

Realizzare una piccola chat in php senza database

L'obiettivo di questa guida è implementare una semplice *chat in linguaggio php*, facilmente inseribile in qualsiasi sito.

La chat che realizzeremo avrà il vantaggio di non dover gestire un *database*, bensì ogni singolo messaggio verrà immagazzinato in un semplice *file di testo*.

Di conseguenza è giusto premettere che la realizzazione di una chat sottoposta a un grande traffico richiede assolutamente la gestione di un database.

La chat che realizzeremo sarà essenziale. Permetterà di inviare un messaggio, dopo aver inserito il proprio nick e visualizzarlo di conseguenza a video.

Sarà utilizzata la tecnologia **Ajax** per aggiornare dinamicamente il box dei messaggi a intervalli regolari.

I file che andremo a realizzare sono essenzialmente 3:

1. **index.php**
2. **messaggio.php**
3. **msg.html**

Il terzo file in elenco (msg.html) sarà lasciato volutamente vuoto perché il suo scopo è mantenere i messaggi di chat inviati.

Il primo passo da compiere è realizzare una semplice *form* in html che possa permettere di inserire il nick e il messaggio e quindi fornire il bottone di invio:

```
<div id="sender" align="center" onKeyUp="keypressed(event);">
<label>Il tuo nickname</label><input type="text" name="nic" size="30" id="nic" />
<label>Il tuo messaggio</label><textarea name="msg" id="msg" cols="30"
rows="3"></textarea>
<button onClick="doWork();">Invia</button>
</div>
```

Nelle linee di codice precedenti sono state inserite le chiamate ad alcune funzioni, *keypressed* e *doWork*, che verranno analizzate in avanti.

Per ora l'importante è aver realizzato il modulo che permetterà all'utente di inviare i propri messaggi.

Il secondo passo è realizzare un box dove far comparire i messaggi di chat:

```
<div id="result">
  <?php
    $data = file("msg.html");
```

```

        foreach ($data as $line) {
            echo $line;
        }
    ?>
</div>

```

Nel *div result* verranno stampati i messaggi tramite la funzione *echo* di php i quali, come detto prima, proverranno dal file msg.html.

Infatti file msg.html viene aperto in lettura e viene effettuato un ciclo fino all'ultima linea presente nel file, cioè l'ultimo messaggio.

Per completare il file index.php è necessario inserire le funzioni in javascript che permettano di gestire le varie funzionalità della chat lato client:

```

<script language="javascript" type="text/javascript">
var httpObject = null;
var link = "";
var timerID = 0;

function getHTTPObject(){
    if(window.ActiveXObject)
        return new ActiveXObject("Microsoft.XMLHTTP");
    else if (window.XMLHttpRequest) return new XMLHttpRequest();
    else {
        alert("Il tuo browser non supporta AJAX.");
        return null;
    }
}

function setOutput(){
    if(httpObject.readyState == 4){
        var response = httpObject.responseText;
        var objDiv = document.getElementById("result");
        objDiv.innerHTML += response;
        objDiv.scrollTop = objDiv.scrollHeight;
        var inpObj = document.getElementById("msg");
        inpObj.value = "";
        inpObj.focus();
    }
}

function setAll(){
    if(httpObject.readyState == 4){
        var response = httpObject.responseText;

```

```

        var objDiv = document.getElementById("result");
        objDiv.innerHTML = response;
        objDiv.scrollTop = objDiv.scrollHeight;
    }
}

function doWork(){
    if(document.getElementById('nic').value.length<1||document.getElementById('
    msg').value<1){
        alert("Campi vuoti");
    }
    else{
        httpObject = getHTTPObject();
        if (httpObject != null) {
            link="messaggio.php?nick="+document.getElementById('nic').value+
"&msg="+document.getElementById('msg').value;
            httpObject.open("GET", link , true);
            httpObject.onreadystatechange = setOutput;
            httpObject.send(null);
        }
    }
}

function doReload(){
    httpObject = getHTTPObject();
    var randomnumber=Math.floor(Math.random()*10000);
    if (httpObject != null) {
        link = "messaggio.php?all=1&rnd="+randomnumber;
        httpObject.open("GET", link , true);
        httpObject.onreadystatechange = setAll;
        httpObject.send(null);
    }
}

function UpdateTimer() {
    doReload();
    timerID = setTimeout("UpdateTimer()", 5000);
}

function keypressed(e){
    if(e.keyCode=='13'){
        doWork();
    }
}

```

</script>

La funzione cuore della chat lato client è **doWork**.

