



VuCube



A.A. 2016/2017

Filippo Mariani, 282484

Giorgio Mazza, 283286



Indice:

Copertina	pg.1
Indice	pg.2
1. Sommario del progetto	pg.3
2. Obiettivo del progetto	pg.3
3. Analisi del progetto	pg.4
3.1. Glossario	pg.4
4. Diagrammi UML	pg.5
4.1. Diagramma di Classe	pg.5
4.2. Diagramma Use Case	pg.6
4.3. Diagramma di Sequenza	pg.6
4.4. Diagramma di Stato	pg.7
4.5. Diagramma di Attività	pg.7
5. Codice sorgente	pg.8
6. Casi di Test Funzionali.....	pg.8
7. Design Pattern	pg.9

1. Sommario del progetto

VuCube è un'applicativo basato sulla Realtà Aumentata (AR), mediante l'ausilio del framework Vuforia e dell'engine Unity.

L'applicativo consiste in una figura tridimensionale dotata di features multimediali e web, proiettata in realtà aumentata tramite un marker.

2. Obiettivo del progetto

Il progetto si pone come obiettivo quello di mostrare, mediante una demo, le potenzialità della tecnologia AR, fornendo all'utente un'esperienza multimediale innovativa.

In oltre, alla base del progetto, vi è la volontà da parte degli studenti di introdursi nel campo dello sviluppo di applicativi mobile e della realtà aumentata.



3. Analisi del progetto

Requisiti di sviluppo: per sviluppare il software si è ritenuto opportuno utilizzare:

- l'engine per lo sviluppo di app 3D Unity
- l'sdk per lo sviluppo in realtà aumentata Vuforia
- un dispositivo/emulatore Android.

Metodologia di sviluppo: Si è deciso di utilizzare il modello a sviluppo incrementale, in quanto il metodo di sviluppo più consono al contesto, e permette la modifica in tempo reale degli obiettivi.

Requisiti grafici: il software vuole visualizzare, tramite il tracking di un ImageTarget da parte dell'ARCamera, un cubo sospeso fornito di features multimediali quali:

- Video Player per la riproduzione video.
- visualizzatore di immagine
- Virtual Button di controllo degli eventi

Requisiti interfaccia: il software vuole consentire:

- la gestione video
- la visualizzazione di pagine web sul browser predefinito

Il tutto tramite l'utilizzo di Virtual button

3.1 Glossario

AR: Acronimo di Realtà Aumentata (Augmented Reality)

Unity: è un programma di packaging integrato multiplatforma per la creazione di videogiochi 3D o altri contenuti interattivi, quali visualizzazioni architettoniche o animazioni 3D in tempo reale.

Vuforia: è un SDK (Software Development Kit) di Realtà Aumentata per devices mobili che permette lo sviluppo di applicazioni in Realtà Aumentata.

ImageTarget: è un simbolo che l'ARCamera riconosce come base per effettuare il tracking dell'oggetto in realtà aumentata.

ARCamera: componente software integrato in Vuforia, che permette la visualizzazione degli oggetti in realtà aumentata.

Virtual Button: bottoni virtuali posizionabili sul target, i quali attivano script all'accadere di determinati eventi.

Video Playback: funzionalità di Vuforia che permette la riproduzione video sull'area del target

