizione	**Descrizione**	**Attivazione del monitoraggio**
Giorno e ora	Giorni della settimana e relativo orario	Durante i giorni e l'orario specificati
Posizioni	Locazione geografica specificata dall'indirizzo inserito	Quando l'utente si trova fisicamente in uno dei luoghi specificati

Condizione	**Descrizione**	**Attivazione del monitoraggio**
Rete	Nome di una (o più) reti Wi-Fi, oppure la connessione dati del dispositivo	Quando il dispositivo è connesso alla rete specificata
Bluetooth	Dispositivi bluetooth memorizzati nel dispositivo	Quando uno (o più) fra i dispositivi selezionati sono collegati allo smartphone
Batteria	Percentuale di carica del dispositivo	Quando il livello di carica del dispositivo è inferiore a quello specificato

Tabella 3.2: Condizioni personalizzabili dall'utente

Dunque, si può schematizzare il processo di creazione di una regola tenendo in considerazione la specifica dei parametri seguenti mostrati in figura:

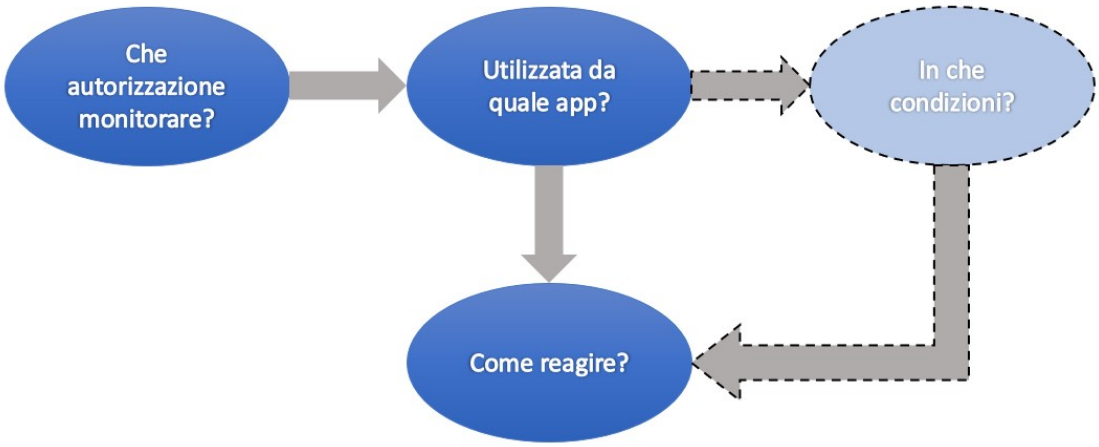


Figura 3.2: Flusso di creazione di una regola

3.2.1 Limiti delle funzionalità offerte dal sistema Android

All'inizio del suo ciclo vitale il sistema Android era caratterizzato dall'essere un sistema molto aperto. La filosofia alla base adottata da Google, infatti, era quella di fornire un sistema flessibile e personalizzabile, in cui gli sviluppatori avevano grandi libertà. Da un punto di vista, questo approccio ha senz'altro contribuito alla diffusione del mercato delle app mobile, ma dall'altro era anche fonte di possibili rischi per la sicurezza dei propri utenti. Con la crescita del mercato delle app,

infatti, aumentava anche il rischio di incorrere in app che presentavano componenti malevole che non venivano in alcun modo bloccate dal sistema operativo. Per questo motivo, la decisione presa da Google negli anni è stata quella di offrire una maggiore protezione per la sicurezza e la privacy dei propri utenti a discapito di alcune libertà nello sviluppo delle applicazioni. Come già discusso nelle sezioni precedenti, è in questo contesto che nasce e si diffonde il meccanismo delle autorizzazioni. A fianco a questo aspetto, inoltre, è stata anche eliminata del tutto la possibilità per gli sviluppatori di accedere a determinate funzionalità (liberamente accessibili in passato) in quanto considerate troppo rischiose per la sicurezza e la privacy degli utenti. Un esempio di questo tipo è l'accesso ai file di sistema. Infatti, da una parte, tale funzionalità lascia sicuramente grande libertà per la personalizzazione del dispositivo, ma dall'altra, se concessa ad un'applicazione malevola, potrebbe compromettere gravemente la sicurezza dell'utente.

Questi limiti posti da Android, dunque, hanno avuto delle ripercussioni per quanto riguarda lo sviluppo dell'applicazione *PrivacyManager*. L'idea alla base del sistema, come detto, è quella di monitorare il comportamento delle app del dispositivo ed avvertire l'utente nel caso in cui (in determinate condizioni personalizzabili) una di queste stia facendo uso di una funzionalità ritenuta critica per la sua sicurezza o privacy. Nella tabella 3.1 si è discusso di quali siano nel concreto queste funzionalità. Tuttavia, a causa dei limiti appena discussi, solamente un insieme ristretto di queste risulta essere effettivamente monitorabile senza la concessione di permessi di root.

L'elenco delle autorizzazioni prese quindi in considerazione nello sviluppo, si riduce al seguente:

Autorizzazione	**Descrizione**	**Possibile rischio**
Localizzazione	Posizione geografica del dispositivo	Attacchi o malware location-based; pubblicità location-based
Calendario	Attività registrate sul calendario dell'utente	Divulgazione di informazioni sugli impegni dell'utente
Fotocamera	Cattura di immagini	Accesso alla funzionalità senza consapevolezza dell'utente

Tabella 3.3: Autorizzazioni considerate per lo sviluppo dell'applicazione

Affiancata a queste, tuttavia, si è anche deciso di tenere conto delle *notifiche* inviate dalle app del dispositivo. Infatti, seppur non strettamente rischiose per la sicurezza dell'utente, potrebbero compromettere la sua privacy in determinate

condizioni. Per questo motivo si è deciso di includerle nell'elenco delle autorizzazioni che è possibile monitorare, vista anche la flessibilità fornita tramite le condizioni personalizzabili dall'utente in fase di definizione della regola. In questo caso, le *azioni* definibili nel caso in cui il sistema riveli l'arrivo di una nuova notifica, accompagnate alle altre discusse in precedenza, sono le seguenti:

- *oscura*: la notifica inviata dall'app non viene mostrata. Al suo posto, il sistema segnala all'utente l'arrivo di una nuova notifica senza specificarne il contenuto. Una volta aperta la segnalazione, viene mostrato all'utente il suo contenuto effettivo e l'app che l'ha inviata;

- *blocca*: la notifica inviata dall'app non viene mostrata. Al suo posto, il sistema segnala all'utente che un'app ha inviato una notifica che è stata bloccata, senza dare informazioni sul contenuto. Una volta aperta la segnalazione viene mostrato solamente qual è stata l'app ad inviare la notifica.

3.2.2 Casi d'uso

Di seguito sono riportati i casi d'uso principali dell'applicazione *PrivacyManager* con il relativo flusso degli eventi.

Segnalazione autorizzazione "Notifiche"

L'utente definisce i parametri necessari e dà inizio al monitoraggio delle applicazioni. Se un'applicazione invia una notifica violando uno o più dei parametri definiti in precedenza, questa viene segnalata all'utente e la notifica viene oscurata oppure bloccata.

Punto	Evento
1	L'utente accede al sistema
2	Il sistema mostra un elenco con una voce cliccabile per ciascuna autorizzazione. È inoltre presente un bottone "Inizia monitoraggio"
3	L'utente clicca sull'autorizzazione "Notifiche" e ha inizio l'esecuzione del caso d'uso di inclusione "Definizione parametri"
4	Il sistema mostra l'elenco delle autorizzazioni e il bottone "Inizia monitoraggio"
5	Inizia l'esecuzione del caso d'uso di inclusione "Rilevazione violazione"

Tabella 3.4: Segnalazione autorizzazione "Notifiche"

Segnalazione autorizzazione “Localizzazione”

L’utente definisce i parametri necessari e dà inizio al monitoraggio delle applicazioni. Se un’applicazione accede al permesso “Localizzazione” violando uno o più dei parametri definiti in precedenza, questa viene segnalata all’utente per mezzo di una notifica.

Punto	**Evento**
1	L’utente accede al sistema
2	Il sistema mostra un elenco con una voce cliccabile per ciascuna autorizzazione. È inoltre presente un bottone “Inizia monitoraggio”
3	L’utente clicca sull’autorizzazione “Localizzazione” e ha inizio l’esecuzione del caso d’uso di inclusione “Definizione parametri”
4	Il sistema mostra l’elenco delle autorizzazioni e il bottone ”Inizia monitoraggio“
5	Inizia l’esecuzione del caso d’uso di inclusione “Rilevazione violazione”

Tabella 3.5: Segnalazione autorizzazione “Localizzazione”

Segnalazione autorizzazione “Calendario”

L’utente definisce i parametri necessari e dà inizio al monitoraggio delle applicazioni. Se un’applicazione accede al permesso “Calendario” violando uno o più dei parametri definiti in precedenza, questa viene segnalata all’utente per mezzo di una notifica.

Punto	**Evento**
1	L’utente accede al sistema
2	Il sistema mostra un elenco con una voce cliccabile per ciascuna autorizzazione. È inoltre presente un bottone “Inizia monitoraggio”
3	L’utente clicca sull’autorizzazione “Calendario” e ha inizio l’esecuzione del caso d’uso di inclusione “Definizione parametri”
4	Il sistema mostra l’elenco delle autorizzazioni e il bottone ”Inizia monitoraggio“
5	Inizia l’esecuzione del caso d’uso di inclusione “Rilevazione violazione”

Tabella 3.6: Segnalazione autorizzazione “Calendario”

Segnalazione autorizzazione "Fotocamera"

L'utente definisce i parametri necessari e dà inizio al monitoraggio delle applicazioni. Se un'applicazione accede al permesso "Fotocamera" violando uno o più dei parametri definiti in precedenza, questa viene segnalata all'utente per mezzo di una notifica.

Punto	Evento
1	L'utente accede al sistema
2	Il sistema mostra un elenco con una voce cliccabile per ciascuna autorizzazione. È inoltre presente un bottone "Inizia monitoraggio"
3	L'utente clicca sull'autorizzazione "Fotocamera" e ha inizio l'esecuzione del caso d'uso di inclusione "Definizione parametri"
4	Il sistema mostra l'elenco delle autorizzazioni e il bottone "Inizia monitoraggio"
5	Inizia l'esecuzione del caso d'uso di inclusione "Rilevazione violazione"

Tabella 3.7: Segnalazione autorizzazione "Fotocamera"

Definizione parametri

Questo caso d'uso è comune a tutti gli altri. L'utente definisce 7 parametri: applicazioni, slot temporale (giorni e ora), posizioni, rete, bluetooth, batteria e azione. In base ai parametri impostati, l'utente ha la possibilità di definire un determinato flusso di monitoraggio, definito nei vari casi d'uso.

Punto	Evento
1	L'utente seleziona le voci relative alle autorizzazioni che vuole monitorare e clicca sul bottone "Avanti"
2	Il sistema mostra un form composto da 7 pagine che l'utente può navigare e compilare autonomamente. La prima pagina mostrata dal sistema è relativa alle "Applicazioni"
3	Il sistema recupera dal dispositivo le informazioni relative alle applicazioni che possono avere accesso alle autorizzazioni selezionate in precedenza dall'utente. Tali informazioni vengono mostrare in un elenco in cui ciascuna voce presenta una checkbox selezionabile
4	L'utente seleziona le applicazioni che vuole che siano monitorate dal sistema e prosegue nella pagina seguente
5	Il sistema salva le informazioni inserite dall'utente e mostra la pagina relativa a "Slot temporale"

Punto	Evento
6	L'utente seleziona i giorni e gli orari in cui vuole che il monitoraggio venga attivato e prosegue nella pagina seguente
7	Il sistema salva le informazioni inserite dall'utente e mostra la pagina relativa a "Posizioni"
8	L'utente aggiunge una posizione per cui, nel caso in cui si trovi fisicamente in quel luogo, il monitoraggio viene attivato
9	Il sistema salva le informazioni inserite dall'utente e mostra la pagina relativa a "Rete"
10	L'utente inserisce una rete per cui, alla cui connessione, il monitoraggio viene attivato
11	Il sistema salva le informazioni inserite dall'utente e mostra la pagina relativa a "Bluetooth"
12	Il sistema recupera dal dispositivo le informazioni relative ai dispositivi bluetooth memorizzati. Tali informazioni vengono mostrare in un elenco in cui ciascuna voce presenta una checkbox selezionabile
13	L'utente seleziona i dispositivi bluetooth per cui, al cui collegamento, il monitoraggio viene attivato
14	Il sistema salva le informazioni inserite dall'utente e mostra la pagina relativa a "Batteria"
15	L'utente inserisce un valore per cui, se la carica della batteria del dispositivo è inferiore ad esso, il monitoraggio viene attivato
16	Il sistema salva le informazioni inserite dall'utente e mostra la pagina relativa ad "Azione"
17	A seconda del fatto che l'utente abbia selezionato o meno l'autorizzazione "Notifiche" precedentemente, il sistema mostra due coppie differenti di alternative selezionabili: "Segnala applicazione" e "Chiudi applicazione", oppure "Segnala applicazione e oscura notifica" e "Chiudi applicazione e blocca notifica"
18	L'utente seleziona l'opzione che vuole venga attivata una volta che venga rilevata un'applicazione che violi la regola che è stata definita
19	Il sistema salva l'informazione relativa all'azione
20	L'utente clicca sul bottone "Salva" e torna alla schermata precedente

Tabella 3.8: Definizione parametri

Rilevazione violazione

Questo caso d'uso è comune a tutti gli altri. Il sistema rileva che un'applicazione sta violando una regola definita dall'utente in base ai parametri che ha inserito. In

base all'azione che è stata definita, viene eseguito un determinato comportamento.

Punto	**Evento**
1	L'utente clicca sul bottone "Inizia monitoraggio" "Azione" definita "Chiudi applicazione"
2	Il sistema rileva la presenza di un'app in esecuzione che sta violando una regola definita dall'utente e verifica il parametro "Azione" definito
3a	"Azione" definita "Segnala applicazione": • Il sistema invia una notifica all'utente segnalando la violazione; • L'utente clicca sulla notifica; • Il sistema mostra una schermata in cui viene spiegata la violazione e il nome dell'applicazione che causato il problema. Viene inoltre mostrato un bottone per risolvere il problema; • L'utente clicca sul bottone; • Il sistema apre le impostazioni del dispositivo alla pagina relativa alle autorizzazioni associate all'applicazione segnalata; • L'utente sceglie se disabilitare l'autorizzazione.
3b	"Azione" definita "Chiudi applicazione": • Il sistema arresta il processo relativo all'applicazione e invia all'utente una notifica; • L'utente clicca sulla notifica; • Il sistema mostra una schermata in cui viene spiegato la violazione e il nome dell'applicazione che è stata chiusa.

Punto	Evento
3c	"Azione" definita "Segnala applicazione e oscura notifica": • Il sistema invia una notifica all'utente segnalando che un'applicazione ha inviato una notifica che è stata oscurata; • L'utente clicca sulla notifica; • Il sistema mostra una schermata in cui viene spiegata la violazione, il nome dell'applicazione che causato il problema e il contenuto della notifica oscurata.
3d	"Azione" definita "Chiudi applicazione e blocca notifica": • Il sistema invia una notifica all'utente segnalando che un'applicazione ha inviato una notifica che è stata bloccata; • L'utente clicca sulla notifica; • Il sistema mostra una schermata in cui viene spiegata la violazione e il nome dell'applicazione che causato il problema.

