OPEN ACCESS

Originalarbeit

Bewältigung technischer und rechtlicher Hürden im mobile-health-Umfeld durch den Einsatz des Mobile-Assistance-Frameworks

Zusammenfassung

Systeme zur Übertragung von Vital-Parametern unter Zuhilfenahme von Smartphones unterliegen per se vielen externen, nicht trivialen Anforderungen. Smartphones sind durch die Akkukapazität und die verwendete CPU limitiert, die verbreiteten Betriebssysteme und APIs unterscheiden sich stark voneinander. Der Umstand eines nicht flächendeckenden Mobilfunknetzes setzt bei der Client-Applikation Offline-Funktionalitäten voraus. Zusätzlich bewegt man sich in einem Metier, welches geprägt ist durch das Medizinproduktegesetz, Datenschutzbestimmungen und Software-Lifecycle-Management. Im Artikel wird die Erfüllung von technischen Anforderungen durch den Einsatz des Mobile-Assistance-Frameworks gezeigt. Es wird der Begriff Off-the-Shelf-Komponente (OTS) erklärt, prominente OTS-Beispiele aufgezeigt und erklärt. Das Mobile-Assistance-Framework wird in die Kategorie OTS-Komponente eingruppiert und die positiven Auswirkungen auf die Zertifizierung nach dem MPG erklärt.

Thomas Huber[1]
Stefan Eckleder[1]
Hadi Maalouf[1]
Robert Diemer[2]

1 Fertl EDV Systeme GmbH, Eching

2 Lehrstuhl für Realzeit-Computersysteme, Technische Universität München, München

1 Einleitung

Mobile-Health-Systeme bestehen klassischerweise aus einer Client-Applikation auf einem Smartphone und einer Server-Komponente, welche Vitalparameter von der Client-Applikation entgegennimmt, diese Parameter aufbereitet und in einem Web-Interface darstellt. Im Forschungsprojekt CrossGeneration, gefördert vom Bundesministerium für Bildung und Forschung befasst sich das Konsortium mit der Entwicklung mikrosystemtechnisch-basierter Dienstleistungen zur Förderung der Lebensqualität und Gesundheit von Senioren in ihrem häuslichen und sozialen Umfeld. Um den Prozess der Entwicklung von Dienstleistungen zu flankieren, wurden rechtliche und technische Aspekte aufbereitet, Abhängigkeiten untereinander identifiziert, bewertet und Lösungsansätze erarbeitet. Bereits in früheren Veröffentlichungen [1] wurde der Stromverbrauch von mobilen Monitoring-Anwendungen gemessen. Muss eine Anwendung für sich einen hohen Stromverbrauch verbuchen, so wird dieser hohe Stromverbrauch und damit verbunden das häufige Aufladen des Smartphones schnell als störend empfunden. In der Veröffentlichung Huber et al. [2] wurde die Wichtigkeit der Offline-Fähigkeiten von Monitoring-Anwendungen anhand der Betrachtung eines elektronischen Asthmatagebuchs besonders hervorgehoben. Das Mobilfunknetz kann aus betriebswirtschaftlichen Gründen nicht flächendeckend realisiert werden. In Tiefgaragen, im Funkschatten von großen Gebäuden oder bei der Überlastung einer Funkzelle durch zu viele gleichzeitig angemeldete Mobiltelefone können plötzliche Verbindungsabrisse auftreten. Smartphone-Firmware-Versionen leiden unter schlechten UMTS-/GPRS-Treibern. Wird ein Smartphone von einer Funkzelle an eine andere übergeben (Hand-Over), kann es zum kurzzeitigen Verlust von Paketen kommen. In den aufgezählten Situationen werden unter Umständen Datenpakete abgesendet, ohne ordnungsgemäß von der empfangenden Instanz quittiert zu werden. Aus der Datenbank-Welt entstammt die Abkürzung ACID (atomicity, consistency, isolation, durability); bei der Datenübertragung zwischen Smartphone und Server muss ebenfalls eine ACID-ähnliche Datenübertragung gewährleistet werden. Um die Erstellung von Dienstleistungen für das Forschungsprojekt zu unterstützen wurde ein Framework zur **sicheren**, **energieeffizienten**, **bidirektionalen**, **Smartphone-Betriebssystem-unabhängigen** Datenreplikation zwischen Smartphone und Server erarbeitet. Unterstützungssysteme für ältere Personen können unter das MPG (Medizinproduktegesetz oder MDD Medical Device Directive) fallen. Die Praxis bei der Zertifizierung von Software-Systemen nach dem MPG hat sich in den vergangenen Jahren verändert. Softwareprodukte stützen sich verstärkt auf bereits vorhandene Komponenten, Betriebssysteme, Virtual Machines, Runtime Environments, Bibliotheken. Nach dem MPG müssen aber alle verwendeten oder selbstentwickelten Programmbestandteile für den Einsatz validiert und verifziert werden. Es muss bei der Entwicklung ein geeigneter Qualitätssicherungsprozess und Software-Lifecycle-Management-Prozess verwendet werden. In dieser Arbeit soll aufgezeigt werden, wie durch Komponentisierung und Verwendung von Standardkomponenten Komplexität aus der eigentlichen Applikation genommen werden kann. Im nächsten Schritt wird dargestellt, wie ein Gesamtsys-