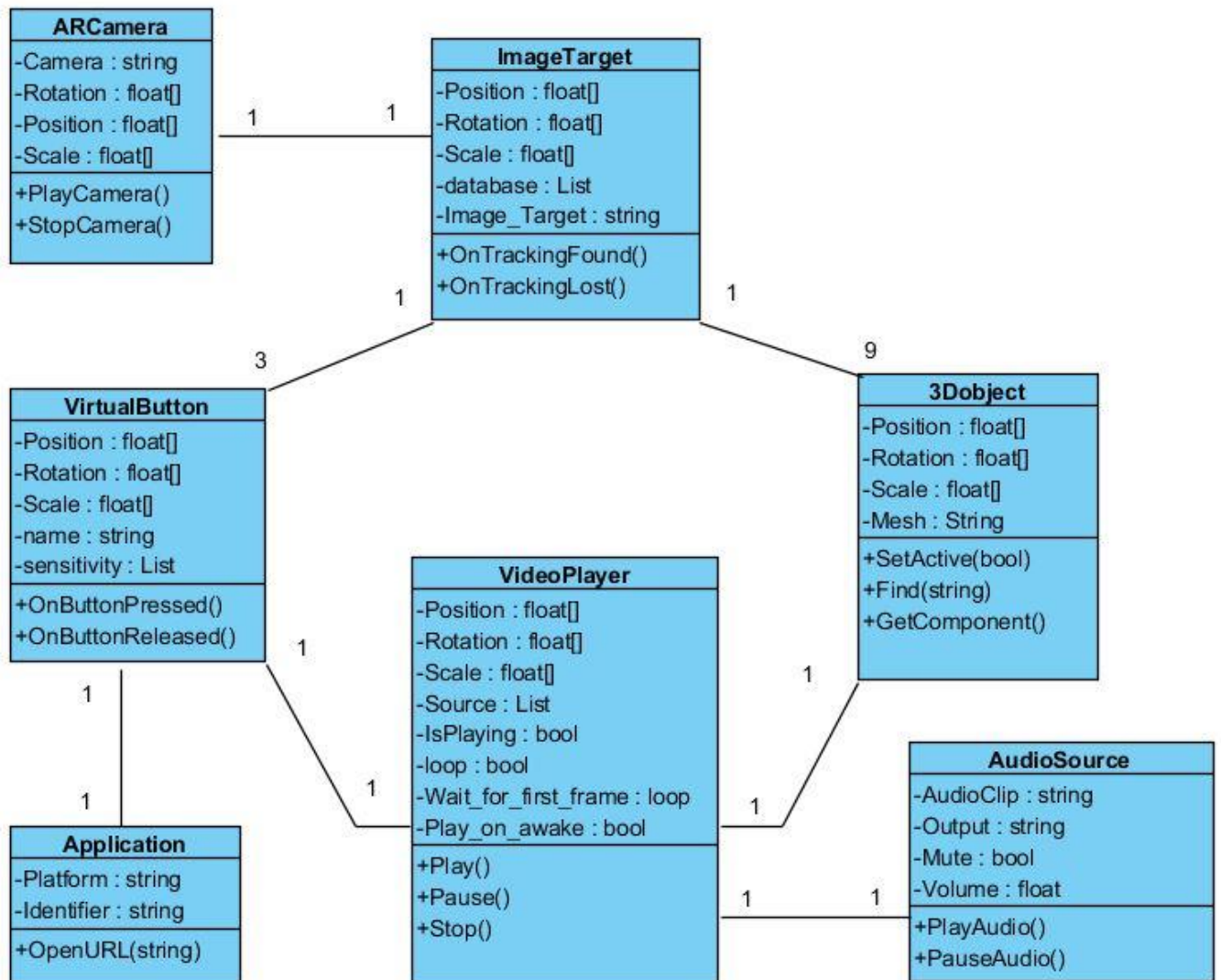
Script: è un insieme di istruzioni scritte con specifici linguaggi di scripting, associati all'esecuzione di un programma.

Emulatore: componente che replica le funzioni di un sistema su un secondo differente dal primo.

4. Diagrammi UML

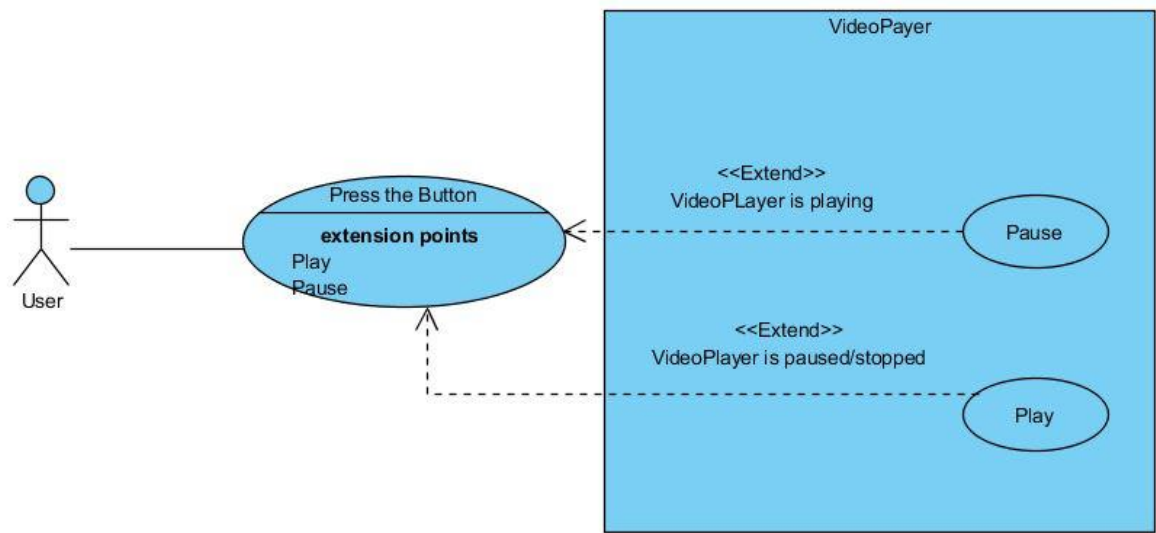
Di seguito i diagrammi relativi al progetto ed alle sue parti:

