



UNIVERSITA' DEGLI STUDI DI
NAPOLI FEDERICO II

Scuola Politecnica e delle Scienze di Base
Corso di Laurea Magistrale in Ingegneria Informatica

Tesi di Laurea Magistrale in Ingegneria del Software II

Sviluppo di un sistema per la scoperta, fruizione e georeferenziazione di immagini del passato

Anno Accademico 2016/17

relatore
Ch.mo Prof. Porfirio Tramontana

Davide Russo
Matr. M63000412

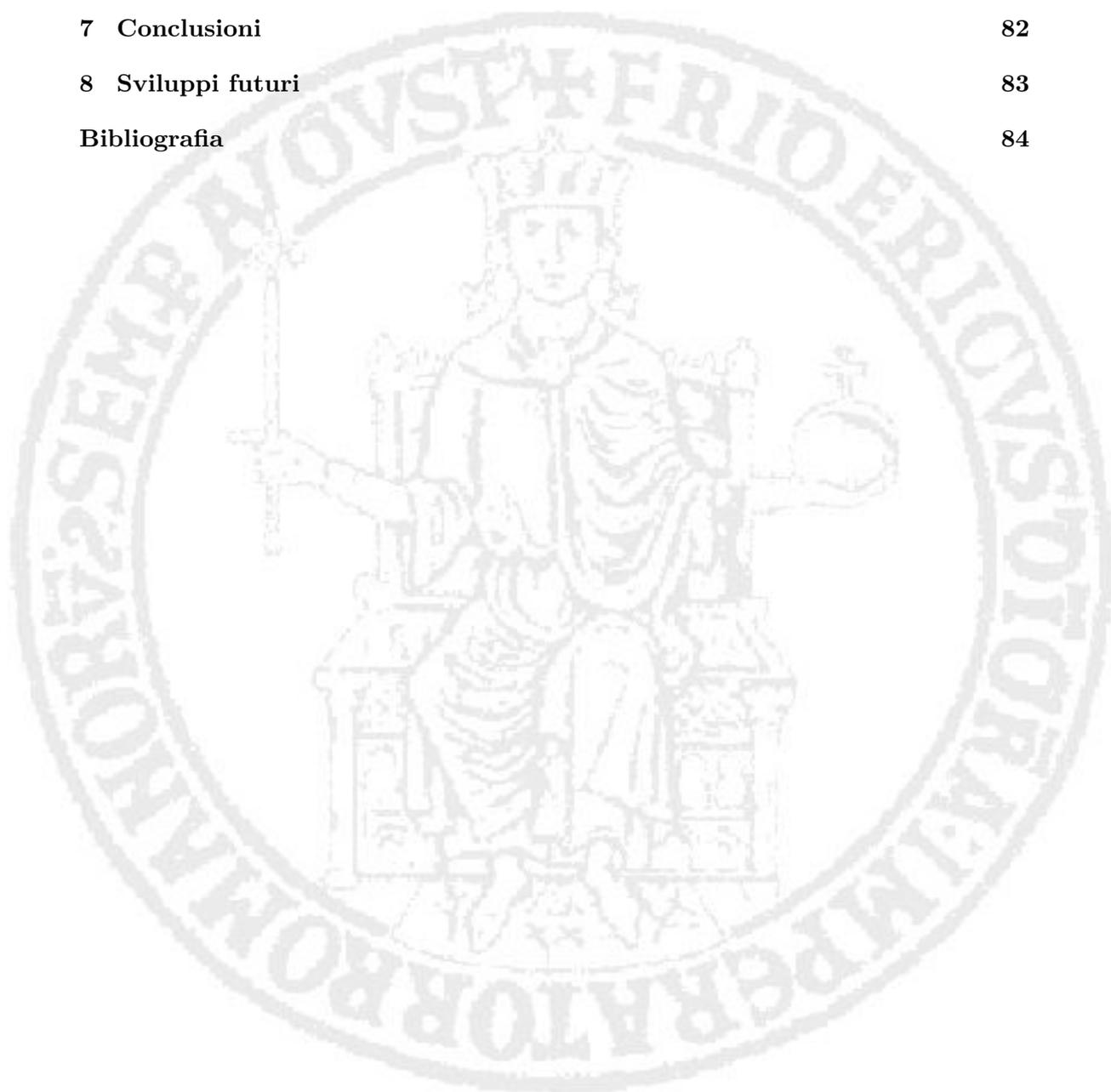
Indice

1	Introduzione	5
2	Problema e stato dell'arte	8
2.1	Caso di studio: foto ed immagini del passato	10
3	Analisi	12
3.1	Requisiti del sistema software	12
3.1.1	Requisiti funzionali utente	12
3.1.2	Requisiti funzionali di sistema	13
3.1.3	Requisiti non funzionali	13
3.2	Diagramma dei casi d'uso	14
3.2.1	Descrizione del caso d'uso UC1 [Visualizza mappa]	17
3.2.2	Descrizione del caso d'uso UC2 [Ricerca per posizione]	18
3.2.3	Descrizione del caso d'uso UC3 [Ricerca per periodo]	18
3.2.4	Descrizione del caso d'uso UC4 [Visualizza foto]	19
3.2.5	Descrizione del caso d'uso UC5 [Login]	20
3.2.6	Descrizione del caso d'uso UC6 [Visualizza galleria]	21
3.2.7	Descrizione del caso d'uso UC7 [Logout]	21
3.2.8	Descrizione del caso d'uso UC8 [Carica foto]	22
3.2.9	Descrizione del caso d'uso UC9 [Convalida foto]	23
3.2.10	Descrizione del caso d'uso UC10 [Modifica foto]	24
3.2.11	Tabella di copertura requisiti/casi d'uso	25
3.3	Diagrammi di contesto	25
3.3.1	Diagramma di contesto	25
3.3.2	Diagrammi di contesto con boundary classes	26
3.4	Modello di dominio	27
3.4.1	Razionale	29
3.5	Diagrammi dinamici	30
3.5.1	Communication diagram - Ricerca per posizione	30
3.5.2	Sequence diagram - Carica foto	30

4	Progettazione	32
4.1	Overview del Sistema	33
4.1.1	Descrizione	33
4.2	Vista architettrale strutturale	34
4.2.1	Descrizione	34
4.2.2	Razionale	34
4.3	Vista architettrale dinamica	35
4.3.1	Descrizione	35
4.3.2	Razionale	35
4.4	Module view	36
4.4.1	Descrizione	36
4.5	Decomposition style	37
4.5.1	Decomposition style Location Server	37
4.5.2	Decomposition style City Server	39
4.5.3	Decomposition style Resource Server	39
4.5.4	Decomposition style Data Manager	42
4.5.5	Decomposition style Data Model	43
4.6	Usage style	44
4.6.1	Elementi non standard	44
4.7	Vista architettrale - Componenti e connettori	46
4.7.1	Descrizione	46
4.8	Tecnologie	47
4.8.1	GIS	47
4.8.2	Base dati relazionale per i dati utente	47
4.8.3	Base dati per la configurazione della topologia della rete	48
4.8.4	SOAP vs REST over HTTP	48
4.9	Deployment diagram	49
4.9.1	Descrizione	49
5	Filtraggio e ranking	50
5.1	Filtraggio delle immagini candidate	51
5.1.1	Variazione massima di angolo a distanza nulla	52
5.1.2	Distanza massima senza rotazione	53
5.1.3	Vincolo nel caso generale	57
5.2	Ranking	61
5.2.1	Scelta della funzione di ranking	61
6	Capacity test	67
6.1	Metodologia	68
6.2	Dimensioni dei numeri di punti	69
6.3	Generazione dei punti	70

Introduzione

6.4	Misurazioni	72
6.5	Risultati	79
7	Conclusioni	82
8	Sviluppi futuri	83
	Bibliografia	84



Capitolo 1

Introduzione

Negli ultimi anni ha riscosso un sentito interesse nel campo della ricerca e nel mondo del mercato la nascita di applicazioni che sfruttino informazioni sulla posizione fisica dell'utente al fine di poter fornire servizi e contenuti dipendenti da questa.

Le informazioni legate alla posizione geografica di un dispositivo risultano importanti in differenti applicazioni, come ad esempio i **Location Based Services** (LBSs).

I LBSs sono utilizzati per fornire accesso all'utente (o in generale al dispositivo) ad uno specifico insieme di servizi sulla base della sua posizione.

Tale tipo di applicazione richiede un utilizzo delle informazioni spaziali che risultano di primo piano all'interno dell'esperienza di utilizzo, in quanto, anche qualora la posizione non sia non direttamente visibile, produce effetti che la rendono centrale e che si manifestano nei servizi accessibili.

In letteratura sono state avanzate differenti proposte per la realizzazione di tali applicazioni sia tramite estensione di architetture basate su servizi per supportare funzionalità dipendenti dalla posizione, sia tramite soluzioni progettate specificamente allo scopo.

Un secondo tipo di applicazione è l'arricchimento di foto e immagini con informazioni di natura spaziale, anche detto **geotagging**.

L'interesse per tali applicazioni è cresciuto rapidamente anche nel campo della ricerca grazie alla disponibilità su larga scala di contenuti generati dall'utente, in quanto ciò apre la via a nuove opportunità di studio e comprensione di eventi e fenomeni reali legati ad un contesto geografico.

Il processo di geotagging può avvenire sia in maniera automatica che manuale. Nel primo caso, ciò è reso possibile dalla presenza di ricevitori GPS in smartphones, fotocamere digitali ed altri dispositivi che permette l'estrazione di latitudine e longitudine associandoli alla foto tramite formati come l'Exchangeable Image File (EXIF). Nel caso di geotagging manuale, invece, è richiesto l'inserimento esplicito delle coordinate da considerare, che può avvenire o nel momento di acquisizione della risorsa (salvando poi le coordinate in un formato come ad esempio EXIF), oppure al momento dell'effettivo trasferimento ad un sistema che può, in generale, implementare un meccanismo arbitrario per associare la foto alla posizione.

Il lavoro di tesi si colloca in questo ultimo scenario ed affronta lo sviluppo di un sistema di scoperta e fruizione basato su georeferenziazione applicato al caso di foto ed immagini del passato.

Il primo capitolo descrive lo stato dell'arte nei sistemi di geotagging. E' presentato il contesto in cui si pone il geotagging, i componenti principali coinvolti nel processo, gli approcci proposti in letteratura ed i loro limiti quando il target della georeferenziazione è una foto, un'immagine o, in generale, un contenuto del passato. E' quindi illustrata una proposta di soluzione per il problema in esame, e dei meccanismi adottati per permettere la scoperta di contenuti da parte di un utente utilizzando la sua posizione geografica in maniera scalabile sia rispetto al numero degli utenti che al numero dei punti presenti nel sistema.

Il secondo capitolo è dedicato alla fase di analisi in cui è stato formalizzato il problema, definendo requisiti funzionali e non funzionali, tramite metodologie proprie dell'ingegneria del software e l'utilizzo di diagrammi UML. In particolare è stato seguito l'approccio proposto dal metodo COMET¹, che prevede la realizzazione di diagrammi di casi d'uso per la definizione dei requisiti funzionali, di modelli statici per esprimere le relazioni strutturali tra le classi del dominio del problema ed infine modelli dinamici per descrivere, per ciascun caso d'uso, gli oggetti che a questo prendono parte e le loro interazioni.

Il terzo capitolo descrive la progettazione dell'architettura software del sistema. La documentazione è stata realizzata tramite l'approccio *Views and Beyond* proposto dal SEI². Tale approccio si basa sul concetto di *View* definita come "un insieme di elementi del sistema e delle relazioni a questi associati"³, la quale rappresenta una inquadratura

¹Concurrent Object Modeling and Architectural Design Method

²Software Engineering Institute

³Views and Beyond: The SEI Approach to Architecture Documentation: "A view represents a set of system elements and the relations associated with them. Views represent the many system structures

del sistema software rispetto ad uno specifico aspetto di progetto.

Tramite l'utilizzo di viste e stili architettonici sono stati dunque mostrati gli aspetti relativi alla modularizzazione del sistema, ai componenti e connettori, alle tecnologie scelte e al deployment.

Il quarto capitolo introduce gli algoritmi proposti per realizzare un filtraggio basato su azimut e per calcolare un ranking delle immagini filtrate.

Il primo problema affrontato è la definizione di una condizione analitica affinché, noti posizione ed azimut dell'utente e di un punto di interesse, sia possibile stabilire se questi è visibile all'utente.

Il secondo problema affrontato è la costruzione di una funzione di ranking che permetta di definire un criterio per l'ordinamento tra i punti filtrati che rifletta una loro prioritizzazione rispetto al punto di vista dell'utente.

Il quinto ed ultimo capitolo illustra una procedura operativa per la valutazione delle scelte legate al dimensionamento dei nodi affinché i vincoli imposti dai requisiti non funzionali risultino rispettati considerando differenti scenari applicativi.

La descrizione della procedura prevede la presentazione degli obiettivi dell'analisi, delle assunzioni fatte, della metodologia adottata, dei tool utilizzati e dei risultati ottenuti.

Il lavoro di tesi ha dunque prodotto sia il sistema di backend che gli applicativi client per le due principali piattaforme per dispositivi mobili: Android ed iOS.

E' infine doveroso citare l'esperienza maturata alla Apple Developer Academy che ha permesso, tramite i mezzi e le conoscenze fornite, non soltanto lo sviluppo materiale dell'applicativo per iOS, proposto come challenge conclusiva del percorso di formazione, ma anche la progettazione di una user interface ed una user experience che rispetti i moderni criteri di ergonomia e design.

Capitolo 2

Problema e stato dell'arte

Il geotagging è il processo di arricchimento di un contenuto multimediale con informazioni di natura spaziale che ne identifichino una posizione geografica.

In letteratura il geotagging, quando il contenuto da georeferenziare è corredato da un riferimento testuale ad un luogo geografico, prevede due fasi:

1. **Toponym recognition:** fase di ricerca all'interno di un testo dei riferimenti testuali a locazioni geografiche (toponimi)
2. **Toponym resolution:** fase di associazione di ciascun toponimo ad una posizione geografica utilizzando un *gazetteer* (database di locazioni)

La toponym recognition è un problema di natural language processing e sono stati proposti in letteratura differenti approcci basati su tecniche di apprendimento supervisionato, come ad esempio il sistema *Web-a-Where* per l'identificazione di tutti i riferimenti geografici all'interno di una pagina Web. Anche la toponym resolution è un problema di classificazione in cui ciascuna delle possibili interpretazioni per ciascun toponimo è classificata come corretta o scorretta.

Tali fasi sono anche note con i nomi rispettivamente di geoparsing e georesolution (o geocoding).

Quando invece il processo di geotagging è applicato a contenuti che non è garantito presentino indicazioni testuali sulla locazione geografica, si possono prospettare due scenari, a seconda che i contenuti siano acquisiti tramite dispositivi dotati di ricevitore GPS o meno.

Nel primo caso, grazie alla presenza di ricevitori GPS integrati in una vasta gamma di dispositivi, il geotagging è semplificato e reso automatico durante la fase di acquisizione dei media (registrazione di un video, scatto di una foto), per cui è possibile ottenere in maniera diretta le coordinate geografiche da associare.

Nel caso in cui, invece, non sia possibile fare affidamento sulla presenza di un ricevitore GPS oppure quando l'associazione effettiva delle coordinate avviene in un secondo momento rispetto all'acquisizione del contenuto, non si dispone delle effettive coordinate ma risulta comunque possibile georeferenziare il contenuto nei seguenti scenari:

- quando si conosce un riferimento testuale alla locazione geografica (toponimo): il problema diventa di risoluzione del testo
- quando si possono fare assunzioni su un'area più o meno ampia a cui il contenuto si suppone afferisca e si ha a disposizione un dataset di contenuti registrati nella stessa area: il problema diventa di matching
- quando si può disporre dell'intervento umano: il problema diventa la conoscenza del luogo

Il primo caso è il già citato problema di toponym resolution.

Il secondo caso è trattato e affrontato in letteratura attraverso l'utilizzo di tecniche di matching nel campo della computer vision, in particolare tramite pattern recognition. Si riporta come esempio un sistema di localizzazione per immagini basato sul servizio Google Maps Street View sviluppato dalla University of Central Florida, Orlando, USA. Benché tali approcci prevedano la possibilità di automatizzare il procedimento di georeferenziazione, le limitazioni risultano tuttavia quelle di presupporre la presenza di altri media nell'area e soprattutto che tali media rispecchino uno stato del luogo quanto più vicino possibile al contenuto da localizzare.

Il terzo caso prevede un approccio basato sull'intervento umano di personale che conosca l'area geografica di interesse e che manualmente fornisca tag geografici (riportando il problema al primo caso) o fornendo, laddove possibile, direttamente le coordinate geografiche.

Il principale svantaggio risulta legato al fatto che l'approccio non risulta in generale (senza la conoscenza di ulteriori vincoli sulla natura dei contenuti da considerare) scalabile rispetto al numero di contenuti che un'applicazione avrebbe interesse a georeferenziare. I vantaggi di tale approccio, tuttavia, supponendo di fare affidamento su personale specializzato, si manifestano nella precisione dei risultati di georeferenziazione e nella possibilità di risolvere il problema del riconoscimento del luogo anche qualora questo abbia subito trasformazioni nel corso del tempo.

2.1 Caso di studio: foto ed immagini del passato

Il lavoro di tesi si basa un caso di studio che si colloca nel contesto descritto e prende in considerazione foto ed immagini del passato.

E' affrontata di seguito la specifica delle caratteristiche di tale particolarizzazione e le implicazioni rispetto alle metodologie appena illustrate.

Nel caso specifico di foto ed immagini del passato il geotagging è inquadrato come un problema dove alcune delle proposte appena illustrate non risultano applicabili.

In particolare, il caso considerato presenta le seguenti caratteristiche:

- la foto o l'immagine non ha alcun riferimento testuale alla posizione geografica: ad esempio quando la foto è digitalizzata ma non catalogata (caso di interesse pratico per gli archivi storici e fotografici). In questo caso non è possibile applicare una procedura di geocoding
- la foto o l'immagine presenta un riferimento testuale alla posizione ma il geocoding non fornisce alcun risultato, o fornisce un risultato inesatto (luogo diverso): questo è un potenziale scenario dovuto al fatto che il nome della località può aver subito cambiamenti nel tempo e gli attuali gazetteers sono realizzati ed aggiornati per mantenere soltanto i toponimi in vigore
- supponendo di poter in qualche modo conoscere a priori l'area a cui la foto afferisce, può risultare comunque difficile se non impossibile evincere la locazione tramite matching e pattern recognition, sia a causa di una quantità insufficiente di punti pre-registrati nell'area utilizzati come base per il confronto, sia perché tali punti potrebbero non riflettere lo stesso stato del luogo della foto da georeferenziare

Dovendo progettare una soluzione specifica per il caso in esame, è necessario tenere in conto le considerazioni mostrate.

La georeferenziazione effettuata tramite selezione manuale delle coordinate geografiche, ossia tramite l'attività di utenti collaboratori, risulta sempre applicabile, ma presenta i seguenti svantaggi:

- conoscenza del luogo
- scalabilità (frequenza di georeferenziazioni vs frequenza di inserimento di nuovi contenuti)

Per quanto concerne il problema della scalabilità, è da notare che lo specifico caso in esame presenta un'ulteriore caratteristica legata al fatto che foto o immagini del passato, per quanto in numero arbitrariamente grande, non possono aumentare in maniera

indefinita come avverrebbe invece per sistemi basati su contenuti nuovi generati in larga scala da utenti quotidianamente.

Tale osservazione trasforma il problema della scalabilità delle georeferenziazioni manuali rispetto alla velocità inserimento di nuovi contenuti in un problema di dimensionamento, in quanto è presente almeno teoricamente un punto di saturazione dei contenuti.

Per mitigare il problema è stato dunque introdotto un meccanismo di partecipazione aperta per l'inserimento dei punti nel sistema, per il quale è richiesta un'indicazione esplicita della posizione.

Per il problema della conoscenza del luogo nel processo di localizzazione, invece, l'idea principale è di introdurre in primis un concetto di raggruppamento di un sottoinsieme di punti in base all'*area*, per poter circoscrivere il campo di azione di un collaboratore all'area in cui questi si trova (una persona presente a Napoli avrà, verosimilmente, maggiori probabilità di conoscere una strada di Napoli che non di Roma).

Questa circoscrizione, comunque, non è progettata in modo da risultare vincolante (in quanto una persona presente a Napoli potrebbe conoscere anche Roma), ma getta comunque le basi per un'architettura costruita su di una suddivisione geografica.

Tale considerazione fa dunque emergere un criterio di base per la suddivisione delle foto, ma per definire un'"area" in maniera più dettagliata sono necessarie ulteriori considerazioni riguardanti il dimensionamento del sistema che saranno affrontate successivamente. Nel seguito le aree saranno assimilate a città per un mera questione nominale (si parlerà di City server).

Il lavoro che si andrà a presentare nasce sia dalle considerazioni illustrate che da aspetti di gamification, i quali non saranno esplicitamente trattati in questa descrizione, ma che implicheranno l'aggiunta nel sistema di requisiti funzionali legati al profilo utente ed al concetto di "scoperta" di una foto.

Capitolo 3

Analisi

Il seguente capitolo descrive la fase di analisi condotta sul sistema. Saranno pertanto definiti ed affrontati in dettaglio gli aspetti relativi all'analisi statica, quali i requisiti del sistema, i casi di utilizzo previsti ed il modello di dominio, e quelli relativi all'analisi dinamica, tramite communication e sequence diagrams.

