

source: [full-customer](#) / [tags / 3.1.25](#) / [app / controller / elementor-crm / Hooks.php](#)

Last change on this file was 3175784, checked in by fullservices, 6 months ago

version 3.1.25

Property svn:eol-style set to `native`

File size: 7.7 KB

Line	
1	<?php
2	
3	namespace Full\Customer\ElementorCrm;
4	
5	use ElementorPro\Modules\Forms\Submissions\Database\Repositories\Form_Snapshot_Repository;
6	
7	class Hooks
8	{
9	private Settings \$env;
10	
11	private function __construct()
12	{
13	\$this->env = new Settings();
14	}
15	
16	public static function attach(): void
17	{
18	\$cls = new self();
19	
20	add_filter('full-customer/active-widgets-menu', [\$cls, 'addMenuPage']);
21	add_action('admin_enqueue_scripts', [\$cls, 'adminEnqueueScripts']);
22	add_action('elementor/frontend/before_render', [\$cls, 'countFormView']);
23	
24	add_action('wp_ajax_full/widget/crm/form/set-stages', [\$cls, 'processStageUpdate']);
25	add_action('wp_ajax_full/widget/crm/form/get-analytics', [\$cls, 'formAnalytics']);
26	add_action('wp_ajax_full/widget/crm/form/get-leads', [\$cls, 'getLeads']);
27	add_action('wp_ajax_full/widget/crm/form/get-fields', [\$cls, 'getFormFields']);
28	
29	add_action('wp_ajax_full/widget/crm/lead/update', [\$cls, 'updateLead']);
30	add_action('wp_ajax_full/widget/crm/lead/delete', [\$cls, 'deleteLead']);
31	add_action('wp_ajax_full/widget/crm/lead/hide', [\$cls, 'hideLead']);
32	}
33	
34	public function countFormView(\$element): void
35	{
36	if (!\$element instanceof \ElementorPro\Modules\Forms\Widgets\Form) :
37	return;
38	endif;
39	
40	global \$fullCrmFormsTracked;
41	
42	if (!isset(\$fullCrmFormsTracked)) :
43	\$fullCrmFormsTracked = [];
44	endif;
45	
46	\$key = 'full/crm/' . \$element->get_id();
47	

```

Line
48     if (current_user_can('manage_options') || in_array($key, $fullCrmFormsTracked)) :
49         return;
50     endif;
51
52     $views = get_option($key, 0);
53     update_option($key, ++$views);
54
55     $fullCrmFormsTracked[] = $key;
56 }
57
58 public function addMenuPage(array $menu): array
59 {
60     $menu[] = [
61         'name' => 'FULL.elementor crm',
62         'endpoint' => 'full-crm'
63     ];
64
65     return $menu;
66 }
67
68 public function adminEnqueueScripts(): void
69 {
70     if (filter_input(INPUT_GET, 'page') !== 'full-crm') :
71         return;
72     endif;
73
74     $version = getFullAssetsVersion();
75     $baseUrl = trailingslashit(plugin_dir_url(FULL_CUSTOMER_FILE)) . 'app/assets/';
76
77     wp_enqueue_style('full-admin-crm', $baseUrl . 'css/crm.css', [], $version);
78     wp_enqueue_script('full-admin-crm', $baseUrl . 'js/admin-crm.js', ['jquery', 'jquery-ui-sortable', 'jquery-
ui-draggable'], $version, true);
79     wp_localize_script('full-admin-crm', 'fullCrm', [
80         'stages' => $this->env->get('stages'),
81         'fragments' => $this->env->get('fragments'),
82         'leadBaseUrl' => admin_url('admin.php?page=e-form-submissions#/')
83     ]);
84 }
85
86 public function formAnalytics(): void
87 {
88     $formId = filter_input(INPUT_POST, 'formId');
89     $stages = isset($this->env->get('stages')[$formId]) ? $this->env->get('stages')[$formId] : [];
90
91     $response = [
92         'values' => [
93             'total_leads' => 0,
94             'total_won' => 0,
95             'total_lost' => 0,
96             'capture_rate' => 0,
97             'conversion_rate' => 0,
98             'total_views' => 0
99         ],
100         'chart' => []
101     ];
102
103     if (!$formId || !$stages) :
104         wp_send_json($response);
105     endif;
106
107     $response['chart'] = array_fill_keys(array_keys($stages), 0);
108
109     $worker = new Leads();

