

Wordpress

META BOX

Metabox

Funkcija pridėtų meta box prie atitinkamos administravimo vietos.

```
add_meta_box(  
    string $id,  
    string $title,  
    callable $callback,  
    string|array|WP_Screen $screen = null,  
    string $context = 'advanced',  
    string $priority = 'default',  
    array $callback_args = null  
);
```

Metabox

Metabox vartojimo pavyzdys:

```
function addMetabox( $post type, $post ) {  
    add_meta_box(  
        'meta-boxas',  
        __( 'Papildomi Laukai' ),  
        'render_metabox',  
        'page',  
        'normal',  
        'default'  
    );  
}
```

Meta box pridėjimas

Meta box turėtų būti iškviečiama prie hook'o `add_meta_boxes`, pvz:

```
add_action( 'add_meta_boxes', 'addMetabox', 10, 2 );
```

Skaičiai:

10 – atitinka prioritetą

2 – nurodo kiek parametrų bus paduota f-jai `addMetabox`

Metabox atvaizdavimas

Funkcijoje `add_meta_box()` nurodome parametrą `callback` kuris nurodo kokia f-ja bus iškviesta norint parodyti `meta_box`.

Pvz.:

```
function render_my_meta_box($object){  
    echo "Musu Metabox";  
}
```

Užduotis

Pamėginkime pridėti prie postų vieną papildomą redaguojamą lauką: PapilTekstas

Sukurkime metabox'ą

Jo render f-joje įdėkime tokį atvaizdavimą:

```
<div>
    <label for="meta-box-text">Text</label>
    <input name="meta-box-text" type="text"
value="<?php echo get_post_meta($object->ID, "PapilTekstas", true); ?>">
</div>
```

Input'o išsaugojimas ir apsaugos

Tam, kad apsisaugotume nuo netinkamų postų, prie išvedimo laukų turėtume sugeneruoti ir token:

```
wp_nonce_field(basename(__FILE__), "meta-box-nonce");
```

Norint patikrinti ar token'as atitinka:

```
if (!isset($_POST["meta-box-nonce"]) ||  
!wp_verify_nonce($_POST["meta-box-nonce"], basename(__FILE__)))  
    ◦ return $post id;
```

Patikrinimas ar vartotojas gali redaguoti

Galime pasinaudoti f-ja `current_user_can`:

```
if(!current_user_can("edit_post", $post_id))  
    return $post_id;
```


Papildomo lauko išsaugojimas

Turėtume susikurti f-ją skirtą metabox išsaugojimui:

```
function save_meta_box_funkcija($post_id, $post, $update){  
    //Apsaugos  
    //Patikrinimai ar galima redaguoti  
    //Duomenų paėmimas  
    //Išsaugojimas  
}
```

Save hook'ui galime naudoti save_post

```
add_action("save_post", "save_meta_box_funkcija", 10, 3);
```

Papildomų laukų išsaugojimas

Laukams išsaugoti galėtume panaudoti tikią programą:

```
function save_meta_box_funkcija($post_id, $post, $update){  
    //Tokeno tikrinimas  
    //Patikrinimai ar galima redaguoti  
  
    $PapildTekstas_value = "";  
    if(isset($_POST["PapilTekstas"]))  
    {  
        $PapildTekstas_value = $_POST["PapilTekstas"];  
    }  
    update_post_meta($post_id,"PapilTekstas", $PapildTekstas_value);  
}
```

Meta informacijos atvaizdavima

Norint gauti ir atvaizduoti meta informaciją galime pasinaudoti šia f-ja:

```
get_post_meta(get_the_ID(), 'PapilTekstas', true);
```

Užduotis

Sukurkime papildomus laukus puslapiams kurie yra skirti įmonės informacijai atvaizduoti (metai ir aprašas).

Metabox atvaizdavimas skirtingo tipo puslapiuose

Norint patikrinti puslapio tipą:

```
global $post;

if(!empty($post))
{
    $pageTemplate = get_post_meta($post->ID, '_wp_page_template', true);
    if($pageTemplate == 'pages/simple.php' )
    {
        //Metabox atvaizdavimas
    }
}
```

Metabox texteditorius

Norint atvaizduoti TinyMCE editorių metabox'e

```
wp_editor( $photo_description, 'my_photo_desc', array(  
    'wpautop'          => true,  
    'media_buttons'    => false,  
    'textarea_name'    => 'my_photo_desc',  
    'textarea_rows'    => 10,  
    'teeny'            => true  
)  
);
```

Metabox paveikslų įterpimas

Norint jog MetaBox pagalba galėtume įkelti paveikslus, turėtume padaryti tokią seką:

Užkrauti MetaBox su inputu ir mygtuku paveikslo pasirinkimui.

Užkrautumėme reikiamus JS failus ir mūsų scriptą

Sukurtume išsaugojimo f-jas

Metabox paveikslų įkėlimui

Pirmiausia turėtume sukurti reikiamus laukus ir prikabinti prie atvaizdavimo. Pvz.:

```
function render_metabox($object) {  
    ?>  
    Paveikslų įkėlimas  
    <p>  
    <input id="image1" name="image1" type="text" value="" /><input  
    class="upload_image_button" id="_btn" type="button" value="Upload"  
    />  
    </p>  
    <?php  
    }
```


Paveikslų įkėlimas

Sukurti JS ir juos pridėti prie Admin panelės:

```
jQuery(document).ready(function() {  
  
    jQuery('.upload_image_button').click(function() {  
        var textfieldid = jQuery(this).prev().attr("id");  
  
        wp.media.editor.send.attachment = function(props, attachment) {  
            jQuery('#' + textfieldid).val(attachment.url);  
        }  
  
        wp.media.editor.open(this);  
        return false;  
    });  
});
```

Papildomų JS registravimo funkcijos

```
function paveikslu_upload_js() {  
    wp_enqueue_script('jquery');  
    wp_enqueue_script('media-upload');  
    wp_enqueue_script('thickbox');  
    wp_register_script('musu-upload',  
        trailingslashit(get_template_directory_uri()).  
        'admin_js/custom_js.js', array('jquery',  
            'media-upload', 'thickbox'));  
    wp_enqueue_script('musu-upload');  
}
```

```
function paveikslu_upload_css() {  
    wp_enqueue_style('thickbox');  
}
```

Papildomų JS registravimo funkcijos

Šias f-jas galime pridėti prie Hook'u: admin_print_scripts ir admin_print_styles

```
add_action('admin_print_scripts', 'paveikslu_media_upload_js');  
add_action('admin_print_styles', 'paveikslu_media_upload_css');
```

Sukurkime paveikslo išsaugojimo veiksmą.

Užduotis namų darbams

Faile paslaugos.html yra pateikiama paslaugų atvaizdavimo forma.

Sukurkite tokio puslapio šabloną. Administracijoje turi eiti:

Apatinį paveiksluką pakeisti iš metata box.

Pridėti failus kurie bus rodomi Taxes Ebook dalyje.

Koreguoti ir pridėti check laukelius (tam galite naudoti vieną textarea)

Koreguoti ir pridėti tekstus („**Skill building**“, „**experience level**“ ...) turi eiti:

- Pakeisti pavadinimus
- Pakeisti iconas
- Pakeisti tekstą
- Pakeisti kiekį (0, 2, 6, 8)