Nachdruck aus dem FileMaker Magazin
siehe auch: www.filemaker-magazin.de

Pro Advanced mittel

(Human) Interface Design (Teil 1)

Der Dialog zwischen Mensch und Anwendung

Holger Darjus

(Jg. 1962) ist Technischer Leiter des Medienvorstufen-Unternehmens Einsatz Creative Production GmbH & Co. KG in Hamburg mit rund 45 Mitarbeitern. Er entwickelt seit 1996 FileMaker Datenbanken speziell für die Medienvorstufe.

h.darjus@einsatz.de

Interface Design oder GUI (Graphical User Interface) – Begriffe, die sicherlich jeder, der sich mit FileMaker beschäftigt, schon gehört hat. Aber was genau ist ein Interface Design und wieso ist es für die Programmierung von anwenderfreundlichen Lösungen so wichtig? Um diese Fragen zu beantworten, ist es zunächst einmal ganz hilfreich, den Begriff „Interface Design" ins Deutsche zu übersetzen: Interface = Schnittstelle, Design = Gestaltung. Mit Schnittstelle ist in unserem Fall die Verbindung zwischen dem Menschen vor dem Computer (= Anwender) und der FileMaker Lösung gemeint. Mit „Gestaltung" ist demzufolge nicht nur das Aussehen einer Lösung gemeint sonder auch der Vorgang wie etwas eingegeben, ausgewertet oder bearbeitet wird.

In meiner Beitragsserie geht es also nicht darum, die Lösung zu präsentieren, die ich am besten oder am schönsten finde. Vielmehr möchte ich grundlegende Überlegungen zum Thema Interface Design anstellen und Vorschläge aufzeigen, wie FileMaker Entwickler ihre Lösungen effizient, anwenderfreundlich, verständlich und spaßfördernd gestalten können.

Denn so funktionstüchtig und umfassend eine Lösung auch ist – der Anwender muss sie verstehen, um effektiv damit arbeiten zu können. Natürlich kann man ihn in ausführlichen Schulungen in das Programm einarbeiten, aber wäre es nicht viel einfacher, wenn sich die Lösung dem Anwender von selbst erschließt? An dieser Stelle kommt das Interface Design ins Spiel.

Ich habe fantastische Lösungen mit unglaublichen Formeln und Scripts gesehen, mit denen man die tollkühnsten Dinge machen konnte. Allerdings kam der Entwickler bei dem Versuch, mir seine Lösung vorzuführen ins Schlingern, weil er nicht mehr wusste, wohin er klicken muss, um zum gewünschten Ergebnis zu kommen. Wie soll dann erst der arme Anwender damit zurechtkommen? Erstaunlicherweise gewöhnt sich der Mensch an vieles und so mancher Kummer gewohnte Anwender von **SAP**-Lösungen kommentiert „merkwürdige" Vorgänge mit den Worten: „Das war schon immer so, das muss man eben nur wissen ..."

Wir können das besser!

Auch ich habe im Laufe meiner Karriere als FileMaker Entwickler so manches „Unheil" angerichtet, aber mittlerweile investiere ich bei der Entwicklung meiner Lösungen schätzungsweise 70 % der Arbeit in die Benutzerführung und die Gestaltung. Das klingt zunächst aufwendig, zahlt sich aber in vielfacher Hinsicht aus. Insbesondere der Support- und Schulungsaufwand ist seitdem extrem zurückgegangen. Ob die Anwender meiner Lösungen nun vor einer Adressverwaltung, der Auftragsabwicklung oder einer Bilddatenbank sitzen – sie finden sich auf Anhieb zurecht und können sich sofort an die Arbeit machen. Auch bei Produkten, die verkauft werden sollen, zahlt sich der Aufwand aus: Das Auge kauft tatsächlich mit

© 1994 - 2010 K&K Verlag GmbH, Hamburg

Nachdruck aus dem FileMaker Magazin
siehe auch: www.filemaker-magazin.de

und insbesondere im Agenturumfeld wird auf das Erscheinungsbild viel Wert gelegt. Aber wie bereits erwähnt: Es geht nicht nur darum, wie etwas aussieht, sondern vor allem um die Eingängigkeit der Funktionsweise.

