

PLATTFORMUNABHÄNGIGE ENTWICKLUNG MOBILER ANWENDUNGEN MIT AUGMENTED REALITY-FUNKTIONALITÄT – EIN PROJEKTBERICHT

Michael Guckert
Cornelius Malerczyk
René Gerlach

Gabriele Taentzer
Steffen Vaupel

Michael Fatum

Technische Hochschule Mittelhessen
KITE – Kompetenzzentrum für
Informationstechnologie
Wilhelm-Leuschner-Straße 13
61169 Friedberg

Philipps-Universität Marburg
Hans-Meerwein-Straße
35032 Marburg

advenco Consulting GmbH
Plockstraße 14
35390 Gießen

ABSTRACT

Erfolgreiche mobile Applikationen müssen auf allen relevanten Plattformen verfügbar sein. Dies führt häufig zu zeit- und kostenintensiven Mehrfachentwicklungen. Modellgetriebene Softwareentwicklung unterstützt einen „Model-Once-Run-Anywhere“-Ansatz und kann diese Entwicklungsaufwände deutlich reduzieren, da Apps damit unabhängig von der Zielumgebung erstellt werden können. Ein neues rollenbasiertes Nutzungskonzept erhöht die Flexibilität des Einsatzes noch weiter. Zusammen mit der Integration von Augmented-Reality (AR)-Funktionalität können so effizient mobile Anwendungen mit innovativer Benutzerinteraktion realisiert werden. Die Möglichkeiten werden anhand mehrerer modellgetriebenen entwickelter Beispiel-Anwendungen vorgestellt.

EINLEITUNG

Der Einsatz mobiler Geräte verschiedenster Ausprägung durchdringt die Abläufe des täglichen Lebens in immer stärkerem Maß. Dies gilt sowohl für das Arbeitsleben als auch für den Freizeitbereich. Die Nutzung dieser Systeme geht dabei weit über Anwendungen mit klassischer Mensch-Maschine-Interaktion (Tastatur- oder Mauseingaben) hinaus: Anwendungen mit automatischer Lokalisierung des Nutzers oder Bild- und Tonerkennung eröffnen neue Möglichkeiten. Die aktuelle Smartphone-Landschaft wird von den Betriebssystemen iOS (Apple) und Android geprägt. Weitere Systeme, wie Windows Mobile, Firefox OS, und weitere ringen um Marktanteile (Statista 2014). Insgesamt ist daher sicherlich weiterhin mit einem heterogenen Markt zu rechnen. Noch immer wird Software meist spezifisch für einzelne Zielplattformen entwickelt und dann ggf. portiert. Zwar gibt es Produkte wie das Framework PhoneGap (PhoneGap 2014) oder Tabris (Tabris 2014), die mit Hilfe von HTML5 bzw. serverbasierten Techniken Multiplattformfähigkeit erreichen. Diese Ansätze führen aber zu Einschränkungen in der Universalität der Programmierung bei der Nutzung

gerätespezifischer Komponenten bzw. sind keine rein client-basierten Apps im eigentlichen Sinne. Ziel soll es sein, nur einmal zu spezifizieren und dann auf allen Plattformen native Apps zur Verfügung stellen zu können. Modellgetriebene Softwareentwicklung arbeitet mit domänenspezifischen Modellierungstechniken, die komplett von der konkreten Implementierung abstrahieren. Für ein konkretes Softwaresystem wird ein domänenspezifisches Modell erstellt, aus dem der größte Teil des Codes generiert werden kann. Dieses inzwischen etablierte Vorgehen kann den Softwareentwicklungsprozess erheblich verkürzen.

Im von der Hessen Agentur geförderten LOEWE-3 Projekt (gefördert unter der Projektnummer 355/12-45) PIMAR (Platform Independent Mobile Augmented Reality) wurde eine Infrastruktur für die modellgetriebene Entwicklung von datengetriebenen, AR-unterstützten Applikationen entwickelt.

