

ATTENTION: In red the part of text you have to modify to make the mail sending of notifications works.

1. Install module "mod_user_live_search"

2. Add at the begin of the file `"/components/com_kunena/views/topic/view.html.php"`, after the row `"defined('_JEXEC') or die ();"` the text below

```
function after ($this, $inthat)
{
    if (!is_bool(strpos($inthat, $this)))
        return substr($inthat, strpos($inthat,$this)+strlen($this));
};

function before ($this, $inthat)
{
    return substr($inthat, 0, strpos($inthat, $this));
};

function between ($this, $that, $inthat)
{
    return before ($that, after($this, $inthat));
};
```

3. Add the table `#_kunena_messages_text_notifications` in your Joomla database, updating the file `"notifications.sql"` or making the queries below

```
CREATE TABLE `#_kunena_messages_text_notifications` (
  `mesid` int(11) NOT NULL DEFAULT '0',
  `notified` int(1) NOT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8;

ALTER TABLE `#_kunena_messages_text_notifications`
  ADD PRIMARY KEY (`mesid`);
```

4. Modify file `"/modules/mod_user_live_search/tmpl/default.php"` as below

```
<?php
defined('_JEXEC') or die;
?>
<div class="page-header">
    <span style="font-size:10pt;color:#333"><?php echo $params->get("uls_intro","This is my Intro Text"); ?></span>
</div><br>
<form role="form" method="post">
    <input type="text" class="form-control" style="height:25px;font-size:10pt" id="ulskeyword" placeholder="<?php echo $uls_placeholder; ?>" autocomplete="off">
</form>
<ul id="uls_content"></ul>
<?php
    if ($params->get("uls_jquery",0) == 1) echo '<script src="https://ajax.googleapis.com/ajax/libs/jquery/1.11.1/jquery.min.js"></script>';
?>
<script type="text/javascript">
    jQuery(document).ready(function() {
        jQuery("#ulskeyword").on('input', function() {
            var searchKeyword = jQuery(this).val();

            if (searchKeyword.length >= <?php echo $params->get("uls_trigger","3"); ?>) {
                jQuery.post('index.php?option=com_ajax&module=user_live_search&method=getUsers&format=json&Itemid=<?php echo $uls_itemid; ?>', { keywords: searchKeyword }, function(data) {
                    jQuery('ul#uls_content').empty()
                    jQuery.each(data, function() {
                        jQuery('ul#uls_content').append('<li>' + this.email + ' <a href="" data-text="' + this.title +
                    ' ">' + this.title + '</a></li>');
                    });
                    $("a[data-text]").click(function(){
                        var value = $("#kbbcode-message").val();
                        $("#kbbcode-message").val(value+$(this).attr("data-text"));
                        return false;
                    })
                }, "json");
            }
        });
    });
</script>
```