Viene prelevato l'oggetto http, realizzata la stringa link che incapsula il nick dell'utente e il messaggio, nonché il nome del file che riceverà tramite Get la stringa in questione:

```
link="messaggio.php?nick="+document.getElementById('nic').value+"&msg="+  
document.getElementById('msg').value;  
httpObject.open("GET", link , true);
```

Viene quindi inviato il messaggio, non prima di aver ripulito la form di invio:

```
httpObject.onreadystatechange = setOutput;  
httpObject.send(null);
```

I metodi *UpdateTimer* e *doReload* permettono di ricaricare a intervalli regolari la chat o più precisamente il box result e quindi aggiornare i messaggi. A tale scopo è importante sottolineare l'importanza ricoperta dall'oggetto XMLHttpRequest e quindi dallo strumento di sviluppo Ajax che permette appunto di ricaricare dinamicamente una sezione di una pagina web.

La funzione keypressed controlla invece il tasto invio, tramutando tale azione nell'invio del messaggio.

Il file index.php è così concluso:

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "DTD/xhtml1-  
transitional.dtd">  
<html>  
<head>  
  <title>Chat php</title>  
  <script language="javascript" type="text/javascript">  
    <!--  
    var httpObject = null;  
    var link = "";  
    var timerID = 0;  
  
    function getHTTPObject(){  
      if (window.ActiveXObject) return new ActiveXObject("Microsoft.XMLHTTP");  
      else if (window.XMLHttpRequest) return new XMLHttpRequest();  
      else {  
        alert("Your browser does not support AJAX.");  
        return null;  
      }  
    }  
  }  
</script>  
</head>  
</html>
```

```

function setOutput(){
    if(httpObject.readyState == 4){
        var response = httpObject.responseText;
        var objDiv = document.getElementById("result");
        objDiv.innerHTML += response;
        objDiv.scrollTop = objDiv.scrollHeight;
        var inpObj = document.getElementById("msg");
        inpObj.value = "";
        inpObj.focus();
    }
}

function setAll(){
    if(httpObject.readyState == 4){
        var response = httpObject.responseText;
        var objDiv = document.getElementById("result");
        objDiv.innerHTML = response;
        objDiv.scrollTop = objDiv.scrollHeight;
    }
}

function doWork(){
    if(document.getElementById('nic').value.length<1||document.getElementById('msg').value<1)
    {
        alert("Campi vuoti");
    }
    else{
        httpObject = getHTTPObject();
        if (httpObject != null) {
            link="messaggio.php?nick="+document.getElementById('nic').value+
            "&msg="+document.getElementById('msg').value;
            httpObject.open("GET", link , true);
            httpObject.onreadystatechange = setOutput;
            httpObject.send(null);
        }
    }
}

function doReload(){
    httpObject = getHTTPObject();
    var randomnumber=Math.floor(Math.random()*10000);
    if (httpObject != null) {
        link = "messaggio.php?all=1&rnd="+randomnumber;
        httpObject.open("GET", link , true);
        httpObject.onreadystatechange = setAll;
        httpObject.send(null);
    }
}

```

```

function UpdateTimer() {
    doReload();
    timerID = setTimeout("UpdateTimer()", 5000);
}

