

# CRUDL FORM GENERATOR

<https://grok.com/chat/0b4d76df-e29f-45a1-bf62-edbafa0000af>

Kamu berperan sebagai Certified and Profesional Software Engineering dan Developer Aplikasi Web berpengalaman 20 tahun lebih. bantu aku menyelesaikan proyek saya, berdasarkan pdf yang telah saya upload aku akan bertanya terkait error dan penyempurnaan proyek. Memahami PHP Framework, JS Framework, CSS Framework, Full Stack, MYSQL. Buat Aplikasi CRUDL (Create Read Update Delete List) Form Builder Anda Generator using Codeigniter 4, Adminlte v4, bootstrap 5, Datatables, SweetAlert2, Toastr, Drag And Drop Forms, dan beberapa fitur berikutnya akan saya update Buat Aplikasi Berbasis Codeigniter 4 Form Builder CRUDL (Create, Read, Update, Delete, List) Builder Membuat (Create): Form Builder memungkinkan pengguna untuk membuat form dinamis berdasarkan tabel yang dipilih. Membaca (Read): Sistem menampilkan data dalam bentuk tabel, dengan filter dan pencarian. Memperbarui (Update): Form bisa dikonfigurasi untuk edit data dengan validasi input otomatis. Menghapus (Delete): Data dapat dihapus secara soft delete atau permanen. Daftar (List): Data ditampilkan dalam tabel dengan fitur pagination, filter, dan sorting. Export/Import Konfigurasi JSON: Konfigurasi CRUDL dapat disimpan dalam format JSON, diekspor, atau diimpor kembali. Form Builder Dinamis Tabel dan Field Otomatis Tabel tidak diinput manual, tapi dipilih dari daftar tabel database. Field otomatis muncul berdasarkan tabel yang dipilih. Tipe Input yang Didukung Text, Number, Email, Password, Date, Textarea, Checkbox, Radio, Toggle, Upload, Select, Daterange. Jika "Select" dipilih, pengguna bisa menghubungkan ke tabel referensi lain di database. Validasi Input Required Min/Max Length Regex Decimal Alpha, Alpha Numeric Unique (Validasi terhadap tabel) Pengaturan "Show On" untuk setiap Field Menentukan apakah field muncul di halaman Create, Read, Update, Delete, atau List. Field created\_at, updated\_at, deleted\_at otomatis disembunyikan di form Create/Edit. 3. CRUDL Generator Otomatis Generator Kode Otomatis CrudGenerator Membuat Controller, Model, dan Views secara otomatis. Menggunakan templating (ModelTemplate, ControllerTemplate, ViewsTemplate) untuk menghasilkan kode yang konsisten. File hasil generate ditempatkan di /app/Controllers/Generated, /app/Models/Generated, /app/Views/Generated/. Penggunaan CodeIgniter Service CrudGenerator dibuat sebagai service agar lebih modular dan reusable. Semua template dihasilkan secara dinamis berdasarkan konfigurasi JSON. Validasi Struktur Tabel Sebelum generate kode, sistem akan memeriksa apakah tabel memiliki created\_at, updated\_at, dan deleted\_at. Jika field tidak ada, sistem akan memberikan notifikasi dan opsi untuk memperbaiki tabel. 4. Workflow yang Ditingkatkan Form Builder → Konfigurasi Field → Simpan Konfigurasi → Generate CRUDL → Preview/Edit → Deploy Konfigurasi dapat direvisi kapan saja tanpa harus mengubah kode secara manual. CRUDL Generator bisa digunakan di CLI atau Web Interface untuk fleksibilitas lebih tinggi. Table Forms: id int title varchar description text structure json status enum table\_name varchar routes varchar validation\_rule json created\_at timestamp updated\_at timestamp deleted\_at timestamp simpan dan pahami dulu ya, tidak perlu direspon dan tunggu perintah saya selanjutnya

Baik, saya telah menyimpan dan memahami informasi proyek Anda berdasarkan deskripsi yang diberikan. Saya siap membantu Anda dengan pertanyaan terkait error, penyempurnaan proyek, atau langkah-langkah implementasi kapan pun Anda membutuhkannya. Silakan berikan perintah selanjutnya!

ini adalah referensi dari crudl form builder ini

Kode CRUDL Form Builder and Generator Versi 2.pdf

Terima kasih atas referensi yang Anda berikan! Saya telah mempelajari dokumen tersebut dan memahami struktur serta kode yang digunakan untuk membangun CRUDL Form Builder berbasis CodeIgniter 4. Berdasarkan deskripsi awal Anda dan referensi ini, saya siap membantu Anda menyelesaikan proyek, baik untuk debugging error maupun penyempurnaan fitur.

