

Facoltà di Ingegneria dell'Informazione
Laboratorio di Human Computer Interaction
Proff. Garzotto
Proff. Paolini

ROBOGAMES DESIGN

Robo Hide & Hunt

Developing team:

Matteo Botta

Alberto Bottinelli

Massimo Luraschi

A.A. 2008-2009

Indice

1	Introduzione	1
2	Risorse	2
3	Stato dell'arte	3
3.1	Il Wii Remote	3
3.1.1	La Telecamera IR	5
3.1.2	L'accelerometro	7
3.1.3	Comunicazioni pc Wii Remote	8
3.1.4	Librerie	9
3.2	Robot Spykee	10
3.3	Robowii	11
4	Requirements	12
4.1	Game goals	12
4.2	Target User	13
4.3	Numero di utenti	13
4.4	Context of use	13
5	Game concept	14
5.1	Regole Principali	14
5.2	Livelli di difficoltà	15
5.3	Controlli	16
5.3.1	Fase pre-gioco	16
5.3.2	Fase di gioco	16
5.4	Macchina a stati	17

5.4.1	Descrizione stati	18
5.4.2	Descrizione eventi	19
5.5	Tipi di interazione	20
5.6	Interazione Uomo - Robot	20
5.7	Feedback utente	22
5.7.1	Effetti sonori	22
5.7.2	Effetti visivi	23
6	Feasibility study	24
6.1	Tecnologie hardware utilizzate	24
6.2	Possibile implementazione software	27
6.2.1	Connessione	27
6.2.2	Gestione telecamera IR	27
6.2.3	Gestione accelerometro	28
6.3	Tecnologie trasmissione dei dati	28
6.4	Specifiche tecniche del gioco	29
6.4.1	Il Robot	30
6.4.2	Il Controller	30
6.4.3	Il Computer	31
6.5	Fattibilità realizzativa	32
7	User Experience Design: Scenarios and Storyboard	33
7.1	Scenario: vittoria Robot	33
7.2	Storyboard: vittoria Robot	34
7.3	Scenario: vittoria Player	43
7.4	Scenario: esaurimento tempo limite partita	43
7.5	Scenario: infrazione regole	44
8	Prototipo	45
9	Conclusioni	46
	Bibliografia	47
A	Breve descrizione del contenuto del CD/DVD	48

Capitolo 1

Introduzione

Lo scopo del nostro progetto è sviluppare un Robogame nel quale il robot agisca in modo autonomo. In questo studio illustreremo una proposta e ne analizzeremo la fattibilità tecnologica. L'obiettivo principale è far sì che l'interazione uomo-robot, sia essa di tipo competitivo o collaborativo, porti a massimizzare il divertimento e il coinvolgimento dell'utente. Il gioco 'Robo Hide & Hunt' consiste in una sfida robot-utente nella quale il player deve riuscire a colpire il nemico (il robot) un numero prefissato di volte evitando di essere individuato.

Capitolo 2

Risorse

Il progetto è stato realizzato da tre persone, le quali hanno dedicato in totale 375 ore. Il lavoro si è svolto principalmente in gruppo, discutendo e valutando insieme ogni singolo aspetto del problema al fine di trovare la migliore soluzione possibile.

Membro del gruppo	Monte ore
Matteo Botta	125
Alberto Bottinelli	125
Massimo Luraschi	125

Capitolo 3

Stato dell'arte

3.1 Il Wii Remote

Il Wii Remote è il principale controller della console Nintendo Wii. Proprio grazie ad esso la console Wii si differenzia dalle altre in commercio poiché permette all'utente di interfacciarsi col videogioco in modo semplice e intuitivo. Per giocare basta compiere movimenti simili a quelli usati nella vita reale o puntare semplicemente lo schermo della tv impugnando il Wiimote. Questo è permesso grazie ai due sensori contenuti all'interno del dispositivo. Il primo è un accelerometro che riconosce i movimenti a cui è soggetto il Wiimote; il secondo è una telecamera sensibile ai raggi infrarossi che permet-



Figura 3.1: Il Wii Remote.

te di determinare il punto del televisore mirato dal controller, grazie a dei led IR incorporati nelle estremità della Wii Sensor Bar (da porre sopra o sotto la televisione). Vi sono inoltre a disposizione del giocatore 10 tasti digitali standard, un tasto "power" che permette di spegnere la console e infine un tasto "sync" creato per connettere via bluetooth il Wiimote alla console. Impugnando il dispositivo si notano inoltre 4 led e uno speaker. All'interno del dispositivo, oltre ai due sensori principali, sono presenti una memoria EEPROM di 16 KiB, un piccolo motore elettrico interno, collegato ad una

massa eccentrica, che permette al Wiimote di vibrare e un chip bluetooth che rende possibile la comunicazione controller console. Nella parte inferiore del dispositivo è situata una porta di espansione alla quale è possibile collegare altri dispositivi, come ad esempio il Nunchuk che a sua volta contiene un accelerometro ed è dotato di 2 tasti digitali e uno stick analogico. Il controller è alimentato da due batterie Standard di tipo AA. Queste caratteristiche fanno sì che il Wiimote possa essere utilizzato in svariati domini applicativi dove è richiesta un'interfaccia uomo applicazione istintiva. La scelta di utilizzare questo controller è stata dettata dal fatto che permette di reperire tutte le informazioni necessarie per il funzionamento della nostra applicazione, potendo contare su di un apparecchio robusto e a basso costo. Di seguito vengono spiegati nel dettaglio i sensori principali e maggiormente coinvolti nel nostro lavoro.

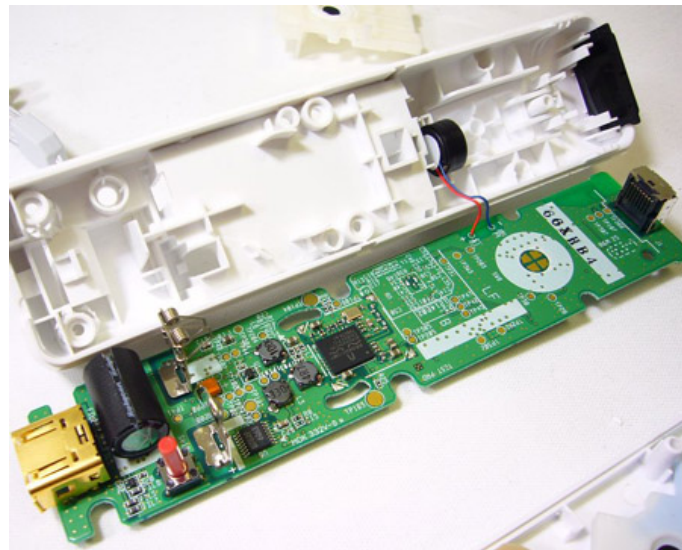


Figura 3.2: Interno del Wii Remote.

3.1.1 La Telecamera IR

Sulla parte anteriore del Wiimote è montato un sensore Pixart¹ che svolge la funzione di telecamera. Più precisamente si tratta di una telecamera monocromatica con definizione 128x96 con applicato un filtro che rende visibile al Wiimote solo luci infrarosse con una lunghezza d'onda definita (intorno ai 940 nm). Integrato nel Wiimote e col-

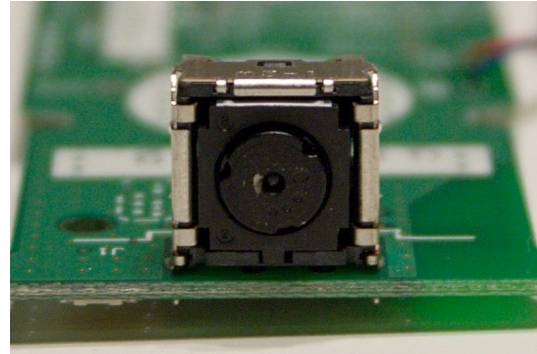


Figura 3.3: Telecamere del Wii Remote.

legato alla telecamera è presente un processore che si occupa di tracciare il movimento di 4 punti. Questo significa che il Wiimote non percepisce l'intera immagine acquisita dalla camera, ma solo i punti tracciati dal processore. Per tracciare l'andamento di questi punti è applicata una 8x subpixel analysis, che rende la dimensione spaziale di 1024x768. Il campo visivo della telecamere è di 45° sull'asse X e di 35° sull'asse Y e la frequenza di scansione è di 100 *hz*. Una volta abilitata la camera è possibile impostare la sensibilità e il formato di dati con il quale si desidera operare. I formati sono i seguenti:

- **BASIC** Nel formato Basic la camera restituisce 10 byte contenenti le coordinate X Y di tutti i 4 punti relative al piano di riferimento della telecamera. Le informazioni dei punti sono trasmesse a coppie, 5 byte per ogni coppia. La posizione di ogni punto è codificata in 10 bit sia per le coordinate sull'asse X che per quelle dell'asse Y con un range che va da 0 - 1023 per il primo e da 0 - 767 per il secondo.

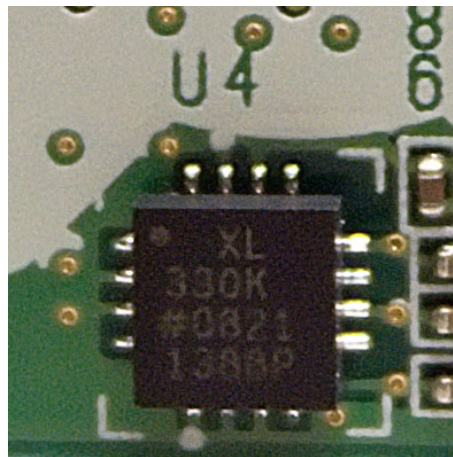
¹Per informazioni dettagliate sul sensore Pixart[6]

- **EXTENDED** Nel formato Extended, la camera oltre a restituire la posizione X Y del punto, ne fornisce anche la dimensione. Per ogni punto il Wiimote invia una stringa di 3 byte, 10 bit per X e 10 bit per Y con lo stesso range del protocollo Basic, più 4 bit rappresentanti la dimensione del punto in base ai pixel occupati sulla matrice del CCD della camera. Questo valore ha un range compreso tra 0 e 15, per un totale di 12 byte per tutti i 4 punti.
- **FULL** Nel formato Full, oltre a tutti i dati forniti dal formato Extended, sono presenti le informazioni riguardanti il bounding box ($\max(X)$, $\max(Y)$, $\min(X)$, $\min(Y)$) e l'intensità dei punti. Il bounding box rappresenta un rettangolo contenente tutti i punti visti dalla telecamera. Per ogni punto questa modalità richiede 9 byte per un totale di 36 byte per i 4 punti.