function keypressed(e){
    if(e.keyCode=='13'){
        doWork();
    }
}
</script>
</head>
<body onLoad="UpdateTimer();">
<div id="result">
    <?php
        $data = file("msg.html");
        foreach ($data as $line) {
            echo $line;
        }
    ?>
</div>
<div id="sender" align="center" onKeyUp="keypressed(event);">
<label>Il tuo nickname</label><input type="text" name="nic" size="30" id="nic" />
<label>Il tuo messaggio</label><textarea name="msg" id="msg" cols="30"
rows="3"></textarea>
<button onClick="doWork();">Invia</button>
</div>
</body>
</html>

```

Il secondo file che implementeremo sarà *messaggio.php* che a tutti gli effetti descrive la logica lato server della nostra applicazione.

Essenzialmente lo scopo di questo file sarà quello di prelevare il messaggio dal Get e scompattarlo per ricavare nick e messaggio. Successivamente sarà calcolato un timestamp (la data attuale) che verrà accodato al messaggio, non prima di aver realizzato la stringa html da scrivere nel file msg.html.

Ecco il codice di messaggio.php :

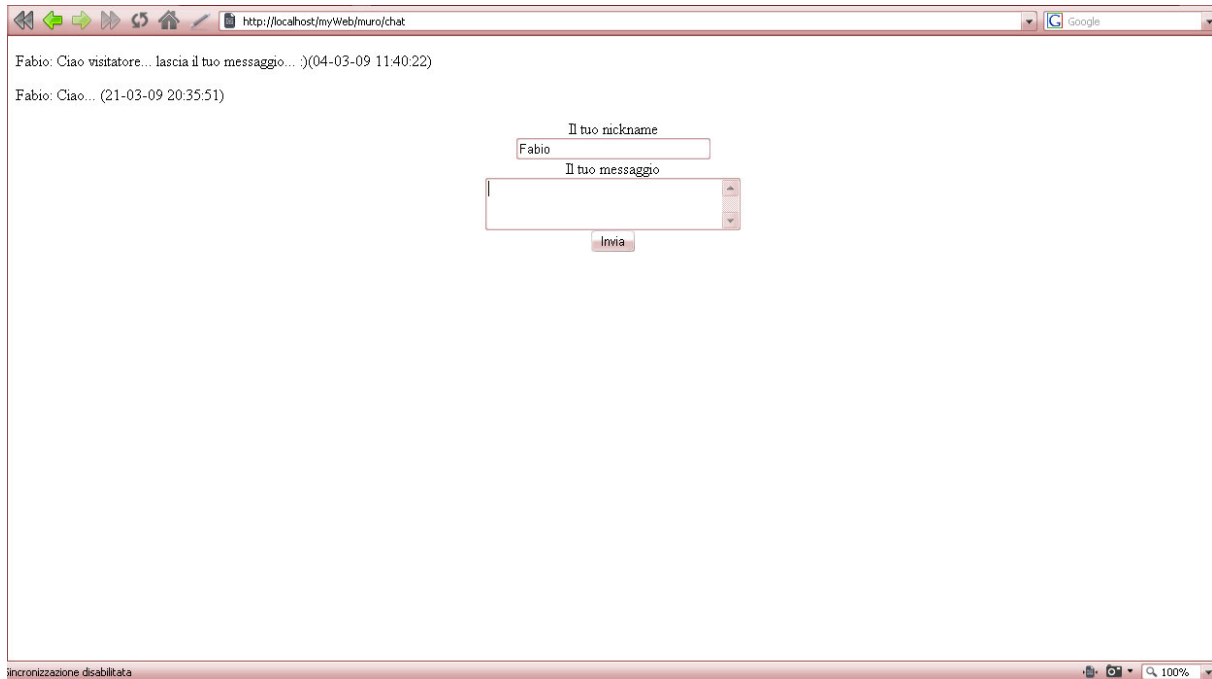
```
<?php
    if (isset($_GET['msg'])){
        if (file_exists('msg.html')) {
            $f = fopen('msg.html',"a+");
        } else {
            $f = fopen('msg.html',"w+");
        }
        $nick = isset($_GET['nick']) ? $_GET['nick'] : "Hidden";
        $msg=stripslashes(htmlentities($_GET['msg'], ENT_QUOTES));
        $data=date("d-m-y");
        $ora=date("G:i:s");
        $line = "<p><span>$nick: </span><span>$msg ($data $ora)</span></p>";
        fwrite($f,$line."\r\n");
        fclose($f);
        echo $line;

    } else if (isset($_GET['all'])) {
        $flag = file('msg.html');
        $content = "";
        foreach ($flag as $value) {
            $content .= $value;
        }
        echo $content;
    }
?>
```

Come detto il messaggio viene scritto nel file incapsulato in un paragrafo <p>. Il messaggio viene quindi stampato a video tramite echo.

Importante analizzare la seconda parte del file messaggio.php, nel quale viene stampato un messaggio vuoto quando risulta necessario ricaricare i messaggi di chat. Infatti questa procedura viene messa in moto nel momento in cui nel messaggio ricevuto viene identificato il parametro all, inviato appunto dal metodo doReload.

Ecco il risultato finale della nostra piccola chat:



Per migliorare l'aspetto grafico della nostra chat si consiglia di realizzare un foglio di stile da associare al file index.php nel quale impostare una grandezza fissa del box result onde evitare lo scorrimento in basso del modulo di inserimento messaggi.