Saya melihat bahwa kode ini sudah mencakup banyak aspek dari kebutuhan Anda, seperti:

- Routing untuk Form Builder (Config/Routes.php).
- Controller untuk mengelola form (FormController.php).
- Model untuk menyimpan konfigurasi form (FormModel.php) dan submisi (FormSubmissionModel.php).
- Views untuk antarmuka pengguna (index.php, configure.php, edit.php).
- Template untuk kode yang di-generate (ModelTemplate.php, RoutesTemplate.php, dll.).
- Integrasi dengan Bootstrap 5, Select2, SweetAlert2, dan Toastr untuk UI/UX yang baik.

Namun, ada beberapa hal yang mungkin perlu diperbaiki atau disempurnakan berdasarkan kebutuhan Anda (misalnya, validasi lebih lanjut, drag-and-drop, atau fitur export/import JSON). Silakan ajukan pertanyaan spesifik terkait error yang Anda temui atau bagian yang ingin disempurnakan, agar saya bisa memberikan solusi yang tepat dan langsung bisa diterapkan.

Apa langkah selanjutnya yang Anda inginkan? Misalnya:

- Debug error tertentu dari kode yang ada?
- Menambahkan fitur baru seperti drag-and-drop?
- Menyempurnakan generator kode untuk mendukung semua tipe input yang Anda sebutkan (Text, Select, dll.)?
- Membantu implementasi fitur spesifik seperti export/import JSON?

Tunggu perintah Anda!

configure.php sama dengan form create

Saya memahami bahwa Anda ingin memastikan bahwa file configure.php (yang ada di Views/formbuilder/configure.php dari referensi Anda) berfungsi sebagai halaman untuk membuat form baru (form create) dalam konteks Form Builder Anda. Dalam hal ini, configure.php memang dirancang untuk menjadi langkah awal konfigurasi form dinamis berdasarkan tabel yang dipilih, yang pada dasarnya sesuai dengan fungsi “Create” dalam CRUDL Form Builder Anda.

Namun, berdasarkan kode yang ada di referensi dan kebutuhan Anda, saya akan membantu memastikan bahwa configure.php benar-benar berfungsi sebagai halaman pembuatan form (Create) yang sesuai dengan spesifikasi Anda, termasuk:

- Memilih tabel dari database secara otomatis.
- Menampilkan field secara dinamis berdasarkan tabel yang dipilih.
- Mendukung konfigurasi tipe input (Text, Number, Select, dll.) dan validasi.
- Menyimpan konfigurasi ke database sebagai langkah “Create”.

Mari kita analisis dan sempurnakan kode configure.php agar sesuai dengan kebutuhan “Form Create”. Berikut adalah penjelasan dan penyempurnaan berdasarkan kode yang ada:

---

## Kode Awal configure.php (Dari Referensi)

Kode awal di Views/formbuilder/configure.php adalah sebagai berikut (dengan sedikit pembersihan dari kesalahan OCR):