Eine entscheidende Erkenntnis habe ich beim Lesen der **FileMaker Training Series**[1] gewonnen. Dort stand: *„Data modeling is the process of turnig a real-world problem into a structure that can be manipulated by a computer."*

Auf Deutsch und im Zusammenhang mit Interface Design könnte man daraus ableiten: *„Interface Design ist die Umsetzung einer Aufgabe aus dem wahren Leben in etwas, was der Benutzer am Computer bearbeiten kann."*

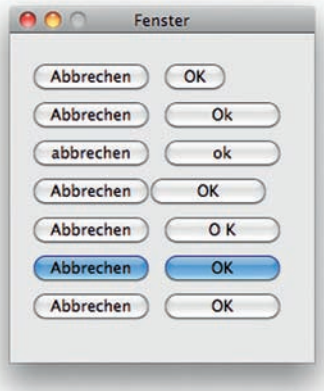
Nun bin ich – wie vermutlich die meisten FileMaker Entwickler – kein ausgebildeter Screen-Designer. Aus diesem Grund habe ich mich auf die Suche gemacht, wo man am besten „abkupfern" und doch seine eigene Handschrift mit der gewünschten Benutzerfreundlichkeit kombinieren kann. Vergebens warte ich nun ja schon seit Jahren auf eine Initiative aus dem Hause FileMaker, die es einem Entwickler einfacher macht, zeitgemäße Layouts und Prozeduren zu erstellen. „Skins" ist das Zauberwort: Man positioniert einfach eine Schaltfläche oder einen OK-Button auf einem Layout und überlässt den Rest dem Betriebssystem. Ein OK-Button sähe dann jeweils so aus, wie es das Betriebssystem vorgibt.

Windows 7 Windows XP Mac OS X

Da dieser Wunsch aber bislang noch nicht erfüllt wurde, müssen wir uns eben selbst um das gute Aussehen und Funktionieren unserer Lösung kümmern.

Weil ich mich nahezu ausschließlich im Apple-Umfeld bewege, habe ich mich mal umgeschaut, was Apple zum Thema Interface Design zu bieten hat – und bin schnell fündig geworden: **Die Human-Interface-Guidelines (IGL)[2] sind eine hervorragende Unterstützung für jeden Entwickler.**

In diesem knapp 400 Seiten umfassenden Werk wird ausführlich beschrieben, wie eine Benutzeroberfläche aufgebaut werden soll und welche Fehler bei der Gestaltung vermieden werden müssen. So wird beispielsweise erklärt, wie ein „OK- und Abbrechen-Button" auszusehen hat. Ist doch ganz klar, denkt sich vielleicht



[1] *www.filemaker-magazin.de/shop/produkt/2282 (IGL) Diese können (leider nur in englischer Sprache) bei Apple heruntergeladen werden:*

[2] *http://developer.apple.com/mac/library/documentation/UserExperience/Conceptual/AppleHIGuidelines/OSXHIGuidelines.pdf*

[3] *http://developer.apple.com/technologies/tools/xcode.html*

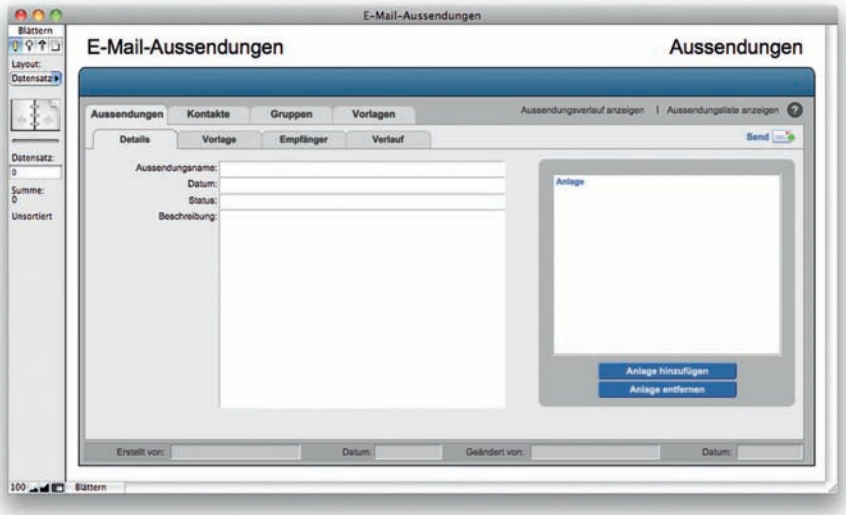
der eine oder andere. Nun, mir fällt gleich eine ganze Reihe von Kombinationen ein, die gar nicht OK sind. Richtig ist bei den links gezeigten Beispielen nur das unterste Paar.