3.1 Requisiti del sistema software

3.1.1 Requisiti funzionali utente

- RF1.** Il sistema deve permettere ad un utente di ricercare i punti di interesse entro un pre-determinato raggio dalla sua posizione corrente, in termini di coordinate GPS
- RF2.** Il sistema deve permettere ad un utente di registrarsi, accedere ed uscire dal sistema
- RF3.** Il sistema deve permettere ad un utente di visualizzare su di una mappa (widget grafico) tutti i punti di interesse *nei dintorni* (vedere *RF1*)
- RF4.** Il sistema deve permettere ad un utente di accedere ai dettagli di un punto di interesse selezionato, comprensivo di immagine, informazioni testuali e riferimenti spaziale e temporale
- RF5.** Il sistema deve permettere ad un utente che sia entrato nel sistema di visualizzare l'elenco contenente ciascuno dei contenuti scoperti, indipendentemente dalla sua posizione corrente
- RF6.** Il sistema deve permettere ad un utente di ricercare, tra i soli contenuti scoperti, i punti di interesse il cui riferimento temporale sia contenuto o parzialmente sovrapposto ad un periodo da questi indicato
- RF7.** Il sistema deve permettere ad un utente collaboratore di caricare nuovi contenuti comprensivi di:

- immagine, nella forma di file da caricare o tramite URL
- informazioni testuali, quali descrizione, autore (se noto) e titolo indicativo
- riferimento temporale, nella forma di periodo o data specifica
- riferimento spaziale, nella forma di area, indirizzo o coordinate GPS
- azimut dell'immagine qualora noto

Tali contenuti saranno in attesa di approvazione da parte di un operatore

RF8. Il sistema deve permettere ad un utente operatore di gestire una richiesta di nuovo contenuto, consentendone la modifica totale o parziale delle informazioni, e successivamente di convalida o di rimozione della richiesta

RF9. Il sistema deve permettere ad un utente operatore di modificare in maniera totale o parziale le informazioni di un punto di interesse selezionato e/o di rimuoverlo

3.1.2 Requisiti funzionali di sistema

RF10. Il sistema deve tenere traccia dei punti di interesse scoperti da ciascun utente registrato

3.1.3 Requisiti non funzionali

RNF1. Il sistema deve risultare scalabile sia rispetto al numero di utenti che al numero di contenuti presenti

RNF2. Il sistema deve essere compatibile con le principali piattaforme per dispositivi mobili al tempo di stesura della presente

RNF3. Il sistema deve essere compatibile con i principali web browser al tempo di stesura della presente

RNF4. Il sistema deve essere usabile¹

¹*"In software engineering, usability is the degree to which a software can be used by specified consumers to achieve quantified objectives with effectiveness, efficiency, and satisfaction in a quantified context of use."* [Ergonomic Requirements for Office Work with Visual Display Terminals, ISO 9241-11, ISO, Geneva, 1998]

3.2 Diagramma dei casi d'uso

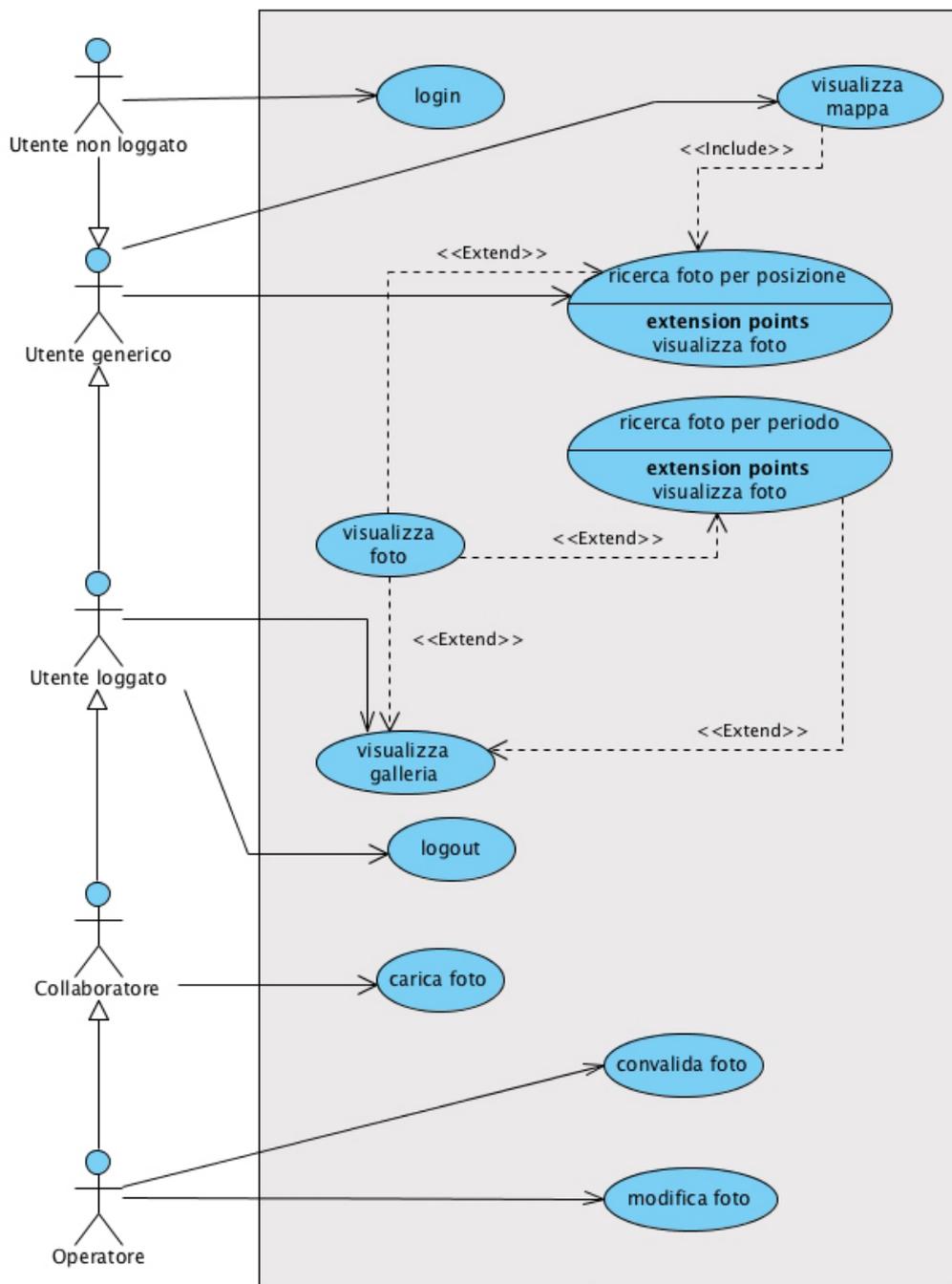


Figura 1: Diagramma dei casi d'uso

Nome elemento	Tipo	Descrizione
Utente generico	Attore primario	Attore umano, accede alle funzionalità di ricerca di una foto in base alla posizione o ad un periodo temporale e alla sua visualizzazione con relative informazioni ad essa associate
Utente non loggato	Attore primario	Attore umano, eredita le funzionalità di un utente generico ed accede alla funzionalità di login all'interno del sistema
Utente loggato	Attore primario	Attore umano, eredita le funzionalità di un utente generico ed accede alla funzionalità di visualizzazione della galleria delle foto trovate e di logout dal sistema
Collaboratore	Attore primario	Attore umano, eredita le funzionalità di un utente loggato ed accede alla funzionalità di upload di un nuovo contenuto (in attesa di convalida)
Operatore	Attore primario	Attore umano, eredita le funzionalità di un collaboratore ed accede alla funzionalità di modifica e convalida dei contenuti (immagini e informazioni descrittive)
Visualizza mappa	Caso d'uso	Descrive le operazioni da compiere per accedere alla mappa contenente i contenuti presenti ricercati su base spaziale dall'utente
Ricerca foto per posizione	Caso d'uso	Descrive le operazioni da compiere per ricercare una foto sulla base di coordinate spaziali
Ricerca foto per periodo	Caso d'uso	Descrive le operazioni da compiere per ricercare una foto sulla base di un riferimento temporale

Visualizza foto	Caso d'uso	Descrive le operazioni da compiere per visualizzare un contenuto comprensivo di immagine, informazioni testuali e riferimenti spaziali e temporali
Login	Caso d'uso	Descrive le operazioni da compiere per effettuare l'accesso all'interno del sistema
Visualizza galleria	Caso d'uso	Descrive le operazioni da compiere per accedere alla galleria delle foto ricercate
Logout	Caso d'uso	Descrive le operazioni da compiere per effettuare l'uscita dal sistema
Carica foto	Caso d'uso	Descrive le operazioni da compiere per l'upload di un nuovo contenuto (in attesa di convalida), comprensivo di immagine, informazioni testuali e riferimenti spaziali e temporali
Convalida foto	Caso d'uso	Descrive le operazioni da compiere per la convalida di un contenuto in attesa di approvazione
Modifica foto	Caso d'uso	Descrive le operazioni da compiere per effettuare la modifica delle informazioni associate ad un contenuto

Tabella 3.1: Elementi del diagramma dei casi d'uso

Segue la descrizione dettagliata di ciascun caso d'uso.

3.2.1 Descrizione del caso d'uso UC1 [Visualizza mappa]

Breve descrizione	L'utente accede ad una mappa centrata nella sua corrente posizione GPS, nella quale saranno mostrati tutti i punti di interesse posti ad una distanza inferiore ad una soglia pre-determinata. L'utente potrà successivamente accedere alle informazioni di ciascuno dei punti presenti
Attori coinvolti	Utente generico
Casi d'uso correlati	Ricerca punti di interesse per posizione
Pre-condizioni	La posizione GPS dell'utente deve essere disponibile
Scenario principale	<ol style="list-style-type: none">1. L'utente accede ad una view contenente la mappa2. In base al servizio di localizzazione, il sistema estrae la posizione GPS dell'utente3. La mappa viene centrata nella posizione estratta4. Caso d'uso: Ricerca punti di interesse per posizione5. Il sistema posiziona ciascuno dei punti trovati sulla mappa in base alle proprie coordinate GPS
Post-condizioni	La mappa contiene tutti i punti di interesse entro un raggio pre-determinato

3.2.2 Descrizione del caso d'uso UC2 [Ricerca per posizione]

Breve descrizione	In base ad una posizione P ed un raggio R, il sistema ricerca l'insieme dei punti di interesse distanti da P al più R
Attori coinvolti	Utente generico
Casi d'uso correlati	Visualizza foto
Pre-condizioni	Nessuna
Scenario principale	<ol style="list-style-type: none"> 1. L'utente effettua una ricerca dei punti specificando come raggio di input un valore eventualmente pre-inserito e come posizione di input quella associata alle proprie coordinate GPS 2. Il sistema ricerca tutti i punti di interesse la cui distanza dalla posizione di partenza sia inferiore del raggio indicato in input 3. Il sistema restituisce la lista dei risultati all'utente

3.2.3 Descrizione del caso d'uso UC3 [Ricerca per periodo]

Breve descrizione	In base ad un periodo temporale T, il sistema ricerca l'insieme dei punti di interesse il cui riferimento temporale (data o periodo) sia contenuto o parzialmente sovrapposto a T
Attori coinvolti	Utente generico
Casi d'uso correlati	Visualizza foto
Pre-condizioni	Nessuna
Scenario principale	<ol style="list-style-type: none"> 1. L'utente effettua una ricerca dei punti specificando un periodo di interesse tramite l'inserimento di almeno un parametro tra: data iniziale; data finale 2. Il sistema ricerca tutti i punti di interesse il cui riferimento temporale (data o periodo) sia contenuto o parzialmente sovrapposto al periodo indicato in input 3. Il sistema restituisce la lista dei risultati all'utente

3.2.4 Descrizione del caso d'uso UC4 [Visualizza foto]

Breve descrizione	Il sistema mostra l'immagine, le informazioni testuali, i riferimenti temporali e le coordinate spaziali associate ad una foto
Attori coinvolti	Utente generico
Casi d'uso correlati	<ul style="list-style-type: none">• Ricerca per posizione• Ricerca per periodo• Visualizza galleria
Pre-condizioni	L'utente deve aver già ricercato il contenuto richiesto
Scenario principale	<ol style="list-style-type: none">1. L'utente seleziona il contenuto di interesse2. Il sistema preleva le informazioni associate al contenuto richiesto3. Il sistema tramite una nuova view mostra all'utente l'immagine, le informazioni testuali, i riferimenti temporali e le coordinate spaziali associate al contenuto

3.2.5 Descrizione del caso d'uso UC5 [Login]

Breve descrizione	L'utente fornisce le proprie credenziali per effettuare l'accesso all'interno del sistema
Attori coinvolti	Utente non loggato
Pre-condizioni	L'utente non è loggato
Scenario principale	<ol style="list-style-type: none">1. L'utente inserisce le proprie credenziali per accedere al sistema2. Il sistema verifica la correttezza delle credenziali inserite3. In caso di credenziali corrette, l'utente risulterà loggato ed avrà accesso al proprio profilo e ai contenuti scoperti
Post-condizioni	L'utente è loggato
Scenario alternativo	<ol style="list-style-type: none">3. In caso di credenziali incorrette, il sistema mostra un messaggio di errore all'utente
Post-condizioni	Nessuna

3.2.6 Descrizione del caso d'uso UC6 [Visualizza galleria]

Breve descrizione	Il sistema mostra all'utente i contenuti scoperti senza esporne i dettagli
Attori coinvolti	Utente loggato
Casi d'uso correlati	<ul style="list-style-type: none"> • Visualizza foto • Ricerca foto per periodo
Pre-condizioni	L'utente deve essere loggato
Scenario principale	<ol style="list-style-type: none"> 1. L'utente accede alla galleria 2. Il sistema recupera tutti i punti di interesse scoperti dall'utente 3. Il sistema restituisce e mostra all'utente i risultati della ricerca in ordine cronologico dei riferimenti temporali associati a ciascun punto

3.2.7 Descrizione del caso d'uso UC7 [Logout]

Breve descrizione	L'utente esce dal sistema
Attori coinvolti	Utente loggato
Pre-condizioni	L'utente è loggato
Scenario principale	<ol style="list-style-type: none"> 1. Il sistema registra l'uscita dal sistema 2. Il sistema aggiorna la view dell'utente riportandolo alla view principale
Post-condizioni	L'utente non è loggato

3.2.8 Descrizione del caso d'uso UC8 [Carica foto]

Breve descrizione	L'utente carica un nuovo contenuto fornendo immagine, descrizione testuale, riferimento temporale e riferimento spaziale della foto (coordinate GPS, area, indirizzo)
Attori coinvolti	Utente collaboratore
Pre-condizioni	Nessuna
Scenario principale	<ol style="list-style-type: none">1. L'utente fornisce la nuova immagine tramite upload o URL2. L'utente inserisce le descrizioni testuali del nuovo punto di interesse3. L'utente indica un riferimento temporale associato alla foto (data/periodo)4. L'utente indica un riferimento spaziale della foto (coordinate GPS, area, indirizzo)5. L'utente conferma i dati inseriti6. Il sistema crea una nuova richiesta in sospeso salvando i dati ad essa associata
Post-condizioni	Creazione di una nuova richiesta in sospeso

3.2.9 Descrizione del caso d'uso UC9 [Convalida foto]

Breve descrizione	L'operatore prende in carico una richiesta in sospeso decidendo se approvarla o rimuoverla
Attori coinvolti	Utente operatore
Pre-condizioni	Nessuna
Scenario principale	<ol style="list-style-type: none"> 1. L'operatore accede alla lista delle richieste in sospeso 2. Il sistema ricerca e restituisce tutte le richieste in sospeso 3. L'operatore prende in carico una richiesta 4. L'operatore modifica, se necessario, i campi inseriti dall'utente 5. L'operatore accerta, ed eventualmente corregge, la posizione GPS del nuovo contenuto indicandone anche l'azimut 6. L'operatore conferma le informazioni sul nuovo punto di interesse 7. Il sistema marca la richiesta come risolta/approvata e aggiunge un nuovo punto di interesse
Post-condizioni	Creazione di un nuovo punto di interesse e rimozione della richiesta in sospeso dalla lista
Scenario alternativo	<ol style="list-style-type: none"> 4. L'operatore rimuove la richiesta 5. Il sistema marca la richiesta come risolta/non approvata
Post-condizioni	Rimozione della richiesta in sospeso dalla lista

3.2.10 Descrizione del caso d'uso UC10 [Modifica foto]

Breve descrizione	L'operatore modifica le informazioni, il riferimento temporale e/o le coordinate spaziali di un punto di interesse
Attori coinvolti	Utente operatore
Pre-condizioni	Nessuna
Scenario principale	<ol style="list-style-type: none"> 1. L'operatore ricerca o seleziona il punto di interesse 2. L'operatore modifica, dove necessario, i dati del contenuto 3. L'operatore accerta, ed eventualmente corregge, la posizione GPS del nuovo contenuto indicandone anche l'azimut 4. L'operatore conferma le informazioni sul nuovo punto di interesse 5. Il sistema aggiorna le informazioni sul punto di interesse
Post-condizioni	Il punto di interesse selezionato risulta modificato
Scenario alternativo	<ol style="list-style-type: none"> 2. L'operatore rimuove il punto di interesse 3. Il sistema rimuove il punto di interesse
Post-condizioni	Cancellazione del punto di interesse dal sistema

3.2.11 Tabella di copertura requisiti/casi d'uso

	UC1	UC2	UC3	UC4	UC5	UC6	UC7	UC8	UC9	UC10
RF1		V								
RF2					V		V			
RF3	V									
RF4				V						
RF5						V				
RF6			V							
RF7								V		
RF8									V	
RF9										V

3.3 Diagrammi di contesto

3.3.1 Diagramma di contesto

Il seguente diagramma di contesto descrive il sistema come una classe aggregata con lo stereotipo «system», e modella l'ambiente esterno sotto forma di classi esterne. Vi è inoltre una descrizione delle associazioni e delle interazioni tra gli attori ed il sistema stesso.

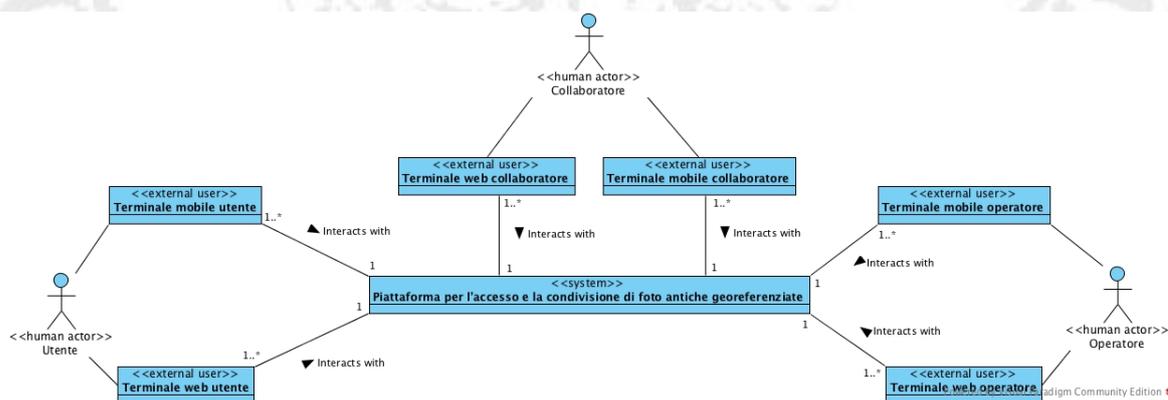


Figura 2: Diagramma di Contesto

3.3.2 Diagrammi di contesto con boundary classes

Ad un livello più basso del diagramma precedente, sono mostrati i moduli del sistema con i quali gli attori esterni interagiscono. Tali moduli sono denominati *boundary classes*.

