

Relazione per l'elaborato finale Progettazione e Produzione Multimediale 2016/2017

Simone Magistri & Tommaso Scarlatti

28 Giugno 2017

1 Introduzione

L'elaborato che ci siamo proposti di realizzare risponde alla seguente consegna: creazione di una web app con Unity 3D per la visualizzazione interattiva di flussi sociali.

In primo luogo è stato necessario determinare quale social network utilizzare per il recupero dei dati. La scelta è ricaduta su Facebook, in particolare sui luoghi. E' stata presa la decisione di recuperare i luoghi vicini alla posizione dell'utente sfruttando le API messe a disposizione da Facebook.

In un secondo momento si è dovuto elaborare una strategia di rappresentazione dei luoghi: si è scelto di mostrarli come oggetti tridimensionali su un piano, conservando le distanze originali tra di essi e dando la possibilità all'utente di spostarsi sul piano per raggiungerli.

Infine si è passati a definire il modo in cui l'utente può interagire con i luoghi tridimensionali: avvicinandosi ad essi può recuperare alcune informazioni basilari in maniera immediata: nome, immagine del luogo, posizione... e può decidere di aprire la pagina Facebook corrispondente o di tracciare un percorso tra la posizione di partenza e quella del luogo su Google Maps.

2 Analisi e Progettazione

2.1 Studio di fattibilità

Una volta ricevuta la consegna del progetto, si è reso necessario uno studio di fattibilità per ogni strategia di sviluppo elaborata, in termini di tempo e di nuove conoscenze da integrare a quelle acquisite durante il corso.

Per formalizzare il tutto si è fatto ricorso a software di prototyping (Balsamiq Mockups 3) per la realizzazione di mockups che permettessero di catturare in modo semplice ed immediato le finalità primarie della strategia proposta.

Sono state elaborate tre differenti alternative per catturare ed organizzare determinati flussi sociali in un ambiente tridimensionale:

1. Mappa 2d (o 3d) sottostante di un determinato luogo a scelta con luoghi di interesse geolocalizzati rappresentati tridimensionalmente e organizzati in base alla propria posizione. Se un oggetto viene "selezionato" mostra persone registrate, foto condivise e recensioni.
2. Si recupera la posizione attuale dell'utente e si mostrano, in un range determinato, i luoghi principali organizzati per categorie. I luoghi sono rappresentati

da sfere, di ampiezza variabile in base alle percentuali di recensioni positive su Facebook. Rispetto al primo progetto i luoghi sono disposti nello spazio tridimensionale senza tener conto della loro distanza effettiva.

3. Si effettua il login con il proprio account Facebook. Viene mostrato in uno spazio tridimensionale il grafo delle amicizie: al centro vi è il nodo che rappresenta il proprio profilo e si sviluppano intorno i nodi degli amici. La distanza dei nodi degli amici dipende dal "grado" dell'amicizia, quantificato in termini di post condivisi, tag e foto insieme..

