

```
// + INCLUDES
dekiapi();

// + PARAMETERS
var par = {
    number: __request.args.number ?? args.number ?? 10,
    language: __request.args.language ?? args.language ?? ( page.language
!= ' ' ? page.language : ( site.language != '' ? site.language : 'en-us' ) ),
    ajax_action: String.tolower( __request.args.ajax_action ??
args.ajax_action ?? 'load' )
};

if ( (par.ajax_action=='load') ||
(par.ajax_action=='reload_gestorcomentarios') ) {

// + CONSTANTS
//var NO_AVATAR = Site.Api..' /files/966/=NoAvatar.jpg';
var NO_AVATAR = '';
var LANGUAGE_ES = {
    see_all: 'Ver Todos',
    see_more: 'Ver mas...',
    at: 'a',

    add_msg: 'Añadir mensaje:',
    edit_msg: 'Edición del mensaje:',
    msg_reply: 'Respuesta al mensaje:',

    write_msg: 'Escribe el mensaje a añadir.',
    question_delete_message: 'Seguro que quiere eliminar este mensaje?',
    error_msg_edit_mode_unknown: 'Modo de edición desconocido',

    logintocomment_msg : 'Los comentarios Anónimos no están permitidos!
<br />Puedes iniciar la sesión utilizando tu proveedor OpenID favorito:',
    logintocomment_link: 'Clica aquí para añadir tu comentario ...',

    add_msg_button: 'Adjuntar Mensaje',
    cancel_msg_button: 'Cancelar',
    reply_msg_button: 'Responder',
    edit_msg_button: 'Editar',
    delete_msg_button: 'Borrarr',
    save_msg_button: 'Guardar Cambios'

};
var LANGUAGE_EN = {
    see_all: 'Show all',
    see_more: 'Show more...',
    at: 'at',

    add_msg: 'Add message:',
```

```

edit_msg: 'Edit message:',
msg_reply: 'Message answer:',

write_msg: 'Write message text here.',
question_delete_message: 'Do you want to delete the message?',
error_msg_edit_mode_unknown: 'ERROR: Unknown edit mode',

logintocomment_msg : 'Anonymous comments are not allowed! <br />You
can log-in using your favorite OpenID provider:',
logintocomment_link: 'Click here to add your comment ...',

add_msg_button: 'Add message',
cancel_msg_button: 'Cancel',
reply_msg_button: 'Reply',
edit_msg_button: 'Edit',
delete_msg_button: 'Delete',
save_msg_button: 'Save changes'
};

//#add @himikel
var LANGUAGE_EU = {
  see_all: 'Ikusi guztiak',
  see_more: 'Ikusi gehiago...',
  at: 'at',

  add_msg: 'Gehitu mezua:',
  edit_msg: 'Editatu mezua:',
  msg_reply: 'Erantzun mezua:',

  write_msg: 'Idatzi hemen testu-mezua.',
  question_delete_message: 'Mezua ezabatu nahi al duzu?',
  error_msg_edit_mode_unknown: 'ERROREA: Edizio modu ezezaguna',

  logintocomment_msg : 'Iruzkin anonimoak ez daude baimendurik! <br
/>Saioa hasi dezakezu gustokoen duzun OpenID hornitzaileaz:',
  logintocomment_link: 'Klikatu hemen zure iruzkina gehitzeko ...',

  add_msg_button: 'Erantsi mezua',
  cancel_msg_button: 'Utzi',
  reply_msg_button: 'Erantzun',
  edit_msg_button: 'Editatu',
  delete_msg_button: 'Ezabatu',
  save_msg_button: 'Gorde aldaketak'
};

var TXTS = { 'es': LANGUAGE_ES, 'es-es': LANGUAGE_ES, 'en': LANGUAGE_EN,
'en-us': LANGUAGE_EN, 'eu': LANGUAGE_EU, 'es-eu': LANGUAGE_EU }; //#mod
@himikel
var lg = par.language;
if (!TXTS[lg])