DIE LÖSUNG

Modellgetriebene Entwicklung mobiler Apps

Dem DRY-Prinzip („Don’t repeat yourself“) folgend reduziert die modellgetriebene Softwareentwicklung die Notwendigkeit sich wiederholender Codesequenzen radikal (Stahl et al. 2007). Die Entwicklung eines Systems wird von der konkreten Programmiersprache und der Zielplattform abstrahiert und erfolgt mit Hilfe geeigneter Modellierungssprachen und Werkzeugen losgelöst von der konkreten Implementierung. Eine effiziente Infrastruktur kann den gesamten Entwicklungsprozess immens beschleunigen. Die Entwickler können sich auf die Modellierung der anwendungsspezifischen Aspekte wie Datenstrukturen, Prozesse mit ihrer Ablauflogik und der Gestaltung der GUI konzentrieren und dabei technische Details vernachlässigen. Gerade im Bereich der mobilen Anwendungen mit ihren sehr kurzen Innovationszyklen ist ein solcher Ansatz besonders vielversprechend. Im Besonderen können auf diese Weise die in der Eingangsdiskussion genannten notwendigen Mehrfachentwicklungen für mehrere Plattformen entfallen bzw. auf ein äußerstes Minimum reduziert werden.

Die vom Projekt bearbeitete Domäne umfasst datengetriebene Apps, für die generische Bausteine zur Verfügung gestellt werden, die dann von Experten weiter verfeinert werden können. Standardfunktionalität wie die CRUD-Prozesse (Create, Read, Update, Delete) können dabei komplett vom Generator bereitgestellt werden. Ein typisches Anwendungsszenario ist eine App, die in Museen oder generell im Bereich Tourismus eingesetzt werden kann. Hier werden Exponate bzw. Sehenswürdigkeiten beschrieben und eventuell zusätzliche Funktionalität (z.B. Links zum Ticketverkauf) zur Verfügung gestellt. Die grundsätzlichen Prozessabläufe in der App bleiben dabei immer gleich und hängen nur im Detail von den darzustellenden Inhalten ab. Dabei kann die App durch Änderung dieser Inhalte auf die jeweilige Anwendungssituation angepasst werden. In Konsequenz daraus sieht das PIMAR-Konzept die Möglichkeit einer rollenbasierten Bereitstellung vor: neben dem klassischen App-Entwickler, der (hier) das grundsätzliche Modell der App mit Hilfe der vom Infrastrukturentwickler verfügbar gemachten Infrastruktur spezifiziert und dem Endbenutzer, der die App bei seinem Museumsbesuch einsetzt, gibt es zusätzlich einen Administrator, der Inhalte für die App erzeugt und sie damit für den Endbenutzer konfiguriert (Vaupel et al. 2014). Der Administrator stellt ein sogenanntes Provider-Modell zur Verfügung. Damit können Inhalte leicht aktuell gehalten werden, was gerade in dieser Domäne von hoher Wichtigkeit ist. Ein Austausch des Provider-Modells führt zu einer „neuen“ App, ohne dass eine Codeänderung mit erneuter Auslieferung über den PlayStore (Google) oder den AppStore (Apple) notwendig wird. Die eigentliche Nutzung der App kann dann ohne Internetverbindung erfolgen, da alle Informationen im Endgerät vorliegen. Das gilt auch für die AR-Funktionalität. Der Proof-of-Concept für dieses Vorgehen konnte mit einer Conference-Guide-App für die internationale Konferenz zu „Model-Driven Environments, Languages, and Systems“ 2014 in Valencia erbracht werden (MoDELS 2014).

Im industriellen Umfeld kann eine solche App zum Beispiel für die Definition von Wartungsrundgängen eingesetzt werden. AR-Funktionalität ermöglicht hier das automatische Erkennen von Maschinen. Prozesse, die im Rahmen der Wartung notwendig werden, können in einer solchen App definiert werden und dann direkt beim Wartungsrundgang ausgeführt werden. Auch hier bietet sich die Arbeit mit unterschiedlichen Rollen an. Während der Maschinenführer als Endbenutzer die App beim Rundgang einsetzt, stellt der Schichtführer das Providermodell mit den von ihm definierten Rundgängen bereit.