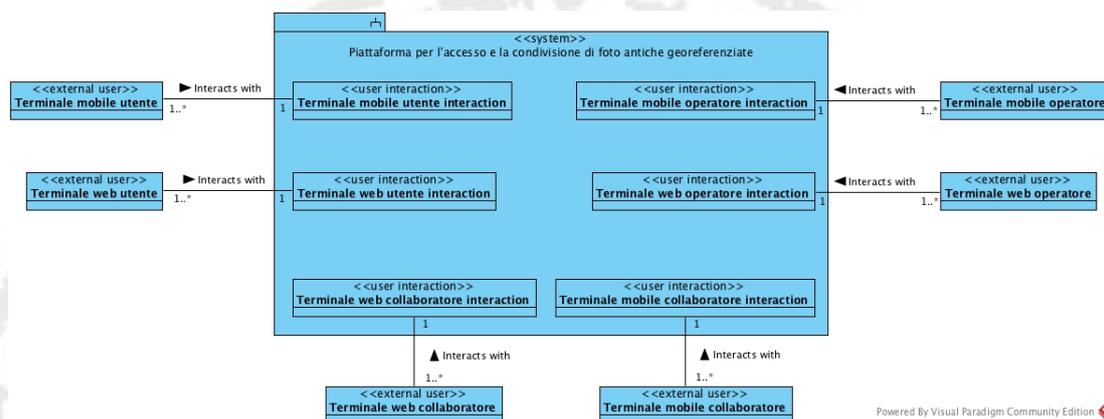


Figura 3: Diagramma di contesto con classi boundary

Nome elemento	Tipo	Descrizione
Utente mobile	External user	Colui il quale usufruisce delle funzionalità di ricerca spaziale e temporale, visualizzazione dei punti scoperti e visualizzazione nel dettaglio di un punto di interesse. Sono per esso disponibili posizione GPS e azimuth
Utente web	External user	Colui il quale usufruisce delle funzionalità di ricerca temporale, visualizzazione dei punti scoperti e visualizzazione nel dettaglio di un punto di interesse
Collaboratore mobile	External user	Colui il quale si occupa dell'upload di nuovi punti di interesse comprensivi di immagine, informazioni testuali e riferimenti spaziale e temporale. Sono per esso disponibili posizione GPS e azimuth

Collaboratore web	External user	Colui il quale si occupa dell'upload di nuovi punti di interesse comprensivi di immagine, informazioni testuali e riferimenti spaziale e temporale
Operatore mobile	External user	Colui il quale si occupa della convalida, modifica e rimozione dei punti di interesse. Sono per esso disponibili posizione GPS e azimut
Operatore web	External user	Colui il quale si occupa della convalida, modifica e rimozione dei punti di interesse

Tabella 3.3: Elementi del diagramma di contesto

3.4 Modello di dominio

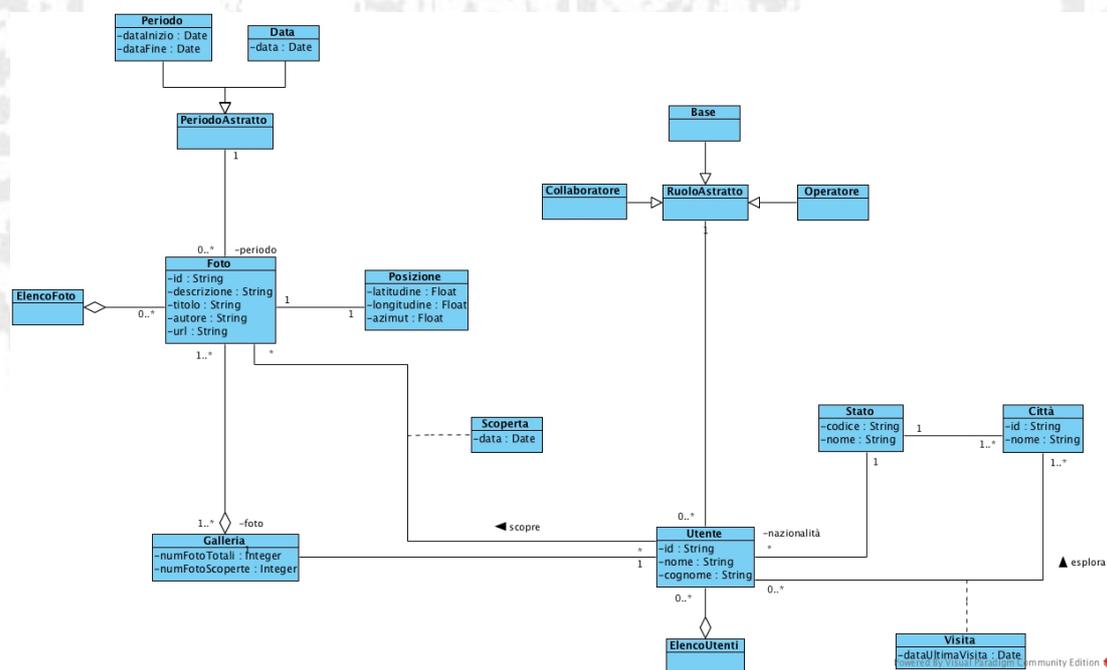


Figura 3: Modello di dominio

Nome elemento	Descrizione
Elenco Foto	Entità ottenuta come aggregazione di Foto, contiene la lista delle foto del sistema
Foto	Entità contenente le informazioni associate alle foto: id, descrizione, titolo, autore e URL
Periodo Astratto	Classe astratta utilizzata per generalizzare il riferimento temporale associato ad una foto
Data	Specializzazione concreta di un periodo astratto. E' il riferimento temporale associato ad una foto sotto forma di specifica data
Periodo	Specializzazione concreta di un periodo astratto. E' il riferimento temporale associato ad una foto sotto forma di data iniziale e data finale
Galleria	Entità contenente l'aggregazione delle foto scoperte da un utente
Utente	Entità contenente le informazioni associate ad un utente: id, nome e cognome
Elenco Utenti	Entità ottenuta come aggregazione di Utente, contiene la lista degli utenti del sistema
Posizione	Contiene le informazioni relative al riferimento spaziale di una foto: latitudine, longitudine e azimut
Scoperta	Classe associativa tra Foto e Utente. Contiene la data di scoperta della foto da parte dell'utente cui fa riferimento
Ruolo astratto	Classe astratta ottenuta per generalizzare il ruolo di un utente del sistema
Collaboratore	Specializzazione concreta di un ruolo astratto. E' il ruolo associato ad un utente collaboratore
Operatore	Specializzazione concreta di un ruolo astratto. E' il ruolo associato ad un utente operatore
Base	Specializzazione concreta di un ruolo astratto. E' il ruolo associato ad un utente normale
Stato	Entità che contiene le informazioni relative al codice ed al nome di uno stato
Città	Entità che contiene le informazioni relative all'id e al nome della città

Visita	Classe associativa tra Utente e Città. Contiene la data di ultima visita di un utente presso la città cui fa riferimento
--------	--

Tabella 3.4: Elementi del modello di dominio

3.4.1 Rationale

Si è deciso di introdurre l'entità Elenco Foto al fine di poter accedere alle foto presenti nel sistema durante l'esecuzione del caso d'uso UC2 (Ricerca per posizione).

Elenco Utenti è un'entità necessaria per la selezione dell'utente loggato che interagisce col sistema durante l'esecuzione dei caso d'uso UC3 (Ricerca per periodo), UC6 (Visualizza galleria), UC7 (Logout), UC8 (Carica foto) e UC9 (Convalida foto).

La presenza di più ruoli per gli utenti del sistema è stata modellata applicando il pattern Player-Role, il quale ha portato alla creazione delle classi Ruolo Astratto, Collaboratore, Operatore ed Utente base.

E' stata inoltre introdotta la classe Periodo Astratto per generalizzare il tipo di riferimento temporale associato ad una foto, il quale può essere una specifica data (classe Data) oppure un periodo (classe Periodo).

3.5 Diagrammi dinamici

Sono mostrati di seguito i diagrammi dinamici per due casi d'uso di particolare interesse: Ricerca per posizione [UC2] e Carica foto [UC9].

3.5.1 Communication diagram - Ricerca per posizione

Il seguente modello descrive l'interazione tra l'utente ed il sistema durante il caso d'uso di ricerca in base alla posizione.

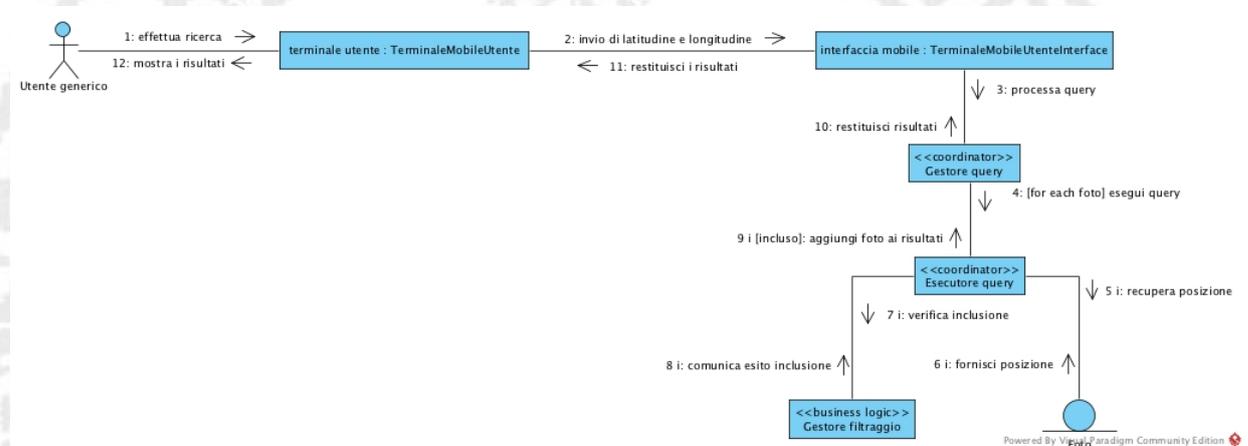


Figura 4: Communication diagram - Ricerca per posizione

3.5.2 Sequence diagram - Carica foto

Il seguente modello descrive l'interazione tra l'utente ed il sistema durante il caso d'uso di upload di una nuova foto, fornendo oltre alla stessa, le informazioni sulla posizioni ed altre informazioni descrittive del contenuto.

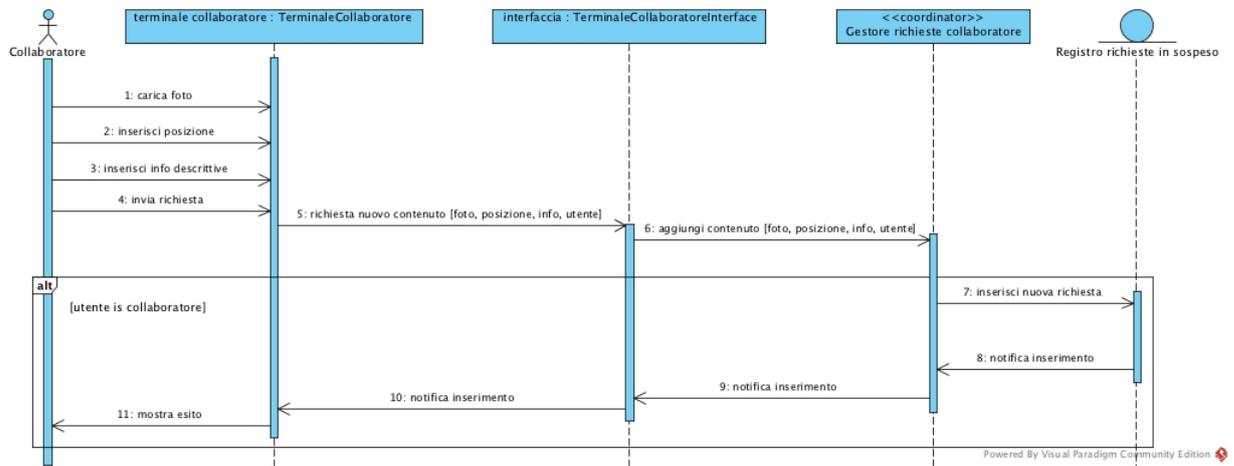
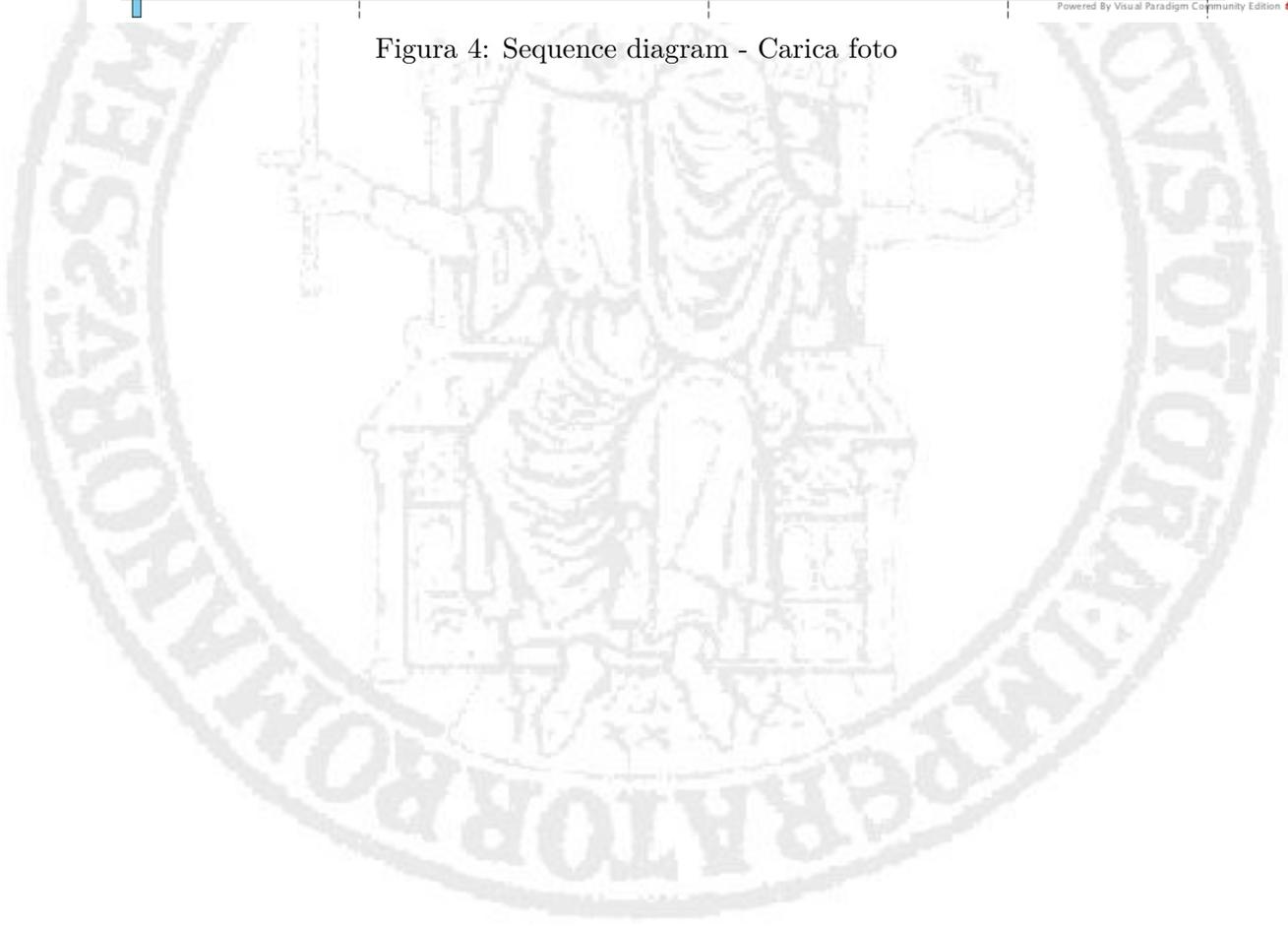


Figura 4: Sequence diagram - Carica foto



Capitolo 4

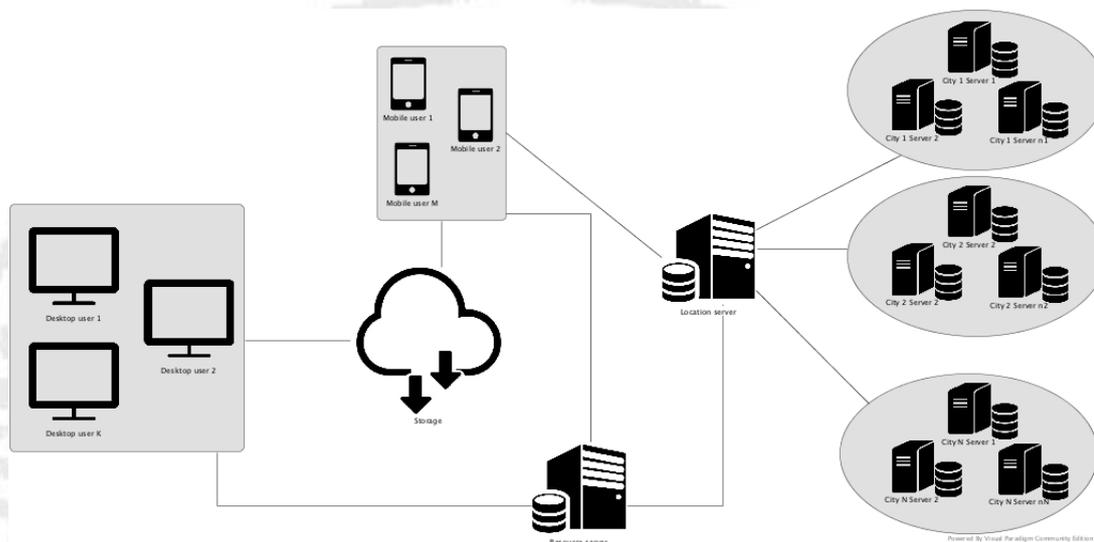
Progettazione

In questo capitolo sono presentate le scelte progettuali, l'architettura del sistema e le viste architettoniche scelte per la descrizione.



4.1 Overview del Sistema

In questa sezione è rappresentata in maniera informale la struttura del Sistema.



4.1.1 Descrizione

Il sistema presenta due differenti tipologie di client - mobile e desktop - che è necessario diversificare per la funzionalità di retrieving dei contenuti georeferenziati accessibile esclusivamente dai dispositivi mobili.

Quest'ultima funzionalità è demandata ad un Location Server, il quale conosce i server afferenti a ciascuna delle città inserite nel sistema - che possono essere, in generale, molteplici - e si occupa del dispatching delle richieste utente su uno di tali server, secondo le scelte progettuali descritte.

Le immagini legate ai contenuti georeferenziati sono fisicamente presenti su un sistema di storage esterno ed acceduto da entrambe le tipologie di utente.

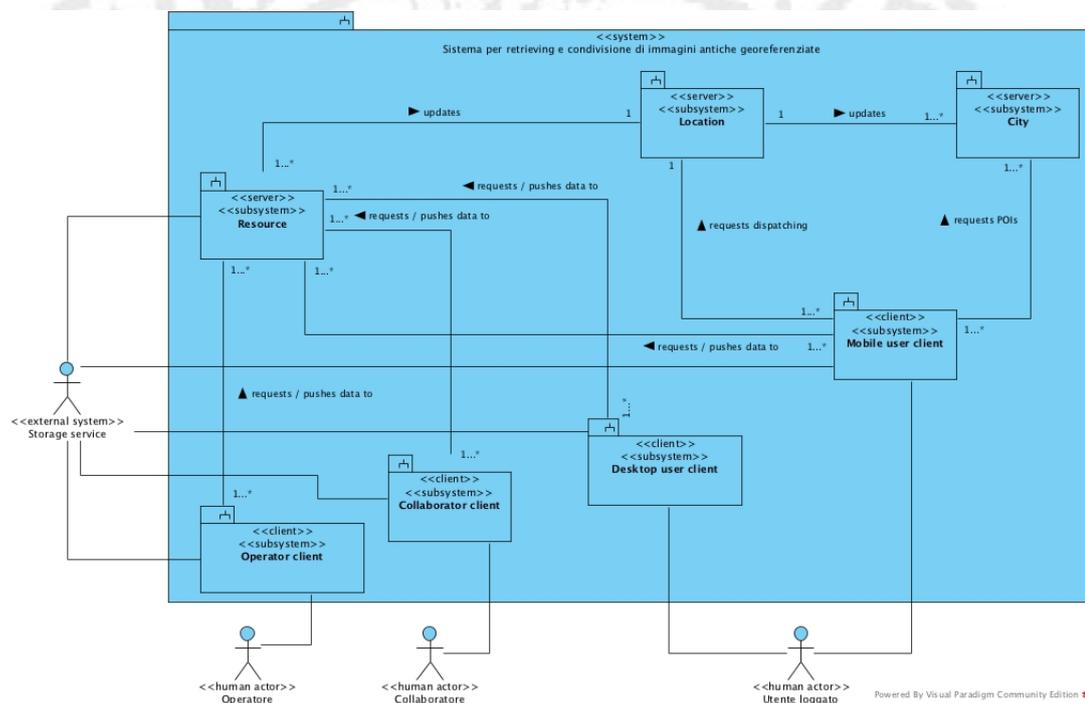
In ultima istanza, è presente un Resource Server responsabile delle procedure di accesso ed uscita dal sistema - login, logout e sessioni utente - e della gestione, in termini di inserimento, aggiornamento, cancellazione e selezione dei dati del profilo utente e relativi dati satellite.

4.2 Vista architetturale strutturale

4.2.1 Descrizione

Si mostrano i componenti fondamentali dell'architettura del sistema, ottenuti andando a decomporle in sottosistemi.

Ciascuno di essi è responsabile della realizzazione di una specifica funzionalità e tale da mostrare uno scarso accoppiamento tra tali sottosistemi.



4.2.2 Rationale

Per la scelta architetturale sono stati tenuti in considerazione il dominio del problema, i requisiti funzionali e non funzionali ed è emersa una natura del sistema che è stata strutturata come segue:

- Una architettura di tipo client-server per il mapping di un client su di un City server e per il retrieving dei POIs presso quest'ultimo
- Una architettura di tipo client-server, in particolare RESTful, per le funzionalità di accesso al sistema e per le operazioni di selezione, creazione, modifica e cancellazione dei dati legati al profilo utente e dati satellite

La differenza per la architettura a microservizi risiede nella scelta di utilizzare un unico set di interfacce per offrire servizi a client eterogenei.

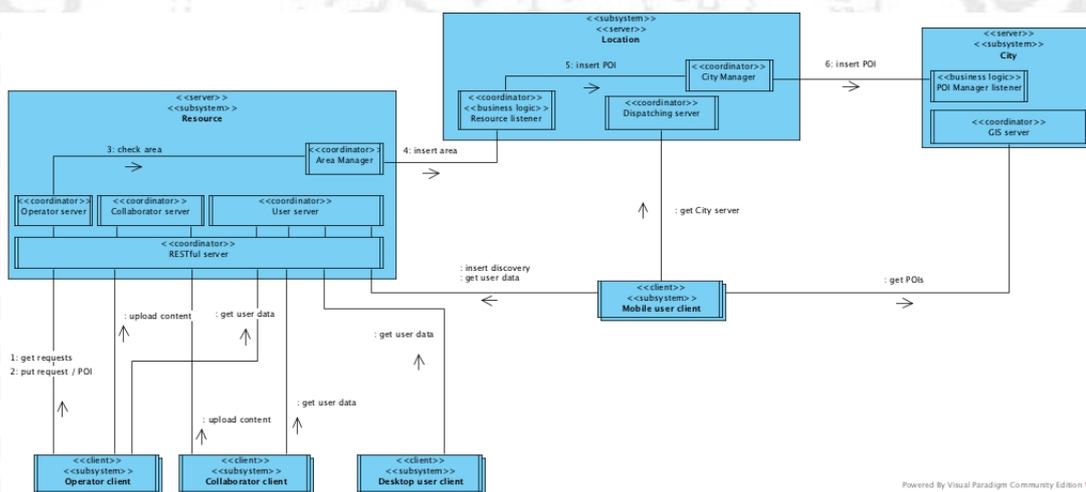
4.3 Vista architetturale dinamica

4.3.1 Descrizione

Questa vista viene utilizzata per mostrare le comunicazioni tra gli elementi architetturali, andando a definire la tipologia di messaggi scambiati.

In particolare sono evidenziati i messaggi scambiati all’atto della convalida di una richiesta di inserimento di un nuovo contenuto tra il client operatore e i server.

Gli altri messaggi presenti non hanno un contatore numerico in quanto sono messaggi stand-alone che non producono ulteriori messaggi tra quelli identificati a questo livello di astrazione.



4.3.2 Rationale

Sebbene il Resource server offra delle interfacce comuni a differenti tipologie di client, esso non risulta unicamente composto dal componente che le espone (e realizza), ma prevede due ulteriori componenti che espongono le interfacce per i servizi offerti ad utenti collaboratori e ad utenti operatori.