Al fine di individuare i punti con maggiore precisione, è possibile utilizzare la massima sensibilità permessa dalla telecamera. Per quanto riguarda invece il formato dei dati, è consigliabile l'Extended in quanto includeva l'informazione riguardante la dimensione del punto. Purtroppo però, dalle prove eseguite, è emerso che questo valore non è affatto affidabile. Avendo a disposizione 4 bit ci si aspetta un range tra 0 e 15, ma purtroppo nella realtà questo varia da 0 a 6 e non segue una distribuzione lineare. La Nintendo Wii utilizza come fonte di illuminazione la Sensor Bar, un dispositivo da posizionare sopra o sotto il televisore composta da 2 gruppi di 5 led infrarossi posti a una distanza predefinita. Questi 2 gruppi di led sono recepiti dal Wiimote come 2 punti e in base a questo la console determina il suo posizionamento rispetto alla Sensor Bar.

3.1.2 L'accelerometro

All'interno del Wiimote è inoltre presente un accelerometro² che rileva i movimenti a cui il controller è sottoposto in riferimento a una terna di assi solidale ad esso (Figura 3.5). Il sensore è posizionato sotto alla superficie superiore del Wiimote leggermente alla sinistra del pulsante A. L'accelerometro è l' ADXL330 inserito all'interno di un circuito integrato (Figura 3.4) prodotto dalla Analog Devices. Il dispositivo è



progettato per rilevare accelerazioni che vanno da $-3g$ a $+3g$ con un'incertezza del 10%. L'accelerometro calcola le accelerazioni in base ai movimenti delle piccole masse racchiuse all'interno dell'integrato.

Figura 3.4: Accelerometro del Wii Remote.

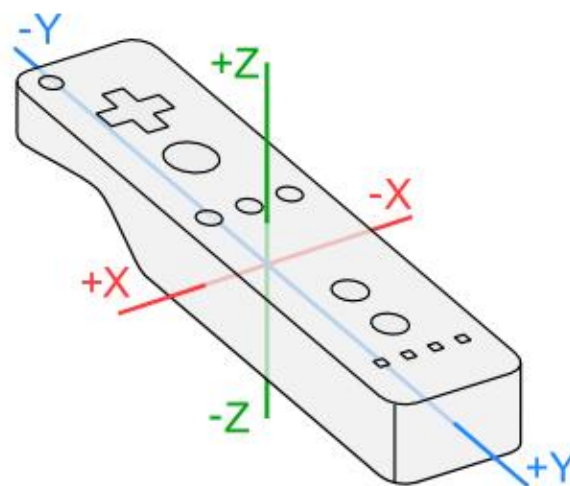


Figura 3.5: Sistema di riferimento solidale al Wiimote.

²Per informazioni dettagliate sull'accelerometro[4]

Il sensore registra sempre un'accelerazione su di un'asse a causa della forza di gravità e l'unica posizione in cui registra il valore 0 è nel caso in cui il Wiimote sia in caduta libera. All'interno della memoria EEPROM del Wiimote è memorizzato il valore equivalente a 0g e a +1g su ognuno dei 3 assi, poiché è necessario, in fase di fabbricazione, calibrare ogni singolo accelerometro. Ogni accelerazione dispone di 8 bit per la sua memorizzazione. Come la telecamera a IR anche l'accelerometro deve essere esplicitamente attivato dopo aver connesso il Wiimote al computer.

3.1.3 Comunicazioni pc Wii Remote

La comunicazione tra il Wiimote e gli altri dispositivi, quali computer o Nintendo Wii, avviene in modalità wireless, più precisamente utilizzando la tecnologia bluetooth. L'apparato che si occupa della comunicazione è un chip Broadcom BCM2042³. Questo provvede a gestire la comunicazione dei dati provenienti dai sensori del Wiimote quali accelerometro, camera IR, tasti digitali, led, vibrazione, speaker e le informazioni riguardanti il controller. Il Wii remote utilizza uno standard bluetooth HID (Human Interface Device) basato sul protocollo USB HID utilizzato per le comuni periferiche I/O (tastiere, mouse e joystick). Questo protocollo permette all'host di inviare una stringa di byte al Wiimote con la quale attivare i sensori desiderati e definire la modalità di trasmissione dei dati elaborati. A questo punto l'host si mette in ascolto dei pacchetti di informazioni provenienti dal Controller. Per connettere il Wiimote al pc non è necessaria nessuna autenticazione. Vi sono 2 modalità con il quale il controller può mettersi in discovery mode, cioè cercare un dispositivo al quale connettersi. La prima chiamata Soft Syncing, si attiva premendo simultaneamente i pulsanti 1 e 2, la seconda detta Hard Syncing si avvia

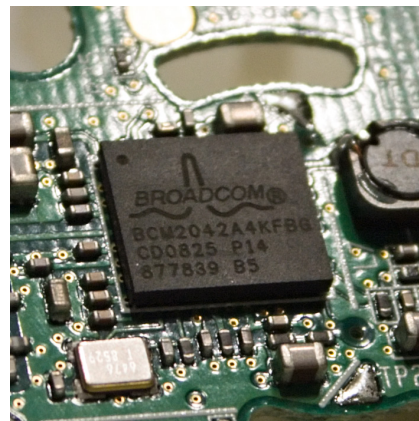


Figura 3.6: Il Chip Bluetooth.

³Per informazioni dettagliate sul dispositivo Bluetooth Broadcom BCM2042[5]

premendo il tasto sync posto all'interno del vano batteria. Per connettere il Wiimote viene normalmente utilizzato il Soft Syncing. Una volta connesso, il Wiimote invia all'host le seguenti informazioni:

Name Nintendo RVL-CNT-01

Vendor ID 0x057e

Product ID 0x0306

Major Device Class 1280

Minor Device Class 4

Service Class 0

Summary of all Class Values 0x002504

Queste informazioni identificheranno il Wiimote e, nel caso si voglia connettere più controller, ci si baserà sull'indirizzo MAC.

3.1.4 Librerie

Anche se le specifiche del protocollo di comunicazione del Wiimote non sono pubbliche, essendo conformi al protocollo Bluetooth HID è stato possibile, tramite reverse engineering, sviluppare facilmente svariate librerie per comunicare con esso. Di seguito riportiamo uno schema riassuntivo delle principali:

Nome	Linguaggio	Sistema Operativo
Wiiuse	C	Linux, Windows
Wiiusej	Java	Linux, Windows
DarwiinRemote	Objective C	OSX
WiiremoteJ	Java	Linux, Windows, OSX
Wiigee	Java	Linux, Windows, OSX
MoteJ	Java	Linux, Windows

3.2 Robot Spykee

Spykee è un robot wi-fi autonomo, che può essere controllato da remoto da un qualsiasi computer connesso ad internet. Sfruttando i numerosi sensori presenti a bordo (Figura 3.7), Spykee è in grado di muoversi, ascoltare, emettere suoni e monitorare l'ambiente circostante. Di seguito sono riportate le caratteristiche hardware.

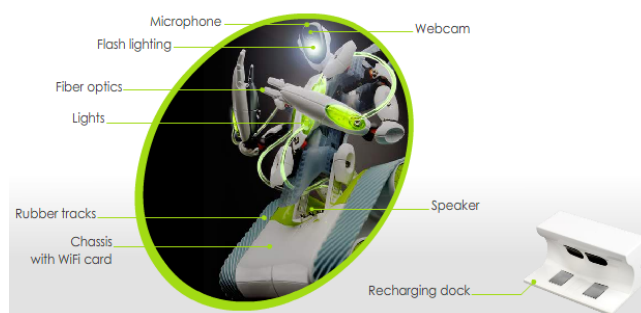


Figura 3.7: Robot Spykee

- PC & MAC compliant
- WiFi - 802.11 b/g
- Joystick compliant
- PC requirement - software: Windows 2000 - XP with .NET framework 2.0
- MAC requirement - software: Mac OS X
- Open source software
- MMI software details: 1.1 Mb - user interface in Microsoft.net
- Compatible with Skype 3.0 (PC) technology
- Camera: 320 X 240 local mode:15 fps - distant mode: depending on the Internet connection
- Speaker: 2W - driver required

3.3 Robowii

Robowii⁴ è un gioco basato sull'interazione tra un utente che utilizza un controller Wiiremote e un robot. Il robot utilizzato nella prima versione del gioco (Figura 3.8) è stato 'preso in prestito' dal progetto Robocup, mentre nell'ultima versione si utilizza un robot Spykee. Questo gioco è stato il primo tentativo di creare un'applicazione ludica utilizzando un robot e il Wii Remote. Il gioco consiste nel tentare di colpire un obiettivo posto sul robot mentre questo parte da un punto iniziale e cerca di arrivare ad una zona finale. Per rendere più interessante il gioco è stato introdotto un meccanismo di carica, infatti per colpire il robot è necessario che quest'ultimo sia stato prima puntato dall'utente per un tempo prefissato.

