

Entwicklung eines mobilen Software-Assistenten zur Unterstützung der vernetzten Pflege

Master-Arbeit
von

Thomas Knapp

An der Fakultät für Wirtschaftswissenschaften
Institut für Informationswirtschaft und -management (IISM)
Information & Market Engineering

Gutachter:	Prof. Dr. rer. pol. Christof Weinhardt
Betreuender Assistent:	Dr. Henner Gimpel
Betreuende Mitarbeiter FZI:	Bruno Rosales Saurer
	Mathias Schmon
	Imanol Bernabeu

31. März 2012

Inhaltsverzeichnis

1. Einleitung	1
2. Ziele der Arbeit	2
3. Analyse der Anforderungen	3
4. Entwurfsentscheidungen zur Entwicklung der mobilen Anwendung unter Andro- id	4
5. Basisanwendung und Plug-Ins zur Realisierung der identifizierten Anforderun- gen	5
5.1. Aufgaben und Struktur der Basisanwendung	5
5.2. Einbinden von Plug-Ins in die Basisanwendung	5
5.3. Das Plug-In "Kontakte"	5
5.4. Das Plug-In "Touren"	6
5.5. Das Plug-In "Hilfe"	6
6. Evaluation	7
6.1. Software-Architektur sowie technische Umsetzung	7
6.2. Ergebnis der Akzeptanztests	7
6.3. Abbildung eines Prozesses aus der Praxis mit CareNet <i>mobile</i>	7
7. Zukünftige Entwicklungs- und Einsatzmöglichkeiten	8
8. Fazit	9
9. Erklärung	10
Anhang	11
A. First Appendix Section	11
Literaturverzeichnis	13

1. Einleitung

- Kontext mit Ärztemangel
- Als Lösungsansatz Delegation von Leistungen an qualifiziertes Pflegepersonal (wie werden MFPs ausgebildet???)
- Effizienzsteigerung
- Einbringen von Statistiken über Bevölkerungsentwicklung, Demographiewandel

Orientierung an Exposé

[Kop10]

dfgsdgsgsetgrtgstr

...

2. Ziele der Arbeit

- Ganz allgemein: Mobile, elektronische Dokumentation delegierter ärztlicher Tätigkeiten durch Angehörige der Alten- und Pflegeberufe.
- Entwicklung einer App zunächst für Android inkl. Serverkommunikation (Aufsetzen eines erweiterbaren Middleware-Ansatzes)
- Anpassung einer SaaS-Lösung, um den Anforderungen von CareNetmobile zu genügen (CareNet)
- Durchführung einer Anforderungsanalyse (Sammeln von Anforderungen, Priorisierung, Abschätzung was möglich ist)
- Implementierung der Kernanforderungen, um eine grundsätzliche Dokumentation zu ermöglichen
- Evaluation in mehreren Schritten:
 - Messen der Architektur an ISO-Software-Qualitätskriterien (vor allem Wartbarkeit) (Focus: Technische Umsetzung)
 - Schulung/Anforderungsevaluation mit Endnutzern (Pflegepersonal) (Focus: Usability/ Akzeptanztests)
 - Überprüfung, ob ein bestimmter Prozess (z.B. Pneumonie) mit der App abgebildet werden kann

3. Analyse der Anforderungen

- Wie wurden die Anforderungen erhoben?
- Welche Quellen gab es hierfür?
 - Richtlinie, die Delegation erlaubt (für grundsätzliche Tätigkeiten)
 - Gespräch mit Mitarbeiter eines Pflegedienstes (Raymond)
 - Anforderungen aus dem Projekt Vitabit (z.B. Schlüsselnummer)
 - Abgeleitete Anforderungen aus Wissen über Endnutzer (besonders Usability)
 - ...
- Kernanforderungen (Prio A)
- Wünschenswerte Eigenschaften, um Nutzungskomfort zu steigern (Prio B)
- Anforderungen, die über die reine Dokumentation oder dessen Unterstützung hinausgehen (Prio C, werden in der Arbeit nicht behandelt)

4. Entwurfsentscheidungen zur Entwicklung der mobilen Anwendung unter Android

- Kurze Erklärung der Kommunikation zwischen App und zentralem System (Care-Net). Erklärung, warum Middleware notwendig (Cross Domain Calling) und warum gewünscht (Erweiterung um Schnittstellen zu Krankenhäusern und Hausarztpraxen).
- Middleware
 - Technologie (PHP)
 - Wo wird Middleware positioniert? Welcher Server? (konzeptionell)
 - Kommunikation mit REST (Was ist es? Was sind die Vorteile? Wie wird Sicherheit hergestellt?)
- Technologie des Software-Assistenten
 - Warum überhaupt Tablet? (Vgl. mit Diplomarbeit der anderen Studentin auf Nokia-Handy)
 - Plattform (Android)
 - Nativ vs. hybrid
- Architektur (Plug-In-Struktur)
 - Warum Plug-In? Vorteile?
 - Argumentation mit Wartbarkeit (ISO-Richtlinie), Erwartete Änderungen der Anforderungen, Verweis auf Entwurfsmuster, Vgl.mit Eclipse als Praktische Implementierung