Als glorreiche Unterstützung gibt es den **Interface Builder**. Dieses Programm ist Bestandteil der **Xcode Tools** und unter „Extras" auf der Installations-DVD von Mac OS X zu finden oder auf den Webseiten bei Apple[3] kostenlos herunterzuladen.

Mit dem **Interface Builder** lassen sich ganz wunderbar die Grundgerüste unserer Lösungen entwickeln. Da es leider keine unmittelbare Unterstützung von FileMaker Anwendungen gibt, muss mit Bildschirmfotos gearbeitet werden. Für die Nachbearbeitung manches Elemente ist es gelegentlich von Vorteil, wenn man sich ein wenig mit Photoshop auskennt.

Aber genug der Vorrede. Im ersten Teil dieser Serie möchte ich ein paar Grundüberlegungen anstellen. Um das Ganze etwas anschaulicher zu gestalten, habe ich eine von FileMaker mitgelieferte Beispieldatei nach den hier angestellten Überlegungen modifiziert.

Die Originaldatei namens ***E-Mail-Aussendungen.fp7*** ist in dem Programm-Order von FileMaker zu finden. (Im „FileMaker 11"-Ordner den „Deutsch Extras"-Ordner öffnen, dann „Vorlagen" und wiederum „Geschäft - Personal & Anlagen" öffnen.)



Schritt 1: Layoutgröße

Um diese zu bestimmen, sollten wir uns einen groben Überblick darüber verschaffen, wie viele Felder, Tasten usw. auf unserem Lay-

© 1994 - 2010 K&K Verlag GmbH, Hamburg

Nachdruck aus dem FileMaker Magazin
siehe auch: www.filemaker-magazin.de

out untergebracht werden müssen. Zudem ist zu klären, von wem die Lösung genutzt wird und wie dessen Arbeitsbedingungen aussehen. Vielleicht sind Sie als Entwickler ja stolzer Besitzer eines 30"-Monitors und können mit den bereitstehenden Pixeln verschwenderisch umgehen. Aber finden die Anwender die gleichen Bedingungen vor? Es sollte als immer nach dem kleinsten gemeinsamen Vielfachen gesucht werden, denn nichts ist nerviger als das Scrollen in einer Eingabemaske. Unsere Beispieldatei weist in der Detailansicht eine Größe von 910 x 645 Pixeln auf. Ich habe es zunächst dabei belassen, zeige aber in einem der nächsten Beiträge, wie die Layoutgröße dynamisch aufgebaut werden kann.

Schritt 2: Struktur

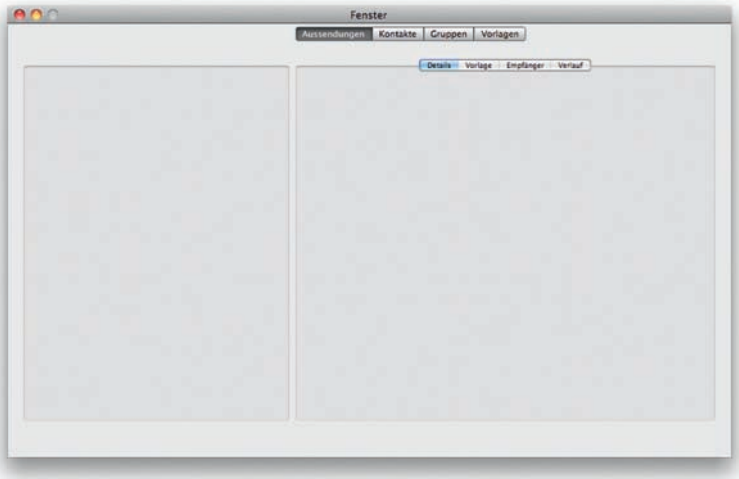
Zunächst müssen wir überlegen, mit welchen Tabellen der Anwender überhaupt in Berührung kommt. In unserem Beispiel sind das die Tabellen ***Aussendungen***, ***Kontakte***, ***Gruppen*** und ***Vorlagen***. Um dem Anwender zu visualisieren, in welcher Tabelle er sich gerade befindet, positioniere ich einen entsprechenden Hinweis im Kopfteil des Layouts. Später werden in diesem Bereich noch weitere Tasten zum Navigieren in den Datensätzen und zum Wechsel zwischen Listen- und Detailansicht untergebracht. Die Navigation zwischen den Tabellen ist vergleichbar mit dem Wechseln von Rubriken. Die **Interface Guidelines** sehen für einen solchen Fall das Element „Segmented Control Attributes“ vor.