tem und im Besonderen die darin beinhalteten Standardkomponenten, nach dem MPG, unter Anwendung der Norm IEC 62304 bzw. unter Anwendung einer FDA Guidance [3], zertifiziert werden können. Im Weiteren wird die Eingruppierung des Mobile-Assistance-Frameworks als OTS-Komponente bestätigt, um abschließend zu zeigen, wie Applikationen durch den Einsatz des Frameworks sowohl technisch (robuster, sicherer), als auch rechtlich (vereinfachter Zertifizierungsprozess) profitieren können.

2 Problemstellung

Da im Bereich der mobilen Gesundheitsapplikationen relativ schnell Prototypen erstellbar sind, diese aber auf Grund rechtlicher Auflagen unzertifiziert nicht als Produkte in den Verkehr gebracht werden dürfen, sollen besonders die rechtlichen und technischen Problemfelder und deren Verbindungen untereinander aufgezeigt werden.

2.1 Rechtliche Problemfelder

Gesundheitsapplikationen stehen durch gesetzliche Auflagen sehr viel stärker als z.B. Social-Networking-Applikationen in der Pflicht hohen Qualitätsstandards zu genügen. In Mauro et al. [4] wird mittels einiger Beispiele beschrieben, in welchen Fällen eine Mobile-Health-Applikation dem MPG unterliegt und in welchen Fällen nicht. Das BDSG (Bundesdatenschutzgesetz) gilt als weitere rechtliche Einflussgröße. Im § 3 Abs. 9 BDSG werden Vitalparameter als besondere Art der personenbezogenen Daten hervorgehoben [5]. Neben der Einwilligung für die Erhebung, gilt auch der Grundsatz der Datenvermeidung und Datensparsamkeit. Des Weiteren müssen die Daten dem Urheber eindeutig zuordenbar sein, die Übertragung verschlüsselt stattfinden. Die Integrität der Daten muss gewährleistet werden können (Vollständigkeit und Schutz vor Verfälschung); hieraus ergeben sich auch die bereits erwähnten ACID-ähnlichen Anforderungen an das System. Die hohen Datenschutzbestimmungen bei Gesundheitsapplikationen erfordern einen sehr sorgfältigen Umgang bei der Authentifizierung und dem autorisierten Zugriff auf und das Ablegen von Daten.

2.1 Technische Problemfelder

Die genannten Themen stehen nicht für sich alleine, sondern beeinflussen sich gegenseitig. Die Anforderungen des MPG und BDSG an die Robustheit der Übertragungsmechanismen erhöhen die Anforderungen an die verwendeten Protokolle, dies wiederum beeinflusst die Akku-Laufzeit der Smartphones. SOAP (Simple Object Access Protocol) steht exemplarisch für einen großen Overhead im Vergleich zum Anteil an Nutzdaten. Das Versenden von udp-Paketen (User Datagram Protocol-Paketen), wie es bei VoIP (Voice over IP) geschieht, steht als Beispiel für sehr schlanke Kommunikation. SOAP erweist sich als nicht optimal, da es die Größe der zu übertragenden Daten oft mehr als verdoppelt und dadurch die Akku-Laufzeit des Smartphones unter Umständen halbiert werden kann. Der VoIP-Lösungsweg ist nicht praktikabel, da nicht mitprotokolliert wird, welche Pakete tatsächlich beim Server angekommen sind. Ein automatisierter, bidirektionaler Abgleich von Daten, wie er bei einer optimistic/lazy Replication von Datenbeständen beschrieben wird, nimmt Komplexität aus der eigentlichen Anwendung und kapselt sie im Replikations-Modul. Der Client-Entwickler muss dafür Sorge tragen, die Daten richtig in den Storage-Objekten zu speichern.

