



Documento di progettazione



IO, LEGO

Autori

- Volpe Davide 862989
- Melissari Luca 865700
- Jasari Besar 862699
- Marcolin Giulio 862611
- Bertin Eleonora 865370

Sommario

Introduzione	3
Scopo del documento	3
Struttura del documento	3
Glossario	3
Modello e struttura del sistema	4
Modello di controllo	5
Diagramma delle classi	6
Diagramma delle attività	8
Attivazione App e connessione Bluetooth	8
Selezione cella e avvio programma	9
Attesa e terminazione programma	10
Interfaccia grafica	11

1. Introduzione

1.1. Scopo del documento

Lo scopo di questo documento è illustrare tutti gli aspetti che hanno a che fare con l'implementazione dell'applicazione Android da noi creata, mostrando in modo approfondito i dettagli necessari alla corretta strutturazione del sistema.

1.2. Struttura del documento

Il documento è suddiviso nelle seguenti parti:

- **Glossario:** è un elenco di termini di uso tecnico utilizzati nel documento, utile a facilitare la comprensione del testo.
- **Struttura del sistema:** in questa sezione mostreremo come è strutturato il sistema, evidenziando la suddivisione in sottosistemi e facendo vedere come essi comunicano fra loro.
- **Modello di controllo:** contiene la descrizione relativa al tipo di controllo che serve a disciplinare le relazioni tra i sottosistemi presenti.
- **Diagramma delle classi:** Sezione finalizzata a descrivere le entità che costituiscono il sistema, le loro relazioni e le varie operazioni effettuabili.
- **Diagramma delle attività:** Qui vengono definite le attività che è necessario compiere in modo da poter mettere in atto le funzionalità offerte dall'applicazione.
- **Diagramma di sequenza:** qui verranno mostrate, in relazione alle varie attività descritte, le chiamate a funzione.
- **Interfaccia grafica:** questa sezione illustrerà i prototipi dell'interfaccia grafica destinata, mostrandone gli schemi più importanti.

1.3. Glossario

Lego EV3 Mindstorms:

Robot commercializzato dall'azienda LEGO, dotato di motori e sensori. Può essere comandato tramite un dispositivo chiamato EV3.

Feedback:

Giudizio inviato da un utente allo sviluppatore contenente eventuali segnalazioni di errori.

Diagramma UML:

Il linguaggio UML è un linguaggio di modellazione e specifica. Viene usato, in questo contesto, per descrivere il dominio applicativo di un sistema software e/o il comportamento e la struttura del sistema stesso.

Android:

Sistema operativo per dispositivi mobili sviluppato da Google, basato su kernel Linux.

Java:

È un linguaggio di programmazione ad alto livello, orientato agli oggetti.

Bluetooth:

È uno standard tecnico-industriale di trasmissione dati per reti personali senza fili. Mette in comunicazione dispositivi diversi entro una decina di metri.

Android Studio:

È un ambiente di sviluppo integrato (IDE) per lo sviluppo di applicazioni per la piattaforma Android.

Firmware:

È un programma integrato direttamente in un componente elettronico programmato. In questo contesto viene utilizzato un firmware Java chiamato LeJos per il controllo del dispositivo EV3.

2. Modello e struttura del sistema

La struttura del sistema sarà a client-server, in quanto facciamo una distinzione di due macchine, ovvero la app android usata attraverso il dispositivo android dell'utente, che sarà gestita attraverso classi specifiche ne permettono il funzionamento e l'invio dei dati all'altra macchina, ovvero il robot Lego EV3 Mindstorm. Quest'ultimo riceverà i dati dall'applicazione e attraverso le classi create con la libreria LeJos e memorizzate sul Firmware, procederà nell'azione di ricerca del colore all'interno della matrice. Saranno due macchine che lavorano indipendentemente attraverso librerie e dati non condivisi. Quindi indicheremo come **client** l'utente che attraverso la app android comunicherà con quello che definiamo **server**, ovvero il robot Lego EV3 Mindstorm. In questo modo il robot potrà essere collegato ad altri differenti dispositivi android che avranno l'applicazione installata e potranno comunicare con il robot senza alcun tipo di vincolo se non quelli di sistema per supportare l'applicazione.

I dati prodotti dall'utente e dal robot verranno inviati attraverso una comunicazione tramite bluetooth. Saranno di piccole dimensione, infatti conterranno la posizione delle celle con i colori che si vorranno controllare e il feedback del robot una volta effettuato il controllo del colore attraverso i suoi sensori.

