Indice

1 - Introduzione

Shared Secrets

Shared Secrets è un'applicazione Android per la condivisione di segreti all'interno di un gruppo.
Tramite quest'applicazione, infatti, è possibile creare gruppi, oppure unirsi, previa autenticazione, a gruppi già esistenti. All'interno dei gruppi, inoltre, si può condividere uno spazio segreto di 256 caratteri, accessibile solo ai membri.
Ciò che rende veramente particolare quest'applicazione però è la modalità di accesso ai segreti. Non basta, infatti, la semplice iscrizione al gruppo, ma è necessario che una certa percentuale di membri (dichiarata al momento della creazione del gruppo) appartenenti al gruppo, sia presente nello stesso luogo in cui si vuole svelare il segreto; in caso contrario, il segreto resterà inaccessibile.

2 - Analisi dei requisiti

Prerequisiti

Shared Secrets è un'applicazione pensata per funzionare su dispositivi Android, in particolare è stata sviluppata per dispositivi che montino tutte le versioni di tale sistema operativo a partire da Jelly Bean (versione 4.1), così da permetterne l'esecuzione su oltre il 95% dei dispositivi Android attualmente in uso.
Per funzionare correttamente, inoltre, il dispositivo sul quale l'applicazione viene installata, deve poter accedere alla rete. Come si vedrà in seguito, infatti, la persistenza dei dati è gestita quasi interamente in remoto, senza di essa dunque risulta impossibile poter accedere alle informazioni riguardanti i gruppi e i loro membri.
Risulta inoltre quasi altrettanto fondamentale poter utilizzare il Global Positioning System (GPS): se tutti i dispositivi legati ad un gruppo non usufruissero di tale sistema, oppure non ne consentissero l'utilizzo, si verrebbe a creare uno stato per cui, solo nei gruppi con impostata una percentuale necessaria di amici nelle vicinanze per svelare il segreto dello 0%, sarebbe assicurato un comportamento dell'applicazione perfettamente aderente alle attese (in quanto tale percentuale scavalcherebbe le informazioni relative alla localizzazione). In tutti gli altri casi, invece, la presenza di membri che non abbiano a disposizione (o che scelgano di non utilizzare) un sistema GPS, potrebbe essere causa di comportamenti anomali dell'applicazione. Disabilitando la localizzazione, inoltre, la funzionalità di geolocalizzazione dei membri dei propri gruppi risulterà totalmente inaffidabile.
In sostanza l'utente che voglia utilizzare *Shared Secrets* si impegna ad installare tale applicazione su un dispositivo che monti Android Jelly Bean o versioni successive e a fornire a tale applicazione accesso alla rete e permessi per effettuare la geolocalizzazione.

Requisiti Richiesti

I requisiti richiesti dal committente dell'applicazione sono stati i seguenti:

- Possibilità di condividere un segreto con un gruppo di persone da parte di un utente,
- Possibilità di poter svelare il segreto se un congruo numero di persone appartenenti allo stesso gruppo si ritrova nello stesso luogo.

Tali requisiti espressi in linguaggio naturale, sono stati successivamente rielaborati ed ampliati al fine di offrire un'esperienza d'uso il più completa possibile all'utente.
Questa operazione ha portato alla stesura e alla successiva implementazione dei seguenti casi d'uso:

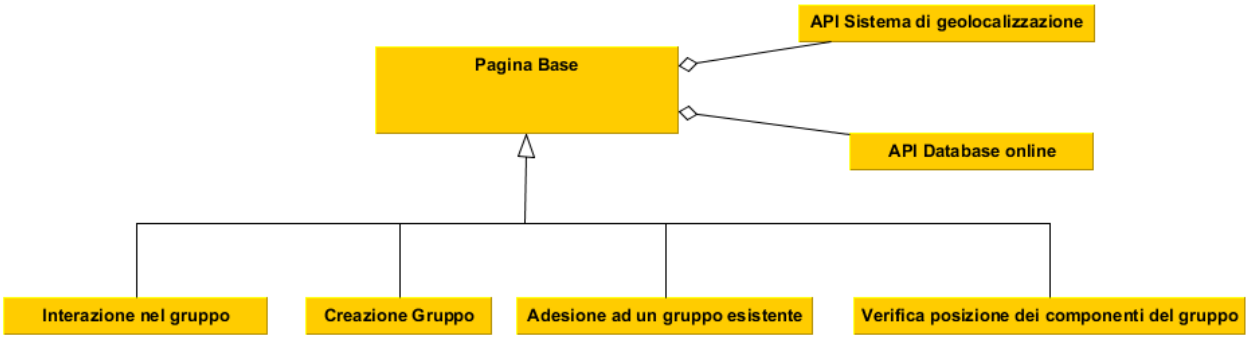

Creazione Gruppo
Adesione a un Gruppo
<<include>>
Verifica del possesso
delle credenziali del gruppo
User
Visualizzazione del segreto
di un gruppo
<<include>>
<<include>>
Selezione di uno
dei propri gruppi
Verifica della vicinanza
di un numero sufficiente
di componenti di un gruppo
<<include>>
<<include>>
Modifica del segreto
di un gruppo
Visualizzazione della posizione
dei componenti di un gruppo
Creazione di un profilo utente
legato al dispositivo utilizzato
<<inlcude>>

3 - Struttura del sistema

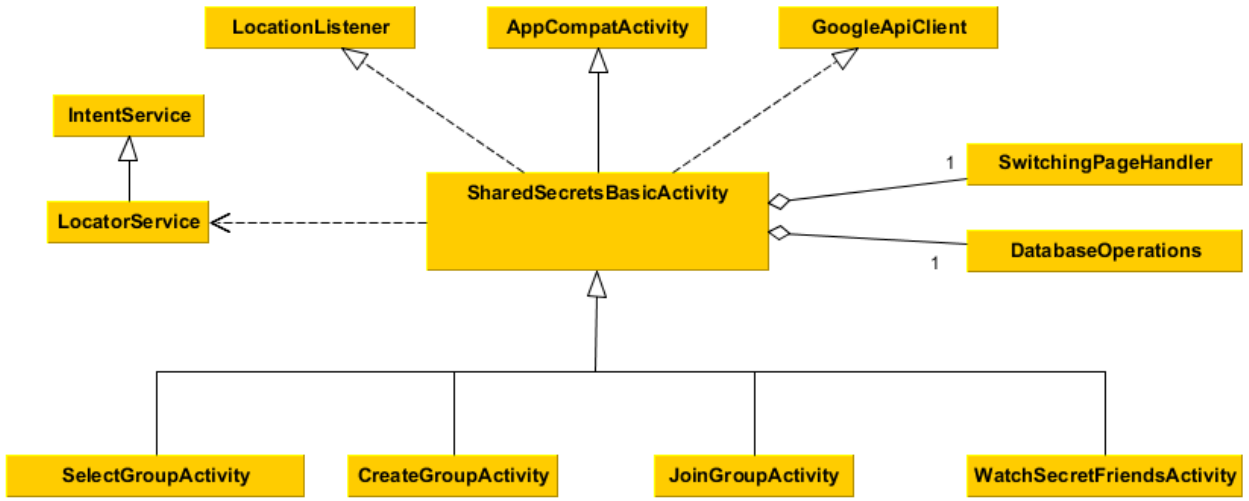
Per la natura dell'applicativo sopracitato, si è reso necessario lo sviluppo parallelo di due sistemi: uno lato client che si occupasse dell'interazione con l'utente e della raccolta ed elaborazione dati, e un secondo sistema lato server che consentisse di immagazzinare tali dati in modo persistente e li rendesse disponibili in tempo reale a tutti gli utenti dell'applicativo. Di seguito verranno esposte le strutture alla base di tali sistemi.

Struttura del sistema lato client

In base ai requisiti visti in precedenza, è stata modellata una prima struttura di partenza del sistema dalla quale si è partiti. Di seguito una rappresentazione di tale modello:



Successivamente, in fase di implementazione, tale schema di partenza è stato rielaborato fino a giungere alla versione finale qui riportata:



Di seguito verranno esplicitate le principali funzioni di ciascuno dei componenti sopra-illustrati.