Die Produkte *key2guide* und *key2operate* des Industriepartners advenco Consulting bieten derartige Funktionalität in konventionellen Apps an. Diese dienen als Referenzanwendungen für die Entwicklung der PIMAR-Infrastruktur, die die modellgetriebene Entwicklung derartiger mobiler Anwendungen ermöglicht.

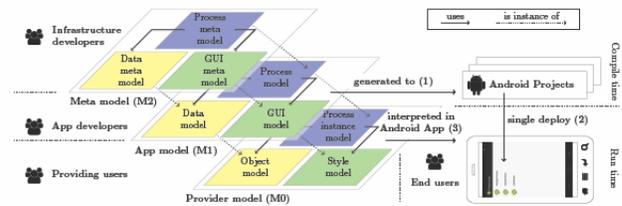


Abbildung 1: Modellierungsansatz für mobile Apps mit rollenbasierten Varianten

Die PIMAR-Entwicklungsinfrastruktur besteht aus einer Modellierungssprache, Editoren für diese Sprache und Generatoren für die Erzeugung nativer Apps für iOS und Android. Die Modellierungssprache lehnt sich stark an das Eclipse Modeling Framework (EMF) (EMF 2014) an. So wird für das Datenmodell direkt mit EMF-Modellen gearbeitet. Für die Definition der Benutzeroberfläche und unterliegenden Prozessen wurden jeweils eigene Teilsprachen entwickelt. Die Sprache für die Modellierung der Prozesse ist stark an die BPMN-Notation angelehnt. Für die resultierende domänenspezifische Modellierungssprache wurden ein textueller und ein grafischer Editor entwickelt, die einen intuitiven Modellierungsprozess sicherstellen. Das Projekt folgt einem „So abstrakt wie möglich, so konkret wie notwendig“-Ansatz. So ist es möglich, den kompletten CRUD-Prozess inklusive einer einfachen Oberfläche komplett auf Basis der Datenstrukturen generieren zu lassen. Wünscht man vom Standard abzuweichen, so kann man dies durch eine explizite Modellierung erreichen.

Details zur diesen Sprachen und zur gesamten Infrastruktur stehen auf der folgenden Webseite zur Verfügung (PIMAR Tools).

Bisher wurde die Infrastruktur mittels Prototypen für verschiedene Szenarien mit vielversprechenden Ergebnissen getestet (siehe Kapitel Beispiel-Apps). Dabei wurde auch die im Projekt entwickelte Augmented Reality-Funktionalität in die Softwaregenerierung integriert.

Augmented Reality

Unter Augmented Reality (kurz: AR, Erweiterte Realität) versteht man die sinnhafte Verbindung von realem Foto- oder Videomaterial mit künstlichen, am Computer erzeugten Objekten.

Die AR-Technologie hat sich in den letzten Jahren durch immer leistungsstärkere Smartphones auf dem Consumer-Markt immens verbreitet. Dadurch haben sich sowohl der Bekanntheitsgrad von Augmented Reality als Technologie als auch die Beliebtheit von mobilen Applikationen mit AR-Szenarien signifikant erhöht. Marktübliche Smartphones sind heute in der Lage, die Position und Orientierung der mobilen Kamera in Echtzeit und ohne Zeitverzögerung allein aus visuellen Parametern zu bestimmen und damit Objekte im Raum zu verfolgen und passgenau mit virtuellen Objekten zu überlagern. Nach der Definition von Augmented Reality nach Azuma (Azuma 1997) sollen dreidimensionale virtuelle Objekte dem zweidimensionalen Kamerabild überlagert werden. Sowohl die mathematischen als auch die technischen Grundlagen dafür sind in der Literatur bereits umfangreich

beschrieben (Adhani 2012). Der Markt bietet eine Vielzahl sowohl freier als auch kostenpflichtiger Softwarebibliotheken, die für die beiden Kernbereiche der AR verwendet werden können und die Basistechnologie sowohl für die Bilderkennung als auch das Rendering von dreidimensionalen Objekten als korrekte Überlagerung des Videostroms übernehmen können.



