

Evolutionäre Entwicklung einer Web-Anwendung im kirchlichen Umfeld

Use-Case-basierte Vorgehensweise in den Projekten
der Produktfamilie KRZ.DaviP

Alexander van der Vekens, avite, Herrenberg
Ursula Schmitt-Wagner, KRZ-SWD, Eggenstein

- **Überblick über die entwickelte Software**
- **Vorgehensmodell der Entwicklungsprojekte**
- **Anforderungsmanagement**
- **Testmanagement**
- **Fazit**



KRZ.DaviP-core	Zentrale Verwaltung der Daten der Kirchenmitglieder und ihrer Familienangehörigen 8,3 Mio. Mitglieder, ca. 11 Mio. Personendatensätze
KRZ.DaviP-org	Web-Anwendung für die Meldewesen-Abteilungen der Landeskirchen und Diözesen 5 Meldewesen-Org-Stellen mit ca. 30 Mitarbeitern Verwaltung von ca. 8000 Pfarrbezirken
KRZ.DaviP-online	Web-Anwendung für den Einsatz im Pfarramt und anderen kirchlichen Einrichtungen ca. 4500 rechtlich selbständige Pfarrgemeinden
KRZ.DaviP-mobile	DaviP-online auch für unterwegs (Prototyp, Zukunftsvision)

Pfarramt benötigt Informationen

- **Spezielle kirchliche Aspekte**
 - **Wen kann ich zum Konfirmandenunterricht einladen?**
 - **Wer steht dieses Jahr zur Firmung an?**
- **Generelle meldepflichtige Informationen**
 - **Wer ist neu in die Gemeinde gezogen?**
 - **Wer hat im nächsten Monat einen runden Geburtstag?**
- **Bereitstellung des Zahlenmaterials für Finanz-, Gebäude- und Personalplanung**
- **Steuerung des Engagements der Kirchen in sozialen Handlungsfeldern**

=> neue, komplexere Anforderungen an die zur Verwaltung der Kirchengemeindeglieder verwendeten Softwaresysteme!

Mitgliederverwaltung für das kirchliche Meldewesen

- **Ökumenisch** für den gleichzeitigen Einsatz in Diözesen und Landeskirchen
- **Sicher** Zertifizierung durch den TÜV-Süd
- **Anpassbar** Abbildung der individuellen Organisationsstrukturen der Diözesen und Landeskirchen
- **Modular** für die Nutzung durch die unterschiedlichen Nutzergruppen
- **Einfach** ein gängiger Internetbrowser und eine sichere Verbindung genügen
- **Ökonomisch** gemeinsame Entwicklung und gemeinsame Nutzung steigern die Effizienz und senken die Kosten



Hummel, Björn, ev - 1207070701 Davipkirchen I

geb.: 16.07.1974, m, vh

Falkenweg 10 88888 Davipkirchen Altstadt (AW) 

[Speichern](#) [Schließen](#) [Ausgabe](#) [Weitere Aktionen](#)

▼ Gemeindemitglieder

[Personensuche](#)
[Eigenerfassung](#)
[Amtshandlungen](#)

▼ Auswertungen

[Assistent](#)
[Gespeicherte](#)
[Geburtstage](#)
[Differenzliste](#)
[Statistik](#)

▼ Strukturen

[Zielgruppen](#)
[Regionalstrukturen](#)
[Straßenübersicht](#)

▼ System

[Benutzer](#)
[Pfarrbezirke](#)
[Einstellungen](#)
[Protokoll](#)

Persönliche Daten | Wohnungsdaten | Amtshandlungen | Sonstige Daten

Name

Familienname*

Namenszusatz

Gebräuchl. Vorname*

Vornamen*

Geburtsname

Geburtsnamenszusatz

Früherer Name

Früh. Namenszusatz

Ordensname

Künstlername

Namen für die Ausgabe

Kurzvorname

Kurznachname

Familie

Familienstand*

Eheschließungsdatum

Geburtsdaten

Geburtsdatum*

Geburtsort

Geburtsland

Weitere Daten

Konfession

Staatsangehörigkeit

Geschlecht*

Beruf

Akad. Grad

Sterbedaten

Sterbedatum

Sterbeort

▼ Familienverband

Hummel, Björn
 (Stif 1 m, 39, ev)
 Geb. Datum 16.07.1974
 Taufe 01.09.1974
 Konfirmation 12.03.1989
 Trauung 02.08.1997

Hummel, Anja
 (Stif 2 w, 38, rk)
 Geb. Datum 14.10.1975
 Taufe 16.11.1975
 Firmung 23.06.1990
 Trauung 02.08.1997

