DOCUMENTAZIONE *GDM HUB*

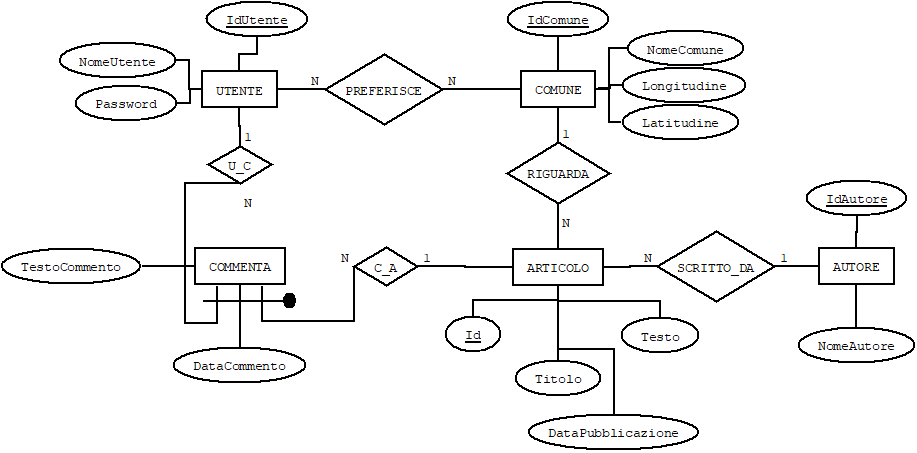
**INDICE**

1. [Obiettivo](#obiettivo)
2. [Progettazione concettuale DB](#prog_conc)
3. [Progettazione logica DB](#prog_logica)
4. [progettazione fisica DB](#prog_fisica)
5. [Interrogazioni SQL](#queries)
   1. [*home.php*](#homephp)
   2. [*geed.php*](#feedphp)
   3. [*getLatestNews.php*](#getlatestnewsphp)
   4. [*crawler.php*](#crawlerphp)
   5. [*articleShow.php*](#articleshowphp)
   6. [*login.php*](#loginphp)
   7. [*register.php*](#registerphp)
   8. [*accountShow.php*](#accountshowphp)
6. [Struttura del sito](#struttura)

1. **Obiettivo**

*Gazzetta Di Mantova Hub* nasce da un’idea semplice: restare aggiornati sulla cronaca locale attraverso un portale con un’interfaccia intuitiva, senza fronzoli e distrazioni. Sono volutamente assenti le immagini di copertina degli articoli, così da far concentrare li lettore esclusivamente sulle parole.