4.1. Diagramma di Classe



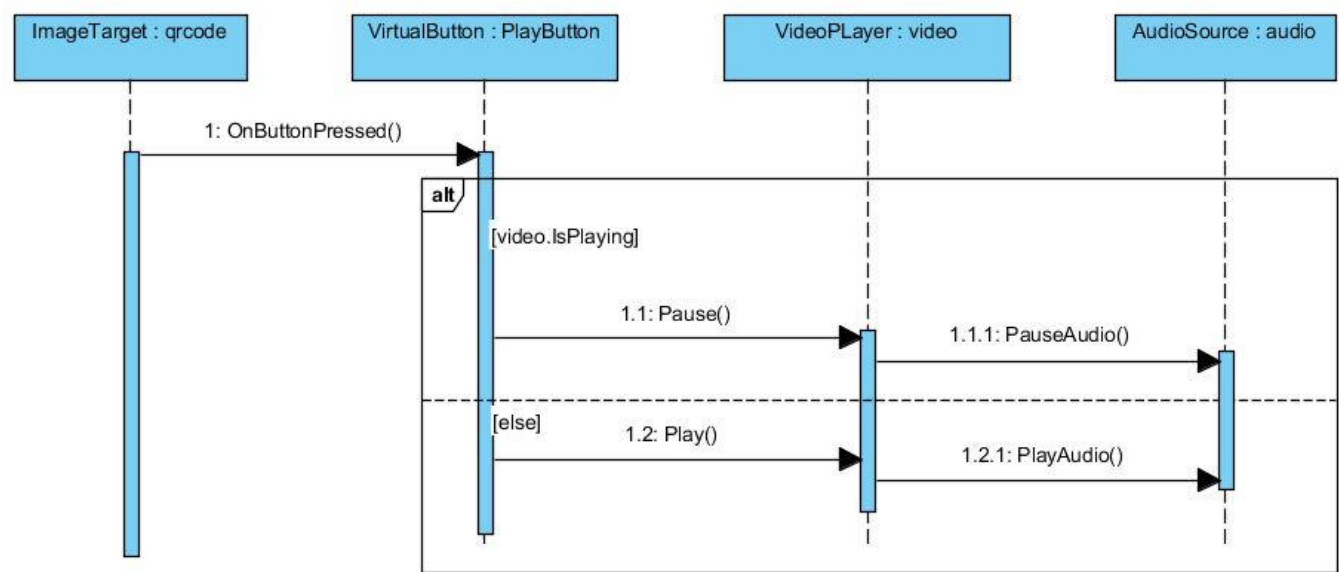
4.2. Diagramma Use Case

Diagramma del caso d'uso "Video Player"



