DipMap

Mappa indoor location tramite QR code

progetto a cura di
Massetti Matteo 288646
anno accademico 2017/2018

INDICE

Abstract del progetto

DipMap offre all'utente la possibilità di conoscere il dipartimento in cui si trova. Attraverso la scannerizzazione di QR code vengono fornite informazioni sulla stanza e sul piano nel quale il codice è posizionato, in modo da potersi orientare nell'edificio.

Inoltre offre ad eventuali altri programmatori la possibilità di parametrizzare l'app, permettendo quindi di rappresentare diversi luoghi senza dover modificare la struttura del codice.

Obiettivo del progetto

L'obiettivo principale era la creazione un'app che permettesse la localizzazione dell'utente in una mappa indoor attraverso la scannerizzazione di un QR code. L'app è stata modellata usando i luoghi del dipartimento di matematica e informatica (unipg), ma è progettata in modo tale da poter essere adattata ad ogni altro dipartimento o edificio.

Analisi del progetto

Campo di applicazione del prodotto

Il prodotto finale deve rispondere al bisogno di avere una mappa indoor a portata di smartphone, anche di quegli edifici visti per la prima volta, in modo da evitare di perdersi o di sprecare tempo per cercare una determinata stanza o aula.

Acquisizione delle competenze

Per quanto riguarda la documentazione preesistente, è necessaria l'acquisizione delle planimetrie dell'edificio. Per la dimostrazione del funzionamento dell'app sono state utilizzate al posto delle planimetrie originali delle fotografie di quest'ultime (data la loro difficile reperibilità), qualità potrebbe dunque risultare inferiore.

Glossario

Qr Code: abbreviazione di Quick Response, è un codice a barre bidimensionale, composto da moduli neri disposti all'interno di uno schema bianco a forma quadrata. Può memorizzare fino a 4.296 caratteri alfanumerici.
API: rappresentano un'interfaccia aperta di un software, ovvero una particolare interfaccia che librerie, software o piattaforme possono usare per interagire con un programma.
Scanner: periferica o software in grado di acquisire una superfice e interpretarla come un insieme di pixel.
Android Studio: ambiente di sviluppo integrato per la creazione di applicazioni Android, usa principalmente il linguaggio di programmazione Java
Canvas: classe che estende la classe “draw” , che utilizzando una Bitmap e altri parametri può disegnare su di essa.
CameraSource.Builder: API di Google fornito per android che permette di creare e configurare un'istanza che permetta l'utilizzo della videocamera associata.
BarcodeDetector.Builder: API di Google fornito per android che permette di creare e configurare un'istanza che cerca barcode in tutti i formati supportati.
Raw: cartella all'interno del programma in cui posso essere salvati qualsiasi tipo di file utili all'esecuzione del programma.
ImageView: oggetto relativo alla programmazione in Android Studio che può contenere immagini

Requisiti

Dalla ricerca e dall'analisi dei requisiti, ne emergono due fondamentali:
[R1] Permettere la scannerizzazione di un codice qr e da questo caricare la relativa mappa
[R2] Permettere la parametrizzazione dell'applicazione modificando i file dei luoghi e delle mappe

Il caso d'uso relativo a “richiesta del servizio di localizzazione” è il seguente:

Progettazione Architetturale

Ci si è subito posti il problema su quale ambiente di sviluppo utilizzare, data la sua completezza la scelta è ricaduta su Android Studio.
Il primo componente di cui si aveva bisogno era lo scanner del qr code, invece di crearlo da zero però si è optato per usare un codice già esistente (scritto in java e kotlin), che inoltre sfrutta delle Application Programming Interface offerte da Google (come ad esempio CameraSource.Builder e BarcodeDetector.Builder).
Le altre funzioni necessarie all'app sono state quindi costruite intorno allo scanner.
Per rendere l'app parametrizzabile si è pensato di salvare le immagini relative alle mappe e il file di testo relativo alle coordinate nella cartella “raw”, cosicché con solo la modifica di questa cartella si rende possbile rappresentare un altro edificio.

Diagrammi UML

I diagrammi riportati di seguito sono stati realizzati con l'ausilio di “gemmymodel.com”.

