



A.D. 1308
unipg

UNIVERSITÀ DEGLI STUDI
DI PERUGIA

Remote Soul Project

Progetto Ingegneria del Software aa. 2021/22

Docente corso: Alfredo Milani

Studente:

Matteo Volpi [341116]



Panoramica

Remote Soul Project nasce dalla volontà, e con l'obiettivo, di realizzare un sistema capace di interpretare comandi remoti ricevuti tramite SMS da numero telefonico autorizzato. La ricezione di questi ultimi provocherà (lato client del progetto) l'esecuzione di uno o più comandi, quali: scattare una foto, riprodurre musica, registrare un audio ed inviare una foto.

Obiettivi

1. Fornire l'Analisi dei requisiti che ha permesso di intraprendere e proseguire lo sviluppo di Remote Soul Project;
2. Fornire l'Architettura del Sistema, pensata in fase di progettazione, attraverso opportuni diagrammi UML ed eventuali precisazioni scritte;
3. Fornire opportuni casi di Test Funzionali per aspetti del progetto ritenuti significativi;
4. Fornire il sorgente prodotto per l'implementazione del progetto disegnato in fase di progettazione.

Per punti

I. Documento dei requisiti

II. Architettura del Sistema

Diagramma di classe completo

Diagramma di Caso d'Uso

Diagramma di Sequenza e Collaborazione

Diagramma di Stato di elemento ritenuto significativo

Diagramma di una Attività del sistema

III. Documentazione Implementazione

Presentazione dell'implementazione prodotta

Breve discussione sulle scelte implementative e Test Funzionali

Documento dei Requisiti

Prefazione

Il seguente documento vuole essere di riferimento, per la comprensione di Remote Soul Project, per qualsivoglia categoria di lettore. In particolar modo tale documento ha l'obiettivo di risultare di facile, e non ambigua, comprensione alle seguenti categorie: Manager del cliente, Ingegneri del Cliente, Progettisti del sistema, Sviluppatori del sistema, Utenti finali del sistema.

Glossario

SMS: con sms si fa riferimento al servizio di telefonia mobile che permette l'invio di brevi messaggi testuali.

Automaticamente: quando, nella specifica dei comandi eseguibili dal sistema Remote Soul, si utilizza il termine "automaticamente" si intende: senza interazione dell'utente.

Attività: insieme di operazioni eseguite in seguito alla ricezione di un particolare comando.

Definizione e Specifica Requisiti

1. Il Sistema Remote Soul dovrà permettere all'utente l'invio e la ricezione di **SMS** contenenti messaggi testuali.
2. Tali messaggi testuali potranno corrispondere a uno o più comandi che il sistema dovrà eseguire.
 - 2.1 Il sistema dovrà essere in grado di verificare se il numero telefonico dal quale proviene l'SMS risulta essere un numero **autorizzato** a richiedere l'esecuzione del comando specificato.
 - 2.2 In seconda battuta il sistema dovrà verificare se il comando specificato risulta essere un comando tra quelli che, lo stesso sistema, vuole rendere disponibili.
 - 2.3 I comandi che il sistema dovrà rendere disponibili sono: scattare una foto, riprodurre musica, registrare un audio ed inviare una foto.
3. A questo punto, verificata l'autorizzazione del mittente del SMS e accertata la validità del comando, il sistema potrà eseguire l'attività legata al comando specificato nel messaggio ricevuto.
 - 3.1 Nel caso in cui, non sia riconosciuto come autorizzato il numero telefonico dal quale proviene l'SMS, oppure il comando ricevuto non rientra tra i comandi che il