3 Lösungsansätze

Das Replikations-Modul ist als Lazy-Optimistic-Master-Slave-Replication-Architektur implementiert. Bei einer Master-Slave-Replication gibt es einen Master-Node und mehrere Slave-Nodes. Die Slaves können ausschließlich mit dem Master kommunizieren, sich aber nicht untereinander abgleichen. Definitionsgemäß das Gegenstück hierzu ist die Master-Master-Replication oder Peer-2-Peer-Replication, hier kommunizieren die zu replizierenden Nodes gleichberechtigt miteinander. Als Eager Replication, das Gegenteil von Lazy Replication, bezeichnet man ein System, welches über alle Nodes hinweg von einem atomar synchronen Zustand in einen anderen atomar synchronen Zustand wechselt. Bei Eager Replication werden alle am Gesamtsystem angeschlossenen Nodes gleichzeitig in atomaren Operationen abgeglichen. Blockiert ein Node die Abarbeitung einer Operation wird bei allen Nodes diese Operation wieder rückgängig gemacht (Rollback). Da sich mobile Endgeräte außerhalb der Netzabdeckung befinden können, müssen Verfahren gewählt werden, welche nachgelagert Datenbestände synchronisieren können, wie es bei der Optimistic Replication beschrieben wird. Per Definition kann also das Replication-Framework kein Eager-Replication-System sein. Bei der Optimistic Replication wird offensiv mit der Möglichkeit von Konflikten umgegangen. Saito und Shapiro [6] befassen sich eingehend mit Optimistic Replication.
Beim Mobile-Assistance-Framework können an einen User-Account bis zu 998 mobile Endgeräte (Slaves) geknüpft werden, alle Geräte (Slaves) werden gleichrangig mit dem Master, also dem Server-Data-Store repliziert. Wird ein neuer Eintrag auf Gerät 1 erzeugt, weil ein Vitalparametersensor ausgelesen wurde, so wird dieser zunächst mit dem Server repliziert und von dort aus auf alle anderen mobilen Endgeräte repliziert. Das Smartphone-Replikationsmodul, das Server-Replikationsmodul und die Protokolle zum Austausch der Daten zwischen Smartphone und Server bewegen sich in einem Spannungsfeld aus

- Leichtgewichtigkeit um den Energieverbrauch im Griff zu behalten
- Konfliktauflösung um Konsistenz zu bewahren
- Zuverlässigkeit/Robustheit um den ACID-ähnlichen Anforderungen zu genügen