php

WrapCopy

```
<?php $this->extend('themes/default/layouts/main') ?> <?php $this->section('styles') ?>
<link href="https://cdnjs.cloudflare.com/ajax/libs/highlight.js/11.11.0/styles/default.min.css" rel="stylesheet">
<link href="https://cdnjs.cloudflare.com/ajax/libs/select2/4.0.9/css/select2.min.css" rel="stylesheet" integrity="sha512-nMnlpuaDPrq1E1s3IX/Q56H36qvBASwb3ipuo3MxeWbsQB18810x0cRv7UPTgB1r1qoynt35KjEwgGUeUXIPnw==" crossorigin="anonymous" referrerpolicy="no-referrer" />
```

```

<style>
    .select2-container .select2-selection--single {
        height: 38px;
    }

    .select2-container--default .select2-selection--single .select2-selection__rendered
{
    line-height: 38px;
}

    .select2-container--default .select2-selection--single .select2-selection__arrow {
        height: 38px;
    }
</style> <?php $this->endSection() ?> <?php $this->section('content') ?> <div class="c
ontainer mt-5">

    <h1>Configure Form for Table: <?= esc($table) ?> </h1>
    <div class='card row' style='margin: 10px;'>
        <div class='card-header col-md-12'>
            <h5>JSON Array</h5>
        </div>
        <div class='card-body col-md-12'>
            <code class="php"> <?= json_encode($tables) ?> </code>
        </div>
    </div> <?php if (session()->getFlashdata('error')): ?> <div class="alert alert-dange
r"> <?= session()->getFlashdata('error') ?> <?php if (isset($missingFields)): ?> <br>
    <a href="/forms/addtime/
        <?= esc($table) ?>" class="btn btn-warning mt-2">Tambahkan Kolom T
imestamp </a> <?php endif; ?>
    </div> <?php endif; ?> <?php if (session()->getFlashdata('success')): ?> <div class
="alert alert-success"> <?= session()->getFlashdata('success') ?> </div> <?php endif;
?> <form action="/forms/save" method="post">
    <input type="hidden" name="table" value="
        <?= esc($table) ?>">
    <div class="row">
        <div class="mb-3 col-md-4">
            <label>Form Title</label>
            <input type="text" name="title" class="form-control" required>

```

```

</div>
<div class="mb-3 col-md-4">
    <label>Form Description</label>
    <textarea name="description" class="form-control"></textarea>
</div>
<div class="mb-3 col-md-4">
    <label>User-Friendly URL</label>
    <input type="text" name="routes" class="form-control" placeholder="Contoh: pegawai" required>
    <small class="text-muted">URL ini akan digunakan untuk routing CRUDL (contoh: /pegawai).</small>
</div>
</div>
<!-- Field Configuration -->
<div id="fieldsContainer"> <?php foreach ($fields as $field): ?> <?php if (in_array($field->name, ['created_at', 'updated_at', 'deleted_at'])) continue; ?> <div class="mb-3 border p-3 row">
    <h5>Field: <?= esc($field->name) ?> </h5>
    <div class="mb-3 col-md-3">
        <label>Field Label</label>
        <input type="text" name="fields[
                                <?= esc($field->name) ?>][label]" class="form-control" value="
                                <?= ucfirst($field->name) ?>" required>
        <input type="hidden" name="fields[
                                <?= esc($field->name) ?>][name]" value="
                                <?= esc($field->name) ?>">
    </div>
    <div class="row fields col-md-3">
        <div class="mb-3 field-type">
            <label>Field Type</label>
            <select name="fields[
                                <?= esc($field->name) ?>][type]" class="form-control select2">
                <option value="text">Text</option>
                <option value="textarea">Textarea</option>
            </select>
        </div>
    </div>
</div>
</div>

```

```

        <option value="select">Select</option>
        <option value="checkbox">Checkbox</option>
        <option value="radio">Radio</option>
    </select>
</div>
<!-- Chained Select2 untuk Select -->
<div class="mb-3 chained-select" style="display: none;">
    <label>Select Option Source</label>
    <select name="fields[
                                                                    <?= esc($field->name) ?>]][select_source]" class="form-control select-source">
        <option value="manual">Manual</option>
        <option value="table">From Table</option>
    </select>
</div>
<!-- Manual Options -->
<div class="mb-3 manual-options" style="display: none;">
    <label>Manual Options</label>
    <textarea name="fields[
                                                                    <?= esc($field->name) ?>]][manual_options]" class="form-control" placeholder="Contoh: Male,Male\nFemale,Female">
                                                                    </textarea>
</div>
<!-- Table Options -->
<div class="mb-3 table-options" style="display: none;">
    <label>Select Table</label>
    <select name="fields[
                                                                    <?= esc($field->name) ?>]][table_reference]" class="form-control select-table">
        <option value="">Pilih Tabel</option> <?php foreach ($tables as $tbl): ?
    > <option value="
                                                                    <?= esc($tbl) ?>"> <?= esc
($tbl) ?> </option> <?php endforeach; ?>
    </select>
</div>
</div>
<div class="mb-3 col-md-3">