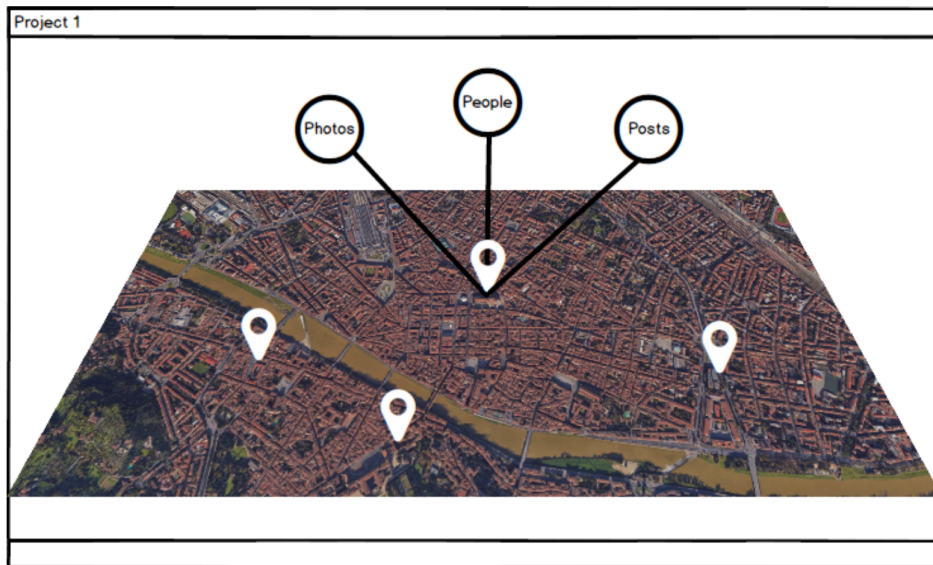


Figure 1: Mockup del primo progetto: Firenze scelta come location.

La scelta finale è ricaduta su una fusione delle due prime strategie: viene utilizzata la mappa reale come texture e i luoghi sono organizzati per categorie, (utilizzando dei colori come marcatori) e mantenendo le distanze originali.

E' stata necessaria quindi un'opera di approfondimento su Unity e sui due linguaggi utilizzati all'interno del software: *C#* e *Javascript*. Inoltre è stata indispensabile un'adeguata documentazione sulle API di Facebook e di Google e sui formati di interscambio dati nelle richieste *HTTP: JSON* e *XML*.

2.2 Analisi dei requisiti

L'applicazione che andremo a creare sarà un'applicazione web, quindi utilizzabile tramite browser. Proprio quest'ultimo può essere utilizzato per il recupero della posizione corrente, tramite l'utilizzo della *Geolocation API* messa a disposizione da *HTML5*. Una soluzione alternativa potrebbe essere quella di includere in una richiesta HTTP il proprio indirizzo *IP* per cercare di approssimarne la latitudine e la longitudine corrispondenti. Il problema fondamentale è che non stiamo lavorando su applicazioni native di sistemi mobile, quindi dobbiamo appoggiarci ad una delle due soluzioni per poter recuperare le coordinate GPS della propria posizione.

Una volta recuperata la posizione abbiamo bisogno della texture della mappa in un determinato intorno del punto per poterla renderizzare su Unity. Per farlo si

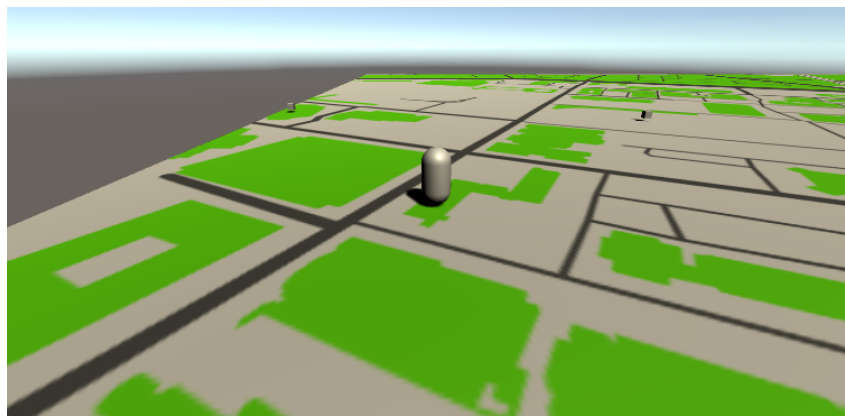


Figure 2: Screen che mostra il funzionamento dell'asset ai confini di un quad

è utilizzato l'asset *GoogleGoMaps* opportunamente modificato per rispondere alle nostre esigenze.

Il funzionamento dell'asset è il seguente: si utilizzano due quad alternativamente visibili per renderizzare la mappa nell'intorno del giocatore. Se il giocatore valica i confini di un quadrato prontamente verrà fatto diventare visibile l'altro, cercando di simulare una continuità nel terreno sottostante. Questa soluzione è stata l'unica possibile: Google infatti non permette di accedere liberamente ad alcune API, quindi si è dovuto scegliere un API che fosse disponibile generando una chiave ad hoc sul momento: Google Maps Static API permette il recupero di porzioni statiche della Google Maps, e saranno proprio queste che utilizzeremo per il nostro rendering.