Shared Secrets Basic Activity

Tale Activity rappresenta il cuore di tutta l'applicazione, imposta e inizializza tutti i sistemi utili all'esecuzione. In particolare Database Operations, che gestisce le comunicazioni col server; Switching Page Handler, il quale si occupa di gestire il passaggio da un Activity all'altro con relativo passaggio di dati; Locator Service, che si occupa di comunicare gli spostamenti dell'utente al server, e infine implementa le interfacce Location Listener e Google Api Client, tramite le quali è in grado di rilevare gli spostamenti dell'utente mentre l'applicativo è in esecuzione. Gestisce inoltre l'arrivo di dati in arrivo da altri Activity precedentemente usati e l'interazione con le Shared Preferences per verificare il primo accesso dell'utente.

Select Group Activity

Tale Activity consente di espletare tre operazioni fondamentali:

- la scelta del gruppo (tra quelli a cui ha aderito) da parte di un utente, da un menù a tendina situato in alto. Nel caso in cui l'utente non avesse ancora aderito né creato alcun gruppo, tale menu risulterà vuoto. Questo elemento viene popolato al momento della creazione dell'Activity effettuando una richiesta al server, a patto che l'utente sia connesso alla rete.
- La possibilità di modificare il segreto del gruppo selezionato. *È presente un campo testuale editabile nel quale è possibile inserire una stringa che andrà a sostituire l'attuale segreto del gruppo. Quando l'operazione di modifica viene confermata tramite la pressione di un apposito tasto, a patto che si sia connessi, viene effettuata una richiesta di modifica. Se nelle vicinanze dell'utente che ha effettuato la richiesta vi sono sufficienti membri del gruppo, allora la richiesta viene accolta e il segreto modificato.*
- *La possibilità di visualizzare il segreto del gruppo selezionato. Analogamente alla modifica, nel momento in cui viene premuto il pulsante per la richiesta di visualizzazione del segreto, tale richiesta viene inoltrata al server che verifica se accoglierla o meno.*

Join Group Activity

Tale Activity consente ad un utente di unirsi ad un gruppo già esistente. *È considerato un requisito necessario e sufficiente per unirsi ad un gruppo, conoscerne sia il nome che la password. Per questo motivo all'interno di questo Activity sono presenti un campo Password e un campo Nome Gruppo, una volta riempiti e confermata la richiesta di ammissione, viene effettuata una richiesta al server per verificare che le credenziali siano corrette.*

Create Group Activity

In questa pagina è possibile creare un nuovo gruppo. In particolare un gruppo è caratterizzato da un nome, una password, un segreto e una percentuale di amici da avere vicino per svelarne il segreto. All'interno di questa pagina sono presenti i campi per delineare tali dati ed effettuare la richiesta al server. Nel momento in cui la richiesta di creazione del gruppo viene accolta, il gruppo viene creato sul server e il creatore ne diviene automaticamente un membro. Come ulteriore vincolo alla creazione di gruppi, si è scelto di rendere univoco il nome di ciascun gruppo, così da non creare confusione nell'utente nel momento in cui diventasse membro di due gruppi omonimi. Per questo motivo, qualsiasi richiesta di creazione di un gruppo che preveda un nome già esistente, è rifiutata.

Watch Secret Group Activity

Tramite tale pagina è possibile monitorare, a patto di avere accesso a internet, l'ultima posizione rilevata da *Shared Secrets*, dei membri del gruppo selezionato nella apposita pagina di selezione. Tramite l'utilizzo delle API di Google, tale pagina richiede la creazione di una mappa e, effettuando una richiesta al server, recupera le posizioni di tutti i membri del gruppo selezionato, infine provvede a posizionare un marker sulla mappa in ciascuna delle posizioni comunicate dal server.

Locator Service

Tutte le operazioni che comportassero il coinvolgimento del server descritto in precedenza, sono effettuate sostanzialmente in modo sincrono con l'ausilio di un thread usa e getta, creato per effettuare la richiesta su server. Tale politica però non viene adottata per l'invio di dati geografici (nella sezione dedicata alle politiche implementative verranno spiegate tali scelte), in questo caso, infatti, viene creato un service apposito il quale, in modo completamente asincrono, si occupa di trasmettere i dati geografici al server.

Database Operations

Tutte le operazioni che coinvolgano richieste al server, usano il Database Operations. Tale sistema, infatti, funge da interfaccia per l'applicativo lato client, consentendo di effettuare richieste al server tramite i suoi metodi, in modo del tutto trasparente.