```

```

Line
110 $funnel = $worker->leadsByStage($formId);
111 $firstStage = array_key_first($stages);
112
113 $response['values']['total_views'] = get_option('full/crm/' . $formId, 0);
114
115 foreach ($funnel as $index => $value) :
116     $key = array_key_exists($index, $stages) ? $index : $firstStage;
117     $stage = $stages[$key];
118
119     $response['values']['total_leads'] += $value;
120
121     if ('won' === $stage['status']) :
122         $response['values']['total_won'] += $value;
123     elseif ('lost' === $stage['status']) :
124         $response['values']['total_lost'] += $value;
125     else :
126         $response['chart'][$key] += $value;
127     endif;
128 endforeach;
129
130 foreach ($response['chart'] as $index => $value) :
131     $key = array_key_exists($index, $stages) ? $index : $firstStage;
132     $stage = $stages[$key];
133
134     if ($stage['status'] !== '') :
135         unset($response['chart'][$index]);
136     endif;
137 endforeach;
138
139 if ($response['values']['total_leads']) :
140     $response['values']['conversion_rate'] = number_format_i18n($response['values']
['total_won'] / $response['values']['total_leads'] * 100, 2) . '%';
141 endif;
142
143 if ($response['values']['total_views']) :
144     $response['values']['capture_rate'] = number_format_i18n($response['values']
['total_leads'] / $response['values']['total_views'] * 100, 2) . '%';
145 endif;
146
147 wp_send_json($response);
148 }
149
150 public function processStageUpdate(): void
151 {
152     check_ajax_referer('full/widget/crm/form/set-stages');
153
154     $stages = $this->env->get('stages');
155     $fragments = $this->env->get('fragments');
156
157     $formId = filter_input(INPUT_POST, 'formId');
158     $formStages = filter_input(INPUT_POST, 'stage', FILTER_DEFAULT, FILTER_REQUIRE_ARRAY) ?? [];
159     $formFragments = filter_input(INPUT_POST, 'fragments', FILTER_DEFAULT, FILTER_REQUIRE_ARRAY) ?? [];
160
161     if (!is_array($stages)) :
162         $stages = [];
163     endif;
164
165     if (!is_array($fragments)) :
166         $fragments = [];
167     endif;
168
169     $stages[$formId] = array_filter($formStages, fn($row) => isset($row['name']) && $row['name']);
170     $fragments[$formId] = array_filter(array_map('sanitize_text_field', $formFragments));

```

```

Line
171
172     $this->env->set('stages', $stages);
173     $this->env->set('fragments', $fragments);
174
175     wp_send_json_success([
176         'stages' => $this->env->get('stages'),
177         'fragments' => $this->env->get('fragments')
178     ]);
179 }
180
181 public function getFormFields(): void
182 {
183     global $wpdb;
184
185     $formId = sanitize_text_field(filter_input(INPUT_POST, 'formId'));
186
187     $sql = " SELECT meta_value FROM {$wpdb->postmeta} ";
188     $sql .= " WHERE `meta_key` = '" . Form_Snapshot_Repository::POST_META_KEY . "'";
189     $sql .= " AND meta_value LIKE '%" . id . "':\"$formId\"%'";
190
191     $data = $wpdb->get_col($sql);
192     $data = is_array($data) ? array_map('json_decode', $data) : [];
193
194     $fields = [];
195
196     foreach ($data as $snapshots) :
197         foreach ($snapshots as $form) :
198             foreach ($form->fields as $field) :
199                 $fields[$field->id] = $field->label;
200             endforeach;
201         endforeach;
202     endforeach;
203
204     wp_send_json($fields);
205 }
206
207 public function getLeads(): void
208 {
209     $worker = new Leads();
210     $formId = filter_input(INPUT_POST, 'formId');
211     $stages = isset($this->env->get('stages')[$formId]) ? $this->env->get('stages')[$formId] : [];
212
213     if (!$formId || !$stages) :
214         wp_send_json([]);
215     endif;
216
217     $list = $worker->list($formId);
218
219     $formatted = [];
220     $firstStage = array_key_first($stages);
221
222     foreach ($list['data'] as $item) :
223         $item['labels'] = [];
224
225         if (!array_key_exists($item['status'], $stages)) :
226             $item['status'] = $firstStage;
227         endif;
228
229         if (!isset($formatted[$item['status']])) :
230             $formatted[$item['status']] = [];
231         endif;
232
233         foreach ($item['form']['fields'] as $field) :

```

Line	
234	<code>\$item['labels'][\$field['id']] = \$field['label'];</code>
235	<code>endforeach;</code>
236	
237	<code>\$formatted[\$item['status']][] = \$item;</code>
238	<code>endforeach;</code>
239	
240	<code>wp_send_json(\$formatted);</code>
241	<code>}</code>
242	
243	<code>public function updateLead(): void</code>
244	<code>{</code>
245	<code> \$submissionId = filter_input(INPUT_POST, 'id', FILTER_VALIDATE_INT) ?? 0;</code>
246	<code> \$stage = sanitize_title(filter_input(INPUT_POST, 'stage') ?? '');</code>
247	
248	<code> \$worker = new Leads();</code>
249	
250	<code> \$worker->updateStage(\$submissionId, \$stage);</code>
251	
252	<code> wp_send_json_success();</code>
253	<code>}</code>
254	
255	<code>public function deleteLead(): void</code>
256	<code>{</code>
257	<code> \$submissionId = filter_input(INPUT_POST, 'id', FILTER_VALIDATE_INT) ?? 0;</code>
258	
259	<code> \$worker = new Leads();</code>
260	
261	<code> \$worker->delete(\$submissionId);</code>
262	
263	<code> wp_send_json_success();</code>
264	<code>}</code>
265	
266	<code>public function hideLead(): void</code>
267	<code>{</code>
268	<code> \$submissionId = filter_input(INPUT_POST, 'id', FILTER_VALIDATE_INT) ?? 0;</code>
269	
270	<code> \$worker = new Leads();</code>
271	
272	<code> \$worker->hide(\$submissionId);</code>
273	
274	<code> wp_send_json_success();</code>
275	<code>}</code>
276	<code>}</code>
277	
278	<code>Hooks::attach();</code>

About

News

Hosting

Donate

Swag

[Documentation](#)

[Developers](#)

[Get Involved](#)

[Learn](#)

[Showcase](#)

[Plugins](#)

[Themes](#)

[Patterns](#)

[WordCamp](#)

[WordPress.TV](#)

[BuddyPress](#)

[bbPress](#)

[WordPress.com](#)

[Matt](#)

[Privacy](#)

[Public Code](#)