



# Documento di Analisi e Specifica dei requisiti



**IO, LEGO**

## Autori

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

# Sommario

<b>1.</b>	<b>Introduzione</b>	<b>3</b>
1.1.	Scopo del documento	3
1.2.	Descrizione del documento	3
1.3.	Funzionalità di progetto	3
<b>2.</b>	<b>Glossario</b>	<b>4</b>
<b>3.</b>	<b>Modelli del sistema</b>	<b>5</b>
3.1.	Montaggio	6
3.2.	Avvio Sistema	6
3.3.	Tutorial	7
3.4.	Schermata Connessione Bluetooth	8
3.5.	Fase preparatoria	9
3.6.	Inizio interazione robot	9
3.7.	Fase finale	12
<b>4.</b>	<b>Definizione dei requisiti funzionali</b>	<b>13</b>
<b>5.</b>	<b>Definizione dei requisiti non funzionali</b>	<b>15</b>
5.1.	Requisiti di prodotto	15
5.2.	Requisiti di processo	16
5.3.	Requisiti esterni	16
<b>6.</b>	<b>Evoluzione del sistema</b>	<b>17</b>
<b>7.</b>	<b>Specifica dei requisiti</b>	<b>17</b>
<b>8.</b>	<b>Appendici</b>	<b>19</b>
8.1.	Requisiti hardware	19

# 1. Introduzione

## 1.1. Scopo del documento

Questo documento ha lo scopo di spiegare in modo dettagliato il progetto *Io Lego*.

## 1.2. Descrizione del documento

Il documento è diviso secondo il seguente schema:

- **Glossario:** elenco contenente le definizioni di tutti i termini tecnici usati nel documento.
- **Modelli del sistema:** definisce la struttura dell'applicazione e l'interazione col robot Lego EV3 Mindstorms tramite diagrammi UML.
- **Definizione dei requisiti funzionali:** servizi offerti all'utente dall'ecosistema applicazione/robot.
- **Definizione dei requisiti non funzionali:** vincoli sui servizi e sulle operazioni supportate dal sistema.
- **Evoluzione del sistema:** descrizione di eventuali funzionalità aggiuntive e futuri sviluppo dell'applicazione.
- **Specifica dei requisiti:** descrizione maggiormente dettagliata dei requisiti funzionali definiti nella sezione 4.
- **Appendici:** requisiti minimi dei dispositivi per l'utilizzo dell'applicazione.

## 1.3. Funzionalità di progetto

Il progetto ha scopo puramente ludico.

Consiste nella realizzazione di un'applicazione in cui l'utente inizialmente sceglie il colore di un numero arbitrario di celle in una griglia virtuale e la posizione iniziale del Lego EV3 (posizione da cui poi il Robot comincerà la ricerca). Posiziona poi i quadrati colorati e il robot nella griglia cartacea. Tutti questi materiali sono forniti in dotazione.

Alla pressione di un pulsante nell'interfaccia utente, il robot inizierà a muoversi verso le posizioni della griglia indicate.

Una volta raggiunto un quadrato colorato, effettuerà il controllo del colore.

Un feedback sarà mostrato all'utente tramite l'interfaccia dell'applicazione.

Questo sistema può essere usato in diversi scopi educativi e ludici, e sarà l'utente a inventare e creare diversi scopi a piacimento, per esempio l'utilizzo educativo nel far apprendere i colori principali ai bambini e nelle iterazioni con un robot. Oppure in un gioco di memorizzazione, dove un utente cerca di disporre diversi colori e un secondo cercherà di ricordarli e avvierà il robot per

la verifica. Questi sono alcuni esempi, utilizzando l'immaginazione se ne possono ottenere molti altri