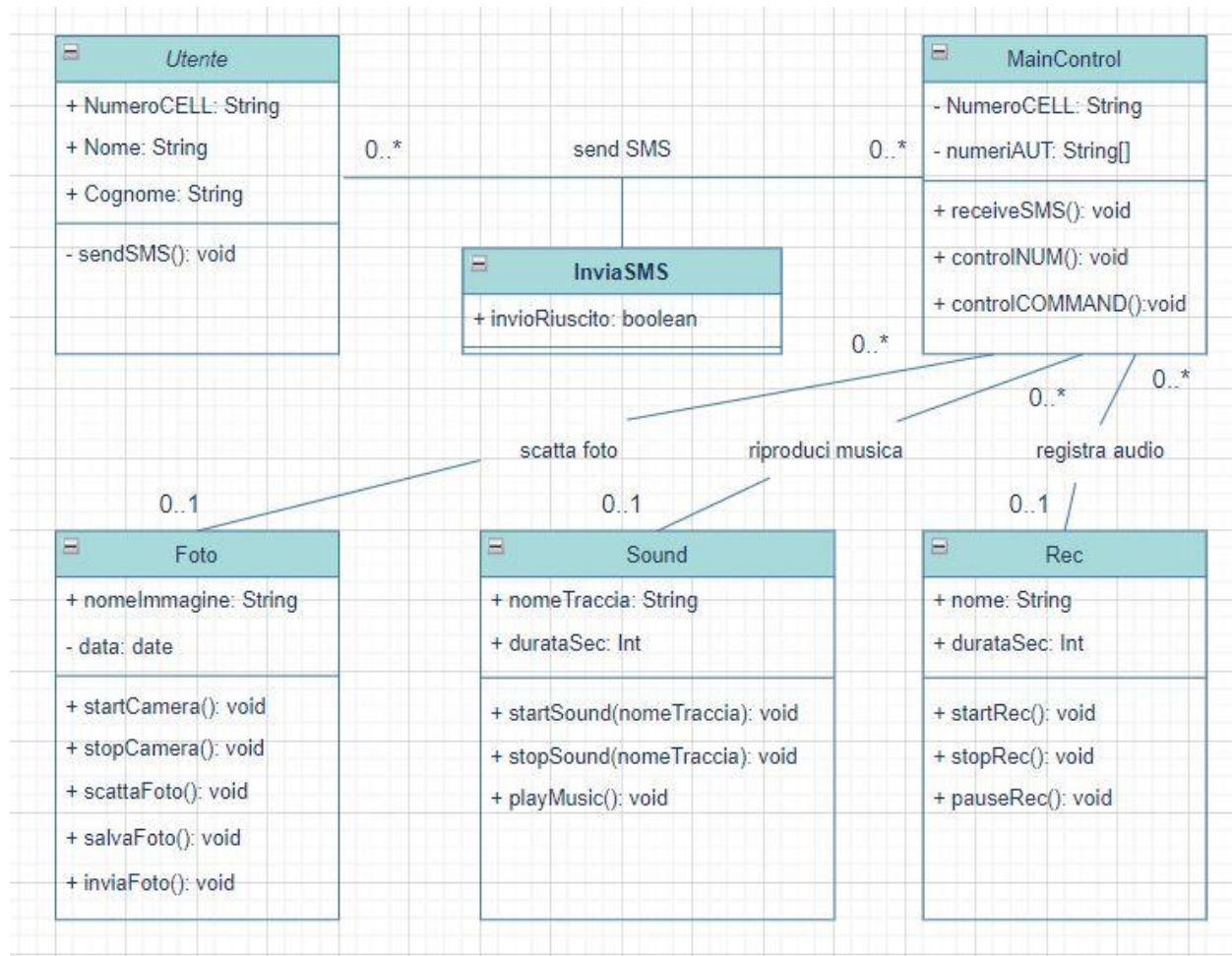
sistema rende disponibili; il sistema segnalerà all'utente rispettivamente "Numero non autorizzato" o "Comando non riconosciuto".

4. In caso contrario, il sistema in base al comando ricevuto esegue:
 - I. Comando Riproduci Musica: in questo caso il sistema dovrà, automaticamente, avviare la riproduzione di una traccia musicale. Il sistema dovrà permettere all'utente di mettere in pausa e di riprendere la riproduzione. Inoltre il sistema dovrà anche permettere di stoppare definitivamente la riproduzione.
 - II. Comando Registra audio: in questo caso il sistema dovrà, automaticamente, registrare un audio di 7 secondi che salverà nella memoria interna del dispositivo.
 - III. Comando Scatta Foto: in questo caso il sistema dovrà, automaticamente, scattare una fotografia e la dovrà salvare nella memoria interna del dispositivo.
 - IV. Comando Invia Foto: in questo caso il sistema dovrà, automaticamente, inviare un SMS contenente un'immagine al numero telefonico che ha richiesto l'esecuzione del comando.
5. Il sistema dovrà rendere possibile l'esecuzione di 1 o più comandi contemporaneamente.
6. Il sistema dovrà essere in grado di verificare se un'attività legata ad un determinato comando è **in esecuzione** e, eventualmente lo fosse, **negare nuove richieste** a quell'attività fintanto che l'esecuzione non sia terminata.