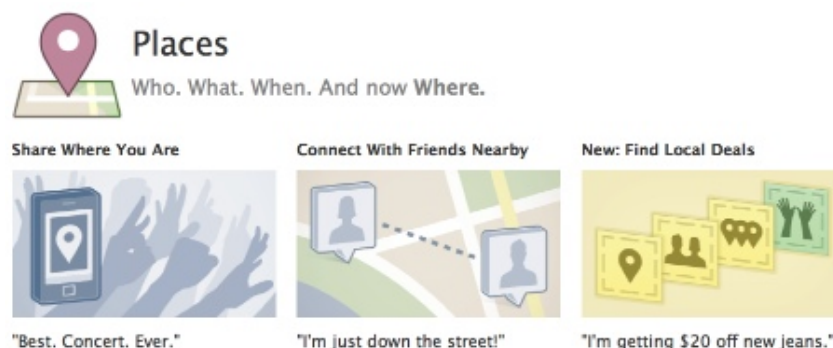


Figure 3: Luoghi Facebook

I campi forniti da Facebook sui luoghi sono tra i più disparati, una lista completa è disponibile sulla documentazione associata. Nella fase di analisi è stato necessario stabilire quali dati recuperare e come utilizzarli. Si è scelto di recuperare elementi caratterizzanti dei luoghi quale il nome, l'immagine, la posizione (utile per disporre il luogo sulla mappa), il numero di registrazioni (checkins) e la media delle recensioni degli utenti. Questi ultimi due campi determinano, in un certo senso, l'"importanza" del luogo. Possiamo utilizzare questo fatto per quantificare la grandezza dei luoghi nello spazio tridimensionale.

2.3 Graphic User Interface

Una volta conclusa la fase di analisi del recupero dati si è passati all'analisi delle possibili interazioni dell'utente con l'applicazione. Si è optato per permettere all'utente di muoversi all'interno dello spazio tridimensionale attraverso l'utilizzo di un third

person controller, ossia un personaggio controllabile in terza persona. In prossimità dei luoghi l'utente potrà poi recuperare le informazioni relative al luogo in due fasi: una prima fase, in cui cliccando sul posto si apre un popup con alcune info, e una seconda fase, che permette dal popup di spostarsi o sulla pagina Facebook corrispondente per recuperare informazioni più dettagliate, oppure su Google Maps per determinare le indicazioni fino a quel posto.

I luoghi vengono suddivisi in 8 categorie di appartenenza. Ogni luogo può appartenere ad una sola categoria. L'utente può decidere quali tipi di luoghi mostrare e quali no mediante un menù formato da toggle, uno per ogni categoria. Questo permette una visualizzazione più chiara nel caso di luoghi particolarmente densi di posti Facebook.



Figure 4: Menù dei toggle della GUI

In ultima battuta si è sentita la necessità di determinare, ad ogni istante, quale fosse la direzione contestuale dell'avatar all'interno dello spazio di Unity. Per farlo, si è scelto di utilizzare una bussola orientata in base alla direzione del player.

3 Sviluppo

3.1 Recupero ed organizzazione dei dati

Per recuperare i dati di interesse in un determinato intorno del player si è fatto ricorso ad una richiesta HTTP al grafo dei luoghi Facebook, in questo modo:

GET [https://graph.facebook.com/v2.9/search?type=place ...](https://graph.facebook.com/v2.9/search?type=place...)

seguita da un numero variabile di campi. Questo ha permesso di ottenere un file di testo in formato JSON con la lista di tutti i posti all'interno di un raggio specificato. Sono state fatte 8 diverse richieste per recuperare posti per ogni categoria specificata. Successivamente si è condotta un'operazione di parsing per organizzare i dati in oggetti C#, creando classi di oggetti con parametri speculari rispetto a quelli nel JSON. In questo modo tutte le informazioni presenti nel JSON codificate come testo ASCII possono essere utilizzate all'interno del codice come variabili!