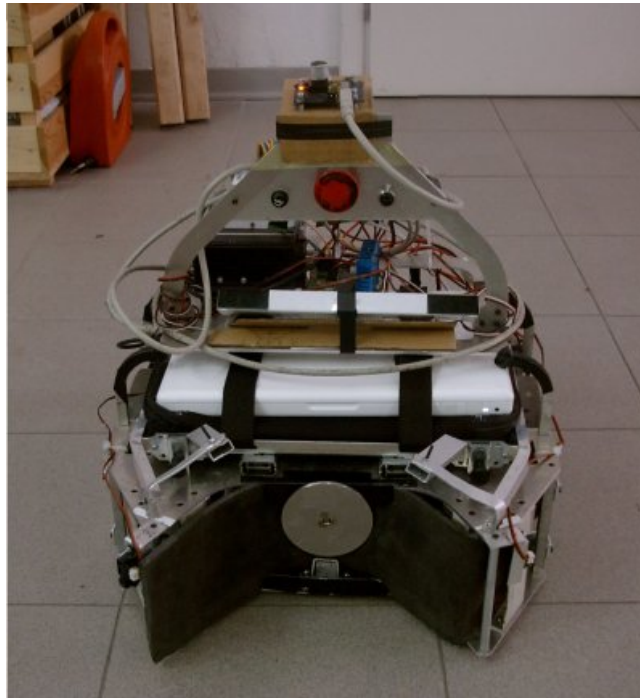


Figura 3.8: Robot utilizzato nella prima versione del gioco

⁴Per informazioni dettagliate sul Robowii[2]

Capitolo 4

Requirements

4.1 Game goals

L'obiettivo del nostro studio è quello di realizzare un robogame partendo da un affermato gioco per bambini: il nascondino. La nostra attenzione si è focalizzata nel rendere il gioco interattivo, divertente e dinamico al fine di creare un'esperienza di gioco nuova e coinvolgente che attragga l'utente. A tale scopo si sono analizzati i principali aspetti del gioco classico, valutandone lati positivi e negativi così da individuare i punti critici sui quali intervenire utilizzando la tecnologia a disposizione. Il restyling del gioco ha apportato le seguenti modifiche:

- L'obiettivo dell'utente non è più quello di raggiungere una meta prefissata senza farsi individuare ma è quello di sconfiggere l'avversario, rappresentato dal robot, senza farsi individuare. In questo modo si nega al giocatore la possibilità di rimanere solo nascosto e lo si obbliga ad interagire con Spykee per far evolvere il gioco.
- Ogni utente non deve più limitarsi a nascondersi, ma ha a disposizione un controller Nintendo Wii Remote con il quale deve sparare al robot per vincere.
- Spykee svolge il ruolo di cacciatore e il suo obiettivo è quello di individuare i giocatori prima che loro lo colpiscano.

- Per rendere equilibrato il gioco si è fatto in modo di limitare il divario tra i vari giocatori, incluso spykee, così da impedire che caratteristiche intrinseche dei soggetti possano avvantaggiarli nel raggiungere la vittoria. Ad esempio il divario motorio tra un bambino e il robot è ancora troppo elevato, per questo motivo si è fatto in modo che la competizione fisica non sia determinante nel gioco.
- Partendo da queste considerazioni sono state studiate particolari regole che vanno ad aumentare ulteriormente l'interazione e il divertimento.

4.2 Target User

L'applicazione è pensata per bambini che rientrano un intervallo di età compreso tra i 6 e 10 anni poiché, come i classici giochi puerili da cui deriva, oltre una certa soglia di età non suscitano più un interesse continuo. L'età minima, invece, è giustificata dalla necessità di comprensione del regolamento di gioco.

4.3 Numero di utenti

Il gioco nasce come uno contro uno (giocatore-robot) ma è possibile espandere la giocabilità a più giocatori umani ed eventualmente più cacciatori robotici.

4.4 Context of use

Il gioco è pensato per essere utilizzato in ambienti interni. Questa scelta è dovuta al fatto che i sensori presenti sul robot potrebbero non funzionare correttamente e rilevare informazioni sbagliate se utilizzati all'aperto. Inoltre non bisogna trascurare le possibili difficoltà di movimento riscontrate dal robot in caso di asperità del terreno, che renderebbe impossibile lo svolgimento del gioco.

Capitolo 5

Game concept

Robo Hide & Hunt consiste in una sfida robot-utente nella quale il player deve riuscire a colpire il nemico (Spykee) un numero prefissato di volte evitando di essere individuato dal robot. Lo sparo dell'utente per essere considerato valido deve essere fatto alle spalle del robot. Durante il gioco l'utente dovrà evitare di essere avvistato dal robot altrimenti il punteggio di Spykee verrà incrementato. Il gioco termina quando uno dei due concorrenti raggiunge il punteggio prefissato. In questo capitolo andremo ad analizzare i principali aspetti del gioco.

5.1 Regole Principali

- **Vittoria Player:** l'utente deve colpire il robot un numero prefissato di volte entro un tempo limite.
- **Vittoria Robot:** Spykee deve stanare il giocatore e mantenerlo inquadrato nel campo visivo per un tempo prefissato (*Continuous mode*) o per un numero prefissato di avvistamenti (*Discrete mode*). Il tempo è cumulabile tra i vari avvistamenti.
- Il gioco inizia con la 'conta' del robot che permette all'utente di nascondersi.
- Il colpo del giocatore viene considerato valido se effettuato quando l'utente non è nel campo visivo del robot, cioè si trova alle sue spalle.

- Il fucile necessita di un tempo di ricarica dopo ogni sparo, durante il quale non può sparare (cooldown).
- Il giocatore non può rimanere nascosto senza sparare per più di un tempo prestabilito altrimenti la partita è vinta dal robot.
- Al giocatore non è consentito portare attacchi fisici al robot.
- l'avvistamento del robot è considerato valido quando la telecamera montata su di esso individua il player.
- Ad ogni sparo andato a vuoto il robot effettua un giro su se stesso per individuare se il player nelle vicinanze.
- Ogni volta che il robot viene colpito rimane 'stordito' per un breve periodo di tempo durante il quale non può avvistare il giocatore.
- Nel caso in cui il tempo limite venga raggiunto vince il giocatore che al termine della partita ha totalizzato più punti.

5.2 Livelli di difficoltà

E' possibile personalizzare il livello di difficoltà del gioco settando diversi parametri attraverso il controller Wiimote, entrando nel menù di configurazione.

Cooldown arma: è possibile modificare il tempo di ricarica dell'arma.

Modalità punteggi robot: è possibile selezionare in che modo il punteggio del robot venga incrementato:

- *Continuos mode:* viene sommato il tempo in cui il robot mantiene il Player all'interno del suo campo visivo
- *Discrete mode:* ogni avvistamento incrementa il punteggio del robot.

Setting Continuos mode: si seleziona il tempo totale di avvistamento per la vittoria del robot.

Setting Discrete mode: si seleziona il numero totale di avvistamenti necessari alla vittoria del robot.

Tempo Limite: si definisce la durata massima della partita.

Tempo Conta: si definisce la durata della conta iniziale.

Tempo Inattività Player: si definisce l'intervallo massimo tra due spari consecutivi.

5.3 Controlli

Di seguito è riportata una tabella con le azione associate ai tasti del controller impugnato dal giocatore in base alla fase di gioco.

5.3.1 Fase pre-gioco

In fase pre gioco all'avvio dell'applicazione una voce guiderà l'utente nella scelta delle varie opzioni.

Tasto	Azione corrispondente
Tasti Direzionali	Permettono di muoversi tra le varie opzioni del menu.
Tasto A	Conferma scelta
Tasto B	Inizia partita
Tasto 1	Annula / back

5.3.2 Fase di gioco

Durante la fase di gioco il robot emetterà suoni in base ai tasti premuti e alle azioni verificatesi. Il punteggio verrà inoltre visualizzato su due barre di led presenti sul robot.

Tasto	Azione Corrispondente
Tasto B	Spara
Tasto A	Il robot enuncia il punteggio attuale
Tasto 1	Torna al menu principale abbandonando la partita
Tasto 2	Attiva e disattiva la pausa

5.4 Macchina a stati

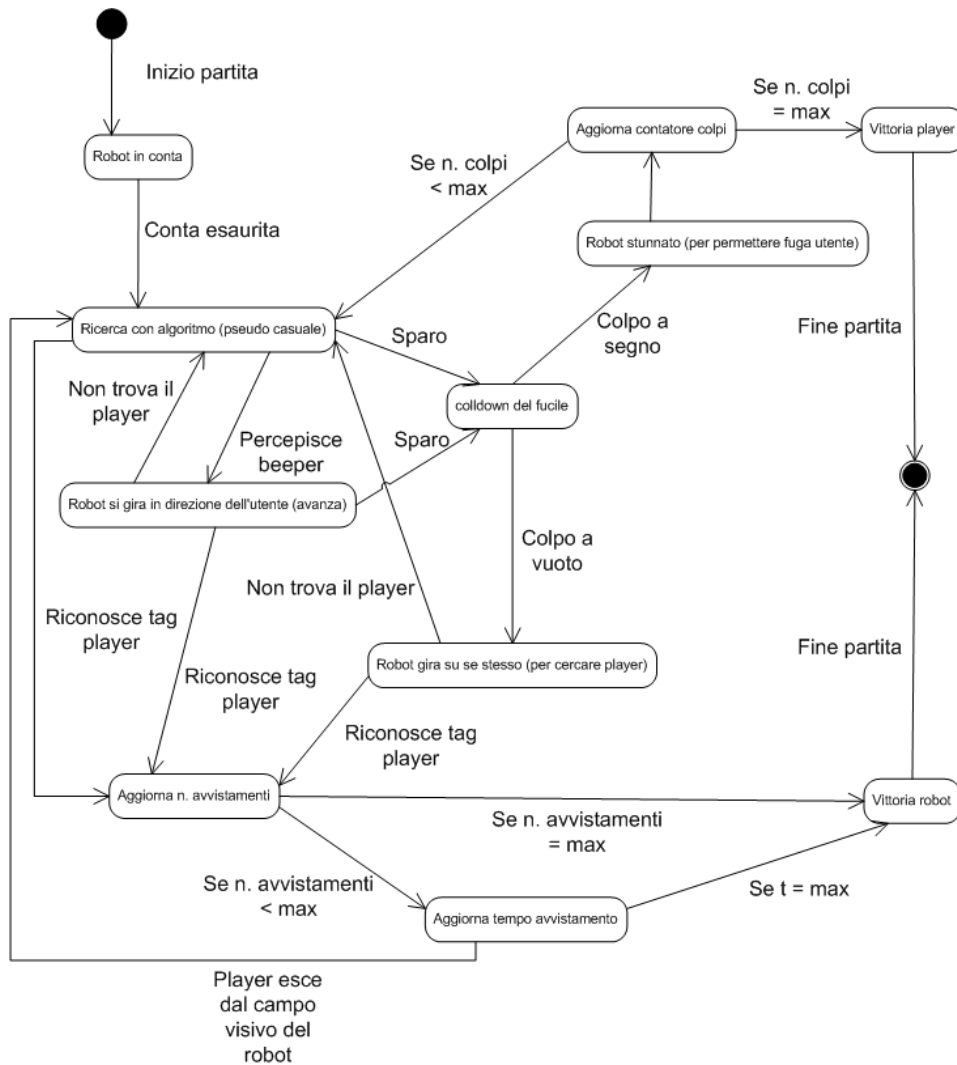


Figura 5.1: Macchina a stati rappresentante le possibili fasi del gioco

5.4.1 Descrizione stati

Robot in conta: all'inizio della partita il robot lascia un margine di tempo di fuga all'utente.