Abbildung 2: AR-Anwendung: Illustration (links), Tracking-3D-Punktwolke (Mitte) und virtuelle Überlagerung (rechts)

Allen Lösungen gemein ist jedoch, dass die 3D-Erkennung von Objekten in Echtzeit eine hohe Anzahl an Bildmerkmalen benötigt, die in dreidimensionale Punkte überführt werden müssen. Dies macht in der Regel eine Vorselektion des zu erkennenden Objektes notwendig. Der Anwender gibt manuell vor, welches Objekt oder welcher Raum erkannt werden soll, woraufhin das in einer Offline-Phase angelernete AR-Modell von der Applikation geladen und die entsprechende Objekterkennung- und -verfolgung gestartet wird.

Die im Projekt PIMAR entwickelten Anwendungsszenarien sollen aber ohne eine manuelle Vorselektion der zu erkennenden Objekte oder das heute übliche aktive Scannen von Markierungen, wie beispielsweise QR-Codes, arbeiten, da dies eine wesentlich intuitivere Handhabung der mobilen Anwendung erlaubt. So soll ein Museumsbesucher ein Kunstwerk oder der Wartungstechniker eine Maschine einfach mit der Kamera fokussieren, woraufhin das Objekt direkt erkannt werden soll.



Abbildung 3: Schaubild des AR-Verfahrens zur Klassifizierung und Klassierung

In Projekt PIMAR wurde deshalb das Verfahren TBOC (Transformation Based Object Classification) (Arlt und Malerczyk 2015) neu entwickelt, das die im Smartphone integrierte Sensorik mit Magnetometer, Accelerometer, GPS und Kamera verwendet, um bei gestarteter Applikation

automatisch die entsprechenden Objekte zu detektieren und damit in einem folgenden Schritt die notwendigen Modelldaten für die Objekterkennung und -verfolgung und für die 3D-Überlagerungen selektiv zu bestimmen. Es handelt sich um ein statistisch begründetes Verfahren, das Objekte durch Extraktion von geeigneten Sensorparametern als Merkmalsvektoren in einem Wahrscheinlichkeitsnetz klassifiziert und die gewonnenen Daten in einer Modelldatei speichert. Dieses Modell wird dann zur Klassierung von zu erkennenden Objekten verwendet. TBOC ist also in zwei Phasen unterteilt: In einer ersten Offline-Trainingsphase werden alle Objekte, die später zur Laufzeit der Anwendung durch den Benutzer erkannt werden sollen, automatisch sensorisch vermessen und die Trainingsergebnisse in einer Datenbank abgespeichert, die zusammen mit der Anwendung selbst auf dem Smartphone des Anwenders installiert wird. Diese Trainingsphase muss pro Objekt nur einmal durchgeführt werden. Die zweite Phase bedient sich des aus den Trainingsdaten gewonnenen statistischen Modells, um zur Laufzeit der Anwendung unter einer Vielzahl von Objekten das am wahrscheinlichsten relevante zu bestimmen und das entsprechende AR-Modell zu initiieren.

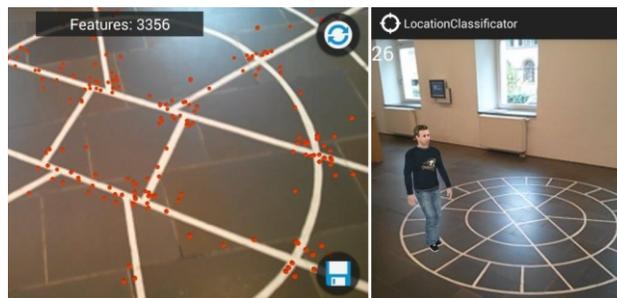


Abbildung 4: Tracking-Merkmale (links) zur AR-Überlagerung einer Animation in der Mathematikum-App

Für das entwickelte Verfahren konnten sowohl unter Laborbedingungen als auch für die im folgenden Abschnitt beschriebenen App für das Mathematikum (MATH 2014) vielversprechende Evaluationsergebnisse gewonnen werden. Trainingsphase und Modellbildung sowie die Erkennung von Objekten sind sehr performant und können auch auf älteren und damit etwas leistungsschwächeren Geräten durchgeführt werden. Die Erkennungsraten liegen bei über 90%.