Ich erstelle also im **Interface Builder** ein neues Fenster und positioniere ein entsprechendes Element mit vier Segmenten:

Weil der Anwender ja auch sehen soll, wo er sich gerade befindet, erstelle ich gleich noch die passenden Grafiken mit dem Attribut „selected“:

Beim Datenteil habe ich mich dazu entschlossen, die Basisinformationen immer links anzeigen zu lassen. Die erweiterten Informationen sind – ähnlich wie in der Vorgabe von FileMaker – unter den einzelnen Registern zu finden. Zum Gruppieren von Informationen sehen die **IGL** „Boxen“ vor: Eine Box ist dann eigentlich nichts weiter als ein Rahmen um eine Gruppe von Feldern und Elementen.

Da es im konkreten Fall nicht viel zu gruppieren gibt, lege ich einfach zwei Boxen auf das Layout – links die Box für die Basisinformationen und daneben die Box mit den Registern. Unser Rohlayout ist damit auch schon fertig.



Ein weiteres Feature des **Interface Builders** ist der **CocoaSimulator**. Dieser wird mit der Tastenkombination ⌘ + R gestartet und ermöglicht es dem Entwickler, das Interface mit all seinen Schaltflächen zu testen.

Die Sache hat allerdings einen Haken: Die einzige Möglichkeit, unser neu erstelltes Interface in FileMaker zu übernehmen, ist bedauerlicherweise ein Bildschirmfoto. Also mache ich einen Screenshot und platziere diesen im Layout von FileMaker. Anschließend setze ich ihn mit **Anordnen → Ganz nach hinten** und fixiere ihn. Alle weiteren Elemente entstehen weitestgehend aus einer Kombination von Einzelgrafiken und den Möglichkeiten von FileMaker.

Schritt 3: Felder und Schriften

In der linken Box positioniere ich nun das erste Feld. Auch für Felder gibt es in den **IGL** Vorgaben bezüglich Größe und Schriftart. Niemals „Arial“! Diese leider sehr häufig verwendete und zudem voreingestellte Schrift ist ein unschönes Plagiat der „Helvetica“ und äußerst ungeeignet für die Bildschirmdarstellung. Ich werde an dieser Stelle nicht weiter darauf eingehen, welche Eigenschaften eine Schrift für eine gute Lesbarkeit am Monitor mitbringen muss. Nur so viel: Namhafte Schriftdesigner haben sich viele Gedanken zu diesem Thema gemacht und exzellente Schriften entworfen. Unter Mac OS wird weitestgehend auf die „Lucida Grande“ gesetzt, während Windows die „Trebuchet MS“ und „Segoe“ verwendet. Im Internet findet man häufig die „Verdana“.

© 1994 - 2010 K&K Verlag GmbH, Hamburg

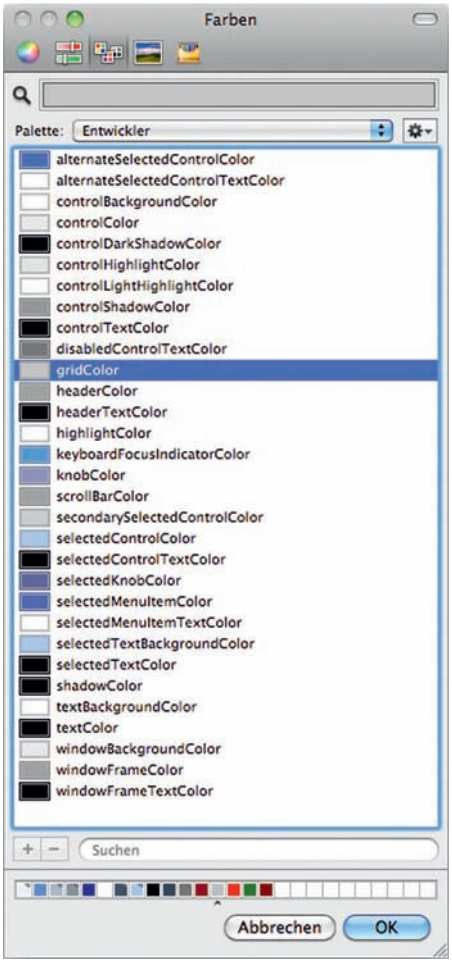
Nachdruck aus dem FileMaker Magazin
siehe auch: www.filemaker-magazin.de