Robot in ricerca (con algoritmo pseudo casuale): il robot cerca il player all'interno dell'ambiente di gioco attraverso un algoritmo di esplorazione. Con algoritmo pseudo casuale si intende un comportamento che permette al robot di esplorare in modo random la stanza in cerca dell'utente.

Robot si gira in direzione dell'utente (avanza): quando il robot percepisce il beeper cerca di dirigersi nella direzione individuata.

Cooldown del fucile: ogni volta che l'utente spara, il fucile rimane inattivo per un tempo prefissato.

Robot stunnato (per permettere la fuga utente): dopo che il robot viene colpito rimane in uno stato di stun (sensori beeper e telecamera disattivati) per un breve intervallo di tempo, che permette all'utente di scappare.

Robot gira su se stesso (per cercare player): dopo uno sparo a vuoto il robot compie una rotazione su se stesso al fine di individuare la presenza dell'utente.

Aggiorna numero avvistamenti: incrementa il numero di avvistamenti del robot, aggiornando il punteggio visualizzato dai led.

Aggiorna contatore colpi: incrementa i colpi messi a segno dal player, aggiornando il punteggio visualizzato dai led.

Aggiorna tempo avvistamento: incrementa il tempo di avvistamento del robot, aggiornando il punteggio visualizzato dai led.

Vittoria robot: il robot vince la partita.

Vittoria player il player vince la partita.

5.4.2 Descrizione eventi

Inizio partita: il gioco ha inizio.

Conta esaurita: termina il periodo di tempo lasciato al player per scappare e il robot inizia la ricerca.

Non trova il palyer: il robot non percepisce nessun seganle dall'utente (ne dalla videocamera ne dal beeper).

Percepisce il beeper: il robot percepisce il segnale emesso dal beeper dell'utente.

Riconosce tag player: il tag dell'utente entra nel capo visivo del robot.

Sparo: il player spara.

Colpo a vuoto: il player spara e manca il robot.

Colpo a segno: il player spara e colpisce il robot.

Se n . avvistamenti $<$ max: il numero di volte che l'utente è entrato nel campo visivo del robot è minore del numero necessario di avvistamenti per la vittoria del robot.

Se n . avvistamenti $=$ max: il numero di avvistamenti dell'utente, da parte di Spykee, ha raggiunto il numero necessario per la vittoria del robot.

Se $t =$ max: il tempo in cui l'utente è stato all'interno del campo visivo del robot è uguale al tempo di avvistamento necessario per far vincere il robot.

Se n . colpi $<$ max: il numero di spari andati a buon fine sono minori di quelli necessari per vincere.

Se n . colpi $=$ max: il numero di spari andati a buon fine sono uguali a quelli necessari per la vittoria.

Fine partita: il gioco termina.

5.5 Tipi di interazione

Competitiva: Il Robot e l'utente si sfidano l'uno contro l'altro durante la fase di gioco.

Diretta: Utente e robot si scambiano direttamente segnali tra loro, sia visivi che sonori (sparo, led luminosi, ecc...).

In real-time: Tutte le interazioni uomo - robot avvengono in real time all'interno del gameplay.

5.6 Interazione Uomo - Robot

In questa sezione vengono illustrati i comportamenti attuati dall'applicazione per interagire con l'utente in risposta ai principali eventi che si verificano durante il gameplay. Quasi tutte le azioni sono accompagnate da segnali vocali e visivi che aiutano il giocatore a comprendere le situazioni in modo semplice, intuitivo e divertente. L'interazione è obbligata all'interno del game per evolvere nelle fasi del gioco ed arrivare alla vittoria. Il gioco è pensato per porre sullo stesso livello uomo e robot, cercando di eliminare il divario intrinseco che li differenzia.

Le tecnologie a nostra disposizione non permettono infatti di creare un'applicazione dove la componente fisica sia discriminante, poichè il gioco risulterebbe impari e di conseguenza noioso e poco stimolante. Basandoci su questi ragionamenti si sono sviluppati i principali punti di interazione player-robot illustrati di seguito.

Inizio ricerca: Il robot scandisce il tempo della conta e ne annuncia il termine con messaggi vocali. Finita la conta il robot inizia ad esplorare l'ambiente in cerca del player.

Avvistamento: quando il robot, durante la ricerca, percepisce il segnale del beeper o il giocatore viene avvistato dalla camera del robot viene emesso un messaggio sonoro. Inoltre, la barra dei led viene accesa gradualmente in base al punteggio del robot. Il controller continua a vibrare per tutto il tempo dell'avvistamento.

Sparo: l'azione dello sparo da parte dell'utente genera i seguenti eventi:

- Vengono attivati tutti i led sul wiimote per indicare lo sparo.
- Inizia la fase di cooldown del fucile segnalato dallo spegnimento graduale dei led. Se il giocatore tenta di sparare in questo lasso di tempo un segnale audio gli indicherà l'impossibilità di eseguire l'azione.
- In base al successo del colpo viene emesso un segnale sonoro adeguato e viene incrementato il punteggio utente (Su di una seconda barra di led posta sul robot).
- Nel caso in cui l'utente si trovi nel campo visivo del robot ed effettui un colpo, il colpo è considerato non valido e spykee emetterà un segnale sonoro ad indicare l'evento.
- il controller vibra.

Fuga: un avviso sonoro segnala l'uscita del giocatore dal campo visivo del robot.

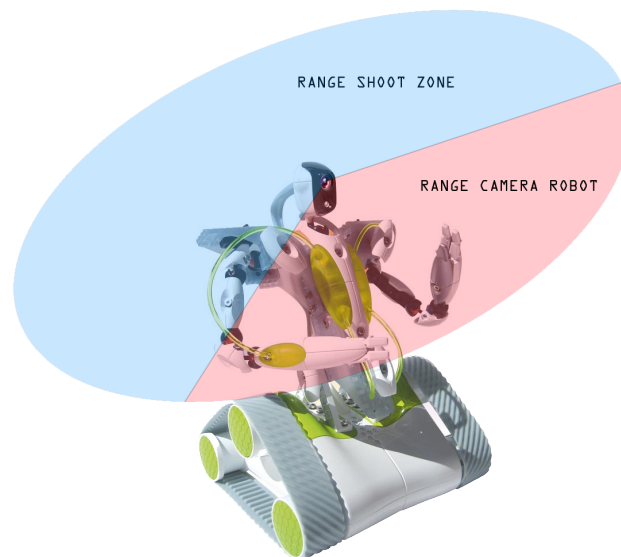


Figura 5.2: zona di avvistamento comparata con la zona in cui il robot può essere colpito.

5.7 Feedback utente

Per garantire un maggiore livello di usability è necessario che ogni evento verificatosi all'interno del gioco sia seguito da un feedback semplice ed immediato, che permetta agli utenti di comprendere istantaneamente l'esito della propria azione. A tal proposito vengono utilizzati numerosi messaggi sonori e visivi all'interno del gameplay, in aggiunta ad un menu iniziale di setting che permette ad ogni utente di customizzare ogni partita, personalizzando e rendendo così differente ogni game experience.

5.7.1 Effetti sonori

Di seguito è riportata una tabella che enuncia i suoni e i messaggi emessi da Spykee durante il gameplay.

Evento	Audio associato
Spykee conta	Enuncia i secondi che passano '1, 2, 3...'
Spykee inizia ricerca	'Vengo a prenderti!'
Spykee percepisce il beeper	'Ti ho sentito!'
Spykee avvista il player	'Ti Vedo!' + vibrazione controller (intermittente)
Sparo	'Bang!'(Suono) + vibrazione controller
Spykee viene colpito	'Mi hai colpito!'
Sparo a vuoto	'Mi hai mancato!'
Sparo non valido	'Non puoi colpirmi!'
Player esce dal campo visivo di Spykee	'Non ti vedo più!'
Termine partita	Enuncia il vincitore
Tempo scaduto (Player inattivo)	'Spykee vince!!'

5.7.2 Effetti visivi

Di seguito è riportata una tabella riassuntiva di tutti gli effetti luminosi che si possono verificare durante il gioco.

Azione	Evento visivo
Punto Player	Ad ogni punto viene acceso un led sulla barra punteggio player
Punto Spykee	Ad ogni punto viene acceso un led sulla barra punteggio robot
Sparo	Si accendono i 4 led presenti sul Wiimote
Cooldown dell'arma	I 4 led presenti sul wiimote si spengono gradualmete, il cooldown termina quando tutti i led sono spenti
Vittoria Player	La barra led player lampeggia + vibrazione controller (continua)
Vittoria Spykee	La barra led robot lampeggia

Capitolo 6

Feasibility study

6.1 Tecnologie hardware utilizzate

In questo paragrafo vengono illustrate e motivate le tecnologie coinvolte nella realizzazione di questo robogame.

Il Robot: in commercio esistono due robot adatti al nostro gioco: Spykee e Rovio, differiscono tra loro principalmente per le capacità di movimento (il secondo è ologonico mentre l'altro no). Su entrambi è possibile installare i dispositivi necessari per il funzionamento del nostro robogame. E' stato scelto Spykee poichè avendo entrambi caratteristiche simili il fattore discriminante è stato il costo. Si elencano le caratteristiche hardware del robot Spykee:

- PC & MAC compliant
- WiFi - 802.11 b/g
- Joystick compliant
- PC requirement - software: Windows 2000 - XP with .NET framework 2.0
- MAC requirement - software: Mac OS X
- Open source software
- MMI software details: 1.1 Mb - user interface in Microsoft.net
- Compatible with Skype 3.0 (PC) technology

- Camera: 320 X 240 local mode:15 fps - distant mode: depending on the Internet connection
- Speaker: 2W - driver required

Il Wii remote¹: e' il controller principale della console Nintendo Wii. La presenza di sensori (accelerometro, telecamera sensibile ai raggi infrarossi), pulsanti e led insieme alla tecnologia bluetooth lo rendono uno strumento flessibile, intuitivo nell'utilizzo e facilmente interfacciabile sia con l'utente che con l'applicazione, perciò adatto ai nostri scopi anche per il suo costo relativamente basso.