Hummel, Nico
 (Stif 3 m, 14, rk)
 Geb. Datum 12.06.1999
 Taufe 01.08.1999
 Erstkomm. 13.04.2008

Hummel, Kay
 (Stif 3 m, 11, rk)
 Geb. Datum 19.12.2002
 Taufe 06.07.2003

[Ausgabe](#) | [Fam. suchen](#)

▼ Zielgruppen

- CHO Kirchenchor
- LIKR Liederkreis

[Bearbeiten](#)

- **Prinzipiell gemäß RUP (**Rational Unified Process**)**
- **Use-case-orientiertes Anforderungsmanagement**
 - Ca. 30 Use Cases für DaviP-online,
 - Ca. 50 Use Cases für DaviP-org, davon 16 Erweiterungen
 - Festgelegter Abnahme-/Freigabeprozess
- **Iterativ-inkrementelles Vorgehen**
 - 3-4 Hauptreleases, 2-3 Zwischenreleases pro Jahr
 - Neue Use Cases, neue Funktionalitäten für bestehende Use Cases
- **Tests gemäß **V-Modell****
 - Teststufen entsprechen Spezifikations- und Umsetzungsschritten

- **Strukturierung der Anforderungen in Use Cases**
 - Einzelanforderungen sind Use Cases zugeordnet
 - Gem. Alistair Cockburn ist ein Use Case die „Nabe“ im „Rad“ der Anforderungen
 - Einige wenige übergreifende Anforderungen
- **Anforderungen an Use Cases**
 - Textuelle Beschreibung, keine (UML-)Diagramme
 - Dokumentation in vorgeschriebenem Format
 - **Word**-Vorlagen, später standardisierte **Wiki**-Seiten (gem. Templates nach Alistair Cockburn)
 - **Nummerierung**
 - Der Anwendungsfälle selbst, z.B. „A20 Statistiken erstellen“
 - Der Aktionen in einem Anwendungsfall
 - Der detaillierten Anforderungen zu einer Aktion, z.B. „Anforderung Nr. A20.4: ...“

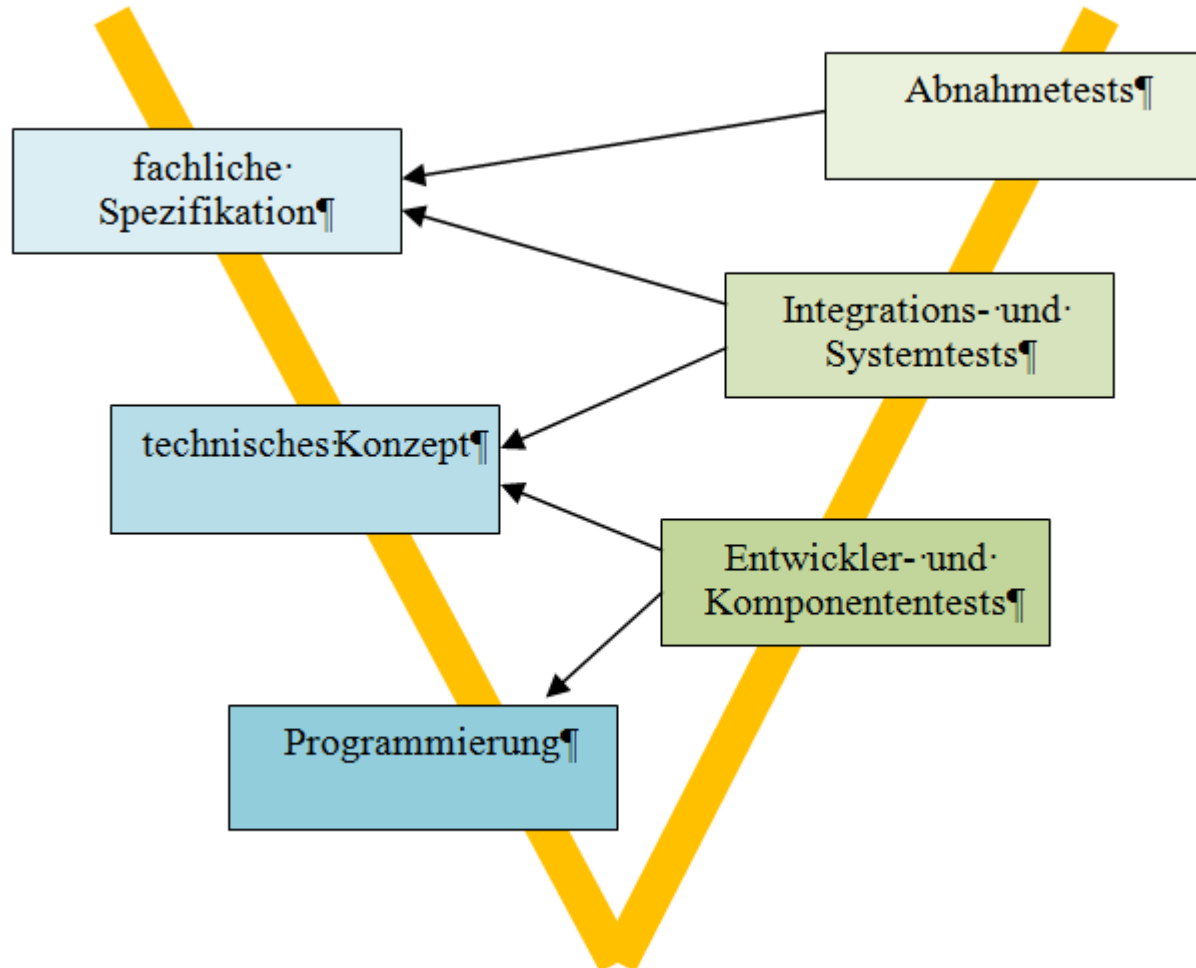
- **Struktur der Use Case Beschreibung**
 - **Überblick**
 - Tabellarisch gem. A. Cockburn
 - **Fachliche Anforderungen**
 - Alle Anforderungen und Nicht-Anforderungen
 - **Benutzeroberfläche**
 - inkl. Skizzen der Masken („Wireframes“)
 - **Technische Spezifikation**
 - von Entwicklern ergänzt (Schnittstellen, Datenbankschema)
 - **Testfallbeschreibung**
 - Testdaten, Testschritte, erwartete Ergebnisse