3. Modello di controllo

Utilizzeremo il modello di controllo ad **eventi** di tipo Interrupt-driven, in quanto le due macchine si comportano diversamente in base agli eventi che verranno prodotti dall'utente, ed inoltre il robot EV3 Mindstorm dovrà rispondere in maniera real-time ad eventi che accadranno durante la sua funzione, come lo spostamento al rilevamento degli ostacoli o la risposta al rilevamento del colore. I tipi di interruzioni saranno :

Per app android:

Invio dei dati → il sistema rimane in attesa del feedback del robot senza permettere altre azioni all'utente.

Ricezione dei dati → il sistema rappresenta l'esito nella matrice attraverso due simboli per indicare se il colore verificato è giusto o no.

Ricezione di fine percorso → il sistema presenterà una schermata in cui mostrerà le statistiche di colori indovinati e ri-permetterà l'utilizzo dell'app.

Per Lego EV3 Mindstorm:

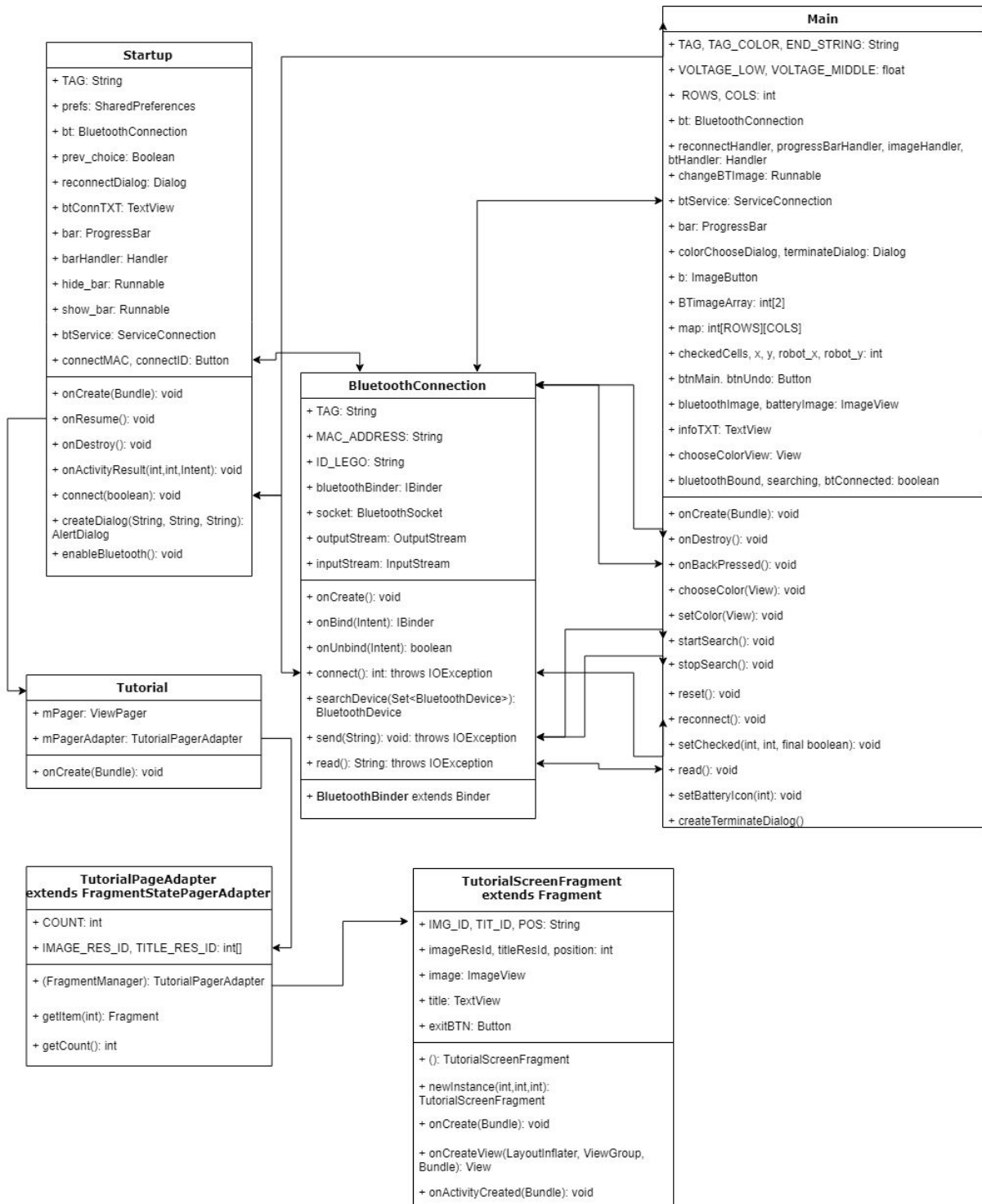
Ricezione del colore → una volta raggiunto la cella di riferimento si genera un'interruzione che avvia il controllo del colore.