```

```

        <label>Show On</label>
        <br>
        <input type="checkbox" name="fields[
                                <?= esc($field->name) ?>][show
_on][]" value="create" checked> Create <input type="checkbox" name="fields[
                                <?= esc($field->name) ?>]
[show_on][]" value="edit" checked> Edit <input type="checkbox" name="fields[
                                <?= esc($field->name)
?>][show_on][]" value="list" checked> List <input type="checkbox" name="fields[
                                <?= esc($field->na
me) ?>][show_on][]" value="read"> Read <input type="checkbox" name="fields[
                                <?= esc($field
->name) ?>][show_on][]" value="delete"> Delete
        </div>
        <div class="mb-3 col-md-3">
            <label>Validation Rules</label>
            <input type="text" name="fields[
                                <?= esc($f
ield->name) ?>][validation_rule]" class="form-control" placeholder="Contoh: required|n
umeric|min_length[5]">
        </div>
        <div class="mb-3">
            <input type="checkbox" name="fields[
                                <?= es
c($field->name) ?>][required]" value="1"> Required
        </div>
    </div> <?php endforeach; ?> </div>
    <button type="submit" class="btn btn-primary">Save Form</button>
</form>
</div> <?php $this->endSection() ?> <?php $this->section('scripts') ?> <script src="ht
tps://cdnjs.cloudflare.com/ajax/libs/highlight.js/11.11.0/highlight.min.js"></script>
<script src="https://cdnjs.cloudflare.com/ajax/libs/select2/4.0.9/js/select2.min.js" i
ntegrity="sha512-9p/L4acAjbJ1aaGXmZfQQ2bV42Het1CLbv8EP0z3rLbQED2TAFUIDvAezY7kumYqg5T8j
HtDd1m1fgIsr5QzKg==" crossorigin="anonymous" referrerpolicy="no-referrer"></script>
<script>
    hljs.highlightAll();
    $(document).ready(function() {

```

```

< ? php
    if (session() - > getFlashdata('success')): ? > toastr.success(' < ? = session
() - > getFlashdata('success') ? > '); < ? php endif; ? > < ? php
        if (session() - > getFlashdata('error')): ? > toastr.error(' < ? = session
() - > getFlashdata('error') ? > '); < ? php endif; ? > $(''.select2').select2(); $(''.
select-table').select2(); $('select[name*="[type]"').change(function() {
    const fieldType = $(this).val();
    const chainedSelect = $(this).closest('.field-type').next('.chained-
select');

    if (fieldType === 'select') {
        chainedSelect.show();
    } else {
        chainedSelect.hide();
    }
}); $(''.select-source').change(function() {
    const selectSource = $(this).val();
    const manualOptions = $(this).closest('.chained-select').next('.manu
al-options');

    const tableOptions = $(this).closest('.chained-select').nextAll('.ta
ble-options').first();

    if (selectSource === 'manual') {
        manualOptions.show();
        tableOptions.hide();
    } else if (selectSource === 'table') {
        manualOptions.hide();
        tableOptions.show();
    } else {
        manualOptions.hide();
        tableOptions.hide();
    }
}); $(''.select-table').change(function() {
    const table = $(this).val();
    if (table) {
        $.ajax({
            url: '/forms/getfields',
            method: 'POST',
            data: {

```



```
        table: table
    },
    success: function(response) {
        try {
            const fields = JSON.parse(response);
            if (fields.error) {
                alert(fields.error);
                return;
            } // Logika untuk menampilkan field dari tabel referensi bisa ditambahkan di sini
        } catch (error) { console.error('Error parsing JSON:', error); alert('Gagal memproses data dari server.')}
    }, error: function() { alert('Gagal mengambil field dari tabel.')}
});
```

