GeoNews:
Trova e leggi sempre le news
da tutto il mondo

Giorgio Mazza

20 luglio 2018

Indice

1 Introduzione

GeoNews è un'applicazione Android in grado di ricercare notizie e articoli da oltre 30.000 fonti, testate giornalistiche e blog provenienti da tutto il mondo. La peculiarità di GeoNews è quella di mostrare le notizie in tempo reale, posizionate contestualmente su di una mappa del Globo in base alla località di provenienza dell'articolo.
L'utente potrà così rimanere aggiornato su ciò che accade nel mondo, in modo non più dispersivo ma bensì contestualizzato e mirato. Infatti sarà l'utente a decidere di quale parte del mondo leggere le ultime notizie.
GeoNews infine, si propone anche come community in cui gli utenti potranno esprimere il proprio parere su ciò che gli accade intorno, mediante il sistema di commenti integrato in ogni articolo.

2 Architettura

L'architettura di GeoNews è rappresentata in figura:

Figura 1: Architettura di GeoNews

Il dispositivo Android funge da Client a due diversi server: uno hostato su http://newsapi.org mediante il quale reperisce gli ultimi articoli ed uno sviluppato appositamente per GeoNews e hostato su http://geonews.altervista.org volto alla gestione della sezione commenti presente in ogni articolo.

3 Aspetti di Android

L'applicazione è strutturata in tre package principali:

- **Activities**: in cui sono presenti tutte le activities che compongono l'applicazione
- **Data_utils**: all'interno del quale ci sono funzioni di utilità riguardanti la manipolazione dei dati
- **Networking**: in cui ci sono tutte quelle classi che svolgono funzioni di networking

3.1 Activities

Le principali activities sono:

- **SplashActivity**: una semplice activity che funge da splash attraverso la quale si potrà godere a pieno del logo di GeoNews.
- **MapsActivity**: activity lanciata dopo SplashActivity, contiene un `Fragment` che visualizza una `Mappa Google` reperita mediante le omonime API.
  La mappa posiziona contestualmente dei markers in alcune nazioni, che indicheranno la presenza di articoli disponibili.
  Selezionato un marker il metodo `onMarkerClick` invierà un Intent all'activity `ListArticlesActivity`, la quale sarà lanciata.
- **ListArticlesActivity**: Activity che visualizza in una `RevyclerView` una lista delle news correlate alla nazione selezionata;
  Questo è reso possibile grazie a:
  - **fetchArticles** del package Networking, il quale reperisce le suddette news da Internet (se ne parlerà successivamente)
  - **RV_adapter** che gestisce la costruzione del layout e l'inserimento degli articoli in esso.
    Da qui se l'utente fa click su uno specifico articolo, il metodo `setOnClickListener` della classe `CustomViewHolder` invierà un Intent con specificato l'`url` dell'articolo e sarà lanciata l'activity `ArticleDetailActivity`

- **ArticleDetailActivity**: Activity che visualizza, mediante una `WebView`, l'articolo selezionato collegandosi all'url passatogli dalla precedente Activity.
  L'activity comprende in sè un `Fragment` che può essere nascosto o visualizzato grazie ad un `Button` apposito.

  - **ArticleCommentFragment**: Fragment che visualizza la lista dei commenti creati dagli utenti in quella specifica news.
    Anch'esso usa due estensioni:

    - **fetchComments** del package Networking, il quale reperisce i commenti dal database hostato us geonews.altervista.org
    - **RV_adapter** che gestisce la costruzione del layout e l'inserimento dei commenti in esso.
      Da qui, mediante la funzione `setOnLongClickListener` nella classe `CustomViewHolder`, se l'utente tiene premuto su un commento viene istanziato un `PopupMenu` che visualizzerà all'utente due scelte: *Rimuovi commento* o *Aggiorna commento*.

3.2 DataUtils

Qui troviamo funzioni di modellazione dei dati come:

- **Models.kt**: Contiene classi personalizzate che rappresentano entità dell'applicazione come *News, Article, Comment...* usate per il parsing da json ad oggetti.