Tabella 3.9: Rilevazione violazione

3.2.3 Struttura dell'applicazione

La funzionalità fondamentale dell'applicazione è rappresentata dalla definizione della regola di sicurezza. Per potere guidare l'utente in maniera semplice ed intuitiva durante questo processo, tenendo a mente i principi dell'End-User Development, si sono considerati gli approcci forniti da altre applicazioni simili, come descritto nella sezione 2.1.1 riguardante le soluzioni esistenti. Quello che si è voluto ottenere, dunque, è un processo che permettesse agli utenti di definire in maniera semplice ogni aspetto fondamentale della regola di sicurezza, basandosi sul flusso illustrato in figura 3.2. Il questionario da compilare per definire la regola, quindi, richiederà all'utente di precisare i parametri seguenti secondo questo ordine specifico:

- *autorizzazioni*: l'utente seleziona una o più autorizzazioni di suo interesse che desidera vengano monitorate. Si tratta del primo componente da definire per potere continuare nella specifica dei parametri seguenti;
- *applicazioni*: una volta selezionate le autorizzazioni, il sistema recupera dal dispositivo le informazioni relative alle app dell'utente che hanno accesso a tali autorizzazioni. A questo punto, l'utente potrà selezionare le app di suo

interesse. Quando una tra queste app farà utilizzo di un'autorizzazione presa in considerazione, allora il sistema segnalerà l'avvenimento all'utente. Anche questo è un parametro fondamentale per la definizione della regola di sicurezza;

- *condizioni*: si tratta delle condizioni personalizzabili descritte nella tabella 3.2. È un parametro opzionale e l'utente non ha la necessità di doverlo definire. Nel caso in cui, invece, voglia specificare delle condizioni, verrà mostrato un ulteriore form in cui è possibile specificare le condizioni di proprio interesse;

- *azione*: una volta definiti i parametri precedenti, l'utente dovrà selezionare il tipo di azione che dovrà eseguire il sistema nel caso in cui venga rilevata una violazione della regola di sicurezza. Le azioni selezionabili sono di segnalazione o di arresto e, nel caso in cui sia stata selezionata anche l'autorizzazione relativa alle notifiche, saranno presenti rispettivamente anche le azioni di oscuramento e di blocco della notifica.

Infine, una volta definiti tutti i parametri in questione, verrà richiesto all'utente anche l'inserimento di un nome per la regola creata, così da poterla salvare e mostrare successivamente.

A questo punto, le altre funzioni importanti da fornire all'utente riguardano la possibilità di vedere un riepilogo delle regole create e la capacità di avviare il monitoraggio tramite la regola in questione. Per questo motivo si è pensato di fornire all'utente una schermata di riepilogo che mostra un elenco contenente le regole salvate. Da qui è data la possibilità di vedere i dettagli relativi ai parametri specificati per la regola ed è anche presente un pulsante che permette l'avvio del monitoraggio, oppure, nel caso in cui la regola sia stata già attivata, il suo arresto. Inoltre, sempre in tale schermata, è anche fornita la possibilità all'utente di modificare i parametri della regola oppure procedere alla sua cancellazione.

Per quanto riguarda invece il processo di monitoraggio, questo opererà come un servizio eseguito in background una volta che una regola è attivata. Il servizio continuerà a funzionare e tenere traccia del comportamento delle applicazioni anche se l'utente chiude l'app *PrivacyManager*, così da garantire una protezione continuativa senza necessità di un intervento attivo da parte dell'utente.

Esiste anche un altro punto da tenere in considerazione, ed è quello relativo alle autorizzazioni richieste dal sistema. Per potere operare efficacemente, infatti, *PrivacyManager* ha bisogno di accedere ad alcune funzionalità del dispositivo che devono essere autorizzate dall'utente. Per questo motivo, si è deciso di creare un'ulteriore schermata che permetta all'utente di mostrare e concedere tutte le autorizzazioni necessarie. Tale schermata verrà presentata all'utente durante il primo avvio dell'applicazione oppure nel caso in cui una delle autorizzazioni fondamentali al funzionamento venga revocata.

Infine, si è deciso di inserire anche un'ultima funzionalità che possa essere utile a guidare l'utente durante l'utilizzo del sistema. Per questo motivo sono stati introdotti un insieme di *tutorial* che hanno lo scopo di presentare le funzioni del sistema e le varie sezioni presenti. I tutorial verranno avviate automaticamente nel caso in cui l'utente apra per la prima volta l'app, in maniera tale da accompagnarlo ed istruirlo adeguatamente. Inoltre, potranno anche essere richiamati successivamente premendo su uno specifico pulsante presente in ogni schermata.

Per quanto riguarda invece il salvataggio dei dati, si è deciso di optare per un semplice salvataggio in locale delle regole di sicurezza. Viene comunque fornita una funzionalità di registrazione per gli utenti e di raccolta dati in cloud, necessaria per valutare l'usabilità dell'applicazione ed ottenere delle statistiche riguardo l'utilizzo che ne viene fatto dagli utenti.

3.2.4 Prototipi delle schermate

A questo punto si può passare ad illustrare con maggiore dettaglio i prototipi delle schermate progettate, accompagnate da un relativo mockup.

All'apertura dell'applicazione, verrà mostrata all'utente la schermata di homepage, illustrata di seguito.

Figura 3.3: Homepage del sistema

In questo modo l'utente ha la possibilità di vedere un riepilogo di tutte le regole di sicurezza che ha salvato, dividendole tra quelle che sono state attivate per la fase di monitoraggio e quelle che sono momentaneamente inattive. Cliccando sulle relative icone è possibile effettuare la modifica dei parametri definiti precedentemente per

una regola oppure cancellarla. Sempre da questa schermata è possibile avviare il processo di monitoraggio e arrestarlo. Cliccando invece sul pulsante "Nuova regola" si avrà l'apertura del relativo form di creazione. Infine, cliccando sul punto interrogativo, verrà mostrato un tutorial all'utente per spiegargli nel dettaglio le varie sezioni.

Cliccando sul nome di una delle regole salvate e mostrate in homepage, si può quindi passare alla schermata di riepilogo in cui vengono mostrati i parametri definiti per la regola in questione.

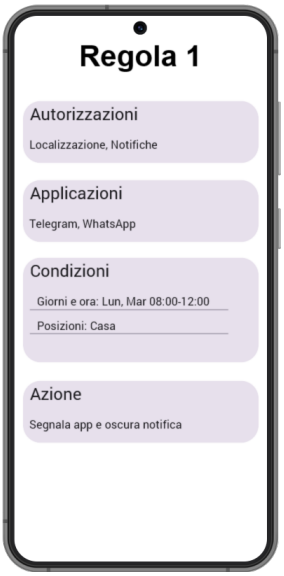


Figura 3.4: Dettagli della regola salvata

Si tratta di una semplice schermata riassuntiva a cui l'utente può accedere sia cliccando sulle regole non ancora attivate sia su quelle attualmente in fase di monitoraggio.

Cliccando dall'homepage, invece, sul pulsante relativo alla creazione di una nuova regola, verrà dunque mostrata la schermata seguente.

Figura 3.5: Form per la creazione della regola di sicurezza

Tramite il form in questione l'utente ha la possibilità di definire i parametri della regola di sicurezza che intenzione di creare. Lo scheletro della schermata è basato sullo schema illustrato in figura 3.2, la quale illustra il flusso di creazione della regola. Come mostrato nello schema, i parametri fondamentali da definire sono quelli relativi alle autorizzazioni, alle applicazioni e alle condizioni. Una volta specificati, l'utente ha la possibilità di salvare la regola, una volta assegnatole un nome. Per poterli specificare è sufficiente cliccare sui pulsanti relativi a ciascun parametro, e verrà quindi mostrata la schermata relativa alla sua definizione. Il parametro relativo alle condizioni, invece, è opzionale. Se l'utente non specifica nessuna condizione, allora viene assunto che abbia l'intenzione di mantenere il monitoraggio sempre in attività, una volta attivata la regola. Come nel caso dell'homepage, anche qui è presente un pulsante che è possibile premere per mostrare un tutorial della schermata in questione. In questo caso, oltre ad illustrare le diverse sezione, il tutorial servirà anche a spiegare nel dettaglio il meccanismo di funzionamento della regola di sicurezza e il ruolo di ciascun parametro.

Di seguito è dunque possibile vedere i prototipi relativi alle schermate di definizione delle autorizzazioni e delle app.

(a) Autorizzazioni **(b)** Applicazioni

Figura 3.6: Definizione parametri "Autorizzazioni" e "Applicazioni"

Prima di procedere alla specifica delle applicazioni, l'utente è richiesto di definire le autorizzazioni che intende monitorare. Una volta selezionate e salvate, potrà quindi passare alla schermata relativa alle applicazioni. Prima di mostrare l'elenco delle app, il sistema recupererà dal dispositivo le informazioni relative solamente alle app che richiedono l'accesso alle autorizzazioni selezionate dall'utente nella prima fase.

Una volta definiti e salvati entrambi i parametri, l'utente potrà dunque passare alla definizione al parametro di azione.

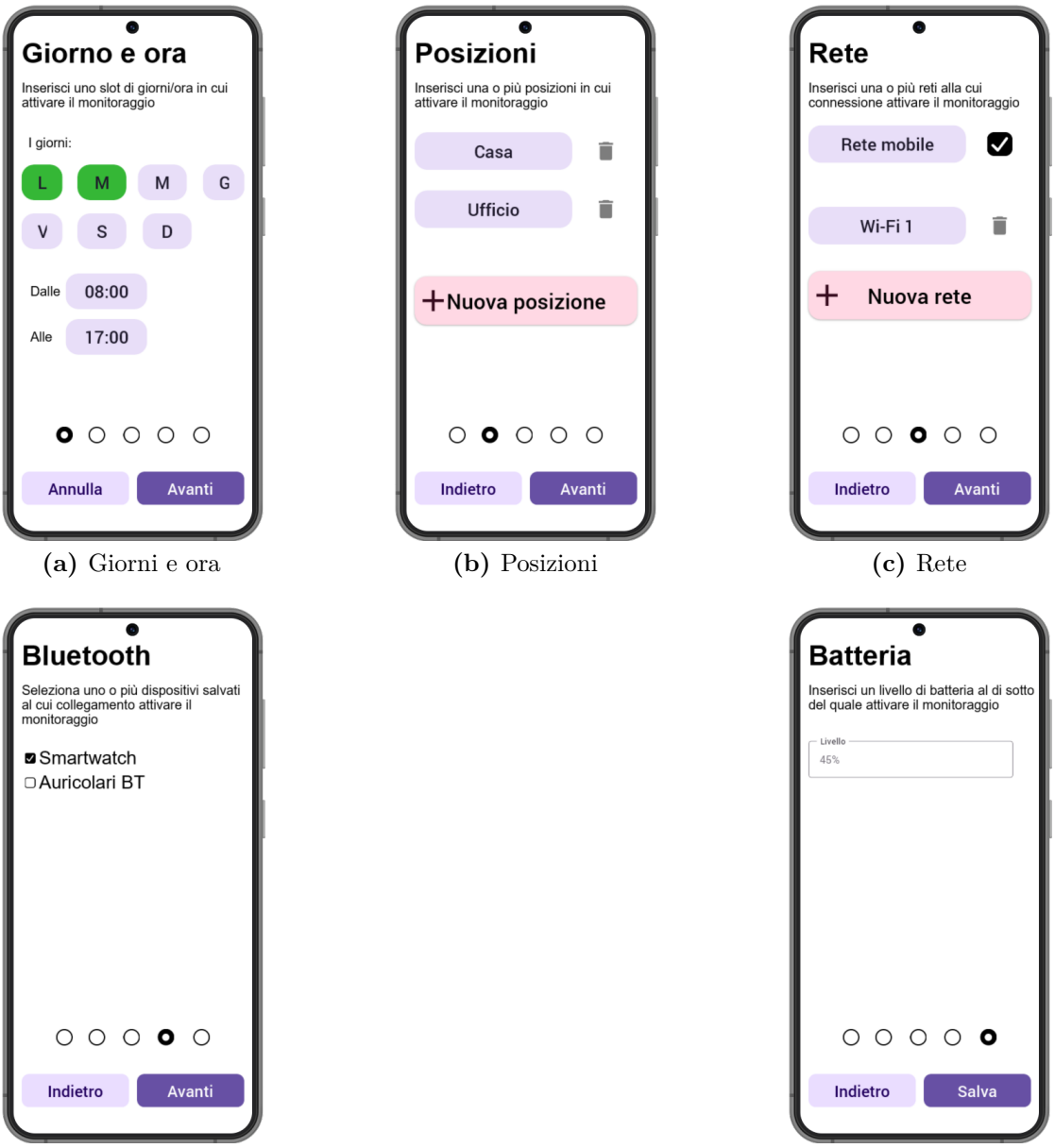
(a) Azione con notifiche

(b) Azione senza notifiche

Figura 3.7: Definizione parametro "Azione"

Com'è possibile osservare, l'azione definibile dall'utente è differente nel caso in cui, tra le autorizzazioni che intende monitorare, abbia selezionato o meno le notifiche. Infatti, avrebbe poco senza segnalare o arrestare un'app che ha inviato una notifica. Si è deciso quindi di optare per questa soluzione, lasciando la possibilità all'utente di oscurarla oppure bloccarla.

Oltre i tre parametri descritti, l'utente ha anche la possibilità di definire le condizioni personalizzabili illustrate nella tabella 3.2. Per fare ciò l'utente può cliccare sul relativo pulsante mostrato nella figura 3.3.



(a) Giorni e ora **(b)** Posizioni **(c)** Rete

(d) Bluetooth **(e)** Batteria

Figura 3.8: Definizione condizioni personalizzabili

Per potere specificare tali condizioni, è stato definito un insieme di schermate navigabili liberamente facendo swipe verso destra e verso sinistra. In questo modo l'utente può decidere quali condizioni inserire all'interno della regola di sicurezza in maniera personalizzata. Una volta definite le condizioni di suo interesse, l'utente può quindi navigare verso l'ultima schermata e da lì salvare le informazioni relative

ai parametri inseriti.

Una volta creata la regola di sicurezza e attivata, ha quindi inizio la fase di monitoraggio. Tale fase avviene in background e serve a tenere traccia del comportamento delle app selezionate dall'utente. Nel caso in cui il sistema rilevi che un'app sta violando una regola allora avvertirà l'utente per mezzo di una notifica.

(a) Notifica violazione

(b) Azione definita "Segnala"

(c) Azione definita "Chiudi"

Figura 3.9: Rilevazione violazione regola di sicurezza

Cliccando sulla notifica verrà dunque mostrata una schermata all'utente dove viene mostrata l'app che ha violato la regola e il nome della regola stessa. A seconda del parametro di azione scelto dall'utente si avranno conseguentemente due comportamenti diversi, evidenziati dalle schermate mostrate sopra. Nel caso dell'azione di segnalazione, infatti, verrà anche mostrato un pulsante per revocare l'autorizzazione dell'app in questione, mentre nel caso dell'azione di arresto l'utente verrà solamente informato riguardo alla violazione e alla chiusura dell'app.

Nel caso in cui tra le autorizzazioni selezionate dall'utente sia presente anche quella relativa alle notifiche, si può anche osservare il comportamento del sistema nel caso in cui la violazione della regola consiste nell'invio di una notifica da parte di un'app monitorata.

(a) Oscuramento notifica

(b) Contenuto notifica oscurata

Figura 3.10: Violazione regola a causa di una notifica

Si può osservare che, in questo caso, il sistema si occuperà di bloccare la notifica inviata dall'app monitorata e mostrare al suo posto una segnalazione riguardante l'arrivo di una nuova notifica, senza mostrarne però il contenuto. Una volta cliccato sulla segnalazione verrà dunque mostrata all'utente una schermata di riepilogo riguardante la violazione, evidenziando l'app che ha violato la regola di sicurezza e mostrando il contenuto della notifica che ha inviato.