Da ich in unserem Beispiel den **IGL** treu bleibe, entscheide ich mich für die „Lucida Grande" mit einer Schriftgröße von 11 Punkt. Zudem mache ich mein Feld 18 Pixel hoch und gebe ihm eine mittige vertikale Textausrichtung. Eine weiße Füllung und eine 1 Punkt starke graue Linie links und oben runden das Ganze ab.

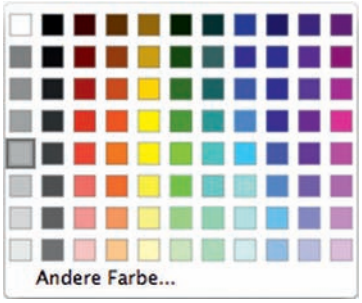
Schritt 4: Farbpalette und verwendete Farben

Hat man unter Mac OS die **XCode-Tools** installiert, bekommt man auch bei der Wahl der Farben Unterstützung vom Betriebssystem.

Klicken Sie dafür im Inspektor das Farbfeld für die Linie an und wählen Sie dort den Menüpunkt „Andere Farbe ..." aus. Damit gelangen Sie in die erweiterte Farbpalette, wo Ihnen das Farbset „Entwickler" zur Verfügung steht, in dem die meisten Farben bereits fertig definiert sind.



Da die Rahmenfarbe in FileMaker „global" ist (d. h. alle vier Kanten haben die gleiche Farbe), muss man hier mit einem Kompromiss leben, wenn man nicht für jedes Feld eine Grafik hinterlegen will. Mir persönlich ist das zu aufwendig Ich wähle „gridColor" als Kantenfarbe für meine Felder – das entspricht in der Standard-Farbpalette bei 88 eingeblendeten Farben dem vierten Kasten von unten in der linken Spalte.



Möchten Sie eine bestimmte Farbe aus einem anderen Layout oder einem anderen Programm übernehmen, können Sie auf die kleine Lupe in der Farbpalette klicken. Diese Lupe ist eigentlich eine Pipette und mit ihrer Hilfe kann die Farbe eines beliebigen Pixels aufgenommen werden.

Wurde mit der Pipette eine Farbe ausgewählt, kann diese anschließend in den unteren Teil des Farbauswahlfensters gezogen werden und steht dort fortan zur Verfügung.

Schritt 5: Schalter / Knöpfe / Tasten

Hierfür stellt der **Interface Builder** die unterschiedlichsten Stile zur Verfügung. Die folgende Abbildung zeigt ein paar Beispiele:

Für jeden dieser Buttons gibt es in den **IGL** eine feste Bestimmung, wo und wann er verwendet werden sollte. Wichtig ist aber vor allem, auf den richtigen Kontext zu achten, denn dem Anwender soll ja etwas zur Verfügung gestellt werden, was ihm vertraut ist und immer „das gleiche" tut. Außerdem können bestimmte Verhaltensweisen von Tasten dem Anwender das Leben entweder erleichtern oder erschweren.

Beispiele

Der Text einer Taste ohne Funktion sollte immer grau sein. Das signalisiert dem Anwender sofort, dass die Funktion zurzeit nicht zur Verfügung steht. In unserer Beispieldatei finden sie zwei „schlechte" Beispiele: Die Tasten „Anlage hinzufügen" und „Anlage entfernen".