1. **Progettazione logica DB**

****

**UTENTI (IdUtente, NomeUtente, Password)**

**AUTORI (IdAutore, NomeAutore)**

**COMUNI (IdComune, Nome, Latitudine, Longitudine)**

**ARTICOLI (IdArticolo, Titolo, Testo, DataPubblicazione, *Comune*(FK)*, Autore*(FK))**

**PREFERISCE(*Utente*(FK), *Comune*(FK), DataPreferenza)**

**COMMENTA(*Utente*(FK), *Articolo*(FK), DataCommento, TestoCommento)**

1. **Progettazione fisica DB**

**CREATE TABLE utenti (**

**IdUtente INTEGER PRIMARY KEY NOT NULL AUTO\_INCREMENT,**

**NomeUtente VARCHAR(128) NOT NULL,**

**Password VARCHAR(512) NOT NULL,**

**UNIQUE(NomeUtente)**

**) ENGINE=InnoDB DEFAULT CHARSET=utf8;**

**CREATE TABLE autori (**

**IdAutore INTEGER PRIMARY KEY NOT NULL AUTO\_INCREMENT,**

**NomeAutore VARCHAR(128) NOT NULL,**

**UNIQUE(NomeAutore)**

**) ENGINE=InnoDB DEFAULT CHARSET=utf8;**

**CREATE TABLE comuni (**

**IdComune INTEGER PRIMARY KEY NOT NULL AUTO\_INCREMENT,**

**Nome VARCHAR(128) NOT NULL,**

**Latitudine FLOAT DEFAULT NULL,**

**Longitudine FLOAT DEFAULT NULL,**

**UNIQUE(Nome, Latitudine, Longitudine)**

**) ENGINE=InnoDB DEFAULT CHARSET=utf8;**

**CREATE TABLE articoli (**

**IdArticolo INTEGER PRIMARY KEY NOT NULL AUTO\_INCREMENT,**

**Titolo varchar(128) NOT NULL,**

**Testo varchar(4096) NOT NULL,**

**DataPubblicazione TIMESTAMP DEFAULT NULL,**

**Comune INTEGER DEFAULT NULL REFERENCES comuni(IdComune),**

**Autore INTEGER NOT NULL REFERENCES autori(IdAutore),**

**UNIQUE(Titolo),**

**FOREIGN KEY(Comune, Autore)**

**) ENGINE=InnoDB DEFAULT CHARSET=utf8;**

**CREATE TABLE preferisce (**

**Utente INTEGER NOT NULL REFERENCES utenti(IdUtente),**

**Comune INTEGER NOT NULL REFERENCES comuni(IdComune).**

**DataPreferenza DATE DEFAULT NULL,**

**PRIMARY KEY(Utente, Comune)**

**) ENGINE=InnoDB DEFAULT CHARSET=utf8;**

**CREATE TABLE commenti (**

**Utente INTEGER NOT NULL REFERENCES utenti(IdUtente),**

**Articolo INTEGER NOT NULL REFERENCES articoli(IdArticolo),**

**DataCommento TIMESTAMP NOT NULL,**

**TestoCommento VARCHAR(300)**

**PRIMARY KEY(Utente, Articolo, DataCommento),**

**) ENGINE=InnoDB DEFAULT CHARSET=utf8;**

1. **Interrogazioni SQL**

* home.php

// ritorna tutti i comuni nelle preferenze dell’utente

**SELECT Comune**

**FROM preferisce**

**WHERE Utente = <IdUtente>**

* feed.php

// ritorna tutti gli articoli e le posizioni della keyword nei loro titoli (tra quadre la parti opzionali che possono apparire o meno in base al contenuto delle barre di ricerca)

**SELECT \*, INSTR(Titolo, '$q')-1 AS KeywordLocation,** **, INSTR(Testo,'".urldecode($q)."')-1 AS KeywordLocationText**

**FROM articoli**

**WHERE Comune IN (<SELECT Comune**

**FROM preferisce**

**WHERE Utente = :IdUtente> |**

**<listaIdComuniSelezionati>)**

**AND Titolo LIKE '%:searchKeyword%'**

**[AND DataPubblicazione >= :da]**

**[AND DataPubblicazione <= :a]**

**ORDER BY DataPubblicazione DESC**

// ritorna tutti gli articoli, nel caso non ci sia nessuna keyword specificata(tra quadre la parti opzionali che possono apparire o meno in base al contenuto delle barre di ricerca)

**SELECT \***

**FROM articoli**

**WHERE Comune IN (<SELECT Comune**

**FROM preferisce**

**WHERE Utente = :IdUtente]> |**

**<listaIdComuniSelezionati>)**

**[AND DataPubblicazione >= :da]**

**[AND DataPubblicazione <= :a]**

**ORDER BY DataPubblicazione DESC**

* getLatestNews.php

// ritorna un massimo di NUM\_ARTICOLI articoli tra i più recenti

// NUM\_ARTICOLI è settabile dalle impostazioni del sito

**SELECT AR.\*, AU.NomeAutore, CO.Nome**

**FROM articoli AR, autori AU, comuni CO**

**WHERE AR.Comune = CO.IdComune**

**AND AR.Autore = AU.IdAutore**

**AND AR.Comune IN (SELECT Comune**

**FROM preferisce**

**WHERE Utente = :idUtente)**

**ORDER BY AR.DataPubblicazione DESC**

**LIMIT :NUM\_ARTICOLI**

* crawler.php

// trova l’id del comune a partire dal nome

**SELECT Nome**

**FROM comuni**

**WHERE IdComune = :idComune**

// trova l’id dell’autore a partire dal nome

**SELECT IdAutore**

**FROM autori**

**WHERE NomeAutore = :nomeAutore**

// se l’autore è un nuovo autore, inseriscilo nel DB

**INSERT INTO autori (NomeAutore)**

**VALUES (:nomeAutore)**

// una volta acquisiti tutti i dati, inserisci l’articolo nel DB

**INSERT INTO articoli (Titolo, Testo, DataPubblicazione, Comune, Autore)**

**VALUES (:titolo,:testo,:data,:comune,:autore)**

* articleShow.php

// preleva dati articolo a partire dall’id

**SELECT AR.\*, AU.NomeAutore, CO.Nome**

**FROM articoli AR, autori AU, comuni CO**

**WHERE AR.Autore = AU.IdAutore**

**AND AR.Comune = CO.IdComune**

**AND AR.IdArticolo = :idArticolo**

* login.php

// verifica se l’utente esiste

**SELECT \***

**FROM utenti**

**WHERE NomeUtente = :username AND Password = :pass**

* register.php

// inserisci il nuovo utente

**INSERT INTO utenti (NomeUtente, Password)**

**VALUES (:username,:pass)**

// inserisci le preferenze dell’utente (collega i comuni all’utente)

**INSERT INTO preferisce (Utente, Comune, DataPreferenza)**

**VALUES (:idUtente, :comune, NOW())**

// ritorna tutti i comuni da mostrare all’utente per effettuare la scelta di quelli preferiti

**SELECT IdComune, Nome**

**FROM comuni**

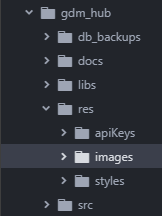
* accountShow.php

// preleva dati utente a partire dall’id

**SELECT \* FROM utenti**

**WHERE IdUtente = :IdUtente**

1. **Struttura del sito**

****

Documentazione

Risorse del sito

Librerie di terze parti (PHP simple DOM parser)

Eventuali token d’acesso alle web API

Immagini

Fogli di stile CSS

Backup della base di dati

Codice sorgente del sito