Per recuperare le informazioni, sia nel caso delle API Facebook che di quelle Google, è stato necessario creare delle chiavi di autenticazione per la nostra applicazione. In questo modo solo chi ha l'autorizzazione può avere accesso a determinate informazioni. Pensiamo ad esempio ad un profilo Facebook con un'impostazione di privacy tale che solo gli amici possono accedere alle informazioni. Tramite API vale lo stesso concetto: ho bisogno di un token di autenticazione per accedere alle info. I

luoghi Facebook, tuttavia, necessitano solo di un token legato ad un'applicazione e non ad un utente. Questi ultimi infatti sono consultabili anche senza essere iscritti alla piattaforma social.

```
{
  "data": [
    {
      "name": "Cafe Orlin",
      "checkins": 24445,
      "picture": {
        "data": {
          "is_silhouette": false,
          "url": "https://scontent.xx.fbcdn.net/v/t1.0-1/c15.0.100.100/p100x100/943881_1097275670291612_7819187236926166576_n.jpg?oh=8e9aed6bbd9763c80399240a7fe91968&oe=59DDA61D"
        }
      },
      "description": "\"Something of a neighborhood institution, this East Villager's satisfying American basics (with some Middle Eastern accents) are dished up in mellow confines; it's a brunch hot spot, so get there early to avoid the line - and even earlier to snag a coveted outdoor seat\" - ZAGAT\\n",
      "location": {
        "city": "New York",
        "country": "United States",
        "latitude": 40.7284999,
        "longitude": -73.98707,
        "state": "NY",
        "street": "41 St Marks Pl",
        "zip": "10003"
      },
      "id": "147652718587250"
    }
  ]
}
```

Figure 5: Una parte del file JSON recuperato tramite l'API Facebook

Per recuperare le immagini statiche della mappa di Google è stata necessaria una chiamata alle API affine, in questo modo:

GET <https://maps.googleapis.com/maps/api/staticmap?...>

seguito da una serie di parametri tra cui figurano: il centro del raggio, la risoluzione della mappa, la scala, il tipo di mappa... Per quanto riguarda quest'ultimo parametro si è optato per un tipo di mappa ibrido, che combina sia la visuale satellitare che quella stradale.

3.2 Disposizione dei dati in Unity

Una volta ottenuti tutti i dati necessari al funzionamento dell'applicazione occorre utilizzarli per disporre gli oggetti nello spazio tridimensionale. Le coordinate che si ottengono sono coordinate reali, come fare a mapparle all'interno dello spazio di Unity? Per ovviare a questo problema si è utilizzata una variazione della proiezione di Mercatore chiamata Web Mercator. Questo è ormai divenuto uno standard di Internet per le applicazioni web mapping (utilizzato per la prima volta da Google nel 2005).

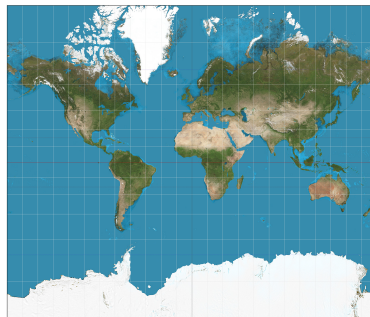


Figure 6: Proiezione di Mercatore

Inizialmente per rappresentare i luoghi si era scelto di utilizzare dei cubi con una texture renderizzata con l'immagine del luogo. Questi oggetti erano in perenne rotazione sul proprio asse e alzati dal livello del terreno in modo che fossero facilmente visibili. Tuttavia questa soluzione non riusciva ad identificare i luoghi Facebook al meglio, per questo si è utilizzato una soluzione più raffinata.

Utilizzando il software di modellazione 3D Sketchup si è creato due modelli tridimensionali. Uno ha la classica forma del segnaposto per un luogo sulle mappe ed andrà a simulare la presenza di un Facebook Place (anch'esso sarà in continua rotazione per essere visibile da tutte le direzioni). L'altro ha la forma di una freccia che punta verso il terreno. Questo marker, dotato di un moto sia oscillatorio che rotatorio, permette di conservare la posizione di partenza dell'utente e la rende visibile anche da grandi distanze grazie anche alla sua posizione elevata.