In altre parole, una volta richiesta (tramite comando) l'esecuzione di una attività, quell'attività risulta **bloccata** fintanto che è in esecuzione. Le richieste (tramite comando) di quell'attività che cadono nel lasso di tempo in cui l'attività è in esecuzione non verranno considerate. Una volta terminata quell'attività allora verrà **sbloccata** e sarà **nuovamente disponibile**.
7. Il sistema dovrà essere in grado di terminare un'attività richiesta tramite comando in maniera totalmente automatica.

Architettura del Sistema

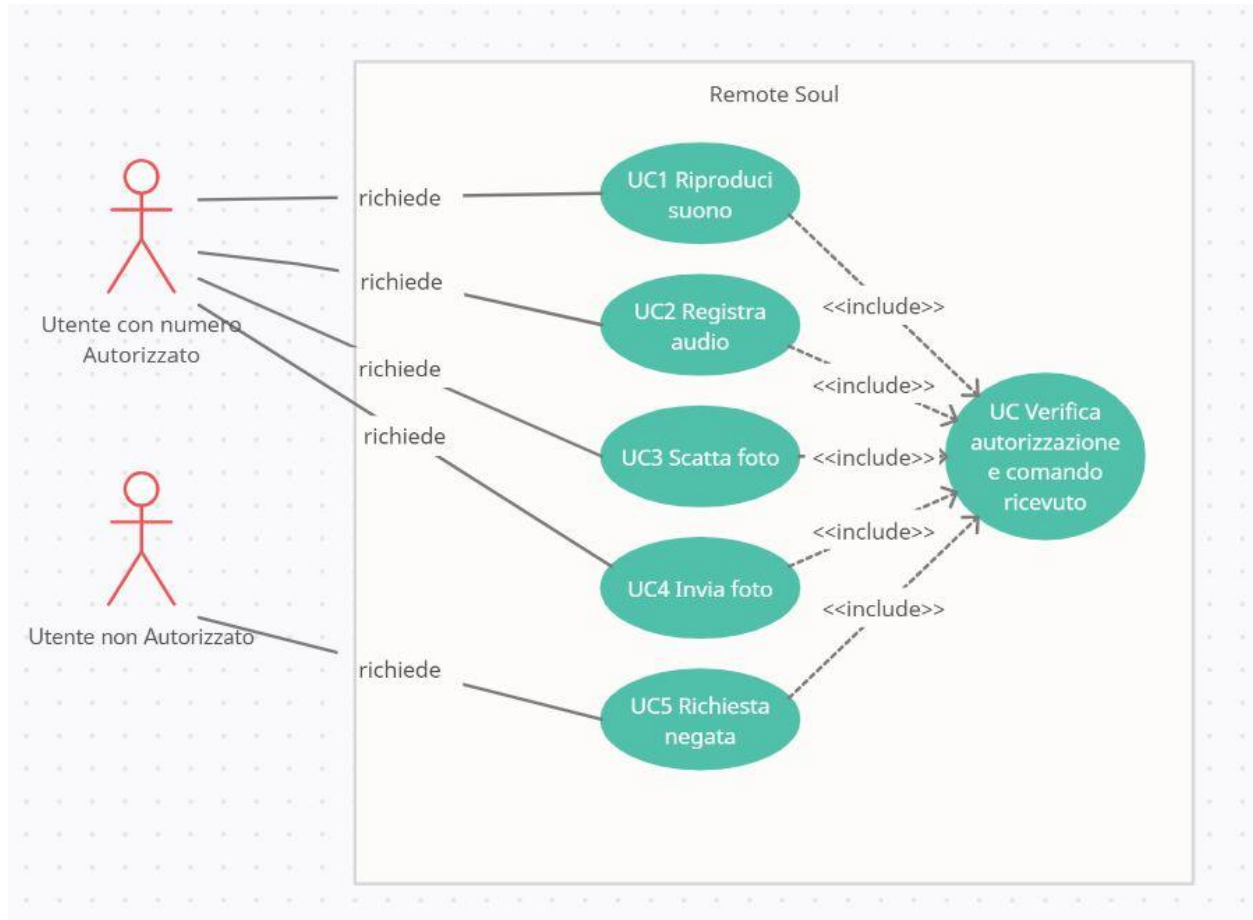
Diagramma di Classe Completo



Osservazioni:

