



GeoNews

Università degli Studi di Perugia

a cura di

Giorgio Mazza

Luglio 2018



❖ Introduzione

Caratteristiche generali

❖ Client

Aspetti relativi all'applicazione come client

❖ Server

Aspetti riguardanti il server personale dei commenti

Introduzione



- ❖ Applicazione Android in grado di ricercare notizie e articoli da oltre 30.000 fonti
- ❖ Notizie visualizzate su una mappa Google (API) posizionate contestualmente



- ❖ Linguaggio client: **Kotlin**
- ❖ Linguaggio scripting server: **PHP**
- ❖ Linguaggio Database online: **MySQL**
- ❖ Uso di **API REST** di newsapi.org
- ❖ Scambio oggetti con **JSON**



✦ **OkHTTP**

client HTTP semplice ed efficiente

✦ **Picasso**

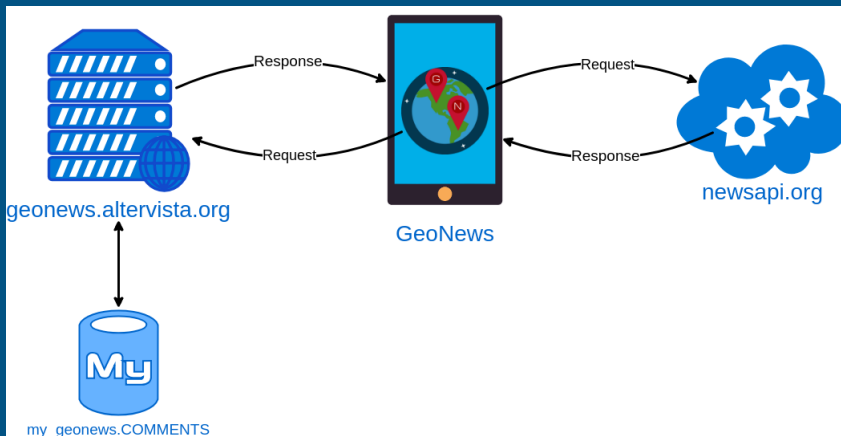
per il caricamento e caching delle immagini

✦ **Gson**

per il parsing JSON



❖ Architettura dell'applicazione



Client



- ❖ Operazioni di reperimento degli articoli svolte dal file `FetchArticles.kt`, che effettua una chiamata **GET** in forma:
`http://newsapi.org/v2/end-point ? user-queries & apiKey=priv-key`
- ❖ Stringa **JSON** passata alla funzione `gson` di `Gson`, che effettuerà il parsing

Fetch Comments



Nel fragment dei commenti (`ArticleCommentFragment`), sono chiamate le funzioni CRUD sul database nel server di `geonews.altervista.org`, mediante l'uso di `okHttp`:

❖ **CREATE:**

`createComment()` invia una richiesta **POST**
`codehttp://geonews.altervista.org/addComment.php`
attraverso la pressione di un bottone "invia"

❖ **READ:** `fetchComments()` invia una richiesta **GET** `http://geonews.altervista.org/getAllComments.php` effettuando uno *swipe to refresh*

❖ **UPDATE:** `updateComment()` invia una richiesta **POST** `http://geonews.altervista.org/updateComment.php` dopo che l'utente ha scelto *aggiorna commento*

❖ **DELETE:** `deleteComment()` invia una richiesta **POST** `http://geonews.altervista.org/deleteComment.php` dopo che l'utente ha scelto *cancella commento*

Server



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

#	Nome	Tipo	Codifica caratteri	Attributi	Null	Predefinito	Extra	Azi
<input type="checkbox"/> 1	id	int(11)			No	Nessuno	AUTO_INCREMENT	
<input type="checkbox"/> 2	comment	varchar(30000)	latin1_swedish_ci		No	Nessuno		
<input type="checkbox"/> 3	url	varchar(10000)	latin1_swedish_ci		No	Nessuno		
<input type="checkbox"/> 4	android_id	varchar(20000)	latin1_swedish_ci		No	Nessuno		
<input type="checkbox"/> 5	usr	varchar(11)	latin1_swedish_ci		No	Nessuno		
<input type="checkbox"/> 6	date	datetime			No	CURRENT_TIMESTAMP		



- ❖ **addComment.php** 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=$url_articolo`
- ❖ **updateComment.php** per le operazioni di UPDATE di un commento. Con la clausola
`WHERE id=$id_commento` si prende il commento da modificare.
- ❖ **deleteComment.php** per le operazioni di DELETE di un commento.

Script PHP - prepared statement



✚ addComment.php

```
$sql = "INSERT INTO `COMMENTS` (`comment`, `url`, `android_id`, `usr`) VALUES (?,?,?,?)";

//Prepare the template query
if($pstmt= $con->prepare($sql)){
    //Get values
    $comment = $_POST['comment'];
    $url = $_POST['url'];
    $android_id= $_POST['android_id'];
    $usr = $_POST['usr'];

    //Bind type values (ssss) whit bind values
    $pstmt->bind_param('ssss', $comment, $url, $android_id, $usr);
    //Execute the query
    $pstmt->execute();
}
```