Capitolo 4

Implementazione

4.1 Architettura

L'applicazione *PrivacyManager* è stata sviluppata in ambiente Android utilizzando il linguaggio nativo Kotlin.

Nell'immagine seguente è possibile osservare la struttura generale delle componenti dell'applicazione.

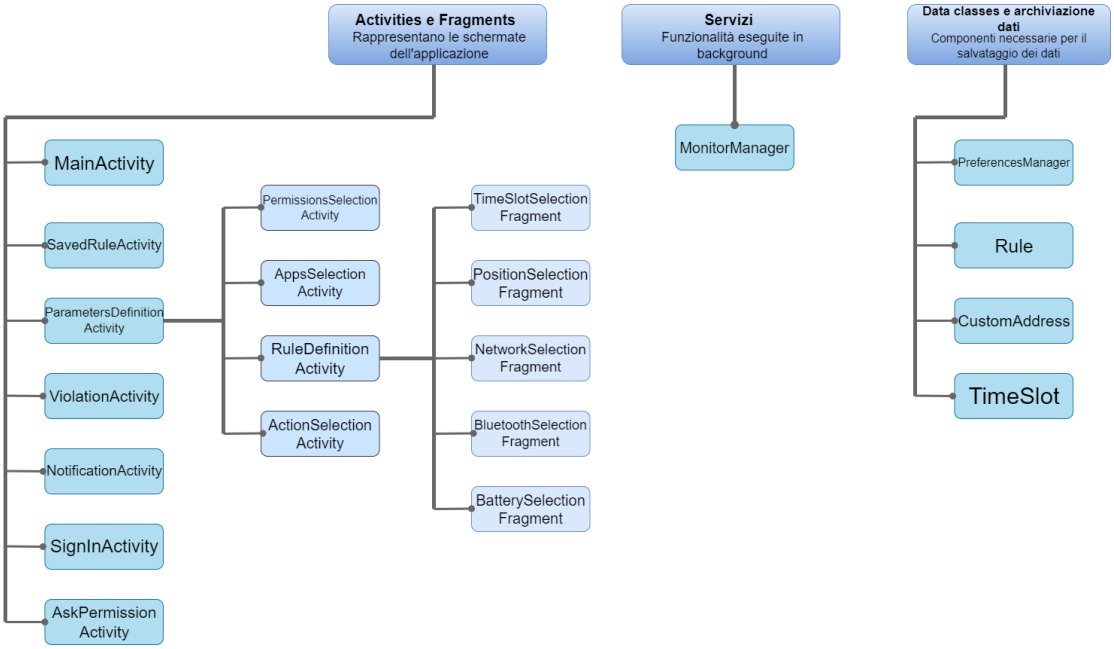


Figura 4.1: Componenti principali dell'applicazione

Le schermate principali dell'applicazione sono state costruite facendo uso di

Activities e Fragments, organizzate logicamente come illustrato sopra. Ciascuna Activity si occupa di gestire un aspetto particolare dell'applicazione:

- **MainActivity**: è mostrata all'utente all'apertura dell'applicazione. Rappresenta la homepage e mostra dunque le regole di sicurezza salvate e quelle attivate e permette di accedere alle altre Activities;
- **SavedRuleActivity**: permette di vedere i parametri definiti per la regola selezionata;
- **ParametersDefinitionActivity**: permette la creazione di una nuova regola di sicurezza. Da tale schermata è possibile navigare verso le altre Activities che permettono di definire ciascuna un parametro differente;
- **ViolationActivity**: permette di vedere un riepilogo contenente le informazioni relative all'applicazione che ha violato la regola di sicurezza;
- **NotificationActivity**: permette di vedere le informazioni relative alla notifica lanciata da una applicazione che era stata monitorata scegliendo l'autorizzazione "Notifiche";
- **SignInActivity**: permette di effettuare il login al sistema;
- **AskPermissionsActivity**: viene mostrata per richiedere all'utente la concessione delle autorizzazioni necessarie per permettere un corretto funzionamento del sistema.

Come spiegato, dall'Activity *ParametersDefinitionActivity* è possibile navigare verso le schermate seguenti:

- **PermissionsSelectionActivity**: permette di selezionare le autorizzazioni che si vogliano monitorare per mezzo della regola di sicurezza;
- **AppsSelectionActivity**: permette di selezionare le app da monitorare. L'elenco delle app mostrate cambia a seconda delle autorizzazioni selezionate nell'Activity precedente;
- **RuleDefinitionActivity**: permette di definire le condizioni opzionali per la regola. Tale Activity è composta da un insieme di Fragments navigabile liberamente dall'utente;
- **ActionSelectionActivity**: permette di scegliere l'azione che si vuole venga eseguita dalla regola di sicurezza una volta per cui il sistema riveli una violazione durante il monitoraggio.

I Fragments che compongono la *RuleDefinitionActivity* sono navigabili per mezzo di un *ViewPager2* [38], il quale permette all'utente di fare swipe tra un Fragment e l'altro e di selezionare solamente le condizioni di suo interesse:

- **TimeSlotSelectionFragment**: permette di selezionare i giorni della settimana per attivare il monitoraggio ed una fascia oraria relativa;
- **PositionSelectionFragment**: permette di inserire delle locazioni geografiche in base alle quali attivare il monitoraggio;
- **NetworkSelectionFragment**: permette di inserire il nome di una rete Wi-Fi o di selezionare la connessione dati;
- **BluetoothSelectionFragment**: mostra un elenco dei dispositivi bluetooth salvati nel dispositivo, permettendo di selezionare quelli di interesse per le condizioni di monitoraggio;
- **BatterySelectionFragment**: permette di inserire una valore percentuale di carica della batteria per influenzare il monitoraggio.

Per quanto riguarda il servizio **MonitorManager**, invece, consiste nella funzionalità di monitoraggio in background offerta dal sistema. Una volta che l'utente avvia una regola di sicurezza, allora avviene allo stesso tempo l'avvio del servizio. Tale servizio viene eseguito come un *foreground service* [39] così da permettere l'esecuzione del monitoraggio delle app del dispositivo senza che l'utente debba interagire attivamente con il sistema.

Infine, la terza componente principale del sistema è rappresentata dalle componenti che si occupano del salvataggio dei dati. La regola di sicurezza viene incapsulata in un oggetto di tipo **Rule**, identificato dal nome assegnato dall'utente e dai vari parametri selezionati per comporre la regola. L'oggetto Rule, in base alle condizioni selezionate dall'utente, potrebbe quindi avere dei campi che fanno riferimento alle posizioni e allo slot temporale, incapsulati in oggetti di tipo **CustomAddress** e **TimeSlot** rispettivamente.

A questo punto è possibile passare ad un'analisi più approfondita di alcuni aspetti appena descritti.

4.1.1 Registrazione

Prima di potere accedere all'applicazione vera e propria, è richiesto all'utente di effettuare una fase di registrazione al sistema.

Si tratta di un passaggio non strettamente necessario per il funzionamento dell'applicazione, ma è fondamentale invece per potere garantire una corretta valutazione delle capacità del sistema durante le fasi successive.

La schermata di registrazione consiste in un semplice messaggio di benvenuto in cui viene spiegato all'utente che per poter accedere al sistema è necessario effettuare questa fase. Viene dunque presentato un pulsante con cui l'utente può confermare i dati del proprio account Google memorizzato nel dispositivo e procedure dunque con l'utilizzo dell'applicazione.

4.1.2 Autorizzazioni

PrivacyManager necessita di diverse funzionalità per poter operare correttamente. Per avere accesso a tali funzionalità è necessario dunque che vengano richieste una serie ti autorizzazioni.

Alcune delle autorizzazioni richieste sono considerate di basso rischio per la sicurezza e la privacy dell'utente, e per questo motivo è sufficiente dichiararle all'interno del file *Manifest* presente alla radice del codice sorgente. Le autorizzazioni che non richiedono una particolare azione da parte dell'utente sono riportate nella tabella di seguito.

Autorizzazione	**Descrizione**
android.permission. *ACCESS_NETWORK_STATE* e *android.permission.* *ACCESS_ WIFI_STATE*	Permettono di ottenere informazioni riguardo la connessione che sta utilizzando il dispositivo. Nel caso di una rete Wi-Fi è possibile ottenere anche il suo SSID. È necessaria per fornire la condizione personalizzabile relativa alla rete
android.permission. *QUERY_ALL_PACKAGES*	Permette di ottenere informazioni sulle app installate nel dispositivo, così da poterne mostrare l'elenco completo durante la fase di definizione della regola di sicurezza
android.permission. *KILL_BACKGROUND_* *PROCESSES*	Permette di arrestare un'altra app che opera in background. È necessaria per fornire l'azione di arresto presente nella definizione della regola
android.permission. *FOREGROUND_SERVICE*	Permette la creazione e il funzionamento del servizio di monitoraggio in background. Una volta creato, viene mostrata all'utente una relativa notifica a bassa priorità

Tabella 4.1: Autorizzazioni a basso rischio richieste da *PrivacyManager*

Sono inoltre presenti altre autorizzazioni necessarie alla corretta operatività di *PrivacyManager*, le quali costituiscono, secondo la classificazione proposta da Android, un rischio maggiore per la sicurezza e la privacy dell'utente. Per

questo motivo devono essere concesse a runtime. È possibile consultarle nell'elenco seguente.

Autorizzazione	Descrizione
android.permission. *ACCESS_FINE_LOCATION* e *android.permission.* *ACCESS_COARSE_LOCATION*	Permettono di accedere alla localizzazione. Sono necessarie sia per fornire la condizione personalizzabile corrispondente che per verificare se un'altra app sta accedendo alla funzionalità
android.permission. *READ_CALENDAR* e *android.permission.* *WRITE_CALENDAR*	Necessarie per monitorare se altre app modificano degli eventi del calendario. Pur non scrivendo mai degli eventi, *PrivacyManager* necessita dell'autorizzazione di scrittura per verificare il comportamento durante il monitoraggio
android.permission. *BLUETOOTH_SCAN*, *android.permission.* *BLUETOOTH_ADVERTISE* e *android.permission.* *BLUETOOTH_CONNECT*	Permettono di ottenere informazioni riguardo i dispositivi bluetooth salvati e attualmente connessi. Sono necessarie per fornire la relativa condizione personalizzabile
android.permission. *POST_NOTIFICATIONS*	Permette l'invio di notifiche all'utente. Prima di *Android 13* (2022) non era necessario richiedere esplicitamente l'autorizzazione all'utente

Tabella 4.2: Autorizzazioni a runtime richieste da *PrivacyManager*

Infine, l'ultimo gruppo di autorizzazioni richieste da *PrivacyManager* riguarda le funzionalità essenziali per fornire in maniera adeguata il monitoraggio delle applicazioni dell'utente. Si tratta di autorizzazioni di tipo speciale, e per questo motivo l'utente deve concederle esplicitamente acconsentendo al loro utilizzo dalle impostazioni di sistema. Le autorizzazioni in questione sono quelle mostrate nella tabella di seguito.

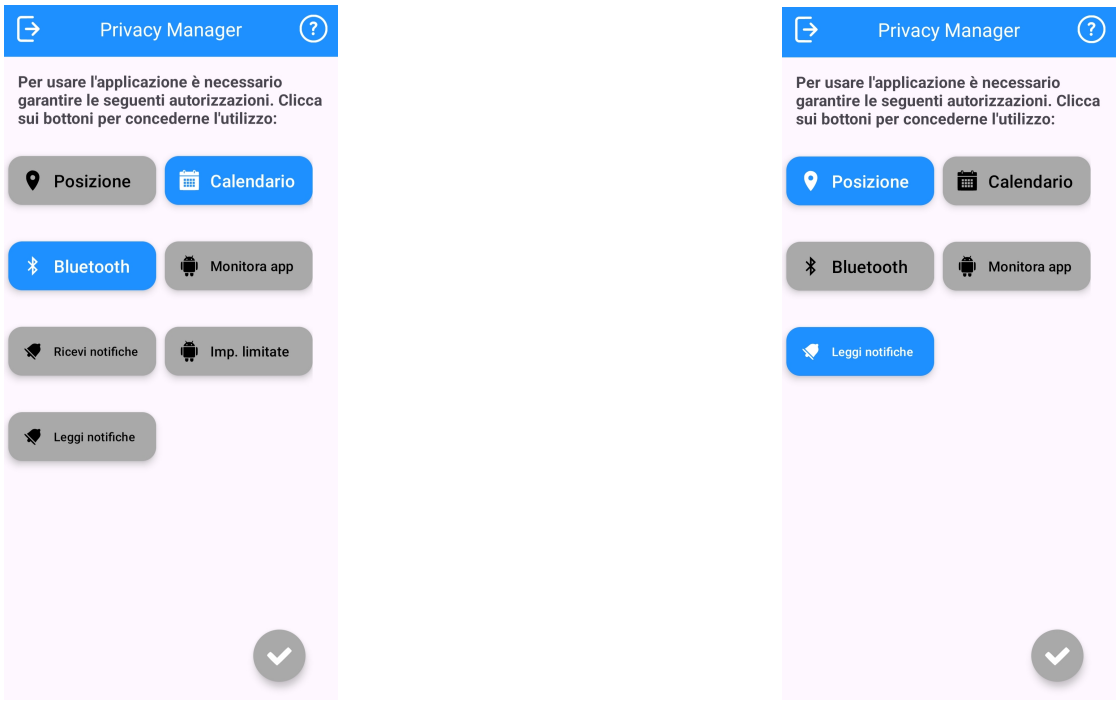
Autorizzazione	Descrizione
android.permission. *PACKAGE_USAGE_STATS*	Permette di ottenere delle statistiche di utilizzo e delle informazioni riguardanti le app in esecuzione sul dispositivo. In questo modo è possibile ottenere un elenco delle app che l'utente sta utilizzando (o che sono in background) e a quali funzionalità del dispositivo sta accedendo
android.permission. *BIND_NOTIFICATION* *_LISTENER_SERVICE*	Permette di monitorare ed operare sulle notifiche lanciate da altre applicazioni. È possibile in questo modo intercettare una notifica lanciata, bloccarla, salvarne il contenuto e mostrarlo in seguito all'utente

Tabella 4.3: Autorizzazioni speciali richieste da *PrivacyManager*

Riguardo l'ultima autorizzazione descritta, relativa al monitoraggio delle notifiche, a partire dalla versione 13 di Android (2022) ne è stato ulteriormente rafforzato il livello di sicurezza. Per poterla garantire, infatti, è richiesto all'utente innanzitutto di modificare la lista di applicazioni che possono accedere alle "Impostazioni Limitate" [40], e solo successivamente è possibile concederne effettivamente l'autorizzazione navigando nella relativa sezione delle impostazioni di sistema. Con l'inasprimento delle misure di sicurezza relativa alla funzionalità, dunque, si vuole limitare il rischio di sicurezza e privacy che la concessione di questa autorizzazione comporta se garantita ad un'applicazione malevola.

Per quanta riguarda *PrivacyManager*, per venire incontro alle difficoltà che possono derivare dalla navigazione nelle impostazioni di sistema per potere garantire l'accesso alle funzionalità sopra descritte, si è deciso di mostrare all'utente una schermata relativa alle autorizzazioni di runtime e a quelle speciali che viene mostrata subito dopo aver completato la fase di descrizione.

Come descritto nelle tabelle precedenti, è possibile notare che a partire dalla versione 13 di Android le misure di sicurezza adottate dal sistema operativo sono cambiate, diventando più dure in determinati casi. È importante però tenere in considerazione che, essendo tale versione di Android stata rilasciata da relativamente poco tempo (metà 2022), è largamente diffusa solamente nei dispositivi di più recente generazione. Per questo motivo, si è deciso di differenziare la schermata relativa alle autorizzazioni tenendo conto di questa differenza, come è possibile vedere nelle immagini seguenti.