1. La classe MainControl rappresenta il nucleo centrale che è in grado di gestire e controllare gli SMS in arrivo e di lanciare le varie attività che sono rappresentate dalle classi Foto, Sound e Rec.
2. La classe di associazione InviaSMS ha un attributo invioRiuscito di tipo boolean che stabilisce se l'invio di un SMS è andato a buon fine. E' stata utilizzata una classe di associazione e non una classe esplicita in quanto in ogni istante vi è al massimo un collegamento tra ogni coppia di oggetti Utente - MainControl.

Diagramma di Caso d'Uso



Osservazione:

Nel diagramma soprastante vengono poste in risalto le due figure che possono interagire con il sistema Remote Soul, queste sono Utente Autorizzato e Utente non Autorizzato.

Nella pagina seguente verrà riportato il caso d'uso **UC1** che coinvolge un Utente Autorizzato e il caso d'uso **UC5** che coinvolge un Utente non Autorizzato.

UC1 - Riproduci Musica

Attore primario: Utente Autorizzato

Precondizioni: Sistema in attesa di ricezione SMS con comando "Riproduci musica"

Postcondizioni: Il sistema sta riproducendo musica.

Scenario Principale:

1. L'utente Autorizzato invia SMS contenente il comando "riproduci musica".
2. Il sistema esegue 'UC - Verifica autorizzazione e comando ricevuto'.
3. Il sistema verifica se l'attività non è già in esecuzione.
4. Il sistema avvia la riproduzione della musica.

Scenario Secondario:

- a. Nel caso in cui il 'UC - Verifica autorizzazione e comando ricevuto' ritorni un esito negativo, il sistema ritorna un messaggio che illustra la motivazione del perché non è stata avviata l'attività.
- b. Nel caso in cui l'attività sia già bloccata il sistema ritornerà un messaggio di "Attività non disponibile momentaneamente".

UC5 - Richiesta negata

Attore primario: Utente non Autorizzato

Precondizioni: Sistema in attesa di ricezione SMS con un comando (uno tra quelli possibili)

Postcondizioni: Sistema non esegue nessuna attività in risposta al comando ricevuto

Scenario Principale:

1. L'Utente non Autorizzato invia un SMS contenente un comando tra quelli possibili.
2. Il sistema esegue 'UC - Verifica autorizzazione e comando ricevuto', che ritorna un messaggio "Utente non Autorizzato".
3. Il sistema non esegue alcuna attività in quanto il numero di telefono del mittente non rientra tra quelli Autorizzati.