Invio del feedback → una volta controllato il colore invia il feedback e riprenderà la funzione di raggiungimento del colore.

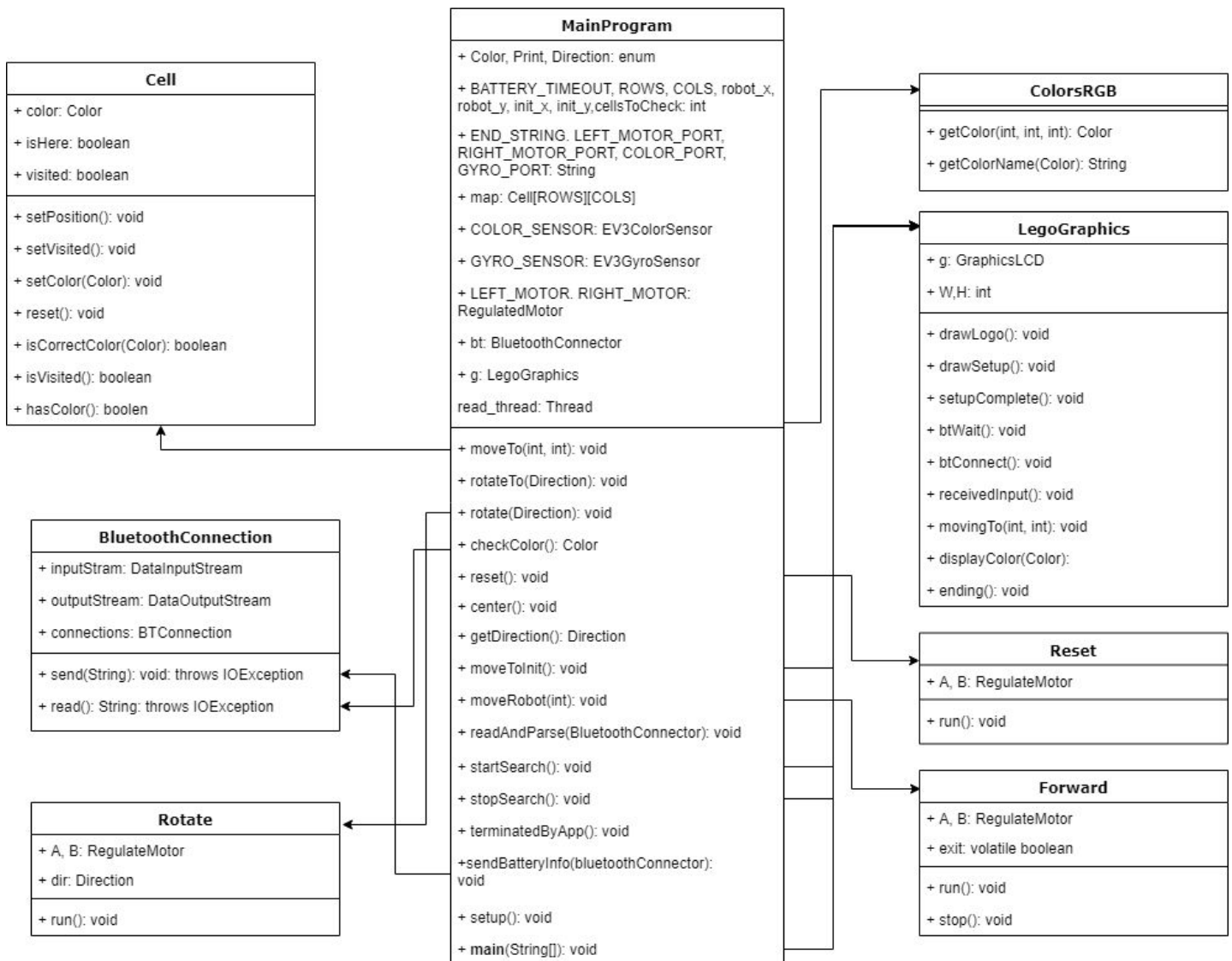
Fine del programma → una volta letto l'ultimo colore invierà una stringa di avviso che farà terminare il programma.

