

Intern: Neues Kundencenter

Kundencenter Relaunch mit Front- und Backend



Kundencenter

Bitte geben Sie Ihre Zugangsdaten ein, um sich für das Kundencenter einzuloggen.

 Login und Passwort auf diesem PC speichern

[Passwort vergessen?](#)

Fabian Rohrka ■ 10.04.2017

- Datenkapselung
- Verbindung zur Datenbank
- Komplette Logik
- Holen und Speichern von Informationen
- `https://domain.tld/API`



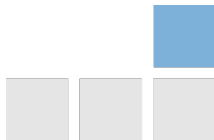
- `domain.tld/API/CONTROLLER/METHOD`
- *CONTROLLER* ist die aufgerufene Klasse des Namespaces `controllers`
- *METHOD*`Action()` ist die Methode der Klasse *CONTROLLER*
- Öffentliche Methoden haben den Zusatz `Action` am Ende des Namens
- Alle Aufrufe per `POST`
- Benötigte Parameter: `authToken` sowie Methoden spezifische Parameter
- `authToken` ist die `SessionId` des Backends



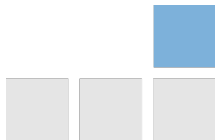
Beispiel

- URL: `domain.tld/API/ftp/get`
- Parameter: `authToken`, `id = 422`
- Backend intern: `\controllers\Ftp::getAction()`

```
1  {
2      "controller": "Ftp",
3      "action": "getAction",
4      "data": {
5          "sys_pk": "422",
6          "fk_contract": "4001",
7          "name": "vallendor",
8          "description": "Dominik Vallendor",
9          "path": ""
10     },
11     "success": true
12 }
```



- Visualisierung von Daten
- Schnittstelle für Kunden und Administratoren





- Models und Controller liegen im Ordner */frontend*
- Controller sorgen für den Austausch mit dem Backend
- Controller liefern meisten ein Model zurück
- Model stellt meist einen Eintrag einer Datenbanktabelle da
- `\model\ApiCaller` setzt letztendlich den API-Call ab und liefert den `data` Wert der Antwort zurück.
- Antworten, die nicht im JSON-Format sind, weil es einen Fehler gab, der nicht abgefangen wurde, werden in die Datei `/log/log.txt` geschrieben

Beispiel FTP-Benutzer holen

```
1 <?php
2
3 namespace controllers;
4
5
6 class FTP {
7
8     private $_apicaller;
9
10    public function __construct () {
11        global $session, $apicaller;
12        $this->_session = $session;
13        $this->_apicaller = $apicaller;
14    }
15
16    public function get($id) {
17        $ftpAccount = null;
18        $ftp = $this->_apicaller->sendRequest( requestUrl: "/ftp/get", array(
19            'authToken' => $this->_session->token,
20            'id' => $id
21        ));
22        if ($ftp != null) {
23            $ftpAccount = new \models\FTP($ftp->sys_pk, $ftp->name, password: "", $ftp->description, $ftp->path);
24        }
25
26        return $ftpAccount;
27    }
28 }
```

- Erweiterung von `$_SESSION`
- Speichert Benutzer-ID
- Rendert HTML Header, Navigation und Footer
- Stellt Funktionen für Fehler- und Erfolgsmeldungen zur Verfügung