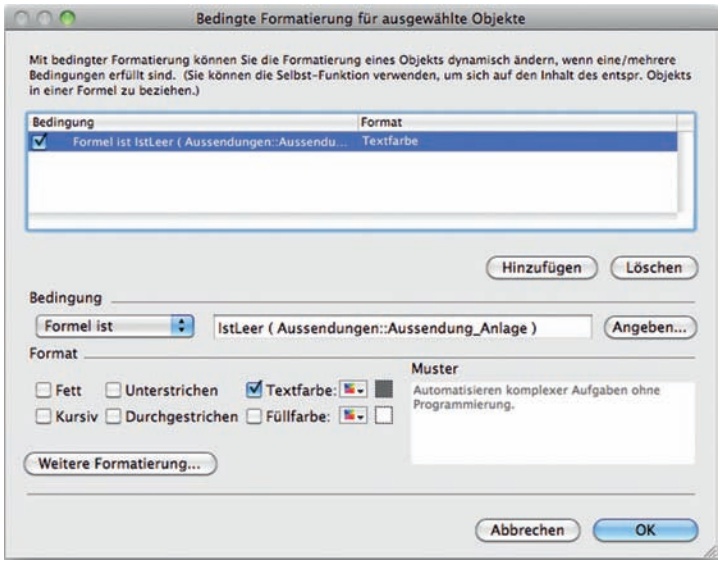
Die Taste „Anlage entfernen" ist nämlich auch anklickbar, wenn gar keine Anlage existiert. (Zum Glück erscheint keine Dialogbox mit so sinnlosem Inhalt wie „Die Anlage kann nicht entfernt werden, weil keine Anlage existiert".)

© 1994 - 2010 K&K Verlag GmbH, Hamburg

Nachdruck aus dem FileMaker Magazin
siehe auch: www.filemaker-magazin.de

Gleiches gilt für die Taste „Anlage hinzufügen": Diese ist auch aktiv, wenn bereits eine Anlage existiert, sodass der Anwender ein neues Dokument hinzufügen kann – ohne einen Hinweis darauf, dass das bereits vorhandene Dokument dabei überschrieben wird. Hier sollte der Anwender auf jeden Fall mithilfe einer Dialogbox über die Folgen seines Handelns informiert werden, damit er den Vorgang im Zweifelsfall abbrechen kann.

Entscheidet man sich dafür, zwei Tasten zu verwenden, sollte mit „Bedingter Formatierung" gearbeitet werden. In unserem konkreten Fall sollte die Taste „Anlage entfernen" wie folgt formatiert werden, damit der Text grau erscheint, wenn keine Anlage hinterlegt ist.

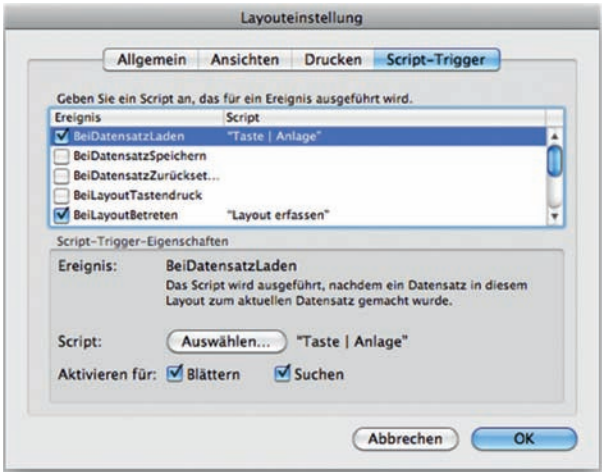


Im Prinzip könnte der Vorgang auch über eine einzige Taste realisiert werden. Dazu kann man entweder ein Formelfeld anlegen oder die Beschriftung der Taste unter Zuhilfenahme einer $$-Variablen erstellen:

```
Wenn (IstLeer (Aussendungen::Aussendung_Anlage);
     "Anlage hinzufügen"; "Anlage verändern")
```

Im letzteren Fall muss jedoch sichergestellt sein, dass durchgängig mit FileMaker 11 (oder höher) gearbeitet wird, Wie auch immer man sich entscheidet, die Tasten sollten auf jeden Fall eindeutig zeigen, was sie tun bzw. was nicht. Mithilfe einer QuickInfo können Sie den Anwender darüber informieren, aus welchem Grund eine Taste grau und somit nicht aktiv ist.