Beide Phasen von TBOC sind als Standardszenarien in die oben beschriebene Modellierungssprache integriert und stehen somit für die modellgetriebene Entwicklung von mobilen Anwendungen zur Verfügung.

BEISPIEL-APPS

Im Laufe des Projektes wurden verschiedene Beispiel-Apps entwickelt, die typische Anwendungsfälle für den Einsatz der Infrastruktur zeigen. Im Projekt wurde ein inkrementelles Vorgehen verfolgt, bei dem die Werkzeuge schrittweise um weitere Funktionalität ergänzt wurden. Eine Integration der AR-Funktionalität erfolgte erst relativ spät im Projekt. Diese wurde zunächst in eigenständigen Apps entwickelt und dann

in standardisierten Funktionsblöcken (Anlernen und Erkennen - die beiden Phasen von TBOC) in die Infrastruktur – also in die Modellierungssprache und die Codegeneratoren – eingebunden.

Die folgend präsentierten Beispiel-Apps zeigen die weite Palette an Anwendungsszenarien, die mit der Infrastruktur PIMAR für modellgetriebene App-Entwicklung abgedeckt werden können.

PhoneBook (Verwaltung eines Telefonbuchs)

Die Anwendung *PhoneBook* (Download) wurde frühzeitig im Projekt entwickelt und in verschiedenen Workshops zur Schulung eingesetzt. Anhand von allgemeinverständlichen Anwendungsfällen (Kontakte anlegen, ändern, löschen, suchen, anrufen, etc.) werden die Funktionalitäten der modellgetriebenen Infrastruktur in einem Tageskurs an Interessenten vermittelt. Für die Anwendung gibt es insgesamt 12 Ausbaustufen, die dem systematischen Erlernen und Verwenden der modellgetriebenen Infrastruktur dienen. Aufgrund der vorhandenen Dokumentation und Materialien sind potentielle Nutzer des Werkzeuges auch in der Lage, sich selbstständig in die Erstellung von mobilen Anwendungen mit der modellgetriebenen Infrastruktur einzuarbeiten.

SmartPlug (Steuerung und Überwachung externer Geräte)

Die Anwendung *SmartPlug* erprobt die Integration und Ansteuerung externer Komponenten und Geräte außerhalb eines Smartphones. Sie kann zum Beispiel im Rahmen einer Hausautomatisierung genutzt werden. Konkret ist es derzeit mit *SmartPlug* möglich, schaltbare Steckdosenleisten (engl. Manageable Power Distribution Units - PDU) zu schalten. Der Mehrwert dieser Anwendung liegt in der differenzierten Aufzeichnung des Energieverbrauches und der Energiekosten einzelner Geräte.

Bildwörterbuch (E-Learning zum Einsatz im berufsbezogenen Spracherwerb)

Innerhalb des Projektes PIMAR konnten bereits Vorarbeiten für einen weiterführenden Einsatz der modellgetriebenen Infrastruktur erfolgen. Das BMBF-geförderte Projekt Alphamar2 des Fachbereichs Germanistik und Kulturwissenschaften der Philipps Universität Marburg (ALPHAMAR2 2014) arbeitet an der Entwicklung eines Bildwörterbuches d.h. einer mobilen E-Learning-Anwendung zum Einsatz im berufsbezogenen Spracherwerb von Migranten. Im Rahmen einer Bachelorarbeit wurde eine erste App entwickelt, die in einer Pilotstudie von Migranten getestet wird.

