

FTP-Modul

„Entwicklung einer RESTful API zur serverübergreifenden Verwaltung von FTP-Zugängen, neu schreiben des Moduls und Integration des selbigen in das neue firmeninterne Framework“

Dennis Rempfer





Agenda

Vorstellung des Unternehmens
Analyse
Durchführung
Implementierung
Wirtschaftlichkeit
Fazit



- Vorstellung des Unternehmens -
Willkommen bei der Provitex GmbH

Vorstellung des Unternehmens

- Full Service Internetagentur
- Sitz in Reutlingen
- 2 Geschäftsführer
- 2 Softwareentwickler
- 1 Grafiker
- 1 Marketer
- 3 Azubis
- 1 Kaffeemaschine

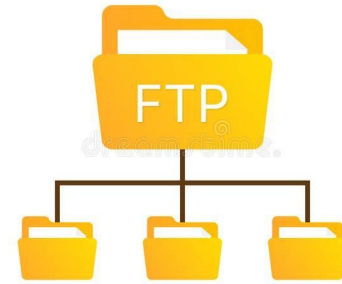




- Analyse -

Wieso, weshalb, warum.

- Verwalten von FTP-Zugängen
 - a) Auch für Reseller
- Anbieten einer Schnittstelle
 - a) Zugänge über RESTful API anlegen/bearbeiten/löschen





Analyse - Was?

- File Transfer Protocol(FTP)
 - Übertragung von Dateien
- Reseller
 - Wiederverkäufer von Provitex „White Label Produkten“
- Framework
 - → Programmgerüst



Analyse – Was ist eMantix?

- Administrations-Software
- Dient zur Verwaltung von:
 - i. Kunden
 - ii. Resellern
 - iii. Geschäftspartnern
 - iv. Infrastruktur
 - v. Mitarbeiter



Analyse – Wieso?

- Veraltete PHP-Version
- Nicht mehr kompatibel zum neuen firmeneigenen Framework
- Veraltete Sicherheitsmechanismen und Verschlüsselung der Passwörter

The screenshot shows a web interface titled "FTP Account bearbeiten". It contains several input fields for account details. The "Account Daten:" section includes fields for "Kundennummer" (10494), "Username" (ftp_10494_01), "Homedir" (/var/www/10494/), and "Zugang Backup Server" (ftp). The "Erweiterte Informationen:" section includes "Backup Server IPv4" (127.0.0.1), "Anzahl Logins" (500), "Erstellt:" (01.12.2020 - 09:43:27 Uhr), and "Letzter Zugriff:" (18.01.2021 - 09:25:30 Uhr).

The screenshot shows a newer web interface titled "Account bearbeiten". It has a navigation bar with "Übersicht", "Löschen", and "Speichern" buttons. The form includes fields for "Kundennummer" (10494), "Username" (ftp_10494_1), "Neues Passwort:" (with a "wiederholen:" field), "Homedir" (/var/www), and "Zugang Backup Server" (checkbox). It also shows "Anzahl Logins" (13672), "Erstellt:" (23.07.2010 11:07:15), and "Letzter Zugriff:" (22.10.2020 16:10:27). A note at the bottom states: "Alle mit Sternchen * gekennzeichneten Felder sind Pflichtfelder."

FTP Account bearbeiten



Account-Daten:

Kundennummer:	10494
Username:	ftp_10494_20
Homedir:	/var/www/10494/
Bemerkung:	-
Zugang Backup Server:	nein ▼

Erweiterte Informationen:

Backup Server IPv4:	127.0.0.1
Anzahl Logins:	907
Erstellt:	01.12.2020 - 09:43:27 Uhr
Letzer Zugriff:	18.01.2021 - 09:20:30 Uhr

 Übersicht

Löschen

Speichern

Account bearbeiten

Kundennummer:

10494

Username:*

ftp_10494_1

Neues Passwort:Neues Passwort
wiederholen:

Homedir:*

/var/www

Bemerkung:

Zugang Backup-Server:

☐

Anzahl Logins:

13672

Erstellt:

23.07.2010 11:07:15

Letzter Zugriff:

22.10.2020 16:10:27

Alle mit Sternchen * gekennzeichneten Felder sind Pflichtfelder.