```

```

let lg = 'en-us';

/** It needs a closing DIV */
var HTML_COMMENT = "...
" <div id='gc_comment_[cindex]' class='gc_comment'>"...
" <div id='gc_comment_content_[cindex]'
class='gc_comment_content[cnew]'>"...
" <table width='100%' class='gc_table'>"...
" <tr id='gc_comment_tr_[cindex]'>"...
" <td style='width: 35px; text-align: center' valign='top'><a
name='comment[cindex]'>[uavatar]<BR />#[cindex]</a></td>"...
" <td valign='top'>"...
" <span class='gc_comment_title'><a href=[uuri] rel='internal'
id='gc_author_[cindex]'>[uname]</a> "..TXTS[lg].at.." [cdate]</span>"...
" <div id='gc_comment_text_[cindex]'
class='gc_comment_text'>[ctext]</div>"...
" </td>"...
" </tr>"...
" </table>"...
" </div>";
// -- !!!! It need's a closing DIV, but i can't put it in the CONSTANT
because it depends on the "recursive" calls !!!!

// + VARS
var cache_avatars = {};
var avatar,source,n,a;

// + DYNAMIC HTML CONTENT
<div id="gc_comentarios" width="100%">

var comments = page.comments;
var xcomments = [ k .. { index:
(String.Replace(String.Split(k.uri,'#')[1],'comment','')), aindex:
__index } foreach var k in comments ];
var ncomments = #xcomments;
var ccomments;
var ccomment;
var htmlout = "";
var thtmlout = "";

// -- Push on the stack on reverse order;
var trobat;
var stack = [];
var new_messages = [];
var roots = [ ];
var lastpage = false;

```

```

if ((ncomments<=par.number) || (par.number=='all')) {
  let stack = [ List.Splice(xcomments,0,__index) foreach var k in
xcomments where !String.StartsWith(k.text,'#')]
  let roots = [ (k[0].index) foreach var k in stack];
  let lastpage = true;
  let stack = List.Reverse(stack);
}
else
{
  var rxcomments = List.Reverse(xcomments);
  var parent;

  foreach(var k in rxcomments) {
    let new_messages ..= [ (k.index) ];
    let trobat = k;
    // -- Let's search for the root of this comment
    foreach(var tonto in Num.Series(0,k.aindex)) {
      if (String.StartsWith(trobat.text,'#')) {
        let parent =
String.SubStr(trobat.text,1,String.IndexOf(trobat.text,':')-1);
        let trobat = List.Select(xcomments,'$.index=='..parent);
        if (#trobat>0)
          { let trobat = trobat[0]; };
      }
      else { break; }
    }

    if (!List.Contains(roots,trobat.index)) {
      let roots ..= [ (trobat.index) ];
      let stack ..= [ List.Splice(xcomments,0,trobat.aindex) ];
    }

    if (__index==(par.number-1)) { break; }
  }
}

if (lastpage) {
  <a href="#" id='gc_ver_mas' number=(par.number) style='display:
none'>TXTS[lg].see_more</a>
  " ";
  <a href="#" id='gc_ver_todos' style='display:
none'>TXTS[lg].see_all</a><br />
}
else
{
  <a href="#" id='gc_ver_mas' number=(par.number)>TXTS[lg].see_more</a>
  " ";
  <a href="#" id='gc_ver_todos'>TXTS[lg].see_all</a><br />
}

```

```

var tstack = [];
let max_iterations = #xcomments*#xcomments; // Crec que seria n-log n
pero ara no estic per calcular el logaritme

foreach (var i in Num.Series(0,max_iterations)) {

    if (#stack==0) { break; }
    // -- Pop on the stack
    let ccomments = stack[#stack-1];
    let stack = List.Splice(stack,#stack-1,1);
    // -- End Pop
    let ccomment = ccomments[0];
    if ( ccomment.index==--400) { let htmlout ..="</div>"; }
    else
    {
        if (List.Contains(roots,ccomment.index)) { let htmlout ..='<hr
/>'; }
        let ccomments = List.Splice(ccomments,0,1);

        // -- Let Search for the avatar and save it to the cache.
        let un = ccomment.author.name;
        if (!Map.Contains(cache_avatars,un)) {
            let source=wiki.getpage('/User:' .. un).files;
            let avatar=NO_AVATAR;
            foreach (var f in source) {
                let n=f.name;
                let a=f.api;
                if (String.tolower(n) == 'avatar.jpg') { let avatar=f.api; }
            }
            if (avatar==NO_AVATAR) { let avatar =
ccomment.author.gravatar; }
            let cache_avatars ..= {(un): ("<img src='\"..avatar..\"'
width='30px' valing='top' />") };
        }

        // -- Let search hyperlinks and add href - #add @himikel
        var comhref=ccomment.text;
        var imov=0;
        foreach (var ihl in String.IndexesOf(comhref,"http://")) {
            let ihl+=imov;
            var href=String.SubStr(comhref,ihl);
            var endpos=List.Min(List.Select([String.IndexOf(href,"
"),String.IndexOf(href,"\\n"),String.Length(href)],"$>0"));
            var shltxt=String.SubStr(href,0,endpos);
            var shlref="<a href='\"..shltxt..\"'>\"..shltxt..\"</a>";
            let imov+=String.Length(shlref)-String.Length(shltxt);
            let
comhref=String.SubStr(comhref,0,ihl)..shlref..String.SubStr(comhref,ihl+String.L
}