Tuttavia questa strutturazione dei componenti del sottosistema è mascherata ai client, i quali vedranno un’unica interfaccia comune (**façade**) e la logica di accesso ai servizi è demandata ad un componente interno che realizza un meccanismo di autenticazione ed autorizzazione.

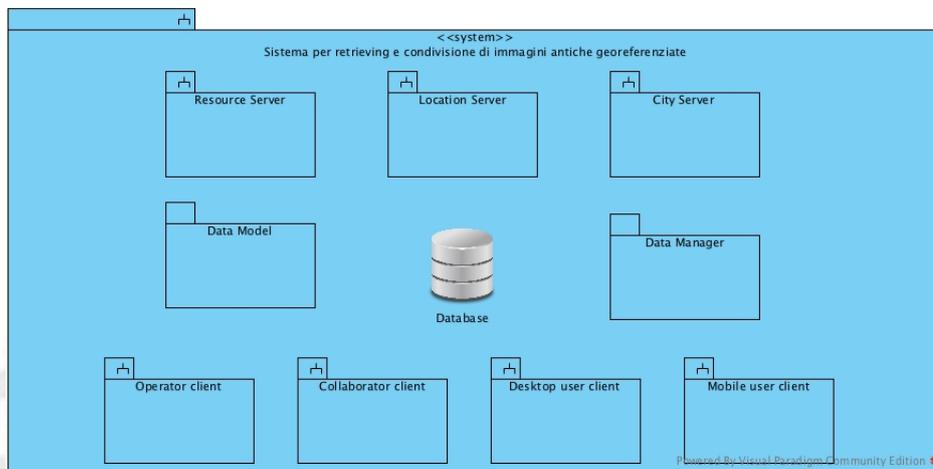
4.4 Module view

4.4.1 Descrizione

Il seguente diagramma illustra la vista a moduli, ad alto livello, del sistema. Sono stati individuati i seguenti sottosistemi:

- Resource Server: sistema per l'esposizione delle interfacce dei servizi di accesso e modifica dei dati utente e dati satellite
- Location Server: sistema per il dispatching delle richieste dei client mobile e che si occupa di conoscere e modificare l'architettura fisica dei server di città, in termini di disposizione e tipologia dei contenuti
- City Server: sistema per la ricerca e il retrieving dei punti di interesse entro un certo raggio dalla posizione richiesta
- Data Manager: responsabile della gestione delle entità e della persistenza dei dati utente e satellite
- Data Model: modello dei dati utilizzato per descrivere il dominio dei dati utente e satellite (ad un livello di astrazione maggiore delle classi Entity)
- Operatore client: sistema che permette ad un operatore di accedere alla lista delle richieste in sospeso, e successivamente di scartarle, modificarle e approvarle
- Collaboratore client: sistema che permette ad un collaboratore di caricare nuovi contenuti
- Client desktop utente: sistema che permette la visualizzazione dei dati profilo e satellite dell'utente che ha effettuato l'accesso
- Client mobile utente: sistema che permette la scoperta di punti di interesse entro un certo raggio dalla posizione attuale del dispositivo e che permette la visualizzazione dei dati profilo e satelliti dell'utente che ha effettuato l'accesso
- Database: base dati per la persistenza delle informazioni

Da notare che nel seguente diagramma la natura della base dati (SQL/NoSQL) non è esplicitata, né tantomeno il numero di istanze o la specifica tecnologia.



4.5 Decomposition style

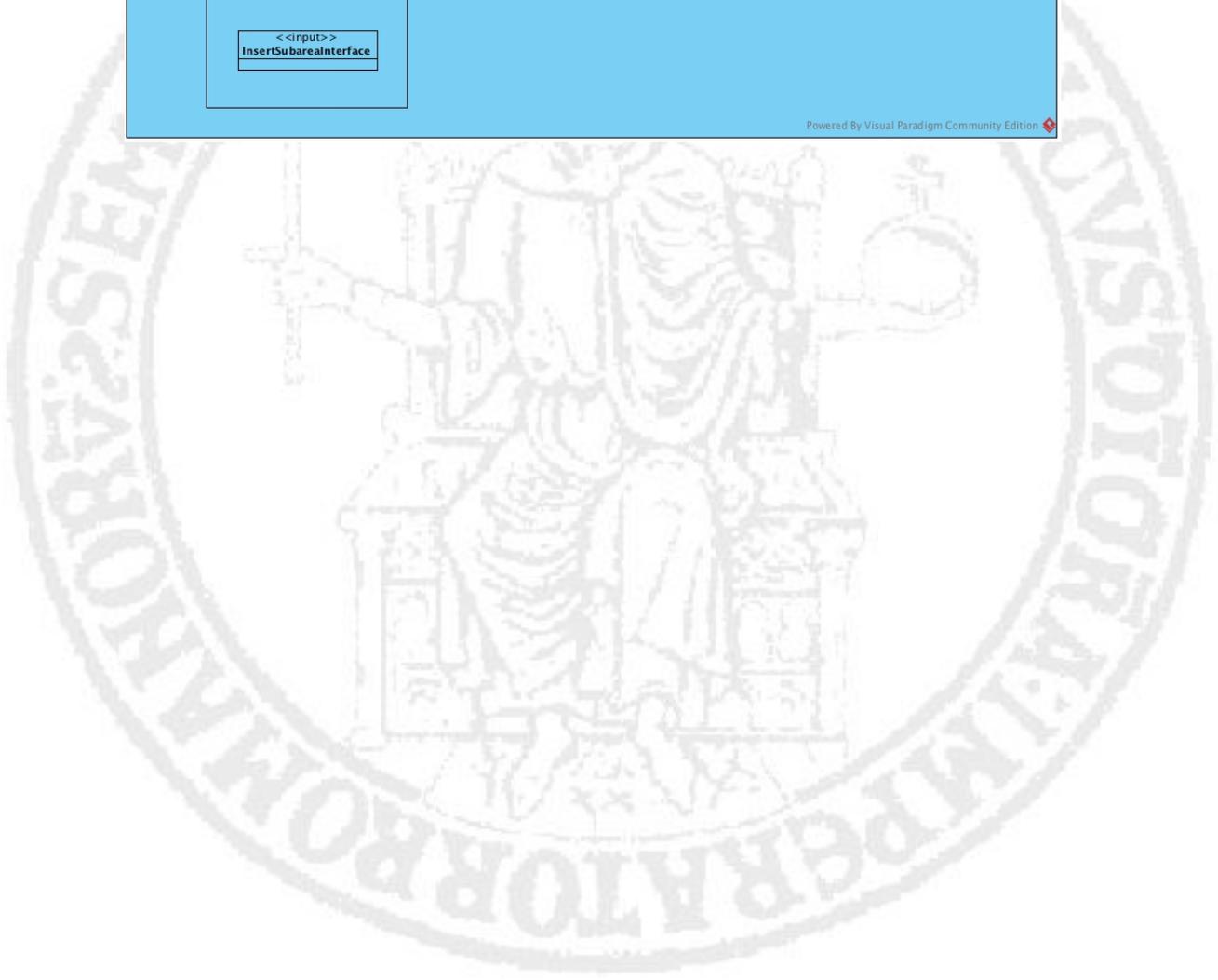
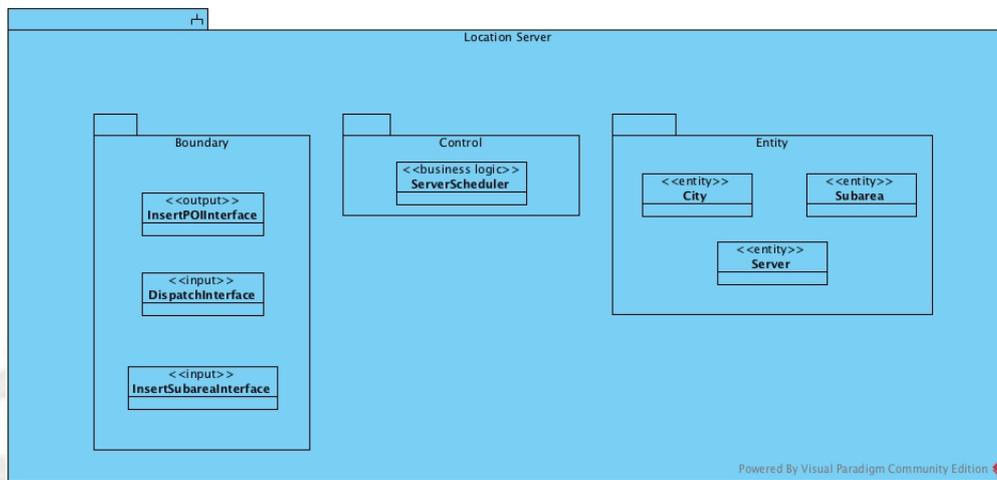
Nel seguito sono mostrati i diagrammi ottenuti applicando il decomposition style al precedente modello. I moduli ottenuti seguono in linea di principio un'architettura Boundary-Control-Entity.

4.5.1 Decomposition style Location Server

Descrizione

È stato applicato il decomposition style al sottosistema "Location Server", individuando i seguenti componenti:

- Boundary, contenente:
 - InsertPOIInterface: per l'interfacciamento con il City server
 - DispatchInterface: interfaccia per il servizio di dispatching offerto ai mobile client
 - InsertSubareaInterface: interfaccia per il servizio di inserimento di una nuova sotto-area
- Control, contenente:
 - ServerScheduler: logica di business per lo scheduling dei server per una data città
- Entity, contenente le classi entity rappresentanti città, aree e server

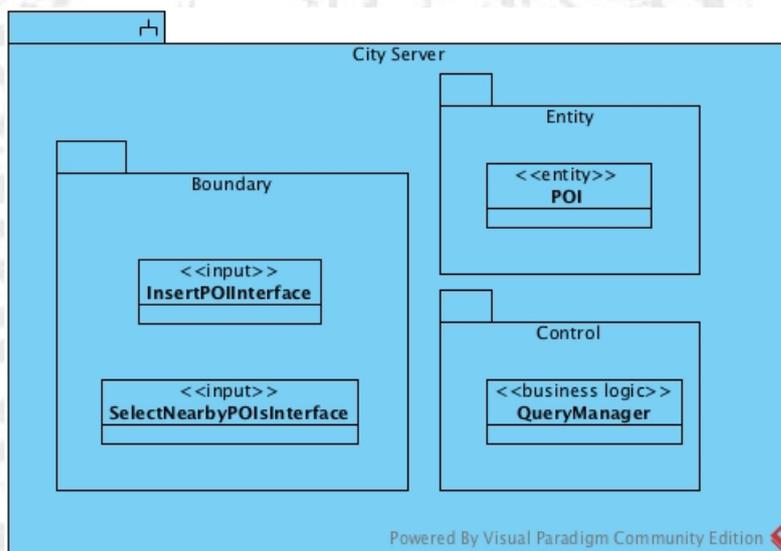


4.5.2 Decomposition style City Server

Descrizione

È stato applicato il decomposition style al sottosistema "City Server", individuando i seguenti componenti:

- Boundary, contenente:
 - InsertPOIInterface: interfaccia per il servizio di inserimento di un nuovo POI
 - SelectNearbyPOIsInterface: interfaccia per il servizio di retrieving dei POIs entro un raggio da una posizione in input
- Control contenente un Query Manager responsabile di preparare le query con i parametri di input per poi sottometerle al proprio GIS
- Entity, contenente la classe POI



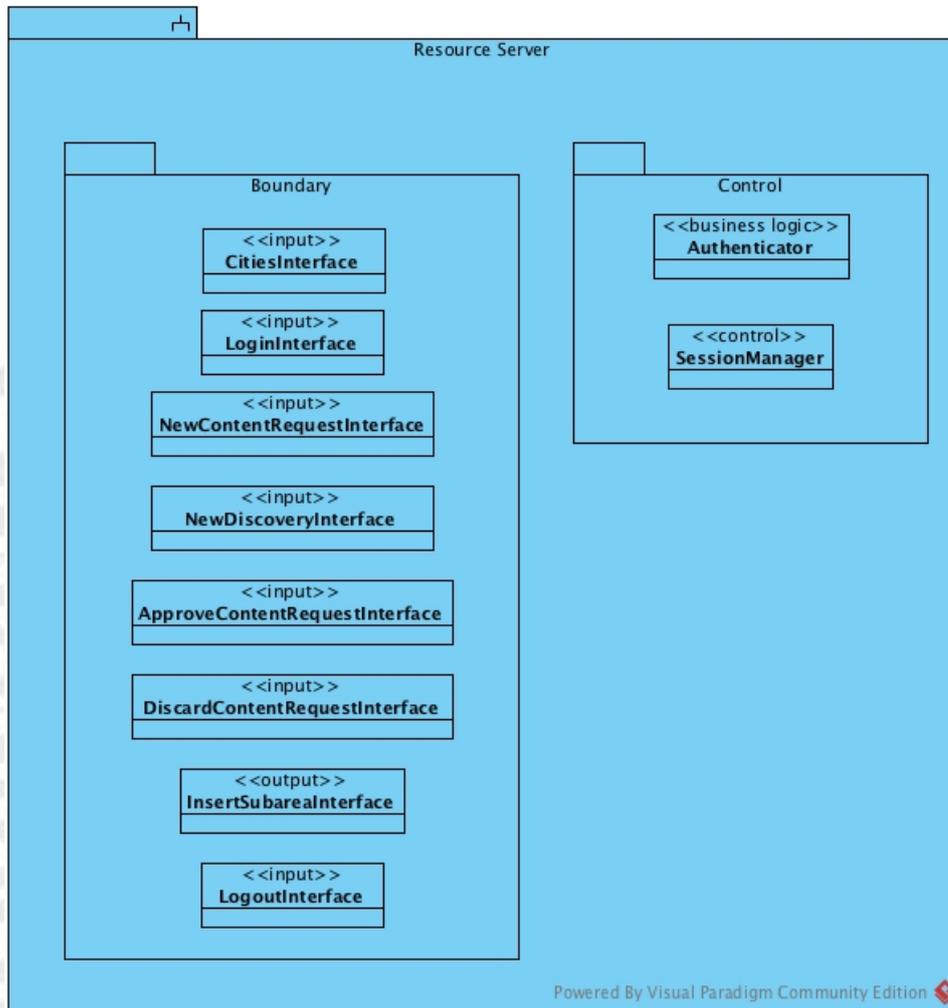
4.5.3 Decomposition style Resource Server

Descrizione

È stato applicato il decomposition style al sottosistema "Resource Server", individuando i seguenti componenti:

- Boundary, contenente:
 - CitiesInterface: interfaccia per il servizio di retrieving della lista città

- LoginInterface: interfaccia per il servizio di accesso al sistema
- LogoutInterface: interfaccia per il servizio di uscita dal sistema
- NewContentRequestInterface: interfaccia per il servizio di upload di una nuova richiesta
- NewDiscoveryInterface: interfaccia per il servizio di registrazione di una nuova scoperta di un utente ad un POI
- ApproveContentRequestInterface: interfaccia per il servizio di approvazione di una richiesta pendente e conseguente inserimento del POI proposto all'interno del sistema
- DiscardContentRequestInterface: interfaccia per il servizio di rimozione di una richiesta pendente
- InsertSubareaInterface: interfaccia per il servizio di inserimento nel sistema di una nuova area afferente ad una città in input
- Boundary, contenente:
 - Authenticator: componente responsabile di implementare un meccanismo di autenticazione
 - SessionManager: componente responsabile della gestione in termini di creazione, salvataggio e rimozione delle sessioni utente

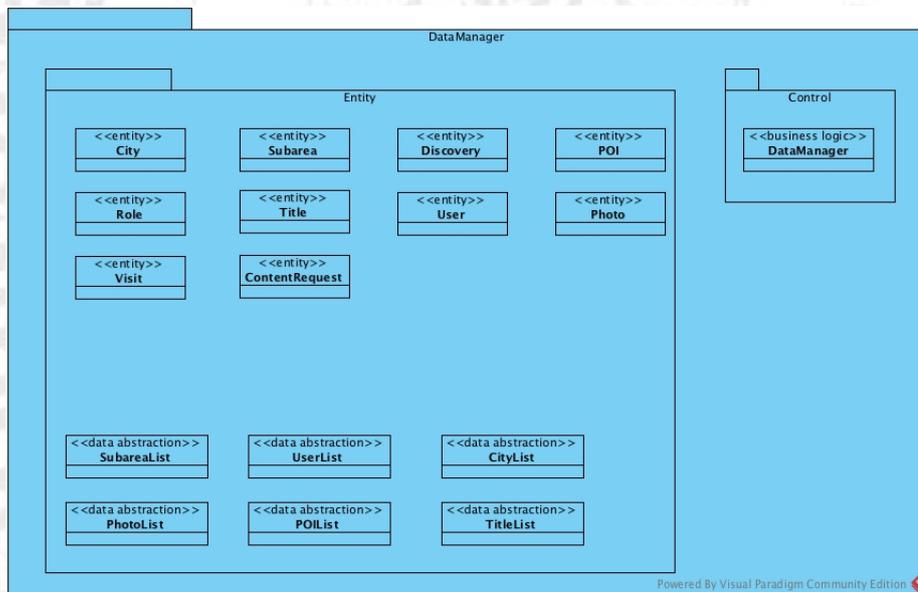


4.5.4 Decomposition style Data Manager

Descrizione

È stato applicato il decomposition style al sottosistema "DataManager", individuando i seguenti componenti:

- Control, contenente:
 - DataManager: componente responsabile della gestione in termini di selezione, inserimento, aggiornamento, cancellazione e persistenza dei dati utente e satellite
- Entity, contenente le classi descrittive del modello di dominio, e prevede inoltre strutture dati aggregate che compongono una vista in memoria sui dati

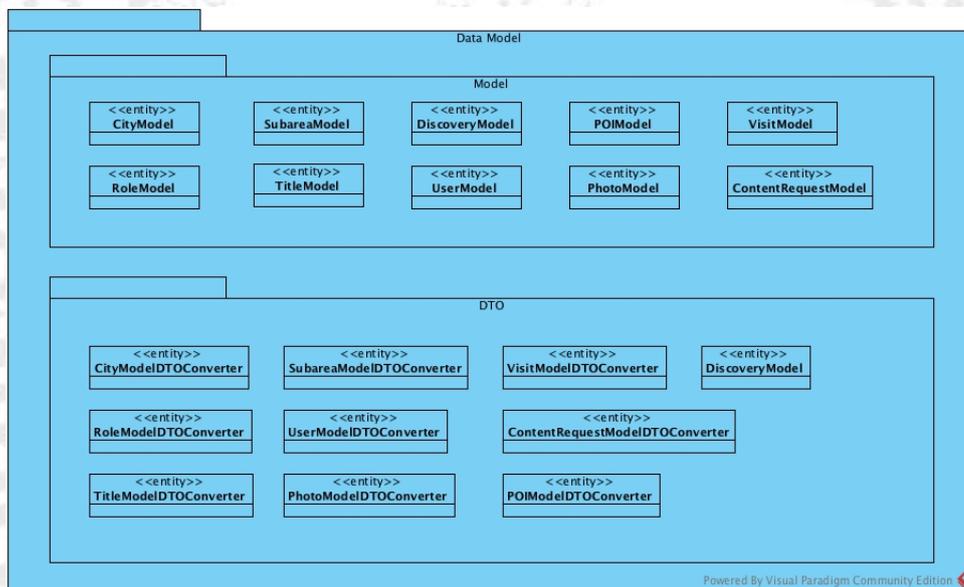


4.5.5 Decomposition style Data Model

Descrizione

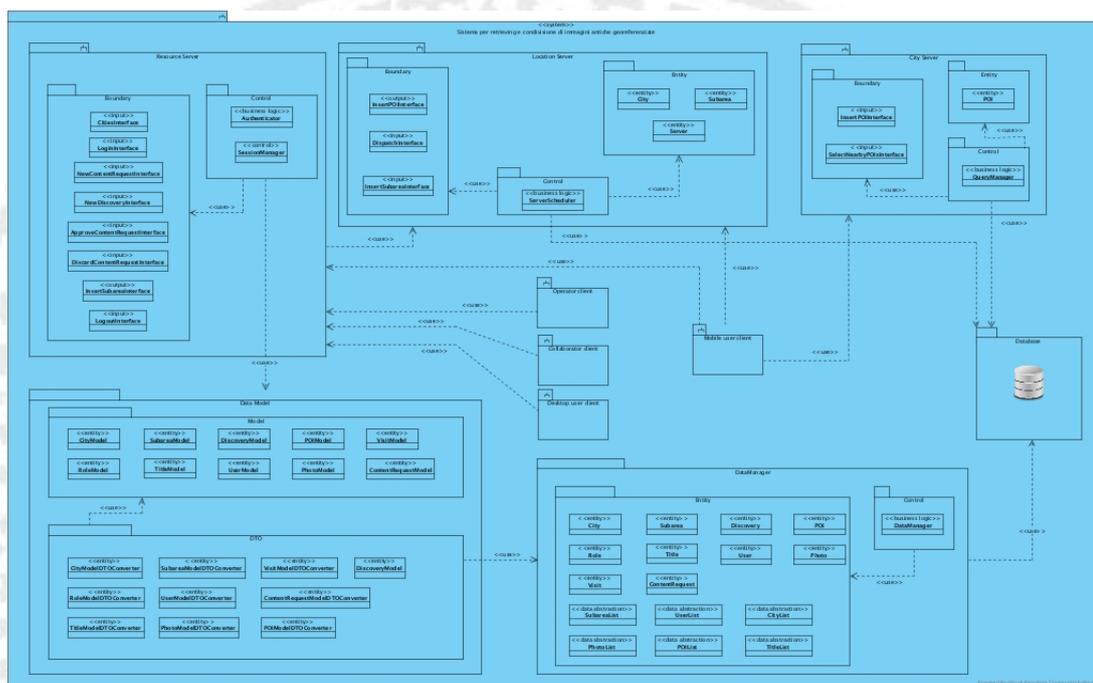
È stato applicato il decomposition style al sottosistema "DataModel", individuando i seguenti componenti:

- Model, contenente il corrispettivo delle classi entity ma ad un livello di astrazione maggiore, rappresentando pienamente il modello di dominio del problema
- DTO Converter, classi di utilità utilizzate per la trasformazione dal modello più grezzo (Entity) a quello più astratto (Model)