(a) Android 13 e successive **(b)** Versioni precedenti

Figura 4.2: Schermata di concessione autorizzazioni *PrivacyManager*

Com'è possibile osservare, i pulsanti presenti nella schermata permettono all'utente di concedere l'autorizzazione corrispondente. Nel caso di autorizzazioni di tipo speciale ("Monitora app", "Impostazioni limitate" e "Leggi notifiche"), cliccando sul relativo pulsante viene aperta la corrispondente pagina delle impostazioni di sistema del dispositivo per poterla garantire. In questo modo si evita la possibile difficoltà per l'utente nel navigare autonomamente le impostazioni del proprio dispositivo.

4.1.3 Homepage

Una volta che l'utente abbia garantito tutte le autorizzazioni necessarie, allora può accedere alla homepage del sistema. Questa rappresenta la schermata principale dell'applicazione ed è quella che viene mostrata all'utente ogni volta che accede al sistema.

Da qui è possibile ottenere le informazioni relative a tutte le regole di sicurezza che sono state create e vedere tutte quelle che sono attive al momento per la fase di monitoraggio. Tali informazioni sono visibili in due sezioni separate presenti nella

medesima schermata. Inoltre, è da qui che è possibile procedere all'attivazione di una regola di sicurezza, alla sua modifica oppure alla sua eliminazione.

Da questa schermata, infine, è possibile la navigazione verso tutte le altre parti dell'applicazione.

4.1.4 Creazione di una regola

Da tale schermata è possibile procedere con la creazione della regola di sicurezza, definendone tutti i parametri fondamentali.

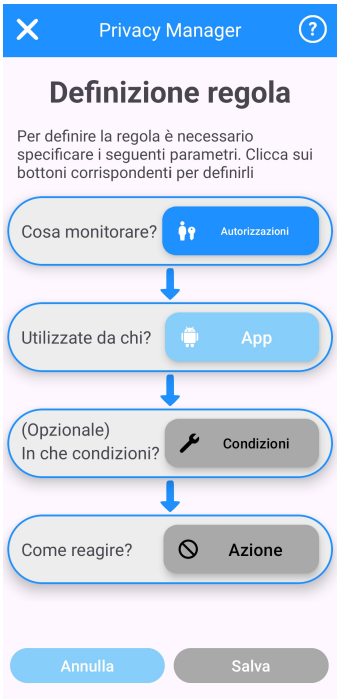


Figura 4.3: Creazione di una regola di sicurezza

Cliccando su ciascun pulsante è infatti possibile specificare il parametro corrispondente. Come spiegato in precedenza, il parametro relativo alle condizioni personalizzabili è opzionale e dunque l'utente può procedere con il salvataggio della regola anche nel caso in cui non l'abbia specificato.

Una volta che l'utente abbia specificato i parametri relativi alle autorizzazioni, alle app e all'azione, allora sarà possibile cliccare sul pulsante di salvataggio. Verrà quindi richiesto all'utente l'inserimento di un nome per identificare la regola e procedere dunque al suo salvataggio.

4.2 Flusso di utilizzo

Per gestire lo spostamento tra le varie schermate dell'applicazione ci si è basati principalmente sull'utilizzo degli *Intent* [41], i quali permettono di iniziare una nuova attività a partire dalla precedente e di scambiare delle informazioni tra le due Activities.

L'esecuzione del processo di monitoraggio, invece, avviene per mezzo di un *Foreground Service* [39], il quale è utile per effettuare delle operazioni in maniera indipendente dall'interazione dell'utente

4.2.1 Navigazione in homepage

La homepage, come spiegato, rappresenta la schermata principale dell'applicazione, ed è quella che viene mostrata ogni qual volta l'utente accede al sistema.

Figura 4.4: Homepage di *PrivayManager*

Oltre a mostrare informazioni circa le regole di sicurezza salvate e quelle attivate per la fase di monitoraggio, permette anche la navigazione verso le ulteriori componenti del sistema.

Cliccando infatti sull'icona dell'ingranaggio in alto a sinistra, è possibile ritornare alla schermata di concessione delle autorizzazioni, mostrata in figura 4.2, per rivedere un riepilogo delle funzionalità richieste dal sistema.

Cliccando invece sull'icona in basso a destra, è possibile navigare verso la schermata di creazione di una regola di sicurezza, mostrata in figura 4.3. In questo caso verrà creato un intent che presenta solamente il nome dell'attività da lanciare, senza nessun altro parametro.

Nel caso in cui l'utente voglia invece modificare i parametri della regola di sicurezza, allora potrà cliccare sull'icona a forma di matita presente accanto al nome della regola. In questo caso, prima di far partire la nuova attività, vengono recuperate le informazioni relative ai parametri che definiscono la regola, e incapsulate in un oggetto JSON. Tale oggetto è dunque messo ad accompagnamento dell'intent, in maniera tale da conservare le informazioni relative alla regola che si ha intenzione di modificare e mostrarle all'utente nella nuova schermata.

Se l'utente voglia procedere invece alla cancellazione della regola salvata, cliccando sull'icona relativa verrà mostrato un pop-up di conferma per procedere all'eliminazione. Verrà quindi richiamata la componente *PreferencesManager* che si occuperà di gestire la cancellazione della regola.

Sempre da tale schermata, è anche possibile procedere all'attivazione delle regole di sicurezza. Cliccando sulla relativa icona, la regola corrispondente verrà inserita nell'elenco di quelle attive e verrà dunque mostrata nella sezione corrispondente.

Infine, cliccando sul nome della regola di sicurezza (sia che sia presente nell'elenco di quelle attive che viceversa), è possibile passare alla schermata riepilogativa riguardante i parametri definiti per essa.

4.2.2 Creazione di una regola

Dalla homepage del sistema è possibile navigare verso la schermata di creazione della regola di sicurezza.

Il parametro di base per la creazione della regola è rappresentato dalle autorizzazioni. Solamente dopo aver selezionato e salvato le autorizzazioni che si ha intenzione di monitorare è possibile scegliere il parametro successivo, relativo alle applicazioni. Una volta che si sono scelta anche queste, allora verranno resi cliccabili sia il pulsante relativo alle condizioni che quello relativo all'azione. Per potere procedere al salvataggio della regola è comunque sufficiente specificare solamente l'azione che si vuole aggiungere.

Nel caso in cui l'utente voglia invece specificare delle condizioni personalizzabili, allora verrà mostrata un'ulteriore attività che si occupa di gestire i 5 Fragments relativi a ciascuna condizione. I Fragments sono contenuti all'interno di un *ViewPager2* [38], così da permettere all'utente la navigazione tra un Fragment e l'altro per mezzo di swipe a destra/sinistra (o con i relativi pulsanti presenti nella parte

Figura 4.5: Parametri definiti per una regola salvata

bassa della schermata). Per gestire in maniera coerente i dati immessi dall'utente nei vari Fragment, è stata creata un'interfaccia chiamata **ParameterListener**, che definisce un unico metodo *onParameterEntered*. Tale metodo riceve un parametro *parameter* di tipo "String" e un parametro *data* di tipo "Any?". In questo modo l'Activity contenente tutti i Fragments si occuperà di gestire il salvataggio dei parametri mano a mano che l'utente li definisce nei Fragments corrispondenti.

Tale schermata è utilizzata anche per effettuare la modifica dei parametri di una regola definita in precedenza. Una volta che l'attività viene fatta partire, infatti, verrà effettuato un controllo sull'intent ricevuto, per verificare la presenza di parametri opzionali. Nel caso in cui non sia presente nessun parametro, allora non verrà effettuata alcuna operazione aggiuntiva, e l'utente potrà quindi procedere nella definizione della regola di sicurezza. Viceversa, nel caso in cui sia presente un parametro che specifica un oggetto di tipo Rule già salvato in precedenza, cliccando su ciascun pulsante, è possibile vedere i parametri che erano stati salvati, ed eventualmente modificarli. L'unica cosa da tenere in considerazione in questo caso è che, pur essendo possibile modificare anche le autorizzazioni che vengono monitorate dalla regola, se ciò avviene allora tutti gli altri parametri verranno resettati e sarà necessario reinserirli. Questo è un passaggio fondamentale per non creare incoerenze tra ciò che è già stato salvato in precedenza e ciò che si salverà con

la modifica. Infatti, le applicazioni mostrate nella relativa schermata e l'azione che è possibile scegliere dipendono da quali sono le autorizzazioni di base della regola, ed è per questo che è necessario resettarle nel caso in cui avvenga un cambiamento di questo tipo. Nel caso in cui l'utente stia effettuando un'operazione di modifica allora al momento del salvataggio non verrà più richiesto l'inserimento di un nome, in quanto già presente nei parametri ricevuti dall'intent.

4.2.3 Attivazione del monitoraggio

Come detto, l'attivazione di una regola di sicurezza avviene cliccando sulla relativa icona nella homepage del sistema. L'utente ha la possibilità di attivare più regole di sicurezza contemporaneamente. In questo caso, l'elenco delle regole attive è visionabile nella relativa sezione della schermata di homepage.

La componente che si occupa di gestire il monitoraggio del dispositivo in base alle regola attivate è il *MonitorManager*. Come spiegato, esso opera come un foreground service che rimane attivo in background ed effettua ciclicamente le sue operazioni di controllo.

Nel caso in cui non era presente nessun altra regola nell'elenco di quelle attive, allora all'attivazione della prima verrà contemporaneamente creato e avviato il servizio, in seguito alla sua inizializzazione. Essendo, infatti, il servizio un foreground service, allora è richiesta la presenza di una notifica a bassa priorità da mostrare all'utente. È vero infatti che il servizio opera in maniera tale che l'utente non abbia bisogna di interagire in prima persona, risulta però importante informarlo riguarda la presenza dell'applicazione che opera in background. Dopo aver creato la notifica a bassa priorità, è necessario inoltre procedere alla creazione di due canali di notifica. Il primo verrà utilizzato per l'invio di notifiche relative alla violazione di una regola di sicurezza, mentre il secondo verrà invece usato per provvedere all'oscuramento delle notifiche proveniente da altre applicazioni.

Il funzionamento del *MonitorManager* è basato su un *handler* [42] che si occupa di eseguire periodicamente un controllo sulle app basandosi sull'elenco di regole di sicurezza attualmente attive. Considerato tale elenco, da questo ne viene creato un successivo più ristretto, in cui vengono considerate solamente le regole per cui sono verificate tutte le condizioni personalizzabili definite dall'utente (slot temporale, posizioni geografiche, connessione internet, dispositivi bluetooth connessi, livello di carica del dispositivo).

Contemporaneamente, viene considerata la lista delle applicazioni che sono attualmente in esecuzione. Per costruire questa lista, e per ottenere delle ulteriori informazioni sulle applicazioni in questione, si è utilizzato il componente di Android *UsageStatsManager* [43]. Tramite questo componente è possibile ottenere informazioni riguardanti le applicazione che l'utente ha eseguito in un dato intervallo di tempo e alcune statistiche riguardo l'utilizzo.

In base all'intersezione tra la lista delle applicazioni da monitorare e quelle utilizzate dall'utente in base alle statistiche dell'UsageStatsManager, si può dunque monitorare il comportamento di queste applicazioni relativamente alle autorizzazioni definite nella regola di sicurezza.

4.2.4 Segnalazione di una violazione

Una volta avviato il servizio di monitoraggio tramite il *MonitorManager*, sarà tale componente stesso a segnalare la violazione di una regola all'utente per mezzo di una notifica.

Il processo di segnalazione viene attivato nel momento in cui una delle applicazioni che sta utilizzando l'utente, che era stata definita all'interno della regola di sicurezza, sta accedendo ad una funzionalità definita da un'autorizzazione monitorata dalla regola stessa (al verificarsi delle condizioni personalizzabili specificate dall'utente).

Di seguito è riportato un elenco delle componenti che si occupano di monitorare e segnalare il comportamento delle app in base alle diverse autorizzazioni.

Elemento	**Tipo**	**Componente Android**	**Descrizione**
MonitorLocation	Funzione	PackageManager [44]	Verifica se all'app sia garantita l'autorizzazione relativa alla localizzazione
MonitorCalendar	HandlerThread	ContentObserver [45]	Verifica se l'app stia eseguendo un'azione (creazione, modifica, cancellazione) su un evento del calendario
MonitorCamera	Funzione	CameraManager [46]	Verifica la disponibilità della fotocamera e, nel caso in cui non sia disponibile, segnala l'app che la sta usando

Elemento	Tipo	Componente Android	Descrizione
Notification Listener	Servizio	Notification ListenerService [47]	Quando viene intercettata una notifica, questa viene bloccata e il contenuto è salvato per mostrarlo sucessivamente all'oscuramento

Tabella 4.4: Componenti per il monitoraggio e la segnalazione

Nel momento in cui venga intercettata una violazione, allora viene eseguita l'azione definita nella regola di sicurezza. In particolare, nel caso in cui l'azione selezionata riguardava l'arresto dell'app, allora viene richiamata la funzione *killBackgroundProcesses* dell'*ActivityManager* [48]. In ogni caso, verrà poi lanciata una notifica con lo scopo di informare l'utente dell'avvenuta violazione. Nel momento in cui l'utente clicchi sulla notifica, verrà dunque aperta una schermata di riepilogo in cui è possibile vedere l'applicazione che violato la regola di sicurezza e il nome della regola che ha violato.

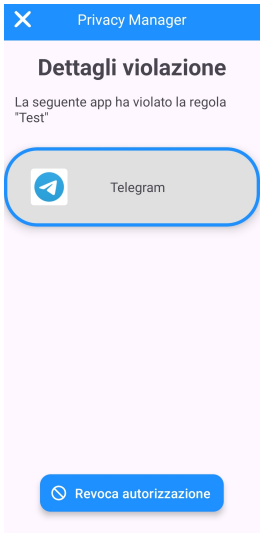
Un processo similare avviene nel caso in cui la violazione consista nella ricezione di una notifica da oscurare. In questo caso, infatti, il *NotificationListener* si occuperà del blocco della notifica originale e di lanciare una notifica oscurata con lo scopo di informare l'utente. Nel momento in cui l'utente clicchi su questa notifica verrà quindi mostrata una schermata in cui è possibile vedere l'applicazione che ha violato la regola di sicurezza, il nome della regola che violato e il contenuto della notifica oscurata.

4.3 Prima apertura

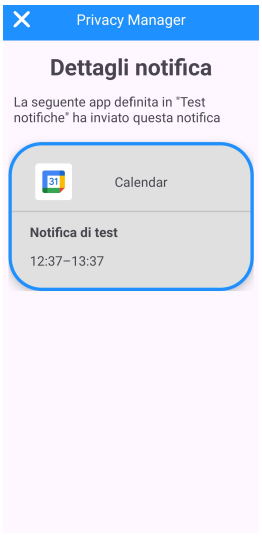
Prima di permettere all'utente un utilizzo completo del sistema, è necessario effettuare dei passaggi preliminari che verranno illustrati nel dettaglio di seguito. Alcuni di questi passaggi sono necessari per garantire un corretto funzionamento delle diverse parti del sistema, mentre altre hanno invece lo scopo di istruire l'utente e guidarlo all'interno dell'applicazione.

4.3.1 Registrazione

Nel primo momento in cui l'utente acceda all'applicazione, verrà richiesto di registrarsi utilizzando il proprio account Google.



(a) Violazione regola



(b) Oscuramento notifica

Figura 4.6: Riepilogo violazione regola di sicurezza

Pur non essendo una funzionalità strettamente necessaria per il corretto funzionamento dell'applicazione si è deciso di inserire questa fase allo scopo di valutare, in una fase successiva, l'usabilità del sistema.