## 2. 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:

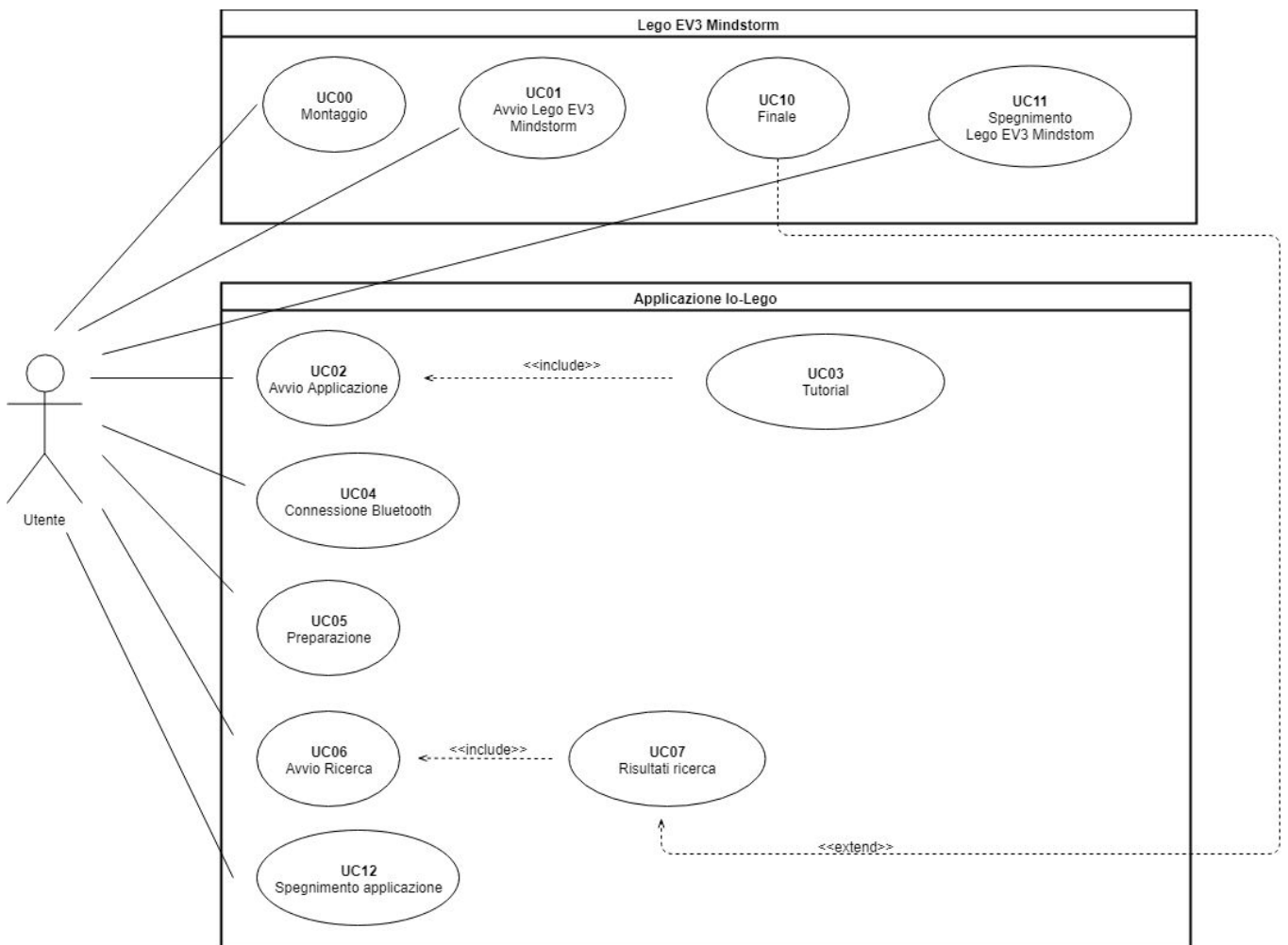
È un programma integrato direttamente in un componente elettronico programmato. In questo contesto viene utilizzato un firmware Java chiamato LeJos per il controllo del dispositivo EV3.

GitHub:

Servizio di hosting per progetti software. Permette, tramite il software GIT, il controllo versione del progetto.

### 3. Modelli del sistema

Il seguente diagramma UML descrive genericamente tutte le possibili azioni effettuabili nel sistema.



Di seguito verranno mostrati i casi d'uso del sistema secondo la seguente tabella:

Codice	Codice che identifica un caso d'uso
Nome caso d'uso	Il nome del caso d'uso
Obiettivo	Lo scopo che ha il caso d'uso per l'attore
Attori	Chi interagisce con l'applicazione
Pre-Condizioni	Condizioni necessarie al caso d'uso
Trigger	Evento che attiva il caso d'uso
Descrizione	Descrizione del caso d'uso
Alternative	Casi d'uso alternativi che possono essere usati per soddisfare lo stesso obiettivo
Post-Condizioni	Condizioni soddisfatte alla fine del caso d'uso