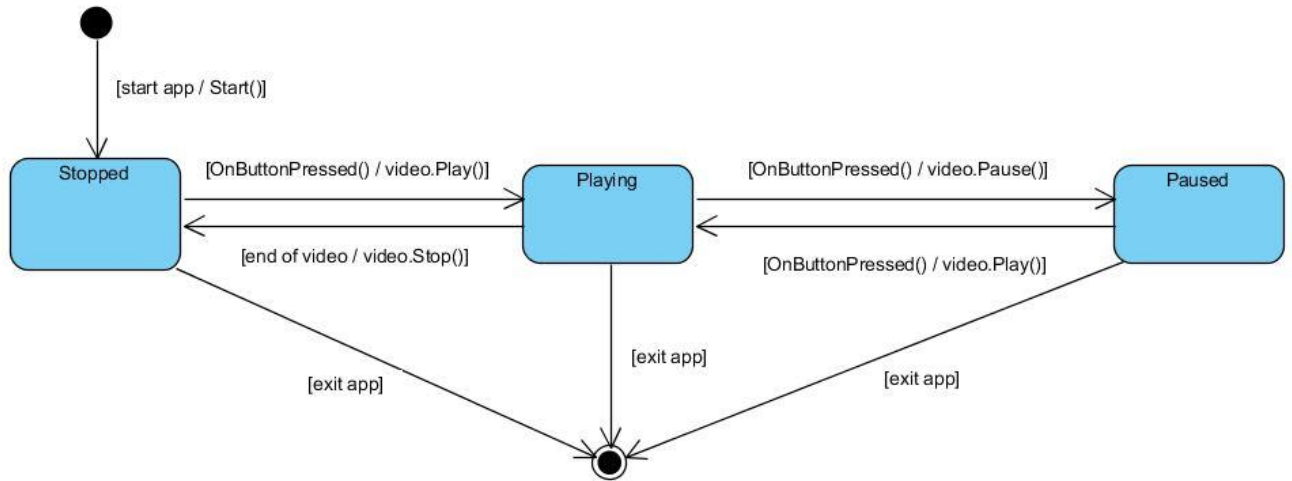
4.3. Diagramma di Sequenza

Diagramma di sequenza riferito a "Video Player"



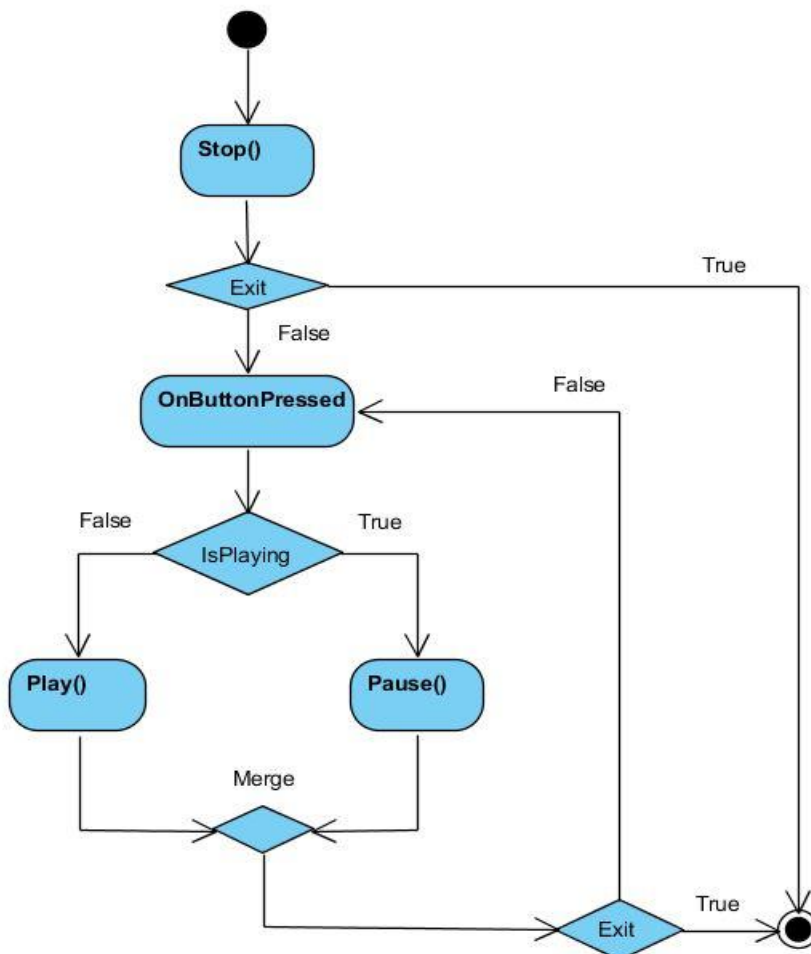
4.4. Diagramma di Stato

Diagramma di stato di "Video Player"



4.5. Diagramma di Attività

Diagramma di attività di "VideoPlayer"



5. Codice sorgente

Il codice sorgente è in allegato nel file zip, ed è in oltre disponibile al link

<https://github.com/Disorganizzazione/VuCube>

6. Casi di Test Funzionali

I seguenti sono i casi di test relativi al riconoscimento da parte dell'ARCamera dell'Image Target; I valori ottenuti possono variare in base alla qualità della fotocamera e dell'illuminazione ambientale.