Analyse – Wieso?

- Veraltete PHP-Version
- Nicht mehr kompatibel zum neuen firmeneigenen Framework
- Veraltete Sicherheitsmechanismen und Verschlüsselung der Passwörter

The screenshot shows a web interface titled "FTP Account bearbeiten". It contains several input fields for account details. The "Account Daten:" section includes fields for "Kundennummer" (10494), "Username" (ftp_10494_01), "Homedir" (/var/www/10494/), and "Zugang Backup Server" (ftp). The "Erweiterte Informationen:" section includes "Backup Server IPv4" (127.0.0.1), "Anzahl Logins" (500), "Erstellt:" (01.12.2020 - 09:43:27 Uhr), and "Letzter Zugriff:" (18.01.2021 - 09:25:30 Uhr).

The screenshot shows a modern web interface titled "Account bearbeiten". It has a navigation bar with "Übersicht", "Löschen", and "Speichern" buttons. The form includes fields for "Kundennummer" (10494), "Username" (ftp_10494_1), "Neues Passwort:" (with a "Neues Passwort wiederholen:" field), "Homedir" (/var/www), and "Bemerkung:". There is a checkbox for "Zugang Backup Server:". The "Erstellt:" field shows "23.07.2010 11:07:15" and "Letzter Zugriff:" shows "22.10.2020 16:10:27". A note at the bottom states: "Alle mit Sternchen * gekennzeichneten Felder sind Pflichtfelder."

Analyse – Wie viel?



SOFTWARE



HARDWARE



PERSONAL



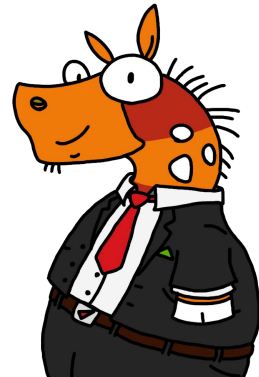
Analyse – Wie viel?

Vorgang	Anz. Mitarbeiter	Dauer	Kosten	Gesamt
Analyse	2	3 Stunden	$3 \times 25\text{€}$ + $3 \times 50\text{€}$	225€
Entwurf	2	3 Stunden	$3 \times 25\text{€}$ + $3 \times 50\text{€}$	225€
Implementa-tion	1	50 Stunden	$50 \times 25\text{€}$	1250€
Qualitäts-sicherung	1	6 Stunden	$6 \times 25\text{€}$	150€
				1850€



Azubi = 25€
Mitarbeiter = 50€

Interne
Verrechnungssät-
ze



Analyse – Wie viel?



Gesamtkosten:
1850,-€



- Durchführung -
Schaffe, schaffe, Häusle baua.

Durchführung – Anforderungen und Rahmenbedingungen?



- Maximal 70 Stunden
- Soll/Ist-Vergleich mit Zielsetzung muss erkennbar sein
- Wirtschaftlichkeit muss berücksichtigt werden
- Reales Projekt



- Im Unternehmen etablierte Technologien
- Wirtschaftlichkeit/Nutzen
- Programming style guide
- eMantix

Durchführung – Programming style guide:Beispiel

```
$data = new \User\Hosting\Classes\Model\FTPAccount();  
$data->setTimestampAdd(new \DateTime());
```

```
if (  
    null ==  
    ($account = $repository->findOneBy(  
        ['uid' => $kundennummer],  
        ['id' => 'DESC']  
    ))  
) {
```

Args mehrzeilig, ab gewisser Länge

```
    $a_tmp = explode('_', $account->getUsername());  
    if (isset($a_tmp[2]) && is_numeric($a_tmp[2])) {  
        $username = 'ftp_' . $kundennummer . '_' . ($a_tmp[2] + 1);
```

Array-Initialisierung in Kurzschreibweise

```
    } else {  
        $pre_username = 'ftp_' . $kundennummer . '_';  
        $c = 1;  
        while (  
            null ≠ $repository->findOneByUsername($pre_username . $c)  
        ) {  
            $c++;  
        }  
        $username = $pre_username . $c;  
    }
```