Input:

Accelerometro: è in grado rilevare un accelerazione rispetto ad una terna d'assi solidale al controller con un range di funzionamento che va da -3g/+3g.

Sensore Ir: Telecamera ir in grado di effettuare il tracking di al massimo 4 punti emettitori di infrarossi, risoluzione 128x96 pixel.

Pulsanti: tasti posizionati in modo chiaro e ugualmente utilizzabili sia da utenti destri che mancini.

Output:

Led: 4 led posizionati sulla base del wiimote.

Vibrazione

Speaker

Cintura sonar: è applicata al robot per permettergli di evitare gli ostacoli (obstacle avoidance). Il principio di funzionamento consiste nell'emissione di un segnale sonoro e nella registrazione dell'eco di ritorno consentendo di individuare ostacoli nelle vicinanze (come fanno ad esempio i pipistrelli).

Barra led luminosi: sul robot verrà applicata una barra di led luminosi per comunicare all'utente eventi particolari del gioco.

Led infrarossi: il robot sarà reso visibile al Wiimote tramite un numero di led infrarossi posti in una posizione chiave (sulle spalle di spykee) in modo da essere riconoscibili dalla telecamera del controller. I led devono avere una lunghezza d'onda pari a 940 nm per essere meglio visualizzati dalla camera del controller.

Cinturino player: per rendere visibile e localizzabile il giocatore da parte del robot, questo deve indossare un particolare cinturino/braccialetto dotato di un beeper in grado di emettere un costante segnale sonoro (ad una frequenza prestabilita non udibile all'orecchio umano) che permetta al robot di localizzare approssimativamente il giocatore. Inoltre quando l'utente è nel campo visivo del robot dovrà essere individuato univocamente tramite un tag presente sullo stesso cinturino (led, materiale colorato, ecc).

Riconoscitore del beeper: per localizzare il giocatore attraverso il beeper, sono necessari dei microfoni, del tutto simili a quelli della cintura sonar posti sul robot, tarati sulla frequenza prestabilita del beeper. Si utilizza una tecnica denominata D.O.A.² (*direction of arrival*): questa tecnica individua la direzione dalla quale un'onda sonora si propaga e raggiunge un punto.

²Per informazioni riguardanti questa tecnica[7]

6.2 Possibile implementazione software

Per gestire l'applicazione robo Hide & Hunt è possibile utilizzare come linguaggio di programmazione Java, in quanto è multiplatforma (utilizzabile su diversi sistemi operativi). Per far sì che Java riesca a interfacciarsi e utilizzare il Wiimote ci si basa sulla libreria open-source Motej³ 0.8⁴ dopo averla adeguatamente adattata e modificata. Di seguito trattiamo i punti principali da sviluppare.

6.2.1 Connessione

Una volta trovato un device, vengono scritte sul file di log le seguenti informazioni: Name, Bluetooth Address, Major Device Class, Minor Device Class e Service Classes. Queste informazioni identificano in modo univoco il dispositivo. Per verificare che il dispositivo collegato sia di tipo Wii remote bisogna controllare che il Name inizi con Nintendo, che il Major Device Class corrisponda a 1280 e il Minor Device Class sia 4. Se le informazioni rilevate corrispondono, il device verrà accettato e connesso al software, il quale gli assocerà la variabile "Mote" attraverso la quale sarà possibile ottenere e inviare tutte le informazioni necessarie.

6.2.2 Gestione telecamera IR

0x33: Core Buttons and Accelerometer with 12 IR bytes

This mode returns data from the buttons, accelerometer, and IR Camera in the Wii Remote:

```
(a1) 33 BB BB AA AA AA II II II II II II II II II II II
```

BBBB is the core Buttons data. AA AA AA is the Accelerometer data. The 12 II bytes are from the built-in IR Camera.

Figura 6.1: Struttura dati da noi utilizzata per leggere i dati relativi al Wiimote.

Come si può notare (Figura 6.1) il primo byte di informazione riguardante la camera è il settimo. Questo contiene il valore della coordinata X del

³Per un ulteriore approfondimento[1]

⁴Per tutte le informazioni riguardanti libreria Motej[3]

primo punto visto dal sensore, mentre il byte numero 8 contiene il valore della coordinata Y. Osservando il formato Extended (Figura 2.7) si può notare che ogni coordinata è rappresentata con 10 bit non consecutivi. La coordinata X è contenuta nel byte 0 più 2 bit nel byte 2, mentre la coordinata Y è contenuta nel byte 1 più 2 bit del byte 2. Inoltre 4 bit vengono occupati dalla dimensione del punto. A fronte di ciò il software si preoccupa di estrapolare il valore delle coordinate svolgendo operazioni logiche sui bit in questione. Una volta estrapolati i valori e controllata la loro veridicità, questi vengono letti dal Listener che si occupa della loro gestione.

6.2.3 Gestione accelerometro

Come si può notare (Figura 6.1) il primo byte di informazione riguardante l'accelerometro è il quarto. Questo contiene il valore dell'accelerazione misurata sull'asse X, mentre i byte numero 5 e 6 contengono rispettivamente il valore dell'accelerazione letti sull'asse Y e sull'asse Z. Osservando la struttura dati, è possibile notare che ogni asse ha a disposizione 8 bit per memorizzare l'informazione. Perciò il software si preoccupa di estrapolare il valore delle accelerazioni sui relativi assi e, una volta estrapolati, questi vengono letti dal Listener che si occupa della loro gestione.

6.3 Tecnologie trasmissione dei dati

Le comunicazioni pc-robot avvengono tramite tecnologie wi-fi, in quanto è necessario avere una connessione robusta, senza fili e di portata elevata. Per quanto riguarda il Wiimote viene utilizzata la tecnologia bluetooth per connetterlo all'elaboratore. Con questa configurazione il computer centrale gestisce e regola tutte le parti dell'applicazione. Questo robogame è stato pensato per essere usato in ambiente chiuso con una superficie circoscritta a causa dei limiti imposti dalla tecnologia usata. Infatti i dongle bluetooth in commercio hanno un range operativo massimo di 100 m e quindi è necessario che la distanza bluetooth wiimote non superi mai questo limite.

6.4 Specifiche tecniche del gioco

In questa sezione si illustrano le specifiche relative ai componenti utilizzati all'interno dell'applicazione. In particolare si analizza la struttura e l'utilizzo delle varie parti del robot e del controller.

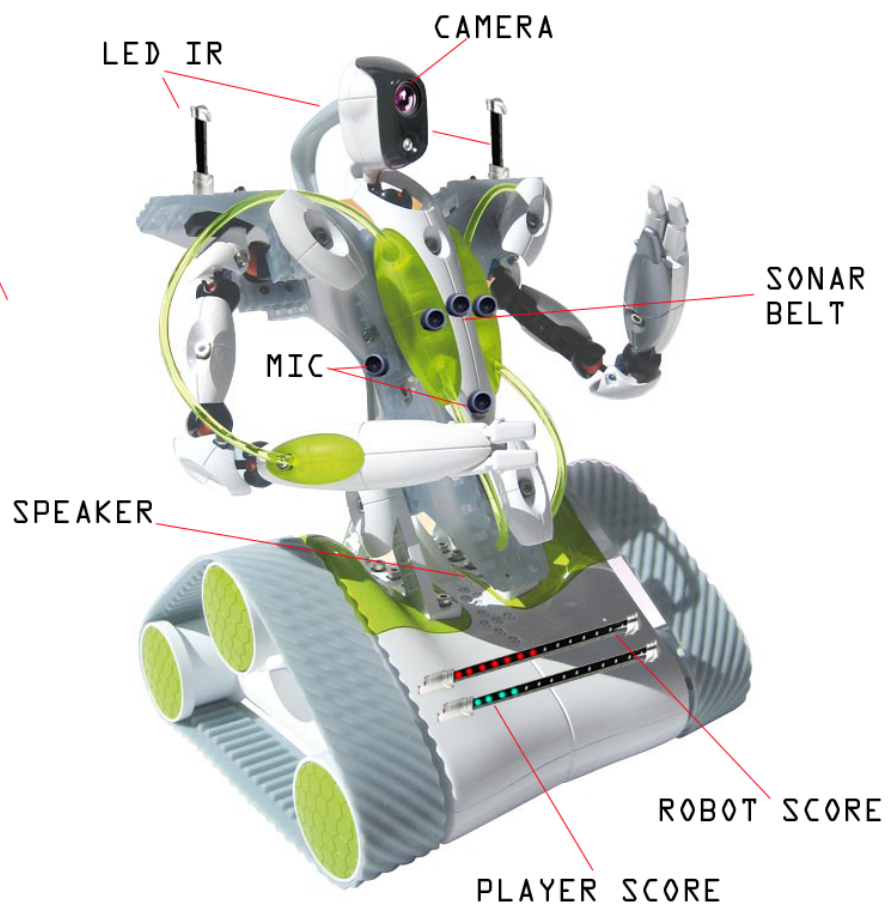


Figura 6.2: Spykee modificato

6.4.1 Il Robot

Per questo progetto si è deciso di utilizzare Spykee, un robot mobile presente in commercio, adeguatamente modificato sulla base delle nostre valutazioni. Sulla parte frontale del robot vengono applicate due barre di led luminosi di colori diversi, utilizzate per segnalare il punteggio relativo a player e robot. Sul corpo del robot è inoltre aggiunta una cintura sonar per permettere a Spykee di evitare gli ostacoli presenti sul suo cammino. E' inoltre dotato di una serie di microfoni in grado di captare il segnale del beeper, i quali permettono al robot di dirigersi verso la direzione con una maggior intensità sonora, presumibilmente dove si trova il palyer. Sulle 'spalle' di Spykee sono posizionati due led ad infrarossi (lunghezza d'onda 940 *nm*) per rendere visibile il robot al Wiimote. Segnali vocali e suoni, saranno emessi dallo speaker presente sul robot per aumentare il grado di interazione e coinvolgimento dell'utente duranre il gioco. Il player è identificato dalla telecamera posta su spykee tramite i tag presenti sul cinturino del giocatore.