4.6 Usage style

Applicando al sistema lo stile "Usage & Decomposition" è stato ottenuto il seguente diagramma:



4.6.1 Elementi non standard

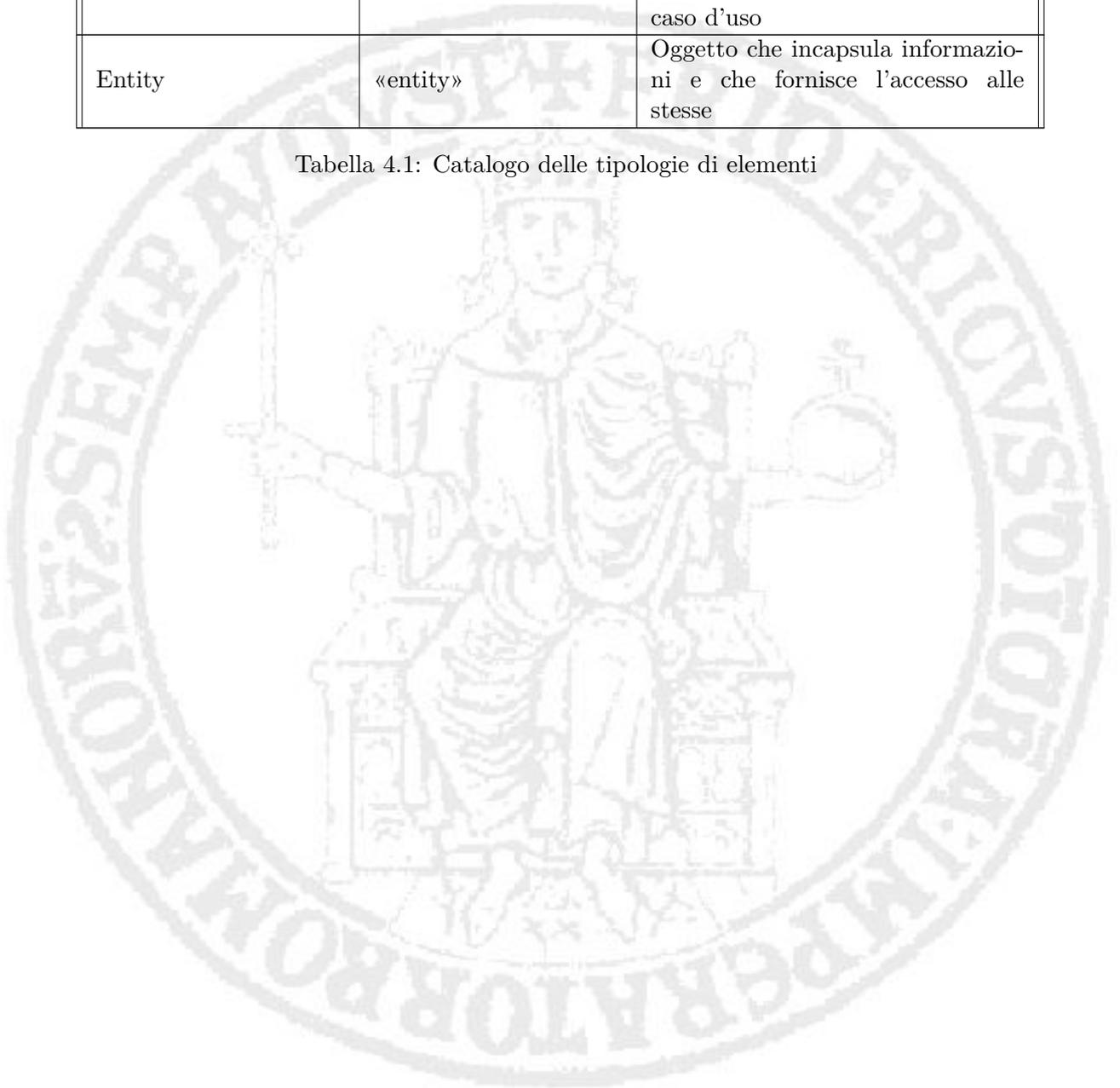
Catalogo delle tipologie di elementi

In questa sezione sono riportate le tipologie di elementi non standard utilizzati.

Nome	Stereotipo testuale	Descrizione
Database	«database»	Base dati. In questa fase la sua natura (SQL/NoSQL) non è esplicitata.
Input	«input»	Interfaccia di input attraverso cui attori, sistemi o sottosistemi esterni comunicano col sistema
Output	«output»	Interfaccia di output attraverso cui il sistema comunica con attori, sistemi o sottosistemi esterni

Control	«control»	Modulo che implementa la logica di controllo al fine di realizzare un caso d'uso
Entity	«entity»	Oggetto che incapsula informazioni e che fornisce l'accesso alle stesse

Tabella 4.1: Catalogo delle tipologie di elementi

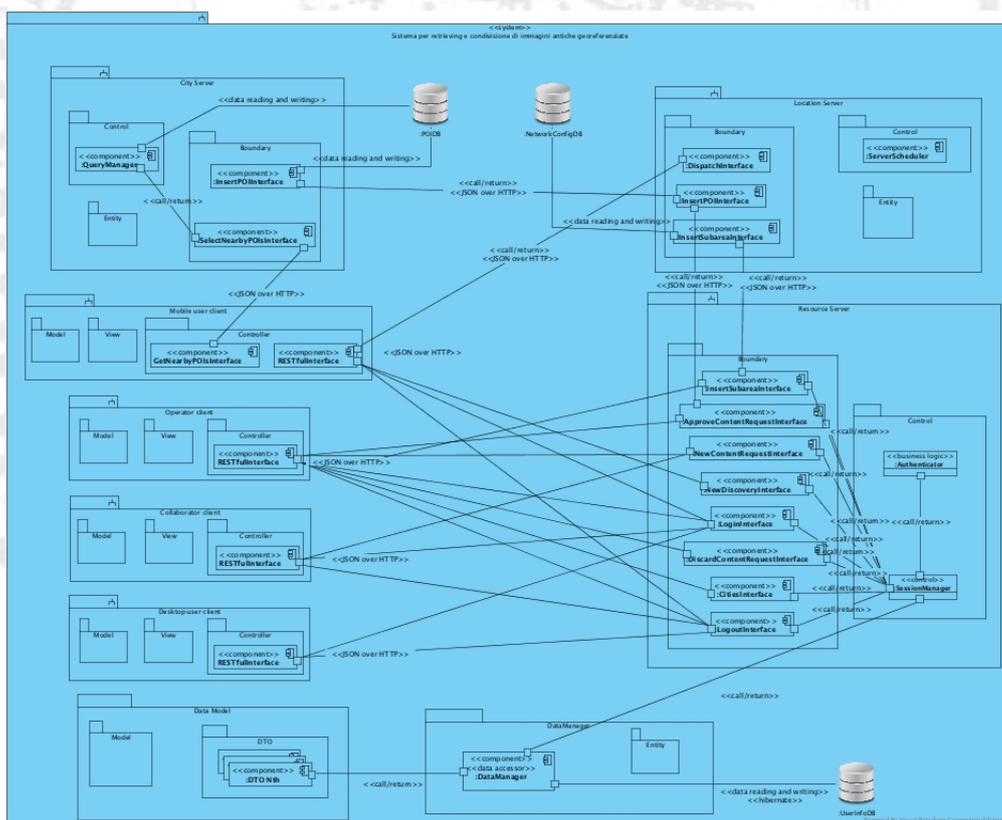


4.7 Vista architetturale - Componenti e connettori

4.7.1 Descrizione

Questa vista a componenti e connettori indica come i diversi componenti dell'architettura comunicano tra di loro e quali tipi di connettori utilizzano per tale comunicazione. In particolare sono stati individuati tre pattern di comunicazione:

- **Call/Return:** Connettore utilizzato per descrivere l'invocazione di un metodo oppure una comunicazione sincrona client/server
- **Data Reading and Writing:** Connettore utilizzato da un componente Data Accessor per richiedere i dati forniti da un componente di tipo Repository. Il nome del connettore specifica la tecnologia utilizzata. Nel caso specifico, la comunicazione col DB avviene tramite JDBC, ma si è scelto di utilizzare Hibernate per il mapping object-relational delle strutture della base dati in oggetti Java



4.8 Tecnologie

La scelta delle tecnologie ha riguardato i seguenti aspetti:

- GIS
- Base dati relazionale per i dati utente e satellite
- Base dati per la configurazione della topologia della rete
- SOAP vs REST

4.8.1 GIS

I sistemi informativi geografici rappresentano uno strumento di notevole interesse nel caso specifico.

La prima questione affrontata è relativa alla natura della base dati da considerare. Dacché punti di interesse e foto sono oggetti la cui rappresentazione - per non introdurre vincoli alla tipologia dei metadati né limitarne l'espandibilità futura (es. aggiunta di dati multimediali audio) - è schemaless, è stata valutato MongoDB con la sua estensione per dati spaziali come prima potenziale scelta tecnologica. I vantaggi di tale soluzione sono la possibilità di scaling orizzontale (o data sharding) semplificato e la facilità di configurazione. Dall'altro lato, c'è lo svantaggio di poter utilizzare un set di funzionalità inerenti le query spaziali che risulta più limitato rispetto ad un sistema puramente GIS. Nonostante la natura schemeless delle informazioni trattate, è stato pensato di denormalizzare il modello dei dati legato a foto e punti di interesse al fine di estrarre un set minimo di campi di natura relazionale (a schema fisso) contenente esclusivamente informazioni spaziali al fine di poter sfruttare una base dati SQL pensata specificamente per realizzare un sistema GIS. Gli svantaggi sono una fase di configurazione più delicata e maggiore effort nello scaling orizzontale del sistema, ma offre la possibilità di processing di query più strutturate della precedente soluzione. La scelta è ricaduta su PostGIS, estensione GIS del sistema Postgresql.

4.8.2 Base dati relazionale per i dati utente

Per quanto concerne i dati utente, è stato da subito scelto un modello di dati relazionale, per garantire il rispetto delle proprietà ACID.

La scelta tecnologica è stata Postgresql, essendo una soluzione open source e per la possibilità di poter uniformare le tecnologie per le basi relazionali.

4.8.3 Base dati per la configurazione della topologia della rete

Il componente responsabile di conoscere la struttura della rete è il Location Server. Tale configurazione è di natura schemaless in quanto rappresenta oggetti con una struttura variabile, comprendente ad esempio:

- quali e quanti server afferiscono ad un'area
- aree afferenti ad uno stesso set di server
- regole di mapping tra la descrizione testuale di un'area ottenuta dall'operazione di reverse geocoding ed il set di server che la copre

La tipologia di base dati selezionata è dunque NoSql ed in particolare di tipo document-oriented, che permettesse di gestire il versioning dei documenti di configurazione, fornendo i vantaggi di un formato per la codifica delle informazioni che fosse leggibile e facilmente manipolabile anche da un operatore umano, come JSON o XML.

La scelta è ricaduta su CouchDB che fornisce interfacce JSON over REST/HTTP e controllo e gestione concorrenza.

4.8.4 SOAP vs REST over HTTP

Nel seguito del paragrafo si parlerà del formato SOAP e dello stile architetturale REST, con la premessa che, nonostante differenti e concettualmente non comparabili in maniera diretta, nel primo caso non si farà riferimento allo formato in quanto tale, ma in quanto standard di descrizione dei metodi utilizzato in un'architettura SOA, mentre si farà invece riferimento allo stile REST nello specifico caso del protocollo HTTP.

La scelta tra SOAP e REST per la progettazione delle interfacce da esporre nel Resource Server si è basata sulle seguenti osservazioni:

- Filosofia di progettazione: mentre SOAP concepisce una visione del Web come piattaforma basata su servizi, REST adotta una visione basata sul concetto di risorsa
- Proprietà dell'approccio: SOAP utilizza HTTP come mero protocollo di trasporto, benché non sia vincolato ad esso, e definisce una propria sovrastruttura su HTTP per l'interoperabilità di servizi sul Web; REST sfrutta appieno le potenzialità di HTTP in quanto protocollo di livello applicazione e non necessita di un ulteriore livello per definire i metodi di accesso alle risorse

Benché il principale vantaggio di SOAP sia la definizione di uno standard indipendente dal Web, permettendo un maggiore disaccoppiamento tra client e server, ciò non risulta, tuttavia, un reale valore aggiunto laddove sia nota dalle specifiche la natura dei client. Inoltre considerando i requisiti funzionali, è possibile inquadrare le operazioni richieste sui dati utente nella tavola CRUD. E' stato pertanto scelto il paradigma RESTful.

4.9 Deployment diagram

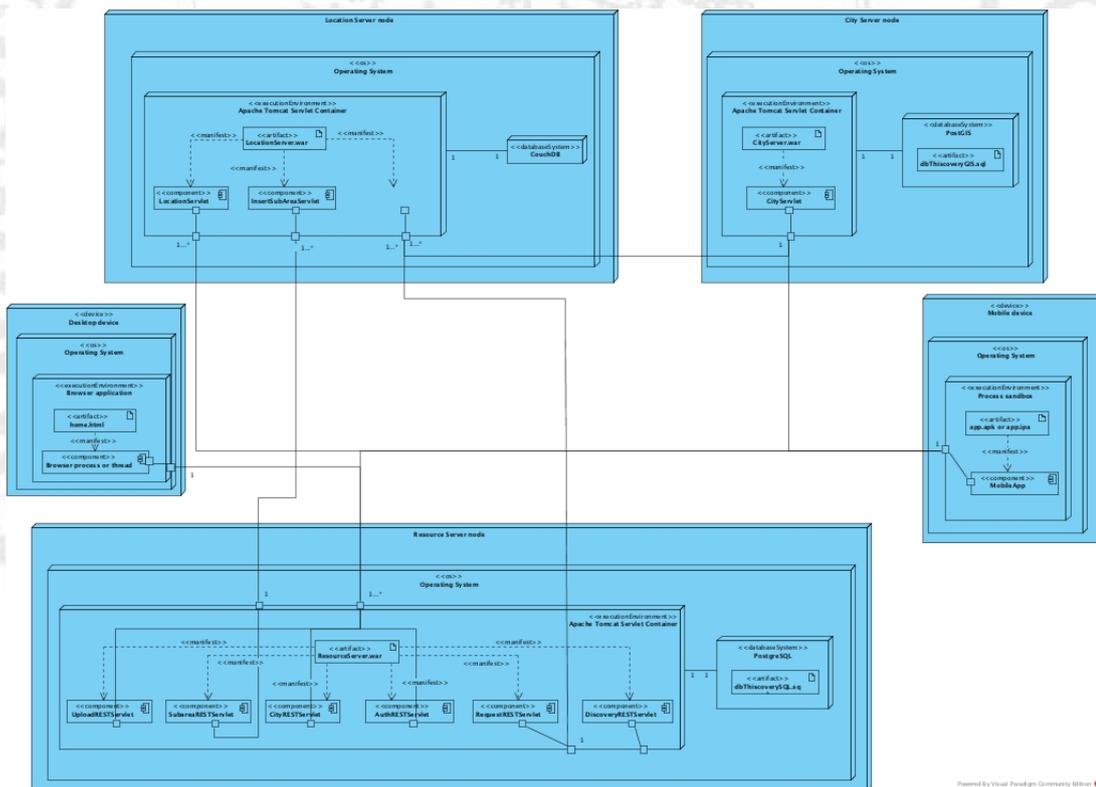
La fase finale della progettazione ha visto l'organizzazione dei sottosistemi individuati negli artifacts riportati di seguito.

I client per utente semplice, collaboratore ed operatore sono stati fisicamente realizzati deployandoli in un unico artifact come applicazione mobile che offre accesso alle funzionalità a seconda del ruolo dell'utente loggato.

4.9.1 Descrizione

La presente viewtype permette di definire i componenti fisicamente allocati, eventuali dipendenze da file di configurazione, eventuale specifica di ambienti di esecuzione ed in che modo tali componenti rilasciati debbano essere effettivamente distribuiti per la messa in esercizio del sistema.

La molteplicità non è riferita ai porti, ma agli elementi cui questi sono collegati.



Capitolo 5

Filtraggio e ranking

La seguente sezione è dedicata alla descrizione degli algoritmi di filtering basato sull'azimut e di ranking delle immagini filtrate.

Il contesto in cui si colloca l'applicazione del filtraggio e del ranking è quello della manipolazione dei risultati ottenuti dalla ricerca su posizione.

Si può infatti descrivere il procedimento di selezione dei punti di interesse da mostrare nel seguente modo:

1. Filtraggio su posizione: dato l'insieme di punti S e nota la posizione utente p , viene selezionato il sottoinsieme S' dei punti che non distano più di un valore prefissato da p
2. Filtraggio su azimut: noti azimut ed ampiezza del campo di visione utente (FOV), è selezionato il sottoinsieme S'' di S' dei punti compresi nel FOV
3. Ranking: dato l'insieme S'' ed una funzione f di visibilità dipendente dall'azimut e dalla posizione utente, vengono ordinati gli elementi di S'' in maniera decrescente in base ad f

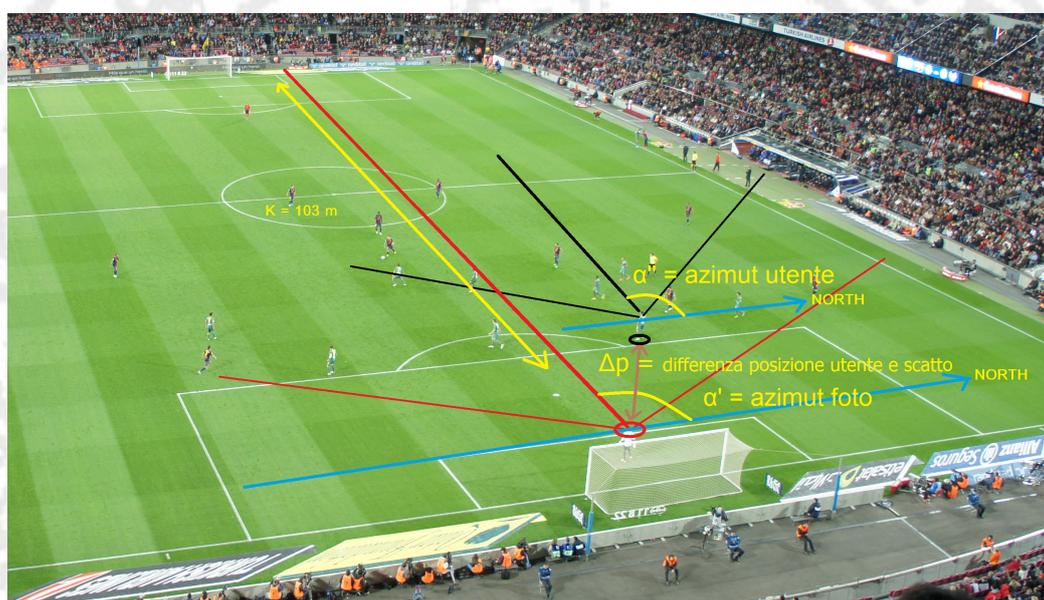
Mentre il primo filtraggio avviene tramite l'esecuzione della query geografica sul GIS, il filtraggio sull'azimut ed il ranking sono eseguiti direttamente sui client.

Sono illustrate di seguito nel dettaglio le soluzioni avanzate e si rimandano a sviluppi futuri eventuali migliorie e i confronti delle stesse con le attuali proposte.

5.1 Filtraggio delle immagini candidate

L'operazione di filtraggio per i punti di interesse candidati è relativa alla fase di selezione delle immagini che sono considerate **valide** come risultato ad una richiesta, noti posizione ed orientamento del dispositivo client.

Di seguito è riportata una immagine che mostra graficamente le informazioni tenute in considerazione per l'operazione di filtraggio. Si supponrà che il campo visivo del difensore rappresenti quello dell'utente, che il campo visivo del portiere corrisponda a quello della foto e che il POI sia la porta avversaria.



La foto, come detto in precedenza, sarà anch'essa caratterizzata da una posizione p_{photo} (punto in cui si trova il portiere) e da un orientamento (azimut α'). Siano, invece, p_{user} e α'' rispettivamente la posizione e l'orientamento dell'utente.

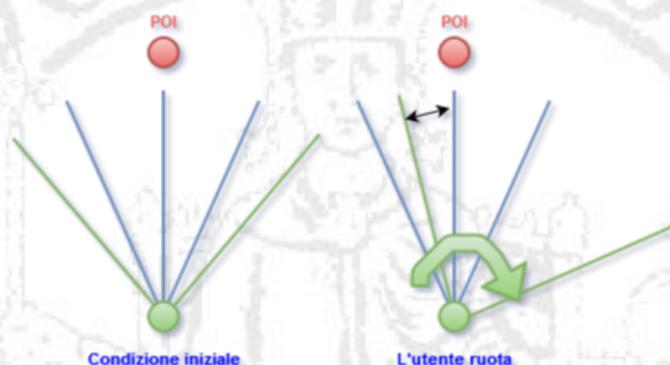
Le prime due grandezze prese in considerazione sono:

- $\Delta\alpha = \alpha' - \alpha''$;
- $\Delta p = d = \text{distance}(p_{photo}, p_{user})$;

La terza grandezza tenuta in considerazione è la distanza di focus, k (indicata dalla freccia di colore giallo), che rappresenta la distanza tra il punto di scatto ed il POI. E' pertanto un attributo associato alla foto.

5.1.1 Variazione massima di angolo a distanza nulla

La prima condizione esposta è relativa alla differenza degli azimut, $\Delta\alpha$. Consideriamo il punto di scatto della foto p_{photo} ed la posizione utente p_{user} e supponiamo che $p_{photo} = p_{user}$. Inoltre immaginiamo che l'utente abbia inizialmente anche lo stesso orientamento della foto e valutiamo lo spostamento (rotazione) massimo in corrispondenza del quale l'utente perde di vista il POI, ossia quando il POI esce dal campo visivo dell'utente:



Nell'immagine mostrata, l'utente (verde) può ruotare della quantità mostrata dalla freccia bidirezionale prima di perdere visione del POI. Andando a quantificare quanto osservato, si ha che la rotazione massima dell'utente è pari esattamente a metà del campo di visione dello stesso utente, ma dacché l'orientamento iniziale dell'utente era pari a quello della foto, lo spostamento compiuto dall'utente è pari proprio a $\Delta\alpha$.