Diagramma di Sequenza e di Collaborazione

I seguenti diagrammi sono relativi a: UC1 - Riproduci Musica

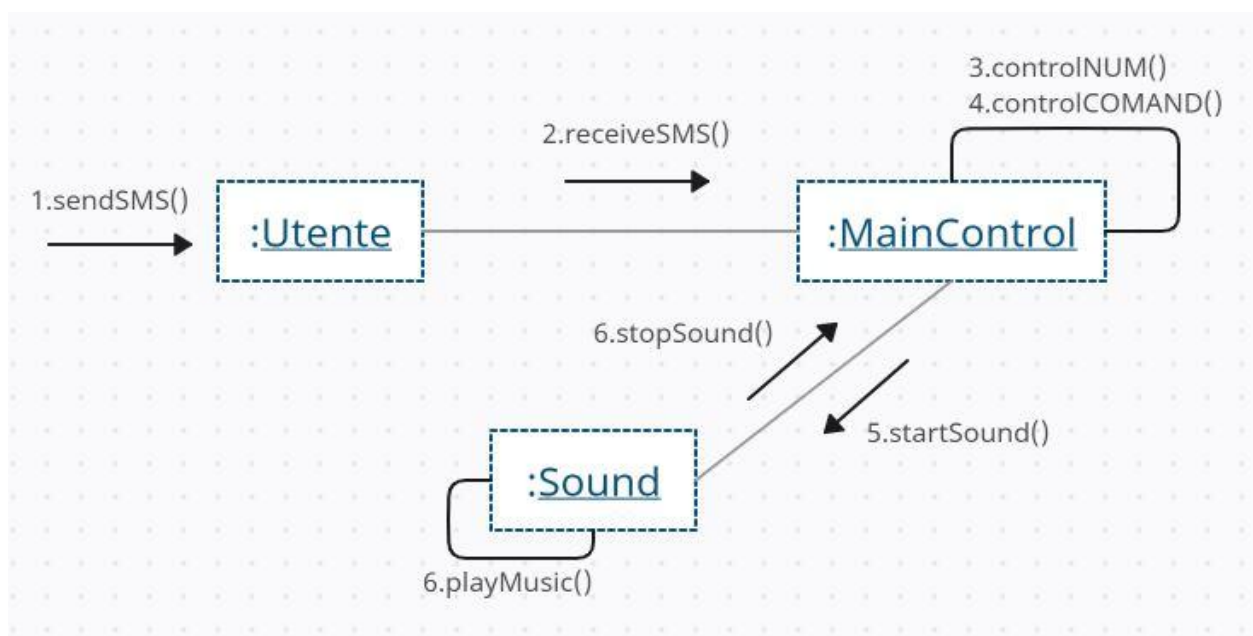
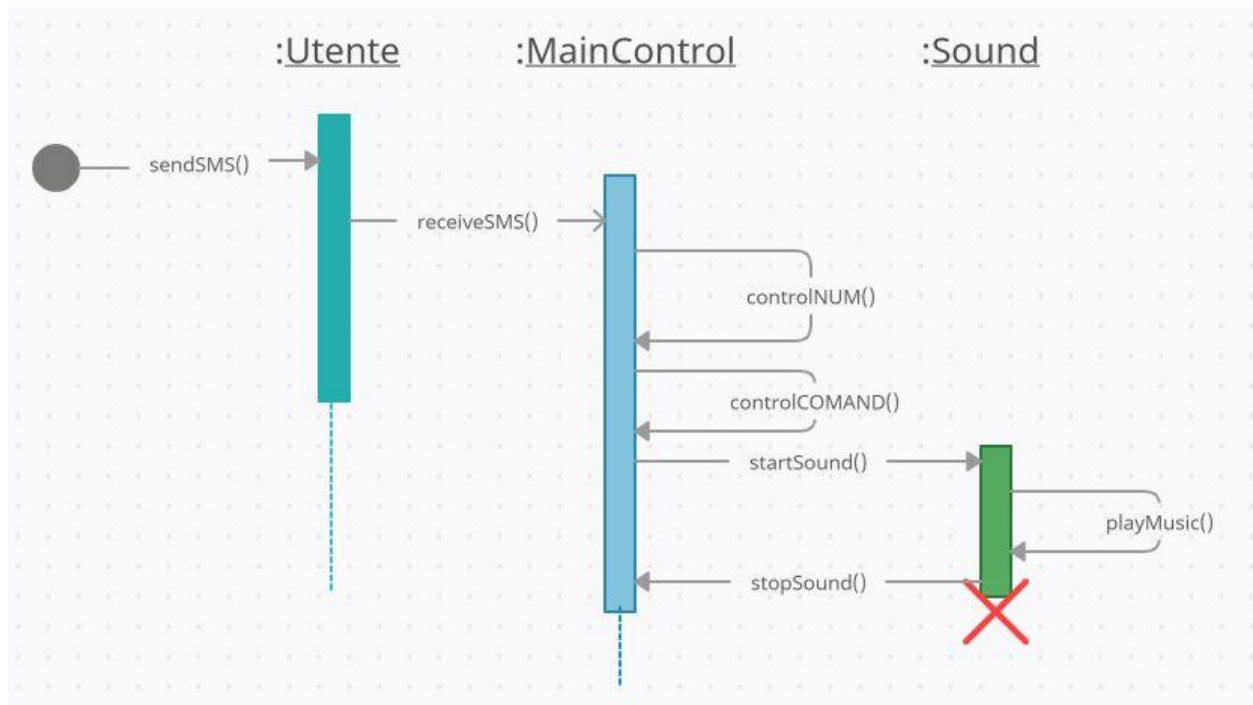
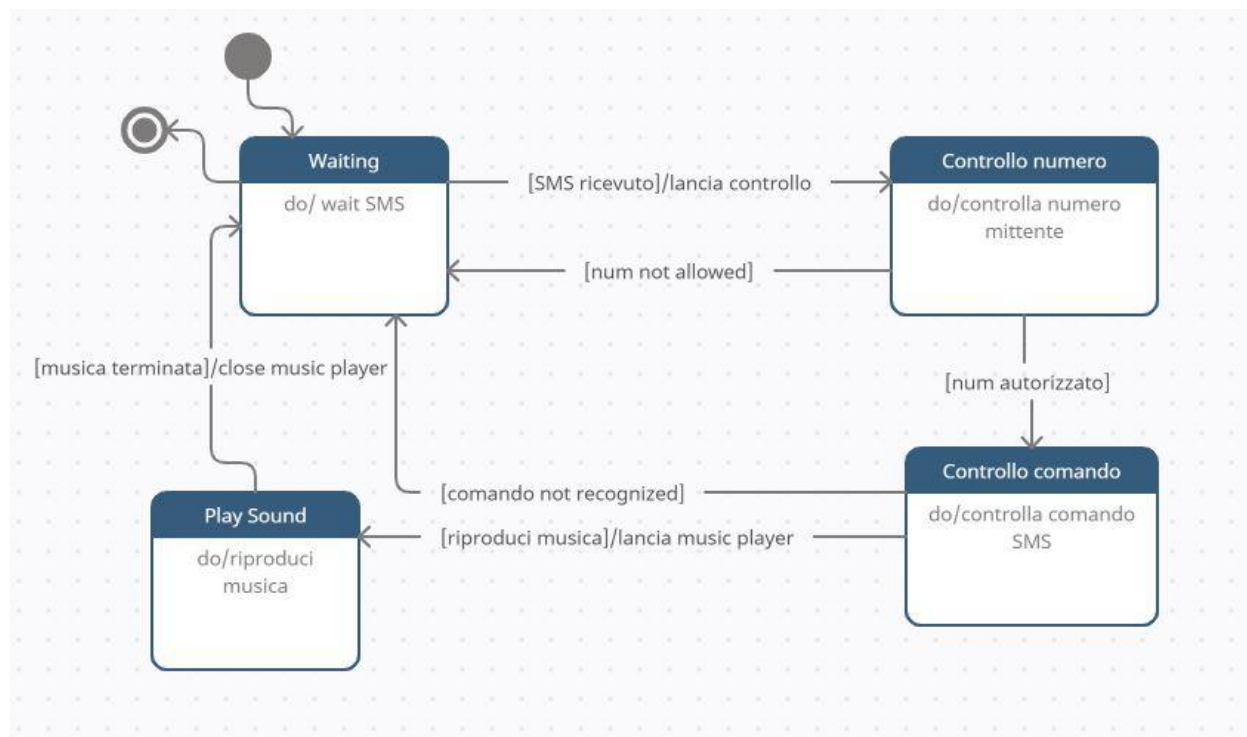