</script> <?php \$this->endSection() ?>

---

## Analisis dan Penyempurnaan

Kode di atas sudah cukup baik untuk fungsi “Create” dalam Form Builder, tetapi ada beberapa kekurangan atau area yang bisa disempurnakan agar sesuai dengan spesifikasi Anda:

### 1. Dukungan Semua Tipe Input yang Disebutkan

Dalam spesifikasi Anda, tipe input yang didukung meliputi: Text, Number, Email, Password, Date, Textarea, Checkbox, Radio, Toggle, Upload, Select, Daterange. Namun, di kode saat ini, hanya ada Text, Textarea, Select, Checkbox, dan Radio. Kita perlu menambahkan opsi lainnya.

### 2. Validasi Input Otomatis

Validasi seperti Required, Min/Max Length, Regex, dll., sudah ada, tetapi perlu diperluas untuk mendukung semua tipe input (misalnya, validasi khusus untuk Email atau Number).

### 3. Drag-and-Drop

Fitur drag-and-drop belum ada di kode ini. Kita bisa menambahkan library seperti jQuery UI atau SortableJS untuk mendukungnya.

### 4. Tabel Referensi untuk Select

Logika untuk mengambil field dari tabel referensi (saat “Select” dipilih) belum lengkap di bagian JavaScript. Kita perlu menyempurnakan AJAX call ke `getTableFields`.

### 5. Struktur JSON

Data yang disimpan ke structure di tabel forms harus mencakup semua konfigurasi field (label, type, show\_on, validation\_rule, dll.) dalam format JSON yang rapi.

---

# Penyempurnaan Kode configure.php

Berikut adalah versi yang disempurnakan untuk configure.php agar benar-benar berfungsi sebagai “Form Create” sesuai kebutuhan Anda:

php

WrapCopy

```
<?php $this->extend('themes/default/layouts/main') ?> <?php $this->section('styles') ?>
<link href="https://cdnjs.cloudflare.com/ajax/libs/highlight.js/11.11.0/styles/default.min.css" rel="stylesheet">
<link href="https://cdnjs.cloudflare.com/ajax/libs/select2/4.0.9/css/select2.min.css" rel="stylesheet">
<link href="https://cdnjs.cloudflare.com/ajax/libs/Sortable/1.15.0/Sortable.min.css" rel="stylesheet">
<!-- Untuk drag-and-drop -->
<style>
    .select2-container .select2-selection--single {
        height: 38px;
    }

    .select2-container--default .select2-selection--single .select2-selection__rendered {
        line-height: 38px;
    }

    .select2-container--default .select2-selection--single .select2-selection__arrow {
        height: 38px;
    }

    .field-config {
        cursor: move;
    }

    /* Styling untuk drag-and-drop */
```

```

</style> <?php $this->endSection() ?> <?php $this->section('content') ?> <div class="c
ontainer mt-5">
    <h1>Create Form for Table: <?= esc($table) ?> </h1> <?php if (session()->getFlashdat
a('error')): ?> <div class="alert alert-danger"> <?= session()->getFlashdata('error')
?> <?php if (isset($missingFields)): ?> <br>
        <a href="/forms/addtime/
                <?= esc($table) ?>" class="btn btn-warning mt-2">Add Times
tamp Columns </a> <?php endif; ?>
    </div> <?php endif; ?> <?php if (session()->getFlashdata('success')): ?> <div class
="alert alert-success"> <?= session()->getFlashdata('success') ?> </div> <?php endif;
?> <form action="/forms/save" method="post" id="formBuilder"> <?= csrf_field() ?> <inp
ut type="hidden" name="table" value="
        <?= esc($table) ?>">
    <div class="row">
        <div class="mb-3 col-md-4">
            <label>Form Title</label>
            <input type="text" name="title" class="form-control" required>
        </div>
        <div class="mb-3 col-md-4">
            <label>Form Description</label>
            <textarea name="description" class="form-control"></textarea>
        </div>
        <div class="mb-3 col-md-4">
            <label>User-Friendly URL</label>
            <input type="text" name="routes" class="form-control" placeholder="e.g., emplo
yees" required>
            <small class="text-muted">This URL will be used for CRUDL routing (e.g., /empl
oyees).</small>
        </div>
    </div>
    <!-- Field Configuration with Drag-and-Drop -->
    <div id="fieldsContainer" class="sortable"> <?php foreach ($fields as $field): ?>
    <?php if (in_array($field->name, ['created_at', 'updated_at', 'deleted_at'])) continu
e; ?> <div class="mb-3 border p-3 row field-config">
        <h5>Field: <?= esc($field->name) ?> </h5>
        <div class="mb-3 col-md-3">
            <label>Field Label</label>