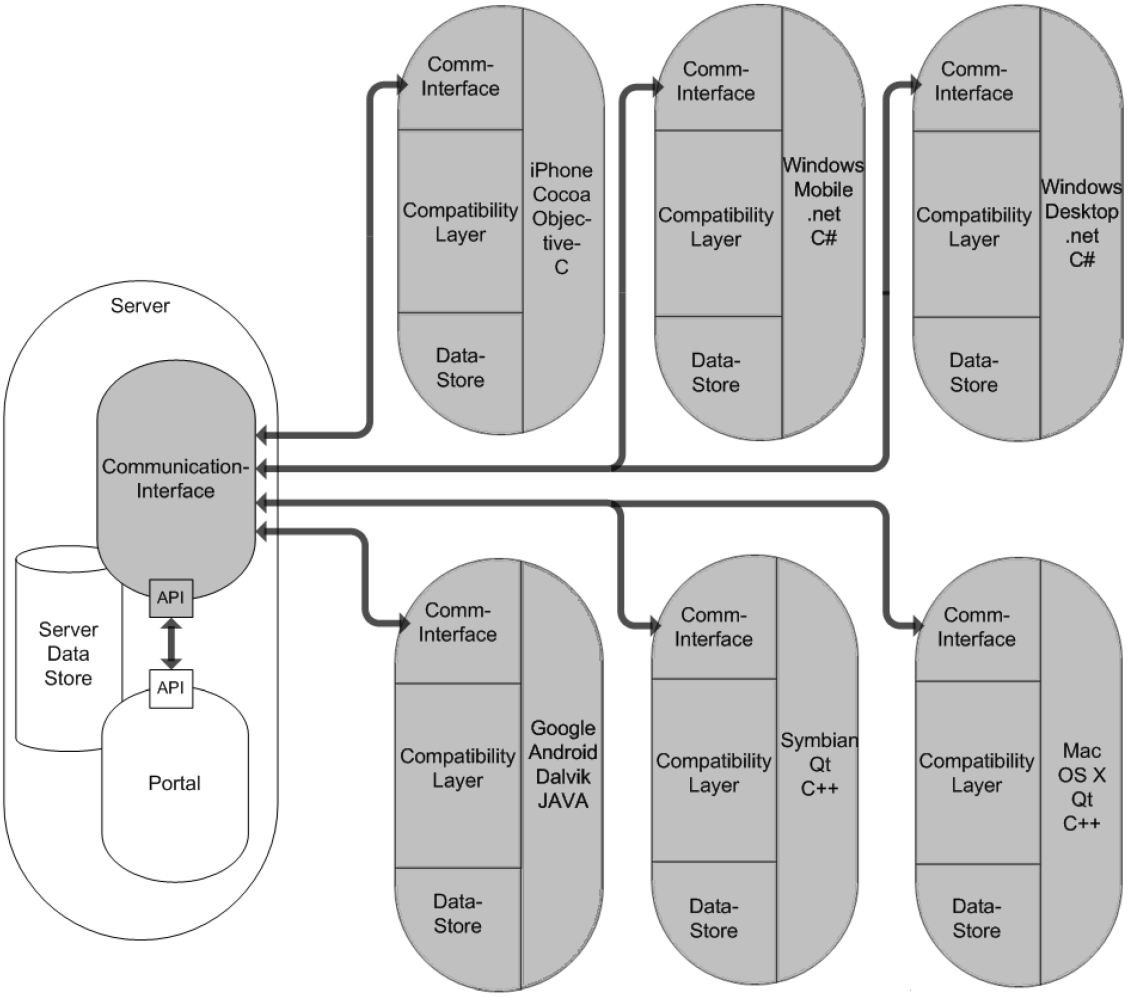

Abbildung 1: Entwicklung von Clients für unterschiedliche Zielsysteme

Ein User kann unterschiedliche Smartphone-Betriebssysteme in unterschiedlichen Kombinationen mit seinem Account verknüpfen. In Abbildung 1 wird beschrieben aus welchen Komponenten das Framework besteht. Die Datenrepräsentation einer Applikation kann innerhalb des Frameworks auf die unterstützten Plattformen portiert werden. Somit wird Komplexität aus der Applikation genommen und an das Framework abgegeben. Ein Design-Tool unterstützt bei der Erstellung der Datenrepräsentation und erstellt Rahmenprojekte, Stubs und Skeletons für die einzelnen Plattformen. Stubs und Skeletons sind Begrifflichkeiten aus der verteilten Anwendungsprogrammierung (RMI – Remote Method Invocation). In den Stubs und Skeletons findet das Umwandeln von Übergabeparametern in der lokalen Programmiersprache in Pakete des darunterliegenden Kommunikationsprotokolls (Marshalling) und das Zurückwandeln in einen Übergabeparameter für die entfernte Programmiersprache (Unmarshalling) statt. Die Replikationskomponenten befinden sich auf beiden Seiten, sowohl im Client-Communication-Interface (Stub) als auch im Server-Communication-Interface (Skeleton). Die Stubs und Skeletons bestehen aus einem generierten Source-Code-Teil und einer Konfigurationsdatei, die in die endgültige Applikation direkt hineinkompiliert wird. Das eben beschriebene System wird im Folgenden als (Closed-Source-)Komponente eines Gesamtsystems betrachtet, dieses System muss nach dem MPG zertifiziert werden.

3.1 OTS oder COTS (Commercial-off-the-shelf)

Da ein Medizinprodukt kritisch betrachtet nur dann zugelassen werden darf, wenn der gesamte Source-Code mit Hilfe eines Qualitätssicherungsprozesses entwickelt wurde, wurde zur Lockerung dieser Regel von der FDA (Food and Drug Association) ein Leitfaden erarbeitet [3], wie (Closed-Source-)Standard-Software-Komponenten zu behandeln sind, um sie in einem Medizinprodukt einsetzen zu dürfen. In der EN 62304 wird dieses Thema auf einen aktuelleren Sockel gestellt, innerhalb der EN 62304 wird von SOUP (Software of unknown Provenance) gesprochen.

Die Begriffe SOUP, OTS und COTS werden innerhalb der wichtigsten Normen weitgehend synonym verwendet. Um Standard-Komponenten in einem Medizinprodukt einsetzen zu können muss evaluiert werden, wie es sich im Gesamtsystem verhält und ob durch den Einsatz ein Risikopotenzial zu erwarten ist.