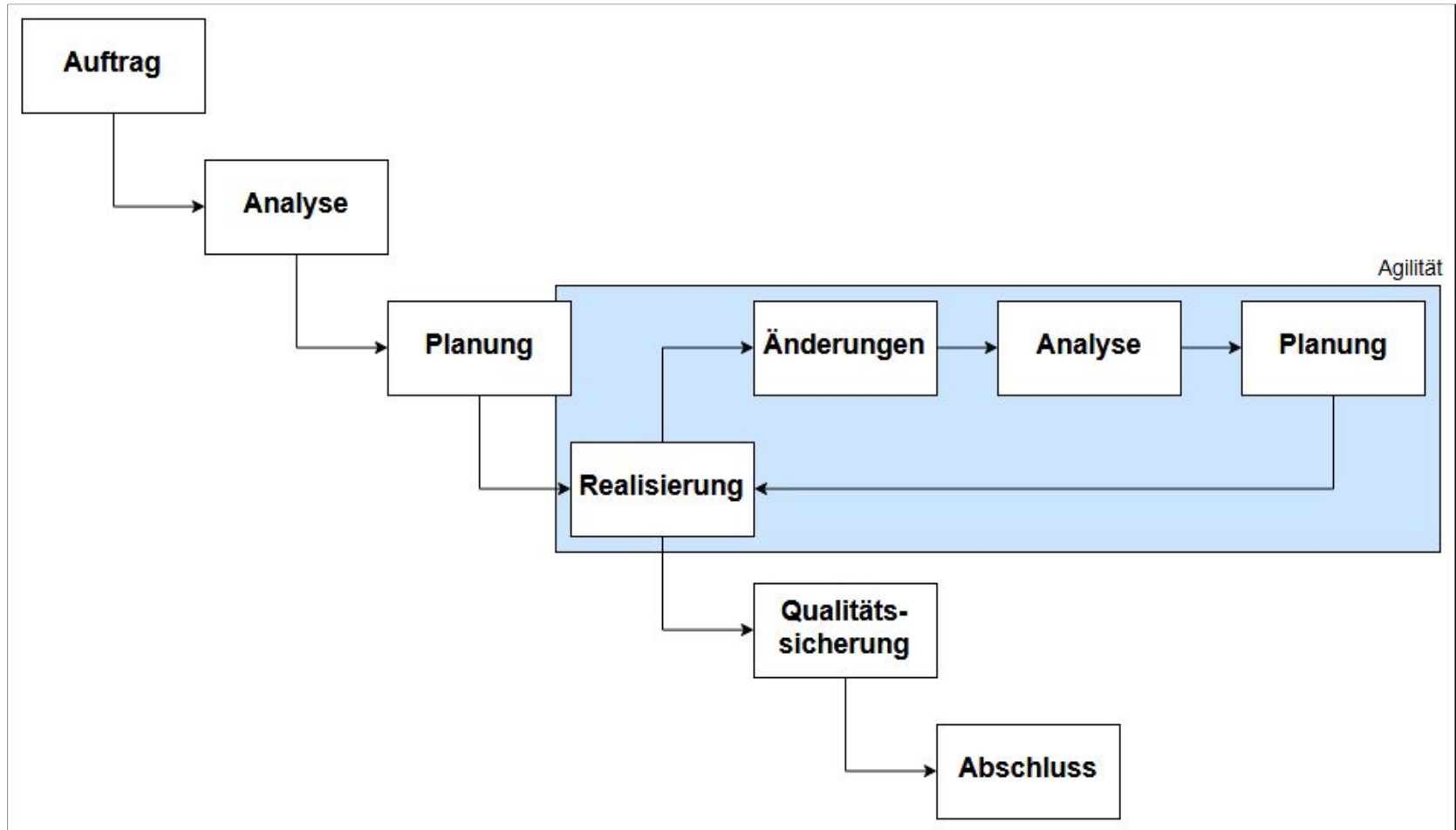
Abstände zwischen Variablen und Operatoren

```
} else {  
    $username = 'ftp_' . $kundennummer . '_1';  
}
```

