

# AFP-Resourcenplanung, Praxisbeispiel Einsatzplanung von Messdiener

Ulrich Hoppe, Hoppe Unternehmensberatung - Heusenstamm (Frankfurt)

Zu einer Zeit, in der ich mich in AFP eingearbeitet habe suchte ich eine anspruchsvolle Aufgabe zur Umsetzung eines Pilotprojekts. (nur so lässt sich eine Sprache/Werkzeug wirklich lernen). Der Einsatzplan der Messdiener in unserer Gemeinde kristallisierte sich schnell zu einer idealen Aufgabenstellung. Zentrale Fragestellung für das Internet war: „Wer macht bei welcher Veranstaltung /Messe welchen Dienst“.

## www.StCaecilia.de/Messdiener

Die Applikation lebt von nur 3 Tabellen: Personen, Veranstaltungen, Aufgaben. Alle Tabellen werden online von einem Messdienerleiter gepflegt.

Messdiener von St.Cäcilia, Heusenstamm

Suchen:  Suchen Hilfe zurück weiter Neu 27.06.2004-03.07.2004

Datum	Uhrzeit	Aufgabe	Person
Sonntag 27.06.2004 18:30 Uhr		großer Altar	Sonja Wilzert
		großer Altar	Franciska Wilzert
		kleiner Altar	Michelle Geiger
		Kleiner Altar	Johannes Lippert
		Sammler	Pia Kämmerer
Freitag 02.07.2004 19:00 Uhr		Zeremonie	Alex Koppert
		großer Altar	Eric Laucht
		großer Altar	Daniel Fuhrt
		kleiner Altar	Regina Paul
		kleiner Altar	Valentin Laucht
Sonntag 03.07.2004 14:00 Uhr		Trauung	

**infoBoxX**  
04.07.2004  
Hier stehen aktuelle Informationen

- Am Sonntag, 27. Juni findet um 19 Uhr in Maria Himmelskron der **Schönbegräbnisfest** statt. Alle sind eingeladen daran teilzunehmen.
- Die **Ferienliste** liegt ab jetzt in der Sakristei aus. Wenn ihr im Sommer in Urlaub seid, gibt es die Möglichkeit sich bis zum 15. Juli in der Liste einzutragen. Auch die Eintragung über Mail und Fax ist möglich.
- **Freitag, 2. Juli: Fußball spielen** für alle Mess: Beginn: 19.30 Uhr, Treffpunkt: Pfarrhaus
- **Missag Oberhausen:** Es besteht noch die Möglichkeit sich anzumelden / Anfahrtszeit am Kirchplatz wird noch bekanntgegeben und angepasst.

In der Hauptmaske werden wochenweise Messen mit einer Liste der Aufgabe dargestellt. Jede Aufgabe wird von genau einer Person abgedeckt. Eine 2. Tabelle zeigt eine Infobox mit News. Über Links kommt man zu einer Rangliste und einem Login.

## Gemeindevorstellung versus AFP-Elemente

Neben den Anforderungen der Gemeinde wollte ich natürlich zahlreiche Elemente von AFP implementieren. Diese Applikation nutzt: Cookies, Session – Variablen, E-Mail Versand, Login-Routine. Ich machte mich mit Submit-Button vertraut. Probleme tauchten auf, da jede Session von unterschiedlichen Threads genutzt wurde. Hier kamen dann Globale- und Session-Variablen zum Einsatz. Gegenüber einer Desktop-

Applikation bereitete mir das Öffnen und Schließen zunächst Schwierigkeiten. Bei einem Absturz waren Tabellen noch offen. Strukturänderungen waren erst möglich, nachdem mein AFP-Hostanbieter alle threads geschlossen hatte. Richtige Fehlerbehandlung und Einträge in den Methoden der afpa.Code-Datei lösten einige Probleme.

Es musste ein Umdenken von modalen- und nichtmodalen Forms zu nichts voneinander wissenden HTML-Seiten stattfinden. (Dies konnte ich schon von WebMaxX)

## Vorgehensweise: *Azige Controls > Edit > valid*

Zunächst erstellte ich AFP-Seiten um Daten aus Tabellen nur anzuzeigen. Hierbei nutze ich Textboxen, Bilder, Checkboxes, Editboxen und die Comboboxen. Dann beschäftigte ich mich mit der Selektion von Daten. Der Anwender gibt ein Suchbegriff ein und nur diese Treffer sollen angezeigt werden. Erst im zweiten Schritt beschäftigte ich mich mit der Dateneingabe in der ich auch Validprüfungen auf Feldebene durchführte. Bei Falscheingaben zeige ich die Seite einfach wieder an. Eine zuvor gefüllte lcErrorMsg wird hierbei angezeigt. Das nächste Ziel war eine AFP-Seite zu erstellen, die mir automatisch eine EMail sendet.