Diagramma di Stato



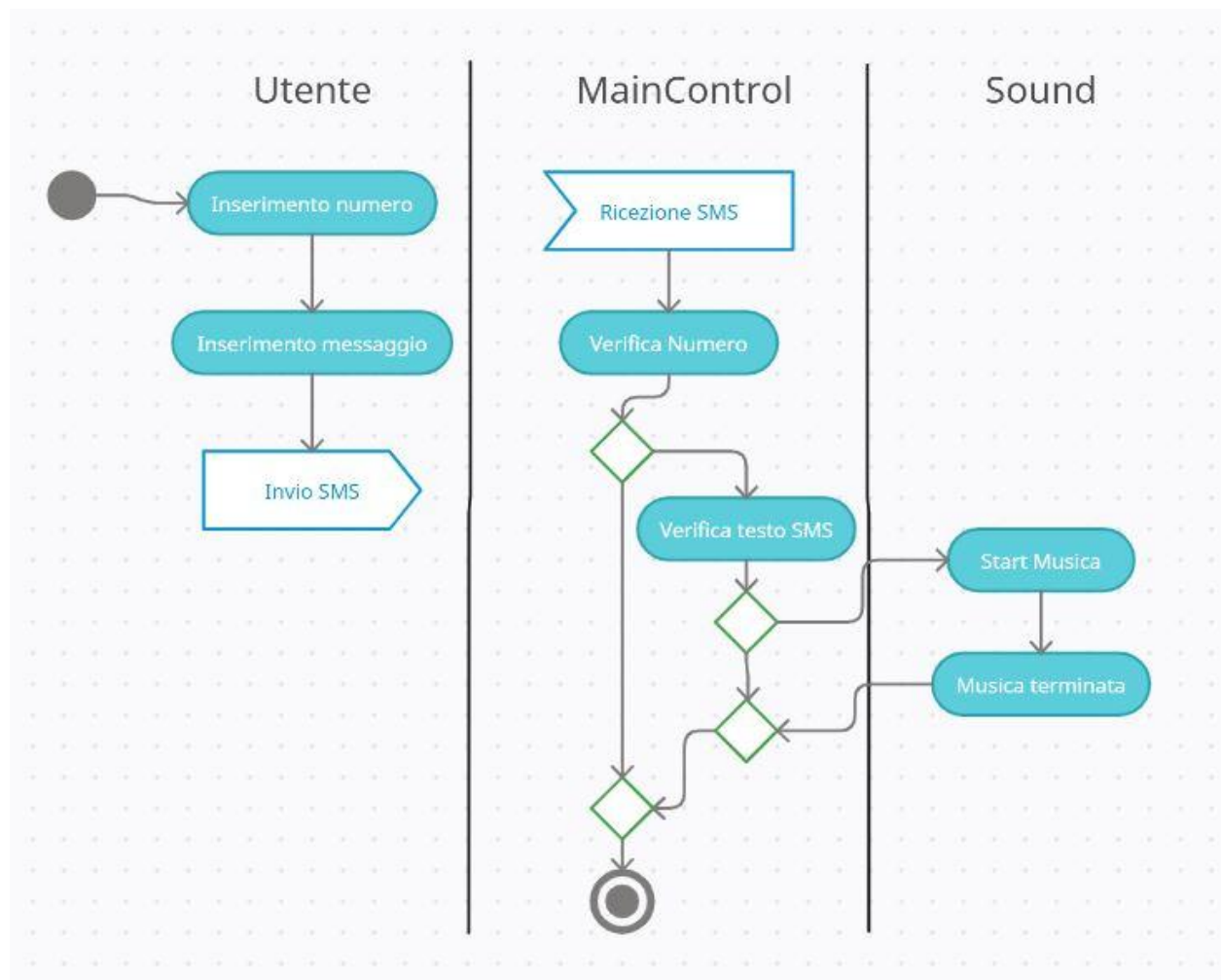
L'elemento del sistema preso in considerazione per la realizzazione del Diagramma di Stato soprastante è MainControl.

PER COMPLETEZZA:

E' stato realizzato il diagramma di stato di MainControl, quando il comando ricevuto tramite SMS è "Riprodurre musica". Per completezza è corretto sottolineare che le altre attività eseguite in risposta ad un comando ricevuto tramite SMS risultano essere speculari, e conseguentemente il diagramma di stato subirebbe appropriate modifiche.

Pertanto il diagramma di stato completo comprenderebbe anche gli altri tre stati, ScattaFoto RegistraAudio e InviaFoto, con le rispettive attività all'interno: do/ScattaFoto, do/Rec e do/InviaFoto ed infine le 3 azioni che rispettivamente riporterebbero MainControl allo stato di Waiting: [FotoScattata]closeScattaFoto, [RecTerminata]stopRec e [FotoInviata]closeInviaFoto.

Diagramma di Attività



Osservazione:

L'attività al quale fa riferimento il diagramma di attività soprastante è quella dell' invio/ricezione di un SMS contenente il comando "riproduci musica" e la conseguente riproduzione.

Documentazione implementazione

Presentazione

Remote Soul Project è stato implementato realizzando due applicazioni Android:

- I. **RemoteControl** —> applicazione utilizzata lato server per invio SMS
- II. **RemoteLabour** —> applicazione utilizzata lato client per elaborazione SMS ricevuti (nucleo progetto)

Le precedenti applicazioni sono state realizzate in **AndroidSDK**.

Breve discussione sulle scelte implementative

RemoteControl

Per RemoteControl è stato realizzato un semplice layout grafico con 2 EditText (1 per il numero di telefono e 1 per testo msg) ed un Button per l'invio del messaggio inserito.

L'applicazione tramite Toast informa l'utente se, eventualmente, un campo non è stato inserito o se i vincoli sulle EditText non sono rispettati.

In alto a destra sulla ActionBar è stata inserita un'icona che in risposta al click ripulisce i campi delle due EditText.

