1. **INTRODUZIONE**

Il tema d’anno svolto consiste nell’apportare modifiche nel progetto open source WEBVOWL (MIT license).

Le modifiche vertono sull’aggiunta delle feature di modifica dell’ontologia visualizzata, rendendo possibile quindi l’inserimento, la cancellazione o la modifica di un elemento.

La struttura del progetto WEBOWL è, nel suo insieme, ovviamente orientata alla visualizzazione dei dati e non alla loro modifica e, pertanto, è stato necessario sviluppare moduli per la modifica ed integrarli in quello che è il progetto originale. È stata trattata una restrizione degli aspetti di un’ontologia che è possibile modificare.

1. **WEBVOWL**

WEBVOWL è una applicazione web atta alla sola visualizzazione di ontologie. Implementa la visualizzazione delle notazioni per le ontologie OWL attraverso un grafo orientato che rappresenta l’ontologia stessa.

E’ bene specificare che l’applicativo lavora convertendo i file owl in formato vowl (formato accettato). La conversione avviene automaticamente in caso di IRI o URI mentre la visualizzazione di una ontologia in locale è possibile convertendo dapprima il file attraverso il software owl2vowl, applicativo java, scaricabile dal sito web di webvowl.

Webvolw, ad oggi, non permette la visualizzazione di tutti i costrutti di OWL2 o la combinazione di essi.

**2.1 Struttura del progetto**

L’applicativo è del tipo client side (utilizzo del solo JAVASCRIPT) ed è modularizzato mediante node.js.

Nei sorgenti, oltre ai file di stile css, sono presenti diversi script javascript suddivisi nelle cartelle webvowl e app. Nella cartella app sono presenti gli script per la gestione della UI (sidebar, menu, etc) mentre nella cartella webvowl sono presenti gli script relativi all’interpretazione e alla visualizzazione dell’ontologia.

Importanti tra questi sono:

* app/js/app.js: si occupa di inizializzare la pagina e di caricare l’ontologia.
* webvowl/js/graph.js: si occupa della rappresentazione dell’ontologia attraverso un grafo.
* Webvowl/js/parser.js: si occupa del parsing dell’ontologia e della relativa rappresentazione interna.
* Gruntfile.js: si occupa di importare nella cartella deploy tutti i file esterni e di supporto necessari al funzionamento dell’applicativo.
* Index.html: corpo della pagina web di webvowl

Nella cartella deploy sono contenuti i sorgenti compilati.

1. **MODIFICHE APPORTATE**

Essendo il progetto originale orientato alla sola visualizzazione di ontologie si è reso necessario apportare modifiche ai sorgenti originali e sviluppare ulteriori script esterni che abilitassero anche la modifica.

Per assicurare una facilità di integrazione delle modifiche sviluppate in future versioni, si è cercato di limitare i cambiamenti ai file originali, preferendo lo sviluppo di moduli esterni.

In particolare, dei file originali sono stati modificati:

* Graph.js: si è inserito uno script che abilitasse la cattura del click del tasto destro sul nodo e visualizzazione del relativo menu contestuale (righe 810-857). È stato inoltre aggiunta la referenza al nodo cliccato dall’utente per la sua relativa elaborazione.
* Index.html: si è modificata la struttura dell’idex.html per la disabilitazione del menu contestuale di default del browser (194-206).
* App.js: esternalizzazione della funzione “loadOntologyFromText” in modo tale sia accessibile dai moduli da noi creati (143-199).
* Gruntfile.js: sono state aggiunti tra gli “import” i file .css, .js e .html da noi creati (69,75,76).
* Parser.js: data la natura differente della elaborazione dei dati che è necessario effettuare nel caso della sola visualizzazione di una ontologia piuttosto che anche della sua modifica, sì è scelto di sviluppare un modulo esterno (myparser.js) che trattasse i dati in modo tale da rendere possibile anche la modifica degli stessi.

Sono stati sviluppati i seguenti moduli:

* Myparser.js: script che si occupa di effettuare un re-parsing dell’ontologia e implementa i metodi per modificarla ed il reperimento dei dati dalla stessa.
* Page.js: script atto alla creazione della pagina popup, mostrata all’utente, per la modifica (insert, edit, delete di un nodo).
* pop.html: pagina popup da mostrare all’utente.
* Css vari per lo styling della pagina.