## Email senden - (Umweg wegen, Execute() ohne parameter)

```
lcQuerystring = Request.cQueryString  
Request.cQueryString="Subject=Veranstaltung  
Show.afp&Body=Zeit:"+time()+lcUsername+  
" "+lcUserVorname  
Request.ResetQueryString()  
server.Execute("Email.afp")
```

```

Request.QueryString=lcQueryString
Request.ResetQueryString()

-----
<%
* Email.afp Parameter: tcSubject,tcBody
? '<body><html>'
? '<head><title>Email</title></head>'

* Parameter abfangen, wenn man über ein Get
kommt ( Aufruf über eine URL)
lcSubject="AUTO-EMAIL:"
+upper(Request.QueryString("Subject"))
lcBody=upper(Request.QueryString("Body"))

lcEmailFunk=FILE.Fullpath()+"Prg\EMailFunk.
fxp"

Local loEMailFunk
loEMailFunk=NewObject("EmailFunk",lcEmailFu
nk)

?loEMailFunk.Mailwebmaster(lcSubject,lcBody
)

* FXP wieder frei geben
release loEMailFunk
clear class EMailFunk
clear Program &lcEMailFunk
loEMailFunk = .NULL.
? '</body></html>'
%>

```

---

#### *Prozedursammlung / einbinden von DLL*

---

Mit der Zeit erkannte ich wiederkehrende Programmteile. Ich fügte eine Klasse und eine Prozedursammlung hinzu. Gegenüber der Desktop-Anwendung entstanden Probleme beim Freigeben der mit „set procedure“ aufgerufenen Funktionen. Geladene DLL müssen ordnungsgemäß entladen werden. Überhaupt musste ich mich innerhalb AFP viel sorgfältiger um das Aufräumen nach dem Aufruf einer AFP-Seite kümmern, als dies bei einer Desktop Applikation der Fall ist. (Datenumgebung)

---

#### *Auf Anrieb 70 Betatester im Alter von 10-18 Jahren*

---

So viel Spass man auch bei der Entwicklung von AFP und Web-Seiten hat, so möchte man doch immer, dass jemand seine Seite auch nutzt. Die Onlinedarstellung wurde schnell zum offiziellen Einsatzplan. 70 Messdiener und Eltern sorgten für regelmäßigen Traffic und stellen eine optimale Gruppe von Betatester dieser unkommerziellen Web-Anwendung dar. Traffic war mir garantiert, denn die Daten wurden gepflegt und es war notwendig, die Informationen regelmäßig abzurufen.

---

#### *Command-Button, Picture-Buttons*

---

Meine Buttons gebe ich allen den selben Namen (cmdButton). Zur Differenzierung frage ich dann die Caption von dem Submit Button ab. Die optimal Lösung für die Button mit Bildern habe ich noch nicht gefunden. Ein Bild hat keine Caption.

---

#### *Gestaltung CSS*

---

Nachdem die Funktionsweise stabil schien, beschäftigte ich mich mit dem zielgruppenspezifischen Layout (10-18-jährige Anwender). Alle Informationen sollen übersichtlich und leicht verständlich dargestellt werden. Im Projektverlauf wurden die allgemein sichtbaren Daten reduziert. Aus Datenschutzgründen sollten nur registrierte Anwender über eine Login-Maske in der Lage sein, zu mehr Informationen zu gelangen. Ich stellte um auf Style-Sheets mit CSS Dateien. Super: TopStyle-Lite (siehe [www.BradSoft.com](http://www.BradSoft.com))

---

#### *Wie geht es weiter, mandantenfähig->Verbreitung*

---

Ich werde diese Anwendung künftig weiteren Gemeinden kostenfrei zur Verfügung stellen. Mein nächster Schritt ist hierzu eine mandantenfähige Version.

---

#### *Verfügbarkeit Source-Code*

---

Dieses Projekt entstand, weil ich eine Problemstellung für AFP suchte und in dieser Applikation ein optimales Betätigungsfeld fand. Dieses Grundgerüst werde ich auch künftig nutzen können. AFP ist gut strukturiert und es macht Spass hiermit zu arbeiten. Die Einsatzplanung werde ich auch in Zukunft ehrenamtlich weiter entwickeln. In diesem Projekt verbergen sich Lösungsansätze, die für einen AFP-Einsteiger sicherlich einige AFP-Beispielseiten zeigen. Auf Anfrage hin stelle ich den dokumentierten Source-Code gerne kostenfrei zur Verfügung.

Siehe: [www.StCaecilia.de/Messdiener](http://www.StCaecilia.de/Messdiener)

Kontakt: [Ulrich.Hoppe@VFP-Consulting.de](mailto:Ulrich.Hoppe@VFP-Consulting.de)