- **Abgleich mit „Use Cases effektiv erstellen“ von A. Cockburn**
 - **gleich:**
 - **Nummerierung, textuelle/tabellarische Beschreibung**
 - **Inkrementelles Vorgehen (3-4 Releases pro Jahr)**
 - **Möglichkeit, einen Use Case zuerst nur teilweise umzusetzen**
 - **Gruppierung von Use Cases (15-20 UC pro Gruppe)**
 - **unterschiedlich:**
 - **Use Cases unterschiedlicher Ebenen**
 - **Unterschiedliche Templates**
 - **Wichtigkeit der Akteure/Stakeholder**
 - **Bedeutung des GUI-Designs**

- **Sich positiv auswirkende Rahmenbedingungen**
 - **Stabile personelle Vertretung der Anwender**
 - „Arbeitskreis Meldewesen“
 - **Nachvollziehbarkeit von Änderungen und Beschlüssen**
 - Sitzungsprotokolle, Track+, Wiki-Historie
 - Gegenseitige Referenzierung
 - **Einsatz mehrerer Tools statt eines integrierten Tools**
 - Schneller Einarbeitung, flexiblere Anpassbarkeit
 - **Konsequentes Einhalten der Vorgaben notwendig**
 - Kürzere Diskussionen

- **Orientiert am ISTQB-Standard**
- **Berücksichtigung der Architektur**
 - **Webserver ↔ Mainframe**
- **Mehrstufige Tests/Testumgebungen**
 - **Entwicklungsumgebungen → Entwicklertests**
 - Lokale Komponententests
 - Übergreifende Tests (Continuous Integration)
 - **Integrationsumgebung → Integrations-/Systemtests**
 - Fachliche Tests durch internes Testteam
 - Auf dedizierten Versionsständen (SVN)
 - Regressionstests (Selenium, soapUI)
 - **Preproduction → Abnahmetests**
 - Fachliche Tests durch Anwender
 - Nachtest von Bug Fixes für Hot Fixes, Performancetests

- **Angepasstes, „versetztes“, allgemeines V-Modell**



- **Ableitung von Testfällen aus Use Cases**
 - **Eine Testfallbeschreibung/Test-Suite pro Use Case**
 - **Nummerierung der Test-Suites, Testfälle/-schritte**
 - **Anwendungsfall A01**
 - **Testfälle TC01**
 - **Testdaten TC01.csv**
 - **Dokumentation im Projekt-Wiki**
 - **Verlinkungen mit Anforderungsbeschreibungen und Testprotokollen**
 - **Vereinfacht Abgleich zwischen Anforderungen und Tests**