```

```

        <input type="text" name="fields[
                                <?= esc($field->name) ?>][label]"
class="form-control" value="
                                <?= ucfirst($field->name) ?>" requ
ired>
        <input type="hidden" name="fields[
                                <?= esc($field->name) ?>][nam
e]" value="
                                <?= esc($field->name) ?>">
    </div>
    <div class="mb-3 col-md-3">
        <label>Field Type</label>
        <select name="fields[
                                <?= esc($field->name) ?>]
[type]" class="form-control select2 field-type">
            <option value="text">Text</option>
            <option value="number">Number</option>
            <option value="email">Email</option>
            <option value="password">Password</option>
            <option value="date">Date</option>
            <option value="textarea">Textarea</option>
            <option value="checkbox">Checkbox</option>
            <option value="radio">Radio</option>
            <option value="toggle">Toggle</option>
            <option value="upload">Upload</option>
            <option value="select">Select</option>
            <option value="daterange">Daterange</option>
        </select>
    </div>
    <!-- Dynamic Options for Select -->
    <div class="mb-3 chained-select" style="display: none;">
        <label>Select Option Source</label>
        <select name="fields[
                                <?= esc($field->name) ?>]
[select_source]" class="form-control select-source">
            <option value="manual">Manual</option>
            <option value="table">From Table</option>

```



```

me) ?>][show_on][]" value="edit" checked> Edit <input type="checkbox" name="fields[
                                <?= esc($field
->name) ?>][show_on][]" value="list" checked> List <input type="checkbox" name="fields
[
                                <?= esc($f
ield->name) ?>][show_on][]" value="read"> Read <input type="checkbox" name="fields[
                                <?= es
c($field->name) ?>][show_on][]" value="delete"> Delete
    </div>
    <div class="mb-3 col-md-3">
        <label>Validation Rules</label>
        <input type="text" name="fields[
                                <?
= esc($field->name) ?>][validation_rule]" class="form-control" placeholder="e.g., requ
ired|numeric|min_length[5]">
        <input type="checkbox" name="fields[
                                <?
= esc($field->name) ?>][required]" value="1"> Required
    </div>
</div> <?php endforeach; ?> </div>
    <button type="submit" class="btn btn-primary">Save Form</button>
</form>
</div> <?php $this->endSection() ?> <?php $this->section('scripts') ?> <script src="ht
tps://cdnjs.cloudflare.com/ajax/libs/highlight.js/11.11.0/highlight.min.js"></script>
<script src="https://cdnjs.cloudflare.com/ajax/libs/select2/4.0.9/js/select2.min.js">
</script>
<script src="https://cdnjs.cloudflare.com/ajax/libs/Sortable/1.15.0/Sortable.min.js">
</script>
<!-- Untuk drag-and-drop -->
<script>
    hljs.highlightAll();
    $(document).ready(function() { // Flash messages
        < ? php
            if (session() - > getFlashdata('success')): ? > toastr.success(' < ? = session
() - > getFlashdata('success') ? > '); < ? php endif; ? > < ? php
            if (session() - > getFlashdata('error')): ? > toastr.error(' < ? = session
() - > getFlashdata('error') ? > '); < ? php endif; ? > // Initialize Select2 $('<sel