```

```

    // -- Let fill the comment template
    let thtmlout =
String.Replace(HTML_COMMENT, '[cindex]', ccomment.index);
    let thtmlout =
String.Replace(thtmlout, '[cnew]', (List.Contains(new_messages, ccomment.index)? '_n
    let thtmlout =
String.Replace(thtmlout, '[uavatar]', cache_avatars[un]);
    let thtmlout =
String.Replace(thtmlout, '[uuri]', ccomment.author.uri);
    let thtmlout =
String.Replace(thtmlout, '[uname]', ccomment.author.name);
    // let thtmlout =
String.Replace(thtmlout, '[cdate]', date.format(date.changetimezone(ccomment.date,
MM/yyyy - HH:mm")));
    let thtmlout =
String.Replace(thtmlout, '[cdate]', date.format(date.addhours(ccomment.date, 2), "dd
MM/yyyy - HH:mm"));
    let thtmlout = String.Replace(thtmlout, '[ctext]', comhref); // #mod
@himikel
    //let thtmlout =
String.Replace(thtmlout, '[ctext]', ccomment.text); // #del @himikel

    // -- Let concatenate the comment template to the whole comments
html out
    let htmlout .= thtmlout;

    // -- Let's do the "recursive" thing (iterating).
    let tstack = [];
    foreach (var knext in xcomments) {
        if (String.StartsWith(knext.text, "#"..ccomment.index..":")) {
            // -- Push on the stack
            let tstack .= [ (List.Splice(xcomments, 0, __index)) ];
        }
    }
    if (#tstack)
    {
        let tstack .= [ [{ index: -400 } ] ];
        let stack .= List.Reverse(tstack);
    }
    else
    { let stack .= [ [{ index: -400} ] ]; }
}

web.html(htmlout);

</div>

if (par.ajax_action=='load') {

```

```

// + STATIC HTML CONTENT
//@himikel add 2010-05-27 : Only Authenticated Users can Add Comments
var sMsgFormStyle="";
if (!user.anonymous)
  let sMsgFormStyle="display: inline";
else
  let sMsgFormStyle="display: none";

//@himikel mod 2010-05-27 : Only Authenticated Users can Add Comments
// -- Sending messages Form
<form id="gc_messageform" onsubmit="#gc_messageform.value='';"
style=(sMsgFormStyle)>
  <span id="messageheaderspan1">TXTS[lg].add_msg</span>
  <BR />
  <textarea id="gc_messageform" class='gc_messageform_class'
rows='10'>TXTS[lg].write_msg</textarea>
  <br />
  <input type="button" id="gc_sendmessage"
value=(TXTS[lg].add_msg_button) />
</form>

//@himikel add 2010-05-27 : Show message to Anonymous users and link to
log-in by OpenID
if (user.anonymous)
{
  //Anonymous Users need to log-in by OpenID first (or using native
account)
  var
sLoginUrl=site.uri.."Special:OpenIdLogin?returntotitle="..Uri.Encode(Uri.Encode(
  var sMsg="<br />"..TXTS[lg].logintocomment_msg;
  let sMsg.."<br /><b><a href='"..sLoginUrl
.."'">..TXTS[lg].logintocomment_link.."</a></b><br /><br />";
  Web.Html(sMsg);
}//if

// -- Reply & Edit Messages Form
<div id='gc_f_load_messageform' style='display: none; position:
absolute;'>
  <form id="gc_f_messageform" onsubmit="#gc_f_messageform.value='';">
    <span id="messageheaderspan2">TXTS[lg].msg_reply</span>
    <BR />
    <textarea id="gc_f_messageform" class='gc_messageform_class'
rows='10' ></textarea>
    <br />
    <input type="button" id="gc_f_sendmessage"
value=(TXTS[lg].add_msg_button) />
    <input type="button" id="gc_f_cancelmessage"
value=(TXTS[lg].cancel_msg_button) />
  </form>