3.2 Ein Web-Portal/Web-CMS (Content-Management-System bzw. WCMS – Web-Content-Management-System) als Beispiel für COTS-Software

In anderen sicherheitskritischen Anwendungsumgebungen wie der Luftfahrt-Industrie dienen Portal-Systeme als Frameworks für Projekt- und Dokumenten-Management-Systeme. Hat man z.B. eine Reihe von WCMS zur Auswahl, können diese nach den Metriken aus der ISO 9126, Functionality, Reliability, Usability, Efficiency, Maintainability und Portability bewertet werden. Durch diese formalisierte Bewertung und der Auswahl des am besten geeigneten WCMS wird bereits ein Punkt bei der Validierung einer COTS-Software erfüllt. Durch die Nutzung der vom Web-Portal exponierten APIs und der Best-Practice-Examples können WebParts/Portlets/Module in die Portale integriert, die exponierten Infrastruktur-/Sicherheitsmechanismen instrumentalisiert werden. Portlets sind Module in einem Portal, z.B. besitzen viele Portale das Modul „Forum“. Es können also die im Portal registrierten User auf die Forums-Funktionen zugreifen. Durch die Eingabe von Login und Passwort authentifiziert sich der Benutzer am Portal und kann die erhaltenen Credentials für die Portlets als Single-Sign-On-Mechanismus verwenden. Das Forum kann sich der Permissions-API des Portals bedienen und private Bereiche für einzelne Gruppen implementieren. Beim Einsatz eines geeigneten Web-Portals vererben sich bei der Implementierung von Portlets/WebParts Fähigkeiten wie Robustheit, Desaster Recovery, High Availability, Sicherheit, Installierbarkeit, Konformität, Koexistenz, Änderbarkeit (Updateability) auf diese Portlets/Webparts. Z.B. stellen Web-Portale Sicherheitsmechanismen wie das Blocken von DDoS-Attacken (Distributed-Denial-of-Service-Attacken) ihren Portlets zur Verfügung. Bei konformer Programmierung erbt das WebPart/Portlet diesen Sicherheitsmechanismus. Des Weiteren dient das Portal als Rahmen, die Navigations-Menüs können beliebig verändert werden. Ober- und Unterpunkte im Navigationsmenü werden je nach Berechtigungen ein- und ausgeblendet. Gartner berichtet über die Entwicklung im Portal- und Web-Content-Management-Markt [7], [8], vergleicht die einzelnen Produkte an Hand unterschiedlicher Kriterien und erstellt ein Magic-Quadrant-Rating. Die Übergänge von Portal-Systemen zu Web-Content-Management-Systemen sind fließend, die Einsatzgebiete lassen sich nicht scharf voneinander abgrenzen. Das Rating von Gartner kann bereits als Teil einer ISO 9126-Bewertung genutzt werden, da sich viele Metriken in den Dokumenten wiederfinden.

Mit der Benutzung eines Portals werden praxisbewährte Rollen in das System integriert. Der administrative Account wird anders als ein Patient, Arzt, Call-Center-Mitarbeiter oder Krankenpfleger behandelt. Ein Nutzer kann Kombinationen aus Rollen bekleiden. Ein Patient kann gleichzeitig am Asthma-, Adipositas- und am Diabetes-Dienst teilnehmen. Sind Benutzer nicht für den Dienst registriert können sie im Web-Interface weder auf die Ansicht zugreifen noch Daten über die Replikationsmodule übertragen. Das Replikationsmodul bedient sich der Authentifizierungs- und Autorisierungsmechanismen, die durch das Portal exponiert werden (vgl. Abbildung 1). Das Mobile-Assistance-Framework instrumentalisiert das WCMS als Gate-Keeper und stützt sich damit auf eine COTS-Komponente, ist gleichzeitig selbst eine COTS-Komponente.