5. Basisanwendung und Plug-Ins zur Realisierung der identifizierten Anforderungen

Bis hier hin wurde Plug-In-Architektur vorgestellt. Jetzt muss konkret beschrieben werden, welche Eigenschaften der Gesamtarchitektur und der Plug-Ins welche Anforderungen erfüllen. Hierzu werden zunächst die Aufgaben der Basisanwendungen beschrieben (Bereitstellen einer grafischen Struktur, eines Systems zur Einbindung von Plug-Ins, Herstellen von Sicherheit mittels Log-In und Verwaltung der Kommunikation mit einem zentralen Server). Anschließend werden die wichtigsten Plug-Ins im Details beschrieben.

5.1. Aufgaben und Struktur der Basisanwendung

...

5.2. Einbinden von Plug-Ins in die Basisanwendung

...

- Namenskonventionen
- Plug-Struktur (Dateien)
- Vorgang des Einbindens
- Probleme, Erweiterungsmöglichkeiten

5.3. Das Plug-In ”Kontakte”

Für alle Plug-Ins feste Beschreibungsstruktur:

- Begründung für Plug-In (aus welchen Anforderungen geht Plug-In hervor)

- Aufbau/Navigationsstruktur
- Zentrale Frage, die beantwortet werden muss: Welche Funktionalität geht aus welcher Anforderung hervor?

...

5.4. Das Plug-In "Touren"

...

5.5. Das Plug-In "Hilfe"

...

6. Evaluation

6.1. Software-Architektur sowie technische Umsetzung

...

6.2. Ergebnis der Akzeptanztests

...

6.3. Abbildung eines Prozesses aus der Praxis mit CareNet *mobile*

...

7. Zukünftige Entwicklungs- und Einsatzmöglichkeiten

Ausblick auf mögliche Erweiterungen:

- Erweiterbarkeit des Ansatzes ausgrund von Plug-In-Struktur herausstellen
- Weitere Funktionen, um Komfort zu steigern (Prio C Anforderungen)
 - Schließen eines Falles beim Entfernen vom Einsatzort (damit keine überlangen Verweildauern gespeichert werden)
 - Login zu verschiedenen Instanzen von CareNet (Auswahl beim Login)
 - Unterschiedliche Verfügbarkeit von Plug-ins in Abhängigkeit des Rechten desjenigen, der sich anmeldet

8. Fazit

1. Kurzer Vergleich der nativen Entwicklung unter Android mit der hybriden Entwicklung (Wie hat sich die Entwicklung gestaltet?)
 - a) Typsicherheit
 - b) Namensräume (z.B. Gefahr von Überschneidung von Variablennamen und Methodennamen)
 - c) Unterstützung durch native Java-Bibliotheken vs. JavaScript-Bibliotheken
 - d) Einfachheit der Implementierung mit HTML/JavaScript
 - e) Vorteil der Portierbarkeit (Test wäre hier praktisch)

Erste Worte des Fazits. . . Und eine Referenz [Kop10]

Und noch eine [Pre06]

9. Erklärung

Ich versichere hiermit wahrheitsgemäß, die Arbeit selbständig verfasst und keine anderen als die angegebenen Quellen und Hilfsmittel benutzt, die wörtlich oder inhaltlich übernommenen Stellen als solche kenntlich gemacht und die Satzung des Karlsruher Instituts für Technologie (KIT) zur Sicherung guter wissenschaftlicher Praxis in der jeweils gültigen Fassung beachtet zu haben.

Karlsruhe, den 31. 03. 2012

Thomas Knapp

Anhang

A. First Appendix Section

ein Bild

Abbildung A.1.: A figure

...

Literaturverzeichnis

- [Kop10] D. T. Kopetsch, “Dem deutschen gesundheitswesen gehen die Ärzte aus! studie zur altersstruktur-und arztzahlentwicklung,” Bundesärztekammer und Kassenärztliche Bundesvereinigung, August 2010.
- [Pre06] S. B. Pressestelle, “Bevölkerung deutschlands bis 2050 - 11. koordinierte bevölkerungsvorausberechnung,” November 2006.