### 3.1. Avvio Sistema

Codice	UC01
Nome caso d'uso	Avvio Lego EV3 Mindstorms
Obiettivo	Accendere il Lego EV3
Attori	Utente
Pre-Condizioni	Avere il robot funzionante
Trigger	Pressione sul pulsante di avvio del Lego EV3 Mindstorms
Descrizione	L'utente attraverso il pulsante di avvio accenderà il Lego EV3 Mindstorms
Alternative	-
Post-Condizioni	Lego EV3 Mindstorms sarà attivato. Si passa a UC02

Codice	UC02
Nome caso d'uso	Avvio applicazione
Obiettivo	Avviare l'applicazione sul dispositivo Android personale
Attori	Utente
Pre-Condizioni	UC01 L'applicazione deve essere installata dall'App Store
Trigger	'Tap' sull'icona dell'applicazione
Descrizione	L'utente cerca l'applicazione nel menù del dispositivo e clicca l'icona per avviarla
Alternative	-
Post-Condizioni	L'applicazione sarà avviata Se è il primo avvio si passa a UC03 altrimenti si passa a UC04

### 3.2. Tutorial

Codice	UC03
Nome caso d'uso	Tutorial
Obiettivo	Istruire l'utente sul corretto utilizzo dell'applicazione
Attori	Utente
Pre-Condizioni	UC02
Trigger	Primo avvio dell'applicazione
Descrizione	Schermata istruttiva sull'utilizzo dell'applicazione mostrata solamente al primo avvio
Alternative	Saltare il tutorial
Post-Condizioni	L'utente sa come utilizzare l'applicazione intuitivamente Si passa a UC04



### 3.3. Schermata Connessione Bluetooth

Codice	UC04
Nome caso d'uso	Connessione Bluetooth
Obiettivo	Connettere il dispositivo mobile al Robot
Attori	Utente
Pre-Condizioni	UC01 e UC02
Trigger	UC02
Descrizione	Aspettare l'accensione della luce verde del dispositivo EV3 che ne segna l'avvio del programma. L'applicazione tenta la connessione Bluetooth con il Lego EV3. Se dovesse fallire (perché ad esempio il Robot non è ancora completamente acceso) il sistema avviserà l'utente, che potrà decidere di ritentare la connessione tramite apposito pulsante.
Alternative	L'utente si collega manualmente al robot tramite le impostazioni bluetooth di sistema del proprio dispositivo
Post-Condizioni	Si passa a UC05

### 3.4. Fase preparatoria

Codice	UC05
Nome caso d'uso	Preparazione
Obiettivo	Scegliere le posizioni virtuali (tramite applicazione) e fisiche (sulla griglia cartacea) dei cartoncini colorati e del Lego EV3
Attori	Utente
Pre-Condizioni	UC04
Trigger	-
Descrizione	<p>L'utente sceglie dove posizionare i propri colori sulla griglia fisica coerentemente con la scelta fatta tramite applicazione. I colori possono non combaciare, in quanto l'obiettivo del Robot è proprio quello di verificarne la correttezza.</p> <p>Infine bisogna scegliere in entrambi gli schemi (fisico e virtuale) la cella iniziale del robot</p>
Alternative	-
Post-Condizioni	<p>I quadratini colorati e il Robot Lego sono posizionati correttamente sulla griglia fisica secondo la disposizione scelta nella griglia virtuale.</p> <p>Si passa a UC06</p>

### 3.5. Inizio interazione robot

Codice	UC06
Nome caso d'uso	Avvio ricerca
Obiettivo	Far iniziare la ricerca dei quadratini e controllare il colore di ciascuno, comunicandolo all'applicazione.
Attori	Applicazione, Robot
Pre-Condizioni	UC05
Trigger	'Tap' sul bottone di inizio ricerca
Descrizione	Il robot riceve dall'applicazione i dati delle posizioni dei quadratini e si muove verso tali posizioni. Raggiunta la posizione controlla il colore dell'oggetto presente della cella e la comunica all'applicazione.
Alternative	-
Post-Condizioni	Continua la ricerca finché ci sono colori da cercare Al termine passa a UC07

Codice	UC07
Nome caso d'uso	Risultati ricerca
Obiettivo	Comunicare all'utente i risultati della ricerca
Attori	Applicazione
Pre-Condizioni	UC06
Trigger	-
Descrizione	L'applicazione mostra un messaggio in base ai risultati della ricerca e quindi del controllo del colore.
Alternative	-
Post-Condizioni	Ho i risultati. Si passa a UC08