3.3 Bewertung von COTS-Software nach den Vorgaben der FDA

Weitere typische Beispiele für COTS-Software sind Webserver, Datenbank, Betriebssystem und Tabellenkalkulation. Soll eine Datenbank für ein Medizinprodukt eingesetzt werden, so muss dokumentiert werden, dass die Datenbank den Anforderungen genügt. Da aber die Datenbank als Black-Box betrachtet werden muss, werden Bewertungskriterien gegen das System gespiegelt, um die Gebrauchstauglichkeit darlegen zu können. Im Leitfaden [3] und in der IEC 62304 wird darauf eingegangen, wie COTS-Software bewertet werden muss um als Building Block für ein Medizinprodukt zugelassen zu werden. Es wird unterschieden zwischen Minor, Moderate und Major LOC (Level of Concern). Analog dazu ist in der IEC 62304 die Rede von den Software-Sicherheitsklassen A (Minor), B (Moderate) und C (Major). Je nach LOC sind andere Maßnahmen zu treffen. Wird eine COTS-Komponente nach der Risikoanalyse als Minor LOC eingestuft, muss eine Basis-Dokumentation, bestehend aus einer Beschreibung der COTS-Komponente (Hersteller Name, Major Version, Minor Version, Release Date und weitere Merkmale), Computer System Spezifikation und weitere Merkmale, angefertigt werden. Wird ein Moderate oder Major LOC ermittelt, können Gegenmaßnahmen ergriffen werden, um Schadenspotentiale zu verringern. Gegenmaßnahmen können aus Design-Maßnahmen, schützenden Vorkehrungen oder Warnhinweisen bestehen. Wird durch den Abschwächungsschritt der LOC auf Minor gesenkt, kann das COTS im Produkt eingesetzt werden, ansonsten kann mehrfach durch den Schritt des Abschwächens iteriert werden. Weitere Details über die Dokumentationspflichten finden sich in dem Leitfaden [3]. Datenbanken, Web-Server, Portal-Systeme sind in der Informatik durch die weite Verbreitung und dem Einsatz in kritischen Umgebungen ähnlich akzeptiert wie der Einsatz von integrierten Bauteilen (Integrated Circuits), wie z.B. Operationsverstärker oder AD-/DA-Wandler in elektrischen Schaltungen von Medizinprodukten. Auch in diesen Anwendungsfällen muss an Hand des Datenblatts und eigener Messungen bewiesen werden, dass der Betriebsbereich des jeweiligen Bauteils innerhalb der Spezifikationen liegt. Entwickler greifen in kritischen Schaltungen zu Bauelementen, die nach z.B. Militärstandards engere Toleranzen erfüllen, oder in einem größeren thermischen Bereich betrieben werden können. Analog dazu wird COTS-Software bevorzugt verwendet, welche bereits in

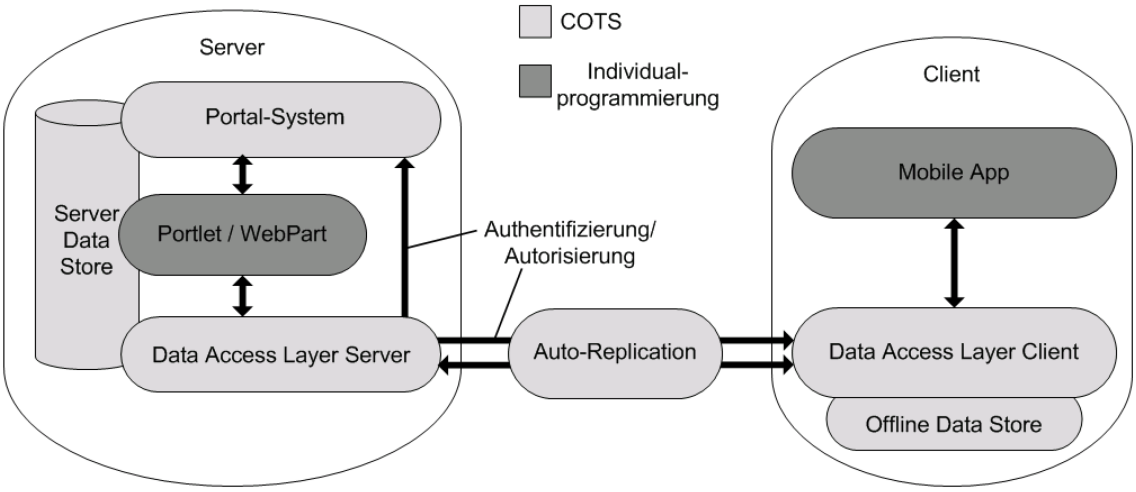

Abbildung 2: Anteil Commercial-off-the-Shelf – Individual-Programmierung

geschäftskritischen Anwendungsfällen ihre Tauglichkeit bewiesen hat.

Wie im Abschnitt 3 betrachtet sind Replikationsmechanismen wissenschaftlich gut fundiert. In verteilten Datenbank-Anwendungen sind sie als Standardkomponenten etabliert. Bisher wurden aber Best-Practices aus dem Datenbank-Themenfeld nicht für die Vereinfachung des Smartphone-Applikations-Entwicklungsprozesses genutzt. Das Mobile-Assistance-Framework verbindet diese beiden Themenfelder. Durch die Entstehung der neuen Kategorie „mobiles Smartphone-Replikations-Framework“ im weiten Feld der Standardkomponenten vereinfacht sich der Zertifizierungsprozess von mobilen Gesundheitsanwendungen.