Struttura del sistema Lato server

Struttura del server

Ciascuna delle richieste server sopracitate viene inoltrata in rete tramite il Database Operations, ognuna di queste richieste viene poi gestita tramite un apposito script PHP presente sul server, che opera su un database MySql.
Di seguito verranno presentati gli script PHP che gestiscono le richieste per ciascuna delle operazioni implementate:

- Connessione al DB

```php
$con=mysqli_connect([redacted]);
/* check connection */
if (mysqli_connect_errno()) {
    echo "Connection failed: ". mysqli_connect_error();
    exit();
}
```

- Recupero gruppi di un utente

```php
$query="SELECT DISTINCT(`GroupName`) FROM `Group` WHERE `Id`
      IN (SELECT `GroupId` FROM `GroupDevice` WHERE `UserId`='".$_GET["UserId"]."')";
if ($result = mysqli_query($con, $query)) {
    while ($obj = $result->fetch_object()) {
        echo "";
        echo $obj->GroupName;
        echo ",";
    }
    mysqli_free_result($result);
}
```

- Inserimento di un utente

```php
$query="INSERT INTO `UserDevice`(`UserId`, `LastLat`, `LastLong`)
      VALUES ('".$_GET["UserId"]."','".$_GET["Lat"]."','".$_GET["Long"]."')";
if (mysqli_query($con,$query)){echo "Operation Successfully Done";
}else{echo "ERROR: couldn't create your user profile";}
```

- Aggiunta di un utente ad un gruppo

```
$query="SELECT `Id` FROM `Group` WHERE `GroupName`='".$_GET["GroupName"]."'
        AND `Password`='".$_GET["Password"]."'";
if ($result = mysqli_query($con, $query)) {
    $count=0;
    while ($obj = $result->fetch_object()) {
        $id = $obj->Id;
        $count++;
    }
    if($count==0){
        echo "ERROR: GroupName OR Password incorrect";
    }else{
        $query="INSERT INTO `GroupDevice`(`GroupId`, `UserId`)
                VALUES ('".$id."','".$_GET["UserId"]."')";
        if (mysqli_query($con,$query)){
            echo "Operation Successfully Done";
        }else{echo "ERROR: can't let you join this group twice";}
    }
    mysqli_free_result($result);
}
```

- Creazione di un gruppo

```
$query="INSERT INTO `Group`(`GroupName`, `Password`, `Secret`,`FriendsPercentage`)
      VALUES ('".$_GET["GroupName"]."','".$_GET["Password"]."','".$_GET["Secret"]."','".$_GET["Friends"]."')";
if (mysqli_query($con,$query)){
    $query="SELECT `Id` FROM `Group` WHERE `GroupName`='".$_GET["GroupName"]."' AND `Password`='".$_GET["Password"]."'";
    if ($result = mysqli_query($con, $query)) {
        $count=0;
        while ($obj = $result->fetch_object()) {
            $id = $obj->Id;
            $count++;
        }
        if($count==0){
            echo "ERROR: GroupName OR Password incorrect";
        }else{
            $query="INSERT INTO `GroupDevice`(`GroupId`, `UserId`) VALUES ('".$id."','".$_GET["UserId"]."')";
            if (mysqli_query($con,$query)){
                echo "Operation Succesfully Done";
            }else{
                echo "ERROR: could't let you join the created group";
            }
        }
        mysqli_free_result($result);
    }

}else{
    echo "ERROR: a Group with this name still exist, please choose another name";
}
```