- **Constant**: Contiene tutte le costanti come gli url per comunicare con il server geonews.altervista.org

3.3 Networking

Classi *singletons* usate per interrogare il web service *newsapi.org* e, tramite parser, ottenere dalla risposta in json oggetti usabili nell'applicazione. Abbiamo:

- **FetchArticles** che, con l'aiuto di `okHttp` e `gson` ritrova gli articoli da newsapi.org e li passa all'`Adapter` di `ListArticlesActivity`.
- **FetchComments** che, con l'aiuto di `okHttp` e `gson` ritrova i commenti da geonews.altervista.org e li passa all'`Adapter` di `ArticleCommentFragment`.
- **CommentsUtils** che implementano le operazioni CRUD sui commenti.

Si rimanda alla lettura della sezione successiva per l'approfondimemnto del package Networking.

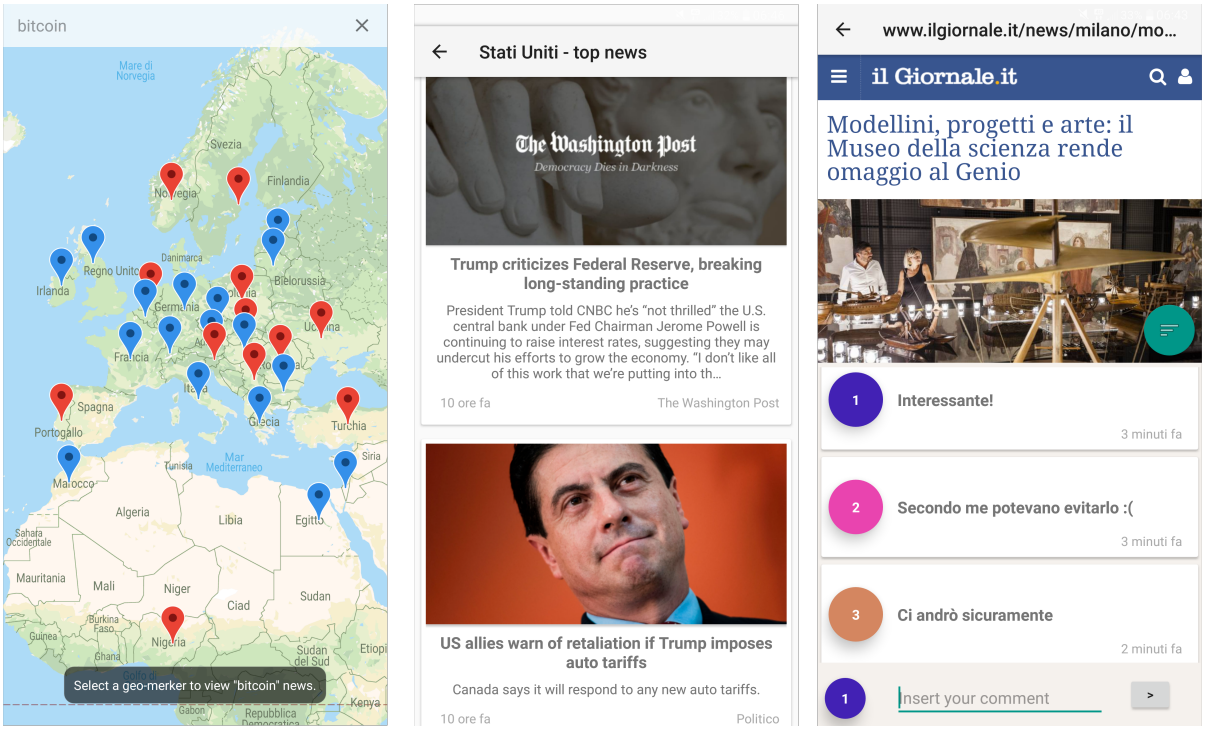

Figura 2: MapActivity, ListArticlesActivity e ArticleDetailActivity

4 Aspetti di Networking

4.1 Il client: l'app Android

Gli aspetti client dell'applicazione riguardano la comunicazione con i due server
Per reperire gli articoli, GeoNews fa riferimento a `newsAPI`, un sito che mette a disposizione delle API REST che ritornato metadati in formato JSON da oltre 30.000 sorgenti, testate giornalistiche e blogs.
Per questo motivo si è deciso di basare le comunicazioni tra il client ed i due server sullo scambio di JSON; Per fare il parsing di quest'ultimo, il client si serve della libreria open source **Gson** di Google.
Per quanto rigurada il networking, si sono utilizzate due librerie in particolare:

- **OkHttp**: client http efficiente e di facile utilizzo
- **Picasso**: usato per il caricamento efficace delle immagini

4.1.1 FetchArticles.kt - Reperire gli articoli

Per quanto riguarda le operazioni di reperimento degli articoli, queste sono svolte all'interno del file `FetchArticles.kt` che effettuerà una chiamata **GET** in forma:
`http://newsapi.org/v2/end-point ? user-queries & apiKey=priv-key`
dove:

- **end-point** rappresenta uno degli `end-points` messi a disposizione da newsapi.org:
  - per la ricerca delle **top news** si è usato l'end-pont *top-headlines* che fornisce le top news sottoforma di titolo, descrizione e link all'articolo.
  - per la ricerca delle news **per argomento** si è usato l'end-point *everything* che permette di visualizzare news anche meno recenti e consente più parametri di ricerca.
- **user-queries** sono le queries che l'utente può fare alle API a fini di ricerca mirata di news. In questo caso, è stata utilizzata la query `country="zona della mappa selezionata"` per visualizzare le news di quella specifica zona.

Dopo di che, la stringa **JSON** sarà passata alla funzione gson di Gson, che effettuerà il parsing in oggetti utilizzabili in Kotlin, usando come modelli la classe `Models.kt`.
Una volta selezionato un articolo, sarà visualizzato sottoforma di *WebView*.

4.1.2 FetchComments.kt - Reperire i commenti

Per quanto riguarda la sezione commenti, lì'app mette a disposizione un `Fragment` che l'utente può nascondere o visualizzare durante la lettura dell'articolo; Questo contiente all'interno l'elenco dei commenti già scritti da altri utenti in quel determinato articolo, ed un `EditText` attraverso il quale l'utente può postare sul server di geonews.altervista.org i propri commenti.

CommentsUtils : Nel fragment dei commenti (`ArticleCommentFragment`), a seconda dell'accadere dei relativi eventi scatenati dall'utente, vengono chiamate le funzioni presenti all'interno del file `CommentsUtils`, che implementa le principali operazioni CRUD sul database dei commenti nel server di `geonews.altervista.org`, mediante l'uso di `okHttp`:

- **CREATE**: `createComment()` invia una richiesta **POST** all'url `http://geonews.altervista.org/addComment.php` attraverso la pressione del bottone per inviare un commento
- **READ**: `fetchComments()` invia una richiesta **GET** all'url `http://geonews.altervista.org/getAllComments.php` attraverso la creazione del fragment ma anche dopo aver effettuato uno *swipe to refresh*
- **UPDATE**: `updateComment()` invia una richiesta **POST** all'url `http://geonews.altervista.org/updateComment.php` dopo che l'utente ha scelto *aggiorna commento* dal `PopupMenu` comparso alla pressione prolungata su di un suo commento
- **DELETE**: `deleteComment()` invia una richiesta **POST** all'url `http://geonews.altervista.org/deleteComment.php` dopo che l'utente ha scelto *cancella commento* dal `PopupMenu` comparso alla pressione prolungata su di un suo commento

4.2 Il server: sezione per i commenti

Lato server sono stati realizzati un database MySQL per la memorizzazione dei commenti, e alcuni script PHP per la loro fruizione.