Öffnende Klammer in gleicher Zeile

```
$data  
    ->setIdGeschaeftspartner($geschäftspartner->getId())  
    ->setUsername($username)  
    ->setUid($kundennummer)
```

Durchführung – Vorgehensmodell



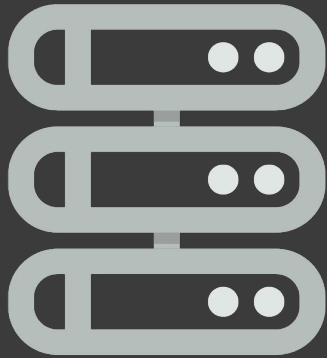
- Darstellung des abgewandelten Wasserfallmodells

Durchführung – Projektstrukturplan

The screenshot displays a Kanban board with three columns: 'Offen' (0 items), 'To Do' (14 items), and 'Closed' (43 items). The 'To Do' column contains three items: 'Pentest' (#54), 'Projektanbahnung' (#25), and 'Jerry approvalment' (#50). The 'Closed' column contains five items: 'API mit verschiedenen Apps testen' (#78), 'Änderung des Verschlüsselungsalgo' (#48), 'Directory traversal verhindern' (#69), 'Password-Richtlinien anheben' (#60), and 'Responsive Test mit verschiedenen Devices' (#68). Each item has a title, a status label, a number, and a user icon.

Column	Item Title	Status	Number	User
To Do	Pentest	FTP Verwaltung, QS	#54	[User Icon]
	Projektanbahnung	FTP Verwaltung, QS	#25	[User Icon]
	Jerry approvalment		#50	[User Icon]
Closed	API mit verschiedenen Apps testen	API	#78	[User Icon]
	Änderung des Verschlüsselungsalgo	Doing	#48	[User Icon]
	Directory traversal verhindern	FTP Verwaltung, security	#69	[User Icon]
	Password-Richtlinien anheben	FTP-Verwaltung, security	#60	[User Icon]
	Responsive Test mit verschiedenen Devices	Design	#68	[User Icon]

- Darstellung des Kanban-Boards und der jeweiligen Arbeitsschritte



- Implementierung -
Never change a running system?

Implementierung – Datenbank



- Objekt-relationale Abbildung(ORM)
- Separation von „Business-Logik“ und der Logik für die Datenbank
- Arbeiten mit Models



- Relationales Datenbankmanagementsystem (DBMS)
- Abspaltung von MySQL
- Standard auf Linux-Servern

Implementierung – Migration

- Backup der bestehenden Datenbank (Kein Backup – kein Mitleid!)
- Datenbank auf einem anderen Server erneut anlegen
- Tabelle erweitern → Verschlüsselungsalgorithmus
 - MD5 → SHA256 (128bit → 256bit Schlüssel)
- Anpassen der Datenbankabfrage → Beide Felder prüfen
- Befüllen der API mit Dummy-Angaben



ftp

Show/Hide List Operations Expand Operations

DELETE	/ftp/account	Löscht einen Datensatz
GET	/ftp/account	Ermittelt die Daten eines spezifischen Accounts anhand von \$a_params
PUT	/ftp/account	Updated einen Datensatz anhand des Payloads \$a_params
POST	/ftp/account	Legt einen neuen Datensatz anhand des Payloads \$a_params an

Implementation Notes

```
{
  "Kundennummer": 123456,
  "Homedir": "/var/www/123456/",
  "password": "SuperSicheresPasswort123",
  "Description": "Mein Passwort ist unsicher, weil da noch ein Sonderzeichen fehlt"
}
```

Parameters

Parameter	Value	Description	Parameter Type	Data Type
a_params	(required)	Required: Kundennummer, Password (atleast 8 chars, 1 uppercase, 1 lowercase letter, 1 special character), Additional: Homedir, Description	body	string

Parameter content type: application/json

Try it out!