```

```

ect2').select2(); $('.select-table').select2(); $('.value-field').select2(); $('.label-
-field').select2(); // Drag-and-Drop with SortableJS const fieldsContainer = document.
getElementById('fieldsContainer'); Sortable.create(fieldsContainer, { animation: 150,
handle: '.field-config', onEnd: function(evt) { console.log('Field order changed:', ev
t.oldIndex, 'to', evt.newIndex); } }); // Field type change handler $('.field-type').c
hange(function() { const fieldType = $(this).val(); const chainedSelect = $(this).clos
est('.field-config').find('.chained-select'); const manualOptions = $(this).closest('.
field-config').find('.manual-options'); const tableOptions = $(this).closest('.field-c
onfig').find('.table-options'); if (fieldType === 'select') { chainedSelect.show(); ma
nualOptions.hide(); tableOptions.hide(); } else { chainedSelect.hide(); manualOptions.
hide(); tableOptions.hide(); } }); // Select source change handler $('.select-sourc
e').change(function() { const selectSource = $(this).val(); const manualOptions = $(th
is).closest('.field-config').find('.manual-options'); const tableOptions = $(this).clo
sest('.field-config').find('.table-options'); if (selectSource === 'manual') { manualO
ptions.show(); tableOptions.hide(); } else if (selectSource === 'table') { manualOptio
ns.hide(); tableOptions.show(); } else { manualOptions.hide(); tableOptions.hide(); }
}); // Fetch fields for table reference $('.select-table').change(function() { const t
able = $(this).val(); const valueField = $(this).closest('.table-options').find('.valu
e-field'); const labelField = $(this).closest('.table-options').find('.label-field');
if (table) { $.ajax({ url: '/forms/getfields', method: 'POST', data: { table: table },
success: function(response) { const fields = typeof response === 'string' ? JSON.parse
(response) : response; if (fields.error) { toastr.error(fields.error); return; } let v
alueOptions = '
        < option value = "" > Select Value Field < /option>'; let labelOptions
= ' < option value = "" > Select Label Field < /option>'; fields.forEach(field => { va
lueOptions += ` < option value = "${field.name}" > $ {
            field.name
        } < /option>`; labelOptions += ` < option value = "${field.name}" > $
{
            field.name
        } < /option>`; }); valueField.html(valueOptions).prop('disabled', fals
e); labelField.html(labelOptions).prop('disabled', false); }, error: function() { toas
tr.error('Failed to fetch fields from table.');
```

---

## Penjelasan Penyempurnaan

### 1. Semua Tipe Input Ditambahkan

Opsi seperti Number, Email, Password, Date, Toggle, Upload, dan Daterange telah ditambahkan ke dropdown Field Type.

### 2. Drag-and-Drop

Library SortableJS ditambahkan untuk mendukung drag-and-drop pada field. Setiap blok field (.field-config) bisa di-drag untuk mengubah urutan.

### 3. Tabel Referensi untuk Select

Logika AJAX untuk select-table disempurnakan untuk mengisi dropdown value\_field dan label\_field berdasarkan field dari tabel referensi yang dipilih.

### 4. Validasi Input

Input validation\_rule mendukung aturan CodeIgniter (misalnya, required|numeric|min\_length[5]), dan checkbox required tetap ada untuk kemudahan.

### 5. CSRF Protection

Ditambahkan `<?= csrf_field() ?>` untuk keamanan form submission di CodeIgniter 4.

### 6. UI/UX Lebih Baik

Menggunakan Toastr untuk notifikasi dan Select2 untuk dropdown yang lebih interaktif.

---

## Cara Kerja

1. Pengguna memilih tabel dari halaman index.php.
  2. Sistem mengarahkan ke `/forms/configure/{table}` yang menampilkan configure.php.
  3. Pengguna mengisi judul, deskripsi, URL, dan mengonfigurasi field (label, tipe, validasi, dll.).
  4. Setelah submit, data disimpan ke tabel forms melalui `FormController::save`.
- 

## Langkah Selanjutnya

- Jika Anda ingin menambahkan fitur lain (misalnya, preview langsung di halaman ini), beri tahu saya.
- Jika ada error saat menjalankan kode ini, laporkan pesan errornya agar saya bisa debug.
- Jika Anda ingin configure.php digabungkan dengan fungsi lain (misalnya, edit), kita bisa sesuaikan lagi.

Apa yang ingin Anda lakukan dengan configure.php berikutnya?