```

```
</div>
```

```
// -- Button Bars
```

```
<div id='gc_button_bar' style='display: none; position: absolute;' >
```

```
  <a id='gc_reply'><span class='icon'><img src='/skins/common/icons/
  icon-trans.gif' class='move' alt='' /><span
  class='text'>TXTS[lg].reply_msg_button</span></span></a>
```

```
    " ";
```

```
  <a id='gc_edit'><span class='icon'><img src='/skins/common/icons/
  icon-trans.gif' class='edit' alt='' /><span
  class='text'>TXTS[lg].edit_msg_button</span></span></a>
```

```
    " ";
```

```
  <a id='gc_delete'><span class='icon'><img src='/skins/common/icons/
  icon-trans.gif' class='delete' alt='' /><span
  class='text'>TXTS[lg].delete_msg_button</span></span></a>
```

```
</div>
```

```
// ++ EVENT MESSAGE MANAGER
```

```
<script type="text/jem">
```

```
var message_id = 0;
```

```
function Right(str, n) {
```

```
  if (n <= 0)
```

```
    return '';
```

```
  else if (n > String(str).length)
```

```
    return str;
```

```
  else {
```

```
    var iLen = String(str).length;
```

```
    return String(str).substring(iLen, iLen - n);
```

```
  }
```

```
}
```

```
when(#gc_sendmessage.click) {
```

```
  @gc_sendmessage_msg({confirmmessagesent: 'true', message:
```

```
#gc_messagegettext.val()});
```

```
  #gc_messagegettext.val('');
```

```
}
```

```
when(#gc_reply.click) {
```

```
  if (!gc_edit_mode) {
```

```
    gc_edit_mode = 1;
```

```
    $('#messageheaderspan2').text('"..TXTS[lg].msg_reply..");
```

```
    $('#gc_f_sendmessage').val('"..TXTS[lg].add_msg_button..");
```

```
    $('#gc_f_load_messageform').css('display', 'block');
```

```
    $('#gc_messageform').css('display', 'none');
```

```
    $('#gc_comment_tr_'+gc_current_reply).after('<tr
```

```
id=gc_dummy_row><td colspan=2><div style=\"height:
```

```
'+gc_f_load_messageform_height+'px;\"></div></td></tr>');
```

```
$('#gc_f_load_messageform').css('top', $('#gc_dummy_row').position().top);
```

```

$('#gc_f_load_messageform').css('left',$('#gc_dummy_row').position().left+20);

    }
}

when(#gc_edit.click) {
    if (!gc_edit_mode) {
        gc_edit_mode = 2;
        $('#messageheaderspan2').text('"..TXTS[lg].edit_msg..');
        $('#gc_f_sendmessage').val('"..TXTS[lg].save_msg_button..');
        $('#gc_comment_text_'+gc_current_reply).css('display','none');
        $('#gc_f_load_messageform').css('display','block');
        $('#gc_messageform').css('display','none');
        $('#gc_comment_tr_'+gc_current_reply).after('<tr
id=gc_dummy_row><td colspan=2><div style=\"height:
'+gc_f_load_messageform_height+'px;\"></div></td></tr>');

        $('#gc_f_load_messageform').css('top',$('#gc_dummy_row').position().top);

        $('#gc_f_load_messageform').css('left',$('#gc_dummy_row').position().left+20);

        #gc_f_mesagetext.val($('#gc_comment_text_'+gc_current_reply).text());
    }
}

when(#gc_delete.click) {
    if (!gc_edit_mode) {
        gc_edit_mode = 3;
        if (confirm('"..TXTS[lg].question_delete_message..')) {
            Deki.$.get( '/deki/gui/comments.php', {
                'action' : 'delete',
                'titleId' : Deki.PageId,
                'commentNum' : gc_current_reply
            }, function( data ) {
                Deki.$('#gc_comment_'+gc_current_reply).remove();
                gc_edit_mode = 0;
            }, 'html' );
        }
        else { gc_edit_mode = 0; }
    }
}

when(#gc_f_cancelmessage.click) {
    if(gc_edit_mode==2) {
        $('#gc_comment_text_'+gc_current_reply).css('display','block');
    }
    $('#gc_f_load_messageform').css('display','none');
}