DELETE	/ftp/account	Delete	Löscht einen Datensatz
GET	/ftp/account	Read	Ermittelt die Daten eines spezifischen Accounts anhand von \$a_params
PATCH	/ftp/account	Update	Updated einen Datensatz anhand des Payloads \$a_params
POST	/ftp/account	Create	Legt einen neuen Datensatz anhand des Payloads \$a_params an

Implementation Notes

~~DRUC?~~ CRUD

```
{
  "kundennummer" : 123456,
  "homedir" : "/var/www/123456/",
  "password" : "SuperSicheresPassword123",
  "description" : "Mein Passwort ist unsicher, weil da noch ein Sonderzeichen fehlt"
}
```

Parameters

Parameter	Value	Description	Parameter Type	Data Type
a_params	<div>(required)</div> <div>Parameter content type: application/json ▼</div>	Required: Kundennummer, Password (atleast 8 chars, 1 uppercase-, 1 lowercase letter, 1 special character), Additional: Homedir, Description	body	string

Try it out!



- Wirtschaftlichkeit -

Geld nennt man heute Knete,
weil man jeden damit weich
bekommt.

Wirtschaftlichkeit – Amortisation

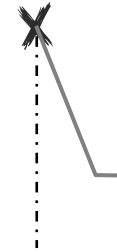
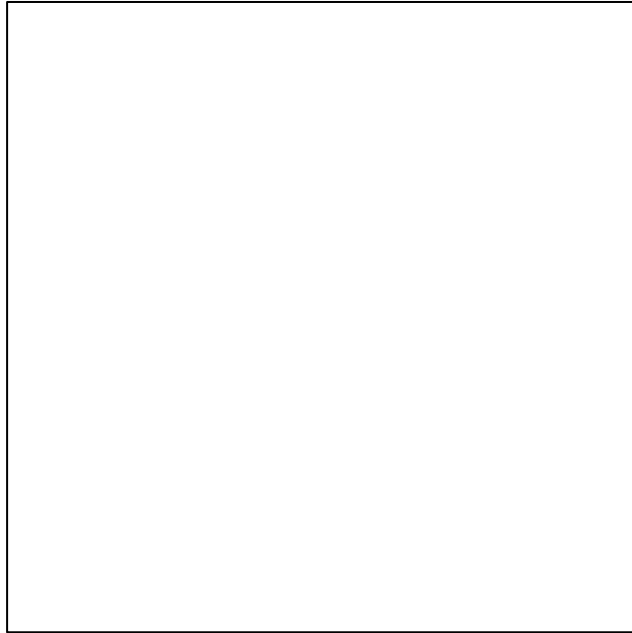
Vorgang	Anz. Vorgänge	Dauer(Ist) pro Vorgang in Minuten	Dauer(Ver- gleich) pro Vorgang in Minuten	Gesamtersparnis in Minuten
Anlegen eines Accounts	20/10	3/0	4/7	$20 * 3 + 10 * 0 = 60$ $20 * 4 + 10 * 7 = 150$ $150 - 60 = 90$
Bearbeiten eines Accounts	30/25	3/0	4/7	$30 * 3 + 25 * 0 = 90$ $30 * 4 + 25 * 7 = 295$ $295 - 90 = 205$
Löschen eines Accounts	3/10	2/0	3/6	$3 * 2 + 10 * 0 = 6$ $3 * 3 + 10 * 6 = 69$ $69 - 6 = 63$
				358



Erste Angabe
= Mitarbeiter

Zweite Angabe
= Kunde

Wirtschaftlichkeit – Amortisation



break-even point
Nach ca. 6,4 Monaten
erreicht

$\frac{\text{Monat}}{\text{Monat}} \times \text{Stundensatz €} = \text{Einsparung pro Monat €}$

$$\Rightarrow \frac{358}{60} \times 50 \text{ €} = 289,33 \text{ €}$$

$$\frac{\text{Projektkosten}}{\text{Einsparung € pro Monat}} = \text{Amortisationsdauer}$$

$$\Rightarrow \frac{1850 \text{ €}}{289,33} = 6,4 \text{ Monate}$$





- Fazit -

Prüfungen erwarte bis zuletzt.



Fazit – Resultat und Bewertung

- ✓ Wirtschaftlich
- ✓ Funktionell
- ✓ Innovativ
- ✓ Mit den Vorgaben konform

Vielen Dank für Ihre Aufmerksamkeit!