- Aggiornamento di un segreto

```
$query="SELECT (`FriendsPercentage`/100)AS `FriendsPercentage`,(SELECT COUNT(`UserId`)/(SELECT COUNT(`UserId`)
        FROM `UserDevice` WHERE `UserId`IN (SELECT `UserId` FROM `GroupDevice` WHERE `GroupId`IN(SELECT `Id` FROM
        `Group` WHERE `GroupName`='".$_GET["GroupName"]."')))FROM  `UserDevice` WHERE `LastLat`=(SELECT `LastLat` FROM
        `UserDevice` WHERE `UserId`='".$_GET["UserId"]."' ) AND `LastLong`=(SELECT `LastLong` FROM `UserDevice` WHERE
        `UserId`='".$_GET["UserId"]."' ) AND`UserId` IN (SELECT `UserId` FROM `GroupDevice` WHERE `GroupId`IN
        (SELECT `Id` FROM `Group` WHERE `GroupName`='".$_GET["GroupName"]."' ) ))AS `RealFriendsPercentage`FROM `Group`
        WHERE `GroupName`='".$_GET["GroupName"]."' ";
if ($result = mysqli_query($con, $query)) {
    while ($obj = $result->fetch_object()) {
        $declaredPercentage = $obj->FriendsPercentage;
        $realPercentage =$obj->RealFriendsPercentage;

    }
    if($declaredPercentage>$realPercentage){
        echo "ERROR: you need some other friends near you to uncover the secret";
    }else if($declaredPercentage<=$realPercentage){
        $query="UPDATE `Group` SET `Secret`='".$_GET["Secret"]."' WHERE `GroupName`='".$_GET["GroupName"]."'";
        if (mysqli_query($con,$query)){
            echo "Operation Succesfully Done";
        }else{
            echo "ERROR: couldn't modify the secret";
        }
    }else{
        echo "ERROR: db misfunction";
    }
}else{
    echo "ERROR: db misfunction";
}
```

- Lettura di un segreto

```
$query="SELECT (`FriendsPercentage`/100)AS `FriendsPercentage`,(SELECT COUNT(`UserId`)/(SELECT COUNT(`UserId`)
        FROM `UserDevice` WHERE `UserId`IN (SELECT `UserId` FROM `GroupDevice` WHERE `GroupId`IN
        (SELECT `Id` FROM `Group` WHERE `GroupName`='".$_GET["GroupName"]."')))FROM  `UserDevice` WHERE `LastLat`=
        (SELECT `LastLat` FROM `UserDevice` WHERE `UserId`='".$_GET["UserId"]."' ) AND `LastLong`=
        (SELECT `LastLong` FROM `UserDevice` WHERE `UserId`='".$_GET["UserId"]."' ) AND`UserId` IN
        (SELECT `UserId` FROM `GroupDevice` WHERE `GroupId`IN(SELECT `Id` FROM `Group`
        WHERE `GroupName`='".$_GET["GroupName"]."' ) ))AS `RealFriendsPercentage`FROM `Group`
        WHERE `GroupName`='".$_GET["GroupName"]."' ";
if ($result = mysqli_query($con, $query)) {
    while ($obj = $result->fetch_object()) {
        $declaredPercentage = $obj->FriendsPercentage;
        $realPercentage =$obj->RealFriendsPercentage;

    }
    if($declaredPercentage>$realPercentage){
        echo "ERROR: you need some other friends near you to get the secret".$declaredPercentage." ".$realPercentage;
    }else if($declaredPercentage<=$realPercentage){
        $query="SELECT `Secret` FROM `Group` WHERE `GroupName`='".$_GET["GroupName"]."'";
        if ($result = mysqli_query($con, $query)) {
            $count=0;
            while ($obj = $result->fetch_object()) {
                echo $obj->Secret;
                $count++;
            }
            if($count == 0){
                echo "ERROR: couldn't retrieve the secret";
            }
        }else{echo "ERROR: db misfunction";}
    }else{echo "ERROR: db misfunction";}
}else{echo "ERROR: db misfunction";}
```

- Aggiornamento posizione dell'utente

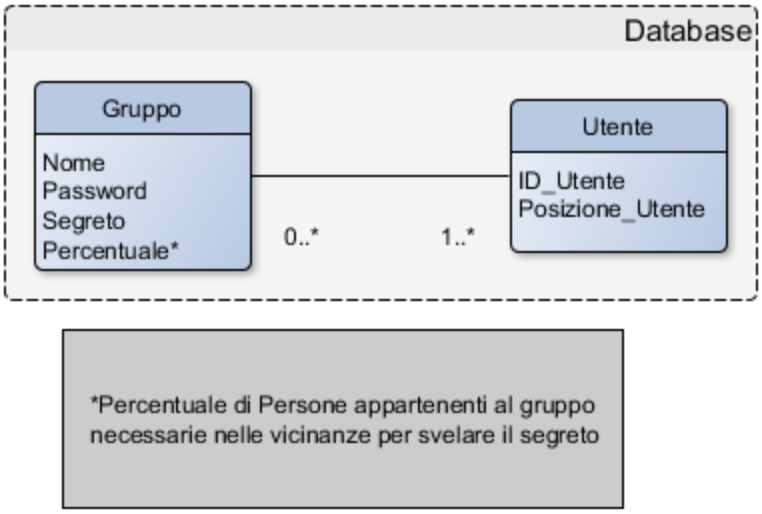
```
$query="UPDATE `UserDevice` SET `LastLat`='".$_GET["Lat"]."',`LastLong`='".$_GET["Long"]."'
        WHERE `UserId`='".$_GET["UserId"]."'";
if (mysqli_query($con,$query)){
    echo "Operation Succesfully Done";
}else{
    echo "ERROR: couldn't modify the position";
}
```