4. Diagramma delle classi

Il seguente diagramma mostra la definizione delle classi dell'applicazione Android e le loro interazioni.

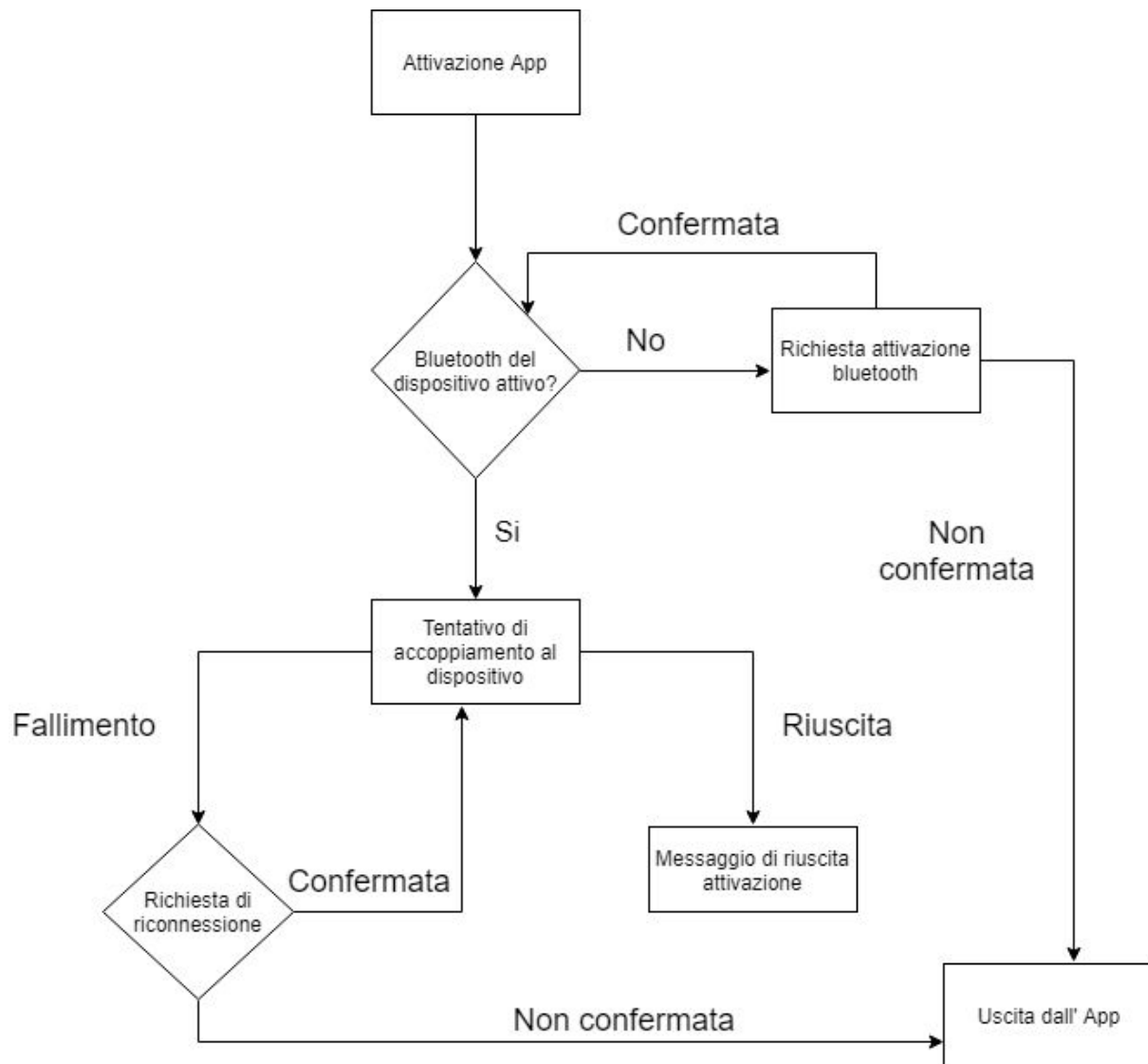


Il seguente diagramma mostra invece la definizione delle classi dell'EV3 e le loro interazioni.