Pertanto si ha:

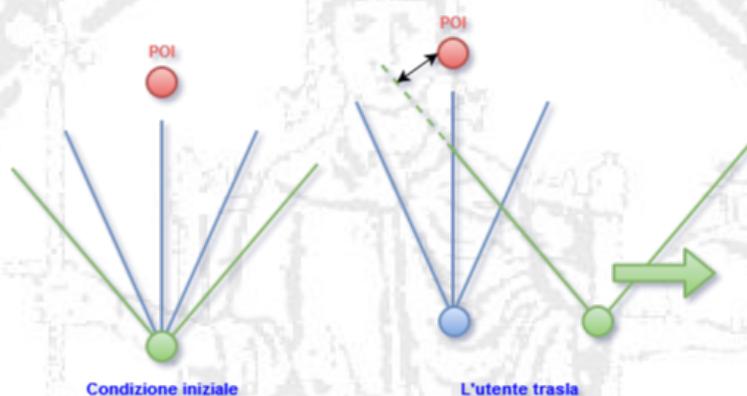
$$\Delta\alpha_{max} = \frac{fieldAngle_{user}}{2}$$

Il vincolo relativo diventa dunque:

$$(1) |\Delta\alpha| < \Delta\alpha_{max}$$

5.1.2 Distanza massima senza rotazione

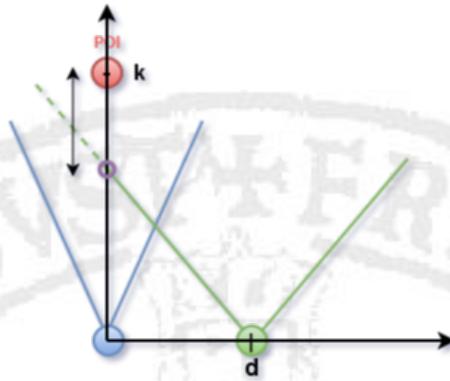
La seconda condizione esposta è relativa alla massima distanza, \mathbf{d} , tra la posizione utente e quella di scatto della foto affinché il POI risulti ancora visibile all'utente. Consideriamo il punto di scatto della foto p_{photo} ed la posizione utente p_{user} e supponiamo che $\Delta\alpha = 0$, ossia l'utente non effettui rotazioni ma solo traslazioni. Inoltre immaginiamo che l'utente abbia inizialmente anche la stessa posizione dello scatto della foto e valutiamo lo spostamento (traslazione) massimo in corrispondenza del quale l'utente perde di vista il POI, ossia quando il POI esce dal campo visivo dell'utente:



Nell'immagine mostrata, l'utente (verde) può traslare della quantità mostrata dalla freccia bidirezionale prima di perdere visione del POI. Per a quantificare quanto osservato, si consideri la posizione della foto come l'origine di un sistema di riferimento in cui la direzione dell'asse delle ordinate coincide con la direzione di focus della foto e l'asse delle ascisse, ad esso perpendicolare, è crescente nel verso di destra.

Avendo un sistema di riferimento, è possibile calcolare il punto in cui l'utente perde visione del POI come l'intersezione tra il prolungamento del segmento del campo di visione utente e la direzione di focus della foto. Se tale intersezione risulta essere minore di \mathbf{k} (**distanza di focus**), l'utente vede il POI.

Pertanto l'utente si può spostare lungo l'asse delle ascisse di una quantità pari alla distanza per cui l'intersezione avviene nel punto di coordinate $(0, k)$.



Per indicare la retta di interesse, consideriamone una generica espressione:

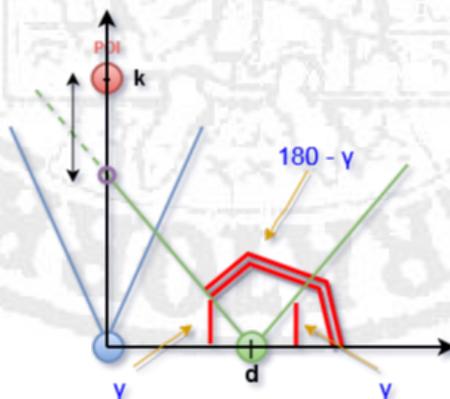
$$y = mx + q$$

Dall'immagine mostrata, si può vedere che q è pari a 0. Questo è dovuto al fatto che nell'esempio in esame abbiamo assunto implicitamente un solo grado di libertà spostandoci unicamente lungo "l'asse x " e non contemplando (ancora) un generico spostamento sul piano.

Per definire la retta, dunque, bisogna trovare m , il coefficiente angolare che può essere calcolato conoscendo l'angolo di campo dell'utente.

Infatti, assumendo che gli angoli siano in gradi e detto $fieldAngle_{user}$ l'angolo di campo dell'utente, l'angolo che si forma tra il segmento del campo di visione dell'utente e l'asse x , γ , è la metà dell'angolo supplementare di $fieldAngle_{user}$, ossia:

$$\gamma = \frac{180 - fieldAngle_{user}}{2}$$



A questo punto m è calcolato come la tangente dell'angolo trovato. Dunque, si avrà:

$$\begin{aligned} m &= \tan(180 - \gamma) \\ y &= \tan(180 - \gamma)x \\ y &= \tan\left(180 - \frac{180 - \text{fieldAngle}_{user}}{2}\right)x = \tan\left(90 + \frac{\text{fieldAngle}_{user}}{2}\right)x = \\ &= \cot\left(-\frac{\text{fieldAngle}_{user}}{2}\right)x = -\cot\left(\frac{\text{fieldAngle}_{user}}{2}\right)x \end{aligned}$$

Lo spostamento massimo in corrispondenza del quale la retta interseca il POI è:

$$x_{max} = -\frac{k}{\cot\left(\frac{\text{fieldAngle}_{user}}{2}\right)}$$

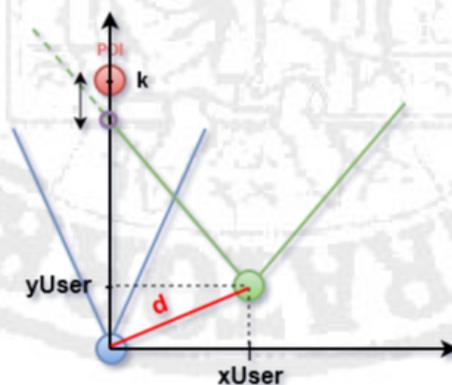
E' da notare che se avessimo considerato lo spostamento dell'utente verso sinistra, avremmo avuto, come intuibile, l'analogo risultato simmetrico. In quel caso, per il calcolo di m , anziché considerare $\tan(180-\gamma)$, avremmo considerato direttamente $\tan(\gamma)$, ottenendo:

$$\begin{aligned} y &= \tan\left(\frac{180 - \text{fieldAngle}_{user}}{2}\right)x = \tan\left(90 - \frac{\text{fieldAngle}_{user}}{2}\right)x = \\ &= \cot\left(\frac{\text{fieldAngle}_{user}}{2}\right)x \end{aligned}$$

In definitiva, dato che dello spostamento massimo abbiamo interesse a conoscere soltanto il modulo (e non il verso), si ha che:

$$d_{max} = \frac{k}{\left|\cot\left(\frac{\text{fieldAngle}_{user}}{2}\right)\right|}$$

Nel caso in cui l'utente si sposti contemporaneamente anche lungo l'asse y , rimane valido il discorso sulla pendenza della retta ma si aggiunge all'espressione della retta uno spiazzamento q pari proprio alla distanza su y dal punto di scatto della foto:



Dato che in questo caso la distanza dipende da entrambe le differenze (x, y) dell'utente rispetto al punto di scatto della foto, anziché fornire una condizione di limite sulla distanza, può risultare più comodo ragionare sulle coordinate stesse.

Infatti, dette x_{user} e y_{user} le distanze dell'utente dalla posizione di scatto della foto, rispettivamente sugli assi x ed y, la condizione per cui l'utente veda il POI è:

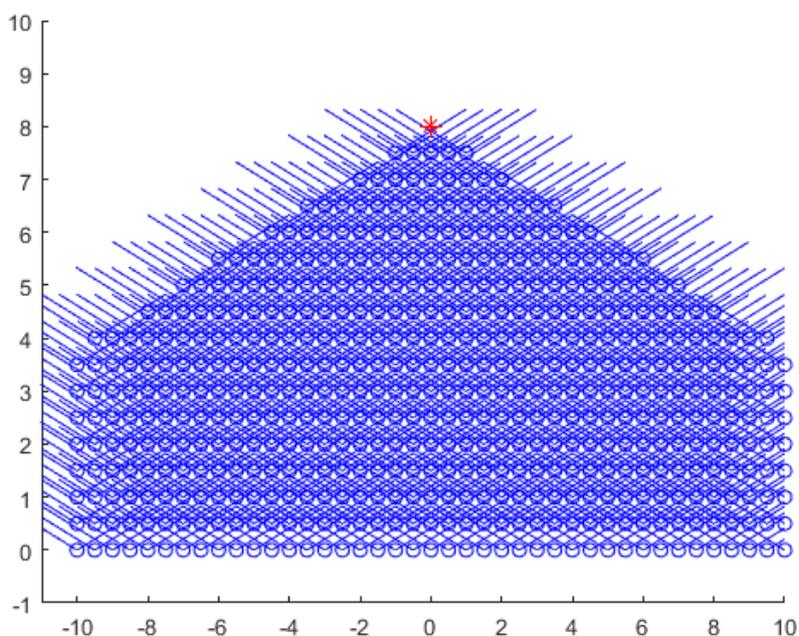
$$(2) -\cot(\theta)x_{user} + y_{user} < k$$

$$\text{con } \theta = \begin{cases} \frac{fieldAngle_{user}}{2} & x_{user} \geq 0 \\ -\frac{fieldAngle_{user}}{2} & x_{user} < 0 \end{cases}$$

dove k è la distanza di focus del POI dalla posizione di scatto della foto.

Per un riscontro visivo della relazione (2), di seguito è riportato un grafico generato con uno script Matlab relativo a tutte le possibili posizioni valide degli utenti rispetto ad una data foto, con:

- $fieldAngle = 135$ gradi;
- $k = 8$ metri;
- Spostamento verso destra = 10 metri;
- Spostamento verso sinistra = 10 metri;

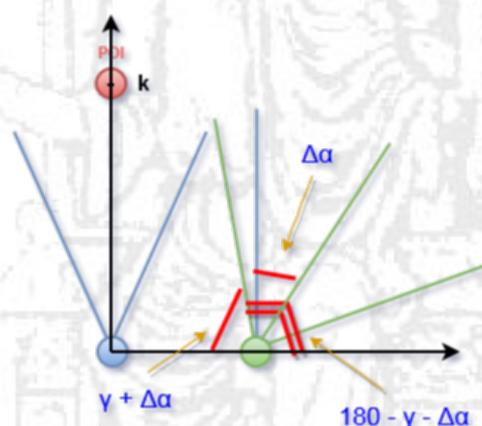


Si notino il POI indicato da un asterisco rosso posto nel punto di coordinate (0, 8) e al di sotto di esso tutti (e soli) gli utenti (con i relativi campi di visione) in grado di vedere il POI. Benché nello script fossero esaminate tutte le combinazioni di posizioni con x compreso tra -10 e 10 (con un incremento di 0.5) ed y compreso tra 0 e 10 (con un incremento di 0.5), la condizione (2) limita la visualizzazione ai soli utenti nel cui campo di visione rientra il POI.

5.1.3 Vincolo nel caso generale

Nel caso in cui l'utente abbia contemporaneamente una differenza di orientamento e di posizione rispetto al punto della foto, è possibile combinare in modo opportuno i risultati ottenuti per i casi esaminati singolarmente.

Per tenere in conto il $\Delta\alpha$ possiamo ricondurci allo scenario visto nel caso di sola traslazione osservando gli effetti di una operazione di rotazione:



Con una differenza di azimut, l'angolo utilizzato precedentemente per il calcolo del coefficiente angolare della retta comprendente il segmento relativo al campo di visione utente, indicato come γ , risulta "allargato" di quantità pari esattamente alla differenza di azimut.

Pertanto, rispetto al caso precedente, la tangente sarà applicata ad un angolo più piccolo di $\Delta\alpha$.

$$m = \tan(180 - \gamma - \Delta\alpha)$$

$$y = \tan(180 - \gamma - \Delta\alpha)x + q$$

$$y = \tan\left(180 - \frac{180 - \text{fieldAngle}_{user}}{2} - \Delta\alpha\right)x + q = \tan\left(90 + \frac{\text{fieldAngle}_{user}}{2} - \Delta\alpha\right)x + q =$$

$$= \cot\left(\Delta\alpha - \frac{\text{fieldAngle}_{user}}{2}\right)x + q$$

Se però, in questo caso, l'utente si trova a sinistra del punto di scatto della foto (x_{user} negativo), l'angolo da considerare per il calcolo di m è quello a destra dell'utente, che è pari a $\gamma - \Delta\alpha$:

$$\begin{aligned} m &= \tan(\gamma - \Delta\alpha) \\ y &= \tan(\gamma - \Delta\alpha)x + q \\ y &= \tan\left(\frac{180 - fieldAngle_{user}}{2} - \Delta\alpha\right)x + q = \tan\left(90 - \frac{fieldAngle_{user}}{2} - \Delta\alpha\right)x + q = \\ &= \cot\left(\Delta\alpha + \frac{fieldAngle_{user}}{2}\right)x + q \end{aligned}$$

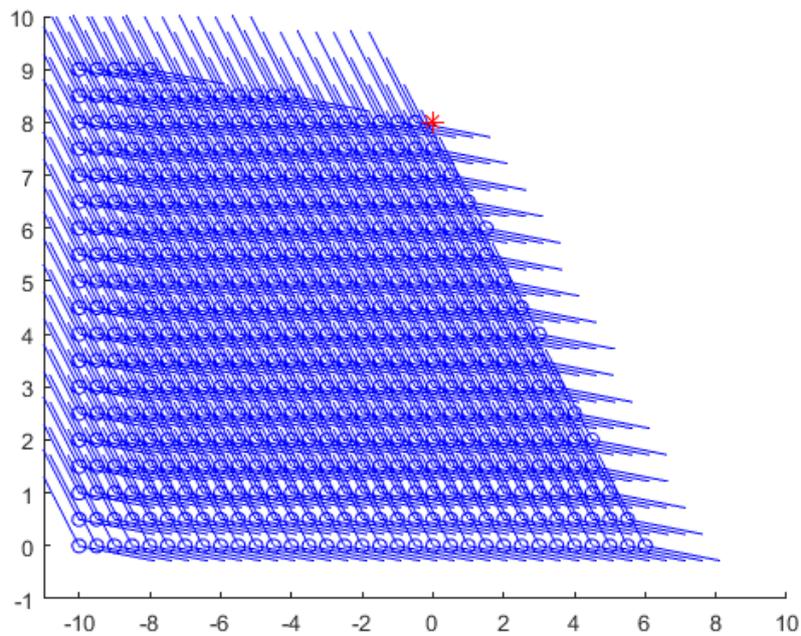
Indicando con x_{user} e y_{user} le differenze sugli assi x ed y dell'utente rispetto al punto di scatto della foto, la traslazione sarà dunque vincolata dalla condizione:

$$(2) \quad -\cot(\theta)x_{user} + y_{user} < k$$

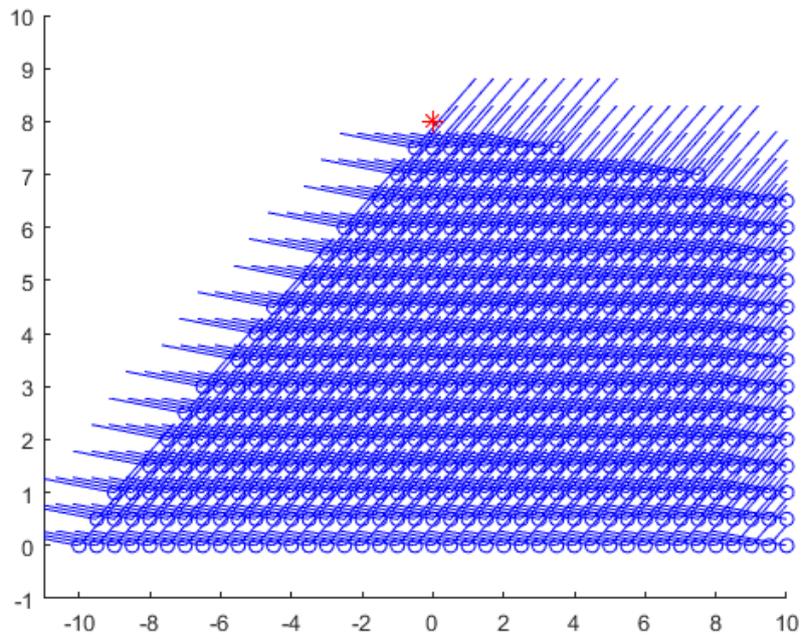
$$dove \theta = \begin{cases} \Delta\alpha - \frac{fieldAngle_{user}}{2} & x_{user} \geq 0 \\ \Delta\alpha + \frac{fieldAngle_{user}}{2} & x_{user} < 0 \end{cases}$$

Per un riscontro visivo della relazione (2), di seguito è riportato un grafico generato con uno script Matlab relativo a tutte le possibili posizioni valide degli utenti rispetto ad una data foto, con:

- $\Delta\alpha = 30$ gradi in senso orario;
- $fieldAngle = 135$ gradi;
- $k = 8$ metri;
- Spostamento verso destra = 10 metri;
- Spostamento verso sinistra = 10 metri;



Cambiando la variazione di azimut al valore $\Delta\alpha = 15$ gradi in senso antiorario (-15) si ottiene, invece:



5.2 Ranking

Selezionate le foto valide, la fase successiva è relativa al calcolo di un punteggio da assegnare a ciascuna di esse per determinare la "migliore", secondo un'accezione che punta a massimizzare il modo in cui la foto riflette ciò che effettivamente l'utente può osservare nella posizione in cui si trova e secondo il proprio orientamento.

La funzione per la generazione di un punteggio dovrà dunque dipendere da:

- differenza di orientamento, $\Delta\alpha$;
- distanza dal punto di scatto, d ;
- distanza di focus, k ;

5.2.1 Scelta della funzione di ranking

Per costruire una funzione di ranking bisogna definirne in primis l'obiettivo.

Nel caso in esame, i requisiti che la funzione deve rispettare sono:

- a parità di d e k , crescere per valori decrescenti di $\Delta\alpha$ (preferenza di orientamento);
- a parità di k e $\Delta\alpha$, crescere per valori decrescenti di d (preferenza di vicinanza al punto di scatto della foto);

Aggiungiamo inoltre il vincolo che il codominio per la funzione di ranking sia: $[0, 1]$. Si noti che tra i requisiti della funzione di ranking non compare la distanza dal punto di focus, in quanto la si associa ad una scelta progettuale, per la quale si potrebbe, in un determinato contesto, preferire di dare priorità ai POI più vicini o viceversa.

La prima funzione di ranking analizzata è:

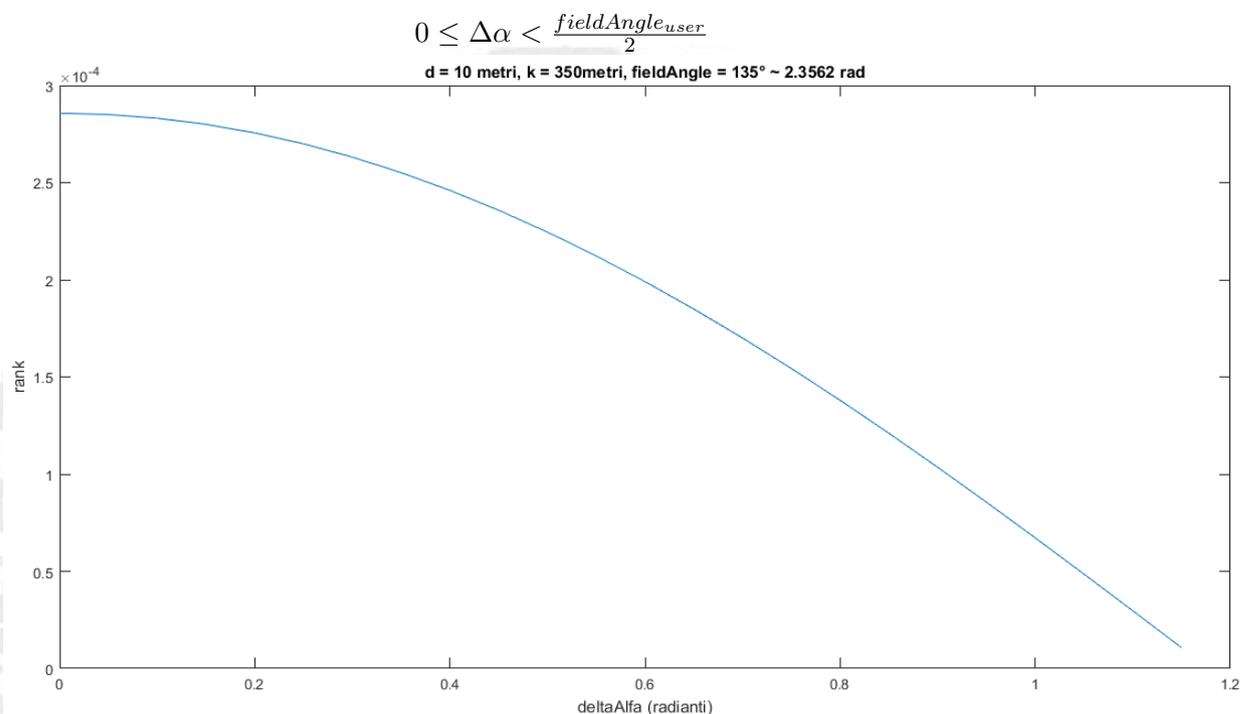
$$\text{rank}(d, k, \Delta\alpha) = \frac{1}{dk} \cos(\lambda \Delta\alpha)$$

dove λ è un fattore di adattamento all'angolo di campo e , supposto che fieldAngle_{user} sia espresso in gradi, vale:

$$\lambda = \frac{180}{\text{fieldAngle}_{user}}$$

Di seguito sono riportati i grafici ottenuti dalla funzione di ranking fissando 2 variabili e facendone variare una di volta in volta.