RemoteLabour

Per RemoteLabour è stata eliminata l'ActionBar ed è stata inserita, in basso, una BottomNavigationView con 4 icone.

Sono stati costruiti 4 Fragment con i rispettivi Layout grafici che tramite la BottomNavigationView vengono mostrati sul display.

4 Fragment:

1. Home
2. PlaySound
3. Recorder
4. Photo

La **MainActivity** si occupa principalmente di: verificare e richiedere le permission necessarie, effettuare un controllo sul numero telefonico del mittente e effettuare un controllo sulla validità dei comandi ricevuti.

La ricezione degli SMS è gestita da un **BroadcastReceiver**.

Per l'esecuzione dei comandi "Riproduci musica", "Registra audio" e "Scatta Foto" sono stati realizzati **3 servizi**: **PlaySoundService**, **CameraService** e **RecorderService** che lavorano in background e che *terminano automaticamente* una volta che l'attività del servizio è ritenuta terminata, valutazione, questa, che varia a seconda della diversa attività eseguita.

Inoltre è importante sottolineare che un servizio, se in esecuzione, risulta essere **bloccato** ovvero che per richiedere tale servizio è necessario che esso termini la propria esecuzione e che torni **disponibile**.

La vita dei servizi è monitorata tramite messaggi sul **Logcat**.

Il Fragment PlaySound presenta due Button che permettono di interagire con il **PlaySoundService**:

1. con uno è possibile stoppare e riprendere la riproduzione;
2. con l'altro viene stoppata la riproduzione e terminato il servizio.

Il Fragment Photo presenta una ImageView che mostrerà la foto scattata dal **CameraService**.

Il Fragment Recorder visualizza l'immagine di un recorder, ma non permette interazione all'utente. Il **RecorderService** tramite Toast comunica all'utente l'inizio e la fine della registrazione.

L'invio di una foto invece è gestito dalla MainActivity tramite un metodo InviaFoto() che invierà l'icona di RemoteLabour, tale metodo riscontra due problemi:

1. L'applicazione richiede all'utente di scegliere l'app di messaggistica per inviare l'MMS e richiede di premere il bottone di sending per l'invio dell'immagine.
2. L'applicazione essendo testata su emulatori senza un vero numero telefonico e un gestore telefonico presenta problemi di configurazione APN, per cui questa parte di codice non è stata testata approfonditamente in questi termini.

Test Funzionali BlackBox

Per quanto riguarda **RemoteLabour**: l'applicazione è priva, quasi totalmente, di interazione con l'utente; non ci sono form o campi da riempire, in ogni modo l'applicazione non riceve un input che possa essere oggetto di test funzionali.

Per quanto riguarda **RemoteControl**: l'applicazione presenta due EditText che possono essere riempite dall'utente con il numero di telefono del destinatario e il testo del messaggio.

Copertura Classi di equivalenza e Analisi Valori Estremi

Input	Classe Valida	Classe NON valida	Classe NON valida
TEL	len(TEL) = 10	len(TEL) < 10	len(TEL) > 10
TEL Valori Estremi	len(TEL) = 10	len(TEL) = 9	len(TEL) = 11
Testo	1<=len(Txt)<=80	len(Txt) < 1	len(Txt) > 80
Testo Valori Estremi	len(Txt) = 1 len(Txt) = 80	len(Txt) = 0	len(Txt) = 81

Casi di Test Funzionali Minimali

TEL	TESTO	Validità
5555215554	prova1prova2345	VALIDO
5555215554	-	NON VALIDO TESTO < 1
5555215554	prova1prova2.. (string 90 car.)	NON VALIDO TESTO > 80
555521	prova1prova2345	NON VALIDO TEL < 10
5555215554693	prova1prova2345	NON VALIDO TEL > 10

Casi di Test Funzionali addizionali generati da Analisi Valori Estremi

TEL	TESTO	Validità
5555215554	p	VALIDO
5555215554	prova1prova2.. (string 80 car.)	VALIDO
5555215554	-	NON VALIDO TESTO = 0
5555215554	prova1prova2.. (string 81 car.)	NON VALIDO TESTO = 81
555521555	prova1prova2345	NON VALIDO TEL = 9
55552155545	prova1prova2345	NON VALIDO TEL = 11

RemoteControl risponde secondo le aspettative sopra indicate.