Conference Guide (Konferenzführer für die MoDELS-Konferenz 2014)

Der Conference Guide wurde den Teilnehmern der MoDELS-Konferenz 2014 (MoDELS 2014) in Valencia zur Verfügung gestellt, die diesen zur Orientierung während der Konferenz nutzen konnten. Den Benutzern dieser App ist es möglich, konferenzrelevante Inhalte (Sessions, Papers, Autoren, etc.) einzusehen. Der Conference Guide ist eine stark

datengetriebene App mit wenig Kontrollstrukturen und konnte mit dem Generator vollständig generiert werden. Die App wurde dabei in zwei Varianten generiert. Zum einen als Endanwender-App für die Konferenzteilnehmer zur Einsicht bestehender Daten (nur mit Leserecht) und zum anderen als Administrator-App für die Konferenzveranstalter zur Verwaltung aller Daten (mit Lese- und Schreibrecht). Die App steht samt einer Anleitung für den Einsatz zum Download zur Verfügung (ConfGuide 2014).

Mathematikum-App (Präsentation von Exponaten und Einbindung verschiedener Spiele)

Derzeit wird ein Prototyp einer App im Mathematikum in Gießen getestet. Diese App unterstützt den Besucher beim Verstehen einzelner Exponate durch augmentierte und interaktive Inhalte. Des Weiteren wird den Besuchern des Museums ermöglicht, einzelne Exponate interaktiv in Form von Spielen auf dem Smartphone nachzuvollziehen. Museums-Exponate werden dreidimensional im Raum verfolgt und mit animierten virtuellen Objekten erweitert.

FAZIT UND AUSBLICK

Die modellgetriebene Entwicklung rollenbasierter Apps konnte für verschiedene Anwendungsszenarien getestet werden und hat aus unserer Sicht ein hohes Potenzial für die Entwicklung qualitativ hochwertiger Apps. Arbeiten für die nähere Zukunft sehen die systematische Evaluation der Entwicklung von mobilen Anwendungen mit der PIMAR-Infrastruktur vor. Die Projektpartner beabsichtigen die industrielle Anwendbarkeit des Konzepts zu verstärken und wollen dies in einem umfassenderen Industrie4.0-Kontext weiterentwickeln.

LITERATUR

Adhani N.I. und Awang Rambli, D.R. 2012. "A Survey of Mobile Augmented Reality Applications". 1st International Conference on Future Trends in Computing and Communication Technologies

ALPHAMAR2. Online 1.12.2014
<http://www.uni-marburg.de/fb09/igs/arbeitsgruppen/daf/alphamar2>

Arlt, H; Malerczyk, C. 2015. Transformation Based Object Classification. Ein Verfahren zur automatischen Initialisierung für Augmented Reality-Anwendungen, Friedberger Hochschulschriften, ISBN 1439-1112, 2015

Azuma R. 1979. "A Survey of Augmented Reality". In: *Presence: Teleoperators and Virtual Environments*. 6, No. 4, 355–385

ConfGuide. Online 1.12.2014
<http://www.uni-marburg.de/fb12/swt/research/software/pimar3#conference-guide>

EMF. Online. 01.12.2014.
<https://projects.eclipse.org/projects/modeling.emf>

PIMAR Tools. Online 1.12.2014
www.uni-marburg.de/fb12/swt/forschung/software/pimar

- KITE. Online. 01.12.2014
<http://www.thm.de/site/kompetenzzentren/kite.html>
- MATH. Online. 01.12.2014
<http://www.mathematikum.de>
- MoDELS 2014. Online. 01.12.2014
<http://models2014.webs.upv.es>
- PhoneGap. Online. 01.12.2014
<http://phonegap.com>
- Stahl, T.; M. Völter; S. Efftinge und A.Haase. 2007.
“Modellgetriebene Softwareentwicklung”, Heidelberg
- Statista. Online. 13.1.2015
<http://de.statista.com/statistik/daten/studie/182363/umfrage/prognostizierte-marktanteile-bei-smartphone-betriebssystemen/>
- Tabris. Online. 01.12.2014.
<http://developer.eclipsesource.com/tabris>
- Vaupel, S.; G. Taentzer; J.P. Harries; R. Stroh; R. Gerlach und M. Guckert. 2014. “Model-driven development of mobile applications allowing role-driven variants”. In *Proceedings of the 2014 17th International Conference on Model-Driven Engineering Languages and Systems* (Valencia, Spain, Sep.28 – Oct.03). 1–17.