5. Modify "displayMessageContents" function in file "/components/com_kunena/views/topic/view.html.php" with the code below

```
// RENDERING MESSAGES
```

```
function displayMessageContents()
{
```

```
    $test = $this->loadTemplateFile('message');
```

```
    if (JFactory::getUser()->id) {
        $tags = explode("@", $test);
        foreach ($tags as $x => $x_value) {
```

```
            if ($x != 0) {
```

```
                $x_value = "@#@-#" . $x_value;
                $x_value = between("@#@-#", "]", $x_value);
```

```
                // GET USER ID FROM USERNAME
                $used =& JFactory::getUser($x_value)->get('id');
```

```
                // CHECK IF USER EXIST
```

```
                if ($used != 0) {
```

```
                    // SEND E-MAIL NOTIFICATION TO USER
```

```
                    $userm = JFactory::getUser($used);
```

```
                    $mailuser = $userm->email;
```

```
                    $idcat = $this->category->id;
```

```
                    $idtop = $this->topic->id;
```

```
                    $idmex = $this->message->id;
```

```
                    $db = JFactory::getDbo();
```

```
                    $query = $db->getQuery(true);
```

```
                    $query->select('notified');
```

```
                    $query->from($db->quoteName('#__kunena_messages_text_notifications'));
```

```
                    $query->where($db->quoteName('mesid')." = ".$db->quote($idmex));
```

```
                    $db->setQuery($query);
```

```
                    $results = $db->loadResult();
```

```
                    if ($results == 1) {
```

```
                    } else {
```

```
                        // SEND E-MAIL
```

```
                        $fromName = "Website mail | yourwebsite.com";
```

```
                        $fromMail = "info@yourwebsite.com";
```

```
                        $headers = "MIME-Version: 1.0\r\n";
```

```
                        $headers .= "Content-Type: text/html; charset=\r\n";
```

```
                        $headers .= "From: $fromName <$fromMail>\r\n";
```

```
                        $headers .= "Reply-To: $fromName <$fromMail>\r\n";
```

```
                        $headers .= "X-Priority: 3\r\n";
```

```
                        $headers .= "X-MSMail-Priority: Low\r\n";
```

```
                        $headers .= "X-Mailer: Website | yourwebsite.com | Forum\r\n";
```

```
                        $subject = "Hi " . $x_value . ", you were tagged in the forum.";
```

```
                        $body = "<html><body><p>Hi " . $x_value . ", you were tagged in the forum.<br><br>Click <a
```

```
href='http://'. $_SERVER['HTTP_HOST'] . $_SERVER['REQUEST_URI'] . "#". $idmex . "' target='_blank'>here</a> to see the post.</p></body></html>";
```

```
                        mail($mailuser, $subject, $body, $headers);
```

```
                    }
```

```
                // REPLACE $x_value (USER TAG) WITH PROFILE ADDRESS
```

```
                $vowels = "{@". $x_value . "}"
```

```
                $test = str_replace($vowels, '<a style="color:#ff0081 !important" href="forum/profile/'. $used . '".
```

```
$x_value . 'html">@'. $x_value . '</a>', $test);
```

```
            }
```

```
        }
```

```
    }
```

```
    echo "<span style='font-size:11pt;line-height:17px;'>" . $test . "</span>";
```

```
    // MAKE MESSAGE NOT AVAILABLE FOR NOTIFICATIONS
```

```
    // GET STATUS OF MESSAGE FROM DB
```

```
    $db = JFactory::getDbo();
```

```
    $query = $db->getQuery(true);
```

```
    $query->select('mesid');
```

```
    $query->from($db->quoteName('#__kunena_messages_text_notifications'));
```

```
    $query->where($db->quoteName('mesid')." = ".$db->quote($idmex));
```

```
    $db->setQuery($query);
```

```
    $results = $db->loadResult();
```

```
    if ($results == $idmex) {
```

```
        // MODIFY EXISTING MESSAGE TO NOTIFIED
```

```
        $db = JFactory::getDbo();
```

```
        $query = $db->getQuery(true);
```

```
        $query->update($db->quoteName('#__kunena_messages_text_notifications'))->set($db->quoteName('notified') . ' = 1')-
```

```
>where($db->quoteName('mesid') . ' = ' . $idmex);
```

```
        $db->setQuery($query);
```

```
        $result = $db->execute();
```

```
    } else {
```

```
        // INSERT MESSAGE IF NOT PRESENT
```

```
        $db = JFactory::getDbo();
```

```
        $query = $db->getQuery(true);
```

```
        $columns = array('mesid', 'notified');
```

```
        $values = array($idmex, '1');
```

```
        $query->insert($db->quoteName('#__kunena_messages_text_notifications'))->columns($db->quoteName($columns))-
```

```
>values(implode(',', $values));
```

```
        $db->setQuery($query);
```

```

        $db->execute();
    }

} else {

    $tags = explode("{@", $test);
    foreach ($tags as $x => $x_value) {

        if ($x != 0) {

            $x_value = "@#@" . $x_value;
            $x_value = between ("@" . $x_value, "");

            // GET USER ID FROM USERNAME
            $used =& JFactory::getUser($x_value)->get('id');

            // CHECK IF USER EXIST
            if ($used != 0) {
                // REMOVE "{" AND "}" FROM USER TAG
                $vowels = "{@";
                $test = str_replace($vowels, '@', $test);
                $vowels = "}";
                $test = str_replace($vowels, "", $test);
            }

        }

    }

    echo "<span style='font-size:11pt;line-height:17px;'>". $test . "</span>";

}

}

```

6. Add at the end of file `"/components/com_kunena/template/blue_eagle/html/topic/edit_editor.php"` the text below

```

<div style="position: fixed; left: 70%; top: 5%; z-index: 9999; background-color: #ddd; border: 2px dashed #333; padding: 10px;">
<?php
    $modules = JModuleHelper::getModules("tagutente");
    $document = JFactory::getDocument();
    $attribs = array();
    $attribs['style'] = 'xhtml';

    foreach ($modules as $mod)
    {
        echo JModuleHelper::renderModule($mod, $attribs);
    }
?>
</div>

```

7. Add the installed module `"mod_user_live_search"` from the backend in all pages, in the way below

The screenshot shows the Joomla! administrator interface for configuring the 'tagutente' module. The module is titled 'User Live Search' and is assigned to the 'tagutente' module. The configuration includes settings for the intro text, trigger characters, community system, email addresses, and publication options.