La scelta di permettere il login utilizzando il proprio account Google, invece, deriva dalla ricerca di una semplicità di utilizzo per l'utente finale che testerà il sistema, che si accompagna bene all'architettura del sistema Android. Infatti, a ciascun utente del sistema operativo è richiesto di avere configurato nel proprio dispositivo un account Google, necessario per utilizzare i servizi messi a disposizione da Google, appunto, e per poter scaricare le applicazioni dal Google Play Store.

Una volta che l'utente abbia confermato l'accesso al proprio account Google, dunque, ha la possibilità di proseguire nell'utilizzo del sistema *PrivacyManager*.

4.3.2 Autorizzazioni

Subito dopo aver completato con successo la fase di registrazione, allora viene mostrata all'utente una schermata in cui vengono presentati le diverse autorizzazioni richieste dal sistema.

Non è possibile proseguire nell'utilizzo del sistema se l'utente non garantisce l'accesso per ciascuna funzionalità. Infatti, il comportamento messo in atto da *PrivacyManager* necessita di diverse componenti che sono messe a disposizione dal sistema operativo Android ma, rappresentando un fattore di rischio abbastanza

elevato per la sicurezza e la privacy dell'utente, devono essere esplicitamente concesse da questo. È inoltre necessario che vengano spiegate chiaramente le ragioni per ciascuna autorizzazione richiesta, così da guidare l'utente in questo processo.

In particolare, *PrivacyManager* necessita di alcune autorizzazioni che sono considerate sensibili per i dati degli utenti, come illustrato nella sezione 4.1.2. Dunque, per proteggere adeguatamente gli utenti meno esperti dai possibili rischi derivanti, il sistema Android richiede la navigazione all'interno di diverse sezioni delle impostazioni di sistema del dispositivo. Questa scelta del sistema Android, quindi, da un lato è comprensibile vista la sensibilità delle funzionalità che ne derivano, ma dall'altro richiede uno sforzo notevole da parte degli utenti per poterne garantire l'accesso per le applicazioni che ne necessitano. Per venire incontro a questa situazione, si è deciso quindi di implementare i pulsanti relativi alle autorizzazioni di questo tipo di un relativo intent che possa permettere l'apertura dell'impostazione corretta senza che l'utente debba navigare autonomamente nei menu del proprio dispositivo. In particolare, nella tabella seguente è possibile vedere gli intent di questo tipo che sono stati utilizzati.

Intent	**Autorizzazione associata**
Settings.ACTION_MANAGE_APPLICATIONS_SETTINGS	Impostazioni limitate
android.settings.ACTION_NOTIFICATION_LISTENER_SETTINGS	android.permission.BIND_NOTIFICATION_LISTENER_SERVICE
Settings.ACTION_USAGE_ACCESS_SETTINGS	android.permission.PACKAGE_USAGE_STATS

Tabella 4.5: Intent per autorizzazioni sensibili

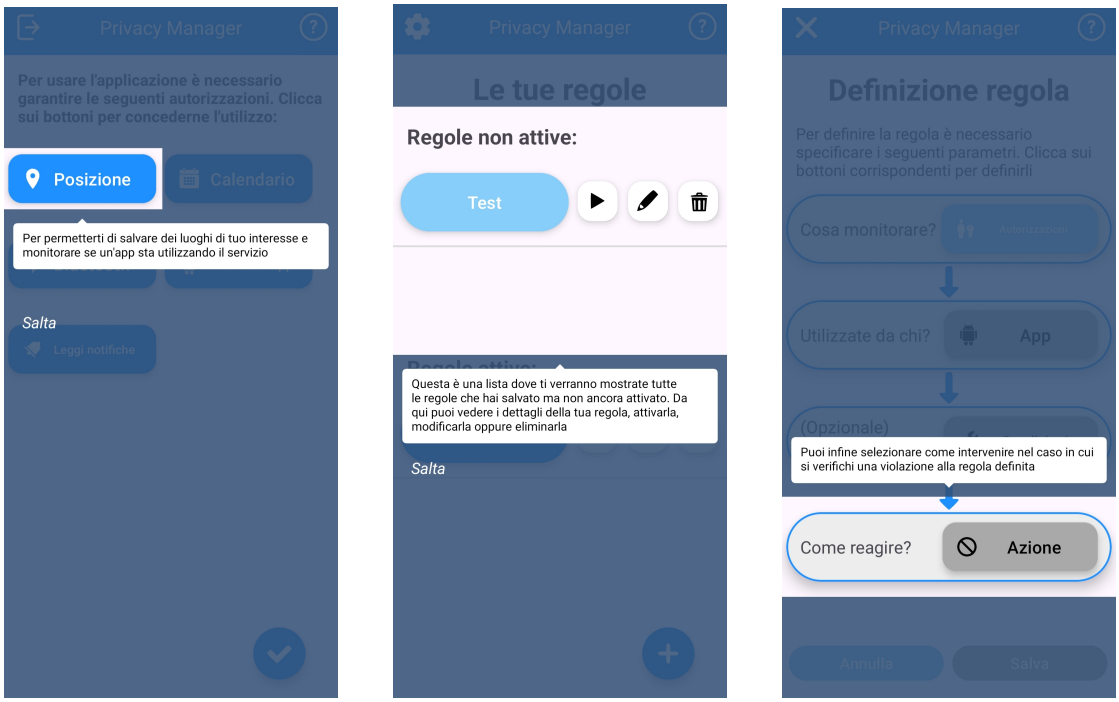
4.3.3 Tutorial

Considerata la relativa complessità del sistema, per evitare di lasciare disorientato l'utente e permettergli di comprendere chiaramente le diverse parti dell'applicazione, sono stati implementati alcuni tutorial. Per la realizzazione dei tutorial si è fatto utilizzo della libreria *MaterialShowcaseView* [1].

Si è deciso di inserire tre diversi tutorial all'interno dell'applicazione, corrispondenti alle 3 schermate principali del sistema. In ciascun tutorial vengono spiegate nel dettaglio le varie sezioni della relativa schermata e il modo in cui l'utente può sfruttare il comportamento del sistema. In particolare, le schermate prese

[1] https://github.com/deano2390/MaterialShowcaseView

in considerazione sono quelle relative alle autorizzazioni, alla homepage e alla schermata di definizione di una nuova regola di sicurezza.



(a) Autorizzazioni **(b)** Homepage **(c)** Creazione regola

Figura 4.7: Esempi di tutorial di *PrivacyManager*

I diversi tutorial vengono mostrati all'utente nel momento in cui effettui il primo accesso al sistema, una volta completata la fase di registrazione, ma sono comunque visionabili ancora cliccando sull'icona in alto a destra presente in ciascuna schermata.

4.4 Dati e statistiche di utilizzo

Verrà adesso analizzata la parte dell'applicazione relativa al salvataggio dei dati. Il componente principale che si occupa di ciò è rappresentato dal *PreferencesManager* per quanto riguarda il salvataggio in locale. Oltre a questa, è anche presenta una parte di raccolta dati che sono invece salvati in cloud, sfruttando la piattaforma Firebase.

4.4.1 Salvataggio dei dati

Per quanto riguarda la permanenza delle regole di sicurezza create dall'utente si è scelto di sfruttare un salvataggio in locale per i dati di questo tipo.

Una volta che l'utente crea una nuova regola di sicurezza, al momento del salvataggio le proprietà che la definiscono vengono incapsulate all'interno di un oggetto di classe *Rule*. L'oggetto presenta la struttura riassunta nella tabella seguente.

Proprietà	**Tipo**	**Descrizione**
name	String	Nome inserito dall'utente per identificare la regola di sicurezza
permissions	List<String>	Lista delle autorizzazioni monitorate dalla regola di sicurezza
apps	List<String>	Liste dei nomi delle applicazioni monitorate dalla regola di sicurezza
packageNames	List<String>	Lista dei nomi dei pacchetti associati alle applicazioni monitorate
timeSlot	TimeSlot?	Slot di giorni e ora inseriti come condizione opzionale
positions	List<CustomAddress>?	Lista di posizioni geografiche inserite come condizione opzionale
networks	List<String>?	Lista di reti internet inserite come condizione opzionale
bt	List<String>?	Lista di dispositivi bluetooth inseriti come condzione opzionale
battery	Int?	Valore percentuale di batteria inserito come condizione opzionale
action	String	Azione definita dalla regola di sicurezza, da effettuare al momento della rilevazione di una violazione
active	Boolean	Flag per indicare se la regola di sicurezza è al momento attiva o no

Tabella 4.6: Proprietà della regola di sicurezza

Com'è possibile osservare, l'oggetto presenta due proprietà il cui tipo è definito da una data class personalizzata, utilizzata per semplificare il processo di salvataggio dei relativi parametri. Le classi in questione sono le seguenti:

- *TimeSlot*: presenta un campo relativo alla lista dei giorni ("List<String") e uno relativo all'orario di inizio e fine associato ("Pair(String, String)");
- *CustomAddress*: presenta un campo relativo all'indirizzo inserito dall'utente ("String"), uno riferito alla latitudine e uno riferito alla longitudine (entrambi "Double").

Una volta creato l'oggetto *Rule* relativo alla regola di sicurezza, questo viene serializzato in un corrispettivo oggetto JSON. In questo modo è possibile salvare tale oggetto serializzato all'interno delle *SharedPreferences* [49].

La scelta di questo meccanismo di salvataggio deriva dal fatto che, per la natura dei dati da salvare, una struttura relazionale per la loro rappresentazione sarebbe stata poco adatta. Per questo motivo si è voluto utilizzare un approccio non strutturato, rappresentando le regole di sicurezza come oggetti JSON identificati dal proprio nome. Vista anche la non necessità di effettuare query articolate per il recupero dei dati e la mancanza di concorrenza considerando la natura dell'applicazione, si è voluto trovare un metodo semplice e diretto per il salvataggio dei dati, ma che fosse al contempo altrettanto efficace. Inoltre, la rappresentazione tramite oggetti JSON serializzati delle regole di sicurezza permette anche un semplice flusso delle informazione tra una Activity e l'altra, comunicando per mezzo degli intent.

4.4.2 Statistiche raccolte

In accompagnamento alla funzionalità di registrazione discussa nelle sezioni precedenti, si è deciso raccogliere dei dati relativi all'uso fatto dagli utenti che utilizzano il sistema. Lo scopo di ciò consiste nel verificare l'efficacia dell'applicazione e valutare il modo in cui viene usata. Tali statistiche raccolte non vengono salvata in locale, come accade invece per le regole di sicurezza. Al contrario, in questo caso, tutti i dati vengono memorizzati facendo uso del database in cloud Firestore.

Alcuni dei dati raccolti non sono necessari al processo di verifica di efficacia del sistema, ma sono stati inseriti per poter valutare quando può interrompersi il processo di testing per il relativo utente, come verrà discusso in maggiore dettaglio in seguito. I dati di questo tipo sono l'*email* dell'utente e il *timestamp* relativo alla sua registrazione al sistema.

I dati considerati per il processo di valutazione, invece, sono i seguenti:

- *regola di sicurezza*: tutte le proprietà della regola sono memorizzate con lo scopo di valutare quali sono le autorizzazioni e le app di maggiore interesse

e le condizioni opzionali che vengono scelte più spesso. Al contempo è utile capire l'intensità con cui l'utente ha utilizzato l'applicazione osservando il numero delle regole che ha creato;

- *numero di attivazioni* e *tempo di attivazione*: proprietà relative ad una singola regola di sicurezza. Permettono di valutare l'efficacia del sistema per l'utente, osservando quante volte ha attivato una determinata regola e per quanto tempo l'ha tenuta attiva;

- *violazioni rilevate* e *timestamp violazione*: proprietà relative ad una singola regola di sicurezza. Lo scopo di questa statistica consiste nel valutare l'utilità e l'efficacia del sistema nella protezione dell'utente.

4.4.3 Firebase

L'infrastruttura utilizzata per il processo di registrazione al sistema e per il salvataggio in cloud delle statistiche di utilizzo è rappresentata da *Firebase* [2].

Si tratta di una componente che non è strettamente necessaria per il corretto funzionamento del sistema, ma è stato comunque importante introdurla per potere intraprendere la fase di valutazione da parte degli utenti.

Firebase è una piattaforma cloud messa a disposizione da Google che offre diverse componenti per curare aspetti differenti di un'applicazione Android. Nel caso di *PrivacyManager*, le componenti di cui si è fatto utilizzo sono *Firebase Authentication* [50] e *Firestore* [51]. La prima ha permesso di gestire la fase di registrazione al sistema, integrando il processo per mezzo dell'account Google degli utenti. Il secondo, invece, ha permesso il salvataggio in cloud dei dati e delle statistiche prodotte dagli utenti durante l'utilizzo del sistema.

Al momento dell'autenticazione, Firebase genererà per ciascun utente un codice identificativo, per permettere il salvataggio dei dati relativi. Tale riferimento verrà restituito alla fine del processo di login, incapsulato all'interno di un oggetto del tipo *FirebaseUser*. Il riferimento verrà dunque memorizzato localmente nelle SharedPreferences e recuperato ogni qual volta ci sia la necessità di salvare o aggiornare dei dati nel cloud.

La struttura noSQL su cui si basano i documenti di Firestore si accompagna bene all'incapsulamento delle regole di sicurezza in oggetti JSON per il salvataggio in locale. Dunque, al momento della creazione della regola, basta far sì che lo stesso oggetto venga salvato sia localmente che in cloud, aggiungendo poi a quest'ultimo i campi necessari per la fase di raccolta delle statistiche. In questo momento verrà

[2] https://firebase.google.com/

dunque utilizzato il riferimento dell'utente salvato localmente durante la fase di autenticazione per permettere di aggiornare il corretto documento Firestore.

Capitolo 5

Valutazione

L'applicazione è stata testata *in-the-wild*. Gli utenti del sistema testano dunque l'app in condizioni reali, che corrispondono alla loro quotidianità e alle loro abitudini, senza nessun controllo diretto delle loro azioni. Il modello utilizzato per il test è del tipo *within-subject*. In questo modo, ciascun partecipante ha a disposizione tutte le funzionalità offerte dall'applicazione. I partecipanti a questa fase di test sono persone appartenenti alla mia sfera personale, reclutate tramite messaggi privati per spiegare le diverse fasi dell'esperimento.

Durante la fase di test i partecipanti hanno usato l'applicazione *PrivacyManager* in condizioni reali, potendo accedere a tutte le funzionalità messe a disposizione dal sistema.

5.1 Struttura del test

Il test ha avuto una durata di una settimana, ed è stato suddiviso in:

- questionario iniziale, visionabile all'appendice A.1;
- utilizzo dell'applicazione per una settimana;
- questionario finale, visionabile all'appendice A.2.

La durata del test è stata scelta per permettere agli utenti la definizione di diverse regole di sicurezza e una loro applicazione in condizioni reali, raccogliendo contestualmente delle statistiche riguardo l'utilizzo dell'applicazione, come discusso nella relativa sezione.

Il questionario iniziale è utilizzato sia per richiedere il consenso al trattamento dei dati che per fornire delle informazioni preliminari sull'applicazione. Le domande che lo compongono sono incentrate principalmente sulle percezioni dell'utente riguardo l'utilizzo del proprio dispositivo e la protezione della propria privacy, oltre

che su una raccolta delle informazioni demografiche dell'utente. Al termine del questionario è stato richiesto all'utente di scaricare l'applicazione tramite un link che consentiva l'accesso al file apk.