Diagramma di classe completo

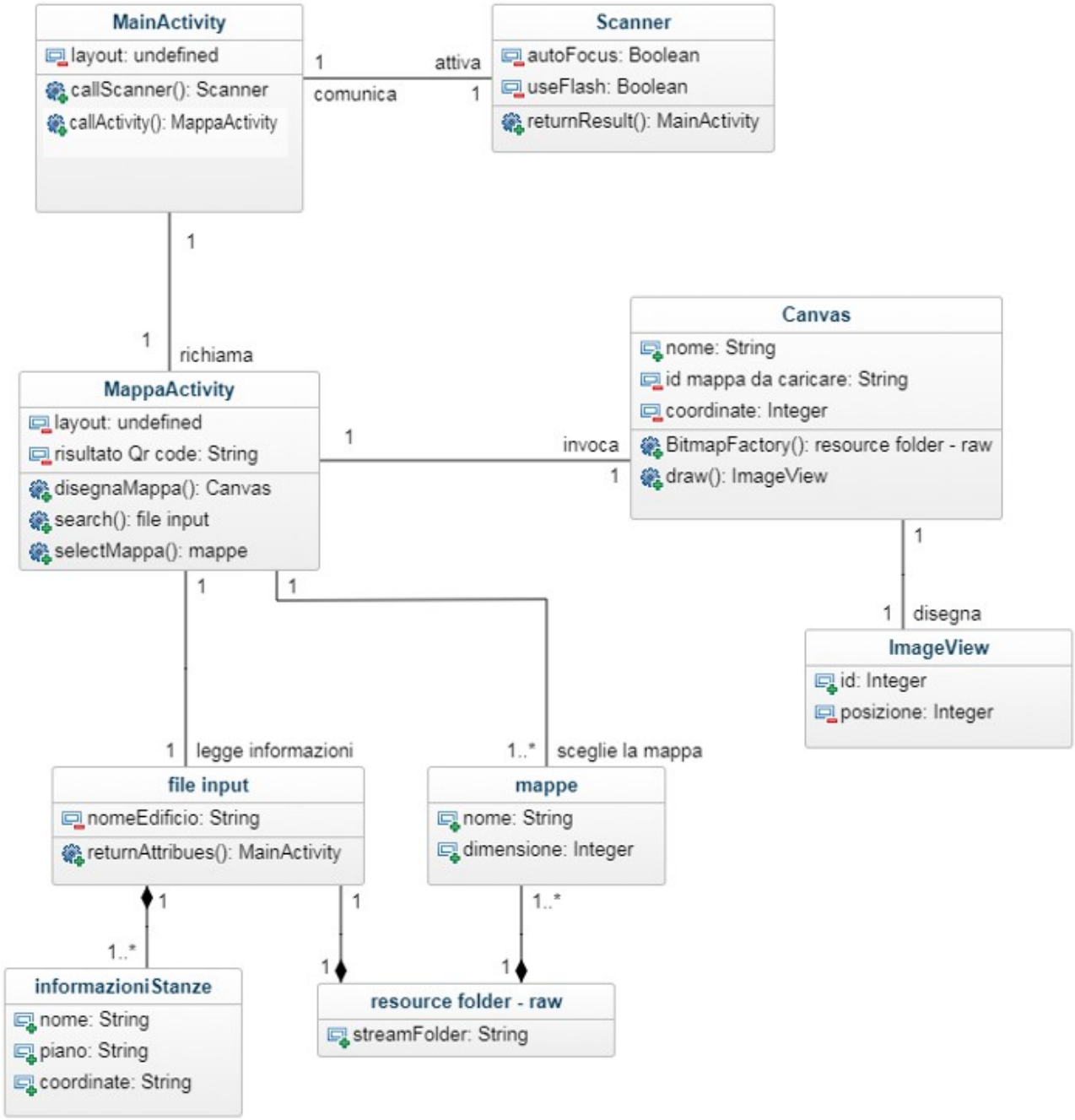


Diagramma di sequenza riferito al caso d'uso descritto in precendenza:

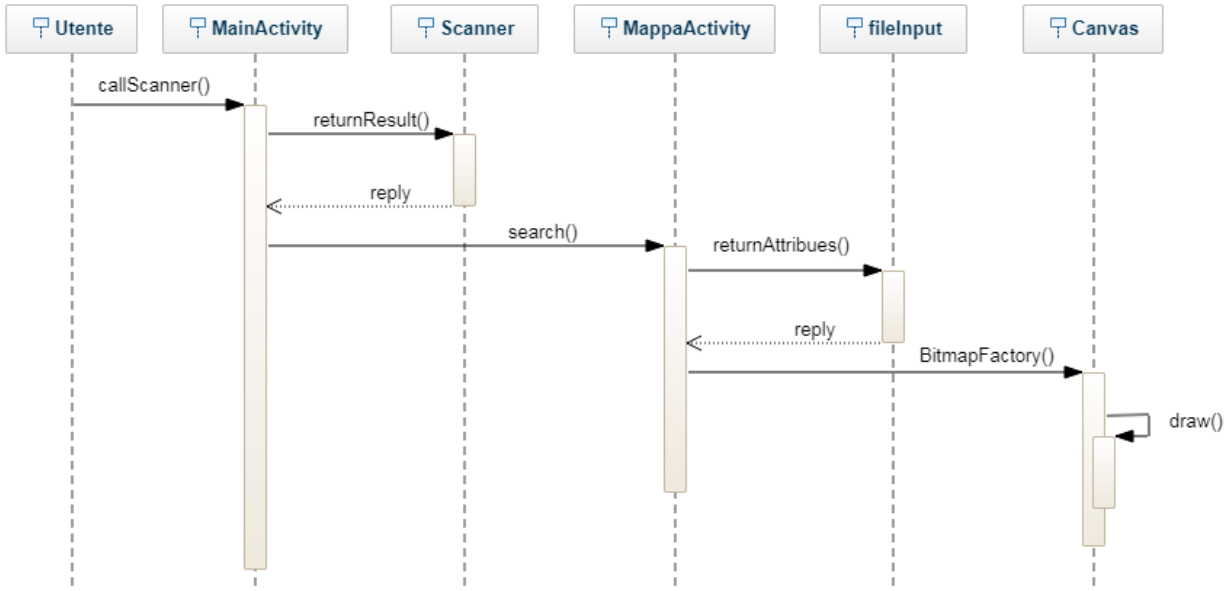


Diagramma di collaborazione riferito al caso d'uso descritto in precedenza:

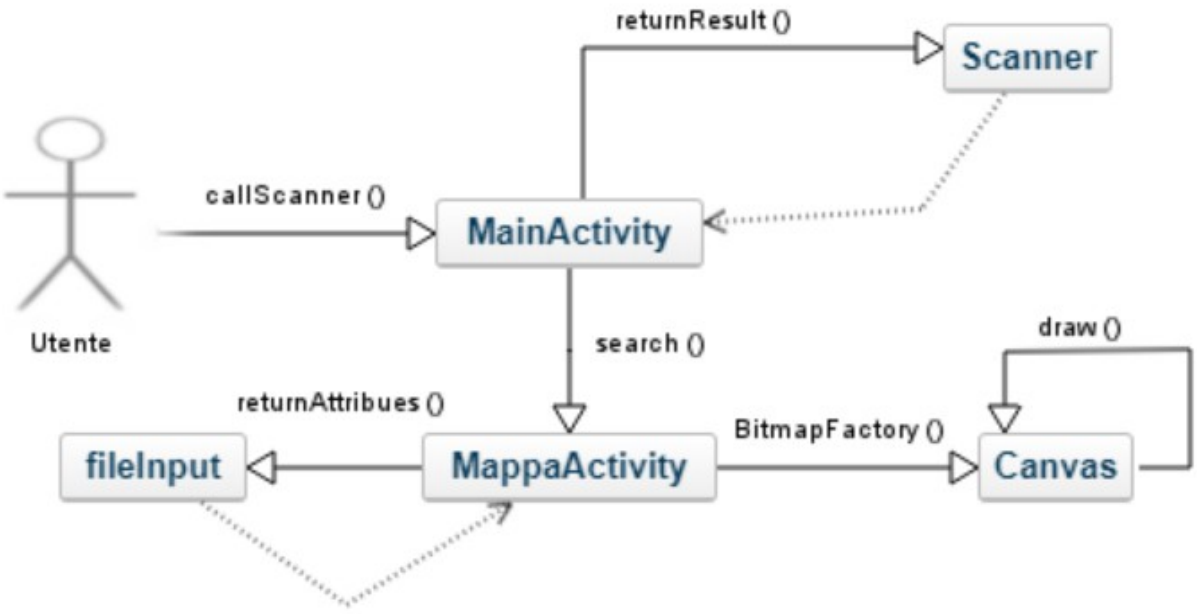
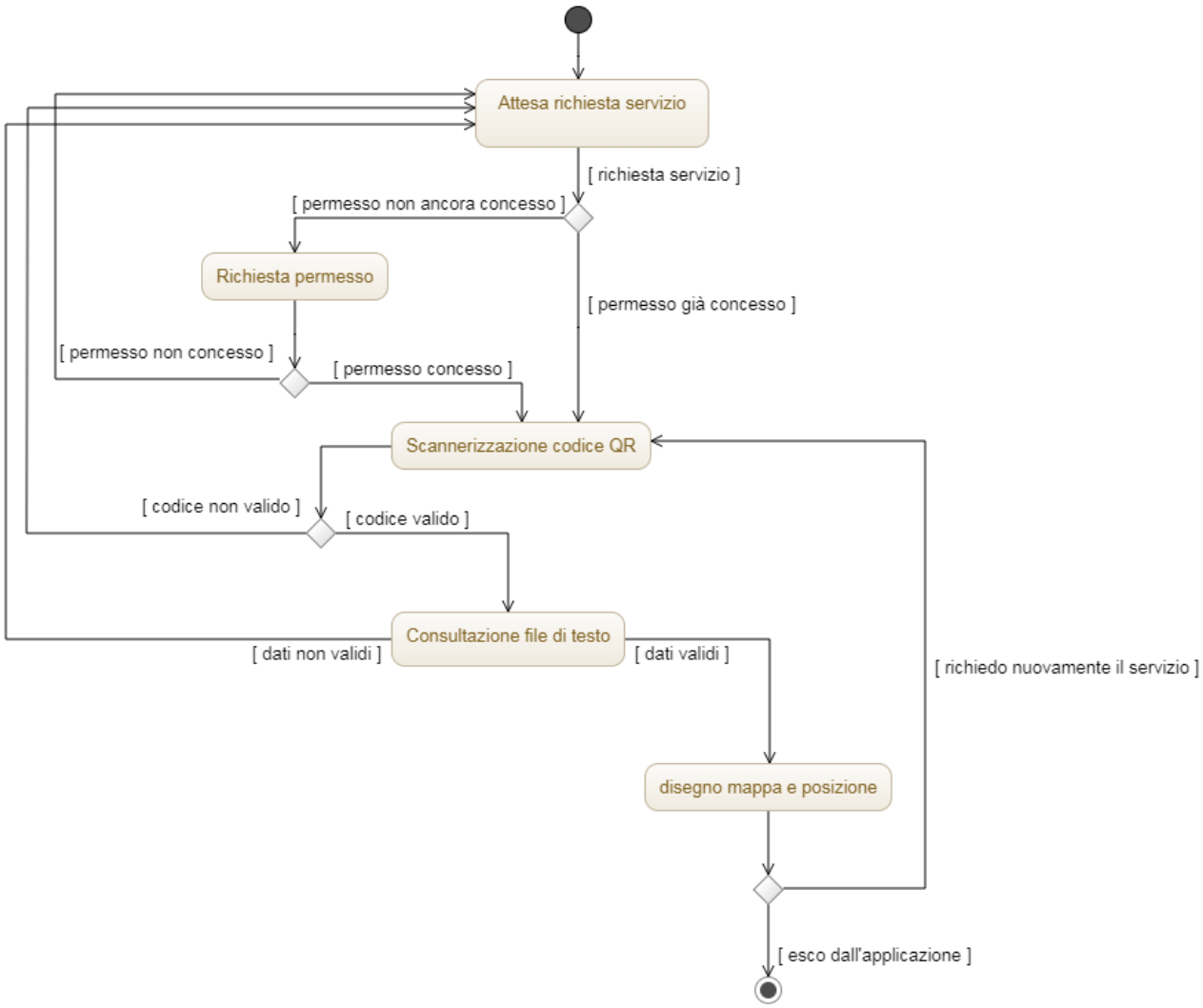