Module: tagutente

Assegnazione menu: tagutente

Permessi modulo: User Live Search

Avanzate: Tutte

Registered: Registered

Tutte: Tutte

102

Titolo * tagutente

Modulo: Assegnazione menu Permessi modulo Avanzate

User Live Search

Sito

User Live Search with Community Profile Integration and more.

Intro Text: TAGGA UN UTENTE

Trigger Live Search after x Characters: 1 2 3 4

Select a community system: Kunena

Who can also see Email Addresses: Scrivi o seleziona alcune opzioni

Attention: Because of Privacy and Security only Admins should be able to see User Email Addresses!

Load JQuery from GoogleAPIs: No Si

Mostra titolo: Mostra Nascondi

Posizione: tagutente

Stato: Pubblicato

Inizio pubblicazione:

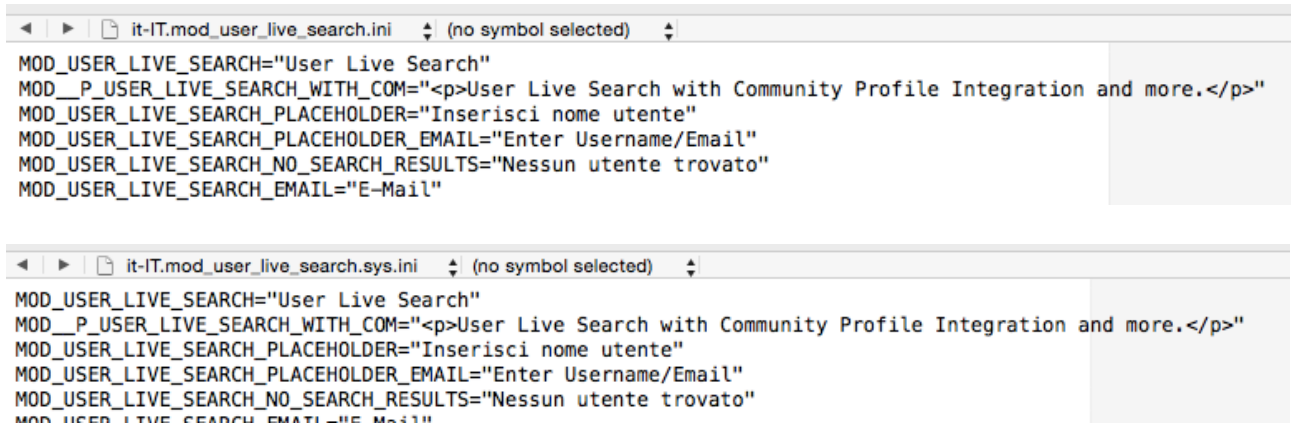
Fine pubblicazione:

Accesso: Registered

Ordinamento: 1. tagutente

Lingua: Tutte

8. To manage language in the frontend (for example italian language), I copy from the folder *"/language/en-GB/"* the files *"en-GB.mod_user_live_search.ini"* and *"en-GB.mod_user_live_search.sys.ini"* in the folder *"/language/it-IT/"*, renaming them as *"it-IT.mod_user_live_search.ini"* and *"it-IT.mod_user_live_search.sys.ini"*. After that I change the english strings we will see in the frontend inside the files translating them in italian like below



The image displays two screenshots of a code editor, likely Notepad++, showing the translation of language files. The top screenshot shows the file `it-IT.mod_user_live_search.ini` with the following content:

```
MOD_USER_LIVE_SEARCH="User Live Search"
MOD__P_USER_LIVE_SEARCH_WITH_COM="<p>User Live Search with Community Profile Integration and more.</p>"
MOD_USER_LIVE_SEARCH_PLACEHOLDER="Inserisci nome utente"
MOD_USER_LIVE_SEARCH_PLACEHOLDER_EMAIL="Enter Username/Email"
MOD_USER_LIVE_SEARCH_NO_SEARCH_RESULTS="Nessun utente trovato"
MOD_USER_LIVE_SEARCH_EMAIL="E-Mail"
```

The bottom screenshot shows the file `it-IT.mod_user_live_search.sys.ini` with the following content:

```
MOD_USER_LIVE_SEARCH="User Live Search"
MOD__P_USER_LIVE_SEARCH_WITH_COM="<p>User Live Search with Community Profile Integration and more.</p>"
MOD_USER_LIVE_SEARCH_PLACEHOLDER="Inserisci nome utente"
MOD_USER_LIVE_SEARCH_PLACEHOLDER_EMAIL="Enter Username/Email"
MOD_USER_LIVE_SEARCH_NO_SEARCH_RESULTS="Nessun utente trovato"
MOD_USER_LIVE_SEARCH_EMAIL="E-Mail"
```

FINISH