- **Anforderungsspezifikation anhand von Use Cases**
 - Intuitive Strukturierung von Anforderungen
 - Ideale Granularität der Artefakte
 - Zielgerichtete Diskussionen im Projekt
 - Schnelle und genaue Antworten auf Benutzeranfragen
 - **Voraussetzung für gute SW-Qualität**
- **Organisatorische Notwendigkeiten**
 - Kontinuierliches und konsequentes Anforderungsmanagement
 - Systematische und vollständige Dokumentation

- **Erfolgsfaktoren des Vorgehens in den DaviP-Projekten**
 - Strukturierung der Anforderungen als Use Cases
 - feste Vorlage zur Beschreibung von Use Cases und Tests
 - rein textuelle Dokumentation der Use Cases, keine UML-Diagramme
 - Iterativ-inkrementelles Vorgehen, auch für Use Case Beschreibungen
 - geregelte Review-Prozesse, Vermeidung vieler Review-Zyklen durch frühzeitiges Einbeziehen aller betroffenen Personen
 - gleichartige Prozesse und gleiche Tools für neue und geänderte Anforderungen (Anforderungs- und Änderungsmanagement)
 - Verwendung mehrerer, einfacher Tools mit gegenseitiger Referenzierung anstatt eines komplexen integrierten Tools
 - Dokumentation in einem (zentralen) Projekt-Wiki
 - systematische Nummerierung von Anwendungs- und Testfällen
 - stufenweises Testen, größtenteils automatisierte Regressionstests
 - Tests mit synthetischen Daten und Echtdaten



Alexander van der Vekens
freiberuflicher
Senior Software Consultant



avite
Zwickauer Straße 3/1, 71083 Herrenberg

Tel: +49 7032 956376
E-Mail: avekens@avite.de
Web: www.avite.de



Ursula Schmitt-Wagner
Abteilungsleitung
Kirchliches Meldewesen



Stiftung Kirchliches Rechenzentrum
Südwestdeutschland, Junkersring 10,
76344 Eggenstein-Leopoldshafen

Tel: +49 721 7086-207
E-Mail: U.Schmitt-Wagner@krz-swd.de
Web: www.krz-swd.de

Beschreibung des Anwendungsfalls

1. Allgemeine Angaben

Name	Person einer Zielgruppe zuordnen
Nummer	A17
Ziel	Die Zugehörigkeit einer einzelnen Person zu Zielgruppen soll geändert werden.
Anwendungskontext	Personen
Primärer Akteur	Pfarrbüro-Mitarbeiter
Sekundäre Akteure	keine
Betroffene (Stakeholder)	Gemeindemitglied (Datenschutz, gezielte Information)
Vorbedingungen	Eine Person ist ausgewählt.
Auslöser	Mitteilung über neue Zielgruppenzugehörigkeit(en)
Eingehende Inform.	Name(n) der betroffenen Zielgruppe(n)
Ergebnisse	Anzeige der neuen Zielgruppenzuordnung
Nachbed. Erfolgsfall	geänderte Zielgruppenzugehörigkeit gespeichert, Aktion protokolliert.
Nachbed. Fehlerfälle	Das System ist nach dem Abbruch in einem unveränderten Zustand, Fehler wurde geloggt.

2. üblicher Ablauf

Schritt #	Interaktionsschritt	benutzter Anw.-Fall
1	eine Person wird angezeigt.	A02
2	aktuelle Zuordnungen der Person zu Zielgruppen werden angezeigt	
3	Benutzer wählt eine neue Zielgruppe für die Zuordnung zu der Person aus	
4	Die neue Zuordnung wird bestätigt	

3. Alternative Abläufe

Ausnahmen/Fehlerfälle keine

Variationen:

Schritt #	Interaktionsschritt
3a	Eine Zuordnung zu einer Zielgruppe wird gelöscht

4. Bezug zu anderen Anwendungsfällen

A02	Daten eines Gemeindemitgliedes anzeigen (Voraussetzung)
-----	---

- **Abnahme/Freigabeprozess**
 - **Anforderung → Track+ → Vorabklärung
→ Use Case Beschreibung → internes Review
→ Review durch Anwender → Freigabe**
 - **Übergabe an Entwickler/Tester („Kickoff-Meeting“)**
 - **Besonderheiten bei übergreifenden Änderungen:**
 - **Review/Abnahme Konzeptsdokumentation**
 - **nach Freigabe Übernahme in Use Case Beschreibungen**

- **Besonderheiten in den DaviP-Projekten**
 - **Berücksichtigung der Verarbeitungsart**
 - Benutzer, Batch, Mischformen
 - **Testdaten**
 - Test mit **synthetischen Daten** → systematisch Abdeckung der fachlichen Anforderungen
 - Test mit **Echtdaten** → Extremfälle aus der Realität, Mengenproblematik