- Recupero della posizione dei componenti di un gruppo

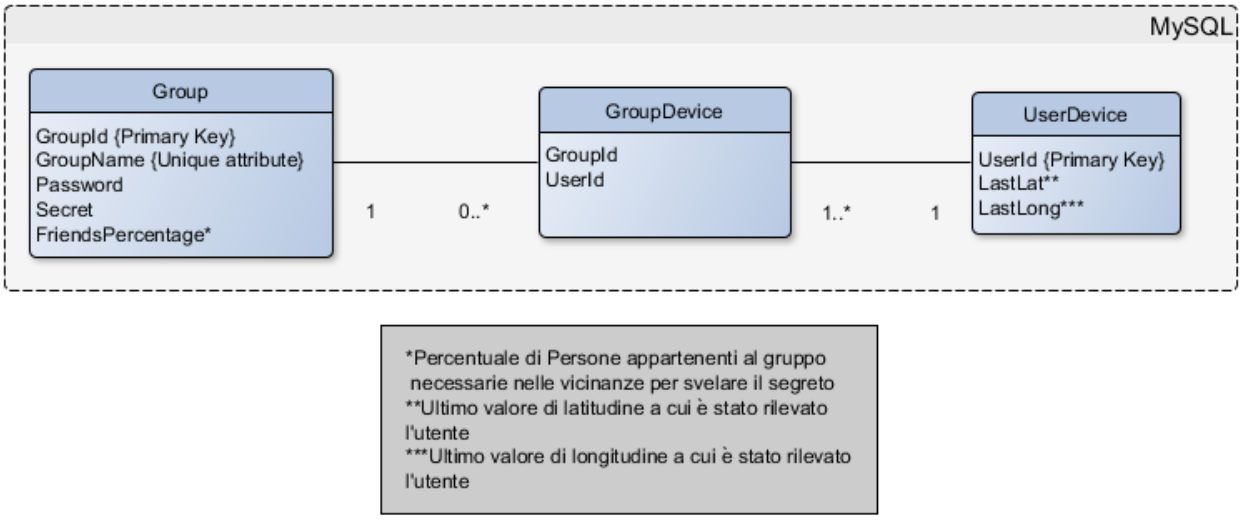
```php
$query="SELECT `LastLat`,`LastLong` FROM `UserDevice` WHERE `UserId`IN
        (SELECT `UserId` FROM `GroupDevice`WHERE `GroupId` IN
        ( SELECT `Id`FROM `Group`WHERE `GroupName`='".$_GET["GroupName"]."'))";
if ($result=mysqli_query($con,$query)){
    $count=0;
    while ($obj = $result->fetch_object()) {
        if($count!=0){
            echo "--";
        }
        echo $obj->LastLat;
        echo ",,";
        echo $obj->LastLong;

        $count++;
    }
    if($count == 0){
        echo "ERROR: couldn't retrieve the secret";
    }
}else{
    echo "ERROR: couldn't retrieve the positions of the group members";
}
```

Struttura del database

Gli script precedentemente illustrati operano su un database progettato nel seguente modo:



Tale progetto iniziale è stato successivamente tradotto su database MySql con la seguente struttura:



4 - Politiche di sviluppo adottate

Politiche di gestione della sicurezza adottate

Poiché l'applicativo si basa sulla condivisione di segreti, risulta assolutamente fondamentale qualsiasi scelta riguardante la sicurezza dell'applicativo. In primo luogo è stato scelto di non consentire all'utente di creare direttamente un proprio profilo, ma ci si è affidati alla classe Settings Secure, dalla quale è possibile ottenere, tramite l'invocazione di un apposito metodo, un id univoco per il dispositivo. Tale id viene usato da *Shared Secrets* come identificativo per l'utente; così facendo, si rende il profilo creato indissolubilmente legato al dispositivo dell'utente e, in questo modo, risulta impossibile accedere ai segreti di un utente se non si è in possesso del suo device. L'accesso al segreto di un gruppo non è possibile senza l'adesione al gruppo stesso; per questo motivo l'adesione ai gruppi non è libera, ma è vincolata di volta in volta da una password della quale l'utente interessato ad un determinato gruppo, deve necessariamente essere fornito.