La distanza si riferisce a quella che intercorre tra il dispositivo e il target, i gradi si riferiscono all'angolazione del dispositivo rispetto all'asse perpendicolare al piano del target.

- Casi di test funzionali relativi a "Rilevamento dell'Image Target"

Validità	Numero	Target visibile	Gradi	Distanza
V	1	Sì	37°	20 cm
NV	2	No	/	/
NV	3	Sì	90°	20 cm
NV	4	Sì	37°	5 cm
NV	5	Sì	37°	100 cm

E relativi casi estremi

	Accettati	Non accettati
Gradi	(0°, 45°)	(46°, 90°)
Distanza	(13 cm, 70 cm)	(0 cm, 12 cm) U (71 cm, MAX_DIST)

- Casi di test funzionali relativi a "Mantenimento tracking dell'Image Target" (una volta rilevato)

Validità	Numero	Target visibile	Gradi	Distanza
V	1	Sì	60°	90 cm
NV	2	No	/	/
NV	3	Sì	90°	90 cm
NV	4	Sì	60°	1 cm
NV	5	Sì	60°	200 cm

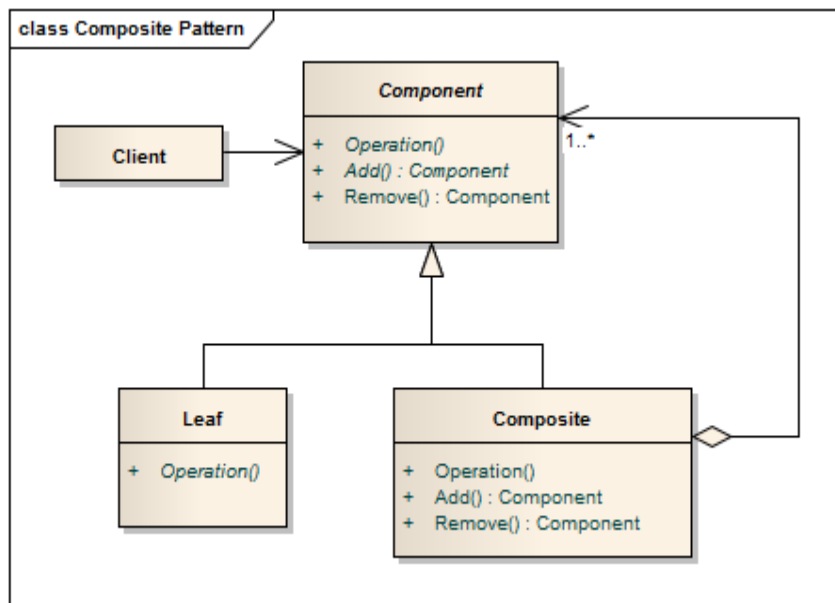
E relativi casi estremi

	Accettati	Non accettati
Gradi	(0°, 85°)	(86°, 90°)
Distanza	(3 cm, 160 cm)	(0 cm, 2 cm) U (161 cm, MAX_DIST)

7. Design Pattern

Si potrebbe applicare al progetto, in vista di sviluppi futuri, il design pattern “**Composite**”.

Quest’ultimo consiste in un pattern strutturale che tratta più oggetti come fossero l’istanza di un solo oggetto, organizzandoli ad albero. Questo pattern si adatta in modo particolare alla struttura organizzativa dei GameObject di Unity, che possono essere “attaccati” l’uno nell’altro. Gli oggetti della gerarchia possono quindi essere o oggetti semplici, o oggetti contenitori che a loro volta sono composti da altri oggetti semplici/contenitori, ma in ogni caso il sistema vedrà il gruppo di oggetti come un singolo oggetto, semplificando il lavoro del Client.



1. **Client**: chi invoca l’operazione di interesse
2. **Component**: definisce l’interfaccia degli oggetti della composizione.
3. **Leaf**: l’oggetto foglia della composizione. Non ha figli.
4. **Composite**: definisce il comportamento degli oggetti usati come contenitori e detiene il riferimento ai componenti “figli”.