Diagramma di stato relativo all'imageView contenente la mappa e il punto:

Diagramma di attività relativo alla richiesta del servizio di localizzazione da parte dell'utente:



Implementazione

Dal momento che la parte relativa allo scanner era già stata implementata usando un codice esistente, era necessaria solo l'implementazione del codice relativo all'interrogazione del file di input per ottenere le informazioni e al caricamento dell'immagine (con il relativo puntino) corrispondente.
Il file di input è un file di testo in cui la prima riga indica il nome dell'edificio (nell'esempio il nome è “dipartimento di matematica e informatica”), mentre dalla seconda in poi sono registrate le informazioni delle stanze in questo ordine: nome della stanza, coordinata x, coordinata y, raggio del punto da disegnare, piano, un valore che indica se il piano è interrato oppure no. Quest'ultimo parametro si è reso necessario dato che durante la conversione da stringa a intero il “-” si perdeva.
Una volta quindi ottenuta la stringa dal Qr code la si va a ricercare nel file di input e da di prendono le relative informazioni, con queste usando la classe Canvas si disegna a schermo la mappa relativa al piano con il punto relativo alla posizione.
Nel caso in cui si dovesse leggere uno QR code errato o che contiene un nome non presente nella lista, o ancora se nel file di input vengono inseriti valori non convertibili in interi il programma restituisce un errore e permette una nuova scannerizzazione.

Design Pattern

Dal momento che ci sono diverse classi che devono avere un unica istanza ciascuna, il Design Pattern Singleton poteva trovare spazio nell'applicazione.
Singleton infatti risponde perfettamente alla necessità, assicurando l'esistenza di una ed una sola istanza di una classe in tutta l'applicazione.
Conseguentemente questo desing pattern permette anche il controllo completo di come e quando i client accedono all'interfaccia, evita il proliferare di variabili globali e può permettere un numero massimo e preciso di istanze attive.

Casi di test funzionali

Nel progetto sono presenti due aspetti dai quali si possono ricavare casi prova significativi, ovvero la stringa letta dal QR code e le stringhe lette dal fileInput.
I casi di test riportati di seguito permettono una copertura comlpeta su tutte i possibili valori che il fileInput può contenere.
Nello specifico i parametri per ogni stanza sono:

-coordinata_x: numero minore di 1500

-coordinata_y: numero minore di 1500

-piano: numero minore o uguale a 2

-negativo: può assumere solamente 0 (valido con tutti i piani) e 1 (valido con solo il piano 1)

-raggio: deve essere un numero

inoltre ogni campo deve non essere nullo (indicato con /)

NB non sono considerati i casi in cui i numeri sono negativi, dato che nella conversione da stringa a intero il "-" si perde.

n°	coordinata_x	coordinata_y	Piano	Negativo	Raggio	Risultato test
1	600	700	1	1	12	Test valido
2	1700*	550	2	0	10	Test non valido
3	Abc*	0	0	0	20	Test non valido
4	/*	1	0	0	15	Test non valido
5	550	2000*	1	0	15	Test non valido
6	200	Duecento*	2	0	5	Test non valido
7	300	/*	1	1	250	Test non valido
8	150	70	3*	0	7	Test non valido
9	20	20	Piano_terra*	0	15	Test non valido
10	12	13	/*	1	12	Test non valido
11	15	1400	0*	1*	12	Test non valido
12	1500	1500	0	2*	13	Test non valido
13	0	0	1	N*	5	Test non valido
14	1500	0	2	/*	1000	Test non valido
15	25	25	0	0	A*	Test non valido
16	37	35	1	1	/*	Test non valido

Il parametro o i parametri che invalidano il test sono contrassegnati con *

QR Code

Codice posto sulla porta di ingresso inferiore dell'aula A0.

Codice posto sulla porta di ingresso superiore dell'aula A0.

Codice posto sulla porta dell'aula B1.

Codice contenete una stringa non presente nel file di input

Codice contenente una stringa presente nel file di input ma di cui si hanno informazioni errate.