Politiche di geolocalizzazione adottate

Per quanto riguarda il sistema di geolocalizzazione, si è scelto di affidarsi al FLP (Fused Location Provider) del Google Play Services. Tale sistema, previa concessione da parte dell'utente di poter usufruire dei sistemi di localizzazione, consente di fondere le info provenienti da WPS (Wi-fi Positioning System), GPS e Cell-ID per determinare la posizione dell'utente. Poiché determinare correttamente la posizione dell'utente può fare la differenza tra poter visualizzare il segreto o meno, si è scelto di sacrificare il consumo di batteria, per ottenere in cambio una elevata accuratezza da parte del FLP. Il sistema sviluppato richiede la posizione al FLP ogni 9 secondi, tempo giudicato sufficiente per rilevare spostamenti significativi in tempo reale da parte dell'utente e, allo stesso tempo, non sacrificare eccessivamente la performance, visto che il processo atto a salvare la posizione dell'utente su server è particolarmente impegnativo.

Politiche di trasmissione dei dati

Dati trasmessi al server

Si è scelto di creare un thread ad-hoc per le operazioni su server, che operasse in modo sincrono con il main thread, per ogni operazione richiesta direttamente dall'utente. Ciò ha comportato da un lato un lieve ritardo nella risposta dell'applicativo e ha reso il sistema, così

per come è visto dall'utente, soggetto alla velocità della connessione; d'altro canto è risultato un po' insensato accogliere una richiesta dell'utente e poi consentirgli di continuare ad effettuare operazioni senza prima avere una risposta. Consentire all'utente di poter effettuare più operazioni contemporaneamente sul proprio profilo, in possibili condizioni di scarsa connettività, inoltre, aumenterebbe considerevolmente il rischio di lasciare il database in uno stato inconsistente; il fatto stesso che l'applicazione aspetti il completamento di tali operazioni e dunque le esegua in modo del tutto sequenziale, aiuta a scongiurare questo pericolo.

La scelta di aggiornamento delle informazioni riguardanti la geolocalizzazione che, come detto in precedenza, è stata affidata al Locator Service, è stata invece decisamente differente. Si è preferito infatti far sì che l'invio di coordinate fosse del tutto trasparente all'utente e inficiasse il meno possibile la performance dell'applicativo. Per questo motivo, il Locator Service viene avviato in Background all'interno di un processo differente rispetto a quello utilizzato per l'esecuzione del main thread, riceve, come precedentemente detto, un aggiornamento ogni 9 secondi e, in modo autonomo, provvede ad inviare l'aggiornamento al server.

Shared Preferences

Per evitare di dover verificare, ad ogni avvio, tramite una richiesta al server, che il profilo utente sia già stato creato, che non sia la prima volta che il telefono utilizzi l'applicazione e, quindi, per ottimizzare la performance; si utilizza un campo salvato in locale sul dispositivo, tramite il metodo delle Shared Preferences.

5 – Sviluppi Futuri

L'applicazione si presta a molti possibili ampliamenti: in primo luogo, sarebbe ideale passare da un database di tipo classico ad uno spaziale, che gestisca in modo migliore le query riguardanti Latitudine e Longitudine, tramite l'ausilio di metodi d'indicizzazione ad-hoc e query semplificate.
Sarebbe inoltre auspicabile passare dalla presente struttura del server ad un sistema definito in modo più formale, così da migliorare principi architetturali quali scalabilità o modificabilità. A questo proposito una situazione valida individuata potrebbe essere lo sviluppo di un web service ReSTful.
Infine potrebbe essere utile ampliare le funzionalità a disposizione dell'utente, aggiungendo, ad esempio, la figura di un utente amministratore che abbia la possibilità di eliminare gruppi o escludere persone da un gruppo.