6.4.2 Il Controller

All'interno della nostra applicazione il giocatore interagisce con il robot grazie all'utilizzo del controller Wii Remote. Con questo dispositivo il player potrà mirare e sparare al robot per colpirlo. L'azione dello sparo va a buon fine se e solo se i led ad infrarossi presenti su Spikè entrano nel campo visivo della telecamera IR posta sul Wiimote. Premendo il tasto B situato sul retro del controller, il Wiimote 'fa fuoco' e lo segnala al giocatore vibrando. Dopo ogni sparo è necesario attendere un intervallo di tempo predefinito per poter esplodere un nuovo colpo. I led sul controller indicheranno il countdown del tempo di ricarica. Sul dispositivo o sul cinturino da portare al braccio, verranno applicati beeper e tag che permettono al robot di individuare il giocatore. I tag potranno essere eventualmente posizionati in zone diverse in base alle esigenze dell'utente ricordando però che questi devono essere facilmente visibili al robot (una collana, una cavigliera o un paio di occhiali).



Figura 6.3: il controller Wii Remote

6.4.3 Il Computer

L'applicazione sarà interamente caricata su un pc al quale verranno connessi il robot e il controller. L'intera elaborazione è affidata al PC, come anche l'analisi degli input forniti dall'utente tramite il controller e l'intera gestione del gioco (punteggi player e robot, inizio partita, ecc). Il computer deve essere posizionato in zona centrale rispetto al campo di gioco in relazione alla portata dei dispositivi wireless, in modo che il robot e il controller rimangano sempre connessi al PC. Un'alternativa è collocare il computer direttamente on-board a Spykee aumentando così la libertà di movimento durante il gioco.

6.5 Fattibilità realizzativa

Tutte le componenti sia hardware che software trattate all'interno di questo elaborato sono state da noi approfonditamente analizzate al fine di verificarne l'effettiva possibilità di implementazione e realizzazione. Le scelte progettuali da noi adottate sono la conseguenza di attente prove pratiche e di approfondite ricerche.

Capitolo 7

User Experience Design: Scenarios and Storyboard

7.1 Scenario: vittoria Robot

Il player, un bambino di età compresa tra i 6 e i 10 anni, avvia l'applicazione e seguendo i messaggi vocali configura, tramite il wiimote, i parametri relativi alla partita impostando il numero di punti necessari alla vittoria uguale a 3 e impostando il *discrete mode*. Avviato il gioco il robot concede un intervallo di tempo, settato in precedenza, al giocatore per allontanarsi e nascondersi. Terminata la conta il robot inizia la ricerca del player. L'utente spara, il colpo va a vuoto e si attiva il cooldown dell'arma. Il robot compie un giro completo su se stesso, individua il giocatore presente nelle vicinanze, mettendo così a segno un punto a suo favore. Il robot annuncia l'avvistamento del player che quindi fugge e torna a nascondersi. A questo punto il robot rizia a cercare. Il player da posizione nascosta spara alle spalle del robot e mette a segno il colpo. Il robot, finito il tempo di 'stordimento', compie una rotazione per avvistare il player che nel frattempo è andato a nascondersi. Da questa posizione il player tenta di colpire nuovamente il nemico incurante della sua posizione frontale, esponendosi così alla visuale del robot, che lo individua e aggiorna il punteggio. L'utente colto impreparato, tenta di fuggire invano, restando nel campo visivo dell'avversario, facendo così raggiungere al robot il numero di avvistamenti necessari per vincere la partita.

7.2 Storyboard: vittoria Robot

Di seguito è riportato lo storyboard relativo allo scenario illustrato in precedenza.

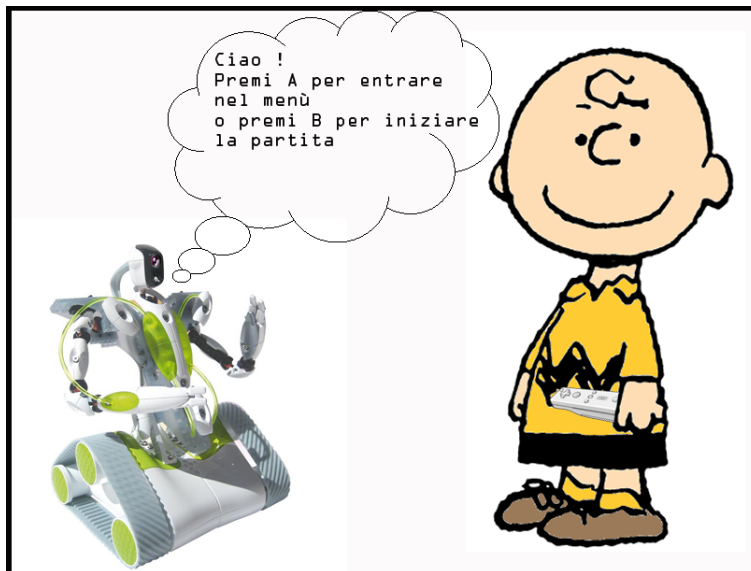


Figura 7.1: Charlie avvia il gioco.



Figura 7.2: Charlie preme B per iniziare la partita.

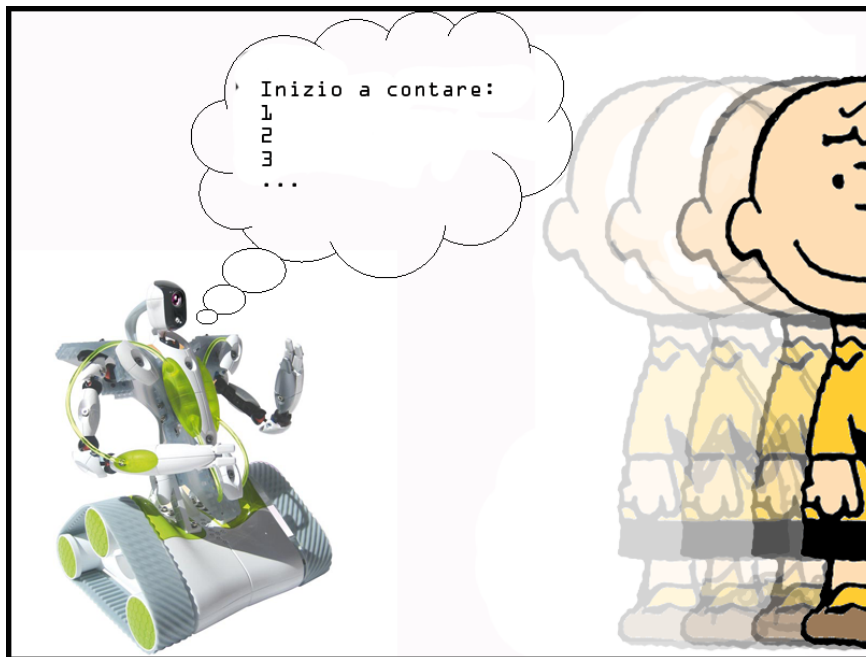


Figura 7.3: Spykee inizia la conta e Charlie scappa a nascondersi.

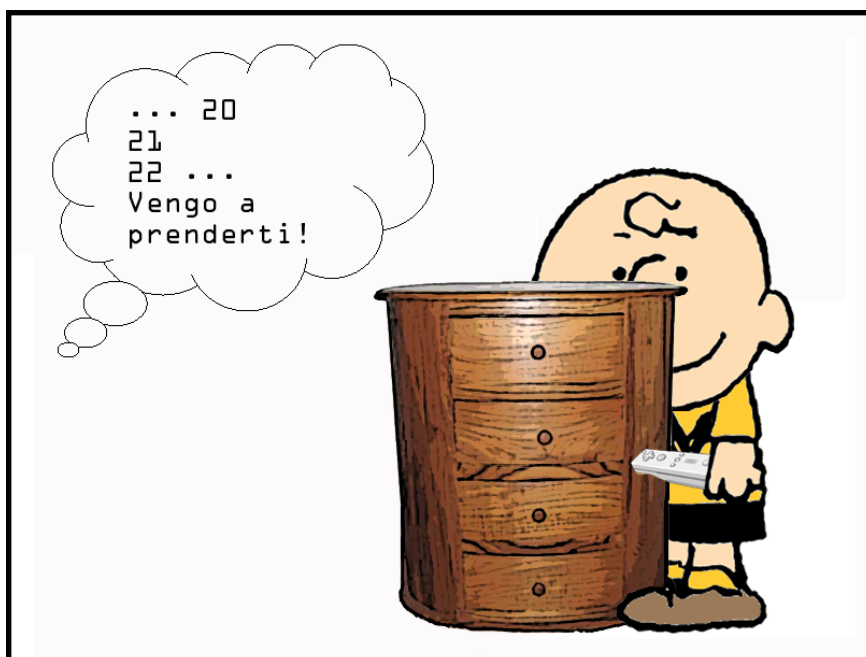


Figura 7.4: Spykee finisce la conta e inizia a cercare.



Figura 7.5: Charlie preme B per sparare a Spykee.