```

```

    $('#gc_button_bar').css('display','none');
    $('#gc_messageform').css('display','inline');
    $('#gc_dummy_row').remove();
    #gc_f_messagetext.val('');
    gc_edit_mode = 0;
}

when(#gc_f_sendmessage.click) {
    if (gc_edit_mode) {
        if (gc_edit_mode==1) {
            @gc_sendmessage_msg({confirmmessagesent: 'true',reply_to:
gc_current_reply, message: #gc_f_messagetext.val()});
        }
        else
        {
            if (gc_edit_mode==2) {
                @gc_sendmessage_msg({confirmmessagesent: 'true',
message_id: gc_current_reply, message: #gc_f_messagetext.val()});
            }
            else { alert('"..TXTS[lg].error_msg_edit_mode_unknown..");
gc_edit_mode =0; }
        }
        $('#gc_f_load_messageform').css('display','none');
        $('#gc_button_bar').css('display','none');
        $('#gc_messageform').css('display','inline');
        $('#gc_dummy_row').remove();
        #gc_f_messagetext.val('');
    }
}

when(@gc_sendmessage_msg){
    var message = @gc_sendmessage_msg.message;
    var now = new Date();
    message_id = 0;
    if (@gc_sendmessage_msg.message_id) { message_id =
@gc_sendmessage_msg.message_id; }
    var reply_to = @gc_sendmessage_msg.reply_to;
    if (reply_to) { message = '#'+reply_to+' : '+message; }

    Deki.$.post( '/deki/gui/comments.php', {
        'action' : 'post',
        'titleId' : Deki.PageId,
        'comment' : message,
        'showAll' : MTComments.ViewingAll,
        'commentNum' : ((message_id==0) ? null :
message_id )
    }, function( data ) {

```

```

        if (message_id) {

$( '#gc_comment_text_'+gc_current_reply ).text (message);

$( '#gc_comment_text_'+gc_current_reply ).css ('display','block');
        }
        else
        {
            var lcomments = data.match (/comment\\d+/gi);
            var id_new_comment =
lcomments[lcomments.length-1].replace (/comment/, '');
            var html_nc = HTML_COMMENT;
            html_nc = html_nc.replace (/\\\[cindex\\]/
g, id_new_comment);

            html_nc = html_nc.replace (/\\\[ctext\\]/
g, message);

            html_nc = html_nc.replace (/\\\[cnew\\]/g, '_new');
            html_nc = html_nc.replace (/\\\[cdate\\]/
g, Right ('00'+now.getDate (), 2)+'/'+Right ('00'+(now.getMonth ()+1), 2)+'/'
'+now.getFullYear ()+' - '+now.getHours ()+':'+now.getMinutes ());
            if (reply_to) {
Deki.$ ('#gc_comment_'+gc_current_reply ).append (html_nc); }
                else { Deki.$ ('#gc_comentarios').append ('<hr
/>'+html_nc); }

                    GestorComentariosControls ();
                }
                gc_edit_mode = 0;
            }, 'html' );

        }

"</script>

// ++ GLOBAL EVENT BINDINGS
var javascript_html_comment = HTML_COMMENT.."</div>";

// -- Let Search for the avatar and save it to the cache.
let un = user.name;
if (!Map.Contains (cache_avatars, un)) {
    let source=wiki.getpage ('/User:' .. un).files;
    let avatar=NO_AVATAR;
    foreach (var f in source) {
        let n=f.name;
        let a=f.api;
        if (String.toLowerCase (n) == 'avatar.jpg') { let avatar=f.api; }
    }
    if (avatar==NO_AVATAR) { let avatar = user.gravatar; }
    let cache_avatars ..= {(un): ("<img src='\"..avatar..\"' width='30px'
valing='top' />") };