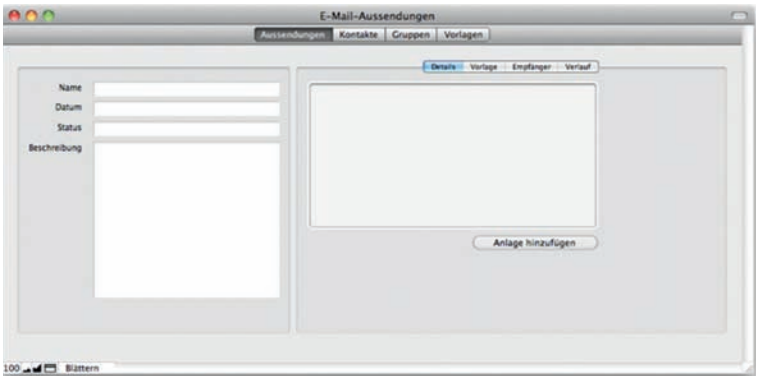
Die Variable **$$Taste | Anlage** muss dann per Script beim Betreten des Datensatzes gesetzt werden. Das lässt sich sehr elegant per Script-Trigger in den Layouteinstellungen realisieren.



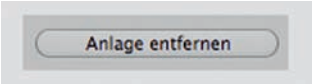
Natürlich muss das auf der Taste liegende Script entsprechend der Beschriftung reagieren:

```
Taste Anlage
◆ Kommentar
  Text:  "© 10.08.2010 E I N S A T Z Creative Production | Holger Darjus
         Steuert die Funktion der Taste „Anlage" und fügt wahlweise eine
         Anlage in das Feld „Anlagen" oder hebt die Zuordnung auf.
◆ Wenn ▼
  Formel:  IstLeer (Aussendungen::Aussendung_Anlage)
◆ Datei einfügen
  Zielfeld angeben:  Aussendungen::Aussendung_Anlage
◆ Sonst ◆
◆ Eigenes Dialogfeld anzeigen
  Mitteilung:  "Anlage löschen: " & LiesAlsText (Aussendungen::Aussendung_Anlage)
  Schaltflächen:  "Abbrechen", "Löschen"
◆ Wenn ▼
  Formel:  Hole (LetzteMeldungswahl) = 2
◆ Feldwert setzen
  Zielfeld angeben:  Aussendungen::Aussendung_Anlage
  Berechnetes Ergebnis:  ""
◆ Ende (wenn) ▲
◆ Ende (wenn) ▲
```

Bis hierhin sieht unsere modifizierte Anwendung so aus:



In der nächsten Folge behandle ich dann die Themen „Portale" und Listen, „modale Dialogboxen" sowie ein paar kleine Tricks, wie sich solch unschöne Effekte beim Anklicken von Tasten verhindern lassen:



◆

© 1994 - 2010 K&K Verlag GmbH, Hamburg

FileMaker Magazin

Das FileMaker Magazin

- Einzige, deutschsprachige Fachzeitschrift zu FileMaker
- Wissen aus erster Hand von anerkannten FileMaker Fachautoren
- Große Themenvielfalt für Anwender, Entwickler und Fortgeschrittene

Exklusiv für Premium-Abonnenten

- Sechs FMM Ausgaben pro Jahr
- Kostenlose Nutzung des Abonnentenbereichs auf ***www.filemaker-magazin.de***
- PDF-Online-Archiv mit allen bisher erschienenen Ausgaben
- Jede Ausgabe mit kostenlosen Beispieldateien und Zusatzinfos zum Download

Unser Service

- Aktuelle Neuheiten, Tipps und Infos, Kleinanzeigen und vieles mehr jederzeit online auf unseren Webseiten
- Hilfe bei allen Fragen zu FileMaker im FMM Forum
- Kompetente Beratung zum Kauf von FileMaker Lizenzen: Einfach anrufen +49 (0)40 589 65 79 70.

Hier finden Sie Aktuelles zu FileMaker Produkten, egal ob Sie kaufen, mieten oder sich einfach informieren möchten.

Eine kostenlose Leseprobe des FileMaker Magazins erhalten Sie, wenn Sie hier klicken.

Wenn Sie sich für ein FileMaker Magazin Abo interessieren, klicken Sie bitte hier!

K&K Verlag GmbH · www.filemaker-magazin.de · Telefon +49 40 589657970