Durante il periodo di utilizzo, l'applicazione collezionava una serie di dati e statistiche riguardo l'utilizzo della stessa da parte dell'utente. Le statistiche vengono memorizzate nel database Firestore. In tal modo ci si è posti l'obiettivo di verificare l'efficacia dell'applicazione e il modo in cui è stata utilizzata. Il dettaglio sui dati raccolti e sulla loro archiviazione è stato discusso nella relativa sezione 4.4.2.

Una volta completato il periodo di utilizzo, gli utenti sono dunque stati invitati alla compilazione del questionario finale, costituito da:

- questionario System Usability Scale (SUS) [52], per lo studio dell'usabilità dell'applicazione;

- domande chiuse in cui viene richiesto all'utente di condividere la propria opinione sull'app.

Le domande chiuse sono state utilizzate per valutare l'efficacia dei tutorial iniziali presenti all'interno del sistema e per chiedere più nel dettaglio se ci fossero state parti dell'applicazione poco chiare o che hanno funzionato in maniera errata.

5.2 Risultati

5.2.1 Percezioni iniziali

I partecipanti reclutati inizialmente erano 10. Di questi, 8 hanno completato tutte le fasi del test. Soltanto i dati di questi ultimi verranno quindi considerati per l'analisi dei risultati.

Degli 8 partecipanti 5 sono donne e 3 sono uomini. L'età varia tra i 20 e i 30 anni, con una media di circa 24 anni.

Per quanto riguarda le percezioni degli utenti riguardo l'efficacia dei meccanismi esistenti per la protezione della propria privacy, l'opinione generale è medio-bassa com'è possibile vedere in Figura 5.1.

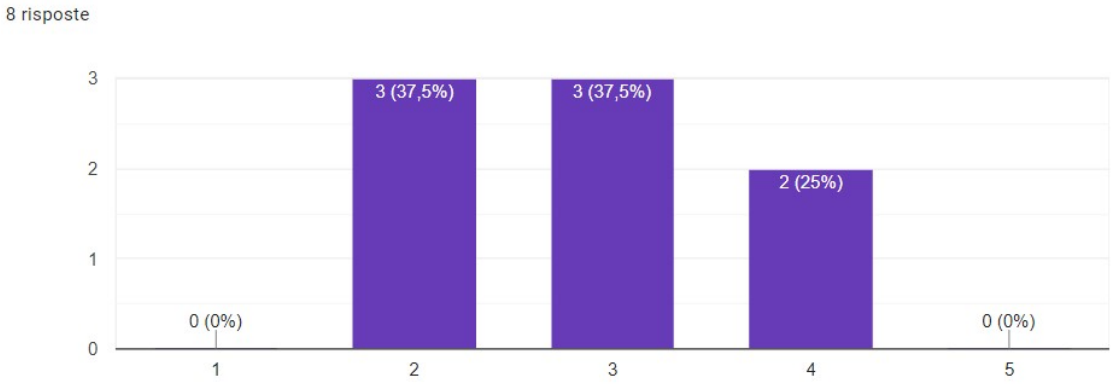


Figura 5.1: Percezione degli utenti sull'efficacia dei meccanismi di protezione esistenti. Scala Likert da "per nulla" (1) a "moltissimo" (5)

Considerando invece la semplicità di utilizzo delle soluzioni messe a disposizione dal dispositivo per la protezione della propria sicurezza e della propria privacy, l'opinione risulta essere ancora più bassa, visto che è possibile osservare (in Figura 5.2) anche come due partecipanti abbiano assegnato il punteggio minimo nella scala Likert.

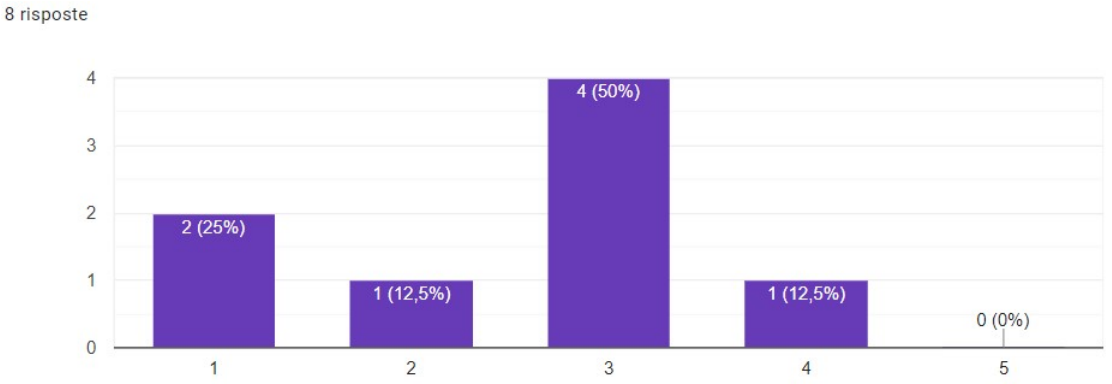


Figura 5.2: Percezione degli utenti sulla semplicità dei meccanismi di protezione esistenti. Scala Likert da "per nulla" (1) a "moltissimo" (5)

L'ultima domanda, infine, consisteva nel valutare l'interesse ad avere un maggiore controllo sul modo in cui il dispositivo e le applicazioni utilizzate gestiscono la propria privacy e sicurezza. Le risposte, in questo caso, sono state nettamente positive, com'è possibile osservare in Figura 5.3.

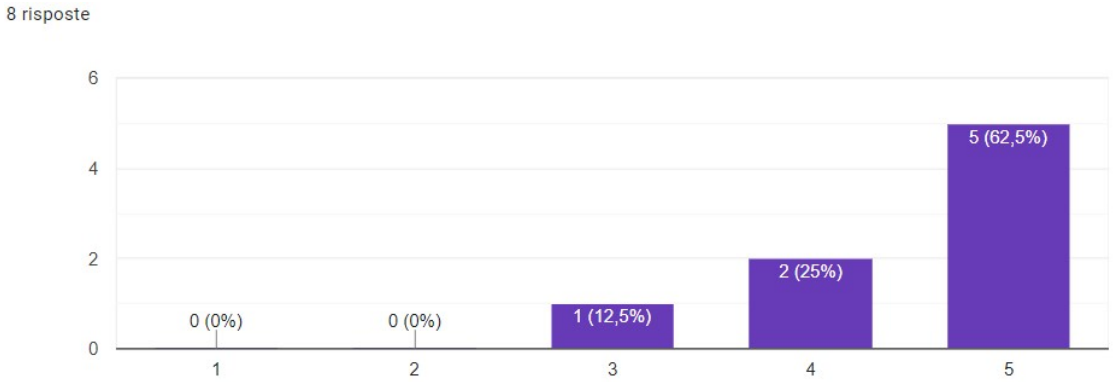


Figura 5.3: Interesse degli utenti per un maggiore controllo sulla propria sicurezza e privacy. Scala Likert da "per nulla" (1) a "moltissimo" (5)

5.2.2 Usabilità

Per valutare l'usabilità del sistema si è fatto utilizzo della System Usability Scale [52]. La valutazione tramite questionario SUS è un metodo comunemente utilizzato per ottenere una stima sull'usabilità del sistema in esame, ottenendo un relativo punteggio in centesimi.

L'applicazione *PrivacyManager* è stata valutata con un punteggio medio di 76.56. Un risultato di questo tipo è superiore al valore medio di 68 dato come riferimento dalla scala, ed indica il generale apprezzamento nell'usabilità dell'applicazione.

Per quanto riguarda le domande aggiuntive al questionario SUS, invece, è da notare l'apprezzamento per la presenza dei tutorial che spiegano come iniziare ad utilizzare l'applicazione, come è possibile vedere in Figura 5.4.

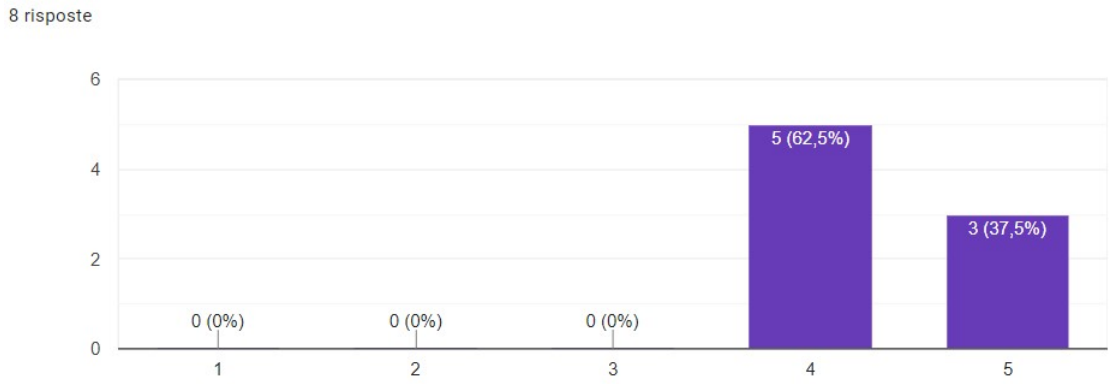


Figura 5.4: Valutazione dell'utilità dei tutorial dell'applicazione. Scala Likert da "per nulla" (1) a "moltissimo" (5)

Infine, nessuno degli utenti intervistati ha segnalato problemi nell'utilizzo dell'applicazione o parti che risultavano poco chiare, dimostrando quindi un corretto funzionamento generale del sistema.

5.2.3 Efficacia

Durante l'utilizzo dell'applicazione nella fase di testing, gli utenti hanno creato in totale 25 regole di sicurezza. Le regole create da ciascun utente variano tra un minimo di 1 ed un massimo di 12. La maggior parte degli utenti (5 su 8) ha creato solamente una regola di sicurezza. È sorprendente il dato relativo a due utenti che hanno creato invece, rispettivamente, 6 e 12 regole di sicurezza a testa. In Figura 5.5 si può vedere il dettaglio del dato raccolto.

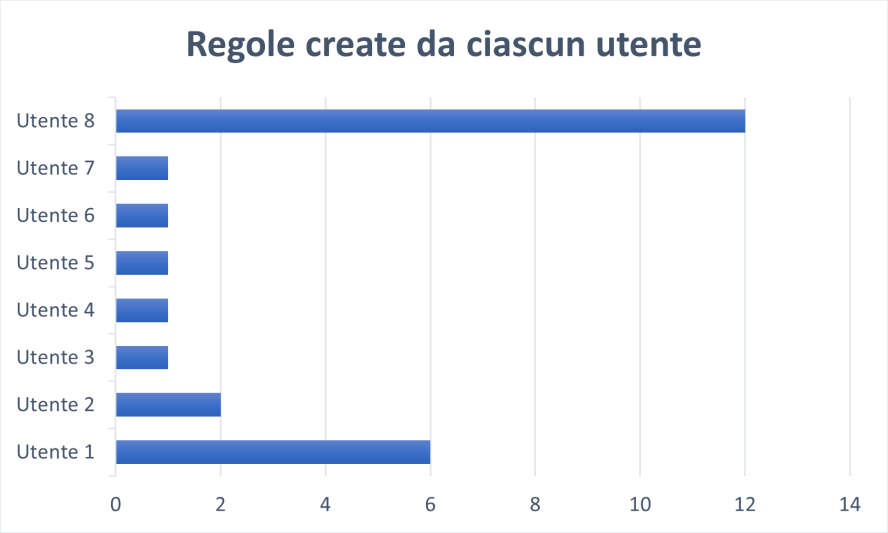


Figura 5.5: Numero di regole create da ciascun utente

Osservando le regole create, si può osservare come queste siano abbastanza eterogenee tra di loro dal punto di vista delle autorizzazioni monitorate Generalmente, si ricava che gli utenti hanno preferito monitorare una singola autorizzazione per regola (72% dei casi), probabilmente quella che ritenevano di maggior interesse. Tra le autorizzazioni singole che si è preferito controllare le preferenze maggiori sono state per le notifiche e la localizzazione (36% e 24% rispettivamente). Il dato relativo alla localizzazione ha senso, in quanto è stato analizzato da diversi studi (di cui si è discusso nei capitoli iniziali) come tra le preoccupazioni maggiori in ambito di privacy rientri proprio il tracciamento della propria posizione geografica. Il dato sulle notifiche, invece, risulta una novità rispetto alle trattazioni iniziali. Le notifiche, infatti, non rappresentano un vero pericolo in ambito di sicurezza dei propri dati, ma gli utenti hanno mostrato invece un grande interesse nel loro controllo.

Nel 16% dei casi, invece, gli utenti hanno creato delle regole per monitorare tutte le autorizzazioni contemporaneamente, probabilmente per testare il funzionamento di PrivacyManager nel suo insieme. Nel restante 12% si sono invece preferite altre combinazioni di due o tre autorizzazioni monitorate. Risulta dunque chiaro come gli utenti abbiano mostrato un maggiore interesse nel monitoraggio delle notifiche e della localizzazioni, com'è osservabile nel grafico riepilogativo di Figura 5.6.

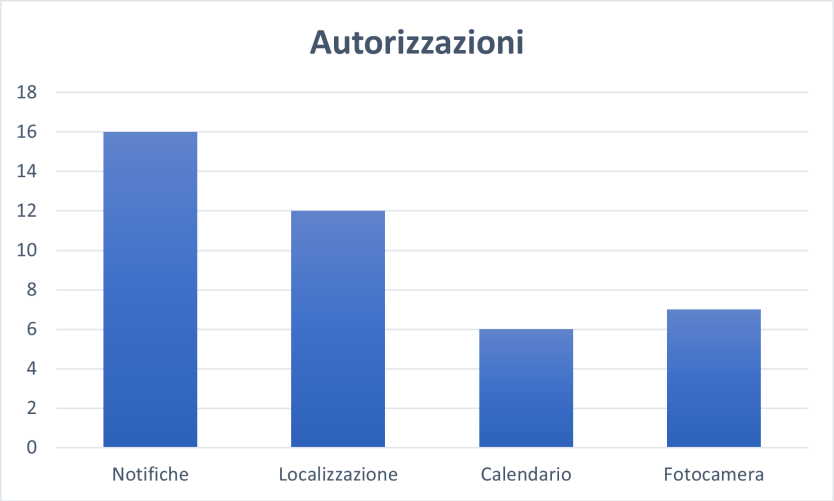


Figura 5.6: Riepilogo delle autorizzazioni maggiormente selezionate durante la creazione di una regola di sicurezza

Per quanto riguarda le condizioni opzionali selezionabili durante la creazione della regola di sicurezza, è invece possibile osservare una maggioranza molto significativa per il parametro relativo allo slot orario. Alternativamente, gli utenti hanno spesso scelto di non aggiungere nessuna condizione, e in questo modo avere il monitoraggio sempre in azione durante l'attivazione della regola. In Figura 5.7 un dettaglio sul dato raccolto.

Un altro parametro importante della regola di sicurezza è rappresentato anche dalle applicazioni da monitorare. In questo caso gli utenti hanno variato molto le loro selezioni nelle varie regole. Infatti, si osserva come in totale siano state selezionate ben 119 applicazioni distinte. Anche in questo caso si può osservare una grande varietà di scelta fatta per ogni regola. Nella maggior parte dei casi, gli utenti hanno preferito monitorare un numero limitato di applicazioni alla volta (tra 1 e 6). Un secondo gruppo di regole, invece, è stato definito per monitorarne un numero più sostanzioso (tra 11 e 15). Risultano infine due regole in cui l'utente ha preferito monitorare la maggior parte delle applicazioni nel proprio dispositivo, com'è possibile vedere in Figura 5.8.

