**Progetto #2 - Caso B**

Gruppo GossipWeb:

*Lara Longhi (1079261)*

*Federico Carrara (1080669)*

*Alessia Lazzari (1078863)*

*Consegna*:

Con i webservice deve essere possibile migrare i dati che avete inserito su Altervista nel 1° progetto e migrarli verso un database locale su Postgres

Il webservice remoto è consultabile al link: <http://servsanitariopw9.altervista.org/WS.php>

Il webservice locale e la servlet sono visibili al link: <https://github.com/LLonghi02/PW24_WebService>

**“Dump” del database di origine**: DB – Ex3: Servizio Sanitario

Immagine che contiene testo, schermata, software, Icona del computer

Descrizione generata automaticamente

|  |  |
| --- | --- |
| Tabella Cittadino Immagine che contiene testo, software, Icona del computer, Software multimediale  Descrizione generata automaticamente | Tabella Ospedale Immagine che contiene testo, schermata, software, Icona del computer  Descrizione generata automaticamente |
| Tabella Patologia | Tabella Ricovero  Immagine che contiene testo, schermata, software, Icona del computer  Descrizione generata automaticamente |

|  |  |  |
| --- | --- | --- |
| Tabella PatologiaCronica  Immagine che contiene testo, software, Icona del computer, Software multimediale  Descrizione generata automaticamente | Tabella PatologiaMortale | Tabella PatologiaRicovero  Immagine che contiene testo, schermata, software, Icona del computer  Descrizione generata automaticamente |

**Fasi per l’installazione**

1. Scarica o clona la seguente repository: <https://github.com/LLonghi02/PW24_WebService>
2. Avvia TomCat ed effettua il deploy del file WAR …
3. Avvia Django …

Descrizione file presenti nella cartella *PW24\_SSanitario*:

1. importer
2. java\_project

Il file web.xml serve per la configurazione di una servlet in un'applicazione web Java. Si specifica il nome della servlet (Servlet), la classe java che la implementa (com.example.SanitarioServlet), il nome della servlet a cui si fa riferimento e il pattern dell’URL a cui questa servlet risponderà.

La classe Serlet.java implementa una servlet Java che:

* 1. Effettua una richiesta HTTP GET a un webservice remoto.
  2. Legge e raccoglie la risposta da questo webservice.
  3. Invia i dati ottenuti a un altro webservice locale (applicazione Django) tramite una richiesta HTTP POST.

1. PW24\_SSanitario

wsgi.py: Questo file configura l'interfaccia WSGI (Web Server Gateway Interface) per il progetto Django. WSGI è uno standard per la comunicazione tra i server web e le applicazioni web scritte in Python.

urls.py: Questo file mappa gli URL alle viste corrispondenti nel progetto Django.

asgi.py: Questo file configura l'interfaccia ASGI (Asynchronous Server Gateway Interface) per il progetto Django. ASGI supporta applicazioni web asincrone in Python.

settings.py: Questo file contiene tutte le impostazioni di configurazione per il progetto Django.