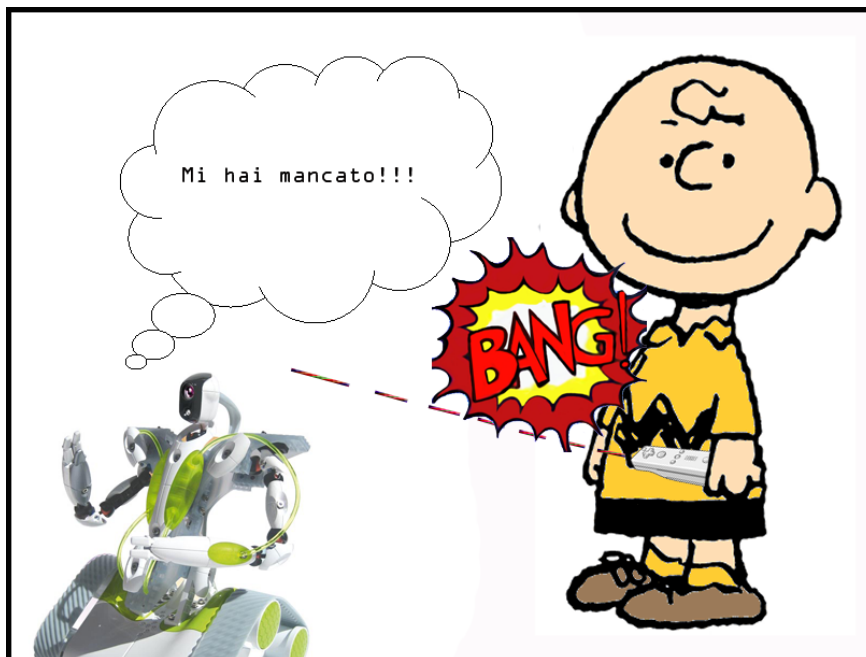


Figura 7.6: il colpo di Charlie va a vuoto.

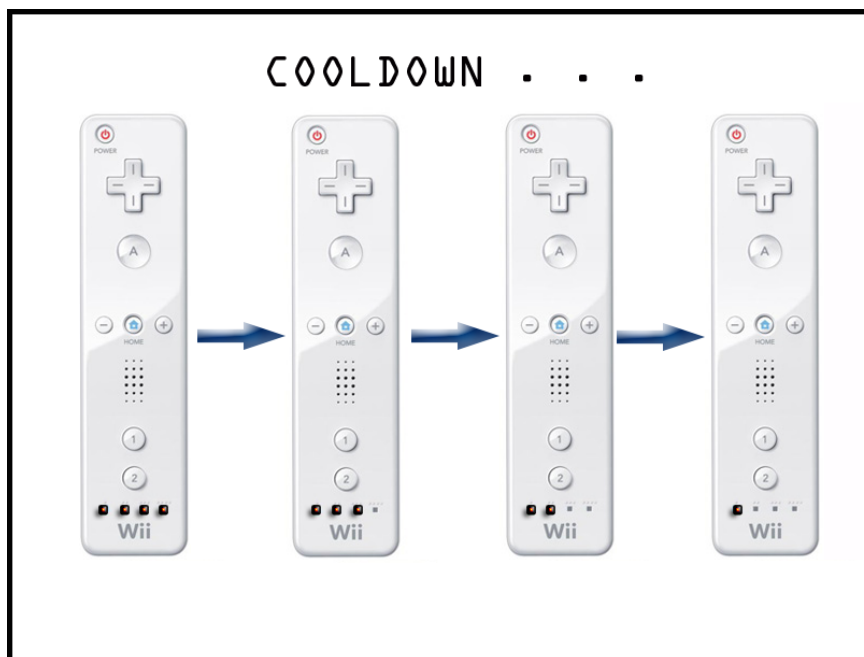


Figura 7.7: Si attiva il cooldown.

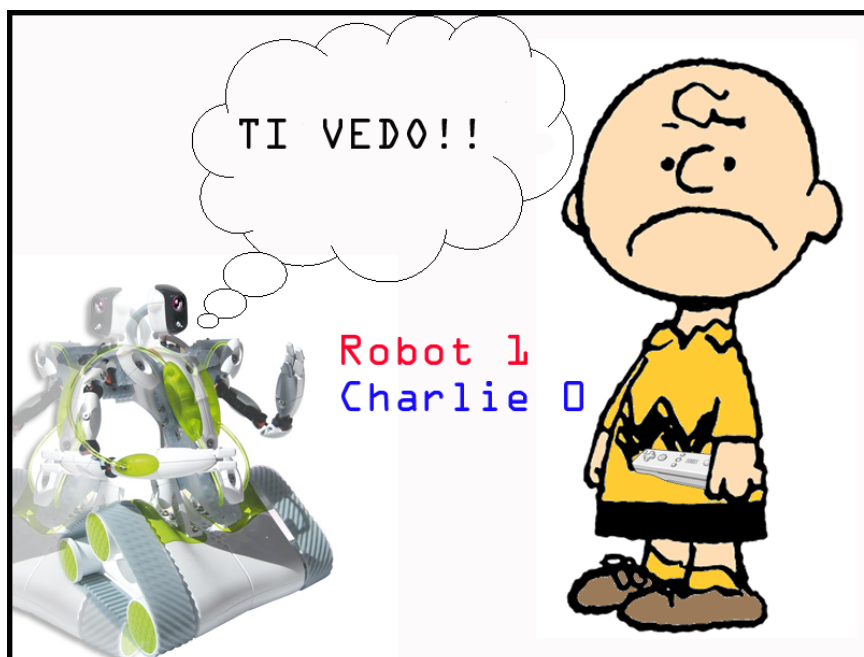


Figura 7.8: Spykee si gira, vede Charlie e mette a segno un punto a suo favore.

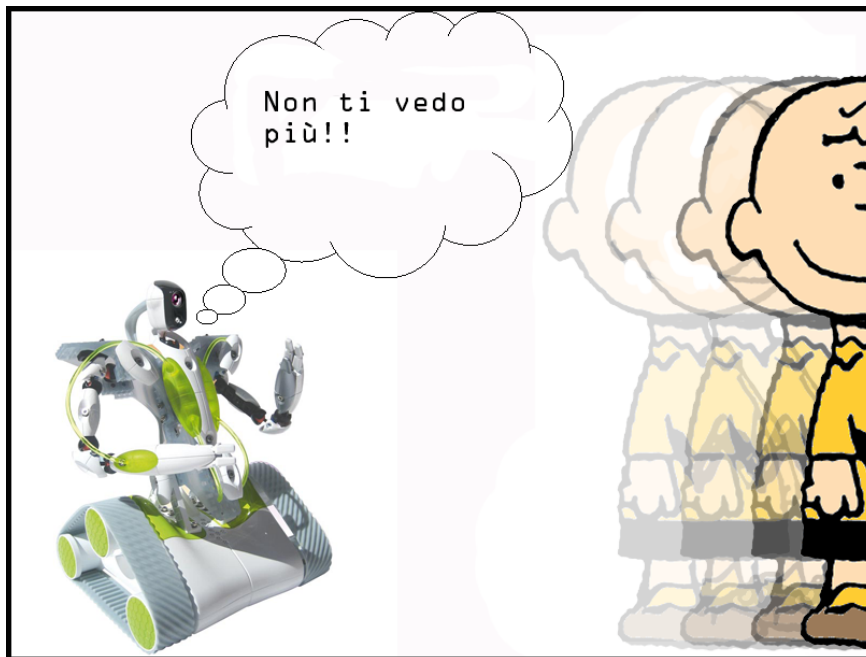


Figura 7.9: Charlie scappa, uscendo dal campo visivo di Spykee.



Figura 7.10: Charlie si nasconde per non farsi individuare.



Figura 7.11: Charlie preme B per sparare a Spykee.

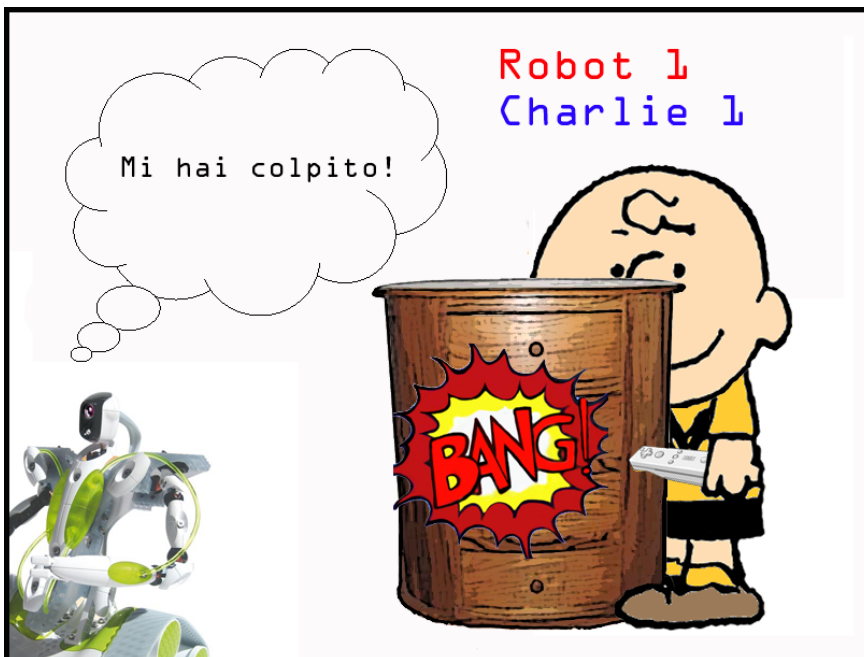


Figura 7.12: Charlie da posizione nascosta colpisce Spykee e mette a segno un punto.

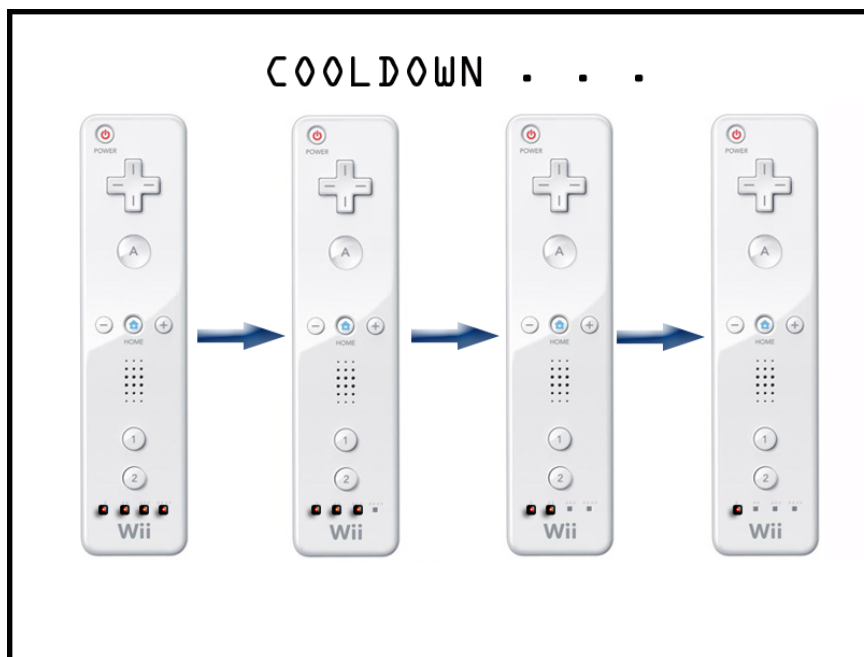


Figura 7.13: Si attiva il cooldown.