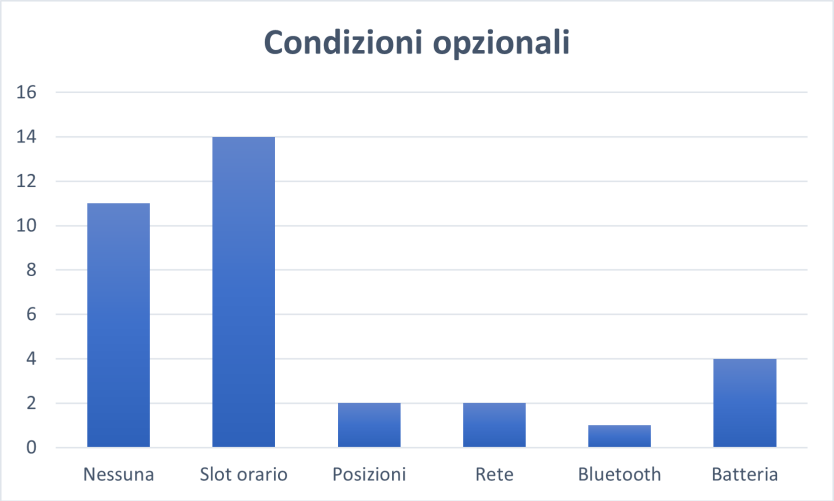


Figura 5.7: Riepilogo delle condizioni opzionali maggiormente selezionate durante la creazione di una regola di sicurezza

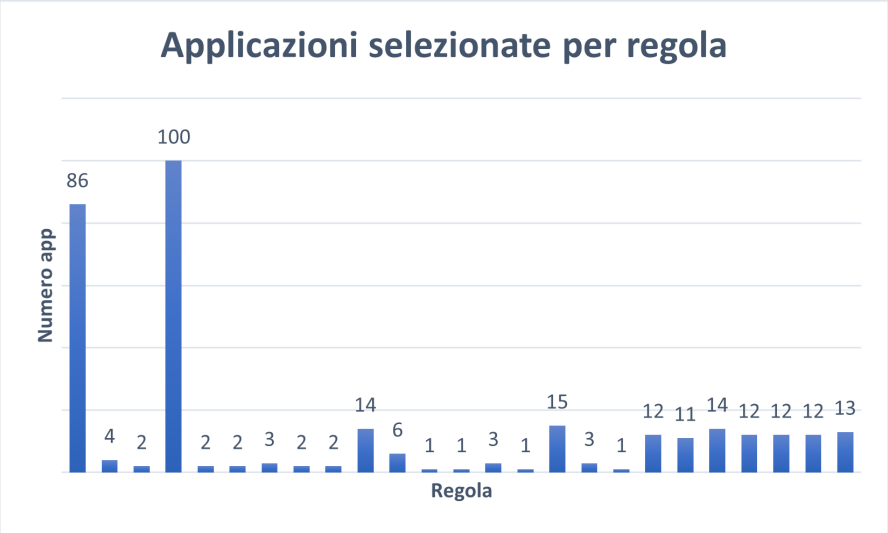


Figura 5.8: Numero di applicazioni monitorate da ciascuna regola

Considerato ciò, è comunque possibile osservare un maggiore interesse riguardo il monitoraggio di certe applicazioni piuttosto che altre. Nel grafico in Figura 5.9 sono mostrate alcune delle app maggiormente selezionate, in particolare 22 tra le applicazioni che sono state selezionate almeno 3 volte.

Per quanto riguarda il numero di attivazioni di ciascuna regola, l'88% di queste è stata attivata una sola volta, probabilmente al momento di creazione della regola, quando dopo il salvataggio viene richiesto se si preferisce attivarla subito. Altre 2

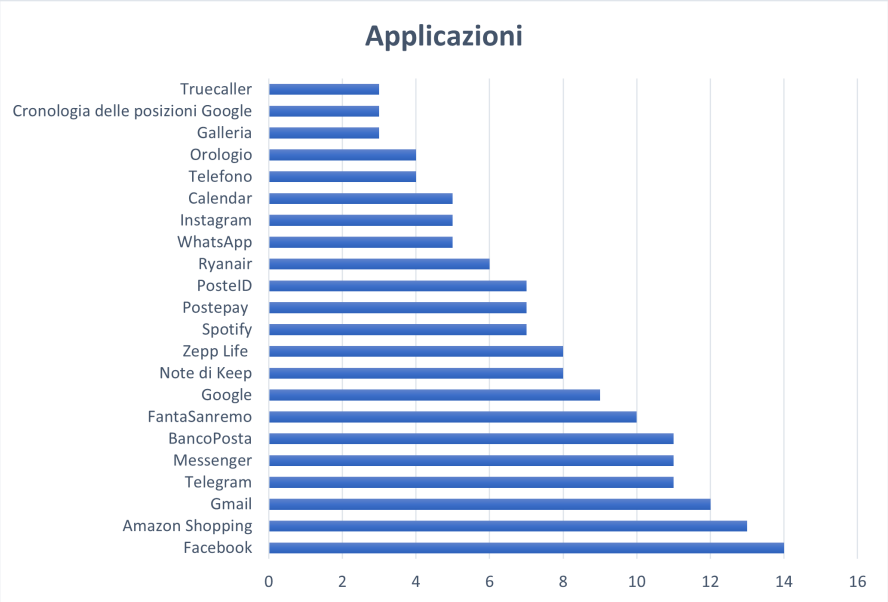


Figura 5.9: Riepilogo delle applicazioni maggiormente selezionate durante la creazione di una regola di sicurezza

regole, invece, sono state attivate per 2 volte. Solamente 1 è stata attivata per 4 volte e un'altra invece non è stata attivata nemmeno una volta. In questo caso, è probabile che si sia trattato di una regola di prova creata dall'utente per prendere confidenza con l'applicazione.

Relativamente al periodo di tempo per cui è stata tenuta attiva la regola di sicurezza, invece, possiamo dividere le regole in 4 gruppi principali in base alla durate del loro periodo di monitoraggio. Il 40% delle regole (la maggioranza) è stato tenuto attivo per tutta la durate della settimana di testing. Ha senso che la maggior parte degli utenti abbia deciso di monitorare il comportamento delle applicazioni per tutto il periodo di prova, così da avere un'idea chiara sull'efficacia di PrivacyManager. È poi presente un gruppo (rappresentante il 28% del totale) che è stato tenuto attivo solamente per pochi secondi. In questo caso si tratta probabilmente di alcune regole di prova create dagli utenti per prendere confidenza con l'applicazione. Una percentuale pari al 12% ha operato invece per qualche minuto (tra i 3 e i 34 minuti). Infine, il 16% delle regole ha monitorato il dispositivo dell'utente per diverse ore (tra le 6 e le 47 ore). In queste due categorie rientrano probabilmente gli utenti che hanno preferito testare il funzionamento delle regole di sicurezza in relazione ad un loro normale utilizzo dello smartphone.

Infine, si osserva dai dati raccolti che le violazioni segnalate dal sistema durante la settimana di utilizzo da parte degli utenti risultano essere 24 in totale. Una

percentuale pari al 75% di queste risulta essere causata dalle notifiche. Le autorizzazioni meno segnalate sono invece il calendario e la fotocamera, segnalate entrambe solamente 1 volta. In Figura 5.10 si può osservare il riepilogo delle violazioni segnalate. Un dato di questo tipo risulta sicuramente schiacciante dal punto di vista della segnalazione delle notifiche, ma è bene comunque tenere in considerazione che siano state proprio queste ultime a rappresentare l'autorizzazione maggiormente monitorata dagli utenti, a discapito soprattutto di calendario e fotocamera, che riportano infatti un numero esiguo di segnalazioni.

Figura 5.10: Autorizzazioni segnalate dalle regole di sicurezza

5.3 Discussione

Il principale obiettivo di questo lavoro di tesi ha riguardato lo sviluppo e il testing di un'applicazione per sistemi Android che permettesse gli utenti di gestire efficacemente e semplicemente la sicurezza e la privacy dei propri dati personali. Per raggiungere questo obiettivo ci si è basati sulle tecniche di End-User Development descritte nei capitoli iniziali, lasciando all'utente la possibilità di creare un artefatto software adatto alle proprie esigenze. Le percezioni degli utenti riguardo il modo in cui le applicazioni utilizzano i propri dati personali e i meccanismi messi a disposizione dai dispositivi per potersi difendere sottolineano come temi di questo tipo siano fonte di insicurezza, pur avendo comunque la consapevolezza necessaria per comprendere i problemi che ne derivano. Partendo da queste basi è stato possibile vedere l'interesse mostrato dagli utenti verso nuovi meccanismi che permettano loro di gestire in maniera più semplice ed efficace la privacy dei dati presenti nel proprio dispositivo.

Durante la settimana di test, gli utenti hanno utilizzato *PrivacyManager* abbastanza frequentemente, e il numero delle regole di sicurezza create lo dimostra. In accordo con le analisi fatte durante la fase di progettazione, i risultati ottenuti dimostrano come una delle autorizzazioni che gli utenti desiderano maggiormente proteggere è quella relativa alla localizzazione, essendo la seconda più selezionata tra tutte le regole di sicurezza (presente nel 48% delle regole). Una scelta molto efficace, invece, si è rivelata essere l'introduzione delle notifiche nell'elenco delle autorizzazioni monitorabili. Infatti, pur non essendo fonti primarie di rischio per la sicurezza degli utenti, questi ultimi sono stati molto propensi a monitorarle durante il periodo di testing, essendo state selezionate per il 64% delle regole create. Meno popolare è invece stata la scelta relativa alle autorizzazioni di calendario e fotocamera, selezionate solamente nel 24% e 28% dei casi rispettivamente. Questo dato dimostra dunque come queste due funzionalità non rappresentino un rischio concreto per gli utenti, che hanno preferito salvaguardare l'accesso alla propria localizzazione e modificare il comportamento relativo alle notifiche inviate dalle app.

Per quanto riguarda le condizioni opzionali che sono state messe a disposizione, una grande parte degli utenti ha deciso di non inserirne nessuna (44% dei casi). Si evince dunque, in questo caso, come gli utenti abbiano deciso di avere il servizio di monitoraggio e segnalazione funzionante per tutto il tempo di attivazione della regola di sicurezza. Tuttavia, si evidenzia come, nella maggior parte dei casi, gli utenti abbiano deciso di inserire uno slot orario (56% dei casi), così da limitare la fase di monitoraggio solamente alle parti della giornata o della settimana che vengono considerate maggiormente sensibili. Poco successo ha invece riscontrato la condizione relativa ai dispositivi bluetooth collegati, selezionata solamente in 1 caso su 25.

Infine, osservando le applicazioni maggiormente selezionate dagli utenti in fase di creazione della regola di sicurezza, si può osserverare come alcune categorie di app vengano considerate più rischiose per la propria privacy, e per questo motivo sono state molto selezionate per la fase di monitraggio. In particolare, si evince come l'interesse maggiore sia posto verso le app social (Facebook, Telegram, WhatsApp, Instagram) e quelle di home banking (BancoPosta, PosteID, PostePay). Da segnalare inoltre una scelta diffusa riguardo l'app di Amazon e di Gmail.

Da queste considerazioni si può dunque ricavare come la filosofia proposta dall'End-User Development sia stata efficace in questo contesto. Infatti, pur evidenziando delle preferenze (alcune molto marcate) nelle scelte degli utenti, è da sottolineare la varietà di parametri selezionati nella definizione delle regole di sicurezza. Il principio di base, infatti, è quello di lasciare nelle mani degli utenti la possibilità di creare parti software che si adattino alle proprie esigenze, perché solamente loro sono in grado di sapere cosa è più efficace per i propri bisogni.

Capitolo 6

Conclusioni

Gli obiettivi principali di questo lavoro di tesi hanno riguardato lo sviluppo e il testing di un'applicazione mobile per dispositivi Android, con lo scopo di aiutare gli utenti a proteggere la propria privacy e fornire uno strumento adattabile alle esigenze personali di ognuno.

Per arrivare a questo fine ci si è avvalsi dell'approccio End-User Development, che ha permesso di creare un sistema estendibile a seconda delle preferenze dell'utente. Analizzando i lavori già esistenti nel Capitolo 2 e studiando l'efficacia di essi in diversi ambiti, si è deciso di creare un sistema di monitoraggio delle funzionalità utilizzate dalle applicazioni del dispositivo utente, realizzato per mezzo di regole di sicurezza definite dall'utente stesso.

In accordo con i dati raccolti e discussi all'interno del Capitolo 5, i risultati ottenuti sono promettenti per quanto riguarda l'utilizzo delle tecniche EUD anche in ambito mobile. Per quanto riguarda l'efficacia dell'applicazione, si può affermare che l'obiettivo di fornire uno strumento adattabile alle esigenze personali degli utenti sia stato raggiunto, così come lo scopo di protezione della privacy raggiunto per mezzo della definizione di regole di sicurezza personalizzate, al netto delle limitazioni tecniche illustrate successivamente.

6.1 Limiti dello studio

Lo studio esposto fino a questo punto è caratterizzato da alcune importanti limitazioni legate alla fase di test:

- il ridotto numero di partecipanti;
- la breve durata della fase di test.

Visto il numero esiguo di partecipanti non è possibile ricavare dei risultati che possano essere generalizzati per un contesto più ampio.

Per quanto riguarda invece il tempo impiegato in questa fase, la settimana di analisi predisposta per gli utenti permette sicuramente di stimare in maniera generale l'efficacia dell'applicazione, ma non permette una valutazione più approfondita delle varie caratteristiche della stessa. La possibilità di inserire condizioni personalizzabili di diverso tipo (come i luoghi o gli slot temporali), ad esempio, permette di dare un elevato grado di flessibilità alle regole di sicurezza, ma a causa della breve durata considerata, non è possibile valutare quanto efficacemente abbiano funzionato i meccanismi di sicurezza in questo contesto.

Oltre ai limiti legati alla fase di test, bisogna tenere in considerazione anche alcune limitazioni che hanno influenzato il processo di progettazione e sviluppo dell'applicazione *PrivacyManager*. Come analizzato nella sezione 3.2.1, infatti, è stato possibile notare negli anni una tendenza da parte di Google nel limitare l'accesso ad alcune funzionalità del dispositivo. Se da un lato questo porta il grosso vantaggio di aumentare la sicurezza e la privacy dei dati degli utenti, dall'altro crea delle restrizioni importanti dal punto di vista degli sviluppatori. Per questo motivo non è stato possibile includere tra le autorizzazioni monitorabili da *PrivacyManager* quelle relative, ad esempio, al microfono o alla memoria del dispositivo, che sono invece considerate molto sensibili secondo la percezione degli utenti.

Da segnalare in questo contesto sono anche alcuni limiti relativi alla rilevazione, e conseguente segnalazione, dell'accesso ad alcune funzionalità che è possibile monitorare con *PrivacyManager*. Per quanto riguarda l'autorizzazione relativa al calendario, infatti, si evidenzia come la possibilità data agli sviluppatori consista nel poter intercettare il caso relativo alla modifica di un evento del calendario da parte di un'applicazione, tuttavia non è possibile ricavare alcuna informazione sull'evento effettivamente modificato (data dell'evento, nome, tipo di modifica effettuata...). Per quanto riguarda invece l'autorizzazione relativa alla localizzazione, come analizzato nella sezione 3.1.2, non è possibile differenziare l'evento relativo ad un'applicazione che accede alla posizione dell'utente per fornire la sua funzionalità principale oppure per effettuare un altro tipo di attività (ad esempio la profilazione a scopo pubblicitario). Di conseguenza, questo porta la possibilità che *PrivacyManager* segnali alcune applicazioni che accedono ai servizi di localizzazione senza però rappresentare un rischio concreto per la privacy dell'utente.