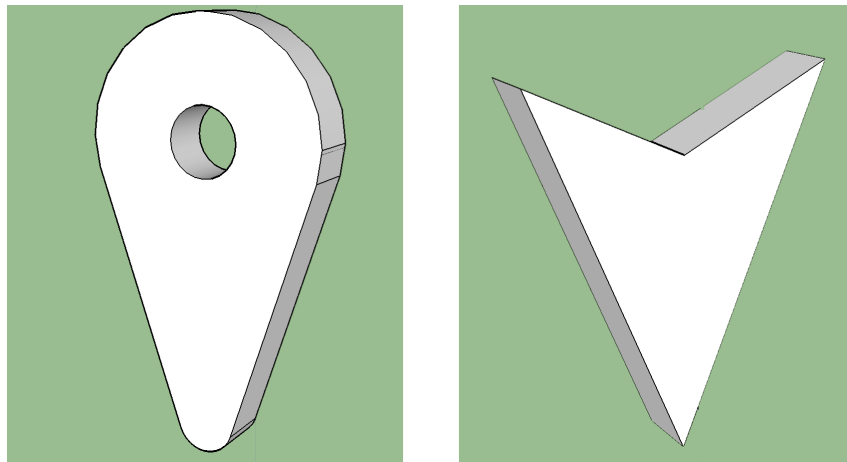


Figure 7: Modelli Sketchup dei luoghi e del marker per la posizione iniziale

Al marker è stato attribuito un caratteristico colore verde chiaro, mentre per i luoghi si è scelto di utilizzare 8 tra colori primari e secondari: questo li rende facilmente distinguibili tra di loro anche a grandi distanze. Per rendere la navigazione e la visualizzazione dei luoghi e della mappa più gradevole è stato poi escogitato un sistema di luci direzionali che permettesse di avere una buona illuminazione in ogni posizione della mappa e in qualsiasi direzione assunta dal giocatore.

Come è emerso nella fase di analisi, vogliamo mappare l'importanza di un posto in termini di registrazioni e valutazioni degli utenti nel nostro ambiente tridimensionale. Per farlo si è scelto di ingrandire i luoghi a seconda del proprio *scaleFactor*, un valore floating point così calcolato:

$$scaleFactor = \alpha * c + (1 - \alpha) * r$$

Questa altro non è che una media pesata tra il numero di checkins e la media delle valutazioni degli utenti da 1 fino a 5 stelle. Il fattore di peso è pari a 0.7, in quanto si è scelto di privilegiare il numero di registrazioni come indice preponderante per determinare la popolarità di un luogo. Sorge però un problema: il numero dei checkins è illimitato superiormente. Per risolverlo si utilizza un sistema di intervalli che permette di discretizzare il valore nc in un valore c utilizzabile nella formula.

In questo modo lo *scaleFactor* potrà raggiungere un valore compreso tra 0 e 5 estremi inclusi ed andrà quindi a determinare la dimensione finale del luogo tridimensionale.

Number of checkins nc	c
$nc < 10$	0
$10 \leq nc \leq 100$	1
$100 \leq nc \leq 1.000$	2
$1000 \leq nc \leq 10.000$	3
$10000 \leq nc \leq 100.000$	4
$nc > 100.000$	5

Table 1: Tabella dei valori del parametro c

La scelta del fattore di peso è arbitraria ma permette di limitare la dimensione di luoghi con poche recensioni ma estremamente positive.

3.3 Implementazione della GUI

La fase di analisi dell'interfaccia grafica destinata all'interazione dell'utente ha fatto emergere la necessità di mantenere sempre visibile la posizione attuale del giocatore e la sua direzione contestuale. Questo è stato raggiunto attraverso l'utilizzo di un asset dotato di una bussola configurabile e attraverso l'introduzione di una minimap.

Quest'ultima permette di avere una visuale ad occhio di falco costantemente centrata sul giocatore (ivi rappresentato da un cerchio blu) e, in azione combinata con la bussola, garantisce un orientamento completo in ogni zona della mappa che si decida di esplorare.