### 3.6. Fase finale

Codice	UC8
Nome caso d'uso	Finale
Obiettivo	Far tornare il Robot alla posizione iniziale
Attori	Robot
Pre-Condizioni	Assenza di oggetti da ricercare
Trigger	-
Descrizione	Se durante la fase di Avvio ricerca non risulta presente nessun oggetto da analizzare, il Robot ritornerà automaticamente alla posizione iniziale
Alternative	-
Post-Condizioni	Si può ritornare a UC05 se si desidera ricominciare la ricerca dei blocchi oppure UC09 e UC10

Codice	UC9
Nome caso d'uso	Spegnimento Lego EV3 Mindstorms
Obiettivo	Spegnere il Robot
Attori	Utente
Pre-Condizioni	-
Trigger	Pressione sul pulsante di spegnimento del Robot
Descrizione	Quando si desidera terminare la sessione di gioco si deve spegnere il Robot.
Alternative	-
Post-Condizioni	Il robot è spento

Codice	UC10
Nome caso d'uso	Chiusura applicazione
Obiettivo	Chiudere l'applicazione
Attori	Utente
Pre-Condizioni	-
Trigger	Premere il tasto ESC dalla schermata principale dell'applicazione
Descrizione	Per chiudere l'applicazione si preme il tasto ESC con relativa conferma.
Alternative	-
Post-Condizioni	L'applicazione sarà chiusa

## 4. Definizione dei requisiti funzionali

Di seguito verranno mostrati i requisiti funzionali del sistema secondo la seguente tabella:

Codice	Codice requisito funzionale
Nome funzione	Nome del requisito funzionale
Definizione	Descrizione e utilizzo del requisito funzionale
Motivazione	Lo scopo di tale requisito
Influenza	Su quale azione influisce
Specifica	Codice della specifica del requisito funzionale

Codice	REQ-F0
Nome funzione	Connessione Bluetooth automatica
Definizione	Una volta aperta l'applicazione apparirà la schermata richiedente il permesso alla connessione Bluetooth
Motivazione	Permettere il collegamento tra telefono e Lego EV3
Influenza	-
Specifica	SRF-00

Codice	REQ-F1
Nome funzione	Scelta della cella e del colore da controllare
Definizione	L'utente dovrà selezionare una cella sulla griglia della schermata e inserire un colore messo a disposizione dall'applicazione
Motivazione	Soddisfare UC05
Influenza	Il percorso del Robot sarà calcolato
Specifica	SRF-01

Codice	REQ-F2
Nome funzione	Calcolo del percorso
Definizione	Un algoritmo specifico procederà al calcolo del percorso dal punto di partenza alla cella selezionata
Motivazione	Permette di soddisfare UC06 e UC08
Influenza	Svolgimento della funzione principale del Robot
Specifica	SRF-02

Codice	REQ-F3
Nome funzione	Controllo del colore
Definizione	Una volta arrivato alla cella selezionata, il robot provvederà alla verifica del colore tramite il suo apposito sensore
Motivazione	Soddisfare UC06
Influenza	-
Specifica	SRF-03

Codice	REQ-F4
Nome funzione	Feedback del robot
Definizione	Una volta verificato il colore della cella, il Robot invia un messaggio che stabilisce se il colore trovato corrisponde a quello cercato.
Motivazione	Completare obiettivo del progetto
Influenza	Si attiva la scelta dello spostamento dell'oggetto
Specifica	SRF-04

## 5. Definizione dei requisiti non funzionali

### 5.1. Requisiti di prodotto

- **RNF-01:** L'applicazione non deve essere troppo lenta nel computare le funzionalità richieste.
- **RNF-02:** L'applicazione deve adattarsi a tutti i dispositivi che possono supportare l'applicazione in base alla versione del loro sistema operativo. Ne garantiamo la copertura a partire dalla versione Android Lollipop.
- **RNF-03:** Il Robot deve essere perfettamente funzionante. Deve essere esente da difetti di fabbrica oppure non subire danni che compromettano le sue parti meccaniche. Inoltre ogni sua componente dovrà essere conforme all'originale. Tutto ciò in modo da garantire il suo corretto svolgimento.
- **RNF-04:** Il Robot dovrà essere usato solamente attraverso la sua griglia cartacea data in dotazione. Questo garantirà che l'applicazione sappia esattamente la posizione del Robot per poter controllare con esattezza i suoi spostamenti.
- **RNF-05:** La versione Beta che verrà sviluppata ne garantirà il funzionamento con la possibilità di riscontrare dei bug a livello software, che non compromettano le componenti del robot



- **RNF-06:** La versione finale dell'applicazione verrà rilasciata dopo un accurato periodo di correzione e di testing avrà una percentuale di bug relativamente bassa.