Un'altra restrizione che deriva dai limiti esposti, è rappresentata anche dall'accesso alle informazioni relative alle reti internet salvate dall'utente. Per questo motivo, quando l'utente definisce la condizione opzionale relativa alla rete, è richiesto di inserire manualmente il nome della rete Wi-Fi da considerare per la fase di monitoraggio, piuttosto che presentare una lista delle varie reti memorizzate dal dispositivo. Si tratta tuttavia in questo caso di un limite relativo alla facilità di utilizzo per l'utente, che non ostacola il corretto funzionamento del sistema.

6.2 Sviluppi futuri

Un primo passo da considerare per possibili lavori futuri è quello di considera una fase di test più approfondita per valutare più efficacemente i diversi parametri dell'applicazione.

Nel caso in cui si osservi una maggiore apertura da parte di Google verso le funzionalità che è possibile accedere dal lato degli sviluppatori, mantenendo comunque un adeguato livello di protezione per i dati degli utenti, sarebbe utile offrire la possibilità di monitorare altre autorizzazioni per mezzo delle regole di sicurezza, come quelle relative al microfono o alla memoria accennate precedentemente.

Considerati poi i risultati ottenuti analizzando le statistiche di utilizzo del sistema, e osservando la grande maggioranza di utilizzo che ha avuto lo slot temporale rispetto alle altre condizioni personalizzabili, potrebbe anche essere utile introdurre nuove condizioni che permettano di venire maggiormente incontro alle esigenze di personalizzazione degli utenti.

Appendice A

Questionari

A.1 Questionario iniziale

Domanda	Tipo
Email	Testuale
Età	Testuale
Genere	Scelta multipla
Quanto valuti sufficienti le protezioni della tua privacy e della tua sicurezza offerte dalle applicazioni che utilizzi?	Scala Likert (1-5)
Quanto ritieni semplice il modo in cui il tuo smartphone ti permetta di gestire la tua privacy e la tua sicurezza?	Scala Likert (1-5)
Quanto saresti interessato ad avere un maggiore controllo sul modo in cui il tuo smartphone e le applicazioni che utilizzi gestiscono la tua privacy e la tua sicurezza?	Scala Likert (1-5)

A.2 Questionario finale

Domanda	**Tipo**
Penso che mi piacerebbe utilizzare questa applicazione frequentemente	Scala Likert (1-5)
Ho trovato l'applicazione complessa senza che ce ne fosse troppo bisogno	Scala Likert (1-5)
Ho trovato l'applicazione molto semplice da usare	Scala Likert (1-5)
Penso che avrei bisogno del supporto di una persona già in grado di utilizzare l'applicazione	Scala Likert (1-5)
Ho trovato le varie funzionalità dell'applicazione ben integrate	Scala Likert (1-5)
Ho trovato incoerenze tra le varie funzionalità dell'applicazione	Scala Likert (1-5)
Penso che la maggior parte delle persone potrebbe imparare ad usare l'applicazione facilmente	Scala Likert (1-5)
Ho trovato l'applicazione molto macchinosa da utilizzare	Scala Likert (1-5)
Ho avuto molta confidenza con l'applicazione durante il suo utilizzo	Scala Likert (1-5)

Domanda	**Tipo**
Ho avuto bisogno di imparare molti processi prima di riuscire ad utilizzare al meglio l'applicazione	Scala Likert (1-5)
Quanto hai trovato utili i tutorial iniziali per capire come utilizzare l'applicazione?	Scala Likert (1-5)
Hai trovato parti dell'applicazione che hanno funzionato in modo errato?	Scelta multipla
Se sì, indica quali parti hanno funzionato in maniera errata	Aperta
Hai trovato parti dell'applicazione poco chiare?	Scelta multipla
Se sì, indica quali parti erano poco chiare:	Aperta
Se hai suggerimenti su come modificare o ampliare l'applicazione, scrivili di seguito	Aperta

Bibliografia

[1] Alisa Frik, Juliann Kim, Joshua Rafael Sanchez e Joanne Ma. «Users' Expectations About and Use of Smartphone Privacy and Security Settings». In: *Proceedings of the 2022 CHI Conference on Human Factors in Computing Systems*. CHI '22. New York, NY, USA: Association for Computing Machinery, 2022. ISBN: 9781450391573. DOI: 10.1145/3491102.3517504. URL: https://doi.org/10.1145/3491102.3517504 (cit. alle pp. 1, 2, 11, 12).

[2] Carol Fung, Vivian Motti, Katie Zhang e Yanjun Qian. «A Study of User Concerns about Smartphone Privacy». In: *2022 6th Cyber Security in Networking Conference (CSNet)*. 2022, pp. 1–8. DOI: 10.1109/CSNet56116.2022.9955623 (cit. alle pp. 1, 2, 9, 12).

[3] Adrienne Porter Felt, Elizabeth Ha, Serge Egelman, Ariel Haney, Erika Chin e David Wagner. «Android permissions: user attention, comprehension, and behavior». In: *Proceedings of the Eighth Symposium on Usable Privacy and Security*. SOUPS '12. Washington, D.C.: Association for Computing Machinery, 2012. ISBN: 9781450315326. DOI: 10.1145/2335356.2335360. URL: https://doi.org/10.1145/2335356.2335360 (cit. alle pp. 1, 2, 11).

[4] Yun Zhou, Alexander Raake, Tao Xu e Xuyun Zhang. «Users' perceived control, trust and expectation on privacy settings of smartphone». In: *Cyberspace Safety and Security: 9th International Symposium, CSS 2017, Xi'an China, October 23–25, 2017, Proceedings*. Springer. 2017, pp. 427–441 (cit. alle pp. 1, 2, 11).

[5] Android Documentation. *Android Permissions*. 2023. URL: https://source.android.com/docs/core/permissions (cit. alle pp. 1, 10, 14).

[6] Android Documentation. *Runtime Permissions*. 2023. URL: https://source.android.com/docs/core/permissions/runtime_perms (cit. alle pp. 2, 10).

[7] Abdallah Namoun, Athanasia Daskalopoulou, Nikolay Mehandjiev e Zhang Xun. «Exploring mobile end user development: existing use and design factors». In: *IEEE Transactions on Software Engineering* 42.10 (2016), pp. 960–976 (cit. a p. 3).

[8] Daniela Fogli, Rosa Lanzilotti e Antonio Piccinno. «End-user development tools for the smart home: a systematic literature review». In: *Distributed, Ambient and Pervasive Interactions: 4th International Conference, DAPI 2016, Held as Part of HCI International 2016, Toronto, ON, Canada, July 17-22, 2016, Proceedings 4*. Springer. 2016, pp. 69–79 (cit. alle pp. 3, 6).

[9] Ajay Krishna, Michel Le Pallec, Radu Mateescu e Gwen Salaün. «Design and deployment of expressive and correct web of things applications». In: *ACM Transactions on Internet of Things* 3.1 (2021), pp. 1–30 (cit. alle pp. 3, 6).

[10] Henry Lieberman, Fabio Paternò, Markus Klann e Volker Wulf. «End-user development: An emerging paradigm». In: *End user development*. Springer, 2006, pp. 1–8 (cit. a p. 5).

[11] Amy J Ko et al. «The state of the art in end-user software engineering». In: *ACM Computing Surveys (CSUR)* 43.3 (2011), pp. 1–44 (cit. a p. 5).

[12] W. Harrison. «From the Editor: The Dangers of End-User Programming». In: *IEEE Software* 21.4 (2004), pp. 5–7. DOI: 10.1109/MS.2004.13 (cit. a p. 6).

[13] Bernardo Breve, Giuseppe Desolda, Francesco Greco e Vincenzo Deufemia. «Democratizing Cybersecurity in Smart Environments: Investigating the Mental Models of Novices and Experts». In: *International Symposium on End User Development*. Springer. 2023, pp. 145–161 (cit. a p. 6).

[14] Qi Wang, Wajih Ul Hassan, Adam Bates e Carl Gunter. «Fear and logging in the internet of things». In: *Network and Distributed Systems Symposium*. 2018 (cit. a p. 6).

[15] Camille Cobb, Milijana Surbatovich, Anna Kawakami, Mahmood Sharif, Lujo Bauer, Anupam Das e Limin Jia. «How Risky Are Real Users'{IFTTT} Applets?» In: *Sixteenth Symposium on Usable Privacy and Security (SOUPS 2020)*. 2020, pp. 505–529 (cit. a p. 6).

[16] IFTTT. *Every thing works better together*. 2020. URL: https://ifttt.com/ (cit. a p. 7).

[17] Bernardo Breve, Gaetano Cimino, Giuseppe Desolda, Vincenzo Deufemia e Annunziata Elefante. «On the User Perception of Security Risks of TAP Rules: A User Study». In: *International Symposium on End User Development*. Springer. 2023, pp. 162–179 (cit. a p. 7).

[18] Milijana Surbatovich, Jassim Aljuraidan, Lujo Bauer, Anupam Das e Limin Jia. «Some recipes can do more than spoil your appetite: Analyzing the security and privacy risks of IFTTT recipes». In: *Proceedings of the 26th International Conference on World Wide Web*. 2017, pp. 1501–1510 (cit. a p. 7).

[19] Qi Wang, Pubali Datta, Wei Yang, Si Liu, Adam Bates e Carl A Gunter. «Charting the attack surface of trigger-action IoT platforms». In: *Proceedings of the 2019 ACM SIGSAC conference on computer and communications security*. 2019, pp. 1439–1453 (cit. a p. 8).

[20] Ding Xiao, Qianyu Wang, Ming Cai, Zhaohui Zhu e Weiming Zhao. «A3ID: an automatic and interpretable implicit interference detection method for smart home via knowledge graph». In: *IEEE Internet of Things Journal* 7.3 (2019), pp. 2197–2211 (cit. a p. 8).

[21] Qi Wang, Wajih Ul Hassan, Adam Bates e Carl Gunter. «Fear and logging in the internet of things». In: *Network and Distributed Systems Symposium*. 2018 (cit. a p. 8).

[22] Barbara Rita Barricelli, Fabio Cassano, Daniela Fogli e Antonio Piccinno. «End-user development, end-user programming and end-user software engineering: A systematic mapping study». In: *Journal of Systems and Software* 149 (2019), pp. 101–137 (cit. a p. 8).

[23] MIT. *MIT App Inventor*. 2010. URL: `https://appinventor.mit.edu/` (cit. alle pp. 8, 9).

[24] joaomgcd. *Tasker - Total Automation for Android*. 2010. URL: `https://tasker.joaoapps.com/` (cit. alle pp. 8, 9).

[25] Gabriella Lucci e Fabio Paternò. «Understanding end-user development of context-dependent applications in smartphones». In: *Human-Centered Software Engineering: 5th IFIP WG 13.2 International Conference, HCSE 2014, Paderborn, Germany, September 16-18, 2014. Proceedings 5*. Springer. 2014, pp. 182–198 (cit. a p. 8).

[26] Alan F Westin. «Privacy and freedom». In: *Washington and Lee Law Review* 25.1 (1968), p. 166 (cit. a p. 9).

[27] Hengshu Zhu, Hui Xiong, Yong Ge e Enhong Chen. «Mobile app recommendations with security and privacy awareness». In: *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining*. 2014, pp. 951–960 (cit. a p. 9).

[28] Heng Xu, Sumeet Gupta, Mary Beth Rosson e John M Carroll. «Measuring mobile users' concerns for information privacy». In: (2012) (cit. a p. 9).

[29] Brenda K Wiederhold. *The role of psychology in enhancing cybersecurity*. 2014 (cit. a p. 10).

[30] Android Documentation. *Request runtime permissions*. 2024. URL: `https://developer.android.com/training/permissions/requesting` (cit. a p. 10).

[31] Scott R Moore, Huangyi Ge, Ninghui Li e Robert W Proctor. «Cybersecurity for android applications: Permissions in android 5 and 6». In: *International Journal of Human–Computer Interaction* 35.7 (2019), pp. 630–640 (cit. a p. 11).

[32] Infosecurity Magazine. *92% of Top 500 Android Apps Carry Security or Privacy Risk*. 2014. URL: `https://www.infosecurity-magazine.com/news/92-of-top-500-android-apps-carry-security-or/` (cit. a p. 13).

[33] Google. *Potentially Harmful Applications (PHAs)*. 2019. URL: `https://developers.google.com/android/play-protect/potentially-harmful-applications` (cit. a p. 13).

[34] Android Documentation. *Application Sandbox*. 2024. URL: `https://source.android.com/docs/security/app-sandbox` (cit. a p. 14).

[35] Jian Kang, Doug Steiert, Dan Lin e Yanjie Fu. «MoveWithMe: Location privacy preservation for smartphone users». In: *IEEE Transactions on Information Forensics and Security* 15 (2019), pp. 711–724 (cit. a p. 15).

[36] Kassem Fawaz e Kang G Shin. «Location privacy protection for smartphone users». In: *Proceedings of the 2014 ACM SIGSAC Conference on Computer and Communications Security*. 2014, pp. 239–250 (cit. a p. 15).

[37] Rui Liu, Jiannong Cao, Kehuan Zhang, Wenyu Gao, Junbin Liang e Lei Yang. «When Privacy Meets Usability: Unobtrusive Privacy Permission Recommendation System for Mobile Apps Based on Crowdsourcing». In: *IEEE Transactions on Services Computing* 11.5 (2018), pp. 864–878. DOI: `10.1109/TSC.2016.2605089` (cit. a p. 16).

[38] Android Documentation. *Slide between fragments using ViewPager2*. 2024. URL: `https://developer.android.com/develop/ui/views/animations/screen-slide-2` (cit. alle pp. 38, 45).

[39] Android Documentation. *Foreground services*. 2024. URL: `https://developer.android.com/develop/background-work/services/foreground-services` (cit. alle pp. 38, 44).

[40] Android Documentation. *Learn about restricted settings*. 2024. URL: `https://support.google.com/android/answer/12623953` (cit. a p. 41).

[41] Android Documentation. *Intent*. 2024. URL: `https://developer.android.com/reference/kotlin/android/content/Intent` (cit. a p. 44).

[42] Android Documentation. *Handler*. 2024. URL: `https://developer.android.com/reference/kotlin/android/os/Handler` (cit. a p. 47).

[43] Android Documentation. *UsageStatsManager*. 2024. URL: `https://developer.android.com/reference/kotlin/android/app/usage/UsageStatsManager` (cit. a p. 47).

[44] Android Documentation. *PackageManager*. 2024. URL: `https://developer.android.com/reference/kotlin/android/content/pm/PackageManager` (cit. a p. 48).

[45] Android Documentation. *ContentObserver*. 2024. URL: `https://developer.android.com/reference/kotlin/android/database/ContentObserver` (cit. a p. 48).

[46] Android Documentation. *CameraManager*. 2024. URL: `https://developer.android.com/reference/kotlin/android/hardware/camera2/CameraManager` (cit. a p. 48).

[47] Android Documentation. *NotificationListenerService*. 2024. URL: `https://developer.android.com/reference/kotlin/android/service/notification/NotificationListenerService` (cit. a p. 49).

[48] Android Documentation. *ActivityManager*. 2024. URL: `https://developer.android.com/reference/kotlin/android/app/ActivityManager` (cit. a p. 49).

[49] Android Documentation. *SharedPreferences*. 2024. URL: `https://developer.android.com/reference/kotlin/android/content/SharedPreferences` (cit. a p. 54).

[50] Google. *Firebase Authentication*. 2024. URL: `https://firebase.google.com/docs/auth` (cit. a p. 55).

[51] Google. *Firestore*. 2024. URL: `https://firebase.google.com/docs/firestore` (cit. a p. 55).

[52] John Brooke. «Sus: a "quick and dirty'usability». In: *Usability evaluation in industry* 189.3 (1996), pp. 189–194 (cit. alle pp. 58, 60).