4.2.1 Tabella COMMENTS MySQL

Lo schema della tabella **COMMENTS** è il seguente:

	#	Nome	Tipo	Codifica caratteri	Attributi	Null	Predefinito	Extra	Azione
☐	1	**id**	int(11)			No	*Nessuno*	AUTO_INCREMENT	Modifica Elimina Primaria
☐	2	**comment**	varchar(30000)	latin1_swedish_ci		No	*Nessuno*		Modifica Elimina Primaria
☐	3	**url**	varchar(10000)	latin1_swedish_ci		No	*Nessuno*		Modifica Elimina Primaria
☐	4	**android_id**	varchar(20000)	latin1_swedish_ci		No	*Nessuno*		Modifica Elimina Primaria
☐	5	**usr**	varchar(11)	latin1_swedish_ci		No	*Nessuno*		Modifica Elimina Primaria
☐	6	**date**	datetime			No	CURRENT_TIMESTAMP		Modifica Elimina Primaria

Figura 3: Struttura della tabella COMMENTS

In cui:

- **id**: è l'identificativo unico di ogni commento, *chiave primaria* della tabella ed *autoincrementato*
- **comment**: è il commento scritto dall'utente vero e proprio
- **url**: è l'indirizzo internet dell'articolo su cui è stato scritto quel determinato commento
- **android_id**: è l'identificativo unico del dispositivo dal quale il commento è stato scritto
- **usr**: è l'id a cui sono associati tutti i commenti di un utente nella sezione dei commenti relativa ad un determinato articolo (es. al primo utente che commenta gli viene assegnato usr=1, al secondo utente *distinto* usr=2 etc.)
- **date**: è la data di pubblicazione del commento

4.2.2 Script PHP

Per la gestione dei commenti sono stati realizzati 5 script PHP, di cui i primi quattro sono quelli fondamentali ed implementano le operazioni CRUD sui commenti:

- **addComment.php** per le operazioni di `INSERT` di un commento mediante una richiesta `GET`
- **getAllComment.php** per le operazioni di `SELECT` su tutti i commenti di un determinato articolo, questo grazie alla clausola `WHERE url=$url_articolo`
- **updateComment.php** per le operazioi di `UPDATE` di un commento. Per semplicità, la verifica di appartenenza del commento selezionato all'utente che lo vuole agggiornare, viene eseguita direttamente da codice in Kotlin. Se questa condizione è soddisfatta, con la clausola `WHERE id=$id_commento` si prende il modifica il commento selezionato. Altrimenti l'utente sarà avvisato del fatto che non può modificare commenti altrui.
- **deleteComment.php** per le operazioi di `DELETE` di un commento. La verifica di appartenenza del commento selezionato all'utente avviene in modo analogo all'update

E gli ultimi due sono solo di utilità all'applicazione android, in particolare per la funzione `getUsrID` di `CommentsUtils.kt`:

- **getUsr.php** nel caso in cui l'utente abbia già scritto un commento in quell'articolo (si verifica mediante la clausola `WHERE url=$url AND android_id=$android_id`) e ritorna il suo `user id`.
- **getMaxUsr.php** nel caso in cui l'utente non ha ancora scritto alcun commento in quell'articolo. In questo caso gli verrà assegnato un nuovo `user id` selezionando l'ultimo user id assegnato (mediante la clausola `SELECT MAX(usr)` ed incrementandolo di uno.

E' opportuno notare che negli script che implementano le operazioni CRUD si è deciso realizzarli mediante l'uso dei **prepared statement** per evitare *SQL injection* e per separare la logica dei dati da quella di SQL, considerando anche il fatto che per scrivere un commento non c'è alcun sistema di login ma sarà fatto il tutto in forma anonima.

5 Further Works

Sviluppi futuri sull'applicazione GeoNews riguardano la traduzione in tempo reale non solo di titoli e descrizioni, ma anche degli articoli stessi, mediante una previa scelta delle lingue preferite dall'utente in fase di onboarding (ma comunque modificabili attraverso le impostazioni dell'applicazione). Questo porterebbe la fruibilità di GeoNews ad un livello superiore in quanto permetterebbe la fruizione di news estere da parte di chiunque.