Variazione di $\Delta\alpha$:

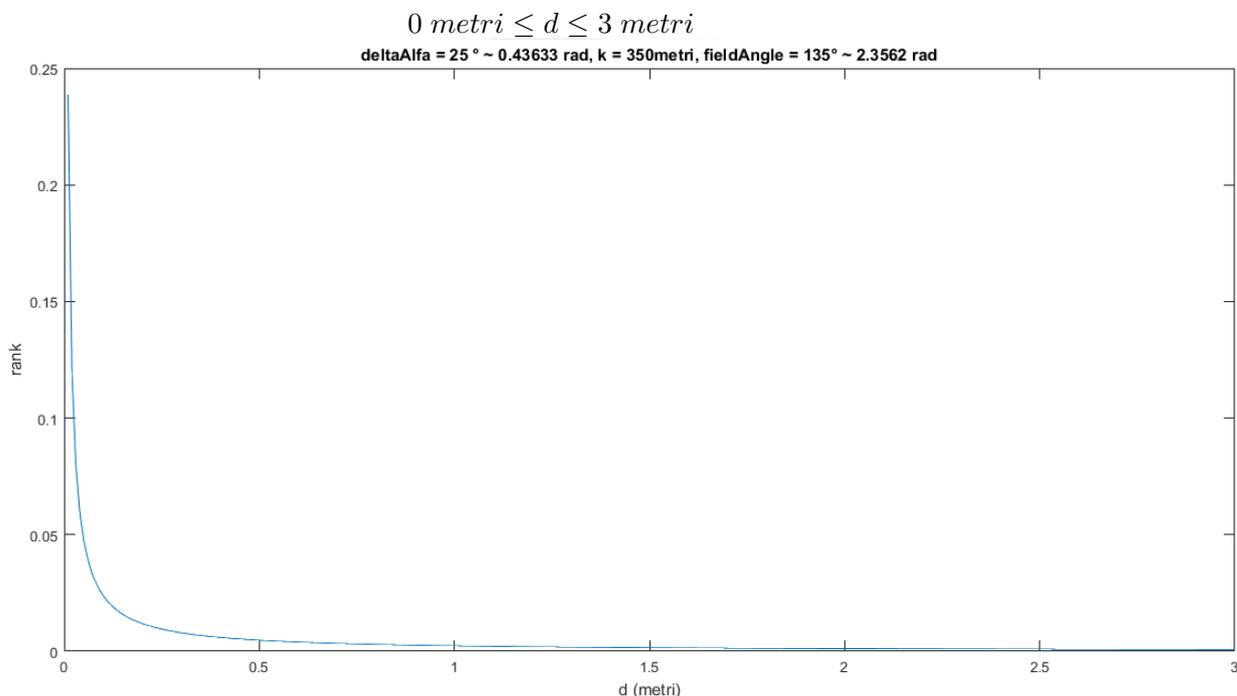


La variazione del ranking in funzione della differenza d'angolo è:

$$\frac{\partial \text{rank}(d, k, \Delta\alpha)}{\partial \Delta\alpha} = -\frac{\lambda}{dk} \sin(\lambda \Delta\alpha)$$

Si può vedere che la funzione rispetta il requisito sulla monotonia decrescente: abbiamo infatti una costante negativa che moltiplica il seno che in questo contesto è sempre crescente, in quanto il suo argomento è limitato ad un valore inferiore a 90 gradi. Inoltre risulta avere un andamento idoneo ai fini di generare rank che si discostano poco per input piccoli e con un gap che cresce al crescere dei valori di input.

Variazione di **d**:



La variazione del ranking in funzione della distanza dal punto di scatto è:

$$\frac{\partial \text{rank}(d, k, \Delta\alpha)}{\partial d} = -\frac{1}{d^2 k} \cos(\lambda \Delta\alpha)$$

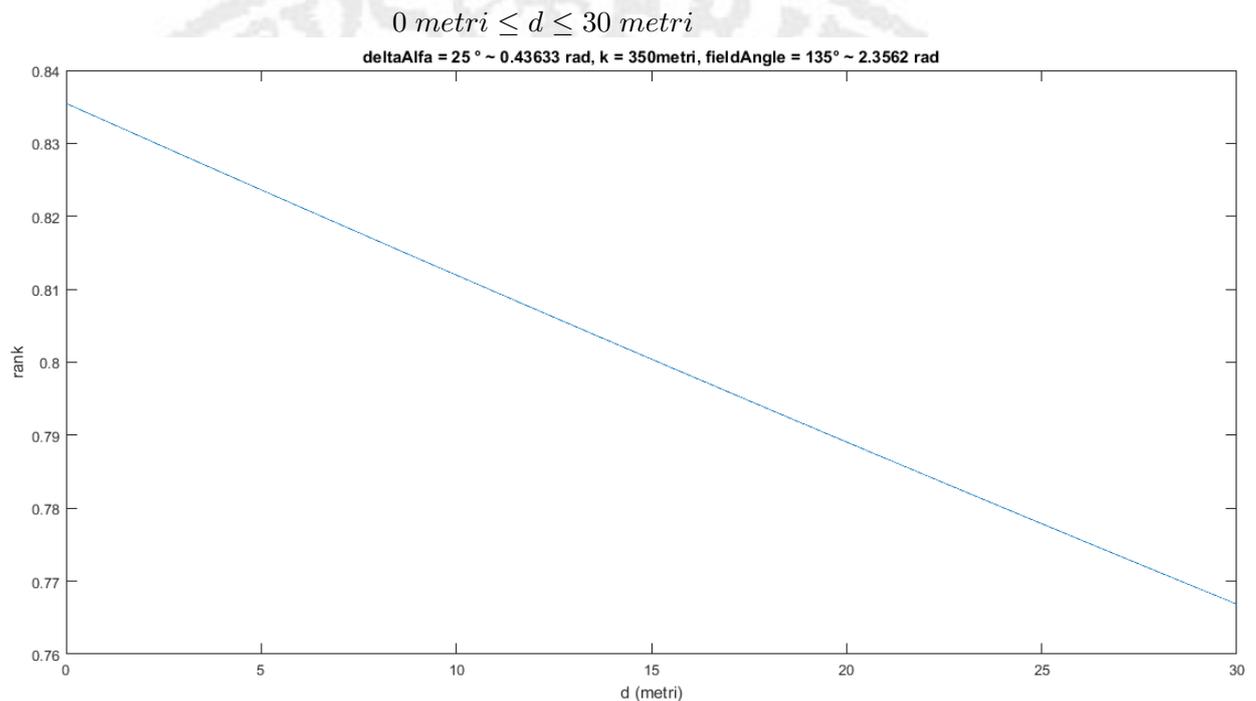
ossia le variazioni sui rank sono inversamente proporzionali al quadrato della distanza. Ciò comporta una rapida "caduta" a valori di rank bassi. Notiamo che contrariamente a quanto avviene per le variazioni sull'angolo di orientamento, i rank hanno tasso di variazione molto elevato per le distanze più piccole mentre tendono ad assestarsi a partire da un punto approssimativamente pari al ginocchio della curva in figura. Inoltre la funzione non risulta definita nello 0, tendendo a $+\infty$, non rispettando il vincolo (benché non stringente) sul codominio.

Discorso perfettamente analogo è fatto per k , in quanto la funzione di ranking proposta dipende da k e da d allo stesso modo.

Per risolvere tali problemi si può pensare ad una funzione di ranking della forma:

$$rank(d, k, \Delta\alpha) = e^{-\frac{d}{k}} \cos(\lambda \Delta\alpha)$$

Il cui grafico imponendo la variazione rispetto a d , fissando k e $\Delta\alpha$ è:



La variazione del ranking in funzione della distanza dal punto di scatto è:

$$\frac{\partial rank(d, k, \Delta\alpha)}{\partial d} = -\frac{e^{-\frac{d}{k}}}{k} \cos(\lambda \Delta\alpha)$$

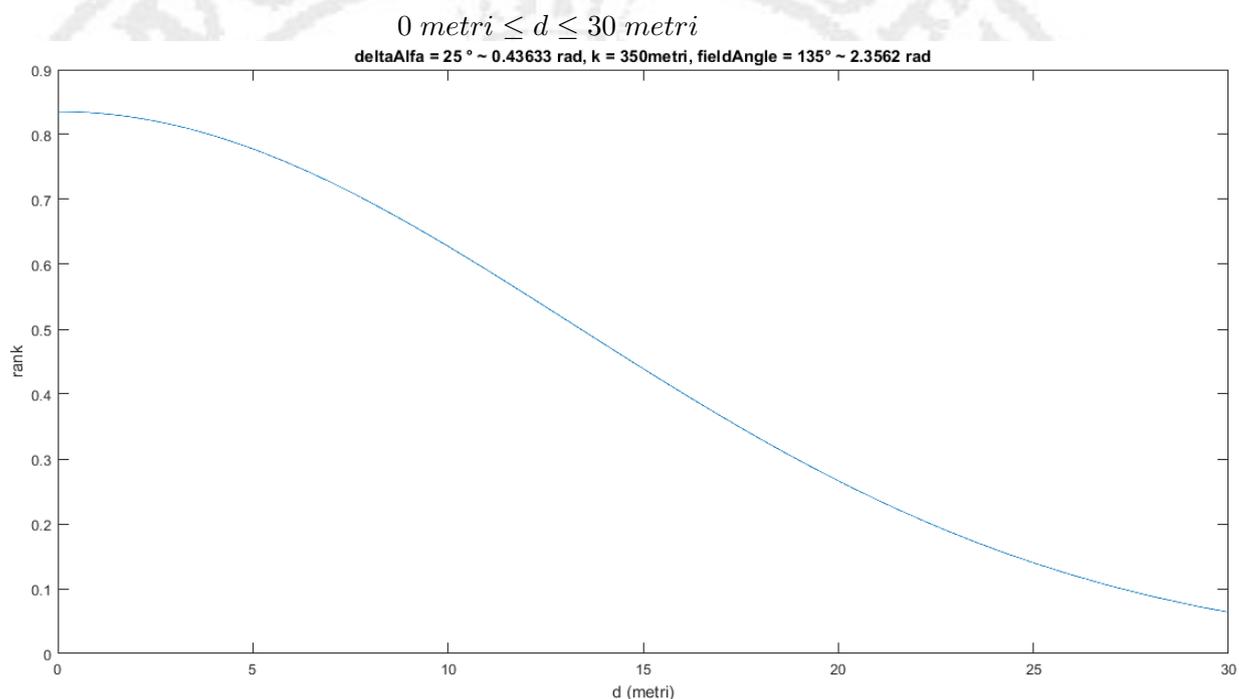
In questo caso si nota subito che è possibile estendere il range delle distanze ad intervalli più grandi di oltre un ordine di grandezza rispetto alla precedente funzione di ranking: ciò grazie al fatto che il tasso di variazione è ridotto di un fattore k , il quale nell'esempio è fissato a 350 metri, ma non sarebbe inusuale considerarlo addirittura maggiore in un caso reale.

Ciò porta ad una discesa troppo "lenta" dei rank che si traduce in una cattiva capacità di discriminare le reali distanze di interesse nei casi comuni.

Una modifica alla funzione di ranking precedente che può risultare molto più utile per la discriminazione sulle distanze, mantenendo i vantaggi legati al rispetto del vincolo del codominio e del requisito di monotonia decrescente per gli intervalli di interesse degli input, è la seguente:

$$rank(d, k, \Delta\alpha) = e^{-\frac{d^2}{k}} \cos(\lambda \Delta\alpha)$$

Il cui grafico imponendo la variazione rispetto a d , fissando k e $\Delta\alpha$ è:



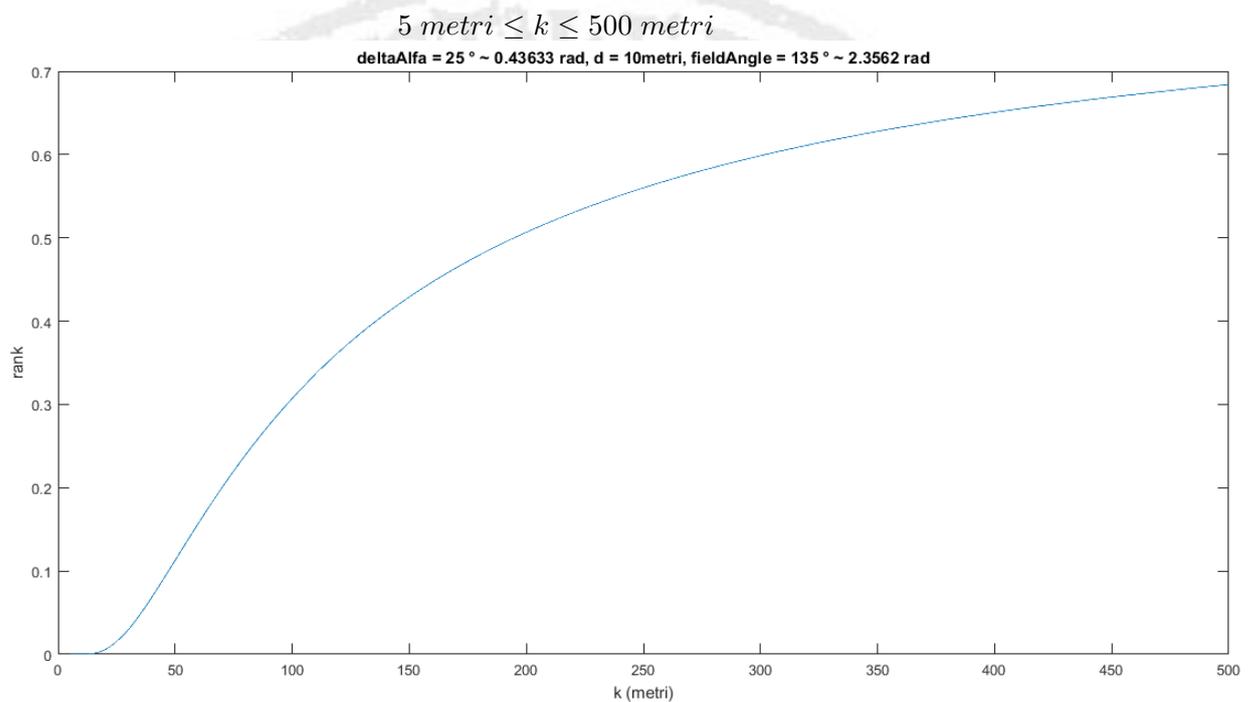
La variazione del rank in funzione della distanza dal punto di scatto è:

$$\frac{\partial rank(d, k, \Delta\alpha)}{\partial d} = -2\frac{d}{k} e^{-\frac{d^2}{k}} \cos(\lambda \Delta\alpha)$$

L'idea è quella di "accelerare" la variazione dei valori di ranking riducendo l'abbattimento del fattore k utilizzando la distanza stessa come fattore moltiplicativo. Infatti si può osservare dal grafico come utilizzando lo stesso intervallo per la distanza ($[0, 30]$) si ricopra un insieme di valori di ranking molto più ampio, passando dall'intervallo $[0.765, 0.835]$ all'intervallo $[0.08, 0.84]$, con una percentuale di copertura del codominio di circa 76%.

L'ultimo step da effettuare è relativo alla valutazione dell'impatto della variazione di k sulla funzione di ranking progettata.

Variazione di k :



La variazione del ranking in funzione della distanza di focus è:

$$\frac{\partial \text{rank}(d, k, \Delta\alpha)}{\partial k} = \frac{d^2}{k^2} e^{-\frac{d^2}{k}} \cos(\lambda \Delta\alpha)$$

Come si può osservare dalla derivata parziale, la monotonia della funzione di ranking è strettamente crescente. Ciò si traduce in una prioritizzazione dei POI più lontani che potrebbe, pertanto, favorire la scelta di panorami, piazze e più in generale luoghi di interesse visibili da lunghe distanze, risultando un approccio utile nel caso di viste dall'alto dell'utente.

Capitolo 6

Capacity test

Il presente capitolo è dedicato ad illustrare gli esperimenti condotti, la metodologia seguita, mostrando infine i risultati ottenuti.

L'obiettivo dei seguenti esperimenti è stabilire, noto un modello di macchina target, il carico per il singolo nodo - in termini di numero di punti presenti all'interno del GIS - per cui risultino soddisfatti i requisiti non funzionali e risultino altresì minimizzati i costi, valutando il numero minimo di nodi necessari al variare del numero di punti presenti nel sistema.

La macchina target sulla quale sono state effettuate le misurazioni presenta le seguenti caratteristiche:

- OS Template: Ubuntu 16.04 x64
- Core: 1
- RAM: 1GB
- Spazio disco: 20GB SSD

Tale scelta non è arbitraria ma rispecchia una comune configurazione base per servizi cloud.

Sviluppi futuri del lavoro prevedono l'ottimizzazione dei costi ripetendo gli esperimenti, valutando differenti macchine target e considerando il costo per macchina.

6.1 Metodologia

Gli esperimenti sono stati condotti nel seguente modo:

1. Sono state definite le dimensioni da considerare in termini di numeri di punti nel GIS
2. Per ciascun valore individuato è stata creata una tabella in PostgreSQL contenente tale numero di punti, generati casualmente all'interno di un'area
3. Per ciascuna tabella creata è stato effettuato un capacity test misurando throughput ed elapsed time al fine di individuare la knee capacity della macchina target quando il GIS contiene il carico in esame
4. I risultati ottenuti sono stati plottati in due grafici che mappano uno specifico numero di punti del GIS al throughput e all'elapsed time associati alla knee capacity corrispondente al numero di punti considerato

Ricordando che la **knee capacity** è il punto di carico di un sistema oltre il quale a fronte di un modesto aumento di throughput si ottiene un forte aumento della latenza del sistema (e quindi del response time), è stato scelto quest'ultimo come parametro di interesse in quanto in condizioni ottimali un server dovrebbe lavorare al di sotto di tale carico per garantire un buon throughput con un response time ritenuto accettabile.

L'analisi condotta descriverà, dunque, i risultati ottenuti per ciascuna delle dimensioni individuate.

6.2 Dimensioni dei numeri di punti

Le dimensioni considerate per il numero di punti sono le seguenti:

- 50 punti
- 500 punti
- 5000 punti
- 50000 punti
- 500000 punti

La scelta di procedere con uno step di fattore moltiplicativo 10 tra una dimensione campione e la successiva è legato al fatto che il numero di server, ottenuto tramite la relazione

$$n_{server} = \frac{n_{puntiDesiderato}}{n_{puntiPerServer}}$$

varierebbe in termini di ordini di grandezza, piuttosto che di singole unità.

Può risultare, infatti, di maggiore interesse, ai fini della valutazione di costi e scalabilità del sistema, riuscire ad estrapolare un ordine di grandezza per le risorse richieste (in questo caso i server) piuttosto che un preciso numero, il quale risulterebbe maggiormente dipendente dalla scelta fatta inizialmente della macchina target.

6.3 Generazione dei punti

La generazione dei punti è stata effettuata tramite uno script PL/pgSQL (Procedural Language / PostgreSQL) e l'utilizzo di stored procedures per estrarre un punto geografico di coordinate casuali all'interno di una specifica area rettangolare. E' stato dunque creato un nuovo schema contenente la tabella di interesse e popolata quest'ultima con i punti generati.

Da notare che la scelta dell'area (e quindi la densità di punti) non risulta importante ai fini delle misurazioni, in quanto gli esperimenti sono stati condotti in modo da effettuare query che restituissero sempre un insieme vuoto di punti.

L'osservazione importante in questo contesto era garantire un'uniformità nella risposta del server evitando di aggiungere un'ulteriore componente aleatoria dovuta all'effettivo numero di punti presenti nell'area richiesta.

Se la risposta avesse previsto un numero di punti maggiore di zero, nell'ipotesi che tale numero di punti fosse lo stesso per tutte le richieste, ciò avrebbe comportato esclusivamente un incremento nell'elapsed time¹ di un valore costante.

Si riporta un riscontro grafico del risultato di generazione per il caso di 500 punti:

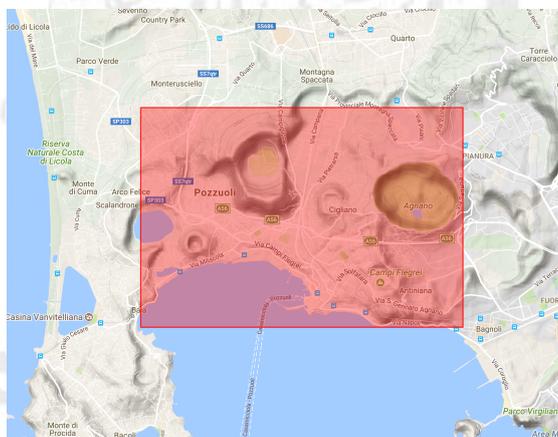


Figura 6.1: Estensione considerata per posizionare i 500 punti

¹Definizione di elapsed time dal glossario di Apache JMeter, <http://jmeter.apache.org/usermanual/glossary.html> : "*JMeter measures the elapsed time from just before sending the request to just after the last response has been received. JMeter does not include the time needed to render the response, nor does JMeter process any client code, for example Javascript.*"

Mentre i punti effettivamente generati risultano:

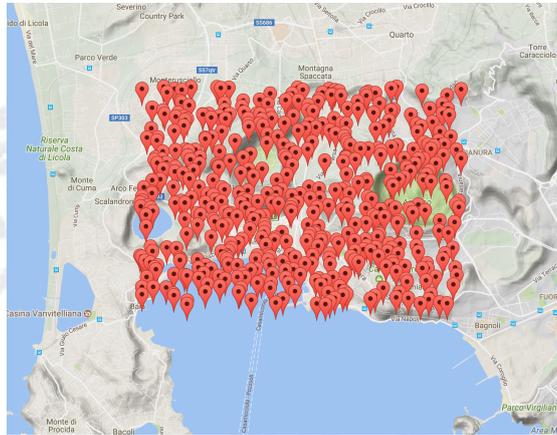


Figura 6.2: Risultato dello script di generazione nell'area considerata

6.4 Misurazioni

La terza fase è relativa alle misurazioni.