Figura 7.14: Spykee a causa del colpo subito rimane stordito.



Figura 7.15: Charlie spara premendo B.

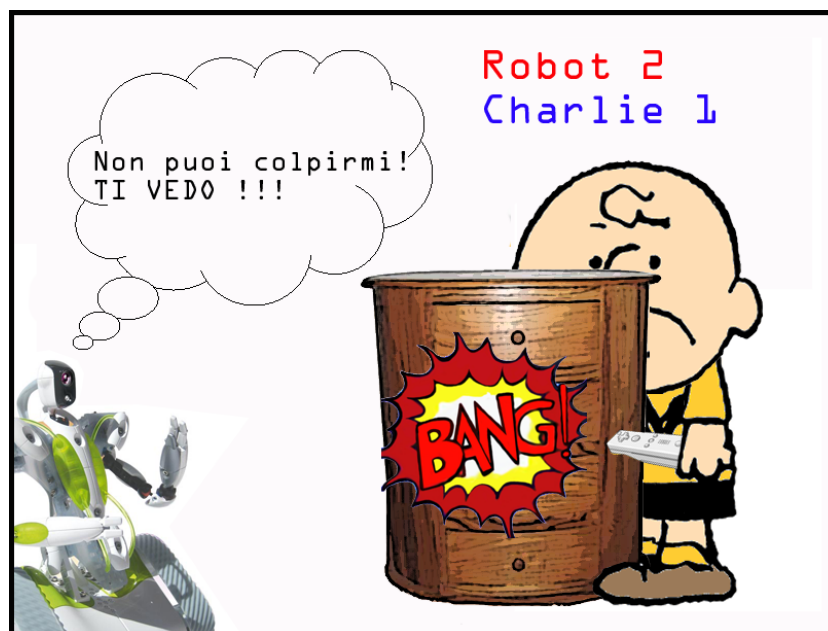


Figura 7.16: Charlie spara trovandosi difronte a Spykee, il colpo non è valido e quindi è il robot a fare punto dato che il palyer è nel suo campo visivo.

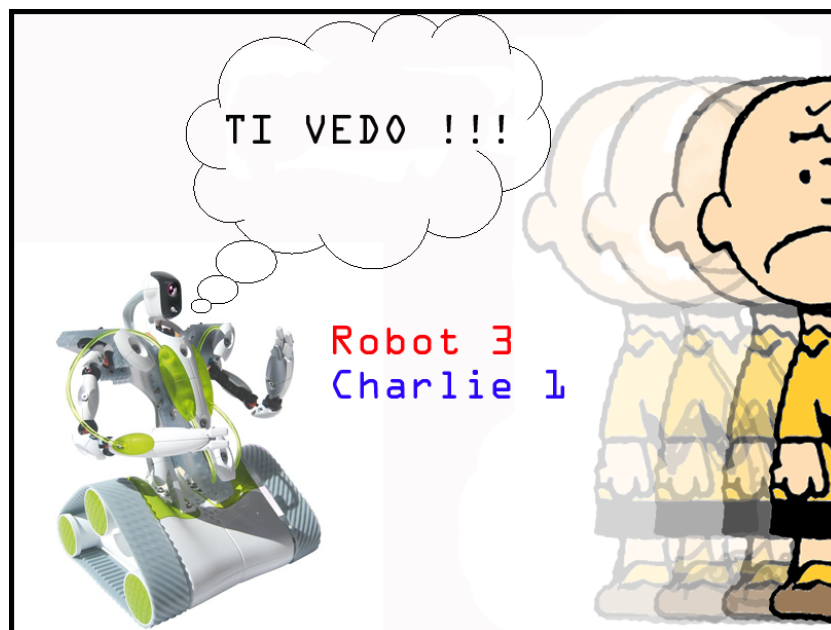


Figura 7.17: Charlie tenta di scappare, ma rimane lo stesso nel campo visivo di spykee consentendogli così di mettere a segno il punto decisivo.

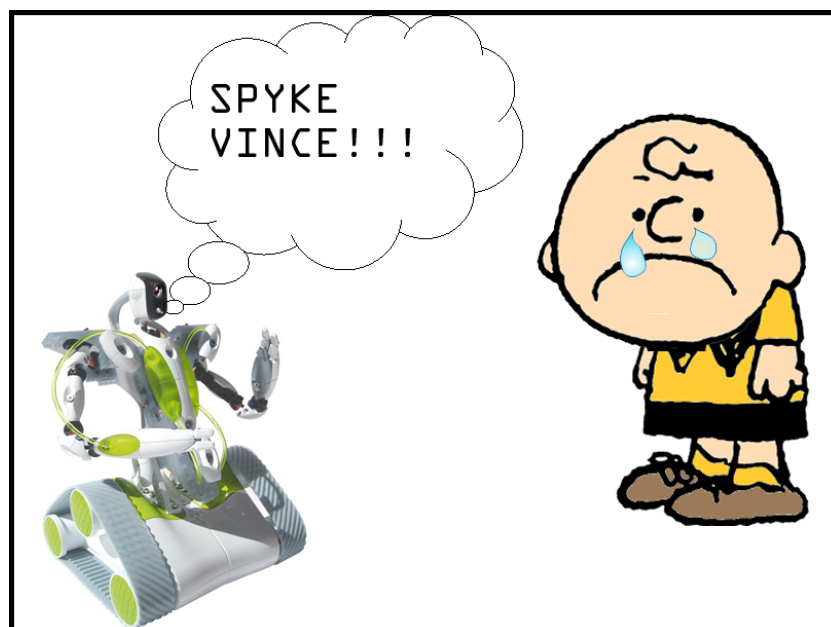


Figura 7.18: Charlie è triste perchè Spykee ha vinto.

7.3 Scenario: vittoria Player

Il player, un bambino di età compresa tra i 6 e i 10 anni, avvia l'applicazione e seguendo i messaggi vocali configura, tramite il wiimote, i parametri relativi alla partita (durata, cooldown, modalità punteggio, ecc..) impostando il numero di punti necessari alla vittoria uguale a 3 e impostando il *discrete mode*. Avviato il gioco il robot concede un intervallo di tempo, settato in precedenza, al giocatore per allontanarsi e nascondersi (conta). Terminata la conta il robot inizia la ricerca del player. L'utente spara, il colpo va a segno, si incrementa il punteggio del player e si attiva il cooldown dell'arma. Finito il periodo di 'stordimento' il robot rinizia a cercare. Il player da posizione nascosta spara e mette a segno un altro colpo. Il robot, finito il tempo in cui rimane inattivo a causa dello stordimento, compie una rotazione per avvistare il player che nel frattempo è andato a nascondersi. Da questa posizione il player tenta di colpire nuovamente il nemico incurante della sua posizione frontale, esponendosi così alla visuale del robot, che lo individua e incrementa il suo punteggio. Il player infine da una posizione strategica esplosivo il colpo definitivo che gli permette così di vincere la partita.

7.4 Scenario: esaurimento tempo limite partita

Il player, un bambino di età compresa tra i 6 e i 10 anni, avvia l'applicazione e seguendo i messaggi vocali configura, tramite il wiimote, i parametri relativi alla partita (durata, cooldown, modalità punteggio, ecc..). Avviato il gioco il robot concede un intervallo di tempo, settato in precedenza, al giocatore per allontanarsi e nascondersi (conta). Terminata la conta il robot inizia la ricerca del player. L'utente spara, il colpo va a vuoto e si attiva il cooldown dell'arma. Il robot compie un giro completo su se stesso, individua il giocatore presente nelle vicinanze, mettendo così a segno un punto a suo favore. Il robot annuncia l'avvistamento del player che quindi fugge e torna a nascondersi. A questo punto Spykee rinizia a cercare. Il player da posizione nascosta spara alle spalle e mette a segno il colpo. Il robot, finito il tempo di 'stordimento', compie una rotazione per avvistare il player che nel frattempo è andato a nascondersi. Da questa posizione il player tenta di col-

pire nuovamente il nemico incurante della sua posizione frontale, esponendosi così alla visuale del robot, che lo individua e aggiorna il punteggio. A questo punto termina il tempo limite della partita e la vittoria viene assegnata a spykee poiché ha un punteggio superiore a quello del player.

7.5 Scenario: infrazione regole

Il player, un bambino di età compresa tra i 6 e i 10 anni, avvia l'applicazione e seguendo i messaggi vocali configura, tramite il wiimote, i parametri relativi alla partita (durata, cooldown, modalità punteggio, ecc.). Avviato il gioco il robot concede un intervallo di tempo, settato in precedenza, al giocatore per allontanarsi e nascondersi (conta). Terminata la conta il robot inizia la ricerca del player. L'utente rimane nascosto senza sapere neanche un colpo, raggiunto un tempo prestabilito il robot esclama 'Spykee vince!' poiché il player è rimasto inattivo troppo a lungo.

Capitolo 8

Prototipo

Capitolo 9

Conclusioni

Bibliografia

- [1] Matteo Botta Alberto Bottinelli. *Where WiiAre: sistema di localizzazione basato sul dispositivo Wii Remote.*
- [2] Ben Chen Antonio Bianchi. *Sviluppo di un gioco tramite l'interazione fra un robot ed il controller Wii Remote.*
- [3] Motej. Motej - a wiimote library for java.
- [4] Wiibrew. Accelerometr.
- [5] Wiibrew. Bluetooth communication.
- [6] Wiibrew. Ir camera.
- [7] Wikipedia. Direction of arrival.

Appendice A

Breve descrizione del contenuto del CD/DVD