```

```

}

let javascript_html_comment =
String.Replace(javascript_html_comment, '[uavatar]', cache_avatars[user.name]);
let javascript_html_comment =
String.Replace(javascript_html_comment, '[uuri]', user.uri);
let javascript_html_comment =
String.Replace(javascript_html_comment, '[uname]', user.name);

<script type="text/jem">
var gc_f_load_messageform_height = 0;
var gc_button_bar_width = 0;
var gc_current_reply = -1;
var gc_edit_mode = 0;
var gc_send_message_no_repeat = 0;
var HTML_COMMENT = \"\"..javascript_html_comment..\"\";

Deki.$(document).ready(function() {

    Deki.$('#comments').remove();

    $('#gc_button_bar').show();
    gc_button_bar_width = $('#gc_button_bar').width();
    $('#gc_button_bar').hide();

    $('#gc_f_load_messageform').show();
    gc_f_load_messageform_height =
$('#gc_f_load_messageform').height();
    $('#gc_f_load_messageform').hide();

});

function GestorComentariosControls() {

// --- GESTIÓ EDICIÓ COMENTARI
var f_mouseover = function() {
    if (!gc_edit_mode) {
        Deki.$(this).css({'border': '1px solid red'});
        var arr = Deki.$(this).attr('id').split('_');
        gc_current_reply = arr[arr.length-1];
        var idcomment = '#gc_comment_'+gc_current_reply;

Deki.$('#gc_button_bar').css('top', Deki.$('#gc_comment_'+gc_current_reply).posit

Deki.$('#gc_button_bar').css('left', Deki.$('#gc_comment_'+gc_current_reply).posi
        Deki.$('#gc_button_bar').css('display', 'block');

        var author = Deki.$('#gc_author_'+gc_current_reply).html();

//@himikel add 2010-05-27 : Only Authenticated Users can Reply

```

## Comments

```

if ('true'!='"..user.anonymous..")
    Deki.$('#gc_reply').css('display','inline');
else
    Deki.$('#gc_reply').css('display','none');

if ((author=="..user.name..") || ('true'=="..user.admin..")) {
    Deki.$('#gc_edit').css('display','inline');
    Deki.$('#gc_delete').css('display','inline');
}
else
{
    Deki.$('#gc_edit').css('display','none');
    Deki.$('#gc_delete').css('display','none');
}
}

};

Deki.$('.gc_comment_content').bind('mouseover',f_mouseover);
Deki.$('.gc_comment_content_new').bind('mouseover',f_mouseover);

var f_mouseout = function() {
    if (!gc_edit_mode) {
        Deki.$(this).css({'border': '1px solid white'});
    }
};

Deki.$('.gc_comment_content').bind('mouseout',f_mouseout );
Deki.$('.gc_comment_content_new').bind('mouseout',f_mouseout );

// -- GESTIÓ NOU MISSATGE
Deki.$('#gc_messagetext').bind('focus', function() {
    var defval = '"..TXTS[lg].write_msg..";
    if (#gc_messagetext.val() == defval)
        { #gc_messagetext.val(''); }
    #gc_messagetext.css({'color': '#000000', 'font-style': 'normal'});
});

Deki.$('#gc_messagetext').bind('blur', function() {
    var defval = '"..TXTS[lg].write_msg..";
    if (#gc_messagetext.val() == '') {
        #gc_messagetext.css({'color': '#778899', 'font-style':
'italic'});
        #gc_messagetext.val(defval);
    }
});

Deki.$('#gc_ver_mas').bind('click', function() {
    MindTouch.Deki.Reload(#gc_comentarios, { number:
(parseInt(Deki.$(this).attr('number'))+10), ajax_action:

```

```
'reload_gestorcomentarios' }, function() { GestorComentariosControls();
});
    return false;
});

    Deki.$('#gc_ver_todos').bind('click', function() {
        MindTouch.Deki.Reload('#gc_comentarios, { number: 'all',
ajax_action: 'reload_gestorcomentarios' }, function() {
GestorComentariosControls(); });
        return false;
    });

}
GestorComentariosControls();
```

```
"</script>
```

```
// ++ CSS DEFINITION
<style type="text/css">
```

```
div.gc_comment {
    padding-left: 40px;
    font-style: italic;
    color: #778899;
}
```

```
table.gc_table {
    border: 0px !important;
    padding: 0px !important;
    margin: 0px !important;
}
```

```
div.gc_comment_content {
    font-style: italic;
    border: 1px solid white;
    color: #778899;
}
```

```
div.gc_comment_content_new {
    font-style: italic;
    border: 1px solid white;
    background-color: #bbbbbb;
    color: #778899;
}
```

```
span.gc_comment_title {
    font-style: italic;
    font-size: 10px;
    color: black;
```

```
    }  
  
    div.gc_comment_text {  
        white-space:pre-line;  
        line-height: auto;  
        margin-top: 10px;  
        padding: 4px;  
        border: 1px dashed #000;  
        background-color: #fafafa;  
        overflow: visible;  
        font-style: normal;  
        color: black;  
        width: 95%;  
    }  
    .gc_messagetext_class {  
        width: 600px;  
    }  
"</style>  
} // -- if (par.ajax_action=='load') ...  
} // -- if ((par.ajax_action=='load') ||  
(par.ajax_action=='reload_gestorcomentarios')) ..
```