4 Ergebnisse

Das Mobile-Assistance-Framework erfüllt die Rahmenbedingungen, um als COTS-Software in einem Medizinprodukt eingesetzt werden zu können. Der COTS-Hersteller kann den Validierungsprozess eines Medizinprodukts unterstützen in dem eine FMEA-Analyse der COTS-Komponente durchgeführt wird und nachgewiesen wird, dass ein Qualitätssicherungs-Prozess bei der Entwicklung verwendet wurde. In der ISO 13485 wird das V-Modell als ein geeignetes Vergehensmodell zur Entwicklung von Medizinprodukten genannt. In [9] werden weitere Kriterien bei der Auswahl der COTS-Komponenten genannt. Dokumente, die während des Entwicklungsprozesses der COTS-Komponente entstanden sind, können dem Auditingprozess beigesteuert werden. Diese Dokumente, bei der FDA unter dem Begriff Device Master Record zusammengefasst, können vom COTS-Softwarehersteller exklusiv an den Auditor ausgehändigt werden. Hiermit wird vermieden, dass tiefer greifendes Firmen-Know-How des COTS-Herstellers unnötig an Lizenznehmer weitergegeben werden muss. Um die hohen Anforderungen an ein Authentifizierungs- und Autorisierungssystem stemmen zu können wird von einer Eigenentwicklung innerhalb des Mobile-Assistance-Frameworks abgesehen. An Stelle dessen wird das Daten-Replikations-System mit dem Authentifizierungs- und Rechte-System eines ausgereiften Portals wie Sharepoint, IBM Websphere Portal Server oder einem Open-Source-Web-Portal-System verknüpft. Die COTS-Komponente Mobile-Assistance-Framework instrumentalisiert die COTS-Komponente Web-Portal.

In Abbildung 2 wird veranschaulicht wie gering der Anteil der Individualprogrammierung werden kann, wenn das Replikations-Framework als COTS-Komponente im Gesamtsystem Verwendung findet. Nicht nur die Verkürzung des Entwicklungsprozesses, sondern auch die nachhaltige Weiterentwicklung der Standardkomponente Mobile-Assistance-Framework führt zu einer steten Reifung derartiger Applikationen und führt in den weiteren Schritten auch dazu die Zertifizierung von Mobile-Health-Applikationen zu vereinfachen.

In weiteren Zertifizierungen vereinfacht sich der Nachweis der Gebrauchstauglichkeit, da er bereits bei der ersten Zertifizierung erbracht werden konnte. Nur die geänderten Rahmenbedingung (der Intended Use) muss erneut detailliert beleuchtet werden, nicht aber die COTS-Komponente an sich.

Technische Hürden werden demnach abgebaut durch die Vereinfachung des Entwicklungsprozesses und die robustere Auslegung der entstehenden Applikationen.

Gesundheitsapplikationen unterliegen schnell dem Medizinproduktegesetz; bei Fehlverhalten mit gesundheitsschädlichen Folgen kann der Anbieter mit empfindlichen Strafen belegt werden. Unterliegen Gesundheitsapplikationen dem MPG und wurden zertifiziert, sieht die Rechtsprechung keine bzw. kleinere Strafen bei Fehlverhalten vor als bei nicht zertifizierten Applikationen. **Rechtliche Hürden** werden abgebaut durch die Vereinfachung des Zertifizierungsprozesses.

5 Fazit

Mobile Gesundheitsanwendungen werden von vielen Anforderungen flankiert. Das Client-Server-Modell, der occasionally connected Charakter, die rechtlichen Einflüsse durch Bundesdatenschutzgesetz, Telemediengesetz und Medizinproduktegesetz, der technische Einfluss, bedingt durch einen fragmentierten Smartphone-Betriebssystem-Markt, durch CPU-, RAM- und Akku-limitierte Systeme, all diese Einflüsse im Paket erfordern neue Building Blocks um Komplexität in unterschiedlichen Dimensionen bewältigbar zu machen. Ein Replication-Framework wurde erstellt um viele dieser Anforderungen innerhalb einer Komponente zu adressieren und sich häufig wiederholende Tätigkeiten durch Tools zu automatisieren. Das System erbt die Authentisierungs- und Berechtigungsmechanismen aus dem verwendeten Portal- und Web-Content-Management-System. Das Gesamtsystem wurde unter der Annahme, dass es bei einer Zertifizierung unter der Rubrik Commercial-Off-the-Shelf-Software eingeordnet wird, gegen das MPG gespiegelt. Es wurden geeignete Schritte erklärt, um eine Closed-Source-Komponente für die Validierung aufzubereiten. Gleichzeitig wurde der Vorteil der klaren Kapselung der Daten von der Applikationslogik erörtert und die damit einhergehende Verkürzung des Entwicklungsaufwands. In zukünftigen Schritten wird das Framework für die Übertragung besonders großer Datenpakete erweitert. Hierfür muss das segmentweise Übertragen von BLOBs (Binary Large Objects) implementiert werden. Als weiteren wichtigen Schritt wird die Software-Komponente Mobile-Assistance-Framework auf Dienstleistungs-Beine gestellt. Hierfür müssen Anforderungen des Service-Consumers an den Service-Provider formuliert werden. Der Server soll als PaaS (Platform as a Service) an den Service-Consumer vermietet werden, dieser wiederum kann mobile-health-Anwendungen oder auch mobile Logistikanwendungen mit einer derartigen Plattform realisieren.