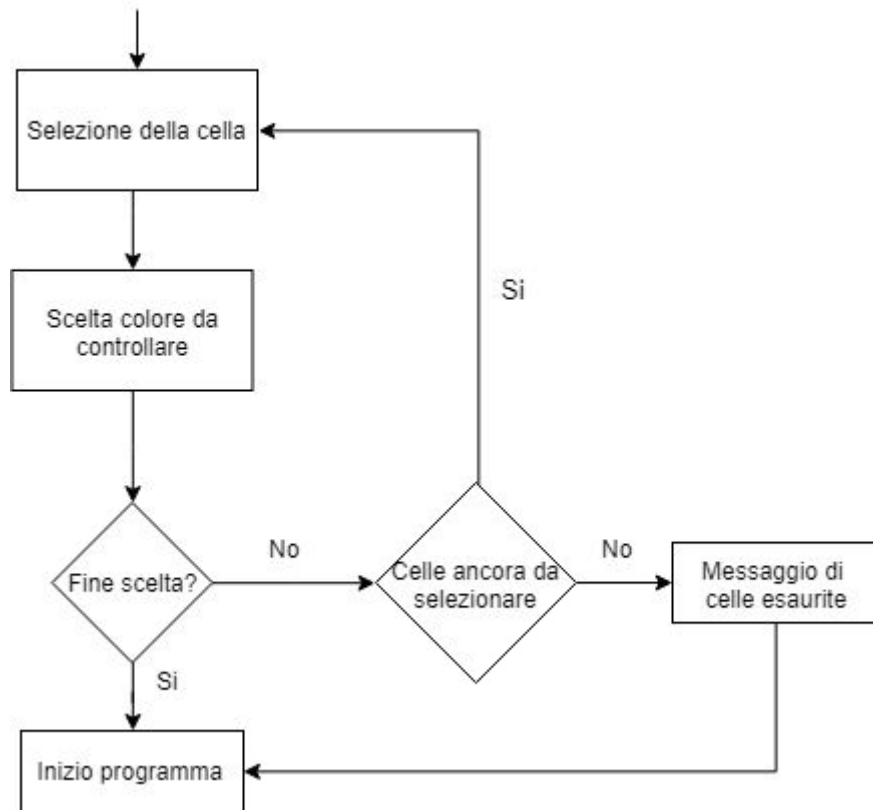


5. Diagramma delle attività

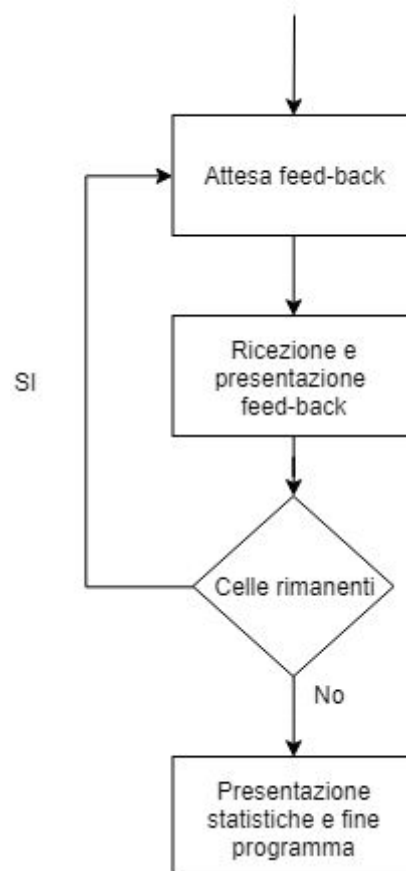
5.1. Attivazione App e connessione Bluetooth



