How to Modify FormyChat Form Fields with the “formychat_form_fields” Hook

FormyChat · https://wppool.dev/docs/how-to-modify-formychat-form-fields/

Overview

The ***formychat_form_fields*** filter hook in FormyChat allows developers to customize the default form fields. This hook is useful for adding, modifying, or removing fields dynamically, providing greater flexibility in form customization.

How It Works

The hook filters the ***$fields*** array before returning it, enabling customization of the form fields based on user needs. Developers can attach functions to this hook using ***add_filter*** to modify the structure and properties of the fields.

Default Fields

By default, the *$fields* array includes the following form fields:

Field Name	Type	Default Value	Attributes
name	text	''	placeholder, required
email	email	''	placeholder, required (Shown only if name is not empty)
phone	tel	''	placeholder, required, minlength (7), maxlength (15), min, max
message	textarea	''	placeholder, required (false)

Supported Field Types

The following field types are supported and can be used when modifying or adding new fields:

- text
- password
- email
- tel (phone number)
- textarea
- url
- number
- date
- datetime-local

Using the Hook

Developers can modify form fields using this hook by adding their custom function through add_filter.

Below are common use cases:

1. Adding a New Field

To add a new field, such as a date of birth field:

```
add_filter( 'formychat_form_fields', function( $fields ) {
    $fields[] = [
        'name' => 'dob',
        'type' => 'date',
        'default' => '',
        'help_text' => 'Enter your Date of Birth',
        'attributes' => [
            'placeholder' => __( 'Select your Date of Birth', 'social-contact-form' ),
            'required' => false,
],
    ];
    return $fields;
} );
```

2. Removing an Existing Field

If a field like phone is unnecessary, it can be removed:

```
add_filter( 'formychat_form_fields', function( $fields ) {

    return array_filter( $fields, function( $field ) {

        return $field['name'] !== 'phone';

    } );

} );
```

3. Modifying an Existing Field

To make the email field optional instead of required:

```
add_filter( 'formychat_form_fields', function( $fields ) {

    foreach ( $fields as &$field ) {

        if ( $field['name'] === 'email' ) {

            $field['attributes']['required'] = false;

        }

    }

    return $fields;

} );
```

Best Practices

- **Always validate user input** when adding new fields to ensure data security.
- **Use meaningful placeholders and help texts** to improve user experience.
- **Set appropriate attributes** (e.g., required, maxlength) to enforce data integrity.
- **Test modifications thoroughly** to prevent errors or unexpected behaviors in the form.

Conclusion

The formychat_form_fields hook is a powerful tool for customizing form fields dynamically in FormyChat. By leveraging this hook, developers can enhance user interactions, collect relevant information, and optimize form usability.