Anmerkung

Interessenkonflikte

Die Autoren erklären, dass sie keine Interessenkonflikte in Zusammenhang mit diesem Artikel haben.

Literatur

1. Huber T, Kreuzer J, Diemer R. Mobiles Monitoring: Energiebedarf von Sensoren und Smartphone für die Verarbeitung und Übertragung relevanter Daten auf einen Server. In: Leimeister JM, et al., Hrsg. Proceedings zum 7. Workshop der GMDS-Arbeitsgruppe Mobiles Computing in der Medizin, Augsburg, 20. September 2007. Aachen: Shaker Verlag; 2007. S. 94-104.
2. Huber T, Kreuzer J, Diemer R. Telemedizin in der Sekundärprävention: Ein Feldversuch mit asthmakranken Jugendlichen. GMS Med Inform Biom Epidemiol. 2008;4(3):Doc15. Available from: http://www.egms.de/en/journals/mibe/2008-4/mibe000074.shtml
3. U.S. Department Of Health And Human Services, Food and Drug Administration, Center for Devices and Radiological Health, Hrsg. Guidance for Industry, FDA Reviewers and Compliance on Off-The-Shelf Software Use Medical Devices. 1999 [zitiert 7. Juni 2010]. Available from: http://www.fda.gov/medicaldevices/deviceregulationandguidance/guidancedocuments/ucm073778.htm
4. Mauro C, et al. Mobile Anwendungen im Kontext des Medizinproduktegesetzes. In: Eymann T, et al., Hrsg. Proceedings zum 9. Workshop der GI- und GMDS-Arbeitsgruppe Mobile Informationstechnologie in der Medizin, Lübeck, 29. September 2009. Aachen: Shaker Verlag; 2009. S. 101-113.
5. Bultmann M, et al. Datenschutz und Telemedizin - Anforderungen an Medizinnetze. In: Der Bayrische Landesbeauftragte für den Datenschutz. 2002 [zitiert 7. Juni 2010]. Available from: http://www.datenschutz-bayern.de/verwaltung/DatenschutzTelemedizin.pdf
6. Saito Y, Shapiro M. Optimistic Replication. ACM Computing Surveys (CSUR). 2005;37(1): 42-81. DOI: 10.1145/1057977.1057980
7. MacComascaigh M, et al. Magic Quadrant for Web Content Management. 2009 [zitiert 7. Juni 2010]. Available from: http://www.gartner.com/technology/media-products/reprints/oracle/article91/article91.html
8. Gootitz D, et al. Magic Quadrant for Horizontal Portals. 2009 [zitiert 7. Juni 2010]. Available from: http://www.gartner.com/technology/media-products/reprints/oracle/article95/article95.html
9. Chojnowski B. COTS Software: A Broader Picture. Medical Device and Diagnostic Industry. 2009;31(10) [zitiert 7. Juni 2010]. Available from: http://www.mddionline.com/article/cots-software-broader-picture

Korrespondenzadresse:
Thomas Huber
Fertl EDV Systeme GmbH, Bichlmannstr. 4, 84174 Eching
thomas_huber@fertledv.de

Bitte zitieren als
Huber T, Eckleder S, Maalouf H, Diemer R. Bewältigung technischer und rechtlicher Hürden im mobile-health-Umfeld durch den Einsatz des Mobile-Assistance-Frameworks. GMS Med Inform Biom Epidemiol. 2011;7(1):Doc03.
DOI: 10.3205/mibe000117, URN: urn:nbn:de:0183-mibe0001178

Artikel online frei zugänglich unter
http://www.egms.de/en/journals/mibe/2011-7/mibe000117.shtml

Veröffentlicht: *17.10.2011*