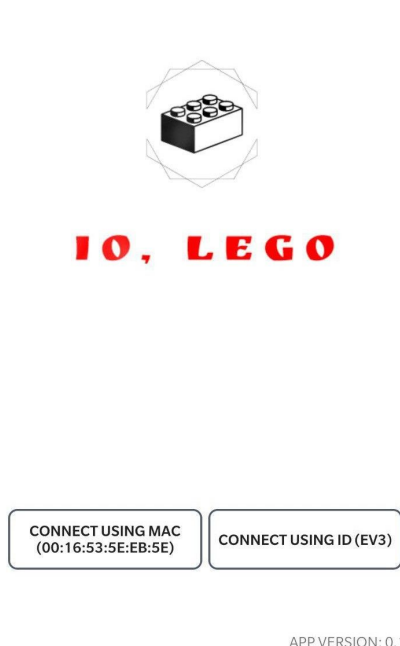
5.2. Selezione cella e avvio programma



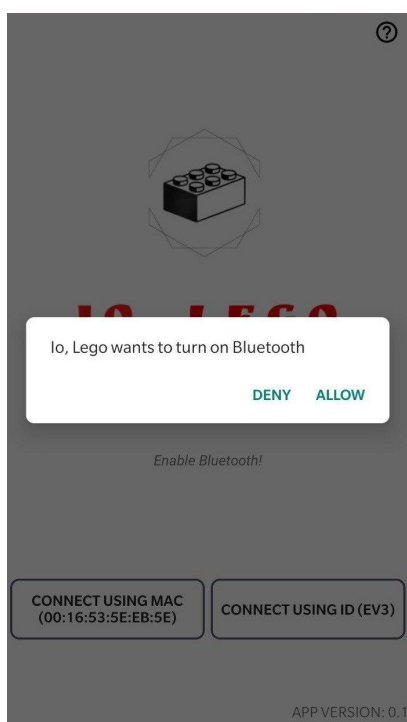
5.3. Attesa e terminazione programma



6. Interfaccia grafica

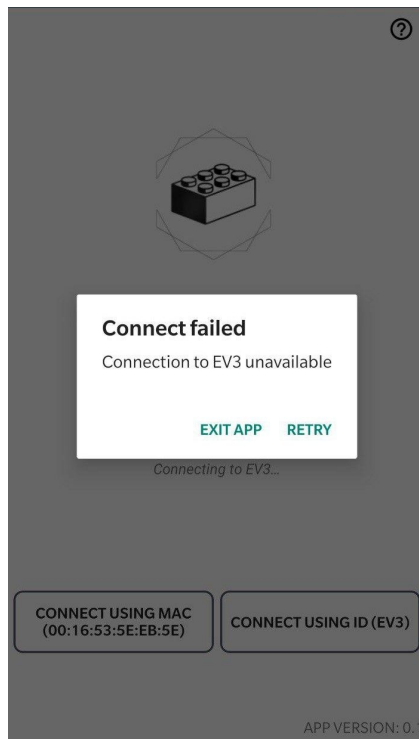
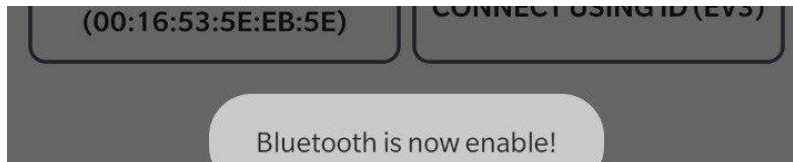


Primo avvio: Al primo avvio dell'applicazione comparirà la schermata principale nella quale si cerca di stabilire la connessione tra applicazione ed EV3.

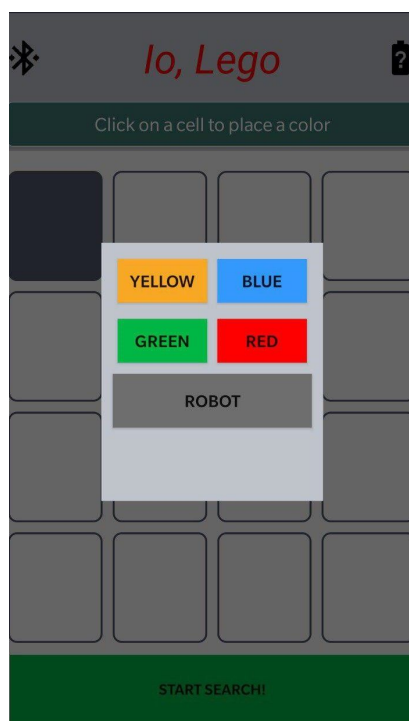


Richiesta attivazione Bluetooth: Il Bluetooth del proprio dispositivo risulta spento, comparirà quindi un pop-up nel quale è richiesto il permesso per l'attivazione del Bluetooth in modo da poter stabilire la connessione app-Robot.