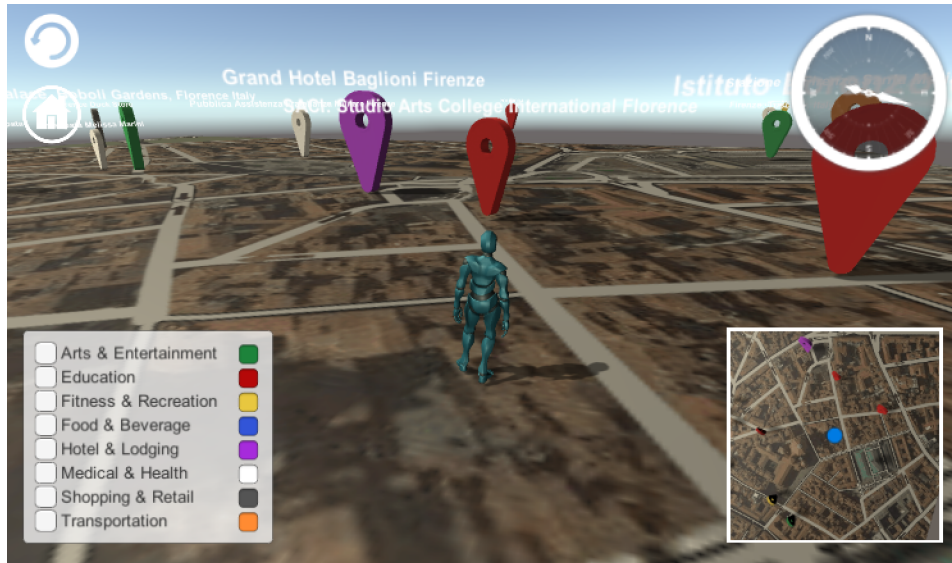


Figure 8: Visuale dell'applicazione con posizione iniziale Firenze

All'apertura dell'applicazione il pannello in basso a sinistra ha tutti i toggle disabilitati. Attesi 5 secondi di caricamento, durante i quali non è possibile né cliccare sui toggle né muovere il giocatore, è possibile iniziare ad usare l'applicazione utilizzando tutte le sue funzioni. Abilitare un toggle comporterà la visualizzazione dei luoghi: questi saranno semplicemente settati visibili e non si effettua una nuova chiamata all'API Facebook. Questo permette di ridurre notevolmente la latenza di caricamento, spostando la gestione dell'aggiornamento dei luoghi caricati lato utente.

Inizialmente si era pensato ad un aggiornamento dei luoghi automatico, utilizzando un timer ciclico. Successivamente si è preferito implementare un pulsante di refresh in alto a sinistra per dare la possibilità all'utente di decidere quando ricaricare

i posti, a seconda delle esigenze del momento. Durante l'aggiornamento per ragioni di sicurezza tutte le interazioni con la GUI sono disabilitate.

Il pulsante home, posizionato sotto quello di refresh, permette infine di spostarsi direttamente alla posizione di partenza, in qualsiasi luogo ci si trovi. Questo è molto utile nel caso che, una volta esplorata una zona in una determinata direzione, si decida di esplorarne un'altra in un'altra direzione.

4 Conclusioni

L'applicazione web che è stata sviluppata ha permesso di inquadrare in un contesto tridimensionale un insieme di flussi sociali provenienti dai luoghi Facebook. Essa si rivela utile nel caso si voglia fare una "passeggiata 3D" per scoprire i luoghi di interesse in un determinato intorno, organizzati per categorie e per rilevanza.

L'utilizzo delle API messe a disposizione sia da Facebook che da Google si è dimostrato cruciale nel tentativo di recupero dei dati a noi interessati. Si tratta di oggetti potentissimi che mascherando la complessità dell'implementazione rendono elementari le operazioni più complesse.

Lo sviluppo dell'applicazione è stato condotto in tre settimane di lavoro intenso che hanno portato a percorrere, in sequenza, tutte le sezioni descritte in questa relazione. La cooperazione ha giocato un ruolo fondamentale: gestire parti separate dell'applicazione ha permesso di approfondire singolarmente dei casi di studio per poi integrarli a vicenda dopo averli fatti propri in maniera consapevole.

L'applicazione si presenta sicuramente bene ad integrazioni future, offrendo un'ampia possibilità di scelte. Sarebbe utile, ad esempio, fornire un ambiente tridimensionale anche per il terreno sottostante, utilizzando un parametro come l'altitudine, o ancora: dare la possibilità all'utente di loggarsi con il proprio account Facebook per poter recuperare altre informazioni nei luoghi quali gli amici che si sono registrati.