## 5.2. Requisiti di processo

- **RNF-07:** L'applicazione verrà sviluppata con Android Studio, mentre il funzionamento del Robot verrà sviluppato e controllato per mezzo del firmware LeJos e programmato in JAVA.
- **RNF-08:** Il Robot sarà costruito interamente con i componenti di Lego Mindstorms Education, utilizzando come mezzo di controllo l'EV3.
- **RNF-09:** Utilizzo di GitHub come sistema di versioning per il codice e i documenti.
- **RNF-10:** Utilizzo di Google docs per la creazione e la modifica di documenti.

## 5.3. Requisiti esterni

- **RNF-11:** Rispettare le direttive del committente e gli obiettivi definiti

## 6. Evoluzione del sistema

L'applicazione si basa sulle seguenti assunzioni fondamentali:

- **Connessione Bluetooth:** L'intero programma dovrà essere connesso al Robot tramite bluetooth per permettere la comunicazione app-Robot e per essere in grado di installare eventuali aggiornamenti del firmware LeJos.
- **Connessione Internet:** Il sistema dovrà essere regolarmente aggiornato in quanto dopo la Beta potranno essere rilasciate diverse versioni migliorate.

Gli aggiornamenti, rilasciati durante il periodo di testing, andranno a modificare l'applicazione android e il firmware LeJos come descritto nei punti indicati:

- **Miglioramento della grafica app android:** Potranno essere apportate migliorie dal punto di vista grafico.
- **Risoluzione possibili bug:** Saranno risolti i bug sia nella parte android che nella parte di LeJos, trovati nell'analisi dei feedback.
- **Ottimizzazione firmware:** Verranno risolti i possibili problemi di ottimizzazione del tipo:
  - velocizzare il robot nel completamento del percorso e nella verifica del colore
  - minor spreco delle risorse della batteria del EV3
  - riduzione dei tempi della comunicazione robot-app android
- **Aggiunta di differenti lingue :** Saranno supportati diversi linguaggi a seconda dell'espansione del prodotto
- **Aggiunta di funzionalità aggiuntive :** spostamento oggetti anche se non sono del colore ricercato

## 7. Specifica dei requisiti

Di seguito verranno mostrati le specifiche dei requisiti del sistema secondo la seguente tabella:

ID	Codice specifica dei requisiti
Input	Comando/azione in entrata nell'applicazione o nel robot che permette di effettuare un successivo comando/azione
Output	Azione effettuata in risposta ad un input
Destinazione	Mezzo coinvolto (applicazione, robot)
Pre-Condizione	Requisito soddisfatto per svolgere tale specifica
Post-Condizione	Stato dell'applicazione o del robot dopo l'azione svolta

ID	SRF-00
Input	Tap sull'icona per connettere il robot via bluetooth
Output	Attivazione del bluetooth
Destinazione	Robot
Pre-Condizione	Avviare l'applicazione
Post-Condizione	il robot e l'applicazione saranno connessi via bluetooth

ID	SRF-01
Input	Tap sulla cella della matrice virtuale presentata sull'app e scelta del colore
Output	Creazione informazioni da inviare al robot
Destinazione	Utente
Pre-Condizione	Collegamento via Bluetooth con il robot attivo
Post-Condizione	Su passa a SRF-02

ID	SRF-02
Input	Invio delle coordinate e del colore dall'app android al firmware LeJos
Output	Attivazione delle funzioni per il movimento del robot
Destinazione	Robot
Pre-Condizione	Cella e colore selezionato da parte dell'utente
Post-Condizione	Raggiungimento della cella e attivazione della funzione del controllo del colore

ID	SRF-03
Input	Raggiungimento cella selezionata
Output	Attivazione funzione controllo del colore
Destinazione	Robot
Pre-Condizione	Completamento del percorso
Post-Condizione	invio feedback

ID	SRF-04
Input	Feedback ricevuto da parte del robot
Output	Verifica colore da parte dell'app e apparizione messaggio di correttezza
Destinazione	Utente
Pre-Condizione	Colore verificato da parte del sensore del robot
Post-Condizione	Il messaggio dirà se il colore selezionato era quello scelto dall'utente.

## 8. Appendici

### 8.1. Requisiti hardware

Il nostro progetto richiede un dispositivo personale in cui installare l'applicazione, che deve avere le seguenti caratteristiche:

- Sistema operativo Android, versione minima 5.0 (*Lollipop*)
- Connessione Bluetooth supportata