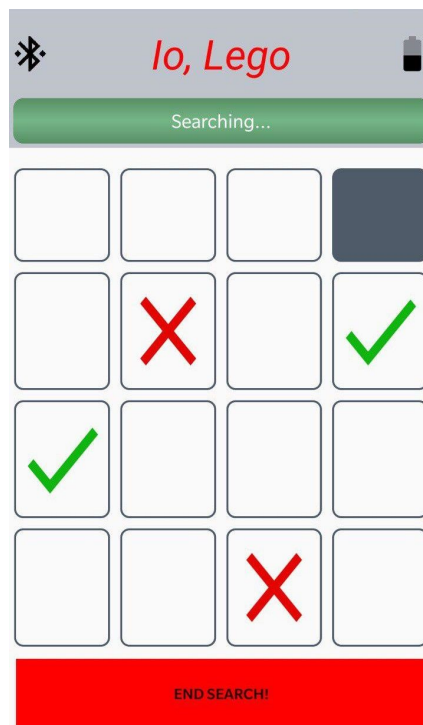
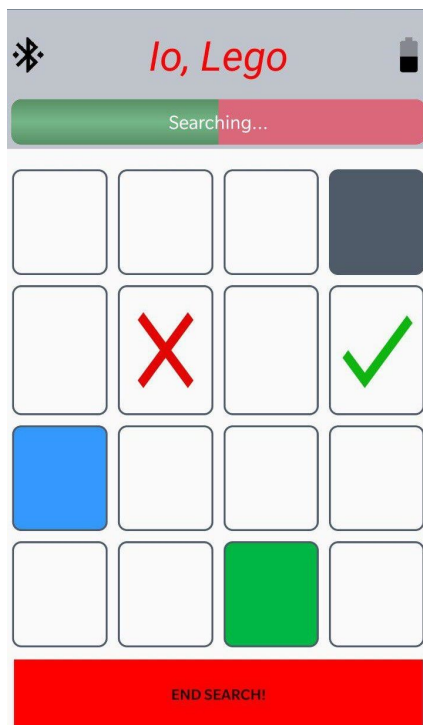
Bluetooth abilitato: Quando il Bluetooth risulta disponibile viene mostrata la notifica dell'avvenuta abilitazione.



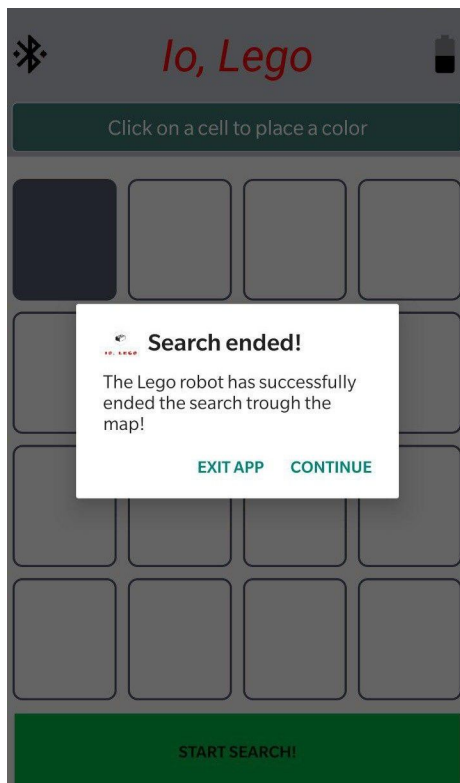
Connessione fallita: Pop-Up che viene mostrato quando la connessione è fallita e che dà la possibilità di tentare di nuovo di ristabilire la connessione oppure di uscire.



Scelta colore: Dietro vi è la schermata principale dell'applicazione che mostra la matrice di gioco. Una volta che si preme su uno dei quadrati della matrice compare il pop-up che dà la possibilità di scegliere uno dei quattro colori disponibili. Una volta scelto il colore di tutte le celle che si desidera cercare, in basso ci sarà il tasto START SEARCH che farà partire la ricerca dell'oggetto nella matrice.



Colore trovato-non trovato: Quando il Robot trova un colore, nella matrice virtuale compare una ✓ se il colore trovato sarà uguale al colore scelto, una X altrimenti.



Fine ricerca: Quando il robot termina la ricerca, viene mostrato un popup che permette di uscire dall'applicazione oppure di fare un'altra ricerca. L'applicazione viene resettata automaticamente, togliendo tutti i precedenti colori e settando la posizione del robot iniziale.