Per lo scopo è stata implementata una servlet di test e deployata presso il nodo in esame.

Tale servlet, in base al valore di un parametro in input corrispondente al numero di punti, effettua la query geografica sullo schema e sulla tabella richieste del GIS.

L'analisi del sistema è stata effettuata utilizzando il tool JMeter per sollecitare con un carico crescente di richieste il server. Potendo impostare il numero di utenti (threads) che effettuano richieste contemporaneamente ed il throughput per utente - in termini di numero di richieste al minuto - è possibile effettuare misurazioni al fine di monitorare parametri come:

- Numero di campioni nell'esperimento
- Elapsed time medio (ms)
- Percentuale di errore nelle risposte
- Throughput (numero di risposte al secondo)
- Numero medio di bytes inviati al secondo
- Numero medio di bytes ricevuti al secondo

Gli elementi di JMeter utilizzati per le misurazioni sono:

- Sampler: componente che svolge l'effettivo lavoro di generazione, invio della richiesta e monitoring del risultato. Ogni sampler genera uno o più risultati campione, ciascuno dei quali presenta gli attributi sopra elencati
- HTTP/HTTPS request: caso particolare di sampler per il protocollo HTTP che definisce le specifiche della richiesta da inviare ad un web server (nome/IP server, porto, metodo, path, encoding e parametri)
- Logic controllers: componente che determina l'ordine di processamento dei Samplers
- Random controller: caso particolare di logic controller che effettua una selezione casuale dei Samplers
- Timer: componente che si occupa di gestire la sincronicità dell'esperimento

- Constant throughput timer: caso particolare di timer utilizzato per garantire uno specifico rate di richieste al minuto da effettuare²

E' stato, quindi, utilizzato un random controller per generare richieste al sistema, mentre il request rate è stato gestito mediante un constant throughput timer.

Impostando il numero di richieste al minuto, JMeter si occupa di garantire tale frequenza gestendo dinamicamente i tempi tra richieste consecutive di ogni utente.

A seconda del numero considerato di punti nel GIS sono stati utilizzati diversi valori del parametro di req/sec da impostare per il constant throughput timer.

Di seguito sono riportati interamente solo alcuni dei risultati delle misurazioni effettuate tramite JMeter, mentre saranno specificati in dettaglio i valori di knee capacity trovati con i relativi throughput ed elapsed time.

Requests/minute	Throughput (resp/sec)	Response time (ms)
20000/minute = 333.3/sec	333.5	56
23000/minute = 383.3/sec	383.3	60
25000/minute = 416.6/sec	415.7	73
28000/minute = 466.6/sec	464.2	82
30000/minute = 500/sec	497.5	95
31000/minute = 516/sec	510.7	215
32000/minute = 533.3/sec	529.9	300
33000/minute = 550/sec	532.1	300
35000/minute = 583/sec	541.2	392
36000/minute = 600/sec	543	488
37000/minute = 616,6/sec	531.1	559
38000/minute = 633.3/sec	522.4	559
40000/minute = 666,6/sec	519.2	720

Tabella 6.2: Risultati delle misurazioni per il GIS con 50 punti

²Dalla documentazione ufficiale di Apache JMeter: "*This timer introduces variable pauses, calculated to keep the total throughput (in terms of samples per minute) as close as possible to a give figure. Of course the throughput will be lower if the server is not capable of handling it, or if other timers or time-consuming test elements prevent it.*"

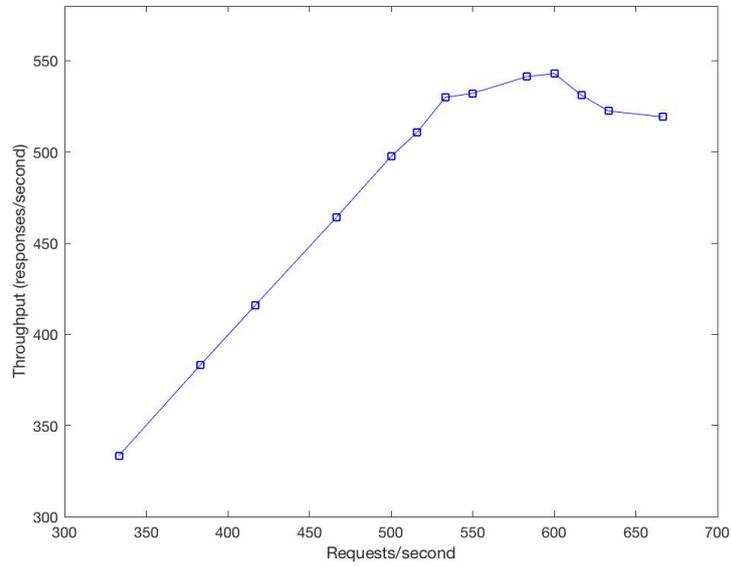


Figura 6.3: Throughput per il GIS con 50 punti

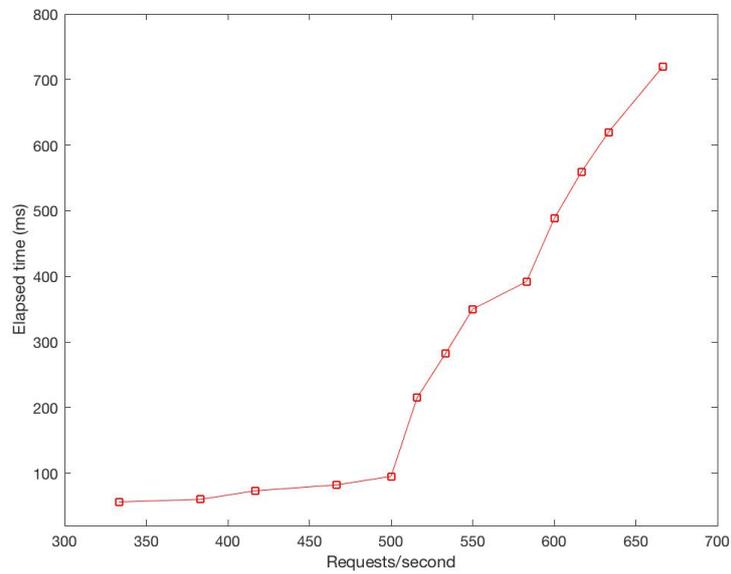


Figura 6.4: Elapsed time per il GIS con 50 punti

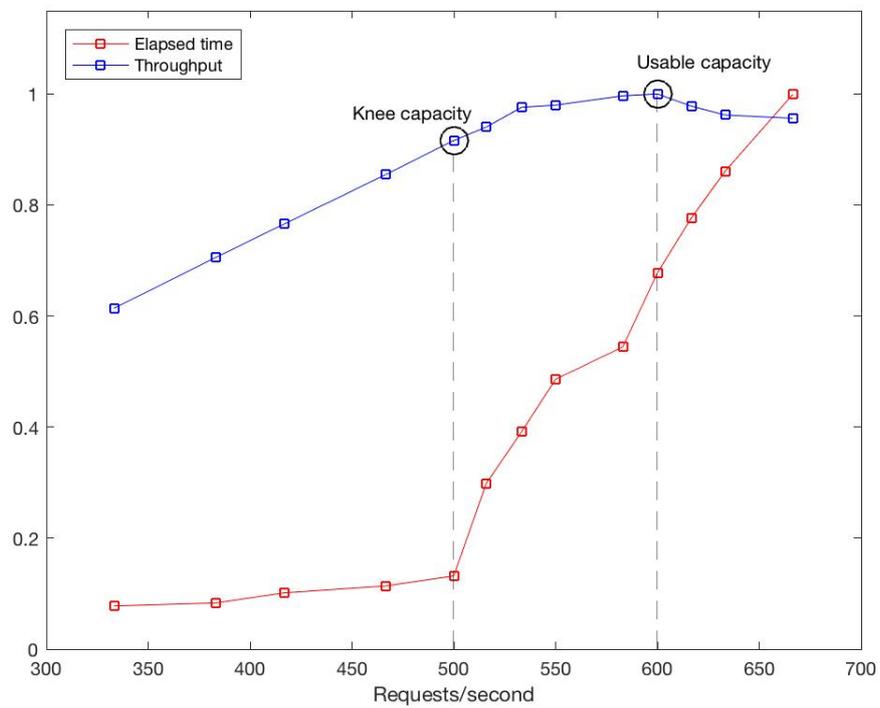


Figura 6.5: Usable e knee capacity per il GIS con 50 punti

Requests/minute	Throughput (resp/sec)	Response time (ms)
500/minute = 8.3/sec	8,3	105
600/minute = 10/sec	10	98
700/minute = 11.6/sec	11,7	101
800/minute = 13.3/sec	13,3	103
900/minute = 15/sec	15	108
1000/minute = 16.6/sec	16,6	106
1100/minute = 18.3/sec	18,3	105
1200/minute = 20/sec	19,9	119
1300/minute = 21.6/sec	21,6	135
1400/minute = 23.3/sec	23,1	306
1500/minute = 25/sec	23,4	475
1600/minute = 26.6/sec	23,5	520
1700/minute = 28.3/sec	23,7	570
1800/minute = 30/sec	23,5	628
1900/minute = 31.6/sec	23,5	694
2000/minute = 33.3/sec	23	765
2100/minute = 35/sec	23,2	788
2200/minute = 36.6/sec	23,1	864

Tabella 6.4: Risultati delle misurazioni per il GIS con 50000 punti

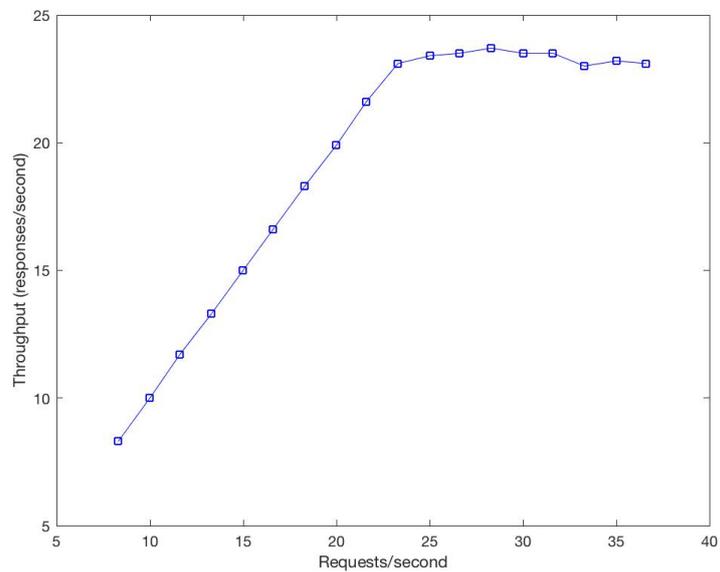


Figura 6.6: Throughput per il GIS con 50000 punti

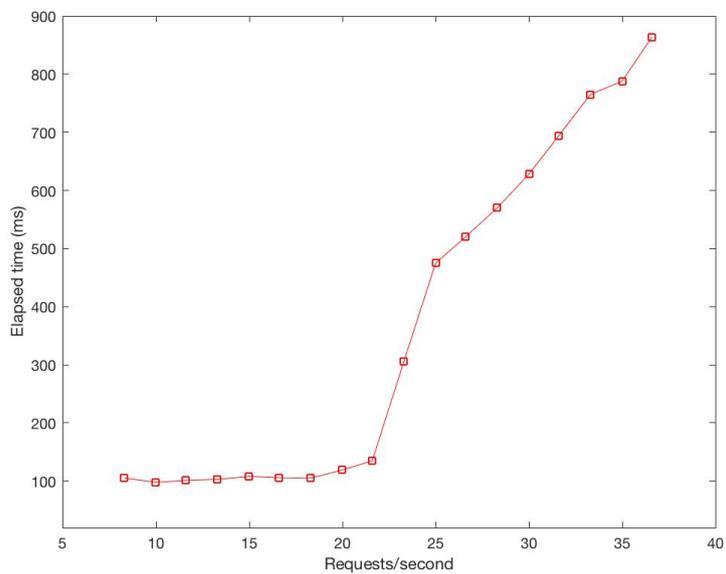


Figura 6.7: Elapsed time per il GIS con 50000 punti

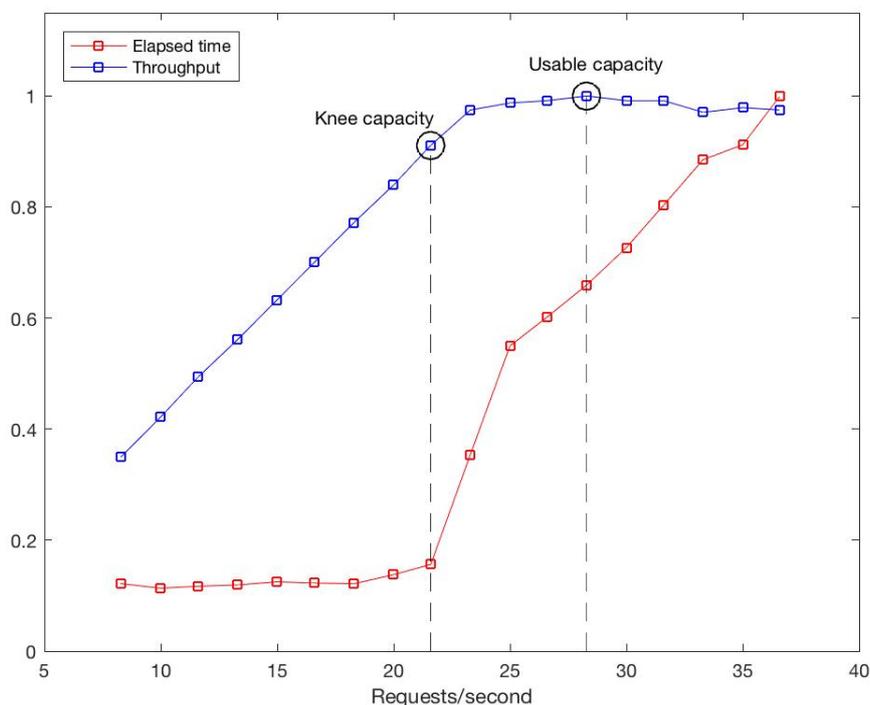


Figura 6.8: Usable e knee capacity per il GIS con 50000 punti

Lo stesso procedimento è stato ripetuto per gli altri valori, ottenendo:

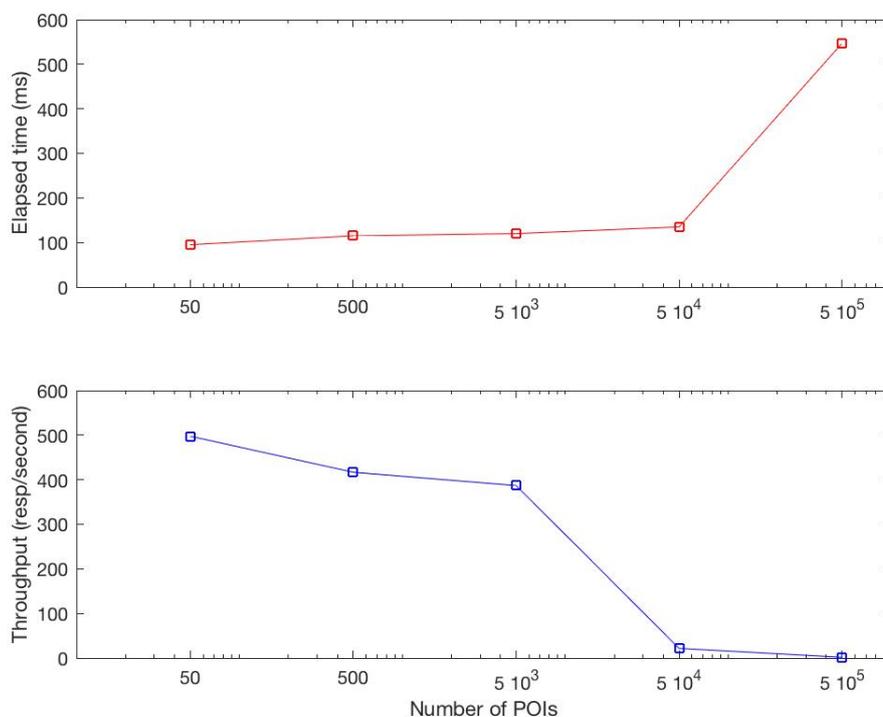
	Knee capacity (req/sec)	Throughput (resp/sec)	Response time (ms)
50 punti	500	497.5	95
500 punti	433.3	416.7	115
5000 punti	400	386.8	120
50000 punti	21.6	21.6	135
500000 punti	1.6	1.6	547

6.5 Risultati

Nell'ultima fase sono stati plottati i risultati riportati nella precedente tabella. Il grafico è stato disegnato utilizzando una scala logaritmica per l'asse delle ascisse, per comodità di visualizzazione.

Nel primo grafico è presente l'interpolazione dei valori di elapsed time associati alle knee capacities dei punti di ascissa considerati.

Nel primo grafico è presente l'interpolazione dei valori di throughput associati alle knee capacities dei punti di ascissa considerati.



E' possibile, a questo punto, riuscire a determinare, anche graficamente, quale ordine del carico del GIS risulta soddisfare i requisiti non funzionali.

Si consideri come primo vincolo quello relativo al tempo di risposta medio del sistema. Riportando da "Usability Engineering", J. Nielsen, tre importanti limiti determinati dalle abilità percettive umane:

- **0.1 second** is about the limit for having the user feel that the system is reacting instantaneously, meaning that no special feedback is necessary except to display the result.
- **1.0 second** is about the limit for the user's flow of thought to stay uninterrupted, even though the user will notice the delay. Normally, no special feedback is necessary during delays of more than 0.1 but less than 1.0 second, but the user does lose the feeling of operating directly on the data.
- **10 seconds** is about the limit for keeping the user's attention focused on the dialogue. For longer delays, users will want to perform other tasks while waiting for the computer to finish, so they should be given feedback indicating when the computer expects to be done. Feedback during the delay is especially important if the response time is likely to be highly variable, since users will then not know what to expect.

La scelta di un tempo di risposta ragionevole può essere quindi fatta a partire da tale suddivisione.

Guardando al solo grafico del response time, e considerando la suddivisione riportata, avendo anche nel caso peggiore un elapsed time inferiore ad 1 secondo, non ci sono casi che possano risultare, ad una prima analisi, proibitivi.

Si osserva tuttavia un aumento drastico (in percentuale) del response time passando da 50000 a 500000 punti, dovuto all'aumento nel tempo di esecuzione della query geografica. Volendo dunque fissare un limite al tempo di risposta indicativamente a 200 ms, risulta scartato il caso di 500000 punti ma per i restanti bisogna invece entrare nel dettaglio del throughput.

Il throughput misurato è in risposte per secondo, e conoscendo "l'attività" media del client del sistema, in termini di numero medio di sottomissioni al secondo, è possibile ricondurre i risultati ottenuti sul throughput ad un numero medio di utenti supportati dal singolo server quando questo lavora in condizioni di carico ottimale (si ricordi che si sta considerando la knee capacity per ciascuna dimensione del GIS).

Guardando l'andamento del throughput si può notare, come nel caso precedente, che vi è un punto di peggioramento drastico del valore osservato, che cade in corrispondenza, questa volta, dei 50000 punti, passando da 23000 a circa 1300 risposte al minuto.

Nonostante si voglia minimizzare il numero di server per ridurre i costi (il che porterebbe a scegliere proprio il valore di 50000), ciò tuttavia non è detto che coincida necessariamente con la massimizzazione del profitto (come definito di seguito), per cui è necessario considerare anche il numero degli utenti per singolo server (e quindi il throughput).

Supponendo di conoscere il guadagno medio per utente g_{utente} ed il costo per server c_{server} si definiscono:

$$costo = N_{server}c_{server}$$

$$guadagno = N_{utenti}g_{utente}$$

$$profitto = guadagno - costo = N_{server}c_{server} - N_{utenti}g_{utente}$$

che nel caso corrente diventa:

$$profitto_{5000} = 23000g_{utente} - 10c_{server}$$

$$profitto_{50000} = 1300g_{utente} - c_{server}$$

Dunque il massimo tra $profitto_{5000}$ e $profitto_{50000}$ determina la scelta tra il valore di 5000 o 50000 punti per GIS.

Risulta conveniente una dimensione di 5000 punti per GIS aumentando il numero di server quando

$$profitto_{5000} > profitto_{50000}$$

ossia quando

$$23000g_{utente} - 10c_{server} > 1300g_{utente} - c_{server}$$

$$21700g_{utente} > 9c_{server}$$

altrimenti conviene utilizzare server con 50000 punti per GIS.

Capitolo 7

Conclusioni



Capitolo 8

Sviluppi futuri



Bibliografia

- [1] Beyan, Oya *An Overview of Geotagging*, 2016
- [2] Pryss, Geiger, Schickler, Schobel, Reichert *Advanced Algorithms for Location-Based Smart Mobile Augmented Reality Applications*, 2016
- [3] Geiger, Schickler, Pryss, Schobel, Reichert *Location-based Mobile Augmented Reality Applications* 2014
- [4] Donoser, Schmalstieg *Discriminative Feature-to-Point Matching in Image-Based Localization*, 2014
- [5] Gelernter, Balaji *An algorithm for local geoparsing of microtext*, 2013
- [6] Lieberman, Samet *Adaptive Context Features for Toponym Resolution in Streaming News*, 2012
- [7] Sattler, Leibe, Kobbelt *Fast Image-Based Localization using Direct 2D-to-3D Matching*, 2011
- [8] Zamir, Shah *Accurate Image Localization Based on Google Maps Street View*, 2010
- [9] Crandall, Backstrom, Huttenlocher, Kleinberg *Mapping the Worlds Photos*, 2009
- [10] Amitay, Har'El, Sivan, Soffer *Web-a-Where: Geotagging Web Content*, 2004
- [11] Schiller, Voisard *Location-based services*, 2004
- [12] José, Moreira, Rodrigues, Davies *The AROUND Architecture for Dynamic Location-Based Services*, 2003.